

การหาจุดเปลี่ยนของวิดีโอโดยใช้ภาพกราฟแจกแจงความถี่

VIDEO SCENE CHANGE DETECTION USING LUMINANCE HISTOGRAM

วีระ คมปริยรัตน์
WEERA KOMPRAYARAT



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี

พ.ศ. 2554

การหาจุดเปลี่ยนของวิดีโอโดยใช้ภาพกราฟแจกแจงความถี่

วีระ คมปริยรัตน์



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี

พ.ศ. 2554

VIDEO SCENE CHANGE DETECTION USING LUMINANCE HISTOGRAM

WEERA KOMPREYARAT



A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR
THE DEGREE OF MASTER OF ENGINEERING
IN ELECTRICAL ENGINEERING DEPARTMENT OF ELECTRICAL ENGINEERING
FACULTY OF ENGINEERING

RAJAMANGALA UNIVERSITY OF TECHNOLOGY THANYABURI

2011

วิทยานิพนธ์ฉบับนี้ เป็นงานวิจัยที่เกิดจากการค้นคว้าและวิจัยที่ข้าพเจ้าร่วมกับอาจารย์ผู้
ควบคุมวิทยานิพนธ์ และผู้ที่มีความชำนาญ เชี่ยวชาญ รวมถึงคำแนะนำจากผู้ที่เกี่ยวข้อง ดังนั้น
ข้อความต่างๆ ในวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอรับรองว่าไม่มีการคัดลอกหรือนำงานวิจัยของผู้อื่นมา
นำเสนอในชื่อของข้าพเจ้า

งานวิจัยในวิทยานิพนธ์ฉบับนี้ถือเป็นลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี



นายวิระ คมปรียารัตน์



ใบรับรองวิทยานิพนธ์

คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี

หัวข้อวิทยานิพนธ์

การหาจุดเปลี่ยนของวิดีโอโดยใช้ภาพกราฟแฉกแสงความถี่
VIDEO SCENE CHANGE DETECTION USING LUMINANCE
HISTOGRAM

ชื่อนักศึกษา

นายวีระ คมปริยารัตน์

รหัสประจำตัว

114970402019-1

ปริญญา

วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา

วิศวกรรมไฟฟ้า

อาจารย์ที่ปรึกษาวิทยานิพนธ์

ดร. จักรี ศรีนนท์ฉัตร

วัน เดือน ปี ที่สอบ

7 พฤษภาคม 2554

สถานที่สอบ

ห้องประชุมรวงข้าว ชั้น 7 ณ อาคารเฉลิมพระเกียรติ 80 พรรษา
5 ธันวาคม 2550 คณะวิศวกรรมศาสตร์

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ดร. วินัย วิชัยพาณิชย์)

..... กรรมการ
(รองศาสตราจารย์ ดร.สมเกียรติ อุดมหารธรรษากุล)

..... กรรมการ
(ดร. อำนวย เรืองวารี)

..... กรรมการ
(ดร. จักรี ศรีนนท์ฉัตร)

.....
(ผู้ช่วยศาสตราจารย์ ดร. สมหมาย ผิวสอาด)
คณบดีคณะวิศวกรรมศาสตร์

หัวข้อวิทยานิพนธ์	การหาจุดเปลี่ยนของวิดีโอโดยใช้ภาพกราฟแจกแจงความถี่
นักศึกษา	นายวีระ คมปริยารัตน์
รหัสประจำตัว	114970402019-1
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมไฟฟ้า
	แขนงวิศวกรรมอิเล็กทรอนิกส์และโทรคมนาคม
ปีการศึกษา	2553
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ดร.จักรี ศรีนนท์ฉัตร

บทคัดย่อ

งานวิจัยนี้ได้ทำการพัฒนาอัลกอริทึมสำหรับหาจุดเปลี่ยนในวิดีโอ แบบไม่มีการบีบอัดของ ข้อมูลวิดีโอ โดยใช้ภาพกราฟแจกแจงความถี่ (Image Histogram) ของวิดีโอ

อัลกอริทึมที่นำเสนอจะแบ่งภาพออกเป็นส่วนๆ (Image Segmentation) การแบ่งภาพแบบ การทับซ้อนกันของแต่ละส่วน (Cascade Segmentation) และการแบ่งภาพแบบไม่มีการทับซ้อนกัน ของแต่ละส่วน (Non-cascade Segmentation) ก่อนที่จะทำการหาภาพกราฟแจกแจงความถี่ของภาพ แต่ละส่วน และนำไปหาจุดเปลี่ยนโดยใช้ทฤษฎีการหาจุดเปลี่ยนโดยใช้ ค่าสัมบูรณ์ของความแตกต่าง ของภาพกราฟแจกแจงความถี่ (Absolute Value of Histogram) ใคกำลังสองของภาพกราฟแจกแจง ความถี่ (Chi-Square Value of Histogram) สำหรับการหาจุดเปลี่ยนแบบที่มีการเปลี่ยนแปลงแบบ ทันทีทันใด (Rapidly Scene Change) และ อัลกอริทึมการนับจำนวนของภาพกราฟแจกแจงความถี่ ใน กรณีที่เป็นการหาจุดเปลี่ยนแบบที่มีการเปลี่ยนแปลงแบบช้าๆ (Gradually Scene Change) แบบ Fade และ Dissolve และนำแต่ละส่วนมาเปรียบเทียบเพื่อตรวจสอบว่าเป็นจุดเปลี่ยนหรือไม่ การเปรียบเทียบและ ทดสอบประสิทธิภาพจะกระทำในรูปแบบเชิงคุณภาพที่เพิ่มขึ้น โดยใช้ทฤษฎีทางสถิติ Combine Recall-Precision เปรียบเทียบกับเวลาที่ใช้ในกระบวนการที่เพิ่มขึ้น โดยใช้ข้อมูลจากการทดสอบบน โปรแกรม Matlab

ผลการทดสอบพบว่าอัลกอริทึมสามารถทำงานได้ดี ในการหาจุดเปลี่ยนแบบทันทีทันใดและยัง สามารถใช้สำหรับการหาจุดเปลี่ยนแบบที่มีการเปลี่ยนแปลงแบบช้าๆ ได้อย่างมีประสิทธิภาพ นอกจากนี้ยัง ได้นำเสนอวิธีการแก้ปัญหาในกรณีที่มีสัญญาณรบกวน และมีภาพโลโก้ติดอยู่ตลอดทั้งวิดีโอในการ หาจุดเปลี่ยนแบบ Fade อีกด้วย

คำสำคัญ : จุดเปลี่ยนของวิดีโอ ภาพกราฟแจกแจงความถี่

Thesis Title : VIDEO SCENE CHANGE DETECTION USING LUMINANCE
HISTOGRAM

Student Name : Mr. Weera Kompreyarat

Student ID : 114970402019-1

Degree Award : Master of Engineering

Study Program : Electrical Engineering
(Electronics and Telecommunication Engineering)

Academic Year : 2010

Thesis Advisor : Dr. Jakkree Srinonchat

ABSTRACT

The Research will be focused on Video Scene Change Detection Algorithm in Un-compression Domain Using Image Histogram Information.

The proposed algorithm uses non-cascade image segmentation and cascade image segmentation as pre-processing of the algorithm. The standard scene change detection method (Absolute value of histogram and Chi-Square value of Histogram) is used to detect Rapid Scene change. Counting histogram Bin is used to detect gradual scene change (Fade and Dissolve). The Result is compared with the general histogram based algorithm using Combine Recall-Precision method and Time consuming of the algorithm. The simulation will be done using Matlab.

The proposed algorithm show very good result in Rapid Scene change detection and gradually scene change. The algorithm also overcomes the noise and logo with in the video in fade scene change detection.

Keywords : Scene Change Detection, Image Histogram

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงไปได้ ด้วยความช่วยเหลือเป็นอย่างดีของ ดร.จักรี ศรีนนท์ฉัตร ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ และสมาชิกในห้องปฏิบัติการและวิจัยทางด้านการประมวลผลสัญญาณทุกท่านสำหรับคำแนะนำและข้อคิดเห็นต่างๆ ทั้งในส่วนของการทำงานวิจัยและในส่วนของการทำรูปเล่มวิทยานิพนธ์ ขอขอบพระคุณคณาจารย์ทุกท่านที่ได้อบรมสั่งสอนและประสิทธิ์ประสาทความรู้วิชาการต่างๆ ให้ตลอดระยะเวลาที่ผ่านมา

ขอขอบคุณมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรีที่ให้การสนับสนุนทางการเงินท้ายที่สุดนี้ขอขอบคุณบิดามารดาและญาติพี่น้องทุกคนสำหรับกำลังที่มีให้เสมอมา

วีระ คมปริยรัตน์

7 พฤษภาคม 2554



สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ฉ
สารบัญรูป	ช
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 ความมุ่งหมายและวัตถุประสงค์	2
1.3 ขอบเขตของการศึกษา	2
1.4 ขั้นตอนการศึกษา	3
1.5 ข้อจำกัดของการศึกษา	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	4
2.1 ภาพจิตตอล	4
2.2 ระบบสี	4
2.3 ภาพโทนสีเทา	6
2.4 ภาพกราฟแจกแจงความถี่	7
2.5 จุดเปลี่ยนของวิดีโอ	8
2.6 การหาจุดเปลี่ยนของวิดีโอ	14
2.7 การใช้ทฤษฎี Recall-Precision ในการทดสอบประสิทธิภาพอัลกอริทึม	27
บทที่ 3 วิธีการดำเนินการวิจัย	28
3.1 ข้อกำหนดงานวิจัย	28
3.2 การหาจุดเปลี่ยนแบบทันทีทันใด	28
3.3 การหาจุดเปลี่ยนแบบต่อเนื่อง	31
บทที่ 4 ผลการวิจัย	41
4.1 การหาจุดเปลี่ยนแบบทันทีทันใด	41
4.2 การหาจุดเปลี่ยนแบบ Fade	43
4.3 การหาจุดเปลี่ยนแบบ Dissolve	47
4.4 การหาจุดเปลี่ยนแบบ Wipe	50

สารบัญ (ต่อ)

	หน้า
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	52
5.1 สรุปผลที่ได้จากการวิจัย	52
5.2 ข้อเสนอแนะและแนวทางในการพัฒนา	53
เอกสารอ้างอิง	54
ภาคผนวก ผลงานตีพิมพ์เผยแพร่	56
ประวัติผู้เขียน	80



สารบัญตาราง

ตารางที่	หน้า
4.1 จำนวนภาพและจุดเปลี่ยนแบบ Rapid ที่ใช้ในการทดสอบ	40
4.2 การเปรียบเทียบผลการทดสอบประสิทธิภาพอัลกอริทึมของการเปลี่ยนภาพแบบ Rapid Scene change Detection	41
4.3 การเปรียบเทียบผลการทดสอบประสิทธิภาพอัลกอริทึมในด้านความเร็วเฉลี่ยของการเปลี่ยนภาพแบบ Rapid	42
4.4 จำนวนภาพชุดที่ 1 และจุดเปลี่ยนแบบ Fade ที่ใช้ในการทดสอบ	43
4.5 จำนวนภาพชุดที่ 2 และจุดเปลี่ยนแบบ Fade ที่ใช้ในการทดสอบ	44
4.6 ผลการทดสอบประสิทธิภาพอัลกอริทึมของการเปลี่ยนภาพแบบ Fade	44
4.7 การเปรียบเทียบผลการทดสอบประสิทธิภาพอัลกอริทึมในด้านความเร็วเฉลี่ยของการเปลี่ยนภาพแบบ Fade	45
4.8 ผลการทำงานของอัลกอริทึมขณะที่มี Noise หรือ Logo บนภาพ	46
4.9 จำนวนจุดเปลี่ยนแบบ Dissolve ที่ใช้ในการทดสอบ	47
4.10 ผลการทดสอบประสิทธิภาพอัลกอริทึมของการเปลี่ยนภาพแบบ Dissolve	48
4.11 การเปรียบเทียบผลการทดสอบประสิทธิภาพอัลกอริทึมในด้านความเร็วเฉลี่ยของการเปลี่ยนภาพแบบ Dissolve	49
4.12 จำนวนจุดเปลี่ยนแบบ Wipe ที่อัลกอริทึมสามารถจับได้	50
4.13 ผลการทดสอบประสิทธิภาพอัลกอริทึมของการเปลี่ยนภาพแบบ Wipe	50
4.14 การเปรียบเทียบผลการทดสอบประสิทธิภาพอัลกอริทึมในด้านความเร็วเฉลี่ยของการเปลี่ยนภาพแบบ Wipe	51

สารบัญรูป

รูปที่	หน้า
1.1 ขอบเขตของ MPEG-7 [5]	1
1.2 ระบบการทำงานของมาตรฐาน MPEG-7 [5]	2
2.1 แบบสี RGB ในรูปแบบพิกัด 3 มิติ	5
2.2 ค่าสีใน Gray Scale	6
2.3 แบบสี RGB ในรูปแบบพิกัด 3 มิติ แสดงจุด pixel ที่ต้องการหา	7
2.4 การเปลี่ยนระบบสี RGB เป็น Gray Scale	7
2.5 ตัวอย่าง Histogram ของภาพ	8
2.6 การเปลี่ยนภาพแบบ Rapid Scene Change	9
2.7 การเปลี่ยนภาพแบบ Fade Scene Change ลักษณะ Fade in	10
2.8 อธิบายการทำงานแบบ Fade in	10
2.9 การเปลี่ยนภาพแบบ Fade Scene Change ลักษณะ Fade out	11
2.10 อธิบายการทำงานแบบ Fade out	12
2.11 การเปลี่ยนภาพแบบ Dissolve Scene Change	12
2.12 การเปลี่ยนภาพแบบ Wipe Scene Change	14
2.13 ตัวอย่าง Template matching based Absolute Value	15
2.14 การหาความแตกต่างของภาพ โดยแสดงเป็นกราฟ	16
2.15 ลำดับของภาพและ Histogram ของภาพในแต่ละเฟรม	18
2.16 การหาผลลัพธ์ตามทฤษฎี Chi-square value of histogram	19
2.17 การหาผลลัพธ์ตามทฤษฎี Absolute value of histogram	20
2.18 ตัวอย่างค่าทางสถิติของภาพ	21
2.19 ตัวอย่าง Histogram แบบ Fade-in	22
2.20 ตัวอย่าง Histogram แบบ Fade-in	23
2.21 Histogram ของการเปลี่ยนภาพแบบ Dissolve	24
2.22 กราฟเมื่อเกิด Dissolve ในวิดีโอ	25
2.23 Histogram ของการเปลี่ยนภาพแบบ Wipe	26
3.1 การแบ่งส่วนของภาพในการหา จุดเปลี่ยนของภาพแบบ Rapid	29
3.2 การแบ่งส่วนของภาพแบบซ้อนทับในการหา จุดเปลี่ยนของภาพแบบ Rapid	29
3.3 Flowchart การทำงานของการหาจุดเปลี่ยนของภาพแบบ Rapid	30
3.4 ตัวอย่างภาพที่ได้เมื่อสิ้นสุด Fade-out จาก VCD	31

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.5 ตัวอย่างภาพที่ได้เมื่อสิ้นสุด Fade-out เมื่อมี Logo บริษัทสีเขียวพื้นสีดำ	31
3.6 ค่าของ F(i) ขณะที่เกิด Fade-in และ Fade-out	32
3.7 Flow Chart การทำงานของ Fade Scene Change Detection	33
3.8 ค่าในอุดมคติขณะที่เกิด Dissolve Scene Change	34
3.9 Flow Chart การทำงานของ Dissolve Scene Change Detection	35
3.10 การหาจุดเปลี่ยนของภาพแบบ Wipe	36
3.11 กราฟเมื่อเกิดการเปลี่ยนภาพแบบ Wipe ตามลำดับ	38
3.12 Flowchart การทำงานของการหาจุดเปลี่ยนของภาพแบบ Wipe	39



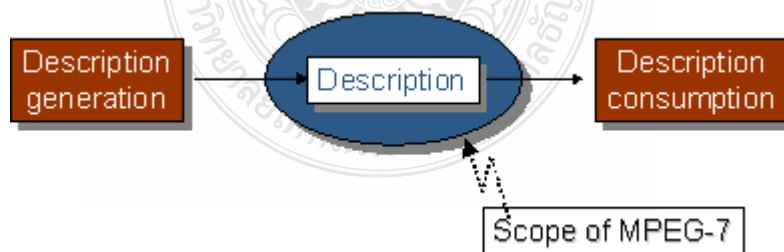
บทที่ 1

บทนำ

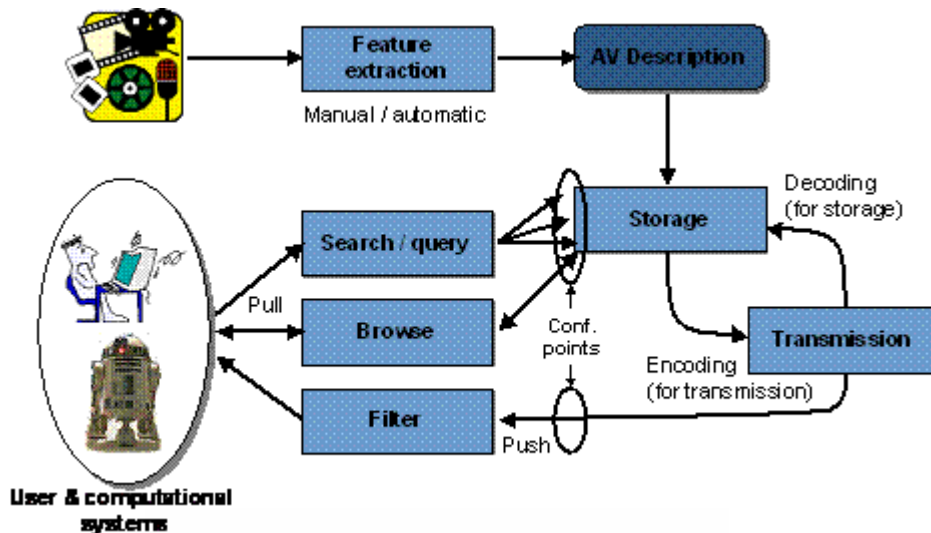
1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันความก้าวหน้าทางด้านเทคโนโลยีคอมพิวเตอร์และการสื่อสารได้พัฒนาไปอย่างรวดเร็ว ไม่ว่าจะเป็นทางด้านประสิทธิภาพ และความเร็วในการใช้งาน อีกทั้งการสื่อสารในปัจจุบันเป็นสิ่งจำเป็น ประกอบกับจำนวนผู้ใช้งานข้อมูลได้เพิ่มขึ้นอย่างรวดเร็ว จากสถิติของ Internet World Stats พบว่าปัจจุบันมีจำนวนผู้ใช้งานเครือข่าย ณ.วันที่ 30 กันยายน 2550 จำนวน 1,244,449,601 คน เพิ่มขึ้น 244.7% เมื่อเทียบกับปี 2543 ช่วงเวลาเดียวกัน ทำให้มีข้อมูลข่าวสารทั้งที่เป็นข้อความเอกสารภาพและเสียง (Audio-Visual Information) ได้มีจำนวนเพิ่มมากขึ้นตามไปด้วย

จากการประชุมเพื่อสร้างมาตรฐานในการค้นหาข้อมูล Audio-Visual โดยสาระหลักของการประชุมได้ข้อสรุปดังบทความ [1,2,3,4] เมื่อเดือนเมษายน 2540 ที่บิสโทรล และเดือน กรกฎาคม 2540 ขึ้นที่กรุงสต็อกโฮม การประชุมยังได้กำหนดขอบเขตของมาตรฐานไว้ที่ข้อมูลปลายทาง (Content Description) โดยเรียกชื่อมาตรฐานนี้ว่า MPEG-7 ดังรูปที่ 1.1 โดยไม่สนใจวิธีการที่ทำให้ได้ข้อมูลนั้นมา อีกทั้งยังกำหนดให้วิธีการที่ให้มาได้มาซึ่งข้อมูลต่างๆ เป็นแบบอัตโนมัติ กึ่งอัตโนมัติ หรือกระทำเองก็ได้ดังรูปที่ 1.2 ทำให้งานวิจัยในสาขานี้เปิดกว้างขึ้นเป็นอย่างมาก หลังจากนั้นได้มีการพัฒนาความต้องการเรื่อยมาจนถึง MPEG-7 (version 10) [5] จากการประชุมที่ Palma de Mallorca เมื่อเดือนตุลาคม 2547



รูปที่ 1.1 ขอบเขตของ MPEG-7 [5]



รูปที่ 1.2 ระบบการทำงานของมาตรฐาน MPEG-7 [5]

จากมาตรฐานที่กำหนดขึ้นมา ทำให้งานวิจัยหลักอยู่ที่การสร้าง Description ของข้อมูล (Feature Extraction) หนึ่งในข้อกำหนดของความต้องการใน MPEG-7 คือการค้นหาข้อมูลในวิดีโอ โดยการค้นหาวิดีโอ นั้นอาจจะค้นหาจากคุณสมบัติของวิดีโอเช่น สี การเคลื่อนไหวของวัตถุ (Objects Motion) โครงร่างของวัตถุ ฯลฯ แต่ในวิดีโอ นั้นประกอบขึ้นมาจากกลุ่มของฉากย่อยๆ (Video Scene) มาประกอบกัน ทำให้ยากต่อการทำความเข้าใจกับวิดีโอโดยเครื่องคอมพิวเตอร์ ดังนั้นการจัดกลุ่มของวิดีโอ (Video Abstraction) จึงเป็นความจำเป็นพื้นฐานในการทำให้การค้นหาข้อมูลต่างๆภายในวิดีโอ เป็นไปได้ง่ายขึ้น การที่จะให้ได้มาซึ่งกลุ่มของ Video Scene นั้นจำเป็นที่จะต้องหาจุดที่มีการเปลี่ยนแปลงจากฉากหนึ่งไปยังอีกฉากหนึ่ง ซึ่งกระบวนการนี้เรียกว่าการหาจุดเปลี่ยนของวิดีโอ (Scene Change Detection)

1.2 ความมุ่งหมายและวัตถุประสงค์

- 1.2.1 เพื่อศึกษาแนวทางการหาจุดเปลี่ยนของวิดีโอที่ไม่มีการบีบอัด
- 1.2.2 เพื่อพัฒนาระบบการหาจุดเปลี่ยนของวิดีโอที่ไม่มีการบีบอัดที่มีประสิทธิภาพยิ่งขึ้น
- 1.2.3 เพื่อพัฒนาแนวทางใหม่ในการหาจุดเปลี่ยนทั้งแบบทันทีทันใดและแบบที่มีการเปลี่ยนแปลงช้า ๆ

1.3 ขอบเขตของการศึกษา

- 1.3.1 วิเคราะห์และออกแบบอัลกอริทึม ในการหาจุดเปลี่ยนของวิดีโอที่ไม่มีการบีบอัดของข้อมูล โดยใช้ข้อมูลภาพกราฟแฉกแฉงความถี่โดยใช้การตัดแบ่งส่วนของภาพเป็นขนาดต่างๆ ไม่มีการทับซ้อนกันของส่วนที่จะแบ่ง

1.3.2 ใช้ Absolute value of histogram and Chi-Square Value of Histogram ในการพัฒนา อัลกอริทึมของการเปลี่ยนของวิดีโอแบบทันทีทันใด โดยใช้ข้อมูลจากข้อ 1.3.1 เปรียบเทียบ ประสิทธิภาพของผลที่ได้กับอัลกอริทึมที่ไม่มีการแบ่งส่วนของภาพ ด้านประสิทธิภาพ และ ความเร็ว

1.3.3 พัฒนาปรับปรุงอัลกอริทึมของการเปลี่ยนแบบช้าๆ เปรียบเทียบประสิทธิภาพของผลที่ได้ กับอัลกอริทึมที่ไม่มีการแบ่งส่วนของภาพ ด้านประสิทธิภาพ และ ความเร็ว

1.4 ขั้นตอนการศึกษา

1.4.1 ศึกษาทฤษฎีที่เกี่ยวข้องกับการหาจุดเปลี่ยนของวิดีโอ

1.4.2 ศึกษาการใช้งานโปรแกรม MATLAB

1.4.3 พัฒนาตัวอย่างข้อมูลที่ใช้ในการทดสอบ

1.4.4 พัฒนาโปรแกรมหาจุดเปลี่ยนของวิดีโอโดยใช้โปรแกรม MATLAB

1.4.5 วิเคราะห์และสรุปผลการศึกษา

1.5 ข้อยกเว้นของการศึกษา

1.5.1 ใช้สำหรับสัญญาณวิดีโอที่ไม่มีการบีบอัดของข้อมูล

1.5.2 พัฒนาแยกกันระหว่างการทำจุดเปลี่ยนแบบทันทีทันใดและแบบค่อยๆ เปลี่ยน

1.5.3 Effect ที่ใช้ทดสอบมี 2 แบบคือ Fade และ Dissolve

1.5.4 ข้อมูลที่ใช้ในการทดสอบจะใช้ Luminance Histogram ของสัญญาณภาพ

1.5.5 ทดสอบกับภาพที่มีคุณภาพของภาพไม่น้อยกว่า VCD

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงรายละเอียดของทฤษฎีและความรู้ต่างๆ ที่นำมาใช้ในวิทยานิพนธ์ โดยเริ่มตั้งแต่พื้นฐานการประมวลผลสัญญาณภาพ จุดเปลี่ยนของวิดีโอ การหาจุดเปลี่ยนของวิดีโอ การวัดคุณภาพของอัลกอริทึมในการหาจุดเปลี่ยนของวิดีโอ รวมถึง งานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์นี้

2.1 ภาพดิจิทัล

ภาพดิจิทัลนั้นจะประกอบด้วยส่วนประกอบความเข้มแสง (Luminance component) และส่วนประกอบความเข้มสี (Chrominance component) ซึ่งข้อมูลเหล่านี้จะอยู่ในรูปแบบของจำนวนจุดภาพต่อหนึ่งหน่วยความยาว โดยแต่ละจุดได้จากการสุ่มตัวอย่างและการควอนไทซ์ โดยทั่วไปโดยทั่วไปแต่ละภาพเมื่อผ่านการควอนไทซ์และจะถูกแบ่งออกเป็น 256 ระดับ หรือ 8 บิต

ในการประมวลผลภาพดิจิทัล เราจะต้องทราบว่าในการอ้างอิงถึงภาพดิจิทัลเรากำหนดให้อยู่ในรูปเมทริกซ์ ขนาด 2 มิติ โดยค่า x และ y เป็นพิกัด และกำหนดให้ ฟังก์ชัน $f(x,y)$ คือ ความเข้มของแสงในแต่ละจุด หรือ ในแต่ละพิกเซล (Pixel) ของภาพ โดยที่ ค่าต่างๆ ในภาพดิจิทัลจะเป็นค่าที่จำกัด และเป็นปริมาณที่ไม่ต่อเนื่องระบบสีของ Digital image

