ANALYSIS GRADE OF RIBBED SMOKED SHEET WITH IMAGE PROCESSING TECHNIQUE

NARET CHANTHARANGSIKUL

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIRMENTS FOR
THE DEGREE OF MASTER OF SCIENCE
(COMPUTER SCIENCE)
FACULTY OF GRADUATE STUDIES
MAHIDOL UNIVERSITY
2010

Thesis
entitled

ANALYSIS GRADE OF RIBBED SMOKED SHEET WITH IMAGE PROCESSING TECHNIQUE

Naret Chantharangsikul

....

Acting 1Lt. Naret Chantharangsikul
Candidate

Chomtip Pornpanomchai

....

Asst. Prof. Chomtip Pornpanomchai, Ph.D.
Major advisor

....

Assoc. Prof. Damras Wongsawang, Ph.D.
Co-advisor

Sukanya P.

....

Asst. Prof. Sukanya Phongsuphap, Ph.D.
Co-advisor

B. Mahai

....

Prof. Banchong Mahaisavariya, M.D.
Dean
Faculty of Graduate Studies
Mahidol University

....

Assoc. Prof. Supachai Tangwongsan, Ph.D.
Program Director
Master of Science Programme
in Computer Science
Faculty of Information and
Communication Technology
Mahidol University

Thesis
entitled
ANALYSIS GRADE OF RIBBED SMOKED SHEET WITH IMAGE PROCESSING TECHNIQUE

was submitted to the Faculty of Graduate Studies, Mahidol University
for the degree of Master of Science (Computer Science)
on
February 23, 2010

Naret Chantharangsikul
....
Acting 1Lt. Naret Chantharangsikul
Candidate

Panjai Tantatsanawong
....
Assoc. Prof. Panjai Tantatsanawong, Ph.D.
Chair

Chomtip Pornpanomchai
....
Asst. Prof. Chomtip Pornpanomchai, Ph.D.
Member

Sukanya P.
....
Asst. Prof. Sukanya Phongsuphap, Ph.D.
Member

Damras Wongsawang
....
Assoc. Prof. Damras Wongsawang, Ph.D.
Member

B. Mahai
....
Prof. Banchong Mahaisavariya, M.D.
Dean
Faculty of Graduate Studies
Mahidol University

Jarernsri L. Mitrpanont
....
Assoc. Prof. Jarernsri L. Mitrpanont, Ph.D.
Acting Dean
Faculty of Information and
Communication Technology
Mahidol University

ACKNOWLEDGEMENTS

First of all, I would like to express my sincere gratitude to my advisor, Asst. Prof. Chomtip Pornpanomchai whose help, stimulating suggestions, invaluable guidance, patience, responsiveness and encouragement helped me in all the time of research for and writing of this thesis. I'm also very gratefully to my co-advisor, Assoc. Prof. Damras Wongsawang and Asst. Prof. Sukanya Phongsuphap, for their invaluable advice and support in this research.

I wish to extend my thanks to Mr. Somchit Sakharintramas, the chief of the Central Rubber Market for his permissions of allowing me to obtain images of Ribbed Smoked Sheets. In this connection, I am most indebted to Mr. Sukhum Kaewklub, Mr. Sophon Chantarachote for their help in collecting most of the data images. I must reacknowledge my debt to Mr. Wisut Phetmongkol who not only helps in acquiring a lot of images but also suggestion me about property of every RSS grades. Most of all, I am grateful to those people who caring and cheering me at the office of Hat Yai Central Rubber Market for their wonderful friendships.

I would like to thank Miss Woramon Chulacharit for her kindness to help me proof and improve my wording in this thesis. I also wish to express my gratitude to my uncle, Mr.Somjai Chantharangsikul for his time driving me to collect data at Songkhla province and my relatives who support accommodations for me.

Especially, I would like to give my special thanks to my mother for her encouragement, entirely care, love and helped me everything even collecting data. The usefulness of this thesis, I dedicate to my father, my mother and all the teachers who have taught me and my sister.

Naret Chantharangsikul

ANALYSIS GRADE OF RIBBED SMOKED SHEET
WITH IMAGE PROCESSING TECHNIQUE

NARET CHANTHARANGSIKUL 4737222 SCCS/M

M.Sc. (COMPUTER SCIENCE)

THESIS ADVISORY COMMITTEE: CHOMTIP PORNPANOMCHAI, Ph.D., DAMRAS WONGSAWANG, Ph.D., SUKANYA PHONGSUPHAP, Ph.D.

ABSTRACT

Currently, rubber factories employ experts or scientists to analyse the grade of Ribbed Smoked Sheet (RSS) using visual sight and experience. Grading is carried out by visual inspection according to the standards specified in the Green Book. An expert or scientist will inspect rubber sheets and separate the RSS to RSS1, RSS2, RSS3, RSS4 and RSS5 depending on the properties of each rubber sheet.

The objective of this research is to propose a computer system that can help rubber experts or agriculturists to determine RSS grades. This system is called the "Analysis Grade of Ribbed Smoked Sheet (AGRSS) system". The system consists of 5 main processes, which are 1) Image Acquisition, 2) Segmentation, 3) Clustering, 4) Calculation of RSS grades, and 5) Display of results. In the image acquisition process, we use a digital camera to take RSS images in a controlled environment box. In the segmentation process, we apply several image processing methods to prepare a suitable RSS image for a clustering process. In the clustering process, we apply L*a*b* color space (The L*a*b* color space consists of a luminosity layer 'L*', a chromaticity-layer 'a*' that indicates where the color falls along the red-green axis, and a chromaticity-layer 'b*' that indicates where the color falls along the blue-yellow axis.), Euclidean distance and K-means clustering to automatically find the best groupings for the RSS image. The calculation of RSS grades will classify RSS into five grades, which are RSS1 – RSS5. In the display of results process, we create a graphic user interface (GUI) for displaying RSS grade results.

This research tested the system by using 398 RSS images for a training dataset and another 398 RSS images for a test dataset. The precision rate of this work was 80.65 percent for the test dataset. The results indicate that the computer method is almost as accurate when compared with a human expert who inspects the RSS grades. The average access time for the AGRSS was around 10.83 seconds per RSS image.

KEY WORDS: RIBBED SMOKED SHEET (RSS) / RSS GRADING
CLUSTERING / COLOR IMAGE PROCESSING

259 pages

การวิเคราะห์ชั้นของยางแผ่นรมควันด้วยวิธีการประมวลผลภาพ
ANALYSIS GRADE OF RIBBED SMOKED SHEET WITH IMAGE PROCESSING TECHNIQUE

นเรศ จันทรังสิกุล 4737222 SCCS/M

วท.ม. (วิทยาการคอมพิวเตอร์)

คณะกรรมการที่ปรึกษาวิทยานิพนธ์ : ชมทิพ พรพนมชัย, Ph.D., ดำรัส วงศ์สว่าง, Ph.D., สุกัญญา พงษ์สุภาพ, Ph.D.

บทคัดย่อ

ในปัจจุบันนี้ อุตสาหกรรมยางพาราได้จ้างงานผู้เชี่ยวชาญหรือนักวิทยาศาสตร์ ในการปฏิบัติงานด้านการวิเคราะห์ชั้นของยางแผ่นรมควันด้วยสายตาและประสบการณ์ การวิเคราะห์ชั้นของยางแผ่นรมควันอาศัยตามมาตรฐานสากล (Green Book) ผู้เชี่ยวชาญหรือนักวิทยาศาสตร์จะตรวจสอบแผ่นยางแล้วแยกแผ่นยางออกเป็น ยางแผ่นรมควันชั้น1, ชั้น2, ชั้น3, ชั้น4, และชั้น5 โดยยางแผ่นรมควันแต่ละชั้นจะขึ้นอยู่กับคุณภาพของยางแต่ละแผ่น

วัตถุประสงค์ของงานวิจัยนี้มุ่งหมายที่จะนำเสนอระบบงานที่สามารถนำไปประยุกต์ใช้กับผู้เชี่ยวชาญและเกษตรกรในการตรวจสอบชั้นของยางแผ่นรมควัน ระบบงานนี้เรียกว่า “โปรแกรมการวิเคราะห์ชั้นของยางแผ่นรมควัน” ระบบงานนี้ประกอบไปด้วย 5 กระบวนการหลักดังนี้ 1) การนำภาพเข้าสู่ระบบ 2) การแบ่งออกเป็นส่วน 3) การจัดกลุ่ม 4) การคำนวณชั้นของยางแผ่นรมควัน 5) การแสดงผลลัพธ์ ในขั้นตอนการนำภาพเข้าสู่ระบบ ใช้กล้องถ่ายภาพดิจิตอลในการถ่ายภาพยางแผ่นรมควัน ภายในกล่องที่มีการควบคุมสภาพแวดล้อม ในขั้นตอนการแบ่งออกเป็นส่วน ได้ประยุกต์ใช้เทคนิคการประมวลผลภาพหลายวิธีเพื่อเตรียมภาพให้เหมาะสมสำหรับนำไปเข้ากระบวนการจัดกลุ่ม ในขั้นตอนการจัดกลุ่ม ได้ประยุกต์ใช้ปริภูมิสี L*a*b* (ปริภูมิสี L*a*b* ประกอบด้วย ชั้นของระดับความเข้มข้นของแสงสว่าง ‘L*’, ชั้นของค่าสี ‘a*’ คือค่าสีที่เกิดขึ้นในช่วงแกนสีแดงถึงสีเขียว, ชั้นของค่าสี ‘b*’ คือค่าสีที่เกิดขึ้นในช่วงแกนสีน้ำเงินถึงสีเหลือง), การวัดระยะแบบยูคลิด และ K-means clustering ในการจัดกลุ่มภาพถ่ายยางแผ่นรมควันแบบอัตโนมัติ ขั้นตอนการคำนวณชั้นของยางแผ่นรมควัน จะคัดแยกยางแผ่นรมควันออกเป็น 5 กลุ่ม ประกอบด้วย ยางแผ่นรมควันชั้น 1 – ชั้น 5 ในการแสดงผลลัพธ์ ได้สร้างหน้าจอสำหรับติดต่อกับผู้ใช้ สำหรับนำเสนอผลลัพธ์ชั้นของยางแผ่นรมควัน

งานวิจัยนี้ใช้ภาพถ่ายยางแผ่นรมควัน 398 ภาพ ในการศึกษาตัวอย่าง และภาพถ่ายอีก 398 ภาพ ในการทดสอบ ความแม่นยำจากการทดสอบระบบคิดเป็น 80.65 เปอร์เซ็นต์ โดยเปรียบเทียบความถูกต้องกับผลลัพธ์ที่ผู้เชี่ยวชาญวิเคราะห์ เวลาเฉลี่ยที่ใช้ในระบบประมาณ 10.83 วินาทีต่อภาพ

259 หน้า

CONTENTS

	Page
ACKNOWLEDGEMENTS	**iii**
ABSTRACT (ENGLISH)	**iv**
ABSTRACT (THAI)	**v**
LIST OF TABLES	**viii**
LSIT OF FIGURES	**x**
CHAPTER I INTRODUCTION	**1**
1.1 Motivation	1
1.2 Problem Statement	2
1.3 Objective	3
1.4 Scope of work	4
1.5 Thesis organization	4
CHAPTER II LITERATURE REVIEW	**6**
2.1 Related Theories	6
2.2 Rubber Problem	27
2.3 Visually Graded Rubber	30
2.4 Digital Image Processing	30
2.5 Related work	32
2.6 Other related works	40
CHAPTER III METHODOLOGY	**44**
3.1 Overall Architecture	44
3.2 System Overview	45
3.3 Structure Chart	47
3.4 Analysis Graded of Ribbed Smoked Sheet (AGRSS) system	48
3.5 K-means clustering	70

CONTENTS (cont.)

	Page
CHAPTER IV SYSTEM DESIGN AND IMPLEMENTATION	**75**
4.1 System Overview	75
4.2 Step of Analysis Grade of Ribbed Smoked Sheet (AGRSS) system	77
4.3 Graphic User Interface (GUI) part	90
4.4 System training phase	99
4.5 System testing phase	114
CHAPTER V EXPERIMENTAL RESULTS	**118**
5.1 Data images acquisition	118
5.2 Testing phase	124
5.3 Discussion	144
CHAPTER VI CONCLUSIONS & SUGGESTIONS FOR FUTURE WORK	**150**
6.1 Conclusion	150
6.2 Suggestions for future work	152
REFERENCES	**154**
APPENDICES	**157**
Appendix A Details of K-means clustering function	158
Appendix B Ribbed Smoked Sheet (RSS) testing set images	180
Appendix C Ribbed Smoked Sheet (RSS) testing set data	231
Appendix D Memorandum of visually graded rubber by experts	258
BIOGRAPHY	**259**

LIST OF TABLES

Table		Page
2.1	Components of the latex	8
2.2	Concentrated latex Thai regulation standard	11
2.3	Unsmoked Sheet quality standard	19
2.4	Standard of Ribbed Smoked Sheet 1 – 5; Property of Fungus	23
2.5	Standard of Ribbed Smoked Sheet 1 – 5; Property of Rubber Sheet	24
2.6	Standard of Ribbed Smoked Sheet 1 – 5; Property of Acceptable flaw	25
2.7	Standard of Ribbed Smoked Sheet 1 – 5; Property of Unacceptable flaw	26
3.1	Lists all the types of images that function *imread* can read	49
3.2	Lists the class of the returned array for all data types used by the file formats	51
3.3	The argument type specifies the conversions	61
3.4	Abbreviations to represent for each color space	62
3.5	The parameters use in calculation of RSS grades	66
3.6	The parameters to control the iterative algorithm	72
4.1	Decision RSS grades table : RSS1	102
4.2	Decision RSS grades table : RSS2	103
4.3	Decision RSS grades table : RSS3	104
4.4	Decision RSS grades table : RSS4	108
4.5	Decision RSS grades table : RSS5	110
4.6	Example of RSS images training set data (color RGB information)	113

LIST OF TABLES (cont.)

Table		Page
4.7	Example of RSS images training set data (process time's information)	113
4.8	Example of RSS images testing set data (color RGB information)	117
4.9	Example of RSS images testing set data (process time's information)	117
5.1	The specification of digital camera	120
5.2	The number of selected images for training and testing phases	124
5.3	Comparing results between "a*b*" and "RGB" color space using k cluster equal to 1 and 2	125
5.4	Comparing k cluster's result in testing phase images group by RSS1 – RSS5	126
5.5	The value in testing phase group by degree of validity (RSS1 – RSS5)	127
5.6	The value of result in testing phase group by RSS grades (RSS1 – RSS5)	127
5.7	The value of result testing phase images group by RSS1	128
5.8	The value of result testing phase images group by RSS2	128
5.9	The value of result testing phase images group by RSS3	131
5.10	The value of result testing phase images group by RSS4	135
5.11	The value of result testing phase images group by RSS5	140

LIST OF FIGURES

Figure		Page
2.1	Chemistry Structure	7
2.2	Spin concentrated latex machine split follow the horizon	9
2.3	Concentrated latex procedure	10
2.4	Production processes of ADS/ RSS	20
2.5	The same picture in three different colors mode	33
3.1	The Physical Diagram for AGRSS system	44
3.2	The System Diagram for AGRSS system	45
3.3	The structure chart for AGRSS system	47
3.4	Example of Ribbed Smoked Sheet	51
3.5	Example of Ribbed Smoked Sheet in grayscale	52
3.6	Example of binary Ribbed Smoked Sheet from Otsu's Method	53
3.7	Morphological operations specify a radius of 2 pixels	54
3.8	Picture of RSS after remove Noise	55
3.9	Find coordinates of RSS image	56
3.10	Trace region boundaries in a binary image	57
3.11	RSS after crop coordinates	58
3.12	Image of RSS after clustering	64
3.13	Example RSS images Cluster 1 – Cluster 2	66
3.14	Result of RSS grade	69
4.1	Process of Analysis Grade of Ribbed Smoked Sheet system	76
4.2	Flow Chart of Take images of RSS	77
4.3	Take images of RSS	78
4.4	Flow Chart of Convert RGB image into grayscale	79
4.5	Convert RGB image into grayscale	80
4.6	Convert grayscale image into binary	80
4.7	Remove Noise	80

LIST OF FIGURES (cont.)

Figure		Page
4.8	Flow Chart of Threshold color pixel	81
4.9	Flow Chart of Remove Noise	82
4.10	Flow Chart of Calculation of RSS area	83
4.11	Find coordinates of RSS area	84
4.12	Flow Chart of Find coordinates of 4 sides	85
4.13	Flow Chart of Crop coordinates of RSS area	86
4.14	Crop coordinates of RSS area	86
4.15	Flow Chart of Clustering process	87
4.16	Clustering process	88
4.17	Calculation of RSS grades	88
4.18	Flow Chart of Calculation of RSS grades	89
4.19	Graphic user interface (GUI) of AGRSS application	90
4.20	Graphic user interface (GUI) of Load File button	91
4.21	Graphic user interface (GUI) of One button show grade panel	92
4.22	Graphic user interface (GUI) of Crop Edge panel	93
4.23	Graphic user interface (GUI) of Crop (Manual) button	94
4.24	Graphic user interface (GUI) of Select RSS1-5 or RSS3-5 panel	95
4.25	Graphic user interface (GUI) of Clustering button	96
4.26	Graphic user interface (GUI) of Before & After Crop button	97
4.27	Graphic user interface (GUI) of Clear Screen button	98
4.28	Process of system training phase	99
4.29	RGB range of RSS grades : RSS1 – RSS5	101
4.30	RGB range of RSS grades : RSS1	102
4.31	RGB range of RSS grades : RSS2	103
4.32	RGB range of RSS grades : RSS3	104
4.33	RGB range of RSS grades : RSS4	108
4.34	RGB range of RSS grades : RSS5	110

LIST OF FIGURES (cont.)

Figure		Page
4.35	Example of RSS images for training	112
4.36	Process of system testing phase	114
4.37	Example of RSS images for testing	116
5.1	Size of dark box simulation	119
5.2	Canon PowerShot A620	119

CHAPTER I
INTRODUCTION

1.1 Motivation

Thailand is the 1st ranking rubber exporter in the World. Manufacturing has produced more than 2 million tons per year [1]. Approximately 90 % of the total rubber production in Thailand was exported and the rest was domestically consumed. Evaluate rubber exported value is about 100,000 million baht per year. Most of them are in form of Ribbed Smoked Sheet (RSS), Rubber Block (RB) and Rubber Latex. Rubber industry was become very important industry in forefront.

Although Thailand is the leader rubber exporter in the world but on the strength of export it has a weakness hidden. Problem issues that caused trouble to agriculturist and manufacturer in Thailand which they could not escape are infrastructure of rubber production from manufacturer such as Ribbed Smoked Sheet was defective in quality control.

Ribbed Smoked Sheets (RSS) are marketed based on visual assessment of quality. RSS grading is carried out by visual inspection according to the standards specified in the Green Book published by the Rubber Manufacturers Association Inc, Washington.

The most serious problem found that agriculturist was exploitation by the middlemen and this situation will not change as long as the agriculturist does not have direct contact with the manufacturer. Some cases caused by dishonest dealers or agriculturist who mix ungraded sheets with grade sheets and sell as good RSS grade to get higher prices. If these situations frequently happen it will be harmful both the producer as well as the consumer.

The lack of research and development of new technology to improve rubber manufacturing cause us to realize that it is necessary to conduct research on rubber in order to develop the quality of rubber product. Then, we will be able to determine the quality of the rubber and with this reason it would be to develop a visually computerized grading system.

1.2 Problem Statement

Most of the rubber growers in Thailand are small groups or individual. They have to bring the products to sell with middlemen in the rubber market. Sometimes they deny dealer reasonable price for the products because of uncertainly standard justification from visually grading system. Even if they reduce the prices but they have to agree with or change to another dealer but it is not a good solution because new dealer may be get a less price and make them lose opportunity to work, pay more traveling expense or waste times.

Grade quality of Ribbed Smoked Sheets (RSS) was analysed based on visual assessment. RSS grading is carried out by visual inspection according to the standards specified in the Green Book or conduct tests for any specific parameters which consumers should require.

Currently, rubber factories employed experts or scientists to analyse the grade of Ribbed Smoked Sheet (RSS) with visual sight and experience. An expert or scientist will inspect rubber sheets and separate the RSS to RSS1, RSS2, RSS3, RSS4 and RSS5 depending on the properties of each rubber sheet.

In the rubber market, most agriculturists had poor education. It caused them to be a victim of immoral middlemen but if they have more experienced about analyse RSS grades it will decrease risk to lost benefit and price from their products. But, if you are a new beginner in this commercial and do not have any knowledge about RSS grades you may be a victim of these dishonest dealers.

Agriculturists will pass on knowledge to their descendant from the beginning. They will teach from their experience on how to plant and how to look after the rubber trees, latex produce from rubber tree, how to collect fresh latex and processes to make rubber sheet thin and clean, and the ways to analyse RSS grades before selling in order to avoid losing the price or benefit from immoral middlemen. This knowledge will pass on from generation to generation.

Sometimes standard may be depend on each expert, scientist, dealer or agriculturists. However the experienced or knowledge of each expert may be caused the uncertainly standard. Uncertainly standard might occur from several factors such as period of time of experienced or different expertise from each group of expert whose performances of skilled may be reduce after hours of long hours work and the important thing is their code of conduct.

Preliminary analysis of RSS grade will affect the rubber price and agriculturist should get benefit when they sold RSS. For these reason, the author intends to study and develop the system that cans analyse grade of Ribbed Smoked Sheet using an image processing technique. The goal is to develop process in analysis grade of Ribbed Smoked Sheet and make it accurately enough and useful or become to standard in the future. The author hopes that this system will be valuable for agriculturist, new grading staff, new scientist or rubber manufacturer that nowadays have a grown up trend.

1.3 Objectives

1. To proposed a system that can analyse grade of Ribbed Smoked Sheet (RSS1 – RSS5) using an image processing technique.
2. Specify color ranges level of each RSS1 – RSS5 grades.
3. To apply clustering technique to cluster nearby color level into groups for analysis RSS grades.

1.4 Scope of work

1. Analysis RSS grades only RSS1 – RSS5.
2. Use surface image of RSS1 – RSS5.
3. RSS1 – RSS5 images must take on dark background.
4. Control light with white light in the dark box.

1.5 Thesis organization

This thesis consists of six chapters listed below;

Chapter 1 : Introduction

This chapter comprises of the motivation of this work, the problem statement, objectives, as well as scope and the thesis organization.

Chapter 2 : Literature Review

This chapter describes about related theories and techniques used in this thesis. Identify the rubber problems and traditional analysis visually graded of Ribbed Smoked Sheet (RSS). A survey on previous related works and other related worked is conducted.

Chapter 3 : Methodology

This chapter explains about overall architecture, system overview, structure chart, clustering technique, and processes of Analysis Graded of Ribbed Smoked Sheet (AGRSS) system that shows functions, variables and definition using in MATLAB programming.

Chapter 4 : System Design and Implementation

This chapter discusses about detailed design and development methods in the experiments that include system overview, the step of Analysis Grade of Ribbed Smoked Sheet (AGRSS) system, graphic user interface (GUI) part, system training and system testing phase.

Chapter 5 : Experimental Results

This chapter illustrates the experimental results and its accuracies. The difficulties problems that are found in work. Discussed the system and described the results.

Chapter 6 : Conclusions and Suggestions for Future Work

This chapter contains the conclusion of the study. In addition, the improvements for future work are also suggested.

CHAPTER II
LITERATURE REVIEW

This chapter describes the related theories, normal operating conditions to analyse grades of Ribbed Smoked Sheets and identifies common problems. Then, the desirable features to analyse grades of Ribbed Smoked Sheets are presented. A survey on relate work is conducted.

2.1 Related Theories

The natural rubber (NR) presently used by industry is obtained by tapping the sap known as 'Latex', from the large forest tree *Hevea Brasiliensis,* which were found in the southern equatorial region of America. By the end of eighteenth century the properties of rubber as obtained from the Hevea tree available at that time entirely in the forest of Amazon valley, were known throughout Europe. The Europeans found that by systematically tapping the tree, the latex can be extracted regularly. With the development of plantation in the Far East, it was found that latex could be preserved by adding ammonia to it immediately after it is collected [2].

2.1.1 Growing rubber in Thailand

Trang was the first city where rubber was planted. Praya Ratsadanupradit Mahison Phakdi brought rubber from Malaysia and planted it here before anywhere else in southern Thailand in 1899 [3]. The first rubber seeds were planted in Kantang district, villagers calls this first rubber group that "Yang Tesa" then have the rubber reproduction comes to grow in the Trang area and Narathiwat. In 1911 have the rubber reproduction in Chanthaburi which be the East of Thailand by Luang Rachamontree. Then rubber reproduction spreads all 14 southern provinces, eastern, middle, northeast, and northern since 1991. Since then, rubber has become South's major commodity and is an important source of income for agriculturist in Thailand.

2.1.2 Chemistry Structure

Natural rubber is an elastic hydrocarbon polymer that naturally occurs as a milky colloidal suspension, or *latex*, in the sap of some plants. It can also be synthesized. Synthetic rubbers are made by the polymerization of a single monomer or a mixture of monomers to produce polymers. Synthetic rubber can be made as a polymer of isoprene or various other monomers. These form part of a broad range of products extensively studied by polymer science and rubber technology. Examples are SBR or styrene-butadiene rubber, BR or butadiene rubber, CR or chloroprene rubber and EPDM (ethylene-propylene-diene rubber).

The entropy model of rubber was developed in 1934 by Werner Kuhn. It has the large-sized molecule compose the division digests one kind and important characteristic is the flexibility. Chemistry structure of natural rubber compose with 5 atom carbons, and 8 atom hydrogens (C_5H_8) be known in chemistry way that *isoprene*. When the division digests connect to molecule it will arrange into cis-1, 4 configuration as follow;

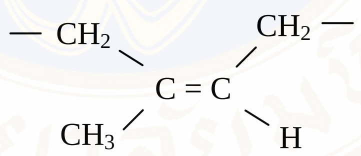


Figure 2.1 Chemistry Structure

Natural rubber be known in chemistry method is called *polyisoprene*, has molecule weigh more than 1 million. The plants which give the latex and can apply the advantage in commercial are natural rubber get from *Hevea*'s family plant or call that Para rubber and Guayule rubber. The scientific name for the rubber tree is *Hevea brasiliensis*.

2.1.3 Components of the latex

Fresh latex from the rubber tree is a white liquid or light brown by having rubber suspendible particle stay in the medium fluently. Rubber particle has round figure or pear-shaped, there is the 0.05 - 5 micrometer size, densities 0.975 – 0.980 gram per milliliter. There is the acid – alkaline about 6.5 – 7.0, generally latex quantity in nature latex might vary from 25–45% but on the average have component as follow;

Table 2.1 Components of the latex

Components	Average Percentage (By the weight)
A substance that is all solid	**36**
Dry latex	33
Group protein substance and the fat	1 – 1.2
Group carbohydrate substance	1
Ashes	1
Water	**64**

The skin of rubber particle has a membrane covers that compose the fat and the protein. Each the particle has anion of protein stays the periphery. The power pushes between rubber particles make the latex available in a state of liquid. When there is membrane destruction covers the particle will make rubber suspendible particle in the medium which is born combination crystallize.

2.1.4 Latex Preservation

When we get fresh latex from the rubber tree, if there is not state latex preservation, the microorganism in the air will mixed up in latex and used group of sugar substance as the food which causes acid. That is cation happen and react anion of protein around particle rubber skin. It will make the latex lose the state before induce to process. Therefore, we must have state latex preservation by chemicals addition such as;

Ammonia Water	0.05	Percentage per latex weight
Sodium Sulfide	0.02 - 0.05	Percentage per latex weight

Transform fresh latex status for producing the raw material of rubber products to rubber industry can separate in 2 types as following: 1. Concentrated latex production, 2. Dry rubber production.

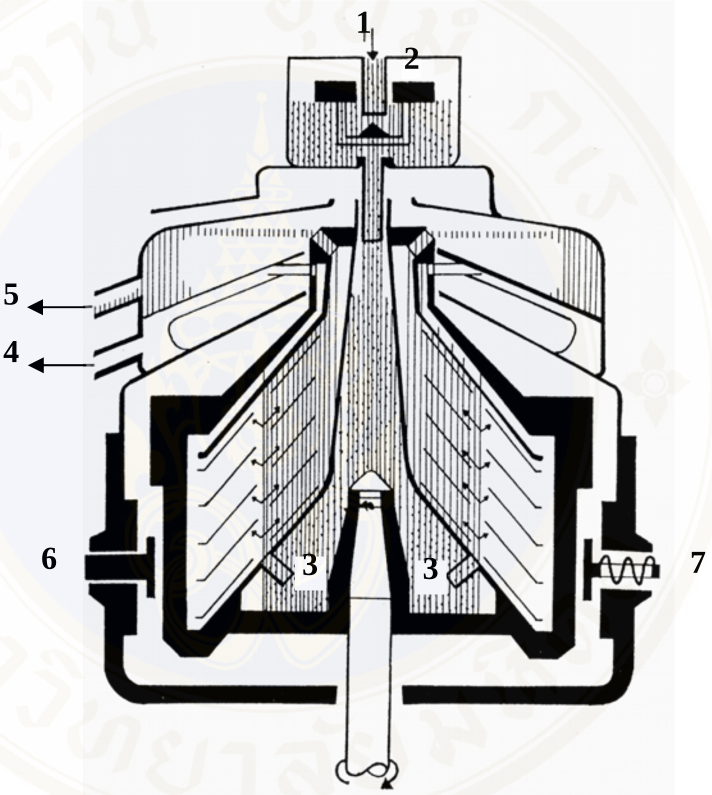


Figure 2.2 Spin concentrated latex machine split follow the horizon

(Image from Rubber Technical documents [1])

1. Pipe takes
2. Pot latexes rest and a child floats
3. Pipe pays
4. Tail latex exit
5. Concentrated latex exit
6. Mechanical break
7. Magnet break

Fresh latex from a rubber plantation has rubber quantity average about 33 percentages which is inconvenient in shipping and trading. Besides that, there is still inappropriate suit to reach the procedure produces to maintain good quality regularly. So, we should keep it as a concentrated latex and have rubber at least 60 percentages.

There are 4 procedures to produces concentrated latex: 1. Evaporate the water, 2. Collect the remaining which is the latex, 3. Spin, and 4. Separate with the electricity. But concentrated latex production in Thailand use the spinning with high speed millstone, for separating the water from other substance which remain when the water drain out. The principle works illustrating in Figure 2.2 and the process in concentrated on latex production was show in Figure 2.3.

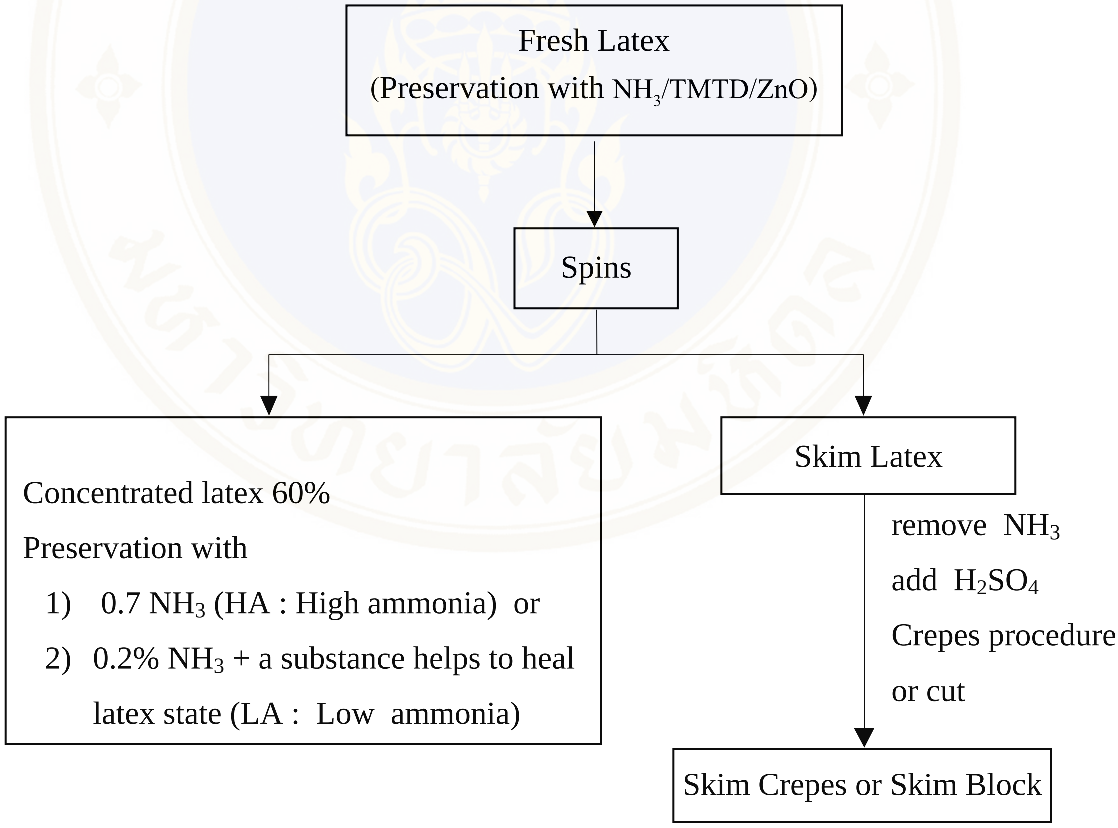


Figure 2.3 Concentrated latex procedure

Concentrated latex consumer would likes to buy the standardized quality control that is concentrated latex by Thai regulation standard show in Table 2.2.

Concentrated latex preservation has 5 systems which are;

1. Concentrated latex high Ammonia, use Ammonia 0.7 percentage quantities.
2. Concentrated latex low Ammonia, use Ammonia 0.2 percentage quantities with Sodium pentachlorophenate 0.2 percentage quantities.
3. Concentrated latex low Ammonia, use Ammonia 0.2 percentage quantities with Bolic acid 0.24 percentage quantities.
4. Concentrated latex low Ammonia, use Ammonia 0.2 percentage quantities with Zinc diethyl dithiocarbamate (in substance suspendible) 0.2 percentage quantities.
5. Concentrated latex low Ammonia, use Ammonia 0.2 percentage quantities with Tetramethylthiuram disulphide (TMTD) (in substance suspendible) 0.013 percentage quantities and Zinc Oxide (Zn O) 0.013 percentage quantities.

Table 2.2 Concentrated latex Thai regulation standard
(Thai Industrial Standards Institute 980 - 1990)

Property	**Concentrated latex limitation (kind Spins)**	
	HA	**LA**
All solid quantity[1/], %(mass/mass), Minimum	61.5	61.5
Latex dry quantity, %(mass/mass), Minimum	60.0	60.0
Solid quantity that is not latex[2/], %(mass/mass), Maximum	1.8	1.8
Alkaline (Ammonia), %(mass/mass) of latex	0.60 (Min)	0.29 (Max)
The machinery stability time[3/] , second, Minimum	650	650
The quantity of latex stick together, %(mass/mass), Maximum	0.05	0.05
Copper quantity, milligram/kilogram, of all solid quantity, Maximum	8	8
Manganese quantity, milligram/kilogram, of all solid quantity, Maximum	8	8
Sediment quantity, %(mass/mass), Maximum	0.10	0.10

Table 2.2 Concentrated latex Thai regulation standard (Continued)

Property	Concentrated latex limitation (kind Spins)	
	HA	LA
Fatty acid amount can evaporate (VFA No.)	according to the agreement between the producer and the user but must not exceed than 0.15	
Potassium hydroxide quantity[4/] (KOH No.)	according to the agreement between the producer and the user but must not exceed than 1.0	
Checking color with the sight.	Blue or gray are not acceptable.	
Checking smell later making is impartial with Bolic acid.	Does not have the offensive odor such as decays smelling.	

[1/] All solid quantity can choose according to desire.

[2/] The difference between all solid quantities with dry rubber quantity.

[3/] Stability time per minimum machinery might have value more than the value that notes.

[4/] If latex composes with Bolic acid, the amount of Potassium Hydroxide might exceed more than the value that noted because the quantity has equivalent to Bolic acid which test by the ISO 1802 method.

2.1.5 Latex Tapping

Tapping is usually done by shaving about one or two millimeters thickness of bark with each cut, usually in the early morning hours, after which latex flows for several hours and gets collected in cups mounted on each tree. The cut is made with special knife or gouge, sloping from left to right at about 20 - 30° from the horizontal. The content of each latex cup is transferred to five-gallon containers and transported to storage tanks at bulking station [2]. For convenience on tapping the sap and still heal the clean of fresh latex. Agriculturist ought to keep the shell for collection latex clean, the sharpness of a knife that uses to tapping must always sharp and should consider slope level of a trace cuts.

Latex tapping time: Appropriate time to tapping the sap is 6.00 – 8.00 o'clock in the morning because at this times we can see the rubber trees clearly and get latex quantity in similar to tapping rubber in the early morning but the sap shaving between 1.00 - 4.00 o'clock in the very early morning will get latex quantity more than at dawn about 4 – 5 percentage. The very early morning is the times that get the best latex quantity but it also have disadvantage such as miss cut a membrane thrives of rubber trees that cause the skin rubber disease and unsafely from beasts or criminal.

Rest tapping time: In summer, a leaf sheds leaves or the season that have new bloom agriculturists will stop tapping because tapping the rubber at this time will effect of leaf and rubber development.

2.1.5.1 Increase tapping time

- **Compensate tapping**: The tapping day that lost in rainy season, agriculturist could shaving between 6.00 – 8.00 o'clock in the morning if it is raining at night.

- **Late tapping**: If the rubber wet or it is raining, agriculturist can tapping after regularly times by shaving rubber in the morning or in the evening but in hot whether should not shaving rubber.

2.1.6 Latex Collection

The shell of half a coconut is used as the collection container for the latex. The shells are attached to the tree via a short sharp stick and the latex drips down into it overnight. This usually produces latex up to a level of half or three quarters of the shell. The latex from multiple trees will poured into flat pans and mixed with formic acid which serves as a coagulant resulting in rubber tray. After a few hours, the very wet sheets of rubber are wrung out and putting them through a press machine before sent them to factories for vulcanization and further processing is done to it.

2.1.7 Rubber Sheet Production

When collect fresh latex from a rubber plantation already, use the filter for separating the dirtiness and the adulterated thing. Add a substance that makes the latex coagulant and putting them through a press machine that dried them. By might make Unsmoked Sheet (USS), Air Dried Sheet (ADS) or Ribbed Smoked Sheet (RSS).

2.1.7.1 Unsmoked Sheet production

The quality of Unsmoked Sheet depends on the way how to produces. The prices in the market different follow the quality of rubber sheet. Therefore, if rubber plantation owner or agriculturist can produce rubber sheet that have high quality to sell will get higher price than low quality rubber sheet. High quality rubber sheet have an easy principle procedure that should be careful such as rubber cleanliness, press rubber sheet thinly, a rubber sheet has a same color regularly, use the water and acid ratio by the instruction as follows;

Step 1: Latex collection

1. Wipe the rubber cup clean before support the latex.
2. Clean bucket latex before using every time.
3. Don't add rubber gum or leaf in bucket latex. It will make rubber dirty, coagulate fast and difficult to filter the latex.
4. Latex bucket is supposed to have the lid for protection the latex bulges out of the bucket during transport them to rubber sheet factory.

Step 2: Cleaning tool

Clean every rubber sheet tools before and after used. Rubber sheet tools should be wet every time before using for the convenience in cleaning after finished using them.

- **Necessary tools to make rubber sheet**
  1. Wires filter number (no.) 40 and 60.
  2. Tray.
  3. Buckets for water and latex.
  4. Table for thresh rubber.
  5. Press machine type slip and stripe.
  6. The building or easily shed.
  7. A can to measures the latex and the water.
  8. Paddle for stir the latex.
  9. Container for mixes acid.

Step 3: Latex filtration

Filter the latex with wire filter no.40 and 60 to get rid the dirtiness by laying the filter overlap on each other. Wire filter no.40 is lay on and no.60 is lay below.

Step 4: Measuring latex

Place the measure latex that has already passed the filter into clean tray (3 liters per tray).

Step 5: Mixing latex and water

Add clean water into the tray that kept 2 liters of latex per tray will get the ratio mixes between the latex and the water in the rate 3 ingredients per 2 ingredients (the ratio mixes might can modify depend on latex quantity).

Step 6: Choosing acid and mixing acid

For rubber has hardened and get rubber sheet high-quality follow the requirement of the buyer, factory or industry should select formic acid intensity 90 %.

- **Advantages of Formic acid are**;

1. Rubber sheet harden regularly if make diluted with the clean water correctly.
2. Can evaporate and not left over in the rubber sheet.
3. Do not make rubber sheet viscous.
4. The property and the flexibility of rubber sheet remain as ever.
5. Do not make the building and rubber sheet stink.
6. Do not damage tool and the equipment so they will be life long for using.

Mixing Formic acid to make rubber sheet harden in 30 – 45 minutes should mix Formic acid in the ratio 30 milliliters (2 tablespoons) mix with the clean water 1,170 milliliters (3 milk cans) or prepare Formic acid in the ratio 0.2 – 0.5 grams mix in the clean water 100 milliliters then stir blend by pour acid in the water and should use the utensils that is the glazed tile or plastic gallon in mixing.

Step 7: Using acid mixes the latex

Use paddle to stir the latex 1 – 2 times then measure acid already mixed 390 milliliters (1 milk can) or using Formic acid 0.2 – 0.6 grams per latex 100 grams pour in the latex throughout a tray while pour acid uses paddle to stir the latex about 6 times (1 Formic bottle can make rubber sheet about 90 – 100 sheets).

Step 8: Bubble latex sweep

Bubble will happen while stir the latex, use paddle sweeps bubble go out all from tray. Pick collect into the utensils keeps to sell is rubber first-class leavings. If do not sweep bubble latex out when take rubber sheet go to smoke will see the air bubble in rubber sheet and make rubber sheet low grade more that should be.

Step 9: Using material cover tray

Should use galvanized iron plate or any material to cover tray for protect dust or dirtiness falls in the latex that is hardening, left the times about 30 – 45 minutes**.**

Step 10: Threshing rubber sheet

When latex harden before take to threshing should pour the clean water moistens to tray for convenient in forcing rubber down out of tray. Threshing rubber ought to thresh rubber on clean table which lay using the aluminium or galvanize iron plate threshing by hand or round wood depend upon skillful. Thresh rubber thickly about 1 centimeter.

Step 11: Threshing rubber sheet with machine slip

Take rubber sheet that thresh already into machine slip about 3 **–** 4 times make thinly about 3 – 4 millimeters.

Step 12: Threshing rubber sheet with machine stripe

After take rubber sheet in machine slip already then take rubber sheet in machine stripe again it will help rubber sheet dry faster when take it to smoked.

Step 13: Washing rubber sheet

Rubber sheet that thresh by machine stripe already should wash with the clean water for washing acid and the dirtiness which stick on the skin of rubber sheet away out.

Step 14: Air dried sheet

Rubber sheet that wash with the clean water already should be hung and let them dry in shady. Should not dry them in a sun because it will make rubber sheet can deteriorate easily. Do not lay rubber sheet on road, dusty places or places that easy to get dirty.

Step 15: Keeping rubber sheet to sell

After air dry rubber sheet about 6 hours we should collecting rubber sheet by hanging on a railing in the building for keeping until sell. If there is smoke building should import to smoke or bake rubber sheet in solar energy building for protects rubber sheet from a fungus and can keep them longer. Owner of the rubber plantation or agriculturist would like to sell rubber sheet at good price, it's very necessary to produce rubber sheet that has high-quality and gathering into a group then sell with a great number of rubber sheets.

- **Character of high quality rubber sheet**

1. Clean rubber sheet have no drying acid mark or get sticky wet. When lift rubber sheet up to check must not have the dirtiness or alkaline black dot mixes in rubber sheet and must not have any dot of bubble.
2. Thin sheet, the thickness of sheet is between 2.8 – 3.2 millimeters. Rubber sheet is a rectangle width about 40 - 45 centimeters and length about 80 – 85 centimeters.
3. Rubber sheet should be wholly dry, a color of rubber sheet is same color through sheet regularly, no black alkaline, no alternate stripes or too dark color.
4. Rubber sheet have a stripe clearly, there is the flexibility, when pull rubber sheet straight out to check it must not be torn easily or have a riddled hole.

Table 2.3 Unsmoked Sheet quality standard

List	Quality			
	1	**2**	**3**	**4**
1. Clean Sheet	100%	100%	100%	100%
Dirty Sheet	0%	Little	Little	Little
Air bubble in sheet	0%	Little	Little	Little
2. The thick size not exceed (millimeter)	3	4	4	4
3. Moistness in rubber sheet not exceed (percent)	1.5	2	3	4.5
4. Color of rubber sheet	Transparent	Regularly	Not clear extremely	Not clear
Darkness	0	Might have	Darkness	Darkness
Black stain	0	Might have	Rather Dark	Dark
5. Sheet flexible	Good	Good	Good	Good
Stripes sheet that appear	Clear	Clear	Clear	Clear

2.1.7.2 Production process of Air Dried Sheet (ADS) or Ribbed Smoked Sheet (RSS)

Production process of Air Dried Sheet (ADS) or Ribbed Smoked Sheet (RSS) show in Figure 2.4, standard of RSS1, RSS2, RSS3, RSS4 and RSS5 such as property of fungus, rubber sheet, acceptable flaw, or unacceptable flaw shown in Table 2.4 – 2.7.

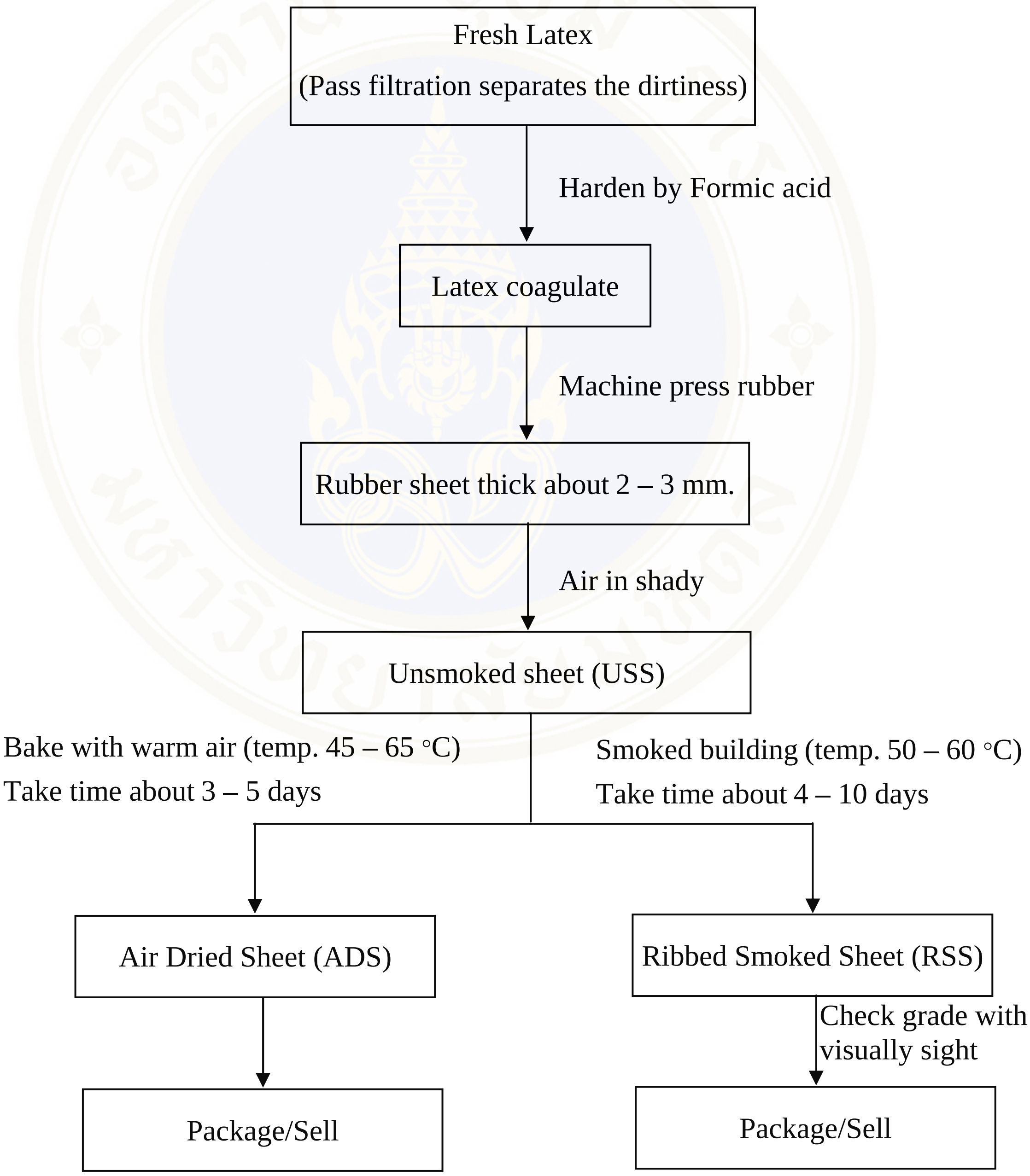


Figure 2.4 Production processes of ADS / RSS

- **Standard of Air Dried Sheet (ADS)**

1. Air Dried Sheet 1 (ADS 1)

The importance character in respectively as follows;

1. Thin rubber sheet, the thick not exceed than 3 millimeters.
2. Clear rubber sheet, gold yellow or light yellow color, there is a smooth color through sheet (valuable of a color is equal to Block Rubber grade 5L), do not have darkness color, spot or black stain on rubber sheet.
3. Air Dried Sheet 100%.
4. Clean rubber sheet, do not have air bubble, do not get wet or blistered and without the adulterated thing.
5. Sheet has good flexibility, clear stripes sheet.
6. In case of compresses to make a block, must have a standardized as follows;

   6.1 Block rubber is a rectangle, the size is 33 centimeters width and 67 centimeters height.

   6.2 Block rubber weight equal to 33.33 kilograms per block.

   6.3 Block rubber must have same color all of the block and have no strange color rubber sheet mixed up.

2. Air Dried Sheet 2 (ADS 2)

The importance character in respectively as follows;

1. Thin rubber sheet, the thick not exceed than 3 millimeters.
2. Clear rubber sheet, gold yellow or light yellow color, there is a smooth color regularly through sheet, allow to have a darkness color, a little spot or black stain on rubber sheet.
3. Air Dried Sheet 100%.
4. Clean rubber sheet, do not have air bubble, do not get wet or blistered and without the adulterated thing.
5. Sheet has good flexibility, clear stripes sheet.
6. In case of compresses to make a block, must have a standardized as follows;

   6.1 Block rubber is a rectangle, the size is 33 centimeters width and 67 centimeters height.

   6.2 Block rubber weight equal to 33.33 kilograms per block.

   6.3 Block rubber must have same color all of the block and have no strange color rubber sheet mixed up.

Table 2.4 Standard of Ribbed Smoked Sheet 1 – 5; Property of Fungus

Grade 1	Grade 2	Grade 3	Grade 4	Grade 5
Must not have a fungus or have dry tiny fungus only at rubber sheet use to wrap.	There is a little of rust fungus or have dry tiny fungus only at rubber sheet use to wrap but not exceed 5% on the sample check.	There is a little of rust fungus or have dry tiny fungus only at rubber sheet use to wrap but not exceed 10% on the sample check.	There is a little of rust fungus or have dry tiny fungus only at rubber sheet use to wrap but not exceed 20% on the sample check.	There is a little of rust fungus or have dry tiny fungus only at rubber sheet use to wrap but not exceed 30% on the sample check.

Table 2.5 Standard of Ribbed Smoked Sheet 1 – 5; Property of Rubber Sheet

Grade 1	Grade 2	Grade 3	Grade 4	Grade 5
Every rubber sheet must have property as follow; - Dried - Harden - No blistered dot - No gravel - No contaminated - No flaw - Clean - No rust fungus	Every rubber sheet must have property as follow; - Dried - Harden - No blistered dot - No gravel - No contaminated - No flaw - Clean	Every rubber sheet must have property as follow; - Dried - Harden - No blistered dot - No gravel - No contaminated	Every rubber sheet must have property as follow; - Dried - Harden - No blistered dot - No gravel - No contaminated	Every rubber sheet must have property as follow; - Dried - Harden - No blistered dot - No gravel - No contaminated

Table 2.6 Standard of Ribbed Smoked Sheet 1 – 5; Property of Acceptable flaw

Grade 1	**Grade 2**	**Grade 3**	**Grade 4**	**Grade 5**
- There is air bubble which size is about top of needle spreading throughout rubber sheet.	- There is little air bubble.	- There is little air bubble.	- There is the air bubble has moderate.	- There is the big air bubble.
- There is black dot of the little bark.	- There is black dot of the little bark.	- There is black dot of the little bark.	- There is the dirtiness has moderate.	- There is big dirtiness.
			- There is black spot color has moderate.	- Darkness color and has black spot moderate.
			- A little sticky.	- Increase a little sticky.
			- A little more smoked.	- Increase a little more smoked.
			- Good dry and have not contaminated thing.	- A little bit smoked.

Table 2.7 Standard of Ribbed Smoked Sheet 1 – 5; Property of Unacceptable flaw

Grade 1	Grade 2	Grade 3	Grade 4	Grade 5
- Rubber sticky and wet - Rubber has less latex - Rubber is overdone - Rubber is burnt - Rubber has less smoked - Rubber has more smoked. - Rubber opaque	- Rubber sticky and wet - Rubber has less latex - Rubber is overdone - Rubber is burnt - Rubber has less smoked - Rubber has more smoked. - Rubber opaque	- Rubber sticky and wet - Rubber has less latex - Rubber is overdone - Rubber is burnt - Rubber has less smoked - Rubber has more smoked. - Rubber opaque	- Rubber sticky and wet - Rubber has less latex - Rubber is overdone - Rubber is burnt - Rubber has less smoked	- Rubber sticky and wet - Rubber has less latex - Rubber is overdone - Rubber is burnt

2.2 Rubber Problems

Many people might still don't know that Thailand is the 1st ranking natural rubber exporter in the World. There is a lot of production latency about 2 million tons per year (33 percentages of worldwide producers). Thailand exporting market shares in the world high value about 40 percentages. Calculate of exporting cost in a type of the raw material and rubber products about 100,000 million baht yearly. Abandon rival such as Indonesia has exported 28 percentages and 3rd rank such as Malaysia has exported 15 percentages. Thailand is one in not many countries in the world that can produce rubber sheets exporting to world market. Regard as economical strength of Thailand [4].

In addition to the strength of the quantity of rubber tree produce that has exported to the worldwide already. Growing rubber tree is still a main occupation in all part of Thailand. Especially in the South of Thailand the rubber tree can be called culture plant that integrates and harmonious suit becomes to be the symbol in the southern. Now more than 6 million people in Thailand have the way of life pertaining to the rubber tree on cultivated area more than 20,000 million square meters. Grow most in the southern about 16,800 million square meters. In eastern about 2,400 million square meters and grows some in northeast of Thailand.

Not only that, the rubber tree still the plant which is worthy for the way of life in Thai, but also, it can create a job and bring the income to more than 70,000 Thai people. Because nowadays in Thailand, there have more than 237 factories to processes the rubber latex and a factory that produces rubber products more than 710 factories. This industry can help to spread the income to the economy at root grass level of Thailand.

Although Thailand is the leader exporter rubber in the world but on this strength of export it is still a weakness hiding which is the problem issue that cause impact to the agriculturists and Thai rubber producers which they can not avoid.

2.2.1 Preliminary rubber process problem.

1. The structure of the rubber process production is limited.
2. The quality of rubber product such as Ribbed Smoked Sheet (RSS), Concentrated latex still lack of regularity.
3. Lack of labor and wage rate trend is growing up.
4. Lack of raw material that is appropriate for applying in Rubber Block (RB) production then make production capital more than rival country.
5. The agriculturist lack of knowledge in business manner and in-house marketing information.
6. The agriculturists have no strong organization then they do not have power to negotiate the selling prices.
7. Lack of facilities in the exporting such as harbor which make high cost in the exporting from Thailand.
8. Exporting market of Thailand group is still limit in some area.

