TEXTURE BASED CLASSIFICATION OF HIGH RESOLUTION REMOTELY SENSED IMAGERY USING WEBER LOCAL DESCRIPTOR

MR. DECKY ASPANDI LATIF

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE
DEGREE OF MASTER OF SCIENCE (COMPUTER ENGINEERING)
FACULTY OF ENGINEERING
KING MONGKUT'S UNIVERSITY OF TECHNOLOGY THONBURI
2013

Texture Based Classification Of High Resolution Remotely Sensed Imagery Using Weber Local Descriptor

Mr. Decky Aspandi Latif B.Sc. (Computer Science)

A Thesis Submitted in Partial Fulfillment
of the Requirements for
the Degree of Master of Science (Computer Engineering)
Faculty of Engineering
King Mongkut's University of Technology Thonburi
2013

Thesis Committee

……………………………………………… Chairman of Thesis Committee
(Muhammad Sumaryono, Ph.D.)

……………………………………………… Member and Thesis Advisor
(Lect. Sally E. Goldin, Ph.D.)

………………………………………………. Member
(Lect. Kurt. T. Rudahl, M.Sc.)

………………………………………………. Member
(Assoc. Prof. Suthep Madarasmi, Ph.D.)

………………………………………………. Member
(Preesan Rakwatin, Ph.D.)

Copyright reserved

Thesis Title	Texture Based Classification Of High Resolution Remotely Sensed Imagery Using Weber Local Descriptor
Thesis Credits	12
Candidate	Mr Decky Aspandi Latif
Thesis Advisor	Dr. Sally E. Goldin.
Program	Master of Science
Field of Study	Computer Engineering
Department	Computer Engineering
Faculty	Engineering
Academic Year	2013

Abstract

With the increasing use of high resolution images for remote sensing image analysis, texture has become a critical characteristic to be utilized in improving classification. This is due to limitations of conventional pixel-based classification with such images, in which regions that represent a single class often contain pixels with widely varying image values. In this study.

We apply the relatively new and robust Weber Local descriptor (WLD) to high resolution remote sensing through supervised Per-pixel texture-based classification approach, and compare its performance against well known Local Binary Pattern (LBP) algorithm. Furthermore, we will also try to incorporate Variance (VAR) texture descriptor on WLD with simple concatenation approach, and compare its capability over state of the art Local Binary Pattern Rotation Invariant Uniform Variance (LBPRIUVAR).

Two subsets of panchromatic Quickbird imagery containing several texture classes were selected as the study area. The parameter selection was conducted prior to the main classification process. We found that the best parameter values depended on characteristics of the different study areas. In the main experiment, we found that WLD is robust and also precise in classifying several texture classes, with the high accuracy results compared to the LBP texture descriptor and its rotation invariant version. In the contrast contribution upon WLD, we see a slight contribution when classifying an area with heavy illumination changes. This suggests the VAR descriptor can be combined with WLD in contrast affected areas. The results of WLDVAR outperformed the results of LBPRIUVAR overall. Given its high accuracy results, this texture descriptor is a promising choice in real world applications.

Keywords : Weber Local Descriptor / Texture-Based Classification / Binary Coding / Bhatacharya Distance

ACKNOWLEDGEMENTS

Upon my progress of finishing the master thesis, I received a lot of aid and support from the many people that surround my life enabling me to stand and finally able to complete this thesis. Firstly I would like to express my deep gratitude and thank to my first thesis advisor, Dr. Sally E. Goldin, for her warm support and substantial advice also suggestion in every step of my thesis development. It would be impossible for me to finish this thesis without her strong guidance and understanding of my capability. Second I also would like to thank Aj. Kurt T. Rudahl for his help, especially in the assistance on developing the software used in this research, and third is my Indonesian advisor, Dr. Sumaryono, for his strong commitment and suggestion for me, especially when I study in Indonesia. And the last is Dr. John Lukens, who helped me to learn the study area. I also thank the Geoinformatics and Space Technology Development Agency (GISTDA) for letting me to used the image data for this study.

I gratefully thank Assoc. Prof. Dr. Songrit Maneewongvatana and Dr. Erwin, for their substantial effort in managing the administration process between our university that allowed me to study in both in KMUTT and Mulawarman University, and also to Assoc. Prof. Dr. Trinaee Achalakul, who has helped me to meet my advisor at KMUTT. I also want to say thank you to Khun Tow, and Khun Nirma, for their individual help on assisting me in daily administration process on both study program. I also appreciate the support from all of my colleagues who faithfully accompany me, through my study in our Double Degree Program together.

Finally, I would like to offer my sincere gratitude to my family, who always enlighten my days with their unconditional care and affection toward me. Without their encouragement, support and motivation, I would not be able to reach this far.

CONTENTS

PAGE

ABSTRACT ii
ACKNOWLEDGEMENTS iii
CONTENTS iv
LIST OF TABLES vi
LIST OF FIGURES ix
LIST OF SYMBOLS xi
LIST OF TECHNICAL VOCABULARY AND ABBREVIATIONS xii

CHAPTER
1 INTRODUCTION 1
1.1 Overview of Problem 1
1.2 Introduction to Texture Classification 1
1.3 Objectives 3
1.4 Scope of The Study 3
1.5 Contribution 3
1.6 Organization of Research 3

2 BACKGROUND AND LITERATURE REVIEW 5
2.1 Texture 5
2.2 Survey of Prior Research in Texture Descriptors 5
2.3 Local Binary Pattern (LBP) 7
2.3.1 Rotation Invariant LBP (LBPRIU) 9
2.3.2 LBPRIU with Variance (LBPRIUVAR) 10
2.4 Weber Local Descriptor (WLD) 11
2.5 Texture-based Image Classification 12
2.5.1 Multiresolution Binary Coding Texture Descriptor 13
2.6 Evaluation 14
2.6.1 Confusion Matrix 14
2.6.2 Kappa Coefficient 15

3 METHODOLOGY 16
3.1 Overview 16
3.1.1 Finding the Data 16
3.1.2 Ground Truth 18
3.2 Sample Selection 20
3.3 Per-pixel Texture-based Classification 20
3.4 Parameters and Distance Metric Tuning 21
3.4.1 Selection of Sample Size for Training 22
3.4.2 Ringsize Selection 22
3.4.3 Distance Metric 22
3.4.4 VAR Parameter Tuning 22
3.4.5 WLD Parameters 23
3.5 Main Experiments 23
3.5.1 Multiresolution Experiments 24
3.5.2 Concatenation of VAR with LBP and WLD 24

PAGE

3.6 Evaluation 24
3.6.1 Cropping Ground Truth with the Size of Ringsize 24
3.6.2 Excluding Samples from Ground Truth 25
3.7 Tools 25

4 RESULTS AND DISCUSSION 27
4.1 Program Pseudocode 27
4.1.1 Texture Descriptors Pseudocode 27
4.1.2 Distance Metrics Pseudocode 31
4.1.3 Per-pixel Texture-based Classification Pseudocode 32
4.2 Parameter Tuning Results 32
4.2.1 Sample Size 33
4.2.2 Ringsize 33
4.2.3 Distance Measure 33
4.2.4 VAR Parameters 34
4.2.5 WLD Parameters. 35
4.3 Main Experiment Results 36
4.3.1 LBP and LBPRIU Versus WLD 36
4.3.2 The Effect of Variance on WLD and Comparison With LBPRIUVAR 49
4.4 Analysis and Discussion 60
4.4.1 Parameter Selection 60
4.4.2 Comparison of LBP, LBPRIU and WLD 62
4.4.3 Contribution of VAR over WLD and Comparison against LBPRIUVAR 63
4.4.4 Multiresolution Analysis 64
4.4.5 Computation Time 65

5 CONCLUSION 68
5.1 Conclusion 68
5.2 Summary of Contribution 70
5.3 Limitation of Work 70
5.4 Future Research 70

REFERENCES 72

APPENDIX

A. Classified Image Results 76
B. List of Confusion Matrix 94
C. Running Time Of Several Texture Descriptors 115

CURRICULUM VITAE 119

LIST OF TABLES

TABLE PAGE

3.1 Selected texture classes with pixel quantity per region 18
3.2 Statistic of Texture Classes 19
4.1 Percent correct classification with different sample sizes. 33
4.2 Percent correct classification with different ringsize 33
4.3 Percent correct classification with several distance metrics 34
4.4 Percent correct classification utilizing different VAR 'max' value 34
4.5 WLD percent correct classification using different T values 35
4.6 WLD percent correct classification using different M values 35
4.7 WLD results using different S values 35
4.8 Overall percent correct classification from LBP, LBPRIU and WLD on region 1 36
4.9 Overall percent correct classification from LBP, LBPRIU and WLD on region 2 37
4.10 Confusion Matrix of LBP (P: 16, R: 2) in Region 1 39
4.11 Confusion Matrix of LBPRIU (Multiscale) in Region 1 39
4.12 Confusion Matrix of WLD (P: 8 R: 1) in Region 1 39
4.13 Confusion Matrix of LBP (P: 8, R:1) in Region 2 42
4.14 Confusion Matrix of LBPRIU (P: 8, R:1) in Region 2 42
4.15 Confusion Matrix of WLD (Multiscale) in Region 2 42
4.16 Percent correct classification of region 1 using several TD including combination of VAR 49
4.17 Percent correct classification of region 2 using several TD including combination of VAR 49
4.18 Confusion Matrix of VAR (P: 8, R: 1) in Region 1 52
4.19 Confusion Matrix of LBPRIUVAR (P: 8, R: 1) in Region 1 52
4.20 Confusion Matrix of WLDVAR (P: 8, R: 1) in Region 1 52
4.21 Confusion Matrix of VAR (P: 8, R: 1) in Region 2 55
4.22 Confusion Matrix of LBPRIUVAR (P: 24, R: 3) in Region 2 55
4.23 Confusion Matrix of WLDVAR (P: 8, R: 1) in Region 2 55
A.1 Image Results with Different Sample Size on Region 1 77
A.2 Image Results with Different Sample Size on Region 2 77
A.3 Image Results with Different Ringsize on Region 1 77
A.4 Image Results with Different Ringsize on Region 2 78
A.5 Results of LBP (P: 8 and R: 1) on Each Region 78
A.6 Results of LBP (P: 16 and R: 2) on Each Region 79
A.7 Results of LBP (P: 24 and R: 3) on Each Region 79
A.8 Results of LBP (Multiscale) on Each Region 80
A.9 Results of LBPRIU (P: 8 and R: 1) on Each Region 80
A.10 Results of LBPRIU (P: 16 and R: 2) on Each Region 81
A.11 Results of LBPRIU (P: 24 and R: 3) on Each Region 81
A.12 Results of LBPRIU (Multiscale) on Each Region 82
A.13 Results of WLD (P: 8 and R: 1) on Each Region 82
A.14 Results of WLD (P: 16 and R: 2) on Each Region 83
A.15 Results of WLD (P: 24 and R: 3) on Each Region 83
A.16 Results of WLD (Multiscale) on Each Region 84
A.17 Results of VAR (P: 8 and R: 1) on Each Region 84

TABLE **PAGE**

A.18 Results of VAR (P: 16 and R: 2) on Each Region 85
A.19 Results of VAR (P: 24 and R: 3) on Each Region 85
A.20 Results of VAR (Multiscale) on Each Region 86
A.21 Results of LBPRIUVAR (P: 8 and R: 1) on Each Region 86
A.22 Results of LBPRIUVAR (P: 16 and R: 2) on Each Region 87
A.23 Results of LBPRIUVAR (P: 24 and R: 3) on Each Region 87
A.24 Results of LBPRIUVAR (Multiscale) on Each Region 88
A.25 Results of WLDVAR (P: 8 and R: 1) on Each Region 88
A.26 Results of WLDVAR (P: 16 and R: 2) on Each Region 89
A.27 Results of WLDVAR (P: 24 and R: 3) on Each Region 89
A.28 Results of WLDVAR (Multiscale) on Each Region 90
A.29 Best Results of LBPRIU on Each Region 90
A.30 Best Results of WLD on Each Region 91
A.31 Best Results of VAR on Each Region 92
A.32 Best Results of LBPRIUVAR on Each Region 92
A.33 Best Results of WLDVAR on Each Region 93
B.1 Confusion Size selection (Region 1) 95
B.2 Confusion matrix on sample Size selection (Region 2) 96
B.3 Confusion matrix on ringsize selection (Region 1) 97
B.4 Confusion matrix on ringsize selection (Region 2) 97
B.5 Confusion Matrix of LBP on Region 1 (P: 8 and R: 1) 98
B.6 Confusion Matrix of LBP on Region 1 (P: 16 and R: 2) 98
B.7 Confusion Matrix of LBP on Region 1 (P: 24 and R: 3) 99
B.8 Confusion Matrix of LBP on Region 1 (Multiscale) 99
B.9 Confusion Matrix of LBP on Region 2 (P: 8 and R: 1) 99
B.10 Confusion Matrix of LBP on Region 2 (P: 16 and R: 2) 100
B.11 Confusion Matrix of LBP on Region 2 (P: 24 and R: 3) 100
B.12 Confusion Matrix of LBP on Region 2 (Multiscale) 100
B.13 Confusion Matrix of LBPRIU on Region 1 (P: 8 and R: 1) 101
B.14 Confusion Matrix of LBPRIU on Region 1 (P: 16 and R: 2) 101
B.15 Confusion Matrix of LBPRIU on Region 1 (P: 24 and R: 3) 101
B.16 Confusion Matrix of LBPRIU on Region 1 (Multiscale) 102
B.17 Confusion Matrix of LBPRIU on Region 2 (P: 8 and R: 1) 102
B.18 Confusion Matrix of LBPRIU on Region 2 (P: 16 and R: 2) 102
B.19 Confusion Matrix of LBPRIU on Region 2 (P: 24 and R: 3) 103
B.20 Confusion Matrix of LBPRIU on Region 2 (Multiscale) 103
B.21 Confusion Matrix of WLD on Region 1 (P: 8 and R: 1) 103
B.22 Confusion Matrix of WLD on Region 1 (P: 16 and R: 2) 104
B.23 Confusion Matrix of WLD on Region 1 (P: 24 and R: 3) 104
B.24 Confusion Matrix of WLD on Region 1 (Multiscale) 104
B.25 Confusion Matrix of WLD on Region 2 (P: 8 and R: 1) 105
B.26 Confusion Matrix of WLD on Region 2 (P: 16 and R: 2) 105
B.27 Confusion Matrix of WLD on Region 2 (P: 24 and R: 3) 105
B.28 Confusion Matrix of WLD on Region 2 (Multiscale) 106
B.29 Confusion Matrix of VAR on Region 1 (P: 8 and R: 1) 106
B.30 Confusion Matrix of VAR on Region 2 (P: 16 and R: 2) 106
B.31 Confusion Matrix of VAR on Region 1 (P: 24 and R: 3) 107

TABLE **PAGE**

B.32 Confusion Matrix of VAR on Region 1 (Multiscale) 107
B.33 Confusion Matrix of VAR on Region 2 (P: 8 and R: 1) 107
B.34 Confusion Matrix of VAR on Region 2 (P: 16 and R: 2) 108
B.35 Confusion Matrix of VAR on Region 2 (P: 24 and R: 3) 108
B.36 Confusion Matrix of VAR on Region 2 (Multiscale) 108
B.37 Confusion Matrix of LBPRIUVAR on Region 1 (P: 8 and R: 1) 109
B.38 Confusion Matrix of LBPRIUVAR on Region 1 (P: 16 and R: 2) 109
B.39 Confusion Matrix of LBPRIUVAR on Region 1 (P: 24 and R: 3) 109
B.40 Confusion Matrix of LBPRIUVAR on Region 1 (Multiscale) 110
B.41 Confusion Matrix of LBPRIUVAR on Region 2 (P: 8 and R: 1) 110
B.42 Confusion Matrix of LBPRIUVAR on Region 2 (P: 16 and R: 2) 110
B.43 Confusion Matrix of LBPRIUVAR on Region 2 (P: 24 and R: 3) 111
B.44 Confusion Matrix of LBPRIUVAR on Region 2 (Multiscale) 111
B.45 Confusion Matrix of WLDVAR on Region 1 (P: 8 and R: 1) 111
B.46 Confusion Matrix of WLDVAR on Region 1 (P: 16 and R: 2) 112
B.47 Confusion Matrix of WLDVAR on Region 1 (P: 24 and R: 3) 112
B.48 Confusion Matrix of WLDVAR on Region 1 (Multiscale) 112
B.49 Confusion Matrix of WLDVAR on Region 2 (P: 8 and R: 1) 113
B.50 Confusion Matrix of WLDVAR on Region 2 (P: 16 and R: 2) 113
B.51 Confusion Matrix of WLDVAR on Region 2 (P: 24 and R: 3) 113
B.52 Confusion Matrix of WLDVAR on Region 2 (Multiscale) 114
C.1 Quantizing Running Time of LBP, LBPRIU and WLD over Region 1 116
C.2 Quantizing Running Time of LBP, LBPRIU and WLD over Region 2 116
C.3 Classification Running Time of LBP, LBPRIU and WLD over Region 1 116
C.4 Classification Running Time of LBP, LBPRIU and WLD over Region 2 117
C.5 Quantizing Running Time of TD with Combination of VAR over Region 1 117
C.6 Quantizing Running Time of TD with Combination of VAR over Region 2 117
C.7 Classification Running Time of TD with Combination of VAR over Region 1 117
C.8 Classification Running Time of TD With Combination of VAR over Region 2 118

LIST OF FIGURES

FIGURE PAGE

2.1 Example of Binary Code Calculation in a 3x3 Neighborhood. Pixel code is 1 if the value greater than center pixel, else is 0. The final binary code of this example would be “100111110”. (Source: Z. Li et al, 2010) 8
2.2 Construction of LBP Histogram with P value of 8 8
2.3 The rotated LBP pattern when the image is rotated 9
2.4 The 36 unique rotation invariant binary patterns that can occur in the circularly symmetric neighbor set of LBP. Black and white circles correspond to bit values of 0 and 1 in the 8-bit output of the operator. The first row contains the nine uniform patterns. The numbers inside each pattern is its LBPRIU code 10
2.5 Concatenation of two histograms. LBPRIU histogram (a), VAR histogram (b), LBPRIUVAR histogram (c). 11
2.6 Computing a WLD feature of a pixel 11
2.7 An illustration of a WLD histogram feature for a given image 12
2.8 Rectangular neighborhood with different P and R, (a) P: 8 and R: 1, (b) P: 16 and R: 2, (c) P: 24 and R: 3 13
3.1 Flowchart of the research 16
3.2 Location of Study Area. Location of Pathumthani in Thailand (a), Subdistrict location of study area in Pathumthani (b), Selected test regions (marked by two red rectangular shapes) on study area (c) 17
3.3 Study areas, (a) Region 1 and (b) Region 2 17
3.4 Ground truth for the study area, (a) Region 1 and (b) Region 2. Colors correspond to Table 3.1. 19
3.5 Sample selection, (a) subset of region 1 and (b) subset of region 2 20
3.6 Overview of Per-pixel texture classification approach 21
3.7 Procedure on Parameter and Distance Metric Selection 22
3.8 VAR Quantization, (a) VAR Histogram and (b) quantized VAR Histogram 23
3.9 Per-pixel Texture Classification left boundary of ringsize, (a) Window sliding, (b) Area with boundary of ringsize 24
3.10 Example of corrected ground truth, (a) region1 and (b) region 2 25
3.11 Software Architecture 26
4.1 Pseudocode of LBP 27
4.2 Illustration of getNeighbour function 28
4.3 Pseudocode of LBPRIU 28
4.4 Pseudocode of VAR 29
4.5 Pseudocode of WLD 30
4.6 Pseudocode of LBPRIUVAR and WLDVAR 31
4.7 Pseudocode of concatenation process 31
4.8 Pseudocode of Euclidean Distance 31
4.9 Pseudocode of Bhatacharya Distance 32
4.10 Pseudocode of Intersection Distance 32
4.11 Pseudocode of Per-pixel texture classification 32
4.12 Trend of accuracy with different VAR 'max' values 34

FIGURE **PAGE**

4.13 Trend of results from different T,M and S 35
4.14 Best Classified Image of several TD in region 1 38
4.15 Best Classified Image of several TD in region 2 41
4.16 The confusion between crop and water texture class 43
4.17 Accuracy results of several resolutions of LBP,LBPRIU and WLD,(a) results on region 1 and (b) results on region 2 44
4.18 Quantizing time of WLD,LBPRIU and LBP, over region 1 (a) and over region 2 (b) 45
4.19 Classification time using WLD, LBPRIU and LBP, (a) Region 1 and (b) Region 2 46
4.20 Scatter-plot of Quantizing versus Classification running time. (a) Region 1 and (b) Region 2 48
4.21 Best Classified Image of several TD with VAR in region 1 51
4.22 Best Classified Image of several TD with VAR in region 2 54
4.23 Graph Results of several resolutions of TD, Region 1 (a) and Region 2 (b) 56
4.24 Quantizing time of several TD includes combination of VAR, (a) region 1 and (b) region 2 57
4.25 Classification time of several TD including combination of VAR, (a) region 1 and (b) region 2 58
4.26 Scatter-plot of Quantizing versus Classification running time of Each TD Including VAR. (a) Region 1 and (b) Region 2 59
4.27 Different sample size on each region, (a) portion of first region and (b) second region. 60
4.28 Region 2 overlaid with area of texture (a) and respective ground truth with areas where samples are taken (b) 63
4.29 VAR contribution over WLD, (a) Texture area affected by contrast change, (b) Corresponding ground truth, (c) WLD result, (d) VAR result, (e) WLDVAR result 64

LIST OF SYMBOLS

ξ = Differential excitation
θ = Orientation angle
Φ = Dominant Orientation
P = Number of Neighbor values
R = Radius of spatial resolution
T = WLD parameter of Dominant orientation
M = WLD parameter to define the segment in a 2 Dimensional histogram
S = WLD parameter to define size of bin in a 2 Dimensional histogram

LIST OF TECHNICAL VOCABULARY AND ABVREVIATIONS

CDT = C Programming Language Development Toolkit
IDE = Integrated Development Environment
LBP = Local Binary Pattern
LBPRIU = Local Binary Pattern Rotation Invariant
LBPRIUVAR = Local Binary Pattern Rotation Invariant with Variance
max = Maximum Value of Quantized VAR
bin = Bin Count of Quantized VAR
WLD = Weber Local Descriptor
WLDVAR = Weber Local Descriptor with Variance
VAR = Variance Texture Descriptor
GLCM = Gray Level Co-occurrence Matrix
Ringsize = The radius of the circular area used to compute the texture metric for a pixel to be classified
RAM = Random Access Memory
RS = Remote Sensing
SD = Standard Deviation
TD = Texture Descriptor

CHAPTER I INTRODUCTION

1.1 Overview of Problem

Currently, high spatial resolution images have been increasingly used in remote sensing for classification, segmentation and other needed processes (Moran, 2010). However, traditional pixel-based image classification approaches have some limitations in handling this type of imagery (Kima et al., n.d.) since this methodology can only use single pixel features without making use of pixel neighborhood distribution information. High resolution images contain rich geometric structure and texture features as well, which might have an irregular character in local areas but will be regular in larger regions (Zhang et al., 2011). At high spatial resolution, objects are not distinguished by homogeneous spectral characteristics. That is, adjacent pixels are frequently very different colors. On the other hand, objects frequently show a consistent texture over their extent.

One of the most commonly-used texture descriptors (TD) in classification is Local Binary Pattern (LBP) which maintains information about the neighborhood of surrounding pixels (Ojala et al., 2002). However, some of the current research has reported that conventional LBP has some limitations. Due to the small size neighborhood it uses, it is unable to recognize the dominant features with large scale structures (Huang et al., 2011). Recently, by extending some of the concepts of Binary Coding, Weber Local Descriptor (WLD) (Chen et al., 2008) and Local Phase Quantization (LPQ) (Ojansivu et al., 2008) have been introduced as new and robust texture descriptors. These methods have been applied with quite comparable results on the textured-based classification of facial images and other natural photos (Ojansivu and Heikkilä, 2008).

To the best of our knowledge, this methodology has not been widely applied in the remote sensing area. This thesis will employ these two related methodologies of WLD and LBP to evaluate their performance for texture-based classification of high resolution remote sensing images. Furthermore, we will also incorporate Variance texture descriptor into the WLD, to evaluate its contribution in reducing the effect of varying illumination which might lower the accuracy of texture based classification.

1.2 Introduction to Texture Classification

The continuing deployment of satellites for remote sensing has made many of the characteristics of the earth become directly visible. Remotely sensed images can provide many important kinds of information when subjected to various processing techniques. One important category of processing is classification, which strives to differentiate the various regions in the imagery according to land cover, land use, etc.

The earliest approaches to remote sensing classification used the spectral characteristics of individual pixels. These methods examine a single pixel to form decisions on the similarity of the pixel values in multiple spectral bands to other pixels in a class (Long and Srihann, 2004). Thus these methods seek groups of homogeneous pixels. However, because these methods are solely based only on single pixel values, they do not consider multi pixel features which exist in the remotely sensed area (such as texture, spatial structure, etc) (Zhang et al., 2011) which could be quite useful to increase the classification accuracy.

Based on this fact, texture has been a considered as a way to improve the classification Haralick et al., (1973) proposed the most traditional texture descriptor, based on the gray level co-occurrence matrix of a region (GLCM). GLCM is constructed by considering the spatial relationship between two pixels in the image which are called as reference and neighbor pixels. This relationship will define the tabulation of how often the different combination of gray levels co-occur in an image. By utilizing this GLCM, 14 texture features in the image can be extracted, such energy, entropy and others. Using these features will improve the classification results. However, other research suggests that this complete number of the features is not really important and should be avoided to not waste time on unnecessary calculation. For instance, Zhang and Huang (2010) have found that only 10 features are needed. Baraldi and Parmiggiani (1995) discovered another number of features that should be used. Thus the ideal number of features derived from GLCM that should be used to get the optimum feature description is still uncertain. Furthermore, each feature will contribute to the computation complexity; thus utilizing even ten features overall will required quite intensive computation.

Object Based Image Analysis (OBIA) has emerged as an alternative approach to consider texture features. This general method has shown quite good performance doing classification (Blaschke, 2010) with capability to resolve the features in remotely sensed areas. OBIA yields quite high classification accuracy compared with several other methods including GLCM (Zhang and Huang, 2010). In remote sensing images however, this methodology requires initial manual segmentation prior to classification, which takes a lot of the time and effort.

Recently, based on a binary coding approach, the Local Binary Pattern (LBP) texture descriptor has been introduced by Ojala et al. (2002). LBP has a popular reputation for texture classification because of its simplicity and low cost in computation, and has been heavily employed for image texture classification and also segmentation. The LBP method tries to describe the local features of the texture by using a moving window to calculate the number of different salient patterns of the center pixel compared to its neighborhood pixels. LBP produces a histogram that represents the particular texture in the neighborhood of a pixel. The main drawback of original LBP is the small size of the main window. Huang et al. (2011) modified this to capture texture in larger area. And recently (Musci et al., 2013) proposed the LBPRIUVAR which extension of LBP with added contrast invariant feature. However these improvements were sometimes quite complex in calculation and decreased the simplicity that makes LBP so desirable.

Following the high reputation of the LBP, the Local Phase Quantization (LPQ) and Weber Local Descriptor (WLD) methods were introduced at nearly same time (2008). LPQ was proposed by Ojansivu et al. (2008) and surpasses the ability of LBP in image texture classification. The technique basically works with the Fourier transform which enables LPQ to be robust on a blurred image. However, new research has reported that LPQ does not give significant improvement in the classification accuracy when applied to remote sensing imagery (Musci et al., 2013).

As previously mentioned, WLD was presented by Chen et al. (2008). This variant of binary texture descriptor uses the Weber law to measure the local change occurring in textured areas with the concern of the threshold level implied by Weber's law. WLD measures two major things, gradient orientation and central pixel value differentiation. WLD has been shown to rival some other classification methods based on texture such as SIFT and Gabor Filter (Chen et al., 2010).

Currently, to the best of our knowledge, few if any studies in remote sensing have utilized this methodology and none of them focused on texture classification in high resolution imagery. In this thesis, we apply the WLD method to remote sensing images employing a supervised pixel-wise classification approach and measure the performance of WLD in texture-based classification. We compare WLD with the LBP texture descriptor. Furthermore, the contribution of Variance Texture descriptor (VAR) in neutralizing the contrast change is observed and compared with state of the art LBPRIUVAR.

1.3 Objectives

The main objectives of this thesis are:

1. Measure the capability of the WLD texture descriptor for texture-based classification of high resolution RS imagery.
2. Compare the performance of WLD with another popular texture descriptor (LBP) on texture-based classification and with state of the art LBPRIUVAR which utilizes the Variance Texture Descriptor.
3. Observe the texture feature effectiveness as indicated by classification accuracy.

1.4 Scope of The Study

The boundaries of this research are:

1. Only texture features are considered although other features exist in imagery, such as color, spatial shape, entropy and others.
2. The new binary coding texture descriptor, WLD, is utilized to involve texture feature in classification of high resolution remote sensing (RS) imagery.
3. The comparison is conducted with another robust texture descriptor, LBP, on texture-based classification, and with LBPRIUVAR on the variance concatenation process of WLD.

1.5 Contribution

The contribution of this research are as follow:

1. Evaluate the effectiveness of newly proposed texture features for increasing the classification accuracy in the remote sensing domain.
2. Observe the capacity of the WLD as a new texture descriptor for RS classification.
3. See the relative capability of WLD over LBPRIU and state of the art of LBPRIUVAR.
4. See the VAR contribution to neutralize the effects of contrast change in the selected study area.
5. Design and implement the texture-based classification with real data based on high resolution RS imagery

1.6 Organization of Research

This thesis consists of five chapters as follows. Chapter 1 provides the introduction to this research. In Chapter 2 we explain the fundamental concepts used in this work and discuss related research. In Chapter 3 we describe the design framework and the details of our method. In Chapter 4, we present experimental results and discussion. Finally Chapter 5 presents discussion and conclusions.

CHAPTER 2 BACKGROUND AND LITERATURE REVIEW

2.1 Texture

Texture can be described as the intensity variations originating typically from the roughness of an object. Texture features can describe many visual properties such as coarseness, homogeneity, and contrast (Vatsavai, 2011). The major characteristic of texture is the repetition of a pattern or patterns over a region. The pattern may be repeated exactly, or as a set of small variations, possibly as a function of position. There is also a random aspect to texture, because size, shape, color and orientation of pattern elements (sometimes called textons) can vary over the region (Lucieer et al., 2003). Since the texture properties of images appear to be carrying some notable information for discrimination purposes, it is important to develop features for classifying images based on the varying texture.

Early texture models exhibited computational complexity. Randen and Husoy (1999) concluded in their review study that a direction for future research is the development of powerful texture measures that can be extracted and used with a low computational complexity. Local binary pattern operator (LBP) (Ojala et al., 1996) is a relatively new and simple texture model. It is an efficient multi-resolution approach to greyscale and rotation invariant texture classification based on local binary patterns and nonparametric discrimination of sample and prototype distributions. Weber Local Descriptor (WLD) is another new and potentially valuable texture model proposed by Chen et al. (2008). This method is comparable to LBP based on its simplicity and robustness in texture based classification.

2.2 Survey of Prior Research in Texture Descriptors

Automatic classification of remote sensing imagery commenced with the method that attempts to group single pixels based on their spectral similarity as derived from statistical calculations (Long and Srihann, 2004). These pixel-based methods can be categorized into two types, supervised classification and unsupervised classification. The main difference between them is that supervised classification needs some sample data and a training process before it is used to classify the real imagery. On the other hand, unsupervised classification works directly by clustering techniques without the need of training data by finding a pre-specified number of statistical clusters in multispectral or hyperspectral space (Hasmadi et al., 2009). The main disadvantages of both methods is that they cannot consider other features that involve multi-pixel parts, for instance texture, that have become available with the arrival of high-resolution imagery (Zhang et al., 2011). It is important that these features be exploited to produce more accurate classification results.

Haralick et al. (1973) started the attempt to quantify the texture features contained in an image and produced 14 extractable textural features based on the Gray Level Co-occurrence Matrix (GLCM). These textural features are derived from gray-tone spatial dependency matrices that enable the quantification of differences between neighboring cells. This traditional method has been used widely until the present with quite satisfactory results. For example Sehgal (2012) used a neural network with accompanying texture features extracted using this GLCM method and reported some improvement in classification accuracy. Giannini et al. (2013) applied GLCM features in urban area classification and produced a more accurate result than with traditional pixel-based methods. However, there are still some weaknesses of the GLCM. One problem is that calculation of the GLCM for an entire image is computationally expensive (De O Bastos et al., 2008). Another is that there is no conventional agreement about the number of features that should be extracted from the GLCM to model the texture. Zhang and Huang (2010) used 10 features from GLCM while Baraldi and Parmiggiani (1995) mentioned only six features as significant to be used based on their investigation. This indicates that an *apriori* selection of feature must be made prior to calculation of each feature. This takes effort as well.

Another technique called Object Based Image Analysis (OBIA) also appears to improve the classification accuracy of RS images, by considering various features that exist in the image. This method is quite powerful and has been shown to be able to classify images with adequate accuracy. For example Laliberte and Rango (2009) employed OBIA with aerial photo imagery classification using texture derived from GLCM and produced good accuracy results. The basic idea of this method is to group the spatially adjacent pixels into homogeneous objects in terms of their texture, and then conduct classification on objects (not pixels) as the minimum processing unit (Zhang and Huang, 2010). There are three main required steps to produce the classification. The first is segmentation, the second is texture feature extraction, and the last is classification (Li et al., 2010). Despite the high classification accuracy which can be achieved by OBIA, there are still some disadvantages that exist within this method. The first issue is the fact that manual segmentation is required as the first step along with the difficulty of selecting areas of interest. Another set of issues arise when OBIA is applied to high resolution images. For instance, a single pixel is not related to vegetation physiognomy as a whole, and vegetation always shows heterogeneity as a result of irregular shadow or shade (Blaschke, 2010), hence producing poor results.

Chang and Kuo (1993) introduced the Gabor filter. Gabor filter is a linear band-pass filter whose impulse response is defined as a Gaussian function modulated with a complex sinusoid. When used for texture classification, the Gabor filter assumes that images have homogeneous texture and uses means with standard deviation to form the texture descriptor. Gabor filter has been applied for texture classification in previous research (Risojević et al., 2011; Zhan et al, 2009). Regarding the emergence of other Gabor filter variants, Li et al. (2010) conducted experiments to select the most accurate Gabor variant through filtering of texture classification, and reported that Classic Gabor Filter could achieve greater accuracy than other available filters. However, this filter is quite computationally expensive (Shi, 1998; Naskar and Parekh, 2011).

Another filtering method, called Scale Invariant Feature Transform (SIFT) was first proposed by Lowe (2004). SIFT consists of four stages: scale-space extrema detection, key point localization, orientation assignment and key point descriptor. In the texture classification, this method tries to find the key points of interest and form a feature vector on the image to represent the existing texture feature. This method has been applied in the texture classification in collaboration with other methods, such as bag of words and generated quite satisfactory results (Wang et al., 2012). This filter is comparable with aforementioned Gabor filter (Yang and Newsam, 2008), but has some additional advantages over the Gabor filter in that this filter is positive, scale invariant, rotation invariant, illumination invariant and viewpoint invariant. However, there are still weaknesses of this method. It is as computationally expensive as the Gabor filter and it is sensitive to light color changes and non-uniform illumination (shadows) as well.

Ojala et al. (1996) constructed and proposed Local Binary Pattern (LBP) which is a texture descriptor that is capable of capturing the texture variety in an image. The essence of LBP is that it strives to localize the existing texture features into a histogram, by using a moving window with some fixed radius and obtaining the number of corresponding pixels against the neighborhood pixels. The main strength of this method is that it is simple and also low in computation cost. This method also has gained considerable support and has been heavily applied in some areas of image analysis, for example in texture analysis (Li et al, 2010 ; Ahonen et al, 2004), in face recognition, and in segmentation (Tekeli et al., 2007). However, LBP still has a major limitation, caused by the small size of the moving window. Also it is sensitive to rotation and variance (that is, illumination) changes. This problem can be solved by using some modifications (Ojala et al., 2002) that increase the capability of LBP in contexts of multi-resolution and under rotation variance. Musci et al. (2013) tried to extend the capacity of LBP to be variance invariant by concatenating Variance Texture Descriptor (VAR) proposed in (Ojala et al., 2002) to become LBPRIUVAR with encouraging results, even surpassing the capability of notable GLCM. Yet, this enhancement added to the computational costs, and thus reduces the simplicity and efficiency of LBP.

Following the success of LBP, Local Phase Quantization (LPQ) was presented by Ojansivu et al. (2008). The LPQ method is based on the blur invariance property of the Fourier phase spectrum computed in local image windows, which enables LPQ to be blur invariant. LPQ uses the local phase information extracted using the 2-D Discrete Fourier Transform (DFT) for texture classification. This method has been applied for face recognition and surpassed the ability of LBP (Ahonen et al., 2008). However, another current research study reported that LPQ does not achieve big differences in classification accuracy of remote sensing imagery, despite its higher computational cost (Musci et al., 2013).

Weber Local Descriptor (WLD), another texture descriptor described and proposed by Chen et al. (2008) emerged at nearly the same times as LPQ. This method is inspired by Weber's law, which states that the change of a stimulus that will be just noticeable is a constant ratio of the original stimulus. If the change is smaller than this constant ratio, it cannot be recognized. Based on this idea, WLD characterizes local texture using two components, differential excitation ξ and orientation θ. Just like other binary coding descriptors, this method will produce an appropriate histogram for each selected texture. The main advantage of this method is that it considers not only the local contrast but also the structure information represented by gradient histogram (Shiyong Cui et al., 2013). This method appears to rival some of other state-of-the-art texture descriptors, such as SIFT, Gabor and LBP texture classification (Chen et al., 2010). However, based on our knowledge, there are few if any applications of WLD in the remote sensing area, and none of them evaluates this method's capacity in texture-based classification of high resolution imagery.

2.3 Local Binary Pattern (LBP)

LBP was first proposed by Ojala et al. (1996) to encode the pixel-wise information in images with significant texture variation. The LBP method attempts to decompose the texture into small texture units and the texture features are defined by the distribution (histogram) of the LBP code calculated for each pixel in the region under analysis. Figure 2.1 gives an example of binary code in a 3 × 3 neighborhood which generates 2^8 possible standard texture units.

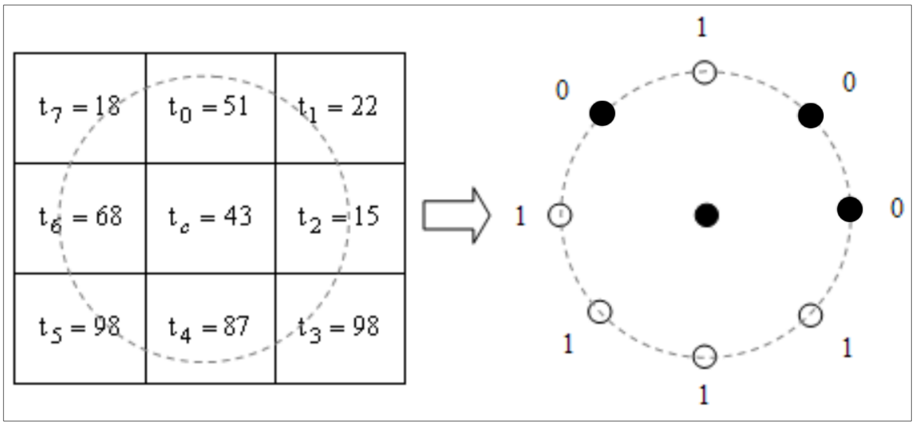


Figure 2.1 Example of Binary Code Calculation in a 3x3 Neighborhood. Pixel code is 1 if the value greater than center pixel, else is 0. The final binary code of this example would be “100111110”. (Source: Z. Li et al, 2010)

The LBP value for the center pixel is calculated using the following equation:

$$LBP_{P,R} = \sum_{i=0}^{p-1} u(t_i - t_c) \times 2^i \tag{2.1}$$

where P is the total number of neighboring pixels, R is the radius used to form the circularly symmetric set of neighbors. In this thesis, we generally use $P = 8$, $R = 1$. The binary labels of the neighboring pixels are obtained by applying a simple threshold operation with respect to the center pixel t_c . $u(t_i - t_c)$ represents a step function, where $u(x) = 1$ when $x \geq 0$; else, $u(x) = 0$. This will result in a single number that represents the texture points of each step in the window. By moving the window over the whole image containing the texture patterns, the method will produce histograms of LBP values from windows centered on each pixel. Figure 2.2 illustrates the construction of LBP histogram by collecting the number resulting from calculations of moving window.

