



การปรับปรุงความแม่นยำสำหรับการวัดชีพจรโดยไม่สัมผัสด้วยการวิเคราะห์องค์ประกอบอิสระ



โดย
นายณัฐพงศ์ แต่งจ้อย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

ภาควิชาคอมพิวเตอร์

บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

ปีการศึกษา 2558

ลิขสิทธิ์ของบัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

การปรับปรุงความแม่นยำสำหรับการวัดชีพจรโดยไม่สัมผัสด้วยการวิเคราะห์ห้วงค์ประกอบอิสระ



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ

ภาควิชาคอมพิวเตอร์

บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

ปีการศึกษา 2558

ลิขสิทธิ์ของบัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

**ACCURACY ENHANCEMENTS FOR CONTACT-FREE PULSE MEASUREMENT USING
INDEPENDENT COMPONENT ANALYSIS**



**By
Mr. Nattapong Tangjui**

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree

Master of Science Program in Information Technology

Department of Computing

Graduate School, Silpakorn University

Academic Year 2015

Copyright of Graduate School, Silpakorn University

บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร อนุมัติให้วิทยานิพนธ์เรื่อง “ การปรับปรุงความ
แม่นยำสำหรับการวัดชีพจรโดยไม่สัมผัสด้วยการวิเคราะห์องค์ประกอบอิสระ ” เสนอโดย นายณัฐพงศ์
แดงจួយ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยี
สารสนเทศ

.....
(รองศาสตราจารย์ ดร.ปานใจ ชารัตน์วงศ์)

คณบดีบัณฑิตวิทยาลัย

วันที่.....เดือน..... พ.ศ.....

อาจารย์ที่ปรึกษาวิทยานิพนธ์

อาจารย์ ดร.กัญญา แท้ประสาทสิทธิ์

คณะกรรมการตรวจสอบวิทยานิพนธ์

..... ประธานกรรมการ

(อาจารย์ ดร.สุนีย์ พงษ์พิทักษ์กัญญา)

..... กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร. นัทธี นิภานันท์)

..... กรรมการ

(อาจารย์ ดร.รัชดาพร คณาวัณษ์)

..... กรรมการ

(อาจารย์ ดร.กัญญา แท้ประสาทสิทธิ์)



57309202 : สาขาวิชาเทคโนโลยีสารสนเทศ

คำสำคัญ : การวัดชีพจรโดยไม่สัมผัสร่างกาย/การวิเคราะห์องค์ประกอบอิสระ

ณัฐพงศ์ แต่งจ้อย : การปรับปรุงความแม่นยำสำหรับการวัดชีพจรโดยไม่สัมผัสด้วยการวิเคราะห์องค์ประกอบอิสระ. อาจารย์ที่ปรึกษาวิทยานิพนธ์ : อ.ดร.ภิญโญ แท้ประสาทสิทธิ์. 85หน้า.

งานวิจัยนี้มีวัตถุประสงค์เพื่อปรับปรุงการวัดค่าชีพจรโดยไม่สัมผัสร่างกายด้วยการวิเคราะห์องค์ประกอบอิสระ (Independent Component Analysis : ICA) โดยวิเคราะห์จากการเปลี่ยนแปลงค่าสีที่เกิดจากการไหลเวียนของเลือดในวิดีโอ แยกองค์ประกอบของชีพจรด้วยการวิเคราะห์องค์ประกอบอิสระ (ICA) และวิเคราะห์ความถี่การเปลี่ยนแปลงด้วย Fast-Fourier Transform (FFT) ศึกษาผลกระทบ 2 ด้าน คือ 1. ผลกระทบจากการเปลี่ยนโมเดลสีและขนาดหน้าต่าง 2. ปรับปรุงความแม่นยำในการระบุค่าชีพจรด้วยวิธีการที่นำเสนอประกอบด้วย multi-component accumulation, range-power spectrum, และปรับปรุง error suppression multi-component-accumulation เป็นเทคนิคที่ช่วยให้สามารถรวมรวมข้อมูลที่เป็นประโยชน์เกี่ยวกับการเปลี่ยนแปลงของสีได้มากกว่าการใช้โมเดลสีเดียว range-power-spectrum เป็นหลักการที่ใช้แก้ปัญหาเกี่ยวกับการเปลี่ยนแปลงชีพจรของผู้ทำการทดลองที่เกิดขึ้นในหน้าต่าง การปรับปรุง error suppression ทำให้สามารถจัดการกับความผิดพลาดได้ โดยการตัดส่วนที่ไม่เกี่ยวข้องกับชีพจรที่มีค่า power spectrum ต่ำ การใช้หลักการที่นำเสนอร่วมกันทำให้ได้วิธีการที่มีความแม่นยำมากกว่างานอื่น

ทดสอบระบบการวัดค่าชีพจรโดยไม่สัมผัสร่างกายกับผู้ทำการทดลอง 20 คน อายุระหว่าง 10 – 55 ปี เพศชาย 5 คนและเพศหญิง 15 คน ยืนยันผลการทดลองของหลักการที่เสนอโดยบันทึกวิดีโอ 2 วิดีโอสำหรับผู้ทำการทดลองแต่ละคน ประกอบด้วย iPad3 และ Logitech C920 ผลการวิจัยพบว่าระบบสามารถระบุค่าชีพจรได้แม่นยำสูงขึ้น โดยมีค่าความผิดพลาดเฉลี่ยที่ 5.42 bpm เมื่อใช้ขนาดหน้าต่าง 40 วินาที ซึ่งมีความแม่นยำมากกว่างานอื่นประมาณ 0.50 bpm

ภาควิชาคอมพิวเตอร์

บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

ลายมือชื่อนักศึกษา.....

ปีการศึกษา 2558

ลายมือชื่ออาจารย์ที่ปรึกษาวิทยานิพนธ์

57309202 : MAJOR : INFORMATION TECHNOLOGY

KEY WORD : CONTACT-FREE PULSE MEASUREMENT/INDEPENDENT COMPONENT ANALYSIS

NATTAPONG TANGJUI : ACCURACY ENHANCEMENTS FOR CONTACT-FREE PULSE MEASUREMENT USING INDEPENDENT COMPONENT ANALYSIS. THESIS ADVISOR : PINYO TAEPRASARTSIT, Ph.D. 85 pp.

The purpose of this research is to enhance the contact-free pulse measurement by applying Independent Component Analysis (ICA) which analyze the change of color in video sequence caused by blood perfusion. ICA separates the pulse source and utilize Fast-Fourier Transform (FFT) to find a dominant frequency of the subtle changes. The study was executed in two aspects: (1) evaluating the impacts of color models and time-window length and (2) enhancing accuracy by our proposed techniques which include multi-component accumulation, range-power spectrum, and improved error suppression.

The multi-component-accumulation technique helps gathering more useful information about color change that a single color model may not be able to provide. Our range-power-spectrum method can handle a case where a subject's heart rate significantly changes in a single time window, while improved error suppression copes with errors by filtering out unlikely heart rates whose FFT power spectrums are comparatively low. By combining these three proposed techniques together, we achieve a robust method that is more accurate than prior work.

The contact-free pulse measurement system was tested by 20 subjects, aged from 10 to 55 years old. Five were male and fifteen were female. To validate whether the method works well for video sequences recorded by different devices, we captured two videos for each subject; one was recorded by iPad3 and another by Logitech C920. The results showed that the proposed method could improve accuracy of heart-rate measurement over past work for both devices. Its average error from both devices was 5.42 bpm when a time-window length was 40 seconds. This was more accurate than other past work by approximately 0.50 bpm.

Department of Computing

Student's signature

Thesis Advisor's signature

Graduate School, Silpakorn University

Academic Year 2015

กิตติกรรมประกาศ

งานวิทยานิพนธ์นี้สำเร็จลุล่วงได้ด้วยความอนุเคราะห์จาก อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ อาจารย์ที่ปรึกษางานวิทยานิพนธ์ ท่านได้ให้ความรู้ คำปรึกษาแนะนำ ช่วยเหลือให้กำลังใจชี้แนะแนวทาง แก้ไขส่วนที่บกพร่อง ด้วยดีตลอดมา ขอกราบขอบพระคุณอย่างสูง

ผู้วิจัยขอกราบขอบพระคุณบุพการี และคณะวิทยาศาสตร์ สาขาเทคโนโลยีสารสนเทศ ที่มอบโอกาสในการศึกษาจนประสบความสำเร็จ รวมถึงกัลยาณมิตรที่ดี 20 ท่าน ที่เป็นส่วนสำคัญในงานวิทยานิพนธ์นี้ด้วย

ท้ายสุดผู้วิจัยหวังเป็นอย่างยิ่ง งานวิทยานิพนธ์นี้จะเป็นประโยชน์ต่อผู้ที่สนใจ และสามารถนำไปใช้ต่อยอดการเรียน การสอน การทำงาน ได้อย่างมีประสิทธิภาพ และเกิดประสิทธิผลมากที่สุด หากมีข้อผิดพลาดประการใด ผู้วิจัยขออภัยมา ณ ที่นี้ด้วย



สารบัญ

หน้าที่

บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญตาราง.....	ญ
สารบัญภาพ.....	ฎ
บทที่	
1 บทนำ.....	1
ความเป็นมาและความสำคัญของปัญหา.....	1
วัตถุประสงค์ของการวิจัย.....	2
ประโยชน์ที่คาดว่าจะได้รับ.....	3
โครงร่างหัวข้อวิทยานิพนธ์.....	3
2 งานวิจัยที่เกี่ยวข้อง.....	4
Non-contact, automated cardiac pulse measurements using video imaging and blind source separation.....	4
Noncontact Automatic Heart Rate Analysis in Visible Spectrum by Specific Face Regions.....	6
Eulerian Video Magnification for Revealing Subtle Changes in the World.....	7
Preprocessing Realistic Video for Contactless Heart Rate Monitoring Using Video Amplification Methodologies.....	9
Android-based implementation of Eulerian Video Magnification for vital signs monitoring.....	11

สรุปปัจจัยที่มีผลต่อการวัดชีพจรโดยไม่สัมผัสร่างกาย	12
กล้องและโมเดลสี	12
ขนาดหน้าต่างและ framerate	12
สีผิวและแสงสว่าง	13
3 ทฤษฎีที่เกี่ยวข้อง	14
Color model	14
Frame blocking & Normalization	17
Band-pass filter	21
Independence Component Analysis (ICA)	23
Fast-Fourier Transform (FFT)	24
Multi-component accumulation (MCA)	27
Range-power spectrums (RPS)	29
Error suppression (ES)	32
4 วิธีการดำเนินงานวิจัย	36
โครงสร้างระบบ	36
Hardware	36
Software	36
ขั้นตอนการทำงานของระบบ	37
Pre-processing	37
Heart rate measurement	41
Post-processing	43
5 การทดสอบระบบและผลการดำเนินการวิจัย	48

การทดสอบระบบ	48
ผลการดำเนินการวิจัย.....	52
กราฟชีพจร.....	53
ผลกระทบจาก MCA	63
ผลกระทบจาก RPS	65
ผลกระทบจาก ES	67
เปรียบเทียบความแม่นยำระหว่าง MCA และ MCA&ES (ไม่ทำ RPS)	67
เปรียบเทียบความแม่นยำระหว่าง MCA&RPS และ MCA&RPS&ES	67
เปรียบเทียบความแม่นยำระหว่าง ES ที่นำเสนอ และ ES ของ Poh	68
ผลกระทบจากการเปลี่ยนขนาดหน้าต่าง.....	69
ผลกระทบจากการเปลี่ยนโมเดลสี.....	70
ผลกระทบจากอุปกรณ์	71
6 สรุปลผลการทดลอง และข้อเสนอแนะ.....	72
สรุปลผลการทดลอง.....	72
ข้อเสนอแนะ.....	73
รายการอ้างอิง	75
ภาคผนวก.....	77
ภาคผนวก ก เอกสารการนำเสนอผลงานทางวิชาการ	78
ประวัติผู้ทำวิจัย	85

สารบัญตาราง

	หน้าที่
ตารางที่ 1 เปรียบเทียบความเร็วการคำนวณ DFT และ FFT	25
ตารางที่ 2 แสดงความถี่ 5 อันดับแรก เรียงตามค่า Magnitude ของผู้ทำการทดลองคนที่ 8 หน้าต่างที่ 34	27
ตารางที่ 3 แสดงความถี่ 5 อันดับแรก เรียงตามค่า Magnitude ของผู้ทำการทดลองคนที่ 8 หน้าต่างที่ 41	28
ตารางที่ 4 ความถี่ 10 อันดับแรก เรียงตามค่า Magnitude ของวิธี MCA จากตารางที่ 2.....	28
ตารางที่ 5 ความถี่ 10 อันดับแรก เรียงตามค่า Magnitude ของวิธี MCA จากตารางที่ 3.....	29
ตารางที่ 6 ความถี่ 20 อันดับแรก เรียงตามค่า Magnitude ของวิธี RPS จากตารางที่ 4	31
ตารางที่ 7 ความถี่ 20 อันดับแรก เรียงตามค่า Magnitude ของวิธี RPS จากตารางที่ 5	31
ตารางที่ 8 ความเป็นไปได้ในการของการเปลี่ยนแปลงค่าซีพจรใน 1 วินาที.....	33
ตารางที่ 9 แสดงการเลือกค่า Magnitude ที่สูงกว่า 70% ของ Magnitude สูงสุด.....	33
ตารางที่ 10 ความถี่แต่ละหน้าต่าง ก่อนพิจารณาความถี่หน้าต่างที่ 1 จาก 3 หน้าต่างแรก.....	34
ตารางที่ 11 ความถี่แต่ละหน้าต่าง หลังพิจารณาความถี่หน้าต่างที่ 1 จาก 3 หน้าต่างแรก	34
ตารางที่ 12 แสดงการคำนวณ ES	34
ตารางที่ 13 ผลลัพธ์การคำนวณซีพจรด้วย ES จากตารางที่ 10.....	35
ตารางที่ 14 ผลลัพธ์การคำนวณซีพจรด้วย ES จากตารางที่ 11.....	35
ตารางที่ 15 แสดงจำนวนหน้าต่างในแต่ละขนาดหน้าต่าง.....	39
ตารางที่ 16 ความถี่ที่มีค่า Magnitude สูงสุด คำนวณจาก FFT ของสัญญาณที่ผ่านการทำ ICA.....	42
ตารางที่ 17 ความถี่ที่มีค่า Magnitude สูงสุด คำนวณจาก MCA และ RPS.....	44

ตารางที่ 18 ผลลัพธ์การคำนวณชีพจรในแต่ละหน้าต่าง โดยใช้วิธี MCA, RPS และ ES เปรียบเทียบกับชีพจรที่ได้จาก Arduino pulse sensor.....	45
ตารางที่ 19 ข้อมูลความดันเลือดและชีพจรผู้ทำการทดลอง	50
ตารางที่ 20 ข้อเปรียบเทียบของทั้ง 2 อุปกรณ์.....	51
ตารางที่ 21 แสดงผลลัพธ์การคำนวณผลต่างของชีพจรที่วัดได้กับ Arduino pulse sensor	52
ตารางที่ 22 ผลกระทบจากวิธี MCA ต่อความแม่นยำ.....	63
ตารางที่ 23 ผลกระทบจากวิธี RPS ต่อความแม่นยำ.....	66
ตารางที่ 24 เปรียบเทียบความแม่นยำการทำและไม่ทำ ES โดยทำเพียง MCA สำหรับโมเดลสี MMS	67
ตารางที่ 25 เปรียบเทียบความแม่นยำการทำและไม่ทำ ES โดยทำ MCA ร่วมกับ RPS สำหรับโมเดลสี MMS	68
ตารางที่ 26 เปรียบเทียบความแม่นยำระหว่าง ES ของ Poh กับ ES ที่นำเสนอ.....	68
ตารางที่ 27 แสดงการเปรียบเทียบความแม่นยำในการระบุค่าชีพจรของแต่ละโมเดลสี โดยเฉลี่ยผลลัพธ์จากทุกขนาดหน้าต่างยกเว้นขนาด 5 วินาทีต่อหน้าต่าง	70
ตารางที่ 28 เปรียบเทียบความแม่นยำของ 2 อุปกรณ์ คือ iPad3 และ Logitech C920	71

สารบัญภาพ

หน้าที่

รูปที่ 1 แสดงสัญญาณที่ได้จากแต่ละเฟรม a) วิดีโอเฟรมต้นฉบับ b) แยกสัญญาณวิดีโอเฟรม ออกเป็น 3 แชนแนล c) สัญญาณแต่ละแชนแนลของ RGB d) สัญญาณ ICA ทั้ง 3 องค์ประกอบ.....	4
รูปที่ 2 แสดงผลการวิเคราะห์ FFT ของสัญญาณ ICA แต่ละองค์ประกอบ	5
รูปที่ 3 อุปกรณ์วัดชีพจร BVP	6
รูปที่ 4 แสดงผลลัพธ์การเลือกตำแหน่งใบหน้าการทำ AAM.....	6
รูปที่ 5 ผลลัพธ์การเลือกส่วนที่สนใจด้วย AAM และแยกออกเป็น 3 แชนแนล.....	7
รูปที่ 6 Flow chart วิธีการ EVM ที่เสนอโดย Wu.....	7
รูปที่ 7 ผลลัพธ์การวัดชีพจรด้วย EVM ด้านบนคือ วิดีโอเฟรมต้นฉบับ ด้านล่างคือ การแสดงการเปลี่ยนแปลงของสีผิว.....	8
รูปที่ 8 Flow chart การวัดชีพจรที่เสนอโดย Ahmed Alzahrani.....	9
รูปที่ 9 Flow chart การวัดชีพจรที่เสนอโดย	12
รูปที่ 10 วิดีโอเฟรมต้นฉบับ โมเดลสี RGB.....	14
รูปที่ 11 วิดีโอเฟรมที่แปลงโมเดลสีเป็น HSV (ซ้าย) และ HLS (ขวา)	15
รูปที่ 12 วิดีโอเฟรมที่แปลงโมเดลสีเป็น CIE L*a*b* (ซ้าย) และ CIE L*u*v* (ขวา)	15
รูปที่ 13 วิดีโอเฟรมที่แปลงโมเดลสีเป็น TSL (ซ้าย) และ YCbCr (ขวา).....	15
รูปที่ 14 แสดงวิธีการสร้างโมเดล MMS.....	16
รูปที่ 15 การตัดแบ่งช่วงสัญญาณ.....	17
รูปที่ 16 การตัดแบ่งช่วงสัญญาณโดยมีการซ้อนทับสัญญาณ	18
รูปที่ 17 สัญญาณต้นฉบับ ก่อนการแบ่งสัญญาณ	18

รูปที่ 18 แบ่งสัญญาณด้วยขนาดหน้าต่าง 20 วินาที/หน้าต่าง (600 frame/window) ซ้ายคือ หน้าต่างที่ 1 และขวาคือหน้าต่างที่ 2.....	19
รูปที่ 19 แบ่งสัญญาณด้วยขนาดหน้าต่าง 20 วินาที/หน้าต่าง (600 frame/window) ซ้ายคือ หน้าต่างที่ 3 และขวาคือหน้าต่างที่ 4.....	19
รูปที่ 20 สัญญาณต้นฉบับทั้ง 3 แชนแนล.....	20
รูปที่ 21 สัญญาณทั้ง 3 แชนแนลที่ผ่านการ Normalized	21
รูปที่ 22 การตอบสนองความถี่ของ Butterworth Band-pass filter มีความถี่ cut-off ที่ 0.75 – 4 Hz	22
รูปที่ 23 สัญญาณทั้ง 3 แชนแนล ที่ผ่านการ Normalized และกรองความถี่ ของสัญญาณใน รูปที่ 20.....	22
รูปที่ 24 สัญญาณที่ผ่านการทำ ICA ของสัญญาณในรูปที่ 23	24
รูปที่ 25 หลักการทำ FFT.....	25
รูปที่ 26 ผลลัพธ์การวิเคราะห์ความถี่ด้วย FFT ของสัญญาณในรูปที่ 23	26
รูปที่ 27 แสดงชีพจรที่บันทึกได้จาก Arduino Pulse Sensor ของผู้ทำการทดลอง 5 คน ในเวลา 1 นาที	30
รูปที่ 28 Flowchart การวัดชีพจรแบบไม่สัมผัสร่างกาย	37
รูปที่ 29 Flowchart การทำ Pre-processing.....	37
รูปที่ 30 a) วิดีโอเฟรมที่ได้จากการบันทึก b) เลือกตัดส่วนอวัยวะที่สนใจเพื่อวัดชีพจร c) แปลงโมเดลสี	38
รูปที่ 31 คำนวณค่าเฉลี่ยสีทุกแชนแนลในแต่ละเฟรม	38
รูปที่ 32 สัญญาณค่าเฉลี่ยสีต้นฉบับ.....	40
รูปที่ 33 ผลลัพธ์การวิเคราะห์ ICA ของสัญญาณในรูปที่ 32.....	40
รูปที่ 34 Flowchart การวัดค่าชีพจร	41
รูปที่ 35 Flowchart การทำ Post-processing.....	43

รูปที่ 36 a) กล้องควบคุมแสง b) ขณะบันทึกวิดีโอด้วยกล้องควบคุมแสงในห้อมมือ c) วิดีโอเฟรมที่ได้จากการบันทึก.....	48
รูปที่ 37 ด้านซ้ายคือ เครื่องวัดความดันโลหิต ด้านขวาคือ Arduino pulse sensor.....	49
รูปที่ 38 แสดงชีพจรที่บันทึกได้จาก Arduino Pulse Sensor ของผู้ทำการทดลอง 20 คน ในเวลา 1 นาที	49
รูปที่ 39 อุปกรณ์บันทึกวิดีโอ ซ้าน iPad3 ขวา Logitech C920.....	51
รูปที่ 40 แสดงการคำนวณ CE3	53
รูปที่ 41 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี RGB ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS.....	54
รูปที่ 42 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี RGB ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ซ้าย) และ 2011 (ขวา).....	55
รูปที่ 43 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี HSV ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS.....	55
รูปที่ 44 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี HSV ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ซ้าย) และ 2011 (ขวา).....	56
รูปที่ 45 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี CIE L*a*b* ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS.....	56
รูปที่ 46 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี CIE L*a*b* ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ซ้าย) และ 2011 (ขวา).....	57
รูปที่ 47 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยการผสมโมเดลสีแบบ CMU ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS	57
รูปที่ 48 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยการผสมโมเดลสีแบบ MMS ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS	58
รูปที่ 49 กราฟชีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี RGB ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS.....	58

รูปที่ 50 กราฟสีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี RGB ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ซ้าย) และ 2011 (ขวา).....	59
รูปที่ 51 กราฟสีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี HSV ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS.....	59
รูปที่ 52 กราฟสีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี RGB ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ซ้าย) และ 2011 (ขวา).....	60
รูปที่ 53 กราฟสีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี CIE L*a*b* ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS.....	60
รูปที่ 54 กราฟสีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี CIE L*a*b* ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ซ้าย) และ 2011 (ขวา).....	61
รูปที่ 55 กราฟสีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยการผสมโมเดลสีแบบ CMU ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS.....	61
รูปที่ 56 กราฟสีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยการผสมโมเดลสีแบบ MMS ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS.....	62
รูปที่ 57 เปรียบเทียบความแม่นยำระหว่าง MMS, RGB (MCA), ICA Component 2 (Poh 2010) และ Maximum Magnitude (Poh 2011) โดยใช้หลักการ MCA เพียงอย่างเดียว.....	64
รูปที่ 58 เปรียบเทียบความแม่นยำระหว่าง MMS, RGG (MCA & RPS), ICA Component 2 (Poh 2010) และ Maximum Magnitude (Poh 2011) โดยใช้หลักการ MCA & RPS เพียงอย่างเดียว.....	65
รูปที่ 59 เปรียบเทียบความผิดพลาดในการระบุค่าสีพจร เมื่อเปลี่ยนขนาดหน้าต่าง	69
รูปที่ 60 กราฟสีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี HSV ขนาดหน้าต่าง 40 วินาที/หน้าต่าง ใช้วิธี MCA และ RPS.....	74

บทที่ 1

บทนำ

1.1. ความเป็นมาและความสำคัญของปัญหา

การวัดสัญญาณชีพของมนุษย์ไม่ว่าจะเป็น อัตราการเต้นของหัวใจและอัตราการหายใจ จำเป็นต้องอาศัยเครื่องมือ เช่น electrocardiogram (ECG) , Photo-plethysmography (PPG) และเครื่องวัดความดันโลหิต ซึ่งอุปกรณ์เหล่านี้จำเป็นต้องสัมผัสกับร่างกายจึงจะสามารถวัดชีพจรได้ จึงอาจก่อความรำคาญให้กับผู้ที่ต้องใช้เครื่องเหล่านี้เป็นเวลานานติดต่อกัน ดังนั้นจึงมีการพยายาม ตรวจวัดสัญญาณชีพโดยไม่มีการสัมผัสกับร่างกาย งานของ (Fadel Adib et al., 2014: 261-262) ใช้ การคำนวณระยะห่างจากการรับ-ส่งสัญญาณวิทยุในการวัดอัตราการหายใจ โดยวางเซ็นเซอร์ห่างจาก ผู้ทดลอง 10 – 15 เซนติเมตร สำหรับการวัดอัตราการเต้นของหัวใจโดยไม่มีการสัมผัสกับร่างกายถูก เสนอโดย (Ming-Zher Poh et al., 2010: 10762-10774) เป็นการพิจารณาการเปลี่ยนแปลงของค่า สีที่เกิดขึ้นภายในวิดีโอเฟรมที่เกิดจากการเปลี่ยนแปลงของสีผิวเนื่องจากการไหลเวียนของเลือด Independence Component Analysis (ICA) ใช้ในการสร้างตั้งสัญญาณการเปลี่ยนแปลงของค่าสี ดังกล่าวที่เกิดขึ้นภายในวิดีโอเฟรมและนำมาวิเคราะห์ความถี่ด้วย Fast-Fourier Transform (FFT)

(Wu et al., 2012: 651-658) เสนอการวัดค่าชีพจรโดยไม่สัมผัสที่พิจารณาจากวิดีโอเฟรม อีกวิธีหนึ่งคือ Eulerian Video Magnification (EVM) เป็นวิธีการที่จะแสดงการเปลี่ยนแปลงเล็กน้อย ที่เกิดขึ้นภายในวิดีโอเฟรมได้เรียกว่า Color และ Motion magnification ซึ่งเราสามารถประยุกต์ใช้ การเปลี่ยนแปลงของสีผิวที่ตรวจพบมาวัดชีพจรได้โดยไม่ต้องสัมผัสร่างกาย งานวิจัยหลายงานนำ EVM มาใช้ในการวัดอัตราการเต้นของหัวใจโดยใช้กล้องเว็บแคมและกล้องโทรศัพท์แอนดรอยด์ (Android)

งานวิจัยนี้เลือกวัดค่าชีพจรโดยไม่สัมผัสกับร่างกายโดยวิเคราะห์จากวิดีโอเฟรมตามหลักการ ICA เช่นเดียวกับ Poh ซึ่งการนำหลักการนี้มาใช้มีปัจจัยหลายอย่างต้องพิจารณา ได้แก่ 1. การเลือก องค์ประกอบของโมเดลสีที่นำมาวิเคราะห์ความถี่การเปลี่ยนแปลงที่สอดคล้องกับการเปลี่ยนแปลง

ของซีพจร 2. ปัญหาด้านความเหมาะสมของแสงสว่าง โดยแสงสว่างต้องไม่มากและไม่น้อยเกินไป เพราะ ถ้าน้อยเกินไปจะไม่สามารถเห็นถึงการเปลี่ยนแปลงได้ และถ้ามากเกินไปการเปลี่ยนแปลงของแสงจะเด่นชัดมากกว่าการเปลี่ยนแปลงของสีผิว 3. สีผิวคล้ำจะเห็นการเปลี่ยนแปลงน้อยกว่าสีผิวขาว 4. ประสิทธิภาพเซ็นเซอร์รับแสงของกล้องมีผลต่อความแม่นยำ 5. ความนิ่งของผู้ทำการทดลองขณะบันทึกวีดีโอมีผลต่อความแม่นยำ เพราะ ถ้าผู้ทำการทดลองมีการขยับมากเกินไปการเปลี่ยนแปลงของการขยับจะเด่นชัดมากกว่าการเปลี่ยนแปลงของสีผิว งานวิจัยที่ศึกษามาแก้ปัญหาเหล่านี้โดยการเพิ่มปริมาณแสงด้วยการเพิ่มค่าให้กับแชนแนล V ของโมเดลสี HSV เน้นการเปลี่ยนแปลงของสีผิวโดยการเพิ่มค่าแชนแนล R ของโมเดลสี RGB และการปรับค่าแชนแนล L^* และ b^* ของโมเดลสี CIE $L^*a^*b^*$ การทำ Pre-processing เลือกส่วนที่สนใจ (Region of interest : ROI) เพื่อลดการขยับของผู้ทำการทดลอง เช่น การตรวจจับใบหน้า (Face detection) การทำ Video stabilization

งานวิจัยที่ศึกษามาเลือกวิเคราะห์โมเดลสี RGB และ YIQ เป็นหลัก และเลือกใช้ขนาดหน้าต่าง (Time-window) เพียงขนาดเดียวและแตกต่างกันไปตามงานวิจัย ดังนั้นงานวิจัยนี้จึงศึกษาผลกระทบของการเปลี่ยนโมเดลสีและขนาดหน้าต่างที่มีผลต่อประสิทธิภาพความแม่นยำ พร้อมทั้งเสนอวิธีการปรับปรุงคุณภาพในการวัดซีพจรแบบไม่สัมผัสร่างกายตามหลักการของ ICA

1.2. วัตถุประสงค์ของการวิจัย

งานวิจัยนี้มีเป้าหมายเพื่อปรับปรุงความแม่นยำในการวัดซีพจรแบบไม่สัมผัสร่างกาย โดยแบ่งการศึกษาออกเป็นสามประเด็นย่อย ดังนี้

1. ผลกระทบจากการเปลี่ยนโมเดลสีและการผสมโมเดลสีร่วมกันในการวัดซีพจร โดยศึกษาการเปลี่ยนโมเดลสีทั้งหมด 7 โมเดล ได้แก่ RGB, HLS, HSV, CIE $L^*a^*b^*$ (Lab), CIE $L^*u^*v^*$ (Luv), TSL และ YCbCr การผสมโมเดลสี 2 แบบ คือ Multi-model Summation (MMS) และ Color-Model Union (CMU)
2. ผลกระทบจากการเปลี่ยนขนาดหน้าต่างที่ใช้ในการวัดซีพจร โดยเปลี่ยนศึกษาการเปลี่ยนขนาดหน้าต่าง 7 ขนาด คือ 5 10 15 20 30 40 และ 50 วินาทีต่อหน้าต่าง

3. เสนอวิธีการปรับปรุงการวัดชีพจรด้วยการทำ Post-processing 3 วิธี คือ Multi component accumulation (MCA), Range power spectrum (RPS) และ Error suppression (ES)

1.3. ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถสรุปโมเดลสีและการผสมโมเดลสีที่ทำให้การวัดชีพจรแบบไม่สัมผัสร่างกายมีความแม่นยำสูงสุด
2. สามารถสรุปการเลือกใช้ขนาดหน้าต่างที่ส่งผลให้มีความแม่นยำ
3. สามารถปรับปรุงการวัดชีพจรแบบไม่สัมผัสร่างกาย โดยการเพิ่มส่วน Post-processing 3 ส่วน คือ MCA, RPS และ ES

1.4. โครงร่างหัวข้อวิทยานิพนธ์

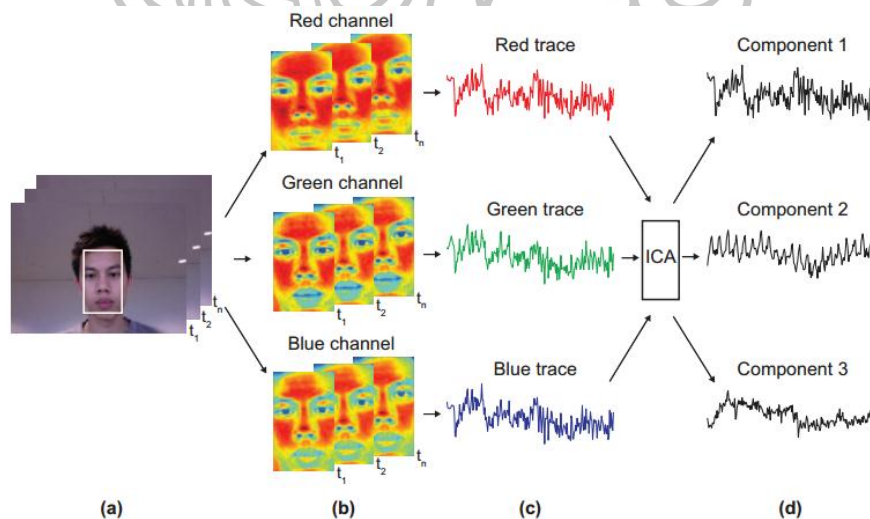
ลำดับหัวข้องานวิจัยจะเป็นไปดังนี้คือ บทที่ 2 เป็นงานวิจัยที่เกี่ยวข้องพร้อมทั้งปัจจัยต่างๆที่ส่งผลต่อการวัดชีพจรแบบไม่สัมผัสด้วยวิธี ICA บทที่ 3 คือทฤษฎีที่เกี่ยวข้อง บทที่ 4 คือวิธีการดำเนินงานวิจัย บทที่ 5 เป็นการทดสอบระบบและผลการดำเนินการวิจัย และส่วนการสรุปผลการทดลองและข้อเสนอแนะอยู่ในบทที่ 6



