สำนักหอสมุดกลาง_พระจอมเกล้าลาดกระบัง

เครื่องคิดราคาอาหาร

Food Price Calculator

โดย

นางสาวภัทรพรรณ จันศรีวิริยะ

นายวชิร ปราโมทย์พันธุ์

เลขหมู่.....
เลขทะเบียน 45942
วัน, เดือน, ปี 26 ก.พ. 2546

.b.....
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

วิศวกรรมระบบควบคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2544

ภาควิชาวิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง..เครื่องคิดราคาอาหาร (Food Price Calculator)

ผู้จัดทำ

1. นางสาวภัทรพรรณ จันศรีวิริยะ
2. นายวชิร ปราโมทย์พันธุ์

..... อาจารย์ที่ปรึกษา

(อาจารย์สุมิตร พนาอุดมทรัพย์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องคิดราคาอาหาร

ภัทรพรรณ จันศรีวิริยะ
วชิร ปราโมทย์พันธุ์
อาจารย์สุมิตร พนาอุดมทรัพย์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2544

บทคัดย่อ

ในปริญญานิพนธ์ฉบับนี้ เรียบเรียงขึ้นจากผลงานที่ได้พัฒนาขึ้นมีชื่อว่า เครื่องคิดราคาอาหาร (Food Price Calculator) ซึ่งใช้สำหรับคิดราคาอาหารโดยอัตโนมัติ โดยใช้กระบวนการประมวลผลภาพ (Digital Image Processing) คิดหาขนาดของจานซึ่งจะเป็นตัวแยกแยะราคา โดยจะเริ่มจากใช้กล้องวีดีโอรับภาพถาดอาหารเข้ามาผ่านวีดีโอการ์ดมาในคอมพิวเตอร์เพื่อประมวลผล ซึ่งโปรแกรมจะเริ่มจากแปลงภาพจริงที่รับเข้ามาเป็นภาพไบนารีเพื่อให้ประมวลผลได้ง่ายขึ้น หลังจากนั้นจะทำการกำจัดสัญญาณรบกวน และใช้กระบวนการหาขอบภาพเพื่อหาขอบของจาน ซึ่งจะทำให้คำนวณหาขนาดของจานออกมาได้ และจะทำการคิดราคาโดยจะตั้งราคาให้กับจานแต่ละขนาด หลังจากนั้นจะแสดงผลออกทางจอมอร์นิเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FOOD PRICE CALCULATOR

Pattarapan Jansriviriya

Wachira Pramotpan

Sumit Panaudomsap Advisor

2001

Abstract

This Thesis is an application of Food Price Calculator that use to calculate food 's price by digital image processing and find the size of plate that can separate price by one size for one price.First,the video camera will capture the picture of tray pass the video card to computer in order to process.The program visual basic will change the real picture to binary picture in order to easy process.Then get rid of noise and use the process of finding the edge in order to find edge of plate.Next the finding the edge of plate can tell the size of plate that will calculate food 's price by set price for each size of plate.Finally the food 's price is shown at monitor of computer.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎี	2
บทที่ 3 หลักการทำงานของโปรแกรม	29
บทที่ 4 การทดลองและผลการทดลอง	41
บทที่ 5 บทวิจารณ์และสรุป	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

ภาพที่ 2.1 การแทนสัญญาณอนาลอกด้วยสัญญาณเชิงเลข 2
ภาพที่ 2.2 ฮีสโตแกรมระดับความเข้มของภาพ g(x,y) ที่มีความไม่เสมอของระดับความเข้ม 5
ภาพที่ 2.3 ฮีสโตแกรมระดับความเข้มของภาพอินพุทโดยวิธีการทำเทรชโฮลด์แบบครอบคลุม 7
ภาพที่ 2.4 ฮีสโตแกรมระดับความเข้มของภาพอินพุทโดยวิธีการทำเทรชโฮลด์แบบปรับค่า 7
ภาพที่ 2.5 การแบ่งข้อมูลภาพออกเป็นภาพย่อยๆ และหาค่าเทรชโฮลด์ในแต่ละภาพย่อยๆ 8
ภาพที่ 2.6 แนวคิดในการคำนวณค่าเทรชโฮลด์โดยวิธีพิจารณาจากฮีสโตแกรม 10
ภาพที่ 2.7 ตัวอย่างจุดภาพในเมตริกซ์ 2 มิติ 12
ภาพที่ 2.8 ตัวอย่างการทำไดเลชั่น 13
ภาพที่ 2.9 ตัวอย่างการทำอิรอชั่น 13
ภาพที่ 2.10 ตัวอย่างการทำโคลส์ซิง : (A+B) – B 14
ภาพที่ 2.11 ตัวอย่างการทำโอเพนนิง : (A - B) + B 14
ภาพที่ 2.12 การพิจารณากลุ่มของจุดภาพแกนกับกลุ่มของจุดภาพรอบข้าง 15
ภาพที่ 2.13 การเลือกเติมจุดภาพแกนเป็นจุดภาพดำและภาพผลลัพธ์ 16
ภาพที่ 2.14 ตัวอย่างการกำจัดสัญญาณรบกวนโดยใช้ kFill 17
ภาพที่ 2.15 (ก) – (ง) แสดงขอบภาพชนิดต่างๆ 18
ภาพที่ 2.16 วินโดว์สำหรับการคำนวณหาค่ากราเดียนท์ 20
ภาพที่ 2.17 วินโดว์ขนาด 2x2 สำหรับการคำนวณหาค่ากราเดียนท์ 21
ภาพที่ 2.18 วินโดว์ขนาด 2x2 สำหรับการคำนวณหาค่ากราเดียนท์โดยวิธีของโรเบิร์ต 21
ภาพที่ 2.19 วินโดว์ขนาด 3x3 สำหรับการคำนวณหาค่ากราเดียนท์โดยวิธีของโซเบล 22
ภาพที่ 2.20 วินโดว์ขนาด 3x3 สำหรับการคำนวณหาค่ากราเดียนท์โดยวิธีของพรีเวท 23
ภาพที่ 2.21 แสดงความสัมพันธ์ระหว่างหาขอบภาพโดยใช้อนุพันธ์เชิงเส้นอันดับที่1 กับการหาขอบภาพโดยใช้อนุพันธ์เชิงเส้นอันดับที่ 2 25
ภาพที่ 2.22 (ก) – (ค) แสดงตัวดำเนินการหาขอบภาพโดยวิธีลาปาเซี่ยน 26
ภาพที่ 2.23 แสดงลักษณะการหาขอบเขตของภาพ 27
ภาพที่ 3.1 ภาพหลังจากทำการเพิ่มเครื่องมือ (Add component) 29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

ภาพที่ 3.2 แผนผังการทำงานของโปรแกรมทั้งหมด 30
ภาพที่ 3.3 การแปลงเป็นภาพไบนารี 31
ภาพที่ 3.4 การระบายสีภายนอกขอบถาด 32
ภาพที่ 3.5 การกำจัดสัญญาณรบกวน 33
ภาพที่ 3.6 การหาขอบจาน 33
ภาพที่ 3.7 ตัวอย่างกรณีจานไม่มีจุดหัก 34
ภาพที่ 3.8 ตัวอย่างกรณีจานมีจุดหัก 35
ภาพที่ 3.9 การหาจุดหัก 35
ภาพที่ 3.10 การหารัศมีของจานใบเดียว 36
ภาพที่ 3.11 จานมีขอบซ้อนกับจานอื่นหลายใบ 37
ภาพที่ 3.12 แผนผังการทำงานของการหารัศมีของจานกรณีขอบจานซ้อนกัน 38
ภาพที่ 3.13 การหารัศมีของจานกรณีขอบจานซ้อนกัน 38
ภาพที่ 3.14 การระบายสีจานที่นับแล้ว 39
ภาพที่ 4.1 การรับภาพเข้ามาในโปรแกรม 41
ภาพที่ 4.2 การแปลงภาพเป็นไบนารี 41
ภาพที่ 4.3 การระบายภายนอกขอบถาด 42
ภาพที่ 4.4 ผลของสัญญาณรบกวนที่เกิดจากแสงภายนอก 42
ภาพที่ 4.5 การกำจัดสัญญาณรบกวนเมื่อใช้ถาดที่ไม่สะท้อนแสงและปรับค่าเทรชโฮลด์ 43
ภาพที่ 4.6 ขอบจานของจานชุดแรก 43
ภาพที่ 4.7 ขอบจานของจานชุดที่สอง 44
ภาพที่ 4.8 ขอบจานของจานชุดที่สาม 44
ภาพที่ 4.9 การหาจุดหักของจาน 44
ภาพที่ 4.10 การหารัศมีของจานกรณีจานใบเดียว 45
ภาพที่ 4.11 การจุดศูนย์กลางเพื่อหารัศมีของจานเมื่อช่วงของจุด 3 จุดแคบ 46
ภาพที่ 4.12 การจุดศูนย์กลางเพื่อหารัศมีของจานเมื่อช่วงของจุด 3 จุดกว้าง 46
ภาพที่ 4.13 การจุดศูนย์กลางเพื่อหารัศมีของจานกรณีขอบจานซ้อนกัน 46
ภาพที่ 4.14 การระบายสีจานชุดแรก 47
ภาพที่ 4.15 การระบายสีจานชุดที่สอง 47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

ภาพที่ 4.16 การระบายสีจานชุดที่สาม 47
ภาพที่ 4.17 การทดสอบในการคิดราคาจริงครั้งที่ 1 48
ภาพที่ 4.18 การทดสอบในการคิดราคาจริงครั้งที่ 2 48
ภาพที่ 4.19 การทดสอบในการคิดราคาจริงครั้งที่ 3 49
ภาพที่ 4.20 การทดสอบในการคิดราคาจริงครั้งที่ 4 49
ภาพที่ 4.21 การทดสอบในการคิดราคาจริงครั้งที่ 5 50
ภาพที่ 4.22ตารางแสดงฐานข้อมูลราคาอาหารที่คิด 50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่1
บทนำ

ที่มาของโครงงาน

เนื่องจากในปัจจุบันการคิดราคาอาหารบางครั้งทำให้ยากต่อการคำนวณ ทำให้เสียเวลาในการรอนาน กรณีเช่น ตามโรงอาหารใหญ่ๆ หรือ ร้านอาหารฟาสต์ฟูด ทั่วๆไป ซึ่งในการคิดราคาอาหารแต่ละครั้งต้องแยกคิดราคาอาหารแต่ละจานแล้วนำมาคิดรวมกัน ทำให้เสียเวลามาก ถ้ายิ่งแต่ละคนซื้ออาหารเป็นจำนวนมากๆทำให้คนที่รอในการคิดราคาต้องรอนาน วิธีแก้ไขก็อาจต้องจ้างพนักงานเพิ่มและต้องซื้อเครื่องมือในการรองรับลูกค้าจำนวนมาก ทำให้เสียค่าใช้จ่ายสูงในระยะยาว ดังนั้นถ้าสามารถคิดราคาอาหารได้รวดเร็วโดยคิดได้ครั้งมากๆ ก็ทำให้ประหยัดเวลาและต้นทุนได้

เครื่องคิดราคาอาหารใช้สำหรับคำนวณราคาอาหารจากภาพของจานในถาดซึ่งจานที่ใช้ในการทำจะเป็นจานชนิดกลม โดยรับสัญญาณภาพมาจากกล้องวีดีโอ ซึ่งเป็นสัญญาณอนาลอกและแปลงเป็นสัญญาณดิจิตอล โดยผ่านวีดีโอการ์ด แล้วนำมาประมวลผลผ่านทางคอมพิวเตอร์โดยใช้โปรแกรมวิชวลเบสิก มาเป็นตัวควบคุมการทำงาน โดยจะนำภาพมาเก็บในmemory(array)ของโปรแกรม และทำการประมวลผลโดยใช้ทฤษฎีการประมวลผลภาพ เช่น การแปลงภาพเป็นไบนารี การกำจัดสัญญาณรบกวน การหาขอบภาพ การแยกวัตถุออกจากภาพมาเป็นตัวแยกขนาดและจำนวนของจานออกจากภาพ

วัตถุประสงค์

1) เพื่อศึกษาแนวทางในการพัฒนาโปรแกรมเพื่อคิดราคาอาหารให้มีความรวดเร็วถูกต้อง โดยใช้หลักกการการประมวลผลภาพเป็นพื้นฐานในการพัฒนาโปรแกรม
2) พัฒนาต้นแบบโครงงานคิดราคาอาหารตามที่ได้ศึกษาไว้
3) เพื่อให้สามารถเครื่องคิดราคาอาหารสามารถใช้งานได้จริง

ขอบเขต

1. อาหารจะต้องไม่เลยขอบจาน
2. กล้องวีดีโอที่รับภาพจะถูก fix ตำแหน่งเพื่อให้ภาพมีขนาดเท่ากัน

ประโยชน์ที่คาดว่าจะได้รับ

เครื่องคิดราคาอาหารที่พัฒนาขึ้นสามารถนำไปใช้งานได้ มีความถูกต้องและรวดเร็วทำให้เกิดความสะดวก ประหยัดเวลาในการคิดราคาอาหาร รวมทั้งผู้จัดทำได้รับความรู้และมีประสบการณ์ในการทำโครงงานครั้งนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่2
ทฤษฎีหรือหลักการ

2.1 การประมวลผลภาพ

การเกิดภาพเป็นการเปลี่ยนข้อมูลของวัตถุจาก 3 มิติเป็นข้อมูลภาพ 2 มิติ ข้อมูลภาพจึงเป็นค่าฟังก์ชัน 2 มิติของความเข้มแสง คือ f(x,y) ดังนั้นค่าของฟังก์ชันที่พิกัดสปาเทียล(Spatial) (x,y) จะแสดงความเข้มของภาพ ณ พิกัดนั้น แต่เนื่องจากภาพที่ได้จากกล้องวีดีโอเป็นภาพแบบอนาลอกโดยค่าความเข้ม f(x,y) จะมีความต่อเนื่อง ดังนั้นในการที่จะนำภาพมาประมวลผลโดยคอมพิวเตอร์จึงจำเป็นต้องทำให้ภาพลักษณะต่อเนื่องกลายเป็นภาพ Digital หรือภาพเชิงตัวเลขเสียก่อน โดยการทำดิจิไทเซชั่น(Digitization) ซึ่งเป็นการแปลงฟังก์ชันต่อเนื่อง f(x,y) ให้เป็นฟังก์ชันไม่ต่อเนื่อง g(x,y) โดยการแบ่ง f(x,y) เป็นช่วงๆ (Quantized) ซึ่งผลลัพธ์ที่ได้ก็คือ ค่าความเข้มของจุดภาพที่มีอยู่ L ระดับรวมทั้งค่าพิกัด(x,y) ซึ่งค่าเหล่านี้จะเป็นจำนวนเต็มที่ไม่เป็นเลย โดยทั่วไปจุดภาพแต่ละจุดจะเป็นสมาชิกของเมตริกซ์ที่มีขนาด M แถว N คอลัมน์ (MxN) เพราะฉะนั้น x,y จะมีค่าอยู่ในช่วง(1<=y<=M,1<=x<=N) และจำนวนช่วงระดับความเข้มของจุดภาพ L จะบ่งบอกถึงความละเอียดของภาพเชิงตัวเลข (Digital Image) โดยทั่วไปแล้ว L จะมีระดับความเข้มเท่ากับ 256 ระดับ ซึ่งจะทำให้ค่าระดับความเข้มของจุดภาพอยู่ในช่วง (0-255) โดยใช้ 8 บิท (2^8 = 256) สำหรับเก็บข้อมูลภาพในแต่ละจุด ในกรณีที่ต้องการภาพที่มีความละเอียดของระดับความเข้มสูงๆ L อาจต้องการจำนวนบิทสำหรับเก็บข้อมูลมากกว่า 8 บิท คือ อาจเป็น 16 หรือ 24 บิท โดยค่าระดับความเข้มของจุดภาพจะเท่ากับ 2^{16} หรือ 2^{24} บิท ตามลำดับ ในภาพที่2.1 แสดงถึงการแทนสัญญาณอนาลอกด้วยสัญญาณเชิงเลข

ภาพที่2.1 การแทนสัญญาณอนาลอกด้วยสัญญาณเชิงเลข

จากการนำภาพเชิงตัวเลขไปใช้ในการประมวลผลในรูปแบบต่าง ๆ มากมายนั้น ทำให้สามารถที่จะแบ่งรูปแบบการประมวลผลภาพเหล่านั้นออกเป็น 2 ระดับด้วยกัน คือ การประมวลผลภาพในระดับต่ำ (Low-level Image Processing) และการประมวลผลภาพในระดับสูง (High-level Image Processing)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประมวลผลภาพในระดับต่ำ เป็นการประมวลผลเชิงตัวเลขเกือบทั้งหมดเพื่อหาตัวแปรต่าง ๆ มาอธิบายข้อมูลภาพและมีจุดประสงค์ที่จะนำตัวแปรเหล่านี้ไปใช้ในการประมวลผลภาพในระดับสูงต่อไป ซึ่งการประมวลผลภาพในระดับสูง คือ การทำให้คอมพิวเตอร์รู้จักและเข้าใจภาพได้ เช่น การจดจำรูปแบบของตัวเลขสำหรับการประมวลภาพในระดับต่ำ โดยทั่วไปประกอบไปด้วย การประมวลผลภาพก่อน , การกำจัดสัญญาณรบกวน , การทำให้ภาพคมชัด , การหาขอบภาพ , การแบ่งแยกวัตถุภายในภาพ , การจำแนกภาพ , การสร้างภาพไบนารี เป็นต้น

ความแตกต่างของการประมวลผลภาพในระดับต่ำและระดับสูง คือ ข้อมูลที่นำมาใช้ในการประมวลผลภาพ ซึ่งการประมวลผลในระดับต่ำจะใช้ค่าความสว่างหรือระดับความเข้มของจุดภาพโดยตรง ส่วนการประมวลภาพในระดับสูงข้อมูลที่นำมาประมวลผลจะถูกแสดงในรูปของสัญลักษณ์ โดยสัญลักษณ์เหล่านี้จะแสดงถึงสิ่งต่าง ๆ ที่อยู่ในภาพ และใช้ตัวแปรที่ได้จากประมวลผลในระดับต่ำมาอธิบายถึงสัญลักษณ์เหล่านี้ ซึ่งจะเห็นได้ว่าการประมวลผลภาพในระดับต่ำนั้นมีความสำคัญมากสำหรับที่จะทำให้คอมพิวเตอร์สามารถเข้าใจและรู้จักภาพได้

การประมวลผลภาพเชิงตัวเลข

การประมวลผลภาพเชิงตัวเลข หมายถึง การนำภาพที่พบทั่วไปมาประมวลผลด้วยเครื่องคอมพิวเตอร์โดยภาพที่นำมาประมวลผลด้วยเครื่องคอมพิวเตอร์นี้จะถูกแทนที่ด้วยตัวเลขให้อยู่ในรูปของเมตริกซ์ แต่ภาพที่ได้โดยส่วนมากแล้วจะเป็นภาพที่ได้จากตัวรับสัญญาณซึ่งอยู่ในรูปของฟังก์ชัน f(x,y) ที่ต่อเนื่องในระนาบ 2 มิติ (แกน x และแกน y) โดยจะเป็นสัดส่วนกับความสว่างหรือความเข้มของภาพที่ตำแหน่ง (x,y) ซึ่งเรียกว่าระดับสีเทา (Gray Level)

การแทนภาพด้วยข้อมูลแบบดิจิตอล

ภาพข้อมูลแบบดิจิตอล เป็นภาพที่ถูกแปลงมาจากอนาลอกอยู่ในรูปของตัวเลขโดยภาพอนาลอกถูกแบ่งเป็นพื้นที่สี่เหลี่ยมเล็กๆ เรียกว่า พิกเซล (Pixel) ในแต่ละพิกเซลจะถูกระบุตำแหน่งโดย (x,y) และค่าระดับสีเทาของพิกเซล โดยเราสามารถแปลงภาพเป็นข้อมูลแบบดิจิตอลได้ซึ่งมีขั้นตอนและวิธีการดังนี้

เมื่อเรานำสัญญาณอนาลอกที่ต้องการประมวลผลมาผ่านส่วนที่เรียกว่า ดิจิไทเซอร์ ซึ่งมีหน้าที่เปลี่ยนสัญญาณอนาลอกให้เป็นสัญญาณดิจิตอล จากนั้นทำการควอนไทซ์เพื่อที่จะประมวลผลสัญญาณด้วยระบบคอมพิวเตอร์ ฟังก์ชันของภาพ f(x,y) จะถูกทำให้เป็นสัญญาณที่ไม่ต่อเนื่องทั้งระนาบของภาพ ซึ่งเราเรียกว่า การสุ่มภาพของฟังก์ชันที่เรียกได้ว่า การควอนไทเซชั่นระดับสีเทา (Gray Level Quantization) ก็จะได้ข้อมูลที่เป็นดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1 **การสร้างภาพไบนารี**

อุปกรณ์ที่มีความสามารถในการแสดงผลได้แค่ 2 ระดับหรือ 2 สี คือสีขาวกับสีดำ ยังมีการใช้กันอย่างแพร่หลาย เช่น เครื่องพิมพ์ , เครื่องโทรสาร , จอภาพแสดงผลแบบโมโนโครม เป็นต้น เนื่องจากอุปกรณ์เหล่านี้เป็นอุปกรณ์ที่มีราคาถูกดังนั้นการที่จะแสดงผลหรือพิมพ์รูปภาพที่มีระดับความเข้มของภาพหลายระดับซึ่งมีมากกว่าความสามารถในการแสดงผลของอุปกรณ์เหล่านั้นที่มีแค่เพียง 2 ระดับเท่านั้น

จะเห็นได้ว่าการที่จะแก้ปัญหาการแสดงผลภาพที่มีความเข้มหลายระดับบนอุปกรณ์ที่สามารถแสดงผลได้ 2 ระดับนั้น จะต้องทำการแปลงข้อมูลภาพให้เป็นภาพไบนารีก่อน ซึ่งการสร้างภาพไบนารี หมายถึง การแปลงข้อมูลภาพที่มีความเข้มหลายระดับให้เป็นภาพที่มีระดับความเข้มเพียง 2 ระดับ นั้นคือ 1 จุดภาพมีได้แค่ 2 ค่าเท่านั้นคือ 0 กับ 1 โดยจุดภาพที่แทนด้วย 1 จะหมายถึงจุดภาพที่มีสีดำ ส่วนจุดภาพที่แทนด้วย 0 จะหมายถึงจุดภาพที่มีสีขาว เมื่อทำการแปลงภาพเป็นภาพไบนารีแล้วจึงนำภาพนั้นไปแสดงผลบนอุปกรณ์ จะเห็นได้ว่าการแปลงข้อมูลภาพหลายระดับเป็นภาพไบนารีมีความจำเป็นและมีประโยชน์มากในการแสดงผลภาพที่มีระดับความเข้มของภาพหลายระดับความเข้มของภาพหลายระดับบนอุปกรณ์ที่มีความสามารถในการแสดงผลได้ 2 ระดับ สำหรับประโยชน์อีกประการหนึ่งในการแปลงข้อมูลภาพเป็นไบนารีแล้วสามารถลดลงได้ถึง 8 เท่านั่นคือ 1 จุดภาพจะใช้เนื้อที่ในการเก็บ 1 บิท อีกทั้งยังสามารถนำไปประยุกต์ใช้งานได้อย่างแพร่หลาย

2.1.1.1การทำเทรชโฮลด์ (Thresholding)

การทำเทรชโฮลด์ถือว่าเป็นเทคนิคที่สำคัญในการประมวลผลภาพในส่วนของการเซกเมนต์ภาพ ซึ่งจุดประสงค์ของการเซกเมนต์ภาพ คือ การแยกองค์ประกอบของภาพออกไปเป็นส่วนประกอบย่อย ๆ ที่มีสัมพันธ์กันทางกายภาพของภาพนั้น และส่วนประกอบที่ถูกแยกออกมานั้นอาจถูกนำไปประมวลผลในส่วนอื่นต่อไป ซึ่งการเซกเมนต์ภาพจะมีหลักการทำงานในแนวเดียวกันกับสายตาของคน คือ สามารถแยกลักษณะเด่นออกมาจากภาพที่มองเห็นได้ และเทคนิคการทำเทรชโฮลด์ (Thresholding technique) ถือว่าเป็นเทคนิคในการแบ่งแยกองค์ประกอบภาพที่ง่ายเทคนิคหนึ่ง โดยที่ระดับความเข้มหนึ่งสามารถที่จะแบ่งกลุ่มของจุดภาพออกเป็น 2 กลุ่มอย่างชัดเจน คือ กลุ่มของวัตถุจะมีระดับความเข้มของภาพ g (x,y) ค่อนข้างต่ำกับกลุ่มของส่วนพื้นหลังจะมีระดับความเข้มของภาพ g (x,y) ค่อนข้างสูง ดังเช่นภาพที่ 2.2 ซึ่งแสดงฮีสโตแกรมระดับความเข้มของภาพ ของภาพออกไปเป็นส่วนประกอบย่อย ๆ g (x,y) ที่มีองค์ประกอบส่วนที่เป็นวัตถุค่อนข้างมืดบนส่วนพื้นหลังที่สว่าง ดังนั้นการแยกกลุ่มทั้งสองออกจากกันอย่างชัดเจนสามารถทำได้โดยเลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

