ระบบฐานข้อมูลรูปภาพจัดการค้นคืนแบบอัตโนมัติโดยใช้นิวรอนเน็ตเวิร์ค

AUTOMATIC MACHINE INTERACTION USING SELF-ORGANIZING NEURAL NETWORK FOR IMAGE RETRIEVAL

นายกฤตพน พงษ์วิทยานุกฤต รหัส 43370279

นางสาวพงันญา ฉันทกานันท์ รหัส 43370550

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ - 9 ส.ค. 2547
เลขทะเบียน 4700200
เลขเรียกหนังสือ
มหาวิทยาลัยนเรศวร

ปร. ก277ร 2546

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2546

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน	ระบบฐานข้อมูลรูปภาพจัดการค้นคืนแบบอัตโนมัติโดยใช้นิวรอนเน็ตเวิร์ค
ผู้ดำเนินโครงงาน	นายกฤตพน พงษ์วิทยานุกฤต รหัส 43370279
	นางสาวพรัญญา ฉันทกานันท์ รหัส 43370550
อาจารย์ที่ปรึกษา	ดร. ไพศาล มุณีสว่าง
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2546

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงงานวิศวกรรม

..............................ประธานกรรมการ
(ดร. ไพศาล มุณีสว่าง)

..............................กรรมการ
(ดร. สุรเชษฐ์ กานต์ประชา)

..............................กรรมการ
(ดร. สุชาติ แย้มเม่น)

หัวข้อโครงงาน	ระบบฐานข้อมูลรูปภาพจัดการค้นคืนแบบอัตโนมัติโดยใช้นิวรอนเน็ตเวิร์ค
ผู้ดำเนินโครงงาน	นายกฤตพน พงษ์วิทยานุกฤต รหัส 43370279
	นางสาวพงันญา ฉันทกานันท์ รหัส 43370550
อาจารย์ที่ปรึกษา	ดร. ไพศาล มุณีสว่าง
สาขาวิชา	วิศวกรรมศาสตร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2546

....

บทคัดย่อ

ในปัจจุบันนี้การใช้งานระบบดิจิตอลและระบบเครือข่ายอินเตอร์เน็ต ได้เข้ามามีบทบาทในชีวิตประจำวันของเรามากขึ้น และการใช้ข้อมูลข่าวสารในปัจจุบันก็กำลังมีแนวโน้มที่จะเข้าไปใช้เทคโนโลยีนี้กันมากขึ้น ข้อมูลประเภทภาพถ่ายก็เป็นอีกสิ่งหนึ่งที่กำลังเป็นที่นิยมในปัจจุบัน จะเห็นได้จากมีการเกิดขึ้นของ กล้องระบบดิจิตอลที่จะนำเทคโนโลยีนี้เข้าไปใช้ แต่เมื่อมีการใช้ระบบดิจิตอลเข้ามาเกี่ยวข้องแล้ว ปัญหาที่ตามมาคือ เราจะเก็บข้อมูลประเภทนี้ได้อย่างไร และจะเรียกคืนหรือนำกลับมาใช้งานได้อย่างไร เมื่อข้อมูลเหล่านี้มีจำนวนมากขึ้นซึ่งเป็นที่มาของโครงงานวิจัยนี้

ในโครงงานนี้จะเป็นการสร้างซอฟต์แวร์ที่ใช้เก็บรวบร่วมรูปภาพและสามารถค้นคืนภาพนั้นๆ ให้กับผู้ใช้ โดยโครงงานนี้จะเป็นการพัฒนาซอฟต์แวร์ให้สามารถใช้งานผ่านระบบเครือข่ายอินเตอร์เน็ต หรือระบบเน็ตเวิร์คใดๆ เพื่อเป็นที่เก็บรวบรวมข้อมูลประเภทภาพถ่าย จุดเด่นของโครงงานนี้อยู่ที่ มีการนำเทคโนโลยีที่เรียกว่า Neural network มาใช้กับโครงงานวิจัย เพื่อช่วยในการค้นหาและสอนให้ระบบ ทำการค้นหาภาพถ่ายนั้นๆ แทนผู้ใช้งานระบบ และเป็นการลดปัญหาในการเกิด Traffic ในระบบเน็ตเวิร์คอีกทางหนึ่ง

Project Title	Automatic Machine Interaction Using Self-Organizing Neural Network For Image Retrieval
Name	Mr. Kritpon Pongvittayanukrit ID. 43370279
	Miss Panganya Chantakanan ID. 43370550
Project Advisor	Dr. Paisarn Muneesawang
Major	Computer Engineering
Department	Electrical and Computer Engineering
Academic Year	2003

ABSTRACT

This moment, using digital system and internet network system are involved with everyday life. Currently information technology trends to be more and more available. Type of information photograph is very popular used. And now we can see that digital camera system uses this kind of technology. When we used digital system, there are many problems as following when there are more information.

This project intended to construct software that collecting photograph and recalling any pictures for user. This project is also developing software for transferring internet system or any internet system in collecting photograph data. The highlight of this project is Neural network technology, which uses for finding and teaching system for recalling those photographs instead of user. This technology will also decrease problem in traffic system in network.

กิตติกรรมประกาศ

ในการทำโครงงานวิศวกรรมครั้งนี้ คณะผู้จัดทำขอกราบขอบพระคุณ อาจารย์ไพศาล มุณีสว่าง ซึ่งเป็นอาจารย์ที่ปรึกษาโครงงาน ตามที่ท่านได้ให้ความกรุณาแนะนำวิธีในการทำงานให้เข้าใจถึงการศึกษาอย่างเป็นระบบขั้นตอน อีกทั้งสละเวลาเพื่อตรวจสอบการทำงานและชี้แนวทางแก้ไขในทุกขั้นตอนตลอดการศึกษา

นายกฤตพน พงษ์วิทยานุกฤต
นางสาวพรันญา ฉันทกานันท์

สารบัญ

หน้า

บทคัดย่อภาษาไทย....ก

บทคัดย่อภาษาอังกฤษ....ข

กิตติกรรมประกาศ....ค

สารบัญ....ง

สารบัญตาราง....ช

สารบัญรูป....ซ

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญของโครงงาน....1

1.2 วัตถุประสงค์ของโครงงาน....4

1.3 ขอบข่ายของโครงงาน....4

1.4 ขั้นตอนการดำเนินงาน....5

1.5 ผลที่คาดว่าจะได้รับ....5

1.6 งบประมาณที่ใช้....6

บทที่ 2 งานวิจัยที่เกี่ยวข้อง

2.1 คุณสมบัติพื้นฐานของรูปภาพ....7

2.2 การค้นหาภาพโดยใช้ระบบการติดต่อสื่อสารกับผู้ใช้ในพื้นฐานของภาพหรือหลักการค้นหาแบบป้อนกลับ (Relevance Feedback)....7

2.3 การค้นหารูปภาพแบบอัตโนมัติโดยใช้ CBIR....8

2.4 SOTM (Self-Organizing Tree Map)....9

บทที่ 3 Neural Network แบบไม่มีผู้สอนโครงสร้างแบบต้นไม้ (SOTM)

3.1 Neural Network แบบไม่มีผู้สอนโครงสร้างแบบต้นไม้....10

3.1.1 อัลกอริทึมของ SOTM....10

3.2 การทดลองเบื้องต้น....12

3.2.1 การทดลองที่ 1....12

3.2.2 สรุปผลการทดลองที่ 1....15

สารบัญ (ต่อ)

หน้า

3.2.3 การทดลองที่ 2 16

3.2.4 สรุปผลการทดลองที่ 2 19

บทที่ 4 วิธีดำเนินการวิจัย

4.1 ขั้นตอนการทำ image indexing 20

4.2 การทำงานของ SOTM 24

4.3 กระบวนการการค้นคืน (Relevance Feedback algorithm) 24

บทที่ 5 ผลการทดลอง

5.1 ผลการทดลองโดยใช้ Texture เป็นตัวหา 30

5.2 การทดลองเรื่องความถูกต้องในการเลือกภาพของ SOTM 35

5.3 ผลการทดลองโดยใช้ Color เป็นตัวค้นหา 46

5.4 ตัวอย่างการทดลอง Simple Search กับภาพในลักษณะ Texture และ Color ที่มีลักษณะความยากง่ายต่างกัน 52

5.5 การทดลองในเรื่องของการลดจำนวนการใช้ระบบเครือข่าย 55

บทที่ 6 บทสรุป

6.1 สรุปผล 56

6.2 ข้อเสนอแนะ 57

6.3 ปัญหาที่พบในการทำโครงงานวิจัย 58

6.4 แนวทางในการทำโครงงานวิจัย 58

เอกสารอ้างอิง 59

ภาคผนวก

ก Source Code Java 60

ข ตัวอย่างการทดลอง Simple Search กับภาพในลักษณะ Texture และ Color ที่มีลักษณะความยากง่ายต่างกัน 62

สารบัญ (ต่อ)

หน้า

ค การทดลองเรื่องการใช้ SOTM ในการจำแนกประเภท Texture 64

ประวัติผู้เขียนโครงงาน..70

สารบัญตาราง

ตารางที่ หน้า

3.1 กลุ่มตัวอย่างในการทดสอบ....13
3.2 ค่า a(t) ที่คำนวณได้....13
3.3 ค่าหลังจากที่ได้ผ่านการทดสอบแล้ว....13
5.1 บันทึกผลการทดลอง Simple Search Method โดยใช้ Texture....31
5.2 บันทึกผลการทดลอง SOTM Method โดยใช้ Texture....32
5.3 บันทึกผลการทดลอง Semi Automatic Method โดยใช้ Texture....33
5.4 บันทึกผลการทดลอง User Feedback Method โดยใช้ Texture....34
5.5 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 268.jpg....35
5.6 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 254.jpg....36
5.7 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 278.jpg....37
5.8 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 3.jpg....38
5.9 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 24.jpg....39
5.10 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 740.jpg....40
5.11 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 755.jpg....41
5.12 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 872.jpg....42
5.13 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 1174.jpg....43
5.14 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 1203.jpg....44
5.15 เปรียบเทียบการทำงานแต่ละ Method ของการใช้ Texture ค้นหา....45
5.16 บันทึกผลการทดลอง simple search โดยใช้ Color....48
5.17 บันทึกผลการทดลอง user feedback โดยใช้ Color....49
5.18 บันทึกผลการทดลอง SOTM โดยใช้ Color....50
5.19 บันทึกผลการทดลอง Semi Automatic Method โดยใช้ Color....51
5.20 เปรียบเทียบการทำงานแต่ละ Method ของการใช้ Color ค้นหา....52
5.21 ผลการทดสอบการรับส่งข้อมูล....55
6.1 การเปรียบเทียบระหว่าง Method ต่างๆ....57

สารบัญรูป

รูปที่ หน้าที่

2.1 หน้าจอการที่ซอฟต์แวร์โต้ตอบกับผู้ใช้.....8
2.2 การค้นหารูปภาพแบบอัตโนมัติโดยใช้ CBIR.....9
2.3 การจัดกลุ่มของ SOTM ระหว่าง 1 กลุ่มกับ 2 กลุ่ม.....9
3.1 การใช้ SOTM หาจุดศูนย์กลางของกลุ่มเวคเตอร์กลุ่มเดียว.....12
3.2 Java code สำหรับ SOTM.....14
3.3 เอาต์พุต (Output) ที่ได้จาก Source Program Java.....15
3.4 ผลของจุดศูนย์กลางของกลุ่มตัวอย่าง.....15
3.5 ขั้นตอนการทำงานของ SOTM ในการหาจุดศูนย์กลางของกลุ่มสองกลุ่ม.....16
3.6 ตัวอย่างการใช้ SOTM ในหารหาจุดศูนย์กลางของกลุ่มแบบสองกลุ่ม.....17
3.7 กลุ่มตัวอย่างที่ใช้ในการหาจุดศูนย์กลางของกลุ่มแบบสองกลุ่ม.....17
3.8 ผลเอาต์พุต (Output) ที่ได้จาก Source Program Java.....18
4.1 การแบ่งค่า H และ S ส่วนบนในโมเดล HSV.....21
4.2 การแบ่งค่าในโมเดล HSV.....21
4.3 ตัวอย่างการคิดค่า histogram จากการใช้โมเดล.....22
4.4 ภาพตัวอย่างที่นำมาเปรียบเทียบ.....23
4.5 กราฟผลของการทำ histogram ของภาพตัวอย่าง.....23
4.6 การทำงานที่มี SOTM เข้ามาช่วย.....24
4.7 ภาพตัวอย่างแนวคิดหาความคล้ายคลึงของภาพโดยไม่คิดน้ำหนัก.....25
4.8 ภาพแนวคิดการหาค่าความคล้ายคลึงของภาพโดยคิดน้ำหนัก.....27
5.1 ภาพการแยกกลุ่มเปรียบเทียบระหว่างคน กับ SOTM โดยรูป.....30
5.2 กราฟเปรียบเทียบการทำงานแต่ละ Method ของการใช้ Texture เป็นตัวหา.....45
5.3 ผลที่ได้จากการทำ simple search ของการใช้ Color เป็นตัวหา.....46
5.4 ผลที่ได้หลังจากผู้ใช้มี feedback ครั้งแรก.....46
5.5 ผลที่ได้หลังจากผู้ใช้มี feedback ให้ครั้งที่สอง.....47
5.6 ผลที่ได้หลังจากผู้ใช้มี feedback ให้ครั้งที่สาม.....47
5.7 กราฟการทำงานแต่ละ Method ของการใช้ Color ในการหา.....52
5.8 ตัวอย่าง Simple Search ที่เป็น Query ที่เป็นภาพลักษณะง่าย.....53
5.9 ตัวอย่าง Simple Search ที่เป็น Query ที่เป็นภาพลักษณะปานกลาง.....54

สารบัญรูป (ต่อ)

รูปที่ หน้าที่

6.1 ผลที่ได้จากการทดลองเปรียบเทียบทุก Methods....56

ก-1 Source code การคำนวณหาจุดศูนย์กลาง....60

ข-1 ตัวอย่าง Simple Search ที่เป็น Query ที่เป็นภาพลักษณะยากมาก....62

ข-2 ตัวอย่าง Simple Search ที่เป็น Query ที่เป็นภาพลักษณะง่าย....63

ข-3 ตัวอย่าง Simple Search ที่เป็น Query ที่เป็นภาพลักษณะปานกลาง....64

ข-4 ตัวอย่าง Simple Search ที่เป็น Query ที่เป็นภาพลักษณะยาก....65

ค-1 ภาพการแยกกลุ่มเปรียบเทียบระหว่างคนกับ SOTM....66

ค-2 ผลการแยกกลุ่ม....68

มหาวิทยาลัยนเรศวร

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ในปัจจุบันเทคโนโลยีอินเตอร์เน็ต ได้เข้ามีบทบาทกับมนุษย์มากขึ้น เพราะในอินเตอร์เน็ต ได้รวบรวมข้อมูลข่าวสารไว้อย่างมากมาย จึงมีผู้สนใจและผู้ใช้บริการจำนวนมาก เนื่องจากใช้ได้ไม่จำกัดสถานที่และมีค่าใช้จ่ายน้อยเมื่อเทียบกับสื่ออื่นๆ ผู้ใช้สามารถเข้าไปค้นหาข้อมูลต่างๆ ที่ต้องการได้ ไม่ว่าจะเป็น ทางด้านการศึกษา การบันเทิง ข่าวสาร และบริการอื่นๆ อีกมากมาย รวมทั้งการค้นหาข้อมูลภาพด้วย

ในอดีตการค้นหาข้อมูลภาพนั้นยังทำได้ไม่สะดวก เพราะการจัดเก็บข้อมูลประเภทรูปภาพนั้นจะมีการจัดเก็บเป็นพวก ภาพถ่ายหรือฟิล์ม ซึ่งต้องเก็บไว้ในสถานที่เฉพาะเพื่อป้องกันการเสียหาย ภาพถ่ายหรือฟิล์มจึงจะมีอายุการใช้งานที่ยาวนาน เมื่อปริมาณมีมากขึ้นพื้นที่ที่ใช้จัดเก็บก็จะเพิ่มปริมาณตามไปด้วย ทำให้การค้นหาไม่สะดวกเมื่อเทคโนโลยีด้านคอมพิวเตอร์ได้มีการพัฒนาขึ้น ได้มีความคิดที่จะแก้ปัญหาดังกล่าว โดยการจัดเก็บข้อมูลประเภทรูปภาพนั้นลงบนเครื่องคอมพิวเตอร์ ซึ่งปลอดภัยและประหยัดพื้นที่ในการจัดเก็บ แต่ปัญหาที่ตามมานั้น คือการค้นหาภาพที่ต้องการนั้นยังเป็นปัญหาอยู่ หากข้อมูลต้องการไม่สามารถค้นกลับมาใช้ได้ก็ถือได้ว่าข้อมูลนั้นไม่มีประโยชน์อันใด และเมื่อมีความต้องการจากผู้ใช้เป็นจำนวนมาก จึงมีผู้คิดค้นที่จะจัดเก็บข้อมูลประเภทนี้ให้อยู่ในรูปแบบของฐานข้อมูล เพื่อช่วยลดปัญหาในการค้นคืนรูปภาพที่ต้องการได้สะดวกมากขึ้นและกระจายการใช้งานให้ผู้ใช้พร้อมๆกันหลายๆคน

ในการค้นหาข้อมูลภาพที่เป็นภาพดิจิตอล เพื่อสนองตอบความต้องการของผู้ใช้โดยทั่วไปไม่ว่าการค้นหาจากเน็ตเวิร์ค (Network) เช่น อินเตอร์เน็ต (Internet) หรือจาก ฐานข้อมูล (Database) นั้น ในอดีตนั้น จะจัดเก็บข้อมูลเหล่านี้ลงในฐานข้อมูลโดยมีคีย์เวิร์ด (keyword) ใช้ช่วยในการค้นหาข้อมูลกลับในเวลาที่ต้องการใช้งาน ซึ่งคีย์เวิร์ดที่ใช้นั้นอาจเป็น ชื่อสถานที่ รูปร่างของภาพ หรือ อาจเป็นสีของรูปภาพ ฯลฯ ซึ่งคีย์เวิร์ดจะถูกกำหนดขึ้นในขั้นตอนของการจัดเก็บข้อมูลลงในฐานข้อมูลโดยผู้บันทึกข้อมูล และเพื่อไม่ให้เกิดความคิดไม่ตรงกันระหว่างผู้ใช้กับผู้บันทึกข้อมูลจึงจำเป็นต้องมีมาตรฐานในการกำหนดคีย์เวิร์ดเพื่อให้เข้าใจไปในทางเดียวกันและเพื่อความถูกต้องในการค้นหาข้อมูล การใช้คีย์เวิร์ดในการค้นหาภาพนั้นทำให้การค้นหาสะดวกและง่าย แต่การที่จะให้ผู้ใช้และผู้บันทึกข้อมูลมีความคิดตรงกันในการใช้คีย์เวิร์ดนั้นเป็นเรื่องยาก เพราะผู้บันทึกอาจใช้ คีย์เวิร์ดอย่างหนึ่งในการอธิบายรูปภาพ ส่วนผู้ใช้กลับใช้คีย์เวิร์ด อีกอย่างหนึ่งในการค้นหารูปภาพ เป็นผลให้ผู้ใช้ไม่สามารถค้นหาข้อมูลภาพตามที่ต้องการได้ และปัญหาอีกอย่างใน

ขั้นตอนการจัดเก็บก็คือผู้จัดเก็บเก็บข้อมูลภาพลักษณะเดียวกัน แต่ใช้คีย์เวิร์ดต่างกัน ทำให้ไม่สามารถค้นหาข้อมูลภาพจากฐานข้อมูลได้

จากปัญหาที่พบดังกล่าว ได้มีการแก้ไขโดยใช้ GUI (Graphic User Interface) เข้ามาช่วย ทำให้ผู้ใช้งานมีการโต้ตอบกับซอฟต์แวร์หรือฐานข้อมูลได้ โดยในการโต้ตอบนั้นจะเป็นการสอนให้คอมพิวเตอร์เรียนรู้สิ่งที่ผู้ใช้ต้องการ เพื่อนำไปใช้ในการค้นหารูปภาพที่มีลักษณะที่เหมือนหรืออยู่ในกลุ่มของภาพตัวอย่างที่ผู้ใช้เลือก เมื่อซอฟต์แวร์พบกลุ่มของข้อมูล หรือ รูปภาพที่มีลักษณะเดียวกันกับกลุ่มภาพตัวอย่างที่ผู้ใช้ระบุไว้ในขั้นต้นแล้วนั้น ก็จะส่งข้อมูลภาพดังกล่าวกลับไปให้ผู้ใช้ ในการส่งข้อมูลภาพกลับมาให้ผู้ใช้ (supervised) ในครั้งแรกนี้นั้นภาพที่ได้อาจมีจำนวนมากและยังไม่ตรงกลับความต้องการของผู้ใช้ในทันที ดังนั้นผู้ใช้จึงจำเป็นที่จะต้องคัดเลือกภาพที่ต้องการเพื่อส่งกลับไปเป็นภาพตัวอย่างหรือเงื่อนไขให้ซอฟต์แวร์นำกลับไปเลือกภาพให้อีกครั้งหนึ่ง เมื่อซอฟต์แวร์นำเงื่อนไขที่ระบุในครั้งนี้กลับมาคัดเลือกภาพ กลุ่มของภาพที่คัดเลือกได้ก็จะเริ่มจำกัดขอบเขตให้แคบลง และซอฟต์แวร์ก็จะส่งภาพที่คัดเลือกได้กลับไปให้ผู้ใช้ตรวจสอบอีกครั้งหนึ่ง ความถูกต้องของภาพจะขึ้นอยู่กับจำนวนครั้งที่ผู้ใช้กับซอฟต์แวร์โต้ตอบกัน ถ้ามีการโต้ตอบกันระหว่างผู้ใช้กับซอฟต์แวร์มากเท่าไรความถูกต้องของภาพก็จะมากขึ้นเท่านั้น เป็นที่ทราบกันดีอยู่แล้วว่าข้อมูลประเภทรูปภาพนั้นมีขนาดใหญ่จึงต้องใช้แบนวิด (bandwidth) ในการรับส่งค่อนข้างมากในแต่ละครั้ง ซึ่งในการส่งข้อมูลกลับไปกลับมาแต่ละครั้งนั้นเรียกว่า RF (relevance feedback) ทำให้เกิด traffic ขึ้นบนระบบเน็ตเวิร์ค

