โปรแกรมค้นหาความคล้ายคลึงของรูปภาพ

IMAGE RETRIEVAL SOFTWARE

นางสาวฉัตรระพี แจ่มอ่วม รหัส 51371239

นายสุรเศรษฐ์ หงษ์ทอง รหัส 51371628

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 9 ก.ย. 2556
เลขทะเบียน 1638039
เลขเรียกหนังสือ ป9
มหาวิทยาลัยนเรศวร จ235 น 2556

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2555

ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อโครงงาน	โปรแกรมค้นหาความคล้ายคลึงของรูปภาพ		
ผู้ดำเนินโครงงาน	นางสาวฉัตรระพี	แจ่มอ่วม	รหัส 51371239
	นายสุรเศรษฐ์	หงษ์ทอง	รหัส 51371628
อาจารย์ที่ปรึกษา	อาจารย์เศรษฐา ตั้งค้าวานิช		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2555		

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์

.....**ที่ปรึกษาโครงงาน**
(อาจารย์เศรษฐา ตั้งค้าวานิช)

.....**กรรมการ**
(ผู้ช่วยศาสตราจารย์ ดร.พนมขวัญ ริยะมงคล)

.....**กรรมการ**
(อาจารย์ภาณุพงศ์ สอนคม)

.....**กรรมการ**
(อาจารย์จิราพร พุกสุข)

ชื่อโครงงาน	โปรแกรมค้นหาความคล้ายคลึงของรูปภาพ		
ผู้ดำเนินโครงงาน	นางสาวฉัตรระพี	แจ่มอ่วม	รหัส 51371239
	นายสุรเศรษฐ์	หงษ์ทอง	รหัส 51371628
ที่ปรึกษาโครงงาน	อาจารย์เศรษฐา ตั้งค้าวานิช		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2555		

บทคัดย่อ

โครงงานนี้เป็นโครงงานที่ศึกษาเกี่ยวกับการพัฒนาโปรแกรมค้นหาภาพในฐานข้อมูล โดยการวิเคราะห์ภาพจะใช้ทฤษฎีกราฟแสดงความถี่ความเข้มสี ในการวิเคราะห์สีของภาพ เพื่อสำหรับเก็บเป็นดัชนี เก็บไว้ในฐานข้อมูล โดยดัชนีในฐานข้อมูลจะมีไว้สำหรับเปรียบเทียบกับดัชชีของภาพตัวอย่าง เพื่อสามารถทำการค้นหาภาพต้นแบบที่มีความคล้ายคลึงในฐานข้อมูล จากการค้นหาเราจะศึกษา การค้นหาด้วยสี ซึ่งโปรแกรมจะถูกพัฒนาบนวินโดว์แอพพลิเคชั่น ของไมโครซอฟต์ วิชวลสตูดิโอ 2010โดยใช้ภาษา C# ในการพัฒนาโปรแกรม

จากผลการทดลอง ได้ทำการสรุปผลที่ได้จากการทดลองเพื่อหาค่าความถูกต้อง โดยการนำภาพต้นแบบไปเปรียบเทียบกับภาพที่อยู่ในฐานข้อมูล ซึ่งขึ้นอยู่กับขนาดฐานข้อมูลที่มีด้วยเช่นกัน

Project title	IMAGE RETRIEVAL SOFTWARE	
Name	Ms.Chatrapee Jam-Oum	ID.51371239
	Mr. Suraseth Hongtong	ID.51371628
Project advisor	Mr.Settha Thangkawanit	
Major	Computer Engineering	
Department	Electrical and Computer Engineering	
Academic year	2012	

Abstract

This project is a project which focuses on the development of a database search programs. By image analysis using Color Histogram.In the analysis of color images the index stored of the database.The index in the database for comparison with the index of example picture,To be able to find images that are similar prototype in the database.We will study one of the search results with color. The program is developed on windows applications of Microsoft Visual Stdio 2010. Using C# for development programs like Image Processing.

Results.Have concluded that it will meet most needs.By applying this model to compare with the images in the database.Depending on the size of the database as well.

กิตติกรรมประกาศ

ในการจัดทำโครงงานนี้สำเร็จได้ด้วยความอนุเคราะห์อย่างดียิ่งของ อาจารย์เศรษฐา ตั้งค้าวานิช อาจารย์ที่ปรึกษาปริญญานิพนธ์ ผู้ช่วยศาสตราจารย์ ดร. พนมขวัญ ริยะมงคล อาจารย์ภาณุพงศ์ สอนคม และอาจารย์จิราพร พุกสุข ที่ให้ความอนุเคราะห์ให้คำปรึกษา ชี้แนะ ตรวจสอบและตรวจทานแก้ไขให้แก่ผู้จัดทำเป็นอย่างดียิ่ง ผู้จัดทำขอกราบขอบพระคุณอาจารย์ทั้งสี่ท่านเป็นอย่างสูงไว้ ณ ที่นี้

ขอกราบขอบพระคุณอาจารย์ ทุกท่านที่ช่วยประสิทธิ์ประสาทวิชาให้แก่ศิษย์ตั้งแต่ระดับอนุบาลจนถึงระดับปริญญาตรีขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และครอบครัว ที่ช่วยเป็นกำลังใจ เป็นแรงผลักดัน เป็นที่ปรึกษา และเป็นทุกอย่างของลูก ลูกขอกราบขอบพระคุณอย่างสุดซึ้งที่ทำปริญญานิพนธ์ฉบับนี้สำเร็จได้ด้วยดีและทำให้ลูกได้มีวันนี้

ขอขอบคุณ อาจารย์สุรเดช จิตประไพกุลศาล คุณสรียา รุ่งจำรัส คุณน้ำอ้อย วันตา ที่ช่วยให้ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี ผู้จัดทำขอขอบคุณอย่างสุดซึ้ง

ขอขอบคุณญาติพี่น้อง เพื่อนๆ ทุกคน ที่คอยไต่ถามสารทุกข์สุกดิบและเป็นกำลังใจให้ผู้จัดทำมาโดยตลอด

คณะผู้จัดทำ
ฉัตรระพี แจ่มอ่วม
สุรเศรษฐ์ หงษ์ทอง

สารบัญ

	หน้า
หน้าอนุมัติ	ก
บทคัดย่อ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	ฉ
สารบัญตาราง	ญ
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญของโครงการ	1
1.2 วัตถุประสงค์ของโครงงาน	1
1.3 ขอบข่ายของโครงงาน	2
1.4 ขั้นตอนการดำเนินงาน	2
1.5 แผนการดำเนินงาน	3
1.6 ผลที่คาดว่าจะได้รับจากโครงงาน	4
1.7 งบประมาณ	4
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	5
2.1 การค้นหาภาพโดยใช้องค์ประกอบพื้นฐาน (Content – Base – Image - Retrieval)	5
2.2 มาตรฐานของสี	5
2.3 ระบบสี	5
2.4 ภาพโทนสีเทา	8
2.5 การกรองข้อมูลภาพ	9
2.6 การแยกส่วนภาพ	20
2.7 ภาพกราฟแจกแจงความถี่ (Image Histogram)	23
2.8 ระบบการค้นคืนภาพด้วยเนื้อหา (Content-Based Image Retrieval: CBIR)	24
2.9 ลักษณะเฉพาะของภาพ (Image Feature)	28

สารบัญ (ต่อ)

หน้า

2.10 ความรู้เบื้องต้นเกี่ยวกับไลบรารีเอฟอาร์ต 28

บทที่ 3 ขั้นตอนการดำเนินงาน 31

3.1 ภาพโดยรวมในการค้นหาภาพของระบบ 31

3.2 การออกแบบหน้าต่างโปรแกรมส่วนติดต่อกับผู้ใช้ 43

บทที่ 4 ผลการทดลอง 50

4.1 ขั้นตอนการทดลองค้นหาภาพโดยวิเคราะห์ด้วยสี 50

4.2 ผลการทดลองค้นหาภาพโดยวิธีการวิเคราะห์ด้วยสี 51

4.3 กรณีที่ระบบแสดงตัวอย่างรูปภาพภายในฐานข้อมูล 61

4.4 สรุปผลการทดลอง 62

บทที่ 5 บทสรุปและข้อเสนอแนะ 62

5.1 สรุปผลการดำเนินโครงการ 62

5.2 ปัญหาในการดำเนินโครงการ 62

5.3 ข้อเสนอแนะ 63

บรรณานุกรม 64

ภาคผนวก ก 66

ภาคผนวก ข 71

ประวัติผู้ดำเนินโครงการ 75

สารบัญภาพ

หน้า

ภาพที่

2.1 แบบสี RGB ในรูปแบบพิกัด 3 มิติ 7

ระบบสี HSV (Hue Saturation Value) ในรูปแบบพิกัด 3 มิติ 7

2.2 แสดงค่าสีใน ภาพโทนสีเทา 8

2.3 แบบสี RGB ในรูปแบบพิกัด 3 มิติ แสดงจุด pixel ที่ต้องการหา 9

2.4 การเปลี่ยนระบบสี RGB เป็น ภาพโทนสีเทา 9

2.5 การกรองข้อมูลภาพ 10

การกรองข้อมูลภาพโดยใช้ค่ามัธยฐาน 13

2.6 การกรองข้อมูลภาพโดยใช้ค่าฐานนิยม 13

2.9 การกรองโดยใช้หน้าต่าง 14

2.7 การกรองข้อมูลภาพโดยใช้หน้าต่าง (ก) ใช้การเฉลี่ยแบบคณิตศาสตร์ (แสดงผลลัพธ์หลังการปัดเศษ) (ข) ใช้มัธยฐาน (ค) ใช้ฐานนิยม (ใช้มัธยฐานแทนสำหรับจุดที่มีปัญหา) (ง) ใช้ k-closest averaging (แสดงผลลัพธ์หลังการปัดเศษ) 15

2.8 ผลของการเติมค่าศูนย์รอบภาพเริ่มต้นก่อนทำการคอนโวลูชัน 16

2.12 ผลของการเติมค่าศูนย์รอบภาพเริ่มต้นก่อนทำการคอนโวลูชัน 18

2.13 การกรองภาพด้วยเทมเพลตกรองความถี่สูง และความถี่ต่ำ 20

2.10 ตัวอย่างผลลัพธ์ประเภทการแยกส่วนภาพ 21

2.11 Detection of Isolated Point 22

2.12 Boundary (Border) Following 23

2.15 ตัวอย่าง Histogram ของภาพ 24

2.16 แผนภาพแสดงระบบการค้นคืนภาพด้วยเนื้อหา (CBIR) 25

2.17 ขั้นตอนการสร้างดัชนีภาพ 26

2.18 ขั้นตอนการค้นคืนภาพ 27

3.1 ขั้นตอนการพัฒนาซอฟต์แวร์ 31

3.2 ส่วนประกอบต่างๆภายในโปรแกรมค้นหาความคล้ายคลึงของรูปภาพ 32

3.3 ระบบแสดงตัวอย่างรูปภาพภายในแฟ้มข้อมูล 32

3.4 กระบวนการการปรับปรุงภาพ 33

สารบัญภาพ (ต่อ)

หน้า

ภาพที่

3.5 กระบวนการการปรับปรุงภาพ 34

3.6 กระบวนการการปรับปรุงภาพ 35

3.7 กระบวนการการปรับปรุงภาพในโดเมนความถี่ (frequency domain) 35

3.8 กระบวนการปรับปรุงภาพด้วยตัวกรองเกาส์เซียน 36

3.9 ปรับภาพกลับตำแหน่งเดิม 36

3.10 ปรับภาพให้เป็นภาพระดับสี RGB 37

3.11 ปรับภาพให้เป็นภาพระดับสี HSV 37

3.12 ตัวอย่างภาพ ที่ใช้ในงานปริญญานิพนธ์ 38

3.13 การแสดงค่าฮิสโตแกรมเป็นกราฟ 39

3.14 การแสดงค่าพิกเซล 41

3.15 ระยะห่าง (Distance) 41

3.16 ตัวอย่างการแสดงผลของโปรแกรม 42

3.17 หน้าการRunโปรแกรมค้นหาความคล้ายคลึงของรูปภาพ 44

3.18 หน้าจอแอพพลิเคชันค้นหาความความคล้ายคลึงของรูปภาพ 44

3.19 แสดงหน้าการค้นหารูปภาพปุ่ม Image Search 45

3.20 แสดงหน้าต่าง Image Search 45

3.21 แสดงหน้าการเลือกแฟ้มข้อมูล ที่ใช้เก็บข้อมูลรูปภาพ 46

3.22 แสดงการค้นหารูปภาพ 47

3.23 หน้าต่างการแสดงผลการค้นหารูปภาพที่ต้องการ 47

3.24 หน้าต่างการแสดงผลการตัวอย่างรูปภาพในแฟ้มข้อมูล 48

สารบัญภาพ (ต่อ)

หน้า

ภาพที่

4.1 ตัวอย่างเสื้อสีแดงที่มีอยู่ในแฟ้มข้อมูล 51
4.2 ผลการค้นหารูปภาพเสื้อสีแดงพื้นและมีลาย โดยการวิเคราะห์จากสี 52
4.3 การแสดงตัวอย่างรูปภาพภายในแฟ้มข้อมูล 61

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ในปัจจุบันนี้ เทคโนโลยีทางด้านคอมพิวเตอร์ได้เข้ามามีบทบาทและมีความสำคัญเป็นอย่างมากและได้เข้ามามีบทบาทสำคัญในชีวิตประจำวันของมนุษย์ ไม่ว่าจะเป็นในระดับองค์กรขนาดใหญ่จนกระทั่งถึงองค์กรขนาดเล็ก ก็ได้มีการนำเอาคอมพิวเตอร์เข้ามาใช้เพื่ออำนวยความสะดวกและลดปัญหาต่าง ๆ เพื่อช่วยในการจัดเก็บข้อมูล สืบค้นข้อมูล และพัฒนาระบบงานต่าง ๆ หนึ่งในนั้นคือ รูปภาพ การค้นหารูปภาพในปัจจุบันนั้นเป็นการยากที่จะหารูปภาพให้ได้ตรงตามความต้องการของผู้ค้นหา ได้สะดวกและรวดเร็วมากขึ้นได้

ในการพัฒนาโปรแกรม ในการค้นหารูปภาพโดยใช้องค์ประกอบพื้นฐาน(CBIR) คือการนำภาพมาวิเคราะห์ให้ได้ค่าดัชนี(Index) ของภาพซึ่งเป็นคุณสมบัติพื้นฐานของภาพนั้นๆ โดยนำค่าพื้นฐานต่างๆของภาพ ดังนั้นจึงมีการสร้างเครื่องมือค้นหารูปภาพขึ้น ซึ่งปัจจุบันโปรแกรมค้นหา (Search Engine) ที่มีอยู่นั้นจะทำการหารูปภาพโดยค้นหารูปภาพตามที่ผู้ค้นหาป้อนความต้องการให้กับโปรแกรม แต่ก็มีรูปภาพมากมายที่ไม่ใช่ภาพที่ต้องการและไม่ตรงกับคุณสมบัติที่ป้อนให้ไป ดังนั้นจึงคิดว่าน่าจะมีการค้นหารูปภาพด้วยวิธีอื่นที่สะดวกแม่นยำ และรวดเร็วกว่าเดิมโดยน่าจะมีการนำภาพตัวอย่างที่ผู้ต้องการค้นหานั้นมาใช้ค้นหารูปภาพต่างๆ ตอบสนองต่อผู้ใช้งานมากขึ้น ฉะนั้นจึงมีการค้นหารูปภาพโดยใช้องค์ประกอบพื้นฐานของภาพ(Content-based Image Retrieval (CBIR)) ขึ้นได้

จากปัญหาตามที่กล่าวมาข้างต้น จึงเป็นที่มาแนวคิดที่จะพัฒนาโปรแกรมสืบค้นภาพ เพื่อใช้ในการจัดเก็บข้อมูลรูปภาพและสามารถทำการสืบค้นข้อมูลด้วยรูปภาพได้รวดเร็วและตรงตามความต้องการของผู้ใช้ได้มีประสิทธิภาพมากขึ้น

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อให้ผู้ใช้สามารถค้นหาสินค้าจากรูปภาพที่ต้องการ และนำมาเปรียบเทียบในคลังสินค้าของเรา และแสดงสินค้าที่มีความใกล้เคียงให้แก่ผู้ใช้ และตรงความต้องการมากที่สุด

1.2.2 ศึกษากระบวนการป้อนกลับ (Relevance Feedback) และการค้นหาโดยใช้องค์ประกอบ

ภาพ (Content-based Image Retrieval (CBIR))

1.2.3 ศึกษาการทำฐานข้อมูลรูปภาพ ในการจัดการฐานข้อมูลภาพและดัชนีของภาพที่ได้จากการวิเคราะห์รูปภาพได้

1.2.4 เพื่อพัฒนาโปรแกรมสืบค้นภาพโดยใช้โปรแกรม Microsoft Visual Studio 2010 ได้

1.3 ขอบข่ายของโครงการ

ปริญญานิพนธ์ฉบับนี้ได้นำเสนอแนวทางการค้นหาสินค้าในหมวดแฟชั่นดังนี้ เสื้อ กระโปรง กางเกง เครื่องประดับ โดยใช้โปรแกรมในการดาวน์โหลดข้อมูลภาพสินค้า

โดยมีศูนย์การค้าจำลองเพื่อใช้ในการค้นหาสินค้า โดย การประยุกต์ แสดงรายละเอียดของสินค้าและรูปภาพสินค้า ที่ผู้ใช้ค้นหาโดยใช้รูปภาพในการค้นหาแล้วนำไปเปรียบเทียบในกลุ่มสินค้าเพื่อให้สินค้าที่ทำการค้นหาตรงตามความต้องการของลูกค้ามากที่สุด

1.4. ขั้นตอนการดำเนินงาน

1.4.1 ศึกษาการใช้ visual studio 2010 ในการพัฒนาโปรแกรม

1.4.2 ศึกษาการใช้ AForge.NET ในการพัฒนาโปรแกรม

1.4.4 ศึกษาหลักการฮีสโทแกรมสี และการเข้ารหัสบีบอัดภาพ

1.4.5 ทดสอบการทำงานของ โปรแกรมสืบค้นภาพ ในการค้นหารูปภาพที่ต้องการและแก้ไขปัญหาที่เกิดขึ้นกับโปรแกรมที่ทำการพัฒนาได้

1.4.6 สรุปการดำเนินงานและเขียนปริญญานิพนธ์

1.5 แผนการดำเนินงาน

รายละเอียด	ระยะเวลาการดำเนินการ									
	2555							2556		
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1.ศึกษาการใช้ visual studio 2010 ในการพัฒนาโปรแกรม										
2.ศึกษาการใช้ AForge.NET ในการพัฒนาโปรแกรม										
3.ศึกษาการออกแบบการจัดเก็บข้อมูลรูปภาพในฐานข้อมูล										
4.ศึกษาหลักการฮีสโทแกรมสี และการเข้ารหัสบีบอัดภาพ										
5. เขียนโปรแกรมทดสอบการค้นหารูปภาพ										
6.ทดสอบการทำงานของโปรแกรมสืบค้นภาพ										
7.แก้ไขข้อผิดพลาดของโปรแกรม										
8.สรุปการดำเนินงาน										

1.6 ผลที่คาดว่าจะได้รับจากโครงงาน

1.6.1 ได้โปรแกรมค้นหา เพื่อใช้ในการค้นหาไฟล์รูปภาพได้อย่างรวดเร็วและมีประสิทธิภาพ

1.6.2 ได้ฐานข้อมูลรูปภาพที่ใช้เก็บไฟล์รูปภาพ และสามารถแก้ไขปรับปรุงฐานข้อมูลให้มีประสิทธิภาพและนำมาใช้ร่วมกับโปรแกรมในการค้นหาที่สร้างขึ้นได้

1.7 งบประมาณ

1.7.1 ค่าใช่จ่ายในการซื้ออุปกรณ์สำนักงาน	1,000	บาท
1.7.2 ค่าจัดทำปริญญานิพนธ์	2,500	บาท
รวมเป็นเงินทั้งสิ้น	3,500	บาท

หมายเหตุ เฉลี่ยค่าใช้จ่ายทุกรายการ

มหาวิทยาลัยนเรศวร

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

เนื้อหาในบทนี้กล่าวถึงความรู้พื้นฐานและทฤษฎีที่เกี่ยวข้องกับวิทยานิพนธ์ ได้แก่ ทฤษฎีการประมวลผลภาพดิจิตอล การเข้ารหัสและการบีบอัดภาพ ทฤษฎีฟัซซีลอจิก ระบบการค้นคืนภาพด้วยเนื้อหาของภาพ ลักษณะเฉพาะของภาพ ซึ่งประกอบด้วย สี รูปร่าง และพื้นผิว การเปรียบเทียบเวกเตอร์ลักษณะเฉพาะ และการวัดประสิทธิภาพของวิธีการค้นคืนภาพ

2.1 การค้นหาภาพโดยใช้องค์ประกอบพื้นฐาน (Content – Base – Image - Retrieval)

ในที่นี้จะนำรูปภาพคุณสมบัติพื้นฐาน จากนั้นจะใช้วิธีวิเคราะห์รูปภาพ คือ สี เมื่อได้ดัชนีแล้ว นำดัชนีที่ได้ไปเปรียบเทียบกับภาพในฐานข้อมูล แล้วคำนวณหาค่าความแตกต่างเพื่อนำค่าที่ได้ไปหาภาพที่ต้องการในฐานข้อมูล ซึ่งในส่วนการวิเคราะห์นั้น ได้ทำการวิเคราะห์ คือ สี ของภาพ โดยมีทฤษฎีที่เกี่ยวข้อง ดังนี้

2.1.1 สี วิเคราะห์ด้วยทฤษฎีกราฟแสดงความถี่ความเข้มสี (Color Histogram)

2.1.2 การค้นหาภาพย้อนกลับ (Relevance Feedback)

2.2 มาตรฐานของสี

ในปี 1964 ณ ห้องวิจัย Jet Propulsion (Pasadena California) ได้ทำการคิดค้นมาตรฐานของสี เพื่อนำมาใช้ในกระบวนการประมวลผลภาพ นับว่าเป็นความรู้พื้นฐานที่สำคัญ เพื่อนำมาใช้วิเคราะห์ข้อมูล และจำแนกรายละเอียดภายของรูปภาพ โดยสีที่ใช้อยู่ในปัจจุบันมีอยู่หลายระบบด้วยกัน ทั้งนี้จะขึ้นอยู่กับการนำไปใช้ แต่โดยทั่วไปแล้วทุกมาตรฐานจะมีแนวคิดเดียวกันคือ การแทนจุดสีด้วยจุดที่อยู่ภายปริภูมิสีมาตรฐาน (Standard color space) 3 มิติ โดยจะมีแกนอ้างอิงสำหรับจุดสีนั้น ในปริภูมิสีแต่ละแกนจะมีความเป็นอิสระต่อกัน

2.3 ระบบสี

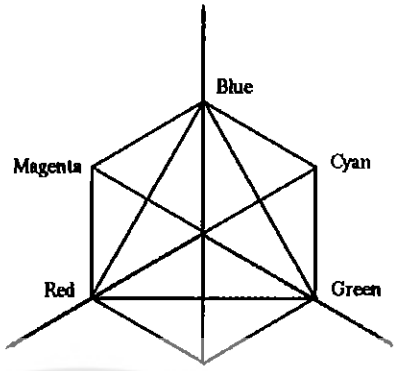
มาตรฐานของสีที่ใช้อยู่ในปัจจุบันมีอยู่หลายระบบด้วยกัน ทั้งนี้จะขึ้นอยู่กับการนำไปใช้ แต่โดยทั่วไปแล้วทุกมาตรฐานจะมีแนวคิดเดียวกัน คือ การแทนจุดสีด้วยจุดที่อยู่ภายในสเปส 3 มิติ โดยจะมีแกนอ้างอิงสำหรับจุดสีนั้นในสเปส ซึ่งแต่ละแกนจะมีความเป็นอิสระต่อกัน ตัวอย่างเช่น ในระบบ RGB จะมีแกนสีคือ แกนสีแดง สีเขียว และสีน้ำเงิน ในระบบ HLS จะมีแกนเป็นค่าสี

(hue) ความสว่าง (lightness) และความบริสุทธิ์ของสี (saturation) ตัวอย่างระบบสีที่นิยมใช้กันได้แก่ ระบบ RGB และ HSV (Hue Saturation Value)

2.3.1 ระบบสี RGB

ระบบสี RGB เป็นระบบสีของแสง ซึ่งเกิดจากการหักเหของแสงผ่านแท่งแก้วปริซึมจะเกิดแถบสีที่เรียกว่า สีรุ้ง ซึ่งแยกสีตามที่สายตามองเห็นได้ 7 สี คือ สีแดง สีแสด สีเหลือง สีเขียว สีน้ำเงิน สีคราม สีม่วง ซึ่งเป็นพลังงานอยู่ในรูปของรังสีที่มีช่วงคลื่น ที่สายตามนุษย์สามารถมองเห็นได้ แสงสีม่วงมีความถี่คลื่นสูงที่สุด คลื่นแสงที่มีความถี่สูงกว่าแสงสีม่วงเรียกว่าอัลตราไวโอเลต และคลื่นแสงสีแดงมีความถี่คลื่นต่ำที่สุด คลื่นแสงที่ต่ำกว่าแสงสีแดงเรียกว่าอินฟราเรด คลื่นแสงที่มีความถี่สูงกว่าสีม่วงและต่ำกว่าสีแดงนั้น สายตาของมนุษย์ไม่สามารถรับได้ และเมื่อศึกษาดูแล้วแสงสีทั้งหมดเกิดจาก แสงสี 3 สี คือ สีแดง สีน้ำเงิน และสีเขียว ทั้งสามสีถือเป็นแม่สีของแสง เมื่อนำมาฉายรวมกันจะทำให้เกิดสีใหม่อีก 3 สี คือ สีมาเจนต้า สีฟ้าไซแอน และสีเหลือง และถ้าฉายแสงสีทั้งหมดรวมกันจะได้แสงสีขาว จากคุณสมบัติของแสง ได้นำมาใช้ประโยชน์ทั่วไปในการฉายภาพยนตร์ การบันทึกภาพวิดีโอ ภาพโทรทัศน์ การสร้างภาพเพื่อการนำเสนอทางจอคอมพิวเตอร์และการจัดแสงสีในการแสดง เป็นต้น (http://research.rdi.ku.ac.th/world/cache/03/SuwatchaiCHUAll.pdf)