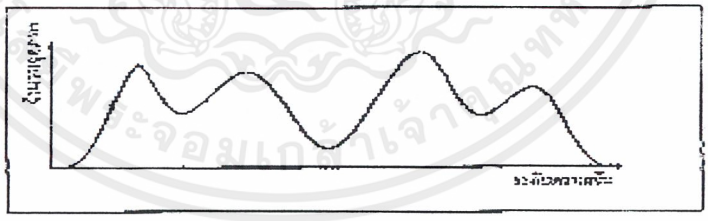
ค่าเทรชโฮลด์ที่มีค่าระดับความเข้มอยู่ระหว่างกลุ่มทั้งสองบนระดับความเข้มของภาพ และทำการตรวจสอบแต่ละจุดภาพ ถ้าค่า g (x,y) น้อยกว่าค่าเทรชโฮลด์ (g (x,y) $<$ Thr) ถือว่าเป็นจุดภาพของวัตถุ และถ้ามากกว่าหรือเท่ากับค่าเทรชโฮลด์ (g(x,y)$\geq$Thr) ถือว่าเป็นจุดภาพของส่วนพื้นหลัง ดังนั้นข้อมูลภาพ g(x,y) ที่ผ่านการทำเทรชโฮลด์สามารถนิยามด้วยสมการดังต่อไปนี้

$$g_{thr}(x,y) = \begin{cases} 1 \text{ if } g_{thr}(x,y) < Thr \\ 0 \text{ if } g_{thr}(x,y) \geq Thr \end{cases} \qquad (2.1)$$

โดยที่

$g_{thr}(x,y)$	ข้อมูลภาพผลลัพธ์เป็นภาพไบนารี
g(x,y)	ข้อมูลภาพอินพุทที่มีระดับความเข้ม 0 ถึง L ระดับ
Thr	ค่าเทรชโฮลด์เป็นค่าคงที่ที่มีอยู่ระหว่าง 0 ถึง L ระดับ
1	จุดดำ
0	จุดขาว
L	ระดับความเข้มของจุดภาพสูงสุด

จุดภาพที่นิยามด้วย 1 คือ จุดภาพของส่วนที่เป็นวัตถุ และจุดภาพที่นิยามด้วย 0 จะเป็นจุดภาพของพื้นหลังของภาพ



ภาพที่ 2.2 ฮีสโตแกรมระดับความเข้มของภาพ g(x,y) ที่มีความไม่สม่ำเสมอของระดับความเข้มในองค์ประกอบของภาพ

การเซกเมนต์ภาพโดยใช้เทคนิคการทำเทรชโฮลด์เพื่อให้ได้ภาพผลลัพธ์ที่มีความถูกต้องเหมาะสมนั้นสิ่งสำคัญที่สุด คือ ค่าเทรชโฮลด์ที่ใช้ เนื่องจากถ้าเลือกค่าเทรชโฮลด์ที่ไม่มีเหมาะสมภาพผลลัพธ์ที่ได้อาจไม่ถูกต้อง ดังนั้นปัญหาของการเซกเมนต์ภาพโดยวิธีการทำเทรชโฮลด์นี้ก็คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำอย่างไรจึงจะสามารถคำนวณหาค่าเทรชโฮลด์หลายวิธี โดยแต่ละวิธีอาจจะเหมาะสมกับภาพเฉพาะอย่างยิ่งที่แตกต่างกันออกไป

2.1.1.2 รูปแบบในการทำการเทรชโฮลด์

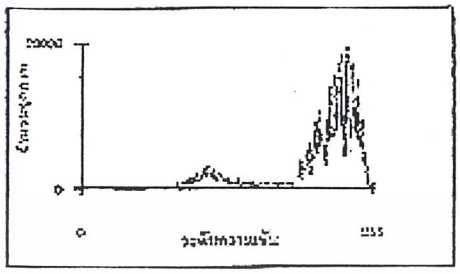
ส่วนรูปแบบในการ ทำการเทรชโฮลด์กับจุดภาพนั้น ภาพที่จุดภาพของส่วนที่เป็นวัตถุกับจุดภาพของส่วนพื้นหลังในแต่ละส่วนมีระดับความเข้มมีความสม่ำเสมอ และระดับความเข้มระหว่างทั้ง 2 ส่วนแตกต่างกันอย่างชัดเจนทั่วทั้งภาพ ซึ่งภาพลักษณะดังกล่าวนี้สามารถใช้ค่าเทรชโฮลด์แบบครอบคลุม (Global Thresholding) แต่ถ้าภาพที่มีระดับความเข้มไม่สม่ำเสมอเกิดขึ้นในส่วนของวัตถุหรือส่วนของวัตถุหรือส่วนของพื้นหลังหรือทั้งสองส่วนค่าโกบอลเทรชโฮลด์เพียงค่าเดียวไม่เหมาะสมสำหรับการทำเทรชโฮลด์กับแต่ละจุดภาพทั่วทั้งภาพ ค่าเทรชโฮลด์ที่ดีควรมีการปรับเปลี่ยนค่าตามตำแหน่งของจุดภาพนั้น คือ การใช้ค่าเทรชโฮลด์ที่ต่างกันสำหรับจุดภาพที่ตำแหน่งต่างกันและเรียกการทำเทรชโฮลด์ในลักษณะดังกล่าวว่า การทำเทรชโฮลด์แบบปรับค่า(Adaptive Thresholding)

2.1.1.3 การทำเทรชโฮลด์แบบครอบคลุม

สำหรับขั้นตอนการหาค่าเทรชโฮลด์ที่ครอบคลุมภาพทั้งภาพโดยอัตโนมัติโดยทั่วไปแล้วจะมีพื้นฐานของการดำเนินการอยู่บนฮีสโตแกรมระดับความเข้มของจุดภาพ ซึ่งฮีสโตแกรมระดับความเข้มสามารถคำนวณจากจำนวนจุดภาพในแต่ละระดับของความเข้มของภาพออก และทำการหาค่าเทรชโฮลด์ในรูปแบบต่างๆ ที่สามารถแบ่งฮีสโตแกรมระดับของความเข้มของภาพออกเป็น 2 ส่วนได้อย่างถูกต้อง คือ ส่วนที่เป็นระดับความเข้มของวัตถุกับส่วนที่เป็นระดับความเข้มของพื้นหลัง ตัวอย่างเช่น กรณีของภาพที่มีอัตราความแตกต่างของระดับความเข้มระหว่างส่วนที่เป็นส่วนวัตถุกับส่วนของพื้นหลังสูง และระดับความเข้มที่เกิดขึ้นในแต่ละส่วนมีความสม่ำเสมอ กรณีนี้สามารถใช้ระดับความเข้มที่มีจำนวนของจุดภาพที่ต่ำที่สุดและอยู่ระหว่างกลุ่มระดับความเข้มทั้ง 2 กลุ่มบนฮีสโตแกรมเป็นค่าเทรชโฮลด์ หรือ ในกรณีทั่วๆ ไป ค่าเทรชโฮลด์อาจจะพิจารณาจากค่าระดับความเข้มที่สามารถแบ่งฮีสโตแกรมออกเป็น 2 กลุ่ม แล้วทำให้ความแปรปรวนที่เกิดขึ้นระหว่างกลุ่มมีค่ามากที่สุด และความแปรปรวนที่เกิดขึ้นภายในกลุ่มที่มีค่าต่ำสุด หลังจากนั้นนำค่าเทรชโฮลด์ที่คำนวณหาได้ไปทำเทรชโฮลด์กับแต่ละจุดภาพทั่วทั้งภาพเพื่อให้ได้ภาพผลลัพธ์ ภาพฮีสโตแกรมระดับความเข้มของภาพอินพุทแสดงในภาพที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

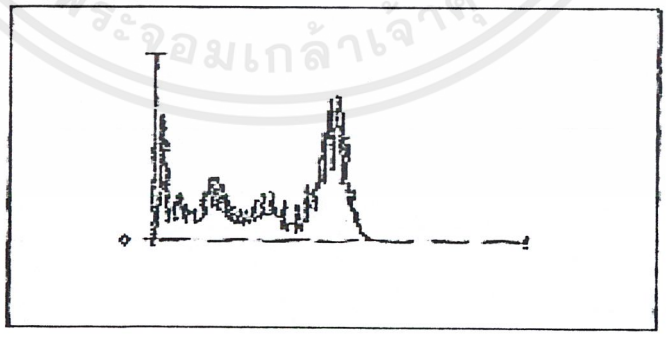
เทรชโฮลด์ที่คำนวณหาได้ไปทำเทรชโฮลด์กับแต่ละจุดภาพทั่วทั้งภาพเพื่อให้ได้ภาพผลลัพธ์ ภาพฮีสโตแกรมระดับความเข้มของภาพอินพุทแสดงในภาพที่ 2.3



ภาพที่ 2.3 ฮีสโตแกรมระดับความเข้มของภาพอินพุทโดยวิธีการทำเทรชโฮลด์แบบครอบคลุม

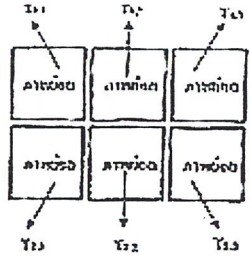
2.1.1.4 การทำเทรชโฮลด์แบบปรับค่า

กรณีที่ข้อมูลภาพมีความไม่สม่ำเสมอเกิดขึ้นในส่วนของวัตถุหรือส่วนของพื้นหลังหรือทั้งสองส่วน ซึ่งภาพในลักษณะเช่นนี้ ฮีสโตแกรมระดับความเข้มของภาพจะมีลักษณะดังเช่นภาพที่ 2.4 การใช้ค่าเธรดโฮลด์แบบครอบคลุมเพียงค่าเดียวกับภาพทั้งภาพอาจทำให้ภาพผลลัพธ์ที่ได้ไม่ถูกต้อง จากปัญหาที่เกิดขึ้นสามารถแก้ไขได้โดยการแบ่งข้อมูลภาพทั้งภาพออกเป็นภาพย่อยๆที่แสดงดังภาพที่ 2.5 และแต่ละภาพย่อยก็จะมีการหาค่าเทรชโฮลด์ในรูปแบบที่กำหนด เพื่อให้ได้ค่าเธรดโฮลด์ที่เหมาะสมสำหรับภาพย่อยนั้นและใช้ค่าเทรชโฮลด์นั้นทำเซกเมนเตชั่นกับแต่ละภาพย่อย ขั้นตอนสุดท้ายนำแต่ละภาพย่อยที่ผ่านการเซกเมนเตชั่นมารวมกันตามพิกัดเดิม



ภาพที่ 2.4 ฮีสโตแกรมระดับความเข้มของภาพอินพุทโดยการทำเทรชโฮลด์แบบปรับค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.5 การแบ่งข้อมูลภาพออกเป็นภาพย่อยๆ และหาค่าเทรชโฮลด์ในแต่ละภาพย่อยๆ

2.1.1.5 วิธีการหาค่าเทรชโฮลด์

ขั้นตอนในการทำเซกเมนต์ภาพโดยใช้เทคนิคเทรชโฮลด์เพื่อให้ได้ผลลัพธ์ที่เหมาะสมและคมชัดนั้น สิ่งสำคัญที่สุดคือ ค่าเทรชโฮลด์ เนื่องจากถ้าเลือกค่าเทรชโฮลด์ที่ไม่เหมาะสม(ค่าเทรชโฮลด์มีค่ามากหรือน้อยเกินไป) หรือภาพที่ได้มีสิ่งรบกวนเกิดขึ้น อันเป็นผลทำให้ภาพผลลัพธ์ที่ได้ไม่สวยงามเท่าที่ควร ดังนั้นปัญหาของการสร้างภาพไบนารี่โดยวิธีเทรชโฮลด์นี้ก็คือ ทำอย่างไรจึงจะสามารถคำนวณหาค่าเทรชโฮลด์ที่เหมาะสมสำหรับแต่ละภาพที่จะนำมาทำการเซกเมนต์ ได้มีผู้เสนอวิธีการในการคำนวณหาค่าเทรชโฮลด์หลายวิธี โดยแต่ละวิธีเหมาะกับลักษณะการทำงานที่แตกต่างกันไป เช่น การหาค่าเทรชโฮลด์โดยการกำหนดค่าล่วงหน้า(Preassigned Threshold value) การหาค่าเทรชโฮลด์จากค่ากลาง (Mid-Range Threshold Value) และการหาค่าเทรชโฮลด์โดยพิจารณาจากฮิสโตแกรม(Histogram Threshold Value) โดยที่แต่ละวิธีสามารถอธิบายได้ดังนี้

2.1.1.6 การหาค่าเทรชโฮลด์โดยการกำหนดค่าล่วงหน้า

การหาค่าเทรชโฮลด์โดยการกำหนดค่าล่วงหน้า เป็นวิธีที่ง่ายที่สุด เป็นวิธีการคำนวณค่าเทรชโฮลด์โดยการกำหนดเองจากผู้ใช้ซึ่งการกำหนดนี้ขึ้นอยู่กับประสบการณ์ของผู้ใช้นั้นๆ โดยการเลือกค่าคงที่ค่าหนึ่ง ซึ่งเรียกค่านั้นว่า ค่าเทรชโฮลด์ โดยค่าที่เลือกมานี้จะเป็นค่าที่อยู่ระหว่างค่าต่ำสุดและค่าสูงสุด ของระดับความเข้มของข้อมูลอินพุท เช่น ภาพข้อมูลอินพุทมีระดับเทา 256 ระดับ จะมีค่าระดับเทาได้ตั้งแต่ 0 – 255 ค่าเทรชโฮลด์ที่สามารถเลือกได้ก็คือ ค่าที่อยู่ระหว่าง 0 – 225 เมื่อเลือกค่าเทรชโฮลด์ได้แล้วสามารถทำการเซกเมนต์ภาพได้โดยใช้สมการที่ 2.1

การหาค่าเทรชโฮลด์โดยการกำหนดค่าล่วงหน้า อาจทำได้โดยทำการเลือกค่าเทรชโฮลด์ขึ้นมา เช่น 50,100,150.200 เพื่อนำมาทดลองทำการเซกเมนต์ภาพ แล้วดูผลลัพธ์ที่ได้ จากนั้นจึงเลือกค่าที่เหมาะสมมาใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.7 การหาค่าเทรชโฮลด์จากค่ากลาง

การหาค่าเทรชโฮลด์โดยพิจารณาจากค่ากลาง เป็นการหาค่าเทรชโฮลด์ที่แตกต่างจากการหาค่าเทรชโฮลด์วิธีแรก สำหรับวิธีนี้จะเป็นการคำนวณหาค่าเทรชโฮลด์โดยอัตโนมัติโดยไม่ต้องให้ผู้ใช้เป็นผู้กำหนด โดยการหาค่าเทรชโฮลด์วิธีนี้ได้อาศัยการคำนวณพื้นฐานทางสถิติในเรื่องของการหาค่ากลางหรือค่าเฉลี่ยมาประยุกต์ใช้ ค่าเทรชโฮลด์ที่คำนวณได้จะเป็นค่าที่ได้จากค่ากึ่งกลางที่อยู่ระหว่างค่าระดับความเข้มสูงสุดและค่าระดับความเข้มต่ำสุดของข้อมูลภาพอินพุท สำหรับการคำนวณหาค่ากึ่งกลางนี้สามารถคำนวณได้จากสมการที่2.2

$$Thr = \frac{Maximum(g(x,y)) + Minimum(g(x,y))}{2} \qquad (2.2)$$

โดยที่

Thr	ค่าเทรชโฮลด์
g(x,y)	ข้อมูลภาพอินพุทที่มีระดับความเข้ม 0 ถึง L ระดับ
Maximum(g(x,y))	ค่าสูงสุดของระดับสีเทาของข้อมูลอินพุท
Minimum(g(x,y))	ค่าต่ำสุดของระดับสีเทาของข้อมูลอินพุท

เมื่อคำนวณค่าเทรชโฮลด์ได้แล้ว ก็สามารถทำการเซกเมนต์ภาพได้ โดยนำค่าเทรชโฮลด์ที่ได้มาแทนค่าในสมการที่2.1

2.1.1.8 การหาค่าเทรชโฮลด์โดยพิจารณาจากฮีสโตแกรม

การหาค่าเทรชโฮลด์โดยวิธีพิจารณาจากฮีสโตแกรม เป็นการพิจารณาจากค่าของฮีสโตแกรมระดับสีเทาของข้อมูลภาพที่เป็นอินพุท โดยที่การหาค่าเทรชโฮลด์วิธีนี้ข้อมูลภาพอินพุทที่เหมาะสมต้องมีลักษณะที่สามารถแบ่งแยกข้อมูลออกเป็นสองกลุ่มได้อย่างชัดเจน คือ กลุ่มหนึ่งจะเป็นกลุ่มของวัตถุและอีกกลุ่มหนึ่งจะเป็นพื้นหลัง ซึ่งแนวคิดในการคำนวณหาค่าเทรชโฮลด์ของวิธีนี้สามารถแสดงได้ดังรูปที่ 2.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 2.6แนวคิดในการคำนวณค่าเทรชโฮลด์โดยวิธีพิจารณาจากฮีสโตแกรม

สำหรับการคำนวณหาค่าเทรชโฮลด์โดยวิธีพิจารณาจากฮีสโตแกรมสามารถสรุปเป็นขั้นตอนได้ดังนี้

1. ข้อมูลภาพอินพุทที่มีระดับความเข้ม 0 ถึง L ระดับ
2. คำนวณหาฮีสโตแกรมระดับความเข้มของภาพ โดยการนับจำนวนจุดภาพที่มีระดับความเข้มแต่ละระดับ
3. จากฮีสโตแกรมจะพบว่าจะเกิดกลุ่มของระดับความเข้ม 2 กลุ่มคือกลุ่มของวัตถุและกลุ่มของพื้นหลัง
4. จะเลือกค่าต่ำสุดที่อยู่ระหว่าง 2 กลุ่มนั้น ซึ่งค่านั้นก็คือ ค่าเทรชโฮลด์นั่นเอง
5. ทำการเซกเมนต์ภาพ โดยนำค่าเทรชโฮลด์ที่ได้มาแทนในสมการที่2.1

จากขั้นตอนการทำงานข้างต้น ถ้าได้ทดลองทำกับภาพ 256 ระดับสีเทาที่เป็นภาพตัวอักษรบนพื้นกระดาษที่สีอ่อนกว่า ซึ่งเมื่อทำการสร้างฮีสโตแกรมระดับสีเทาของภาพแล้วจะปรากฏกลุ่มของข้อมูล 2 กลุ่ม คือ กลุ่มหนึ่งจะเป็นกลุ่มของข้อความ และอีกกลุ่มหนึ่งจะเป็นกลุ่มของพื้นหลัง จากนั้นทำการเลือกค่าต่ำสุดระหว่าง 2 กลุ่มนั้นเป็นค่าเทรชโฮลด์ซึ่งค่าที่ได้จากวิธนี้จะมีค่าเที่ยงตรงที่สุด แต่อย่างไรก็ดีวิธีนี้ก็ไม่เหมาะสมกับภาพที่ไม่สามารถแยกกลุ่มของสิ่งที่อยู่ในภาพได้อย่างชัดเจนระหว่างกลุ่มของวัตถุและกลุ่มของพื้นหลัง เนื่องจากว่าถ้าหากภาพอินพุทไม่สามารถแยกได้แล้วจะทำให้ค่าเทรชโฮลด์ที่คำนวณได้จะผิดไปจากความเป็นจริง คือ อาจจะมากเกินไปหรืออาจจะน้อยเกินไป อันเป็นผลทำให้ภาพที่ได้ไม่มีเหมาะสม รายละเอียดบางส่วนขาดหายไปอาจจะขาวหรือดำเกินไป

2.1.2 การกำจัดสัญญาณรบกวน

ข้อมูลภาพที่รับเข้ามาจากกล้องวีดีโออาจเป็นข้อมูลภาพที่ไม่พร้อมสำหรับการประมวลผลภาพจริง เพราะอาจเกิดจุดภาพที่ไม่พึงประสงค์ซึ่งไม่ควรจะนำไปประมวลผล ดังนั้นเราต้องทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำจัดจุดภาพที่ไม่ต้องการเหล่านั้นออกไป ซึ่งได้มีวิธีในการกำจัดสัญญาณรบกวน 2 รูปแบบคือ การใช้มอร์โฟโลยี (Morphology) และการใช้อัลกอริธึมของ kFill โดยการกำหนดมาส์ก (Mask) หรือหน้าต่างในการทดลองเท่ากับ 3x3

การกำจัดสัญญาณรบกวนถือว่าเป็นการประมวลผลภาพขั้นต้น ซึ่งจะต้องทำก่อนการประมวลผลภาพจริง การกำจัดสัญญาณรบกวนเป็นขั้นตอนที่ใช้ในการกำจัดจุดภาพที่ไม่พึงประสงค์ออกจากภาพ สัญญาณรบกวนบนข้อมูลภาพอาจเกิดจากจุดภาพจากการรับภาพทางกล้องวีดีโอหรืออาจเกิดจากฟังก์ชันในการแปลงจากข้อมูลเชิงอนาลอกไปเป็นข้อมูลเชิงดิจิตอล

โดยทั่วไปแล้วลักษณะของข้อมูลภาพที่จะเป็นสัญญาณรบกวน(ทั้งประเภทจุดภาพเดียวและกลุ่มของจุดภาพ) จะมีอยู่ 2 รูปแบบคือสัญญาณรบกวนประเภทที่เป็นจุดภาพดำไปปรากฏอยู่บนกลุ่มของจุดภาพที่เป็นพื้นหลังขาว และสัญญาณรบกวนประเภทที่เป็นพื้นหลังขาวที่ไปปรากฏอยู่บนตัวเนื้อของกลุ่มจุดภาพดำนั้นก็คือ จุดภาพที่เป็นสัญญาณรบกวนต้องถูกล้อมรอบด้วยข้อมูลอื่น ดังนั้นในการกำจัดสัญญาณรบกวนก็จะต้องเป็นการพิจารณากระหว่างจุดภาพสัญญาณรบกวนกับจุดภาพที่อยู่รอบข้าง โดยการกำหนดเป็นกรอบหน้าต่างหรือเมตริกซ์จตุรัส

2.1.2.1 การกำจัดสัญญาณรบกวนโดยวิธีมอร์โฟโลยีทางคณิตศาสตร์

มอร์โฟโลยี หมายถึงการนำหลักการทางคณิตศาสตร์มาใช้ในการศึกษารูปแบบและโครงสร้างของวัตถุหรือสิ่งของอย่างใดอย่างหนึ่ง ซึ่งคณิตศาสตร์ที่จะใช้ในกระบวนการมอร์โฟโลยี คือ ทฤษฎีเซต (Set Theory)

บทนิยามเบื้องต้นของมอร์โฟโลยีทางคณิตศาสตร์

ถ้าเรากำหนดเซต A และ B เป็นเซตอยู่ในมิติ 2 มิติ โดยกำหนดให้ $a=(a_1,a_2)$, $b=(b_1,b_2)$ และ $x=(x_1,x_2)$ นิยามเบื้องต้นของมอร์โฟโลยีทางคณิตศาสตร์ที่สำคัญมีดังนี้

1. ทรานสเลชั่น(Translation) เป็นการย้ายตำแหน่งของจุดภาพในเซต A โดยสเกลาร์ $x=(x_1,x_2)$ เขียนแทนด้วย $(A)_x$ นิยามไว้ดังนี้

$$(A)_x = \{c|c = a+x \; ; \; a \in A\} \tag{2.3}$$

2. รีเฟล็กชั่น(Reflection) ของเซต B เขียนแทนด้วย $\hat{B}$ นิยามไว้ดังนี้

$$\hat{B} = \{x|x = -b \; ; \; b \in B\} \tag{2.4}$$

*3. คอมพลีเมนต์(Complement)*ของเซต A เขียนแทนด้วย A^c นิยามไว้ดังนี้

$$A^c = \{x|x \notin A\} \tag{2.5}$$

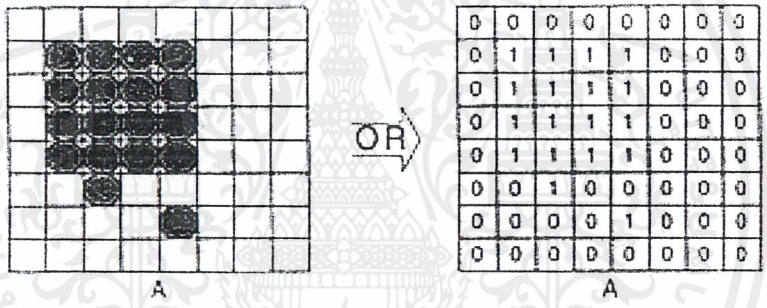
4. ผลต่างของเซต Aและ B(Difference) เขียนแทนด้วย A-B นิยามไว้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$A-B = \{x|x \in A, x \notin B\} \quad (2.6)$$

โดยทั่วไปหลักการพื้นฐานของมอร์โฟโลยีทางคณิตศาสตร์จะเป็นการพิจารณาเฉพาะจุดภาพที่เป็นตัวเนื้อข้อมูลเท่านั้น ซึ่งนั่นก็คือถ้าเรากำลังสนใจภาพ 2 ระดับ การทำมอร์โฟโลยีก็จะเป็นการพิจารณาเฉพาะกลุ่มของจุดภาพดำซึ่งเป็นตัวเนื้อข้อมูลเท่านั้น ส่วนกลุ่มของจุดภาพขาวที่เป็นภาพพื้นหลังที่ไปปรากฏอยู่จะไม่นำมาพิจารณา มอร์โฟโลยีทางคณิตศาสตร์มีรูปแบบการทำงานที่สำคัญคือ ไดเลชั่น อิรอชั่น โคลส์ซิง และ โอเพนนิ่ง