2.2 ระบบสี

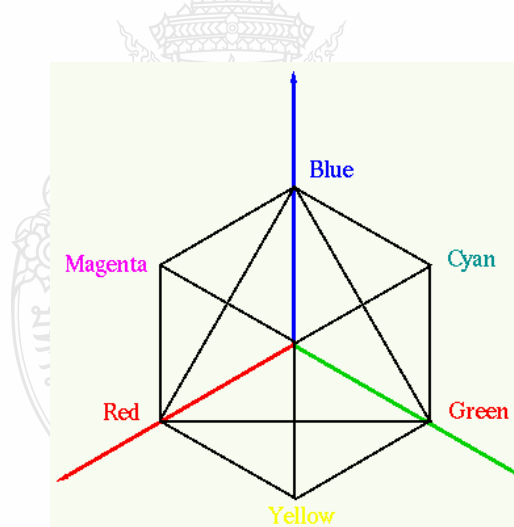
มาตรฐานของสีที่ใช้อยู่ในปัจจุบันมีอยู่หลายระบบด้วยกัน ทั้งนี้ขึ้นอยู่กับนำไปใช้ แต่โดยทั่วไปแล้ว ทุกมาตรฐานจะมีแนวคิดเดียวกันคือ การแทนจุดสีด้วยจุดที่อยู่ภายในสเปก 3 มิติ โดยจะมีแกนอ้างอิงสำหรับจุดสีนั้นในสเปกซึ่งแต่ละแกนจะมีความเป็นอิสระต่อกัน ตัวอย่างเช่นในระบบ RGB จะมีแกนสีคือ แคนสีแดง เขียว และน้ำเงินในระบบ HLS จะมีแกนเป็น ค่าสี(hue) ความสว่าง(lightness)และความบริสุทธิ์ของสี(saturation) ตัวอย่างระบบสีที่นิยมใช้กัน ได้แก่ ระบบ RGB HSV (Hue Saturation Value) ฯลฯ

2.2.1 ระบบสี RGB

ระบบสี RGB เป็นระบบสีของแสง ซึ่งเกิดจากการหักเหของแสงผ่านแท่งแก้วปริซึม จะเกิดแถบสีที่เรียกว่า สเปกตรัม (Spectrum) ซึ่งแยกสีตามที่สายตามองเห็นได้ 7 สี คือ แดง แสด เหลือง เขียว น้ำเงิน คราม ม่วง ซึ่งเป็นพลังงานอยู่ในรูปของรังสี ที่มีช่วงคลื่นที่สายตา สามารถมองเห็นได้ แสงสีม่วงมีความถี่คลื่นสูงที่สุดคลื่นแสงที่มีความถี่สูงกว่าแสงสีม่วงเรียกว่า อุลตราไวโอเล็ต (Ultra Violet) และคลื่นแสงสีแดง มีความถี่คลื่นต่ำที่สุด คลื่นแสงที่ต่ำกว่าแสงสีแดงเรียกว่า

อินฟราเรด (Infrared) คลื่นแสงที่มีความถี่สูงกว่าสีม่วง และต่ำ กว่าสีแดงนั้น สายตาของมนุษย์ไม่สามารถรับได้ และเมื่อศึกษาดูแล้วแสงสีทั้งหมดเกิดจาก แสงสี 3 สี คือ สีแดง (Red) สีน้ำเงิน (Blue) และสีเขียว (Green) ทั้งสามสีถือเป็นแม่สีของแสง เมื่อนำมาฉายรวมกันจะทำให้เกิดสีใหม่ อีก 3 สี คือ สีแดง มาเจนต้า สีฟ้าไซแอน และสีเหลือง และถ้าฉายแสงสีทั้งหมดรวมกันจะได้แสงสีขาว จากคุณสมบัติของแสงนี้เราได้นำมาใช้ประโยชน์ทั่วไป ในการฉายภาพยนตร์ การบันทึกภาพ วิดีโอ ภาพโทรทัศน์ การสร้างภาพเพื่อการนำเสนอทางจอคอมพิวเตอร์ และการจัดแสงสีในการแสดง เป็นต้น

ระบบ RGB เป็นระบบที่เกิดจากการผสมแสงสีหลัก 3 สี คือ แดง (Red) เขียว (Green) น้ำเงิน (Blue) โดยจะแยกกันอยู่คนละแชนแนล (Channel) สำหรับภาพขนาด 24 บิตต่อ พิกเซล (Pixel) ในแต่ละแชนแนลจะมีค่าความเข้มแสง 255 ระดับ คือ ตั้งแต่ 0 ถึง 255 หรือ เท่ากับ 8 บิต (2^8) ซึ่งค่าสีในแต่ละพิกเซลจะเกิดจากการรวมกันของข้อมูลทั้ง 3 แชนแนล โดยระบบ คาร์ทีเซียน โคออดิเนตในการระบุค่าสีแต่ละสี ซึ่งสามารถแสดงได้เป็นรูปลูกบาศก์ โดยมีค่า RGB เป็นค่ามุมทั้ง 3 และมีสีดำเป็น โคออดิเนตที่ (0,0,0) และมีสีขาวที่โคออดิเนต (255,255,255) ดังแสดงในรูป ส่วนค่าสีต่างๆ จะเป็นจุดภายในลูกบาศก์ ซึ่ง กำหนดโดยเวกเตอร์จากจุดกำเนิด ขึ้นไปยัง จุดนั้น ๆ



รูปที่ 2.1 แบบสี RGB ในรูปแบบพิกัด 3 มิติ

2.2.2 ระบบสี HSV

ระบบสี HSV (Hue Saturation Value) เป็นการพิจารณาสีโดยใช้ Hue Saturation และ Value ซึ่ง Hue คือ ค่าสีของสีหลัก(แดง เขียวและน้ำเงิน)ในทางปฏิบัติจะอยู่ระหว่าง 0 และ 255 ซึ่งถ้า Hue มีค่าเท่ากับ 0 จะแทนสีแดง และเมื่อ Hue มีค่าเพิ่มขึ้นเรื่อย ๆ สีก็จะเปลี่ยนแปลงไปตามสเปกตรัมของสีจนถึง 256 จึงจะกลับมาเป็นสีแดงอีกครั้ง ซึ่งสามารถแทนให้อยู่ในรูปขององศาได้ ดังนี้คือ สีแดง = 0 องศา สีเขียวเท่ากับ 120 องศา สีน้ำเงินเท่ากับ 240 องศา

2.2.3 ระบบสี YUV

ระบบสีแบบ YUV ใช้สำหรับโทรทัศน์แบบ PAL และ SECAM ซึ่งยังมีใช้อยู่ในหลาย ๆ ประเทศ โดย Y คือ ค่าความสว่างของภาพ ส่วนสัญญาณ U และ V เป็นสัญญาณที่เก็บค่าสีของภาพ ต่อมาได้มีระบบ YIQ มาใช้แทน เนื่องจากพบว่าสัญญาณ I และ Q สามารถลด Bandwidth ได้มากกว่าสัญญาณ U และ V ในขณะที่ได้ภาพที่มีคุณภาพเท่ากัน

2.2.4 ระบบสี YIQ

เป็นระบบที่ใช้ใน TV Broadcasting สำหรับ NTSC ประโยชน์หลักก็เพื่อให้ใช้งานได้กับโทรทัศน์แบบขาว-ดำ โดยที่ y คือความสว่างของภาพ ส่วน I และ Q จะเป็นสัญญาณที่เข้ารหัสสีของภาพไว้ ดังนั้นสำหรับโทรทัศน์ขาว-ดำ นั้นสามารถใช้ค่า Y ค่าเดียวก็สามารถได้ภาพที่สมบูรณ์

2.3 ภาพโทนสีเทา (Gray Level Image)

เป็นอัตราส่วนของโทนสีเทา ซึ่งมีการไล่ระดับความอ่อนแก่ ที่อยู่ระหว่างสีขาวกับสีดำ Halftone Image การสร้างภาพให้มีระดับสีต่างๆ อย่างต่อเนื่อง ด้วยการใช้จุดสีที่มีขนาดต่างกัน หรือมีความหนาแน่นของจุดต่างกัน

ค่าในแต่ละ pixel ของ gray image คือค่าความเข้มของแสง ณ แต่ละตำแหน่งของ pixel ซึ่งจะอยู่ในรูปของ gray scale ดังรูปที่ 2.2 ค่าที่เป็นไปได้ของ gray scale จะขึ้นอยู่กับจำนวน bit ที่ใช้ ตัวอย่างเช่น 8-bit monochrome จะมี gray scale ทั้งหมด 256 ระดับ ดังรูป

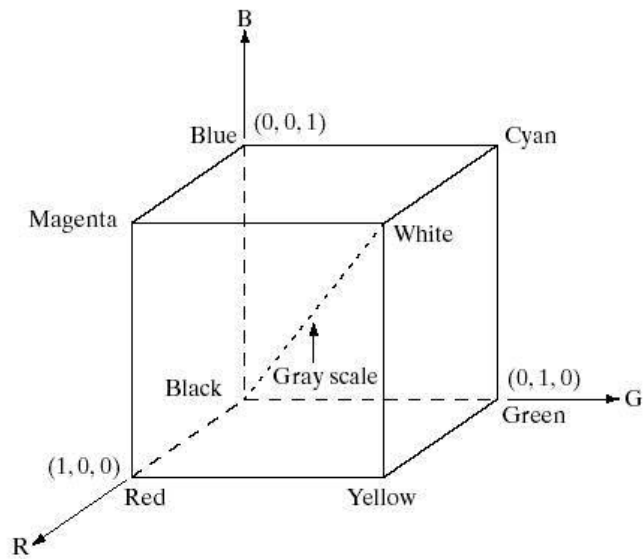


รูปที่ 2.2 แสดงค่าสีใน Gray Scale

เราสามารถเปลี่ยนภาพจากระบบสี RGB เป็น Gray Scale โดยใช้สมการดังต่อไปนี้ในการเปลี่ยน

$$y = 0.3R + 0.59G + 0.11B \quad (2.1)$$

โดย y แทน ค่า Gray Scale นะจุด pixel ที่เราต้องการหา
R แทน ค่าของสีแดงของจุดที่ต้องการหา
G แทน ค่าของสีเขียวของจุดที่ต้องการหา
B แทน ค่าของสีน้ำเงินของจุดที่ต้องการหา



รูปที่ 2.3 แบบสี RGB ในรูปแบบพิกัด 3 มิติ แสดงจุด pixel ที่ต้องการหา

ตัวอย่าง เมื่อนำภาพมาเปลี่ยนจากระบบสี RGB เป็น Gray Scale



รูปที่ 2.4 การเปลี่ยนระบบสี RGB เป็น Gray Scale

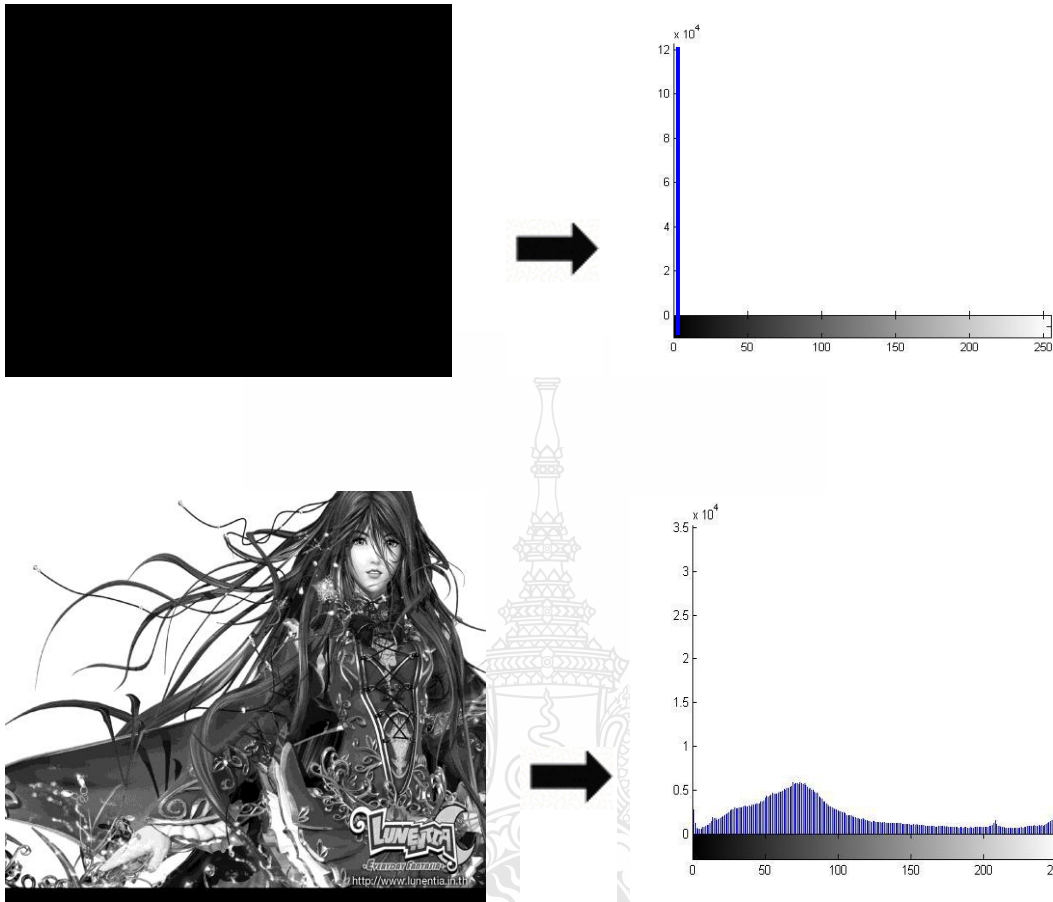
2.4 ภาพกราฟแจกแจงความถี่ (Image Histogram)

เนื่องจากข้อมูลทางสถิติจะอยู่ภายใต้ความแปรผันเสมอ ในการวิเคราะห์ ดังนั้น การพิจารณาจะทำได้ง่าย ถ้าข้อมูลอยู่ถูกสรุปให้อยู่ในรูปฮิสโตแกรม ซึ่งฮิสโตแกรมหมายถึง กราฟที่แสดงความแปรผันของข้อมูล ทั้งแนวโน้มสู่ศูนย์กลาง ค่าการกระจายและรูปทรงของการแปรผัน

ฮิสโตแกรมของภาพ Gray Scale เป็นกราฟแสดงความถี่หรือจำนวนพิกเซลที่ระดับความสว่างแต่ละค่า โดยมีแกนอนเป็นระดับความเข้มของแสง ตั้งแต่ 0 ถึง 255 และมีแกนตั้งแสดงจำนวนพิกเซลที่ความเข้มแสงที่ตำแหน่งนั้นๆ เช่น ถ้าตำแหน่งความเข้มของแสงใดมีความถี่สูงที่สุดก็แสดง

ว่าภาพนั้นมีพิกเซลที่มีค่าความเข้มของแสงที่ค่าันมากที่สุด
ตัวอย่าง

ดังตัวอย่างของฮิสโตแกรมของภาพ



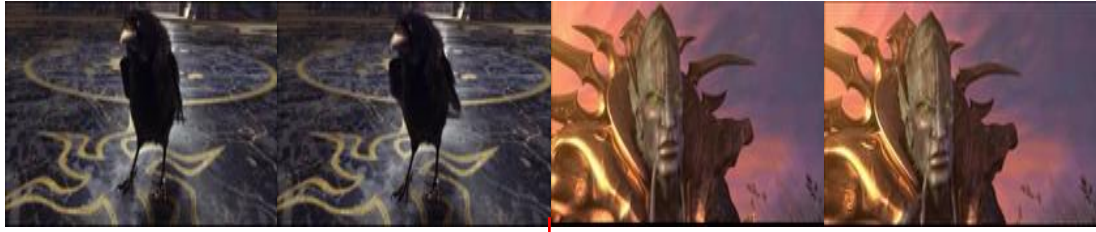
รูปที่ 2.5 ตัวอย่าง Histogram ของภาพ

2.5 จุดเปลี่ยนของวิดีโอ

จุดเปลี่ยนของวิดีโอ (Scene Change) เป็นจุดที่ผู้จัดทำวิดีโอตัดต่อภาพจากฉากหนึ่งไปยังอีกฉากหนึ่ง ซึ่งเราเรียกแต่ละฉากของวิดีโอว่า Scene Shot โดยที่ในวิดีโอ 1 เรื่องจะประกอบไปด้วย Scene Shot มากมายและแต่ละ Scene Shot จะถูกเชื่อมโยงกันโดยใช้ Effect ในการตัดต่อวิดีโอ ซึ่งสามารถแบ่งออกเป็น 2 ประเภทหลักๆ คือ

2.5.1 การเปลี่ยนแบบทันทีทันใด (Rapid Scene Change)

เป็นเอฟเฟ็คในการเปลี่ยนภาพที่นิยมใช้มากที่สุด การเปลี่ยนแบบ Rapid เป็นการเปลี่ยนของวิดีโอแบบทันทีทันใด คือ เปลี่ยนจาก ฉากหนึ่งไปเป็นอีกฉากหนึ่ง โดยการนำ Scene Shot 2 อันมาต่อกัน ดังตัวอย่างในรูปที่ 2.6



จุดที่เกิดการเปลี่ยนซีนของภาพ VDO



รูปที่ 2.6 การเปลี่ยนภาพแบบ Rapid Scene Change

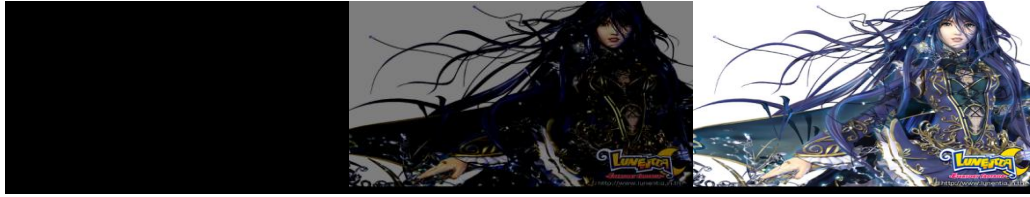
2.5.2 จุดเปลี่ยนแบบต่อเนื่อง (Gradually Scene Change)

คือจุดเปลี่ยนของวิดีโอที่นำเอา Scene Shot 2 อันมาต่อกันโดยการใช้ Effect ต่างๆ ทางคอมพิวเตอร์กราฟิกส์มาใช้ เพื่อให้การเชื่อมต่อดูราบเรียบขึ้น หรือสวยงามขึ้น จุดเปลี่ยนแบบนี้ทำให้เกิดเฟรมที่ใช้ในการเชื่อมต่อมากกว่า 1 เฟรม สามารถแบ่งจุดเปลี่ยนประเภทนี้ได้เป็น 3 ประเภทหลักๆ คือ

ก. Fade Scene Change

การเปลี่ยนแบบ fade เป็นการเปลี่ยนอีกแบบที่นิยมใช้กันมากอีกแบบหนึ่ง การเปลี่ยนแบบ fade สามารถแบ่งออกได้เป็น 2 แบบ คือ fade in และ fade out ซึ่งทั้งสองแบบมีวิธีการเปลี่ยนดังนี้คือ

Fade in ลักษณะการเปลี่ยนแบบ fade in คือ ภาพวิดีโอจะปรากฏขึ้นอย่างช้าๆ จาก Blank screen จนกระทั่งปรากฏเป็นภาพ ดังตัวอย่าง



ภาพวิดีโอจะปรากฏขึ้นมาจากภาพที่ดำมืด



รูปที่ 2.7 การเปลี่ยนภาพแบบ Fade Scene Change ลักษณะ Fade in

Fade-in สามารถเขียนเป็นสมการทางคณิตศาสตร์ได้ดังสมการที่ 2.2

$$S_n = \begin{cases} \left[1 - \left(\frac{n-L_1}{F}\right)\right]C + \left(\frac{n-L_2}{F}\right)g_n(i, j) & 0 \leq n < L_1 \\ \left[1 - \left(\frac{n-L_1}{F}\right)\right]C + \left(\frac{n-L_2}{F}\right)g_n(i, j) & L_1 \leq n \leq (L_1 + F) \\ \left(\frac{n-L_2}{F}\right)g_n(i, j) & (L_1 + F) < n \leq L_2 \end{cases} \quad (2.2)$$

$S_n(i, j)$ คือ สัญญาณภาพออก

$g_n(i, j)$ คือ สัญญาณภาพจาก B

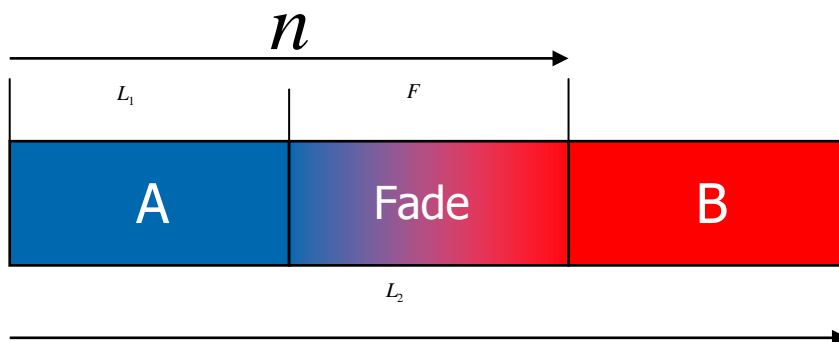
C คือ สัญญาณภาพจาก A ที่เป็นค่าคงที่

L_1 คือ ความยาวของสัญญาณภาพ A

F คือ ความยาวของเอฟเฟ็ค

L_2 คือ ความยาวของสัญญาณภาพทั้งหมด

n คือ จำนวนเฟรม



รูปที่ 2.8 อธิบายการทำงานแบบ Fade in

Fade out ลักษณะการเปลี่ยนแบบ fade out คือ ภาพวิดีโอจะหายไปอย่างช้าๆ จนกลายเป็น bank screen ดังตัวอย่าง



ภาพวิดีโอจะค่อยๆหายไป



รูปที่ 2.9 การเปลี่ยนภาพแบบ Fade Scene Change ลักษณะ Fade out

Fade-out สามารถเขียนเป็นสมการทางคณิตศาสตร์ได้ดังสมการที่ 2.3

$$S_n = \begin{cases} 1 & 0 \leq n < L_1 \\ \left[1 - \left(\frac{n - L_1}{F}\right)\right] g_n(i, j) + \left(\frac{n - L_2}{F}\right) C & L_1 \leq n \leq (L_1 + F) \\ 0 & (L_1 + F) < n \leq L_2 \end{cases} \quad (2.3)$$

$S_n(i, j)$ คือ สัญญาณภาพออก

$g_n(i, j)$ คือ สัญญาณภาพจาก A

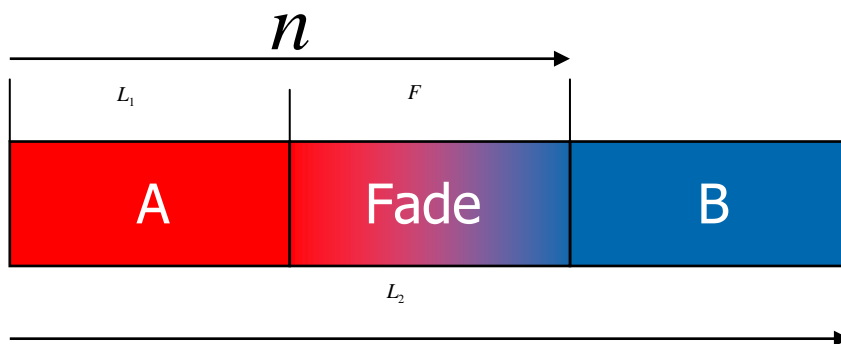
C คือ สัญญาณภาพจาก B ที่เป็นค่าคงที่

L_1 คือ ความยาวของสัญญาณภาพ A

F คือ ความยาวของเอฟเฟ็ค

L_2 คือ ความยาวของสัญญาณภาพทั้งหมด

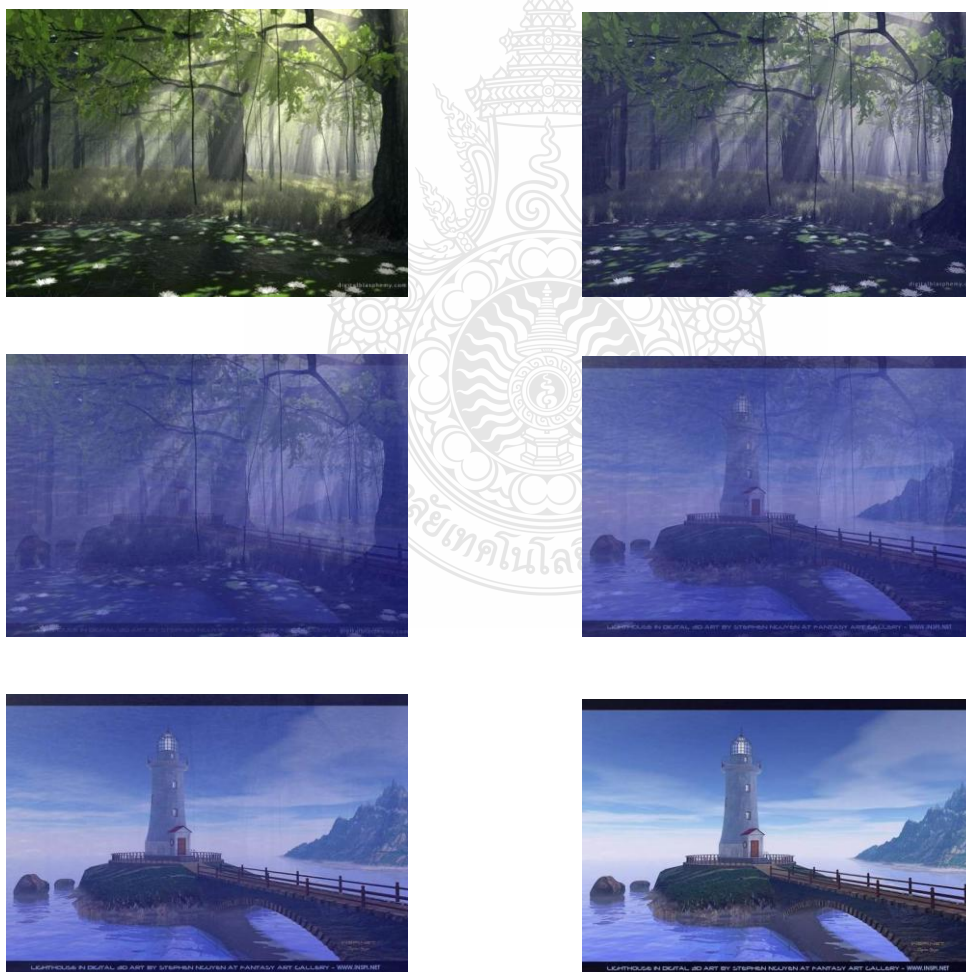
n คือ จำนวนเฟรม



รูปที่ 2.10 อธิบายการทำงานแบบ Fade out

ข. Dissolve Scene Change

เป็นเอฟเฟกต์ที่มีความนิยมใช้ในการตัดต่อ วิดีโอ อีกแบบหนึ่ง ซึ่งลักษณะของการเปลี่ยนแบบ Dissolve คือ ภาพหนึ่งจะค่อยๆหายไป และในขณะเดียวกันจะมีภาพใหม่ค่อยๆปรากฏขึ้น



