การรู้จำอักขระชนิดสีด้วยแสง

Color Optical Character Recognition

นางสาวสุรยา	ชินะกาญจน์	รหัส 46362158
นายเขมภพ	บุญพลอย	รหัส 46362224
นายเอกวิทย์	กลิ่นลั่นทม	รหัส 46362257

i5080496 e.2

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 15 ต.ค. 2550
เลขทะเบียน 5000061
เลขเรียกหนังสือ ปร
มหาวิทยาลัยนเรศวร ศ ๒๕๔๙

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
ปีการศึกษา 2549

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน	การรู้จำอักขระชนิดสีด้วยแสง		
ผู้ดำเนินโครงงาน	นางสาวสุรยา	ชินะกาญจน์	รหัส 46362158
	นายเขมภพ	บุญพลอย	รหัส 46362224
	นายเอกวิทย์	กลิ่นลั่นทม	รหัส 46362257
อาจารย์ที่ปรึกษา	ดร.สุรเดช	จิตประไพกุลศาล	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2549		

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงงานวิศวกรรม

.....ประธานกรรมการ
(ดร.สุรเดช จิตประไพกุลศาล)

.....กรรมการ
(ดร.พนมขวัญ ริยะมงคล)

.....กรรมการ
(ดร.อัครพันธ์ วงศ์กังแห)

หัวข้อโครงงาน	การรู้จำอักขระชนิดสีด้วยแสง		
ผู้ดำเนินโครงงาน	นางสาวสุรยา	ชินะกาญจน์	รหัส 46362158
	นายเขมภพ	บุญพลอย	รหัส 46362224
	นายเอกวิทย์	กลิ่นลั่นทม	รหัส 46362257
อาจารย์ที่ปรึกษา	ดร.สุรเดช	จิตประไพกุลศาล	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2549		

.....

บทคัดย่อ

ในโครงงานนี้ศึกษาและพัฒนาโปรแกรมการรู้จำอักขระชนิดสีด้วยแสง ซึ่งจะใช้หลักการการแยกสีพื้นฐานทางคอมพิวเตอร์ เพื่อหาความแตกต่างระหว่างตัวอักษรกับพื้นหลัง โดยใช้การตีกรอบและแบ่งช่วงหาจุดตัดในการเก็บข้อมูลของตัวอักษร ซึ่งโปรแกรมที่พัฒนาขึ้นนี้จะเขียนด้วยภาษา C# โดยใช้ Microsoft Visual Studio 2005

จากผลการทดลองนี้แสดงให้เห็นว่าวิธีการที่ใช้ในโปรแกรม Color OCR นั้นสามารถรู้จำตัวอักษรที่เป็น สี, เฉดสี และตัวอักษรที่มีรูปแบบต่างกันได้จริง มีประสิทธิภาพเพียงพอที่จะนำไปประยุกต์ใช้ให้เกิดประโยชน์

Project title	Color Optical Character Recognition		
Name	Miss Suraya	Chinakarn	ID. 46362158
	Mr.Khemmapop	Boonploy	ID. 46362224
	Mr.Aekawit	Klinluntom	ID. 46362257
Project advisor	Dr. Suradet	Jitprapaikulsan	
Major	Computer Engineering		
Department	Electrical and Computer Engineering		
Academic year	2006		

.....

Abstract

This project studies and develops a program for color optical character recognition. Our program separates basic color from each other to find the discrepancy between characters and background by framing and dividing data sets into interval to find a cutting point for storing characters into the database. The program was written using Visual C# version 2005.

The results of experiments have showed that our color OCR method can recognize color characters, characters with shading color and characters in different fonts.

กิตติกรรมประกาศ

ขอขอบพระคุณ ดร. สุรเดช จิตประไพกุลศาล อาจารย์ที่ปรึกษา และดร.พนมขวัญ ริยะมงคล และ ดร.อัครพันธ์ วงศ์กังแห อาจารย์ที่ปรึกษาร่วม ที่คอยให้คำปรึกษา ความช่วยเหลือตลอดจนคำแนะนำต่าง ๆ ในการทำโครงงานชิ้นนี้ สุดท้ายต้องขอขอบพระคุณอาจารย์ทุกท่านและเพื่อน ๆ ทุกคนที่ยังไม่ได้เอ่ยนามที่ให้การสนับสนุนผู้จัดทำโครงงานให้สามารถทำโครงงานชิ้นนี้จนสำเร็จลุล่วงไปได้ด้วยดี

สารบัญ

หน้า

บทคัดย่อภาษาไทย .....ก

บทคัดย่อภาษาอังกฤษ .....ข

กิตติกรรมประกาศ .....ค

สารบัญ .....ง

สารบัญตาราง .....ฉ

สารบัญรูป .....ช

บทที่ 1 1

1.1 ที่มาและความสำคัญ 1

1.2 วัตถุประสงค์ 1

1.3 ขอบข่ายของโครงงาน 1

1.4 ขั้นตอนการดำเนินงาน 2

1.5 แผนการดำเนินงาน 2

1.6 ผลที่คาดว่าจะได้รับ 2

1.7 งบประมาณ 3

บทที่ 2 4

2.1 ทฤษฎีพื้นฐานของ OCR 4

2.2 โหมดสี CMYK 4

2.3 โหมดสีแบบ RGB 5

บทที่ 3 7

3.1 การศึกษาข้อมูล 7

3.1.1 การผสมสีของคอมพิวเตอร์ 7

3.1.2 เทคนิคการปรับแต่งภาพในคอมพิวเตอร์ 7

3.1.3 เทคนิคการเก็บข้อมูลที่ได้จากรูปภาพ 8

3.1.3 เทคนิคการเก็บข้อมูลที่ได้จากรูปภาพ 8

3.1.4 เทคนิคการแม็ปตัวอักษรที่ได้จากรูปภาพกับฐานข้อมูล 8

สารบัญ

หน้า

3.2 หลักการและขั้นตอนการทำงานของโปรแกรม 9
3.2.1 ขั้นตอนการแยกสี RGB ของภาพอักษร 9
3.2.2 ขั้นตอนการแปลงภาพที่ได้จากการแยกสี RGB ไปเป็นภาพขาวดำ 10
3.2.3 ขั้นตอนการตีกรอบตัวอักษรเพื่อหาช่วงการเก็บข้อมูล และการเก็บข้อมูล 10
3.2.4 ขั้นตอนการเทียบข้อมูลทั้ง 3 ชุดข้อมูล จากขั้นตอนที่ 3 เพื่อหาข้อมูลที่ถูกต้อง....11
3.2.5 ขั้นตอนการนำชุดข้อมูลของภาพตัวอักษรเปรียบเทียบกับฐานข้อมูลของตัวอักษรที่เตรียมไว้....12
3.2.6 ขั้นตอนการบันทึกข้อมูลลงฐานข้อมูลกรณีที่ไม่มีชุดข้อมูลตัวอักษรในฐานข้อมูล 12
3.3 การออกแบบส่วนประกอบของโปรแกรม....15
3.3.1 ส่วนควบคุม 15
3.3.2 ส่วนแสดงภาพตัวอักษรต้นฉบับ 15
3.3.3 ส่วนแสดงผลลัพธ์....15
3.3.4 ส่วนของการโหลดฐานข้อมูลเก่า....16
3.4 การเขียนโปรแกรม 16
3.4.1 การแปลงภาพสีเป็นภาพขาวดำ....16
3.4.2 การเก็บข้อมูลลงในฐานข้อมูล 16
3.4.3 ปุ่มการเรียกฐานข้อมูลเก่า และสร้างฐานข้อมูลใหม่ 16
3.4.3 ปุ่ม Recognition 16

บทที่ 4....17
4.1 วิธีการใช้งานโปรแกรม 17
4.2 ผลการทดลอง....22
ตารางที่4.1 ผลการทดลองที่ 1....26

บทที่ 5....33
5.1 ผลการทดลอง....33
5.2 ปัญหาและแนวทางแก้ไข....34
5.3 สรุปผลการทดลอง 34
5.4 ข้อเสนอแนะ....34

สารบัญ

หน้า

บรรณานุกรม 32
ภาคผนวก ก 33

มหาวิทยาลัยนเรศวร

สารบัญตาราง

ตารางที่ หน้า

ตารางที่ 2.1 แสดงจำนวนสีที่เป็นไปได้ซึ่งขึ้นกับจำนวนบิต.....6

ตารางที่4.1 ผลการทดลองที่ 1.....26

ตารางที่ 4.2 ผลการทดลองที่ 2.....27

ตารางที่ 4.3 ผลการทดลองที่ 3.....28

ตารางที่ 4.4 ผลการทดลองที่ 4.....29

ตารางที่ 4.5 ผลการทดลองที่ 5.....30

สารบัญรูป

รูปที่ หน้า

รูปที่ 2.5 โหมดสี CMYK [4] 4
รูปที่ 2.6 โหมดสี RGB [4] 5
รูปที่ 3.1 ผลการแยกชั้นสี RGB 10
รูปที่ 3.2 ผลการแปลงภาพสีเป็นขาวดำ 10
รูปที่ 3.3 ตัวอย่างการเก็บข้อมูลจากรูปภาพ 11
รูปที่ 3.4 การเปรียบเทียบชุดข้อมูล RGB 12
รูปที่ 3.5 ขั้นตอนการเก็บข้อมูลจากตัวอักษร 13
รูปที่ 3.6 วิธีการเก็บข้อมูลลงฐานข้อมูล 14
รูปที่ 3.7 ขั้นตอนการเทียบข้อมูล 14
รูปที่ 3.7 หน้าตาส่วนประกอบของโปรแกรม 15
รูปที่ 4.1 การโหลดไฟล์ฐานข้อมูล 17
รูปที่ 4.2 การเลือกไฟล์ฐานข้อมูล 18
รูปที่ 4.3 การเปิดไฟล์รูปภาพที่มีตัวอักษร 18
รูปที่ 4.4 รูปที่ได้หลังจากการเปิดไฟล์รูปภาพ 19
รูปที่ 4.5 แสดงหน้าต่างเมื่อไม่มีตัวอักษรตรงกับรูปตัวอักษร 19
รูปที่ 4.6 แสดงการ Teach ให้กับฐานข้อมูล 20
รูปที่ 4.7 แสดงเมื่อทำการ Teach ให้กับฐานข้อเสร็จ 20
รูปที่ 4.8 แสดงผลการทำงานของโปรแกรม 21
รูปที่ 4.9 ตัวอักษรที่ใช้ในการทดลองที่ 1 22
รูปที่ 4.10 ตัวอักษรที่ใช้ในการทดลองที่ 2 22
รูปที่ 4.11 ตัวอักษรที่ใช้ในการทดลองที่ 3 23
รูปที่ 4.12 ตัวอักษรที่ใช้ในการทดลองที่ 4 24
รูปที่ 4.13 ตัวอักษรที่ใช้ในการทดลองที่ 5 24
รูปที่ 4.14 ตัวอักษรที่ใช้ในการทดลองที่ 6.1 25
รูปที่ 4.15 ตัวอักษรที่ใช้ในการทดลองที่ 6.2 25

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

คอมพิวเตอร์เป็นหนึ่งในเทคโนโลยีที่มีการพัฒนาอย่างต่อเนื่องและนิยมใช้กันมากที่สุด เราสามารถใช้คอมพิวเตอร์ในการพิมพ์ข้อความแทนการเขียนด้วยมือ สามารถวาดรูปภาพต่าง ๆ ได้โดยไม่ต้องใช้พู่กัน เราสามารถนำภาพและตัวอักษรจากสิ่งต่าง ๆ เข้ามาปรับปรุงแก้ไขในคอมพิวเตอร์ได้ โดยการสแกน, การถ่ายภาพจากกล้องดิจิตอล แต่เราไม่สามารถนำตัวอักษรที่อยู่ในรูปภาพต่าง ๆ มาแก้ไขและปรับปรุงได้โดยตรง ดังนั้นจึงต้องมีโปรแกรมซึ่งแปลงข้อความจากรูปภาพให้เป็นตัวอักษรในคอมพิวเตอร์

สิ่งหนึ่งที่น่าสนใจและยังมีการพัฒนาอยู่แต่ไม่มากนักก็คือการรู้จำอักษร(Character recognition)จากรูปภาพสี แต่การพัฒนายังมีปัญหาอยู่มาก เพราะหากว่ารูปภาพที่นำมาประมวลผลเป็นภาพสีที่มีความแตกต่างของสีตัวอักษรกับสีของพื้นหลังไม่มากนักจะทำให้โปรแกรมประมวลผลออกมามีข้อผิดพลาด ในโครงการนี้เราจะทดลองนำหลักการของ Image processing และทฤษฎีสี (Color theory) มาใช้ประกอบกับหลักการทาง Artificial Intelligence (AI) เพื่อพัฒนาโปรแกรมรู้จำอักขระชนิดสีด้วยแสงให้มีประสิทธิภาพมากยิ่งขึ้น