ถึงแม้ว่าจะเป็นการแก้ปัญหาที่เกิดขึ้นในอดีต แต่ก็พบปัญหาใหม่เกิดขึ้นอีกนั่นก็คือเมื่อมีการรับส่งข้อมูลจำนวนมากผ่านทางระบบเน็ตเวิร์คทำให้เกิด traffic ขึ้น ในการลด traffic จะต้องลดการส่งข้อมูลกลับไปกลับมาระหว่างซอฟต์แวร์ทางฝั่งเซิร์ฟเวอร์ (server) กับผู้ใช้ระบบ (Supervised) การที่จะลดได้นั้นจะต้องทำให้มีการประมวลผลเฉพาะบนฝั่งของเซิร์ฟเวอร์ให้มากขึ้นโดยให้มีการติดต่อกับผู้ใช้ระบบน้อยลง เป็นการลดขั้นตอนในการใช้งานทำให้การใช้งานสะดวกมากขึ้น

ดังนั้นจึงจำเป็นที่จะต้องออกแบบระบบการติดต่อกับผู้ใช้ (Graphic User Interface) รวมถึงขั้นตอนในการประมวลผลในฝั่งของเซิร์ฟเวอร์ใหม่ เพื่อลดปัญหาดังกล่าวลง โดยในการออกแบบจะคำนึงถึงการทำให้ซอฟต์แวร์ในฝั่งเซิร์ฟเวอร์นั้นให้สามารถเรียนรู้ (Learning Machine) ในเงื่อนไขที่ผู้ใช้กำหนดขึ้นในขั้นต้น แล้วนำเงื่อนไขนั้นไปค้นหาและกำจัดข้อมูล (Query) ที่เก็บไว้ในรูปของภาพดิจิตอลที่ไม่ตรงต่อความต้องการออกจากกลุ่มข้อมูลภาพได้เอง โดยพึ่งผู้ใช้ให้น้อยที่สุด ซึ่งจะอยู่ในรูปแบบของการวิเคราะห์โดยอัตโนมัติ (Fully Automated Image Analysis)

ในการวิเคราะห์เนื้อหาของภาพที่ผู้ใช้เรียกมาจากฐานข้อมูลนั้นจะใช้แสงสีของภาพ (Brightness and Color) รายละเอียด (Texture) และข้อมูลในด้านรูปร่าง (Region Shape) เพื่อบอกถึง

ความแตกต่างของภาพและแยกภาพออกจากกัน ข้อมูลที่ได้มาจะอยู่ในรูปของเวคเตอร์ซึ่งแตกต่างจากการแยกภาพโดยใช้ keyword ซึ่งจะใช้ข้อความในการบรรยายภาพ

จากที่กล่าวมาแล้วขั้นต้นเป็นที่มาของโครงงานนี้ ซึ่งเป็นโครงงานในการพัฒนาซอฟต์แวร์ เพื่อการสืบค้นข้อมูลภาพจากฐานข้อมูลบนเครือข่ายอินเตอร์เน็ต ในรูปแบบอัตโนมัติ โดยใช้ภาพหรือเนื้อหาของภาพในการค้นหา โครงงานนี้จะช่วยแก้ปัญหาดังกล่าวโดยใช้ภาพหรือเนื้อหาของภาพในการที่จะค้นหาภาพที่ผู้ใช้ต้องการ ซึ่งในการบอกให้คอมพิวเตอร์รับรู้ว่าภาพที่ผู้ใช้ข้อมูลต้องการนั้นเป็นอย่างไร จำเป็นต้องมีการอินพุต (input) ข้อมูลเบื้องต้นเพื่อเป็นเงื่อนไขในการคัดเลือกแก่ซอฟต์แวร์ เสียก่อนซึ่งข้อมูลเบื้องต้นที่อินพุตให้นั้นจะเป็นกลุ่มตัวอย่างของภาพที่ซอฟต์แวร์จะใช้ในการเปรียบเทียบกับภาพในฐานข้อมูลหลัก หลังจากนั้นซอฟต์แวร์จะส่งภาพที่ค้นมาได้ในขั้นต้นกลับมาให้ผู้ใช้เลือกอีกครั้งเพื่อตรวจสอบและระบุเงื่อนไขในการค้นหาภาพให้แคบลง ซึ่งจะได้ภาพที่ตรงและถูกต้องมากขึ้น ซึ่งในการเลือกภาพในขั้นตอนนี้จำนวนครั้งในการเลือกเพื่อระบุเงื่อนไขและความถูกต้องขึ้นอยู่กับจำนวนภาพและความถูกต้อง ถ้าต้องการให้ซอฟต์แวร์มีความถูกต้องสูงจำนวนครั้งก็จะเพิ่มตามไปด้วย จึงทำให้การค้นหาไม่สะดวกและช้า และในการที่คอมพิวเตอร์จะต้องติดต่อกับผู้ใช้ตลอดเวลานั้นจำเป็นที่จะต้องใช้แบนวิด (bandwidth) ในการส่งข้อมูลภาพกับไปกลับมาระหว่างผู้ใช้งานกับคอมพิวเตอร์ทำให้เกิด traffic ขึ้นในระบบเน็ตเวิร์ค

ดังนั้นจึงจำเป็นที่จะต้องลดความล้าช้าและลดขั้นตอนการที่ผู้ใช้งานจะต้องติดต่อกับซอฟต์แวร์ในฝั่งเซิร์ฟเวอร์ลง เพื่อลด traffic โดยในส่วนที่เป็นการทำงานซ้ำๆ นั้นจะให้ซอฟต์แวร์ในฝั่ง เซิร์ฟเวอร์เป็นผู้กระทำงานซ้ำๆ นั้นแทนผู้ใช้ โดยไม่จำเป็นที่จะต้องส่งข้อมูลกลับมาให้ผู้ใช้ระบุเงื่อนไข เพื่อเป็นการลดขั้นตอนและลดการใช้แบนวิดบนระบบเน็ตเวิร์คลง จึงจำเป็นที่จะต้องสอนให้ คอมพิวเตอร์นั้นรับรู้ในสิ่งที่เป็นเงื่อนไขที่ผู้ใช้ต้องการและสามารถตัดสินใจในการเลือกภาพได้ ซึ่งในการสอนให้คอมพิวเตอร์เรียนรู้นั้นเรียกว่า “Learning Machine” โดยในขั้นตอนแรกนั้นผู้ใช้จะต้อง อินพุตข้อมูลเบื้องต้นให้แก่คอมพิวเตอร์และให้คอมพิวเตอร์ค้นหาและคัดเลือกภาพจากฐานข้อมูลหลัก แล้วหลังจากนั้นการส่งข้อมูลภาพที่มีความสำพันธ์กับภาพที่เป็นเงื่อนไขขั้นต้นกลับมา (Relevance Feedback) ให้ผู้ใช้เลือกอีกครั้งนั้น จะถูกตัดตอนให้เป็นการทำงานของซอฟต์แวร์ที่จำลองเป็นการทำงานของผู้ใช้แทน โดยแทนที่ผู้ใช้จะเป็นผู้ระบุภาพหรือเงื่อนไขที่ต้องการอีกครั้งนั้น จะให้คอมพิวเตอร์เป็นผู้กระทำงานดังกล่าว โดยจะใช้เทคนิคที่เรียกว่า “Neural Network” เข้ามาช่วยในการทำงานนี้ (Learning is Implemented by a Novel Neural Network)

การที่ใช้การทำงานของ Neural Network เข้ามาช่วยนั้น จะเป็นการทำงานที่ฝั่งของเซิร์ฟเวอร์ โดยที่จะไม่มีการส่งข้อมูลภาพกลับมาให้ผู้ใช้เลือก จึงทำให้ไม่ต้องใช้แบนวิดที่สูงอยู่ตลอดเวลาเป็นการลด traffic บนระบบเน็ตเวิร์ค ซึ่งการทำงานของ Neural Network นั้นเมื่อคอมพิวเตอร์ได้รับเงื่อนไขขั้นต้นแล้วไปประมวลผลได้ข้อมูลภาพกลับมา จะถูกส่งให้ Neural

Network เป็นผู้ประมวลผล ซึ่ง Neural Network จะเป็นผู้เก็บเงื่อนไขขั้นต้น และจะใช้ข้อมูลนั้นในการวิเคราะห์เพื่อระบุเงื่อนไขของภาพใหม่ขึ้นมาเพื่อจำกัดขอบเขตของภาพให้แคบลงแทนผู้ใช้แล้วส่งกลับไปให้หน่วยค้นหา (Search Unit) ประมวลผลต่อไปอีกครั้งหนึ่ง การทำงานดังกล่าวอาจเป็นอัตโนมัติ (Fully Automatic) หรือกึ่งอัตโนมัติ (Semi-Automatic RF System)

ซึ่งที่กล่าวมาขั้นต้นนั้นจะทำให้ผู้ใช้ทำงานได้สะดวก และเป็นการลด traffic บนระบบเน็ตเวิร์คทำให้การทำงานง่ายและรวดเร็วขึ้น

1.2 วัตถุประสงค์ของโครงงาน

a) สร้างโปรแกรมค้นหาข้อมูลประเภทรูปภาพ (Search Engine) โดยใช้องค์ประกอบพื้นฐานของรูปภาพ (CBIR) มาเป็นเงื่อนไขในการค้นหาภาพแทนคีย์เวิร์ด (keyword)

b) เพื่อนำทฤษฎีทางด้าน Neural Network และ Machine Intelligent มาสร้างเป็นระบบอัตโนมัติ (Automatic) ในส่วนของ Search Engine เพื่อแก้ปัญหาการใช้แบนวิดมากเกินไปในระบบการติดต่อสื่อสารกับผู้ใช้ (User Interaction Interface)

c) ศึกษาการใช้ภาษา JAVA (Java 2 Enterprise Edition)ในการพัฒนาโปรแกรมประเภท Search Engine และ GUI (Graphic User Interface) บนระบบเครือข่ายอินเตอร์เน็ต ซึ่งเป็น Web-Base Application ภายใต้ การรองรับการใช้งานจากผู้ใช้หลายคน (Multi user)

d) ช่วยลดช่องว่างของการใช้เทคโนโลยีกับผู้ใช้ลง โดยพยายามทำให้การใช้งานมีความสะดวกและรวดเร็วในการค้นหาข้อมูลภาพเป็นหลัก

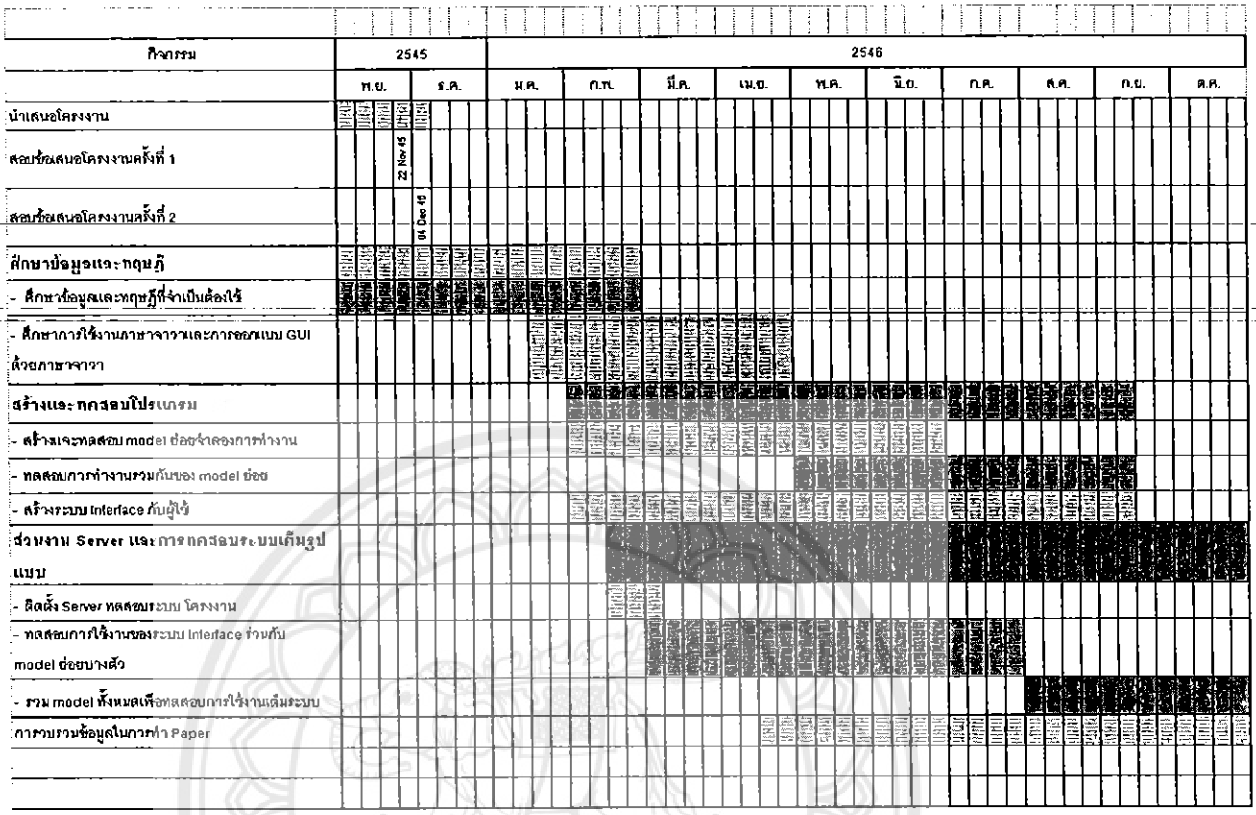
e) เพื่อพัฒนาระบบฐานข้อมูลประเภทรูปภาพที่ใช้งานภายใต้ระบบจัดการแบบอัตโนมัติและกึ่งอัตโนมัติ (Fully Automated image analysis)

1.3 ขอบข่ายของโครงงาน

(a) โครงงานจะใช้ Neural Network และ Machine Intelligent มาสร้าง Search Engine ที่ใช้พื้นฐานของรูปภาพ เพื่อให้ทำงานเป็นแบบการป้อนกลับแบบอัตโนมัติ (Automatic Relevance Feedback) เพื่อลดขั้นการทำงานของผู้ใช้ และลดการใช้งานแบนวิดบนระบบเน็ตเวิร์ค

(b) สร้าง Search Engine ที่มีการติดต่อผู้ใช้แบบ GUI (Graphic User Interface) ด้วยภาษาจาวา (Java 2 Enterprise Edition)ทำงานบนเครือข่ายอินเตอร์เน็ตผ่าน Web-Browser เพื่อให้การใช้งานเป็นแบบผู้ใช้หลายคน ซึ่งจะทำให้มีการใช้งานได้จากผู้ใช้หลายคนพร้อมกัน

1.4 ขั้นตอนการดำเนินงาน



1.5 ผลที่คาดว่าจะได้รับ

(a) แอพพลิเคชั่น (Application) นี้เป็น web-base application ซึ่งพัฒนาบน JAVA platform (Java 2 Enterprise Edition) จึงทำให้โครงงานนี้สามารถรองรับการใช้งานจากผู้ใช้หลายคนผ่านระบบเครือข่ายอินเตอร์เน็ตและแอพพลิเคชั่นนี้ไม่ขึ้นกับแพลตฟอร์ม (platform) ในการเรียกใช้งานแอพพลิเคชั่น จึงทำให้สามารถเรียกใช้งานได้จากทุกระบบปฏิบัติการ (Multi Operating System)

(b) แอพพลิเคชั่น (Application) เป็นแอพพลิเคชั่นบนอินเตอร์เน็ตการใช้งานจึงไม่จำกัดขอบเขตเฉพาะเครือข่ายในองกรเท่านั้น ทำให้การใช้ข้อมูลเป็นไปในวงกว้างเป็นการกระจายการใช้ข้อมูลอย่างแท้จริง

1.6 งบประมาณที่ใช้

1.6.1 ค่าใช้จ่ายในการซื้อหนังสือ	1,200	บาท
1.6.2 ค่าใช้จ่ายในการทำรายงาน	400	บาท
1.6.3 ค่าใช้จ่ายเบ็ดเตล็ด	400	บาท
รวมทั้งสิ้น	2,000	บาท

มหาวิทยาลัยนเรศวร

บทที่ 2
งานวิจัยที่เกี่ยวข้อง

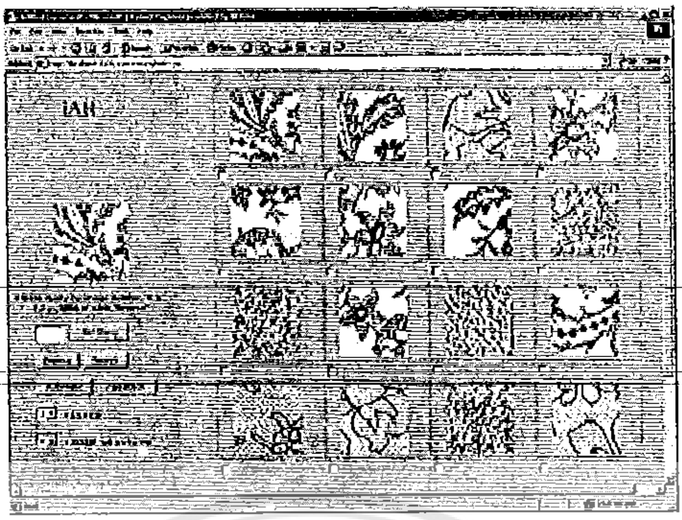
จากที่กล่าวมาบ้างแล้วในบทที่ 1 ในเรื่องของการค้นหาภาพโดยการใช้คีย์เวิร์ดเป็นตัวชี้ (index) ในการค้นหา ซึ่งอาจจะเป็นชื่อสถานที่ ชื่อวัตถุ สีของภาพหรือสิ่งที่บ่งบอกลักษณะเฉพาะของภาพ การใช้คีย์เวิร์ดในการค้นหาภาพนั้นทำได้สะดวกและง่ายต่อการเข้าใจของผู้ใช้แต่จะเห็นได้ว่ายังมีข้อเสียอยู่ก็คือ ผู้ใช้กับผู้ดูแลระบบจะใช้ข้อมูลที่ไม่ตรงกันในการค้นหาภาพ ดังนั้นจึงได้มีการแก้ปัญหาที่เกิดขึ้นดังนี้

2.1 คุณสมบัติพื้นฐานของรูปภาพ (Content-based image retrieval)

คือ คุณสมบัติขั้นพื้นฐานของภาพประกอบด้วยรูปร่าง (shape) ,สี (color) ,พื้นผิว (texture) คุณสมบัติเหล่านี้ใช้ในการอธิบายองค์ประกอบหรือลักษณะของภาพเพื่อบอกให้เข้าใจว่าภาพนั้นเป็นภาพอะไรแต่ในทางคอมพิวเตอร์แล้วพื้นฐานของรูปภาพไม่สามารถจะอธิบายให้คอมพิวเตอร์ทราบได้ว่าภาพนั้นเป็นภาพอะไร เช่น ภาพรถสีน้ำเงินจอดอยู่บนสนามหญ้า คอมพิวเตอร์ไม่สามารถจะบอกได้ว่าสิ่งที่อยู่ในภาพเป็นรถ แต่สามารถอธิบายได้ว่าเป็นรูปสี่เหลี่ยมสีน้ำเงินซ้อนทับอยู่บนพื้นสีเขียว ซึ่งจากลักษณะดังนี้เราจะนำคุณสมบัติเหล่านี้มาใช้ ในการทำ image indexing ซึ่งคุณสมบัติเหล่านี้เราเรียกว่า “low-level feature” เพื่อใช้ในการเปรียบเทียบรูปภาพว่ามีลักษณะเหมือนหรือแตกต่างกันอย่างไร

2.2 การค้นหาภาพโดยใช้ระบบการติดต่อสื่อสารกับผู้ใช้ในพื้นฐานของภาพหรือหลักการค้นหาแบบป้อนกลับ (Relevance Feedback)

จะเป็นการแก้ปัญหาที่เกิดขึ้นจากการใช้คีย์เวิร์ด (keyword) ในการค้นหาภาพ ซึ่งจะใช้ GUI (Graphic User Interface) เข้ามาช่วยแก้ปัญหา โดยสร้างการโต้ตอบกันระหว่างโปรแกรมกับผู้ใช้ เพื่อให้โปรแกรมนั้นรับข้อมูลจากผู้ใช้ โปรแกรมจะนำรูปที่ผู้ใช้ต้องการไปทำการตรวจสอบแล้วส่งกลับให้ผู้ใช้เลือกอีกครั้งหนึ่ง ความถูกต้องขึ้นอยู่กับจำนวนการตอบโต้ระหว่างซอฟต์แวร์กับผู้ใช้ เราเรียกการตอบโต้นี้ว่า RF (Relevance Feedback)[1] จะเห็นว่าในการประมวลผลแต่ละครั้งจะเป็นการใช้งานแบนวิด (bandwidth) เป็นจำนวนมากทำให้เกิด Traffic บนระบบเน็ตเวิร์คและทำให้ผู้ใช้จำเป็นที่ต้องคอยตอบคำถามจากซอฟต์แวร์ตลอดเวลาโดยไม่จำเป็น