2.2.2 Rubber production problem

1. Medium and small factories have many limitations of the technology.
2. Lack of research and development technology.
3. Improvement procedure products have many limitations.
4. Products quality must pass standard checking especially test service standard of the state agency which is still not enough and are defective according to the buyer.
5. Medium and small factories have limitations of foreign countries market and in-house market not big enough to support products in Thailand.
6. Lack of the latency in the arrangement touts for order or business to export.
7. The tax rate of imports chemical and synthesis rubber that use in rubber production are high rate.

Preliminary problems of rubber manufactures in the Thailand. Thai rubber producer could not stop thinking to develop production and learn strategy from competition.

Although preliminary problem is the important variable in Thailand but economic of the Thai condition and world economy will be important variable that affect exporting rubber market of Thailand. Because of 70 percentages use in tire industry and the remainder will use in many production industry such as car equipment, material and medical equipment, communication, construction etc. These rubber production demands more or less are depend on world economy. Although we can not avoid that problem but Thailand can solve that problem by create new innovation in production procedure.

Because of economy downturn since 1997 caused United State of America, Japan and countries in Europe consuming rubber was decrease. Except the People's Republic of China which the economy still growing up 7 percentages per year. According to the national economic policy to promote their own market instead of importing products from other countries.

The economy of Chinese still expands continuously until now. So it make Chinese step up to the 1st ranking rubber consumer in the world replace superpower country like the United States of America which in the past was always be the first. From the evidence of using rubber quantity in the People's Republic of China, the United States of America and Japan in 2001 which are 17, 14 and 11 percentage respectively.

Thailand becomes rubber world major exporter. There is the crucial necessity to leader in the innovation for keeping former market share and expands new rubber market increase especially the market in the People's Republic of China which this is a trend that there will have rubber requirement increases.

2.3 Visually Graded Rubber

Currently, in the rubber factories employ experts or scientists to analyse the grade of Ribbed Smoked Sheet (RSS) with visual sight and experience that have description as follow;

2.3.1 Visually sight from experts or scientists

Grading quality of Ribbed smoked sheets (RSS) in rubber marketed are based on visual assessment. Grading is carried out by visual inspection according to the standards specified in the Green Book published by the Rubber Manufacturers Association Inc, Washington. Some cases the customer could require conduct tests for any specific parameters.

Experts or scientists will consider rubber sheet from a visual sight. They will determine and estimate the grade of rubber sheet by using experience and knowledge of rubber properties. They will separate RSS to RSS1, RSS2, RSS3, RSS4 and RSS5 depending on the properties of each rubber sheet. Each of RSS standards has shown in Table 2.4 – Table 2.7.

2.4 Digital Image Processing

Import images to process, evaluate or adjust to show the thing that we interested. There is the brief principle as follow; **Image Acquisition**: It is the first step of processes which in this step will import images to computer. Next step is **Segmentation:** In this part will segment image that we interest from background**.** Next step is **Clustering:** At this stage we will separate groups of objects with nearby colors and calculate the variables that we need from these images.

2.4.1. Image Acquisition

Image Acquisition is the step that loads an image into computer through any equipment such as digital camera or scanner. When equipment already get images through sensor then represent image into digital image and send data to collect in the computer.

2.4.2. Segmentation

Segmentation is the step of separate interested image out of the background. This step will find coordinates and cut edges off to get interested area.

2.4.3. Clustering

Clustering is a way to separate groups of objects. K-means clustering treats each object as having a location in space. It finds partitions such that objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible.

2.4.4 Calculation of RSS grades

This process wills calculate minimum, maximum and mean RGB (red, green, blue) color space data in each cluster images which will be a core in system. It will be used to analyse interesting RSS images and calculation of RSS grades.

2.4.5 Display of results

This process will display of calculation result of Ribbed Smoked Sheet (RSS) grades by evaluate mean RGB (red, green, blue) information from the most cluster image's area.

2.5 Related Works

2.5.1 Convert RGB images to Grayscale images

To convert any color to its most approximate level of gray, use the standard NTSC conversion formula to calculating the effective luminance of a pixel as follow.

Grayscale = (0.2989 x **R**ed) + (0.5870 x **G**reen) + (0.1140 x **B**lue)

First must obtain the values of its RGB (red, green, blue) primaries. Then, add 30% of the Red value, 59% of the Green value, and 11% of the Blue value, together. Regardless of the scale employed (0.0 to 1.0, 0 to 255, 0% to 100%, etc.), the resultant number is the desired gray value, such that a new RGB color would have red, green, and blue values equal to the new number. These percentages are chosen due to the different relative sensitivity of the normal human eye to each of the primary colors.

This is the method used to obtain the luminance in the YUV and related color models, used in standard color TV and video systems as PAL and NTSC, as well as in the L*a*b* color model.

2.5.2 Grayscale images

Grayscale images contain brightness information only, no color information. The number of bits used for each pixel determines the number of different brightness levels available. The typical image contains 8-bit per pixel data, which allows us to have 256 (0-255) different brightness (gray) levels [5]. Displayed images of this format composed of shades of gray, vary from black at the weakest intensity to white at the strongest.

Grayscale images are distinct from black-and-white images, which in the context of computer imaging are images with only two colors, black and white; grayscale images have many shades of gray in between. In most contexts other than digital imaging, the term "black and white" is used in place of "grayscale"; for example, photograph which has shades of gray is typically called "black-and-white photograph".

Grayscale images are often the result of measuring the intensity of light at each pixel in a single band of the electromagnetic spectrum (e.g. visible light). The accuracy provided by this format is barely sufficient to avoid visible banding artifacts, but very convenient for programming. In certain application, such as medical imaging or astronomy, 12 or 16-bit per pixel representations are used. These extra brightness levels become useful only when the image is “blown-up” that is, a small section of image is enlarged. In this case we may be able to discern details that might be missing without this additional brightness resolution [5].

(a) (b) (c)

Figure 2.5 The same picture in three different colors mode

In Figure 2.5 we see the same picture in three different colors mode. Figure 2.5a is the original photograph, in full color. Figure 2.5b is in grayscale; all the colors contained in it are black, white, or any shade of gray. And in Figure 2.5c we have the results of black and white, or monochrome; the colors used in the picture are black and white, with no intermediary grays.

2.5.3 Binary Images

Binary images are the simplest type of images, and can take on two values, typically black and white, or "0" and "1". A binary image is referred to as a 1-bit per pixel image, because it takes only 1 binary digit to represent each pixel. These types of images are most frequently used in computer applications.

Binary images are often created from grayscale images via a threshold operation, where every pixel above the threshold value is turned white ("1"), and those below it are turned black ("0"). Although in this process much information is lost, the resulting image file is much smaller making it easier to store and transmit [5].

2.5.4 Thresholding

Thresholding is the simplest method for segmentation processing [6]. Normally refers to setting all the grey levels below a certain level to zero; or above a certain level to a maximum brightness level. The maximum brightness will be 255 on an 8-bit plane system, 15 on a 4-bit plane, 1023 on a 10-bit plane, and so on [7].

Individual pixels in a grayscale image are marked as “object” pixels if their value is greater than some threshold value (assuming an object to be brighter than the background) and as “background” pixels otherwise. Typically, an object pixel is given a value of “1” while a background pixel is given a value of “0”.

It will be much easier to find an acceptable threshold value with proper lighting and good contrast between the object and the background. This will screen out unwanted variation in an image where all those variations are around, above, or below a certain gray level. If objects do not touch each other, and if their gray levels are clearly distinct from background gray-levels, thresholding is a suitable segmentation method [8-9].

The key parameter in thresholding is obviously the choice of the threshold. Several different methods for choosing a threshold exist. The simplest method would be to choose the mean or median value, the rationale being that if the object pixels are brighter than the background, they should also be brighter than the average.

In a noiseless image with uniform background and object values, the mean or median will work beautifully as the threshold. A more sophisticated approach might be to create a histogram of the image pixel intensities and use the valley point as the threshold [10-11]. The histogram approach assumes that there is some average value for the background and object pixels, but that the actual pixel values have some variation around these average values. However, computationally this is not as simple as we would like, and many image histograms do not have clearly defined valley points. Ideally we are looking for a method for choosing the threshold which is simple, does not require too much prior knowledge of the image, and works well for noisy images. A good such approach is an iterative method, as follows:

An initial threshold (T) is chosen, this can be done randomly or according to any other method desired.

1. The image is segmented into object and background pixels as described above, creating two sets:
    1. $G_1 = \{f(m,n):f(m,n)>T\}$ (object pixels)
    2. $G_2 = \{f(m,n):f(m,n)<=T\}$ (background pixels) (note, $f(m,n)$ is the value of the pixel located in the m^{th} column, n^{th} row)

2. The average of each set is computed.
    1. m_1 = average value of G_1
    2. m_2 = average value of G_2

3. A new threshold is created that is the average of m_1 and m_2
    1. $T' = (m_1 + m_2)/2$

4. Go back to step two, now using the new threshold computed in step four, keep repeating until the new threshold matches the one before it (i.e. until convergence has been reached).

This iterative algorithm is a special one-dimensional case of the K-means clustering algorithm, which has been proven to converge at a local minimum-meaning that a different initial threshold may result in a different final result [12-13].

2.5.4.1 Otsu's method

In computer vision and image processing, Otsu's method is used to automatically perform histogram shape-based image thresholding [13], or, the reduction of a gray level image to a binary image.

The algorithm assumes that the image to be thresholded contains two classes of pixels (e.g. foreground and background) then calculates the optimum threshold separating those two classes so that their combined spread (intra-class variance) is minimal [11]. The extension of the original method to multi-level thresholding is referred to as the Multi Otsu method [14].

In Otsu's method, the threshold that minimizes the intra-class variance, defined as a weighted sum of variances of the two classes:

$$\sigma^2_{\omega}(t) = \omega_1(t)\,\sigma^2_1(t) + \omega_2(t)\,\sigma^2_2\ t)$$

Weights ωi are the probabilities of the two classes separated by a threshold t and σ^2_i variances of these classes.

Otsu shows that minimizing the intra-class variance is the same as maximizing inter-class variance [11];

$$\sigma_b^2(t) = \sigma^2 - \sigma_\omega^2(t) = \omega_1(t)\,\omega_2(t)\,[\mu_1(t) - \mu_2(t)]^2$$

which is expressed in terms of class probabilities ωi and class means μi which in turn can be updated iteratively. This idea yields an effective algorithm.

Algorithm

1. Compute histogram and probabilities of each intensity level
2. Set up initial $\omega i(0)$ and $\mu i(0)$
3. Step through all possible thresholds t = 1 maximum intensity
    3.1 Update ωi and μi
    3.2 Compute $\sigma_b^2(t)$
4. Desired threshold corresponds to the maximum $\sigma_b^2(t)$

2.5.5 Color space selection

Appropriate color space must be selected because of the performance of segmentation methods based on clustering is directly related to the shape and the center position of each cluster [15]. Generally, the color difference is evaluated using the distance between two color points in a color space. The most common distance is Euclidean distance. You can visually distinguish colors from one another by ignore variations in brightness [16].

The L*a*b* color space (also known as CIELAB or CIE L*a*b*) enables you to quantify these visual differences. So, we use CIELAB color space, which is a uniform chromaticity color space. It is known that Euclidean distance of two colors is proportional to the difference that human visual system perceived in the CIELAB color space.

The L*a*b* color space is derived from the CIEXYZ tristimulus values. The L*a*b* space consists of a luminosity layer 'L*', a chromaticity-layer 'a*' that indicates where the color falls along the red-green axis, and a chromaticity-layer 'b*' that indicates where the color falls along the blue-yellow axis. All of the color information is in the 'a*' and 'b*' layers. You can measure the difference between two colors using the Euclidean distance metric.

Actually, the image is composed of RGB color components. So we must convert RGB color components to CIELAB color component. To do this, we first convert RGB to CIEXYZ using following equations [15].

$$X = (0.490\times R) + (0.310\times G) + (0.200\times B)$$
$$Y = (0.177\times R) + (0.813\times G) + (0.011\times B)$$
$$Z = (0.000\times R) + (0.010\times G) + (0.990\times B)$$

The conversion to CIEXYZ color to CIELAB color is performed according to the following relation

$$L = 25 \times (100\times Y/Y_o)^{1/3} - 16$$
$$a = 500 \times [(X/X_o)^{1/3} - (Y/Y_o)^{1/3}]$$
$$b = 200 \times [(Y/Y_o)^{1/3} - (Z/Z_o)^{1/3}]$$

2.5.6 K-means algorithm

The K-means algorithm is methodology use to find automatically the best groupings and means of K clusters [7]. It is similar to the expectation-maximization algorithm for mixtures of Gaussians in that they both attempt to find the centers of natural clusters in the data [17-19]. It assumes that the object attributes form a vector space. The objective it tries to achieve is to minimize total intra-cluster variance, or, the squared error function.

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2$$

where there are k clusters S_i, and μ_i is the centroid or mean point of all the points $x_j \in S_i$.

2.5.7 K-means clustering

The K-means algorithm assigns each point to the cluster whose center (also called centroid) is nearest. The center is the average of all the points in the cluster. That is, its coordinates are the arithmetic mean for each dimension separately over all the points in the cluster.

Example: The data set has three dimensions and the cluster has two points;

$X = (x_1, x_2, x_3)$ and $Y = (y_1, y_2, y_3)$. Then the centroid Z becomes $Z = (z_1, z_2, z_3)$, where $z_1 = (x_1 + y_1)/2$ and $z_2 = (x_2 + y_2)/2$ and $z_3 = (x_3 + y_3)/2$.

The algorithm steps are [20]:

- Choose the number of clusters, k.
- Randomly generate k clusters and determine the cluster centers, or directly generate k random points as cluster centers.
- Assign each point to the nearest cluster center.
- Recomputed the new cluster centers.
- Repeat the two previous steps until some convergence criterion is met (usually that the assignment hasn't changed).

The main advantages of this algorithm are its simplicity and speed which allows it to run on large datasets. It disadvantage is that it does not yield the same result with each run, since the resulting clusters depend on the initial random assignments. It minimizes intra-cluster variance, but does not ensure that the result has a global minimum of variance [21-24].

2.6 Other Related Works

In previous work about Rubber Sheet there does not have work about grades analysis. But it has work about impurity detection and elimination from Ribbed Smoked Sheet (RSS) [25], or inspection of printed for rubber sheet by Image Processing [26]. The similarity work such as classifying a young aromatic coconuts flesh from the external [16], or automatically separating oil glands from pummelo fruit surface [27-28].

The discussions about related work carry on many researchers. Systems were generally composed of several main processing steps. These include inspection of rubber sheet or fruits, color image segmentation, and clustering interested objects. Many techniques has been proposed and applied in each step. Most research represents results from these steps separately so it is reasonable to discuss each step on its own.

2.6.1 Inspection of Rubber Sheet or fruits

Inspection of rubber sheet or fruits is the main important process for agricultural product manufacturing, many techniques based on gray scale or binary images, but some works tested directly on color images.

Leeraphan and Leeraphan's work [25], use the color images value pixel in each partition of images to find out Histogram. Then use threshold to detection impurities out of background and removing.

Santi and Thanate [26] use compared number of an ideal sample to determine printed of rubber sheet if the line number is acceptable quality. The experimental results indicate that method is more than 50% correct when compared with a human expert who inspects the printed rubber sheets. However, if the rubber sheet has regularly printed but it has more impurities spread all around the surface such as Unsmoked Sheet (USS) grade 5 the result will report that it is irregularly because of printed line number is too much than specify condition.

Naratorn and Thanate [16] measuring the quantities of color from the external covering of young aromatic coconut flesh. Every image pixels represent in each photo were analyzed by L*a*b* mode comparing with the colors of the whole covering surface. The result indicated that the quantities of the colors were respectively positive correlating with the ages of the fruits.

Krisada and Koarakot [27] present a program development for automatically separating oil glands from pomelo fruit surface for maturation analysis of Kao Nampheung pomelo by using size of oil glands. The comparison of the average difference of the oil glands sizes between this program and the old program from Noppadol [28] with each image, each pomelo and each gland are 8.44 %, 2.68 % and 0.69 %, respectively from 165 images.

2.6.2 Color Image Segmentation

Thresholding is a popular method for the segmentation step used in most work explored so far. However, a comparative work on different approaches was conducted by Naratorn and Thanate [16] proposed color image segmentation in CIE L*a*b* color space because of this color mode similarity to visually sight from human and using K-means clustering algorithm to separate 3 groups of image color pixel. Color image segmentation performed an important role in the image processing and computer vision. Many algorithms have been proposed to segment the color image [15]. They segment the image using coherency of color value or spatial distribution of color. Clustering and histogram based techniques are the method that divide the specific feature space derived from a color image

Santi and Thanate [26] proposed color image segmentation technique by modified image 256 gray level by weighting image pixel to reduce noise and impurity that spread all around rubber sheet and use morphological operations to illumination background. Then minus illumination image with weighting image will get darkness image result. After that contrast the darkness result with thresholding technique and transform image result to binary image.

Krisada, Noppadol and Koarakot [27-28] divided a whole image to 16 subimage and transformed color image to binary image using adaptive thresholding technique. The object and background are tracked using clustering technique.

Kuk-Jin Yoon and In-So Kweon [15] propose a novel color image segmentation algorithm to divide the specific feature space and consideration of human visual sensitivity for color pattern variations by generalizing K-means clustering. They define the CCM (Color Complexity Measure) by calculating the absolute deviation with Gaussian weighting within the local mask and assign weight value to each color vector using CCM value.

2.6.3 Clustering

Clustering is the assignment of objects into groups (called *clusters*) so that objects from the same cluster are more similar to each other than objects from different clusters. Often similarity is assessed according to a distance measure. Clustering is a common technique for statistical data analysis, which is used in many fields, including machine vision, machine learning, data mining, pattern recognition, image analysis and bioinformatics.

The most common clustering technique is the K-means algorithm. It is known as a powerful method to deal with the large color pixel set to get the optimal clustering. In this scheme, each pixel is mapped to the one point in the feature space according to its feature (typically color). The feature points that having similar feature are grouping into the same cluster. Then, the each feature point has the cluster index and it is inversely mapped to the image space.

Kuk-Jin Yoon and In-So Kweon [15] use K-means clustering to weighted color vectors and the shape and the center position of each cluster which is formed according to the color distribution in the image. They adaptively determine optimal K value, which is the number of cluster, by using the statistics of the color complexity measure that implies the complexity of color image.

Naratorn and Thanate [16] process the result from color image segmentation in CIE L*a*b* color space using K-means clustering algorithm to separate 3 groups of image color pixel. Find an average (mean) of the center clusters and sorting the clusters. Then selection cluster interest and convert color image to binary image. Plot the circle on the center of image which has radius one - quarter of image. Delete noises and uninterested area out of circle to inspection interested pixels in the circle area. After that counting bit value equal to 1 in circle area and classification stages of young aromatic coconuts flesh.

Krisada and Koarakot [27] using clustering technique to tracked the object and background. Then define K cluster equal to 2 because of it appropriate to separate image that have only object and background without receive external value from the user [29].

CHAPTER III
METHODOLOGY

This chapter explains about overall architecture, system overview, structure chart, clustering technique, and processes of Analysis Graded of Ribbed Smoked Sheet (AGRSS) system that shows functions, variables and definition using in MATLAB programming. This research uses K-means algorithm to cluster Ribbed Smoked Sheet (RSS) images and calculate RSS grades.

3.1 Overall Architecture

Analysis Grade of Ribbed Smoked Sheet (AGRSS) system is based on color image processing and clustering. The following diagram provides an overview of the system architecture:

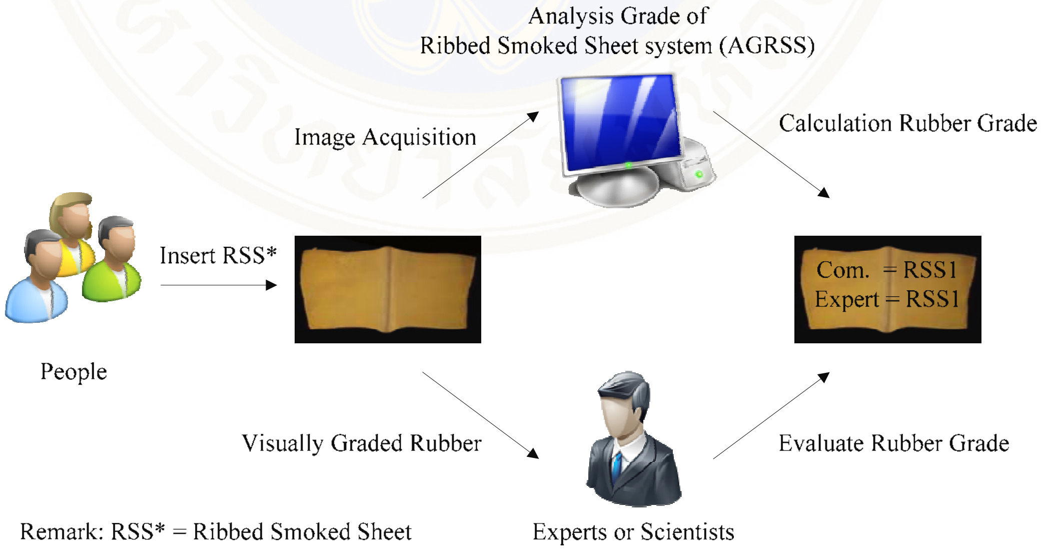


Figure 3.1 The Physical Diagram for AGRSS system

Physical people are represented as agriculturists, dealers or middlemen. They insert Ribbed Smoked Sheet (RSS) into system. The system was separated to two sections; the first one is Experts or Scientists, and the other one is AGRSS system.

Experts or Scientists is represented the person who visually graded rubber by employing his knowledge or experience then evaluate RSS grade. AGRSS show that the system will acquire image into system and calculation of rubber grades from RSS range training conditions. Both of them will report RSS grade at the end of process.

3.2 System Overview

As shown in Figure 3.2, the system training phase consists of 4 main processes, including image acquisition, segmentation, clustering, and create RSS (Ribbed Smoked Sheet) range from mean RGB (red, green and blue) color space data. The testing phase contain with 4 main processes. They are image acquisition, segmentation, clustering and calculation of RSS (Ribbed Smoked Sheet) grades. An overview of system operation will show in Figure 3.2 and explain details in next step.

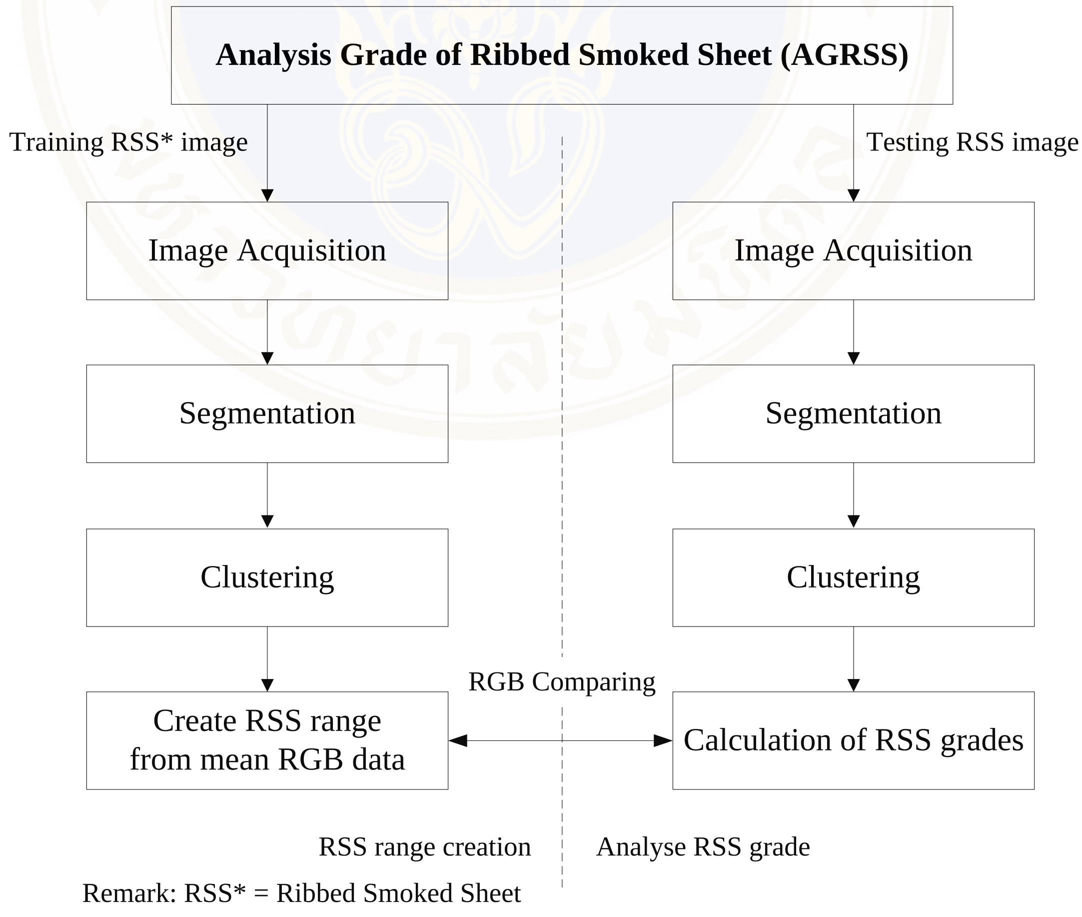


Figure 3.2 The System Diagram for AGRSS system

The input to the system is a top view of Ribbed Smoked Sheet (RSS) image. RSS image must cover rubber sheet up at least 80 % of the rubber image area and background area may be included. The image is 3701 x 2304 pixels and resize to 640 x 480 pixels in dimension with 24-bit color, encoded in JPEG file format. RSS image will be converted to a 256 gray scale, binary and other color space such as L*a*b* color space (The L*a*b* color space consists of a luminosity layer 'L*', a chromaticity-layer 'a*' that indicates where the color falls along the red-green axis, and a chromaticity-layer 'b*' that indicates where the color falls along the blue-yellow axis.) before further processing. The system will work as described in the following:

1. The user insert RSS image with dark background into Analysis Grade of Ribbed Smoked Sheet (AGRSS) system.
2. Convert RGB image to grayscale then calculate threshold value to separate RSS area and background for produce binary image.
3. Remove noise in binary image and find coordinates of RSS then crop coordinates fit RSS area.
4. Convert RGB image to L*a*b* color space and classify color in a*b* space using K-means clustering.
5. Label every pixel in the image using the results from K-means clustering and create RSS image segmentation by color.
6. Clustering will represent two clusters images and show objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible.
7. Calculate minimum, maximum and mean RGB information from the most clustering area and next cluster area to classify range RGB conditions for each RSS grades in training phase and use to comparing RGB data in testing phase.
8. Display of results from calculation of RSS grades.

3.3 Structure Chart

This section displays the structure chart of Analysis Grade of Ribbed Smoked Sheet (AGRSS) system, consisting of difference processes as illustrated in Figure 3.3 and describes detail in section 3.4**.**

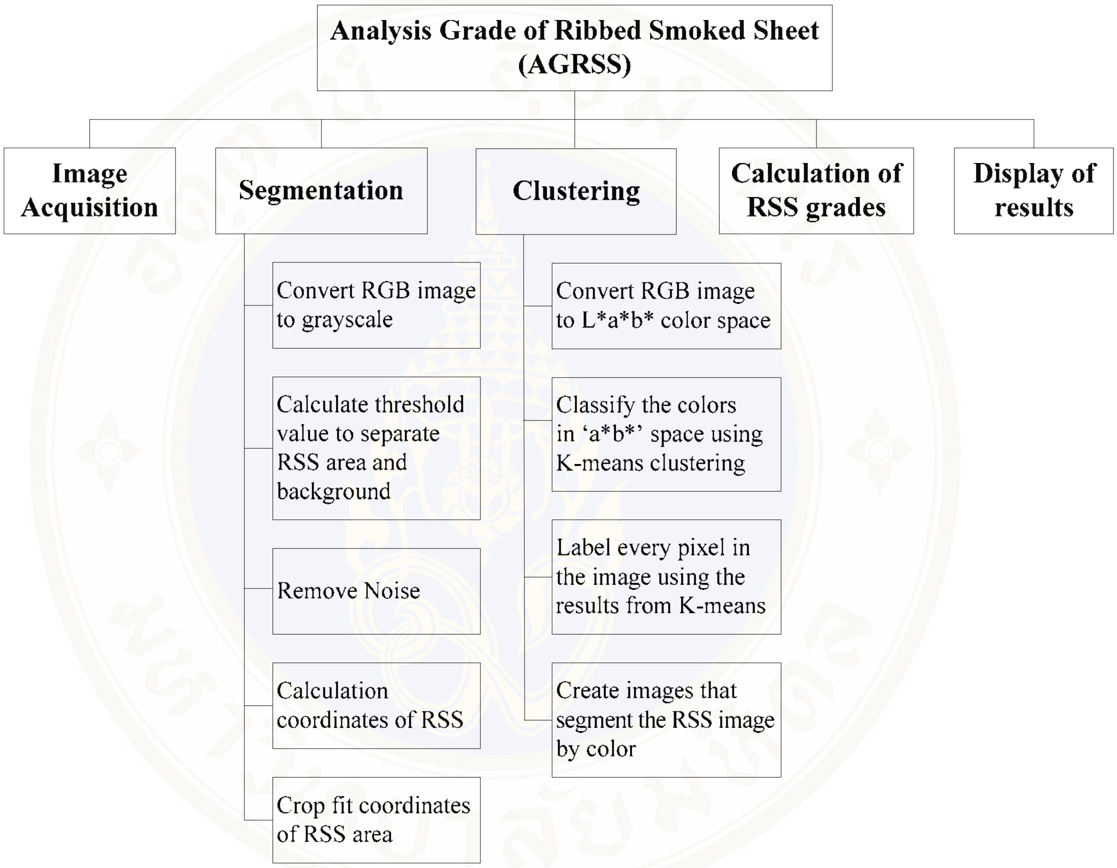


Figure 3.3 The structure chart for AGRSS system

3.4 Analysis Graded of Ribbed Smoked Sheet (AGRSS) system

Operations in this phase are very important in order to develop an Analysis Graded of Ribbed Smoked Sheet (AGRSS) system. This section will show functions, variables and definition using in MATLAB. Ribbed Smoked Sheet (RSS) images will pass through the first three processes to collect information of RSS images. The goal of collecting is to gain more sufficient information of minimum, maximum and mean RGB (red, green, blue) from RSS image clusters to classify RSS grades. Some extracted data images of the same RSS grade from different color may be made some variations. The collected information of more variants RSS image color will gain more complicated to analysis RSS grades. However, collected information of too many training images will cause accuracy to evaluated RSS grade condition.

3.4.1 Image Acquisition

Receive Ribbed Smoked Sheet (RSS) images using dark background and dark environment in a dark box, control white light using white fluorescence on the top of dark box in 2 sides and taken images by digital camera. Function *imread* use in process to read image from graphics file into system. We will define code to acquire image in the following:

I = imread(filename,fmt)

The system will reads image *(I)* from the file specified by the string *filename*, where the string *fmt* specifies the format of the file. For a list of all the possible values for *fmt*, see Table 3.1. Function *imread* will returns the image data in the array *I*. If the file contains a grayscale image, *I* is a two-dimensional (M-by-N) array. If the file contains a color image, *I* is a three-dimensional (M-by-N-by-3) array. The class of the returned array depends on the data type used by the file format. See class support show in Table 3.2 for more information.

3.4.1.1 Supported Formats

Table 3.1 lists all the types of images that *imread* can read, in alphabetical order by the *fmt* abbreviation [30].

Table 3.1 Lists all the types of images that function *imread* can read

Format	Full Name	Variants
'bmp'	Windows Bitmap (BMP)	1-bit, 4-bit, 8-bit, 16-bit, 24-bit, and 32-bit uncompressed images and 4-bit and 8-bit run-length encoded (RLE) images
'cur'	Windows Cursor resources (CUR)	1-bit, 4-bit, and 8-bit uncompressed images
'gif'	Graphics Interchange Format (GIF)	1-bit to 8-bit images
'hdf'	Hierarchical Data Format (HDF)	8-bit raster image data sets, with or without an associated colormap, and 24-bit raster image data sets
'ico'	Windows Icon resources (ICO)	1-bit, 4-bit, and 8-bit uncompressed images
'jpg' or 'jpeg'	Joint Photographic Experts Group (JPEG)	Any baseline JPEG image or JPEG image with some commonly used extensions, including: **Image Type** / **Bitdepth** / **Compression** grayscale / 8- or 12-bit / lossy grayscale / 8-, 12-, or 16-bit / lossless RGB / 24- and 36-bit / lossy or lossless
'pbm'	Portable Bitmap (PBM)	1-bit images using either raw (binary) or ASCII (plain) encoding

Table 3.1 Lists all the types of images that function *imread* can read (Continued)

Format	Full Name	Variants
'pcx'	Windows Paintbrush (PCX)	1-bit, 8-bit, and 24-bit images
'pgm'	Portable Graymap (PGM)	ASCII (plain) encoding with arbitrary color depth, or raw (binary) encoding with up to 16 bits per gray value
'png'	Portable Network Graphics (PNG)	1-bit, 2-bit, 4-bit, 8-bit, and 16-bit grayscale images; 8-bit and 16-bit indexed images; and 24-bit and 48-bit RGB images
'pnm'	Portable Anymap (PNM)	PNM is not a file format itself. It is a common name for any of the other three members of the Portable Bitmap family of image formats: Portable Bitmap (PBM), Portable Graymap (PGM) and Portable Pixel Map (PPM).
'ppm'	Portable Pixmap (PPM)	ASCII (plain) encoding with arbitrary color depth or raw (binary) encoding with up to 16 bits per color component
'ras'	Sun Raster (RAS)	1-bit bitmap, 8-bit indexed, 24-bit truecolor and 32-bit truecolor with alpha data
'tif' or 'tiff'	Tagged Image File Format (TIFF)	Any baseline image, including 1-bit, 8-bit, and 24-bit uncompressed images; 1-bit, 8-bit, and 24-bit images with packbits compression; 1-bit images with CCITT compression; and 16-bit grayscale, 16-bit indexed, and 48-bit RGB images
'xwd'	X Windows Dump (XWD)	1-bit and 8-bit ZPixmaps, XYBitmaps, and 1-bit XYPixmaps

3.4.1.2 Class Supported

For most file formats, function *imread* uses 8 or fewer bits per color plane to store pixels. The following Table 3.2 lists the class of the returned array for all data types used by the file formats [30].

Table 3.2 Lists the class of the returned array for all data types used by the file formats

Data Type Used in File	Class of Array Returned by function *imread*
1-bit	logical
8-bits (or fewer) per color plane	uint8
12-bits	uint16
16-bits (JPEG, PNG, and TIFF)	uint16
16-bits (BMP only)	uint8

Figure 3.4 Example of Ribbed Smoked Sheet image

3.4.2 Segmentation

This process will define position of RSS (Ribbed Smoked Sheet) area, calculate RSS coordination and crop raw image to get image fit RSS area.

3.4.2.1 Convert RGB images to grayscale

In this process we will convert RGB (red, green, blue) images to grayscale. After this process the system will calculate RSS area by count white pixels which it is RSS and separate black pixels that it is background of RSS.

Figure 3.5 Example of Ribbed Smoked Sheet in grayscale

This work use MATLAB function *rgb2gray* converts RGB images to grayscale by eliminating the hue and saturation information while retaining the luminance.

I = rgb2gray(RGB) converts the truecolor image RGB to the grayscale intensity image *I*.

newmap = rgb2gray(map) returns a grayscale colormap equivalent to map.

- **Class Supported**

If the input is an RGB image, it can be of class uint8, uint16, single, or double. The output image *I* is of the same class as the input image. If the input is a colormap, the input and output colormaps are both of class double.

- **Algorithm**

Function *rgb2gray* converts the RGB values to NTSC coordinates, sets the hue and saturation components to zero, and then converts back to RGB color space.

3.4.2.2 Calculate threshold value to separate RSS area and background

Compute global image threshold using Otsu's method [11]. Produce binary images from RGB images. Use threshold to convert grayscale image to binary. Choose the threshold to minimize the intraclass variance of the black and white pixels. The output binary image has values of 1 (white) for all pixels in the input image with luminance greater than level and 0 (black) for all other pixels.

Figure 3.6 Example of binary Ribbed Smoked Sheet from Otsu's Method

In this work use MATLAB function *graythresh* computes a global threshold (level) that can be used to convert an intensity image to a binary image with function *im2bw*. The variable *level* is a normalized intensity value that lies in the range [0, 1].

level = graythresh(I)

Multidimensional arrays are converted automatically to 2-D arrays using reshape and ignore any nonzero imaginary part of image *I*.

3.4.2.3 Remove Noise

In this process will remove Noise from binary RSS image. First remove a binary image all connected components (objects) that have fewer than 2 pixels. Then join the small black point in the image together by filling in the gaps between them and by smooth their outer edges and producing another binary image.

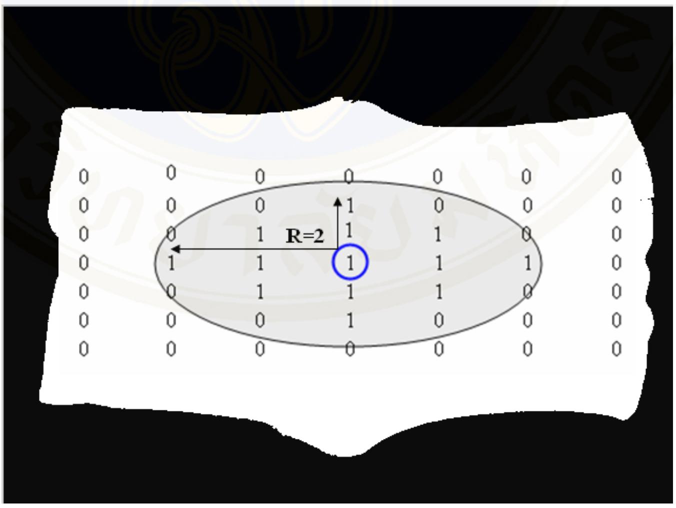


Figure 3.7 Morphological operations specify a radius of 2 pixels

Create a disk-shaped structuring element. Use a disk structuring element to preserve the circular nature of the object.

SE = strel('disk',R,N) creates a flat, disk-shaped structuring element, where *R* specifies the radius. *R* must be a nonnegative integer. *N* must be *0, 4, 6, or 8*. When *N* is greater than *0*, the disk-shaped structuring element is approximated by a sequence of *N* periodic-line structuring elements. When *N* equals *0*, no approximation is used, and the structuring element members consist of all pixels whose centers are not greater than *R* away from the origin. If *N* is not specified, the default value is *4*. In this work specify a radius of 2 pixels so that the largest gap gets filled follow this source code.

```
SE   =  strel('disk',2);
IM2  =  imclose(IM,SE);
BW2 =  imfill(BW,'holes';)
```

Morphological operations [26, 30] run much faster when the structuring element uses approximations (N > 0) than when it does not (N = 0). However, structuring elements that do not use approximations (N = 0) are not suitable for computing granulometries. Sometimes it is necessary for *strel* to use two extra line structuring elements in the approximation, in which case the number of decomposed structuring elements used is N + 2.

Figure 3.8 Picture of RSS after remove Noise

IM2 = imclose(IM,SE) performs morphological closing on the grayscale or binary image *IM*, returning the closed image, *IM2*. The structuring element, *SE*, must be a single structuring element object, as opposed to an array of objects.

BW2 = imfill(BW,'holes') fill any holes in the binary image *BW*. A hole is a set of background pixels that cannot be reached by filling in the background from the edge of the image.

3.4.2.4 Calculation coordinates of RSS

We will trace the exterior boundaries of RSS (Ribbed Smoked Sheet), as well as boundaries of holes inside these objects, in the binary image (BW). Function *bwboundaries* also descends into the outermost objects (parents) and traces their children (objects completely enclosed by the parents). BW must be a binary image where nonzero pixels belong to an object and 0 pixels constitute the background. Figure 3.10 will illustrates these components.

Figure 3.9 Find coordinates of RSS image

Function *bwboundaries* returns *B*, a *P-by-1* cell array, where *P* is the number of objects and holes. Each cell in the cell array contains a *Q-by-2* matrix. Each row in the matrix contains the row and column coordinates of a boundary pixel. *Q* is the number of boundary pixels for the corresponding region.

Figure 3.10 Trace region boundaries in a binary image [30]

[B,L,N] = bwboundaries(BW) returns the label matrix *L* as the second output argument. Objects and holes are labeled. Variable *L* is a two-dimensional array of nonnegative integers that represent contiguous regions. The k_{th} region includes all elements in *L* that have value *k*. The number of objects and holes represented by *L* is equal to *max(L(:))*. The zero-valued elements of *L* make up the background. Returns *N*, the number of objects found. The first *N* cells in *B* are object boundaries.

- **Class Supported**

BW (binary image) can be logical or numeric and it must be real, two-dimensional, and nonsparse. Variable *L* and *N* are double and *A* is sparse logical.

3.4.2.5 Crop fit coordinates of RSS area

This operation automatically crop RSS (Ribbed Smoked Sheet) area from calculation coordinates of RSS value and this process can get value from manual crop selected RSS area by the users.

Figure 3.11 RSS after crop coordinate**s**

Function *imcrop* will crops an image to a specified rectangle. In the syntaxes below, *imcrop* displays the input image and waits for the user to specify the crop rectangle with the mouse.

RGB2 = imcrop(RGB)

To specify the rectangle, for a single-button mouse, press the mouse button and drag to define the crop rectangle. Finish by releasing the mouse button. For a two or three button mouse, press the left mouse button and drag to define the crop rectangle. Finish by releasing the mouse button.

If the users hold down the Shift key while dragging, or press the right mouse button on a two or three button mouse, function *imcrop* constrains the bounding rectangle to be a square. When the users release the mouse button, function *imcrop* returns the cropped image in a new figure.

In this work has specify the cropping rectangle *(RECT)* automatically, using these syntaxes:

RGB2 = imcrop(RGB, RECT)

Variable *RECT* is a four-element vector with the form [xmin ymin width height]; these values are received from calculation coordinates RSS process.

- **3.4.2.5.1 Class Supported**

If you specify *RECT* as an input argument, the input image can be logical or numeric, and must be real and nonsparse. *RECT* is of class double.

If you do not specify *RECT* as an input argument, function *imcrop* calls function *imshow*. Function *imshow* expects *I* (image) to be logical, uint8, uint16, int16, single, or double. RGB (red, green, blue) color space can be uint8, int16, uint16, single, or double. Variable *X* can be logical, uint8, uint16, single, or double. The input image must be real and nonsparse. If you specify an image as an input argument, the output image has the same class as the input image.

If you don't specify an image as an input argument, i.e., you call function *imcrop* with no input arguments, the output image has the same class as the input image except for the int16 or single. The output image is double if the input image is int16 or single.

To specify a nondefault spatial coordinate system for the input image, precede the other input arguments with two, two-element vectors specifying the XData and YData. For example:

[...] = imcrop(x,y,...)

3.4.3 Clustering

Clustering is a way to separate groups of objects. The goal is to segment colors in an automated fashion using the L*a*b* color space and K-means clustering. K-means clustering treats each object as having a location in space. It finds partitions such that objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible.

K-means clustering requires that you specify the number of clusters to be partitioned and a distance metric to quantify how close two objects are to each other.

Since the color information exists in the 'a*b*' space, your objects are pixels with 'a*' and 'b*' values. Use function *kmeans* to cluster the objects into three clusters using the Euclidean distance metric.

3.4.3.1 Convert RGB image to L*a*b* color space

The L*a*b* color space (also known as CIELAB or CIE L*a*b*) enables to quantify visual differences. The L*a*b* color space is derived from the CIEXYZ tristimulus values. The L*a*b* space consists of a luminosity layer 'L*', a chromaticity-layer 'a*' that indicates where color falls along the red-green axis, and a chromaticity-layer 'b*' that indicates where the color falls along the blue-yellow axis. All of the color information is in the 'a*' and 'b*' layers and can measure the difference between two colors using the Euclidean distance metric.

Convert the image to L*a*b* color space using function *makecform* and function *applycform* like this.

```
cform  =  makecform('srgb2lab');
lab_I   =  applycform(I,cform);
```

Function *makecform* will creates the color transformation.

Structure *C* that defines the color space conversion specified by type. To perform the transformation, pass the color transformation structure as an argument to the *applycform* function.

C = makecform(type)

The type argument specifies one of the conversions listed in the Table 3.3. Function *makecform* supports conversions between members of the family of device independent color spaces defined by the CIE, Commission Internationale de l'Éclairage (International Commission on Illumination). In addition, function *makecform* supports conversions to and from the *sRGB* standard. For a list of the abbreviations used by the image processing for each color space, see the Table 3.3 and Table 3.4.

Table 3.3 The argument type specifies the conversions [30]

Type	Description
'lab2lch'	Convert from *L*a*b** to the *L*ch* color space.
'lab2srgb'	Convert from *L*a*b** to the *srgb* color space.
'lab2xyz'	Convert from *L*a*b** to the *XYZ* color space.
'lch2lab'	Convert from *L*ch* to the *L*a*b** color space.
'srgb2lab'	Convert from *srgb* to the *L*a*b** color space.
'srgb2xyz'	Convert from *srgb* to the *XYZ* color space.
'upvpl2xyz'	Convert from *u'v'L* to the *XYZ* color space.
'uvl2xyz'	Convert from *uvL* to the *XYZ* color space.
'xyl2xyz'	Convert from *xyY* to the *XYZ* color space.
'xyz2lab'	Convert from *XYZ* to the *L*a*b** color space.
'xyz2srgb'	Convert from *XYZ* to the *srgb* color space.
'xyz2upvpl'	Convert from *XYZ* to the *u'v'L* color space.
'xyz2uvl'	Convert from *XYZ* to the *uvL* color space.
'xyz2xyl'	Convert from *XYZ* to the *xyY* color space.

Table 3.4 Abbreviations to represent for each color space [30]

Abbreviation	**Description**
xyz	1931 CIE XYZ tristimulus values (2° observer)
xyl	1931 CIE xyY chromaticity values (2° observer)
uvl	1960 CIE uvL values
upvpl	1976 CIE the u’v’L values
lab	1976 CIE L*a*b* values
lch	Polar transformation of CIE L*a*b* values, where c = chroma and h = hue
srgb	Standard computer monitor RGB values, (IEC 61966-2-1)

Function *applycform* will converts the color values in *I* (image) to the color space specified in the color transformation structure *C*. The color transformation structure specifies various parameters of the transformation.

out = applycform(I,C)

If *I* is two-dimensional, each row is interpreted as a color. Variable *I* typically has either three or four columns, depending on the input color space. Variable *out* has the same number of rows and either three or four columns, depending on the output color space.

If *I* is three-dimensional, each row-column location is interpreted as a color, and size*(I,3)* is typically either three or four, depending on the input color space. Variable *out* has the same number of rows and columns as *I*, and size*(out,3)* is either three or four, depending on the output color space.

- **Class Support**

Variable *I* must be a real, nonsparse, finite array of class uint8, uint16, or double. The output array out has the same class and size as the input array, unless the output color space is XYZ. If the input is XYZ data of class uint8, the output is of class uint16, because there is no standard 8-bit encoding defined for XYZ color values.

3.4.3.2 Classify the colors in 'a*b*' space using K-means clustering

Since the color information exists in the 'a*b*' space, objects are pixels with 'a*' and 'b*' values. Use function *kmeans* to cluster the objects into two clusters using the Euclidean distance metric. The cluster_center contains the mean 'a*' and 'b*' value for each cluster.

```
ab  =  double(lab_I(:,:,2:3));
nrows  =  size(ab,1);
ncols  =  size(ab,2);
ab  =  reshape(ab,nrows*ncols,2);
```

Repeat the clustering 3 times to avoid local minima

```
nColors = 2;
[cluster_idx cluster_center] =kmeans(ab,nColors,'distance', ...
                                        'sqEuclidean','Replicates',3);
```

3.4.3.3 Label every pixel in the image using the results from K-means

For every object in RSS image, function *kmeans* returns an index corresponding to a cluster and label every pixel *(pixel_label)* in the image with its cluster index (*cluster_idx*). Function *reshape* will returns the *nrows-by-ncols* matrix *pixel_label* whose elements are taken column-wise from *cluster_idx*.

```
pixel_labels  =  reshape(cluster_idx,nrows,ncols);
```

3.4.3.4 Create images that segment the RSS image by color

Using *pixel_labels*, it's can separate objects in RSS image by color, which will result in two images. Function *cell* will creates an m-by-n (rows-by-columns) cell array of empty matrices. Function *repmat* will creates a large matrix *rgb_label* consisting of an m-by-n tiling of copies of *pixel_labels*.

```
segmented_images = cell(1,3);
rgb_label = repmat(pixel_labels,[1 1 3]); nColors = 2;
for k = 1:nColors
   color = color_image;
   color(rgb_label ~= k) = 0;
   segmented_images{k} = color;
end
imshow(segmented_images{1}), title('Objects in Cluster 1');
imshow(segmented_images{2}), title('Objects in Cluster 2');
```

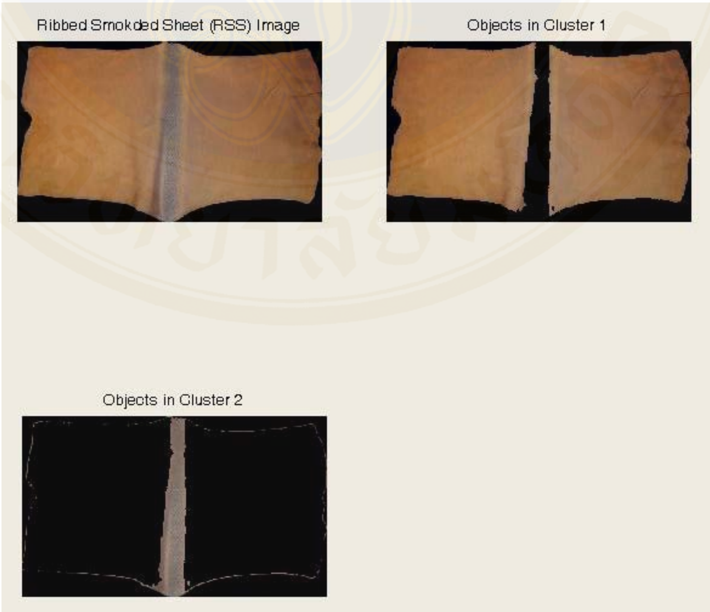


Figure 3.12 Image of RSS after clustering

3.4.4 Calculation of RSS grades

This process wills calculation of RSS grades by calculate mean of RGB information from the most clustering area. Using area of RSS image show in Figure 3.12 to evaluate and group of color values show in Figure 3.13, for a list of the parameter used by the calculation of RSS grades, see the Table 3.5. The calculation of process use cluster1 which be the most cluster area. This is an example processes calculation of cluster1 following:

```
if [meanR1 meanG1 meanB1]

are in range of [minRrss1 minGrss1 minBrss1] and [maxRrss1 maxGrss1 maxBrss1]
then result = 1, msg('RSS Grade = RSS1');

else if
are in range of [minRrss2 minGrss2 minBrss2] and [maxRrss2 maxGrss2 maxBrss2]
then result = 2, msg('RSS Grade = RSS2');

else if
are in range of [minRrss3 minGrss3 minBrss3] and [maxRrss3 maxGrss3 maxBrss3]
then result = 3, msg('RSS Grade = RSS3');

else if
are in range of [minRrss4 minGrss4 minBrss4] and [maxRrss4 maxGrss4 maxBrss4]
then result = 4, msg('RSS Grade = RSS4');

else if
are in range of [minRrss5 minGrss5 minBrss5] and [maxRrss5 maxGrss5 maxBrss5]
then result = 5, msg('RSS Grade = RSS5');

else msg('This mean RGB information not in RSS1 – RSS5');
end;
```

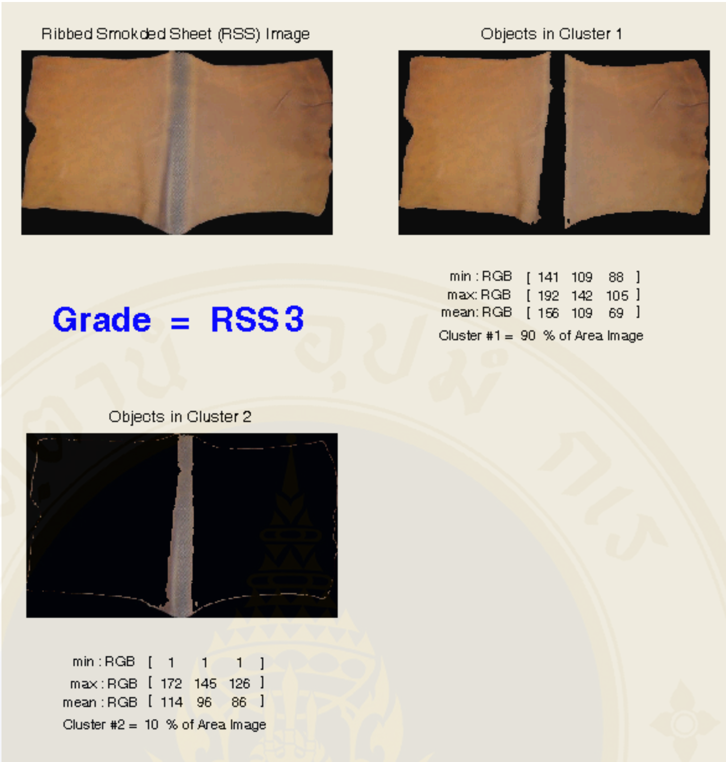


Figure 3.13 Example RSS images Cluster 1 – Cluster 2

Table 3.5 The parameters use in calculation of RSS grades

Parameter	Value
meanR1	Mean of **R**ed color information from cluster number 1.
meanG1	Mean of **G**reen color information from cluster number 1.
meanB1	Mean of **B**lue color information from cluster number 1.
meanR2	Mean of **R**ed color information from cluster number 2.
meanG2	Mean of **G**reen color information from cluster number 2.
meanB2	Mean of **B**lue color information from cluster number 2.
minRrss1	Minimum mean of **R**ed color information from training RSS grade 1.
minGrss1	Minimum mean of **G**reen color information from training RSS grade 1.
minBrss1	Minimum mean of **B**lue color information from training RSS grade 1.
maxRrss1	Minimum mean of **R**ed color information from training RSS grade 1.
maxGrss1	Minimum mean of **G**reen color information from training RSS grade 1.
maxBrss1	Minimum mean of **B**lue color information from training RSS grade 1.

Table 3.5 The parameters use in calculation RSS grades (Continued)

Parameter	**Value**
minRrss2	Minimum mean of **R**ed color information from training RSS grade 2.
minGrss2	Minimum mean of **G**reen color information from training RSS grade 2.
minBrss2	Minimum mean of **B**lue color information from training RSS grade 2.
maxRrss2	Maximum mean of **R**ed color information from training RSS grade 2.
maxGrss2	Maximum mean of **G**reen color information from training RSS grade 2.
maxBrss2	Maximum mean of **B**lue color information from training RSS grade 2.
minRrss3	Minimum mean of **R**ed color information from training RSS grade 3.
minGrss3	Minimum mean of **G**reen color information from training RSS grade 3.
minBrss3	Minimum mean of **B**lue color information from training RSS grade 3.
maxRrss3	Maximum mean of **R**ed color information from training RSS grade 3.
maxGrss3	Maximum mean of **G**reen color information from training RSS grade 3.
maxBrss3	Maximum mean of **B**lue color information from training RSS grade 3.
minRrss4	Minimum mean of **R**ed color information from training RSS grade 4.
minGrss4	Minimum mean of **G**reen color information from training RSS grade 4.
minBrss4	Minimum mean of **B**lue color information from training RSS grade 4.
maxRrss4	Maximum mean of **R**ed color information from training RSS grade 4.
maxGrss4	Maximum mean of **G**reen color information from training RSS grade 4.
maxBrss4	Maximum mean of **B**lue color information from training RSS grade 4.
minRrss5	Minimum mean of **R**ed color information from training RSS grade 5.
minGrss5	Minimum mean of **G**reen color information from training RSS grade 5.
minBrss5	Minimum mean of **B**lue color information from training RSS grade 5.
maxRrss5	Maximum mean of **R**ed color information from training RSS grade 5.
maxGrss5	Maximum mean of **G**reen color information from training RSS grade 5.
maxBrss5	Maximum mean of **B**lue color information from training RSS grade 5.