1.2 วัตถุประสงค์

เพื่อนำความรู้ทาง Image Processing มาประยุกต์ใช้

1.2.1 เพื่อนำความรู้ทาง ทฤษฎีของแสง, สี มาประยุกต์ใช้

1.2.2 เพื่อประยุกต์ใช้ความรู้ทาง Artificial Intelligence (AI) โดยเฉพาะในเรื่อง pattern recognition

1.2.3 เพื่อนำทฤษฎีของแสง, สี, Image Processing ตลอดจน AI มาผสมผสานให้เกิดเป็นวิธีการในการทำ Color OCR

1.3 ขอบข่ายของโครงงาน

1.3.1 ศึกษารูปแบบของ OCR

1.3.2 ศึกษาทฤษฎีของสี

1.3.3 ศึกษาทฤษฎี Image Processing

1.3.4 เขียนโปรแกรมพัฒนา software ที่สามารถนำนำภาพที่มีตัวอักษรชนิดสีที่ต้องการมาแปลงให้อยู่ในรูปของข้อความในคอมพิวเตอร์

1.4 ขั้นตอนการดำเนินงาน

1.4.1 ศึกษารายละเอียดเกี่ยวกับ C++, AI, Image processing และหลักการของ OCR

1.4.2 รวบรวมข้อมูล

1.4.3 เขียนโปรแกรม

1.4.4 ทำการทดสอบโปรแกรม

1.4.5 แก้ไขข้อผิดพลาดในส่วนต่าง ๆ ของโครงงาน

1.4.6 ตรวจสอบรอบที่สอง

1.4.7 สรุปผลการทดลองและจัดทำรูปเล่มโครงงาน

1.5 แผนการดำเนินงาน

กิจกรรม	ปี 2548		ปี 2549								
	พ.ย.	ธ.ค.	ม.ค	ก.พ.	มี.ค.	เม.ย	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.
1. ศึกษารายละเอียดเกี่ยวกับ C++, AI, Image processing และหลักการของ OCR	←	→									
2. รวบรวมข้อมูล			←	→							
3. เขียนโปรแกรม					←	→					
4. ทำการทดสอบโปรแกรม							←	→			
5. แก้ไขข้อผิดพลาดในส่วนต่างๆ ของโครงงาน								←	→		
6. ตรวจสอบรอบที่สอง									←	→	
7. สรุปผลการทดลองและจัดทำรูปเล่มโครงงาน										←	→

1.6 ผลที่คาดว่าจะได้รับ

1.5.1 เข้าใจหลักการของ OCR

1.5.2 สามารถนำหลักการของ Image processing ประกอบกับหลักการทาง AI มาพัฒนาโปรแกรม Color OCR ได้ดีและมีประสิทธิภาพ

1.5.3 โปรแกรม Color OCR ที่พัฒนาสามารถนำไปประยุกต์และใช้งานได้

1.5.4 โปรแกรม OCR ที่พัฒนาจะเป็นประโยชน์ต่อผู้ที่สนใจและนำไปศึกษาต่อ

1.7 งบประมาณ

1.7.1 ค่าถ่ายเอกสารและค่าเข้าเล่มโครงงาน

1.7.2 ค่าอุปกรณ์ Hardware

1.7.4 ค่าหมึกพิมพ์

รวมเป็นเงิน 3,000 บาท (สามพันบาทถ้วน)

บทที่ 2

หลักการและทฤษฎี

ในโครงงานเรื่อง Color optical character recognition นี้จะศึกษาในเรื่องของ pattern recognition และหลักการต่าง ๆ ที่เกี่ยวข้อง คือ ทฤษฎีพื้นฐานของ OCR, Cognitive Sciences, โหมดสี CMYK, โหมดสี RGB, ASCII Code โดยจะมีรายละเอียด ต่าง ๆ ดังนี้

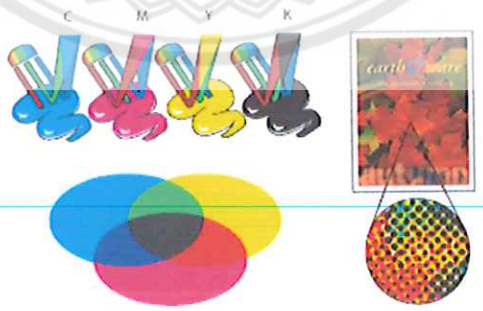
2.1 ทฤษฎีพื้นฐานของ OCR

OCR ย่อมาจาก Optical Character Recognition (การรู้จำอักขระด้วยแสง) ซึ่งได้รับความสนใจและพัฒนามานานกว่า 70 ปีแล้ว OCR เป็นการรู้จำรูปแบบตัวอักษร ซึ่งเป็นงานวิจัยในสาขาการรู้จำรูปแบบ (Pattern Recognition) เป็นเทคโนโลยีที่ส่งผลให้ระบบคอมพิวเตอร์สามารถระบุรูปแบบได้อย่างถูกต้อง คือ สามารถบอกได้ว่าตัวอักษรนั้นคือตัวอักษรอะไร

โปรแกรม OCR นี้ใช้ในการนำเข้าเอกสารที่เป็นตัวอักษร จากสิ่งพิมพ์ภายนอกผ่าน scanner แล้วเก็บเป็นภาพ แล้วจึงนำภาพที่ได้มาให้โปรแกรมวิเคราะห์และแปลงตัวอักษรในภาพนั้นให้เป็นรหัส ASCII ซึ่งวิธีการนี้ไม่จำเป็นต้องพิมพ์เอกสารเอง ทำให้การนำเข้าเอกสารทำได้สะดวกมากยิ่งขึ้น หลักการในการวิเคราะห์ตัวอักษรของโปรแกรมนี้ใช้วิธีการเก็บรหัสของกลุ่มจุดภาพของตัวอักษรมาตรฐานแต่ละตัวซึ่งได้จากการคำนวณแล้วโดยจะทำการเก็บลงไว้ในฐานข้อมูลเมื่อทำงานจริงก็จะนำรหัสจากฐานข้อมูลมาเทียบกับรหัสที่ได้จากภาพตัวอักษรที่ต้องการวิเคราะห์ เพื่อหารหัส ASCII อีกครั้ง

2.2 โหมดสี CMYK

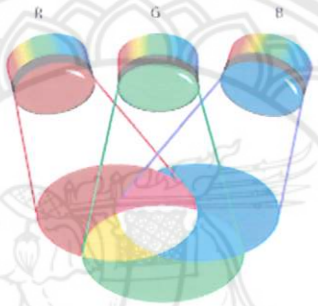
โหมดสีจาก Channel สี 4 สี คือ Cyan (ฟ้า), Magenta (บานเย็น), Yellow, Black แต่ละสีเก็บข้อมูล 8 บิต เป็นโหมดที่เหมาะสำหรับงานพิมพ์



รูปที่ 2.5 โหมดสี CMYK [4]

CMYK Model มีลักษณะพื้นฐานคือการดูดซับสีของกระดาษที่ซับเอาหมึกกึ่งโปร่งแสง (Translucent Inks) ซึ่งเป็นหมึกที่สามารถทำให้มองเห็นการทับซ้อนของสีจนเกิดเป็นสีใหม่ขึ้นมาได้ เช่น เมื่อพิมพ์สีเหลืองทับสีฟ้าบนกระดาษสีใหม่ที่เกิดขึ้นจะเป็นสีเขียว (ในขณะที่สีน้ำมันหรือสีที่มีคุณสมบัติทึบแสงจะไม่เกิดสีดังกล่าวขึ้นมา เช่น การทำงานของสีในระบบซิลค์สกรีน) สีหลักที่ใช้ใน CMYK Model เพื่อสร้างให้เกิดสีต่าง ๆ ขึ้นมานั้น เราเรียกว่า แม่สีทางการพิมพ์ มีทั้งหมด 4 สี คือ Cyan, Magenta, Yellow, Black อย่างไรก็ดีเมื่อมีการทับซ้อนกันมากขึ้นหรือทับกันครบทุกสีความโปร่งแสงก็จะลดลงจนไม่สามารถมองเห็นสีอื่นได้ นอกจากสีดำ เราจึงเรียกสีในแบบนี้ว่า สีแบบลบ (Subtractive Colors)

2.3 โหมดสีแบบ RGB



รูปที่ 2.6 โหมดสี RGB [4]

การผสมสีบนคอมพิวเตอร์นั้นอาศัย primary hues 3 สีคือ สีแดง สีเขียว และ สีน้ำเงิน หรือที่เราเรียกว่า RGB colors (Red-Green-Blue) ความเหมือนจริงของสีคอมพิวเตอร์นั้น ขึ้นอยู่กับในหนึ่งจุด (pixel) ของการแสดงผลนั้นใช้ระดับของสี หรือ color depths ว่ามีค่าเท่าไหร่ เช่นถ้าสี RGB มี color depths เป็น 8 planes นั่นคือ เราใช้ 8 บิตเก็บข้อมูลหนึ่งสี หมายความว่า เฉพาะ primary hues เช่น สีแดง สีเดียวก็จะมีความเป็นแดง อยู่ถึง $2^8 = 256$ ระดับ เมื่อเราผสมสี หนึ่งสี จาก แดง-เขียว-น้ำเงิน (RGB) เราต้องใช้สีแดง กี่ส่วนจาก 0 ถึง 255 ส่วน ใช้เขียว กี่ส่วนจาก 0 ถึง 255 และเช่นเดียวกัน สีน้ำเงินกี่ส่วนจาก 0 ถึง 255 สีที่เกิดขึ้นก็จะเกิดจากการผสมของสีทั้งสาม ในอัตราส่วนต่าง ๆ กัน ตัวอย่าง สีเหลืองธรรมดา เกิดจากการผสมสี แดง 255 ส่วน สีเขียว 255 ส่วน และสีน้ำเงิน 0 ส่วน ซึ่งระดับของสีแดง มีถึง 256 ระดับ สีเขียว 256 ระดับ สีน้ำเงิน 256 ระดับ ดังนั้น RGB ทั้งหมดใช้ 24 บิต (8+8+8) ในการแสดงสี RGB ของหนึ่งจุด (pixel) ซึ่งสามารถแสดงสีได้มากถึง 256x256x256 = 16.7 ล้านสี ตารางข้างล่างแสดงถึง จำนวนสีที่เป็นไปได้ ซึ่งขึ้นกับจำนวนบิต ที่ใช้กำหนดระดับของสี

ตารางที่ 2.1 แสดงจำนวนสีที่เป็นไปได้ซึ่งขึ้นกับจำนวนบิต

จำนวนบิตที่ใช้เก็บสีต่อหนึ่งจุด	Color Mode Name	จำนวนสีที่แสดงได้
1	Black and White	2
4	16-Color (EGA)	16
8	Pseudo Color	256
16	Hi-Color	65,536
24	True Color	16,777,216

บทที่ 3

วิธีการดำเนินการ

โปรแกรม Color Optical Character Recognition นี้สามารถใช้วิธีการประมวลผลภาพ และจดจำข้อมูลได้หลายวิธี การวางแผนขั้นตอนการทำจึงต้องพิจารณาวิธีที่ดี และเหมาะสมสำหรับการพัฒนา

โครงงานนี้จะใช้การเขียนโปรแกรมภาษา C# ในการพัฒนาเพราะมีฟังก์ชั่นที่ทำงานเกี่ยวกับภาพหลายฟังก์ชั่น และสะดวกในการแก้ไขส่วนที่ผิดพลาด

3.1 การศึกษาข้อมูล

ข้อมูลที่ศึกษาจะเป็นข้อมูลเกี่ยวกับวิธีการผสมสีของคอมพิวเตอร์ เทคนิควิธีการเก็บข้อมูลที่ได้จากรูปภาพ และเทคนิคการปรับแต่งรูปภาพในคอมพิวเตอร์

3.1.1 การผสมสีของคอมพิวเตอร์

ในทางคอมพิวเตอร์นั้น การที่ซอฟต์แวร์จะแสดงผลออกมาให้ผู้ใช้สามารถติดต่อกับฮาร์ดแวร์ได้นั้นต้องอาศัยมอนิเตอร์หรือหน้าจอแสดงผล ซึ่งในอดีตการแสดงผลสามารถแสดงได้เพียง 2 สีเท่านั้น คือ สีขาวและสีดำ จำนวนบิตที่ใช้แสดงผลนั้นใช้เพียง 1 บิตเท่านั้น แต่ปัจจุบันวิทยาการทางด้านคอมพิวเตอร์ได้มีการพัฒนาขึ้นอย่างรวดเร็ว ทำให้การแสดงผลทางหน้าจอคอมพิวเตอร์มีความหลากหลายมากขึ้น สามารถแสดงผลข้อมูลในรูปแบบของภาพเหมือนจริงทางหน้าจอคอมพิวเตอร์ได้

