สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โปรแกรมตรวจจับสีและลวดลายเสื้อผ้าแสดงผลบนโทรศัพท์มือถือ

Color and Pattern Detector for Cloth on Mobile Phone

T 0 6 1 3 3 7

นายสิทธิชัย ส่งเสริมภักดี

เลขหมู่.....
เลขทะเบียน 61337
วัน,เดือน,ปี 17 ก.ค. 2549

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมตรวจจับสีและลวดลายเสื้อผ้าแสดงผลบนโทรศัพท์มือถือ

Color and Pattern Detector for Cloth on Mobile Phone

โดย

นายสิทธิชัย ส่งเสริมภักดี

อาจารย์ที่ปรึกษา

ดร.สมศักดิ์ วลัยรัชต์

ดร.อรัญญา วลัยรัชต์

อาจารย์สมเกียรติ วังศิริพิทักษ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2547

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมตรวจจับสีและลวดลายเสื้อผ้าแสดงผลบนโทรศัพท์มือถือ

Color and Pattern Detector for Cloth on Mobile Phone

ผู้จัดทำ นายสิทธิชัย ส่งเสริมภักดี รหัส 44010533

..... อาจารย์ที่ปรึกษา

(ดร.สมศักดิ์ วลัยรัชต์)

..... อาจารย์ที่ปรึกษา

(ดร.อรัญญา วลัยรัชต์)

..... อาจารย์ที่ปรึกษา

(อาจารย์สมเกียรติ วังศิริพิทักษ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมตรวจจับสีและลวดลายเสื้อผ้าแสดงผลบนโทรศัพท์มือถือ

นายสิทธิชัย ส่งเสริมภักดี 44010533
ดร.สมศักดิ์ วลัยรัชต์ อาจารย์ที่ปรึกษา
ดร.อรัญญา วลัยรัชต์ อาจารย์ที่ปรึกษา
อาจารย์สมเกียรติ วังศิริพิทักษ์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2547

บทคัดย่อ

โครงงานนี้พัฒนา application ในการวิเคราห์ภาพถ่ายที่ได้จากกล้องซึ่งภาพถ่ายที่ได้คือเสื้อผ้า โดยที่ application นี้ถูกสร้างและพัฒนาบนโทรศัพท์มือถือ เพื่อจะวิเคราะห์ให้ได้ว่าวัตถุที่ถ่ายภาพมาซึ่งเป็นเสื้อผ้านั้นมีสีอะไรบ้างและลายอะไร หลังจากนั้นจะบอกผลลัพธ์ออกมาในรูปแบบของเสียงเพื่ออำนวยความสะดวกให้คนตาบอดได้รับรู้

ในการวิเคราะห์ภาพนั้นการใช้ application บนโทรศัพท์มือถือมาใช้ในการวิเคราะห์เพราะว่าโทรศัพท์มือถือเป็นอุปกรณ์ที่สามารถพกพาได้สะดวกและปัจจุบันบันโทรศัพท์มือถือมีการติดตั้งกล้องไว้ใช้ในการถ่ายภาพ โดยการสร้าง Application บนโทรศัพท์มือถือนั้นใช้ภาษา J2ME และหลักการ image processing มาใช้ในการสร้าง application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Color and Pattern Detector for Cloth on Mobile Phone

Mr. Sittichai Songsermpukdee 44010533
Dr. Somsak Walairacht Advisor
Dr. Aranya Walairacht Advisor
Mr. Somkiat Wangsiripitak Advisor
Academic Year 2004

ABSTRACT

Color and Pattern Detector for Cloth on Mobile Phone is a project that developed an application for analyzing image which taking by a digital camera. The application is built and developed on a mobile phone to analyzes the material of the image. In this case, we use cloths for example, it will analyzes what color and pattern of the cloths that the picture have taken is and process in the result of man voice to the blind man.

We use this application on mobile phone because the mobile phone is portable and now a day, it is built-in with digital camera.We used J2ME and Image processing theory in creating and developing this application.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้คงไม่อาจเสร็จได้ด้วยดีหากไม่ได้รับความช่วยเหลือและร่วมมือจากหลายๆ ฝ่ายด้วยกัน บุคคลที่ต้องกล่าวถึงเพราะเป็นส่วนสำคัญที่ทำให้ปริญญานิพนธ์นี้เสร็จลงได้ก็คือ ดร.สมศักดิ์ วลัยรัชต์ ดร.อรัญญา วลัยรัชต์ และอาจารย์สมเกียรติ วังศิริพิทักษ์อาจารย์ที่ปรึกษาปริญญานิพนธ์ที่ให้ความช่วยเหลือ เอาใจใส่และแนะนำมาตลอดซึ่งต้องขอขอบพระคุณเป็นอย่างมาก และที่ขาดไม่ได้เพื่อนๆร่วมห้องโปรเจคมัลติมีเดียที่คอยให้คำปรึกษาและช่วยเหลือซึ่งกันและกันเป็นอย่างดี

และต้องขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้ข้าพเจ้ามีวันนี้ก็คือ บิดา มารดา อันเป็นที่เคารพรักยิ่ง ซึ่งได้เลี้ยงดูผู้เขียนมาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจเอาใจใส่เสมอมาในทุกๆด้านอันหาที่เปรียบมิได้ ข้าพเจ้าขอระลึกพระคุณอันสุดประมาณและขอกราบ ขอบพระคุณมา ณ ที่นี้

สิทธิชัย ส่งเสริมภักดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูปภาพ	VI
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์ของงานวิจัย	1
1.3 ขอบเขตของงานวิจัย	1
1.4 วิธีดำเนินงาน	2
บทที่ 2 ทฤษฎีการประมวณผลภาพดิจิตอล	3
2.1 พื้นฐานและระบบของสี	3
2.1.1 ระบบสี RGB	3
2.1.2 ระบบสี HSI	4
2.2 ลักษณะภาพดิจิตอล	5
2.3 ลักษณะการจัดเก็บข้อมูลภาพแบบดิจิตอล	8
2.4 การประมวณผลภาพเชิงตัวเลข (Digital Image Processing)	9
2.5 Resize image	9
2.6 การแบ่งส่วนของภาพ (Image Segmentation)	10
2.6.1 Thresholding	10
2.6.2 Flood Fill	11
บทที่ 3 ภาษาจาวาสำหรับอุปกรณ์ไร้สาย	12
3.1 ภาษาจาวาทั่วไป	12
3.2 ภาษาจาวาบนอุปกรณ์สื่อสารไร้สาย	12
3.3 J2ME	12
3.3.1 เลเยอร์จาวาเวอร์ชวลแมชชีน (Java Virtual Machine)	12
3.3.2 เลเยอร์คอนฟิกูเรชั่น	13
3.3.3 เลเยอร์โพรไฟล์ (Profile)	13
3.4 คอนฟิกูเรชั่นใน J2ME	13
3.4.1 CDC	13
3.4.2 CLDC	13
3.5 โพรไฟล์ของ J2ME	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 J2ME สำหรับอุปกรณ์ไร้สาย 15
3.7 ความต้องการของระบบ 17
บทที่ 4 การทำงานของโปรแกรม 18
4.1 โครงสร้างของโปรแกรมตรวจจับสีและลวดลายเสื้อ 18
4.2 โปรแกรมหลัก 19
4.3 ขบวนการ Pixel Grabbing 20
4.4 ขบวนการ Resizing Image 21
4.5 ขบวนการระบุสีในแต่ละ Pixel 25
4.6 ขบวนการ Pattern Recognition 27
4.6.1 ขบวนการหาลวดลายพื้นเรียบ 29
4.6.2 ขบวนการหาลวดลายแนวตั้ง 30
4.6.3 ขบวนการหาลวดลายแนวนอน 31
4.6.4 ขบวนการหาลวดลายจุด 32
4.7 การแสดงผลในรูปของเสียง 32
บทที่ 5 การทดลองและผลการทดลอง 33
บทที่ 6 วิจารณ์และสรุป 38
บรรณานุกรม 39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปที่	หน้าที่
รูปที่ 2.1 แสดงระบบสี RGB	3
รูปที่ 2.2 แสดงระบบสี RGB แบบ 24 บิท	3
รูปที่ 2.3 แสดงระบบสี HIS	4
รูปที่ 2.4 แสดงภาพดิจิตอล	5
รูปที่ 2.5 RGB Image	5
รูปที่ 2.6 Grayscale Image	5
รูปที่ 2.7 แสดงการทำ Digital Image	6
รูปที่ 2.8 รูปภาพที่มีการเก็บข้อมูลแบบต่อเนื่องและผลลัพธ์จาก sampling quantization	6
รูปที่ 2.9 Coordinate ที่ใช้แทนค่า Digital Image	7
รูปที่ 2.10 Matrix M× N Digital image	7
รูปที่ 2.11 Matrix M× N Digital image อีกรูปแบบ	7
รูปที่ 2.12 รูปที่ยังไม่ได้ resize	9
รูปที่ 2.13 ทำการ resize รูปภาพจากรูปที่ 2.12	9
รูปที่ 2.14 ผังขั้นตอนการทำ Thresholding	10
รูปที่ 2.15 ผังขั้นตอนการทำ Flood Fill	11
รูปที่ 3.1 แสดงความสัมพันธ์ระหว่าง J2SE และคลาสไลบรารี่ใน CDC และ CLDC	14
รูปที่ 3.2 โครงสร้างของเลเยอร์ต่างๆ	16
รูปที่ 4.1 แผนผังการทำงานของโปรแกรมหลัก	19
รูปที่ 4.2 แผนผังการทำงาน Pixel Grabbing	20
รูปที่ 4.3 แผนผังการทำงาน Box Filter Resizing	22
รูปที่ 4.4 แผนผังการทำงาน Horizontal Resampling	23
รูปที่ 4.5 แผนผังการทำงาน Vertical Resampling	24
รูปที่ 4.6 แสดง 216 Safe Color	25
รูปที่ 4.7 แผนผังการระบุสี	26
รูปที่ 4.8 แผนผังการทำ pattern recognition	28
รูปที่ 4.9 แผนผังการหาพื้นเรียบ	29
รูปที่ 4.10 แผนผังการหาลวดลายแนวตั้ง	30
รูปที่ 4.11 แผนผังการหาลวดลายแนวนอน	31
รูปที่ 5.1 ภาพตัวอย่างที่นำมาทดสอบ	33
รูปที่ 5.2 ผลของการทำงานบนอิมูเลเตอร์	33
รูปที่ 5.3 แสดงภาพจริงที่นำมาทดสอบ	34

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.4 ผลของการทำงานบนอิมูเลเตอร์ 34
รูปที่ 5.5 ภาพถ่ายที่มีปัญหา 34
รูปที่ 5.6 ภาพทดสอบลายจุดจากภาพที่สร้างขึ้นโดยโปรแกรม 36
รูปที่ 5.7 ภาพทดสอบลายจุดจากภาพถ่ายจริง 36
รูปที่ 5.8 ภาพที่นำมาทดสอบ 37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 หลักการและเหตุผล

ผู้จัดทำโครงงานมีแนวความคิดที่จะช่วยเหลือคนที่มีปัญหาทางสายตาสามารถรับรู้ได้ว่าเสื้อผ้าที่กำลังจะใส่นั้นมีสีอะไรและมีลวดลายอะไรได้ด้วยตัวเองเพื่อประกอบการพิจารณาว่าเสื้อผ้าที่จะใส่นั้นเหมาะสมหรือไม่ เพื่อที่สามารถตอบสนองความต้องการข้างต้นผู้จัดทำจึงพัฒนา application ที่สามารถวิเคราะห์ภาพซึ่งเป็นภาพถ่ายเสื้อผ้าว่ามีสีอะไรและมีลวดลายอะไร ในการสร้าง application นั้นผู้จัดทำเลือกพัฒนา application บนโทรศัพท์มือถือเพราะว่าโทรศัพท์มือถือเป็นอุปกรณ์ที่สามารถพกพาได้สะดวกและปัจจุบันบันโทรศัพท์มือถือส่วนใหญ่มีการติดตั้งกล้องไว้ใช้ในการถ่ายภาพทำให้มีความเหมาะสมในการใช้งาน

1.2 ขอบเขตของโครงงาน

ขอบเขตของงานวิจัยนี้จะใช้ภาษา J2ME ซึ่งตัวภาษานี่จะต่างกับภาษาจาวาทั่วไปโดยที่จะลดความสามารถต่างๆให้น้อยลงและในงานวิจัยนี้จะใช้ตัวอิดิเตอร์ (Editor) ในการนำเสนอการทำงานและอิมูเลเตอร์ (Emulator) แสดงถึงแอพพลิเคชั่นที่พัฒนาขึ้นโดยที่ภาพที่นำมาใช้ในการวิเคราะห์นั้นจะใช้ภาพจากกล้องบนโทรศัพท์มือถือซึ่งเมื่อถ่ายเสร็จจะนำข้อมูลของภาพมาโหลดผ่านสาย data cable ให้ข้อมูลมาอยู่บน PC ก่อนที่จะใช้ข้อมูลของภาพนั้นมาทดสอบบน application ที่สร้างขึ้น

สำหรับส่วนของซอฟท์แวร์ที่ใช้ในการพัฒนา applicationนั้นจะใช้ Java2 Software Development kit (J2SDK) ซึ่งประกอบด้วย Java Virtual Machine และ Java Complier, J2ME wireless toolkit (WTK) และ Editor คือ Borland Jbuilder Enterprise

1.3 วัตถุประสงค์ของโครงงาน

1.เพื่อศึกษาหลักการทำงานของ J2ME หรือ Java Micro Edition คือเทคโนโลยีหนึ่งในตระกูลจาวาซึ่งเป้าหมายของ J2ME คือ การนำซอฟต์แวร์ของ JAVA เข้าไปรันและทำงานอยู่ในอุปกรณ์เคลื่อนที่อิเล็กทรอนิกส์ อย่างเช่น โทรศัพท์มือถือ PDA หรือเพจเจอร์ซึ่งอุปกรณ์เหล่านี้มีขนาดเล็กหน่วยความจำน้อย และมีขีดความสามารถในการประมวลผลต่ำกว่าระบบคอมพิวเตอร์ทั่วไป

2.เพื่อศึกษาทฤษฎีการประมวลผลภาพด้วยคอมพิวเตอร์

3.นำภาพที่ได้จาการถ่ายด้วยกล้องดิจิตอลมาประมวลผลให้ได้ผลลัพธ์ เพื่อช่วยเหลือคนที่มีปัญหาทางสายตาสามารถรับรู้ได้ว่าเสื้อผ้าที่กำลังจะใส่นั้นมีสีอะไรและมีลายอะไรได้ด้วยตัวเองเพื่อประกอบการพิจารณาว่าเสื้อผ้าที่จะใส่นั้นเหมาะสมหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 วิธีดำเนินการ

14.1 เป้าหมาย

- รู้โครงสร้างพร้อมทั้งวิธีการเขียนโปรแกรมด้วย J2ME
- สร้างแอพพลิเคชั่นขึ้นมาโดยใช้ความรู้จากการศึกษา J2ME และทฤษฎีการประมวลผลภาพด้วยคอมพิวเตอร์

14.2 ขั้นตอนการดำเนินงาน

- หาข้อสรุปและขอบเขตของโครงงาน
- ศึกษาหาความรู้เกี่ยวกับโครงงาน และส่วนที่เกี่ยวข้อง
- ทำการออกแบบแอพพลิเคชั่น
- สร้างแอพพลิเคชั่นที่ได้ทำการออกแบบไว้
- ทดสอบแอพพลิเคชั่นที่ได้สร้างขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