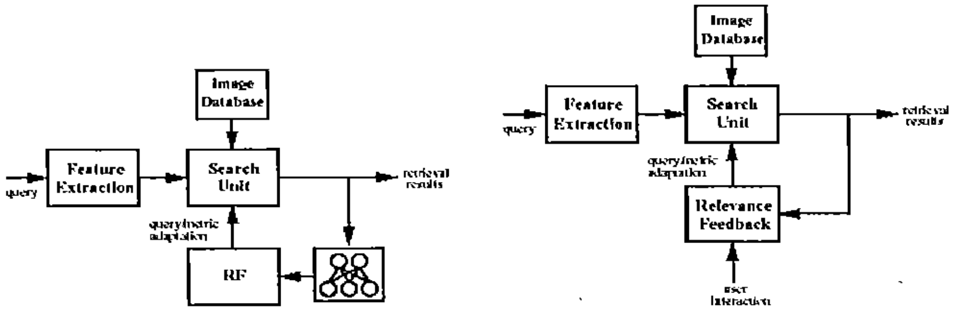
(a)

รูปที่ 2.1 หน้าจอการที่ซอฟต์แวร์โต้ตอบกับผู้ใช้

จากรูปที่ 2.1 นั้นผู้ใช้จะต้องทำเครื่องหมายใต้รูปที่ต้องการ และซอฟต์แวร์จะนำไปประมวลผลแล้วส่งกลับมาให้ผู้ใช้ในแต่ละครั้ง[1]

2.3 การค้นหารูปภาพแบบอัตโนมัติโดยใช้ CBIR (Automatic Machine Interaction for CBIR Using a Self-Organizing Tree map)

จากปัญหาที่เกิดขึ้นในหัวข้อ 2.2 นั้น สามารถปัญหาได้โดยการลด Traffic ที่เกิดจากการติดต่อระหว่างซอฟต์แวร์กับผู้ใช้ลง โดยการที่จะลดการใช้แบนวิดนี้ได้นั้นจำเป็นที่จะต้องทำให้ซอฟต์แวร์นี้มีความฉลาดที่จะกระทำการเองแทนผู้ใช้งานได้ ดังนั้นเราจะสร้างกระบวนการทำงานที่จะให้ซอฟต์แวร์รับอินพุต (Input) จากผู้ใช้ในขั้นแรกและทำงานจนเสร็จด้วยตัวของซอฟต์แวร์เอง ในการทำงานดังกล่าวจะต้องใช้เทคโนโลยีที่เรียกว่า “Neural network” [2]เข้ามาช่วย จะเห็นได้จากภาพ โดยรูป 2.1(a) การทำงานระบบที่มี Neural network ช่วยในการประมวลผลทำให้ไม่ต้องส่งข้อมูลกลับไปให้ผู้ใช้เลือก แต่ในรูปที่ 2.1 (b) จะเป็นระบบที่ไม่มี Neural network ดังนั้นการที่ซอฟต์แวร์จะเรียนรู้ได้นั้นจะต้องรับผ่านผู้ใช้ จึงต้องส่งข้อมูลกลับไปให้ผู้ใช้อีกครั้งทำให้ต้องใช้แบนวิดในการส่งข้อมูลมาก



(a) (b)

รูปที่ 2.2 การค้นหารูปภาพแบบอัตโนมัติโดยใช้ CBIR

2.4 SOTM (Self-Organization Tree Map)

จากปัญหาที่กล่าวมาแล้วในขั้นต้น ซึ่งพูดถึงปัญหาของ traffic และปัญหาของซอฟต์แวร์ในการติดต่อกับผู้ใช้ ซึ่งทั้ง 2 ปัญหาจะทำให้ต้องใช้แบนวิดเป็นจำนวนมาก เนื่องจากผู้ใช้ (user) ต้องเป็นผู้คัดเลือก (classified) รูปภาพ เพื่อส่งให้คอมพิวเตอร์ในการประมวลผล เราจึงจะตัดความจำเป็นในการใช้ผู้ใช้เป็นผู้คัดเลือกออก โดยใช้อัลกอริทึมของ SOTM มาใช้ในการทำงานของผู้ใช้ โดยอัลกอริทึมนี้จะเป็นผู้จำแนกภาพ (image) ออกเป็น 2 กลุ่มให้กับคอมพิวเตอร์ คือ ภาพที่ถูกต้อง (relevant) กับภาพที่ไม่ถูกต้อง (non-relevant) ส่งให้กับ relevance feedback โดยจะกล่าวถึงในบทที่ 3 ต่อไป

รูปที่ 2.3 การจัดกลุ่มของ SOTM ระหว่าง 1 กลุ่มกับ 2 กลุ่ม

a) Single-class approach b) Multi-class approach

บทที่ 3

Neural Network แบบไม่มีผู้สอนโครงสร้างแบบต้นไม้ (SOTM)

ปัญหาที่พบในบทที่ 2 ในเรื่องของการใช้การติดต่อสื่อสารกับผู้ใช้ (User interface) ในพื้นฐานของรูปภาพ (CBIR) และ Automatic Machine Interaction for CBIR นั้น สามารถแก้ไขได้โดยใช้ SOTM เข้ามาช่วยในการแก้ปัญหา ซึ่งจะมีความสามารถในการเรียนรู้ โดยปราศจากผู้สอนภายนอก

ในระบบ Neural network แบบไม่ต้องใช้ผู้ใช้ (unsupervised) จะถูกใช้สำหรับการแบ่งหมวดหมู่ของ relevance ใน interactive retrieval procedure จะนำ SOTM [1],[2] มาใช้ SOTM เป็น hierarchical ที่ใช้ค้นหาที่ตั้งของกลุ่มศูนย์กลาง (cluster center) ซึ่งระบุ topological relations จาก input space และสร้างกลุ่มของข้อมูลที่แตกต่างกันขึ้น

3.1 Neural Network แบบไม่มีผู้สอนโครงสร้างแบบต้นไม้ (Self-Organizing Tree Map)

SOTM หรือ SELF ORGANIZING TREE MAP นั้นเป็นอัลกอริทึมอย่างหนึ่ง ซึ่งจะทำหน้าที่ค้นหาจุดศูนย์กลางของกลุ่ม vector โดยการ map และค้นหาว่าในกลุ่มเวคเตอร์ตัวอย่างที่นำเข้าไปประมวลผลนั้น มีอยู่กี่กลุ่ม และมีจุดศูนย์กลางอยู่ที่ใดบ้าง

การแยกประเภทเวคเตอร์อินพุตที่จับกันเป็นกลุ่มก้อนและหาจุดศูนย์กลางของแต่ละกลุ่ม ซึ่งศูนย์กลางที่แน่นอนจะถูกสร้างโดยเวคเตอร์อินพุตที่แตกต่างกัน ประเภทที่แตกต่างกันของภาพที่ถูกต้อง (relevant) กับภาพที่ไม่ถูกต้อง (non-relevant) ก็สามารถทำให้เกิดศูนย์กลางของพวกมันเอง โดยทำการ map กัน ดังนั้นโมเดลที่คล้ายกันจะอยู่ใกล้กันและ model ที่แตกต่างกันจะอยู่ไกลกัน

3.1.1 อัลกอริทึมของ SOTM

กำหนดให้ training vector set = $\{x_1,..,x_k,..,x_K\}$

1. <Initialization>

ทำการเลือก root node $\{w_i\}$ จากเซตที่หาได้จาก input vector $\{x_k\}$ ในการสุ่มตัวอย่าง

2. <Similarity matching>

ทำการสุ่มเลือกจุด x อันใหม่ และค้นหา winning node ที่ดีที่สุด ที่เวลา t โดยใช้ Euclidean criterion หาระยะทางที่น้อยที่สุด

3. <Updating>

ถ้า || x(t)-wj* || < H(t) เมื่อ H(t) เป็นฟังก์ชันที่ใช้ในการควบคุมลำดับของ tree ทำตามฟังก์ชันนี้ไปเรื่อยๆ ตั้งแต่ x(t) ไปจนถึงกลุ่มที่ jth เพื่อจัดกลุ่ม vector

$$W_j(t+1) = W j(t) + \alpha(t)[x(t) - W_j(t)] \tag{3.1}$$

เมื่อ $\alpha(t)$ เป็นอัตราการเรียนรู้ ซึ่งเมื่อเข้าใกล้ศูนย์กลางค่าของ $\alpha(t)$ จะมีค่าลดลงเรื่อยๆ โดยจะแปรผันไปตามเวลา $0<\alpha(t)\leq 1$ แต่ถ้า || x(t)-wj* || > H(t) มันจะจัดกลุ่มใหม่เป็นอีกกลุ่มหนึ่ง

4. <Continuation>

ทำจากข้อ 2 ไปเรื่อยๆ จนกระทั่งไม่มีการเปลี่ยนแปลงใน feature map ที่เห็นได้ชัด

เป็นที่ทราบกันดีอยู่แล้วว่า ถ้าเราสามารถแยกประเภทเวคเตอร์อินพุตที่จับกันเป็นกลุ่มก้อนและหาศูนย์กลางของแต่ละกลุ่ม ซึ่งศูนย์กลางที่แน่นอนถูกสร้างโดยเวคเตอร์ที่แตกต่างกันประเภทที่แตกต่างกันของภาพที่ถูกต้อง (relevant) กับภาพที่ไม่ถูกต้อง (non-relevant) ก็สามารถทำให้เกิดศูนย์กลางชองพวกมันเองได้ ดังนั้นโมเดลที่คล้ายกันจะใกล้กันและโมเดลที่แตกต่างกันจะอยู่ไกลกัน

แต่การที่จะหาว่าเวคเตอร์นั้นจับกลุ่มกันหรือไม่และมีจะศูนย์กลางอยู่ที่ตำแหน่งใดนั้นจะต้องอาศัย SOTM Algorithm ซึ่งย่อมาจาก SELF ORGANIZING TREE MAP เข้าช่วยอัลกอริทึมของ SOTM นั้น จะทำหน้าที่ในการค้นหาจุดศูนย์กลางของกลุ่มเวคเตอร์ และค้นหาว่าในกลุ่มของตัวอย่างเวคเตอร์ที่นำเข้าไปประมวลผลนั้น มีอยู่กี่กลุ่ม และมีจุดศูนย์กลางอยู่ที่ใดบ้าง

SOTM เป็นส่วนขยายของ self-organizing map สามารถแยกประเภทเวคเตอร์ ที่จับกันเป็นกลุ่มและหาศูนย์กลางของแต่ละกลุ่ม ซึ่งศูนย์กลางของแต่ละกลุ่มถูกสร้างโดยที่แตกต่างกันตามประเภทที่แตกต่างกันของ relevant กับ non-relevant image และทำให้เกิดศูนย์กลางของมันเองใน tree map หรืออีกนัยหนึ่งคือ map ดังนั้นโมเดลที่คล้ายกันจะอยู่ใกล้กันและโมเดลที่แตกต่างกันจะอยู่ไกลกัน

เพื่อให้เข้าใจในหลักการหลักการของ SOTM มากขึ้น เราจะแบ่งการทดลองออกเป็นสองส่วนโดยส่วนแรกจะเป็นการทำงานของ SOTM กับกลุ่มเวคเตอร์ตัวอย่างเพียงกลุ่มเดียว และส่วนที่สองจะเป็นการทำงานของ SOTM กับ กลุ่มตัวอย่างสองกลุ่ม

3.2 การทดลองเบื้องต้น

3.2.1 การทดลองที่ 1 (การใช้ SOTM หาจุดศูนย์กลางของกลุ่มเวคเตอร์กลุ่มเดียว)

ขั้นตอนการทำงาน

1 กำหนดกลุ่มเวคเตอร์ที่เป็นกลุ่มตัวอย่าง

2 เราจะสุ่มเวคเตอร์จากตัวอย่างมาหนึ่งตัว เพื่อใช้เป็น จุดศูนย์กลางเริ่มต้น ต่อไปจะเรียกว่า w(t) และนำไปคำนวณหาจุดศูนย์กลางถัดไป

3 สุ่มเลือกค่าจากตัวอย่างมาอีกหนึ่งตัว ต่อไปจะเรียกว่า x(t) เพื่อใช้ในการคำนวณ นำมาคำนวณร่วมกัน w โดยใช้สมการต่อไปนี้

$$w(t+1)=w(t)+a(t)[x(t)-w(t)]$$

4 จากสมการ a(t) จะเปลี่ยนแปลตลอด โดยคำนวณจาก

$$a(t+1) = a(t)/(1+a(t))$$

5 เราจะได้ w(t) ตัวใหม่ขึ้นมา และ w(t) ตัวใหม่ที่ได้จะเคลื่อนเข้าหาจุดศูนย์กลางของกลุ่ม

6 และวนกลับไปทำใน ข้อ 3 ถึง ข้อ 6 อีกครั้งหนึ่ง จนกว่า w(t) จะไม่มีการเปลี่ยนแปลหรือ มีการเปลี่ยนแปลน้อยมากเราจึงจะหยุด

รูปที่ 3.1 การใช้ SOTM หาจุดศูนย์กลางของกลุ่มเวคเตอร์กลุ่มเดียว

ตารางที่ 3.1 กลุ่มตัวอย่างในการทดสอบ

x	x	5	3	5	3	7	3	5	5	7	7	5
	y	5	2	3	4	4	3	4	2	2	3	1

ตารางที่ 3.2 ค่า a(t) ที่คำนวณได้

a(t)	1	2	3	4	5	6	7	8	9	10	11	12
	0.9550	0.4885	0.3282	0.2471	0.1981	0.1654	0.1419	0.1243	0.1105	0.0995	0.0905	0.0830

ค่า a(t) ที่คำนวณได้ และจะใช้ในการคำนวณหาค่าศูนย์กลาง

ตารางที่ 3.3 ค่าหลังจากที่ได้ผ่านการทดสอบแล้ว

x		w		
x	y	x	y	a(t)
5	5	5.0000	1.0000	0.90000
3	2	3.2000	1.9000	0.47368
5	3	4.0526	2.4211	0.32143
3	4	3.7143	2.9286	0.24324
7	4	4.5135	3.1892	0.19565
3	3	4.2174	3.1522	0.16364
5	4	4.3455	3.2909	0.14063
5	2	4.4375	3.1094	0.12329
7	2	4.7534	2.9726	0.10976
7	3	5.0000	2.9756	0.09890

นำกลุ่มตัวอย่างมาผ่าน SOTM ผลที่ได้จะอยู่ในแถวของ w(t) จะเห็นว่า w(t) ที่เกิดขึ้นจะมีการเปลี่ยนแปลงเข้าหา 5 และ 3 ซึ่งเป็นจุดศูนย์กลางของกลุ่มดังตารางที่ 3.3 ซึ่งอัตราการเปลี่ยนแปรจะขึ้นอยู่กับ a(t)

```
class Value {
        public static void main(String args[])
        {
        double Wx[] = new double[12];        double Wy[] = new double[12];
        double Xx[] = new double[12];        double Xy[] = new double[12];
        double A[] = new double[12];

        A[0]=.9550;    //กำหนดค่า กลุ่มตัวอย่าง
        Xx[0]=5;       Xy[0]=5;       Xx[1]=3;       Xy[1]=2;
        Xx[2]=5;       Xy[2]=3;       Xx[3]=3;       Xy[3]=4;
        Xx[4]=7;       Xy[4]=4;       Xx[5]=3;       Xy[5]=3;
        Xx[6]=5;       Xy[6]=4;       Xx[7]=5;       Xy[7]=2;
        Xx[8]=7;       Xy[8]=2;       Xx[9]=7;       Xy[9]=3;
        Xx[10]=5;      Xy[10]=1;

        for (int x = 1; x < 11; x++)
        {
                A[x]=A[x-1]/(1+A[x-1]);      //ใช้คำนวณค่า a(t)
        }
        Wx[0]=5;       Wy[0]=1;


        for(int i=1;i<11;i++)     //Loop ใช้คำนวณหาจุดศูนย์กลาง
        {
                Wx[i] = Wx[i-1]+A[i]*(Xx[i-1] - Wx[i-1]);

                        Wy[i] = Wy[i-1]+A[i]*(Xy[i-1] - Wy[i-1]);
        }

        for(int i=0;i<11;i++)     // Loop ใช้แสดงค่า ที่คำนวณได้
        {
                System.out.println(Wx[i]+ "," +Wy[i]);
        }
    }
```

รูปที่ 3.2 Java code สำหรับ SOTM

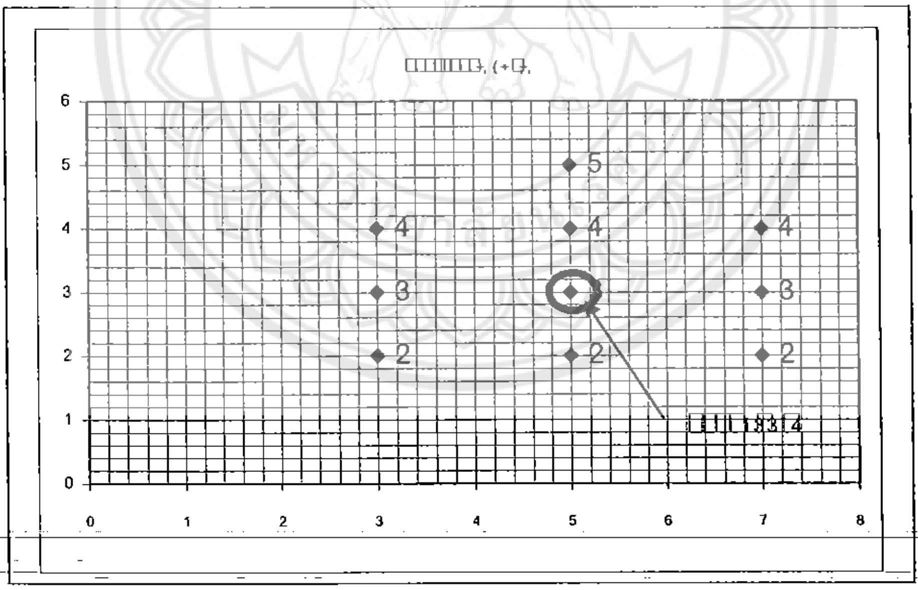
```
G:\java-work\pj>java Value
Wx(0) = 5.0, Wy(0) = 1.0
Wx(1) = 5.0, Wy(1) = 2.9539641943734014
Wx(2) = 4.343642611683849, Wy(2) = 2.640893470790378
Wx(3) = 4.50582147477361, Wy(3) = 2.7296248382923674
Wx(4) = 4.20746887966805, Wy(4) = 2.9813278008298756
Wx(5) = 4.669264069264069, Wy(5) = 3.1497835497835496
Wx(6) = 4.432392273402675, Wy(6) = 3.1285289747399703
Wx(7) = 4.502927781392323, Wy(7) = 3.236824983734548
Wx(8) = 4.55787037037037, Wy(8) = 3.100115740740741
Wx(9) = 4.800937988535695, Wy(9) = 2.9906201146430433
Wx(10) = 5.0, Wy(10) = 2.9914691943127965

G:\java-work\pj>
```

รูปที่ 3.3 เอาต์พุต (Output) ที่ได้จาก Source Program Java

3.2.2 สรุปผลการทดลองที่ 1

จากการทดลองดังรูปที่ 3.4 ได้ผลปรากฏว่า w(t) จะเข้าใกล้จุดศูนย์กลางของกลุ่ม ซึ่งค่าที่เปลี่ยนแปลงนั้น จะขึ้นอยู่กับจำนวนครั้งที่เราคำนวณ โดยจากการทดลองใช้จำนวนครั้งทั้งหมด 10 ครั้ง ผลที่ได้จึงจะเข้าถึงจุดศูนย์กลางของกลุ่ม



รูปที่ 3.4 ผลของจุดศูนย์กลางของกลุ่มตัวอย่าง

3.2.3 การทดลองที่ 2 (การใช้ SOTM ในหารหาจุดศูนย์กลางของกลุ่มแบบสองกลุ่ม)

ขั้นตอนการทำงาน

1 กำหนดกลุ่มเวคเตอร์ที่เป็นกลุ่มตัวอย่าง

2 เราจะสุ่มเวคเตอร์จากตัวอย่างมาสองตัว เพื่อใช้เป็น จุดศูนย์กลางเริ่มต้นต่อไปจะเรียกว่า w1(t) และ w2(t) และนำไปคำนวณหาจุดศูนย์กลางถัดไป

3 สุ่มเลือกค่าจากตัวอย่างมาอีกหนึ่งตัว ต่อไปจะเรียกว่า x(t) เพื่อใช้ในการคำนวณ

4 นำมาคำนวณร่วมกัน w1(t) และ w2(t) เพื่อหา d1 และ d2 ซึ่ง d1 และ d2 คือ ระยะหาง จาก x(t) ถึง w1(t) และจาก x(t) ถึง w2(t) โดยใช้สมการต่อไปนี้

$$D1 = |x(t) - w1(t)|$$

$$D2 = |x(t) - w2(t)|$$

เราจะได้ระยะทางจากจุด x(t) ถึงจุดศูนย์กลางมากสองตัว

5 จากนั้นเราจะนำ d1 และ d2 มาเทียบกัน โดยที่จะเป็นไปตามเงื่อนไขดังนี้

```
If d1 < d2 then
   เข้ากระทำในขั้นตอนที่ 6.1
Else if d2 < d1 then
   เข้ากระทำในขั้นตอนที่ 6.2
End if
```