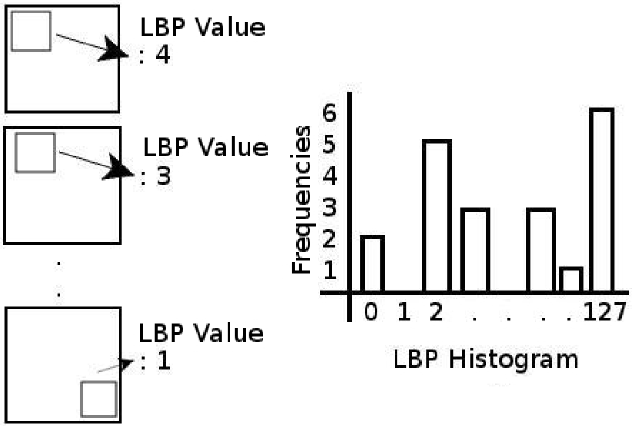


Figure 2.2 Construction of LBP Histogram with P value of 8

2.3.1 Rotation Invariant LBP (LBPRIU)

The binary pattern in LBP depends on the pixel values in each spatial neighborhood. When the image is rotated, the pixel neighborhood is also rotated. This will cause the binary pattern to move along the perimeter of the circle around the center pixels (t_0) as shown in Figure 4.18 below.

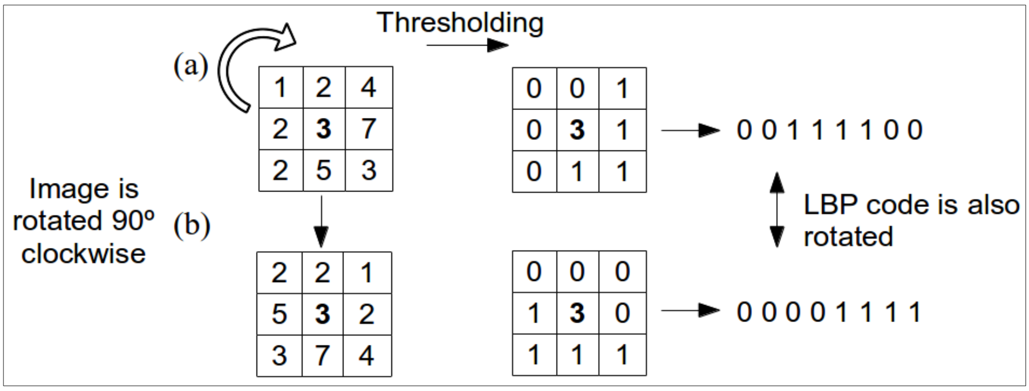


Figure 2.3 The rotated LBP pattern when the image is rotated

To neutralize this rotation effect, a mapping procedure is established by counting the value of the LBP binary pattern as its bin position instead of as a weight as LBP does. This process will able to reserve the discrimination of the LBP pattern when the LBP pattern is rotated as the image is rotated. For example, in the above figure, the patterns of (a) and (b) will be mapped as 4 regardless of how the neighborhood is rotated. The mapping procedure uses the equation below:

$$LBP_{P,R}^{riu2} = \begin{cases} \sum_{p=0}^{P-1} s(t_p - t_c) \text{ if } U(LBP_{P,R}) \leq 2 \\ P+1 \; otherwise, \end{cases} \tag{2.2}$$

The uniformness of LBP is assessed by the U Function belows:

$$U(LBP_{P,R}) = |s(t_{p-1} - t_c) - s(t_0 - t_c)| + \sum_{p=1}^{P-1} |s(t_p - t_c) - s(t_{p-1} - t_c)| \tag{2.3}$$

The value generated from U, corresponds to the number of transitions (changes from 0/1) if we consider the LBP value as a string of binary digits. For example, patterns of 00000000_2 and 11111111_2 will have U value of 0, since there are no transitions from 0 to 1 and vice versa, whereas patterns of 00000100_2 and 00000010_2 will have same U value of 2, since there are 2 transition from 0 to 1 and 1 to 0 for both of these patterns. As can be seen above, that the mapping procedure will collect the uniform LBP values into distinct classes and the non-uniform value into the one miscellaneous class (P+1). Figure 2.4 below shows the unique U sample of LBP Value.

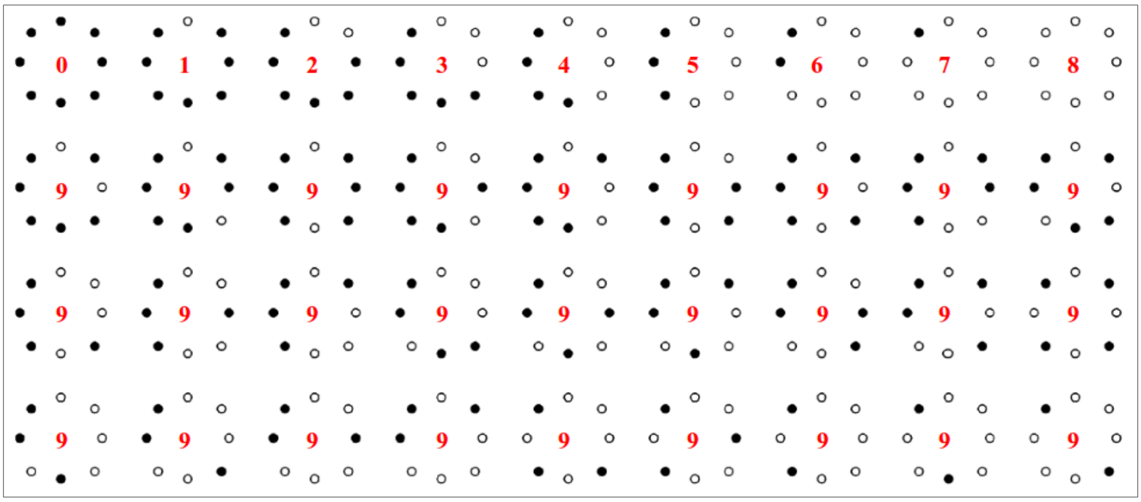


Figure 2.4 The 36 unique rotation invariant binary patterns that can occur in the circularly symmetric neighbor set of LBP. Black and white circles correspond to bit values of 0 and 1 in the 8-bit output of the operator. The first row contains the nine uniform patterns. The numbers inside each pattern is its LBPRIU code
(Source: Ojala et al, 2002)

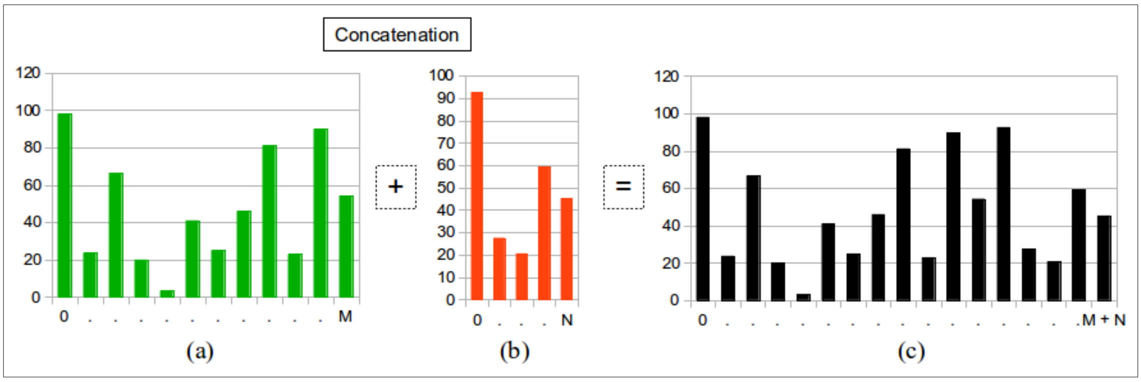
The first row in the figure above shows all the patterns with U value of 2 or less. Each of these 9 patterns will be assigned to its own respective LBPRIU code, which is marked in the center of their pattern in the figure above. Each will be accumulated in a separate bin in the histogram since these patterns are fundamental properties of texture (Ojala et al., 2002). Meanwhile the other 27 patterns with U value at least of 4, will be accumulated in the single miscellaneous class (P+1).

2.3.2 LBPRIU with Variance (LBPRIUVAR)

Both the LBP and the LBPRIU descriptors are invariant to monotonic gray scale changes and consequently do not capture the contrast information. Ojala et al. (2002) propose a local contrast descriptor, denoted as $VAR_{P,R}$, which is also rotation invariant, defined as

$$VAR_{P,R}(w)=\frac{1}{P}\sum_{p=0}^{p-1}(t_p-u)^2 \quad \text{Where} \quad u=\frac{1}{P}\sum_{p=0}^{p-1}t_p \tag{2.4}$$

$VAR_{P,R}$ is an approximation of local variance, that can be computed efficiently if performed concurrently with the computation of $LBPRIU_{P,R}$. The joint histogram of $LBPRIU_{P,R}$ and VAR will make LBPRIU invariant to contrast changes. This can be achieved using concatenation of LBPRIU with VAR as introduced by Musci et al. (2013). Figure 2.5 represents the process of concatenating LBPRIU and VAR histograms.



(a) (b) (c)

Figure 2.5 Concatenation of two histograms. LBPRIU histogram (a), VAR histogram (b), LBPRIUVAR histogram (c).

Note that the joined histogram will have more bins (that is summation of bins from LBPRIU and VAR), thus the computation time will be raised. This combination produces the LBPRIUVAR texture descriptor, which has been shown to compete successfully with several algorithms including GLCM (Musci et al., 2013).

2.4 Weber Local Descriptor (WLD)

Weber Local Descriptor (WLD) was introduced by Chen et al. (2008). This descriptor is based on the Weber's Law, which is a physiological law. It states that the amount of change of a stimulus (such as sound, lighting, etc.) that will be just noticeable is a constant ratio of the original stimulus. When the change is smaller than this constant, a human being would recognize it as a background noise rather than a valid signal. WLD adopts this idea by using a ratio rather than a difference as in LBP to represent differences in the values of neighboring pixels.

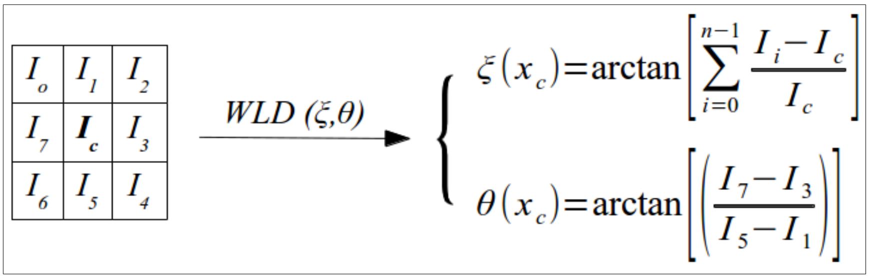

I_0 I_1 I_2
I_7 I_c I_3
I_6 I_5 I_4
WLD (ξ,θ)
$\xi(x_c)=\arctan\left[\sum_{i=0}^{n-1}\frac{I_i-I_c}{I_c}\right]$
$\theta(x_c)=\arctan\left[\left(\frac{I_7-I_3}{I_5-I_1}\right)\right]$

Figure 2.6 Computing a WLD feature of a pixel

WLD consists of two components: differential excitation (ξ) and orientation (θ). ξ is a function of the Weber fraction (i.e., the relative intensity differences of its neighbors against a current pixel and the current pixel itself). θ is a gradient orientation of the current pixel. These two components are calculated for every step of moving window through the entire texture image. The excitation is defined by:

$$\xi(x_c)=\arctan\left[\sum_{i=0}^{n-1}\frac{I_i-I_c}{I_c}\right] \tag{2.5}$$

where n is the number of neighboring pixels and I_i is pixel value at position i. The orientation is calculated as the gradient orientation which is defined by

$$\theta(x_c)=\arctan\left[\left(\frac{I_7-I_3}{I_5-I_1}\right)\right] \tag{2.6}$$

Based on the two terms, a joint histogram can be constructed, followed by converting to a 1-D histogram, which is the WLD descriptor. The conversion of the 2-D histogram of WLD to its respective 1-D histograms, requires three parameters to be selected. The first one is T which partitions the excitation based on the dominant orientation, second is M to partition each histogram of its T value, and last is S, which defines the number of bins for each histogram corresponding to its T and M value. These three values have default values defined in the original paper (Chen et al, 2008), where T is 8, M is 6 and S is 20. All of these numbers should be tuned to get better accuracy. The Figure 2.7 below, illustrates the conversion process from 2-D to 1-D histogram of WLD.

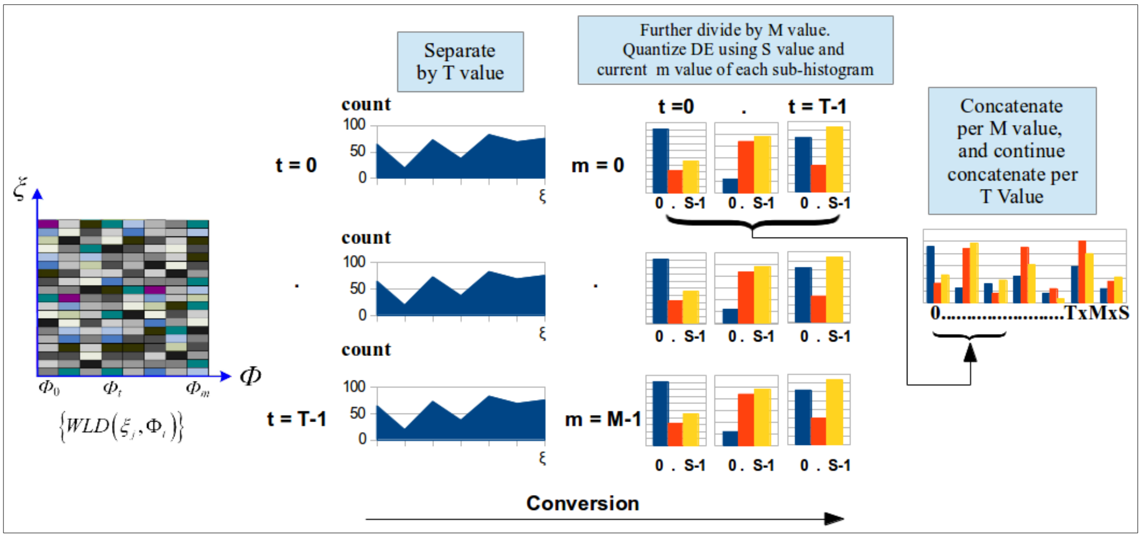

Separate by T value
Further divide by M value. Quantize DE using S value and current m value of each sub-histogram
Concatenate per M value, and continue concatenate per T Value
ξ
Φ
Φ0
Φt
Φm
{WLD(ξj,Φt)}
t = 0
t = T-1
count
100
50
0
ξ
m = 0
m = M-1
t =0
t = T-1
0 . S-1
0........................TxMxS
Conversion

Figure 2.7 An illustration of a WLD histogram feature for a given image

As can be seen in figure above, several steps needed to do the conversion. Those are:

1. Divide the 2D histogram into T histograms of differential excitations with different ranges of orientation.
2. Split each T differential excitation histogram into several sub-histograms, based on M differential range value. This process basically to group the DE based on its value level (high or low).
3. Quantize differential excitation sub-histogram of T and M into S bins.
4. Group (concatenate) the histograms that have a common M value.
5. Concatenate the histograms from step 4 into a single histogram.

Notice that the final bin count of WLD will always be TxMxS regardless the value of P. This is different from previous texture descriptors, where the number of bins depends on the neighborhood size.

2.5 Texture-based Image Classification

Texture-based image classification is a process to classify each pixel into a particular group by considering the texture features existing within neighboring pixels. Texture classes are known *apriori*, and an independent sample of each of them will be used in training. The class assignment is done on a pixel-by-pixel basis. Each pixel is classified into one of the training classes by placing a disk centered on the pixel, calculating the histogram over that neighborhood, calculating a similarity measure against all classes and placing the pixel into the most similar class (Topi et al., 2000).

In classification, the similarity between a training sample and a model feature distribution is measured by a distance metric. There are several available distance metrics which can be employed to examine the similarity. The first distance metric is the Bhatacharya distance which is calculated as below:

$$BD(H^{1,} H^{2})=-\ln\left(\sum_{x=X}\sqrt{H^{1,i}\times H^{2,i}}\right) \tag{2.7}$$

This distance metric has the medium complexity and produces values bounded between 0 to 1. Smaller values imply more similarity between two histograms. The second distance metric is the intersection distance which minimizes the difference between two histograms. It is calculated as shown as below:

$$\Pi(H^{1,} H^{2})=\sum_{i=1}^{L} min(H^{1,i}, H^{2,i}), \tag{2.8}$$

This distance metric has low complexity, but its values are unbounded. This problem can be solved by doing normalization. The last distance metric is the Euclidean distance with equation:

$$D=|H^{1}-H^{2}|=\sqrt{\sum_{i=1}^{n}|H^{1,i}-H^{2,i}|^{2}} \tag{2.9}$$

Similar to the intersection distance, this distance metric is unbounded with low complexity. In this thesis, these three distance metrics are evaluated, and we choose the one with the smallest error in preliminary classification. Then we use that as the final distance metric in the main experiment.

2.5.1 Multiresolution Binary Coding Texture Descriptor

The concept of a multiresolution binary coding TD was first introduced by Topi et al. (2000) for LBP. These researchers found that multiresolution version of LBP produced better accuracy over the single resolution version of LBP. The multiresolution version is basically formed by concatenating several histograms of each TD using several pairs of P and R value. Figure 2.8 below shows the different windows used to form binary coding TD using different pairs of P and R.

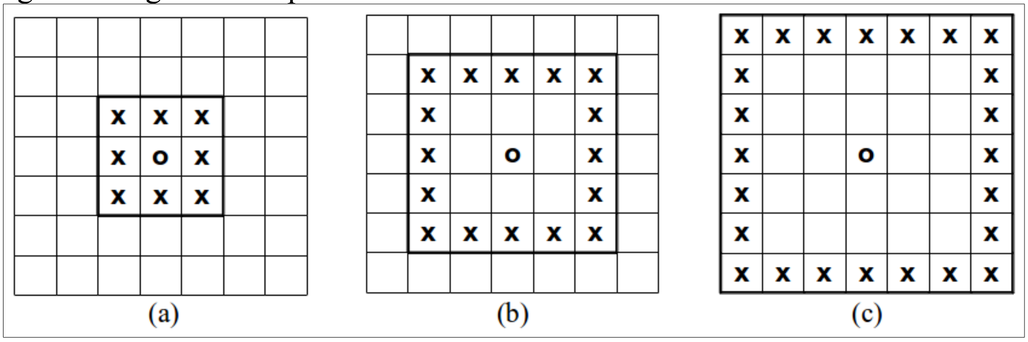


Figure 2.8 Rectangular neighborhood with different P and R, (a) P: 8 and R: 1, (b) P: 16 and R: 2, (c) P: 24 and R: 3

As can be seen, each different resolution considers a different neighborhood. Using these different windows to accumulate TDs will generate different histograms that can be used to evaluate the texture in different scales. Concatenating these histograms will enable multiscale analysis to be conducted. This approach is compatible to other binary coding TD in the family of LBP, and has already been applied in synthetic images with LBPRIU in (Ojala et al., 2002) and WLD in (Chen et al., 2010). In both cases, the multi resolution version gave higher general accuracy than a single resolution. A contrasting result however is shown in (Musci et al., 2013), where the LBPRIU and LBPRIUVAR is applied to remotely sensed imagery. In that study, the multiscale version of LBPRIU did not give significant improvement over the single resolution of LBPRIU.

2.6 Evaluation

2.6.1 Confusion Matrix

The accuracy evaluation of pixel-based supervised classification is usually based on a confusion or error matrix. The confusion matrix requires information about the true class of pixels used for testing. Often this “correct” identification is based on ground survey data. A confusion matrix contains information about actual and predicted classifications done by a classification system. Each pixel that has been categorized from the image is compared to the class label of the same site in the field or reference data set. The result of an accuracy assessment typically provides the users with an overall accuracy of the classification result for each class in the result. The percentage of overall accuracy is calculated using following formula:

$$Overal\,Accuracy\,(\%) = \frac{Total\,number\,of\,correct\,samples}{Total\,number\,of\,samples} \times 100 \qquad (2.10)$$

For example in remote sensing imagery classification, suppose the forested area consists of 250 total pixels as number of samples. If 200 of these pixels are correctly classified as forested area, then the overall accuracy will be 80% for this forest class classification.

Besides the overall accuracy, classification accuracy of individual classes can be calculated in a similar way. The two approaches are user's accuracy and producer's accuracy. The producer's accuracy or recall is derived by dividing the number of correct pixels in one class divided by the total number of test pixels of that class as derived from reference data. The producer's accuracy measures how accurately a certain category has been classified. It captures the errors of omission, that is, the proportion of observed features on the ground that are not classified correctly in the result.

On the other hand, user’s accuracy or precision is computed by dividing the number of correctly classified pixels in each category by the total number of pixels that were classified in that category. The user’s accuracy measures commission errors and indicates the probability that a pixel classified into a given category actually represents that category on ground. Producer’s and user’s accuracy are derived from following formulae:

$$Producer's\ accuracy\ (\%) = \left(1 - \frac{omission\,error}{total\,samples}\right) \times 100 \qquad (2.11)$$

$$User's\ accuracy\ (\%) = \left(1-\frac{commision\ error}{total\ samples}\right)\times 100 \tag{2.12}$$

2.6.2 Kappa Coefficient

Kappa coefficient (K) proposed by Cohen (1960) is another measurement used in this study. Kappa statistic is a less biased measurement of classification agreement and thus gives better assessment of interclass discrimination than overall accuracy. The calculation of Kappa statistic ķ is as follows:

$$k=\frac{\theta_1-\theta_2}{1-\theta_2}$$

$$\theta_1=\frac{\sum_{i=1}^{n} x_{ij}}{N} \quad \theta_2=\frac{\sum_{i=1}^{n} x_{i+}x_{+i}}{N} \tag{2.13}$$

where, x_{ij} = count of pixels in the ijth cell of the confusion matrix; N = total number of pixels in the confusion matrix; x_{i+} = marginal total of row i and x_{+i} = marginal total of column *i* (Hasmadi et al., 2009) . The value of Kappa lies between 0 and 1, where 0 represents agreement due to chance only. 1 represents complete agreement between the two data sets. Negative values can occur but they are spurious and usually expressed as a percentage (%).

CHAPTER 3 METHODOLOGY

3.1 Overview

This research applies Weber Local Descriptor (WLD) as the texture descriptor for supervised texture-based classification of high resolution RS imagery. We compare the result with results obtained from the Local Binary Pattern (LBP). We also explore the usefulness of VAR capability to tackle illumination changes and assess its relative performance against the LBPRIUVAR TD. Figure 3.1 presents an overview of the steps in the research.

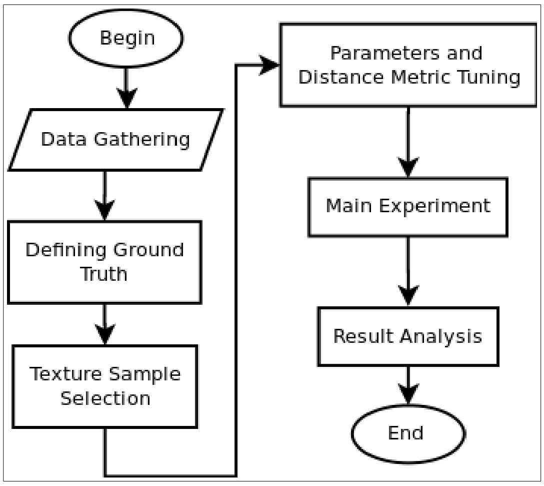


Figure 3.1 Flowchart of the research

In total, there are six steps in our research. The first is the data gathering which is the process to obtain the data for the study area. Next is the process for defining the ground truth which will be used as our reference data for the accuracy measurement. We continue with the texture sample selection to choose the appropriate sample regions along with their size. The next step is the parameter tuning over the available variables on this research and selection of a distance metric. After all of the preceding steps are done, the main experiment will be performed with the setup defined corresponding to the parameters obtained before, then followed with result analysis and ending up with the conclusion

3.1.1 Finding the Data

In this study, we used a panchromatic Quickbird image with 0.6 m spatial resolution acquired on 1 March 2007 and provided by the Geo-Informatics and Space Technology Development Agency (GISTDA). The satellite image covers part of Pathumthani province of Thailand located in 47 N 678206.700mE 1557602.100mN as can be seen in Figure 3.2. This area is comprised of several crop areas holding the distinct texture, naturally vegetated areas, barren areas and water bodies containing texture as well. Two 1024 x 1024 regions were extracted from the raw Quickbird imagery data as the study area as depicted in Figure 3.3 below.

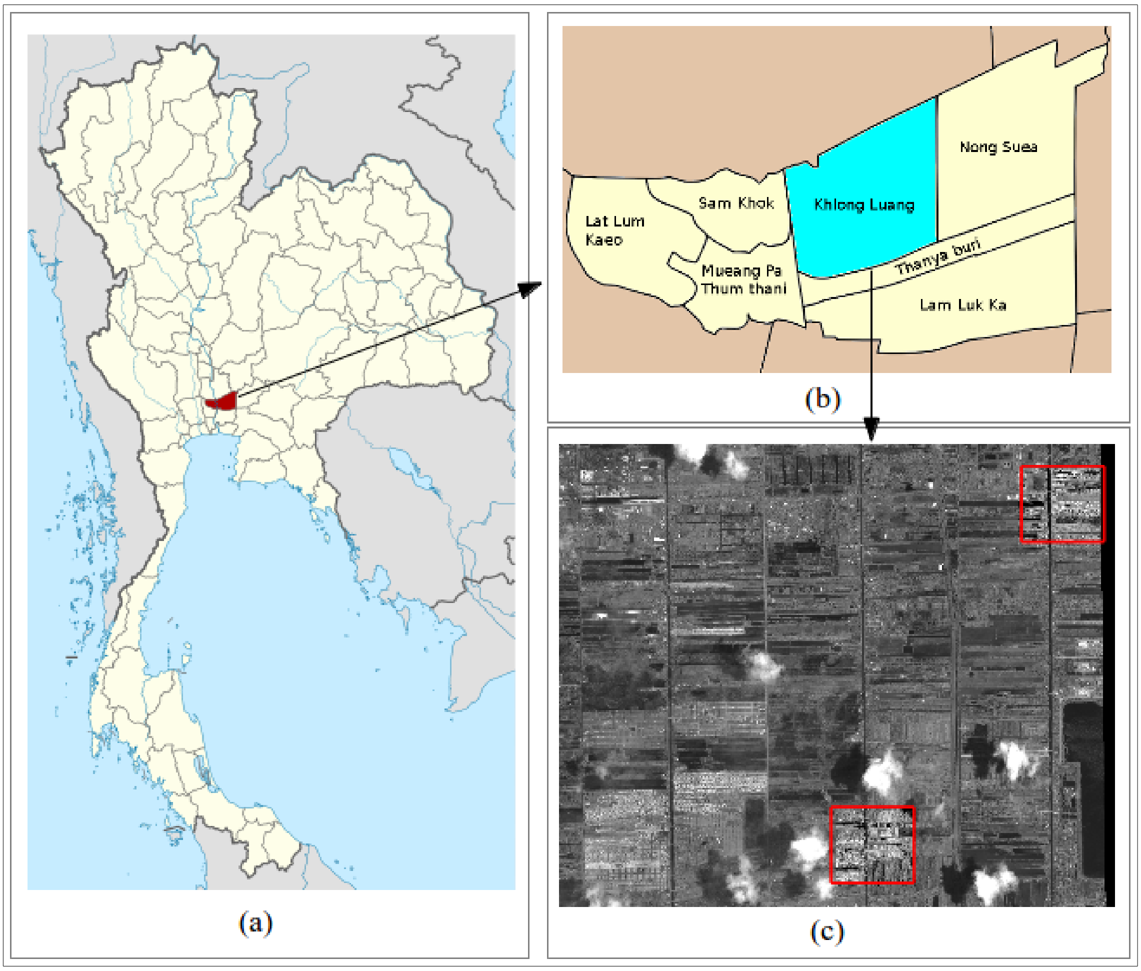


Figure 3.2 Location of Study Area. Location of Pathumthani in Thailand (a), Subdistrict location of study area in Pathumthani (b), Selected test regions (marked by two red rectangular shapes) on study area (c)

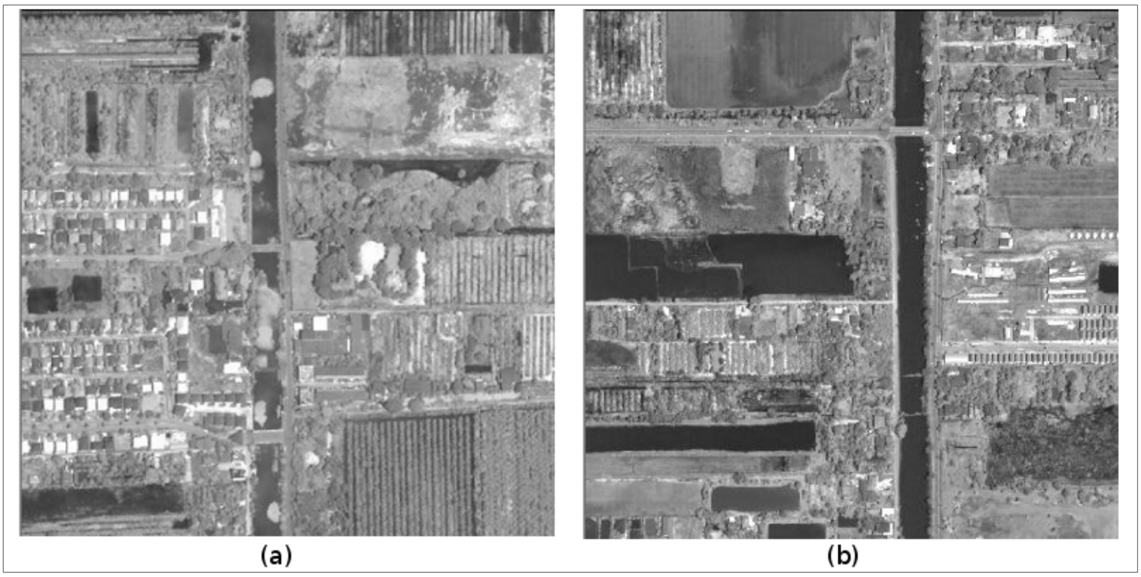


Figure 3.3 Study areas, (a) Region 1 and (b) Region 2

As can be seen, both regions consist of a number of textures, such as crop, paddy, water body, and even some areas with natural changes that are affected in their texture, for example barren, scrub and residential site. The main difference between the two selected regions is that the first region comprises well-defined textures and contains a residential site, which is useful to assess the robustness of several texture descriptors that will be employed in this thesis to represent its texture. The second region has several texture areas affected by contrast changes that can be utilized to evaluate the contrast change invariant level of the TD when quantizing their texture.

Since the texture is clearer in higher resolution and is unaffected by color change, we used only panchromatic imagery. Panchromatic Quickbird imagery has better resolution (0.6 m) compared with multispectral Quickbird images (2 m). The higher resolution imagery present more of a challenge to traditional pixel-based classification and may be expected to benefit more from TD-based approach.

3.1.2 Ground Truth

To create the ground-truth for evaluation of classification accuracy for each of study area, we asked an independent expert analyst to identify the available texture classes in each region, then to form the corresponding ground truth for each of available texture classes with distinct class values. Ten texture-based classes were successfully classified based on appearance. The analyst used both panchromatic and multispectral images in his visual interpretation.

After examining all available texture classes defined from the analysis of independent expert, we conducted a further selection process to choose relatively homogeneous texture classes for both regions. Some of the texture classes identified by the analyst showed great random changes in their textures, which would make them difficult to classify accurately, and several others are too small to provide both training and test areas. Four texture classes per region were selected with a total of five texture classes. Table 3.1 provides the selected texture classes along with the pixel counts per region.

Table 3.1 Selected texture classes with pixel quantity per region

No.	Texture Class	Color	Pixels in Region 1	Pixels in Region 2
1.	Residential	White	115142	0
2.	Orchard	Light Green	176459	74720
3.	Water	Light Blue	67534	147822
4.	Barren	Dark Brown	100061	121952
5.	Crop	Yellow	0	145214

As shown in Table 3.1, in region 1, the orchard site is majority against the other available classes because this area mostly covers an active orchard area. On other hand, the water class is the largest in region 2. Figure 3.4 below, shows the corresponding ground truth over region 1 and region 2, with the statistical data in Table 3.2.

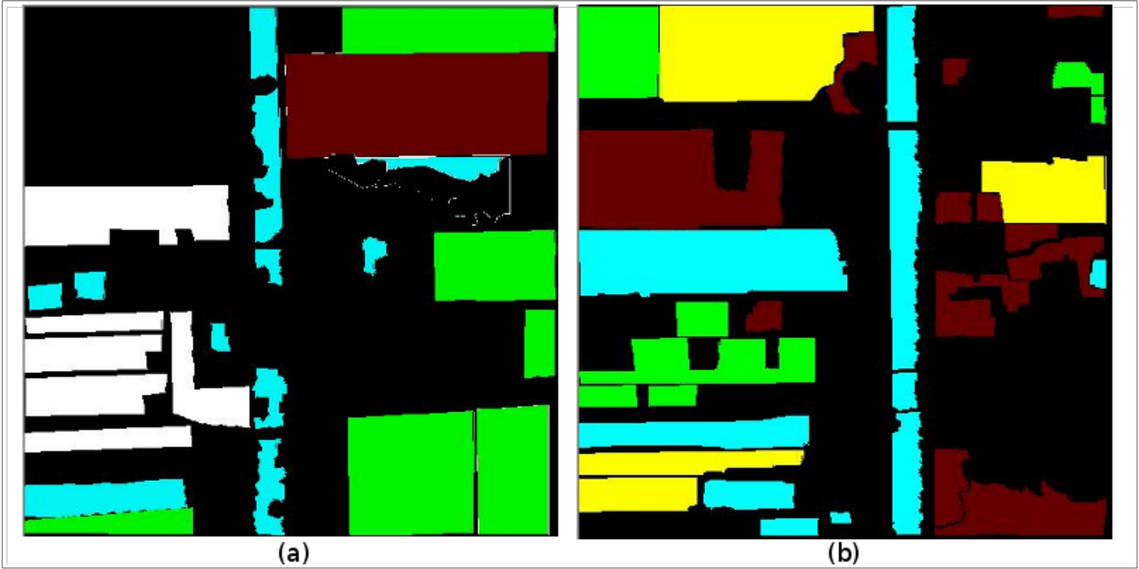


Figure 3.4 Ground truth for the study area, (a) Region 1 and (b) Region 2. Colors correspond to Table 3.1.

Table 3.2 Statistic of Texture Classes

No.	Texture Class	Region 1				Region 2			
		Min	Max	Mean	SD	Min	Max	Mean	SD
1.	Residential	18	230	49.33	14.81	-	-	-	-
2.	Orchard	19	84	39.75	9.24	11	80	39.55	10.35
3.	Water	15	79	27.42	9.28	10	57	17.73	4.71
4.	Barren	26	84	47.63	8.59	13	112	39.48	7.64
5.	Crop	-	-	-	-	13	68	33.96	5.78

Examining the statistical data in Table 3.2, we see that in region 1, the water class has the lowest mean value with the residential as the highest. This signifies that water class has relatively dark pixels and residential as the brightest compared with the other classes in this region. In region 2, water class also has the lowest mean value with the orchard site that has the highest value. The overall mean values of all classes between the both regions also slightly different. The overall mean values of all classes are higher in region 1 than region 2, indicating that region2 may be brighter.

Observing the standard deviation (SD) value in each class for each region, we can see that the residential site class in region 1 and orchard class in region 2 have the highest values. This indicates that these texture classes have great pixel value variation in their area. The orchard texture class in the region 2 also has slightly higher SD compared with region 2. This means that variation of this class is higher compared with region 1. It is expected that the classes with larger variability (larger SD) will benefit more from the texture-based classification approach.

3.2 Sample Selection

After the study areas were selected and ground-truth defined, the next step was training sample selection,. The selected training samples are the model to be compared in texture-based classification which will be explained in the next section. There are two things that need to be done in this process. The first is to select specific areas under the selected classes where each homogeneous texture is present. This to ensure that the obtained sample represents at least one occurrence of its texture class. The second is to choose the sample size. In this thesis, three sizes of sample were considered: 30 x 30, 40 x 40 and 50 x 50. Sizes smaller than this range will not be large enough to include a single occurrence of the texture, whereas larger sizes will require longer time to extract the texture samples. Figure 3.5 shows the subsets of each region where several samples were selected to represent the texture under that area

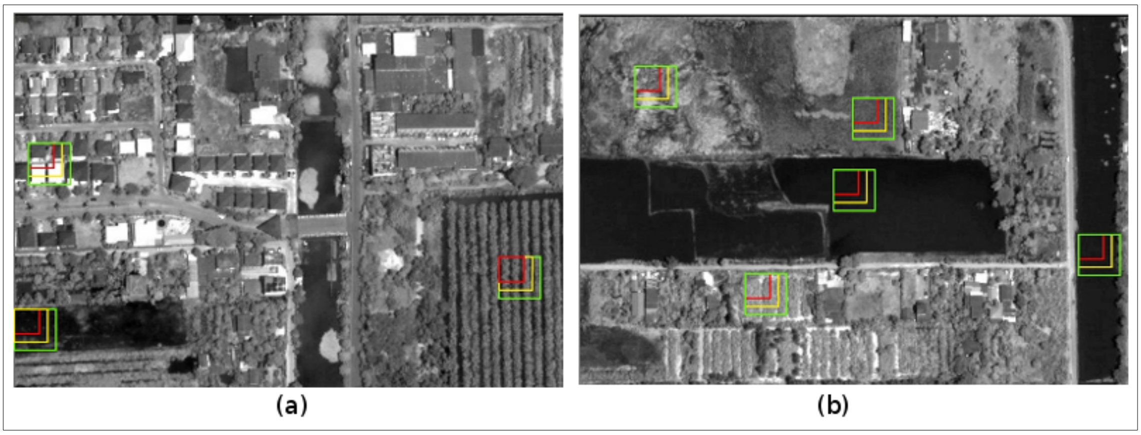


Figure 3.5 Sample selection, (a) subset of region 1 and (b) subset of region 2

Notice that there are 3 rectangles for each sample selection, where red, yellow and green represent the different sample sizes: 30 x 30, 40 x 40 and 50 x 50 respectively.