กำหนดให้ข้อมูลภาพที่ศึกษาเป็นข้อมูลในเมตริกซ์ 2 มิติ กล่าวคือ แต่ละจุดภาพสามารถกำหนดให้อยู่ในรูปความสัมพันธ์ของระบบแกนพิกัดมุมฉาก (XY-Plane) ภาพที่ 2.7 แสดงตัวอย่างที่อยู่ในเมตริกซ์ 2 มิติ



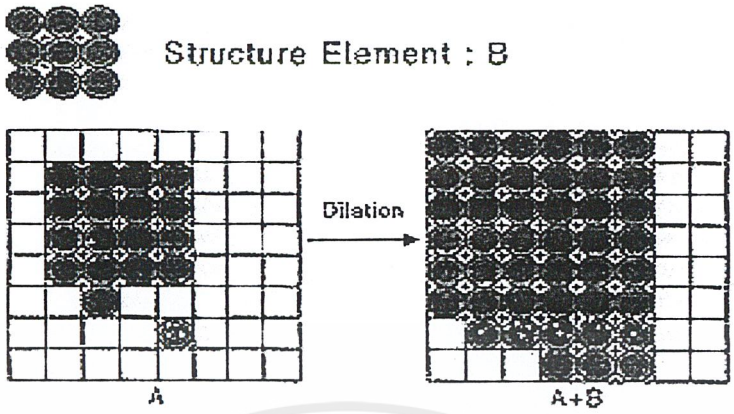
ภาพที่ 2.7 ตัวอย่างจุดภาพในเมตริกซ์ 2 มิติ

1. ไดเลชั่น (Dilation) มีเครื่องหมายบวก(+)เป็นสัญลักษณ์ทางคณิตศาสตร์ ไดเลชั่นเป็นการศึกษาพฤติกรรมของภาพเพื่อทำการเพิ่มจำนวนจุดภาพดำบนภาพ โดยการพิจารณาร่วมกันระหว่างเซตของจุดภาพ A และสตรัคเจอริอิลิเมนต์ (Structure Element : B) ซึ่งทั้งสองเซตเมื่อนำมาทำการไดเลชั่นกันจะเป็นไปตามความสัมพันธ์ของนิยามที่2.7

$$A+B = \{x|(\hat{B})_x \cap A \neq \varnothing\} \quad (2.7)$$

การทำไดเลชั่นระหว่างเซต A กับ B จะเป็นการขยายขนาดของกลุ่มจุดภาพดำในเซต A ให้มีขนาดกว้างขึ้น ส่งผลให้ช่องไฟระหว่างกลุ่มจุดดำมีขนาดแคบลง หรือบางครั้งก็อาจทำให้กลุ่มของจุดภาพดำสองกลุ่มที่อยู่ใกล้ชิดกันเชื่อมติดถึงกันได้ ตัวอย่างของการทำไดเลชั่น ได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.8 ตัวอย่างการทำไดเลชั่น

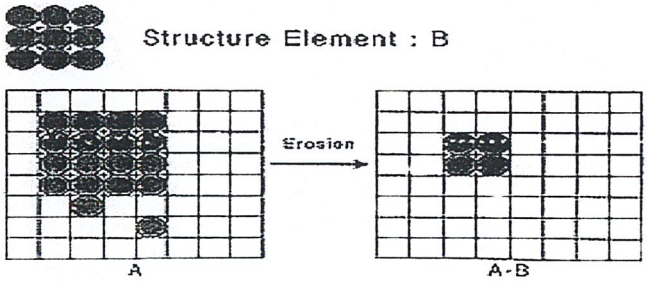
ตำแหน่งจุดภาพเริ่มต้นของเซต A,เซตB และการทำไดเลชั่นของเซต A+B

2. อิรอชั่น(Erosion) มีเครื่องหมายลบ(-)เป็นสัญลักษณ์ทางคณิตศาสตร์ อิรอชั่นเป็นการศึกษาพฤติกรรมของภาพเพื่อทำการลดจำนวนจุดภาพดำบนภาพ การลดจุดภาพดำจะต้องพิจารณาประกอบกันระหว่างเซตจุดภาพกับสตรัคเจอร์อิลิเมนต์ โดยจะเป็นไปตามความสัมพันธ์ของนิยามที่2.8

$$A-B = \{x|(B)_x \subseteq A\} \qquad (2.8)$$

จากนิยามที่2.8ผลของการทำอิรอชั่นระหว่างเซต A และ B จะทำให้เราได้คำตอบเป็น x อยู่ในสเปซ 2 มิติ โดย x จะต้องเป็นสับเซตของเซต A

นอกจากการอิรอชั่นจะเป็นการลดจำนวนของจุดภาพดำบนข้อมูลภาพแล้ว การทำอิรอชั่นยังจะทำให้ขนาดของช่องไฟระหว่างกลุ่มของจุดภาพดำแต่ละกลุ่มมีขนาดใหญ่มากขึ้น นั่นก็คือกลุ่มของจุดภาพดำแต่ละกลุ่มจะแยกจากกันอย่างชัดเจน นอกจากนี้ถ้ากลุ่มของจุดภาพดำมีขนาดเล็กมาก กลุ่มจุดภาพดำนั้นก็จะถูกขจัดออกจากภาพเอกสารไปโดยอัตโนมัติเมื่อผ่านกระบวนการทำอิรอชั่น

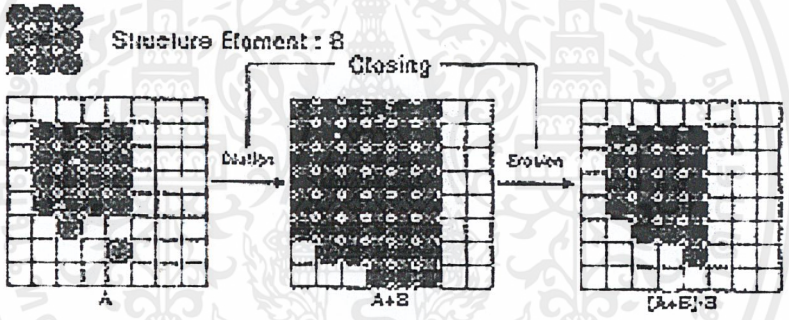


ภาพที่ 2.9 ตัวอย่างการทำอิรอชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

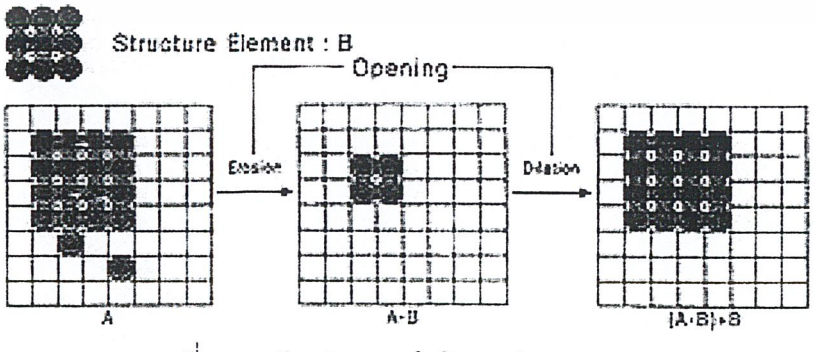
ตำแหน่งจุดภาพเริ่มต้นของเซต A,เซต Bและการทำอิรอชั่นของเซต A-B

3. โคลส์ซิง(Closing)เป็นการแปลงมอร์โฟโลยีทางคณิตศาสตร์ที่เริ่มจากการไดเลชั่นเมื่อเสร็จแล้วตามด้วยการทำอิรอชั่นเท่ากับจำนวนครั้งของการทำไดเลชั่น นั่นก็คือการทำโคลส์ซิงจะเป็นการเพิ่มจำนวนจุดภาพดำลงไปในภาพก่อน จากนั้นค่อยทำการกัดเซาะของจุดภาพดำเหล่านั้นออกโดยใช้สตรัคเจอร์อิลิเมนต์ จากที่กล่าวมาการทำโคลส์ซิงมอร์โฟโลยีก็เป็นการปรับปรุงคุณภาพอย่างหนึ่งที่จะทำให้ขนาดของกลุ่มจุดภาพดำขยายใหญ่ขึ้นจากการทำไดเลชั่นหลายๆ ครั้ง และเมื่อทำอิรอชั่นจะเป็นการไปกัดเซาะแนวของกลุ่มจุดภาพดำนั้นๆ ทำให้คุณภาพของกลุ่มจุดภาพดำมีความเรียบมากขึ้น ในการทำโคลส์ซิงมอร์โฟโลยีถ้ากำหนดจำนวนรอบการทำซ้ำไว้มากๆ อาจทำให้กลุ่มของจุดภาพดำ 2 กลุ่มที่อยู่ใกล้กันเชื่อมเข้าเป็นกลุ่มของจุดภาพดำเดียวกันได้ นอกจากนี้การโคลส์ซิงจะทำให้จุดภาพดำที่เป็นสัญญาณรบกวนขยายขนาดขึ้นเองโดยอัตโนมัติจนไม่สามารถขจัดออกไปได้



ภาพที่ 2.10 ตัวอย่างการทำโคลส์ซิง : (A+B) - B

4. โอเพนนิ่ง(Opening) เป็นการแปลงมอร์โฟโลยีชนิดที่ทำงานตรงกันข้ามกับการทำโคลส์ซิงมอร์โฟโลยีชนิดที่ทำงานตรงกันข้ามกับการทำโคลส์ซิงมอร์โฟโลยี กล่าวคือขั้นตอนการทำงานเริ่มจากการลดจำนวนจุดภาพดำด้วยกระบวนการอิรอชั่น แล้วตามด้วยการเติมจุดภาพดำกลับด้วยกระบวนทำไดเลชั่นให้เท่ากับจำนวนครั้งของการทำอิรอชั่น



ภาพที่ 2.11 ตัวอย่างการทำโอเพนนิง : (A-B) + B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2.2 การกำจัดสัญญาณรบกวนโดยอัลกอริธึมkFill

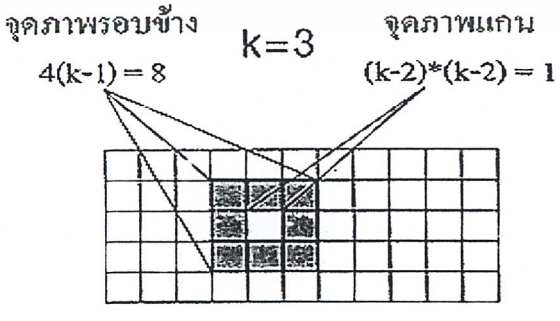
สัญญาณรบกวนมี 2 ประเภทคือพวกที่เป็นจุดภาพดำไปปรากฏอยู่บนกลุ่มของจุดภาพขาวที่เป็นภาพพื้นหลัง และพวกจุดภาพขาวไปปรากฏอยู่บนกลุ่มของจุดภาพดำ ดังนั้นการกำจัดสัญญาณรบกวนต้องพิจารณาเติมค่าทั้งสองกรณีไปพร้อมกัน โดยที่เราจะไม่สามารถพิจารณาเฉพาะกลุ่มของจุดภาพดำแต่อย่างเดียวดังเช่นการทำมอร์โฟโลยีทางคณิตศาสตร์ได้

kFillจัดเป็นวิธีที่เหมาะสำหรับการกำจัดสัญญาณรบกวนบนข้อมูลภาพโดยเป็นวิธีการเลือกเติมค่าของทุกๆ ตำแหน่งจุดภาพโดยพิจารณาร่วมกับค่าของจุดภาพข้างเคียงเช่นจำนวนจุดภาพขาวหรือดำที่ปรากฏอยู่ในตำแหน่งของกลุ่มจุดภาพข้างเคียงทั้งหมด จำนวนจุดภาพขาวหรือดำในตำแหน่งจุดมุมทั้ง 4 ของกรอบหน้าต่างและช่วงของความต่อเนื่องกันของตำแหน่งจุดภาพขาวหรือดำที่อยู่ล้อมรอบกลุ่มของจุดภาพที่เราสนใจ

หลักการของkFill

การพิจารณาเติมค่าลงในตำแหน่งจุดภาพใดๆบนภาพโดยวิธีkFill จะต้องพิจารณาร่วมกนระหว่างค่าของกลุ่มตำแหน่งจุดภาพที่กำลังสนใจกับค่าตำแหน่งจุดภาพที่อยู่รอบข้าง ดังนั้นเราจึงต้องสร้างกรอบหน้าต่างหรือมาส์ก สำหรับการพิจารณาเติมค่า ซึ่งเราจะกำหนดขนาดของหน้าต่างขนาด kxk จุดภาพ จะพบว่ามีกลุ่มของจุดภาพที่เรียกว่า"จุดภาพแกน" (Core) ซึ่งจะใช้กำหนดให้เป็นกลุ่มของจุดภาพที่เราสนใจพิจารณาเติมค่าได้เท่ากับ (k-2) x (k-2) จุดภาพและอีก 4(k-1) เป็นกลุ่มจุดภาพที่อยู่รอบนอกของกลุ่มจุดภาพที่เราสนใจพิจารณาเติมค่าเราจะเรียกว่า"จุดภาพรอบข้าง"

ในขั้นตอนของการพิจารณาเติมค่ากลุ่มจุดภาพที่เป็นกลุ่มจุดภาพแกนของแต่ละกรอบหน้าต่างว่าจะเติมเป็นกลุ่มจุดภาพดำ(On)หรือขาว(Off) จะเป็นการพิจารณากลุ่มของจุดภาพแกนกับเงื่อนไขของกลุ่มของจุดภาพรอบข้างดังนี้



ภาพที่ 2.12 การพิจารณากลุ่มของจุดภาพแกนกับกลุ่มของจุดภาพรอบข้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพิจารณาเติมค่ากลุ่มของจุดภาพแกนให้เป็นกลุ่มของจุดภาพดำต้องเป็นไปตามเงื่อนไขดังนี้

1. กลุ่มของจุดภาพแกนทั้งหมดต้องเป็นจุดภาพขาว
2. เงื่อนไขของกลุ่มจุดภาพรอบข้างให้พิจารณาตัวแปร 3 ตัวแปรดังนี้

   n : จำนวนของจุดภาพดำ

   c : จำนวนกลุ่มของจุดภาพดำที่เชื่อมต่อเนื่องกัน

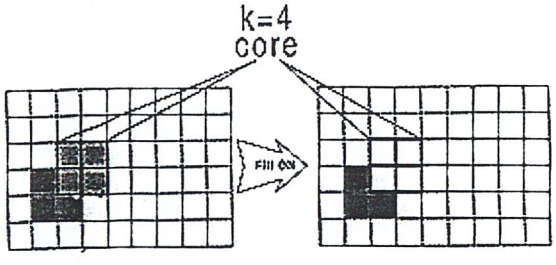
   r : จำนวนของจุดภาพดำที่พบบนมุมทั้ง 4 ของหน้าต่าง

โดยค่าของตัวแปรทั้ง 3 ต้องเป็นไปตามสมการ

$$(c=1) \text{ and } [(n>3k-4) \text{ or } (n=3k-4) \text{ and } (r=2))] \qquad (2.9)$$

ถ้าเราพบว่ากลุ่มของจุดภาพแกนและจุดภาพรอบข้างเป็นไปตามเงื่อนไขทั้ง 2 ก็ให้เติมค่ากลุ่มของจุดภาพแกนเป็นกลุ่มของจุดภาพดำ แต่ถ้าหากว่าไม่เป็นไปตามเงื่อนไขที่กำหนดไว้ก็ให้เติมกลุ่มของจุดภาพแกนเป็นกลุ่มของจุดภาพขาว สำหรับการเลือกเติมกลุ่มของจุดภาพแกนให้เป็นกลุ่มของจุดภาพขาวก็พิจารณาในทางตรงกันข้ามกันกับการพิจารณาเติมค่ากลุ่มของจุดภาพแกนเป็นกลุ่มของจุดภาพดำ

จากสมการที่2.9 เป็นสมการอธิบายสมการความสัมพันธ์ของการพิจารณาเติมค่าตำแหน่งจุดภาพจากจุดภาพรอบข้าง ความหมายของตัวแปรในตำแหน่งจุดภาพรอบข้างแต่ละตัวจะอธิบายได้ดังนี้ n และ r จะเป็นไปตามขนาดของหน้าต่าง k โดยตัวแปร n จะเป็นพารามิเตอร์ที่สำคัญที่จะบอกว่าควรจะเติมจุดภาพแกนเป็นอะไรดี ยกตัวอย่างเช่น ถ้าจุดภาพแกนเป็นจุดภาพขาวทั้งหมดนั่นคือต้องพิจารณาเติมจุดภาพแกนเป็นจุดภาพดำ และถ้าจำนวนจุดภาพดำที่อยู่ในกลุ่มภาพรอบข้างมากกว่า 3k-4 จุดภาพก็ควรจะเติมจุดภาพแกนเป็นจุดภาพดำ แต่ก็ยังเติมไม่ได้เสียทีเดียวเพราะต้องพิจารณาตัวแปร c ประกอบตัวหนึ่งเพราะตัวแปร c จะเป็นตัวแปรสำหรับรักษาคุณสมบัติของวัตถุ 2 วัตถุ ที่อยู่ใกล้กันไม่ให้เกิดการเชื่อมติดกันหรือกรณีหนึ่งก็คือ ไม่ทำให้วัตถุหนึ่งวัตถุใดเกิดการแยกขาดจากกัน ภายหลังจากการเติมค่าจุดภาพแกน



ภาพที่ 2.13 การเลือกเติมจุดภาพแกนเป็นจุดภาพดำ และภาพผลลัพธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

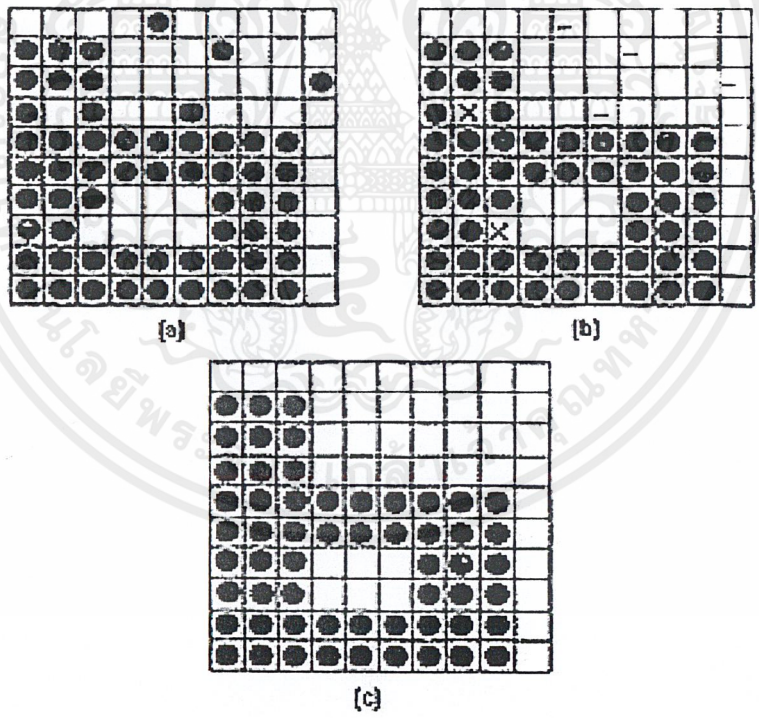
จากตัวอย่างภาพที่ 2.13 เมื่อพิจารณาจะเห็นได้ว่า จุดภาพแกนทุกจุดเป็นภาพดำ (On) ดังนั้นเราจะพิจารณาค่าจุดภาพแกนเป็นจุดภาพขาว (Off) เมื่อพิจารณาเงื่อนไขของจุดภาพขาวในตำแหน่งจุดภาพรอบข้าง พบว่าจำนวนช่วงของความต่อเนื่องกันของจุดภาพขาว :c=1 และเมื่อพิจารณาจุดภาพขาวนับได้ :(n=9) > (3k-4=8) ฉะนั้นให้เติมจุดภาพแกนเป็นจุดภาพขาว

ในความเป็นจริงแล้ว อัลกอริธึม kFill ที่นำเสนอนี้ ในแต่ละรอบของการพิจารณาเติมค่าจุดภาพแกน จะแบ่งการประมวลผลออกเป็น 2 ขั้นตอนย่อยๆ คือ

1.การพิจารณาเติมค่ากลุ่มของจุดภาพแกนให้เป็นกลุ่มจุดภาพดำ (Fill On)

2.การพิจารณาเติมค่ากลุ่มของจุดภาพแกนให้เป็นกลุ่มจุดภาพขาว(Fill Off)

กล่าวคือในทุกกลุ่มจุดภาพที่พิจารณานั้น จะต้องทำ 2 ขั้นตอนไปพร้อมๆ กันซึ่งจากวิธีการนี้ทำให้ทุกกลุ่มจุดภาพแกนได้รับการพิจารณาทุกกรณีทั้งการเติมจุดภาพขาวและการเติมจุดภาพดำ



ภาพที่ 2.14 ตัวอย่างการกำจัดสัญญาณรบกวนโดยใช้ kFill

จากภาพที่ 2.14 เริ่มจากการพิจารณาจากภาพที่ 2.14 a) เป็นภาพที่มีสัญญาณรบกวนทั้งที่เป็นจุดภาพดำและขาว ภาพที่ 2.14 b) ถ้าเรากำหนดขนาดของหน้าต่างเพื่อการประมวลผลเป็น 3x3 จะได้จุดภาพแกน (k-2)x(k-2) เท่ากับ 1 จุดภาพ และจุดภาพรอบข้างจำนวน 4(k-1) เท่ากับ 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

45942

จุดภาพ ส่วนภาพที่ 2.14 c) เป็นผลลัพธ์จากการกำจัดสัญญาณรบกวนโดยใช้ kFill เมื่อกำหนดขนาดหน้าต่างเป็น 3x3

เครื่องหมาย(-)สำหรับใช้แทนจุดภาพที่จะพิจารณาเติมค่าเป็นจุดภาพขาว หรือจุดภาพพื้นหลัง

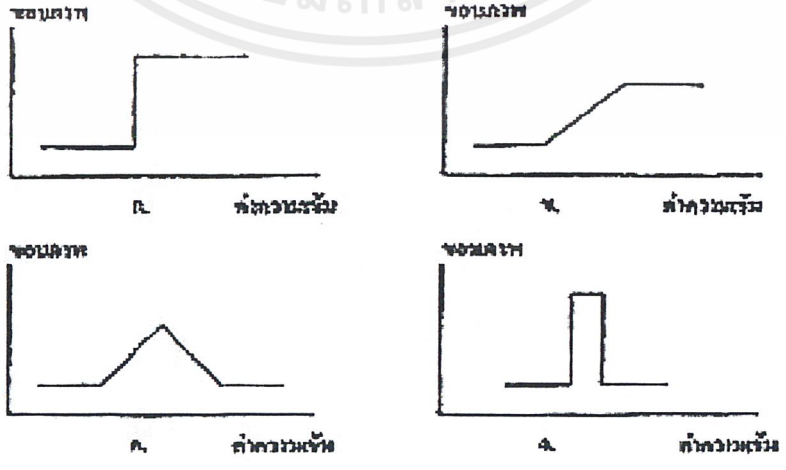
เครื่องหมาย(x)สำหรับใช้แทนจุดภาพที่จะพิจารณาเติมค่าเป็นจุดภาพดำ หรือจุดภาพของวัตถุ

2.1.3 การหาขอบภาพ

การหาขอบภาพเป็นการหาขอบเขตของวัตถุภายในภาพ ซึ่งขอบเขตของวัตถุจะเป็นคุณสมบัติที่เด่นและมีความสำคัญมากที่นำไปรู้จักวัตถุนั้นๆ โดยคอมพิวเตอร์ การหาขอบภาพก็คือการดึงคุณลักษณะโครงร่างเด่นของวัตถุออกมา ลักษณะเด่นของวัตถุที่เรามองเห็นทั่วไปก็คือ ส่วนที่เป็นสันหรือขอบของวัตถุ เมื่อส่วนนี้มีแสงมาตกกระทบก็จะมีความสว่างมาก ซึ่งลักษณะเช่นนี้ก็เกิดขึ้นในภาพเชิงตัวเลขที่นำมาประมวลผลเช่นกัน

ขอบภาพเชิงตัวเลข คือ ฟังก์ชันค่าความเข้มของจุดภาพที่มีการเปลี่ยนแปลงค่าความเข้มอย่างทันทีทันใด และจุดที่เป็นขอบภาพของวัตถุจะแบ่งพื้นผิวของวัตถุหนึ่งออกจากวัตถุชิ้นหนึ่งหรืออาจจะแบ่งแยกวัตถุออกจากสีของพื้นหลัง โดยทั่วไปลักษณะของขอบภาพที่ดีจะต้องมีลักษณะดังต่อไปนี้

1. มีความบางคือส่วนที่เป็นขอบภาพจะต้องมีความกว้างเพียงจุดเดียว
2. มีความต่อเนื่อง โดยจุดของขอบภาพในวัตถุเดียวกันควรมีความต่อเนื่องกัน ถ้าจุดของขอบภาพมีอยู่เพียงจุดเดียว ไม่ต่อเนื่องกับจุดใดในย่านใกล้เคียงเลยอาจเป็นขอบภาพที่ไม่สมบูรณ์หรืออาจเป็นสัญญาณรบกวนก็ได้