6 ขั้นตอนนี้จะเป็นการหาจุดศูนย์กลางของกลุ่มโดยจะแบ่งเป็นสองข้อ คือ

6.1 หาจุดศูนย์กลางของกลุ่มที่หนึ่ง โดยใช้สมการดังนี้

$$W1(i+1)=w1(i)+a(t)[x(t)-w1(i)] \quad ;i=0,1,2,\ldots,n$$

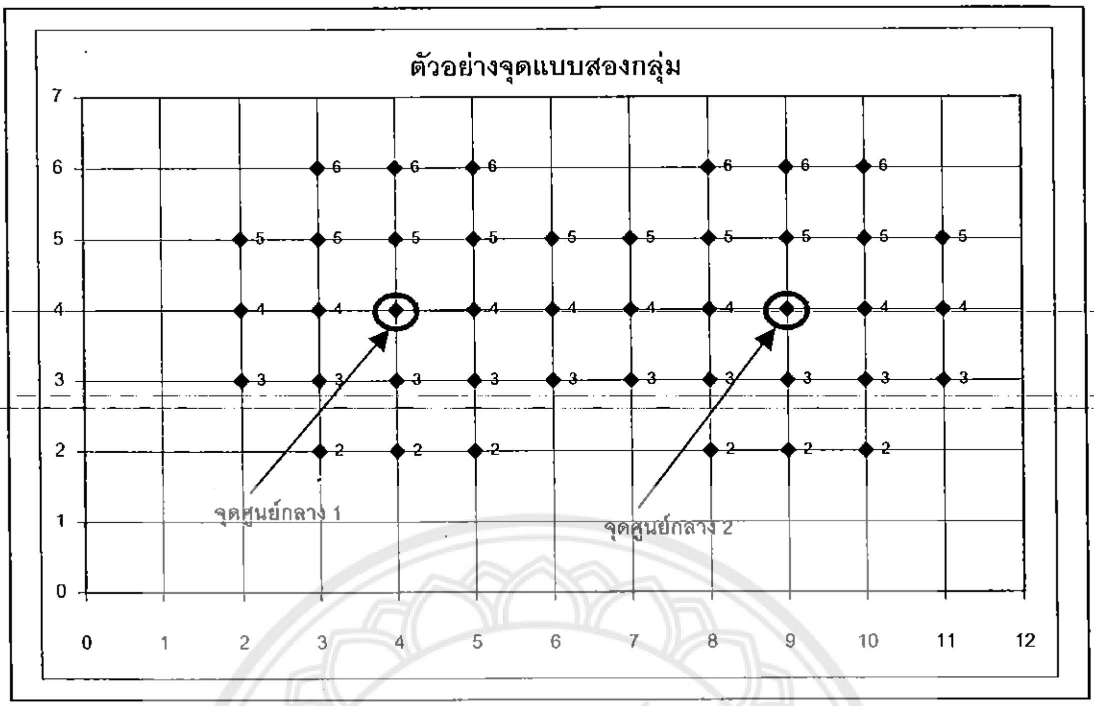
6.2 หาจุดศูนย์กลางของกลุ่มที่สอง

$$W2(k+1)=w2(i)+a(t)[x(t)-w2(i)] \quad ;k=0,1,2,\ldots,n$$

โดยที่ a(t) ทั้งสองหาได้จาก $a(t+1)=a(t)/(1+a(t))$

7 และวนกลับไปทำใน ข้อ 3 ถึง ข้อ 6 อีกครั้งหนึ่ง จนกว่า w(t) จะไม่มีการเปลี่ยนแปลหรือมีการเปลี่ยนแปลน้อยมากเราจึงจะหยุด

รูปที่ 3.5 ขั้นตอนการทำงานของ SOTM ในการหาจุดศูนย์กลางของกลุ่มสองกลุ่ม



รูปที่ 3.6 ตัวอย่างการใช้ SOTM ในหารหาจุดศูนย์กลางของกลุ่มแบบสองกลุ่ม

```
Xx[0]=2;    Xy[0]=5;    Xx[1]=2;    Xy[1]=4;
Xx[2]=2;    Xy[2]=3;    Xx[3]=3;    Xy[3]=6;
Xx[4]=3;    Xy[4]=4;    Xx[5]=3;    Xy[5]=3;
Xx[6]=3;    Xy[6]=2;    Xx[7]=4;    Xy[7]=6;
Xx[8]=4;    Xy[8]=5;    Xx[9]=4;    Xy[9]=4;
Xx[10]=4;   Xy[10]=3;   Xx[11]=4;   Xy[11]=2;
Xx[12]=5;   Xy[12]=6;   Xx[13]=5;   Xy[13]=5;
Xx[14]=5;   Xy[14]=4;   Xx[15]=5;   Xy[15]=3;
Xx[16]=5;   Xy[16]=2;   Xx[17]=6;   Xy[17]=5;
Xx[18]=6;   Xy[18]=4;   Xx[19]=6;   Xy[18]=3;
Xx[40]=7;   Xy[40]=5;   Xx[20]=7;   Xy[20]=4;
Xx[21]=7;   Xy[21]=3;   Xx[22]=8;   Xy[22]=6;
Xx[23]=8;   Xy[23]=5;   Xx[24]=8;   Xy[24]=4;
Xx[25]=8;   Xy[25]=3;   Xx[26]=8;   Xy[26]=2;
Xx[27]=9;   Xy[27]=6;   Xx[28]=9;   Xy[28]=5;
Xx[29]=9;   Xy[29]=4;   Xx[30]=9;   Xy[30]=3;
Xx[31]=9;   Xy[31]=2;   Xx[32]=10;  Xy[32]=6;
Xx[33]=10;  Xy[33]=5;   Xx[34]=10;  Xy[34]=4;
Xx[35]=10;  Xy[35]=3;   Xx[36]=10;  Xy[36]=2;
Xx[37]=11;  Xy[37]=5;   Xx[38]=11;  Xy[38]=4;
Xx[39]=11;  Xy[39]=3;
```

รูปที่ 3.7 กลุ่มตัวอย่างที่ใช้ในการหาจุดศูนย์กลางของกลุ่มแบบสองกลุ่ม

โปรแกรมในภาคผนวก ก เป็นโปรแกรมภาษา Java ที่ใช้ในการคำนวณหาจุดศูนย์กลางของกลุ่มซึ่งผลที่ได้จะเป็นรูปที่ 3.8 ซึ่ง ค่า w(t) จะถูกแยกออกจากกันเป็นสองกลุ่ม คือ w1(t) และ w2(t)

```
Wx1 = 3.0,   Wy1 = 6.0
Wx1 = 2.526315789473684,   Wy1 = 5.526315789473684
Wx1 = 2.357142857142857,   Wy1 = 5.035714285714286
Wx1 = 2.27027027027027,   Wy1 = 4.54054054054054
Wx1 = 2.4130434782608696,   Wy1 = 4.826086956521739
Wx1 = 2.5090909090909093,   Wy1 = 4.690909090909091
Wx1 = 2.578125,   Wy1 = 4.453125
Wx1 = 2.6301369863013697,   Wy1 = 4.1506849315068495
Wx1 = 2.7804878048780486,   Wy1 = 4.353658536585366
Wx1 = 2.901098901098901,   Wy1 = 4.417582417582418
Wx1 = 3.0,   Wy1 = 4.38
Wx1 = 3.0825688073394497,   Wy1 = 4.26605504587156
Wx1 = 3.152542372881356,   Wy1 = 4.093220338983051
Wx1 = 3.283464566929134,   Wy1 = 4.228346456692914
Wx1 = 3.3970588235294117,   Wy1 = 4.279411764705883
Wx1 = 3.496551724137931,   Wy1 = 4.262068965517242
Wx1 = 3.5844155844155843,   Wy1 = 4.188311688311689
Wx1 = 3.662576687116564,   Wy1 = 4.067484662576688
Wx1 = 3.784883720930232,   Wy1 = 4.116279069767443
Wx1 = 3.895027624309392,   Wy1 = 4.060773480662984
Wx1 = 3.994736842105263,   Wy1 = 3.8684210526315796
Wx1 = 3.9497487437185925,   Wy1 = 3.9195979899497493
Wx1 = 0.0,   Wy1 = 0.0
Wx1 = 0.0,   Wy1 = 0.0
Wx1 = 0.0,   Wy1 = 0.0
```

รูปที่ 3.8 ผลเอาต์พุต (Output) ที่ได้จาก Source Program Java

```
Wx2 = 9.0,   Wy2 = 5.0
Wx2 = 8.052631578947368,   Wy2 = 4.526315789473684
Wx2 = 7.714285714285714,   Wy2 = 4.035714285714286
Wx2 = 7.783783783783784,   Wy2 = 4.513513513513513
Wx2 = 7.826086956521739,   Wy2 = 4.608695652173913
Wx2 = 7.8545454545454545,   Wy2 = 4.509090909090909
Wx2 = 7.875,   Wy2 = 4.296875
Wx2 = 7.890410958904109,   Wy2 = 4.013698630136986
Wx2 = 8.012195121951219,   Wy2 = 4.2317073170731705
Wx2 = 8.10989010989011,   Wy2 = 4.3076923076923075
Wx2 = 8.19,   Wy2 = 4.28
Wx2 = 8.256880733944953,   Wy2 = 4.174311926605505
Wx2 = 8.313559322033898,   Wy2 = 4.008474576271187
Wx2 = 8.433070866141732,   Wy2 = 4.149606299212599
Wx2 = 8.536764705882351,   Wy2 = 4.205882352941177
Wx2 = 8.62758620689655,   Wy2 = 4.1931034482758625
Wx2 = 8.707792207792206,   Wy2 = 4.123376623376624
Wx2 = 8.779141104294476,   Wy2 = 4.006134969325154
Wx2 = 8.8953488372093,   Wy2 = 4.058139534883721
Wx2 = 8.999999999999996,   Wy2 = 4.055248618784531
```

รูปที่ 3.8 (ต่อ) ผลเอาต์พุต (output) ที่ได้จาก Source Program Java

3.2.4 สรุปผลการทดลองที่ 2

จากการทดลองดังรูปที่ 3.9 ได้ผลปรากฏว่า w(t) ทั้งสองจะเคลื่อนเข้าใกล้จุดศูนย์กลางของกลุ่ม ซึ่งค่าที่เปลี่ยนแปลงนั้น จะขึ้นอยู่กับค่า a(t) และจำนวนครั้งที่เราคำนวณ โดยจากการทดลองใช้จำนวนตัวอย่างทั้งหมด 40 ตัวอย่าง และจำนวนครั้งทั้งหมด 20 ครั้งต่อกลุ่ม ผลที่ได้จึงจะเข้าถึงจุดศูนย์กลางของกลุ่ม ตามรูปที่ 3.9

3.3 สรุปผลการทดลอง

จากการทำการทดลองขั้นต้น เราสามารถที่จะจำแนกจุดศูนย์การของกลุ่มข้อมูลที่เราสนใจได้โดยแบ่งออกเป็นสองกลุ่ม ซึ่งในการจำแนกจุดศูนย์กลางของกลุ่มนั้นเราจะนำไปใช้ในการจำแนกกลุ่มที่เราสนใจหรือกลุ่มตัวอย่างของภาพได้ ซึ่งในการทดลองนี้เพื่อให้ง่ายในการทำและในการทำความเข้าใจ เราจึงได้ทำการทดลอง โดยใช้เวคเตอร์เพียงสองมิติเท่านั้น ซึ่งในความเป็นจริงแล้วการนำไปใช้งานจะต้องมีจำนวนของมิติมากกว่านี้

จากการทดลองที่ผ่านมาทั้งหมด ทำให้เราได้เข้าใจหลักการทำงานของ SOTM ว่า SOTM นั้นมีการทำงานอย่างไร และสามารถที่จะนำหลักการทำงานดังกล่าวไปใช้ในการทำการโครงงานของเราได้อย่างไร

บทที่ 4

วิธีดำเนินการวิจัย

ระบบฐานข้อมูลภาพแบบอัตโนมัติโดยใช้ Neural Network นั้นเป็นซอฟต์แวร์ประเภทค้นหารูปภาพ โดยใช้ CBIR เป็นข้อมูลในการทำงาน และมีระบบ SOTM ใช้ในการทำให้ซอฟต์แวร์สามารถทำงานในแบบอัตโนมัติได้ ส่วนของการค้นหานั้นมี Relevance Feedback เป็นตัวกระทำการ

ดังที่กล่าวมาแล้วจะแบ่งส่วนของการทำงานออกเป็น 3 ส่วน คือ ส่วนของการทำ Image Indexing คือส่วนของการทำข้อมูลเพื่อเก็บลงในฐานข้อมูลเพื่อใช้ในการค้นหา

4.1 ขั้นตอนการทำ image indexing

แบ่งได้ 2 ส่วน ตามความละเอียดของข้อมูล

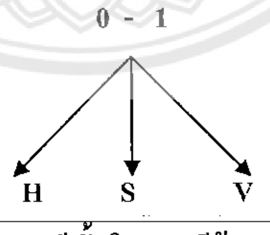
4.1.1 ชนิด 64 บิน เป็น Color histogram โดยใช้ model สี แบบ HSV เป็นข้อมูลพื้นฐาน

4.1.2 ชนิด 115 บิน ประกอบด้วยข้อมูล 3 ลักษณะ คือ สี (color) , พื้นผิว (texture) , รูปร่าง (shape) รวมกันเป็นข้อมูลพื้นฐาน

1) ชนิด 64 บิน เป็น Color histogram โดยใช้ model สี แบบ HSV เป็นข้อมูล

HSV

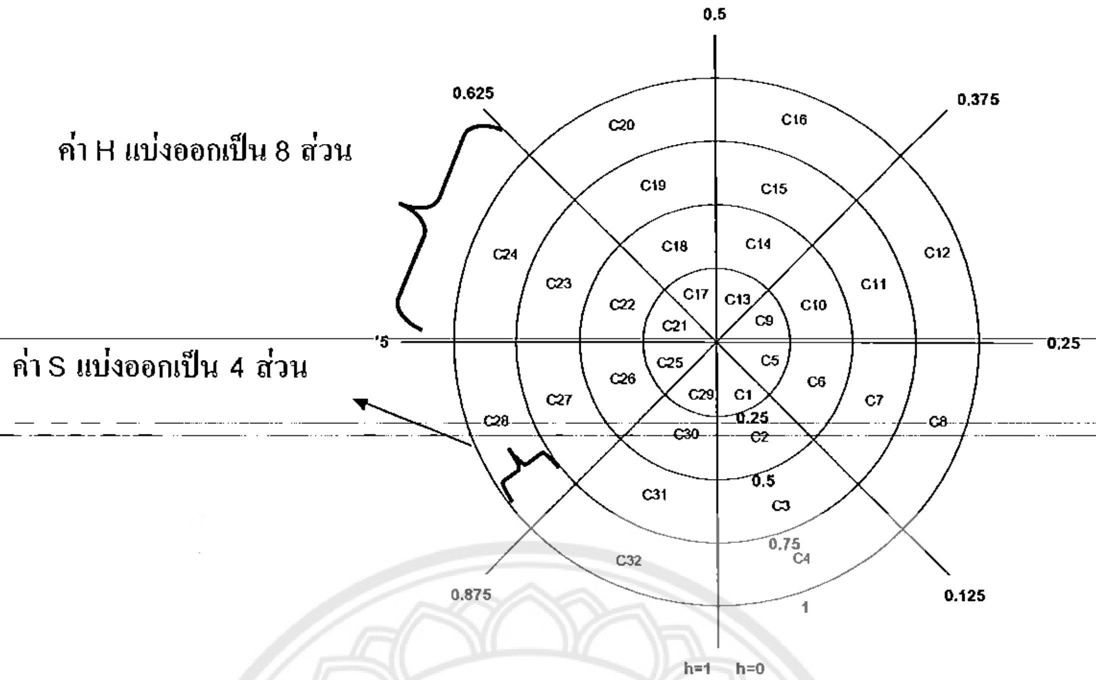
ในระบบ HSV ข้อมูลของสีหนึ่งจุดจะถูกเก็บแสดงด้วยเลขทศนิยม (float) 3 ตัว แต่ละตัวมีค่าระหว่าง 0 ถึง 1 คือ



• ค่า **Hue** บอกตำแหน่งของสีนั้นในแถบสีรุ้ง (แดง แสด เหลือง เขียว น้ำเงิน คราม ม่วง) เริ่มจากสีแดงเป็นค่าน้อยคือ 0 ไปถึงสีม่วงเป็นค่ามากคือ 1

• ค่า **Saturation** บอกความบริสุทธิ์ (purity) ของสีนั้น เริ่มจากสีขาว เป็นค่าน้อยคือ 0 ไปถึงสีที่มี ค่า hue นั้น เป็นค่ามากคือ 1

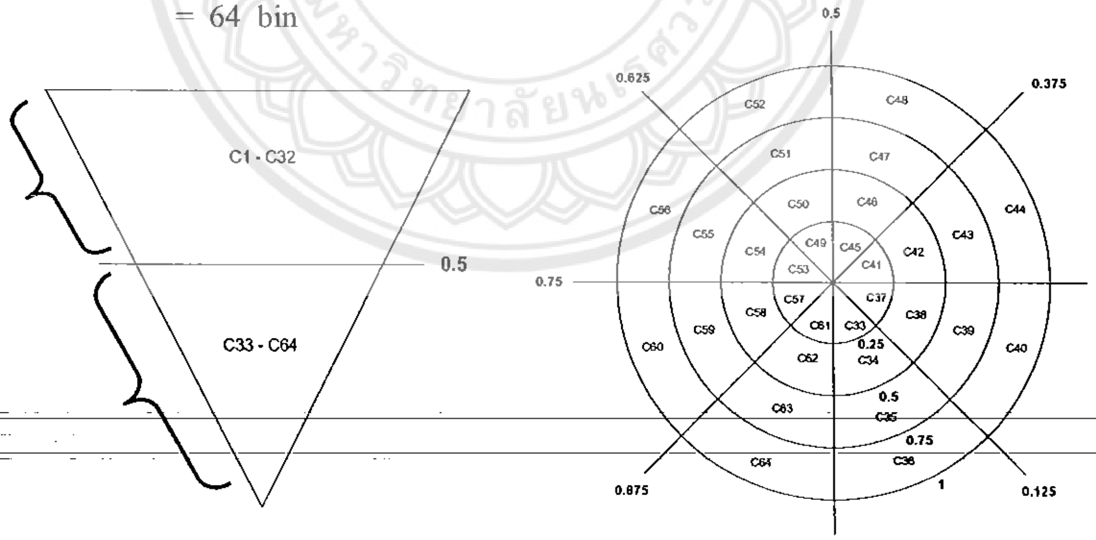
• ค่า **Value** บอกความสว่าง (luminance) ของสี เริ่มจากสีดำ เป็นค่าน้อยคือ 0 ไปถึงสีที่มีค่า hue นั้น เป็นค่ามากคือ 1



รูปที่ 4.1 การแบ่งค่า H และ S ส่วนบนในโมเดล HSV

สามารถแบ่ง HSV ออกเป็น 64 ส่วน โดย

$$\begin{aligned} & \quad\ \ \mathrm{H}\ \ \mathrm{S}\ \ \mathrm{V} \\ \mathbf{HSV} &= 8 * 4 * 2 \\ &= 64\ \text{bin} \end{aligned}$$



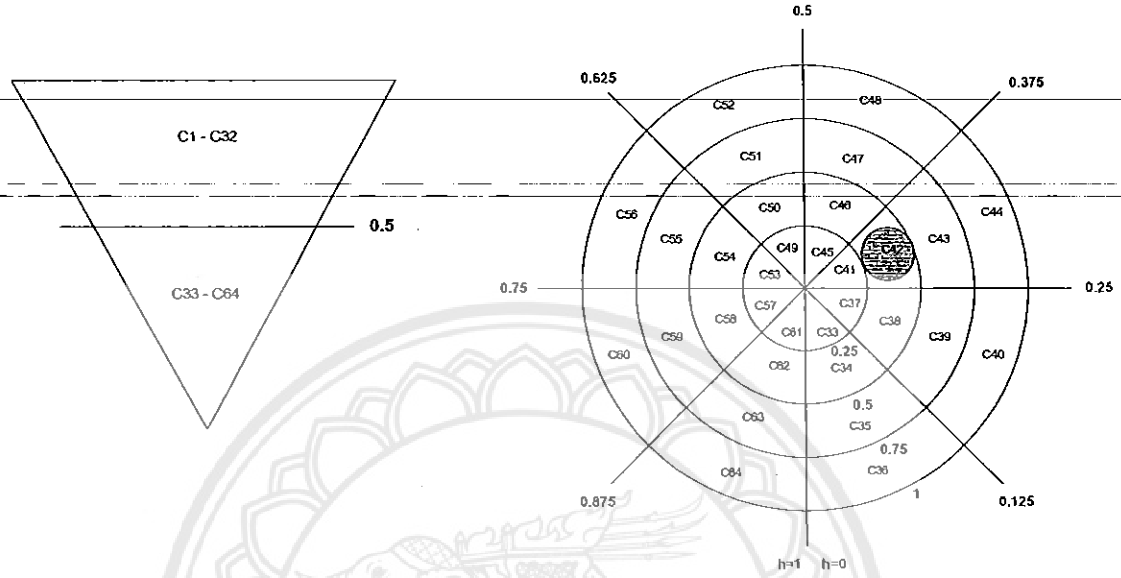
a) การแบ่งค่า V ในโมเดล b) การแบ่งค่า H และ S ส่วนล่างในโมเดล

รูปที่ 4.2 การแบ่งค่าในโมเดล HSV

ตัวอย่างการคิดว่าผลที่ได้จะตกลงตรงส่วนไหนของโมเดล HSV

<u>**สมมุติ**</u> บอกค่า HSV มาดังนี้

H	S	V
(0.28	, 0.35	, 0.86)