3.3 Per-pixel Texture-based Classification

This is the essential process which is employed to do the classification. The main idea behind this approach is to incorporate information about the surrounding pixels when evaluating a single centered pixel to be classified, by moving a window with specified radius size called the ringsize. Figure 3.6 Provides an overview of the steps in the Per-pixel Texture-based classification

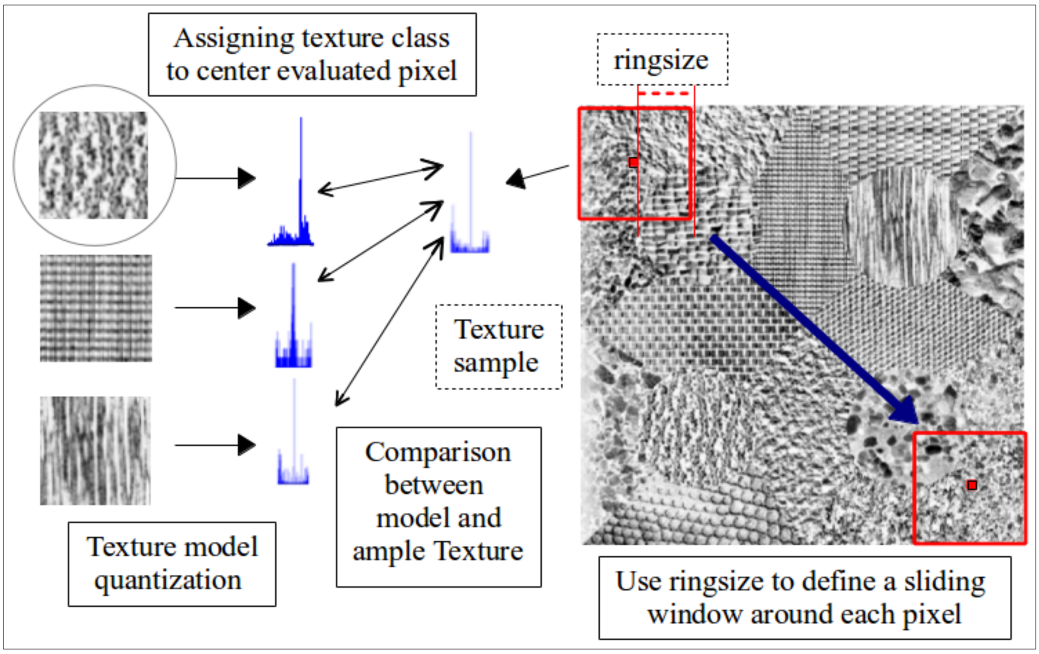

Assigning texture class to center evaluated pixel
ringsize
Texture sample
Comparison between model and ample Texture
Texture model quantization
Use ringsize to define a sliding window around each pixel

Figure 3.6 Overview of Per-pixel texture classification approach

As can be seen, that there are several steps required to do the texture-based classification as follows:

1. Do the texture quantization of training samples. In this step, texture descriptors are calculated to represent the model texture as histograms for each class.
2. Define the ringsize of the window filter. Move the window from the top left side of the image until the bottom right side, one pixel at a time. At each location, form the sample histogram for the region using current texture feature.
3. Compare the image texture histogram with the model histogram using the chosen distance metric.
4. Assign the current pixel to the most similar texture class, that is the one which produces the minimum difference between neighborhood and model histograms.

In this thesis, the classification will executed twice. First we will use it in the parameter tuning process, and then in the main experiments, where the classification is performed using the defined optimum value for each parameter to reach the maximum accuracy.

3.4 Parameters and Distance Metric Tuning

Besides the sample selection, there are other parameters that need to be tuned together, in order to obtain a good accuracy level. Figure 3.7 shows the steps to do the selection of parameter and distance metrics

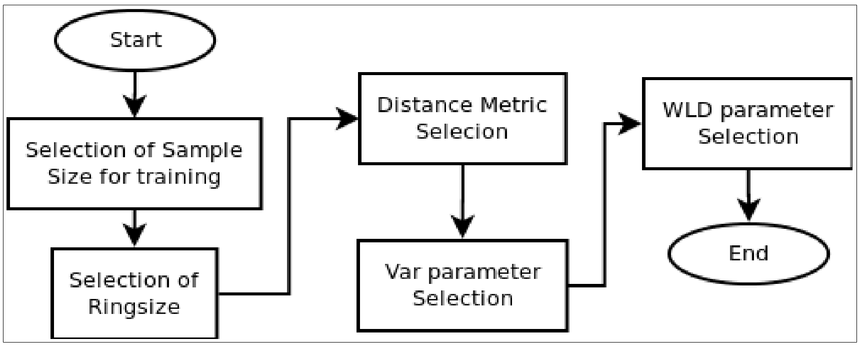


Figure 3.7 Procedure on Parameter and Distance Metric Selection

3.4.1 Selection of Sample Size for Training

We tested three sample sizes, which will be used for the training prior conducting texture-based classification. Those are 30 x 30, 40 x 40 and 50 x 50. We evaluated those sizes by performing the texture-based classification with standard TD of LBPRIU, since it is relatively fast compared with other TDs (with exception of LBP) and able to generate an acceptable rate of the error. We used the ringsize that was half of the sample size. This way, the optimum size of sample selection can be selected appropriately.

3.4.2 Ringsize Selection

The ringsize is the radius of the window, within which we will calculate the histogram for the focused pixels. This is an important parameter in classification since this value controls the number of neighbor pixels to be incorporated in the texture histogram for one pixel. Very small ringsize will be unstable and unable to capture textures with large repeating units. On other hand, a bigger ringsize will able to locate a relatively large texture yet may fail in locating the boundary between textures accurately.

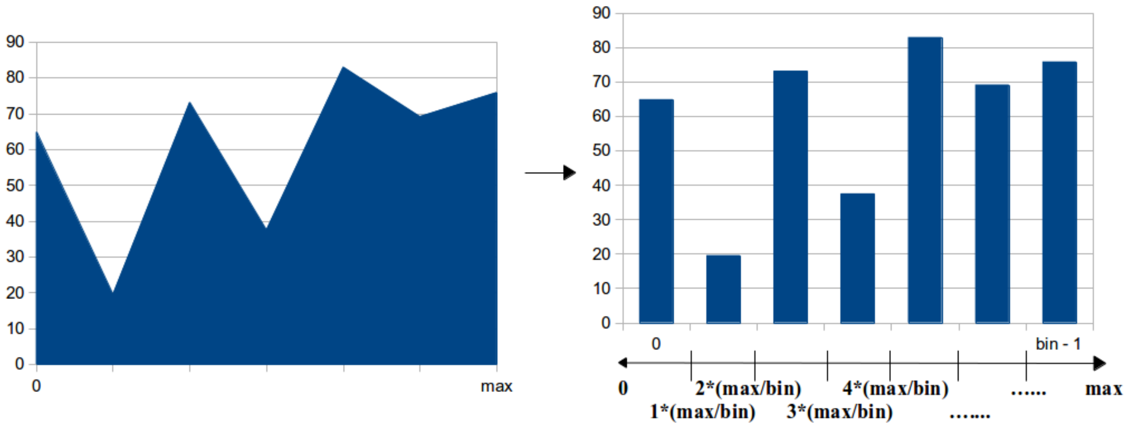
Since there is no general rule to select the ringsize, we opted to select the ringsize experimentally. We varied the ringsize from 10 to 35 and performing a standard classification using LBPRIU and Bhathacarya distance for each size. It might seen reasonable to use a ringsize that is half of the sample size, so that the window covers roughly the same number of pixels as the training samples. However, there is no guarantee that this default value will produce the most accurate results.

3.4.3 Distance Metric

The next item to select is the distance metric. From three distance metrics, we selected the metric that generated the highest accuracy results, using similar setup to prior steps where we selected the ringsize parameter.

3.4.4 VAR Parameter Tuning

For the VAR TD, two values needed to be chosen. First is the VAR bin count (*bin*) that defines the number of distinct values in the VAR histogram, and second is maximum value of the VAR (*max*), which determines the range of values in each bin and in the histograms as a whole. Figure 3.8 depicts the process to quantize the VAR histogram, from continuous to discrete values.



(a) (b)

Figure 3.8 VAR Quantization, (a) VAR Histogram and (b) quantized VAR Histogram

As can be seen, that VAR *bin* count will determine the number of bins in the discrete histogram, and the *max* value will govern the value range in each bin. Every continuous value in that range will be accumulated as one occurrence in the associated bin.

In this thesis, a value of 8 was used for the VAR *bin* count following the setup in the Musci et al (2013). Ideally, *max* could be obtained by collecting the maximum and minimum values observed during training. However, due to the limited number of sample pixels that was not possible. Following the procedure similar to preceding parameters, we observed the optimum VAR *max* value based on the accuracy of preliminary classification results.

3.4.5 WLD Parameters

The last parameters to be selected related to WLD. There are three main WLD parameters required to be tuned. Those are T, M and S with the default values of 8, 16 and 20 respectively (Chen et al., 2010). There is no general rule and limitation of the value selection on these parameters, but trade offs exist. Choosing low values will reduce the dimensionality of histogram, and hence will decrease the discrimination ability but increase the speed of computation, and vice versa. In this thesis, the values of T, M and S were selected by varying those parameters values individually from their default values. By inspecting the trend of accuracy, the appropriate values for each parameter can be obtained. This process was conducted after all of preceding parameters had been chosen, such as ringsize and sample size using this WLD texture descriptor.

3.5 Main Experiments

For the main experiments, all TD were evaluated utilizing the best selected parameters defined in the previous steps. There are four main Texture Descriptors that were evaluated: LBP, LBPRIU, VAR, and WLD. Supervised per-pixel texture-based classification was employed to generate the corresponding classified region with respect of available texture classes. Besides comparison over basic TD, we also did two other experiments. One examined the different scales of Texture Descriptor and also combined them to be a multiresolution version of those TD. The other explored the VAR contribution when concatenated with WLD to deal with the contrast changes within a texture area.

3.5.1 Multiresolution Experiments

This experiment was designed to see the capability of several resolutions of each TD. This evaluates the performance of each TD when quantizing the texture in several scales. This experiment was conducted by varying the value of P and R as discussed in section 2.5.1. In this thesis, three different pairs of P and R were evaluated. Those are P of 8 and R of 1, P of 16 and R of 2, and P of 24 and R of 3. We also concatenated the histograms of those three resolutions to produce a multiresolution histogram to see their multiscale analysis accuracy.

3.5.2 Concatenation of VAR with LBP and WLD

In this experiment, we observed the contribution of VAR to neutralizing the illumination changes on the texture area, by concatenating the histogram resulting from VAR with WLD following the approach illustrated Figure 3.2. Basically we generated the individual histograms of VAR and WLD, then concatenated their histogram producing the joint histogram. By comparing this WLDVAR to LBPRIUVAR, we will able to see its relative capability in classification of remote sensing area.

3.6 Evaluation

To assess the accuracy of each TD classification condition, we used the overall accuracy that expresses the joint results of user accuracy and producer accuracy, along with the their kappa value. Before we did the accuracy calculation, we corrected the ground truth to get more reliable results as discussed in section 3.6.1 and 3.6.2.

3.6.1 Cropping Ground Truth with the Size of Ringsize

The classification procedure omits pixels in boundary of each image and leaves them unevaluated, as shown in Figure 3.9. Therefore we cropped the images to eliminate these border pixels from the accuracy calculation.

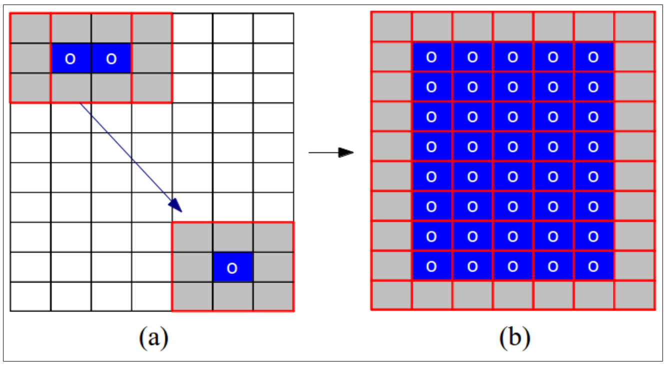
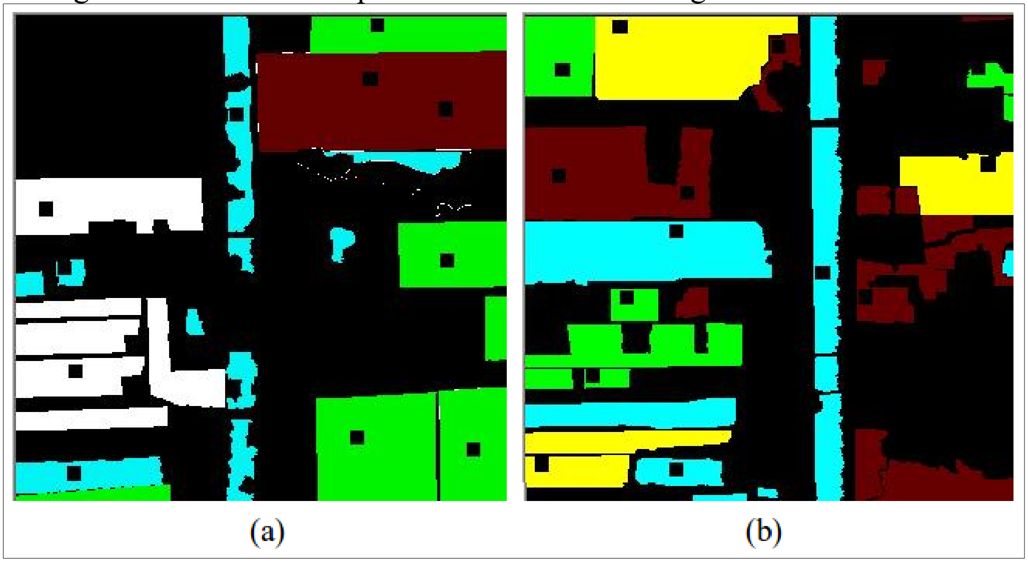


Figure 3.9 Per-pixel Texture Classification left boundary of ringsize, (a) Window sliding, (b) Area with boundary of ringsize

As can be seen in (a), the moving window evaluates its current center pixel incorporating the surrounded pixels of one radius. For each iteration, there is at least one pixel on the border that can not be evaluated since it cannot form a complete window. Thus the final results as shown in the (b) will leave the boundary containing no class value (gray colored cells). In this thesis, this boundary is discarded by cropping the evaluated pixels on the classified images and also on the ground-truth. Since the ringsize value is relatively small compared to the whole image dimension, this process will not significantly affect the accuracy of results.

3.6.2 Excluding Samples from Ground Truth

This process removed the ground truth pixels where the sample areas are taken. This makes our test data independent of the sample data, Figure 3.10 presents the example of corrected ground truth with sample size of 30 x 30 and ringsize 15.



(a) (b)

Figure 3.10 Example of corrected ground truth, (a) region1 and (b) region 2

Notice that the black rectangular areas are spread across the ground truth. These areas are actually the areas where texture samples were taken. They are excluded from ground truth to become the unclassified class.

3.7 Tools

The hardware and software used in this experiment were as follows:

Personal computer	: Intel Centrino dual core processor 2.1 Ghz, with 4 GB of RAM
Operating system	: Ubuntu 12.04
Software	: Eclipse IDE, Dragon ips®, and Dragon Programmer's Toolkit

To develop the program for this research, we used the C++ programming language since it executes faster than Java or scripting languages. In addition, we utilized the Dragon Programmer's Toolkit to do the basic image write and read operations along with other built in capabilities. The program was coded in the IDE of Eclipse CDT and Dragon/ips® was used to do the standard remote sensing procedures, such as defining ground truth, clipping, calculation of accuracy, image displays, etc. Figure 3.11 presents an overview of the final system architecture.

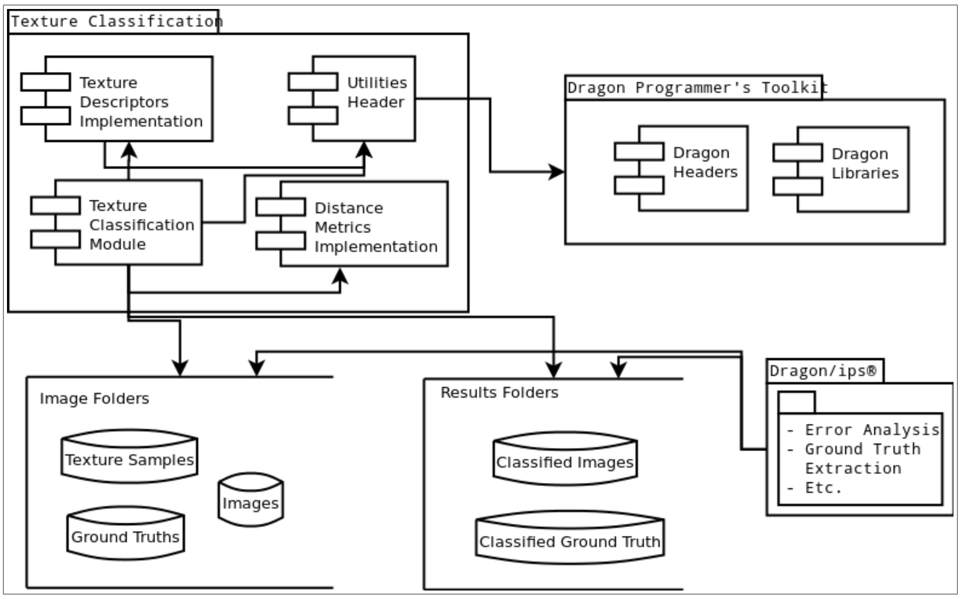


Figure 3.11 Software Architecture

As can be observed, there are several components in the main texture classification module. This module integrates the components from the Dragon Programmer's Toolkit to perform several operations. The image folders is the folder that contains the texture samples and the images with corresponding ground-truth. These image will further be processed by texture classification module and yields the results stored in results folder. Finally, Dragon/ips® will be invoked to do the final accuracy assessment.

CHAPTER 4 RESULTS AND DISCUSSION

This chapter consist of three sections. The first is the pseudocode of all algorithms used in this research. The second is the results of selecting the optimum values for each evaluated parameter. The third is results from the main experiments, where we will compared the accuracy of different TDs using the previously chosen parameters.

4.1 Program Pseudocode

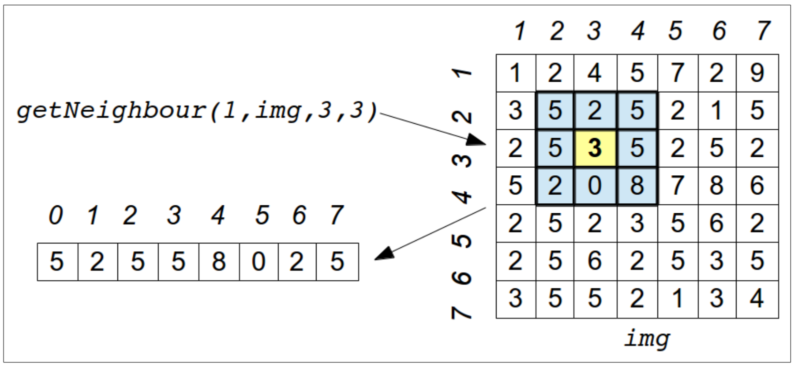
This section presents pseudocode for important algorithms implemented in this research. There are three categories of pseudocode. First is the pseudocode of various texture descriptors. Second is the distance metrics pseudocode. Last is the per-pixel classification algorithms.

4.1.1 Texture Descriptors Pseudocode

```
1   LBP(P,R,Img)
2         integer NB[P]
3         integer LBP_HIST[pow(2,P)]
4         for i → 0 + R to Img.rows – R do
5               for j → 0 + R to Img.cols – R do
6                     index = 0
7                     NB = getNeighbour(R,img,i,j)
8                     for k → 0 to P do
9                           diff = calcDiff(NB[k],Img(i,j))
10                          if diff < Img(i,j)
11                                diff = 0
12                          else
13                                diff = 1
14                          weight = pow(2,k)
15                          index = index + diff * weight
16                    end for
17                    LBP_HIST[index]  =  LBP_HIST[index] +1
18              end for
19        end for
20        return LBP_HIST
21  end LBP
```

Figure 4.1 Pseudocode of LBP

In the above figure, we can see that basically the LBP process is scanning the image pixel by pixel to obtain the texture in the images (following the illustration in Figure 2.2). This image can be either the texture samples used for training, or the subsets of image from the moving window in per-pixel texture-based classification. The getNeighbour function in line 7 returns a one-dimensional array of neighbor pixels given the R value as the distance from supplied center pixel with coordinate i and j as illustrated in Figure 4.2 below.



0	1	2	3	4	5	6	7
5	2	5	5	8	0	2	5

	1	2	3	4	5	6	7
1	1	2	4	5	7	2	9
2	3	5	2	5	2	1	5
3	2	5	3	5	2	5	2
4	5	2	0	8	7	8	6
5	2	5	2	3	5	6	2
6	2	5	6	2	5	3	5
7	3	5	5	2	1	3	4

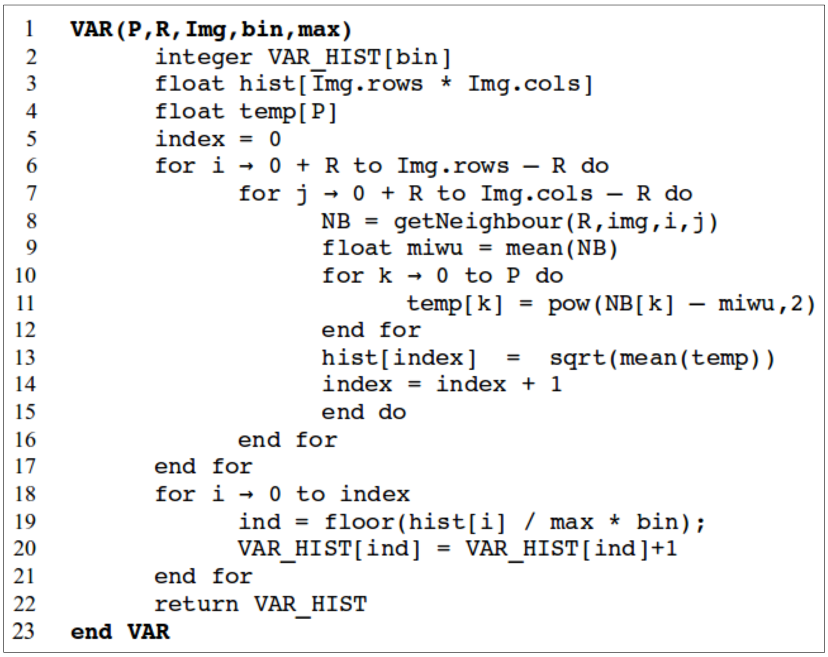
Figure 4.2 Illustration of getNeighbour function

The calcDiff function returns a 0 or 1 depending on whether the center pixel is smaller or larger than value of the neighbor pixel being examined. Notice that in line 14, that the weight value grows with respect to its P value. Furthermore, the combination between weight and diff values directly affects the index value in line 15. Thus a bigger P (the number of neighbor pixels) will generate more possible index values and thus create a larger histogram.

```
1   LBPRIU(P,R,Img)
2         integer LBPRIU_HIST[P+2]
3         for i → 0 + R to Img.rows - R do
4               for j → 0 + R to Img.cols - R do
5                     NB = getNeighbour(R,img,i,j)
6                     for k → 0 to P do
7                           diff[k] = calcDiff(NB[k],Img(i,j))
8                     end for
9                     U = getU(P,diff)
10                    if(U <= 2)
11                          index = getRiu(diff,P)
12                    else
13                          index = P+2
14                    LBPRIU_HIST[index]  =  LBPRIU_HIST[index] +1
15              end for
16        end for
17        return LBPRIU_HIST
18  end LBPRIU
19
20  getRiu(diff,P)
21        index → 0
22        for i → 0 to P do
23              index = index + diff[i]
24        return index
25  end getRiu
```

Figure 4.3 Pseudocode of LBPRIU

The pseudo-code for LBPRIU is similar to LBP. The main difference is that in this TD, there is no usage of weight value and mapping procedure is conducted as shown in line 9 to line 13. This mapping procedure involves two main functions. First is the getU function which returns the uniform value of supplied binary pattern following equation (2.3). Second is getRiu function that produces the bin position as in formula (2.2). This process will makes the number of bins in LBPRIU smaller than LBP.



```
1   VAR(P,R,Img,bin,max)
2         integer VAR_HIST[bin]
3         float hist[Img.rows * Img.cols]
4         float temp[P]
5         index = 0
6         for i → 0 + R to Img.rows – R do
7               for j → 0 + R to Img.cols – R do
8                     NB = getNeighbour(R,img,i,j)
9                     float miwu = mean(NB)
10                    for k → 0 to P do
11                          temp[k] = pow(NB[k] – miwu,2)
12                    end for
13                    hist[index]  =  sqrt(mean(temp))
14                    index = index + 1
15                    end do
16              end for
17        end for
18        for i → 0 to index
19              ind = floor(hist[i] / max * bin);
20              VAR_HIST[ind] = VAR_HIST[ind]+1
21        end for
22        return VAR_HIST
23  end VAR
```

Figure 4.4 Pseudocode of VAR

As can be observed, the pseudo-code of VAR follows a similar approach with LBP and LBPRIU. The first difference is in the quantizing process. VAR TD calculates two means, one of the neighborhood pixels and one of their squared distance (in line 9 and line 13 respectively). As the size of the neighborhood increases, more calculation time will be needed.

The second difference is the presence of quantizing process with respect to the given *bin* and *max* value (in line 17 to 20), that is provided as parameter for this module. This process is performed to quantize the continuous value calculated for variance into discrete ranges.

```
1   WLD(P,R,Img,T,M,S)
2          integer WLD_HIST[T*M*S]
3          float excite_WLD[Img.rows][Img.cols]
4          integer orient_WLD[Img.rows][Img.cols]
5          integer wld2D[m][t][s]
6          float diff[P]
7          for i → 0 + R to Img.rows – R do
8                 for j → 0 + R to Img.cols – R do
9                        NB = getNeighbour(R,img,i,j)
10                       for k → 0 to P do
11                              diff = NB[P] - Img(i,j)
12                       end for
13                       excite_WLD[i,j] = arctan(sum(diff) / Img(i,j))
14                       theta = arctan2(NB[P/2] – NB[0], NB[P*(3/4)] – NB[P/4])
15                       orient_WLD[i,j] = getOrient(theta,T)
16                end for
17         end for
18         for i → 0 + R to Img.rows - R
19                for j → 0 + R to Img.cols - R
20                       m = getM(excite_WLD[i,j],M)
21                       t = orient_WLD[i,j]
22                       s = getS(excite_WLD[i,j],m,S)
23                       wld2d[m][t][s] = wld2d[m][t][s] + 1
24                end for
25         end for
26         MapBy(WLD_HIST,MapBy(WLD2d,M),T)
27         return WLD_HIST
28  end WLD
```

Figure 4.5 Pseudocode of WLD

For the WLD TD, we can see that two features are calculated in this TD in line 9 to 15 compared with preceding TD. Furthermore, there is an additional process in line 18 to 25. That is the conversion of two dimensional histogram to be one dimensional histogram. This conversion requires three steps:

1. Getting the corresponding index value of m, t, and s from current excitation and orientation value (in line 20 to line 22)
2. Accumulating each excitation as one increment to its respective histogram position (in line 23), by using calculated m, t, and s value in preceding step
3. Arranging and concatenating the available histograms using the MapBy procedure in line 26.

These extra processes make WLD more complex compared with LBP, LBPRIU and VAR.

```
LBPRIUVAR(P,R,Img,bin,max)
      integer LBPRIU_HIST[P+2] = LBPRIU(P,R,Img)
      integer VAR_HIST[bin] = VAR(P,R,Img,bin,max)
      integer LBPRIU_VAR_HIST[P+2+bin]
      concatenate(LBPRIU_VAR_HIST,LBPRIU_HIST,VAR_HIST)
      return LBPRIU_VAR_HIST
end LBPRIUVAR

WLDVAR(P,R,Img,T,M,S,bin,max)
      integer WLD_HIST[T*M*S] = WLD(P,R,Img)
      integer VAR_HIST[bin] = VAR(P,R,Img,bin,max)
      integer WLD_VAR_HIST[T*M*X + bin]
      concatenate(WLD_VAR_HIST,WLD_HIST,VAR_HIST)
      return WLD_VAR_HIST
end WLDVAR
```

Figure 4.6 Pseudocode of LBPRIUVAR and WLDVAR

For the combination TD pseudocode, that is LBPRIUVAR and WLDVAR, basically these two TDs use the similar approach. First we perform the individual TD calculation. Next we use a concatenation procedure to merge the generated histograms together. Figure 4.7 below shows the concatenation pseudocode.

```
concatenate(hist[],hist1[],hist2[])
      for i → 0 to hist1.bin + hist2.bin do
            if i < hist1.bin
                  hist[i] = hist1[i]
            else
                  hist[i] = hist2[i-hist1.bin]
      end for
end concatenate
```

Figure 4.7 Pseudocode of concatenation process

The pseudocode shows that the concatenation procedure needs to scan all the bins of both histograms. Thus the more bins that are present, the more time is needed to complete this process. This concatenate procedure is also applied to the multiresolution version of each TD to merge different spatial resolution of corresponding TDs.

4.1.2 Distance Metrics Pseudocode

```
dist_euclidean(int model[], int sample[], int binNumber)
      result = 0
      for i → 0 to binNumber do
            result = result + sqrt(pow(theInput[varX] - theOutput[varX],2))
      end for
      return sqrt(result)
dist_euclidean
```

Figure 4.8 Pseudocode of Euclidean Distance

```
dist_bhathacarya(int model[], int sample[], int binNumber){
        temp = 0
        for i → 0 to binNumber
                temp = temp + sqrt(model[i] * sample[i])
        end for
        temp = 1 - temp;
        if temp <= 0
                temp=0;
        return sqrt(temp)
end dist_bhatacharya
```

Figure 4.9 Pseudocode of Bhatacharya Distance

```
dist_intersection(int model[], int sample[], int binNumber){
        result = 0;
        for i→  0 to binNumber
                result = result + getMin(model[i],sample[i]);
        end for
        tmp = (1-result) //mapping to 0 - 1
        if tmp < 0
                return tmp * -1
        else
                return tmp
end dist_intersection
```

Figure 4.10 Pseudocode of Intersection Distance

From the above pseudocode, we can see that all distance metrics need to travel through both of the histograms being compared. As with concatenation process, the processing time depends on the number of bins in the histograms. Observing the individual pseudocode, it can be seen that Bhatacharya distance is the most complex computation, since it involves more multiplications.

4.1.3 Per-pixel Texture-based Classification Pseudocode

```
1   TextureClassification(img,nSample)
2         for i → 0 to img.rows
3               for j → 0 to img.cols do
4                     subHist = getNeighbour(P)
5                     calculateHist(P,sub) //using selected texture descriptors
6                     for k → 0 to nSample
7                           diff[k] = compareHist(subHist,histModel[k])
8                     end for
9                     assignPixel(i,j,min(diff))
10              end for
11        end for
12  end TextureClassification
```

Figure 4.11 Pseudocode of Per-pixel texture classification

This is the substantive process in this thesis, which is classification pseudocode. This process will utilize the two algorithms presented earlier. The first is the texture descriptor that is used in line 5. The second is distance metric employed in line 7.

4.2 Parameter Tuning Results

This section will describe the results of the parameter value selection. The selection is based on classification accuracy in a test classification scenario. Usually we select the parameter value with highest accuracy. For the selection for TD parameters, we will choose one that will best preserve the discrimination capability of TD as long it has acceptable accuracy as well.

4.2.1 Sample Size

For sample size selection, we considered three sample sizes, from 30 x 30 pixels up to 50 x 50 pixels. We chose those sizes based on inspection of the texture in images by locating the homogeneous texture area in each class, with the results that these three sizes are wide enough to capture the texture occurrences. Additionally, all of the samples were taken from relatively homogeneous texture areas. Employing supervised classification and using LBPRIU as the initial TD over region 1 and region 2 as explained in section 3.4.1, we obtained the results presented in Table 4.1.

Table 4.1 Percent correct classification with different sample sizes.

Sample Size	Region 1 (%)	Region 2 (%)
30 x 30	56.55	**63.15**
40 x 40	60.18	61.62
50 x 50	**61.87**	60.95

As can be seen, samples with 50 x 50 and 30 x 30 size are most appropriate for region 1 and region 2 respectively, since those sizes yields the highest accuracy results.

4.2.2 Ringsize

Ringsize is the radius of area that defines the neighborhood used to compute histograms for test data pixels. To find the optimum value of ringsize, we varied the size in intervals of 5. The setup was similar to that used for the selection of sample size. The results are presented in Table 4.2.

Table 4.2 Percent correct classification with different ringsize

Region (%)	Ringsize					
	10	15	20	25	30	35
Region 1	57.88	60.1	61.72	61.87	62	**62.09**
Region 2	**64.33**	63.15	61.41	59.76	56.6	52.97

Based from the results, we can see that ringsize 35 x 35 is most suitable on region 1 whereas 10 x 10 is best for region 2 since the usage of this ringsize yields highest accuracy. Note that the “default” ringsize, one half of the sample size (25 for region 1) does not produce as accurate results as the chosen ringsize.

4.2.3 Distance Measure

This is the last parameter of the classification process that needed to be tuned. By using the similar setup and selected parameters from preceding sections, we tested the several distance metrics with the results as reported in Table 4.3 below .

Table 4.3 Percent correct classification with several distance metrics

Distance Metric	Region 1 (%)	Region 2 (%)
Bhathacharya	**62**	**64.33**
Euclidean	61.58	63.75
Intersection	61.58	63.81

Although the differences are small, nevertheless we still select the metric with highest accuracy results. Bhatacharya distance appears to be the most appropriate distance metric for both regions. This is consistent with reports in Jenicka and Suruliandi (2011). This distance metric will be used in the main experiments.

4.2.4 VAR Parameters

The parameter of VAR to be chosen is the VAR *max* with the defined VAR *bin* count of 8, described in section 3.4.4 and illustrated in line 18 of the pseudocode in Figure 4.4. This parameter value was selected by varying its value, using the similar setup beforehand but with utilizing the VAR TD instead of LBPRIU. The results are shown in Table 4.4.

Table 4.4 Percent correct classification utilizing different VAR *'max'* value

Region (%)	VAR maximum value					
	5	7	10	15	20	25
Region 1	68.16	68.56	69.07	**69.21**	69.1	67.82
Region 2	68.8	**69.36**	67.21	63.95	61.79	59.54

As can be seen that the results of the different VAR *max* size will also generates the different results, and we can see the clear trend on this results on the Figure 4.12 belows

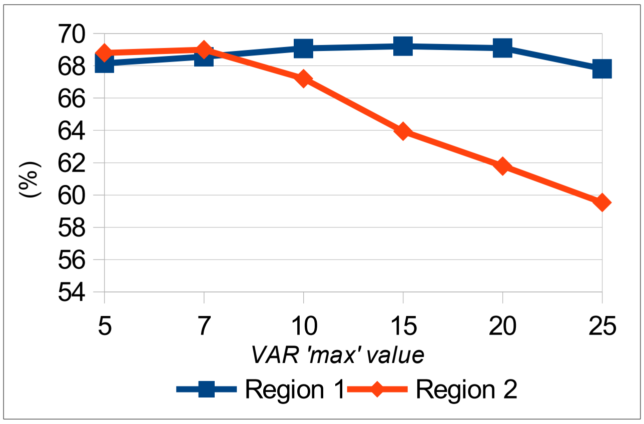


Figure 4.12 Trend of accuracy with different VAR 'max' values

Notice that for region 1, the accuracy gradually increases from value of 5, reaches a peak at 15 and then slowly drops after this value. Hence we selected the value of 15 as the appropriate VAR *max* value for region 1. Repeating this similar observation over region 2 results, we chose the value of 7 as the most appropriate value for region 2.

4.2.5 WLD Parameters.

These are the last parameters to be selected prior to conducting the main experiments. We use similar approach by using texture-based classification over region 1 and 2 utilizing WLD texture descriptor. We varied the values of T, M and S individually from their default values (based on Chen et al. (2010)), and selected the one with acceptable accuracy results. The corresponding results are shown in Table 4.5, Table 4.6, and Table 4.7.

Table 4.5 WLD percent correct classification using different T values

Region (%)	T value			
	4	8	22	30
Region 1	77.27	**82.74**	80.67	75.31
Region 2	72.97	73.17	**72.88**	70.17

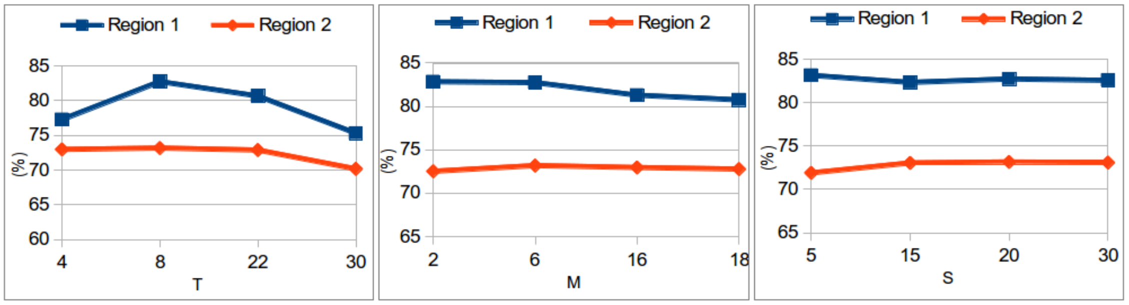
Table 4.6 WLD percent correct classification using different M values

Region (%)	M value			
	2	6	16	18
Region 1	82.84	**82.74**	81.29	80.75
Region 2	72.51	73.17	**72.95**	72.77

Table 4.7 WLD results using different S values

Region (%)	S value			
	5	15	20	30
Region 1	**83.16**	82.34	82.74	82.58
Region 2	71.91	73.06	73.17	**73.1**

The trend of the results from each parameter can be observed in Figure 4.13 below.



(a) (b) (c)

Figure 4.13 Trend of results from different T,M and S

For the parameter T, we selected the value of 8 for the region 1 since it has noticeably better accuracy than the others. We selected the value of 22 for region 2, even though results from usage of value of 22 are not the absolute best, because it preserves more bins thus will generate a more discriminative histogram. Using similar logic, we can obtain the results for the parameter of M and S as well, that is 6 and 16 for M value in region 1 and region 2 respectively, and 5 and 30 for S value in region 1 and region 2. Note that the values used for T, M, and S in Chen et al. (2008) were 8, 6, and 15 respectively.

4.3 Main Experiment Results

This section presents the results of supervised per pixel texture-based classification utilizing several evaluated texture descriptors, using the parameters selected in preceding sections. There are two subsections under this section. The first is the evaluation from the comparison of WLD against LBP and LBPRIU, and the second is the results from the concatenation of VAR over WLD and LBP. In each section, the accuracy, running time and the results from multiresolution version of each TD will be presented and discussed.

4.3.1 LBP and LBPRIU Versus WLD

In this section, we compare the accuracy of LBP, LBPRIU and WLD using supervised classification applied to region 1 and region 2. We also examine different resolutions that corresponding to different P and R for each TD. This to measure the performance of their different resolutions. In addition, we also consider the running time for quantizing the texture and performing the classification.

4.3.1.1 General Accuracy Comparison

Table 4.8 and Table 4.9 show the overall accuracy results from application of LBP, LBPRIU, and WLD texture descriptors on regions 1 and 2 respectively. We present only the overall accuracy in this section, since the pattern of results from overall accuracy are similar to kappa value, even though the kappa value is lower than overall accuracy.