บทที่ 2 งานวิจัยที่เกี่ยวข้อง

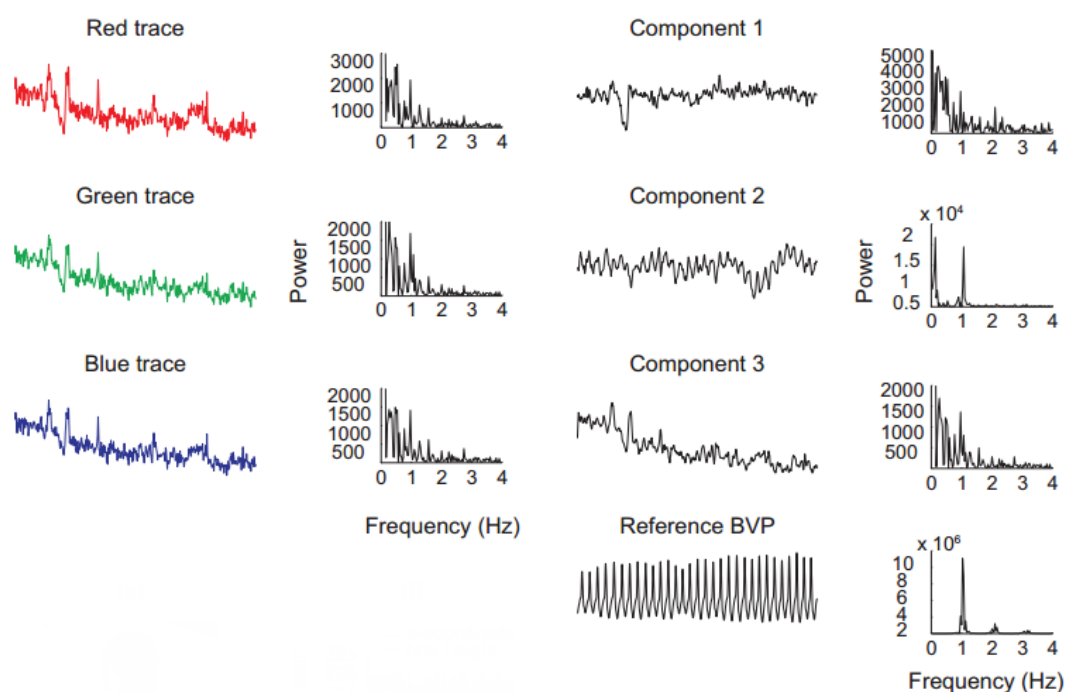
2.1. Non-contact, automated cardiac pulse measurements using video imaging and blind source separation

(Poh et al., 2010: 10762-10774) เสนอวิธีการวัดชีพจรแบบไม่สัมผัสกับร่างกายโดยพิจารณาจากภาพวิดีโอเฟรม โดยใช้หลักการการแยกแหล่งกำเนิดที่มองไม่เห็น (blind source separation) ด้วยวิธี Independence Component Analysis (ICA) และวิเคราะห์การเปลี่ยนแปลงความถี่ของสัญญาณด้วย Fast-Fourier Transform (FFT) เพื่อวัดค่าสัญญาณชีพ รูปที่ 1 แสดงหลักการทำงานของระบบการวัดชีพจรที่เสนอโดย Poh เริ่มจากการเลือกส่วนที่สนใจ (Region of Interest : ROI) จากวิดีโอเฟรม โดยเลือกส่วนของใบหน้า แบ่งวิดีโอเฟรมออกเป็นหน้าต่างโดยแต่ละหน้าต่างมีขนาด 20 วินาที และซ้อนทับสัญญาณ (Overlap) แต่ละหน้าต่าง 1 วินาที นำสัญญาณที่ได้แต่ละแขนงมา normalize ให้มีค่าเฉลี่ยเป็น 0 และส่วนเบี่ยงเบนมาตรฐานเป็น 1 หลังจากนั้นนำสัญญาณไปวิเคราะห์ Independence Component Analysis (ICA)



รูปที่ 1 แสดงสัญญาณที่ได้จากแต่ละเฟรม a) วิดีโอเฟรมต้นฉบับ b) แยกสัญญาณวิดีโอเฟรมออกเป็น 3 แชนแนล c) สัญญาณแต่ละแขนงของ RGB d) สัญญาณ ICA ทั้ง 3 องค์ประกอบ

สัญญาณที่ได้จะนำมาวิเคราะห์ความถี่ด้วย Fast-Fourier Transform (FFT) โดยเลือกวิเคราะห์ความถี่ช่วงความถี่ 0.75 – 4 Hz (45 – 240 bpm) เท่านั้นดังแสดงดังรูปที่ 2 ซึ่งเป็นความถี่การเปลี่ยนแปลงของสัญญาณชีพ เมื่อนำมาเปรียบเทียบกับสัญญาณอ้างอิงจาก blood volume pulse (BVP, รูปที่ 3) พบว่าสัญญาณองค์ประกอบที่ 2 ของการทำ ICA มีค่าใกล้เคียงกับสัญญาณอ้างอิง ดังนั้นงานวิจัยนี้จึงเลือกวัดค่าชีพจรจากองค์ประกอบที่ 2 ของการทำ ICA เพียงองค์ประกอบเดียว



รูปที่ 2 แสดงผลการวิเคราะห์ FFT ของสัญญาณ ICA แต่ละองค์ประกอบ



รูปที่ 3 อุปกรณ์วัดชีพจร BVP

ที่มา : NeXus Blood Volume Pulse Sensor, accessed April 20, 2016, available from <http://www.ycanaustralia.com/BLOOD-VOLUME-PULSE-SENSOR-FOR-NEXUS-4-10-32F%20>

Poh เสนอวิธีการจำกัดข้อผิดพลาดในการระบุค่าชีพจรจากแต่ละหน้าต่าง โดยการนำผลลัพธ์มาเปรียบเทียบกับหน้าต่างก่อนหน้า ถ้าชีพจรในหน้าต่างปัจจุบันต่างจากหน้าต่างก่อนหน้ามากกว่า 12 bpm ระบบจะวิเคราะห์ความถี่ในลำดับถัดไปที่มีค่า Magnitude ต่ำกว่าจนพบความถี่ที่ตรงตามเงื่อนไข ถ้าไม่มีความถี่ใดใกล้เคียงกับชีพจรจากหน้าต่างก่อนหน้า ระบบจะนำผลลัพธ์ชีพจรจากหน้าต่างก่อนหน้ามาเป็นผลลัพธ์ชีพจรในหน้าต่างปัจจุบัน เรียกว่า Error suppression

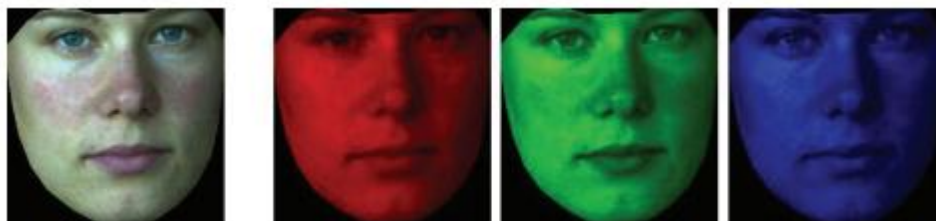
(Poh et al., 2011: 7-11) เสนอการปรับปรุงงานวิจัยใหม่ในปี 2011 โดยปรับปรุงการเลือกองค์ประกอบที่จะใช้เป็นตัวแทนของสัญญาณชีพจร เนื่องจากพบว่า การเลือกองค์ประกอบที่ 2 ของการวิเคราะห์ ICA เป็นตัวแทนสัญญาณชีพจรทำให้การระบุค่าชีพจรผิดพลาด ดังนั้นจึงปรับปรุงการระบุค่าชีพจรโดยพิจารณาจากค่า Magnitude จากทั้ง 3 องค์ประกอบของการทำ ICA และเลือกค่าความถี่ที่มีค่า Magnitude สูงสุด

2.2. Noncontact Automatic Heart Rate Analysis in Visible Spectrum by Specific Face Regions

(Dragos Datcu, Marina Cidota, Stephan Lukosch, and Leon Rothkrantz. 2013: 120-127) นำงานวิจัยของ Poh มาพัฒนาต่อโดยปรับปรุงการเลือกส่วนสนใจ (ROI) ซึ่งเป็นส่วนของใบหน้า โดยที่งานของ Poh จะเลือกตัดส่วนเป็นสี่เหลี่ยมทำให้ได้ภาพใบหน้าไม่ครบถ้วนและอาจได้ภาพบริเวณส่วนพื้นหลัง (Background region) ที่ไม่ใช่ใบหน้าได้ ดังนั้นงานวิจัยนี้จึงเลือกส่วนใบหน้าด้วยหลักการ Active appearance model (AAM) เป็นหลักการทางสถิติที่จะพิจารณาการเปลี่ยนแปลงของรูปร่างและพื้นผิว (Shape and texture variation) ดังรูปที่ 4 และผลลัพธ์การเลือกส่วนใบหน้า (Face segmentation) แสดงดังรูปที่ 5



รูปที่ 4 แสดงผลลัพธ์การเลือกตำแหน่งใบหน้าการทำ AAM

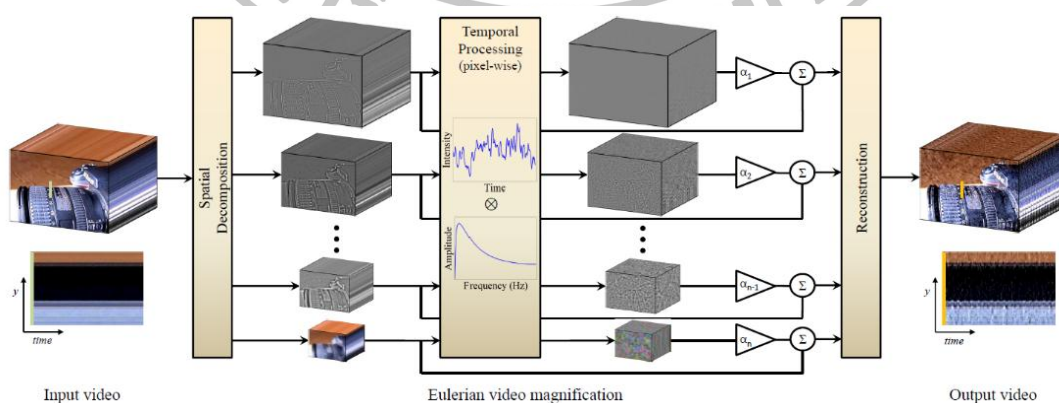


รูปที่ 5 ผลลัพธ์การเลือกส่วนที่สนใจด้วย AAM และแยกออกเป็น 3 แชนแนล

วิดีโอเฟรมใบหน้าที่ได้จะนำมาหาค่าเฉลี่ยในแต่ละแชนแนลและนำสัญญาณค่าสีมา Normalization และกรองความถี่ในช่วง $0.75 - 4$ Hz สัญญาณที่ได้จะนำมาวิเคราะห์ ICA และคำนวณความถี่ด้วย FFT โดยเลือกวิเคราะห์ความถี่เฉพาะองค์ประกอบที่ 2 ของการทำ ICA เท่านั้น ผลลัพธ์จากแต่ละหน้าต่างจะนำมาเปรียบเทียบกับหน้าต่างก่อนหน้าเช่นเดียวกับวิธีของ Poh

2.3. Eulerian Video Magnification for Revealing Subtle Changes in the World

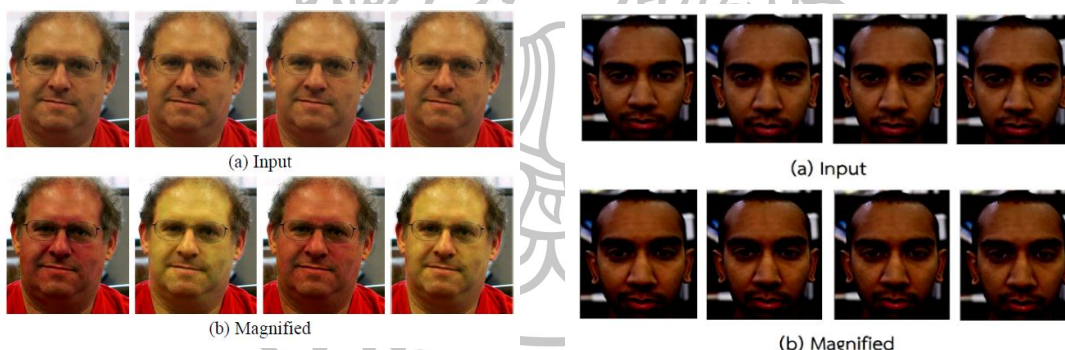
Eulerian Video Magnification (EVM) ถูกเสนอโดย (Hao-Yu Wu, et al. 2012: 651-658) เป็นวิธีการวัดอัตราการเต้นของหัวใจโดยวิเคราะห์จากภาพวิดีโอเฟรมอีกวิธีหนึ่ง วิธีนี้สามารถแสดงการเคลื่อนไหวเล็กน้อยที่ไม่สามารถมองเห็นได้ด้วยตาเปล่าแบ่งเป็น Motion magnification และ Color magnification หลักการของ EVM จะเป็นการวิเคราะห์ความถี่การเปลี่ยนแปลงและเน้นค่าความถี่ช่วงที่สนใจให้เด่นกว่าส่วนอื่น มีวิธีการดังรูปที่ 6



รูปที่ 6 Flow chart วิธีการ EVM ที่เสนอโดย Wu

1. Spatial decomposition จะเป็นการแบ่งภาพออกเป็นส่วนๆตามความถี่ของภาพ เพื่อเพิ่มค่า Signal to Noise Ratio (SNR) และเลือกส่วนที่สนใจ (Region of interest : ROI)
2. Temporal filtering เป็นการกรองความถี่สัญญาณให้อยู่ในช่วงความถี่ที่ต้องการ สำหรับการวัดชีพจรจะใช้ตัวกรองความถี่ 0.75 – 4 Hz (45 – 240 beat per minute)
3. เพิ่มค่าสัญญาณที่ผ่านตัวกรองด้วยการคูณค่า α เพื่อเน้นการเปลี่ยนแปลงในช่วงความถี่ที่สนใจให้เด่นกว่าความถี่อื่นและแสดงการเปลี่ยนแปลงของสีที่เกิดกับวิดีโอ
4. รวมสัญญาณจากแต่ละความถี่ของภาพเพื่อสร้างภาพขึ้นใหม่ (Reconstruction)

วิดีโอที่ถูกวิเคราะห์ด้วยวิธี EVM สามารถแสดงการเปลี่ยนแปลงของสีผิวที่เกิดจากการไหลเวียนของเลือดได้ดังรูปที่ 7 แถวบนเป็นวิดีโอเฟรมต้นฉบับ แถวล่างเป็นวิดีโอเฟรมที่ผ่านการทำ EVM จะเห็นว่ารูปด้านซ้ายล่างวิดีโอเฟรมเกิดการเปลี่ยนแปลงของสีเนื่องจากการไหลเวียนของเลือดชัดเจนกว่ารูปด้านขวาล่างจะเห็นการเปลี่ยนแปลงน้อยมาก เนื่องจากผู้ทำการทดลองมีสีผิวคล้ำกว่า



รูปที่ 7 ผลลัพธ์การวัดชีพจรด้วย EVM ด้านบนคือ วิดีโอเฟรมต้นฉบับ ด้านล่างคือ การแสดงการเปลี่ยนแปลงของสีผิว

ผลลัพธ์ของวิธี EVM สามารถนำไปวัดชีพจรที่เกิดจากการไหลเวียนของเลือดบนใบหน้าได้ ซึ่งงานวิจัยนี้เสนอข้อเสนอแนะในการบันทึกวิดีโอ ดังนี้ 1. ควบคุมการขยับของผู้ทำการทดลอง 2. เลือกใช้กล้องที่มีคุณภาพเช่นเซอร์ตี และจัดการแสงสว่างให้มากพอ 3. บันทึกวิดีโอให้มีความละเอียดและมีความยาวมากที่สุดเท่าที่จะเป็นไปได้ เพราะส่งผลให้ได้สัญญาณที่มีคุณภาพ และถ้าเป็นไปได้

เลือกบันทึกวิดีโอแบบไม่บีบอัด (Uncompressed video) จะส่งผลให้ได้สัญญาณที่ดีขึ้น สำหรับงานวิจัยนี้ประมวลผลด้วยเครื่องคอมพิวเตอร์ที่มีหน่วยประมวลผลกลาง (cpu) 6 core และ Ram 32 GB บันทึกวิดีโอผ่านแล็ปท็อป (laptop) ด้วย 45 frame per second

2.4. Preprocessing Realistic Video for Contactless Heart Rate Monitoring Using Video Amplification Methodologies

(Ahmed Alzahrani and Anthony Whitehead. 2015: 261-268) เสนอวิธีการทำ Pre-processing ก่อนทำ Video magnification ดังรูปที่ 8 โดยจะเปรียบเทียบการทำ Pre-processing 2 วิธีคือ

1. เลือกส่วนที่สนใจด้วยการเลือกส่วนใบหน้า (Face detection and tracking)
2. จากวิธี 1. นำวิดีโอเฟรมที่ได้มาตรวจจับสีผิวและลบส่วนมือ (Skin extraction and delete hands) โดยใช้ threshold ของค่าสี u^* และ v^* จากโมเดลสี CIE $L^*u^*v^*$ โดยสีผิวจะมีค่า $u > 12$ และ $v > 12$ และตัดการเคลื่อนไหวที่ไม่ต้องการด้วยวิธี matching stabilization โดยมี 3 ขั้นตอน คือ motion estimation, motion smoothing, and image composition

วิดีโอจะถูกบันทึกด้วย iPhone5 เป็นเวลา 2 นาที ขณะผู้ทำการทดลองวิ่งบนลู่วิ่ง และเปรียบเทียบผลลัพธ์ที่ได้กับนาฬิกา Garmin-Forerunner 410 วิดีโอเฟรมที่ได้จะแบ่งเป็น 6 ส่วน โดยไม่มีการ Overlap สัญญาณ (20 วินาทีต่อส่วน)



รูปที่ 8 Flow chart การวัดชีพจรที่เสนอโดย Ahmed Alzahrani

สัญญาณที่ได้จะนำมากรองความถี่ช่วง 0.83 – 1.67 Hz (50 – 100 bpm) และ 1.33 – 2.17 Hz (80 – 130 bpm) หลังจากนั้นนำวิดีโอเฟรมมาเลือกส่วนที่สนใจขนาด 25 pixel เพื่อลดปริมาณสัญญาณรบกวนที่ได้รับ งานวิจัยนี้เปรียบเทียบการวัดค่าชีพจร 2 วิธี คือ 1. นับจำนวนพีก (peak) จากกราฟการเปลี่ยนแปลงของสีแดงด้วยมนุษย์ 2. วิเคราะห์ความถี่การเปลี่ยนแปลงของสีแดงด้วย Fast Fourier Transform (FFT) ผลลัพธ์การวัดชีพจรของงานวิจัยนี้พบว่า การทำ Pre-processing ก่อนวัดชีพจรทั้ง 2 วิธี มีความผิดพลาดเฉลี่ยต่ำกว่าการไม่ทำ Pre-processing ด้วยวิธีใดเลย และการวัดชีพจรแบบวิเคราะห์ความถี่ด้วย FFT มีผลลัพธ์ที่ดีกว่าการนับจำนวนพีกของสัญญาณด้วยมนุษย์

นอกจากการทำ Pre-processing แล้วงานวิจัยนี้เสนอการควบคุมแสงด้วยการเปลี่ยนโมเดลสีหลังจากทำ Pre-processing แล้ว โดยแบ่งเป็น 2 ขั้นตอน คือ

ขั้นตอนที่ 1. เพิ่มค่าแสงในวิดีโอเฟรมโดยมี 3 ขั้นตอน

1.1 การแปลงโมเดลสี RGB เป็น HSV

1.2 เพิ่มค่า V ขึ้นจากเดิม 50%

1.3 แปลงโมเดลสีกลับเป็น RGB

ขั้นตอนที่ 2. เพิ่มการเปลี่ยนแปลงของสีแดงโดยเปรียบเทียบ 2 วิธีคือ

2.1 เพิ่มค่าสีแดงของโมเดลสี RGB ขึ้น 30%

2.2 แปลงโมเดลสี RGB เป็น CIE $L^*a^*b^*$ เพื่อทำให้วิดีโอเฟรมมีสีชมพูมากขึ้น โดยเพิ่มค่าแกนแนล L ด้วย 11 และลดค่าแกนแนล b^* ด้วย 11 และแปลงโมเดลสีกลับเป็น RGB

งานวิจัยนี้เลือกใช้คอมพิวเตอร์ที่มี CPU เป็น 3.4 GHz Quad Core และ Ram 12 GB พร้อมสรุปสิ่งที่เสนอเพื่อเพิ่มความแม่นยำให้กับวิธีการวัดชีพจรแบบ EVM ทั้งหมด 5 กรณีคือ

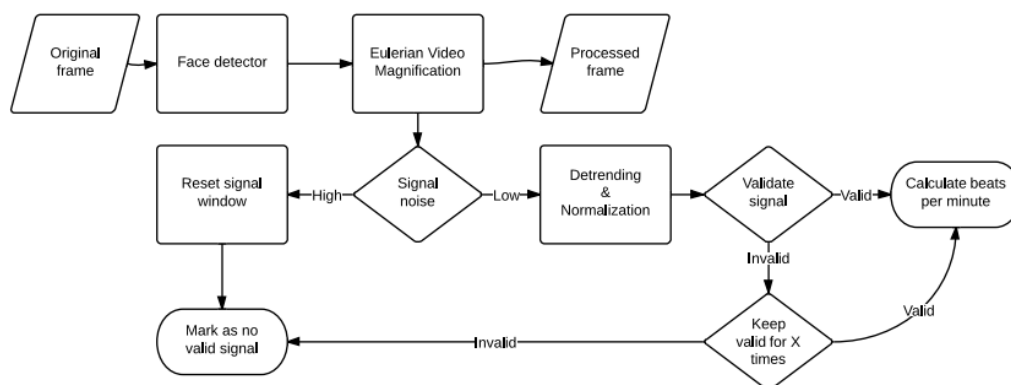
1. การวัดชีพจรด้วยวิธี EVM โดยไม่ทำ Pre-processing จะทำให้ความแม่นยำลดลง เนื่องจากการขยับตัวของผู้ทำการทดลองจะทำให้การเปลี่ยนแปลงของสีผิวไม่ชัดเจน
2. การเลือกตำแหน่งอวัยวะมีผลกระทบต่อความแม่นยำ เนื่องจากอวัยวะแต่ละส่วนได้รับผลกระทบจากแสงไม่เท่ากัน

3. จากกรณีที่ 1 งานวิจัยนี้จึงเพิ่มค่าความสว่างของวิดีโอเฟรมด้วยการเพิ่มค่า V ของโมเดลสี HSV ขึ้น 50 %
4. งานวิจัยนี้ศึกษาการเปลี่ยนแปลงของสีแดงซึ่งพบว่าเมื่อความสว่างไม่เพียงพอหรือสีผิวคล้ำจะส่งผลต่อความแม่นยำ ดังนั้นจึงเพิ่มการเปลี่ยนแปลงของสีแดงโดยการเพิ่มค่าแชนแนล R ของโมเดลสี RGB ขึ้น 30 %
5. จากกรณีที่ 3 เปลี่ยนวิธีการเพิ่มการเปลี่ยนแปลงของสีแดงโดยแปลงโมเดลสีเป็น CIE $L^*a^*b^*$ และเพิ่มค่าแชนแนล L^* ด้วย 11 และลดค่าแชนแนล b^* ด้วย 11 หลังจากนั้นแปลงโมเดลสีกลับเป็น RGB

2.5. Android-based implementation of Eulerian Video Magnification for vital signs monitoring

(Pedro B. Chambino. 2014) สร้าง application สำหรับวัดชีพจรด้วยวิธี EVM บน Android แบบ real-time รูปที่ 9 แสดง flow chart ของระบบ เริ่มจากการตรวจจับใบหน้า (Face detection) เพื่อเลือกส่วนที่สนใจแล้วนำไปทำ EVM และคำนวณชีพจร หลังจากนั้นนำสัญญาณไปตรวจสอบสัญญาณรบกวนโดยคำนวณจากค่าส่วนเบี่ยงเบนมาตรฐานของค่าเฉลี่ยของแชนแนล G ของโมเดลสี RGB ในแต่ละเฟรม ถ้าค่าส่วนเบี่ยงเบนมาตรฐานมีค่ามากกว่าครึ่งหนึ่งของค่า α ที่จะนำมาคูณสัญญาณเพื่อนเน้นความถี่ส่วนที่สนใจ สัญญาณนั้นจะไม่ถูกนำมาคำนวณต่อและระบบจะกำหนดหน้าต่างในการตรวจจับใบหน้าใหม่ หากมีสัญญาณรบกวนน้อยระบบจะนำสัญญาณไปผ่านตัวกรอง, Normalized และตรวจสอบสัญญาณ (Validate signal) ด้วยการวิเคราะห์ที่รูปร่างแล้ว ช่วงเวลาการเกิดสัญญาณโดยแบ่งสัญญาณออกเป็นช่วงเล็กๆ ช่วงละ 200 มิลลิวินาทีและหาค่ามากที่สุด ซึ่งค่ามากที่สุดจะต้องไม่อยู่ที่ขอบของสัญญาณหรือต่ำกว่าค่า threshold ที่กำหนด และพีคของสัญญาณจากแต่ละหน้าต่างจะต้องห่างกันมากกว่า 200 มิลลิวินาที

การทดสอบระบบงานวิจัยนี้บันทึกวิดีโอโดยขนาดของแต่ละหน้าต่างจะมีความยาว 5 วินาที (20 fps) และ overlap 30 frame ดังนั้นสำหรับงานวิจัยนี้ที่เลือกใช้การตรวจจับใบหน้าก่อนวัดชีพจรด้วย EVM คล้ายกับงานวิจัยอื่น แต่เลือกไม่คำนวณเฟรมที่มีสัญญาณรบกวนมากเพื่อลดความเสี่ยงที่จะวัดชีพจรพลาด ทำให้ต้องตรวจจับใบหน้าใหม่จึงส่งผลให้ผู้ทำการทดลองต้องอยู่นิ่งมากๆ และประสิทธิภาพของโทรศัพท์มือถือมีผลต่อความเร็วในการวิเคราะห์



รูปที่ 9 Flow chart การวัดชีพจรที่เสนอโดย Chambino

2.6. สรุปปัจจัยที่มีผลต่อการวัดชีพจรโดยไม่สัมผัสร่างกาย

2.6.1 กล้องและโมเดลสี

งานวิจัยที่เกี่ยวข้องกับการวัดชีพจรโดยไม่สัมผัสกับร่างกายพิจารณาการเปลี่ยนแปลงของค่าสี ซึ่งมีการใช้โมเดลสีหลายโมเดล เช่น YIQ, RGB, HSV, CIE L*a*b* และ RGBCO ซึ่งโมเดลสีหลักที่ใช้ในการคำนวณการเปลี่ยนแปลงคือ RGB โดยโมเดลสีอื่นถูกใช้เพื่อปรับปรุงค่าสีหรือเน้นค่าสีเท่านั้น กล้องที่ใช้ในการบันทึกวิดีโอมีหลากหลายแบบเช่น กล้อง webcam กล้องโทรศัพท์มือถือ กล้องความเร็วสูง Pike และกล้อง DSLR ดังนั้นงานวิจัยนี้จึงศึกษาผลกระทบจากการเปลี่ยนโมเดลสีที่ใช้ในการวัดค่าชีพจร

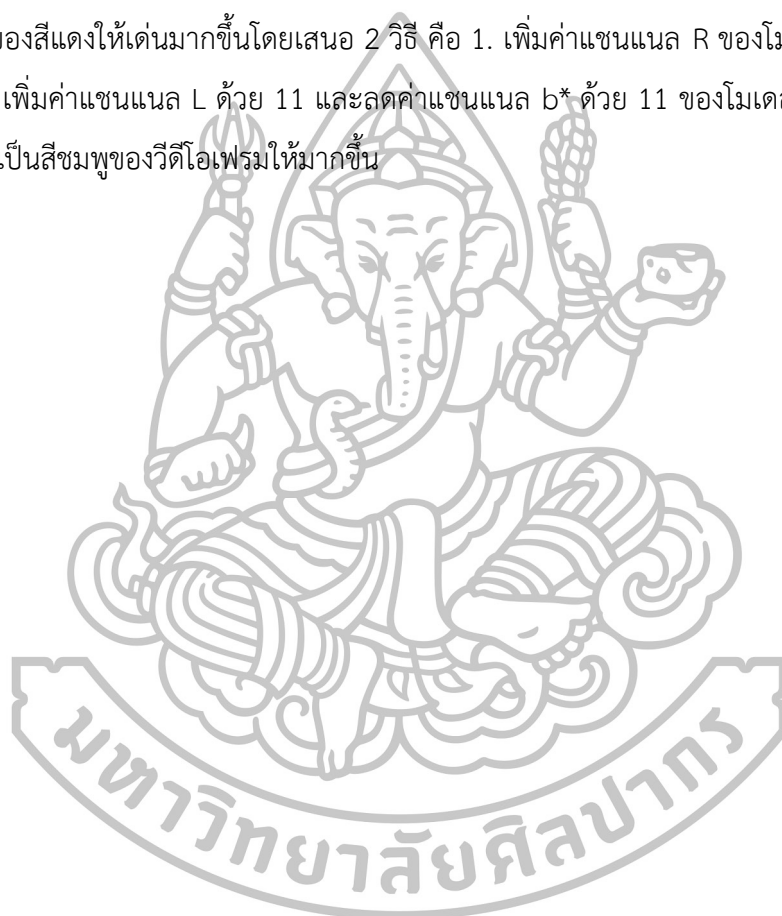
2.6.2 ขนาดหน้าต่างและ framerate

ขนาดหน้าต่าง (Window size) และการซ้อนทับสัญญาณ (Overlap) สัญญาณที่ใช้ในการแบ่งสัญญาณออกเป็นหน้าต่างมีความแตกต่างกันในแต่ละงานวิจัยคือ 4 วินาที และ 0.5 วินาที, 5 วินาที และ 1 วินาที, 20 วินาที และไม่ overlap สัญญาณ และ 30 วินาที และ 1 วินาที ค่าเฟรมเรต (Framerate, Frame per second : FPS) ที่ใช้เป็น 15 fps, 20 fps, 30 fps และ 45 fps ซึ่งค่า Framerate ที่ใช้ไม่ควรต่ำกว่า 15 fps

สำหรับขนาดหน้าต่างที่มีค่าต่างกันโดยที่น้อยที่สุดคือ 4 วินาทีต่อหน้าต่าง และมากที่สุด 30 วินาทีต่อหน้าต่าง แต่ไม่มีงานใดศึกษาเกี่ยวกับผลของการเปลี่ยนขนาดหน้าต่างที่ส่งผลต่อความแม่นยำ ดังนั้นงานวิจัยนี้จึงศึกษาการปรับเปลี่ยนขนาดหน้าต่างที่ใช้ในการแบ่งสัญญาณ

2.6.3 สีผิวและแสงสว่าง

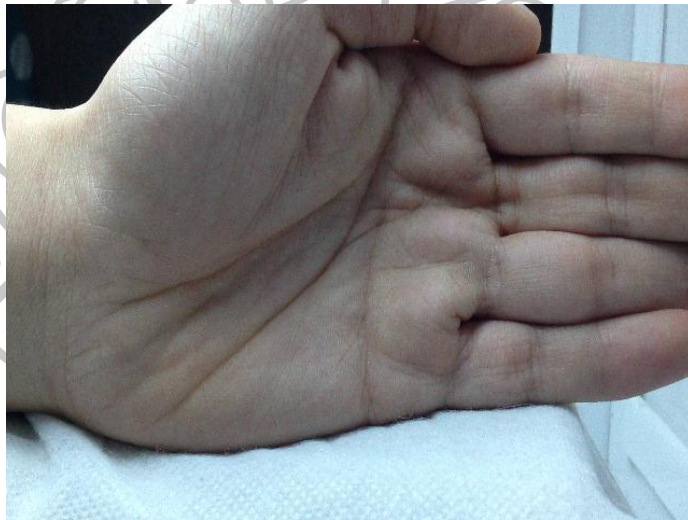
งานวิจัยที่ศึกษามายืนยันเหมือนกันว่าแสงสว่างมีผลต่อวิธีการวัดสีพจรแบบไม่สัมผัส เนื่องจากการสะท้อนของแสงที่มากไปอาจบดบังการเปลี่ยนแปลงของสีผิวได้ และการวัดสีพจรกับผู้ทำการทดลองที่ผิวสีคล้ำนั้นมีความแม่นยำที่ต่ำกว่าผิวสีขาว งานวิจัย ในหัวข้อ 2.3 จึงเสนอแนวทางแก้ปัญหา คือ การเพิ่มค่าความสว่างด้วยการเพิ่มค่าแชนแนล V ของโมเดลสี HSV ขึ้น 50% เพิ่มสัดส่วนของแสงสว่างที่ขาดไป ส่วนการวัดสีพจรกับผู้ทำการทดลองที่ผิวสีคล้ำนั้นจะเน้นการเปลี่ยนแปลงของสีแดงให้เด่นมากขึ้นโดยเสนอ 2 วิธี คือ 1. เพิ่มค่าแชนแนล R ของโมเดลสี RGB ขึ้น 30% และ 2. เพิ่มค่าแชนแนล L ด้วย 11 และลดค่าแชนแนล b^* ด้วย 11 ของโมเดลสี CIE $L^*a^*b^*$ เพื่อเพิ่มความเป็นสีชมพูของวีดีโอเฟรมให้มากขึ้น



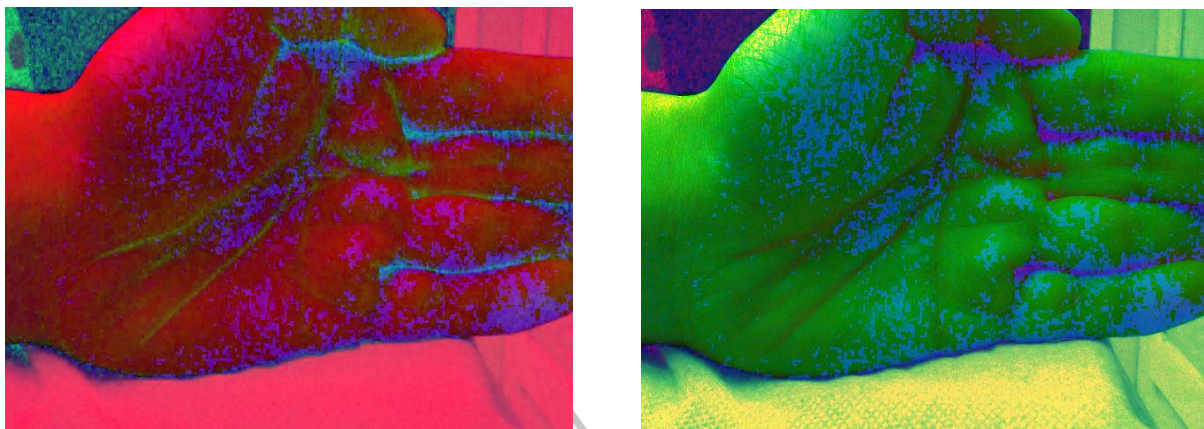
บทที่ 3 ทฤษฎีที่เกี่ยวข้อง

3.1. Color model

งานวิจัยนี้ศึกษาผลกระทบจากการเปลี่ยนและผสมโมเดลสีประกอบด้วย RGB, HLS, HSV, CIE L*a*b*, CIE L*u*v*, TSL, YCbCr และนำโมเดลสีมาผสมกันเรียกว่า Color-model Union (CMU) และ Multi-model Summation (MMS) รูปที่ 10 เป็นวิดีโอเฟรมต้นฉบับที่ได้จากวิดีโอที่บันทึก ซึ่งมีค่าสีเป็นโมเดลสี RGB ผลการแปลงค่าสีจากโมเดลสี RGB เป็น HLS, HSV, CIE L*a*b*, CIE L*u*v*, TSL และ YCbCr แสดงดังรูปที่ 11 - รูปที่ 13