รูปที่ 2.11 การเปลี่ยนภาพแบบ Dissolve Scene Change

การเปลี่ยนของภาพ VDO แบบ Dissolve ก็คือการเปลี่ยนของภาพ VDO แบบ Fade สมการนี้คือการเปลี่ยนแปลงโดยใช้เอฟเฟ็คแบบ Dissolve จากสัญญาณภาพ A ไป B สำหรับสมการ Fade-in คือสมการ Dissolve ที่สัญญาณภาพ A มีค่าคงที่ขณะที่ Fade-out คือสมการ Dissolve ที่สัญญาณภาพ B มีค่าคงที่นั่นเอง สามารถเขียนสมการ Dissolve ได้ดังสมการที่ 2.4

$$S_n = \begin{cases} \left[1 - \frac{f_n(i,j)}{F} \right] f_n(i,j) + \left(\frac{n-L_2}{F} \right) g_n(i,j) & 0 \leq n < L_1 \\ & L_1 \leq n \leq (L_1 + F) \\ & (L_1 + F) < n \leq L_2 \end{cases} \quad (2.4)$$

$S_n(i,j)$ คือ สัญญาณภาพออก

$f_n(i,j)$ คือ สัญญาณภาพจาก A

$g_n(i,j)$ คือ สัญญาณภาพจาก B

L_1 คือ ความยาวของสัญญาณภาพ A

F คือ ความยาวของเอฟเฟ็ค

L_2 คือ ความยาวของสัญญาณภาพทั้งหมด

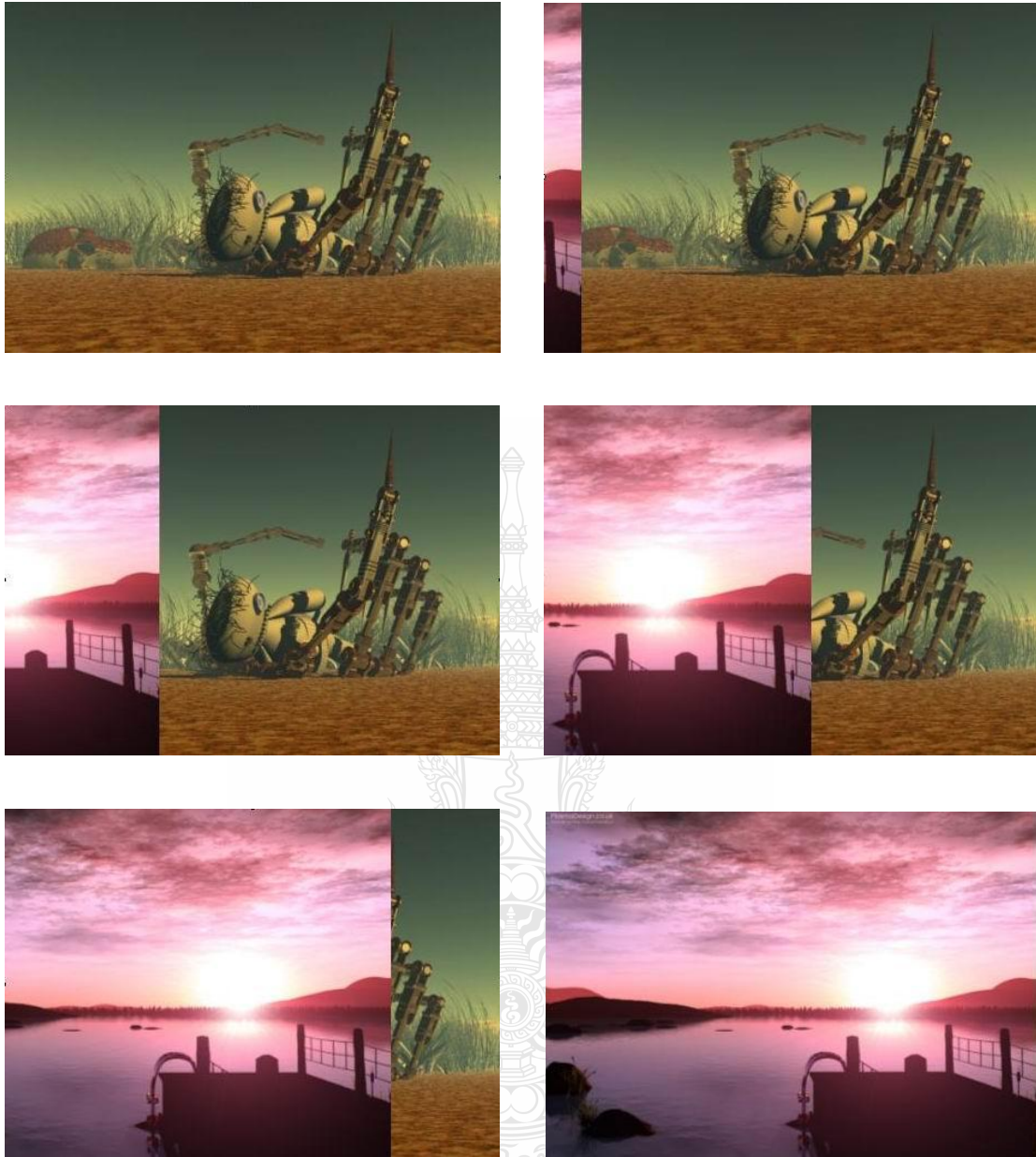
n คือ จำนวนเฟรม

ก. การเปลี่ยนแบบ Wipe Scene Change

เป็นเอฟเฟ็คการเปลี่ยนของภาพอีกแบบหนึ่งที่พบได้บ่อย โดยมีลักษณะของการเปลี่ยนคือภาพใหม่จะแทนที่เข้ามาแทนที่ภาพเก่า การเปลี่ยนแบบ WIPE สามารถทำได้หลายแบบ สามารถแบ่งออกเป็นประเภทใหญ่ ๆ 4 อย่าง คือ

- 1) การกวาดภาพทางแนวนอน จากซ้ายไปขวา หรือจากขวาไปซ้าย
- 2) การกวาดภาพทางแนวตั้ง จากบนลงล่าง และจากล่างขึ้นบน
- 3) การกวาดภาพทางแนวเฉียง จากมุมซ้าย หรือจากมุมขวา
- 4) การกวาดภาพในรูปทรงเรขาคณิต เช่น วงกลม สามเหลี่ยม

ตัวอย่าง Wipe Scene Change แบบการกวาดภาพทางแนวนอน จากซ้ายไปขวาดังรูปที่ 2.12



รูปที่ 2.12 การเปลี่ยนภาพแบบ Wipe Scene Change

2.6 การหาจุดเปลี่ยนของวิดีโอ

2.6.1 การหาจุดเปลี่ยนของการเปลี่ยนแบบทันทีทันใด (Rapid Scene Change Detection)

โดยพื้นฐานแล้วมีวิธีการหาทั้งหมด 3 วิธีคือ

ก. Template Matching Based Scene Change Detection [6]

หลักการของ Template matching based scene change detection คือ ถ้าความเข้มของแสงในภาพ มีการเปลี่ยนแปลงในจุดต่างๆเป็นจำนวนมาก แสดงว่ามีโอกาสที่จะมีการเปลี่ยนแปลง

ของภาพสูง ถ้าภาพมีความเข้มของแสงเปลี่ยนแปลงในจุดต่างๆเป็นจำนวนน้อย โอกาสที่จะมีการเปลี่ยนแปลงของภาพน้อย

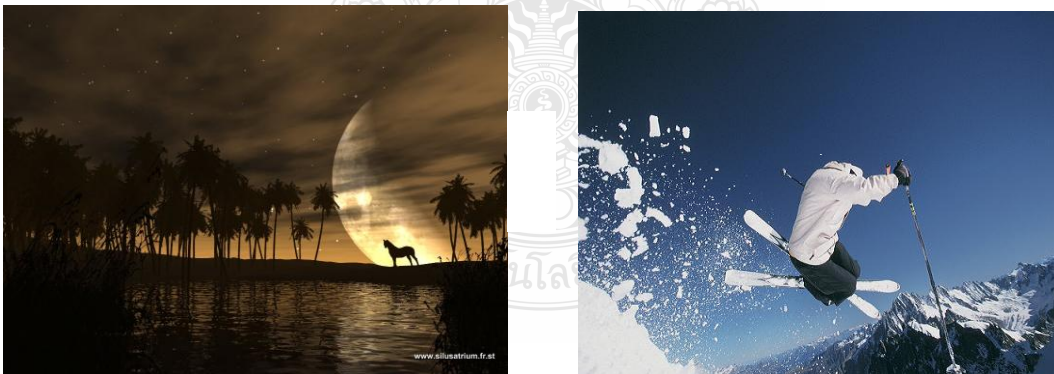
การนำ Template matching based โดยการใช้ Absolute Value เป็นการนำภาพ 2 ภาพมาลบกัน โดยตรงเพื่อดูว่าความเข้มของแสงในภาพมีการเปลี่ยนแปลงมากน้อยเพียงใด ดังสมการที่ 2.5

$$\delta = \frac{\sum_{i=1}^M |I(x, y, j) - I(x, y, k)|}{\sum_{i=1}^M \{I(x, y, j) + I(x, y, k)\}}, 0 \leq \delta \leq 1 \quad (2.5)$$

ตัวอย่างของการใช้ Absolute Value แสดงในรูปที่ 2.13



ค่าที่ได้ = 0.0336



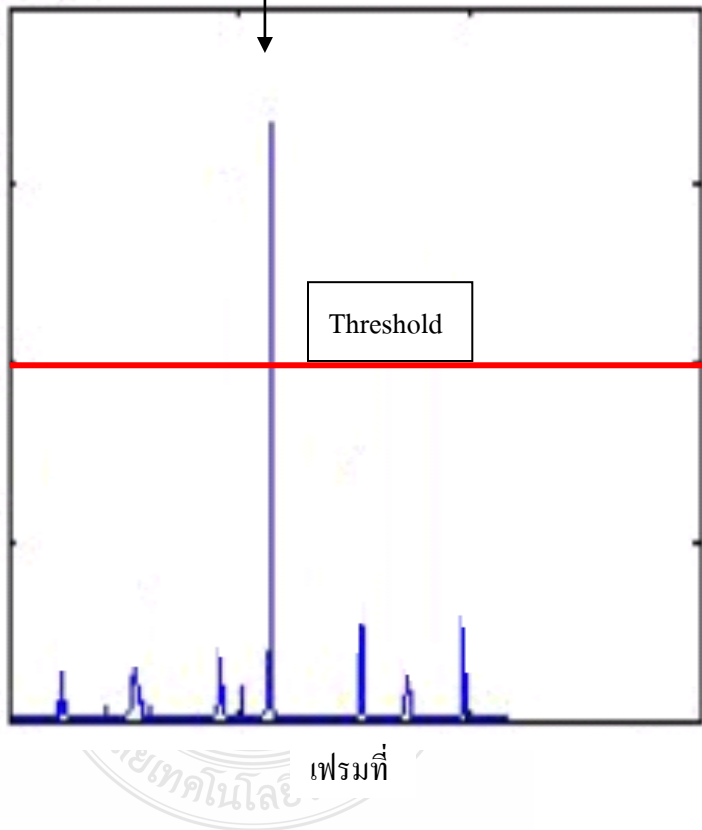
ค่าที่ได้ = 0.852

รูปที่ 2.13 ตัวอย่าง Template matching based Absolute Value

จากรูปจะเห็นได้ว่าเมื่อภาพมีลักษณะใกล้เคียงกันจะได้ผลลัพธ์ที่มีค่าน้อย แต่หากภาพที่มีความแตกต่างกันจะได้ค่ามาก ดังนั้นเมื่อนำความแตกต่างของค่าที่ได้มา Plot จะได้รูปที่ 2.14 ค่าที่มีค่ามากคือจุดเปลี่ยนของวิดีโอนั่นเอง



จุดที่เกิดค่าความแตกต่างกันมากคือ
จุดที่มีการเปลี่ยน



รูปที่ 2.14 การหาความแตกต่างของภาพโดยแสดงเป็นกราฟ

นอกจากนี้ ยังสามารถนำทฤษฎี Template matching มาใช้กับทฤษฎี Chi-square ดังสมการที่ 2.6

$$x^2 = \sum_{i=1}^M \frac{\{I(x, y, j) - I(x, y, k)\}^2}{\{I(x, y, j) + I(x, y, k)\}}, x^2 \geq 0 \quad (2.6)$$

เนื่องจากภาพมีมากมายหลากหลายรูปแบบ เมื่อนำภาพมาลบกันขึ้นเดียวกันย่อมให้ผลที่มีค่าความแตกต่างน้อย ในขณะที่ภาพที่อยู่ในคนละซีนน่าจะทำให้มีค่าความแตกต่างมากขึ้น และทำให้ทราบได้ว่าการเปลี่ยนแปลงภาพแล้ว แต่ไม่เป็นเช่นนั้นเสมอไป

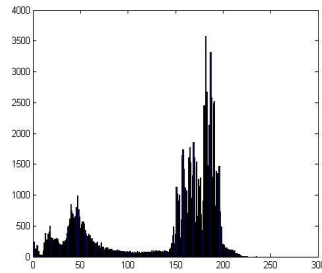
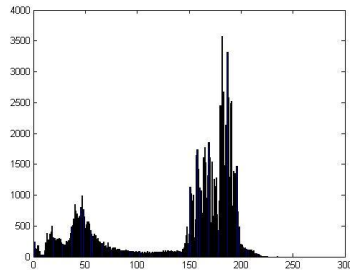
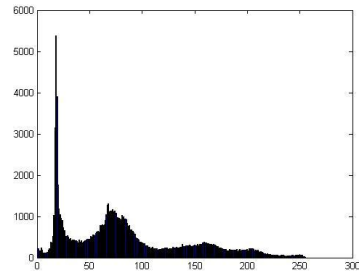
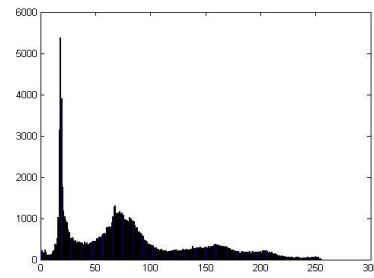
หากมีวิธีโอที่การเปลี่ยนซีนเกิดขึ้นแล้ว ซีนที่เกิดการเปลี่ยนแปลง กลับมีความคล้ายคลึงกับซีนเดิม ถึงแม้ว่านำมาลบกันได้ค่าความแตกต่างมาค่าหนึ่ง แต่ก็ไม่เพียงพอที่จะระบุได้ว่ามีการเปลี่ยนแปลงซีนวิธีโอเกิดขึ้น เราสามารถแก้ไขได้โดยการนำมายกกำลังสองเข้าไปจะทำให้ค่าผลความแตกต่าง แยกออกจากกันอย่างเห็นได้ชัด

ข. ทฤษฎี Histogram based Scene Change Detection [6]

ในเมื่อความเข้มของแสงแต่ละจุดภายในภาพมีการเปลี่ยนแปลง Histogram ของแต่ละภาพน่าที่จะมีการเปลี่ยนแปลงด้วย ดังนั้น Histogram ของภาพที่อยู่ในซีนเดียวกันก็จะมี การเปลี่ยนแปลงของ Histogram น้อย แต่ถ้าเป็นภาพที่อยู่กันคนละซีน ค่าของ Histogram จะมีการเปลี่ยนแปลงมากเพราะฉะนั้น เราสามารถหาค่าความแตกต่างของภาพได้โดยการนำ Histogram ของแต่ละรูปมาลบกัน ถ้าผลลัพธ์แตกต่างกันน้อย โอกาสที่จะเกิดการเปลี่ยนแปลงของซีนก็มีความเป็นไปได้ต่ำ แต่ถ้าผลลัพธ์แตกต่างกันมาก โอกาสที่จะเกิดการเปลี่ยนแปลงของซีนก็มีความเป็นไปได้สูง

วิธีการนำไปใช้จะคล้ายกับ Template matching รวมทั้งการนำไปประยุกต์ใช้ กับ Chi-square และ การนำไปใช้กับ Absolute ด้วย

จากรูปที่ 2.15 แสดงภาพของ Histogram ของภาพที่มีความใกล้เคียงกันและ ต่างกัน จะเห็นได้ว่าถ้าภาพที่อยู่ติดกัน และเป็น Scene Shot เดียวกัน จะมีลักษณะของภาพกราฟ Histogram ที่คล้ายกัน หากภาพแตกต่างกันจะได้ภาพกราฟที่แตกต่างกันด้วย



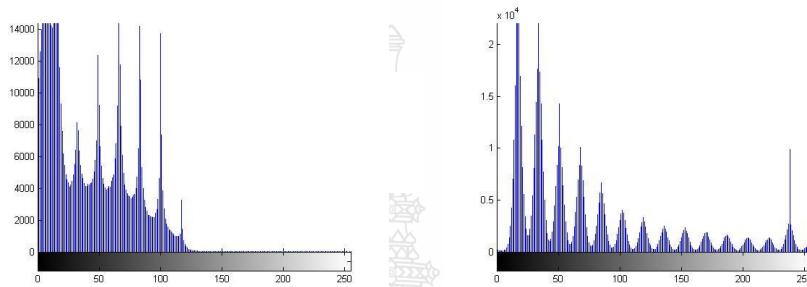
รูปที่ 2.15 ลำดับของภาพและ Histogram ของภาพในแต่ละเฟรม

- Chi-square value of Histogram
เมื่อรูปภาพมีความหลากหลายและมากมาย เราสามารถใช้สมการที่ 2.7 ในการหาจุดเปลี่ยนของวิดีโอได้ ดังตัวอย่างในรูปที่ 2.16

$$x^2 = \sum_{i=1}^M \frac{\{h_j(i) - h_k(i)\}^2}{\{h_j(i) + h_k(i)\}}, x^2 \geq 0 \quad (2.7)$$



=



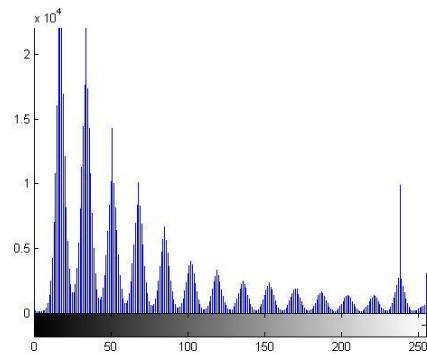
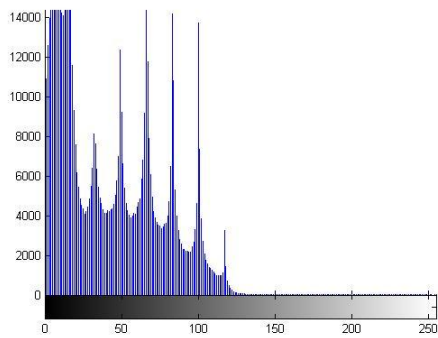
ค่าที่คำนวณได้ = 1.1865e+005

รูปที่ 2.16 การหาผลลัพธ์ตามทฤษฎี Chi-square value of histogram

- Absolute Value of Histogram

เป็นอีกวิธีการหนึ่งที่นิยมใช้ในการหาจุดเปลี่ยนเนื่องจากมีประสิทธิภาพที่ดี และใช้เวลาในการประมวลผลน้อย สมการที่ใช้คือสมการที่ 2.8 และตัวอย่างผลลัพธ์ดังรูปที่ 2.17

$$\delta = \frac{\sum_{i=1}^M \{h_j(i) - h_k(i)\}}{\sum_{i=1}^M \{h_j(i) + h_k(i)\}}, 0 \leq \delta \leq 0 \quad (2.8)$$



ค่าที่คำนวณได้ = 0.4735

รูปที่ 2.17 การหาผลลัพท์ตามทฤษฎี Absolute value of histogram

ก. Statistical Based Scene Change Detection [6]

ภาพแต่ละภาพจะมีค่าทางสถิติ คือ ค่ากลาง (Intensity mean) และค่าเบี่ยงเบนมาตรฐาน (intensity standard deviation) อยู่ด้วยเสมอ สามารถที่จะนำค่าทางสถิติของรูปภาพมาใช้ในการคำนวณ โดยค่าที่ทางสถิติที่นำมาใช้ในการคำนวณคือ ค่ากลาง (Intensity mean) และค่าเบี่ยงเบนมาตรฐาน (intensity standard deviation) เช่นทฤษฎี Likelihood ratio, assuming uniform 2nd order statistics. ดังสมการที่ 9 และค่าต่างที่คำนวณได้ดังตัวอย่างในรูป 2.18

$$\lambda = \frac{\left[\frac{\sigma_j + \sigma_k}{2} + \left(\frac{\mu_j - \mu_k}{2} \right)^2 \right]^2}{\sigma_j * \sigma_k} \quad (2.9)$$

λ คือ ค่า Likelihood Ration

σ คือ ค่าเบี่ยงเบนมาตรฐาน

μ คือ ค่ากลาง



$$\sigma = 50.0174$$
$$\mu = 145.3350$$



$$\sigma = 43.3147$$
$$\mu = 181.2906$$

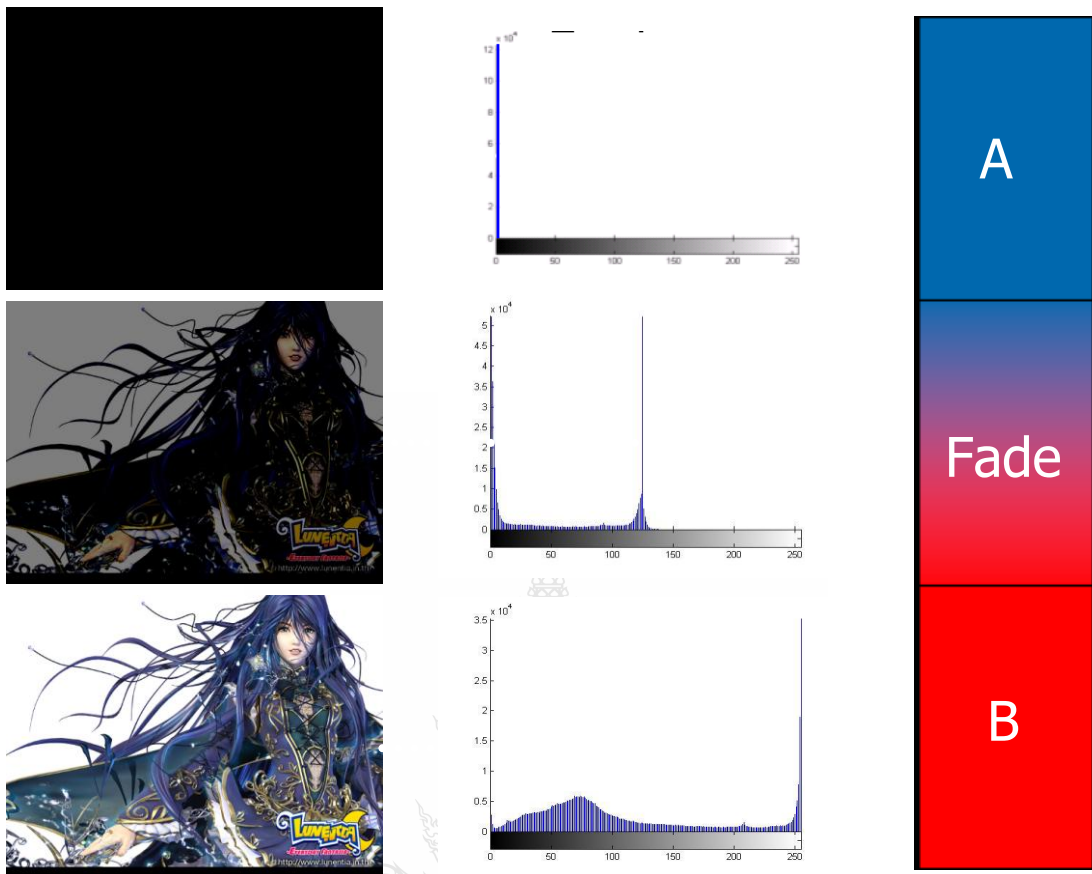
รูปที่ 2.18 ตัวอย่างค่าทางสถิติของภาพ

2.6.2 การหาจุดเปลี่ยนของการเปลี่ยนแบบ Fade

การเปลี่ยนของภาพ VDO แบบ Fade นี้ เป็นที่นิยมในปัจจุบัน เราจึงต้องศึกษาการหาจุดเปลี่ยนภาพของการเปลี่ยนภาพแบบ Fade ซึ่งการเปลี่ยนแบบนี้สามารถแบ่งออกเป็น 2 ลักษณะคือ Fade in และ Fade out ซึ่งทฤษฎีที่เราจะนำมาศึกษาลักษณะการเปลี่ยนภาพแบบ Fade คือ ทฤษฎี Automatic Detection Of Fade-in And Fade-out In Video Sequences Using Histograms [10,11]