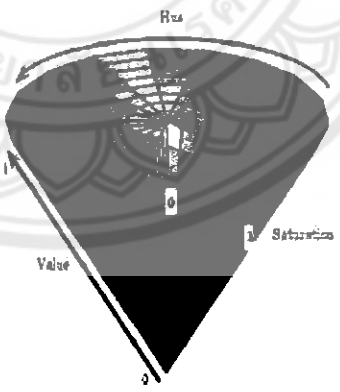
ระบบ RGB เป็นระบบที่เกิดจากการผสมแสงสีหลัก 3 สี คือ สีแดง สีเขียว สีน้ำเงิน โดยจะแยกกันอยู่คนละแชนแนล สำหรับภาพขนาด 24 บิตต่อพิกเซล ในแต่ละแชนแนลจะมีค่าความเข้มแสง 255 ระดับ คือ ตั้งแต่ 0 ถึง 255 หรือ เท่ากับ 8 บิต (2^8) ซึ่งค่าสีในแต่ละพิกเซลจะเกิดจากการรวมกันของข้อมูลทั้ง 3 แชนแนล โดยระบบคาร์ทีเซียนโคออร์ดิเนตในการระบุค่าสีแต่ละสี ซึ่งสามารถแสดงได้เป็นรูปลูกบาศก์ โดยมีค่า RGB เป็นค่ามุมทั้ง 3 และมีสีดำเป็นโคออร์ดิเนตที่ (0, 0, 0) และมีสีขาวที่โคออร์ดิเนต (255,255,255) ดังแสดงในรูป ส่วนค่าสีต่างๆ จะเป็นจุดภายในลูกบาศก์ ซึ่งกำหนดโดยเวคเตอร์จากจุดกำเนิดชี้ไปยังจุดนั้นๆ



รูปที่ 2.1 แบบสี RGB ในรูปแบบพิกัด 3 มิติ

2.3.2 ระบบสี HSV

ระบบสี HSV (Hue Saturation Value) เป็นการพิจารณาสีโดยใช้ Hue Saturation และ Value ซึ่ง Hue คือ ค่าสีของสีหลัก (สีแดง สีเขียว และสีน้ำเงิน) ในทางปฏิบัติจะอยู่ระหว่าง 0 และ 255 ซึ่งถ้า Hue มีค่าเท่ากับ 0 จะแทนสีแดง และเมื่อ Hue มีค่าเพิ่มขึ้นเรื่อยๆสีก็จะเปลี่ยนแปลงไปตามสเปกตรัมของสีจนถึง 256 จึงจะกลับมาเป็นสีแดงอีกครั้ง ซึ่งสามารถแทนให้อยู่ในรูปขององศาได้ดังนี้ คือ สีแดงเท่ากับ 0 องศา สีเขียวเท่ากับ 120 องศา สีน้ำเงินเท่ากับ 240 องศา



รูปที่ 2.2 ระบบสี HSV (Hue Saturation Value) ในรูปแบบพิกัด 3 มิติ

ที่มา: http://exceed.cpe.ku.ac.th/wiki/index.php/Color_Classify

2.3.3 ระบบสี YUV

ระบบสีแบบ YUV ใช้สำหรับโทรทัศน์แบบ PAL และ SECAM ซึ่งยังมีใช้อยู่ในหลายๆประเทศ โดย Y คือ ค่าความสว่างของภาพ ส่วนสัญญาณ U และ V เป็นสัญญาณที่เก็บค่าสี

ของภาพ ต่อมาได้มีระบบ YIQ มาใช้แทน เนื่องจากพบว่าสัญญาณ I และ Q สามารถลด Bandwidth ได้มากกว่าสัญญาณ U และ V ในขณะที่ได้ภาพที่มีคุณภาพเท่ากัน

2.3.4 ระบบสี YIQ

เป็นระบบที่ใช้ใน TV Broadcasting สำหรับ NTSC ประโยชน์หลักก็เพื่อให้ใช้งานได้กับโทรทัศน์แบบขาว-ดำ โดยที่ y คือ ความสว่างของภาพ ส่วน I และ Q จะเป็นสัญญาณที่เข้ารหัสสีของภาพไว้ ดังนั้นสำหรับโทรทัศน์ขาว-ดำนั้น สามารถใช้ค่า Y ค่าเดียวก็สามารถได้ภาพที่สมบูรณ์

2.4 ภาพโทนสีเทา (Gray Level Image)

เป็นอัตราส่วนของภาพโทนสีเทา ซึ่งมีการไล่ระดับความอ่อน แก่ ที่อยู่ระหว่างสีขาวกับสีดำ การสร้างภาพให้มีระดับสีต่างๆอย่างต่อเนื่อง ด้วยการใช้จุดสีที่มีขนาดต่างกันหรือมีความหนาแน่นของจุดต่างกัน

ค่าในแต่ละพิกเซลของภาพโทนสีเทา คือ ค่าความเข้มของแสง ณ แต่ละตำแหน่งของ พิกเซล ซึ่งจะอยู่ในรูปของภาพโทนสีเทา ดังรูปที่ 2.3 ค่าที่เป็นไปได้ของภาพโทนสีเทา จะขึ้นอยู่กับจำนวนบิตที่ใช้ ตัวอย่างเช่น 8-bit monochrome จะมีภาพโทนสีเทา ทั้งหมด 256 ระดับ ดังรูป

รูปที่ 2.3 แสดงค่าสีในภาพโทนสีเทา

ที่มา: http://research.rdi.ku.ac.th/world/cache/03/SuwatchaiCHUAll.pdf

สามารถเปลี่ยนภาพจากระบบสี RGB เป็น ภาพโทนสีเทา โดยใช้สมการดังต่อไปนี้ในการเปลี่ยน

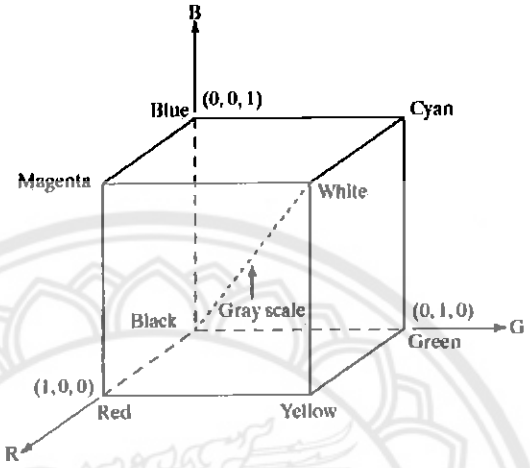
$$y = 0.3R + 0.59G + 0.11B$$

โดย y แทน ค่า ภาพโทนสีเทา ณ จุดพิกเซล ที่ต้องการหา

R แทน ค่าของสีแดงของจุดที่ต้องการหา

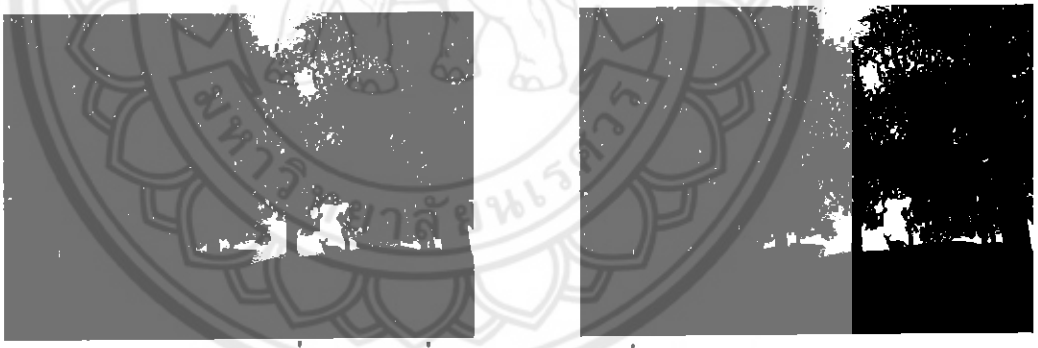
G แทน ค่าของสีเขียวของจุดที่ต้องการหา

B แทน ค่าของสีน้ำเงินของจุดที่ต้องการหา



รูปที่ 2.4 แบบสี RGB ในรูปแบบพิกัด 3 มิติ แสดงจุด พิกเซล ที่ต้องการหา

ตัวอย่าง เมื่อนำภาพมาเปลี่ยนจากระบบสี RGB เป็น ภาพโทนสีเทา



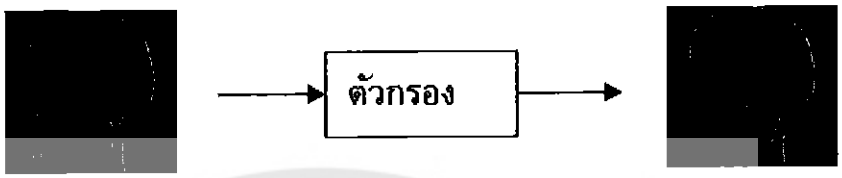
รูปที่ 2.5 การเปลี่ยนระบบสี RGB เป็น Gray Scale

2.5 การกรองข้อมูลภาพ

การกรองข้อมูลภาพ (Image Filtering) คือ การนำภาพไปผ่านตัวกรองสัญญาณ เพื่อให้ได้ภาพผลลัพธ์ออกมา ภาพผลลัพธ์ที่ได้จะมีคุณสมบัติแตกต่างจากภาพเริ่มต้น วัตถุประสงค์หลักของการกรองข้อมูลภาพ คือ การเน้น หรือลดทอน คุณสมบัติบางประการของภาพ เพื่อให้ได้ภาพที่มีคุณสมบัติตามต้องการ

การกรองข้อมูลภาพ คือ การประมวลผลภาพอย่างหนึ่งที่จำเป็นมาก เนื่องจากในการใช้งานจริง ภาพที่ได้มามักมีสัญญาณรบกวนหรือสัญญาณไม่พึงประสงค์อื่นๆปะปนอยู่ด้วย การกรอง

ข้อมูลภาพสามารถปรับปรุงให้ภาพมีคุณสมบัติที่ดีขึ้น เหมาะแก่การประมวลผลในขั้นต่อไป การกรองข้อมูลภาพสามารถแสดงได้ดังรูปที่ 2.6



รูปที่ 2.6 การกรองข้อมูลภาพ

ที่มา: https://docs.google.com/viewer?a=v&q=cache:MB7z6GxWqz4J:

fivedots.coe.psu.ac.th/~montri/Teaching/image/chap5.doc

องค์ประกอบสำคัญของการกรองข้อมูลภาพ คือ ตัวกรอง หากเปรียบภาพเป็นสัญญาณไฟฟ้าที่มีความถี่ต่างๆผสมกันอยู่ ตัวกรองก็คือวงจรไฟฟ้าที่ทำหน้าที่เลือก หรือกรองให้สัญญาณไฟฟ้าที่มีความถี่ในช่วงที่ต้องการผ่านออกไปได้ คุณสมบัติของตัวกรอง คือ ตัวกำหนดคุณสมบัติของภาพผลลัพธ์

อาจมองข้อมูลของภาพๆหนึ่งให้เป็นสัญญาณๆหนึ่งได้ ด้วยการกำหนดให้ระดับความเข้มแสงของแต่ละจุด คือ ขนาด ของสัญญาณ ณ ตำแหน่งนั้นๆ ข้อแตกต่างระหว่างสัญญาณไฟฟ้ากับภาพ คือ ขนาดของสัญญาณไฟฟ้าคือค่าแรงดันหรือกระแส แต่ขนาดของข้อมูลภาพ คือ ระดับความเข้มแสงของจุดภาพ

การเปลี่ยนแปลงของสัญญาณไฟฟ้าเป็นการเปลี่ยนแปลงเทียบกับเวลา ความถี่ของสัญญาณไฟฟ้าถูกกำหนดโดยอัตราการเปลี่ยนแปลงของขนาดของสัญญาณในหนึ่งช่วงเวลา แต่การเปลี่ยนแปลงของข้อมูลภาพเป็นการเปลี่ยนแปลงเทียบกับตำแหน่งของจุดภาพ ความถี่ของการเปลี่ยนแปลงขึ้นอยู่กับอัตราการเปลี่ยนระดับความเข้มแสงของจุดที่อยู่ถัดกันไป สัญญาณไฟฟ้าเป็นสัญญาณมิติเดียว แต่ภาพเป็นสัญญาณ 2 มิติ

ตัวกรอง คือ ระบบๆหนึ่งซึ่งรับสัญญาณเข้า ประมวลผลสัญญาณ และส่งสัญญาณออก โดยทั่วไปตัวกรองจะถูกสร้างให้เป็นระบบเชิงเส้น เนื่องจากออกแบบได้ง่ายและมีประสิทธิภาพดี ปัจจุบันมีทฤษฎีและเทคนิคมากมายเกี่ยวกับการออกแบบตัวกรองสัญญาณแบบเชิงเส้น

ในการกรองข้อมูลภาพ มักพิจารณาว่าภาพ คือ สัญญาณ 2 มิติที่ประกอบขึ้นจากสัญญาณความถี่ต่างๆผสมกันอยู่ในสัดส่วนที่ต่างกัน การออกแบบตัวกรองจึงเป็นการกำหนดว่าต้องการกำจัดสัญญาณความถี่ใดออกไป (หรือต้องการเลือกสัญญาณความถี่ใดบ้าง) ความรู้พื้นฐานเกี่ยวกับการกรองสัญญาณไฟฟ้า ก็จะสามารถทำความเข้าใจเกี่ยวกับการกรองข้อมูลภาพได้ไม่ยาก เพราะการกรองข้อมูลภาพ คือ ส่วนขยายของความรู้เดิมให้รองรับการประมวลผลสัญญาณ 2 มิติ (https://docs.google.com/viewer?a=v&q=cache:MB7z6GxWqz4J:fivedots.coe.psu.ac.th/~montri/Teaching/image/chap5.doc)

2.5.1 ประเภทของตัวกรอง

ตัวกรองแบ่งออกได้เป็น 4 ประเภทตามลักษณะการเลือกความถี่คือ

2.5.1.1 ตัวกรองความถี่ต่ำผ่าน (Low-pass Filter)

2.5.1.2 ตัวกรองความถี่สูงผ่าน (High-pass Filter)

2.5.1.3 ตัวกรองแถบความถี่ผ่าน (Band-pass Filter)

2.5.1.4 ตัวกรองหยุดแถบความถี่ (Band-stop Filter)

ค่าพารามิเตอร์หลักในการกำหนดคุณสมบัติของตัวกรองคือ ค่าความถี่คัตออฟ (cut-off frequency) ความถี่คัตออฟคือความถี่ที่ระบุจุดตัดของสัญญาณว่าจะให้ผ่าน หรือไม่ผ่าน

ตัวอย่างเช่น ตัวกรองความถี่ต่ำผ่านที่มีค่าความถี่คัตออฟเท่ากับ 1,000 เฮิรตซ์จะยอมให้สัญญาณที่มีความถี่ต่ำกว่า 1,000 เฮิรตซ์ผ่านไปได้ แต่จะไม่ยอมให้สัญญาณที่มีความถี่สูงกว่า 1,000 เฮิรตซ์ผ่าน สำหรับตัวกรองความถี่สูงผ่านจะทำงานตรงข้ามกับตัวกรองความถี่ต่ำผ่าน คือ ไม่ยอมให้สัญญาณที่มีความถี่ต่ำกว่าความถี่คัตออฟผ่านไปได้ แต่จะยอมให้ความถี่ที่สูงกว่าความถี่คัตออฟผ่านได้ วงจรกรองแถบความถี่ผ่านยอมให้สัญญาณในช่วงความถี่หนึ่งผ่านไปได้ หากสัญญาณมีความถี่อยู่นอกช่วงจะถูกลดทอนหรือไม่ยอมให้ผ่านไป สำหรับวงจรหยุดแถบความถี่จะมีลักษณะการทำงานที่ตรงข้ามกัน คือ จะลดทอนสัญญาณที่มีความถี่ในช่วงที่กำหนดลง และจะผ่านความถี่ที่อยู่นอกช่วง ในการกรองสัญญาณใดๆต้องทราบความถี่ หรือช่วงความถี่ของ

สัญญาณที่ต้องการและสัญญาณที่ไม่ต้องการ จากนั้นจะเลือกตัวกรองที่เหมาะสมมาใช้ เพื่อกำจัดสัญญาณที่ไม่ต้องการออก หรือเน้นสัญญาณที่ต้องการให้เด่นชัดยิ่งขึ้น ตัวอย่างเช่น สัญญาณรบกวน

2.5.2 การกรองโดยการเฉลี่ยจากหลายภาพ

หากมีชุดของภาพคุณภาพต่ำหลายๆภาพ ซึ่งถ่ายจากมุมกล้องเดียวกัน สามารถสร้างภาพใหม่ที่มีคุณภาพสูงกว่าจากชุดภาพนั้นได้ หากสัญญาณรบกวนเกิดขึ้นแบบสุ่ม ภาพที่เก็บแต่ละครั้งย่อมมีลักษณะแตกต่างกัน หากความเข้มแสงของจุดในภาพหนึ่งถูกรบกวน สามารถนำข้อมูลความเข้มแสงของจุดจากภาพอื่น ณ ตำแหน่งเดียวกันมาแทนแต่ละจุดในภาพ ผลลัพธ์ที่ได้จะเกิดจากการเฉลี่ย (หรือเลือก) จากจุดที่ตรงกันของภาพต่างๆในชุดภาพ

2.5.3 การกรองข้อมูลภาพโดยใช้ค่าเฉลี่ยแบบคณิตศาสตร์ (mean filtering)

วิธีการนี้จะใช้ค่าเฉลี่ยแบบคณิตศาสตร์ของจุดทั้งหมด หากมีภาพขนาด N x M ทั้งหมด K ภาพ สามารถคำนวณหาภาพใหม่ได้ดังนี้

$$\hat{I}(x,y) = \frac{1}{K}\sum_{j=1}^{K} I_j(x,y)$$

$\hat{I}(x,y)$ คือ ความเข้มแสงของจุด ณ ตำแหน่ง (x, y) ในภาพผลลัพธ์

$\hat{I}(x,y)$ คือ ความเข้มแสงของจุด ณ ตำแหน่ง (x, y) ในภาพที่ j

วิธีนี้เป็นการลดทอนสัญญาณรบกวน ภาพที่ได้จะมีสัญญาณรบกวนลดลง

2.5.4 การกรองข้อมูลภาพโดยใช้ค่ามัธยฐาน (median filtering)

วิธีการนี้จะนำเอาความเข้มแสงของจุดที่ตรงกันในภาพต่างๆ มาเรียงลำดับ จากน้อยไปหามาก จากนั้นจะเลือกค่าที่อยู่ตรงกลางไปใช้ หากจำนวนภาพทั้งหมดเป็นจำนวนคู่ ค่าทั้งสองที่อยู่ตรงกลางจะนำมาหาค่าเฉลี่ย วิธีการนี้จะต้องใช้การเรียงลำดับ ซึ่งเป็นกระบวนการที่ใช้เวลาในการคำนวณสูง แต่ข้อดีคือไม่สูญเสียความคมชัด การกรองข้อมูลภาพโดยใช้ค่ามัธยฐานสามารถแสดงได้ดังรูปที่ 2.7

ภาพที่ 1					ภาพที่ 2					ภาพที่ 3					ผลลัพธ์			
1	2	1	3		2	3	4	3		3	2	1	4		2	2	1	3
4	2	2	1		5	3	4	1		2	1	4	0		4	2	4	1
0	1	1	3	,	3	2	4	2	,	1	4	2	0	=	1	2	2	2
2	2	1	1		1	3	1	2		2	4	0	2		2	3	1	2

รูปที่ 2.7 การกรองข้อมูลภาพโดยใช้ค่ามัธยฐาน

2.5.5 การกรองข้อมูลภาพโดยใช้ค่าฐานนิยม (modal filtering)

วิธีการนี้คล้ายกับวิธีใช้ค่ามัธยฐาน แต่ไม่ใช้การเรียงลำดับข้อมูล ระดับความเข้มแสงที่ใช้บ่อยที่สุดจะถูกเลือกไปใช้ วิธีนี้เสมือนการโหวตลงคะแนนเสียง ผู้ที่ได้คะแน**นเสียงสูงที่สุดคือ**ผู้ชนะ วิธีนี้เหมาะสำหรับการลดสัญญาณรบกวนที่เกิดขึ้นไม่บ่อย (https://docs.google.com/viewer?a=v&q=cache:MB7z6GxWqz4J:fivedots.coe.psu.ac.th/~montri/Teaching/image/chap5.doc) การกรองข้อมูลภาพโดยใช้ค่าฐานนิยมสามารถแสดงได้ดังรูปที่ 2.8

ภาพที่ 1					ภาพที่ 2					ภาพที่ 3					ผลลัพธ์			
1	2	1	3		2	3	4	3		3	2	1	4		2	2	1	3
4	2	2	1		5	3	4	1		2	1	4	0		4	2	4	1
0	1	1	3	,	3	2	4	2	,	1	4	2	0	=	1	2	2	2
2	2	1	1		1	3	1	2		2	4	0	2		2	3	1	2

รูปที่ 2.8 การกรองข้อมูลภาพโดยใช้ค่าฐานนิยม

2.5.6 การกรองโดยใช้หน้าต่าง

การกรองข้อมูลภาพวิธีนี้ จะใช้หน้าต่างในการกำหนดขอบเขตของการพิจารณา เพื่อหาระดับความเข้มแสงของจุดต่างๆ ในภาพผลลัพธ์ความเข้มแสงของจุดที่อยู่รอบๆจุดกึ่งกลางของหน้าต่าง จะถูกนำมาหาค่าเฉลี่ย ค่าเฉลี่ยที่ได้ คือ ค่าความเข้มแสงของจุดในภาพผลลัพธ์ หน้าต่างจะถูกเลื่อนไปยังตำแหน่งต่างๆในภาพจนครบทุกจุด การกรองโดยใช้หน้าต่างสามารถแสดงได้ดังรูปที่ 2.9

รูปที่ 2.9 การกรองโดยใช้หน้าต่าง

จากรูปที่ 2.9 จะเห็นว่ามีหน้าต่างขนาด 3x3 ครอบอยู่ที่มุมบนด้านซ้ายของภาพเริ่มต้น ความเข้มแสง ณ จุดกึ่งกลางของหน้าต่างมีค่าเท่ากับ 1 ความเข้มแสงของจุดภาพในภาพผลลัพธ์ ณ ตำแหน่งที่ตรงกับกึ่งกลางของหน้าต่างที่ครอบอยู่บนภาพเริ่มต้น (จุด A) สามารถคำนวณได้จากค่าเฉลี่ยความเข้มแสงของทุกจุดในหน้าต่าง การหาค่าเฉลี่ยสามารถทำได้ 3 แบบคือ การหาค่าเฉลี่ยแบบคณิตศาสตร์ การหาค่าเฉลี่ยแบบมัธยฐาน และการหาค่าเฉลี่ยแบบฐานนิยม

1. การหาค่าเฉลี่ยแบบคณิตศาสตร์ ทำได้โดยการหาผลรวมของค่าความเข้มแสงของจุดทุกจุดในหน้าต่าง แล้วหารด้วยจำนวนจุดทั้งหมดในหน้าต่าง ค่าความเข้มแสงที่จุดอื่นๆสามารถคำนวณได้โดยการเลื่อนหน้าต่างให้จุดกึ่งกลางตรงกับจุดที่ต้องการหาค่า

2. การหาค่าเฉลี่ยแบบมัธยฐาน ทำได้โดยการนำค่าทั้งหมดในตารางมาเรียงลำดับจากน้อยไปหามาก (หรือจากมากไปหาน้อยก็ได้) จากนั้นจะเลือกค่าที่อยู่ตรงกลางของลำดับเป็นค่าความเข้มแสงของจุดในภาพผลลัพธ์ หากจำนวนจุดในหน้าต่างเป็นจำนวนคู่ ผลลัพธ์จะคำนวณได้จากการเฉลี่ยค่าระหว่างจุดกึ่งกลางทั้งสอง

3. การหาค่าเฉลี่ยแบบฐานนิยม ทำได้โดยการเลือกระดับความเข้มแสงที่ใช้บ่อยที่สุดในหน้าต่างมาเป็นคำตอบ ปัญหาที่อาจเกิดจากการใช้วิธีนี้ คือ มีระดับความเข้มแสงที่ใช้บ่อยที่สุดมากกว่า (มีหลายคำตอบ) วิธีการแก้ไข คือ การหาค่าเฉลี่ยหรือเปลี่ยนไปใช้การหาค่าเฉลี่ยแบบมัธยฐาน