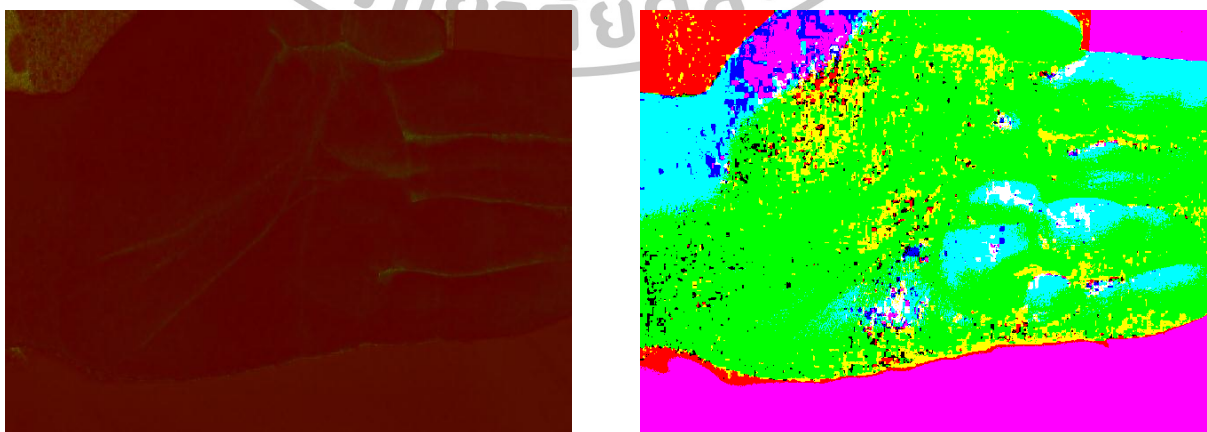
รูปที่ 10 วิดีโอเฟรมต้นฉบับ โมเดลสี RGB



รูปที่ 11 วิดีโอเฟรมที่แปลงโมเดลสีเป็น HSV (ซ้าย) และ HLS (ขวา)



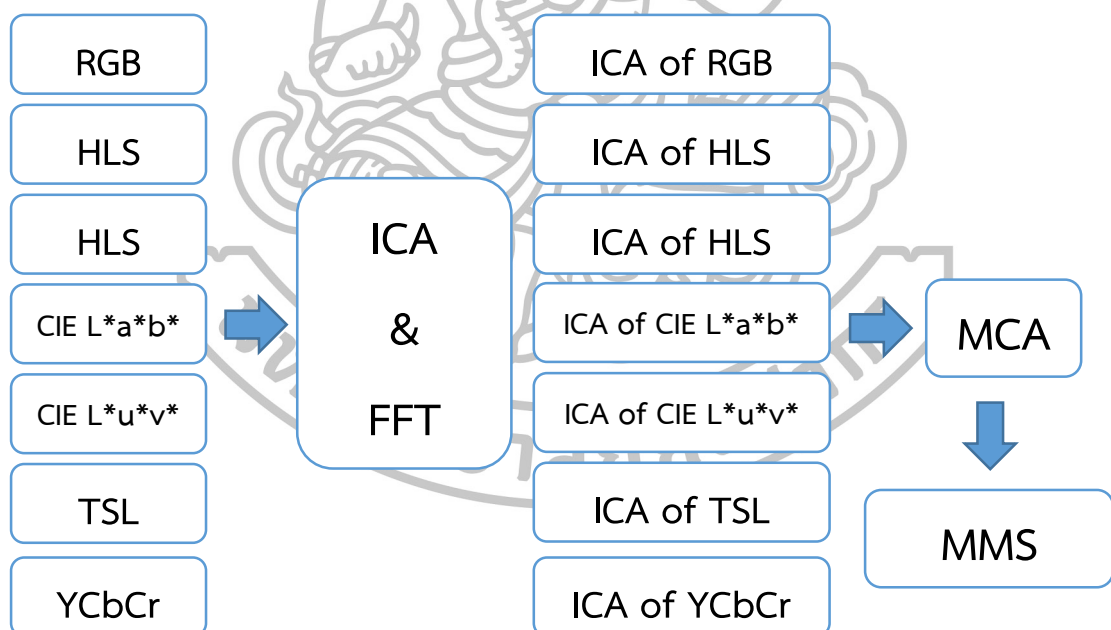
รูปที่ 12 วิดีโอเฟรมที่แปลงโมเดลสีเป็น CIE L*a*b* (ซ้าย) และ CIE L*u*v* (ขวา)



รูปที่ 13 วิดีโอเฟรมที่แปลงโมเดลสีเป็น TSL (ซ้าย) และ YCbCr (ขวา)

Color-model Union (CMU) เป็นโมเดลสีที่นำโมเดลสีทั้ง 7 โมเดลสีมาผสมกันโดยตัด แชนแนล (Channel) ที่เหมือนกันออก คือ แชนแนล H ของโมเดลสี HLS และ HSV แชนแนล L ของ โมเดลสี CIE L*a*b* และ CIE L*u*v* เพราะฉะนั้นแชนแนลของโมเดลสี CMU จะมีทั้งหมด 19 แชนแนล คือ แชนแนล R, G และ B จากโมเดลสี RGB, แชนแนล H, L และ S จากโมเดลสี HLS, แชนแนล S และ V จากโมเดลสี HSV, แชนแนล L, a และ b จากโมเดลสี CIE L*a*b*, แชนแนล u และ v จากโมเดลสี CIE L*u*v*, แชนแนล T, S และ L จากโมเดลสี TSL, และแชนแนล Y, Cb และ Cr จากโมเดลสี YCbCr (RGBHLSSVLabuvTSLYCbCr)

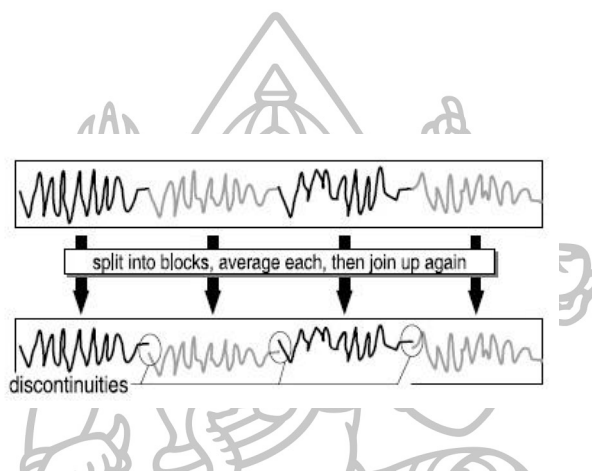
ผลลัพธ์การคำนวณค่าความถี่ด้วย Fast-Fourier Transform (FFT) ของสัญญาณการ เปลี่ยนแปลงสีที่ผ่านการคำนวณ Independence Component Analysis (ICA) ทั้ง 7 โมเดลคือ RGB, HLS, HSV, CIE L*a*b*, CIE L*u*v*, TSL และ YCbCr จะนำมารวมค่า Magnitude ด้วยวิธี Multi-component accumulation (MCA) เรียกว่า MMS ดังรูปที่ 14



รูปที่ 14 แสดงวิธีการสร้างโมเดล MMS

3.2. Frame blocking & Normalization

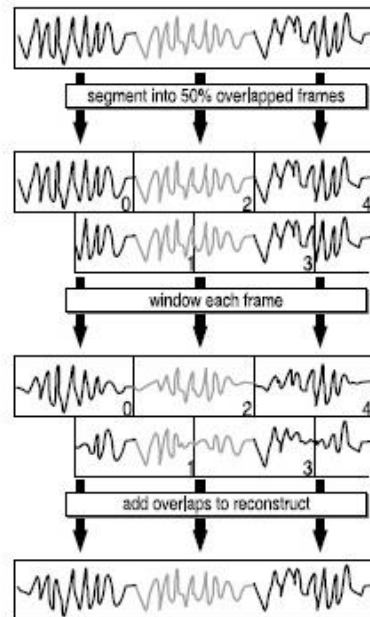
เนื่องจากสัญญาณภาพวิดีโอที่บันทึกมีความยาวมาก (ประมาณ 60 วินาที/วิดีโอ) จึงแบ่งสัญญาณออกเป็นช่วงเล็กๆ เพื่อเพิ่มความละเอียดในการวิเคราะห์แต่ละช่วง เรียกวิธีการนี้ว่า Frame Blocking จากรูปที่ 15 แสดงการตัดแบ่งช่วงของสัญญาณโดยไม่มีการซ้อนทับสัญญาณ (Overlap) จะสังเกตเห็นว่าสัญญาณที่สร้างขึ้นใหม่ (Reconstruct) จะไม่ต่อเนื่องกัน ในกรณีที่ Overlap สัญญาณพบว่าเมื่อสร้างสัญญาณใหม่จะมีความต่อเนื่องมากขึ้นและใกล้เคียงกับสัญญาณต้นฉบับดังแสดงดังรูปที่ 16



รูปที่ 15 การตัดแบ่งช่วงสัญญาณ

ที่มา : Ian Mccloughlin, Applied Speech and Audio Processing with MATLAB Examples,
New York: Cambridge University Press, 2009

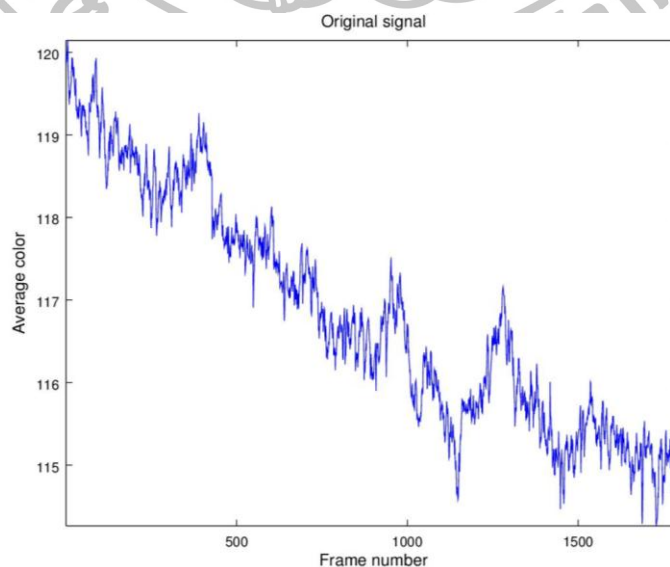




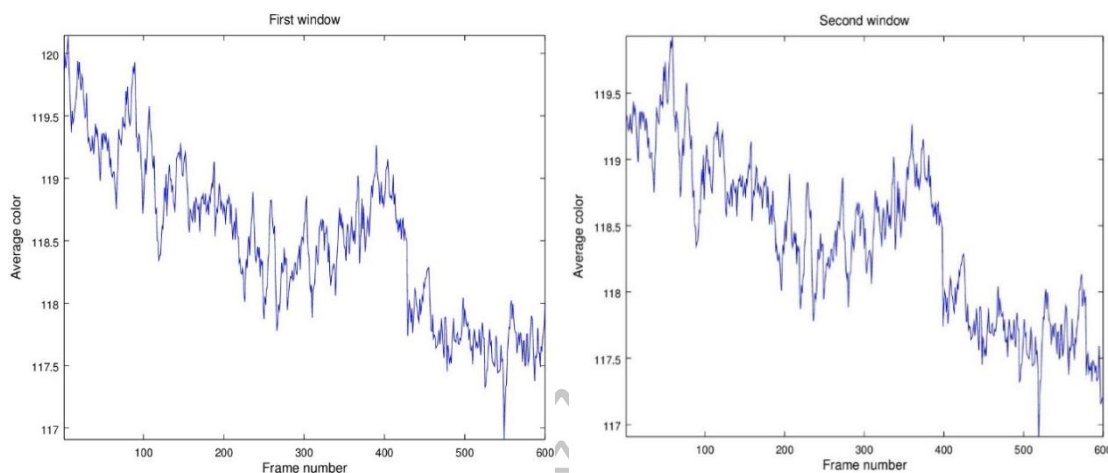
รูปที่ 16 การตัดแบ่งช่วงสัญญาณโดยมีการซ้อนทับสัญญาณ

ที่มา : Ian Mccloughlin, Applied Speech and Audio Processing with MATLAB Examples,
New York: Cambridge University Press, 2009

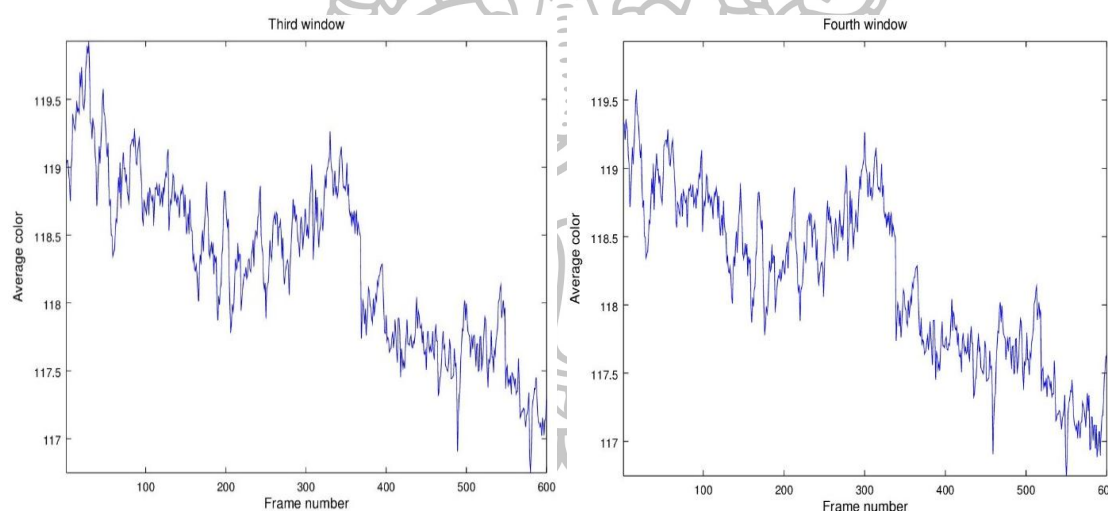
รูปที่ 17 แสดงสัญญาณก่อนการแบ่งมีขนาด 1800 frame และตัวอย่างการแบ่งสัญญาณ เป็นหน้าต่างขนาด 20 วินาทีต่อหน้าต่าง(600 frame/window) โดยซ้อนทับสัญญาณ (Overlap) 1 วินาที มีทั้งหมด 11 หน้าต่าง โดยแสดงเฉพาะ 4 หน้าต่างแรกดังรูปที่ 18 และรูปที่ 19



รูปที่ 17 สัญญาณต้นฉบับ ก่อนการแบ่งสัญญาณ



รูปที่ 18 แบ่งสัญญาณด้วยขนาดหน้าต่าง 20 วินาที/หน้าต่าง (600 frame/window) ซ้ายคือ หน้าต่างที่ 1 และขวาคือหน้าต่างที่ 2



รูปที่ 19 แบ่งสัญญาณด้วยขนาดหน้าต่าง 20 วินาที/หน้าต่าง (600 frame/window) ซ้ายคือ หน้าต่างที่ 3 และขวาคือหน้าต่างที่ 4

สัญญาณแต่ละหน้าต่างจะนำมาทำนอร์มัลไลเซชัน (Normalization) ให้มีค่าเฉลี่ยเป็น 0 และส่วนเบี่ยงเบนมาตรฐานเป็น 1 ดังสมการที่ 1 รูปที่ 20 แสดงสัญญาณค่าเฉลี่ยสีของโมเดลสี RGB โดยใช้ขนาดหน้าต่าง 20 วินาที (600 frame) โดยแถวบนคือ สัญญาณแชนแนลที่ 1 (แชนแนล R), แถวที่สองคือ สัญญาณแชนแนลที่ 2 (แชนแนล G) และ แถวที่สามคือ สัญญาณแชนแนลที่ 3 (แชนแนล B) และผลลัพธ์การนอร์มัลไลซ์ (Normalize) แสดงดังรูปที่ 21

$$x'_i(t) = \frac{x_i(t) - \mu}{\sigma}$$

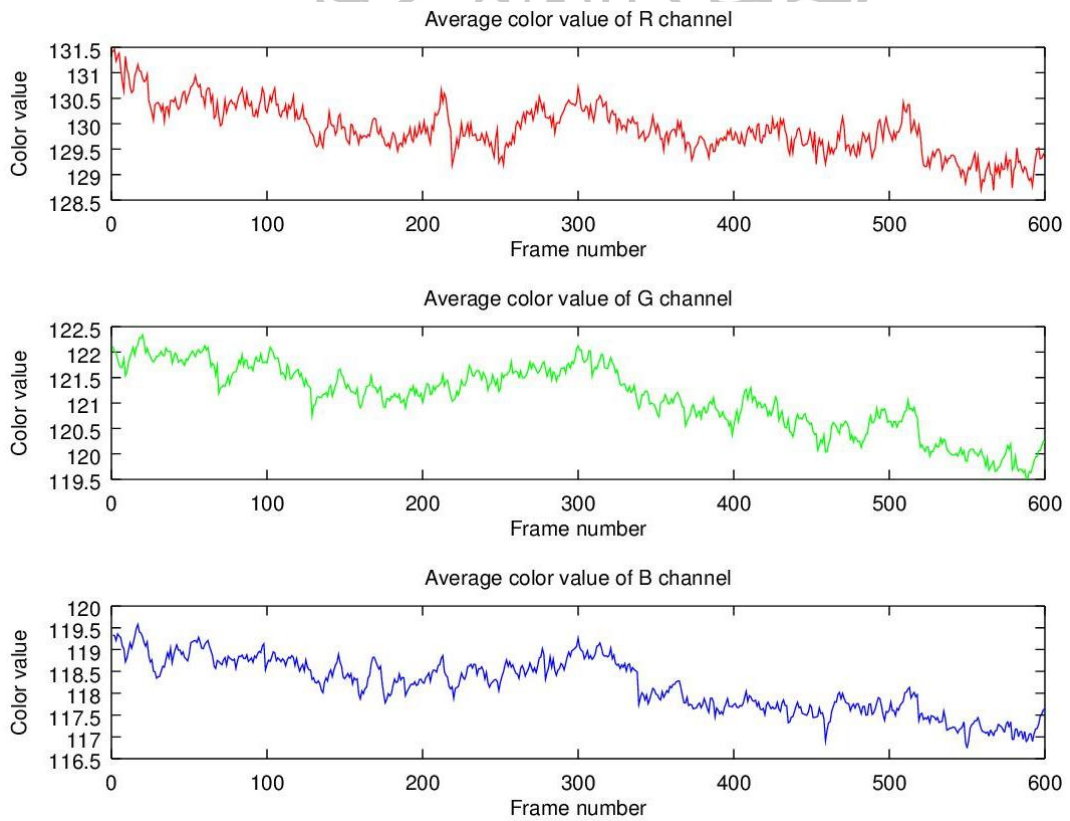
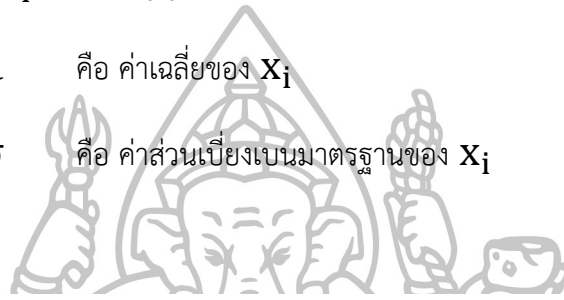
สมการที่ 1 สมการการ Normalization

โดย X_i คือ สัญญาณต้นฉบับ

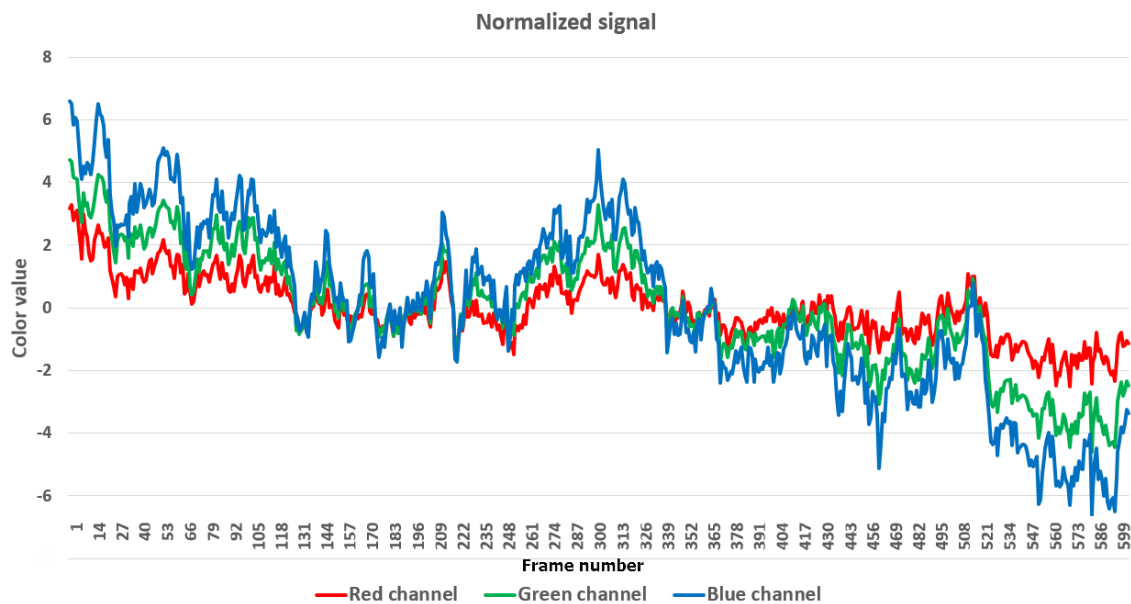
X'_i คือ สัญญาณที่ Normalize แล้ว

μ คือ ค่าเฉลี่ยของ X_i

σ คือ ค่าส่วนเบี่ยงเบนมาตรฐานของ X_i



รูปที่ 20 สัญญาณต้นฉบับทั้ง 3 แชนแนล

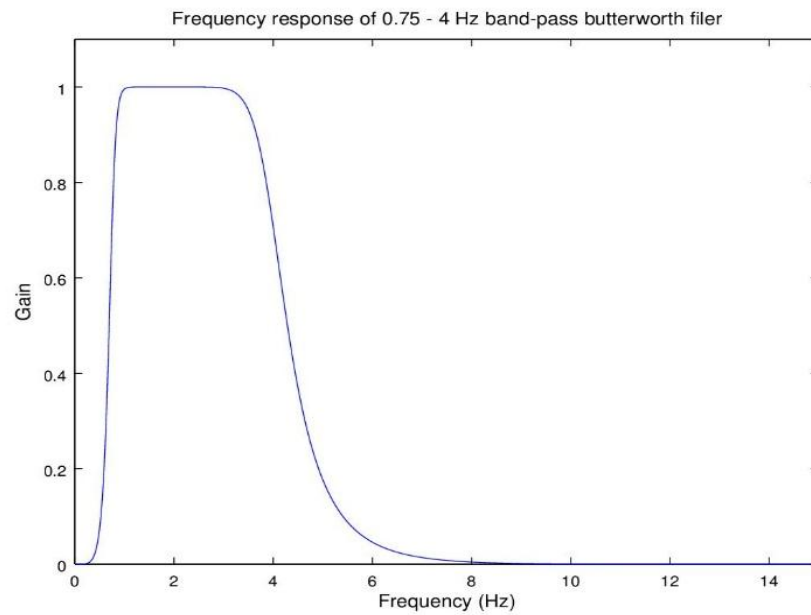


รูปที่ 21 สัญญาณทั้ง 3 แชนแนลที่ผ่านการ Normalized

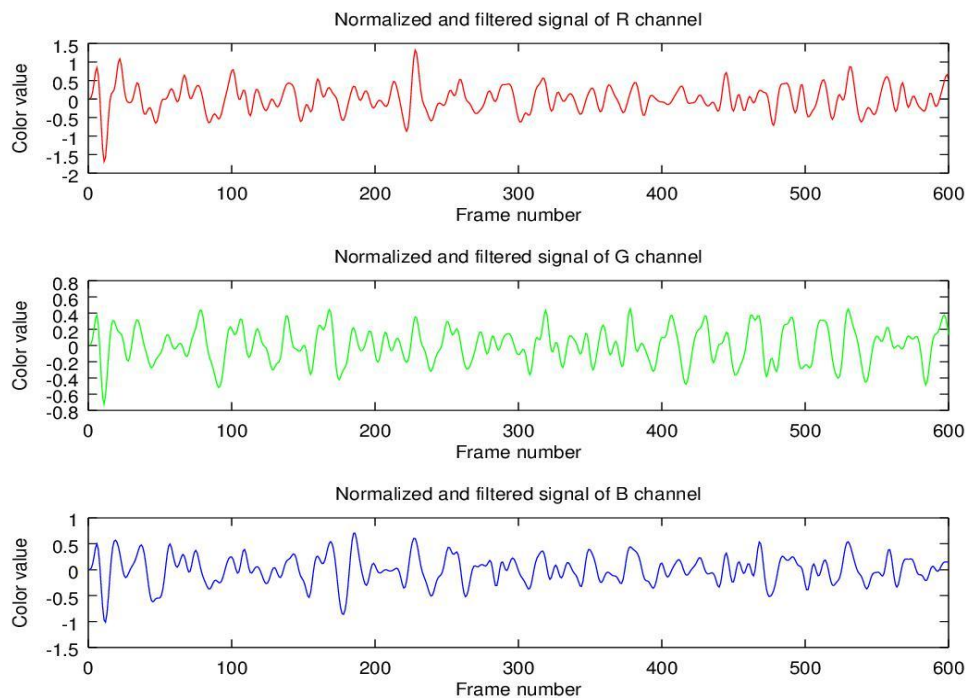
3.3. Band-pass filter

การกรอง (Filtering) คือ การยอมให้ความถี่ช่วงความถี่ใดความถี่หนึ่งเท่านั้นที่สามารถผ่านเข้ามาได้ ซึ่งลักษณะเฉพาะของตัวกรองจะสามารถแบ่งออกเป็นสามส่วนสำคัญ คือ ส่วนแถบความถี่ผ่าน (Pass-Band) ช่วงการเปลี่ยนสถานะ (Transition) และส่วนแถบความถี่หยุด (Stop-Band) (Robert J. Schilling, Sandra L. Harris. Fundamentals of digital signal processing using MATLAB, 2012) การออกแบบตัวกรองความถี่มีหลากหลายแบบ เช่น Butterworth filter, Chebyshev filter, Elliptic filter

งานวิจัยนี้ใช้ ตัวกรองความถี่ผ่าน (Band-pass Filter) แบบ Butterworth filter โดยกำหนดค่าความถี่ cut-off (Cut-off frequency) เป็น $0.75 - 4$ Hz ซึ่งมีค่าเท่ากับความถี่การเต้นของชีพจรคือ $45 - 240$ bpm รูปที่ 22 แสดงการตอบสนองความถี่ของ Butterworth Band-pass filter สัญญาณที่ผ่านการ Normalized และกรองความถี่แล้วแสดงดังรูปที่ 23



รูปที่ 22 การตอบสนองความถี่ของ Butterworth Band-pass filter มีความถี่ cut-off ที่ 0.75 - 4 Hz



รูปที่ 23 สัญญาณทั้ง 3 แชนแนล ที่ผ่านการ Normalized และกรองความถี่ ของสัญญาณในรูปที่ 20

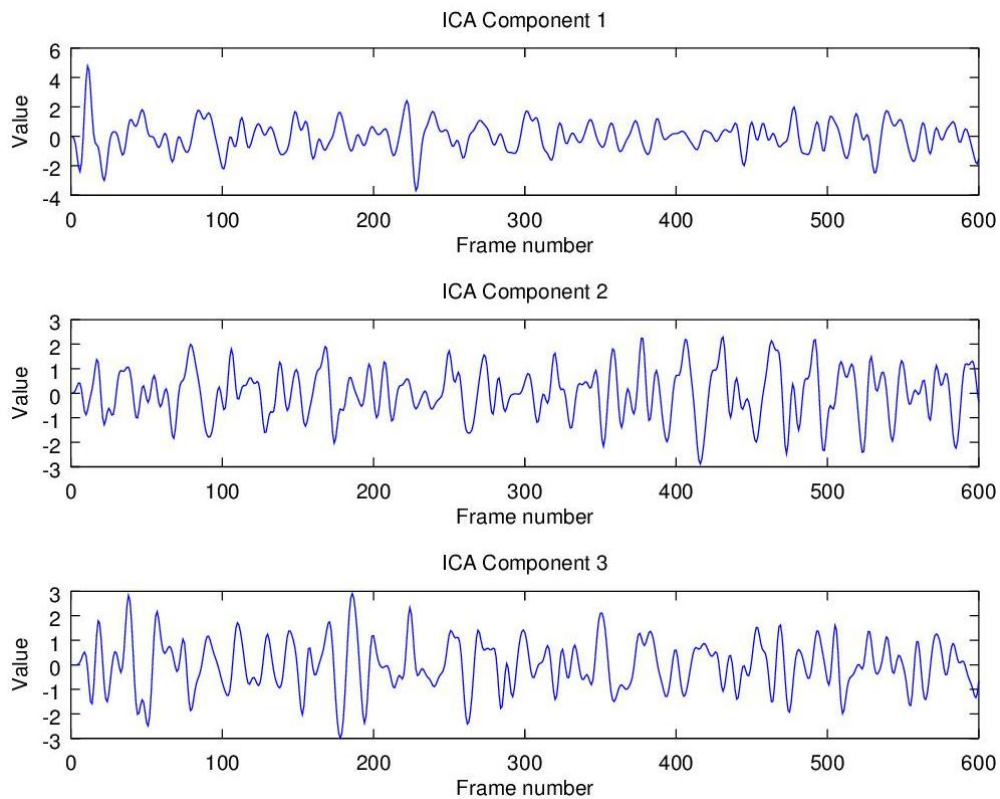
3.4. Independence Component Analysis (ICA)

Independence Component Analysis (ICA) เป็นเทคนิคที่ใช้จำแนกสัญญาณต้นกำเนิดที่มองไม่เห็น (blind source separation) ออกจากสัญญาณที่บันทึกได้ซึ่งเป็นสัญญาณที่เกิดการผสมของสัญญาณแล้วเท่านั้น ตัวอย่างเช่น สัญญาณ ECG (Electrocardiograph) ที่บันทึกการเปลี่ยนแปลงของศักย์ไฟฟ้าที่ผิวของร่างกายเกิดจากการเต้นของหัวใจ แต่สัญญาณที่บันทึกได้อาจเกิดการผสมค่าศักย์ไฟฟ้ามากกว่าหนึ่งค่า การพิจารณาค่าชีพจรควรแยกสัญญาณคลื่นไฟฟ้าหัวใจออกจากสัญญาณที่บันทึกได้ ICA สามารถแยกสัญญาณเหล่านั้นได้แม้จะวิเคราะห์สัญญาณที่เกิดจากการผสมแล้ว

การบันทึกวิดีโอเพื่อใช้ในการวัดค่าชีพจรแบบไม่สัมผัสสร้างการ ค่าสีที่ได้เกิดจากการรวมกันของค่าสีแดงและการเปลี่ยนแปลงที่เกิดจากการเต้นของหัวใจ ดังนั้นงานวิจัยนี้จึงนำ ICA มาวิเคราะห์เพื่อดึงสัญญาณการเปลี่ยนแปลงค่าสีที่เกิดจากการเต้นของหัวใจในวิดีโอเฟรม ด้วยหลักการ Joint Approximate Diagonalization of Eigen-matrices (JADE) (Jean-Francois Cardoso. Blind separation of real signals using JADE, version 1.9, 2015) สัญญาณค่าเฉลี่ยสีจากวิดีโอเฟรมที่ได้มีค่าเป็นโมเดลสี RGB แยกเป็น 3 แชนแนลคือ $y_1(t)$, $y_2(t)$ และ $y_3(t)$ เมื่อ t คือเวลาที่บันทึกสัญญาณ สัญญาณที่แยกได้คือ $x_1(t)$, $x_2(t)$ และ $x_3(t)$ และ A คือ ค่าสัมประสิทธิ์ที่ใช้แยกแชนแนลของสัญญาณที่ผสมกัน (mixture coefficients) ดังสมการที่ 2 รูปที่ 24 แสดงผลลัพธ์การคำนวณ ICA ของสัญญาณจากรูปที่ 23

$$y(t) = A \times x(t)$$

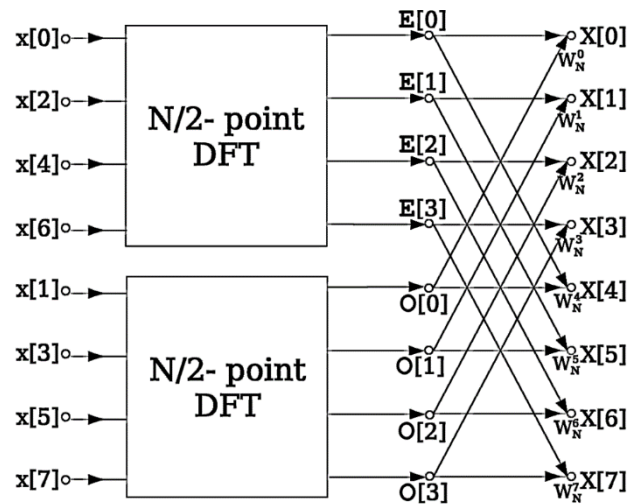
สมการที่ 2 การวิเคราะห์ ICA



รูปที่ 24 สัญญาณที่ผ่านการทำ ICA ของสัญญาณในรูปที่ 23

3.5. Fast-Fourier Transform (FFT)

การแปลงฟูเรียร์แบบไม่ต่อเนื่อง (Discrete Fourier Transform : DFT) เป็นการคำนวณทางคณิตศาสตร์เพื่อแปลงค่าสัญญาณจากโดเมนของเวลาเป็นโดเมนของความถี่ เพื่อนำมาใช้ในการวิเคราะห์สัญญาณ อาทิเช่น ความถี่ของสัญญาณชีพ สัญญาณรบกวน และความถี่ของสัญญาณรับเข้า การทำ FFT เป็นอัลกอริทึมที่ใช้เพิ่มความเร็วในการแปลง DFT โดยที่ผลลัพธ์ของการแปลงค่ายังคงเท่าเดิม การวิเคราะห์ FFT จะทำการแบ่งข้อมูลที่ละครั้งแล้วสลับตำแหน่งการคูณสัญญาณกับค่าถ่วงน้ำหนัก (weight) ทำอย่างนี้ไปเรื่อยๆจนไม่สามารถแบ่งข้อมูลได้แล้วจึงหยุด



