

ระบบรู้จำท่ามือบนพื้นฐานการรับรู้ภาพสำหรับผู้สูงอายุ



นางสาวจิตาภรณ์ กนกรัตน

จุฬาลงกรณ์มหาวิทยาลัย

CHULALONGKORN UNIVERSITY

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)  
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ ที่ส่งผ่านทางบัณฑิตวิทยาลัย

The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository (CUIR)  
are the thesis authors' files submitted through the University Graduate School.

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2559

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย



Vision-based Hand Gesture Recognition System for Elderly People

Miss Thittaporn Ganokratanaa



A Thesis Submitted in Partial Fulfillment of the Requirements  
for the Degree of Master of Engineering Program in Electrical Engineering

Department of Electrical Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2016

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	ระบบรู้จำท่ามือบนพื้นฐานการรับรู้ภาพสำหรับผู้สูงอายุ
โดย	นางสาวฐิตาภรณ์ กนกรัตน์
สาขาวิชา	วิศวกรรมไฟฟ้า
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร. สุรีย์ พุ่มรินทร์

---

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้บัณฑิตวิทยานิพนธ์ฉบับนี้เป็นส่วน  
หนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์  
(รองศาสตราจารย์ ดร. สุพจน์ เตชวรสินสกุล)

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ  
(รองศาสตราจารย์ ดร. สุภาวดี อร่ามวิทย์)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก  
(ผู้ช่วยศาสตราจารย์ ดร. สุรีย์ พุ่มรินทร์)

.....กรรมการภายนอกมหาวิทยาลัย  
(รองศาสตราจารย์ ดร. เอกชัย ลีลารัมย์)

ฐิตาภรณ์ กนกรัตน : ระบบรู้จำท่ามือบนพื้นฐานการรับรู้ภาพสำหรับผู้สูงอายุ (Vision-based Hand Gesture Recognition System for Elderly People) อ.ที่ปรึกษา  
วิทยานิพนธ์หลัก: ผศ. ดร. สุรีย์ พุ่มรินทร์, 127 หน้า.

ในปัจจุบันเทคโนโลยีจัดเป็นส่วนหนึ่งในชีวิตของมนุษย์ที่มีความสำคัญจนเกิดเป็นศาสตร์การปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์ (Human computer interaction: HCI) ซึ่งศาสตร์นี้สามารถนำมาประยุกต์ใช้ได้กับเทคโนโลยีอำนวยความสะดวกและระบบทางการแพทย์ การแสดงท่ามือจัดเป็นวิธีที่เกิดจากสัญชาตญาณของมนุษย์จึงทำให้ง่ายต่อการสื่อสารต่อคอมพิวเตอร์ การใช้ท่ามือจึงเป็นวิธีที่เหมาะสมสำหรับการติดต่อสื่อสารผ่านทางระบบเฝ้าระวังของผู้สูงอายุที่ไม่สามารถเดินหรือแสดงความต้องการผ่านทางกรพูดกับผู้ดูแล ดังนั้นงานวิจัยนี้นำเสนอระบบรู้จำท่ามือสำหรับผู้สูงอายุแบบทันทีโดยใช้บอร์ด Raspberry Pi ระบบนี้พัฒนาโดยใช้วิธีการรับรู้ภาพสำหรับการตรวจจับและจำแนกท่ามือแบบเคลื่อนไหว โดยแบ่งออกเป็นขั้นตอนหลัก ๆ 3 ขั้นตอน ได้แก่ การหาเส้นขอบ การหาคอนเวกซ์ฮัล และการใช้กฎการตัดสินใจ จากผลการทดลองพบว่าระบบสามารถรู้จำท่ามือ 9 ท่าได้ดี แม้ว่าจะมีการเคลื่อนที่ท่ามือในองศา ระยะห่าง และมุมมองที่แตกต่างกัน โดยค่าความแม่นยำในการจำแนกความหมายและการส่งข้อความเพื่อแจ้งเตือนผู้ดูแลมีค่าเท่ากับ 99.48 เปอร์เซ็นต์ และ 100 เปอร์เซ็นต์ตามลำดับ

จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

ภาควิชา วิศวกรรมไฟฟ้า

ลายมือชื่อนิสิต .....

สาขาวิชา วิศวกรรมไฟฟ้า

ลายมือชื่อ อ.ที่ปรึกษาหลัก .....

ปีการศึกษา 2559

# # 5870138621 : MAJOR ELECTRICAL ENGINEERING

KEYWORDS: HUMAN - COMPUTER INTERACTION / VISION-BASED METHOD / HAND GESTURE RECOGNITION / ELDERLY CAREGIVER / RASPBERRY PI / MONITORING SYSTEM

THITTAPORN GANOKRATANAA: Vision-based Hand Gesture Recognition System for Elderly People. ADVISOR: ASST. PROF. SUREE PUMRIN, Ph.D., 127 pp.

Technology becomes a part of human lives for decades, especially in human - computer interaction (HCI) that considered as the important research area involving with an assistive technology and a medical system. Hand gesture is classified as an intuitive method for human to convey information and interact with the computer. It is useful for elderly people who cannot walk or express their feelings by words to contact with the caregiver via the monitoring system. This research proposes a real time hand gesture recognition system for elderly people using an inexpensive Raspberry Pi. A vision-based method is developed to detect and classify dynamic hand gestures. There are three main procedures; contour detection, convex extraction, and rule-based classification. The system can recognize nine different gestures in various orientations, angles, distances, and occlusions. The experiment showed good results in classifying their meanings as lingual descriptions and sending message to the care giver's smartphone at 99.48 percent and 100 percent, respectively.

Department: Electrical Engineering      Student's Signature .....

Field of Study: Electrical Engineering      Advisor's Signature .....

Academic Year: 2016

## กิตติกรรมประกาศ

วิทยานิพนธ์เรื่องระบบรู้จำท่ามือบนพื้นฐานการรับรู้ภาพสำหรับผู้สูงอายุสำเร็จลุล่วงไปได้ด้วยดีด้วยความช่วยเหลือและคำแนะนำจากอาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร. สุรีย์ พุ่มรินทร์ ที่คอยให้คำปรึกษาและสนับสนุนมาโดยตลอดทั้งในการทำวิทยานิพนธ์ การเรียนและงานวิจัย

ขอขอบคุณคณะกรรมการสอบวิทยานิพนธ์ รองศาสตราจารย์ ดร. สุภาวดี อร่ามวิทย์ ประธานกรรมการสอบวิทยานิพนธ์ และรองศาสตราจารย์ ดร. เอกชัย ลีลาวัศมี ผู้ทรงคุณวุฒิภายนอกมหาวิทยาลัย สำหรับคำแนะนำในการปรับปรุงผลงานและการทำวิทยานิพนธ์

ขอขอบคุณจุฬาลงกรณ์มหาวิทยาลัยที่สนับสนุนทุนการศึกษาสำหรับการเรียนและการทำงานวิจัย โดยวิทยานิพนธ์นี้ได้รับการสนับสนุนจากห้องปฏิบัติการวิจัยระบบสมองกลฝังตัวและการออกแบบวงจรรวม (Embedded System and IC Design research laboratory) ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ โครงการสร้างเสริมพลังจухาฯ ก้าวสู่ศตวรรษที่ 2 (Chulalongkorn Academic Advancement into Its 2nd Century Project) ทุนอุดหนุนการศึกษาระดับบัณฑิตศึกษาจุฬาลงกรณ์มหาวิทยาลัย เพื่อเฉลิมฉลองวโรกาสที่พระบาทสมเด็จพระเจ้าอยู่หัวทรงเจริญพระชนมายุครบ 72 พรรษา และทุน 90 ปีจุฬาลงกรณ์มหาวิทยาลัย กองทุนรัชดาภิเษกสมโภช

สุดท้ายนี้ ขอขอบพระคุณมารดา ผู้ที่สำคัญที่สุดในชีวิต ขอขอบคุณครอบครัวและครูอาจารย์ที่ให้กำลังใจ แนะนำ ส่งเสริม เป็นแรงใจที่ทำให้มีทุกวันนี้ได้ รวมถึงเพื่อน ๆ พี่ ๆ ที่อยู่เคียงข้างกันเสมอมา

## สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	1
สารบัญรูปภาพ.....	2
บทที่ 1 บทนำ.....	6
1.1 ความเป็นมา.....	6
1.2 ทบทวนวรรณกรรม.....	7
1.3 วัตถุประสงค์ของงานวิจัย.....	10
1.4 ขอบเขต.....	10
1.5 ประโยชน์.....	10
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง.....	11
2.1 ความรู้ด้านการประมวลผลภาพ (Image processing).....	11
2.2.1 หลักการประมวลผลภาพ (Image processing).....	11
2.2.2 โมเดลสี RGB.....	12
2.2.3 โมเดลสี YUV.....	13
2.2.4 ภาพระดับเทา (Grayscale image).....	15
2.2.5 ภาพไบนารี (Binary image).....	16
2.2 การประมวลผลเบื้องต้น (Pre-processing).....	16
2.2.1 หลักการรับภาพ (Image acquisition).....	16
2.2.2 หลักการปรับแต่งภาพ (Image enhancement).....	17



2.2.3	หลักการการปรับภาพเรียบ .....	18
2.2.4	การเปลี่ยนแปลงลักษณะรูปร่างหรือโครงสร้างของภาพ (Morphology operations).....	19
2.2.5	การลบใบหน้า (Face subtraction) .....	21
2.3	การตรวจจับมือ (Hand detection) .....	22
2.3.1	การลบพื้นหลัง (Background subtraction).....	22
2.3.2	การตรวจจับสีผิว (Skin color detection).....	22
2.3.3	การตรวจจับเส้นขอบ (Contour detection) .....	23
2.4	การติดตามมือ (Hand tracking).....	24
2.4.1	การติดตามวัตถุ (Multi-Object tracking).....	24
2.4.1.1	การไหลของแสง (Optical flow).....	24
2.4.2	การติดตามวัตถุที่สนใจ (Interested object tracking).....	25
2.4.2.1	การวิเคราะห์blob (Blob analysis).....	25
2.5	การวิเคราะห์ท่ามือ (Hand gesture recognition) โดยการตัดสินใจท่ามือ (Hand gesture decision) .....	25
2.5.1	การสกัดคอนเวกซ์ (Convex extraction) .....	25
2.5.1.1	คอนเวกซ์ฮัลล์ (Convex hull) .....	25
2.5.1.2	จุดลึกสุดในรอยเว้า (Convexity defect).....	26
2.5.2	ระยะห่างระหว่างพิกเซลสองพิกเซล (Euclidean distance).....	27
2.5.3	การเรียนรู้ต้นไม้ตัดสินใจ (Decision tree learning) .....	28
2.5.4	SURF (Speeded-up robust features).....	28
2.5.5	หลักการในการตัดสินใจท่ามือ (Hand gesture decision) .....	29
2.6	บอร์ด Raspberry Pi และการเชื่อมต่อระหว่างบอร์ดและสมาร์ทโฟน (Raspberry Pi board and its connection with smart phone).....	30

2.6.1	บอร์ด Raspberry Pi 3 Model B และโมดูลกล้อง V2 .....	30
2.6.2	การเชื่อมต่อระหว่างบอร์ดและสมาร์ทโฟน .....	31
บทที่ 3	วิธีดำเนินการทดลอง .....	32
3.1	ฮาร์ดแวร์ระบบรู้จำท่าทางมือแบบเคลื่อนไหว .....	32
3.1.1	โมดูลกล้อง V2 .....	32
3.1.2	บอร์ด Raspberry PI 3 model B .....	33
3.2	ขั้นตอนการทำงานของระบบรู้จำท่าทางมือแบบเคลื่อนไหว .....	33
3.2.1	อัลกอริธึมรู้จำท่าทางมือแบบสถิตสำหรับผู้สูงอายุโดยใช้กฎการจำแนก .....	34
3.2.1.1	การนำเข้าภาพนิ่ง .....	35
3.2.1.2	การกำหนดบริเวณที่สนใจ .....	35
3.2.1.3	การแปลงภาพสีเป็นภาพระดับเทา .....	35
3.2.1.4	การลบพื้นหลัง .....	36
3.2.1.5	การตรวจจับเส้นขอบ .....	36
3.2.1.6	การทำโมเมนต์ภาพ (Image Moments) .....	38
	จุดศูนย์กลางมวล (Centre of Mass) .....	38
	ความแปรปรวน (Variance) .....	39
	การกำหนดทิศทาง (Orientation) .....	39
3.2.1.7	การสกัดคอนเวกซ์ .....	40
3.2.1.8	การใช้กฎการจำแนก .....	41
3.2.1.9	แสดงผลลัพธ์ .....	42
3.2.2	อัลกอริธึมรู้จำท่าทางมือสำหรับผู้สูงอายุโดยใช้ Optical Flow และ SURF .....	44
3.2.2.1	การนำวิดีโอเข้าระบบ .....	44
3.2.2.2	การจับเฟรมภาพ .....	45

3.2.2.3 การแปลงสีเป็นภาพระดับเทา.....	46
3.2.2.4 การไหลของแสง .....	46
3.2.2.5 การคำนวณค่าเฉลี่ย.....	47
3.2.2.6 ตัวกรองค่ามัธยฐาน .....	47
3.2.2.7 การปรับปรุงโครงสร้างภาพ .....	48
3.2.2.8 การวิเคราะห์หีบห่อ .....	49
3.2.2.9 วิธีการ SURF .....	50
3.2.2.10 แสดงผลลัพธ์.....	52
3.2.3 อัลกอริธึมที่นำเสนอสำหรับระบบรู้จำท่ามือบนพื้นฐานการมองเห็นสำหรับผู้สูงอายุ..	53
3.2.3.1 การรับภาพเคลื่อนไหว.....	54
3.2.3.2 การตรวจจับใบหน้า.....	54
3.2.3.3 ปรับความเรียบ.....	55
3.2.3.4 การลบพื้นหลัง .....	57
3.2.3.5 การทำเทรซโฮลต์ .....	58
3.2.3.6 การเปลี่ยนแปลงลักษณะรูปร่างหรือโครงสร้างของภาพ.....	60
3.2.3.7 ปรับความเรียบภาพไบนารี.....	61
3.2.3.8 การทำเทรซโฮลต์ภาพไบนารีและการรวมสีกับภาพต้นฉบับ .....	62
3.2.3.9 การแปลงภาพสีเป็นภาพ YUV .....	62
3.2.3.10 การตรวจจับสีผิว .....	64
3.2.3.11 การปรับความเรียบของภาพ .....	65
3.2.3.12 การทำเทรซโฮลต์.....	66
3.2.3.13 การลบใบหน้า .....	66
3.2.3.14 การหาเส้นขอบของมือ .....	67

3.2.3.15 การหาจุดศูนย์กลางของฝ่ามือและการกำหนดรัศมี.....	68
3.2.3.16 การกำหนดตำแหน่งของปลายนิ้วมือ.....	69
3.2.3.17 กฎการจำแนก (Rule Based Classifier).....	70
บทที่ 4 ผลการทดลอง.....	79
4.1 ผลการทดลองของอัลกอริทึมรู้จำท่ามือแบบสถิติสำหรับผู้สูงอายุโดยใช้กฎการจำแนก .....	79
4.2 ผลการทดลองของอัลกอริทึมสำหรับผู้สูงอายุโดยใช้ Optical Flow และ SURF .....	80
4.3 ผลการทดลองของอัลกอริทึมที่นำเสนอสำหรับระบบรู้จำท่าบนพื้นฐานการมองเห็น สำหรับผู้สูงอายุ.....	83
4.3.1 การทดลองท่ามือขณะอยู่นิ่ง.....	83
4.3.1.1 การทดลองท่ามือขณะอยู่นิ่งบนฉากหลังแบบเรียบ.....	83
4.3.1.2 การทดลองท่ามือขณะอยู่นิ่งบนฉากหลังแบบซับซ้อน.....	85
4.3.2 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกัน.....	87
4.3.2.1 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนฉากหลังแบบ เรียบ .....	87
4.3.2.2 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนฉากหลังแบบ ซับซ้อน.....	90
4.3.3 การทดลองท่ามือเคลื่อนไหวในองศาที่แตกต่างกัน.....	95
4.3.4 การทดลองท่ามือเคลื่อนไหวในมุมมองที่แตกต่างกัน.....	97
4.3.5 การทดลองท่ามือเคลื่อนไหวในระยะที่แตกต่างกัน.....	99
4.3.6 การทดลองการส่งข้อความแจ้งเตือนบนสมาร์ตโฟน .....	103
4.4 การทดสอบกับผู้สูงอายุ .....	104
4.5 การเปรียบเทียบค่าความแม่นยำของการจำแนกท่ามือ.....	110
4.4 การออกแบบหน้าจ้อินเตอร์เฟซ .....	111
บทที่ 5 สรุป อภิปรายผล และข้อเสนอแนะ .....	114

5.1	สรุปผลการทดลอง.....	114
5.1.1	อัลกอริธึมรู้จำท่ามือแบบสถิติสำหรับผู้สูงอายุที่ใช้กฎการจำแนก.....	114
5.1.2	อัลกอริธึมรู้จำท่ามือแบบสถิติสำหรับผู้สูงอายุที่ใช้โดยใช่ Optical Flow และ SURF .....	115
5.1.3	อัลกอริทึมที่นำเสนอสำหรับระบบรู้จำท่าบนพื้นฐานการมองเห็นสำหรับผู้สูงอายุ ....	115
5.2	อภิปรายผล.....	117
5.2.1	ท่ามือขณะอยู่นิ่ง .....	117
5.2.2	ท่ามือเคลื่อนไหวในตำแหน่งที่ต่างกัน .....	118
5.2.3	ท่ามือเคลื่อนไหวในองศาที่ต่างกัน .....	119
5.2.4	ท่ามือเคลื่อนไหวในมุมมองที่ต่างกัน .....	120
5.2.5	ท่ามือเคลื่อนไหวในระยะที่ต่างกัน .....	121
5.2.6	การส่งข้อความแจ้งเตือนบนสมาร์ตโฟน .....	121
5.2.7	การเปรียบเทียบค่าความแม่นยำของการจำแนกท่ามือ .....	122
5.3	ข้อเสนอแนะ .....	122
	รายการอ้างอิง .....	123
	ประวัติผู้เขียนวิทยานิพนธ์ .....	127

## สารบัญตาราง

ตารางที่ 2.1 รายละเอียดบอร์ด Raspberry Pi 3 Model B .....	30
ตารางที่ 4.1 ผลการทดลองของการจำแนกท่ามือแบบเคลื่อนไหว .....	81
ตารางที่ 4.2 การทดลองท่ามือขณะอยู่นิ่งบนฉากหลังแบบเรียบ.....	84
ตารางที่ 4.3 การทดลองท่ามือขณะอยู่นิ่งบนฉากหลังแบบซับซ้อน.....	85
ตารางที่ 4.4 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนฉากหลังแบบเรียบ .....	87
ตารางที่ 4.5 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนพื้นหลังแบบซับซ้อน.....	90
ตารางที่ 4.6 การทดลองท่ามือเคลื่อนไหวในองศาที่แตกต่างกัน .....	95
ตารางที่ 4.7 การทดลองท่ามือขณะเคลื่อนไหวที่ในมุมมองที่แตกต่างกัน .....	97
ตารางที่ 4.8 การทดลองท่ามือเคลื่อนไหวในระยะที่แตกต่างกัน .....	99
ตารางที่ 4.9 การทดลองการส่งข้อความแจ้งเตือนบนสมาร์ตโฟน .....	103
ตารางที่ 4.10 การทดสอบท่ามือที่ผู้สูงอายุสามารถทำได้โดยง่ายในแต่ละวัย .....	105
ตารางที่ 4.11 การเปรียบเทียบค่าความแม่นยำในการจำแนกท่ามือขณะอยู่นิ่งบนพื้นหลังแบบไม่ซับซ้อน .....	110

## สารบัญรูปภาพ

รูปที่ 2.1 การแปลงภาพแอนะล็อกเป็นภาพดิจิทัล.....	12
รูปที่ 2.2 แสดงโมเดลสี RGB .....	13
รูปที่ 2.3 แสดงโมเดลสี YUV .....	14
รูปที่ 2.4 ภาพระดับเทา .....	15
รูปที่ 2.5 ตัวอย่างภาพไบนารีแสดงความสัมพันธ์ระหว่างค่าและสีของพิกเซล .....	16
รูปที่ 2.6 กระบวนการรับภาพ.....	17
รูปที่ 2.7 กระบวนการขยายขนาดด้วยส่วนประกอบโครงสร้างขนาด 3x3 .....	20
รูปที่ 2.8 กระบวนการการกรองขนาดด้วยส่วนประกอบโครงสร้างขนาด 3x3.....	20
รูปที่ 2.9 แสดงรูปแบบของวิธี Haar-like features [16] .....	21
รูปที่ 2.10 วิธีการหาความยาวส่วนโค้ง .....	23
รูปที่ 2.11 เส้นขอบ คอนเวกซ์ฮัลล์ และจุดที่อยู่ลึกที่สุดในรอยเว้า .....	27
รูปที่ 2.12 การหาระยะทางระหว่างจุดในปริภูมิ 2 มิติ.....	27
รูปที่ 2.13 การวิเคราะห์เชิงพื้นที่โดยตัวกรองที่เชื่อมกับภาพอินทิกรัล [26] .....	29
รูปที่ 2.14 ผลงานแสดงหลักการในการตัดสินใจทำมือ .....	29
รูปที่ 2.15 แสดงการเชื่อมต่อระหว่างบอร์ด Raspberry Pi และสมาร์ตโฟน.....	31
รูปที่ 3.1 แผนผังแสดงฮาร์ดแวร์ระบบรู้จำท่าทางมือแบบเคลื่อนไหว .....	32
รูปที่ 3.2 บอร์ด Raspberry PI 3 model B และโมดูลกล้อง V2.....	33
รูปที่ 3.3 แผนผังแสดงการทำงานของระบบรู้จำท่าทางมือแบบเคลื่อนไหว.....	33
รูปที่ 3.4 แผนผังการทำงานของอัลกอริธึมรู้จำท่าทางมือแบบสถิตสำหรับผู้สูงอายุโดยใช้กฎการ จำแนก .....	34
รูปที่ 3.5 การกำหนดบริเวณที่สนใจ .....	35
รูปที่ 3.6 ภาพระดับเทาของท่ามือทั้ง 6 ท่าที่ได้จากการแปลงภาพสี .....	35

รูปที่ 3.7 ภาพที่ได้จากการลบพื้นหลัง .....	36
รูปที่ 3.8 ผลลัพธ์การตรวจจับเส้นขอบมือ .....	38
รูปที่ 3.9 ภาพผลลัพธ์จากการหาจุดโมเมนต์มือ.....	40
รูปที่ 3.10 ภาพผลลัพธ์จากการหาจุดลึกสุดในรอยนิ้ว.....	40
รูปที่ 3.11 กฎการจำแนก.....	41
รูปที่ 3.12 ระบบแสดงผลการจำแนกความหมายของท่ามือทั้ง 6 ท่ามือ.....	43
รูปที่ 3.13 แผนผังการทำงานของอัลกอริธึมรู้จำท่ามือโดยใช้ Optical Flow และ SURF .....	44
รูปที่ 3.14 การนำเข้าวิดีโอ .....	45
รูปที่ 3.15 ภาพต้นฉบับของท่ามือทั้ง 6 ท่ามือที่ได้จากไฟล์วิดีโอ.....	45
รูปที่ 3.16 การแปลงภาพสีเป็นภาพระดับเทา .....	46
รูปที่ 3.17 การวิเคราะห์การเคลื่อนที่โดยใช้วิธีการไหลของแสงในเฟรมที่ต่อเนื่องกัน .....	47
รูปที่ 3.18 วิธีการเลื่อนหน้าต่าง .....	48
รูปที่ 3.19 ตัวกรองมัลติสเกลบนภาพท่ากำมือ .....	48
รูปที่ 3.20 กระบวนการการปรับปรุงโครงสร้างภาพ .....	48
รูปที่ 3.21 กระบวนการ Flood fill.....	49
รูปที่ 3.22 ภาพท่ามือแบบเคลื่อนไหว 6 ท่าจากกระบวนการการวิเคราะห์หับลือบ .....	50
รูปที่ 3.23 การจับคู่จุดที่สนใจของท่ามือเคลื่อนไหวโดยใช้อัลกอริธึม SURF .....	52
รูปที่ 3.24 ขั้นตอนการทำงานของระบบรู้จำท่าทางมือแบบเคลื่อนไหว .....	53
รูปที่ 3.25 แผนผังแสดงการรับภาพท่ามือแบบเคลื่อนไหวจากระบบ .....	54
รูปที่ 3.26 การทำงานของการรวมตัวจำแนกกลุ่มแบบต่อเรียง .....	55
รูปที่ 3.27 การปรับภาพเรียบหลังจากการรับภาพจากกล้อง .....	56
รูปที่ 3.28 แสดงการปรับภาพเรียบของแต่ละช่องสีของโมเดล RGB .....	57
รูปที่ 3.29 การลบพื้นหลังในแต่ละองค์ประกอบสี .....	58
รูปที่ 3.30 การทำเทรซโฮลต์ในแต่ละองค์ประกอบสี.....	59



รูปที่ 3.31 เฟรมภาพที่ได้จากการรวมองค์ประกอบสี R G และ B .....	59
รูปที่ 3.32 การเปลี่ยนแปลงลักษณะรูปร่างหรือโครงร่างของภาพ .....	61
รูปที่ 3.33 การเบลอภาพไบนารี .....	61
รูปที่ 3.34 การทำเทรซโฮลด์ภาพไบนารี .....	62
รูปที่ 3.35 การรวมภาพไบนารีเทรซโฮลด์กับพิกเซลสีจากภาพต้นฉบับ .....	62
รูปที่ 3.36 เฟรมภาพที่ได้จากการรวมองค์ประกอบ U และ V ของโมเดลสี YUV .....	63
รูปที่ 3.37 การทำเทรซโฮลด์ตรวจจับสีผิวขององค์ประกอบ U และ V ของโมเดลสี YUV .....	64
รูปที่ 3.38 การรวมองค์ประกอบสี U และ V จากการทำเทรซโฮลด์ตรวจจับสีผิวของโมเดลสี YUV .....	65
รูปที่ 3.39 การปรับความเรียบของภาพที่ได้จากการตรวจจับสีผิว .....	65
รูปที่ 3.40 การลบใบหน้า .....	66
รูปที่ 3.41 การหาเส้นขอบของมือ .....	67
รูปที่ 3.42 การหาจุดศูนย์กลางของมือและการกำหนดรัศมี .....	69
รูปที่ 3.43 พารามิเตอร์ที่เกี่ยวข้องในการหาท่ามือ .....	70
รูปที่ 3.44 แสดงความหมายท่ากำมือ .....	72
รูปที่ 3.45 แสดงความหมายท่าชูนิ้วโป่ง เทรซโฮลด์มากกว่า 0.4 .....	73
รูปที่ 3.46 แสดงความหมายท่าชูนิ้วชี้ เทรซโฮลด์น้อยกว่าหรือเท่ากับ 0.4 .....	73
รูปที่ 3.47 แสดงความหมายท่าชูนิ้วชี้และนิ้วกลาง .....	74
รูปที่ 3.48 แสดงความหมายท่าชูนิ้วโป่ง นิ้วชี้ และนิ้วก้อย เทรซโฮลด์มากกว่า 0.7 .....	75
รูปที่ 3.49 แสดงความหมายท่าชูนิ้วชี้ นิ้วกลาง และนิ้วนาง เทรซโฮลด์น้อยกว่าหรือเท่ากับ 0.7 .....	75
รูปที่ 3.50 แสดงความหมายท่าชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย .....	76
รูปที่ 3.51 แสดงความหมายท่าแบมือ .....	77
รูปที่ 3.52 แสดงความหมายท่าโบกมือ ระยะนิ้วกลางระหว่างเฟรมมากกว่า 30 จุดพิกัด และมากกว่า 4 เฟรม .....	77

รูปที่ 3.53	แผนผังแสดงกฎการจำแนก.....	78
รูปที่ 4.1	ผลลัพธ์การจำแนกท่าแบมือแบบภาพนิ่งในมุมมองต่าง ๆ .....	79
รูปที่ 4.2	แสดงผลท่ามือจากข้างซ้ายไปข้างขวาจำนวน 6 ท่าในองศาที่แตกต่างกัน .....	80
รูปที่ 4.3	เวกเตอร์เคลื่อนที่จากการใช้วิธีการไหลของแสง.....	82
รูปที่ 4.4	การปรับปรุงโครงร่างภาพท่ามือ .....	82
รูปที่ 4.5	ผลลัพธ์จากการใช้การวิเคราะห์หีบห่อ การไหลของแสง และ SURF.....	82
รูปที่ 4.6	การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนฉากหลังแบบเรียบ .....	90
รูปที่ 4.7	การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนฉากหลังแบบซับซ้อน.....	94
รูปที่ 4.8	แสดงองศาที่แตกต่างกันของท่ามือแต่ละท่า .....	96
รูปที่ 4.9	แสดงผลท่ามือที่ระบบจำแนกผิดพลาด.....	97
รูปที่ 4.10	แสดงท่ามือเคลื่อนไหวในมุมมองที่แตกต่างกัน .....	99
รูปที่ 4.11	แสดงท่ามือเคลื่อนไหวในระยะที่แตกต่างกัน .....	102
รูปที่ 4.12	ตัวอย่างการส่งข้อความแจ้งเตือนบนสมาร์ตโฟน .....	104
รูปที่ 4.13	ผู้สูงอายุหญิงวัย 60 ปี.....	107
รูปที่ 4.14	ผู้สูงอายุหญิงวัย 70 ปี.....	108
รูปที่ 4.15	ผู้สูงอายุชายวัย 84 ปี.....	109
รูปที่ 4.16	แสดงแถบหน้าหลักของหน้าจออินเตอร์เฟซของระบบก่อนการจับภาพฉากหลัง.....	111
รูปที่ 4.17	แสดงแถบหน้าหลักของหน้าจออินเตอร์เฟซของระบบหลังการจับภาพฉากหลัง .....	112
รูปที่ 4.18	แสดงแถบการตั้งค่าของหน้าจออินเตอร์เฟซของระบบ .....	113
รูปที่ 4.19	แสดงแถบเกี่ยวกับเราของหน้าจออินเตอร์เฟซของระบบ .....	113

## บทที่ 1

### บทนำ

#### 1.1 ความเป็นมา

ในปัจจุบันเทคโนโลยีมีความเจริญก้าวหน้าอย่างรวดเร็ว และได้รับการพัฒนาให้มีประสิทธิภาพ เพื่อตอบสนองความต้องการของผู้ใช้งาน มีเทคโนโลยีมากมายที่มีส่วนเกี่ยวข้องในชีวิตประจำวันของมนุษย์ อาทิ สมาร์ทโฟน แท็บเล็ต กล้องถ่ายรูป กล้องวงจรปิด เป็นต้น โดยสิ่งที่สำคัญสำหรับการพัฒนาเทคโนโลยีคือ การปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์ (Human computer interaction: HCI) ซึ่งถือเป็นศาสตร์การเรียนรู้แบบบูรณาการที่นำความรู้หลายแขนงมาประยุกต์ใช้ให้เกิดประโยชน์สูงสุด อาทิ วิทยาการคอมพิวเตอร์ พฤติกรรมศาสตร์ จิตวิทยา และการออกแบบ โดยจะให้ความสำคัญกับผู้ใช้งาน คอมพิวเตอร์ และวิธีการทำงานร่วมกันระหว่างผู้ใช้งานและคอมพิวเตอร์ ทั้งนี้การปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์สามารถใช้ในการพัฒนาเทคโนโลยี สิ่งอำนวยความสะดวก ใช้ด้านบันเทิง เช่น พัฒนาเกมส์ ใช้สำหรับการจัดการภาวะวิกฤต การบรรเทาภัยพิบัติ การติดต่อสื่อสารระหว่างมนุษย์และหุ่นยนต์ และระบบทางการแพทย์

ในการพัฒนาการปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์สำหรับระบบทางการแพทย์จะให้ความสำคัญที่ระบบเฝ้าระวังเป็นหลัก โดยนำมาใช้กับการดูแลสุขภาพเพื่อติดตามกิจกรรมทางกายภาพ ความรู้สึก และเฝ้าระวังการเกิดอาการของโรคของผู้ใช้งาน [1] ซึ่งจะเป็นประโยชน์ต่อคนไข้ ผู้พิการ และผู้สูงอายุ อย่างไรก็ตามเทคโนโลยีการปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์ในด้านทางการแพทย์ยังไม่มีการระบุกลุ่มเป้าหมายที่ชัดเจนนัก โดยเฉพาะอย่างยิ่งในการใช้งานกับกลุ่มผู้สูงอายุ จากรายงานการประมาณการจำนวนประชากรในผู้สูงอายุ [2] พบว่าผู้สูงอายุเป็นกลุ่มประชากรที่มีแนวโน้มมีอัตราการเพิ่มขึ้นของจำนวนประชากรมากที่สุดในปี พ.ศ. 2586 ดังนั้นผู้สูงอายุจึงควรได้รับการดูแลอย่างเหมาะสมมากขึ้น โดยส่วนมากแล้วผู้สูงอายุจะอยู่ภายใต้การดูแลของแพทย์ พยาบาล หรือผู้ดูแล ทั้งนี้หากผู้สูงอายุพักอยู่ที่โรงพยาบาล ผู้ดูแลจะไม่มีเวลามากพอสำหรับการดูแลคนไข้เพียงคนเดียวตลอดเวลา ซึ่งทำให้ผู้สูงอายุไม่สามารถที่จะสื่อสารหรือบอกความต้องการของตนได้ อาทิ เมื่อต้องการดื่มน้ำ เข้าห้องน้ำ และหรือแจ้งผู้ดูแลหากเกิดอาการผิดปกติหรือเกิดอุบัติเหตุขึ้น [3] ดังนั้นการใช้เทคโนโลยีระบบเฝ้าระวังจึงมีความจำเป็นสำหรับการติดตามกิจกรรมของผู้สูงอายุและแจ้งเตือนไปยังผู้ดูแล

การปฏิสัมพันธ์กับคอมพิวเตอร์มีความจำเป็นต้องใช้การสื่อสารแบบธรรมชาติ การใช้มือจัดเป็นหนึ่งในวิธีการสื่อสารแบบธรรมชาติที่สามารถสื่อสารได้ง่าย รวดเร็ว และตรงกับความรู้และสัญชาตญาณ

ญาณของมนุษย์ เนื่องจากมนุษย์สามารถควบคุมการเคลื่อนไหวและท่าทางของมือเพื่อสื่อความหมายที่แตกต่างกันออกไปได้ [4] การใช้มือได้รับความนิยมนำมาใช้สำหรับเทคโนโลยีการสื่อสารระหว่างมนุษย์และคอมพิวเตอร์ โดยสามารถแบ่งประเภทวิธีการออกเป็น 2 ประเภท ได้แก่ วิธีการบนพื้นฐานของอุปกรณ์ (Device-based method) และวิธีการบนพื้นฐานของการมองเห็น (Vision-based method) สำหรับวิธีการบนพื้นฐานของอุปกรณ์ ผู้สูงอายุจำเป็นต้องติดอุปกรณ์ไว้บนร่างกายเพื่อให้ระบบสามารถตรวจจับท่าทางมือได้ เช่น การติดเซ็นเซอร์บริเวณข้อมือ การติดป้ายจุดระบุตำแหน่งมือและนิ้วมือ การใส่ถุงมือ และการเชื่อมสายระหว่างผู้สูงอายุและอุปกรณ์เพื่อเชื่อมต่อกับคอมพิวเตอร์ เป็นต้น ซึ่งทำให้ยุ่งยากต่อผู้ใช้งานและเสี่ยงต่อการเกิดโรคมะเร็ง [5] ในขณะที่วิธีการบนพื้นฐานของการมองเห็นสามารถทำงานได้โดยไม่ต้องมีอุปกรณ์ใดๆติดกับร่างกายผู้สูงอายุ ดังนั้นเทคโนโลยีบนพื้นฐานของการมองเห็นจึงให้ผลลัพธ์ที่ดีกว่าในแง่ของการใช้งาน ทำให้ผู้สูงอายุมีความสะดวกสบายในการใช้งานมากขึ้น