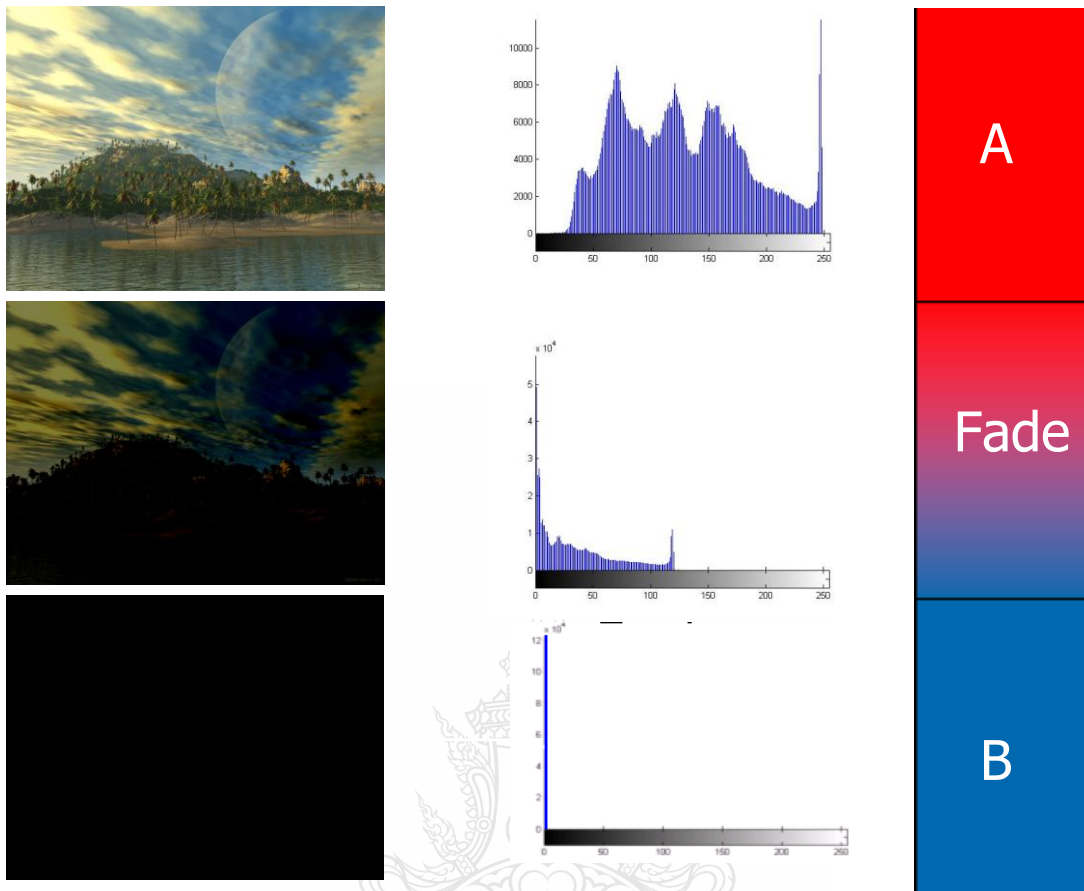
การหาจุดเปลี่ยนตั้งอยู่บน สมมติฐานที่ว่าเมื่อเกิดเหตุการณ์ Fade-in จะเริ่มจากการที่หน้าจออยู่บนฉากเปล่าๆ สีเดียว ดังนั้นจึงมี Histogram อยู่เพียงแค่ 1 แห่ง เมื่อภาพค่อยๆ ปรากฏขึ้น จำนวนของ Histogram ก็จะเพิ่มขึ้นตาม โดย WAC Fernando ได้เสนอวิธีการนำเอาค่าสูงสุดของภาพมาลบออกจากค่าต่ำสุดของภาพ หากมีค่าเพิ่มขึ้นเรื่อยๆตามจำนวนครั้งที่กำหนดแสดงว่ามี Fade-in เกิดขึ้น สำหรับการเกิด Fade-out หากว่ามีค่าน้อยลงเรื่อยๆ แสดงว่าเกิด Fade-out ขึ้นนั่นเอง

ก. ในกรณีของ Fade in สามารถแสดงผลของ Histogram ได้ดังรูปที่ 2.19



รูปที่ 2.19 ตัวอย่าง Histogram แบบ Fade-in

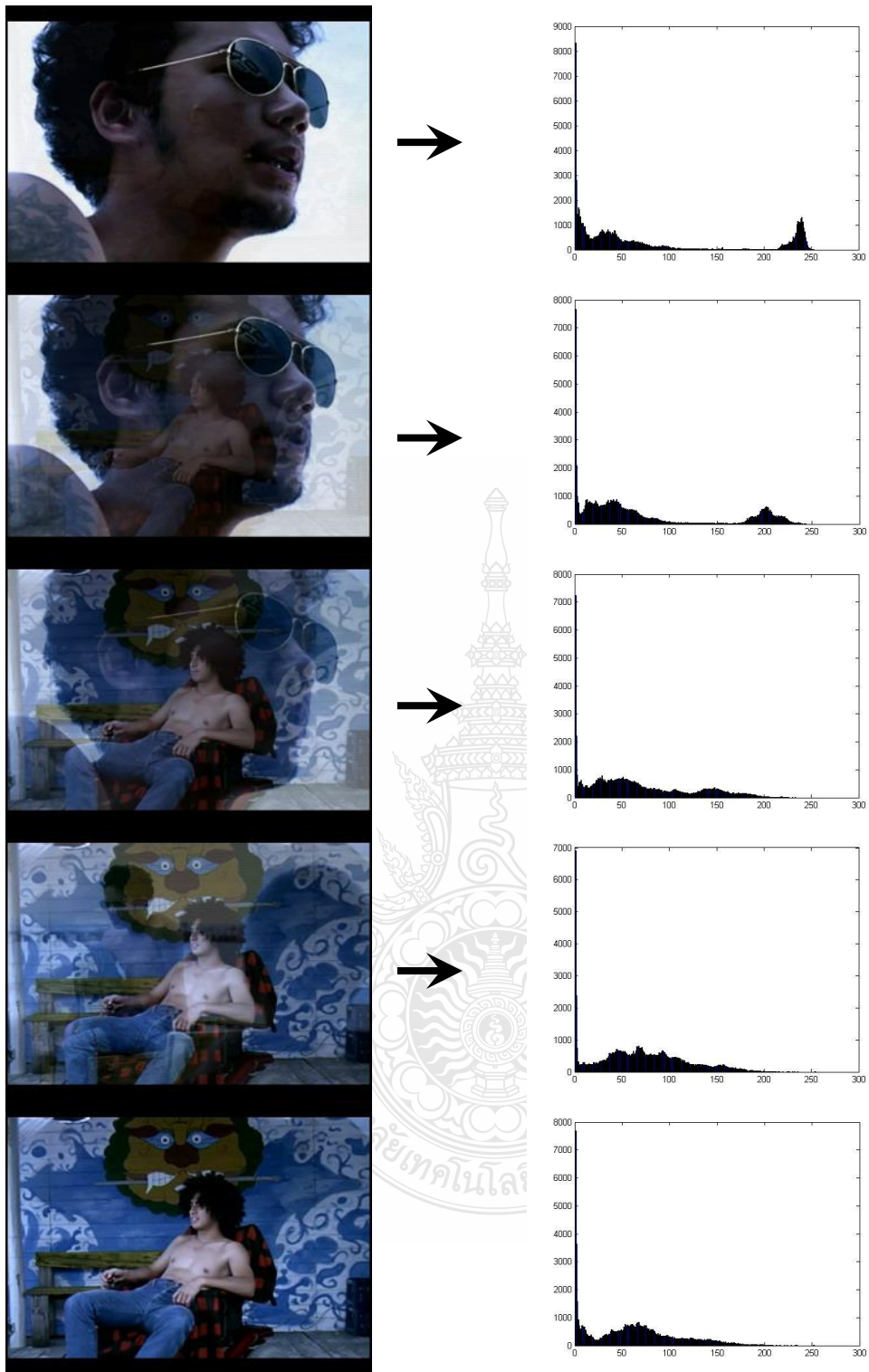
ข. ในกรณีของ Fade out สามารถแสดงผลของ Histogram ได้ดังรูปที่ 2.20



รูปที่ 2.20 ตัวอย่าง Histogram แบบ Fade-in

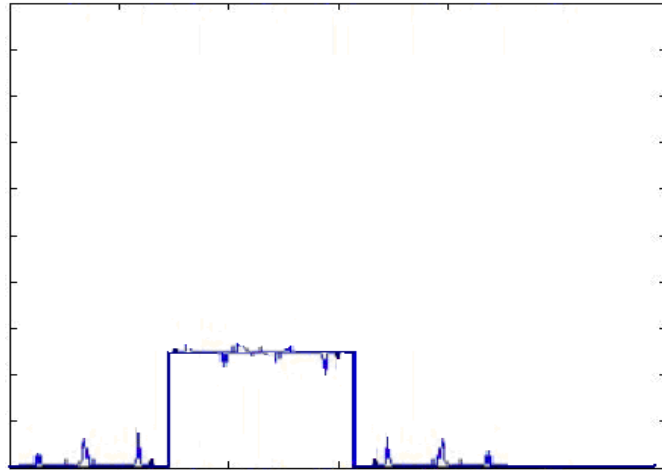
2.6.3 การหาจุดเปลี่ยนของการเปลี่ยนแบบ Dissolve

การเปลี่ยนของภาพ VDO แบบ Dissolve ก็คือการเปลี่ยนของภาพ VDO แบบ Fade สมการนี้คือการเปลี่ยนแปลงโดยใช้เอฟเฟ็คแบบ Dissolve จากสัญญาณภาพ A ไป B สำหรับสมการ Fade-in คือสมการ Dissolve ที่สัญญาณภาพ A มีค่าคงที่ขณะที่ Fade-out คือสมการ Dissolve ที่สัญญาณภาพ B มีค่าคงที่นั่นเอง สมการ Dissolve ดังสมการที่ 4 ตัวอย่างของ Dissolve เป็นดังรูปที่ 2.21



รูปที่ 2.21 Histogram ของการเปลี่ยนภาพแบบ Dissolve

การหาจุดเปลี่ยนของ Dissolve, WAC Fernando [12] ได้นำเสนอวิธีการหาใน วิดีโอที่ไม่ถูกบีบอัดโดยใช้วิธีการเดียวกันกับการหาของ Fade ซึ่งตัวอย่างผลดังแสดงในรูปที่ 2.22

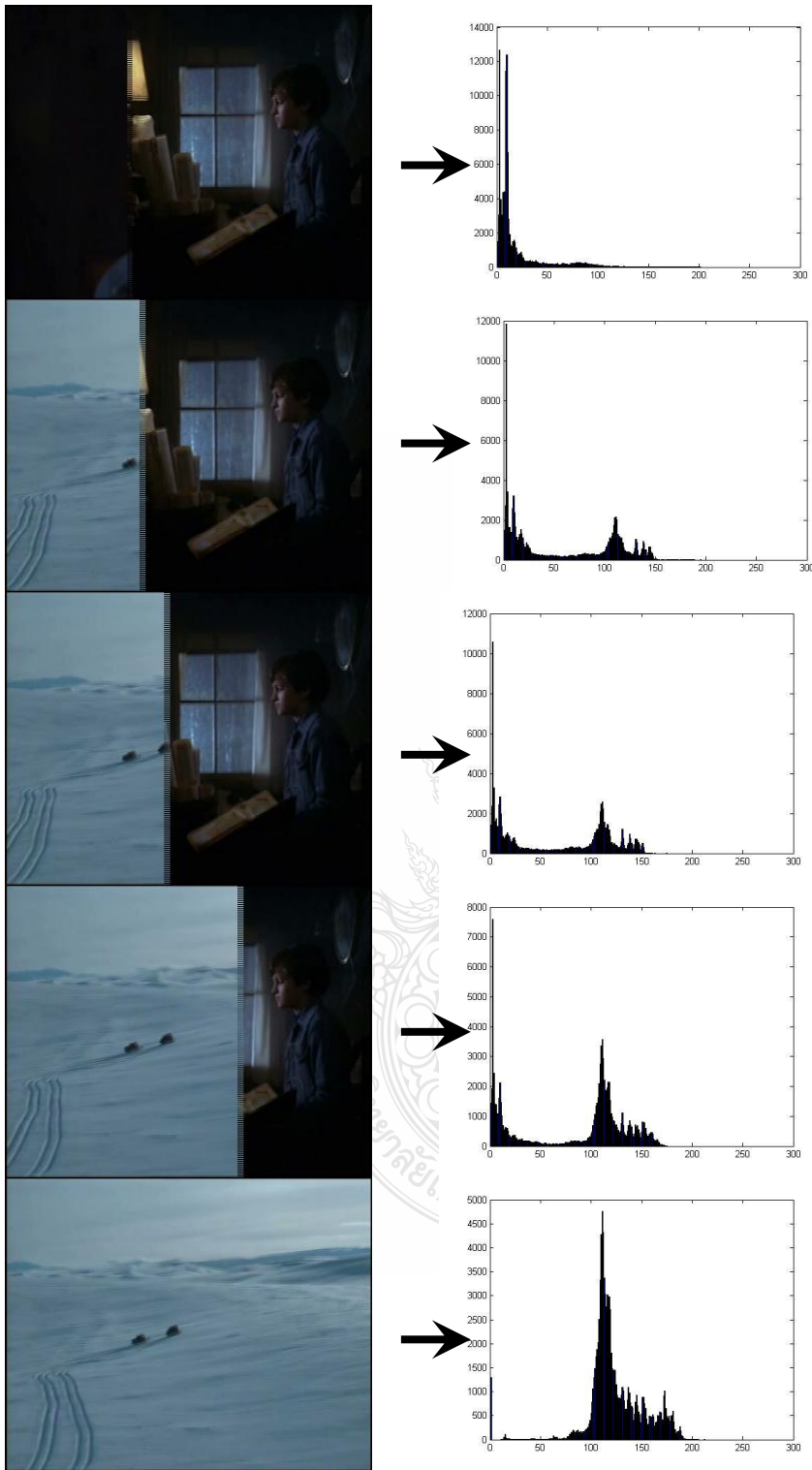


รูปที่ 2.22 กราฟเมื่อเกิด Dissolve ในวิดีโอ

จากรูปจะเห็นได้ว่าเมื่อเกิด Dissolve ขึ้นจะทำให้เกิดช่วงของค่าที่เพิ่มขึ้นเป็นระยะเวลาตามจำนวนเฟรมของการเกิด Dissolve นั้นเอง

2.6.4 การหาจุดเปลี่ยนของการเปลี่ยนแบบ Wipe

การเปลี่ยนภาพแบบ Wipe จะมีหลักการเดียวกับ Dissolve ก็คือการเปลี่ยนของภาพ VDO แบบ Fade สมการนี้คือการเปลี่ยนแปลงโดยใช้เอฟเฟ็คแบบ Wipe จากสัญญาณภาพ A ไป B สำหรับสมการ Fade-in คือสมการ Wipe ที่สัญญาณภาพ A มีค่าคงที่ขณะที่ Fade-out คือสมการ Wipe ที่สัญญาณภาพ B มีค่าคงที่นั่นเอง จากรูปที่ 2.23 จะเห็นได้ว่าการเปลี่ยนแปลงของ Histogram ที่คล้ายคลึงกันกับ Dissolve ดังนั้นเราสามารถนำหลักการของการหาจุดเปลี่ยนของ Dissolve มาใช้ได้ W.A.C.Fernando, C.N.Canagarajah and D.R.Bull [13] ได้นำเสนอวิธีการอย่างง่ายโดยการตรวจสอบการเปลี่ยนแปลงของ Histogram แต่จากการทดสอบได้ผลไม่ดีเท่าที่ควรเพราะการเปลี่ยนแปลงของภาพแค่บางจุดเท่านั้น



รูปที่ 2.23 Histogram ของการเปลี่ยนภาพแบบ Wipe

2.7 การใช้ทฤษฎี Recall-Precision ในการทดสอบประสิทธิภาพอัลกอริทึม

ในการทดสอบประสิทธิภาพของอัลกอริทึม จะใช้การทดสอบที่ใช้กัน โดยทั่วไปเรียกว่า Recall-Precision จากสมการที่ 2.10 และ 2.11

$$\text{Recall} = \frac{\text{Detects}}{\text{Detects} + \text{MissedDetect}} \times 100 \quad (2.10)$$

$$\text{Precision} = \frac{\text{Detects}}{\text{Detects} + \text{falsealarms}} \times 100 \quad (2.11)$$

โดยที่

Recall	คือ อัตราส่วนที่สามารถตรวจจับได้
Precision	คือ ค่าความถูกต้อง
Detects	คือ ค่าที่อัลกอริทึมตรวจจับได้
Missed Detect	คือ ค่าที่อัลกอริทึมตรวจจับไม่ได้
False alarms	คือ ค่าที่อัลกอริทึมตรวจจับผิด

โดยที่ค่า Recall คือค่าที่อัลกอริทึมที่สร้างขึ้นสามารถตรวจจับได้ถูกต้องและค่า Precision คือค่าความถูกต้องอัลกอริทึมที่สร้างขึ้น ดังนั้นค่าชี้วัดว่าอัลกอริทึมที่สร้างขึ้นดีหรือไม่ดีดูจากทั้ง 2 ค่า ไม่สามารถดูจากค่าใดค่าหนึ่งได้

บทที่ 3

วิธีการดำเนินการวิจัย

ในบทนี้จะกล่าวถึงรายละเอียดของวิธีการและขั้นตอนต่างๆ ที่ได้ดำเนินการในงานวิจัยของวิทยานิพนธ์นี้ โดยจะแบ่งส่วนของขั้นตอนหลักๆ ได้แก่ การออกแบบวิธีการจุดเปลี่ยนของวิดีโอของจุดเปลี่ยนแบบทันทีทันใด การหาจุดเปลี่ยนแบบ Fade การหาจุดเปลี่ยนแบบ Dissolve และการหาจุดเปลี่ยนแบบ Wipe โดยที่ออกแบบโดยการเปลี่ยนเป็นบล็อกขนาด 2*2 ทั้งแบบที่ไม่มีการซ้อนทับและซ้อนทับ นอกจากนี้ยังนำเสนอวิธีการแก้ปัญหาอัลกอริทึมของการหาจุดเปลี่ยนแบบ Fade ซึ่ง WAC Fernando [4] นำเสนอไว้แต่ไม่สามารถใช้งานได้กับวิดีโอโดยทั่วไปหลายประเภท

3.1 ข้อกำหนดของงานวิจัย

3.1.1 เครื่องคอมพิวเตอร์ที่ใช้

Software: MATLAB version 7.8

OS: Windows XP with service pack 2

CPU: Intel Pentium 4, 2GHz with 512 Kbytes cache memory,
400Mhz front side bus

Memory: DDR2100 512 MB

3.1.2 Format ของภาพที่ใช้ในการทดสอบ

BMP window bitmap file เพื่อให้ภาพที่ได้ไม่มีการบีบอัดของข้อมูลและทำให้มี Noise ของภาพที่น้อย

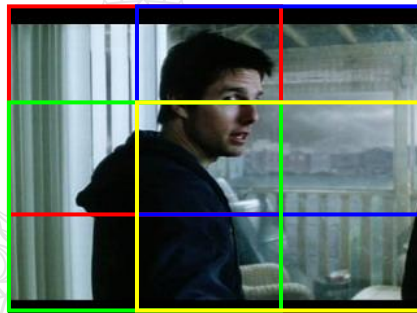
3.2 การหาจุดเปลี่ยนแบบทันทีทันใด (Rapid Scene Change Detection)

ในการหาจุดเปลี่ยนของภาพแบบ Rapid เราใช้วิธีการแบ่งภาพออกเป็นส่วนๆ เช่น คิดแบ่งเป็น 4 ส่วนของภาพ 1 ภาพ จากการแนะนำของ WAC Fernando [4] เพื่อให้มีความเหมาะสมระหว่างความเร็วและประสิทธิภาพดังรูปที่ 3.1

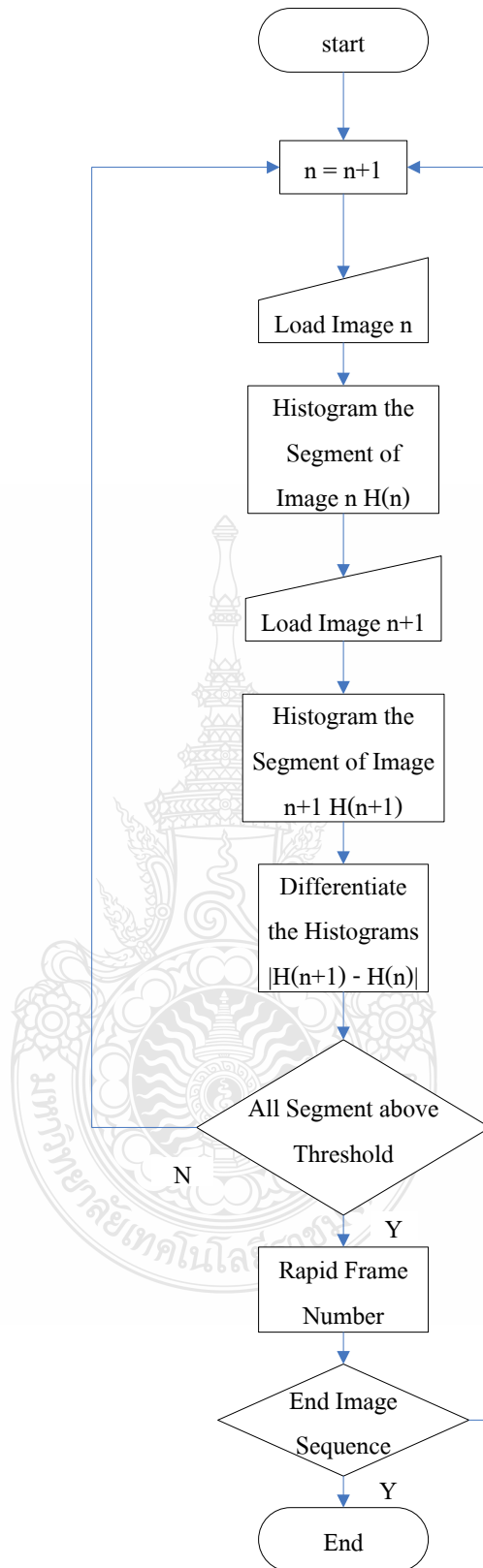


รูปที่ 3.1 การแบ่งส่วนของภาพในการหา จุดเปลี่ยนของภาพแบบ Rapid

และอีก 1 วิธีในการหาจุดเปลี่ยนของภาพ VDO แบบ Rapid โดยการแบ่งภาพออกเป็นส่วนๆ เช่นกัน แต่จะให้แต่ละส่วนซ้อนทับกัน ดังรูปที่ 3.2 เพื่อเพิ่มประสิทธิภาพของอัลกอริทึม เนื่องจากว่าอัลกอริทึมที่นำเสนอจะเพิ่มการประมวลข้อมูลเพิ่มขึ้นเป็น 4 เท่า ดังนั้นเราจึงใช้แค่ผลต่างของ Histogram เท่านั้น เพื่อลดการประมวลข้อมูลลง อัลกอริทึมที่นำเสนอดังรูปที่ 3.3



รูปที่ 3.2 การแบ่งส่วนของภาพแบบซ้อนทับในการหา จุดเปลี่ยนของภาพแบบ Rapid

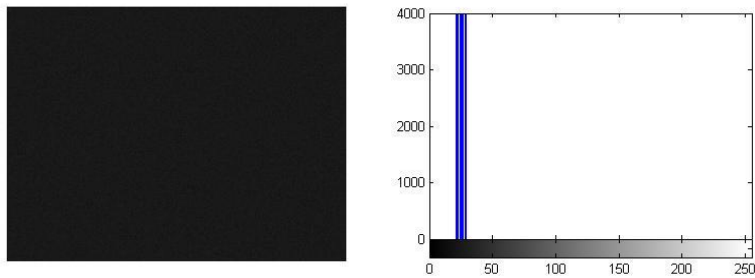


รูปที่ 3.3 Flowchart การทำงานของการหาจุดเปลี่ยนของภาพแบบ Rapid

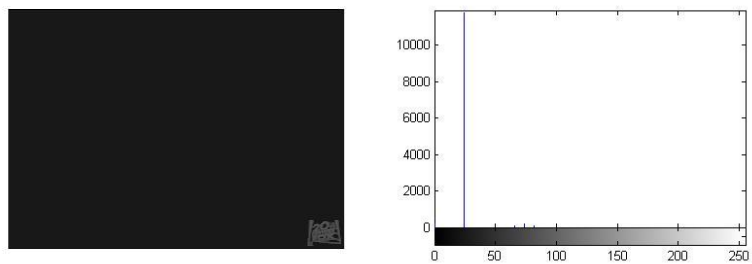
3.3 การหาจุดเปลี่ยนแบบต่อเนื่อง (Gradually Scene Change)

3.3.1 Fade Scene Change Detection [14]

อ้างอิงจากอัลกอริทึมที่นำเสนอโดย WAC Fernando [10,11] ซึ่งใช้ค่าสูงสุดของ Histogram ลบออกจาก ค่าต่ำสุดของ Histogram โดยการแบ่งส่วนของภาพออกเป็น 4 ส่วนเพื่อเพิ่มความถูกต้องของอัลกอริทึม เนื่องจากภาพที่ใช้ในการทดสอบได้ตั้งอยู่บนสมมติฐานที่ว่า การเริ่มต้นเมื่อเกิด Fade-in สัญญาณภาพที่ได้มีค่าเพียงค่าเดียวไม่มี noise ในภาพ และจุดสิ้นสุดของ Fade-out ก็มีลักษณะเช่นเดียวกันกับจุดเริ่มต้นของ Fade-in ดังนั้นอัลกอริทึมที่นำเสนอมาจึงไม่สามารถใช้งานกับวิดีโอที่เกิดการบีบอัดโดยใช้ Lossy Compression Algorithm ได้ นอกจากนี้โดยปกติแล้วตราของช่องเช่น ช่อง 7 สี หรือสัญลักษณ์ของบริษัทผู้ผลิต อาจแสดงอยู่ตลอดเวลา ทำให้ภาพที่ได้ ไม่มีทางที่จะเกิดเหตุการณ์ที่จะมีภาพที่มีสีเพียงสีเดียวหรือค่าเดียวได้เลย ทำให้อัลกอริทึมของ WAC Fernando ไม่สามารถทำงานได้เลย ตัวอย่างดังรูปที่ 3.4 และ รูปที่ 3.5



รูปที่ 3.4 ตัวอย่างภาพที่ได้เมื่อสิ้นสุด Fade-out จาก VCD



รูปที่ 3.5 ตัวอย่างภาพที่ได้เมื่อสิ้นสุด Fade-out เมื่อมี Logo บริษัทสีเดียวพื้นสีดำ

ซึ่งอัลกอริทึมที่นำเสนอจะสามารถแก้ปัญหาได้โดยการใช้การนับจำนวนของ histogram แทนที่จะใช้ค่าสูงสุดหรือต่ำสุด โดยเริ่มต้นจากการคำนวณค่า histogram ของภาพที่ถูกแบ่งส่วนเป็น 4 ส่วนเรียกว่า $H(i, n)$, เมื่อ i คือค่าของ histogram และ n คือลำดับช่องที่ของ histogram หลังจากนั้นก็ทำการคำนวณค่าจำนวนแห่งของ histogram จากสมการที่ 3.1

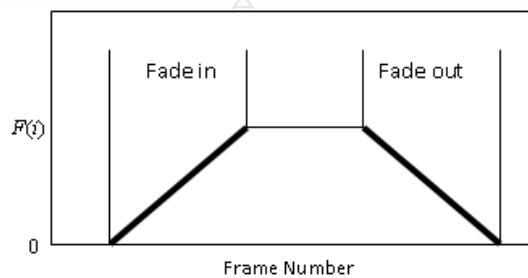
$$C(i) = \sum_{n=1}^{256} G(i, n) \quad \text{Where } G(i, n) = \begin{cases} 1 & H(i, n) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.1)$$

เมื่อ i มีค่าเท่ากับ $1..M$ โดยที่ M คือจำนวนเฟรมทั้งหมดของวิดีโอ

จากนั้นจึงทำการแก้ปัญหาเรื่องของ noise และ Logo โดยการใช้นon-linear Quantization เพื่อให้ค่าของ noise ที่ได้ถูกกำจัดไปได้จริงๆ จึงได้ทำการปรับเปลี่ยนสมการโดยใช้ค่าต่ำสุดที่หาได้จากช่วงเวลานี้เป็นเกณฑ์ ทำให้ได้การ Quantization ดังสมการที่ 3.2

$$F(i) = \frac{C(i) - \min(C)}{256 - \min(C)} \quad (3.2)$$

สมการที่ 13 ทำให้ ส่วนที่มีค่าน้อยถูกทำให้มีค่าเท่ากับ 0 ดังนั้นปัญหาเรื่องของ noise ก็จะถูกตัดทิ้งไปเราสามารถที่จะตรวจจับว่าเกิด Fade-in และ Fade-out ขึ้นเมื่อเกิดค่าที่เปลี่ยนแปลงของ F(i) ดังรูปที่ 3.6