รูปที่ 25 หลักการทำ FFT

ที่มา : Robert J. Schilling, Sandra L. Harris, Fundamentals of digital signal processing using MATLAB, 2nd ed, Toronto, Ont: Thomson, 2012

ตารางที่ 1 เปรียบเทียบความเร็วการคำนวณ DFT และ FFT

ที่มา : Ales Cerný. Mathematical techniques in finance: Tools for incomplete markets, 2009

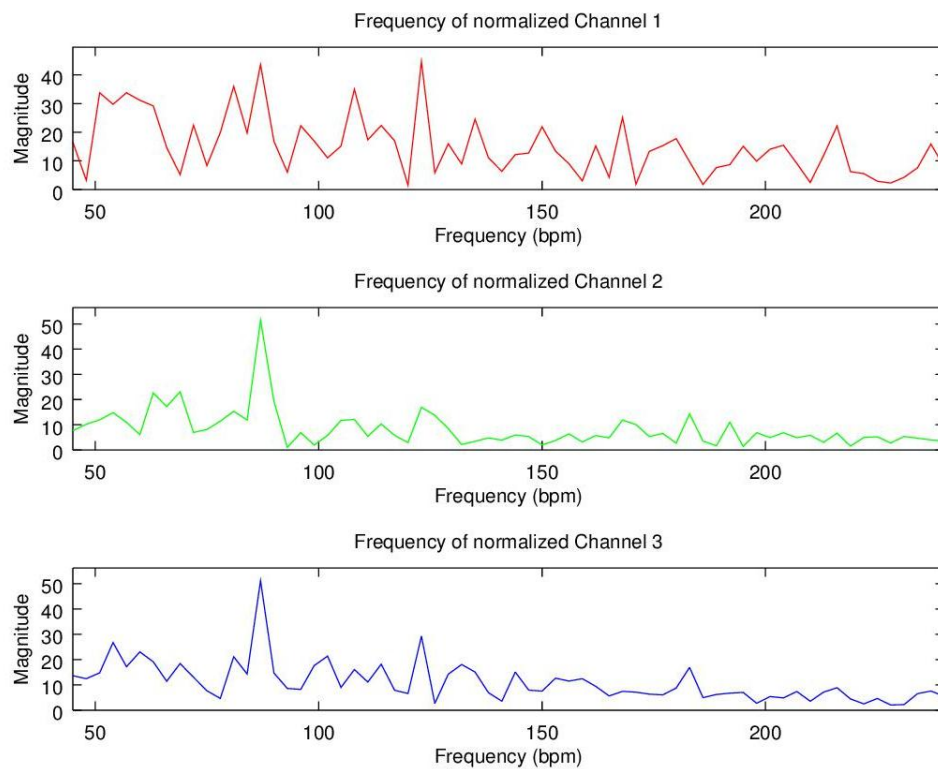
Number of point	Execution time (second)		Ratio
	DFT	FFT	
1008	0.6	0.003	200
2016	2.3	0.006	383.3
6048	20.8	0.022	945.5
30240	510	0.27	1888.9

การทำ FFT ถูกพัฒนาให้มีความรวดเร็วเพิ่มขึ้นจากการทำ DFT ซึ่งใช้เวลาลดลงจาก N^2 เหลือเพียง $N \log_2 N$ โดย N คือ จำนวน point ในการทำ FFT (N-point FFT) ดังแสดงในตารางที่ 1 พบว่าการทำ FFT เร็วกว่าการทำ DFT เช่นเมื่อ N มีค่า 1008 เวลาที่ใช้คำนวณ FFT เร็วกว่า DFT 200 เท่า และอัตราส่วนจะเพิ่มขึ้นเมื่อค่า N สูงขึ้น สมการที่ 3 แสดงการคำนวณค่า FFT ผลลัพธ์การคำนวณ FFT แสดงในรูปที่ 26 โดยเลือกแสดงในช่วงความถี่ 45 – 240 bpm

$$X(k) = \sum_{n=0}^{N-1} x(n) \times W_N^{nk} \quad ; k = 0, 1, \dots, N - 1$$

สมการที่ 3 แสดงการทำ FFT

โดย $x(n)$ คือ สัญญาณต้นฉบับ
 $X(k)$ คือ สัญญาณผลลัพธ์ที่ได้จากการคำนวณ FFT
 $W_N = e^{-j\frac{2\pi}{N}}$



รูปที่ 26 ผลลัพธ์การวิเคราะห์ความถี่ด้วย FFT ของสัญญาณในรูปที่ 23

3.6. Multi-component accumulation (MCA)

การหาความถี่สะสม (MCA) เป็นการนำค่า Magnitude ของความถี่ที่มีค่าเดียวกันจากแต่ละองค์ประกอบมารวมกัน เพื่อถึงความถี่สัญญาณชีพจรที่อาจมีค่า Magnitude ต่ำกว่าสัญญาณรบกวน อาจเนื่องมาจากการขยับของผู้ทำการทดลองหรือการสะท้อนของแสง การวัดชีพจรของ Poh ปี 2010 เลือกใช้องค์ประกอบที่ 2 จากการทำ ICA เพียงองค์ประกอบเดียวในการวัดชีพจร เพราะมีองค์ประกอบของสัญญาณชีพ และงานวิจัยของ Poh ปี 2011 เลือกใช้องค์ประกอบที่มีค่า Magnitude สูงสุดเป็นองค์ประกอบที่ใช้วัดค่าชีพจร แต่จากการทดลองพบว่าองค์ประกอบของสัญญาณชีพไม่ได้อยู่ในองค์ประกอบ 2 ของ ICA หรืออยู่ในองค์ประกอบใดองค์ประกอบหนึ่งเสมอไป เช่นในตารางที่ 2 แสดงความถี่ 5 อันดับแรกเรียงตามค่า Magnitude ของทั้ง 3 องค์ประกอบ พบว่าความถี่อันดับแรกขององค์ประกอบที่ 1 และ 3 ของ ICA มีค่าความถี่เป็น 87 และ 90 bpm ตามลำดับ มีใกล้เคียงกับชีพจรจริงที่มีค่าเป็น 87.65 bpm หรือในตารางที่ 3 ความถี่อันดับแรกขององค์ประกอบที่มีค่า Magnitude สูงสุด มีค่าความถี่เท่ากับ 69 bpm มีค่าต่างจากชีพจรจริงที่มีค่า 88.10 bpm

ตารางที่ 2 แสดงความถี่ 5 อันดับแรก เรียงตามค่า Magnitude ของผู้ทำการทดลองคนที่ 8 หน้าต่างที่ 34

ความถี่ 5 อันดับแรก เรียงตามค่า Magnitude ของผู้ทำการทดลองที่ 8 หน้าต่างที่ 34					
Component 1		Component 2		Component 3	
Magnitude	Pulse (BPM)	Magnitude	Pulse (BPM)	Magnitude	Pulse (BPM)
123.27	87	107.17	177	128.86	81
99.71	66	105.50	138	118.70	90
95.68	60	98.09	90	88.72	60
85.74	72	87.94	72	88.12	159
84.58	93	85.03	180	79.80	114

ตารางที่ 3 แสดงความถี่ 5 อันดับแรก เรียงตามค่า Magnitude ของผู้ทำการทดลองคนที่ 8 หน้าต่าง
ที่ 41

ความถี่ 5 อันดับแรก เรียงตามค่า Magnitude ของผู้ทำการทดลองที่ 8 หน้าต่างที่ 41					
Component 1		Component 2		Component 3	
Magnitude	Pulse (BPM)	Magnitude	Pulse (BPM)	Magnitude	Pulse (BPM)
131.63	69	119.92	84	101.37	90
104.10	78	103.55	72	97.51	177
100.87	84	97.11	87	90.98	81
100.48	90	84.55	63	90.92	102
87.41	63	78.04	117	88.60	75

การเลือกใช้เพียงแขนแนลใดแขนแนลหนึ่งหรือองค์ประกอบใดองค์ประกอบหนึ่งไม่เพียงพอต่อการระบุค่าชีพจร งานวิจัยนี้จึงนำความถี่ขององค์ประกอบทั้งหมดที่ได้จากทั้ง 3 แขนแนลมาวัดชีพจรโดยนำความถี่ 5 อันดับแรกเรียงตามค่า Magnitude มาหาความถี่สะสม เรียกว่า Multi-component accumulation (MCA) ผลลัพธ์การคำนวณ MCA ของตารางที่ 2 และตารางที่ 3 แสดงดังตารางที่ 4 และตารางที่ 5

ตารางที่ 4 ความถี่ 10 อันดับแรก เรียงตามค่า Magnitude ของวิธี MCA จากตารางที่ 2

ความถี่ 10 อันดับแรก เรียงตามค่า Magnitude ของวิธี MCA จากตารางที่ 2					
อันดับ	Magnitude	Pulse (BPM)	อันดับ	Magnitude	Pulse (BPM)
1	216.79	90	6	107.17	177
2	184.40	60	7	105.50	138
3	173.68	72	8	99.71	66
4	128.86	81	9	88.12	159
5	123.27	87	10	85.03	180

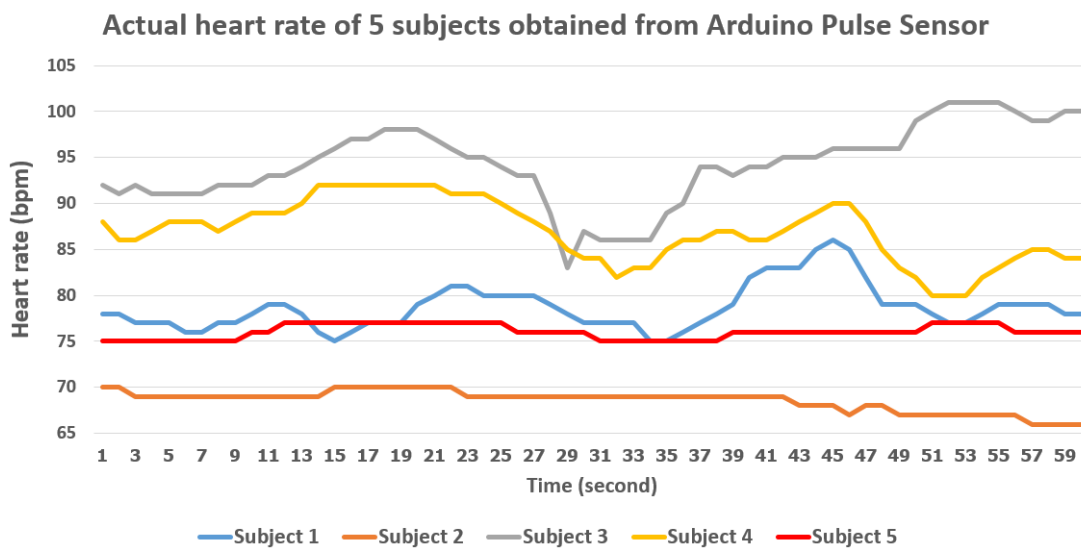
ตารางที่ 5 ความถี่ 10 อันดับแรก เรียงตามค่า Magnitude ของวิธี MCA จากตารางที่ 3

ความถี่ 10 อันดับแรก เรียงตามค่า Magnitude ของวิธี MCA จากตารางที่ 3					
อันดับ	Magnitude	Pulse (BPM)	อันดับ	Magnitude	Pulse (BPM)
1	220.80	84	6	103.55	72
2	201.84	90	7	97.51	177
3	171.96	63	8	97.11	87
4	131.63	69	9	90.98	81
5	104.10	78	10	90.92	102

ตารางที่ 4 ค่า Magnitude ของความถี่ 90 bpm เกิดจากการรวมค่า Magnitude จากองค์ประกอบที่ 2 และ 3 คือ $98.09 + 118.70 = 216.79$ ความถี่ 60 bpm จะมีค่า Magnitude เป็น $95.68 + 88.72 = 184.40$ พบว่าค่าความถี่ที่มีค่า Magnitude สูงสุดซึ่งมีค่าเท่ากับ 90 bpm มีค่าใกล้เคียงกับความถี่ชีพจรจริงคือ 87.65 bpm เช่นเดียวกับตารางที่ 5 ค่าความถี่ที่มีค่า Magnitude สูงสุดคือ 84 bpm เกิดจากการรวมกันของ $100.87 + 119.92 = 220.80$ ซึ่งใกล้เคียงกับค่าความถี่ชีพจรจริงคือ 88.1 bpm

3.7. Range-power spectrums (RPS)

อัตราการเต้นของชีพจรอาจมีการเปลี่ยนแปลงไปในช่วงเวลาอันสั้น (<5 วินาที) ดังแสดงในรูปที่ 27 แสดงค่าชีพจรที่วัดได้จาก Arduino Pulse Sensor ของผู้ทำการทดลอง 5 คน พบว่าในเวลา 5 วินาที ชีพจรที่วัดได้เปลี่ยนแปลงไป 10 bpm (ผู้ทำการทดลองที่ 4 ช่วงวินาทีที่ 46 – 51) ซึ่งอาจทำให้ผลลัพธ์การคำนวณค่าความถี่ด้วย FFT ในหน้าต่างเดียวกันอาจได้ค่าที่ไม่ตรงกับค่าความถี่ชีพจรจริงหรือมีค่าใกล้เคียงกับความถี่ชีพจรจริงได้มากกว่า 1 ค่า จากตารางที่ 4 ความถี่ชีพจรจริงมีค่าเป็น 87.65 bpm เมื่อวัดชีพจรพบว่าค่าความถี่ที่ใกล้เคียงกับชีพจรจริงคือ 87 90 และ 93 bpm เช่นเดียวกับตารางที่ 5 ความถี่ 84 87 และ 90 bpm มีค่าใกล้เคียงกับ 88.1 bpm ที่เป็นความถี่ชีพจรจริง



รูปที่ 27 แสดงชีพจรที่บันทึกได้จาก Arduino Pulse Sensor ของผู้ทำการทดลอง 5 คน
ในเวลา 1 นาที

งานวิจัยนี้เสนอการนำค่า Magnitude ของความถี่ที่มีค่าห่างกันไม่เกิน 2 bpm (± 2 bpm) มารวมกันเรียกว่า Range power spectrums (RPS) เช่น ค่า Magnitude ของความถี่ 90 bpm จะเกิดจากการรวมกันของความถี่ 88 89 90 91 และ 92 bpm ตารางที่ 6 แสดงผลลัพธ์การคำนวณ RPS จากตารางที่ 4 ความถี่ 88 bpm จะมีค่า Magnitude เป็น $123.27 (87 \text{ bpm}) + 216.76 (90 \text{ bpm}) = 340.06$ ซึ่งเท่ากับความถี่ 89 bpm ดังนั้นความถี่ที่มีค่า Magnitude สูงสุดมีค่าเป็นค่าเฉลี่ยของ 88 และ 89 bpm เท่ากับ 88.50 bpm และมีค่าใกล้เคียงกับชีพจรจริงที่ 87.65 bpm เช่นเดียวกับตารางที่ 7 เป็นผลลัพธ์ RPS จากตารางที่ 5 ความถี่ที่มีค่า Magnitude สูงสุดเท่ากับ 85.5 bpm เกิดจากค่าเฉลี่ยของ 86 และ 85 bpm ซึ่งใกล้เคียงกับความถี่ 88.10 bpm ที่เป็นความถี่ชีพจรจริง

ตารางที่ 6 ความถี่ 20 อันดับแรก เรียงตามค่า Magnitude ของวิธี RPS จากตารางที่ 4

ความถี่ 20 อันดับแรก เรียงตามค่า Magnitude ของวิธี RPS จากตารางที่ 4					
อันดับ	Magnitude	Pulse (BPM)	อันดับ	Magnitude	Pulse (BPM)
1	340.06	88	11	184.40	60
2	340.06	89	12	184.40	62
3	301.37	92	13	173.68	74
4	301.37	91	14	173.68	72
5	216.79	90	15	173.68	70
6	192.20	179	16	173.68	71
7	192.20	178	17	173.68	73
8	184.40	58	18	128.86	83
9	184.40	61	19	128.86	82
10	184.40	59	20	128.86	79

ตารางที่ 7 ความถี่ 20 อันดับแรก เรียงตามค่า Magnitude ของวิธี RPS จากตารางที่ 5

ความถี่ 20 อันดับแรก เรียงตามค่า Magnitude ของวิธี RPS จากตารางที่ 5					
อันดับ	Magnitude	Pulse (BPM)	อันดับ	Magnitude	Pulse (BPM)
1	317.91	86	11	201.84	91
2	317.91	85	12	201.84	90
3	311.78	82	13	195.08	79
4	311.78	83	14	195.08	80
5	298.96	89	15	192.70	76
6	298.96	88	16	192.70	77
7	235.17	70	17	192.15	74
8	235.17	71	18	192.15	73
9	220.80	84	19	171.96	61
10	201.84	92	20	171.96	64

3.8. Error suppression (ES)

การวัดค่าชีพจรโดยการคำนวณความถี่การเปลี่ยนแปลงของสีด้วย FFT อาจคำนวณค่าความถี่ได้ต่างจากความถี่ของชีพจร เนื่องจากผลกระทบของการสะท้อนแสงหรือผู้ทำการทดลองขยับมากเกินไป ซึ่งทำให้ผลลัพธ์ที่ได้จากแต่ละหน้าต่ามีค่าไม่สอดคล้องกัน Poh (2010: 10762-10774) จึงนำค่าที่ได้จากแต่ละหน้าต่ามาเปรียบเทียบกันโดยพิจารณาจากค่าเทรสโฮลด์ (Threshold) เรียกว่า Error Suppression (ES)

การเปรียบเทียบดังกล่าวจะพิจารณาผลต่างระหว่างค่าความถี่ชีพจรในหน้าต่าปัจจุบันกับหน้าต่าก่อนหน้า ถ้าผลต่างมากกว่าค่า Threshold ระบบจะตรวจสอบความถี่ที่มีค่า Magnitude ต่ำกว่าจนพบค่าความถี่ที่มีผลต่างต่ำกว่าค่า Threshold ถ้าไม่มีค่าความถี่ที่ตรงตามเงื่อนไข ระบบจะคงค่าความถี่เดิมไว้ ซึ่งงานวิจัยของ Poh (2010: 10762-10774) และ Datcu (2013: 120-127) กำหนดค่า Threshold เป็น 12 bpm

การเลือกใช้ค่า Threshold เป็น 12 bpm เป็นค่าเปรียบเทียบอาจมากเกินไปและอาจได้ค่าชีพจรที่ไม่สอดคล้องกับชีพจรจริงหรือชีพจรจากหน้าต่าก่อนหน้า จากการศึกษาพบว่าเมื่อวิเคราะห์ความถี่ด้วย MCA และ RPS ส่งผลให้ระบุชีพจรได้แม่นยำขึ้น ดังนั้นงานวิจัยนี้จึงพัฒนา ES 3 ประการคือ

1. จากการศึกษาค่าชีพจรที่ได้จาก Arduino พบว่าค่าชีพจรในวินาทีใดๆต่างจากในวินาทีก่อนหน้าสูงสุด 11 bpm และมีโอกาสเกิดขึ้น 0.08% ซึ่งน้อยมาก ต่างจากกรณีที่ชีพจรไม่เปลี่ยนแปลงเลย ซึ่งมีโอกาสเกิดขึ้น 55.76% เพราะผู้ทำการทดลองไม่ได้อยู่ในสภาวะกดดัน ดังแสดงใน จะเห็นว่าค่าความต่างของชีพจรในช่วง 1 วินาทีที่ต่ำกว่า 3 bpm จะมีค่าเข้าใกล้ 0% ดังนั้นงานวิจัยนี้จึงกำหนดค่า Threshold ของการทำ Error Suppression เป็น 3 bpm

ตารางที่ 8 ความเป็นไปได้ในการของการเปลี่ยนแปลงค่าชีพจรใน 1 วินาที

ความต่างของชีพจร ในช่วง 1 วินาที (bpm)	โอกาสการเกิด (%)
0	55.76
1	30.59
2	6.69
3	5.84
4	0.51
5	0.17
6	0.17
7	0
8	0.085
9	0.085
10	0
11	0.085

2. ในกรณีที่ค่าชีพจรจากหน้าต่างปัจจุบันต่างจากหน้าต่างก่อนหน้ามากกว่า 3 bpm ระบบจะตรวจสอบความถี่ในลำดับถัดไปที่มีค่า Magnitude จะมากกว่า 70% ของค่า Magnitude สูงสุดเท่านั้น ตารางที่ 9 แสดงการเลือกความถี่ที่มีค่า Magnitude สูงกว่า 70% ของค่า Magnitude สูงสุด จากตารางค่า Magnitude สูงสุดเท่ากับ 2763.01 ดังนั้น 70% ของ Magnitude สูงสุดเท่ากับ 1934.1 เพราะฉะนั้นค่าความถี่ที่มีค่า Magnitude เกิน 70% คือ 65, 91 และ 88 bpm

ตารางที่ 9 แสดงการเลือกค่า Magnitude ที่สูงกว่า 70% ของ Magnitude สูงสุด

Magnitude	2763.01	2598.56	2271.26	1833.74	...
Pulse (bpm)	65	91	88	94	...

3. การกำหนดค่าชีพจรในหน้าต่างแรกเป็นสิ่งสำคัญ เนื่องจากความถี่การทำ ES ในหน้าต่างที่ 1 จะมีค่าเท่ากับค่าความถี่ที่วัดได้เนื่องจากไม่มีผลลัพธ์ในหน้าต่างก่อนหน้าให้เปรียบเทียบ ถ้าผลลัพธ์ชีพจรในหน้าต่างแรกผิดพลาด จะทำให้การระบุค่าชีพจรในหน้าต่างอื่นผิดพลาดตามไปด้วย ดังนั้นงานวิจัยนี้เสนอการพิจารณาค่าชีพจรในหน้าต่างที่ 1 จากการเปรียบเทียบค่าจากหน้าต่าง 3 หน้าต่างแรก โดยพิจารณาจากผลต่างของทั้ง 3 หน้าต่างคือ หน้าต่างที่ 1 กับ 2, หน้าต่างที่ 1 กับ 3 และหน้าต่างที่ 2 กับ 3 และเลือกค่าความถี่ชีพจรของหน้าต่างหลัง (หน้าต่างที่ 2 ที่นำมาหาผลต่าง) ที่มีค่าผลต่างน้อยที่สุด ตารางที่ 10 แสดงหลักการคำนวณค่าชีพจรในหน้าต่างแรกโดยพิจารณาจาก 3 หน้าต่างแรก ค่าชีพจรในหน้าต่างที่ 1 กับ 2 ต่างกัน 25.5 bpm (88.5 – 63) หน้าต่างที่ 1 กับ 3 ต่างกัน 27 bpm (90 – 63) และหน้าต่างที่ 2 กับ 3 ต่างกัน 1.5 bpm (90 – 88.5) ผลต่างที่น้อยที่สุดคือ หน้าต่างที่ 2 กับ 3 ดังนั้นค่าความถี่ในหน้าต่างที่ 1 จะมีค่าเท่ากับค่าความถี่ในหน้าต่างที่ 3 (หน้าต่างหลัง) คือ 90 bpm ตารางที่ 11 เป็นผลลัพธ์การพิจารณาค่าความถี่หน้าต่างที่ 1 จาก 3 หน้าต่างแรก

ตารางที่ 10 ความถี่แต่ละหน้าต่าง ก่อนพิจารณาความถี่หน้าต่างที่ 1 จาก 3 หน้าต่างแรก

หน้าต่าง	1	2	3	4	5	6	7	8	9	10	11
ความถี่	67	88	90	88	88	65	94	94	94	91	90

ตารางที่ 11 ความถี่แต่ละหน้าต่าง หลังพิจารณาความถี่หน้าต่างที่ 1 จาก 3 หน้าต่างแรก

หน้าต่าง	1	2	3	4	5	6	7	8	9
ความถี่	90	88	88	65	94	94	94	91	90

ตารางที่ 12 แสดงการคำนวณ ES

หน้าต่าง	1	2	3	4	5	6
ก่อนทำ ES	90	88	88	65	94	94
หลังทำ ES	90	88	88	91	94	94

ตารางที่ 12 แสดงการทำงานของ ES ผลลัพธ์ค่าชีพจรในหน้าต่างที่ 1 จะมีค่าเท่ากับความเร็วที่วัดได้ซึ่งเท่ากับ 90 bpm เพราะไม่มีหน้าต่างก่อนหน้าให้เปรียบเทียบ ค่าความเร็วที่คำนวณได้ในหน้าต่างที่ 6 มีค่าเท่ากับตารางที่ 9 พบว่า ค่าชีพจรสูงสุดต่างจากค่าชีพจรในหน้าต่างที่ 5 อยู่ 23 bpm (88 - 65) ซึ่งมากกว่าค่า threshold ที่ตั้งไว้คือ 3 bpm ดังนั้นระบบจะค้นหาค่าความเร็วในลำดับถัดไปคือ 91 bpm ซึ่งต่างจากค่าความเร็ว 88 bpm อยู่ 3 bpm ดังนั้นค่าความเร็วในหน้าต่างที่ 6 จึงมีค่าเป็น 91.5 bpm

ตารางที่ 13 และตารางที่ 14 แสดงผลลัพธ์การทำ ES ของตารางที่ 10 และตารางที่ 11 เทียบกับชีพจรอ้างอิงตามลำดับ พบว่าผลลัพธ์การทำ ES ในหน้าต่างที่ 1 และ 2 ของตารางที่ 13 วัดค่าชีพจรผิดพลาด เนื่องจากวัดค่าความเร็วในหน้าต่างแรกผิดพลาด จึงทำให้ผลลัพธ์ในหน้าต่างที่ 2 ผิดพลาดด้วย ต่างจากตารางที่ 14 ที่ค่าความเร็วในหน้าต่างที่ 1 พิจารณามาจาก 3 หน้าต่างแรก ทำให้การระบุค่าความเร็วในหน้าต่างที่ 1 ถูกต้อง

ตารางที่ 13 ผลลัพธ์การคำนวณชีพจรด้วย ES จากตารางที่ 10

หน้าต่าง	1	2	3	4	5	6	7	8	9	10	11
ก่อนทำ ES	67	88	90	88	88	65	94	94	94	91	90
หลังทำ ES	67	67	90	88	88	88	88	88	88	91	90
ค่าชีพจรอ้างอิง	88	88	88	88	88	89	89	90	92	93	94

ตารางที่ 14 ผลลัพธ์การคำนวณชีพจรด้วย ES จากตารางที่ 11

หน้าต่าง	1	2	3	4	5	6	7	8	9
ก่อนทำ ES	90	88	88	65	94	94	94	91	90
หลังทำ ES	90	88	88	91	94	94	94	91	90
ค่าชีพจรอ้างอิง	88	88	88	89	89	90	92	93	94

บทที่ 4

วิธีการดำเนินงานวิจัย

บทนี้จะกล่าวถึงวิธีการดำเนินงานวิจัย และขั้นตอนการพัฒนากระบวนการวัดชีพจรแบบไม่สัมผัสร่างกายตามที่คุณวิจัยได้ตั้งสมมุติฐานมาก่อนหน้านี้

4.1. โครงสร้างระบบ

4.1.1. Hardware

a. อุปกรณ์ส่วนการประมวลผล

หน่วยประมวลผลกลาง (CPU) : Intel core i5-3210M (2.50 GHz)

หน่วยความจำ (Memory) : 8 GB DDR3

b. อุปกรณ์ส่วนการรับภาพวิดีโอ

iPad 3

Webcam Logitech C920

c. อุปกรณ์ส่วนการรับสัญญาณชีพจร

Arduino

Pulse sensor

4.1.2. Software

a. ระบบปฏิบัติการ Window 10

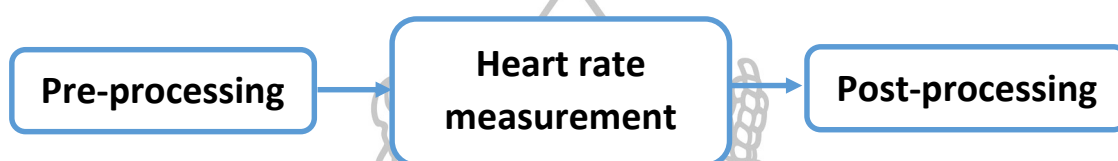
b. โปรแกรม Java

c. โปรแกรม Octave

4.2. ขั้นตอนการทำงานของระบบ

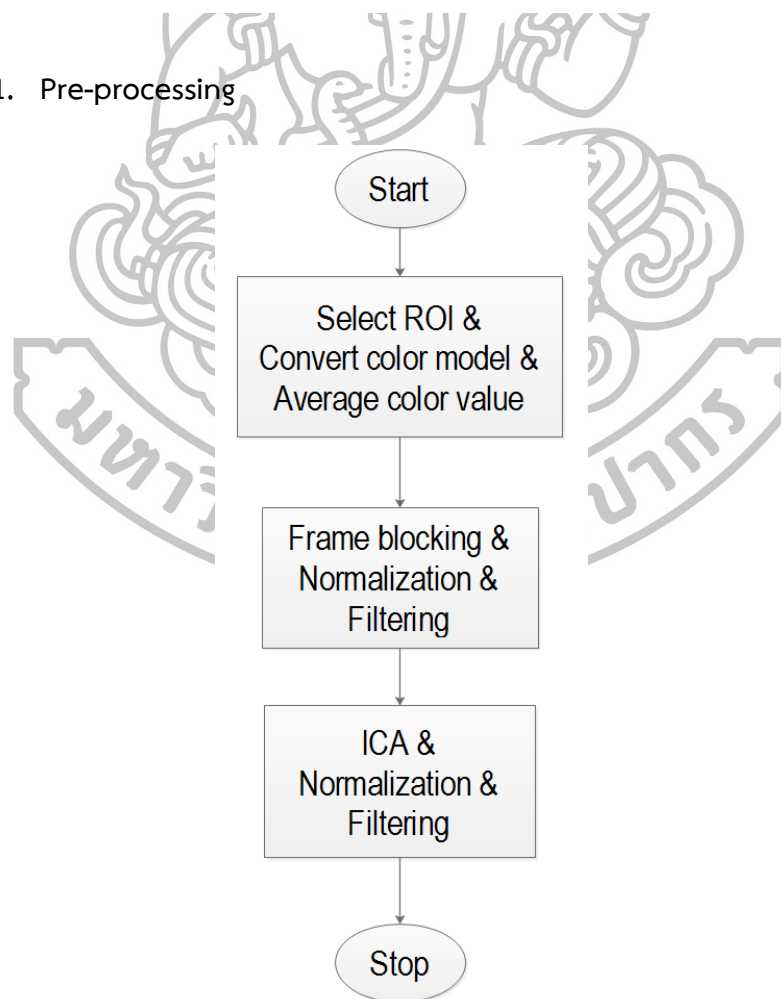
กำหนดขั้นตอนการทำงานของระบบการวัดชีพจรแบบไม่สัมผัสร่างกาย โดยผู้วิจัยได้ออกแบบขั้นตอนการทำงานไว้ 3 ขั้นตอนดังรูปที่ 28 โดยมีรายละเอียดดังนี้

- a. ขั้นตอนที่ 1 การประมวลผลเบื้องต้น (Pre-processing)
- b. ขั้นตอนที่ 2 การวัดชีพจร (Heart rate measurement)
- c. ขั้นตอนที่ 3 Post-processing



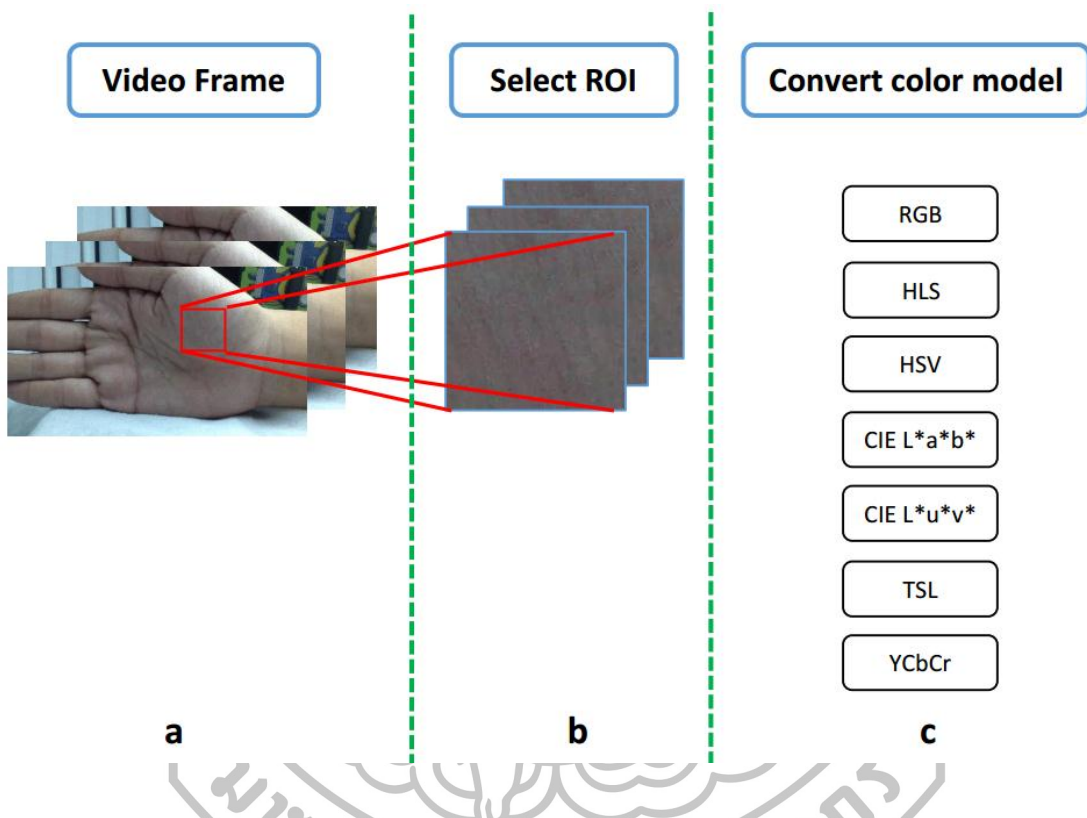
รูปที่ 28 Flowchart การวัดชีพจรแบบไม่สัมผัสร่างกาย