การแสดงผลของรูปภาพนี้อาศัยแสงสีหลัก 3 สีคือ สีแดง สีเขียว และสีน้ำเงิน แต่ละสีสามารถปรับเปลี่ยนค่าได้ 256 ค่า ทั้งสามสีซ้อนทับกันอยู่เป็นเลเยอร์ทำให้เกิดสีต่าง ๆ ได้ จุดที่ใช้แสดงค่าสีเรียกว่า pixel ในแต่ละ pixel สามารถแสดงค่าสีได้เท่ากับ 256x256x256 ค่าสี เป็นการแสดงผลทางหน้าจอคอมพิวเตอร์ที่มีมิติมากขึ้นจากเดิม ซึ่งจากหลักการส่วนนี้สามารถนำไปใช้แยกแสงสีแต่ละสี RGB (Red-Green-Blue) ออกจากภาพที่ต้องการศึกษาได้

3.1.2 เทคนิคการปรับแต่งภาพในคอมพิวเตอร์

ในโครงงานนี้ใช้เทคนิคในการเปลี่ยนภาพตัวอักษรสี RGB ให้เป็นภาพตัวอักษรขาวดำ (black & white) วิธีการทำคือ นำ pixel ของพื้นหลังเป็นค่าอ้างอิง แล้วนำ pixel ทุก ๆ pixel ในภาพมาเปรียบเทียบ หากว่า ค่าของ pixel ที่นำมาเปรียบเทียบมีค่าต่างจาก pixel อ้างอิงเกิน 10% ก็จะทำการบันทึกข้อมูลเป็นภาพใหม่โดยมีค่า pixel ตำแหน่งนั้นเป็นสีดำ ในทางกลับกันหาก pixel ที่นำมาเปรียบเทียบมีค่าใกล้เคียงหรือเท่ากับ pixel อ้างอิงก็จะบันทึกข้อมูลในภาพใหม่ที่ตำแหน่งนั้นเป็นสีขาว

จากวิธีการนี้ ภาพใหม่ที่ได้จากการเปรียบเทียบ pixel นั้นจะแสดงอยู่ในรูปแบบของภาพขาวดำที่มีพื้นหลังเป็นสีขาว และตัวอักษรเป็นสีดำ

3.1.3 เทคนิคการเก็บข้อมูลที่ได้จากรูปภาพ

เทคนิคที่ศึกษาเป็นเทคนิคพื้นฐาน คือ การแบ่งช่วงของภาพขาวดำทั้งทางแนวตั้งและแนวนอน ซึ่งในโครงงานนี้ได้ทดลองหาความเหมาะสมของการแบ่งช่วงได้ 20 ช่วงในแนวตั้งและแนวนอน เก็บข้อมูลทั้งหมด 440 ตำแหน่ง และเก็บค่าจากจุด ณ ตำแหน่งของเส้นตัดระหว่างแนวตั้งและแนวนอน

การเก็บข้อมูลจะเรียกเก็บค่าจากตำแหน่งที่เป็นจุดตัดของเส้นแบ่งโดยตรง เช่น หาก ณ ตำแหน่งเส้นแนวตั้งที่ 1 และตำแหน่งเส้นแนวนอนที่ 1 มีค่าของ pixel เท่ากับ 0 หรือสีดำ ก็จะเก็บข้อมูลในฐานข้อมูลตามที่ตกลง (ในโครงงานนี้จะเก็บข้อมูลเป็น “1” หากตำแหน่งนั้นมีสีดำ และเก็บข้อมูลเป็น “0” หากตำแหน่งนั้นมีสีขาว)

การบันทึกข้อมูลจะเป็นในรูปแบบของชุดข้อมูลที่ประกอบด้วยตัวเลข “0” และ “1” เรียงกันจำนวน 440 ตัว

3.1.3 เทคนิคการเก็บข้อมูลที่ได้จากรูปภาพ

เทคนิคที่ศึกษาเป็นเทคนิคพื้นฐาน คือ การแบ่งช่วงของภาพขาวดำทั้งทางแนวตั้งและแนวนอน ซึ่งในโครงงานนี้ได้ทดลองหาความเหมาะสมของการแบ่งช่วงได้ 20 ช่วงในแนวตั้งและ 22 ช่วงในแนวนอน เก็บข้อมูลทั้งหมด 440 ตำแหน่ง และเก็บค่าจากจุด ณ ตำแหน่งของเส้นตัดระหว่างแนวตั้งและแนวนอน

การเก็บข้อมูลจะเรียกเก็บค่าจากตำแหน่งที่เป็นจุดตัดของเส้นแบ่งโดยตรง เช่น หาก ณ ตำแหน่งเส้นแนวตั้งที่ 1 และตำแหน่งเส้นแนวนอนที่ 1 มีค่าของ pixel เท่ากับ 0 หรือสีดำ ก็จะเก็บข้อมูลในฐานข้อมูลตามที่ตกลง (ในโครงงานนี้จะเก็บข้อมูลเป็น “1” หากตำแหน่งนั้นมีสีดำ และเก็บข้อมูลเป็น “0” หากตำแหน่งนั้นมีสีขาว)

การบันทึกข้อมูลจะเป็นในรูปแบบของชุดข้อมูลที่ประกอบด้วยตัวเลข “0” และ “1” เรียงกันจำนวน 440 ตัวตัวที่ 441 จะเป็นค่าตัวอักษร เช่น

0011001100…….0001111A

1010110001…….0011101B

1101010100…….0010110C

3.1.4 เทคนิคการแม็ปตัวอักษรที่ได้จากรูปภาพกับฐานข้อมูล

นำชุดตัวเลข 0, 1 ที่ได้มาแม็ปกับฐานข้อมูลทีละชุดข้อมูลจนครบทุกชุดข้อมูลแล้ว ชุดตัวเลขที่เราได้เหมือนกับฐานข้อมูลชุดไหนมากที่สุด ก็จะโชว์ตัวอักษรที่อยู่หลังชุดข้อมูลออกมา เช่น

ฐานข้อมูลคือ

0011001100…….0001111A

1010110001…….0011101B

1101010100…….0010110C

ชุดตัวเลขที่ได้จากรูปภาพคือ

0011001100…….0001111

นำชุดตัวเลขที่ได้จากรูปภาพมาเทียบกับฐานข้อมูลทีละชุดข้อมูลคือ

ชุดข้อมูลที่ได้จากภาพ 0011001100…….0001111

ฐานข้อมูลชุดที่ 1 0011001100…….0001111A

เทียบข้อมูลทีละหลักถ้าเหมือนกันก็ให้นับเพิ่มทีละ1 ต่อไปก็นำชุดตัวเลขที่ได้จากรูปภาพไปเทียบกับฐานข้อมูลชุดที่ 2, 3 ไปเรื่อย ๆ จนถึงฐานข้อมูลชุดสุดท้ายแล้วก็เทียบกันว่าชุดตัวเลขที่ได้จากรูปภาพเหมือนฐานข้อมูลชุดไหนก็ให้แสดงตัวอักษรที่อยู่ในตำแหน่งที่ 441 ออกมา

3.2 หลักการและขั้นตอนการทำงานของโปรแกรม

หลักการของการสร้างโปรแกรมนี้ คือ การใช้เทคนิคการแยกแสงสีของภาพดิจิตอลออกเป็นชั้น แล้วนำสิ่งที่ได้จากหลักการแยกสี มาทำการหาภาพของตัวอักษร จากนั้นเปรียบเทียบข้อมูลที่ได้จากภาพกับข้อมูลของตัวอย่างในคอมพิวเตอร์ เพื่อแสดงให้เห็นว่า ภาพที่นำมาประมวลผลนั้นเป็นตัวอักษรตัวใดในคอมพิวเตอร์

ขั้นตอนการทำงาน

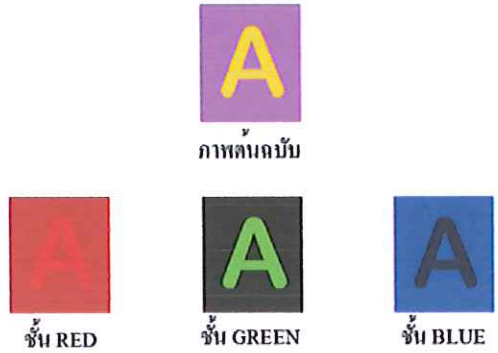
โปรแกรมรู้จำอักขระชนิดสีด้วยแสงมีการทำงานเป็นขั้นตอนดังนี้

1. ขั้นตอนการแยกสี RGB ของภาพอักษร
2. ขั้นตอนการแปลงภาพที่ได้จากการแยกสี RGB ไปเป็นภาพขาวดำ
3. ขั้นตอนการตีกรอบตัวอักษรเพื่อหาช่วงการเก็บข้อมูล และการเก็บข้อมูล
4. ขั้นตอนการเปรียบเทียบข้อมูลทั้ง 3 ชุดข้อมูล จากขั้นตอนที่ 3 เพื่อหาข้อมูลที่ถูกต้อง
5. ขั้นตอนการนำชุดข้อมูลของภาพตัวอักษรมาเปรียบเทียบกับฐานข้อมูลของตัวอักษร
6. ขั้นตอนการบันทึกข้อมูลลงฐานข้อมูล กรณีที่ไม่มีตัวอักษรที่ได้จากภาพในฐานข้อมูล

3.2.1 ขั้นตอนการแยกสี RGB ของภาพอักษร

เริ่มจากการนำภาพของตัวอักษรสีที่มีสีของตัวอักษรต่างกับสีของพื้นหลัง มาแยกเป็นชั้นของสี โดยการแยกสีนี้ได้มาจากหลักการของการผสมสีบนคอมพิวเตอร์ที่อาศัย primary hues 3 สีคือ สีแดง สีเขียว และสีน้ำเงิน หรือที่เรา เรียกว่า RGB colors ตัวอย่าง สีเหลืองธรรมดา เกิดจากการผสมสี แดง 255 ส่วน สีเขียว 255 ส่วน และสีน้ำเงิน 0 ส่วน ซึ่งแต่ก่อนสีที่ใช้แสดงผลข้อมูลของคอมพิวเตอร์นั้นมีแค่ 2 สี หรือใช้จำนวนบิตในการเก็บค่าของสีเพียง 1 บิตเท่านั้น

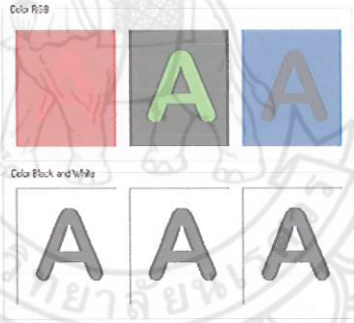
ผลที่ได้จากการแยกสีนี้ทำให้ทราบว่า ในแต่ละชั้นนั้น บางกรณีสามารถแยกภาพของตัวอักษรออกจากพื้นหลังได้ชัดเจน ขณะที่ภาพต้นฉบับนั้นมีสีตัวอักษรและพื้นหลังใกล้เคียงกัน ดังรูปที่ 3.1



รูปที่ 3.1 ผลการแยกชั้นสี RGB

3.2.2 ขั้นตอนการแปลงภาพที่ได้จากการแยกสี RGB ไปเป็นภาพขาวดำ

หลังจากที่แยกชั้นของภาพตัวอักษรแล้ว จะได้ทั้งหมด 3 ชั้น โดยในแต่ละชั้นจะมีค่าของสีต่างกันออกไป เช่น ชั้นของสีแดง ก็จะมีค่าของสีเขียวและสีน้ำเงินเท่ากับ 0 ค่าของสีแดงก็จะอยู่ในช่วง 0 – 255 ดังนั้นการแปลงชั้น ทั้ง 3 ชั้น ให้เป็นสีขาวดำ ทำให้สามารถแบ่งพื้นหลังกับตัวอักษรออกได้อย่างชัดเจน ดังรูปที่ 3.2



