สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

# การศึกษาความเป็นไปได้ในการคัดแยกถั่วเหลืองฝักสดโดยใช้กระบวนการประมวลผลภาพ

# FEASIBILITY STUDY OF GREEN SOYBEAN SORTING USING IMAGE PROCESSING

โดย

นายณัฐพล แสงจันทร์

นางสาวดาวพระศุกร์ นุ่มเอี่ยม

ปก
ณ 2429
2548

เลขหมู่.....
เลขทะเบียน 62579
วัน,เดือน,ปี 19 ส.ค. 2549

b 11626379
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมเกษตร คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

# การศึกษาความเป็นไปได้ในการคัดแยกถั่วเหลืองฝักสดโดยใช้กระบวนการประมวลผลภาพ

# FEASIBILITY STUDY OF GREEN SOYBEAN SORTING USING IMAGE PROCESSING

นายณัฐพล แสงจันทร์
นางสาวดาวพระศุกร์ นุ่มเอี่ยม

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมเกษตร คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2548

ปริญญานิพนธ์ปีการศึกษา 2548

ภาควิชาวิศวกรรมเกษตร

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การศึกษาความเป็นไปได้ในการคัดแยกถั่วเหลืองฝักสดโดยใช้กระบวนการประมวลผลภาพ

Feasibility Study of Green Soybean Sorting using Image Processing


ผู้จัดทำ

1. นายณัฐพล แสงจันทร์ รหัสประจำตัว 45010241
2. นางสาวดาวพระศุกร์ นุ่มเอี่ยม รหัสประจำตัว 45010267


..... อาจารย์ที่ปรึกษา

(รศ.ดร.ปานมนัส ศิริสมบูรณ์)

..... อาจารย์ที่ปรึกษา

(ผศ.ดร.ธนารัตน์ ชลิดาพงศ์)

..... อาจารย์ที่ปรึกษา

(ดร.สุวณี บุญมั่ง)

# การศึกษาความเป็นไปได้ในการคัดแยกถั่วเหลืองฝักสดโดยใช้กระบวนการประมวลผลภาพ

ณัฐพล แสงจันทร์ 45010241
ดาวพระศุกร์ นุ่มเอี่ยม 45010267
รศ.ดร. ปานมนัส ศิริสมบูรณ์ อาจารย์ที่ปรึกษา
ผศ.ดร. ธนารัตน์ ชลิดาพงศ์ อาจารย์ที่ปรึกษา
ดร. สุวณี บุญมั่ง อาจารย์ที่ปรึกษา
ปีการศึกษา 2548

## บทคัดย่อ

โครงงานนี้มีวัตถุประสงค์เพื่อศึกษาความเป็นไปได้ในการคัดแยกถั่วเหลืองฝักสด โดยใช้กระบวนการประมวลผลภาพ ใช้พัฒนาโปรแกรมเพื่อใช้ในการคัดแยกขนาด สี และรูปร่าง โดยใช้โปรแกรม Matlab เวอร์ชั่น 6 โดยเริ่มจากการแยกวัตถุออกจากพื้นหลัง คำนวณพื้นที่ภาพฉายของถั่วเหลืองฝักสดแล้วนำมาเปรียบเทียบกับพื้นที่ภาพฉายมาตรฐาน ซึ่งฝักที่มีขนาดเล็กกว่ามาตรฐานจะถูกคัดออก จากนั้นใช้แบบจำลองค่าสีมาใช้ในการประมาณค่าสีของถั่วเหลืองฝักสดเพื่อคัดแยกฝักที่มีสีไม่ต้องการออก และขั้นสุดท้ายคือการคัดแยกถั่วเหลืองฝักสดที่มีลักษณะบิดงอ โดยทำการหาค่าความแปรปรวนของมุมของแต่ละจุดบนเส้นขอบของถั่วเหลืองฝักสด ฝักที่บิดงอจะมีค่าความแปรปรวนของมุมสูงกว่าฝักปกติ ในการวิเคราะห์ใช้ภาพถ่ายของถั่วเหลืองฝักสดพันธุ์ AGS 292 ทั้งแบบที่ต้องการและไม่ต้องการ ผลคือ

1) การหาพื้นที่ภาพฉายยังได้ค่าสูงกว่าที่วัดได้โดย Planimeter 9.8 % ซึ่งในส่วนนี้ไม่มีผลต่อการคัดแยกโดยใช้การประมวลผลภาพ
2) คัดผิด โดยคัดฝักที่ต้องการออกไป 8 % และฝักที่ไม่ต้องการยังอยู่ 21.4 %

# FEASIBILITY STUDY OF GREEN SOYBEAN SORTING USING IMAGE PROCESSING

Nattapon Sangjun 45010241

Daoprasuk Numieam 45010267

Assoc.Prof.Dr. Panmanas Sirisomboon Advisor

Assis.Prof .Dr. Thanarat Chalidabhongse Advisor

Dr. Suwanee Boonmung Advisor

2005

**Abstract**

The objective of this project was to test the possibility in sorting by size, color, and shape of green soybeans by image processing. The program for sorting developed using Matlab version 6 was initially conducted for segmentation to separate object from background, finding projected area and comparing with the standard projected area of green soybeans to dismiss green soybeans which have small size. Later, the hue model was used to dismiss green soybeans which had unacceptable color. Last was to sort the twisted pods by calculating the variance of angles of points on the boundaries of green soybean pods. The twisted pods had the variance higher than normal pods. The photographs of the AGS 292 green soybeans were used for evaluation of the program developed. The result showed that

1) The projected area obtained from the method was higher than that obtained from planimeter measurement by 9.8%.
2) Errors in sorting were 8% of the acceptable pods in the unacceptable pods and 21.4% of unacceptable pods remained in acceptable pods.

# กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี ด้วยคำแนะนำและความอนุเคราะห์จาก

รศ.ดร. ปานมนัส ศิริสมบูรณ์

ผศ.ดร. ธนารัตน์ ชลิดาพงศ์

ดร. สุวณี บุญมั่ง อาจารย์ที่ปรึกษาปริญญานิพนธ์ ตลอดจนอาจารย์ทุกท่าน ที่ให้ความเอาใจใส่ แนะนำ และช่วยเหลือเสมอมา และเพื่อนๆทุกคนที่เป็นกำลังใจให้ ซึ่งต้องขอขอบพระคุณเป็นอย่างมาก

และต้องขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้ข้าพเจ้ามีวันนี้ ก็คือ บิดา มารดา อันเป็นที่เคารพรักยิ่ง และยังให้กำลังใจ เอาใจใส่เสมอมา อบรมสั่งสอนแต่สิ่งที่ดีตลอดมา

คณะผู้จัดทำ

นาย ณัฐพล แสงจันทร์

นางสาว ดาวพระศุกร์ นุ่มเอี่ยม

# สารบัญ

หน้าที่


บทคัดย่อภาษาไทย ก

บทคัดย่อภาษาอังกฤษ ข

กิตติกรรมประกาศ ค

สารบัญ ง

สารบัญภาพ ช

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์ของโครงงาน 2

1.3 ขอบเขตของการศึกษา 2

1.4 ผลที่คาดว่าจะได้รับ 2

1.5 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 2

1.6 ขั้นตอนและการดำเนินการ 3

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 4

2.1 การประมวลผลเชิงตัวเลข (Digital Image Processing) 4

2.2 แบบจำลองสี (Color Model) 5

2.2.1 แบบจำลองสี RGB 5

2.2.2 แบบจำลองสี HSI 6

2.3 มอร์โฟโลยีทางคณิตศาสตร์ (Mathematical Mophology) 9

2.3.1 อีรอชั่น (Erosion) 10

2.3.2 ไดเลชั่น (Dilation) 11

2.3.3 โอเพนนิง (Opening) 12

2.3.4 โคลสชิง (Closiing) 12

2.4 กระบวนการแยกส่วน (Segmentation ) 13

2.4.1 การจำแนกภาพ (Image Segmentation) 13

2.5 การวิเคราะห์องค์ประกอบที่เชื่อมต่อกันของภาพ (Connected Components) 16

2.6 การติดตามเส้นขอบ (Contour Following) 18

2.7 ทฤษฎีทางคณิตศาสตร์ 19

2.7.1 การหาระยะทางระหว่างจุดสองจุด 19

2.7.2 การวัดมุม 19

# สารบัญ (ต่อ)

หน้าที่


2.7.3 กฎของโคไซน์ (The law of Cosines) 20

2.7.4 การแจกแจงปกติ (Normal distribution) 20

2.7.4.1 คุณสมบัติการแจกแจงแบบปกติ 21

2.7.4.2. ค่าเฉลี่ยและค่าเบี่ยงมาตรฐานของการแจกแจงปกติ 22

บทที่ 3 วิธีการดำเนินงาน 23

3.1 การเขียนโปรแกรม 24

3.2 การเก็บข้อมูลตัวอย่างภาพถั่วเหลืองฝักสด 24

3.3 ทำการแยกถั่วเหลืองฝักสดออกจากพื้นหลัง (Segmentation) 27

3.3.1 การสร้างแบบจำลองค่า Hue ของถั่วเหลืองฝักสด 27

3.3.2 การหาค่าเฉลี่ยมาตรฐานและค่าเบี่ยงเบนมาตรฐาน 28

3.3.3 การแยกส่วนของถั่วเหลืองฝักสดออกจากพื้นหลัง 29

3.3.4 การกำจัดสัญญาณรบกวน โดยใช้มอร์โฟโลยีทางคณิตศาสตร์ 29

3.4 การหาขนาดของถั่วเหลืองฝักสด 30

3.4.1 Labeling of connected components 30

3.4.2 การหาพื้นที่ภาพฉาย (Projected area) 31

3.5 สีของถั่วเหลืองฝักสด (Color) 32

3.5.1 การคัดแยกสีของถั่วเหลืองฝักสด 32

3.5.2 การหาจุดด่างดำบนฝักถั่วเหลืองฝักสด 33

3.6 รูปร่าง(Shape) 35

3.6.1 การกำหนดเส้นขอบ (Boundary Line) 35

3.6.2 การหามุม 35

บทที่ 4 ผลการดำเนินงาน 37

4.1 ขนาดพื้นที่ภาพฉาย (Projected area) 37

4.2 สี(Color) 45

4.2.1 ค่าสี Hue 45

4.2.2 จุดด่างดำ 47

4.3 รูปร่าง (Shape) 50

บทที่ 5 บทสรุปและวิจารณ์ 54

ภาคผนวก 55

# สารบัญ (ต่อ)

หน้าที่

ก ส่วนของโปรแกรมหลัก bean_sorting 56

ข ส่วนโปรมแกรมย่อย bound_edge_v 66

เอกสารอ้างอิง 69

# สารบัญภาพ

**รูปที่** **หน้าที่**


2.1 โมเดล RGB 5

2.2 โมเดล HIS 6

2.3 ตัวอย่างสตรักเจอร์อิลิเมนต์ขนาด 3x3 พิกเซล 10

2.4 การทำอีรอชั่น $A \ominus B$ 11

2.5 การทำไดเลชั่น $A \oplus B$ 11

2.6 การทำโอเพนนิง $A \circ B$ 12

2.7 การทำโคลสซิง $A \bullet B$ 13

2.8 ฮีสโตแกรมแสดงค่าเธรดโชลด์แบบครอบคลุม 15

2.9 ฮีสโตแกรมแสดงค่าเธรดโชลด์แบบปรับค่า 15

2.10 การเลือกค่า Threshold ที่เหมาะสม 16

2.11 องค์ประกอบของภาพ 17

2.12 การระบุหมายเลขให้กับองค์ประกอบของภาพโดยพิจารณาใน 4 ทิศทาง 17

2.13 วิธีการติดตามเส้นขอบ 18

2.14 ระยะห่างระหว่างจุดสองจุด 19

2.15 ตัวอย่างมุมที่เกิดขึ้น 19

2.16 ตัวอย่างสามเหลี่ยม 20

2.17 การแจกแจงแบบปกติ 21

3.1 ภาพตัวอย่างลักษณะของถั่วเหลืองฝักสดที่นำมาวิเคราะห์ 23

3.2 (a) ภาพถั่วเหลืองฝักสดบนกระดาษตารางขาวดำ 24

(b) ภาพถั่วเหลืองฝักสดบนพื้นกระดาษสีขาว 24

3.3 แผนผังการทำงานของโปรแกรม 25

3.4 ฮีสโตแกรมค่า Hue 28

3.5 ภาพถั่วเหลืองฝักสดที่ crop เอาแต่รูปของถั่วเหลืองฝักสด 28

3.6 ภาพของถั่วเหลืองฝักสดที่ถูกแยกออกจากพื้นหลัง 29

3.7 การกำจัดสัญญาณรบกวนโดยทำการโอเพนนิง 30

3.8 การกำหนดหมายเลขให้กับถั่วเหลืองฝักสด 30

3.9 ตารางสีขาวดำที่นำไปนับจำนวนพิกเซลต่อ 1 $cm^2$ 31

3.10 (a) ถั่วเหลืองฝักสดที่มีหลายสีปะปนกันอยู่ 33

(b) ถั่วเหลืองที่มีสี Hue น้อยกว่า Threshold color ถูกคัดออก 33

## สารบัญภาพ (ต่อ)

**รูปที่** **หน้าที่**

3.11 จุดสีขาวคือจุดด่างดำที่เกิดขึ้นบนฝักถั่วเหลืองฝักสด 34
3.12 จุด Centroid และการวัดระยะห่างจากจุด Centroid กับจุดด่างดำ 34
3.13 การกำหนดเส้นขอบของถั่วเหลืองฝักสดโดยวิธีการติดตามเส้นขอบ 35
3.14 การหามุมบนเส้นขอบของถั่วเหลืองฝักสด 36
4.1 (a) ภาพถ่ายรูปถั่วเหลืองฝักสดที่มีทั้งแบบสมบูรณ์และไม่สมบูรณ์ 37
(b) ถั่วเหลืองฝักสดที่ใช้ขนาดในการคัดแยก 37
4.2 ถั่วเหลืองฝักสดที่ถูกคัดแยกโดยค่าเฉลี่ยสี Hue 47
4.3 ถั่วเหลืองฝักสดที่ถูกคัดแยกโดยจุดด่างดำ 50
4.4 ถั่วเหลืองฝักสดที่ถูกคัดแยกโดยค่าความแปรปรวนของมุมบนเส้นขอบ 52

# บทที่ 1

# บทนำ

## 1.1 ความเป็นมาและความสำคัญของปัญหา

ถั่วเหลืองฝักสดหรือถั่วแระญี่ปุ่น เป็นถั่วเหลืองที่เก็บเกี่ยวในระยะที่ฝักเต่งและฝักยังมีสีเขียวอยู่ เป็นพืชที่มีแหล่งกำเนิดอยู่ในเอเชียตะวันออก เช่น จีน ญี่ปุ่น เกาหลี ปัจจุบันชาวญี่ปุ่นบริโภคถั่วเหลืองฝักสดปีละประมาณ 150,000 ตัน แต่เนื่องจากต้นทุนการผลิตในญี่ปุ่นสูง จึงนำเข้าจากต่างประเทศประมาณร้อยละ 50 ของความต้องการบริโภคภายประเทศ โดยมีประเทศจีน ไต้หวัน ไทย และอินโดนีเซียเป็นผู้ส่งออกที่สำคัญ [1]

แหล่งปลูกถั่วเหลืองฝักสดของไทยจะกระจายอยู่ในจังหวัดต่างๆ ทางภาคเหนือ เช่น กำแพงเพชร เชียงราย เชียงใหม่ น่าน แพร่ พิจิตร ลำปาง พิษณุโลก เพชรบูรณ์ และอุทัยธานี ส่วนใหญ่เป็นการปลูกเพื่อการส่งออกซึ่งต้องใช้พันธุ์เฉพาะตามที่ตลาดกำหนด เช่น AGS 292 หรือ NO.75 [1]

เนื่องจากถั่วเหลืองฝักสดเป็นพืชที่ให้ผลตอบแทนสูง และคนไทยส่วนใหญ่ยังไม่ค่อยรู้จักถั่วเหลืองฝักสด และยังไม่รู้ถึงข้อมูลเกี่ยวกับคุณค่าทางโภชนาการของถั่วเหลืองฝักสดที่มีประโยชน์ต่อร่างกายเป็นอย่างมาก คือ เป็นแหล่งโปรตีนราคาถูกเมื่อเทียบกับโปรตีนจากเนื้อสัตว์มีใยอาหารสูง มีวิตามิน เอ บี และซี มีแร่ธาตุที่ร่างกายต้องการ เช่น เหล็ก แคลเซียม ฟอสฟอรัส มี Isoflavones ซึ่งเป็นสารช่วยลดความเสี่ยงต่อการเป็นโรคหัวใจ โรคมะเร็งเต้านม มะเร็งต่อมลูกหมากและลดอาการวัยทอง [1]

ถั่วเหลืองฝักสดสามารถบริโภคได้ทั้งในรูปแบบอาหารว่าง คือ นำไปต้มทั้งฝัก รับประทานเป็นอาหารคาว คือ การใช้ผัดแทนผักต่างๆ หรือใส่ในแกงไตปลา แกงส้ม สลัด ทำซุป เป็นต้น และอาหารหวาน สามารถทำเป็นวุ้นถั่วเหลืองฝักสด ไอศครีมถั่วเหลืองฝักสด สาคูถั่วเหลืองฝักสด เป็นต้น จะเห็นได้ว่า สามารถนำถั่วเหลืองฝักสดมาทำได้สารพัด [1]

ปัจจุบันเครื่องเขย่าที่เป็นเชิงกลที่ใช้อยู่ยังใช้งานได้ไม่ดีเนื่องจากเครื่องเขย่าที่ใช้อยู่ยังมีปัญหาทางด้านฝักอุดตันอยู่ที่รูตะแกรง และยังใช้แรงงานคนในการคัดแยกถั่วเหลืองฝักสดที่มีรูปร่างไม่ปกติและสีที่ไม่ถูกต้องตามมาตรฐานออก และเทคโนโลยีทางด้านแมทชีนวิชัน (Machine Vision) ได้เข้ามามีบทบาทกับงานทางการเกษตรมากขึ้น ซึ่งได้นำมาใช้กับผลผลิตทางด้านการเกษตร เช่น การตรวจสอบคุณภาพ (Quality Inspection) การจัดประเภท (Classification) การคัดคุณภาพ (Grading) การคัดเลือก (Sorting) เป็นต้น ดังนั้นเพื่อช่วยลดต้นทุน ลดเวลา ลดความผิดพลาดที่เกิดขึ้น [5] โครงงานนี้จึงได้ใช้วิธีการประมวลผลภาพด้วยคอมพิวเตอร์ เข้ามาช่วยในการวัดลักษณะทางกายภาพ

ของถั่วเหลืองฝักสด ซึ่งการประมวลผลภาพทางคอมพิวเตอร์จะช่วยในการคัดแยก ซึ่งอาจจะได้ผลดีกว่าการคัดแยกด้วยเครื่องเขย่า

## 1.2 วัตถุประสงค์ของโครงงาน

*เพื่อพัฒนา Software เพื่อใช้ในการคัดแยกขนาด รูปร่าง และสีของถั่วเหลืองฝักสดเพื่อการส่งออก*

## 1.3 ขอบเขตของการศึกษา

1.3.1 ใช้ถั่วเหลืองฝักสดพันธุ์ AGS 292 เท่านั้น

1.3.2 ศึกษาการคัดแยกขนาด รูปร่าง และสี

## 1.4 ผลที่คาดว่าจะได้รับ

1.4.1 Software ที่สามารถคัดแยกขนาด รูปร่างและสีได้ตามต้องการ

1.4.2 ผลผลิตในการส่งออกมีคุณภาพดีขึ้น ไม่มีการตีกลับ

1.4.3 สามารถนำโครงงานนี้ไปไปพัฒนาต่อเพื่อสร้างระบบอัตโนมัติมาช่วยเกษตรในการคัดขนาดถั่วเหลืองฝักสด แทนการใช้แรงงานคนและอุปกรณ์ทางการเกษตร

1.4.4 อาจเป็นแนวทางที่จะนำ Image Processing มาใช้กับผลิตผลอย่างอื่นด้วย

## 1.5 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ถั่วเหลืองฝักสดเมื่อแบ่งออกเป็น 3 กลุ่มใหญ่ตามการนำไปใช้ประโยชน์[2] จะได้ว่า

กลุ่มที่ 1 ซึ่งเป็นฝักเพื่อการส่งออก ประกอบด้วยฝัก 3 เมล็ด และ ฝัก 2 เมล็ดใหญ่ มีสัดส่วนคือ 15% และ 32% ตามลำดับ

กลุ่มที่ 2 ซึ่งเป็นฝักที่จำหน่ายในประเทศไทยได้แก่ฝัก 2 เมล็ดเล็ก 21% ฝัก 1 เมล็ด 9%

*กลุ่มที่ 3 คือฝักที่ไม่ต้องการ ได้แก่ฝักลีบ 16% ฝักงอ 2% และฝักเสีย (มีตำหนิจากแมลง) 5%*

ผลวิเคราะห์ความแปรปรวนทางสถิติของตัวแปรต่างๆ ซึ่งได้แก่ ความกว้าง ความยาว ความหนา น้ำหนักต่อฝัก น้ำหนักในน้ำ พื้นที่ภาพฉาย ความหนาแน่นปรากฏของฝักถั่วเหลืองฝักสด ความหนาแน่นรวมและความหนาแน่นเนื้อ พบว่า ความยาว น้ำหนักต่อฝัก น้ำหนักในน้ำ พื้นที่ภาพฉาย และความหนาแน่นเนื้อของเมล็ดของถั่วเหลืองฝักสดฝักสมบูรณ์ (ฝัก 3 เมล็ด และฝัก 2 เมล็ดใหญ่) มี

ความแตกต่างกับฝักไม่สมบูรณ์ (ฝัก 2 เมล็ดเล็ก ฝัก 1 เมล็ด ฝักลีบและฝักงอ) ส่วนความกว้างและความหนาของฝักถั่วเหลืองฝักสมบูรณ์ไม่แตกต่างจากฝัก 1 เมล็ด เนื่องจากหนึ่งเมล็ดนั้นมีขนาดที่ใหญ่เท่ากับฝักสมบูรณ์ อย่างไรก็ดีขนาดของฝักที่แตกต่างกันมีผลต่อความแน่นเนื้อ อาจทำให้รสชาติการขบเขี้ยวที่แตกต่างกัน โดยฝัก 2 เมล็ดใหญ่มีความหนาแน่นเนื้อสูง ฝัก 2 เมล็ดเล็ก และฝัก 1 เมล็ดมีสมบัติเดียวที่มีความแตกต่างจากฝักที่ไม่ต้องการคือฝักลีบและฝักบิดงอ คือ ความหนา นอกจากนี้ความกว้าง น้ำหนักต่อฝักและน้ำหนักในน้ำของฝัก 2 เมล็ดเล็กและลีบไม่แตกต่างกัน