Table 3.5 The parameters use in calculation RSS grades (Continued)

Parameter	**Value**
minR1	Minimum **R**ed color information from cluster number 1.
minG1	Minimum **G**reen color information from cluster number 1.
minB1	Minimum **B**lue color information from cluster number 1.
maxR1	Maximum **R**ed color information from cluster number 1.
maxG1	Maximum **G**reen color information from cluster number 1.
maxB1	Maximum **B**lue color information from cluster number 1.
minR2	Minimum **R**ed color information from cluster number 2.
minG2	Minimum **G**reen color information from cluster number 2.
minB2	Minimum **B**lue color information from cluster number 2.
maxR2	Maximum **R**ed color information from cluster number 2.
maxG2	Maximum **G**reen color information from cluster number 2.
maxB2	Maximum **B**lue color information from cluster number 2.
Clust1_Area	Quantity of color information from cluster number 1.
Clust2_Area	Quantity of color information from cluster number 2.
result	Result of calculation RSS grades.
tLoad	Load image process time.
tCrop	Crop image process time.
tKmean	K-means clustering process time.
tSum	The summation time of load image, crop image and clustering image.

3.4.5 Display of results

This process will display of results from calculation of Ribbed Smoked Sheet (RSS) grades using mean RGB (red, green, blue) information. Evaluate results of RSS image show in Figure 3.14**.**

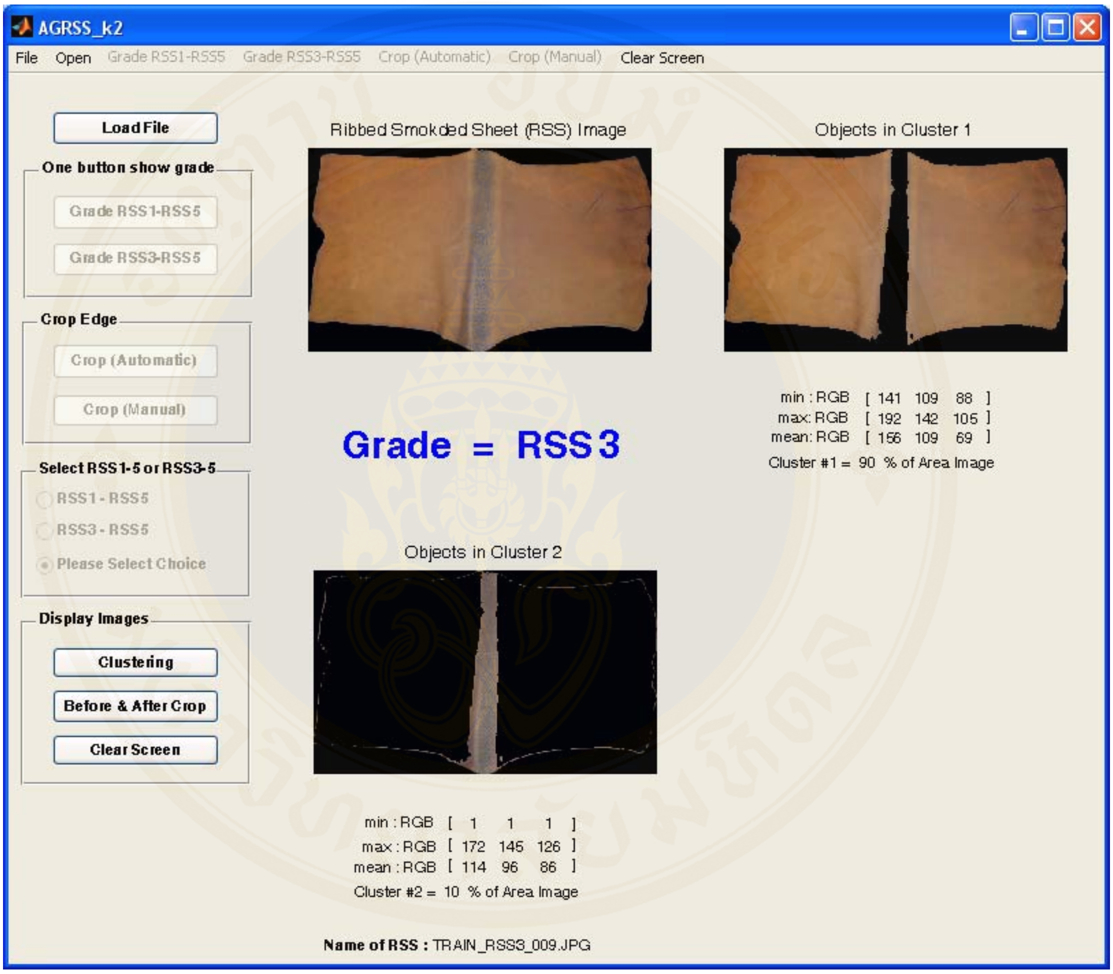


Figure 3.14 Result of RSS grade

3.5 K-means clustering

This section gives a description and an example of using the MATLAB function *kmeans* for K-means clustering [30].

K-means clustering can best be described as a partitioning method [30]. That is, the function *kmeans* partitions the observations in your data into *K* mutually exclusive clusters, and returns a vector of indices indicating to which of the *k* clusters it has assigned each observation. Function *kmeans* does not create a tree structure to describe the groupings in your data, but creates a single level of clusters. Another difference is that K-means clustering uses the actual observations of objects or individuals in your data, and not just their proximities. These differences often mean that *kmeans* is more suitable for clustering large amounts of data.

Function *kmeans* treats each observation in your data as an object having a location in space. It finds a partition in which objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible. You can choose from five different distance measures, depending on the kind of data you are clustering. Each cluster in the partition is defined by its member objects and by its centroid, or center. The centroid for each cluster is the point to which the sum of distances from all objects in that cluster is minimized. Function *kmeans* computes cluster centroids differently for each distance measure, to minimize the sum with respect to the measure that you specify.

Function *kmeans* uses an iterative algorithm that minimizes the sum of distances from each object to its cluster centroid, over all clusters. This algorithm moves objects between clusters until the sum cannot be decreased further. The result is a set of clusters that are as compact and well-separated as possible. You can control the details of the minimization using several optional input parameters to *kmeans*, including ones for the initial values of the cluster centroids, and for the maximum number of iterations.

3.5.1 Description

[...] = kmeans(...,'param1',val1,'param2',val2,...) enables you to specify optional parameter name-value pairs to control the iterative algorithm used by *kmeans*. Valid parameters are showing in the Table 3.6.

This work has specified the K-mean clustering using these syntaxes:

[cluster_idx cluster_center] = kmeans(ab,nColors,'distance', ...
'sqEuclidean', 'Replicates',3);

This iterative partitioning minimizes the sum, over all clusters, of the within-cluster sums of point-to-cluster-centroid distances. Function *kmeans* returns an n-by-1 vector, *[cluster_idx cluster_center]* containing the cluster indices of each point. By default, *kmeans* uses squared Euclidean distances.

For the component of this syntaxes you can see at subsection 3.4.3.2 Classify the colors in 'a*b*' space using K-means clustering (page 63).

Table 3.6 The parameters to control the iterative algorithm [30]

Parameter	**Value**	
'distance'	Distance measure, in p-dimensional space, that kmeans minimizes with respect to. kmeans computes centroid clusters differently for the different supported distance measures:	
	'sqEuclidean'	Squared Euclidean distance (default). Each centroid is the mean of the points in that cluster.
	'cityblock'	Sum of absolute differences, i.e., L1. Each centroid is the component-wise median of the points in that cluster.'cosine'One minus the cosine of the included angle between points (treated as vectors). Each centroid is the mean of the points in that cluster, after normalizing those points to unit Euclidean length.
	'correlation'	One minus the sample correlation between points (treated as sequences of values). Each centroid is the component-wise mean of the points in that cluster, after centering and normalizing those points to zero mean and unit standard deviation.
	'Hamming'	Percentage of bits that differ (only suitable for binary data). Each centroid is the component-wise median of points in that cluster.

Table 3.6 The parameters to control the iterative algorithm (Continued)

Parameter	Value	
'start'	Method used to choose the initial cluster centroid positions, sometimes known as "seeds." Valid starting values are:	
	'sample'	Select k observations from X at random (default).
	'uniform'	Select k points uniformly at random from the range of X. Not valid with Hamming distance.
	'cluster'	Perform a preliminary clustering phase on a random 10% subsample of X. This preliminary phase is itself initialized using 'sample'.
	Matrix	k-by-p matrix of centroid starting locations. In this case, you can pass in [] for k, and kmeans infers k from the first dimension of the matrix. You can also supply a 3-dimensional array, implying a value for the 'replicates' parameter from the array's third dimension.
'replicates'	Number of times to repeat the clustering, each with a new set of initial cluster centroid positions. kmeans returns the solution with the lowest value for sumd. You can supply 'replicates' implicitly by supplying a 3-dimensional array as the value for the 'start' parameter.	
'maxiter'	Maximum number of iterations. Default is 100.	
'emptyaction'	Action to take if a cluster loses all its member observations. Can be one of:	
	'error'	Treat an empty cluster as an error. (default)
	'drop'	Remove any clusters that become empty. kmeans sets the corresponding return values in C and D to NaN.
	'singleton'	Create a new cluster consisting of the one point furthest from its centroid.

Table 3.6 The parameters to control the iterative algorithm (Continued)

Parameter	**Value**	
'display'	Controls display of output.	
	'off'	Display no output.
	'iter'	Display information about each iteration during minimization, including the iteration number, the optimization phase (see Algorithm), the number of points moved, and the total sum of distances.
	'final'	Display a summary of each replication.
	'notify'	Display only warning and error messages. (default)

3.5.2 Algorithm

Function *kmeans* uses a two-phase iterative algorithm to minimize the sum of point-to-centroid distances, summed over all *k* clusters:

- The first phase uses what the literature often describes as "batch" updates, where each iteration consists of reassigning points to their nearest cluster centroid, all at once, followed by recalculation of cluster centroids. This phase as providing a fast but potentially only approximate solution as a starting point for the second phase.

- The second phase uses what the literature often describes as "online" updates, where points are individually reassigned if doing so will reduce the sum of distances, and cluster centroids are recomputed after each reassignment. Each iteration during this second phase consists of one pass though all the points.

Function *kmeans* can converge to a local optimum, in this case, a partition of points in which moving any single point to a different cluster increases the total sum of distances. This problem can only be solved by an exhaustive choice of starting points.

CHAPTER IV
SYSTEM DESIGN AND IMPLEMENTATION

This chapter describes the system design, implementation and GUI (graphic user interface) of this research. First section explains about system overview of image processing techniques that applied with this research such as Image Acquisition, Segmentation, Clustering, Calculation of RSS (Ribbed Smoked Sheet) grades, and Display of results. Second section describes about step of AGRSS (Analysis Grade of Ribbed Smoked Sheet) system processes. Third section show GUI part with step by step to use this application. Forth section is system training phase and the last section is system testing phase.

4.1 System Overview

The process of Analysis Grade of Ribbed Smoked Sheet (AGRSS) system using an image processing technique content with 5 main processes as follow:

4.1.1 Image Acquisition

Take images of Ribbed Smoked Sheet (RSS) by digital camera. Put RSS on dark background and then insert it into dark box (inside dark box has a control white light). Take images and input those images into AGRSS system.

4.1.2 Segmentation

Identify position of Ribbed Smoked Sheet (RSS) out of background. Find coordinates of RSS and cut background off to get interested only RSS area.

4.1.3 Clustering

Clustering is a way to separate groups of objects. K-means clustering treats each object as having a location in the space. It finds partitions such that objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible.

4.1.4 Calculation of RSS grades

Calculation grades of each RSS from the clustering process.

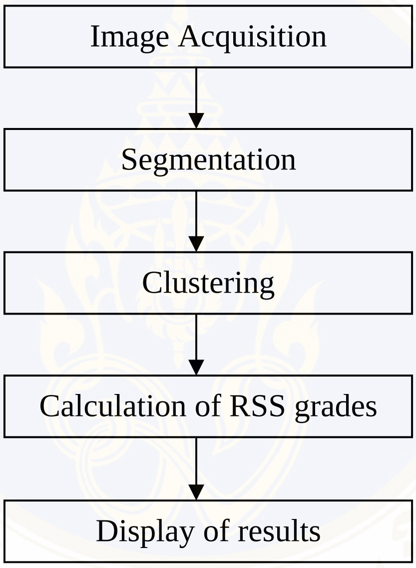


Figure 4.1 Process of Analysis Grade of Ribbed Smoked Sheet system

4.1.5 Display of results

Display of RSS grade calculation results in GUI (graphic user interface) part. Show two clustering image, minimum, maximum and mean RGB (red, green, blue) values of each cluster and RSS grade results.

4.2 Step of Analysis Grade of Ribbed Smoked Sheet (AGRSS) system

4.2.1 Take images of RSS

Take images of Ribbed Smoked Sheet (RSS) by digital camera. Put RSS on dark background and then insert it into dark box (inside dark box has a control white light). Take images and input those images into AGRSS system (the example image was shown in Figure 4.3, page 78).

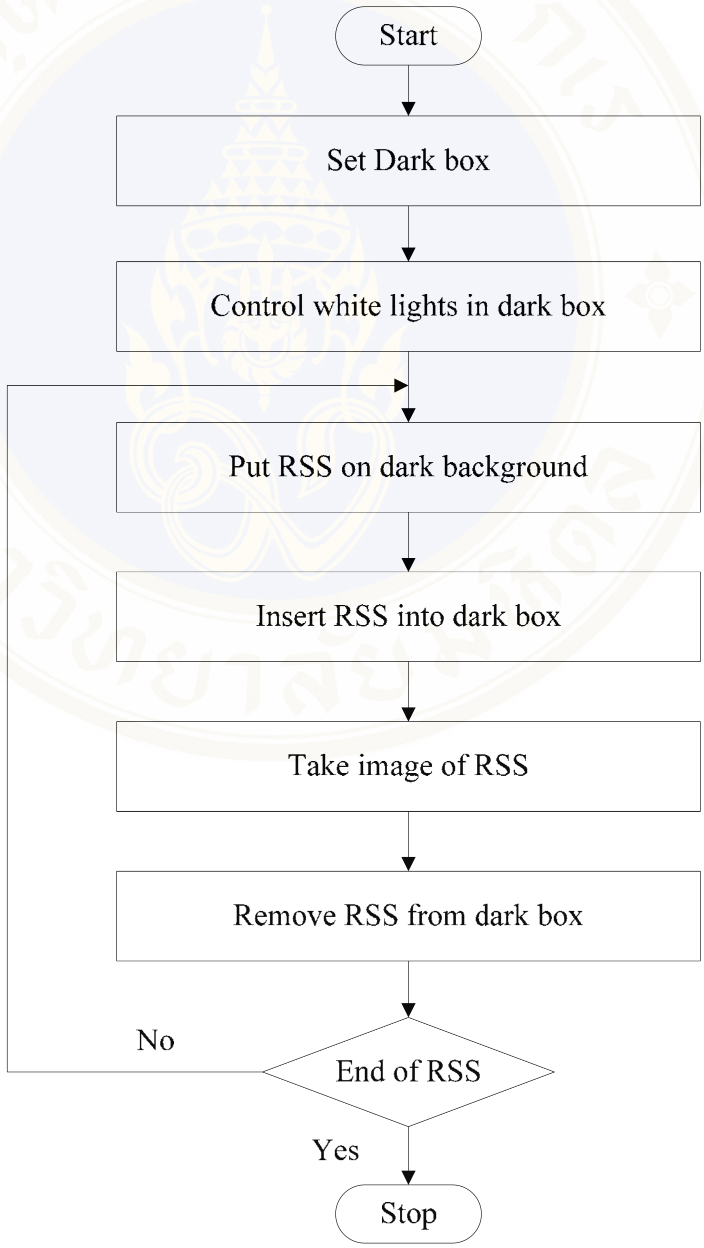


Figure 4.2 Flow Chart of Take images of RSS

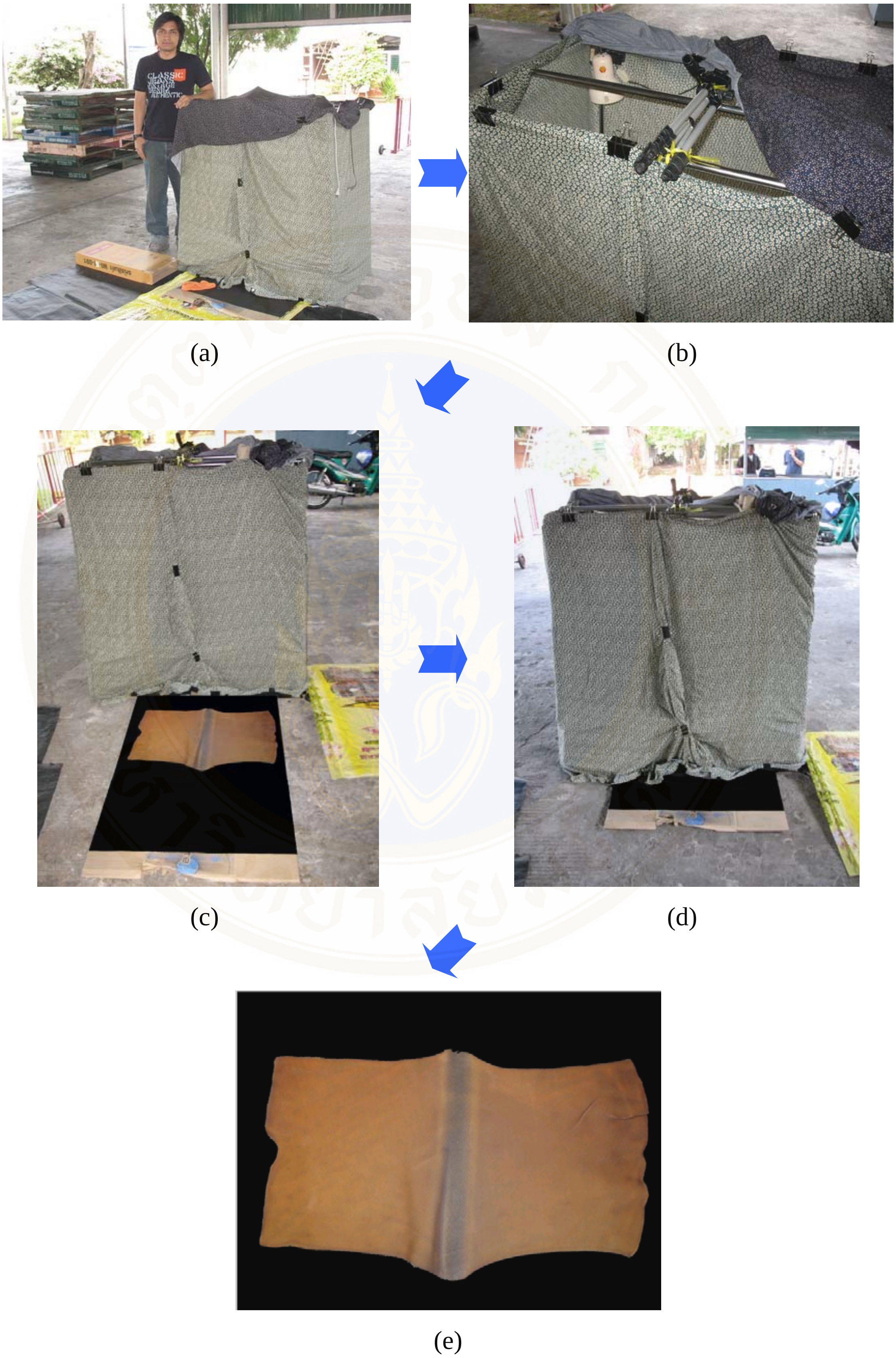


Figure 4.3 Take images of RSS

4.2.2 Convert RGB image to grayscale

Read RGB (red, green, blue) with values 0 – 255 color pixel from RSS image then convert RGB image to grayscale (the example image was shown in Figure 4.5, page 80).

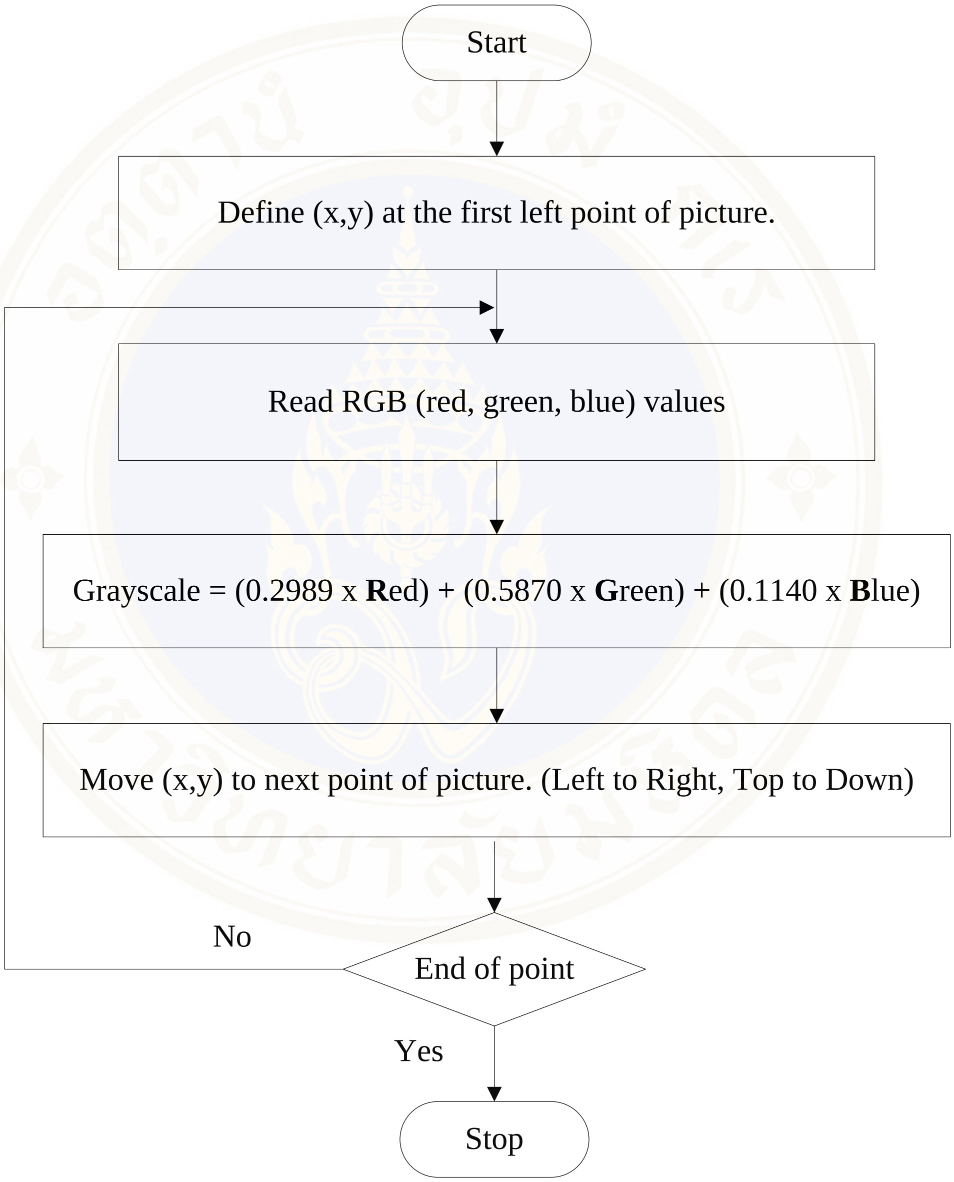


Figure 4.4 Flow Chart of Convert RGB image into grayscale

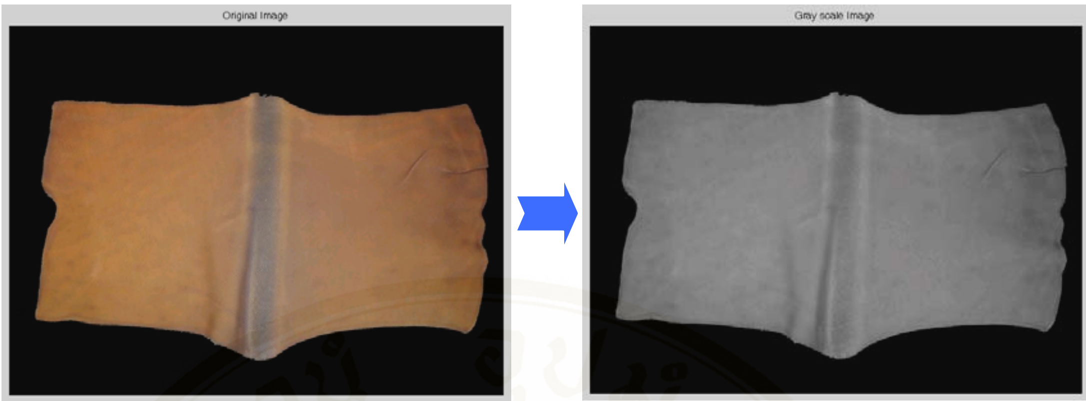


Figure 4.5 Convert RGB image into grayscale

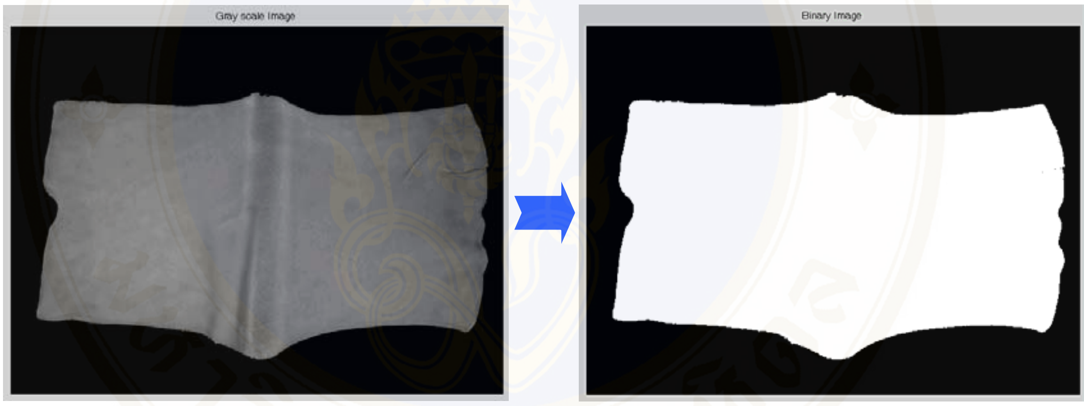


Figure 4.6 Convert grayscale image into binary

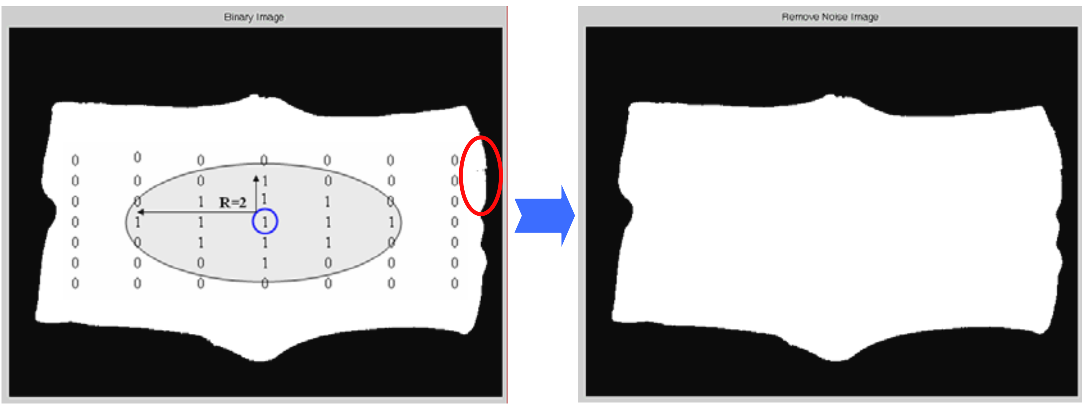


Figure 4.7 Remove Noise

4.2.3 Calculate Threshold value to separate RSS area and background

Calculate Threshold value is the method to separate RSS area and background. It find Histogram from images by Otsu's method calculate value to point Threshold (T). This method separate RSS area and background by compare every pixel with Threshold. Then define point that has value more than Threshold equal to 1 and point has value less than Threshold equal to 0. Define 1 is pixel that was white so it is RSS area and define 0 is pixel that has black color so it is background (the example image was shown in Figure 4.6, page 80). This is a flowchart which shows processes.

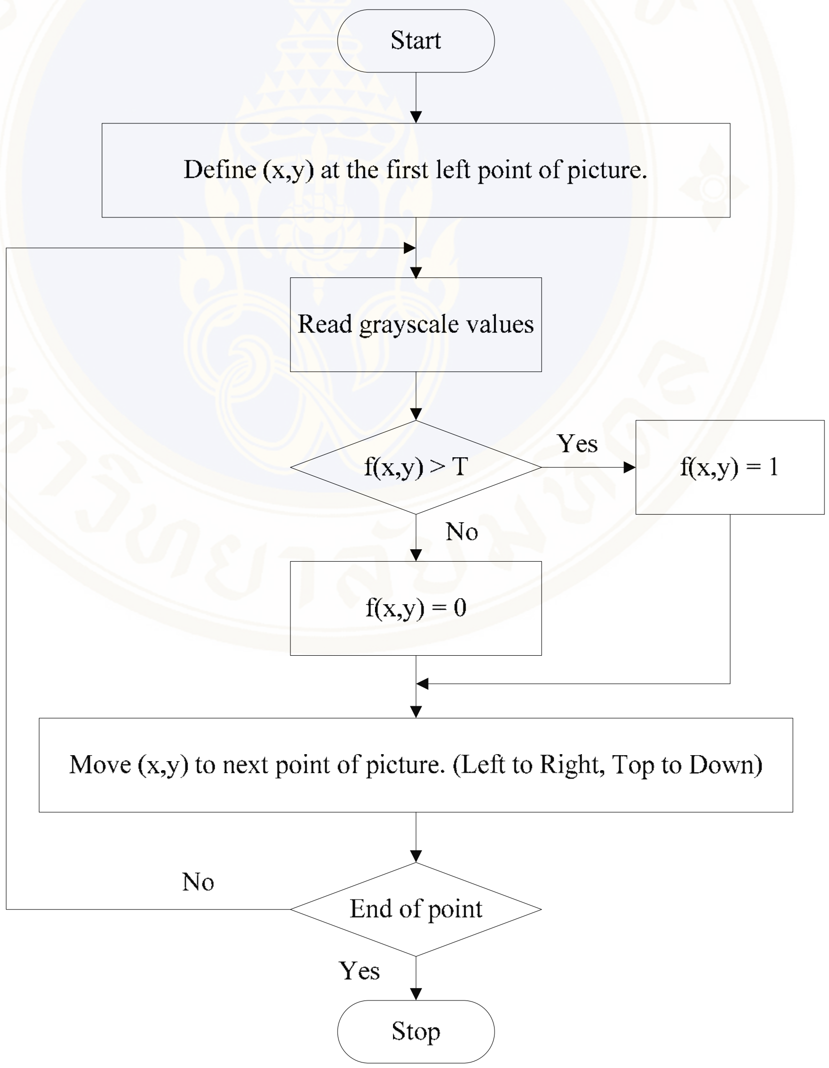


Figure 4.8 Flow Chart of Threshold color pixel

4.2.4 Remove Noise

This process will remove Noise from binary RSS image. First remove a binary image all connected objects that have fewer than 2 pixels. Then join the small black point together by filling in the gaps between them and by smoothing their outer edges then producing a new binary image (the example image was shown in Figure 4.7, page 80).

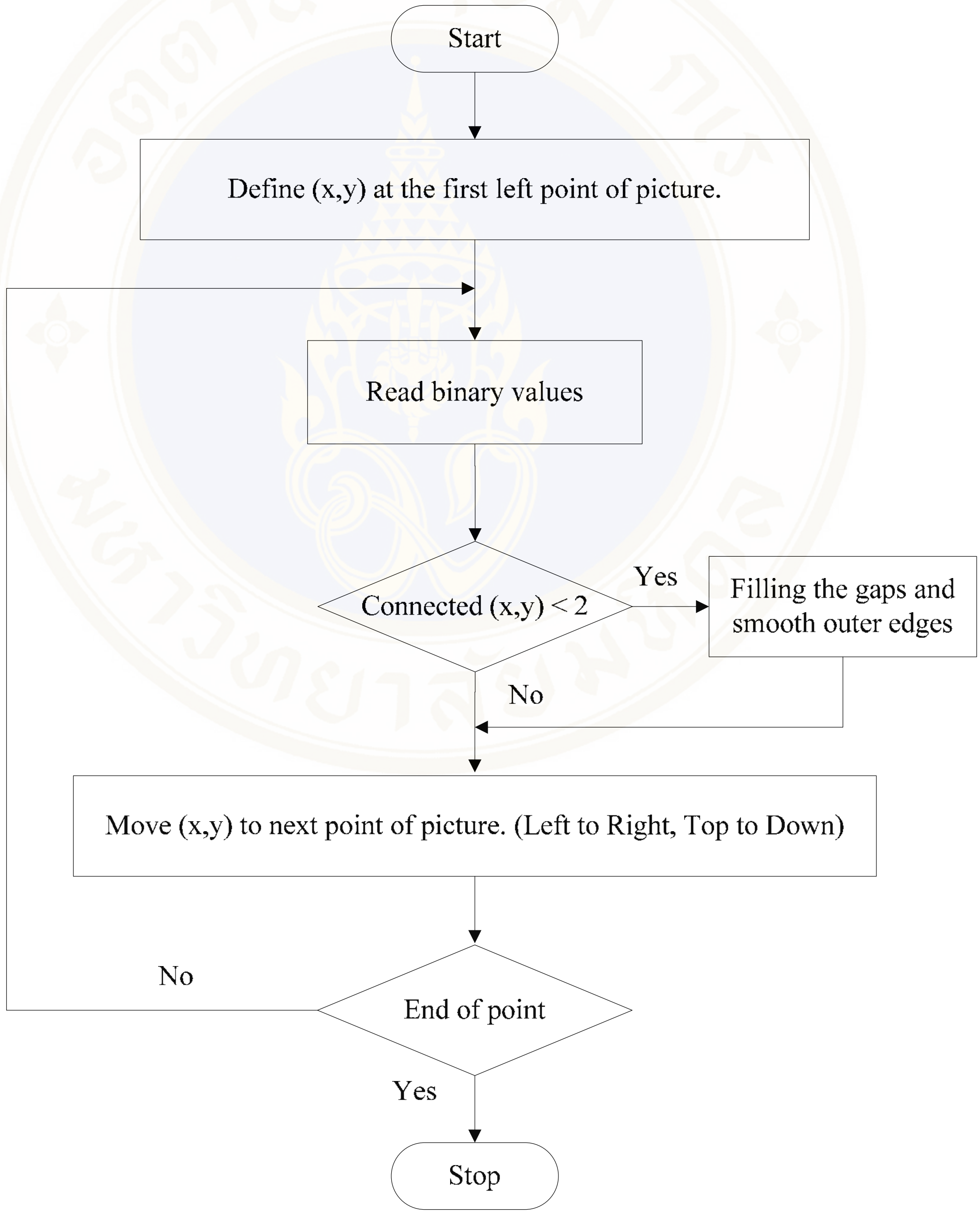


Figure 4.9 Flow Chart of Remove Noise

4.2.5 Calculation of RSS Area

This process will read binary (black and white / BW) values, consider white pixel and define 1 then count total of pixel.

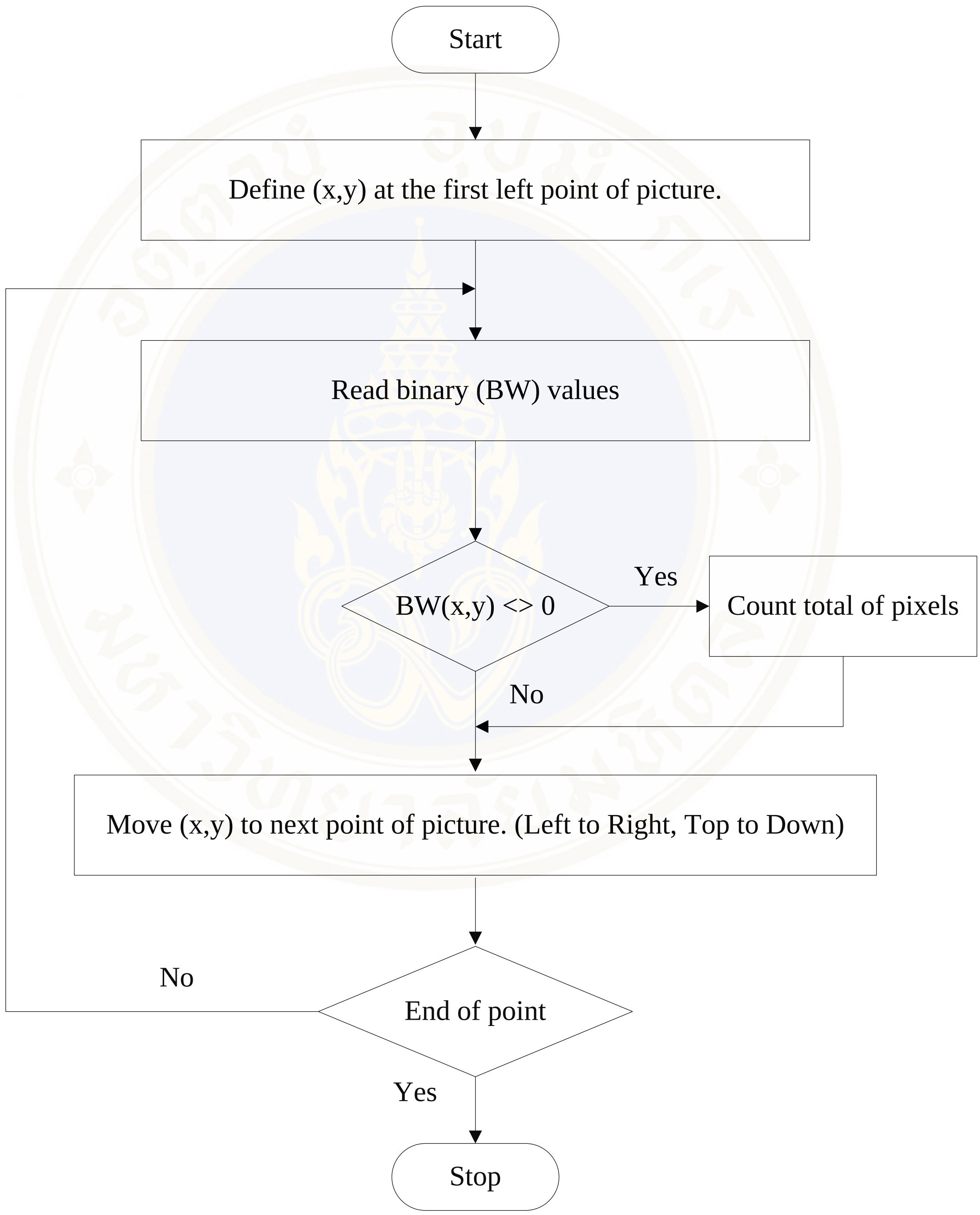


Figure 4.10 Flow Chart of Calculation of RSS area

4.2.6 Find coordinates of RSS area

This process will find coordination of RSS by considering point that has value not equal to 0. Then find Min and Max area of 4 sides by keep value of each position to consider.

Step of Process:

1. Consider Pixel interest is (P<>0)

    If P <> 0 Keep value position (x,y)

    If P = 0 Move to next position

2. Compare coordinate of 4 side by consider x,y define:

    If x was Minimum then x min = Left Margin

    If x was Maximum then x max = Right Margin

    If y was Minimum then y min = Top Margin

    If y was Maximum then y max = Down Margin

When we get coordinate of 4 sides then keep value to crop RSS images.

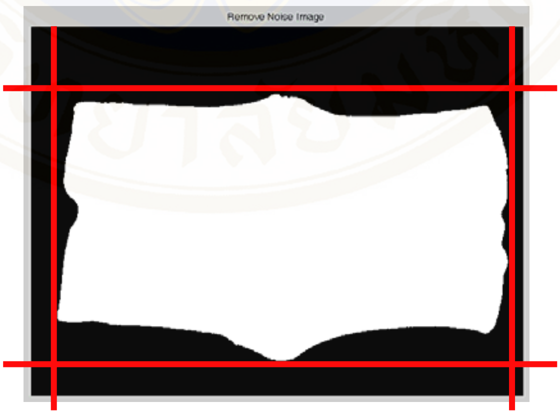


Figure 4.11 Find coordinates of RSS area

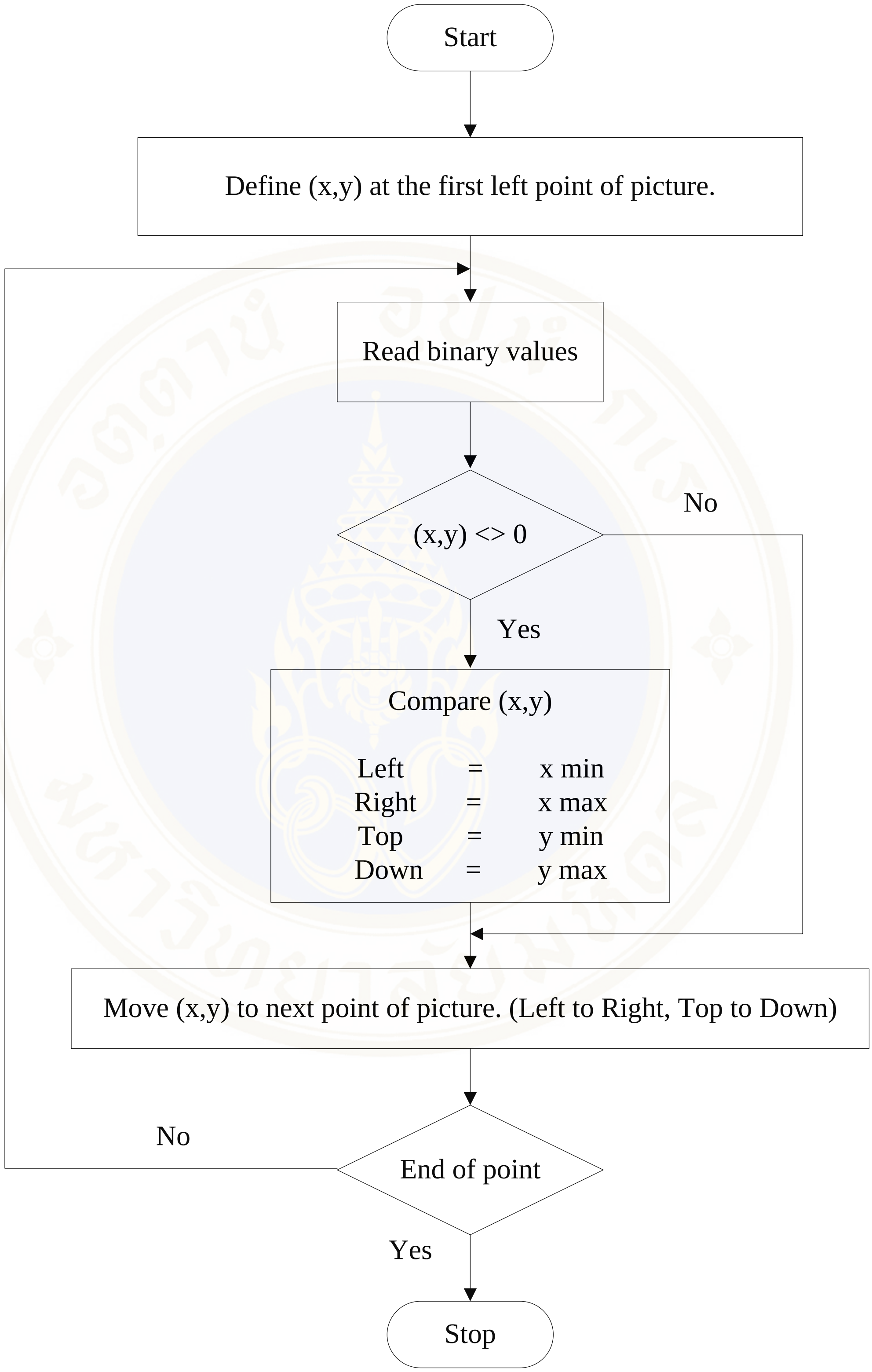


Figure 4.12 Flow Chart of Find coordinates of 4 sides

4.2.7 Crop coordinates of RSS area

This process will receive (x min, y min) and (x max, y max) values from calculation coordinate of RSS area. After that the system will read value RGB (red, green, blue) of RSS image to crop background out at the position of (x min, y min) and (x max, y max) then show the image that fit coordinate of RSS area.

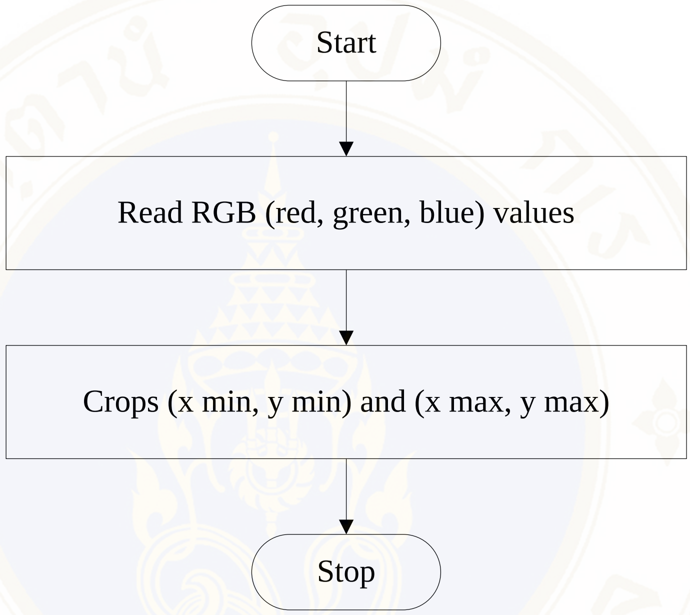


Figure 4.13 Flow Chart of Crop coordinates of RSS area

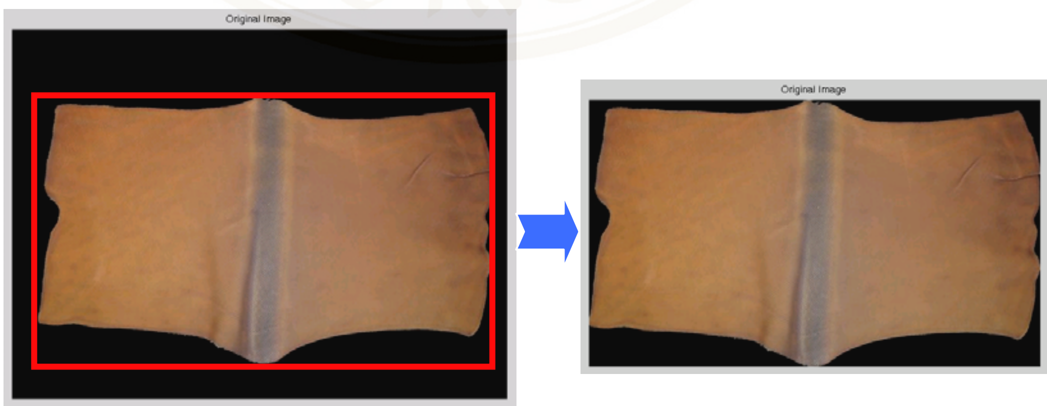


Figure 4.14 Crop coordinates of RSS area

4.2.8 Clustering

Clustering is a way to separate groups of objects and segment colors in an automated fashion using the L*a*b* color space [28] and K-means clustering. Since the color information exists in the 'a*b*' space, your objects are pixels with 'a*' and 'b*' values (the example image was shown in Figure 4.16, page 88).

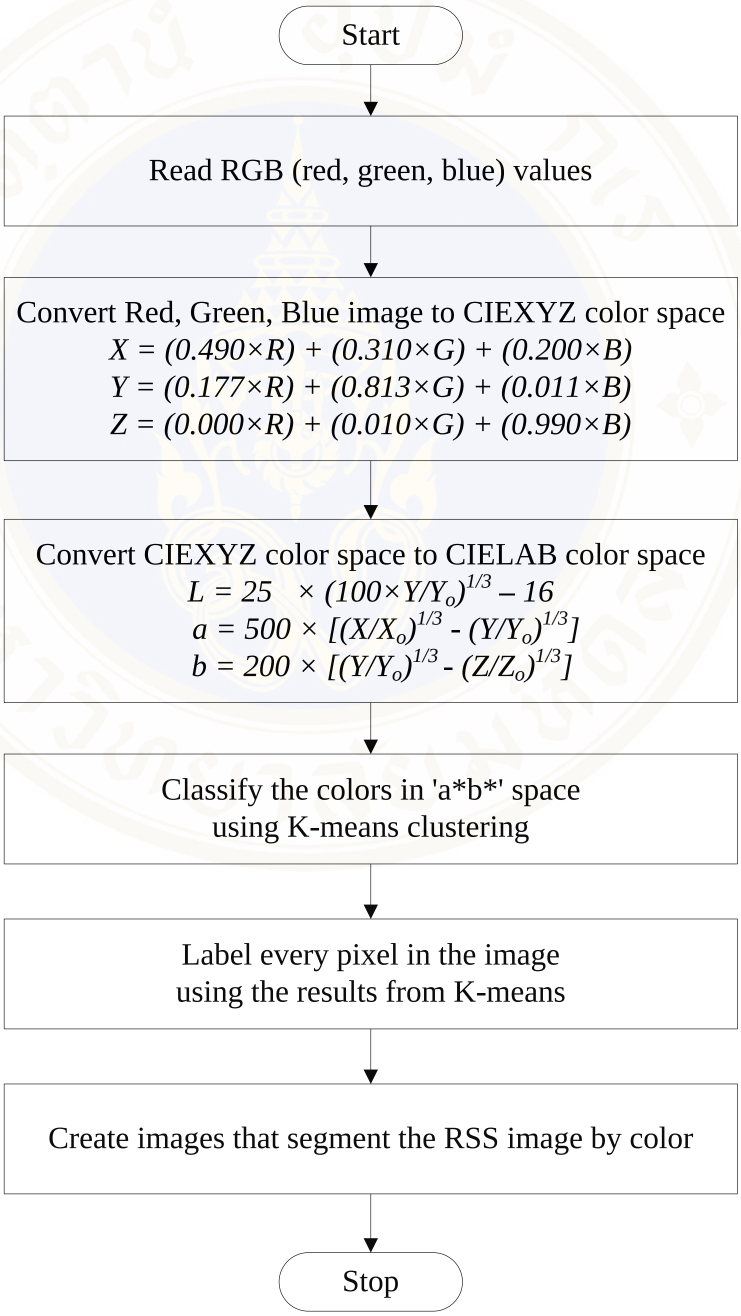


Figure 4.15 Flow Chart of Clustering process

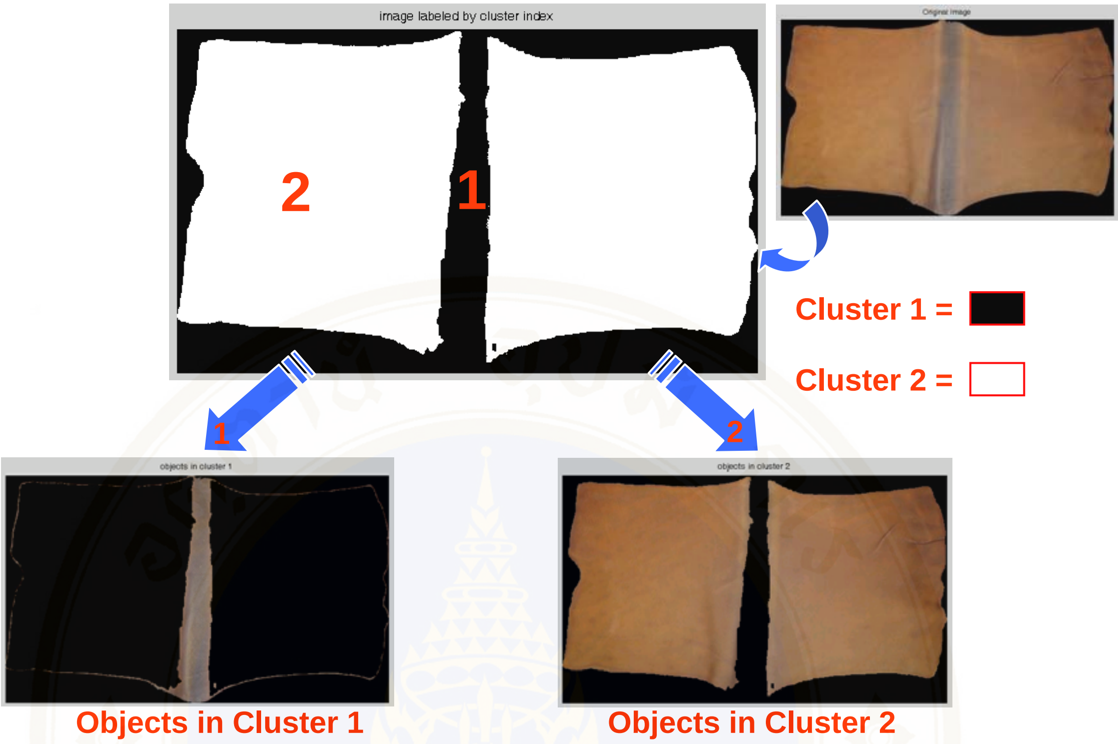


Figure 4.16 Clustering process

meanB1	69											
meanG1	93-102	103-107			108-115					116-119		
meanR1	141-184	141-156	157-159	160-184	141-153	154-155	156-159	160-165	166-184	141-146	147-149	150-184
Result	RSS3	RSS3	RSS4	RSS3	RSS3	RSS5	RSS3	RSS4	RSS3	RSS3	RSS2	RSS3

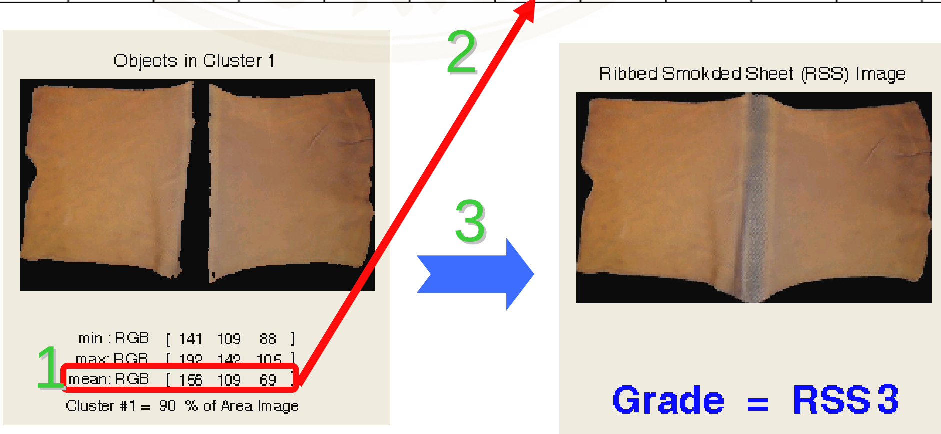


Figure 4.17 Calculation of RSS grades

4.2.9 Calculation of RSS grades

This process wills calculation of RSS grades by retrieve mean RGB information from the first clustering area which is the most area and comparing in the decision RSS grade table (see the example calculation image in Figure 4.17, page 88).