4.2.1. Pre-processing

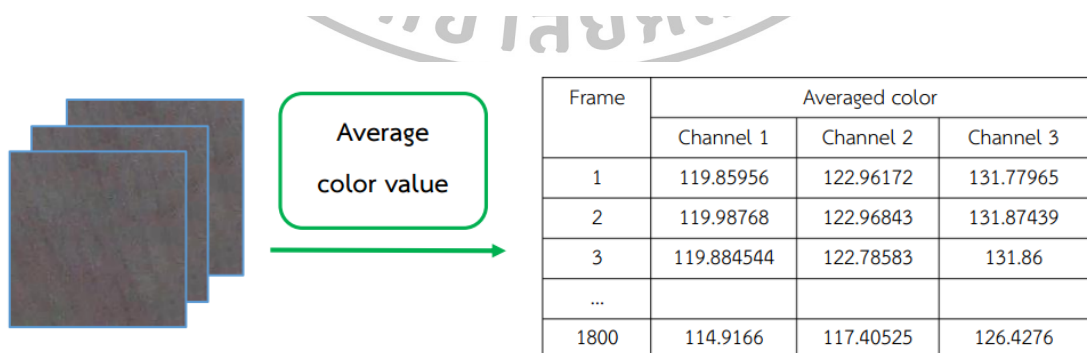


รูปที่ 29 Flowchart การทำ Pre-processing

การประมวลผลเบื้องต้น (Pre-processing) เป็นการเตรียมและสร้างฟีเจอร์ที่ใช้ในการวัดค่าชีพจร มีขั้นตอนการทำงานดังรูปที่ 29 เริ่มจากการเลือกส่วนที่สนใจ (Select Region of Interest) จากวิดีโอเฟรม โดยตัด (Crop) ภาพขนาด 150 x 151 pixel จากส่วนบริเวณฝ่ามือ บริเวณเดียวกันในผู้ทำการทดลองทุกคนและนำมาแปลงค่าโมเดลสี ดังหัวข้อที่ 3.1 รูปที่ 30 แสดงขั้นตอนดังกล่าว วิดีโอเฟรมที่ได้จะนำมาคำนวณค่าเฉลี่ยสีในแต่ละแกนแนล ดังรูปที่ 31



รูปที่ 30 a) วิดีโอเฟรมที่ได้จากการบันทึก b) เลือกตัดส่วนอวัยวะที่สนใจเพื่อวัดชีพจร c) แปลงโมเดลสี

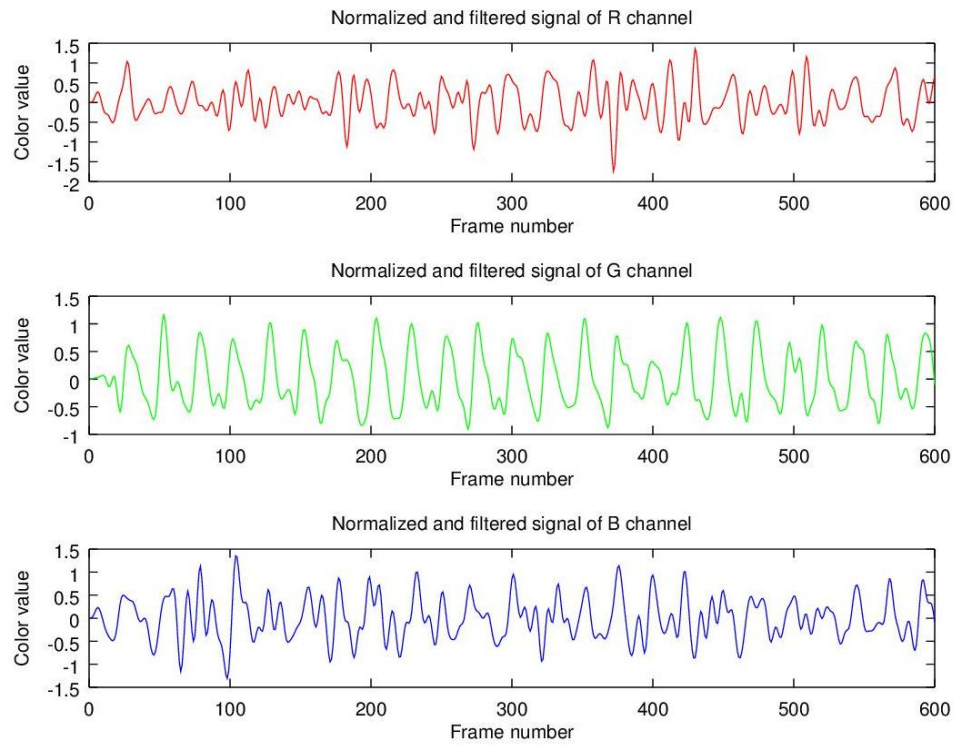


รูปที่ 31 คำนวณค่าเฉลี่ยสีทุกแกนแนลในแต่ละเฟรม

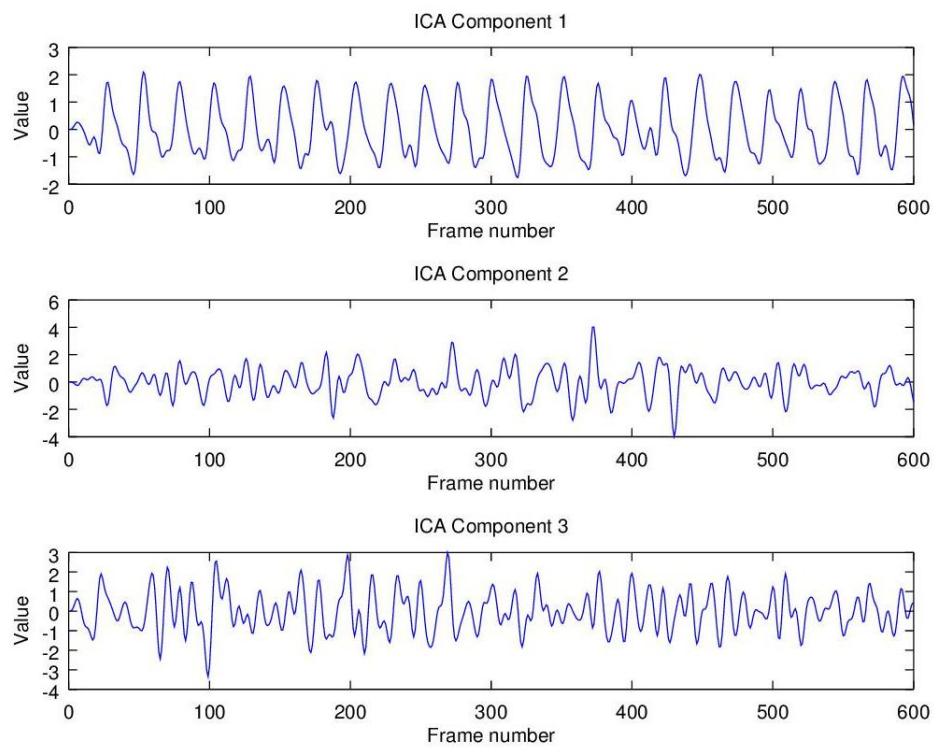
ตารางที่ 15 แสดงจำนวนหน้าต่างในแต่ละขนาดหน้าต่าง

ขนาดหน้าต่าง		จำนวนหน้าต่าง
วินาทีต่อหน้าต่าง	เฟรมต่อหน้าต่าง (30 FPS)	
5	150	56
10	300	51
15	450	46
20	600	41
30	900	31
40	1200	21
50	1500	11

งานวิจัยนี้มีเป้าหมายที่จะศึกษาผลกระทบจากการเปลี่ยนขนาดหน้าต่างที่ใช้ในการวัดค่าซีพจรจึงศึกษาการแบ่งสัญญาณออกเป็นหน้าต่างละ 5, 10, 15, 20, 30, 40 และ 50 วินาที ซ้อนทับสัญญาณ (Overlap) 1 วินาที ตารางที่ 15 แสดงจำนวนหน้าต่างที่แบ่งได้ในแต่ละขนาดหน้าต่าง นำสัญญาณที่ได้ไปทำการ Normalization และกรองความถี่ ตามหัวข้อที่ 3.2 และ 3.3 ตามลำดับ รูปที่ 32 แสดงสัญญาณการเปลี่ยนแปลงค่าสีของโมเดลสี RGB ที่คำนวณได้จากขั้นตอนดังกล่าว และรูปที่ 33 แสดงสัญญาณที่ผ่านการคำนวณ ICA ที่ทำการ Normalization และกรองความถี่อีกครั้ง

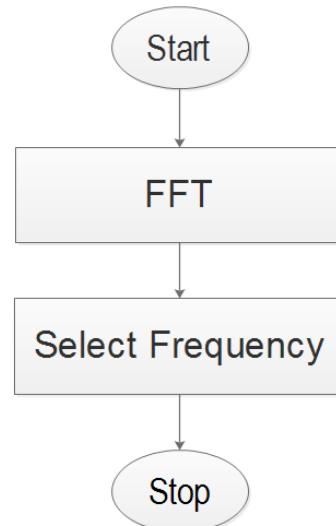


รูปที่ 32 สัญญาณค่าเฉลี่ยสีต้นฉบับ



รูปที่ 33 ผลลัพธ์การวิเคราะห์ ICA ของสัญญาณในรูปที่ 32

4.2.2. Heart rate measurement



รูปที่ 34 Flowchart การวัดค่าชีพจร

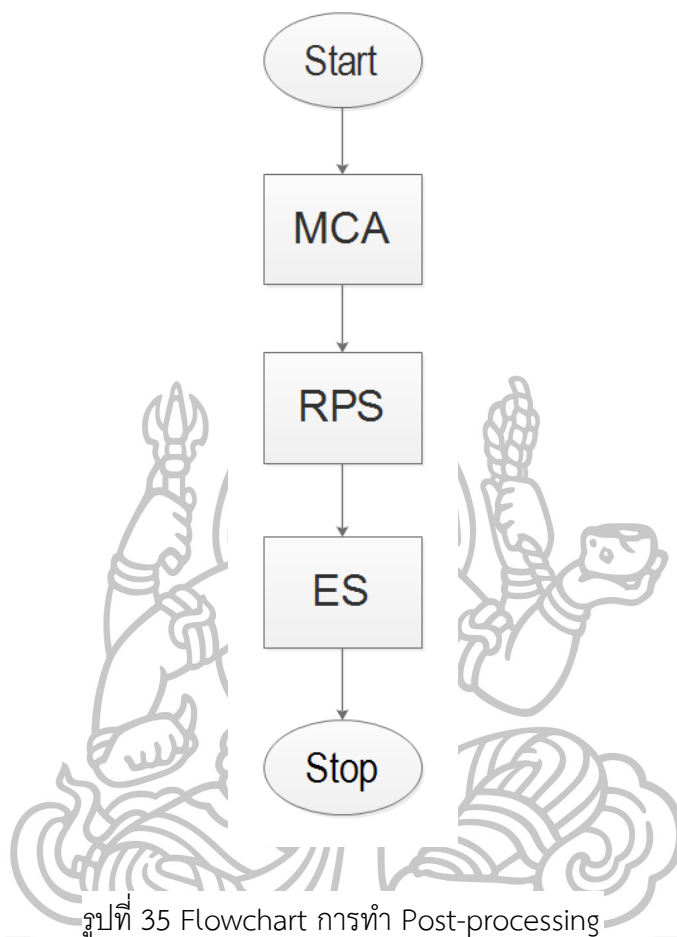
สัญญาณค่าสี RGB ที่ผ่านการคำนวณ ICA จะนำมาวัดค่าความถี่ด้วย FFT โดยเลือกค่าความถี่ที่อยู่ในช่วง 45 – 240 bpm ตารางที่ 16 แสดงตัวอย่างผลลัพธ์การคำนวณความถี่ของผู้ทำการทดลองทั้ง 20 คน เปรียบเทียบกับชีพจรที่วัดได้จาก Arduino pulse sensor จะเห็นว่าค่าความถี่ที่มีค่า Magnitude สูงสุด ที่ใกล้เคียงกับชีพจรอ้างอิง อาจอยู่ในองค์ประกอบใดก็ได้หรือสามารถอยู่ได้มากกว่าหนึ่งองค์ประกอบ



ตารางที่ 16 ความถี่ที่มีค่า Magnitude สูงสุด คำนวณจาก FFT ของสัญญาณที่ผ่านการทำ ICA

ผู้ทำการทดลอง	ความถี่ที่มีค่า Magnitude สูงสุด คำนวณจาก FFT ของสัญญาณที่ผ่านการทำ ICA (bpm)			ชีพจรที่ได้จาก Arduino pulse sensor (bpm)
	องค์ประกอบที่ 1	องค์ประกอบที่ 2	องค์ประกอบที่ 3	
คนที่ 1	79.2	84.0	67.2	78.7
คนที่ 2	75.6	108.0	75.6	69.0
คนที่ 3	87.6	87.6	66.0	91.8
คนที่ 4	86.4	85.2	85.2	87.8
คนที่ 5	121.2	74.4	61.2	75.9
คนที่ 6	81.6	74.4	80.4	79.9
คนที่ 7	79.2	78.0	79.2	75.5
คนที่ 8	90.0	90.0	88.8	87.4
คนที่ 9	67.2	134.4	64.8	65.3
คนที่ 10	63.6	63.6	63.6	61.6
คนที่ 11	61.2	70.8	72.0	72.6
คนที่ 12	61.2	61.2	85.2	82.7
คนที่ 13	61.2	79.2	54.0	82.5
คนที่ 14	67.2	74.4	74.4	75.3
คนที่ 15	57.6	75.6	81.6	80.4
คนที่ 16	79.2	78.0	58.8	82.8
คนที่ 17	50.4	112.8	68.4	119.8
คนที่ 18	86.4	73.2	76.8	84.7
คนที่ 19	50.4	78.0	75.6	80.0
คนที่ 20	62.4	68.4	109.2	64.0

4.2.3. Post-processing



รูปที่ 35 Flowchart การทำ Post-processing

การเลือกค่าความถี่จะเลือกจากทุกองค์ประกอบที่ได้จากการทำ ICA โดยนำค่าความถี่ที่ได้มาหาความถี่สะสมด้วย MCA และ RPS ตารางที่ 17 แสดงตัวอย่างความถี่ที่คำนวณจาก MCA และ RPS ของผู้ทำการทดลอง 20 คน เช่นเดียวกับตารางที่ 16

ตารางที่ 17 ความถี่ที่มีค่า Magnitude สูงสุด คำนวณจาก MCA และ RPS

ผู้ทำการทดลอง	ความถี่ที่มีค่า Magnitude สูงสุด ของวิธีการที่ นำเสนอ		ชีพจรที่ได้จาก Arduino pulse sensor (bpm)
	MCA	RPS	
คนที่ 1	79.2	80.0	78.7
คนที่ 2	75.6	76.0	69.0
คนที่ 3	87.6	88.0	91.8
คนที่ 4	85.2	86.0	87.8
คนที่ 5	61.2	76.0	75.9
คนที่ 6	80.4	81.0	79.9
คนที่ 7	79.2	78.0	75.5
คนที่ 8	90.0	91.0	87.4
คนที่ 9	67.2	67.0	65.3
คนที่ 10	63.6	66.0	61.6
คนที่ 11	70.8	72.0	72.6
คนที่ 12	61.2	61.0	82.7
คนที่ 13	61.2	63.0	82.5
คนที่ 14	74.4	67.0	75.3
คนที่ 15	81.6	82.0	80.4
คนที่ 16	79.2	78.0	82.8
คนที่ 17	112.8	51.0	119.8
คนที่ 18	86.4	87.0	84.7
คนที่ 19	50.4	78.0	80.0
คนที่ 20	68.4	110.0	64.0

ผลลัพธ์ที่ได้จากทุกหน้าต่างจะนำมาเปรียบเทียบกับหน้าต่างก่อนหน้าด้วย ES เพื่อจำกัดความผิดพลาดในการระบุค่าชีพจร ตารางที่ 18 แสดงผลลัพธ์การวัดค่าชีพจรจากวิธีที่นำเสนอเปรียบเทียบกับชีพจรที่ได้จาก Arduino pulse sensor ของผู้ทำการทดลอง 20 คน โดยใช้ขนาด

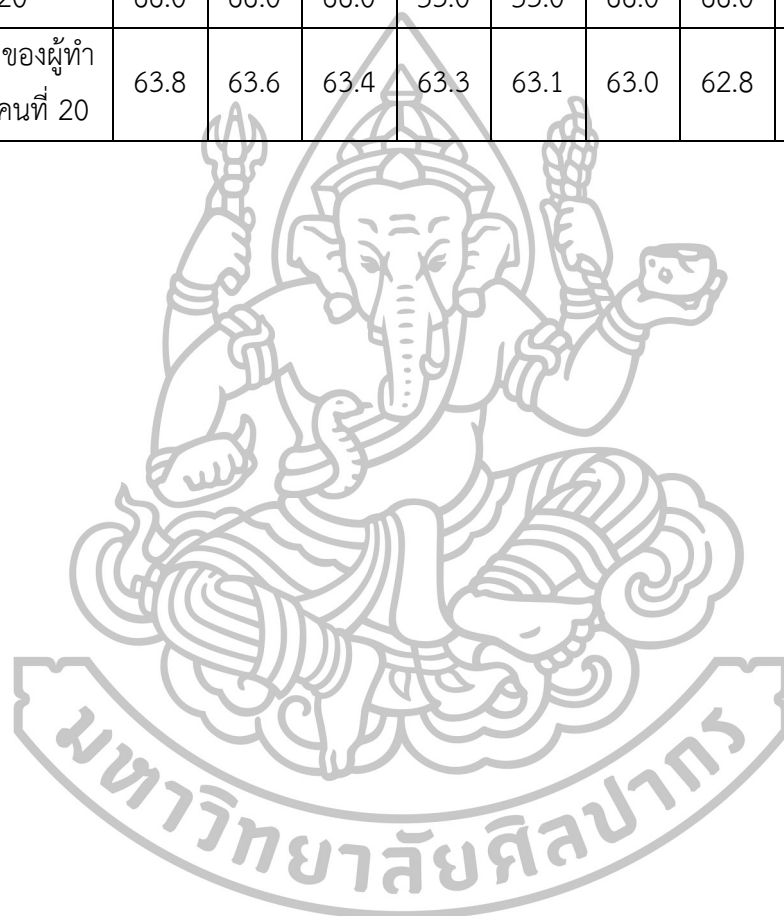
หน้าต่าง 1500 frame/window จะมีทั้งหมด 9 หน้าต่าง (หน้าต่างที่ 1 เกิดจากการวิเคราะห์จาก 3 หน้าต่างแรก)

ตารางที่ 18 ผลลัพธ์การคำนวณชีพจรในแต่ละหน้าต่าง โดยใช้วิธี MCA, RPS และ ES เปรียบเทียบกับชีพจรที่ได้จาก Arduino pulse sensor

ผู้ทำการทดลอง	ชีพจรจากแต่ละหน้าต่าง								
	1	2	3	4	5	6	7	8	9
คนที่ 1	79.0	81.0	83.5	84.0	82.0	80.0	79.0	82.0	80.0
ชีพจรอ้างอิงของผู้ทำการทดลองคนที่ 1	78.7	78.7	78.7	78.8	78.8	78.9	78.9	79.0	79.0
คนที่ 2	75.0	75.0	75.0	75.0	75.0	75.0	75.0	75.0	76.0
ชีพจรอ้างอิงของผู้ทำการทดลองคนที่ 2	68.9	68.8	68.8	68.7	68.7	68.6	68.6	68.5	68.5
คนที่ 3	88.0	88.0	89.5	90.0	88.0	90.0	88.0	88.0	88.0
ชีพจรอ้างอิงของผู้ทำการทดลองคนที่ 3	92.2	92.4	92.6	92.8	93.0	93.1	93.3	93.4	93.6
คนที่ 4	87.0	87.0	87.0	87.0	87.0	87.0	87.0	87.0	86.0
ชีพจรอ้างอิงของผู้ทำการทดลองคนที่ 4	87.6	87.4	87.3	87.2	87.2	87.1	87.1	87.0	86.9
คนที่ 5	76.0	76.0	76.0	76.0	76.0	76.0	76.0	76.0	76.0
ชีพจรอ้างอิงของผู้ทำการทดลองคนที่ 5	76.0	76.1	76.1	76.1	76.2	76.2	76.2	76.2	76.2
คนที่ 6	80.0	80.0	80.0	81.5	81.0	81.0	81.0	81.0	81.0
ชีพจรอ้างอิงของผู้ทำการทดลองคนที่ 6	79.9	79.8	79.8	79.8	79.8	79.8	79.8	79.8	79.8
คนที่ 7	78.0	78.0	78.0	78.0	78.0	78.0	78.0	78.0	78.0
ชีพจรอ้างอิงของผู้ทำการทดลองคนที่ 7	75.7	75.8	75.9	75.9	76.0	76.1	76.1	76.2	76.2

คนที่ 8	88.0	88.0	88.0	88.0	88.0	89.5	90.0	89.5	90.5
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 8	87.5	87.5	87.5	87.5	87.5	87.6	87.6	87.7	87.8
คนที่ 9	66.0	66.5	66.5	66.5	66.5	66.5	66.5	66.5	66.5
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 9	65.3	65.3	65.3	65.3	65.3	65.3	65.3	65.3	65.3
คนที่ 10	63.0	64.0	66.0	66.0	66.0	63.0	66.0	66.0	66.0
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 10	61.6	61.6	61.6	61.6	61.7	61.7	61.7	61.8	61.8
คนที่ 11	60.0	60.0	71.0	71.0	60.0	60.5	71.5	71.5	71.0
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 11	72.7	72.8	72.9	73.0	73.1	73.2	73.3	73.3	73.4
คนที่ 12	60.5	60.5	60.5	60.5	60.5	60.5	60.5	60.5	60.5
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 12	82.9	83.0	83.0	83.0	83.1	83.1	83.1	83.4	83.6
คนที่ 13	78.0	78.5	78.0	81.0	81.0	81.0	78.5	81.0	81.0
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 13	82.7	82.8	82.9	83.1	83.2	83.4	83.5	83.6	83.7
คนที่ 14	74.0	74.0	76.0	76.0	74.0	72.5	72.5	72.5	74.0
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 14	75.7	75.8	75.9	76.0	76.0	75.9	75.9	75.8	75.6
คนที่ 15	80.5	81.5	83.0	83.0	82.0	82.0	82.0	82.0	82.0
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 15	80.4	80.4	80.3	80.3	80.3	80.2	80.2	80.1	80.1
คนที่ 16	78.0	78.0	78.0	78.5	78.0	78.0	78.0	78.0	77.5
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 16	82.8	82.8	82.8	82.7	82.7	82.7	82.7	82.7	82.7
คนที่ 17	54.0	54.0	54.0	54.0	54.0	56.0	56.0	56.0	56.0
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 17	119.9	119.9	120.0	120.0	120.0	120.0	119.8	119.5	119.3

คนที่ 18	85.5	86.5	86.0	85.5	86.5	85.0	86.5	86.5	86.5
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 18	84.8	84.8	84.8	84.9	84.9	84.9	84.9	84.8	84.8
คนที่ 19	52.5	54.5	54.5	54.0	54.0	54.5	54.5	54.0	54.0
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 19	80.0	79.9	79.9	79.9	79.8	79.8	79.7	79.6	79.6
คนที่ 20	66.0	66.0	66.0	53.0	53.0	66.0	66.0	66.0	68.0
ซีพจรอ้างอิงของผู้ทำ การทดลองคนที่ 20	63.8	63.6	63.4	63.3	63.1	63.0	62.8	62.7	62.6



บทที่ 5

การทดสอบระบบและผลการดำเนินการวิจัย

5.1. การทดสอบระบบ

ทดสอบระบบวัดชีพจรกับผู้ทำการทดลอง 20 คน เป็นชาย 5 คนและหญิง 15 คน มีอายุ ตั้งแต่ 10 – 55 ปี โดยทดลองในห้องปิดและใช้กล่องควบคุมแสงด้วยหลอด led ในห้องที่มี แหล่งกำเนิดแสงแหล่งเดียวเพื่อตัดปัญหาเกี่ยวกับแหล่งกำเนิดแสง เพราะต้องการวิเคราะห์ผลกระทบ ของโมเดลสี ดังรูปที่ 36 a) คือ กล่องที่ใช้บันทึก b) คือ ขณะบันทึกวิดีโอ และ c) คือ วิดีโอเฟรมที่ได้ จากการบันทึก วิดีโอถูกบันทึกด้วยกล้อง iPad3 และ Logitech C920 ข้อมูลชีพจรและความดันของ ผู้ทำการทดลองถูกวัดด้วยเครื่องวัดความดันเลือด (Digital blood pressure) และ Arduino pulse sensor รูปที่ 37 แสดงอุปกรณ์การวัดสัญญาณชีพด้วยเครื่องวัดความดันเลือดและ Arduino pulse sensor ตารางที่ 19 แสดงข้อมูลของผู้ทำการทดลองทั้ง 20 คน โดยผลลัพธ์ที่ได้จาก Arduino pulse sensor จะนำชีพจรทั้ง 60 วินาทีมาหาค่าเฉลี่ย ค่าสูงสุด และค่าต่ำสุด รูปที่ 38 แสดงสัญญาณชีพจร ของผู้ทำการทดลองทั้ง 20 คนที่วัดได้จาก Arduino pulse sensor ในเวลา 1 นาที

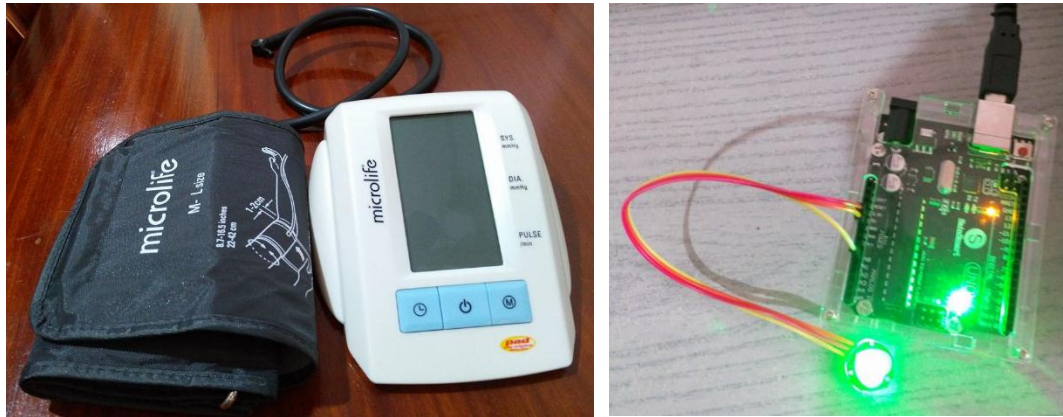


a

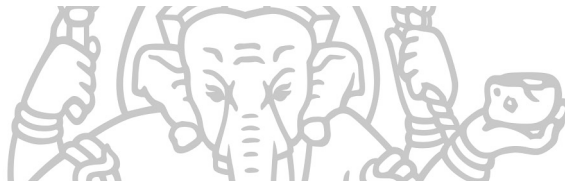
b

c

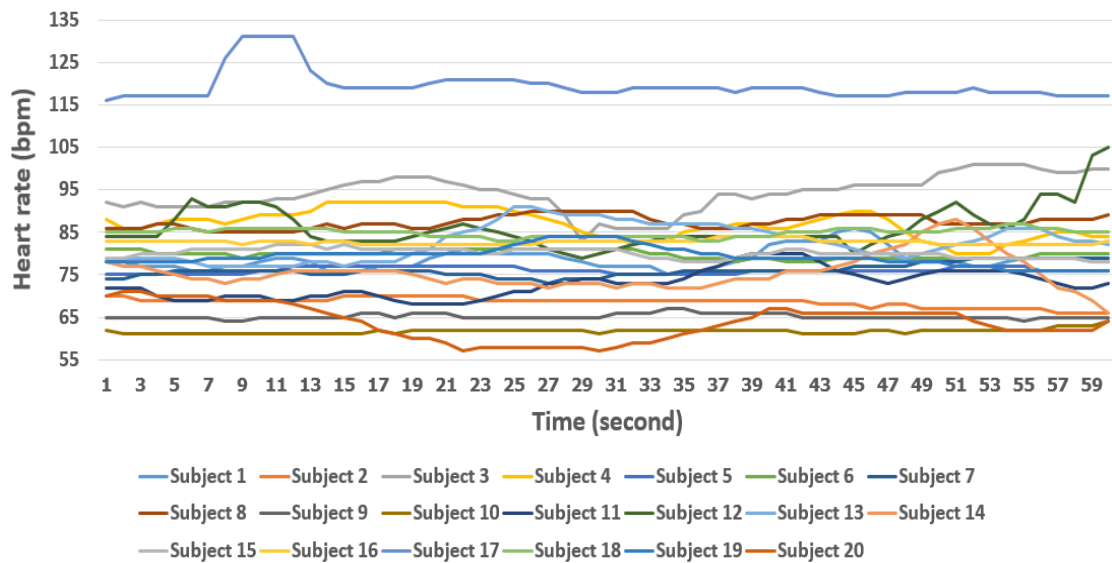
รูปที่ 36 a) กล่องควบคุมแสง b) ขณะบันทึกวิดีโอด้วยกล่องควบคุมแสงในห้องมืด
c) วิดีโอเฟรมที่ได้จากการบันทึก



รูปที่ 37 ด้านซ้ายคือ เครื่องวัดความดันโลหิต ด้านขวาคือ Arduino pulse sensor



Actual heart rate of 20 subjects obtained from Arduino Pulse Sensor



รูปที่ 38 แสดงชีพจรที่บันทึกได้จาก Arduino Pulse Sensor ของผู้ทำการทดลอง 20 คน
ในเวลา 1 นาที

ตารางที่ 19 ข้อมูลความดันเลือดและชีพจรผู้ทำการทดลอง

ผู้ทำการทดลอง (Subject)	ข้อมูลผู้ทำการทดลอง				Digital Blood Pressure		Arduino		
	เพศ	อายุ (ปี)	น้ำหนัก (kg)	ส่วนสูง (cm)	ความดัน ตัวบน	ความดัน ตัวล่าง	ชีพจรจากทุกหน้าต่าง		
							Average	Max	Min
1	ชาย	25	78	178	121	73	78.65	86	75
2	หญิง	52	58	162	107	73	68.58	70	66
3	หญิง	10	40	140	108	58	94.23	101	83
4	ชาย	21	76	170	112	68	86.98	92	80
5	หญิง	42	55	157	130	80	76.03	77	75
6	หญิง	48	54	155	110	76	79.87	81	78
7	หญิง	49	53	162	118	72	76.10	79	73
8	ชาย	50	63	170	116	70	87.45	90	85
9	หญิง	53	53	153	160	86	65.25	67	64
10	หญิง	20	46	158	91	58	61.72	64	61
11	หญิง	19	50	166	110	53	72.85	80	68
12	หญิง	46	46	159	96	53	86.17	105	79
13	หญิง	43	50	166	95	63	82.75	91	77
14	ชาย	47	72	170	116	79	75.57	88	66
15	หญิง	40	59	165	92	63	80.12	82	78
16	หญิง	48	59	158	112	64	82.72	84	82
17	หญิง	55	95	156	140	85	119.48	131	116
18	หญิง	38	53	158	122	107	84.93	87	83
19	ชาย	52	64	165	110	78	79.38	84	76
20	หญิง	54	47	157	98	70	63.80	71	57

การบันทึกวิดีโอจะบันทึกด้วยอุปกรณ์ 2 อุปกรณ์ คือ กล้อง iPad3 และกล้องเว็บแคม Logitech C920 ตารางที่ 20 แสดงข้อเปรียบเทียบของทั้ง 2 อุปกรณ์ สำหรับกล้อง Logitech C920 จะบันทึกวิดีโอที่ความละเอียด 960 x 720 pixel เพื่อรักษาค่า frame rate ไว้ที่ 30 fps รูปที่ 39 แสดงตัวอย่างวิดีโอเฟรมที่บันทึกได้จากทั้ง 2 อุปกรณ์ ด้านซ้ายเป็นวิดีโอเฟรมจากกล้อง iPad3 และด้านขวาจากกล้อง Logitech C920



รูปที่ 39 อุปกรณ์บันทึกวิดีโอ ซ้าย iPad3 ขวา Logitech C920

ตารางที่ 20 ข้อเปรียบเทียบของทั้ง 2 อุปกรณ์

ข้อเปรียบเทียบ	กล้อง iPad	กล้อง Logitech C920
ความละเอียด	5 megapixel with iSight camera	15 megapixel with Carl-zeiss lens
ความละเอียดวิดีโอที่บันทึก	1920 x 1080 pixel	960 x 760 pixel
FPS	30	30

5.2. ผลการดำเนินการวิจัย

การแสดงผลการดำเนินการวิจัยจะแสดงการเปรียบเทียบผลลัพธ์การระบุค่าชีพจรที่ได้จากแต่ละวิธีกับค่าชีพจรที่ได้จาก Arduino pulse sensor ซึ่งจะแสดงการเปรียบเทียบ 2 ลักษณะคือ 1. ในหัวข้อที่ 5.2.1 จะแสดงการเปรียบเทียบผลลัพธ์การวัดค่าชีพจรแยกในแต่ละผู้ทำการทดลองและโมเดลสี และ 2. สำหรับหัวข้อที่ 5.2.2 - 5.2.7 จะแสดงการเปรียบเทียบโดยหาค่าเฉลี่ยจากผู้ทำการทดลองทั้ง 20 คนในรูปแบบตาราง ผลลัพธ์การเปรียบเทียบทั้ง 2 แบบจะถูกแสดงบางส่วนตามความเหมาะสม และส่วนที่เหลือจะแสดงไว้ในแผ่น CD และแหล่งเก็บข้อมูลออนไลน์ (https://drive.google.com/a/silpakorn.edu/folderview?id=B_NPbewWpv9GT0duRjJuaHA5eXM&usp=sharing)