นอกจากการหาค่าผลลัพธ์โดยวิธีหาค่าเฉลี่ยทั้ง 3 ดังที่ได้กล่าวมาแล้ว ยังมีวิธีการหาผลลัพธ์อีกวิธีหนึ่ง คือ การหาค่าเฉลี่ยจากจุด k จุดที่มีค่าความเข้มใกล้กับค่าความเข้มแสงของจุดกึ่งกลางของหน้าต่าง วิธีนี้เรียกว่า k-closest averaging การคำนวณหาผลลัพธ์ เริ่มจากการนำค่า

ความเข้มแสงของทุกจุดในหน้าต่างมาเรียงลำดับจากน้อยไปหามาก จากนั้นค่าที่อยู่รอบๆค่าของจุดกึ่งกลางหน้าต่างจำนวน k ค่า จะถูกเลือกมาเพื่อหาค่าเฉลี่ย ค่าเฉลี่ยนี้คือความเข้มแสงของจุดในภาพผลลัพธ์ ในการหาค่าเฉลี่ยอาจนำค่าของความเข้มสีที่จุดกึ่งกลางมาคิดด้วยก็ได้ รูปที่ 2.10 แสดงภาพผลลัพธ์ที่ได้จากวิธีต่างๆ (https://docs.google.com/viewer?a=v&q=cache: MB7z6GxWqz4J: fivedots.coe.psu.ac.th/~montri/Teaching/image/chap5.doc)

1 2 2 1	0 1 1 0	0 0 0 0	1 2 1 3
1 2 2 2	1 2 2 1	0 2 2 1	2 2 3 1
1 2 2 1	0 2 1 0	0 0 0 0	1 4 2 1
(ก)	(ข)	(ค)	(ง)

รูปที่ 2.10 การกรองข้อมูลภาพโดยใช้หน้าต่าง (ก) ใช้การเฉลี่ยแบบคณิตศาสตร์ (แสดงผลลัพธ์หลังการปัดเศษ) (ข) ใช้มัธยฐาน (ค) ใช้ฐานนิยม (ใช้มัธยฐานแทนสำหรับจุดที่มีปัญหา) (ง) ใช้ k-closest averaging (แสดงผลลัพธ์หลังการปัดเศษ)

2.5.7 การกรองโดยวิธีคอนโวลูชัน

วิธีการกรองข้อมูลภาพที่กล่าวมาส่วนใหญ่อาศัยหลักของการหาค่าเฉลี่ย โดยอาจเป็นการหาค่าเฉลี่ยของจุดเดียวกันจากภาพหลายๆภาพ หรืออาจเป็นการหาค่าเฉลี่ยจากจุดต่างๆที่อยู่รอบๆจุดที่สนใจ เนื่องจากการหาค่าเฉลี่ยเป็นการลดการเปลี่ยนแปลงของข้อมูล วิธีการที่ผ่านมาจึงใช้ได้ดีกับการกำจัดสัญญาณรบกวนที่เป็นสัญญาณความถี่สูง ดังที่ได้กล่าวมาแล้วว่า การกรองสัญญาณมีวัตถุประสงค์เพื่อเน้นคุณสมบัติบางอย่างที่ต้องการในภาพให้เด่นชัดขึ้น ในขณะที่ลดทอนคุณสมบัติที่ไม่ต้องการลง หากต้องการเน้นการเปลี่ยนแปลงของระดับความเข้มของจุดต่างๆภายในภาพให้เด่นชัดขึ้น ในที่นี้จะเสมือนกับการกรองสัญญาณความถี่สูงผ่าน จะไม่สามารถใช้วิธีการหาค่าเฉลี่ยได้ วิธีที่สามารถนำมาใช้ได้ คือ การคอนโวลูชัน

ในการประมวลผลภาพ การคอนโวลูชัน คือการกระทำกันระหว่างเทมเพลต กับภาพ เทมเพลต คือ เมตริกซ์ขนาด n x m ของชุดตัวเลขที่จะนำไปซ้อนทับภาพที่ตำแหน่งต่างๆ เพื่อหาผลลัพธ์ของการคอนโวลูชัน ถ้ากำหนดให้เทมเพลต T(x, y) เป็นเทมเพลตขนาด n x m และภาพ

I(X, Y) มีขนาด N x M การคอนโวลูชันระหว่างเทมเพลตกับภาพสามารถแสดงได้ดังสมการต่อไปนี้

$$I'(X,Y) = T * I = \sum_{i=0}^{n-1}\sum_{j=0}^{m-1} T(i,j)\cdot I(X-i,Y-j) \qquad (1)$$

โดย $I'(X,Y)$ คือ ภาพผลลัพธ์จากการคอนโวลูชัน

จากสมการ จะเห็นว่าระดับความเข้มแสง ณ จุด (X, Y) ในภาพผลลัพธ์ ได้จากการหาผลรวมของผลคูณของระหว่างค่าในเทมเพลตกับค่าระดับความเข้มแสงของภาพในบริเวณที่เทมเพลตซ้อนทับอยู่ จากสมการตัวชี้ตำแหน่งจุดในภาพ (X-i, Y-j) แสดงให้เห็นว่ามีการพลิกเทมเพลตทางแกนนอนและแกนตั้ง สมการที่ (2) แสดงการคอนโวลูชันที่ไม่ต้องมีการพลิกเทมเพลต ซึ่งวิธีการนี้มีชื่อที่แท้จริงว่า cross-correlation และเป็นที่นิยมใช้ในด้านการประมวลผลภาพ

$$I'(X,Y) = T * I = \sum_{i=0}^{n-1}\sum_{j=0}^{m-1} T(i,j)\cdot I(X+i,Y+j) \qquad (2)$$

ขั้นตอนของการคอนโวลูชันประกอบด้วยการเลื่อน บวก และคูณ สามารถใช้การคอนโวลูชันในการประมวลผลภาพได้ในหลายลักษณะ เช่น กรองสัญญาณภาพ การหาขอบภาพ (edge detection) หรือการหารูปทรงของวัตถุในภาพ เป็นต้น

โดยทั่วไป ในการคอนโวลูชันจะไม่ยอมให้มีการเลื่อนเทมเพลตออกนอกขอบเขตของภาพ ดังนั้นถ้าเทมเพลตมีขนาดใหญ่กว่า 1x1 ภาพผลลัพธ์จะมีขนาดเล็กกว่าภาพเริ่มต้นเสมอ ตัวอย่างเช่น การคอนโวลูชันระหว่างภาพขนาด 4 x 5 กับ เทมเพลตขนาด 2 x 2 ดังรูปที่ 2.11

เทมเพลต		ภาพเริ่มต้น		ภาพผลลัพธ์
1 0 0 1	*	1 1 3 3 4 1 1 4 4 3 2 1 3 3 3 1 1 1 4 4	=	2 5 7 6 * 2 4 7 7 * 3 2 7 7 * * * * * *

รูปที่ 2.11 ผลของการเติมค่าศูนย์รอบภาพเริ่มต้นก่อนทำการคอนโวลูชัน

จะให้ภาพผลลัพธ์ที่มีขนาด 3 x 4 จากตัวอย่างข้างต้น ค่าความเข้มสี 3 ในภาพผลลัพธ์ได้จากการหาผลรวมของผลคูณระหว่างเทมเพลตกับภาพในบริเวณที่แรเงา ซึ่งมีค่าเท่ากับ (1x2)+(0x1)+(0x1)+(1x1) = 3

จากตัวอย่าง พบว่าจุดมุมบนซ้ายของเทมเพลต คือ จุดอ้างอิงในการกำหนดจุดในภาพผลลัพธ์ ในความเป็นจริงแล้ว สามารถเลือกจุดใดๆในเทมเพลตให้เป็นจุดอ้างอิงก็ได้ การเลือกเอาจุดกึ่งกลางของเทมเพลตที่มีความกว้างและสูงเป็นจำนวนคี่ (เช่นเทมเพลตขนาด 3x3, 3x5, 5x5 และ 7x7 เป็นต้น) เป็นจุดอ้างอิงนับว่ามีความเหมาะสมยิ่ง อย่างไรก็ตาม ในแง่ของการเขียนโปรแกรมแล้ว การใช้จุดมุมเป็นจุดอ้างอิงจะลดความซับซ้อนในการเขียนโปรแกรมลง เนื่องจากไม่มีปัญหาสำหรับการคอนโวลูชันโดยใช้เทมเพลตที่มีขนาดไม่คงที่ ดังนั้นจะใช้จุดมุมบนซ้ายเป็นจุดอ้างอิงในการทำคอนโวลูชันทุกครั้ง เว้นแต่มีการกำหนดให้เป็นอย่างอื่น

การคอนโวลูชันที่ไม่ยอมให้เทมเพลตเลื่อนออกนอกบริเวณขอบภาพ เรียกว่า การคอนโวลูชันแบบไม่เป็นรายคาบ (aperiodic convolution) วิธีการนี้จะได้ภาพที่มีขนาดเล็กลง หากต้องการคงขนาดภาพไว้ จะต้องใช้การคอนโวลูชันแบบเป็นรายคาบ (periodic convolution) การคอนโวลูชันแบบนี้เปรียบเสมือนการม้วนภาพให้ขอบซ้ายมาชนกับขอบขวา และม้วนให้ขอบบนมาชนกับขอบล่าง เมื่อเทมเพลตเลื่อนตกขอบข้างใดข้างหนึ่งๆ ส่วนของเทมเพลตที่เลยขอบก็จะไปทาบกับขอบภาพอีกด้านหนึ่ง

วิธีการอย่างง่ายที่ทำให้ภาพผลลัพธ์มีขนาดเท่ากับภาพเริ่มต้น คือ การเติมค่าศูนย์บริเวณรอบๆภาพเริ่มต้น เพื่อทำให้ภาพเริ่มต้นมีขนาดใหญ่ขึ้น หลังจากการคอนโวลูชันจะได้ภาพผลลัพธ์ที่มีขนาดเท่ากับภาพเริ่มต้นก่อนมีการชดเชย รูปที่ 2.12 แสดงการคอนโวลูชันที่มีการเติมค่าศูนย์ให้กับภาพเริ่มต้น

รูปที่ 2.12 ผลของการเติมค่าศูนย์รอบภาพเริ่มต้นก่อนทำการคอนโวลูชัน

แม้เป็นกระบวนการประมวลผลภาพที่ง่าย แต่การคอนโวลูชันต้องใช้เวลาในการคำนวณสูง โดยเฉพาะอย่างยิ่งการคอนโวลูชันระหว่างภาพและเทมเพลตที่มีขนาดใหญ่ หากภาพมีขนาด M x M และเทมเพลตมีขนาด N x N จะต้องมีการคูณถึง M^2N^2 ครั้ง ถ้า M = 512 และ N = 16 จะต้องมีการคูณประมาณ 32 ล้านครั้ง การคำนวณที่มากขนาดนี้ทำให้ไม่สามารถประมวลผลภาพเคลื่อนไหวในเวลาจริง (real-time) ได้ เว้นแต่จะมีฮาร์ดแวร์ที่ออกแบบมาเฉพาะ สำหรับภาพและเทมเพลตที่มีขนาดใหญ่ (M >= 512 และ N >= 32) การแปลงภาพและเทมเพลตให้อยู่ในรูปของข้อมูลในโดเมนความถี่ (frequency domain) จะช่วยลดการคำนวณลงได้อย่างมาก จากตัวอย่างข้างต้น การคอนโวลูชันในโดเมนความถี่จะลดจำนวนครั้งของการคูณลงเหลือเพียง 256,000 ครั้งเท่านั้น การคอนโวลูชันในโดเมนความถี่จะได้กล่าวถึงในหัวข้อถัดไป

2.5.8 เทมเพลตสำหรับกรองความถี่ต่ำผ่าน

เทมเพลตขนาด 3 x 3 ต่อไปนี้ สามารถใช้เพื่อลดการเปลี่ยนแปลงอย่างฉับพลันของค่าความเข้มแสงในภาพ

1 1 1
1 1 1
1 1 1

ผลของการคอนโวลูชันกับเทมเพลตนี้ จะเหมือนกับการหาผลรวมของจุดภาพทั้ง 9 จุดที่เทมเพลตซ้อนทับอยู่ การเปลี่ยนแปลงใดๆในบริเวณดังกล่าวจะถูกเฉลี่ยให้มีความราบเรียบ ผล

ที่ได้คือสัญญาณที่มีการเปลี่ยนแปลง (เช่น สัญญาณรบกวนความถี่สูง) จะถูกลดทอน ภาพที่ได้จะมีความคมลดลง คุณสมบัติเช่นนี้เปรียบได้กับการกรองสัญญาณความถี่ต่ำผ่าน

เทมเพลตการกรองความถี่ต่ำผ่านที่นิยมใช้อีกแบบหนึ่ง คือ

$$\begin{matrix} 1 & 3 & 1 \\ 3 & 16 & 3 \\ 1 & 3 & 1 \end{matrix}$$

เทมเพลตนี้จะเน้นความสำคัญของจุดที่อยู่ตรงกลางเทมเพลตเป็นพิเศษ โดยจะให้จุดกลางมีน้ำหนัก 50% ของทั้งหมด และให้น้ำหนักรวมของจุดทั้ง 4 ที่อยู่ด้านบน ด้านล่าง ด้านซ้าย และด้านขวาของจุดกลางมีค่าเท่ากับ 40% ส่วนจุดมุมทั้ง 4 มีน้ำหนักเพียง 10% โดยจุดที่อยู่ใกล้จุดศูนย์กลางจะมีน้ำหนักมากกว่าจุดที่อยู่ห่างออกไป

2.5.9 เทมเพลตสำหรับกรองความถี่สูงผ่าน

สัญญาณความถี่สูง คือ สัญญาณที่มีการเปลี่ยนแปลงค่าไปมาอย่างรวดเร็ว ต่างกับสัญญาณความถี่ต่ำ ซึ่งมีการเปลี่ยนค่าอย่างช้าๆหรือไม่เปลี่ยนแปลงเลย การกรองความถี่สูงผ่าน (High pass filter) ก็คือการกรองสัญญาณที่เพิ่มความแรงของสัญญาณที่มีความถี่สูง และลดความแรงของสัญญาณที่มีความถี่ต่ำ เทมเพลตต่อไปนี้ใช้สำหรับการกรองความถี่สูงผ่าน

$$\begin{matrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{matrix}$$

จะเห็นว่า ผลรวมของทุกค่าในเทมเพลตมีค่าเท่ากับศูนย์ ซึ่งหมายความว่า ถ้าวางเทมเพลตนี้ลงบนบริเวณของภาพที่มีค่าความเข้มแสงคงที่ ผลลัพธ์ที่ได้จะมีค่าเป็นศูนย์ อย่างไรก็ตาม

ถ้าค่าที่บริเวณตรงกลางแตกต่างกับค่ารอบๆ ผลลัพธ์ที่ได้จะแสดงค่าความแตกต่างยิ่งขึ้นรูปที่ 2.13 แสดงตัวอย่างการกรองภาพด้วยเทมเพลตกรองความถี่สูง และความถี่ต่ำ

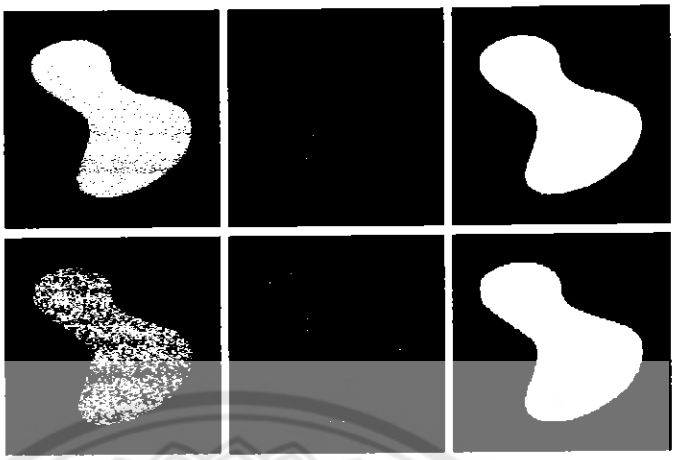
ภาพ	หลังจากการกรองความถี่สูงผ่าน	หลังจากการกรองความถี่ต่ำผ่าน
0 0 0 0 0		
0 1 1 1 0	2 1 2	4 6 4
0 1 1 1 0	1 0 1	6 9 6
0 1 1 1 0	1 0 1	6 9 6
0 1 1 1 0	1 –5 1	11 14 11
0 1 6 1 0	-4 20 -4	11 14 11
0 1 1 1 0	2 –4 2	9 11 9
0 0 0 0 0		

รูปที่ 2.13 การกรองภาพด้วยเทมเพลตกรองความถี่สูง และความถี่ต่ำ

จากรูปที่ 2.13 จะเห็นว่าหลังการกรองความถี่สูงผ่านขอบภาพจะเด่นชัด ส่วนที่เป็นค่าคงที่จะกลายเป็นศูนย์ และส่วนที่มีการเปลี่ยนแปลงจาก 1 เป็น 6 ถูกขยายเป็นจาก –4 ไป 20 สำหรับการกรองความถี่ต่ำผ่าน การเปลี่ยนถูกลดทอนลง ภาพผลลัพธ์มีความราบเรียบขึ้น

2.6 การแยกส่วนภาพ

การแยกส่วนภาพ เป็นการแยกส่วนของข้อมูลภาพโดยแบ่งขอบเขต หรือส่วนของวัตถุในภาพออกเป็นส่วนๆ มักทำหลักจากการปรับปรุงคุณภาพของภาพให้ดีขึ้นแล้ว ลักษณะการแยกขึ้นอยู่กับวัตถุประสงค์ที่จะนำไปใช้งานต่อ เช่น เพื่อวิเคราะห์รูปลักษณะของวัตถุ เพื่อคำนวณขนาดของวัตถุ เพื่อนับจำนวนวัตถุภายในภาพ เป็นต้น



รูปที่ 2.14 ตัวอย่างผลลัพธ์ประเภทการแยกส่วนภาพ

ที่มา: http://www.ce.kmitl.ac.th

หลักการพื้นฐานที่ใช้ในการแบ่งส่วนภาพ มีดังนี้

2.6.1 Point , Line , and Edge Detection เป็นเทคนิคการตรวจจับความ**ไม่ต่อเนื่องของ**ภาพโดยมีวิธีการตรวจจับ 3 วิธีคือ

2.6.1.1 ตรวจจับจุด

2.6.1.2 ตรวจจับเส้น

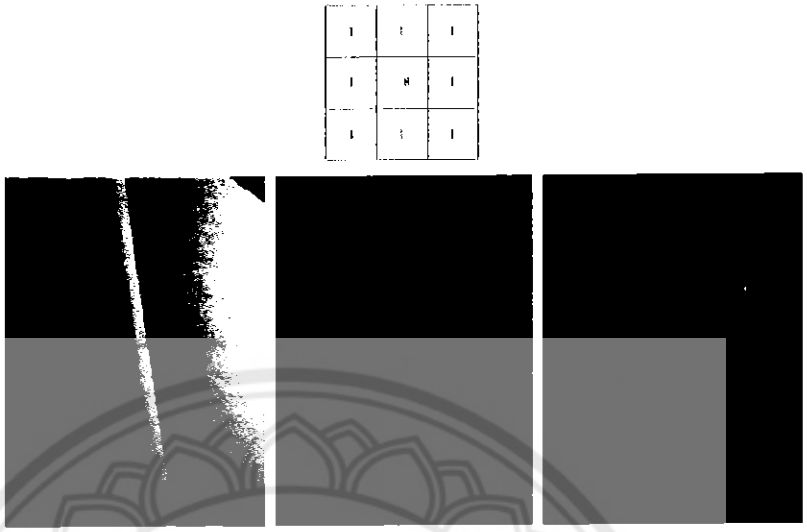
2.6.1.3 ตรวจจับขอบภาพ

2.6.1.4 ตรวจสอบความไม่ต่อเนื่องของภาพจากการใช้ Mask มา

การคำนวณซึ่งผลลัพธ์ที่ได้จะเป็นค่าผลรวมของค่าที่อยู่ในแต่ละพิ**กเซลของ** Mask คูณกับค่าสัมประสิทธิ์ (http://www.ce.kmitl.ac.th)

2.6.2 Detection of Isolated Point เป็นการตรวจจับหาจุดที่แตกต่างจาก**จุดอื่นของภาพ**โดยจุดที่แตกต่างนี้จะถูกกำหนดให้อยู่ตรงกลางของ Mask ซึ่งการที่จุดแต่ละจุดจะ**ถูกตรวจจับใน**

ภาพที่ 2.15 Detection of Isolated Point



รูปที่ 2.15 Detection of Isolated Point

ที่มา: http://www.ce.kmitl.ac.th

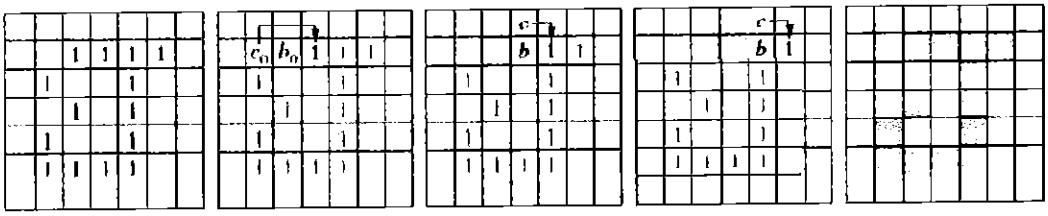
2.6.3 Image Representation เป็นการแทนรูปแบบในภาพ เพื่อให้เหมาะแก่การประมวลผลขั้นต่อไป โดยนำผลที่ได้จากการแบ่งส่วนภาพ มาประมวลผลทำการแทนรูปแบบในภาพ การแทนรูปแบบในบริเวณใดบริเวณหนึ่ง แบ่งได้เป็น 2 แนวทาง

1. แทนที่รูปแบบของลักษณะภายนอก (Boundary)
2. แทนที่รูปแบบของลักษณะภายใน (The pixels comprising the region)

การแบ่งส่วนภาพจะได้ผลลัพธ์ซึ่งเป็นข้อมูลจริงจากจุดภาพ ที่เป็นขอบเขตหรือจุดภายในภาพ ซึ่งมักจะใช้ข้อมูลจากการแบ่งส่วนภาพมาแทนที่เพื่อประโยชน์ต่อการการคำนวณ หรือใช้ในการบีบอัดข้อมูลที่ใช้เก็บให้น้อยลง

2.6.4 Representation

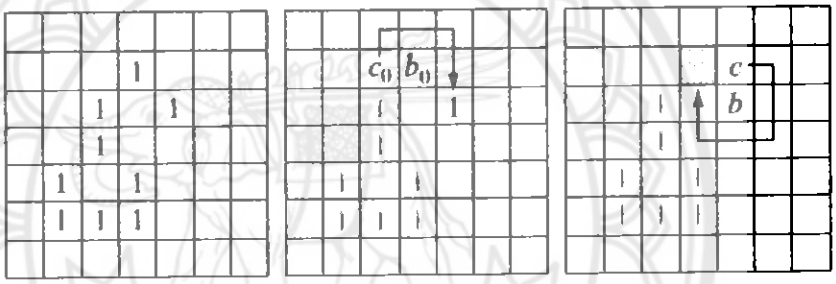
1. Boundary (Border) Following คือ กระบวนการในการแทนรูปแบบ จำเป็นต้องกำหนดจุดเริ่มต้นในรูปแบบ (ในตัวอย่าง เริ่มจากด้านบนสุด และซ้ายสุด) แล้วตรวจสอบทิศทางแบบตามเข็มนาฬิกา (หรือทวนเข็มนาฬิกา) วนจนครบกลับมายังจุดเริ่มต้น



รูปที่ 2.16 Boundary (Border) Following

ที่มา: http://www.ce.kmitl.ac.th

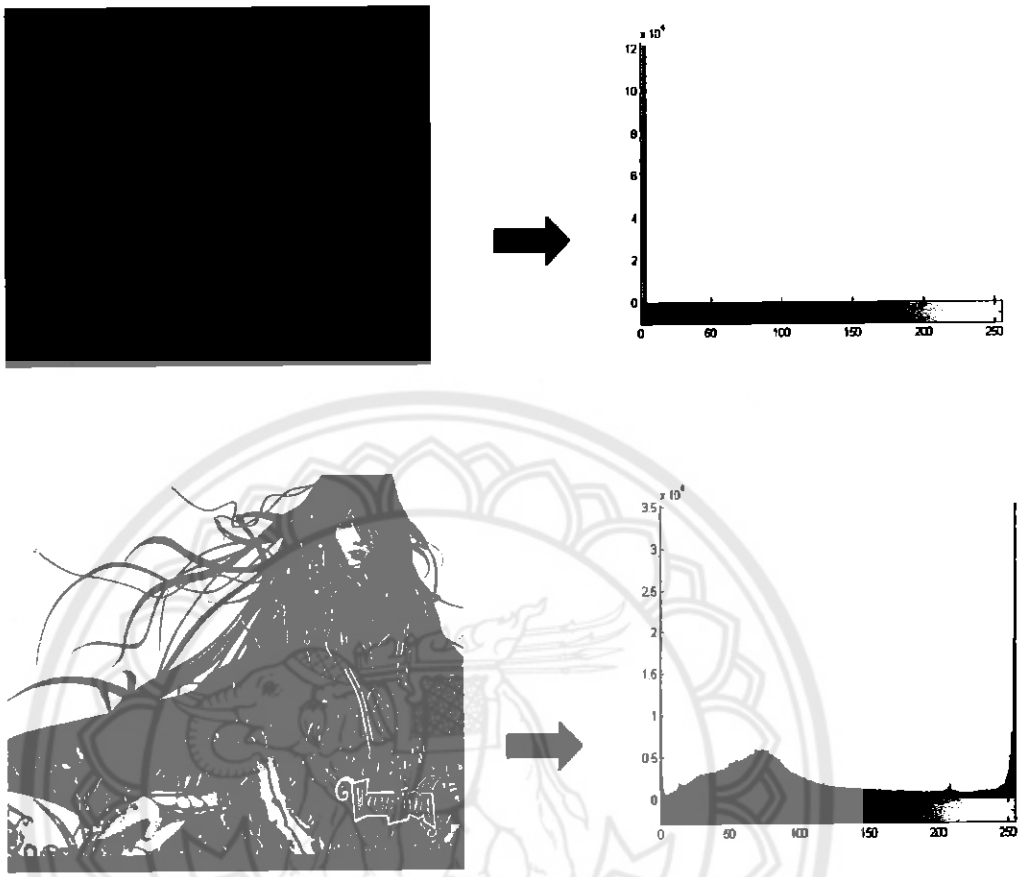
ในการหาขอบเขตจะหยุดทำกระบวนการเมื่อวนกลับมายังจุดเริ่มต้น ซึ่งมีบางกรณีที่ทำให้การหาขอบเขตด้วยวิธีนี้ไม่สำเร็จ เช่น