Table 4.8 Overall percent correct classification from LBP, LBPRIU and WLD on region 1

P,R	LBP (%)	LBPRIU (%)	WLD (%)
8,1	73.94	62	**83.16**
16,2	**83.59**	65.01	80.44
24,3	70.48	60	**74.83**
Multiscale	**84.49**	66.52	81.36
Mean	78.125	63.3825	**79.9475**

Table 4.9 Overall percent correct classification from LBP, LBPRIU and WLD on region 2

P,R	LBP (%)	LBPRIU (%)	WLD (%)
8,1	66.91	64.33	**70.5**
16,2	52.99	60.84	**69.51**
24,3	35. 99	51.01	**64.07**
Multiscale	62.17	60.03	**71.18**
Mean	54.515	58.7875	**68.815**

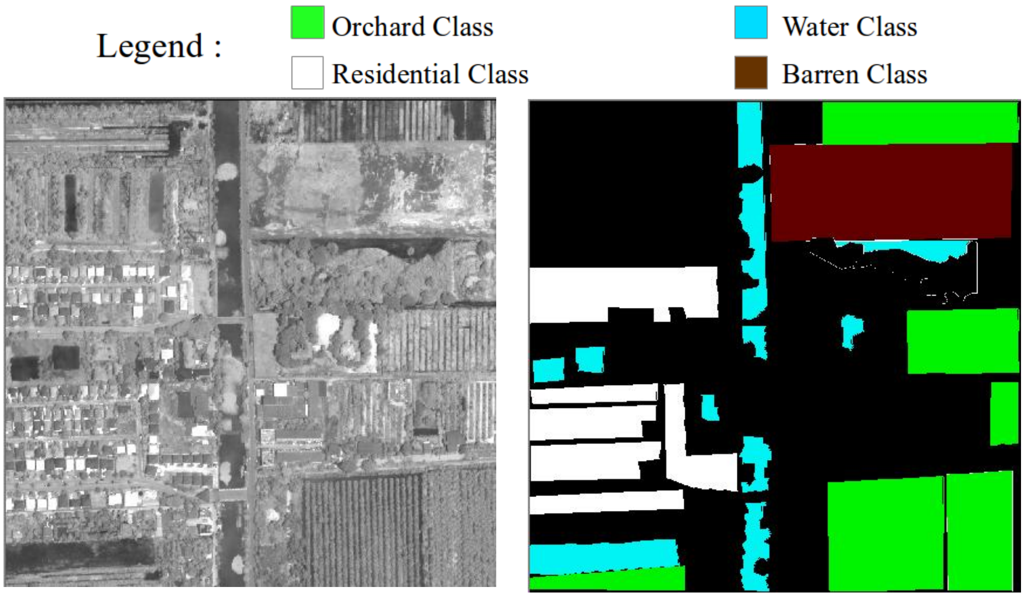
As can be seen from the results, over region 1 and region 2, WLD consistently outperforms both generic LBP and LBPRIU which is the rotation invariant version of LBP, with an exception on the region 1 on basic LBP with the spatial resolution of 2 and multiscale. In this resolution however, LBP is extremely slow (see section 4.3.1.4 for the running time analysis) and becomes unpractical for the real world applications.

Note that the overall accuracy of all TD is better on region 1 than region 2. This is basically comes from the characteristic of the second region where texture is affected by the contrast changes. This makes the classification problem harder than in the first region that comprises relatively homogeneous textures in terms of brightness variation. Another notable result is that adding rotation invariance to LBP to become LBPRIU does not provide any consistent improvements on accuracy, especially in the region 1, where basic LBP has better results than LBPRIU overall. In region 2, we found that LBPRIU sometimes produced better results than LBP.

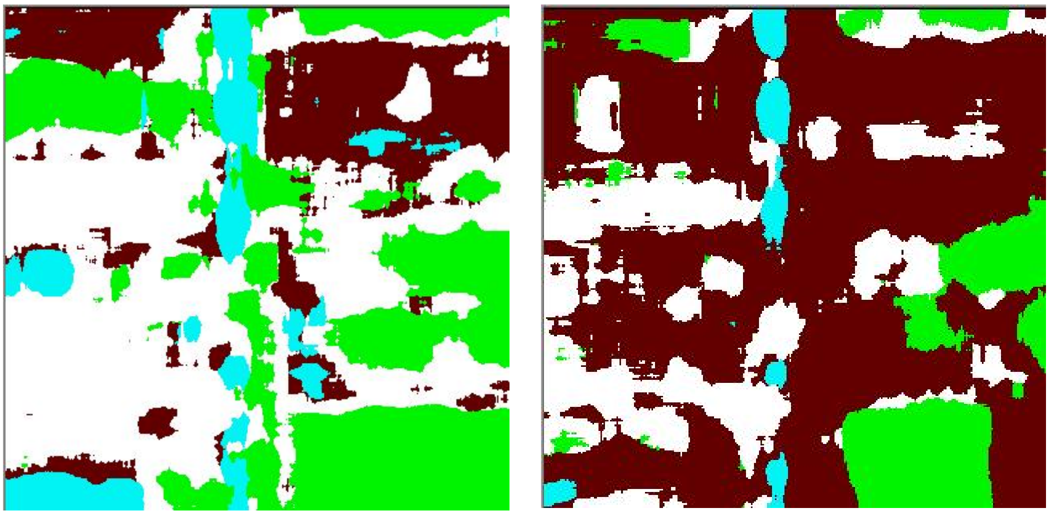
4.3.1.2 Specific Accuracy of Each Texture Descriptor

This section presents the image results for each texture descriptor along with the respective confusion matrix. This to see capability of each TD to classify the individual texture classes. The image results for each TD in region 1 can be found in the Figure 4.14.

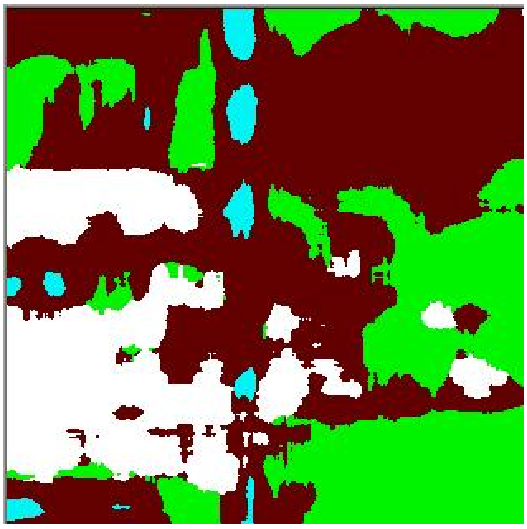
Note that accuracy determination only considering pixel positions that correspond to one of the colored areas (defined texture classes). Pixels that are black in the ground truth image are ignored in the evaluation. Our classification process forces every pixel to be assigned to one of the classes under consideration, even if it represents some totally different type of land cover. We could have masked out these unevaluated pixels, but we believe the complete image give a better sense of the overall capabilities of each TD



(a) Original Image (b) Ground Truth



(c) LBP Image (P: 16, R: 2) (d) LBPRIU Image (Multiscale)



(e) WLD Image (P: 8, R: 1)

Figure 4.14 Best Classified Image of several TD in region 1

Three tables below, shows the corresponding confusion matrix in the above figure.

Table 4.10 Confusion Matrix of LBP (P: 16, R: 2) in Region 1

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	95530	749	232	2412	73.00	96.57
Orchard	9661	103353	1618	1248	93.05	89.19
Water	8458	4575	36188	4370	88.34	67.53
Barren	17208	2393	2925	69071	89.59	75.41
Overall Accuracy						**84.49**
Kappa Value						**0.79**

Table 4.11 Confusion Matrix of LBPRIU (Multiscale) in Region 1

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	64729	2637	0	31557	61.91	65.43
Orchard	11936	77413	0	26531	96.68	66.80
Water	15776	0	17859	19956	100.00	33.32
Barren	12107	18	0	79472	50.45	86.76
Overall Accuracy						**66.52**
Kappa Value						**0.54**

Table 4.12 Confusion Matrix of WLD (P: 8 R: 1) in Region 1

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	91180	548	0	31557	98.06	92.17
Orchard	113	102985	0	26531	98.39	88.87
Water	1688	0	14729	19956	100.00	27.48
Barren	0	1135	0	79472	61.28	98.76
Overall Accuracy						**83.16**
Kappa Value						**0.77**

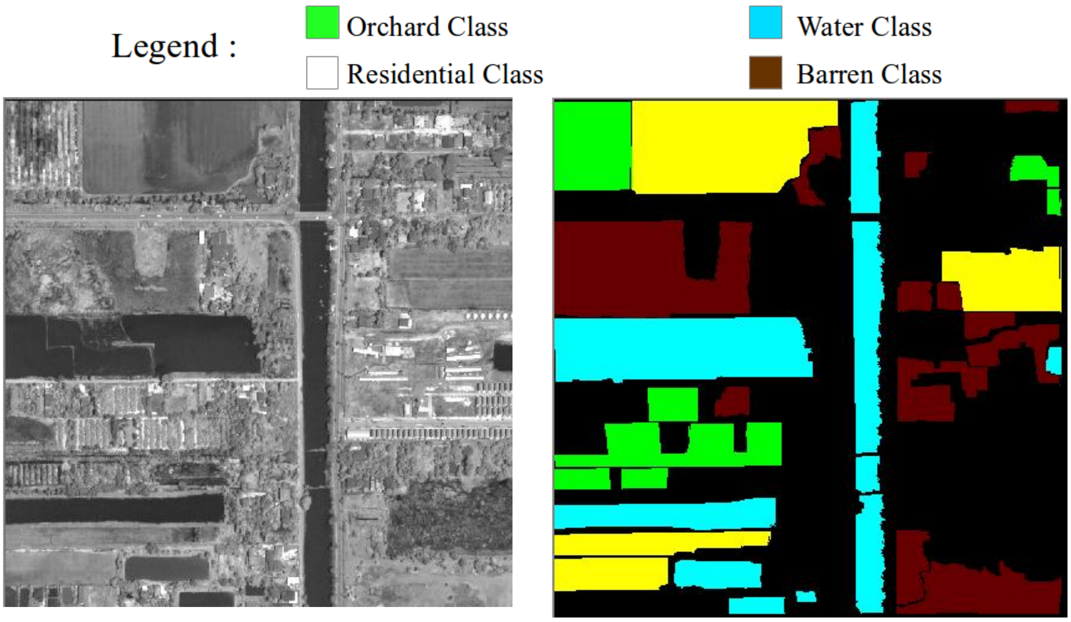
For all TD in general, the high user and producer accuracy is shown in the orchard texture class, compare with other available texture classes. This indicates that all TD are good in recognizing this well-defined texture area. The housing texture area has lower recognition point than orchard area, probably because this texture class is less uniform and contains some changes in their texture units. Both water and barren texture classes have low recognition point shows that these textures are hard to classify. Although these textures produce similar levels of accuracy, more pixels are misclassified as barren area since this texture class has bigger coverage area in region 1.

Specifically, orchard texture class has relatively higher average accuracy over other texture classes with low confusion with other texture classes. The biggest confusion in this class is occurred when recognizing housing and barren area. In other hand, water texture class has good user accuracy overall, especially in LBPRIU and WLD, where the water has perfect user accuracy, which means that no pixel in other texture classes is misclassified as water body. However, lower producer accuracy in this class causes the low average accuracy.

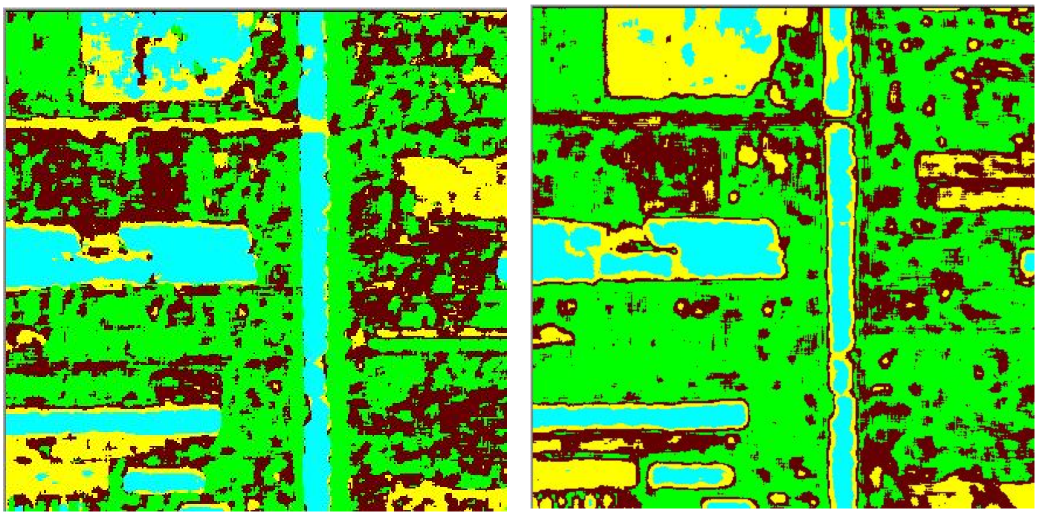
For the housing estate texture class, quite a few pixels in other classes are misclassified as this class, hence lowering its user accuracy. But, the number is slightly lower in WLD. which suggests that WLD is better in dealing the texture site which has natural changes and high variation between pixels. The last is the barren area, that has low recognition rate. Large number of other texture class pixel are misclassified as this class. This is not surprising since this texture has relatively less well-defined texture compared with other texture classes.

The big confusion is occurred mainly between house and barren texture classes. Where number of these texture classes are misclassified each other thus lowering their respective accuracy. Yet, this confusion is lower in WLD case. The another confusion also occurred between house and orchard area, which is greater in LBPRIU but lower in WLD.

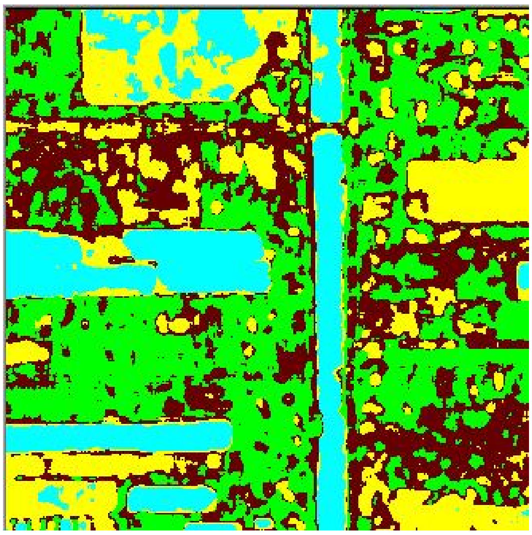
The image results on second region for each TD, can be seen in the Figure 4.15.



(a) Original Image (b) Ground Truth



(c) LBP Image (P: 8, R: 1) (d) LBPRIU Image (P: 8, R: 1)



(e) WLD Image (Multiscale)

Figure 4.15 Best Classified Image of several TD in region 2

Three tables below, shows the corresponding confusion matrix in the above figure.

Table 4.13 Confusion Matrix of LBP (P: 8, R:1) in Region 2

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	52765	0	1337	14257	51.95	77.19
Water	4016	112004	19656	6139	78.14	78.98
Crop	6014	31330	60280	17134	63.38	52.53
Barren	38777	5	13831	83273	68.93	61.28
Overall Accuracy						**66.91**
Kappa Value						**0.55**

Table 4.14 Confusion Matrix of LBPRIU (P: 8, R:1) in Region 2

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	63866	0	129	4364	52.54	93.43
Water	1441	88042	42563	9769	91.33	62.08
Crop	2461	8357	74927	29013	57.60	65.29
Barren	53797	0	12459	69630	61.74	51.24
Overall Accuracy						**64.33**
Kappa Value						**0.53**

Table 4.15 Confusion Matrix of WLD (Multiscale) in Region 2

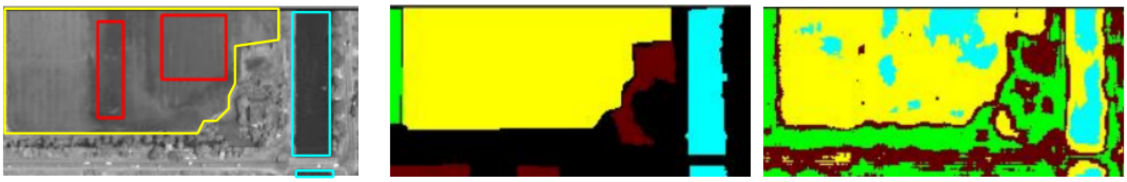
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	53352	0	743	14264	75.32	78.05
Water	376	121756	16087	3596	81.74	85.86
Crop	664	26973	82023	5098	55.72	71.47
Barren	16442	227	48355	70862	75.53	52.15
Overall Accuracy						**71.18**
Kappa Value						**0.61**

The texture class recognition in the second region is different from the region 1. In this region, the water body is the best recognized texture class following by the orchard texture area which has lower accuracy. The crop and barren area have lowest accuracy, where a lot of pixels in other classes are misclassified as these classes. This signifies that these texture classes are hard to classify by all current evaluated TD. In overall, the accuracy of all texture classes for all TD is lower than region 1.

In general, water body texture class has better accuracy compared with other texture classes. The results also higher than in region 1 which is lower. This may comes from the lower variation of pixel color (SD) than in region 2. Hence makes this texture area become smoother which follows the characteristic of water body class. This let the recognition process to be easier.

In other hand, the orchard area has lower accuracy compared with region 1. This is caused by the different texture characteristics in both region, with higher SD in second region and also the present of subtle contrast change in several area in this texture class. These characteristics make the classification harder. The lower recognition is found in crop and barren. Especially in their lower user accuracy where many pixels in the other texture classes is misclassified as both of these textures.

The majority of confusions occurred between crop and water thus lowering their corresponding accuracy. The misclassified area is largely exists in the area where these classes possess similar plain texture characteristics. Figure 4.16 belows shows the part of the second region where crop texture classes has similar texture characteristic with water body texture.



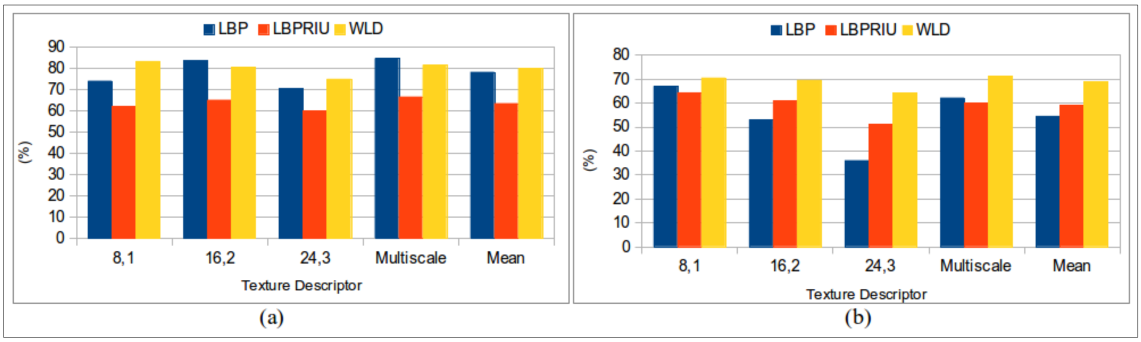
(a) Ground image (b) Ground truth (c) LBPRIU results

Figure 4.16 The confusion between crop and water texture class

It can be seen that in (a), there are several crop areas (in yellow polygon) where the texture are plain (in red rectangles) which is similar with the water body area (in blue rectangles). This harder the classification process and cause the misclassification is occurred between classes as in (c). This confusion lowers their accuracy. The other confusion is also arises between orchard and barren with larger number of orchard pixels misclassified as barren area, which also reduces accuracy for both classes.

4.3.1.3 Multiresolution Analysis

This section examines pattern of results from the usage of several scales of WLD, LBP and LBPRIU. Figure 4.17 charts the results from several resolutions of LBP, LBPRIU and WLD (values from Table 4.8 and Table 4.9).



(a) (b)

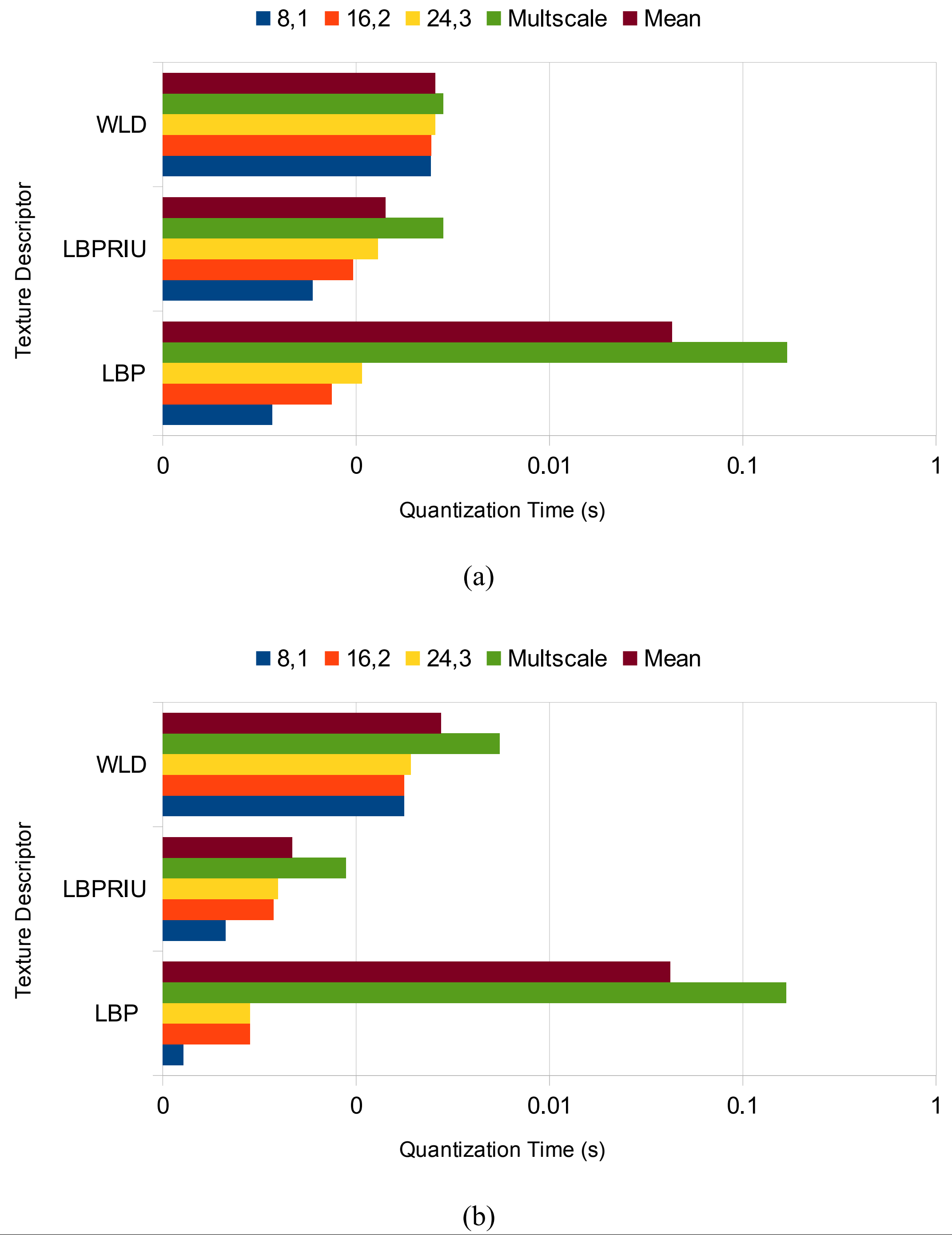
Figure 4.17 Accuracy results of several resolutions of LBP,LBPRIU and WLD, (a) results on region 1 and (b) results on region 2

As can be seen on the results for both region, different scales will yield different results. Generally, the spatial resolution of 1 for each TD has the better accuracy value over other spatial resolutions. Moreover, no improvement is found on the multi-scale version of each of TD over the single resolution of TD. Instead, the multiscale results are close to the mean results across all resolution for each TD. This indicates that multiscale version can be utilized to approximate the relative capability of its TD, even though this multiscale version running time is larger than a single scale version.

We can make other observations. First WLD provides more consistent results across different resolutions. In other words, WLD is less sensitive to variation of parameters. This robustness is a desirable feature. The other is that accuracy of different resolutions very likely depends on the characteristic of textures in the specific images. Thus it is possible that resolutions with value more than 1 would be suitable for images with larger texture units.

4.3.1.4 Running time Comparison

We compared the running time for two subprocesses. The first is the texture quantizing process, that forms the model of training histograms from the available textures using the selected TD. The second is the actual classification using different TD. The running time is obtained by executing the single independent program, and recorded in the unit of seconds. Figure 4.18 shows the running time for several TD when quantizing the texture samples for each of region (the detailed table for these results can be seen in Appendix C)



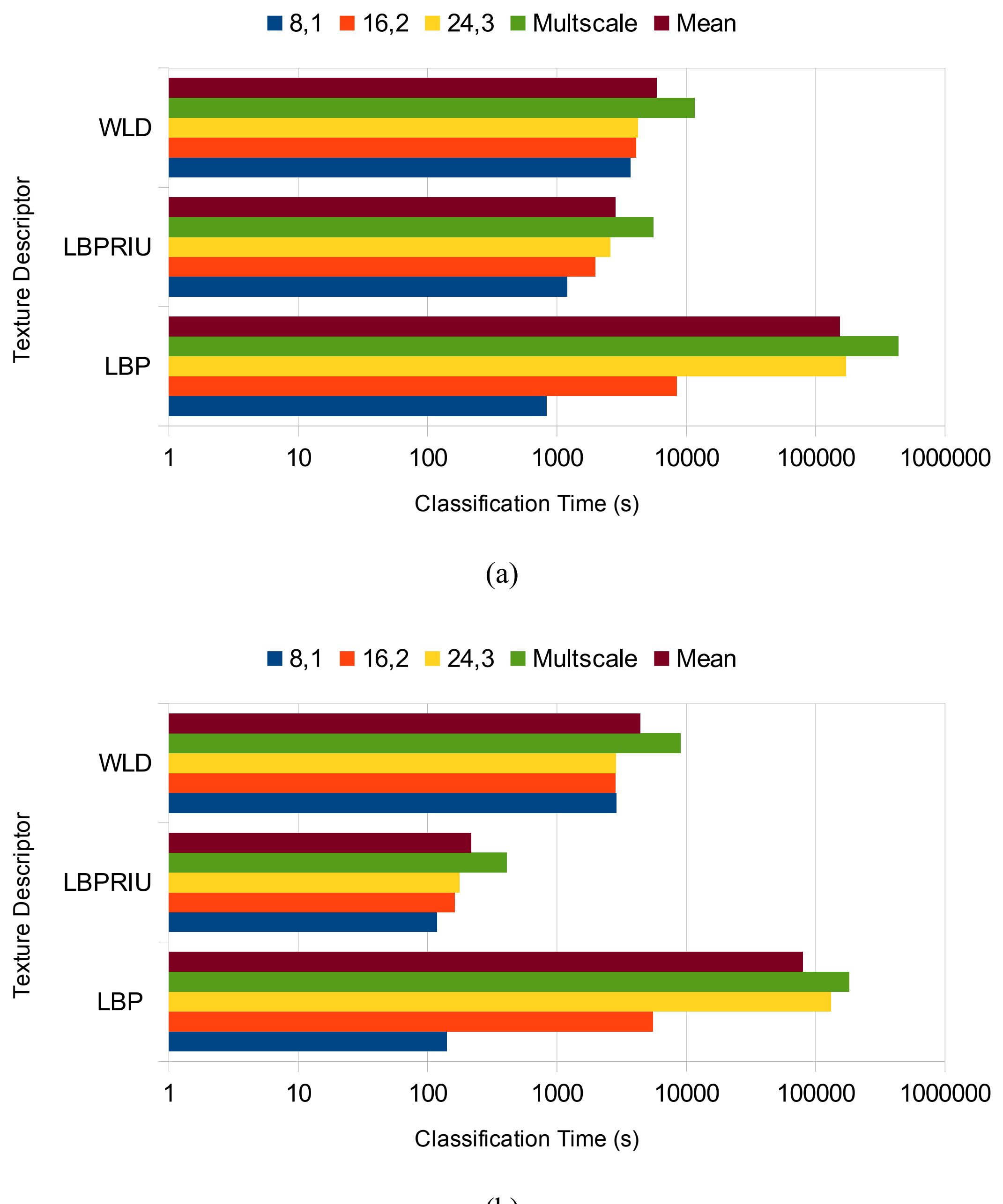
(a)

(b)

Figure 4.18 Quantizing time of WLD,LBPRIU and LBP, over region 1 (a) and over region 2 (b)

As can be seen on the graph, increasing the P and R value will increase computational time for quantizing the texture. The multiscale version of all TD takes the longest time in all cases. This is not surprising since it involves combining calculations from the individual resolutions. LBPRIU is fastest among WLD and LBP based on the mean of running time, looking at every scale of the TD. The LBP and LBPRIU are comparable in the scale of 1 and 2. However, in the resolution of 3 which is the widest one and also in multiscale, the LBP is much slower than LBPRIU. WLD running time is more stable, not as much affected by scale and it lies between LBP and LBPRIU on all scales.

The next stage of running time evaluation is the actual classification utilizing these three texture descriptors, Figure 4.19 presents the actual running time for each TD condition.



(a)

(b)

Figure 4.19 Classification time using WLD, LBPRIU and LBP, (a) Region 1 and (b) Region 2

The pattern is similar to the quantizing running time, in that larger spatial resolution will require more computational time, with multiscale as the slowest one. The LBP running time is dramatically slower when the spatial resolution is higher than 1. Even with P:16 and R:2, the time is close to 10,000 seconds. The time is more than 100,000 of seconds (more a day) in resolution of 3. This further increased in multiscale resolution as well. This makes LBP using any resolution other than 1 totally impractical in real applications. Note that these results come from classifying small (1024 x 1024) images, that is, about a million pixels. Full scenes from modern high resolution images tend to be in the range of 10,000 x 10,000 pixels or more.

To see the relationship between both time and see the effect of the increasing quantization time to the entire classification elapsed time. Figure 4.20 belows shows the scatter-plot of quantization time of each TD with its respective classification time.

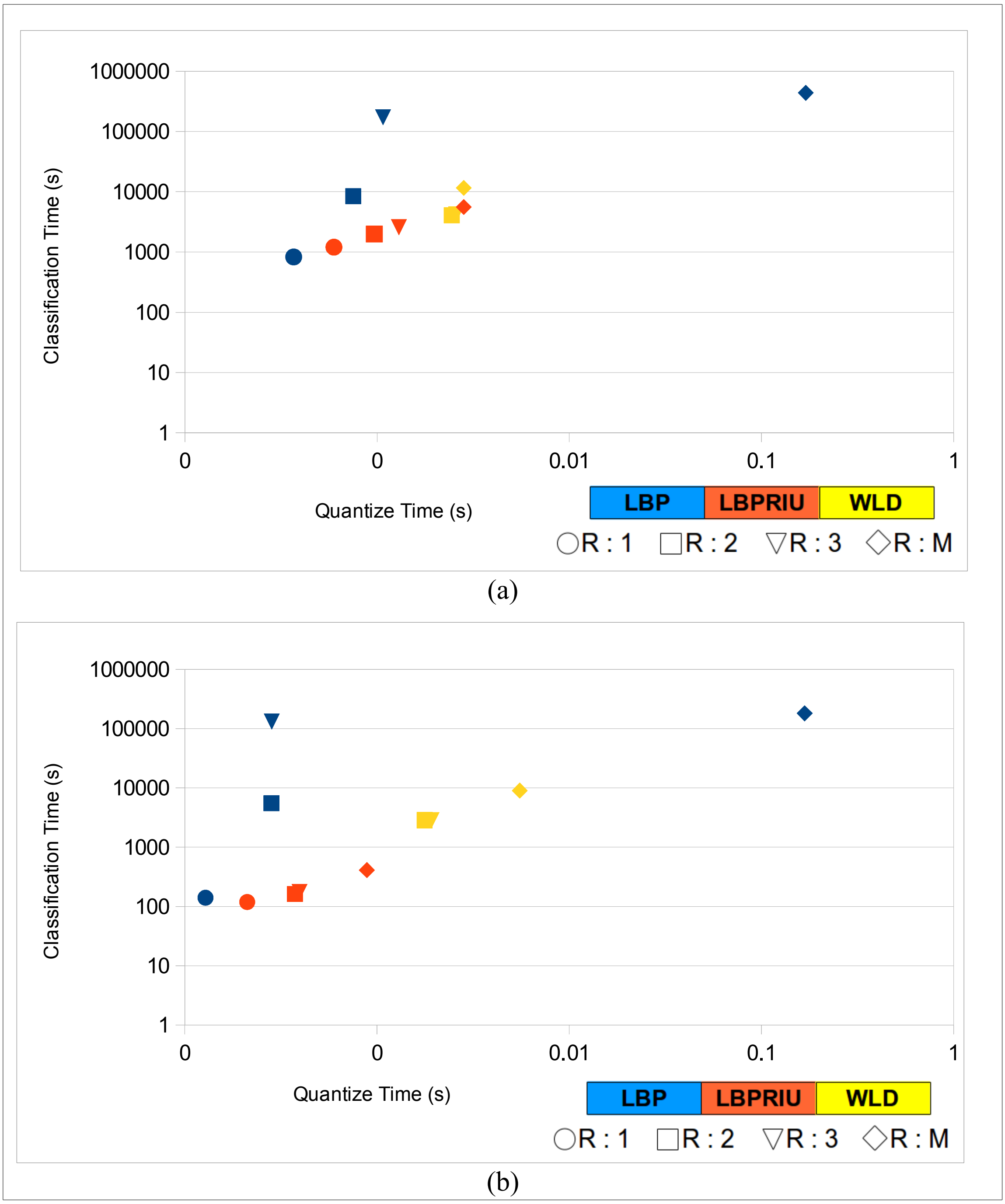


Figure 4.20 Scatter-plot of Quantizing versus Classification running time. (a) Region 1 and (b) Region 2

As can be seen, for LBPRIU and WLD classification time increases roughly linearly as the quantization time is increased. The scatter-plot of LBP shows an exponential curve as the quantization time increases. This exponential growth corresponds to the exponential increase in the number of bins in LBP as resolution increases. All these bins need to be scanned for each pixel in order to calculate the distance metric. Thus, looking for this correlation shows that the classification time for LBP is dominated by the scanning bin process in the distance measurement. On the other hand, LBPRIU exhibits almost linear pattern as the quantization time increases. This indicates that calculations of the TD itself is the main determinant of classification time.

The LBPRIU quantization time is slightly increased as the resolution is increased mainly due to mapping process in LBPRIU. Since this process needs to scan the binary patterns which is also grow as the neighborhood increased (which is directly controlled by R value). Thus larger resolution will cause the LBPRIU quantization time to slightly increase. This is different for WLD with almost constant time for both quantization and classification. WLD does not do the mapping process and with the number of bins which always the same regardless of the resolution. The only exception is in the multiresolution version, where the histograms of all the contributing scales are concatenated.

4.3.2 The Effect of Variance on WLD and Comparison With LBPRIUVAR

In this section, we present the results of applying the VAR texture descriptor to region 1 and region 2 using a similar setup on the last section, then joining the VAR histogram results with the WLD and LBPRIU histogram, by using a simple concatenation scheme as explained in section 3.5.2. We evaluate the results of this combination including several scales and multiscale version, in each region based on the accuracy and also the running time.

4.3.2.1 Accuracy Comparison

Applying the the VAR and its combination with WLD and LBPRIU, we obtain the accuracy results for each region as shown in Table 4.16 and Table 4.17 belows.

Table 4.16 Percent correct classification of region 1 using several TD including combination of VAR

P,R	LBPRIU (%)	WLD(%)	VAR(%)	LBPRIUVAR(%)	WLDVAR(%)
8,1	62	**83.16**	69.21	69.35	78.57
16,2	65.01	**80.44**	67.23	69.42	74.45
24,3	60	74.83	66.67	71.22	**77.91**
Multiscale	66.52	**81.36**	68.15	70.36	78.28
Mean	63.3825	**79.9475**	67.82	70.0875	77.3025

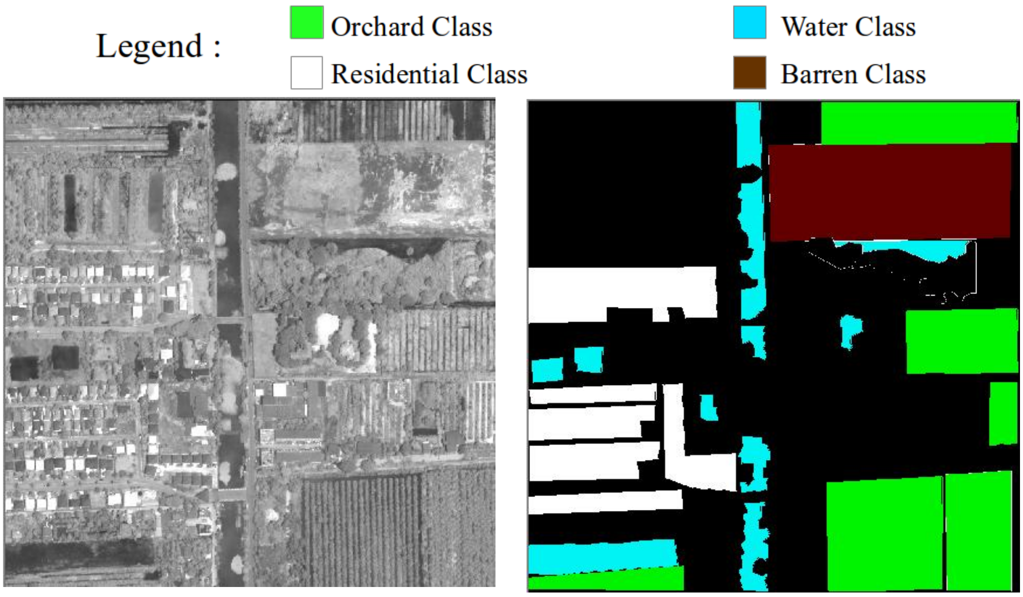
Table 4.17 Percent correct classification of region 2 using several TD including combination of VAR

P,R	LBPRIU (%)	WLD(%)	VAR(%)	LBPRIUVAR(%)	WLDVAR(%)
8,1	64.33	70.5	69.36	69.02	**73.58**
16,2	60.84	69.51	66.39	66.5	**70.27**
24,3	51.01	64.07	62.65	62.95	**65.58**
Multiscale	60.03	**71.18**	66.4	66.4	70.8
Mean	58.7875	68.815	66.2	66.2175	**70.0575**

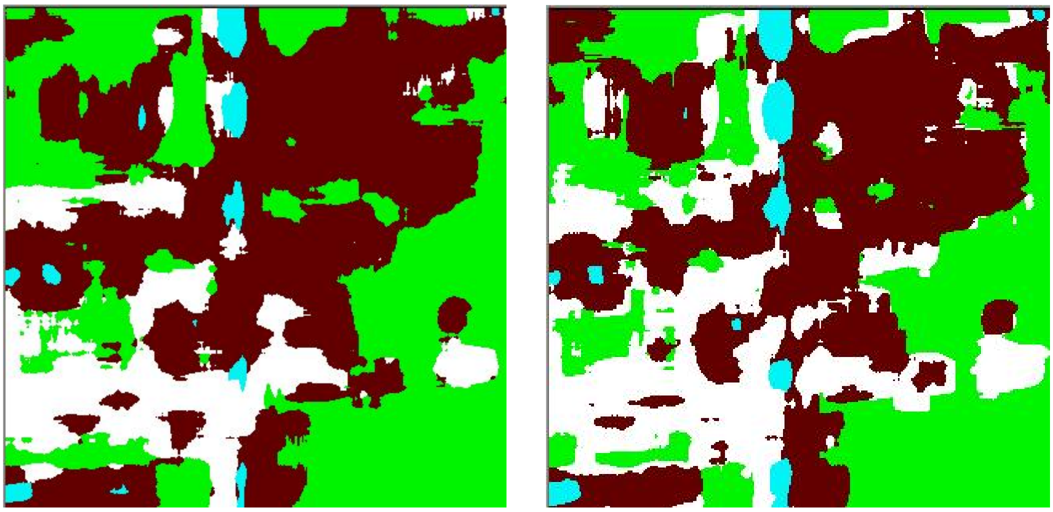
Observing the results, we can see that VAR TD by itself is capable of classifying image quite well, with the mean in about 67% for both regions. However in region 1, at least, WLD continue to provide higher accuracy in most cases. Furthermore, we found that a notable improvement is present by combining LBPRIU and VAR on both regions. This result is agrees with Musci et al. (2013). However, VAR makes no contribution to WLD, at least in region 1 with the inferior results against the generic WLD except in resolution of 3. In region 2, VAR slightly enhances the overall WLD results with the exception on the multiscale version, with the maximum increase in accuracy of about 3%.. The detail results for each TD will be presented in the next section.