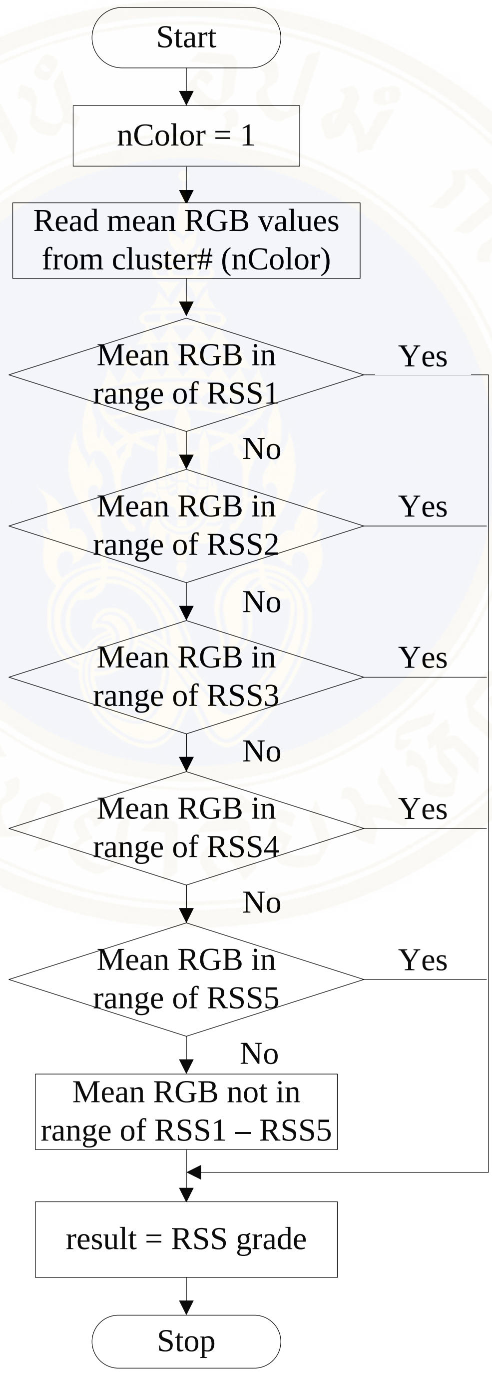


Figure 4.18 Flow Chart of Calculation of RSS grades

4.3 Graphic User Interface (GUI) part

This section will explain about GUI (graphic user interface) part. Start from interface of Analysis Grade of Ribbed Smoked Sheet (AGRSS) application. Next topic will display step by step to use this application.

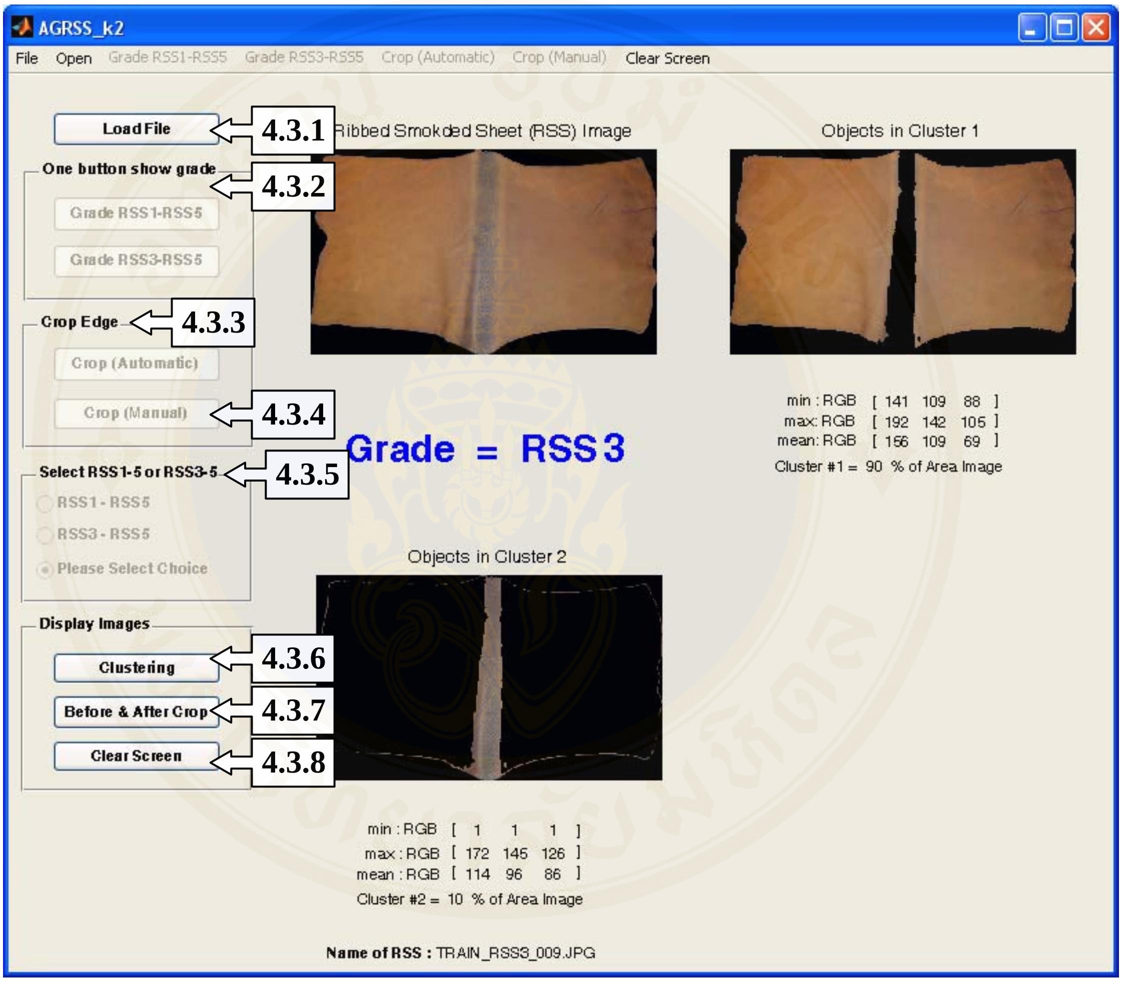


Figure 4.19 Graphic user interface (GUI) of AGRSS application

4.3.1 Load File button

Press this button the system will pop up windows for the users to browse RSS (Ribbed Smoked Sheet) image file that user want to evaluate RSS grades with Analysis Grade of Ribbed Smoked Sheet (AGRSS) application.

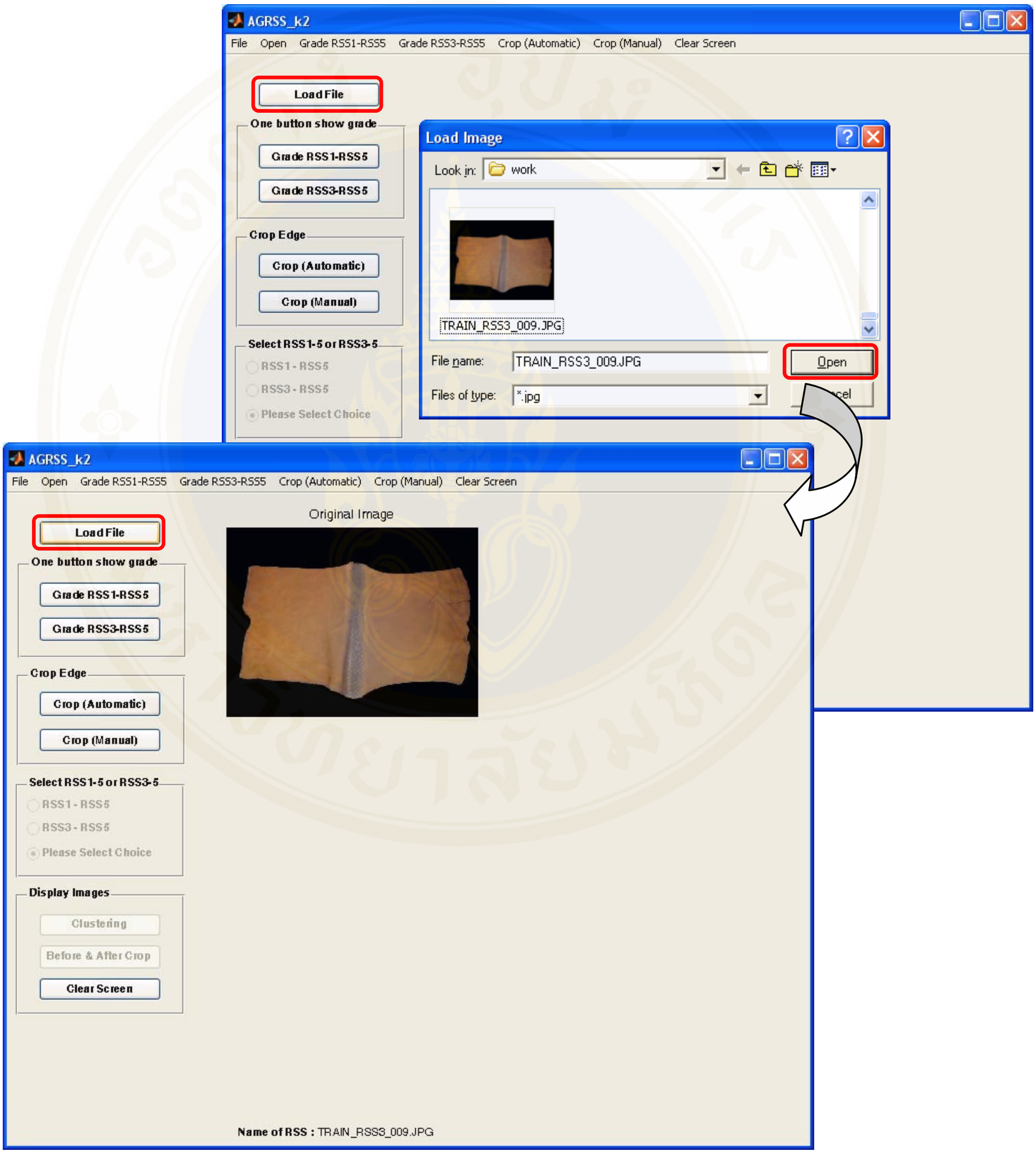


Figure 4.20 Graphic user interface (GUI) of Load File button

4.3.2 One button show grade panel

This panel is all in one button zone. After user browse RSS image file into AGRSS application when user press Grade RSS1-RSS5 button or Grade RSS3-RSS5 button, the system will run automatically from detect edge, automatic crop edge, clustering until display result of RSS grades to user.

Grade RSS1-RSS5 button, this button will separate RSS image into 5 grades start from RSS1, RSS2, RSS3, RSS4 and RSS5. The Grade RSS3-RSS5 button will separate RSS image into only 3 grades start from RSS3, RSS4 and RSS5.

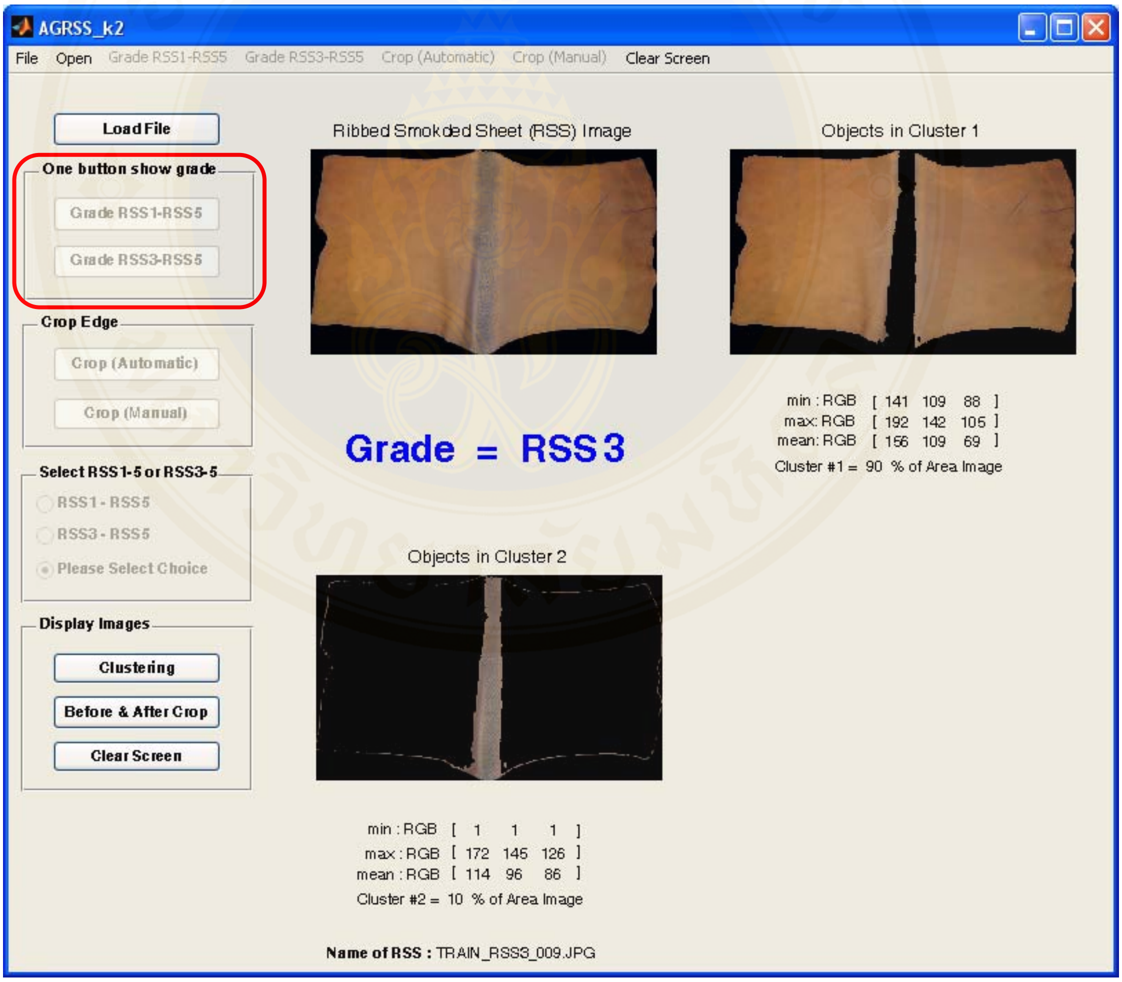


Figure 4.21 Graphic user interface (GUI) of One button show grade panel

4.3.3 Crop Edge panel

This panel contain with Crop (Automatic) button and Crop (Manual) button. After user browse RSS image file into AGRSS application, if user press Crop (Automatic) button, the system will find coordinates of RSS area and crop background out automatically leaving image only RSS area. In the other ways, if users want to selected interested RSS area by manual, the user can use Crop (Manual) button to select anywhere else in RSS image to process which explain in section 4.3.4.

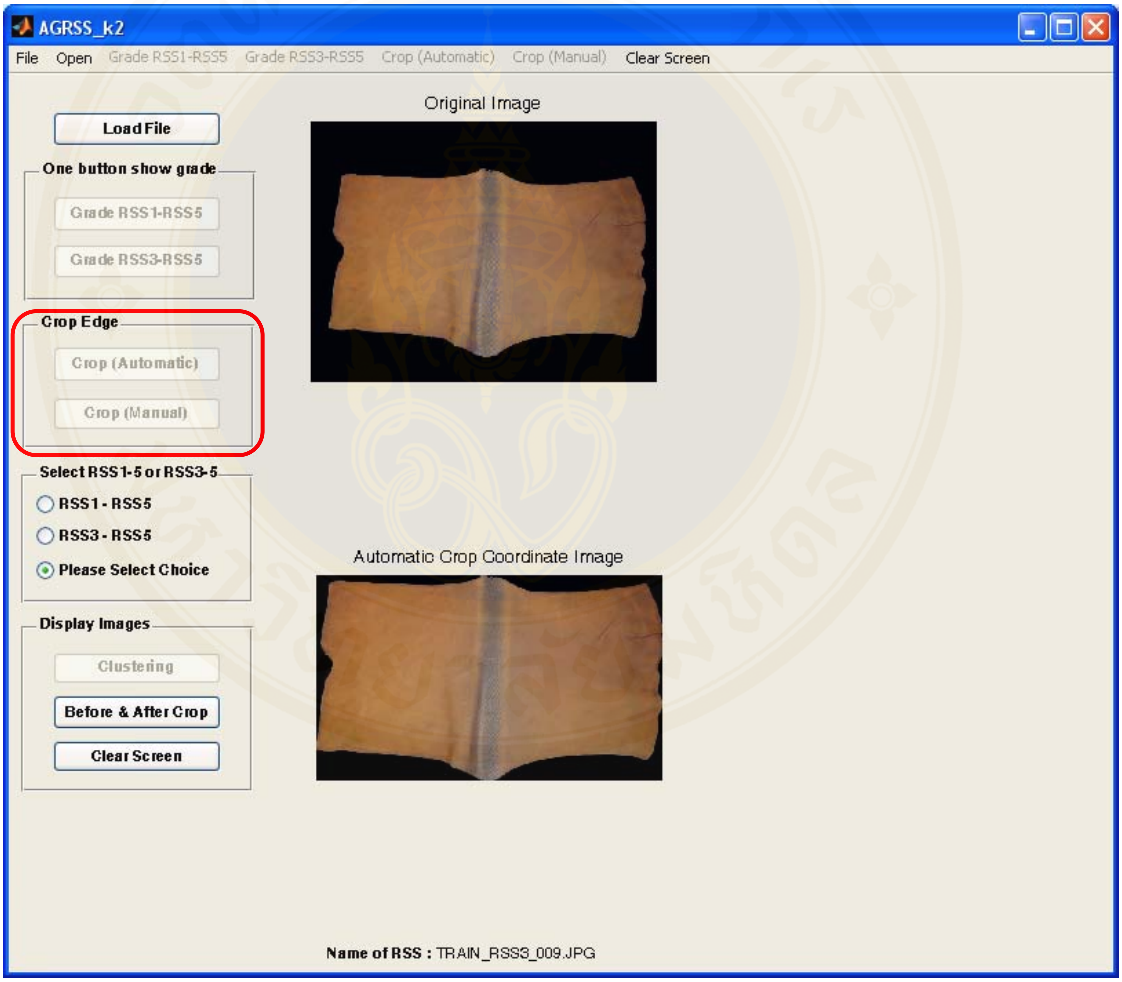


Figure 4.22 Graphic user interface (GUI) of Crop Edge panel

4.3.4 Crop (Manual) button

If users want to manual select interested area, after user browse RSS image file into AGRSS application the user can press Crop (Manual) button, the cursor will changes to crosshairs when it is over the image. Position the crosshairs over a corner of the crop region and press and hold the left mouse button. When users drag the crosshairs over the image it's specify the rectangular crop region. The system will draws a rectangle around the area that user is selecting. When users release the mouse button, the system will creates a new image from the selected region. After that the system will call selected coordinate value to process automatic crop and creates image fit only RSS area.

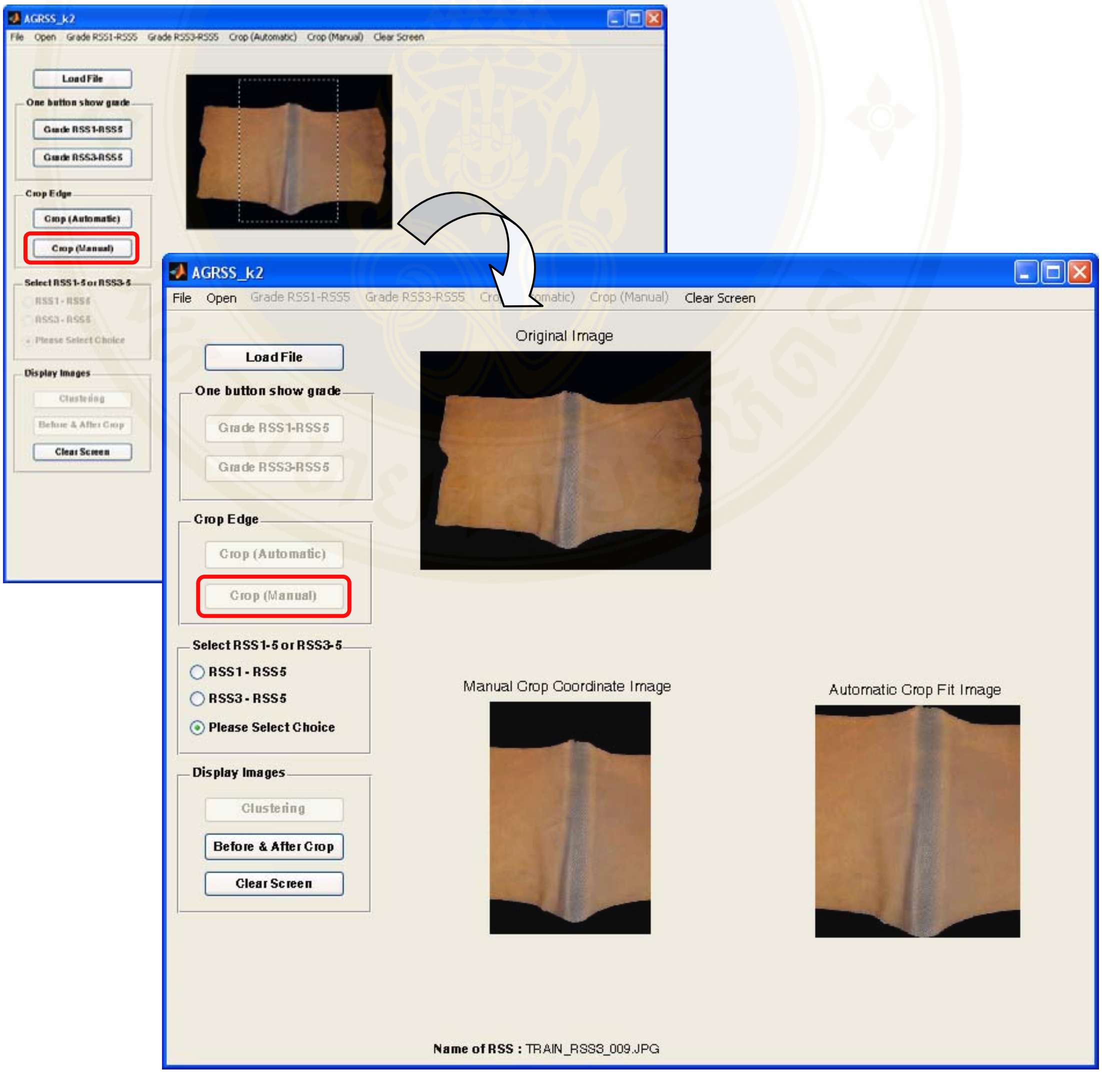


Figure 4.23 Graphic user interface (GUI) of Crop (Manual) button

4.3.5 Select RSS1-5 or RSS3-5 panel

This ration button zone contain with two functions to calculate RSS grade into RSS1-RSS5 or RSS3-RSS5. After the system receive coordinates values of RSS image from the previous processes, the system will clustering RSS image into two clusters that objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible. Then the system will find minimum, maximum and mean RGB (red, green, blue) values from each cluster and display the last result of RSS grade to the user.

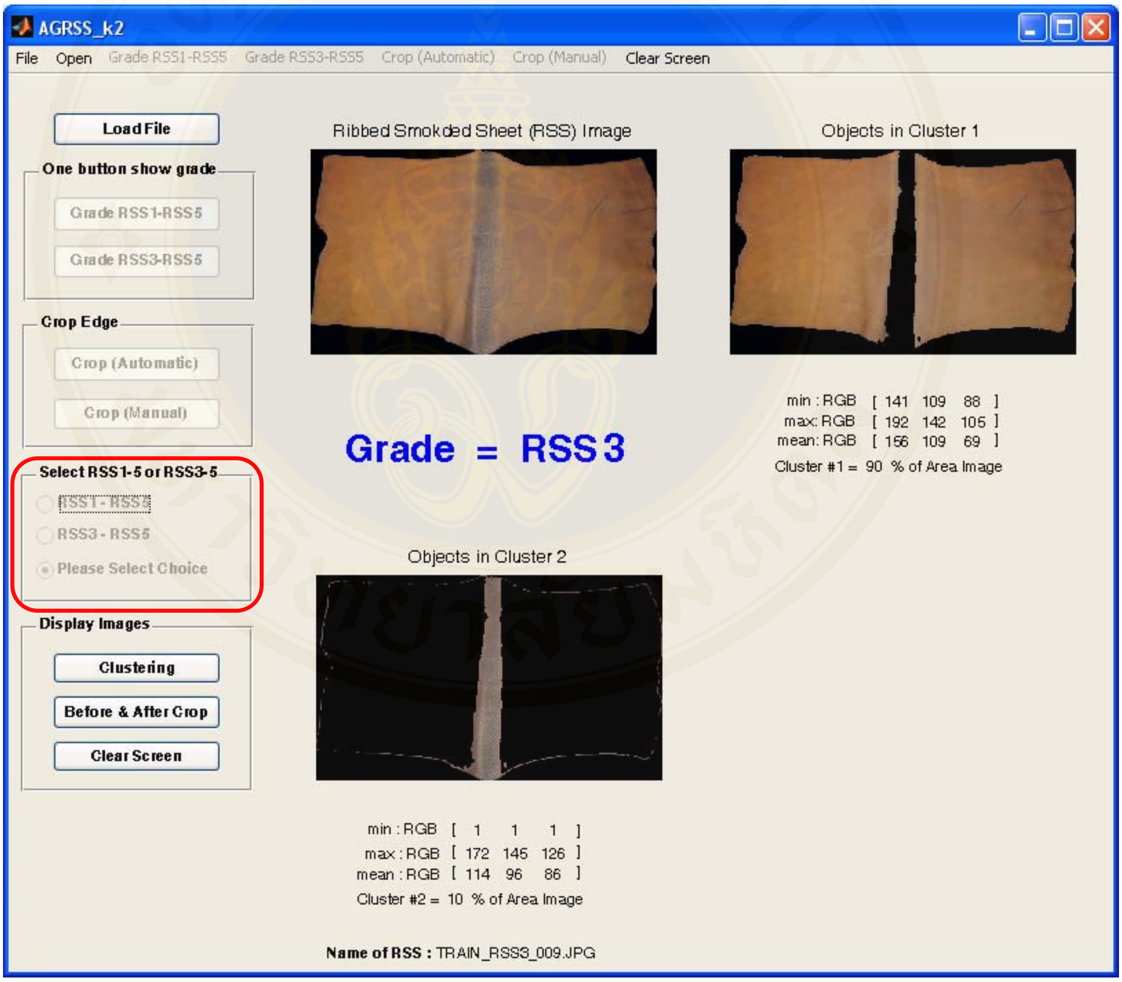


Figure 4.24 Graphic user interface (GUI) of Select RSS1-5 or RSS3-5 panel

4.3.6 Clustering button

This button will show two clustering images that objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible. The minimum, maximum and mean RGB (red, green, blue) values from each clusters and result of calculation of grades will show again with this button.

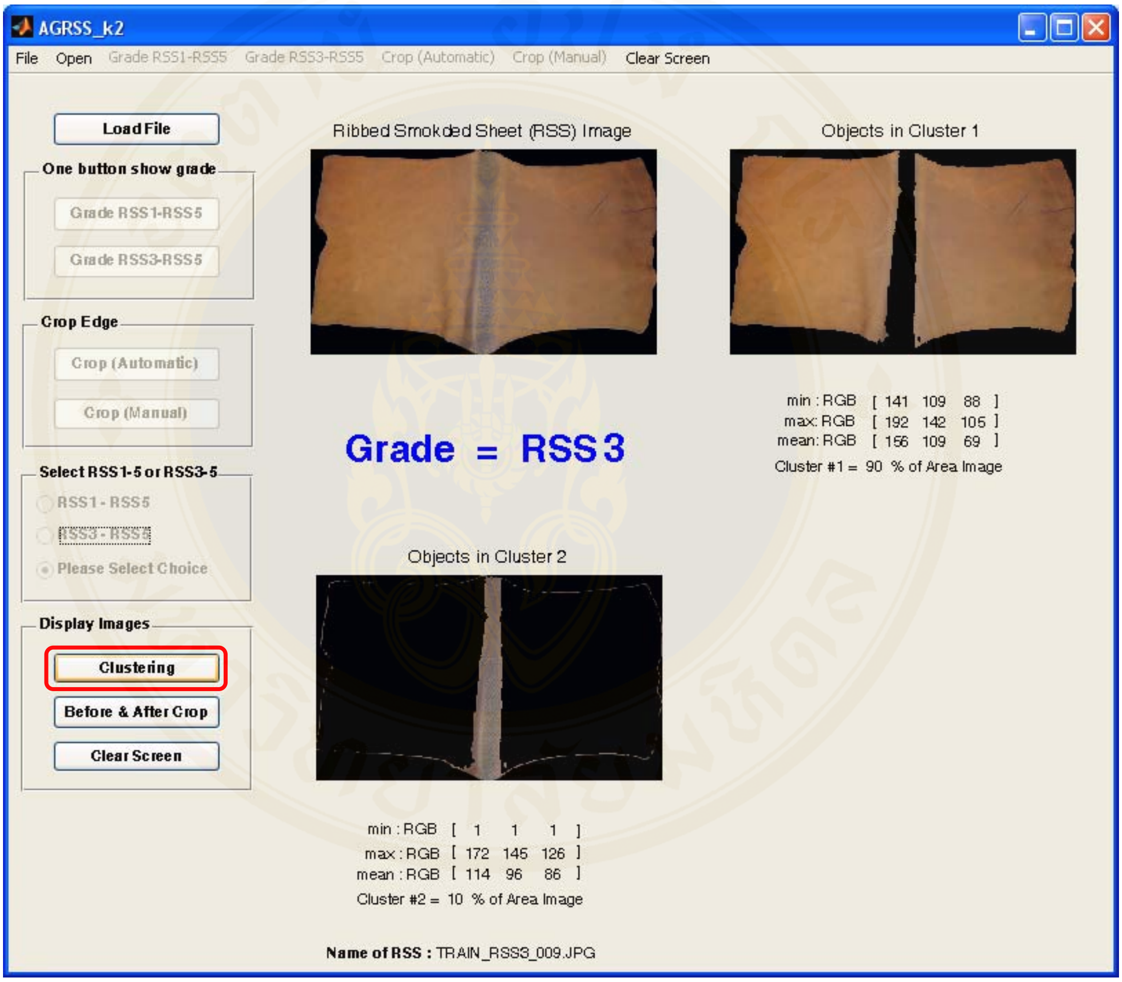


Figure 4.25 Graphic user interface (GUI) of Clustering button

4.3.7 Before & After Crop button

This button will show 3 RSS (Ribbed Smoked Sheet) images start from original RSS image, crop edge image, and crop fit RSS area image. The system will show these image when the users press this button.

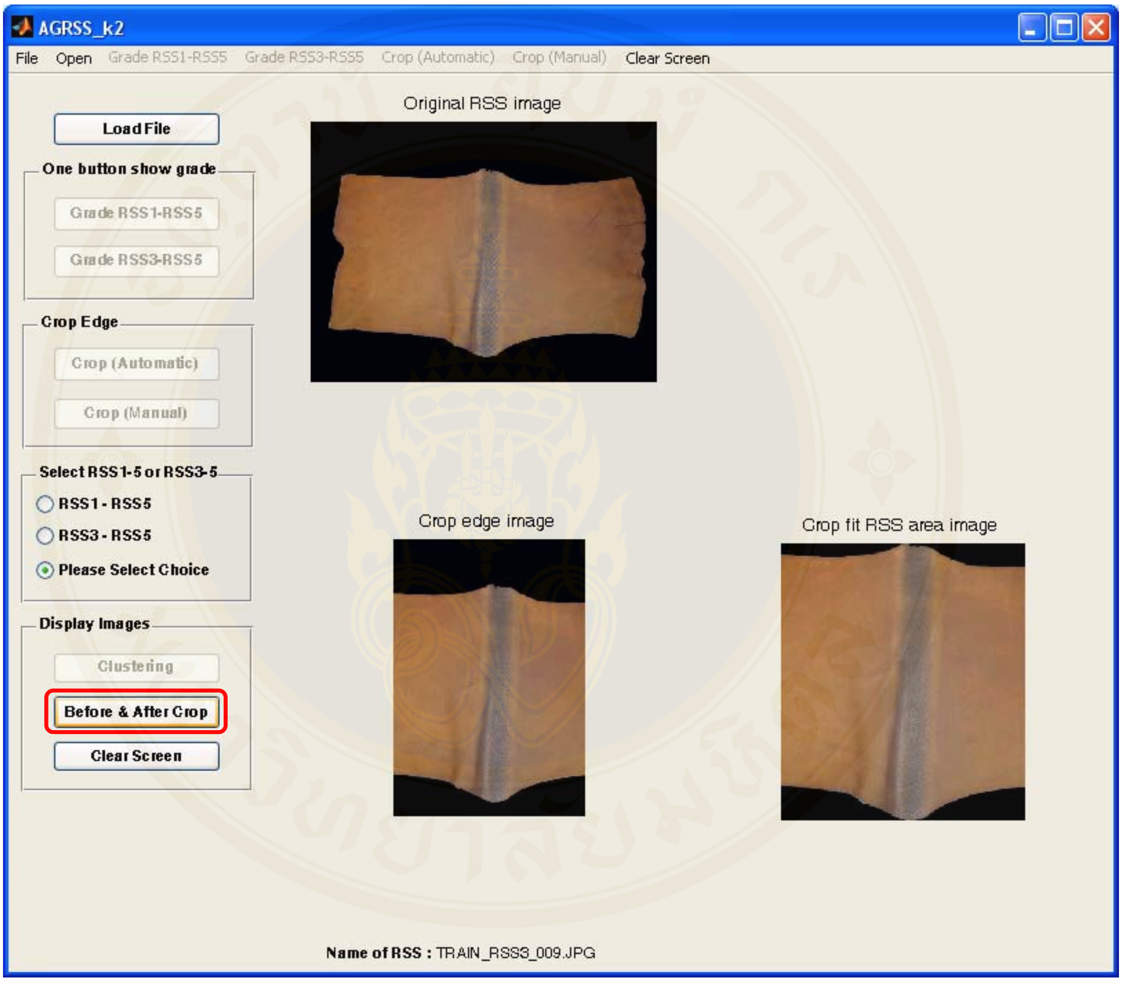


Figure 4.26 Graphic user interface (GUI) of Before & After Crop button

4.3.8 Clear Screen button

This button will clear every images objects on the GUI (graphic user interface) prepare for the users to load new file or see the previous image processing such as clustering image or image before & after crop. The previous image processes will be cleared from the front of GUI by pressing this button.

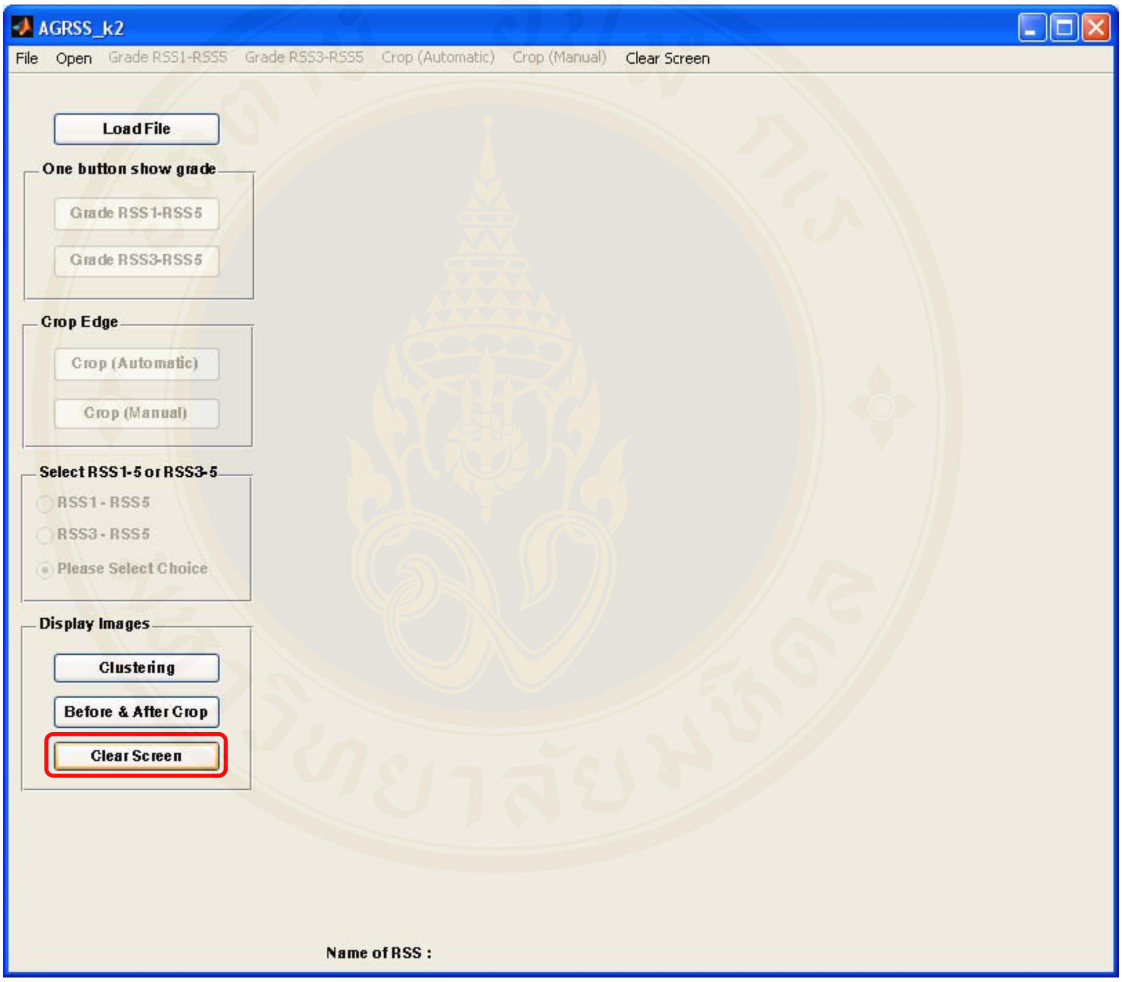


Figure 4.27 Graphic user interface (GUI) of Clear Screen button

4.4 System training phase

The system training phase consists of 4 main processes as shown in Figure 4.28. The processes composed of image acquisition, segmentation, clustering, and create RSS (Ribbed Smoked Sheet) range from mean RGB (red, green, blue) color space data.

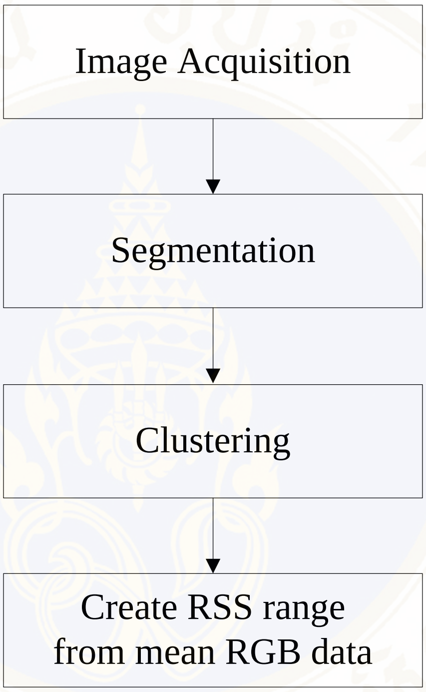


Figure 4.28 Process of system training phase

The input to the system is a top view of Ribbed Smoked Sheet (RSS) image from digital camera in controlled environment box. RSS image must cover rubber sheet up at least 80 % of the rubber image area and background area may be included.

The user insert RSS image with dark background into Analysis Grade of Ribbed Smoked Sheet (AGRSS) system. The image is 3701 x 2304 pixels and resize to 640 x 480 pixels in dimension with 24-bit color, encoded in JPEG file format.

AGRSS system will separate RSS area and background by convert RGB image to grayscale then calculate threshold value to produce binary image. The system will remove noise in binary image and find coordinates of RSS image. After that the system will crop coordinates fit RSS area.

AGRSS system will use new image fit RSS area to clustering by convert RGB color space to L*a*b* color space (The L*a*b* color space consists of a luminosity layer 'L*', a chromaticity-layer 'a*' that indicates where the color falls along the red-green axis, and a chromaticity-layer 'b*' that indicates where the color falls along the blue-yellow axis.) and classify color in a*b* space using K-means clustering. The system will label every pixel in the image using the results from K-means clustering and create RSS image segmentation by color.

The clustering process will represent two clusters images and show objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible. The system will calculate and displayed minimum, maximum and mean RGB information from the most clustering area and next cluster area then collected and classify RSS range from RGB data for each RSS grades in training phase.

The training phase use original images in control environment to determine range color level of each RSS grades. A total of 398 images made up a training set for RSS1, RSS2, RSS3, RSS4 and RSS5. Figure 4.29 represents overall of RGB range for RSS five grades from training set. Figure 4.30 – Figure 4.34 and Table 4.1 – Table 4.5 show the details of decision RSS grades tables. For a list of the parameter used by the decision RSS grades table in Table 4.1 - Table 4.5, see the Table 3.5 in chapter III Methodology, page 66.

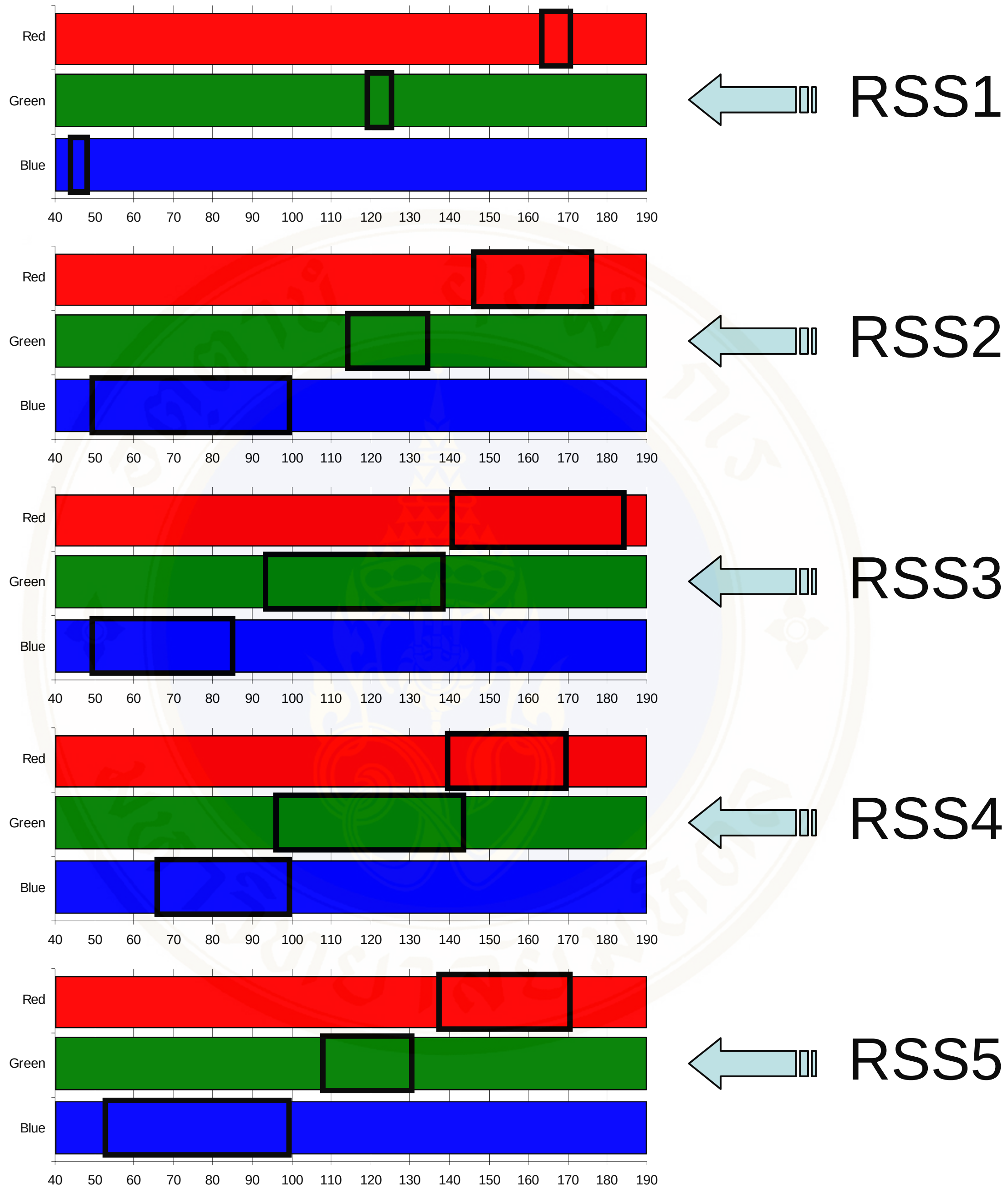


Figure 4.29 RGB range of RSS grades : RSS1 – RSS5

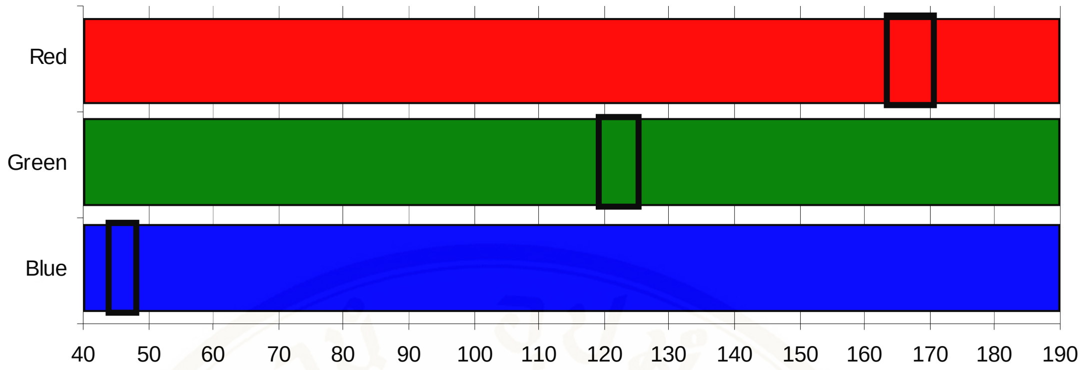


Figure 4.30 RGB range of RSS grades : RSS1

Table 4.1 Decision RSS grades table : RSS1

meanB1	**meanG1**	**meanR1**
44-48	119-126	163-171

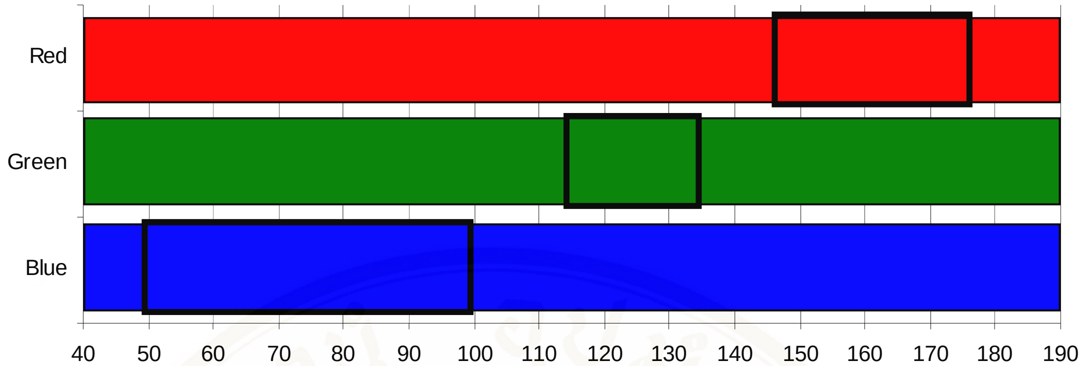


Figure 4.31 RGB range of RSS grades : RSS2

Table 4.2 Decision RSS grades table : RSS2

(a)

meanB1	meanG1	meanR1
49-51	125-127	165-167
52-53	114-115	146-176
	116-117	146-161
		164-176
	118-134	146-176
54-55	117-123	160-165
56	114	146-176
	115-116	146-160
		166-176
	119-134	146-176
57	119-120	158-164
	121-126	164-167
58	121-128	164-170
59	122-123	160-162
	124-129	167-169
60	120-122	164-165
	123	158-160
	124	162-164
61	125-131	164-165
	132-134	174-176
62	121	159-161
	122	162-163
	127	163-165

(b)

meanB1	meanG1	meanR1
63	122-126	156-157
		162-163
64-65	122-129	162-163
66-67	116-122	146-148
		163-164
	123-125	157-158
68	118-122	149-153
	123-128	160-162
69	116-119	147-149
	122-125	152-158
	126-130	162-164
70-71	118-123	147-153
	124-127	157-159
	128-130	159-161
72-73	118-122	146-149
	123-129	160-161
	130-132	161-163
74-75	126-129	155-159
84-85	127-128	150-151
88-89	130-132	153-155
90-99	130-132	152-154

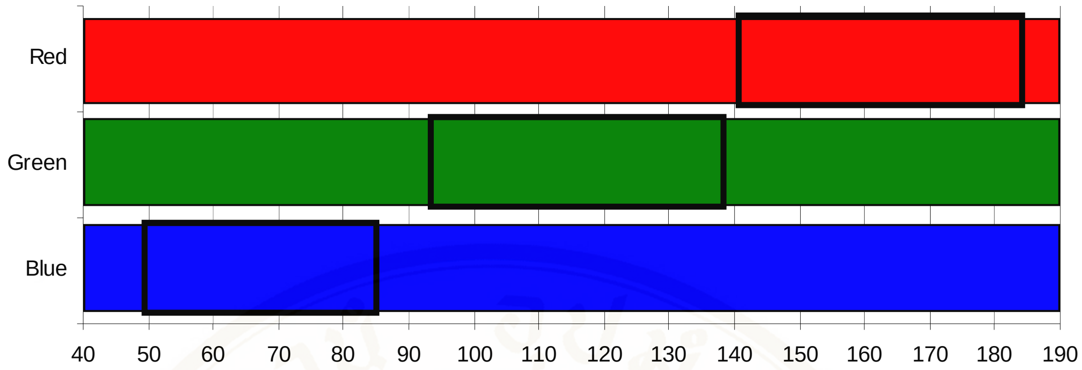


Figure 4.32 RGB range of RSS grades : RSS3

Table 4.3 Decision RSS grades table : RSS3

(a)

meanB1	meanG1	meanR1
49-51	93-124	141-184
	125-127	141-164
		168-184
	128-138	141-184
54-55	93-116	141-184
	117-123	93-159
		166-184
	124-138	141-184
57	93-115	141-184
	116-117	141-156
		161-184
	118	141-184
	119-120	141-157
		169-184
	121-126	141-163
		168-184
	127-138	141-184

(b)

meanB1	meanG1	meanR1
58	93-108	141-184
	109-112	141-160
		164-184
	113-114	141-184
	115-120	141-162
		169-184
	121-128	141-163
		171-184
	129-138	141-184
59	93-110	141-184
	111-114	141-161
		164-184
	115-117	141-184
	118-119	141-163
		166-184
	120-121	141-161
		164-184
	122-123	141-159
		163-184
	124-129	141-166
		170-184
	130-138	141-184

Table 4.3 Decision RSS grades table : RSS3 (Continued)

(c)

meanB1	meanG1	meanR1
60	93-111	141-184
	112-116	141-152
		156-164
		170-184
	117-118	141-184
	119	141-160
		166-184
	120-122	141-163
		166-184
	123	141-157
		161-184
	124	141-161
		165-184
	125-138	141-184
61	93-110	141-184
	111-113	141-158
		163-184
	114-120	141-164
		172-184
	121-124	141-162
		165-184
	125-131	141-163
		168-184
	132-134	141-173
		177-184
	135-138	141-184

(d)

meanB1	meanG1	meanR1
62	93-107	141-184
	108-110	141-158
		162-184
	111-113	141-184
	115-120	141-164
		170-184
	121	141-158
		162-184
	122	141-161
		164-184
	123-124	141-184
	125	141-162
		165-184
	126	141-166
		169-184
	127	141-162
		166-184
	128	141-163
		166-184
	129-138	141-184
63	93-112	141-184
	113-115	141-160
		164-184
	116	141-184
	117-118	141-155
		161-184
	119-121	141-160
		163
		167-184
	122-126	141-155
		158-161
		167-184
	127-138	141-184

Table 4.3 Decision RSS grades table : RSS3 (Continued)

(e)

meanB1	meanG1	meanR1
64-65	93-112	141-184
	113-116	141-156
		164-184
	117-118	141-160
		165-184
	119-120	141-159
		161-184
	122-129	141-161
		164-165
		167-184
	130-138	141-184
66-67	93-101	141-184
	102-104	141-153
		159-184
	105-108	141-154
		158-184
	109-115	141-157
		160
		165-184
	116-122	141-145
		149-156
		165-184
	123-125	141-156
		159-184
	126-130	141-160
		165-166
		170-184
	131-133	141-159
		163-184
	134-138	141-184

(f)

meanB1	meanG1	meanR1
68	93-107	141-184
	108-109	141-158
		162-184
	110-112	141-184
	113-117	141-155
		159-161
		165-184
	118-122	93-148
		154-184
	123-128	141-159
		163-164
		167-184
	129-138	141-184
69	93-102	141-184
	103-107	141-156
		160-184
	108-115	141-153
		156-159
		166-184
	116-119	141-146
		150-184
	120-121	141-184
	122-125	141-151
		159-164
		169-184
	126-130	141-161
		165-166
		170-184
	131-138	141-184

Table 4.3 Decision RSS grades table : RSS3 (Continued)

(g)

meanB1	meanG1	meanR1
70-71	93-103	141-184
	104-110	141-157
		160-184
	111-117	141-150
		156-159
		165-184
	118-123	141-146
		154-155
		161-184
	124-127	141-156
		160-184
	128-130	141-158
		162-184
	131-138	141-184
72-73	93-96	141-184
	97-101	141-146
		150-184
	102-105	141-184
	106-110	141-149
		152-154
		160-184
	111-117	141-184
	118-122	141-145
		150
		154-184
	123-129	141-155
		162
		166-184
	130-132	141-160
		164-184
	133-138	141-184

(h)

meanB1	meanG1	meanR1
74-75	108-113	152-154
	116-122	157-159
	123-125	153-154
76-77	115-120	155-158
	121-123	159-163
78-79	93-95	141-143
	96-99	145-147
	118-120	154-157
80-81	119-123	158-159
84-85	119-120	150-152

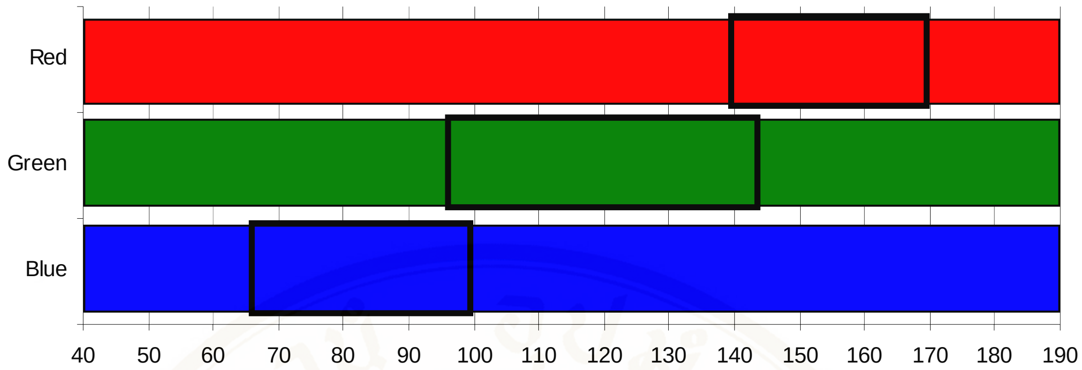


Figure 4.33 RGB range of RSS grades : RSS4

Table 4.4 Decision RSS grades table : RSS4

(a)

meanB1	meanG1	meanR1
66-67	102-104	154-158
	105-108	155-157
	109-115	161-164
	131-133	160-162
68	108-109	159-161
	113-117	162-164
69	103-107	157-159
	108-115	160-165
70-71	104-110	158-159
	111-117	160-164
	118-123	156-158
72-73	97-101	147-149
	106-110	150-151
		155-159
	118-122	151-153
	123-129	156-159

(b)

meanB1	meanG1	meanR1
74-75	96-107	139-169
	108-113	139-151
		155-169
	114-115	139-169
	116-122	139-149
		162-169
	123-125	139-152
		155-169
	126-129	139-154
		160-169
	130-143	139-169
76-77	96-114	139-169
	115-120	139-149
		154
		159-169
	121-123	139-154
		157-158
		164-169
	124-143	139-169

Table 4.4 Decision RSS grades table : RSS4 (Continued)

(b)

meanB1	meanG1	meanR1
78-79	96-99	139-144
		148-169
	100-117	139-169
	118-120	139-147
		153
		158-169
	121-143	139-169
80-81	96-118	139-169
	119-123	139-145
		149-157
		160-169
	124-127	139-157
		160-169
	128-143	139-169
82-83	96-115	139-169
	116-119	139-144
		149-169
	120-143	139-169
84-85	96-112	139-169
	113-118	139-143
		150-169
	119-120	139-147
		153-169
	121-126	139-169
	127-128	139-149
		152-169
	129-143	139-169

(d)

meanB1	meanG1	meanR1
86-87	96-111	139-169
	112-114	140-169
	115-116	139-169
	117-120	139-144
		150-169
	121-124	139-146
		148-150
		154-169
	125-143	139-169
88-89	96-129	139-169
	130-132	139-152
		156-169
	133-143	139-169
90-99	96-116	139-169
	117-126	139-140
		144-169
	127-129	139-169
	130-132	139-151
		155-169
	133-139	139-169

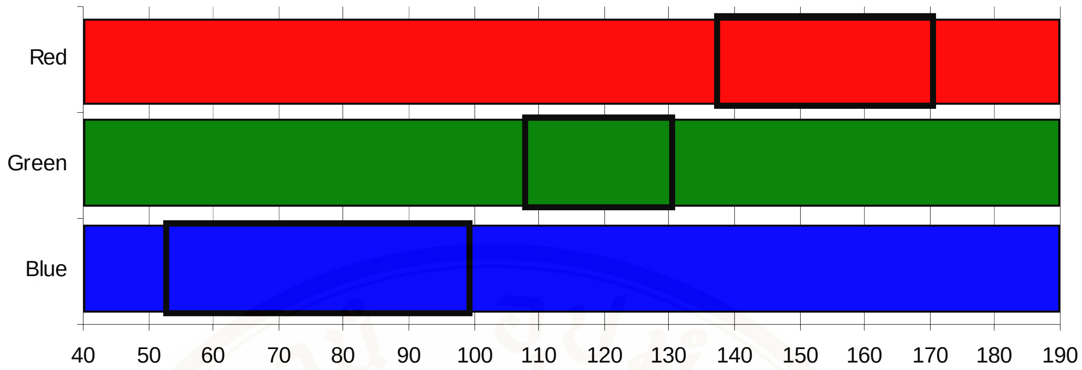


Figure 4.34 RGB range of RSS grades : RSS5

Table 4.5 Decision RSS grades table : RSS5

(a)

meanB1	meanG1	meanR1
52-53	116-117	162-163
56	115-116	161-165
57	116-117	157-160
	119-120	165-168
58	109-112	161-163
	115-120	163-168
59	111-114	162-163
	118-119	164-165
	120-121	162-163
60	112-116	153-155
		165-169
	119	161-162
61	111-113	159-162
	114-120	165-171
	121-124	163-164
	125-131	166-167
62	108-110	159-161
	115-120	165-169
	125	163-164
	126	167-168
	128	164-165

(b)

meanB1	meanG1	meanR1
63	113-115	161-163
	117-118	156-160
	119-121	161-162
		164-166
	122-126	164-166
64-65	113-116	157-163
	117-118	161-164
	119-120	160
	122-129	166
66-67	109-115	158-159
	116-122	157-162
	126-130	161-164
		167-169
68	113-117	156-158
	123-128	165-166
69	108-115	154-155
	122-125	165-168
	126-130	167-169
70-71	111-117	151-155
	118-123	159-160
72-73	123-129	163-165

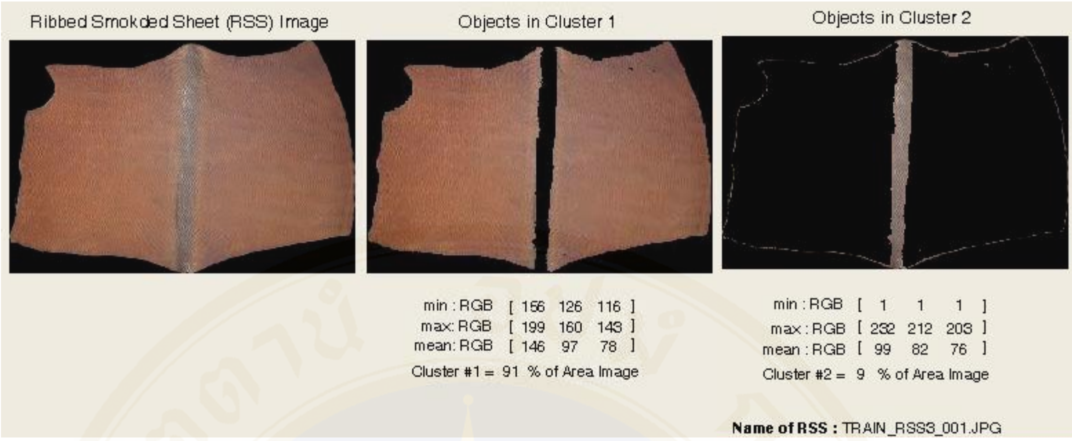
Table 4.5 Decision RSS grades table : RSS5 (Continued)

(c)

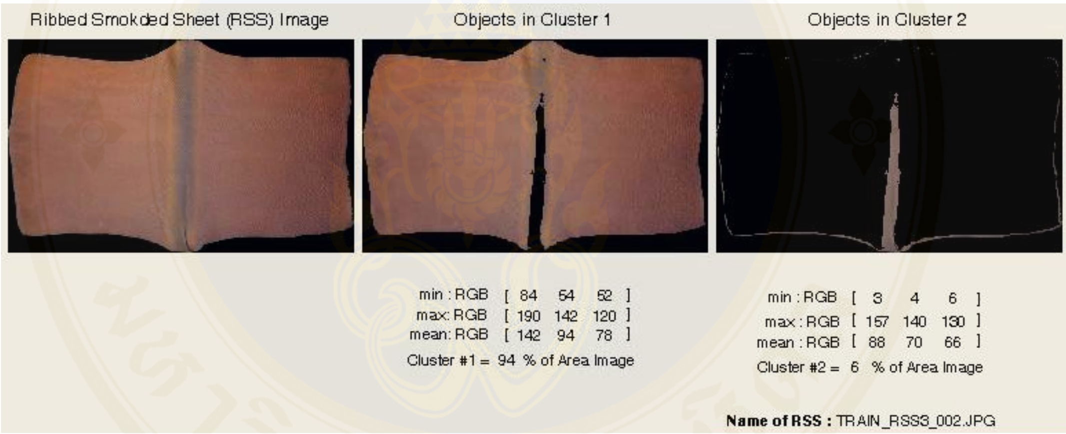
meanB1	**meanG1**	**meanR1**
74-75	116-122	150-156
		160-161
76-77	115-120	150-153
	121-123	155-156
78-79	118-120	148-152
80-81	119-123	146-148
	124-127	158-159
82-83	116-119	145-148
84-85	113-118	144-149
	119-120	148-149
86-87	112-114	137-139
	117-120	145-149
	121-124	147
		151-153
90-99	117-126	141-143

In Figure 4.35 was shown the three examples of RSS images for training set data. Figure 4.35a was the image name TRAIN_RSS3_001.JPG, Figure 4.35b was the image name TRAIN_RSS3_002.JPG, and in Figure 4.35c was the image name TRAIN_RSS3_003.JPG.