2.7 ภาพกราฟแจกแจงความถี่ (Image Histogram)

เนื่องจากข้อมูลทางสถิติจะอยู่ภายใต้ความแปรผันเสมอในการวิเคราะห์ ดังนั้นการพิจารณาจะทำได้ง่าย ถ้าข้อมูลอยู่ถูกสรุปให้อยู่ในรูปฮิสโตแกรม ซึ่งฮิสโตแกรม หมายถึง กราฟที่แสดงความแปรผันของข้อมูล ทั้งแนวโน้มสู่ศูนย์กลาง ค่าการกระจายและรูปทรงของการแปรผัน

ฮิสโตแกรมของภาพ ภาพโทนสีเทา เป็นกราฟแสดงความถี่หรือจำนวนพิกเซลที่ระดับความสว่างแต่ละค่า โดยมีแกนนอนเป็นระดับความเข้มของแสง ตั้งแต่ 0 ถึง 255 และมีแกนตั้งแสดงจำนวนพิกเซลที่ความเข้มแสงที่ตำแหน่งนั้นๆ เช่น ถ้าตำแหน่งความเข้มของแสงใดมีความถี่สูงที่สุดก็แสดงว่าภาพนั้นมีพิกเซลที่มีค่าความเข้มของแสงที่ค่านั้นมากที่สุด ดังตัวอย่างของฮิสโตแกรมของภาพตัวอย่าง



รูปที่ 2.17 ตัวอย่าง Histogram ของภาพ

ที่มา: www.media.rmutt.ac.th

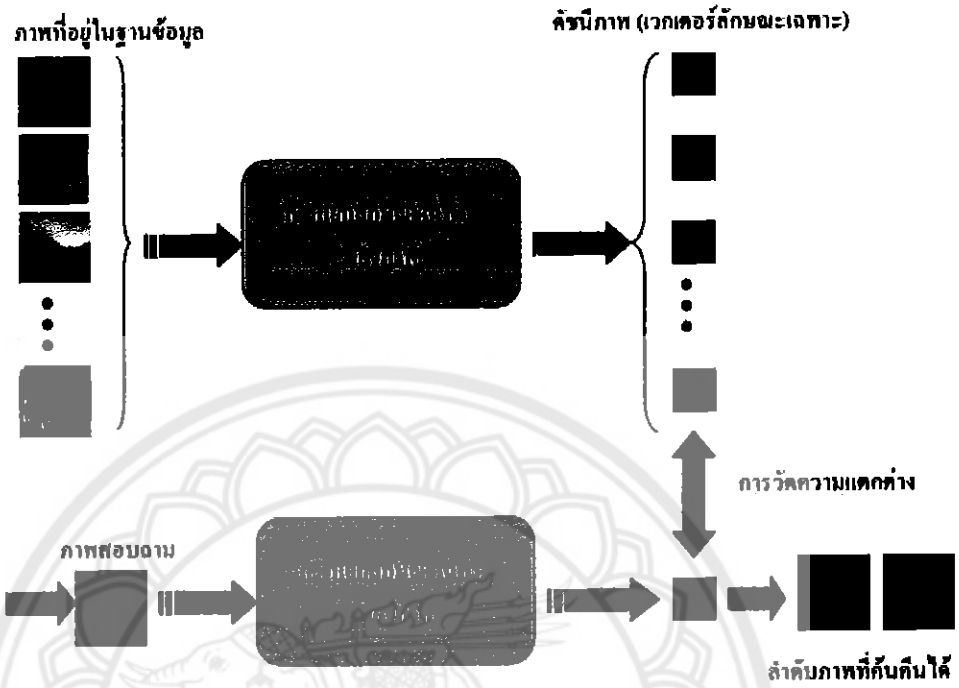
2.8 ระบบการค้นคืนภาพด้วยเนื้อหา (Content-Based Image Retrieval: CBIR)

คำนิยามของระบบการค้นคืนภาพด้วยเนื้อหาหรือที่เรียกกันว่าระบบ CBIR นั้น ถูกตั้งขึ้นโดย Kato เมื่อปี 1992 เพื่อใช้อธิบายลักษณะการค้นคืนภาพจากฐานข้อมูล โดยอาศัยลักษณะเฉพาะทางสีและรูปร่าง ซึ่งระบบสามารถดึงเอาลักษณะเฉพาะเหล่านั้นของภาพออกมาได้อย่างอัตโนมัติ และทำการค้นคืนภาพโดยใช้ลักษณะเฉพาะของภาพในการสอบถามระบบการค้นคืนภาพส่วนใหญ่ประกอบด้วย 3 กระบวนการหลัก ๆ ได้แก่

- การแยกลักษณะเฉพาะของภาพ (Image Feature Extraction)
- การสร้างดัชนีภาพ (Image Indexing)
- การค้นคืนภาพ (Image Retrieval)

ระบบการค้นคืนภาพด้วยเนื้อหา แสดงดังภาพที่ 2.18

ห้องสมุดคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร



รูปที่ 2.18 แผนภาพแสดงระบบการค้นคืนภาพด้วยเนื้อหา (CBIR)

ที่มา: www.media.rmutt.ac.th

2.8.1 การแยกลักษณะเฉพาะของภาพ (Image Feature Extraction)

การแยกลักษณะเฉพาะของภาพ เป็นการแยกหรือสกัดเอาข้อมูลที่สำคัญของภาพออกมา ซึ่งลักษณะเฉพาะของภาพเป็นคุณสมบัติที่สามารถหาได้โดยใช้ขั้นตอนวิธีการประมวลผลภาพโดยที่ลักษณะเฉพาะพื้นฐานของภาพประกอบด้วย 3 ส่วน คือ สี รูปร่าง และพื้นผิว (www.media.rmutt.ac.th)

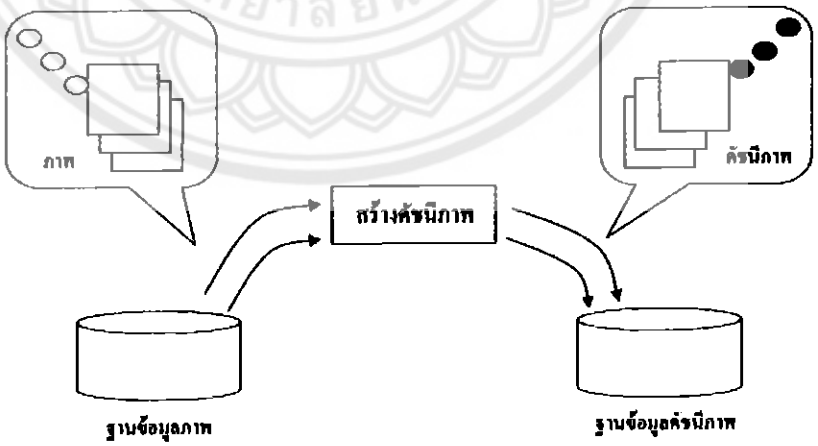
2.8.1.1 สี เป็นลักษณะเฉพาะของภาพที่มีบทบาทสำคัญในระบบค้นคืนภาพ เช่น ฮิสโตแกรมสี ซึ่งเป็นลักษณะเฉพาะของสีที่ถูกนำมาใช้บ่อยๆ เนื่องจากสีเป็นสิ่งที่สามารถมองเห็นได้ง่าย และเป็นสิ่งแรกที่สามารถสังเกตเห็นได้จากการมองภาพ นอกจากนี้ สียังสามารถใช้ในการแยกแยะกลุ่มของภาพออกตามเนื้อหาได้เป็นอย่างดี เช่น สีฟ้าของน้ำทะเล สีแดงของดอกไม้ สีเขียวของต้นไม้ เป็นต้น

2.8.1.2 รูปร่าง เป็นลักษณะเฉพาะของภาพที่ใช้อธิบายถึงรูปร่างและลักษณะ รวมถึงขนาดของวัตถุภายในภาพ ซึ่งทำให้สามารถแยกวัตถุออกจากพื้นหลัง หรือแยกแยะระหว่างวัตถุที่มีรูปร่างแตกต่างกันออกจากกันได้

2.8.1.3 พื้นผิว เป็นลักษณะเฉพาะที่ใช้อธิบายความหยาบ ความละเอียด หรือความซับซ้อนของวัตถุภายในภาพ ซึ่งแต่ละภาพอาจจะประกอบด้วยวัตถุที่มีลักษณะพื้นผิวที่แตกต่างกันออกไป การวิเคราะห์พื้นผิวจะช่วยให้สามารถแยกแยะความแตกต่างของวัตถุได้ดียิ่งขึ้น การค้นคืนภาพที่ใช้พื้นผิวเป็นลักษณะเฉพาะของภาพ ส่วนใหญ่จะถูกนำไปใช้ในการค้นหาภาพจากกลุ่มภาพพื้นผิว เช่น ชุดภาพพื้นผิวของหิน ชุดภาพพื้นผิวของใบไม้ เป็นต้น ลักษณะเฉพาะของภาพที่เหมาะสมในการค้นคืนภาพจะแตกต่างกันออกไป ดังนั้นในการค้นคืนภาพแต่ละประเภทจึงต้องเลือกใช้ลักษณะเฉพาะของภาพที่เหมาะสมกับภาพประเภทนั้นๆ เพื่อให้ผลลัพธ์ที่ได้จากการค้นคืนมีประสิทธิภาพมากที่สุด

2.8.2 การสร้างดัชนีภาพ (Image Indexing)

การสร้างดัชนีภาพ เป็นการสร้างดัชนีเพื่อใช้ในการแสดงถึงแต่ละภาพในฐานข้อมูล โดยทั่วไปดัชนีของภาพจะได้มาจากการแยกลักษณะเฉพาะของภาพ (Image Feature) ซึ่งดัชนีของภาพเหล่านี้จะถูกจัดเก็บเอาไว้อยู่ในฐานข้อมูล ดัชนีภาพลักษณะเฉพาะของภาพแต่ละภาพที่ถูกแยกออกมาจะอยู่ในรูปของค่าที่เป็นตัวเลข จำนวน n ค่า (ขึ้นอยู่กับเงื่อนไขของแต่ละวิธี) หรือเวกเตอร์ขนาด n มิติ ซึ่งก็คือเวกเตอร์ลักษณะเฉพาะ (Feature Vector) ของภาพนั้นๆนั่นเอง ขั้นตอนการสร้างดัชนีภาพแสดงดังรูปที่ 2.19 กล่าวคือ ภาพที่นำเข้ามาในฐานข้อมูลทุกภาพจะต้องผ่านกระบวนการสร้างดัชนีภาพและถูกนำไปจัดเก็บไว้ในฐานข้อมูลดัชนีภาพ

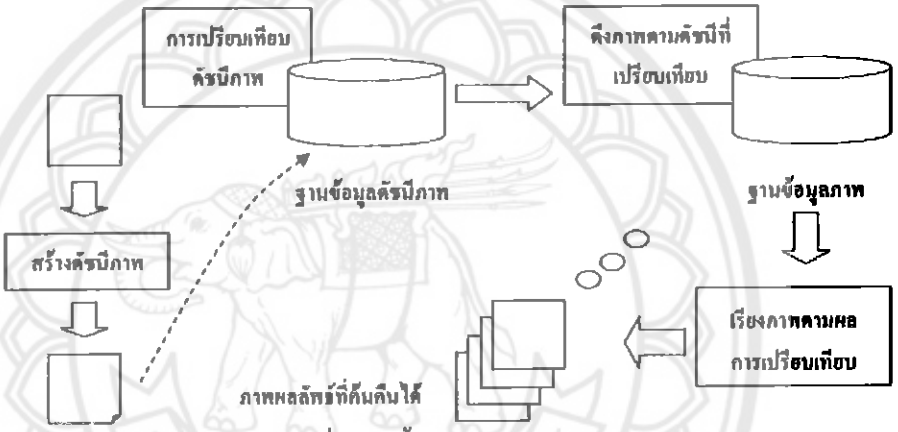


รูปที่ 2.19 ขั้นตอนการสร้างดัชนีภาพ

ที่มา: www.media.rmutt.ac.th

2.8.3 การค้นคืนภาพ (Image Retrieval)

ขั้นตอนแรกของกระบวนการค้นคืนภาพ คือ การเลือกภาพสอบถาม (Query Image) จากนั้นจึงทำการดึงเอาเวกเตอร์ลักษณะเฉพาะของภาพสอบถามออกมา และนำไปเปรียบเทียบกับเวกเตอร์ลักษณะเฉพาะของภาพในฐานข้อมูล เพื่อค้นคืนภาพที่มีลักษณะคล้ายกับภาพสอบถามที่สุดออกมา การเปรียบเทียบความแตกต่างระหว่างภาพ 2 ภาพ สามารถทำได้โดยการคำนวณระยะทางระหว่างเวกเตอร์ลักษณะเฉพาะของภาพทั้งสอง โดยที่ความแตกต่างระหว่างภาพทั้งสองจะแปรผันตามระยะทางระหว่างเวกเตอร์ลักษณะเฉพาะ ขั้นตอนการค้นคืนภาพแสดงดังรูปที่ 2.20



รูปที่ 2.20 ขั้นตอนการค้นคืนภาพ

2.8.4 การสอบถาม (Querying)

วิธีการสอบถามหาภาพที่ต้องการจากฐานข้อมูล สามารถกระทำได้หลายวิธี ได้แก่

1. การสอบถามด้วยลักษณะเฉพาะอย่างง่ายที่มองเห็นได้ (Simple Visual Feature Query) เป็นการสอบถามโดยการระบุค่าลักษณะเฉพาะของภาพที่ต้องการเป็นระดับเปอร์เซ็นต์ เช่น ต้องการหาภาพที่มีองค์ประกอบสีฟ้า 40% สีแดง 30% และสีดำ 30% เป็นต้น

2. การสอบถามด้วยลักษณะเฉพาะหลายแบบร่วมกัน (Feature Combination Query) เป็นการสอบถามที่ผู้ใช้สามารถใช้ลักษณะเฉพาะของภาพมากกว่า 1 แบบในการสอบถาม โดยมีการกำหนดค่าน้ำหนักให้กับลักษณะเฉพาะแต่ละแบบ เช่น ต้องการหาภาพที่มีสีเขียว 75% และมีพื้นผิวเป็นลายใบไม้ 25% เป็นต้น

3. การสอบถามด้วยภาพตัวอย่าง (Query By Example) เป็นการป้อนตัวอย่างภาพเข้ามาเป็นภาพสอบถาม เพื่อค้นคืนภาพที่ต้องการจากฐานข้อมูล โดยผู้ใช้สามารถเลือกภาพสอบถามและลักษณะเฉพาะของภาพที่ต้องการ จากนั้นระบบก็จะนำลักษณะเฉพาะของภาพไปเปรียบเทียบ

กับภาพในฐานข้อมูล เพื่อค้นคืนภาพที่มีลักษณะเฉพาะคล้ายกับภาพสอบถามที่สุดออกมาสำหรับปริญญานิพนธ์นี้ เลือกใช้การสอบถามด้วยภาพตัวอย่าง ซึ่งสามารถรองรับการใช้งานของผู้ใช้ได้หลากหลาย และเป็นวิธีที่ค่อนข้างง่ายและสะดวกในการใช้งาน โดยที่ผู้ใช้ไม่จำเป็นต้องมีความรู้เกี่ยวกับระบบหรือลักษณะเฉพาะของภาพก็สามารถทำการค้นคืนภาพได้

2.9 ลักษณะเฉพาะของภาพ (Image Feature)

ลักษณะเฉพาะของภาพ เป็นปัจจัยหนึ่งที่มีผลกระทบต่อประสิทธิภาพของระบบการค้นคืนภาพ เนื่องจากการเลือกใช้ลักษณะเฉพาะของภาพที่เหมาะสมกับการค้นคืนภาพแต่ละภาพนั้น เป็นสิ่งที่สำคัญ เพราะหากเลือกใช้ไม่เหมาะสม ก็อาจทำให้เกิดความผิดพลาดต่อภาพผลลัพธ์ที่ได้จากการค้นคืนได้ ลักษณะเฉพาะของภาพที่สามารถมองเห็นได้ สามารถแบ่งได้เป็น 3 ระดับ ดังนี้

2.9.1 ลักษณะเฉพาะของภาพในระดับต่ำ (Low-level Image Features)

เป็นลักษณะเฉพาะที่หาได้จากภาพโดยตรง เช่น ข้อมูลสี รูปร่าง และพื้นผิว

2.9.2 ลักษณะเฉพาะของภาพในระดับกลาง (Middle-level Image Features)

เช่น บริเวณหรือขอบเขต (Region) เป็นลักษณะเฉพาะที่ต้องเอาลักษณะเฉพาะในระดับต่ำมาผ่านกระบวนการประมวลผลภาพบางอย่างก่อน เช่น การแบ่งส่วนของภาพแล้วนำเอาลักษณะเฉพาะของทั้งขอบหรือบริเวณไปใช้งาน เป็นต้น

2.9.3 ลักษณะเฉพาะของภาพในระดับสูง (High-level Image Features)

ลักษณะเฉพาะชนิดนี้ต้องใช้ความสามารถของมนุษย์เป็นผู้ช่วยในการกำหนด เช่น การแบ่งกลุ่มสีร้อน-เย็น เป็นต้น ระบบการค้นคืนภาพส่วนใหญ่นิยมนำเอาลักษณะเฉพาะของภาพในระดับต่ำมาใช้งาน เนื่องจากสามารถแยกได้อย่างอัตโนมัติ ประมวลผลได้ง่าย นอกจากนี้ยังสะดวกและรวดเร็วในการค้นคืนอีกด้วย ลักษณะเฉพาะของภาพในระดับต่ำซึ่งประกอบด้วย สี รูปร่าง และพื้นผิว แต่ในงานปริญญานิพนธ์นี้จะกล่าวถึงลักษณะเฉพาะของภาพประเภทสีเท่านั้น

2.10 ความรู้เบื้องต้นเกี่ยวกับไลบรารีเอฟอร์ต

AForge.NET สร้างมาสำหรับภาษา C# โดยเฉพาะ โดยออกแบบมารองรับฟังก์ชั่นต่างๆ สำหรับใช้ในการประมวลผลภาพ AForge.NET เป็นเฟรมเวิร์คสำหรับภาษา C# ที่เป็นโอเพนซอร์ซที่ถูกพัฒนาให้สามารถนำไปใช้งานทางด้านคอมพิวเตอร์วิทัศน์ศาสตร์ และปัญญาประดิษฐ์ ได้แก่ การประมวลผลภาพ โครงข่ายประสาทเทียม การประยุกต์ใช้พันธุกรรม เป็นต้น สำหรับเฟรมเวิร์คใน AForge.NET จะประกอบด้วยชุดของไลบรารีรวมถึงตัวอย่างการใช้งาน ได้แก่

2.10.1 AForge.Imaging - เป็นไลบรารีสำหรับใช้งานในด้านการประมวลผลภาพ

2.10.2 AForge.Vision - เป็นไลบรารีสำหรับใช้งานในด้านคอมพิวเตอร์วิทัศน์ AForge.Video - เป็นไลบรารีสำหรับใช้งานในด้านการประมวลผลวิดีโอ

2.10.3 AForge.Neuro - เป็นไลบรารีสำหรับใช้งานในด้านโครงข่ายประสาทเทียม

2.10.4 AForge.MachineLearning – เป็นไลบรารีสำหรับใช้งานการเรียนรู้ของเครื่อง

บทที่ 3
ขั้นตอนการดำเนินงาน

ในการดำเนินการ โปรแกรมค้นหาความคล้ายคลึงของรูปภาพนั้น จะเป็นการสร้างโปรแกรมค้นหาความคล้ายคลึงของรูปภาพโดยให้โปรแกรมทำการประมวลผลการค้นหาความคล้ายคลึงของภาพ โดยใช้หลักการค้นหาฮีสโตแกรม ในการสืบค้นภาพจากภาพตัวอย่างที่ต้องการ โดยโปรแกรมที่ใช้ค้นหาพัฒนาด้วยโปรแกรม Microsoft Visual Studio 2010และ AForge.NETโดยได้แบ่งการดำเนินงานออกเป็นขั้นตอนดังนี้

รูปที่ 3.1 ขั้นตอนการพัฒนาซอฟต์แวร์

3.1 ภาพโดยรวมในการค้นหาภาพของระบบ

3.1.1 การค้นหาภาพ

โปรแกรมที่ใช้ค้นหาภาพนั้น ได้พัฒนาขึ้นภายใต้สถาปัตยกรรม .NET โดยใช้ภาษา C# ซึ่งการที่จะค้นหาภาพได้นั้น ต้องมีข้อมูลของภาพอยู่ในแฟ้มข้อมูลก่อน โดยพัฒนาโปรแกรมขึ้นเพื่อวิเคราะห์องค์ประกอบของภาพให้ออกมาเป็นค่าตัวเลข ซึ่งใช้แทนองค์ประกอบของสีภายในภาพ การค้นหานั้นจะเริ่มจากการนำภาพต้นแบบมาวิเคราะห์องค์ประกอบของสี เพื่อให้ได้ค่าเป็นตัวเลขแล้วนำไปเปรียบเทียบกับข้อมูลของภาพในแฟ้มข้อมูล เพื่อที่จะหาภาพที่ใกล้เคียงกับภาพต้นแบบมากที่สุด และนำไปแสดงผล โดยลำดับการแสดงผลจะเรียงลำดับความคล้ายคลึงกันของภาพจากความคล้ายมากลงมาตามลำดับ แต่อาจจะได้ภาพที่ตรงกับความต้องการไม่ทุกภาพ สุดท้ายโปรแกรมก็จะแสดงภาพที่ได้จากการประมวลผลออกมาแสดง

ขั้นตอนการทำงานของระบบการค้นหาภาพที่มีความคล้ายคลึงกัน

1. ทำการเลือกรูปภาพเพื่อนำมาเป็นแฟ้มข้อมูลโปรแกรมจะสนับสนุนรูปแบบไฟล์ภาพแบบบิตแมป (*JPG, *JPEG, *JPE.)เท่านั้น

2. ทำการสร้างแฟ้มข้อมูลของภาพเพื่อใช้ในการเก็บข้อมูลภาพที่ใช้ในการค้นหา

3. ทำการเลือกภาพต้นแบบ

4. นำภาพที่ได้ไปทำการประมวลผลภาพเบื้องต้น โดยใช้ ฮิสโตแกรมของภาพเป็นการเก็บค่าความถี่ของค่าระดับความเข้มแสงหรือสี

5. นำภาพที่ได้ไปทำการประมวลผลภาพเบื้องต้น โดยใช้ฟูเรียร์ทรานส์ฟอร์ม (Fourier Transform) เพื่อลดสัญญาณรบกวน และทำให้ภาพมีความชัดเจนมากขึ้น

6. ค้นหาภาพที่คล้ายคลึงกันและที่ซ้ำกันทั้งหมดที่อยู่ในแฟ้มข้อมูลของผู้ใช้ค้นหาภาพที่คล้ายคลึงกันของผู้ใช้ที่ระบุโดยใช้ภาพตัวอย่างเปรียบเทียบ

7. ผู้ใช้งานสามารถเลือกภาพตัวอย่างได้ โดยกดปุ่ม Find Duplicates ระบบจะแสดงภาพตัวอย่างที่อยู่ในแฟ้มข้อมูลซึ่งมีความคล้ายคลึงกันของภาพที่มีอยู่ในแฟ้มข้อมูล