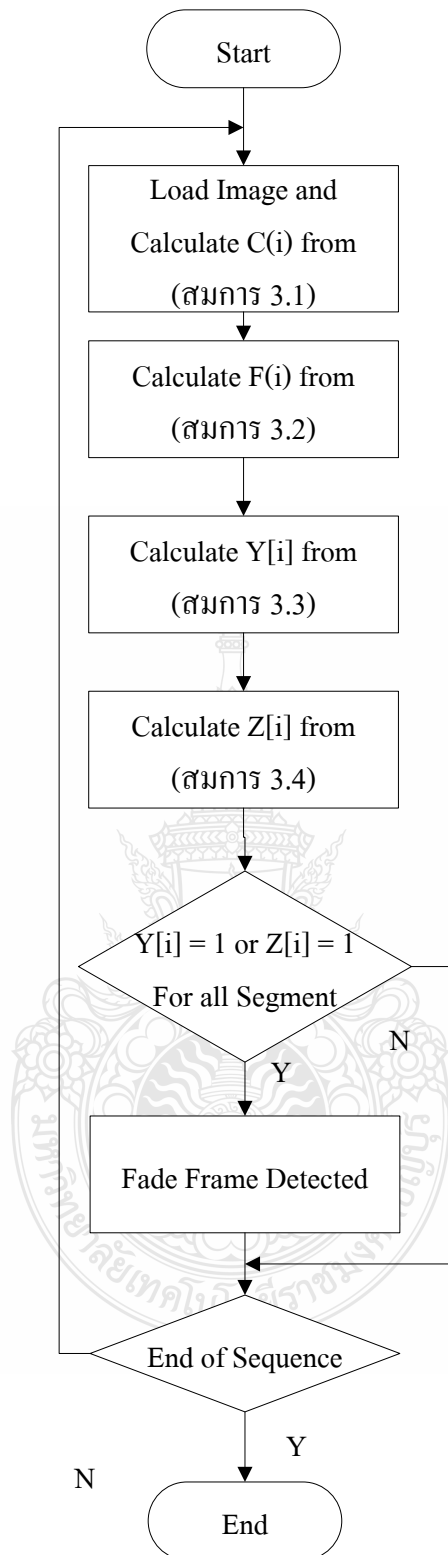
รูปที่ 3.6 ค่าของ F(i) ขณะที่เกิด Fade-in และ Fade-out

จากรูปจะเห็นได้ว่ามี 2 กรณีเกิดขึ้นคือ เมื่อเกิดกรณีที่ เป็น Fade-in ค่าของ F(i) จะเพิ่มขึ้นเรื่อยๆ โดยที่มีค่าเริ่มต้นจาก 0 ขณะเดียวกันเมื่อเกิด Fade-out ค่าของ F(i) ก็จะลดลงเรื่อยๆ เช่นกัน จนกระทั่งค่าที่ได้มีค่าเท่ากับ 0 จึงสิ้นสุด เพื่อตรวจจับจุดที่เกิด Fade-in และ Fade-out เราจะใช้สมการที่ 3.3 และ สมการที่ 3.4 ตามลำดับ

$$Y(i) = \begin{cases} True & F(i) < F(i+1) < F(i+2) < F(i+3) \\ False & otherwise \end{cases} \quad (3.3)$$

$$Z(i) = \begin{cases} True & F(i) > F(i+1) > F(i+2) > F(i+3) \\ False & otherwise \end{cases} \quad (3.4)$$

เมื่อ Y(i) มีค่าเท่ากับ 1 แสดงว่าเกิด Fade-in ขึ้นและต่อเนื่องไปเรื่อยๆจนกระทั่งถูกเปลี่ยนค่า ขณะเดียวกัน Z(i) มีค่าเท่ากับ 1 แสดงว่าเกิด Fade-out ขึ้นและต่อเนื่องไปเรื่อยๆจนกระทั่งถูกเปลี่ยนค่าเช่นเดียวกัน Flow chart การทำงานเป็นดังรูปที่ 3.7



รูปที่ 3.7 Flow Chart การทำงานของ Fade Scene Change Detection

3.3.2 Dissolve Scene Change Detection [15]

การหาจุดเปลี่ยนของแบบ Dissolve โดยใช้ข้อมูลของ Histogram สามารถที่จะเริ่มต้น โดยการคำนวณโดยการกำหนดค่าน้ำหนักของ Histogram ของแต่ละแท่งของ Histogram ทำให้ได้ Histogram ใหม่ดังสมการที่ 3.5

$$h1_{i,j} = h_{i,j} * n_i \quad (3.5)$$

เมื่อ,

$h_{i,j}$ = ค่าของ Histogram เฟรมที่ j

n_i = ค่าคงที่ มีค่า [1..256]

$h1_{i,j}$ = ค่าของ Histogram ใหม่เฟรมที่ j

จากนั้นทำการคำนวณหาค่าเฉลี่ยของ Histogram ที่ได้ตามสมการที่ 3.6

$$m_j = \text{mean}(h1_j) \quad (3.6)$$

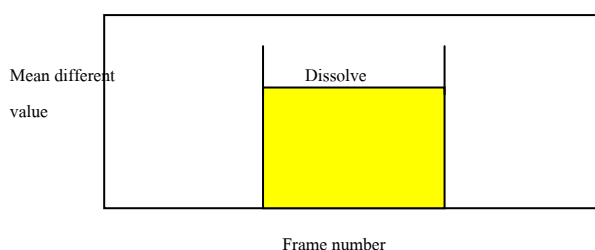
เมื่อ,

$h1_j$ = ค่าของ Histogram ใหม่เฟรมที่ j

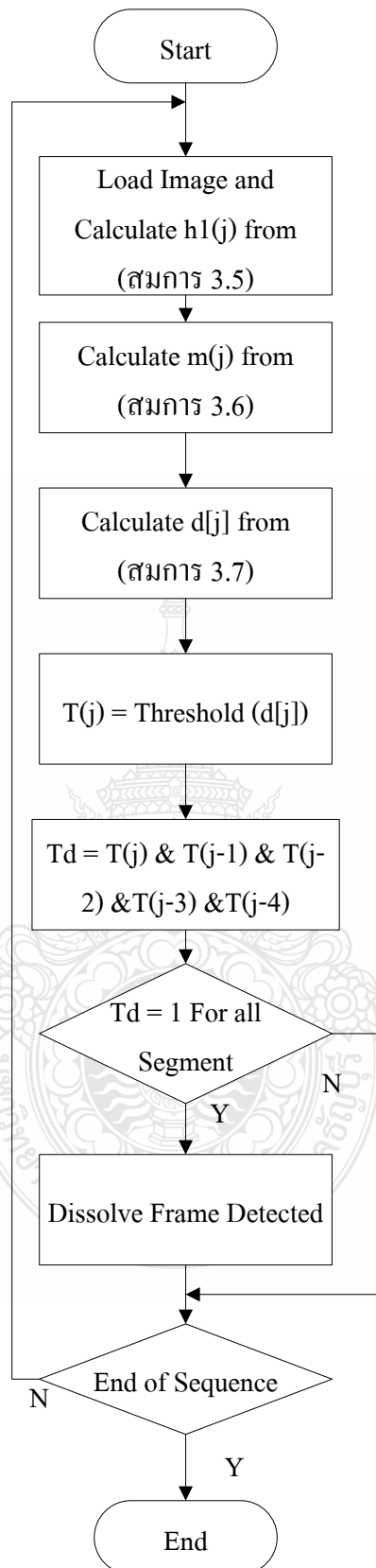
m_j = ค่าของ Histogram ใหม่เฟรมที่ j

จากนั้นทำการคำนวณหาค่าความแตกต่างของ Histogram ตามสมการที่ 3.7 ภายใต้ สมมติฐานที่ว่า เมื่อมีการเปลี่ยนแปลง Dissolve ขึ้น จะทำให้มีค่าความแตกต่างของค่าเฉลี่ยที่เพิ่มมากขึ้นด้วย โดยที่ผลลัพธ์ตามสมมติฐานตามอุดมคติจะได้ตามรูปที่ 3.8 และรูปที่ 3.9 แสดง Flow Chart การคำนวณหา Dissolve Scene Change Detection

$$d_j = |m_j - m_{j+1}| \quad (3.7)$$



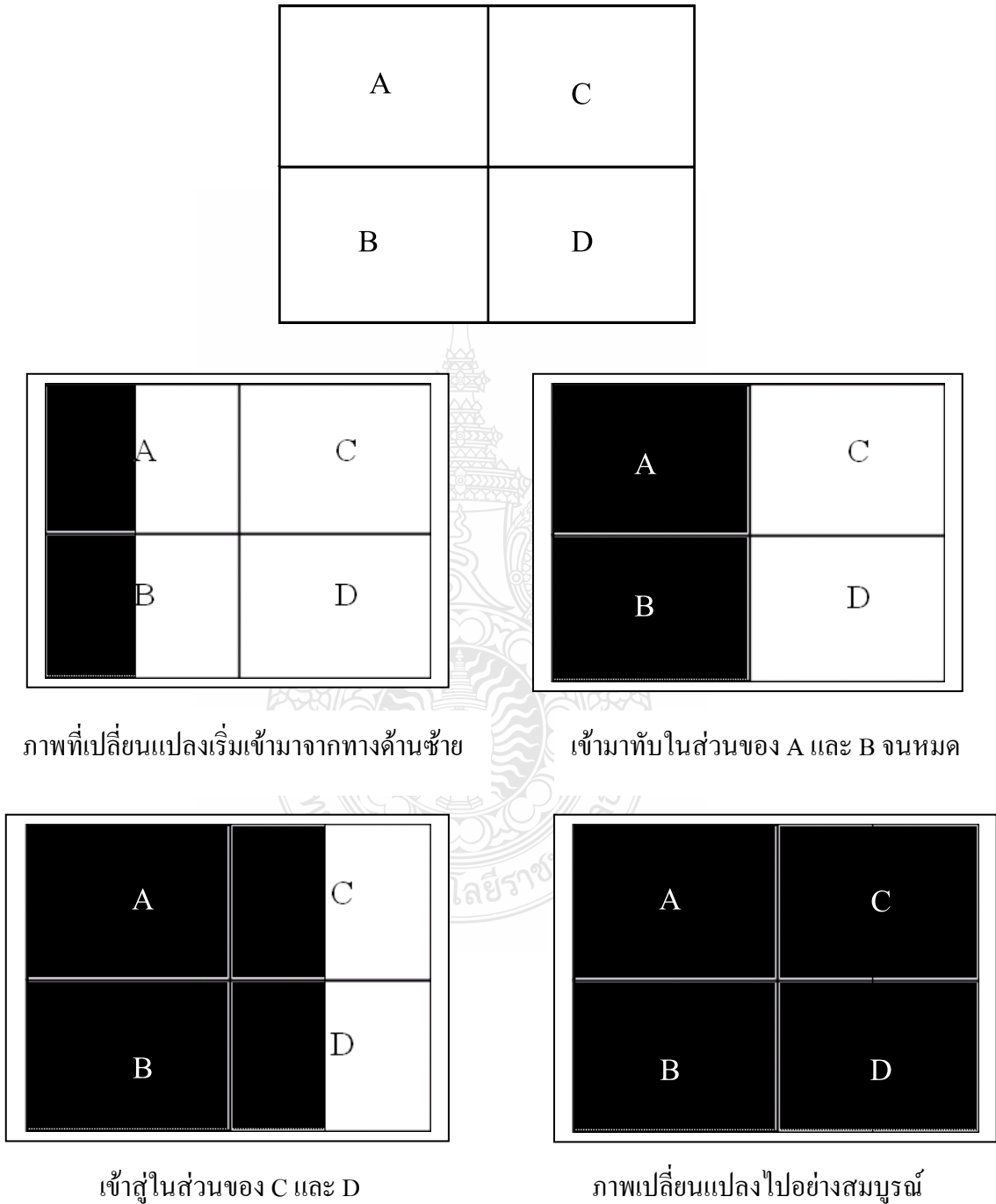
รูปที่ 3.8 ค่าในอุดมคติขณะที่เกิด Dissolve Scene Change



รูปที่ 3.9 Flow Chart การทำงานของ Dissolve Scene Change Detection

3.3.3 Wipe Scene Change Detection

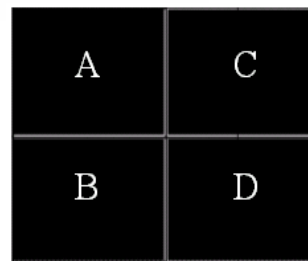
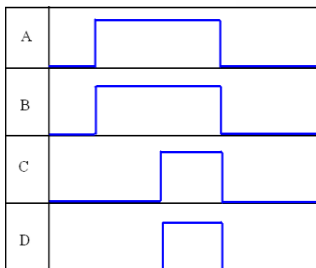
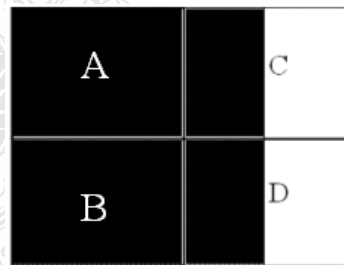
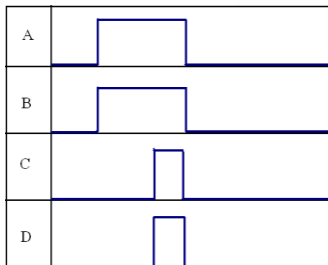
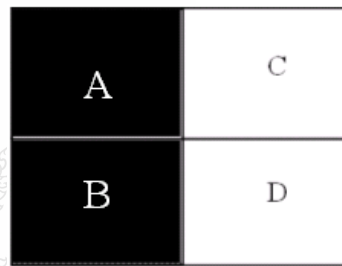
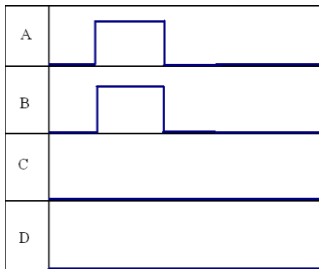
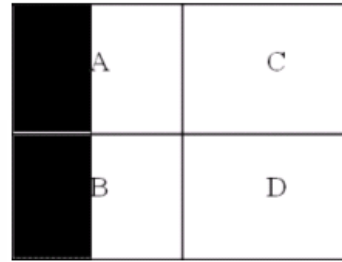
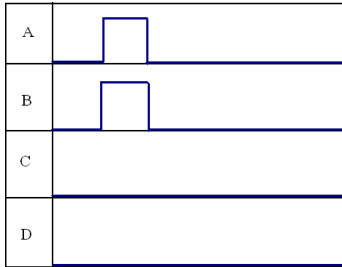
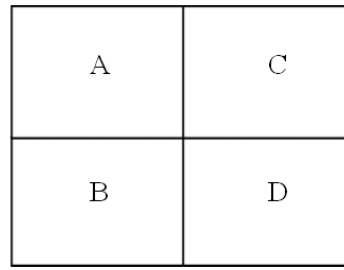
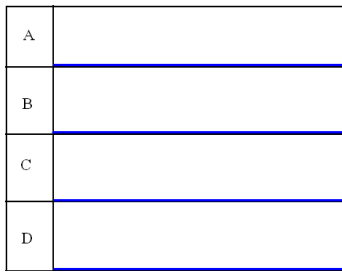
การหาจุดเปลี่ยนแบบ Wipe จำกระทำโดยการแบ่งภาพออกเป็น 4 ส่วน แล้วทำการศึกษาถึง Pattern การเปลี่ยนแปลงของ Segment ต่างๆ เมื่อเกิดการเปลี่ยนแปลงแบบ Wipe จากรูปที่ 2.8 แสดงตัวอย่างการเปลี่ยนแปลงของภาพเมื่อมีการเลื่อนเข้าของภาพจากซ้ายมาขวา



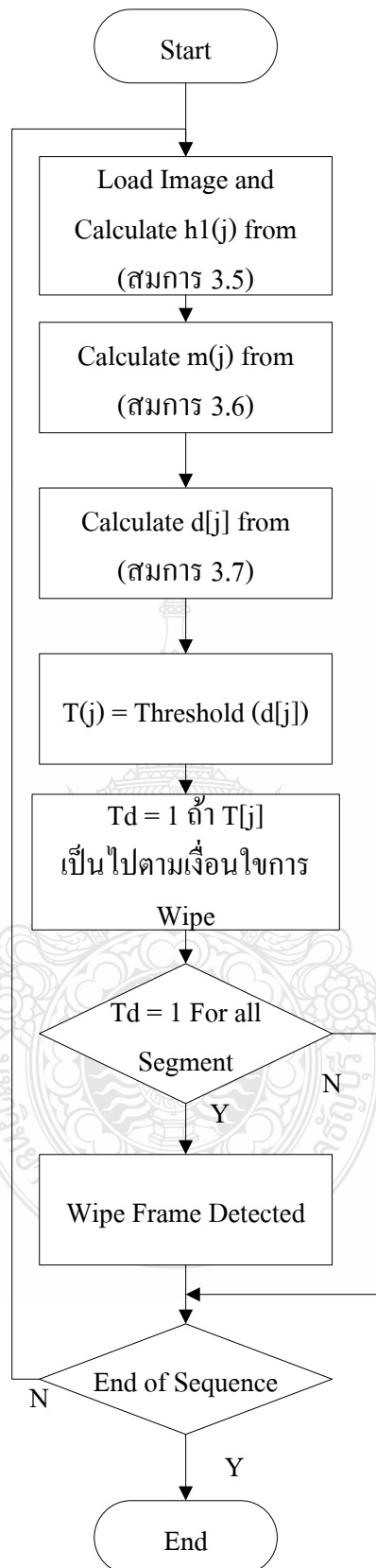
รูปที่ 3.10 การหาจุดเปลี่ยนของภาพแบบ Wipe

จากรูปที่ 3.10 เป็นตัวอย่างของจุดเปลี่ยนที่มีการเลื่อนเข้าจากซ้ายไปขวา จะเห็นได้ว่าเมื่อภาพมีการเลื่อนเข้ามาจะมีลักษณะของการเปลี่ยนแปลงของ Histogram ใน Segment ด้านซ้ายก่อนแล้วค่อยๆ เพิ่มขึ้น จนกระทั่งเลยจุดกึ่งกลางแล้วจึงมีการเปลี่ยนแปลงของ Segment ในด้านขวา และลักษณะการเปลี่ยนแปลงก็ได้ตั้งสมมติฐานว่ามีลักษณะเดียวกันกับ Dissolve เพราะมีการรวมกันของภาพ 2 นั้นเอง ดังนั้นเราจะใช้สมการของ Dissolve ตามสมการที่ 3.5, 3.6 และ 3.7 ในการคำนวณหาจุดเปลี่ยนแต่จะเปลี่ยนวิธีการตัดสินใจว่าเป็นจุดเปลี่ยนหรือไม่ โดยดูจาก Pattern การเคลื่อนของการเปลี่ยนแปลงของแต่ละ Segment นั้นเอง รูปที่ 3.11 เป็นตัวอย่าง Pattern การเปลี่ยนแปลงของแต่ละ Segment โดยที่ช่วงสัญญาณ 1 คือตำแหน่งมือเกิด Dissolve และ 0 คือช่วงที่ไม่มีการเปลี่ยนแปลง รูปที่ 3.12 แสดงการทำงานของการทำงานของการหาจุดเปลี่ยนของภาพแบบ Wipe ตามวิธีการที่นำเสนอ โดยที่ Pattern ของอัลกอริทึมที่นำเสนอ จะมีการเคลื่อนที่เข้าของภาพจาก ซ้าย ขวา บน ล่าง และจากทั้ง 4 มุมของภาพ





รูปที่ 3.11 กราฟเมื่อเกิดการเปลี่ยนภาพแบบ Wipe ตามลำดับ



รูปที่ 3.12 Flowchart การทำงานของการหาจุดเปลี่ยนของภาพแบบ Wipe

บทที่ 4

ผลการวิจัย

ในบทนี้จะกล่าวถึงรายละเอียดผลการงานวิจัยที่ได้นำเสนอหลักการไว้ในบทที่ 3 โดยผลการทดลองได้ทำการสรุปออกมาเป็นตารางแยกออกเป็น 4 วิธีหลักๆคือ Rapid Scene Change Detection, Fade Scene Detection, Dissolve Scene Change Detection และ Wipe Scene Change Detection

4.1 การหาจุดเปลี่ยนแบบทันทีทันใด

การทดสอบ Rapid Scene Change Detection จะใช้ข้อมูลของภาพต่อเนื่องซึ่งได้มาจากการตัดต่อวิดีโอโดยใช้โปรแกรมตัดต่อ และภาพจากวิดีโอภาพยนตร์จริงๆ จำนวนทั้งหมด 6 เรื่อง โดยที่แหล่งกำเนิดของข้อมูลเป็นวิดีโอที่มีคุณภาพ MPEG-1 หรือ VCD เพราะเป็นภาพที่มีคุณภาพที่ใช้งานกันอยู่โดยทั่วไป อีกทั้งยังทำให้สามารถวัดความแตกต่างของอัลกอริทึมได้ชัดเจนยิ่งขึ้น เพราะมี Noise อยู่ด้วยนั่นเอง ภาพที่ใช้และจำนวนจุดเปลี่ยนเป็นดังตารางที่ 4.1 และได้ประสิทธิภาพการทำงานของอัลกอริทึมตามตารางที่ 4.2 และ ตารางที่ 4.3 ตามลำดับ

ตารางที่ 4.1 จำนวนภาพและจุดเปลี่ยนแบบ Rapid ที่ใช้ในการทดสอบ

ชื่อภาพ	ขนาด	จำนวนเฟรม	จำนวนจุดเปลี่ยน
Video 1	320*240	907	31
Video 2	320*240	728	11
Video 3	320*240	725	17
Video 4	320*240	457	17
Video 5	240*180	529	12
Video 6	156*88	1225	33

ตารางที่ 4.2 การเปรียบเทียบผลการทดสอบประสิทธิภาพอัลกอริทึมของการเปลี่ยนภาพ
แบบ Rapid Scene change Detection

ชื่อภาพ	Absolute value of histogram		Chi-square value of histogram		แบบไม่ซ้อนทับ		แบบซ้อนทับ	
	Re	Pre	Re	Pre	Pre	Pre	Re	Pre
Video 1	100	93.8	100	100	93.8	93.8	100	93.8
Video 2	80.8	64.0	100	64.0	100	50.0	100	64
Video 3	93.5	100.0	93.5	93.5	93.5	93.5	93.5	93.5
Video 4	93.5	93.5	93.5	93.5	93.5	93.5	93.5	93.5
Video 5	100	100	100	100	100	100	100	100
Video 6	91.1	65.5	91.1	66.7	91.1	66.7	91.1	66.7
เฉลี่ย	93.15	86.13	96.35	86.28	96.35	83.95	96.35	86.28
เฉลี่ย Re, Pre	89.64		91.32		90.15		91.32	

Re = Recall มีหน่วยเป็น % , Pre = Precision มีหน่วยเป็น %

จากตารางที่ 4.2 เห็นได้ว่า อัลกอริทึมที่นำเสนอสามารถใช้ในการหาจุดเปลี่ยนได้ดี โดยที่อัลกอริทึมแบบที่มีการซ้อนทับกันของ Segment มีประสิทธิภาพสูงเท่ากันกับการใช้ อัลกอริทึม Chi-Square value of Histogram และมีประสิทธิภาพสูงกว่า การใช้อัลกอริทึม Absolute Value of Histogram ขณะที่หากใช้อัลกอริทึมแบบที่ไม่มีการซ้อนทับกันของ Segment จะใช้ผลลัพธ์ที่ดีกว่าการใช้อัลกอริทึม Absolute Value of Histogram แต่ดีกว่า อัลกอริทึม Chi-Square value of Histogram

ตารางที่ 4.3 การเปรียบเทียบผลการทดสอบประสิทธิภาพอัลกอริทึมในด้านความเร็วเฉลี่ย
ของการเปลี่ยนภาพแบบ Rapid

ชื่อภาพ	Time (ms)			
	Absolute value of histogram	Chi-square value of histogram	แบบไม่ซ้อนทับ	แบบซ้อนทับ
Video 1	0.09	0.09	0.12	0.13
Video 2	0.08	0.04	0.08	0.09
Video 3	0.07	0.08	0.1	0.1
Video 4	0.08	0.06	0.08	0.09
Video 5	0.05	0.09	0.1	0.11
Video 6	0.06	0.09	0.09	0.11
เฉลี่ย	0.072	0.075	0.095	10.5

ตารางที่ 4.3 แสดงการเปรียบเทียบเวลาที่ใช้ในการประมวลผลข้อมูลของอัลกอริทึมทั้ง 4 แบบ คือ Absolute value of histogram, Chi-square value of histogram, อัลกอริทึมแบบที่มีการซ้อนทับกันของ Segment และ อัลกอริทึมแบบที่ไม่มีการซ้อนทับกันของ Segment จะเห็นได้ว่าอัลกอริทึมที่นำเสนอใช้เวลาในการประมวลผลที่นานกว่า และ อัลกอริทึมแบบที่มีการซ้อนทับกันของ Segment ใช้เวลาในการประมวลผลของข้อมูลนานที่สุดเนื่องจากต้องมีการประมวลผลข้อมูล Histogram ทั้งหมด 4 ชุด แต่หากมองในเรื่องของความยากง่ายในการที่จะใช้อัลกอริทึมมาทำการผลิตเพื่อใช้งาน เป็น Hardware จริงๆแล้วจะพบว่า อัลกอริทึมที่นำเสนอนั้น สามารถนำไปผลิตได้ง่ายกว่าเนื่องจากไม่มีการใช้งานยกกำลังและหาร เป็นเพียงแค่การลบกับของข้อมูลใน Arrays จำนวน 4 ชุดเท่านั้นเอง ดังนั้นอัลกอริทึมนี้จึงเป็นแนวทางหนึ่งในการเลือกใช้งาน นอกจากนี้ยังมีประสิทธิภาพเทียบเท่า Chi-Square value of Histogram อีกด้วย