สำหรับการตรวจจับและรู้จำท่าทางมือสามารถแบ่งตามพื้นฐานการมองเห็นได้เป็น 2 ประเภท ได้แก่ การรู้จำท่าทางมือแบบสถิต และการรู้จำท่าทางมือแบบเคลื่อนไหว โดยการรู้จำท่าทางมือแบบสถิตจะตรวจจับและจำแนกท่าทางมือในขณะที่อยู่นิ่ง แต่สำหรับผู้สูงอายุแล้วการคงอยู่ในท่าใดท่าหนึ่งนานๆอาจก่อให้เกิดการปวดเมื่อยกล้ามเนื้อ เนื่องจากความผิดปกติที่ระบบประสาทส่วนกลางหรือเรียกว่าโรคพาร์กินสันที่ส่งผลให้การแสดงท่าทางมือของผู้สูงอายุมีอาการสั่น เกร็ง และเคลื่อนไหวช้า [6] จากอาการดังกล่าวส่งผลให้ผู้สูงอายุบางรายไม่สามารถแสดงท่าทางมือแบบสถิตได้ ดังนั้นในงานวิจัยนี้ผู้จัดทำจึงนำเสนอระบบรู้จำท่าทางมือแบบเคลื่อนไหวบนพื้นฐานของการมองเห็นสำหรับผู้สูงอายุแบบเวลาจริง (Real Time) เพื่อช่วยเหลือผู้สูงอายุให้สามารถติดต่อกับผู้อื่นแม้ในขณะที่อยู่คนเดียวได้

## 1.2 ทบทวนวรรณกรรม

การรู้จำท่าทางมือบนพื้นฐานของการมองเห็นจะให้ความสำคัญที่การพัฒนาอัลกอริทึม และแอปพลิเคชันสำหรับการเรียนรู้ของเครื่อง (Machine Learning) ในการพัฒนาอัลกอริทึมจำนวนมากที่ใช้ในการตรวจจับ ติดตาม และรู้จำท่าทางมือ ในกระบวนการการรู้จำบนพื้นฐานของการมองเห็นดำเนินการโดยใช้กล้องหนึ่งตัวหรือหลายตัวในการรับวิดีโอสตรีมเป็นข้อมูลนำเข้าแบบเวลาจริง บางเทคนิคอาจประมวลผลแบบง่ายและไม่มีความทนทานต่อฉากที่มีการเคลื่อนไหวหรือมีวัตถุอยู่เป็นจำนวนมาก ข้อดีของการรู้จำบนพื้นฐานของการมองเห็นคือ ใช้งานง่าย ไม่เสี่ยงรบกวน ราคาถูก มีความปลอดภัย และมีความยืดหยุ่นในการใช้งาน คือสามารถประมวลผลด้านอื่นนอกเหนือจากการรู้จำท่าทางมือได้ สำหรับท่าทางที่ใช้ในการรู้จำบนพื้นฐานของการมองเห็น ควรเป็นท่าที่แสดงได้ชัดเจน ง่ายต่อการจำและการแสดงท่าทางของผู้สูงอายุ มีความชัดเจนและเป็นธรรมชาติ ไม่ควรใช้ท่ามือที่

แสดงได้ยากและมีความหมายซับซ้อน ทั้งนี้ความหมายของแต่ละท่ามือขึ้นอยู่กับบริบทการตกลงร่วมกันระหว่างผู้สูงอายุและผู้ดูแล เนื่องจากท่ามือของแต่ละบุคคลจะมีความแตกต่างกันตามพื้นฐานความรู้ความเข้าใจในท่าทางนั้นๆ และวัฒนธรรม สิ่งที่สำคัญสำหรับการรู้จำท่ามือคือ ท่ามือจะต้องสอดคล้องกับความหมายที่กำหนดขึ้น เช่น กำมือและชูนิ้วโป้งหมายถึงตกลง ชี้นิ้วชี้หมายถึงชี้ทิศทางที่ต้องการไป เป็นต้น

การรู้จำท่าทางมือแบบเคลื่อนไหวจะพิจารณาการเคลื่อนที่ของมือ ตำแหน่งมือ และการปรับองศามือ ณ เวลาใดๆ จัดเป็นท่าทางมือแบบต่อเนื่อง (Conscious Dynamic Gestures) แบ่งออกได้เป็น 4 ประเภท ได้แก่ การควบคุม (Regulator) คือการใช้ท่ามือเพื่อควบคุมการสั่งการทำงาน การแสดงผลกระทบ (Affect Display) ใช้แสดงออกทางอารมณ์ สัญลักษณ์ (Emblem) มีลักษณะคล้ายภาษามือ โดยขึ้นอยู่กับสภาพแวดล้อมและวัฒนธรรม มีความหมายในตัวเอง สามารถเรียนรู้ได้ เช่น การโบกมือหมายถึงทักทายหรือลาก่อน และภาพประกอบ (Illustrator) คือการใช้ท่ามือเพื่อแสดงถึงคำพูด โดยสามารถแบ่งแยกย่อยออกมาได้เป็น 5 ประเภท ได้แก่ ท่าทางที่ติดต่อกัน (Cohesive Gesture) เป็นวิธีการเชื่อมต่อส่วนต่างๆของการเล่าเรื่องราวร่างกาย ท่าทางการเปรียบเทียบ (Metaphoric Gesture) คือสัญลักษณ์มือที่แสดงถึงความคิดแบบนามธรรม ท่าทางแบบชี้เฉพาะ (Deictic Gesture) คือการชี้วัตถุจริงหรือสถานที่ที่เกิดขึ้น เช่นการเริ่มต้นการสนทนาหรือการเริ่มต้นพูดหัวข้อใหม่ที่อ้างอิงถึงวัตถุ ท่าทางสัญลักษณ์ (Iconic Gesture) คือสัญลักษณ์มือที่แสดงความหมายในรูปแบบทางกายภาพและลักษณะที่เกิดขึ้นร่วมกับการพูด และท่าทางจังหวะ (Beat Gesture) จะตรงข้ามกับท่าทางสัญลักษณ์ โดยแบบจังหวะจะแสดงการเคลื่อนไหวของมือแบบเร็วและสั้นเพื่อนำเสนอให้น้อยที่สุด สำหรับการเคลื่อนไหวของมือสองข้างแบบต่อเนื่องเพื่อใช้ในการสื่อสารด้วยภาษามือนั้น (American Sign Language : ASL) เป็นความท้าทายสำหรับการพัฒนาอัลกอริทึมในอนาคต เนื่องจากระบบต้องสามารถประมวลผลรวดเร็วและมีความทนทานในการตรวจจับและรู้จำข้อมูลที่มีขนาดใหญ่ที่มีความแตกต่างกันทั้งท่าทางและจำนวนมือในแต่ละท่าได้

งานวิจัยที่เกี่ยวข้องสำหรับการรู้จำท่าทางมือมีดังนี้

Sawasdee, S., Pumrin, S. [7] ได้นำเสนอการรู้จำท่าทางมือแบบสถิตในเวลาจริงโดยใช้คุณลักษณะแบบฮาร์ (Haar-like Features) และอัลกอริทึมการเรียนรู้แบบเอดาบู้สต์ (Adaboost Learning Algorithm) พัฒนาระบบปฏิบัติการแอนดรอยด์สำหรับการรู้จำท่าทางมือ เนื่องจากการใช้เทคนิคฮาร์ต้องการรูปแบบภาพนำเข้าที่แน่นอน เช่น ใบหน้า จึงเหมาะสมกับการตรวจจับใบหน้ามากกว่าการรู้จำท่าทางมือ ระบบให้ความถูกต้องเท่ากับ 87.45% สำหรับทดสอบวิธีการ และ 70.29% สำหรับการทดสอบในสถานการณ์จำลอง โดยค่าความถูกต้องขึ้นอยู่กับอัตราการรู้จำของระบบ

Ohn-Bar, E., Trivedi, M. M. [8] ได้นำเสนอการใช้เทคนิคโมเดลสี (RGB) ร่วมกับตัวบ่งชี้ความลึก (Depth Descriptor) เพื่อใช้ตรวจจับท่าทางมือในบริเวณที่สนใจ (Region of Interest; ROI) สำหรับการปฏิสัมพันธ์ระหว่างผู้ใช้งานและระบบต่อเนื่องหลายรูปแบบในยานพาหนะ เช่นการสั่งคำสั่งบนอินเทอร์เน็ตเฟสบริเวณตรงกลางของคอนโซลรถยนต์ ทั้งนี้การใช้โมเดลสีในการตรวจจับจะมีความไวต่อการเปลี่ยนแปลงของแสง และจากการทดลองพบว่าระยะห่างของเซ็นเซอร์และท่ามือมีผลให้เกิดการทับซ้อนของเงามือของผู้ใช้งาน

Liu, K. and Kehtarnavaz, N. [9] นำเสนอการรู้จำท่าทางมือโดยใช้ภาพสเตอริโอแบบเวลาจริง โดยพัฒนาความทนทานในการตรวจจับและการรู้จำท่าทางการเคลื่อนไหวของมือและนิ้วมือโดยพิจารณาจากความละเอียดต่ำของภาพสเตอริโอถ่ายจากกล้องสเตอริโอ อัลกอริทึมประมวลผลทั้งมือซ้ายและขวาเพื่อเพิ่มประสิทธิภาพของการรู้จำมือ โดยใช้กฎการจำแนก (Rule-based) ช่วยขั้นตอนการตรวจจับกับวิธีตรวจจับเส้นขอบ (Contour Detection) และหาเปลือกนูน (Convex Hull) ที่นิ้ว ใช้รูปแบบแจกแจงผสมแบบเกาส์ (Gaussian Mixture Models: GMM) และรูปแบบสี CrCb ร่วมกับ CamShift สำหรับติดตามมือ และจำแนกท่าทางมือด้วย DTW ความถูกต้องของระบบรู้จำมือเคลื่อนไหวเท่ากับ 93% และนิ้วมือเท่ากับ 92%

Shan, C. et al. [10] นำเสนอระบบติดตามท่าทางมือแบบเคลื่อนไหวในเวลาจริง โดยพัฒนาอัลกอริทึมซึ่งรวมข้อดีของวิธี Mean Shift (MS) และ Particle Filter (PF) เรียกว่า MSEPE เพื่อนำมาพัฒนาการติดตามตำแหน่งของมือสำหรับการควบคุมอินเทอร์เน็ตเฟสของรถเข็นในสภาพแวดล้อมแบบเคลื่อนไหว จากการเปรียบเทียบ MSEPE กับ MS และ PF เพื่อหาผลลัพธ์ที่ดีที่สุดสำหรับการติดตามและการรู้จำท่าทางมือ พบว่า PF และ MSEPE ให้ประสิทธิภาพในการติดตามมือได้สูงที่สุดที่ 100% รองลงมาคือ MS มีค่าเท่ากับ 73%

Hsieh, C. C. และ Liou, D. H. [11] นำเสนอการรู้จำท่าทางมือด้วยวิธีคุณลักษณะฮาร์ (Haar features) และ Support Vector Machine (SVM) เพื่อใช้สำหรับการควบคุมเครื่องใช้ไฟฟ้าภายในบ้านแบบเวลาจริง ขั้นตอนการประมวลผลประกอบด้วย 3 ขั้นตอน ได้แก่ การขยายแบบดิจิทัลสำหรับระบุบริเวณใบหน้าของผู้ใช้งาน การประยุกต์ใช้การตรวจจับสีผิวบริเวณมือ โดยการตรวจจับสีผิวนี้จะต้องมีความทนทานต่อการเปลี่ยนสภาพแสง และกระบวนการการรู้จำท่าทางมือทั้งแบบสถิตย์และแบบเคลื่อนไหว สำหรับแบบเคลื่อนไหวจะใช้ Motion History Image (MHI) ร่วมกับคุณลักษณะฮาร์หลังจากขั้นตอนการขยายแบบดิจิทัล ผลลัพธ์ของการรู้จำท่าทางมือแบบเคลื่อนไหวมีความถูกต้องเท่ากับ 95.66%

Suk, H.I. et al. [12] นำเสนอการรู้จำท่าทางมือแบบเคลื่อนไหว โดยข้อมูลนำเข้ามาจากไฟล์วิดีโอ ระบบประมวลผลโดยใช้วิธี dynamic Bayesian network (DBN) เพื่อเพิ่มประสิทธิภาพในการ

ควบคุมการแสดงสีมีเดียและการนำเสนอสไลด์ กระบวนการหลักๆประกอบด้วย การสกัดคุณลักษณะของมือโดยใช้การตรวจจับสีผิว(Skin Color Detection) ในโมเดลสี YIQ การสร้างรูปแบบมือด้วยเครือข่ายวงกลม (Cyclic Network) การติดตามการเคลื่อนไหวของมือด้วยบล็อบบlob (Blob tracking) และการไหลของแสง (Optical Flow) และการรู้จำท่าทางมือด้วย DBN พบว่าวิธีรู้จำแบบ DBN ให้ค่าความถูกต้องสำหรับท่ามือแบบภาพเดี่ยวที่ 99.59% และท่ามือแบบภาพเคลื่อนไหวต่อเนื่องที่ 80.77%

### 1.3 วัตถุประสงค์ของงานวิจัย

1. เพื่อศึกษาทฤษฎีที่ใช้ในการจำแนกความแตกต่างของท่ามือในแต่ละรูปแบบและศึกษาการทำงานบนระบบสมองกลฝังตัว
2. เพื่อสร้างระบบที่สามารถตรวจจับและจำแนกท่ามือแบบเคลื่อนไหวได้ในเวลาจริง
3. เพื่อแสดงความหมายของท่ามือในแต่ละรูปแบบได้อย่างถูกต้อง

### 1.4 ขอบเขต

1. ระบบทำงานบนบอร์ด Raspberry Pi และโมดูลกล้อง
2. ระบบจำแนกท่ามือแบบเคลื่อนไหวจำนวน 9 ท่ามือ ได้แก่ ท่ากำมือ ท่าชูหนึ่งนิ้ว(นิ้วชี้) ท่าชูสองนิ้ว(นิ้วชี้และนิ้วกลาง) ท่าชูสามนิ้ว(นิ้วชี้ นิ้วกลาง และนิ้วนาง) ท่าชูสี่นิ้ว (นิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย) ท่าแบมือ ท่าชูนิ้วโป้ง ท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย และท่าโบกมือ
3. ระบบสามารถตรวจจับระยะห่างจากกล้องถึงท่ามือไม่เกิน 2 เมตร และฉากหลังไม่ควรมีการเคลื่อนไหวนอกจากท่ามือ

### 1.5 ประโยชน์

1. ช่วยอำนวยความสะดวกให้แก่ผู้สูงอายุให้สามารถสื่อสารกับผู้ดูแลได้ในขณะที่อยู่คนเดียว
2. ช่วยอำนวยความสะดวกให้แก่การทำงานของผู้ดูแลและโรงพยาบาลในการเฝ้าระวังผู้สูงอายุ
3. ช่วยป้องกันการเกิดอุบัติเหตุที่อาจเกิดขึ้นกับผู้สูงอายุได้ทันเวลา

## บทที่ 2

### ทฤษฎีและหลักการที่เกี่ยวข้อง

ทฤษฎีและหลักการที่เกี่ยวข้องสามารถแบ่งออกได้เป็น 5 ขั้นตอนหลัก ได้แก่

ความรู้ด้านการประมวลผลภาพ (Image processing)

2.1 การประมวลผลเบื้องต้น (Pre-processing)

2.2 การตรวจจับมือ (Hand detection)

2.3 การติดตามมือ (Hand tracking)

2.4 การวิเคราะห์ท่ามือและการตัดสินใจท่ามือ (Hand gesture recognition and decision)

2.5 บอร์ด Raspberry Pi และการเชื่อมต่อระหว่างบอร์ดและสมาร์ทโฟน (Raspberry Pi board and its connection with smart phone)

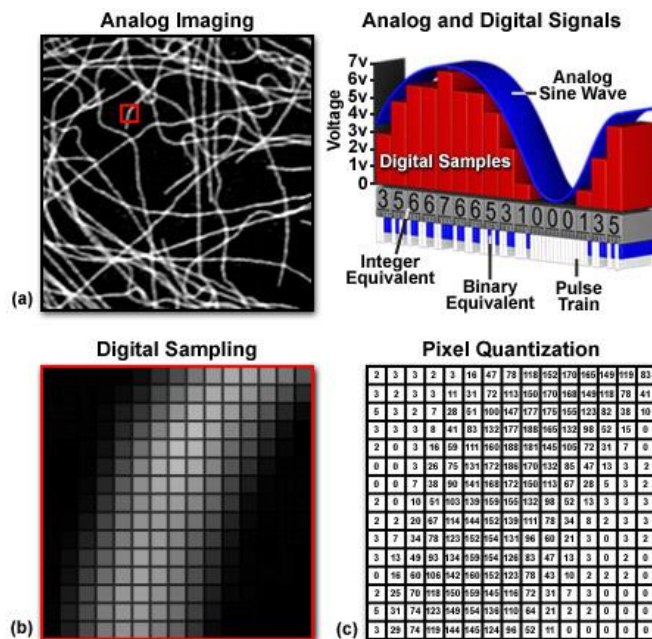
**2.1 ความรู้ด้านการประมวลผลภาพ (Image processing)**

**2.2.1 หลักการประมวลผลภาพ (Image processing)**

การประมวลผลภาพดิจิทัลเกี่ยวข้องกับการแปลงข้อมูลรูปที่เป็นสัญญาณแอนะล็อกให้อยู่ในรูปของสัญญาณดิจิทัล เพื่อใช้ในการประมวลผลผ่านทางคอมพิวเตอร์ได้ และยังสามารถนำมาใช้ในการลดปัญหาของภาพ เช่น ลดสัญญาณรบกวนภายในภาพ เป็นต้น

ในการแปลงภาพให้เป็นสัญญาณดิจิทัลนั้น ระบบจะนำรูปที่รับเข้ามาไปคำนวณ โดยกระบวนการ Sampling และ Quantization และส่งข้อมูลออกมาในรูปแบบดิจิทัล คอมพิวเตอร์จะเก็บข้อมูลภาพลงหน่วยความจำ โดยการจองหน่วยความจำภายในเครื่องในรูปแบบของอาร์เรย์ โดยค่าในแต่ละช่องของ อาร์เรย์แสดงถึงคุณสมบัติต่างๆ ของรูปที่จุดพิกเซลนั้นๆ และตำแหน่งของช่องอาร์เรย์ก็เป็นตัวกำหนดตำแหน่งของจุดพิกเซลภายในภาพด้วย [13]





รูปที่ 2.1 การแปลงภาพแอนะล็อกเป็นภาพดิจิทัล

ที่มา: <http://zeiss-campus.magnet.fsu.edu/articles/basics/digitalimaging.html>

ภาพดิจิทัลที่ได้จะมีรูปแบบการเก็บเป็นเมทริกซ์ ซึ่งจะมีการจัดเก็บภาพแต่ละชนิดต่างกัน ขึ้นอยู่กับโมเดลสีของภาพ อาทิ โมเดลสี RGB YUV นอกจากนี้ยังมีภาพที่ได้จากการแปลง อาทิ ภาพระดับเทา และภาพไบนารี ดังอธิบายดังนี้

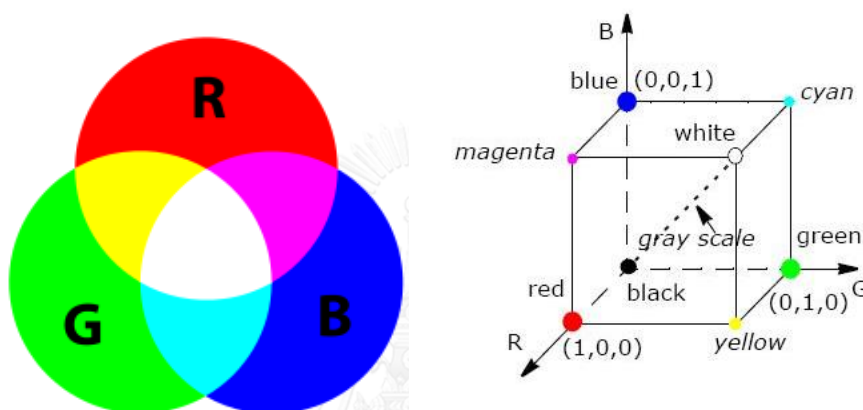
## 2.2.2 โมเดลสี RGB

ภาพสี RGB หรือ Truecolor Image เป็นรูปที่เก็บโดยใช้อาร์เรย์ 3 มิติ ขนาด  $m \times n \times 3$  โดยที่  $m$  คือความยาว และ  $n$  คือความกว้างของภาพในหน่วยพิกเซล ส่วนมิติสุดท้ายนั้น ในแต่ละมิติ จะเก็บค่าสีแยกกัน คือสีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) การรวมตัวของสีเป็นแบบ Additive กล่าวคือเมื่อรวมสีแดงและสีน้ำเงินจะได้สีม่วง เมื่อรวมสีน้ำเงินและสีเขียวจะได้สีฟ้า เมื่อรวมสีเขียวและสีแดงจะได้สีเหลือง เมื่อรวมสีหลักทั้งสามสีจะได้สีขาว ดังแสดงในรูปที่ 4.4

โดยปกติภาพที่ได้จากกล้องจะเป็นภาพสีระบบ RGB แต่เนื่องจากโมเดลสี RGB ไม่มีการแยกช่องสำหรับความสว่าง (Luminance) ออกมาจากช่องสี (Chrominance) ทำให้เมื่อมีการประมวลผลภาพจะส่งผลต่อสีช่องอื่น ๆ ได้ ดังนั้นเพื่อทำให้ระบบมีความทนทานต่อการเปลี่ยนแปลงสภาพแสง ภาพสี RGB จึงมักจะถูกนำไปแปลงเป็นโมเดลสีประเภทอื่น เช่น ภาพระดับเทา ภาพ HSV ภาพ

YCrCb เป็นต้น ในงานนี้ได้ใช้โมเดลสี HSV เนื่องจากให้ผลลัพธ์ที่ดี รวดเร็ว และสะดวกต่อการประมวลผล

โดยทั่วไปโมเดลสี RGB จะนำไปใช้ในจอภาพแบบ CRT (Cathode ray tube) ส่วนของที่ CIE (Commission International l 'Eclairage) พัฒนานั้นมีสีแดงเท่ากับ 700 นาโนเมตร สีเขียวเท่ากับ 546.1 นาโนเมตร และสีน้ำเงิน 435.8 นาโนเมตร เพื่อใช้สำหรับการแสดงภาพของจอภาพแบบ CRT และเป็นมาตรฐานสำหรับผู้ผลิตแบบ CRT ให้มีลักษณะเดียวกัน



(ก) การรวมกันขององค์ประกอบสี (ข) ระบบพิกัดคาร์ทีเซียนในลูกบาศก์

รูปที่ 2.2 แสดงโมเดลสี RGB

ที่มา: <https://software.intel.com/en-us/node/503873>

จากรูปที่ 2.2 พบว่า (ก) แสดงการรวมกันขององค์ประกอบสี (ข) แสดงโมเดลสี RGB ตามระบบระบบพิกัดคาร์ทีเซียนในลูกบาศก์ โดยที่แต่ละสีจะแทนด้วยค่าตำแหน่งในพิกัด ค่าสีระดับเทาจะไล่ระดับตั้งแต่สีดำไปจนถึงสีขาวที่มีค่าเท่ากับ (1,1,1) [14]

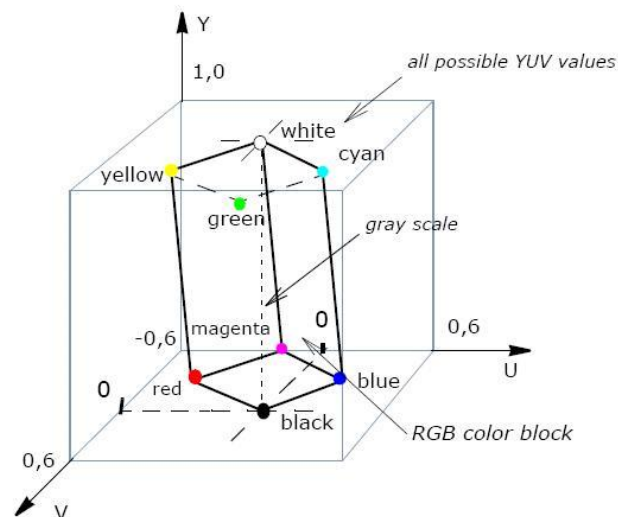
### 2.2.3 โมเดลสี YUV

โดยทั่วไปแล้วภาพต้นฉบับที่ได้รับจากกล้อง คือภาพสี RGB อย่างไรก็ตามช่องสีแบบ RGB แต่ละช่องสีนั้นจะขึ้นอยู่กับช่องสีอื่น ๆ ซึ่งหากเกิดการเปลี่ยนแปลงของสภาพแสงก็จะส่งผลกระทบต่อช่องสีทุกช่อง ดังนั้นการแปลงฐานข้อมูลสี (Color space conversion) จึงเป็นสิ่งสำคัญสำหรับการแยกแสง (Luminance) ออกจากองค์ประกอบสีภายในภาพ (Chrominance)

สัญญาณ YUV ถูกสร้างมาจากข้อมูลสี RGB (สีแดง สีเขียวและสีน้ำเงิน) ดั้งเดิมที่รับจากวัตถุ เมื่อนำเอาค่าของสีทั้งสามสีมารวมเข้าด้วยกันจะทำให้เกิดสัญญาณ Y (Luminance) เป็นผลให้เกิดส่วนที่

สว่างและส่วนที่มีมืดของภาพขึ้น สัญญาณ U เกิดจากการตัดข้อมูล Y ออกจากสัญญาณสีน้ำเงิน (B-Y) ส่วนสัญญาณ V เกิดจากการตัดข้อมูล Y ออกจากสัญญาณสีน้ำแดง (R-Y) นอกจากนี้โมเดลสี YUV จัดเป็นโมเดลสีที่มีความเหมาะสมกับการรับรู้ตามสัญชาตญาณของมนุษย์ ในปัจจุบันมีโมเดลสีมากมายที่ได้รับความนิยม อาทิ HSV, YUV, และ YCbCr เป็นต้น โมเดลสีเหล่านี้ประกอบไปด้วย 3 องค์ประกอบ ได้แก่ Hue Saturation และ Value พบว่า องค์ประกอบสี ได้แก่ Hue และ Saturation มีความทนทานต่อการเปลี่ยนแปลงของแสง

โมเดลสี YUV เป็นโมเดลพื้นฐานสำหรับการแพร่สัญญาณอนาล็อกของโทรทัศน์ ในการใช้งานโมเดลสี YUV พบว่าข้อดีคือมีการแยกองค์ประกอบของแสงและสี ทำให้เมื่อประมวลผลแล้วการประมวลผลขององค์ประกอบแสงจะไม่ส่งผลกระทบต่อองค์ประกอบสี นอกจากนี้โมเดลสี YUV ยังมีความใกล้เคียงกับโมเดลสี YCbCr ซึ่งจะแบ่งข้อมูลของจุดภาพตามการรับรู้ของประสาทตาของคน โดยจะแยกข้อมูลความสว่างออกจากข้อมูลระดับสี นั่นคือใช้ช่องประกอบ Y ในการเก็บค่าความสว่างของจุดภาพ ส่วน U คือค่าความแตกต่างระหว่างค่าสีน้ำเงินกับค่าความสว่าง (B-Y) องค์ประกอบ V คือค่าความแตกต่างระหว่างค่าสีแดงกับค่าความสว่าง (R-Y) ส่วนค่าความแตกต่างระหว่างค่าสีเขียวกับค่าความสว่างนั้นไม่จำเป็นต้องเก็บ เนื่องจากเมื่อรวมกันแล้วจะมีค่าเป็นค่าคงที่ อาจกล่าวได้ว่า องค์ประกอบ U และ V แทนองค์ประกอบ Cb และ Cr ตามลำดับ [14]



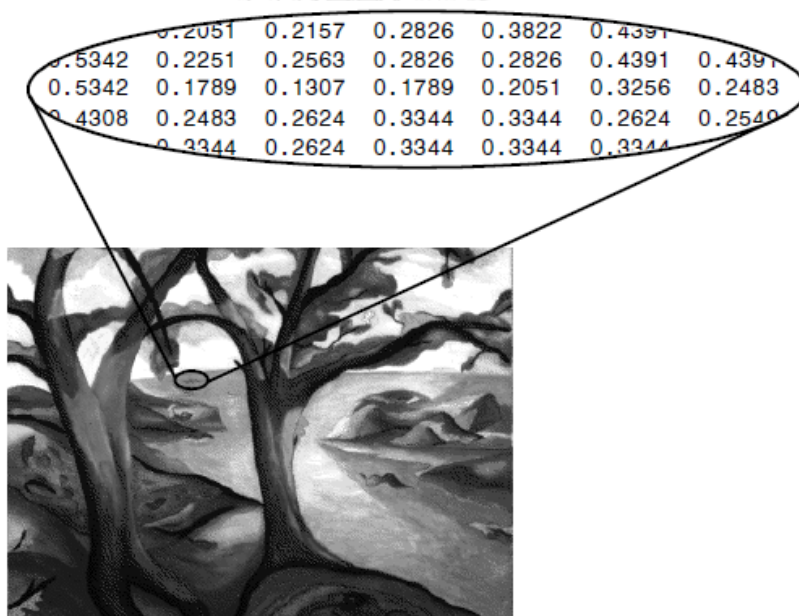
รูปที่ 2.3 แสดงโมเดลสี YUV

ที่มา: <https://software.intel.com/en-us/node/503873>

## 2.2.4 ภาพระดับเทา (Grayscale image)

ภาพระดับเทาเป็นภาพที่เก็บโดยใช้รูปแบบของอาร์เรย์ 2 มิติ โดยค่าที่เก็บจะมีค่าอยู่ในช่วงๆหนึ่ง ซึ่งระดับของสีขึ้นอยู่กับขนาดของบิตที่ใช้เก็บค่าสี หนึ่งในกระบวนการการเปลี่ยนโมเดลสีที่ใช้ในแอปพลิเคชันต่าง ๆ คือการเปลี่ยนภาพสี (RGB) เป็นภาพระดับเทา (Grayscale) เพราะเป็นกระบวนการที่ทำให้การประมวลผลภาพรวดเร็วขึ้น ซึ่งถ้านำภาพสีมาเข้ากระบวนการทำงานจะทำให้ระบบประมวลผลช้า เนื่องจากภาพสีแต่ละภาพจะประกอบไปด้วย 3 องค์ประกอบสี ได้แก่ ภาพโทนสีแดง (R) ภาพโทนสีเขียว (G) และภาพโทนสีน้ำเงิน (B) ฉะนั้นการประมวลผลก็ต้องเข้าถึงข้อมูลทั้งสามข้อมูล (แดง เขียว น้ำเงิน) ในทางกลับกันภาพระดับเทานั้นจะทำได้ง่ายและเร็วกว่า เพราะมีเพียงระดับความเข้มเท่านั้น โดยภาพระดับเทาจะวัดเฉพาะแสงสว่างเท่านั้น [15] การแปลงภาพสีเป็นภาพระดับเทาสามารถทำได้ดังสมการที่ 2.1 คิดค้นโดย Craig Markwart

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



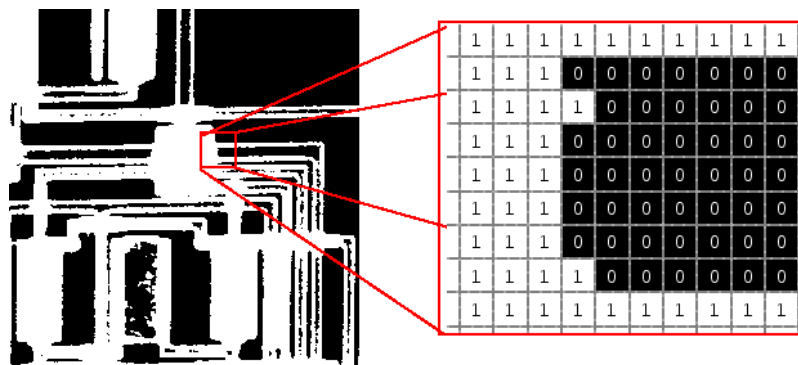
รูปที่ 2.4 ภาพระดับเทา

ที่มา: <https://yapayus.wordpress.com/>

## 2.2.5 ภาพไบนารี (Binary image)

ภาพไบนารี หรือ ภาพขาว-ดำ เป็นรูปที่ใช้เนื้อที่เพียง 1 บิตต่อพิกเซล ในทางดิจิทัล หมายความว่า มีเพียง 2 สถานะคือ 0 และ 1 โดยพิกเซลที่มีค่าสีเท่ากับ 0 พิกเซลนั้นจะแสดงสีดำ และพิกเซลที่มีค่าสีเท่ากับ 1 พิกเซลนั้นจะแสดงสีขาว

ซึ่งการแปลงภาพเกรย์สเกลเป็นภาพไบนารีนั้นจะต้องกำหนดค่าเทรชโฮล (Threshold Value) ซึ่งค่านี้จะถูกกำหนดโดยผู้ใช้หรือการใช้อัลกอริทึมในการหาค่าเทรชโฮลโดยอัตโนมัติก็ได้ หากพิกเซลมีค่าต่ำกว่าค่าเทรชโฮลที่กำหนดก็จะมีสถานะเป็น 0 แต่หากพิกเซลใดมีค่าความเข้มสีเทา มากกว่าค่าเทรชโฮลก็จะให้สถานะเป็น 1 [15]



รูปที่ 2.5 ตัวอย่างภาพไบนารีแสดงความสัมพันธ์ระหว่างค่าและสีของพิกเซล  
ที่มา: [http://www.geocities.ws/me\\_b02/notes/labcv5/lab5.html](http://www.geocities.ws/me_b02/notes/labcv5/lab5.html)

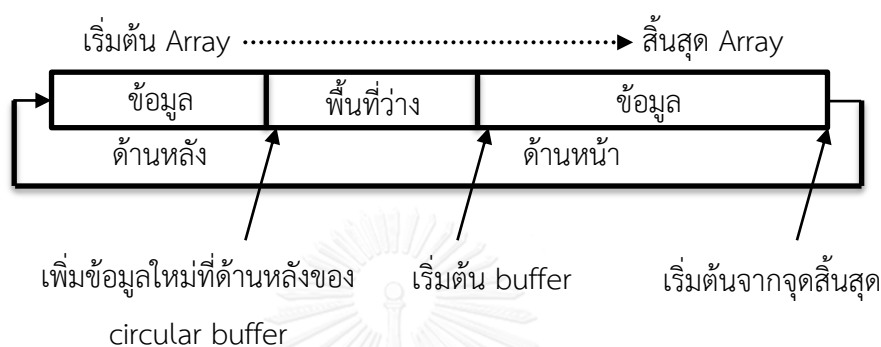
## 2.2 การประมวลผลเบื้องต้น (Pre-processing)

### 2.2.1 หลักการรับภาพ (Image acquisition)

การรับภาพจากกล้องที่ใช้บันทึกภาพคือการนำภาพเข้าระบบ โดยอาจจะมีการประมวลผลในขั้นตอนแรกๆ ของการทำงาน (preprocessing) เพื่อปรับคุณภาพภาพให้ดีขึ้น และลดสัญญาณรบกวน ทั้งนี้เฟรมที่มีการเคลื่อนไหวปะติดปะต่อกันเป็นจำนวนมากจนมีอัตราจำนวนเฟรมต่อวินาทีที่สูง จำเป็นจะต้องมีการแยกแต่ละเฟรมเพื่อมาประมวลผล ซึ่งเฟรมที่ได้จะมีความคมชัดมากขึ้นขึ้นอยู่กับชนิดของกล้องและคุณภาพของเลนส์ การใช้วิธีการปรับแต่งภาพ (Image Enhancement) ให้มีความคมชัดมากขึ้นจึงมีความจำเป็นสำหรับการเพิ่มประสิทธิภาพในการประมวลผลดังอธิบายในหัวข้อถัดไป