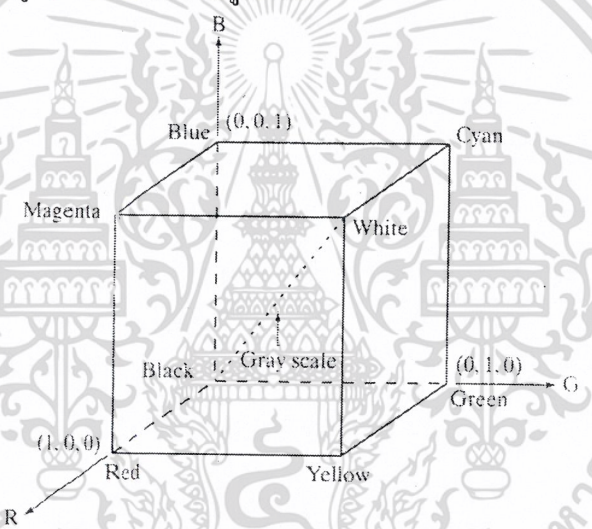
บทที่ 2
ทฤษฎีการประมวณผลภาพดิจิตอล

2.1 พื้นฐานและระบบของสี

ระบบสีพื้นฐานที่ใช้ในการประมวลผลภาพทั่วไปมี 2 ระบบ คือ ระบบสี RGB และระบบสี HSI

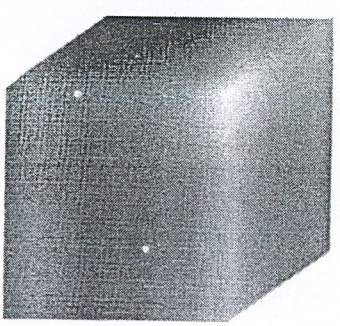
2.1.1 ระบบสี RGB

ระบบสีนี้เป็นระบบที่คอมพิวเตอร์ใช้อยู่ในปัจจุบัน โดยมีสีพื้นฐาน 3 สี คือ red, green และ blue ระบบสีนี้มีลักษณะเป็นสี่เหลี่ยมลูกบาศก์ ดังรูปที่ 2.6 โดยที่สีพื้นฐาน 3 สี จะอยู่ที่มุม 3 มุม และสีที่เกิดจากการผสมกันระหว่างสีแต่ละคู่ (cyan, magenta และ yellow) จะอยู่ที่มุมที่เหลืออีก 3 มุม สีดำจะอยู่ที่จุดกำเนิด สีขาวจะอยู่ห่างจากจุดกำเนิดมากที่สุด และเส้นที่ลากเชื่อมระหว่างสีขาวและสีดำ จะเรียกว่าระดับสีเทา ค่าสี คือจุดที่อยู่บนผิว หรือ ภายในลูกบาศก์



รูปที่ 2.1 แสดงระบบสี RGB

ปัจจุบันเครื่องคอมจะกำหนดสีพื้นฐานให้มีขนาด 8 บิท ซึ่งการนำค่าเหล่านี้มาผสมกันทำให้ได้สีที่แตกต่างกัน เพราะฉะนั้นในปัจจุบันนี้คอมพิวเตอร์สามารถให้สีที่แตกต่างกันมากกว่า 16 ล้านสี (2^{24})

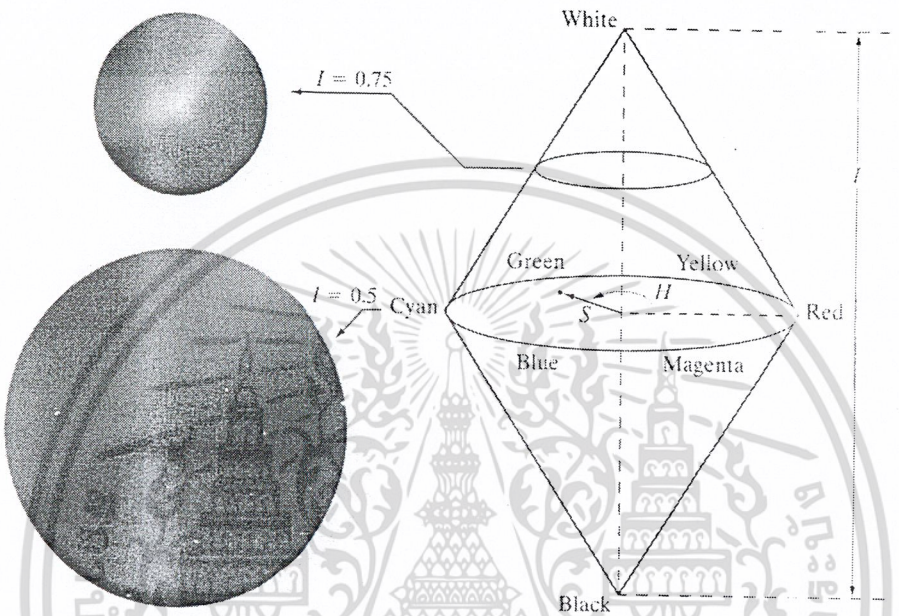


รูปที่ 2.2 แสดงระบบสี RGB แบบ 24 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 ระบบสี HSI

ระบบสีนี้มีค่าพื้นฐาน 3 ค่า คือ ค่าบ่งบอกว่าเป็นสีใด(Hue) , ค่าอิ่มตัวของสี (Saturation) ซึ่งบอกว่าสีเข้มหรือขาวขนาดไหน และค่าความเข้มของแสง(Intensity) ซึ่งบอกว่าสีสว่าง หรือ ดำขนาดไหน ระบบสีนี้มีลักษณะเป็นกรวย ดังรูปที่ 2.6 โดยที่ค่า Hue บอกด้วยองศาการหมุนในกรวย ค่า Saturation บอกด้วยระยะห่างจากตรงกลางกรวย และค่า Intensity บอกด้วยความสูงในกรวย



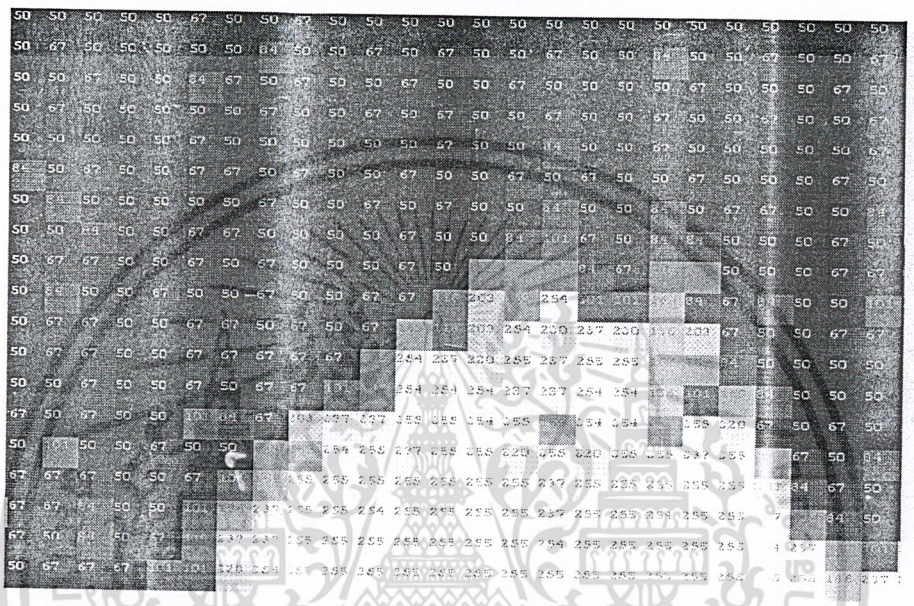
รูปที่ 2.3 แสดงระบบสี HSI

ระบบสีนี้นำมาใช้ในการทำการประมวลผลภาพได้ดีกว่า เพราะค่า Hue ค่าเดียวสามารถบอกได้ถึงความแตกต่างของค่าสีได้แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

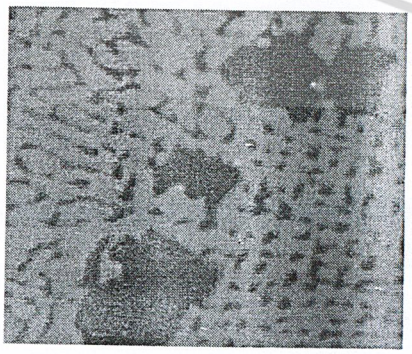
2.2 ลักษณะภาพดิจิตอล

คือรูปภาพที่จัดเก็บข้อมูลโดยระบบคอมพิวเตอร์ที่ประกอบไปด้วย array 2 มิติของพื้นที่สี่เหลี่ยมเล็กๆ ซึ่งเรียกว่า pixel ในกรณีของ Monochrome (หรือที่รู้จักว่า gray-scale หรือ black and white image) ค่าของ brightness ของแต่ละ pixel จะถูกแทนค่าด้วยตัวเลข โดยที่ค่าอยู่ระหว่าง 0–255 ซึ่งค่า 0 แทนสีดำ ค่า 255 แทนสีขาว ส่วนค่าระหว่าง 0-255 แทน shade ของสีเทา ซึ่งตัวอย่างด้านล่างโชว์ค่า pixel ไว้ดังรูปที่ 2.4

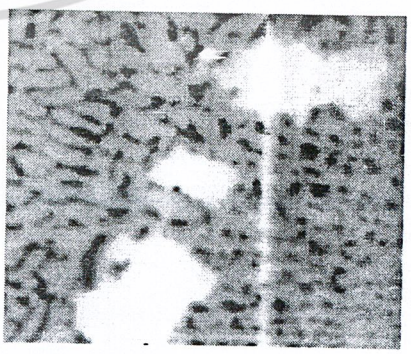


รูปที่ 2.4 แสดงภาพดิจิตอล

ส่วนภาพสีจะแทนด้วย array 2 มิติ หรือpixel ค่าของสีแดง เขียว น้ำเงิน (RG B color model)จะถูกแทนด้วยค่าซึ่งอยู่ระหว่าง 0-255 ค่า 0 หมายความว่าไม่มีสีนั้นอยู่เลย ส่วนค่า 255 หมายความว่าสีนั้นมีค่ามากที่สุด รูป digital image สี และ แบบ Grayscale แสดงดังรูปที่ 2.5 และ 2.6 ตามลำดับ



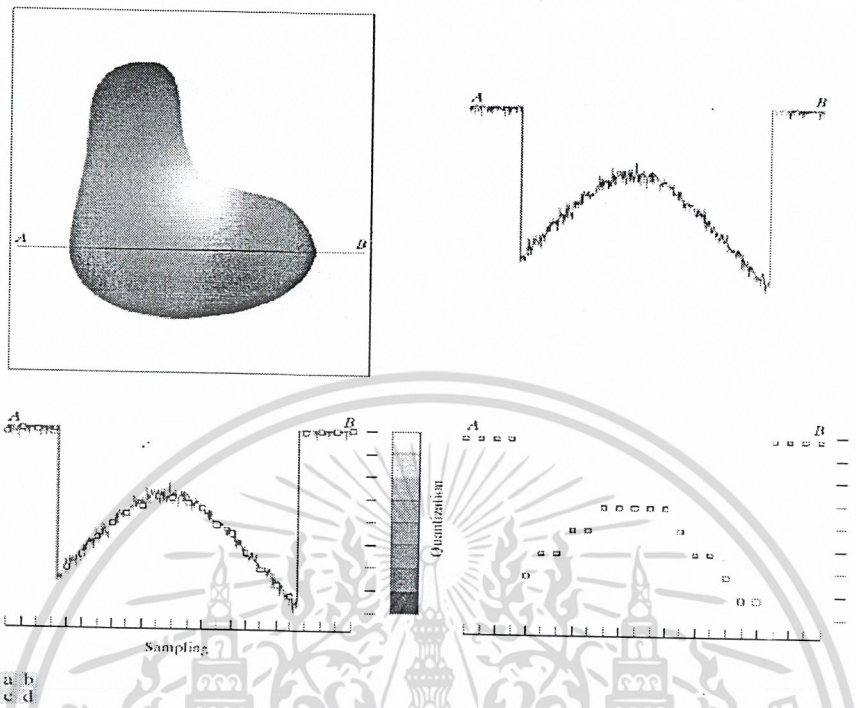
รูปที่ 2.5 RGB Image



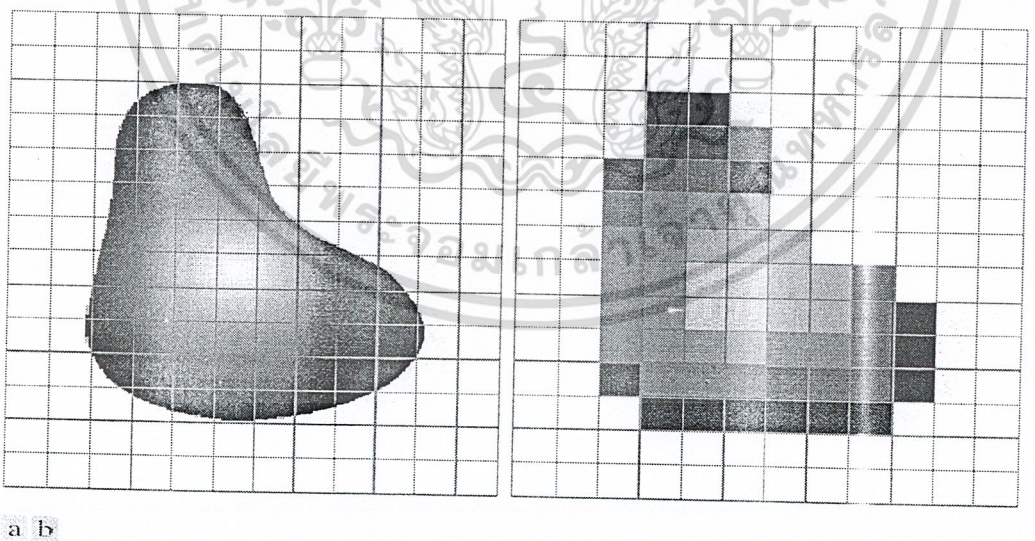
รูปที่ 2.6 Grayscale Image

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการสร้างภาพดิจิตอลเราต้องทำการเปลี่ยนแปลงข้อมูลของภาพจากเดิมที่เป็นข้อมูลที่มีลักษณะต่อเนื่องให้อยู่ในรูปแบบที่ของ digital ซึ่งต้องใช้กรรมวิธีที่เรียกว่า sampling และ quantization



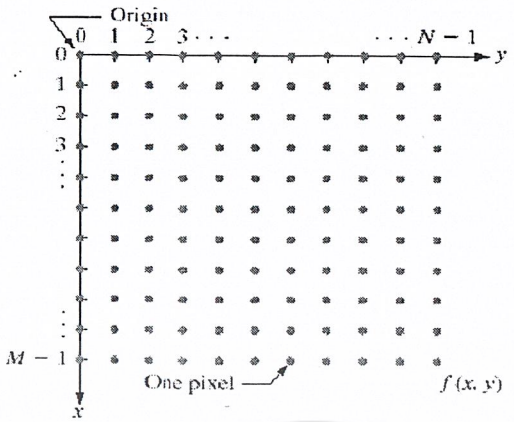
รูปที่ 2.7 แสดงการทำ Digital Image (a) ภาพที่มีข้อมูลแบบต่อเนื่อง (b) การแสกนเส้นจาก A ไป B ในภาพที่มีข้อมูลแบบต่อเนื่อง (c) ทำการ sampling และ quantization (d) แสกนเส้นแบบ Digital



รูปที่ 2.8 (a) รูปภาพที่มีการเก็บข้อมูลแบบต่อเนื่อง (b) ผลลัพธ์จากการทำ sampling และ quantization