4.3.2.2 Specific Accuracy of Each Texture Descriptor With VAR

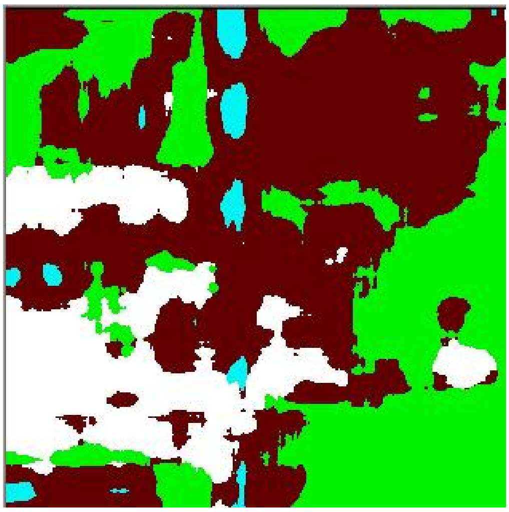
This section describes the results for current evaluated texture descriptors along with the respective confusion matrix as in 4.3.1.2. The image results for each TD with VAR in region 1 can be found in the Figure 4.21 below.



(a) Original Image (b) Ground Truth



(c) VAR Image (P: 8, R: 1) (d) LBPRIUVAR Image (P: 8, R: 1)



(e) WLDVAR Image (P: 8, R: 1)

Figure 4.21 Best Classified Image of several TD with VAR in region 1

three tables below show the confusion matrices that correspond with the above figures.

Table 4.18 Confusion Matrix of VAR (P: 8, R: 1) in Region 1

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	61704	20004	0	17215	83.48	62.38
Orchard	1151	100259	209	14261	73.83	86.52
Water	9868	0	12322	31401	98.33	22.99
Barren	1196	15530	0	74871	54.35	81.74
Overall Accuracy						**69.21**
Kappa Value						**0.57**

Table 4.19 Confusion Matrix of LBPRIUVAR (P: 8, R: 1) in Region 1

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	65205	23475	0	10243	76.21	65.91
Orchard	8333	98962	434	8151	73.43	85.40
Water	6638	0	18337	28616	97.69	34.22
Barren	5389	12341	0	73867	61.11	80.64
Overall Accuracy						**71.22**
Kappa Value						**0.60**

Table 4.20 Confusion Matrix of WLDVAR (P: 8, R: 1) in Region 1

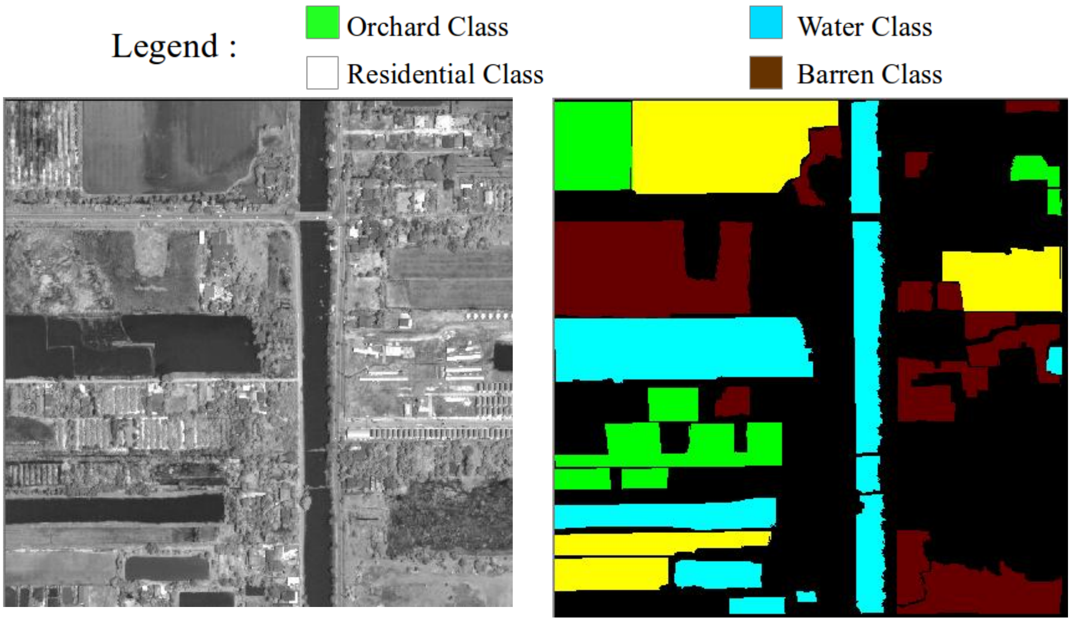
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	84113	3870	0	10940	94.69	85.03
Orchard	232	101572	113	13963	89.49	87.65
Water	4481	0	13611	35499	99.18	25.40
Barren	0	8058	0	83539	58.04	91.20
Overall Accuracy						**78.57**
Kappa Value						**0.70**

The accuracy pattern of all TD in this region is similar with the preceding analysis. That the highest accuracy is on the orchard following by the housing estate. However, even the highest accuracy of WLDVAR in current analysis is still inferior to the WLD in previous analysis. Both barren and water have lowest recognition point with a high number of pixels that are misclassified as these classes. This lowers their respective accuracy.

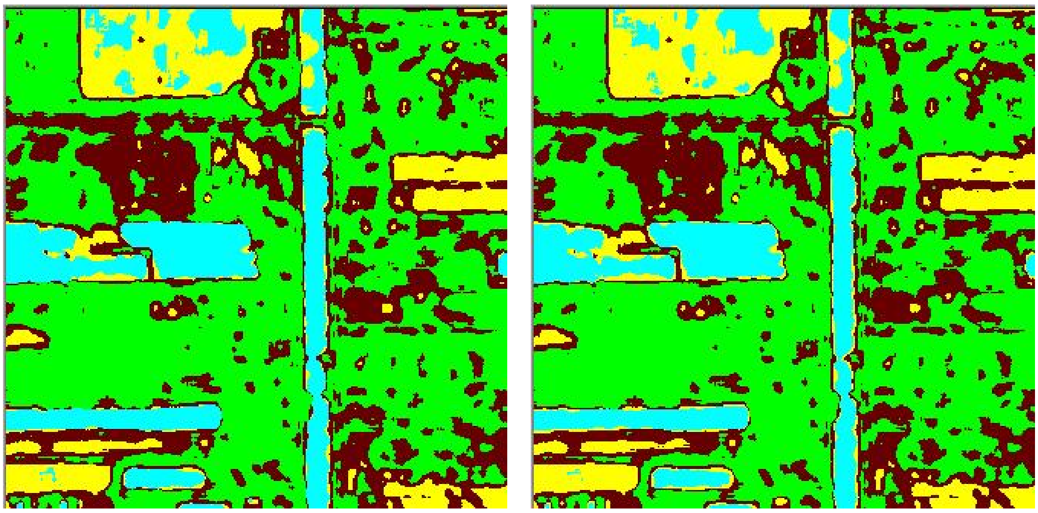
Even though the pattern is similar, there are some differences. In the house texture class, it is found that the user accuracy is generally higher in all TD than in the previous experiment with exception of WLD (which is highest). However, the user accuracy reduction is also occurred in water class with no current evaluated TD achieve the perfect user accuracy. In the barren area, it is found that lower numbers of house and orchard are misclassified as this class, but more in water classes. Thus the accuracy of the barren does not really change. Last, in the orchard area we found a larger number of pixel in other texture class except water are misclassified as this class. This reduces the user accuracy of this class.

The general confusion pattern also same with the last analysis but with several differences. There are fewer pixels of house area that recognized as the barren area and vice versa, which signifies that these evaluated TDs are good in recognizing these different classes. The bigger confusion also occurs between house and orchard than in region 1 which lower these textures accuracy as well.

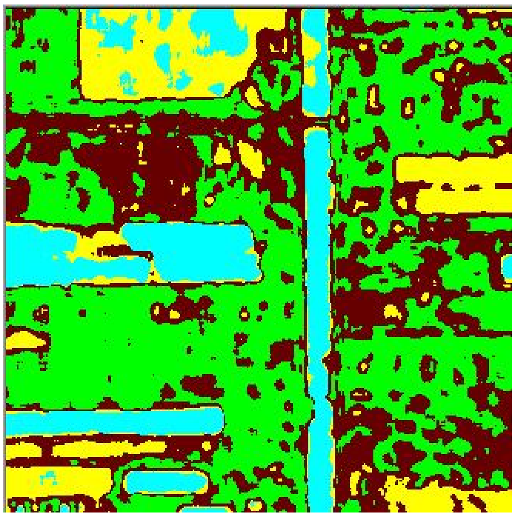
For the result in second region, the image results for each TD with VAR can be found in Figure 4.22.



(a) Original Image (b) Ground Truth



(c) VAR Image (P: 8, R: 1) (d) LBPRIUVAR Image (P: 24, R: 3)



(e) WLD Image (P: 8, R: 1)

Figure 4.22 Best Classified Image of several TD with VAR in region 2

Three tables below show the corresponding confusion matrixes in the above figure.

Table 4.21 Confusion Matrix of VAR (P: 8, R: 1) in Region 2

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	65600	0	0	2759	56.20	95.96
Water	4336	100552	15774	21153	85.89	70.90
Crop	1841	16519	75690	20708	72.34	65.96
Barren	44952	0	13165	77769	63.54	57.23
Overall Accuracy						**69.36**
Kappa Value						**0.59**

Table 4.22 Confusion Matrix of LBPRIUVAR (P: 24, R: 3) in Region 2

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	65609	0	0	2750	56.53	95.98
Water	3438	96633	22149	19595	87.74	68.14
Crop	1818	13506	78052	21382	69.01	68.01
Barren	45205	0	12900	77781	64.01	57.24
Overall Accuracy						**69.02**
Kappa Value						**0.59**

Table 4.23 Confusion Matrix of WLDVAR (P: 8, R: 1) in Region 2

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	61240	0	19	7100	66.91	89.59
Water	602	107817	20472	12924	86.17	76.03
Crop	410	17300	81838	15210	67.77	71.31
Barren	29279	0	18437	88170	71.45	64.89
Overall Accuracy						**73.58**
Kappa Value						**0.64**

The pattern in the second region also similar with the previous analysis with the best recognition being water class. However, water class accuracy is almost equal with the orchard class. This signifies that orchard class has higher average accuracy than the result from previous TDs. Both crop and barren has the lowest accuracy, but the average accuracy of crop class also higher than previous TDs can achieved. Overall, the average results of all texture recognition is better than previous TDs, especially LBPRIUVAR and WLDVAR with better accuracy than the generic version of these TDs.

Besides the general improvement on orchard and crop texture class. There is also a reduction of the number of misclassified pixels overall. There are fewer pixels from other texture area that are misclassified as crop area which increases their respective accuracy. A similar reduction also occurs on the barren area, where less number of orchard area are misclassified as barren area than in region 1.

The pattern of confusion also similar but with several improvements, specifically that lower confusion is found between the barren and orchard compared with previous TD. The confusion between crop and water also found to be decreased which increases their both accuracy level.

4.3.2.3 Multiresolution Analysis

Following the similar procedure on 4.3.1.3, we obtain the results for the application of current evaluated texture descriptors shown on Figure 4.23 belows

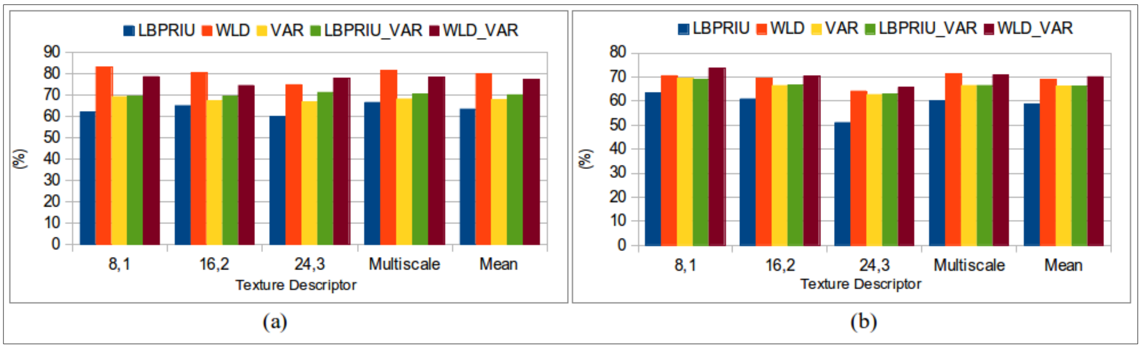


Figure 4.23 Graph Results of several resolutions of TD, Region 1 (a) and Region 2 (b)

From the graphs above, we can see a similar pattern to the previous multiresolution analysis, namely that using different spatial resolution will generate distinct results, and the multiscale version generates the results close to the mean of individual resolutions.

4.3.2.4 Computation Time Comparison

Quantizing and actual classification classification running time of the VAR, WLDVAR and LBPRIUVAR were compared, along with the last results of WLD and LBP. For reference, Figure 4.24 shows the graph of running time for currently evaluated texture descriptors.

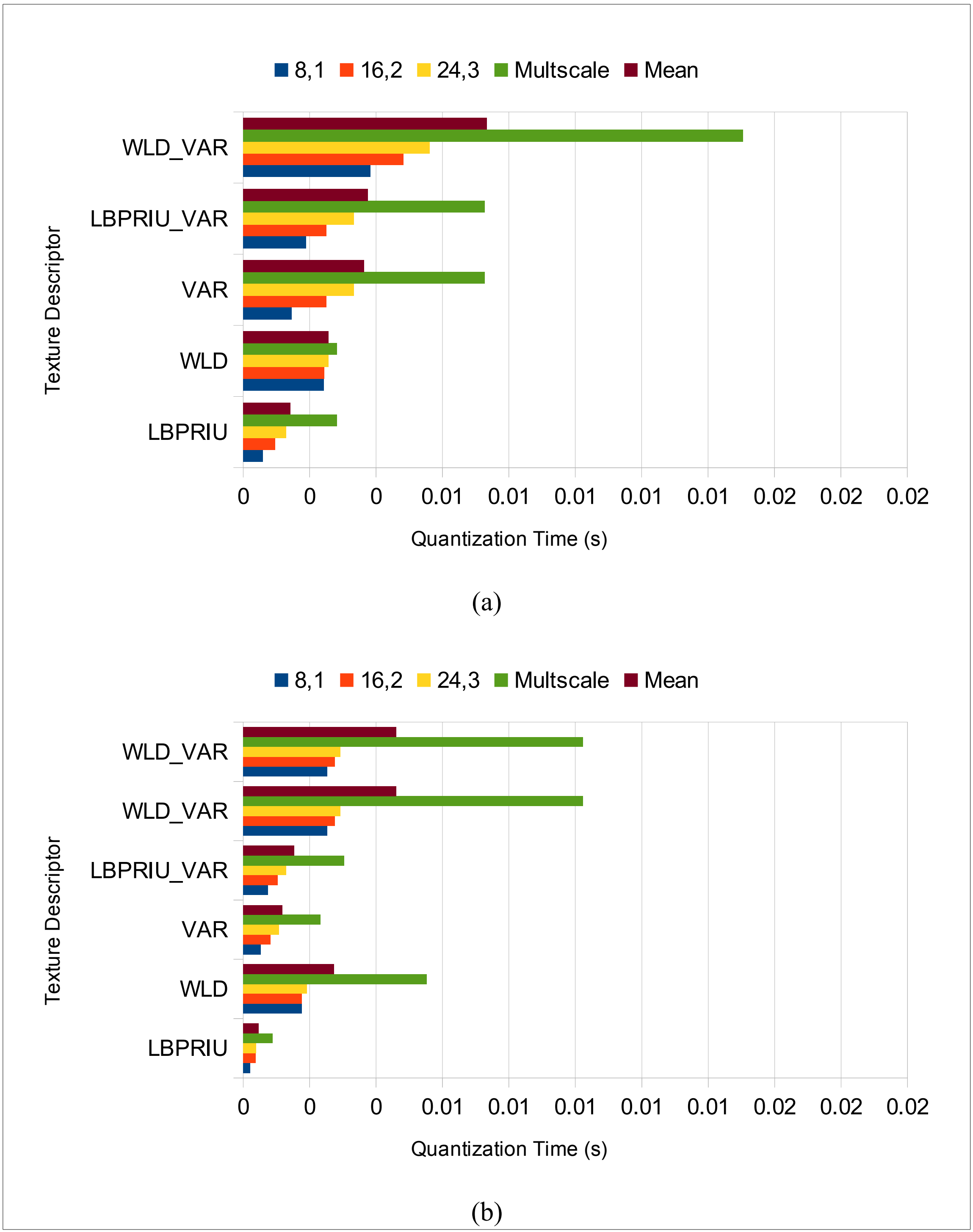
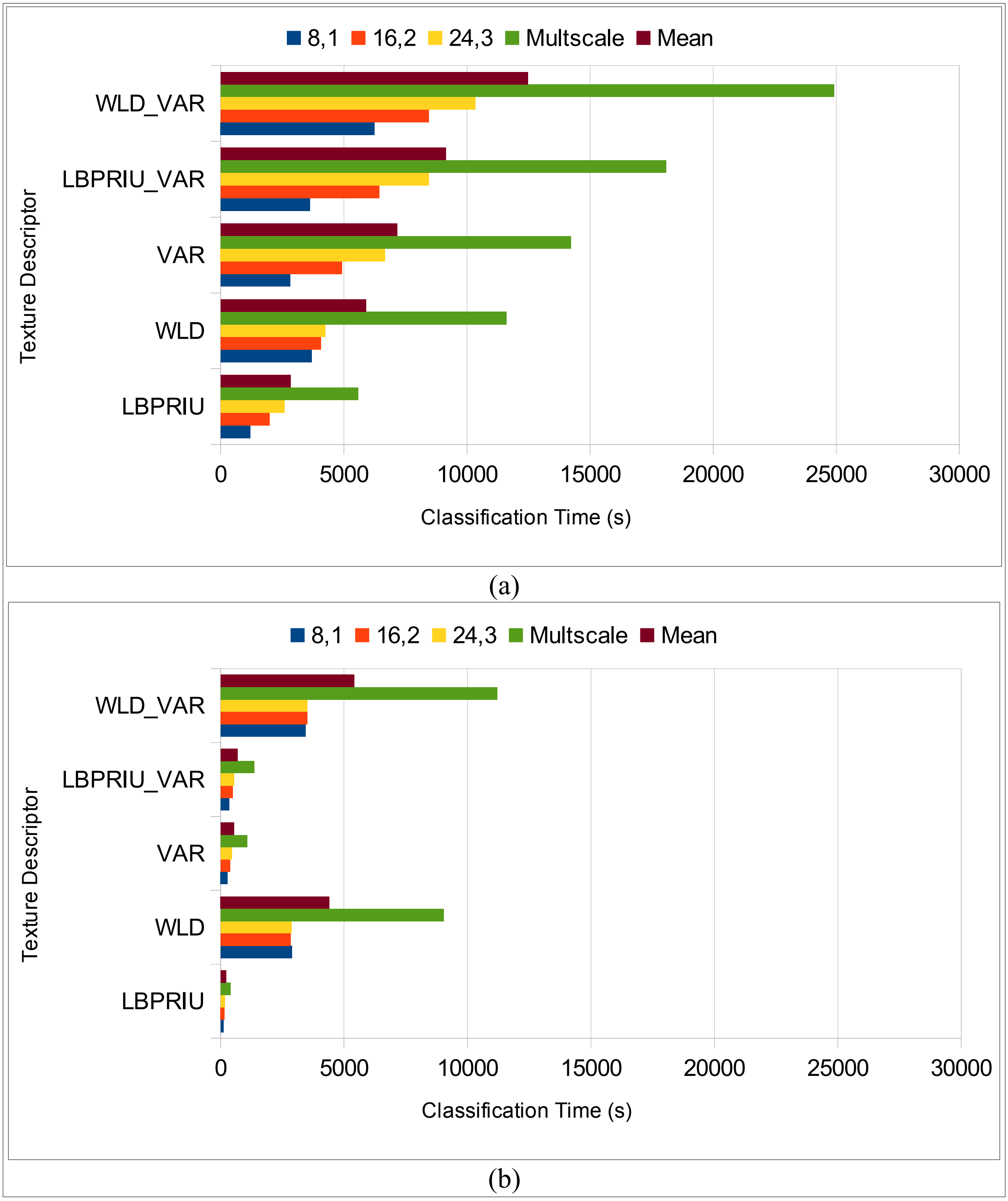


Figure **4.24** Quantizing time of several TD includes combination of VAR, (a) region 1 and (b) region 2

In these results, we can see the similar pattern between several scales of current evaluated TD, with scale of 1 as the fastest and multiresolution as the slowest. The results also reveal that running time for VAR Texture is noticeably different between two regions, for all of its scales. In region1, VAR quantizing time is almost identical with WLD which contrasts with the results on region 2, where VAR computational time is lower than WLD and close to LBPRIU. The other obvious result is that

LBPRIU_VAR clearly needs more time than basic LBPRIU or VAR ,and WLDVAR take longest time of all texture descriptors. For the running time of texture-based classification, we can see the results in the Figure 2.4.



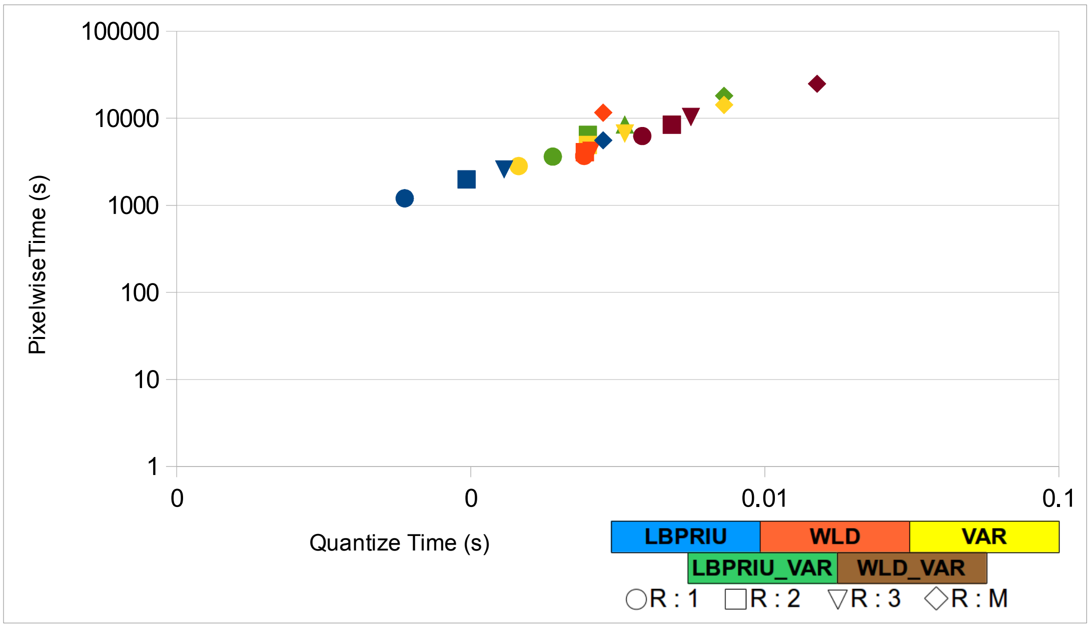
(a)

(b)

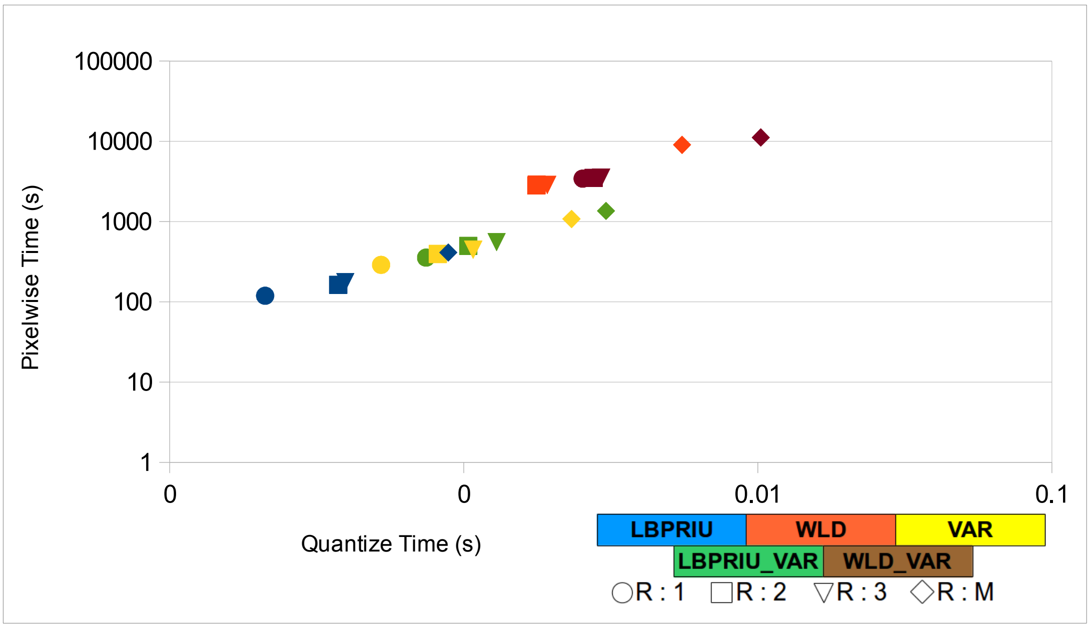
Figure 4.25 Classification time of several TD including combination of VAR, (a) region 1 and (b) region 2

A pattern similar to the quantizing time is also present in the result charts above, where the different of running time between VAR and WLD is small in region 1, and large in region 2 with a difference of more than 2000 seconds (more than half an hour). The large running time in region 1 also affects the running time of LBPRIUVAR and also

WLD. However in region 2, the running time of LBPRIUVAR and WLD are not different. Lastly, the WLDVAR is the slowest over all of current available texture descriptors. To see the contribution of the quantization time to overall classification time. Figure 4.26 below shows the scatter-plot of classification versus quantization for each evaluated TD.



(a)



(b)

Figure 4.26 Scatter-plot of Quantizing versus Classification running time of Each TD Including VAR. (a) Region 1 and (b) Region 2

As can be seen that the pattern is similar with the pattern in section 4.3.2.4. that all TD classification time almost grows linearly with respect to their quantization time.

LBPRIUVAR progression is similar with both LBP and VAR create the linear pattern. For the WLDVAR, the relationship is similar with WLD in that there is no increment in any resolution with the exception in multiresolution, where the bin count is three times larger than single resolution. This increases both quantization and classification time.

4.4 Analysis and Discussion

In this thesis, texture based classification is applied to high resolution Quickbird imagery from Pathumthani province. We used two regions differentiated by the characteristics of the texture contained in each of region. Region 1 generally is comprised of well defined textures excluding the residential site, which is not homogeneous yet essential to test the robustness of the TD capability to represent this texture. On the other hand, although region 2 has several areas with well defined texture, some other areas are affected by random changes of brightness. The orchard and barren area are examples. Furthermore, contrast change is also present in some texture areas. This make the classification in this region more difficult than in region 1, where there is little contrast change. Thus in this region, the contrast invariant of each TD can be tested optimally. Also, the texture units in region 1 are larger than those in region 2. Thus we need larger neighborhood and sample sizes to capture these textures.

4.4.1 Parameter Selection

In the parameter selection, since there are no conventional rules to select the parameters, this thesis relied on an experimental process by testing several possible parameters and assessing the accuracy of results. Figure 4.27 presents the part of both regions, where respective samples are taken with different size.

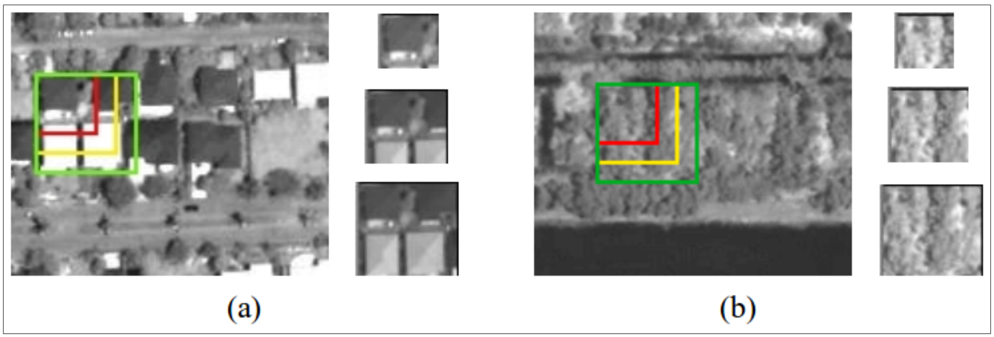


Figure 4.27 Different sample size on each region, (a) portion of first region and (b) second region.

Note that in image (a), the sample size of 30 x 30 (red) is not reliable for texture samples representing the example texture class, which is residential sites. This is because the coverage area using this size is not wide enough to capture one occurrence of a residential texton, i.e a single house. Extending the sample size to 50 x 50 (which is the largest sample size on the figure (a)), will be more appropriate, since it captures more houses. This sample size leads to better texture representation. On the other hand, in figure (b), it can be seen that a sample size of 30 x 30 is wide enough to capture the basic pattern of this texture (crop). Extending the sample size will generally just repeat the occurrence of texture units.

We can observe the effect of using the different sample sizes on the image results of this process. Table A.1 and Table A.2 in Appendix A show the different appearance of the results utilizing different sample size, and it can be seen, that in region 1, the usage of sample size 30 x 30 will result in low recognition on residential site. However, the result is better when utilizing 50 x 50. This contrasts with region 2 where smaller sample size is more appropriate. Further details of classification progression results utilizing different sample sizes can be seen in Table B.1 and Table B.2 in Appendix B.

On the ringsize parameter selection results, we found the similar pattern with the results of sample selection process, as can be seen on the Table B.3 and Table B.4, where on the region 1, a wider ringsize produces smoother results than the smaller which is sparse, and vice versa in region 2. Again, the texture characteristics of the image play an important role in determining the best value for this parameter. In region 1, the wider ringsize is more appropriate to take the texture samples, especially in residential sites that have wide area in this region. In region 2, a smaller ringsize is more appropriate Table B.3 and Table B.4 present details that confirm this analysis.

The previous analysis about sample and ringsize shows that different size of the texture units determines the extension for both of these parameters. Both sample and ring size need to be big enough to capture the occurrence of the texture, since the results shows the lower accuracy in the region 1 when the extension the sample and ringsize is not big enough compared to the results with adequate size of both of this parameters. However, the size of both ringsize and sample size should not be larger than necessary. In region 2, where texture unit is smaller, using big sample and ringsize will lower the accuracy compared with the smaller size that is already able to capture one occurrence of the texture. Note that larger ringsize will also produce slower running time.

The VAR '*max*' value selection to quantize the VAR texture descriptor also yielded the distinct results, that is 15 for region 1 and 7 for region 2, This result signifies that different contrast level existed within available texture classes where the contrast change is higher in region 1and lower in region 2. This contrast level is associated with the standard deviation in the images as shown in the Table 3.2, where this standard deviation actually represent the variation of the pixel brightness under a texture class and is generally higher in first region compared with the second region. Since the '*max*' value determines the range for each *bin* in quantization process as shown in Figure 3.8, selecting the '*max*' value as close as to the maximum variation in each image is needed to make the distribution of members for each bin uniform.

In the distance metric selection, it is found that Bhatacharya distance was the best distance metric in term of accuracy. This result is consistent with Jenicka and Suruliandi. (2011). However, this step can be considered as an ancillary process since the difference between the accuracy of three metrics is small (less than 1%).

Last is the selection of parameters for WLD. We found that slightly lower values of T,M and S were appropriate for region 1 versus region 2. This indicates that more bins are required in region 2 over region 1 to accommodate the texture pattern over their area. This is basically because of the higher contrast between textures that is present in region 2 which can be seen in the Table 3.2. Generally in each region, the variation of SD value between texture classes is slightly higher in region 2 compared in region 1.

Which is more than 1 in the region 2 compared with less than 0.5 in region 1. With the exception of residential area for region 1 and orchard in region 2 since these texture classes have distinct SD compared with other texture classes.

The WLD case is similar to the VAR, since WLD quantizes the texture using differential excitation, that in fact also measures the relative contrast different between evaluated pixel and its neighborhood. WLD utilizes the central value as a normalizer (Eq. 2.5) where VAR uses mean of pixels (Eq. 2.4). However, the VAR *bin* count is fixed (8) thus the optimization is conducted by selecting the *'max'* value appropriately as described in earlier analysis, to get a relatively uniform number for each bin. In WLD in other hand, the bin count can be controlled by using the specific value of T, M and S. An image with larger class SD values will need more bins because the image values vary more widely.

4.4.2 Comparison of LBP, LBPRIU and WLD

Comparison experiments of WLD over LBP and its extension LBPRIU, reveals that WLD generally outperforms the other TDs in terms of accuracy in both regions, with the exception in region 1 that WLD attains the maximum accuracy of 83.16% compares with 84.49% of LBP. However, the large running time of LBP in this resolution makes it impractical. In region 2, we found that WLD exceeds the capability of LBP and LBPRIU consistently over all of texture classes. Also we obtain an average differences of more than 8% between these two TDs over both of regions.

Analyzing the patterns of recognition for each texture class, we found that generally the confusion occurred when recognizing barren class in both regions. Other texture classes are misclassified as barren area. This may comes from the texture in barren which is not uniform and well defined compared with the other textures. The texture suffers random changes as well, especially housing area in region 1 which has the highest confusion compared with other texture classes. This confusion is not found in the orchard area which is relatively well defined texture, since it is the controlled area under the farmer management. This lowers its confusion as shown in the respective results of section 4.3.1.2 compared with other classes. The other finding is that the similar texture characteristic between texture classes leads to the confusion of all evaluated TDs when recognizing this class, especially between crop and water in region 2.

As an additional finding, we do not see any improvement of LBPRIU over LBP. Specifically in region 1, LBPRIU yields only about 66.52% maximum accuracy, which is inferior to the results from both LBP and WLD. This can be explained as follows. LBPRIU, in its mapping process to enable its rotation invariance, will place the non-uniform values that originally are in distinct bin locations into a single miscellaneous bin location. This actually makes the LBPRIU lose some of the discrimination capabilities of original LBP, thus lowering its accuracy compared to LBP in this region, where almost no rotation in texture occurs as shown in Figure 4.28 below.

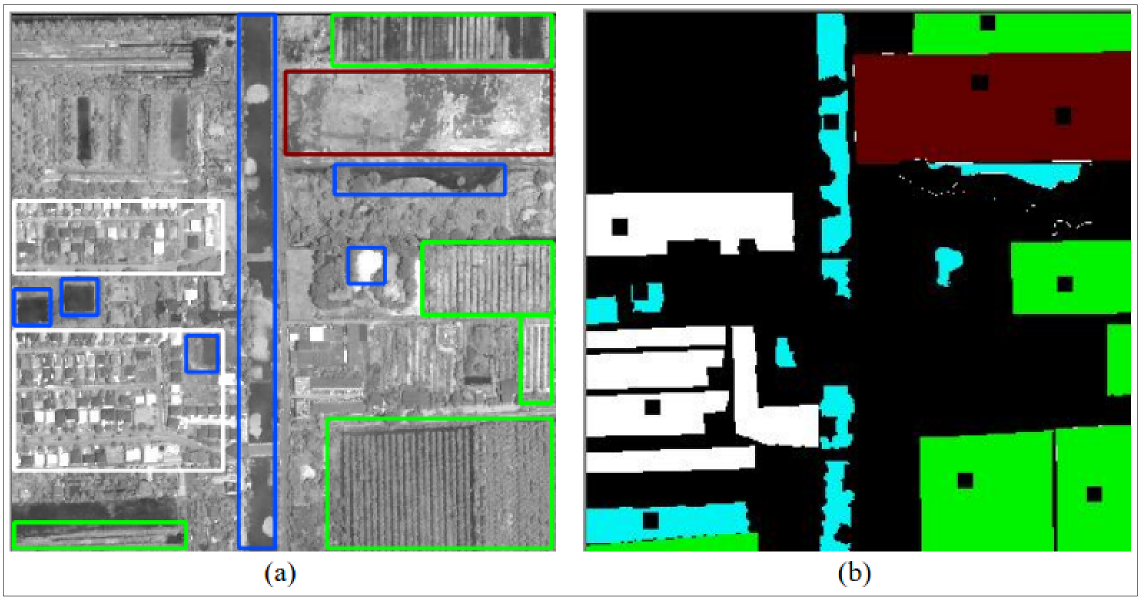

(a) (b)

Figure 4.28 Region 2 overlaid with area of texture (a) and respective ground truth with areas where samples are taken (b)

Notice that almost all areas of the orchard class (green boxes) have a vertical striped texture except for a small area in the bottom left. Similarly, the buildings and the roads in the housing areas all show the same orientation. In region 2 however, we can see that LBPRIU results are slightly better than LBP, with the 59.05% mean of accuracy compared with 54.515% of LBP as shown in Table 4.9.

4.4.3 Contribution of VAR over WLD and Comparison against LBPRIUVAR

In the experiment concatenating VAR with WLD and LBP, results show that WLDVAR is able to surpass the capability of both VAR and LBPRIUVAR. This mainly because WLDVAR retains the high discrimination of WLD in texture quantizing. With regard to the contribution of VAR to WLD, we don't see any improvement made for the result of WLDVAR in region 1, compared with the result of WLD which more accurate. This result is related to the condition of the texture in region 1 in terms of contrast change. VAR is expected to enhance the WLD capability to tackle the illumination changes that are present, but in region 1, there are no obvious contrast changes, thus VAR does not give any improvement in WLDVAR. Instead it lowers the discrimination capability originating from WLD as described in section 4.3.2.2 However in region 2, there are several contrast changes influencing the available texture classes. Figure 4.29 illustrates one example of the VAR contribution over WLD in classifying the contrast affected texture area.

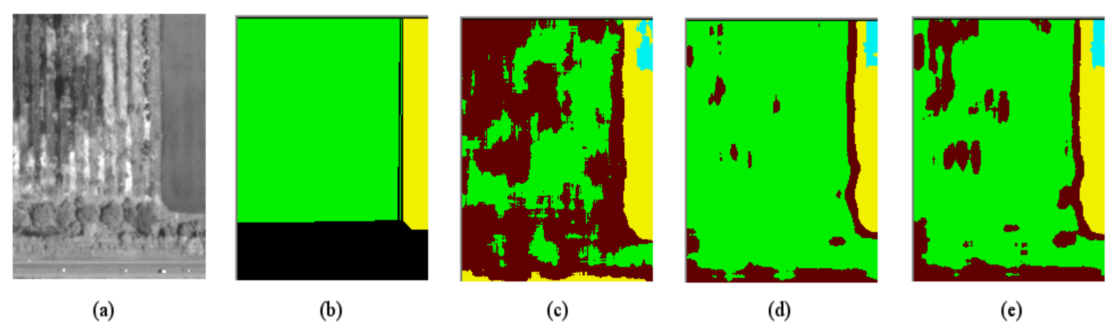


Figure 4.29 VAR contribution over WLD, (a) Texture area affected by contrast change, (b) Corresponding ground truth, (c) WLD result, (d) VAR result, (e) WLDVAR result

Notice that the results of WLD are sparse and include spurious pixels. This because the heavy contrast change affects the part of this texture area, and leads WLD to mis-classify this area as the incorrect texture class. Conversely, VAR is able to produce smooth results which is confirmed that VAR is able to handle the contrast change better than WLD. Consequently joining VAR to WLD will aid the WLD on recognizing this area that is shown in (b), with plain and smoother results than original WLD.