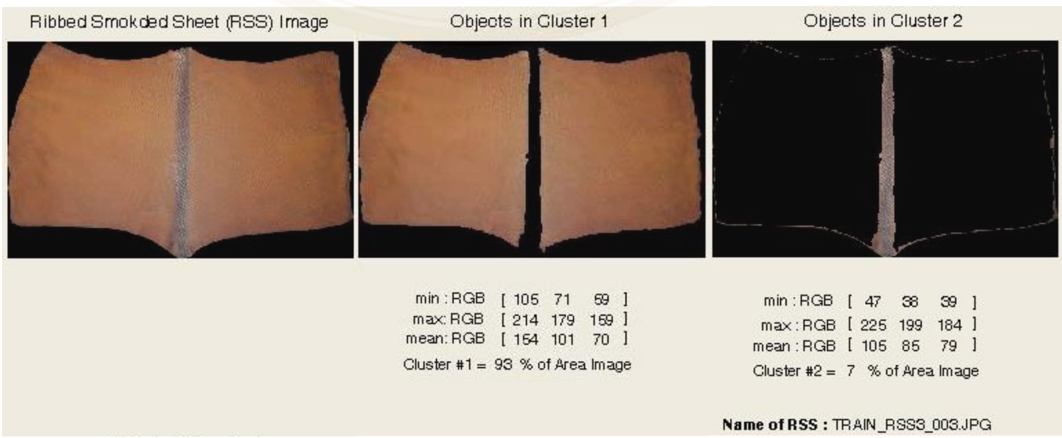
See these three examples of RSS images training set data in the part of color RGB information in Table 4.6 and the process time's information in Table 4.7. For a list of the parameter used by the training set data in Table 4.6 and Table 4.7, see the Table 3.5 in chapter III Methodology, page 66.



(a)



(b)



(c)

Figure 4.35 Example of RSS images for training

Table 4.6 Example of RSS images training set data (color RGB information)

IMAGES	TRAIN_RSS3_001.JPG	TRAIN_RSS3_002.JPG	TRAIN_RSS3_003.JPG
Value	**Result**	**Result**	**Result**
minR1	156	84	105
minG1	126	54	71
minB1	116	52	59
maxR1	199	190	214
maxG1	160	142	179
maxB1	143	120	159
minR2	1	3	47
minG2	1	4	38
minB2	1	6	39
maxR2	232	157	225
maxG2	212	140	199
maxB2	203	130	184
meanR1	146	142	154
meanG1	97	94	101
meanB1	78	78	70
meanR2	99	88	105
meanG2	82	70	85
meanB2	76	66	79
Clust1_Area	91	94	93
Clust2_Area	9	6	7
result	**3**	**3**	**3**

Table 4.7 Example of RSS images training set data (process time's information)

IMAGES	TRAIN_RSS3_001.JPG	TRAIN_RSS3_002.JPG	TRAIN_RSS3_003.JPG
Using Time	**Process Time (seconds)**	**Process Time (seconds)**	**Process Time (seconds)**
tLoad	0.04	0.04	0.04
tCrop	0.35	0.34	0.35
tKmean	10.91	9.49	11.63
tSum	**11.30**	**9.87**	**12.02**

4.5 System testing phase

The testing phase contain with 4 main processes as show in Figure 4.36. They are image acquisition, segmentation, clustering and calculation of RSS (Ribbed Smoked Sheet) grades.

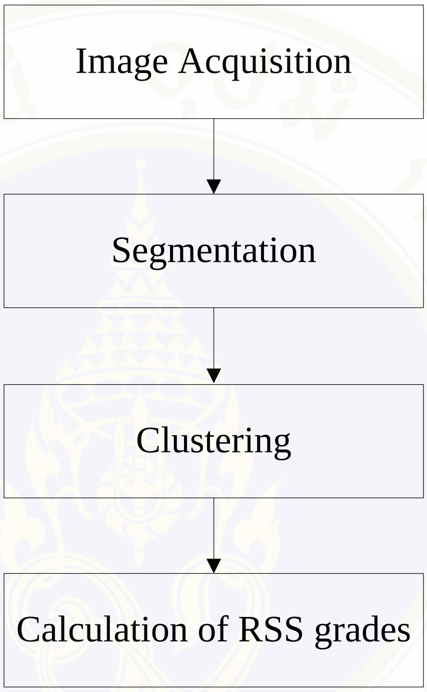


Figure 4.36 Process of system testing phase

The input to the system is a top view of Ribbed Smoked Sheet (RSS) image from digital camera in controlled environment box. The user insert RSS image with dark background into the Analysis Grade of Ribbed Smoked Sheet (AGRSS) system.

The image is 640 x 480 pixels in dimension with 24-bit color, encoded in JPEG file format. AGRSS system will separate RSS area and background. The system will find coordinates of RSS image then crop coordinates fit RSS area.

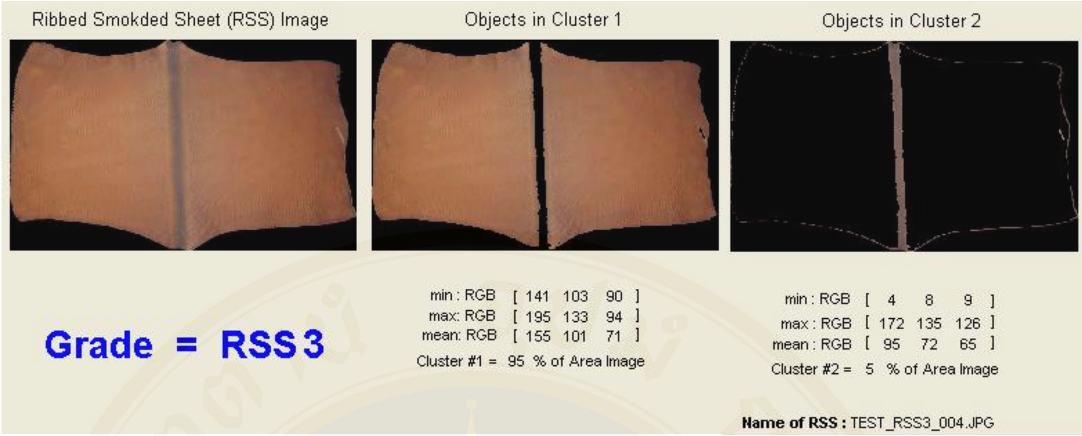
AGRSS system will use new image fit RSS area to clustering. The clustering process will represent two clusters images on graphic user interface (GUI). The system will calculate and displayed minimum, maximum and mean RGB information from the most clustering area and next cluster area to compared and classify RSS grade from RSS decision grades table.

AGRSS system will retrieve mean RGB from the most cluster area and compare in the decision RSS grades table. The system will start compare the mean of blue value and then compare with mean of green and red values after that displays of RSS grade results.

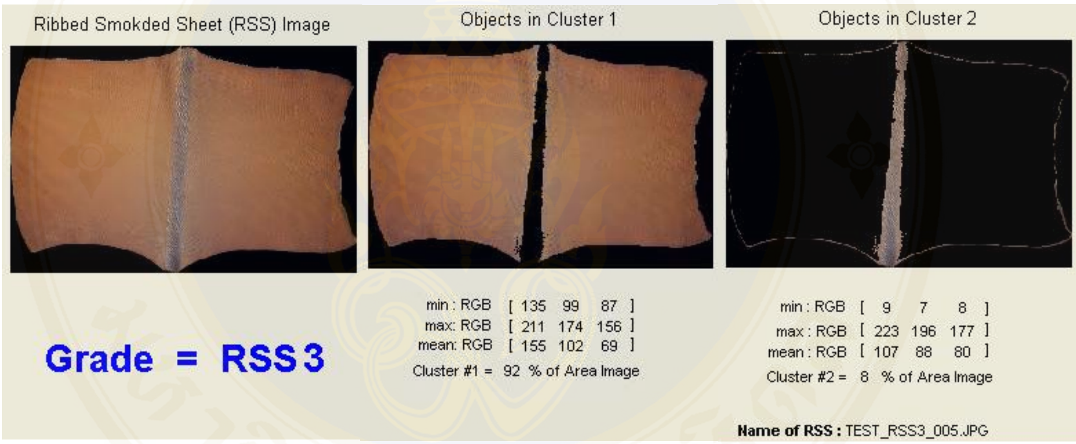
System testing phase used original image in control environment to calculate RSS (Ribbed Smoked Sheet) grade. A total of 398 images made up a test dataset for RSS1, RSS2, RSS3, RSS4 and RSS5.

In Figure 4.37 we see the three examples of RSS images for testing set data. Figure 4.37a was the image name TEST_RSS3_001.JPG, Figure 4.37b was the image name TEST_RSS3_002.JPG, and in Figure 4.37c was the image name TEST_RSS3_003.JPG.

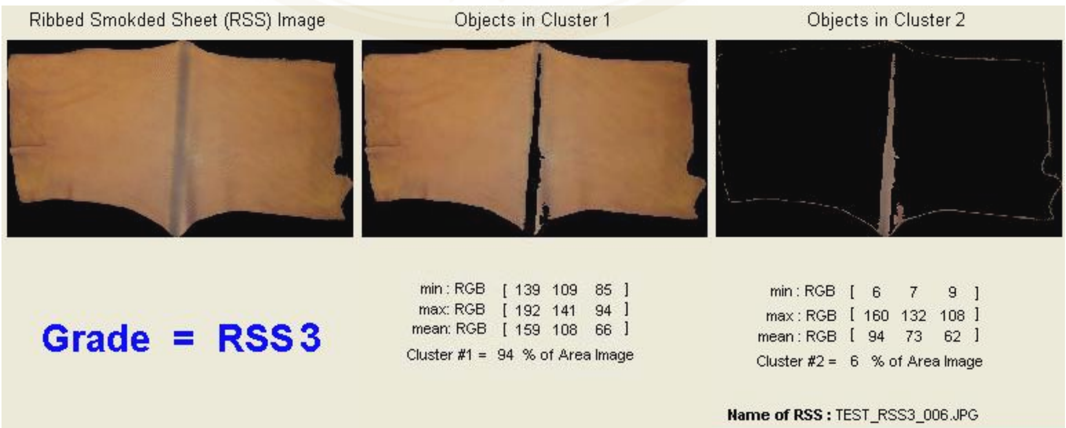
See these three examples of RSS images test dataset in the part of color RGB information in Table 4.8 and the process time's information in Table 4.9. For a list of the parameter used by the test dataset in Table 4.8 and Table 4.9, see the Table 3.5 in chapter III Methodology, page 66.



(a)



(b)



(c)

Figure 4.37 Example of RSS images for testing

Table 4.8 Example of RSS images testing set data (color RGB information)

IMAGES Value	TEST_RSS3_004.JPG Result	TEST_RSS3_005.JPG Result	TEST_RSS3_006.JPG Result
minR1	141	135	139
minG1	103	99	109
minB1	90	87	85
maxR1	195	211	192
maxG1	133	174	141
maxB1	94	156	94
minR2	4	9	6
minG2	8	7	7
minB2	9	8	9
maxR2	172	223	160
maxG2	135	196	132
maxB2	126	177	108
meanR1	155	155	159
meanG1	101	102	108
meanB1	71	69	66
meanR2	95	107	94
meanG2	72	88	73
meanB2	65	80	62
Clust1_Area	95	92	94
Clust2_Area	5	8	6
result	**3**	**3**	**3**

Table 4.9 Example of RSS images testing set data (process time's information)

IMAGES	TEST_RSS3_004.JPG	TEST_RSS3_005.JPG	TEST_RSS3_006.JPG
Using Time	**Process Time (seconds)**	**Process Time (seconds)**	**Process Time (seconds)**
tLoad	0.05	0.05	0.04
tCrop	0.41	0.41	0.41
tKmean	11.85	10.96	11.28
tSum	**12.31**	**11.42**	**11.73**

CHAPTER V
EXPERIMENTAL RESULTS

This chapter discusses about processes of Ribbed Smoked Sheet (RSS) images which used in this experimental. RSS images which acquire into this work. The result of experimented in the testing phase. The testing phase is experiment for separated unknown RSS grade to RSS1, RSS2, RSS3, RSS4 and RSS5. The last section discusses about the problems found in the work and the performance of the system based on experimental outcomes. Some solutions of the problems are also presented.

5.1 Data images acquisition

All of original images in this work were taken by a digital camera. Before collecting images we have to prepare a completely dark box simulation. The sizes of dark box were 120 cm. (width) x 150 cm. (height) x 70 cm (depth) and the structure was shown in Figure 5.1. This dark box has to set in the indoor area to avoid various sunlight conditions and have to choose dark background and dark environment in this box. Next step was to set two lamps which were white light (100 watts) on the top of the dark box to control light. Set up camera image resolution for the best quality with the size 3071 x 2304 pixels. Then fix digital camera at the top center of the dark box and parallel camera lens to the Ribbed Smoked Sheet (RSS). The distances far from RSS to camera lens were about 150 centimeters (60 inches) for covering all area of RSS. Flash built in camera must disable because it will effect to the data images. Prepare RSS near the area of the dark box. After that, insert RSS into the dark box and taken an image. Remove RSS that was taken already out of the dark box and insert new one until out of RSS.

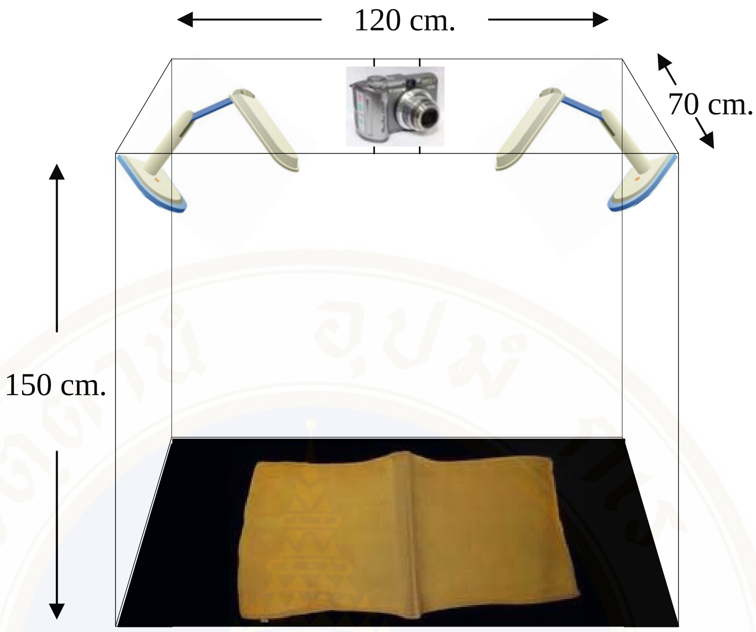


Figure 5.1 Size of dark box simulation

5.1.1 Digital camera specification

In this work used digital camera brand Canon, model PowerShot A620 was shown in Figure 5.2. It was taken all of original Ribbed Smoked Sheet (RSS) images. The detail specifications are showing in the Table 5.1.

Figure 5.2 Canon PowerShot A620

Table 5.1 The specification of digital camera

Contents	Description
Type of Camera	Compact digital still camera with built-in flash, 4x Optical/ 4x Digital/ 16x Combined Zoom.
Image Capture Device	Type: 7.1 Megapixel, 1/1.8 inch charge coupled device (CCD) Total Pixels: Approx. 7.4 Megapixels Effective Pixels: Approx. 7.1 Megapixels
Lens	Focal Length: 7.3 - 29.2mm f/2.8-4.1 (35mm film equivalent: 35-140mm) Digital Zoom: 4x Focusing Range: Normal: 1.5 ft./45cm-infinity Macro: 0.4 in.-1.5 ft./1-45cm (WIDE), 9.8 in. - 1.5 ft./25-45cm (except WIDE) Autofocus System: TTL Autofocus (continuous/single)
Viewfinder & Monitor	Viewfinder: Real-image optical zoom viewfinder LCD Monitor: 2.0 inch low-temperature polycrystalline silicon TFT color LCD (vari-angle) LCD Pixels: Approx. 115,000 pixels LCD Coverage: 100%
Aperture and Shutter	Maximum Aperture: f/2.8 (W) - f/4.1 (T) Shutter Speed: 15-1/2000 sec.; Slow shutter operates with noise reduction when manually set at 1.3-15 sec.
Exposure Control	ISO Sensitivity: AUTO, ISO 50/100/200/400 equivalent Light Metering: Evaluative, Center-weighted average, Spot*. Method: *Metering frame with Spot Mode fixed to center or linked to AF frame. Exposure Control Method: Program AE, Shutter-priority AE, Aperture-priority AE, Manual. AE Lock is available. Exposure Compensation: +/- 2 stops in 1/3-stop increments.

Table 5.1 The specification of digital camera (Continued)

Contents	Description
White Balance	White Balance Control: Auto, Preset (Daylight, Cloudy, Tungsten, Fluorescent, Fluorescent H, Underwater), Custom.
Flash	Built-in Flash: - Operation Modes: Auto, On/Off, Red-Eye Reduction On/Off, Slow Synchro. Flash Range: - Normal: 1.5-13.8 ft./45cm - 4.2m (W), 1.5-9.8 ft./45cm - 3.0m (T) - Macro: 0.8-1.5 ft./25-45cm (when sensitivity is set to ISO 100 equivalent) Recycling Time: Approx. 10 sec. or less (battery voltage=6.0V) Flash Exposure Compensation: +/- 2 stops in 1/3-stop increments
Shooting Specifications	Shooting Modes: Auto; Creative: P, Av, Tv, M, C; Image: Portrait, Landscape, Night Scene, Special Scene (Foliage, Snow, Beach, Fireworks, Underwater, Indoor, Kids & Pets, Night Snapshot), My Colors, Stitch Assist, Movie Photo Effects: Vivid, Neutral, Low Sharpening, Sepia, Black & White. Self-Timer: Activates shutter after an approx. 2 sec./10 sec. delay, Custom. Wireless Control: Not available. Continuous Shooting: Approx. 1.9 fps.

Table 5.1 The specification of digital camera (Continued)

Contents	Description
Image Storage	Storage Media: SD Memory Card, MultiMediaCard. File Format: Design rule for camera file system, DPOF Version 1.1 (Digital Print Order Format). Image Compression: Still Image: Exif 2.2 (JPEG) Movie: AVI (Image: Motion JPEG; Audio: WAVE (Monaural)). JPEG Compression Mode: Normal, Fine, SuperFine. Number of Recording Pixels: Still Image: 3,072 x 2,304 (Large), 2,592 x 1,944 (Medium 1), 2,048 x 1,536 (Medium 2), 1,600 x 1,200 (Medium 3), 640 x 480 (Small) Movie: 640 x 480/320 x 240 (30 fps/15 fps), 320 x 240 (1 min. at 60 fps), 160 x 120 (3 min. at 15 fps)
Playback Specifications	Playback Modes File: - Still Image: Single, Magnification (approx. 2x-10x), Jump, Auto Rotate, Rotation, Histogram, Index (9 thumbnails), Sound Memos, Slide Show. - Movie: Normal Playback, Special Playback, Slow Motion, Editing.
Erasing Specifications	Erase Modes: - Still Image: Single image, All images - Movie: Part of movie, All of movie
Interfaces	Computer Interface: USB 2.0 Hi-Speed (mini-B PTP). Video Out: NTSC/PAL. Audio Out: Monaural. Other: Memory card slot. Direct connection to Canon CP, SELPHY, PIXMA Photo Printers and PictBridge compatible printers via camera's USB cable.

Table 5.1 The specification of digital camera (Continued)

Contents	Description
Power Supply	Power Source: 1. AA-size Alkaline Battery (x4) 2. Rechargeable AA-size NiMH Battery (x4) 3. AC Adapter Kit ACK600 Shooting Capacity: Still Image: approx. 350 shots (AA-size Alkaline Battery), approx. 500 shots (AA-size NiMH Battery) *[1] Playback Time: Approx. 960 min. (AA-size Alkaline Battery), approx. 960 min. (AA-size NiMH Battery)*
Physical Specifications	Operating Temperature: 32-104° F/0-40° C Operating Humidity: 10-90% Dimensions (WxHxD): 4.13 x 2.60 x 1.93 in /104.8 x 66.0 x 49.1mm Weight: Approx. 8.29 oz./235g (camera body only)

Note: *[1]CIPA Standard, *Canon Standard Test Method, 22 display languages provided (English, German, French, Dutch, Danish, Finnish, Italian, Norwegian, Swedish, Spanish, Simplified/Traditional Chinese, Japanese, Russian, Portuguese, Korean, Greek, Polish, Czech, Hungarian, Turkish, Thai).

All data are based on Canon's Standard Test Method. Canon, PowerShot and DIGIC are registered trademarks, and SELPHY and PIXMA are trademarks, of Canon Inc. in the United States and may also be registered trademarks or trademarks in other countries. Microsoft, Windows and the Windows logo are trademarks or registered trademarks of the Microsoft Corporation in the United States and/or other countries. Mac is a trademark of Apple Computer, Inc., registered in the United States and other countries. All other products and brand names are registered trademarks, trademarks or service marks of their respective owners.

Table 5.2 The number of selected images for training and testing phases

Phase	Number of images
Training	398
Testing	398
Total	**796**

5.2 Testing phase

Images input to the system are a top view of Ribbed Smoked Sheet (RSS) images from digital camera in dark box simulation. The user insert RSS image with dark background and dark environment into Analysis Grade of Ribbed Smoked Sheet (AGRSS) system. The image is 640 x 480 pixels in JPEG file format. AGRSS system will separate RSS area out off background. The system will find coordinates of RSS image and then crop coordinates fit RSS area. AGRSS system use new image fit RSS area for clustering process. The clustering process will represent new clusters images on graphic user interface (GUI). The system will calculate and displayed of minimum, maximum and mean of RGB (red, green, blue) information of the most clustering area and next cluster area to compared and classify RSS grade from RSS decision grades table. AGRSS system will call mean of RGB data from the new cluster image and compare in the decision RSS grades table. The system will start compare from the mean of blue value and then compare using mean of green and red values for RGB color space. After that, the system displays of RSS grades result. System testing phase used a total of 398 images in control environment made up a test dataset and classify RSS grade to RSS1, RSS2, RSS3, RSS4 and RSS5. For L*a*b* color space (The L*a*b* color space consists of a luminosity layer 'L*', a chromaticity-layer 'a*' that indicates where the color falls along the red-green axis, and a chromaticity-layer 'b*' that indicates where the color falls along the blue-yellow axis.) data testing, the system will distinguish colors from one another by ignore variations in brightness (‘L*’). AGRSS will start compare value from the mean of ‘a*’ value and then compare using mean of ‘b*’ values.

This experiment compares the test dataset between a*b* color space data and RGB color space data by define k cluster equal to 1 and 2. The testing results of images calculation are show that using mean of RGB color space data is more accurate than using mean of a*b* color space data. This work was tested with 398 RSS images. The summation results of comparing between “a*b*” and “RGB” color space represented in Table 5.3.

Table 5.3 Comparing results between “a*b*” and “RGB” color space using k cluster equal to 1 and 2

RSS grades	**k cluster**	**k = 1 (100%)**		**k = 2 (100%)**	
	color space	**a*b***	**RGB**	**a*b***	**RGB**
RSS 1	✓	87.50 %	100 %	100 %	100 %
	✗	12.50 %	0 %	0 %	0 %
RSS 2	✓	36.07 %	40.98 %	62.30 %	68.85 %
	✗	63.93 %	59.02 %	37.70 %	31.15 %
RSS 3	✓	52.14 %	61.54 %	58.12 %	80.34 %
	✗	47.86 %	38.46 %	41.88 %	19.66 %
RSS 4	✓	68.03 %	82.79 %	86.89 %	87.70 %
	✗	31.97 %	17.21 %	13.11 %	12.30 %
RSS 5	✓	41.11 %	34.44 %	44.44 %	77.78 %
	✗	58.89 %	65.56 %	55.56 %	22.22 %
Total of RSS1-RSS5	✓	**52.76 %**	**59.55 %**	**65.33 %**	**80.65 %**
	✗	**47.24 %**	**40.05 %**	**34.67 %**	**19.35 %**

The next experiment testing is compare the result by define k cluster equal to 1, 2, 3 and 4 was shown in Table 5.4. The best testing results of images calculation is define k cluster equal to 2 and presented details in Table 5.5 – 5.11. This work was tested with 398 images and 80.65% accuracy rate of calculation of RSS grades. More details are presented in the following subsection.

Table 5.4 Comparing k cluster's result in testing phase images group by RSS1 – RSS5

RSS grades	Result	k = 1 (100%)	k = 2 (100%)	k = 3 (100%)	k = 4 (100%)
RSS 1	✓	100 %	100 %	100 %	50 %
	✗	0 %	0 %	0 %	50 %
RSS 2	✓	40.98 %	68.85 %	21.31 %	22.95 %
	✗	59.02 %	31.15 %	78.69 %	77.05 %
RSS 3	✓	61.54 %	80.34 %	68.38 %	40.17 %
	✗	38.46 %	19.66 %	31.62 %	59.83 %
RSS 4	✓	82.79 %	87.70 %	70.49 %	72.95 %
	✗	17.21 %	12.30 %	29.51 %	27.05 %
RSS 5	✓	34.44 %	77.78 %	33.33 %	24.44 %
	✗	65.56 %	22.22 %	66.67 %	75.56 %
Total of RSS1-RSS5	✓	**59.55 %**	**80.65 %**	**54.52 %**	**44.22 %**
	✗	**40.05 %**	**19.35 %**	**45.48 %**	**55.78 %**

5.2.1 RSS grades calculation to RSS1, RSS2, RSS3, RSS4 and RSS5

The best testing results of images calculation is define k cluster equal to 2. The summation results of RSS grades calculation was shown in Table 5.5 which shows that most of images selected for testing were successfully processed. The value of result group by RSS grades was represented in Table 5.6.

- Images in category *"Successfully"* mean images which the system was able to calculate the results as same as the result from experts or scientists. This category has 80.65% successfully.

- Images in category *"Failure"* mean the result that system calculates images are different from the result of the experts or scientists.

Table 5.5 The value in testing phase group by degree of validity (RSS1 – RSS5)

Validity of images	Number of Images	Ratio (%)
Successfully	321	80.65
Failure	77	19.35
Total	**398**	**100**

The summary discussions about these problems are discussed in section 5.3 (Discussion), page 144.

Table 5.6 The value of result in testing phase group by RSS grades (RSS1 – RSS5)

<table>
<tr><th>RSS grades</th><th>Validity of images</th><th>Number of Images</th><th>Ratio (%)</th></tr>
<tr><td rowspan="3">RSS 1</td><td>Successfully</td><td>8</td><td>100</td></tr>
<tr><td>Failure</td><td>0</td><td>0</td></tr>
<tr><td>Total</td><td>8</td><td>100</td></tr>
<tr><td rowspan="3">RSS 2</td><td>Successfully</td><td>42</td><td>68.85</td></tr>
<tr><td>Failure</td><td>19</td><td>31.15</td></tr>
<tr><td>Total</td><td>61</td><td>100</td></tr>
<tr><td rowspan="3">RSS 3</td><td>Successfully</td><td>94</td><td>80.34</td></tr>
<tr><td>Failure</td><td>23</td><td>19.66</td></tr>
<tr><td>Total</td><td>117</td><td>100</td></tr>
<tr><td rowspan="3">RSS 4</td><td>Successfully</td><td>107</td><td>87.70</td></tr>
<tr><td>Failure</td><td>15</td><td>12.30</td></tr>
<tr><td>Total</td><td>122</td><td>100</td></tr>
<tr><td rowspan="3">RSS 5</td><td>Successfully</td><td>70</td><td>77.78</td></tr>
<tr><td>Failure</td><td>20</td><td>22.22</td></tr>
<tr><td>Total</td><td>90</td><td>100</td></tr>
<tr><td rowspan="3">Total of RSS1-RSS5</td><td>Successfully</td><td>321</td><td>80.65</td></tr>
<tr><td>Failure</td><td>77</td><td>19.35</td></tr>
<tr><td>Total</td><td>398</td><td>100</td></tr>
</table>

A total of 398 images made up a test dataset. Table 5.6 show summary of each RSS sources, Table 5.7 – 5.11 will describe result in detail of each RSS grades. The result of images will compare between human experts and AGRSS system then report of result that is correct or wrong calculation.

Table 5.7 The value of result testing phase images group by RSS1

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
1	TEST_RSS1_001	1	1	✓	10.19
2	TEST_RSS1_002	1	1	✓	9.68
3	TEST_RSS1_003	1	1	✓	10.07
4	TEST_RSS1_004	1	1	✓	8.84
5	TEST_RSS1_005	1	1	✓	9.73
6	TEST_RSS1_006	1	1	✓	9.31
7	TEST_RSS1_007	1	1	✓	9.79
8	TEST_RSS1_008	1	1	✓	9.61
Total = 8			✓ = 8	✗ = 0	77.22 / 8 = 9.65

Table 5.8 The value of result testing phase images group by RSS2

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
1	TEST_RSS2_001	2	2	✓	10.94
2	TEST_RSS2_002	2	5	✗	10.36
3	TEST_RSS2_003	2	3	✗	11.44
4	TEST_RSS2_004	2	3	✗	11.22
5	TEST_RSS2_005	2	3	✗	10.13
6	TEST_RSS2_006	2	3	✗	10.04
7	TEST_RSS2_007	2	2	✓	10.28
8	TEST_RSS2_008	2	3	✗	11.51

Table 5.8 The value of result testing phase images group by RSS2 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
9	TEST_RSS2_009	2	2	✓	9.31
10	TEST_RSS2_010	2	3	✗	11.89
11	TEST_RSS2_011	2	2	✓	11.79
12	TEST_RSS2_012	2	3	✗	12.07
13	TEST_RSS2_013	2	3	✗	9.21
14	TEST_RSS2_014	2	2	✓	10.42
15	TEST_RSS2_015	2	4	✗	12.14
16	TEST_RSS2_016	2	4	✗	10.60
17	TEST_RSS2_017	2	2	✓	9.95
18	TEST_RSS2_018	2	2	✓	9.68
19	TEST_RSS2_019	2	2	✓	10.25
20	TEST_RSS2_020	2	3	✗	9.62
21	TEST_RSS2_021	2	2	✓	10.20
22	TEST_RSS2_022	2	2	✓	8.87
23	TEST_RSS2_023	2	2	✓	11.04
24	TEST_RSS2_024	2	2	✓	9.20
25	TEST_RSS2_025	2	2	✓	10.53
26	TEST_RSS2_026	2	5	✗	11.38
27	TEST_RSS2_027	2	2	✓	11.19
28	TEST_RSS2_028	2	2	✓	11.22
29	TEST_RSS2_029	2	2	✓	8.73
30	TEST_RSS2_030	2	2	✓	9.71
31	TEST_RSS2_031	2	2	✓	8.92
32	TEST_RSS2_032	2	2	✓	10.77
33	TEST_RSS2_033	2	3	✗	11.29
34	TEST_RSS2_034	2	5	✗	10.56
35	TEST_RSS2_035	2	2	✓	14.66

Table 5.8 The value of result testing phase images group by RSS2 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
36	TEST_RSS2_036	2	2	✔	13.68
37	TEST_RSS2_037	2	2	✔	14.03
38	TEST_RSS2_038	2	2	✔	14.24
39	TEST_RSS2_039	2	2	✔	11.83
40	TEST_RSS2_040	2	2	✔	13.89
41	TEST_RSS2_041	2	2	✔	14.15
42	TEST_RSS2_042	2	2	✔	13.95
43	TEST_RSS2_043	2	2	✔	14.12
44	TEST_RSS2_044	2	2	✔	13.76
45	TEST_RSS2_045	2	2	✔	10.13
46	TEST_RSS2_046	2	4	✘	11.54
47	TEST_RSS2_047	2	2	✔	10.40
48	TEST_RSS2_048	2	3	✘	8.72
49	TEST_RSS2_049	2	2	✔	10.18
50	TEST_RSS2_050	2	2	✔	9.12
51	TEST_RSS2_051	2	2	✔	9.51
52	TEST_RSS2_052	2	1	✘	9.83
53	TEST_RSS2_053	2	2	✔	8.54
54	TEST_RSS2_054	2	2	✔	9.49
55	TEST_RSS2_055	2	3	✘	10.42
56	TEST_RSS2_056	2	2	✔	9.19
57	TEST_RSS2_057	2	2	✔	9.69
58	TEST_RSS2_058	2	2	✔	9.35
59	TEST_RSS2_059	2	2	✔	10.73
60	TEST_RSS2_060	2	3	✘	8.83
61	TEST_RSS2_061	2	2	✔	11.17
Total = 61			✔ = 42	✘ = 19	661.61 / 61 = 10.85

Table 5.9 The value of result testing phase images group by RSS3

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
1	TEST_RSS3_001	3	4	✗	11.13
2	TEST_RSS3_002	3	4	✗	10.94
3	TEST_RSS3_003	3	3	✓	11.61
4	TEST_RSS3_004	3	3	✓	12.31
5	TEST_RSS3_005	3	3	✓	11.42
6	TEST_RSS3_006	3	3	✓	11.73
7	TEST_RSS3_007	3	3	✓	13.52
8	TEST_RSS3_008	3	3	✓	10.92
9	TEST_RSS3_009	3	3	✓	11.53
10	TEST_RSS3_010	3	4	✗	11.96
11	TEST_RSS3_011	3	3	✓	12.67
12	TEST_RSS3_012	3	3	✓	13.72
13	TEST_RSS3_013	3	3	✓	12.43
14	TEST_RSS3_014	3	3	✓	12.33
15	TEST_RSS3_015	3	3	✓	11.58
16	TEST_RSS3_016	3	3	✓	12.19
17	TEST_RSS3_017	3	3	✓	10.55
18	TEST_RSS3_018	3	3	✓	11.27
19	TEST_RSS3_019	3	3	✓	13.13
20	TEST_RSS3_020	3	3	✓	11.39
21	TEST_RSS3_021	3	3	✓	11.71
22	TEST_RSS3_022	3	4	✗	10.97
23	TEST_RSS3_023	3	3	✓	11.97
24	TEST_RSS3_024	3	4	✗	11.97
25	TEST_RSS3_025	3	4	✗	10.20
26	TEST_RSS3_026	3	4	✗	11.32
27	TEST_RSS3_027	3	3	✓	10.71

Table 5.9 The value of result testing phase images group by RSS3 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
28	TEST_RSS3_028	3	4	✗	9.71
29	TEST_RSS3_029	3	4	✗	11.23
30	TEST_RSS3_030	3	4	✗	12.07
31	TEST_RSS3_031	3	3	✓	11.26
32	TEST_RSS3_032	3	3	✓	11.15
33	TEST_RSS3_033	3	3	✓	11.65
34	TEST_RSS3_034	3	3	✓	13.24
35	TEST_RSS3_035	3	3	✓	9.10
36	TEST_RSS3_036	3	3	✓	9.18
37	TEST_RSS3_037	3	3	✓	10.15
38	TEST_RSS3_038	3	3	✓	10.37
39	TEST_RSS3_039	3	3	✓	10.27
40	TEST_RSS3_040	3	3	✓	10.84
41	TEST_RSS3_041	3	3	✓	10.21
42	TEST_RSS3_042	3	2	✗	9.04
43	TEST_RSS3_043	3	3	✓	10.07
44	TEST_RSS3_044	3	3	✓	9.11
45	TEST_RSS3_045	3	2	✗	9.69
46	TEST_RSS3_046	3	2	✗	10.67
47	TEST_RSS3_047	3	3	✓	10.78
48	TEST_RSS3_048	3	3	✓	9.47
49	TEST_RSS3_049	3	2	✗	8.19
50	TEST_RSS3_050	3	3	✓	10.65
51	TEST_RSS3_051	3	3	✓	9.96
52	TEST_RSS3_052	3	3	✓	8.58
53	TEST_RSS3_053	3	3	✓	10.29
54	TEST_RSS3_054	3	3	✓	9.07

Table 5.9 The value of result testing phase images group by RSS3 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
55	TEST_RSS3_055	3	3	✓	10.33
56	TEST_RSS3_056	3	3	✓	9.36
57	TEST_RSS3_057	3	3	✓	9.76
58	TEST_RSS3_058	3	2	✗	9.76
59	TEST_RSS3_059	3	3	✓	9.20
60	TEST_RSS3_060	3	3	✓	10.27
61	TEST_RSS3_061	3	1	✗	8.80
62	TEST_RSS3_062	3	3	✓	9.00
63	TEST_RSS3_063	3	3	✓	9.11
64	TEST_RSS3_064	3	3	✓	8.61
65	TEST_RSS3_065	3	5	✗	9.49
66	TEST_RSS3_066	3	2	✗	9.97
67	TEST_RSS3_067	3	3	✓	9.76
68	TEST_RSS3_068	3	3	✓	9.10
69	TEST_RSS3_069	3	3	✓	8.95
70	TEST_RSS3_070	3	3	✓	8.63
71	TEST_RSS3_071	3	3	✓	8.83
72	TEST_RSS3_072	3	3	✓	8.69
73	TEST_RSS3_073	3	3	✓	10.08
74	TEST_RSS3_074	3	3	✓	9.49
75	TEST_RSS3_075	3	3	✓	10.16
76	TEST_RSS3_076	3	3	✓	8.19
77	TEST_RSS3_077	3	3	✓	9.73
78	TEST_RSS3_078	3	3	✓	9.69
79	TEST_RSS3_079	3	3	✓	7.68
80	TEST_RSS3_080	3	3	✓	8.55
81	TEST_RSS3_081	3	3	✓	8.57

Table 5.9 The value of result testing phase images group by RSS3 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
82	TEST_RSS3_082	3	3	✓	9.49
83	TEST_RSS3_083	3	3	✓	14.34
84	TEST_RSS3_084	3	3	✓	11.23
85	TEST_RSS3_085	3	3	✓	11.63
86	TEST_RSS3_086	3	2	✗	12.31
87	TEST_RSS3_087	3	4	✗	12.84
88	TEST_RSS3_088	3	2	✗	12.57
89	TEST_RSS3_089	3	3	✓	11.10
90	TEST_RSS3_090	3	3	✓	12.07
91	TEST_RSS3_091	3	3	✓	12.16
92	TEST_RSS3_092	3	3	✓	11.23
93	TEST_RSS3_093	3	3	✓	10.47
94	TEST_RSS3_094	3	3	✓	11.96
95	TEST_RSS3_095	3	3	✓	11.08
96	TEST_RSS3_096	3	3	✓	10.82
97	TEST_RSS3_097	3	3	✓	12.95
98	TEST_RSS3_098	3	3	✓	14.12
99	TEST_RSS3_099	3	3	✓	14.19
100	TEST_RSS3_100	3	3	✓	11.88
101	TEST_RSS3_101	3	3	✓	12.68
102	TEST_RSS3_102	3	3	✓	12.02
103	TEST_RSS3_103	3	3	✓	12.61
104	TEST_RSS3_104	3	3	✓	5.79
105	TEST_RSS3_105	3	3	✓	13.59
106	TEST_RSS3_106	3	3	✓	12.63
107	TEST_RSS3_107	3	3	✓	12.26
108	TEST_RSS3_108	3	3	✓	12.51

Table 5.9 The value of result testing phase images group by RSS3 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
109	TEST_RSS3_109	3	5	✗	13.20
110	TEST_RSS3_110	3	3	✓	11.94
111	TEST_RSS3_111	3	3	✓	13.84
112	TEST_RSS3_112	3	3	✓	11.13
113	TEST_RSS3_113	3	3	✓	11.66
114	TEST_RSS3_114	3	3	✓	12.92
115	TEST_RSS3_115	3	3	✓	12.56
116	TEST_RSS3_116	3	2	✗	11.89
117	TEST_RSS3_117	3	3	✓	12.99
Total = 117			✓ = 94	✗ = 23	1,277.60 / 117 = 10.92

Table 5.10 The value of result testing phase images group by RSS4

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
1	TEST_RSS4_001	4	4	✓	13.85
2	TEST_RSS4_002	4	4	✓	14.25
3	TEST_RSS4_003	4	4	✓	14.28
4	TEST_RSS4_004	4	4	✓	14.19
5	TEST_RSS4_005	4	4	✓	13.18
6	TEST_RSS4_006	4	4	✓	13.37
7	TEST_RSS4_007	4	4	✓	14.10
8	TEST_RSS4_008	4	4	✓	10.09
9	TEST_RSS4_009	4	4	✓	11.05
10	TEST_RSS4_010	4	4	✓	8.93
11	TEST_RSS4_011	4	4	✓	9.96
12	TEST_RSS4_012	4	4	✓	9.22

Table 5.10 The value of result testing phase images group by RSS4 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
13	TEST_RSS4_013	4	4	✓	8.07
14	TEST_RSS4_014	4	4	✓	9.87
15	TEST_RSS4_015	4	4	✓	11.78
16	TEST_RSS4_016	4	4	✓	13.91
17	TEST_RSS4_017	4	4	✓	11.04
18	TEST_RSS4_018	4	4	✓	11.47
19	TEST_RSS4_019	4	4	✓	8.72
20	TEST_RSS4_020	4	4	✓	10.28
21	TEST_RSS4_021	4	4	✓	9.35
22	TEST_RSS4_022	4	4	✓	9.38
23	TEST_RSS4_023	4	4	✓	9.98
24	TEST_RSS4_024	4	4	✓	8.72
25	TEST_RSS4_025	4	4	✓	9.65
26	TEST_RSS4_026	4	4	✓	9.01
27	TEST_RSS4_027	4	4	✓	10.76
28	TEST_RSS4_028	4	4	✓	10.11
29	TEST_RSS4_029	4	4	✓	9.53
30	TEST_RSS4_030	4	4	✓	10.04
31	TEST_RSS4_031	4	3	✗	10.65
32	TEST_RSS4_032	4	4	✓	9.65
33	TEST_RSS4_033	4	4	✓	10.05
34	TEST_RSS4_034	4	4	✓	9.57
35	TEST_RSS4_035	4	4	✓	9.73
36	TEST_RSS4_036	4	4	✓	9.20
37	TEST_RSS4_037	4	4	✓	6.22
38	TEST_RSS4_038	4	4	✓	4.45
39	TEST_RSS4_039	4	4	✓	8.99

Table 5.10 The value of result testing phase images group by RSS4 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
40	TEST_RSS4_040	4	4	✓	7.82
41	TEST_RSS4_041	4	4	✓	10.73
42	TEST_RSS4_042	4	4	✓	9.08
43	TEST_RSS4_043	4	4	✓	9.21
44	TEST_RSS4_044	4	4	✓	8.70
45	TEST_RSS4_045	4	4	✓	9.64
46	TEST_RSS4_046	4	4	✓	8.41
47	TEST_RSS4_047	4	4	✓	9.14
48	TEST_RSS4_048	4	5	✗	9.94
49	TEST_RSS4_049	4	4	✓	9.62
50	TEST_RSS4_050	4	4	✓	10.40
51	TEST_RSS4_051	4	4	✓	7.75
52	TEST_RSS4_052	4	4	✓	10.37
53	TEST_RSS4_053	4	4	✓	7.44
54	TEST_RSS4_054	4	4	✓	11.13
55	TEST_RSS4_055	4	4	✓	10.89
56	TEST_RSS4_056	4	4	✓	11.69
57	TEST_RSS4_057	4	4	✓	12.49
58	TEST_RSS4_058	4	3	✗	11.00
59	TEST_RSS4_059	4	4	✓	11.04
60	TEST_RSS4_060	4	4	✓	11.66
61	TEST_RSS4_061	4	4	✓	10.95
62	TEST_RSS4_062	4	4	✓	11.69
63	TEST_RSS4_063	4	3	✗	11.88
64	TEST_RSS4_064	4	4	✓	12.19
65	TEST_RSS4_065	4	3	✗	11.83
66	TEST_RSS4_066	4	4	✓	12.81

Table 5.10 The value of result testing phase images group by RSS4 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
67	TEST_RSS4_067	4	4	✓	11.94
68	TEST_RSS4_068	4	4	✓	12.26
69	TEST_RSS4_069	4	5	✗	11.45
70	TEST_RSS4_070	4	4	✓	10.29
71	TEST_RSS4_071	4	4	✓	10.73
72	TEST_RSS4_072	4	5	✗	11.43
73	TEST_RSS4_073	4	5	✗	12.83
74	TEST_RSS4_074	4	4	✓	11.34
75	TEST_RSS4_075	4	4	✓	12.25
76	TEST_RSS4_076	4	4	✓	11.18
77	TEST_RSS4_077	4	4	✓	9.20
78	TEST_RSS4_078	4	4	✓	11.43
79	TEST_RSS4_079	4	4	✓	13.85
80	TEST_RSS4_080	4	4	✓	12.06
81	TEST_RSS4_081	4	4	✓	12.41
82	TEST_RSS4_082	4	4	✓	11.31
83	TEST_RSS4_083	4	4	✓	10.83
84	TEST_RSS4_084	4	4	✓	11.93
85	TEST_RSS4_085	4	4	✓	9.37
86	TEST_RSS4_086	4	4	✓	8.97
87	TEST_RSS4_087	4	4	✓	12.48
88	TEST_RSS4_088	4	4	✓	10.04
89	TEST_RSS4_089	4	4	✓	9.29
90	TEST_RSS4_090	4	3	✗	10.38
91	TEST_RSS4_091	4	4	✓	11.83
92	TEST_RSS4_092	4	4	✓	10.93
93	TEST_RSS4_093	4	4	✓	10.46

Table 5.10 The value of result testing phase images group by RSS4 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
94	TEST_RSS4_094	4	4	✓	10.82
95	TEST_RSS4_095	4	3	✗	11.80
96	TEST_RSS4_096	4	4	✓	11.89
97	TEST_RSS4_097	4	4	✓	12.43
98	TEST_RSS4_098	4	4	✓	8.80
99	TEST_RSS4_099	4	4	✓	12.25
100	TEST_RSS4_100	4	4	✓	11.85
101	TEST_RSS4_101	4	4	✓	11.64
102	TEST_RSS4_102	4	4	✓	10.50
103	TEST_RSS4_103	4	4	✓	9.74
104	TEST_RSS4_104	4	4	✓	10.14
105	TEST_RSS4_105	4	4	✓	10.09
106	TEST_RSS4_106	4	4	✓	10.54
107	TEST_RSS4_107	4	4	✓	10.47
108	TEST_RSS4_108	4	4	✓	9.59
109	TEST_RSS4_109	4	4	✓	9.63
110	TEST_RSS4_110	4	3	✗	9.68
111	TEST_RSS4_111	4	3	✗	10.01
112	TEST_RSS4_112	4	3	✗	10.32
113	TEST_RSS4_113	4	4	✓	9.80
114	TEST_RSS4_114	4	4	✓	9.93
115	TEST_RSS4_115	4	4	✓	10.66
116	TEST_RSS4_116	4	4	✓	11.31
117	TEST_RSS4_117	4	4	✓	9.52
118	TEST_RSS4_118	4	3	✗	12.38
119	TEST_RSS4_119	4	3	✗	9.07
120	TEST_RSS4_120	4	4	✓	10.47

Table 5.10 The value of result testing phase images group by RSS4 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
121	TEST_RSS4_121	4	4	✓	12.57
122	TEST_RSS4_122	4	4	✓	11.79
Total = 122			✓ = 107	✗ = 15	1,298.31 / 122 = 10.64

Table 5.11 The value of result testing phase images group by RSS5

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
1	TEST_RSS5_001	5	5	✓	9.11
2	TEST_RSS5_002	5	5	✓	8.95
3	TEST_RSS5_003	5	5	✓	8.89
4	TEST_RSS5_004	5	5	✓	9.39
5	TEST_RSS5_005	5	5	✓	10.94
6	TEST_RSS5_006	5	3	✗	10.19
7	TEST_RSS5_007	5	5	✓	10.98
8	TEST_RSS5_008	5	5	✓	9.41
9	TEST_RSS5_009	5	5	✓	9.82
10	TEST_RSS5_010	5	5	✓	10.07
11	TEST_RSS5_011	5	3	✗	10.80
12	TEST_RSS5_012	5	2	✗	9.47
13	TEST_RSS5_013	5	5	✓	10.08
14	TEST_RSS5_014	5	5	✓	10.60
15	TEST_RSS5_015	5	5	✓	11.15
16	TEST_RSS5_016	5	5	✓	13.07
17	TEST_RSS5_017	5	5	✓	13.31
18	TEST_RSS5_018	5	5	✓	8.63
19	TEST_RSS5_019	5	5	✓	11.90

Table 5.11 The value of result testing phase images group by RSS5 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
20	TEST_RSS5_020	5	5	✓	12.24
21	TEST_RSS5_021	5	5	✓	10.02
22	TEST_RSS5_022	5	5	✓	10.87
23	TEST_RSS5_023	5	3	✗	12.26
24	TEST_RSS5_024	5	5	✓	10.37
25	TEST_RSS5_025	5	5	✓	9.75
26	TEST_RSS5_026	5	5	✓	11.31
27	TEST_RSS5_027	5	5	✓	11.36
28	TEST_RSS5_028	5	5	✓	11.83
29	TEST_RSS5_029	5	5	✓	11.59
30	TEST_RSS5_030	5	5	✓	9.95
31	TEST_RSS5_031	5	5	✓	10.10
32	TEST_RSS5_032	5	5	✓	10.54
33	TEST_RSS5_033	5	5	✓	10.97
34	TEST_RSS5_034	5	5	✓	9.99
35	TEST_RSS5_035	5	5	✓	10.90
36	TEST_RSS5_036	5	5	✓	10.07
37	TEST_RSS5_037	5	5	✓	11.94
38	TEST_RSS5_038	5	5	✓	12.15
39	TEST_RSS5_039	5	4	✗	11.40
40	TEST_RSS5_040	5	5	✓	10.45
41	TEST_RSS5_041	5	5	✓	8.21
42	TEST_RSS5_042	5	3	✗	9.32
43	TEST_RSS5_043	5	3	✗	8.87
44	TEST_RSS5_044	5	5	✓	11.44
45	TEST_RSS5_045	5	3	✗	10.78
46	TEST_RSS5_046	5	3	✗	12.02

Table 5.11 The value of result testing phase images group by RSS5 (Continued)

No.	RSS SOURCE	Expert / Scientist	AGRSS System	Result	Processing Time (second / image)
47	TEST_RSS5_047	5	3	✘	11.95
48	TEST_RSS5_048	5	5	✔	11.54
49	TEST_RSS5_049	5	5	✔	10.57
50	TEST_RSS5_050	5	5	✔	9.95
51	TEST_RSS5_051	5	3	✘	11.18
52	TEST_RSS5_052	5	5	✔	10.56
53	TEST_RSS5_053	5	3	✘	9.74
54	TEST_RSS5_054	5	5	✔	9.61
55	TEST_RSS5_055	5	5	✔	13.03
56	TEST_RSS5_056	5	5	✔	13.33
57	TEST_RSS5_057	5	5	✔	13.56
58	TEST_RSS5_058	5	5	✔	11.77
59	TEST_RSS5_059	5	5	✔	12.12
60	TEST_RSS5_060	5	4	✘	11.45
61	TEST_RSS5_061	5	5	✔	12.51
62	TEST_RSS5_062	5	5	✔	12.77
63	TEST_RSS5_063	5	5	✔	10.28
64	TEST_RSS5_064	5	5	✔	9.63
65	TEST_RSS5_065	5	5	✔	11.57
66	TEST_RSS5_066	5	5	✔	11.18
67	TEST_RSS5_067	5	5	✔	10.84
68	TEST_RSS5_068	5	4	✘	12.27
69	TEST_RSS5_069	5	5	✔	11.35
70	TEST_RSS5_070	5	5	✔	12.04
71	TEST_RSS5_071	5	5	✔	10.98
72	TEST_RSS5_072	5	5	✔	12.27
73	TEST_RSS5_073	5	5	✔	12.25

Table 5.11 The value of result testing phase images group by RSS5 (Continued)

No.	**RSS SOURCE**	**Expert / Scientist**	**AGRSS System**	**Result**	**Processing Time (second / image)**
74	TEST_RSS5_074	5	3	✗	9.45
75	TEST_RSS5_075	5	4	✗	10.15
76	TEST_RSS5_076	5	5	✓	12.30
77	TEST_RSS5_077	5	5	✓	11.14
78	TEST_RSS5_078	5	5	✓	14.59
79	TEST_RSS5_079	5	5	✓	12.27
80	TEST_RSS5_080	5	5	✓	11.65
81	TEST_RSS5_081	5	5	✓	12.03
82	TEST_RSS5_082	5	5	✓	13.34
83	TEST_RSS5_083	5	3	✗	9.92
84	TEST_RSS5_084	5	5	✓	11.63
85	TEST_RSS5_085	5	5	✓	11.05
86	TEST_RSS5_086	5	5	✓	12.88
87	TEST_RSS5_087	5	3	✗	11.13
88	TEST_RSS5_088	5	3	✗	10.44
89	TEST_RSS5_089	5	3	✗	11.18
90	TEST_RSS5_090	5	5	✓	11.10
Total = 90			✓ = 70	✗ = 20	994.01 / 90 = 11.04

5.3 Discussion

This section discusses about difficulties that found throughout this work, the system performance such as accuracy rate and response time from the graphic user interface (GUI) part and suggest some possible solutions to solve the problem that found in the work.