รูปที่ 3.2 ผลการแปลงภาพสีเป็นขาวดำ

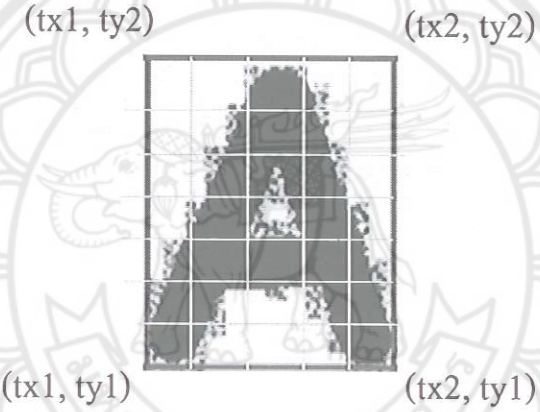
วิธีการแปลงคือ จะทำการเทียบสี pixel แรก (0, 0) กับ pixel ถัดไปว่าใกล้เคียงกันเกิน 90% หรือไม่ หากใกล้เคียงกัน pixcl นั้นก็จะเป็น pixcl ของพื้นหลัง ถ้าต่างกันก็จะเป็น pixcl ของตัวอักษร ในที่นี่จะให้ค่า pixel ของพื้นหลังเท่ากับ (0, 0, 0) และ pixel ของตัวอักษรเท่ากับ (255, 255, 255) หลังจากการแปลงภาพสีเป็นขาวดำ ก็จะได้ภาพขาวดำ จำนวน 3 ภาพออกมาเพื่อนำไปเปรียบเทียบในขั้นตอนต่อไป

3.2.3 ขั้นตอนการตีกรอบตัวอักษรเพื่อหาช่วงการเก็บข้อมูล และการเก็บข้อมูล

การตีกรอบตัวอักษรจะเริ่มจากการค้นหา pixel สีดำในแนวแกน X เมื่อเจอ pixel แรกจะให้ tx1, tx2, ty1, ty2 มีค่าเท่ากับพิกัดของจุด pixel นั้น และจะค้นหาต่อไปจนหมดทุก pixel ในรูปภาพ เมื่อพบ

จุดสีดำก็จะนำพิกัดของจุดนั้นมาเทียบกับค่ากับ tx1, tx2, ty1, ty2 ถ้าจุด tx ที่ได้ในพิกัดใหม่ น้อยกว่าค่า tx1 จะทำการเปลี่ยนค่า tx1 ให้เท่ากับ tx และถ้า tx ที่ได้ในพิกัดใหม่มีค่ามากกว่า tx2 จะทำการเปลี่ยนค่า tx2 ให้เท่ากับ tx สำหรับการเปรียบเทียบค่า ty ก็จะทำในทำนองเดียวกับการเปรียบเทียบค่า tx ส่วนการเก็บข้อมูลของโปรแกรมจะเก็บโดยแบ่งช่วงของภาพระหว่างค่า tx1 และ tx2 เป็น 20 ช่วง ty1 และ ty2 เป็น 22 ช่วง ซึ่งจะได้จุดตัดของเส้นแบ่งทั้งหมดประมาณ 440 จุด จากนั้นก็จะนำค่าของ pixel ที่อยู่ ณ ตำแหน่งของจุดตัดนั้นเก็บไว้ในชุดข้อมูล โดยถ้าจุดนั้นมีค่าของ RGB เท่ากับ (0, 0, 0) ก็จะบันทึกข้อมูลเป็น "0" ในชุดข้อมูล แต่หากค่า ณ จุดนั้นมีค่า (255, 255, 255) จะบันทึกข้อมูลเป็น "1"

ผลจากการแบ่งช่วงการบันทึกข้อมูลนั้นเป็นชุดข้อมูลแบบ binary คือมีตัวเลข "0" และ "1" เป็นข้อมูล และ 1 ชุดข้อมูล คือ 1 ตัวอักษร



รูปที่ 3.3 ตัวอย่างการเก็บข้อมูลจากรูปภาพ

3.2.4 ขั้นตอนการเทียบข้อมูลทั้ง 3 ชุดข้อมูล จากขั้นตอนที่ 3 เพื่อหาข้อมูลที่ถูกต้อง

จากการตีกรอบเก็บข้อมูล จะได้ชุดข้อมูลออกมา 3 ชุดที่เป็นชุดข้อมูลของชั้นสี RGB ทั้ง 3 ชั้น นี้จะมีข้อมูลใกล้เคียงกัน ซึ่งในขั้นตอนนี้จะเป็นขั้นตอนของการเปรียบเทียบข้อมูลเพื่อจะหาข้อมูลที่ถูกต้องที่สุดในการนำไปเก็บในฐานข้อมูล

การเปรียบเทียบข้อมูลทำได้โดย นำชุดข้อมูลทั้ง 3 ชุดที่ได้จากชั้นสี RGB มาเปรียบเทียบกันทีละตัวและจะบันทึกข้อมูลที่เปรียบเทียบได้จาก 2 ใน 3 ที่เท่ากันของชุดข้อมูล เช่น มีชุดข้อมูลดังรูปที่ 3.4

ชั้น Red	:1101001110111010101010101000
ชั้น Green	:1101001110111010100010101000
ชั้น Blue	:1000000110110010100010101000
ผลรวมของ 3 ชั้น	:3202002330332030301030303000
ข้อมูลที่จะนำไปเทียบ	:1101001110111010101010101000

รูปที่ 3.4 การเปรียบเทียบชุดข้อมูล RGB

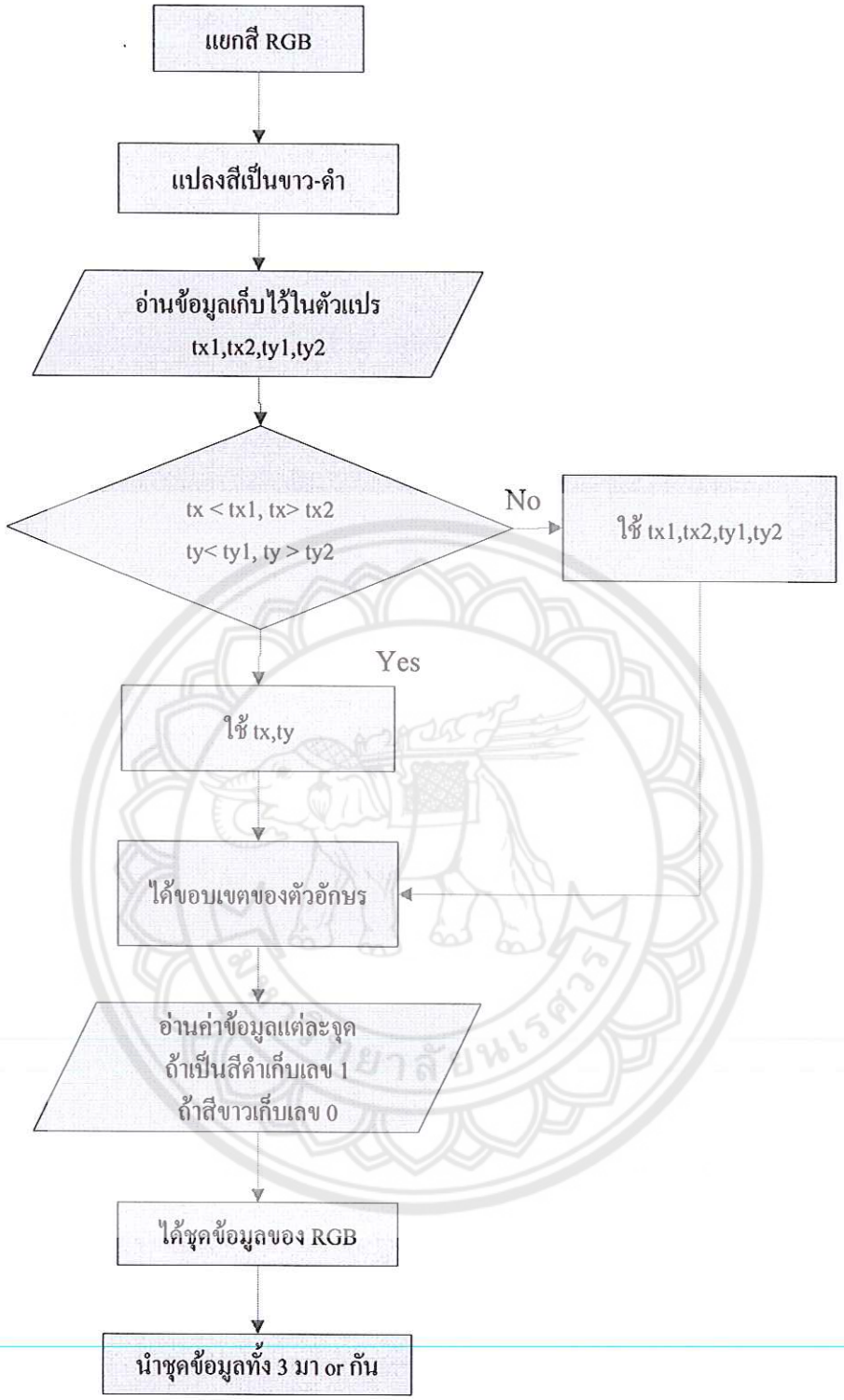
ข้อมูลตัวแรกของทั้ง 3 ชั้น มีค่าเดียวกันคือ 1 โปรแกรมก็จะทำการบันทึกข้อมูลลงในฐานข้อมูลตัวแรกเป็น 1 ส่วนข้อมูลตัวที่ 2 ในชุดข้อมูลของชั้นสี B เป็น 0 ส่วนข้อมูลของชั้นสี R และ G เป็น 1 ดังนั้นการบันทึกข้อมูลลงในฐานข้อมูลก็จะเป็น 1 วิธีการนี้จะทำให้ได้ข้อมูลที่ใกล้เคียงข้อมูลจริงมากที่สุด

3.2.5 ขั้นตอนการนำชุดข้อมูลของภาพตัวอักษรเปรียบเทียบกับฐานข้อมูลของตัวอักษรที่เตรียมไว้

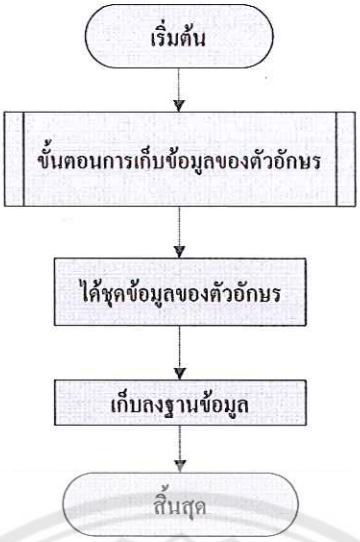
หลังจากการเปรียบเทียบข้อมูลจะได้ชุดข้อมูลของตัวอักษรนั้นมา ต่อไปจะเป็นการเปรียบเทียบชุดข้อมูลตัวอักษรกับฐานข้อมูลตัวอักษรทั้งหมดที่เป็นไปได้ (ในโครงงานนี้จะทดลองเฉพาะตัวอักษรภาษาอังกฤษตัวพิมพ์ใหญ่เท่านั้น) เมื่อเทียบแล้วชุดข้อมูลตรงกับข้อมูลชุดไหนมากที่สุด ก็จะแสดงตัวอักษรนั้นออกมาในรูปของอักษรในคอมพิวเตอร์