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลลัพธ์จากการทำ sampling และ quantization จะอยู่ในรูปของ matrix ที่เก็บค่าจำนวนจริง



รูปที่ 2.9 Coordinate ที่ใช้แทนค่า Digital Image

ซึ่งเราสามารถเขียน M× N Digital image ในรูปแบบของ matrix ดังรูปที่ 6.1

$$f(x, y) = \begin{pmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{pmatrix}$$

รูปที่ 2.10 Matrix M× N Digital image

ในบางครั้งจะดีกว่าถ้าเราใช้รูปแบบของ matrix อีกแบบเพื่อทำการแทนค่า Digital image และค่าแต่ละ element ของมันนั่นก็คือ matrix ที่อยู่ในรูปแบบดังรูปที่ 6.2

$$A = \begin{pmatrix} a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\ \vdots & \vdots & & \vdots \\ a_{M-1,0} & a_{M-1,1} & \cdots & a_{M-1,N-1} \end{pmatrix}$$

รูปที่ 2.11 Matrix M× N Digital image อีกรูปแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทางด้านขวาของสมการเป็นการกำหนดค่าของ Digital image ซึ่งแต่ละส่วนของ matrix array เราจะเรียกว่า image element, picture element, pixel หรือ pel จากกระบวนการสร้างภาพดิจิตอลข้างต้น จะเห็นว่าสามารถทราบขนาดของความละเอียดของภาพ N×N พิกเซล และจำนวนระดับของเกรย์เสกล ในทางปฏิบัติการทำ quantization ในระบบภาพดิจิตอลจะมีค่าดังสมการที่ 2.1

$$B = N \times N \times N \text{ บิต} \quad \text{..... (2.1)}$$

เมื่อ B = ขนาดของข้อมูลภาพที่เป็นดิจิตอล

G = จำนวนของเกรย์เสกลที่ต้องการใช้ในการเก็บข้อมูลภาพ

M = จำนวนบิตที่ใช้ในการแทนข้อมูลภาพ 1 พิกเซล

โดย M สามารถหาได้จาก G = 2 ยกกำลัง M

2.3 ลักษณะการจัดเก็บข้อมูลภาพแบบดิจิตอล

โดยทั่วไปแล้ว ข้อมูลภาพจะมีค่าความเข้มตั้งแต่ 2 ระดับขึ้นไป แต่ที่ใช้กันมากจะใช้กันที่ค่าระดับความเข้มของจุดภาพเท่ากับ 256 ระดับ ซึ่งจะทำให้ค่าของจุดภาพอยู่ในช่วง 0-255 โดยใช้เนื้อที่การเก็บข้อมูลภาพขนาด 1 ไบต์ หรือ 8 บิต สำหรับข้อมูล 1 จุดภาพ (2 ยกกำลัง 8 = 256) ในกรณีที่ต้องการภาพที่มีความละเอียดของระดับความเข้มสูงๆ อาจจะต้องการจำนวนบิตสำหรับการเก็บข้อมูลมากกว่า 8 บิต คืออาจเป็น 16 หรือ 24 บิต โดยค่าความเข้มของจุดภาพจะแยกให้เห็นชัดเจนเป็นดังนี้

1. ภาพ 2 ระดับ คือ มีเพียงแค่จุดขาวกับจุดดำเท่านั้น โดยแต่ละจุดภาพเป็นข้อมูลขนาด 1 บิต
2. ภาพ 16 ระดับ คือ ในแต่ละจุดภาพจะมีขนาดของข้อมูล 4 บิต ซึ่งทำให้สามารถแสดงได้ 16 ระดับสี หรือ 16 เกรย์สเกล ขึ้นอยู่กับภาพนั้นเป็นภาพสีหรือภาพขาวดำ
3. ภาพ 256 ระดับ คือ ในแต่ละจุดภาพจะมีขนาดของข้อมูล 8 บิต ซึ่งทำให้สามารถแสดงภาพได้ 256 ระดับสี หรือ 256 เกรย์สเกล ขึ้นอยู่กับภาพนั้นเป็นภาพสีหรือภาพขาวดำ
4. ภาพทรูคัลเลอร์ (True color) คือ ในแต่ละจุดภาพจะมีขนาดของข้อมูล 24 บิต ทำให้สามารถแสดงผลภาพได้เหมือนภาพจริงที่สุด เพราะสามรถแสดงสีได้ถึง 16,777,216 สี ภาพทรูคัลเลอร์สามารถแสดงได้เฉพาะภาพสีเท่านั้น ไม่สามารถแสดงผลภาพขาวดำได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การประมวณผลภาพเชิงตัวเลข (Digital Image Processing)

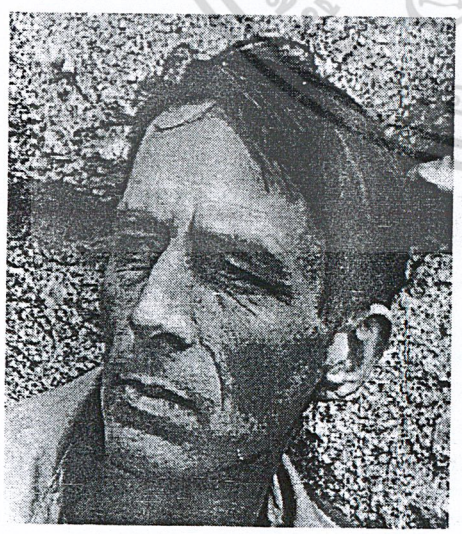
การประมวณผลภาพเชิงตัวเลข หมายถึงการนำภาพที่พบทั่วไปมาประมวณผลด้วยเครื่องคอมพิวเตอร์ โดยภาพที่นำมาประมวณผลด้วยเครื่องคอมพิวเตอร์นี้จะถูกแทนที่ให้อยู่ในรูปแบบของเมตริกซ์ดังที่กล่าวมาแล้ว

โดยทั่วไปวิธีการประมวณผลภาพเชิงตัวเลขที่ทำให้วัตถุสามารถรู้จักวัตถุภายในภาพได้นั้น พอจะแบ่งได้ 2 ระดับคือ การประมวณผลในระดับต่ำ (Low Level Image Processing) และการประมวณผลภาพในระดับสูง (High Level Image Processing) การประมวณผลภาพในระดับต่ำจะเป็นการประมวณผลเชิงตัวเลขเกือบทั้งหมด เพื่อหาตัวแปรต่างๆ มาอธิบายข้อมูลภาพ โดยมีจุดประสงค์เพื่อนำตัวแปรเหล่านั้นไปประมวณผลในระดับสูงต่อไป โดยทั่วไปแล้วการประมวณผลภาพระดับต่ำจะประกอบด้วย การประมวลผลภาพก่อน (preprocessing) การกำจัดสัญญาณรบกวน หรือการทำให้ภาพคมชัด การหาขอบภาพ เป็นต้น

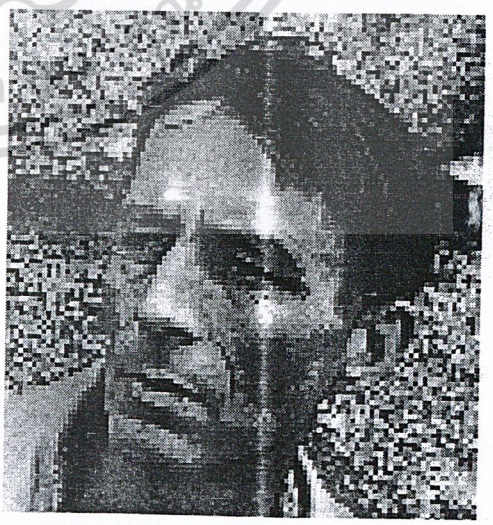
การประมวณผลภาพในระดับสูงเป็นการนำผลลัพธ์ หรือสัญลักษณ์ที่ได้จากการประมวณผลระดับต่ำมาตีความ หรือประมวณผลเพื่อให้คอมพิวเตอร์สามารถรู้จักและเข้าใจภาพได้ สำหรับความแตกต่างของการประมวลผลระดับต่ำและระดับสูงคือ ข้อมูลที่นำมาใช้ในการประมวลผลโดยที่การประมวลผลภาพระดับต่ำจะใช้ค่าความสว่างของจุดโดยตรง ส่วนการประมวลผลภาพในระดับสูงนั้น ข้อมูลของภาพที่นำมาประมวลผลจะถูกแสดงในรูปของสัญลักษณ์ ซึ่งสัญลักษณ์เหล่านี้จะแสดงถึงสิ่งต่างๆ ที่มีอยู่ในภาพ เช่น ขนาดวัตถุ รูปร่าง และความสัมพันธ์กันระหว่างวัตถุภายในภาพ

2.5 Resize image

การลดขนาดของภาพให้มีขนาดเล็กลงผลลัพธ์ก็คือทำให้จำนวน pixel ของรูปภาพที่ resize แล้วจะมีจำนวนน้อยลงเป็นจำนวนตามที่เรากำหนดแสดงดังรูปที่ 2.12 และ 2.13 ตามลำดับ



รูปที่ 2.12



รูปที่ 2.13 ทำการ resize รูปภาพจากรูปที่ 2.12

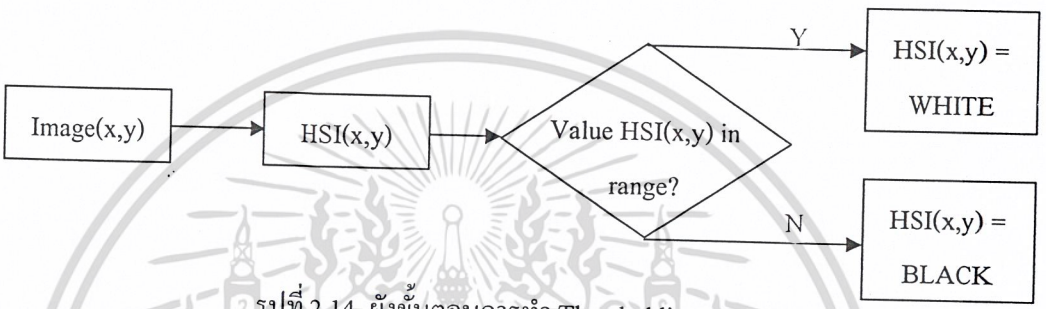
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 การแบ่งส่วนของภาพ (Image Segmentation)

การแบ่งส่วนของภาพ เป็นกระบวนการแยกส่วนที่สนใจออกจากส่วนอื่นๆ ของภาพซึ่งสามารถแบ่งส่วนของภาพได้หลายวิธี แต่ในโครงงานมีวิธีแบ่งส่วนอยู่ 2 ขั้นตอนดังนี้

2.6.1 Thresholding

เป็นการเปลี่ยนสีของพิกเซล (Pixel) ของภาพให้เป็นสีขาวหากค่าสีอยู่ในช่วงที่สนใจ (Threshold Range) หากไม่อยู่ในช่วงสีที่สนใจให้เปลี่ยนเป็นสีดำ การทำ Thresholding นี้ จะช่วยให้สามารถแยกช่วงสีที่สนใจออกมาจากส่วนอื่นของภาพได้ โดยสีเหล่านี้จะอยู่เป็นกลุ่มซึ่งแสดงด้วยจุดสีขาว เรียกว่า *Segment*



รูปที่ 2.14 ผังขั้นตอนการทำ Thresholding

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.2 Flood Fill

เป็นกระบวนการหาจุดศูนย์กลางของ segment โดยวิธีการของ Flood Fill คือ ทำการหาจุดสีขาวก่อน จากนั้นจะเริ่มขยายตัวไปยังจุดสีขาวรอบด้าน โดยดูว่าถ้ารอบข้างเป็นสีขาวก็ให้จุดขาวด้านข้างนั้นเป็นจุดเริ่มในการขยายตัวต่อไป จนครบทั้ง Segment เพื่อหาขอบเขตต่ำสุดและสูงสุดของ segment

ซึ่งเราจะใช้หลักการของ Flood Fill มาใช้ในการหาลายจุดของเสื้อ ซึ่งขั้นตอนการหาลายจุดจะอธิบายต่อในบทที่ 4 เรื่องการหาลวดลายจุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ภาษาจาวาสำหรับอุปกรณ์สื่อสารไร้สาย

3.1 ภาษาจาวาทั่วไป

หลังจากที่บริษัทซัน ไมโครซิสเต็มส์ออกจาวา 2 สแตนดาร์ทอิดิชั่น (Java 2 Standard Edition : J2SE) และจาวา 2 เอ็นเทอร์ไพร์สอิดิชั่น (Java 2 Enterprise Edition : J2EE) ก็ได้ออกแพลตฟอร์มใหม่เพิ่มเติม คือ J2ME ซึ่งได้รับการออกแบบโดยเน้นกลุ่มผู้ใช้อุปกรณ์และเครื่องใช้อิเล็กทรอนิกส์

3.2 ภาษาจาวาบนอุปกรณ์สื่อสารไร้สาย

ภาษาจาวาได้กลายเป็นแพลตฟอร์มการพัฒนาแอพพลิเคชั่นแบบเชิงวัตถุ (Object-oriented) ในอุปกรณ์และงานต่างๆ อย่างเต็มตัว นับตั้งแต่แอพพลิเคชั่นเซิร์ฟเวอร์ระดับองค์กร เครื่องคอมพิวเตอร์แบบตั้งโต๊ะทั่วไป จนถึงแอพพลิเคชั่นฝังตัวสำหรับอุปกรณ์ขนาดเล็ก

แพลตฟอร์มจาวา 2 ที่ใช้งานอยู่ในขณะนี้มี 3 รุ่นด้วยกัน

- จาวา 2 เอ็นเทอร์ไพร์สอิดิชั่น ใช้งานกับแอพพลิเคชั่นบนเครื่องเซิร์ฟเวอร์สำหรับองค์กรที่รองรับระบบงานใหญ่ๆ และไคลเอนต์จำนวนมาก
- จาวา 2 สแตนดาร์ทอิดิชั่น ใช้งานกับแอพพลิเคชั่นบนเครื่องคอมพิวเตอร์แบบตั้งโต๊ะทั่วไป
- J2ME ใช้งานกับแอพพลิเคชั่นรุ่นใหม่ซึ่งu3648 .น้นกลุ่มผู้ใช้อุปกรณ์อิเล็กทรอนิกส์และอุปกรณ์ฝังตัว (Embeded)

3.3 J2ME

J2ME เป็นแพลตฟอร์มจาวาที่ออกแบบมาเพื่อใช้กับแอพพลิเคชั่นที่ทำงานบนอุปกรณ์ขนาดเล็ก เช่น โทรศัพท์มือถือ พีดีเอ โทรศัพท์พร้อมจอภาพที่ต่อกับอินเตอร์เน็ต โทรทัศน์ดิจิตอลขนาดเล็ก อุปกรณ์บันเทิงและระบบนำทางในรถยนต์ สวิตซ์ในระบบเครือข่าย คอมโพเน็นท์ของเครื่องอำนวยความสะดวกภายในบ้าน ฯลฯ

J2ME ได้นำโครงสร้างแบบโมดูลที่มีความยืดหยุ่นสูงเข้ามาใช้ เพื่อให้สามารถสนับสนุนการทำงานอุปกรณ์หลากหลายประเภท J2MEกำหนดชั้นของซอฟต์แวร์ (Software) ไว้ 3 เลเยอร์ (Layer) ด้วยกัน โดยเลเยอร์ทั้งหมดจะอยู่เหนือชั้นระบบปฏิบัติการของอุปกรณ์ ดังนี้