การคัดแยกถั่วเหลืองฝักสด ซึ่งจะเห็นได้ว่าการคัดแยกถั่วเหลืองฝักสดตามสมบัติซึ่งแตกต่างระหว่างฝักสมบูรณ์กับฝักไม่สมบูรณ์ ได้แก่ ความยาว น้ำหนักต่อฝัก น้ำหนักในน้ำ พื้นที่ภาพฉายและความแน่นเนื้อ เนื่องจากคุณสมบัติของถั่วเหลืองฝักสด 2 กลุ่มมีความแตกต่างกัน

ในงานวิจัยการคัดแยกผลมะม่วง[3] ในการคัดขนาดพบว่าจำนวนพิกเซลของภาพถ่ายของมะม่วงมีความสัมพันธ์เชิงเส้นตรงกับน้ำหนักของมะม่วงมากที่สุด ซึ่งแสดงให้เห็นว่าทั้งความกว้าง ความยาว และพื้นที่ภาพฉายของมะม่วงมีความสัมพันธ์กับน้ำหนักของมะม่วงทั้งสิ้น แต่ทั้งนี้พื้นที่ภาพฉายจะเป็นตัวแทนของน้ำหนักมะม่วงได้ดีที่สุด เพราะมะม่วงมีความกว้างและความยาวมากย่อมมีพื้นที่ภาพฉายมากด้วย

**1.6 ขั้นตอนและการดำเนินงาน**

1. หาข้อมูลเบื้องต้น เช่น สมบัติทางกายภาพ ขนาด รูปร่าง และสีที่ถูกต้องตามมาตรฐานที่ได้กำหนดไว้
2. ศึกษาโปรแกรม Matlab เวอร์ชั่น 6
3. เก็บภาพตัวอย่างที่นำมาใช้การศึกษา
4. ศึกษาแบบจำลองสีต่างๆ และการเปลี่ยนค่าแบบจำลองสี
5. เขียนโปรแกรมในการดึงภาพถั่วเหลืองฝักสดออกจากพื้นหลัง
6. เขียนโปรแกรมในการวิเคราะห์ลักษณะทางกายภาพของถั่วเหลืองฝักสด เช่น ขนาด สี และรูปร่าง
7. ทำการทดลอง
8. ปรับปรุงแก้ไขโปรแกรมให้สมบูรณ์ยิ่งขึ้น
9. สรุปผล
10. รวบรวมข้อมูลจัดทำเอกสารนำเสนอเป็นโครงงาน

# บทที่ 2

# ทฤษฎีที่เกี่ยวข้อง

## 2.1 การประมวลผลภาพเชิงตัวเลข (Digital Image Processing)

การประมวลผลภาพเชิงตัวเลข คือ กระบวนการจัดการทางรูปภาพที่ Input ที่ป้อนเข้าสู่กระบวนการ และ Output ที่ออกจากกระบวนการต่างก็เป็นรูปภาพเช่นเดียวกัน เป็นกระบวนการดึงคุณสมบัติของภาพออกมาพิจารณา อีกทั้งยังรวมไปถึงวิธีการรวมเอาสิ่งที่สนใจแต่ละอย่างมาพิจารณาโดยคอมพิวเตอร์จะใช้เลขฐาน 2 เป็นหลักในการคำนวณ โดยภาพจะถูกแทนที่ด้วยตัวเลขให้อยู่ในรูปของเมตริกซ์ ดังนั้นเมื่อนำภาพมาแปลงเข้าสู่คอมพิวเตอร์ภาพนั้นจะกลายเป็นภาพดิจิตอล (Digital Image) ซึ่งจะเรียกการทำงานนี้ว่าเป็นการประมวลผลเชิงตัวเลข [4]

ระบบพิกัดระนาบ 2 มิติ (Spatial Coordinate) จะใช้ในการแสดงภาพดิจิตอล ซึ่งมีขนาดความกว้างและความสูงของภาพแสดงในแกน X และ Y ส่วนจุดใดๆ ที่อยู่บนระนาบ XY จะเรียกแทนจุดใดๆ นั้นว่าพิกเซล (Pixel) โดยจะมีฟังค์ชัน f(x,y) ที่แสดงถึงระดับความเข้มที่มีอยู่ L ระดับ [6]

จำนวนช่วงระดับความเข้ม L ของจุดภาพจะบ่งบอกถึงระดับความเข้มของภาพเชิงตัวเลข ซึ่งโดยทั่วไประดับความเข้ม L จะมีค่าตั้งแต่ 2 ระดับขึ้นไป แต่จะนิยมใช้กันมากที่ค่าระดับความเข้ม 256 ระดับ นั่นคือค่าความเข้มของจุดภาพอยู่ในช่วง[0-255] โดยจะใช้เนื้อที่ในการเก็บขนาด 1 ไบต์ (Byte) หรือ 8 บิต (Bit) สำหรับข้อมูล 1 จุดภาพ ($2^8 = 256$) [5]

รูปแบบในการประมวลผลภาพสามารถแบ่งออกได้ 2 ระดับ คือ การประมวลผลภาพในระดับต่ำ (Low-level Image Processing) และการประมวลผลภาพในระดับสูง (High-level Image Processing) มีรายละเอียดดังต่อไปนี้ [5]

การประมวลผลภาพในระดับต่ำ เป็นการประมวลผลภาพเชิงตัวเลขเพื่อหาตัวแปรต่างๆ มาอธิบายข้อมูลภาพ โดยทั่วไปแล้วการประมวลผลภาพในระดับต่ำจะหมายถึง การประมวลผลภาพก่อน (Image Preprocessing) เช่น การกำจัดสัญญาณรบกวน การจำแนกภาพ การหาขอบภาพ การเปลี่ยนแปลงภาพ การทำให้ภาพคมชัด เป็นต้น โดยการประมวลผลในระดับต่ำจะใช้ค่าความสว่างหรือระดับความเข้มของจุดภาพโดยตรง ซึ่งถือว่าเป็นกระบวนการเบื้องต้นที่มีความสำคัญมาก เพื่อจะนำตัวแปรที่ได้ไปใช้ในการประมวลผลภาพในระดับสูงต่อไป [5]

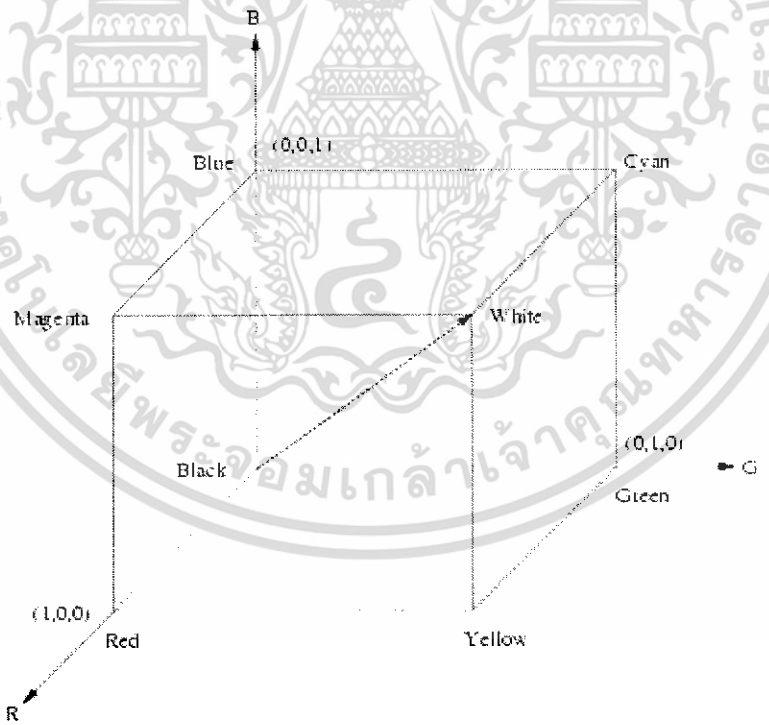
การประมวลภาพในระดับสูง เป็นการนำผลลัพธ์หรือตัวแปรที่ได้จากการประมวลผลภาพในระดับต่ำมาตีความหรือประมวลผลและแสดงให้อยู่ในรูปของสัญลักษณ์ โดยสัญลักษณ์เหล่านี้จะแสดงถึงสิ่งต่างๆ ในภาพเพื่อทำให้คอมพิวเตอร์สามารถเข้าใจภาพได้ เช่น การรู้จำรูปแบบของตัวอักษร ความสัมพันธ์ระหว่างวัตถุในภาพ ขนาดหรือรูปร่างของวัตถุ เป็นต้น [5]

## 2.2 แบบจำลองสี (Color Model)

ภาพดิจิตอลจะแทนด้วยเมตริกซ์ใน 2 มิติ โดยแต่ละหน่วยในเมตริกซ์จะเรียกว่าพิกเซล(Pixel) กระบวนการแสดงภาพให้ปรากฏขึ้นมานั่นคือการแสดงค่าของเมตริกซ์ โดยแต่ละพิกเซลจะมีค่าเฉพาะตัวแสดงถึงสีต่างๆ โดยปกติภาพขาว-ดำ จะมีค่าของพิกเซลอยู่ระหว่าง 0 ถึง 255 ส่วนภาพสีนั้นแต่ละพิกเซลจะประกอบไปด้วยสีแดง (R) สีเขียว (G) สีน้ำเงิน (B) ซึ่งแต่ละสีจะเป็นขนาด 8 บิต ดังนั้นในความลึกของแต่ละพิกเซลจะมีขนาดเป็น 24 บิต โดยแต่ละสีจะมีค่าของพิกเซลอยู่ระหว่าง 0 ถึง 255 เช่นกัน ในงานวิจัยนี้ได้ศึกษาแบบจำลองสีต่างๆ ดังนี้ [7]

### 2.2.1 ระบบ RGB (Red/Green/Blue) [7]

หากจะให้เข้าใจได้ชัดเจนมากขึ้น สามารถพิจารณาได้จากสี่เหลี่ยมที่มีพิกัดแทนด้วย(R,G,B) สีดำจะมีพิกัด (0,0,0) สีขาวจะมีพิกัด (255,255,255) สีแดงมีพิกัด (255,0,0) สีเขียวมีพิกัด (0,255,0) สีน้ำเงินมีพิกัด (0,0,255) สีเหลืองมีพิกัด (255,255,0) สีฟ้าคราม (0,255,255) สีม่วงมีพิกัด(255,0,255) ซึ่งสีทั้งแปดนี้จะอยู่มุมกล่องสีเหลี่ยมลูกบาศก์ ส่วนค่าสีที่อยู่ในกล่องจะประกอบไปด้วยค่า RGB ที่มีอัตราส่วนที่แตกต่างกันออกไปและแสดงด้วยสีเฉพาะตัวสำหรับแต่ละอัตราส่วนด้วย ดังรูปที่ 2.1

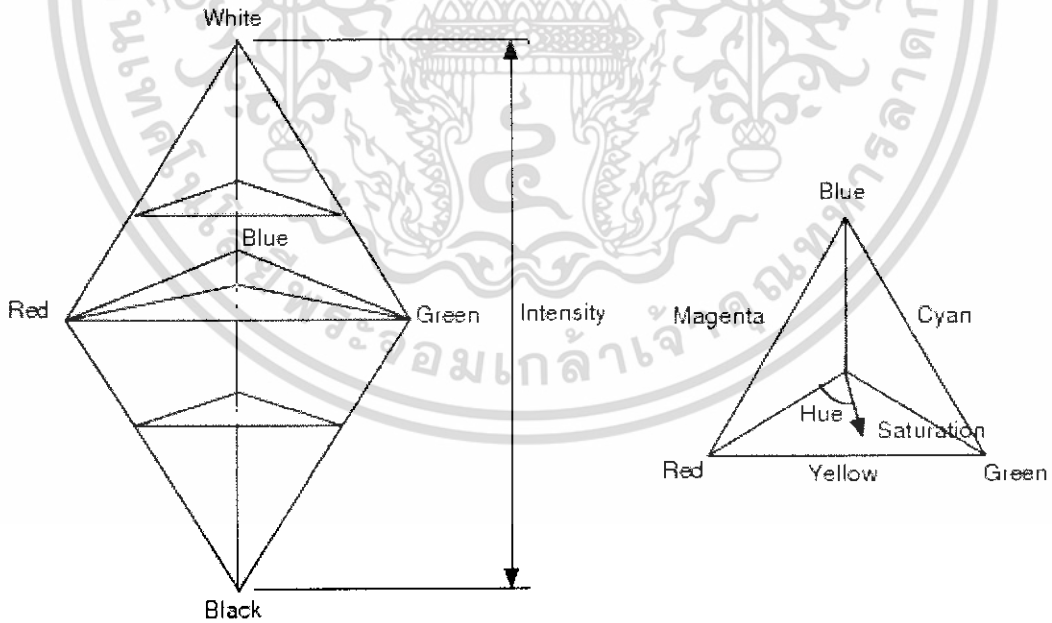



รูปที่ 2.1 โมเดล RGB จากจุดสีดำ (0,0,0) ถึง จุดสีขาว (1,1,1) คือ Gray Scale [9]

**2.2.2 ระบบ HSI (Hue/Saturation/Intensity)**

ระบบ HSI เป็นระบบที่สร้างมาจากสัญชาตญาณการนึกคิดของมนุษย์ในการกำหนดค่าสี โดยที่ค่า Hue (สี) จะเป็นค่าตัวแทนสีตั้งแต่ 0 (แดง) ถึง 120 (เขียว) ถึง 240 (น้ำเงิน) และถึง 360 (แดงอีกครั้ง) จะเห็นได้ว่ามีลักษณะการแทนค่าคล้ายวงล้อ ส่วนค่า Saturation (ค่าความเข้มของสี) ในความเข้มระดับต่ำตั้งแต่ 0-20% จะให้ผลลัพธ์สีออกมาเป็นเทา, ความเข้มระดับกลาง 40-60% ผลลัพธ์จะเป็นสีอ่อน และ ความเข้มสูง 80-100% ผลลัพธ์จะได้สีที่จัด หรือแจ่มใสนั้นเอง สำหรับค่าสุดท้ายที่ใช้คือ Intensity (ค่าความสว่างของสี) ซึ่งมีช่วงตั้งแต่ 0% (มืด หรือดำ) จนถึง 100% (สว่าง หรือขาว) นั้นเอง โดยปกติในซอฟท์แวร์ หรือเครื่องมือช่วยในการสร้างสีนั้นมักจะมีการใช้ทั้งระบบ RGB และ HSI ร่วมกัน [6]

HSI สามารถที่จะทำความเข้าใจได้โดยใช้ HSI model ค่า Hue (H) ของจุดสี P ก็คือมุมของเวคเตอร์ที่เทียบกับแกนสีแดง ดังนั้นสีแดงค่า Hue จะเป็น 0 สีเหลืองค่า Hue จะมีค่าเป็น 60 ตามลำดับ ส่วนค่า Saturation (S) ของจุด P คืออัตราส่วนของระยะห่างของจุด P ไปยังจุดศูนย์กลางของสามเหลี่ยมกับระยะจาก จุดศูนย์กลางไปยังจุดปลายของสามเหลี่ยมที่ตัดผ่านจุด P ดังนั้นยิ่งจุด P อยู่ห่างจากจุดศูนย์กลางเท่าใด ค่า Saturation ก็จะมากขึ้นเท่านั้น ในภาพเส้นตรงที่ตั้งฉากกับระนาบสามเหลี่ยมและผ่านจุดศูนย์กลางของสามเหลี่ยม นั้นจะแสดงค่า Intensity ซึ่งก็เป็นเส้นตรงเดียวกับที่ลากจากจุด สีดำพิกัด (0, 0, 0) มายังจุดสีขาวพิกัด (255, 255, 255) นั่นเอง [8]




รูปที่ 2.2 โมเดล HIS [9]

**การแปลงภาพ RGB model ให้เป็น HSI model** [10]

ด้วยความสัมพันธ์ระหว่าง RGB model และ HSI model สามารถที่จะคำนวณค่า HSI ในรูปของฟังค์ชันตัวแปร RGB ได้ดังสมการต่อไปนี้

กำหนดให้

$H = \theta$ ถ้า $B \leq G$

$H = 360 - \theta$ ถ้า $B > G$

$$\theta = \cos^{-1}\left\{\frac{\frac{1}{2}\left[(R-G)+(R-B)\right]}{\left[(R-G)^2+(R-B)(G-B)^{\frac{1}{2}}\right]}\right\} \quad (1)$$

$$S = 1-\frac{3}{(R+B+G)}\left[\min(R,G,B)\right] \quad (2)$$

$$I = \frac{1}{3}(R+B+G) \quad (3)$$

**การแปลงภาพ HSI model ให้เป็น RGB model**

ถ้าอยู่ในช่วง RG ($0^0 \leq H < 120^0$) ค่า R ,G และ B สามารถคำนวณได้จาก

$$B = I(1-S) \quad (4)$$

$$R = I\left[1+\frac{S\cos H}{\cos(60^0-H)}\right] \quad (5)$$

$$G = 3I-(R+B) \quad (6)$$

สำหรับในช่วง GB ($120^0 \leq H < 240^0$) ก็สามารถหาค่า R, G และ B ได้คล้ายๆ กับสมการข้างต้น ดังนี้

$$H = H - 120^0 \tag{7}$$

$$R = I\left(1 - S\right) \tag{8}$$

$$G = I\left[1 + \frac{S\cos H}{\cos\left(60^0 - H\right)}\right] \tag{9}$$

$$B = 3I - \left(R + G\right) \tag{10}$$

และสำหรับช่วงสุดท้าย ช่วง BR ($240^0 < \mathrm{H} \leq 360^0$ )

$$H = H - 240^0 \tag{11}$$

$$G = I\left(1 - S\right) \tag{12}$$

$$B = I\left[1 + \frac{S\cos H}{\cos\left(60^0 - H\right)}\right] \tag{13}$$

$$R = 3I - \left(G + B\right) \tag{14}$$

## 2.3 มอร์โฟโลยีทางคณิตศาสตร์ (Mathematical Morphology) [5]

ภาพของวัตถุที่ได้จากการสแกนด้วยเครื่องสแกน (Scanner) หรือได้จากการถ่ายภาพของกล้องดิจิตอล (Digital Camera) เพื่อนำมาประมวลผลภาพด้วยคอมพิวเตอร์นั้น ภาพที่ได้อาจจะมีสัญญาณรบกวนเกิดขึ้น ซึ่งสัญญาณรบกวนที่เกิดขึ้นอาจเกิดได้จากหลายสาเหตุ เช่น ความไม่สม่ำเสมอของแสง เป็นต้น ซึ่งจะทำให้คุณภาพของภาพนั้นลดลงไป ดังนั้นก่อนที่จะประมวลผลภาพต่อไปจึงต้องมีการวิเคราะห์เพื่อกำจัดสัญญาณรบกวนที่เกิดขึ้นนั้นออกไป

มอร์โฟโลยีทางคณิตศาสตร์เป็นวิธีที่นำหลักการทางคณิตศาสตร์ คือทฤษฎีเซต มาใช้ในการนำเสนอรูปแบบหรือโครงสร้างของวัตถุในภาพ ซึ่งสามารถเป็นวิธีที่จะนำมาใช้ในการกำจัดสัญญาณรบกวนแบบเกลือและพริกไทย (Salt-and-pepper Noise) ที่เกิดขึ้นบนภาพไบนารีได้ โดยมีบทนิยามเบื้องต้นดังนี้

กำหนดให้ A และ B เป็นเซตของจุดภาพในเชิง 2 มิติ ซึ่งประกอบด้วยจุดภาพ $a = (a_1,a_2)$ และ $b = (b_1,b_2)$ ตามลำดับ

1. ทรานสเลชั่น (Translation) การทำทรานสเลชั่นของ A โดย $x = (x_1,x_2)$ เขียนแทนด้วย $(A)_x$ เป็นการย้ายตำแหน่งจุดภาพที่อยู่ในเซต A ด้วยขนาด x มีนิยามดังนี้

$$(A)_x = \{c \mid c = a + x; a \in A\} \tag{15}$$

2. รีเฟล็กชั่น (Reflection) การทำรีเฟล็กชั่นของ A เขียนแทนด้วย $\hat{A}$ เป็นการกลับตำแหน่งจุดภาพที่ไม่ได้อยู่ในเซต A มีนิยามดังนี้

$$\hat{A} = \{x \mid x = -a; a \in A\} \tag{16}$$

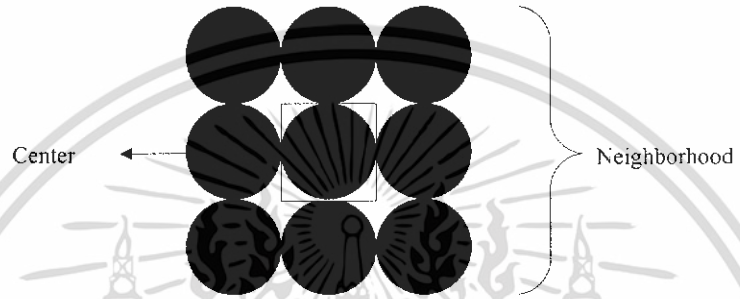
3. คอมพลีเมนต์ (Complement) การทำคอมพลีเมนต์ของเซต A เขียนแทนด้วย $A^c$ เป็นการกำหนดตำแหน่งจุดภาพที่ไม่ได้อยู่ในเซต A มีนิยามดังนี้

$$A^c = \{x \mid x \notin A\} \tag{17}$$

4. ผลต่าง (Difference) การทำผลต่างของเซต A และเซต B เขียนแทนด้วย A-B เป็นการกำจัดจุดภาพบางส่วนในเซต A ซึ่งเป็นภาพที่อยู่ในเซต B มีนิยามดังนี้

$$A - B = \{x \mid x \in A; x \notin B\} = A \cap B^c \tag{18}$$

หลักการทำงานของมอร์โฟโลยีทางคณิตศาสตร์จะพิจารณาเฉพาะกลุ่มที่เป็นวัตถุในภาพตามที่ได้กำหนดไว้ นั่นคืออาจจะพิจารณาเฉพาะกลุ่มจุดภาพดำซึ่งอยู่บนพื้นหลังสีขาว หรืออาจจะพิจารณาเฉพาะกลุ่มจุดภาพขาวซึ่งอยู่บนพื้นหลังสีดำ ในแต่ละกลุ่มจุดภาพอาจจะมีสัญญาณรบกวน (Noise) ซึ่งเป็นจุดภาพไม่พึงประสงค์เกิดขึ้น จึงควรจะมีการกำจัดสัญญาณรบกวนที่เกิดขึ้นนี้ออกไปก่อนที่จะนำภาพไปประมวลผลต่อไป ในการพิจารณาการกำจัดสัญญาณรบกวนนั้นจะต้องมีการกำหนดสตรักเจอร์อิลิเมนต์ (Structure Element) ขึ้นมาช่วยในการพิจารณา กำหนดให้เป็นสตรักเจอร์อิลิเมนต์ B มีตัวอย่างดังต่อไปนี้




รูปที่ 2.3 ตัวอย่างสตรักเจอร์อิลิเมนต์ขนาด 3x3 พิกเซล

จากตัวอย่างสตรักเจอร์อิลิเมนต์ B ดังแสดง เป็นโครงสร้างจุดภาพดำที่มีขนาด 3x3 พิกเซล โดยจะนำจุดตรงกลาง (Center) และจุดรอบข้าง (Neighborhood) ของสตรักเจอร์อิลิเมนต์ B มาพิจารณาร่วมกับวัตถุที่เป็นเซตของจุดภาพดำ