รูปที่ 4.3 ตัวอย่างการคิดค่า histogram จากการใช้โมเดล

จะเห็นได้ว่าเมื่อคิดออกมาแล้วค่าตกอยู่ในตำแหน่งที่ C42 ก็จะนำตำแหน่งที่ได้ไปเพิ่มค่าขึ้นที่ตำแหน่งที่ตรงกันของ histogram นั่นก็คือถ้าตรงกับบิน (bin) ไหนก็บวกที่ bin นั้นขึ้นไปแต่ยังไม่รู้ว่าเพิ่มไปเท่าไร

สูตรการคำนวณหาตำแหน่งใน model HSV

ถ้าค่า V อยู่ในช่วง 0 - 0.5

- $(H * 4) - (4 - S) + 0$

ถ้าค่า V อยู่ในช่วง 0.51 – 1

- $(H * 4) - (4 - S) + 32$

จากตัวอย่างเดิม H=0.28,S=0.35,V=0.86 สามารถทำการคำนวณจากสูตรได้ดังนี้

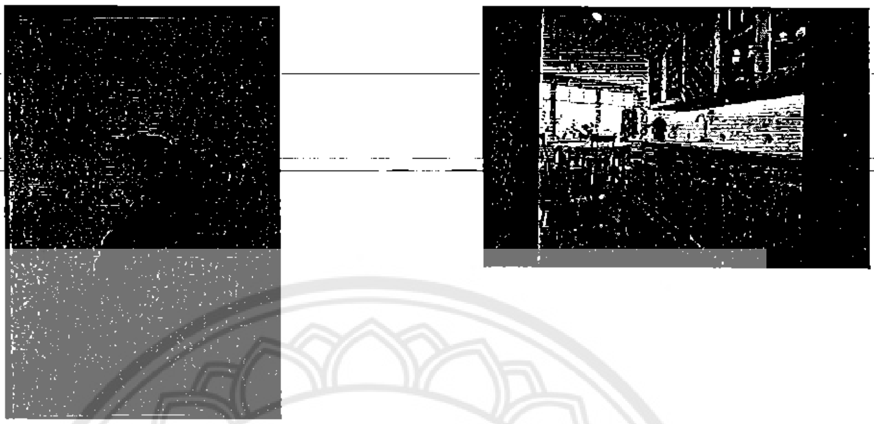
$(3 * 4) - (4 - 2) + 32 = 42$

2) ชนิด 115 บิน ประกอบด้วยข้อมูล 3 ลักษณะ คือ color , texture , shape รวมกันเป็นข้อมูลพื้นฐาน

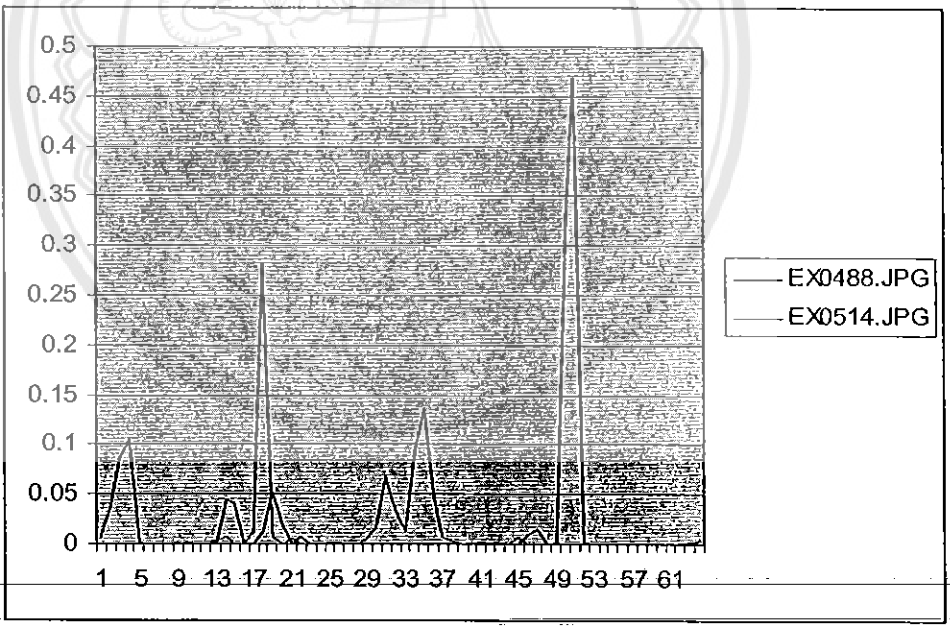
Color => color histogram

Shape => ทำจาก Fourior description

Texture => Gabor wavlet



รูปที่ 4.4 ภาพตัวอย่างที่นำมาเปรียบเทียบ

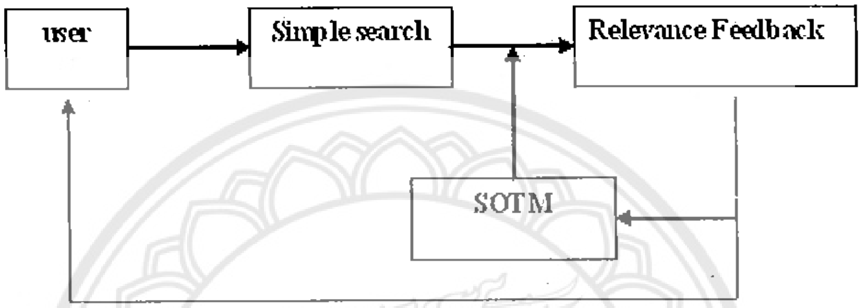


รูปที่ 4.5 กราฟผลของการทำ histogram ของภาพตัวอย่าง

จากรูปที่ 4.5 เห็นได้ว่ากราฟทั้ง 2 ไม่มีส่วนที่ซ้อนกันเลย ถึงมีก็มีน้อยมาก จึงสรุปได้เลยว่าภาพ 2 ภาพ นั้นไม่มีความคล้ายคลึงกันของโทนสีเลย ถ้าดูจากรูปก็จะเห็นว่าเป็นไปตามข้อสรุป

4.2 การทำงานของ SOTM (Self-Organization Tree Map)

ในส่วนของ SOTM จะทำหน้าที่แทน user คือที่จริงแล้วผู้ใช้จะเป็นผู้คัดเลือกรูปภาพจากภาพผลลัพธ์ทั้งหมดที่ได้จาก simple search โดยจะแบ่งเป็นรูปที่ถูกต้องและรูปที่ไม่ถูกต้อง แต่เราจะใช้ SOTM เข้ามาแทนในขั้นตอนนี้ เพื่อลดการทำงานของผู้ใช้และลด bandwidth ในการที่ผู้ใช้ จะต้องมี feedback กลับไปให้คอมพิวเตอร์ โดย SOTM จะทำการแบ่งกลุ่มภาพออกเป็น 2 กลุ่ม คือ กลุ่ม relevant และ non-relevant แล้วจะนำกลุ่มที่ relevant ไปทำการประมวลผลต่อไปจนกระทั่งได้ภาพที่ใกล้เคียงที่สุด ก็จะส่งกลับมาให้ผู้ใช้



รูปที่ 4.6 การทำงานที่มี SOTM เข้ามาช่วย

4.3 กระบวนการการค้นคืน (Relevance Feedback algorithm)

RF (Relevance Feedback) แบบ simple search คือ กระบวนการค้นคืนข้อมูล โดยจะรับข้อมูลขั้นต้นหรือ Training Vector จาก User เพื่อนำไปใช้ทำ Query เรียกข้อมูลในฐานข้อมูลอีกครั้งหนึ่ง ซึ่งข้อมูลในฐานข้อมูล นั้นจะเป็น F.V. ที่เก็บค่าฮิสโตแกรมของโมเดลสีแบบ HSV

ซึ่งในการทำ RF นั้น Training Vector จะเป็นการรับชื่อของภาพที่ผู้ใช้ได้เลือกแล้วนำไปทำเป็นเงื่อนไขใน script SQL เพื่อสร้าง Query เรียกข้อมูลในฐานข้อมูลออกมา

เมื่อได้ฮิสโตแกรมของ Training Vector จะนำมาเข้ากระบวนการหาค่าระยะห่าง (Distance) โดยนำมาเปรียบเทียบกับค่าฮิสโตแกรมของภาพทั้งหมดในฐานข้อมูลตามสูตร

จากนั้นเมื่อได้ค่า D ออกมา จะนำมาเรียงลำดับจากน้อยไปมากและตัด 200 ภาพแรกออกมา เนื่องจากมีความใกล้เคียงกับ Training Vector มากที่สุด เพราะมีค่า D น้อยที่สุด และส่งผลเป็นเอาต์พุตให้ผู้ใช้อีกครั้งหนึ่ง

การหาความคล้ายคลึงของภาพโดยไม่คิดน้ำหนัก (Simple search)

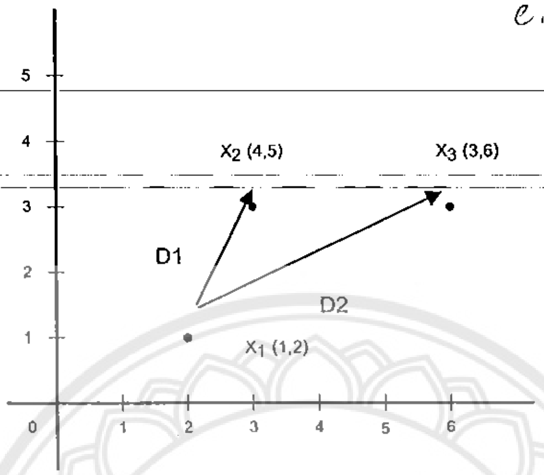
เมื่อเราได้ค่าฮิสโตแกรมของรูปในแต่ละบินออกมาแล้ว เราจะนำมาหาความคล้ายคลึงของภาพโดยไม่คิดน้ำหนัก คือเราจะนำค่าในแต่ละบินของรูป มาทำการเปรียบเทียบกัน โดยการหา

- 9 ส.ค. 2547 ห้องสมุดคณะวิศวกรรมศาสตร์ 4700200

ค่า D ออกมา ในการคำนวณหาความคล้ายคลึงกันของรูปภาพในแบบนี้ เราจะไม่คิดค่าน้ำหนัก(W) โดยค่าน้ำหนักจะเป็น 1 เท่ากันทั้งหมด

15067029 ปธ. ก275ร 2546 ฉ.2

ตัวอย่างแนวคิด



รูปที่ 4.7 ภาพตัวอย่างแนวคิดหาความคล้ายคลึงของภาพโดยไม่คิดน้ำหนัก

$$\textbf{Distance function} \equiv D(v, v_q)$$

เมื่อ $v_q = [v_{q1}, v_{q2}, \ldots, v_{qN}]^T$ คือ feature vector ของ query image

$v_i = [v_1, v_2, \ldots, v_N]^T$ คือ feature vector ของ image ในฐานข้อมูล

Euclidean distance $$D_j(\overline{v}_j, \overline{v}_q) = \sqrt{\sum_{i=1}^{n} \left(\overline{v}_i - \overline{v}_q\right)^2} \quad j=1,2,3,\ldots,n \qquad (4.1)$$

ในขั้นแรกให้ทำการคำนวณ $D_j(\overline{v}_j, \overline{v}_q)$ แต่ละภาพในฐานข้อมูล

$$D_1(\overline{v}_1, \overline{v}_q), D_2(\overline{v}_2, \overline{v}_q), D_3(\overline{v}_3, \overline{v}_q), \ldots, D_j(\overline{v}_j, \overline{v}_q)$$

ต่อไปให้เรียงค่า D จากน้อยไปมาก

$$D_5 < D_3 < D_8 < D_1 < D_7 < D_2 < D_4 < D_6$$

$$\mathbf{D_1(X1,X2)} = \sqrt{(x_{11}-x_{21})^2-(x_{12}-x_{22})^2}$$

$$\mathbf{D_2(X1,X3)} = \sqrt{(x_{11}-x_{31})^2-(x_{12}-x_{32})^2}$$

$$D_1(X1,X2) = \sqrt{(1-4)^2-(2-5)^2} = 4.2426407$$

$$D_2(X1,X3) = \sqrt{(1-3)^2-(2-6)^2} = 4.4721360$$

จะเห็นได้ว่าค่า $D_1 < D_2$

จากตัวอย่างได้ยกตัวอย่างคู่อันดับขึ้นมาแค่ 2 คู่เท่านั้น เพื่อให้ง่ายต่อการทำความเข้าใจการคำนวณเราจะหาค่า D ของแต่ละคู่ ซึ่งก็จะได้ค่า D_1,D_2 ออกมา ค่า D ก็คือระยะห่างระหว่างภาพ query กับภาพในฐานข้อมูลทั้งหมด แต่ในที่นี่ยกมาเพียง 2 คู่ จากตัวอย่างค่า D_1,D_2 ที่ได้เราจะนำมาทำการเปรียบเทียบกันว่าค่าไหนมีค่าน้อยกว่ากัน ซึ่งได้ว่า $D_1 < D_2$ จึงสามารถสรุปได้ว่า X_2 มีความเหมือนกับภาพ query มากกว่า X_3 ภาพ query ของเราในที่นี้คือ X_1

การหาความคล้ายคลึงของภาพโดยคิดน้ำหนัก (Relevance Feedback)

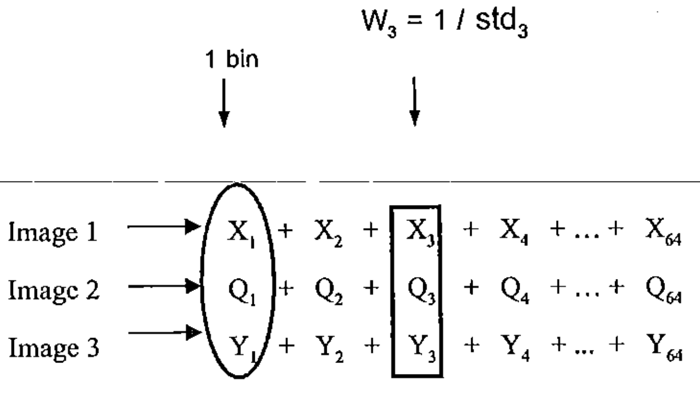
<u>สูตรการหาค่า Standard (std)</u>

$$\textbf{Standard(std)} = \sqrt{\frac{\sum_{i=1}^{n}\left(\bar{v}_i-\bar{v}_q\right)^2}{N-1}} \qquad (4.2)$$

เมื่อ **N** = จำนวน image

$\mathbf{v}_q = [\, v_{q1}, v_{q2}, \ldots, v_{qN} \,]^T$ คือ feature vector ของ query image

$\mathbf{v}_i = [\, v_1, v_2, \ldots, v_N \,]^T$ คือ feature vector ของ image ในฐานข้อมูล



$\overline{X}_1 = (X_1 + Q_1 + Y_1)/3$

.

.

$\overline{X}_{64} = (X_{64} + Q_{64} + Y_{64})/3$

สูตรการหาค่า W

$$W = 1 / std \quad (4.3)$$

$W_1 = 1 / std_1$

. .

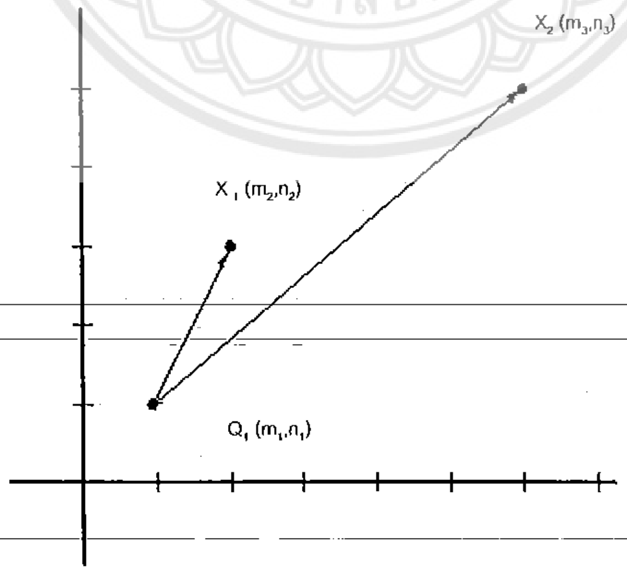
. .

$W_{64} = 1 / std_{64}$

- ถ้าค่า W ที่บิน (bin) ไหนมีค่ามากจะทำให้ bin ดังกล่าวมีความสำคัญ**และนำมาคิดแทนในสูตร**
- ถ้าค่า W ที่บิน (bin) ไหนมีค่าน้อยคือเข้าใกล้ 0 จะทำให้ bin ดังกล่าว**ไม่มีความสำคัญ**และไม่นำมาคิดแทนในสูตร

ดังนั้นจะทำให้เราได้ภาพผลลัพธ์ที่ใกล้เคียงกับภาพ Query มากที่สุด

ตัวอย่างแนวคิด



รูปที่ 4.8 ภาพแนวคิดการหาค่าความคล้ายคลึงของภาพโดยคิดน้ำหนัก

$Std_1 = (((m_1 - x_1)/(3 - 1)) + ((m_2 - x1)/(3 - 1)) + ((m_3 - x1)/(3 - 1)))^{1/2}$

$Std_2 = (((n_1 - x_1)/(3 - 1)) + ((n_2 - x1)/(3 - 1)) + ((n_3 - x1)/(3 - 1)))^{1/2}$

$W_1 = 1 / std_1$

$W_2 = 1 / std_2$

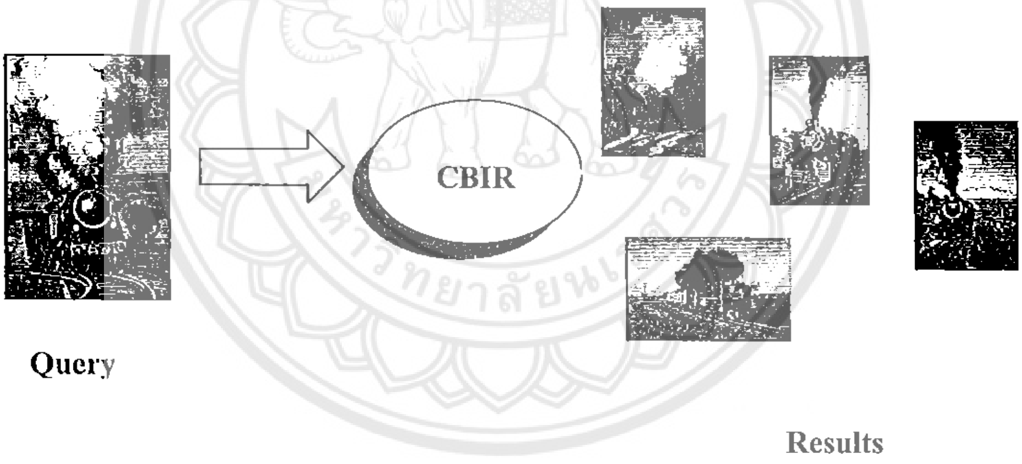
แล้วนำมาเข้าสูตร

$$D_j(\overline{v}_j, \overline{v}_q) = \sqrt{\sum_{i=1}^{n} w_i (\overline{v}_i - \overline{v}_q)^2} \qquad (4.4)$$

เมื่อได้ค่า D แต่ละอันมาแล้วก็นำค่ามาเปรียบเทียบกันเหมือนในเรื่อง simple search

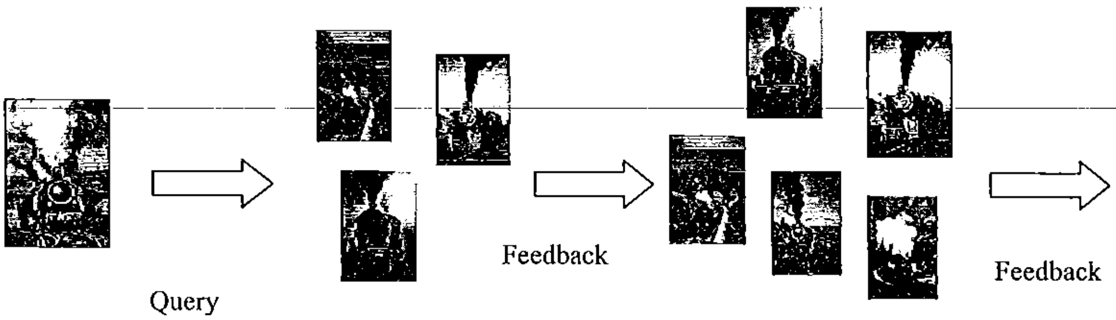
Query by Example

ให้ภาพตัวอย่างและเพื่อให้ระบบค้นคืนหาภาพที่เหมือนกับภาพตัวอย่างกลับมาให้



Relevance Feedback

- ผู้ใช้มีการป้อนกลับ (feedback) ไปเพื่อให้ได้ภาพผลลัพธ์
- ระบบการค้นคืนมีการคำนวณโดยคิดค่าน้ำหนักและ modified query



บทที่ 5

ผลการทดลอง

จากบทที่ผ่านมานั้นเราได้พูดถึงว่าโครงงานนี้คือโครงงานอะไร ถูกคิดขึ้นหรือถูกสร้างขึ้นด้วยจุดมุ่งหมายและเหตุผลใด มีอะไรเป็นปัญหาหลักในการที่จะนำเอาส่วนโครงงานวิจัยนี้ไปใช้ช่วยแก้ปัญหา และโครงงานนี้มีผู้ใช้เป็นบุคคลกลุ่มใดขอบข่ายของงานเป็นอย่างไร ใช้หลักการและทฤษฎีใดบ้างที่จะประกอบขึ้นมาเป็นตัวโครงงานวิจัยนี้

แต่สิ่งที่กล่าวมาแล้วในขั้นต้นทั้งหมดนั้น จะไม่มีความหมายอันใดเลยถ้าโครงงานวิจัยนี้ไม่ได้มีประสิทธิภาพ หรือไม่สามารถใช้งานได้จริงตามที่ได้กล่าวผ่านมาแล้วในบทก่อนหน้านี้