3.3.1 เลเยอร์จาวาเวอร์ชวลแมชชีน (Java Virtual Machine) เป็นเลเยอร์ของจาวาเวอร์ชวลแมชชีนปรับแต่งให้เข้ากับระบบปฏิบัติการของอุปกรณ์ และรองรับแต่ละ คอนฟิกูเรชั่น (Configuration) ของ J2ME และเวอร์ชวลแมชชีน ของ J2MEได้แก่ (Connected Virtual Machine : CVM) และ (Kilobyte Virtual Machine : KVM) ทั้ง CVM และ KVM ต่างสนับสนุนยูติลิตี้ JavaCodeCompact หรือคลาส Prelinker preloader และ ROmnizer ยูติลิตี้นี้จะโยงจาวาคลาสเข้ากับเวอร์ชวลแมชชีน ช่วยลดระยะเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มใช้งาน VM ได้มากขึ้น ตัวอย่างเช่น เรียก KVM และ CLDC ขึ้นมาก่อนโหลดไปยังหน่วยความจำแบบ ROM ของอุปกรณ์ คลาส CLDC จะถูกโยงเข้าใน KVM โดยตรง เพื่อปรับปรุงประสิทธิภาพการทำงาน เราเรียกขั้นตอนนี้ว่าการโหลดเข้าหน่วยความจำแบบ ROM (ROMized)

3.3.2 เลเยอร์คอนฟิกูเรชั่น เป็นเลเยอร์ของคอนฟิกูเรชั่นของJ2ME ซึ่งกำหนดคลาสไลบรารี่อุปกรณ์ใช้งานทั่วไปหรือกลุ่มอุปกรณ์ที่มีความต้องการหน่วยความจำและหน่วยประมวลผลใกล้เคียงกัน คอนฟิกูเรชั่นในจาวาสำหรับอุปกรณ์ขนาดเล็ก มี 2 ประเภทด้วยกัน คือ CDC (Connected Device Configuration) และ CLDC (Connected Limited Device Configuration)

3.3.3 เลเยอร์โพรไฟล์ (Profile) เป็นเลเยอร์ที่สร้างเหนือเลเยอร์คอนฟิกูเรชั่น โดยกำหนดคลาสไลบรารี่เพื่อสนองตอบความต้องการขอตลาดเฉพาะกลุ่ม ตัวอย่างของโพรไฟล์ในเลเยอร์นี้ ได้แก่ PDA Profile , MID Profile , Foundation Profile และ Personal Profile

3.4 คอนฟิกูเรชั่นใน J2ME

คอนฟิกูเรชั่นและโพรไฟล์เป็นองค์ประกอบหลักของ J2ME โดยมีจุดสำคัญ คือ เพื่อปรับแต่งเวอร์ชวลแมชชีนและคลาสไลบรารี่ให้เหมาะสมกับอุปกรณ์แต่ละประเภท คอนฟิกูเรชั่น คือ ชุดที่มีคุณสมบัติขั้นต่ำของ จาวาเวอร์ชวลแมชชีน และจาวาคลาสสำหรับอุปกรณ์แต่ละประเภท เป็นตัวแทนของอุปกรณ์ที่มีลักษณะคล้ายคลึงกัน อาจกล่าวได้ว่า คอนฟิกูเรชั่นเป็นตัวกำหนดคุณสมบัติหรือไลบรารี่ขั้นต่ำของแพลตฟอร์มจาวาที่นักพัฒนาคาดว่าจะต้องมีในทุกอุปกรณ์ ขณะที่คลาสไลบรารี่ที่กำหนดในคอนฟิกูเรชั่นจะมีในทุกอุปกรณ์ที่อยู่ในกลุ่มเดียวกัน

ปัจจุบันคอนฟิกูเรชั่นใน J2ME แบ่งเป็น 2 ประเภทคือ CDC (Connected Device Configuration) และ CDC (Connected Limited Device Configuration)

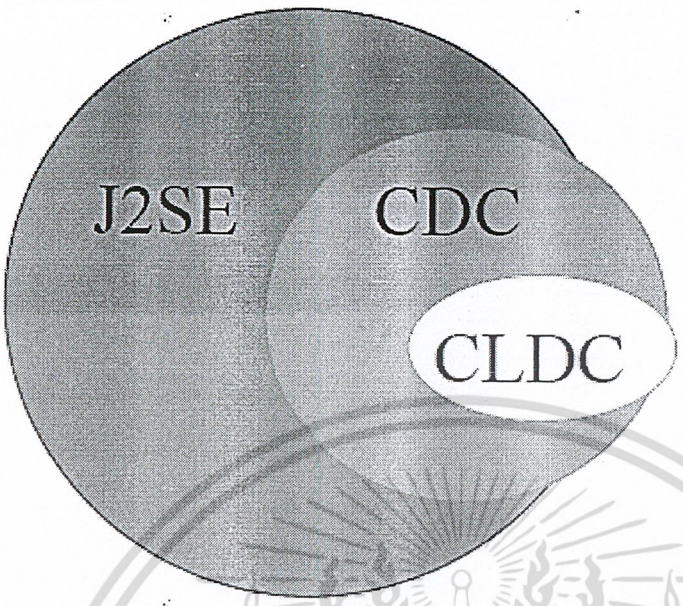
3.4.1 CDC เป็นอุปกรณ์ใช้ร่วมกับอุปกรณ์อื่น ติดตั้งตายตัว และใช้เชื่อมต่อข้อมูล โดยปรกติมักมีส่วนติดต่อกับผู้ใช้หลากหลายแบบ มีหน่วยความจำประมาณ 2-16 เม็กกะไบต์ ใช้หน่วยประมวลผลแบบ 32 บิตหรือมากกว่า เชื่อมต่อเครือข่ายที่มีแบนด์วิดท์ (Bandwidch) สูงอย่างต่อเนื่องโดยอาศัยพอร์ต TCP/IP ตัวอย่างอุปกรณ์ประเภทนี้ ได้แก่ โทรทัศน์ดิจิตอลขนาดเล็ก อินเตอร์เน็ตทีวี โทรศัพท์พร้อมจอภาพที่เชื่อมต่อกับอินเตอร์เน็ต อุปกรณ์สื่อสารที่มีความซับซ้อนสูง อุปกรณ์บันเทิงและระบบนำทางในรถยนต์

3.4.2 CLDC เป็นอุปกรณ์ส่วนบุคคล พกพาได้ และใช้เชื่อมต่อข้อมูล โดยปรกติมักมีส่วนติดต่อกับผู้ใช้แบบง่ายๆเมื่อเทียบกับระบบบนเครื่องคอมพิวเตอร์ตั้งโต๊ะ มีหน่วยความจำประมาณ 128 กิโลไบต์ – 1 เม็กกะไบต์ ใช้หน่วยประมวลผลแบบ 16 หรือ 32 บิต เชื่อมต่อกับเครือข่ายที่มีแบนด์วิดท์ต่ำเป็นระยะเวลาสั้นๆ โดยไม่อาศัยพอร์ต TCP/IP ตัวอย่างอุปกรณ์ประเภทนี้ ได้แก่ โทรศัพท์มือถือแบบไม่ซับซ้อนมากนัก เพจเจอร์รับส่งข้อความ เครื่องปาล์มโอเอสแบบพกพา

ในชั้นของคอนฟิกูเรชั่น มีคลาส 2 ประเภทด้วยกัน คือ คลาสที่นำมาจาก J2SE และที่ออกแบบเฉพาะอุปกรณ์ขนาดเล็ก คลาสที่นำมาจาก J2SE จะมีคุณสมบัติอย่างเดียวกันกับคลาสใน J2SE หรือเป็นซับคลาสของ J2SE เช่น แพ็คเก็จ java.io และ java.util จากรูป 2.1 แสดงความสัมพันธ์ระหว่าง J2SE และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คลาสไลบรารี่ของ CDC และ CLDC จากภาพจะเห็นว่าคลาสใน CLDC ส่วนใหญ่ทำงานร่วมกับคลาสใน CDC ได้ดีเช่นเดียวกับกรณีของคลาสใน CLDC และคลาสใน J2SE



รูปที่ 3.1 แสดงความสัมพันธ์ระหว่าง J2SE และคลาสไลบรารี่ใน CDC และ CLDC

คลาสที่ไม่ได้นำมาจาก J2SE และออกแบบเพื่อใช้เฉพาะอุปกรณ์มักทำงานร่วมกับ J2SE ได้ไม่ค่อยดีใน CLDC คลาสเหล่านี้จะอยู่ในกลุ่มกรอบการติดต่อสื่อสารทั่วไป (Generic Connection Framework) โดยระบุไว้ในแพ็คเก็จ javax.microedition.io

คอนฟิกูเรชั่นยังระบุคุณสมบัติของจาวาเวอร์ชวลแมชชีน ในเลเยอร์ด้านล่างอีกด้วย ในโครงสร้างปัจจุบัน CDC และ CLDC มีเวอร์ชวลแมชชีนที่ปรับแต่งมาเฉพาะตัวอยู่แล้ว เวอร์ชวลแมชชีนของ CDC คือ C Virtual Machine (CVM) มีคุณสมบัติครบถ้วนเหมือน Java 2 Virtual Machine แต่ขนาดเล็กกว่า ออกแบบมาสำหรับอุปกรณ์ที่สลับซับซ้อน CVM มีความต้องการหน่วยความจำ 256 กิโลไบต์ ขณะที่หน่วยความจำแบบ ROM ของ CDC มีขนาด 1 เม็กกะไบต์ เวอร์ชวลแมชชีนของ CLDC คือ K Virtual Machine (KVM) แม้จะมีขนาดเล็กแต่มีความสามารถในการทำงานสูง ออกแบบมาเพื่อใช้งานกับอุปกรณ์ที่มีทรัพยากรจำกัด K ใน KVM หมายถึง กิโล โดยเรียกตามหน่วยความจำที่นับเป็นกิโลไบต์ ส่วนคอมพิวเตอร์แบบตั้งโต๊ะนั้นหน่วยนับเป็นเม็กกะไบต์ KVM เหมาะจะนำมาใช้กับหน่วยประมวลผล RISC/CISC แบบ 16/32 บิต ซึ่งมีหน่วยความจำทั้งหมดไม่กี่ร้อยกิโลไบต์เท่านั้นประมาณ 128 กิโลไบต์ ปัจจุบัน KVM มีความต้องการหน่วยความจำอยู่ในช่วง 40 – 80 กิโลไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 โพรไฟล์ของ J2ME

โพรไฟล์จะกำหนดชุดของ API ที่ต้องใช้เพิ่มเติม ตลอดจนกำหนดคุณสมบัติที่เป็นที่ต้องการของตลาดเฉพาะกลุ่มหรืออุปกรณ์เฉพาะประเภท คลาสไลบรารี่ในโพรไฟล์ช่วยให้นักพัฒนาสร้างฟังก์ชั่นเฉพาะอุปกรณ์ เช่น ส่วนติดต่อกราฟิกกับผู้ใช้การเชื่อมต่อเครือข่าย หน่วยเก็บข้อมูล Persistent Storage

ฯลฯ ตามปรกติแล้วจะไม่สามารถนำคลาสไลบรารี่ที่สร้างเพื่อใช้งานกับโพรไฟล์หนึ่งไปใช้กับโพรไฟล์อื่นได้ ขณะนี้มีบางโพรไฟล์ได้ถูกกำหนดแล้ว แต่ก็มีหลายโพรไฟล์ที่ยังอยู่ในขั้นตอนการดำเนินการอยู่ โพรไฟล์ Foundation และ RMI ซึ่งสร้างบน CDC กำลังเผยแพร่อยู่ ส่วนที่สร้างบน CLDC มีโพรไฟล์เดียว คือ MID Profile (Mobile Information Device Profile : MIDP) มันจะเตรียมส่วนติดต่อผู้ใช้และหน่วยเก็บข้อมูล Persistent Storage ความสามารถด้านเครือข่าย แบบจำลอง API สำหรับแอพพลิเคชั่น ไว้ให้อุปกรณ์ไร้สาย เช่น โทรศัพท์ที่ไม่ซับซ้อนมากนักและเพจเจอร์รับส่งข้อความ ส่วนโพรไฟล์ พีดีเอ (PDAP) กำลังได้รับการวิจัยอยู่

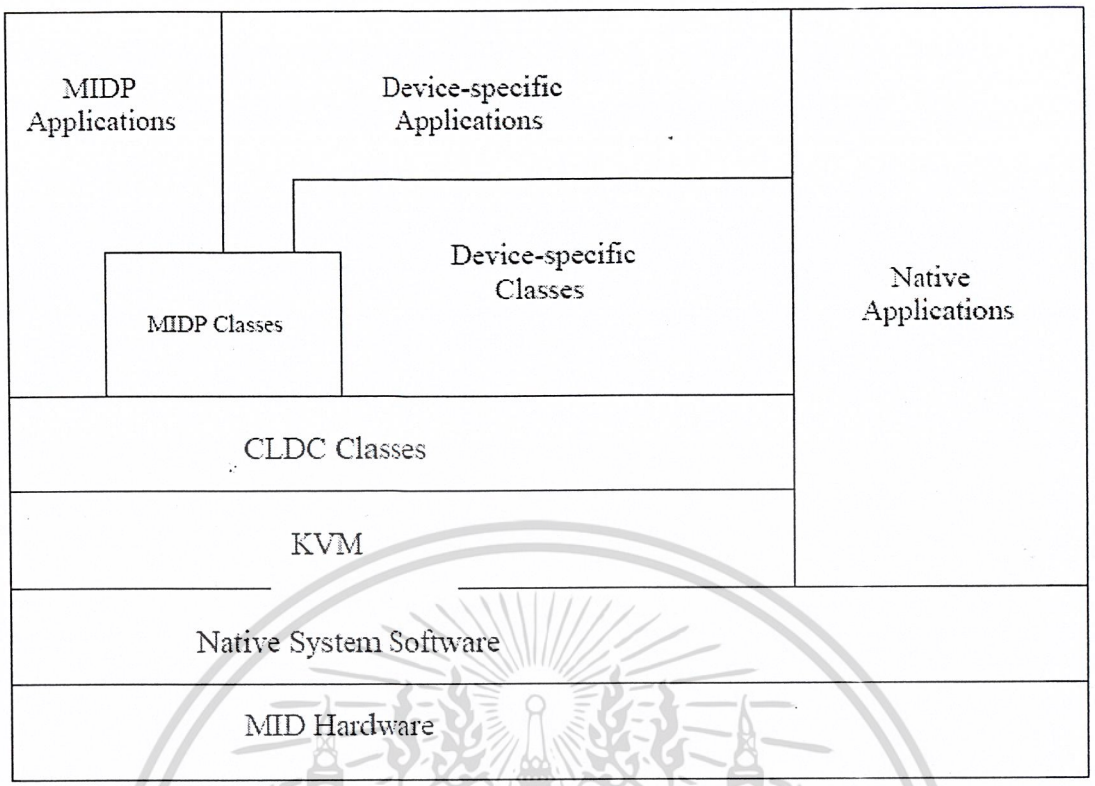
อุปกรณ์หนึ่งๆอาจมีโพรไฟล์ใช้งานมากกว่า 1 ชนิด และบางโพรไฟล์ใช้งานเฉพาะบางอุปกรณ์หรือแอพพลิเคชั่นเท่านั้น ตัวอย่างเช่น โพรไฟล์บน CDC ส่วนใหญ่ เช่น RMI และ Personal จะสร้างไว้เหนือ Profile Foundation หากไม่มี Profile Foundation และ CDC รองรับแอพพลิเคชั่นที่เขียนก็ไม่สามารถทำงานได้