This result is also substantiated with the results in Table A.2, where the improvement on orchard texture area occurs with a mean 78% accuracy compared with WLD that is 76%. Furthermore we can also see that VAR results are comparable with WLD in region 2 as presented on Table 4.17. Although the results are still slightly lower than WLD, yet this indicates VAR can work well in this second region. On the results of LBPRIUVAR itself, we see constant improvement compared to plain LBPRIU in both regions. This corroborates the result from original paper proposing this TD (Musci et al., 2013). However, the accuracy is still less than the WLDVAR result.

Observing the class by class pattern of these TDs, we found that the pattern is similar with preceding evaluated TDs. That is, the confusion is mainly occurred in recognizing the barren area for both region. The VAR TDs pattern is similar with both LBP and LBPRIU but has better user accuracy in recognizing the housing area in region 1 and average accuracy to orchard area in region 2. This suggest that VAR is better in recognizing the texture area where the contrast variation is high. Specifically in region 2, we found the improvement in user accuracy of several classes and also the reduction of the confusion between texture classes as explained in 4.3.2.2, especially the confusion between water and crop which lower than results from previous TDs. The other finding is that the pattern of texture recognition for both LBPRIUVAR and WLDVAR comes from the two original TDs which contribute to these TDs.

4.4.4 Multiresolution Analysis

Raising the spatial scale from 1 to 2 increases the accuracy found in LBP and LBPRIU. This indicates that higher number neighboring (P) and wider spatial relationship (R) is needed when capturing the local spatial pattern using these TDs. However, we did not find a similar pattern in the WLD. Instead increasing the scale (the window size) lowered its accuracy. This accuracy difference over different couple of P and R is consistent with the previous application in several resolutions over these TDs as described in section 3.5.1. In region 1, all of TD suffer a reduction on accuracy results as scale increases. This shows that spatial resolution of 1 is most appropriate in this region.

The other important conclusion is that using the the multiscale version of each TD will not effectively increase the accuracy over the single resolution version of each TD. Instead it generally produce an average accuracy value, over all of its spatial resolutions including the WLD. This can be explained since the multiscale resolution actually combine the results of all of this three spatial resolutions (1, 2 and 3), hence resulting in average value of them. This spatial resolution might be considered to be used to get the general capability of the related texture descriptor, although the computational time will definitely be higher than for a single spatial resolution of this TD.

The finding that the multiscale version giving no improvement to all TDs is consistent with (Musci et al., 2013) with the exception to WLDVAR and WLD. For WLDVAR, it is relatively a novel TD. For WLD, our results conflicts with the results in (Chen et al., 2010) where the multiscale version of WLD generally achieved higher classification accuracy over the single resolution version of it. There may be several explanations:

1. The image used in (Chen et al., 2010) is the synthetic image which has the different characteristics in terms of the texture nature and the clearness of the boundary between the textures. Generally, the texture in the remotely sensing imagery has natural changes in their texture as shown in Figure 3.3 and explained in section 3.1.1. This difference may affect the final classification result.
2. A Support Vector Machine (SVM) was used in original paper versus the pixel-histogram comparison in this research.
3. The samples used by Chen et al were more extensive. Since their image was synthetic, samples are relatively easy to acquire. More than 4000 image samples were used in their original experiments. This impossible to be applied in this research due to limited area to be used as the sample area.

To close this multiresolution analysis. we also found that WLD accuracy is quite consistent across all evaluated spatial resolutions. This signifies that WLD is robust against the spatial resolution changes.

4.4.5 Computation Time

In assessing the running time of quantizing texture, the differences are actually small (less than 1 second) because the sample and ringsize (in training process) is small. Thus the different complexity of each TD does not greatly affect the quantization time. However, in the classification running time, we can observe the quite differences since the quantization and scanning process is done repeatedly. To analyze the reason behind the clear differences between each TDs elapsed classification time, the pseudocode of each TD will be the important item since it defines the algorithm of each TD. The pseudocode of each TD and the other substantial algorithm can be found in earlier section of 4.1 and are discussed in the following analysis.

Analyzing the correlation between the quantization time and classification time, we found that generally all TDs shows classification time that increases in a linear way as shown in Figure 4.20 and Figure 4.26. This shows that the quantization time dominates the classification time compared with the scanning process as the other main process as shown in Figure 4.11. The exception occurs in LBP which shows an exponential relationship. A small escalation in quantization time makes the classification time of this TDs grow up exponentially. Clearly this growth does not originate from the small progression of quantization process but the distance measurement, which needs to scan the exponentially growing number of bins in LBP as scale increases.

Specifically in texture-based classification, LBP time is comparable with the LBPRIU, and this two TD are the fastest among other TD in spatial resolution of 1. However, in larger spatial resolutions, the LBP running time is very slow. Based on the results from Table C.3 for region 1, and Table C.4 for region 2, the LBP computational time increases exponentially, starting from 829 seconds in spatial resolution 1, to be more than 8000 seconds in spatial resolution 2, which is ten times higher. This keep increasing with up to 437522 second (close to 5 days). This slow running time makes LBP totally unpractical in real applications for resolutions greater than 1. The reason behind this slowness is the large number of bins in the histogram of this TD on this spatial resolution (line 2 in Figure 4.1). Thus the scanning process to calculate the distance metric (line 3 for each distance metrics in Figure 4.8, Figure 4.9 and Figure 4.10) will makes the running time skyrocket.

The slowness of LBP in multiscale version on both regions comes from the huge number of bins as previously mentioned in analysis of quantization and classification time correlation. Which is 2 to the power of (exponential) over its spatial resolution, (line 2 in Figure 4.1). The concatenation process actually needs to scan all of the bins from the each spatial resolution (line 2 to 6 in Figure 4.7). Due to the huge number of bins, this process will be slower than a single spatial resolution of this TD, where iteration process is not required. Multiscale version of LBP actually produces a histogram within more than 16 million bins. For WLD, we found different parameters of T, M and S between two regions, which directly regulate the bin count of WLD (line 25 in Figure 4.5), does not affect the WLD quantizing running time significantly. However, this higher number of bins will take longer to iterate, which will directly affect classification time, since this process needs to scan all of the bins for each TD similarly in the multiscale case of TD.

Regarding the different time running in different spatial resolution of each TD, with the exception of WLD, results shows that increasing the spatial resolution subsequently will also raise the computation time. This is due to the larger number of neighbors causing the higher required time to fetch this increasing neighbor pixels from actual images. Obviously multiresolution will produce the largest computational time over single resolution since it combines all of the single resolution computation time. The WLD itself has the nearly constant computational time regardless of different spatial relationship with the exception in multiresolution, since in the multiresolution the bin count of WLD is three times than single resolution of this TD. The nearly constant speed of WLD in single resolution indicates that WLD does not depend on the number of involved neighbors unlike the other TDs, because the number of bins does not increase with neighborhood size.

In overall, it can be seen that all TD running times is differ by region, that is slower in region 1 than in region 2. This mainly because the difference of the ringsize, which is larger in the region 1 than region 2. The larger ringsize will require all TD to do more intensive per-pixel calculations in the classification process since the evaluated texture area is bigger. Furthermore in the VAR TD, we can further observed that a noticeable classification time difference exists between two regions. This results mainly comes from distinct quantizing time on each region which influences the classification on the sample quantization process (line 5 in Figure 4.11). This process occurs repetitively with respect to image dimension and ringsize. Finally, VAR contributes to the computational time for LBPRIU and WLD, and this confirmed on the results of Table C.7 and Table C.8 for region 1 and region 2 respectively, where running time of LBPRIUVAR and WLDVAR are larger than original LBPRIU and WLD.

CHAPTER 5 CONCLUSION

5.1 Conclusion

In this research, texture based classification was conducted upon high resolution remotely sensed imagery from Pathumthani province in Thailand, to solve the limitation of pixel based classification which can not accommodate the great disparity in the selected class due to high resolution of the evaluated imagery. Since the texture is clearer in higher resolution, this feature can be utilized to attain better classification results.

We employed a supervised per-pixel classification approach to classify the texture classes. The relatively new robust Weber Local Descriptor was utilized as our main texture descriptor. We compared its relative performance against well known LBP texture descriptor and further assessed the contribution of VAR texture descriptor to WLD to neutralize illumination differences within the texture area.

Before we did the actual classification process, we used a sequence of prior classifications to select the set of optimum parameters required in final classification process. We experimentally found that the sample size and ringsize are related to the available textures in each test region where the classification procedure will be conducted. Region 1 required larger sample and ringsize to identified the available texture class than region 2, where a smaller size of these two parameters was more appropriate. Furthermore as expected. utilizing the ringsize with the size half of sample size, resulted in the average accuracy value on the ringsize parameter tuning process. This size can further be utilized in the real application to get the rough performance of selected TD, before doing actual ringsize parameter tuning. Other findings are that Bhatacharya distance was the best distance metric, and that contrast within a texture class area also influences the VAR and WLD parameter selection directly.

In the comparison of WLD over LBP and LBPRIU, we found that WLD appears to be robust to identify residential sites, and also accurate in recognizing the well defined active orchard area, with the best accuracy of 83%. Overall, this TD outperformed both LBP and LBPRIU. We further found that generic LBP produced results as good as LBPRIU signifying that adding rotation invariance does not affect the accuracy of LBP substantially, at least in our images, where there are few instances of rotated textures. We also found consistent differences in accuracy of TD across the regions. Since the level of contrast change is greater in region 2, this region had consistently lower accuracy with all of TD with the best accuracy of 71.18% possessed by WLD.

By concatenating the VAR with WLD, we obtain the results that the VAR contribution is directly affected by the characteristics of the region. In region 1, we see no enhancement in the accuracy of WLDVAR over WLD. In fact adding VAR lowered accuracy. On the other hand in region 2, we found a slight improvement from VAR combined with WLD in terms of accuracy, especially in the accuracy for recognizing the active orchard texture class where the contrast change occurs. This indicates that VAR is suitable to be concatenated to WLD where the illumination changes are present. Furthermore, comparing WLDVAR with LBPRIUVAR, we see a constant superiority of WLDVAR over LBPRIUVAR in most of evaluated spatial resolutions.

Exploring the different results of several spatial resolutions of each TD, we see that each TD results in different accuracy with its corresponding spatial resolution. The accuracy of results generally decreased by increasing the spatial resolution, with some exceptions in the region 1, where both LBP and LBPRIU gain the accuracy enhancement in spatial resolution of 2. Moreover, the multiscale version yields the average accuracy with respect to mean of its different spatial resolution results. This results can be useful in initial observation, to see the general capability of the selected TD used in classification process. However, the running time of this spatial resolution is significantly higher than a single spatial resolution.

The running time of each TD was found differ according to their computation complexity. WLD TD runs in nearly constant time though it is slower than LBP. On average it required about 1.5 hours to do classification on 1024 x 1024 images with four evaluated texture classes. Observing the relationship between the quantization and classification time, generally all TDs shows the positive relationship that classification time grows linearly with respect to the escalation of quantization time, with the exception of the LBP, which shows an exponential curve due to its exponential growth of bin count. That makes the distance measurement time is more dominant against the quantization time in overall classification time.

Besides the classification running time, we found that bin count also be another factor that contributes in computation time, especially toward the required scanning procedure on each of this TD generation, and in classification process. As addition, we also see the introduction of VAR in WLDVAR and LBPRIUVAR increases their running time compared with original WLD and LBPRIU.

There are several issues that showed be further investigated to enhance the potential performance of this WLD texture descriptor. One improvement would be to obtain more reliable ground truth, since in this thesis, several classes have the quite differentiable texture variation within one class, especially in the barren class and some crop areas. This will make the classification less accurate. Getting better ground truth, especially under the site where the texture is substantially distinct between the available texture class will ease the discrimination process and should raise the accuracy results of overall TD. Note that the expert was not doing the identification based his interpretation only on texture but also color and context, thus the ground-truth might contained several textures under one class.

In the parameter selection, it can be considered to extend the sample and ring size more than that used in this thesis when the texture unit within image is larger than ringsize used in this research, even though it will be slower. It is possible that more accurate results will be attained because more texture is captured in this wider size. For the VAR parameter selection, it may useful to gather more samples, so that we can automatically select a value for *'max'* based on the input data.

Specifically on the VAR concatenation process to enhance the WLD, there are at least two different approaches worth exploring. One is to incorporate VAR in the WLD two dimension histogram formation, by treating it as a weight factor instead using one weight in accumulation process, and the second is to build a three dimensional histogram, with respect to differential excitation, gradient orientation, and VAR.

However the latter will substantially affect the complexity of WLD. Furthermore in the future, we would like to explore the capability of WLD in unsupervised segmentation. With the promising results of WLD on texture based classification, this texture descriptor is applicable in the real life application of classification of high resolution imagery, since it provides superior accuracy over currently available texture descriptors.

5.2 Summary of Contribution

1. In the best of our knowledge, this is the first research that attempts to assess the capability of WLD to quantize the texture in remote sensing images. With the high accuracy result through per-pixel texture-based classification results, we believe this method is promising and applicable in real world applications.
2. This research reveals some observation results which can be used when performing the texture-based classification, such as sample size, ringsize and several other parameters for each of TD
3. We investigate relative performance of WLD over several binary texture descriptors and the state of the art LBPRIUVAR in remote sensing images.
4. WLD is very stable in term of accuracy and running time over different parameter sets.
5. By concatenating WLD with VAR, we were able to see its contribution to the classification process upon selected study area with different texture characteristics.
6. This is the first investigation and comparison of multiresolution and single resolution performance of WLD and WLDVAR in remotely sensed imagery classification.

5.3 Limitation of Work

1. Since the ground truth is obtained from only one source, thus we can not do validation upon existing ground truth.
2. Only a panchromatic image is used instead of multiple multispectral bands, since it has significantly better resolution.
3. In this thesis, only four texture classes were chosen over the rest of available identified texture classes, based on their texture homogeneity.
4. We limited the sample size extension and several parameters to keep the program running time within reasonable limits.
5. The comparison is applied solely to the family of binary coding descriptor, since these methods are relatively fast with generally high accuracy results.

5.4 Future Research

1. Obtain more available ground truth with respective taken date of study area, to get more reliable validation data on accuracy assessment.
2. Use more classes with distinct texture class to see the WLD capability to deal with more well defined and distinct texture classes.
3. Extend the values of parameters in the parameter selection, to explore more texture area and discrimination for corresponding TD.
4. Using different approach on combining the VAR to WLD in order to enhance the WLD more significantly.
5. Incorporate color into the texture classification, to possibly enhance the texture quantization and classification process
6. Explore the capability of WLD in unsupervised segmentation, to see its capability to deal with all available textures in selected region.

7. Explore modification to the algorithm which can speed up the process, especially on classification. E.g. fast rejection of very distant classes (by approximating the global difference between histograms) or saving and reusing intermediate results within a neighborhood (in the ring) as the window moves.

REFERENCES

Ahonen, T., Hadid, A. and Pietikäinen, M., "2004. Face Recognition with Local Binary Patterns", **Computer Vision-ECCV** 2004. Springer, pp. 469–481.

Ahonen, T., Rahtu, E., Ojansivu, V. and Heikkila, J., 2008. "Recognition of Blurred Faces using Local Phase Quantization", **19th International Conference**. pp. 1–4.

Baraldi, A. and Parmiggiani, F., 1995. "An Investigation of the Textural Characteristics Associated with Gray Level Cooccurrence Matrix Statistical Parameters". Geosci. Remote Sens. **IEEE Trans.** Vol. 33, pp. 293–304.

Basile Giannini, M., Merola, P. and Allegrini, A., 2013. "Texture analysis for Urban Areas Classification in High Resolution Satellite Imagery". **Appl. Remote Sens. J.** Vol. 2, 65–71.

Blaschke, T., 2010. "Object Based Image Analysis for Remote Sensing". **ISPRS J. Photogram Remote Sens.** Vol. 65, 2–16. doi:10.1016/j.isprsjprs.2009.06.004

Chang, T. and Kuo, C.-C., 1993. "Texture Analysis and Classification with Tree-Structured Wavelet Transform. Image Process", **IEEE Trans.** Vol. 2, 429–441.

Chen, J., Shan, S., He, C., Zhao, G., Pietikäinen, M., Xilin Chen. and Wen Gao, 2010. "WLD: A Robust Local Image Descriptor". **IEEE Trans. Pattern Anal. Mach. Intell**. Vol. 32, pp. 1705–1720. doi:10.1109/TPAMI.2009.155

Chen, J., Shan, S., Zhao, G., Chen, X., Gao, W. and Pietikainen, M., 2008. "A Robust Descriptor Based on Weber's Law", Computer Vision and Pattern Recognition, 2008. CVPR 2008. **IEEE Conference** pp. 1–7.

Cohen J, 1960. "A Coefficient of Agreement for Nominal Scales". **Educ. Psychol. Meas.** Vol. 20, pp. 37–46.

De O Bastos, L., Liatsis, P. and Conci, A., 2008. "A Novel Algorithm for Grey Level Co-Occurence Matrix Computation in Real Time Biomedical Image Analysis". **The 8th world Congress on Computational Mechanics (WCCM8)**.

Hasmadi, M., Pakhriazad, H.Z. and Shahrin, M.F., 2009. "Evaluating supervised and unsupervised techniques for Land Cover Mapping Using Remote Sensing Data". **Geogr. Malays. J. Soc. Space** Vol. 5, pp. 1–10.

Huang, D., Shan, C., Ardebilian, M. and Chen, L., 2011. "Facial Image Analysis Based on Local Binary Patterns: A Survey". **Submitt. IEEE Publ.**

Jenicka, S. and Suruliandi, A., 2011. "Empirical evaluation of Distance Measures For Supervised Classification of Remotely Sensed Image with Modified Multivariate Local Binary Pattern", **Emerging Trends in Electrical and Computer Technology (ICETECT), 2011 International Conference** pp. 762–767.

Kima, M., Xu, B. and Maddena, M., 2008. “Object-based Vegetation Type Mapping from an Orthorectified Multispectral IKONOS Image using Ancillary Information”. **ISPRS Conference, Vol 38**

Laliberte, A.S. and Rango, A., 2009. “Texture and Scale in Object-Based Analysis of Subdecimeter Resolution Unmanned Aerial Vehicle (UAV) Imagery”. **IEEE Trans. Geosci. Remote Sens.** Vol. 47, pp. 761–770. doi:10.1109/TGRS.2008.2009355

Li, W., Mao, K., Zhang, H. and Chai, T., 2010. “Selection of Gabor filters for Improved Texture Feature Extraction, Image Processing (ICIP)”, 2010 **17th IEEE International Conference** pp. 361–364.

Li, Z., Hayward, R., Zhang, J., Jin, H. and Walker, R., 2010. “Evaluation of Spectral and Texture Features for Object-based Vegetation Species Classification using Support Vector Machines”, **ISPRS Technical VII Symposium.** pp. 122–127.

Long III, W. and Srihann, S., 2004. “Land cover Classification of SSC Image: Unsupervised and Supervised Classification using ERDAS Imagine”, Geoscience and Remote Sensing Symposium, 2004. IGARSS’04. Proceedings. 2004 **IEEE International.** pp. 2707–2712.

Lowe, D.G., 2004. “Distinctive Image Features from Scale-Invariant Keypoints”. **Int. J. Comput. Vis.** Vol. 60, pp. 91–110.

Lucieer, A., Stein, A. and Fisher, P., 2003. “Texture-Based Segmentation of High-Resolution Remotely Sensed Imagery for Identification Of Fuzzy Objects”, **Proceedings of GeoComputation.**

M. Haralick, R., Shanmugam, K. and Its’hak D, 1973. “Textural Features for Image Classification”. **IEEE Trans. Syst. Man Cybern. SMC-3**, pp. 610–621.

Moran, E.F., 2010. “Land Cover Classification in a Complex Urban-Rural Landscape with Quickbird Imagery”. **Photogramm. Eng. Remote Sens.** Vol. 76, pp. 1159.

Musci, M., Feitosa, R.Q., Costa, G.A.O.P. and Velloso, M.L.F., 2013. “Assessment of Binary Coding Techniques for Texture Characterization in Remote Sensing Imagery”. **IEEE Geosci. Remote Sens. Lett. Vol.** 10, pp. 1607–1611. doi:10.1109/LGRS.2013.2267531

Naskar, M. and Parekh, R., 2011. “Texture Recognition in Images using Wavelets”. **Int. Conf. on Scientific Paradigm Shift in Information Technology & Management (SPSITM '11)** pp. 113-117

Ojala, T., Pietikäinen, M. and Harwood, D., 1996. “A Comparative Study of Texture Measures With Classification Based on Featured Distributions”. **Pattern Recognition.** Vol. 29, pp. 51–59.

Ojala, T., Pietikainen, M. and Maenpaa, T., 2002. “Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns”. **Pattern Anal. Mach. Intell. IEEE Trans.** Vol. 24, pp. 971–987.

Ojansivu, V. and Heikkilä, J., 2008. “Blur Insensitive Texture Classification using Local Phase Quantization”, **Image and Signal Processing. Springer,** pp. 236–243.

Ojansivu, V., Rahtu, E. and Heikkila, J., 2008. “Rotation Invariant Local Phase Quantization for Blur Insensitive Texture Analysis, Pattern Recognition”, **ICPR 2008. 19th International Conference,** pp. 1–4.

Randen, T. and Husoy, J.H., 1999. “Filtering for Texture Classification: A Comparative Study”. **Pattern Anal. Mach. Intell. IEEE Trans.** Vol. 21, 291–310.

Risojević, V., Momić, S. and Babić, Z., 2011. “Gabor Descriptors for Aerial Image Classification, Adaptive and Natural Computing Algorithms”. **Springer,** pp. 51–60.

Sehgal, S., 2012. “Remotely Sensed LANDSAT Image Classification Using Neural Network Approaches”. **Int. J. Eng. Res. Appl.**

Shi, B.E., 1998. “Gabor-type Filtering in Space and Time with Cellular Neural Networks”. **Circuits Syst. Fundam. Theory Appl. IEEE Trans.** Vol. 45, pp. 121–132.

Shiyong Cui, Dumitru, C.O. and Datcu, M., 2013. “Ratio-Detector-Based Feature Extraction for Very High Resolution SAR Image Patch Indexing”. **IEEE Geosci. Remote Sens. Lett.** Vol. 10, pp. 1175–1179. doi:10.1109/LGRS.2012.2235406

Tekeli, E., Cetin, M. and Erccil, A., 2007. “Shape and Data-driven Texture Segmentation using Local Binary Patterns”. **15th European Signal Processing Conference.** Vol. 15, pp. 1442 – 1446.

Topi, M., Matti, P. and Timo, O., 2000. “Texture Classification by Multi-Predicate Local Binary Pattern Operators, Pattern Recognition”, 2000. **Proceedings. 15th International Conference,** pp. 939–942.

Vatsavai, R.R., 2011. “High-resolution Urban Image Classification using Extended Features, Data Mining Workshops (ICDMW)”, **2011 IEEE 11th International Conference,** pp. 869–876.

Wang, X., Wang, Y., Yang, X. and Zuo, H., 2012. “Texture Classification Based on SIFT Features and Bag-Of-Words in Compressed Domain”, Image and Signal Processing (CISP), **2012 5th International Congress,** pp. 941–945.

Yang, Y. and Newsam, S., 2008. “Comparing SIFT Descriptors and Gabor Texture Features for Classification of Remote Sensed Imagery, Image Processing”, 2008. ICIP 2008. **15th IEEE International Conference,** pp. 1852–1855.

Zhang, L. and Huang, X., 2010. “Object-oriented Subspace Analysis for Airborne Hyperspectral Remote Sensing Imagery”. **Neurocomputing** Vol. 73, pp. 927–936. doi:10.1016/j.neucom.2009.09.011

Zhang, Y. and Zhang, J., Huang, G. and Zhao, Z., 2011. “Object-Oriented Classification of Polarimetric SAR Imagery Based on Texture Features”, Image and Data Fusion (ISIDF), **2011 International Symposium,** pp. 1–4.

Zhan, X., Xingbo, S. and Yuerong, L., 2009. “Comparison of Two Gabor Texture Descriptor for Texture Classification”. **IEEE,** pp. 52–56. doi:10.1109/ICIE.2009.20

APPENDIX A
Classified Image Results

Table A.1 Image Results with Different Sample Size on Region 1

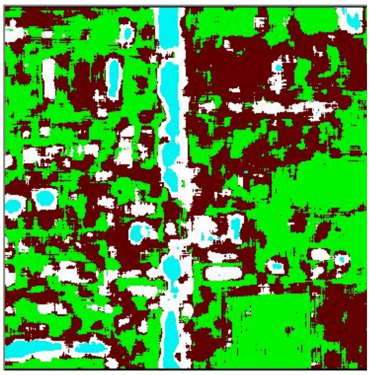
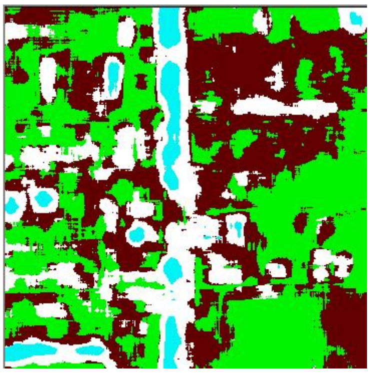
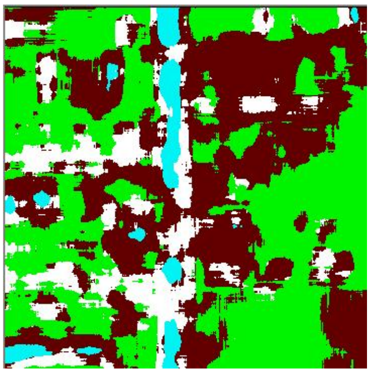
Sample Size: 30 x 30	Sample Size: 40 x 40	Sample Size: 50 x 50
		

Table A.2 Image Results with Different Sample Size on Region 2

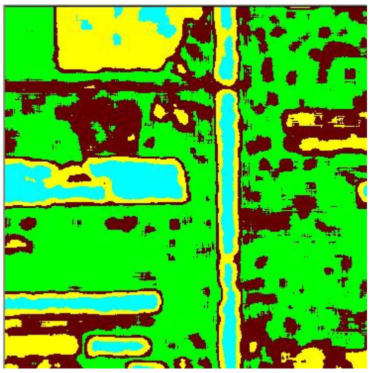
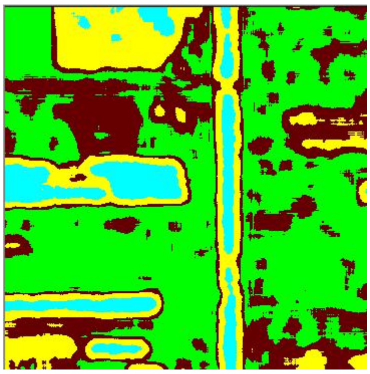
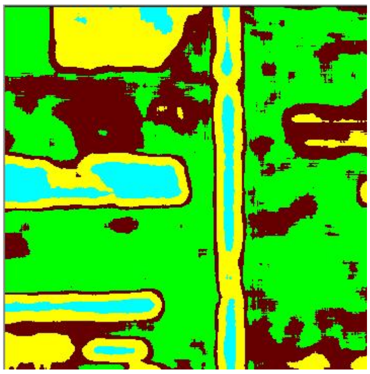
Sample Size: 30 x 30	Sample Size: 40 x 40	Sample Size: 50 x 50
		

Table A.3 Image Results with Different Ringsize on Region 1

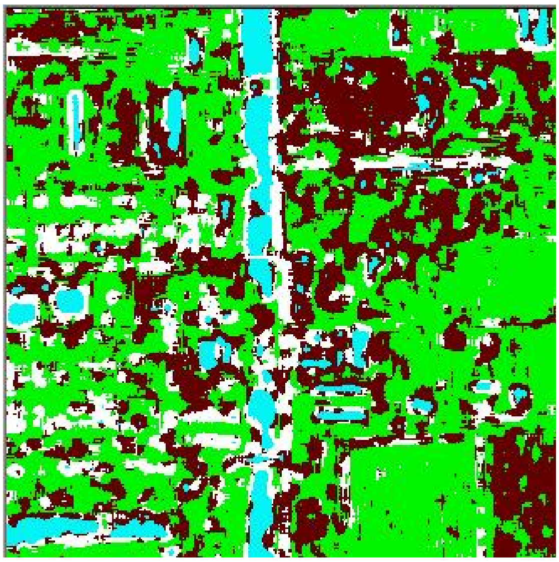
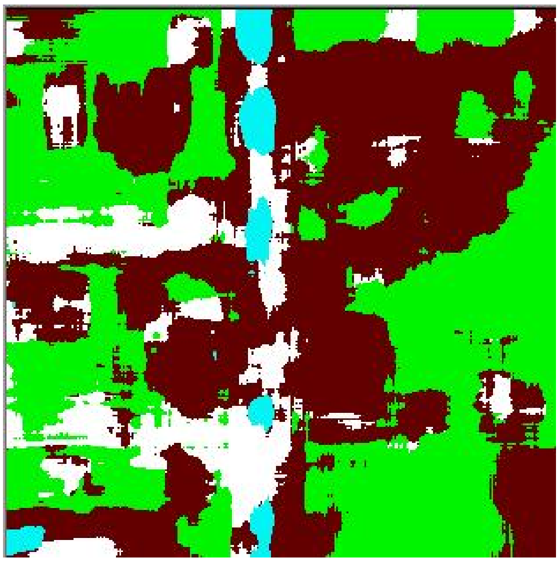
Ringsize 10 x 10 (Region 1)	Ringsize 35 x 35 (Region 1)
	

Table A.4 Image Results with Different Ringsize on Region 2

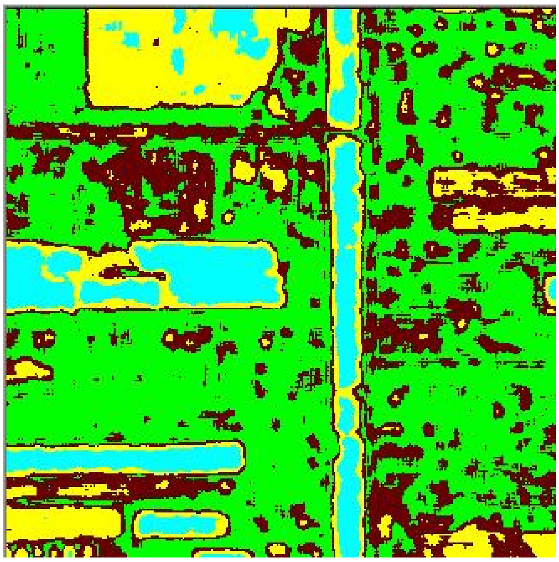
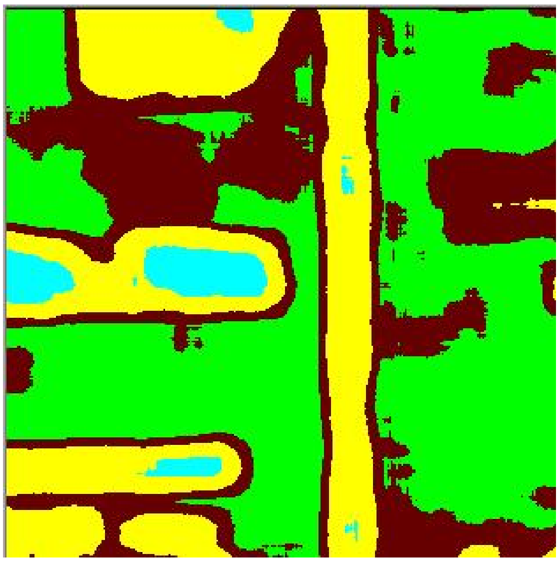
Ringsize 10 x 10 (Region2)	Ringsize 30 x 30 (Region 2)
	

Table A.5 Results of LBP (P: 8 and R: 1) on Each Region

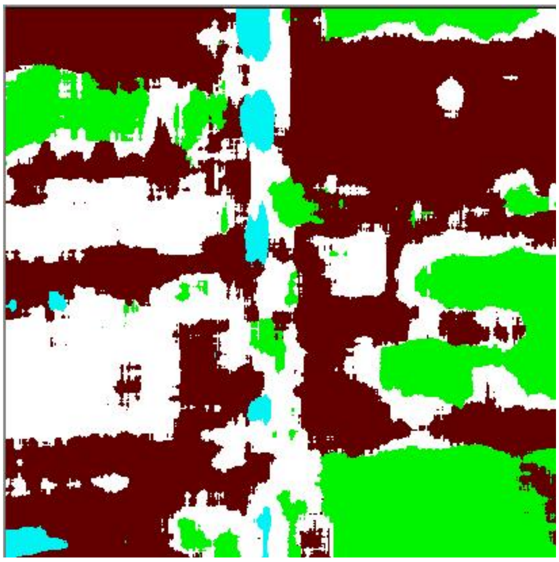
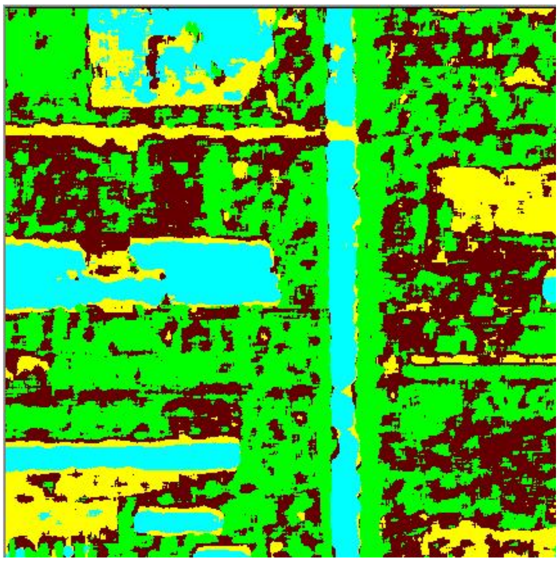
Region 1	Region 2
	

Table A.6 Results of LBP (P: 16 and R: 2) on Each Region

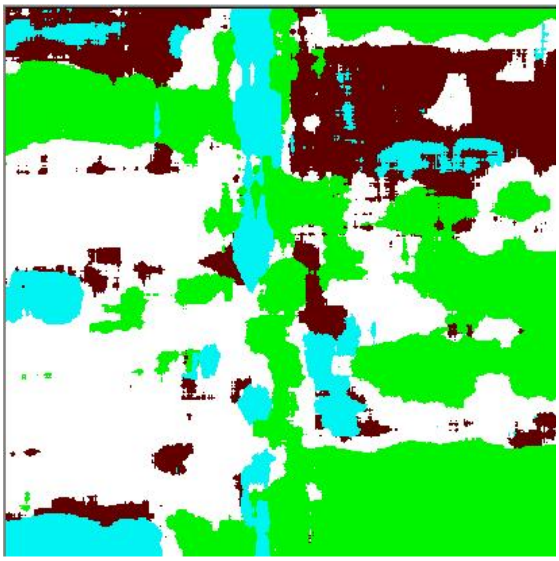
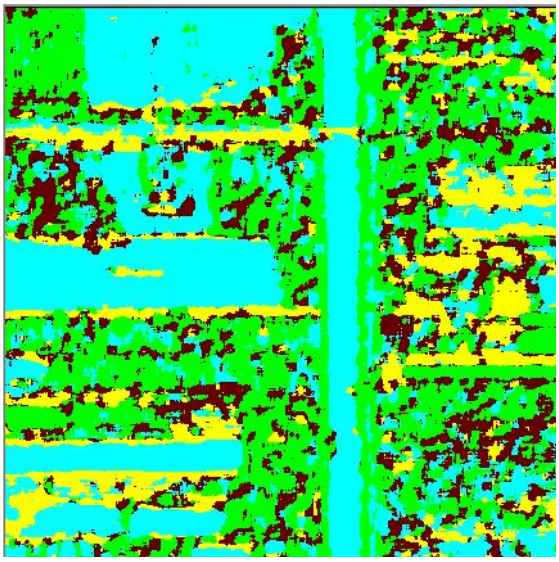
Region 1	Region 2
	

Table A.7 Results of LBP (P: 24 and R: 3) on Each Region

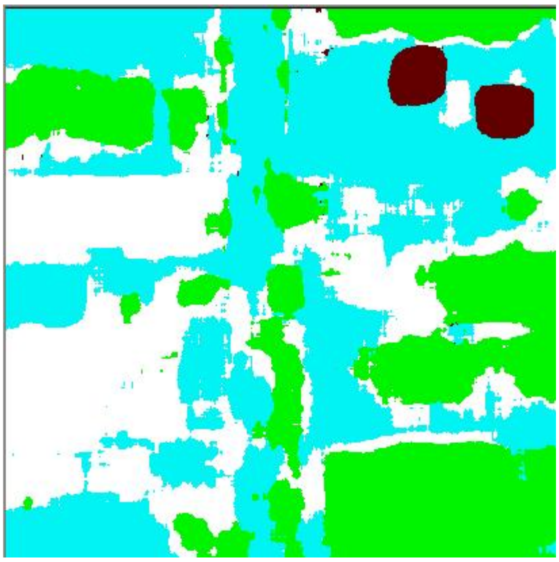
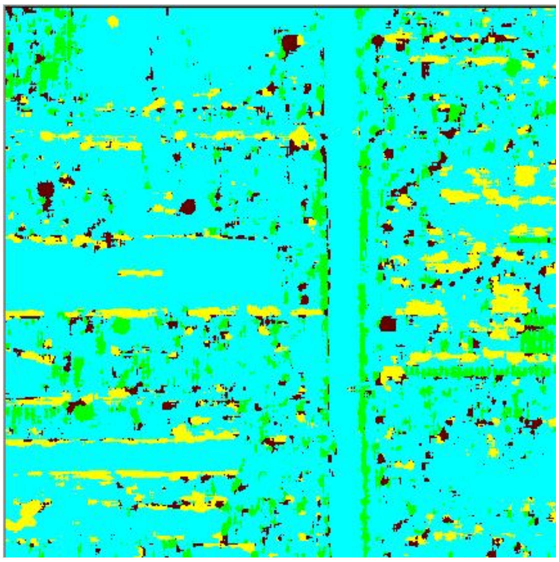
Region 1	Region 2
	

Table A.8 Results of LBP (Multiscale) on Each Region

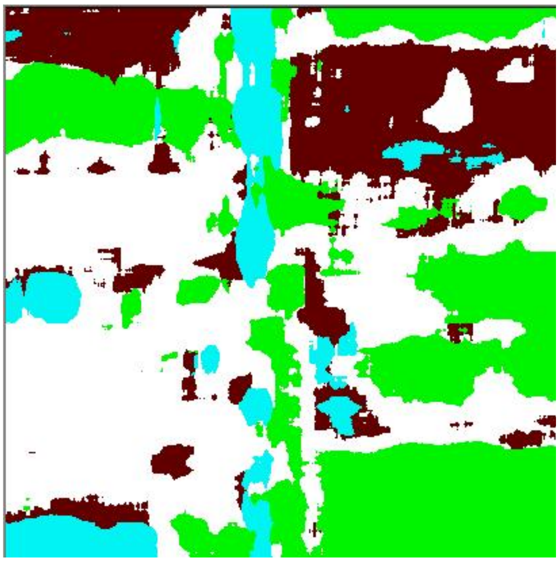
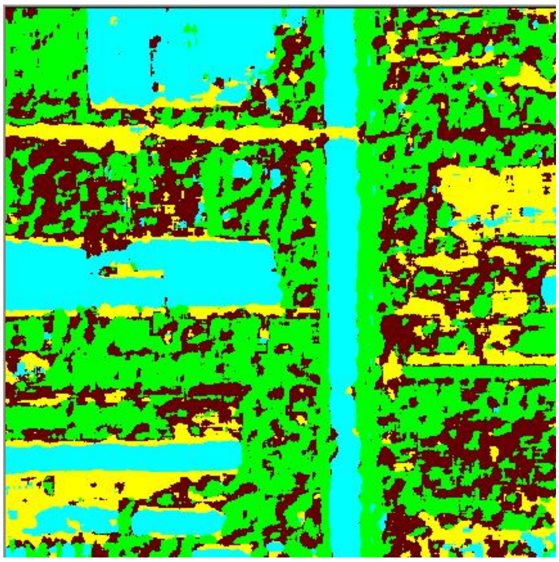
Region 1	Region 2
	