ภาพที่ 2.15 (ก) – (ง) แสดงขอบภาพชนิดต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของขอบภาพที่ปรากฏในรูปภาพโดยทั่วไป จะประกอบด้วยขอบภาพหลายชนิด ดังแสดงในภาพที่ 2.15 ซึ่งภาพที่ 2.15 (ก) เป็นของภาพที่มีการเปลี่ยนแปลงอย่างทันทีทันใด ขอบภาพชนิดนี้ส่วนมากจะอยู่ในภาพที่มนุษย์จำลองขึ้นมา ส่วนขอบภาพในรูปภาพทั่วไปจะเป็นขอบภาพที่มีการเปลี่ยนแปลงค่าความเข้มทีละน้อย ดังแสดงในภาพที่2.15 (ข),2.15 (ค) เป็นขอบภาพที่ค่าความเข้มค่อยๆเพิ่มขึ้นหรือลดลงทีละน้อยในลักษณะของรูปหน้าจั่ว และภาพที่2.15(ง)แสดงของภาพที่เป็นเส้น

การเปลี่ยนแปลงหรือความไม่ต่อเนื่องในฟังก์ชันค่าความเข้มของจุดภาพที่เกิดจากการส่องสว่าง หรือลักษณะทางกายภาพของตัววัตถุเอง เช่น ลักษณะของพื้นผิว,รูปทรง คุณสมบัติเหล่านี้จะเป็นปัจจัยพื้นฐานที่สำคัญของรูปภาพที่จะบอกถึงขอบเขตทางกายภาพของวัตถุที่อยู่ในภาพนั้นฟังก์ชันจุดภาพ g(x,y) ที่เป็นขอบภาพจะมีการเปลี่ยนแปลงจากค่าความเข้มหนึ่งไปยังอีกค่าความเข้มหนึ่ง โดยค่าความเข้มนั้นจะต้องแตกต่างกันอย่างมีนัยสำคัญ คำว่าแตกต่างอย่างมีนัยสำคัญของค่าความเข้มในที่นี้มีตัววัดมากมายหลายวิธี และตัววัดความแตกต่างเหล่านี้จะเรียกว่าตัวดำเนินการหาขอบภาพซึ่งจะอธิบายแต่ละวิธีได้ดังนี้

2.1.3.1 การหาขอบภาพโดยใช้อนุพันธ์เชิงเส้นอันดับหนึ่ง

การหาขอบภาพเป็นขั้นตอนที่สำคัญในการสืบหาบริเวณที่มีการเปลี่ยนแปลงระดับความเข้มของจุดภาพที่เกิดขึ้น โดยการหาค่ากราเดียนท์บนข้อมูลภาพซึ่งกราเดียนท์เป็นฟังก์ชันที่วัดอัตราการเปลี่ยนแปลงระดับความเข้มที่เกิดขึ้นของจุดภาพ ซึ่งอยู่บนพื้นฐานการหาอนุพันธ์เชิงเส้นอันดับหนึ่ง และกราเดียนท์ $\nabla f(x,y)$ ที่เป็นฟังก์ชันต่อเนื่องในระนาบสองมิติสามารถกำหนดอยู่ในรูปของเวคเตอร์ได้ดังสมการต่อไปนี้

$$\nabla f(x,y) = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \partial f / \partial x \\ \partial f / \partial y \end{bmatrix} \qquad (2.10)$$

ขนาดของกราเดียนท์(Magnitude of Gradient) จะเท่ากับ อัตราการเปลี่ยนแปลงระดับความเข้มของจุดภาพในทิศทาง G ที่กำหนด (ทิศทาง G ในที่นี้คือ ทิศทางในแนวนอนและแนวตั้ง) ซึ่งขนาดของกราเดียนท์สามารถหาได้จากสมการดังต่อไปนี้

$$\nabla f(x,y) = \sqrt{(G_x^2 + G_y^2)} \qquad (2.11)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาดของกราเดียนท์โดยประมาณ (Approximate Magnitude of Gradient) สามารถหาได้จากสมการค่าสัมบูรณ์ดังต่อไปนี้

$$\nabla f(x,y) \approx |G_x| + |G_y| \qquad (2.12)$$

ทิศทางของกราเดียนท์จะเท่ากับ

$$\alpha(x,y) = \tan^{-1}(G_x / G_y) \qquad (2.13)$$

สำหรับภาพเชิงตัวเลขการหาค่ากราเดียนท์ของฟังก์ชันสองมิติ g(x,y) ซึ่งเป็นฟังก์ชันไม่ต่อเนื่องที่พิกัด (x,y) สามารถกระทำในรูปของสมการดังต่อไปนี้

$$G_x = g(x+1,y) - g(x,y) \qquad (2.14)$$

$$G_y = g(x,y) - g(x,y+1) \qquad (2.15)$$

จากสมการ2.14 และ 2.15 สามารถพิจารณาในรูปของการนำวินโดว์ขนาด 1x2 และ 2x1 ที่แสดงดังภาพที่ 2.16 มาคอนโวลูชั่นหรือการประสานงานกับข้อมูลภาพเชิงตัวเลขเพื่อใช้สำหรับคำนวณหาค่า G_x และ G_y ได้ตามลำดับ

(ก) (ข)

ภาพที่ 2.16 วินโดว์สำหรับการคำนวณหาค่ากราเดียนท์ (ก) วินโดว์ขนาด 1x2 ที่ใช้สำหรับหาค่า G_x (ข) วินโดว์ขนาด 2x1 ที่ใช้สำหรับหาค่า G_y

จากขั้นตอนการนำวินโดว์ขนาด 1x2 และ 2x1 ข้างต้นมาคอนโวลูชั่นกับข้อมูลภาพเชิงตัวเลขเพื่อหาค่ากราเดียนท์ในทิศทาง x และ y ตามลำดับ พบว่าค่า G_x ที่คำนวณได้นั้นเปรียบเสมือนค่าที่เกิดจากการอินโทโปรเลตหรือการทำนายเชิงเส้นที่พิกัดจุดภาพ ((x+1)/2 , y) โดยอ้างอิงระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเข้มของจุดภาพข้างเคียงที่มีพิกัดสอดคล้องกับพิกัดสอดคล้องกับพิกัดบนวินโดว์ ส่วนค่า G_y ที่คำนวณได้นั้นเปรียบเสมือนค่าที่เกิดจากการอินโทโปรเลตที่พิกัดจุดภาพ (x,(y+1)/2) โดยอ้างอิงระดับความเข้มของจุดภาพข้างเคียงที่มีพิกัดสอดคล้องกับพิกัดบนวินโดว์เช่นกัน นอกจากนี้ยังสามารถที่นำวินโดว์ขนาด 2x2 ดังภาพที่ 2.17 (ก) และ (ข) มาคอนโวลูชั่นกับภาพเชิงตัวเลขเพื่อหาค่า G_x และ G_y ตามลำดับ และขนาดของกราเดียนท์ที่เกิดขึ้นจะสูงกว่าคอนโวลูชั่นด้วยวินโดว์ขนาด 1x2และ 2x1 และจากการคอนโวลูชั่นข้อมูลภาพด้วยวินโดว์ขนาด 2x2 พบว่าค่ากราเดียนท์ที่คำนวณได้ทั่วทิศทาง x และทิศทาง y เปรียบเสมือนค่าที่เกิดจากการอินโทโปเลตที่พิกัดจุดภาพ g(x+?, y+?) โดยอ้างอิงระดับความเข้มของจุดภาพที่มีพิกัดสอดคล้องกับพิกัดบนวินโดว์ หรืออาจจะใช้วินโดว์ขนาด 3x3 มาคำนวณหากราเดียนท์ของจุดภาพที่พิกัดศูนย์กลางของวินโดว์ก็ได้

-1	1
-1	1

(ก)

1	1
-1	-1

(ข)

ภาพที่ 2.17 วินโดว์ขนาด 2x2 ที่ใช้สำหรับหาค่ากราเดียนท์ (ก) สำหรับหาค่า G_x (ข) สำหรับหาค่า G_y

2.1.3.2 การหาขอบภาพโดยวิธีของโรเบิร์ต การหาขนาดของกราเดียนท์แบบไม่ต่อเนื่องโดยวิธีของโรเบิร์ตสามารถคำนวณจากสมการดังต่อไปนี้

$$\nabla g(x,y) = |g(x,y) - g(x+1,y+1)| + |g(x,y+1) - g(x+1,y)| \qquad (2.16)$$

จากสมการที่2.16 สามารถพิจารณาในรูปของการคอนโวลูชั่นข้อมูลภาพเชิงตัวเลขด้วยวินโดว์โดยวิธีของโรเบิร์ตจะใช้วินโดว์ขนาด 2x2 ดังภาพที่ 2.18 (ก) และ(ข) สำหรับการหากราเดียนท์ของภาพในทิศทาง x และ y ตามลำดับ

1	0
0	-1

(ก)

0	-1
1	0

(ข)

ภาพที่ 2.18 วินโดว์ขนาด 2x2 ที่ใช้สำหรับหาค่ากราเดียนท์โดยวิธีของโรเบิร์ต (ก) สำหรับหาค่า G_x (ข) สำหรับหาค่า G_y

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3.3 การหาขอบภาพโดยวิธีของโซเบล สำหรับการหาขนาดของกราเดียนท์แบบไม่ต่อเนื่องโดยวิธีของโซเบลสามารถคำนวณจากสมการดังต่อไปนี้

$$\nabla g(x,y) = |g(x-1,y+1) + c*g(x,y+1) + g(x+1,y+1) - g(x-1,y-1) + c*g(x,y-1) - g(x+1,y-1)| - |g(x-1,y-1) + c*g(x-1,y) + g(x-1,y+1) - g(x+1,y-1) + c*g(x+1,y) - g(x+1,y+1)|$$

สมการ2.17

โดยค่าคงที่ c = 2

จากสมการการหาขนาดกราเดียนท์ข้างต้น สามารถพิจารณาในรูปของการคอนโวลูชั่นข้อมูลภาพด้วยวินโดว์ โดยวิธีของโซเบลจะใช้วินโดว์ขนาด 3x3 ดังภาพที่ 2.19 (ก) และ(ข) สำหรับการหากราเดียนท์ของภาพในทิศทาง x และ y ตามลำดับ

-1	0	1
-2	0	2
-1	0	1

(ก)

-1	-2	-1
0	0	0
1	2	1

(ข)

ภาพที่ 2.19 วินโดว์ขนาด 3x3 ที่ใช้สำหรับหาค่ากราเดียนท์โดยวิธีของโซเบล (ก) สำหรับหาค่า G_x (ข) สำหรับหาค่า G_y

2.1.3.4 การหาขอบภาพโดยวิธีของพรีเวท สมการที่ใช้ในการคำนวณหาขนาดกราเดียนท์โดยวิธีของพรีเวทจะเป็นสมการเดียวกับสมการที่ใช้หาขนาดของกราเดียนท์โดยวิธีของโซเบล แต่ต่างกันตรงที่สมการของพรีเวท ค่าคงที่เท่ากับ 1 ดังนั้นวินโดว์ที่ใช้ในการหากราเดียนท์ของภาพ ทั้งในทิศทาง x และ y แสดงดังภาพที่2.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-1	0	1
-1	0	1
-1	0	1

(ก)

1	1	1
0	0	0
-1	-1	-1

(ข)

ภาพที่ 2.20 วินโดว์ขนาด 3x3 ที่ใช้สำหรับหาค่ากราเดียนท์โดยวิธีของพรีเวท (ก) สำหรับหาค่า G_x (ข) สำหรับหาค่า G_y

จากการหาขอบภาพโดยวิธีต่างๆที่กล่าวข้างต้น พบว่าวินโดว์ที่นำมาใช้คอนโวลูชั่นกับข้อมูลภาพในแต่ละวิธีจะมีค่าถ่วงน้ำหนักต่างไปจากการหากราเดียนท์ของสมการเริ่มต้นคือ สมการ 2.15 และ 2.16 ซึ่งมีค่าถ่วงน้ำหนักเพียง 2 ระดับเท่านั้น คือ (1,-1) ส่วนการหาขอบภาพโดยวิธีของโรเบิร์ตและพรีเวท จะใช้ค่าถ่วงน้ำหนักถึง 3 ระดับคือ (-1,0,1) และวิธีของโซเบลจะใช้ค่าถ่วงน้ำหนักถึง 5 ระดับ คือ (-2,-1,0,1,2)

2.1.3.5 การหาขอบภาพโดยใช้อนุพันธ์เชิงเส้นอันดับที่สอง

การหาขอบภาพโดยวิธีนี้อาศัยหลักการที่ว่า ที่ตำแหน่งของจุดภาพใดมีการเปลี่ยนแปลงระดับความเข้มอย่างทันทีทันใด และเมื่อหาอนุพันธ์เชิงเส้นอันดับที่สองที่ตำแหน่งจุดภาพนั้นจะพบว่ามีซีโร่ครอสซิ่งเกิดขึ้น (ดังภาพที่ 2.21)ขั้นตอนการหาขอบภาพที่อยู่บนพื้นฐานของหลักการดังกล่างมีอยู่หลายวิธีดังที่จะกล่าวต่อไปนี้

ตัวดำเนินการหาขอบภาพ ลาปลาเซี่ยน ของฟังก์ชัน 2 มิติ g(x,y) ซึ่งเป็นฟังก์ชันไม่ต่อเนื่องพิกัด (x,y) ได้มาจากการประมาณอนุพันธ์เชิงเส้นอันดับที่สองของฟังก์ชันต่อเนื่องดังสมการที่ 2.17

$$\nabla^2 f(x,y) = \partial^2 f/\partial x^2 + \partial^2 f/\partial y^2 \qquad (2.18)$$

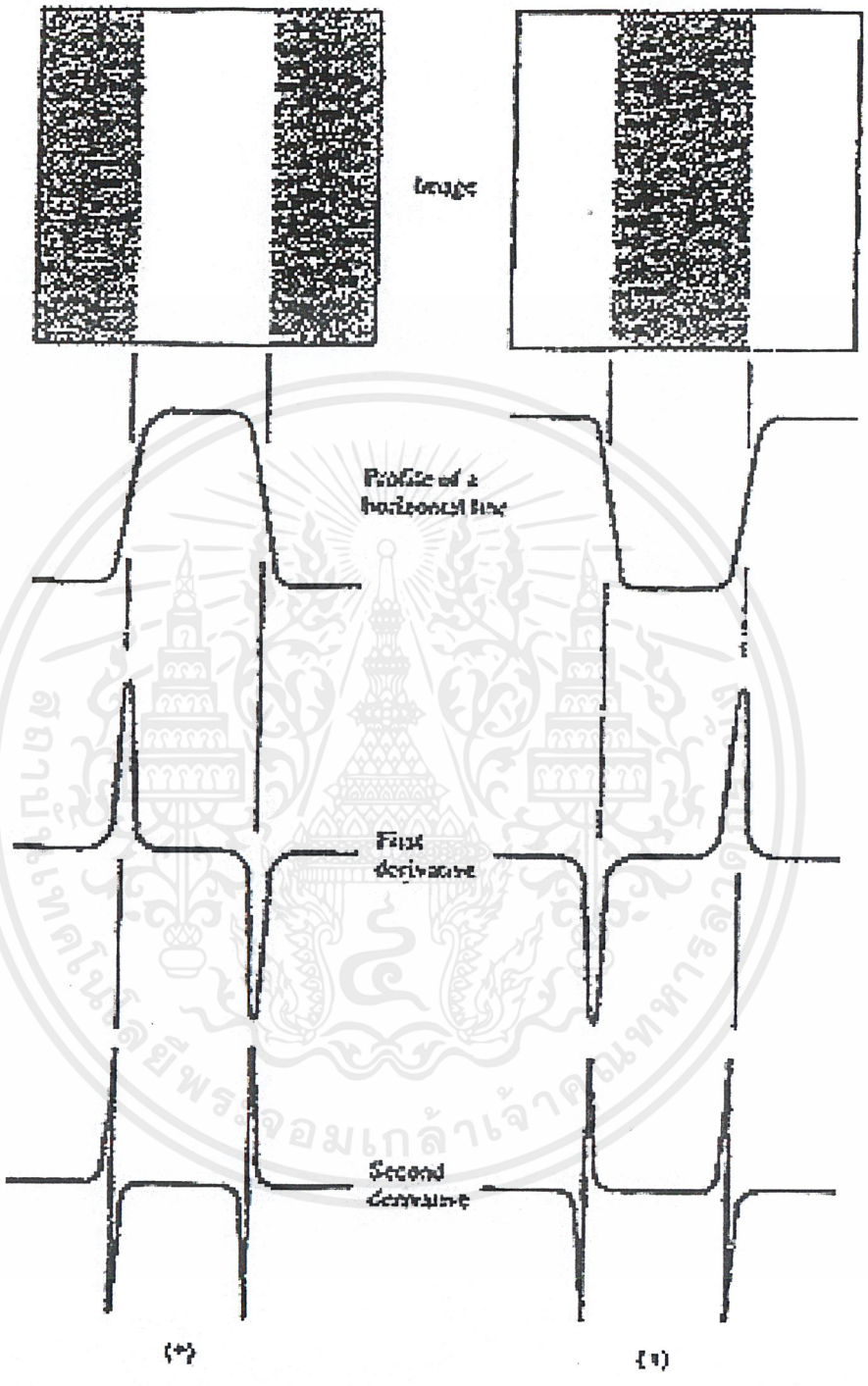
สำหรับในกรณีของ $\nabla^2 f(x,y)$ ที่ได้มาจากการประมาณสมการที่2.18 เพื่อใช้ในการหาขอบภาพในภาพเชิงตัวเลข สามารถกำหนดได้ดังนี้

$$\nabla^2 g(x,y) = 4g(x,y) - ((g(x+1,y) + g(x-1,y) + g(x,y+1) + g(x,y-1)) \qquad (2.19)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการที่2.19 สามารถพิจารณาในรูปของการคอนโวลูชั่นข้อมูลภาพเชิงตัวเลขด้วยวินโดว์ขนาด 3x3 โดยสัมประสิทธิ์ในวินโดว์ที่ใช้คูณอยู่ที่พิกัดที่ต้องการหาขอบภาพจะต้องมีค่าเป็นบวกเสมอ ส่วนสัมประสิทธิ์ในตำแหน่งอื่นๆอาจมีค่าเป็นบวกหรือลบก็ได้ เนื่องจากอนุพันธ์ของลาปลาเซี่ยน ผลรวมของสัมประสิทธิ์ทั้งหมดจะต้องเป็นศูนย์เสมอ ซึ่งในกรณีนี้ถ้าจุดภาพที่อยู่ในพื้นที่ขนาด 3x3 มีค่าเท่ากันหมด เมื่อนำคอนโวลูชั่นกับตัวดำเนินการนี้ผลลัพธ์ที่ได้ก็คือ ศูนย์ ซึ่งแสดงว่าจุดนี้ไม่ใช่ขอบภาพอย่างแน่นอนใน ภาพที่2.22 แสดงวินโดว์ของลาปลาเซี่ยนในรูปแบบต่างๆทั้งที่ใช้จุดภาพในบริเวณใกล้เคียง 4 จุด ดังรูปซึ่งจะมีผลของขอบภาพต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.21 แสดงความสัมพันธ์ระหว่างหาขอบภาพโดยใช้อนุพันธ์เชิงเส้นอันดับที่ 1 กับการหาขอบภาพโดยใช้อนุพันธ์เชิงเส้นอันดับที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0	-1	0
-1	4	-1
0	-1	0

(ก)

-1	-1	-1
-1	8	-1
-1	-1	-1

(ข)

1	-2	1
-2	4	-2
1	-2	1

(ค)

ภาพที่ 2.22 (ก) – (ค) แสดงตัวดำเนินการหาขอบภาพโดยวิธีของลาปาเซี่ยน

2.1.4 เทคนิคการหาขอบเขตของภาพ

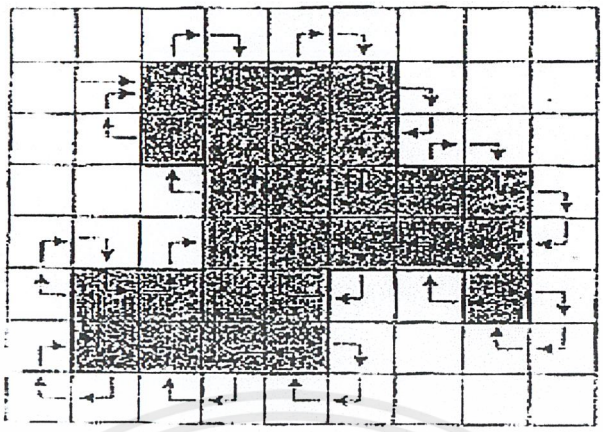
เทคนิคการหาขอบเขตของภาพ หรือเทคนิคการติดตามรอยขอบภาพ เป็นเทคนิคที่ใช้ในการหาขอบเขตหรือบริเวณของวัตถุที่สนใจที่ปรากฏอยู่ในภาพ โดยทั่วไปแล้วเทคนิคนี้จะใช้กับภาพไบนารี โดยที่จุดภาพที่มีค่าเป็น 1 แทนจุดดำหรือจุดที่เป็นส่วนของวัตถุและจุดภาพที่มีค่าเป็น 0 แทนจุดขาวหรือจุดที่เป็นช่องว่าง บนกระดาษหรือพื้นเบื้องหลัง

การทำงานของเทคนิคการติดตามรอยขอบของภาพ เป็นการเดินไต่ไปตามขอบระหว่างส่วนที่เป็นวัตถุกับส่วนที่เป็นพื้นหลัง โดยจะตรวจกวาดไปทุกๆ จุดภาพ โดยจะเริ่มจากจุดมุมซ้ายบนของข้อมูลภาพ ตรวจกวาดไปในทิศทางจากซ้ายไปขวาและเลื่อนจากบนลงล่าง เมื่อตรวจกวาดมาพบจุดภาพใดๆที่มีค่าของจุดภาพเป็น 1 ก็จะเปลี่ยนลักษณะการเคลื่อนที่ไปยังจุดภาพจุดถัดไปเสียใหม่ โดยมีเงื่อนไขการเคลื่อนที่ดังนี้

1. ถ้าจุดที่อยู่ปัจจุบันเป็นจุดของวัตถุหรือมีค่าของวัตถุเป็น 1 ให้เลี้ยวซ้าย แล้วก้าวเดินตรงไปข้างหน้าไปยังจุดถัดไป
2. ถ้าจุดที่อยู่ปัจจุบันเป็นจุดของพื้นหลังหรือมีค่าของจุดเป็น 0 ให้เลี้ยวขวา แล้วก้าวเดินตรงไปข้างหน้าไปยังจุดถัดไป
3. การเคลื่อนที่จะสิ้นสุดลง เมื่อจุดที่อยู่ปัจจุบันเป็นจุดเดียวกันกับจุดเริ่มต้น

สามารถแสดงลักษณะการทำงานของเทคนิคการหาขอบเขตของภาพได้ดังภาพที่2.23 ซึ่งจะแสดงการเคลื่อนที่ไปตามจุดต่างๆ ที่เป็นขอบของภาพ เริ่มจากจุดที่ถูกแรเงาไว้ซึ่งเป็นจุดของภาพจุดแรกที่ตรวจกวาดมาพบ การเคลื่อนที่จะเป็นไปตามเงื่อนไขที่กำหนด เมื่อการเคลื่อนที่วนกลับมาถึงจุดที่เป็นจุดเริ่มต้นก็จะทราบพิกัดที่เป็นขอบเขตของภาพได้ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.23 แสดงลักษณะการหาขอบเขตของภาพ

2.2 วิชวลเบสิก

วิชวลเบสิก เป็นภาษาที่ได้รับความนิยมนำมาใช้ในการพัฒนาโปรแกรมบนวินโดว์ เนื่องจากเป็นภาษาที่ใช้เทคโนโลยีในลักษณะวิชวลไลซ (Visualize) ซึ่งเพียงแต่เลือกคอนโทรล (Control) ที่เหมาะสมแล้ววางบนฟอร์ม (Form) ก็สามารถสร้างจอภาพใช้ติดต่อกับผู้ใช้รวมทั้งใช้เทคนิคการเขียนโปรแกรมเพื่อกำหนดการทำงานให้คอนโทรลต่างๆที่สร้างขึ้นตามเหตุการณ์ (Event) ที่เกิดขึ้นเช่นการเลื่อนเมาส์ การรับข้อมูลจากคีย์บอร์ด เป็นต้น โดยภาษาที่ใช้เขียนโปรแกรมจะเป็นภาษาเบสิก

รูปแบบโปรแกรมได้มีการเปลี่ยนแปลงนับตั้งแต่ระบบปฏิบัติการวินโดว์ได้ถูกใช้อย่างแพร่หลาย รูปแบบของการเขียนโปรแกรมได้เปลี่ยนไปกล่าวคือ เหตุการณ์ที่เกิดขึ้นในวินโดว์ไม่ได้เกิดขึ้นเป็นลำดับเหมือนเดิม แต่จะเกิดตามที่ผู้ใช้กำหนดขึ้นเอง เช่น ผู้ใช้อาจเลือกกดปุ่มที่ 2 ก่อนปุ่มที่ 1 ทั้งที่ผู้พัฒนาโปรแกรมตั้งใจจะกดปุ่มที่ 1 ก่อนเป็นต้น ด้วยเหตุนี้แนวความคิดในการเขียนโปรแกรมในรูปแบบเดิมจึงไม่สามารถรองรับความเสามารถที่เพิ่มขึ้นของระบบปฏิบัติการวินโดว์ได้ แนวความคิดในการเขียนโปรแกรมแบบ Event-Driven จึงเกิดขึ้น