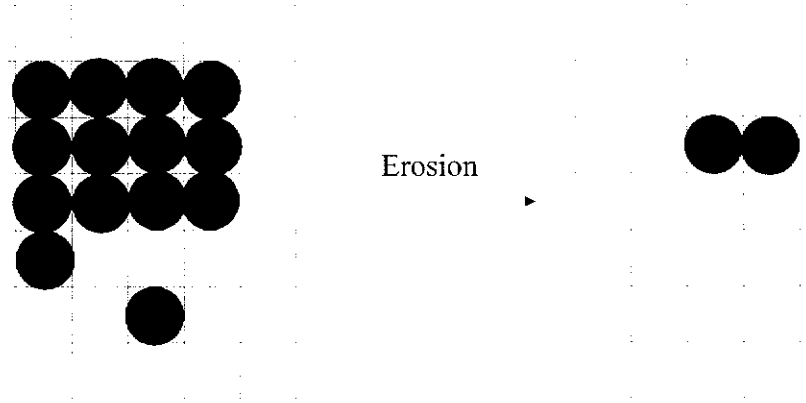
การทำงานโดยพื้นฐานของมอร์โฟโลยีจะประกอบด้วย อีรอชั่น (Erosion) และไดเลชั่น (Dilation) ซึ่งมีการทำงานดังนี้

**2.3.1 อีรอชั่น (Erosion)**

การทำอีรอชั่นของเซต A กับสตรักเจอร์อิลิเมนต์ B เขียนแทนด้วย $A\Theta B$ มีนิยามดังนี้

$$A\Theta B = \left\{ x \mid (B)_x \subseteq A \right\} \tag{19}$$

เป็นการลดจำนวนจุดภาพดำในเซต A ซึ่งจะเพิ่มช่วงห่างให้กว้างขึ้นระหว่างแต่ละกลุ่มของจุดภาพดำที่อยู่ตำแหน่งใกล้กัน หรือเป็นการกำจัดจุดภาพดำที่มีขนาดเล็กมากออกไป ดังตัวอย่างต่อไปนี้




รูปที่ 2.4 การทำอีรอชั่น $A \ominus B$

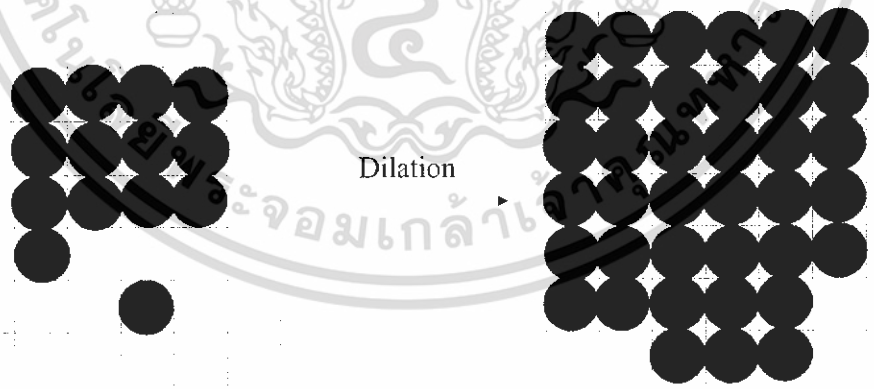
**2.3.2 ไดเลชั่น (Dilation)**

การทำไดเลชั่นของเซต A กับสตรักเจอร์อิลิเมนต์ B เขียนแทนด้วย $A \oplus B$ มีนิยามดังนี้

$$A \oplus B = \left\{ x \middle| \left( \hat{B} \right)_x \cap A \neq \varnothing \right\} \tag{20}$$

การทำไดเลชั่นเป็นการทำงานที่ตรงข้ามกับอีรอชั่นคือ เป็นการเพิ่มจำนวนจุดภาพดำเพื่อขยายขนาดกลุ่มของจุดภาพดำในเซต A ทำให้เป็นการลดช่วงห่างให้แคบลง หรืออาจเกิดการเชื่อมต่อกันระหว่างแต่ละกลุ่มจุดภาพดำที่อยู่ตำแหน่งใกล้กัน ดังตัวอย่างต่อไปนี้

ในการเพิ่มประสิทธิภาพของมอร์โฟโลยีทางคณิตศาสตร์สามารถรวมการทำงานของทั้ง 2 พื้นฐานเข้าด้วยกัน ทำให้เกิดวิธีการ โอเพนนิง (Opening) และ โคลสซิง (Closing) ซึ่งมีการทำงานดังนี้



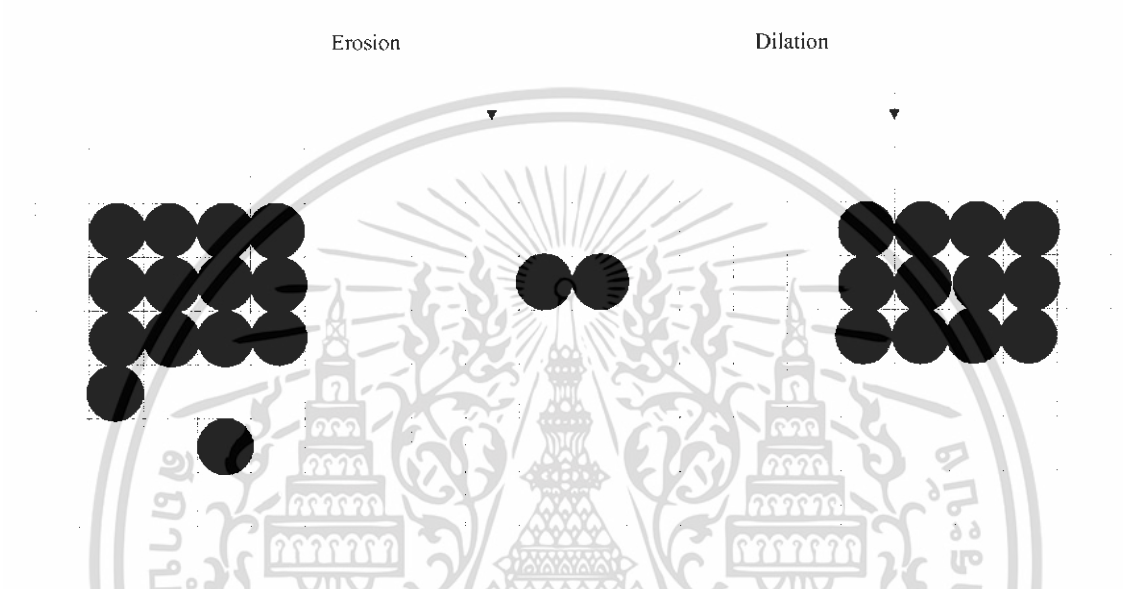

รูปที่ 2.5 การทำไดเลชั่น $A \oplus B$

### 2.3.3 โอเพนนิง (Opening)

การทำโอเพนนิงของเซต A กับสตรักเจอร์อิลิเมนต์ B เขียนแทนด้วย $A \circ B$ มีนิยามดังนี้

$$A \circ B = \{A \Theta B\} \oplus B \tag{21}$$

เป็นการทำงานร่วมกันระหว่างการทำอีรอชั่นและไดเลชั่นตามลำดับ คือจะเริ่มจากการทำอีรอชั่นโดยลดจำนวนจุดภาพดำ แล้วตามด้วยการทำไดเลชั่นโดยเพิ่มจำนวนจุดภาพดำลงไปในภาพให้เท่ากับจำนวนของการทำอีรอชั่น ดังตัวอย่างต่อไปนี้



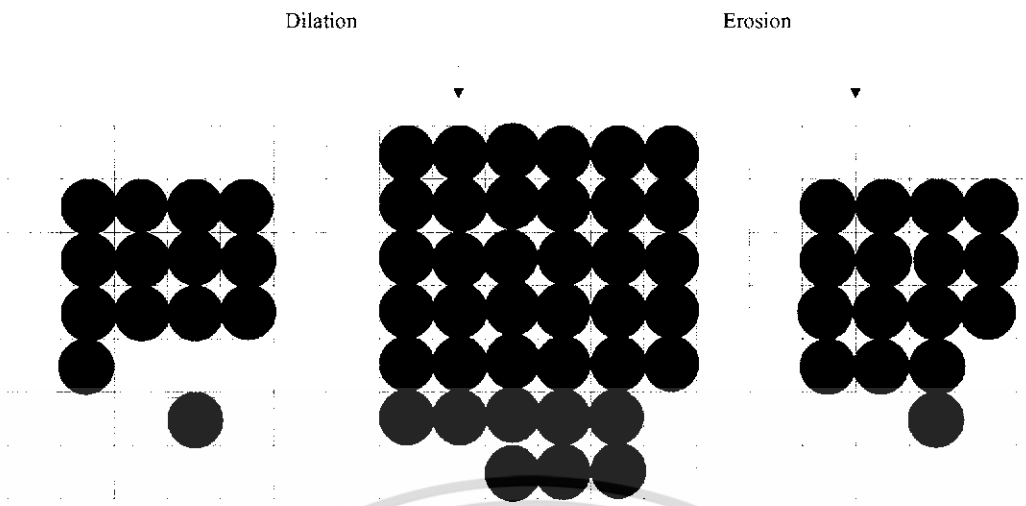

รูปที่ 2.6 การทำโอเพนนิง $A \circ B$

### 2.3.4 โคลสซิง (Closing)

การทำโคลสซิงของเซต A กับสตรักเจอร์อิลิเมนต์ B เขียนแทนด้วย $A \bullet B$ มีนิยามดังนี้

$$A \bullet B = \{A \oplus B\} \Theta B \tag{22}$$

เป็นการทำงานที่มีลำดับตรงข้ามกับการทำโอเพนนิง โดยจะทำไดเลชั่นและอีรอชั่นตามลำดับ คือจะเริ่มจากการทำไดเลชั่นโดยเพิ่มจำนวนจุดภาพดำ แล้วตามด้วยการทำอีรอชั่นโดยลดจำนวนจุดภาพดำของภาพให้เท่ากับจำนวนรอบของการทำไดเลชั่น ดังตัวอย่างต่อไปนี้




รูปที่ 2.7 การทำโคลสซิง $A \bullet B$

จากการทำงานข้างต้น เป็นการทำงานที่ยังไม่ครอบคลุมพอในการกำจัดสัญญาณรบกวนแบบเกลือและพริกไทย จึงมีการพัฒนาผลลัพธ์ที่ได้จากการทำโอเพนนิงและโคลสซิง โดยนำทั้งสองการทำงานนี้มาทำงานร่วมกันคือ อาจจะทำโอเพนนิงก่อนแล้วตามด้วยโคลสซิง หรือในทางตรงกันข้ามอาจจะทำโคลสซิงก่อนแล้วตามด้วยโอเพนนิง ทำให้ผลลัพธ์ที่ได้มีประสิทธิภาพมากขึ้น

## 2.4 กระบวนการแยกส่วน (Segmentation ) [4]

กระบวนแบ่งส่วนเป็นกระบวนการที่แบ่งภาพออกเป็นส่วนประกอบย่อยๆ ซึ่งกระบวนการนี้เป็นกระบวนการสำคัญกระบวนการหนึ่งที่ใช้ในการวิเคราะห์ภาพของระบบเครื่องจักร เพราะกระบวนการนี้จะดึงสิ่งที่น่าสนใจจากภาพ เพื่อใช้ในกระบวนการอื่น ๆ ภายหลังวิธีการแบ่งส่วนโดยทั่วไปแล้วตั้งอยู่บนคุณสมบัติของค่า gray-level 2 ประการ นั่นคือ ความไม่ต่อเนื่อง และ ความเหมือน ในกรณีแรกเราจะแบ่งรูปภาพโดยมีการเปลี่ยนแปลงของ gray-level อย่างฉับพลัน ซึ่งขอบเขตของสิ่งที่สนใจภายใต้กรณีนี้ คือ การค้นหาจุดที่แยกออกจากกัน และการค้นหาเส้นและขอบของรูป หลักการที่ใช้ในกรณีที่สองตั้งอยู่บนค่าตัวตัด (Thresholds), region growing, region splitting และ merging แนวคิดของกระบวนการแบ่งส่วนรูปภาพอยู่บนพื้นฐานของความไม่ต่อเนื่อง หรือ ความเหมือนของค่า gray-level ของจุด (pixel) ซึ่งเหมาะสมทั้งกับรูปที่เคลื่อนไหว และไม่เคลื่อนไหว อย่างไรก็ดี การเคลื่อนไหวก็เป็นอีกแนวทางหนึ่งที่จะใช้ในการปรับปรุงวิธีการของการแบ่งส่วน

### 2.4.1 การจำแนกภาพ (Image Segmentation) [5]

ภาพถ่ายโดยทั่วไปจะประกอบด้วยส่วนวัตถุ (Foreground) ซึ่งเป็นส่วนหน้าที่สนใจและส่วนพื้นหลัง (Background) ซึ่งเป็นฉากของภาพ ในการประมวลผลวัตถุของภาพจะต้องมีการจำแนก

ภาพเพื่อแยกวัตถุออกจากพื้นหลังของภาพ การจำแนกภาพจึงหมายถึงการแยกองค์ประกอบของภาพออกเป็นส่วนย่อย ๆ ในการจำแนกภาพสามารถพิจารณาองค์ประกอบต่าง ๆ ซึ่งมีด้วยกันหลายวิธี เช่น ขอบรอยต่อของภาพ (Edge) ลักษณะพื้นผิว (Texture) สี (Color) หรือรูปทรง (shape) เป็นต้น ซึ่งองค์ประกอบย่อยเดียวกันจะมีคุณสมบัติพื้นฐานที่เหมือนกันหรือใกล้เคียงกัน

ในการจำแนกภาพด้วยความคล้ายคลึงกันของค่าสี เป็นวิธีหนึ่งที่ง่ายในการนำมาพิจารณา นั่นคือ วัตถุเดียวกันจะมีช่วงค่าสีที่ใกล้เคียงกัน และจะมีช่วงค่าสีที่แตกต่างจากส่วนอื่น ๆ ในการพิจารณาช่วงค่าสีของแต่ละส่วนในภาพสามารถใช้ฮีสโตแกรมของสี (Color Histogram) มาช่วยในการพิจารณาภาพได้ โดยฮีสโตแกรมจะแสดงออกมาเป็นกราฟแท่งเพื่อแสดงจำนวนจุดภาพของแต่ละค่าสีในภาพซึ่งจะไม่ได้สื่อถึงตำแหน่งในภาพ โดยทั่วไปแล้วภาพจะมีฮีสโตแกรมที่แน่นอน แม้ว่าภาพนั้นจะมีการเคลื่อนย้าย (Translate) หรือมีการหมุนของภาพ (Rotation) ฮีสโตแกรมก็จะไม่เปลี่ยนแปลง และในภาพที่ต่างกันอาจจะมีฮีสโตแกรมที่คล้ายกันได้

ในการจำแนกส่วนประกอบต่าง ๆ ของภาพด้วยฮีสโตแกรมของค่าสี จะต้องมีจุดที่ใช้ในการแบ่งแยกช่วงค่าสีที่แตกต่างกันออกมาเป็นองค์ประกอบต่าง ๆ เรียกวิธีนี้ว่าการทำ เธรดโชลด์ (Thresholding Technique) นั่นคือจะใช้ค่าเธรดโชลด์ในการเปรียบเทียบเพื่อใช้เป็นการแบ่งแยกช่วงกลุ่มค่าสีที่แตกต่างกัน

Threshold เป็นสิ่งสำคัญอย่างหนึ่งในการแบ่งส่วนรูปภาพใน section นี้ เราจะพัฒนาเทคนิคสำหรับ Threshold และจะอธิบายถึงข้อดีและข้อจำกัดของวิธีการเหล่านี้

ข้อจำกัดของการใช้ Histogram สำหรับ Thresholding

- Histogram ใช้เมื่อรูปมีค่า gray value คงที่ ไม่สามารถใช้ได้กับรูปที่เป็นเงาหรือซับซ้อน
- ข้อจำกัดที่ตั้งอยู่บนความจริงที่ว่า Histogram จะอธิบายถึงการแจกแจงความเข้มอย่างง่าย และไม่พิจารณาถึงข้อมูลเชิงพื้นที่เกี่ยวกับค่าความเข้มของรูปภาพ

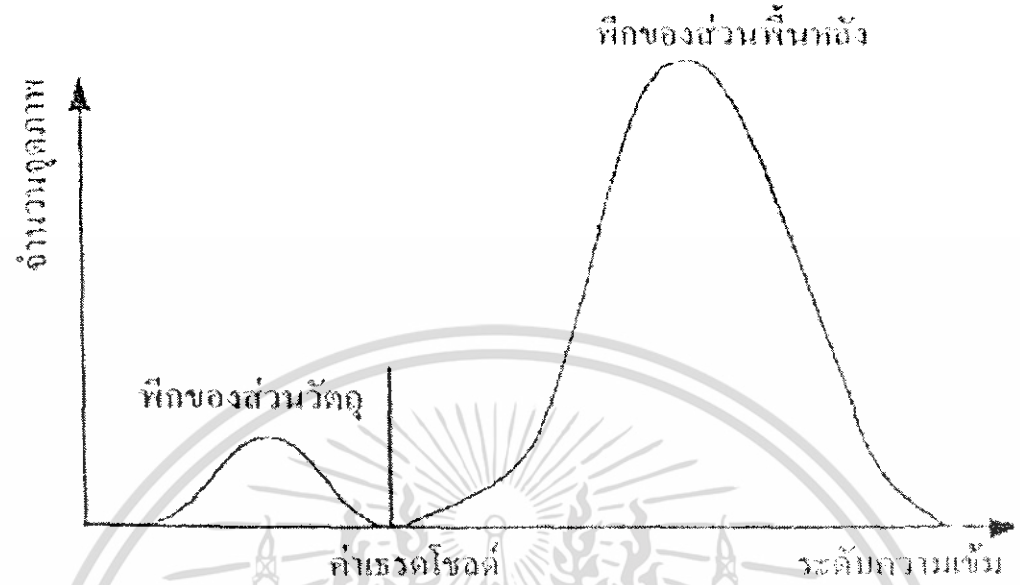
ในการเลือกเธรดโชลด์ที่เหมาะสมจะสามารถทำให้แยกองค์ประกอบของภาพได้อย่างถูกต้องมากขึ้น โดยในแต่ละวิธีก็จะเหมาะสมกับภาพที่แตกต่างกันออกไป ดัง 2 วิธีต่อไปนี้

1. การทำเธรดโชลด์แบบครอบคลุม (Global Thresholding)

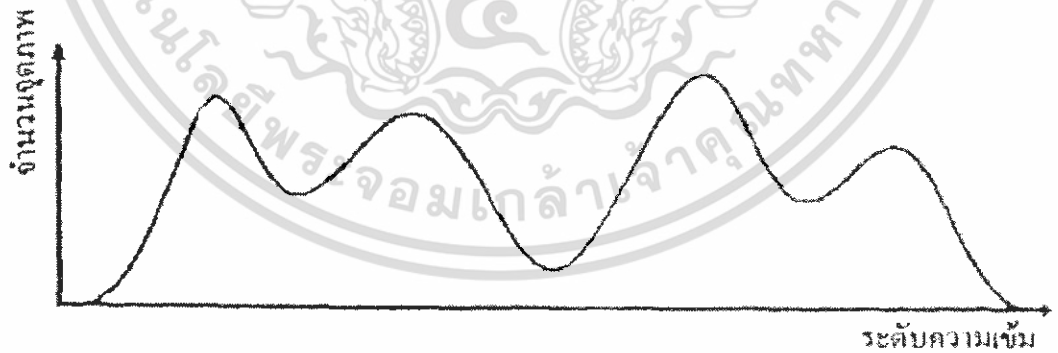
เป็นการหาค่าเธรดโชลด์ที่สามารถนำมาใช้พิจารณาได้ทั่วทั้งภาพ วิธีนี้เหมาะกับภาพที่มีความสม่ำเสมอของส่วนวัตถุและส่วนพื้นหลัง ซึ่งทั้ง 2 ส่วนจะแตกต่างกันอย่างชัดเจนทั่วทั้งภาพ ดังแสดงในรูปที่ 2.8

2. การทำเธรดโชลด์แบบปรับค่า (Adaptive Thresholding)

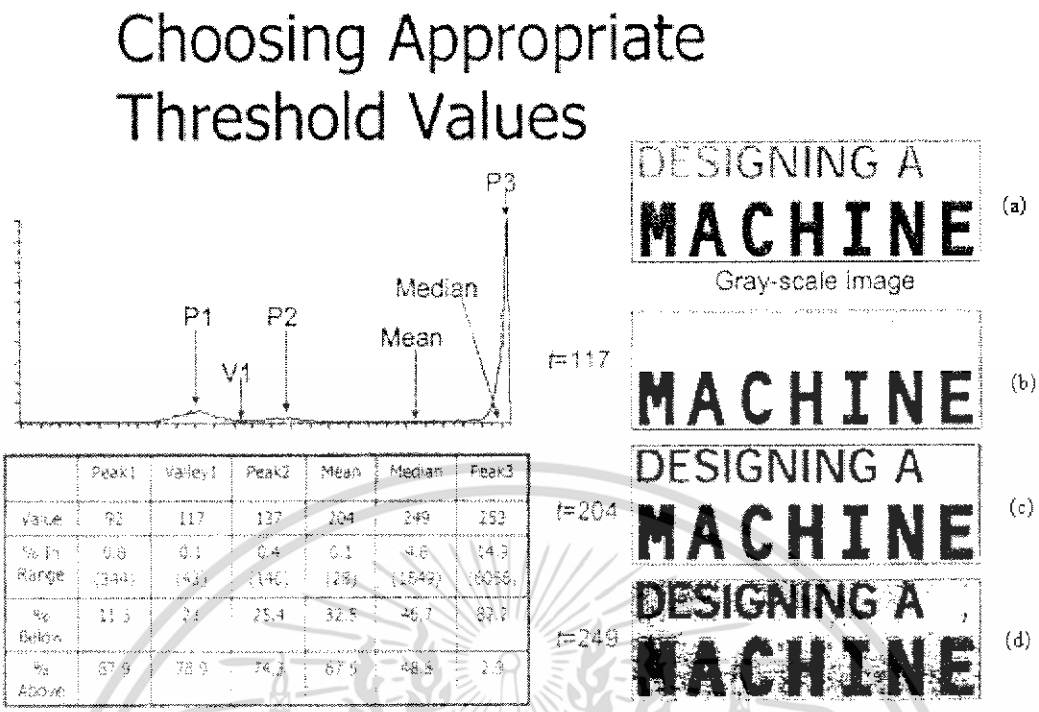
เป็นการหาค่าเธรดโชลด์ที่มีการปรับเปลี่ยนไปตามตำแหน่งของภาพโดยจะแบ่งภาพออกเป็นส่วนย่อย ๆ แล้วจึงเธรดโชลด์ของแต่ละส่วนย่อยนั้น นั่นคือในภาพจะมีค่าเธรดโชลด์หลายค่าที่ต่างกันสำหรับตำแหน่งภาพที่ต่างกัน ดังแสดงในรูปที่ 2.9