3.2.6 ขั้นตอนการบันทึกข้อมูลลงฐานข้อมูลกรณีที่ไม่มีชุดข้อมูลตัวอักษรในฐานข้อมูล

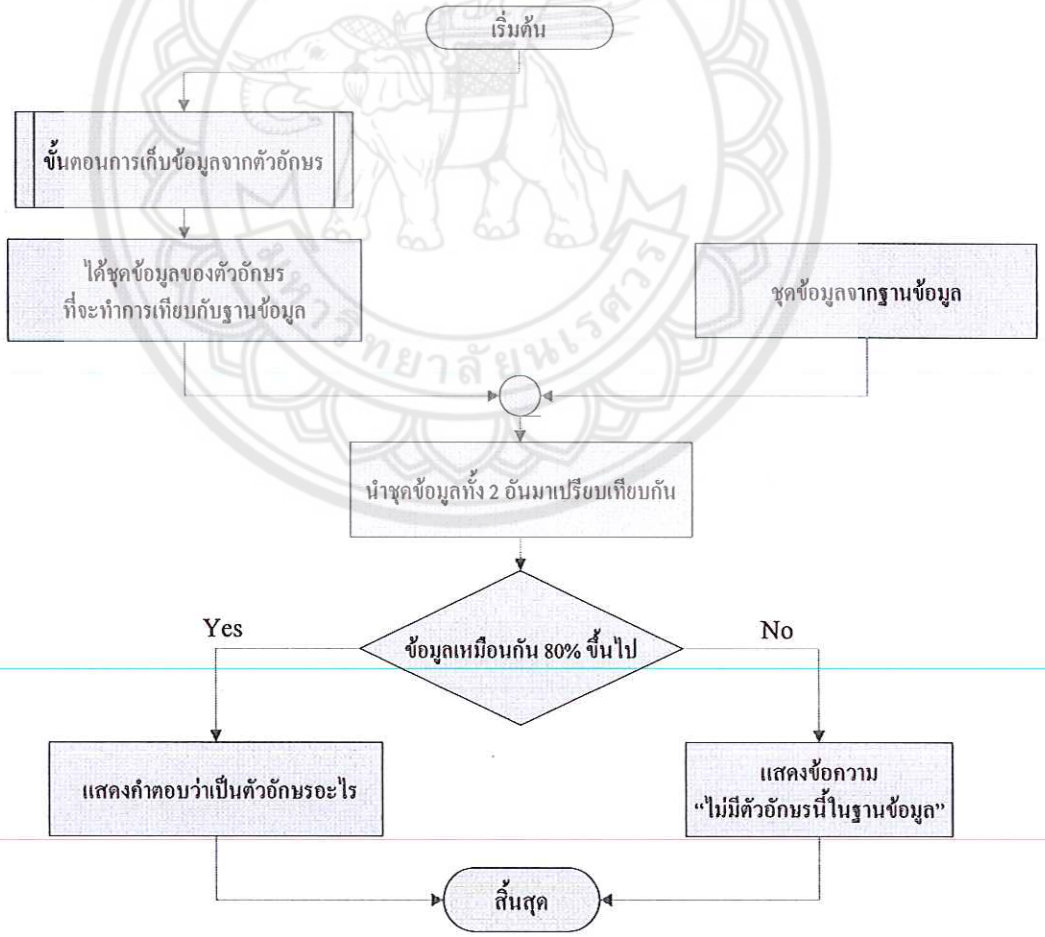
ในกรณีที่ไม่มีฐานข้อมูลของตัวอักษรที่จะทำการตรวจสอบ โปรแกรมก็จะทำการบันทึกข้อมูลใหม่เพิ่มลงในฐานข้อมูล และจะต่อท้ายชุดข้อมูลด้วยตัวอักษรที่ผู้ใช้เป็นคนป้อนข้อมูล เช่นไม่มีชุดข้อมูลของตัวอักษร A ในฐานข้อมูล โปรแกรมก็จะบันทึกชุดข้อมูลที่ได้จากขั้นตอนที่ 3 ลงในฐานข้อมูล และตามด้วยตัวอักษร “A” ที่ผู้ทำการทดลองเป็นผู้ป้อนข้อมูล เพื่อเป็นตัวอ้างอิง และนำไปใช้แสดงเมื่อมีข้อมูลชุดใหม่ที่เหมือนกันนำมาเปรียบเทียบ



รูปที่ 3.5 ขั้นตอนการเก็บข้อมูลจากตัวอักษร



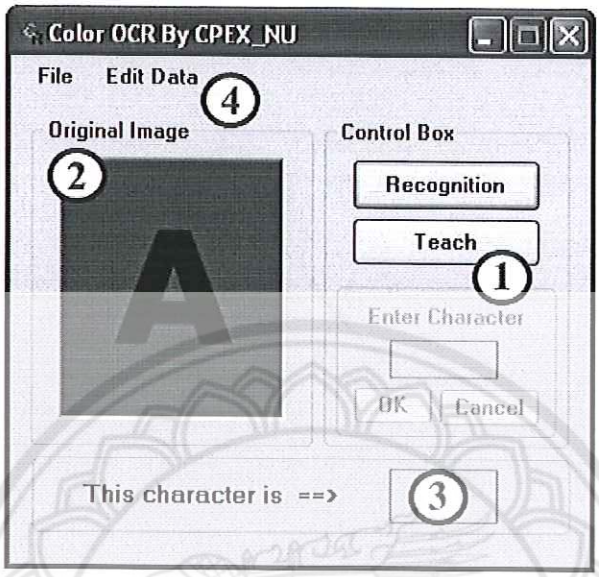
รูปที่ 3.6 วิธีการเก็บข้อมูลลงฐานข้อมูล



รูปที่ 3.7 ขั้นตอนการเทียบข้อมูล

3.3 การออกแบบส่วนประกอบของโปรแกรม

หลังจากวางแผนการทำงานของโปรแกรมแล้ว ต่อไปก็เป็นกระบวนการของการออกแบบหน้าตาของโปรแกรมให้เหมาะสม ตามรูปที่ 4



รูปที่ 3.7 หน้าตาส่วนประกอบของโปรแกรม

3.3.1 ส่วนควบคุม

ส่วนนี้เป็นส่วนที่รวบรวมการควบคุมต่าง ๆ คือ การรู้จำ (recognition) การบันทึกข้อมูลใหม่ (teach) และส่วนของการรับข้อมูลที่ต้องการบันทึกลงในฐานข้อมูล ในส่วนที่เป็นการรับข้อมูลนั้นจะขึ้นอยู่กับปุ่มบันทึกข้อมูลถ้าปุ่มบันทึกข้อมูลไม่ถูกกระทำ ส่วนของการรับข้อมูลก็จะถูก disable ไว้เพื่อป้องกันการเพิ่มข้อมูลตัวอักษรโดยที่ไม่มีฐานข้อมูลมีรองรับ

3.3.2 ส่วนแสดงภาพตัวอักษรต้นฉบับ

ส่วนนี้เป็นกล่องแสดงรูปภาพที่กำหนดขนาดตายตัวคือ 130x150 pixel ใช้แสดงภาพต้นฉบับที่จะนำมาทำการทดลอง และถึงแม้ว่าภาพที่นำมาทำการทดลองจะมีขนาดใหญ่หรือเล็กกว่ากล่องแสดงภาพ โปรแกรมก็จะทำการย่อขยายภาพให้เท่ากับกล่องแสดงภาพ

3.3.3 ส่วนแสดงผลลัพธ์

ในส่วนนี้จะทำการแสดงผลลัพธ์ที่ได้จากการประมวลผลและเทียบรูปภาพกับฐานข้อมูลแล้ว โดยจะนำตัวอักษรในฐานข้อมูลที่บันทึกไว้ท้ายชุดข้อมูลมาแสดงผลรูปแบบของ text ในกล่องแสดงข้อความ และสามารถคัดลอกตัวอักษรไปใช้ประโยชน์อื่น ๆ ได้

3.3.4 ส่วนของการโหลดฐานข้อมูลเก่า

ฐานข้อมูลเก่าคือ ข้อมูลของตัวอักษรที่จะนำมาใช้เป็นตัวอ้างอิงในการเปรียบเทียบ ซึ่งในโปรแกรมนี้จะทำการบันทึกข้อมูล train เอาไว้เพื่อเป็นต้นแบบในการทำการทดลอง และจะเก็บเป็นชุดตามแบบ (font) ของตัวอักษร

3.4 การเขียนโปรแกรม

โครงงานนี้ใช้ภาษา C# ในการพัฒนา เพราะฟังก์ชั่นต่าง ๆ ที่เกี่ยวข้องกับกราฟฟิกนั้นมีมาในตัวของภาษา C# แล้ว ไม่จำเป็นต้องเรียกใช้ฟังก์ชั่นเหมือนภาษาอื่น ๆ

3.4.1 การแปลงภาพสีเป็นภาพขาวดำ

วิธีการแปลงภาพสีเป็นขาวดำกระทำโดยการนำค่า pixel จุดแรกของภาพเป็นค่าอ้างอิง จากนั้นนำค่าจาก pixel ถัดไปมาเปรียบเทียบถ้ามีค่าสูง-ต่ำจาก pixel แรกไม่เกิน 30 ก็กำหนดให้เก็บข้อมูลเป็นสีขาว หรือพื้นหลัง แต่หากเปรียบเทียบแล้วเกินก็จะเก็บข้อมูลเป็นสีดำ

ค่าที่ได้จากการเปลี่ยนเป็นขาวดำของแต่ละชั้นจะถูกเก็บไว้ชั่วคราวในไฟล์ rBuf.OCR, gBuf.OCR และ bBuf.OCR เพื่อใช้ในการเปรียบเทียบต่อไป

3.4.2 การเก็บข้อมูลลงในฐานข้อมูล

โปรแกรมจะนำค่าจากไฟล์ rBuf.OCR, gBuf.OCR และ bBuf.OCR มาทีละตำแหน่งจากนั้นทำการบันทึกลงในไฟล์ใหม่โดย หากชั้นใดชั้นหนึ่งมีค่าเป็นสีดำ การบันทึกค่าในไฟล์ใหม่จะต้องเป็นสีดำ แต่หากทั้งสามชั้นมีค่าเป็นสีขาวจึงจะบันทึกค่าเป็นสีขาว

3.4.3 ปุ่มการเรียกฐานข้อมูลเก่า และสร้างฐานข้อมูลใหม่

ปุ่มทั้งสองนี้จะอยู่ภายในส่วนของ menu bar การเรียกฐานข้อมูลเก่าจะกระทำเมื่อผู้ใช้ต้องการเรียกฐานข้อมูลที่มีอยู่แล้วมาทดลองกับตัวอักษรที่ต้องการทราบ ปุ่มคำสั่งนี้เมื่อถูกเรียกใช้งาน จะแสดงหน้าต่างสำหรับเปิดไฟล์ฐานข้อมูล การเรียกฐานข้อมูลสามารถเรียกได้เฉพาะไฟล์นามสกุล .OCR, .dat หรือ .txt เท่านั้น

การสร้างฐานข้อมูลใหม่ จะกระทำเมื่อผู้ใช้ต้องการสร้างฐานข้อมูลขึ้นมาใหม่นอกจากฐานข้อมูลที่มีอยู่เดิม ปุ่มคำสั่งนี้จะแสดงหน้าต่างสำหรับบันทึกไฟล์ฐานข้อมูลใหม่โดยผู้ใช้สามารถเลือกสถานที่ที่บันทึกได้ และเมื่อบันทึกแล้วจะได้ฐานข้อมูลใหม่ที่มีชื่อว่า .OCR

3.4.3 ปุ่ม Recognition

เป็นปุ่มที่ใช้บอกว่าภาพที่นำมาทดลองคือภาพตัวอักษรอะไร เมื่อมีการกระทำกับปุ่มนี้ตัวอักษรที่ตรงกับภาพจะถูกแสดงออกทาง text box

บทที่ 4

ผลการทดลอง

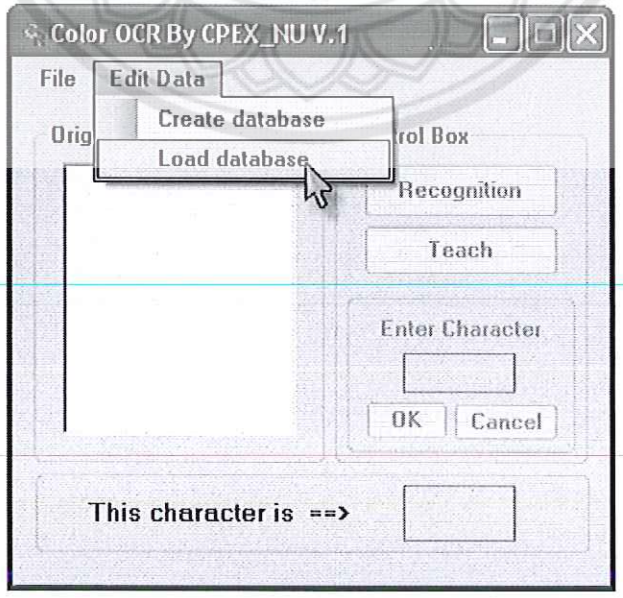
ในบทนี้จะอธิบายการใช้โปรแกรมและผลการทดลองในแบบต่าง ๆ ไม่ว่าจะเป็นการเปลี่ยนสี-พื้นกับสีตัวอักษร การย้ายตำแหน่งตัวอักษร การย่อขยายตัวอักษร การย้ายตำแหน่งตัวอักษร การเอียงของตัวอักษร