4.2 การหาจุดเปลี่ยนแบบ Fade

การทดสอบ Fade Scene Change Detection จะใช้ข้อมูลในการทดสอบทั้งหมด 2 ชุดโดยข้อมูลชุดแรกเป็นดังตารางที่ 4.4 มีวิดีโอทั้งหมด 7 ชุดข้อมูลโดยที่ทุกชุดเป็นข้อมูลที่ได้ได้มาจากข้อมูลวิดีโอต้นฉบับเป็น MPEG-2 หรือ DVD ดังนั้นภาพที่ได้จึงเป็นภาพที่มีคุณภาพดี มีปริมาณของ Noise ที่น้อย เพื่อทดสอบเพื่อทดสอบอัลกอริทึมที่นำเสนอเปรียบเทียบกับอัลกอริทึมที่นำเสนอโดย WAC Fernando [10] ภาพชุดที่ 2 ใช้สำหรับการทดสอบเพื่อเปรียบเทียบสมมติฐานที่ว่าในกรณีที่เกิด Noise ขึ้นหรือเมื่อมี Logo ติดอยู่ในภาพตลอดเวลาอัลกอริทึมที่นำเสนอโดย WAC Fernando ไม่สามารถทำงานได้ โดยภาพชุดที่ 2 มีทั้งหมด 3 ชุดข้อมูลประกอบไปด้วยภาพจาก DVD ที่มีจุดเปลี่ยนที่ชัดเจน 1 ชุดข้อมูล ภาพที่มี Noise นำมาจาก MPEG-1 หรือ VCD 1 ชุดข้อมูล และภาพที่มี Logo ติดในขณะที่มีการ Fade อีก 1 ชุดข้อมูล โดยการทดสอบแรกจะนับจำนวนเฟรมของการเปลี่ยนเป็นจุดเปลี่ยน 1 จุดด้วย ในขณะที่การทดสอบที่ 2 จะถือว่าหากมีการตรวจจับได้ในเฟรมที่เป็นจุดเปลี่ยนมากกว่า 4 เฟรมจะถือว่าถูกต้อง เพื่อต้องการทดสอบสมมติฐานที่สร้างขึ้นเท่านั้น ภาพที่ใช้ในการทดสอบอยู่ในตารางที่ 4.5

ตารางที่ 4.4 จำนวนภาพชุดที่ 1 และจุดเปลี่ยนแบบ Fade ที่ใช้ในการทดสอบ

ชื่อภาพ	ขนาด	จำนวนเฟรม	จำนวนจุดเปลี่ยน	
			Fade-in	Fade-out
Video 1	360*288	592	7	7
Video 2	360*288	912	7	7
Video 3	360*288	1083	7	7
Video 4	360*288	1130	9	9
Video 5	320*240	1845	11	11
Video 6	320*240	789	4	4
Video 7	320*240	1579	8	8

ตารางที่ 4.5 จำนวนภาพชุดที่ 2 และจุดเปลี่ยนแบบ Fade ที่ใช้ในการทดสอบ

ชื่อภาพ	ขนาด	จำนวนเฟรม	จำนวนจุดเปลี่ยน Fade-in	จำนวนจุดเปลี่ยน Fade-out
King and I 1 (from DVD)	352*288	2440	17	17
King and I 2 (from VCD)	352*288	2440	17	17
Castaway (Movie trailer)	320*240	1780	9	11

ผลการทดสอบแรกเพื่อเปรียบเทียบประสิทธิภาพของอัลกอริทึมและความเร็วของอัลกอริทึม
เป็นไปตามตารางที่ 4.6 และตารางที่ 4.7 ตามลำดับ

ตารางที่ 4.6 ผลการทดสอบประสิทธิภาพอัลกอริทึมของการเปลี่ยนภาพแบบ Fade

ชื่อภาพ	WAC Fernando		แบบไม่ซ้อนทับ		แบบซ้อนทับ	
	Re	Pre	Re	Pre	Re	Pre
Video 1	41.13	94.20	66.45	91.30	81.64	90.21
Video 2	46.77	100	69.35	100	77.95	97.97
Video 3	55.95	96.90	78.57	92.30	77.95	93.28
Video 4	46.12	97.27	80.17	96.37	82.73	93.87
Video 5	71.56	90.17	86.69	80.76	90.82	77.64
Video 6	69.73	77.94	52.63	72.72	60.52	70.76
Video 7	36.90	93.93	67.85	94.21	69.04	79.45
เฉลี่ย	52.59	92.91	71.67	89.67	77.43	86.17
เฉลี่ย Re, Pre	72.75		80.67		81.8	

Re = Recall มีหน่วยเป็น % , Pre = Precision มีหน่วยเป็น %

จากตารางผลการทดลองที่ 4.6 จะเห็นได้ว่าอัลกอริทึมที่นำเสนอสามารถตรวจจับจุดเปลี่ยนแบบ Fade ได้ดีทั้ง Fade-in และ Fade-Out โดยดูจากค่าเฉลี่ยของ Recall และ Precision โดยเทคนิคของ WAC Fernando ได้ค่าเฉลี่ยอยู่ที่ 72.75% ในขณะที่อัลกอริทึมที่นำเสนอมีค่าเฉลี่ยอยู่ที่ 80.67% สำหรับอัลกอริทึมแบบไม่มีการซ้อนทับกันของ Segment และ 81.80 % สำหรับอัลกอริทึมแบบมีการซ้อนทับกันของ Segment

ตารางที่ 4.7 การเปรียบเทียบผลการทดสอบประสิทธิภาพอัลกอริทึมในด้านความเร็วเฉลี่ยของการเปลี่ยนภาพแบบ Fade

ชื่อภาพ	Time (ms)		
	WAC Fernando	แบบไม่ซ้อนทับ	แบบซ้อนทับ
Video 1	0.07	0.07	0.08
Video 2	0.05	0.05	0.05
Video 3	0.05	0.06	0.06
Video 4	0.06	0.06	0.07
Video 5	0.10	0.12	0.12
Video 6	0.05	0.05	0.05
Video 7	0.09	0.11	0.12
เฉลี่ย	0.067	0.074	0.078

จากตารางผลการทดลองที่ 4.7 แสดงให้เห็นถึงความเร็วในการทำงานของอัลกอริทึมของทั้ง 3 วิธี จากการทดลองพบว่าอัลกอริทึมที่นำเสนอต้องใช้เวลาในการประมวลผลที่นานกว่าอัลกอริทึมของ WAC Fernando โดยที่ค่าเฉลี่ยเวลาที่ใช้ของ WAC Fernando อยู่ที่ 0.067ms ในขณะที่อัลกอริทึมที่นำเสนอมีค่าเฉลี่ยอยู่ที่ 0.074 ms สำหรับอัลกอริทึมแบบไม่มีการซ้อนทับกันของ Segment และ 0.078 ms สำหรับอัลกอริทึมแบบมีการซ้อนทับกันของ Segment

ตารางที่ 4.8 ผลการทำงานของอัลกอริทึมขณะที่มี Noise หรือ Logo บนภาพ

ชื่อภาพ	Fade-in นำเสนอ (WAC Fernando)		Fade-out นำเสนอ (WAC Fernando)		Fade นำเสนอ (WAC Fernando)	
	Re	Pre	Re	Pre	Re	Pre
	King and I 1	100 (100)	94.4 (94.4)	100 (100)	100 (100)	100 (100)
King and I 2	100 (0)	94.4 (0)	100 (0)	100 (0)	100 (0)	97.1 (0)
Castaway	100 (0)	100 (0)	100 (0)	91.7 (0)	100 (0)	95.2 (0)

Re = Recall มีหน่วยเป็น % , Pre = Precision มีหน่วยเป็น %

ตารางที่ 4.8 นำเสนอผลการทดสอบของอัลกอริทึมโดยการตรวจจับทั้งภาพโดยไม่มี Segment ของภาพ โดยภาพที่ใช้เป็นภาพที่เป็นภาพที่ Extract มาจากส่วนของภาพยนต์ที่มี Fade-in และ Fade-out จากการทดสอบพบว่า ในกรณีที่ภาพมีแค่ Fade โดยที่ไม่มี Noise หรือ Logo อยู่ อัลกอริทึมทั้งสองสามารถตรวจจับได้ผลลัพธ์ที่มีค่าเหมือนกัน เมื่อมี Noise หรือ Logo อัลกอริทึมของ WAC Fernando ไม่สามารถทำงานได้เลยเนื่องจากค่าความแตกต่างของสัญญาณสูงสุดต่ำสุดของ Histogram ไม่สามารถเป็น 0 ได้ทั้งจุดเริ่มต้นของการเกิด Fade-in และสิ้นสุดของการเกิด Fade-out

4.3 การหาจุดเปลี่ยนแบบ Dissolve

การทดสอบ Dissolve Scene Change Detection จะใช้ข้อมูลในการทดสอบทั้งหมด 6 ชุดโดยข้อมูลชุดแรกเป็นดังตารางที่ 4.9 มีวิดีโอทั้งหมด 7 ชุดข้อมูลโดยที่ทุกชุดเป็นข้อมูลที่ได้มาจากข้อมูลวิดีโอต้นฉบับเป็น MPEG-2 หรือ DVD ดังนั้นภาพที่ได้จึงเป็นภาพที่มีคุณภาพดี มีปริมาณของ Noise ที่น้อย เพื่อทดสอบเพื่อทดสอบอัลกอริทึมที่นำเสนอเปรียบเทียบกับอัลกอริทึมที่นำเสนอโดย WAC Fernando [16] โดยการกำหนด Threshold ที่ดีที่สุดเพียง Threshold เดียว

ตารางที่ 4.9 จำนวนจุดเปลี่ยนแบบ Dissolve ที่ใช้ในการทดสอบ

ชื่อภาพ	ขนาด	จำนวนเฟรม	จำนวนจุดเปลี่ยน
Video 1	360*288	1158	8
Video 2	320*240	1132	8
Video 3	320*240	696	6
Video 4	320*240	811	10
Video 5	320*240	1845	11
Video 6	320*240	1579	8



ตารางที่ 4.10 ผลการทดสอบประสิทธิภาพอัลกอริทึมของการเปลี่ยนภาพแบบ Dissolve

ชื่อภาพ	WAC Fernando		แบบไม่ซ้อนทับ		แบบซ้อนทับ	
	Re	Pre	Re	Pre	Re	Pre
Video 1	93.75	47.94	84.82	61.88	89.73	60.36
Video 2	11.86	23.07	77.34	69.69	76.77	64
Video 3	100	28.11	98.38	67.03	100	45.92
Video 4	63.92	85.89	62.1	84.47	73.51	76.30
Video 5	100	16.91	100	23.23	81.53	56.53
Video 6	62.63	13.13	61.57	22.11	69.47	60.97
เฉลี่ย	72.06	35.84	80.7	54.74	81.83	60.68
เฉลี่ย Re, Pre	53.95		67.72		71.26	

Re = Recall มีหน่วยเป็น % , Pre = Precision มีหน่วยเป็น %

จากตารางที่ 4.10 จะเห็นได้ว่าอัลกอริทึมที่นำเสนอสามารถทำงานได้ดีกว่าอัลกอริทึมที่นำเสนอโดย WAC Fernando ในสภาวะการใช้งานจริงเพราะว่าเราไม่สามารถที่จะกำหนด Threshold ที่แน่นอนสำหรับแต่ละวิดีโอได้ จากผลการทดลองพบว่าเมื่อกำหนด threshold ที่ดีที่สุด อัลกอริทึมที่นำเสนอโดย WAC Fernando ได้ค่าเฉลี่ยอยู่ที่ 53.95% ในขณะที่อัลกอริทึมที่นำเสนอมีค่าเฉลี่ยอยู่ที่ 67.72 % สำหรับอัลกอริทึมแบบไม่มีการซ้อนทับกันของ Segment และ 71.26 % สำหรับอัลกอริทึมแบบมีการซ้อนทับกันของ Segment

ตารางที่ 4.11 การเปรียบเทียบผลการทดสอบประสิทธิภาพอัลกอริทึมในด้านความเร็วเฉลี่ย
ของการเปลี่ยนภาพแบบ Dissolve

ชื่อภาพ	Time (ms)		
	WAC Fernando	แบบไม่ซ้อนทับ	แบบซ้อนทับ
Video 1	0.08	0.09	0.09
Video 2	0.08	0.09	0.09
Video 3	0.08	0.08	0.08
Video 4	0.02	0.03	0.03
Video 5	0.10	0.11	0.11
Video 6	0.09	0.09	0.09
เฉลี่ย	0.075	0.082	0.082

จากตารางผลการทดลองที่ 4.11 แสดงให้เห็นถึงความเร็วในการทำงานของอัลกอริทึมของทั้ง 3 วิธี จากการทดลองพบว่าอัลกอริทึมที่นำเสนอต้องใช้เวลาในการประมวลผลที่นานกว่าอัลกอริทึมของ WAC Fernando โดยที่ค่าเฉลี่ยเวลาที่ใช้ของ WAC Fernando อยู่ที่ 0.075ms ในขณะที่อัลกอริทึมที่นำเสนอมีค่าเฉลี่ยอยู่ที่ 0.082 ms สำหรับอัลกอริทึมแบบไม่มีการซ้อนทับกันของ Segment และ 0.082 ms สำหรับอัลกอริทึมแบบมีการซ้อนทับกันของ Segment

4.4 การหาจุดเปลี่ยนแบบ Wipe

การทดสอบ Wipe Scene Change Detection จะใช้ข้อมูลในการทดสอบทั้งหมด 6 ชุดโดยข้อมูลชุดแรกเป็นดังตารางที่ 4.12 มีวิดีโอทั้งหมด 4 ชุดข้อมูลโดยที่ทุกชุดเป็นข้อมูลที่ได้ได้มาจากข้อมูลวิดีโอต้นฉบับเป็น MPEG-2 หรือ DVD ดังนั้นภาพที่ได้จึงเป็นภาพที่มีคุณภาพดี มีปริมาณของ Noise ที่น้อย โดยข้อมูลของจุดเปลี่ยนจะเป็นการเลื่อนเข้ามาจากด้านข้างของภาพ 8 ทิศทางคือซ้าย ขวา บน ล่าง และ ทั้ง 4 มุมของภาพ การทดสอบอัลกอริทึมจะเปรียบเทียบกับพื้นฐานการหาจุดเปลี่ยนแบบ Chi-Square[6] ซึ่งจะใช้ค่า Threshold ที่ดีที่สุด

ตารางที่ 4.12 จำนวนจุดเปลี่ยนแบบ Wipe ที่อัลกอริทึมสามารถจับได้

ชื่อภาพ	ขนาด	จำนวนเฟรม	จำนวนจุดเปลี่ยน
Video 1	360*288	853	5
Video 2	360*288	822	4
Video 3	320*240	1579	7
Video 4	360*240	467	4

ตารางที่ 4.13 ผลการทดสอบประสิทธิภาพอัลกอริทึมของการเปลี่ยนภาพแบบ Wipe

ชื่อภาพ	Chi-Square		แบบไม่ซ้อนทับ		แบบซ้อนทับ	
	Re	Pre	Re	Pre	Re	Pre
Video 1	21.53	35.44	56.92	60.65	75.38	62.02
Video 2	10.43	63.15	93.04	87.70	85.21	85.96
Video 3	9.316	13.76	25.46	13.35	36.64	14.78
Video 4	73.11	64.76	65.59	88.40	66.66	89.85
เฉลี่ย	28.60	44.28	60.25	62.53	65.97	63.15
เฉลี่ย Re, Pre	36.44		61.39		64.56	

Re = Recall มีหน่วยเป็น % , Pre = Precision มีหน่วยเป็น %

จากตารางที่ 4.13 จะเห็นได้ว่าอัลกอริทึมที่นำเสนอสามารถทำงานได้ดีกว่าอัลกอริทึมแบบ Chi-Square จากผลการทดลองพบว่าเมื่อกำหนด threshold ที่ดีที่สุด อัลกอริทึมที่แบบ Chi-Square ได้ค่าเฉลี่ยอยู่ที่ 36.44% ในขณะที่อัลกอริทึมที่นำเสนอมีค่าเฉลี่ยอยู่ที่ 61.39 % สำหรับอัลกอริทึมแบบไม่มีการซ้อนทับกันของ Segment และ 64.56 % สำหรับอัลกอริทึมแบบมีการซ้อนทับกันของ Segment

ตารางที่ 4.14 การเปรียบเทียบผลการทดสอบประสิทธิภาพอัลกอริทึมในด้านความเร็วเฉลี่ยของการเปลี่ยนภาพแบบ Wipe

ชื่อภาพ	Time (ms)		
	Chi-Square	แบบไม่ซ้อนทับ	แบบซ้อนทับ
Video 1	0.09	0.09	0.1
Video 2	0.08	0.08	0.08
Video 3	0.07	0.08	0.08
Video 4	0.04	0.05	0.06
เฉลี่ย	0.07	0.075	0.08

จากตารางผลการทดลองที่ 4.14 แสดงให้เห็นถึงความเร็วในการทำงานของอัลกอริทึมของทั้ง 3 วิธี จากการทดลองพบว่าอัลกอริทึมที่นำเสนอต้องใช้เวลาในการประมวลผลที่นานกว่าอัลกอริทึมแบบ Chi-Square โดยที่ค่าเฉลี่ยเวลาที่ใช้ของอัลกอริทึมแบบ Chi-Square อยู่ที่ 0.07 ms ในขณะที่อัลกอริทึมที่นำเสนอมีค่าเฉลี่ยอยู่ที่ 0.075 ms สำหรับอัลกอริทึมแบบไม่มีการซ้อนทับกันของ Segment และ 0.08 ms สำหรับอัลกอริทึมแบบมีการซ้อนทับกันของ Segment

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

จากผลการทดสอบอัลกอริทึมที่นำเสนอจะเห็นได้ว่าสามารถที่จะตรวจจับหาจุดเปลี่ยนของภาพวิดีโอชนิดต่างๆได้ และทำงานได้ผลเป็นที่น่าพอใจเมื่อเปรียบเทียบกับวิธีการที่นำเสนอ และใช้กันโดยทั่วไปประกอบกับการทำงานที่น้อยเมื่อเทียบกับอีกหลายๆวิธีการ เพราะทำงานกับภาพต้นแบบที่เป็น Gray Scale นอกจากนี้การแบ่งภาพก่อนที่จะนำมาใช้ในการค้นหาจุดเปลี่ยน ยังช่วยให้การแยกแยะจุดเปลี่ยนเป็นไปได้อย่างดีอีกด้วย

5.1 สรุปผลที่ได้จากการวิจัย

5.1.1 Rapid Scene Change Detection

จากการทดลองพบว่าอัลกอริทึมที่นำเสนอสามารถทำงานได้ดีกว่า Absolute value of Histogram แต่ใช้เวลาในการประมวลผลที่มากกว่าเนื่องจากต้องประมวลผลข้อมูล 4 ชุดแทนที่จะเป็นแค่ชุดเดียว ในกรณีที่ใช้การซ้อนทับกับของ Segment สามารถให้ผลลัพธ์ความถูกต้องสูงเท่ากับ Chi-Square Value of Histogram แต่ใช้เวลาในการประมวลผลมากกว่า แต่หากว่าดูให้ดีจะพบว่าอัลกอริทึมที่นำเสนอใช้แค่การลบกันของข้อมูลเท่านั้นเอง ดังนั้นจึงเป็นทางเลือกหนึ่งในการพัฒนาอัลกอริทึมที่สามารถลดความซับซ้อนในการออกแบบและประมวลผลในทาง Hardware ได้เป็นอย่างดี

5.1.2 Fade Scene Change Detection

การทดลองแบ่งออกเป็นสองส่วนคือทดสอบประสิทธิภาพการทำงานจริงของอัลกอริทึมและทดสอบประสิทธิภาพของอัลกอริทึมที่นำเสนอในกรณีที่มีสัญญาณรบกวนหรือมีสัญญาณต่างๆเกิดขึ้นตลอดทั้งวิดีโอ การทดลองแรกพบว่าอัลกอริทึมที่นำเสนอสามารถทำงานได้ดีกว่าอัลกอริทึมที่นำมาเปรียบเทียบได้อย่างชัดเจนทั้งสองวิธีแต่ใช้เวลาเฉลี่ยในการทำงานมากกว่าเล็กน้อยเนื่องจากต้องประมวลผลข้อมูลมากขึ้นนั่นเอง สำหรับการทดลองที่สองพบว่าอัลกอริทึมที่นำเสนอมีความคงทนต่อสัญญาณรบกวนสูงในขณะที่อัลกอริทึมที่นำมาเปรียบเทียบไม่สามารถทำงานได้เลย ดังนำเสนอในกาประชุมวิชาการนานาชาติ [14]

5.1.3 Dissolve Scene Change Detection

จากการทดลองพบว่าอัลกอริทึมที่นำเสนอสามารถทำงานได้ดีกว่าอัลกอริทึมที่นำมาเปรียบเทียบเนื่องจากในสภาพการใช้งานจริงเราไม่สามารถทราบได้ถึงอัตราการเปลี่ยนแปลงที่แน่นอนของวิดีโอได้ดังนั้นจึงไม่สามารถที่จะกำหนด Threshold ที่แน่นอนในแต่ละช่วง การ

ทดลองนี้เราจึงใช้ Threshold ที่ดีที่สุดในการทดลองนี้ อย่างไรก็ตามยังพบว่าต้องใช้เวลาในการประมวลผลมากกว่าเล็กน้อยจากจำนวนข้อมูลที่ต้องประมวลผลเพิ่มมากขึ้น

5.1.4 Wipe Scene Change Detection

จากการทดลองพบว่าเมื่อทำการเปรียบเทียบผลการทดลองกับวิธีการพื้นฐาน อัลกอริทึมที่นำเสนอสามารถทำงานได้ดีกว่าอย่างชัดเจนเนื่องจาใช้การตรวจจับ Pattern การเปลี่ยนแปลงของข้อมูลแทนที่จะใช้ค่าตรงๆจากการคำนวณ สาเหตุที่ทำให้ผลลัพธ์ออกมาค่อนข้างแตกต่างกันอย่างชัดเจนเพราะว่าข้อมูลที่ใช้ในการทดสอบมีช่วงของการเปลี่ยนแปลงการเคลื่อนไหวของภาพซ้ำทำให้เกิดการเปลี่ยนแปลงระหว่ภาพที่น้อย ดังนั้นจึงทำให้ตรวจจับทำได้ยากขึ้น โดยเฉพาะการใช้ Threshold แต่อัลกอริทึมยังต้องใช้เวลาในการประมวลผลที่มากกว่าเล็กน้อย

5.1.5 สรุป

จากการวิจัยพบว่าสามารถใช้ Image histogram ในการประมวลผลเพื่อตรวจจับการเปลี่ยนแปลงของ Scene ได้โดยที่อัลกอริทึมที่นำเสนอสามารถทำงานและให้ผลลัพธ์ที่ดีแต่ต้องใช้เวลาในการประมวลผลที่มากขึ้น

5.2 ข้อเสนอแนะและแนวทางในการพัฒนา

การใช้ Image histogram ในการหาจุดเปลี่ยนมีข้อดีที่ชัดเจนคือความเร็วในการประมวลผล ข้อมูลมูลเนื่องจากจำนวนข้อมูลที่น้อยเพียงแค่ 256 ระดับข้อมูลเท่านั้น แต่ผลของการทดลองพบว่ามันมีประสิทธิภาพค่อนข้างสูงในการหาจุดเปลี่ยนที่เป็น Fade และ Rapid แต่ยังได้ผลที่ไม่ค่อยดีนักสำหรับ Wipe และ Dissolve ดังนั้น หากสามารถใช้ข้อมูลทางสถิติมาทำงานร่วมกับอัลกอริทึมที่นำเสนอ อาจทำให้ประสิทธิภาพของอัลกอริทึมดีขึ้น โดยที่ไม่เพิ่มปริมาณการคำนวณมากนัก เช่น การใช้ ค่าเฉลี่ย ส่วนเบี่ยงเบนมาตรฐาน ของภาพหรือ Histogram เป็นต้น

เอกสารอ้างอิง

- [1] MPEG Requirements Group, “MPEG-7: Context and Objectives International Organisation for Standardisation, July 1997.
- [2] MPEG Requirements Group, “Second Draft of MPEG-7 Requirements, International Organisation for Standardisation, July 1997.
- [3] MPEG Requirements Group, “MPEG-7 Applications document”, International Organisation for Standardisation, July 1997.
- [4] MPEG Requirements Group, “Report on MPEG-7 Seminar in Bristol”, International Organisation for Standardisation, April 1997.
- [5] MPEG Requirements Group, “MPEG-7: Overview”, International Organisation for Standardisation, Palma de Mallorca, October 2004
- [6] M. Ralph Ford. Metrics for Scene Change Detection in Digital Video Sequences, IEEE, 1997.
- [7] Qi, Y., Hauptmann, A., and Liu, T.: “Supervised classification for video shot segmentation”. Proc. IEEE Conf. on Multimedia Expo (ICME),2003, Vol. 2, pp. 689–692
- [8] “TREC Video retrieval evaluation”, 2005 [Online]. Available: <http://www.nlpir.nist.gov/projects/trecvid/>
- [9] L. Gao, J. Jiang, J. Liang, S. Wang, S. Yang and Y. Qin, “PCA-based approach for video scene change detection on compressed video”, ELECTRONICS LETTERS 23rd November 2006 Vol. 42 No. 24
- [10]W.A.C.Fernando, C.N.Canagarajah and D.R.Bull, *Automatic Detection Of Fade-in And Fade-out In Video Sequences*, IEEE International Symposium on Circuit and Systems, Page Number-789, 1999
- [11]W.A.C.Fernando, C.N.Canagarajah and D.R.Bull, *Automatic Detection Of Fade-in And Fade-out In Video Sequences Using Histograms*, IEEE International Symposium on Circuit and Systems, May, 2000, pp 28-31.
- [12]W.A.C.Fernando, C.N.Canagarajah and D.R.Bull, *A UNIFIED APPROACH TO SCENE CHANG DETECTION IN UNCOMPRESSED AND COMPRESSED VIDEO*, IEEE Transactions on Consumer Electronics, V46, No 3, Aug 2000, pp 769-779.
- [13]W.A.C.Fernando, C.N.Canagarajah and D.R.Bull, *Wipe Scene Change Detection in Video Sequences*, in Proc, IEEE, *ICIP*, 1999.b

- [14]W.Kompreyarat, J.Srinonchat, “Fade Scene Change Detection using Modify Luminance Histogram Bin”, ICCSIT, Singapore August 29 - September 2, 2008
- [15]W.Kompreyarat, J.Srinonchat,”Dissolve Scene Change Detection using Modify Histogram”, JCSSE2008, 7-9 May 2008, Kanchanaburi, Thailand
- [16]W.A.C Fernando, CN Canagarajah, D.R.Bull, “Fade and dissolve detection in uncompressed and compressed video sequence”. International Conference on Image Processing, 24-28 October 1999. **3**: p. 299 -303.