รูปที่ 2.8 ฮีสโตแกรมแสดงค่าเธรดโชลด์แบบครอบคลุม [5]




รูปที่ 2.9 ฮีสโตแกรมแสดงค่าเธรดโชลด์แบบปรับค่า [5]

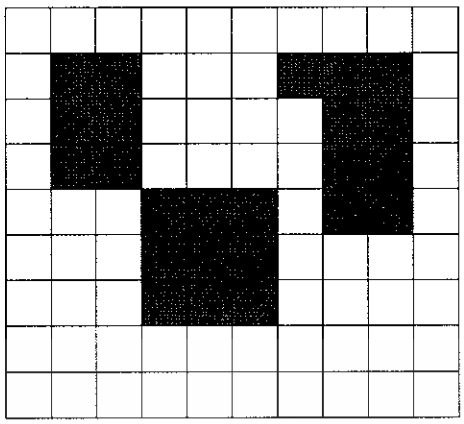



รูปที่ 2.10 การเลือกค่า Threshold ที่เหมาะสม [11]

จากรูปที่ 2.10 (a) ถ้าเลือกค่า Threshold ที่ 204 ซึ่งเป็นค่า Mean จะได้รูปออกมาดีที่สุดเมื่อเปรียบเทียบกับค่า Threshold ที่ 117 ซึ่งมีตัวอักษรบางส่วนหายไปและค่า Threshold ที่ 249 ซึ่งมีสัญญาณรบกวนเป็นจำนวนมาก

**2.5 การวิเคราะห์องค์ประกอบที่เชื่อมต่อกันของภาพ (Connected Components)** [5]

ในภาพๆ หนึ่งอาจจะมีองค์ประกอบหรือวัตถุอยู่วัตถุเดียวหรือมากกว่าหนึ่งวัตถุขึ้นไป โดยในแต่ละวัตถุอาจจะอยู่ในตำแหน่งที่ต่อเนื่องกัน ซ้อนทับกัน หรือห่างจากกัน ซึ่งในภาพที่เป็นแบบไบนารีจะมีการพิจารณาการเชื่อมต่อของวัตถุ โดยพิจารณาจากพิกเซลที่เป็นส่วนของวัตถุกับพิกเซลรอบข้าง ๆ (Neighborhoods) ว่ามีการเชื่อมต่อกันหรือไม่ ซึ่งสามารถพิจารณาได้ทั้งใน 4 ทิศทาง (4-connected Neighborhoods) หรือใน 8 ทิศทาง (8-connected Neighborhoods) ซึ่งถ้าพิกเซลที่เป็นส่วนของวัตถุมีการเชื่อมต่อกันจะถือว่าเป็นวัตถุชนิดเดียวกัน แต่ถ้าพิกเซลที่เป็นส่วนของวัตถุขาดออกจากกันหรือไม่เชื่อมต่อกันจะกำหนดให้เป็นวัตถุที่ต่างกัน ดังแสดงในรูปตัวอย่างดังต่อไปนี้



รูปที่ 2.11 องค์ประกอบของภาพ

จากรูปที่แสดงดังตัวอย่างข้างบนนั้นกำหนดให้พิกเซลสีขาวเป็นส่วนของพื้นหลัง และพิกเซลสีดำเป็นส่วนของวัตถุ เมื่อนำภาพมาวิเคราะห์องค์ประกอบที่เชื่อมต่อกันของภาพทำให้สามารถระบุหมายเลข (Labeling of Connected Components) ให้กับองค์ประกอบของภาพได้ดังรูปต่อไปนี้

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 3 | 3 | 3 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 3 | 3 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 3 | 3 | 0 |
| 0 | 0 | 0 | 2 | 2 | 2 | 0 | 3 | 3 | 0 |
| 0 | 0 | 0 | 2 | 2 | 2 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 2 | 2 | 2 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

รูปที่ 2.12 การระบุหมายเลขให้กับองค์ประกอบของภาพโดยพิจารณาใน 4 ทิศทาง

จากรูปข้างบนเป็นผลการระบุหมายเลขให้กับองค์ประกอบของภาพ ซึ่งภาพจะประกอบไปด้วย 4 องค์ประกอบด้วยกัน โดยหมายเลข 0 แสดงส่วนของพื้นหลัง หมายเลข 1 แสดงวัตถุหนึ่ง หมายเลข 2 แสดงวัตถุที่สอง และหมายเลข 3 แสดงวัตถุที่สามของภาพ ซึ่งจากผลที่ได้ข้างเป็นการวิเคราะห์องค์ประกอบที่เชื่อมต่อกันของภาพโดยใช้ 4 ทิศทาง แต่ถ้าเป็นการวิเคราะห์โดยใช้ 8 ทิศทางจะทำให้ภาพข้างต้นประกอบไปด้วย 2 วัตถุเท่านั้น นั่นคือวัตถุที่หนึ่งและวัตถุที่สองเป็นวัตถุที่เชื่อมต่อกันในแนวทแยงจึงถือว่าเป็นวัตถุเดียวกัน

62579

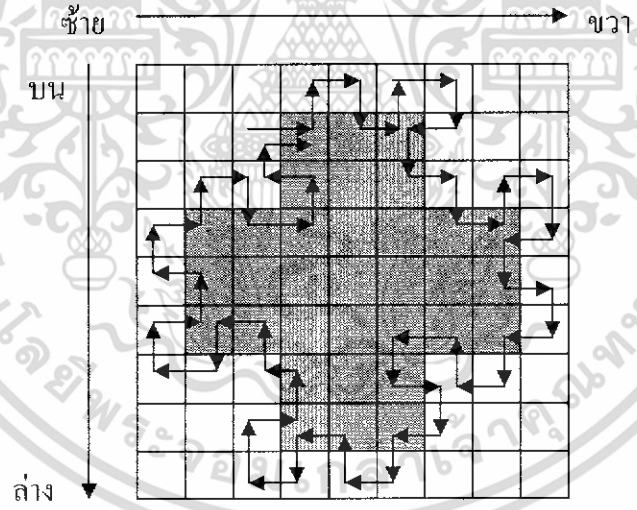
**2.6 การติดตามเส้นขอบ (Contour Following)** [5]

การติดตามเส้นขอบเป็นการติดตามรอบขอบของวัตถุในภาพ ซึ่งเป็นการไต่ไล่ไปตามขอบโดยใช้ความแตกต่างของสีจุดภาพระหว่างวัตถุกับพื้นหลัง ซึ่งโดยทั่วไปแล้ววิธีการนี้จะใช้กับภาพไบนารี เช่น แทนจุดภาพดำเป็นส่วนของวัตถุ และแทนจุดภาพขาวเป็นส่วนของพื้นหลัง ดังนั้นในการทำงานจะตรวจกวาดไปทุก ๆ จุดภาพ โดยเริ่มจากจุดภาพแรกที่มุมซ้ายบนของภาพ และตรวจกวาดไปในทิศทางจากซ้ายไปขวาและเลื่อนจากบนลงล่าง เมื่อตรวจกวาดมาพบจุดภาพดำจุดแรกซึ่งเป็นส่วนของวัตถุ จะเปลี่ยนลักษณะการเคลื่อนที่ไปยังจุดภาพถัดไปใหม่ โดยมีเงื่อนไขการเคลื่อนที่ดังต่อไปนี้

กรณีที่จุดภาพปัจจุบันเป็นจุดภาพของวัตถุให้เลี้ยวซ้ายแล้วเคลื่อนตรงไปยังจุดภาพข้างหน้าถัดไป

กรณีที่จุดภาพปัจจุบันเป็นจุดภาพของพื้นหลังให้เลี้ยวขวาแล้วเคลื่อนตรงไปยังจุดภาพข้างหน้าถัดไป

ทำการเคลื่อนที่จุดภาพที่พิจารณาเช่นนี้ไปเรื่อย ๆ จนกว่าจุดปัจจุบันเป็นจุดเดียวกันกับจุดที่เริ่มต้นของวัตถุ วิธีการหาขอบเขตของภาพสามารถแสดงได้ดังรูปต่อไปนี้

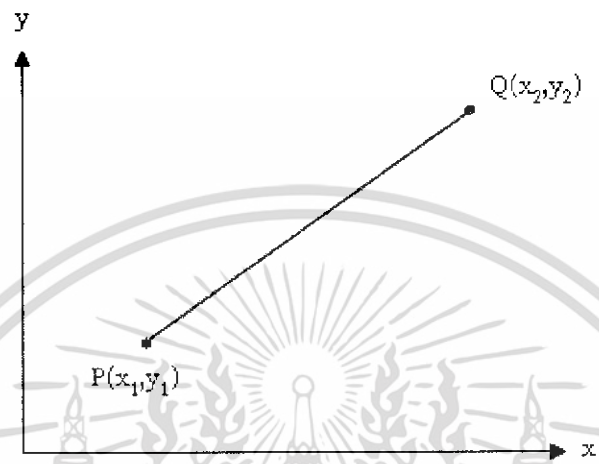



รูปที่ 2.13 วิธีการติดตามเส้นขอบ

## 2.7 ทฤษฎีทางคณิตศาสตร์

### 2.7.1 การหาระยะทางระหว่างจุดสองจุด [5]

จุดสองจุดที่อยู่บนระนาบพิกัดสามารถหาความยาวระหว่างสองจุดนั้นได้ ดังรูปต่อไปนี้



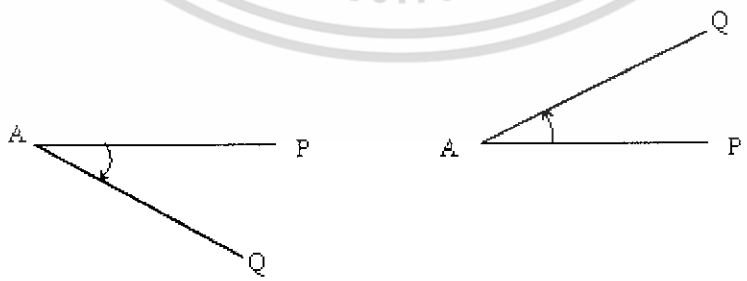

รูปที่ 2.14 ระยะห่างระหว่างจุดสองจุด

จากรูป ให้ $P(x_1,y_1)$ และ $Q(x_2,y_2)$ เป็นจุดสองจุดบนระนาบ ดังนั้นระยะห่างระหว่างจุด P ถึงจุด Q สามารถคำนวณได้จากสมการดังต่อไปนี้

$$|PQ| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \qquad (23)$$

### 2.7.2 การวัดมุม[5]

ถ้าหมุนส่วนโค้งของเส้นตรง AP รอบจุด A ไปอยู่ในแนว AQ สิ่งที่เกิดขึ้นจะเรียกว่ามุม แสดงดังรูปตัวอย่างต่อไปนี้




รูปที่ 2.15 ตัวอย่างมุมที่เกิดขึ้น

ส่วนต่าง ๆ ที่เกี่ยวข้องกับมุมมีดังนี้

จุด A เรียกว่าจุดยอด (Vertex) ของมุม

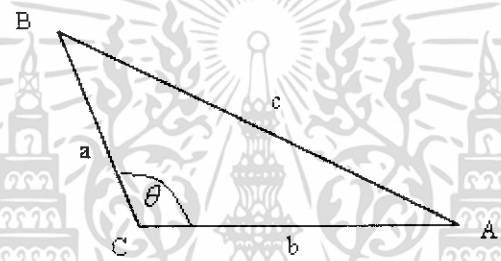
เส้นตรง AP เรียกว่าด้านเริ่มต้น (Initial side) ของมุม

เส้นตรง AQ เรียกว่าด้านสิ้นสุด (Terminal side) ของมุม

หน่วยในการวัดมุมที่นิยมใช้คือองศา โดยถือว่ามุมที่เกิดจากการหมุนส่วนของเส้นตรงไปครบหนึ่งรอบมีขนาด 360 องศา และหน่วยในการวัดมุมที่สำคัญอีกหน่วยคือเรเดียน (Radian) โดย 360 องศา เท่ากับ $2\pi$ เรเดียน

**2.7.3 กฎของโคไซน์ (The law of Cosines)** [5]

ในความสัมพันธ์ของด้านและมุมของรูปสามเหลี่ยมใด ๆ สามารถแสดงได้ด้วยกฎของโคไซน์ ดังตัวอย่างของรูปต่อไปนี้




รูปที่ 2.16 ตัวอย่างสามเหลี่ยม

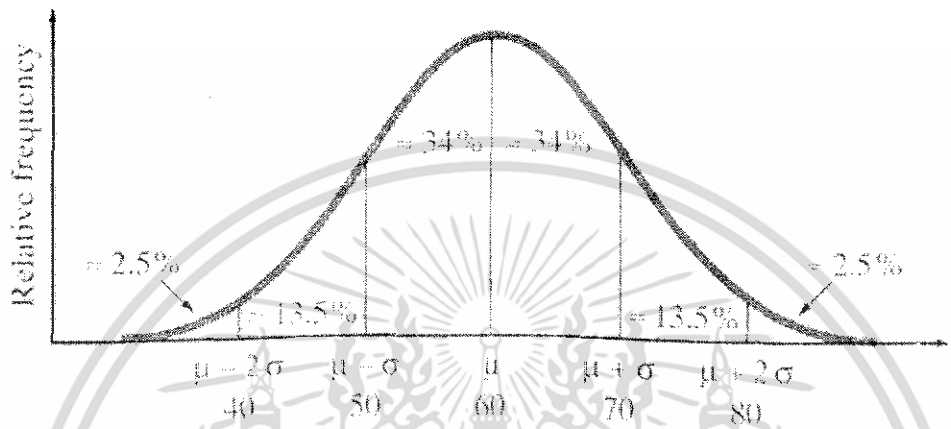
จากรูป a, b และ c เป็นด้านของสามเหลี่ยม ABC โดย $\theta$ เป็นมุมตรงข้ามกับด้าน c จะสามารถเขียนความสัมพันธ์ของกฎโคไซน์ดังสมการต่อไปนี้

$$c^2 = a^2 + b^2 - 2ab\cos\theta \qquad (24)$$

**2.7.4 การแจกแจงปกติ (Normal distribution)** [12]

เป็นการแจกแจงความน่าจะเป็นของข้อมูลที่มีค่าต่อเนื่องกัน เมื่อนำข้อมูลมาพล็อตตามแบบเส้นโค้งความถี่แล้ว ได้ลักษณะเส้นโค้งแบบระฆังคว่ำ ด้านซ้ายและด้านขวาของเส้นโค้งมีส่วนเท่ากัน หรือเรียกว่ามีความสมมาตร (Symmetry) ส่วนสูงของเส้นโค้งอยู่ตรงกลางพอดี และตรงกับค่าเฉลี่ย ($\mu$) ของข้อมูล X เป็นตัวแปรซึ่งแทนค่าใดๆ โดยปกติการแจกแจงใดๆ นั้น ตัวแปรใดๆ ที่มีจำนวนมากจะมีการแจกแจงใกล้เคียงโค้งปกติ เส้นโค้งจะถูกกำหนดด้วยค่าคงตัวสองค่า หรือที่เรียกว่าค่าพารามิเตอร์ (Paramrater) ได้แก่ค่าเฉลี่ย ($\mu$) และส่วนเบี่ยงเบนมาตรฐาน ($\sigma$) นั่นคือตัวแปรนั้นจะมีค่าพารามิเตอร์ที่ควบคุม หรือกำหนดลักษณะของมันได้ตามคุณสมบัติของการแจกแจงปกติ และการทดสอบตามวิธีการสถิติแบบใด ๆ เกี่ยวกับตัวแปรนั้น สามารถใช้คุณสมบัติแห่งการแจกแจงปกติได้ และเรียกการทดสอบแบบนั้นว่าแบบพาราเมตริก (Parametric test)

ค่าเฉลี่ย ($\mu$) แสดงแนวกลางของข้อมูล และค่าส่วนเบี่ยงเบนมาตรฐาน ($\sigma$) แสดงถึงการกระจายของข้อมูล ดังในรูปที่ 2.17




รูปที่ 2.17 การแจกแจงแบบปกติ [13]

**2.7.4.1 คุณสมบัติการแจกแจงแบบปกติ** [12]

1. มีลักษณะของรูประฆังคว่ำ มีความสมมาตร (Symmetry) ทั้ง 2 ด้าน ปลายทั้ง 2 ด้านของเส้นโค้งขนานกับแนวนอนที่ $-\infty$ และ $+\infty$

2. มี $\mu$ เป็นแกนกลางมีค่าเฉลี่ย ($\mu$) = 0 และส่วนเบี่ยงเบนมาตรฐาน ($\sigma$) = 1

3. พื้นที่และความน่าจะเป็นภายใต้เส้นโค้งมีค่าเท่ากับ 1

4. คุณสมบัติเกี่ยวกับความน่าจะเป็น (พื้นที่) ภายใต้เส้นโค้ง

(1) ค่าของตัวแปรซึ่งอยู่ระหว่าง $\mu-\sigma$ กับ $\mu+\sigma$ มีจำนวนสมาชิก 68 เปอร์เซ็นต์ของจำนวนทั้งหมดในข้อมูล

(2) ค่าตัวแปรซึ่งอยู่ระหว่าง $\mu-2\sigma$ กับ $\mu+2\sigma$ มีจำนวนสมาชิก 95 เปอร์เซนต์ของจำนวนทั้งหมดในข้อมูล

(3) ค่าตัวแปรซึ่งอยู่ระหว่าง $\mu-3\sigma$ กับ $\mu+3\sigma$ มีจำนวนสมาชิก 99 เปอร์เซนต์ของจำนวนทั้งหมดในข้อมูล

5. มีค่า Z เป็นปกติมาตรฐาน(Standard normal variable)

$$z = \frac{x - \mu}{\sigma}$$

**2.7.4.2 ค่าเฉลี่ยและค่าเบี่ยงมาตรฐานของการแจกแจงปกติ** [12]

ค่าเฉลี่ย (mean) คือ ผลรวมของค่าข้อมูลทุกค่าหารด้วยจำนวนข้อมูลทั้งหมด ค่าเฉลี่ยเป็นค่าปานกลางของทุกค่าในข้อมูล นั่นคือความสูงต่ำของแต่ละค่าถูกเกลี่ยลงหรือขึ้นให้เสมอกัน ซึ่งค่าเฉลี่ยจะมีสมการในการคำนวณดังนี้

การหาค่าเฉลี่ยมาตรฐาน

$$\mu = \sum_{i=1}^{N} \frac{x_i}{N} = \frac{x_1 + x_2 + \ldots + x_N}{N} \qquad (25)$$

$\mu$ ค่าเฉลี่ยมาตรฐาน

$x_i$ เป็นค่าข้อมูลตัวที่ i

i เป็นการลำดับที่ของข้อมูลมีค่าตั้งแต่ 1-N

N เป็นจำนวนข้อมูล

$\sum$ ผลรวมของ $x_i$ ตั้งแต่ 1 ถึง N

ค่าเบี่ยงเบนมาตรฐาน (Standard Deviation) เป็นการวัดการกระจายของข้อมูล ซึ่งเป็นการวัดระยะห่างของค่าข้อมูลแต่ละค่าจากค่าเฉลี่ย ทำให้สามารถศึกษาได้ถึงความแตกต่างหรือการกระจายของค่าในกลุ่มข้อมูล ซึ่งค่าเบี่ยงเบนมาตรฐานจะมีวิธีการคำนวณดังต่อไปนี้

การหาค่าเบี่ยงเบนมาตรฐาน

$$\sigma = \sqrt{\sum_{i=1}^{N} \frac{(x_i - \mu)^2}{N}} \qquad (26)$$

$\sigma$ ค่าเบี่ยงเบนมาตรฐาน

$x_i$ เป็นค่าข้อมูลตัวที่ i

$\mu$ ค่าเฉลี่ยมาตรฐาน

i เป็นการลำดับที่ของข้อมูลมีค่าตั้งแต่ 1-N

N เป็นจำนวนข้อมูล

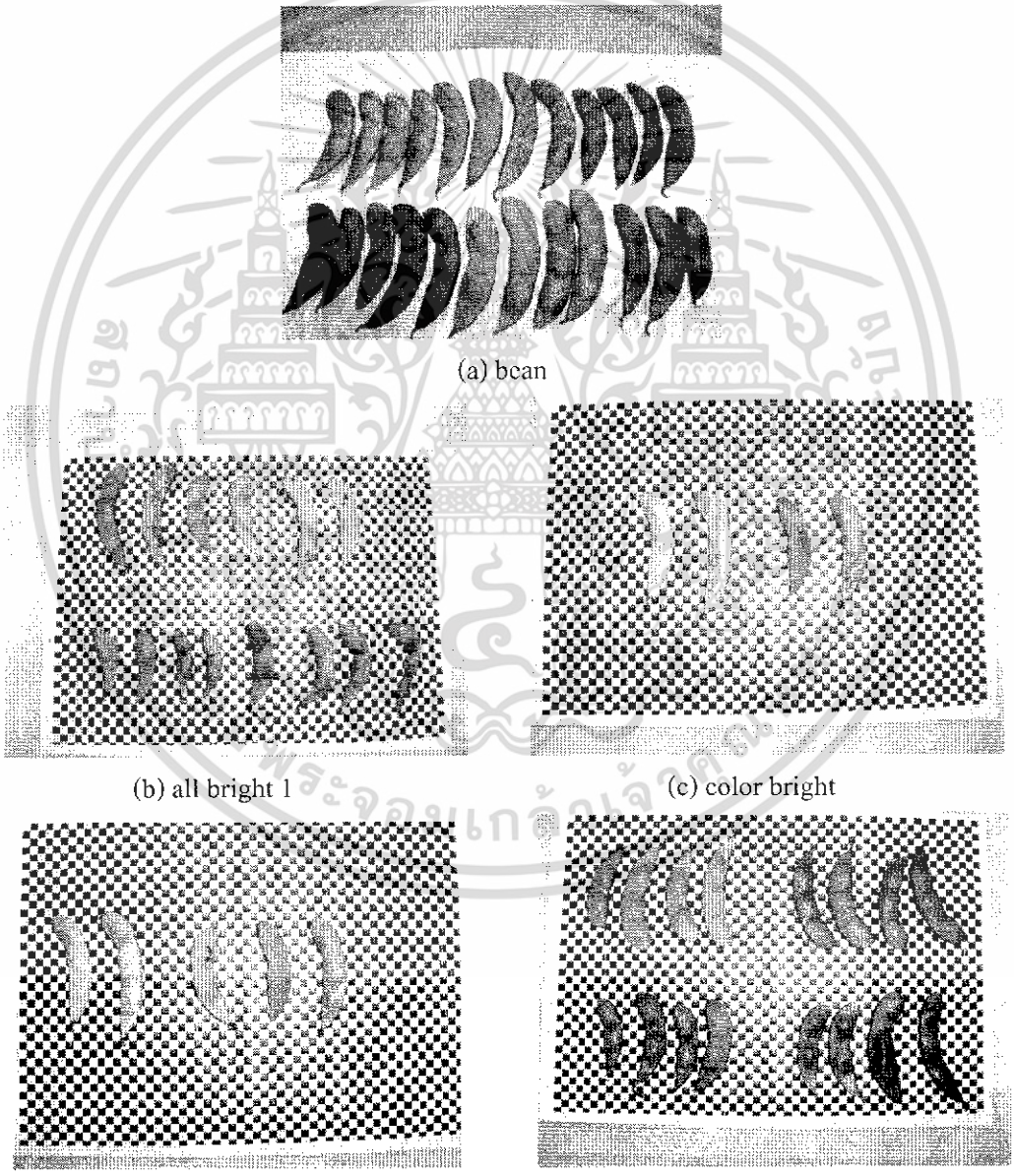
$\sum$ ผลรวมของ $X_i$ ตั้งแต่ 1 ถึง N

# บทที่ 3

## วิธีดำเนินงาน

ในงานวิจัยนี้ได้ใช้ถั่วเหลืองฝักสดเป็นภาพตัวอย่างในการวิเคราะห์ โดยถั่วเหลืองฝักสดที่ใช้เป็นถั่วเหลืองฝักสดพันธ์ AGS 292 จากจังหวัดเชียงใหม่ ซึ่งเป็นพันธุ์ที่ใช้เพื่อการส่งออก