ดังนั้นในบทนี้จะกล่าวถึงการทดสอบประสิทธิภาพของส่วนต่างๆ ของตัวโครงงานวิจัย ซึ่งตัวโครงงานนี้จุดประสงค์หลักคือ ใช้ในการทำการค้นหาภาพตามที่ผู้ใช้ต้องการโดยใช้ SOTM มาช่วยในการค้นหา การใช้ SOTM นี้จะช่วยแบ่งเบาภาระของผู้ใช้ที่จะต้องติดต่อกับระบบ และจะเป็นการลดจำนวนการใช้ช่องสัญญาณในเครือข่ายลง ซึ่งประสิทธิภาพที่พูดถึงนี้เป็นจุดมุ่งหมายหลังของโครงงานวิจัย

ในการทดลอบโครงงานวิจัยนี้จะแบ่งการวัดผลโครงงานออกเป็นดังนี้

1. วัดผลจากการทำ Simple Search Method
2. วัดผลจากการทำ SOTM Method
3. วัดผลจากการทำ User-Feedback Method
4. วัดผลจากการทำ Semi-Automatic Method

ซึ่งการทำการทดลองทั้งหมดนี้ จะทำให้ทราบถึงประสิทธิภาพการทำงานของแต่ละโมดูลในตัวโครงงานวิจัย และเป็นการเปรียบเทียบประสิทธิภาพในการใช้งานจริงกับ Query ในแบบต่างๆ และเป็นแนวทางในการพัฒนาเพื่อเพิ่มความสามารถให้กับโครงงานให้มีประสิทธิภาพมากขึ้นต่อไป

สิ่งที่ใช้ในการทำการทดลอง

1. โปรแกรม J2EE ใช้ในการทำ WEB Server
2. โปรแกรม J2SDK ใช้ในการรันโปรแกรมภาษาจาวา
3. โปรแกรมที่แยกเป็นส่วนๆ
   3.1 โปรแกรม Simple Search
   3.2 โปรแกรม SOTM

3.3 โปรแกรม Relevance Feedback

4. โปรแกรม Data Base โดยใช้ MySQL

5. รูปภาพที่ใช้ในการทำการทดลอง

5.1 ภาพ color 5000 ภาพ

5.2 ภาพ Texture 1856 ภาพ

5.3 ภาพที่ใช้ทำเป็น Query อย่างละ 30 ภาพ

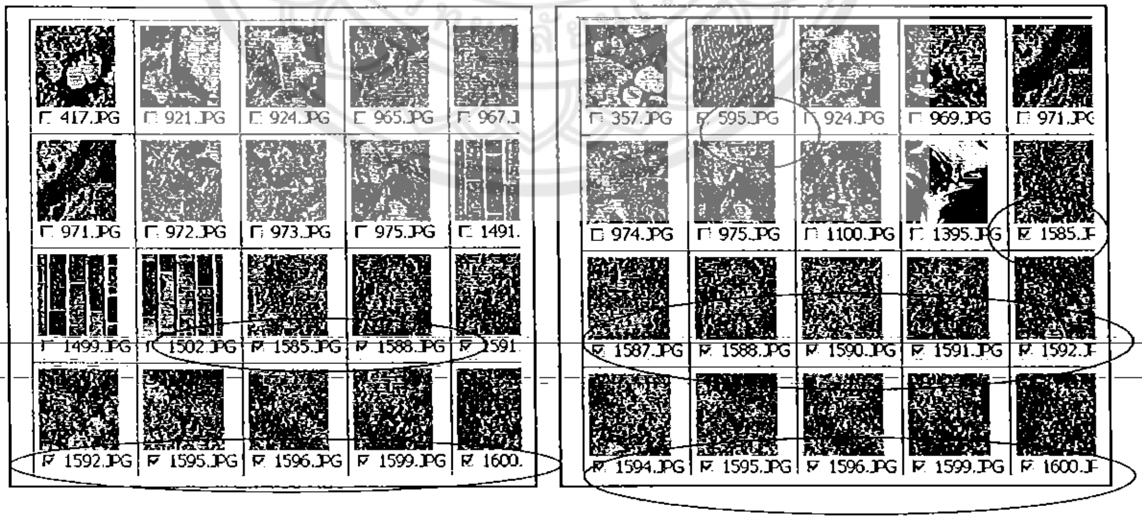
ขั้นตอนการทำการทดลอง

1. เตรียมโครงงานในส่วนของ Simple Search Method
2. เตรียมภาพ Query 30 ภาพ ที่ไม่อยู่ในกลุ่มเดียวกัน
3. นำภาพ Query ส่งให้ส่วน Simple Search Method ทำการค้นหาภาพที่ต้องการ
4. บันทึกผลการทดลองลงตารางแล้วหาค่าเฉลี่ยของการทำงาน

5.1 ผลการทดลองโดยใช้ Texture เป็นตัวค้นหา

วิธีการทำการทดลอง

จะยกการทดลอง ใช้ SOTM กับภาพแบบ Texture มาเป็นตัวอย่างโดยถ้าเป็นการทดลองอื่นก็ให้เปลี่ยนการใช้ Methods ไปเป็น Methods นั้นๆ ในการทำการทดลองเริ่มจากการรันโปรแกรมและส่งค่าของอินพุต ซึ่งในที่นี้อินพุตจะเป็นผลที่ได้จากการทำ Simple Search



รูปที่ 5.1 ภาพการแยกกลุ่มเปรียบเทียบระหว่างคน กับ SOTM โดยรูป
a) เป็นผลการแยกกลุ่มของคน b) เป็นการแยกกลุ่มของSOTM

เมื่อได้บันทึกผลการทดลองแล้ว จะต้องทำการคำนวณค่า Precision ซึ่งการทำการคำนวณเป็นดังที่แสดงข้างล่างนี้

การคิด Precision

จำนวนภาพทั้งหมด 20 ภาพ

จำนวนภาพที่ SOTM เลือกถูก 8 ภาพ (ถ้าเป็น Methods อื่นก็ให้ใช้ผลตามนั้น)

คิดเป็น % ความถูกต้อง $\left(\frac{8}{20}\right)\times 100 = 40\%$

เมื่อทำการทดลองและคำนวณค่า Precision ของแต่ละ Method เรียบร้อยแล้วก็นำค่าทั้งหมดมาเปรียบเทียบกับ User Feedback Methods และสรุปผล

ในที่นี้จะเป็นการแสดงตารางบันทึกผลการทดลอง ที่ได้จากการทำการทดลองทั้ง 4 Methods

ตารางที่ 5.1 บันทึกผลการทดลอง Simple Search Method โดยใช้ Texture

ลำดับ	ภาพ Query	จำนวนรูปที่ถูก	precision (%)	ลำดับ	ภาพ Query	จำนวนรูปที่ถูก	precision (%)
1	1.jpg	8	50%	16	824.jpg	10	63%
2	19.jpg	4	25%	17	1496.jpg	7	44%
3	40.jpg	5	31%	18	888.jpg	13	81%
4	149.jpg	12	75%	19	1195.jpg	15	94%
5	225.jpg	15	94%	20	1460.jpg	8	50%
6	1217.jpg	14	88%	21	70.jpg	7	44%
7	550.jpg	15	94%	22	179.jpg	10	63%
8	720.jpg	3	19%	23	130.jpg	13	81%
9	802.jpg	12	75%	24	116.jpg	10	63%
10	648.jpg	10	63%	25	212.jpg	15	94%
11	1098.jpg	11	69%	26	254.jpg	16	100%
12	852.jpg	6	38%	27	450.jpg	4	25%
13	1047.jpg	16	100%	28	466.jpg	3	19%
14	285.jpg	9	56%	29	486.jpg	6	38%
15	305.jpg	13	81%	30	501.jpg	16	100%
					Sum	275	1913
					AVG	9.17	63.75

ตารางที่ 5.2 บันทึกผลการทดลอง SOTM Method โดยใช้ Texture

ลำดับ	ภาพ Query	จำนวนรูปที่ถูก	precision (%)
1	1.jpg	13	81%
2	19.jpg	4	25%
3	40.jpg	7	44%
4	149.jpg	10	63%
5	225.jpg	16	100%
6	1217.jpg	16	100%
7	550.jpg	14	88%
8	720.jpg	5	31%
9	802.jpg	15	94%
10	648.jpg	11	69%
11	1098.jpg	11	69%
12	852.jpg	6	38%
13	1047.jpg	14	88%
14	285.jpg	13	81%
15	305.jpg	12	75%
16	824.jpg	3	19%
17	1496.jpg	7	44%
18	888.jpg	14	88%
19	1195.jpg	15	94%
20	1460.jpg	4	25%
21	70.jpg	7	44%
22	179.jpg	10	63%
23	130.jpg	11	69%
24	116.jpg	12	75%
25	212.jpg	14	88%
26	254.jpg	16	100%
27	450.jpg	3	19%
28	466.jpg	6	38%
29	486.jpg	9	56%
30	501.jpg	16	100%
	Sum	314	1963%
	AVG	10.467	65%

ตารางที่ 5.3 บันทึกผลการทดลอง Semi Automatic Method โดยใช้ Texture

ลำดับ	ภาพ Query	จำนวนรูปที่ถูก	precision (%)	รอบ
1	1.jpg	12	75%	2
2	19.jpg	5	31%	2
3	40.jpg	10	63%	2
4	149.jpg	14	88%	3
5	225.jpg	16	100%	2
6	1217.jpg	16	100%	2
7	550.jpg	16	100%	2
8	720.jpg	5	31%	2
9	802.jpg	18	113%	2
10	648.jpg	13	81%	2
11	1098.jpg	11	69%	3
12	852.jpg	7	44%	2
13	1047.jpg	16	100%	1
14	285.jpg	15	94%	2
15	305.jpg	15	94%	3

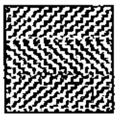
ลำดับ	ภาพ Query	จำนวนรูปที่ถูก	precision (%)	รอบ
16	824.jpg	11	69%	2
17	1496.jpg	9	56%	2
18	888.jpg	16	100%	2
19	1195.jpg	16	100%	1
20	1460.jpg	9	56%	2
21	70.jpg	10	63%	2
22	179.jpg	11	69%	2
23	130.jpg	15	94%	2
24	116.jpg	12	75%	1
25	212.jpg	16	100%	2
26	254.jpg	16	100%	1
27	450.jpg	4	25%	2
28	466.jpg	8	50%	3
29	486.jpg	14	88%	2
30	501.jpg	16	100%	1
	Sum	372	2325	59
	AVG	12.400	77.500	1.96667

ตารางที่ 5.4 บันทึกผลการทดลอง User Feedback Method โดยใช้ Texture

ลำดับ	ภาพ Query	จำนวนรูปที่ถูก	precision (%)	รอบ	ลำดับ	ภาพ Query	จำนวนรูปที่ถูก	precision (%)	รอบ
1	1.jpg	12	75%	2	**16**	824.jpg	11	69%	3
2	19.jpg	5	31%	2	**17**	1496.jpg	9	56%	3
3	40.jpg	10	63%	2	**18**	888.jpg	16	100%	3
4	149.jpg	14	88%	3	**19**	1195.jpg	16	100%	2
5	225.jpg	16	100%	2	**20**	1460.jpg	9	56%	2
6	1217.jpg	16	100%	3	**21**	70.jpg	10	63%	3
7	550.jpg	16	100%	3	**22**	179.jpg	11	69%	3
8	720.jpg	5	31%	3	**23**	130.jpg	15	94%	3
9	802.jpg	18	113%	2	**24**	116.jpg	13	81%	2
10	648.jpg	13	81%	2	**25**	212.jpg	16	100%	3
11	1098.jpg	12	75%	3	**26**	254.jpg	16	100%	2
12	852.jpg	7	44%	3	**27**	450.jpg	4	25%	2
13	1047.jpg	16	100%	2	**28**	466.jpg	8	50%	3
14	285.jpg	15	94%	3	**29**	486.jpg	14	88%	3
15	305.jpg	15	94%	3	**30**	501.jpg	16	100%	2
						Sum	374	2337.500	77
						AVG	12.467	77.917	2.567

5.2 การทดลองเรื่องความถูกในการเลือกภาพของ SOTM

ในโครงงานนี้ได้ใช้ SOTM ในการทำงานแทนคนดังนั้น ความถูกต้องของ SOTM จึงมีความสำคัญ ในการทดลองนี้จะบอกได้ว่า SOTM ทำงานถูกต้องเพียงใด

1.  ชื่อ Query 268.jpg

รูปที่ถูกต้องที่ได้จากการทำ Simple Search =16

ตารางที่ 5.5 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 268.jpg

SOTM				Feedback	
รอบ	inputที่ถูกต้องมี	ที่ SOTM เลือกถูก	precision	จำนวนรูปที่ถูกต้อง	precision
0	16	6	0.375	13	0.813
1	12	7	0.583	13	0.813
2	11	7	0.636	13	0.813
3	13	7	0.538	13	0.813
4	11	7	0.636	13	0.813
5	13	7	0.538	13	0.813
6	11	7	0.636	13	0.813
7	13	7	0.538	13	0.813
8	11	7	0.636	13	0.813
9	13	7	0.538	13	0.813
	Sum	69	5.658	130	8.125
	Average	6.9	0.566	13	0.813
	%		56.576		81.250

2. ชื่อ Query 254.jpg

รูปที่ถูกต้องที่ได้จากการทำ Simple Search = 16

ตารางที่ 5.6 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 254.jpg

SOTM				Feedback	
รอบ	inputที่ถูกต้องมี	ที่ SOTM เลือกถูก	precision	จำนวนรูปที่ถูกต้อง	precision
0	16	9	0.563	16	1.000
1	16	9	0.563	16	1.000
2	16	9	0.563	16	1.000
3	16	9	0.563	16	1.000
4	16	9	0.563	16	1.000
5	16	9	0.563	16	1.000
6	16	9	0.563	16	1.000
7	16	9	0.563	16	1.000
8	16	9	0.563	16	1.000
9	16	9	0.563	16	1.000
	Sum	90	5.625	160	10.000
	Average	9	0.563	16	1.000
	%		56.250		100.000

3. ชื่อ Query 278.jpg

รูปที่ถูกต้องที่ได้จากการทำ Simple Search = 8

ตารางที่ 5.7 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 278.jpg

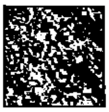
SOTM				Feedback	
รอบ	inputที่ถูกต้องมี	ที่ SOTM เลือกถูก	precision	จำนวนรูปที่ถูกต้อง	precision
0	8	4	0.500	10	0.625
1	10	8	0.800	10	0.625
2	10	8	0.800	10	0.625
3	10	8	0.800	10	0.625
4	10	8	0.800	10	0.625
5	10	8	0.800	10	0.625
6	10	8	0.800	10	0.625
7	10	8	0.800	10	0.625
8	10	8	0.800	10	0.625
9	10	8	0.800	10	0.625
	Sum	76	7.700	100	6.250
	Average	7.6	0.770	10	0.625
	%		77.000		62.500

4. ชื่อ Query 3.jpg

รูปที่ถูกต้องที่ได้จากการทำ Simple Search = 12

ตารางที่ 5.8 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 3.jpg

SOTM				Feedback	
รอบ	inputที่ถูกต้องมี	ที่ SOTM เลือกถูก	precision	จำนวนรูปที่ถูกต้อง	precision
0	12	6	0.500	13	0.813
1	13	6	0.462	13	0.813
2	13	6	0.462	13	0.813
3	13	6	0.462	13	0.813
4	13	6	0.462	13	0.813
5	13	6	0.462	13	0.813
6	13	6	0.462	13	0.813
7	13	6	0.462	13	0.813
8	13	6	0.462	13	0.813
9	13	6	0.462	13	0.813
	Sum	60	4.654	130	8.125
	Average	6	0.465	13	0.813
	%		46.538		81.250

5.  ชื่อ Query 24.jpg

รูปที่ถูกต้องที่ได้จากการทำ Simple Search = 7

ตารางที่ 5.9 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 24.jpg

SOTM				Feedback	
รอบ	inputที่ถูกต้องมี	ที่ SOTM เลือกถูก	precision	จำนวนรูปที่ถูกต้อง	precision
0	7	6	0.857	7	0.438
1	7	3	0.429	7	0.438
2	7	7	1.000	7	0.438
3	7	6	0.857	7	0.438
4	7	7	1.000	7	0.438
5	7	6	0.857	7	0.438
6	7	7	1.000	7	0.438
7	7	6	0.857	7	0.438
8	7	7	1.000	7	0.438
9	7	6	0.857	7	0.438
	Sum	61	8.714	70	4.375
	Average	6.1	0.871	7	0.438
	%		87.143		43.750

6. ชื่อ Query 740.jpg

รูปที่ถูกต้องที่ได้จากการทำ Simple Search = 16

ตารางที่ 5.10 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 740.jpg

SOTM				Feedback	
รอบ	inputที่ถูกต้องมี	ที่ SOTM เลือกถูก	precision	จำนวนรูปที่ถูกต้อง	precision
0	16	9	0.563	16	1.000
1	16	7	0.438	16	1.000
2	16	8	0.500	16	1.000
3	16	7	0.438	16	1.000
4	16	8	0.500	16	1.000
5	16	7	0.438	16	1.000
6	16	8	0.500	16	1.000
7	16	7	0.438	16	1.000
8	16	8	0.500	16	1.000
9	16	7	0.438	16	1.000
	Sum	76	4.750	160	10.000
	Average	7.6	0.475	16	1.000
	%		47.500		100.000

7. ชื่อ Query 755.jpg

รูปที่ถูกต้องที่ได้จากการทำ Simple Search = 13

ตารางที่ 5.11 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 755.jpg

SOTM				Feedback	
รอบ	inputที่ถูกต้องมี	ที่ SOTM เลือกถูก	precision	จำนวนรูปที่ถูกต้อง	precision
0	13	5	0.385	14	0.875
1	14	5	0.357	14	0.875
2	14	5	0.357	14	0.875
3	14	5	0.357	14	0.875
4	14	5	0.357	14	0.875
5	14	5	0.357	14	0.875
6	14	5	0.357	14	0.875
7	14	5	0.357	14	0.875
8	14	5	0.357	14	0.875
9	14	5	0.357	14	0.875
	Sum	50	3.599	140	8.750
	Average	5	0.360	14	0.875
	%		35.989		87.500

8. ชื่อ Query 872.jpg

รูปที่ถูกต้องที่ได้จากการทำ Simple Search = 16

ตารางที่ 5.12 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 872.jpg

SOTM				Feedback	
รอบ	inputที่ถูกต้องมี	ที่ SOTM เลือกถูก	precision	จำนวนรูปที่ถูกต้อง	precision
0	16	7	0.438	16	1.000
1	16	7	0.438	16	1.000
2	16	7	0.438	16	1.000
3	16	7	0.438	16	1.000
4	16	7	0.438	16	1.000
5	16	7	0.438	16	1.000
6	16	7	0.438	16	1.000
7	16	7	0.438	16	1.000
8	16	7	0.438	16	1.000
9	16	7	0.438	16	1.000
	Sum	70	4.375	160	10.000
	Average	7	0.438	16	1.000
	%		43.750		100.000

9. ชื่อ Query 1174.jpg

รูปที่ถูกต้องที่ได้จากการทำ Simple Search = 11

ตารางที่ 5.13 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 1174.jpg

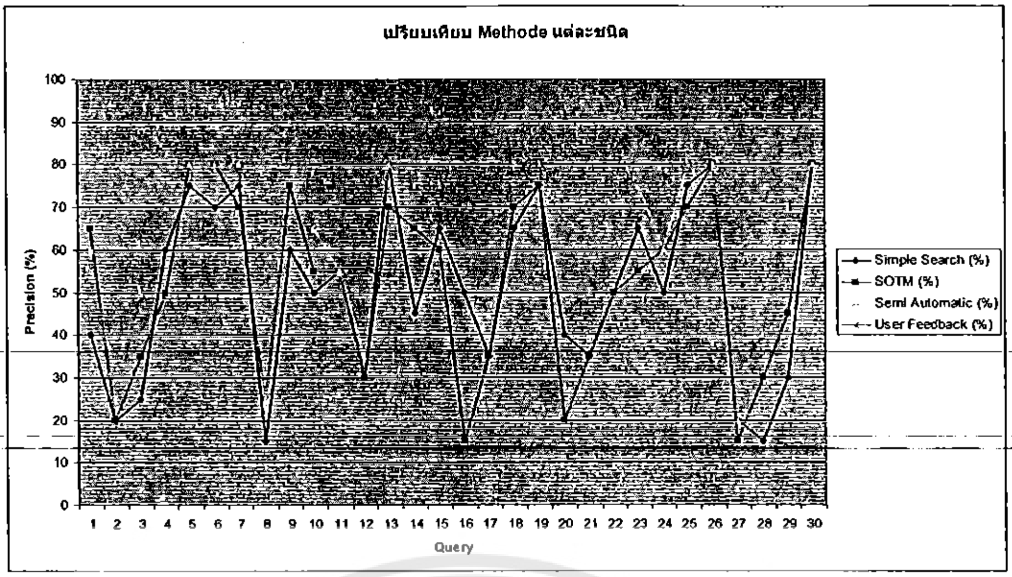
SOTM				Feedback	
รอบ	inputที่ถูกต้องมี	ที่ SOTM เลือกถูก	precision	จำนวนรูปที่ถูกต้อง	precision
0	11	11	1.000	13	0.813
1	13	1	0.077	6	0.375
2	6	6	1.000	11	0.688
3	11	8	0.727	11	0.688
4	11	5	0.455	11	0.688
5	11	8	0.727	11	0.688
6	11	5	0.455	11	0.688
7	11	8	0.727	11	0.688
8	11	5	0.455	11	0.688
9	11	8	0.727	11	0.688
	Sum	65	6.350	107	6.688
	Average	6.5	0.635	10.7	0.669
	%		63.497		66.875