ภาคผนวก
ผลงานตีพิมพ์เผยแพร่



International Conference on

Computer Science and Information Technology



Singapore August 29 – September 2, 2008

IEEE
computer
society

 **IEEE**

Editors

Jianhong Zhou, Yi Xie, Sheng Chai, Desheng Wen

ICCSIT 2008

Organizing Committee

Honorary Chairs

Richard M. Voyles, USA
Denver, Colorado University

Nazir Ahmad Zafar, Pakistan
Faculty of Information Technology, University of Central Punjab

Poramate Manoonpong, Germany
Bernstein Center for Computational Neuroscience, University of Gottingen

Organizing Chairs

Jianhong Zhou, Singapore
Nanyang Technological University

Sheng Chai, China
China Computer Federation

S.R. Bhadra Chaudhuri, India
Department of Electronics and Tele-communication Engineering, Bengal Engineering & Science University,
West Bengal

Conference Steering Committee

Tawiwat Veeraklaew, Thailand
Mechanical Engineering Department, Chulachomklao Royal Military Academy (CRMA)

Safaa S. Mahmoud, Egypt
E-learning Center, Ain Shams University

Program Committee Chairs

Qiang Li, Japan
School of Science and Technology, Meiji University

Tahseen A. Jilani, Pakistan
University of Karachi

Publication Chair

Desheng Wen, China
China Computer Federation

Publicity Chairs

Yang Cheng, China
Sichuan University, College of Computer Science

Sung Ho Ha, Korea
Kyungpook National University

Jakkree Srinonchat, Thailand
Rajamangala University of Technology Thanyaburi, Phathumtani

Local Organization Committee Chair

Yi Xie, Singapore
Nanyang Technological University, School of Electrical & Electronic Engineering

Committee Members

Daya Gupta, Delhi University, IN
J. Srinonchat, Rajamangala University of Technology Thanyaburi, TH
Alessandro Cincotti, Japan Advanced Institute of Science and Technology, JP
Wai Phyo Aung, Mandalay Technological University, MM
Pattanapol Wichai, University of Otago, NZ
Anupam Shukla, Indian Institute of Information Technology, IN
Chao Chen-Yeng, Jinwen University of Science and Technology, China
Amir Masoud Rahmani, Islamiz Azad University, IR
Luiz Fernando Capretz, University of Western Ontario, CA
Najla Algadi, Ahmed Ali, Sudan University of Science and Technology, SD
Ion Smeucanu, Academy of Economic Studies Bucharest, RO
T. Klause, German Association for Information Technology, Telecommunications and New Media, GE
Zhihong Xiao, Zhejiang Wanli University, CN
H.M. Abdul-Kader, Menufiya University, EG
A. Jangwanitlert, King Mongkut's Institute of Technology Ladkrabang, TH
V. Saravanan, Karunya University, Tamil Nadu, IN
Lagaros Nikos D., National Technical University of Athens, GR
Gao Xiao-Zhi, Helsinki University of Technology, FI
Prasad Girijesh, University of Ulster, UK
Ahmed Wasif Reza, Multimedia University, Malaysia

International Conference on Computer Science and Information Technology 2008

ICCSIT 2008

Table of Contents

Preface.....	xix
Organizing Committee.....	xxi
Committee Members.....	xxiii

Session 1

Application of Digital Watermark and Mobile Agent in Copyright Protection System	3
<i>Liu Quan and Liu Hong</i>	
Multi-agent Architecture for Collaborative CAD System	7
<i>Xue Ligong, Zhou Zude, and Liu Quan</i>	
Scheduling Algorithm Based on Task Priority in Heterogeneous Computing Environment	12
<i>Yu Zhenxia, Meng Fang, and Shangming Sheng</i>	
A Certificate-Based Signcryption Scheme	17
<i>Ming Luo, Yingyou Wen, and Hong Zhao</i>	
A New Fuzzy Support Vector Machine Method for Named Entity Recognition	24
<i>Alireza Mansouri, Lilly Suriani Affendy, and Ali Mamat</i>	
The Design of the Secure Electronic Payment System Based on the SET Protocol	29
<i>Yabo Li</i>	
Hedged Predictions for Traditional Chinese Chronic Gastritis Diagnosis with Confidence Machine.....	34
<i>Huazhen Wang, Chengde Lin, Fan Yang, and Xueqin Hu</i>	
Fade Scene Change Detection using Luminance Histogram Bin	39
<i>W. Kompreyarat and J. Srinonchat</i>	

Fade Scene Change Detection using Luminance Histogram Bin

W.Kompreyarat and J.Srinonchat

Author Affiliation(s) Department of Electronic and Telecommunication Engineering
Faculty of Engineering, Rajamangala University of Technology Thanyaburi
Klong 6, Thanyaburi, Patumtani 12110, Thailand
weera.k@en.rmutt.ac.th, jakkree.s@en.rmutt.ac.th

Abstract

Partitioning video sequence is very important for video applications such as video indexing, retrieval or editing. A lot of techniques have been presented based on information such as structures of the image, image histogram and intensity statistics etc. The histogram-based technique is generally used due to computation complexity when compare with other techniques is less, and most of image processing applications, image histogram have been already computed. This paper presents a new technique to detect the fade scene change from the video scene that contain noise and/or contain Logo and compared it with other algorithm that using histogram. The result showed that the proposed algorithm is much better.

1. Introduction

Due to the successive of histogram base method in the use of rapidly scene change detection, the section has proposed the enhance algorithm of using histogram to detection the fade scene change detection. Many fade in and fade out algorithms have been proposed such as Campisi, P.; Neri, A.; Sorgi, L [1] use the correlation techniques to detect the fade. O. Fatemi, S.Zhang and S. Panchanathan [2] introduced the use of optical flow motion based fade scene change detection. W.A.C. Fernando, C.N. Canagarajah and D.R. Bull [3-6] has showed an algorithm to detect the fade scene change using statistical feature of the images for uncompressed domain and they also using DC-estimator to detect the fade for compression domain. According to the use of histogram to detect fade in and fade out, very few researches have been done. Only W.A.C. Fernando, C.N. Canagarajah and D.R. Bull [7] have proposed a Fade in and Fade out detection system using luminance histograms. This technique is known here as "Different Histogram Bins (DHB)". However, DHB has some problem in dealing with noisy images

sequence and the images sequence that contain logo, which is normally appeared within the video scene. This problem will be discussed later this section, and the algorithm to overcome the problem is presented and compared with DHB algorithm.

2. Fade-in and fade-out

Fade in is a gradually scene change that the image gradually appear into background image, while fade out the image is gradually disappear and finally become background. Figure 1 and Figure 2 show the Fade the video scene during fade in and fade out respectively.



Figure 1 Illustrate fade in transition



Figure 2 Illustrate fade out transition

Technically, fade in is the proportion of the picture that changes from 0% to 100% frame by frame gradually, while fade out is also the proportion of the picture that changes from 100% to 0% gradually.

W.A.C. Fernando, C.N. Canagarajah and D.R. Bull [8, 12] has given the mathematical model for fade in and fade out as in equation 1 and 2 respectively.

$$S_{f(i,j)} = \begin{cases} f_{f(i,j)} & 0 \leq n < L_1 \\ \left[1 - \left(\frac{n-L_1}{F}\right)\right] \cdot C + \left(\frac{n-L_1}{F}\right) \cdot g_{f(i,j)} & L_1 \leq n \leq (L_1 + F) \\ g_{f(i,j)} & (L_1 + F) < n \leq L_2 \end{cases} \quad (1)$$

$$S_n(i,j) = \begin{cases} f_n(i,j) & 0 \leq n < L_1 \\ \left[1 - \left(\frac{n-L_1}{F}\right)\right] \cdot f_n(i,j) + \left(\frac{n-L_1}{F}\right) \cdot C & L_1 \leq n \leq (L_1 + F) \\ g_n(i,j) & (L_1 + F) < n \leq L_2 \end{cases} \quad (2)$$

Where, C is the video signal level

$S_n(i,j)$ is the resultant video signal

$f_n(i,j)$ is picture A

$g_n(i,j)$ is picture B

L_1 is length of video sequence A

F is length of fading sequence

L_2 is length of total video sequence

According to this mathematical model, when picture A is a solid value, it is known as fade in, when picture B is a solid value, it is fade out and if both of them are picture it is known as dissolve which can be rewrite as the histogram equation in equation 3 and 4 respectively

$$H_{S_n}(x) = \begin{cases} H_{f_n}(x) & 0 \leq n < L_1 \\ \left[1 - \left(\frac{n-L_1}{F}\right)\right] \cdot H_{f_n}(x) + \left(\frac{n-L_1}{F}\right) \cdot H_{g_n}(x) & L_1 \leq n \leq (L_1 + F) \\ H_{g_n}(x) & (L_1 + F) < n \leq L_2 \end{cases} \quad (3)$$

$$H_{S_n}(x) = \begin{cases} H_{f_n}(x) & 0 \leq n < L_1 \\ \left[1 - \left(\frac{n-L_1}{F}\right)\right] \cdot H_{f_n}(x) + \left(\frac{n-L_1}{F}\right) \cdot C & L_1 \leq n \leq (L_1 + F) \\ H_{g_n}(x) & (L_1 + F) < n \leq L_2 \end{cases} \quad (4)$$

Where, $H_{S_n}(x)$, $H_{f_n}(x)$ and $H_{g_n}(x)$ are the histogram of $S_n(i,j)$, $f_n(i,j)$ and $g_n(i,j)$ respectively.

This is clear from the equation that during fading, the histogram is scaled and shift along the x-axis of the histogram. The number of different between maximum and minimum $x(n)$ of histogram is decreasing during fade out operation, and it is increasing during fade in operation. In case of fade in, the histogram different along x-axis will start from zero and increasing until the picture fully appears. Figure 3, Figure 4 and Figure 5 show the examples of histogram at the starting point of fade in, during fade in and at the end of fade in respectively. Figure 6 shows an ideal $x(n)$ during fade in and fade out operation.

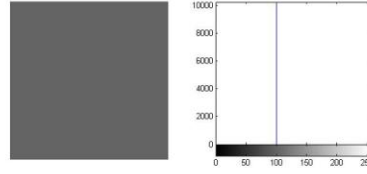


Figure 3 Image and its histogram at the start of fade in

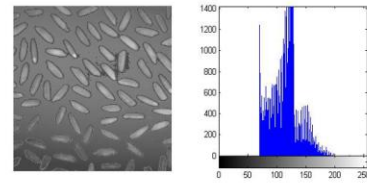


Figure 4 Image and its histogram during fade in

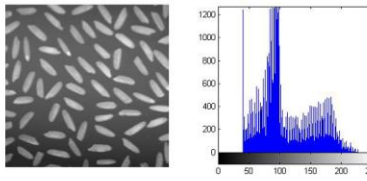


Figure 5 Image and its histogram at the end of fade in

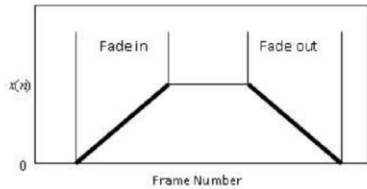


Figure 6 Ideal $x(n)$ during fade in and fade out

3. Fade-in and fade-out using image histogram Bin

W.A.C. Fernando, C.N. Canagarajah and D.R. Bull have suggested the DHB. The DHB suggested dividing

the image into a number of regions to improve the reliability of the algorithm. If choose the region is too small, the reliability will be decreased. On the other hand, if the region is too large, will increase the computation cost. So, DHB has suggested using the fixed four regions to optimize. The four histograms the produce from four sub images is used to detect the fade scene change. Four segments of the image and its parameters that need to calculate is showed in Figure 7.

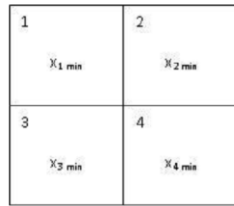


Figure 7 Four sub-regions of the image and its parameter

The weakness of DHB algorithm is the fade in is only started when $x(n) = 0$, which is rarely happened in real video sequence due to the noise which is not able to remove completely from the sequence. It is also, most of the time; the video sequence contains the logo of the company, or the logo of the broadcasting station. This will make DHB fail to detect any of them. The example of image that contains noise or more than one intensity level, logo and their histogram is show in Figure 8 and Figure 9 respectively.

It is clear from the figure that the DHB algorithm will not be able to work on both image sequences. Especially in figure 8, there is no different when observe with human eye when compare with one solid intensity. In order to use DHB algorithm for these sequence, some preprocessing need to be used such as filters, however, this will use a lot more computation cost to process.

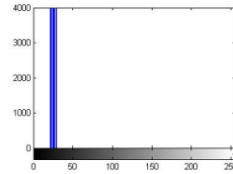


Figure 8 Image and its histogram that contain more than one intensity level

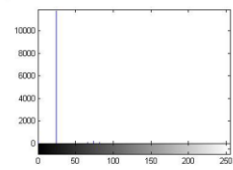


Figure 9 Image and its histogram that contain logo

4. Propose Algorithm

It is clear that the original image sequences need to be preprocessed in order to use DHB for scene change detection. The new technique to deal with this situation is presented in the section. Basically the technique will use the number of histogram bins instead of using maximum and minimum value of the images. This will make the algorithm more efficient, especially working under offline process, because the noise level and the logo can be easily eliminated, without the modification of original images sequences.

The Propose algorithm namely Counting Histogram Bin (CHB) started with the calculation of histograms for all images $H(i, n)$, where I is frame number and n is the histogram bins. Calculate the number of histograms

bin by counting the number of that contains histogram within the frame. This process is done using equation 5.

$$C(i) = \sum_{n=1}^{256} G(i,n) \text{ Where } G(i,n) = \begin{cases} 1 & H(i,n) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Where $i = 1..M$ and M is number of frame in the scene.

A non-linear quantisation is employed to push down the level of lowest point of noises or lowest signal within the whole sequence to zero. The quantisation process is showed in equation 6 and 7.

$$C'(i) = C(i) - \min(C) \quad (6)$$

$$Q(i) = \text{round}(C'(i) * B / 256) \quad (7)$$

Where B = number of quantisation bins, and $Q(i)$ is the quantisation data.

The $Q(i)$ from equation 7 is the quantisation data the all the noise and logo are push down into zero level but it is going to be very difficult to detect the scene change from this data due to the change of the histogram level might be very low, while the quantisation step too high. In order to overcome this, the quantisation signal is pushed back into the original scale of the histogram. The process is pushed every signal back to it original except the lowest level of the signal, which is supposed to be noise or logo. The process is done using equation 8.

$$F(i) = \frac{C(i) * Q(i)}{B} \quad (8)$$

Where, $F(i)$ is the original level of data excluding noise and logo.

$F(i)$ is presented the ideal histogram that will be able to detect the fade scene change. The Ideal Fade in and fade out is showed in Figure 10.

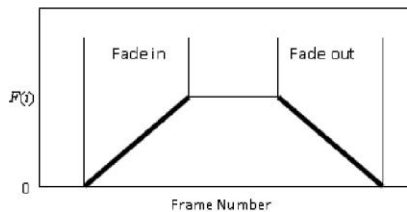


Figure 10 Ideal $F(i)$ during fade in and fade out

There are two cases of fade will be happened, 1. Fade in $F(i)$ will be increasing from frame to frame started from zero, 2 fade out $F(i)$ will be decreasing from frame to frame until it is become zero. The $F(i)$ changes during fade in and fade out is showed in equation 9 and 10 respectively.

$$Y(i) = \begin{cases} True & F(i) < F(i+1) < F(i+2) < F(i+3) \\ False & \text{otherwise} \end{cases} \quad (9)$$

$$Z(i) = \begin{cases} True & F(i) > F(i+1) > F(i+2) > F(i+3) \\ False & \text{otherwise} \end{cases} \quad (10)$$

$Y(i) = 1$ continuously during fade in and $Z(i) = 1$ continuously during fade out.

5. Result and discussion

Scene change detection using DHB and CHB have been implemented and compared each other in three cases, high quality image sequence without noise and logo, image sequence contains noise and image sequence contains logo. Three different images sequence that is used are show in Table 1.

Scene name	Size of image	Number of frame	Number of Fade in	Number of Fade out
King and I 1 (from DVD)	352*288	2440	17	17
King and I 2 (from VCD)	352*288	2440	17	17
Castaway (Movie trailer)	320*240	1780	9	11

Table 1 Scenes for testing fade scene change detection algorithm

The king and I 1 sequences is a very clear scene extracted from MPEG-2 (DVD) and reduced the size to 352*288. The king and I 2 is the same as the king and I 1 image sequence, its extract from MPEG-1 (VCD) image sequence. This image sequence contains noises due to the MPEG-1compression algorithm. Finally, the castaway is a movie trailer from Twenty Century Fox, its contain a logo of Twenty Century Fox at the bottom right of the whole image sequence.

In order to evaluate the result of fade detection, the recall-precision evaluation approach has been employed. The example of the output of the detection algorithm is showed in Figure 11. Finally, the comparison of both DHB and CHS are showed in Table 2.

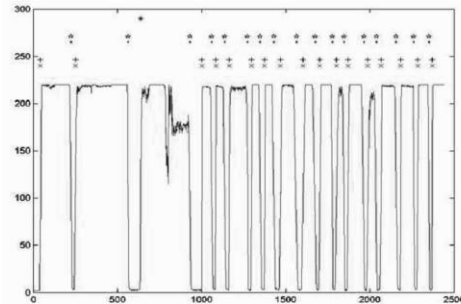


Figure 11 An example of CHB fade-in and fade-out detection applied to The King and I 2 sequences.

Where x and + are actual fade-in and fade-in detected respectively from proposed algorithm, • and ✕ are actual fade-out and fade-out detected respectively from proposed algorithm, and * is the fade-in and fade-out the algorithm failed to detect

Scene	CHB Fade-in (DHB Fade-in)		CHB Fade-out (DHB Fade-out)		CHB Fade-in and Fade-out (DHB Fade-in and Fade out)	
	Re	Pre	Re	Pre	Re	Pre
King and I 1	100 (100)	94.4 (94.4)	100 (100)	100 (100)	100 (100)	97.1 (97.1)
King and I 2	100 (0)	94.4 (0)	100 (0)	100 (0)	100 (0)	97.1 (0)
Castaway	100 (0)	100 (0)	100 (0)	91.7 (0)	100 (0)	95.2 (0)

Table 2 Recall and Precision evaluation from CHB and DHB

It is fairly clear from the results that the new CHB algorithm can handle with above situation very well, while DHB cannot detect anything at all. DHB can only work under normal situation when the image sequence is very clean, which is very rare in real world image sequence. Anyways, the CHB algorithm is still failed to detect in some particular circumstances. Such as when the images is gradually appeared some part of the image and continuously to full image without the change of intensity.

6. References

- [1] Campisi, P.; Neri, A.; Sorgi, L, Automatic dissolve and fade detection for video sequences. *14th International Conference on Digital Signal Processing*, 1-3 July 2002. 2: p. 567-570.
- [2] O. Fatemi, S.Zhang , S. Panchanathan, "Optical flow based model for scene cut detection" *Visual computing and communication laboratory*, University of Ottawa.
- [3] W.A.C Fernando, CN Canagarajah, D.R.Bull, "Automatic detection of fade-in and fade-out in video sequences" *IEEE International Symposium on Circuits and Systems*, 1999. 4: p. 255-258.
- [4] W.A.C Fernando, CN Canagarajah, D.R.Bull, "Fade and dissolve detection in uncompressed and compressed video sequence" *International Conference on Image Processing*, 24-28 October 1999. 3: p. 299 -303.
- [5] W.A.C Fernando, CN Canagarajah, D.R.Bull, "A unified approach to scene change detection in uncompressed and compressed video" *ICCE*, 13-15 June 2000: p. 350-351."
- [6] W.A.C Fernando, CN Canagarajah, D.R.Bull, "Fade, dissolve and wipe production in MPEG-2 compressed video" *ICCE*, 13-15 June 2000: p. 322-323.
- [7] W.A.C Fernando, CN Canagarajah, D.R.Bull, "Fade-in and fade-out detection in video sequence using histograms" *IEEE International Symposium on Circuits and Systems*, 2000. 4: p. 709 -712.



The 5th International Joint Conference on Computer Science and Software Engineering (JCSSE2008)

May 7th - 9th, 2008
Department of Computing, Faculty of Science
Silpakorn University, Thailand

Vol.1 : International Sessions





The 5th International Joint Conference on Computer Science and Software Engineering
(JCSSE2008)

May 7 – 9, 2008

JCSSE is an international conference held annually in Thailand. It provides a central forum for experts and developers to promote, share, and discuss various issues and developments in the broad field of computer science and software engineering. JCSSE also provide an opportunity for young researchers to demonstrate their talents and interesting research ideas. The conference will benefit people who are actively involved in research related to computer science, computer engineering, software engineering and digital library.

Selected papers will be published on the JCSSE 2008 proceedings. Excellent papers will be endorsed to submit to international journals including ECTI-CIT.

Computational Science and Engineering

- Bioinformatics
- DNA Computing
- Molecular Computing
- Nano Technology
- Numerical Methods

Computer Intelligence

- Artificial Intelligence
- Evolutionary Computing
- Fuzzy and Neural Computing
- Genetic Algorithms
- Intelligent Agents
- Machine Intelligence and Applications
- Natural Language Processing
- Neural Networks
- Robotics and Automation

Computer Graphics

- Computer Vision
- Multimedia Technology
- Visualization

Computer Systems

- Computer Architecture
- Computer Simulation and Modeling
- Computer Security
- Embedded Control Systems
- Fault-Tolerant Computing
- High Performance Computing
- Human-Computer Interface
- Optimization
- Parallel and Distributed Computing
- VLSI Design and Applications

Computer Networks

- Computer Networks
- Communication and QoS
- Data Communications
- Mobile Computing

Information & Communication Technology

- e-Business Technologies and Applications
- Internet Technology and Applications
- Semantic Web & Ontology
- Web Services

Computer Software

- Computer Aided Instruction
- Data Mining
- Database and Information Retrieval
- Decision Support Systems
- Distributed Database Systems
- Image and Signal Processing
- Geographical Information Systems
- Object Oriented Computing
- Programming Languages and Compilers

Digital Library

- Multimedia DLDB Engine
- Circulation Management
- Automatic Resource Category
- Digital Archive Management
- Digital Right Management (DRM)
- Automatic Indexing
- Smart Search Engine
- Automatic Collection Tools
- Ontology
- DL Architecture
- User Interface Service: speech, content delivery

Software Engineering

- Formal Methods
- Programming Techniques
- Software Requirements and Specifications
- System Modeling
- Software Components
- Software Prototyping
- Software Designs
- Software Validation
- Software Management
- Software Testing

Software Application & Development

- Tools & Products
- Experiences, Surveys & Reports
- Education & Training

Special Sessions

- Special sessions are to be peer-reviewed with the same criteria used for the contributed papers

Committee

Honorary General Chair

Kanchit Malaiwongs, The Royal Institute, Thailand

General Chair

Chantana Phongpensri, SU

General Co-Chairs

Bundit Thipakorn, KMUTT

Chidchanok Lursinsap, CU

Wanchai Rivepiboon, CU

Intenational Advisory Committee

Bin Xiao, Hong Kong Polytechnic U., China

Chokchai Leangsuksun, Louisiana Tech U., USA

Edwin Sha, U. of Texas, Dallas, USA

Gail Ridley, Tasmania U., Australia

Judy Young, Tasmania U., Australia

Vincent Belmont, Grenoble U., France

Vilas Wuwongse, AIT, Thailand

Steering Committee

Annart Pohthong, PSU

Anongnart Srivihok, KU

Boonchuay Srithammasak, RU

Chantana Phongpensri, SU

Chularat Tanprasert, NECTEC

Darunee Smavatkul, CMU

Jarernsri Mitranont, MAHIDOL

Krisana Chinnasarn, BUU

Kosin Chamnongthai, KMUTT

Nawapak Eua-Anant, KKU

Opas Wongtaweasap, SU

Peraphon Sophatsathit, CU

Pornsiri Muenchaisri, CU

Prabhas Chongstitwattana, CU

Punladda Subpayanon, UBU

Punnee Sittidech, NU

Rachada Kongkachandra, TU

Sartra Wongthanawasu, KKU

Sathit Nakkrasae, RU

Seree Chinodom, BUU

Sinchai Kamolphiwong, PSU

Somnuk Puangpronpitak, MSU

Sunee Pongpinigpinyo, SU

Sunee Raksakietisak, SWU

Surapong Auwatanamongkol, NIDA

Umaporn Sirathranont, KU

Veera Boonjing, KMITL

Yaowadee Temtanapat, KMITNB

Organizing Committee

Apisake Hongwitayakorn, SU

Bandid Bhurichitiporn, SU

Chantana Phongpensri, SU

Kata Praditwong, SU

Kittipong Kongaed, KRU

Kristsana Seepanomwan, SU

Narongdech Ratananonsatearn, KRU

Nattachote Promrit, SU

Nunnapat Toaditep, SU

Opas Wongtaweasap, SU

Panjai Tantatsanawong, SU

Rawitat Pulam, SU

Samran Quankuar, KRU

Sansanee Sithisak, SU

Saowalak Arampongsanuwat, SU

Sirak Kaewjamnong, SU

Suchitra Adulkasem, SU

Sunee Pongpinigpinyo, SU

Tasanawan Soonklang, SU

Wisoot Worasangasilp, SU

JCSSE2008 Secretariat

Opas Wongtaweasap, SU

Reviewers

- Prof.Dr. Boonserm Kijirikul
Prof.Dr. Chidchanok Lursinsap
Prof.Dr. Prabhas Chongstitvatana
Prof.Dr. Rattikorn Hewett
Prof.Dr. Walter Dosch
Asso.Prof.Dr. Chantana Phongpensri
Asso.Prof.Dr. Chuleerat Jaruskulchai
Asso.Prof.Dr. Ekachai Leelarasmee
Asso.Prof.Dr. Ekawit Nantajeewarawat
Asso.Prof.Dr. Jack Asavanant
Asso.Prof.Dr. Kitsana Waiyamai
Asso.Prof.Dr. Kosin Chamnongthai
Asso.Prof.Dr. Lance C.C. Fung
Asso.Prof.Dr. Montri Karnjanadecha
Asso.Prof.Dr. Naruemon Wattanapongsakorn
Asso.Prof.Dr. Nittaya Kerdprasop
Asso.Prof.Dr. Ouen Pinnern
Asso.Prof.Dr. Peraphon Sophatsathit
Asso.Prof.Dr. Pornsiri Muenchaisri
Asso.Prof.Dr. Pradondet Nilagupta
Asso.Prof.Dr. Prasong Praneetpolgrang
Asso.Prof.Dr. Punpiti Piamsa-nga
Asso.Prof.Dr. Sartra Wongthanavas
Asso.Prof.Dr. Supachai Tangwongsan
Asso.Prof.Dr. Suphamit Chittayasothorn
Asso.Prof.Dr. Surapong Auwatanamongkol
Asso.Prof.Dr. Thumrongrat Amornraksa
Asso.Prof.Dr. Vara Varavithya
- Asso.Prof.Dr. Veera Boonjing
Asso.Prof.Dr. Wael Badawy
Asso.Prof.Dr. Wanida Kanarkard
Asso.Prof.Dr. Wichian Chutimaskul
Asso.Prof.Dr. Yaowadee Temtanapat
- Asso.Prof.Dr. Yongyuth Permpoontanalarp
Asst.Prof.Dr. Amnart Pohthong
Asst.Prof.Dr. Apirak Jirayusakul
Asst.Prof.Dr. Arthit Thongtak
Asst.Prof.Dr. Bin Xiao
Asst.Prof.Dr. Bundit Thipakorn
Asst.Prof.Dr. Bunyarit Uyyanonvara
Asst.Prof.Dr. Chomtip Pornpanomchai
Asst.Prof.Dr. Daranee Horndee
- Chulalongkorn University
Chulalongkorn University
Chulalongkorn University
Texas Tech University
University of Luebeck
Silpakorn University
Kasetsart University
Chulalongkorn University
Thammasat University
Chulalongkorn University
Kasetsart University
King Mongkut's University of Technology Thonburi
Murdoch University
Prince of Songkla University
King Mongkut's University of Technology Thonburi
Suranaree University of Technology
King Mongkut's Institute of Technology Ladkrabang
Chulalongkorn University
Chulalongkorn University
Kasetsart University
Sripatum University
Kasetsart University
Khonkaen University
Mahidol University
King Mongkut's Institute of Technology Ladkrabang
National Institute of Development Administration
King Mongkut's University of Technology Thonburi
King Mongkut's University of Technology North Bangkok
King Mongkut's Institute of Technology Ladkrabang
University of Calgary
Khonkaen University
King Mongkut's University of Technology Thonburi
King Mongkut's University of Technology North Bangkok
King Mongkut's University of Technology Thonburi
Prince of Songkla University
Ramkhamhaeng University
Chulalongkorn University
Hong Kong Polytechnic University
King Mongkut's University of Technology Thonburi
Thammasat University
Mahidol University
Khonkaen University