3.6 J2ME สำหรับอุปกรณ์ไร้สาย

J2ME ได้ให้กำเนิดแอพพลิเคชั่นยุคใหม่บนอุปกรณ์ไร้สาย ช่วยให้เกมส์แบบหลายผู้เล่นที่ทำงานผ่านอินเตอร์เน็ตการทำธุรกรรมทางโทรศัพท์มือถือ แอพพลิเคชั่นสำหรับองค์กรทั้งไคลเอนต์และเซิร์ฟเวอร์ เกิดขึ้นได้บนโทรศัพท์มือถือและเพจเจอร์รับส่งข้อความ MIDP CLDC และ _____KVM ได้กลายมา

เป็นรากฐานในการพัฒนาจาวาแอพพลิเคชั่นบนอุปกรณ์ไร้สายยุคใหม่ มี 3 ปัจจัยในการสร้างแพลตฟอร์มสำหรับ แอพพลิเคชั่นไร้สาย คือ MIDP CLDC KVM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 โครงสร้างของเลเยอร์ต่างๆ

โครงสร้างของเลเยอร์ต่างๆจากล่างขึ้นบน ดังนี้

1. เลเยอร์ฮาร์ดแวร์ MID หมายถึง ตัวโทรศัพท์มือถือ

2. เลเยอร์ซอฟต์แวร์ของระบบที่ติดตั้งมากับอุปกรณ์ หมายถึง ระบบปฏิบัติการและไลบรารี่ของระบบที่บริษัทผู้ผลิตให้มา

3. เลเยอร์ KVM เป็นส่วนที่เตรียม Runtime Environment ไว้ให้แอพพลิเคชั่นบนอุปกรณ์ไร้สาย

4. เลเยอร์ CLDC เป็นส่วนที่ตรียม API หลักของจาวาให้แอพพลิเคชั่นบนอุปกรณ์ไร้สาย

5. เลเยอร์ MIDP เป็นส่วนที่เตรียมไลบรารี่สำหรับส่วนติดต่อกราฟิกกับผู้ใช้ และหน่วยเก็บข้อมูล Persistent Storage ระบบเครือข่าย และไทม์เมอร์

นอกจากคลาสไลบรารี่สำหรับ MIDP ผู้ผลิตอาจเตรียมคลาสไลบรารี่เฉพาะอุปกรณ์ไว้ให้นักพัฒนา เพื่อดึงความสามารถของฟังก์ชั่นที่มีอยู่แล้วไปใช้งานให้เกิดประโยชน์สูงสุด อาทิ การโทรศัพท์ การแชร์ข้อมูลกับแอพพลิเคชั่นที่ติดตั้งมาในเครื่อง จำพวก ปฏิทิน สมุดจดที่อยู่ การตรวจสอบข้อมูลอุปกรณ์ เช่น อายุแบตเตอรี่ ความแรงของสัญญาณ ฯลฯ แม้ว่าการนำคลาสเฉพาะอุปกรณ์ที่ผู้ผลิตเตรียมไว้มาใช้งาน จะช่วยเพิ่มความสามารถแก่แอพพลิเคชั่นบนอุปกรณ์ไร้สาย แต่ไม่สามารถเคลื่อนย้ายไปสู่อุปกรณ์อื่นที่ใช้ MIDP ได้ เนื่องจากคลาสที่นำมาใช้อยู่นอกเหนือขอบเขตของ MIDP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

3.7 ความต้องการของระบบ

3.7.1 อุปกรณ์ไร้สายจะทำงานสนับสนุน J2ME ได้ดีเมื่อมีคุณสมบัติตรงตามข้อกำหนด หากต้องการให้ KVM ทำงานได้อย่างมีประสิทธิภาพ โดยใช้ไลบรารี่ CLDC จะต้องมีคุณสมบัติของระบบขั้นต่ำ ดังนี้

- มีหน่วยความจำ 160 – 512 กิโลไบต์ สำหรับสร้างแพลตฟอร์มจาวา
- มีหน่วยประมวลผลแบบ 16-32 บิตความเร็ว 25 เม็กกะเฮิรซ์
- ใช้พลังงาน โดยมากมักทำงานโดยใช้แบตเตอรี่
- เชื่อมต่อกับเครือข่ายได้ในช่วงสั้นๆ อาศัยระบบไร้สายเป็นส่วนใหญ่ แบนด์วิดท์ค่อนข้างจำกัด ความเร็ว 9600 ไบต์ต่อวินาทีหรือน้อยกว่า
- มีหน่วยความจำชั่วคราวขนาด 32 กิโลไบต์ สำหรับเก็บจาวา รันไทม์และหน่วยความจำของอ็อบเจกต์

3.7.2 การใช้งาน MIDP ฮาร์ดแวร์จะต้องมีคุณสมบัติดังนี้

3.7.2.1 การแสดงผล

- หน้าจอขนาด 96*54
- ความลึกของสี 1 บิต
- สัดส่วนของภาพ 1:1

3.7.2.2 การรับข้อมูลเข้า

- ใช้กลไกการป้อนข้อมูลอย่างไรอย่างหนึ่ง อาทิ แป้นพิมพ์ หรือจอสัมผัส

3.7.2.3 หน่วยความจำ

- หน่วยความจำถาวรขนาด 128 กิโลไบต์สำหรับเก็บคอมโพเน็นท์ของ MIDP
- หน่วยความจำถาวรขนาด 8 กิโลไบต์สำหรับเก็บข้อมูลที่แอพพลิเคชั่นสร้างขึ้น
- หน่วยความจำชั่วคราวขนาด 32 กิโลไบต์ สำหรับเก็บจาวา รันไทม์ เช่น จาวา ฮีป

3.7.2.4 เครือข่าย

- เครือข่ายรับส่งข้อมูลผ่านระบบไร้สาย เชื่อมต่อได้ในช่วงสั้นๆ และมีแบนด์วิดท์จำกัด

61337
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทำงานของโปรแกรม

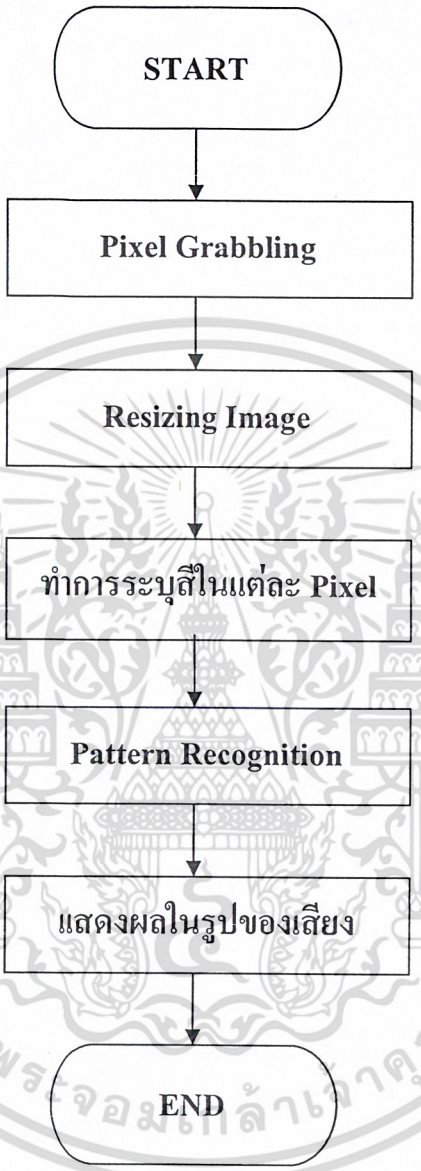
4.1 โครงสร้างของโปรแกรมตรวจจับสีและลวดลายเสื้อผ้าแสดงผลบนโทรศัพท์มือถือ

โปรแกรมการตรวจจับสีและลวดลายเสื้อผ้าแสดงผลบนโทรศัพท์มือถือมีขั้นตอนดังต่อไปนี้ คือ ในขั้นแรกคือการทำเทคนิคที่เรียกว่า pixel grabbing เพื่อทำการเก็บค่า A R G B แต่ละ pixel ของรูปที่นำมาประมวณผลไว้ใน อาเรย์ที่กำหนด จากนั้นจะทำการลดขนาดของภาพ (Resizing) ของภาพที่ต้องการประมวณผล โดยทำการลดขนาดของภาพให้เหลือขนาด กว้าง 30 pixel ยาว 30 pixel จากการนำข้อมูลในอาเรย์ที่เก็บค่าไว้ เมื่อผ่านขั้นตอนนี้แล้วจะทำการระบุสีของแต่ละ pixel ของรูปที่ resize แล้วว่าเป็นสีอะไร โดยใช้ค่า R G B ของแต่ละ pixel มาคำนวณ โดยใช้ Color Model ที่เรากำหนดเป็นเครื่องมือช่วยในการระบุสีใน Pixel นั้น เมื่อทำการระบุสีได้แล้วจะทำการเปลี่ยนข้อมูลในอาเรย์จากเดิมที่เป็นค่า A,R,G,B ของแต่ละ Pixel ให้เป็นค่าสีที่ระบุ จากนั้นทำการหาลายเสื้อโดยวิธีการ pattern recognition จากข้อมูลในอาเรย์ ซึ่งขั้นตอนทั้งหมดนี้ถูกสร้างขึ้นโดยใช้ภาษา J2ME ซึ่งได้แบ่งโปรแกรมออกเป็นส่วนของโปรแกรมหลักและโปรแกรมย่อยเพื่อแยกการประมวณผลออกเป็นส่วนๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 โปรแกรมหลัก

ส่วนของโปรแกรมหลัก จะมีขั้นตอนในการเรียกใช้โปรแกรมย่อยดังแสดงในรูปที่ 4.1

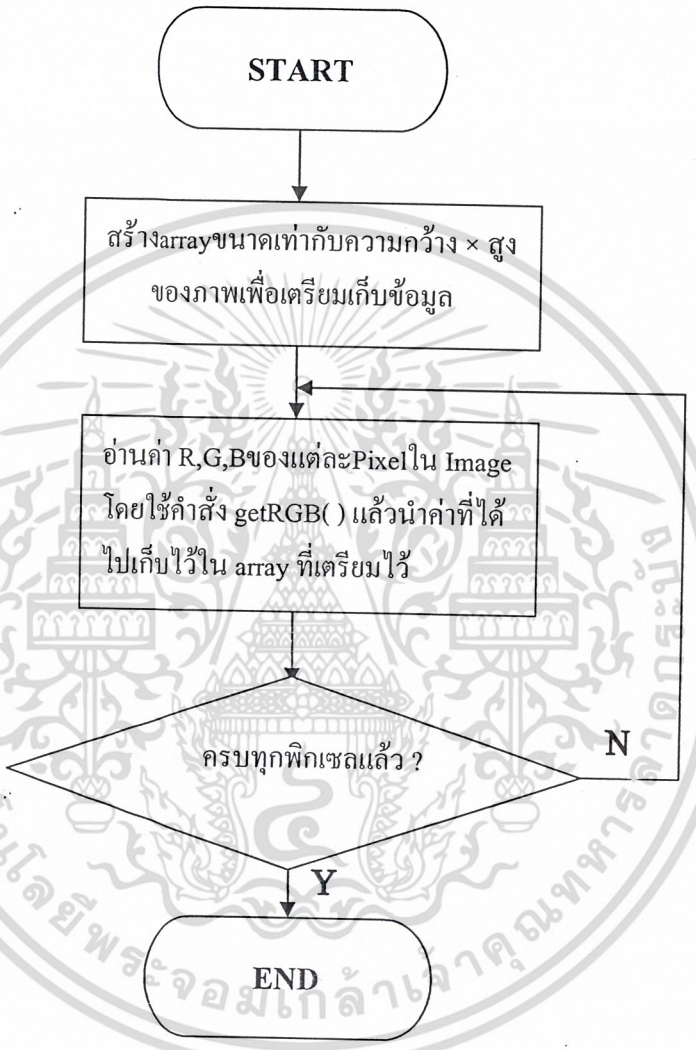


รูปที่ 4.1 แผนผังการทำงานของโปรแกรมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ขบวนการทำ Pixel Grabbing

คือการเก็บค่า Pixel ของรูปภาพไว้ใน Array ที่เรากำหนดเพื่อนำข้อมูลที่อยู่ใน Array ไปใช้ในการทำ image processing หรือขั้นตอนอื่นๆ ต่อไปขั้นตอนการทำ Pixel Grabbing แสดงได้ดังแผนผังที่ 4.2



รูปที่ 4.2 แผนผังการทำงาน Pixel Grabbing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ขบวนการทำ Resizing Image

การลดขนาดของภาพ (Resizing) ของภาพที่ต้องการประมวณผล คือทำการลดขนาดของภาพให้เหลือขนาด 30 pixel × 30 pixel ตามที่เรากำหนดซึ่งโปรแกรมสามารถ Resizing ได้ 2 แบบคือ

แบบที่ 1 point sample

เป็นวิธีการแรกที่ใช้ในการ resize หลักการก็คือเมื่อเรากำหนดขนาดภาพที่เราต้องการจะ resize ว่าจะให้มีขนาดลดลงเป็นกี่ pixel ขนาดกว้างยาวเป็นเท่าไหร่เราสามรถหาค่า R G B ของแต่ละ pixel โดยการจับคู่กับ pixel ที่เหมาะสมบนภาพต้นแบบโดยค่า R G B ของ pixel ที่จับคู่ได้นั้นจะกำหนดให้มีค่าเท่ากันวิธีการนี้แสดงได้ source code ดังนี้

```
for (int destY = 0; destY < destH; ++destY) {
for (int destX = 0; destX < destW; ++destX) {
        int srcX = (destX * srcW) / destW;
        int srcY = (destY * srcH) / destH;
        destPixels[destX + destY * destW] = srcPixels[srcX + srcY * srcW]; } }
```

src คือ source image หรือภาพที่ต้องการจะ resize

srcW คือค่าความกว้างของภาพที่ต้องการจะ resize

srcH คือค่าความสูงของภาพที่ต้องการจะ resize

dest คือ destination image หรือภาพที่ resize เสร็จแล้ว

destW คือค่าความกว้างของรูปที่ resize เสร็จแล้ว

destH คือค่าความสูงของรูปที่ resize เสร็จแล้ว

แบบที่ 2 box filter

เป็นวิธีการที่ 2 ที่ใช้ในการ resize หลักการคือจะแบ่งขั้นตอนการ resize เป็น 2 ขั้นตอนคือ Horizontal resampling และ Vertical resampling และขั้นตอนย่อยที่สำคัญที่ทำให้การ resize แบบ Box filter นี้ที่ทำให้ได้ภาพออกมามีความเหมือนจริงกับภาพต้นแบบมากกว่าแบบแรกก็คือ ในการทำ resampling นั้นจะทำการเฉลี่ยค่าสีในแต่ละ pixel ข้างเคียงในภาพต้นแบบ ก่อนที่จะแทนค่าสีที่ได้ลงไปใน pixel ปลายทางที่เป็นของรูปที่ resize แล้ว

ขั้นตอนการทำ resize แบบ box filter

1.ทำการ Horizontal resampling นั่นคือการทำ resampling โดยให้ค่าความสูงเท่าเดิมหรือทำการresampling เป็นแถวๆ นั่นเองแล้วเก็บค่าที่ได้ไว้ใน temporary buffer

1.1 ในการทำ resampling ในแต่ละแถวนั้นจะนำค่าสีคือ A R G B ของ pixel ข้างเคียงมาเฉลี่ยเพื่อหาค่าสีเฉลี่ยเพื่อเป็นตัวแทนต่อไปซึ่งจำนวน pixel ที่นำมาคิดนั้นได้จากการคำนวณโดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อาศัยความสัมพันธ์ระหว่างความกว้างของรูปภาพต้นแบบกับความกว้างของรูปภาพที่ resize เรียบร้อยแล้ว