แนวคิดในการเขียนโปรแกรมแบบอีเว็นท์-ไดเว็น จึงมีการเปลี่ยนมาสนใจเหตุการณ์ที่เกิดขึ้นในโปรแกรมมากกว่าการกำหนดขั้นตอนการทำงานของโปรแกรมแบบเดิม แต่อย่างไรก็ตามก็ยังคงต้องอาศัยแนวความคิดในการเขียนโปรแกรมแบบเดิมอยู่บ้าง เพราะการกำหนดการทำงานให้แต่ละอีเว็นท์ยังคงต้องกำหนดอย่างเป็นขั้นตอนอยู่ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Active X เป็นเทคโนโลยีการเขียนโปรแกรมที่สร้างขึ้นบนพื้นฐานของ COM (Component Object Model เป็นสถาปัตยกรรมที่นิยมใช้ในการกำหนดการติดต่อระหว่างอ็อบเจ็คในโปรแกรมต่างๆ) ที่ซึ่งยอมให้โปรแกรมที่ทำงานเกี่ยวข้องกับเครือข่าย เช่นโปรแกรมที่ทำงานอยู่บนอินเทอร์เน็ต และ World Wide Web ซึ่งสร้างขึ้นด้วยภาษาคอมพิวเตอร์ที่แตกต่างกันสามารถติดต่อกันได้

ดังนั้น Active X Control คือคอนโทรลที่ประกอบด้วยฟังก์ชันการทำงานเพื่อทำงานอย่างใดอย่างหนึ่งและสามารถนำไปใช้ในโปรแกรมต่างๆ ได้ โดยในวิชวลเบสิก แอ็คทีฟเอ็กซ์คอนโทรลที่เราเพิ่มเข้ามาจะนำมาไว้ใน Toolbox โดยการแอ็ดคอมโพเนนท์

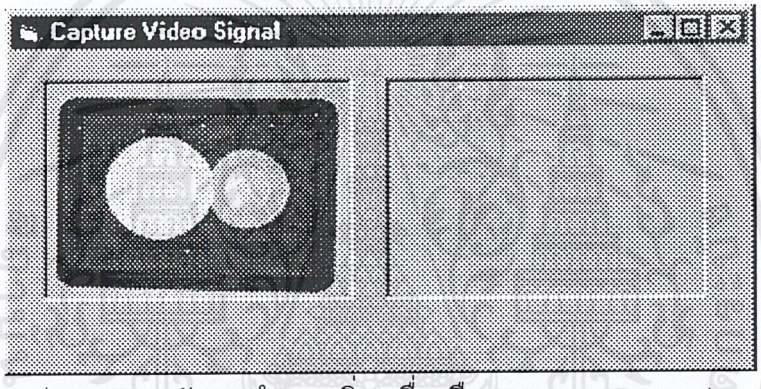
File .ocx เป็นไฟล์ที่ใช้เก็บแอ็คทีฟเอ็กซ์คอนโทรล ที่เรานำมาใช้งานในวิชวลเบสิก แต่จะแตกต่างจากคอนโทรลมาตรฐานที่เราได้ศึกษามา เนื่องจากคอนโทรลมาตรฐานทั่วๆ ไปไม่ได้เก็บอยู่ในไฟล์ประเภทนี้ โดยไฟล์ .ocx 1 ไฟล์อาจมีแอ็คทีฟเอ็กซ์คอนโทรลมากกว่า 1 คอนโทรล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
หลักการของเครื่องคิดราคาอาหาร

3.1 ส่วนติดต่อกับกล้องวีดีโอ

ในการรับภาพเข้ามาจากภายนอก จะใช้ภาพที่ได้มาจากสัญญาณวีดีโอ ซึ่งจะใช้กล้องวีดีโอรับภาพผ่านเข้ามาทาง วีดีโอการ์ด (Video card) และเข้ามาใน โปรแกรม ในที่นี้จะใช้คอมโพเนนท์ของโปรแกรมวิชวลเบสิก โดยการเพิ่มเครื่องมือ (Add Component) เป็นไฟล์ซนิด .ocx ชื่อ Ezvidcap.ocx

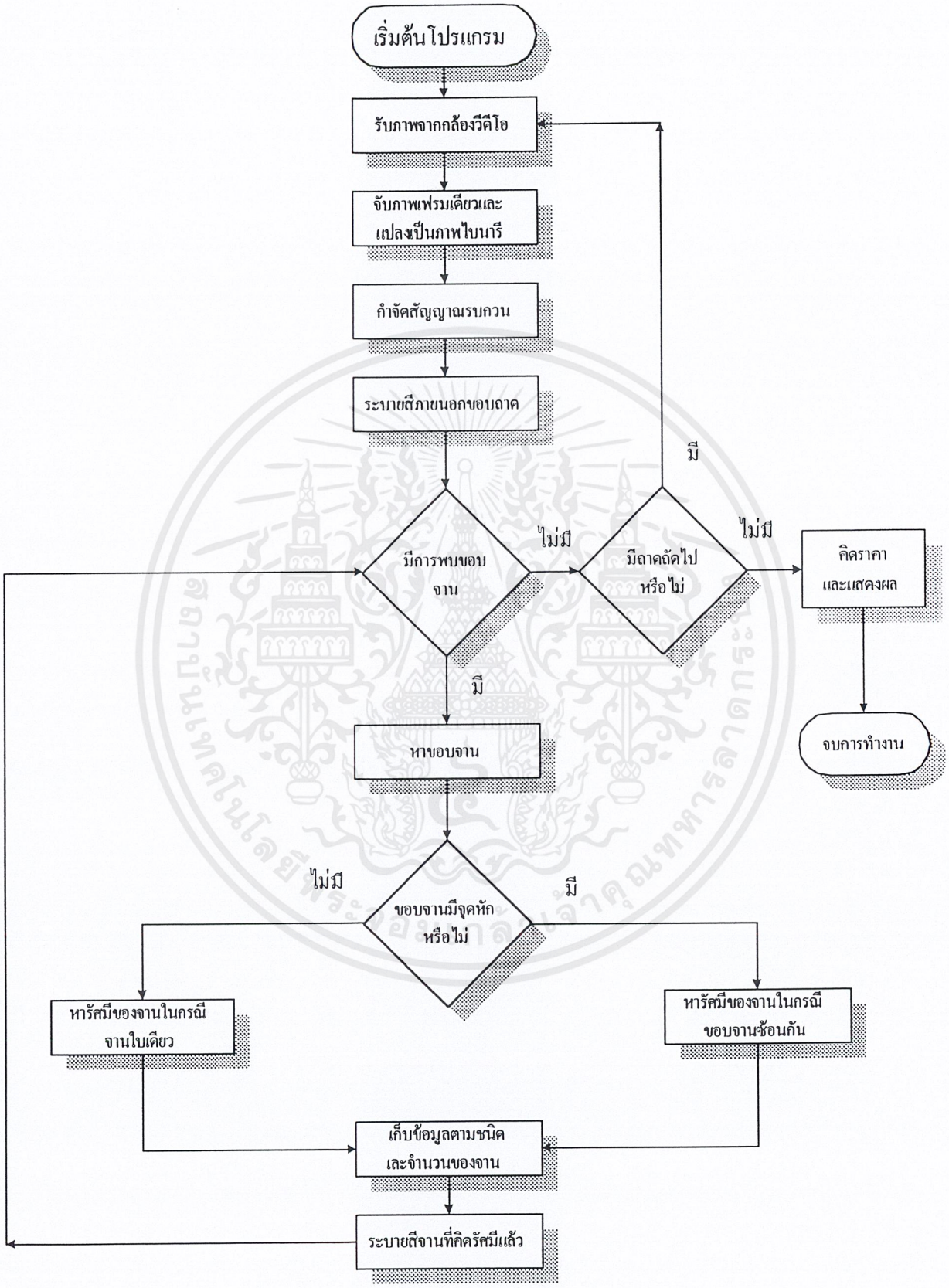


รูป 3.1 ภาพหลังจากทำการเพิ่มเครื่องมือ (Add component)

3.2 หลักการทำงานของโปรแกรม

เมื่อรับภาพเข้ามาได้แล้ว ขั้นตอนต่อไปคือจับภาพให้อยู่นิ่ง แล้วแปลงภาพเป็นภาพไบนารี เพื่อให้ง่ายต่อการประมวลผล จากนั้นจะกำจัดสัญญาณรบกวน และสิ่งที่อยู่ภายนอกถาดออกไป เพื่อให้เหลือแต่จานแล้วจึงทำการหาขอบจานทีละใบ โดยการสแกนจากซ้ายไปขวา และจากบนลงล่างจนกระทั่งพบกับจุดแรกที่เปลี่ยนจากสีดำ -> ขาว นั่นคือขอบจานและวนจนครบรอบได้เป็นเส้นรอบวงของจาน นำเส้นรอบวงมาคำนวณหาจุดศูนย์กลาง เพื่อนำมาหารัศมีของจาน ทำให้รู้ว่าเป็นจานขนาดใดแล้วเก็บค่าไว้ และระบายสีจานที่ได้ทำการนับแล้วให้เป็นสีดำ เพื่อการนับรอบถัดไปไม่ให้ซ้ำใบเดิม ทำซ้ำจนครบทั้งถาด นำมาคิดราคา และแสดงราคาออกทางจอภาพ แสดงแผนผังการทำงานของโปรแกรม ดังรูป 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.2 แผนผังการทำงานของโปรแกรมทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 จับภาพเฟรมเดียว และแปลงภาพเป็นไบนารี

ในการประมวลผลจะใช้ภาพเพียงภาพเดียว เนื่องจากภาพที่รับเข้ามาจากกล้องวีดีโอเป็นภาพจริงที่เคลื่อนไหว จึงต้องทำการจับภาพนิ่งเพียงเฟรมเดียวจากภาพเคลื่อนไหว

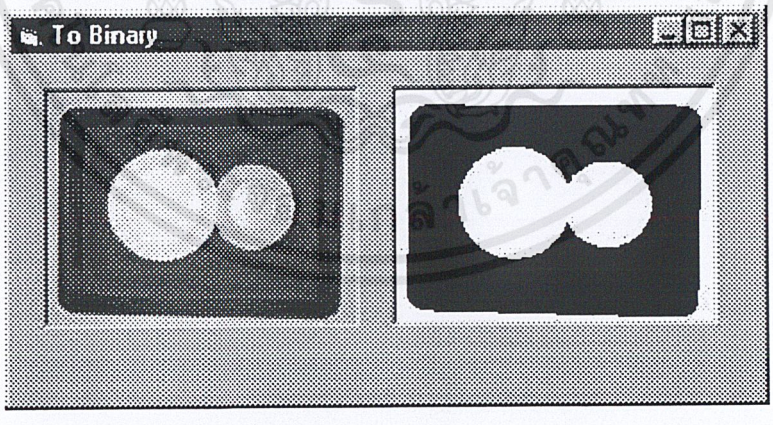
หลังจากจับภาพนิ่งได้แล้ว เพื่อให้ภาพเหลือเพียงสีที่เราสนใจ และสามารถแยกความแตกต่างของจานกับถาดได้ชัดเจนยิ่งขึ้นโดยทำการแปลงภาพเป็นไบนารีให้เหลือเพียง 2 สี คือ ขาว และดำ โดยในที่นี้สีขาวจะเป็นจาน และสีดำจะเป็นถาด ซึ่งมีขั้นตอนดังนี้

1.เลือกค่าเทรชโฮลด์ ที่สามารถแยกแยะกลุ่มของสีขาว และสีดำ ได้ชัดเจน

2.นำค่าสีของภาพจริงแต่ละจุด มาเปรียบเทียบกับค่าเทรชโฮลด์ โดยถ้าค่าสีของภาพจริงมากกว่าค่าเทรชโฮลด์จะให้ค่าไบนารีเป็น 1 (สีดำ) และถ้าน้อยกว่าหรือเท่ากับค่าเทรชโฮลด์ จะให้ค่าไบนารีเป็น 0 (สีขาว)

3.เก็บค่าไบนารีทั้งหมดในหน่วยความจำใหม่ ตามตำแหน่งของสีแต่ละจุด

ในการเก็บภาพนั้นจะเก็บสีของภาพแต่ละจุดในอาร์เรย์ 2 มิติ คือ A(X , Y)
ตัวอย่างเช่น A(10 , 20) = 1 จะหมายถึง ค่าสีของภาพที่ตำแหน่งคู่อันดับ (10 , 20) มีค่าเป็นสีดำ เป็นต้น



รูป 3.3 การแปลงเป็นภาพไบนารี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 การระบายสีภายนอกขอบถาด

ในบางกรณีภายนอกขอบถาดเป็นสีขาว ทำให้ในการสแกนหาจานสีขาวเกิดการผิดพลาด เพราะพบกับสีขาวภายนอกขอบถาด จึงแก้ไขโดยการระบายสีภายนอกขอบถาดให้เป็นสีดำทั้งหมด เพื่อให้การสแกนผ่านสีขาวภายนอกขอบถาดเข้าไปภายในถาดได้

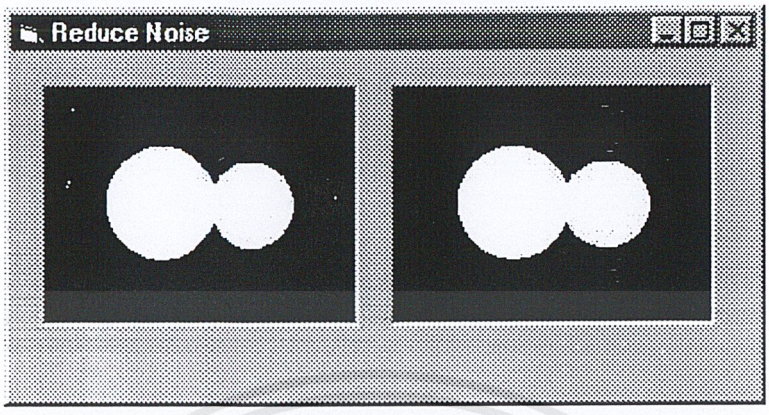
รูป 3.4 การระบายสีภายนอกขอบถาด

3.2.3 การกำจัดสัญญาณรบกวน

เนื่องจากในการประมวลผลอาจมีจุดสีดำไปปรากฏในกลุ่มของจุดสีขาวหรืออาจมีจุดสีขาวไปปรากฏในจุดสีดำจึงตัวมีการกำจัดจุดที่ไม่พึงประสงค์นั้นออกไป โปรแกรมจะถูกกำหนดไว้ว่าสีขาวที่พบนั้นเป็นขอบจาน ดังนั้นหากมีสีขาวที่ไม่ใช่จานเกิดขึ้นจะทำให้การประมวลผลเกิดการผิดพลาดขึ้นได้ จึงต้องกำจัดสีขาวที่ไม่เกี่ยวข้องนี้ออกไปหรืออาจมีจุดสีดำไปปรากฏในส่วนที่เป็นจานก็จะต้องกำจัดจุดสีดำออกไป ซึ่งก็คือ การกำจัดสัญญาณรบกวนนั่นเอง

ในการกำจัดสัญญาณรบกวนนั้นจะทำการตรวจสอบสีขาวทุกจุดในภาพ กับกลุ่มของสีรอบข้าง รวมทั้งตรวจสีดำทุกจุด กับกลุ่มสีรอบข้าง แล้วตัดสินใจว่าสีขาวหรือสีดำที่เกิดขึ้นนั้น เป็นสัญญาณรบกวนหรือไม่ ถ้าเป็นจะกำจัดโดยเปลี่ยนสีขาวจุดนั้นให้เป็นสีดำ หรือเปลี่ยนสีดำจุดนั้นให้เป็นสีขาว

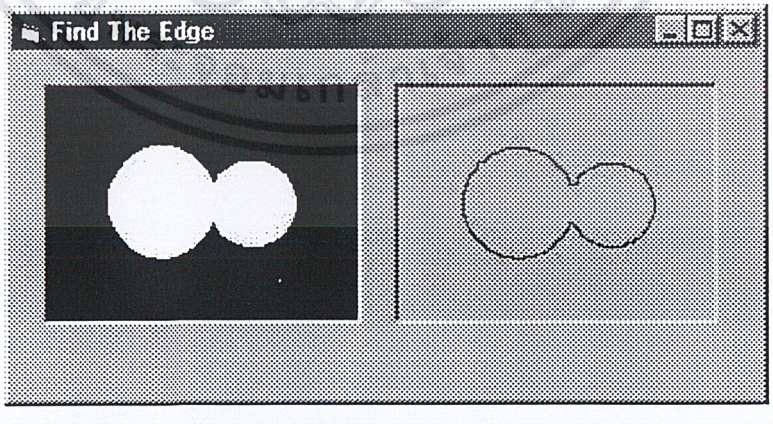
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.5 การกำจัดสัญญาณรบกวน

3.2.4 การหาขอบจาน

การหาขอบจานจะเริ่มตั้งแต่การสแกนหาสีขาวจุดแรก โดยในการสแกนนั้นจะเริ่มสแกนจากมุมซ้ายบนและจบที่มุมขวาล่าง หรือจากซ้าย->ขวา และจากบน->ล่าง เมื่อพบสีขาวจุดแรกแล้ว จะใช้จุดนั้นเป็นจุดเริ่มต้นในการวนหาขอบโดยพิจารณาจากจุดที่เป็นรอยต่อของสีดำและขาวไปเรื่อยๆ พร้อมทั้งเก็บค่าคู่อันดับรอยต่อนั้นจนครบรอบในทิศทางตามเข็มนาฬิกาซึ่งจะได้ขนาดของจานออกมา ซึ่งปกติจุดแรกที่พบจะเป็นจุดบนสุดของจาน



รูปที่ 3.6 การหาขอบจาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 การหาจุดหักของขอบจาน

เพื่อให้โปรแกรมรู้ว่าลักษณะของจานที่รับเข้ามาได้นั้นเป็นจานที่เป็นใบเดียว หรือจานที่มีขอบติดกันหลายใบ เพราะ ในแต่ละกรณีจะแยกการประมวลผลออกจากกัน โดยอาศัยหลักการพิจารณาทิศทางของเส้นตรง 2 เส้น ในทิศทางตามเข็มนาฬิกา ไล่ไปตามขอบจานที่เก็บค่ามาได้ดังนี้

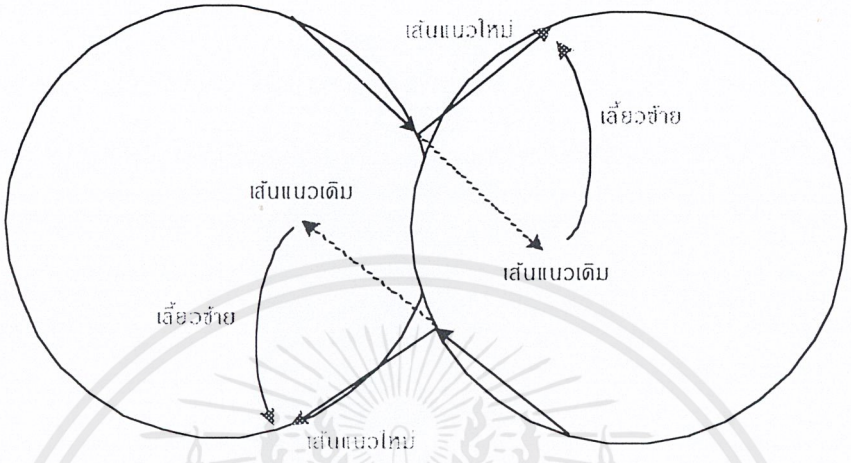
กรณี 1 ไม่มีจุดหัก

รูป 3.7 กรณีไม่มีจุดหัก

การที่เส้นแนวใหม่ เอนไปทางขวาจากเส้นแนวเดิมแสดงว่า ไม่มีจุดหักเกิดขึ้น ดังนั้นสรุปได้ว่าเส้นขอบที่ได้มาเป็นเส้นขอบที่เกิดจากจานใบเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

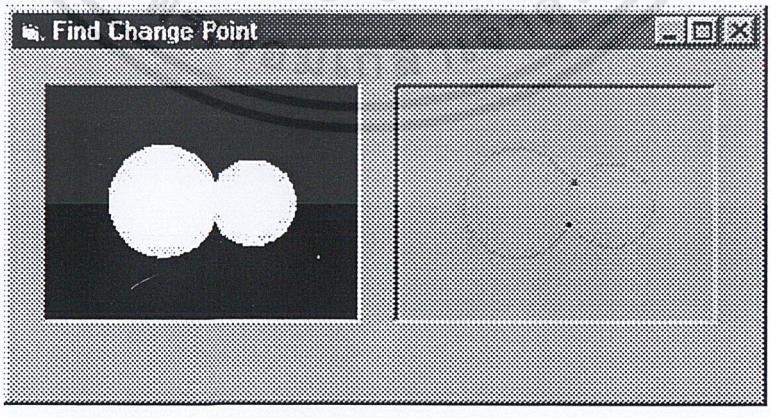
กรณีที่ 2 มีจุดหักเกิดขึ้น



รูป 3.8 กรณีมีจุดหักเกิดขึ้น

การที่เส้นแนวใหม่ เอนไปทางซ้ายจากเส้นแนวเดิมแสดงว่า มีจุดหักเกิดขึ้น ดังนั้นสรุปได้ว่าเส้นขอบที่ได้มาเป็นเส้นขอบที่เกิดจากจานหลายใบมีขอบซ้อนกัน

ในการเขียนโปรแกรมจะใช้หลักการนี้แต่จะมีความละเอียดมากกว่านี้ และหลังจากที่รู้ว่ามีจุดเปลี่ยนเกิดขึ้น ก็จะมีการเก็บค่าของจุดเปลี่ยนเหล่านั้นเพื่อใช้ในการประมวลผลต่อไป



รูป 3.9 การหาจุดหัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.6 การหารัศมีของจานกรณีจานใบเดียว

หลังจากทราบว่าไม่มีจุดหักเกิดขึ้น แสดงว่าเป็นจานใบเดียว ในการคิดหารัศมีจะคิดจากการนำค่าของตำแหน่งแถว (Y) ของจุดบนสุด(จุดที่พบสีขาวครั้งแรก) กับจุดล่างสุด(จุดสุดท้ายหารสอง) ซึ่งจะเป็นตำแหน่งที่อยู่ตรงข้ามกัน มาลบกันจะได้เป็นเส้นผ่านศูนย์กลาง และนำค่าที่ได้มาหารสองจะได้เป็นรัศมีออกมา

ตัวอย่างเช่น A(40 , 80) กับ A(40 , 120) เป็นด้านตรงข้ามกัน

รูป 3.10 การหารัศมีของจานใบเดียว

ดังนั้น รัศมี = (Y2 – Y1) / 2 = (120 – 80) / 2 = 20

3.2.6 การหารัศมีของจานกรณีขอบจานซ้อนกัน

ในกรณีจานซ้อนกันมีหลายรูปแบบมาก เช่น ซ้อนกันแนวนอน แนวทแยง เป็นกลุ่ม ซ้อนกัน 2 ใบ 3 ใบ หรือมากกว่านั้น ซึ่งไม่มีรูปแบบตายตัว ทำให้ไม่สามารถหารัศมีด้วยวิธีของจานใบเดียวได้ จึงใช้วิธีทางคณิตศาสตร์เข้ามาช่วยโดย การรู้จุด 3 จุดบนเส้นรอบวงสามารถหาจุดศูนย์กลางวงกลมได้จากสมการวงกลม

$$y^2 + x^2 + Dx + Ey + F = 0 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการสุ่มตำแหน่ง (x,y) บนเส้นรอบวง 3 ค่า คือ (x1,y1),(x2,y2), (x3,y3) แทนค่าในสมการข้างต้นจะได้