8. ระบบทำการให้คะแนนแต่ละภาพ

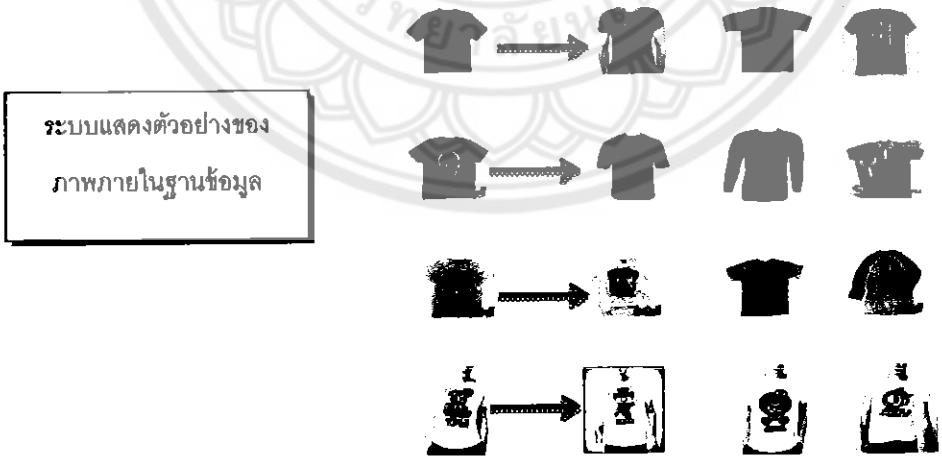
9. จัดเรียงลำดับคะแนนของภาพในแฟ้มข้อมูล

10. พัฒนาโปรแกรมเพื่อการให้คะแนน และจัดเรียงลำดับของภาพ

11. ระบบทำการประมวลผลภาพที่มีความคล้ายกันเพื่อแสดงผล

รูปที่ 3.2 ตัวอย่างการค้นหาจากภาพต้นแบบ

จากรูปที่ 3.2 เมื่อทำการเลือกภาพต้นแบบที่ใช้ในการค้นหา โปรแกรมจะทำการค้นหาภาพที่มีความคล้ายคลึงกับภาพต้นแบบออกมาแสดงผล โดยใช้การเปรียบเทียบจากค่า Hue

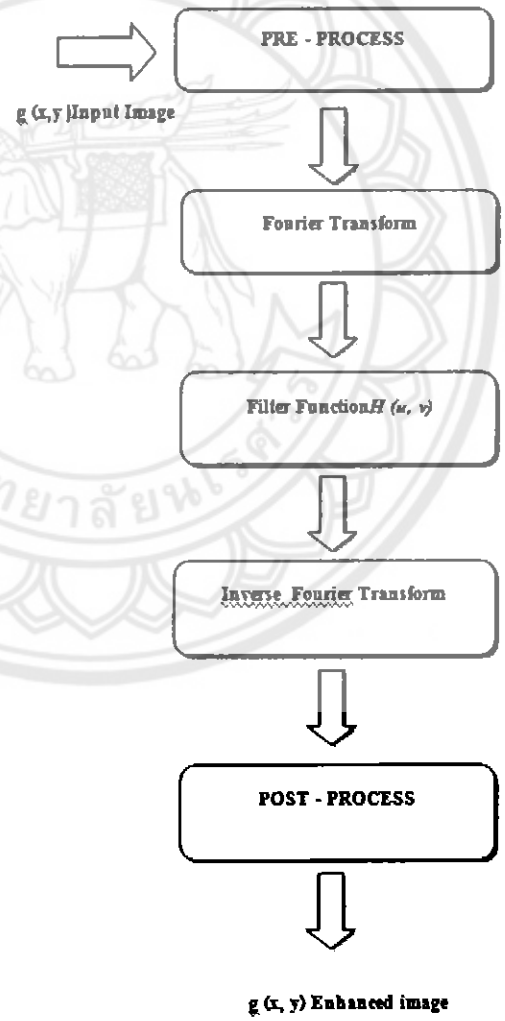


รูปที่ 3.3 ระบบแสดงตัวอย่างของการค้นหาภาพจากภาพภายในแฟ้มข้อมูล

จากรูปที่ 3.3 ตัวอย่างการค้นหาภาพจากภาพในแฟ้มข้อมูล โปรแกรมทำการแสดงตัวอย่างของภาพภายในแฟ้มข้อมูลที่มีความคล้ายคลึงกัน

3.1.2 การออกแบบโปรแกรมการค้นหาความคล้ายคลึงกันของภาพ

ออกแบบพัฒนาระบบในการค้นหาความคล้ายคลึงกันของภาพพบว่าภาพที่คล้ายกันของภาพภายในแฟ้มข้อมูลของคอมพิวเตอร์ ได้จากการค้นหาภาพที่คล้ายคลึงกันโดยใช้วิธีฮีสโตแกรม (Histogram) และฟูเรียร์ทรานส์ฟอร์ม (Fourier Transform) เพื่อค้นหาภาพที่คล้ายคลึงกันวิธีการดังกล่าวสามารถหาภาพที่มีคล้ายคลึงกันแม้ว่าจะอยู่ในรูปแบบภาพที่แตกต่างกันที่ระดับความค่าของความเข้มแสงสีที่แตกต่างหรือขนาดของภาพ โดยระบบจะคำนวณของความคล้ายคลึงกันของภาพที่จะนำมาใช้ในการค้นหา สามารถระบุความแตกต่างของภาพและขนาด เมื่อกระบวนการเสร็จสิ้นการค้นหา โปรแกรมก็จะแสดงผลลัพธ์แสดงให้เห็นภาพที่คล้ายคลึงกันออกมา



รูปที่ 3.4 กระบวนการการปรับปรุงภาพ

3.1.3 การประมวลผลภาพเบื้องต้น(Pre-Processing)

การปรับปรุงภาพให้เหมาะสมกับงานเฉพาะทาง เพื่อทำให้ภาพชัดเจนขึ้นในสายตามนุษย์ เนื่องมาจากภาพที่ทำการรับเข้ามาจะมีความไม่ชัดเจน เช่น ปรับปรุงความคมชัด ปรับความสว่าง หาขอบเขตของวัตถุให้ชัดเจนขึ้น กำจัดสิ่งรบกวนหรือนอยส์ ซึ่งการปรับปรุงคุณภาพของภาพเนื่องจากมีปัจจัยภายนอกเข้ามากระทำ เช่น การถ่ายภาพนั้นมีความสว่างน้อยเกินไป หรือขาดความคมชัด ดังนั้นเราจึงต้องทำการปรับปรุงภาพ เพื่อทำให้ภาพมีความชัดเจนขึ้น และช่วยปรับปรุงคุณภาพของภาพนั้นให้ดีขึ้นทำให้ระบบมีการประมวลถูกกต้องแม่นยำยิ่งขึ้น

3.1.3.1 ภาพระดับสีRGB

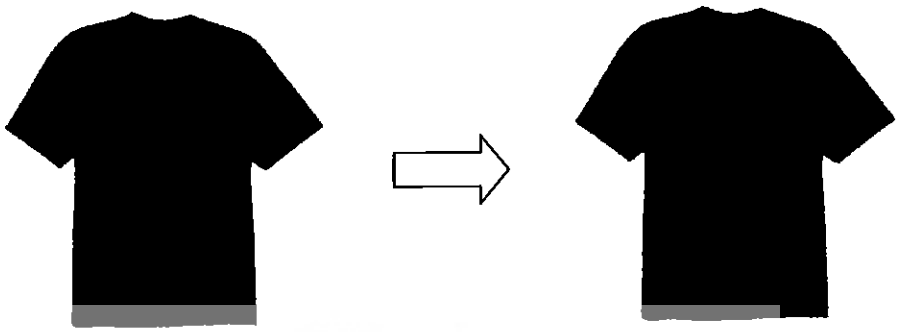
รูปที่ 3.5 ตัวอย่างภาพต้นแบบระดับสี RGB

ทำการเลือกภาพต้นแบบจากรูปที่ 3.5 ภาพต้นแบบจะอยู่ในระบบสี RGB และทำการเก็บค่าสีของรูปภาพ โดยพิจารณาที่ตำแหน่ง X = 64 , Y =64 เพื่อทำการคืนค่าสีให้รูปภาพหลังจากกระบวนการปรับปรุงภาพ

3.1.3.2 ภาพโทนสีเทา(Gray Level Image)

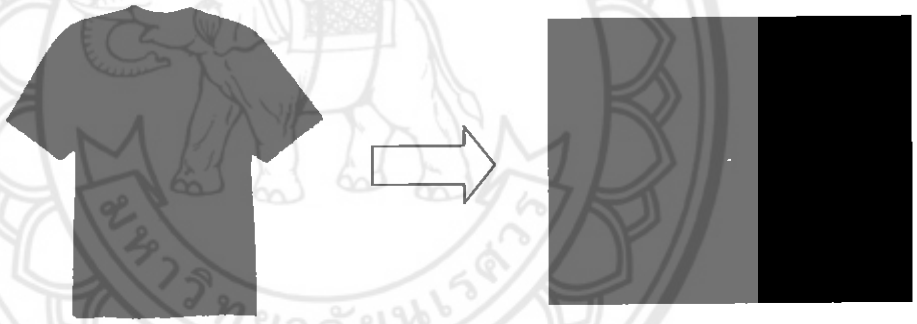
เป็นอัตราส่วนของโทนสีเทา ซึ่งมีการไล่ระดับความอ่อนแก่ ที่อยู่ระหว่างสีขาวกับสีดำ การสร้างภาพให้มีระดับสีต่างๆอย่างต่อเนื่องด้วยการใช้จุดสีที่มีขนาดต่างกัน หรือมีความหนาแน่นของจุดต่างกัน

ค่าในแต่ละพิกเซลของภาพโทนสีเทาคือค่าความเข้มของแสง ณ แต่ละตำแหน่งของ พิกเซล ซึ่งจะอยู่ในรูปของ ภาพโทนสีเทาค่าที่เป็นไปได้ของ ภาพโทนสีเทาจะขึ้นอยู่กับจำนวนบิตที่ใช้ ตัวอย่างเช่น 8 บิต จะมี ภาพโทนสีเทาทั้งหมด 256 ระดับ เพื่อลดแชนแนลของสีให้น้อยลง ประมวลผลได้เร็วขึ้น ดังแสดงรูปที่ 3.6



รูปที่3.6 ตัวอย่างการเปลี่ยนภาพสี RGB เป็นภาพระดับโทนสีเทา

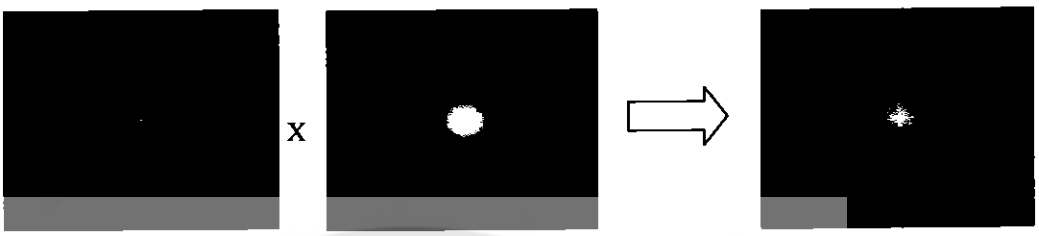
3.1.3.3 ปรับภาพในโดเมนความถี่ (Frequency Domain)



รูปที่ 3.7 ตัวอย่างการแปลงภาพให้อยู่ในโดเมนความถี่ (Frequency Domain)

การแปลงข้อมูลภาพในโดเมนตำแหน่งให้กลายเป็นข้อมูลภาพในโดเมนความถี่ เพื่อประมวลผลภาพ

3.1.3.4 ทำการปรับปรุงภาพด้วยตัวกรองเกาส์เซียน(Gaussian filter)



รูปที่3.8 ตัวอย่างภาพที่ผ่านตัวกรองเกาส์เซียนในโดเมนความถี่

ตัวกรองเกาส์เซียนเป็นตัวกรองที่มีลักษณะคล้ายระฆังคว่ำ ใช้สำหรับลดสัญญาณรบกวนและเบลอภาพ โดยการนำตัวกรองเกาส์เซียนไปทำการกรองกับภาพที่นำมาประมวลผล

3.1.3.5 ทำการแปลงภาพกลับตำแหน่งเดิม

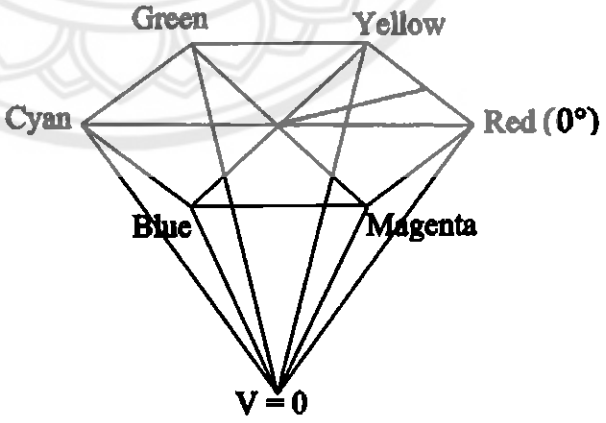
รูปที่3.9 ปรับภาพสู่โดเมนเดิมของภาพต้นแบบ

และเมื่อทำการประมวลผลภาพแล้ว จะใช้สมการแปลงผกผันเพื่อเปลี่ยนข้อมูลภาพให้กลับมาอยู่ในโดเมนตำแหน่งดังเดิมหลังจากผ่านประบวนการปรับปรุงภาพเพื่อลดสัญญาณรบกวนในช่วงความถี่ต่ำ

3.1.3.6 ปรับภาพให้เป็นภาพระดับสีRGB

รูปที่3.10 คืนค่าสีให้กับภาพในรูปที่ 3.9

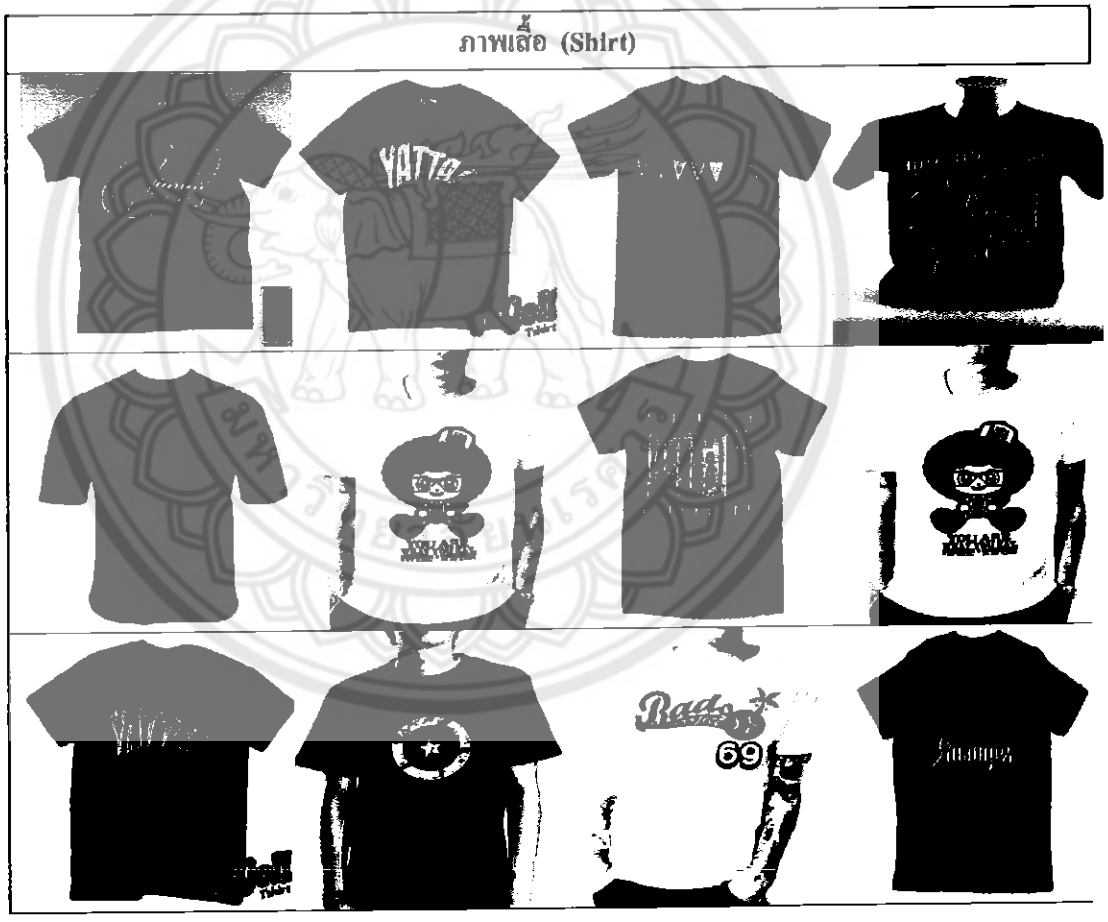
ทำการปรับภาพไปเป็นภาพสี RGB เพื่อทำกระบวนต่อไป โดยทำการเปลี่ยนระดับสีจาก RGB เป็น HSV ทำการเก็บค่า Hue เพื่อนำไปเปรียบเทียบ และใช้ค้นหาภาพที่มีความคล้ายคลึงกัน



รูปที่ 3.11 ปรับภาพให้เป็นภาพระดับสี HSV

3.1.4 เลือกภาพต้นแบบ

ในการเก็บภาพที่ใช้ในระบบงานได้ใช้การค้นหาจากเว็บไซต์ ปรับความละเอียดของภาพอยู่ที่ขนาด (255 x 255 พิกเซล) แล้วทำการบันทึกในแฟ้มข้อมูลที่เราสร้างไว้ทั้งหมด 1 กลุ่มภาพคือ เสื้อเป็นจำนวน 500 ภาพจะทำให้ได้ภาพทั้งหมดในระบบที่ใช้ในการทดลองรวม 500 ภาพ โดยภาพที่ได้จะนำมาใช้เป็นภาพตัวอย่างสำหรับทำการวิเคราะห์ภาพของระบบ ดังตัวอย่างภาพดังรูป



รูปที่ 3.12 ตัวอย่างภาพ ที่ใช้ในงานปริญญานิพนธ์

3.1.5 ทำการวิเคราะห์สี

ในการวิเคราะห์สี จะใช้มาตรฐานสีแบบ RGB ในการวิเคราะห์คุณสมบัติของภาพ โดยมาตรฐานสีแบบ RGB นี้จะมีการหาค่าสี 3 สี ได้แก่ สีแดง สีเขียว และสีน้ำเงิน ซึ่งสี 3 สีนี้ เมื่อนำมาประสมกันจะได้สีออกมาแตกต่างกันหลากหลายสี

รูปภาพแต่ละภาพก็คือการรวมตัวกันของเม็ดสีจำนวนมาก โดยรูปภาพที่แปลงเป็นไฟล์ดิจิตอลแล้วจะมีการมองรูปภาพออกเป็นพิกเซล โดยแต่ละรูปภาพก็มีขนาดแตกต่างกันออกไป

รูปที่มีขนาด 255x255 พิกเซลซึ่งการที่จะวิเคราะห์เม็ดสีแต่ละเม็ดนี้ว่ามีค่าสีแดง สีเขียว และสีน้ำเงินอย่างละเท่าไรทำได้โดยการพล็อตกราฟแสดงความถี่ความเข้มสี โดยกราฟแสดงความถี่ความเข้มสีของรูปภาพแต่ละรูปนั้น จะมีการแยกสี แดง เขียว และน้ำเงินออกมาจึงทำให้ได้กราฟออกมา 3 กราฟ ซึ่งการกราฟแต่ละกราฟก็คือกราฟความถี่ของความเข้มสีแต่ละพิกเซล

รูปที่ 3.13 การแสดงค่า ฮิสโตแกรมเป็นกราฟ

หลังจากนั้นทำการปรับภาพภาพเป็นระบบสี HSV (Hue Saturation Value) เพื่อหาค่าฮิว (Hue) เพื่อง่ายต่อการนำมาเปรียบเทียบจึงเก็บค่าฮิว (Hue) ของแต่ละภาพเอาไว้ ตัวอย่างเช่น ทำการเก็บค่าฮิว(Hue)ของภาพ ค่าฮิว(Hue) คือค่าสีของสีหลัก(สีแดง สีเขียวและสีน้ำเงิน)จะอยู่ระหว่าง 0 และ 360 ซึ่งถ้าฮิว Hue มีค่าเท่ากับ 0 จะแทนสีแดงและเมื่อ Hue มีค่าเพิ่มขึ้นเรื่อยๆ สีก็จะ

เปลี่ยนแปลงไปตามสเปกตรัมของสีจนถึง 360 จึงจะกลับมาเป็นสีแดงอีกครั้ง ซึ่งสามารถแทนให้อยู่ในรูปขององศาได้ ดังนี้คือ สีแดงเท่ากับ 0 องศา สีเขียวเท่ากับ 120 องศา สีน้ำเงินเท่ากับ 240 องศา ดังนั้นจึงต้องทำการเก็บค่าความถี่ของสีแต่ละระดับลงในแฟ้มข้อมูล ตัวเลขทั้งหมด 360 ค่า

3.1.6 เพิ่มข้อมูลของภาพลงในแฟ้มข้อมูล

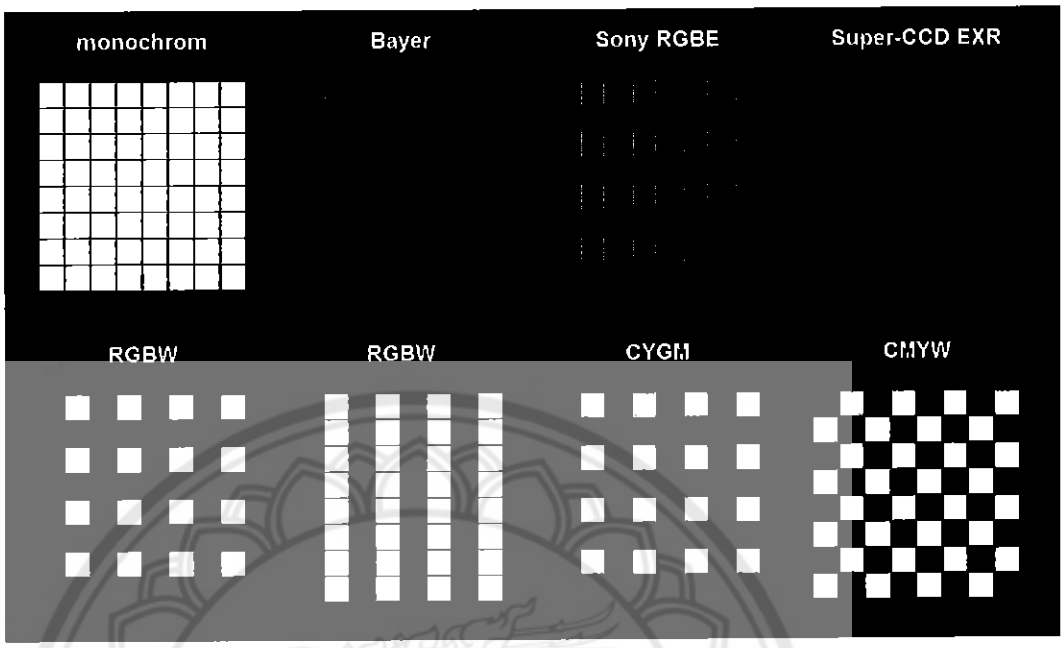
นำข้อมูลของภาพต้นแบบที่ผ่านการวิเคราะห์ให้เป็นเวคเตอร์แทนสีและไปเก็บในแฟ้มข้อมูลรูปภาพในที่นี้คือการสร้างแฟ้มข้อมูล ในการเก็บข้อมูลรูปภาพต่างๆที่ต้องการแล้วนำรูปภาพที่ต้องการเก็บนั้นเก็บไว้ในแฟ้มข้อมูล นั้นได้และเมื่อต้องการทำการค้นหารูปภาพที่ต้องการก็สามารถเลือกแฟ้มข้อมูล ในการค้นหาได้จากโปรแกรม เพื่อที่จะใช้ในการเปรียบเทียบหาภาพที่ต้องการได้

3.1.7 เปรียบเทียบค่าในแฟ้มข้อมูล

จะนำค่าที่ได้จากการวิเคราะห์จากภาพต้นแบบ ไปเปรียบเทียบกับค่าในแฟ้มข้อมูลเพื่อหาภาพที่คล้ายกับภาพต้นแบบ ซึ่งมีวิธีการหาความคล้ายคลึงของภาพ ดังนี้

3.1.7.1 พิกเซล

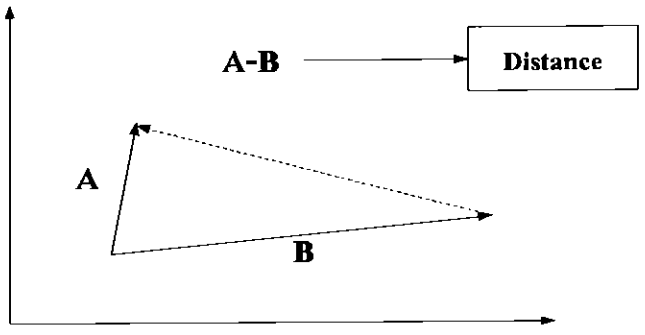
พิกเซล คือ ส่วนหนึ่งที่เล็กที่สุดของภาพดิจิตอล เป็นส่วนของการแสดงผลภาพบนสื่อดิจิตอลแสดงโดยค่าดิจิตอลพิกเซลเป็นค่าแสดงผล และสามารถใช้เป็นหน่วยของการวัด(ตัวชี้วัด)ความละเอียดของภาพเช่น640 x 480 พิกเซล หมายความว่า มีความละเอียด 307,200 พิกเซล ใน1 พิกเซล จะมีสีใดสีหนึ่งเพียงสีเดียวเท่านั้นจะมีสีอื่นไม่ได้ เนื่องจากว่าเป็นส่วนที่เล็กที่สุดของการแสดงผล ดังนั้นรูปที่เราเห็นจึงประกอบด้วยหลายๆ พิกเซล มาต่อเรียงกันจนเกิดภาพนั่นเองรูปภาพ 1 รูป ภายในจะประกอบไปด้วย พิกเซล จำนวนมากเรียงต่อกันเป็นอาเรย์2 มิติ ตามขนาดของรูปนั้นๆ เช่น รูปภาพขนาด 255 x 255 จะเห็นได้ว่าที่ตำแหน่ง 0,0 (มุมซ้ายบนสุด) จะมีค่าพิกเซล = 0 ซึ่งค่า พิกเซล = 0 นั้นจะหมายถึงสีดำหากต้องการเข้าไปเปลี่ยนข้อมูลรูปภาพในระดับพิกเซล นั้นสามารถทำได้