ทั้งนี้การเก็บข้อมูลภาพลงหน่วยความจำสามารถทำได้โดยการจองหน่วยความจำไว้ในรูปของตัวแปรอะเรย์ (array) ในหน่วยความจำของตนเองคือ Frame Buffer โดยค่าในแต่ละช่องของอะเรย์แสดงถึงคุณสมบัติของจุดภาพ (pixel) และตำแหน่งของช่องอะเรย์เป็นตัวกำหนดตำแหน่งของจุดภาพ กระบวนการรับภาพนี้แสดงในรูปที่ 2.6 สำหรับอัตราความเร็วในการแสดงผลหรือการแสดงผลเฟรมต่อ

วินาที (fps) จะมีความแตกต่างกันออกไป วิธีการคือเมื่อมีข้อมูลภาพเคลื่อนไหวแล้ว ระบบจะนำข้อมูลทั้งหมดมาเก็บในอาร์เรย์สามมิติ (HxWxS) โดย H หมายถึงลำดับแถว L หมายถึงคอลัมน์และ S หมายถึงลำดับของเฟรมข้อมูลภาพเคลื่อนไหว สมมติให้อาร์เรย์ A คือที่เก็บเฟรมข้อมูลภาพเคลื่อนไหวทั้งหมด ดังนั้นเมื่อต้องการเรียกภาพเฟรมที่ 34 ก็ทำได้โดยเรียกใช้ A(:, :, 34) ถ้าต้องการเข้าถึงจุดภาพแถวที่ 15 คอลัมน์ที่ 30 ของเฟรมที่ 18 ก็ทำได้โดยเรียก A (15,30,18)



รูปที่ 2.6 กระบวนการรับภาพ

## 2.2.2 หลักการปรับแต่งภาพ (Image enhancement)

การปรับปรุงคุณภาพภาพ (Image enhancement) คือการแปลงข้อมูลภาพเพื่อที่จะเน้นรายละเอียดในภาพตามที่ต้องการ หรือปรับโทนแสงที่ต้องการของภาพซึ่งบริเวณที่ปรับแต่งแล้วจะแตกต่างกับรายละเอียดอื่น ๆ ของภาพ เมื่อทำการปรับภาพจะมีการเน้นสารสนเทศในข้อมูลบางส่วน และอาจจะไปกดสารสนเทศในข้อมูลอีกส่วนหนึ่ง วิธีการปรับปรุงภาพสามารถดำเนินการได้หลายครั้งในภาพ ๆ หนึ่ง ซึ่งการปรับปรุงภาพในแต่ละวิธีจะช่วยให้การแปลงภาพในแง่มุมมองที่ต่างกัน ผู้ใช้ควรจะต้องรู้ลักษณะหลักการทำงาน และรู้จักผลที่ได้จากการใช้วิธีต่าง ๆ เพราะมีฉะนั้นจะทำให้การแปลความหมายผิดไป นอกจากนี้เมื่อมีการเปลี่ยนวิธีในการปรับภาพ ผู้ใช้จะต้องเรียนรู้ความหมายของสีที่แสดงออกมา การปรับปรุงภาพโดยการเน้นบางส่วน และลดบางส่วน จะช่วยให้ผู้ใช้สามารถสกัดสารสนเทศที่สำคัญได้อย่างประหยัด ถูกต้อง และแม่นยำ

วิธีที่นิยมใช้ในการปรับปรุงภาพ คือการทำภาพสีผสม (color composition) และการแปลงสี (color transformation) การทำภาพสีผสมมีความสำคัญต่อการแปลงภาพด้วยสายตา เนื่องจากตามนุษย์สามารถแยกภาพสีได้มากกว่าโทนสีเทา การทำภาพสีผสมเป็นพื้นฐานของการผสมแสงของแม่สี 3 สี การอธิบายคุณสมบัติของสีของวัตถุหนึ่งในภาพ เราอาจใช้สัดส่วนองค์ประกอบแดง เขียว น้ำเงิน หรือใช้ในรูปของ hue intensity และ saturation (H/I/S) ซึ่งอธิบายถึงความรู้สึกทางสี ความสว่าง และความบริสุทธิ์ของสี ตามลำดับ การแปลงสีอาจช่วยในการควบคุมการปรับปรุงภาพให้ดีขึ้น

ตัวอย่างเช่น การเปลี่ยนความอิ่มของสี (Saturation) ความอิ่มของสี คือความเด่นชัดของสีในภาพ หากลดค่าความอิ่มของสีลง โดยลดค่าความอิ่มของสีให้ค่อนข้างน้อยในโทนของภาพขาวดำ

### 2.2.3 หลักการการปรับภาพเรียบ

การกรองข้อมูลภาพ (Image Filtering) คือการใช้ตัวกรองสัญญาณกับภาพซึ่งจะได้ภาพที่มีคุณภาพที่ดีขึ้นและแตกต่างจากภาพต้นฉบับ โดยเป้าหมายหลัก คือการเพิ่ม (enhance) หรือลด (attenuate) คุณลักษณะบางอย่างในภาพ ซึ่งส่วนใหญ่อาศัยหลักการหาค่าเฉลี่ย โดยอาจเป็นการหาค่าเฉลี่ยของจุดเดียวกันจากภาพหลาย ๆ ภาพ หรือจากจุดต่าง ๆ ที่อยู่รอบๆ จุดที่สนใจ แต่การหาค่าเฉลี่ยเป็นการลดการเปลี่ยนแปลงของข้อมูล จึงใช้ได้ดีกับการกำจัดสัญญาณรบกวนที่เป็นสัญญาณความถี่สูง ซึ่งหากต้องการเน้นการเปลี่ยนแปลงของระดับความเข้มของจุดต่าง ๆ ภายในภาพให้เด่นชัดขึ้น จะต้องทำการกรองสัญญาณความถี่สูงผ่าน ดังนั้นวิธีที่สามารถนำมาใช้ได้ คือการคอนโวลูชัน (convolution)

การปรับภาพเรียบสามารถทำได้โดยการ convolution เหมเพลตที่สร้างขึ้นกับภาพด้วยตัวกรองความถี่ต่ำ (a low-pass filter kernel) โดยที่ความถี่สูงอย่างสัญญาณรบกวนหรือจุดเล็ก ๆ ที่ไม่ต้องการจะถูกกรองออกจากภาพไป การคอนโวลูชัน (Convolution) ระหว่างเหมเพลต (template) กับภาพ (image) สามารถทำได้โดยกำหนดเหมเพลตเป็นเมตริกซ์ขนาด  $n \times m$  ของชุดตัวเลขที่จะนำไปซ้อนทับภาพที่ตำแหน่งต่างๆ เพื่อหาผลลัพธ์ของการคอนโวลูชัน (Convolution) ถ้ากำหนดให้เหมเพลต  $T(x,y)$  เป็นเหมเพลตขนาด  $n \times m$  และภาพ  $I(X,Y)$  มีขนาด  $N \times M$  การคอนโวลูชันระหว่างเหมเพลตกับภาพสามารถแสดงได้ดังสมการต่อไปนี้

$$I'(X,Y) = T * I = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} T(i,j) \cdot I(X-i, Y-j) \quad (2.2)$$

โดย  $I'(X,Y)$  คือภาพผลลัพธ์จากการคอนโวลูชัน จากสมการที่ 2.2 จะเห็นว่าระดับความเข้มแสง ณ จุด  $(X,Y)$  ในภาพผลลัพธ์ได้จากการหาผลรวมของผลคูณของระหว่างค่าในเหมเพลตกับค่าระดับความเข้มแสงของภาพในบริเวณที่เหมเพลตซ้อนทับอยู่ จากสมการ ตัวชี้ตำแหน่งจุดในภาพ  $(X-i, Y-j)$  แสดงให้เห็นว่ามีการพลิกเหมเพลตทางแกนนอน และแกนตั้ง

สำหรับวิธีที่นิยมใช้ในการประมวลผลภาพคือการคอนโวลูชันที่ไม่ต้องมีการพลิกเหมเพลต หรือเรียกว่า cross-correlation ดังแสดงในสมการที่ 2.3

$$I'(X,Y) = T * I = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} T(i,j) \cdot I(X+i, Y+j) \quad (2.3)$$

ขั้นตอนของการคอนโวลูชันประกอบด้วยการเลื่อน บวก และคูณ ซึ่งสามารถใช้การคอนโวลูชัน (Convolution) ในการประมวลผลภาพได้ในหลายลักษณะ เช่น กรองสัญญาณภาพ การหาขอบภาพ (edge detection) หรือการหารูปทรงของวัตถุในภาพ เป็นต้น สำหรับวิธีการที่ใช้ในการปรับภาพให้มีความเรียบนั้นมีหลายวิธี อาทิ ตัวกรองแบบค่าเฉลี่ย (Average filter) ตัวกรองแบบเกาส์เซียน (Gaussian filter) ตัวกรองแบบค่ามัธยฐาน (Median filter) และ ตัวกรองแบบสองฝ่าย (Bilateral filter) ทั้งนี้พบว่าตัวกรองเกาส์เซียนมีการประมวลผลที่รวดเร็วและให้ผลลัพธ์ที่ดีที่สุด

#### 2.2.4 การเปลี่ยนแปลงลักษณะรูปร่างหรือโครงร่างของภาพ (Morphology operations)

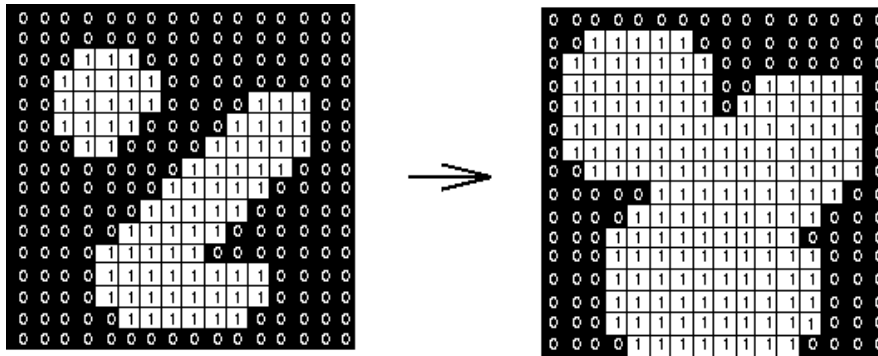
Morphology operations คือกระบวนการเปลี่ยนแปลงลักษณะรูปร่างหรือโครงร่างของภาพ โดยเกี่ยวกับการประมวลผลของภาพทางด้านโครงสร้าง การแยกส่วนประกอบของภาพออกเพื่อใช้ในการแสดงรูปร่าง ในเมทริกซ์จะประกอบไปด้วยค่าระดับขาว-ดำ 2 ค่า คือ 0 และ 1 กระบวนการการเปลี่ยนแปลงลักษณะรูปร่างประกอบไปด้วย 2 วิธีการหลักๆ คือ การขยาย (Dilation) และการกร่อนขนาด (Erosion) ดังนี้

การขยาย (Dilation) ทำได้โดยการกำหนดส่วนประกอบโครงสร้าง (Structuring element) และนำส่วนประกอบโครงสร้างไปกราดบนข้อมูลภาพตามลำดับตลอดทั้งภาพ โดยเมื่อจุดเริ่มต้นของส่วนประกอบโครงสร้างหรือจุดกำเนิดตรงกับตำแหน่งข้อมูลภาพที่เท่ากับ 1 จะทำการยูเนียนส่วนประกอบโครงสร้างเข้ากับข้อมูลภาพ นิยามของกระบวนการการขยายแสดงในสมการที่ 2.4

$$I \oplus H = \{(p+q) \mid p \in I, q \in H\} \quad (2.4)$$

เมื่อให้ส่วนประกอบโครงสร้าง  $H$  ทำการดำเนินการของเซตบนภาพ  $I$  ซึ่งกระบวนการขยายจะยังคงรักษาเส้นขอบต้นฉบับของมือไว้ แต่จะทำการขยายเส้นขอบของมือซึ่งเป็นบริเวณฉากหน้า (Foreground) ออกจากฉากหลัง (Background) ในภาพ สำหรับตัวอย่างการกำหนดส่วนประกอบโครงสร้าง (Structuring element) ขนาด  $3 \times 3$  ในการขยายแสดงในรูปที่ 2.7





รูปที่ 2.7 กระบวนการการขยายขนาดด้วยส่วนประกอบโครงสร้างขนาด 3x3

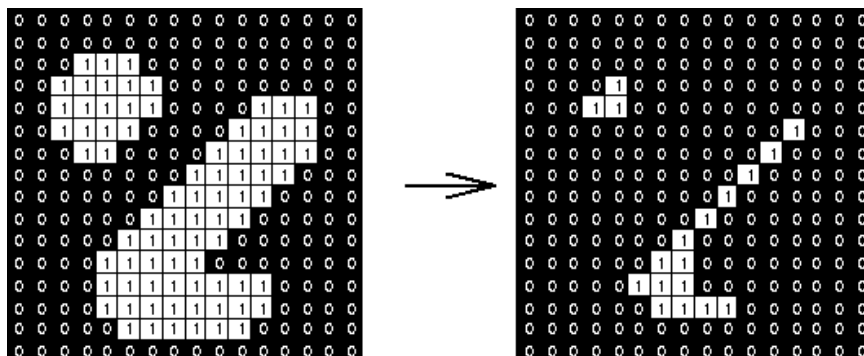
ที่มา: [www.cs.princeton.edu/~pshilane/class/mosaic/](http://www.cs.princeton.edu/~pshilane/class/mosaic/)

การกร่อนขนาด (Erosion) เป็นการกร่อนขนาดบริเวณขอบของวัตถุ กล่าวคือระบบจะทำการลบเส้นขอบของมือให้กลายเป็นค่า 0 ซึ่งการกร่อนมีวิธีคล้ายกับการขยาย คือ สร้างส่วนประกอบโครงร่างขึ้นมาแล้วนำไปกราดตามข้อมูลภาพ โดยจะเลื่อนไปทุกตำแหน่งเปรียบเทียบกับข้อมูลภาพ ถ้าข้อมูลมีค่าเหมือนกับส่วนประกอบโครงร่างจะทำการกำหนดค่าข้อมูลภาพที่ตรงกับตำแหน่งที่ตรงกับจุดเริ่มต้นหรือ จุดกำเนิดของส่วนประกอบโครงร่างให้เท่ากับ 1

นิยามของกระบวนการการกร่อนขนาดแสดงในสมการที่ 2.5

$$I \ominus H = \{p \in Z^2 \mid (p+q) \in I, \text{ สำหรับทุก } q \in H\} \quad (2.5)$$

เมื่อให้ส่วนประกอบโครงร่าง H ทำการดำเนินการของเซตบนภาพ I สำหรับตัวอย่างการกำหนดส่วนประกอบโครงร่าง (Structuring element) ขนาด 3x3 ในการกร่อนขนาดแสดงในรูปที่ 2.8

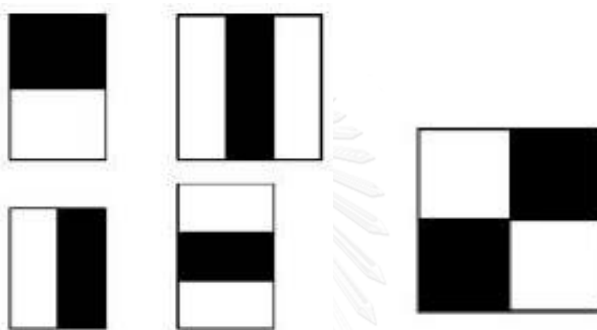


รูปที่ 2.8 กระบวนการการกร่อนขนาดด้วยส่วนประกอบโครงสร้างขนาด 3x3

ที่มา: [www.cs.princeton.edu/~pshilane/class/mosaic/](http://www.cs.princeton.edu/~pshilane/class/mosaic/)

## 2.2.5 การลบใบหน้า (Face subtraction)

การตรวจจับใบหน้าเป็นหนึ่งในขั้นตอนที่ช่วยทำให้ผลลัพธ์ดีขึ้น เพราะจะช่วยให้ระบบสามารถแยกแยะบริเวณใบหน้าและมือได้มากขึ้น โดยเฉพาะกรณีที่มีมืออยู่บริเวณด้านหน้าของใบหน้า ซึ่งในกรณีนี้จะใช้เทคนิคการตรวจจับสีผิวเพียงอย่างเดียวไม่ได้ ระบบจำเป็นต้องลบส่วนของใบหน้าออกไปก่อนเพื่อที่จะไม่ให้เกิดความสับสนกับท่ามือ โดยทั่วไปแล้วตัวจำแนกใบหน้าจะประกอบไปด้วยวิธีการพื้นฐาน 2 วิธี อาทิ ต้นไม้ตัดสินใจ (Decision-tree) ที่มีใบอย่างน้อยสองใบ และ Haar-like feature ซึ่งการใช้วิธี Haar-like feature [16] ถือเป็นหนึ่งทางเลือกที่ใช้สำหรับจำแนกรูปร่าง ตำแหน่ง และขนาด ลักษณะที่ใช้ในการจำแนกถูกออกแบบให้มีลักษณะเป็นรูปทรงสี่เหลี่ยม ดังแสดงในรูปที่ 2.9



(ก) คุณลักษณะขอบ (ข) คุณลักษณะเส้น (ค) คุณลักษณะสี่เหลี่ยมผืนผ้า

รูปที่ 2.9 แสดงรูปแบบของวิธี Haar-like features [16]

จากรูปที่ 2.9 พบว่า (ก) แสดงคุณลักษณะขอบ (Edge features) (ข) แสดงคุณลักษณะเส้น (Line features) และ (ค) แสดงคุณลักษณะสี่เหลี่ยมผืนผ้า (Four-rectangle features) โดยค่าเฉลี่ยของพิกเซลในบริเวณพื้นที่สีดำได้มาจากการลบกันของค่าเฉลี่ยของพิกเซลในบริเวณพื้นที่สีขาว ทั้งนี้คุณลักษณะจะยังคงอยู่เมื่อค่าความแตกต่างมากกว่าค่าเทรชโฮลด์ที่ตั้งไว้ ซึ่งคุณลักษณะหลักๆ สำหรับการตรวจจับใบหน้า ได้แก่ ตา แก้ม หน้าผาก และจมูก โดยจะแบ่งเป็นบริเวณที่มีสีเข้ม เช่น ตา คิ้ว และเงา และบริเวณที่มีสีเข้มอ่อน เช่น แก้ม หน้าผาก และจมูก

จากลักษณะดังกล่าวทำให้การใช้ Haar-like feature ได้รับความนิยมเป็นอย่างมากสำหรับการตรวจจับใบหน้าโดยใช้ร่วมกับตัวจำแนกแบบ cascade ใช้ไฟล์ XML ที่ประกอบไปด้วยตัวอย่างใบหน้าเป็นจำนวนมากจึงทำให้ง่ายต่อการตรวจจับใบหน้ามนุษย์ที่ได้รับจากกล้อง ไฟล์ XML นี้ผ่านการเทรนดักกลุ่มตัวอย่างใบหน้ามนุษย์หรือเรียกว่า Positive examples และภาพอื่นที่ไม่ใช่ใบหน้ามนุษย์หรือเรียกว่า Negative examples เพื่อให้เกิดความแม่นยำในการตรวจจับและจำแนกใบหน้าในเฟรมที่มีขนาดและสภาพแวดล้อมแตกต่างกันได้

## 2.3 การตรวจจับมือ (Hand detection)

### 2.3.1 การลบพื้นหลัง (Background subtraction)

การลบพื้นหลังหรือเรียกว่าการตรวจจับฉากหน้า (Foreground detection) คือการหาตำแหน่งของวัตถุบนภาพนั้น ๆ เพื่อที่จะได้ทราบตำแหน่งที่อยู่ของวัตถุบนภาพ วิธีการนี้นำมาใช้อย่างแพร่หลายในด้านการติดตามเส้นระยะวิ่ง หรือการประเมินท่าทางมนุษย์ ในการลบภาพพื้นหลัง จะต้องมีการเก็บข้อมูลของภาพพื้นหลังไว้ก่อนเพื่อนำมาเปรียบเทียบกับลำดับภาพอื่นๆ ซึ่งภาพพื้นหลังจะต้องเป็นภาพนิ่งรวมถึงกล้องที่ใช้ในการรับภาพก็ต้องนิ่งด้วย โดยมากแล้วจะใช้กับการภาพที่มาจากวิดีโอเฟรม เพื่อหาวัตถุที่มีการเคลื่อนที่ในวิดีโอั้น ๆ [17]

หลักการของวิธีการตรวจจับการเคลื่อนไหวพื้นฐาน ใช้วิธีการหาความแตกต่างระหว่างเฟรมทำได้โดยการตั้งค่าเทรชโฮลด์ เพื่อหาความแตกต่างระหว่างเฟรมภาพเก่าหรือเฟรมภาพอ้างอิงและเฟรมภาพใหม่ โดยในการหาความแตกต่างของจุดภาพนั้น (Pixel difference) นั้นเฟรมภาพแต่ละภาพจะถูกนำไปเปรียบเทียบกับภาพอ้างอิงทุกจุดภาพ หากความเข้มของพิกเซลนั้นมีค่ามากกว่าหรือเท่ากับค่าเทรชโฮลด์ กำหนดให้พิกเซลนั้นเป็นสีขาวหรือมีค่าเป็น 1 นั่นคือสามารถแยกวัตถุออกมาจากพื้นหลังได้ ในทางตรงกันข้ามถ้าหากความเข้มของพิกเซลนั้นมีค่าน้อยกว่าค่าเทรชโฮลด์ กำหนดให้พิกเซลนั้นเป็นสีดำหรือมีค่าเป็น 0 นั่นคือไม่สามารถแยกวัตถุออกจากพื้นหลังได้ อธิบายได้ดังสมการนี้

$$\| \text{frame}_i - \text{frame}_{i+1} \| > th \quad (2.6)$$

เมื่อ  $\text{frame}_i$  คือค่าพิกเซลในภาพที่เวลา  $i$

$\text{frame}_{i+1}$  คือค่าพิกเซลในภาพที่เวลา  $i+1$

$th$  คือค่าเทรชโฮลด์ในการจำแนกวัตถุที่เคลื่อนไหวออกจากฉากหลัง

### 2.3.2 การตรวจจับสีผิว (Skin color detection)

การตรวจจับสีผิว (Skin color detection) เป็นวิธีการที่ได้รับความนิยมและมีการใช้อย่างแพร่หลาย อาทิ การตรวจจับหน้า การตรวจจับและติดตามมือ เป็นต้น หากพิจารณาการจำแนกประเภทของการตรวจจับสีผิว พบว่าสามารถแบ่งได้เป็น 2 ประเภท นั่นคือ การจำแนกพิกเซลสีผิว (Skin-pixel) และ การจำแนกพิกเซลสีอื่น (Nonskin-pixel) ทั้งนี้เนื่องจากสีผิวของมนุษย์แต่ละกลุ่มจะมีความแตกต่างกันในพื้นที่ขนาดเล็กภายในโมเดลสีเมื่ออยู่ในสภาพแวดล้อมที่ควบคุมด้วยแสงสว่าง วิธีที่ใช้ในการจำแนกที่ดีจะต้องสามารถจำแนกหลากหลายลักษณะสีผิว อาทิ ผิวโทนเหลือง น้ำตาล ชมพู และขาว เป็นต้น ดังนั้นหนึ่งในวิธีที่ง่าย รวดเร็ว และได้รับความนิยมมากที่สุด คือการกำหนด

ขอบเขตการตัดสินใจของกลุ่มสีผิวที่แตกต่างกันสำหรับองค์ประกอบโมเดลสี ทำได้โดยกำหนดช่วงของค่าเทรซโฮลด์ช่วงหนึ่ง หรือหลายช่วงสำหรับแต่ละองค์ประกอบของโมเดลสี ค่าพิกเซลที่อยู่ในช่วงที่กำหนดจะกลายเป็นส่วนประกอบสีทั้งหมดของพิกเซลสีผิว [18]

ทั้งนี้เพื่อที่จะได้รับผลลัพธ์ที่ดี การตั้งช่วงค่าเทรซโฮลด์จะต้องมีความเหมาะสม ดังนั้นจึงต้องคำนึงถึงการประมวลผลบนประเภทและองค์ประกอบของโมเดลสี ซึ่งการแยกองค์ประกอบของแสง (Luminance) ออกจากองค์ประกอบสี (Chrominance) สามารถช่วยให้ได้ค่าความถูกต้องในการตรวจจับสีผิวมากขึ้น ดังนั้นภาพต้นฉบับโมเดลสี RGB ที่ได้รับจึงต้องได้รับการแปลงเป็นโมเดลสีประเภทอื่นก่อนที่จะทำการกำหนดเทรซโฮลด์ ข้อดีของการใช้เทรซโฮลด์ คือเป็นวิธีที่ให้ความรวดเร็วและไม่ซับซ้อนซึ่งเหมาะกับการประมวลผลแบบทันที (Real time)

### 2.3.3 การตรวจจับเส้นขอบ (Contour detection)

เส้นขอบ (Contour) หมายถึงเส้นโค้งที่เชื่อมต่อพิกเซลที่มีค่าความเข้มเท่ากัน ในหนึ่งภาพเป็นไปได้ว่าจะมีเส้นขอบมากกว่า 1 เส้น เส้นขอบจะแสดงถึงความลาดชันโดยการวาดกรอบรอบ ๆ รูปร่างมือซึ่งกำหนดโดยค่าเทรซโฮลด์ โดยที่กรอบอาจจะเกิดขึ้นมาหลายขนาดขึ้นอยู่กับสัญญาณรบกวนในภาพ หากมีสัญญาณรบกวนน้อยกรอบจะแสดงเส้นขอบขนาดใหญ่ ซึ่งเส้นขอบขนาดใหญ่จะถูกใช้สำหรับการตรวจจับท่ามื่อดังเส้นสีเขียวที่แสดงในรูปที่ 2.10 ที่เป็นเส้นขอบที่มีขนาดใหญ่ที่สุดของมือ

เพื่อหาลักษณะของเส้นโค้งของวัตถุที่มีรูปร่างปิด จึงใช้วิธีการหาความยาวส่วนโค้ง (Arc length) ร่วมกับการประมาณรูปหลายเหลี่ยม (Polygon approximation) โดยวิธีการหาความยาวส่วนโค้ง (Arc length) จะทำการคำนวณหาเส้นต่อเนื่องที่สร้างขอบเขตของรูปทรงเรขาคณิตแบบปิดหรือความยาวของเส้นโค้ง การทำงานของวิธีการหาความยาวส่วนโค้งแสดงดังรูปที่ 2.10



(ก)

(ข)

(ค)

รูปที่ 2.10 วิธีการหาความยาวส่วนโค้ง

ที่มา: [http://docs.opencv.org/3.1.0/dd/d49/tutorial\\_py\\_contour\\_features.html](http://docs.opencv.org/3.1.0/dd/d49/tutorial_py_contour_features.html)

จากรูปที่ 2.10 (ข) จะเห็นว่าเส้นสีเขียวแสดงเส้นโค้งที่ได้รับการประมาณค่าสำหรับ epsilon เท่ากับ 10% ของความยาวส่วนโค้ง ในขณะที่รูป (ค) แสดง epsilon เท่ากับ 1% ของความยาวส่วนโค้งและแสดงเส้นที่เข้ามาแบบเส้นโค้งปิด สำหรับอัลกอริทึมการประมาณรูปหลายเหลี่ยม (Polygon approximation) จะลดจำนวนจุดต่าง ๆ ของเส้นขอบ ช่วยปรับปรุงคุณภาพภาพให้เรียบเนียนขึ้น จากสัญญาณรบกวนที่ได้จากแยกขอบของวัตถุ ทั้งนี้รูปหลายเหลี่ยมที่พอดีกับเส้นขอบของวัตถุสามารถใช้ในการประมาณคุณลักษณะที่อธิบายรูปร่างของวัตถุได้ อาทิ การประมาณรูปร่างเส้นโค้งที่มีรูปหลายเหลี่ยมอื่น ๆ เกิดขึ้นและมีจุดยอดมนน้อยกว่าเส้นโค้ง [19]

## 2.4 การติดตามมือ (Hand tracking)

### 2.4.1 การติดตามวัตถุ (Multi-Object tracking)

การติดตามวัตถุมีความสำคัญต่อการติดตามการเคลื่อนที่ของวัตถุที่สนใจ หลักการของกระบวนการติดตามวัตถุคือการติดตามหลายวัตถุ (Multi-object tracking: MOT) ที่จัดเป็นวิธีการทดลองสำหรับระบุตำแหน่งวัตถุที่มีการเคลื่อนที่ วิธีการนี้ได้รับการพัฒนาและนำมาประยุกต์ใช้ในแอปพลิเคชันต่าง ๆ อาทิ การเฝ้าระวังวิดีโอ การติดตามการจราจร และการจำแนกวัตถุในภาพ อย่างไรก็ตามวิธีนี้ยังมีปัญหาในเรื่องของการพัฒนาความถูกต้องสำหรับภาพที่มีความซับซ้อน ดังนั้นเพื่อแก้ไขปัญหานี้และพัฒนาวิธีนี้ติดตามวัตถุให้สามารถใช้ได้ในสถานการณ์จริง จึงมีการใช้วิธีการอื่น ๆ เข้ามาช่วย ทั้งนี้กระบวนการการตรวจจับการเคลื่อนที่ไหวแบ่งออกได้เป็น 2 ประเภทหลัก ๆ ได้แก่ ประเภทการใช้วิธีการลบพื้นหลัง ซึ่งใช้สำหรับการตรวจจับการเคลื่อนที่ของวัตถุในสภาพแวดล้อมที่ฉากหลังไม่เปลี่ยนแปลง และประเภทการใช้วิธี Optical Flow ในการหาความเปลี่ยนแปลงระหว่างเฟรมภาพ 2 เฟรม สำหรับการคำนวณการเคลื่อนที่ที่ส่วนแสงสว่างนั้นจะป้องกันการสูญเสียค่าความเข้มของพิกเซล วิธีการนี้สามารถใช้ได้กับกล้องวิดีโอที่เคลื่อนที่ได้ นอกจากนี้ยังสามารถใช้การติดตามคุณลักษณะโดยพิจารณาจากลักษณะรูปร่างและสีสำหรับตรวจจับวัตถุร่วมในกระบวนการนี้ได้ตั้งอธิบายในหัวข้อการติดตามวัตถุที่สนใจ

#### 2.4.1.1 การไหลของแสง (Optical flow)

ออฟติคอลลอว์เป็นวิธีการที่ใช้สำหรับการตรวจจับวัตถุที่เคลื่อนที่ภายในภาพ มีความทนทานต่อการเปลี่ยนแปลงของฉากหลังและการเคลื่อนที่ของกล้องเนื่องจากมีการใช้สมการเชิงอนุพันธ์ระหว่างเฟรมภาพในกระบวนการตรวจจับ ที่แสดงถึงตำแหน่งและเวลาในรูปแบบอนุกรมเทเลอร์ (Taylor Series) จากการศึกษาพบว่า Horn และ Schunck [20] ใช้การประมาณออฟติคอลลอว์ด้วยสมมุติฐานว่าช่วงเวลาสั้น ๆ ระหว่างเฟรมไม่ส่งผลต่อภาพความเข้มระหว่างเฟรมมากนัก

ข้อดีของวิธีนี้คืออำนวยความสะดวกรวดเร็วต่อการประมวลผลของระบบ ความซับซ้อนน้อย ให้ค่าความถูกต้องที่แม่นยำ เป็นต้น

## 2.4.2 การติดตามวัตถุที่สนใจ (Interested object tracking)

กระบวนการนี้จะนำวัตถุเคลื่อนที่ที่ตรวจจับได้มาวิเคราะห์เพื่อหาคุณลักษณะที่สนใจ อาทิ รูปร่าง สี ตำแหน่ง ความลึก เป็นต้น วิธีการที่ใช้ในกระบวนการนี้มีหลายวิธี อาทิ การวิเคราะห์กลุ่มของจุดภาพไบนารี หรือ การใช้ Mean shift โดยวิธีที่ได้รับความนิยม คือ วิธีการติดตามโดยใช้ Blob นั้นเอง โดยวิธีนี้จะทำการวิเคราะห์กลุ่มของจุดภาพขาวดำ หรือภาพไบนารี โดยการแยกกลุ่มของจุดภาพที่ต่างกันจากการคำนวณหาขนาดและตำแหน่งของวัตถุ จากนั้นสร้างบล็อบสี่เหลี่ยมล้อมรอบวัตถุที่สนใจ ซึ่งจากบล็อบจะทำให้ทราบขนาด ความกว้าง ความยาว และจุดกึ่งกลางของวัตถุ

### 2.4.2.1. การวิเคราะห์บล็อบ (Blob analysis)

การวิเคราะห์บล็อบคือการคำนวณค่าเชิงสถิติสำหรับบริเวณที่กำหนดในภาพระดับเข้ม โดยจะดำเนินการเชิงปริมาณ อาทิ การหา centroid bounding box label matrix และ blob count เป้าหมายของวิธีการนี้คือเพื่อตรวจจับบริเวณที่เกี่ยวข้องในภาพที่ถูกลดขนาดของภาพเดียวกัน [21]

## 2.5 การวิเคราะห์ท่ามือ (Hand gesture recognition) โดยการตัดสินใจท่ามือ (Hand gesture decision)

### 2.5.1 การสกัดคอนเวกซ์ (Convex extraction)

การสกัดคอนเวกซ์ คือการสกัดคุณลักษณะที่ประกอบไปด้วย 2 คุณลักษณะ ได้แก่ คอนเวกซ์ฮัลล์ (Convex Hull) และจุดลึกสุดในรอยเว้า (Convexity Defect) ดังแสดงรายละเอียดดังนี้

#### 2.5.1.1 คอนเวกซ์ฮัลล์ (Convex hull)

คอนเวกซ์ฮัลล์ (Convex hull) หมายถึงเส้นขอบที่ใหญ่ที่สุดซึ่งไม่มีขอบเว้าหรือเรียกว่าเปลือกนูน คอนเวกซ์ฮัลล์จะแสดงเส้นขอบรอบๆรูปร่างมือโดยที่ไม่เว้าเข้าไปตามนิ้วมือ [22] คอนเวกซ์ฮัลล์สามารถแบ่งออกเป็น 3 แบบ ได้แก่ เส้น (lines) ส่วน (segments) และหลายเหลี่ยม (polygons)

เส้น (lines) สามารถแสดงเป็น  $l$  ที่ดำเนินการกับ (a,b,c) โดยที่ a, b, และ c คือค่าสัมประสิทธิ์ของสมการเส้นตรง ดังนี้

$$ax + by + c = 0 \quad (2.7)$$

นอกจากนี้สำหรับจุดสองจุดที่แตกต่างกัน อย่างเช่น  $q_1$  และ  $q_2$  สามารถเชื่อมต่อกันได้ด้วยเส้น  $l$  โดยการใช้ระบบพิกัดคาร์ทีเซียนกับ  $q_1$  และ  $q_2$  เช่น  $(x_1, y_1)$  และ  $(x_2, y_2)$  การคำนวณเส้น  $l$  ที่ผ่านจุด  $q_1$  และ  $q_2$  สามารถแสดงได้ดังสมการที่ 2.8

$$\frac{x-x_1}{x_2-x_1} = \frac{y-y_1}{y_2-y_1} \quad (2.8)$$

จะได้

$$a = (y_2 - y_1) \quad (2.9)$$

$$b = -(x_2 - x_1) \quad (2.10)$$

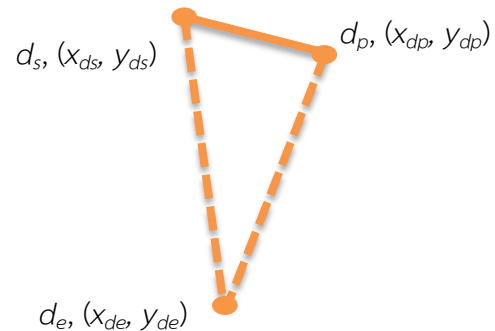
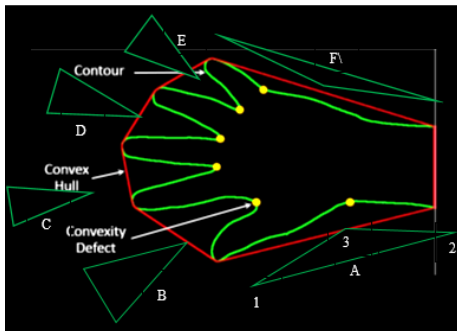
$$c = y_1(x_2 - x_1) - x_1(y_2 - y_1), \quad (2.11)$$