สำหรับการเปรียบเทียบแบบตารางจะเปรียบเทียบจากค่าสัมบูรณ์ของผลต่างของชีพจรที่วัดได้กับชีพจรจริงจากแต่ละหน้าตตามสมการที่ 4 จากตารางที่ 21 แสดงค่าผลต่างจากแต่ละหน้าต่านำค่าที่ได้มาคำนวณผลต่างเฉลี่ย (Mean), ส่วนเบี่ยงเบนมาตรฐาน (SD), ผลต่างที่มากที่สุด (Max), จำนวนครั้งที่ผลต่างมีค่ามากกว่า 3 bpm (Count 3 : C3), จำนวนครั้งที่ผลต่างมากกว่า 3 bpm ติดต่อกัน 2 ครั้ง (Continue error 3 : CE3), จำนวนครั้งที่ผลต่างมีค่ามากกว่า 10 bpm (Count 10 : C10) และ จำนวนครั้งที่ผลต่างมากกว่า 10 bpm ติดต่อกัน 2 ครั้ง (Continue error 10 : CE10)

$$Diff = |predictedPulse - realPulse|$$

สมการที่ 4 การคำนวณค่าผลต่างการระบุค่าชีพจร

ตารางที่ 21 แสดงผลลัพธ์การคำนวณผลต่างของชีพจรที่วัดได้กับ Arduino pulse sensor

หน้าต่าที่	1	2	3	4	5
ชีพจรที่วัดได้ (predicted Pulse)	91.5	91.5	88.5	85.5	85.5
ชีพจรจาก Arduino (real Pulse)	95.35	96.55	92.65	88.55	87.45
Diff	3.85	5.05	4.15	3.05	1.95

จากตารางที่ 21 ค่าผลต่างเฉลี่ย (Mean) มีค่าเท่ากับ 3.61 bpm ค่าส่วนเบี่ยงเบนมาตรฐานคือ 1.17 bpm ผลต่างที่มากที่สุดคือ 5.05 bpm สำหรับ C3 และ C10 จะคำนวณโดยเปรียบเทียบจากจำนวนครั้งที่มากที่สุดที่ผลต่างจะมากกว่า 3 และ 10 bpm ตามลำดับ สำหรับ CE3 และ CE10 จะเปรียบเทียบกับโอกาสที่จะเกิดความผิดพลาดต่อเนื่องได้จากตารางที่ 21 จำนวนครั้งที่ผิดพลาดได้มากที่สุดคือ 5 ซึ่งมีค่าเท่ากับจำนวนหน้าต่าง พบว่าจำนวนครั้งที่ผลต่างมากกว่า 3 และ 10 คือ 4 และ 0 ครั้ง ดังนั้น ค่า C3 จะมีค่าเท่ากับ $\frac{4 \times 100}{5} = 80\%$ และค่า C10 มีค่าเท่ากับ $\frac{0 \times 100}{5} = 0\%$ จากรูปที่ 40 แสดงวิธีการคำนวณ CE สำหรับ CE3 พบว่าการระบุค่าชีพจรที่ผิดพลาดมากกว่า 3 bpm ต่อเนื่องกันทั้งหมด 3 ครั้ง โดยสามารถผิดพลาดต่อเนื่องได้มากที่สุด 4 ครั้ง ซึ่งมีค่าเท่ากับจำนวนหน้าต่างทั้งหมด - 1 ดังนั้นค่า CE3 เท่ากับ $\frac{3 \times 100}{4} = 75\%$ และค่า CE10 เท่ากับ $\frac{0 \times 100}{4} = 0\%$ เนื่องจากไม่มีการระบุค่าชีพจรเกิน 10 bpm

หน้าต่างที่	ผลต่าง
1	3.85
2	5.05
3	4.15
4	3.05
5	1.95

ผิดพลาดมากกว่า 3 bpm ต่อเนื่องครั้งที่ 1

ผิดพลาดมากกว่า 3 bpm ต่อเนื่องครั้งที่ 2

ผิดพลาดมากกว่า 3 bpm ต่อเนื่องครั้งที่ 3

รูปที่ 40 แสดงการคำนวณ CE3

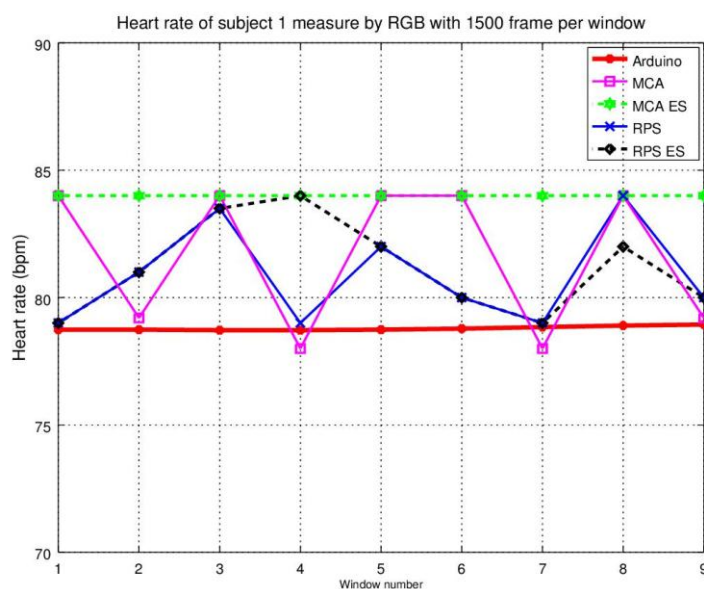
5.2.1. กราฟชีพจร

ส่วนนี้จะแสดงกราฟชีพจรที่วัดได้เปรียบเทียบกับค่าชีพจรที่ได้จาก Arduino pulse sensor แยกตามผู้ทำการทดลองและโมเดลสี โดยจะแสดงการเปรียบเทียบเฉพาะผู้ทำการทดลองที่ 1 (ชาย) และ 2 (หญิง) เปรียบเทียบการเลือกองค์ประกอบ 3 แบบ คือ เลือกองค์ประกอบที่ 2 ของการทำ ICA, เลือกความถี่ที่มีค่า Magnitude สูงสุด และ เลือกใช้ทุกองค์ประกอบด้วย MCA และ RPS (วิธีที่

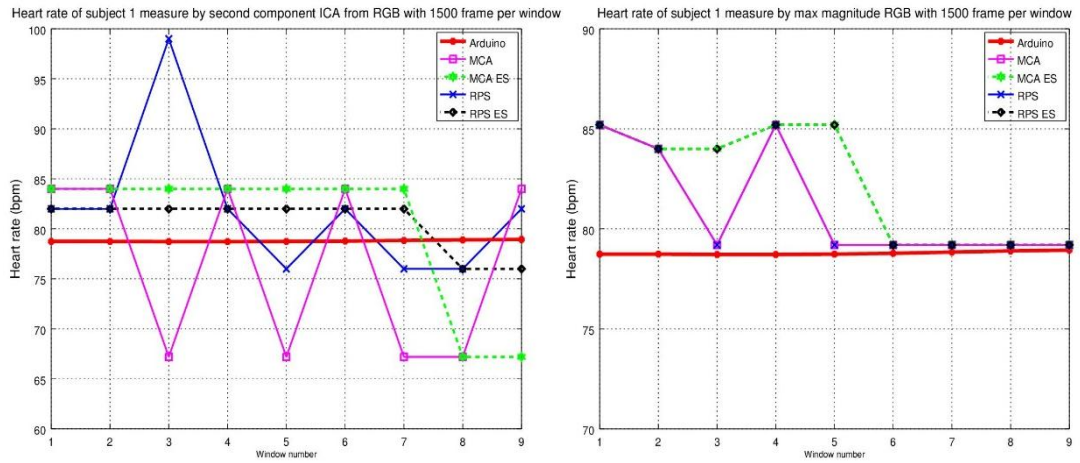
นำเสนอ) โดยเปรียบเทียบผลลัพธ์การวัดค่าชีพจรด้วยขนาดหน้าต่าง 50 วินาที/หน้าต่าง (1500 frame/window) ที่ได้จากโมเดลสี RGB, HSV, CIE L*a*b*, MMS และ CMU เท่านั้น การเปรียบเทียบส่วนอื่นจะแสดงไว้ในแสดงไว้ในแผ่น CD และแหล่งเก็บข้อมูลออนไลน์ (https://drive.google.com/a/silpakorn.edu/folderview?id=B_NPbewWpv9GT0duRjJuaHA5eXM&usp=sharing) กราฟการเปรียบเทียบแต่ละกราฟจะประกอบด้วยกราฟชีพจร 5 กราฟ คือ

- Arduino : สัญญาณชีพจรที่วัดจาก Arduino
- MCA : สัญญาณชีพจรที่วัดจากสัญญาณที่ผ่านการทำ ICA และ MCA
- MCA ES : สัญญาณชีพจรที่วัดจากสัญญาณที่ผ่านการทำ ICA, MCA และ ES
- RPS : สัญญาณชีพจรที่วัดจากสัญญาณที่ผ่านการทำ ICA, MCA และ RPS
- RPS ES : สัญญาณชีพจรที่วัดจากสัญญาณที่ผ่านการทำ ICA, MCA, RPS และ ES

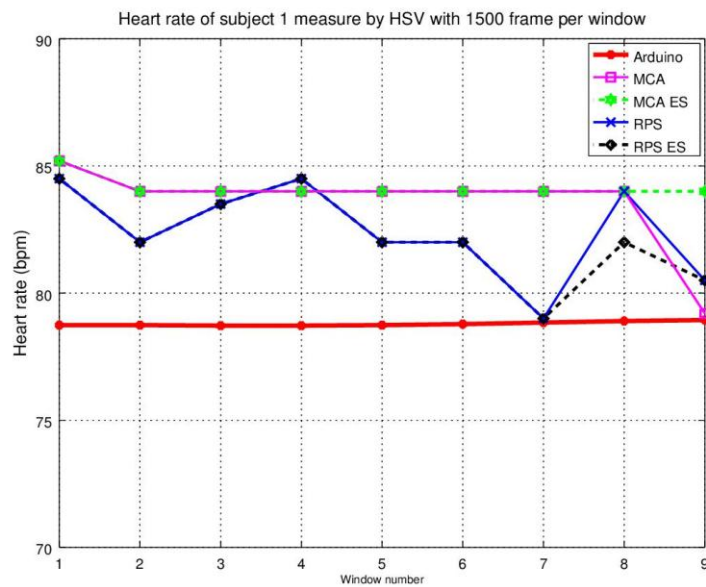
5.2.1.1. กราฟชีพจรของผู้ทำการทดลองคนที่ 1



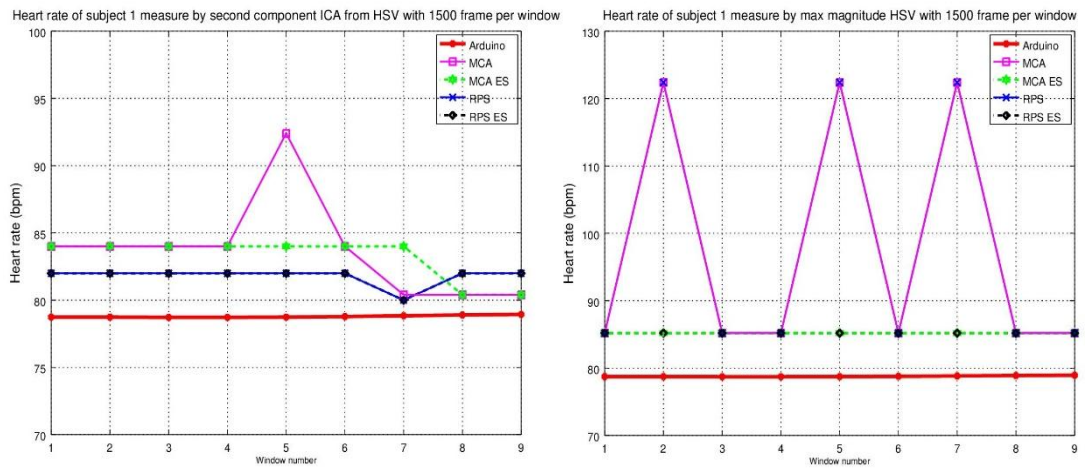
รูปที่ 41 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี RGB ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS



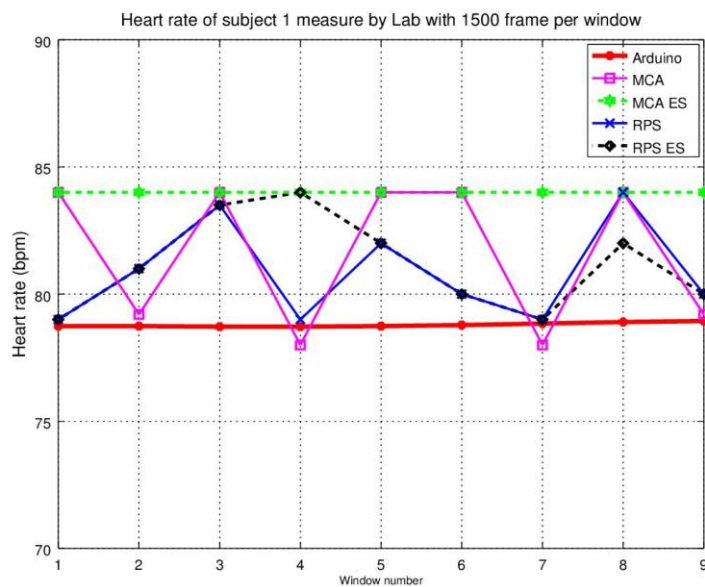
รูปที่ 42 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี RGB ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ซ้าย) และ 2011 (ขวา)



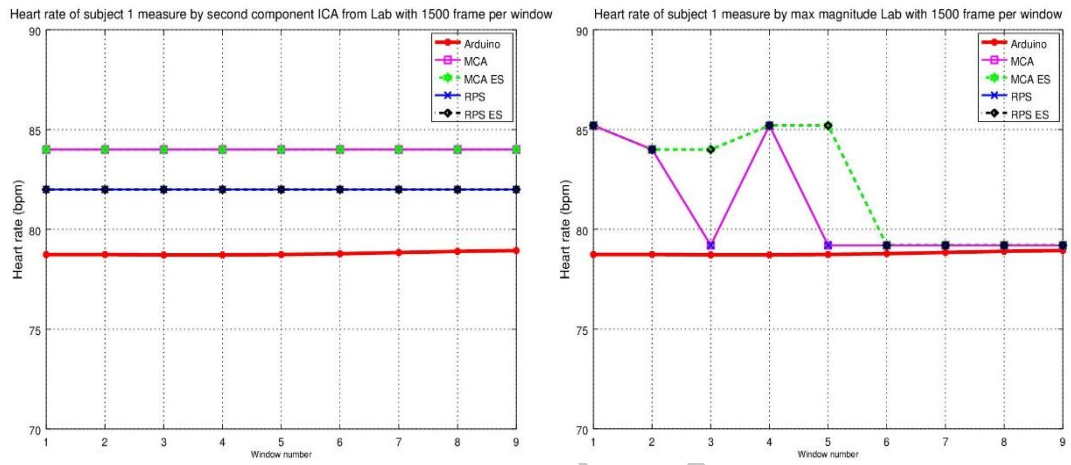
รูปที่ 43 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี HSV ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS



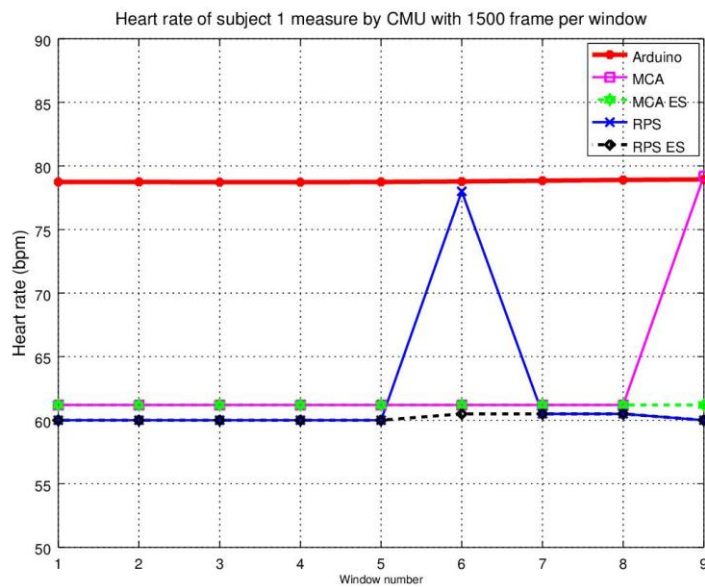
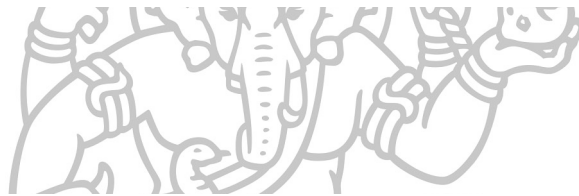
รูปที่ 44 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี HSV ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ซ้าย) และ 2011 (ขวา)



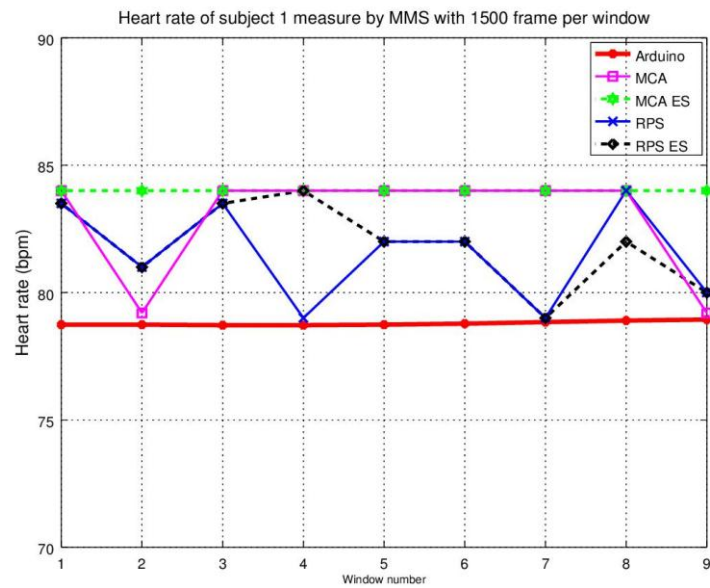
รูปที่ 45 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี CIE L*a*b* ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS



รูปที่ 46 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี CIE L*a*b* ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ชาย) และ 2011 (ขวา)

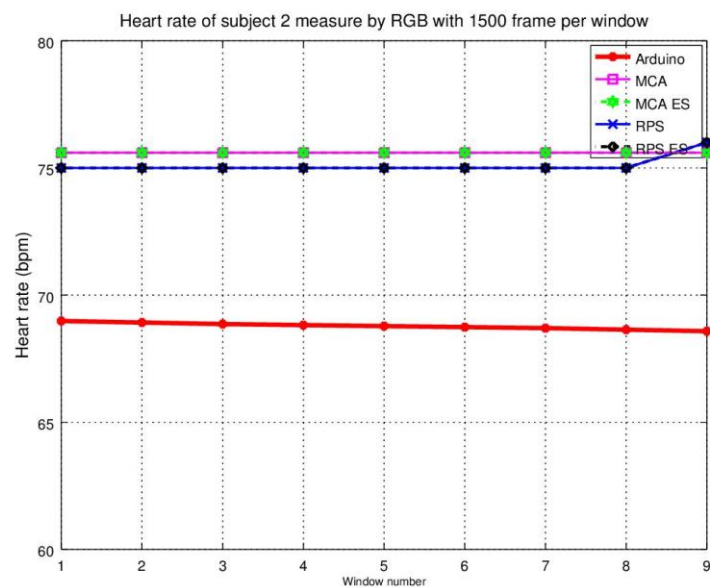


รูปที่ 47 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยการผสมโมเดลสีแบบ CMU ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS

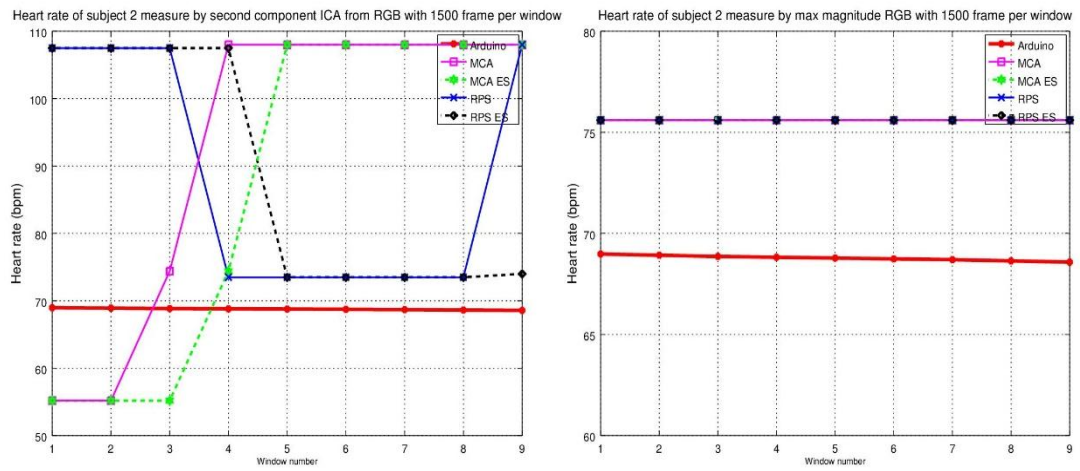


รูปที่ 48 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยการผสมโมเดลสี่แบบ MMS ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS

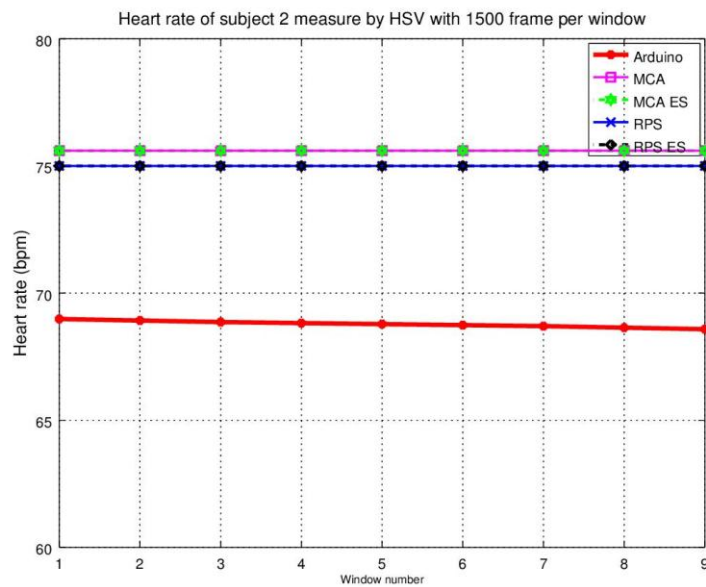
5.2.1.2. กราฟชีพจรของผู้ทำการทดลองคนที่ 2



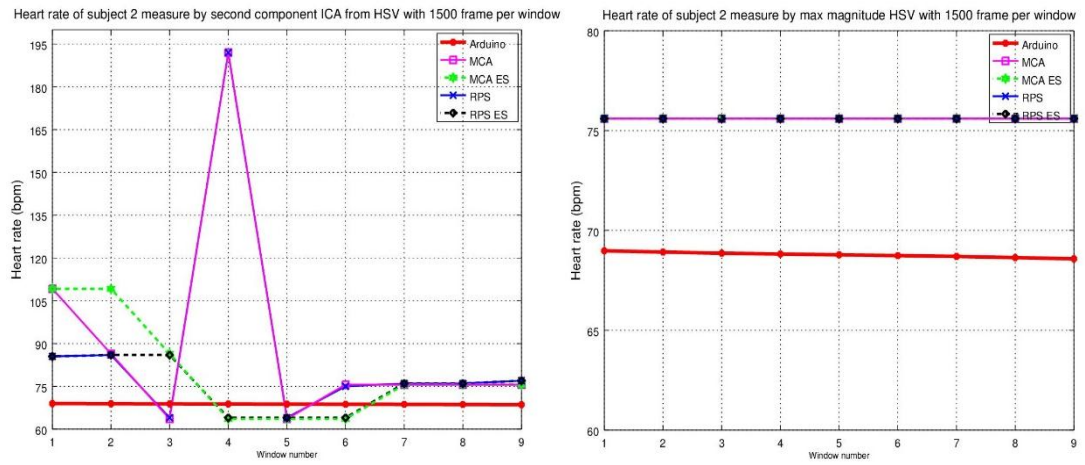
รูปที่ 49 กราฟชีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี่ RGB ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS



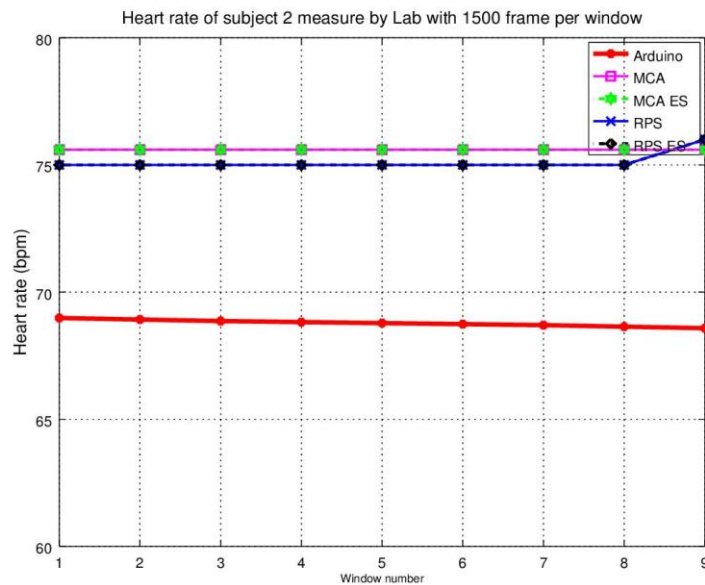
รูปที่ 50 กราฟชีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี RGB ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ชาย) และ 2011 (สาว)



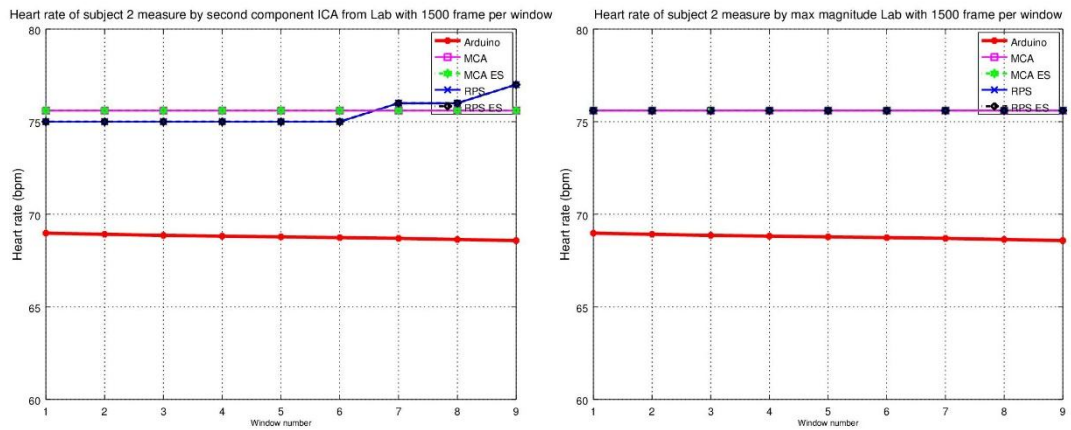
รูปที่ 51 กราฟชีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี HSV ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS



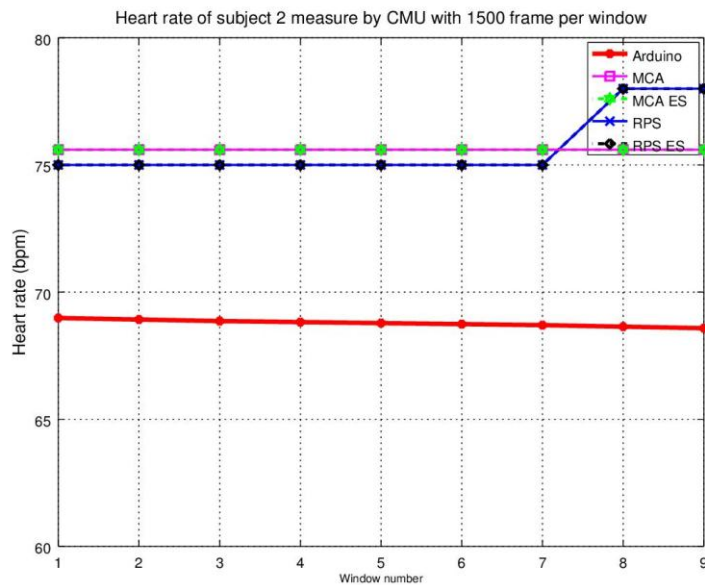
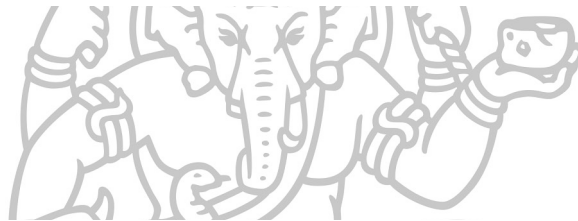
รูปที่ 52 กราฟชีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี RGB ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ชาย) และ 2011 (ขวา)



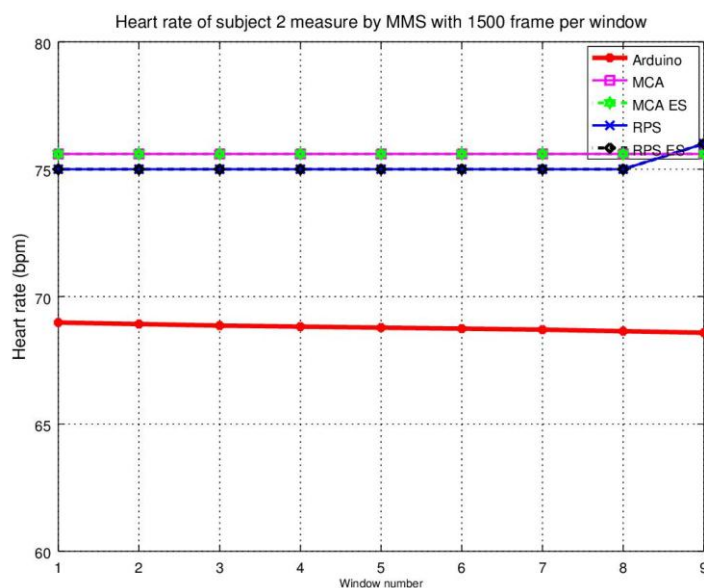
รูปที่ 53 กราฟชีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี CIE L*a*b* ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS



รูปที่ 54 กราฟชีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยโมเดลสี CIE L*a*b*
ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธีการของ Poh 2010 (ซ้าย) และ 2011 (ขวา)



รูปที่ 55 กราฟชีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยการผสมโมเดลสีแบบ CMU
ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS



รูปที่ 56 กราฟชีพจรของผู้ทำการทดลองคนที่ 2 วัดด้วยการผสมโมเดลสี่แบบ MMS ขนาดหน้าต่าง 50 วินาที/หน้าต่าง และใช้วิธี MCA และ RPS

จากรูปกราฟชีพจรพบว่าการเลือกใช้เพียงองค์ประกอบที่ 2 ของการทำ ICA เพียงองค์ประกอบเดียวไม่เพียงพอต่อการวัดชีพจร เนื่องจากความถี่ชีพจรอาจไม่อยู่ในองค์ประกอบที่ 2 หรือมีค่า Magnitude ต่ำ ยกเว้นกรณีใช้โมเดลสี่ CIE L*a*b* จะส่งผลให้วัดค่าชีพจรได้แม่นยำ ดังนั้นการเลือกใช้ทุกองค์ประกอบของการทำ ICA มาคำนวณชีพจรจึงมีความแม่นยำสูงกว่า เพราะจะนำสัญญาณชีพที่ปรากฏอยู่ในองค์ประกอบอื่นมาคำนวณด้วย แต่การนำทุกองค์ประกอบมาคำนวณจะเป็นการนำองค์ประกอบที่ไม่เกี่ยวข้องกับสัญญาณชีพมาคำนวณด้วยเช่นกัน จึงอาจส่งผลให้ความแม่นยำต่ำลงเป็นสิ่งที่เกิดขึ้นกับ CMU ที่นำทุกโมเดลมารวมกันก่อนแล้วจึงคำนวณ ICA อาจทำให้ไม่สามารถวิเคราะห์ความถี่การเปลี่ยนแปลงของสัญญาณชีพที่เกิดจากการหาความสัมพันธ์ของแต่ละเซนแนลได้ แต่เมื่อนำผลลัพธ์ความถี่ที่ได้จากแต่ละโมเดลมารวมกันตามหลักการของ MMS ส่งผลให้การระบุความถี่ชีพจรแม่นยำขึ้น

การทำ MCA และ RPS มีลักษณะของกราฟที่ใกล้เคียงกัน โดย RPS จะดีกว่าเล็กน้อย การทำ ES ส่งผลให้วัดค่าชีพจรได้แม่นยำมากขึ้น กราฟการเปลี่ยนแปลงชีพจรเป็นไปในทางเดียวกับกราฟชีพจรจาก Arduino