1.2 ในการทำการเฉลี่ยสีนั้นเราต้องทำการแยกค่าสีออกจากค่าสีรวมโดยใช้ source code แยกค่าสีดังนี้

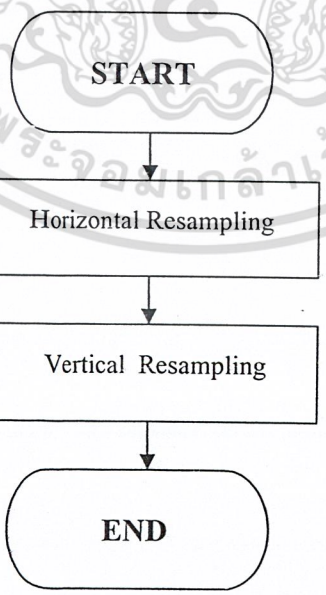
```
argb = srcPixels[ ];
a = (argb & 0xff000000) >> 24); // alpha channel
r = (argb & 0x00ff0000) >> 16); // red channel
g = (argb & 0x0000ff00) >> 8); // green channel
b = (argb & 0x000000ff); // blue channel
```

1.3 ทำการรวมค่าสีแต่ละค่าสีคือแต่ละค่าของ a r g และ b ไว้ด้วยกันและนำค่าที่ได้หารด้วยจำนวน pixel ที่ทำการรวมค่าสีจะได้ค่าเฉลี่ยแต่ละสีออกมาจากนั้นทำการรวมค่าสีไว้ด้วยกันโดย source code ดังนี้

```
temporary buffer [ ] = ( (a << 24) | (r << 16) | (g << 8) | b);
```

2.ทำการ Vertical resampling นั่นคือการนำค่าที่ได้จากการทำ Horizontal resampling ที่อยู่ในtemporary buffer มาทำการ resample ในแต่ละหลักโดยให้ค่าความกว้างเท่าเดิมโดยขั้นตอนการ resampling ทำเหมือนกับการทำ Horizontal resampling โดยค่าที่ได้จะเก็บไว้ใน destPixel เพื่อทำการเรียกแสดงผลบนหน้าจอต่อไปหรือใช้ทำการ image processing ต่อไป

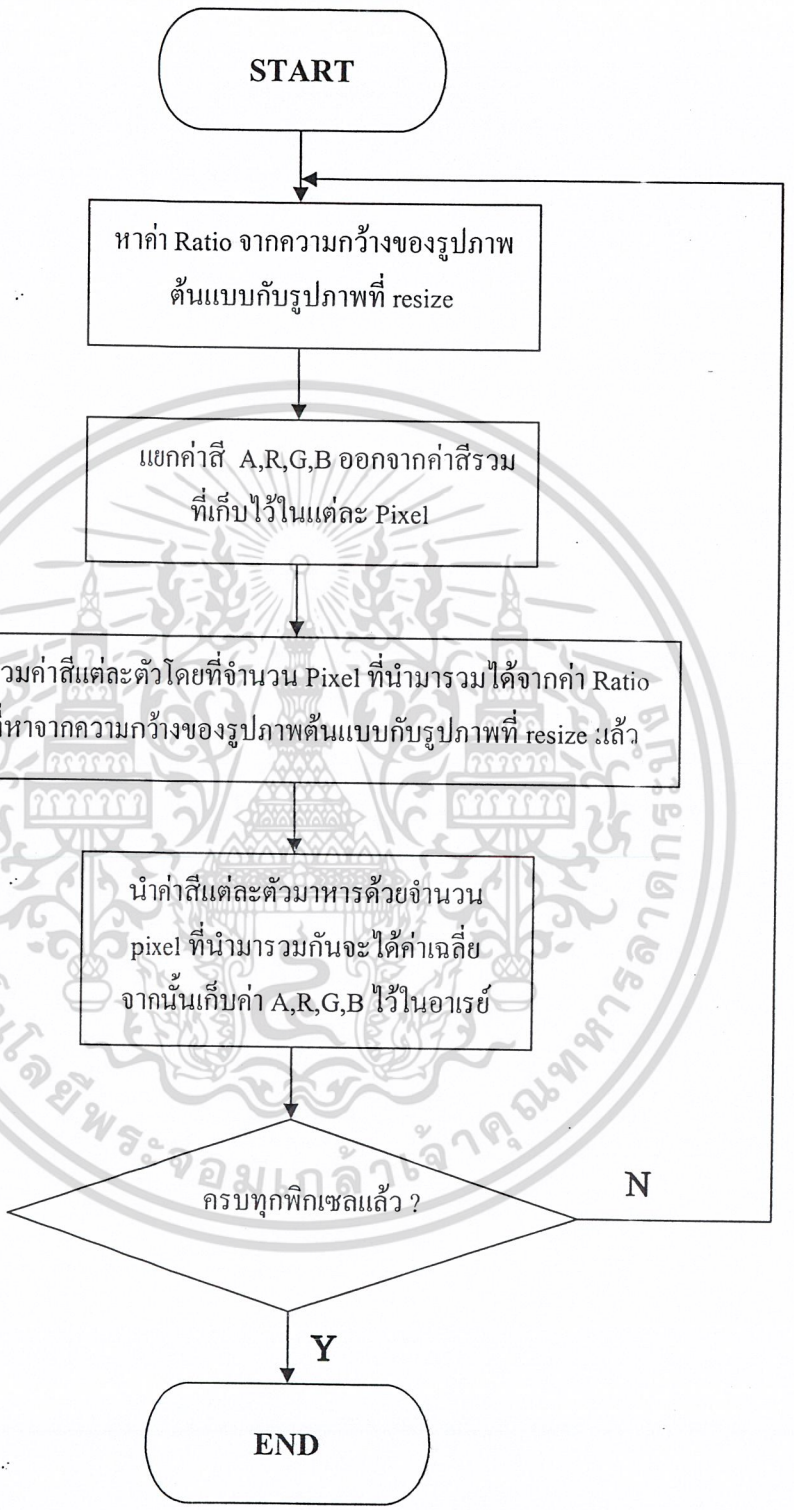
ขั้นตอนการทำ Resize แบบ box filter แสดงได้ดังแผนผังรูปที่ 4.3



รูปที่ 4.3 แผนผังการทำงาน Box Filter Resizing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

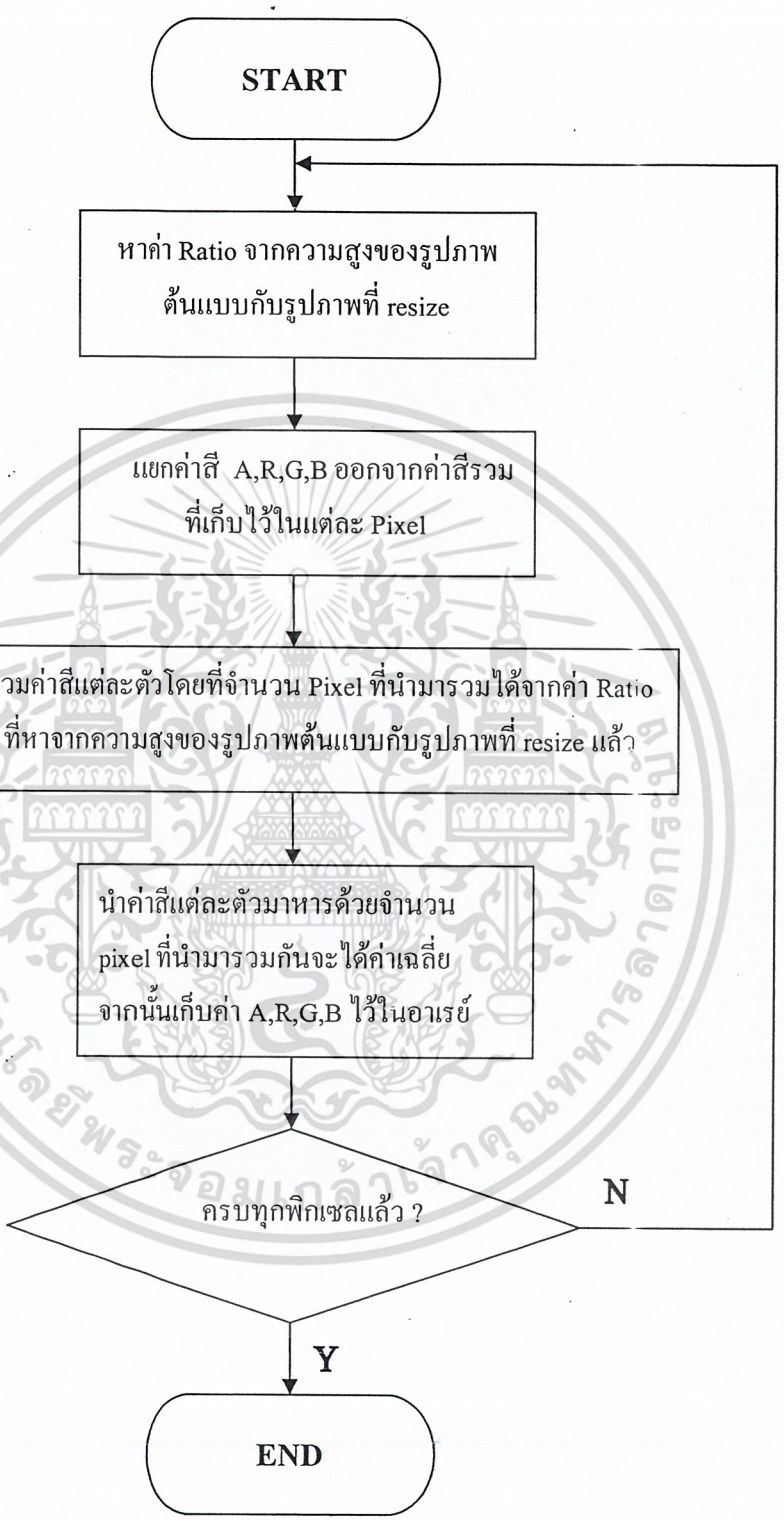
ขั้นตอนการทำ Horizontal Resampling แสดงได้ดังแผนผังที่ 4.4



รูปที่ 4.4 แผนผังการทำงาน Horizontal Resampling

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทำ Vertical Resampling แสดงได้ดังแผนผังที่ 4.5



รูปที่ 4.5 แผนผังการทำงาน Vertical Resampling

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทำการระบุสีแต่ละพิกเซลแสดงได้ดังแผนผังที่ 4.7

รูปที่ 4.7 แผนผังการระบุสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 ขบวนการทำ Pattern Recognition

หลังจากทำการระบุสีแต่ละ pixel แล้วค่าที่อยู่ในอาเรย์จะเปลี่ยนเป็นค่า integer 1-7 ซึ่งมีความหมายดังนี้คือ

เลข 1 แทนค่าสีดำ

เลข 2 สีน้ำเงิน

เลข 3แทนค่าสีเขียว

เลข 4 แทนค่าสีแดง

เลข 5 แทนค่าสีเหลือง

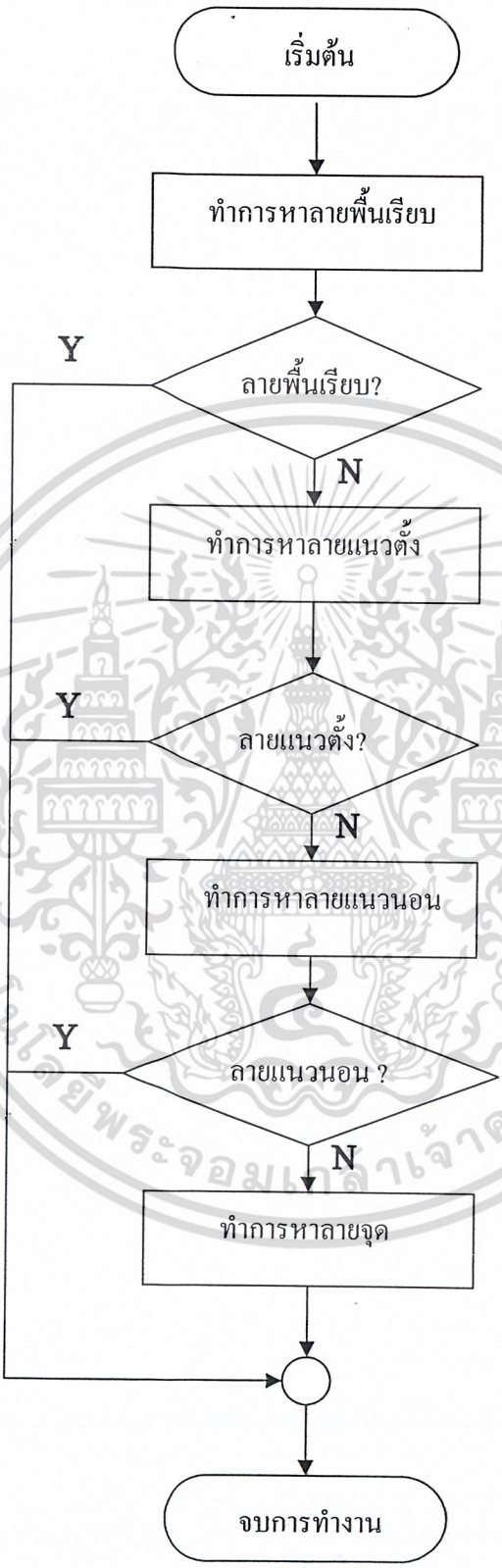
เลข 6 แทนค่าสีส้ม

เลข 7 แทนค่าสีขาว

และจำนวนพิกเซลหลังจาก resize แล้วจะเหลือขนาด 30×30 พิกเซล ซึ่งจะมีอาเรย์ในการเก็บข้อมูล 900 ช่องซึ่งขั้นตอนที่จะทำต่อไปนี้คือการหาลวดลายของเสื้อโดยใช้ทฤษฎี pattern recognition โดยมีขั้นตอนดังนี้คือ เราจะหาลวดลายของเสื้อตามลำดับดังนี้คือ ลายสีพื้นเรียบ ลายนอน ลายตั้ง และลายจุด โดยขั้นตอนย่อยของแต่ละลายจะอธิบายต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทำ pattern recognition อธิบายได้ดังแผนผังรูปที่ 4.8