ในการทดลองนั้นจะทดลองกับถั่วเหลืองฝักสดหลายแบบ อย่างเช่น แบบ 3 เมล็ด 2 เมล็ดใหญ่ 2 เมล็ดเล็ก 1 เมล็ด ทั้งที่เป็นฝักสีเขียวและฝักสีเหลือง ซึ่งจะเก็บภาพถั่วเหลืองฝักสดทั้งหมด 63 ฝักดัง รูปที่ 3.1



(a) bean

(b) all bright 1

(c) color bright

(d) color&spot bright

(e) cook color bright 1

รูปที่ 3.1 ภาพตัวอย่างลักษณะของถั่วเหลืองฝักสดที่นำมาวิเคราะห์

### 3.1 การเขียนโปรแกรม

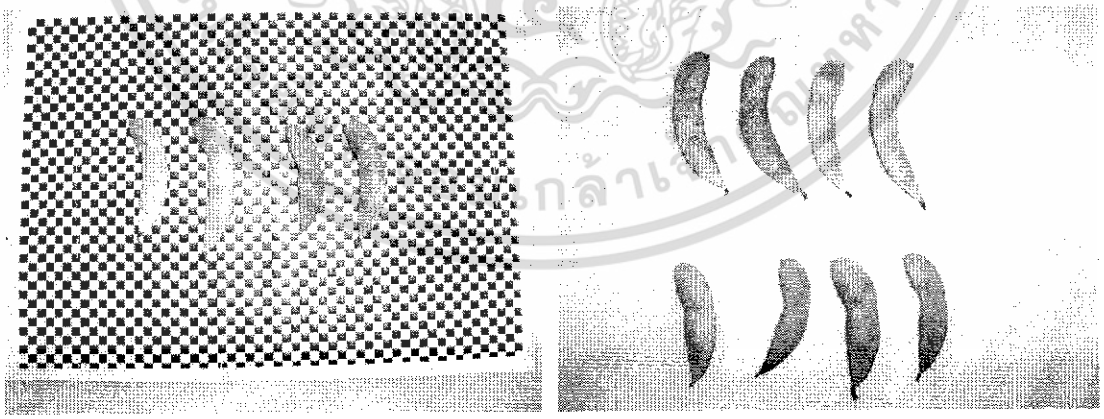
มีขั้นตอนตามรูปที่ 3.3 ซึ่งโปรแกรมที่เขียนแบ่งออกเป็น 2 ส่วน คือ โปรแกรมหลัก (bean_sorting) และโปรแกรมย่อย (bound_edge_v) โปรแกรมหลักจะคำนวณขนาดพื้นที่ภาพฉาย สี และจุดด่างดำ ส่วนโปรแกรมย่อย จะคำนวณมุมทุกจุดบนเส้นขอบ แล้วส่งค่ามุมมาให้โปรแกรมหลักทำการหาค่าความแปรปรวนของมุม

### 3.2 การเก็บข้อมูลตัวอย่างภาพถั่วเหลืองฝักสด

ในการเก็บภาพถั่วเหลืองฝักสดนั้นจะทำการถ่ายภาพถั่วเหลืองฝักสดที่วางอยู่บนกระดาษตารางและบนกระดาษสีขาว ถ่ายรูปโดยใช้มุมภาพด้านบน ซึ่งภาพจะเก็บอยู่ในแบบสี RGB
อุปกรณ์ที่ใช้ในการถ่ายภาพถั่วเหลืองฝักสดมีดังต่อไปนี้

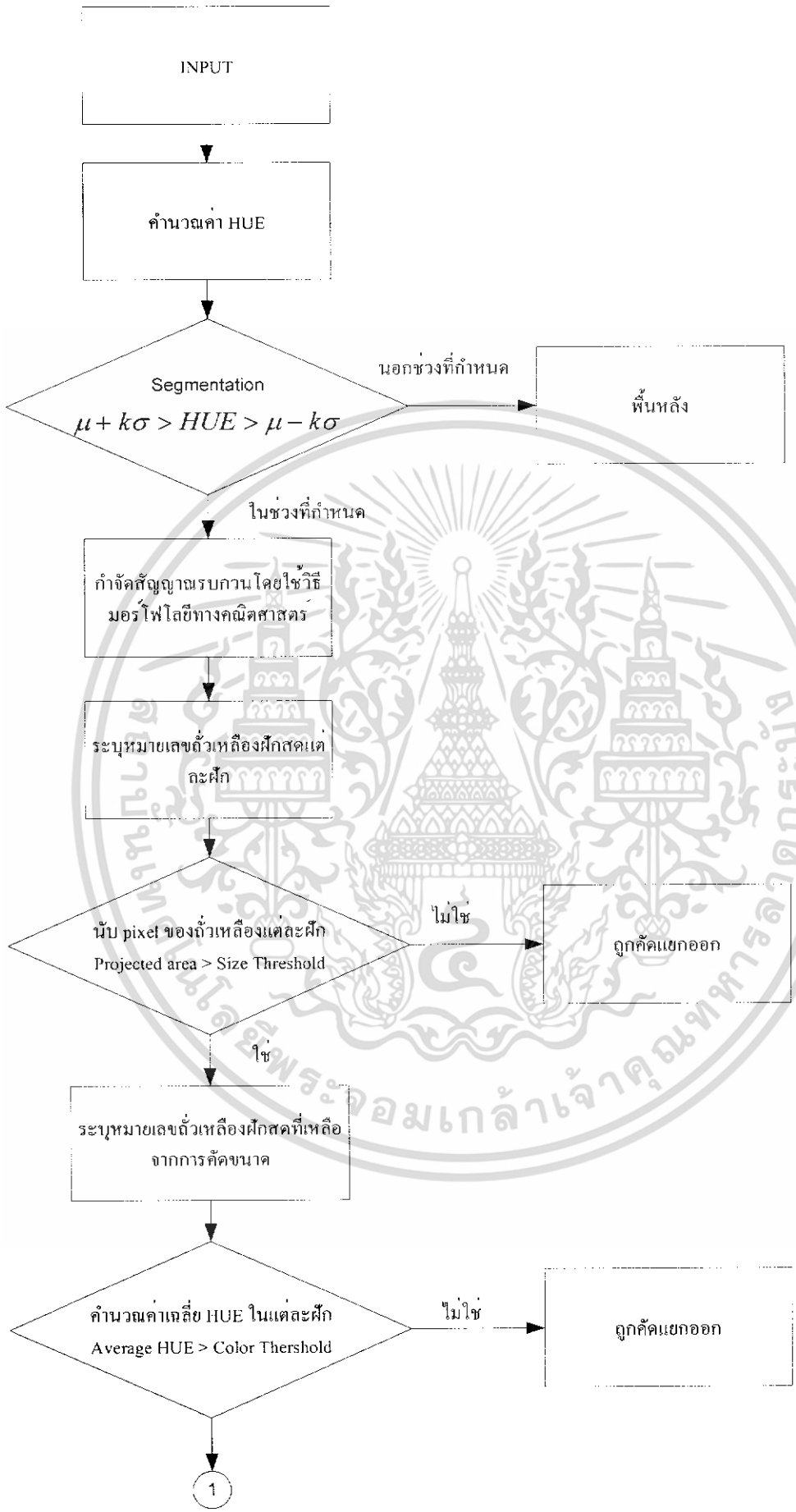
1. กล้องดิจิตอล 1 เครื่อง
2. กระดาษตารางขาวดำ 1 แผ่น(1 ช่อง มีค่าเท่ากับ 0.25 ตร.ซม.)
3. กระดาษสีขาว

ภาพที่ได้นั้นจะเป็นภาพสีซึ่งเป็นภาพของถั่วเหลืองฝักสดที่มีแผ่นตารางสีขาวดำเป็นพื้นหลัง โดยมีถั่วเหลืองฝักสดวางอยู่บนกระดาษหลายฝักด้วยกัน จะได้ภาพดังนี้

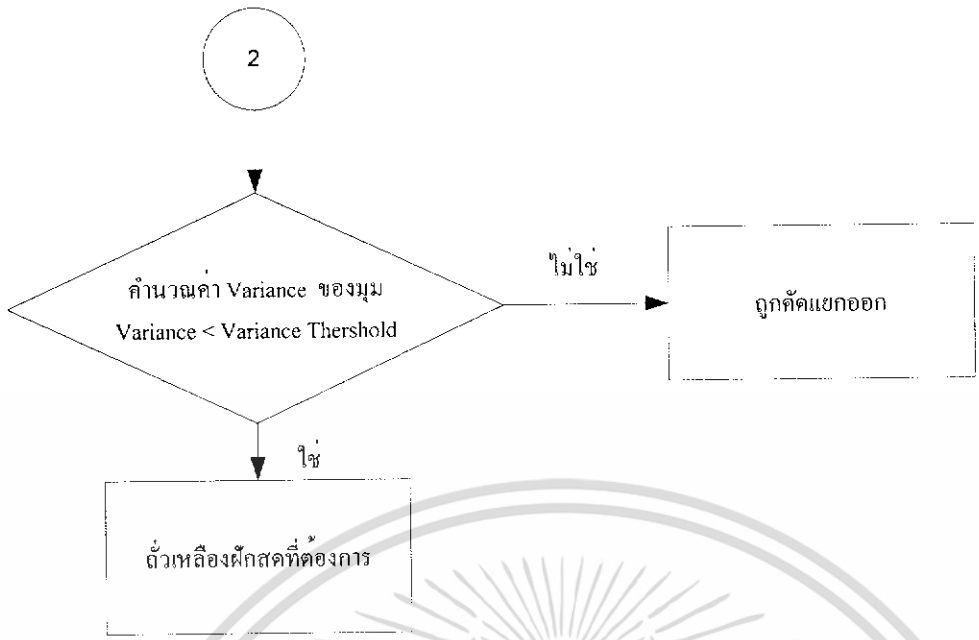


(a) (b)

รูปที่ 3.2 (a) ภาพถั่วเหลืองฝักสดบนกระดาษตารางขาวดำ
(b) ภาพถั่วเหลืองฝักสดบนพื้นกระดาษสีขาว


INPUT
คำนวณค่า HUE
Segmentation
$\mu + k\sigma > HUE > \mu - k\sigma$
นอกช่วงที่กำหนด
พื้นหลัง
ในช่วงที่กำหนด
กำจัดสัญญาณรบกวนโดยใช้วิธีมอร์โฟโลยีทางคณิตศาสตร์
ระบุหมายเลขถั่วเหลืองฝักสดแต่ละฝัก
นับ pixel ของถั่วเหลืองแต่ละฝัก
Projected area > Size Threshold
ไม่ใช่
ถูกคัดแยกออก
ใช่
ระบุหมายเลขถั่วเหลืองฝักสดที่เหลือจากการคัดขนาด
คำนวณค่าเฉลี่ย HUE ในแต่ละฝัก
Average HUE > Color Thershold
ไม่ใช่
ถูกคัดแยกออก
1

1
ระบุหมายเลขถั่วเหลืองฝักสด
ที่เหลือจากการคัดแยกโดยขนาด
และสี
หาจุด Centroid
ของถั่วเหลืองฝักสดแต่ละฝัก
จุดด่างดำโดยใช้ hole บนถั่วเหลืองฝักสด
มี
ไม่มี
ระยะจากจุดด่างดำถึงจุด Centroid
น้อยสุด
ถูกคัดแยกออก
ระบุหมายเลขถั่วเหลืองฝักสด
ที่เหลือจากการคัดแยกโดยขนาด
สี และรูปร่าง
หาเส้นขอบของถั่วเหลืองฝักสด
โดยใช้วิธีติดตามเส้นขอบและส่ง
ภาพไปที่โปรแกรมย่อย
คำนวณขนาดมุมทุกจุด pixel
บนเส้นขอบแล้วส่งค่ามุมออกมา
ที่โปรแกรมหลัก
2



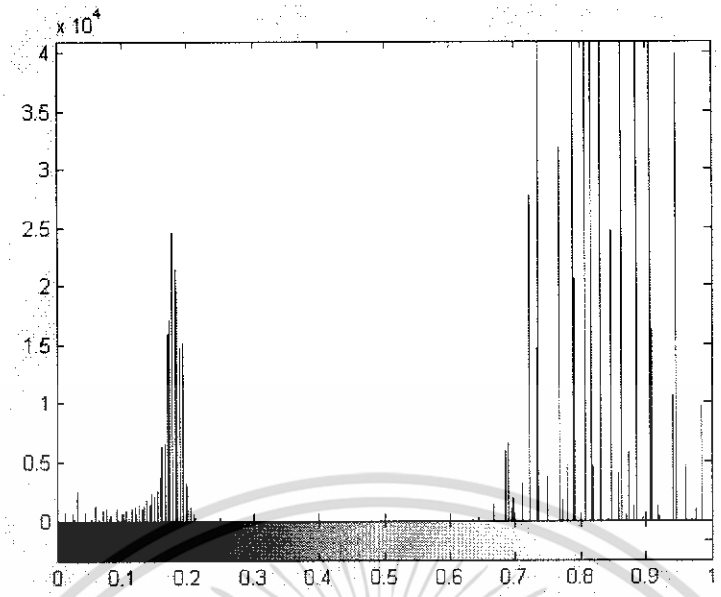

รูปที่ 3.3 แผนผังการทำงานของโปรแกรม

### 3.3 การแยกส่วนของถั่วเหลืองฝักสดออกจากพื้นหลัง (Segmentation)

ในการวิเคราะห์ลักษณะทางกายภาพของถั่วเหลืองฝักสดจะวิเคราะห์กับภาพถ่ายดังแสดงในรูป 3.2 ซึ่งภาพนั้นเป็นภาพของถั่วเหลืองฝักสดหลายฝัก กับพื้นหลังที่ประกอบไปด้วยหลายกลุ่มสี ดังนั้นก่อนที่จะสามารถวิเคราะห์ลักษณะทางกายภาพของถั่วเหลืองฝักสดได้ จึงจะต้องแยกส่วนของถั่วเหลืองฝักสดออกจากพื้นหลังเสียก่อน ซึ่งก็จะมีวิธีในการดำเนินการต่อไปนี้

#### 3.3.1 การสร้างแบบจำลองค่า Hue ของถั่วเหลืองฝักสด

จะเริ่มต้นโดยการนำภาพถ่ายตัวอย่างของถั่วเหลืองฝักสดที่มีพื้นหลังสีขาวในมุมภาพด้านบน ซึ่งเป็นภาพที่อยู่ในรูปแบบของสี RGB จากนั้นจะทำการแปลงค่าสีของภาพจาก RGB เป็นค่า HSI เพื่อนำมาสร้างแบบจำลองค่า Hue เฉพาะในส่วนของถั่วเหลืองฝักสด เมื่อนำมาทำเป็นฮีสโตแกรมของค่า Hue จะได้ดังรูปที่ 3.4 ซึ่งจะอยู่ในรูปแบบของการแจกแจงปกติโดยมีค่าเฉลี่ยที่ได้เป็น 0.226 และค่าเบี่ยงเบนมาตรฐานเป็น 0.0215 ดังนั้นจึงสามารถนำแบบจำลองค่า Hue ของถั่วเหลืองฝักสดที่สร้างขึ้นมาใช้ในการพิจารณากับภาพเพื่อแยกส่วนของถั่วเหลืองฝักสดออกจากพื้นหลัง




รูปที่ 3.4 ฮีสโตแกรมค่า Hue รูปถั่วเหลืองฝักสดกับพื้นหลังสีขาว มี 2 ลูป ลูปที่หนึ่งเป็นค่า Hue ของถั่วเหลืองฝักสด ส่วนลูปที่ 2 เป็นลูปของพื้นหลัง

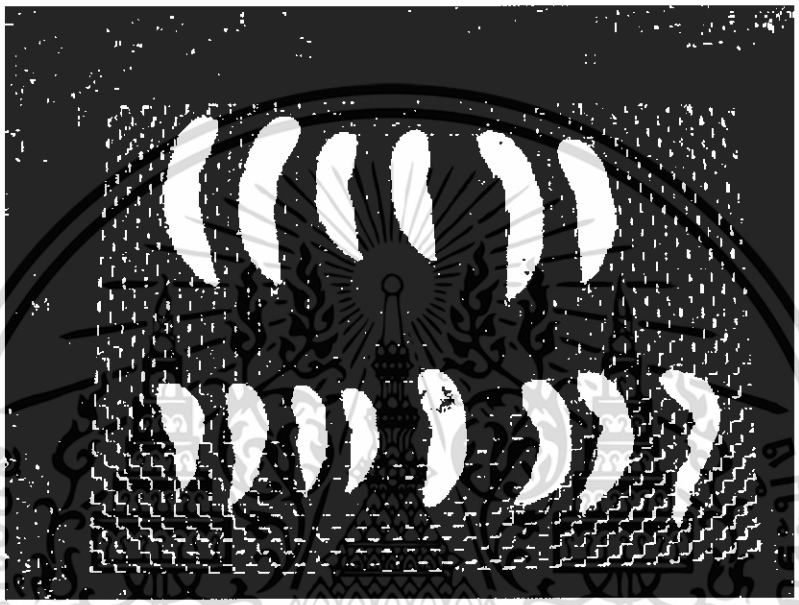
### 3.3.2 การหาค่าเฉลี่ยมาตรฐานและค่าเบี่ยงเบนมาตรฐาน

จะเริ่มต้นโดยการนำภาพถั่วเหลืองฝักสดทั้ง 24 ฝัก ซึ่งเป็นภาพ RGB มา crop เอาแต่เฉพาะส่วนที่เป็นรูปของถั่วเหลืองฝักสดดังรูปที่ 3.5 นำไปแปลงเป็นค่าสี Hue แล้วนำมาค่าสี Hue ที่ได้มาทำการหาค่าเฉลี่ยมาตรฐานและค่าเบี่ยงเบนมาตรฐานได้ตามสูตรในบทที่ 2

รูปที่ 3.5 ภาพถั่วเหลืองฝักสดที่ crop เอาแต่รูปของถั่วเหลืองฝักสด

### 3.3.3 การแยกส่วนของถั่วเหลืองฝักสดออกจากพื้นหลัง

ภาพของถั่วเหลืองฝักสดตัวอย่างที่มีรูปแบบสี RGB ทั้งหมดจะถูกแปลงมาเป็นค่า Hue จากนั้นจะนำค่า Hue ที่ได้มาพิจารณากับแบบจำลองค่าสี Hue ของภาพของถั่วเหลืองฝักสดที่สร้างขึ้น โดยที่จะกำหนดว่าพิกเซลใดที่มีค่า Hue อยู่ภายในขอบเขตที่กำหนด คือ มีค่าอยู่ระหว่าง $\mu - k\sigma$ ถึง $\mu + k\sigma$ จะถือว่าพิกเซลนั้นเป็นส่วนของถั่วเหลืองฝักสด ทำให้ได้ภาพของถั่วเหลืองฝักสดที่ถูกแยกออกจากพื้นหลัง ดังแสดงในรูปต่อไปนี้



รูปที่ 3.6 ภาพของถั่วเหลืองฝักสดที่ถูกแยกออกจากพื้นหลัง

จากรูปที่ 3.6 เป็นภาพส่วนของถั่วเหลืองฝักสดที่เป็นสีขาวและมีพื้นหลังเป็นสีดำ ซึ่งในส่วนพื้นหลังอาจจะยังมีสัญญาณรบกวนที่เป็นสีขาวเกิดขึ้น ดังนั้นจึงต้องมีการกำจัดสัญญาณรบกวนที่เกิดขึ้นเพื่อการวิเคราะห์จะได้ไม่ผิดเพี้ยนไป โดยใช้วิธีการวิเคราะห์องค์ประกอบที่เชื่อมต่อกันของภาพมาพิจารณาถึงองค์ประกอบที่ใหญ่ของภาพ ซึ่งจะเลือกองค์ประกอบที่ใหญ่ของภาพเป็นถั่วเหลืองฝักสด ทำให้สามารถกำจัดจุดภาพเล็กๆ ที่เกิดขึ้นบนพื้นหลังรอบๆ ส่วนของถั่วเหลืองฝักสดออกไปได้

### 3.3.4 การกำจัดสัญญาณรบกวนโดยใช้มอร์โฟโลยีทางคณิตศาสตร์

ภาพที่ได้จากการ Segmentation จะเป็นภาพไบนารี่ที่มีสัญญาณรบกวน ซึ่งจะสามารถกำจัดสัญญาณรบกวนได้โดยใช้วิธีมอร์โฟโลยีทางคณิตศาสตร์ คือ การทำโอเพนนิง จะเป็นการทำให้จำนวนจุดสีขาวของภาพลดลง แล้วตามด้วยการเพิ่มจำนวนจุดภาพสีขาวลงไปในภาพ เพื่อเป็นการลบแหลมที่ยื่นออกมาเล็กๆน้อยๆ ซึ่งก็จะถูกกำจัดออกหมดจนเหลือภาพในลักษณะเป็นถั่วเหลืองฝักสดที่ไม่มีสัญญาณรบกวนดังรูปที่ 3.7

รูปที่ 3.7 การกำจัดสัญญาณรบกวนโดยทำการโอเพนนิง

### 3.4 การหาขนาดของถั่วเหลืองฝักสด

ในกระบวนการประมวลผลภาพ จะใช้พื้นที่ภาพฉาย (Projected area) คือ พื้นที่ส่วนที่เป็นของถั่วเหลืองฝักสด ซึ่งพื้นที่ภาพฉายจะมีขนาดแปรผันตามขนาดของถั่วเหลืองฝักสด โดยที่พื้นที่ภาพฉายที่มีขนาดใหญ่ถั่วเหลืองฝักสดก็จะมีขนาดใหญ่ด้วยเช่นกัน แบ่งขั้นตอนการทำออกเป็น 2 ขั้นตอน ดังนี้

#### 3.4.1 *Labeling of connected components*

เนื่องจากในภาพ 1 ภาพ มีถั่วเหลืองฝักสดอยู่หลายฝักด้วยกัน เพื่อที่จะได้ทราบว่าขนาดพื้นที่ภาพฉายของถั่วเหลืองฝักใดมีขนาดเท่าไหร่ จึงต้องมีการระบุหมายเลขให้กับถั่วเหลืองฝักสดแต่ละฝัก ดังรูป 3.8