10. ชื่อ Query 1203.jpg

รูปที่ถูกต้องที่ได้จากการทำ Simple Search = 12

ตารางที่ 5.14 ผลค่าความถูกต้องในการเลือกของ SOTM ของรูป 1203.jpg

SOTM				Feedback	
รอบ	inputที่ถูกต้องมี	ที่ SOTM เลือกถูก	precision	จำนวนรูปที่ถูกต้อง	precision
0	12	11	0.917	12	0.750
1	12	11	0.917	13	0.813
2	13	3	0.231	12	0.750
3	12	12	1.000	13	0.813
4	13	3	0.231	12	0.750
5	12	12	1.000	13	0.813
6	13	3	0.231	12	0.750
7	12	12	1.000	13	0.813
8	13	3	0.231	12	0.750
9	12	12	1.000	13	0.813
	Sum	82	6.756	125	7.813
	Average	8.2	0.676	12.5	0.781
	%		67.564		78.125



รูปที่ 5.2 กราฟเปรียบเทียบการทำงานแต่ละ Method ของการใช้ Texture เป็นตัวหา

จากการทดลองเราได้บันทึกผลการทดลองและทำเป็นตารางออกมา 4 ตาราง พร้อมกับทำกราฟเปรียบเทียบการทดลองทั้ง 4 เพื่อให้ดูเข้าใจได้ง่ายขึ้น ซึ่งจากกราฟนั้น เราสามารถสรุปผลดังนี้

จากกราฟ จะเห็นว่า User Feedback มี Precision ที่ดีที่สุด เนื่องจาก เป็นการจัดการโดยผู้ใช้ซึ่งเป็นมนุษย์ และมีความคิดมองภาพและเข้าใจได้ว่าภาพแบบนี้ควรจะอยู่หรือไม่อยู่ในกลุ่มเดียวกับภาพ Query หรือไม่ ซึ่งมนุษย์สามารถแยกได้ดีว่าคอมพิวเตอร์ จึงได้ผลดีกว่า Methods อื่นๆ แต่ถ้าดูจากตารางข้างล่างแล้วจะเห็นว่า การใช้ SOTM และการทำงานแบบ Semi Automatic นั้น ใช้ User น้อยกว่าการทำงานแบบ User Feedback แต่ในขณะที่ผลของงานของ Semi Automatic นั้นมีค่าใกล้เคียงกับ User Feedback มาก

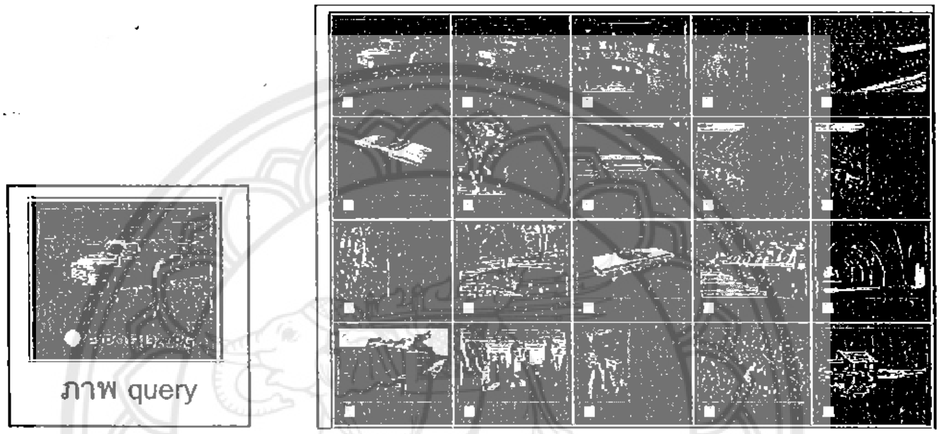
ตารางที่ 5.15 เปรียบเทียบการทำงานแต่ละ Method ของการใช้ Texture ค้นหา

Methods	Average Retrieval Rate (%)	User's RF (Iter.)
Simple Search	63.75%	-
SOTM	65%	-
Semi Automatic	77.50%	1.9667
User Feedback	77.971%	2.567

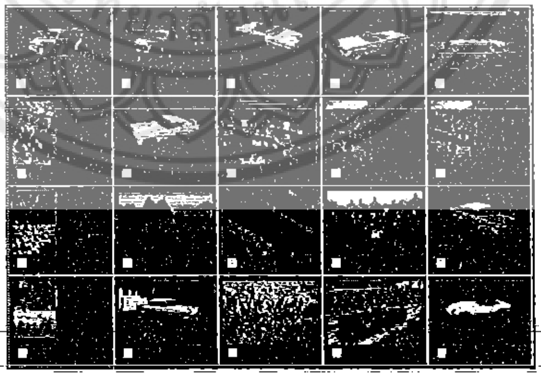
5.3 ผลการทดลองโดยใช้ Color เป็นตัวค้นหา

ตัวอย่างข้างล่างนี้จะเป็นการทดลอง ใช้ user feedback กับภาพแบบ Color มาเป็นตัวอย่าง การทำ user feedback จะต้องผ่านการทำ simple search ก่อนแล้วจึงให้ผู้ใช้ทำการเลือกภาพตามที่ ผู้ใช้ต้องการ ผู้ใช้จะทำการเลือกจนกว่าจะได้ภาพที่ต้องการมากที่สุด แล้วทำการบันทึกผลการทดลอง หลังจากนั้นให้คิดหาค่า precision เหมือนกับหัวข้อที่ 5.1

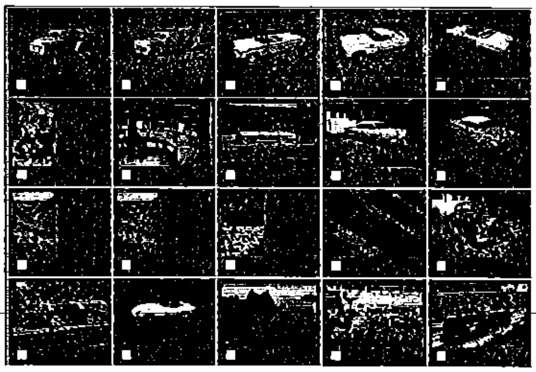
วิธีการทดลอง



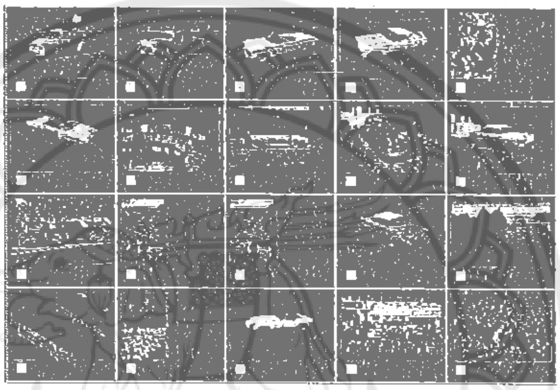
รูปที่ 5.3 ผลที่ได้จากการทำ simple search ของการใช้ Color เป็นตัวหา



รูปที่ 5.4 ผลที่ได้หลังจากผู้ใช้มี feedback ครั้งแรก



รูปที่ 5.5 ผลที่ได้หลังจากผู้ใช้มี feedback ให้ครั้งที่สอง



รูปที่ 5.6 ผลที่ได้หลังจากผู้ใช้มี feedback ให้ครั้งที่สาม

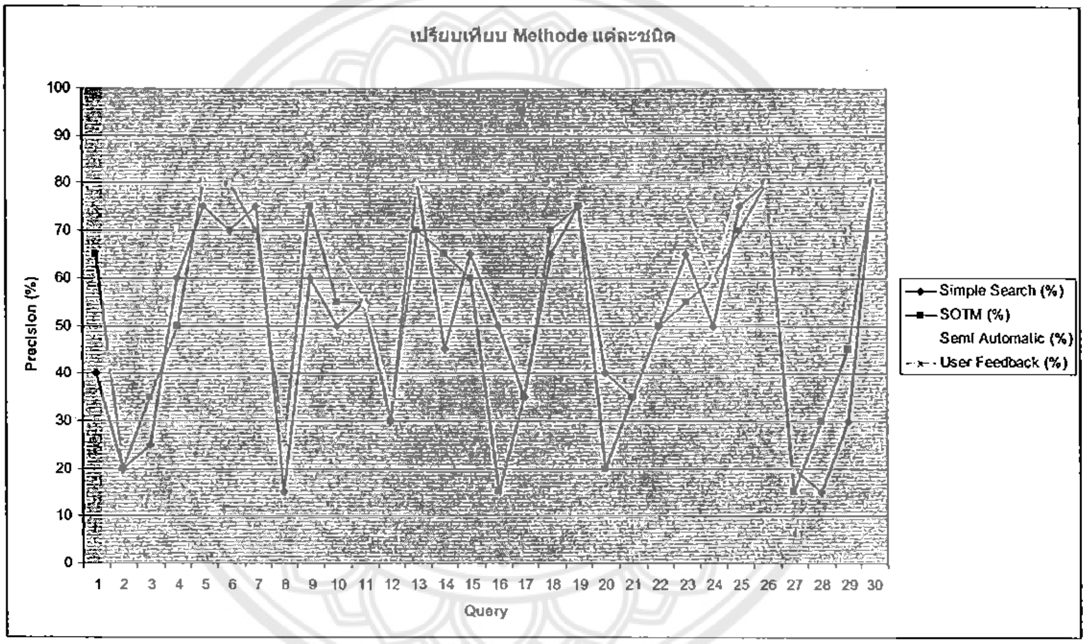
จะสังเกตเห็นว่าการที่ใช้ user feedback จะได้ผลออกมาดี เนื่องจากเราใช้มนุษย์เป็นคนเลือกไม่ใช่คอมพิวเตอร์เลือก ทำให้ได้ผลที่แม่นยำขึ้น

บทที่ 6

บทสรุป

จากบทที่ผ่านมา เราได้ทำการทดสอบการทำงานของแต่ละ Methods และได้เห็นถึงผลที่ได้ในการทำงานของแต่ละ Methods ซึ่งในบทนี้จะเป็นการนำเอาผลของบทที่ผ่านมา นำมาสรุป เพื่อให้เห็นถึงข้อดีข้อเสียของแต่ละ Methods และเพื่อเป็นแนวทางในการพัฒนาโครงงานนี้ต่อไป

6.1 สรุปผล



รูปที่ 6.1 ผลที่ได้จากการทดลองเปรียบเทียบทุก Methods

จากการทดลองในบทที่ผ่านมาเมื่อนำผลการทดลองทั้งหมดมาเขียนเป็นกราฟในรูปที่ 6.1 จะพบว่าในการทำงานโดยใช้ User Feedback Methods นั้นจะให้ผลที่ดีกว่าในการใช้ Methods อื่นๆ แต่ว่าในการทำงานในแบบ User Feedback นั้น ระบบจะต้องมีการติดต่อกับผู้ใช้อยู่หลายครั้ง ซึ่งจะเห็นได้จากรูปที่ 6.1 ซึ่งในการทำงานจริงนั้น การที่จะส่งภาพผ่านระบบเน็ตเวิร์คนั้นเราจะต้องการใช้ แบนวิดมาก ทำให้การทำงานของระบบ Network ในขณะนั้นเกิด traffic ขึ้นได้

ตารางที่ 6.1 การเปรียบเทียบระหว่าง Method ต่างๆ

Methods	Average Retrieval Rate (%)	User's RF (Iter.)
Simple Search	63.750%	-
SOTM	81.125%	-
Semi Automatic	89.083%	1.867
User Feedback	93.500%	2.867

ซึ่งจากตารางที่ 6.1 นั้น จะเห็นว่า ผลของ User Feedback จะได้ผลที่ดีแต่ การใช้งานผู้ใช้และ ระบบเน็ตเวิร์คนั้นจะสูง แต่ถ้าเปรียบเทียบกับการใช้งานวิธีอื่นๆ แล้ว จะเห็นว่า ในการใช้งานวิธีของ SOTM และกึ่งอัตโนมัติ (Semi Automatic) นั้นจะเป็นการใช้งานผู้ใช้และระบบเน็ตเวิร์คน้อยกว่า แต่ได้ผลในการทำงานที่ใกล้เคียงกัน ซึ่งจะเป็นการประหยัดทรัพยากรของระบบ และการใช้งานของผู้ใช้ก็จะสะดวกมากขึ้น ซึ่งเป็นไปตามจุดประสงค์ที่ได้ตั้งไว้

6.2 ข้อเสนอแนะ

ในการทำโครงงานครั้งนี้มีปัญหาที่เกิดขึ้นมากมาย เนื่องจากโครงงานนี้เป็นการเริ่มการวิจัยโครงงานในขั้นแรก ถ้ามีผู้ที่ต้องการศึกษาต่อหรือพัฒนาโครงงานนี้ต่อไป ทางผู้ทำโครงงานวิจัยในครั้งนี้มีข้อเสนอแนะว่า

- ควรจะได้มีการเปลี่ยนการใช้ฐานข้อมูลในการทำงาน เนื่องจากฐานข้อมูลเดิมรองรับข้อมูลได้จำนวนไม่มากนักและทำงานช้า

- เวอร์ชันของ J2EE ที่ใช้นี้เป็นเวอร์ชันเก่ามากแล้วซึ่งในขณะนี้ได้มีเวอร์ชันใหม่ออกมาในตลาดแล้วจึงควรจะเปลี่ยนการใช้งาน

- ในการทำโครงงานครั้งนี้ยังไม่ได้ทดสอบกับ Linux ซึ่งเป็น OS ที่สามารถใช้งานโดยไม่มีค่าใช้จ่ายซึ่งถ้าได้นำโครงงานนี้ขึ้นไปทำงานบน OS ดังกล่าวจะทำให้มีประโยชน์กับคนทั่วไปในการใช้งานโดยไม่เสียค่าใช้จ่าย

- ด้าน User Interface นั้นเนื่องจากการทำโครงงานนี้เป็นขั้นตอนของการวิจัยจึงมุ่งเน้นในการทำให้เห็นผลของ SOTM มากกว่าจะให้ใช้งานได้สะดวกดังนั้น ถ้าได้มีการพัฒนาต่อจึงควรจะมีการพัฒนาในส่วนของการใช้งานให้ง่ายขึ้นมากกว่านี้

6.3 ปัญหาที่พบในการทำโครงงานวิจัย

1. ในการเปรียบเทียบภาพสองภาพนั้น การใช้สีเพียงอย่างเดียวอาจจะไม่เพียงพอ ในการที่จะให้ได้ผลที่ดีนั้นจะต้องมีข้อมูลอื่นเข้ามาช่วย

2. การทำการประมวลผลข้อมูลจำเป็นจะต้องใช้เซิร์ฟเวอร์ที่มีประสิทธิภาพสูง คือ

   2.1 CPU จะต้องทำงานเร็ว

   2.2 HDD จะต้องมีความเร็วรอบที่สูงในการค้นหาข้อมูล

   2.3 RAM หน่วยความจำจะต้องมีมากพอที่จะใช้งาน

3. การเขียนโปรแกรมติดต่อกับฐานข้อมูลจะต้องระวังเรื่องการติดต่อ ไม่ควรเรียกใช้ตารางใน ฐานข้อมูลบ่อยครั้งเกินไป

4. ฐานข้อมูลจะต้องทำงานได้เร็วพอ เพื่อจะรับการค้นหาข้อมูลที่มีจำนวนมากๆ ได้

6.4 แนวทางในการพัฒนาโครงงานวิจัย

1. พัฒนารูปแบบการติดต่อกับผู้ใช้ (GUI)
2. พัฒนาข้อมูลในการที่จะใช้แบ่งแยกภาพให้มากขึ้นเพื่อเพิ่มความถูกต้อง
3. พัฒนาส่วนการทำงานของ SOTM ให้มีประสิทธิภาพเพิ่มขึ้นในเรื่องความถูกต้อง

เอกสารอ้างอิง

[1] Paisarn Muneesawang and Ling Guan, "Minimizing User Interaction By Automatic And Semi-Automatic Relevance Feedback For Image Retrieval" Proc. Of IEEE Int. Conf. On Image Processing,

[2] Paisarn Muneesawang and Ling Guan, "Automatic Machine Interactions for Content-Based Image Retrieval Using s Self-Organizing Tree Map Architecture," Proc. Of IEEE Int. Conf. On Neural Networks, Vol. 13, No.4, July 2002

มหาวิทยาลัยนเรศวร

ภาคผนวก ก

Source Code Java

```
class Value2 {
public static void main(String args[])
{
        double Wx1[]= new double[40];       double Wy1[]= new double[40];
        double Wx2[] = new double[41];double Wy2[] = new double[41];
        double Xx[] = new double[41];       double Xy[] = new double[41];
        double A[] = new double[41];
        double d1,d2 ; int j,k;
        j=k=0; A[0]=.9550;
        //กำหนดค่าเวคเตอร์ตัวอย่าง
        Xx[0]=2;    Xy[0]=5;    Xx[1]=2;    Xy[1]=4;
        Xx[2]=2;    Xy[2]=3;    Xx[3]=3;    Xy[3]=6;
        Xx[4]=3;    Xy[4]=4;    Xx[5]=3;    Xy[5]=3;
        Xx[6]=3;    Xy[6]=2;    Xx[7]=4;    Xy[7]=6;
        Xx[8]=4;    Xy[8]=5;    Xx[9]=4;    Xy[9]=4;
        Xx[10]=4;   Xy[10]=3;   Xx[11]=4;   Xy[11]=2;
        Xx[12]=5;   Xy[12]=6;   Xx[13]=5;   Xy[13]=5;
        Xx[14]=5;   Xy[14]=4;   Xx[15]=5;   Xy[15]=3;
        Xx[16]=5;   Xy[16]=2;   Xx[17]=6;   Xy[17]=5;
        Xx[18]=6;   Xy[18]=4;   Xx[19]=6;   Xy[18]=3;
        Xx[40]=7;   Xy[40]=5;   Xx[20]=7;   Xy[20]=4;
        Xx[21]=7;   Xy[21]=3;   Xx[22]=8;   Xy[22]=6;
        Xx[23]=8;   Xy[23]=5;   Xx[24]=8;   Xy[24]=4;
        Xx[25]=8;   Xy[25]=3;   Xx[26]=8;   Xy[26]=2;
        Xx[27]=9;   Xy[27]=6;   Xx[28]=9;   Xy[28]=5;
        Xx[29]=9;   Xy[29]=4;   Xx[30]=9;   Xy[30]=3;
        Xx[31]=9;   Xy[31]=2;   Xx[32]=10;  Xy[32]=6;
        Xx[33]=10;  Xy[33]=5;   Xx[34]=10;  Xy[34]=4;
        Xx[35]=10;  Xy[35]=3;   Xx[36]=10;  Xy[36]=2;
        Xx[37]=11;  Xy[37]=5;   Xx[38]=11;  Xy[38]=4;
        Xx[39]=11;  Xy[39]=3;

        for (int x = 1; x < 40; x++)
        {
                A[x]=A[x-1]/(1+A[x-1]);       //คำนวณหาค่า a(t)
        }
```

รูปที่ ก-1 Source code การคำนวณหาจุดศูนย์กลาง

```
Wx1[0]=3;               Wy1[0]=6;
Wx2[0]=9;               Wy2[0]=5;
        for (int x = 1; x < 40; x++)
        {
           A[x]=A[x-1]/(1+A[x-1]);
        }
                double Ux1=0,Uy1=0,Ux2=0,Uy2=0;
                d1=d2=0;
        for(int i=0;i<40;i++)
        {
                Ux1=(Xx[i]-Wx1[k-1]);        Uy1=(Xy[i]-Wy1[k-1]);
                Ux2=(Xx[i]-Wx2[j-1]);        Uy2=(Xy[i]-Wy2[j-1]);
                d1=Math.sqrt((Ux1*Ux1)+ (Uy1*Uy1));
                d2=Math.sqrt((Ux2*Ux2)+ (Uy2*Uy2));
        if (d1 < d2 )
        {
                Wx1[k] = Wx1[k-1]+A[k]*(Xx[i] - Wx1[k-1]);
                Wy1[k] = Wy1[k-1]+A[k]*(Xy[i] - Wy1[k-1]);
                k=k+1;
        }else if(d2 < d1){
                Wx2[j] = Wx2[j-1]+A[j]*(Xx[i] - Wx2[j-1]);
                Wy2[j] = Wy2[j-1]+A[j]*(Xy[i] - Wy2[j-1]);
                j=j+1;
        }
}
        for (int i=0;i<25 ;i++ )
        {
                System.out.println("Wx1 = "+Wx1[i]+ "," + "   Wy1 = "+Wy1[i]);
        }
        for (int i=0;i<25 ;i++ )
        {
                System.out.println("Wx2 = "+Wx2[i]+ "," + "   Wy2 = "+Wy2[i]);
        }
}
}
```