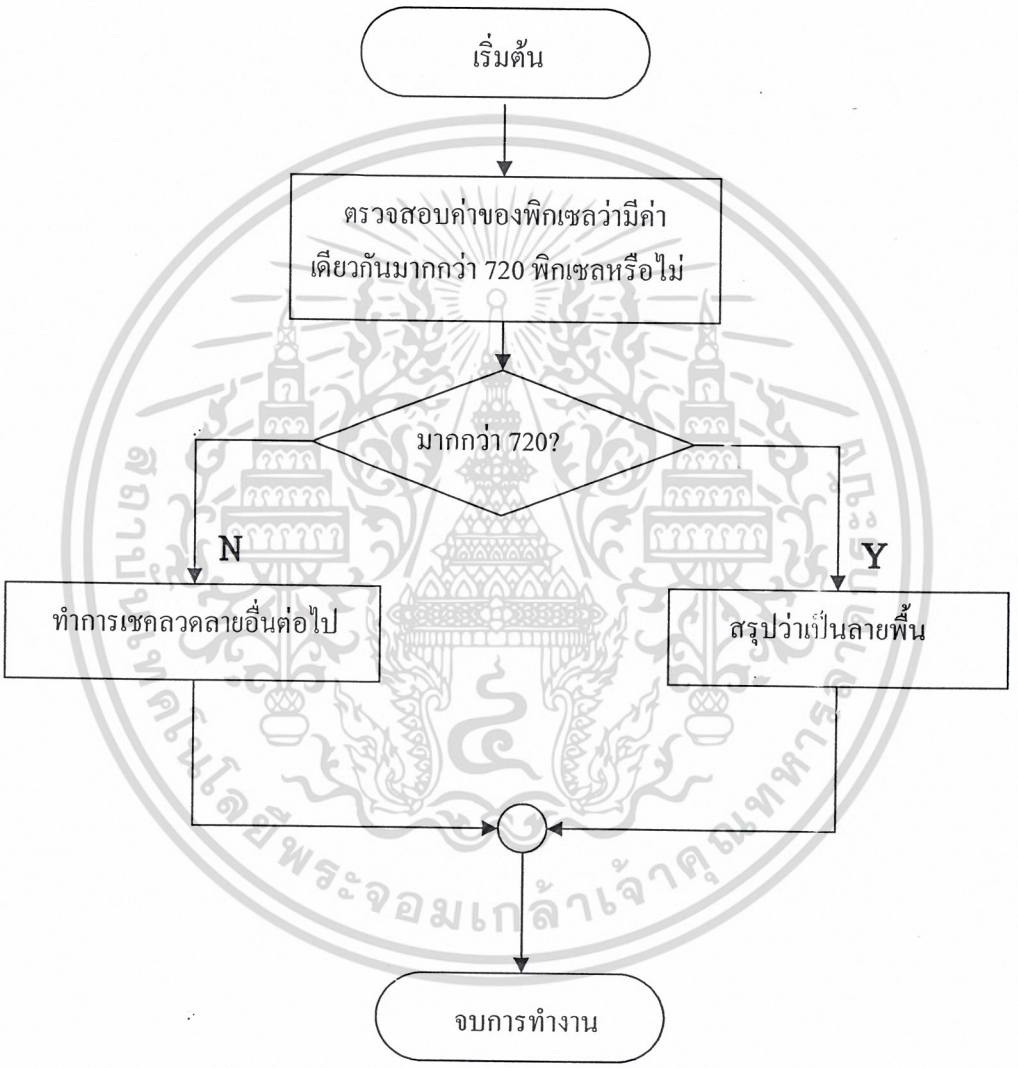


รูปที่ 4.8 แผนผังการทำ pattern recognition

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6.1 ขวนการหาลวดลายพื้นเรียบ

หลักการในการหาพื้นเรียบคือ เมื่อตรวจดูค่าในอาเรย์ที่เก็บค่าพิกเซลของรูปแล้ว ซึ่งรูปที่ resize แล้วจะมีขนาด 900 พิกเซล เราจะพิจารณาว่าถ้าในอาเรย์มีค่าเดียวกันมากกว่า 720 พิกเซล แสดงว่าเป็นลายพื้นเรียบถ้าไม่จะทำการเชคลายอื่นๆต่อไปที่ต้องตรวจสอบแค่เพียง 720 ก็เพราะว่าในการถ่ายภาพจริงๆ อาจจะมีการผิดเพี้ยนของสีเนื่องจากแสงที่ตกกระทบไม่เท่ากันได้ดังนั้นจึงกำหนดค่าขั่นต่ำในการทดสอบลวดลายพื้นเรียบขึ้นมาคือ 720 ขั้นตอนการทำงานดังแผนผังรูปที่ 4.9

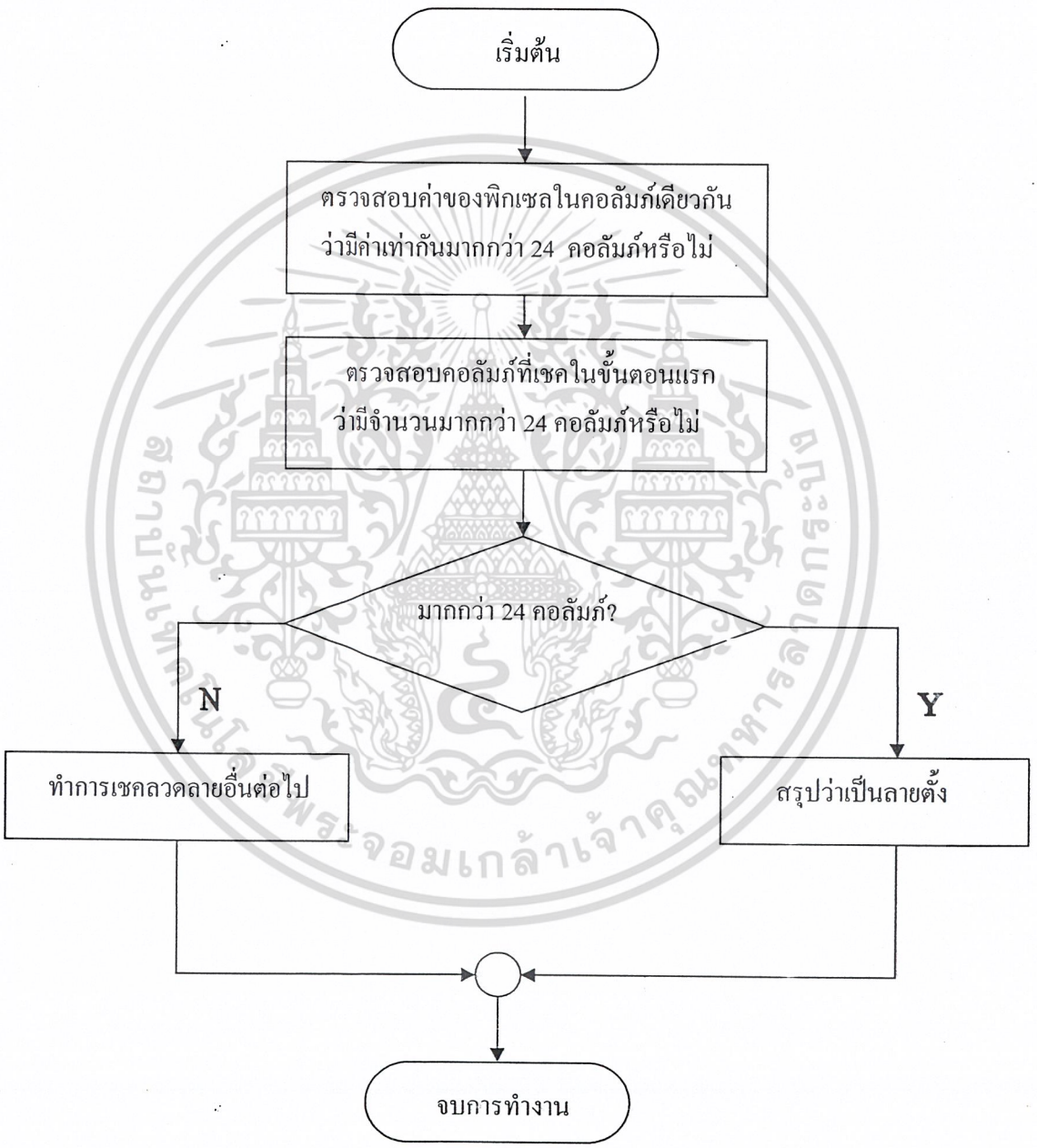


รูปที่ 4.9 แผนผังการหาพื้นเรียบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6.2 ขบวนการหาลวดลายแนวตั้ง

หลักการในการหาลายแนวตั้งคือ ทำการเชคค่าพิกเซลในคอลัมภ์เดียวกันว่ามีค่าเดียวกันหรือไหม โดยที่ต้องมีค่าเดียวกันมากกว่า 24 พิกเซลถ้ามากกว่า 24 พิกเซลจะสรุปว่าเป็นเส้นแนวตั้ง และจะต้องมีเส้นตั้งทั้งหมดมากกว่า 24 เส้นจึงจะสรุปได้ว่าเป็นลายแนวตั้ง ขั้นตอนการทำงานดังแผนผังรูปที่ 4.10

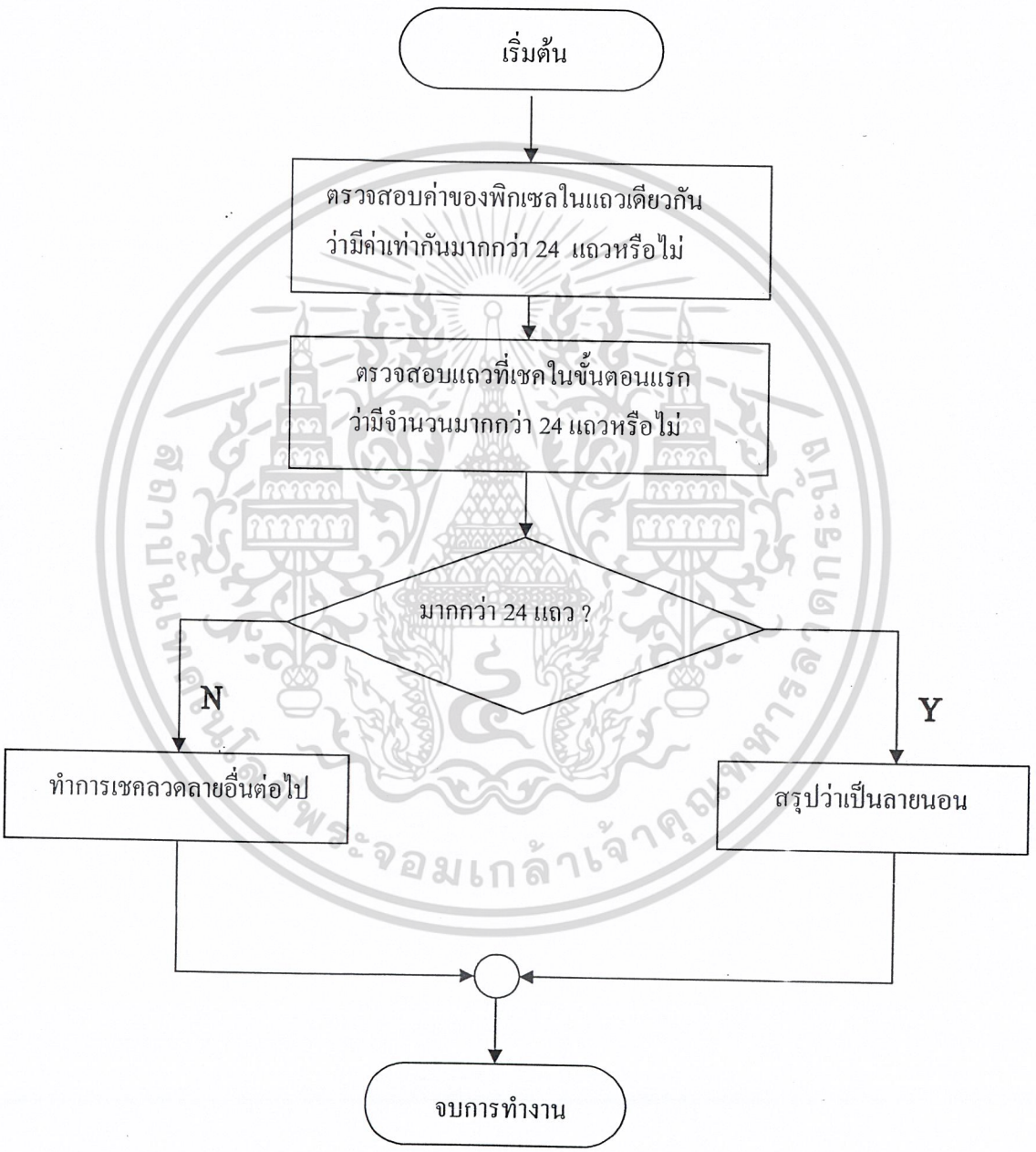


รูปที่ 4.10 แผนผังการหาลวดลายแนวตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6.3 ขบวนการหาลวดลายแนวนอน

หลักการในการหาลายแนวนอนคือ ทำการเชคค่าพิกเซลในแถวเดียวกันว่ามีค่าเดียวกันหรือไหม โดยที่ต้องมีค่าเดียวกันมากกว่า 24 พิกเซลถ้ามากกว่า 24 พิกเซลจะสรุปว่าเป็นเส้นแนวนอน และจะต้องมีเส้นแนวนอนทั้งหมดมากกว่า 24 เส้นจึงจะสรุปได้ว่าเป็นลายแนวนอน ขั้นตอนการทำงานดังแผนผังที่รูปที่ 4.11



รูปที่ 4.11 แผนผังการหาลวดลายแนวนอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6.4 ขบวนการหาลวดลายจุด

หลักการในการเชคว่าเป็นลายจุดหรือไม่สามารถใช้เทคนิคที่เรียกว่า Flood Fill เพื่อทำการเชคว่าพิกเซลข้างเคียงกับพิกเซลที่เชคมีค่าสีเดียวกันหรือไม่โดยที่ถ้ามีค่าสีเดียวกันและมีจำนวนมากกว่า 3 พิกเซลให้เก็บค่าที่ได้ไว้ว่าเป็นค่าสีอะไรที่เหมือนกันและเหมือนกันเป็นกลุ่มจำนวนพิกเซลเท่าใด หลังจากนั้นจะทำการเชคพิกเซลถัดไปจนกระทั่งเชคหมดทั้งภาพ หลังจากนั้นเราจะนำค่าที่เก็บไว้มาวิเคราะห์ว่าเป็นลายจุดหรือไม่ ถ้าเป็นลายจุดเป็นลายจุดในลักษณะไหนเป็นจุดเดียวหรือหลายจุด

4.7 การแสดงผลในรูปของเสียง

ในการแสดงผลจะแสดงผลลัพธ์ที่ประมวณผลได้ของ application คือสีเสื้อและลวดลายของเสื้อโดยที่แสดงผลออกมาเป็นเสียงโดยที่เราจำทำการบันทึกไฟล์เสียงของ output ที่เป็นไปได้ทั้งหมดก่อนหลังจากที่ application ส่วนที่ทำงานเกี่ยวกับ Image Processing ประมวณผลเสร็จแล้ว เราค่อยทำการแสดงไฟล์เสียงที่เหมาะสมกับ output ที่ได้

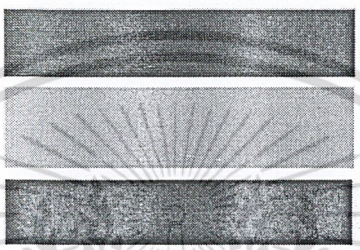
บทที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

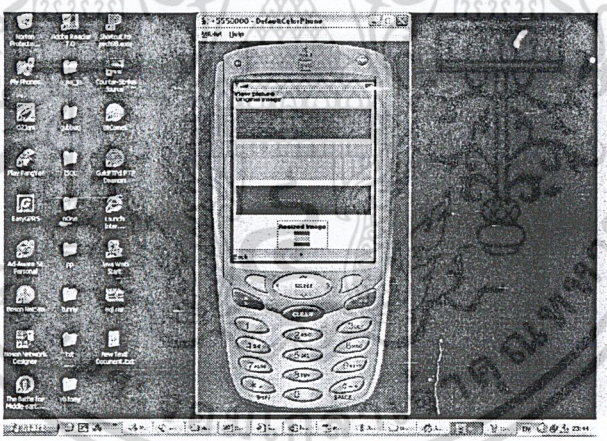
บทที่ 5

การทดลองและผลการทดลอง

ในการทดสอบการใช้งานในช่วงแรกของการทดสอบ application เราใช้ภ พสีที่สร้างโดยโปรแกรม Paint บน WINDOW XP มาทดสอบว่าเขียน Algorithm ถูกต้องหรือไม่ โดยภาพตัวอย่างที่นำมาทดสอบดังรูปที่ และเมื่อประมวณผลด้วย application ที่เขียนขึ้นจะเห็นผลขอ การทำงานบน Emulator โทรศัพท์ดังรูปที่ 5.1