5.2.2. ผลกระทบจาก MCA

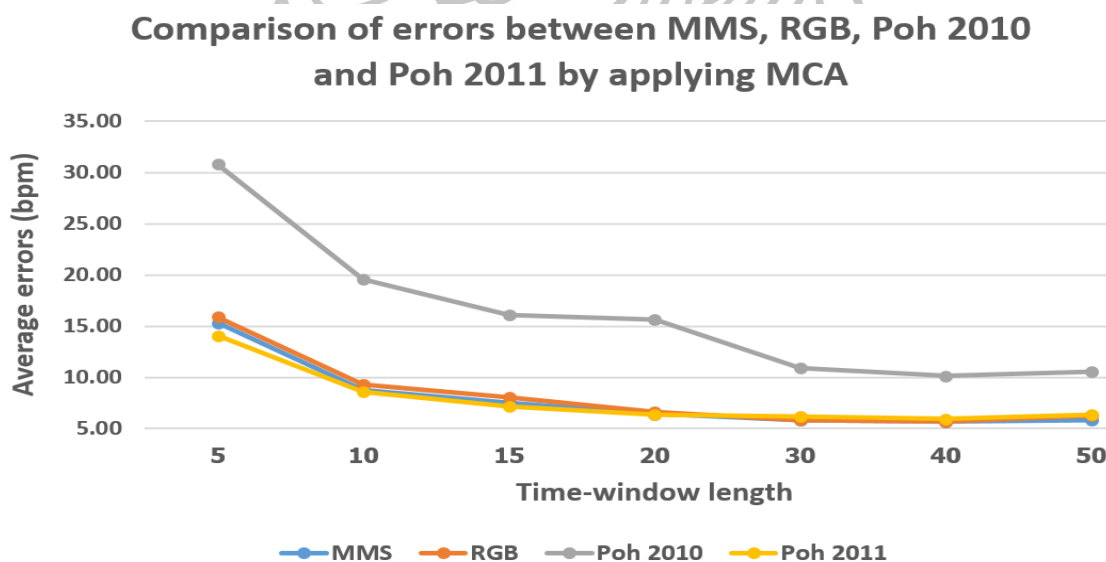
การวัดค่าซีพจรโดยวิเคราะห์จากทุกองค์ประกอบด้วยการหาความถี่ที่เหมาะสมส่งผลให้สามารถวัดค่าซีพจรได้ใกล้เคียงกว่าการเลือกใช้อีองค์ประกอบใดองค์ประกอบหนึ่ง ตารางที่ 22 แสดงการเปรียบเทียบวิธีที่นำเสนอ (MCA), การเลือกใช้อีองค์ประกอบที่ 2 ของการทำ ICA และการเลือกใช้อีองค์ประกอบที่มีค่า Magnitude สูงสุด โดยกำหนดการทดลองตามที่งานวิจัยของ Poh กำหนดคือเลือกใช้โมเดลสี RGB และเลือกใช้ขนาดหน้าต่างต่าง 20 และ 30 วินาทีต่อหน้าต่าง

ตารางที่ 22 ผลกระทบจากวิธี MCA ต่อความแม่นยำ

Method	Average	StdDev	Max
20 second/window			
MMS with MCA	6.79	4.52	19.69
RGB with MCA	7.05	5.19	23.07
RGB (Poh 2010)	16.12	17.18	65.34
RGB (Poh 2011)	6.73	4.31	19.38
30 second/window			
MMS with MCA	6.02	3.50	14.28
RGB with MCA	6.10	3.56	14.57
RGB (Poh 2010)	11.69	12.64	45.69
RGB (Poh 2011)	6.34	3.64	14.34
Average from all time-window lengths			
MMS with MCA	8.59	6.07	29.38
RGB with MCA	9.11	7.18	33.61
RGB (Poh 2010)	18.25	18.59	70.11
RGB (Poh 2011)	8.40	6.04	29.47
Average from all time-window lengths NOT INCLUDED 5 second/window			
MMS with MCA	7.02	4.24	19.25
RGB with MCA	7.38	5.09	23.17
RGB (Poh 2010)	15.19	15.87	60.15
RGB (Poh 2011)	7.03	4.42	20.03

จากตารางพบว่าการเลือกใช้เพียงองค์ประกอบที่ 2 ของการทำ ICA มีความแม่นยำต่ำสุด (ค่าในตารางสูงที่สุด) เมื่อเปรียบเทียบวิธีผลของ Poh 2011, MCA และ MMS พบว่าในกรณีขนาดหน้าต่าง 20 วินาทีต่อหน้าต่าง วิธีของ Poh 2011 มีความแม่นยำสูงกว่า แต่ในขนาดหน้าต่าง 30 วินาที วิธี MCA และ MMS มีความแม่นยำสูงกว่าวิธีของ Poh เมื่อพิจารณาค่าเฉลี่ยความผิดพลาดทุกขนาดหน้าต่าง (All time-window lengths) พบว่าวิธีการของ Poh 2011 มีความแม่นยำสูงกว่าเล็กน้อย เมื่อตัดส่วนที่มีความผิดพลาดสูงออก คือ ขนาดหน้าต่าง 5 วินาทีต่อหน้าต่าง พบว่าวิธี MMS และ Poh 2011 มีค่าความผิดพลาดเฉลี่ยใกล้เคียงกัน

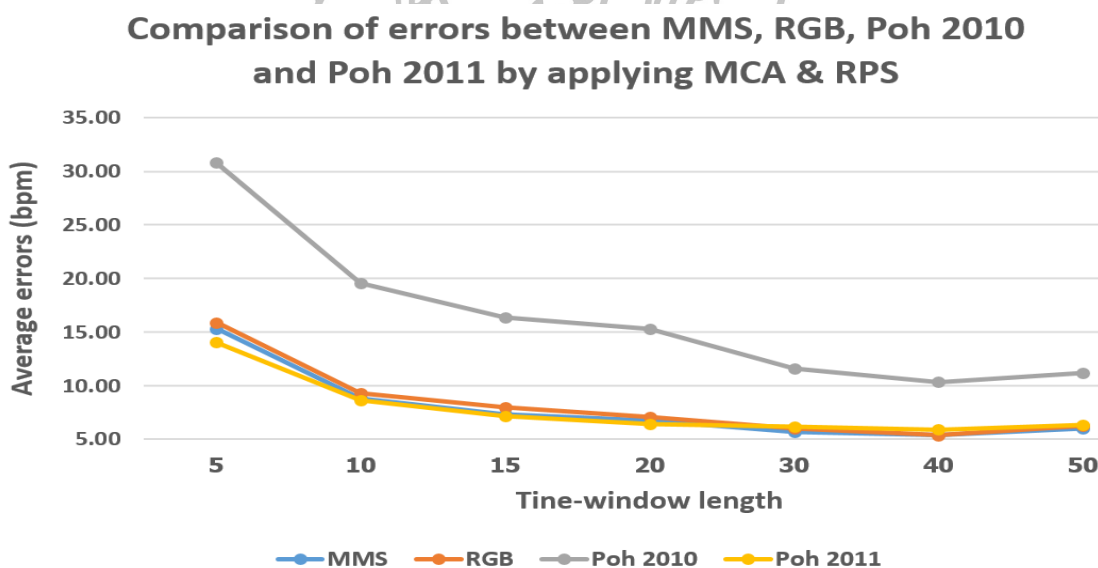
รูปที่ 57 แสดงการเปรียบเทียบความแม่นยำของทั้ง 4 หลักการ พบว่าเมื่อพิจารณาแยกแต่ละขนาดหน้าต่างการเลือกใช้หลักการของ Poh 2011 (Maximum magnitude), MCA, และ MMS มีค่าใกล้เคียงกัน ดังนั้นการเลือกองค์ประกอบด้วยวิธี MCA อาจไม่ทำให้ความแม่นยำในการระบุค่าชีพจรดีกว่าวิธีการของ Poh แต่เป็นการเปิดโอกาสให้สามารถใช้วิธีการ RPS เพื่อเพิ่มความแม่นยำในการระบุค่าชีพจรได้



รูปที่ 57 เปรียบเทียบความแม่นยำระหว่าง MMS, RGB (MCA), ICA Component 2 (Poh 2010) และ Maximum Magnitude (Poh 2011) โดยใช้หลักการ MCA เพียงอย่างเดียว

5.2.3. ผลกระทบจาก RPS

การรวมค่า Magnitude ของความถี่ที่ใกล้เคียงกัน (ห่างกันไม่เกิน 2 bpm) ด้วยหลักการของ RPS ทำให้การระบุค่าซีพจรใกล้เคียงกับค่าซีพจรจริงมากขึ้น เนื่องจากค่าซีพจรมีการเปลี่ยนแปลงตลอดเวลาและเปลี่ยนแปลงได้สูงสุด 10 bpm ดังนั้นการระบุค่าซีพจรด้วยค่าเพียงค่าเดียวจึงไม่สอดคล้องกับค่าซีพจรจริง ผลการทดลองส่วนนี้แสดงการเปรียบเทียบการระบุค่าซีพจรด้วยวิธี MCA & RPS, การเลือกใช้องค์ประกอบที่ 2 ของการทำ ICA และการเลือกใช้องค์ประกอบที่มีค่า Magnitude สูงสุด โดยกำหนดการทดลองเหมือนกับหัวข้อที่ 5.2.2 สำหรับวิธีการของ Poh 2010 สามารถนำมาวิเคราะห์ร่วมกับ RPS ได้เนื่องจากพิจารณาค่าความถี่ที่องค์ประกอบที่ 2 ต่างจากวิธีการของ Poh 2011 ที่เลือกเฉพาะค่าความถี่ที่มีค่า Magnitude สูงสุดเพียงค่าเดียวเท่านั้นจึงไม่สามารถวิเคราะห์ RPS ร่วมด้วยได้



รูปที่ 58 เปรียบเทียบความแม่นยำระหว่าง MMS, RGG (MCA & RPS), ICA Component 2 (Poh 2010) และ Maximum Magnitude (Poh 2011) โดยใช้หลักการ MCA & RPS เพียงอย่างเดียว

ตารางที่ 23 ผลกระทบจากวิธี RPS ต่อความแม่นยำ

Method	Average	StdDev	Max
20 second/window			
MMS with MCA & RPS	6.88	4.28	17.78
RGB with MCA & RPS	7.17	4.60	17.62
RGB (Poh 2010)	16.04	17.02	64.06
RGB (Poh 2011)	6.73	4.31	19.38
30 second/window			
MMS with MCA & RPS	5.85	3.07	10.96
RGB with MCA & RPS	6.21	3.78	13.85
RGB (Poh 2010)	12.02	13.30	47.74
RGB (Poh 2011)	6.34	3.64	14.34
Average from all time-window lengths			
MMS with MCA & RPS	8.45	5.95	27.96
RGB with MCA & RPS	9.07	6.98	31.91
RGB (Poh 2010)	18.34	18.61	69.93
RGB (Poh 2011)	8.40	6.04	29.47
Average from all time-window lengths NOT INCLUDED 150 frame/window			
MMS with MCA & RPS	6.86	4.10	17.59
RGB with MCA & RPS	7.34	4.86	21.19
RGB (Poh 2010)	15.30	15.90	59.94
RGB (Poh 2011)	7.03	4.42	20.03

จากตารางที่ 23 พบว่าการเลือกใช้เพียงองค์ประกอบที่ 2 ของการทำ ICA เมื่อวิเคราะห์ร่วมกับ RPS ส่งผลให้มีความแม่นยำสูงขึ้นแต่ยังคงมีความแม่นยำต่ำสุดใน 3 วิธี เมื่อเปรียบเทียบวิธีของ Poh 2011, MCA และ MMS ส่งผลเหมือนกับหัวข้อ 5.2.2 แต่เมื่อพิจารณาการตัดส่วนที่มีความผิดพลาดสูงออก คือ 5 วินาทีต่อหน้าต่าง พบว่ามีความแม่นยำสูงกว่า Poh 2011 รูปที่ 58 แสดงการเปรียบเทียบการวัดชีพจร โดยมีผลลัพธ์ใกล้เคียงกับรูปที่ 57

5.2.4. ผลกระทบจาก ES

การจำกัดการระบุค่าชีพจรผิดพลาดเนื่องจากการเปลี่ยนแปลงของชีพจรที่มากเกินไป โดยเปรียบเทียบผลลัพธ์ที่ได้กับหน้าต่างก่อนหน้า ส่งผลเด่นชัดในส่วนของค่าผลต่างที่มากที่สุด (Max) ผลการทดลองของ ES จะนำมาเปรียบเทียบ 3 วิธีคือ 1. การวัดค่าชีพจรโดยไม่ทำ RPS (ทำเฉพาะ MCA และ ES) 2. ทำ RPS ร่วมด้วย (ทำทั้ง 3 วิธี) และ 3. เปรียบเทียบวิธีการทำ ES ที่ปรับปรุงใหม่กับวิธีของ Poh 2010 ผลลัพธ์ในหัวข้อนี้เกิดจากการหาค่าเฉลี่ยจากผู้ทำการทดลองทุกคน, อุปกรณ์บันทึกทุกอุปกรณ์, ขนาดหน้าต่างทุกขนาด และใช้การผสมโมเดลสี่แบบ MMS

5.2.4.1. เปรียบเทียบความแม่นยำระหว่าง MCA และ MCA&ES (ไม่ทำ RPS)

ผลการทดลองส่วนนี้จะแสดงผลกระทบจากการทำ ES สำหรับวิธี MCA พบว่าการทำ ES ส่งผลให้การระบุค่าชีพจรแม่นยำขึ้น โดยค่าความผิดพลาดเฉลี่ย (Average) ลดลง 0.34 bpm และค่าผลต่างที่มากที่สุด (Max) ลดลง 4.91 bpm ดังแสดงในตารางที่ 24

ตารางที่ 24 เปรียบเทียบความแม่นยำการทำและไม่ทำ ES โดยทำเพียง MCA สำหรับโมเดลสี่ MMS

Methods	Errors (bpm)		
	Average	StdDev	Max
MCA	7.02	4.24	19.25
MCA & ES	6.67	3.37	14.34
Improvements if using ES	0.34	0.87	4.91

5.2.4.2. เปรียบเทียบความแม่นยำระหว่าง MCA&RPS และ MCA&RPS&ES

ตารางที่ 25 แสดงผลการเปรียบเทียบการวัดค่าชีพจร เมื่อทำ RPS ร่วมกับการทำ MCA ทำให้สามารถระบุค่าชีพจรได้แม่นยำขึ้น 0.16 bpm (เทียบกับ ตารางที่ 24) และเมื่อทำ ES ส่งผลให้ค่าความผิดพลาดเฉลี่ย (Average) ลดลง 0.27 bpm และค่าผลต่างที่มากที่สุด (Max) ลดลง 4.05 bpm

ตารางที่ 25 เปรียบเทียบความแม่นยำการทำและไม่ทำ ES โดยทำ MCA ร่วมกับ RPS สำหรับโมเดล
สี่ MMS

Methods	Errors (bpm)		
	Average	StdDev	Max
MCA & RPS	6.86	4.10	17.59
MCA & RPS & ES	6.59	3.41	13.54
Improvements if using ES	0.27	0.68	4.05

5.2.4.3. เปรียบเทียบความแม่นยำระหว่าง ES ที่นำเสนอ และ ES ของ Poh

จาก 2 หัวข้อข้างต้นพบว่าการทำ ES ส่งผลให้การระบุค่าชีพจรมีความแม่นยำสูงขึ้น ซึ่ง ES ที่นำเสนอเป็นการปรับปรุงจาก ES ของ Poh ดังนั้นผลการทดลองส่วนนี้จึงเป็นการเปรียบเทียบความแม่นยำของ ES ที่นำเสนอกับวิธีการของ Poh แสดงในตารางที่ 26 พบว่า ES ที่ปรับปรุงทำให้ความแม่นยำในการระบุชีพจรสูงขึ้น ในกรณีที่ทำเพียง MCA พบว่า ES ที่นำเสนอทำให้ค่าความผิดพลาดเฉลี่ยลดลง 1.23 bpm และเมื่อทำ MCA ร่วมกับ RPS ทำให้ความผิดพลาดเฉลี่ยลดลง 1.30 bpm

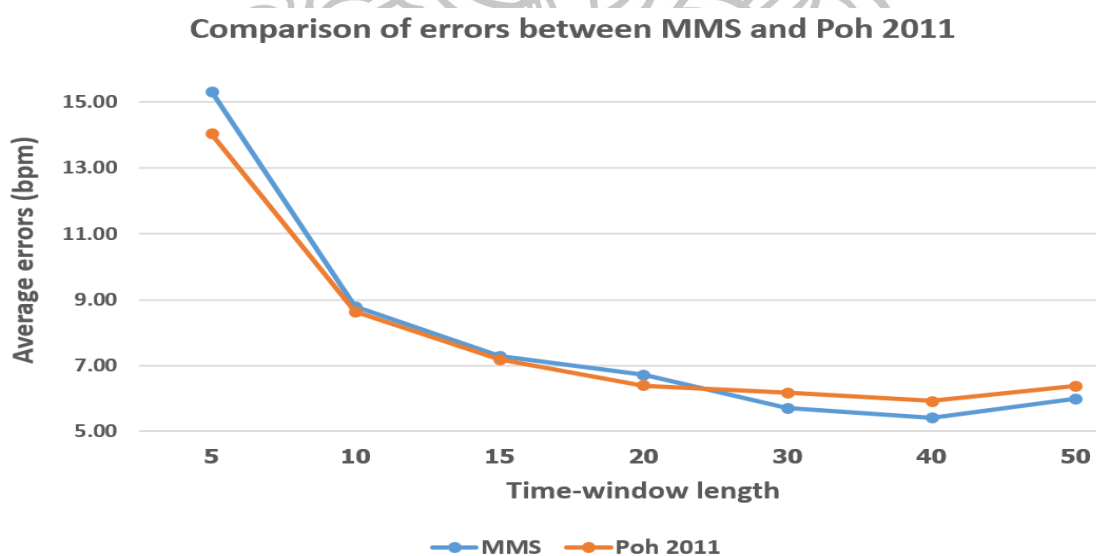
ตารางที่ 26 เปรียบเทียบความแม่นยำระหว่าง ES ของ Poh กับ ES ที่นำเสนอ

MCA with Poh's ES			MCA with proposed ES			Improvements if using proposed ES		
Avg	StdDev	Max	Avg	StdDev	Max	Avg	StdDev	Max
7.90	4.57	19.88	6.67	3.37	14.34	1.23	1.20	5.54
MCA&RPS with Poh's ES			MCA&RPS with proposed ES			Improvements if using proposed ES		
Avg	StdDev	Max	Avg	StdDev	Max	Avg	StdDev	Max
7.89	4.58	19.34	6.59	3.41	13.54	1.30	1.17	5.80

5.2.5. ผลกระทบจากการเปลี่ยนขนาดหน้าต่าง

ผลการทดลองส่วนนี้แสดงการเปรียบเทียบผลของการเปลี่ยนขนาดหน้าต่างที่ได้ทำการตั้งสมมุติฐานไว้ว่าขนาดหน้าต่างที่สูงขึ้นส่งผลให้มีความแม่นยำมากขึ้น โดยผลการทดลองส่วนนี้จะเป็นการหาค่าเฉลี่ยจากผู้ทำการทดลองทุกคนและอุปกรณ์บันทึกวิดีโอทั้ง 2 อุปกรณ์ โดยเปรียบเทียบผลการผสมโมเดลสี MMS ที่มีความแม่นยำสูงสุด กับวิธีการของ Poh ในปี 2011

รูปที่ 59 แสดงความผิดพลาดเฉลี่ยในการระบุค่าชีพจรเมื่อเปลี่ยนขนาดหน้าต่าง พบว่าเมื่อเพิ่มความยาวของขนาดหน้าต่างทำให้ค่าความผิดพลาดเฉลี่ยต่ำลง โดยแนวโน้มการเปลี่ยนแปลงของทั้ง 2 แบบเป็นในทิศทางเดียวกัน หน้าต่างขนาด 40 วินาทีต่อหน้าต่าง มีความแม่นยำมากที่สุดโดยมีค่าความผิดพลาดเฉลี่ย 5.92 bpm สำหรับวิธีการของ Poh ในปี 2011 และ 5.42 bpm สำหรับการวัดค่าชีพจรด้วยการผสมโมเดลสีแบบ MMS เมื่อเพิ่มขนาดหน้าต่างวิธีการที่นำเสนอมีความแม่นยำสูงกว่าวิธีการของ Poh เนื่องจาก RPS สามารถนำข้อมูลที่มีประโยชน์มาใช้คำนวณค่าความถี่ได้มากขึ้น



รูปที่ 59 เปรียบเทียบความผิดพลาดในการระบุค่าชีพจร เมื่อเปลี่ยนขนาดหน้าต่าง

5.2.6. ผลกระทบจากการเปลี่ยนโมเดลสี

จากสมมุติฐานที่กล่าวไว้ข้างต้นว่าโมเดลสีมีผลต่อการวัดค่าชีพจรและการผสมโมเดลสีแบบ MMS ส่งผลให้การวัดค่าชีพจรมีความแม่นยำสูงขึ้น ดังนั้นผลการทดลองส่วนนี้จึงเปรียบเทียบผลการวัดชีพจรจากโมเดลสีที่ศึกษาและการผสมโมเดลสีประกอบด้วย โมเดลสีเดี่ยวทั้ง 7 โมเดล (RGB, HLS, HSV, CIE L*a*b*, CIE L*u*v*, TSL และ YCbCr) และการผสมโมเดลสี (CMU และ MMS เพื่อสรุปว่าโมเดลสีหรือหลักการใดมีความแม่นยำในการระบุค่าชีพจรสูงสุด โดยผลลัพธ์ส่วนนี้จะเป็นการเฉลี่ยจากผู้ทำการทดลองทุกคนและทุกขนาดหน้าต่าง ยกเว้นขนาดหน้าต่าง 5 วินาทีต่อหน้าต่าง เนื่องจากมีความผิดพลาดมากที่สุด

ตารางที่ 27 แสดงการเปรียบเทียบความแม่นยำในการระบุค่าชีพจรของแต่ละโมเดลสี โดยเฉลี่ยผลลัพธ์จากทุกขนาดหน้าต่างยกเว้นขนาด 5 วินาทีต่อหน้าต่าง

Color model	Errors from All time-window length NOT INCLUDED 5-second window length		
	Average	StdDev	Max
MMS	6.59	3.41	13.54
CMU	9.08	5.69	23.36
BGR	6.96	3.88	16.39
HLS	6.92	4.37	17.24
HSV	6.81	4.27	16.99
CIE L*a*b*	6.95	3.92	16.33
CIE L*u*v*	6.95	3.90	15.96
TSL	8.46	5.62	23.19
YCrCb	6.94	3.89	16.15

จากตารางที่ 27 พบว่าการผสมโมเดลสีแบบ MMS ส่งผลให้มีความแม่นยำสูงสุด โดยมีค่าความผิดพลาดเฉลี่ยอยู่ที่ 6.59 bpm ต่างจากการผสมโมเดลสีแบบ CMU ที่มีค่าความแม่นยำลดต่ำลงเนื่องจากนำองค์ประกอบที่ไม่เกี่ยวกับชีพจรมาคำนวณด้วย สำหรับการเลือกใช้โมเดลสีเดี่ยวพบว่าโมเดลสี HSV มีความแม่นยำสูงกว่าโมเดลสีอื่นๆ และโมเดลสีที่มีความแม่นยำต่ำสุดคือ TSL

5.2.7. ผลกระทบจากอุปกรณ์

ประสิทธิภาพเซนเซอร์กล้องของอุปกรณ์แต่ละตัวส่งผลกระทบต่อการวัดค่าชีพจร เนื่องจากความสามารถการวัดและบันทึกปริมาณค่าสีต่างกัน ตารางที่ 28 แสดงการเปรียบเทียบความผิดพลาดในการระบุค่าชีพจรของทั้ง 2 อุปกรณ์ (iPad3 และ Logitech C920) ในแต่ละขนาดหน้าต่าง ซึ่งวัดค่าชีพจรด้วยโมเดลสี MMS พบว่า Logitech C920 มีความแม่นยำสูงกว่ากล้อง iPad3 โดยมีค่าความผิดพลาดเฉลี่ยต่ำสุดที่ 4.48 bpm โดยใช้ขนาดหน้าต่าง 50 วินาทีต่อหน้าต่าง ต่างจากกล้อง iPad3 ที่มีค่าความผิดพลาดเฉลี่ยต่ำสุดที่ 6.38 bpm ที่วัดด้วย 30 วินาทีต่อหน้าต่าง

ตารางที่ 28 เปรียบเทียบความแม่นยำของ 2 อุปกรณ์ คือ iPad3 และ Logitech C920

Time-window lengths	Comparison of average errors between iPad3 and Logitech C920					
	iPad3			Logitech C920		
	Average	StdDev	Max	Average	StdDev	Max
5	18.85	19.39	94.77	17.15	14.77	85.57
10	9.93	10.26	50.14	9.34	4.87	23.18
15	8.36	8.95	38.17	7.31	3.39	15.62
20	7.15	6.35	25.22	6.42	2.69	14.16
30	6.38	5.66	20.17	5.66	1.34	8.39
40	6.61	3.64	12.80	5.07	0.99	6.75
50	7.47	1.92	10.52	4.48	0.81	5.86

บทที่ 6

สรุปผลการทดลอง และข้อเสนอแนะ

6.1. สรุปผลการทดลอง

การวัดชีพจรแบบไม่สัมผัสสร้างกายโดยพิจารณาการเปลี่ยนแปลงของสีที่เกิดจากการไหลเวียนของเลือดภายในวิดีโอเฟรมด้วยหลักการการวิเคราะห์องค์ประกอบอิสระ (ICA) สามารถนำมาวัดชีพจรได้ โดยมีค่าเฉลี่ยความผิดพลาดต่ำสุดคือ 5.52 bpm เมื่อวัดด้วยขนาดหน้าต่าง 40 วินาทีต่อหน้าต่าง (1200 frame/window) โมเดลแต่ละโมเดลส่งผลให้ความแม่นยำในการระบุค่าชีพจรแตกต่างกัน โดยโมเดลที่ส่งผลให้ได้ความแม่นยำสูงที่สุดคือ การผสมโมเดลแบบ MMS มีค่าความผิดพลาดเฉลี่ยจากทุกขนาดหน้าต่างเท่ากับ 6.59 bpm

การเลือกองค์ประกอบที่ 2 ของการทำ ICA (Poh 2010) และค่าความถี่ที่มีค่า Magnitude สูงสุด (Poh 2011) เพื่อคำนวณค่าชีพจรมีความแม่นยำโดยเฉลี่ยต่ำกว่าการเลือกใช้การผสมโมเดลแบบ MMS ซึ่งนำผลการวิเคราะห์จากทั้ง 3 องค์ประกอบจากการทำ ICA ของทุกโมเดลมารวมกัน และรวมค่า Magnitude ของความถี่ที่ต่างกันไม่เกิน 2 bpm ด้วย RPC โดย MMS มีความแม่นยำสูงกว่าการเลือกองค์ประกอบที่ 2 8.44 bpm และ 0.17 bpm เมื่อเทียบกับการเลือกค่าความถี่ที่มี Magnitude สูงสุด โดยเฉลี่ย ซึ่งในกรณีที่ใช้ขนาดหน้าต่างต่ำ (≤ 20 วินาที/หน้าต่าง) การเลือกใช้ค่าความถี่ที่มีค่า Magnitude สูงสุดมีความแม่นยำมากกว่าวิธีที่นำเสนอ อาจเนื่องมาจากการข้อมูลที่ใช้คำนวณ RPS มีไม่มากพอและนำองค์ประกอบที่ไม่เกี่ยวกับสัญญาณชีพมาคำนวณด้วยซึ่งส่งผลกระทบต่อ CMU อย่างชัดเจนที่มีความแม่นยำต่ำกว่าการเลือกใช้โมเดลเดียว

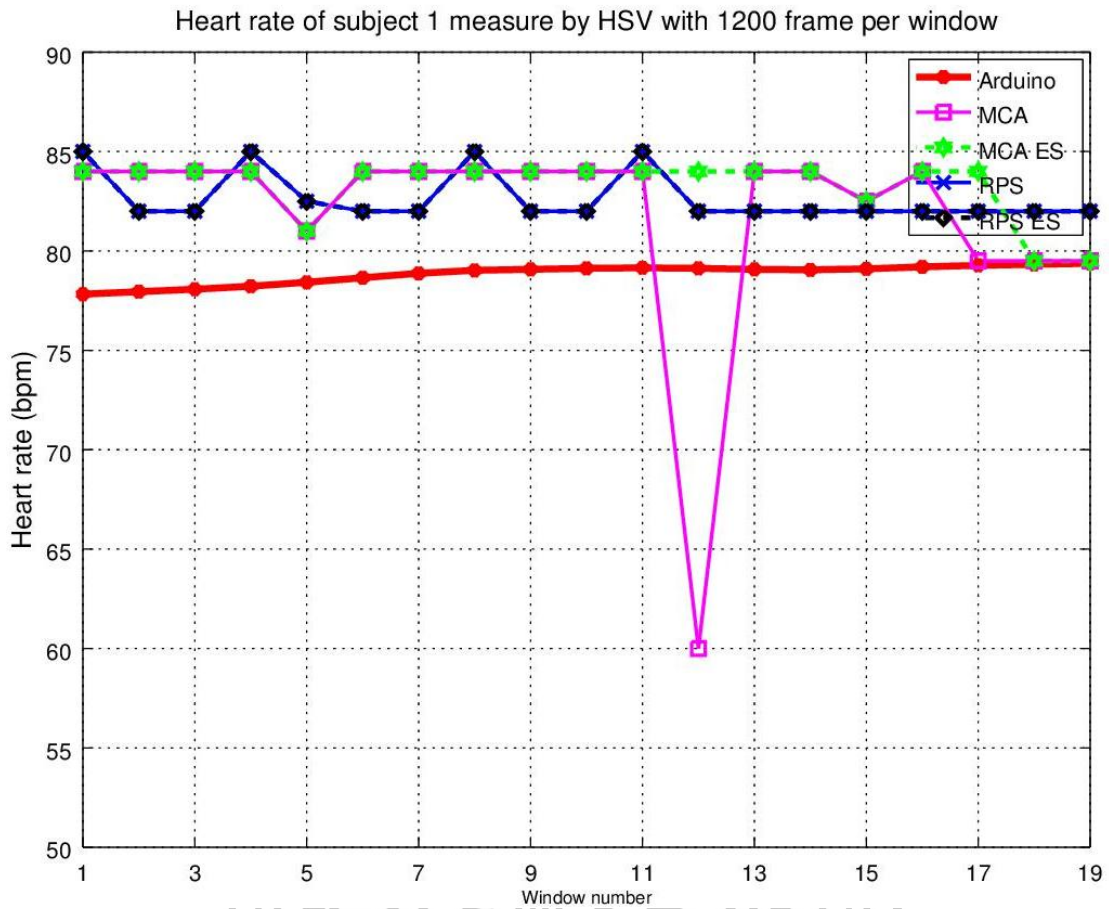
ขนาดหน้าต่างที่ใช้ในการแบ่งสัญญาณเพื่อวิเคราะห์ความถี่มีผลต่อประสิทธิภาพความแม่นยำ โดยเมื่อขนาดหน้าต่างสูงขึ้นส่งผลให้มีความแม่นยำมากขึ้น เนื่องจากข้อมูลที่นำมาประมวลผลมีมากขึ้น ขนาดหน้าต่างที่มีความแม่นยำมากที่สุดคือ 40 วินาทีต่อหน้าต่าง

การทำ MCA ทำให้การระบุค่าชีพจรแม่นยำขึ้น ผลต่างเฉลี่ย (Average) ลดลง 0.34 bpm และผลต่างการระบุชีพจรที่มากที่สุด (Max) ลดลง 4.91 bpm และการทำ RPS ทำให้ค่าผลต่างเฉลี่ย ลดลง 0.27 bpm และผลต่างที่มากที่สุดลดลง 4.05 bpm เมื่อทำ ES ร่วมด้วย ถึงแม้ค่าผลต่างเฉลี่ย จะลดลงไม่มากแต่ทำให้การระบุชีพจรในแต่ละครั้งใกล้เคียงกับค่าชีพจรจริงมากขึ้น เนื่องจากค่า Max ลดลงมาก ประสิทธิภาพกล้องส่งผลต่อการวัดค่าชีพจร ในกรณีนี้กล้อง Logitech C920 มีค่าความ แม่นยำมากกว่ากล้อง iPad3 โดยมีค่าความผิดพลาดเฉลี่ยต่ำสุดอยู่ที่ 4.48 bpm การปรับปรุง ES ทำให้สามารถระบุค่าชีพจรได้แม่นยำขึ้น 1.3 bpm

6.2. ข้อเสนอแนะและงานที่จะศึกษาต่อ

การนำแขนแนลของโมเดลสีทั้ง 3 แขนแนลมาวิเคราะห์ เพื่อตั้งค่าการเปลี่ยนแปลงสีที่เกิด จากการเต้นของหัวใจด้วย ICA อาจทำให้ความแม่นยำต่ำลง ซึ่งเกิดจากการรวมสัญญาณที่ไม่เกี่ยวข้องกับ ชีพจร โดยส่งผลกระทบต่อชัดเจนกับการผสมโมเดลสีแบบ CMU ดังนั้นการเลือกวิเคราะห์เฉพาะ แขนแนลที่มีองค์ประกอบของชีพจรจึงน่าจะส่งผลให้มีความแม่นยำที่สูงขึ้น

ศึกษาผลกระทบของการระบุค่าชีพจรในหน้าต่าที่ติดกัน เนื่องจากสัญญาณในหน้าต่าต่างๆ จะต่างจากสัญญาณในหน้าต่าก่อนหน้าเพียง 1 วินาทีหรือ 30 ค่าเท่านั้น (30 เฟรมต่อวินาที) แต่ใน บางกรณีผลลัพธ์การคำนวณค่าความถี่มีความแตกต่างกันมาก ดังแสดงดังรูปที่ 60 จากรูปเป็นค่าชีพ จรที่คำนวณได้ของผู้ทำการทดลองคนที่ 1 ใช้โมเดลสี HSV และขนาดหน้าต่า 40 วินาทีต่อหน้าต่า (1200 เฟรมต่อหน้าต่า) พบว่าในหน้าต่าที่ 12 ค่าความถี่ที่คำนวณได้ต่างจากหน้าต่าที่ 11 อยู่มาก ทำให้การระบุค่าชีพจรมีความผิดพลาดสูง



รูปที่ 60 กราฟชีพจรของผู้ทำการทดลองคนที่ 1 วัดด้วยโมเดลสี HSV ขนาดหน้าต่าง

40 วินาที/หน้าต่าง ใช้วิธี MCA และ RPS

เปรียบเทียบการวัดค่าความผิดพลาดในการระบุค่าชีพจรด้วยวิธีการอื่น เช่น Root Mean Square Error (RMS) และเปอร์เซ็นต์ความผิดพลาด เพื่อแสดงมุมมองการเปรียบเทียบผลกระทบของปัญหาให้มากขึ้น นำระบบการวัดชีพจรแบบไม่สัมผัสกับร่างกายไปใช้วัดชีพจรบริเวณใบหน้าในสภาพแวดล้อมจริงเพื่อเปรียบเทียบกับงานวิจัยอื่นให้ได้ผลลัพธ์ที่แม่นยำมากขึ้น และพัฒนางานต่อยอดทางการแพทย์เพื่อระบุปัญหาเกี่ยวกับการต่อเส้นเลือดกับเนื้อเยื่อ เนื่องจากการต่อเส้นเลือดที่สมบูรณ์จะทำให้เลือดไหลเวียนเข้าและออกจากเนื้อเยื่อ ดังนั้นสามารถนำหลักการที่เสนอไปวิเคราะห์การเปลี่ยนแปลงสีผิวของเนื้อเยื่อที่เกิดจากการไหลเวียนของเลือดได้

รายการอ้างอิง

- Ahmed Alzahrani and Anthony Whitehead. (2015). "Preprocessing Realistic Video for Contactless Heart Rate Monitoring Using Video Magnification." **Computer and Robot Vision (CRV)** 12 (June): 261-268.
- Ales Cerný. (2009). **Mathematical techniques in finance: Tools for incomplete markets**. 2nd ed. Princeton, N.J: Princeton University Press.
- Alex Beatson and Richard Green. (2014). "Calculating a Vagal Tone Index using Webcam Photoplethysmography." In **Proceedings of the 29th International Conference on Image and Vision Computing New Zealand (IVCNZ '14)** (November): 242-246.
- Dragos Datcu, Marina Cidota, Stephan Lukosch, and Leon Rothkrantz. (2013). "Noncontact automatic heart rate analysis in visible spectrum by specific face regions." In **Proceedings of the 14th International Conference on Computer Systems and Technologies** (June): 120-127.
- Fadel Adib, Zachary Kabelac, Hongzi Mao, Dina Katabi, and Robert C. Miller. (2014). "Demo: real-time breath monitoring using wireless signals." In **Proceedings of the 20th annual international conference on Mobile computing and networking**, 14: 261-262.
- Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John Guttag, Frédo Durand, and William Freeman. (2012). "Eulerian video magnification for revealing subtle changes in the world." **ACM Trans. Graph.** 31, 4 (July): 651-658.
- Ian Mccloughlin. (2009). **Applied Speech and Audio Processing with MATLAB Examples**. New York: Cambridge University Press.