Table A.9 Results of LBPRIU (P: 8 and R: 1) on Each Region

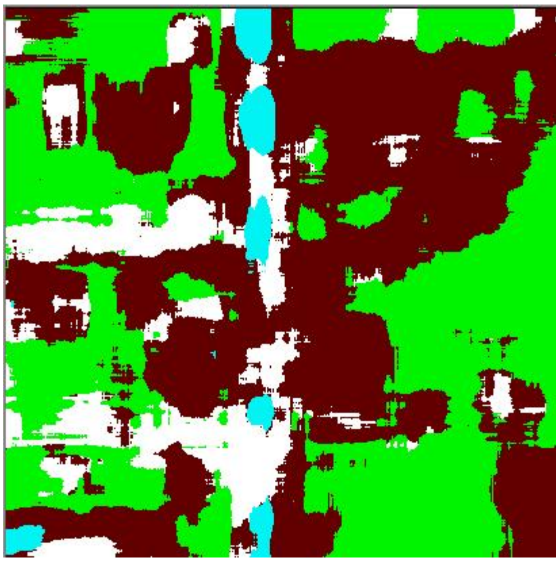
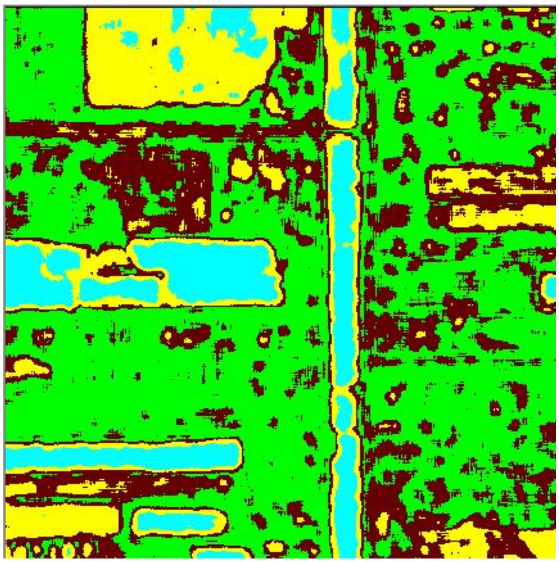
Region 1	Region 2
	

Table A.10 Results of LBPRIU (P: 16 and R: 2) on Each Region

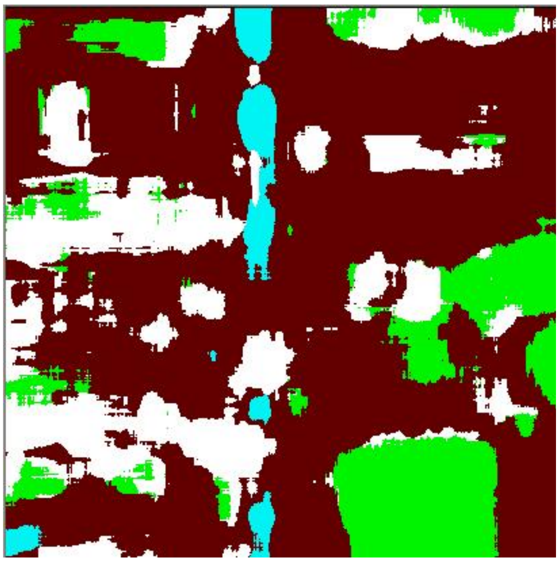
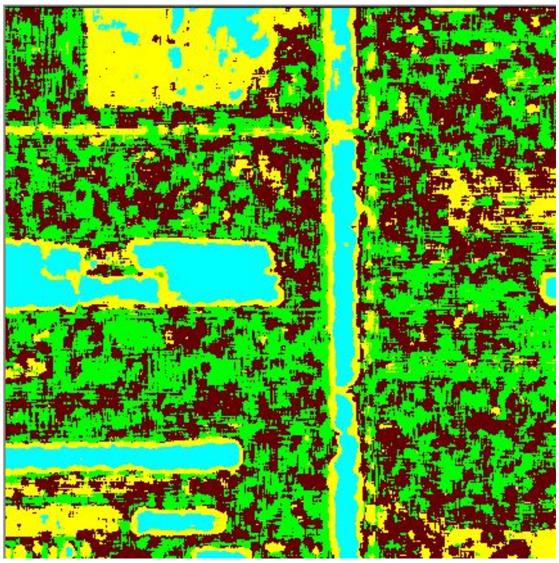
Region 1	Region 2
	

Table A.11 Results of LBPRIU (P: 24 and R: 3) on Each Region

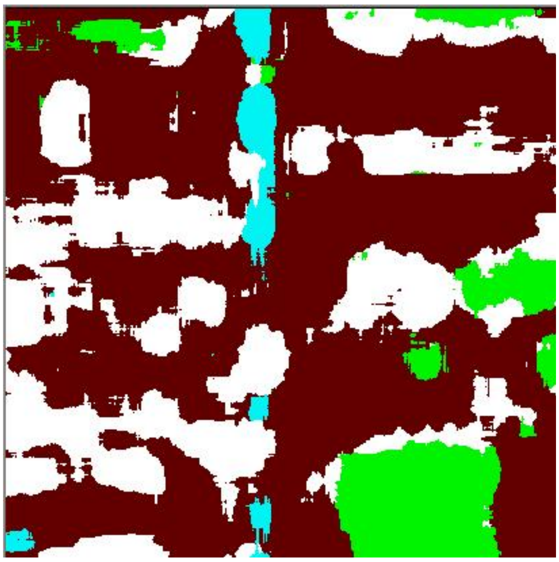
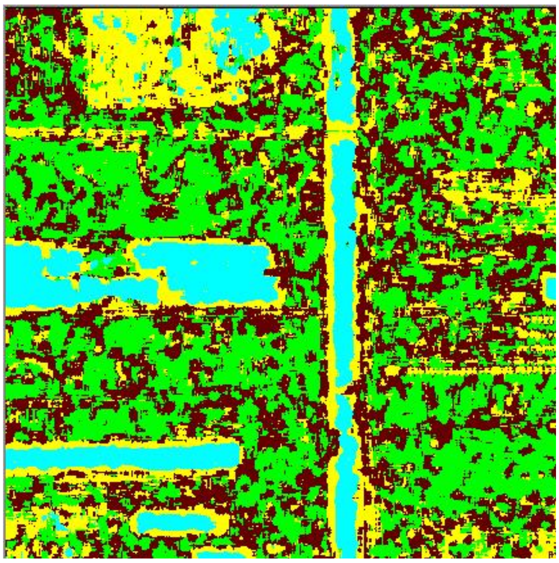
Region 1	Region 2
	

Table A.12 Results of LBPRIU (Multiscale) on Each Region

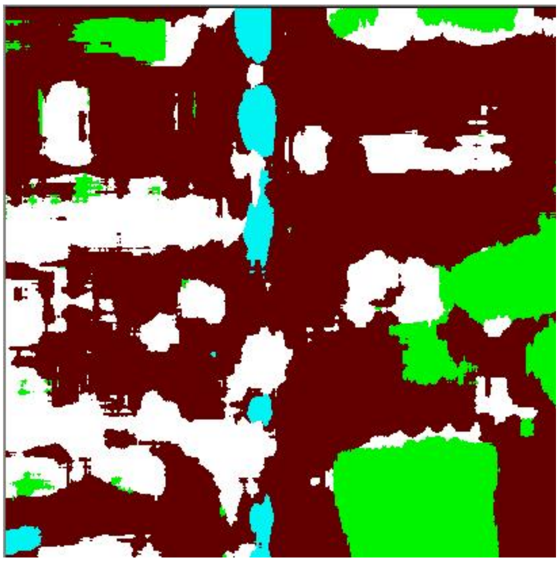
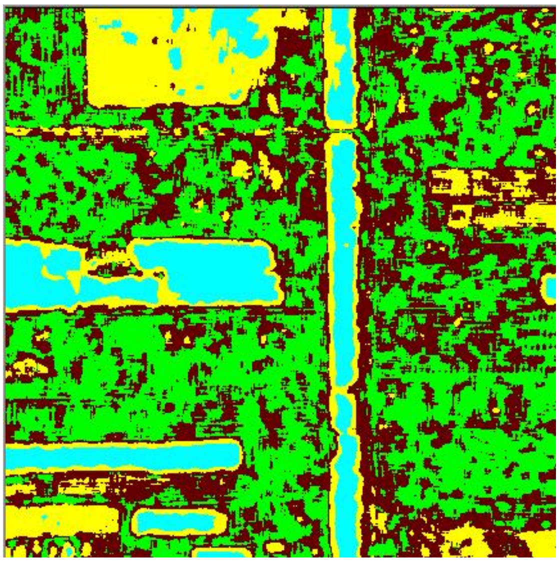
Region 1	Region 2
	

Table A.13 Results of WLD (P: 8 and R: 1) on Each Region

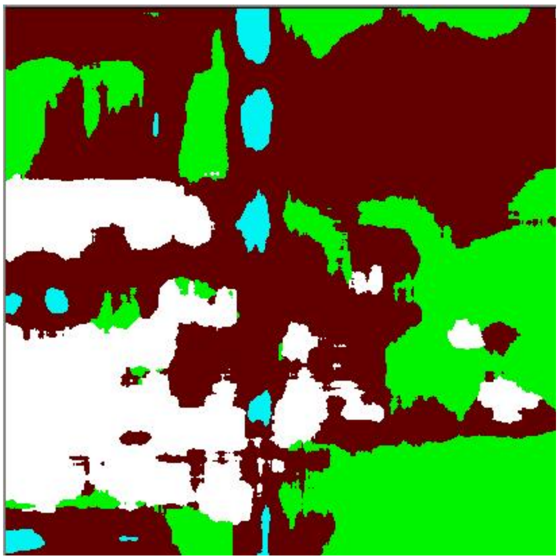
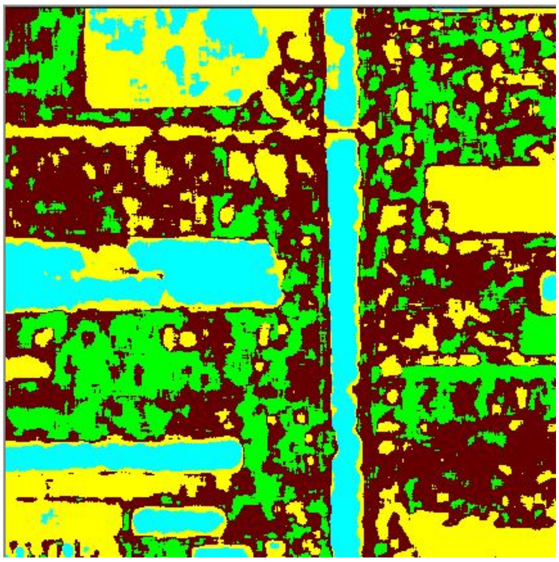
Region 1	Region 2
	

Table A.14 Results of WLD (P: 16 and R: 2) on Each Region

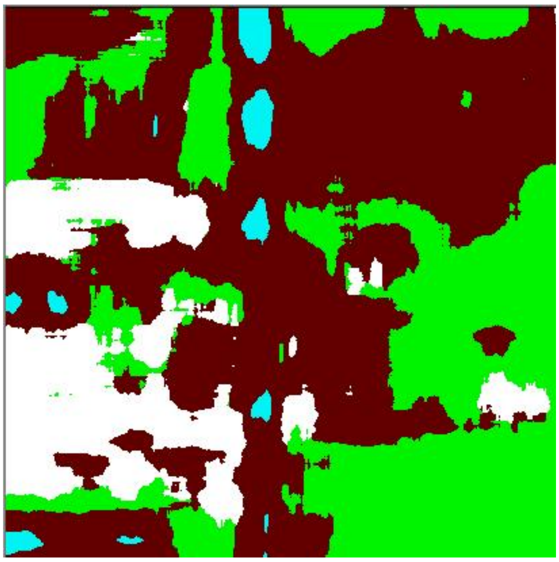
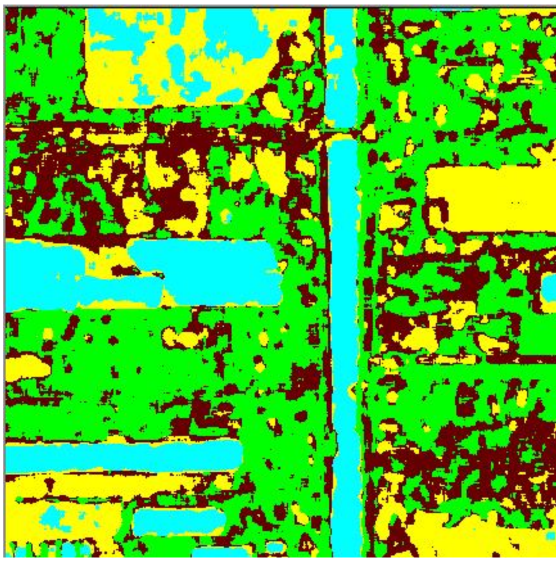
Region 1	Region 2
	

Table A.15 Results of WLD (P: 24 and R: 3) on Each Region

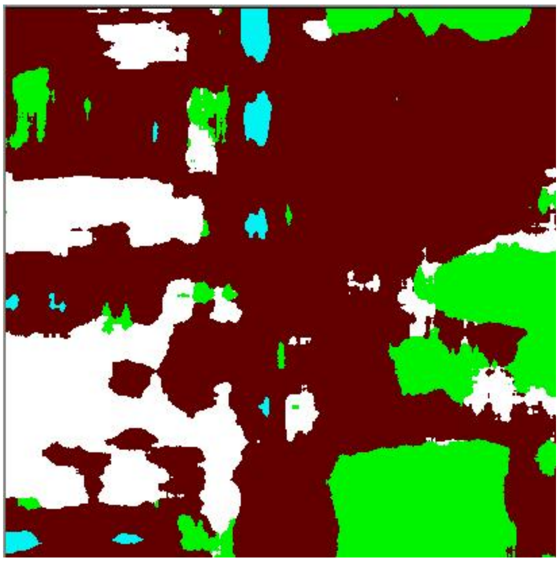
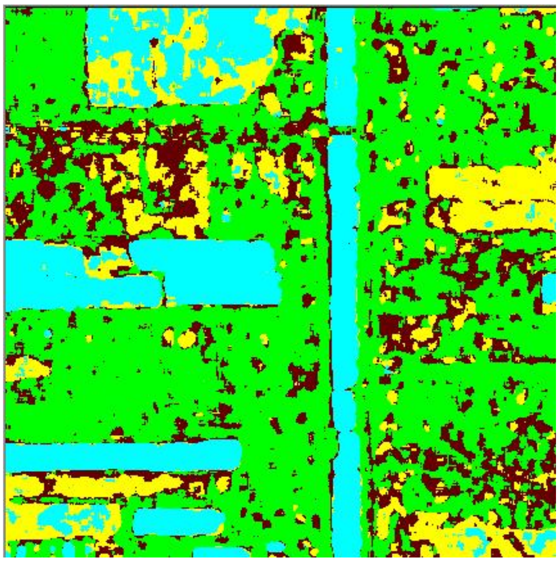
Region 1	Region 2
	

Table A.16 Results of WLD (Multiscale) on Each Region

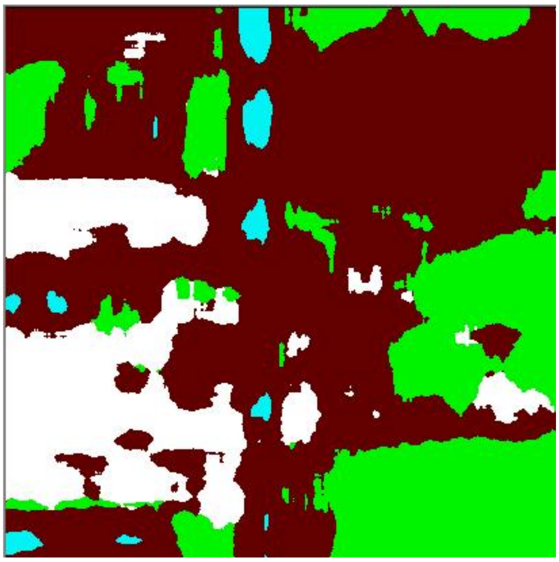
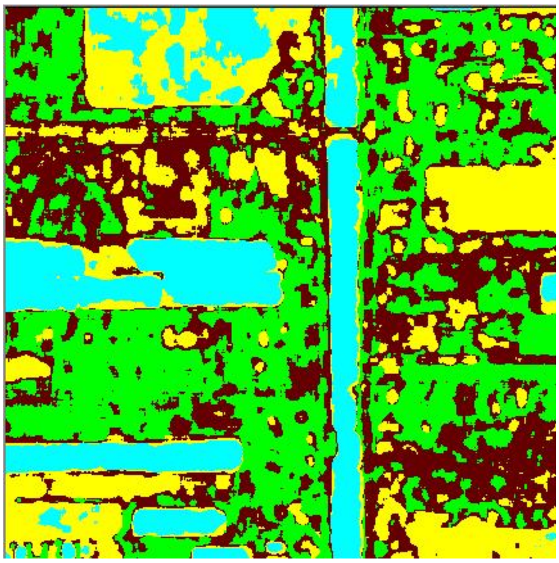
Region 1	Region 2
	

Table A.17 Results of VAR (P: 8 and R: 1) on Each Region

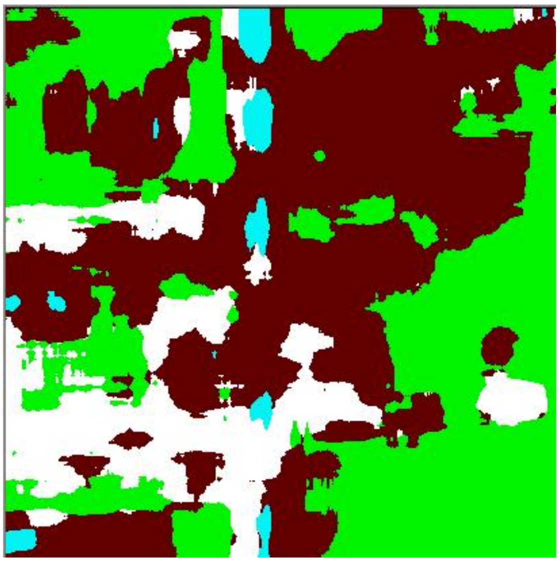
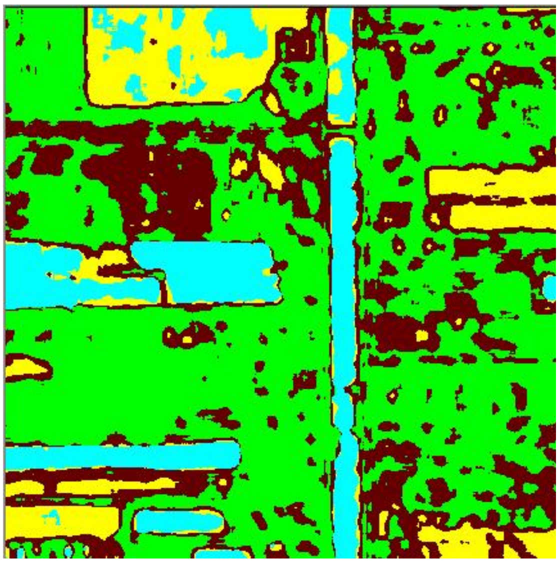
Region 1	Region 2
	

Table A.18 Results of VAR (P: 16 and R: 2) on Each Region

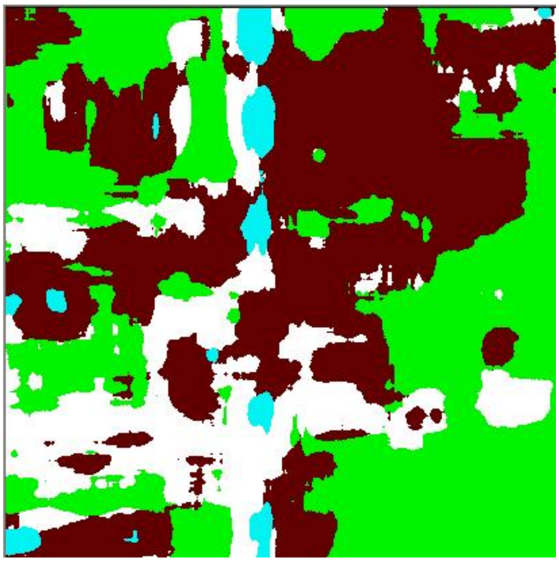
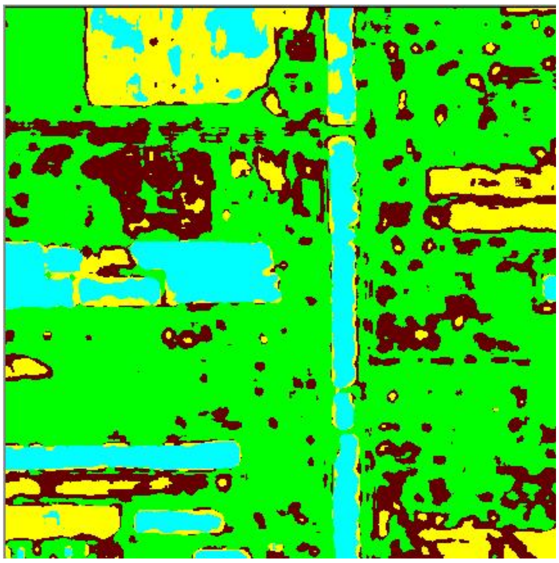
Region 1	Region 2
	

Table A.19 Results of VAR (P: 24 and R: 3) on Each Region

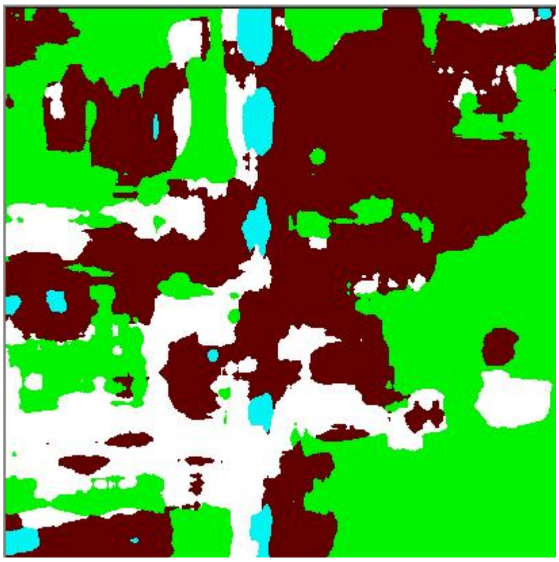
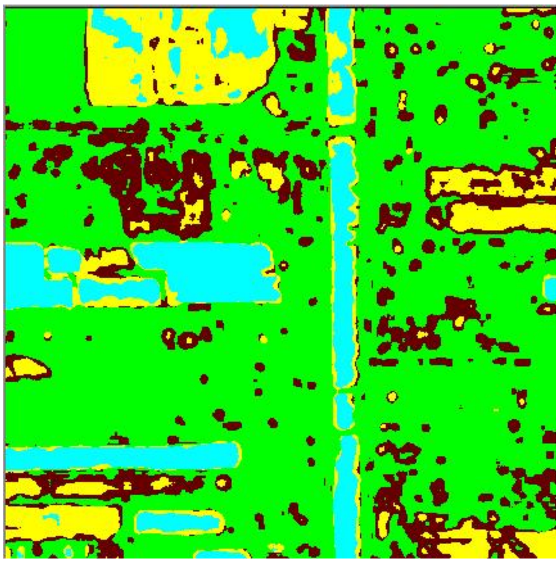
Region 1	Region 2
	

Table A.20 Results of VAR (Multiscale) on Each Region

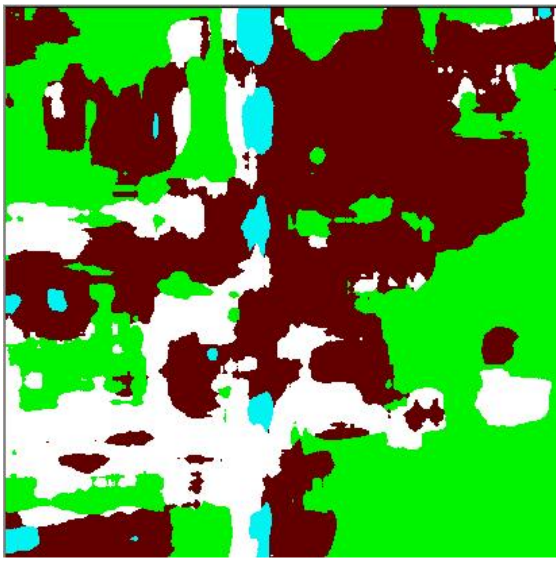
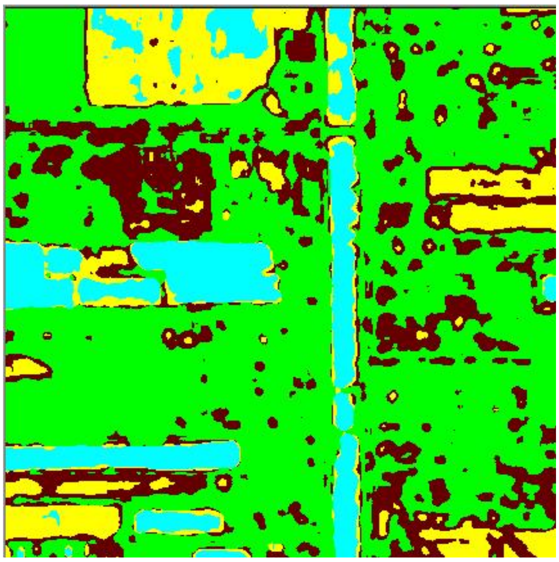
Region 1	Region 2
	

Table A.21 Results of LBPRIUVAR (P: 8 and R: 1) on Each Region

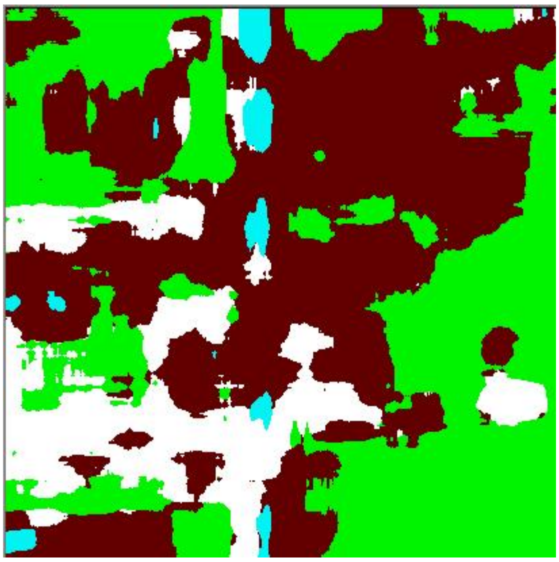
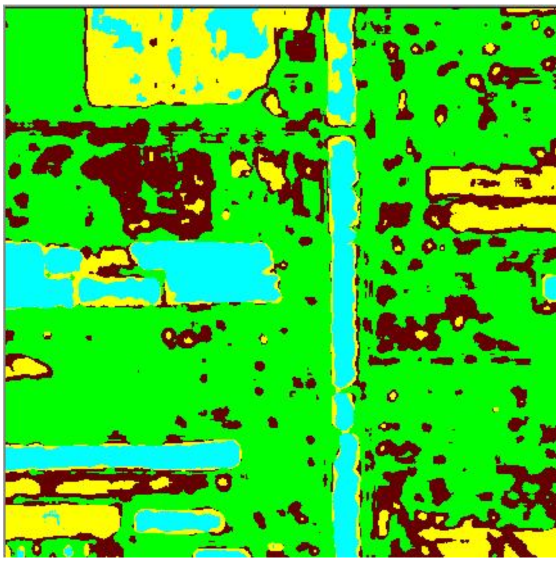
Region 1	Region 2
	

Table A.22 Results of LBPRIUVAR (P: 16 and R: 2) on Each Region

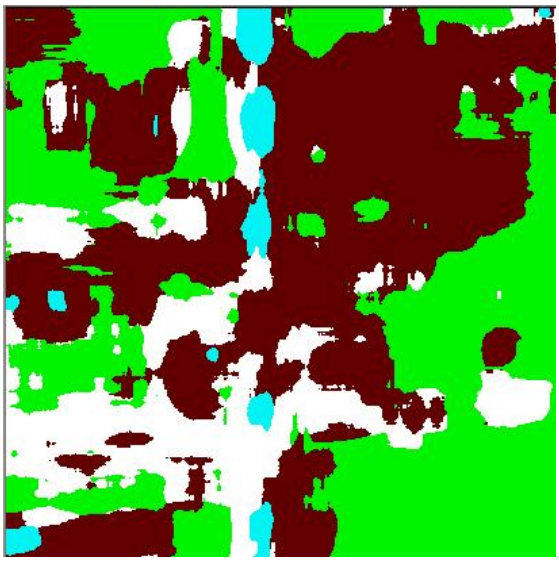
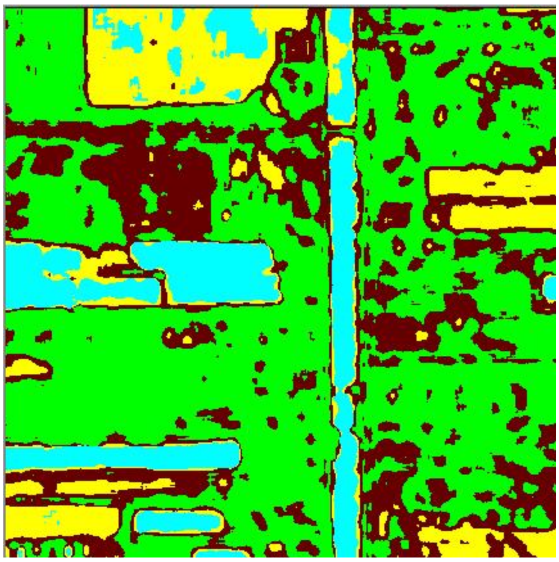
Region 1	Region 2
	

Table A.23 Results of LBPRIUVAR (P: 24 and R: 3) on Each Region

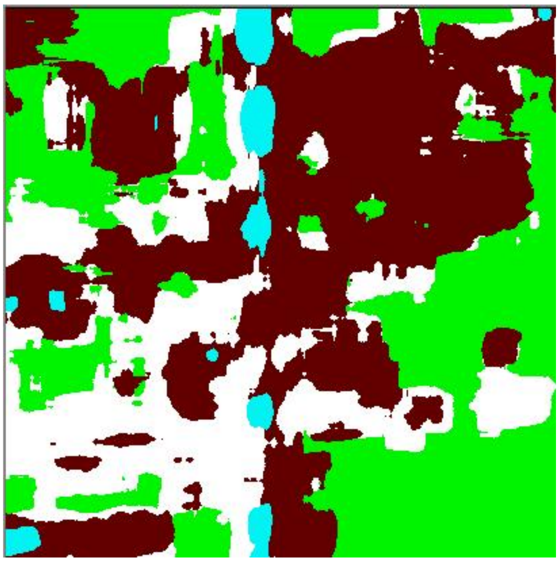
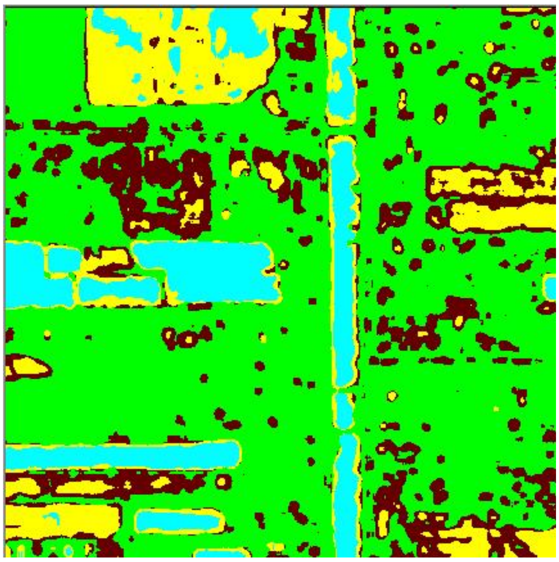
Region 1	Region 2
	

Table A.24 Results of LBPRIUVAR (Multiscale) on Each Region

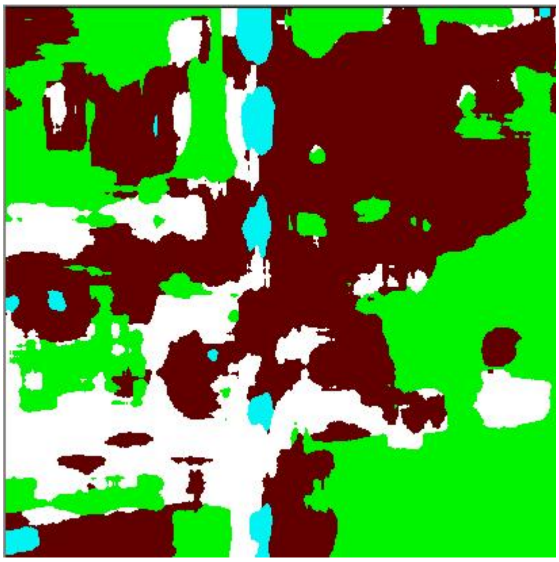
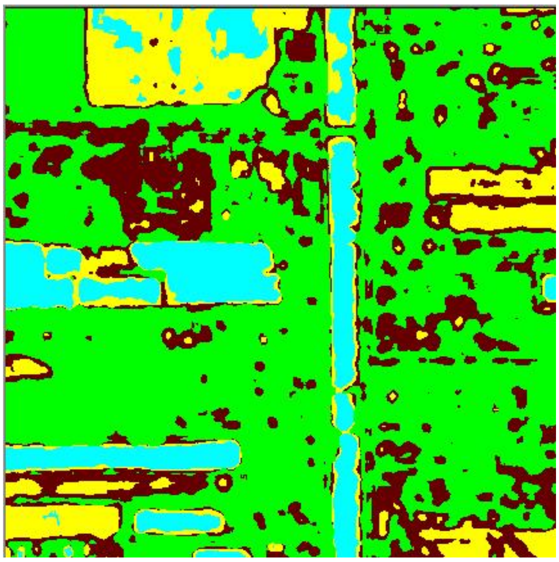
Region 1	Region 2
	

Table A.25 Results of WLDVAR (P: 8 and R: 1) on Each Region

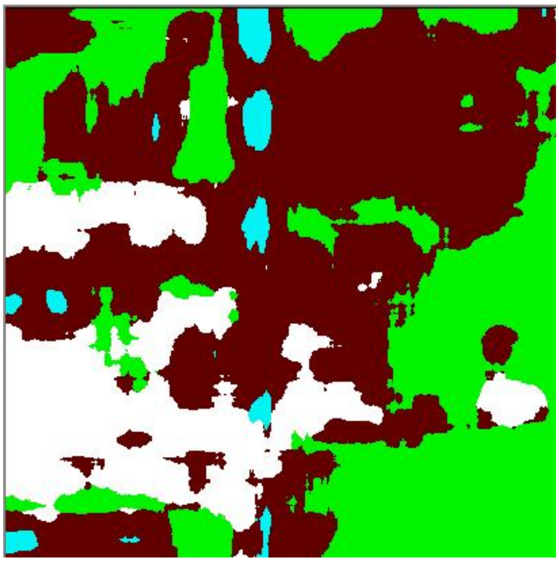
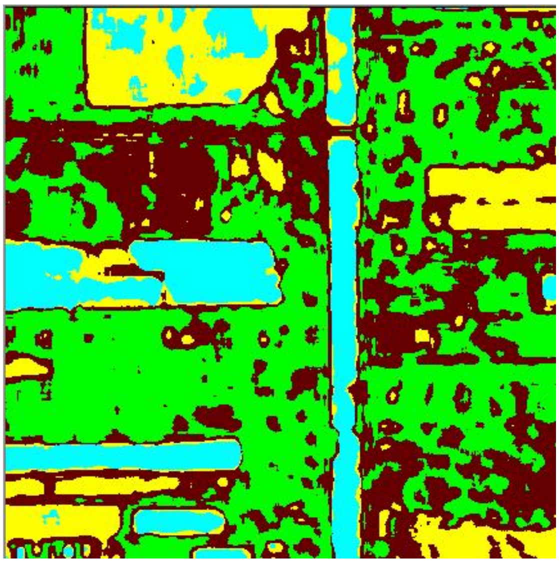
Region 1	Region 2
	

Table A.26 Results of WLDVAR (P: 16 and R: 2) on Each Region

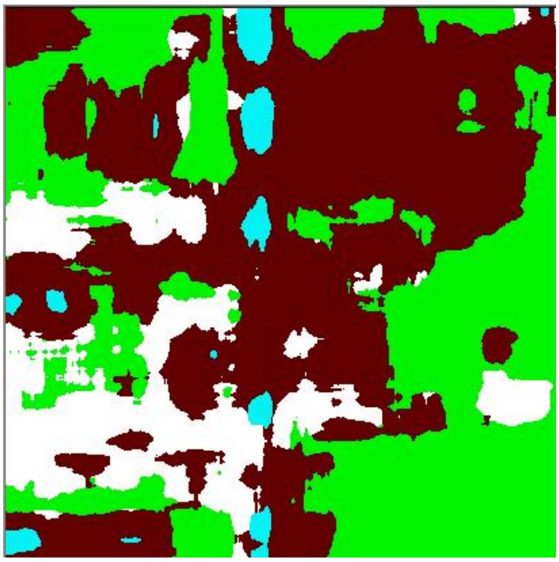
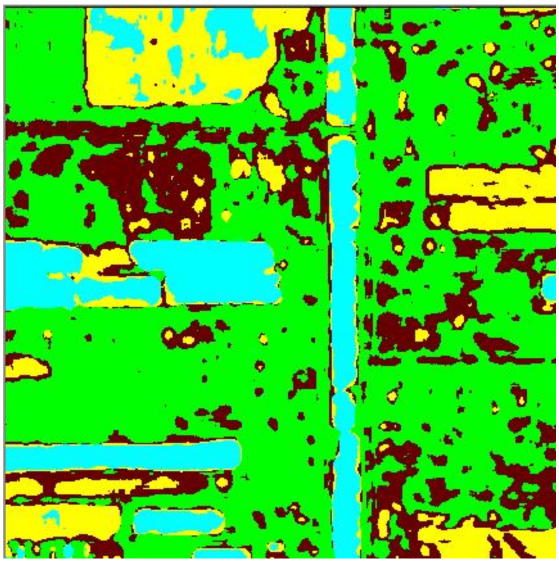
Region 1	Region 2
	

Table A.27 Results of WLDVAR (P: 24 and R: 3) on Each Region

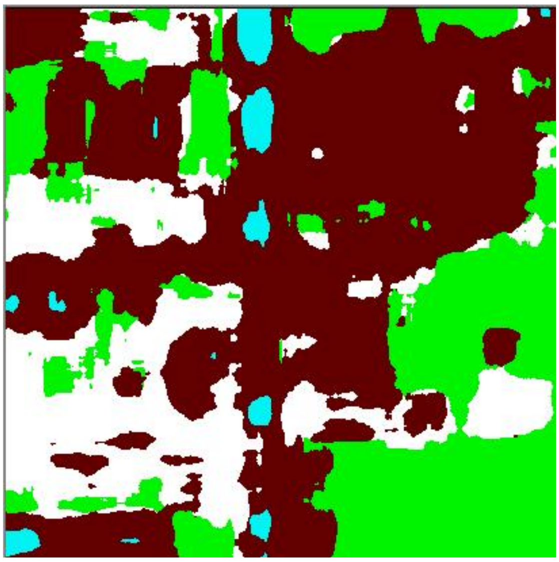
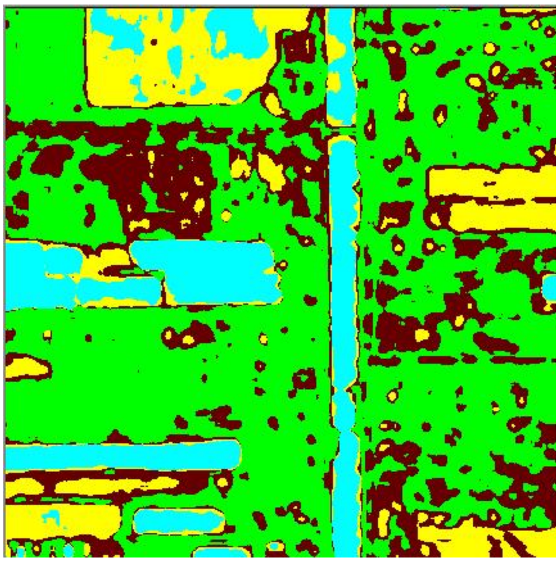
Region 1	Region 2
	