รูปที่ 5.1 ภาพตัวอย่างที่นำมาทดสอบ

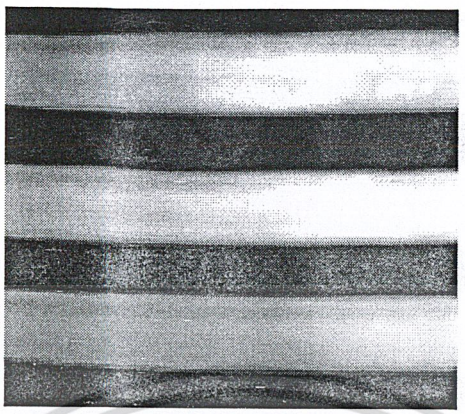


รูปที่5.2 ผลของการทำงานบนอิมูเลเตอร์

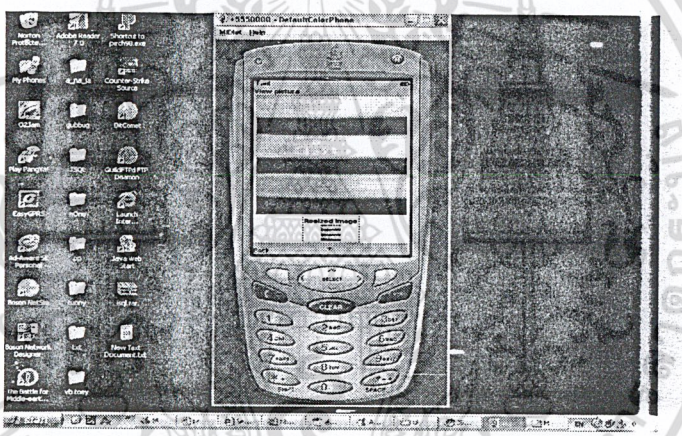
ซึ่งผลการทำงานของโปรแกรมที่ใช้ภาพที่สร้างขึ้นเองจากที่กล่าวข้างต้นส เมารถทำงานได้อย่างถูกต้องโดยที่สามารถบอกสีทั้งหมดบนภาพและลายซึ่งในตัวอย่างนี้เป็นลายขวาง เ าะเมื่อนำภาพอื่นๆ มาทดสอบก็พบว่าได้ค่าที่ถูกต้องดังนั้นจึงอนุมานได้ว่าโปรแกรมที่เขียนขึ้นนั้นมีอัลก ริทึมถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนต่อไปในการทดสอบจะใช้ภาพจริงมาทดสอบ โดยใช้ภาพที่ถ่ายจากกล้องที่ติดตั้งบนมือถือซึ่งหนึ่งในภาพที่นำมาทดสอบก็คือรูปที่แสดงดังภาพที่ และผลจากการรันโดยโปรแกรมแสดงได้ดังรูปที่ 5.3

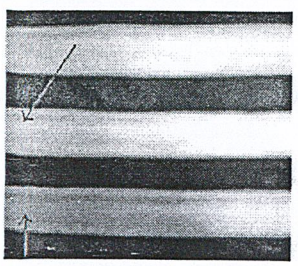


รูปที่ 5.3 แสดงภาพจริงที่นำมาทดสอบ



รูปที่5.4 ผลของการทำงานบนอิมูเลเตอร์

ซึ่งเมื่อมีการทำงานของโปรแกรมจะพบว่าการแสดงค่าของสีมีการแสดงผลของสีที่ไม่น่าจะเกิดขึ้นนั่นคือสีม่วง จากการวิเคราะห์ภาพที่ถ่ายมาจะพบว่ามีบางส่วนของภาพที่แสงมาตกกระทบได้น้อยกว่าส่วนอื่นๆ ทำให้ค่าที่ R,G,B ของพิกเซลบริเวณนั้นผิดเพี้ยนไปที่ควรเป็นดังแสดงได้ดังรูปที่ 5.5



รูปที่ 5.5 ภาพถ่ายที่มีปัญหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งหากเกิดปัญหานี้จะทำให้การทำงานของโปรแกรมมีปัญหาอย่างมากเนื่องจากการที่ข้อมูลที่ผิดเพี้ยนไปจะทำให้การคำนวณค่าสีและการหาลายของเสื้อที่ถ่ายมาไม่ถูกต้อง

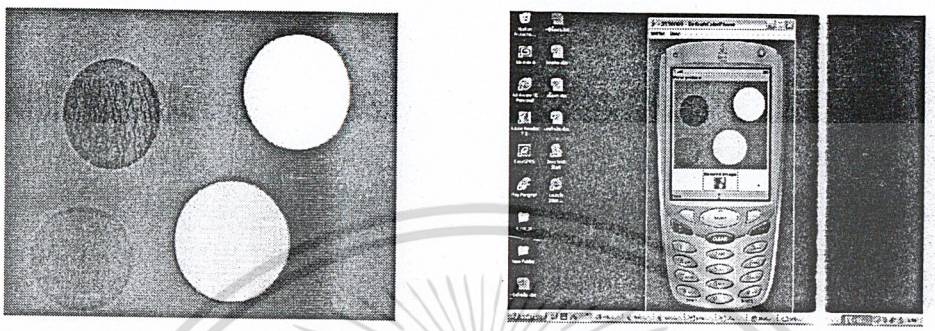
ดังนั้นการแก้ไขปัญหาที่เกิดขึ้นจึงมีดังนี้คือ

1. นำกล้องที่มีแฟลชถ่ายภาพมาทดสอบดูเพื่อจะแก้ปัญหาเรื่องแสงที่ตกกระทบจะพบว่าสามารถแก้ปัญหาได้ดีขึ้นมากแต่ก็มีบางส่วนของภาพยังมีปัญหาอยู่

2.การพยายามปรับแต่งอัลกอริทึมให้ดีขึ้นอย่างเช่นในการจะบอกว่าภาพที่ถ่ายมานั้นมีสีอะไรบ้างจากปกติเราควรจะบอกผลลัพธ์ของสีที่ตรวจพบทุกๆ พิกเซลแต่จากปัญหาเรื่องแสงที่เกิดขึ้นบางพิกเซลก็มีสีที่ผิดเพี้ยนไปถ้าแสดงผลลัพธ์ออกมาทั้งหมดอาจทำให้ผู้ใช้โปรแกรมเข้าใจผิดได้ ดังนั้นเราสามารถปรับหลักการในการเขียนฟังก์ชันเป็นในการบอกว่ามีสีอะไรบ้างนั้นเราจะนำมาเทียบกับจำนวนพิกเซลทั้งหมดก่อน โดยสามารถกำหนดเป็นเปอร์เซนต์ว่าถ้ามีมากกว่ากี่เปอร์เซนต์จึงจะรายงายผลลัพธ์ออกมาทำให้สามารถแก้ปัญหาดังที่กล่าวมาได้ และการในแก้ปัญหาโดยการปรับปรุงเรื่องอัลกอริทึมอีกวิธีหนึ่งก็คือในการหาลายแนวตั้งปกติเราควรจะพิจารณาว่าค่าของพิกเซลในคอลัมภ์เดียวกันควรจะมีค่าเดียวกันทั้งหมดและทุกๆ คอลัมภ์จึงจะเป็นลายตั้ง แต่เนื่องจากปัญหาเรื่องของแสงที่มาตกกระทบไม่เท่ากันทุกส่วนดังที่กล่าวมาข้างต้นเราจึงปรับปรุงอัลกอริทึมเป็นในการหาลายแนวตั้งคือ ทำการเชคค่าพิกเซลในคอลัมภ์เดียวกันว่ามีค่าเดียวกันหรือไหมโดยที่ต้องมีค่าเดียวกันมากกว่า 24 พิกเซลถ้ามากกว่า 24 พิกเซลจะสรุปว่าเป็นเส้นแนวตั้ง และจะต้องมีเส้นตั้งทั้งหมดมากกว่า 24 เส้นจึงจะสรุปได้ว่าเป็นลายแนวตั้ง ทั้งนี้เนื่องจากการที่เรากำหนดให้พิจารณาเพียง 24 พิกเซลจากที่ควรจะเป็น 30 พิกเซลเพราะเราจะพิจารณาว่าหากมีบางพิกเซลข้อมูลยังผิดเพราะแสงที่มาตกกระทบน้อยนั้นก็ไม่ทำให้ผลลัพธ์นั้นผิดพลาด ในกรณีแนวนอนนั้นก็ใช้หลักการเดียวกับแนวตั้งดังที่กล่าวมา ส่วนในกรณีของลายจุดนั้นเราสามารถลดข้อผิดพลาดของสีที่เกิดจากปัญหาแสงดังที่กล่าวมา โดยในการเชคว่าลายนั้นเป็นกลุ่มจุดหรือไม่เราจะพิจารณาเริ่มต้นจากจำนวน 3 พิกเซลโดยที่ถ้ามีกลุ่มพิกเซลที่มีสีเดียวกันน้อยกว่า 3 เราจะไม่พิจารณาว่าเป็นจุดแต่จะพิจารณาว่าเป็นส่วนที่เกิดข้อผิดพลาดจากแสงแทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนต่อไปคือการทดสอบด้วยลายจุด ซึ่งเราจะใช้ภาพที่สร้างโดยโปรแกรม Paint บน WINDOW XP มาทดสอบก่อนจากรูปที่ 5.6 ผลลัพธ์ที่ได้ก็คือโปรแกรมจะแสดงผลว่าเป็นภาพหลายจุดมีสีดำ ขาว น้ำเงิน เหลืองและแดง ส่วนขั้นตอนต่อไปคือนำภาพถ่ายจริงของเสื้อที่มีลักษณะลายจุดดังรูปที่ 5.7 ซึ่งผลลัพธ์ที่ได้ออกมาก็คือโปรแกรมจะแสดงผลว่าเป็นภาพหลายจุดมีสีขาว เขียวและแดง ซึ่งเป็นผลลัพธ์ที่ถูกต้อง

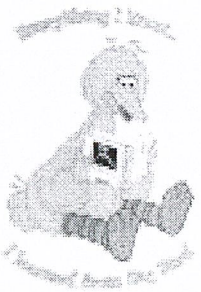


รูปที่ 5.6 การทดสอบลายจุดจากภาพที่สร้างขึ้นโดยโปรแกรม

รูปที่ 5.7 การทดสอบลายจุดจากภาพถ่ายจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปภาพต่อไปที่นำมาทดสอบเป็นรูปภาพที่ค่อนข้างซับซ้อนดังรูปที่ 5.8 ซึ่งผลลัพธ์ที่ได้ก็คือมีลักษณะหลายจุด มีสีเหลืองสีส้ม และสีขาวซึ่งเป็นผลลัพธ์ที่ค่อนข้างน่าพอใจ



รูปที่ 5.8 ภาพที่นำมาทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
บทวิจารณ์และสรุป

6.1 สรุปผลการทดลอง

เมื่อทดสอบการทำงานจากหลายๆ รูปภาพแล้วพบว่าการทำงานเป็นไปอย่ งน่าพอใจ โปรแกรมสามารถทำงานตอบสนองความต้องการพื้นฐานของผู้สร้างได้ แต่ยังมีปัญหาอยู่บ้าง กี่ยวกับเรื่องแสงที่ตกกระทบวัตถุไม่เท่ากันทำให้ค่า R G B ของบางพิกเซลผิดเพี้ยนทำให้การประมวณผลอาจจะผิดพลาดอยู่บ้าง และยังมีปัญหาเกี่ยวกับเวลาที่ใช้ในการประมวณผลอยู่พอสมควร เนื่องจากโปรแกรมที่สร้างขึ้นค่อนข้างใหญ่และหน่วยความจำบนโทรศัพท์มือถือนั้นยังมีไม่มาก ส่วนลวดลายที่สามารถตรวจจับได้นั้นยังมีไม่กี่ลายทั้งนี้เป็นผลอันเนื่องมาจากข้อจำกัดทางด้านหน่วยความจำบนโทรศัพท์มือถือนั่นเอง

6.2 ปัญหาและอุปสรรค

ปัญหาของการทำโครงงานนี้มีดังนี้

1) ภาพถ่ายที่นำมาวิเคราะห์ส่วนใหญ่ยังมีปัญหาเรื่องของแสงที่มาตกกระทบไม่เท่ากันทำให้ค่าของพิกเซลที่ได้มามีสีผิดเพี้ยนไปจากความเป็นจริง
2) ความเร็วในการประมวณผลค่อนข้างช้าเป็นปัญหาอันเนื่องมาจากหน่วยความจำของอุปกรณ์โทรศัพท์มือถือมีขนาดเล็ก

6.3 แนวทางในการพัฒนาต่อ

6.2.1 ศึกษาการเรียกใช้ข้อมูลบนหน่วยความจำบนโทรศัพท์มือถือเพื่อทำให้โปรแกรมนี้สามารถใช้งานได้จริงบนโทรศัพท์มือถือซึ่งโปรแกรมจำเป็นต้องเรียกใช้ข้อมูลภาพที่เก็บไว้บนหน่วยความจำ

6.2.2 ลดความผิดพลาดที่เกิดจากแสงดังที่กล่าวมา ซึ่งอาจจะแก้ปัญหาทางด้านฮาร์ดแวร์คือกล้องหรือด้านซอฟต์แวร์คือการปรับปรุงอัลกอริทึม

6.2.3 พัฒนาการประมวลผลให้มีประสิทธิภาพมากขึ้น โดยให้สามารถคำนวณลายของเสื้อผ้าได้มากขึ้นโดยใช้เวลาที่น้อยที่สุด

6.2.4 สร้าง user interface ที่เหมาะสมกับคนตาบอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ภาษาไทย

[1] จันทรมาส สาณะเสน พันธุ์คำ , “สู่ก้าวใหม่แห่งการพัฒนาโปรแกรมบนโทรศัพ ์เคลื่อนที่ด้วย J2ME” , สำนักพิมพ์ซายน์ซอฟต์ คอร์ปอเรชั่น

ภาษาอังกฤษ

[1] Vartan Piroumian , “ Wireless J2ME Platform Programming” , A Prentice Hal Title

[2] John w. Muchow , “Core J2ME Technology & MIDP” , Prentice Hall PTR , U pper Saddle River , NJ 07458

[3] Paul Tremblett , “Instant Wireless Java with J2ME” , McGraw-Hill/Osborne

[4] Rafael C. Gonzalez , Richard E. Woods, ”Digital Image Processing “A Prentic e Hall Title

เว็บไซต์

[1] http://www.samspublishing.com

[2] http://www.webyu.com

[3] http://java.sun.com/aboutJava/communityprocess/

[4] http://java.sun.com/j2me/

[5] http://java.sun.com/product/cldc/

[6] http://java.sun.com/aboutJava/communityprocess/final/jsr030/index.html

[7] http://java.sun.com/aboutJava/communityprocess/final/jsr037/index.html

[8] http://java.sun.com/products/javaphone/

[9] http://www.sun.com/forte/ffj/

[10] http://sun.java.com/products/j2mewtoolkit/

[11] http://java.sun.com/product/midp/

[12] http://archives.java.com/archives/kvm-interest.html

[13] http://www.wirelessdevnet.com

[14] http://www.kvmworld.com

[15] www.phptr.com

[16] www.reindeergraphics.com/

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้