- Jean-Francois Cardoso. (2015). **Blind separation of real signals using JADE, version 1.9**. Accessed July 20. Available from <http://perso.telecomparistech.fr/~cardoso/Algo/Jade/jadeR.m>
- Joel Murphy and Yury Gitman. (2015). **Pulse Sensor Amped Processing Visualizer**. Accessed November 20. Available from <http://pulsesensor.com>
- MicroLife. (2558). **เครื่องวัดความดัน Microlife 3AQ1**. เข้าถึงเมื่อ 20 พฤศจิกายน. เข้าถึงได้จาก <http://www.microlife.com/consumer-products/blood-pressure/upper-arm-automatic>
- Ming-Zher Poh, Daniel J. McDuff, and Rosalind W. Picard. (2010). "Non-contact, automated cardiac pulse measurements using video imaging and blind source separation." **Opt. Express** 18, 10 (May): 10762-10774.
- Ming-Zher Poh, Daniel J. McDuff, and Rosalind W. Picard. (2011). "Advancements in noncontact, multiparameter physiological measurements using a webcam." **IEEE Trans. Biomed. Eng.** 58, 1 (January): 7–11.
- Nadica Miljković and Dragan Trifunović. (2014) "Pulse rate assessment: Eulerian Video Magnification vs. electrocardiography recordings." **Neural Network Applications in Electrical Engineering (NEUREL)** 12 (November): 17-20.
- Pedro B. Chambino. (2014). "Android-based implementation of Eulerian Video Magnification for vital signs monitoring." Ph.D. dissertation, Faculdade de Engenharia, Universidade do Porto.
- Robert J. Schilling, Sandra L. Harris. (2012). **Fundamentals of digital signal processing using MATLAB**. 2nd ed. Toronto, Ont: Thomson.
- Xiaochuan He, Rafik A. Goubran, and Xiaoping P. Liu. (2014). "Using Eulerian video magnification framework to measure pulse transit time." **Medical Measurements and Applications (MeMeA), 2014 IEEE International Symposium on** (June): 1-4.



Using FFT Range-Power Spectrum to Improve Accuracy for Contact-Free Pulse Measurement

Nattapong Tangjui, Pinyo Taeprasartsit*
 Department of Computing, Faculty of Science
 Silpakom University
 Nakhon Pathom, Thailand
 * taeprasartsit_p@silpakom.edu

Abstract — Independent Component Analysis (ICA) on a color model can be applied to measure heart rate without physical contact with subjects. Unfortunately, current non-contact systems need accuracy improvements to make them suitable for some medical applications. While others employed the most dominant frequency calculated by FFT as a heart rate, we propose to accumulate power spectrums across frequencies and compute weighted average. This is highly effective when a subject's heart rate significantly changes during data acquisition. This work also systematically studied impacts of color models and found that straightforward combination of color models could lower performance. Yet, combining power spectrums from each color models marginally increases accuracy. Finally, impacts of time-window length were examined and the proposed method demonstrates improved robustness on most time-window lengths.

Keywords—independent component analysis; pulse measurement;

I. INTRODUCTION

Visual perception of a human has several limitations. For example, we cannot perceive subtle variations of skin color caused by blood perfusion or breathing motion of a baby. Poh et al. proposed to employ independent component analysis (ICA) to separate sources of subtle color changes caused by blood perfusion [1][2]. Then, estimated heart rate is computed by FFT on a specific ICA component. Wu et al. proposed a method for color amplification and motion magnification by applying Eulerian Video Magnification to reveal these subtleties [3]. Their work led to several innovations and intriguing research such as pulse-transit-time measurement [4]. These systems take a standard video sequences as an input, performs spatial decomposition and temporal filtering on video frames to reveal the subtle change in video sequence, such as changing of color caused by blood perfusion. Despite of its potential, accuracy of ICA-based systems needs improvements before they can be applied in some medical applications.

A. ICA-based Pulse Measurement

Subtle changes of skin color caused by blood perfusion appear to be periodic due to a heart-beat pattern. Common ICA-based pulse measurement detects this periodic pattern by utilizing Fast-Fourier Transform (FFT) to find dominant

frequency of the subtle changes. The change of skin color, however, may not be well represented in a single color channel (e.g. either red, green, or blue). Consequently, blind-source separation of the color components is executed to extract independent components underlying the color change. In this work, jadeR is employed for such Independent Component Analysis (ICA) [5]. Technically, we apply FFT on results from ICA as ICA components is significantly more corresponding to color change than a specific color component.

Some prior work interpreted the most dominant frequency from a specific ICA component as the heart rate, although we do not know in advance what component is best for finding the actual heart rate. Furthermore, if a heart rate has changed over the period of observation, the most dominant frequency may not close to the actual heart rate and a system produces highly inaccurate measurement.

B. Limitations in Prior Work and Accuracy Factors

Several existing systems using ICA-based pulse monitoring have technical limitations. For example, the method of Poh et al. did not address effects of skin color on accuracy [1]. Systems implemented on mobile phones or tablets cannot process data in a timely fashion; better hardware and software stack is required [6]. Also, a system testing on a moving subject reportedly had less accuracy and required a sophisticated method for video stabilization to reduce error [7].

Prior work employed different equipment and color models. For instance, Poh employed a typical webcam and RGB, while Miljkovic used a DSLR and RGBCO [1][8]. Yet, in prior study, it was unclear how much impact color models have on accuracy.

Furthermore, it is possible that a heart rate of a subject changes in a short period of time (< 30 seconds). While a longer time window of video sequences may have more data, such changes may prevent FFT from producing dominant frequency that well corresponds to the actual heart rate. Therefore, some systems employ a short time window (≤ 5 seconds) [6][8], but this may provide insufficient data for reliable measurement.

II. OUR GOAL AND OVERVIEW OF PROPOSED METHODS

Heart-rate variation over a short period of time is one of major challenges that we want to study. We aim at finding a method that can improve accuracy even when this challenge is present. We first evaluated the accuracy of a conventional method by employing an RGB color model as similarly done by Poh et al. [1].

Then, we experimented with various color models. Since color variation is the major data source for ICA-based pulse measurement, we explored how color models affected accuracy. We aimed at assessing impacts of color models on accuracy and finding whether changing or combining color models could improve accuracy. In this aspect, we propose to employ multiple models together and apply our FFT power-spectrum summation for accuracy enhancement.

Next, we propose a novel method that improve reliability of ICA-based pulse measurement by using relatively long time windows for video sequences along with our FFT power-spectrum combining mechanism. The main advantage of using a long time window is that spectrum of the most dominant frequency will be comparatively high and it is more likely to detect the actual heart rate, provided that the heart rate of a subject barely changes during observation. Yet, according to our experiments, the heart rate of a subject may significantly change in a short duration (< 30 seconds), thereby causing another challenge if a longer time window is to be utilized. This is a reason why FFT power-spectrum combining mechanism is introduced.

When the length of observation increases, it is more likely that multiple heart rates will occur and none of them may possess strong power spectrum. However, these involving heart rates are usually among the top when decomposed FFT frequencies are ranked by their power spectrums. Since the ICA component having most relevant information on heart rate is not known in advance, the proposed method relies on accumulated power spectrums of each frequency across ICA components.

Also, because there may be multiple involving heart rates in a single time window, we add up power spectrums of frequencies within a small range. For example, assume that we want to find the combined power spectrums for the frequency corresponding to 70 beats per minutes (bpm), we sum up power spectrums of 68 – 72 bpm, including. For the sake of clarity, we will refer to the accumulation of power spectrums across ICA components as *multi-component accumulation* (MCA) and refer to the sum of power spectrums within a specified bpm range as a *range power spectrum* (RPS).

Nonetheless, noise within a certain time window may be so strong that power spectrums of actual heart rates become weak by comparison and the output heart rate is inaccurate. The method suppresses such an error by using historical estimation and rejects a suspicious output that is highly different from those of consecutive windows [1].

III. METHODS

A. Data Acquisition

Since we focus on impacts of color models, time-window length, and accuracy improvement utilizing proposed multiple color models and range power spectrum, we acquired data by employing a light-control box to limit illumination variation. The camera device was iPad 3 and the frame rate was 30 frame per second (fps). Five subjects (three females and two males) between the age of 10 to 52 years old participated in this study. All subjects were not under stress. Videos focused on palm skin as shown in Figure 1. Each video is approximately 60 seconds in length and was recorded in RGB.



Figure 1 Settings for video acquisition. Left: light-control box and a tripod for holding a tablet. Right: a sample video frame focusing on a palm of a subject.

B. Color-Feature Preparation

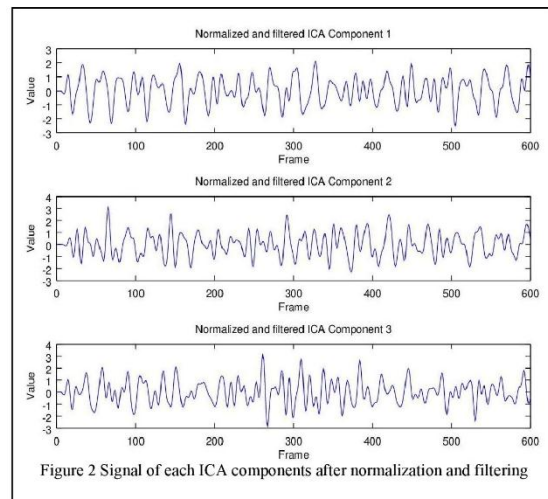


Figure 2 Signal of each ICA components after normalization and filtering

Color features are prepared as follows.

- A manually selected skin region was segmented from each video frame. This region approximately had 20,000 pixels.
- Find an average value of each color component (R, G, and B) in the selected region. Each frame corresponds to 3 values, averages of R, G, and B.
- Arrange the 3 values in order of time. In this way, a 3-dimensional signal representing average color

components is obtained. In our settings, values from 600 frames (20 seconds of video) are arranged together.

- Employ jadeR on the signal to obtain independent components underlying the color change. It is worth noting that the number of independent components may be less than 3 in some cases. (Figure 2)

C. MCA and RPS

Once a signal of independent components is obtained, an n -point FFT is executed over the entire signal in a time window with $n = f \cdot t$ where f is the number of frames per second and t is the duration of a window. We cut off the frequencies outside range $0.75 - 4$ Hz (corresponding to heart rate of 45 to 240 bpm). The proposed method then applies MCA to add up the FFT power spectrums of each frequency from all independent components. In other words, if there are 3 independent components, the method adds up power spectrums of the same frequency from the 3 components together.

Now, we provide details on how the proposed method improve accuracy over previous work. Poh's method in 2010 employed Component 2 from ICA to compute heart rate [1]. However, Table 1 shows that Component 2 may not be a reliable source for measurement. This table shows the first five dominant frequencies and their corresponding heart rate of each ICA components. We observe that the top frequency (the one with the maximum magnitude) of each component corresponds to different heart rates. We compared these heart rates with the reference (76.45 bpm) measured by the Arduino-based Pulse Sensor. Only are results from the first and third components close to reference. This makes Poh's work inaccurate [1].

Table 1 First five dominant frequencies of each ICA component (reference heart rate is 76.45 bpm).

Component 1		Component 2		Component 3	
Magnitude	Pulse (bpm)	Magnitude	Pulse (bpm)	Magnitude	Pulse (bpm)
182.87	78	138.84	63	120.63	75
130.18	63	113.43	84	108.86	120
128.13	75	96.82	135	96.39	78
109.74	150	88.31	153	85.08	57
101.87	123	86.60	69	78.45	144

Nonetheless using the heart rate from the component with highest magnitude among three ICA components (Component 1 in the above table) seem to yield a good result, it may produces inaccurate output. As shown in Table 2, the obtained heart rate is 69 bpm, while the reference one is 78. This makes Poh's 2011 method highly imprecise [2].

Our proposed MCA method, however, accumulates magnitude of frequencies across components and picks the one with highest accumulated magnitude and tends to produce more accurate result. In this example, our method outputs 87 bpm.

This result is obtained because the sum of magnitudes corresponding to 87 bpm from all components is the largest when compared to the sums of other frequencies. For the same example, the magnitude sum of 87 bpm is $168.5 + 95.43 + 121.57 = 385.50$.

Table 2 First five dominant frequencies of each ICA component (reference heart rate is 87 bpm).

Component 1		Component 2		Component 3	
Magnitude	Pulse (bpm)	Magnitude	Pulse (bpm)	Magnitude	Pulse (bpm)
168.50	87	110.92	84	172.83	69
110.49	111	107.76	72	121.57	87
93.48	123	95.42	87	98.17	57
88.46	75	91.75	93	81.40	54
82.86	84	85.36	81	80.45	129

Next, to further improve accuracy, range power spectrums are calculated. For each heart rate b within range 45 to 240 bpm, the method sums up power spectrums of $b - 2$ to $b + 2$ bpm, including, to combine power spectrums within a small range. As for the next step, our method selects the frequency with the highest range power spectrum as the most likely heart rate.

Table 3 shows the frequencies and their magnitudes calculated by RPS from data in Table 1. For instance, the RPS magnitude of 77 bpm is the sum of magnitude corresponding to heart rates of 75 - 79 bpm, including, which is $128.13 + 120.63 + 182.87 + 96.39 = 528.02$. Similarly, RPS magnitude of 76 bpm is summed by magnitudes corresponding to heart rates of 74 - 78 bpm, which equals to 77 bpm in this case. The frequencies of highest RPS magnitude are 77 and 76 bpm and the output heart rate is 76.50 bpm, which is calculated by averaging them. This provides a result particularly close to the reference heart rate.

Table 3 RPS frequencies and magnitudes of ICA components from Table 1.

Rank	Magnitude	Pulse (bpm)	Rank	Magnitude	Pulse (bpm)
1	528.02	77	11	248.76	74
2	528.02	76	12	248.76	73
3	279.26	80	13	248.76	75
4	279.26	79	14	210.74	122
5	279.26	78	15	210.74	121
6	269.02	65	16	198.04	152
7	269.02	64	17	198.04	151
8	269.02	61	18	113.43	82
9	269.02	62	19	113.43	84
10	269.02	63	20	113.43	85

D. Error Suppression (ES)

A detected heart rate from a current time window is checked against the result from the previous. If the difference is less than 3 bpm, the method rejects the result from the current time window and use the previous result. However, if the result from the next time window is close to the rejected one, the method switches to the new one since it is more likely that the rejected one is, in fact, correct [1].

E. Employing Additional Color Models

We explored possibility of using multiple color models to improve accuracy of the method. Our study on color models was carried out as follows.

- From RGB color model, convert it to others: HSV, HLS, CIE Lab, CIE Luv, TSL (Tint Saturation Lightness), and YCbCr.
- For each model, compute a heart rate by using a similar method for RGB. Technically, we obtained 6 more output heart rates for comparative study.

The proposed method further combines all power spectrums of each color model to obtained from the above process and selects the frequency with maximum range power spectrum to be the output heart rate. For the sake of clarity, we will refer to this use of multiple color models as *multi-model summation (MMS)*.

For the last part of this color-model study, all color components from the seven color models are grouped together (duplicated components were removed). Then, ICA was executed to the whole group. Again, for clarity, this method will be referred to as *color-model union (CMU)*. The difference of the last two methods is that MMS performs ICA seven times, each on a color model, while the CMU groups all color components together and performs ICA only once.

F. Finding Proper Length of Time Window

Using short or long time window is a trade off in terms of computational speed, how quickly a method can respond to change of actual heart rate, and how much data available for frequency detection. Since prior work seems to focus on computational speed, to the best of our knowledge, we did not find any prior work study the impact of time-window length on the accuracy.

In this work, we experimented with various time-window lengths ranging from 5 to 60 seconds. Each consecutive time windows overlap by 1 second. For the whole experiments, the heart rate for combining power spectrums was set to ± 2 bpm as mentioned in Section C.

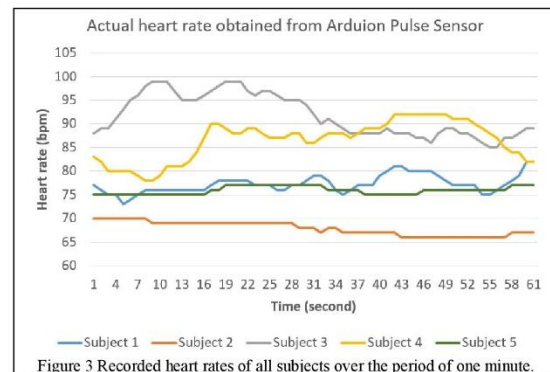
G. Performance Validation

We validated the accuracy of our methods by comparing our outputs with those from an Arduino-based Pulse Sensor [9]. Time stamps of each sampling were recorded and matched with video-recording starting time. The outputs corresponding to a time window were compared with the average sampled heart rate of the same time window. It is important to note that such an average heart rate may not be a good representation of actual heart rates if actual heart rates considerably varies during observation. Nonetheless, average heart rate within a time window is still a reasonable choice for this study.

IV. EXPERIMENTS AND RESULTS

A. Actual Heart Rates

Arduino-based Pulse Sensor was employed to record heart rate over a recording time as shown in Figure 3. It is worth noting that heart rates of two subjects significantly varied within 60 seconds. The minimum and maximum heart rates of these subjects differed up to 14 bpm. This suggests that power spectrums belonging to a specific actual heart rate may not be adequately high when compared to noise spectrums. Therefore, combining power spectrums across heart rates may improve accuracy as shown in Section C.



B. Impacts of MCA on Accuracy

We compared the error rates from our MCA method when RGB color model was employed against results from Poh's methods [1, 2]. Although our method can work with any color model, the color model in this MCA experiment was RGB. This allows direct comparison with Poh's methods.

The results shown in Table 4 indicated that the MCA outperformed Poh's 2010 method, but was less accurate than Poh's 2011 one. MCA, however, is proposed here since it performs relatively well and provides useful data for our RPS method, which is more accurate than Poh's 2011 work as discussed in the next section.

Table 4 Comparison of errors between MCA, Poh's 2010, and Poh's 2011. The results were averaged from all subjects and time-window lengths. All errors are shown in bpm.

Methods	Without ES			With ES		
	Avg	StdDev	Max	Avg	StdDev	Max
MCA	6.35	4.31	21.47	5.78	2.61	10.62
Poh's 2010 [1]	12.89	12.38	48.62	12.77	10.91	35.41
Poh's 2011 [2]	5.12	2.96	14.73	5.02	2.13	9.74

C. Impacts of Using Range Power Spectrum

In this experiment, we compared errors from using and not using range power spectrum (RPS) and we focused only on output heart rates obtained with error suppression. Table 5 displays comparative results between our method and Poh's. In this setting, our method employed both MCA and RPS together. It is worth noting that Poh's 2010 method can gain benefits from using RPS if other frequencies in Component 2 are included in analysis. The results showed that using RPS improved accuracy for both our MCA and Poh's 2010 methods. Furthermore, our proposed method outperformed Poh's 2011 work.

To stress the benefits of using RPS, we also compared similar results obtained from other color models. Results displayed in Table 6 were calculated from all subjects when a 20-second time window was applied (similar to experiment setting of Poh's 2010). It indicates that utilizing range power spectrum tended to increase average accuracy, regardless of color models. Also, the standard deviation and maximum of errors tended to decrease.

Table 5 Comparison of errors between MCA & ES, Poh's 2010, and Poh's 2011. The results were averaged from all subjects and time-window lengths.

Methods	Without RPS			With RPS		
	Avg	StdDev	Max	Avg	StdDev	Max
MCA & ES	5.78	2.61	10.62	4.43	1.92	8.48
Poh's 2010 [1]	12.77	10.91	35.41	12.34	9.85	32.52
Poh's 2011 [2]	5.02	2.13	9.74			

Table 6 Comparison of errors from employing range power spectrum with MCA. Numbers in Column "Improvements if Using RPS" were computed by errors from Table 7, Column "With Error Suppression" subtracted by corresponding values in Column "Not Using RPS" in this table. Positive numbers after subtraction indicates increase in accuracy.

Color Model	Not Using RPS			Using RPS			Improvements if Using RPS		
	Avg	SD	Max	Avg	SD	Max	Avg	SD	Max
RGB	4.8	3.2	13.8	4.0	2.1	8.0	0.7	1.1	5.7
HLS	5.7	3.0	11.4	4.6	3.4	14.2	1.1	-0.4	-2.8
HSV	5.5	2.8	9.5	5.0	3.6	14.3	0.6	-0.9	-4.7
CIE Lab	4.8	3.2	13.8	4.0	2.1	8.0	0.8	1.2	5.7
CIE Luv	4.7	2.9	12.0	4.0	2.1	8.0	0.7	0.8	3.9
TSL	8.2	8.4	32.7	7.6	6.8	21.9	0.5	1.7	10.8
YCbCr	4.7	3.2	13.8	4.0	2.1	8.0	0.7	1.1	5.7

D. Impacts of Color Models on Accuracy

To study impacts of color models, we fixed the length of time window to 20 seconds (600 frames) and employed both multi-model combination and RPS. The processing was performed over the 60 seconds of videos. Therefore, there were 41 time windows and output heart rates for a subject. Errors were calculated by the absolute difference between the output and heart rate from Arduino Pulse Sensor in bpm unit. Then, average, standard deviation and maximum of errors from a subject were calculated. These statistics from all subjects were then averaged and reported in Table 7.

From the table, it is clear that using multi-model summation (MMS) barely improved the accuracy of the measurement. Color-model union (CMU), however, led to lower accuracy. It is possible that independent components obtained in CMU mostly contained no useful information about color changes caused by blood perfusion. Consequently, the proposed MCA, which adds up all components together, could not increase the chance that the actual heart rate would have highest power spectrums, as power spectrums of noise frequency in many components might add up and appear stronger than those of actual heart rates.

MMS, on the other hand, computes independent components of each color model separately. Therefore, there may be one or more independent components of each color model that are good for heart-rate measurement. When all independent components were employed, the summed power spectrum of actual heart rates still stood out and the results remained accurate. In this experiment, MMS was the best, but just marginally outperformed RGB color model.

Figure 4 illustrates how measurement errors of each subject distributed without error suppression for MMS. We disabled error suppression here to show how results from each individual time window were. It seems that maximum error from a certain window could be high (27 bmp), but since the probabilities of such high errors to occur was low, error suppression could reject them.

Table 7 Comparison of average errors from employing different color models. These results were averaged from all subjects. All errors are shown in bpm.

Color Model	Without Error Suppression			With Error Suppression		
	Avg	StdDev	Max	Avg	StdDev	Max
RGB	4.0	2.1	7.8	4.0	2.1	8.0
HLS	5.5	5.2	23.2	4.6	3.4	14.2
HSV	5.6	5.2	23.2	5.0	3.6	14.3
CIE Lab	4.0	2.2	9.5	4.0	2.1	8.0
CIE Luv	4.1	2.5	12.2	4.0	2.1	8.0
TSL	8.2	7.6	31.9	7.6	6.8	21.9
YCbCr	4.0	2.1	7.8	4.0	2.1	8.0
MMS	4.0	2.5	14.4	3.9	2.4	9.7
CMU	6.8	5.0	21.0	6.5	4.2	16.0

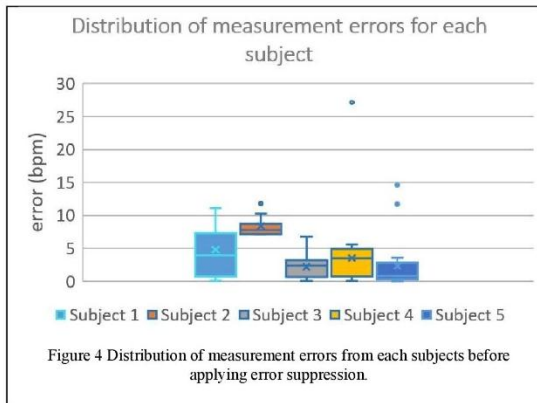


Figure 4 Distribution of measurement errors from each subjects before applying error suppression.

E. Impacts of Time-Window Length

For the last part of analysis, errors obtained from different time-window lengths were compared. The lengths we studied varied from 5 to 60 seconds. The results are calculated by utilizing MMS, MCA, RPS, and ES altogether (Figure 5). The outcomes indicated that longer time window tended to give better accuracy. However, increasing the length too much can reduce performance as the heart rates within the duration may vary in greater extent. A time window of length 45 seconds gave the most accurate outputs both our method and Poh's 2011 with errors of only 2.53 and 3.16 bpm on average. The proposed method outperformed Poh's in most cases.

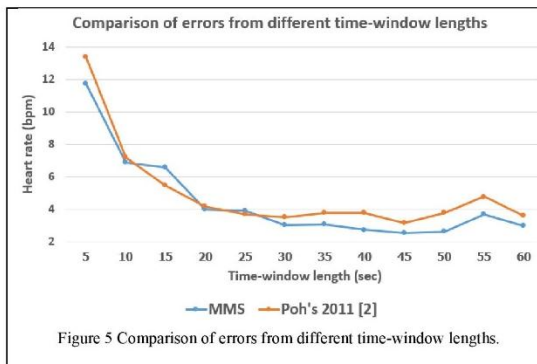


Figure 5 Comparison of errors from different time-window lengths.

F. Comparison of Proposed Method

In this section, we summarize the results of proposed methods in order to discuss their impacts on accuracy. These methods include using MCA exclusively, MCA & ES, MCA & RPS, and MCA & RPS & ES. The results in Table 8 were averaged from all subjects and all time-window lengths. These indicated that utilizing all proposed methods together (MCA & RPS & ES) had best accuracy in terms of both average and maximum errors.

Focusing on impact of ES, we found that using ES reduced average and maximum errors. By applying ES with MCA average and maximum errors were reduced by 1.03 bpm and 12.61 bpm, while applying ES with MCA & RPS decreased errors by 1.19 bpm and 12.81 bpm. Exploring how RPS affected accuracy, we observed that employing RPS with MCA reduced errors by 1.02 and 1.99 bpm. Similarly, applying RPS with MCA & ES decreased errors by 1.18 and 2.19 bpm.

Table 8 Comparison of proposed method from all subject and all errors are display in bpm.

Method	Avg errors	Avg of max errors
MCA	10.403	35.006
MCA & ES	9.376	22.394
MCA & RPS	9.384	33.014
MCA & RPS & ES	8.199	20.202

G. Comparison with Other Prior Work

In this part, we summarized the performance of our methods with Poh's. We set the time-window lengths to 20 and 30 seconds and employed RGB as a color model (similar to Poh's settings).

Figure 6 visualizes output heart rates from various methods overtime, including reference Arduino-based Pulse Sensor. We found that output heart rates of 2nd component of ICA was not reliable [1], unlike the others that were much less varied and generally accurate. For this time-window length, our proposed method produced similar outputs to Poh's [2]. Table 9 compares the proposed method against Poh's 2010 [1], and Poh's 2011 [2]. We can see that using only 2nd component of ICA [1] is more susceptible to errors and our method had lowest average errors, even in their preferred time-window settings.

Table 9 Comparison of proposed method against prior work with the RGB color model. All errors are shown in bpm.

Methods	Window size					
	20 seconds			30 seconds		
	Avg	SD	Max	Avg	SD	Max
MCA & RPS & ES	4.01	2.11	8.05	3.01	1.36	5.71
Poh [1]	12.08	13.57	38.15	5.58	5.89	23.46
Poh [2]	4.20	1.68	7.34	3.51	1.48	5.77

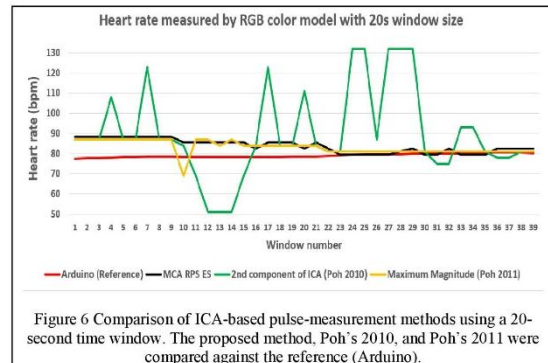


Figure 6 Comparison of ICA-based pulse-measurement methods using a 20-second time window. The proposed method, Poh's 2010, and Poh's 2011 were compared against the reference (Arduino).

V. CONCLUSIONS AND FUTURE WORK

We proposed multi-component accumulation and range power spectrum to gather more useful statistics from FFT and deal with a situation where a subject's heart rate changes during measurement. Our study showed that the proposed method was effective in accuracy improvements, regardless of color model. Excessively combining color models, however, could be detrimental to performance.

We are going to perform tests on a larger number of subjects to further validate impacts of the proposed methods on accuracy.

REFERENCES

- [1] M.-Z. Poh, D. J. McDuff, and R. W. Picard, "Non-contact, automated cardiac pulse measurements using video imaging and blind source separation," *Opt. Express* 18, 10762-10774 (2010).
- [2] M.-Z. Poh, D. J. McDuff, and R. W. Picard, "Advancements in noncontact, multiparameter physiological measurements using a webcam," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 1, pp. 7-11, Jan. 2011.
- [3] H.-Y. Wu, M. Rubinstein, E. Shih, J. Gutttag, F. Durand, and W. T. Freeman, "Eulerian video magnification for revealing subtle changes in the world," *ACM Trans. Graph.* (Proceedings SIGGRAPH 2012), vol. 31 (4), pp. 651-658, July 2012.
- [4] X. He, R. A. Goubran and X. P. Liu, "Using Eulerian video magnification framework to measure pulse transit time," *Med. Measurements and Appl. (MeMeA), 2014 IEEE Int. Sym. on, Lisboa, 2014*, pp. 1-4.
- [5] Jean-Francois Cardoso. (2016). *Blind separation of real signals using JADE*, version 1.9, Aug. 2013 [Online]. Available: <http://perso.telecom-paristech.fr/~cardoso/Algo/Jade/jadeR.m>.
- [6] P. B. Chambino, "Android-based implementation of Eulerian Video Magnification for vital signs monitoring," Ph.D. dissertation, Faculdade de Engenharia da Universidade do Porto, 2013, Available on Oct. 2014.
- [7] A. Alzahrani and A. Whitehead, "Preprocessing Realistic Video for Contactless Heart Rate Monitoring Using Video Magnification," *Computer and Robot Vision (CRV), 2015 12th Conference on, Halifax, NS, 2015*, pp. 261-268.
- [8] N. Miljković and D. Trifunović, "Pulse Rate Assessment Eulerian Video Magnification vs. Electrocardiography Recordings. Neural Network Applications in Electrical Engineering (NEUREL), 2014 12th Sym. on, pp. 17-20, Jan. 2015.
- [9] Joel Murphy and Yury Gitman. (2016). *Pulse Sensor Amped Processing Visualizer*. Available: <http://pulsesensor.com>.



Certificate
Oral Presentation

The 13th International Joint Conference on Computer Science
 and Software Engineering
 July 13-15, 2016

Department of Computer Science
 Faculty of Science, Khon Kaen University, Thailand

Awards This Certificate To :

**Using FFT Range-Power Spectrum to Improve Accuracy
 for Contact-Free Pulse Measurement**

Nattapong Tangjui and Pinyo Taeprasartsit

P. Pipat Reungsang

Asst.Prof. Pipat Reungsang
 General Chair

Prof. Kosin Chamnongthai

Prof. Kosin Chamnongthai
 Technical Program Chair

ประวัติผู้วิจัย

ชื่อ – สกุล

ณัฐพงศ์ แต่งจ้อย

ที่อยู่

14/4 ซ.โรจนฤทธิ1 ต.พระปฐมเจดีย์ อ.เมือง จ.นครปฐม 73000

ประวัติการศึกษา

พ.ศ. 2557

สำเร็จการศึกษาปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์และระบบคอมพิวเตอร์
มหาวิทยาลัยศิลปากร
อาจารย์ที่ปรึกษาวิทยานิพนธ์ ผศ.ดร.ชูเกียรติ สอดศรี
GPA : 3.55

พ.ศ. 2557

ศึกษาต่อระดับปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
มหาวิทยาลัยศิลปากร
อาจารย์ที่ปรึกษาวิทยานิพนธ์ ดร.ภิญโญ แท้ประสาทสิทธิ์
GPA : 3.90