ในขณะที่ ส่วน (Segment:  $S$ ) มาจากคู่ของจุด  $(p, q)$  ในระนาบ  $(x, y)$  ที่สร้างจุดสิ้นสุด และให้เส้นผ่านจุดในช่วงของพิกัด  $xy$  ที่มีการจำกัดพื้นที่ไว้สำหรับส่วนนั้นๆ

รูปหลายเหลี่ยม (Polygon:  $P$ ) จะแสดงโดยการนำลำดับวงกลมของจุดเรียกว่าจุดยอดของ  $P$  โดยขอบของ  $P$  คือส่วนระหว่างจุดยอดหลายจุดที่ติดต่อกัน รูปหลายเหลี่ยมถือว่าเป็นเปลือกนูนเมื่อลักษณะของเส้นมีความเรียบ และมุมทุกมุมของเส้นมีค่าน้อยกว่า  $\pi$

### 2.5.1.2 จุดลึกสุดในรอยเว้า (Convexity defect)

จุดที่อยู่ลึกที่สุดในรอยเว้า (Convexity defect) หมายถึงจุดบนเส้นขอบที่อยู่ห่างจากคอนเวกซ์ฮัลล์มากที่สุดในแต่ละรอยเว้า จุดลึกสุดของเส้นจะถูกกำหนดไว้ภายในคอนเวกซ์ฮัลล์แต่ห่างออกมา เช่น เมื่อมีการชูนิ้วขึ้นมาสองนิ้ว จุดลึกสุดจะอยู่ตรงกลางระหว่างนิ้วมือทั้งสอง จุดลึกสุดในรอยเว้าแสดงดังจุดสีเหลืองในรูปที่ 2.11(ก) การหาจุดที่อยู่ลึกที่สุดในรอยเว้าสามารถทำได้โดยการจัดหมวดหมู่เวกเตอร์ พบว่าเวกเตอร์คุณสมบัติจะได้รับมาจากจุด 3 จุด โดยค่าสำหรับทุกจุดจะถูกสร้างเป็นเวกเตอร์ และสามารถใช้ในการระบุจุดพิกัดของเส้น [23] ประกอบไปด้วย จุดเริ่มต้น  $d_s, (x_{ds}, y_{ds})$  จุดสิ้นสุด  $d_e, (x_{de}, y_{de})$  และจุดตำแหน่งลึกสุด  $d_p, (x_{dp}, y_{dp})$  ดังแสดงในรูปที่ 2.11(ข) เมื่อลากเส้นผ่านจุดทั้งสามจุดจะได้เวกเตอร์แสดงดังหมายเลข 1 2 และ 3 ของรูปสามเหลี่ยมขนมเปียกปูน A ตามลำดับ จากรูปที่ 2.11 พบว่า (ก) แสดงเส้นขอบ คอนเวกซ์ฮัลล์ และจุดลึกสุดในรอยเว้ารูปสามเหลี่ยม 6 รูป ได้แก่ สามเหลี่ยม A B C D E และ F และ (ข) แสดงจุดพิกัดสามจุดในแต่ละรูปสามเหลี่ยม ได้แก่ จุดเริ่มต้น  $d_s, (x_{ds}, y_{ds})$  จุดสิ้นสุด  $d_e, (x_{de}, y_{de})$  และจุดตำแหน่งลึกสุด  $d_p, (x_{dp}, y_{dp})$  ดังแสดงในหมายเลข 1 2 และ 3 ตามลำดับ ดังนั้นรูปสามเหลี่ยมขนมเปียกปูน A สามารถอธิบายได้ว่าเป็นเวกเตอร์  $V_{td}^A$  ได้แก่  $[A(x_{ds}) A(y_{ds}) A(x_{dp}) A(y_{dp}) A(x_{de}) A(y_{de})]$



(ก) เส้นขอบ คอนเวกซ์ฮัลล์ และจุดลึกสุด (ข) จุดพิกัดในรูปสามเหลี่ยม  
รูปที่ 2.11 เส้นขอบ คอนเวกซ์ฮัลล์ และจุดที่อยู่ลึกที่สุดในรอยเว้า

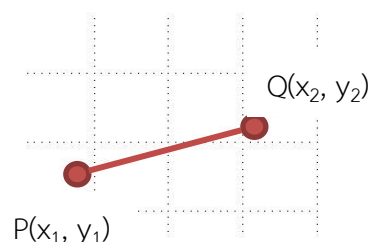
## 2.5.2 ระยะห่างระหว่างพิกเซลสองพิกเซล (Euclidean distance)

ในระบบพิกัดฉากและเวกเตอร์ ปริภูมิ (space) คือส่วนที่ไร้ขอบเขต เป็นปริมาณที่มีมิติในวัตถุและเหตุการณ์ และมีตำแหน่งสัมพันธ์และทิศทาง สำหรับนักคณิตศาสตร์หมายถึงโลกส่วนตัวที่สร้างขึ้นโดยสมาชิกที่อาศัยอยู่ในโลกนี้จะต้องมีคุณสมบัติตามกฎที่นักคณิตศาสตร์กำหนดขึ้นอย่างเคร่งครัด การศึกษาปริภูมิเวกเตอร์สามมิติจะใช้คู่อันดับ  $(x, y)$  แทนจุดในสองมิติ ระบบพิกัดฉากในปริภูมิสองมิติ ประกอบด้วยระนาบ 2 ระนาบ เรียกว่าระนาบอ้างอิงที่มีความตั้งฉากกัน ระนาบทั้งสองจะตัดกันได้ รอยตัดเป็นเส้นตรงสองเส้นเรียกว่าแกนพิกัด (Coordinate axes) [24]

เรียกจุดตัดของแกนทั้งสองว่า จุดกำเนิด (Origin) แทนด้วย  $O$  การกำหนดพิกัดของจุด เราจะเริ่มกำหนดทิศทางบวกทั้งสองแกน จะวัดจากจุดกำเนิดไปตามแต่ละแกนในทิศใดทิศหนึ่งในระบบมือขวา หรือระบบมือซ้าย โดยปกติจะกำหนดแกนโดยใช้ระบบมือขวา สำหรับปริภูมิ 2 มิติ ระยะทางระหว่างจุด  $P(x_1, y_1)$  และ  $Q(x_2, y_2)$  สามารถคำนวณได้จากสมการยูคลิด (Euclidean) ดังนี้

$$|PQ| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (2.12)$$

การหาระยะทางระหว่างจุดในปริภูมิ 2 มิติสามารถแสดงได้ดังรูปที่ 2.12



รูปที่ 2.12 การหาระยะทางระหว่างจุดในปริภูมิ 2 มิติ



### 2.5.3 การเรียนรู้ต้นไม้ตัดสินใจ (Decision tree learning)

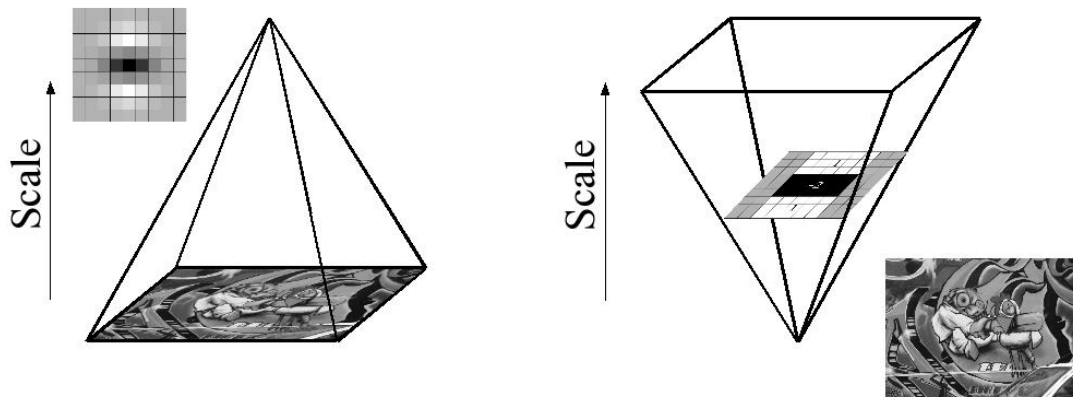
การเรียนรู้ต้นไม้ตัดสินใจ (decision tree learning) หรือ Classification tree คือส่วนหนึ่งของทฤษฎีการตัดสินใจ (decision theory) และทฤษฎีกราฟ จัดเป็นวิธีการพื้นฐานสำหรับการทำเหมืองข้อมูล ที่คิดค้นโดย Dome และ JA โดยทำการประมาณฟังก์ชันที่มีค่าไม่ต่อเนื่อง (discrete-value function) ด้วยแผนผังต้นไม้เพื่อให้มนุษย์สามารถเข้าใจการตัดสินใจของต้นไม้ได้ง่าย โดยจัดทำเซตของกฎต่างๆ อาทิ ถ้า-แล้ว (if-then) ซึ่งในทางการเรียนรู้ของเครื่อง (machine learning) นั้น วิธีนี้เป็นเทคนิคแบบ Supervised Learning ที่ไม่ซับซ้อนนัก โครงสร้างของต้นไม้จะแตกแขนงไปตามเงื่อนไขหรือข้อมูลที่ได้อาคาดคะเนไว้ว่าจะเกิดขึ้น ที่แบ่งออกเป็น 3 องค์ประกอบ ได้แก่ ใบ (Leaf) เป็นส่วนของผลลัพธ์ที่ได้ กิ่ง (Branch) เป็นตัวเชื่อมระหว่างโหนด และราก (Root) เป็นจุดเริ่มต้นของเหตุการณ์ [25]

### 2.5.4 SURF (Speeded-up robust features)

SURF หรือ Speeded-up robust features [26] มีหลักการพื้นฐานและกระบวนการเหมือนกับ SIFT (Scale-Invariant Feature Transform) แต่มีโครงสร้างที่แตกต่างกัน อีกทั้งยังให้ผลลัพธ์ที่ดี รวดเร็ว และทนทานต่อการแปลงภาพมากกว่า อัลกอริธึม SURF แบ่งออกเป็นสามองค์ประกอบหลัก ๆ ได้แก่ การตรวจจับจุดที่สนใจ (interest point detection) คำอธิบายพื้นที่ใกล้เคียง (local neighborhood description) และการจับคู่ (matching)

สำหรับการตรวจจับจุดที่สนใจนั้น (interest point detection) จะใช้ตัวกรองกล่องสี่เหลี่ยมขนาดใด ๆ ในการเบลอภาพแบบวิธีเกาส์เซียนร่วมกับภาพอินทิกรัลที่ทำให้การคำนวณผลรวมของพิกเซลง่ายและรวดเร็ว ซึ่งกระบวนการนี้มีเป้าหมายเพื่อการระบุตำแหน่งและขนาดต่าง ๆ ที่สามารถระบุได้จากมุมมองที่ต่างกันหลายมุมมองที่มาจากวัตถุอันเดียวกัน โดย SURF สามารถคำนวณได้โดยตรงจากตัวกรองภาพในแต่ละขนาดโดยไม่ต้องใช้ผลลัพธ์จากขนาดก่อนหน้า นอกจากนี้ในการกรองนั้น ตัวกรองมักจะถูกเพิ่มขนาดให้ใหญ่ขึ้นมากกว่าลดขนาดของภาพให้เล็กลง

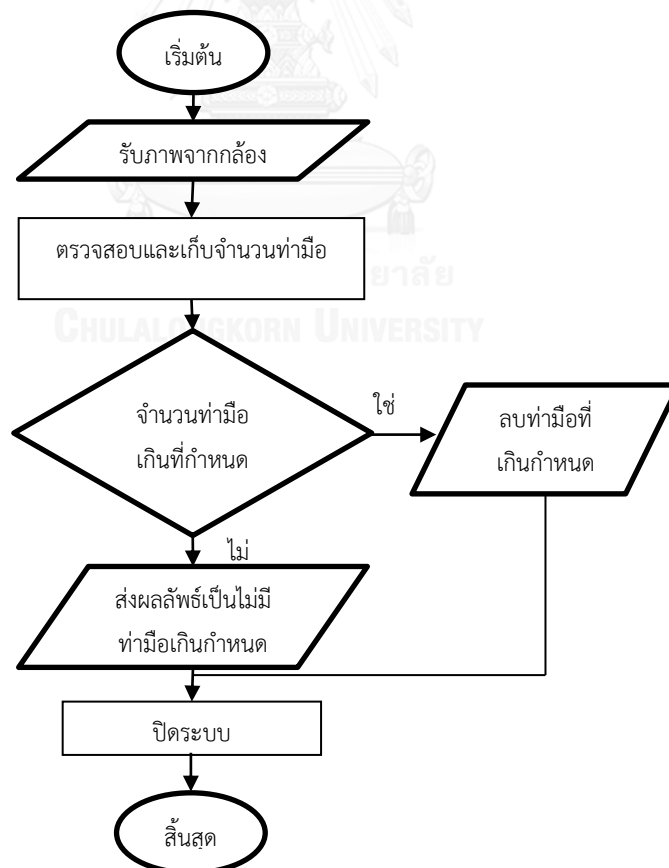
สำหรับการใช้วิธี hessian based blob detector นั้น เมทริกซ์ของมันจะแสดงให้เห็นการเปลี่ยนแปลงในพื้นที่รอบ ๆ บริเวณและขอบเขตของการตอบสนอง ทั้งนี้พบว่าจุดที่สนใจสามารถพบได้ในขนาดที่แตกต่างกันของภาพ ซึ่งภาพที่ได้รับการกรองแบบเกาส์เซียนและแบ่งกลุ่มตัวอย่างเพื่อให้ได้ระดับที่สูงขึ้นโดยพิจารณาเป็นพีรามิดดังแสดงในรูปที่ 2.13 จากความสำคัญของการใช้ภาพอินทิกรัลและตัวกรองสี่เหลี่ยมขนาดใด ๆ ลงบนภาพต้นฉบับในการวิเคราะห์เชิงพื้นที่ (Scale space analysis) ดังนั้นตัวกำหนด hessian based blob detector จึงได้รับความสำคัญมากกว่าการวัดความแตกต่างของขนาดและตำแหน่ง



รูปที่ 2.13 การวิเคราะห์เชิงพื้นที่โดยตัวกรองสี่เหลี่ยมกับภาพอินทิกรัล [26]

### 2.5.5 หลักการในการตัดสินใจท่ามือ (Hand gesture decision)

กระบวนการตัดสินใจจำเป็นต้องมีการสร้างเงื่อนไขเพื่อให้ระบบมีการตัดสินใจการจำแนกท่ามือได้อย่างมีประสิทธิภาพ การตัดสินใจของระบบสามารถแสดงได้ในผังงานแสดงการตัดสินใจรูปที่ 2.1



รูปที่ 2.14 ผังงานแสดงหลักการในการตัดสินใจท่ามือ

## 2.6 บอร์ด Raspberry Pi และการเชื่อมต่อระหว่างบอร์ดและสมาร์ทโฟน (Raspberry Pi board and its connection with smart phone)

Raspberry Pi เป็นคอมพิวเตอร์บอร์ดเดียวที่พัฒนาขึ้นเพื่อสนับสนุนและช่วยในการสอนการเขียนโปรแกรมและการคำนวณ นอกจากนี้ยังเป็นจุดเริ่มต้นในการพัฒนาโครงการ Internet of Things (IoT) บอร์ด Raspberry Pi สามารถประยุกต์ใช้งานได้หลากหลายไม่ว่าจะเป็นคอมพิวเตอร์ เซิร์ฟเวอร์ หรืออุปกรณ์เฝ้าระวัง บอร์ดนี้ทำงานบนระบบปฏิบัติการลินุกซ์ ซึ่งในการดำเนินงานผู้วิจัยได้ทดลองใช้บอร์ด Raspberry Pi 3 Model B ซึ่งมีรายละเอียดดังนี้

### 2.6.1 บอร์ด Raspberry Pi 3 Model B และโมดูลกล้อง V2

บอร์ด Raspberry Pi 3 Model B มีขนาดเท่ากับบอร์ด Raspberry Pi 2 Model B แต่มีองค์ประกอบภายในบอร์ดมากกว่า โดยบอร์ด Raspberry Pi 3 Foundation ใช้ชิพ Broadcom BCM2837 64-bit Quad-Core ARM Cortex-A53 ARMv8 ความเร็ว 1.2 GHz มีหน่วยความจำ LPDDR 2 SDRAM ขนาด 1 GB ชิพ Broadcom BCM43438 เป็น Wi-Fi 802.11 b/g/n และ Bluetooth 4.1 (Classic และ Low-Energy) ทำให้สามารถเชื่อมต่อได้อย่างรวดเร็วโดยไม่ต้องต่ออุปกรณ์อื่น ๆ เพิ่ม พร้อมสายอากาศแบบ Chip Antenna บนบอร์ด สำหรับความเร็วการประมวลผลพบว่าชิพของบอร์ด Pi 3 มีความเร็วมากกว่าบอร์ด Pi 2 มากขึ้นจากของเดิม 900MHz ไป 1.2 GHz รองรับรุ่นล่าสุดของ ARM GNU/Linux distributions และ Windows 10 IoT นอกจากนี้ตำแหน่งคอนเน็คเตอร์ของบอร์ด Raspberry Pi 3 Model B มีหลอด LED แสดงสถานะ Power กับ Activity ถูกแทนที่ด้วยเสาอากาศทำให้ LED ย้ายไปอยู่บริเวณภาคจ่ายไฟ สำหรับโมดูลกล้องที่ใช้คือ กล้องอินฟาเรดโมดูล V2 (Pi NoIR) ที่มีเซ็นเซอร์ Sony IMX219 8-megapixel รองรับภาพความละเอียด 1080p30 720p60 และโหมดวิดีโอ VGA90 รวมถึงภาพนิ่ง มีการปรับสีและการทำงานในที่แสงน้อยได้ดี [27] รายละเอียดของบอร์ด Raspberry Pi 3 Model B แสดงในตารางที่ 2.1

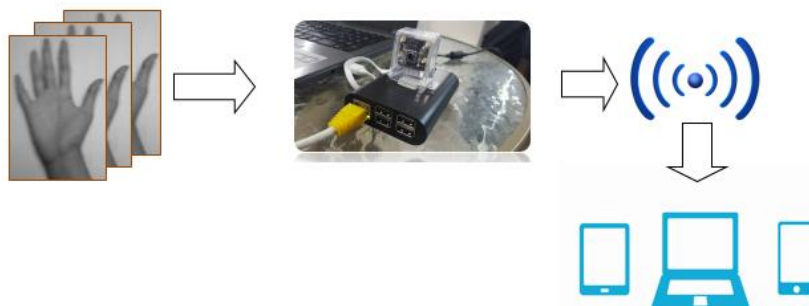
ตารางที่ 2.1 รายละเอียดบอร์ด Raspberry Pi 3 Model B

บอร์ด	รายละเอียด
การประมวลผล	Broadcom BCM2837
CPU Core	Quadcore ARM Cortex-A53, 64Bit
Clock Speed	1.2GHz (Roughly 50% faster than Pi2)
RAM	1 GB
GPU	400 MHz VideoCore IV®

การเชื่อมต่อเครือข่าย	1 x 10 / 100 Ethernet (RJ45 Port)
การเชื่อมต่อไร้สาย	802.11n wireless LAN (WiFi) and Bluetooth 4.1
พอร์ต USB	4 x USB 2.0
GPIOs	2 x 20 Pin Header
อินเตอร์เฟซกล้อง	15-pin MIPI
หน้าจอแสดงผล	DSI 15 Pin / HDMI Out / Composite RCA
เอาต์พุตวิดีโอ	HDMI/Composite
เอาต์พุตเสียง	HDMI/Headphone
แหล่งพลังงาน	2.5 A
พื้นที่เก็บข้อมูล	Micro-SD

## 2.6.2 การเชื่อมต่อระหว่างบอร์ดและสมาร์ทโฟน

จากความสำคัญของเทคโนโลยีในปัจจุบัน มีเทคโนโลยีต่าง ๆ ที่นำมาใช้และพัฒนามาบนโทรศัพท์เคลื่อนที่ ซึ่งในปัจจุบันจัดเป็นยุคของโทรศัพท์อัจฉริยะ หรือสมาร์ทโฟน ที่มีความสามารถในการเชื่อมต่อเครือข่ายอินเทอร์เน็ต รองรับการทำงานของแอปพลิเคชันต่าง ๆ อาทิ การท่องเว็บ การส่งข้อความออนไลน์ การถ่ายรูปและปรับแต่งรูป นอกจากนี้พื้นฐานการทำงานแบบเดิม อาทิ การโทรออก การรับสาย การส่งข้อความ เป็นต้น จากการพัฒนาทางด้านเทคโนโลยีของสมาร์ทโฟน จึงสามารถใช้เครือข่ายอินเทอร์เน็ตในการเชื่อมต่อระหว่างผู้ใช้งานและผู้ดูแล โดยใช้แอปพลิเคชันไลน์ (Line) ซึ่งเป็นหนึ่งในแอปพลิเคชันที่เป็นที่นิยมอย่างมากในการส่งข้อความนอกเหนือจาก Facebook และ What's app เนื่องจากความสะดวกรวดเร็วในการติดต่อสื่อสาร ดังนั้นการแจ้งเตือนของระบบ รู้จำท่ามือผู้สูงอายุจึงใช้การประมวลผลจากบอร์ด Raspberry Pi และส่งข้อความแจ้งเตือนความหมายของท่ามือผ่านทางแอปพลิเคชันไลน์ในสมาร์ทโฟน แท็บเล็ต หรือคอมพิวเตอร์ได้ทันที



รูปที่ 2.15 แสดงการเชื่อมต่อระหว่างบอร์ด Raspberry Pi และสมาร์ทโฟน

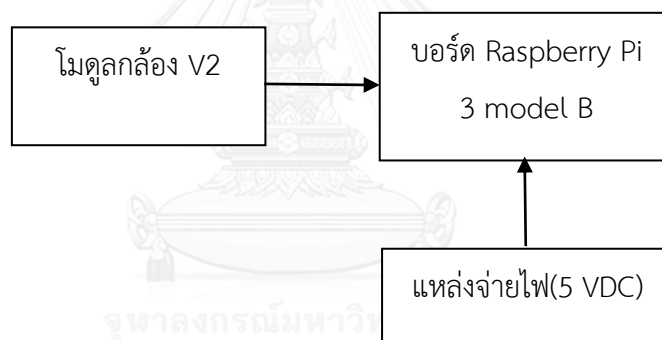
### บทที่ 3

#### วิธีดำเนินการทดลอง

ผู้วิจัยทำการทดลองและหาวิธีการที่เหมาะสมสำหรับการพัฒนาระบบรู้จำท่าทางมือสำหรับผู้สูงอายุ อัลกอริธึมที่พัฒนาขึ้นทำงานบนอุปกรณ์บอร์ด Raspberry Pi 3 model B พัฒนาระบบโปรแกรม Python เวอร์ชัน 2.7 ร่วมกับไลบรารี Open CV เวอร์ชัน 3.1 โดยแบ่งระบบออกเป็น 2 ส่วน ได้แก่ ส่วนฮาร์ดแวร์และซอฟต์แวร์ระบบ ดังนี้

#### 3.1 ฮาร์ดแวร์ระบบรู้จำท่าทางมือแบบเคลื่อนไหว

โครงสร้างของระบบรู้จำท่าทางมือประกอบด้วย 2 ส่วนหลักๆ ได้แก่ ส่วนโมดูลกล้อง V2 และบอร์ด Raspberry Pi 3 model B โดยที่บอร์ดรับพลังงานจากแหล่งจ่ายไฟขนาด 5 โวลต์ ดังแสดงในแผนผังระบบดังนี้



รูปที่ 3.1 แผนผังแสดงฮาร์ดแวร์ระบบรู้จำท่าทางมือแบบเคลื่อนไหว

รายละเอียดและหน้าที่ของส่วนประกอบทั้ง 2 ส่วนมีดังต่อไปนี้

##### 3.1.1 โมดูลกล้อง V2

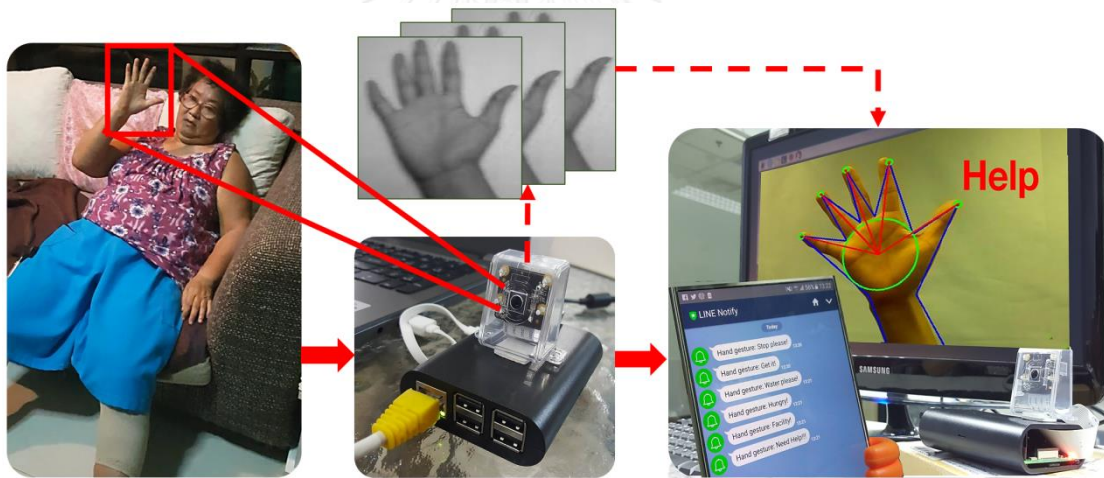
โมดูลกล้อง (Camera Module V2) ทำหน้าที่รับภาพเคลื่อนไหวแบบเวลาจริงแล้วส่งเฟรมที่ได้รับไปประมวลผลยังบอร์ด Raspberry Pi 3 model B ผ่านการเชื่อมต่อทางสายเคเบิลที่ต่อกับพอร์ต CSI บนบอร์ด ภาพที่ได้มีความคมชัดตามการกำหนดค่าความละเอียดของกล้อง V2

### 3.1.2 บอร์ด Raspberry PI 3 model B

บอร์ด Raspberry PI 3 model B ทำหน้าที่ควบคุมการทำงานของโมดูลกล้อง V2 และการประมวลผล ออกแบบให้บอร์ด Raspberry PI 3 model B ได้รับพลังงานจากแหล่งจ่ายไฟตรงขนาด 5 โวลต์



รูปที่ 3.2 บอร์ด Raspberry PI 3 model B และโมดูลกล้อง V2



รูปที่ 3.3 แผนผังแสดงการทำงานของระบบรู้จำท่าทางมือแบบเคลื่อนไหว

## 3.2 ขั้นตอนการทำงานของระบบรู้จำท่าทางมือแบบเคลื่อนไหว

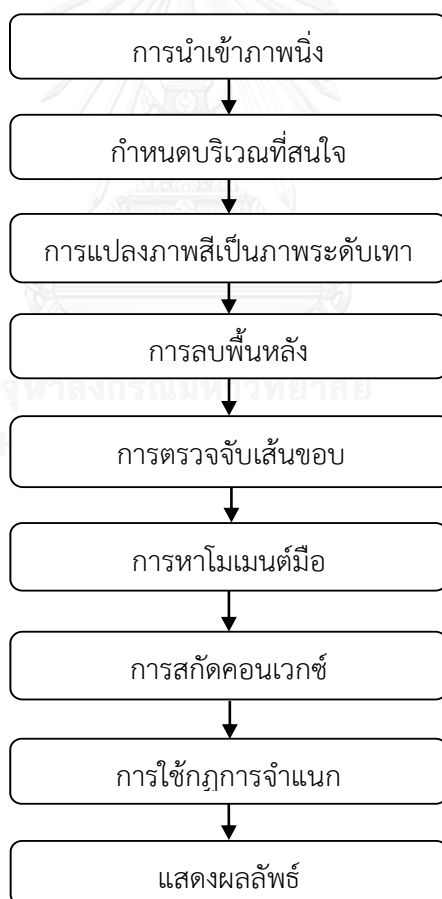
ผู้วิจัยทำการทดลองการตรวจจับท่าทางด้วยอัลกอริทึมที่ต่างกันทั้งหมด 3 อัลกอริทึม ได้แก่

- 3.2.1 อัลกอริทึมรู้จำท่าทางแบบภาพนิ่งสำหรับผู้สูงอายุโดยใช้กฎการจำแนก
- 3.2.2 อัลกอริทึมรู้จำท่าทางสำหรับผู้สูงอายุโดยใช้ Optical Flow และ SURF
- 3.2.3 อัลกอริทึมที่นำเสนอสำหรับระบบรู้จำท่าทางมือบนพื้นฐานการมองเห็นสำหรับผู้สูงอายุ

โดยแต่ละอัลกอริทึมจะมีขั้นตอนการทำงานและวิธีการที่ใช้เหมือนกันและแตกต่างกันในบางกระบวนการ กระบวนการที่มีความเหมือนกัน ได้แก่ การนำเข้าข้อมูล ที่ถือเป็นกระบวนการขั้นพื้นฐานขั้นตอนแรกของระบบ ระบบจะนำเข้าข้อมูลภาพนิ่ง วิดีโอ และภาพเคลื่อนไหวสำหรับอัลกอริทึมในหัวข้อ 3.2.1 3.2.2 และ 3.2.3 ตามลำดับ สำหรับกระบวนการที่มีความแตกต่างกัน ได้แก่ กระบวนการการปรับปรุงคุณภาพภาพ การติดตามมือ การวิเคราะห์ท่ามือ และการจำแนกท่ามือ เป็นต้น

### 3.2.1 อัลกอริทึมรู้จำท่ามือแบบสถิติสำหรับผู้สูงอายุโดยใช้กฎการจำแนก

สำหรับขั้นตอนการทำงานของอัลกอริทึมรู้จำท่ามือแบบสถิติสำหรับผู้สูงอายุโดยใช้กฎการจำแนกสามารถจำแนกได้เป็น 9 ขั้นตอน อธิบายได้ตามแผนผังแสดงขั้นตอนการทำงานหลักของระบบดังแสดงในรูปที่ 3.4



รูปที่ 3.4 แผนผังการทำงานของอัลกอริทึมรู้จำท่ามือแบบสถิติสำหรับผู้สูงอายุโดยใช้กฎการจำแนก

### 3.2.1.1 การนำเข้าภาพนิ่ง

สำหรับอัลกอริธึมรู้จำท่ามือแบบสถิตสำหรับผู้สูงอายุโดยใช้กฎการจำแนก ผู้วิจัยทำการทดลองด้วยการวิเคราะห์ท่ามือจากภาพของท่ามือที่แตกต่างกัน 6 ท่ามือที่ได้จากโมดูลกล้อง V2 ตั้งค่าเริ่มต้นโมดูลกล้องให้มีความละเอียดที่ 800 x 600 พิกเซลโดยเฟรมภาพที่ได้จะมีขนาดเท่ากับความคมชัดของโมดูลกล้องที่ใช้บันทึกโดยไม่ต้องมีการตั้งค่าเพิ่มเติมใดๆ

### 3.2.1.2 การกำหนดบริเวณที่สนใจ

หลังจากระบบนำเข้าไฟล์ภาพนิ่งเก็บไว้ในอาเรย์แล้ว ระบบทำการกำหนดบริเวณที่สนใจภายในเฟรมภาพนั้นคือพื้นที่มือ เพื่อช่วยให้การประมวลผลของระบบมีความรวดเร็วมากยิ่งขึ้น การกำหนดบริเวณที่สนใจทำได้โดยกำหนดขนาดของกรอบสี่เหลี่ยมล้อมรอบท่ามือ ภาพที่ได้จากการระบุบริเวณที่สนใจแสดงในรูปที่ 3.5



(ก) ท่ากำมือ (ข) ท่าชี้นิ้ว (ค) ท่าชูสองนิ้ว (ง) ท่าชูสามนิ้ว (จ) ท่าชูสี่นิ้ว (ฉ) ท่าแบมือ

รูปที่ 3.5 การกำหนดบริเวณที่สนใจ

### 3.2.1.3 การแปลงภาพสีเป็นภาพระดับเทา

เมื่อได้บริเวณที่สนใจแล้ว ขั้นตอนต่อไปคือการแปลงภาพสีเป็นภาพระดับเทา เนื่องจากการประมวลผลบนภาพระดับเทาจะทำให้ระบบมีความเร็วในการประมวลผลสูงขึ้น เพราะภาพระดับเทาจะมีองค์ประกอบสีน้อยกว่าโมเดลสี RGB ภาพผลลัพธ์ที่ได้จากการแปลงสีแสดงในรูปที่ 3.6



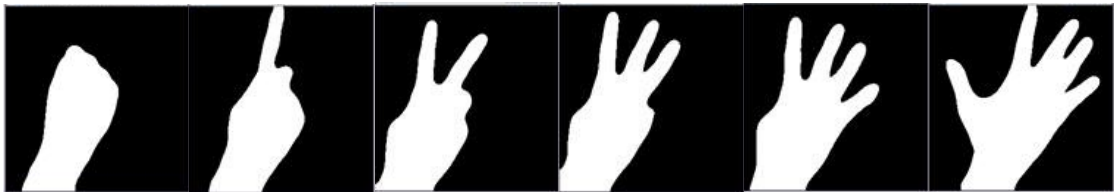
(ก) ท่ากำมือ (ข) ท่าชี้นิ้ว (ค) ท่าชูสองนิ้ว (ง) ท่าชูสามนิ้ว (จ) ท่าชูสี่นิ้ว (ฉ) ท่าแบมือ

รูปที่ 3.6 ภาพระดับเทาของท่ามือทั้ง 6 ท่าที่ได้จากการแปลงภาพสี



### 3.2.1.4 การลบพื้นหลัง

วิธีการลบพื้นหลังถูกนำมาใช้เพื่อพิจารณาพิกเซลที่เป็นส่วนของมือเท่านั้น ในระบบนี้วิธีการลบพื้นหลังทำงานบนภาพระดับเทา โดยคำนวณความแตกต่างของเฟรมจากการเคลื่อนที่ของมือกับภาพพื้นหลังที่ไม่เคลื่อนที่ หลังจากนั้นตั้งค่าเทรชโฮลด์ (Threshold) เพื่อสร้างภาพไบนารี หรือเรียกว่าการทำภาพขาวดำสองระดับ (Binarization) ค่าเทรชโฮลด์จะทำให้ระบบสามารถแยกมือออกจากพื้นหลังได้ โดยพิกเซลสีขาวคือส่วนของมือ และพิกเซลสีดำคือส่วนพื้นหลัง นอกจากนี้ในขั้นตอนการลบพื้นหลังจะมีการใช้กระบวนการแบบเปิด (Opening operation) เพื่อปรับเส้นของมือให้มีความเนียนมากยิ่งขึ้น ภาพผลลัพธ์ที่ได้จากการลบพื้นหลังแสดงในรูปที่ 3.7