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 |
| 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 |
| 1 | 1 | 1 | 0 | 2 | 2 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 3 | 3 |
| 1 | 1 | 1 | 0 | 0 | 3 | 3 | 3 |

รูปที่ 3.8 การกำหนดหมายเลขให้กับถั่วเหลืองฝักสด

จากรูป 3.8 จะเห็นว่าถั่วเหลืองฝักสดฝักแรกที่พบได้ถูกกำหนดด้วยเลข 1 ฝักที่พบต่อมาจะถูกกำหนดด้วยเลข 2 และเลข 3 ตามลำดับแบบนี้ไปเรื่อยๆจนครบทุกฝัก

**3.4.2 การหาพื้นที่ภาพฉาย (Projected area)**

การหาขนาดพื้นที่ภาพฉายมีวิธีการคำนวณโดยนำจำนวนพิกเซลทั้งหมดของส่วนที่เป็นถั่วเหลืองฝักสดแต่ละฝัก มาเปรียบเทียบกับอัตราส่วนของจำนวนพิกเซลพื้นที่หน่วยตารางเซนติเมตร ซึ่งอัตราส่วนมารตฐานจะใช้เทียบกับตารางสีขาวดำที่ได้ทำขึ้นมา

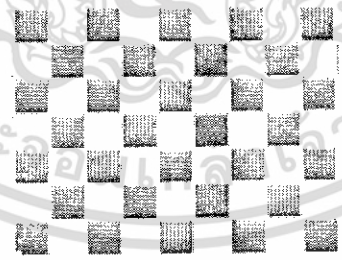
การหาจำนวนพิกเซลของตารางสีขาวดำทำได้โดยการนำภาพส่วนที่เป็นกระดาษสีขาวดำดังรูปที่ 3.9 ไปนับจำนวนพิกเซลใน Paint shop คำนวณดูว่า 1 $cm^2$ มีกี่พิกเซล ซึ่งจำนวนพิกเซลต่อ 1 $cm^2$ จะแปรผันไปตามการซูมเข้าซูมออกของกล้อง ถ้าเปลี่ยนตำแหน่งการวางกล้องจำนวนพิกเซลต่อ 1 $cm^2$ ก็จะเปลี่ยนไป การหาขนาดของถั่วเหลืองฝักสดหาได้ตามสูตร

$$S = \frac{P_b}{P_{bw}} \qquad (3.1)$$

S = ขนาดของถั่วเหลืองฝักสด($cm^2$)

$P_b$ = จำนวนพิกเซลของถั่วเหลืองฝักสด

$P_{bw}$ = จำนวนพิกเซลตารางขาวดำ 1 $cm^2$



รูปที่ 3.9 ตารางสีขาวดำที่นำไปนับจำนวนพิกเซลต่อ 1 $cm^2$ (1 ช่องเท่ากับ 0.5 x 0.5 $cm^2$)

### 3.5 สีของถั่วเหลืองฝักสด (Color)

ในการส่งออกของถั่วเหลืองฝักสดจะต้องคำนึงถึงสีของถั่วเหลืองฝักสดด้วย โดยพันธ์ของถั่วเหลืองฝักสดที่นำมาทดลอง คือ พันธุ์ AGS 292 ซึ่งเป็นพันธุ์เพื่อการส่งออก ถั่วเหลืองฝักสดที่ส่งออกจะต้องมีสีเขียวสด ไม่มีสีเหลือง สีน้ำตาลคล้ำ หรือจุดด่างดำปะปนอยู่ด้วย จึงจำเป็นต้องมีการคัดแยกเอาถั่วเหลืองฝักสดที่มีสีอื่นปะปนอยู่ด้วยออกก่อนที่จะนำส่งออกต่างประเทศ ซึ่งในการคัดแยกสี และจุดด่างดำบนผิวของถั่วเหลืองฝักสด มีขั้นตอนการดำเนินการต่อไปนี้

#### 3.5.1 การคัดแยกสีของถั่วเหลืองฝักสด

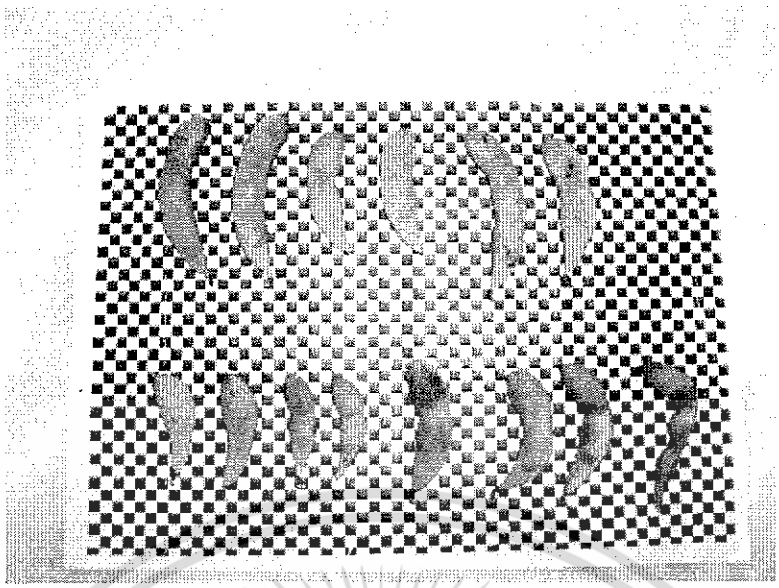
มีวิธีการดำเนินการโดยใช้ค่า Hue ที่หาได้ตั้งแต่ต้นเป็นค่าที่ใช้วัดความแตกต่างของสี โดยการนำค่า Hue ของถั่วเหลืองแต่ละฝักมาเฉลี่ย ถ้าค่าเฉลี่ย Hue ของถั่วเหลืองฝักสดมีค่ามากกว่าค่า Color Threshold ที่กำหนดไว้ ก็แสดงว่าถั่วเหลืองฝักสดฝักนั้นก็มีสีเขียวสดตามต้องการ แต่ถ้าค่าเฉลี่ย Hue มีค่าน้อยกว่า Color Threshold ก็แสดงว่าเป็นถั่วเหลืองฝักสดฝักนั้นมีสีเหลืองและสีน้ำตาลคล้ำปะปนอยู่ด้วย ก็จะถูกคัดออกดังแสดงในรูปที่ 3.10 และการหาค่าเฉลี่ย Hue ของถั่วเหลืองฝักสดเป็นไปตามสูตรต่อไปนี้

$$H_{avg} = \frac{\sum_{1}^{n} h}{n} \tag{3.2}$$

$H_{avg}$ = ค่า Hue เฉลี่ยของถั่วเหลืองฝักสดแต่ละฝัก

h = ค่า Hue แต่ละพิกเซลของถั่วแต่ละฝัก

n = จำนวนพิกเซลของแต่ละฝัก



(a)

(b)

รูปที่ 3.10 (a) ถั่วเหลืองฝักสดที่มีหลายสีปะปนกันอยู่
(b) ถั่วเหลืองที่มีสี Hue น้อยกว่า Color threshold ถูกคัดออก

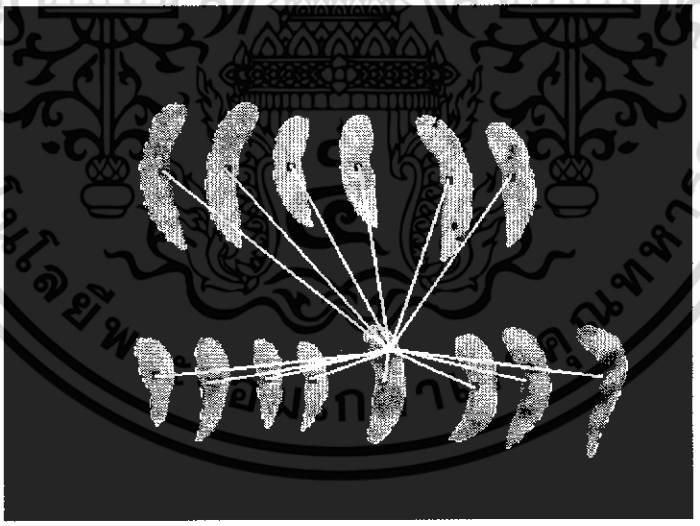
จากรูปที่ 3.10 จะเห็นว่าถั่วเหลืองฝักสดทั้งที่มีฝักสีเขียวสด ฝักที่มีจุดด่างดำ ฝักที่มีสีน้ำตาลและสีเหลืองปะปน เมื่อทำการวิเคราะห์แล้วฝักที่มีสีเหลืองและสีน้ำตาลปะปนอยู่ก็จะถูกคัดออก

**3.5.2 การหาจุดด่างดำบนฝักถั่วเหลืองฝักสด**

เมื่อผ่านการ segmentation มาแล้ว ส่วนที่เป็นจุดด่างดำบนฝักถั่วเหลืองฝักสดจะมี Hole เกิดขึ้น คือเกิดเป็นจุดสีดำในขณะที่ส่วนที่เป็นถั่วเหลืองฝักสดเป็นสีขาวดังรูปที่ 3.7 เนื่องจากใช้ค่า Threshold ของค่าสี Hueในการตัดค่าสี ถ้าพิกเซลใดมีค่าสีต่ำกว่าค่า Threshold ที่กำหนดก็จะเกิดเป็นหลุมจุดด่าง

ดำบนฝักถั่วดังรูปที่ 3.11 เมื่อได้จุดด่างดำแล้ว เพื่อที่จะได้รู้ว่าจุดด่างดำนี้เป็นของฝักไหน ก็จะต้องทำการหาจุด Centroid ของถั่วเหลืองฝักสดแต่ละฝักดังรูปที่ 3.12 ก่อน และหาระยะห่างจากจุด Centroid กับจุดด่างดำ ถ้าฝักไหนมีระยะห่างน้อยที่สุดแสดงว่าจุดด่างดำเป็นของฝักนั้น

รูปที่ 3.11 จุดสีขาวคือจุดด่างดำที่เกิดขึ้นบนฝักถั่วเหลืองฝักสด



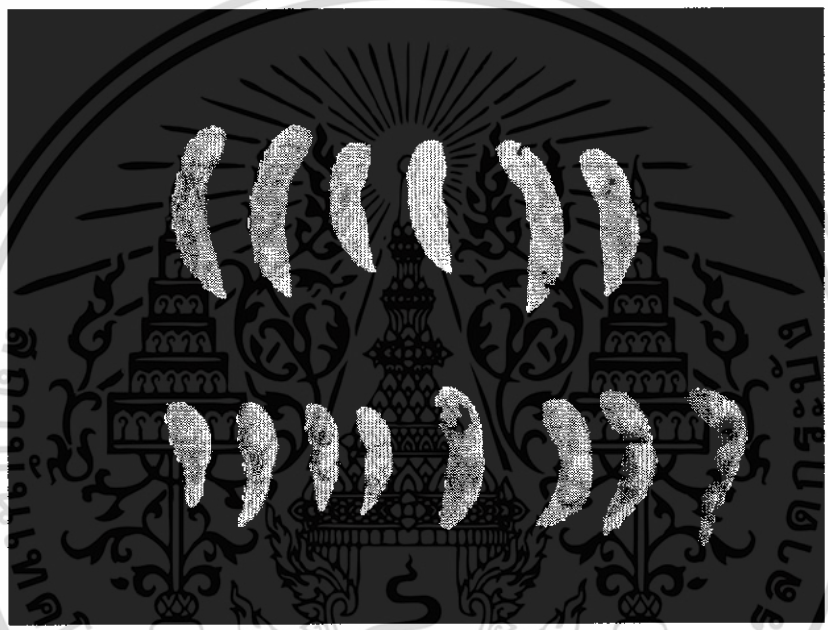
รูปที่ 3.12 จุด Centroid และการวัดระยะห่างจากจุด Centroid กับจุดด่างดำ

### 3.6 รูปร่าง (Shape)

ในการคัดแยกถั่วเหลืองฝักสดรูปร่างก็เป็นส่วนสำคัญส่วนหนึ่ง เพราะถ้าถั่วเหลืองฝักสดมีรูปร่างบิดเบี้ยวโค้งงอมากเกินไปดังรูปที่ 3.13 ก็จะถูกคัดออกก่อนที่จะทำการส่งออก ซึ่งจะมีวิธีการดังต่อไปนี้

#### 3.6.1 การกำหนดเส้นขอบ (Boundary Line)

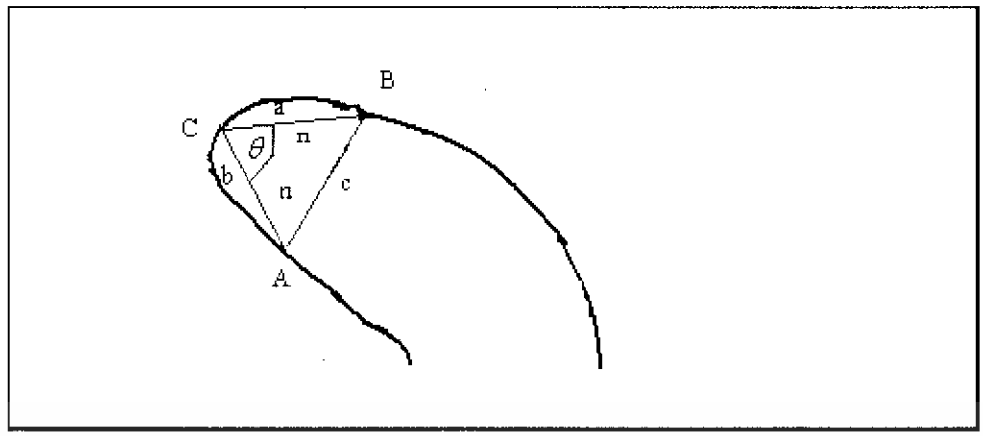
กำหนดเส้นขอบให้ถั่วเหลืองฝักสดแต่ละฝัก ในการกำหนดเส้นขอบของถั่วเหลืองฝักสดจะใช้วิธีการติดตามเส้นขอบ ซึ่งจะทำให้ได้เส้นขอบของถั่วเหลืองฝักสด ดังแสดงในรูปที่ 3.13



รูปที่ 3.13 การกำหนดเส้นขอบของถั่วเหลืองฝักสดโดยวิธีการติดตามเส้นขอบ

#### 3.6.2 การหามุม

เมื่อได้เส้นขอบแล้วก็จะนำพิกัดจุดบนเส้นขอบของถั่วเหลืองฝักสดมาทำการหามุมทุกๆจุด โดยในการพิจารณามุมในแต่ละจุดพิกเซลบนเส้นขอบนั้น จะเว้นระยะของด้านเริ่มต้นมุมและด้านสิ้นสุดมุมตามจำนวน n พิกเซลบนเส้นขอบทั้งสองด้านเท่าๆ กัน ดังแสดงในรูปที่ 3.14




รูปที่ 3.14 การหามุมบนเส้นขอบของถั่วเหลืองฝักสด

จากรูปที่ 3.14 สามารถหาขนาดของด้าน a, b และ c โดยใช้สมการที่ 23 ทำให้ได้ความยาวของด้านเริ่มต้นมุม (ด้าน a) ด้านสิ้นสุดมุม (ด้าน b) และด้านตรงข้ามมุม (ด้าน c) จึงสามารถนำมาหามุมได้ จากการประยุกต์ใช้สมการ 24 ดังต่อไปนี้

$$\theta = \cos^{-1}\left(\frac{a^2 + b^2 - c^2}{2ab}\right) \tag{3.3}$$

เมื่อได้ขนาดของมุมในทุกๆ จุดพิกเซลบนเส้นขอบของถั่วเหลืองฝักสดแล้ว จะพิจารณาว่ามีค่าความแปรปรวนของมุมมากแค่ไหน ถ้ามากกว่าค่า Variance Threshold ที่กำหนดไว้จะถือว่าถั่วเหลืองฝักสดฝักนั้นมีรูปร่างบิดเบี้ยว โดย Variance Threshold ได้มาจากการทดลองแล้วเลือกค่าที่สามารถคัดแยกถั่วเหลืองฝักสดที่มีลักษณะการบิดเบี้ยวออกได้

# บทที่ 4

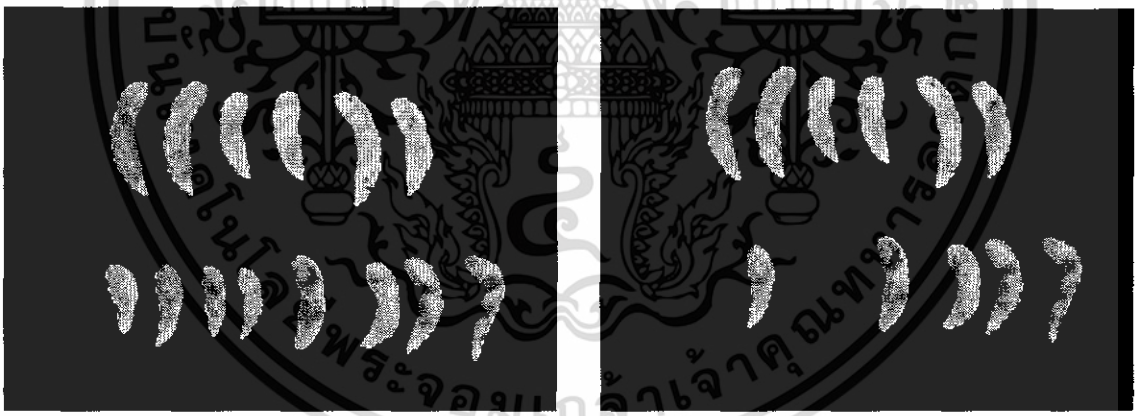
## ผลการดำเนินงาน

เมื่อนำโปรแกรมมาทดลองใช้กับภาพถั่วเหลืองฝักสดได้ผลดังต่อไปนี้

### 4.1 ขนาดพื้นที่ภาพฉาย (Projected area)

ในการใช้โปรแกรมหาขนาดพื้นที่ภาพฉายของถั่วเหลืองฝักสด 39 ฝัก ปรากฏว่าขนาดพื้นที่ภาพฉายที่หาได้โดยวิธีการนับจำนวนพิกเซลนั้นมีขนาดใหญ่กว่าพื้นที่ภาพฉายที่หาได้จากเครื่องมือแพลนนิมิเตอร์ ประมาณ 0.62 $cm^2$ โดยเฉลี่ย ซึ่งเป็นค่าความแตกต่างนี้ไม่มีผลต่อการคัดแยก เนื่องจากค่าต่ำสุดของฝักถั่วมาตรฐานได้มาจากวิธีประมวลผลภาพ ซึ่งเป็นวิธีการวัดที่ใช้กับฝักอื่นเช่นกัน ขนาดพื้นที่ภาพฉายแสดงดังตารางที่ 4.1

เมื่อทราบว่าถั่วเหลืองฝักใดมีขนาดเท่าไหร่ ถ้ามีขนาดเล็กกว่า Size Threshold (Size Threshold = 5.1 $cm^2$) ที่กำหนด ถั่วเหลืองฝักนั้นก็จะถูกคัดแยกออกไปดังรูปที่ 4.1



(a) (b)

รูปที่ 4.1 (a) ภาพถ่ายรูปถั่วเหลืองฝักสดที่มีทั้งแบบสมบูรณ์และไม่สมบูรณ์ (all bright 1)
(b) ถั่วเหลืองฝักสดที่คัดขนาดที่ไม่ต้องการออก

ตารางที่ 4.1 ขนาดพื้นที่ภาพฉายโดยใช้โปรแกรมและแพลนนิมิเตอร์ทั้ง 39 ฝัก(เรียงลำดับจากบนลงล่าง ซ้ายไปขวา)

| ภาพ | ถั่วเหลืองฝักสด ฝักที่ | จำนวน เมล็ด | พื้นที่($cm^2$) | แพลนนิมิเตอร์ ($cm^2$) | ERROR |
|---|---|---|---|---|---|
| **all bright 1** | 1 | 3 | 8.89 | 6.75 | 2.14 |
| | 2 | 3 | 9.26 | 7.75 | 1.51 |
| | 3 | 2 | 6.28 | 4.63 | 1.66 |
| | 4 | 2 | 6.68 | 5.13 | 1.56 |
| | 5 | 3 | 8.77 | 7.00 | 1.77 |
| | 6 | 2 | 7.72 | 6.13 | 1.60 |
| | 7 | 1 | 4.96 | 3.63 | 1.34 |
| | 8 | 1* | 5.65 | 3.88 | 1.78 |
| | 9 | 2 | 4.18 | 3.13 | 1.06 |
| | 10 | 2 | 3.96 | 3.75 | 0.21 |
| | 11 | 2 | 6.96 | 6.25 | 0.71 |
| | 12 | 2 | 7.03 | 4.75 | 2.28 |
| | 13 | 2 | 7.15 | 5.50 | 1.65 |
| | 14 | 3 | 6.48 | 5.25 | 1.23 |
| เฉลี่ย | | | | | **1.46** |
| Standard deviation | | | 1.65 | 1.40 | 0.54 |
| | | | | | |
| **color bright** | 1 | 3 | 8.78 | 8.50 | 0.28 |
| | 2 | 3 | 10.71 | 10.63 | 0.09 |
| | 3 | 2 | 7.80 | 7.63 | 0.18 |
| | 4 | 3 | 8.97 | 9.25 | 0.28 |
| เฉลี่ย | | | | | **0.21** |
| Standard deviation | | | 1.21 | 1.27 | 0.09 |
| | | | | | |

ตารางที่ 4.1 (ต่อ)

| ภาพ | ถั่วเหลืองฝักสด ฝักที่ | จำนวน เมล็ด | พื้นที่($cm^2$) | แพลนนิมิเตอร์ ($cm^2$) | ERROR |
|---|---|---|---|---|---|
| **cook color** | | | | | |
| **bright 1** | 1 | 2 | 6.71 | 6.86 | 0.15 |
| | 2 | 3 | 10.14 | 10.00 | 0.14 |
| | 3 | 2 | 8.05 | 7.86 | 0.19 |
| | 4 | 3 | 9.74 | 9.43 | 0.31 |
| | 5 | 3 | 8.94 | 8.43 | 0.51 |
| | 6 | 3 | 9.84 | 10.00 | 0.16 |
| | 7 | 3 | 7.98 | 8.57 | 0.59 |
| | 8 | 3 | 9.37 | 9.71 | 0.34 |
| | 9 | 2 | 5.30 | 5.14 | 0.16 |
| | 10 | 2 | 8.08 | 8.00 | 0.08 |
| | 11 | 2 | 6.26 | 5.57 | 0.69 |
| | 12 | 2 | 8.21 | 9.00 | 0.79 |
| | 13 | 2 | 7.25 | 7.57 | 0.32 |
| | 14 | 2 | 8.18 | 8.86 | 0.68 |
| | 15 | 3 | 11.15 | 11.43 | 0.28 |
| | 16 | 3 | 9.13 | 8.71 | 0.42 |
| เฉลี่ย | | | | | **0.36** |
| Standard deviation | | | 1.53 | 1.63 | 0.23 |
| | | | | | |
| **color&spot** | | | | | |
| **bright** | 1 | 3 | 8.49 | 8.33 | 0.16 |
| | 2 | 3 | 10.25 | 10.22 | 0.03 |
| | 3 | 2 | 8.17 | 8.00 | 0.17 |
| | 4 | 2 | 7.79 | 7.89 | 0.10 |
| | 5 | 3 | 9.03 | 8.89 | 0.14 |