5.3.1 Difficulties

Occasional difficulties were found that many private manufacturers are not allowing stranger to get in the factory to protect their official secret. Fortunately, this problem can be solved by contact government organization. They welcome student to conduct research at their workplace.

Collecting images of Ribbed Smoked Sheet (RSS) have a limited time because of those RSS have to sell day by day. You can collect RSS images data after the experts or scientists grading RSS already and before those RSS were sent to bidder it has times about 5 hours per day (about 3 hours in the morning and 2 hours in the afternoon).

Experts or scientists are very busy and they can't grade every RSS sheets. They random grading some sheets from group of sheets so the error may be occurs. The acceptable error is in one group such as 10 sheets can not exceed 2 sheets that not in the same grade. Because, if expert found mixes grades more than 2 sheets the agriculturist will lost price to a low grade all of that RSS group, so agriculturist would grading at least once time before sold.

In case of immoral middlemen or agriculturists intend to cheat and the experts found. The punishment for them in the first time is listing their name into the parole's list if it happened again they will lose member's right and can not come to sell at the central rubber market.

RSS1 and RSS2 is a very good rubber grades and very hard to find out. The central rubber market set different 3 prices for 5 RSS grades. They group RSS1, RSS2 and RSS3 together and set same prices (people are always called groups of 3 RSS grades from RSS1-RSS3 be RSS3).

In this work, the experts help the author find out and grading RSS1 and RSS2 from RSS3 group. But in the limited times and limited rubber sheets they found very few of RSS1 and very little of RSS2. So, RSS2 in this work may be look like RSS3. In this experimental results report that RSS2 turn into RSS3 about 20 percent (12 sheets from 61 sheets).

For the segmentation RSS images, the author try to crop coordinate RSS area with true color image. It found that the system will work well only with the clear RSS sheets such as RSS1, RSS2 and some RSS3. The lower rubber grade like RSS4 and RSS5 occur some problem with color segmentation. Some of RSS3 - RSS5 have dark color sheet and when the system crop coordinates of RSS image, some rubber sheets were cut in the RSS area. It will make RSS image lost data and can not estimate the correct area from the remainder image. This problem can be solve by use binary image which convert from grayscale image and remove noise already to crop coordinates out and get only interested RSS area.

K-means clustering does not yield the same result with each run, since the resulting clusters depending on the initial random assignments. It minimizes intra cluster variance, but does not ensure that the result has a global minimum of variance. But this problem can be solved by rearrange cluster after clustering already. The system will compare cluster area and rearrange cluster area automatically from maximum cluster area to minimum cluster area. Sometimes clustering give a result to be an equal cluster so the system will remark two of this clusters to be maximum cluster area and the remain cluster is minimum cluster area.

5.3.2 System performance

In this work, system performance is defined in terms of accuracy rate, response time, and program complexity.

5.3.2.1 Accuracy rate

On consideration of the system accurate, the system performance was reported in Table 5.5 – 5.11. It was the result of RSS images from the graphic user interface (GUI) part which could calculation of RSS grades. This section is the images result processed in test dataset.

The accuracy rate is measured by the percentage of images in which RSS grades are successfully calculation. Table 5.5 show the overall calculation of rate from grading experimental. An important point should be in order to explain the table fairly: the ratio (%) of validity type in “Failure” is the error rate, while “Successfully” means ‘no error at all’, therefore its ratio (%) means accuracy rate. The experimental classify RSS images into RSS1, RSS2, RSS3, RSS4 and RSS5 grades. The “Successfully” rate at 80.65% is acceptable.

Most failed images in these categories resulted from similarity color of RSS images but were in different RSS grades which possibly decrease this inaccurate by applying more clustering technique. A potential development of algorithm should solve this type of error.

5.3.2.2 Response time

The user will work with the GUI part efficiently as response time is acceptable. The only exception is the K-means clustering step to compute every pixel location in the original image. Dimensions of the RSS image sources represent in 640 x 480 pixels. On the personal computer brand ACER, model VERITON M461 with Intel Core 2 Duo, CPU 2.6 GHz, and 1 GB of RAM.

The average interval time spent by the K-means process to clustering images is about 10.83 seconds. This measurement starts after the user selected RSS image to analyse grade and stop after K-means clustering every pixels in selected image and calculation of the RSS grades result already.

The response times also depending on other configurations such as other hardware components, operating system, and other software components running on the same machine. However, it has a little problem because of nowadays new technologies develop so fast. A faster machine will finish the clustering and calculation process more quickly. Otherwise a better clustering and calculation of algorithm is recommended.

5.3.2.3 Program complexity

The graphic user interface (GUI) part was developed in the language of technical computing using MATLAB version 7.0.4, MATLAB is a high-level language and interactive environment that enables user to perform computationally intensive tasks faster than with traditional programming languages such as C, C++, and Fortran.

MATLAB can use in a wide range of applications, including signal and image processing, communications, control design, test and measurement, financial modeling and analysis, and computational biology. Add-on toolboxes (collections of special-purpose MATLAB functions, available separately) extend the MATLAB environment to solve particular classes of problems in these application areas.

The MATLAB language supports the vector and matrix operations that are fundamental to engineering and scientific problems. It enables fast development and execution. With the MATLAB language, you can program and develop algorithms faster than with traditional languages because you do not need to perform low-level administrative tasks, such as declaring variables, specifying data types, and allocating memory.

MATLAB lets you execute commands or groups of commands one at a time, without compiling and linking, enabling you to quickly iterate to the optimal solution.

5.3.3 System flaws

This section describes the system faults which were found during running program in training and testing phases, together with the reason of this insufficiency. The possible solutions of each problem are also proposed for consideration of further improvement work.

5.3.3.1 Hole in RSS

Calculation of color from each clusters in the RSS images. AGRSS system uses colors information from RSS images as the main properties. If a hole occur in the RSS image, the system will calculate this hole to be 1 cluster of image and it will make the system lose one cluster of RSS information to analyse grade and may be make insufficiently result occur.

5.3.3.2 Contaminated objects in RSS

In some cases, the RSS input image has contaminated objects on the surface but in the real situation it is not involve in RSS. This situation may be happen from unpredictable cases or mistake from human. The system will detect the object as a part of RSS image and then following the ordinary system but in the worst case the system will focus on the object instead of point to RSS then both cases will make fault results occur.

5.3.4 Other issues

This section explains about other topics that happen in the system but it is not in the system flaws.

5.3.4.1 System design

In fact, the proposed system may not much use as it requires human interaction, because it was difficult to find the right place or instruments such as dark box, control light, personal computer, or digital camera to collecting RSS images for calculation of RSS grades. It also only works well with a perfect environment and instruments. So, it may be hard to assign these equipments in the real situation.

A realistic system should work well in every environment such as outdoor or even at sunset or sunrise, do not require dark box or control light and can analyse RSS grades with image taken from any devices which able to take an image such as mobile phones. Another possible solution is to use indoor where have enough light to take an image and refine some algorithm to evaluate light and automatic contrast image for calculation of RSS grades

5.3.4.2 Refining algorithm

The K-means clustering used in this work reveals acceptable results. Sometimes, some RSS images look like more than one grade. This situation can solve when the system calculation of image compare to the most result in training phase. But in some RSS images it makes a wrong result. So, a more suitable algorithm is expected to solve this problem. In this work, most supported RSS images acquire into the system can be analyse grade at high accuracy rate by K-means clustering. It works well but it can not apply to other possible problem. It is considerable work to design another flexible algorithm which is effective enough to analyse RSS grades in every environment.

CHAPTER VI
CONCLUSION AND SUGGESTIONS FOR FUTURE WORK

In Thailand, only few research studies have been conducted in this field. Furthermore, the former works are not concerned about the analyse grade of Ribbed Smoked Sheet (RSS) using an image. This work proposes an approach to estimate grade with surface of RSS images. The objectives are to analysis and estimate grade of RSS using an image processing technique. The outputs were proved successful by compare with human expert inspection. The previous chapter reports the experimental results and also discusses the reason why it happened. According to the context, several points of conclusion are made in the following sections. Further studies for improving this work are also suggested at the end of chapter

6.1 Conclusion

Analysis Grade of Ribbed Smoked Sheet (AGRSS) system consists of 5 main processes, which are 1) Image Acquisition, 2) Segmentation, 3) Clustering, 4) Calculation of RSS grades, and 5) Display of results. Image acquisition process uses a digital camera to take RSS images in a controlled environment box. All collected RSS images are selectively separated into two groups; 398 images for training and the other 398 images for testing. In the segmentation process use binary image which convert from grayscale image and remove noise already to crop coordinates out and get only interested RSS area for a clustering process. Clustering process apply L*a*b* color space (The L*a*b* color space consists of a luminosity layer 'L*', a chromaticity-layer 'a*' that indicates where the color falls along the red-green axis, and a chromaticity-layer 'b*' that indicates where the color falls along the blue-yellow axis.), Euclidean distance and K-means clustering to automatically find the best groupings for the RSS image and classify color range levels or calculate RSS grades.

The calculation of RSS grades process will classify RSS into five grades, which are RSS1 – RSS5. Display of results process has created GUI (Graphic User Interface) to serve the objectives of this work. This research tested the system by using 398 RSS images for a training dataset and another 398 RSS images for a test dataset. The precision rate of this work was 80.65 percent for the test dataset. The results indicate that the computer method is almost as accurate when compared with a human expert who inspects the RSS grades. The average access time for the AGRSS was around 10.83 seconds per RSS image.

6.1.1 Effectiveness of the proposed approach

The related theories from this work may be helpful for agriculturists to raise productivity and improve quality of Ribbed Smoked Sheets (RSS). The author hopes that this knowledge will encourage small rubber plantation owner to produce high quality of RSS and inculcate their awareness about the grading system.

Analysis Grade of Ribbed Smoked Sheet (AGRSS) system is supposed to enforce proper grading practices. But it may be not provided, nor insist on dealers that the Green Book be displayed for the small grower's reference. In this humanity justice it will depend each middlemen’s moral. The display of Green Book must be made mandatory at all procurement centers and grading must be made on the basis of what is prescribed therein if only to prevent procurement of Insufficiently Dried Sheets (IDS) and Insufficiently Smoked Sheets (ISS). This would ensure reasonable price for agriculturist.

The input images are taken in various colors in same grade. So many grades such as RSS3, RSS4 or RSS5 have more than one color. The rubber market or dealers should prepare RSS samples for display various grades of RSS at their collection outlets. Then try to show clean on visual grading and guarantee that the graded rubber collected bypassing from middlemen is sold directly to the manufacturer. The State Cooperative Rubber Marketing Federation and other cooperatives must buy only quality sheets such as up to RSS grade 5.

6.2 Suggestion for future work

According to the simplicity of the proposed system and its potential of upgradeability; improving the accuracy rate, encourage performance or adding more features is easy. To enhance efficiency of the method in real world application some improved examples are suggested here.

6.2.1 Automatic RSS grading

Adding this feature will saves a lot of human participation. After taking the whole top view of RSS image, the future system ought to automatic separate RSS from background, clustering, calculate RSS area, classify RSS grades and reported details to the user can be developed.

6.2.2 Apply more algorithm to detect contaminated objects, dirtiness or air bubbles

This capability is an important feature as these situations are difficult to avoid. An appropriate algorithm is needed more advanced clustering to detect and specify contaminated objects, dirtiness or air bubbles. Then display of results from these contaminated objects in GUI (graphic user interface) part.

6.2.3 Automatic configuration RSS grades range

To manage different color schemes in various dealers, different grades and variety of RSS colors. The system may provide segment of the color information in RSS images and apply more sufficient algorithm to separate clearly RSS grades. This additional process should help increasing accuracy rate of calculation of RSS grades.

6.2.4 Integrated with some specific devices

For some area of application a specific solution is needed. For example, new agriculturists want to sell Ribbed Smoked Sheet (RSS) but they don’t know grade. They can download program to the mobile phone and taken images with mobile camera, after the program processing, it can estimate grade of RSS and report of RSS grade results to agriculturists before selling to middlemen or retailer.

6.2.5 Clustering algorithm

The clustering system used in this work proved its efficiency in both training and testing phase. However, its performance can be improved by increase more techniques in color segmentation and clustering to classify RSS grades or detect contaminated objects out of the RSS images and then reported detail to the user in GUI (graphic user interface) part.

6.2.6 More example images build more accurate

Find the appropriate rubber manufacturers that can support your research. The experts or scientists who have enough times to provide you classify every RSS grades. Collect a lot of RSS images as you can for the most system performance and better accuracy rate.

REFERENCES

1. Nujanart Kungpisdan, “Rubber Technical document: Rubber’s Transform Industry”, Department of Agriculture, Ministry of Agriculture and Cooperatives, 2004.
2. Dr. Utpal Kumar Niyogi, “Introduction to Fibre Science and Rubber Technology”, Division of material Science, Shri Ram Institute for Industrial Research, 2007.
3. “Trang Hand Book”, WIDE NATURE, pp. 15 - 47, 2002.
4. Charoenmun Chaliewkreangkrai, Wantanee Jongkham, “Natural Rubber THE PREMIUM GRADE”, Office of the Board of the Innovation Development Fund.
5. Scott E Umbaugh. “Computer Imaging : Digital Image Analysis and Processing”, CRC PRESS BOOK, 2005.
6. Milan Sonka, Vaclav Hlavac, and Roger Boyle, “Image Processing, Analysis, and Machine Vision : Second Edition”, International Thomson Publishing Inc, 1999.
7. Adrian Low, “Introductory Computer Vision and Image Processing : International Edition”, McGraw-Hill Book Company, 1991.
8. Acharya and Ray, “Image Processing: Principles and Applications”, Wiley-Interscience, ISBN 0-471-71998-6, 2005.
9. Russ, “The Image Processing Handbook: Fourth Edition”, ISBN 0-8493-2532-3, 2002.
10. J. S. Weszka, R. N. Nagel, and A. Rosenfeld, “A threshold selection technique”, IEEE Trans. Comput., vol. C-23, pp. 1322 - 1326, 1974.
11. N. Otsu, “A threshold selection method from gray-level histograms”, IEEE Trans. on SMC, Vol. 9, pp. 62 - 66, 1979.
12. Gonzalez, Rafael C. & Woods, Richard E, “Digital Image Processing : SecondEdition”, Pearson Education, 2002.

13. Mehmet Sezgin and Bulent Sankur, "Survey over image thresholding techniques and quantitative performance evaluation", Journal of Electronic Imaging 13(1) doi:10.1117/1.1631315, pp. 146 – 165, January, 2004.

14. Ping-Sung Liao, Tse-Sheng Chen and Pau-Choo Chung, "A Fast Algorithm for Multilevel Thresholding", Journal of information science and engineering 17, pp. 713 - 727, 2001.

15. Kuk-Jin Yoon and In-So Kweon, "Color image segmentation considering of human sensitivity for color pattern variations", Robotics and Computer Vision Laboratory, Dept. of EECS, KAIST, Korea, pp. 269 – 278, 2001.

16. Naratorn and Thanate, "Classifying a young Aromatic coconuts flesh by image processing from The external".

17. J. A. Hartigan, "Clustering Algorithms", Wiley, 1975.

18. J. A. Hartigan and M. A. Wong, "A K-Means Clustering Algorithm", Applied Statistics, Vol. 28, No. 1, pp. 100 - 108, 1979.

19. D. Arthur, S. Vassilvitskii, "How Slow is the k-means Method?", Proceedings of the 2006 Symposium on Computational Geometry (SoCG), 2006.

20. J. B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations", Proceedings of Fifth Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, pp.281-297, 1967.

21. T. Kanungo, D. M. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation", IEEE Trans. Pattern Analysis and Machine Intelligence, pp. 881 - 892, 2002.

22. H. Zha, C. Ding, M. Gu, X. He and H.D. Simon, "Spectral Relaxation for K-means Clustering", Neural Information Processing Systems vol.14 (NIPS 2001), pp. 1057 - 1064, Vancouver, Canada, Dec, 2001.

23. Chris Ding and Xiaofeng He, "K-means Clustering via Principal Component Analysis", Proc. of Int'l Conf. Machine Learning (ICML 2004), pp. 225 - 232, July, 2004.

24. D. Arthur, S. Vassilvitskii, "k-means++ The Advantages of Careful Seeding", 2007 Symposium on Discrete Algorithms (SODA), 2007.

25. Dr.K.Sathirakul, Dr.W.Chinsirikul, Dr.S. Madaratsamee, Dr.P.Somnuk, Dr.D. Laowattana, B. Srisuwan and S. Leeraphan, “Automated Machine for Removing Impurities from Ribbed Smoked Rubber Sheets”, A Cradle of Future Leaders in Robotics.
26. Santi and Thanate, “Inspection of Printed for Rubber Sheet by Image Proceesing”.
27. Krisada and Koarakot, “Programming for Automatically Seprating Oil Glands from Pummelo Fruit Surface”, Department of Electrical Engineering, Faculty of Engineering, Kasetsart University.
28. Koarakot and Noppadol, “A Study on Relationship of Size and Density of Pomelo Oil Pigment Cells with Age”, Department of Electrical Engineering, Faculty of Engineering, Kasetsart University.
29. Savakis, A.E., “Adaptive Document Image Thresholding Using Foreground and Background Clustering”, ICIP, 1998.
30. The Mathwork, Inc., “MATLAB help”, 2005.

APPENDICES

APPENDIX A

DETAILS OF K-MEANS CLUSTERING FUNCTION

This appendix represents the source code details of K-means clustering function.

A1. Example of how to run the K-means clustering function

The following source code gives an example of how to run the K-means clustering.

```
% --- Executes on button press in btnKmean.
function Kmean
% hObject    handle to btnKmean (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

tic
load color_image;
load nX;
load nY;
format bank;

color_image_after = color_image;
I = color_image_after;
axes(handles.axes1)
imshow(color_image_after);
title('Ribbed Smokded Sheet (RSS) Image');
cform = makecform('srgb2lab');
lab_I = applycform(I,cform);
ab = double(lab_I(:,:,2:3));
nrows = size(ab,1);
ncols = size(ab,2);
ab = reshape(ab,nrows*ncols,2);

nColors = 2;
[cluster_idx cluster_center] = kmeans(ab,nColors,'distance','sqEuclidean', 'Replicates',3);

pixel_labels = reshape(cluster_idx,nrows,ncols);
segmented_images = cell(1,3);
rgb_label = repmat(pixel_labels,[1 1 3]);
```

```
 for k = 1:nColors
   color = I;
   color(rgb_label ~= k) = 0;
   segmented_images{k} = color;
end
cluster_im1 = segmented_images{1};
cluster_im2 = segmented_images{2};
imwrite(cluster_im1,'cluster_im1.jpg');
imwrite(cluster_im2,'cluster_im2.jpg');
info_clus1 = imfinfo('cluster_im1.jpg');
info_clus2 = imfinfo('cluster_im2.jpg');
size_clus1 = info_clus1.FileSize;
size_clus2 = info_clus2.FileSize;
atX = 1;, atY = 1;
pixR = 0;, pixG = 0;, pixB = 0;
minR1R = 0;, minG1G = 0;, minB1B = 0;
maxR1R = 0;, maxG1G = 0;, maxB1B = 0;
minR1G = 0;, minR1B = 0;, maxR1G = 0;, maxR1B = 0;
minG1R = 0;, minG1B = 0;, maxG1R = 0;, maxG1B = 0;
minB1R = 0;, minB1G = 0;, maxB1R = 0;, maxB1G = 0;
count1 = 0;, count1R = 0;, count1Area = 0;
meanR1R = 0;, meanR1G = 0;, meanR1B = 0;
count1Rmax =0;
  for atX = 1:1:nX
    atY = 1;
    for atY = 1:1:nY
        pixR = cluster_im1(atX,atY,1);
        pixG = cluster_im1(atX,atY,2);
        pixB = cluster_im1(atX,atY,3);
        if ((pixR ~= 0) &  (pixG ~= 0) & (pixB ~= 0))
          count1R = count1R + 1;
          meanR1R = uint32(meanR1R) + uint32(pixR);
          meanR1G = uint32(meanR1G) + uint32(pixG);
          meanR1B = uint32(meanR1B) + uint32(pixB);
          if count1R > count1Rmax
            count1Rmax = count1R;
          end
          minR1R = pixR;
          minR1G = pixG;
          minR1B = pixB;
          if pixR < minR1R
            minR1R = pixR;
            minR1G = pixG;
            minR1B = pixB;
          elseif pixR > maxR1R
            maxR1R = pixR;
            maxR1G = pixG;
            maxR1B = pixB;
          end
        end
    end
  end
meanR1R, meanR1G, meanR1B
meanR1R = meanR1R/count1Rmax;
meanR1G = meanR1G/count1Rmax;
```

```
meanR1B = meanR1B/count1Rmax;
minR1R;, minR1G;, minR1B;
maxR1R;, maxR1G;, maxR1B;
meanR1R;, meanR1G;, meanR1B;
count1R;, count1Rmax;
atX = 1;, atY = 1;
pixR = 0;, pixG = 0;, pixB = 0;
minR2R = 0;, minG2G = 0;, minB2B = 0;
maxR2R = 0;, maxG2G = 0;, maxB2B = 0;
minR2G = 0;, minR2B = 0;, maxR2G = 0;, maxR2B = 0;
minG2R = 0;, minG2B = 0;, maxG2R = 0;, maxG2B = 0;
minB2R = 0;, minB2G = 0;, maxB2R = 0;, maxB2G = 0;
count2 = 0;, count2R = 0;, count2Area = 0;
meanR2R = 0;, meanR2G = 0;, meanR2B = 0;
count2Rmax =0;
  for atX = 1:1:nX
    atY = 1;
    for atY = 1:1:nY
       pixR = cluster_im2(atX,atY,1);
       pixG = cluster_im2(atX,atY,2);
       pixB = cluster_im2(atX,atY,3);
       if ((pixR ~= 0) &  (pixB ~= 0) & (pixG ~= 0))
         count2R = count2R + 1;
         meanR2R = uint32(meanR2R) + uint32(pixR);
         meanR2G = uint32(meanR2G) + uint32(pixG);
         meanR2B = uint32(meanR2B) + uint32(pixB);
         if count2R > count2Rmax
           count2Rmax = count2R;
         end
         minR2R = pixR;
         minR2G = pixG;
         minR2B = pixB;
         if pixR < minR2R
           minR2R = pixR;
           minR2G = pixG;
           minR2B = pixB;
         elseif pixR > maxR2R
           maxR2R = pixR;
           maxR2G = pixG;
           maxR2B = pixB;
         end
       end
    end
  end
meanR2R, meanR2G, meanR2B
meanR2R = meanR2R/count2Rmax;
meanR2G = meanR2G/count2Rmax;
meanR2B = meanR2B/count2Rmax;
minR2R;, minR2G;, minR2B;
maxR2R;, maxR2G;, maxR2B;
meanR2R;, meanR2G;, meanR2B;
count2R;, count2Rmax;
atX = 1;, atY = 1;
pixR = 0;, pixG = 0;, pixB = 0;
```

```
minR4R = 0;, minG4G = 0;, minB4B = 0;
maxR4R = 0;, maxG4G = 0;, maxB4B = 0;
minR4G = 0;, minR4B = 0;, maxR4G = 0;, maxR4B = 0;
minG4R = 0;, minG4B = 0;, maxG4R = 0;, maxG4B = 0;
minB4R = 0;, minB4G = 0;, maxB4R = 0;, maxB4G = 0;
count4 = 0;, count4R = 0;, count4Area = 0;
meanR4R = 0;, meanR4G = 0;, meanR4B = 0;
count4Rmax =0;
  for atX = 1:1:nX
    atY = 1;
    for atY = 1:1:nY
        pixR = color_image_after(atX,atY,1);
        pixG = color_image_after(atX,atY,2);
        pixB = color_image_after(atX,atY,3);
        if ((pixR ~= 0) &  (pixB ~= 0) & (pixG ~= 0))
          count4R = count4R + 1;
          meanR4R = uint32(meanR4R) + uint32(pixR);
          meanR4G = uint32(meanR4G) + uint32(pixG);
          meanR4B = uint32(meanR4B) + uint32(pixB);
          if count4R > count4Rmax
            count4Rmax = count4R;
          end
          minR4R = pixR;
          minR4G = pixG;
          minR4B = pixB;
          if pixR < minR4R
            minR4R = pixR;
            minR4G = pixG;
            minR4B = pixB;
          elseif pixR > maxR4R
            maxR4R = pixR;
            maxR4G = pixG;
            maxR4B = pixB;
          end
        end
    end
  end
meanR4R, meanR4G, meanR4B
meanR4R = meanR4R/count4Rmax;
meanR4G = meanR4G/count4Rmax;
meanR4B = meanR4B/count4Rmax;
minR4R;, minR4G;, minR4B;
maxR4R;, maxR4G;, maxR4B;
meanR4R;, meanR4G;, meanR4B;
count4R;, count4Rmax;
countAll = count1R + count2R;
count1Area = round((count1R / countAll) * 100);
count2Area = round((count2R / countAll) * 100);
count4Area = round((count4R / count4R) * 100);
count1R;,count2R;,count4R;
   axes(handles.axes2)
   imshow(cluster_im1), title('Objects in Cluster 1');
   clust1 = cluster_im1;
   axes(handles.axes3)
```

```
    imshow(cluster_im2), title('Objects in Cluster 2');
    clust2 = cluster_im2;
    if count1Area < count2Area;
        minEqualR1R = minR2R;, minEqualR1G = minR2G;, minEqualR1B = minR2B;
        maxEqualR1R = maxR2R;, maxEqualR1G = maxR2G;, maxEqualR1B = maxR2B;
        meanEqualR1R = meanR2R;, meanEqualR1G = meanR2G;, meanEqualR1B = meanR2B;
        minR2R = minR1R;, minR2G = minR1G;, minR2B = minR1B;
        maxR2R = maxR1R;, maxR2G = maxR1G;, maxR2B = maxR1B;
        meanR2R = meanR1R;, meanR2G = meanR1G;, meanR2B = meanR1B;
        minR1R = minEqualR1R;, minR1G = minEqualR1G;, minR1B = minEqualR1B;
        maxR1R = maxEqualR1R;, maxR1G = maxEqualR1G;, maxR1B = maxEqualR1B;
        meanR1R = meanEqualR1R;, meanR1G = meanEqualR1G;, meanR1B = meanEqualR1B;
        Equal = count2Area;
        count2Area = count1Area;
        count1Area = Equal;
        axes(handles.axes2)
        imshow(cluster_im2), title('Objects in Cluster 1');
        clust1 = cluster_im2;
        axes(handles.axes3)
        imshow(cluster_im1), title('Objects in Cluster 2');
        clust2 = cluster_im1;
    else
        axes(handles.axes2)
        imshow(cluster_im1), title('Objects in Cluster 1');
        clust1 = cluster_im1;
        axes(handles.axes3)
        imshow(cluster_im2), title('Objects in Cluster 2');
        clust2 = cluster_im2;
    end
result1 = 0;
if  ((meanR1B >= 44)&(meanR1B <= 48))
    result1 = 1;
elseif  ((meanR1B >= 49)&(meanR1B <= 51))
    if ((meanR1G >= 121)&(meanR1G <= 123))&((meanR1R >= 164)&(meanR1R <= 166))
        result1 = 3;
    elseif ((meanR1G >= 125)&(meanR1G <= 127))&((meanR1R >= 165)&(meanR1R <= 167))
        result1 = 2;
    else result1 = 3;
    end
elseif  ((meanR1B >= 52)&(meanR1B <= 59))
    if ((meanR1B >= 52)&(meanR1B <= 53))
      if ((meanR1G >= 114)&(meanR1G <= 115))&((meanR1R >= 160)&(meanR1R <= 163))
            result1 = 2;
  elseif ((meanR1G >= 116)&(meanR1G <= 117))&((meanR1R >= 162)&(meanR1R <= 163))
            result1 = 5;
  elseif ((meanR1G >= 118)&(meanR1G <= 125))&((meanR1R >= 161)&(meanR1R <= 170))
            result1 = 2;
        else result1 = 2;
        end
    elseif  ((meanR1B >= 54)&(meanR1B <= 55))
    if ((meanR1G >= 117)&(meanR1G <= 123))&((meanR1R >= 160)&(meanR1R <= 165))
            result1 = 2;
        else result1 = 3;
        end
    elseif  meanR1B == 56
```

```
    if ((meanR1G >= 115)&(meanR1G <= 116))
        result1 = 5;
    elseif ((meanR1G >= 119)&(meanR1G <= 125))
        if (meanR1R >= 154)&(meanR1R <= 156)|(meanR1R >= 162)&(meanR1R <= 167)
            result1 = 2;
        else result1 = 3;
        end
    else result1 = 2;
    end
elseif  meanR1B == 57
    if ((meanR1G >= 116)&(meanR1G <= 117))
        if (meanR1R >= 157)&(meanR1R <= 160)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 119)&(meanR1G <= 120))
        if (meanR1R >= 158)&(meanR1R <= 164)
            result1 = 2;
        elseif (meanR1R >= 165)&(meanR1R <= 168)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 121)&(meanR1G <= 126))
        if (meanR1R >= 161)&(meanR1R <= 163)
            result1 = 3;
        elseif (meanR1R >= 164)&(meanR1R <= 167)
            result1 = 2;
        elseif (meanR1R >= 168)&(meanR1R <= 170)
            result1 = 3;
        else result1 = 3;
        end
    else result1 = 3;
    end
elseif  meanR1B == 58
    if ((meanR1G >= 109)&(meanR1G <= 112))
        if (meanR1R >= 161)&(meanR1R <= 163)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 115)&(meanR1G <= 120))
        if (meanR1R >= 156)&(meanR1R <= 162)
            result1 = 3;
        elseif (meanR1R >= 163)&(meanR1R <= 168)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 121)&(meanR1G <= 128))
        if (meanR1R >= 160)&(meanR1R <= 163)
            result1 = 3;
        elseif (meanR1R >= 164)&(meanR1R <= 170)
            result1 = 2;
        else result1 = 3;
        end
    else result1 = 3;
    end
```

```
    elseif  meanR1B == 59
        if ((meanR1G >= 111)&(meanR1G <= 114))
            if (meanR1R >= 157)&(meanR1R <= 161)
                result1 = 3;
            elseif (meanR1R >= 162)&(meanR1R <= 163)
                result1 = 5;
            else result1 = 3;
            end
        elseif ((meanR1G >= 115)&(meanR1G <= 117))
            if (meanR1R >= 161)&(meanR1R <= 163)
                result1 = 3;
            else result1 = 3;
            end
        elseif ((meanR1G >= 118)&(meanR1G <= 119))
            if (meanR1R >= 159)&(meanR1R <= 163)
                result1 = 3;
            elseif (meanR1R >= 164)&(meanR1R <= 165)
                result1 = 5;
            elseif (meanR1R >= 166)&(meanR1R <= 169)
                result1 = 3;
            else result1 = 3;
            end
        elseif ((meanR1G >= 120)&(meanR1G <= 121))
            if (meanR1R >= 162)&(meanR1R <= 163)
                result1 = 5;
            elseif (meanR1R >= 164)&(meanR1R <= 169)
                result1 = 3;
            else result1 = 3;
            end
        elseif ((meanR1G >= 122)&(meanR1G <= 123))
            if (meanR1R >= 160)&(meanR1R <= 162)
                result1 = 2;
            else result1 = 3;
            end
        elseif ((meanR1G >= 124)&(meanR1G <= 129))
            if (meanR1R >= 161)&(meanR1R <= 166)
                result1 = 3;
            elseif (meanR1R >= 167)&(meanR1R <= 169)
                result1 = 2;
            else result1 = 3;
            end
        else result1 = 3;
        end
    else result1 = 0;
    end
elseif  ((meanR1B >= 60)&(meanR1B <= 69))
    if  meanR1B == 60
        if (meanR1G >= 112)&(meanR1G <= 116)
            if (meanR1R >= 153)&(meanR1R <= 155)|(meanR1R >= 165)&(meanR1R <= 169)
                result1 = 5;
            elseif (meanR1R >= 156)&(meanR1R <= 164)
                result1 = 3;
            else result1 = 3;
            end
        elseif ((meanR1G >= 117)&(meanR1G <= 118))
```

```
        if (meanR1R >= 160)&(meanR1R <= 165)
            result1 = 3;
        else result1 = 3;
        end
    elseif meanR1G == 119
        if (meanR1R >= 161)&(meanR1R <= 162)
            result1 = 5;
        elseif (meanR1R >= 166)&(meanR1R <= 168)
            result1 = 3;
        else result1 = 3;
        end
    elseif ((meanR1G >= 120)&(meanR1G <= 122))
        if (meanR1R >= 161)&(meanR1R <= 163)
            result1 = 3;
        elseif (meanR1R >= 164)&(meanR1R <= 165)
            result1 = 2;
        else result1 = 3;
        end
    elseif meanR1G == 123
        if (meanR1R >= 158)&(meanR1R <= 160)
            result1 = 2;
        elseif (meanR1R >= 161)&(meanR1R <= 163)
            result1 = 3;
        else result1 = 3;
        end
    elseif meanR1G == 124
        if (meanR1R >= 162)&(meanR1R <= 164)
            result1 = 2;
        else result1 = 3;
        end
    elseif ((meanR1G >= 125)&(meanR1G <= 129))
        if (meanR1R >= 161)&(meanR1R <= 169)
            result1 = 3;
        else result1 = 3;
        end
    else result1 = 3;
    end
elseif  meanR1B == 61
    if (meanR1G >= 111)&(meanR1G <= 113)
        if (meanR1R >= 159)&(meanR1R <= 162)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 114)&(meanR1G <= 120))
        if (meanR1R >= 156)&(meanR1R <= 164)
            result1 = 3;
        elseif (meanR1R >= 165)&(meanR1R <= 171)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 121)&(meanR1G <= 124))
        if (meanR1R >= 151)&(meanR1R <= 162)
            result1 = 3;
        elseif (meanR1R >= 163)&(meanR1R <= 164)
            result1 = 5;
```

```
            elseif (meanR1R >= 165)&(meanR1R <= 169)
                result1 = 3;
            else result1 = 3;
            end
        elseif ((meanR1G >= 125)&(meanR1G <= 131))
            if (meanR1R >= 164)&(meanR1R <= 165)
                result1 = 2;
            elseif (meanR1R >= 166)&(meanR1R <= 167)
                result1 = 5;
            elseif (meanR1R >= 168)&(meanR1R <= 169)
                result1 = 3;
            else result1 = 3;
            end
        elseif ((meanR1G >= 132)&(meanR1G <= 134))
            if (meanR1R >= 174)&(meanR1R <= 176)
                result1 = 2;
            else result1 = 3;
            end
        else result1 = 3;
        end
    elseif  meanR1B == 62
        if (meanR1G >= 108)&(meanR1G <= 110)
            if (meanR1R >= 159)&(meanR1R <= 161)
                result1 = 5;
            else result1 = 3;
            end
        elseif (meanR1G >= 111)&(meanR1G <= 113)
            if (meanR1R >= 150)&(meanR1R <= 152)
                result1 = 3;
            else result1 = 3;
            end
        elseif (meanR1G >= 115)&(meanR1G <= 120)
            if (meanR1R >= 158)&(meanR1R <= 164)
                result1 = 3;
            elseif (meanR1R >= 165)&(meanR1R <= 169)
                result1 = 5;
            else result1 = 3;
            end
        elseif meanR1G == 121
            if (meanR1R >= 159)&(meanR1R <= 163)
                result1 = 2;
            elseif (meanR1R >= 166)&(meanR1R <= 168)
                result1 = 3;
            else result1 = 3;
            end
        elseif meanR1G == 122
            if (meanR1R >= 159)&(meanR1R <= 161)
                result1 = 3;
            elseif (meanR1R >= 162)&(meanR1R <= 163)
                result1 = 2;
            else result1 = 3;
            end
        elseif (meanR1G >= 123)&(meanR1G <= 124)
            if (meanR1R >= 161)&(meanR1R <= 163)
                result1 = 3;
```

```
        else result1 = 3;
        end
    elseif meanR1G == 125
        if (meanR1R >= 163)&(meanR1R <= 164)
            result1 = 5;
        else result1 = 3;
        end
    elseif meanR1G == 126
        if (meanR1R >= 167)&(meanR1R <= 168)
            result1 = 5;
        else result1 = 3;
        end
    elseif meanR1G == 127
        if (meanR1R >= 163)&(meanR1R <= 165)
            result1 = 2;
        elseif (meanR1R >= 166)&(meanR1R <= 168)
            result1 = 3;
        else result1 = 3;
        end
    elseif meanR1G == 128
        if (meanR1R >= 164)&(meanR1R <= 165)
            result1 = 5;
        else result1 = 3;
        end
    else result1 = 3;
    end
elseif  meanR1B == 63
    if ((meanR1G >= 113)&(meanR1G <= 115))
        if (meanR1R >= 161)&(meanR1R <= 163)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 117)&(meanR1G <= 118))
        if (meanR1R >= 156)&(meanR1R <= 160)
            result1 = 5;
        elseif (meanR1R >= 161)&(meanR1R <= 163)
            result1 = 3;
        else result1 = 3;
        end
    elseif ((meanR1G >= 119)&(meanR1G <= 121))
        if (meanR1R >= 153)&(meanR1R <= 160)
            result1 = 3;
        elseif (meanR1R >= 161)&(meanR1R <= 162)
            result1 = 5;
        elseif(meanR1R >= 164)&(meanR1R <= 166)
            result1 = 5;
        elseif(meanR1R >= 167)&(meanR1R <= 168)
            result1 = 3;
        else result1 = 3;
        end
    elseif ((meanR1G >= 122)&(meanR1G <= 126))
        if (meanR1R >= 156)&(meanR1R <= 157)
            result1 = 2;
        elseif(meanR1R >= 158)&(meanR1R <= 161)
            result1 = 3;
```

```
        elseif(meanR1R >= 162)&(meanR1R <= 163)
            result1 = 2;
        elseif (meanR1R >= 164)&(meanR1R <= 166)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 127)&(meanR1G <= 128))
        if (meanR1R >= 162)&(meanR1R <= 165)
            result1 = 3;
        else result1 = 3;
        end
    else result1 = 3;
    end
elseif ((meanR1B >= 64)&(meanR1B <= 65))
    if ((meanR1G >= 113)&(meanR1G <= 116))
        if (meanR1R >= 157)&(meanR1R <= 163)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 117)&(meanR1G <= 118))
        if (meanR1R >= 161)&(meanR1R <= 164)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 119)&(meanR1G <= 120))
        if meanR1R == 160
            result1 = 5;
        elseif (meanR1R >= 161)&(meanR1R <= 163)
            result1 = 3;
        else result1 = 3;
        end
    elseif ((meanR1G >= 122)&(meanR1G <= 129))
        if (meanR1R >= 160)&(meanR1R <= 161)
            result1 = 3;
        elseif (meanR1R >= 162)&(meanR1R <= 163)
            result1 = 2;
        elseif (meanR1R == 166)
            result1 = 5;
        else result1 = 3;
        end
    else result1 = 3;
    end
elseif ((meanR1B >= 66)&(meanR1B <= 67))
    if ((meanR1G >= 102)&(meanR1G <= 104))
        if (meanR1R >= 154)&(meanR1R <= 158)
            result1 = 4;
        else result1 = 3;
        end
    elseif ((meanR1G >= 105)&(meanR1G <= 108))
        if (meanR1R >= 152)&(meanR1R <= 154)
            result1 = 3;
        elseif (meanR1R >= 155)&(meanR1R <= 157)
            result1 = 4;
        elseif (meanR1R >= 158)&(meanR1R <= 159)
            result1 = 3;
```

```
        else result1 = 3;
        end
    elseif ((meanR1G >= 109)&(meanR1G <= 115))
        if (meanR1R >= 158)&(meanR1R <= 159)
            result1 = 5;
        elseif (meanR1R >= 161)&(meanR1R <= 162)
            result1 = 4;
        else result1 = 3;
        end
    elseif ((meanR1G >= 116)&(meanR1G <= 122))
        if (meanR1R >= 146)&(meanR1R <= 148)
            result1 = 2;
        elseif (meanR1R >= 152)&(meanR1R <= 156)
            result1 = 3;
        elseif (meanR1R >= 157)&(meanR1R <= 162)
            result1 = 5;
        elseif (meanR1R >= 163)&(meanR1R <= 164)
            result1 = 2;
        else result1 = 3;
        end
    elseif ((meanR1G >= 123)&(meanR1G <= 125))
        if (meanR1R >= 155)&(meanR1R <= 156)
            result1 = 3;
        elseif (meanR1R >= 157)&(meanR1R <= 158)
            result1 = 2;
        elseif (meanR1R >= 159)&(meanR1R <= 166)
            result1 = 3;
        else result1 = 3;
        end
    elseif ((meanR1G >= 126)&(meanR1G <= 130))
        if (meanR1R >= 161)&(meanR1R <= 164)|(meanR1R >= 167)&(meanR1R <= 169)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 131)&(meanR1G <= 133))
        if (meanR1R >= 160)&(meanR1R <= 162)
            result1 = 4;
        else result1 = 3;
        end
    else result1 = 3;
    end
elseif  meanR1B == 68
    if ((meanR1G >= 108)&(meanR1G <= 109))
        if (meanR1R >= 159)&(meanR1R <= 161)
            result1 = 4;
        else result1 = 3;
        end
    elseif ((meanR1G >= 110)&(meanR1G <= 112))
        if (meanR1R >= 157)&(meanR1R <= 159)
            result1 = 3;
        else result1 = 3;
        end
    elseif ((meanR1G >= 113)&(meanR1G <= 117))
        if (meanR1R >= 156)&(meanR1R <= 158)
            result1 = 5;
```

```
        elseif (meanR1R >= 162)&(meanR1R <= 164)
            result1 = 4;
        else result1 = 3;
        end
    elseif ((meanR1G >= 118)&(meanR1G <= 122))
        if (meanR1R >= 149)&(meanR1R <= 153)
            result1 = 2;
        else result1 = 3;
        end
    elseif ((meanR1G >= 123)&(meanR1G <= 128))
        if (meanR1R >= 160)&(meanR1R <= 162)
            result1 = 2;
        elseif (meanR1R >= 163)&(meanR1R <= 164)
            result1 = 3;
        elseif (meanR1R >= 165)&(meanR1R <= 166)
            result1 = 5;
        else result1 = 3;
        end
    else result1 = 3;
    end
else %%meanR1B == 69
    if ((meanR1G >= 103)&(meanR1G <= 107))
        if (meanR1R >= 144)&(meanR1R <= 146)
            result1 = 3;
        elseif (meanR1R >= 157)&(meanR1R <= 159)
            result1 = 4;
        else result1 = 3;
        end
    elseif ((meanR1G >= 108)&(meanR1G <= 115))
        if (meanR1R >= 154)&(meanR1R <= 155)
            result1 = 5;
        elseif (meanR1R >= 156)&(meanR1R <= 157)
            result1 = 3;
        elseif (meanR1R >= 160)&(meanR1R <= 165)
            result1 = 4;
        else result1 = 3;
        end
    elseif ((meanR1G >= 116)&(meanR1G <= 119))
        if (meanR1R >= 147)&(meanR1R <= 149)
            result1 = 2;
        else result1 = 3;
        end
    elseif ((meanR1G >= 120)&(meanR1G <= 121))
        if (meanR1R >= 146)&(meanR1R <= 148)
            result1 = 3;
        else result1 = 3;
        end
    elseif ((meanR1G >= 122)&(meanR1G <= 125))
        if (meanR1R >= 152)&(meanR1R <= 158)
            result1 = 2;
        elseif (meanR1R >= 165)&(meanR1R <= 168)
            result1 = 5;
        else result1 = 3;
        end
    elseif ((meanR1G >= 126)&(meanR1G <= 130))
```

```
            if (meanR1R >= 162)&(meanR1R <= 164)
                result1 = 2;
            elseif (meanR1R >= 167)&(meanR1R <= 169)
                result1 = 5;
            else result1 = 3;
            end
        else result1 = 3;
        end
    end
elseif  ((meanR1B >= 70)&(meanR1B <= 79))

    if  ((meanR1B >= 70)&(meanR1B <= 71))
        if (meanR1G >= 100)&(meanR1G <= 103)
            if (meanR1R >= 151)&(meanR1R <= 155)
                result1 = 3;
            else result1 = 3;
            end
        elseif (meanR1G >= 104)&(meanR1G <= 110)
            if (meanR1R >= 154)&(meanR1R <= 157)
                result1 = 3;
            elseif (meanR1R >= 158)&(meanR1R <= 159)
                result1 = 4;
            else result1 = 3;
            end
        elseif (meanR1G >= 111)&(meanR1G <= 117)
            if (meanR1R >= 151)&(meanR1R <= 155)
                result1 = 5;
            elseif (meanR1R >= 156)&(meanR1R <= 159)
                result1 = 3;
            elseif (meanR1R >= 160)&(meanR1R <= 164)
                result1 = 4;
            else result1 = 3;
            end
        elseif (meanR1G >= 118)&(meanR1G <= 123)
            if (meanR1R >= 147)&(meanR1R <= 153)
                result1 = 2;
            elseif (meanR1R >= 156)&(meanR1R <= 158)
                result1 = 4;
            elseif (meanR1R >= 159)&(meanR1R <= 160)
                result1 = 5;
            else result1 = 3;
            end
        elseif (meanR1G >= 124)&(meanR1G <= 127)
            if (meanR1R >= 155)&(meanR1R <= 156)
                result1 = 3;
            elseif (meanR1R >= 157)&(meanR1R <= 159)
                result1 = 2;
            elseif (meanR1R >= 160)&(meanR1R <= 161)
                result1 = 3;
            else result1 = 3;
            end
        elseif (meanR1G >= 128)&(meanR1G <= 130)
            if (meanR1R >= 159)&(meanR1R <= 161)
                result1 = 2;
            else result1 = 3;
```

```
        end
      elseif (meanR1G >= 136)&(meanR1G <= 138)
        if (meanR1R >= 182)&(meanR1R <= 184)
          result1 = 3;
        else result1 = 3;
        end
      else result1 = 3;
      end
    elseif ((meanR1B >= 72)&(meanR1B <= 73))
      if (meanR1G >= 97)&(meanR1G <= 101)
        if (meanR1R >= 147)&(meanR1R <= 149)
          result1 = 4;
        elseif (meanR1R >= 150)&(meanR1R <= 153)
          result1 = 3;
        else result1 = 3;
        end
      elseif (meanR1G >= 102)&(meanR1G <= 105)
        if (meanR1R >= 147)&(meanR1R <= 149)
          result1 = 3;
        else result1 = 3;
        end
      elseif (meanR1G >= 106)&(meanR1G <= 110)
        if (meanR1R >= 150)&(meanR1R <= 151)|(meanR1R >= 155)&(meanR1R <= 159)
          result1 = 4;
        else result1 = 3;
        end
      elseif (meanR1G >= 118)&(meanR1G <= 122)
        if (meanR1R >= 146)&(meanR1R <= 149)
          result1 = 2;
        elseif (meanR1R >= 151)&(meanR1R <= 153)
          result1 = 4;
        else result1 = 3;
        end
      elseif (meanR1G >= 123)&(meanR1G <= 129)
        if (meanR1R >= 151)&(meanR1R <= 155)
          result1 = 3;
        elseif (meanR1R >= 156)&(meanR1R <= 159)
          result1 = 4;
        elseif (meanR1R >= 160)&(meanR1R <= 161)
          result1 = 2;
        elseif (meanR1R >= 163)&(meanR1R <= 165)
          result1 = 5;
        else result1 = 3;
        end
      elseif (meanR1G >= 130)&(meanR1G <= 132)
        if (meanR1R >= 161)&(meanR1R <= 163)
          result1 = 2;
        else result1 = 3;
        end
      else result1 = 3;
      end
    elseif ((meanR1B >= 74)&(meanR1B <= 75))
      if ((meanR1G >= 102)&(meanR1G <= 114))
        if (meanR1R >= 149)&(meanR1R <= 151)
          result1 = 4;
```

```
            else result1 = 4;
            end
        elseif ((meanR1G >= 108)&(meanR1G <= 113))
            if (meanR1R >= 152)&(meanR1R <= 154)
                result1 = 3;
            elseif (meanR1R >= 156)&(meanR1R <= 160)
                result1 = 4;
            else result1 = 4;
            end
        elseif ((meanR1G >= 116)&(meanR1G <= 122))
            if (meanR1R >= 150)&(meanR1R <= 156)|(meanR1R >= 160)&(meanR1R <= 161)
                result1 = 5;
            elseif (meanR1R >= 157)&(meanR1R <= 159)
                result1 = 3;
            else result1 = 4;
            end
        elseif ((meanR1G >= 123)&(meanR1G <= 125))
            if (meanR1R >= 153)&(meanR1R <= 154)
                result1 = 3;
            elseif (meanR1R >= 155)&(meanR1R <= 158)
                result1 = 4;
            else result1 = 4;
            end
        elseif ((meanR1G >= 126)&(meanR1G <= 129))
            if (meanR1R >= 155)&(meanR1R <= 159)
                result1 = 2;
            else result1 = 4;
            end
        elseif ((meanR1G >= 130)&(meanR1G <= 134))
            if (meanR1R >= 159)&(meanR1R <= 161)
                result1 = 4;
            else result1 = 4;
            end
        else result1 = 4;
        end
    elseif ((meanR1B >= 76)&(meanR1B <= 77))
        if ((meanR1G >= 99)&(meanR1G <= 105))
            if (meanR1R >= 146)&(meanR1R <= 150)
                result1 = 4;
            else result1 = 4;
            end
        elseif ((meanR1G >= 115)&(meanR1G <= 120))
            if (meanR1R >= 148)&(meanR1R <= 149)
                result1 = 4;
            elseif (meanR1R >= 150)&(meanR1R <= 153)
                result1 = 5;
            elseif (meanR1R >= 155)&(meanR1R <= 158)
                result1 = 3;
            else result1 = 4;
            end
        elseif ((meanR1G >= 121)&(meanR1G <= 123))
            if (meanR1R >= 150)&(meanR1R <= 154)
                result1 = 4;
            elseif (meanR1R >= 155)&(meanR1R <= 156)
                result1 = 5;
```

```
            elseif (meanR1R >= 159)&(meanR1R <= 163)
                result1 = 3;
            else result1 = 4;
            end
        elseif ((meanR1G >= 124)&(meanR1G <= 126))
            if (meanR1R >= 154)&(meanR1R <= 156)
                result1 = 4;
            else result1 = 4;
            end
        elseif ((meanR1G >= 134)&(meanR1G <= 140))
            if (meanR1R >= 160)&(meanR1R <= 165)
                result1 = 4;
            else result1 = 4;
            end
        else result1 = 4;
        end
    else %%((meanR1B >= 78)&(meanR1B <= 79))
        if ((meanR1G >= 93)&(meanR1G <= 95))
            if (meanR1R >= 141)&(meanR1R <= 143)
                result1 = 3;
            else result1 = 4;
            end
        elseif ((meanR1G >= 96)&(meanR1G <= 98))
            if (meanR1R >= 140)&(meanR1R <= 142)
                result1 = 4;
            elseif (meanR1R >= 145)&(meanR1R <= 147)
                result1 = 3;
            else result1 = 4;
            end
        elseif ((meanR1G >= 104)&(meanR1G <= 110))
            if (meanR1R >= 153)&(meanR1R <= 158)
                result1 = 4;
            else result1 = 4;
            end
        elseif ((meanR1G >= 116)&(meanR1G <= 117))
            if (meanR1R >= 153)&(meanR1R <= 155)
                result1 = 4;
            else result1 = 4;
            end
        elseif ((meanR1G >= 118)&(meanR1G <= 120))
            if (meanR1R >= 148)&(meanR1R <= 152)
                result1 = 5;
            elseif (meanR1R >= 154)&(meanR1R <= 157)
                result1 = 3;
            else result1 = 4;
            end
        elseif ((meanR1G >= 121)&(meanR1G <= 124))
            if (meanR1R >= 151)&(meanR1R <= 154)
                result1 = 4;
            else result1 = 4;
            end
        elseif ((meanR1G >= 131)&(meanR1G <= 133))
            if (meanR1R >= 158)&(meanR1R <= 160)
                result1 = 4;
            else result1 = 4;
```

```
            end
        else result1 = 4;
        end
    end
elseif  ((meanR1B >= 80)&(meanR1B <= 89))
    if  ((meanR1B >= 80)&(meanR1B <= 81))
        if (meanR1G >= 96)&(meanR1G <= 98)
            if (meanR1R >= 158)&(meanR1R <= 159)
                result1 = 4;
            else result1 = 4;
            end
        elseif (meanR1G >= 107)&(meanR1G <= 110)
            if (meanR1R >= 155)&(meanR1R <= 159)
                result1 = 4;
            else result1 = 4;
            end
        elseif (meanR1G >= 115)&(meanR1G <= 118)
            if (meanR1R >= 151)&(meanR1R <= 154)
                result1 = 4;
            else result1 = 4;
            end
        elseif (meanR1G >= 119)&(meanR1G <= 123)
            if (meanR1R >= 146)&(meanR1R <= 148)
                result1 = 5;
            elseif (meanR1R >= 152)&(meanR1R <= 157)
                result1 = 4;
            elseif (meanR1R >= 158)&(meanR1R <= 159)
                result1 = 3;
            else result1 = 4;
            end
        elseif (meanR1G >= 124)&(meanR1G <= 127)
            if (meanR1R >= 152)&(meanR1R <= 157)
                result1 = 4;
            elseif (meanR1R >= 158)&(meanR1R <= 159)
                result1 = 5;
            else result1 = 4;
            end
        elseif (meanR1G >= 128)&(meanR1G <= 130)
            if (meanR1R >= 162)&(meanR1R <= 164)
                result1 = 4;
            else result1 = 4;
            end
        elseif (meanR1G >= 131)&(meanR1G <= 134)
            if (meanR1R >= 153)&(meanR1R <= 156)
                result1 = 4;
            else result1 = 4;
            end
        elseif (meanR1G >= 135)&(meanR1G <= 143)
            if (meanR1R >= 159)&(meanR1R <= 165)
                result1 = 4;
            else result1 = 4;
            end
        else result1 = 4;
        end
    elseif  ((meanR1B >= 82)&(meanR1B <= 83))
```

```
        if (meanR1G >= 98)&(meanR1G <= 100)
            if (meanR1R >= 140)&(meanR1R <= 142)
                result1 = 4;
            else result1 = 4;
            end
        elseif (meanR1G >= 103)&(meanR1G <= 105)
            if (meanR1R >= 147)&(meanR1R <= 149)
                result1 = 4;
            else result1 = 4;
            end
        elseif (meanR1G >= 116)&(meanR1G <= 119)
            if (meanR1R >= 145)&(meanR1R <= 148)
                result1 = 5;
            elseif (meanR1R >= 149)&(meanR1R <= 156)
                result1 = 4;
            else result1 = 4;
            end
        elseif (meanR1G >= 120)&(meanR1G <= 129)
            if (meanR1R >= 147)&(meanR1R <= 160)
                result1 = 4;
            else result1 = 4;
            end
        elseif (meanR1G >= 134)&(meanR1G <= 139)
            if (meanR1R >= 157)&(meanR1R <= 169)
                result1 = 4;
            else result1 = 4;
            end
        else result1 = 4;
        end
    elseif ((meanR1B >= 84)&(meanR1B <= 85))
        if (meanR1G >= 99)&(meanR1G <= 101)
            if (meanR1R >= 150)&(meanR1R <= 152)
                result1 = 4;
            else result1 = 4;
            end
        elseif (meanR1G >= 113)&(meanR1G <= 118)
            if (meanR1R >= 144)&(meanR1R <= 149)
                result1 = 5;
            else result1 = 4;
            end
        elseif (meanR1G >= 119)&(meanR1G <= 120)
            if (meanR1R >= 145)&(meanR1R <= 147)
                result1 = 4;
            elseif (meanR1R >= 148)&(meanR1R <= 149)
                result1 = 5;
            elseif (meanR1R >= 150)&(meanR1R <= 152)
                result1 = 3;
            else result1 = 4;
            end
        elseif (meanR1G >= 121)&(meanR1G <= 126)
            if (meanR1R >= 145)&(meanR1R <= 155)
                result1 = 4;
            else result1 = 4;
            end
        elseif (meanR1G >= 127)&(meanR1G <= 128)
```

```
        if (meanR1R >= 150)&(meanR1R <= 151)
            result1 = 2;
        else result1 = 4;
        end
    elseif (meanR1G >= 134)&(meanR1G <= 139)
        if (meanR1R >= 154)&(meanR1R <= 160)
            result1 = 4;
        else result1 = 4;
        end
    else result1 = 4;
    end
elseif ((meanR1B >= 86)&(meanR1B <= 87))
    if (meanR1G >= 100)&(meanR1G <= 102)
        if (meanR1R >= 139)&(meanR1R <= 141)
            result1 = 4;
        else result1 = 4;
        end
    elseif (meanR1G >= 112)&(meanR1G <= 114)
        if (meanR1R >= 137)&(meanR1R <= 139)
            result1 = 5;
        else result1 = 4;
        end
    elseif (meanR1G >= 117)&(meanR1G <= 120)
        if (meanR1R >= 145)&(meanR1R <= 149)
            result1 = 5;
        else result1 = 4;
        end
    elseif (meanR1G >= 121)&(meanR1G <= 124)
        if (meanR1R >= 143)&(meanR1R <= 146)
            result1 = 4;
        elseif (meanR1R == 147)
            result1 = 5;
        elseif (meanR1R >= 148)&(meanR1R <= 150)
            result1 = 4;
        elseif (meanR1R >= 151)&(meanR1R <= 153)
            result1 = 5;
        elseif (meanR1R >= 154)&(meanR1R <= 158)
            result1 = 4;
        else result1 = 4;
        end
    elseif (meanR1G >= 125)&(meanR1G <= 127)
        if (meanR1R >= 152)&(meanR1R <= 156)
            result1 = 4;
        else result1 = 4;
        end
    else result1 = 4;
    end
else %%((meanR1B >= 88)&(meanR1B <= 89))
    if (meanR1G >= 118)&(meanR1G <= 119)
        if (meanR1R >= 140)&(meanR1R <= 155)
            result1 = 4;
        else result1 = 4;
        end
    elseif (meanR1G >= 130)&(meanR1G <= 132)
        if (meanR1R >= 153)&(meanR1R <= 155)
```