(ก) ทำกำมือ (ข) ทำชี้นิ้ว (ค) ทำชูสองนิ้ว (ง) ทำชูสามนิ้ว (จ) ทำชูสี่นิ้ว (ฉ) ทำแบมือ

รูปที่ 3.7 ภาพที่ได้จากการลบพื้นหลัง

### 3.2.1.5 การตรวจจับเส้นขอบ

หลังจากที่ได้ภาพไบนารีของมือจากขั้นตอนการลบพื้นหลังแล้ว ขั้นตอนต่อไปจะเป็นการหาเส้นขอบของมือ เส้นขอบของมือที่คำนวณได้อาจมีหลายเส้น แต่เส้นขอบที่ใหญ่ที่สุดจะสามารถนำมาคำนวณหาจุดศูนย์กลางของมือได้ นอกจากนี้การหาเส้นขอบของมือทำให้สามารถรู้รูปร่างของมือเพื่อนำไปหาคอนเวกซ์ฮัลล์ (Convex Hull) หรือเส้นรอบมือที่ไม่มีส่วนเว้า เพื่อทำการหาจุดยอดมุมของนิ้วมือในลำดับถัดไป โดยในการสร้างแบบจำลองนั้นเส้นขอบจะถูกกำหนดไว้ในระนาบ  $(x, y)$  ของภาพในลักษณะเส้นโค้งพาราเมตริก ดังแสดงในสมการต่อไปนี้

$$v(s) = (x(s), y(s)) \quad (3.1)$$

โดย  $s$  เป็นพารามิเตอร์ที่เพิ่มขึ้นรอบเส้นแสดงรูปร่าง มีความสัมพันธ์กับความยาวเส้นแสดงรูปร่าง นอกจากนี้เส้นขอบหรือ Snake จะเคลื่อนที่ตามพลังงานภายในและภายนอก โดยทั่วไปแล้วเส้นขอบแสดงค่าพลังงาน (Energy:  $e_{snake}$ ) เกิดจากการรวมกันของพลังงาน 3 ประเภท ได้แก่ พลังงานภายใน (Internal energy:  $e_{int}$ ) สำหรับรักษารูปร่างของเส้นโค้ง พลังงานภายนอก

(External energy:  $e_{ext}$ ) สำหรับผลักดันเส้นโค้งไปในทิศทางที่ต้องการ และพลังงานคงที่ (Constrain energy:  $e_{cons}$ ) สามารถคำนวณค่าพลังงานได้ดังนี้

$$e_{snake} = e_{int} + e_{ext} + e_{cons} \quad (3.2)$$

จากสมการดังกล่าวพบว่าตำแหน่งสุดท้ายของเส้นโค้งมีค่าพลังงานน้อย สำหรับพลังงานภายในสามารถคำนวณได้จากการรวมกันของพลังงานยืดหยุ่น (Elastic energy:  $e_{elas}$ ) และพลังงานดัดโค้ง (Bending energy:  $e_{bend}$ )

พลังงานยืดหยุ่น ( $e_{elas}$ ) จะแสดงคุณลักษณะของพลังงานที่มีความยืดหยุ่น ซึ่งจะลดการยืดและเหมาะกับการหดตัวของเส้นโค้ง ดังแสดงในสมการที่ 3.3

$$e_{elas} = \frac{1}{2} \int_S \alpha(s) |v_s|^2 ds \quad (3.3)$$

เมื่อ  $\alpha(s)$  คือ พารามิเตอร์ Weight สำหรับควบคุมพลังงานยืดหยุ่นในแต่ละส่วนของเส้นโค้ง พลังงานดัดโค้ง ( $e_{bend}$ ) เกิดจากการรวมกันของเส้นโค้งที่มีลักษณะสี่เหลี่ยม พลังงานนี้สามารถนำไปใช้กับแถบโลหะบางได้ การคำนวณของพลังงานดัดโค้งแสดงดังสมการที่ 3.4

$$e_{bend} = \frac{1}{2} \int_S \beta(s) |v_{ss}|^2 ds \quad (3.4)$$

เมื่อ  $\beta(s)$  คือ พารามิเตอร์ Weight สำหรับควบคุมความดัดโค้งในแต่ละส่วนของเส้นโค้ง ดังนั้นค่าพลังงานภายในสามารถคำนวณได้ดังนี้

$$e_{int} = e_{elas} + e_{bend} = \int_S \frac{1}{2} (\alpha |v_s|^2 + \beta |v_{ss}|^2) ds \quad (3.5)$$

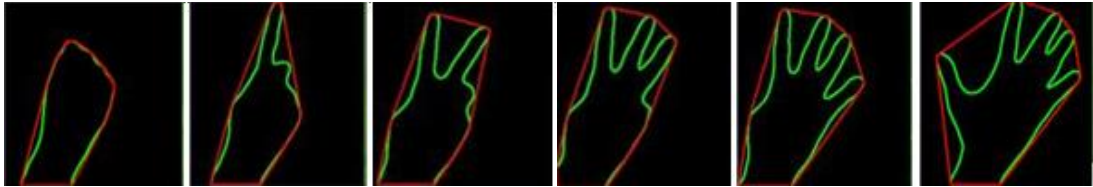
สำหรับการคำนวณหาค่าพลังงานภายนอกจากภาพสามารถทำได้โดยใช้ฟังก์ชัน  $e_{img}(x,y)$  ดังแสดงในสมการที่ 3.6 โดยที่พลังงานภายนอกนี้จะดำเนินการกับค่าเล็กๆ เช่น ขอบของเส้นมือ เป็นต้น

$$e_{ext} = \int_S e_{img}(v(s)) ds \quad (3.6)$$

ดังนั้นพลังงานในการหาเส้นโค้งเพื่อลดฟังก์ชันพลังงานของมือสามารถคำนวณได้โดยใช้สมการ Lagrange ดังแสดงในสมการที่ 3.7

$$e_{snake} = \int_S \frac{1}{2} (\alpha(s)|v_s|^2 + \beta(s)|v_{ss}|^2) + e_{img}(v(s)) ds \quad (3.7)$$

$$\alpha v_{ss} - \beta v_{ssss} - \nabla e_{img} = 0 \quad (3.8)$$



(ก) ท่ากำมือ (ข) ท่าชี้นิ้ว (ค) ท่าชูสองนิ้ว (ง) ท่าชูสามนิ้ว (จ) ท่าชูสี่นิ้ว (ฉ) ท่าแบมือ

รูปที่ 3.8 ผลลัพธ์การตรวจจับเส้นขอบมือ

### 3.2.1.6 การหาโมเมนต์ภาพ (Image Moments)

หลังจากได้เส้นขอบมือมาแล้ว ทำการหาจุดโมเมนต์ของมือหรือคำนวณหาจุดเซนทรอยด์ของฝ่ามือตามแนวแกน  $x$  และแกน  $y$  จากนั้นสร้างวงกลมที่มีการกำหนดรัศมีบริเวณกลางฝ่ามือ เมื่อได้วงกลมแล้ว กำหนดกรอบสี่เหลี่ยมแบบเส้นตรงล้อมพื้นที่มือและวาดกรอบสี่เหลี่ยมครอบคลุมพื้นที่ในการประมวลผลท่าทางมือ

โดยที่โมเมนต์ของภาพ (Image Moments) หรือเซนทรอยด์ของภาพ (Centroid) คือค่าเฉลี่ยถ่วงน้ำหนักของทุกพิกเซล โดยที่น้ำหนักคือความเข้ม โมเมนต์ตามแนวแกน  $x$  และแกน  $y$  ของภาพ คุณลักษณะของโมเมนต์ภาพสามารถดำเนินการได้โดยไม่ต้องพิจารณาตำแหน่งรูปภาพและขนาดของมือ โดยฟังก์ชัน  $M(i,j)$  จะแสดงมือในภาพและสร้างโมเมนต์  $M_{ij}$  เพื่อแทนคุณลักษณะของมือ สามารถคำนวณได้จากสมการที่ 3.9 และ 3.10 ตามลำดับ

$$M_i = \sum_{ij} x_i M(i,j) \quad (3.9)$$

$$M_j = \sum_{ij} y_j M(i,j) \quad (3.10)$$

ทั้งนี้โมเมนต์ของภาพประกอบด้วย 3 องค์ประกอบ ได้แก่ จุดศูนย์กลางมวล (Centre of Mass), ความแปรปรวน (Variance) และ การกำหนดทิศทาง (orientation) ดังนี้

#### จุดศูนย์กลางมวล (Centre of Mass)

จุดศูนย์กลางมวลหรือเรียกว่าเซนทรอยด์สามารถกำหนดได้จากค่าความเข้มของ  $M(i,j)$  ในแต่ละจุด  $(i,j)$  ของภาพ  $(M)$  ซึ่งแสดงถึงมวลของพิกัด  $(i,j)$  โดยค่ามวลวัตถุจะไม่เปลี่ยนแปลงไปตามการ

เปลี่ยนแปลงของโมเมนต์อันดับ 1 ของมือ ณ แกนใดๆ ค่าจุดศูนย์กลางมวลสามารถหาได้จากสมการที่ (3.11) และ (3.12)

$$M_{10} = \sum \sum x_i M(i, j) / M_{00} \quad (3.11)$$

$$M_{01} = \sum \sum y_j M(i, j) / M_{00} \quad (3.12)$$

ความแปรปรวน (Variance)

ค่าความแปรปรวนแปร ( $\sigma_2$ ) ได้จากการคำนวณโมเมนต์อันดับ 2 ของเซนทรอยด์  $M_{ij}^a$

$$\sigma_x^2 = M_{20}^a = M_{20} - M_{10}^2 \quad (3.13)$$

$$\sigma_y^2 = M_{02}^a = M_{02} - M_{01}^2 \quad (3.14)$$

การกำหนดทิศทาง (Orientation)

การกำหนดทิศทาง คือ การทำมุมของแกนที่มีโมเมนต์ความเฉื่อยน้อยที่สุด สามารถคำนวณได้จากสมการที่ (3.15) (3.16) และ (3.17)

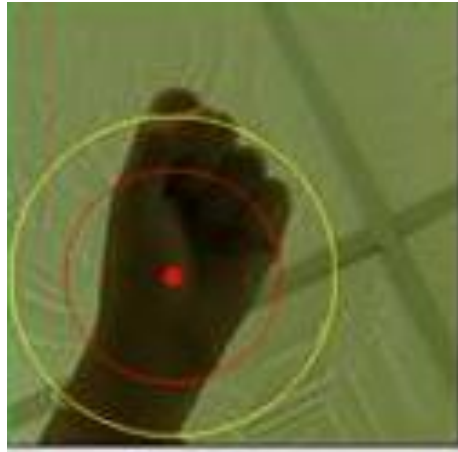
$$\tan(2\theta) = \frac{2M_{11}}{M_{20} - M_{02}} \quad (3.15)$$

ให้  $M_{11} = 0$  และ  $M_{20} = B_{02}$ ,

$$\sin(2\theta) = \frac{\pm 2M_{11}}{\sqrt{(2M_{11})^2 + (M_{20} - M_{02})^2}} \quad (3.16)$$

$$\cos(2\theta) = \frac{\pm (M_{20} - M_{02})}{\sqrt{(2M_{11})^2 + (M_{20} - M_{02})^2}} \quad (3.17)$$

จากสมการข้างต้นพบว่าโมเมนต์อันดับที่ 1 และอันดับที่ 2 ใช้สำหรับการคำนวณหาจุดศูนย์กลางมวล ความแปรปรวนแปร และการกำหนดทิศทางของมือได้ แต่ถ้าโมเมนต์มีค่าคงที่ในการแปลงเชิงเส้น นั้นแสดงว่าภาพต้นฉบับไม่มีความจำเป็นที่จะมีโมเมนต์อันดับที่ 1 และอันดับที่ 2 หลังเสร็จสิ้นการประมวลผลจะพบจุดศูนย์กลางมวลของมือและนำไปใช้ในขั้นตอนต่อไป



รูปที่ 3.9 ภาพผลลัพธ์จากการหาจุดโมเมนต์มือ

### 3.2.1.7 การสกัดคอนเวกซ์

สำหรับการหาจุดยอดมุมนั้นมีวัตถุประสงค์เพื่อแยกแยะจุดปลายนิ้วมือออกจากจุดยอดมุมอื่นๆ การหาจุดยอดมุมนั้นสามารถเริ่มจากการหาคอนเวกซ์ฮัลล์ (Convex hull) จากเส้นขอบที่ใหญ่ที่สุดของมือ ผลลัพธ์ที่ได้คือเส้นขอบล้อมรอบมือที่ไม่มีส่วนเว้า หรือเป็นเส้นขอบที่ครอบคลุมพื้นที่ตั้งแต่ปลายแขนจนไปถึงปลายนิ้วรวมทั้งช่องว่างระหว่างนิ้วมือด้วย ต่อมาจะทำการคำนวณหาจุดบนเส้นขอบที่ใหญ่ที่สุดซึ่งอยู่ห่างจากคอนเวกซ์ฮัลล์ในแต่ละด้านมากที่สุด หรือคำนวณหาจุดที่อยู่ลึกที่สุดในรอยเว้าบนเส้นขอบ ในแต่ละด้านของคอนเวกซ์ฮัลล์ จุดที่อยู่ลึกที่สุดในรอยเว้าเรียกว่า “Convexity defects” เมื่อทราบจุดที่อยู่ลึกที่สุดในรอยเว้า และจุดยอดบนคอนเวกซ์ฮัลล์ระบบจะสามารถระบุตำแหน่งของปลายนิ้วได้ทันที เนื่องจากจุดยอดบนคอนเวกซ์ฮัลล์ใดๆ ที่อยู่ระหว่างจุดที่อยู่ลึกที่สุดในรอยเว้าจะเป็นจุดที่แสดงถึงปลายนิ้วมือ ภาพผลลัพธ์จากการหาจุดลึกที่สุดในรอยเว้าแสดงในรูปที่ 3.10



(ข) ท่าชี้นิ้ว

(ค) ท่าชูสองนิ้ว

(ง) ท่าชูสามนิ้ว

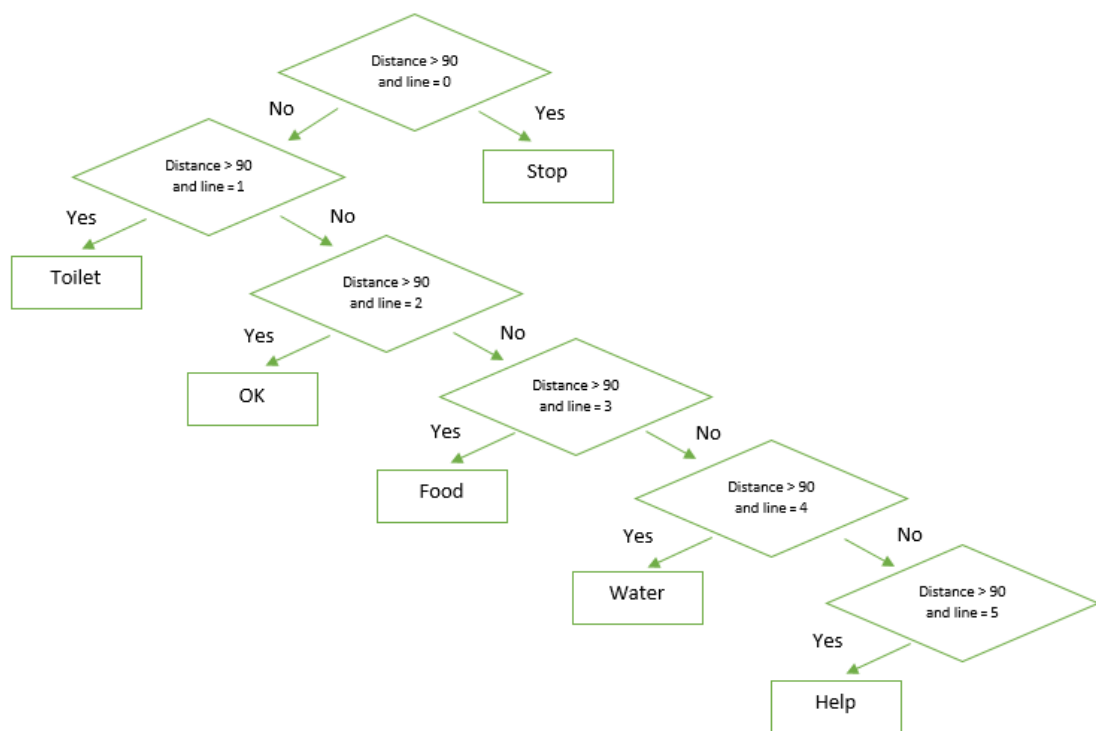
(จ) ท่าชูสี่นิ้ว

(ฉ) ท่าแบมือ

รูปที่ 3.10 ภาพผลลัพธ์จากการหาจุดลึกที่สุดในรอยเว้า

### 3.2.1.8 การใช้กฎการจำแนก

กฎนี้จะพิจารณาระยะห่างระหว่างจุดเซนทรอยด์ของมือและจุดปลายนิ้วมือเพื่อนับจำนวนนิ้วที่ชูขึ้นโดยใช้วิธีการยูคลิด (Euclidean distance) โดยตั้งค่าเทรชโฮลด์เท่ากับ 90 พิกเซล โดยถ้าระยะห่างที่ได้มีค่าน้อยกว่า 90 พิกเซล จะไม่มีเส้นตรงใดๆเกิดขึ้น ดังนั้นระบบจะจำแนกทำนั้นคือทำกำมือมีความหมายว่า “หยุด” แต่ถ้าระยะห่างมีค่ามากกว่า 90 พิกเซล ระบบจะพิจารณาว่ามีท่าอะไรบ้างจากจำนวนเส้นตรงที่เกิดขึ้นและทำการแปลความหมายของแต่ละท่า กฎการจำแนกแบ่งออกได้เป็น 6 ขั้นตอนดังในรูปที่ 3.11



รูปที่ 3.11 กฎการจำแนก

กฎข้อที่ 1: ระบบตีความภาพของท่ากำมือ พบว่าไม่มีเส้นที่มีระยะจากจุดศูนย์กลางของมือไปยังจุดปลายนิ้วมือใด ๆ เกิน 90 พิกเซล ดังนั้นกำหนดให้ทำนั้นมีความหมายว่า หยุด

กฎข้อที่ 2: ระบบตีความภาพของท่าชูนิ้วชี้ พบว่ามีเส้นที่มีระยะจากจุดศูนย์กลางของมือไปยังจุดปลายนิ้วชี้เกิน 90 พิกเซล ดังนั้นกำหนดให้ทำนั้นมีความหมายว่า ห้องน้ำ

กฎข้อที่ 3: ระบบตีความภาพของท่าชูนิ้วชี้และนิ้วกลาง พบว่ามีเส้น 2 เส้นที่มีระยะจากจุดศูนย์กลางของมือไปยังจุดปลายนิ้วชี้และนิ้วกลาง เกิน 90 พิกเซล ดังนั้นกำหนดให้ทำนั้นมีความหมายว่า หยุด

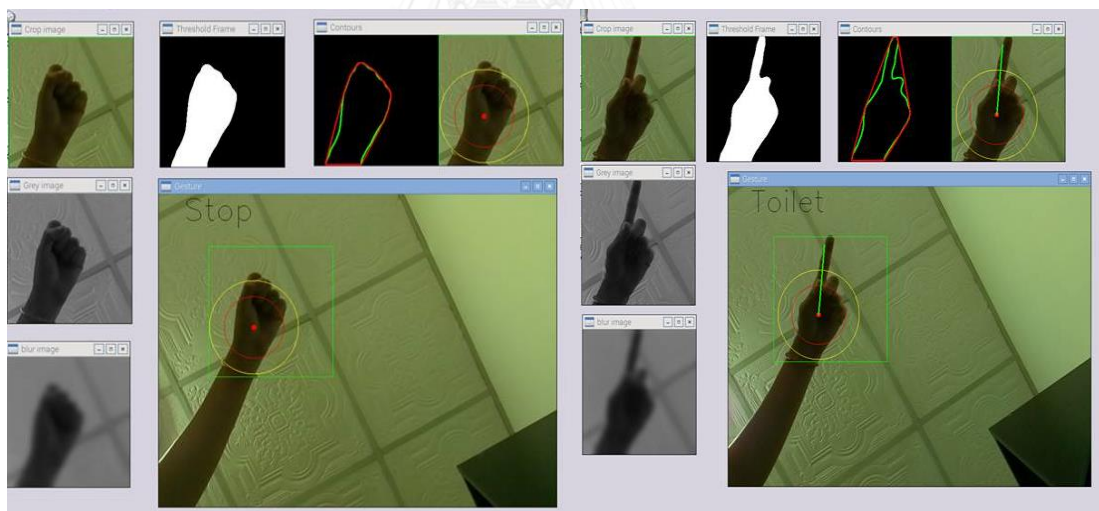
กฎข้อที่ 4: ระบบตีความภาพของท่าชูนิ้วชี้ นิ้วกลาง และนิ้วนาง พบว่ามีเส้น 3 เส้นที่มีระยะจากจุดศูนย์กลางของมือไปยังจุดปลายนิ้วชี้ นิ้วกลาง และนิ้วนาง เกิน 90 พิกเซล ดังนั้นกำหนดให้ท่านี้มีความหมายว่า อาหาร

กฎข้อที่ 5: ระบบตีความภาพของท่าชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย พบว่ามีเส้น 4 เส้นที่มีระยะจากจุดศูนย์กลางของมือไปยังจุดปลายนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย เกิน 90 พิกเซล ดังนั้นกำหนดให้ท่านี้มีความหมายว่า น้ำ

กฎข้อที่ 6: ระบบตีความภาพของท่าแบมือ พบว่ามีเส้น 5 เส้นที่มีระยะจากจุดศูนย์กลางของมือไปยังจุดปลายทั้ง 5 นิ้ว เกิน 90 พิกเซล ดังนั้นกำหนดให้ท่านี้มีความหมายว่า ช่วยด้วย

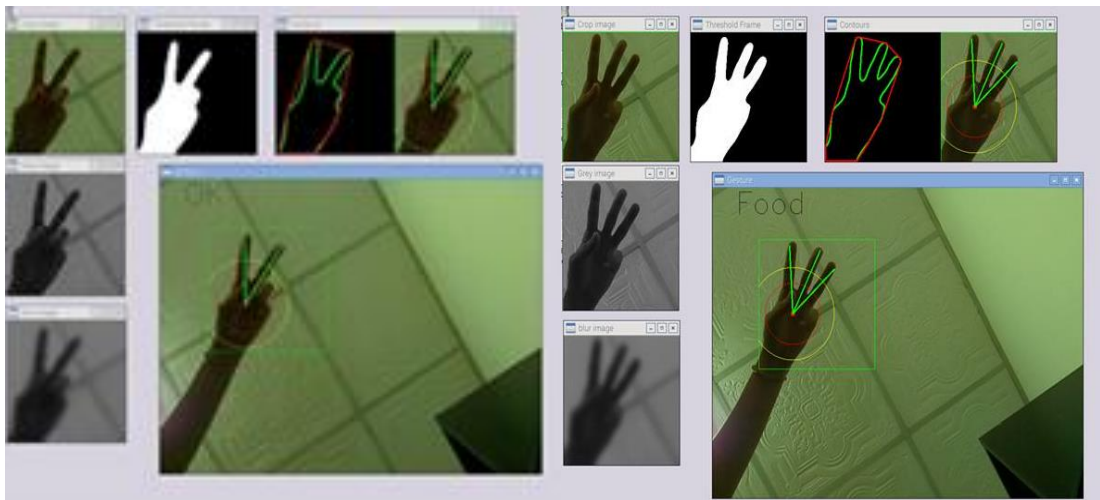
### 3.2.1.9 แสดงผลลัพธ์

ระบบแสดงผลการจำแนกความหมายของท่ามือทั้ง 6 ท่ามือทางจอคอมพิวเตอร์ โดยท่ากำมือหมายความว่า หยุด ท่าชูนิ้วชี้หมายความว่า ห้องน้ำ ท่าชูนิ้วชี้และนิ้วกลางหมายความว่า ตกลง ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนางหมายความว่า อาหาร ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อยหมายความว่า น้ำ และท่าแบมือหมายความว่า ช่วยด้วย ดังแสดงในรูปที่ 3.12



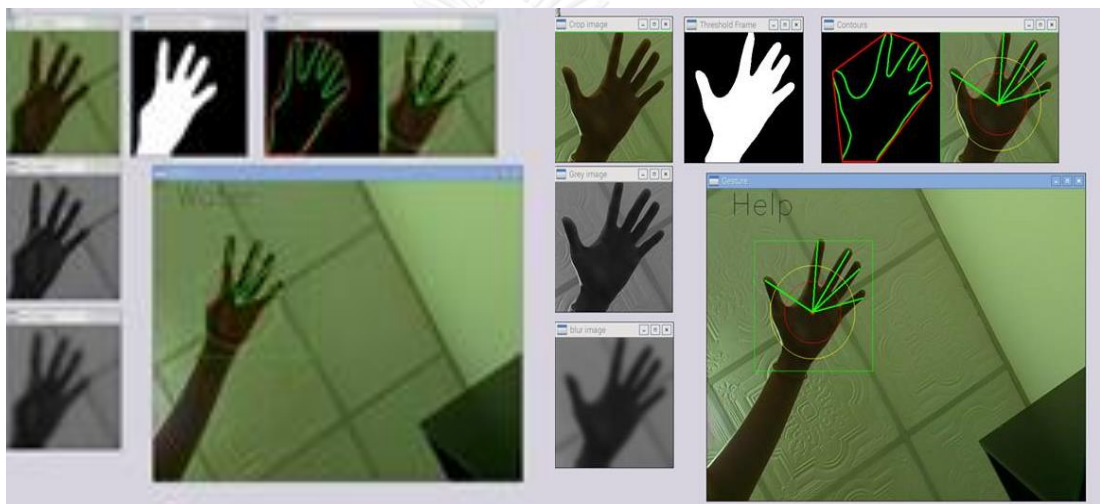
(ก) แสดงความหมายท่ากำมือ

(ข) แสดงความหมายท่าชูนิ้วชี้



(ค) แสดงความหมายท่าชูสองนิ้ว

(ง) แสดงความหมายท่าชูสามนิ้ว



(จ) แสดงความหมายท่าชูสี่นิ้ว

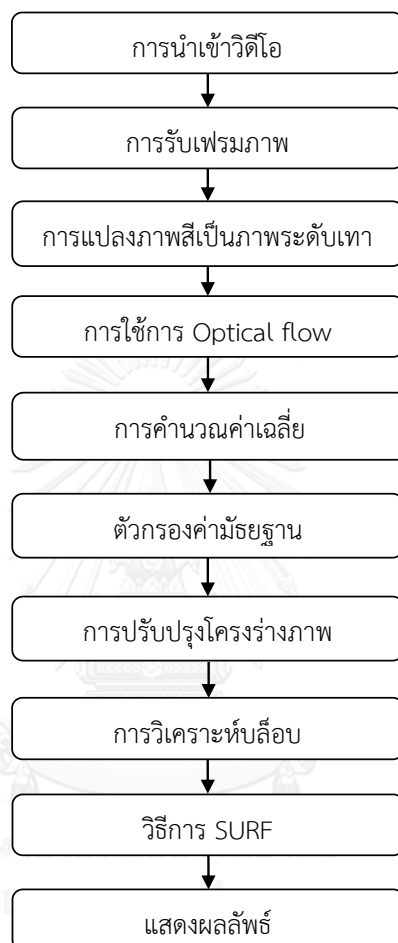
(ฉ) แสดงความหมายท่าแบมมือ

รูปที่ 3.12 ระบบแสดงผลการจำแนกความหมายของท่ามือทั้ง 6 ท่ามือ



### 3.2.2 อัลกอริธึมรู้จำท่ามือสำหรับผู้สูงอายุโดยใช้ Optical Flow และ SURF

สำหรับขั้นตอนการทำงานของอัลกอริธึมรู้จำท่ามือสำหรับผู้สูงอายุโดยใช้ Optical Flow และ SURF สามารถจำแนกได้เป็น 10 ขั้นตอน อธิบายได้ตามแผนผังขั้นตอนการทำงานดังนี้



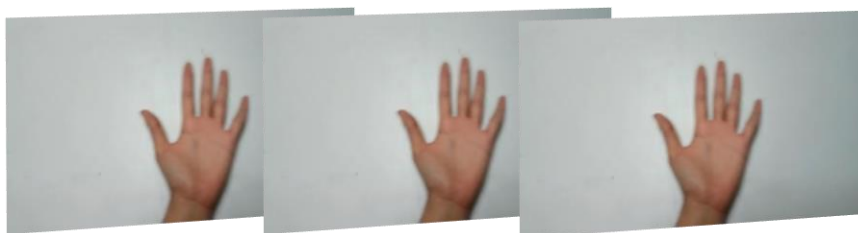
รูปที่ 3.13 แผนผังการทำงานของอัลกอริธึมรู้จำท่ามือโดยใช้ Optical Flow และ SURF

ในงานวิจัยนี้วิธี Optical flow และ SURF techniques คือกระบวนการหลักที่ใช้ในระบบ Optical flow ใช้สำหรับการตรวจจับและการติดตามมือร่วมกับการใช้การวิเคราะห์ blob ในขณะที่ SURF ใช้สำหรับการจำแนกท่ามือ โดยมีขั้นตอนการทำงานดังนี้

#### 3.2.2.1 การนำวิดีโอเข้าระบบ

อัลกอริธึมรู้จำท่ามือสำหรับผู้สูงอายุโดยใช้ Optical Flow และ SURF ผู้วิจัยทำการทดลองด้วยการวิเคราะห์ท่ามือจากไฟล์วิดีโอของท่ามือแบบภาพเคลื่อนไหวที่แตกต่างกัน 6 ท่ามือ โดยไฟล์

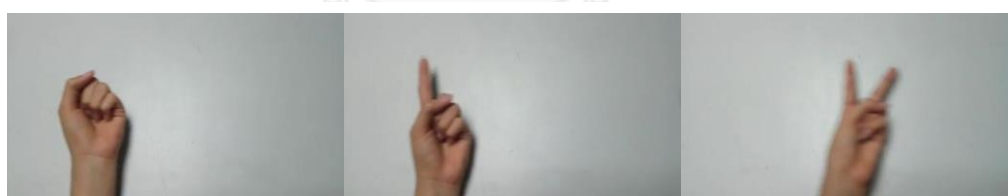
วิดีโอที่รับเข้าระบบจะบันทึกอยู่ในรูปแบบ .avi ตั้งค่าเริ่มต้นโมดูลกล้องให้มีความละเอียดที่ 800 x 600 พิกเซล หลังจากรับไฟล์วิดีโอแล้วระบบจะทำการเก็บเฟรมภาพสี RGB หรือเฟรมต้นฉบับไว้ในอาเรย์เพื่อประมวลผลต่อไป โดยเฟรมภาพที่ได้จะมีขนาดเท่ากับความคมชัดของโมดูลกล้องที่ใช้บันทึกโดยไม่ต้องมีการตั้งค่าเพิ่มเติมใดๆ



รูปที่ 3.14 การนำเข้าวิดีโอ

### 3.2.2.2 การจับเฟรมภาพ

สำหรับอัลกอริธึมที่นำเสนอสำหรับระบบรู้จำท่ามือบนพื้นฐานการมองเห็นสำหรับผู้สูงอายุ ภาพเคลื่อนไหวที่ได้จากโมดูลกล้องแบบเวลาจริงมีลักษณะเป็นเฟรมที่มีการเคลื่อนไหวมาปะติดปะต่อกันเป็นจำนวนมาก มีอัตราจำนวนเฟรมต่อวินาทีที่สูง การที่จะใช้เทคนิคการประมวลผลภาพจึงต้องแยกเฟรมที่เก็บในอาเรย์มาใช้ประมวลผล ทั้งนี้เฟรมที่ได้จากกล้องมีสัญญาณรบกวนเกิดขึ้นจากสภาพแสง ดังนั้นวิธีการปรับแต่งภาพจึงถูกเข้ามาช่วยเพื่อเพิ่มประสิทธิภาพในการประมวลผล



(ก) ทำกำมือ

(ข) ทำชูนิ้วชี้

(ค) ทำชูสองนิ้ว



(ง) ทำชูสามนิ้ว

(จ) ทำชูสี่นิ้ว

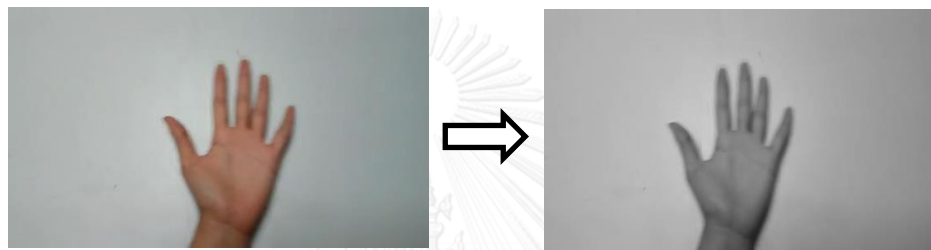
(ฉ) ทำแบมือ

รูปที่ 3.15 ภาพต้นฉบับของท่ามือทั้ง 6 ท่ามือที่ได้จากไฟล์วิดีโอ

### 3.2.2.3 การแปลงสีเป็นภาพระดับเทา

หลังจากระบบนำเข้าไฟล์วิดีโอและเก็บเฟรมภาพไว้ในอาเรย์แล้ว ขั้นตอนต่อไปคือการแปลงภาพสีเป็นภาพระดับเทา เนื่องจากภาพระดับเทามีเพียงระดับเฉดสีเทาเท่านั้น เมื่อมีการเปลี่ยนแปลงบนรูปภาพจะไม่ส่งผลกระทบต่อองค์ประกอบสีอื่น ๆ ทั้งนี้จัดว่าภาพระดับเทามีองค์ประกอบสีน้อยกว่าโมเดลสี RGB ทำให้ระบบประมวลผลได้รวดเร็วและมีประสิทธิภาพมากขึ้น ขั้นตอนนี้สามารถคำนวณได้ด้วยสมการของ Craig Markwart's ดังนี้

$$Y = 0.3R + 0.59G + 0.11B \quad (3.18)$$



(ก) ภาพต้นฉบับ

(ข) ภาพระดับเทา

รูปที่ 3.16 การแปลงภาพสีเป็นภาพระดับเทา

### 3.2.2.4 การไหลของแสง

Horn และ Schunck นำเสนอวิธีการแบบ Global โดยวิธีนี้อันุมานได้ว่าการไหลของแสงมีความหลากหลายได้อย่างราบรื่น จากนั้นพวกเขาทำการจำกัดความหลากหลายของการไหลตามสมการ (3.19) มีการเพิ่มข้อจำกัดการปรับให้ราบเรียบเพื่อทำให้การไหลของแสงเป็นไปอย่างราบรื่น โดยการคำนวณอนุพันธ์การไหล ได้แก่  $\nabla v_x$  และ  $\nabla v_y$  สมการความราบเรียบแสดงดังต่อไปนี้

$$e_s = \iint (\|\nabla v_x\|^2 + \|\nabla v_y\|^2) dx dy \quad (3.19)$$

ค่าความผิดพลาดของการไหลของแสงคือ

$$e_e = \iint \left( \nabla I \cdot v_m + \frac{\partial I}{\partial t} \right)^2 dx dy \quad (3.20)$$