ตารางที่ 4.1 (ต่อ)

| ภาพ | ถั่วเหลืองฝักสด ฝักที่ | จำนวน เมล็ด | พื้นที่($cm^2$) | แพลนนิมิเตอร์ ($cm^2$) | ERROR |
|---|---|---|---|---|---|
| เฉลี่ย | | | | | 0.12 |
| Standard deviation | | | 0.96 | 0.95 | 0.06 |
| | | | | | |
| **all bright 2** | 1 | 3 | 8.93 | 7.43 | 1.50 |
| | 2 | 3 | 9.27 | 7.43 | 1.84 |
| | 3 | 2 | 6.28 | 5.57 | 0.71 |
| | 4 | 2 | 6.73 | 5.86 | 0.87 |
| | 5 | 3 | 8.85 | 7.43 | 1.42 |
| | 6 | 2 | 7.71 | 7.29 | 0.42 |
| | 7 | 1* | 5.05 | 5.00 | 0.05 |
| | 8 | 1 | 5.73 | 5.43 | 0.30 |
| | 9 | 2 | 4.24 | 3.57 | 0.67 |
| | 10 | 2 | 4.06 | 3.57 | 0.49 |
| | 11 | 2 | 7.18 | 6.86 | 0.32 |
| | 12 | 2 | 7.08 | 6.14 | 0.94 |
| | 13 | 2 | 7.21 | 5.57 | 1.64 |
| | 14 | 3 | 6.53 | 4.71 | 1.82 |
| เฉลี่ย | | | | | 0.93 |
| Standard deviation | | | 1.63 | 1.34 | 0.61 |
| | | | | | |
| **color dark** | 1 | 3 | 8.72 | 8.43 | 0.29 |
| | 2 | 3 | 10.56 | 10.29 | 0.27 |
| | 3 | 2 | 7.60 | 8.43 | 0.83 |
| | 4 | 3 | 8.80 | 8.14 | 0.66 |
| เฉลี่ย | | | | | **0.51** |

ตารางที่ 4.1 (ต่อ)

| ภาพ | ถั่วเหลืองฝักสด ฝักที่ | จำนวน เมล็ด | พื้นที่($cm^2$) | แพลนนิมิเตอร์ ($cm^2$) | ERROR |
|---|---|---|---|---|---|
| Standard deviation | | | 1.22 | 0.99 | 0.27 |
| | | | | | |
| **cook color bright 2** | 1 | 2 | 7.04 | 6.00 | 1.04 |
| | 2 | 3 | 10.55 | 9.43 | 1.12 |
| | 3 | 2 | 8.38 | 6.57 | 1.81 |
| | 4 | 3 | 10.08 | 10.00 | 0.08 |
| | 5 | 3 | 9.35 | 8.71 | 0.64 |
| | 6 | 3 | 10.16 | 9.43 | 0.73 |
| | 7 | 3 | 9.69 | 7.71 | 1.98 |
| | 8 | 3 | 9.49 | 8.86 | 0.63 |
| | 9 | 2 | 5.45 | 4.71 | 0.74 |
| | 10 | 2 | 8.36 | 8.57 | 0.21 |
| | 11 | 2 | 6.42 | 5.00 | 1.42 |
| | 12 | 2 | 8.35 | 7.29 | 1.06 |
| | 13 | 2 | 7.43 | 6.00 | 1.43 |
| | 14 | 2 | 8.47 | 8.00 | 0.47 |
| | 15 | 3 | 11.50 | 10.71 | 0.79 |
| | 16 | 3 | 8.26 | 8.57 | 0.31 |
| เฉลี่ย | | | | | **0.90** |
| Standard deviation | | | 1.59 | 1.78 | 0.55 |
| | | | | | |
| color&spot dark | 1 | 3 | 8.48 | 8.75 | 0.27 |
| | 2 | 3 | 10.31 | 9.75 | 0.56 |

ตารางที่ 4.1 (ต่อ)

| ภาพ | ถั่วเหลืองฝักสด ฝักที่ | จำนวน เมล็ด | พื้นที่($cm^2$) | แพลนนิมิเตอร์ ($cm^2$) | ERROR |
|---|---|---|---|---|---|
| | 3 | 2 | 8.27 | 8.63 | 0.36 |
| | 4 | 2 | 7.89 | 8.13 | 0.24 |
| | 5 | 3 | 9.11 | 9.50 | 0.39 |
| เฉลี่ย | | | | | **0.36** |
| Standard deviation | | | 0.95 | 0.66 | 0.13 |
| | | | | | |
| **all dark** | 1 | 3 | 9.38 | 8.29 | 1.09 |
| | 2 | 3 | 9.83 | 8.57 | 1.26 |
| | 3 | 2 | 6.71 | 5.57 | 1.14 |
| | 4 | 2 | 7.20 | 6.29 | 0.91 |
| | 5 | 3 | 9.57 | 8.86 | 0.71 |
| | 6 | 2 | 8.39 | 7.43 | 0.96 |
| | 7 | 1* | 5.15 | 4.57 | 0.58 |
| | 8 | 1* | 5.88 | 5.00 | 0.88 |
| | 9 | 2 | 4.32 | 4.14 | 0.18 |
| | 10 | 2 | 4.12 | 3.71 | 0.41 |
| | 11 | 2 | 7.12 | 7.14 | 0.02 |
| | 12 | 2 | 7.36 | 6.71 | 0.65 |
| | 13 | 2 | 7.42 | 6.57 | 0.85 |
| | 14 | 3 | 6.72 | 6.00 | 0.72 |
| เฉลี่ย | | | | | **0.74** |
| Standard deviation | | | 1.81 | 1.63 | 0.36 |
| | | | | | |
| cook color dark 1 | 1 | 2 | 6.77 | 6.00 | 0.77 |

ตารางที่ 4.1 (ต่อ)

| ภาพ | ถั่วเหลืองฝักสด ฝักที่ | จำนวน เมล็ด | พื้นที่($cm^2$) | แพลนนิมิเตอร์ ($cm^2$) | ERROR |
|---|---|---|---|---|---|
| | 2 | 3 | 10.30 | 9.00 | 1.30 |
| | 3 | 2 | 8.13 | 7.25 | 0.88 |
| | 4 | 3 | 9.75 | 8.88 | 0.88 |
| | 5 | 3 | 8.85 | 8.38 | 0.48 |
| | 6 | 3 | 9.72 | 8.25 | 1.47 |
| | 7 | 3 | 7.75 | 7.13 | 0.63 |
| | 8 | 3 | 9.04 | 8.38 | 0.66 |
| | 9 | 2 | 5.29 | 4.88 | 0.42 |
| | 10 | 2 | 8.13 | 7.00 | 1.13 |
| | 11 | 2 | 6.23 | 6.25 | 0.02 |
| | 12 | 2 | 8.13 | 7.75 | 0.38 |
| | 13 | 2 | 7.04 | 5.88 | 1.17 |
| | 14 | 2 | 8.08 | 7.25 | 0.83 |
| | 15 | 3 | 10.81 | 9.63 | 1.19 |
| | 16 | 3 | 8.69 | 7.88 | 0.82 |
| เฉลี่ย | | | | | **0.81** |
| Standard deviation | | | 3.33 | 1.29 | 0.38 |
| | | | | | |
| cook color dark 2 | 1 | 2 | 6.60 | 6.43 | 0.17 |
| | 2 | 3 | 10.04 | 10.71 | 0.67 |
| | 3 | 2 | 7.93 | 7.14 | 0.79 |
| | 4 | 3 | 9.52 | 10.29 | 0.77 |
| | 5 | 3 | 8.68 | 8.29 | 0.39 |
| | 6 | 3 | 9.50 | 9.14 | 0.36 |
| | 7 | 3 | 7.59 | 8.00 | 0.41 |

ตารางที่ 4.1 (ต่อ)

| ภาพ | ถั่วเหลืองฝักสด ฝักที่ | จำนวน เมล็ด | พื้นที่($cm^2$) | แพลนนิมิเตอร์ ($cm^2$) | ERROR |
|---|---|---|---|---|---|
| | 8 | 3 | 8.88 | 9.43 | 0.55 |
| | 9 | 2 | 5.20 | 5.57 | 0.37 |
| | 1 | 2 | 7.94 | 8.29 | 0.35 |
| | 11 | 2 | 6.13 | 6.71 | 0.58 |
| | 12 | 2 | 7.92 | 9.00 | 1.08 |
| | 14 | 2 | 7.85 | 7.71 | 0.14 |
| | 15 | 3 | 10.64 | 11.00 | 0.36 |
| | 16 | 3 | 8.55 | 8.71 | 0.16 |
| เฉลี่ย | | | | | **0.46** |
| Standard deviation | | | 1.45 | 1.55 | 0.26 |
| เฉลี่ยรวม | | | | | **0.62** |
| Standard deviation รวม | | | | | **0.53** |

จากภาพที่ 4.1 (a) เมื่อใช้โปรแกรมทำการหาพื้นที่ภาพฉายจะเห็นว่าฝักที่ 7, 9 และ 10 มีขนาดเล็กกว่าค่า Size Threshold ที่กำหนดจึงถูกคัดออกดังรูปที่ 4.1 (b)

**การคำนวณเปอร์เซนต์การคัดขนาด**

$$\text{เปอร์เซนต์การคัดขนาด} = \frac{\text{จำนวนถั่วเหลืองฝักสดที่คัดขนาดได้ตามต้องการ}}{\text{จำนวนถั่วเหลืองฝักสดทั้งหมด}} \times 100$$

$$= \frac{120}{124} \times 100$$

$$= 96.8\ \%$$

## 4.2 สี (Color)

เมื่อทำการคัดแยกโดยใช้ขนาดแล้ว ต่อมาโปรแกรมจะทำการคัดแยกโดยใช้ค่าสี Hue เฉลี่ยในการคัดแยก ซึ่งก็จะได้ผลการดำเนินการดังนี้

### 4.2.1 ค่าสี Hue

ในการหาค่าสีโดยใช้แบบจำลองสี Hue จะได้สีในแต่ละพิกเซลของถั่วเหลืองฝักสดออกมา เมื่อนำค่าสีทุกพิกเซลในถั่วเหลืองฝักสดมาเฉลี่ยกันก็จะได้ค่าสีเฉลี่ยของถั่วเหลืองฝักสดแต่ละฝักออกมา ดังตารางที่ 4.2

ตารางที่ 4.2 ค่าเฉลี่ยสี Hue ของถั่วเหลืองฝักสดทั้ง 39 ฝัก (เรียงลำดับจากบนลงล่าง ซ้ายไปขวา)

| ภาพ | ถั่วเหลืองฝักสดฝักที่ | ค่าเฉลี่ยสี Hue |
|---|---|---|
| all bright 1 | 1 | 0.1917 |
| | 2 | 0.1864 |
| | 3 | 0.1793 |
| | 4 | 0.1763 |
| | 5 | 0.1741 |
| | 6 | 0.1897 |
| | 7 | 0.1864 |
| | 8 | 0.1825 |
| | 9 | 0.1829 |
| | 10 | 0.1675 |
| | 11 | 0.1587 |
| | 12 | 0.1817 |
| | 13 | 0.1846 |
| | 14 | 0.1836 |
| Color bright | 1 | 0.1648 |
| | 2 | 0.1628 |
| | 3 | 0.1936 |
| | 4 | 0.1925 |
| cook color bright 1 | 1* | 0.2045 |
| | 2 | 0.1875 |

ตารางที่ 4.2 (ต่อ)

| ภาพ | ถั่วเหลืองฝักสดฝักที่ | ค่าเฉลี่ยสี Hue |
|---|---|---|
| | 3 | 0.1871 |
| | 4 | 0.1963 |
| | 5 | 0.2195 |
| | 6 | 0.2083 |
| | 7 | 0.2201 |
| | 8 | 0.2201 |
| | 9 | 0.1846 |
| | 10 | 0.1753 |
| | 11 | 0.1585 |
| | 12 | 0.1406 |
| | 13 | 0.1841 |
| | 14 | 0.182 |
| | 15 | 0.1917 |
| | 16 | 0.1957 |
| color&spot bright | 1 | 0.1652 |
| | 2 | 0.1631 |
| | 3 | 0.1778 |
| | 4 | 0.1929 |
| | 5 | 0.1921 |

จากตารางจะเห็นว่าค่าสี Hue ของสีเขียวจะอยู่ในช่วงตั้งแต่ 0.17 ขึ้นไป ฉะนั้นการทดลองนี้จึงตั้งค่า Color Threshold ไว้ที่ 0.17 ดังนั้นถั่วเหลืองฝักสดฝักใดที่มีค่าเฉลี่ยสี Hue ต่ำกว่า Color Threshold ก็จะถูกคัดแยกออกดังรูปที่ 4.2 แต่จากภาพ cook color bright 1 ค่าสี Hue ของสีเขียวจะอยู่ในช่วงตั้งแต่ 0.2 ขึ้นไป จึงตั้งค่า Color Threshold ไว้ที่ 0.2 เนื่องจากการถ่ายภาพในมุมและแสงที่เปลี่ยนไปจึงทำให้ค่าสี Hue เปลี่ยนไปด้วย ต้องทำการกำหนดค่า Color Threshold ที่เป็นเฉพาะของภาพถ่ายแต่ละภาพ

รูปที่ 4.2 ถั่วเหลืองฝักสดที่ถูกคัดแยกโดยค่าเฉลี่ยสี Hue (all bright 1)

**การคำนวณเปอร์เซนต์การคัดแยกสี**

$$\text{เปอร์เซนต์การคัดแยกสี} = \frac{\text{จำนวนถั่วเหลืองฝักสดที่คัดแยกสีได้ตามต้องการ}}{\text{จำนวนถั่วเหลืองฝักสดทั้งหมด}} \times 100$$

$$= \frac{38}{39} \times 100$$

$$= 97.4\ \%$$

**4.2.2 จุดด่างดำ**

ถั่วเหลืองฝักสดบางฝักอาจมีจุดด่างดำจุดเล็กๆ อยู่บนฝักซึ่งก็เป็นสิ่งที่ต้องคัดแยกออกเช่นกัน ในการหาจุดด่างดำนั้นบางจุดโปรแกรมก็สามารถหาเจอ บางจุดโปรแกรมก็หาไม่เจอ เมื่อโปรแกรมหาจุดด่างดำเจอแล้วก็จะทำการหาระยะจากจุดด่างดำถึงจุด Centroid ของแต่ละฝักดังตารางที่ 4.3 ถ้าจุดใดมีขนาดน้อยสุดก็จะถูกโปรแกรมคัดแยกออก ดังรูปที่ 4.3

ตารางที่ 4.3 ระยะจากจุดด่างดำถึงจุด Centroid ของแต่ละฝักจำนวน 39 ฝัก (เรียงลำดับจากบนลงล่าง ซ้ายไปขวา)

| ภาพ | ถั่วเหลืองฝักสดฝักที่ | ระยะจากจุดด่างดำถึงจุด Centroid | | |
|---|---|---|---|---|
| | | จุดที่ 1 | จุดที่ 2 | จุดที่ 3 |
| all bright 1 | 1 | 530.82 | 486.82 | |
| | 2 | 460.84 | 376.25 | |
| | 3 | 409.03 | 270.59 | |
| | 4 | 381.34 | 167.15 | |
| | 5 | 373.98 | 67.17 | |
| | 6* | 425.43 | 143.51 | |
| | 7 | 392.54 | 558.61 | |
| | 8 | 295.98 | 484.25 | |
| | 9* | 198.60 | 403.23 | |
| | 10 | 122.83 | 353.68 | |
| | 11 | 10.94 | 296.29 | |
| | 12 | 160.74 | 290.60 | |
| | 13 | 250.06 | 315.45 | |
| | 14 | 380.33 | 388.22 | |
| Color bright | 1 | 0.00 | | |
| | 2 | 0.00 | | |
| | 3 | 0.00 | | |
| | 4 | 0.00 | | |
| cook color bright 1 | 1 | 838.30 | 887.90 | 894.47 |
| | 2 | 743.79 | 793.43 | 800.01 |
| | 3 | 643.51 | 693.37 | 700.05 |
| | 4 | 565.85 | 615.73 | 622.48 |
| | 5 | 363.54 | 411.47 | 418.23 |
| | 6 | 303.51 | 347.74 | 354.25 |
| | 7 | 258.49 | 292.19 | 297.78 |
| | 8 | 265.64 | 283.93 | 287.88 |
| | 9 | 768.18 | 810.79 | 816.07 |

ตารางที่ 4.3 (ต่อ)

| ภาพ | ถั่วเหลืองฝักสดฝักที่ | ระยะจากจุดด่างดำถึงจุด Centroid | | |
|---|---|---|---|---|
| | | จุดที่ 1 | จุดที่ 2 | จุดที่ 3 |
| | 9 | 768.18 | 810.79 | 816.07 |
| | 10 | 666.80 | 708.93 | 714.15 |
| | 11 | 581.95 | 622.65 | 627.67 |
| | 12 | 507.75 | 547.75 | 552.68 |
| | 13 | 273.52 | 308.77 | 313.16 |
| | 14 | 202.85 | 232.55 | 236.33 |
| | 15 | 123.52 | 137.90 | 140.19 |
| | 16* | 123.30 | 86.89 | 82.21 |
| color&spot bright | 1 | 334.45 | 338.28 | |
| | 2 | 201.05 | 207.77 | |
| | 3 | 56.33 | 82.66 | |
| | 4 | 204.18 | 210.24 | |
| | 5* | 350.28 | 353.89 | |

จากตารางที่ 4.3 ภาพ all bright 1 โปรแกรมสามารถหาจุดด่างดำเจอ 2 จุดด้วยกัน และเมื่อโปรแกรมทำการวัดระยะจากจุดด่างดำถึงจุด Centroid ของถั่วเหลืองฝักสดแต่ละฝักนั้น พบว่าถั่วเหลืองฝักสดฝักที่ 5 และฝักที่ 11 มีระยะน้อยสุดของจุดด่างดำแต่ละจุด ดังนั้นโปรแกรมจึงทำการคัดแยกออกดังรูปที่ 4.3

**การคำนวณเปอร์เซนต์การคัดจุดด่างดำ**

$$\text{เปอร์เซนต์การคัดแยกจุดด่างดำ} = \frac{\text{จำนวนถั่วเหลืองฝักสดที่คัดจุดด่างดำได้ตามต้องการ}}{\text{จำนวนถั่วเหลืองฝักสดทั้งหมด}} \times 100$$

$$= \frac{35}{39} \times 100$$

$$= 89.7\ \%$$

รูปที่ 4.3 ถั่วเหลืองฝักสดที่ถูกคัดแยกโดยจุดด่างดำ (all bright 1)

**4.3 รูปร่าง (Shape)**

เมื่อโปรแกรมทำการหาพิกัดบนเส้นขอบ และมุมของทุกจุดบนเส้นขอบของแต่ละฝักแล้ว ใช้ค่า Variance Threshold ในการกำหนดว่าค่า Variance ที่ได้มีค่าแค่ไหนจึงจะถือว่าถั่วเหลืองฝักสดนั้นมีลักษณะบิดเบี้ยวตามตารางที่ 4.4 ในการทดลองนี้ได้กำหนดค่า Variance Threshold เท่ากับ 550 ในกรณีที่ถั่วเหลืองฝักสดมีค่าความแปรปรวนของมุมบนเส้นขอบมากกว่า 550 โปรแกรมก็จะจำแนกว่าถั่วเหลืองฝักสดฝักนั้นมีลักษณะบิดเบี้ยวมากเกินไปก็จะถูกคัดออก

ตารางที่ 4.4 ค่าVariance ของมุมของเส้นขอบถั่วเหลืองฝักสดแต่ละฝักจำนวน 39 ฝัก (เรียงลำดับจากบนลงล่าง ซ้ายไปขวา)

| ภาพ | ถั่วเหลืองฝักสดฝักที่ | Variance |
|---|---|---|
| all bright 1 | 1 | 488.00 |
| | 2 | 401.00 |
| | 3 | 546.00 |
| | 4 | 479.00 |
| | 5* | 829.00 |
| | 6 | 516.00 |
| | 7* | 562.00 |

ตารางที่ 4.4 (ต่อ)

| ภาพ | ถั่วเหลืองฝักสดฝักที่ | Variance |
|---|---|---|
| | 8* | 572.00 |
| | 9* | 703.00 |
| | 10* | 827.00 |
| | 11 | 449.00 |
| | 12* | 436.00 |
| | 13* | 524.00 |
| | 14 | 585.00 |
| color bright | 1 | 368.00 |
| | 2 | 340.00 |
| | 3* | 560.00 |
| | 4 | 381.00 |
| cook color bright 1 | 1 | 529.00 |
| | 2 | 363.00 |
| | 3 | 375.00 |
| | 4 | 370.00 |
| | 5 | 416.00 |
| | 6 | 315.00 |
| | 7 | 426.00 |
| | 8 | 361.00 |
| | 9* | 699.00 |
| | 10 | 471.00 |
| | 11* | 577.00 |
| | 12 | 344.00 |
| | 13 | 483.00 |
| | 14 | 474.00 |
| | 15 | 308.00 |
| | 16 | 452.00 |
| color&spot bright | 1 | 337.00 |
| | 2 | 366.00 |

ตารางที่ 4.4 (ต่อ)

| ภาพ | ถั่วเหลืองฝักสดฝักที่ | Variance |
|---|---|---|
| | 3 | 296.00 |
| | 4 | 494.00 |
| | 5 | 324.00 |

จากตารางที่ 4.4 จะเห็นว่าในภาพ all bright 1 มีฝักที่มีค่าความแปรปรวนของมุมของเส้นขอบมากกว่าค่า Variance Threshold มีฝักที่ 5, 7, 8, 9, 10 และ ฝักที่ 14 ซึ่งถือว่ามีลักษณะการบิดเบี้ยว ซึ่งเมื่อโปรแกรมทำการประมวลผลแล้วฝักที่มีลักษณะบิดเบี้ยวก็จะถูกคัดแยกออกดังรูปที่ 4.4

รูปที่ 4.4 ถั่วเหลืองฝักสดที่ถูกคัดแยกโดยค่า Variance ของมุมบนเส้นขอบ (all bright 1)

จากรูปที่ 4.4 ถั่วเหลืองฝักสดที่ได้ถูกคัดแยกโดยโปรแกรมการคัดแยกถั่วเหลืองฝักสดสามารถคัดแยกได้ ซึ่งบางฝักที่มีรูปร่างที่บิดเบี้ยวเล็กน้อยค่าVarianceของมุมบนเส้นขอบก็ไม่สามารถคัดแยกได้ และจุดด่างดำบางจุดก็ไม่สามารถคัดแยกได้ จากรูปที่ 4.4 เรียงจากบนลงล่าง ซ้ายไปขวา ฝักที่ 1,2,3 และ 4 เป็นฝักที่สมบูรณ์ ที่สามารถส่งออกได้ ส่วนฝักที่ 5 จุดด่างดำเล็กน้อยซึ่งต้องการคัดออก และฝักที่ 7 และ 8 เป็นฝักที่ลักษณะบิดเบี้ยวแต่โปรแกรมไม่สามารถคัดออกได้