รูปที่ 3.14 การแสดงค่าพิกเซล

3.7.1.2 การหาความคล้ายคลึงของภาพ

นำภาพต้นแบบผ่านอัลกอริทึมเช่นเดียวกับการเพิ่มข้อมูลลงในแฟ้มข้อมูลเสร็จแล้วนำข้อมูลนั้น มาเปรียบเทียบกับข้อมูลในแฟ้มข้อมูลโดยการคำนวณหานั้นจะทำการหาค่าระยะห่าง (Distance) ระหว่างภาพต้นแบบกับภาพในแฟ้มข้อมูล แล้วนำมาเปรียบเทียบกับข้อมูลของภาพทุก ๆ ภาพในแฟ้มข้อมูล จากนั้นนำค่าระยะห่างที่น้อยที่สุดมาแสดงผล โดยค่า ฮิสโตแกรม (Histogram) ที่ถูกเก็บอยู่ในรูปเวคเตอร์จะทำการเปรียบเทียบได้ดังนี้



รูปที่ 3.14 ระยะห่าง (Distance)

จากสูตรการหาค่า Vector

$$D(X,Y) = \sqrt{(X_1 - Y_1)^2 + (X_2 - Y_2)^2 \ldots \ldots} \quad (3)$$

เมื่อทำการเปรียบเทียบรูปภาพทุกรูปภาพในแฟ้มข้อมูลแล้ว จะได้ค่าระยะห่าง (Distance) ของแต่ละรูปภาพออกมา โดยค่าระยะห่างที่ได้ออกมามีความแตกต่างกันออกไป จึงทำให้การจัดเรียงลำดับจากน้อยไปหามาก ซึ่งค่าระยะห่างที่มีค่าน้อยที่สุดคือค่าของรูปภาพในแฟ้มข้อมูลที่มีความใกล้เคียงมากที่สุด โดยโปรแกรมจะทำการนำรูปภาพที่มีค่าระยะห่างน้อยที่สุดมาแสดงผล

3.1.8 แสดงภาพที่ใกล้เคียงกับภาพต้นแบบ

เมื่อภาพใกล้เคียงกับภาพต้นแบบ จะนำภาพมาแสดงในโปรแกรมตามลำดับความคล้ายคลึงจากน้อยไปหามาก

รูปที่3.15 ตัวอย่างการแสดงผลของโปรแกรม

3.1.9 ระบบป้อนกลับ (Relevance Feedback)

เป็นขั้นตอนสุดท้ายในการค้นหาภาพคือเมื่อโปรแกรมได้หาค่าสี ของภาพต้นแบบแล้ว อินพุต ตัวต่อไปของโปรแกรมคือค่าสีที่ผู้ใช้ได้เลือกจากภาพที่โปรแกรมได้แสดงขึ้นมาโดยให้ผู้ใช้ทำการป้อนกลับ (Relevance Feedback) ลักษณะของรูปภาพที่ผู้ใช้ต้องการเพื่อให้โปรแกรมรู้

ถึงความต้องการของผู้ใช้ ซึ่งหลักการในการทำระบบป้อนกลับ คือ การนำภาพที่ผู้ใช้เลือกมาคำนวณให้ได้ค่าเรียงลำดับ (Query) ตัวใหม่ แล้วนำค่านี้ไปเปรียบเทียบเวกเตอร์ในแฟ้มข้อมูลโดยการคำนวณ

3.1.10 ได้ภาพตรงกับความต้องการมากขึ้น

หลังจากที่ได้ทำการป้อนกลับ (Relevance Feedback) แล้ว เราจะได้ภาพที่ตรงกับความต้องการมากขึ้น

ในตัวโปรแกรมจะมีส่วนหลักๆอยู่ 5 ส่วน คือ

1. ส่วนที่ใช้เลือกรูปภาพต้นแบบ โดยปุ่ม "Browse" จะเป็นการเลือกภาพที่เราต้องการนำมาเป็นภาพต้นแบบ เพื่อใช้หาภาพที่คล้ายๆกันนี้

2. ส่วนที่ใช้ในการค้นหาภาพ ซึ่งเราสามารถเลือกได้ว่าจะเน้นภาพที่คล้ายกับสีของภาพต้นแบบ ในการค้นหา ซึ่งปุ่ม "Search" จะมีอัลกอริทึมคำนวณเพื่อวิเคราะห์สีและเช่นเดียวกับการเก็บภาพลงในแฟ้มข้อมูล โดยเมื่อคำนวณได้ค่าเป็นเวคเตอร์แล้ว จะนำไปเปรียบเทียบกับภาพในแฟ้มข้อมูลแล้วนำภาพที่ใกล้เคียงกับภาพต้นแบบมาแสดงผล

3. ส่วนนี้จะใช้ในการทำการป้อนกลับ (Relevance Feedback) เพื่อให้โปรแกรมเรียงลำดับของการคำนวณแต่ละภาพ และนำไปเปรียบเทียบกับแฟ้มข้อมูล ซึ่งจะได้ภาพที่ใกล้เคียงกับภาพต้นแบบมากขึ้น

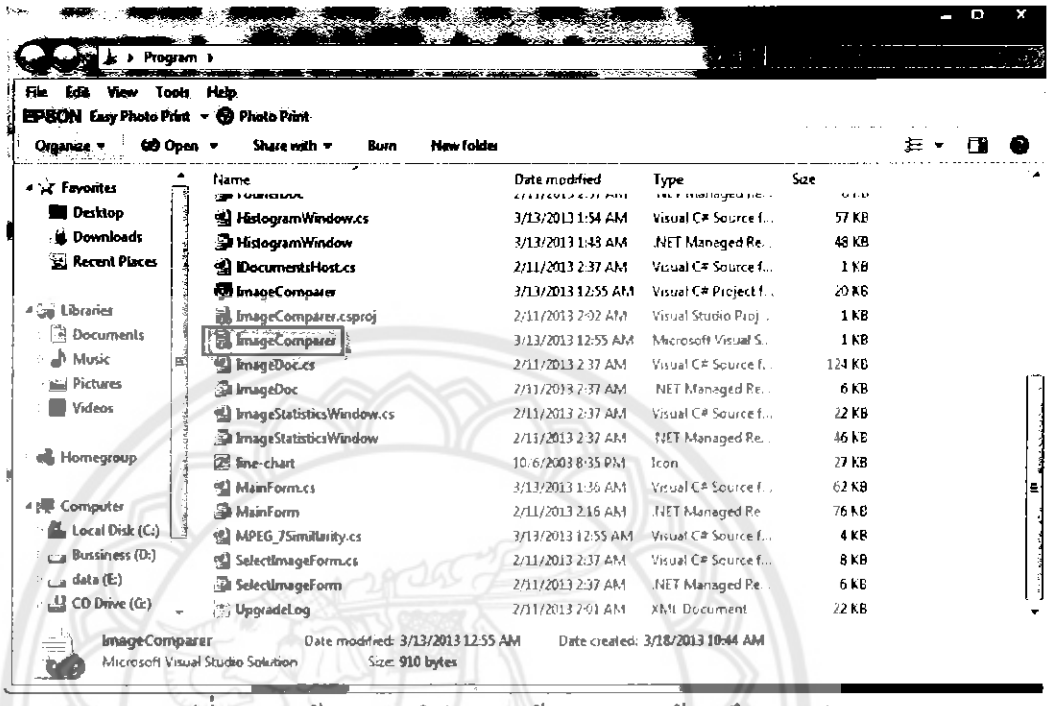
4. ส่วนที่ใช้ในการเพิ่มข้อมูลลงในแฟ้มข้อมูล โดยปุ่ม "Load Images" จะมีไว้สำหรับเพิ่มข้อมูลรูปภาพลงในแฟ้มข้อมูล ซึ่งนำรูปที่ Browse เข้ามา ผ่านอัลกอริทึมเพื่อวิเคราะห์สีออกมาให้อยู่ในรูปเวคเตอร์แล้วเก็บเวคเตอร์นั้นลงในแฟ้มข้อมูล

5. ส่วนที่ใช้แสดงผลของการหาภาพ ซึ่งจะนำภาพที่คล้ายคลึงกับภาพต้นแบบ

3.2 การออกแบบหน้าต่างโปรแกรมส่วนติดต่อกับผู้ใช้

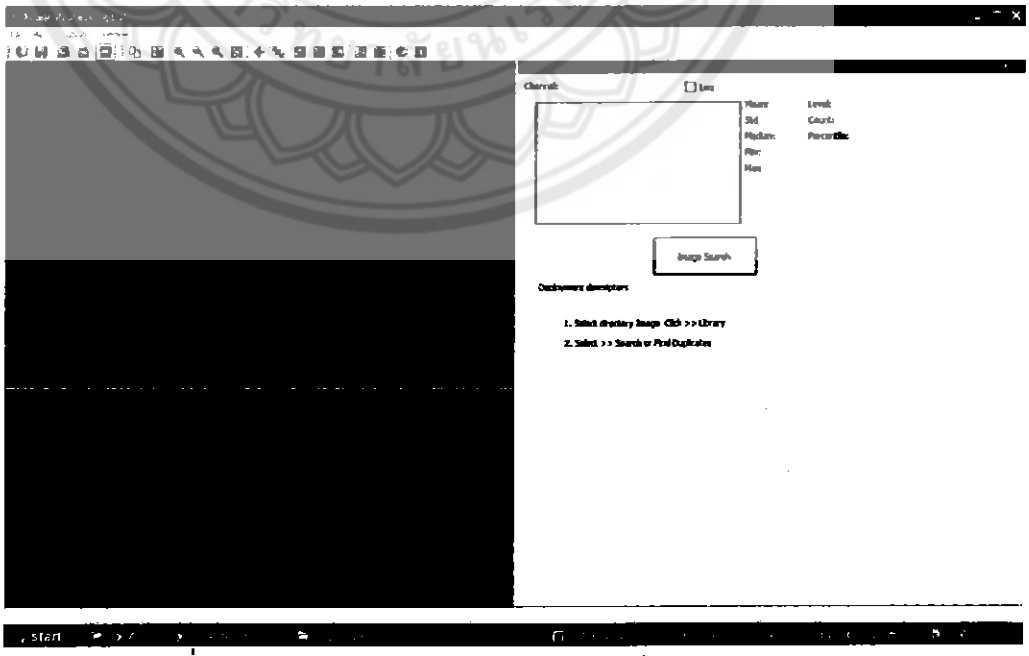
3.2.1 ติดตั้งโปรแกรมค้นหาความคล้ายคลึงของรูปภาพ

1. ผู้ใช้จะต้องนำไฟล์โปรแกรมค้นหาความคล้ายคลึงของรูปภาพไว้ในเครื่องคอมพิวเตอร์หลังจากนั้นทำการเข้าไปที่แฟ้มข้อมูลของโปรแกรมแล้วทำการเปิดโปรแกรมด้วยการดับเบิ้ลคลิกที่ไอคอน ImageComparer.sln



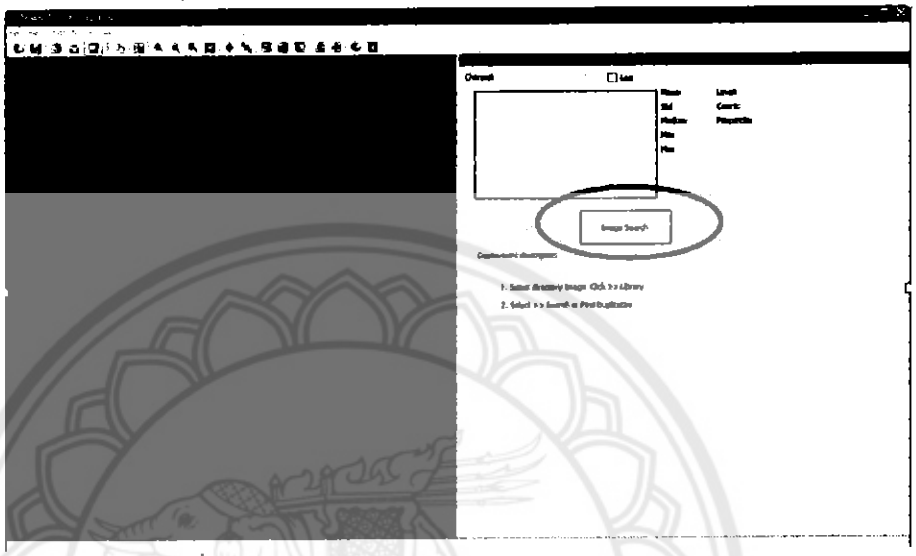
รูปที่ 3.16 หน้าการRunโปรแกรมค้นหาความคล้ายคลึงของรูปภาพ

2. จะแสดงหน้าต่างของโปรแกรมค้นหาความคล้ายคลึงของรูปภาพ



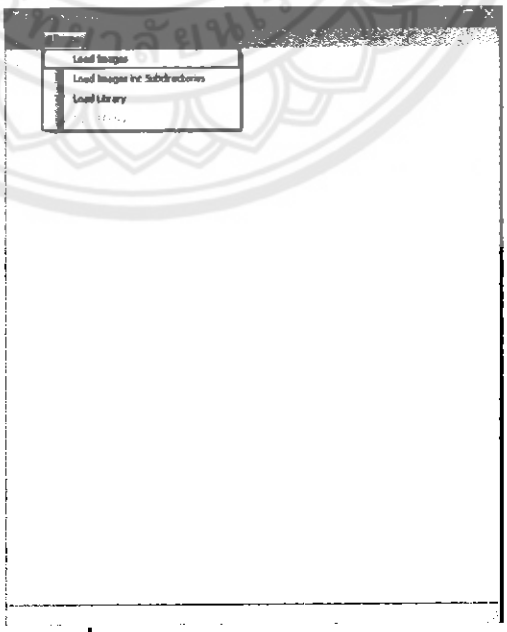
รูปที่ 3.17 หน้าจอแอพพลิเคชันค้นหาความคล้ายคลึงของรูปภาพ

3. เมื่อผู้ใช้ต้องการเริ่มต้นการใช้งานโปรแกรมค้นหาความคล้ายคลึงของรูปภาพ ผู้ใช้สามารถกดเลือกที่ปุ่ม Image Search



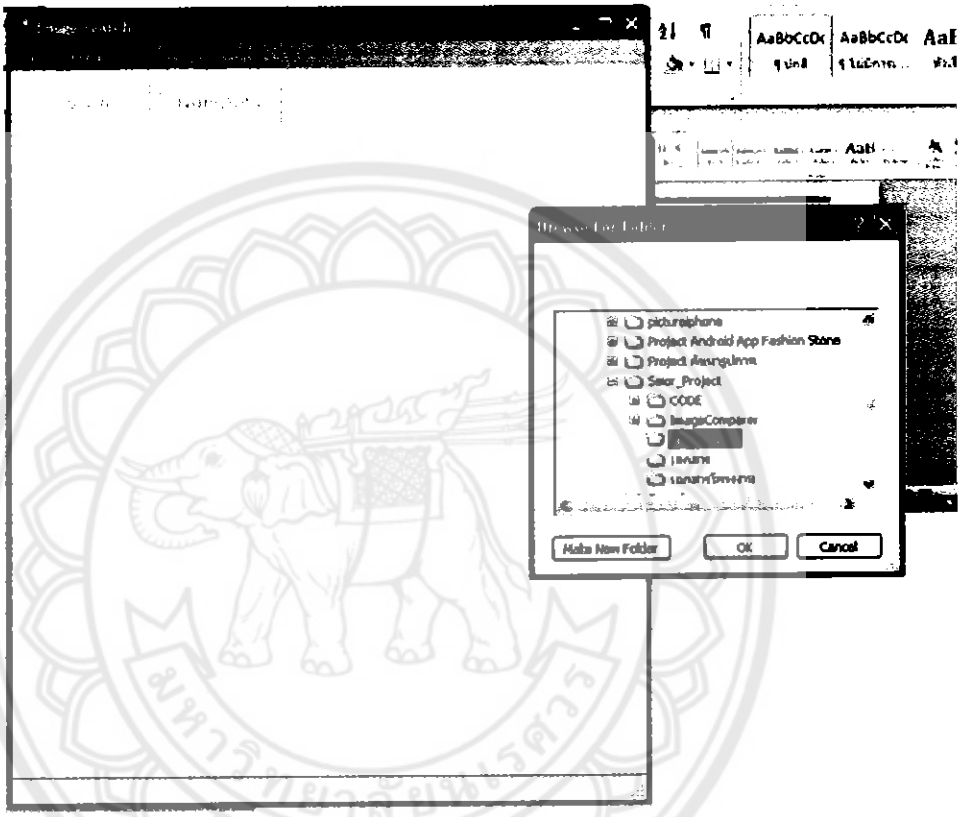
รูปที่ 3.18 แสดงหน้าการค้นหารูปภาพปุ่ม Image Search

4. หลังจากนั้นโปรแกรมจะทำการแสดงหน้าต่าง Image Search ขึ้นมาให้ทำการคลิกที่เมนู Library แล้วเลือก Load Image เพื่อให้ระบบทำการโหลดภาพที่มีอยู่ในแฟ้มข้อมูล



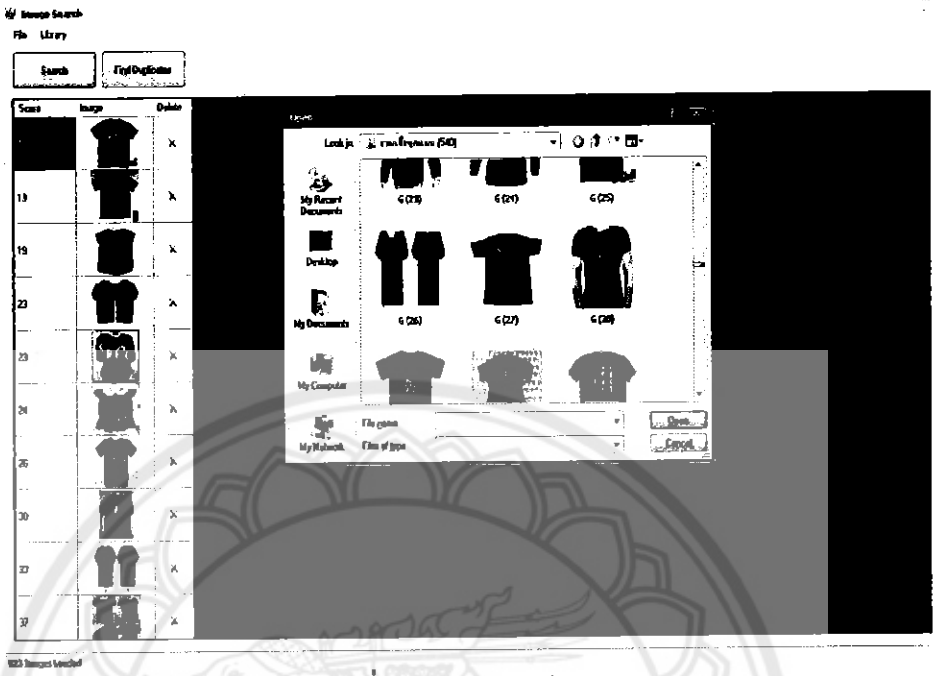
รูปที่ 3.19 แสดงหน้าต่าง Image Search

5. เลือก แฟ้มข้อมูล รูปภาพที่ใช้เก็บข้อมูลภาพนั้นๆ ระบบจึงจะสามารถทำการประมวลผลออกมาว่ามีรูปภาพที่เก็บไว้ทั้งหมดเท่าใด



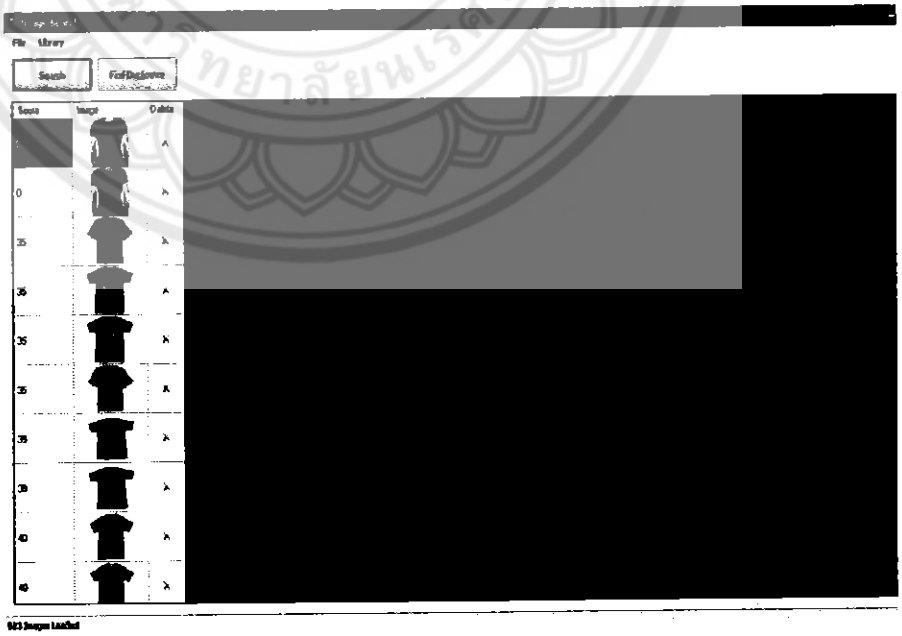
รูปที่ 3.20 แสดงหน้าการเลือกแฟ้มข้อมูล ที่ใช้เก็บข้อมูลรูปภาพ

6. ระบบจะทำการแสดงหน้าต่างข้อมูลภาพต่างๆเป็นหน้าที่ใช้สำหรับทำการค้นหารูปภาพต่อไปได้ด้วยการเลือกภาพที่ต้องการค้นหาแล้วทำการกดที่ปุ่ม Search โดยผู้ใช้สามารถทำการค้นหารูปภาพได้จากแฟ้มข้อมูลรูปภาพในแฟ้มข้อมูล ที่ได้ทำการตั้งค่าไว้ในตอนแรกเท่านั้น



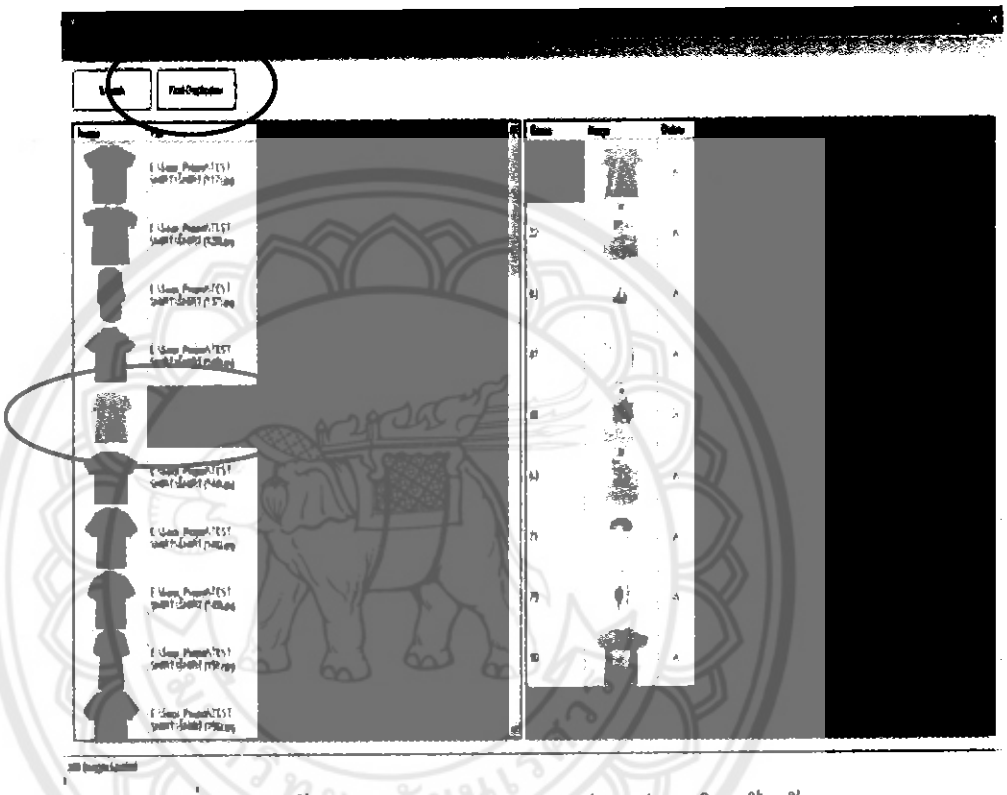
รูปที่3.21 แสดงการค้นหารูปภาพ

7. ระบบจะทำการค้นหารูปภาพที่ผู้ใช้ได้ทำการเลือกรูปภาพที่ต้องการค้นหาโดยจะทำการแสดงผลออกมาดังรูปที่ 3.22



รูปที่3.22 หน้าต่างการแสดงผลการค้นหารูปภาพที่ต้องการ

8. ระบบจะทำการแสดงหน้าต่างข้อมูลภาพต่างๆเป็นหน้าที่ใช้สำหรับทำการค้นหารูปภาพแล้วทำการกดที่ปุ่มFind Duplicates โปรแกรมจะทำการแสดงภาพที่มีอยู่ในแฟ้มข้อมูล