จากนั้นลดค่าผลรวมความผิดพลาดโดยการคำนวณจากสมการที่ (3.19) และ (3.20) ดังนี้

$$e_t = e_e + \lambda e_s \quad (3.21)$$

ผลลัพธ์ของการไหลของแสงแสดงดังนี้



รูปที่ 3.17 การวิเคราะห์การเคลื่อนที่โดยใช้วิธีการไหลของแสงในเฟรมที่ต่อเนื่องกัน

### 3.2.2.5 การคำนวณค่าเฉลี่ย

ค่าเฉลี่ยเลขคณิตเป็นค่ากลางหรือเป็นตัวแทนของข้อมูลที่ดีที่สุด เพราะเป็นค่าที่ไม่เอนเอียง มีความคงเส้นคงวา มีความแปรปรวนต่ำที่สุด และมีประสิทธิภาพสูงสุด สำหรับวัตถุสองมิติสามารถหาได้จากลำดับของอินพุท หรือเฟรมภาพที่กำหนดด้วยค่าที่ระบุแล้วคำนวณด้วยจำนวนวัตถุเพื่อหาค่าเฉลี่ย จากการคำนวณสามารถติดตามค่าเฉลี่ยของแต่ละองค์ประกอบตามลำดับเฟรมภาพที่อยู่ในข้อมูลอาร์เรย์แรกที่มีขนาดไม่เท่ากับ 1 ในช่วงระยะเวลาหนึ่ง ๆ จากวิธีนี้พบว่าแต่ละองค์ประกอบจะพิจารณาเป็นช่อง (Channel) [28]

อย่างไรก็ตามค่าเฉลี่ยเลขคณิตมีข้อจำกัดในการใช้ เช่น ถ้าข้อมูลมีการกระจายมาก หรือข้อมูลบางตัวมีค่ามากหรือน้อยจนผิดปกติ หรือข้อมูลมีการเพิ่มขึ้นเป็นเท่าตัว ค่าเฉลี่ยเลขคณิตจะไม่สามารถเป็นค่ากลางหรือเป็นตัวแทนที่ดีของข้อมูลได้ ดังนั้นจึงต้องทำการใช้วิธีอื่นเพิ่มเติมเพื่อให้ได้ค่าที่ถูกต้อง โดยผู้พัฒนาใช้วิธีตัวกรองมัธยฐานในประมวลผลสำหรับขั้นตอนต่อไป

### 3.2.2.6 ตัวกรองค่ามัธยฐาน

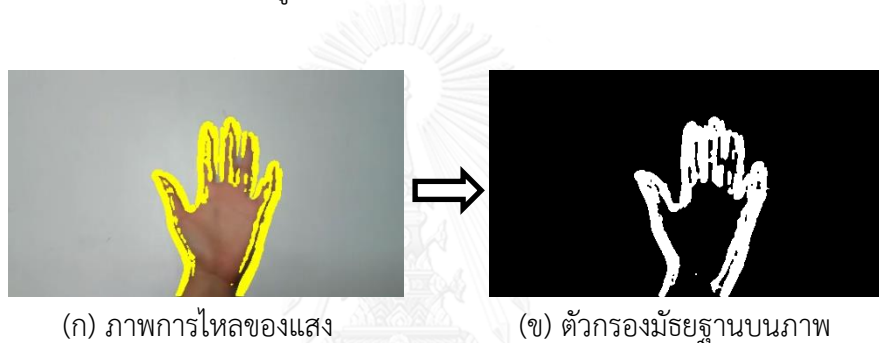
ค่ามัธยฐาน คือค่ากลางของข้อมูลที่ได้จากการพิจารณาค่าแห่งของข้อมูลที่อยู่ตรงกลางโดยที่ข้อมูลต้องทำการเรียงลำดับตามปริมาณจากมากไปน้อย หรือจากน้อยไปมากก็ได้ และค่ามัธยฐานยังสามารถใช้เป็นตัวแทนของข้อมูลได้เป็นอย่างดี ในกรณีที่ข้อมูลมีการกระจายที่ผิดปกติ ซึ่งอาจเกิดจากการที่มีข้อมูลบางตัวมีค่ามากหรือน้อยจนผิดปกติ [28]

ในระบบนี้หลังจากได้ค่าเฉลี่ยของแต่ละช่องในลำดับเวลาของเฟรมภาพ เพื่อตรวจสอบการกระจายตัวของข้อมูล กระบวนการกรองค่ามัธยฐานจะทำการคำนวณค่ามัธยฐานจากการเคลื่อนที่บนเฟรมภาพสองมิติตามแต่ละช่องอย่างอิสระตลอดเวลาโดยใช้วิธีการเลื่อนหน้าต่างตามทีแสดงในรูปที่

3.18 วิธีนี้หน้าต่างที่มีความยาวตามที่ระบุจะเคลื่อนที่ไปตามตัวอย่างของแต่ละช่องและคำนวณค่ามัธยฐานของข้อมูลในแต่ละช่อง [29] ผลการใช้การกรองค่ามัธยฐานแสดงในรูปที่ 3.19



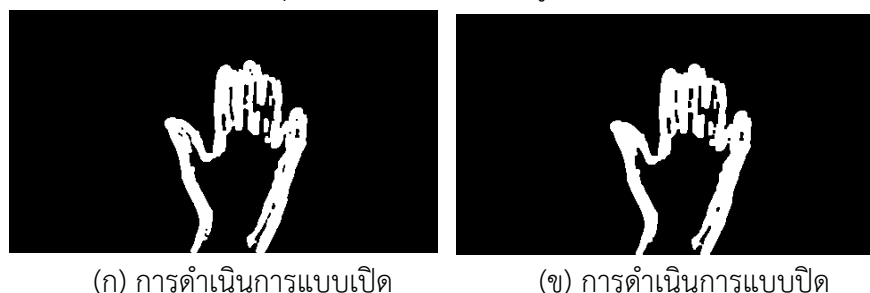
รูปที่ 3.18 วิธีการเลื่อนหน้าต่าง



รูปที่ 3.19 ตัวกรองมัธยฐานบนภาพทำกำมือ

### 3.2.2.7 การปรับปรุงโครงร่างภาพ

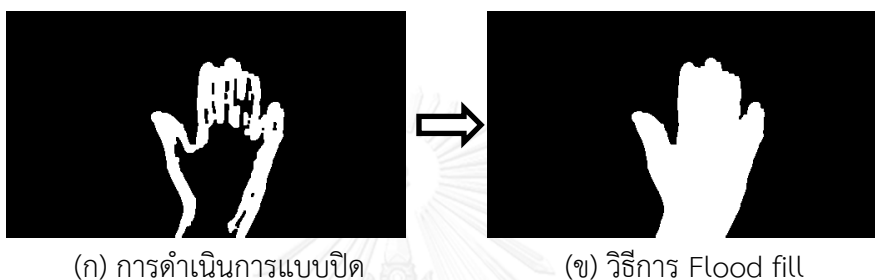
สำหรับการดำเนินการแบบเปิด (Morphological open) จะดำเนินการโดยใช้วิธีการการกัดกร่อนตามด้วยวิธีการขยายโดยใช้องค์ประกอบโครงสร้างเดียวกันสำหรับการดำเนินงานทั้งสอง สำหรับการดำเนินการแบบปิด (Morphological close) จะใช้สำหรับการดึงบริเวณที่สนใจและลบสัญญาณรบกวนออกจากภาพ การดำเนินการแบบปิดประกอบด้วยวิธีการการขยายตามด้วยวิธีการกัดกร่อน ผลลัพธ์ของการใช้การปรับปรุงโครงร่างภาพแสดงในรูปที่ 3.20



รูปที่ 3.20 กระบวนการปรับปรุงโครงร่างภาพ

จากรูปที่ 3.20 พบว่าเส้นสีขาวรอบ ๆ มือของรูป (b) มีความหนามากกว่ารูป (a) นั้นหมายความว่ากระบวนการแบบปิดให้ผลลัพธ์ที่ดีสำหรับการเติมช่องภายในเส้นขอบของมือและกำจัดสัญญาณรบกวน สำหรับบริเวณสีดำภายในมือจะสามารถเติมเต็มได้ด้วยวิธีการ Flood fill

วิธีการ Flood fill คือการเติมหลุมในภาพ วิธีนี้ใช้ได้ทั้งภาพไบนารีและภาพระดับเทา สำหรับภาพไบนารีจะใช้สิ่งที่เกี่ยวข้องกับขอบออกจากภาพ ในงานนี้ใช้เพื่อเติมช่องภายในมือนี่แสดงในรูปที่ 3.21 ดังนั้นจึงทำการเปลี่ยนพิกเซลฉากหลังหรือบริเวณสีดำเป็นพิกเซลฉากหน้าหรือบริเวณสีขาวตามเส้นขอบของมือ การดำเนินการนี้จะหยุดก็ต่อเมื่อเติมช่องจนถึงเส้นขอบของวัตถุ



(ก) การดำเนินการแบบปิด

(ข) วิธีการ Flood fill

รูปที่ 3.21 กระบวนการ Flood fill

### 3.2.2.8 การวิเคราะห์หีบลิบ

สำหรับการวิเคราะห์หีบลิบต้องใช้หลักการเลือกขนาดเพื่อหาขนาดของพื้นที่เฉพาะที่มีความแปรปรวนร่วมกับการแปลงภาพ ในการเลือกเชิงพื้นที่นั้นขนาดของการตอบสนอง Laplacian จะมีค่าสูงสุดที่จุดศูนย์กลางของหีบลิบโดยที่ขนาดของ Laplacian และหีบลิบตรงกัน และจากการที่ Laplacian ของเกาส์เซียนเป็นตัวดำเนินการสมมาตรแบบวงกลมสำหรับการตรวจจับหีบลิบในปริภูมิสองมิติ สามารถเขียนสมการได้ดังนี้

$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \quad (3.22)$$

สำหรับการทำ normalization นั้นมาตราส่วนการตอบสนองของอนุพันธ์ของตัวกรองแบบเกาส์เซียนกับขั้นตอนที่สมบูรณ์แบบจะมีค่าลดลงเมื่อเพิ่ม  $\sigma$  มากขึ้น เพื่อให้ได้การตอบสนองเหมือนกัน (scale-invariant) จะต้องทำการคูณอนุพันธ์เกาส์เซียนโดยใช้  $\sigma$  ทั้งนี้ Laplacian เป็นอนุพันธ์เกาส์เซียนอันดับที่สอง ดังนั้นจึงต้องคูณด้วย  $\sigma^2$

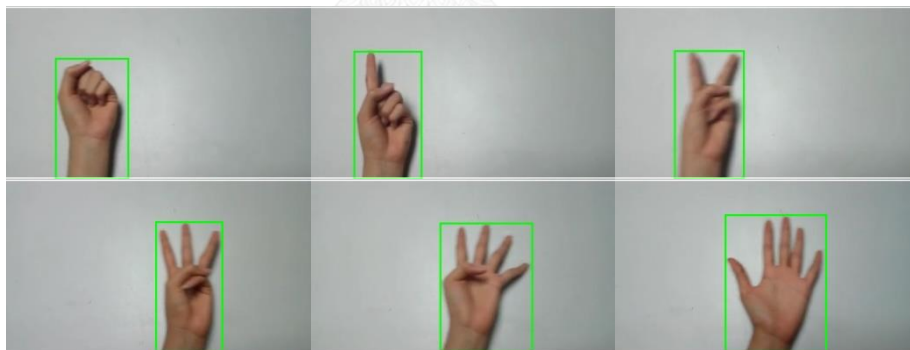
$$\nabla^2_{norm} g = \partial^2 \left( \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \right) \quad (3.23)$$

เพื่อให้ได้คำตอบสนองสูงสุด ค่าศูนย์ของ Laplacian ต้องสอดคล้องกับวงกลม การกำหนด Laplacian ขึ้นอยู่กับขนาด ดังแสดงในสมการที่ 3.24

$$(x^2 + y^2 - 2\sigma^2) e^{-(x^2+y^2)/2\sigma^2} \quad (3.24)$$

ดังนั้นการตอบสนองสูงสุดเกิดขึ้นที่  $\sigma = r/\sqrt{2}$  สำหรับการตรวจจับล็อบเชิงพื้นที่พบว่า มีสองขั้นตอน ขั้นตอนแรกรวมภาพที่มีการ normalize ขนาดของ Laplacian ในหลาย ๆ ขนาด จากนั้นหาค่าสูงสุดของการตอบสนอง Laplacian สี่เหลี่ยมในเชิงพื้นที่

ดังนั้นระบบจึงสามารถติดตามการเคลื่อนที่ของท่ามือได้โดยใช้การตรวจจับจากการไหลของแสงและกระบวนการการปรับโครงสร้างแบบการดำเนินการแบบปิดตั้งได้อธิบายไว้ในขั้นตอนข้างต้น และการติดตามการเคลื่อนที่ด้วยการวิเคราะห์ล็อบดังแสดงในรูปที่ 3.22 ขั้นตอนต่อไปคือการกำหนดความหมายสำหรับแต่ละท่ามือเพื่อเป็นภาษาในการสื่อความหมาย ได้แก่ "ช่วยด้วย", "น้ำ", "อาหาร", "ห้องน้ำ", "อำนวยความสะดวก" และ "อุบัติเหตุ" ตามลำดับ



รูปที่ 3.22 ภาพท่ามือแบบเคลื่อนไหว 6 ท่าจากกระบวนการการวิเคราะห์ล็อบ

### 3.2.2.9 วิธีการ SURF

กำหนดให้ภาพ  $I$  มีจุด  $x = (x, y)$  และเมทริกซ์ Hessian คือ  $H(x, \sigma)$  ใน  $x$  ที่ขนาดเท่ากับ  $\sigma$  เมื่อ  $L_{xy}(x, \sigma)$  คือ การรวมกันของอนุพันธ์ลำดับที่สองแบบเกาส์เซียน ดังแสดงในสมการที่ 3.25 และ 3.26 ดังนี้

$$I(x) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(x, y) \quad (3.25)$$

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \quad (3.26)$$

จากนั้นทำการดำเนินการกับขนาดที่เล็กที่สุด โดยกำหนดตัวกรองสี่เหลี่ยมให้มีขนาด  $9 \times 9$  สำหรับอนุพันธ์ลำดับที่สองแบบเกาส์เซียน และให้  $\sigma$  มีค่าเท่ากับ 0.9 ดังนี้

$$\det(H_{\text{approx}}) = D_{xx}D_{yy} - (0.9D_{xy})^2 \quad (3.27)$$

หลังจากการใช้ตัวกรองสี่เหลี่ยมขนาด  $9 \times 9$  แล้ว ทำการใช้ตัวกรองขนาดใหญ่ขึ้นในเลขเอร์ถัดมา อาทิ การใช้ตัวกรองขนาด  $15 \times 15$  และ  $21 \times 21$  ซึ่งค่าสูงสุดของตัวกำหนด Hessian จะถูกสอดแทรกในส่วนพื้นที่และขนาดของภาพ

สำหรับคำอธิบายพื้นที่ใกล้เคียง (Local neighborhood description) ตัวคำอธิบาย SURF (SURF descriptor) จะแสดงคำอธิบายความทนทาน (Robust description) สำหรับคุณลักษณะของภาพ โดยจะอธิบายความเข้มของพิกเซลรอบจุดที่สนใจ โดยตัวอธิบาย (Description) ประกอบด้วย การกำหนดทิศทางของจุดสนใจ (Orientation assignment) เพื่อปรับปรุงอัลกอริธึมให้มีความทนทานต่อการหมุนมากขึ้น

ในการกำหนดทิศทางของจุดที่สนใจนั้น (Orientation assignment) จะทำการกำหนดทิศทางที่สอดคล้องกับจุดที่สนใจตามคุณสมบัติของรูปภาพในพื้นที่ (local image) โดยทิศทางที่ทำซ้ำได้จะถูกกำหนดด้วยการตอบสนองของ Haar-wavelet ในระนาบ x-y ตามข้อมูลจากพื้นที่วงกลมใกล้เคียงรอบ ๆ จุดที่สนใจ การกำหนดทิศทางสามารถคำนวณได้จากผลรวมของการตอบสนองทั้งในแกน x และแกน y ในหน้าต่างบานเลื่อนที่มีขนาดเท่ากับ  $\pi / 3$

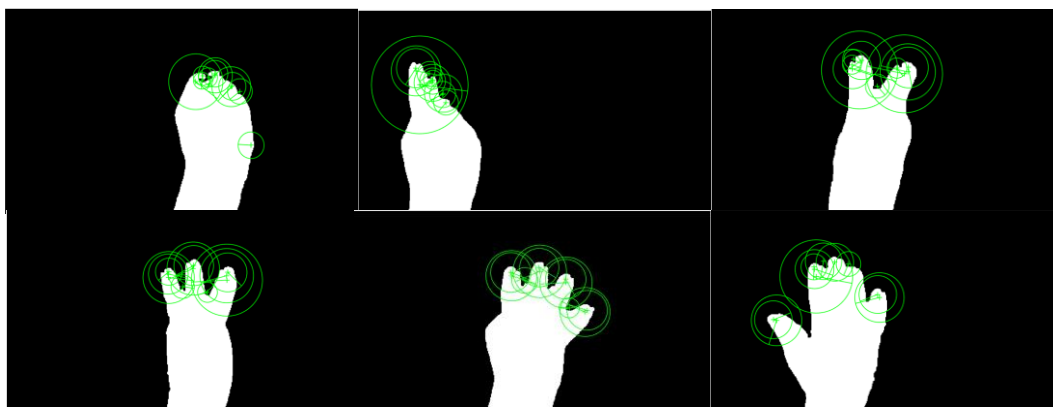
หลังจากรวมค่าการตอบสนองทั้งสองค่าแล้ว ทำการระบุเวกเตอร์กำหนดตำแหน่งพื้นที่ จากนั้นจะต้องมีการอธิบายโดยการสร้างพื้นที่สี่เหลี่ยมจัตุรัสรอบจุดสนใจและกำหนดทิศทาง ในขั้นตอนนี้บริเวณสี่เหลี่ยมจัตุรัสจะถูกลดขนาดลงเหลือเป็นบริเวณย่อย  $4 \times 4$  ทั้งนี้การตอบสนองของ Haar-wavelet สำหรับทิศทาง x ( $d_x$ ) และ y ( $d_y$ ) ใช้เพื่อกำหนดความสัมพันธ์ของการวางทิศทางของจุดที่สนใจ ทั้งสองพารามิเตอร์  $d_x$  และ  $d_y$  จะถูกถ่วงน้ำหนักเป็นอันดับแรกด้วย Gaussian เพื่อปรับปรุงความทนทานของการแปลงรูปทรงเรขาคณิต และรวมแต่ละบริเวณย่อยเข้าด้วยกันเพื่อแสดงเวกเตอร์คุณลักษณะ ตัวบอกคำอธิบายแบบสี่มิติสำหรับเวกเตอร์  $v$  สามารถแสดงได้ดังสมการนี้



$$v = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|) \quad (3.28)$$

สมการที่ 3.28 แสดงบริเวณย่อยแบบสี่เหลี่ยมทั้งหมด  $4 \times 4$  ของความยาวเท่ากับ 64 ทั้งนี้พบว่าอัลกอริธึมแสดงความทนทานต่อการ offset และ contrast ได้

สำหรับการจับคู่ (Matching) อัลกอริธึมทำการเปรียบเทียบคำอธิบายที่ได้รับจากภาพที่แตกต่างกัน เมื่อคู่ที่นำมาเปรียบเทียบเหมือนกัน อัลกอริธึมจะจับคู่ขึ้น ๆ และแสดงผลลัพธ์ออกมาดังรูปที่ 3.23 พบว่าเส้นวงกลมที่ล้อมรอบปลายนิ้วมือคือการจับคู่ที่เหมือนกันของอัลกอริธึม SURF



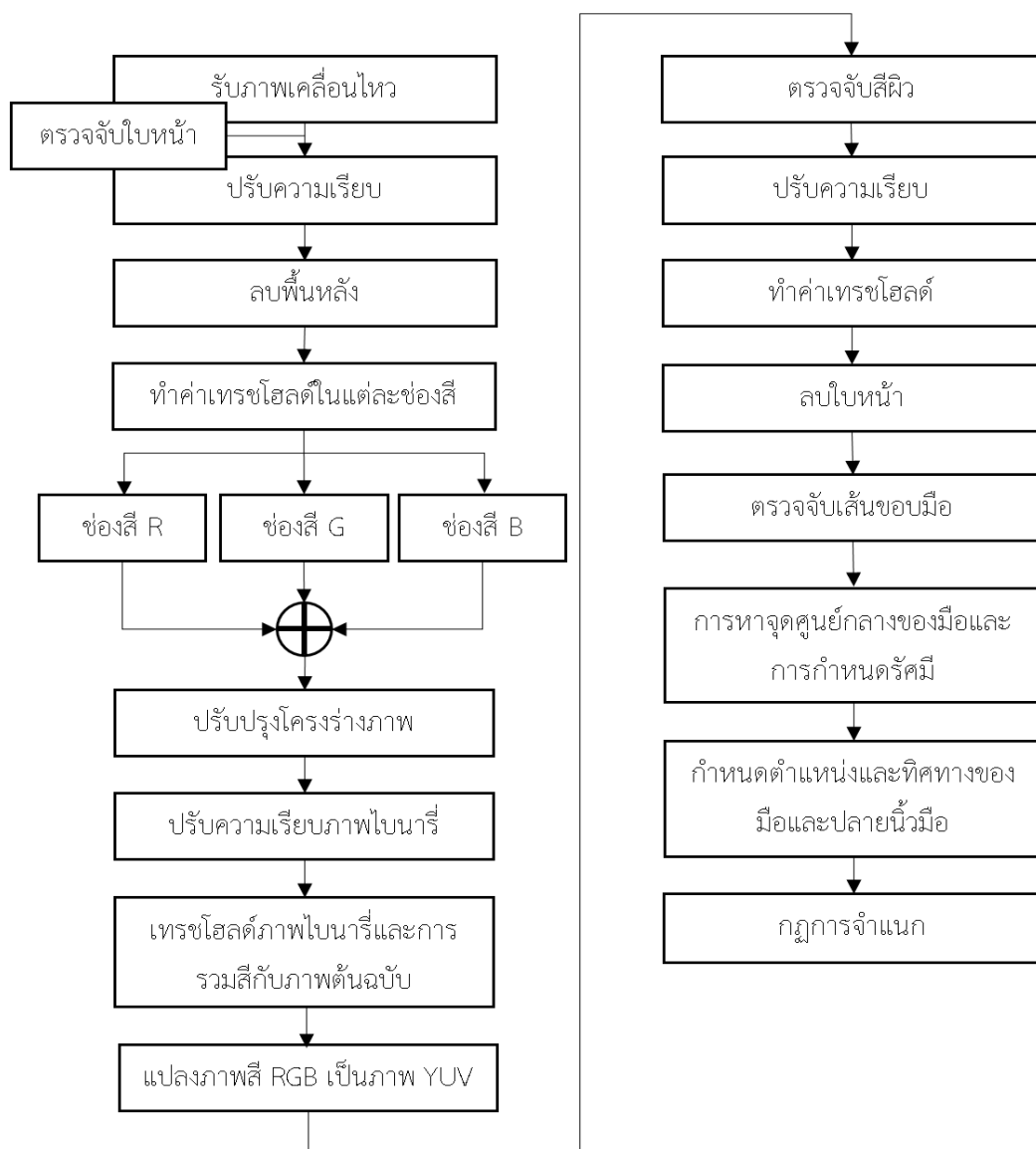
รูปที่ 3.23 การจับคู่จุดที่สนใจของท่ามือเคลื่อนไหวโดยใช้อัลกอริธึม SURF

#### 3.2.2.10 แสดงผลลัพธ์

หลังจากระบบจับคู่ภาพที่เหมือนกันแล้ว ระบบจะแสดงผลลัพธ์ท่ามือเคลื่อนไหวที่มีการแสดงความหมายไปพร้อม ๆ กัน ดังแสดงผลลัพธ์ในหัวข้อที่ 4.2

### 3.2.3 อัลกอริทึมที่นำเสนอสำหรับระบบรู้จำท่ามือบนพื้นฐานการมองเห็นสำหรับผู้สูงอายุ

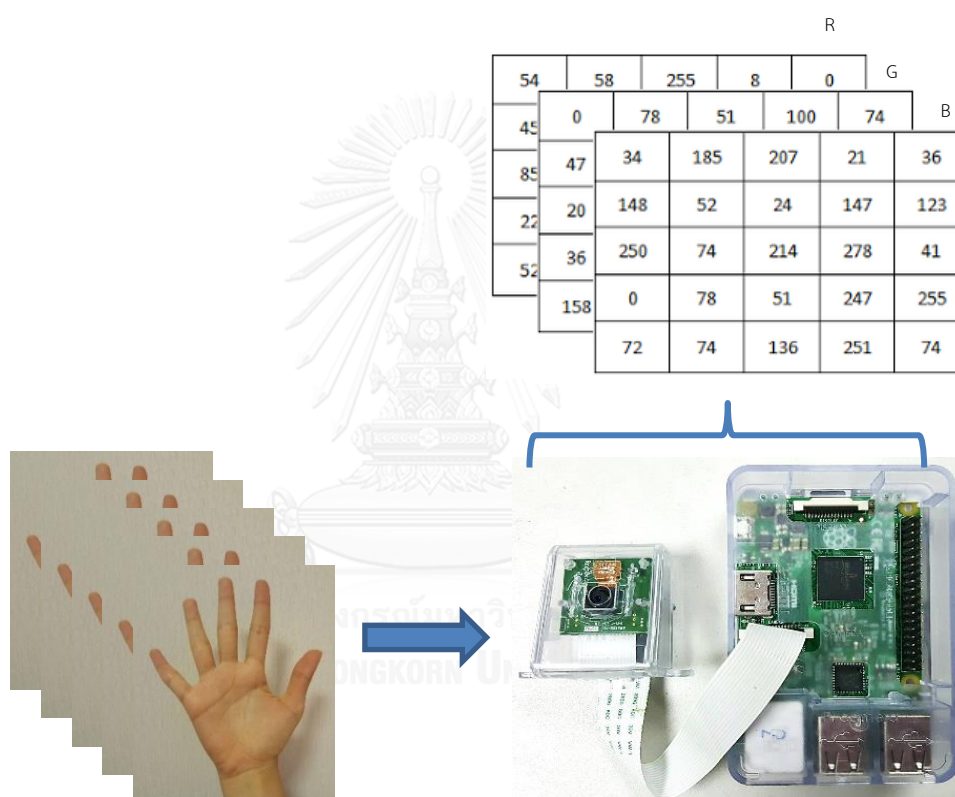
ในหัวข้อนี้ผู้วิจัยนำเสนออัลกอริทึมสำหรับรู้จำท่ามือบนพื้นฐานการมองเห็นของผู้สูงอายุที่มีความทนทานต่อการเปลี่ยนแปลงของสภาพแสง ระยะห่างระหว่างกล้องและมือ และความซับซ้อนของพื้นหลัง นอกจากนี้ยังรวมวิธีการบางวิธีที่น่าสนใจของอัลกอริทึมในหัวข้อ 3.2.1 และ 3.2.2 ดังที่ได้อธิบายข้างต้นเพื่อนำมาประยุกต์ใช้ให้ระบบสามารถรู้จำท่ามือแบบเคลื่อนไหวได้อย่างแม่นยำและทนทานต่อสัญญาณรบกวนมากขึ้นเพื่อนำมาประยุกต์ใช้ให้ระบบมีความแม่นยำและความทนทานมากขึ้น ขั้นตอนการทำงานของระบบแบ่งออกเป็น 17 ขั้นตอนดังแสดงในรูปที่ 3.24



รูปที่ 3.24 ขั้นตอนการทำงานของระบบรู้จำท่ามือแบบเคลื่อนไหว

### 3.2.3.1 การรับภาพเคลื่อนไหว

สำหรับอัลกอริทึมที่นำเสนอสำหรับระบบรู้จำท่ามือบนพื้นฐานการมองเห็นสำหรับผู้สูงอายุ นั้น ระบบรับภาพเคลื่อนไหวได้จากโมดูลกล้อง V2 ของบอร์ด Raspberry PI 3 model B แบบเวลาจริง (Real time) โดยตั้งค่าเริ่มต้นความละเอียดกล้องที่ 800x608 พิกเซล กำหนดอัตราเฟรมเท่ากับ 20 เฟรมต่อวินาที ขนาดของเฟรมภาพให้มีขนาดเท่ากับความคมชัดของโมดูลกล้อง V2 ที่ใช้บันทึก ซึ่งระบบจะทำการเก็บเฟรมภาพสี RGB หรือเฟรมต้นฉบับที่มีลักษณะเป็นเฟรมมาต่อกันหลายๆเฟรมไว้ในอาร์เรย์เพื่อประมวลผลต่อไปดังแสดงในรูปที่ 3.25



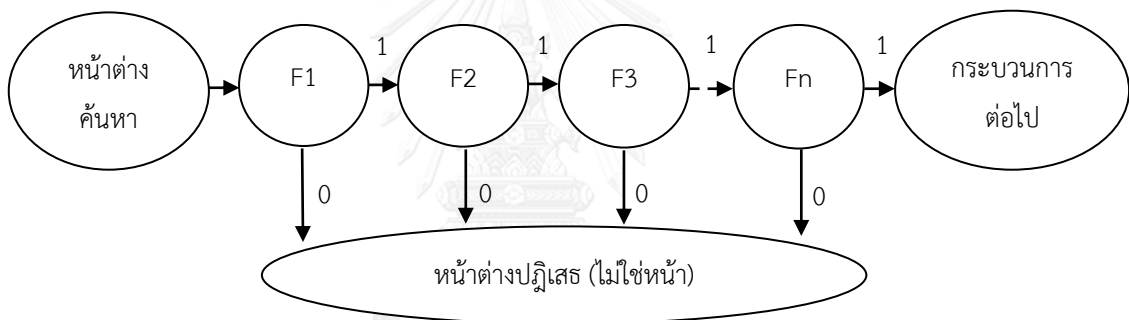
รูปที่ 3.25 แผนผังแสดงการรับภาพท่ามือแบบเคลื่อนไหวจากระบบ

### 3.2.3.2 การตรวจจับใบหน้า

หลังจากขั้นตอนการนำเข้าไฟล์วิดีโอและเก็บเฟรมภาพไว้ในอาร์เรย์แล้ว วิธีการนี้ระบบจะทำการตรวจจับใบหน้า เนื่องจากการทำงานของระบบในหัวข้อ 3.2.1 และ 3.2.2 จะมีปัญหาในการตรวจจับเมื่อใบหน้าอยู่ในเฟรมภาพร่วมกับท่ามือ ดังนั้นจึงการตรวจจับใบหน้าก่อนเพื่อให้ระบบรู้จักใบหน้ามนุษย์ สามารถตรวจจับและคัดแยกใบหน้าออกจากเฟรมนั้นๆได้ เทคนิคที่ใช้สำหรับการตรวจจับใบหน้าคือ Haar like feature พัฒนาขึ้นโดย Viola และ Jones ในกระบวนการตรวจจับ

ของ Haar like feature ทำงานโดยปรับขนาดขนาดของภาพตัวอย่างแบบ Positive เท่ากับ 30x30 พิกเซลในทุกๆเฟรม และปรับขนาดของภาพตัวอย่างแบบ negative ให้มีขนาดเท่ากัน หลังจากนั้นนำตัวจำแนกเหล่านี้ไปใช้ในพื้นที่ที่สนใจ นั่นคือ บริเวณใบหน้ามนุษย์ ซึ่งผลลัพธ์จะแสดงค่าเท่ากับ 1 เมื่อพื้นที่นั้น ๆ มีใบหน้าอยู่ และจะแสดงค่าเท่ากับ 0 เมื่อพื้นที่นั้น ๆ ไม่มีใบหน้า

การทำงานของตัวจำแนกจะทำงานร่วมกับหน้าต่างค้นหา (Search window) เพื่อตรวจสอบหาใบหน้าของพิกเซลในภาพทั้งภาพสำหรับทุกๆเฟรม (F1, F2, F3,...,Fn) และทำการปรับขนาดของพื้นที่ที่สนใจจากขนาดที่แตกต่างกันของภาพต่างๆ ดังนั้นกระบวนการนี้จะดำเนินการซ้ำหลายรอบ และทดสอบกับขนาดที่แตกต่างกัน เพื่อที่จะสามารถระบุพื้นที่ใบหน้าในภาพที่ไม่สามารถระบุขนาดได้ในตอนแรกนั่นเอง กระบวนการสำหรับการรวมตัวจำแนกกลุ่มแบบต่อเรียง (Cascade classification) แบ่งเป็นขั้นตอนต่าง ๆ สำหรับการตัดสินใจ ซึ่งจะปฏิเสธภาพที่เข้ามาที่มีลักษณะไม่เหมือนกับใบหน้า แต่ถ้าภาพที่เข้ามามีลักษณะเหมือนกับใบหน้า ระบบจะยอมให้ผ่านไปยังขั้นตอนต่อไปดังแสดงในรูปที่ 3.26



รูปที่ 3.26 การทำงานของการรวมตัวจำแนกกลุ่มแบบต่อเรียง

### 3.2.3.3 ปรับความเรียบ

จากขั้นตอนข้างต้นพบว่าภาพที่ได้จากกล้องมีสัญญาณรบกวนอยู่ในเฟรมซึ่งส่งผลต่อระดับความเข้มของภาพ โดยสัญญาณรบกวนนี้เกิดจากสภาพแสงเงา ในสภาพแวดล้อมนั้น ๆ ดังนั้นวิธีการปรับความเรียบ (Smoothing) หรือการเบลอภาพ (Blurring) จึงมีความสำคัญเพื่อใช้ลดสัญญาณรบกวน และทำให้ภาพและเส้นขอบมีความเรียบเนียนมากขึ้น

กำหนดความกว้างและความสูงของเคอร์เนลเกาส์เซียน (Gaussian kernel) ให้มีค่าเป็นบวก และขนาดเป็นเลขจำนวนคี่ เกาส์เซียนแบบสองมิติสามารถเขียนได้ดังนี้

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3.29)$$

เมื่อ  $G(x, y)$  คือผลลัพธ์เกาส์เซียนแบบสองมิติ

$\sigma$  คือค่าความปรวนแปร สำหรับทุกตัวแปร  $x$  และ  $y$

เมื่อขนาดของภาพเท่ากับ  $800 \times 608$  กำหนดให้ความกว้างแทนด้วย  $N$  และความยาวแทนด้วย  $M$  กำหนดให้ขนาดของเคอร์เนลที่ใช้ในภาพมีขนาดเท่ากับ  $(2m+1) \times (2m+1)$  ค่าเกาส์เซียนนี้สามารถคำนวณได้จากขนาดของเคอร์เนล ดังนี้

$$\sigma = 0.3 * ((n - 1) * 0.5 - 1) + 0.8 \quad (3.30)$$

เมื่อ  $n = 2m + 1$  คือขนาดของเกาส์เซียน

ซึ่งภาพนำเข้าที่ได้รับจากการเบลอภาพโดยใช้ตัวกรองเกาส์เซียนแสดงรูปที่ 3.27 ดังนี้



(ก) เฟรมที่มีใบหน้า

(ข) เฟรมที่ไม่มีใบหน้า

รูปที่ 3.27 การปรับภาพเรียบหลังจากการรับภาพจากกล้อง

จากรูปที่ 3.27 พบว่าภาพเฟรมทั้งแบบมีใบหน้าและไม่มีใบหน้าจะมีความเบลอเล็กน้อย เพื่อให้กำจัดสัญญาณรบกวนให้มีประสิทธิภาพมากขึ้น ผู้วิจัยจึงใช้วิธีการปรับความเรียบในแต่ละองค์ประกอบสีของโมเดลสี RGB การแยกองค์ประกอบสี R G และ B สำหรับการปรับความเรียบแสดงดังรูปที่ 3.28



(ก) เฟรมที่มีใบหน้าในช่องสี R (ข) เฟรมที่มีใบหน้าในช่องสี G (ค) เฟรมที่มีใบหน้าในช่องสี B



(ง) เฟรมที่ไม่มีใบหน้าในช่องสี R (จ) เฟรมที่ไม่มีใบหน้าในช่องสี G (ฉ) เฟรมที่ไม่มีใบหน้าในช่องสี B