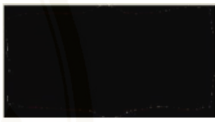
```
            result1 = 2;
         else result1 = 4;
         end
      else result1 = 4;
      end
   end
elseif  ((meanR1B >= 90)&(meanR1B <= 99))
   if  ((meanR1B >= 90)&(meanR1B <= 99))
      if (meanR1G >= 117)&(meanR1G <= 126)
         if (meanR1R >= 141)&(meanR1R <= 143)
            result1 = 5;
         elseif (meanR1R >= 144)&(meanR1R <= 151)
            result1 = 4;
         else result1 = 4;
         end
      elseif (meanR1G >= 130)&(meanR1G <= 132)
         if (meanR1R >= 152)&(meanR1R <= 154)
            result1 = 2;
         else result1 = 4;
         end
      else result1 = 4;
      end
   else %%((meanR1B >= 96)&(meanR1B <= 99))
      result1 = 4;
   end
else result1 = 6;
end
result4 = result1;
result2 = 0;
set(handles.hideResult1,'visible','off');
set(handles.hideResult2,'visible','off');
set(handles.hideResult3,'visible','off');
set(handles.hideResult4,'visible','off');
set(handles.hideResult11,'visible','off')
set(handles.hideResult21,'visible','off')
set(handles.hideResult31,'visible','off')
set(handles.hideResult41,'visible','off')
set(handles.minR1R, 'String', minR1R);
set(handles.minR1G, 'String', minR1G);
set(handles.minR1B, 'String', minR1B);
set(handles.meanR1R, 'String', meanR1R);
set(handles.meanR1G, 'String', meanR1G);
set(handles.meanR1B, 'String', meanR1B);
set(handles.maxR1R, 'String', maxR1R);
set(handles.maxR1G, 'String', maxR1G);
set(handles.maxR1B, 'String', maxR1B);
set(handles.count1Area, 'String', count1Area);
set(handles.result1, 'String', result1);
set(handles.minR2R, 'String', minR2R);
set(handles.minR2G, 'String', minR2G);
set(handles.minR2B, 'String', minR2B);
set(handles.meanR2R, 'String', meanR2R);
set(handles.meanR2G, 'String', meanR2G);
set(handles.meanR2B, 'String', meanR2B);
set(handles.maxR2R, 'String', maxR2R);
```

```
set(handles.maxR2G, 'String', maxR2G);
set(handles.maxR2B, 'String', maxR2B);
set(handles.count2Area, 'String', count2Area);
set(handles.result2, 'String', result2);
set(handles.minR4R, 'String', minR4R);
set(handles.minR4G, 'String', minR4G);
set(handles.minR4B, 'String', minR4B);
set(handles.meanR4R, 'String', meanR4R);
set(handles.meanR4G, 'String', meanR4G);
set(handles.meanR4B, 'String', meanR4B);
set(handles.maxR4R, 'String', maxR4R);
set(handles.maxR4G, 'String', maxR4G);
set(handles.maxR4B, 'String', maxR4B);
set(handles.count4Area, 'String', count4Area);
set(handles.result4, 'String', result4);
result = {'Result';
minR1R;, minR1G;, minR1B;
maxR1R;, maxR1G;, maxR1B;
minR2R;, minR2G;, minR2B;
maxR2R;, maxR2G;, maxR2B;
minR4R;, minR4G;, minR4B;
maxR4R;, maxR4G;, maxR4B;
   result1;, meanR1R;, meanR1G;, meanR1B;
   result2;, meanR2R;, meanR2G;, meanR2B;
   result4;, meanR4R;, meanR4G;, meanR4B;
   count1Area; count2Area; count4Area;};
toc
tKmean = toc
set(handles.btnKmean,'enable','on')
set(handles.btnGrade1to5,'enable','off')
set(handles.btnGrade3to5,'enable','off')
set(handles.btnCropAuto,'enable','off')
set(handles.btnCropManual,'enable','off')
set(handles.rss1to5,'enable','off')
set(handles.rss3to5,'enable','off')
set(handles.select,'enable','off')

save color_image_after;
save tKmean;
save result;
%% Kmean --Stop
```

APPENDIX B
RIBBED SMOKED SHEET (RSS)
TESTING SET IMAGES

B1. Type 1 : Ribbed Smoked Sheet (RSS) images grade 1 of 8 images

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
1. TEST_ RSS1_ 001.JPG				
2. TEST_ RSS1_ 002.JPG				
3. TEST_ RSS1_ 003.JPG				
4. TEST_ RSS1_ 004.JPG				
5. TEST_ RSS1_ 005.JPG				
6. TEST_ RSS1_ 006.JPG				

B1. Type 1 : Ribbed Smoked Sheet (RSS) images grade 1 of 8 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
7. TEST_ RSS1_ 007.JPG				
8. TEST_ RSS1_ 008.JPG				

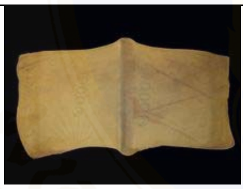
B1. Type 2 : Ribbed Smoked Sheet (RSS) images grade 2 of 61 images

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
1. TEST_ RSS2_ 001.JPG				
2. TEST_ RSS2_ 002.JPG				
3. TEST_ RSS2_ 003.JPG				
4. TEST_ RSS2_ 004.JPG				

B1. Type 2 : Ribbed Smoked Sheet (RSS) images grade 2 of 61 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
5. TEST_ RSS2_ 005.JPG				
6. TEST_ RSS2_ 006.JPG				
7. TEST_ RSS2_ 007.JPG				
8. TEST_ RSS2_ 008.JPG				
9. TEST_ RSS2_ 009.JPG				
10. TEST_ RSS2_ 010.JPG				
11. TEST_ RSS2_ 011.JPG				
12. TEST_ RSS2_ 012.JPG				

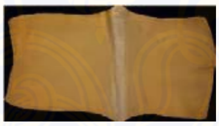
B1. Type 2 : Ribbed Smoked Sheet (RSS) images grade 2 of 61 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
13. TEST_ RSS2_ 013.JPG				
14. TEST_ RSS2_ 014.JPG				
15. TEST_ RSS2_ 015.JPG				
16. TEST_ RSS2_ 016.JPG				
17. TEST_ RSS2_ 017.JPG				
18. TEST_ RSS2_ 018.JPG				
19. TEST_ RSS2_ 019.JPG				
20. TEST_ RSS2_ 020.JPG				

B1. Type 2 : Ribbed Smoked Sheet (RSS) images grade 2 of 61 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
21. TEST_ RSS2_ 021.JPG				
22. TEST_ RSS2_ 022.JPG				
23. TEST_ RSS2_ 023.JPG				
24. TEST_ RSS2_ 024.JPG				
25. TEST_ RSS2_ 025.JPG				
26. TEST_ RSS2_ 026.JPG				
27. TEST_ RSS2_ 027.JPG				
28. TEST_ RSS2_ 028.JPG				

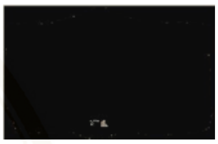
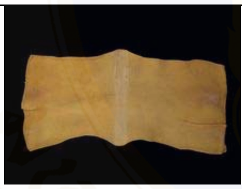
B1. Type 2 : Ribbed Smoked Sheet (RSS) images grade 2 of 61 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
29. TEST_ RSS2_ 029.JPG				
30. TEST_ RSS2_ 030.JPG				
31. TEST_ RSS2_ 031.JPG				
32. TEST_ RSS2_ 032.JPG				
33. TEST_ RSS2_ 033.JPG				
34. TEST_ RSS2_ 034.JPG				
35. TEST_ RSS2_ 035.JPG				
36. TEST_ RSS2_ 036.JPG				

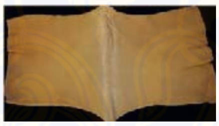
B1. Type 2 : Ribbed Smoked Sheet (RSS) images grade 2 of 61 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
37. TEST_ RSS2_ 037.JPG				
38. TEST_ RSS2_ 038.JPG				
39. TEST_ RSS2_ 039.JPG				
40. TEST_ RSS2_ 040.JPG				
41. TEST_ RSS2_ 041.JPG				
42. TEST_ RSS2_ 042.JPG				
43. TEST_ RSS2_ 043.JPG				
44. TEST_ RSS2_ 044.JPG				

B1. Type 2 : Ribbed Smoked Sheet (RSS) images grade 2 of 61 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
45. TEST_ RSS2_ 045.JPG				
46. TEST_ RSS2_ 046.JPG				
47. TEST_ RSS2_ 047.JPG				
48. TEST_ RSS2_ 048.JPG				
49. TEST_ RSS2_ 049.JPG				
50. TEST_ RSS2_ 050.JPG				
51. TEST_ RSS2_ 051.JPG				
52. TEST_ RSS2_ 052.JPG				

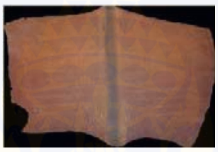
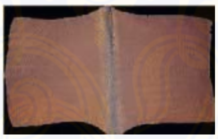
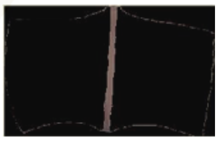
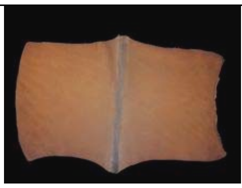
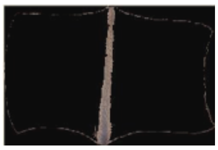
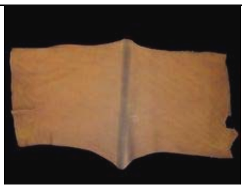
B1. Type 2 : Ribbed Smoked Sheet (RSS) images grade 2 of 61 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
53. TEST_ RSS2_ 053.JPG				
54. TEST_ RSS2_ 054.JPG				
55. TEST_ RSS2_ 055.JPG				
56. TEST_ RSS2_ 056.JPG				
57. TEST_ RSS2_ 057.JPG				
58. TEST_ RSS2_ 058.JPG				
59. TEST_ RSS2_ 059.JPG				
60. TEST_ RSS2_ 060.JPG				

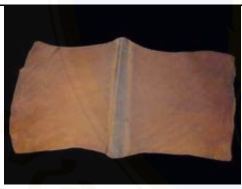
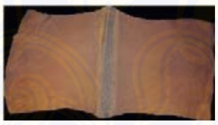
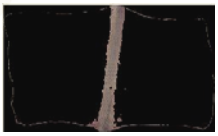
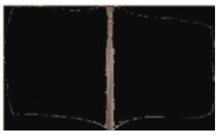
B1. Type 2 : Ribbed Smoked Sheet (RSS) images grade 2 of 61 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
61. TEST_ RSS2_ 061.JPG				

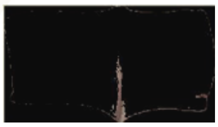
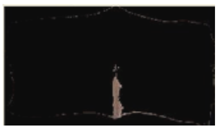
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
1. TEST_ RSS3_ 001.JPG				
2. TEST_ RSS3_ 002.JPG				
3. TEST_ RSS3_ 003.JPG				
4. TEST_ RSS3_ 004.JPG				
5. TEST_ RSS3_ 005.JPG				
6. TEST_ RSS3_ 006.JPG				

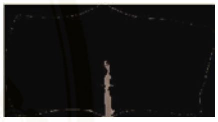
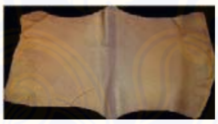
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
7. TEST_ RSS3_ 007.JPG				
8. TEST_ RSS3_ 008.JPG				
9. TEST_ RSS3_ 009.JPG				
10. TEST_ RSS3_ 010.JPG				
11. TEST_ RSS3_ 011.JPG				
12. TEST_ RSS3_ 012.JPG				
13. TEST_ RSS3_ 013.JPG				
14. TEST_ RSS3_ 014.JPG				

B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
15. TEST_ RSS3_ 015.JPG				
16. TEST_ RSS3_ 016.JPG				
17. TEST_ RSS3_ 017.JPG				
18. TEST_ RSS3_ 018.JPG				
19. TEST_ RSS3_ 019.JPG				
20. TEST_ RSS3_ 020.JPG				
21. TEST_ RSS3_ 021.JPG				
22. TEST_ RSS3_ 022.JPG				

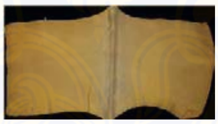
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
23. TEST_ RSS3_ 023.JPG				
24. TEST_ RSS3_ 024.JPG				
25. TEST_ RSS3_ 025.JPG				
26. TEST_ RSS3_ 026.JPG				
27. TEST_ RSS3_ 027.JPG				
28. TEST_ RSS3_ 028.JPG				
29. TEST_ RSS3_ 029.JPG				
30. TEST_ RSS3_ 030.JPG				

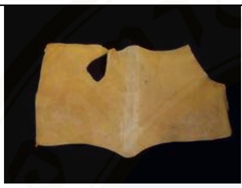
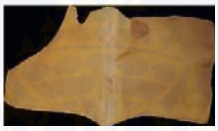
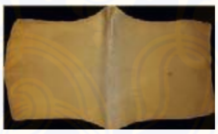
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
31. TEST_ RSS3_ 031.JPG				
32. TEST_ RSS3_ 032.JPG				
33. TEST_ RSS3_ 033.JPG				
34. TEST_ RSS3_ 034.JPG				
35. TEST_ RSS3_ 035.JPG				
36. TEST_ RSS3_ 036.JPG				
37. TEST_ RSS3_ 037.JPG				
38. TEST_ RSS3_ 038.JPG				

B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
39. TEST_ RSS3_ 039.JPG				
40. TEST_ RSS3_ 040.JPG				
41. TEST_ RSS3_ 041.JPG				
42. TEST_ RSS3_ 042.JPG				
43. TEST_ RSS3_ 043.JPG				
44. TEST_ RSS3_ 044.JPG				
45. TEST_ RSS3_ 045.JPG				
46. TEST_ RSS3_ 046.JPG				

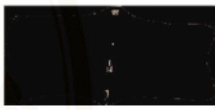
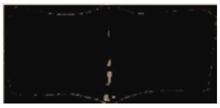
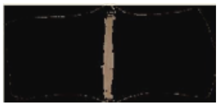
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
47. TEST_ RSS3_ 047.JPG				
48. TEST_ RSS3_ 048.JPG				
49. TEST_ RSS3_ 049.JPG				
50. TEST_ RSS3_ 050.JPG				
51. TEST_ RSS3_ 051.JPG				
52. TEST_ RSS3_ 052.JPG				
53. TEST_ RSS3_ 053.JPG				
54. TEST_ RSS3_ 054.JPG				

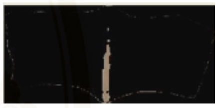
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
55. TEST_ RSS3_ 055.JPG				
56. TEST_ RSS3_ 056.JPG				
57. TEST_ RSS3_ 057.JPG				
58. TEST_ RSS3_ 058.JPG				
59. TEST_ RSS3_ 059.JPG				
60. TEST_ RSS3_ 060.JPG				
61. TEST_ RSS3_ 061.JPG				
62. TEST_ RSS3_ 062.JPG				

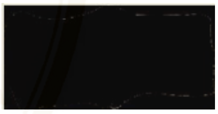
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
63. TEST_ RSS3_ 063.JPG				
64. TEST_ RSS3_ 064.JPG				
65. TEST_ RSS3_ 065.JPG				
66. TEST_ RSS3_ 066.JPG				
67. TEST_ RSS3_ 067.JPG				
68. TEST_ RSS3_ 068.JPG				
69. TEST_ RSS3_ 069.JPG				
70. TEST_ RSS3_ 070.JPG				

B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 717 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
71. TEST_ RSS3_ 071.JPG				
72. TEST_ RSS3_ 072.JPG				
73. TEST_ RSS3_ 073.JPG				
74. TEST_ RSS3_ 074.JPG				
75. TEST_ RSS3_ 075.JPG				
76. TEST_ RSS3_ 076.JPG				
77. TEST_ RSS3_ 077.JPG				
78. TEST_ RSS3_ 078.JPG				

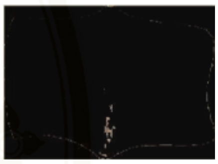
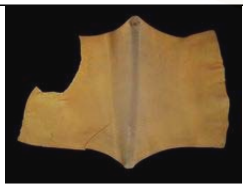
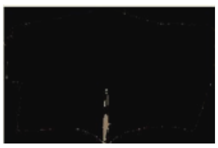
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
79. TEST_ RSS3_ 079.JPG				
80. TEST_ RSS3_ 080.JPG				
81. TEST_ RSS3_ 081.JPG				
82. TEST_ RSS3_ 082.JPG				
83. TEST_ RSS3_ 083.JPG				
84. TEST_ RSS3_ 084.JPG				
85. TEST_ RSS3_ 085.JPG				
86. TEST_ RSS3_ 086.JPG				

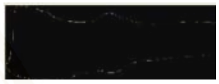
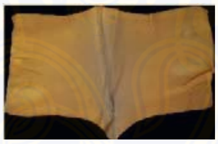
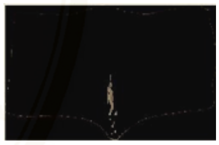
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
87. TEST_ RSS3_ 087.JPG				
88. TEST_ RSS3_ 088.JPG				
89. TEST_ RSS3_ 089.JPG				
90. TEST_ RSS3_ 090.JPG				
91. TEST_ RSS3_ 091.JPG				
92. TEST_ RSS3_ 092.JPG				
93. TEST_ RSS3_ 093.JPG				
94. TEST_ RSS3_ 094.JPG				

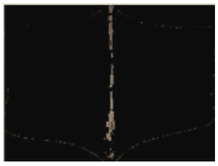
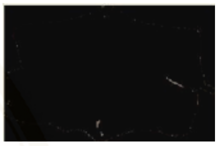
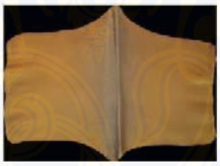
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
95. TEST_ RSS3_ 095.JPG				
96. TEST_ RSS3_ 096.JPG				
97. TEST_ RSS3_ 097.JPG				
98. TEST_ RSS3_ 098.JPG				
99. TEST_ RSS3_ 099.JPG				
100. TEST_ RSS3_ 100.JPG				
101. TEST_ RSS3_ 101.JPG				
102. TEST_ RSS3_ 102.JPG				

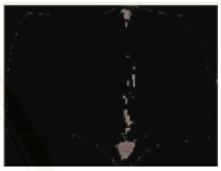
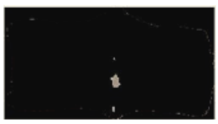
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
103. TEST_ RSS3_ 103.JPG				
104. TEST_ RSS3_ 104.JPG				
105. TEST_ RSS3_ 105.JPG				
106. TEST_ RSS3_ 106.JPG				
107. TEST_ RSS3_ 107.JPG				
108. TEST_ RSS3_ 108.JPG				
109. TEST_ RSS3_ 109.JPG				
110. TEST_ RSS3_ 110.JPG				

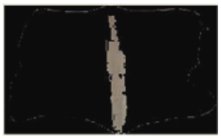
B1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
111. TEST_ RSS3_ 111.JPG				
112. TEST_ RSS3_ 112.JPG				
113. TEST_ RSS3_ 113.JPG				
114. TEST_ RSS3_ 114.JPG				
115. TEST_ RSS3_ 115.JPG				
116. TEST_ RSS3_ 116.JPG				
117. TEST_ RSS3_ 117.JPG				

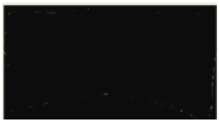
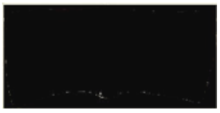
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
1. TEST_ RSS4_ 001.JPG				
2. TEST_ RSS4_ 002.JPG				
3. TEST_ RSS4_ 003.JPG				
4. TEST_ RSS4_ 004.JPG				
5. TEST_ RSS4_ 005.JPG				
6. TEST_ RSS4_ 006.JPG				
7. TEST_ RSS4_ 007.JPG				
8. TEST_ RSS4_ 008.JPG				

B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
9. TEST_ RSS4_ 009.JPG				
10. TEST_ RSS4_ 010.JPG				
11. TEST_ RSS4_ 011.JPG				
12. TEST_ RSS4_ 012.JPG				
13. TEST_ RSS4_ 013.JPG				
14. TEST_ RSS4_ 014.JPG				
15. TEST_ RSS4_ 015.JPG				
16. TEST_ RSS4_ 016.JPG				

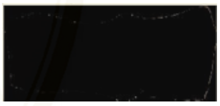
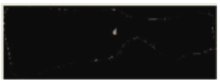
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
17. TEST_ RSS4_ 017.JPG				
18. TEST_ RSS4_ 018.JPG				
19. TEST_ RSS4_ 019.JPG				
20. TEST_ RSS4_ 020.JPG				
21. TEST_ RSS4_ 021.JPG				
22. TEST_ RSS4_ 022.JPG				
23. TEST_ RSS4_ 023.JPG				
24. TEST_ RSS4_ 024.JPG				

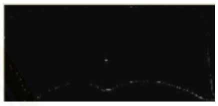
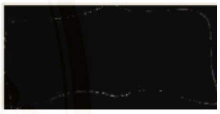
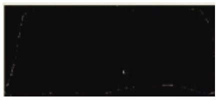
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
25. TEST_ RSS4_ 025.JPG				
26. TEST_ RSS4_ 026.JPG				
27. TEST_ RSS4_ 027.JPG				
28. TEST_ RSS4_ 028.JPG				
29. TEST_ RSS4_ 029.JPG				
30. TEST_ RSS4_ 030.JPG				
31. TEST_ RSS4_ 031.JPG				
32. TEST_ RSS4_ 032.JPG				

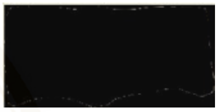
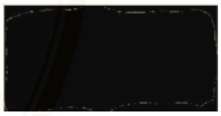
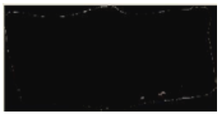
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
33. TEST_ RSS4_ 033.JPG				
34. TEST_ RSS4_ 034.JPG				
35. TEST_ RSS4_ 035.JPG				
36. TEST_ RSS4_ 036.JPG				
37. TEST_ RSS4_ 037.JPG				
38. TEST_ RSS4_ 038.JPG				
39. TEST_ RSS4_ 039.JPG				
40. TEST_ RSS4_ 040.JPG				

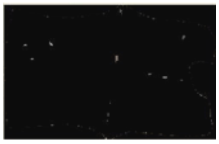
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
41. TEST_ RSS4_ 041.JPG				
42. TEST_ RSS4_ 042.JPG				
43. TEST_ RSS4_ 043.JPG				
44. TEST_ RSS4_ 044.JPG				
45. TEST_ RSS4_ 045.JPG				
46. TEST_ RSS4_ 046.JPG				
47. TEST_ RSS4_ 047.JPG				
48. TEST_ RSS4_ 048.JPG				

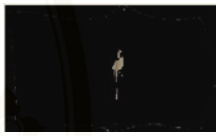
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
49. TEST_ RSS4_ 049.JPG				
50. TEST_ RSS4_ 050.JPG				
51. TEST_ RSS4_ 051.JPG				
52. TEST_ RSS4_ 052.JPG				
53. TEST_ RSS4_ 053.JPG				
54. TEST_ RSS4_ 054.JPG				
55. TEST_ RSS4_ 055.JPG				
56. TEST_ RSS4_ 056.JPG				

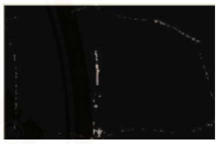
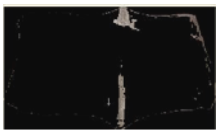
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
57. TEST_ RSS4_ 057.JPG				
58. TEST_ RSS4_ 058.JPG				
59. TEST_ RSS4_ 059.JPG				
60. TEST_ RSS4_ 060.JPG				
61. TEST_ RSS4_ 061.JPG				
62. TEST_ RSS4_ 062.JPG				
63. TEST_ RSS4_ 063.JPG				
64. TEST_ RSS4_ 064.JPG				

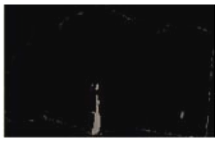
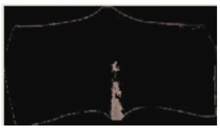
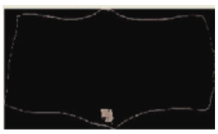
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
65. TEST_ RSS4_ 065.JPG				
66. TEST_ RSS4_ 066.JPG				
67. TEST_ RSS4_ 067.JPG				
68. TEST_ RSS4_ 068.JPG				
69. TEST_ RSS4_ 069.JPG				
70. TEST_ RSS4_ 070.JPG				
71. TEST_ RSS4_ 071.JPG				
72. TEST_ RSS4_ 072.JPG				

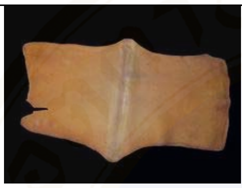
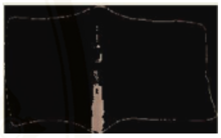
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
73. TEST_ RSS4_ 073.JPG				
74. TEST_ RSS4_ 074.JPG				
75. TEST_ RSS4_ 075.JPG				
76. TEST_ RSS4_ 076.JPG				
77. TEST_ RSS4_ 077.JPG				
78. TEST_ RSS4_ 078.JPG				
79. TEST_ RSS4_ 079.JPG				
80. TEST_ RSS4_ 080.JPG				

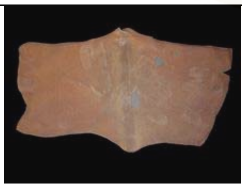
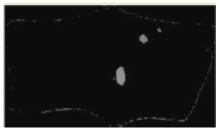
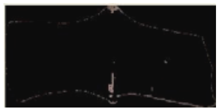
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
81. TEST_ RSS4_ 081.JPG				
82. TEST_ RSS4_ 082.JPG				
83. TEST_ RSS4_ 083.JPG				
84. TEST_ RSS4_ 084.JPG				
85. TEST_ RSS4_ 085.JPG				
86. TEST_ RSS4_ 086.JPG				
87. TEST_ RSS4_ 087.JPG				
88. TEST_ RSS4_ 088.JPG				

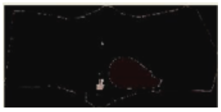
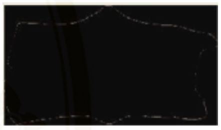
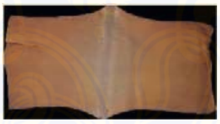
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
89. TEST_ RSS4_ 089.JPG				
90. TEST_ RSS4_ 090.JPG				
91. TEST_ RSS4_ 091.JPG				
92. TEST_ RSS4_ 092.JPG				
93. TEST_ RSS4_ 093.JPG				
94. TEST_ RSS4_ 094.JPG				
95. TEST_ RSS4_ 095.JPG				
96. TEST_ RSS4_ 096.JPG				

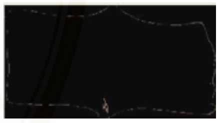
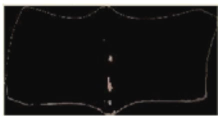
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
97. TEST_ RSS4_ 097.JPG				
98. TEST_ RSS4_ 098.JPG				
99. TEST_ RSS4_ 099.JPG				
100. TEST_ RSS4_ 100.JPG				
101. TEST_ RSS4_ 101.JPG				
102. TEST_ RSS4_ 102.JPG				
103. TEST_ RSS4_ 103.JPG				
104. TEST_ RSS4_ 104.JPG				

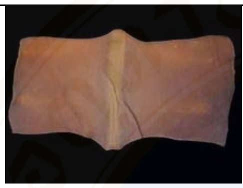
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
105. TEST_ RSS4_ 105.JPG				
106. TEST_ RSS4_ 106.JPG				
107. TEST_ RSS4_ 107.JPG				
108. TEST_ RSS4_ 108.JPG				
109. TEST_ RSS4_ 109.JPG				
110. TEST_ RSS4_ 110.JPG				
111. TEST_ RSS4_ 111.JPG				
112. TEST_ RSS4_ 112.JPG				

B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
113. TEST_ RSS4_ 113.JPG				
114. TEST_ RSS4_ 114.JPG				
115. TEST_ RSS4_ 115.JPG				
116. TEST_ RSS4_ 116.JPG				
117. TEST_ RSS4_ 117.JPG				
118. TEST_ RSS4_ 118.JPG				
119. TEST_ RSS4_ 119.JPG				
120. TEST_ RSS4_ 120.JPG				

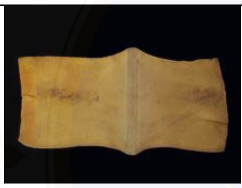
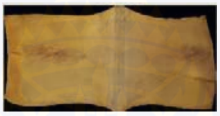
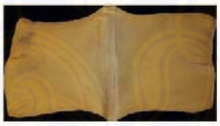
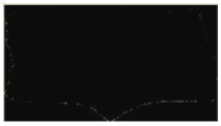
B1. Type 4 : Ribbed Smoked Sheet (RSS) images grade 4 of 122 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
121. TEST_ RSS4_ 121.JPG				
122. TEST_ RSS4_ 122.JPG				

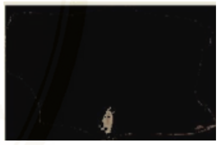
B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
1. TEST_ RSS5_ 001.JPG				
2. TEST_ RSS5_ 002.JPG				
3. TEST_ RSS5_ 003.JPG				
4. TEST_ RSS5_ 004.JPG				
5. TEST_ RSS5_ 005.JPG				

B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
6. TEST_ RSS5_ 006.JPG				
7. TEST_ RSS5_ 007.JPG				
8. TEST_ RSS5_ 008.JPG				
9. TEST_ RSS5_ 009.JPG				
10. TEST_ RSS5_ 010.JPG				
11. TEST_ RSS5_ 011.JPG				
12. TEST_ RSS5_ 012.JPG				
13. TEST_ RSS5_ 013.JPG				

B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
14. TEST_ RSS5_ 014.JPG				
15. TEST_ RSS5_ 015.JPG				
16. TEST_ RSS5_ 016.JPG				
17. TEST_ RSS5_ 017.JPG				
18. TEST_ RSS5_ 018.JPG				
19. TEST_ RSS5_ 019.JPG				
20. TEST_ RSS5_ 020.JPG				
21. TEST_ RSS5_ 021.JPG				

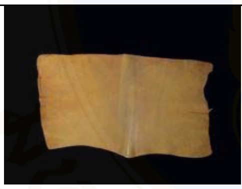
B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
22. TEST_ RSS5_ 022.JPG				
23. TEST_ RSS5_ 023.JPG				
24. TEST_ RSS5_ 024.JPG				
25. TEST_ RSS5_ 025.JPG				
26. TEST_ RSS5_ 026.JPG				
27. TEST_ RSS5_ 027.JPG				
28. TEST_ RSS5_ 028.JPG				
29. TEST_ RSS5_ 029.JPG				

B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
30. TEST_ RSS5_ 030.JPG				
31. TEST_ RSS5_ 031.JPG				
32. TEST_ RSS5_ 032.JPG				
33. TEST_ RSS5_ 033.JPG				
34. TEST_ RSS5_ 034.JPG				
35. TEST_ RSS5_ 035.JPG				
36. TEST_ RSS5_ 036.JPG				
37. TEST_ RSS5_ 037.JPG				

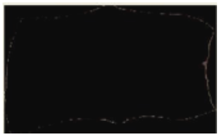
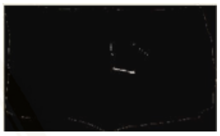
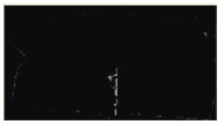
B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
38. TEST_ RSS5_ 038.JPG				
39. TEST_ RSS5_ 039.JPG				
40. TEST_ RSS5_ 040.JPG				
41. TEST_ RSS5_ 041.JPG				
42. TEST_ RSS5_ 042.JPG				
43. TEST_ RSS5_ 043.JPG				
44. TEST_ RSS5_ 044.JPG				
45. TEST_ RSS5_ 045.JPG				

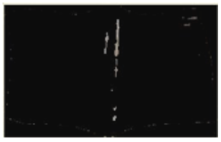
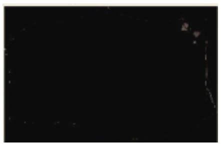
B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
46. TEST_ RSS5_ 046.JPG				
47. TEST_ RSS5_ 047.JPG				
48. TEST_ RSS5_ 048.JPG				
49. TEST_ RSS5_ 049.JPG				
50. TEST_ RSS5_ 050.JPG				
51. TEST_ RSS5_ 051.JPG				
52. TEST_ RSS5_ 052.JPG				
53. TEST_ RSS5_ 053.JPG				

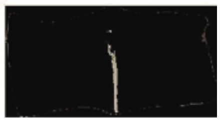
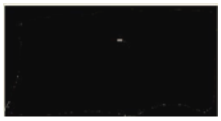
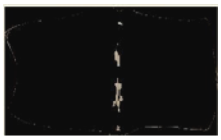
B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
54. TEST_ RSS5_ 054.JPG				
55. TEST_ RSS5_ 055.JPG				
56. TEST_ RSS5_ 056.JPG				
57. TEST_ RSS5_ 057.JPG				
58. TEST_ RSS5_ 058.JPG				
59. TEST_ RSS5_ 059.JPG				
60. TEST_ RSS5_ 060.JPG				
61. TEST_ RSS5_ 061.JPG				

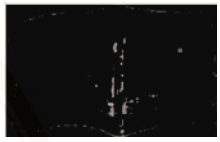
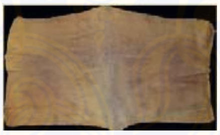
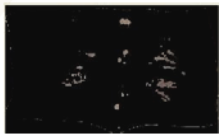
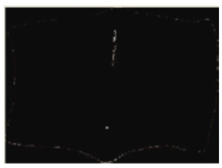
B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
62. TEST_ RSS5_ 062.JPG				
63. TEST_ RSS5_ 063.JPG				
64. TEST_ RSS5_ 064.JPG				
65. TEST_ RSS5_ 065.JPG				
66. TEST_ RSS5_ 066.JPG				
67. TEST_ RSS5_ 067.JPG				
68. TEST_ RSS5_ 068.JPG				
69. TEST_ RSS5_ 069.JPG				

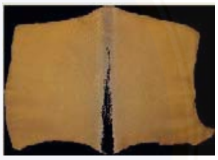
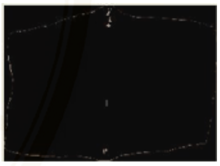
B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
70. TEST_ RSS5_ 070.JPG				
71. TEST_ RSS5_ 071.JPG				
72. TEST_ RSS5_ 072.JPG				
73. TEST_ RSS5_ 073.JPG				
74. TEST_ RSS5_ 074.JPG				
75. TEST_ RSS5_ 075.JPG				
76. TEST_ RSS5_ 076.JPG				
77. TEST_ RSS5_ 077.JPG				

B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
78. TEST_ RSS5_ 078.JPG				
79. TEST_ RSS5_ 079.JPG				
80. TEST_ RSS5_ 080.JPG				
81. TEST_ RSS5_ 081.JPG				
82. TEST_ RSS5_ 082.JPG				
83. TEST_ RSS5_ 083.JPG				
84. TEST_ RSS5_ 084.JPG				
85. TEST_ RSS5_ 085.JPG				

B1. Type 5 : Ribbed Smoked Sheet (RSS) images grade 5 of 90 images (Continued)

Name	Original image	Crop fit image	Cluster # 1	Cluster # 2
86. TEST_ RSS5_ 086.JPG				
87. TEST_ RSS5_ 087.JPG				
88. TEST_ RSS5_ 088.JPG				
89. TEST_ RSS5_ 089.JPG				
90. TEST_ RSS5_ 090.JPG				

APPENDIX C
RIBBED SMOKED SHEET (RSS)
TESTING SET DATA

C1. Type 1 : Ribbed Smoked Sheet (RSS) images testing set data grade 1 of 8 images

TEST_ RSS1	001	002	003	004	005	006	007	008
minR1	125	121	110	149	110	69	137	130
minG1	110	91	88	117	81	45	124	100
minB1	69	53	51	79	51	1	80	66
maxR1	196	203	192	204	209	223	207	209
maxG1	150	145	159	157	163	183	160	171
maxB1	56	72	80	67	78	121	70	122
minR2	2	12	14	1	3	71	56	3
minG2	3	8	2	1	7	51	40	5
minB2	7	5	6	3	10	26	17	4
maxR2	158	156	155	156	171	198	171	170
maxG2	134	131	129	138	146	178	147	144
maxB2	96	100	96	98	116	143	109	109
meanR1	167	166	164	166	165	169	167	168
meanG1	123	123	123	123	121	123	122	124
meanB1	46	48	46	44	47	45	46	48
meanR2	33	31	30	49	54	56	44	57
meanG2	24	23	22	37	40	42	31	42
meanB2	14	14	12	22	26	26	19	26
Clust1_Area	99	99	99	99	98	98	99	99
Clust2_Area	1	1	1	1	2	2	1	1
tLoad	0.04	0.04	0.05	0.04	0.04	0.05	0.05	0.04
tCrop	0.40	0.39	0.40	0.39	0.39	0.40	0.40	0.39
tKmean	9.75	9.25	9.63	8.41	9.29	8.87	9.35	9.18
tSum	10.19	9.68	10.07	8.84	9.73	9.31	9.79	9.61
result	1	1	1	1	1	1	1	1

C1. Type 2 : Ribbed Smoked Sheet (RSS) images testing set data grade 2 of 61 images

TEST_ RSS2	001	002	003	004	005	006	007	008	009	010	011	012	013	014	015	016
minR1	57	145	65	147	77	142	115	76	117	63	91	146	140	143	67	98
minG1	36	129	48	126	63	126	99	52	97	43	77	122	121	119	52	78
minB1	17	95	22	95	36	92	66	28	70	18	50	94	89	95	31	54
maxR1	204	214	193	220	206	193	217	209	208	205	192	209	208	208	207	207
maxG1	173	172	164	198	164	158	161	180	183	170	159	178	170	186	165	167
maxB1	116	98	106	159	88	90	78	122	142	89	92	134	95	145	105	115
minR2	28	2	3	6	45	4	5	1	33	7	6	44	7	41	133	1
minG2	14	2	3	5	37	2	5	2	19	5	9	28	8	32	119	1
minB2	1	2	3	3	14	3	7	7	6	6	16	3	12	15	93	1
maxR2	150	163	146	171	158	161	156	143	184	143	148	160	160	181	166	178
maxG2	128	147	129	150	138	140	135	125	160	124	133	139	141	164	152	158
maxB2	104	114	109	119	111	119	104	105	132	92	104	112	111	138	125	133
meanR1	158	163	156	161	159	159	162	155	162	160	157	154	165	159	162	165
meanG1	125	124	123	126	129	125	124	124	128	125	126	119	130	129	133	134
meanB1	69	61	61	64	69	64	60	68	68	61	70	61	66	75	82	77
meanR2	24	34	22	24	19	25	44	26	23	22	34	19	24	41	23	39
meanG2	17	26	16	18	15	19	33	19	17	16	27	13	18	34	19	31
meanB2	12	18	10	13	10	13	23	14	12	11	19	9	13	27	15	23
Clust1_Area	99	99	99	99	99	99	99	99	99	99	99	99	99	99	99	99
Clust2_Area	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
tLoad	0.05	0.05	0.04	0.04	0.05	0.04	0.05	0.05	0.05	0.04	0.04	0.05	0.04	0.04	0.05	0.04
tCrop	0.41	0.40	0.41	0.41	0.40	0.40	0.41	0.41	0.40	0.40	0.41	0.42	0.40	0.32	0.42	0.41
tKmean	10.48	9.92	10.99	10.77	9.68	9.60	9.83	11.06	8.87	11.44	11.33	11.61	8.76	10.06	11.68	10.15
tSum	10.94	10.36	11.44	11.22	10.13	10.04	10.28	11.51	9.31	11.89	11.79	12.07	9.21	10.42	12.14	10.60
result	**2**	**5**	**3**	**3**	**3**	**3**	**2**	**3**	**2**	**3**	**2**	**3**	**3**	**2**	**4**	**4**

C1. Type 2 : Ribbed Smoked Sheet (RSS) images testing set data grade 2 of 61 images (Continued)

TEST_ RSS2	017	018	019	020	021	022	023	024	025	026	027	028	029	030	031	032
minR1	91	66	88	71	97	154	117	75	60	114	90	121	91	126	103	64
minG1	70	42	65	56	73	129	98	56	36	84	74	101	64	106	82	37
minB1	39	16	34	25	47	98	65	16	8	58	41	68	35	73	51	10
maxR1	204	202	219	194	198	205	195	207	194	215	192	200	196	206	209	211
maxG1	166	169	187	147	164	171	157	180	159	182	139	146	142	177	176	180
maxB1	95	124	130	79	93	100	84	137	93	141	73	58	72	133	125	134
minR2	5	1	2	27	13	2	58	2	103	31	1	60	40	1	5	1
minG2	9	2	2	15	5	2	40	3	88	14	1	45	25	1	5	2
minB2	12	4	2	1	2	2	16	8	55	6	1	24	6	1	7	4
maxR2	151	196	164	153	147	170	150	163	155	212	156	160	162	151	146	174
maxG2	133	173	141	134	127	151	120	138	130	181	129	133	141	115	122	149
maxB2	109	141	110	102	102	121	96	107	100	150	99	103	112	93	94	119
meanR1	163	167	162	165	162	165	161	165	162	165	162	162	166	165	166	165
meanG1	122	122	120	123	122	123	122	122	119	119	120	119	125	122	124	119
meanB1	56	58	57	61	59	58	54	58	56	58	54	53	58	56	56	56
meanR2	26	44	39	34	30	40	31	33	38	77	40	43	39	29	38	35
meanG2	19	33	28	26	22	29	23	24	27	61	30	31	29	20	27	24
meanB2	12	22	18	18	14	18	14	16	18	45	20	20	19	13	17	16
Clust1_Area	99	98	99	98	99	99	99	99	99	99	99	99	99	99	99	99
Clust2_Area	1	2	1	2	1	1	1	1	1	1	1	1	1	1	1	1
tLoad	0.04	0.04	0.04	0.04	0.05	0.04	0.04	0.06	0.04	0.04	0.04	0.05	0.05	0.05	0.04	0.04
tCrop	0.40	0.39	0.40	0.39	0.39	0.39	0.34	0.39	0.39	0.39	0.40	0.41	0.39	0.40	0.39	0.40
tKmean	9.50	9.25	9.81	9.19	9.76	8.44	10.66	8.76	10.09	10.95	10.75	10.77	8.29	9.27	8.49	10.33
tSum	9.95	9.68	10.25	9.62	10.20	8.87	11.04	9.20	10.53	11.38	11.19	11.22	8.73	9.71	8.92	10.77
result	**2**	**2**	**2**	**3**	**2**	**2**	**2**	**2**	**2**	**5**	**2**	**2**	**2**	**2**	**2**	**2**

C1. Type 2 : Ribbed Smoked Sheet (RSS) images testing set data grade 2 of 61 images (Continued)

TEST_ RSS2	033	034	035	036	037	038	039	040	041	042	043	044	045	046	047	048
minR1	104	122	118	137	110	65	87	57	50	124	47	66	93	137	105	66
minG1	78	109	98	120	89	47	71	40	38	110	29	49	85	122	94	53
minB1	51	74	74	94	62	25	48	14	16	81	7	31	64	103	74	21
maxR1	198	193	204	197	188	194	193	184	190	186	187	185	206	196	188	210
maxG1	159	142	173	161	154	165	168	159	170	157	158	152	165	155	158	176
maxB1	94	85	109	101	90	109	112	103	135	99	102	98	103	93	108	128
minR2	58	3	29	3	10	4	20	57	3	2	3	5	1	9	4	2
minG2	42	3	20	3	10	5	14	41	7	2	3	6	1	3	2	4
minB2	26	3	5	3	10	10	2	25	8	2	5	10	1	3	3	3
maxR2	153	167	131	155	138	149	139	129	157	144	144	152	164	159	153	181
maxG2	127	152	111	139	122	133	123	111	141	123	128	136	148	147	140	158
maxB2	104	119	87	113	97	107	100	89	116	102	103	121	132	125	121	127
meanR1	159	164	147	149	148	148	152	149	148	148	146	147	151	152	153	167
meanG1	112	122	119	121	119	120	123	122	118	121	117	118	127	129	130	126
meanB1	51	61	67	68	67	70	71	72	67	71	67	72	85	87	90	63
meanR2	43	39	23	31	26	29	19	18	28	14	23	29	31	40	28	29
meanG2	30	31	18	24	20	22	15	14	21	11	18	23	27	35	24	22
meanB2	20	22	12	16	14	16	11	10	15	8	12	18	22	29	19	15
Clust1_Area	99	98	99	99	99	99	99	99	99	99	99	99	98	99	99	99
Clust2_Area	1	2	1	1	1	1	1	1	1	1	1	1	2	1	1	1
tLoad	0.04	0.05	0.08	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.04	0.04	0.05	0.05	0.04	0.04
tCrop	0.40	0.40	0.44	0.43	0.43	0.43	0.42	0.43	0.42	0.42	0.43	0.43	0.43	0.41	0.41	0.39
tKmean	10.84	10.12	14.14	13.20	13.55	13.76	11.36	13.41	13.68	13.48	13.65	13.29	9.66	11.08	9.94	8.29
tSum	11.29	10.56	14.66	13.68	14.03	14.24	11.83	13.89	14.15	13.95	14.12	13.76	10.13	11.54	10.40	8.72
result	**3**	**5**	**2**	**2**	**2**	**2**	**2**	**2**	**2**	**2**	**2**	**2**	**2**	**4**	**2**	**3**

C1. Type 2 : Ribbed Smoked Sheet (RSS) images testing set data grade 2 of 61 images (Continued)

TEST_ RSS2	049	050	051	052	053	054	055	056	057	058	059	060	061
minR1	130	148	88	62	82	94	91	123	58	84	69	167	116
minG1	115	127	65	48	68	75	71	104	39	64	52	138	91
minB1	82	96	34	13	33	45	44	74	9	29	22	106	60
maxR1	201	216	204	215	212	199	202	227	191	195	213	208	218
maxG1	180	182	171	174	161	168	155	195	156	157	174	174	170
maxB1	137	134	120	112	82	121	87	156	92	94	105	129	106
minR2	32	4	2	5	3	67	58	2	89	3	5	3	10
minG2	20	3	3	3	3	51	42	6	70	5	4	3	6
minB2	4	1	5	14	3	26	17	9	40	2	12	3	5
maxR2	178	169	159	169	161	145	144	184	153	160	178	183	155
maxG2	152	145	133	140	139	121	118	161	134	133	154	163	135
maxB2	127	117	106	110	115	93	91	130	101	103	120	130	108
meanR1	160	162	164	165	167	163	164	163	165	165	163	168	160
meanG1	121	123	122	124	126	123	124	122	123	122	121	127	121
meanB1	62	65	52	48	57	63	62	64	57	58	56	60	62
meanR2	29	28	28	36	31	24	24	29	20	15	34	26	23
meanG2	22	21	21	27	24	18	18	22	15	11	26	20	17
meanB2	15	14	14	16	16	12	12	16	10	7	17	13	11
Clust1_Area	99	99	99	99	99	99	99	99	99	98	99	99	99
Clust2_Area	1	1	1	1	1	1	1	1	1	2	1	1	1
tLoad	0.05	0.05	0.04	0.04	0.05	0.04	0.05	0.05	0.05	0.04	0.04	0.04	0.04
tCrop	0.39	0.39	0.39	0.39	0.40	0.39	0.40	0.39	0.39	0.39	0.40	0.40	0.40
tKmean	9.74	8.69	9.07	9.40	8.10	9.06	9.98	8.75	9.25	8.92	10.29	8.39	10.72
tSum	10.18	9.12	9.51	9.83	8.54	9.49	10.42	9.19	9.69	9.35	10.73	8.83	11.17
result	**2**	**2**	**2**	**1**	**2**	**2**	**3**	**2**	**2**	**2**	**2**	**3**	**2**

C1. Type 3 : Ribbed Smoked Sheet (RSS) images testing set data grade 3 of 117 images

TEST_ RSS3	001	002	003	004	005	006	007	008	009	010	011	012	013	014	015	016
minR1	114	150	97	141	135	139	113	103	126	102	85	132	127	161	99	95
minG1	81	123	68	103	99	109	76	74	95	73	48	101	93	130	67	70
minB1	76	112	54	90	87	85	67	58	74	57	42	80	81	112	54	50
maxR1	227	234	185	195	211	192	188	211	188	205	198	204	192	219	201	186
maxG1	190	202	147	133	174	141	145	177	131	165	166	177	134	187	174	142
maxB1	174	181	111	94	156	94	111	152	60	129	143	150	94	162	147	97
minR2	3	1	1	4	9	6	2	3	15	7	3	1	1	4	10	1
minG2	3	1	1	8	7	7	2	3	7	1	5	1	1	3	4	1
minB2	3	1	1	9	8	9	2	3	5	1	4	1	1	1	4	1
maxR2	162	219	159	172	223	160	168	218	164	188	226	218	217	215	202	162
maxG2	133	193	126	135	196	132	141	192	133	165	198	192	190	189	175	135
maxB2	119	180	119	126	177	108	130	177	113	147	176	169	173	164	154	118
meanR1	148	145	155	155	155	159	150	155	157	148	150	152	149	155	157	153
meanG1	98	99	102	101	102	108	104	102	108	101	99	107	97	105	110	105
meanB1	76	83	73	71	69	66	66	68	65	74	72	69	69	71	71	73
meanR2	101	102	90	95	107	94	97	106	98	106	114	99	107	108	105	89
meanG2	80	85	70	72	88	73	79	86	79	88	94	82	88	88	88	69
meanB2	74	80	64	65	80	62	70	75	66	79	86	75	80	78	76	61
Clust1_Area	91	93	94	95	92	94	94	94	92	92	90	93	90	94	94	94
Clust2_Area	9	7	6	5	8	6	6	6	8	8	10	7	10	6	6	6
tLoad	0.04	0.05	0.05	0.05	0.05	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.04	0.05	0.05
tCrop	0.34	0.40	0.41	0.41	0.41	0.41	0.41	0.40	0.40	0.41	0.41	0.42	0.42	0.42	0.41	0.41
tKmean	10.74	10.49	11.15	11.85	10.96	11.28	13.06	10.48	11.08	11.51	12.21	13.26	11.96	11.87	11.12	11.74
tSum	11.13	10.94	11.61	12.31	11.42	11.73	13.52	10.92	11.53	11.96	12.67	13.72	12.43	12.33	11.58	12.19
result	**4**	**4**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**4**	**3**	**3**	**3**	**3**	**3**	**3**