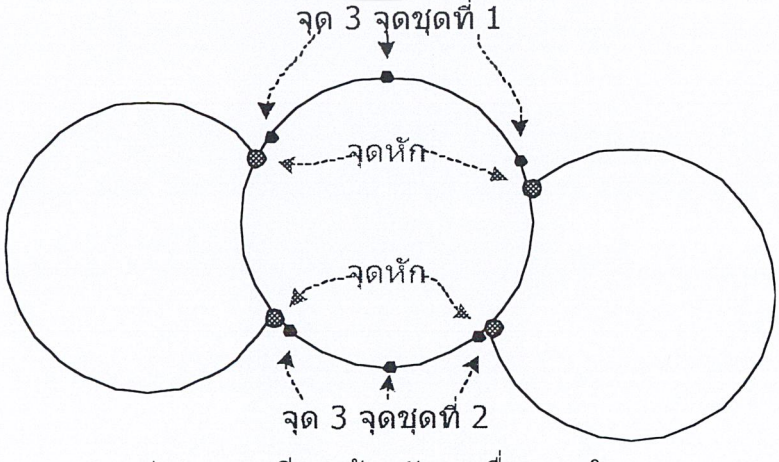
$$\begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix} \begin{bmatrix} D \\ E \\ F \end{bmatrix} = \begin{bmatrix} x_1^2 + y_1^2 \\ x_2^2 + y_2^2 \\ x_3^2 + y_3^2 \end{bmatrix}$$

$$D = \frac{\begin{vmatrix} -(x_1^2 + y_1^2) & y_1 & 1 \\ -(x_2^2 + y_2^2) & y_2 & 1 \\ -(x_3^2 + y_3^2) & y_3 & 1 \end{vmatrix}}{\begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}} \qquad E = \frac{\begin{vmatrix} x_1 & -(x_1^2 + y_1^2) & 1 \\ x_2 & -(x_2^2 + y_2^2) & 1 \\ x_3 & -(x_3^2 + y_3^2) & 1 \end{vmatrix}}{\begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}}$$

$$h = -D/2 \quad , \quad k = -E/2$$

(h,k) คือ พิกัดของจุดศูนย์กลางจากจุด 3 จุดบนเส้นรอบวง

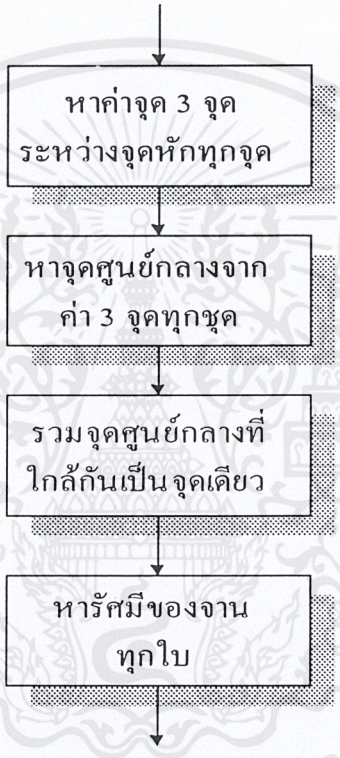
วิธีการหารัศมีของจานหลายใบจะเริ่มจากการหาค่าจุด 3 จุดระหว่างจุดหัก แล้วนำมาหาค่าจุดศูนย์กลางของวงกลม ดังนั้นบางครั้งวงกลมหนึ่งวงอาจมีจุดศูนย์กลางหลายจุดถ้าหากจานใบนั้นมีขอบซ้อนกับจานใบอื่นหลายใบ เช่น



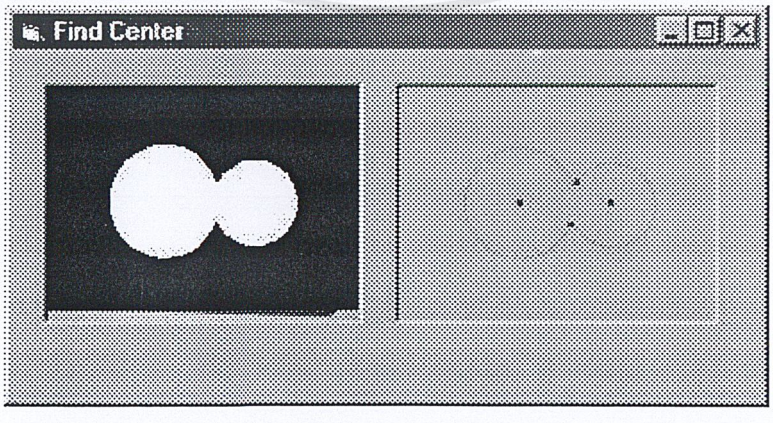
รูป 3.11 จานมีขอบซ้อนกับจานอื่นหลายใบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จานมีขอบซ้อนกับจานอื่นหลายใบ จะทำให้มีจุด 3 จุดเกิดขึ้นมากกว่า 1 ชุดจากรูป 3.11 เกิดขึ้น 2 ชุด และจะเกิดจุดศูนย์กลางขึ้น 2 จุดที่ใกล้เคียงกันในจาน 1 ใบ ซึ่งถ้าคิดค่ารัศมีของจานเลยจะได้รัศมีของจานใบนั้นมา 2 ค่า ทำให้เกิดการผิดพลาดขึ้น ดังนั้นในการคิดจุดศูนย์กลาง 1 จุดต่อจาน 1 ใบ จะต้องทำให้จุดศูนย์กลางที่อยู่ใกล้เคียงกันรวมกันเป็นจุดเดียวโดยการหาค่าเฉลี่ยก่อน แล้วจึงหาค่ารัศมีของจานแต่ละใบจากระยะห่างระหว่างจุดศูนย์กลางและเส้นรอบวงของจานใบนั้นๆต่อไป



รูป 3.12 แผนผังการทำงานของการหารัศมีของจานกรณีขอบจานซ้อนกัน

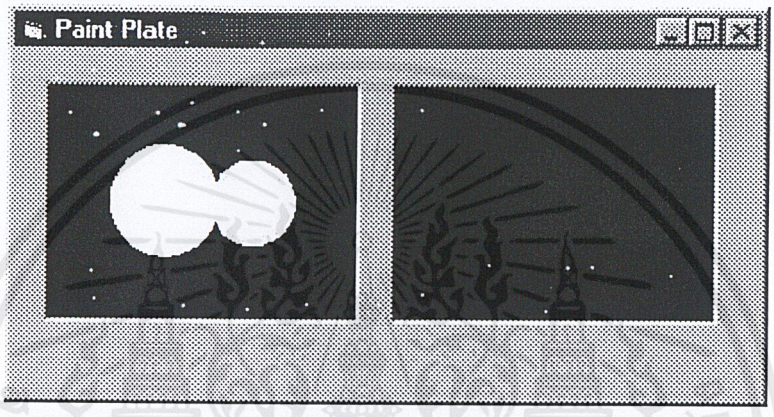


รูป 3.13 การหารัศมีของจานกรณีขอบจานซ้อนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.7 การระบายสีจานที่นับแล้ว

เนื่องจากการนับจานใบใหม่จะเริ่มทำการสแกนหาสีขาวใหม่ ดังนั้นเพื่อไม่ให้วนกลับไปนับจานใบเดิมจึงต้องทำให้การนับรอบใหม่ข้ามจานใบเดิมไป โดยการระบายสีภายในจานทั้งหมดที่นับแล้วให้เป็นสีดำ



รูป 3.14 การระบายสีจานที่นับแล้ว

3.2.8 การแยกชนิดจานและคิดราคา

หลังจากที่ได้รัศมีของจานทุกใบในถาดมาแล้ว ขั้นตอนต่อไปคือนำค่ารัศมีทั้งหมดที่ได้จากการแยกชนิดจานและคิดราคาทั้งหมดออกมา โดยจะแบ่งช่วงแต่ละช่วงให้มีระยะห่างพอสมควรเพื่อให้เกิดข้อผิดพลาดน้อยที่สุด

หากในกรณีที่ไม่สามารถกำจัดสัญญาณรบกวนได้ทั้งหมด ซึ่งอาจเหลืออยู่เล็กน้อย สัญญาณรบกวนเหล่านั้นก็จะไม่มีผลเพราะมีการป้องกันในส่วนของการแยกแยะขนาดจาน ซึ่งสัญญาณรบกวนนั้นจะมีค่าเล็กกว่าจานจริง จึงไม่อยู่ในช่วงที่จะนำมาคิดราคา

ช่วงที่ใช้ในการแยกแยะจานเป็น ดังนี้

รัศมี 10 – 17 พิกเซล เป็นจานขนาด A ราคา 5 บาท

รัศมี 18 – 21 พิกเซล เป็นจานขนาด B ราคา 10 บาท

รัศมี 22 – 27 พิกเซล เป็นจานขนาด C ราคา 15 บาท

รัศมี 28 – 34 พิกเซล เป็นจานขนาด D ราคา 20 บาท

รัศมี 35 – 45 พิกเซล เป็นจานขนาด E ราคา 25 บาท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.9 การเก็บข้อมูลในการคิดราคาอาหาร

ถาดอาหารที่ได้ทำการคิดราคาแล้ว จะถูกเก็บข้อมูลไว้ ซึ่งจะมีข้อมูลในส่วนของ จำนวนของจานแต่ละชนิด ราคาของอาหารในถาด วันที่ และ เวลา ที่ได้ทำการคิดราคา เพื่อให้สามารถเข้ามาตรวจสอบข้อมูลภายหลังได้

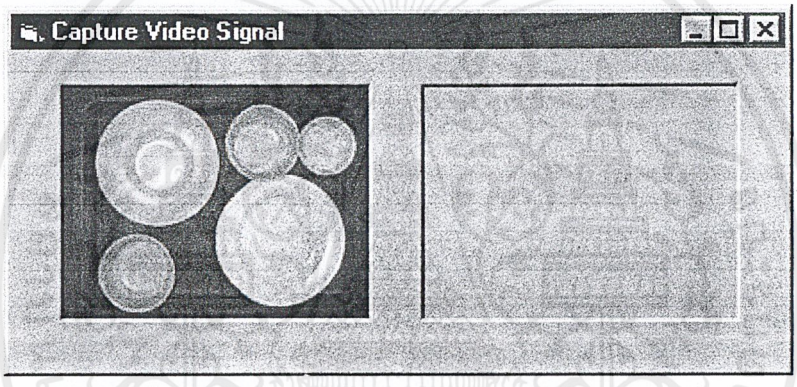
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการทดลอง

ในการแสดงให้เห็นถึงผลการทดลองจะแสดงอ้างอิงจากการทำงานของกรณีตัวอย่าง 1 กรณี ทีละขั้นตอนโดยจะแบ่งเป็น

4.1 การรับภาพเข้ามาในโปรแกรม

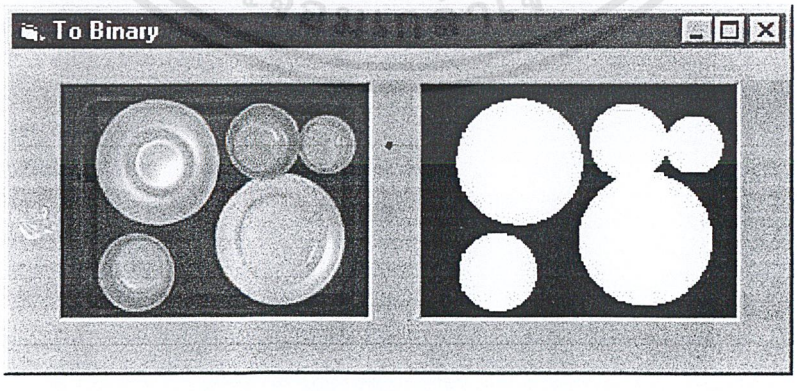
สามารถรับภาพเข้ามาในโปรแกรมได้ จากการเพิ่มคอมโพแนนท์ Ezvidcap.ocx ซึ่งมีการระบุขนาดของภาพไว้แล้ว เป็น 120 x 160 พิกเซล ได้ผลดังนี้



รูป 4.1 การรับภาพเข้ามาในโปรแกรม

4.2 การแปลงภาพเป็นไบนารี

จากการเลือกค่าเทรชโฮลโดยการกำหนดค่าเองได้ผลดังรูป

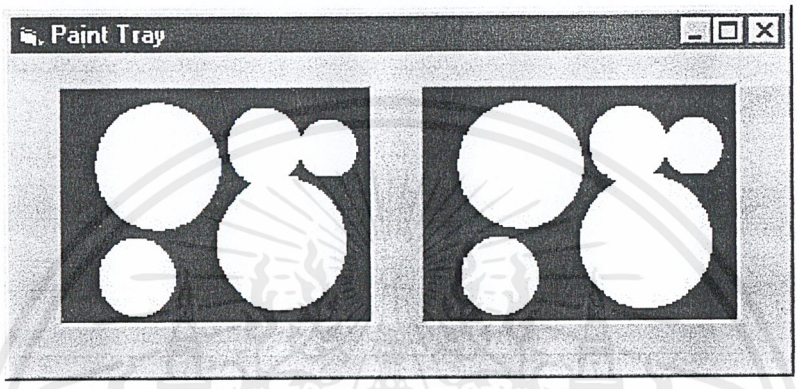


รูป 4.2 การแปลงภาพเป็นไบนารี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การระบายสีภายนอกขอบถาด

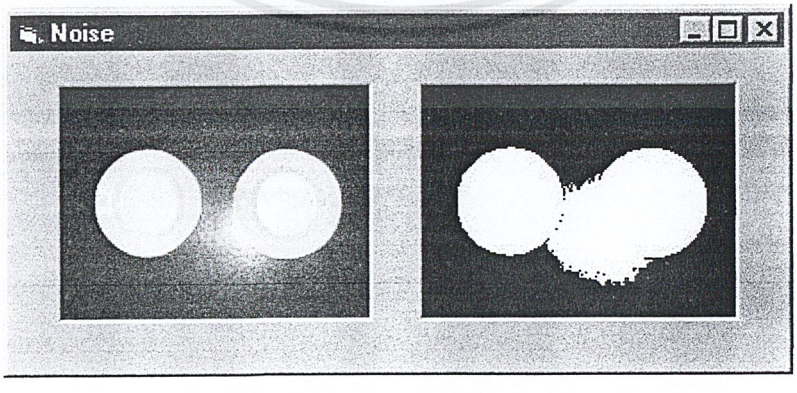
โดยปกติ ตัวเครื่องคิดราคาอาหารจะมีฉากหลังที่เป็นสีดำอยู่แล้ว ดังนั้นโดยปกติจึงไม่จำเป็นต้องมีการระบายสีภายนอกขอบถาด แต่เพื่อกันความผิดพลาดบางครั้งหากมีการกระแทกกล้อง ทำให้ระยะภาพเกินฉากหลังไปทำให้เห็นเป็นสีขาวเกิดขึ้นภายนอกถาด จึงต้องระบายสีภายนอกถาดให้เป็นสีดำทั้งหมดได้ผลดังรูป



รูป 4.3 การระบายภายนอกขอบถาด

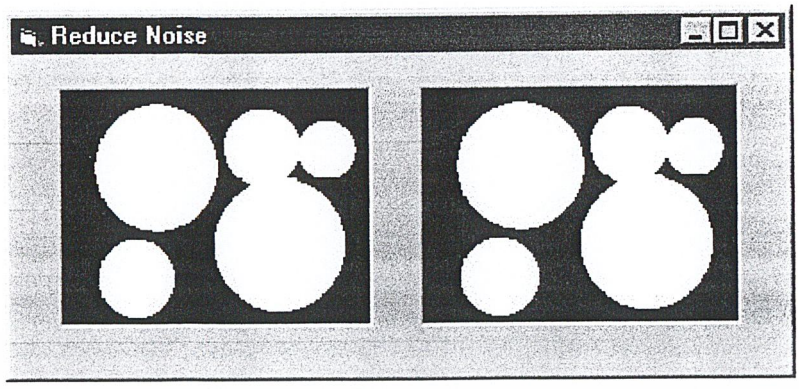
4.4 การกำจัดสัญญาณรบกวน

สิ่งที่มีผลมากในการเกิดสัญญาณรบกวนคือ แสงสะท้อนที่ตกลงบนถาดสีดำ ดังรูปที่ 4.4 ทำให้ภาพที่รับเข้ามาเห็นเป็นสีขาวบริเวณที่แสงตกกระทบ ทำให้การประมวลผลผิดพลาด จึงแก้ไขโดยการใช้ถาดและฉากหลังที่มีพื้นผิวที่ด้านทำให้แสงตกกระทบลงมาที่พื้นผิวได้ไม่มาก และเลือกค่าเทรชโฮลด์ที่หนักไปโทนสีดำมากขึ้น จึงสามารถกำจัดผลกระทบที่เกิดจากแสงที่มาตกกระทบออกไปได้ และจากการใช้การกำจัดสัญญาณรบกวนด้วยวิธีของ Kfill-Algorithm ได้ผลดังรูป 4.5



รูป 4.4 ผลของสัญญาณรบกวนที่เกิดจากแสงภายนอก

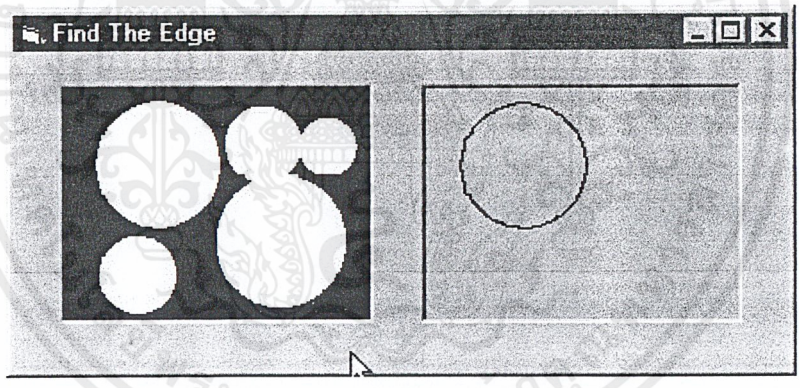
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.5 การกำจัดสัญญาณรบกวนเมื่อใช้ถาดที่ไม่สะท้อนแสงและปรับค่าเทรชโฮลด์

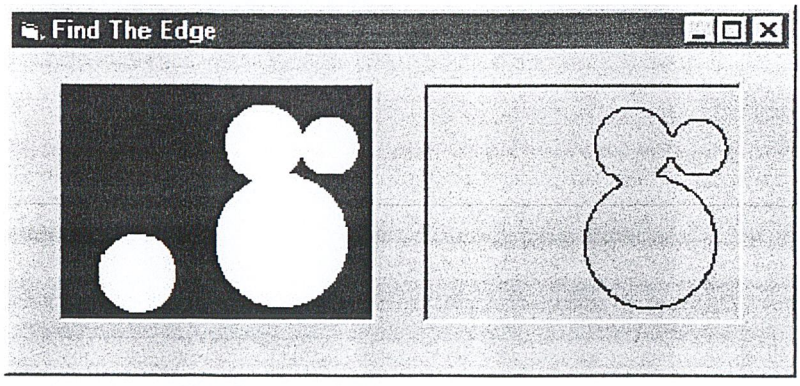
4.5 การหาขอบจาน

ใช้การหาขอบจานโดยเทคนิคการตามรอยขอบภาพ ซึ่งในรูปตัวอย่างจะแบ่งการคิดเป็น 3 ชุด ได้ผลดังรูป

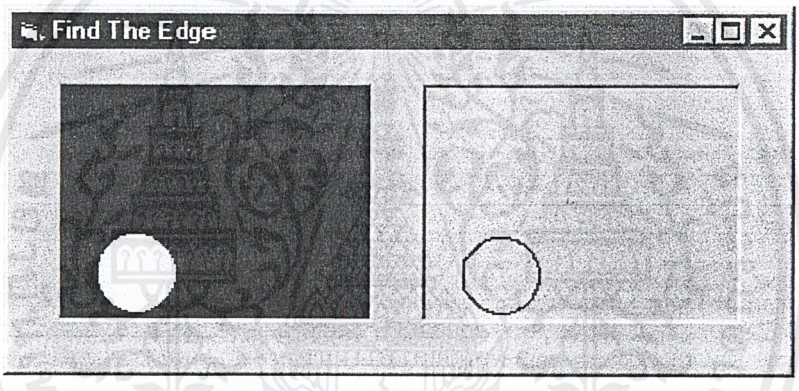


รูป 4.6 ขอบจานของจานชุดแรก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



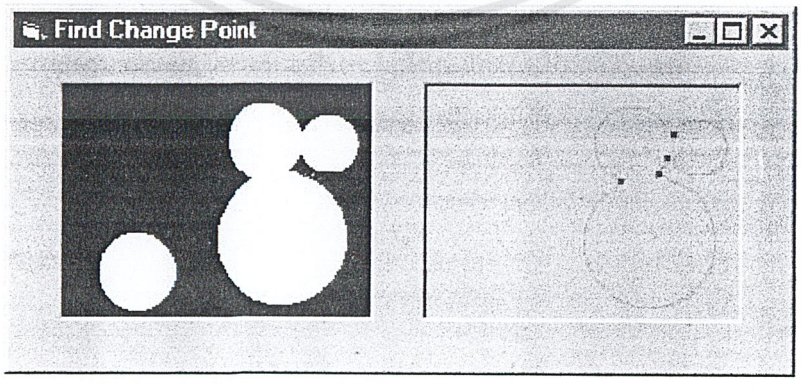
รูป 4.7 ขอบจานของจานชุดที่สอง



รูป 4.8 ขอบจานของจานชุดที่สาม

4.6 การหาจุดหักของขอบจาน

จากหลักการพิจารณาเส้นแนวเดิม กับเส้นแนวใหม่พบจุดหักในจานชุดที่สอง ดังรูป

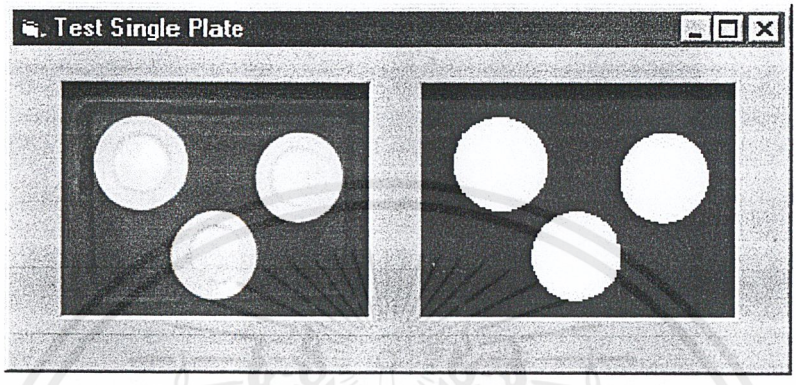


รูป 4.9 การหาจุดหักของจาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7 การหารัศมีกรณีจานใบเดียว

จากการทดลองวางจานขนาดเดิม ณ จุดต่างๆของถาด ได้ค่ารัศมีออกมาเท่ากัน แสดงการทดลองได้ดังรูป



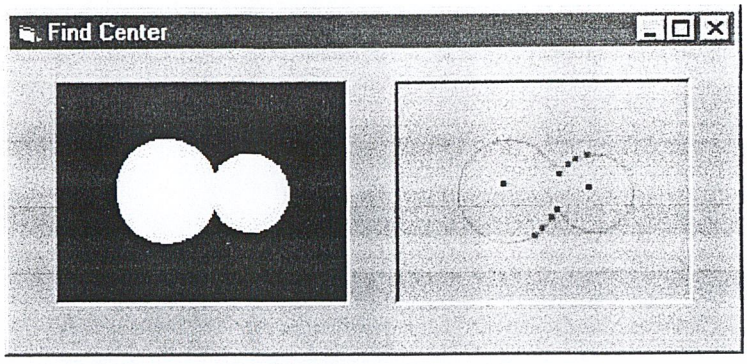
รูป 4.10 การหารัศมีของจานกรณีจานใบเดียว

4.8 การหารัศมีกรณีจานหลายใบมีขอบซ้อนกัน

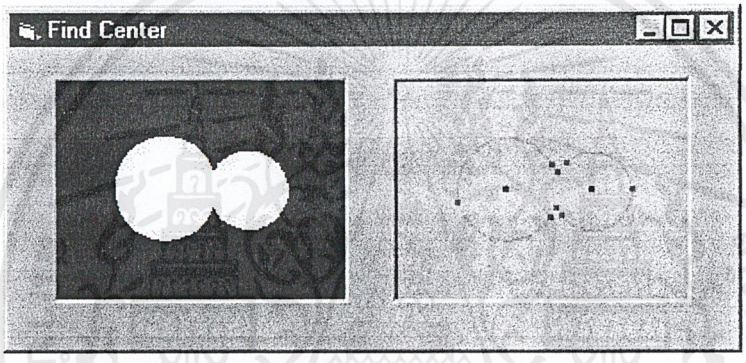
จากรูป 4.11 การหาจุดศูนย์กลางของจานกรณีขอบจานซ้อนกัน จะเห็นว่าถ้าหากเขียนโปรแกรมให้จุด 3 จุดกินส่วนโค้งแคบจะมีความคลาดเคลื่อนในการหาจุดศูนย์กลางเกิดขึ้น และจากรูป 4.12 หากให้จุด 3 จุดกินส่วนโค้งมากขึ้นการหาจุดศูนย์กลางจะมีความถูกต้องยิ่งขึ้น

จากการทดลองความคลาดเคลื่อนจะเกิดมากถ้าหากจุดสามจุดกินระยะส่วนโค้งที่แคบ ดังนั้นในโปรแกรมพยายามเลือกระยะห่างระหว่างจุด 3 จุดให้มากที่สุด เพื่อให้การหาจุดศูนย์กลางถูกต้องที่สุด และกำหนดเงื่อนไขระยะระหว่างจุดหัก 2 จุดว่าถ้ามีระยะห่างน้อยกว่าช่วงช่วงหนึ่งที่กำหนด(ได้จากการทดลอง) ให้ข้ามช่วงนั้นไป และรวมจุดศูนย์กลางที่อยู่ในระยะใกล้เคียงกันไว้ด้วยกันโดยการหาค่าเฉลี่ยของแต่ละจุด ทำให้จุดศูนย์กลางที่ได้ใกล้เคียงความจริงมากขึ้น

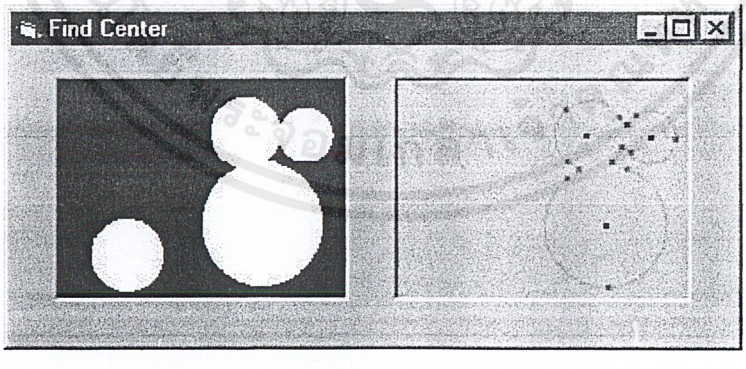
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.11 การจุดศูนย์กลางเพื่อหารัศมีของจานเมื่อช่วงของจุด 3 จุดแคบ