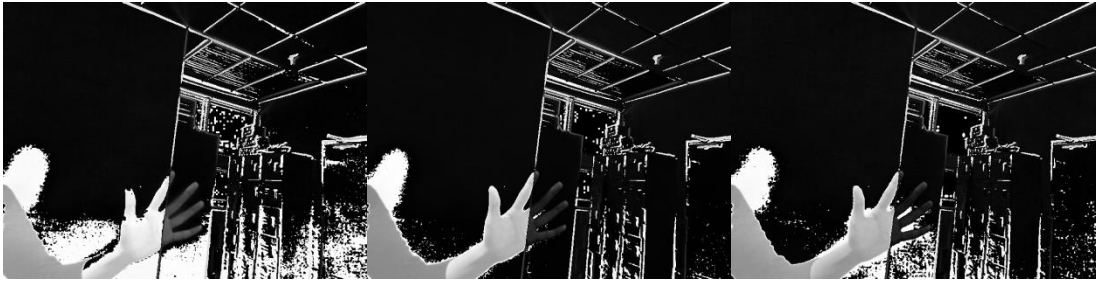
รูปที่ 3.28 แสดงการปรับภาพเรียบของแต่ละช่องสีของโมเดล RGB

#### 3.2.3.4 การลบพื้นหลัง

ในระบบนี้วิธีการลบพื้นหลังทำงานบนโมเดลสี RGB โดยคำนวณความแตกต่างของเฟรมจากการเคลื่อนที่ของมือกับภาพพื้นหลังที่ไม่เคลื่อนที่ ในการดำเนินงานสามารถแบ่งวิธีการลบพื้นหลังออกเป็น 2 แบบ คือ การลบพื้นหลังแบบมีใบหน้าที่ฉากหลัง และการลบพื้นหลังแบบไม่มีใบหน้าที่ฉากหลัง ทั้งนี้จากขั้นตอนการปรับความเรียบของภาพที่มีการแยกองค์ประกอบสีในการประมวลผล ดังนั้นในขั้นตอนนี้จึงทำการลบพื้นหลังในแต่ละองค์ประกอบสีเช่นกัน ผลลัพธ์ที่ได้จากการลบพื้นหลังบนองค์ประกอบสี R G และ B แสดงในรูปที่ 3.29



(ก) เฟรมที่มีใบหน้าในช่องสี R (ข) เฟรมที่มีใบหน้าในช่องสี G (ค) เฟรมที่มีใบหน้าในช่องสี B



(ง) เฟรมที่ไม่มีใบหน้าในช่องสี R (จ) เฟรมที่ไม่มีใบหน้าในช่องสี G (ฉ) เฟรมที่ไม่มีใบหน้าในช่องสี B  
รูปที่ 3.29 การลบพื้นหลังในแต่ละองค์ประกอบสี

จากรูปที่ 3.29 พบว่าการลบพื้นหลังในช่องสี R G และ B ของเฟรมที่มีใบหน้าจะมีสัญญาณรบกวนมากกว่าเฟรมที่ไม่มีใบหน้า ทั้งนี้เกิดจากเฟรมที่มีใบหน้านั้นมีการเคลื่อนที่ของฉากหลังด้วยเล็กน้อย กล่าวคือบริเวณศรีษะและลำตัวมีการเคลื่อนที่ ส่งผลให้สภาพแสงเปลี่ยนไป ดังนั้นจะเห็นว่ามีจุดสีขาวเกิดขึ้นเป็นจำนวนมากเมื่อเทียบกับเฟรมที่ไม่มีใบหน้า อย่างไรก็ตามระบบสามารถหาบริเวณของมือซึ่งเป็นส่วนที่เคลื่อนไหวได้ทั้งหมดไม่ว่าฉากหลังจะมีความซับซ้อนหรือไม่ก็ตาม

### 3.2.3.5 การทำเทรชโฮลด์

การทำเทรชโฮลด์คือการตั้งค่าเพื่อแยกพื้นหลังออกจากวัตถุ ค่าเทรชโฮลด์ทำให้ระบบสามารถแยกมือออกจากพื้นหลังได้ โดยจะให้ภาพไบนารี หรือภาพขาวดำสองระดับ (Binarization) โดยพิกเซลสีขาว คือส่วนของมือ และพิกเซลสีดำ คือส่วนพื้นหลัง และเพื่อให้ได้ผลลัพธ์ที่ดีนั้นผู้วิจัยจึงวิเคราะห์ในแต่ละองค์ประกอบสี RGB โดยทำการกำหนดค่าเทรชโฮลด์แยกไปในแต่ละองค์ประกอบสี R G และ B ค่าเทรชโฮลด์ที่กำหนดแสดงดังนี้

$$31 \leq R \leq 238$$

$$16 \leq G \leq 238$$

$$16 \leq B \leq 238$$

$$\text{เมื่อ } R,G,B = [0,255]$$



(ก) เฟรมที่มีใบหน้าในช่องสี R (ข) เฟรมที่มีใบหน้าในช่องสี G (ค) เฟรมที่มีใบหน้าในช่องสี B



(ง) เฟรมที่ไม่มีใบหน้าในช่องสี R (จ) เฟรมที่ไม่มีใบหน้าในช่องสี G (ฉ) เฟรมที่ไม่มีใบหน้าในช่องสี B  
รูปที่ 3.30 การทำเทรซโฮลต์ในแต่ละองค์ประกอบสี

จากรูปที่ 3.30 พบว่าการทำเทรซโฮลต์ในช่องสี R G และ B ของเฟรมที่มีใบหน้า (ก) (ข) และ (ค) จะมีเส้นสีขาวบางเกิดขึ้นตามรูปร่างมนุษย์เนื่องจากการเคลื่อนที่ของบริเวณศีรษะและลำตัวเล็กน้อย สำหรับเฟรมที่ไม่มีใบหน้าในช่องสี R G และ B (ง) (จ) และ (ฉ) จะเห็นว่ามีเส้นสีขาวบาง ๆ ตามรูปร่างของวัตถุในฉากหลัง โดยเส้นเหล่านี้เกิดจากสภาพแสงที่เปลี่ยนไปจากแสงภายนอก อย่างไรก็ตามพบว่าบริเวณที่ชัดเจนมากที่สุดในเฟรมทั้งสองแบบคือบริเวณมือ

หลังจากได้ส่วนฉากหน้าหรือมือในแต่ละองค์ประกอบสีแล้ว ระบบทำการรวมองค์ประกอบสีทั้งสามสีเข้าด้วยกันอีกครั้งโดยใช้กระบวนการแบบรวม (Addition operation) เพื่อให้ได้ผลลัพธ์ที่ดีที่สุดสำหรับการประมวลผลในขั้นตอนต่อไป ผลลัพธ์การรวมช่องสี R G และ B ที่ผ่านการทำเทรซโฮลต์แสดงในรูปที่ 3.31



(ก) เฟรมที่มีใบหน้า

(ข) เฟรมที่ไม่มีใบหน้า

รูปที่ 3.31 เฟรมภาพที่ได้จากการรวมองค์ประกอบสี R G และ B

จากรูปที่ 3.31 พบว่าหลังจากการรวมกันของช่องสี R G และ B นั้น จะได้บริเวณมือที่ชัดเจนทั้งรูปที่มีใบหน้า (ก) และไม่มีใบหน้า (ข) โดยที่ไม่ได้รับผลกระทบจากความซับซ้อนของพื้นหลังและ



สภาพแสงที่เปลี่ยนไป ดังพิจารณาได้จากรูปที่ไม่มีใบหน้า (ข) ว่าเส้นตรงจากฉากหลังไม่แสดงชั้นที่ ส่วนของมือเหมือนในขั้นตอนการทำเทรซโฮลด์ในแต่ละองค์ประกอบสี

### 3.2.3.6 การเปลี่ยนแปลงลักษณะรูปร่างหรือโครงร่างของภาพ

ขั้นตอนนี้จะทำการเปลี่ยนแปลงลักษณะรูปร่างของวัตถุหรือโครงร่างของภาพ (Morphology operations) สำหรับกำจัดสัญญาณรบกวน และสกัดบริเวณที่สนใจซึ่งก็คือมือนั่นเอง ในขั้นตอนแรกผู้วิจัยใช้วิธีการกร่อนขนาด (Erosion) สำหรับการกำจัดสัญญาณรบกวน ในกระบวนการ Erosion เมื่อจุดพิกเซลทั้งหมดของมือภายใต้ Kernel มีค่าเท่ากับ 1 ระบบจะพิจารณาค่าพิกเซลของมือหรือบริเวณสีขาวในภาพต้นฉบับให้มีค่าเท่ากับ 1 ส่วนค่าอื่น ๆ ที่ไม่ใช่ 1 จะถูกตัดออก

จากนั้นจึงใช้วิธีการขยาย (Dilation) เพื่อขยายขนาดเส้นขอบของมือเมื่อมือน้อยหนึ่งพิกเซลภายใต้ Kernel ที่มีค่าเท่ากับ 1 โดยกระบวนการ Dilation นั้นนอกจากจะขยายเส้นขอบของมือแล้ว ยังเปลี่ยนพิกเซลสีดำเป็นพิกเซลสีขาวเพื่อเติมเต็มบริเวณที่ต้องการ ทำให้ง่ายต่อการระบุท่ามือ กระบวนการที่มีลำดับการดำเนินการเริ่มต้นด้วยวิธีการกร่อนขนาด (Erosion) และตามด้วยวิธีการขยาย (Dilation) เรียกว่ากระบวนการแบบเปิด (Opening operation) สมการการดำเนินการแบบเปิดแสดงดังนี้

$$I \circ H = (I \ominus H) \oplus H \quad (3.31)$$

อย่างไรก็ตามผลลัพธ์ที่ได้ยังคงมีสัญญาณรบกวนอยู่และยังมีบริเวณจุดสีดำอยู่ภายในบริเวณที่ต้องการดังแสดงในรูปที่ 3.32(ก) และรูปที่ 3.32(ค) ดังนั้นผู้วิจัยจึงใช้วิธีการแบบปิด (Closing operation) ดำเนินการโดยวิธีการขยาย (Dilation) ก่อนและจึงใช้วิธีการกร่อนขนาด (Erosion) เพื่อกร่อนขนาดให้เส้นขอบมีขนาดเล็กลง ซึ่งจะได้เส้นโค้งมือที่มีความคมชัดและเรียบเนียนมากขึ้น รวมถึงได้พื้นที่ภายในมือแบบเต็มพื้นที่ดังแสดงในรูปที่ 3.32(ข) และรูปที่ 3.32(ง)

สมการการดำเนินการแบบปิดแสดงดังนี้

$$I \bullet H = (I \oplus H) \ominus H \quad (3.32)$$

การระบุขนาดของ kernel สำหรับวิธีการกร่อนขนาด (Erosion) และวิธีการขยาย (Dilation) สำหรับ 2D convolution กำหนดให้ขนาดเท่ากับ 3x3



(ก) กระบวนการแบบเปิดของเฟรมที่มีใบหน้า (ข) กระบวนการแบบปิดของเฟรมที่มีใบหน้า



(ค) กระบวนการแบบเปิดของเฟรมที่ไม่มีใบหน้า (ง) กระบวนการแบบปิดของเฟรมที่ไม่มีใบหน้า

### รูปที่ 3.32 การเปลี่ยนแปลงลักษณะรูปร่างหรือโครงร่างของภาพ

#### 3.2.3.7 ปรับความเรียบภาพไบนารี

หลังจากได้รับภาพไบนารีที่ผ่านการปรับปรุงโครงร่างภาพแล้ว เพื่อให้เส้นขอบภายในภาพมีความเรียบเนียนมากขึ้น จึงใช้วิธีการปรับภาพแบบเกาส์เซียนในการเบลอภาพอีกครั้ง ดังแสดงภาพผลลัพธ์ในรูปที่ 3.33



(ก) เบลอเฟรมที่มีใบหน้า

(ข) เบลอเฟรมที่ไม่มีใบหน้า

### รูปที่ 3.33 การเบลอภาพไบนารี

### 3.2.3.8 การทำเทรซโฮลด์ภาพไบนารีและการรวมสีกับภาพต้นฉบับ

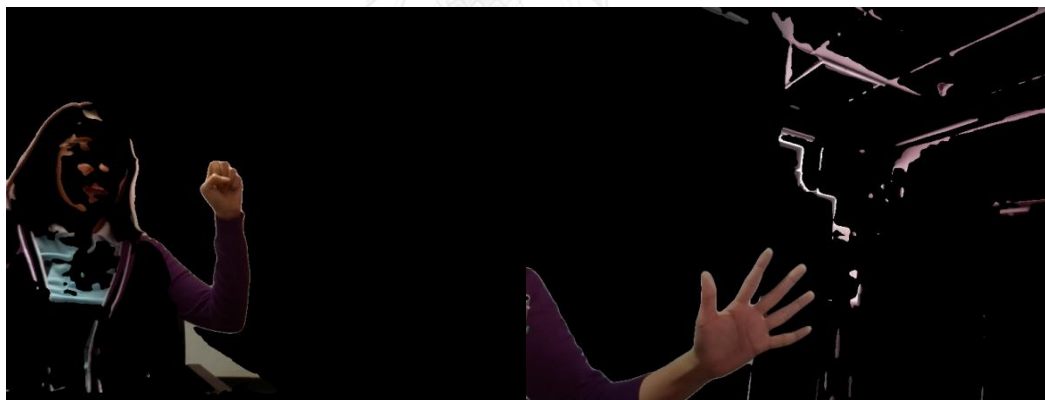
หลังจากขั้นตอนการทำภาพเบลอโดยใช้ตัวกรองเกาส์เซียนแล้ว ทำการปรับเทรซโฮลด์ของภาพไบนารีที่ได้เพื่อให้ภาพมีความคมชัดขึ้นดังแสดงในรูปที่ 3.34 จากนั้นทำการสร้าง Mask พิกเซลสีที่ได้จากภาพต้นฉบับบนภาพที่ผ่านการปรับเทรซโฮลด์ เพื่อให้ได้ภาพสี RGB กลับมาอีกครั้งดังแสดงในรูปที่ 3.35



(ก) เฟรมที่มีใบหน้า

(ข) เฟรมที่ไม่มีใบหน้า

รูปที่ 3.34 การทำเทรซโฮลด์ภาพไบนารี



(ก) เฟรมที่มีใบหน้า

(ข) เฟรมที่ไม่มีใบหน้า

รูปที่ 3.35 การรวมภาพไบนารีเทรซโฮลด์กับพิกเซลสีจากภาพต้นฉบับ

### 3.2.3.9 การแปลงภาพสีเป็นภาพ YUV

การแปลงโมเดลสีเป็นสิ่งสำคัญสำหรับการประมวลผลระบบ โมเดลสี YUV จัดเป็นโมเดลที่ให้ความรวดเร็วในการแปลงสี ในการทำงานของโมเดลสี YUV ผู้วิจัยเลือกประมวลผลบนองค์ประกอบ U และ V ที่เป็นองค์ประกอบสี (Chrominance) เนื่องจากการประมวลผลบนองค์ประกอบสีจะไม่ขึ้นอยู่กับสภาพแสงเหมือนกับองค์ประกอบ Y ที่เป็นองค์ประกอบแสง (Luminance) นอกจากนี้ องค์ประกอบ U และ V ยังมีความทนทานต่อการเปลี่ยนแปลงของสภาพแสง สัญญาณรบกวน ฉาก

หลังที่มีสิ่งของ (Cluttered background) รวมถึงการเปลี่ยนมุมของท่ามือ อาทิ การหมุน การเปลี่ยนระยะใกล้หรือไกล เป็นต้น สำหรับการแปลงค่าโมเดลสี RGB เป็นโมเดลสี YUV สามารถเขียนได้ดังนี้

$$Y = 0.299R + 0.587G + 0.114B \quad (3.33)$$

$$U = 0.492 (B-Y) \quad (3.34)$$

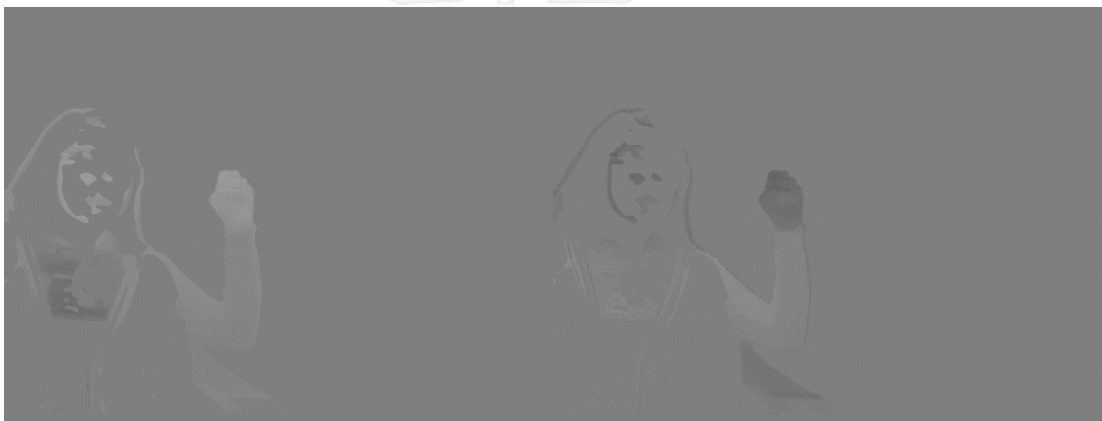
$$V = 0.877 (R-Y) \quad (3.35)$$

หรือเขียนได้ดังนี้

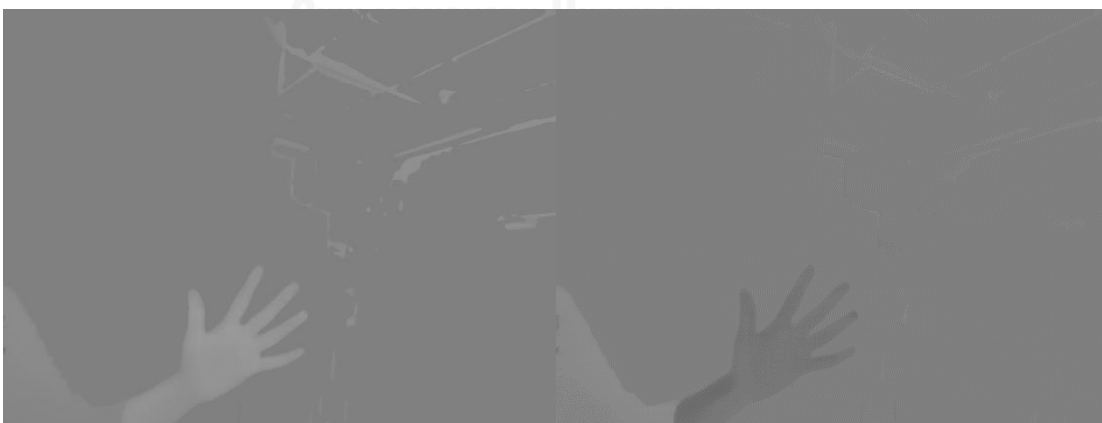
$$Y = 0.299R + 0.587G + 0.114B \quad (3.36)$$

$$U = -0.147R - 0.289G + 0.436B \quad (3.37)$$

$$V = 0.615R - 0.515G - 0.100B \quad (3.38)$$



(ก) ภาพองค์ประกอบ U ของเฟรมที่มีใบหน้า (ข) ภาพองค์ประกอบ V ของเฟรมที่มีใบหน้า



(ค) ภาพองค์ประกอบ U ของเฟรมที่ไม่มีใบหน้า (ง) ภาพองค์ประกอบ V ของเฟรมที่ไม่มีใบหน้า

รูปที่ 3.36 เฟรมภาพที่ได้จากการรวมองค์ประกอบ U และ V ของโมเดลสี YUV

### 3.2.3.10 การตรวจจับสีผิว

สำหรับการตรวจจับสีผิวสามารถทำได้โดยกำหนดค่าเทรชโโฮลด์เพื่อแยกพิกเซลสีผิวออกจากพิกเซลอื่น ๆ ที่ไม่ใช่สีผิว และเพื่อให้ได้ผลลัพธ์ที่ดีที่สุดจึงทำการกำหนดค่าเทรชโโฮลด์ในแต่ละองค์ประกอบสีของโมเดลสี YUV ให้มีช่วงที่เหมาะสมและมีกฎการตัดสินใจเพียงพอ โดยการทำงานของ การตรวจจับสีผิวจะตรวจจับเฉพาะพิกเซลที่มีสีอยู่ในช่วงค่าเทรชโโฮลด์ที่กำหนดเท่านั้น

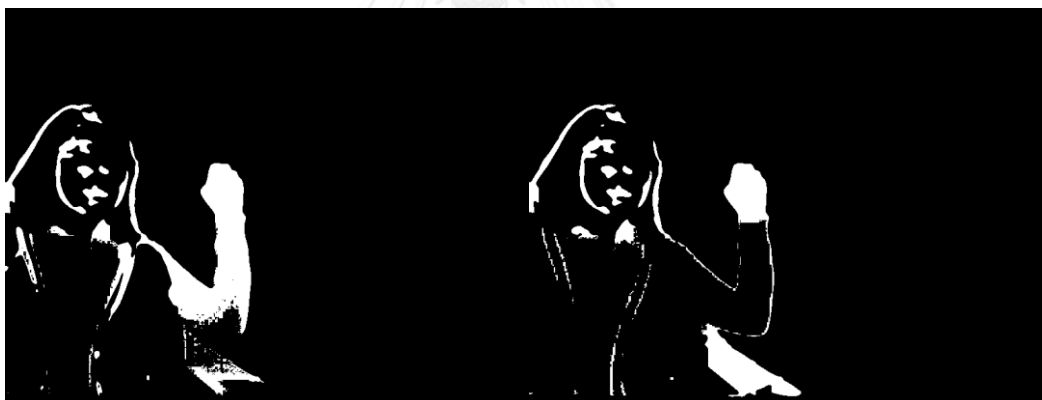
ในงานวิจัยนี้ทำการตั้งค่าเทรชโโฮลด์ 2 ค่า ได้แก่ ค่าขั้นต่ำและค่าขั้นสูง ค่าเทรชโโฮลด์ที่กำหนดในการตรวจจับสีผิวบนโมเดลสี YUV แสดงดังนี้

$$133 \leq U \leq 220$$

$$89 \leq V \leq 127$$

$$\text{เมื่อ } U, V = [0, 255]$$

รูปผลลัพธ์ที่ได้จากการปรับค่าเทรชโโฮลด์สีผิวขององค์ประกอบสี U และ V แสดงดังนี้



(ก) เทรชโโฮลด์บนองค์ประกอบ U แบบมีหน้า

(ข) เทรชโโฮลด์บนองค์ประกอบ V แบบมีหน้า



(ค) เทรชโโฮลด์บนองค์ประกอบ U แบบไม่มีหน้า

(ง) เทรชโโฮลด์บนองค์ประกอบ V แบบไม่มีหน้า

รูปที่ 3.37 การทำเทรชโโฮลด์ตรวจจับสีผิวบนองค์ประกอบ U และ V ของโมเดลสี YUV

ซึ่งหลังจากได้ภาพที่มีการปรับเทรซโฮลด์ของแต่ละองค์ประกอบแล้ว ระบบทำการรวมองค์ประกอบสี U และ V เข้าด้วยกันเพื่อประมวลผลในขั้นตอนต่อไป ภาพผลลัพธ์ของการรวมองค์ประกอบสี U และ V แสดงในรูปที่ 3.38



(ก) รวมองค์ประกอบ UV ของภาพมีใบหน้า (ข) รวมองค์ประกอบ UV ของภาพไม่มีใบหน้า  
รูปที่ 3.38 การรวมองค์ประกอบสี U และ V จากการทำเทรซโฮลด์ตรวจจับสีผิวของโมเดลสี YUV

#### 3.2.3.11 การปรับความเรียบของภาพ

หลังจากใช้ค่าเทรซโฮลด์สีผิวในช่องสี U และ V แล้ว นำองค์ประกอบสีดังกล่าวรวมเข้าด้วยกัน จากนั้นนำภาพที่ได้จากการรวมองค์ประกอบสี U และ V มาทำการเบลอเพื่อปรับภาพให้มีความเรียบเนียนมากขึ้น ผลลัพธ์ที่ได้จากการเบลอภาพแสดงในรูปที่ 3.39



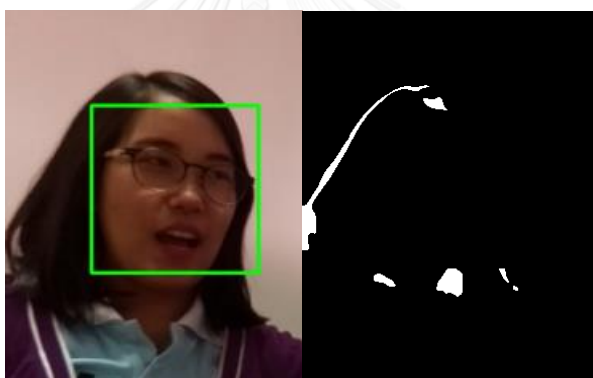
(ก) ภาพที่มีใบหน้า (ข) ภาพที่ไม่มีใบหน้า  
รูปที่ 3.39 การปรับความเรียบของภาพที่ได้จากการตรวจจับสีผิว

### 3.2.3.12 การทำเทรซโฮลด์

ภาพที่ได้จากขั้นตอนการปรับความเรียบจะมีความเบลอ และจากขั้นตอนดังกล่าวพบว่ายังมีเส้นสีขาวอยู่ในบริเวณใบหน้า เส้นผม และลำตัว ดังนั้นจึงทำการใช้ค่าเทรซโฮลด์ในการจำแนกพิกเซล โดยพิกเซลที่มากกว่าค่าเทรซโฮลด์ให้เป็นพิกเซลสีขาวหรือวัตถุฉากหน้า และพิกเซลที่น้อยกว่าค่าเทรซโฮลด์ให้เป็นพิกเซลสีดำหรือฉากหลัง กำหนดให้ค่าเทรซโฮลด์ที่เหมาะสมสำหรับจำแนกท่ามือออกจากฉากหลังเท่ากับ 200

### 3.2.3.13 การลบใบหน้า

เมื่อระบบตรวจจับใบหน้าได้แล้วจากขั้นตอนที่ 3.2.3.2 ระบบจะทำการสร้าง mask รูปสี่เหลี่ยมจตุรัสสีดำบนบริเวณกล่องสี่เหลี่ยมที่ใช้ตรวจจับใบหน้าดังแสดงในรูปที่ 3.40 (ก) ดังนั้นจะได้ผลลัพธ์ที่บริเวณใบหน้าถูกแทนที่ด้วยสี่เหลี่ยมสีดำดังแสดงในรูปที่ 3.40(ข) ส่งผลให้ระบบสามารถแสดงเฉพาะบริเวณมือเท่านั้นดังแสดงในรูปที่ 3.40(ค)



(ก) การตรวจจับใบหน้า (ข) แทนที่ใบหน้าด้วยรูปสี่เหลี่ยม



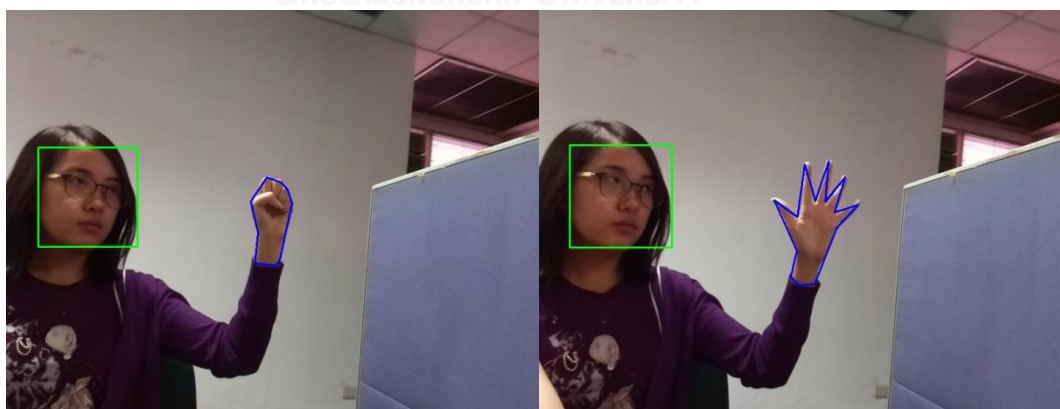
(ค) ระบบแสดงเฉพาะบริเวณมือ

รูปที่ 3.40 การลบใบหน้า

### 3.2.3.14 การหาเส้นขอบของมือ

หลังจากที่ได้ภาพไบนารีของมือแล้ว ขั้นตอนต่อไปจะเป็นการหาเส้นขอบของมือ ทำให้สามารถรู้รูปร่างของมือเพื่อนำไปหาจุดศูนย์กลางมวลของมือหรือจุดโมเมนต์ของมือและจุดยอดมุมของนิ้วได้ เส้นขอบที่คำนวณได้อาจมีหลายเส้น กำหนดให้ระบบเลือกเส้นขอบที่ใหญ่ที่สุด เพื่อนำมาคำนวณหาจุดโมเมนต์ของมือ และหาคอนเวกซ์ฮัลล์ (Convex Hull) หรือเส้นรอบมือที่ไม่มีส่วนเว้าเพื่อหาจุดยอดมุมของมือ ในการหาเส้นขอบจำเป็นต้องทราบว่าเส้นขอบนั้นมีเส้นขอบอื่นที่เป็นแบบปิดและมีจุดยอดมุมน้อยกว่าเส้นขอบเดิมด้วยหรือไม่ เพื่อหาลักษณะของเส้นโค้งบริเวณนิ้วมือ ดังนั้นจึงมีการใช้วิธีการหาความยาวส่วนโค้ง (Arc length) ร่วมกับการประมาณรูปหลายเหลี่ยม (Polygon approximation)

วิธีการหาความยาวส่วนโค้ง (Arc length) จะทำการคำนวณหาเส้นต่อเนื่องที่สร้างขอบเขตของรูปทรงเรขาคณิตแบบปิดหรือความยาวของเส้นโค้ง จากนั้นใช้วิธีการประมาณรูปหลายเหลี่ยม (Polygon approximation) เพื่อประมาณรูปร่างเส้นโค้งเป็นรูปร่างแบบอื่นที่มีจุดยอดมุมน้อยกว่าเส้นโค้งนั้น ๆ ดังนั้นระยะห่างระหว่างเส้นทั้งสองจะมีค่าน้อยกว่าหรือเท่ากับค่าความแม่นยำที่กำหนดไว้ ค่าความแม่นยำจะใช้สำหรับการคาดการณ์ความถูกต้อง โดยเป็นค่าพารามิเตอร์ที่ใช้กำหนดระยะห่างที่มากที่สุดระหว่างเส้นโค้งเดิมกับเส้นโค้งการประมาณ ซึ่งยิ่งค่าน้อย เส้นทั้งสองจะใกล้เคียงกันโดยเส้นประมาณจะมีลักษณะเป็นเส้นโค้ง แต่ถ้าค่ามากเส้นจะแตกต่างกันโดยเส้นประมาณจะมีลักษณะเป็นเส้นตรงมากขึ้น กำหนดให้มีค่าเท่ากับ 0.1 เมื่อทำการตรวจสอบกับค่าที่ได้มาจากวิธีการหาความยาวส่วนโค้ง พบว่าหากค่าที่ได้มีค่าเป็นจริง แสดงว่าเส้นโค้งที่ได้รับประมาณค่าเป็นเส้นโค้งแบบปิด นั่นคือจุดเริ่มต้นและจุดสิ้นสุดของมุมที่เชื่อมต่อกัน



รูปที่ 3.41 การหาเส้นขอบของมือ



### 3.2.3.15 การหาจุดศูนย์กลางของฝ่ามือและการกำหนดรัศมี

จุดศูนย์กลางของฝ่ามือ (Palm center) คือพื้นที่ที่มีการกำหนดด้วยค่าวงกลมสูงสุดภายใน (Maximum inscribed circle) โดยเป็นวงกลมที่มีขนาดใหญ่ที่สุดที่ล้อมรอบโดยเส้นโค้ง หรือเรียกว่า ส่วนที่ไม่ใช่วงกลม (Out of roundness) ดังแสดงในรูปที่ 3.42 (ก) ทั้งนี้ก่อนการคำนวณค่าวงกลมสูงสุดภายใน (Maximum inscribed circle) ควรทำการย่อขนาดเส้นโค้งให้มีขนาดเล็กลงเพื่อให้ระบบประมวลผลได้รวดเร็วมากยิ่งขึ้น

กำหนดค่าสำหรับการลดขนาดเส้นโค้งเท่ากับ 0.5 จากนั้นคัดเลือกเฉพาะส่วนที่สนใจ (Region of interest: ROI) นั่นคือเฉพาะส่วนมือดังแสดงในกรอบสีแดงในรูปที่ 3.42 (ข) โดยการสร้างกรอบสี่เหลี่ยมผืนผ้าโดยใช้วิธีการล้อมรอบด้วยสี่เหลี่ยมผืนผ้าแบบเส้นตรง (Straight bounding rectangle) กรอบสี่เหลี่ยมผืนผ้าที่ได้จะเป็นกรอบที่ไม่ขึ้นกับการหมุนของท่ามือ ดังนั้นพื้นที่ของกรอบจึงไม่ใช่ค่าน้อยสุดและสามารถครอบคลุมมือได้ทั้งหมด พิจารณากรอบบนระนาบ  $x$  และ  $y$  ให้  $x$  แทนความกว้าง และให้  $y$  แทนความสูง

เมื่อกำหนดกรอบเรียบร้อยแล้ว ขั้นตอนต่อไปคือค้นหาระยะห่างที่มากที่สุดจากเส้นขอบมือมาสู่จุดด้านใน การคำนวณจะใช้วิธีการทดสอบจุดในรูปหลายเหลี่ยม (Point Polygon Test) ซึ่งจะหา ระยะห่างที่สั้นที่สุดของจุดภายในเส้นขอบมือไปยังเส้นขอบมือที่ใกล้ที่สุด การคำนวณระยะห่างระหว่างจุดบนระนาบพิกัดใดๆ กำหนดให้พิกเซล  $S$  อยู่ที่ตำแหน่ง  $(i_1, j_1)$  และพิกเซล  $E$  อยู่ที่ตำแหน่ง  $(i_2, j_2)$  ระยะทางระหว่างพิกเซลสองพิกเซล ( $d$ ) สามารถคำนวณได้จากสมการยูคลิด (Euclidean) ดังนี้

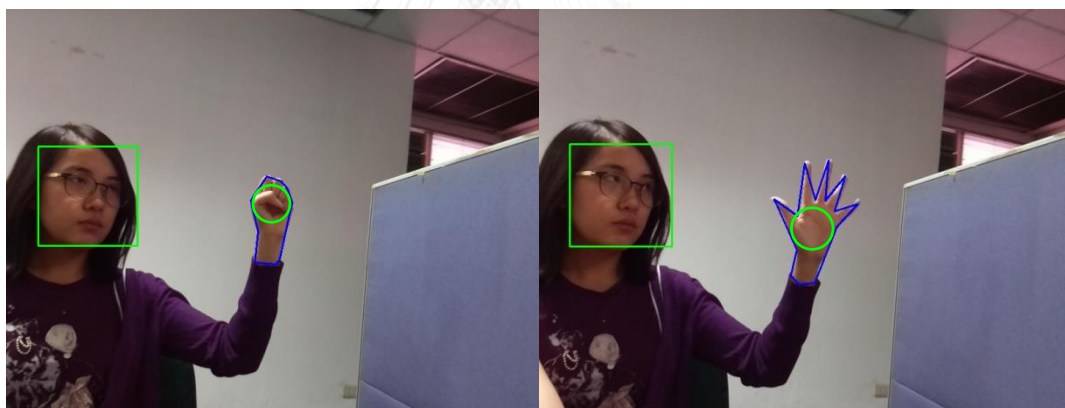
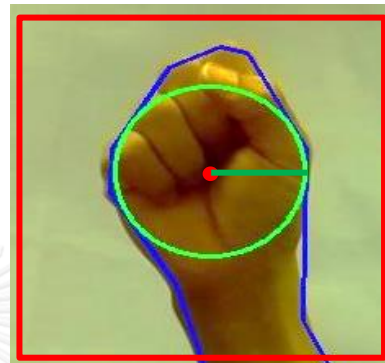
$$d = \sqrt{(i_2 - i_1)^2 + (j_2 - j_1)^2} \quad (3.39)$$