C1. Type 3 : Ribbed Smoked Sheet (RSS) images testing set data grade 3 of 117 images (Continued)

TEST_ RSS3	017	018	019	020	021	022	023	024	025	026	027	028	029	030	031	032
minR1	147	145	107	91	97	110	132	122	51	119	141	118	125	66	98	156
minG1	119	116	75	63	63	87	115	94	28	93	116	91	102	49	69	131
minB1	98	98	60	49	51	69	89	83	10	78	96	74	84	31	53	111
maxR1	193	210	211	209	205	203	196	193	199	197	208	200	198	202	196	206
maxG1	149	169	181	154	157	177	143	160	165	139	182	148	159	182	145	169
maxB1	104	137	155	90	111	152	75	125	120	76	155	90	118	157	102	140
minR2	6	1	4	25	34	6	1	33	42	6	21	6	13	23	2	1
minG2	5	1	5	12	16	4	1	21	32	2	10	6	9	10	2	1
minB2	10	1	9	6	12	5	1	21	23	1	8	6	6	2	4	1
maxR2	169	175	202	193	165	180	166	186	164	181	204	169	164	196	188	204
maxG2	138	152	175	165	140	155	144	160	139	171	178	152	143	180	163	185
maxB2	120	134	156	151	118	135	130	147	119	146	161	126	122	157	143	170
meanR1	153	153	148	151	156	153	157	148	162	154	157	161	157	155	159	156
meanG1	107	106	103	106	110	113	113	114	123	119	120	122	120	120	117	117
meanB1	68	67	67	69	69	76	73	80	81	82	78	78	80	80	75	76
meanR2	92	52	87	86	72	95	59	105	87	104	106	67	69	98	104	113
meanG2	72	38	66	66	56	77	46	85	70	85	87	53	56	82	82	94
meanB2	61	32	58	58	46	68	39	77	60	74	75	45	46	71	71	82
Clust1_Area	97	98	96	96	98	95	97	94	97	95	96	98	98	96	95	94
Clust2_Area	3	2	4	4	2	5	3	6	3	5	4	2	2	4	5	6
tLoad	0.05	0.04	0.05	0.05	0.04	0.05	0.04	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.04
tCrop	0.40	0.41	0.42	0.41	0.40	0.31	0.40	0.41	0.31	0.41	0.31	0.40	0.41	0.41	0.41	0.40
tKmean	10.10	10.81	12.67	10.93	11.26	10.62	11.52	11.52	9.85	10.87	10.35	9.26	10.78	11.61	10.80	10.70
tSum	10.55	11.27	13.13	11.39	11.71	10.97	11.97	11.97	10.20	11.32	10.71	9.71	11.23	12.07	11.26	11.15
result	**3**	**3**	**3**	**3**	**3**	**4**	**3**	**4**	**4**	**4**	**3**	**4**	**4**	**4**	**3**	**3**

C1. Type 3 : Ribbed Smoked Sheet (RSS) images testing set data grade 3 of 117 images (Continued)

TEST_ RSS3	033	034	035	036	037	038	039	040	041	042	043	044	045	046	047	048
minR1	57	116	105	79	111	119	110	93	144	141	107	102	128	117	108	87
minG1	48	95	79	52	95	93	86	69	115	122	83	83	105	97	84	67
minB1	17	66	52	31	62	66	58	45	85	89	59	51	73	64	58	40
maxR1	193	197	215	205	205	201	191	207	215	198	222	201	227	199	195	209
maxG1	157	157	180	159	165	166	141	176	180	172	196	164	191	171	161	141
maxB1	95	96	124	73	95	100	52	132	112	115	161	86	143	132	98	68
minR2	1	29	45	54	9	1	3	15	6	1	5	48	3	89	30	1
minG2	1	22	27	32	3	1	1	11	4	1	1	34	4	73	21	1
minB2	3	12	7	9	13	1	2	8	5	1	2	8	6	57	12	1
maxR2	172	190	177	177	168	160	162	152	158	147	224	155	175	174	165	177
maxG2	152	175	158	161	145	141	143	131	133	130	198	132	155	148	148	145
maxB2	125	152	126	138	113	109	110	102	102	100	173	101	122	121	122	122
meanR1	159	155	167	164	163	161	160	159	160	163	160	166	164	162	159	165
meanG1	118	112	127	125	123	121	120	122	122	128	124	128	122	123	123	126
meanB1	61	55	64	65	60	60	58	61	58	64	67	59	58	65	64	67
meanR2	37	46	46	45	35	44	47	42	44	35	41	36	35	48	43	37
meanG2	29	35	35	35	26	33	35	32	33	27	32	27	27	37	34	29
meanB2	20	24	24	25	17	23	23	22	22	18	23	17	18	27	25	22
Clust1_Area	99	98	98	98	98	99	99	99	99	98	99	98	98	98	98	97
Clust2_Area	1	2	2	2	2	1	1	1	1	2	1	2	2	2	2	3
tLoad	0.04	0.04	0.05	0.05	0.04	0.04	0.04	0.04	0.05	0.04	0.04	0.04	0.04	0.04	0.04	0.05
tCrop	0.41	0.42	0.39	0.39	0.40	0.40	0.39	0.40	0.40	0.40	0.40	0.39	0.39	0.40	0.40	0.39
tKmean	11.20	12.78	8.66	8.74	9.71	9.93	9.84	10.40	9.76	8.59	9.62	8.68	9.25	10.22	10.34	9.02
tSum	11.65	13.24	9.10	9.18	10.15	10.37	10.27	10.84	10.21	9.04	10.07	9.11	9.69	10.67	10.78	9.47
result	3	3	3	3	3	3	3	3	3	2	3	3	2	2	3	3

C1. Type 3 : Ribbed Smoked Sheet (RSS) images grade 3 of 117 images (Continued)

TEST_ RSS3	049	050	051	052	053	054	055	056	057	058	059	060	061	062	063	064
minR1	139	141	91	134	71	143	150	117	139	101	124	105	109	54	147	53
minG1	115	117	78	111	54	122	120	96	116	74	98	89	83	26	117	25
minB1	89	89	44	80	26	91	86	65	85	47	71	55	50	4	91	3
maxR1	208	213	205	209	205	195	233	200	209	242	213	224	200	208	204	223
maxG1	172	189	170	165	175	157	197	164	175	210	179	191	152	164	155	184
maxB1	122	151	106	94	121	108	145	88	104	153	131	137	67	99	88	125
minR2	2	3	1	1	2	3	33	2	44	31	37	6	4	5	38	10
minG2	2	2	1	1	2	2	24	6	32	22	23	4	3	5	25	6
minB2	2	8	1	1	2	7	7	5	18	13	12	7	1	5	16	5
maxR2	160	154	180	164	162	164	189	174	175	181	177	172	168	155	179	152
maxG2	134	139	159	142	142	140	166	151	154	157	166	138	150	134	152	129
maxB2	109	108	130	118	115	114	135	119	125	123	138	110	112	105	125	98
meanR1	163	160	162	167	161	165	161	162	160	163	164	163	166	163	162	167
meanG1	122	124	121	127	122	122	118	123	119	123	126	125	124	118	119	121
meanB1	67	64	60	65	61	64	60	58	58	56	65	58	48	63	64	61
meanR2	29	35	32	31	34	50	52	42	44	54	47	52	38	45	87	39
meanG2	22	26	24	24	25	39	41	32	33	42	36	40	27	33	69	28
meanB2	16	18	17	17	17	27	30	22	23	29	26	28	17	24	54	19
Clust1_Area	98	99	99	98	99	98	98	98	99	98	98	98	99	98	97	99
Clust2_Area	2	1	1	2	1	2	2	2	1	2	2	2	1	2	3	1
tLoad	0.04	0.04	0.04	0.04	0.05	0.04	0.04	0.04	0.04	0.04	0.04	0.05	0.04	0.04	0.04	0.04
tCrop	0.38	0.40	0.40	0.39	0.39	0.39	0.32	0.40	0.39	0.40	0.39	0.40	0.39	0.39	0.30	0.39
tKmean	7.77	10.21	9.52	8.15	9.85	8.64	9.98	8.92	9.33	9.32	8.77	9.82	8.37	8.57	8.77	8.18
tSum	8.19	10.65	9.96	8.58	10.29	9.07	10.33	9.36	9.76	9.76	9.20	10.27	8.80	9.00	9.11	8.61
result	**2**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**2**	**3**	**3**	**1**	**3**	**3**	**3**

C1. Type 3 : Ribbed Smoked Sheet (RSS) images testing set data grade 3 of 117 images (Continued)

TEST_ RSS3	065	066	067	068	069	070	071	072	073	074	075	076	077	078	079	080
minR1	141	120	159	153	121	149	124	134	157	140	128	156	124	64	144	150
minG1	111	105	129	122	95	118	99	113	127	110	104	126	95	37	115	123
minB1	87	72	101	94	70	90	69	84	103	86	76	98	65	18	85	96
maxR1	203	200	203	204	206	205	224	197	198	197	201	208	207	204	207	194
maxG1	144	147	151	151	149	147	160	131	150	150	154	156	156	150	151	145
maxB1	74	79	76	81	72	65	89	57	78	82	82	81	73	78	77	76
minR2	57	42	1	1	11	2	57	11	2	7	10	6	37	5	1	30
minG2	40	29	5	1	10	2	43	9	3	3	9	5	23	6	1	20
minB2	24	10	4	3	8	2	32	10	5	4	5	1	10	10	1	10
maxR2	172	167	173	190	174	187	175	178	168	170	149	161	159	173	184	187
maxG2	144	145	141	159	152	156	151	152	144	143	125	136	131	152	156	163
maxB2	122	121	116	130	128	128	117	129	116	116	97	114	109	121	132	137
meanR1	162	163	163	166	164	166	165	159	161	163	163	169	166	164	168	161
meanG1	120	121	119	120	118	119	118	118	119	118	118	122	120	117	121	117
meanB1	67	66	63	64	62	60	60	65	65	63	59	60	59	62	59	62
meanR2	64	42	69	117	42	59	101	100	102	60	44	49	45	48	101	81
meanG2	50	31	54	94	30	44	81	84	85	46	33	36	33	34	80	65
meanB2	38	22	42	74	21	32	61	67	68	34	23	24	23	25	61	51
Clust1_Area	98	99	97	94	99	98	97	96	96	98	98	99	98	98	96	98
Clust2_Area	2	1	3	6	1	2	3	4	4	2	2	1	2	2	4	2
tLoad	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.06	0.04
tCrop	0.39	0.40	0.40	0.39	0.40	0.39	0.39	0.39	0.39	0.29	0.39	0.38	0.40	0.40	0.38	0.39
tKmean	9.06	9.52	9.32	8.67	8.51	8.20	8.39	8.25	9.64	9.15	9.72	7.76	9.29	9.25	7.24	8.12
tSum	**9.49**	**9.97**	**9.76**	**9.10**	**8.95**	**8.63**	**8.83**	**8.69**	**10.08**	**9.49**	**10.16**	**8.19**	**9.73**	**9.69**	**7.68**	**8.55**
result	**5**	**2**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**3**

C1. Type 3 : Ribbed Smoked Sheet (RSS) images testing set data grade 3 of 117 images (Continued)

TEST_RSS3	081	082	083	084	085	086	087	088	089	090	091	092	093	094	095	096
minR1	103	61	105	101	78	72	126	149	111	104	124	111	113	122	102	109
minG1	77	33	91	90	52	57	116	132	102	80	100	94	96	94	74	82
minB1	50	11	64	62	35	34	89	106	69	56	74	68	66	72	52	61
maxR1	211	203	188	186	202	186	193	210	182	183	190	192	204	190	192	186
maxG1	153	152	159	150	165	160	163	161	151	141	146	165	160	148	151	152
maxB1	79	86	101	76	85	103	103	92	86	81	83	110	89	76	69	89
minR2	29	2	1	1	1	2	14	7	30	4	27	32	53	3	1	1
minG2	17	2	1	1	2	4	7	1	26	2	22	22	41	4	3	2
minB2	3	2	3	1	4	3	1	1	14	3	16	12	17	6	2	4
maxR2	167	157	170	142	172	184	153	156	163	150	147	149	187	187	168	185
maxG2	137	131	153	128	154	164	141	139	146	130	123	132	176	172	148	170
maxB2	111	104	125	101	130	140	117	119	116	103	99	106	148	149	121	147
meanR1	162	162	148	154	154	151	151	153	157	153	155	156	161	154	159	156
meanG1	117	114	120	127	123	122	123	121	126	121	122	129	127	122	126	125
meanB1	59	57	65	70	72	71	76	70	69	63	66	71	69	67	68	71
meanR2	36	42	53	36	47	44	56	46	39	51	55	50	42	54	55	56
meanG2	25	30	42	27	35	35	43	34	31	38	41	39	32	43	43	44
meanB2	17	20	32	18	27	26	35	26	23	27	31	29	23	33	34	33
Clust1_Area	99	98	98	99	98	98	98	99	98	98	98	98	98	98	97	98
Clust2_Area	1	2	2	1	2	2	2	1	2	2	2	2	2	2	3	2
tLoad	0.05	0.04	0.05	0.04	0.04	0.05	0.04	0.04	0.04	0.05	0.05	0.04	0.05	0.04	0.04	0.04
tCrop	0.39	0.39	0.42	0.34	0.41	0.42	0.43	0.42	0.40	0.42	0.41	0.41	0.40	0.41	0.40	0.40
tKmean	8.13	9.05	13.87	10.86	11.18	11.85	12.36	12.11	10.66	11.60	11.70	10.78	10.02	11.51	10.63	10.37
tSum	8.57	9.49	14.34	11.23	11.63	12.31	12.84	12.57	11.10	12.07	12.16	11.23	10.47	11.96	11.08	10.82
result	3	3	3	3	3	2	4	2	3	3	3	3	3	3	3	3

C1. Type 3 : Ribbed Smoked Sheet (RSS) images testing set data grade 3 of 117 images (Continued)

TEST_ RSS3	097	098	099	100	101	102	103	104	105	106	107	108	109	110	111	112
minR1	125	130	126	113	111	135	123	149	142	128	122	73	122	64	116	81
minG1	92	111	102	87	94	116	91	123	112	105	92	49	94	49	92	59
minB1	73	81	76	64	64	86	66	88	86	74	68	23	73	20	64	35
maxR1	229	190	207	232	212	210	218	234	200	197	204	206	198	196	197	210
maxG1	200	144	157	196	157	165	189	181	176	148	180	172	155	162	162	148
maxB1	168	69	88	108	75	80	155	103	132	69	146	134	102	101	98	65
minR2	17	4	1	4	1	48	4	2	40	1	5	53	10	3	4	1
minG2	6	10	1	6	1	33	3	1	22	3	5	29	9	7	4	1
minB2	2	6	1	18	1	14	1	7	10	2	5	5	4	10	2	1
maxR2	214	160	165	194	157	157	255	180	177	167	151	177	185	163	165	154
maxG2	189	143	132	170	131	133	229	149	158	144	127	154	161	144	139	141
maxB2	158	113	113	142	106	107	200	121	126	112	99	122	133	111	112	107
meanR1	157	157	157	158	160	160	159	179	158	161	160	160	161	162	157	163
meanG1	113	118	116	118	117	122	116	135	118	118	119	118	116	119	114	121
meanB1	59	60	61	62	61	64	58	73	58	60	58	57	66	59	61	63
meanR2	72	59	59	44	65	71	112	46	82	67	44	43	77	41	85	44
meanG2	55	47	46	34	51	58	93	36	66	53	33	31	62	31	69	33
meanB2	41	34	34	25	38	44	71	27	49	39	23	21	49	22	52	25
Clust1_Area	98	99	98	99	98	98	93	97	98	98	99	99	98	99	97	99
Clust2_Area	2	1	2	1	2	2	7	3	2	2	1	1	2	1	3	1
tLoad	0.04	0.05	0.05	0.05	0.04	0.04	0.05	0.04	0.04	0.04	0.05	0.05	0.04	0.05	0.05	0.04
tCrop	0.40	0.42	0.43	0.41	0.41	0.41	0.41	0.21	0.42	0.41	0.41	0.42	0.40	0.41	0.42	0.41
tKmean	12.50	13.66	13.72	11.42	12.22	11.57	12.15	5.53	13.13	12.18	11.80	12.05	12.75	11.48	13.37	10.68
tSum	12.95	14.12	14.19	11.88	12.68	12.02	12.61	5.79	13.59	12.63	12.26	12.51	13.20	11.94	13.84	11.13
result	3	3	3	3	3	3	3	3	3	3	3	3	5	3	3	3

C1. Type 3 : Ribbed Smoked Sheet (RSS) images testing set data grade 3 of 117 images (Continued)

TEST_RSS3	113	114	115	116	117
minR1	101	105	127	121	67
minG1	70	79	100	97	42
minB1	49	54	71	69	20
maxR1	197	196	195	196	203
maxG1	152	149	157	141	154
maxB1	71	77	94	58	87
minR2	1	3	2	44	32
minG2	3	1	3	34	20
minB2	2	4	5	22	4
maxR2	167	172	155	172	162
maxG2	143	153	139	155	136
maxB2	115	123	106	125	109
meanR1	160	162	161	162	162
meanG1	120	118	120	122	119
meanB1	63	63	61	62	65
meanR2	36	92	38	75	65
meanG2	26	74	28	62	51
meanB2	18	56	20	46	39
Clust1_Area	99	97	99	97	98
Clust2_Area	1	3	1	3	2
tLoad	0.04	0.05	0.06	0.05	0.05
tCrop	0.41	0.41	0.37	0.37	0.41
tKmean	11.21	12.46	12.14	11.47	12.53
tSum	11.66	12.92	12.56	11.89	12.99
result	3	3	3	2	3

C1. Type 4 : Ribbed Smoked Sheet (RSS) images testing set data grade 4 of 122 images

TEST_ RSS4	001	002	003	004	005	006	007	008	009	010	011	012	013	014	015	016
minR1	88	97	124	93	62	142	129	104	66	77	116	155	118	147	49	123
minG1	72	78	98	80	44	123	117	85	44	58	98	134	93	129	33	100
minB1	57	64	85	61	32	108	101	70	30	41	78	115	71	109	17	84
maxR1	221	183	180	194	184	202	174	193	203	203	194	230	195	187	178	187
maxG1	191	148	162	177	145	186	140	172	181	178	144	213	164	167	143	153
maxB1	157	108	142	157	102	161	95	153	158	147	91	187	120	143	101	115
minR2	32	7	9	58	22	10	11	21	4	3	14	2	36	2	1	11
minG2	19	2	4	55	13	5	8	12	4	3	9	2	26	2	1	6
minB2	13	8	1	46	6	2	3	3	4	3	5	2	17	2	3	3
maxR2	212	157	182	197	175	196	164	192	208	182	169	212	176	163	161	156
maxG2	205	138	165	181	164	183	144	176	187	166	151	196	155	147	139	133
maxB2	199	123	147	165	146	164	133	160	168	153	131	173	136	132	125	115
meanR1	143	150	148	146	147	145	144	151	148	153	156	152	160	153	149	150
meanG1	118	123	122	121	121	122	121	118	112	118	119	117	123	120	122	121
meanB1	85	89	89	86	90	89	90	84	75	82	84	80	81	85	87	81
meanR2	113	81	116	115	124	109	81	67	101	65	68	47	40	62	116	98
meanG2	103	69	99	101	112	99	73	57	88	54	57	37	31	52	103	82
meanB2	91	61	91	92	98	89	64	50	76	46	49	30	25	44	90	70
Clust1_Area	94	97	81	90	86	96	97	98	96	98	97	98	98	98	91	96
Clust2_Area	6	3	19	10	14	4	3	2	4	2	3	2	2	2	9	4
tLoad	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.04	0.04	0.05	0.04	0.04	0.04	0.05
tCrop	0.42	0.42	0.42	0.42	0.41	0.42	0.41	0.40	0.41	0.39	0.28	0.40	0.38	0.39	0.41	0.43
tKmean	13.39	13.78	13.82	13.72	12.72	12.90	13.64	9.65	10.59	8.49	9.63	8.78	7.64	9.43	11.33	13.44
tSum	13.85	14.25	14.28	14.19	13.18	13.37	14.10	10.09	11.05	8.93	9.96	9.22	8.07	9.87	11.78	13.91
result	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4

C1. Type 4 : Ribbed Smoked Sheet (RSS) images testing set data grade 4 of 122 images (Continued)

TEST_ RSS4	017	018	019	020	021	022	023	024	025	026	027	028	029	030	031	032
minR1	72	78	97	75	44	46	50	44	122	93	146	108	126	77	116	81
minG1	57	66	80	60	34	32	33	33	104	79	135	91	104	62	99	59
minB1	26	42	50	37	9	3	3	5	80	53	107	65	83	39	69	45
maxR1	216	192	228	213	194	201	208	194	207	220	218	199	202	207	195	205
maxG1	185	167	194	174	164	177	177	167	170	167	183	173	185	173	162	192
maxB1	138	100	120	99	112	105	94	80	92	75	102	124	141	109	83	158
minR2	1	53	7	7	1	2	23	5	2	1	4	71	33	45	2	37
minG2	1	42	7	6	1	2	16	3	5	1	2	59	17	30	1	30
minB2	3	24	7	4	1	2	6	6	10	1	3	43	2	7	6	22
maxR2	179	181	163	190	182	181	169	184	162	158	168	181	186	185	173	179
maxG2	164	167	149	184	170	166	153	175	152	144	152	170	172	175	152	166
maxB2	143	140	120	158	146	135	128	144	127	117	127	140	146	150	121	147
meanR1	155	156	163	161	160	163	160	163	159	159	155	160	156	160	159	153
meanG1	136	134	139	135	137	139	136	139	137	137	132	132	131	134	130	127
meanB1	87	79	79	78	85	78	80	82	85	82	79	75	81	77	65	88
meanR2	26	25	22	25	25	23	24	28	30	32	32	23	20	18	32	37
meanG2	21	21	18	20	20	19	19	23	24	26	26	17	16	14	24	31
meanB2	16	16	13	15	16	14	14	17	17	19	19	12	12	10	16	26
Clust1_Area	99	99	99	99	99	99	99	99	99	99	99	99	99	99	99	98
Clust2_Area	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2
tLoad	0.04	0.05	0.05	0.05	0.04	0.05	0.04	0.04	0.05	0.05	0.05	0.05	0.04	0.05	0.04	0.04
tCrop	0.41	0.41	0.39	0.40	0.39	0.39	0.40	0.39	0.39	0.40	0.40	0.40	0.40	0.40	0.40	0.39
tKmean	10.58	11.01	8.28	9.83	8.92	8.94	9.54	8.28	9.21	8.57	10.31	9.67	9.09	9.59	10.20	9.22
tSum	11.04	11.47	8.72	10.28	9.35	9.38	9.98	8.72	9.65	9.01	10.76	10.11	9.53	10.04	10.65	9.65
result	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**3**	**4**

C1. Type 4 : Ribbed Smoked Sheet (RSS) images testing set data grade 4 of 122 images (Continued)

TEST_ RSS4	033	034	035	036	037	038	039	040	041	042	043	044	045	046	047	048
minR1	130	173	145	124	56	169	92	101	134	136	118	112	100	86	82	46
minG1	117	160	130	111	34	147	75	82	113	122	95	93	82	65	66	37
minB1	98	141	111	92	10	124	57	65	96	95	81	78	58	48	51	20
maxR1	187	199	182	194	209	228	198	214	234	206	184	209	225	220	198	212
maxG1	163	171	160	163	177	201	163	194	214	169	153	186	207	200	165	196
maxB1	129	134	123	109	126	158	97	167	190	117	107	152	185	173	111	170
minR2	3	5	5	7	1	39	3	25	4	1	6	37	1	1	12	1
minG2	3	6	5	6	2	23	2	18	4	1	6	24	1	1	7	1
minB2	3	8	13	2	4	7	7	12	4	1	6	16	1	1	3	1
maxR2	181	179	175	184	198	173	179	212	241	177	168	172	248	170	173	188
maxG2	173	166	160	171	183	152	163	199	224	149	151	157	233	161	158	180
maxB2	154	150	141	152	162	131	140	180	208	135	133	138	214	146	139	159
meanR1	148	149	151	150	166	162	155	149	150	156	150	149	150	154	149	152
meanG1	125	125	128	124	134	134	126	122	124	125	123	121	121	123	123	124
meanB1	90	89	90	85	82	87	83	85	86	83	84	84	83	81	87	87
meanR2	38	36	34	36	45	19	28	52	42	36	42	38	42	42	35	41
meanG2	32	31	29	30	37	15	23	45	34	29	35	31	35	34	29	35
meanB2	27	26	24	24	28	12	17	38	28	23	29	26	28	27	24	29
Clust1_Area	98	97	98	98	97	98	98	97	98	98	98	98	99	98	98	98
Clust2_Area	2	3	2	2	3	2	2	3	2	2	2	2	1	2	2	2
tLoad	0.05	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.05	0.04	0.04	0.04	0.04	0.05	0.04	0.04
tCrop	0.40	0.39	0.40	0.39	0.21	0.18	0.39	0.40	0.40	0.39	0.39	0.39	0.40	0.39	0.39	0.40
tKmean	9.60	9.13	9.29	8.77	5.97	4.23	8.55	7.37	10.28	8.65	8.78	8.27	9.20	7.98	8.71	9.50
tSum	10.05	9.57	9.73	9.20	6.22	4.45	8.99	7.82	10.73	9.08	9.21	8.70	9.64	8.41	9.14	9.94
result	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	5

C1. Type 4 : Ribbed Smoked Sheet (RSS) images testing set data grade 4 of 122 images (Continued)

TEST_RSS4	049	050	051	052	053	054	055	056	057	058	059	060	061	062	063	064
minR1	60	47	101	138	173	75	128	61	152	111	93	137	127	122	87	127
minG1	42	39	79	120	149	67	117	38	136	86	75	122	115	106	71	110
minB1	20	18	58	98	123	48	97	20	113	64	55	99	93	81	48	84
maxR1	200	198	241	200	226	201	185	194	229	187	197	189	194	192	189	199
maxG1	176	160	216	177	185	172	155	158	207	153	178	159	155	166	151	173
maxB1	140	121	175	136	139	132	105	124	184	90	146	121	96	143	78	140
minR2	1	40	3	1	4	1	6	3	6	30	6	5	5	4	25	22
minG2	2	37	3	1	4	1	5	7	8	24	8	5	6	6	13	13
minB2	6	22	1	1	6	1	1	10	7	12	3	5	10	3	1	4
maxR2	177	197	193	183	198	198	199	180	228	156	169	174	158	197	150	180
maxG2	161	185	177	164	181	196	186	154	211	139	148	156	143	179	136	158
maxB2	138	161	152	147	155	173	167	131	193	111	127	134	122	159	110	134
meanR1	155	153	158	154	162	148	148	158	154	158	154	154	152	157	155	157
meanG1	124	124	126	124	128	124	122	122	126	122	125	124	127	123	124	124
meanB1	80	84	79	82	81	90	87	70	82	68	79	79	85	74	72	72
meanR2	47	43	45	56	45	67	46	27	52	31	27	27	84	47	33	34
meanG2	39	35	37	46	37	62	39	22	44	24	22	21	75	39	25	27
meanB2	31	28	28	38	29	54	33	16	37	17	17	16	62	31	19	20
Clust1_Area	97	98	98	98	98	97	98	99	99	99	99	99	97	99	98	99
Clust2_Area	3	2	2	2	2	3	2	1	1	1	1	1	3	1	2	1
tLoad	0.04	0.04	0.04	0.04	0.05	0.05	0.05	0.04	0.04	0.04	0.05	0.05	0.04	0.05	0.05	0.05
tCrop	0.39	0.40	0.38	0.40	0.38	0.31	0.40	0.41	0.41	0.40	0.40	0.40	0.40	0.41	0.41	0.41
tKmean	9.19	9.95	7.32	9.93	7.01	10.78	10.44	11.23	12.03	10.55	10.59	11.21	10.51	11.24	11.43	11.74
tSum	9.62	10.40	7.75	10.37	7.44	11.13	10.89	11.69	12.49	11.00	11.04	11.66	10.95	11.69	11.88	12.19
result	4	4	4	4	4	4	4	4	4	3	4	4	4	4	3	4

C1. Type 4 : Ribbed Smoked Sheet (RSS) images testing set data grade 4 of 122 images (Continued)

TEST_ RSS4	065	066	067	068	069	070	071	072	073	074	075	076	077	078	079	080
minR1	111	124	93	121	136	53	122	79	99	131	73	42	47	113	126	128
minG1	93	107	71	106	118	35	104	62	77	116	59	24	34	92	107	113
minB1	71	87	50	85	96	15	82	42	56	97	46	10	17	75	93	94
maxR1	239	182	189	180	183	213	190	188	197	207	228	192	192	203	202	192
maxG1	208	153	166	149	154	186	156	167	176	190	210	182	161	182	184	175
maxB1	154	121	132	95	120	157	95	140	147	170	190	157	106	155	162	155
minR2	1	9	6	34	3	4	1	2	23	7	12	5	3	11	3	23
minG2	1	3	2	25	1	5	1	7	14	7	7	6	1	10	3	14
minB2	1	3	3	10	2	9	3	10	7	7	3	8	2	6	1	5
maxR2	163	161	165	174	164	168	172	170	166	223	245	206	183	214	205	194
maxG2	149	146	150	168	146	160	167	152	154	207	229	194	172	195	193	177
maxB2	122	125	121	146	122	139	145	128	130	191	213	178	150	178	181	159
meanR1	154	148	156	150	151	157	154	155	154	150	148	149	156	150	146	151
meanG1	124	119	125	123	120	124	125	123	122	122	122	120	127	122	121	125
meanB1	75	75	73	80	75	76	80	77	74	88	90	83	84	83	90	88
meanR2	26	36	68	37	30	40	36	42	30	87	63	81	91	98	104	60
meanG2	21	29	58	31	24	31	30	36	23	75	54	71	81	85	91	51
meanB2	16	23	46	25	18	23	24	28	17	66	48	60	69	74	83	44
Clust1_Area	99	98	98	98	99	99	98	99	99	96	98	97	97	97	94	98
Clust2_Area	1	2	2	2	1	1	2	1	1	4	2	3	3	3	6	2
tLoad	0.04	0.05	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.04	0.05	0.04	0.05
tCrop	0.41	0.42	0.40	0.40	0.41	0.40	0.41	0.39	0.42	0.41	0.41	0.40	0.40	0.41	0.38	0.41
tKmean	11.38	12.35	11.49	11.81	11.00	9.85	10.28	10.99	12.37	10.89	11.79	10.73	8.76	10.97	13.42	11.60
tSum	11.83	12.81	11.94	12.26	11.45	10.29	10.73	11.43	12.83	11.34	12.25	11.18	9.20	11.43	13.85	12.06
result	**3**	**4**	**4**	**4**	**5**	**4**	**4**	**5**	**5**	**4**	**4**	**4**	**4**	**4**	**4**	**4**

C1. Type 4 : Ribbed Smoked Sheet (RSS) images testing set data grade 4 of 122 images (Continued)

TEST_ RSS4	081	082	083	084	085	086	087	088	089	090	091	092	093	094	095	096
minR1	113	146	135	132	134	129	121	119	162	125	148	138	134	132	57	85
minG1	91	129	106	106	106	96	85	87	127	93	112	106	105	100	34	55
minB1	77	109	90	91	95	89	73	76	107	78	96	91	89	77	16	53
maxR1	187	211	205	191	204	210	188	192	199	206	199	189	182	195	228	224
maxG1	170	187	174	140	178	159	161	162	168	179	172	137	152	136	184	195
maxB1	142	161	153	97	151	130	140	136	140	152	143	89	114	76	139	165
minR2	29	10	25	26	5	2	10	8	4	3	7	1	18	4	8	1
minG2	19	6	12	13	5	2	7	7	9	1	7	1	10	4	6	1
minB2	10	5	6	5	5	2	2	3	15	4	9	1	7	6	11	1
maxR2	158	201	195	163	198	179	189	176	192	212	206	155	179	195	178	150
maxG2	141	178	166	142	177	151	167	154	161	188	178	123	168	169	158	126
maxB2	125	164	150	121	156	139	143	133	143	164	154	110	148	146	133	114
meanR1	147	151	159	153	155	156	155	157	164	162	161	158	157	163	157	145
meanG1	120	122	114	107	108	107	108	111	113	116	111	110	109	115	110	101
meanB1	85	88	76	75	83	80	73	81	69	73	69	70	74	69	65	77
meanR2	77	99	81	67	74	94	92	78	94	78	110	64	72	116	52	63
meanG2	67	85	63	51	59	75	73	62	72	61	87	47	57	93	38	47
meanB2	58	75	55	47	55	72	66	57	61	53	74	41	52	79	31	44
Clust1_Area	97	96	97	96	95	94	94	95	96	97	95	97	96	93	98	96
Clust2_Area	3	4	3	4	5	6	6	5	4	3	5	3	4	7	2	4
tLoad	0.05	0.04	0.05	0.04	0.05	0.04	0.05	0.04	0.05	0.05	0.04	0.04	0.04	0.05	0.06	0.05
tCrop	0.42	0.33	0.40	0.41	0.39	0.29	0.41	0.39	0.39	0.40	0.41	0.40	0.39	0.40	0.41	0.40
tKmean	11.94	10.94	10.39	11.47	8.93	8.64	12.03	9.60	8.85	9.94	11.38	10.48	10.02	10.38	11.33	11.45
tSum	12.41	11.31	10.83	11.93	9.37	8.97	12.48	10.04	9.29	10.38	11.83	10.93	10.46	10.82	11.80	11.89
result	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**3**	**4**	**4**	**4**	**4**	**3**	**4**

C1. Type 4 : Ribbed Smoked Sheet (RSS) images testing set data grade 4 of 122 images (Continued)

TEST_ RSS4	097	098	099	100	101	102	103	104	105	106	107	108	109	110	111	112
minR1	89	132	46	136	127	100	85	78	81	132	102	146	139	136	127	139
minG1	60	113	23	111	100	80	61	51	59	94	66	117	106	104	96	103
minB1	56	99	15	106	89	69	51	34	48	81	52	99	99	79	76	87
maxR1	238	180	181	215	194	226	193	216	220	192	195	193	197	193	190	190
maxG1	207	141	137	180	159	199	163	179	189	134	136	154	167	147	150	139
maxB1	179	110	98	158	131	180	139	152	161	88	80	123	143	98	114	92
minR2	1	8	3	11	3	4	7	6	3	1	2	2	12	1	19	15
minG2	2	3	3	7	3	2	5	2	3	1	2	2	6	5	11	4
minB2	4	10	5	4	1	3	6	3	3	3	2	2	6	8	8	2
maxR2	169	157	147	151	186	211	171	179	197	164	169	190	164	159	170	150
maxG2	151	137	118	132	168	193	143	151	174	134	138	167	142	121	143	123
maxB2	141	130	112	126	148	181	131	129	158	123	118	159	131	108	122	104
meanR1	148	140	142	146	146	148	149	162	145	162	162	156	151	164	159	155
meanG1	102	100	97	104	102	108	103	111	102	113	113	108	102	114	106	104
meanB1	78	85	79	87	75	86	77	71	77	68	67	74	77	64	64	65
meanR2	60	54	75	55	65	76	74	53	47	68	58	83	66	56	65	63
meanG2	46	43	56	44	52	67	58	38	37	51	42	66	53	41	49	46
meanB2	42	41	53	41	48	63	53	31	34	44	35	60	48	33	41	40
Clust1_Area	96	96	94	97	97	97	96	98	95	97	98	96	96	97	97	97
Clust2_Area	4	4	6	3	3	3	4	2	5	3	2	4	4	3	3	3
tLoad	0.05	0.06	0.04	0.05	0.04	0.05	0.04	0.05	0.07	0.04	0.04	0.04	0.04	0.05	0.04	0.05
tCrop	0.38	0.40	0.40	0.41	0.41	0.40	0.39	0.40	0.32	0.32	0.40	0.40	0.39	0.39	0.39	0.40
tKmean	12.00	8.35	11.81	11.40	11.19	10.05	9.30	9.69	9.70	10.18	10.03	9.14	9.19	9.25	9.58	9.88
tSum	12.43	8.80	12.25	11.85	11.64	10.50	9.74	10.14	10.09	10.54	10.47	9.59	9.63	9.68	10.01	10.32
result	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**4**	**3**	**3**	**3**

C1. Type 4 : Ribbed Smoked Sheet (RSS) images testing set data grade 4 of 122 images (Continued)

TEST_ RSS4	113	114	115	116	117	118	119	120	121	122
minR1	127	133	143	126	79	132	145	127	130	120
minG1	98	91	115	90	52	99	112	97	102	97
minB1	84	79	91	78	41	80	97	97	99	83
maxR1	195	193	197	196	190	185	213	228	220	197
maxG1	126	150	132	133	121	128	183	191	179	146
maxB1	95	99	90	79	88	73	155	164	159	117
minR2	45	1	12	3	2	64	11	1	5	4
minG2	28	1	2	1	1	56	7	1	3	2
minB2	21	1	1	2	9	53	4	1	4	3
maxR2	150	169	157	156	175	154	192	153	193	156
maxG2	127	138	134	123	148	123	164	131	165	132
maxB2	111	118	120	106	137	105	142	120	161	130
meanR1	149	162	161	158	157	157	155	144	141	142
meanG1	99	111	111	103	105	108	105	100	100	97
meanB1	72	66	70	67	76	65	70	83	85	78
meanR2	62	72	61	63	61	62	76	58	70	90
meanG2	46	54	45	45	46	46	58	46	55	68
meanB2	42	45	39	39	41	39	52	42	54	66
Clust1_Area	96	97	97	97	96	97	95	97	91	90
Clust2_Area	4	3	3	3	4	3	5	3	9	10
tLoad	0.04	0.04	0.05	0.05	0.04	0.05	0.04	0.04	0.05	0.05
tCrop	0.39	0.39	0.40	0.41	0.40	0.41	0.39	0.40	0.40	0.34
tKmean	9.36	9.49	10.22	10.85	9.08	11.93	8.64	10.03	12.11	11.40
tSum	9.80	9.93	10.66	11.31	9.52	12.38	9.07	10.47	12.57	11.79
result	**4**	**4**	**4**	**4**	**4**	**3**	**3**	**4**	**4**	**4**

C1. Type 5 : Ribbed Smoked Sheet (RSS) images testing set data grade 5 of 90 images

TEST_ RSS5	**001**	**002**	**003**	**004**	**005**	**006**	**007**	**008**	**009**	**010**	**011**	**012**	**013**	**014**	**015**	**016**
minR1	159	105	134	91	147	117	148	63	66	141	149	119	169	140	54	147
minG1	144	86	105	70	134	90	121	48	45	118	133	89	151	117	37	125
minB1	105	54	75	41	100	60	92	19	16	87	97	61	115	83	9	102
maxR1	208	233	200	209	204	209	213	207	207	202	207	202	206	205	225	223
maxG1	157	188	164	153	177	167	150	170	183	160	170	166	178	168	192	202
maxB1	76	120	88	76	108	85	55	90	139	76	100	92	128	88	159	175
minR2	9	1	3	2	27	2	3	43	27	43	36	2	6	8	91	5
minG2	8	1	5	3	18	2	7	26	15	25	24	6	5	8	71	6
minB2	13	1	4	5	1	4	8	10	1	11	8	5	10	6	46	8
maxR2	168	152	166	162	154	174	196	154	173	173	159	165	176	174	170	246
maxG2	155	122	151	137	139	151	173	130	143	150	140	134	152	153	155	236
maxB2	121	94	118	106	108	119	142	102	115	118	108	105	126	124	124	227
meanR1	165	167	166	167	161	165	164	165	164	163	164	166	168	164	164	151
meanG1	125	128	129	125	129	130	125	126	126	123	125	124	127	123	122	115
meanB1	63	61	64	61	67	63	63	63	63	61	59	58	66	63	61	70
meanR2	47	41	41	46	40	29	42	33	36	35	33	45	80	46	50	103
meanG2	36	29	30	33	31	21	32	24	27	27	25	34	63	35	38	89
meanB2	25	20	20	22	22	14	22	16	18	19	17	23	48	26	27	76
Clust1_Area	99	99	99	99	99	99	99	99	99	98	99	99	97	98	98	97
Clust2_Area	1	1	1	1	1	1	1	1	1	2	1	1	3	2	2	3
tLoad	0.04	0.05	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04
tCrop	0.40	0.40	0.39	0.39	0.41	0.40	0.41	0.40	0.40	0.32	0.40	0.39	0.39	0.40	0.40	0.41
tKmean	8.67	8.51	8.45	8.96	10.49	9.75	10.52	8.97	9.38	9.71	10.35	9.03	9.64	10.15	10.70	12.61
tSum	9.11	8.95	8.89	9.39	10.94	10.19	10.98	9.41	9.82	10.07	10.80	9.47	10.08	10.60	11.15	13.07
result	**5**	**5**	**5**	**5**	**5**	**3**	**5**	**5**	**5**	**5**	**3**	**2**	**5**	**5**	**5**	**5**

C1. Type 5 : Ribbed Smoked Sheet (RSS) images testing set data grade 5 of 90 images (Continued)

TEST_ RSS5	017	018	019	020	021	022	023	024	025	026	027	028	029	030	031	032
minR1	127	134	113	135	115	147	132	102	131	156	106	106	129	137	120	141
minG1	103	104	87	114	84	127	100	87	108	127	80	75	102	112	84	112
minB1	77	76	62	83	53	92	77	54	76	97	55	55	75	82	60	82
maxR1	247	215	204	213	214	229	213	228	223	207	227	200	211	211	214	210
maxG1	212	160	173	162	168	174	158	179	180	153	171	156	151	153	169	168
maxB1	174	80	145	99	134	91	101	138	135	81	114	117	81	89	112	130
minR2	9	88	1	9	62	1	2	1	10	39	41	3	54	18	2	58
minG2	9	79	2	4	40	1	4	5	5	23	22	3	37	4	1	38
minB2	9	64	4	8	19	11	3	6	1	7	8	3	21	3	6	14
maxR2	179	174	161	157	164	179	161	192	182	173	195	170	159	151	183	163
maxG2	155	144	135	127	132	163	126	160	163	150	168	144	129	112	163	133
maxB2	131	116	108	103	107	127	104	137	148	118	141	119	105	97	128	107
meanR1	159	171	162	162	169	166	160	166	167	165	162	159	164	165	164	167
meanG1	113	120	111	113	116	116	110	116	114	118	120	114	116	114	117	116
meanB1	66	61	61	61	60	58	57	62	60	59	63	65	58	61	58	60
meanR2	80	38	37	50	42	44	44	44	45	45	42	51	52	53	38	46
meanG2	64	28	26	33	30	29	30	30	32	33	31	37	37	38	26	33
meanB2	51	20	19	26	22	21	22	23	24	24	22	29	27	29	19	25
Clust1_Area	98	98	99	98	99	99	99	98	99	99	99	99	98	98	99	98
Clust2_Area	2	2	1	2	1	1	1	2	1	1	1	1	2	2	1	2
tLoad	0.05	0.04	0.05	0.04	0.05	0.05	0.05	0.04	0.04	0.05	0.05	0.04	0.05	0.04	0.04	0.04
tCrop	0.42	0.40	0.41	0.42	0.40	0.40	0.42	0.40	0.41	0.41	0.41	0.41	0.40	0.40	0.40	0.40
tKmean	12.84	8.19	11.45	11.78	9.58	10.42	11.80	9.93	9.30	10.85	10.90	11.37	11.14	9.51	9.65	10.09
tSum	13.31	8.63	11.90	12.24	10.02	10.87	12.26	10.37	9.75	11.31	11.36	11.83	11.59	9.95	10.10	10.54
result	**5**	**5**	**5**	**5**	**5**	**5**	**3**	**5**	**5**	**5**	**5**	**5**	**5**	**5**	**5**	**5**

C1. Type 5 : Ribbed Smoked Sheet (RSS) images testing set data grade 5 of 90 images (Continued)

TEST_ RSS5	033	034	035	036	037	038	039	040	041	042	043	044	045	046	047	048
minR1	104	136	90	108	113	128	144	140	109	136	76	136	140	135	59	87
minG1	72	111	70	85	83	101	130	121	88	112	51	108	115	108	37	67
minB1	51	81	43	54	59	72	103	91	57	84	29	84	85	87	14	42
maxR1	201	198	202	227	219	202	214	236	234	210	210	219	207	240	228	216
maxG1	152	141	160	172	186	137	177	201	209	164	181	174	174	204	198	178
maxB1	85	62	100	89	153	57	107	179	169	104	139	133	131	156	160	131
minR2	1	63	12	6	2	6	1	1	30	2	34	3	1	3	3	11
minG2	1	41	9	5	2	2	3	1	16	4	14	1	2	4	3	6
minB2	1	20	2	10	2	3	2	1	5	3	3	6	4	8	5	2
maxR2	170	162	149	169	205	155	213	238	229	161	172	173	157	214	231	144
maxG2	149	131	125	142	178	128	196	201	202	139	155	152	131	193	212	119
maxB2	118	103	97	113	151	98	176	183	172	116	125	121	104	166	182	99
meanR1	163	168	165	165	162	164	153	163	166	159	164	161	159	152	160	158
meanG1	118	119	118	116	114	115	118	124	124	110	129	118	119	111	119	119
meanB1	65	58	61	60	59	56	73	73	65	65	71	64	58	68	61	66
meanR2	46	49	34	29	37	34	98	70	44	50	30	54	41	47	58	30
meanG2	33	35	25	21	27	24	85	59	32	36	22	41	30	33	44	21
meanB2	24	25	17	15	20	16	70	51	22	29	15	30	20	27	32	14
Clust1_Area	99	98	99	99	99	99	97	98	99	98	99	98	98	98	98	99
Clust2_Area	1	2	1	1	1	1	3	2	1	2	1	2	2	2	2	1
tLoad	0.05	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.04	0.04	0.04	0.05	0.05	0.04	0.05	0.05
tCrop	0.41	0.40	0.40	0.40	0.41	0.41	0.40	0.40	0.39	0.39	0.39	0.41	0.41	0.42	0.41	0.41
tKmean	10.52	9.54	10.45	9.62	11.49	11.70	10.95	9.99	7.78	8.89	8.43	10.98	10.32	11.56	11.49	11.08
tSum	10.97	9.99	10.90	10.07	11.94	12.15	11.40	10.45	8.21	9.32	8.87	11.44	10.78	12.02	11.95	11.54
result	**5**	**5**	**5**	**5**	**5**	**5**	**4**	**5**	**5**	**3**	**3**	**5**	**3**	**3**	**3**	**5**

C1. Type 5 : Ribbed Smoked Sheet (RSS) images testing set data grade 5 of 90 images (Continued)

TEST_ RSS5	049	050	051	052	053	054	055	056	057	058	059	060	061	062	063	064
minR1	148	94	123	126	61	61	90	48	88	95	107	94	68	77	59	85
minG1	122	70	99	102	39	25	65	39	77	83	89	77	49	64	42	71
minB1	97	46	73	76	16	1	43	22	55	69	69	59	34	48	26	45
maxR1	227	208	200	219	207	208	231	199	196	203	201	217	195	200	200	203
maxG1	195	179	156	168	168	153	197	181	157	171	178	190	161	185	176	163
maxB1	157	139	95	89	93	97	159	145	98	114	134	161	123	164	142	101
minR2	8	5	2	36	25	71	1	3	3	1	25	3	12	5	8	36
minG2	4	6	3	18	15	45	5	1	2	2	17	5	7	5	4	25
minB2	1	1	7	6	6	30	6	2	7	4	14	2	4	5	1	7
maxR2	218	199	178	181	153	157	243	188	167	196	164	195	172	211	173	164
maxG2	192	174	157	165	127	131	214	177	154	180	147	180	155	199	157	146
maxB2	165	143	128	132	104	106	206	171	138	167	129	159	135	187	134	122
meanR1	158	164	162	160	155	159	155	139	149	141	150	149	148	145	151	159
meanG1	116	117	120	120	115	109	116	114	119	117	119	120	118	117	120	122
meanB1	65	64	68	64	66	62	60	87	79	91	78	83	79	83	79	70
meanR2	38	43	56	41	44	50	57	91	70	74	58	71	45	73	31	31
meanG2	26	30	43	30	32	35	44	78	60	62	49	61	36	64	23	24
meanB2	19	22	33	21	24	27	34	72	50	57	41	53	29	56	18	17
Clust1_Area	99	98	98	98	98	98	98	96	98	97	98	97	99	98	99	99
Clust2_Area	1	2	2	2	2	2	2	4	2	3	2	3	1	2	1	1
tLoad	0.05	0.04	0.04	0.04	0.05	0.04	0.05	0.05	0.05	0.05	0.05	0.05	0.04	0.04	0.05	0.04
tCrop	0.41	0.31	0.40	0.41	0.41	0.40	0.42	0.40	0.43	0.42	0.35	0.41	0.42	0.41	0.40	0.30
tKmean	10.12	9.59	10.73	10.10	9.29	9.17	12.57	12.89	13.09	11.31	11.72	10.99	12.05	12.31	9.83	9.29
tSum	10.57	9.95	11.18	10.56	9.74	9.61	13.03	13.33	13.56	11.77	12.12	11.45	12.51	12.77	10.28	9.63
result	**5**	**5**	**3**	**5**	**3**	**5**	**5**	**5**	**5**	**5**	**5**	**4**	**5**	**5**	**5**	**5**

C1. Type 5 : Ribbed Smoked Sheet (RSS) images testing set data grade 5 of 90 images (Continued)

TEST_ RSS5	065	066	067	068	069	070	071	072	073	074	075	076	077	078	079	080
minR1	48	51	90	33	131	76	125	139	60	166	148	124	43	146	143	149
minG1	27	29	64	20	114	63	107	123	48	140	131	104	30	125	124	136
minB1	10	15	49	1	96	46	85	98	32	113	105	80	13	104	109	117
maxR1	189	192	191	191	210	203	214	184	175	215	204	198	207	207	196	192
maxG1	139	157	163	162	177	181	195	150	133	184	182	150	171	165	175	147
maxB1	80	103	124	128	126	144	165	104	83	120	145	78	123	115	144	92
minR2	4	3	3	4	2	29	1	1	37	2	2	53	8	3	2	7
minG2	4	5	3	4	2	23	5	2	25	2	2	46	7	4	2	7
minB2	6	2	5	4	2	9	8	4	9	2	2	38	5	6	2	5
maxR2	167	160	176	149	175	159	159	159	166	205	175	202	182	193	186	166
maxG2	151	147	158	131	156	142	141	141	150	188	153	184	163	171	162	147
maxB2	128	128	134	117	141	124	119	119	134	158	130	160	148	160	152	133
meanR1	153	155	154	146	148	148	153	150	146	160	158	155	153	149	148	144
meanG1	120	123	121	120	120	120	118	119	120	118	126	118	123	117	117	115
meanB1	77	77	74	88	87	84	75	76	86	69	76	75	86	84	87	85
meanR2	27	26	47	35	81	27	34	38	60	92	31	88	82	122	82	66
meanG2	21	20	39	28	66	22	26	30	52	79	25	75	70	100	71	56
meanB2	16	16	30	24	59	18	20	24	46	64	19	62	61	88	63	50
Clust1_Area	99	99	98	98	96	99	98	98	97	97	99	97	98	78	97	98
Clust2_Area	1	1	2	2	4	1	2	2	3	3	1	3	2	22	3	2
tLoad	0.04	0.05	0.04	0.06	0.04	0.05	0.05	0.05	0.05	0.04	0.04	0.05	0.06	0.04	0.04	0.04
tCrop	0.41	0.32	0.40	0.41	0.41	0.41	0.41	0.42	0.37	0.40	0.41	0.41	0.41	0.42	0.41	0.41
tKmean	11.12	10.81	10.39	11.80	10.90	11.59	10.53	11.80	11.84	9.01	9.71	11.84	10.67	14.13	11.82	11.20
tSum	11.57	11.18	10.84	12.27	11.35	12.04	10.98	12.27	12.25	9.45	10.15	12.30	11.14	14.59	12.27	11.65
result	**5**	**5**	**5**	**4**	**5**	**5**	**5**	**5**	**5**	**3**	**4**	**5**	**5**	**5**	**5**	**5**

C1. Type 5 : Ribbed Smoked Sheet (RSS) images testing set data grade 5 of 90 images (Continued)

TEST_ RSS5	081	082	083	084	085	086	087	088	089	090
minR1	83	83	63	152	129	116	105	132	122	137
minG1	68	71	35	127	102	91	75	105	96	113
minB1	45	55	14	97	75	60	51	76	71	79
maxR1	192	213	221	214	206	220	227	212	204	208
maxG1	157	195	176	181	159	183	180	176	160	161
maxB1	101	171	117	140	81	139	112	126	111	89
minR2	7	27	1	2	1	2	2	41	1	3
minG2	7	22	5	1	1	9	2	29	5	1
minB2	9	16	6	6	1	1	2	15	8	4
maxR2	177	212	163	219	161	202	207	206	185	147
maxG2	162	195	137	194	126	173	178	177	159	121
maxB2	143	179	112	164	106	143	146	147	132	98
meanR1	149	145	169	161	163	162	167	166	160	160
meanG1	119	117	130	119	120	115	121	120	118	117
meanB1	86	87	72	60	58	56	59	59	62	57
meanR2	86	87	48	54	47	63	101	108	60	44
meanG2	76	73	36	42	33	49	81	88	47	32
meanB2	66	66	26	31	23	36	60	68	35	21
Clust1_Area	98	96	99	99	99	98	97	96	98	99
Clust2_Area	2	4	1	1	1	2	3	4	2	1
tLoad	0.04	0.05	0.06	0.05	0.04	0.05	0.05	0.04	0.04	0.05
tCrop	0.35	0.41	0.31	0.40	0.41	0.42	0.41	0.41	0.40	0.41
tKmean	11.64	12.88	9.56	11.18	10.60	12.42	10.68	9.99	10.74	10.64
result	**5**	**5**	**3**	**5**	**5**	**5**	**3**	**3**	**3**	**5**

APPENDIX D
MEMORANDUM OF VISUALLY GRADED RUBBER BY EXPERTS

บันทึกข้อความ

ส่วนราชการ สำนักงานตลาดกลางยางพาราหาดใหญ่ โทร. 0 7421-2424-30 โทรสาร 0 7421-2421-2

ที่ **วันที่** 19 กุมภาพันธ์ 2551

เรื่อง ภาพถ่ายยางแผ่นรมควันคุณภาพ 1-5 ผ่านการคัดคุณภาพจากเจ้าหน้าที่คัดคุณภาพ

เรียน อาจารย์ที่ปรึกษาวิทยานิพนธ์

เนื่องด้วย ว่าที่ร.ท.นเรศ จันทรังสิกุล นักศึกษา ระดับปริญญาโท ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยมหิดล ซึ่งทำวิทยานิพนธ์ หัวข้อเรื่อง การคัดเกรดยางแผ่นรมควันด้วยวิธีการประมวลผลภาพ ได้มาเก็บข้อมูลภาพถ่ายยางแผ่นรมควันคุณภาพ 1-5 สำหรับเป็นข้อมูลประกอบการทำวิจัย ซึ่งยางแผ่นรมควันคุณภาพ 1-5 ที่นักศึกษาบันทึกภาพถ่ายไปนั้น ได้ผ่านการคัดคุณภาพจากเจ้าหน้าที่คัดคุณภาพ ของสำนักงานตลาดกลางยางพาราหาดใหญ่เรียบร้อยแล้ว มีรายนามดังต่อไปนี้

1. นายโสภณ จันทรโชติ
2. นายวิสุทธิ์ เพ็ชรมงคล
3. น.ส.กัญญา ตั้งเพ็ง

จึงเรียนมาเพื่อพิจารณา

(นายสุขุม แก้วกลับ)
เจ้าพนักงานการเกษตร 6

BIOGRAPHY

NAME	Acting 1Lt. Naret Chantharangsikul
DATE OF BIRTH	30 August 1981
PLACE OF BIRTH	Bangkok, Thailand
INSTITUTIONS ATTENDED	Saint John's University, 2003 Bachelor of Administrator (Computer Information Management) Mahidol University, 2010 Master of Science (Computer Science)
HOME ADDRESS	6006/108 Prachasongkhro Rd. Din Dang district Bangkok, Thailand
EMPLOYMENT ADDRESS	272 Faculty of Science, Mahidol University Bangkok, Thailand