4.1 วิธีการใช้งานโปรแกรม

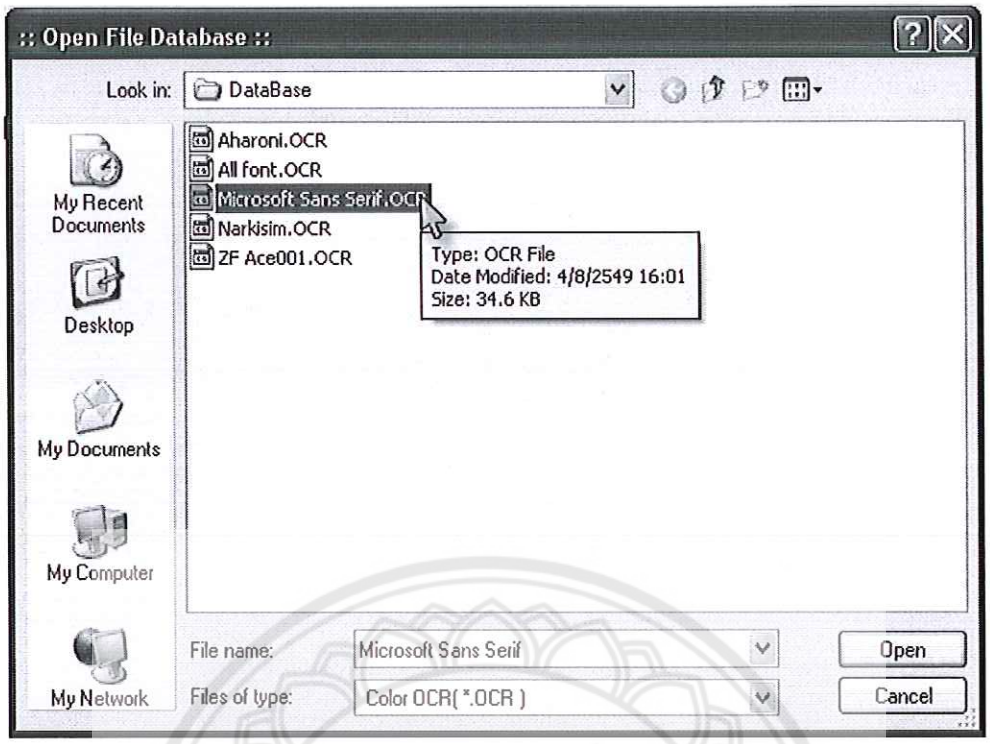
ขั้นตอนการใช้งานโปรแกรม

1. ทำการโหลดฐานข้อมูลที่สร้างไว้ ซึ่งเป็นไฟล์นามสกุล *.OCR
2. เปิดไฟล์ภาพที่ต้องการทดสอบ
3. กดปุ่ม Recognition เพื่อแสดงผลลัพธ์ของตัวอักษรในไฟล์ภาพ
4. เมื่อต้องการเพิ่มฐานข้อมูลตัวอักษรใหม่ ให้ใส่ตัวอักษรในช่อง Enter Character แล้วกดปุ่ม OK

การใช้งานโปรแกรม เริ่มต้นต้องทำการโหลดฐานข้อมูลขึ้นมาโดยการกดปุ่ม **Edit Data -> Load database** แล้วทำการเลือกไฟล์ฐานข้อมูล ดังรูปที่ 4.1 และ 4.2

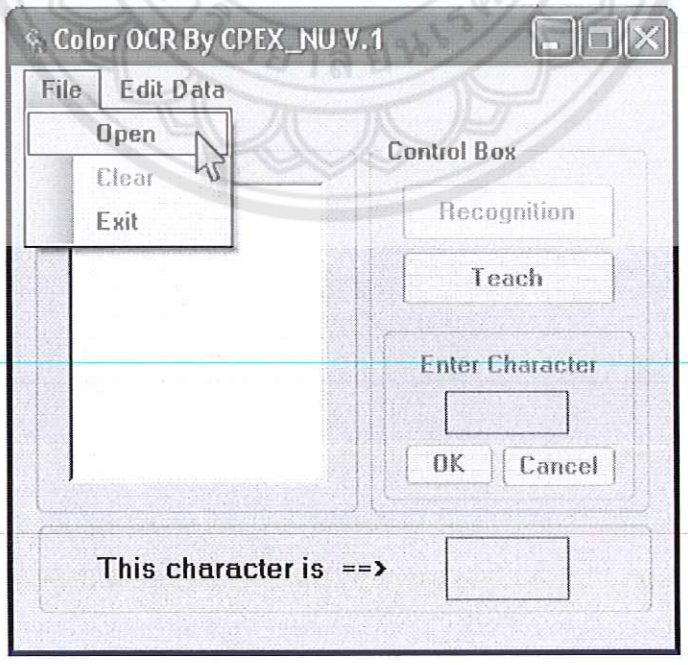


รูปที่ 4.1 การโหลดไฟล์ฐานข้อมูล

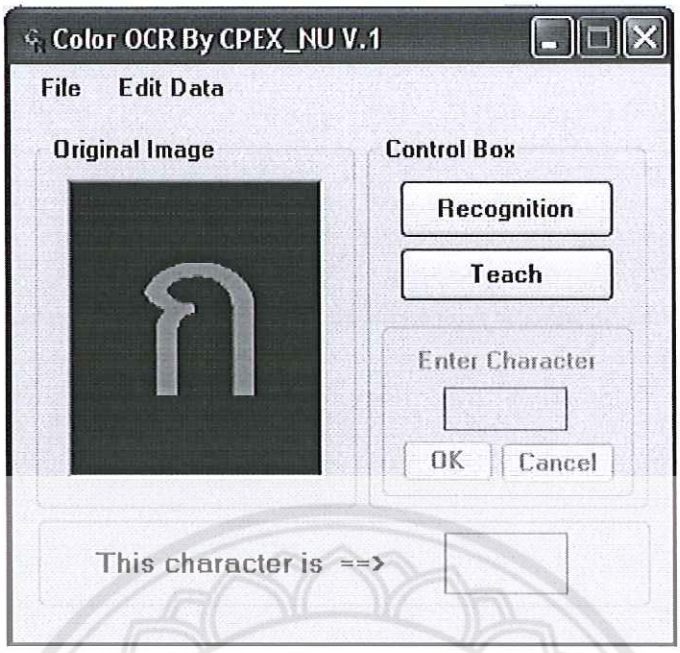


รูปที่ 4.2 การเลือกไฟล์ฐานข้อมูล

เมื่อทำการโหลดฐานข้อมูลขึ้นมาเสร็จ ขั้นต่อไปทำการเปิดไฟล์รูปภาพที่มีตัวอักษรโดยการกดที่ปุ่ม **File -> Open** แล้วทำการเลือกไฟล์รูปภาพ เมื่อได้รูปภาพที่มีตัวอักษรแล้วก็จะได้ดังรูปที่ 4.3 และ 4.4

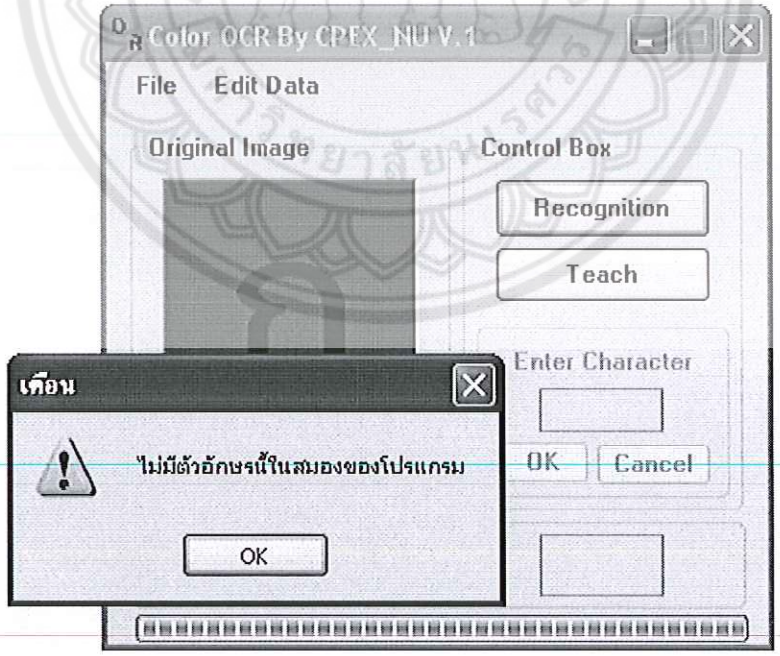


รูปที่ 4.3 การเปิดไฟล์รูปภาพที่มีตัวอักษร



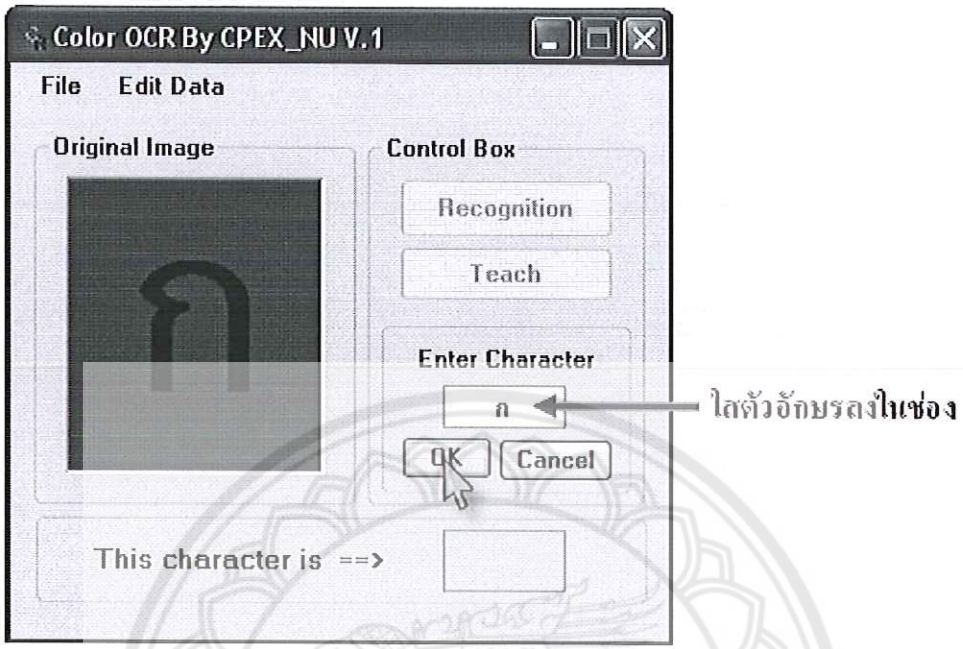
รูปที่ 4.4 รูปที่ได้หลังจากการเปิดไฟล์รูปภาพ

กรณีที่ฐานข้อมูลไม่มีตัวอักษรที่ตรงกับตัวอักษรที่อยู่ในรูปภาพก็จะโชว์ดังรูปที่ 4.5

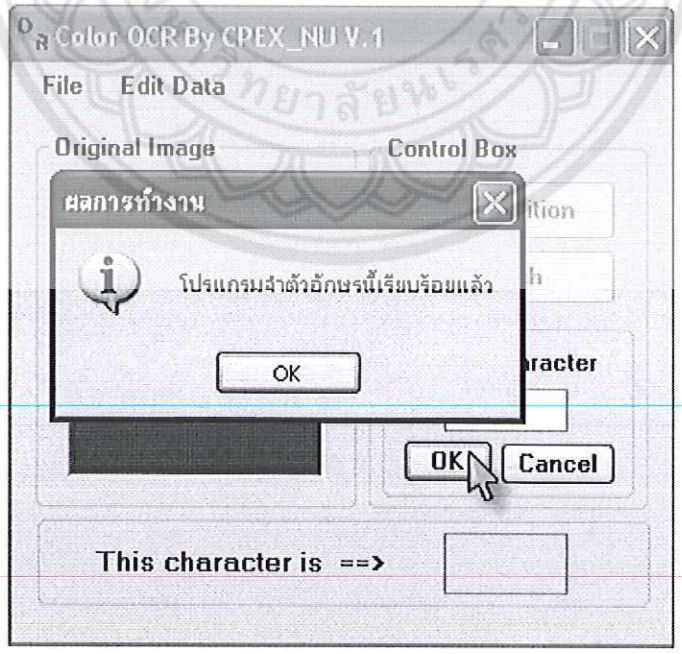


รูปที่ 4.5 แสดงหน้าต่างเมื่อไม่มีตัวอักษรตรงกับรูปตัวอักษร

จากนั้นก็ต้องทำการเก็บข้อมูลตัวอักษรนั้นเข้าไปในฐานข้อมูลโดยการกดที่ปุ่ม **Teach** ใส่ตัวอักษรลงไปในช่อง แล้วก็กดปุ่ม OK ดังรูป และเมื่อกดปุ่ม OK เสร็จก็จะได้ดังรูป 4.6 และ 4.7