Table A.28 Results of WLDVAR (Multiscale) on Each Region

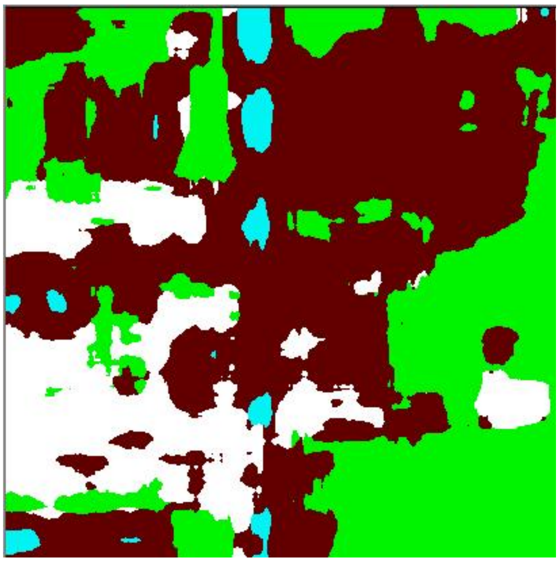
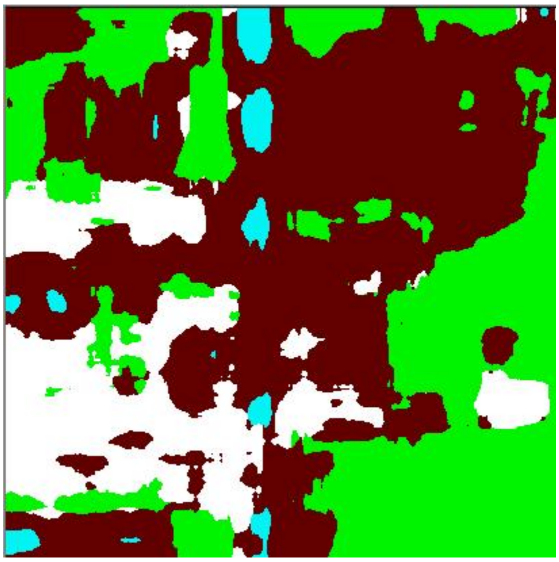
Region 1	Region 2
	

Table A.29 Best Results of LBPRIU on Each Region

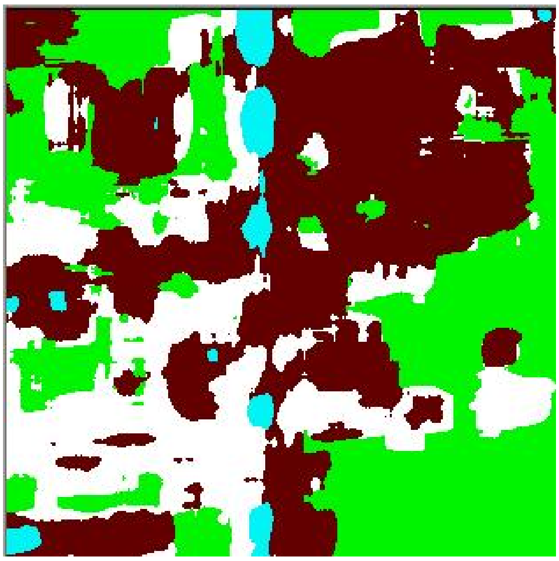
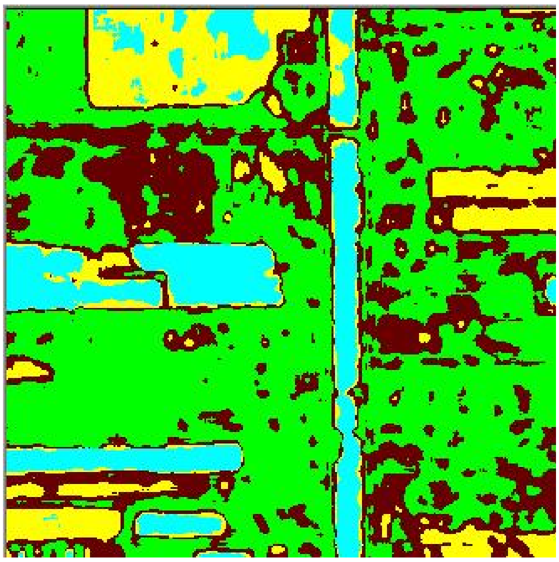
Region 1	Region 2
	

Table A.30 Best Results of WLD on Each Region

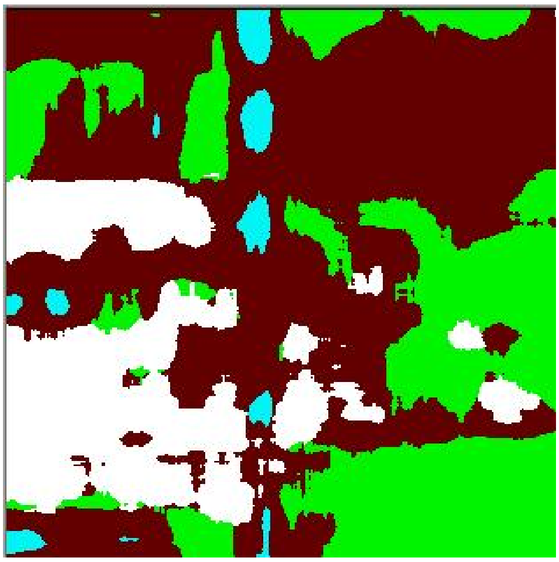
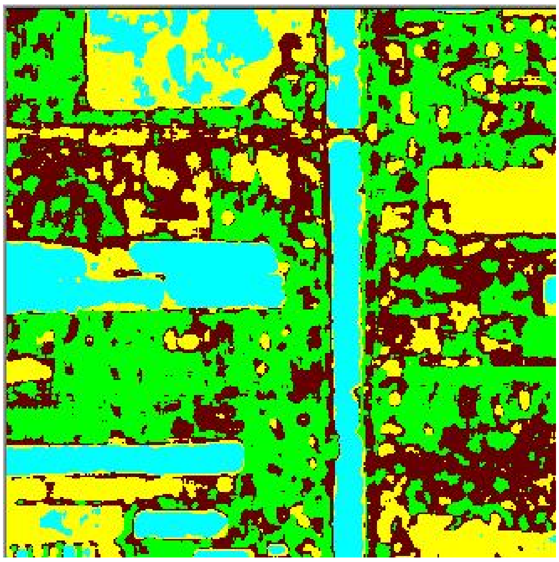
Region 1	Region 2
	

Table A.31 Best Results of VAR on Each Region

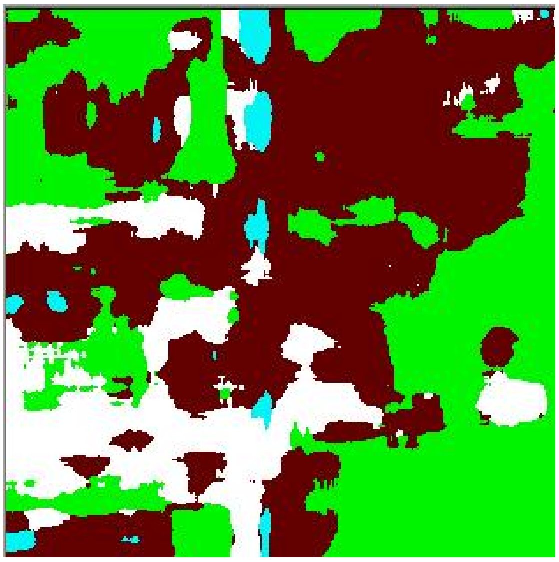
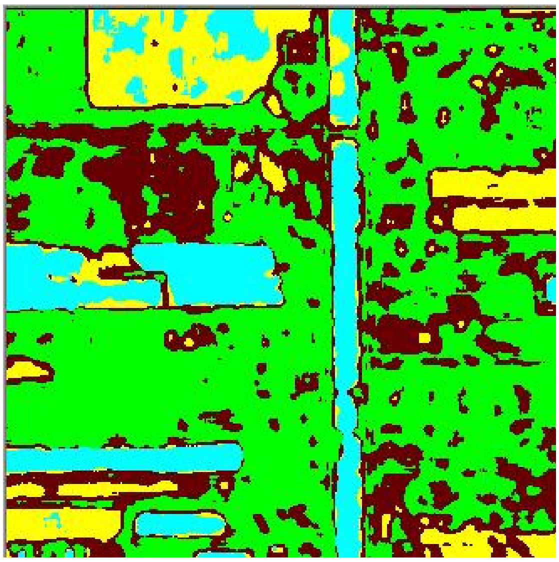
Region 1	Region 2
	

Table A.32 Best Results of LBPRIUVAR on Each Region

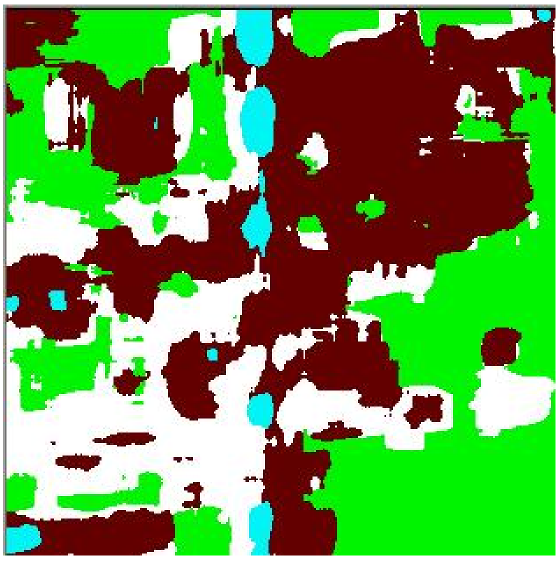
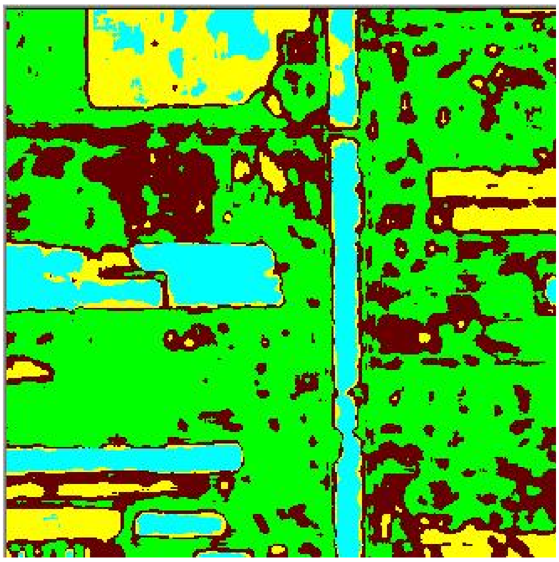
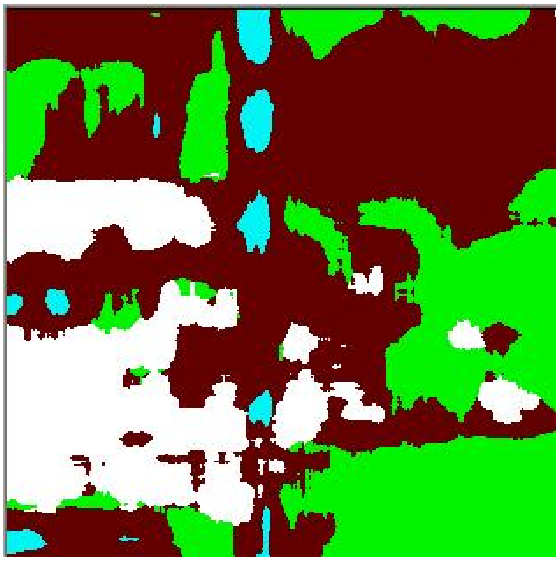
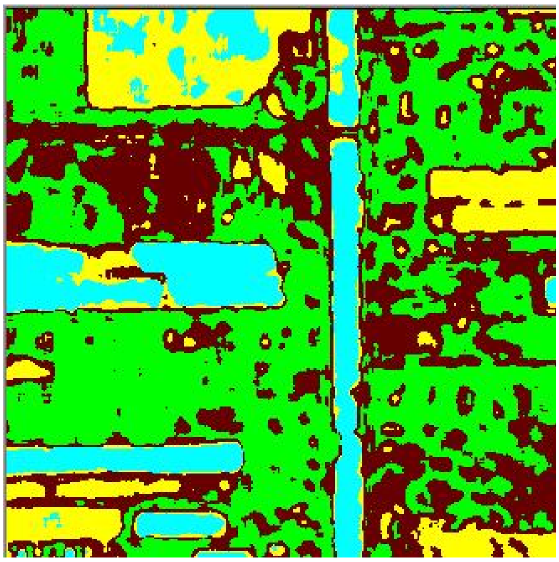
Region 1	Region 2
	

Table A.33 Best Results of WLDVAR on Each Region

Region 1	Region 2
	

APPENDIX B

List of Confusion Matrix

Table B.1 Confusion Size selection (Region 1)

Sample Size: 30 x 30						
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	25305	45500	0	37954	42.84	23.27
Orchard	6160	106527	1308	39296	63.27	69.49
Water	19330	440	33277	9286	95.53	53.39
Barren	8280	15899	249	73951	46.08	75.17
Overall Accuracy						**56.55**
Kappa Value						**0.40**

Sample Size: 40 x 40						
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	38765	48728	0	18079	47.53	36.72
Orchard	8549	101720	881	31307	61.90	71.40
Water	24150	308	29753	5063	97.12	50.20
Barren	10090	13568	0	72669	57.17	75.44
Overall Accuracy						**60.18**
Kappa Value						**0.44**

Sample Size: 50 x 50						
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	39886	48427	0	13866	64.30	39.04
Orchard	6371	96109	643	27948	59.65	73.33
Water	10597	279	28047	16842	97.76	50.29
Barren	5180	16312	0	72085	55.14	77.03
Overall Accuracy						**61.72**
Kappa Value						**0.47**

Table B.2 Confusion matrix on sample Size selection (Region 2)

Sample Size: 30 x 30						
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	62849	0	1	3579	54.60	94.61
Orchard	517	78643	53508	7153	92.02	56.25
Water	731	6818	70373	34140	53.06	62.80
Barren	51006	0	8741	72959	61.92	54.98
Overall Accuracy						**63.15**
Kappa Value						**0.51**

Sample Size: 40 x 40						
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	60818	0	28	3377	54.00	94.70
Orchard	413	71950	60221	5032	91.28	52.28
Water	1187	6876	66284	34569	49.65	60.86
Barren	50199	0	6974	72112	62.66	55.78
Overall Accuracy						**61.62**
Kappa Value						**0.49**

Sample Size: 50 x 50						
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	59076	0	230	4004	55.61	93.31
Orchard	54	61536	68417	5246	92.78	45.50
Water	1621	4790	65603	35612	46.17	60.95
Barren	45487	0	7855	71998	61.61	57.44
Overall Accuracy						**59.84**
Kappa Value						**0.47**

Table B.3 Confusion matrix on ringsize selection (Region 1)

Ringsize: 10 x 10						
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	36033	53296	477	17689	54.87	33.52
Orchard	8368	112841	2445	30594	57.80	73.16
Water	12284	1565	34317	10726	89.20	58.27
Barren	8982	27508	1232	57537	49.37	60.40
Overall Accuracy						**57.88**
Kappa Value						**0.40**

Ringsize: 35 x 35						
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	40650	45788	0	12485	66.52	41.09
Orchard	4895	87360	0	23625	60.38	75.39
Water	12918	192	17564	22917	100.00	32.77
Barren	2642	11352	0	77603	56.80	84.72
Overall Accuracy						**62.00**
Kappa Value						**0.47**

Table B.4 Confusion matrix on ringsize selection (Region 2)

Ringsize: 10 x 10						
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	63866	0	129	4364	52.54	93.43
Orchard	1441	88042	42563	9769	91.33	62.08
Water	2461	8357	74927	29013	57.60	65.29
Barren	53797	0	12459	69630	61.74	51.24
Overall Accuracy						**64.33**
Kappa Value						**0.53**

Ringsize: 35 x 35						
True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	52837	0	16	4834	55.53	91.59
Orchard	52	28036	99227	3653	93.03	21.41
Water	764	2099	61488	35308	37.16	61.70
Barren	41502	0	4723	74090	62.85	61.58
Overall Accuracy						**52.97**
Kappa Value						**0.38**

Table B.5 Confusion Matrix of LBP on Region 1 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	75009	185	0	23729	68.60	75.83
Orchard	13044	90969	0	11867	99.64	78.50
Water	14904	98	15049	23540	100.00	28.08
Barren	6381	50	0	85166	59.02	92.98
Overall Accuracy						**73.94**
Kappa Value						**0.64**

Table B.6 Confusion Matrix of LBP on Region 1 (P: 16 and R: 2)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	94237	2247	337	2102	75.74	95.26
Orchard	8157	104649	1896	1178	88.16	90.31
Water	6668	6985	37386	2552	80.64	69.76
Barren	15366	4827	6745	64659	91.73	70.59
Overall Accuracy						**83.59**
Kappa Value						**0.78**

Table B.7 Confusion Matrix of LBP on Region 1 (P: 24 and R: 3)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	90801	732	7390	0	83.86	91.79
Orchard	6538	102963	6379	0	94.29	88.85
Water	2733	3864	46994	0	36.27	87.69
Barren	8205	1640	68795	12957	100.00	14.15
Overall Accuracy						**70.48**
Kappa Value						**0.61**

Table B.8 Confusion Matrix of LBP on Region 1 (Multiscale)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	95530	749	232	2412	73.00	96.57
Orchard	9661	103353	1618	1248	93.05	89.19
Water	8458	4575	36188	4370	88.34	67.53
Barren	17208	2393	2925	69071	89.59	75.41
Overall Accuracy						**84.49**
Kappa Value						**0.79**

Table B.9 Confusion Matrix of LBP on Region 2 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	52765	0	1337	14257	51.95	77.19
Water	4016	112004	19656	6139	78.14	78.98
Crop	6014	31330	60280	17134	63.38	52.53
Barren	38777	5	13831	83273	68.93	61.28
Overall Accuracy						**66.91**
Kappa Value						**0.55**

Table B.10 Confusion Matrix of LBP on Region 2 (P: 16 and R: 2)

Texture class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	44001	6216	6150	11992	56.73	64.37
Water	1583	134458	4375	1399	50.16	94.81
Crop	3412	80543	25777	5026	45.33	22.46
Barren	28562	46828	20567	39929	68.43	29.38
Overall Accuracy						**52.99**
Kappa Value						**0.35**

Table B.11 Confusion Matrix of LBP on Region 2 (P: 24 and R: 3)

Texture class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	14346	49460	3908	3803	71.92	20.06
Water	685	138118	3105	748	34.19	96.82
Crop	693	102854	10586	1466	37.10	9.16
Barren	4223	113583	10933	7988	57.04	5.84
Overall Accuracy						**36.66**
Kappa Value						**0.10**

Table B.12 Confusion Matrix of LBP on Region 2 (Multiscale)

Texture class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	48663	0	4017	15679	50.44	71.19
Water	2575	129153	7172	2915	65.85	91.07
Crop	4775	62273	38785	8925	54.64	33.80
Barren	40468	4706	21005	69707	71.70	51.30
Overall Accuracy						**62.13**
Kappa Value						**0.49**

Table B.13 Confusion Matrix of LBPRIU on Region 1 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	40650	45788	0	12485	66.52	41.09
Orchard	4895	87360	0	23625	60.38	75.39
Water	12918	192	17564	22917	100.00	32.77
Barren	2642	11352	0	77603	56.80	84.72
Overall Accuracy						**62.00**
Kappa Value						**0.47**

Table B.14 Confusion Matrix of LBPRIU on Region 1 (P: 16 and R: 2)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	58619	12096	0	28208	63.17	59.26
Orchard	12584	75285	0	28011	85.24	64.97
Water	12083	0	18969	22539	100.00	35.40
Barren	9510	938	0	81149	50.75	88.59
Overall Accuracy						**65.01**
Kappa Value						**0.52**

Table B.15 Confusion Matrix of LBPRIU on Region 1 (P: 24 and R: 3)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	63507	83	0	35333	50.86	64.20
Orchard	24377	63288	5	28210	99.26	54.62
Water	18304	285	16758	18244	99.97	31.27
Barren	18681	101	0	72815	47.10	79.49
Overall Accuracy						**65.01**
Kappa Value						**0.52**

Table B.16 Confusion Matrix of LBPRIU on Region 1 (Multiscale)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	64729	2637	0	31557	61.91	65.43
Orchard	11936	77413	0	26531	96.68	66.80
Water	15776	0	17859	19956	100.00	33.32
Barren	12107	18	0	79472	50.45	86.76
Overall Accuracy						**66.52**
Kappa Value						**0.54**

Table B.17 Confusion Matrix of LBPRIU on Region 2 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	63866	0	129	4364	52.54	93.43
Water	1441	88042	42563	9769	91.33	62.08
Crop	2461	8357	74927	29013	57.60	65.29
Barren	53797	0	12459	69630	61.74	51.24
Overall Accuracy						**64.33**
Kappa Value						**0.53**

Table B.18 Confusion Matrix of LBPRIU on Region 2 (P: 16 and R: 2)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	39887	0	936	27536	41.82	58.35
Water	1491	92439	41884	6001	90.48	65.18
Crop	5779	9730	72636	26613	56.86	63.29
Barren	48226	0	12282	75378	55.62	55.47
Overall Accuracy						**60.84**
Kappa Value						**0.47**

Table B.19 Confusion Matrix of LBPRIU on Region 2 (P: 24 and R: 3)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	34737	0	4706	28916	26.84	50.82
Water	3765	92579	37354	8117	89.26	65.28
Crop	18885	11135	58637	26101	50.81	51.10
Barren	72048	2	14718	49118	43.76	36.15
Overall Accuracy						**51.01**
Kappa Value						**0.35**

Table B.20 Confusion Matrix of LBPRIU on Region 2 (Multiscale)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	48740	0	175	19444	42.15	71.30
Water	1059	90960	41887	7909	91.07	64.14
Crop	4114	8921	72786	28937	58.29	63.43
Barren	61727	0	10024	64135	53.26	47.20
Overall Accuracy						**60.03**
Kappa Value						**0.47**

Table B.21 Confusion Matrix of WLD on Region 1 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	91180	548	0	31557	98.06	92.17
Orchard	113	102985	0	26531	98.39	88.87
Water	1688	0	14729	19956	100.00	27.48
Barren	0	1135	0	79472	61.28	98.76
Overall Accuracy						**83.16**
Kappa Value						**0.77**

Table B.22 Confusion Matrix of WLD on Region 1 (P: 16 and R: 2)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	82535	4468	0	11920	98.70	83.43
Orchard	383	104894	0	10603	94.44	90.52
Water	708	30	12218	40635	100.00	22.80
Barren	0	1677	0	89920	58.74	98.17
Overall Accuracy						**80.44**
Kappa Value						**0.73**

Table B.23 Confusion Matrix of WLD on Region 1 (P: 24 and R: 3)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	82672	527	0	15724	96.29	83.57
Orchard	2926	86074	0	26880	99.27	74.28
Water	224	0	9197	44170	99.93	17.16
Barren	31	105	6	91455	51.31	99.84
Overall Accuracy						**74.83**
Kappa Value						**0.65**

Table B.24 Confusion Matrix of WLD on Region 1 (Multiscale)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	86963	455	0	11505	99.18	87.91
Orchard	210	103175	0	12495	99.08	89.04
Water	509	0	11670	41412	100.00	21.78
Barren	0	499	0	91098	58.21	99.46
Overall Accuracy						**66.52**
Kappa Value						**0.54**

Table B.25 Confusion Matrix of WLD on Region 2 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	38137	0	1326	28896	85.15	55.79
Water	10	115386	24291	2128	85.49	81.36
Crop	69	19585	91142	3962	54.99	79.42
Barren	6570	2	48993	80321	69.66	59.11
Overall Accuracy						**70.52**
Kappa Value						**0.6**

Table B.26 Confusion Matrix of WLD on Region 2 (P: 16 and R: 2)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	52795	0	912	14652	69.05	77.23
Water	1052	122541	15069	3153	82.24	86.41
Crop	1446	26192	82882	4238	54.81	72.22
Barren	21162	278	52344	62102	73.80	45.70
Overall Accuracy						**69.51**
Kappa Value						**0.59**

Table B.27 Confusion Matrix of WLD on Region 2 (P: 24 and R: 3)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	63584	9	199	4567	54.82	93.01
Water	4206	127203	5025	5381	71.28	89.70
Crop	3465	46779	56276	8238	56.28	49.04
Barren	44729	4459	38498	48200	72.61	35.47
Overall Accuracy						**64.07**
Kappa Value						**0.52**

Table B.28 Confusion Matrix of WLD on Region 2 (Multiscale)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	53352	0	743	14264	75.32	78.05
Water	376	121756	16087	3596	81.74	85.86
Crop	664	26973	82023	5098	55.72	71.47
Barren	16442	227	48355	70862	75.53	52.15
Overall Accuracy						**71.18**
Kappa Value						**0.61**

Table B.29 Confusion Matrix of VAR on Region 1 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	61704	20004	0	17215	83.48	62.38
Orchard	1151	100259	209	14261	73.83	86.52
Water	9868	0	12322	31401	98.33	22.99
Barren	1196	15530	0	74871	54.35	81.74
Overall Accuracy						**69.21**
Kappa Value						**0.57**

Table B.30 Confusion Matrix of VAR on Region 2 (P: 16 and R: 2)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	51705	32002	0	15216	75.22	52.27
Orchard	3298	100450	397	11735	67.11	86.68
Water	11470	1	17749	24371	97.81	33.12
Barren	2268	17216	0	72113	58.42	78.73
Overall Accuracy						**67.23**
Kappa Value						**0.54**

Table B.31 Confusion Matrix of VAR on Region 1 (P: 24 and R: 3)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	55425	33771	0	9727	73.17	56.03
Orchard	6623	99092	459	9706	65.84	85.51
Water	7069	0	18167	28355	97.54	33.90
Barren	6630	17645	0	67322	58.48	73.50
Overall Accuracy						**66.67**
Kappa Value						**0.54**

Table B.32 Confusion Matrix of VAR on Region 1 (Multiscale)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	56373	29471	0	13079	79.66	56.99
Orchard	2736	100351	386	12407	68.20	86.60
Water	9563	0	16414	27614	97.70	30.63
Barren	2094	17323	0	72180	57.61	78.80
Overall Accuracy						**68.15**
Kappa Value						**0.56**

Table B.33 Confusion Matrix of VAR on Region 2 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	65600	0	0	2759	56.20	95.96
Water	4336	100552	15774	21153	85.89	70.90
Crop	1841	16519	75690	20708	72.34	65.96
Barren	44952	0	13165	77769	63.54	57.23
Overall Accuracy						**69.36**
Kappa Value						**0.59**

Table B.34 Confusion Matrix of VAR on Region 2 (P: 16 and R: 2)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	66753	0	0	1606	46.99	97.65
Water	14625	97849	21862	7479	86.83	69.00
Crop	3301	14835	76590	20032	68.26	66.74
Barren	57394	0	13756	64736	**68.98**	**47.64**
Overall Accuracy						**66.39**
Kappa Value						**0.56**

Table B.35 Confusion Matrix of VAR on Region 2 (P: 24 and R: 3)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	67160	0	0	1199	42.74	98.25
Water	18750	90882	27807	4376	87.43	64.08
Crop	4708	13069	74183	22798	64.57	64.64
Barren	66503	0	12902	56481	66.56	41.56
Overall Accuracy						**62.65**
Kappa Value						**0.51**

Table B.36 Confusion Matrix of VAR on Region 2 (Multiscale)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	66700	0	0	1659	47.90	97.57
Water	12895	96178	23066	9676	87.08	67.82
Crop	3008	14272	76647	20831	68.12	66.79
Barren	56648	0	12800	66438	67.38	48.89
Overall Accuracy						**66.40**
Kappa Value						**0.56**

Table B.37 Confusion Matrix of LBPRIUVAR on Region 1 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	58942	23322	0	16659	84.61	59.58
Orchard	1166	101311	123	13280	72.46	87.43
Water	9082	0	13470	31039	99.10	25.13
Barren	475	15190	0	75932	55.46	82.90
Overall Accuracy						**69.35**
Kappa Value						**0.57**

Table B.38 Confusion Matrix of LBPRIUVAR on Region 1 (P: 16 and R: 2)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	53590	30854	0	14479	78.39	54.17
Orchard	3918	100992	328	10642	69.47	87.15
Water	10138	0	17979	25474	98.21	33.55
Barren	717	13532	0	77348	60.46	84.44
Overall Accuracy						**66.42**
Kappa Value						**0.55**

Table B.39 Confusion Matrix of LBPRIUVAR on Region 1 (P: 24 and R: 3)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	65205	23475	0	10243	76.21	65.91
Orchard	8333	98962	434	8151	73.43	85.40
Water	6638	0	18337	28616	97.69	34.22
Barren	5389	12341	0	73867	61.11	80.64
Overall Accuracy						**71.22**
Kappa Value						**0.60**

Table B.40 Confusion Matrix of LBPRIUVAR on Region 1 (Multiscale)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	59218	26790	0	12915	82.02	59.86
Orchard	3258	100857	304	11461	71.16	87.04
Water	8807	0	16613	28171	98.20	31.00
Barren	913	14090	0	76594	59.31	83.62
Overall Accuracy						**70.36**
Kappa Value						**0.59**

Table B.41 Confusion Matrix of LBPRIUVAR on Region 2 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	65609	0	0	2750	56.53	95.98
Water	3438	96633	22149	19595	87.74	68.14
Crop	1818	13506	78052	21382	69.01	68.01
Barren	45205	0	12900	77781	64.01	57.24
Overall Accuracy						**69.02**
Kappa Value						**0.59**

Table B.42 Confusion Matrix of LBPRIUVAR on Region 2 (P: 16 and R: 2)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	66798	0	0	1561	47.98	97.72
Water	12122	96434	26007	7252	87.79	68.00
Crop	3203	13406	77960	20189	66.36	67.93
Barren	57094	0	13521	65271	69.24	48.03
Overall Accuracy						**66.50**
Kappa Value						**0.56**

Table B.43 Confusion Matrix of LBPRIUVAR on Region 2 (P: 24 and R: 3)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	67202	0	0	1157	44.69	98.31
Water	12655	91307	31721	6132	87.88	64.38
Crop	4130	12587	74488	23553	62.80	64.91
Barren	66380	0	12407	57099	64.93	42.02
Overall Accuracy						**62.95**
Kappa Value						**0.51**

Table B.44 Confusion Matrix of LBPRIUVAR on Region 2 (Multiscale)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	66708	0	0	1651	49.31	97.58
Water	9235	94675	27455	10450	88.01	66.76
Crop	2857	12896	77743	21262	66.03	67.75
Barren	56473	0	12545	66868	66.71	49.21
Overall Accuracy						**66.40**
Kappa Value						**0.56**

Table B.45 Confusion Matrix of WLDVAR on Region 1 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	84113	3870	0	10940	94.69	85.03
Orchard	232	101572	113	13963	89.49	87.65
Water	4481	0	13611	35499	99.18	25.40
Barren	0	8058	0	83539	58.04	91.20
Overall Accuracy						**78.57**
Kappa Value						**0.70**

Table B.46 Confusion Matrix of WLDVAR on Region 1 (P: 16 and R: 2)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	68404	17336	0	13183	93.45	69.15
Orchard	1466	102113	194	12107	78.68	88.12
Water	3325	1	16247	34018	98.82	30.32
Barren	2	10337	0	81258	57.81	88.71
Overall Accuracy						**74.45**
Kappa Value						**0.65**

Table B.47 Confusion Matrix of WLDVAR on Region 1 (P: 24 and R: 3)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	79440	7945	0	11538	92.86	80.30
Orchard	2327	101288	229	12036	89.70	87.41
Water	1072	0	14547	37972	98.45	27.14
Barren	2712	3690	0	85195	58.06	93.01
Overall Accuracy						**77.91**
Kappa Value						**0.69**

Table B.48 Confusion Matrix of WLDVAR on Region 1 (Multiscale)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	House	Orchard	Water	Barren		
House	80195	6951	0	11777	95.83	81.07
Orchard	865	102214	183	12618	87.88	88.21
Water	2621	0	14940	36030	98.79	27.88
Barren	0	7145	0	84452	58.29	92.20
Overall Accuracy						**78.28**
Kappa Value						**0.7**

Table B.49 Confusion Matrix of WLDVAR on Region 2 (P: 8 and R: 1)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	61240	0	19	7100	66.91	89.59
Water	602	107817	20472	12924	86.17	76.03
Crop	410	17300	81838	15210	67.77	71.31
Barren	29279	0	18437	88170	71.45	64.89
Overall Accuracy						**73.58**
Kappa Value						**0.64**

Table B.50 Confusion Matrix of WLDVAR on Region 2 (P: 16 and R: 2)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	66263	0	33	2063	53.82	96.93
Water	5975	107825	20521	7494	86.77	76.03
Crop	2498	16441	80315	15504	67.51	69.99
Barren	48378	0	18095	69413	73.47	51.08
Overall Accuracy						**70.27**
Kappa Value						**0.6**

Table B.51 Confusion Matrix of WLDVAR on Region 2 (P: 24 and R: 3)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	67217	0	4	1138	45.95	98.33
Water	12167	103465	21143	5040	86.10	72.96
Crop	4152	16698	74845	19063	66.56	65.22
Barren	62732	0	16454	56700	69.20	41.73
Overall Accuracy						**65.58**
Kappa Value						**0.55**

Table B.52 Confusion Matrix of WLDVAR on Region 2 (Multiscale)

True class	Assigned class				User Accuracy (%)	Producer Accuracy (%)
	Orchard	Water	Crop	Barren		
Orchard	66066	0	3	2290	56.31	96.65
Water	4287	106677	21074	9777	86.50	75.22
Crop	1818	16656	80276	16008	67.54	69.95
Barren	45153	0	17502	73231	72.29	53.89
Overall Accuracy						**70.8**
Kappa Value						**0.61**

APPENDIX C

Running Time of Several Texture Descriptors

Table C.1 Quantizing Running Time of LBP, LBPRIU and WLD over Region 1

P,R	LBP (s)	LBPRIU (s)	WLD (s)
8.1	0.000368	0.000596	0.00243
16,2	0.00075	0.000967	0.002443
24,3	0.001073	0.001298	0.002573
Multiscale	0.169796	0.002822	0.002822
Mean	0.00124775	0.00142075	0.00142075

Table C.2 Quantizing Running Time of LBP, LBPRIU and WLD over Region 2

P,R	LBP (s)	LBPRIU (s)	WLD (s)
8.1	0.000128	0.000211	0.001768
16,2	0.000282	0.000374	0.001767
24,3	0.000283	0.000395	0.001921
Multiscale	0.17	0.000885	0.005523
Mean	0.00039375	0.00045975	0.00274475

Table C.3 Classification Running Time of LBP, LBPRIU and WLD over Region 1

P,R	LBP (s)	LBPRIU (s)	WLD (s)
8.1	829.001358	1203.21492	3695.83216
16,2	8448.68332	1984.12408	4075.49974
24,3	172240.34	2588.54813	4241.09492
Multiscale	437522.945	5589.3909	11605.4606
Mean	154760.242	2841.23155	5904.47184

Table C.4 Classification Running Time of LBP, LBPRIU and WLD over Region 2

P,R	LBP (s)	LBPRIU (s)	WLD (s)
8.1	140.436048	119.011968	2892.27
16,2	5511.99753	162.880179	2832.88
24,3	131525.736	176.259196	2870.21
Multiscale	181277.084	411.333309	9035.03
Mean	79613.8133	217.37	4407.6

Table C.5 Quantizing Running Time of TD with Combination of VAR over Region 1

P,R	LBPRIU (s)	WLD (s)	VAR (s)	LBPRIUVAR (s)	WLDVAR (s)
8.1	0.000596	0.00243	0.001455	0.001901	0.003831
16,2	0.000967	0.002443	0.002504	0.002504	0.004829
24,3	0.001298	0.002573	0.003342	0.003342	0.00561
Multiscale	0.002822	0.002822	0.007271	0.007271	0.015053
Mean	0.00142075	0.00142075	0.003643	0.003643	0.00733075

Table C.6 Quantizing Running Time of TD with Combination of VAR over Region 2

P,R	LBPRIU (s)	WLD (s)	VAR (s)	LBPRIUVAR (s)	WLDVAR (s)
8.1	0.000211	0.001768	0.000523	0.000744	0.002535
16,2	0.000374	0.001767	0.000816	0.001036	0.002762
24,3	0.000369	0.001921	0.001076	0.001292	0.002928
Multiscale	0.000885	0.005523	0.002323	0.003045	0.010229
Mean	0.00045975	0.00274475	0.0011845	0.00152925	0.0046135

Table C.7 Classification Running Time of TD with Combination of VAR over Region 1

P,R	LBPRIU (s)	WLD (s)	VAR (s)	LBPRIUVAR (s)	WLDVAR (s)
8.1	1203.21492	3695.83216	2825.53922	3628.19489	6252.92081
16,2	1984.12408	4075.09492	4930.1244	6440.13575	8448.68332
24,3	2588.54813	4241.09492	6679.93056	8453.28154	10331.8208
Multiscale	5589.3909	11605.4606	14222.9341	18078.029753	24901.2915
Mean	2841.23155	5904.47184	7164.63208	9149.9104833	12483.6791

Table C.8 Classification Running Time of TD With Combination of VAR over Region 2

P,R	LBPRIU (s)	WLD (s)	VAR (s)	LBPRIUVAR (s)	WLDVAR (s)
8.1	119.011968	2892.27	288.767712	355.526389	3433.05336
16,2	162.880179	2832.88	394.41972	497.556204	3504.99955
24,3	176.259196	2870.21	446.035103	554.189596	3521.08767
Multiscale	411.333309	9035.03	1077.63524	1359.630693	11202.4338
Mean	217.37	4407.6	537.64138	691.72572025	5415.39359

CURRICULUM VITAE

NAME	Mr. Decky Aspandi Latif
DATE OF BIRTH	18 December 1990
EDUCATIONAL RECORD	
DIPLOMA’S DEGREE	High School Graduation Public High School 3, Samarinda ,2008
BACHELOR’S DEGREE	Bachelor of Science(Computer Science) Mulawarman University, 2012
MASTER’S DEGREE	Master of Science (Computer Engineering) King Mongkut’s University of Technology Thonburi, 2014
SCHOLARSHIP	Beasiswa Peningkatan Prestasi Akademik (PPA) 2009 – 2012 Kaltim Scholarship Dual Degree Program 2012 – 2014
EMPLOYMENT RECORD	Lab Assistant Department of Computer Science Mulawarman University Samarinda, 2009 - 2012 Lecturer Assistant Department of Computer Science Mulawarman University Samarinda, 2012 Lecturer Assistant Department of Computer Engineering Mulawarman University Samarinda, 2012 Software Developer Dewan Pendidikan Kalimantan Timur Samarinda, 2012 - 2013
PUBLICATION	Decky Aspandi Latif, Sally E. Goldin and Sumaryono, M., 2014, “Texture Based Classification of High Resolution Imagery using Weber Local Descriptor“, **10 th International Conference on Asia GIS**, Vol. 10, 16-17 June, pp. 46. Chiang Mai, Thailand