รูป 4.12 การจุดศูนย์กลางเพื่อหารัศมีของจานเมื่อช่วงของจุด 3 จุดกว้าง

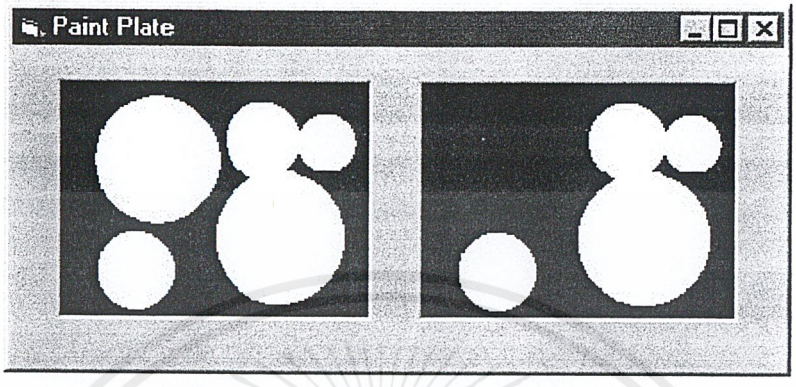


รูป 4.13 การจุดศูนย์กลางเพื่อหารัศมีของจานกรณีขอบจานซ้อนกัน

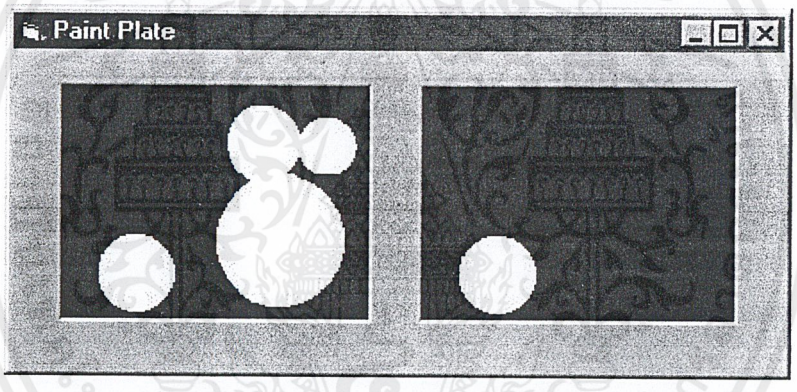
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.9 การระบายสีจานที่คิดราคาแล้ว

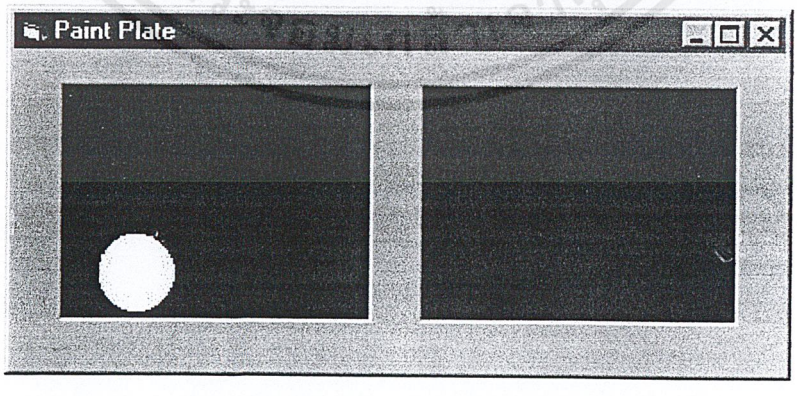
การระบายสีจานที่คิดราคาแล้วได้ผลการทดลอง ดังรูป



รูป 4.14 การระบายสีจานชุดแรก



รูป 4.15 การระบายสีจานชุดที่สอง



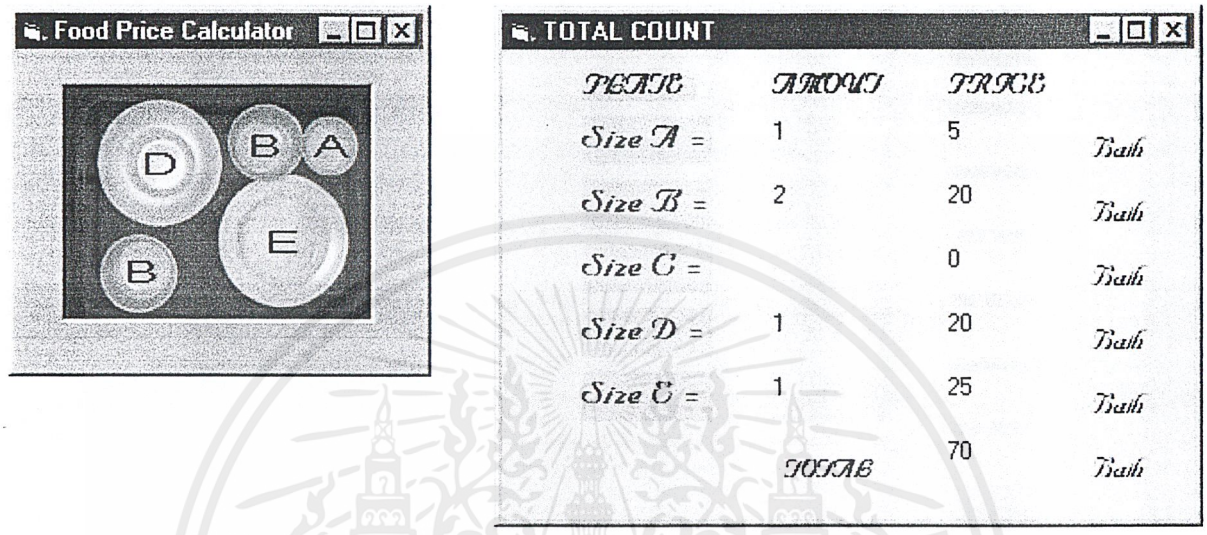
รูป 4.16 การระบายสีจานชุดที่สาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.10 การคิดราคาจากถาดกรณีต่างๆ

เพื่อทดสอบความถูกต้อง จึงทำการทดสอบกับการวางจานในลักษณะต่างๆ ดังนี้

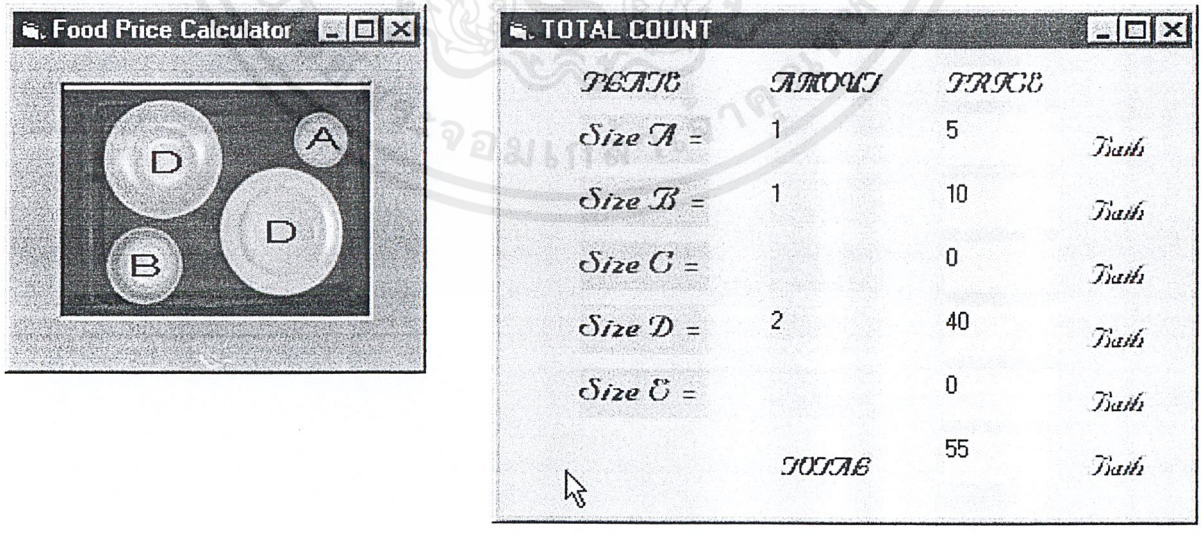
การทดสอบที่ 1



รูป 4.17 การทดสอบในการคิดราคาจริงครั้งที่ 1

การทดสอบครั้งที่ 1 ผลการคิดราคาถูกต้อง

การทดสอบที่ 2

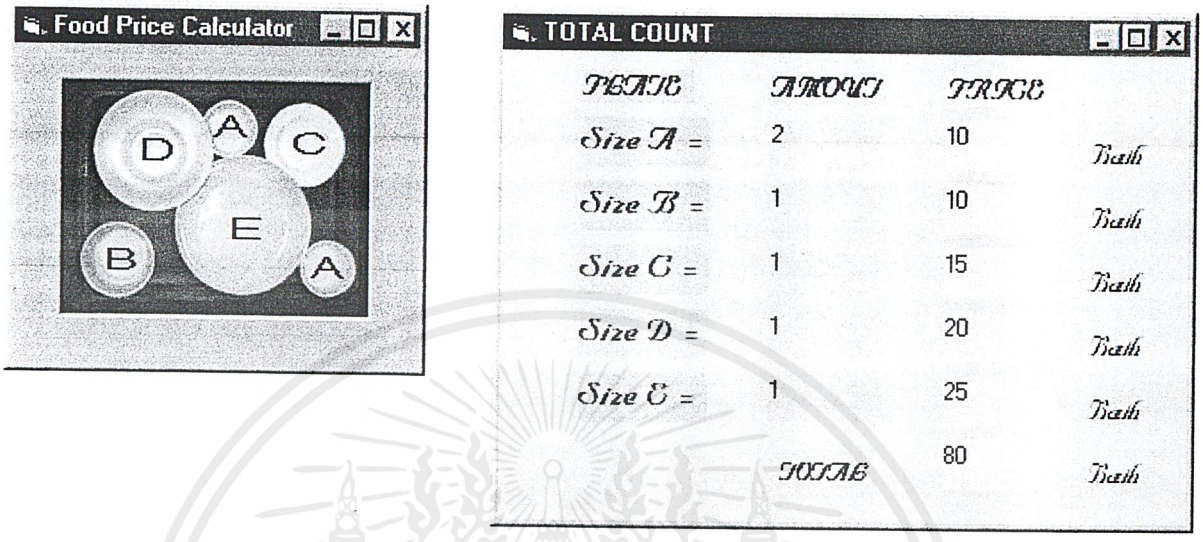


รูป 4.18 การทดสอบในการคิดราคาจริงครั้งที่ 2

การทดสอบครั้งที่ 2 ผลการคิดราคาถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

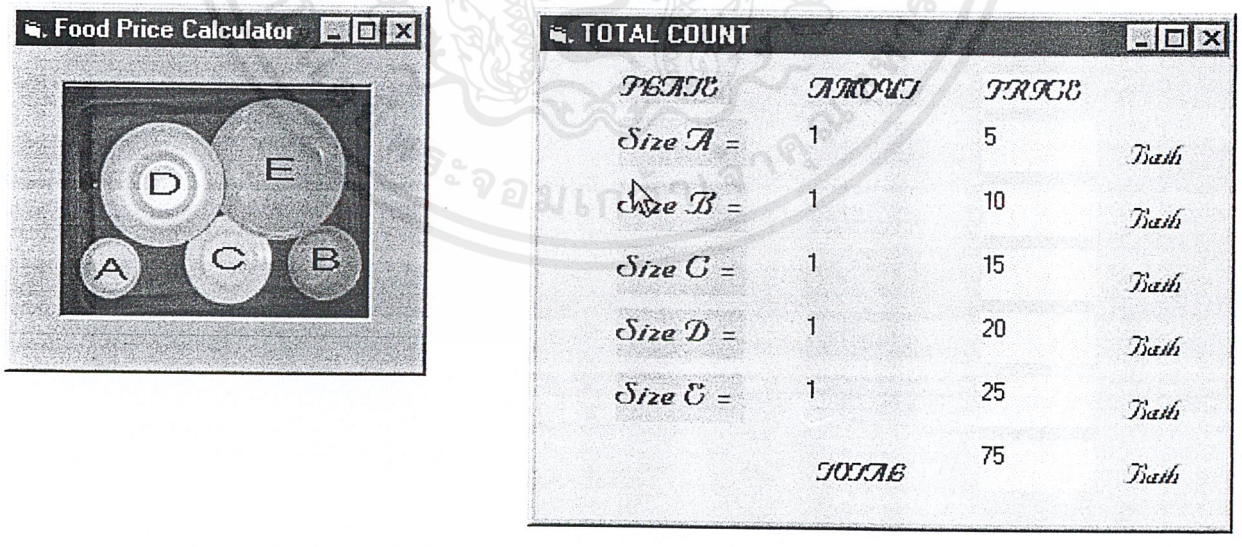
การทดสอบที่ 3



รูป 4.19 การทดสอบในการคิดราคาจริงครั้งที่ 3

การทดสอบครั้งที่ 3 ผลการคิดราคาถูกต้อง

การทดสอบที่ 4

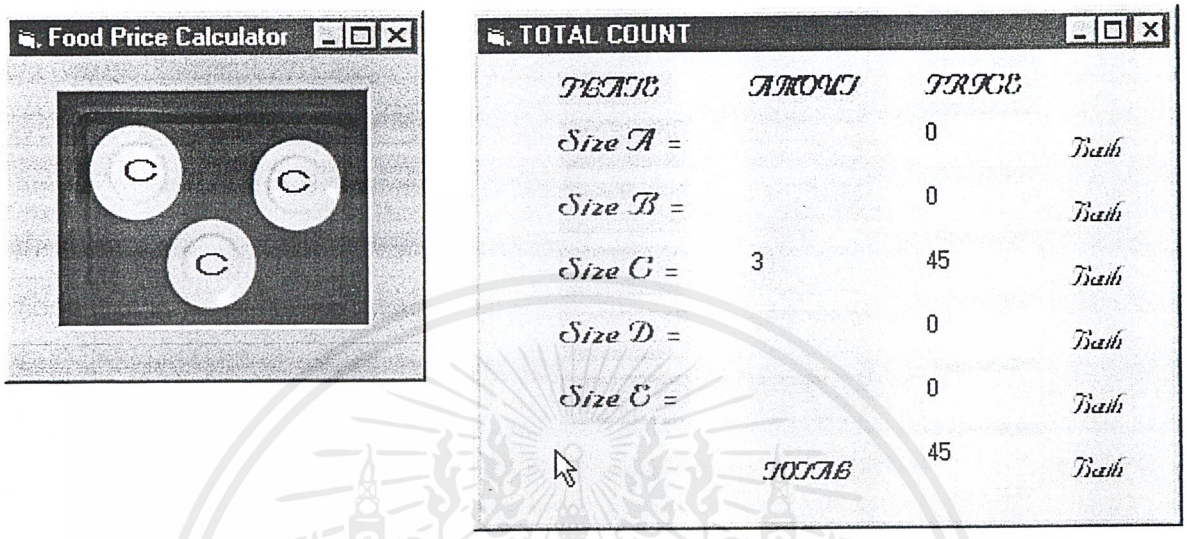


รูป 4.20 การทดสอบในการคิดราคาจริงครั้งที่ 4

การทดสอบครั้งที่ 4 ผลการคิดราคาถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบที่ 5



รูป 4.21 การทดสอบในการคิดราคาจริงครั้งที่ 5

การทดสอบครั้งที่ 5 ผลการคิดราคาถูกต้อง

4.11 ผลการทดลองการเก็บฐานข้อมูลของถาดที่คิดราคาแล้ว

จากการทดลองเก็บข้อมูลของถาดที่คิดราคาย้อนหลัง จากข้อ 4.9 ได้ผลดังรูป

Date	Time	Plate A	Plate B	Plate C	Plate D	Plate E	Price
15/3/45	5:34 PM	1	1	0	2	0	55
15/3/45	5:45 PM	1	2	0	1	0	60
16/3/45	6:30 PM	2	1	1	1	1	80
16/3/45	6:45 PM	1	1	1	1	1	75
16/3/45	6:50 PM	0	0	3	0	0	45
	Total	5	2	5	5	2	315

รูป 4.22 ตารางแสดงฐานข้อมูลราคาอาหารที่คิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่5
บทวิจารณ์และสรุปผลการทดลอง

ในการรับภาพเข้ามาในโปรแกรมจะใช้ component ซึ่งเป็น ActiveX control ที่ชื่อ Ezvidcap.ocx ได้ความละเอียดของภาพ 120 x 160 pixel ซึ่งความละเอียดขณะนี้ เพียงพอแล้วกับการประมวลผล เพราะเป็นการหารัศมีของจานเท่านั้นจึ่งไม่ต้องการความละเอียดมาก

จากการทดลองในการแปลงภาพเป็นไบนารี ค่า เทรชโฮลด์ จะใช้ค่าจากการเลือกจากความเหมาะสมเพราะวัตถุ คือ จานกับพื้นหลัง ซึ่งมีความแตกต่างของสีอย่างชัดเจนและต้องเป็นค่าที่สามารถกำจัดผลของแสงสะท้อนให้หมดไปได้ ซึ่งจะเลือกค่าเทรชโฮลด์ที่มีโทนสีหนักไปทางสีดำมากกว่า จึงทำให้ลดภาระในส่วนการกำจัดสัญญาณรบกวนลงไปได้มาก จนบางครั้งไม่มีการเปลี่ยนแปลงหลังจากทำการกำจัดสัญญาณรบกวน

ในส่วนของการหาขอบภาพ และการกำจัดสัญญาณรบกวนสามารถทำได้อย่างดีดังแสดงในผลการทดลอง

ในส่วนของการหาจุดหักของขอบจานอาจมีปัญหาบ้างในกรณีที่วงกลมนั้นมีรัศมีความโค้งมาก เพราะเส้นคอร์ดอาจสั้นเกินไปจนทำให้มองเส้นรอบวงส่วนนั้นเหมือนเส้นตรง ซึ่งจากการทดลองวงกลมที่มีปัญหาจะเป็นวงที่มีขนาดใหญ่มากเกินกว่าขนาดจานใหญ่ที่สุดที่ใช้ในการทดลอง จึงไม่มีปัญหาส่วนนี้

การหารัศมีของจานในกรณีจานใบเดียว จากการทดลองเมื่อเลื่อนจานไปในส่วนต่างๆ ของถาด ค่ารัศมีที่ได้ก็ยังมีค่าเท่าเดิม จึงสรุปได้ว่าการหารัศมีด้วยวิธีนี้ใช้ได้

การหารัศมีของจานในกรณีที่ขอบจานซ้อนกันการหาจุด 3 จุด จะต้องพยายามให้ครอบคลุมส่วนของเส้นโค้งให้มากสุด เพราะจะให้ค่าจุดศูนย์กลางจานที่ถูกต้องที่สุด ดังนั้นหากจุด 3 จุดกิน ระยะส่วนของเส้นโค้งที่แคบมากจะข้ามการคิดจุด 3 จุดช่วงนั้นไป ซึ่งจากการทดลองผลการหาจุดศูนย์กลางจานในกรณีที่มีขอบจานซ้อนกัน ค่าจุดศูนย์กลางที่ได้อาจมีความคลาดเคลื่อนเล็กน้อยในบางครั้ง แต่ไม่เป็นปัญหาเพราะมีการป้องกันเอาไว้แล้วในส่วนของการกำหนดช่วงที่จะแยกชนิดของจาน โดยการกำหนดช่วงของการระบุชนิดจานไว้กว้างกว่า ช่วงของความผิดพลาดที่อาจเกิดขึ้น มากๆ

ดังนั้นจากการทดลองทั้งหมด สรุปได้ว่าเครื่องคิดราคาอาหารนี้สามารถคิดราคาอาหารได้อย่างถูกต้องแม่นยำ ถ้าใช้งานตามเงื่อนไขที่กำหนดไว้ คือ อาหารไม่พ้นขอบจาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญหาและอุปสรรค

ในโครงงานใช้ภาพที่มีความละเอียด 120 x 160 เพราะเป็น ActiveX รูปที่ถูกกำหนดมาแล้วไม่สามารถเปลี่ยนได้ ดังนั้นหากสามารถทำให้รูปที่รับเข้ามามีความละเอียดมากกว่านี้ จะสามารถคำนวณหาจุดศูนย์กลาง และ ค่ารัศมีได้ถูกต้องแม่นยำขึ้น

ในการแก้ปัญหาแสงสะท้อนแก้ปัญหาโดยการใช้ฉากและถาดเป็นสีดำด้านซึ่งหายากในท้องตลาด จึงต้องพ้นสีฉากและถาดเอง ดังนั้นอาจเป็นปัญหาในการหาถาดและฉากในการนำไปใช้งานจริง

ในส่วนของการหาจุดศูนย์กลางของจานได้ข้ามกรณีที่จุด 3 จุดที่จะนำมาหาจุดศูนย์กลางกินรัศมีความโค้งที่แคบดังนั้น หากต้องการใช้จานที่มีขนาดเล็กมากๆ จะต้องคิดหาวิธีในการหาจุดศูนย์กลางใหม่

แนวทางการพัฒนาในอนาคต

โครงงานชิ้นนี้จุดประสงค์คือ การคิดราคาอาหารให้ได้ทุกรูปแบบซึ่ง และสามารถแยกแยะราคาได้ หลายช่วงราคา ดังนั้นแนวทางในการพัฒนา ควรพัฒนาให้สามารถแยกแยะชนิดของจานจากสีได้ด้วย เหตุที่โครงการนี้ไม่สามารถ นำสีของจากมาแยกแยะชนิดได้เพราะ กล้องวีดีโอเป็นกล้องขาวดำ และหาวิธีแก้ไขจุดบอดของโปรแกรมในกรณีที่อาหารพ้นขอบจานออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้อย่างดีด้วยคำแนะนำ คำปรึกษาและแนวคิดของการทำโครงงาน รวมทั้งอุปกรณ์กล้องวีดีโอ จากอาจารย์สุมิตร พนาอุดมทรัพย์ ซึ่งเป็นอาจารย์ที่ปรึกษาในการทำปริญญานิพนธ์ ผู้จัดทำรู้สึกซาบซึ้งในความอนุเคราะห์จากอาจารย์และขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณอาจารย์ทุกท่านในภาควิชาวิศวกรรมระบบควบคุมที่เห็นความสำคัญและประโยชน์ของงานชิ้นนี้ จึงได้มีการอนุมัติให้เป็นหัวข้อปริญญานิพนธ์ขึ้น

ขอขอบพระคุณบิดาและมารดาของพวกข้าพเจ้าที่ให้ความช่วยเหลือและเป็นกำลังใจในการเรียนและการจัดทำปริญญานิพนธ์ฉบับนี้ให้ประสบความสำเร็จได้ตลอดรอดฝั่ง

ขอขอบคุณเพื่อน ๆ ภาควิชาระบบควบคุมและภาควิชาคอมพิวเตอร์ที่ให้คำแนะนำและข้อมูลต่าง ๆ อันเป็นประโยชน์ทำให้งานชิ้นนี้สำเร็จราบรื่นไปได้ด้วยดี

ขอขอบคุณรุ่นพี่ที่ช่วยตอบคำถาม ข้อสงสัยและให้คำแนะนำที่สามารถนำมาประยุกต์ใช้ในการทำงานครั้งนี้

คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้ ผู้จัดทำขอมอบแด่ผู้มีพระคุณทุกท่าน

นางสาวภัทรพรรณ จันศรีวิริยะ
นายวชิร ปราโมทย์พันธุ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. กิตติ ไพฑูรย์วัฒนกิจ สาธิต อินทจักร์ และ ไพลิน บุญเดช , "นิวรอนเน็ทเวอร์คกับการหาขอบภาพ " ประชุมใหญ่วิชาการทางวิศวกรรมประจำปี 2537 วิศวกรรมสถานแห่งประเทศไทย ฯ , หน้า EE 75-90.2537
2. กฤษณะ ชินสาร , "ระบบตรวจข้อสอบปรนัยอัตโนมัติ" ,หน้า 8 – 18
3. สิทธิชัย บุษหมั่น , "การสร้างภาพไบนารีของภาพเอกสารโดยการจำแนกส่วนของข้อความและการถ่ายภาพ " , หน้า 19-20 , 27-33
4. กิตติ ภักดีวัฒนะกุล และ จำลอง ครูอุตสาหะ , "Visual Basic 6 ฉบับ โปรแกรมเมอร์ "
5. ธาริน สิทธิธรรมชารี , " Visual Basic 6.0 "
6. P.Ioannis ., Introductory Computer Vision and Image Processing .MCGraw.Hill Inc.1992

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim E(1 To 1500) As String ' edge
Dim A(1 To 200, 1 To 200), b As Long 'save picture
Dim P(1 To 200), Center(1 To 15, 1 To 5) As String ' change point
Dim Max, Tran, Tran1, CaseSelect, Casecheck, Radius(1 To 15), StoreTotal, Total, Size,
Size1, Size2, Size3, Size4, Size5, Size6, Size7, Size8 As Integer