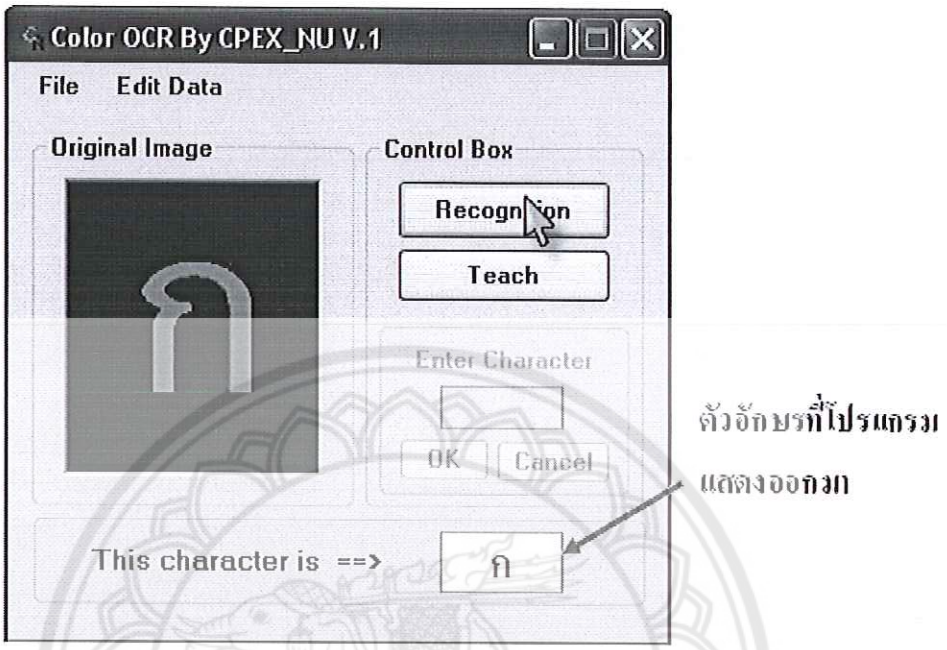


รูปที่ 4.6 แสดงการ Teach ให้กับฐานข้อมูล



รูปที่ 4.7 แสดงเมื่อทำการ Teach ให้กับฐานข้อเสร็จ

หลังจากที่ได้ทำการ teach ให้กับฐานข้อมูลแล้วกดปุ่ม Recognition แล้วโปรแกรมก็จะแสดงตัวอักษรที่อยู่ในรูปภาพออกมาเป็น text file ดังรูปที่ 4.8

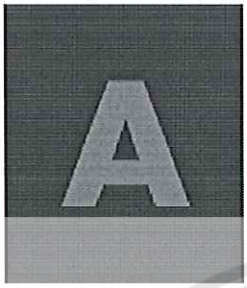


รูปที่ 4.8 แสดงผลการทำงานของโปรแกรม

4.2 ผลการทดลอง

ในการทดลองรูปภาพทั้งหมดที่ใช้นำมาจากการสร้างภาพด้วยโปรแกรม Photoshop CS และได้ทำการทดลองต่าง ๆ ดังนี้

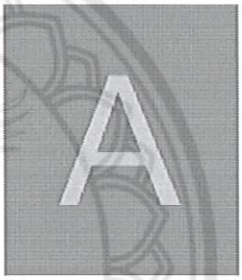
การทดลองที่ 1 ทดลองอักษร 4 font โดยการเก็บลงฐานข้อมูลแค่ครั้งเดียว แล้วทำการเปลี่ยนสีพื้นและสีตัวอักษร เพื่อทดสอบโปรแกรม



font Aharoni

font Narkisim

font ZF Ace001

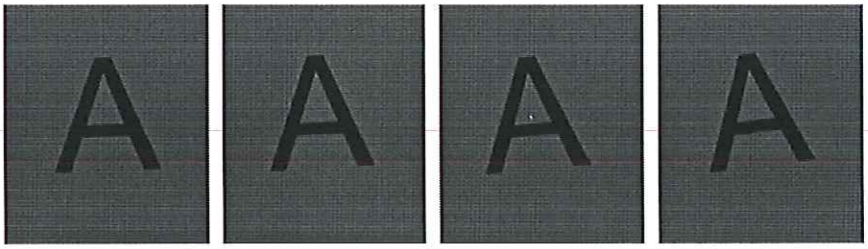


Microsoft Sans Serif

รูปที่ 4.9 ตัวอักษรที่ใช้ในการทดลองที่ 1

การทดลองที่ 2 ทดลองเอียงตัวอักษร เอียงซ้ายและขวา 4 องศา ได้แก่ 2, 3, 5, 7 **องศา**

ตัวอักษรที่ Teach.



เอียง 2 องศา **เอียง 3 องศา 1** **เอียง 5 องศา** **เอียง 7 องศา**

รูปที่ 4.10 ตัวอักษรที่ใช้ในการทดลองที่ 2

<u>การทดลองที่ 3</u> ทดลองย่อขยายตัวอักษร จากที่ Teach ตัวอักษรเก็บไว้ในฐานข้อมูลขนาด 100 แล้วย่อตัวอักษรขนาด 90, 80, 70, 60, 50, 40, 30, 20, 10 และขยายตัวอักษรขนาด 110, 120, 130

ตัวอักษรที่ Teach.

ตัวอักษรขนาด 10

ตัวอักษรขนาด 20

ตัวอักษรขนาด 30

ตัวอักษรขนาด 40

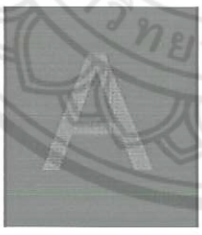
ตัวอักษรขนาด 50

ตัวอักษรขนาด 60

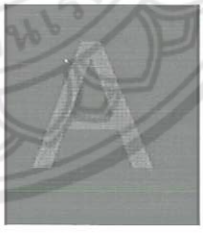
ตัวอักษรขนาด 70

ตัวอักษรขนาด 80

ตัวอักษรขนาด 90



ตัวอักษรขนาด 110



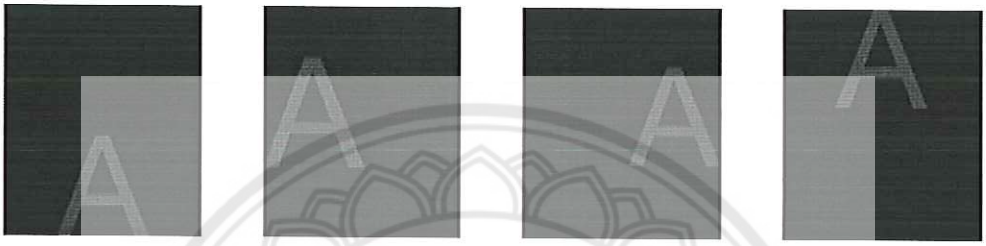
ตัวอักษรขนาด 120

ตัวอักษรขนาด 130

รูปที่ 4.11 ตัวอักษรที่ใช้ในการทดลองที่ 3

การทดลองที่ 4 ทดลองย้ายตำแหน่งตัวอักษร จากที่ Teach ตัวอักษรเก็บไว้ในฐานข้อมูล ตัวอักษรอยู่ตรงกลาง

ตัวอักษรที่ Teach.

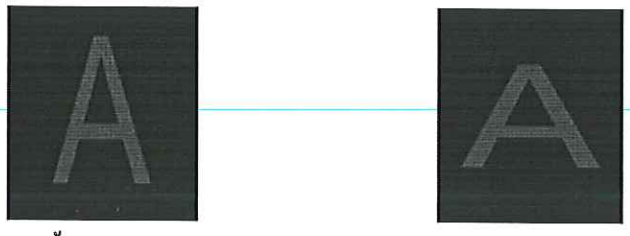


ย้ายตำแหน่งลง ย้ายตำแหน่งไป ย้ายตำแหน่งไป ย้ายตำแหน่งไป

รูปที่ 4.12 ตัวอักษรที่ใช้ในการทดลองที่ 4

การทดลองที่ 5 ทดลองยืดตัวอักษรจากที่ Teach ตัวอักษรเก็บไว้ในฐานข้อมูลเป็นตัวธรรมดา

ตัวอักษรที่ Teach.



ยืดขึ้นด้านบน 1.5 ยืดออกด้านข้าง 1.5

รูปที่ 4.13 ตัวอักษรที่ใช้ในการทดลองที่ 5

ห้องสมุดคณะวิศวกรรมศาสตร์

i7080496. e.2
5000061
จร.
ส852ก.
2549

<u>การทดลองที่ 6.1</u> ทดลองให้สีพื้น มี 1 สี และตัวอักษรเป็นเฉดสี

รูปที่ 4.14 ตัวอักษรที่ใช้ในการทดลองที่ 6.1

<u>การทดลองที่ 6.2</u> ทดลองพื้นเป็นเฉดสี และตัวอักษร 1 สี

รูปที่ 4.15 ตัวอักษรที่ใช้ในการทดลองที่ 6.2

ตารางที่4.1 ผลการทดลองที่ 1 ทดสอบโปรแกรม จากพื้นหลังสามสีคือ แดง เขียว และน้ำเงิน และตัวหนังสือสี ดำ แดง เขียว และน้ำเงิน (ตัวอักษรละ 12 แบบ)

Fonts / ตัวอักษร	font Aharoni ถูก (%)	font Narkisim ถูก (%)	font ZF Ace001 ถูก (%)	Microsoft Sans Serifถูก (%)
A	100	100	100	100
B	100	100	100	100
C	100	100	100	100
D	100	100	100	100
E	100	100	100	100
F	100	100	100	100
G	100	100	100	100
H	100	100	100	100
I	100	100	100	100
J	100	100	100	100
K	100	100	100	100
L	100	100	100	100
M	100	100	100	100
N	100	100	100	100
O	100	100	100	100
P	100	100	100	100
Q	100	100	100	100
R	100	100	100	100
S	100	100	100	100
T	100	100	100	100
U	100	100	100	100
V	100	100	100	100
W	100	100	100	100
X	100	100	100	100
Y	100	100	100	100
Z	100	100	100	100
รวมถูก	100%	100%	100%	100%

ตารางที่ 4.2 ผลการทดลองที่ 2 ทดสอบโปรแกรม ตัวอักษรเอียงไปจากตัวอักษรที่สอนให้กับโปรแกรม โดยใช้ Microsoft Sans Serif ใช้พื้นสีแดง และตัวอักษรสีดำ

ตัวอักษร	เอียงซ้าย				เอียงขวา			
	2 องศา	3 องศา	5 องศา	7 องศา	2 องศา	3 องศา	5 องศา	7 องศา
A	√	√	√	√	√	√	√	√
B	√	√	√	×	√	√	×	×
C	√	√	√	√	√	√	√	√
D	√	√	×	×	√	√	√	×
E	√	√	√	×	√	√	√	×
F	√	√	√	√	√	√	√	√
G	√	√	√	√	√	√	√	√
H	√	√	√	√	√	√	√	√
I	√	√	×	×	√	√	×	×
J	√	√	√	√	√	√	√	√
K	√	√	√	√	√	√	√	√
L	√	√	√	√	√	√	√	√
M	√	√	√	√	√	√	√	√
N	√	√	√	√	√	√	√	√
O	√	√	√	√	√	√	√	√
P	√	√	√	√	√	√	√	√
Q	√	√	√	√	√	√	√	√
R	√	√	√	√	√	√	√	√
S	√	√	√	√	√	√	√	√
T	√	√	√	√	√	√	√	√
U	√	√	√	√	√	√	√	√
V	√	√	√	√	√	√	√	√
W	√	√	√	√	√	√	√	√
X	√	√	√	√	√	√	√	√
Y	√	√	√	√	√	√	√	√
Z	√	√	√	√	√	√	√	√

ตารางที่ 4.3 ผลการทดลองที่ 3 ทดสอบโปรแกรม ย่อขยายตัวอักษร โดยใช้ font Microsoft Sans Serif ใช้พื้นสีน้ำเงิน และตัวอักษรสีแดง

ตัวอักษร	size 10	size 20	size 30	size 40	size 50	size 60	size 70	size 80	size 90	size 110	size 120	size 130
A	√	√	√	√	√	√	√	√	√	√	√	√
B	×	√	√	√	√	√	√	√	√	√	√	√
C	×	√	√	√	√	√	√	√	√	√	√	√
D	×	√	√	√	√	√	√	√	√	√	√	√
E	×	√	√	√	√	√	√	√	√	√	√	√
F	×	√	√	√	√	√	√	√	√	√	√	√
G	×	√	√	√	√	√	√	√	√	√	√	√
H	√	√	√	√	√	√	√	√	√	√	√	√
I	√	√	√	√	√	√	√	√	√	√	√	√
J	√	√	√	√	√	√	√	√	√	√	√	√
K	√	√	√	√	√	√	√	√	√	√	√	√
L	√	√	√	√	√	√	√	√	√	√	√	√
M	×	√	√	√	√	√	√	√	√	√	√	√
N	√	√	√	√	√	√	√	√	√	√	√	√
O	×	×	√	√	√	√	√	√	√	√	√	√
P	√	√	√	√	√	√	√	√	√	√	√	√
Q	×	×	√	√	√	√	√	√	√	√	√	√
R	×	×	√	√	√	√	√	√	√	√	√	√
S	×	√	√	√	√	√	√	√	√	√	√	√
T	√	√	√	√	√	√	√	√	√	√	√	√
U	×	√	√	√	√	√	√	√	√	√	√	√
V	√	√	√	√	√	√	√	√	√	√	√	√
W	×	√	√	√	√	√	√	√	√	√	√	√
X	×	√	√	√	√	√	√	√	√	√	√	√
Y	√	√	√	√	√	√	√	√	√	√	√	√
Z	×	√	√	√	√	√	√	√	√	√	√	√
ถูก (ตัว)	11	23	26	26	26	26	26	26	26	26	26	26