รูปที่ 3.23 หน้าต่างการแสดงผลการตัวอย่างรูปภาพในแฟ้มข้อมูล

ขั้นตอนการทำงานของโปรแกรมค้นหาความคล้ายคลึงของรูปภาพนั้นยังมีข้อจำกัดอยู่บ้างเรื่องของการค้นหารูปภาพเมื่อระบบต้องทำการค้นหารูปภาพที่มีลายละเอียดค่อนข้างมากจะทำให้ระบบประมวลผลในการค้นหานั้นเกิดการผิดพลาดได้ หรือการค้นหารูปภาพที่เลือกนั้นแสดงผลลัพธ์ออกมาไม่ตรงตามความต้องการของผู้ใช้ ดังนั้นในบทต่อไปจะได้กล่าวถึงการทดลองโปรแกรมในเรื่องของความถูกต้องของการค้นหารูปภาพของโปรแกรมค้นหาความคล้ายคลึงของรูปภาพเพื่อให้ผู้ใช้ได้เข้าใจในกรณีที่ระบบไม่สามารถค้นหารูปภาพให้ตรงกับความต้องการของผู้ใช้ได้

บทที่4

ผลการทดลอง

จากการที่ได้ทำการทดลองใช้งานโปรแกรมค้นหารูปภาพจากแฟ้มข้อมูลแล้ว เราจะทำการทดสอบผลการทดลองโดยจะใช้หลักการเปรียบเทียบผลของค่าความถูกต้อง (Precision) โดยค่าความถูกต้องเป็นค่าที่ได้จากการคำนวณดังต่อไปนี้

% ความถูกต้อง = (จำนวนภาพที่ใกล้เคียง / จำนวนภาพทั้งหมด) × 100

เช่น % ความถูกต้อง $= \left(\frac{80}{100}\right) \times 100 = 80\%$

หลักการเปรียบเทียบผลของค่าความผิดพลาด (Trial and error)

% ความผิดพลาด = (เปอร์เซ็นต์- เปอร์เซ็นต์ความถูกต้อง)

เช่น % ความผิดพลาด $= 100 - 80 = 20\ \%$

แล้วนำค่าความถูกต้อง (Precision) มารวมกันแล้วหาค่าเฉลี่ย ก็จะได้เป็นค่าความถูกต้องของการค้นหาและนำค่าความผิดพลาด(Error) มารวมกันแล้วหาค่าเฉลี่ย เพื่อหาความผิดพลาดการค้นหา ซึ่งเมื่อนำไปเปรียบเทียบกับวิธีการอื่น ๆ ก็จะเห็นความแตกต่างของการค้นหา

โดยเราจะทำการเปรียบเทียบการค้นหาภาพดังนี้

1.การค้นหาภาพโดยวิเคราะห์ด้วยสี

ซึ่งในการเปรียบเทียบแต่ละครั้งจะเปรียบเทียบกับภาพที่เก็บไว้ในแฟ้มข้อมูลรูปภาพ ซึ่งมีข้อมูลภาพทั้งหมดกว่า 500 ภาพโดยทำการแบ่งกลุ่มตัวอย่างดังนี้ เสื้อสีเขียว 100 ตัว เสื้อสีน้ำเงิน 100 ตัว เสื้อสีดำ 100 ตัว เสื้อสีแดง 100 ตัว และ เสื้อสีขาว 100 ตัว

4.1 ขั้นตอนการทดลองค้นหาภาพโดยวิเคราะห์ด้วยสี

4.1.1 นำไฟล์รูปภาพตัวอย่างที่ต้องการค้นหา โดยวิเคราะห์แต่สีเท่านั้น

4.1.2 โปรแกรมจะทำการเปรียบเทียบค่าเวคเตอร์ระหว่างไฟล์รูปภาพในแฟ้มข้อมูล

4.1.3 เมื่อเปรียบเทียบค่า Distance d1 < d2 < d3 < d4 < d5 <d6 < d7 < d8 < d9 <...< d16

4.1.4 โชว์ผลการค้นหาทั้งหมดที่ถูกต้อง โดยเรียงลำดับค่าระยะห่าง(Distance)

4.1.5 ทำการเลือกภาพที่เหมือน โดยการทดสอบ ในกรณีนี้เราจะพิจารณาองค์ประกอบของภาพว่ามีส่วนที่เกี่ยวข้องภายในภาพหรือไม่

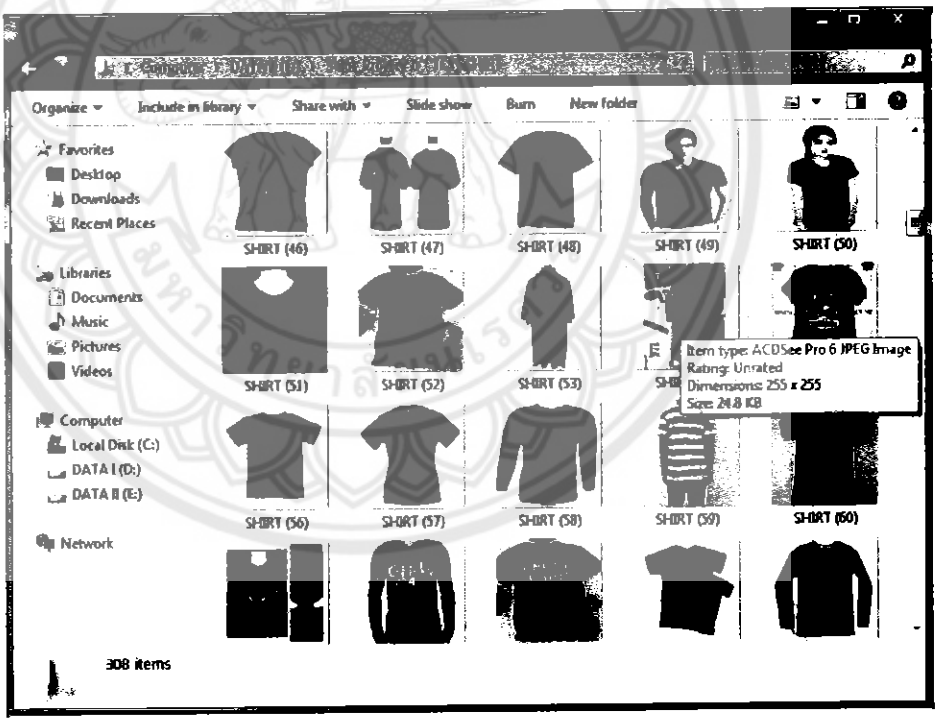
4.2 ผลการทดลองค้นหาภาพโดยวิธีการวิเคราะห์ด้วยสี

ผู้ศึกษานำเสนอผลการทดลอง แบ่งเป็นกรณีดังนี้

4.2.1 ผลการทดลองการค้นหารูปภาพจากเสื้อสีแดงพื้นและมีลาย / เสื้อสีเขียวพื้นและมีลาย / สีน้ำเงินพื้นและมีลาย/ สีดำพื้นและมีลาย/ สีขาวพื้นและมีลาย

4.2.2 ผลการทดลองการค้นหารูปภาพจากเสื้อลายขวาง / เสื้อสีสลับ / เสื้อสีผสม

4.2.3 ผลการทดลองที่ระบบแสดงตัวอย่างรูปภาพภายในแฟ้มข้อมูล



รูปที่ 4.1 ตัวอย่างเสื้อสีแดงที่มีอยู่ในแฟ้มข้อมูล

รูปที่ 4.2 ผลการค้นหารูปภาพเสื้อสีแดงพื้นและมีลาย โดยการวิเคราะห์จากสี

เมื่อทำการทดลองเสร็จสิ้นจะทำการคำนวณค่าความถูกต้อง (Precision) จากการทดลองการการค้นหาด้วยเสื้อสีแดงพื้นและมีลาย จะได้ดังนี้

จากตัวอย่างในรูป จะได้ภาพที่คล้ายคลึง 96 ภาพ นำมาคำนวณหาค่าความถูกต้องจะได้ว่า

$$\%Precision = \left(\frac{96}{100}\right) \times 100 = 96\%$$

$$\%\ \text{Error} = (100 - 96) = 4\%$$

ฉะนั้น เราจะนำค่าความถูกต้อง (Precision) และค่าความผิดพลาด ในแต่ละไฟล์รูปภาพมาบันทึกผลการทดลองได้ดังนี้

ตารางที่ 4.1 บันทึกค่าการทดลองการค้นหาด้วยสีแดงพื้นและมีลาย

ครั้งที่	รูปภาพ	%Precision	%Error
1		92	8
2		74	26
3		96	4
4		92	8
5		96	4
6		92	8
7		92	8
8		90	10
9		96	4
10		83	17

จากการทดลองหาภาพทั้งหมด 10 ภาพ ได้ค่าความถูกต้อง (Precision) อยู่ในช่วง 74 -96 ซึ่งสรุปได้ว่าค่าเฉลี่ยของค่าความถูกต้องสำหรับการหาด้วยสีแดงพื้นและมีลายคือ 90.3%

เมื่อทำการทดลองเสร็จสิ้นจะทำการคำนวณค่าความถูกต้อง(Precision) จากการทดลองการค้นหาด้วยเสื้อสีเขียวพื้นและมีลาย จะได้ดังนี้

จากตัวอย่างในรูป จะได้ภาพที่คล้ายคลึง 70 ภาพ นำมาคำนวณหาค่าความถูกต้องจะได้ว่า

$$\%Precision = \left(\frac{70}{100}\right) \times 100 = 70\%$$

$$\%\ \text{Error} = (100 - 70) = 30\ \%$$

ฉะนั้น เราจะนำค่าความถูกต้อง (Precision) และค่าความผิดพลาด ในแต่ละไฟล์รูปภาพมาบันทึกผลการทดลองได้ดังนี้

ตารางที่ 4.2 บันทึกค่าการทดลองการค้นหาด้วยเสื้อสีเขียวพื้นและมีลาย

ครั้งที่	รูปภาพ	%Precision	%Error
1		60	40
2		70	30
3		45	55
4		40	60
5		35	65
6		45	55

7		50	50
8		40	60
9		45	55
10		50	50

จากการทดลองหาภาพทั้งหมด 10 ภาพ ได้ค่าความถูกต้อง (Precision) อยู่ในช่วง 35-70 ซึ่งสรุปได้ว่าค่าเฉลี่ยของค่าความถูกต้องสำหรับการหาด้วยเสื้อสีเขียวพื้นและมีลายคือ 48%

เมื่อทำการทดลองเสร็จสิ้นจะทำการคำนวณค่าความถูกต้อง(Precision) จากการทดลองการค้นหาด้วยเสื้อสีน้ำเงินพื้นและมีลายจะได้ดังนี้

จากตัวอย่างในรูป จะได้ภาพที่คล้ายคลึง 68 ภาพ นำมาคำนวณหาค่าความถูกต้องจะได้ว่า

$$\%Precision = \left(\frac{68}{100}\right) \times 100 = 68\ \%$$

$$\%\ \text{Error} = (100 - 68) = 32\ \%$$

ฉะนั้น เราจะนำค่าความถูกต้อง (Precision) และค่าความผิดพลาด ในแต่ละไฟล์รูปภาพมาบันทึกผลการทดลองได้ดังนี้

ตารางที่ 4.3 บันทึกค่าการทดลองการค้นหาด้วยสีน้ำเงินพื้นและมีลาย

ครั้งที่	รูปภาพ	%Precision	%Error
1		65	35
2		62	38
3		65	35
4		63	37
5		29	71
6		64	36
7		41	59
8		57	43
9		50	50
10		68	32

จากการทดลองหาภาพทั้งหมด 10 ภาพ ได้ค่าความถูกต้อง (Precision) อยู่ในช่วง 29-68 ซึ่งสรุปได้ว่าค่าเฉลี่ยของค่าความถูกต้องสำหรับการหาด้วยสีน้ำเงินพื้นและมีลายคือ 56.4%

เมื่อทำการทดลองเสร็จสิ้นจะทำการคำนวณค่าความถูกต้อง(Precision)จากการทดลองการค้นหาด้วยสีดำพื้นและมีลายจะได้ดังนี้

จากตัวอย่างในรูป จะได้ภาพที่คล้ายคลึง 96 ภาพ นำมาคำนวณหาค่าความถูกต้องจะได้ว่า

$$\%Precision = \left(\frac{96}{100}\right) \times 100 = 96\,\%$$

$$\%\ \text{Error} = (\ 100 - 96\ ) = 4\,\%$$

ฉะนั้น เราจะนำค่าความถูกต้อง (Precision) และค่าความผิดพลาด ในแต่ละไฟล์รูปภาพมาบันทึกผลการทดลองได้ดังนี้

ตารางที่ 4.4 บันทึกค่าการทดลองการค้นหาด้วยสีดำพื้นและมีลาย

ครั้งที่	รูปภาพ	%Precision	%Error
1		96	4
2		90	10
3		80	20
4		88	12
5		88	12
6		90	10
7		93	7

8		80	20
9		87	13
10		71	29

จากการทดลองหาภาพทั้งหมด 10 ภาพ ได้ค่าความถูกต้อง (Precision) อยู่ในช่วง 71-96 ซึ่งสรุปได้ว่าค่าเฉลี่ยของค่าความถูกต้องสำหรับการหาด้วยสีดำพื้นและมีลาย คือ 86.3%

เมื่อทำการทดลองเสร็จสิ้นจะทำการคำนวณค่าความถูกต้อง(Precision) จากการทดลองการค้นหาด้วยสีสีขาวพื้นและมีลายจะได้ดังนี้

จากตัวอย่างในรูป จะได้ภาพที่คล้ายคลึง 88 ภาพ นำมาคำนวณหาค่าความถูกต้องจะได้ว่า

$$\%Precision = \left(\frac{88}{100}\right) \times 100 = 88\,\%$$

$$\%\ \text{Error} = (\ 100 - 88\ ) = 12\,\%$$

ฉะนั้น เราจะนำค่าความถูกต้อง (Precision) และค่าความผิดพลาด ในแต่ละไฟล์รูปภาพมาบันทึกผลการทดลองได้ดังนี้

ตารางที่ 4.5 บันทึกค่าการทดลองการค้นหาด้วยสีขาวพื้นและมีลาย

ครั้งที่	รูปภาพ	%Precision	%Error
1		72	28
2		83	17
3		75	25
4		74	26
5		85	15
6		85	15
7		88	12
8		86	14
9		83	17
10		87	13

จากการทดลองหาภาพทั้งหมด 10 ภาพ ได้ค่าความถูกต้อง (Precision) อยู่ในช่วง72-88 ซึ่งสรุปได้ว่าค่าเฉลี่ยของค่าความถูกต้องสำหรับการหาด้วยสีขาวพื้นและมีลายคือ 81.8%

เมื่อทำการทดลองเสร็จสิ้นจะทำการคำนวณค่าความถูกต้อง (Precision) จากการทดลองการค้นหาด้วยเสื้อลายขวางหรือสีสลับ หรือสีผสม จะได้ดังนี้

จากตัวอย่างในรูป จะได้ภาพที่คล้ายคลึง 6 ภาพ นำมาคำนวณหาค่าความถูกต้องจะได้ว่า

$$\%Precision = \left(\frac{6}{100}\right) \times 100 = 6\ \%$$

$$\%\ \text{Error} = (\ 100 - 6\ )\ =\ 94\ \%$$

ฉะนั้น เราจะนำค่าความถูกต้อง (Precision) และค่าความผิดพลาด ในแต่ละไฟล์รูปภาพมาบันทึกผลการทดลองได้ดังนี้

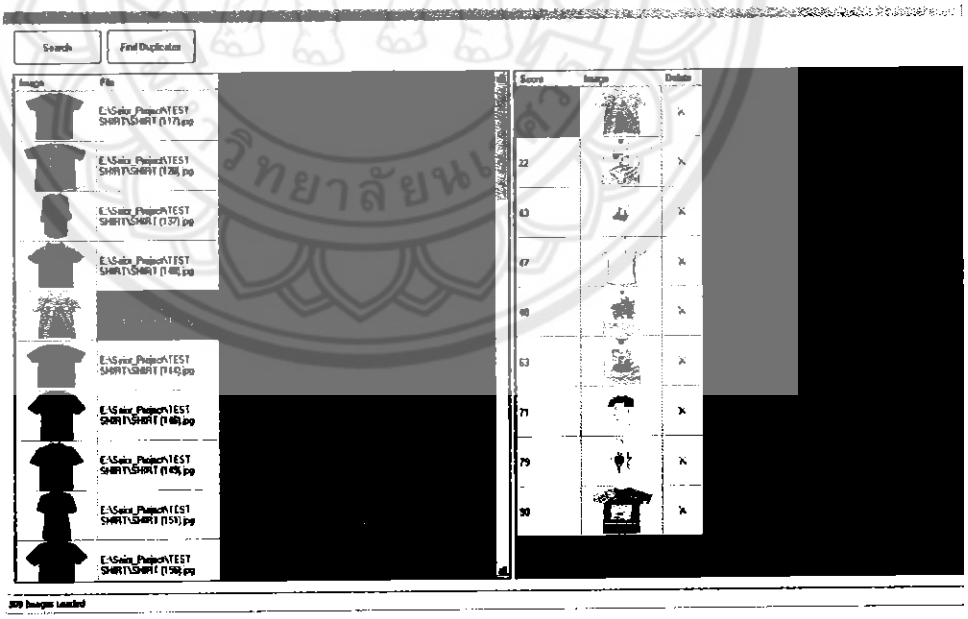
ตารางที่ 4.6 บันทึกค่าการทดลองการค้นหาจากเสื้อลายขวางหรือสีสลับ หรือสีผสม

ครั้งที่	รูปภาพ	%Precision	%Error
1		4	96
2		3	97
3		3	97
4		4	96
5		3	97
6		5	95

7		6	94
8		4	96
9		4	96
10		2	98

จากการทดลองหาภาพทั้งหมด 10 ภาพ ได้ค่าความถูกต้อง (Precision) อยู่ในช่วง 2 -6 ซึ่ง สรุปได้ว่าค่าเฉลี่ยของค่าความถูกต้องสำหรับการหาด้วยสีขาวพื้นและมีลายคือ 81.8%

4.3 กรณีที่ระบบแสดงตัวอย่างรูปภาพภายในแฟ้มข้อมูล



รูปที่ 4.3 การแสดงตัวอย่างรูปภาพภายในแฟ้มข้อมูล

จากผลการแสดงผลของระบบในการแสดงตัวอย่างภาพภายในแฟ้มข้อมูลโดยระบบจะทำการแสดงรูปภาพที่ทำการค้นหาออกมาเป็นตัวอย่างภาพที่มีความใกล้เคียงกับภาพที่ทำการค้นได้ดังภาพที่ 4.3

4.4 สรุปผลการทดลอง

ในบทนี้ได้ทำการทดลองโปรแกรมในเรื่องความถูกต้องของการแก้ปัญหา และการทดลองการค้นหาภาพ รวมถึงกรณีการค้นหาภาพในลักษณะต่างๆ ที่สามารถประมวลผลภาพได้และไม่สามารถประมวลภาพได้

ผลการตรวจค้นเสื้อระบบไม่สามารถตรวจค้นพบเสื้อตามความต้องการได้เนื่องจาก ลายของเสื้อมีลายซึ่งยากต่อการวิเคราะห์ ทำให้ระบบเกิดการค้นหาที่ไม่ตรงตามความต้องการทำให้ไม่สามารถตรวจค้นพบเสื้อได้ และในบทต่อไปจะได้กล่าวถึงการสรุปผลการทดลอง ข้อเสนอแนะและปัญหาที่เกิดขึ้นจากการทดลอง

บทที่ 5
บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการดำเนินโครงการ

จากการดำเนินการศึกษาและทำโครงงานได้ผลสรุปดังนี้ การทดลองโปรแกรมค้นหาความคล้ายคลึงของรูปภาพ พบว่าสามารถทำการค้นหารูปภาพที่ต้องการได้ และแสดงผลลัพธ์ให้ผู้ใช้ได้ซึ่งสามารถแก้ได้อย่างถูกต้อง

การทดลองโปรแกรมค้นหารูปภาพ มีความผิดพลาดอยู่บ้างในเรื่องของการค้นหาภาพที่มีรายละเอียดของภาพมากทำให้การค้นหารูปภาพเกิดการผิดพลาดได้เนื่องจากฮิสโตแกรมมีรายละเอียดมากทำให้ไม่สามารถแยกความแตกต่างได้ และในการค้นหารูปภาพการทดลองการค้นหาด้วยสีแดงพื้นและมีลายจากตารางที่ 4.1 จะพบว่ามีการค้นหาออกมาได้ผลคล้ายภาพต้นแบบหรือเหมือนภาพต้นแบบที่ต้องการค้นหาได้ การค้นหารูปภาพด้วยการทดลองการค้นหาจากเสื้อลายขวางหรือสีสลับ หรือสีผสม จากตารางที่ 4.6พบว่า การค้นหาออกมาได้ผลไม่ตรงตามความต้องการซึ่งเป็นข้อจำกัดของโปรแกรมการค้นหาภาพคล้ายคลึง

5.2 ปัญหาในการดำเนินโครงการ

1. ถ้าหากรูปภาพในแฟ้มข้อมูลมีไม่เพียงพอผลการค้นหาก็จะไม่เป็นไปตามที่ต้องการหรือมีความคล้ายคลึงน้อย

2. ในการเพิ่มภาพในแฟ้มข้อมูลนั้น จำเป็นต้องเพิ่มในแฟ้มข้อมูล ที่เรากำหนดไว้เท่านั้นไม่สามารถทำการค้นหาจากที่อื่นนอกเหนือจากที่กำหนดได้จึงทำให้ต้องทำการเก็บข้อมูลรูปภาพไว้ที่เดียวกันโดยไม่สามารถแยกหมวดหมู่ของรูปภาพได้

3. ถ้าขนาดของรูปภาพที่ใช้การค้นหาไม่ได้มาตรฐานอาจทำให้การค้นหามีความผิดพลาดและคลาดเคลื่อนได้

4. หากภาพที่นำมาค้นหามีลายมากเกินไป เช่น การนำเสื้อลายขวางมาค้นหา ผลการค้นหาอาจเกิดความผิดพลาด ซึ่งอาจไม่ต้องตามความต้องการของผู้ใช้

5.3 ข้อเสนอแนะ

1. ผู้พัฒนาโปรแกรมสามารถพัฒนาโปรแกรมการค้นหาภาพคล้ายคลึงกัน โดยที่สามารถเพิ่มเงื่อนไขในการค้นหาภาพ เช่น การค้นหาจาก รูปร่าง, พื้นผิว, สี, สีและพื้นผิว เพื่อทำให้ผลการค้นหาถูกต้องและตรงตามความต้องการของผู้ใช้มากขึ้น

2. ควรมีข้อมูลรูปภาพในแฟ้มข้อมูลที่เพียงพอเพราะผลการค้นหาที่ไม่ตรงกับภาพที่ต้องการหรือคลาดเคลื่อนอาจเกิดจากข้อมูลในแฟ้มข้อมูลที่ใช้เก็บรูปภาพได้

3. ผู้ให้บริการเว็บไซต์สามารถนำโปรแกรมการค้นหาภาพที่คล้ายกันมาประยุกต์ใช้เพื่อทำให้กระบวนการทางธุรกิจ มีประสิทธิภาพ และตอบสนองความต้องการของลูกค้าให้ตรงตามความต้องการและรวดเร็ว เพื่อขยายโอกาสทางการค้าและการบริการของเว็บไซต์

4. ผู้พัฒนาโปรแกรมสามารถพัฒนาโปรแกรมการค้นหาภาพที่คล้ายคลึงกัน โดยเป็นตัวกลางในการค้นหาสินค้า เช่น ผู้ใช้ทำการค้นหาสินค้า ระบบการทำการประมวลผลหาภาพที่คล้ายคลึงกันและแสดงผลลัพธ์การค้นหารูปที่ต้องการมีในเว็บไซต์ผู้ให้บริการรายใดบ้างที่มีสินค้าชนิดนี้ เพื่อตอบสนองความต้องการของผู้ใช้

บรรณานุกรม

มหาวิทยาลัยเกษตรศาสตร์. (29 สิงหาคม 2550). **การแปลงรูปภาพสเกลสีเทาให้เป็นรูปภาพสีโดยวิธีการแบ่งส่วนและการจัดกลุ่มข้อมูล**. สืบค้นเมื่อ 23 กุมภาพันธ์ 2556, จาก http://research.rdi.ku.ac.th/world/cache/03/SuwatchaiCHUAll.pdf

พนมขวัญ ริยะมงคล. (17 กุมภาพันธ์ 2555). **DIGITAL IMAGE PROCESSING**. สืบค้นเมื่อ 3 มีนาคม 2556, จาก www.ecpe.nu.ac.th/panomkhawn/imagepro/pdf/ch01.pdf

วศิน สินธุภิญโญ. (8 มกราคม 2550). กิจกรรมการบรรยายพิเศษทางด้านวิทยาศาสตร์ คณิตศาสตร์และเทคโนโลยี. **Digital Image Processing and Digital Signal Processing.** สืบค้นเมื่อ 26 มกราคม 2556, จาก www.mwit.ac.th/~jeed/r-sp/doc/r-sp-2-49-2.pdf

สถาบันเทคโนโลยีนานาชาติสิรินธร มหาวิทยาลัยธรรมศาสตร์. (29 พฤษภาคม 2550). **Computer Graphics.** สืบค้นเมื่อ 25 มกราคม 2556, จาก http://www.siamtechu.net/~mullika/Graphic/imageprocessing.pdf

A. Jain, (1989). **Fourier Transform**. Retrieved March 2, 2013, from http://homepages.inf.ed.ac.uk/rbf/HIPR2/fourier.htm