รูปที่ ก-1 (ต่อ) Source code การคำนวณหาจุดศูนย์กลาง

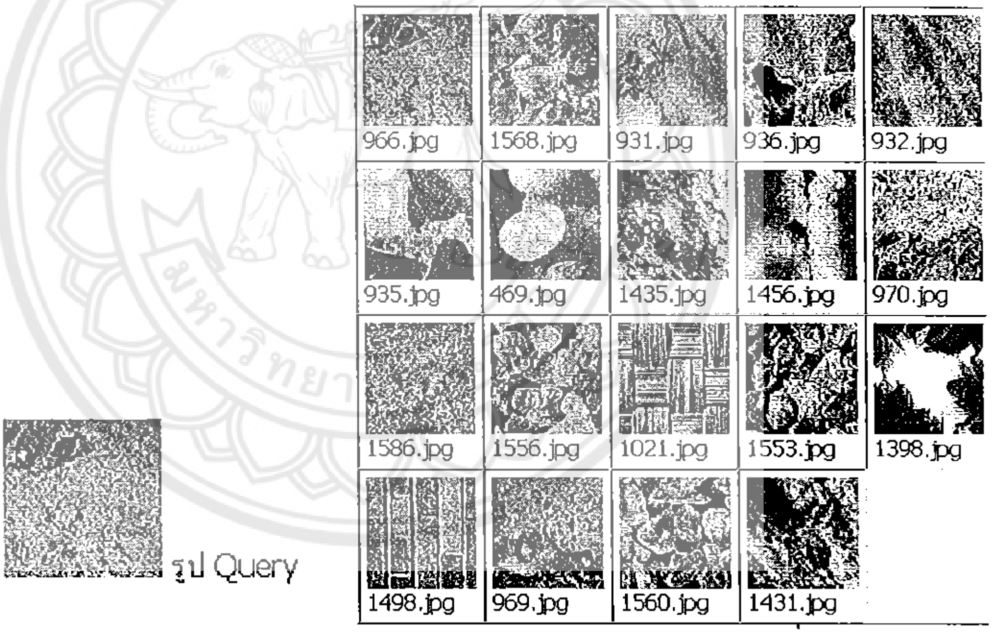
ภาคผนวก ข

การทดลอง Simple Search กับภาพในลักษณะ Texture และ Color ที่มีระดับความยากง่ายต่างกัน

1. การทดลอง Simple Search กับภาพในลักษณะ Texture

การทดลอง Simple Search กับภาพในลักษณะ Texture ที่มีความยากระดับยากมาก

การทำ Simple Search ลักษณะรูปที่มีความยากระดับยากมาก คือ ได้ผลของการค้นหาที่มีรูปในกลุ่มของ Query และ นอกกลุ่ม Query คละกัน แต่มีจำนวนของรูปในกลุ่มของ Query น้อยมากหรืออาจจะไม่พบเลย



รูปที่ ข-1 ตัวอย่าง Simple Search ที่เป็น Query ที่เป็นภาพลักษณะยากมาก

ผลการทดลอง จากรูป Query จะเห็นว่าไม่มีลวดลายที่แน่นอนทำให้การค้นหาทำได้ยาก ผลที่ได้จึงได้ภาพที่อยู่นอกกลุ่มของ Query ติดมาเป็นจำนวนมาก

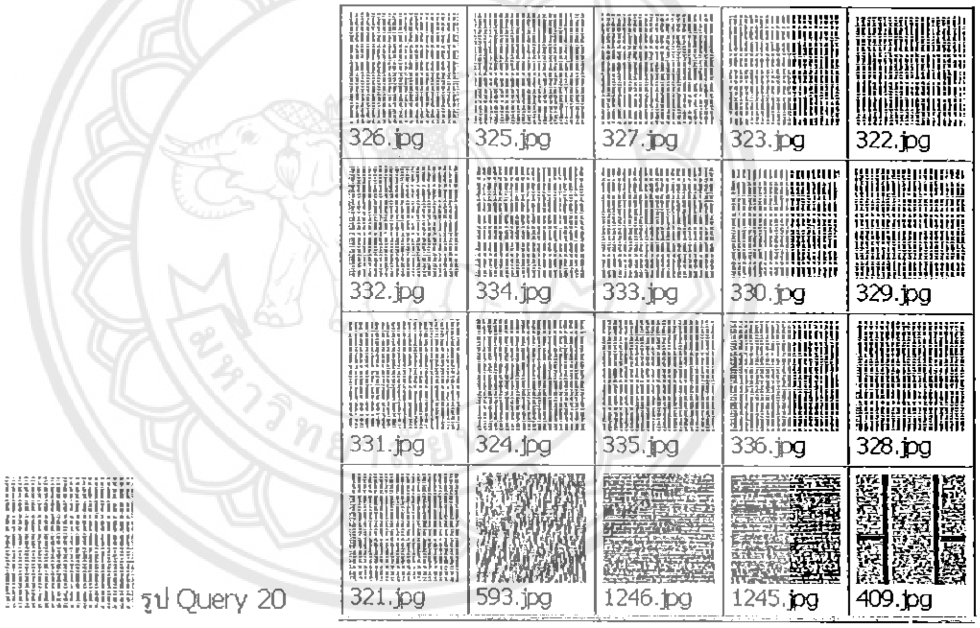
คำนวณหา % ความถูกต้อง

รูปใน Data Base ทั้งหมด	1856	รูป
รูปที่อยู่ในกลุ่มเดียวกับรูป Query มีทั้งหมด	15	รูป
รูปที่ Simple Search ค้นหามาได้	3	รูป

คิดเป็น % จะได้ว่า $\left(\frac{3}{15}\right)\times 100 = 20\%$

- การทดลอง Simple Search กับภาพในลักษณะ Texture ที่มีความยากระดับง่าย

การทำ Simple Search ลักษณะรูปที่มีความยากระดับง่าย คือ การค้นหาจะได้รูปที่อยู่ในกลุ่มของภาพ Query ทั้งหมดหรือติดภาพที่อยู่นอกกลุ่ม Query น้อยมาก



รูปที่ ข-2 ตัวอย่าง Simple Search ที่เป็น Query ที่เป็นภาพลักษณะง่าย

ผลการทดลอง จากรูป Query จะเห็นว่าไม่มีลวดลายที่แน่นอนทำให้การค้นหาทำได้ยาก ผลที่ได้จึงได้ภาพที่อยู่นอกกลุ่มของ Query ติดมาเป็นจำนวนมาก และภาพที่ติดมาก็มีลักษณะที่คล้ายกับภาพ Query เป็นผลมากจากลวดลายที่ชัดเจนทำให้ค้นหาและเทียบเคียงได้ง่าย

คำนวณหา % ความถูกต้อง

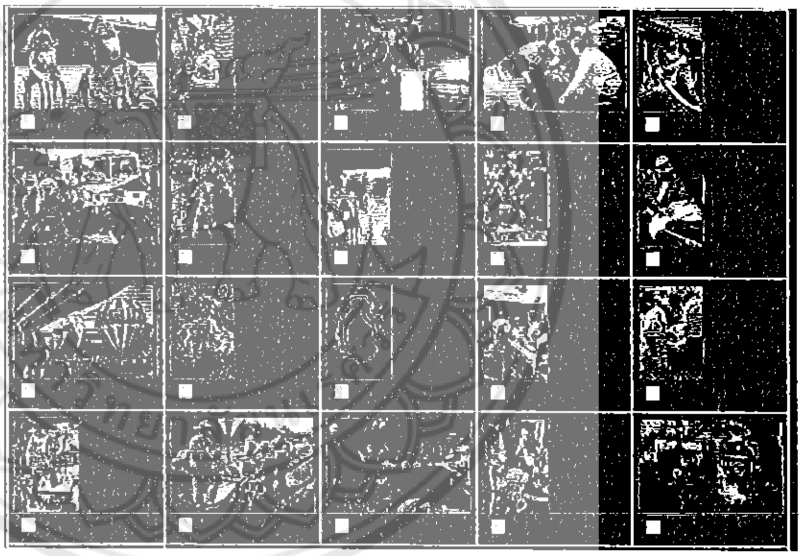
รูปใน Data Base ทั้งหมด	1856	รูป
รูปที่อยู่ในกลุ่มเดียวกับรูป Query มีทั้งหมด	15	รูป
รูปที่ Simple Search ค้นหามาได้	12	รูป

คิดเป็น % จะได้ว่า $\left(\frac{12}{15}\right)\times 100 = 80\%$

- การทดลอง Simple Search กับภาพใน Color ที่มีความยากระดับปานกลาง

การทำ simple search กับภาพที่มี color ความยากระดับปานกลาง คือ การที่จะได้ภาพที่อยู่ในกลุ่มเดียวกับภาพ query คละๆ กับภาพที่อยู่นอกกลุ่ม

ภาพ query



รูปที่ ข-3 ตัวอย่าง Simple Search ที่เป็น Query ที่เป็นภาพลักษณะปานกลาง

ผลการทดลองจากภาพ query จะพบว่าได้ภาพที่อยู่ในกลุ่มเดียวกับ query คละๆ กับที่ไม่อยู่ในกลุ่มถ้าเทียบกับในฐานข้อมูลเพราะภาพมีโทนสีมากขึ้นทำให้ติดภาพที่อยู่นอกกลุ่มมาด้วย ภาพที่อยู่นอกกลุ่มที่ติดมาด้วยได้นั้น เนื่องจากภาพอาจมีโทนสีใกล้เคียงกันแต่ไม่ตรงกับความต้องการของเรา ที่เราต้องการคือภาพหน่วยกู้ภัยไม่ใช่บอลลูนหรืออื่นๆ

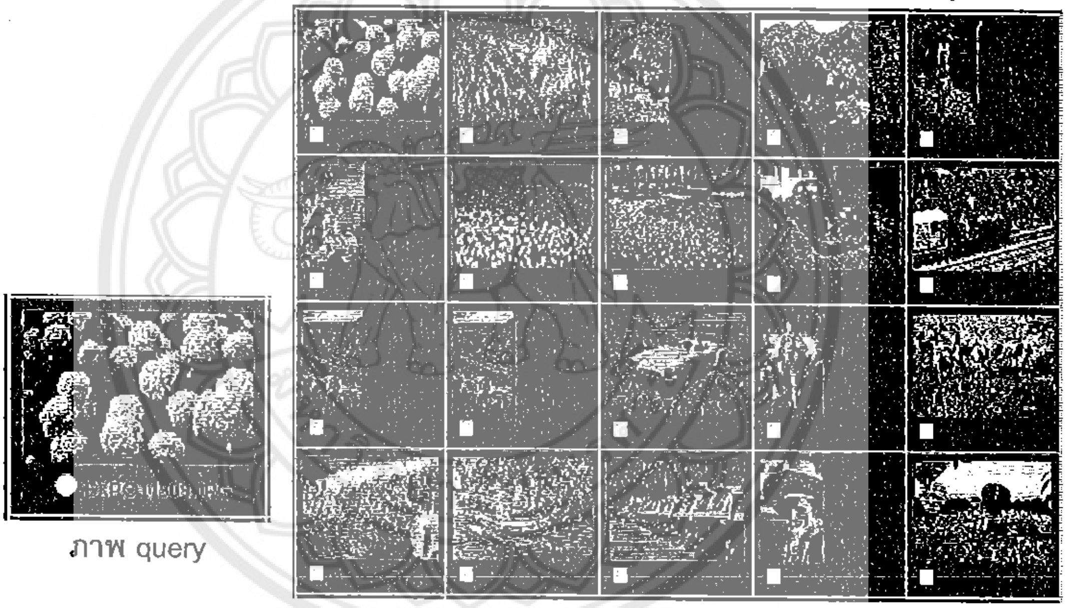
คำนวณหา % ความถูกต้อง

ภาพใน Data Base ทั้งหมด	5000	ภาพ
ภาพที่อยู่ในกลุ่มเดียวกับรูป Query มีทั้งหมด	44	ภาพ
ภาพที่ Simple Search ค้นหามาได้	12	ภาพ

คิดเป็น % จะได้ว่า $\left(\frac{12}{44}\right)\times 100 = 27.27\%$

- การทดลอง Simple Search กับภาพใน Color ที่มีความยากระดับยากมาก

การทำ simple search กับภาพที่มีความหลากหลายของสีมากขึ้น จะทำให้ได้ภาพที่อยู่ในกลุ่มเดียวกันนั้นน้อยกว่าภาพที่อยู่นอกกลุ่ม



รูปที่ ข-4 ตัวอย่าง Simple Search ที่เป็น Query ที่เป็นภาพลักษณะยาก

ผลการทดลองจากภาพ query พบว่าได้ภาพที่อยู่ในกลุ่มเดียวกันน้อยมาก จากในตัวอย่างภาพในรูปจะเป็นดอกไม้สีขาว แต่ที่มีภาพอย่างอื่นที่ไม่ตรงกับภาพ query ออกมาด้วยนั้นก็เพราะว่าภาพที่อยู่นอกกลุ่มนั้นมีโทนสีบางส่วนที่เหมือนกับภาพ query ดังนั้นจึงได้ภาพที่เราไม่ต้องการออกมาด้วย

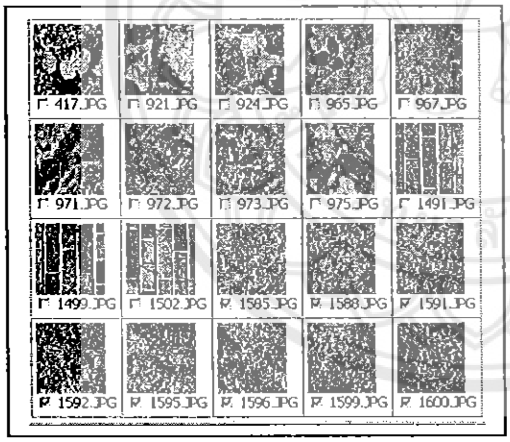
ภาคผนวก ค

การทดลองเรื่องการใช้ SOTM ในการจำแนกภาพประเภท Texture

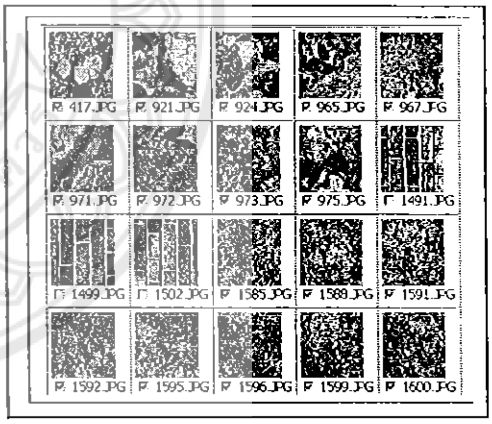
SOTM ย่อมาจาก Self Organizing Tree Map เป็นหลักการของ Neural Network ที่สามารถหาจุดศูนย์กลางของกลุ่มตัวอย่างที่เราต้องการได้และในการค้นหาภาพนั้นเราจะใช้ SOTM ในการจำแนกภาพที่ตรงกับ Query หรือภาพที่ต้องการออกจากภาพที่ไม่อยู่ในกลุ่มของ Query ออกจากกัน

การทดลอง SOTM กับภาพที่มีระดับความยากของภาพในระดับปานกลาง

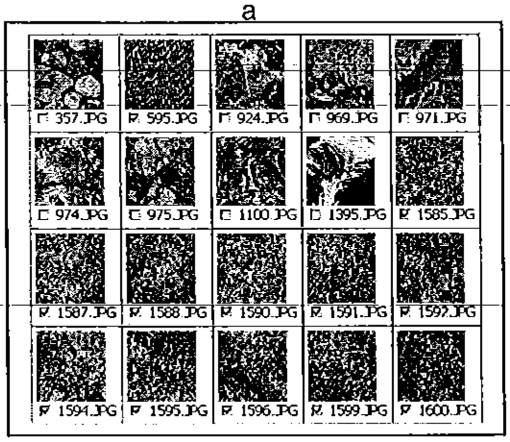
การทดลอง จะใช้ SOTM กับภาพที่มีระดับความยากของภาพในระดับปานกลาง ใช้ผลที่ได้จากการทำ Simple Search มาเป็น input ในการทำ SOTM และเปรียบเทียบผลกับการที่ใช้คนเป็นผู้ทำการแยกกลุ่มของภาพนั้น



a



b



c

รูปที่ ค-1 ภาพการแยกกลุ่มเปรียบเทียบระหว่างคน กับ SOTM โดยรูป
a) เป็นผลการแยกกลุ่มของคน
b) เป็นการแยกกลุ่มของ SOTM ในรอบแรก
c) เป็นการแยกกลุ่มของ SOTM ในรอบที่สอง

จากภาพจะเห็นได้ว่าคนจะสามารถแยกกลุ่มของภาพได้ดีกว่า SOTM ซึ่งในภาพ B) นั้นจะเป็นผลของการใช้ SOTM ในการจำแนกกลุ่มของภาพ ซึ่งเป็นการจำแนกกลุ่มของภาพในรอบแรกของ SOTM

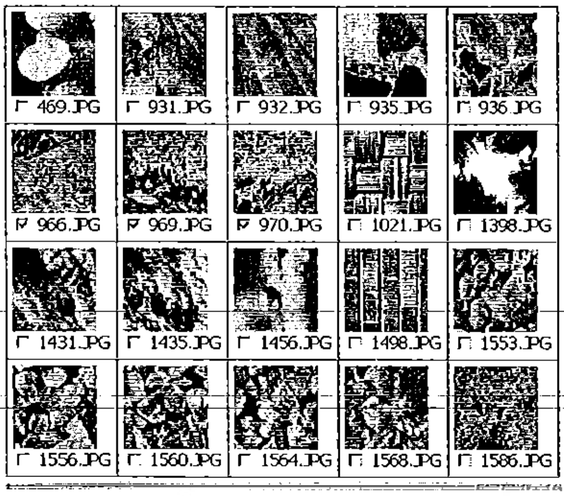
การคิด % ความผิดพลาด

จำนวนภาพทั้งหมด	20	ภาพ
จำนวนภาพที่ถูกต้อง	8	ภาพ
จำนวนภาพที่ SOTM เลือกในรอบแรก	17	ภาพ
จำนวนภาพที่ SOTM เลือกถูก	8	ภาพ
จำนวนภาพที่ SOTM เลือกผิด	9	ภาพ
คิดเป็น % ความผิดพลาด	$\left(\frac{9}{12}\right)\times 100 = 75\%$	

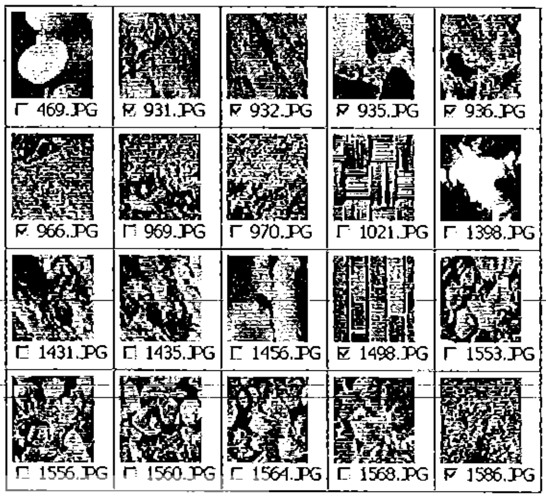
เมื่อรอบการทำ SOTM มีจำนวนสูงขึ้นความถูกต้องก็จะมีมากขึ้น จากภาพจะเห็นได้ว่าในรอบที่สองนี้ความถูกต้องของการเลือกโดยใช้ SOTM จะเป็น 100%

การทดลอง SOTM กับภาพที่มีระดับความยากของภาพในระดับยากมาก

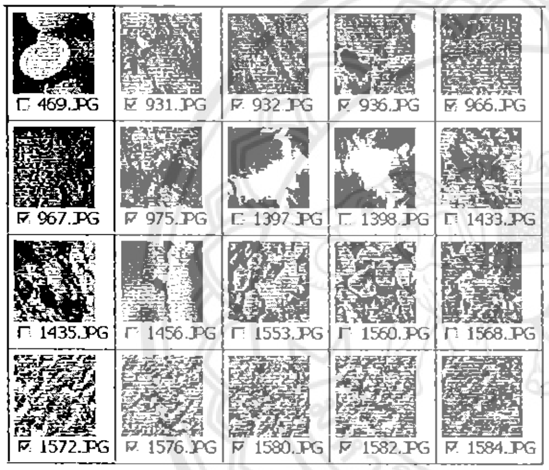
ในการทดลองนี้จะใช้ Simple Search ที่ได้จาก Query ที่มีระดับความยากของภาพในระดับยากมาก เปรียบเทียบกับการแยกโดยคนว่ามีความแตกต่างกันอย่างไร



a



b



c

a) เป็นผลการแยกกลุ่มจากคน

b) เป็นผลการแยกกลุ่มจาก SOTM ในรอบแรก

c) เป็นผลการแยกกลุ่มจาก SOTM ในรอบที่ 4

รูปที่ ค-2 ผลการแยกกลุ่ม

จากการทดลองพบว่าในการเลือกของ SOTM ในภาพที่มีระดับความยากของภาพในระดับยากมากนั้น SOTM ไม่สามารถที่จะจำแนกได้อย่างถูกต้อง ถึงแม้ว่าจะมีการทำ SOTM ซ้ำในรอบที่สูงขึ้นก็ตามแต่ผลที่ได้ก็ยังไม่ถูกต้องเท่าที่ควร

การคิด % ความผิดพลาด

จำนวนภาพทั้งหมด	20	ภาพ
จำนวนภาพที่ถูกต้องจากภาพทั้งหมด	3	ภาพ
จำนวนภาพที่ SOTM เลือกในรอบแรก	7	ภาพ
จำนวนภาพที่เลือกถูก	1	ภาพ
จำนวนภาพที่เลือกผิด	6	ภาพ

คิดเป็น % ความผิดพลาด $\left(\frac{(6-1)}{12}\right)\times 100 = 29.411\%$

ประวัติผู้เขียนโครงงาน

ชื่อ นายกฤตพน พงษ์วิทยานุกฤต

ภูมิลำเนา 13 ธารทิพย์1 สุขาภิบาล3 บางกะปิ กรุงเทพฯ

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากศูนย์การศึกษานอกโรงเรียน
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : kritpon_p@hotmail.com

ชื่อ นางสาวพงันญา ฉันทกานันท์

ภูมิลำเนา 235/1 ม.1 ต.ดอนทอง ถ.พิษณุโลก-เด่นชัย อ.เมือง จ.พิษณุโลก

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนเฉลิมขวัญสตรี
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : nging_cpe@hotmail.com