Reviewers

Asst.Prof.Dr. Jaruloj Chongstitvatana	Chulalongkorn University
Asst.Prof.Dr. Jeerayut Chaijaruwanich	Chiang Mai University
Asst.Prof.Dr. Jittat Fakcharoenphol	Kasetsart University
Asst.Prof.Dr. Kanda Saikaew	Khonkaen University
Asst.Prof.Dr. Kriengkrai Porkaew	King Mongkut's University of Technology Thonburi
Asst.Prof.Dr. Krisana Chinnasarn	Burapha University
Asst.Prof.Dr. Meikang Qiu	University of New Orleans
Asst.Prof.Dr. Natasha Dejdumrong	King Mongkut's University of Technology Thonburi
Asst.Prof.Dr. Nithinant Thammakoranonta	National Institute of Development Administration
Asst.Prof.Dr. Nuanwan Soonthornphisaj	Kasetsart University
Asst.Prof.Dr. Nuttanart Facundes	King Mongkut's University of Technology Thonburi
Asst.Prof.Dr. Orasa Tetiwat	Naresuan University
Asst.Prof.Dr. Ohm Sornil	National Institute of Development Administration
Asst.Prof.Dr. Paisan Nakmahachalasint	Chulalongkorn University
Asst.Prof.Dr. Pattarasinee Bhattarakosol	Chulalongkorn University
Asst.Prof.Dr. Pichaya Tandayya	Prince of Songkla University
Asst.Prof.Dr. Pizzanu Kanongchaiyos	Chulalongkorn University
Asst.Prof.Dr. Ponrudee Netisopakul	King Mongkut's Institute of Technology Ladkrabang
Asst.Prof.Dr. Pusadee Seresangtakul	Khonkaen University
Asst.Prof.Dr. Rajalida Lipikorn	Chulalongkorn University
Asst.Prof.Dr. Rattasit Sukhahuta	Chiang Mai University
Asst.Prof.Dr. Sanpawat Kantabutra	Chiang Mai University
Asst.Prof.Dr. Sarun Intakosum	King Mongkut's Institute of Technology Ladkrabang
Asst.Prof.Dr. Sathit Nakkrasae	Ramkhamhaeng University
Asst.Prof.Dr. Somnuk Keretho	Kasetsart University
Asst.Prof.Dr. Songrit Maneewongvatana	King Mongkut's University of Technology Thonburi
Asst.Prof.Dr. Songsakdi Rongviriyapanish	Thammasat University
Asst.Prof.Dr. Sukanya Phongsuphap	Mahidol University
Asst.Prof.Dr. Sukree Sinthupinyo	Thammasat University
Asst.Prof.Dr. Sunee Raksakietisak	Srinakharinwirot University
Asst.Prof.Dr. Taratip Suwannasart	Chulalongkorn University
Asst.Prof.Dr. Thanwadee T. Sunetnanta	Mahidol University
Asst.Prof.Dr. Tiranee Achalaku	King Mongkut's University of Technology Thonburi
Asst.Prof.Dr. Twittie Senivongse	Chulalongkorn University
Asst.Prof.Dr. Vasaka Visoottiviseth	Mahidol University
Asst.Prof.Dr. Vimolrat Ngamaramvaranggul	Chulalongkorn University
Asst.Prof.Dr. Wiwat Vatanawood	Chulalongkorn University
Asso.Prof. Anongnart Srivihok	Kasetsart University
Asso.Prof. Narong Buabthong	Thammasat University
Asso.Prof. Praiboon Pantaragphong	King Mongkut's Institute of Technology Ladkrabang
Asso.Prof. Prasart Boonsanong	Naresuan University
Asso.Prof. Siripattra Muanmalai	Kasetsart University
Asso.Prof. Surasak Sanguanpong	Kasetsart University
Asso.Prof. Wiroj Taweepworadej	Khonkaen University
Asso.Prof. Yuen Poovarawan	Kasetsart University
Asst.Prof. Bhichate Chiewthanakul	Khonkaen University

Reviewers

Asst.Prof. Darunee Smavatkul
Asst.Prof. Nongluk Covavisaruch
Asst.Prof. Nuansri Denwattana
Asst.Prof. Pobsit Kamolvej
Asst.Prof. Rujchai Ung-Arunyawee
Asst.Prof. Sirikorn Channual
Asst.Prof. Umaporn Sirathranont
Dr. Akara Prayote

Dr. Annupan Rodtook
Dr. Atchara Mahaweerawat
Dr. Atiwong Suchato
Dr. Benchaphon Limthanmaphon

Dr. Chalernpol Charnsripinyo

Dr. Chaiporn Jaikaeo
Dr. Charnchai Pluempitiwiriyawej
Dr. Chavalit Srisathapornphat
Dr. Chotirat Ratanamahatana
Dr. Chularat Tanprasert

Dr. Chumphol Krootkaew

Dr. Ekkasit Tiamkaew
Dr. Gail Ridley
Dr. Jaratsri Rungrattanaubol
Dr. Kamisetty R. Rao
Dr. Kanuengnij Kubola
Dr. Kasidit Chanchio
Dr. Kitt Tientanopajai
Dr. Kritchalach Thitikamol
Dr. Ladda Preechaveerakul
Dr. Md Maruf Hasan
Dr. Monchai Sopitkamon
Dr. Natawut Nupairoj
Dr. Nawapak Eua-Anant
Dr. Nelson Passos
Dr. Nittida Elz
Dr. Olarn Wongwirat
Dr. Panupong Wanjantuk
Dr. Passakon Prathombutr

Dr. Punnee Sittidech
Dr. Sally E. Goldin
Dr. Sethavidh Gertphol

Chiang Mai University
Chulalongkorn University
Burapha University
Kasetsart University
Khonkaen University
Kasetsart University
Kasetsart University
King Mongkut's University of Technology North
Bangkok
Ramkhamhaeng University
Ubon Rajathanee University
Chulalongkorn University
King Mongkut's University of Technology North
Bangkok
National Electronics and Computer Technology
Center
Kasetsart University
Chulalongkorn University
Kasetsart University
Chulalongkorn University
National Electronics and Computer Technology
Center
National Electronics and Computer Technology
Center
Naresuan University
University of Tasmania
Naresuan University
University of Texas at Arlington
Burapha University
Thammasat University
Khonkaen University
IBM
Prince of Songkla University
Shinawatra University
Kasetsart University
Chulalongkorn University
Khonkaen University
Missouri Western State University
Prince of Songkla University
King Mongkut's Institute of Technology Ladkrabang
Khonkaen University
National Electronics and Computer Technology
Center
Naresuan University
King Mongkut's University of Technology Thonburi
Kasetsart University



Reviewers

Dr. Sirapat Chiewchanwattana
Dr. Sirirut Vanichayobon
Dr. Somnuk Puangpronpitag
Dr. Sornthep Vannarat

Dr. Sudsanguan Ngamsuriyaroj
Dr. Sunee Pongpinigpinyo
Dr. Suphakant Phimoltares
Dr. Thitiwan Srinark
Dr. Vajirasak Vanijja
Dr. Waralak V. Siricharoen
Dr. Wasan Pattara-atikom

Dr. Wiphada Wettayaprasit
Dr. Zhiming Zhao
Chatchawit Apornthewan
Kata Praditwong
Kurt T. Rudahl
Nakornthip Prompoon
Opas Wongtaweesap
Robert Elz
Somchoke Ruengittinun
Tasanawan Soonklang
Witcha Feungchan

Khonkaen University
Prince of Songkla University
Mahasarakham University
National Electronics and Computer Technology
Center
Mahidol University
Silpakorn University
Chulalongkorn University
Kasetsart University
King Mongkut's University of Technology Thonburi
University of the Thai Chamber of Commerce
National Electronics and Computer Technology
Center
Prince of Songkla University
University of Amsterdam
Chulalongkorn University
Silpakorn University
King Mongkut's University of Technology Thonburi
Chulalongkorn University
Silpakorn University
Prince of Songkla University
Kasetsart University
Silpakorn University
Khonkaen University

P0096	Dissolve Scene Change Detection using Modify Image Histogram <i>W.Kompreyarat and J.Srinonchat</i>	334
P0105	A New Automated Approach for Semantic Caching on XML Queries <i>HamidReza Kermajani and Mehdi Abbasi</i>	339
P0134	A High Performance System for Printed Thai Character Recognition <i>Supachai Tangwongsan and Orawan Jungthanawong</i>	347
P0138	Grain Inspection System using Image Processing and Control Techniques <i>Zin Mar Nyein, Than Zaw Soe, Theingi and. Maung Maung Latt</i>	353
P0140	Mining Frequent Sequential Patterns with Regular Expression Constraints <i>Songrit Maneewongvatana and Rasamee Methamontri</i>	358
P0156	Decision Forests Classification and Association Rule Mining for Course Registration Model of Undergraduate Students <i>Pathom Pumpuang, Anongnart Srivihok, Prasong Praneetpolgrang and Somchai Numprasertchai</i>	364
P0158	Comparison of General Interpolation Methods in Projection Modification for Metal Artifacts Reduction in CT <i>Thanakorn Suma, Suthae Phoojaruenchanachai and Bunyarit Uyyanonvara</i>	370
P0198	An Insert-Friendly Numbering Scheme for XML Path Expression Indexing <i>Tipa Kongsrima and Songrit Maneewongvatana</i>	374

Digital Library

P0197	Extending the Reach of Typical Library OPAC - A New Way of Community-based Resource Sharing using Distributed Cataloging and Z39.50 Protocol <i>Md Maruf Hasan, Kovit Bonsri and Nophadol Jekjantuk</i>	380
P0206	Wiimote Interface for Cutting and Folding 3D Digital Model <i>Sutas Chaisinanantkul and Pizzanu Kanongchaiyos</i>	385

Dissolve Scene Change Detection using Modify Image Histogram

W.Kompreyarat and J.Srinonchat

Department of Electronic and Telecommunication Engineering
Faculty of Engineering, Rajamangala University of Technology Thanyaburi
Klong 6, Thanyaburi, Patumtani 12110, Thailand
weera.k@en.rmutt.ac.th, jakkree.s@en.rmutt.ac.th

Abstract

This paper presents a new technique to modify the image histogram information and use it as a source to detect the dissolve scene change after the success of rapid scene change detection using image histogram. The output of the proposed algorithm show the significant response changes during dissolve scene change. It is also able to use with different type of dissolve scene change such as additive dissolve, non-additive dissolve, cross dissolve and random dissolve. The result is compared with twin comparison technique which is traditional dissolve scene change detection using luminance histogram and the proposed algorithm shows much improvement in every type of dissolve scene change.

Key Words: Dissolve Scene Change Detection, Image Histogram

1. Introduction

Due to the successive of histogram base method in the use of rapidly scene change detection, the paper has proposed the enhance algorithm of using histogram to detect dissolve scene change. Dissolve is one of the scene change effect that is widely used in video sequence. Many researchers proposed the techniques to detect this type of scene change, such as W.A.C. Fernando, C.N. Canagarajah and D.R. Bull [1] has proposed the techniques to detect dissolve scene change directly from variance and mean of the intensity of image sequence. O. Fatemi, S. Zhang, S. Panchanathan [2] introduced the optical flow motion estimation based model that able to detect the dissolve. G. Lupatini, C. Saraceno, R. Leonardi [3] has proposed the comparative of scene change detection, while Loong-Fah cheong [4] has also proposed the evaluation of each technique. Both of the journals are shown one of the well-known histogram based technique, which is proposed by H.J. Zhang, A.Kankanhalli, S.W. Smoliar [5] namely "Twin comparison technique". This technique will be

implemented and compared with the new histogram based technique presented here.

The basic histogram based scene change detection is widely used because is simplicity and computation cost. The block diagram in figure show the basic scene change detection technique

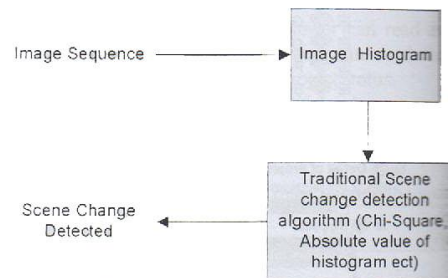


Figure 1 Histogram based Scene change Detection

2. Dissolve Scene Change

Dissolve is the scene change that one image sequence gradually disappear while the other image sequence gradually appear into the same scene Figure 2. Figure 3 and Figure 4 shows the examples of additive dissolve, non-additive dissolves and random dissolve respectively.



Figure 2 Example of additive dissolve



Figure 3 Example of non-additive dissolve

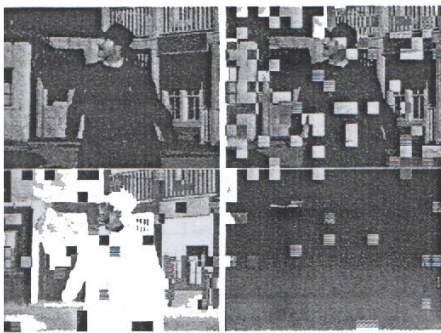


Figure 4 Example of random dissolve

Technically, dissolve is the proportion of the picture A that changes from 100% to 0% frame by frame gradually. This other picture B is also changes from 0% to 100% gradually at the same time.

W.A.C. Fernando, C.N. Canagarajah and D.R. Bull [1] have given the mathematical model for dissolve scene change as shows in equation 1.

$$s_n(i,j) = \begin{cases} f_n(i,j) & 0 \leq n < L_1 \\ \left[1 - \left(\frac{n-L_1}{F}\right)\right] \cdot f_n(i,j) + \left(\frac{n-L_1}{F}\right) \cdot g_n(i,j) & L_1 \leq n \leq (L_1 + F) \\ g_n(i,j) & (L_1 + F) < n \leq L_2 \end{cases} \quad (1)$$

Where, $S_n(i,j)$ is the resultant video signal

$f_n(i,j)$ is picture A

$g_n(i,j)$ is picture B

L_1 is length of video sequence A

F is length of fading sequence

L_2 is length of total video sequence

According to mathematical equation 1, it is showed that picture A is gradually fade out with the ratio of

$\left[1 - \left(\frac{n-L_1}{F}\right)\right]$ while picture B is fade in with the ratio of $\left(\frac{n-L_1}{F}\right)$ at the same time.

3. Dissolve scene change detection using twin comparison [5]

Twin comparison (TC) has developed to detect both gradually and rapid scene change by finding two thresholds, t_b and t_s for the histogram different of the video sequence. The t_b is the high threshold use to detect rapid scene change, due to the significant change of histogram during scene cut, while t_s is a low threshold to detect gradually scene change from the assumption that during gradually scene, the histogram different will be lower than t_b but it is slightly greater than histogram different is normal scene without changes.

In this technique, t_b and t_s will be calculated from the suggestion of V. Kobla, D. DeMenthon, and D. Doermann [6]. They suggested to calculate t_b from the equation 2

$$t_b = \mu + \omega\sigma \quad (2)$$

While, μ is mean of histogram different

σ is variance of histogram different

ω is a constant value, they suggest to use 5

t_s is low threshold that can be calculated by forming histogram different graph and identify the peak of the graph. Half of the peak is assigned to the t_s . They also suggest that t_s should be greater than mean value of histogram different. If t_s is smaller than mean value of histogram different, the marginally value that greater than mean value of histogram is to t_s instead.

4. Proposed Algorithm

The technique presented here is using the simple information from histogram to detect dissolve scene change. First of all, the image histogram $h_j(i)$ is calculated. Mean value of all histograms is constant. From the assumption that during dissolve scene change the histogram will be narrower. In order to classify the significant change of the mean value of histogram, histogram is multiplied by its bins number

that histogram belong to as in equation 3. After that mean of each histogram is calculated from equation 4.

$$hl_{i,j} = h_{i,j} * n_i \quad (3)$$

Where,

$h_{i,j}$ = histogram number i at frame j

n_i = constant matrix of [1..256]

$hl_{i,j}$ = new histogram number i at frame j

$$m_j = \text{mean}(hl_j) \quad (4)$$

Where,

hl_j = new histogram of images sequence at frame j

m_j = mean value of the new histogram at frame j

The different of mean value of new histogram is calculated after that. This process is done because the assumption the each of the mean value during dissolves should be changed slightly more than the scene change without any scene change. Equation 5 shows the way to calculate mean different values d_j .

The ideal of mean different value is show in Figure 5.

$$d_j = |m_j - m_{j-1}| \quad (5)$$

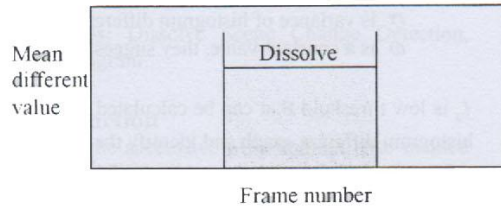


Figure 5 Ideal of mean different value

According to real world situation, some of the noises always occur during normal scene (not zero but very small when compared with the scene during dissolve scene change). In order to make the algorithm more accurate, the small threshold have been employ to get rid of a small value. Anything above the threshold is the candidate for dissolve scene change. Finally, the number of frame has been put in to account, if the number of frames above the threshold more than x frames continuously, that is the dissolve. In this paper, $x = 5$ is chosen, that mean anything smaller than 5 consecutive frames is not the dissolve scene change. Flowchart Diagram in Figure 6 show the whole process of the proposed algorithm.

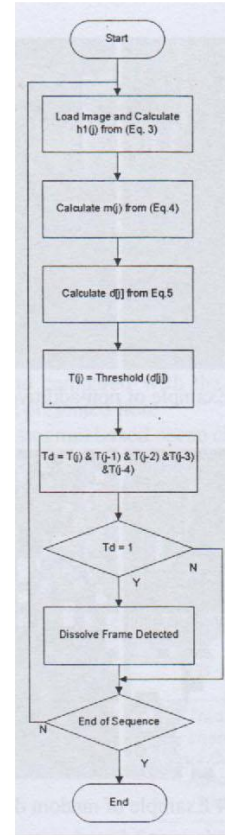


Figure 6 Flowchart of the proposed algorithm

5. Evaluation approach for dissolves scene change detection

Recall and precision are not suitable for this type of scene changes due to the detect dissolve might not cover the whole real dissolve, or might be greater than real dissolve. In order to evaluate the partial mistakes of the algorithm, two parameters, cover recall and cover precision [3,7,8,9] have been introduced. The cover recall is the percentage of cover length of correct detection with the real length of dissolve. The cover precision is the percentage of cover length of correct detection with the declare dissolve. Both cover recall and cover precision are showed in equation 6 and 7 respectively.

$$recall_{cover} = \frac{b}{a} * 100\% \quad (6)$$

$$precision_{cover} = \frac{b}{c} * 100\% \quad (7)$$

Where, a is the length of the real dissolve
 b is the length of the real dissolve cover by
the declared dissolve
 c is the length of the declared dissolve

6. Simulations and Result

Both dissolve scene change detection technique, twin comparison and mean value of image histogram is implemented. The simulation is done using the following computer specification.

Software: MATLAB version 6.1
OS: Windows XP with service pack 2
CPU: Intel Pentium 4, 2GHz with 512 Kbytes
cache memory, 400Mhz front side bus
Memory: DDR2100 512 MB

Four dissolves scene changes that are used to evaluate the algorithm are generated using Adobe Premier Version 6.5. Four types of well-known dissolve scene changes that are additive dissolves, non-additive dissolves, cross dissolves and random dissolves are used. The images sequence specifications are showed in Table 1. Each scene contains 887 frames. The original scene are also very noisy and the gradually change duration is about 40-50 Frame (1.8 Second – 2 Second).

Scene name	Scene 1	Scene2	Scene3	Scene4
Type of scene change	Additive dissolve	Non-additive dissolve	Cross dissolve	Random dissolve
Number of scene change	7	7	7	7

Table 1 List of the sequence to test dissolve scene change detect algorithm

First of all, the implementation of twin comparison is done. The threshold is calculated from absolute histogram different. The example of histogram different is showed in Figure 7 and Figure 8 showed the example of mean of new histogram that is calculated from equation 4. It is clear the example that the dissolve is very easy to identify. Figure 9 showed the example of the output from equation 5. The evaluation approach from equation 6 and equation 7 are employed. The comparison of the result is showed in Table 2.

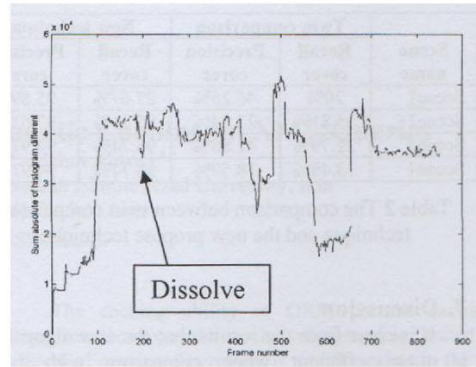


Figure 7 Example of absolute histogram different

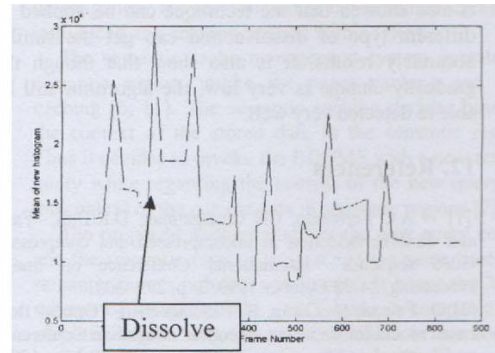


Figure 8 Example of mean of modification histogram

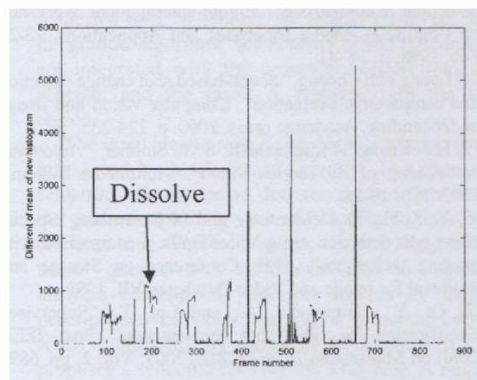


Figure 9 Example of Absolute different-of mean of modification histogram

Scene name	Twin comparison		New technique	
	Recall cover	Precision cover	Recall cover	Precision cover
Scene1	20%	46.25%	97.67%	95.89%
Scene2	15.81%	23.94%	95.81%	97.63%
Scene3	22.79%	41.88%	96.74%	94.97%
Scene4	13.48%	18.59%	96.74%	94.97%

Table 2 The comparison between twin comparison technique and the new propose technique

7. Discussion

It is clear from the results that the new algorithm is more efficient when comparing with twin comparison technique. It is very less fault detection and also has a good cover of the dissolve detection. It is also showed that the technique can be applied to different type of dissolve and can get the similar accurately results. It is also show that though the gradually change is very low, the algorithm still be able to detected very well.

12. References

- [1] W.A.C Fernando, CN Canagarajah, D.R.Bull, "Fade and dissolve detection in uncompressed and compressed video sequence". International Conference on Image Processing, 24-28 October 1999. 3: p. 299 -303.
- [2] O. Fatemi, S. Zhang, S. Panchanathan, "Optical flow based model for scene cut detection". Canadian Conference on Electrical and Computer Engineering, 26-29 May 1996. 1: p. 470 -473.
- [3] G. Lupatini, C. Saraceno, R. Leonardi, "Scene break detection: a comparison". Eighth International Workshop on Continuous-Media Databases and Applications, 23-24 Feb. 1998: p. 34 -41.
- [4] Loong Fah Cheong, "Scene-based shot change detection and comparative evaluation". Computer vision and image understanding, Academic press, 2000: p. 224-235.
- [5] H.J. Zhang, A.Kankanhalli, S.W. Smoliar, "Automatic partitioning of full-motion video". Multimedia Systems, 1993. 1: p. 10-28.
- [6] V. Kobla, D. DeMenthon, and D. Doermann, "Special effect edit detection using VideoTrails: a comparison with existing techniques". SPIE Conference on Storage and Retrieval for Image and Video Databases VII, 1999
- [7] Qi, Y., Hauptmann, A., and Liu, T.: "Supervised classification for video shot segmentation". Proc. IEEE Conf. on Multimedia Expo (ICME),2003, Vol. 2, pp. 689-692
- [8] "TREC Video retrieval evaluation", 2005 [Online]. Available: <http://www.nlpir.nist.gov/projects/trecvid/>
- [9] L. Gao, J. Jiang, J. Liang, S. Wang, S. Yang and Y. Qin, "PCA-based approach for video scene change detection on compressed video", Electronics Letters 23rd November 2006 Vol. 42 No. 24



ประวัติผู้เขียน

ชื่อ - นามสกุล นายวีระ คมปรียารัตน์
วัน เดือน ปีเกิด 9 ตุลาคม พ.ศ.2514
ที่อยู่ 27/188 หมู่ 7 ต.คลองหก อ.คลองหลวง จ.ปทุมธานี 12120
ประวัติการศึกษา สำเร็จการศึกษาระดับวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมไฟฟ้า – คอมพิวเตอร์
จากมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
เมื่อ พ.ศ.2538

ความชำนาญเฉพาะทาง

1. การประมวลผลสัญญาณดิจิทัล
2. การประมวลผลสัญญาณภาพ
3. ระบบฐานข้อมูล
4. การวิเคราะห์และออกแบบระบบ
5. วิศวกรรมซอฟต์แวร์

ประวัติการทำงาน
พ.ศ.2538 - ปัจจุบัน ตำแหน่งอาจารย์ประจำภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี

ผลงานตีพิมพ์เผยแพร่

W. Kompreyaratand J. Srinonchat, "Dissolve Scene Change Detection Using Modify Image Histogram ", 2008 Fifth International Conference on Computer Graphics, Imaging and Visualization (CGiV'08), 26-28 August 2008, Penang Malaysia, 2008, pp. 276-279

W. Kompreyaratand J. Srinonchat, "Fade Scene Change Detection in Noisy Video using Luminance Histogram Bin"