**การคำนวณเปอร์เซนต์การคัดฝักที่รูปร่างผิดปกติออก**

$$\text{เปอร์เซนต์การคัดแยกรูปร่าง} = \frac{\text{จำนวนถั่วเหลืองฝักสดที่คัดรูปร่างได้ตามต้องการ}}{\text{จำนวนถั่วเหลืองฝักสดทั้งหมด}} \times 100$$

$$= \frac{29}{39} \times 100$$

$$= 74.4\%$$

**การคำนวณเปอร์เซนต์การคัดฝักที่ได้ตามต้องการ**

$$\text{เปอร์เซนต์การคัดฝักที่ได้ตามต้องการ} = \frac{\text{จำนวนถั่วเหลืองฝักสดที่คัดได้ตามต้องการ}}{\text{จำนวนถั่วเหลืองฝักสดทั้งหมด}} \times 100$$

$$= \frac{35}{39} \times 100$$

$$= 89.7\%$$

**การคำนวณเปอร์เซนต์การคัดผิดโดยที่คัดฝักที่ต้องการออกไป**

$$\text{เปอร์เซนต์การคัดผิดโดยที่คัดฝักที่ต้องการออกไป} = \frac{\text{จำนวนถั่วเหลืองฝักสดที่คัดฝักที่ต้องการออกไป}}{\text{จำนวนถั่วเหลืองที่ถูกคัดออกทั้งหมด}} \times 100$$

$$= \frac{2}{25} \times 100$$

$$= 8\%$$

**การคำนวณเปอร์เซนต์การคัดผิดโดยที่ฝักที่ไม่ต้องการยังอยู่**

$$\text{เปอร์เซนต์การคัดผิดโดยที่ฝักที่ไม่ต้องการยังอยู่} = \frac{\text{จำนวนถั่วเหลืองฝักสดที่ฝักที่ไม่ต้องการยังอยู่}}{\text{จำนวนถั่วเหลืองฝักสดที่ทำการคัดแล้ว(สมบูรณ์)}} \times 100$$

$$= \frac{3}{14} \times 100$$

$$= 21.4\%$$

# บทที่ 5

# บทสรุปและวิจารณ์

## บทสรุป

จากการที่ได้ทำการศึกษาความเป็นไปได้ในการคัดแยกถั่วเหลืองฝักสดโดยใช้กระบวนการประมวลผลภาพ และได้ทำการเขียนและพัฒนาโปรแกรมที่ใช้ในการคัดแยก ขนาด รูปร่าง และสีของถั่วเหลืองฝักสดนั้น ผลที่ได้เป็นดังนี้คือ ในการคัดแยกขนาดถั่วเหลืองฝักสด สามารถคัดแยกได้โดยใช้ขนาดของพื้นที่ภาพฉายเป็นตัวกำหนด ถ้าฝักใดมีขนาดเล็กกว่าที่ตัวโปรแกรมกำหนดไว้จะถูกคัดออกจากภาพทันที ในการคัดแยกสี โปรแกรมสามารถคัดแยกได้ทั้งสีที่ไม่ต้องการและฝักที่มีจุดด่างดำอยู่บนฝัก และสุดท้ายในเรื่องของรูปร่างไม่ปกติของฝัก โปรแกรมสามารถคัดแยกฝักที่มีความผิดปกติได้ จากภาพตัวอย่างที่ได้นำมาทำการทดลองพบว่า การคัดแยกขนาด รูปร่าง และสีของถั่วเหลืองฝักสด โดยใช้กระบวนการประมวลผลภาพ มีความเป็นไปได้ในการคัดแยกความผิดปกติที่ไม่ต้องการของถั่วเหลืองฝักสด ออกจากส่วนที่ต้องการได้โดยคัดแยกได้ดังนี้

1. คัดขนาดยังได้ค่าสูงกว่าที่วัดได้โดย Planimeter 9.8% และคัดแยกได้ถูกต้อง 96.8 %
2. คัดสีถูกต้อง 97.4 %
3. คัดจุดด่างดำถูกต้อง 89.7 %
4. คัดรูปร่างถูกต้อง 74.4 %

โดยรวมคัดฝักที่ได้ตามต้องการ 89.7% คัดผิดโดยที่คัดฝักที่ต้องการออกไป 8% และคัดฝักที่ไม่ต้องการยังอยู่ 21.4 %

## วิจารณ์

จากการเก็บภาพตัวอย่างที่นำมาทดลองและใช้ศึกษานั้น พบว่ามีการเก็บตัวอย่างภาพโดยการถ่ายรูปซึ่งมีความหลากหลายในการถ่ายเช่น เรื่องแสง พื้นของฉากหลัง มุมที่ทำการถ่าย กระบวนการในการถ่ายรูปนั้นมีผลต่อรูปภาพเป็นอย่างมาก ดังนั้นเพื่อให้ได้ผลที่มีประสิทธิภาพมากขึ้น ควรจะมีมาตรฐานในการถ่ายรูปที่จะใช้ในการทดลอง และอาจใช้ภาพผลผลิตทางการเกษตรอื่น ๆ ที่ต้องการทำการคัดแยกมาทดลอง แต่เนื่องจากเวลาในการดำเนินโครงงานมีจำกัดจึงทำการทดลองกับถั่วเหลืองฝักสดพันธุ์ AGS 292 เพียงอย่างเดียว

*ในส่วนของโปรแกรม* เรื่องการคัดขนาดมีความผิดพลาดจากมาตรฐานที่กำหนดไว้ เพราะรูปภาพจะมีขนาดใหญ่ขึ้นจากการที่กำจัดสัญญาณรบกวนตอนที่ทำการ Segmentation จึงยังทำให้ผลผลิตที่ได้ผิดพลาดไปบ้าง ข้อจำกัดของโปรแกรมใช้ได้กับรูปภาพที่มีส่วนของวัตถุแยกจากส่วนของพื้นหลังอย่างชัดเจน ถ้ารูปใดที่มีการซ้อนทับกันของวัตถุจะทำให้ผลที่ได้มีความผิดพลาดสูง

# เอกสารอ้างอิง


1. สำนักส่งเสริมและจัดการสินค้าเกษตร กรมส่งเสริมการเกษตร. 2548. ถั่วเหลืองฝักสด. [Online]. Available : http://www.ifrpd.ku.ac.th/news/newspaper/oct47/new10.html.
2. ปานมนัส ศิริสมบูรณ์ ,พิมพ์เพ็ญ พรเฉลิมพงศ์ และ ธีรนุต ร่มโพธิ์ภักดิ์. **สมบัติทางกายภาพของถั่วเหลืองฝักสด**. การประชุมครั้งที่ 5 ประจำปี 2547 วันที่ 26-27 เมษายน 2547 ภาควิชาวิศวกรรมเกษตร คณะวิศวกรรมศาตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
3. Poonpat Poonnoy and Ampawon Tansakul. (6 ตุลาคม 2547). Mango Sorter by Machine Vission System ,**วารสารวิจัยพัฒนา มจธ**. ปีที่ 28 ฉบับที่ 1 มกราคม-มีนาคม 2547 หน้า 44-58.
4. Gonzalez ,R.C. and Woods. R.E 1992. **Digital Image Processing** ,Addison-Wesley.
5. พนิตนาฎ ยิ้มแย้ม. 2548. "การวิเคราะห์ลักษณะทางกายภาพของมะม่วงโดยใช้คอมพิวเตอร์วิชั่น." วิทยานิพนธ์หลักสูตรปริญญามหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ บัณฑิตวิทยาลัย, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. (ฉบับร่าง).
6. ร.ท.ศีพัฒน์ นามวัฒน์. 2548. แสง และสีกับการมองเห็นของมนุษย์. [Online]. Available : http://www.rtafa.ac.th/cs/course/cs461/Document/color.doc
7. จันทร์จิรา สินทนะโยธิน. 2545. "การประมวลผลภาพด้วย C++ Builder." **สาร NECTEC**. ฉบับเดือน มกราคม – กุมภาพันธ์ 2545.
8. จันทร์จิรา สินทนะโยธิน. 2545. "การประมวลผลภาพด้วย C++ Builder." **สาร NECTEC**. ฉบับเดือน พฤษภาคม – มิถุนายน 2545.
9. Department of Computer Science, The university of western Australia. 2005. Colour Image Processing. [Online]. Available : http://homepage.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT14/lecture12.html.
10. Rafael C. GonZalez ,Richard E. Woods ,Steven L. Eddins ,R.E 2004. **Digital Image Processing Using Matlab**. Pearson Education.
11. ธนารัตน์ ชลิดาพงศ์, **เอกสารการสอน Image Processing Segmentation II**, คณะเทคโนโลยีสารสนเทศ, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
12. อัจฉรีย์ จันทลักขณา. **หลักสถิติเพื่อการวิเคราะห์ข้อมูลทั่วไป**. กรุงเทพ : มหาลัยเกษตรศาสตร์. 2541.
13. James T.Macclave ,Frank H. Dietrich,II and Terry Sincich. **Statisties**. Seventh edition ,Macclave.Dietrich.Sincich

14. มนัส สังวรศิลป์ ,วรรัตน์ ภัทรอมรกุล. 2543. **คู่มือการใช้ Matlab ฉบับสมบูรณ์**. กรุงเทพ : อินโฟเพรส.

# ภาคผนวก

ก ส่วนของโปรแกรม bean_sorting

ข ส่วนของโปรแกรม bound_edge_v

# ภาคผนวก ก

## ส่วนของโปรแกรมหลัก bean_sorting

```
f  = imread('C:\Documents and Settings\Administrator\Desktop\dao\project\4 Aug 2004\all bright 1.jpg');
f = im2double(f);
fr = f(:, :, 1);
fg = f(:, :, 2);
fb = f(:, :, 3);
% if (r = g = b != 0)
num = 0.5*((fr - fg) + (fr - fb));
den = sqrt((fr-fg).^2 + (fr-fb).*(fg-fb));
theta = acos(num./(den + eps));
H = theta;
H(fb > fg) = 2*pi-H(fb >fg);
H = H/(2*pi);
den(den == 0) = eps;
%imhist(H);
%%%%%%%% compute mean and sd of Huc %%%%%%%%%
%mt = mean(H);
%m = mean(mt);
%st = std(H);
%s = mean(st);

%%%%%%%% segmentaion %%%%%%%%%
mean_bean = 0.2226;
sd_bean = 0.0215;
k = 5;
[M,N] = size(H);
for   r = 1:M
   for  c = 1:N
```

```
      if  abs(H(r,c) - mean_bean) <= (k*sd_bean)
      %if  abs(H(r,c) - 0.2226) <= (0.107)
        T(r,c) = 255;
      else
        T(r,c) = 0;
      end
    end
end

%%%%%%% Postprocessing %%%%%%%%%
se = strel('disk', 5);
%%Tc = imclose(T, se);
Tco = imopen(T, se);

%%%%%%% Region Labeling of Steel Grains %%%%%%
[L,Nu] = bwlabel(Tco,8);
%%%%%%%%%%Count Pixel%%%%%%%%%%%%%%%%%%
blockarea = 1296 %%%%% xxxx pixel : 1 sqcm
%projectarea = 0;
projectarea(Nu) = 0;
count(Nu) = 0;
projectarea = 0;
for   i = 1:Nu
   count(i) = bwarea(L==i)
   projectarea(i) = count(i)/blockarea;
   if (projectarea(i) < 5.1) %%% (size filtering with Th=5.1 sqcm)
      %%% clear pixel in L from value i to 0
      for r = 1:M
         for c = 1:N
            if (L(r,c) == i)
               L(r,c) = 0;
            end
```

```
        end
      end
    end
end
segment = f;
for r = 1:M
   for c = 1:N
      if (L(r,c) == 0)
         segment(r, c, 1)  = 0;
         segment(r, c, 2)  = 0;
         segment(r, c, 3)  = 0;
      end
   end
end
imshow(segment);
projectarea;
%%%  display component
map = [1 0 0; 0 1 0; 0 0 1; 0.8 0 0; 0 0.8 0; 0 0 0.8; 0.6 0 0; 0 0.6 0; 0 0 0.6; 0.4 0 0; 0 0.4 0; 0 0 0.4;
0.2 0 0; 0 0.2 0; 0 0 0.2; 0 0 0];
labelimage = label2rgb(L,map);
%imshow(labelimage);
%%%%%%%  Labeling for color %%%%%%%
segment_2 = Tco;
for r = 1:M
   for c = 1:N
      if (segment(r,c) == 0)
         segment_2(r,c)  = 0;
      else
         segment_2(r,c)  = 1;
      end
   end
end
```

```
[L2,Nu2] = bwlabel(segment_2,8);
%%%%%%%%%%%%%%%%%%%%color%%%%%%%%%%%
counthue = 0;
for    i = 1:Nu2
   counthue(i) = bwarea(L2==i)
end
hue(Nu2) = 0;
for nbean = 1:Nu2
   hue(nbean) = 0;
end
%%%accumulate sum of hue%%%
for r = 1:M
   for c = 1:N
      if (L2(r,c) ~= 0)
         hue(L2(r,c)) = hue(L2(r,c)) +  H(r,c); % accumulate hue value
      end
   end
end
%%%compute hue for each bean%%%
for nbean = 1:Nu2
   avghue(nbean) = hue(nbean)/counthue(nbean);
   if ( avghue(nbean) < 0.17 )%%% (hue filtering with Th=0.17 )
      %%% clear pixel in L from value i to 0
      for r = 1:M
        for c = 1:N
          if (L2(r,c) == nbean)
             L2(r,c) = 0;
          end
        end
      end
   end
end
```

```
% show filtering with color
colImg = segment;
for r = 1:M
   for c = 1:N
      if (L2(r,c) == 0)
         colImg(r, c, 1)  = 0;
         colImg(r, c, 2)  = 0;
         colImg(r, c, 3)  = 0;
      end
   end
end
imshow(colImg);
%%%%%%%%%%%%%%%Centroid point%%%%%%%%%%
segment_22 = segment_2;
for r = 1:M
   for c = 1:N
      if (colImg(r,c) == 0)
         segment_22(r,c)  = 0;
      else
         segment_22(r,c)  = 1;
      end
   end
end
[L22,Nu22] = bwlabel(segment_22,8);
countcent = 0;
for   i = 1:Nu22
   countcent(i) = bwarea(L22==i)
end
row(Nu22) = 0;
col(Nu22) = 0;
%initialize row, col, and hue to be 0 (clear)
for nbean = 1:Nu22
```

```
   row(nbean) = 0;
   col(nbean) = 0;
end
%accumulate sum of row, col%%%
for r = 1:M
   for c = 1:N
      if (L22(r,c) ~= 0)
         row(L22(r,c)) = row(L22(r,c)) + r; % accumulate row
         col(L22(r,c)) = col(L22(r,c)) + c; % accumulate col
      end
   end
end
% compute centroid and average for each bean
%centImg = f;
for nbean = 1:Nu22
   crow(nbean) = row(nbean)/countcent(nbean);
   ccol(nbean) = col(nbean)/countcent(nbean);
   %%% label the centroid with red color
   %for a = -2:2
      %for b = -2:2
         %centImg(uint16(crow(nbean)+a),uint16(ccol(nbean)+b),1) = 1;
         %centImg(uint16(crow(nbean)+a),uint16(ccol(nbean)+b),2) = 0;
         %centImg(uint16(crow(nbean)+a),uint16(ccol(nbean)+b),3) = 0;
         %end
         %end
end
%imshow(centImg);
%%%%%%%%%%%%%%%%%%%%Dark point%%%%%%%%%%
segment_3 = segment_22;
[L3,Nu3] = bwlabel(segment_3,8);
F = imfill(segment_3,'hole');
Diff = F-segment_3;
```

```
%%%%%%%%%%%%%%%%%% Distance %%%%%%%%%%%%%%%%%%
[Di,No] = bwlabel(Diff,8);
if (No ~= 0)
   coutDrk(i) = 0;
   for    i = 1:No
      coutDrk(i) = bwarea(Di==i);
   end
   rowDrk(No) = 0;
   colDrk(No) = 0;
   %initialize row, col to be 0 (clear)
   for nbeanDrk = 1:No
      rowDrk(nbeanDrk) = 0;
      colDrk(nbeanDrk) = 0;
   end
   %accumulate sum of row, col of darkbean
   for r = 1:M
      for c = 1:N
         if (Di(r,c) ~= 0)
            rowDrk(Di(r,c)) = rowDrk(Di(r,c)) + r; % accumulate row
            colDrk(Di(r,c)) = colDrk(Di(r,c)) + c; % accumulate col
         end
      end
   end
   % compute centroid of each dark
   for nbeanDrk = 1:No
      crowDrk(nbeanDrk) = rowDrk(nbeanDrk)/coutDrk(nbeanDrk);
      ccolDrk(nbeanDrk) = colDrk(nbeanDrk)/coutDrk(nbeanDrk);
   end
   %distance dark to centroid each bean
   distzeros(Nu3,No) = zeros;
   mindist(No) = zeros;
   for nbeanDrk = 1:No
```

```
    for nbean = 1:Nu3
      dist(nbeanDrk,nbean) = sqrt((crowDrk(nbeanDrk)-crow(nbean))^2 + (ccolDrk(nbeanDrk)-
ccol(nbean))^2)
    end
    mindist(nbeanDrk) = min(dist(nbeanDrk,:))
    for nbean = 1:Nu3
      if (mindist(nbeanDrk) == dist(nbeanDrk,nbean))
        for r = 1:M
          for c = 1:N
            if (L3(r,c) == nbean)
              L3(r,c) = 0;
            end
          end
        end
      end
    end
  end
  darkImg = colImg;
  for r = 1:M
    for c = 1:N
      if (L3(r,c) == 0)
        darkImg(r, c, 1) = 0;
        darkImg(r, c, 2) = 0;
        darkImg(r, c, 3) = 0;
      end
    end
  end
  imshow(darkImg);
  %%%%%%% Region Labeling of Steel Grains for Contour Following %%%%%%%
  segment_4 = segment_3;
  for r = 1:M
    for c = 1:N
```

```
        if (darkImg(r,c) == 0)
            segment_4(r,c)  = 0;
        else
            segment_4(r,c)  = 1;
        end
      end
   end
adgeImg = darkImg;
else
   segment_4 = segment_3;
   adgeImg = colImg;
end

[L4,Nu4] = bwlabel(segment_4,8);
%%%%%%%%%%%%%%%%%%Contour Following%%%%%%%%%%%%%%%%%%
bx(M*N) = 0;
by(M*N) = 0;
startx = 0;
aa=1;
a=1;
eg =adgeImg;
x=0;
y=0;
while x <= M
   x=x+1;
   while y < N
      y=y+1;
      if (L4(x,y) == a)
         [eg, angle,bx,by,da,db,dc,] = bound_edge_v(a,x,y,L4,eg);
            if (var(angle) > 550)%%%(var filtering with Th=550 )
               %%% clear pixel in L4 from value a to 0
               for r = 1:M
```

```
                for c = 1:N
                  if (L4(r,c) == a)
                    L4(r,c) = 0;
                  end
                end
              end
            end
        a =a+1;
        x=0;
        y=0;
        break;
      end
    end
    y=0;
    if (a==(Nu4+1))
      break;
    end

end
imshow(eg, 'notruesize')
% show filtering with abnormal shape
compImg = adgeImg;
for r = 1:M
  for c = 1:N
    if (L4(r,c) == 0)
      compImg(r, c, 1)  = 0;
      compImg(r, c, 2)  = 0;
      compImg(r, c, 3)  = 0;
    end
  end
end
figure,imshow(compImg);
```

# ภาคผนวก ข

## ส่วนของโปรแกรมย่อย bound_edge_v

```
function [EDGE, bAngle,bx,by,a,b,c] = bound_edge_v(a,x,y,L4,adgeImg)
i = 1;
startx = x;
starty = y;

bx(i) = x;
by(i) = y;

lastx = x;
lasty = y;

x = startx;
y = starty - 1;

while x~= startx | y ~= starty |i<10
    if (L4(x,y) == 0)
     %%%if blackgound turn right%%%
        if (lastx == x & lasty == y+1)
        x = x+1;
        lasty = y;
        elseif (lastx == x-1 & lasty == y)
        y = y+1;
        lastx = x;
        elseif (lastx == x & lasty == y-1)
        x = x-1;
        lasty = y;
        elseif (lastx == x+1 & lasty == y)
        y = y-1;
```

```
        lastx = x;
        end
    elseif (L4(x,y) ==a)
    %%%if bean turn left%%%
    bx(i) =x;
    by(i) =y;
    i = i+1;
        if (lastx ==x & lasty == y-1)
        x = x+1;
        lasty = y;
        elseif (lastx == x-1 & lasty == y)
        y = y-1;
        lastx = x;
        elseif (lastx == x+1 & lasty == y)
        y = y+1;
        lastx = x;
        elseif (lastx == x & lasty == y+1)
        x = x-1;
        lasty = y;
        end
    end
end

img = adgeImg;
 for j = 1:i-1
        img(bx(j), by(j), 1)  = 1;
        img(bx(j), by(j), 2)  = 0;
        img(bx(j), by(j), 3)  = 0;
 end
EDGE = img;
```

```
%%%%angle of bean%%%%%%%%%%%%%%%%%%%%
bAngle(i-1) =0;
rage = 25;
%%%case 1 %%%
for k = rage+1:i-1-rage
   a(k) = sqrt( (bx(k+rage) - bx(k))^2 + (by(k+rage) - by(k))^2 );
   b(k) = sqrt( (bx(k-rage) - bx(k))^2 + (by(k-rage) - by(k))^2 );
   c(k) = sqrt( (bx(k+rage) - bx(k-rage))^2 + (by(k+rage) - by(k-rage))^2 );
   bAngle(k) = ( acos( (a(k)^2 + b(k)^2 - c(k)^2)/(2*a(k)*b(k)) ) )*180/pi;

end

%%% case 2 %%%%
for k = 1:rage
   a(k) = sqrt( (bx(k+rage) - bx(k))^2 + (by(k+rage) - by(k))^2 );
   b(k) = sqrt( (bx(i+k-rage-1) - bx(k))^2 + (by(i+k-rage-1) - by(k))^2 );
   c(k) = sqrt( (bx(k+rage) - bx(i+k-rage-1))^2 + (by(k+rage) - by(i+k-rage-1))^2 );
   bAngle(k) = ( acos( (a(k)^2 + b(k)^2 - c(k)^2)/(2*a(k)*b(k)) ) )*180/pi;
end

%%%case 3%%%%%%
for k = i-rage:i-1
   a(k) = sqrt( (bx(k-i+1+rage) - bx(k))^2 + (by(k-i+1+rage) - by(k))^2 );
   b(k) = sqrt( (bx(k-rage) - bx(k))^2 + (by(k-rage) - by(k))^2 );
   c(k) = sqrt( (bx(k-i+1+rage) - bx(k-rage))^2 + (by(k-i+1+rage) - by(k-rage))^2 );
   bAngle(k) = ( acos( (a(k)^2 + b(k)^2 - c(k)^2)/(2*a(k)*b(k)) ) )*180/pi;
end
```