ตารางที่ 4.4 ผลการทดลองที่ 4 ทดสอบโปรแกรม ย้ายตำแหน่งตัวอักษร โดยใช้ Font Microsoft Sans Serif ใช้พื้นสีน้ำเงิน และตัวอักษรสีแดง

ตำแหน่ง / ตัวอักษร	ย้ายไปด้านซ้าย	ย้ายไปด้านขวา	ย้ายไปด้านบน	ย้ายไปด้านล่าง
A	√	√	√	√
B	√	√	√	√
C	√	√	√	√
D	√	√	√	√
E	√	√	√	√
F	√	√	√	√
G	√	√	√	√
H	√	√	√	√
I	√	√	√	√
J	√	√	√	√
K	√	√	√	√
L	√	√	√	√
M	√	√	√	√
N	√	√	√	√
O	√	√	√	√
P	√	√	√	√
Q	√	√	√	√
R	√	√	√	√
S	√	√	√	√
T	√	√	√	√
U	√	√	√	√
V	√	√	√	√
W	√	√	√	√
X	√	√	√	√
Y	√	√	√	√
Z	√	√	√	√
รวมถูก (ตัว)	26	26	26	26

ตารางที่ 4.5 ผลการทดลองที่ 5 ทดสอบโปรแกรม การยืดตัวอักษรออกไป 1.5 เท่าของตัวอักษรที่สอนให้โปรแกรมโดยใช้ Font Microsoft Sans Serif ใช้พื้นสีน้ำเงิน สีเขียว และตัวอักษรสีแดง ตัวอักษรสีเขียว น้ำเงิน และแดง รวมทั้งหมดตัวอักษรละ 12 แบบ

การยืด / ตัวอักษร	ยืดตัวอักษรแนวนอน 1.5 เท่า (ถูก %)	ยืดตัวอักษรแนวตั้ง 1.5 เท่า (ถูก %)
A	100	100
B	100	100
C	100	100
D	100	100
E	83.33	100
F	91.67	100
G	100	100
H	100	100
I	100	100
J	100	100
K	100	100
L	100	100
M	100	100
N	100	100
O	100	100
P	100	100
Q	100	100
R	100	100
S	100	100
T	91.67	100
U	100	100
V	100	100
W	100	100
X	100	100
Y	100	100
Z	100	100

ตารางที่ 4.6 ผลการทดลองที่ 6.1 ทดลองโดย สีพื้น 1 สี และสีตัวอักษรเป็นเฉดสี

การยึด / ตัวอักษร	ความถูกต้อง (%)
A	100
B	100
C	100
D	100
E	100
F	100
G	100
H	100
I	100
J	100
K	100
L	100
M	100
N	100
O	100
P	100
Q	100
R	100
S	100
T	100
U	100
V	100
W	100
X	100
Y	100
Z	100

ตารางที่ 4.7 ผลการทดลองที่ 6.2 ทดลองโดย สีพื้นเป็นเฉดสี และสีตัวอักษร 1 สี

ตัวอักษร \ การยึด	ถูกต้อง (%)
A	0
B	0
C	0
D	0
E	0
F	0
G	0
H	0
I	0
J	0
K	0
L	0
M	0
N	0
O	0
P	0
Q	0
R	0
S	0
T	0
U	0
V	0
W	0
X	0
Y	0

บทที่ 5

สรุปผล

โครงงานนี้ได้ศึกษาทฤษฎีสี RGB และวิธีการจดจำรูปแบบตัวอักษรพร้อมทั้งจัดทำโปรแกรม color OCR จนสามารถทำงานได้จริง แต่ข้อจำกัดของโปรแกรมคือสามารถทดสอบกับภาพที่มีตัวอักษร 1 ตัวอักษรต่อ 1 ภาพเท่านั้น ซึ่งในอนาคตจะพัฒนาโปรแกรมให้สามารถทดสอบกับภาพที่มีหลายตัวอักษรใน 1 ภาพได้

5.1 ผลการทดลอง

โครงงานนี้มีการออกแบบการทดลองแบ่งเป็น 6 กรณี คือ

1. การเปลี่ยนสีตัวอักษร พื้นหลัง และแบบตัวอักษร
2. การย่อขยายตัวอักษร
3. การเปลี่ยนองศาการเอียงของตัวอักษร
4. การเปลี่ยนตำแหน่งตัวอักษร
5. การยืด หด ตัวอักษร
6. การใช้เฉดสีเข้ามาเกี่ยวข้องในตัวอักษร และพื้นหลัง

จากการทดลองการรู้จำอักขระด้วยวิธีต่าง ๆ ข้างต้น ผลการทดลองเป็นดังนี้

1. กรณีการเปลี่ยนสีตัวอักษร พื้นหลัง และการเปลี่ยนแบบตัวอักษร ผลการทดลองแสดงให้เห็นถึงประสิทธิภาพพื้นฐานของโปรแกรมที่สามารถจดจำตัวอักษรได้ถูกต้องแม่นยำ ไม่เกิดข้อผิดพลาด

2. กรณีการย่อขยายตัวอักษร ผลการทำลองเกิดข้อผิดพลาดขึ้นเมื่อทำการย่อขนาดตัวอักษรลงประมาณ 60% ของตัวอักษรปกติ แต่การขยายตัวอักษรไม่เกิดข้อผิดพลาด

3. กรณีการเปลี่ยนองศาการเอียงของตัวอักษร กรณีนี้ได้ออกแบบให้ตัวอักษรมีองศาการเอียงที่ตำแหน่ง 2, 3, 5, 7 องศาตามลำดับ ผลการทดลองสังเกตได้ว่า ความผิดพลาดจะเกิดขึ้นเมื่อองศาการเอียงมากกว่า 5 องศา เนื่องจากโปรแกรมการวิเคราะห์ภาพไม่เหมาะสมกับการเอียงของตัวอักษร

4. กรณีการเปลี่ยนตำแหน่งตัวอักษร การทดลองคือ ย้ายตำแหน่งตัวอักษรไปจากตำแหน่งเดิมแล้วทำการจดจำ ผลการทดลองไม่เกิดข้อผิดพลาด

5. กรณีการยืดหดตัวอักษร กรณีนี้ต่างจากกรณีการย่อขยายตัวอักษร คือ การยืดหดตัวอักษรจะกระทำในแนวตั้ง ไม่มีการเปลี่ยนขนาดในแนวนอน ผลการทดลองเกิดข้อผิดพลาดเมื่อตัวอักษรถูกลดขนาดความสูงลงจากเดิม แต่การเพิ่มความสูงไม่ทำให้เกิดข้อผิดพลาด

6. กรณีการใช้เฉดสีเข้ามาเกี่ยวข้องในตัวอักษร และพื้นหลัง ผลการทดลองในกรณีนี้เกิดข้อผิดพลาดขึ้นเมื่อทดลองเปลี่ยนสีพื้นหลังเป็นเฉดสี ซึ่งเกิดความผิดพลาดทั้งหมด

จากการทดลองทั้งหมดพบว่าความสามารถในการรู้จำตัวอักษรของโปรแกรมเป็นไปตามจุดประสงค์ โปรแกรมที่พัฒนาสามารถทำการรู้จำตัวอักษรจากการออกแบบการทดลองต่าง ๆ ได้ดี มีประสิทธิภาพ

5.2 ปัญหาและแนวทางแก้ไข

วิธีการปรับปรุงภาพให้ได้ตามความต้องการมีหลายวิธี แต่สำหรับโครงงานนี้ใช้วิธีการพื้นฐานในการศึกษาทำให้เกิดข้อผิดพลาดในการทดลองได้

การเก็บข้อมูลของตัวอักษรจะเพิ่มขึ้นตามจำนวนการบันทึกแต่ละครั้ง ทำให้พื้นที่ที่เก็บข้อมูลต้องเพิ่มขึ้นตามไปด้วย ดังนั้นการพัฒนาโปรแกรมจึงต้องเพิ่มในส่วนของวิธีการเก็บข้อมูลรูปแบบใหม่เพื่อลดการใช้พื้นที่ในการเก็บข้อมูลให้ได้มากที่สุด

ข้อมูลของตัวอักษรยังไม่มีรูปแบบที่แน่นอนทำให้ฐานข้อมูลโตขึ้นเรื่อย ๆ ตามจำนวนข้อมูลที่บันทึก การพัฒนาโปรแกรมจึงต้องวิเคราะห์รูปแบบข้อมูลด้วยว่ามีรูปแบบที่แน่นอนหรือไม่เพื่อลดพื้นที่ในการเก็บฐานข้อมูลของตัวอักษรแต่ละตัว

การจดจำตัวอักษรเอียงเกิดข้อผิดพลาดขึ้นได้ง่าย สังเกตได้จากผลการทดลอง เมื่อเปลี่ยนองศาการเอียงประมาณ 5% โปรแกรมไม่สามารถรู้จำอักษรได้ ดังนั้นเมื่อมีการศึกษาเพิ่มเติมจะต้องหาวิธีการที่สามารถตรวจสอบการเอียงของตัวอักษรได้ และเกิดข้อผิดพลาดในการรู้จำน้อยที่สุด

5.3 สรุปผลการทดลอง

จากการทดลองทั้งแบบตัวอักษรปกติ และตัวอักษรที่ทำการปรับค่าต่าง ๆ ได้แสดงถึงประสิทธิภาพของโปรแกรมที่สามารถรู้จำอักขระชนิดสีด้วยแสงได้ดี สังเกตได้จากการทดลองตัวอักษรปกติที่สามารถรู้จำรูปแบบตัวอักษรต่าง ๆ ได้โดยไม่เกิดข้อผิดพลาด สามารถนำไปใช้ให้เกิดประโยชน์ต่อไปได้ แต่หากยังมีข้อจำกัดบางประการ คือ ความเร็วในการประมวลผลเมื่อฐานข้อมูลโตขึ้น จึงต้องมีพัฒนาให้ดีขึ้นต่อไป

5.4 ข้อเสนอแนะ

1. ใช้เทคนิคขั้นสูงในการปรับปรุงภาพให้ตรงตามความต้องการ เพื่อลดเวลาในการประมวลผลภาพ และให้ได้ภาพที่ดีที่สุด
2. ขั้นตอนและวิธีการทำงานสารมารถนำไปพัฒนาต่อด้วยการเขียนโปรแกรมภาษาอื่น ๆ ได้
3. ใช้วิธีการเก็บข้อมูลตัวอักษรที่มีประสิทธิภาพมากขึ้นกว่าเดิม เพื่อลดเวลา และพื้นที่ในการเก็บข้อมูลตัวอักษร
4. ใช้วิธีการตรวจสอบการเอียงของตัวอักษรที่มีประสิทธิภาพมากกว่าเดิม

บรรณานุกรม

[1] Jeff Kent. **Visual C# 2005: Demystified.** California: McGraw-Hill: 2006.

[2] R. C. Gonzalez and R. E. Woods. **Digital Image Processing**. 2nd Ed., 2002.

[3] อาจารย์จิ๊ก, http://e-mag.en.kku.ac.th/index.php?op=display&id=1, 2537

[4] Anonymous, www.prc.ac.th/newart/webboard/colour08.html

ประวัติผู้เขียนโครงงาน

ชื่อ นางสาวสุรยา ชินะกาญจน์

ภูมิลำเนา 160/2 หมู่ 7 ตำบลบ้านใหญ่ อำเภอเมือง

จังหวัดนครนายก

ประวัติการศึกษา

- จบการศึกษาชั้นมัธยมศึกษาโรงเรียนนครนายกวิทยาคม
- ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4 คณะวิศวกรรมศาสตร์ สาขาคอมพิวเตอร์ มหาวิทยาลัยนเรศวร

E-mail: sky_doctor26@hotmail.com

ชื่อ นายเขมภพ บุญพลอย

ภูมิลำเนา 249 หมู่ 3 ตำบลพลายชุมพล อำเภอเมือง

จังหวัดพิษณุโลก

ประวัติการศึกษา

- จบการศึกษาชั้นมัธยมศึกษาจากโรงเรียนพิษณุโลกพิทยาคม
- ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4 คณะวิศวกรรมศาสตร์ สาขาคอมพิวเตอร์ มหาวิทยาลัยนเรศวร

E-mail: boon_ploy@hotmail.com

ชื่อ นายเอกวิทย์ กลิ่นลั่นทม

ภูมิลำเนา 13 หมู่ 8 ตำบลโพนแพน อำเภออากาศอำนวย

จังหวัดสกลนคร

ประวัติการศึกษา

- จบการศึกษาชั้นมัธยมศึกษาจากโรงเรียนเตรียมอุดมศึกษาภาคตะวันออกเฉียงเหนือ
- ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4 คณะวิศวกรรมศาสตร์ สาขาคอมพิวเตอร์ มหาวิทยาลัยนเรศวร

E-mail: aker_cpex@hotmail.com