Private Declare Function GetDesktopWindow Lib "user32" () As Long
Private Declare Function GetDC Lib "user32" (ByVal hwnd As Long) As Long
Private Declare Function BitBlt Lib "gdi32" _
                        (ByVal hDestDC As Long, _
                         ByVal x As Long, _
                         ByVal y As Long, _
                         ByVal nWidth As Long, _
                         ByVal nHeight As Long, _
                         ByVal hSrcDC As Long, _
                         ByVal xSrc As Long, _
                         ByVal ySrc As Long, _
                         ByVal dwRop As Long) As Long


'Store to Binary
Private Sub Command1_Click()
  Dim i, j As Integer
 Const PIXEL = 3
    ' ezVidCap1.CapSingleFrame
     Picture2.ScaleMode = PIXEL
     Form1.ScaleMode = PIXEL
     hDestDC& = Picture2.hDC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    x& = 0: y& = 0
    nWidth& = Picture2.ScaleWidth
    nHeight& = Picture2.ScaleHeight

    hSrcDC& = Form1.hDC
    xSrc& = 10: ySrc& = 10

    dwRop& = &HCC0020
    Suc& = BitBlt(hDestDC&, x&, y&, nWidth&, nHeight&, hSrcDC&, _
    xSrc&, ySrc&, dwRop&)
' Call Clearsize
For i = 0 To 159
   For j = 0 To 119
      b = Picture2.Point(i, j)
      If b >= 8388608 Then
         b = 0
      Else
         b = 1
      End If

      A(i + 1, j + 1) = b
   Next j
Next i

Call PaintTray
Call Kfill
Call Kfill
   For i = 1 To 160
      A(i, 1) = 1
      A(i, 120) = 1
   Next i
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  For i = 1 To 120
     A(1, i) = 1
     A(160, i) = 1
  Next i
 For i = 1 To 160
  For j = 1 To 120
     If A(i, j) = 1 Then
        b = 0
     Else
        b = 256
     End If
     Picture2.PSet (i - 1, j - 1), RGB(b, b, b)
   Next j
Next i

Call Calculate
Call keep
Label14 = CStr(Total)
End Sub
Sub Clearsize()
If Check1.Value = 0 Then
Size1 = 0
Size2 = 0
Size3 = 0
Size4 = 0
Size5 = 0
Else
Label11.Caption = 0

End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'Paint Tray
Private Sub PaintTray()
Dim x, y As Integer
   For y = 1 To 120
      For x = 1 To 160
         If A(x, y) = 0 Then
         A(x, y) = 1
         Else

         Exit For
         End If

      Next x
   Next y
   For y = 120 To 1 Step -1
      For x = 160 To 1 Step -1
         If A(x, y) = 0 Then
         A(x, y) = 1
         'Picture2.PSet (x - 1, y - 1), RGB(0, 0, 0)

         Else

         Exit For
         End If

      Next x
   Next y
End Sub

'KFILL ALGORITHM
Sub Kfill()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim x, y, i, j, C, r, n, w As Integer
y = 1  'y = y axis
For j = 1 To 118 Step 1
   x = 1  'x = x axis
   y = y + 1
   For i = 1 To 158 Step 1
      x = x + 1
      If A(x, y) = 0 Then
      C = 0
      n = 0
      r = 0
      If A(x - 1, y - 1) = 1 Then
         n = n + 1
         r = r + 1
      End If
      If (A(x - 1, y - 1) = 1 And A(x, y - 1) = 0) Or (A(x - 1, y - 1) = 0 And A(x, y - 1) = 1)
Then
         C = C + 1
      End If

      If A(x, y - 1) = 1 Then
         n = n + 1
      End If
      If (A(x, y - 1) = 1 And A(x + 1, y - 1) = 0) Or (A(x, y - 1) = 0 And A(x + 1, y - 1) =
1) Then
         C = C + 1
      End If


      If A(x + 1, y - 1) = 1 Then
         n = n + 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            r = r + 1
        End If
        If (A(x + 1, y - 1) = 1 And A(x + 1, y) = 0) Or (A(x + 1, y - 1) = 0 And A(x + 1, y) =
1) Then
            C = C + 1
        End If


        If A(x + 1, y) = 1 Then
            n = n + 1
        End If
        If (A(x + 1, y) = 1 And A(x + 1, y + 1) = 0) Or (A(x + 1, y) = 0 And A(x + 1, y + 1)
= 1) Then
            C = C + 1
        End If


        If A(x + 1, y + 1) = 1 Then
            n = n + 1
        r = r + 1
        End If
        If (A(x + 1, y + 1) = 1 And A(x, y + 1) = 0) Or (A(x + 1, y + 1) = 0 And A(x, y + 1)
= 1) Then
            C = C + 1
        End If


        If A(x, y + 1) = 1 Then
            n = n + 1
        End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If (A(x, y + 1) = 1 And A(x - 1, y + 1) = 0) Or (A(x, y + 1) = 0 And A(x - 1, y + 1) =
1) Then
    C = C + 1
End If

If A(x - 1, y + 1) = 1 Then
    n = n + 1
    r = r + 1
End If
If (A(x - 1, y + 1) = 1 And A(x - 1, y) = 0) Or (A(x - 1, y + 1) = 0 And A(x - 1, y) =
1) Then
    C = C + 1
End If

If A(x - 1, y) = 1 Then
    n = n + 1
End If
If (A(x - 1, y) = 1 And A(x - 1, y - 1) = 0) Or (A(x - 1, y) = 0 And A(x - 1, y - 1) = 1)
Then
    C = C + 1
End If

If C = 0 Then  'In case have no white block
    C = 1
Else           'Have white block
    C = C / 2
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            If (C = 1) And ((5 < n) Or (5 = n)) And (r > 1) Then
               A(x, y) = 1
               'Picture2.PSet (x - 1, y - 1), RGB(0, 0, 0)
            End If
            End If
        Next i
    Next j

End Sub

Private Sub Command3_Click()
    Picture2.Cls
End Sub
Sub SetInitial()
Dim i, j As Integer
For i = 1 To Max
    E(i) = ""
Next i
For i = 1 To 200
    P(i) = ""
Next i
For i = 1 To 15
    For j = 1 To 5
        Center(i, j) = ""
    Next j
Next i
For i = 1 To 15
    Radius(i) = 0
Next i
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'Saperate Plate
Sub Calculate()
Dim x, y As Integer
finish:
Call SetInitial
  For y = 1 To 120
    For x = 1 To 160
      If A(x, y) = 0 Then
        Call FindTheEdge(x, y)


        GoTo finish


      End If
    Next x
  Next y
End Sub


Sub FindTheEdge(x, y As Integer)
Dim C, D, n, P As Integer


    E(1) = CStr(x) + "  " + CStr(y)
    n = 1
    P = 4  ' direction into next block : right = 1 down = 2 left = 3 up = 4
    C = x
    D = y - 1
    Do Until C = x And D = y
    If A(C, D) = 0 Then
      n = n + 1
      E(n) = CStr(C) + "  " + CStr(D)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Select Case P ' to move 1 step
    Case 1
      P = 4
      C = C
      D = D - 1
    Case 2
      P = 1
      C = C + 1
      D = D
    Case 3
      P = 2
      C = C
      D = D + 1
    Case 4
      P = 3
      C = C - 1
      D = D
    End Select
  Else
    Select Case P ' to move 1 step
    Case 1
      P = 2
      C = C
      D = D + 1
    Case 2
      P = 3
      C = C - 1
      D = D
    Case 3
      P = 4
      C = C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      D = D - 1
     Case 4
      P = 1
      C = C + 1
      D = D
     End Select
    End If
    Loop
    'Picture2.Cls
    Max = n
    For P = 1 To n
     C = CInt(Left$(E(P), 4))
     D = CInt(Right$(E(P), 4))
    Next P
    Call FindChangePoint
End Sub



Private Sub FindChangePoint()
Dim i, j, k, w, x, y, x1, y1, X2, Y2, X3, Y3, m, n, s As Integer
Dim direction, slope, slope1, slope2, slope3, slope4, slope5 As Integer
j = 0
k = 0
slope1 = 0
slope2 = 0

For i = 1 To Max - 14 Step 1
 slope3 = slope2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
slope2 = slope1
slope1 = slope
x = CInt(Left$(E(i), 4))
y = CInt(Right$(E(i), 4))

x1 = CInt(Left$(E(i + 7), 4))   'Large value heip large plate
y1 = CInt(Right$(E(i + 7), 4))

X2 = CInt(Left$(E(i + 14), 4))
Y2 = CInt(Right$(E(i + 14), 4)) 'Large value heip large plate but error small plate

'X3 = CInt(Left$(E(i + 9), 4))
'Y3 = CInt(Right$(E(i + 9), 4))

If x1 > x And y1 > y Then
  direction = 1
ElseIf x1 = x And y1 > y Then
  direction = 2
ElseIf x1 < x And y1 > y Then
  direction = 3
ElseIf x1 < x And y1 = y Then
  direction = 4
ElseIf x1 < x And y1 < y Then
  direction = 5
ElseIf x1 = x And y1 < y Then
  direction = 6
ElseIf x1 > x And y1 < y Then
  direction = 7
ElseIf x1 > x And y1 = y Then
  direction = 8
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Select Case direction
  Case 1
    If (X2 = x1 And Y2 < y1) Or (X2 > x1 And Y2 < y1) Or (X2 > x1 And Y2 = y1) Then
'Or (X3 = X1 And Y3 < Y1) Or (X3 > X1 And Y3 < Y1) Or (X3 > X1 And Y3 = Y1)
      slope = 1
    Else
      slope = 0
    End If

  Case 2
    If (X2 > x1 And Y2 < y1) Or (X2 > x1 And Y2 = y1) Or (X2 > x1 And Y2 > y1) Then
'Or (X3 > X1 And Y3 < Y1) Or (X3 > X1 And Y3 = Y1) Or (X3 > X1 And Y3 > Y1)
      slope = 1
    Else
      slope = 0
    End If
  Case 3
    If (X2 > x1 And Y2 = y1) Or (X2 > x1 And Y2 > y1) Or (X2 = x1 And Y2 > y1) Then
'Or (X3 > X1 And Y3 = Y1) Or (X3 > X1 And Y3 > Y1) Or (X3 = X1 And Y3 > Y1)
      slope = 1
    Else
      slope = 0
    End If
  Case 4
    If (X2 > x1 And Y2 > y1) Or (X2 = x1 And Y2 > y1) Or (X2 < x1 And Y2 > y1) Then
'Or (X3 > X1 And Y3 > Y1) Or (X3 = X1 And Y3 > Y1) Or (X3 < X1 And Y3 > Y1)
      slope = 1
    Else
      slope = 0
    End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Case 5
        If (X2 = x1 And Y2 > y1) Or (X2 < x1 And Y2 > y1) Or (X2 < x1 And Y2 = y1) Then
'Or (X3 = X1 And Y3 > Y1) Or (X3 < X1 And Y3 > Y1) Or (X3 < X1 And Y3 = Y1)
            slope = 1
        Else
            slope = 0
        End If

    Case 6
        If (X2 < x1 And Y2 > y1) Or (X2 < x1 And Y2 = y1) Or (X2 < x1 And Y2 < y1) Then
'Or (X3 < X1 And Y3 > Y1) Or (X3 < X1 And Y3 = Y1) Or (X3 < X1 And Y3 < Y1)
            slope = 1
        Else
            slope = 0
        End If
    Case 7
        If (X2 < x1 And Y2 = y1) Or (X2 < x1 And Y2 < y1) Or (X2 = x1 And Y2 < y1) Then
'Or (X3 < X1 And Y3 = Y1) Or (X3 < X1 And Y3 < Y1) Or (X3 = X1 And Y3 < Y1)
            slope = 1
        Else
            slope = 0
        End If
    Case 8
        If (X2 < x1 And Y2 < y1) Or (X2 = x1 And Y2 < y1) Or (X2 > x1 And Y2 < y1) Then
'Or (X3 < X1 And Y3 < Y1) Or (X3 = X1 And Y3 < Y1) Or (X3 > X1 And Y3 < Y1)
            slope = 1
        Else
            slope = 0
        End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  End Select

  If slope + slope1 >= 2 And i - k > 10 Then  '+ slope2 + slope3
    j = j + 1
    x = CInt(Left$(E(i + 11), 4))
    y = CInt(Right$(E(i + 11), 4))
    P(j) = CStr(x) + "   " + CStr(i) + "   " + CStr(y)
    'Picture2.PSet (X, Y), RGB(0, 0, 0)
    'Text1.Text = " p" & j & " = " & p(j)
    k = i
  End If
Next i

If P(1) = "" Then
    Call Case1
Else
    'For Show Change point
    For i = 1 To j
       X3 = CInt(Left$(P(i), 4))
       Y3 = CInt(Right$(P(i), 4))
    Next i
    Tran1 = j
    Call Case2
End If

End Sub
'Find the center of the circle
Sub Case1()
    Dim s, m As Integer
    s = CInt(Right$(E(1), 4))
    m = CInt(Right$(E(Max / 2), 4))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Radius(1) = CInt((m - s) / 2)
    CaseSelect = 1
    Call CountPlate
End Sub
Sub Case2()
Dim i, j, k, w, x, y, x1, m, n, s, t As Integer
Dim Linv(1 To 3, 1 To 3), D, C, r(1 To 3) As Long
Dim Det, DetPos, DetNeg, l(1 To 3, 1 To 3) As Long
    t = 0
    s = CInt(Mid$(P(1), 6, 5))
    For i = 1 To Tran1
      If i = Tran1 Then
        n = (CInt(Mid$(P(1), 6, 5))) + (CInt(Mid$(P(1), 6, 5)))

      Else
        k = CInt(Mid$(P(i + 1), 6, 5))
        n = (k) - (s)
      End If
      ' For reduce error but if have n<25
      If n < 26 Then
        GoTo Tooshort
      End If
      n = Int(n / 2)
      'find 3 points

      For m = 1 To 3
        If i = Tran1 Then
        Select Case m
        Case 3
          k = CInt(Mid$(P(1), 6, 5))
          x = CInt(Left$(E(k + 3), 4))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        y = CInt(Right$(E(k + 3), 4))
    Case 2

        x = CInt(Left$(E(Max), 4))
        y = CInt(Right$(E(Max), 4))
    Case 1
        x = CInt(Left$(E(s + 9), 4))
        y = CInt(Right$(E(s + 9), 4))
    End Select
    Else
    Select Case m

    Case 3
        x = CInt(Left$(E(k + 3), 4))
        y = CInt(Right$(E(k + 3), 4))
    Case 2
        x = CInt(Left$(E(s + n + 6), 4))
        y = CInt(Right$(E(s + n + 6), 4))
    Case 1
        x = CInt(Left$(E(s + 9), 4))
        y = CInt(Right$(E(s + 9), 4))
    End Select
    End If

    Center(i, m) = CStr(x) + "   " + CStr(y)
    (m, 1) = x  'L(R,C)
    I(m, 2) = y
    I(m, 3) = 1
    r(m) = -((x * x) + (y * y))
Next m
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    'DET(L)
    GoTo Continue
Chkdet:
    I(1, 1) = I(1, 1) + 1
    ChkdskValue = 1
Continue:
DetPos = ((I(1, 1) * I(2, 2) * I(3, 3)) + (I(1, 2) * I(2, 3) * I(3, 1)) + (I(1, 3) * I(2, 1) * I(3, 2)))
DetNeg = ((I(3, 1) * I(2, 2) * I(1, 3)) + (I(3, 2) * I(2, 3) * I(1, 1)) + (I(3, 3) * I(2, 1) * I(1, 2)))
Det = DetPos - DetNeg
    If Det = 0 Then
        If ChkdskValue = 1 Then
            I(1, 2) = I(1, 2) + 1
            ChkdskValue = 0
        Else
            GoTo Chkdet
        End If
    End If
    'Minor(Lij)*Cofactor
    Linv(1, 1) = (I(2, 2) * I(3, 3)) - (I(3, 2) * I(2, 3))
    Linv(1, 2) = -((I(2, 1) * I(3, 3)) - (I(3, 1) * I(2, 3)))
    Linv(1, 3) = (I(2, 1) * I(3, 2)) - (I(3, 1) * I(2, 2))
    Linv(2, 1) = -((I(1, 2) * I(3, 3)) - (I(3, 2) * I(1, 3)))
    Linv(2, 2) = (I(1, 1) * I(3, 3)) - (I(3, 1) * I(1, 3))
    Linv(2, 3) = -((I(1, 1) * I(3, 2)) - (I(3, 1) * I(1, 2)))
    Linv(3, 1) = (I(1, 2) * I(2, 3)) - (I(2, 2) * I(1, 3))
    Linv(3, 2) = -((I(1, 1) * I(2, 3)) - (I(2, 1) * I(1, 3)))
    Linv(3, 3) = (I(1, 1) * I(2, 2)) - (I(2, 1) * I(1, 2))
    'Transpote Linv
    For m = 1 To 3
        For n = 1 To 3
        I(m, n) = Linv(m, n)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Next n
  Next m
  For m = 1 To 3
    For n = 1 To 3
    Linv(m, n) = I(n, m) / Det ' Divide by det
    Next n
  Next m

  C = (Linv(1, 1) * r(1)) + (Linv(1, 2) * r(2)) + (Linv(1, 3) * r(3))
  D = (Linv(2, 1) * r(1)) + (Linv(2, 2) * r(2)) + (Linv(2, 3) * r(3))

  C = Int(-C / 2)  'Center X
  D = Int(-D / 2)  'Center Y

   t = t + 1

   Center(i, 4) = CStr(C) + "   " + CStr(D)
   'Text1.Text = " center" & i & " = " & center(i)

Tooshort:
   s = k
   Next i
   Tran = t

  k = 0
nonzero1:
For i = 1 To Tran1
      If Center(i, 1) = "" Then
         k = k + 1
         For w = i To Tran1
            For n = 1 To 5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Center(w, n) = Center(w + 1, n)
          Next n
        Next w
      End If
    Next i
    For i = 1 To Tran1 - k
      If Center(i, 1) = "" Then
      GoTo nonzero1
    End If
    Next i
    Call FindRadius
  End Sub


Private Sub FindRadius()
Dim i, j, k, w, x, y, x1, y1, m, n, s, t As Integer
Dim center1, center2 As String
Dim Sum, Sum1 As Long

' Sum many center by average
   w = 0

    For i = 1 To Tran
      x = CInt(Left$(Center(i, 4), 4))
      y = CInt(Right$(Center(i, 4), 4))
      For n = y - 10 To y + 10 'y
        For m = x - 10 To x + 10 'x

          center2 = CStr(m) + "   " + CStr(n)
          For s = 1 To Tran
            j = CInt(Left$(Center(s, 4), 4))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                k = CInt(Right$(Center(s, 4), 4))
                center1 = CStr(j) + "   " + CStr(k)
                If center2 = center1 Then



                   x1 = x1 + j
                   y1 = y1 + k
                   w = w + 1


                End If
             Next s
          Next m
       Next n
       x1 = Int(x1 / w)
       y1 = Int(y1 / w)
       Center(i, 5) = CStr(x1) + "   " + CStr(y1)
       x1 = 0
       y1 = 0
       w = 0
    Next i

    'rearrange center
    t = 0
    'The same center give another as ""
    For i = 1 To Tran
       For j = i + 1 To Tran
       If Center(i, 5) = "" Then
        GoTo pass1
       ElseIf Center(j, 5) = "" Then
        GoTo pass
       Else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        x = CInt(Left$(Center(i, 5), 4))
        y = CInt(Right$(Center(i, 5), 4))
        x1 = CInt(Left$(Center(j, 5), 4))
        y1 = CInt(Right$(Center(j, 5), 4))
       End If
       If x = x1 And y = y1 Then
         For n = 1 To 5
          Center(j, n) = ""
         Next n
         t = t + 1
       End If
pass:
      Next j
pass1:
   Next i

   'Show center
   k = 0
   For w = 1 To Tran
   If Center(w, 5) = "" Then
      GoTo finish
   End If
   k = k + 1
   x = CInt(Left$(Center(w, 5), 4))
   y = CInt(Right$(Center(w, 5), 4))
   Next w

finish:
   ' AVERAGE RADIUS
   For i = 1 To Tran
      If Center(i, 1) = "" Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            GoTo finish1
        End If

        x = CInt(Left$(Center(i, 5), 4))
        y = CInt(Right$(Center(i, 5), 4))
        For j = 1 To 3
            x1 = CInt(Left$(Center(i, j), 4))
            y1 = CInt(Right$(Center(i, j), 4))
            Sum1 = Sqr(((x - x1) * (x - x1)) + ((y - y1) * (y - y1)))
            Sum = Sum + Sum1
        Next j
        Radius(i) = Int(Sum / 3)
        Sum = 0
finish1:
    Next i
    Tran = i - 1

CaseSelect = 2
Call CountPlate
End Sub

Sub CountPlate()
Dim StrSize1, StrSize2, StrSize3, StrSize4, StrSize5 As String
Dim Price1, Price2, Price3, Price4, Price5 As Integer
If CaseSelect = 1 Then
    Tran = 1
End If

For i = 1 To Tran
   Select Case Radius(i)
    Case 10 To 15
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Size1 = Size1 + 1
    Case 16 To 24
        Size2 = Size2 + 1
    Case 25 To 31
        Size3 = Size3 + 1
    Case 32 To 38
        Size4 = Size4 + 1
    Case 39 To 45
        Size5 = Size5 + 1
    End Select

    StrSize1 = CStr(Size1)
    StrSize2 = CStr(Size2)
    StrSize3 = CStr(Size3)
    StrSize4 = CStr(Size4)
    StrSize5 = CStr(Size5)
Next i
    Label1(1) = StrSize1
    Label2(1) = StrSize2
    Label3(1) = StrSize3
    Label4(1) = StrSize4
    Label5(1) = StrSize5

Price1 = Size1 * 5
Price2 = Size2 * 10
Price3 = Size3 * 15
Price4 = Size4 * 20
Price5 = Size5 * 25
Label1(2) = CStr(Price1)
Label2(2) = CStr(Price2)
Label3(2) = CStr(Price3)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Label4(2) = CStr(Price4)
Label5(2) = CStr(Price5)

Label11 = CStr(Price1 + Price2 + Price3 + Price4 + Price5)
Call PaintPlate
End Sub
'Paint all close plate
Private Sub PaintPlate()
 Dim i, j, x, y, m, n As Integer
   For i = 1 To Max
      x = CInt(Left$(E(i), 4))
      y = CInt(Right$(E(i), 4))
      A(x, y) = 2
   Next i
   For y = 1 To 120
      For x = 1 To 160

         If A(x, y) = 2 Then
            For m = x To 159

               If A(m, y) = 2 And A(m + 1, y) = 1 Then  'And j = 1
                  A(m, y) = 0
                  x = m

                  GoTo Endline
               End If

               A(m, y) = 1
            Next m
         End If
Endline:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Next x
  Next y
  For i = 1 To Max
    x = CInt(Left$(E(i), 4))
    y = CInt(Right$(E(i), 4))
   ' Picture2.PSet (x, y), RGB(0, 0, 0)
    A(x, y) = 1
  Next i
  Call Kfill

  For y = 0 To 119
    For x = 0 To 159
      If A(x + 1, y + 1) = 1 Then
       n = 0
      ElseIf A(x + 1, y + 1) = 0 Then
       n = 255
      Else
        n = 0
      End If
     ' Picture2.PSet (x, y), RGB(n, n, n)
    Next x
  Next y
  Call Kfill
  Call Kfill
  Call Kfill

   'call KFILL and then FINISH
End Sub
Private Sub Command11_Click()
For i = 80 To 101
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'x = CInt(Left$(E(i), 4))
'y = CInt(Right$(E(i), 4))
Picture2.PSet (i, 50), RGB(0, 0, 0)

Next i
End Sub

Private Sub Command12_Click()
'x = CInt(Left$(E(88), 4))
'y = CInt(Right$(E(88), 4))
Picture2.PSet (71, 0), RGB(0, 0, 0)
'x = CInt(Left$(E(104), 4))
'y = CInt(Right$(E(104), 4))
'Picture2.PSet (x, y), RGB(0, 0, 0)
End Sub
Sub keep()
Dim F, D As Variant
F = DateValue(Date)
D = TimeValue(Time)
   Form2.Data1.Recordset.AddNew
   Form2.Text1.Text = F
   Form2.Text2.Text = D
   Form2.Text3.Text = Size1
   Form2.Text4.Text = Size2
   Form2.Text5.Text = Size3
   Form2.Text6.Text = Size4
   Form2.Text7.Text = Size5
   Form2.Text8.Text = Label11

   If Check1.Value = 0 And Casecheck = 0 Then
      Total = Total + CInt(Label11)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Casecheck = 0


        Size1 = 0
        Size2 = 0
        Size3 = 0
        Size4 = 0
        Size5 = 0
    ElseIf Check1.Value = 1 Then
        StoreTotal = CInt(Label11)
        'Total = Total + StoreTotal


        Casecheck = 1
    ElseIf Check1.Value = 0 And Casecheck = 1 Then
        Total = Total + CInt(Label11)


        Casecheck = 0
        StoreTotal = 0


        Size1 = 0
        Size2 = 0
        Size3 = 0
        Size4 = 0
        Size5 = 0
    End If


    If D = "12.00PM" Then
    Label14.Caption = 0
    End If
    Form2.Data1.Recordset.Update
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub mnu1_Click()
Form1.Show
End Sub
Private Sub mnu2_Click()
Form2.Show
Call keep
End Sub
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้