John M. Brayer, (1985). **INTRODUCTION TO FOURIER TRANSFORMS FOR IMAGE PROCESSING.** Retrieved March 2, 2013, from http://www.cs.unm.edu/~brayer/vision/fourier.html

SS Plotkin, PNAS, (2007). **Distance**. Retrieved March 8, 2013, from http://en.wikipedia.org/wiki/Distance

Taneja, HC (2008). **Fourier Transform**. Retrieved March 2, 2013, from http://en.wikipedia.org/wiki/Fourier_transform

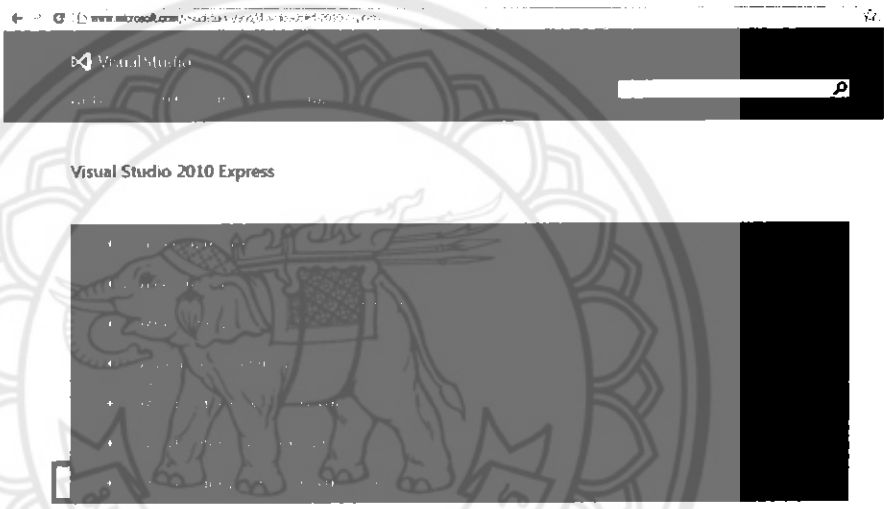
Wikipedia, (2010). **HSL and HSV**. Retrieved January 16, 2013, from http://en.wikipedia.org/wiki/HSL_and_HSV

Wikipedia, (2013). **RGB color model**. Retrieved January 16, 2013, from http://en.wikipedia.org/wiki/RGB_color_model

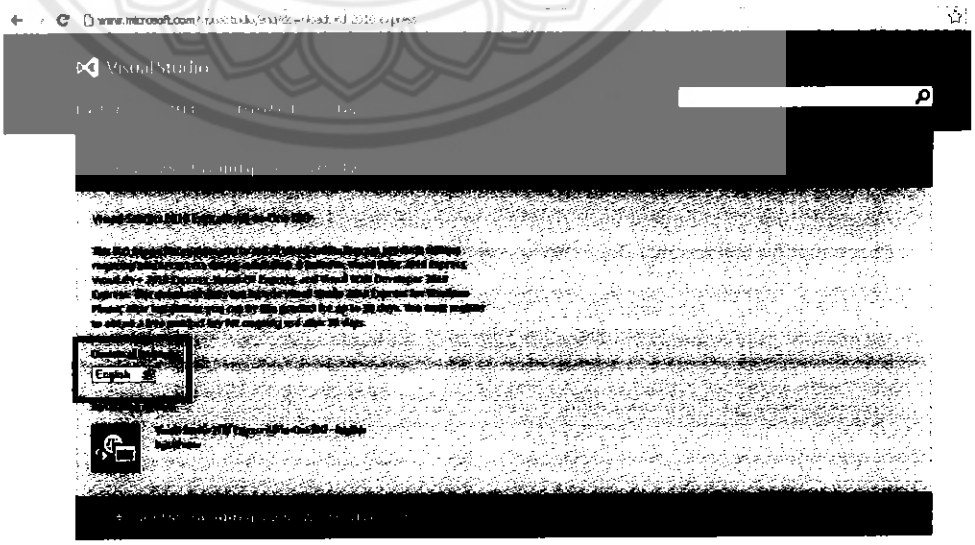
ภาคผนวก ก

ขั้นตอนการติดตั้ง Microsoft Visual Studio 2010 Express

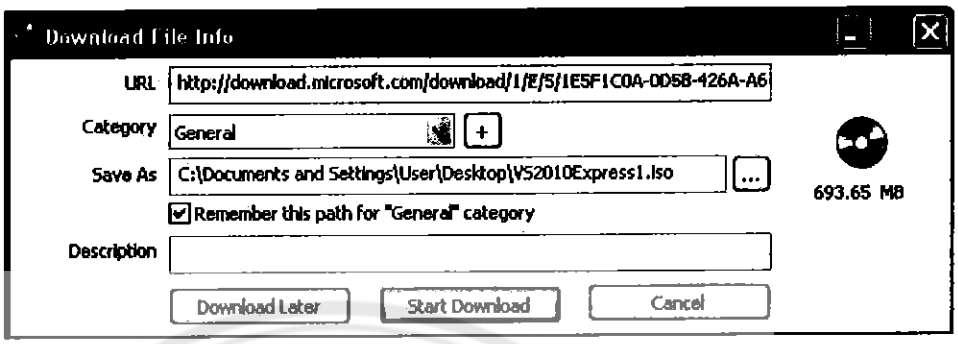
1. เข้าไปที่เว็บไซต์ http://www.microsoft.com/visualstudio/eng/products/visual-studio-express-productsคลิกเลือก All-Offline Install ISO image file ซึ่งเป็นการติดตั้งแบบไม่ต้องผ่าน Internet เมื่อทำการดาวน์โหลดเรียบร้อย ก็สามารถทำการติดตั้งได้



2. เลือกภาษาที่ใช้ในการติดตั้ง ทางผู้จัดทำเลือกใช้ภาษาอังกฤษ



3. คลิกที่ Download File เสร็จแล้วก็เลือก Location ที่เก็บไฟล์คลิกที่ StartDownload



4. หลังจากดาวน์โหลดเสร็จสิ้น จะได้ไฟล์ VS2010Express1.iso

5. ทำการแตกไฟล์ที่เราดาวน์โหลดมาเพื่อทำการติดตั้ง คลิกที่ Visual Studio 2010 Express Setup โปรแกรมจะแสดงให้ เลือก 4 Product

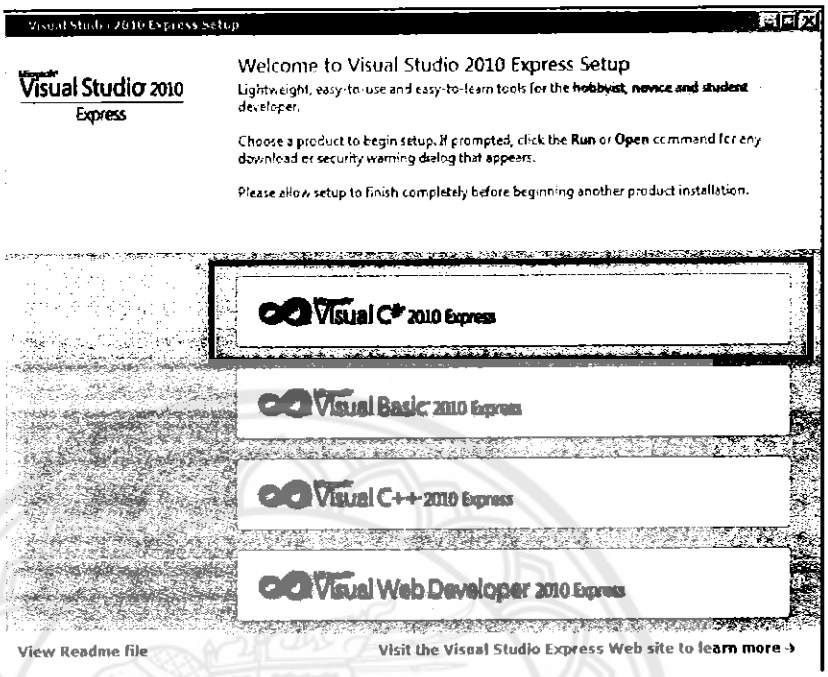
Visual C# 2010 Express

Visual Basic 2010 Express

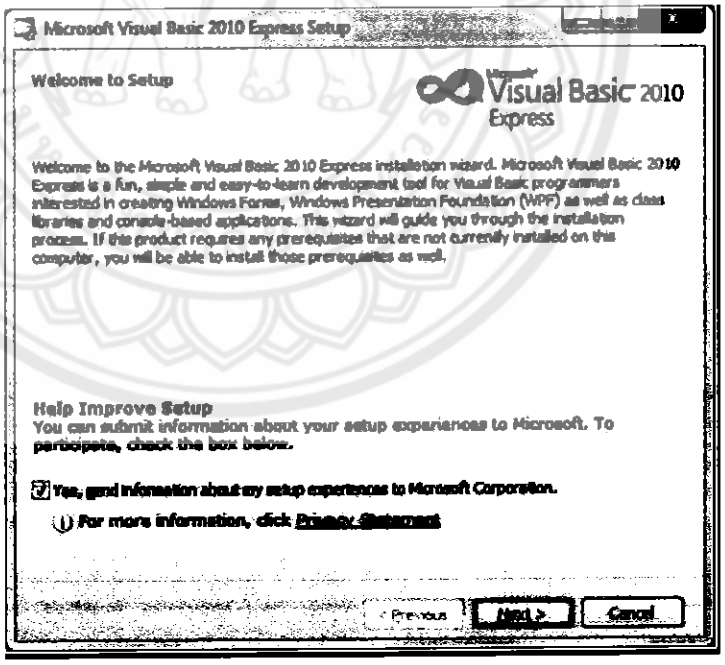
Visual C++ 2010 Express

Visual Web Developer 2010 Express

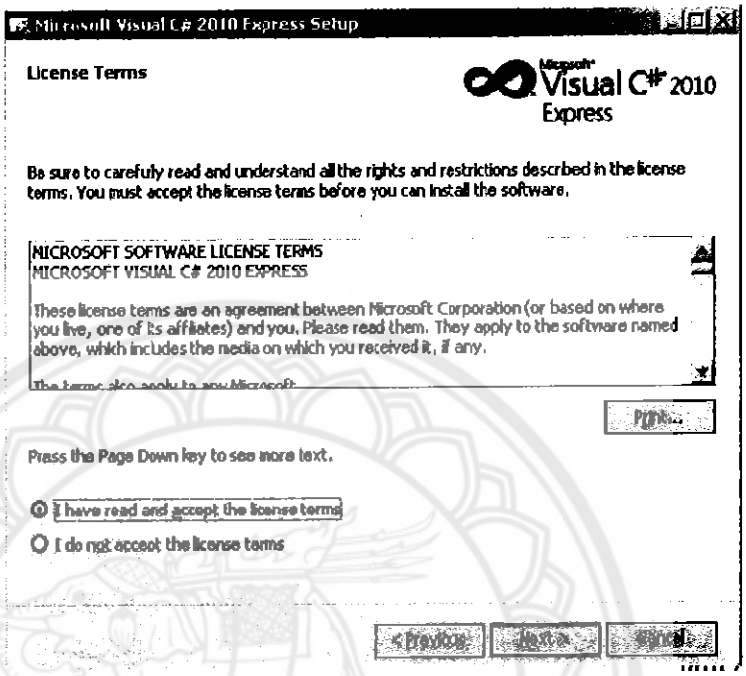
ให้คลิกเลือก Visual C# 2010 Express เพราะเป็นโปรแกรมที่เราใช้พัฒนา



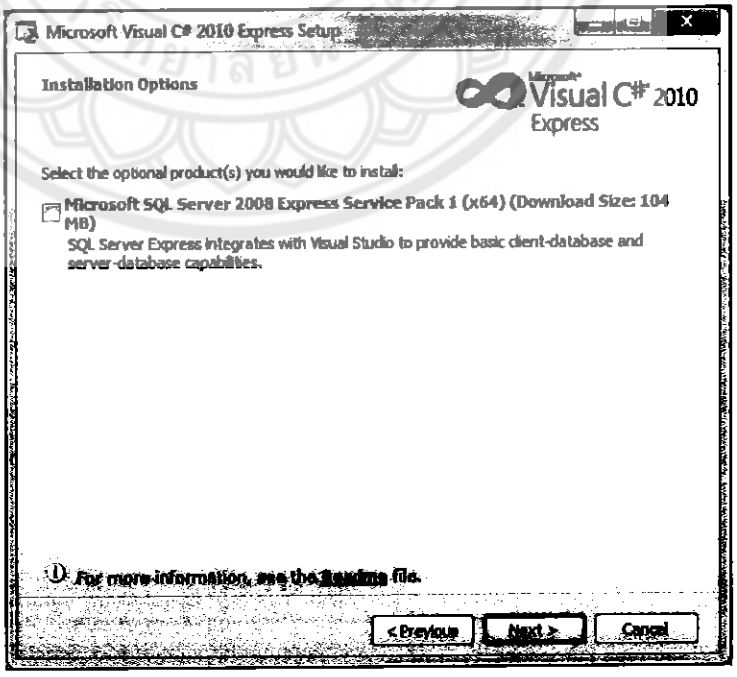
6. ทำเครื่องหมาย ✔ หน้าข้อความ หลังจากนั้น คลิก Next เพื่อทำขั้นตอนต่อไป



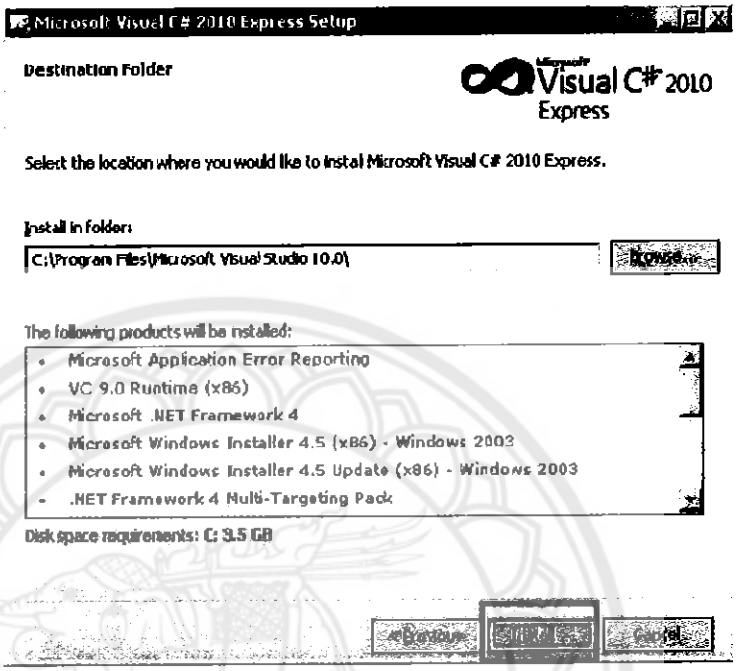
7. คลิกเลือก I have read and accept the license terms



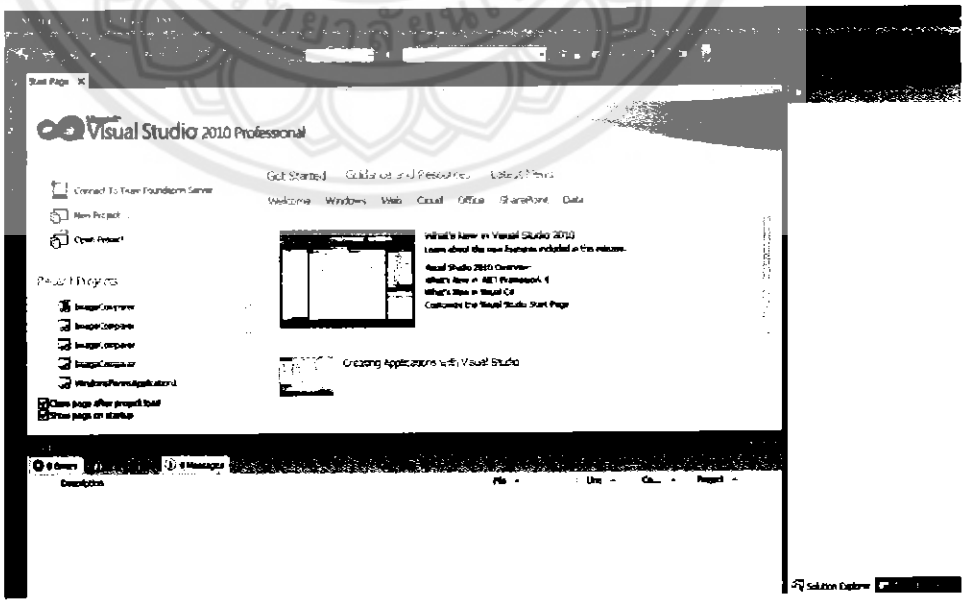
8. คลิกเลือก Microsoft Silverlight และ Microsoft SQL Server 2008 Express Service Pack1 (X 64) คลิกNext เพื่อทำขั้นตอนถัดไป



9. คลิกปุ่ม Install

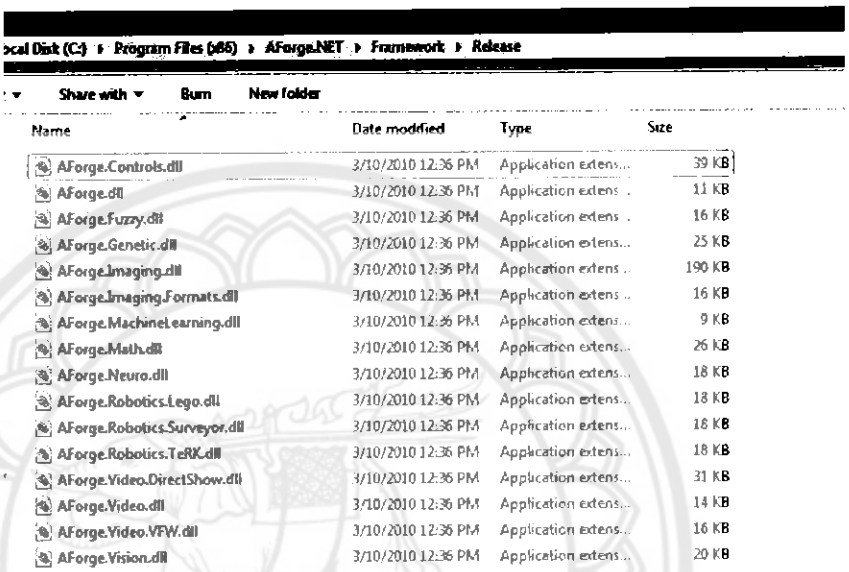


10. หลังจากติดตั้งเสร็จ เรียบร้อยให้ เข้าไปที่ Sart>All Programs > Microsoft Visual Studio 2010 Express ก็จะมี Product ที่ได้ทำการติดตั้งไว้ก่อนหน้านี้ เป็นอันเสร็จสิ้นการติดตั้งMicrosoft Visual Studio 2010 Express ที่ใช้ในการพัฒนาโครงงาน



ภาคผนวก ข

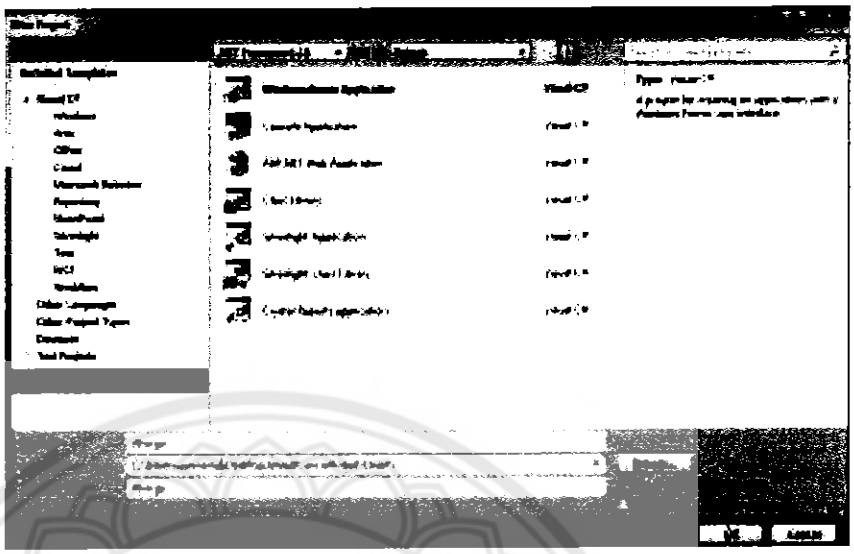
ขั้นตอนการติดตั้ง AForge.NET ในVisual Studio 2010



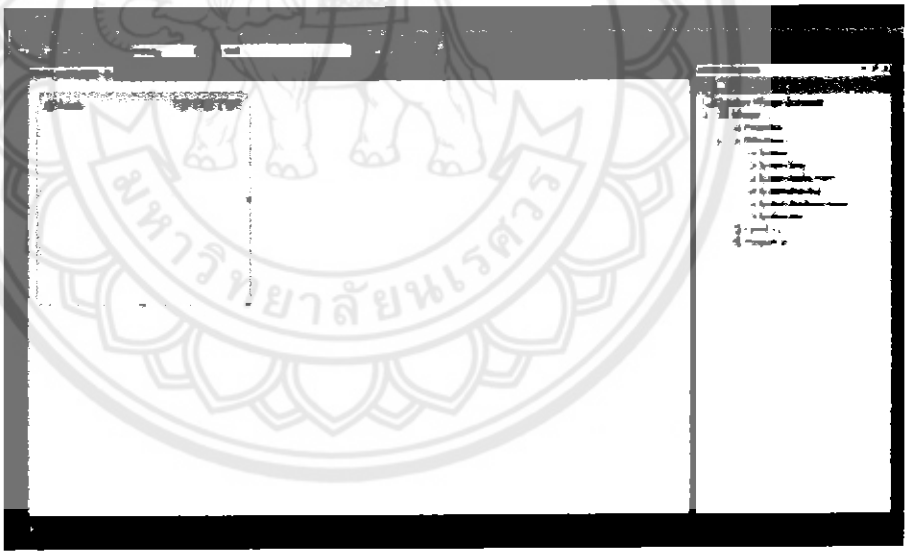
1. เราจะต้องดาวน์โหลด AForge.NET จากที่นี่ -> AForge.NET เมื่อทำการติดตั้งเสร็จเรียบร้อยโปรแกรมจะแสดงไลบรารี(Library) ต่างๆ

C:\Program Files (x86)\AForge.NET\Framework\Release – For 64 bit OS

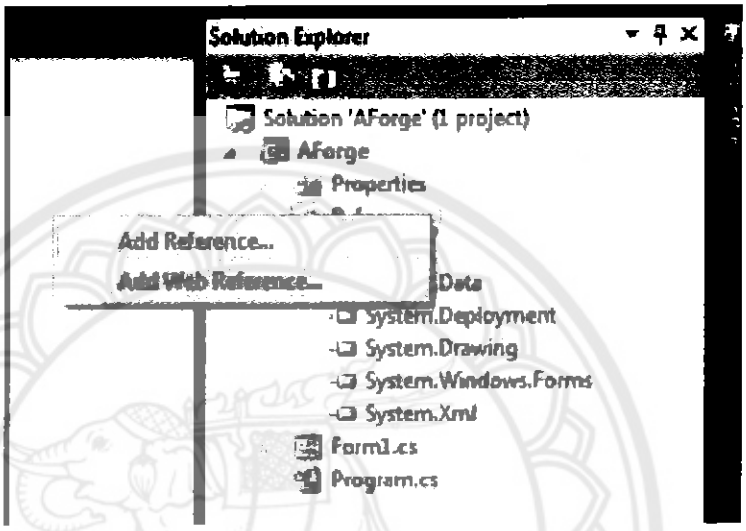
2. ทำการทดสอบโปรแกรมดังนี้ คลิก File >>>New>>>Project>>>Windows Forms



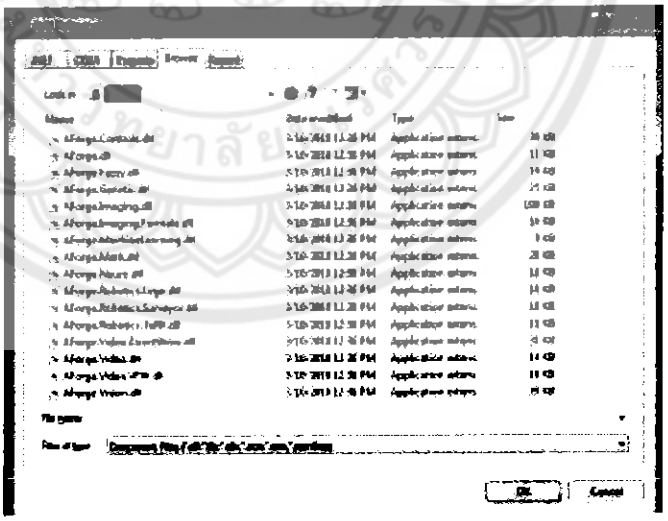
3. จากนั้นจะปรากฏหน้าต่างดังนี้



4. เพิ่ม AForge.NET libraries คลิกขวาที่ Solution Explorer หลังจากนั้นคลิกที่Add Reference



5. หลังจากนั้น โปรแกรมจะทำการแสดงการติดตั้งไลบรารี่(Libraries)



6. หลังจากนั้นเลือก AForge.dllคลิกOKหลังจากนั้น AForge.dll ก็สามารถใช้ร่วมกับ Visual Studio 2010 ได้