จากการเปรียบเทียบจุดกับค่าพารามิเตอร์เส้นโค้งที่กำหนดเพื่อทำการหาจุดศูนย์กลางของวงกลมภายใน พบว่าค่าระยะห่าง ( $d$ ) มีค่าเป็นบวก ดังนั้นจุดที่มีระยะห่างมากที่สุด คือ จุดศูนย์กลางของวงกลมสูงสุดภายใน (Center of maximum inscribed circle) ดังแสดงในรูปที่ 3.42(ก) เมื่อได้จุดศูนย์กลางของวงกลมสูงสุดภายในแล้วทำการปรับขนาดให้กลับไปมีขนาดเท่าเดิม

จากนั้นทำการปรับขนาดจุดศูนย์กลางของวงกลมสูงสุดภายใน (Center of maximum inscribed circle) ให้เป็นจุดศูนย์กลางของฝ่ามือ (Palm center) โดยการปรับค่าพารามิเตอร์ช่วงขนาดของจุดศูนย์กลางของวงกลมภายในให้มีค่าเท่ากับ 3 จากนั้นใช้วิธีการทดสอบจุดในรูปหลายเหลี่ยม (Point Polygon Test) อีกครั้ง ซึ่งมีกระบวนการหาระยะห่างโดยใช้สมการยูคลิดเหมือนการ

หาจุดศูนย์กลางของวงกลมสูงสุดภายใน การคำนวณระยะห่างระหว่างจุดบนระนาบพิกัดใดๆ สามารถคำนวณได้จากสมการยูคลิด (Euclidean) ดังที่ได้อธิบายในสมการที่ (3.39)

จากขั้นตอนนี้จะได้จุดศูนย์กลางของฝ่ามือดังแสดงเป็นจุดสีแดงในรูปที่ 3.42 (ข) และหากต้องการสร้างวงกลมล้อมรอบจุดศูนย์กลางของฝ่ามือ ก็สามารถสร้างได้โดยหารัศมีจากจุดศูนย์กลางของฝ่ามือไปยังเส้นวงกลมสูงสุดภายในจากนั้นจึงวาดเส้นวงกลม ดังแสดงเส้นวงกลมสีเขียวในรูปที่ 3.42 (ข)



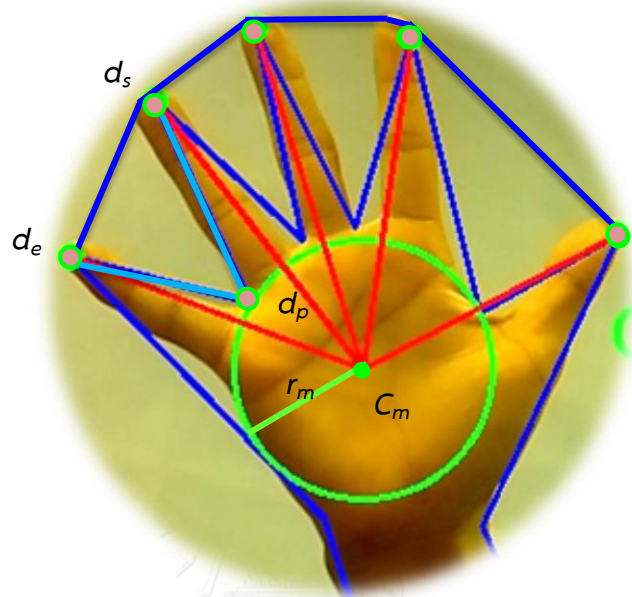
รูปที่ 3.42 การหาจุดศูนย์กลางของมือและการกำหนดรัศมี

### 3.2.3.16 การกำหนดตำแหน่งของปลายนิ้วมือ

การหาจุดยอดมุมของมือนั้นมีวัตถุประสงค์เพื่อแยกจุดปลายนิ้วมือออกจากจุดยอดมุมอื่น ๆ การหาจุดยอดมุมของนิ้วนั้นสามารถทำได้โดยการหาคอนเวกซ์ฮัลล์ (Convex hull) จากเส้นขอบที่ใหญ่ที่สุดของมือ ผลลัพธ์ที่ได้คือเส้นขอบของมือที่ไม่มีขอบเว้า หรือเป็นเส้นขอบที่ครอบคลุมพื้นที่ตั้งแต่ปลายแขนจนไปถึงปลายนิ้วทุกนิ้วรวมทั้งช่องว่างระหว่างนิ้วมือด้วย เมื่อทราบจุดยอดบนคอนเวกซ์

ฮัลล์ระบบจะสามารถระบุตำแหน่งของปลายนิ้วได้ทันที เนื่องจากจุดยอดบนคอนเวกซ์ฮัลล์ใดๆ จะเป็นจุดที่แสดงถึงปลายนิ้วมือ

การวิเคราะห์คุณลักษณะของเส้นขอบทำให้สามารถกำหนดตำแหน่งของมือได้อย่างแม่นยำ ซึ่งการที่จะพิจารณาว่าเป็นนิ้วมือหรือไม่จำเป็นต้องพิจารณาตามเกณฑ์ดังนี้



รูปที่ 3.43 พารามิเตอร์ที่เกี่ยวข้องในการหาท่ามือ

- กำหนดให้
- $C_m$  คือ จุดกลางฝ่ามือ
  - $r_m$  คือ รัศมีของฝ่ามือ
  - $d_s$  คือ จุดเริ่มต้น
  - $d_e$  คือ จุดสิ้นสุด
  - $d_p$  คือ จุดตำแหน่งลึกสุด

### 3.2.3.17 กฎการจำแนก (Rule Based Classifier)

ขั้นตอนนี้เป็นขั้นตอนสุดท้ายของระบบสำหรับการจำแนกความหมายของท่ามือ กฎการจำแนกพิจารณา 2 เงื่อนไข ได้แก่ ระยะห่างระหว่างจุดกลางฝ่ามือถึงจุดปลายนิ้วมือใด ๆ (Distance between palm center and fingertips) และจำนวนนิ้วมือ (Finger number) ในการหาระยะห่างใช้วิธีการหาระยะแบบยุคลิด (Euclidean distance) เพื่อตรวจสอบว่ามีนิ้วมือหรือไม่ โดยระยะห่างที่

ได้นี้จะนำมาคำนวณกับรัศมีของฝ่ามือ (Palm radius) เพื่อหาอัตราส่วนที่เหมาะสมของฝ่ามือและนิ้วมือในระยะต่าง ๆ จากนั้นเมื่อระบบทราบว่ามือนิ้วมือแล้วจะนับจำนวนนิ้วมือนั้น ๆ และเก็บค่าไว้

กำหนดให้ค่าระยะห่างระหว่างจุดกลางฝ่ามือถึงจุดปลายนิ้วมือใด ๆ มีค่าอยู่ระหว่าง 1.9 ถึง 4.2 ความสัมพันธ์ของอัตราส่วนระหว่างระยะห่างและรัศมีของฝ่ามือสามารถเขียนได้ดังแสดงในสมการที่ 3.40 และ 3.41 ดังนี้

$$d_l = 1.9 * r_m \quad (3.40)$$

$$d_h = 4.2 * r_m \quad (3.41)$$

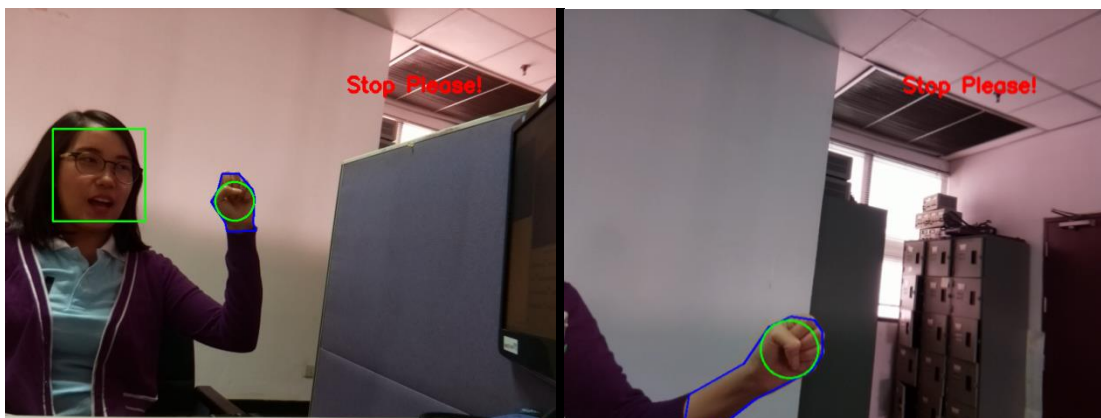
สามารถเขียนความสัมพันธ์ได้ดังนี้

$$d_l < D < d_h$$

เมื่อ	$d_l$	คือ ค่าเทรซโฮลด์ของระยะห่างขั้นต่ำมีค่าเท่ากับ 1.9
	$d_h$	คือ ค่าเทรซโฮลด์ของระยะห่างขั้นสูงมีค่าเท่ากับ 4.2
	$r_m$	คือ รัศมีของฝ่ามือ
	$D$	คือ ระยะห่างระหว่างจุดกลางฝ่ามือถึงจุดปลายนิ้วมือใด ๆ

จากสมการข้างต้นพบว่าถ้าค่าระยะห่าง ( $D$ ) ไม่อยู่ในอัตราส่วนนี้ระบบจะพิจารณาว่าไม่มีนิ้วมือ แต่ถ้าอยู่ในช่วงอัตราส่วนที่กำหนดระบบจะพิจารณาว่ามีนิ้วมือ โดยระบบจะเก็บจำนวนนิ้วมือไว้ หลังจากนั้นระบบจะจำแนกความหมายของท่ามือทั้ง 9 ท่ามือตามจำนวนนิ้วมือ ( $f_n$ ) และระยะห่าง ( $D$ ) กฎการจำแนกสามารถแบ่งออกเป็น 6 กฎดังนี้

*กฎข้อที่ 1:* ระบบไม่พบจำนวนนิ้วมือที่เก็บไว้ ( $f_n = 0$ ) แสดงว่าไม่มีระยะห่างระหว่างจุดกลางฝ่ามือถึงจุดปลายนิ้วมือใด ๆ ( $D$ ) ที่อยู่ในช่วงอัตราส่วนที่กำหนด กล่าวคือไม่มีการชูนิ้วขึ้นมาแน่นอน ดังนั้นกำหนดให้ท่านี้ คือ ท่ากำมือ มีความหมายว่า “หยุด”



รูปที่ 3.44 แสดงความหมายท่ากำมือ

กฎข้อที่ 2: ระบบพบจำนวนนิ้วมือที่เก็บไว้ 1 นิ้ว ( $f_n = 1$ ) แสดงว่ามีระยะห่างระหว่างจุดกลางฝ่ามือถึงจุดปลายนิ้วมือใด ๆ ( $D$ ) 1 นิ้วที่อยู่ในช่วงอัตราส่วนที่กำหนด ทั้งนี้มีท่ามือ 2 ท่าที่มีการชูนิ้วมือจำนวน 1 นิ้วมือเหมือนกัน ได้แก่ ท่าชูนิ้วโป้ง และท่าชูนิ้วชี้ ดังนั้นจึงจำเป็นต้องเปรียบเทียบระยะห่างที่แตกต่างกันระหว่างสองท่ามือนี้

สำหรับท่าชูนิ้วโป้งระบบจะหาค่าเทรซโฮลด์ ( $th$ ) จากการคำนวณความสัมพันธ์ระหว่างระยะห่างจากจุดกลางฝ่ามือถึงจุดปลายนิ้วโป้งถึงจุดปลายนิ้วโป้ง ( $d_{ct}$ ) กับรัศมีของฝ่ามือ ( $r_m$ ) สามารถเขียนสมการได้ดังนี้

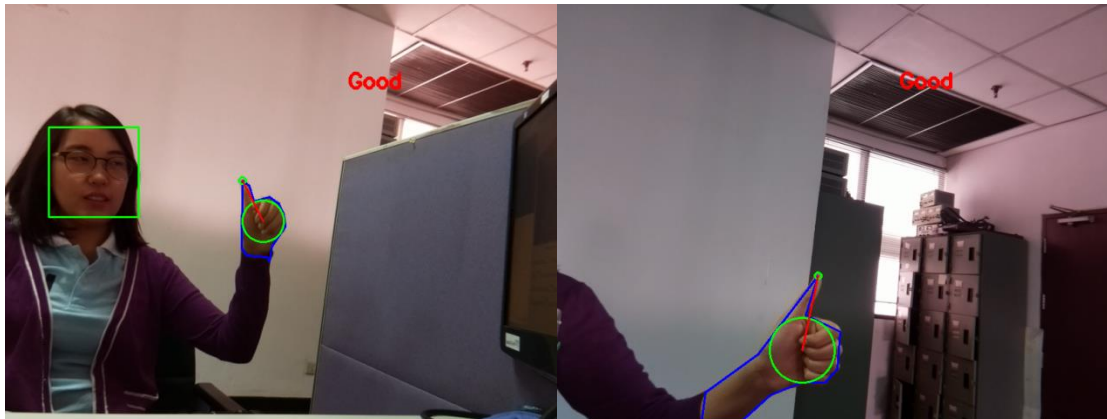
$$th = r_m / d_{ct} \quad (3.42)$$

เมื่อ  $th$  คือ ค่าเทรซโฮลด์

$r_m$  คือ รัศมีของฝ่ามือ

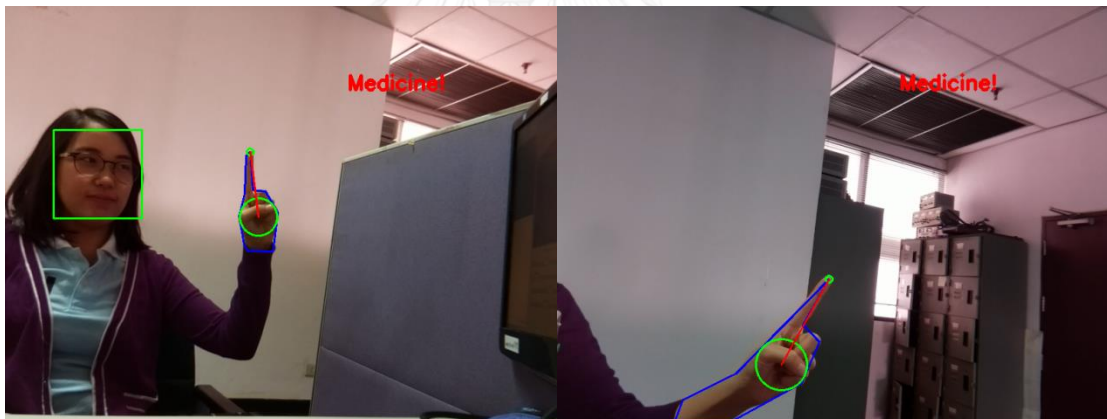
$d_{ct}$  คือ ระยะห่างจากจุดกลางฝ่ามือถึงจุดปลายนิ้วโป้ง

จากสมการข้างต้น พบว่าค่าเทรซโฮลด์ที่ได้ ( $th$ ) เป็นค่าที่ทำให้สามารถแยกท่าชูนิ้วโป้งออกจากท่าชูนิ้วชี้ได้ การจำแนกท่าชูนิ้วโป้งทำได้โดยกำหนดให้  $th > 0.4$  เนื่องจากยิ่งค่า  $th$  มากค่าระยะห่างจากจุดกลางฝ่ามือถึงปลายนิ้วยิ่งน้อย ดังนั้นถ้าอยากได้ท่าชูนิ้วโป้งที่เป็นนิ้วที่มีระยะสั้นที่สุด จึงต้องใช้ค่า  $th$  สูงกว่า 0.4 กำหนดให้ท่าชูนิ้วโป้งนี้มีความหมายว่า “ดี”



รูปที่ 3.45 แสดงความหมายท่าชูนิ้วโป้ง เทรชโสลด์มากกว่า 0.4

ในขณะที่การจำแนกท่าชูนิ้วชี้พิจารณาจากค่าเทรชโสลด์ของกรณีท่าชูนิ้วโป้ง กล่าวคือยิ่งค่า  $th$  น้อย ค่าระยะห่างจากจุดกลางฝ่ามือถึงปลายนิ้วยิ่งมาก ดังนั้นถ้าค่าเทรชโสลด์มีค่าน้อยกว่าหรือเท่ากับ 0.4 ( $th \leq 0.4$ ) กำหนดให้ท่ามือนี้มีความหมายว่า “ยา”



รูปที่ 3.46 แสดงความหมายท่าชูนิ้วชี้ เทรชโสลด์น้อยกว่าหรือเท่ากับ 0.4

กฎข้อที่ 3: ระบบพบจำนวนนิ้วมือที่เก็บไว้ 2 นิ้ว ( $f_n = 2$ ) แสดงว่ามีระยะห่างระหว่างจุดกลางฝ่ามือถึงจุดปลายนิ้วมือใด ๆ ( $D$ ) 2 นิ้วที่อยู่ในช่วงอัตราส่วนที่กำหนด ดังนั้นกำหนดให้ท่าชูนิ้วชี้และนิ้วกลางมีความหมายว่า “น้ำ”



รูปที่ 3.47 แสดงความหมายท่าชูนิ้วชี้และนิ้วกลาง

กฎข้อที่ 4: ระบบพบจำนวนนิ้วมือที่เก็บไว้ 3 นิ้ว ( $f_n = 3$ ) แสดงว่ามีระยะห่างระหว่างจุดกลางฝ่ามือถึงจุดปลายนิ้วมือใด ๆ ( $D$ ) 3 นิ้วที่อยู่ในช่วงอัตราส่วนที่กำหนด ทั้งนี้มีท่ามือ 2 ท่าที่มีการชูนิ้วมือจำนวน 3 นิ้วมือเหมือนกัน ได้แก่ 1) ท่าชูนิ้วโป้ง นิ้วชี้และนิ้วนาง และ 2) ท่าชูนิ้วชี้ นิ้วกลาง และนิ้วนาง ดังนั้นจึงจำเป็นต้องเปรียบเทียบระยะห่างที่แตกต่างกันระหว่างสองท่ามือนี้

สำหรับท่าแรกนั้น ระบบจะหาค่าเทรซโฮลต์ ( $th$ ) จากการคำนวณความสัมพันธ์ระหว่างระยะห่างจากจุดกลางฝ่ามือถึงจุดปลายนิ้วโป้ง ( $d_{ct}$ ) กับระยะห่างจากจุดปลายนิ้วโป้งถึงจุดปลายนิ้วชี้ ( $d_{ti}$ ) และระยะห่างจากจุดปลายนิ้วชี้ถึงจุดปลายนิ้วก้อย ( $d_{il}$ ) ซึ่งสามารถเขียนสมการได้ดังนี้

$$d_{me} = (d_{ti} + d_{il}) / 2 \quad (3.43)$$

$$th = d_{me} / d_{ct} \quad (3.44)$$

เมื่อ  $d_{me}$  คือ ค่าเฉลี่ยผลรวมระยะห่างจากปลายนิ้วโป้งถึงนิ้วชี้และปลายนิ้วชี้ถึงนิ้วก้อย

$th$  คือ ค่าเทรซโฮลต์

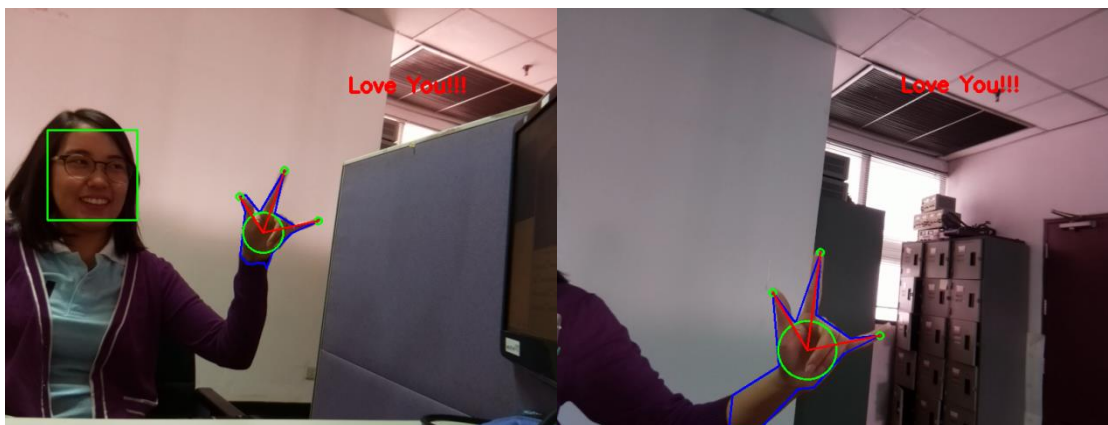
$d_{ct}$  คือ ระยะห่างจากจุดกลางฝ่ามือถึงจุดปลายนิ้วโป้ง

$d_{ti}$  คือ ระยะห่างจากจุดปลายนิ้วโป้งถึงจุดปลายนิ้วชี้

$d_{il}$  คือ ระยะห่างจากจุดปลายนิ้วชี้ถึงจุดปลายนิ้วก้อย

จากสมการข้างต้น เมื่อได้ค่าเฉลี่ยผลรวมระยะห่างจากปลายนิ้วโป้งถึงนิ้วชี้และปลายนิ้วชี้ถึงนิ้วก้อยแล้ว ( $d_{me}$ ) ค่าดังกล่าวถูกนำมาหารด้วยค่าระยะห่างจากจุดกลางฝ่ามือถึงจุดปลายนิ้วโป้ง ( $d_{ct}$ ) เพื่อให้ระบบสามารถตรวจจับและติดตามท่ามือได้ระยะต่าง ๆ ได้

สำหรับการจำแนกท่าเมื่อนั้นพบว่าค่าเทรซโฮลด์ที่ได้ ( $th$ ) สามารถจำแนกท่าชูนิ้วโป้ง นิ้วชี้ และนิ้วก้อยได้ ทำได้โดยกำหนดให้  $th > 0.7$  เนื่องจากท่านี้มีค่าระยะห่างของแต่ละนิ้วที่ใกล้เคียงกัน กล่าวคือระยะห่างจากจุดปลายนิ้วโป้งถึงจุดปลายนิ้วชี้( $d_{ij}$ ) และระยะห่างจากจุดปลายนิ้วชี้ถึงจุดปลายนิ้วก้อย( $d_{ij}$ ) และมีค่าสูงกว่าระยะห่างของแต่ละนิ้วในท่าชูนิ้วชี้ นิ้วกลาง และนิ้วนาง จึงกำหนดให้ค่าเทรซโฮลด์ที่กำหนดมีค่าสูง กำหนดให้ท่าเมื่อนี้มีความหมายว่า “รักคุณ”



รูปที่ 3.48 แสดงความหมายท่าชูนิ้วโป้ง นิ้วชี้ และนิ้วก้อย เทรซโฮลด์มากกว่า 0.7

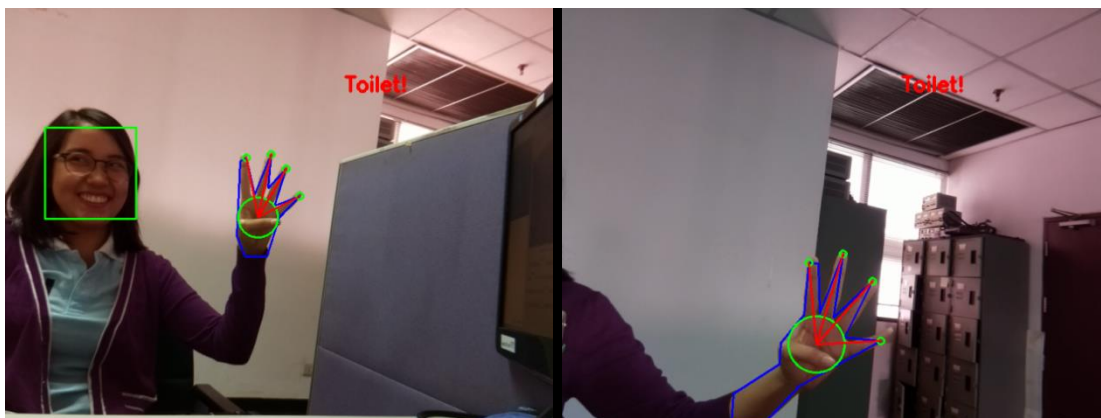
ในขณะที่การจำแนกท่าชูนิ้วชี้ นิ้วกลาง และนิ้วนาง พิจารณาจากค่าเทรซโฮลด์ของกรณีท่ารักคุณ ดังนั้นถ้าค่าเทรซโฮลด์มีค่าน้อยกว่าหรือเท่ากับ 0.7 ( $th \leq 0.7$ ) กำหนดให้ท่าเมื่อนี้มีความหมายว่า “หิว”



รูปที่ 3.49 แสดงความหมายท่าชูนิ้วชี้ นิ้วกลาง และนิ้วนาง เทรซโฮลด์น้อยกว่าหรือเท่ากับ 0.7



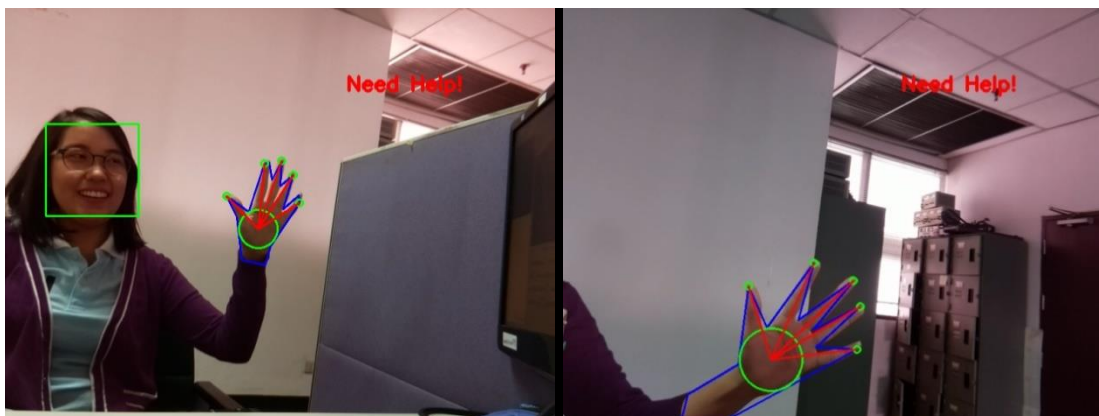
กฎข้อที่ 5: ระบบพบจำนวนนิ้วมือที่เก็บไว้ 4 นิ้ว ( $f_n = 4$ ) แสดงว่ามีระยะห่างระหว่างจุดกลางฝ่ามือถึงจุดปลายนิ้วมือใด ๆ ( $D$ ) 4 นิ้วที่อยู่ในช่วงอัตราส่วนที่กำหนด ดังนั้นกำหนดให้ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อยมีความหมายว่า “ห้องน้ำ”



รูปที่ 3.50 แสดงความหมายท่าชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย

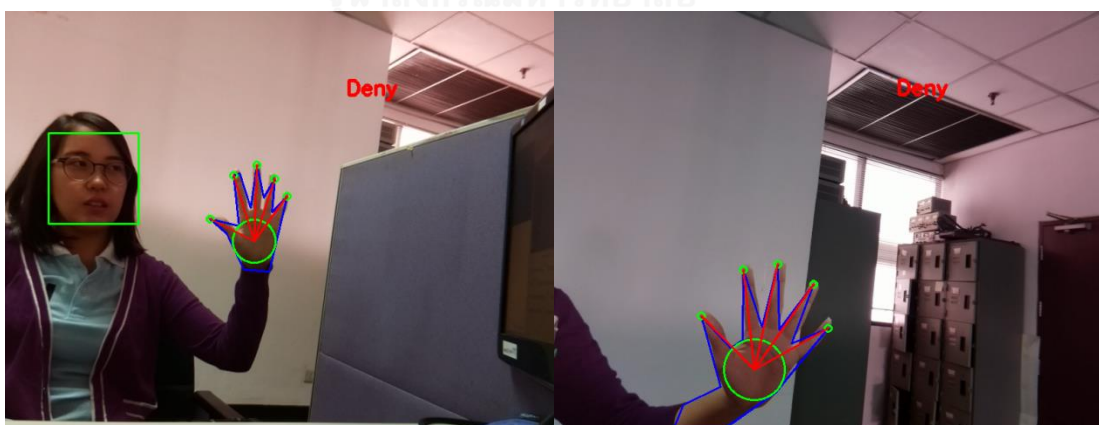
กฎข้อที่ 6: ระบบพบจำนวนนิ้วมือที่เก็บไว้ 5 นิ้ว ( $f_n = 5$ ) แสดงว่ามีระยะห่างระหว่างจุดกลางฝ่ามือถึงจุดปลายนิ้วมือใด ๆ ( $D$ ) 5 นิ้วที่อยู่ในช่วงอัตราส่วนที่กำหนด ทั้งนี้มีท่ามือ 2 ท่าที่มีการชูนิ้วมือจำนวน 5 นิ้วมือเหมือนกัน ได้แก่ 1) ท่าแบมือแบบเคลื่อนไหว และ 2) ท่าแบมือแบบอยู่นิ่ง ท่าทั้งสองท่ามือสามารถจำแนกได้ ดังนี้

ขั้นแรกกำหนดให้เฟรมภาพเฟรมแรกที่เข้ามาเป็นเฟรมอ้างอิง จากนั้นระบบจะเก็บตำแหน่งจุดพิกัดของนิ้วกลางไว้ในอาเรย์เพื่อใช้ในการจำแนกการเคลื่อนไหวของท่าแบมือ โดยหากระบบตรวจจับท่ามือที่มีจำนวนนิ้ว 5 นิ้วและไม่มีการเคลื่อนที่ของมือมากนัก แสดงว่านั่นคือท่าแบมือแบบนิ่ง โดยการเคลื่อนไหวที่เกิดขึ้นอาจมีเพียงเล็กน้อยเท่านั้น อาทิ อาการสั่นของมือในผู้สูงอายุ ดังนั้นกำหนดให้ท่าแบมือแบบนี้แสดงความหมาย “ช่วยด้วย”

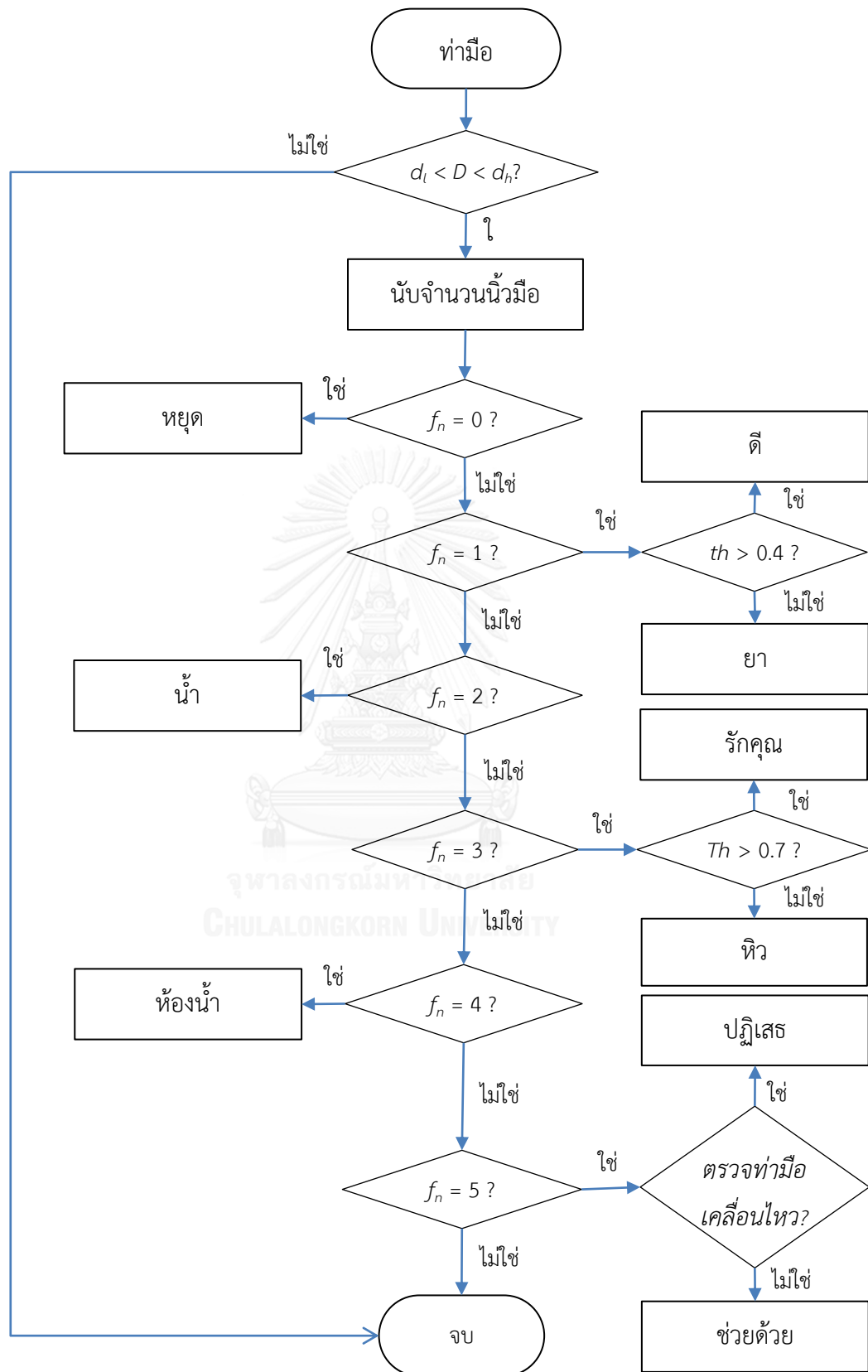


รูปที่ 3.51 แสดงความหมายท่าแบมือ

สำหรับการจำแนกท่าแบมือแบบเคลื่อนไหวหรือท่าโบกมือนั้น กำหนดให้ระบบตรวจจับท่ามือที่มีจำนวนนิ้ว 5 นิ้ว ( $f_n = 5$ ) และมีระยะห่างจากจุดพิกัดนิ้วกลางของเฟรมแรกไปยังเฟรมต่อไปมากกว่า 30 ( $d_m > 30$ ) ซึ่งเป็นระยะที่เหมาะสมสำหรับการจำแนกการเคลื่อนไหวที่ของมือว่าเป็นแบบสั้นเล็กน้อยหรือแบบโบกมือ หลังจากนั้นระบบจะตรวจสอบจำนวนเฟรมภาพที่มีค่าระยะห่างของจุดพิกัดนิ้วกลางในแต่ละเฟรมมากกว่า 30 ( $d_m > 30$ ) ว่ามีมากกว่า 4 เฟรมหรือไม่ ( $f_m > 4$ ) หากมีค่าระยะห่างและจำนวนเฟรมตามค่าเทรชโฮลด์ที่กำหนด ระบบจะแสดงความหมายว่า “ไม่” ทั้งนี้หากต้องการเปลี่ยนกลับมาเป็นท่ามือขอความช่วยเหลือ “ช่วยด้วย” ผู้สูงอายุจะต้องเปลี่ยนท่ามือเป็นท่าอื่น ๆ ก่อน เพื่อให้ระบบเริ่มต้นการทำงานใหม่นั้นเอง



รูปที่ 3.52 แสดงความหมายท่าโบกมือ ระยะนิ้วกลางระหว่างเฟรมมากกว่า 30 จุดพิกัด และมากกว่า 4 เฟรม



รูปที่ 3.53 แผนผังแสดงกฎการจำแนก

## บทที่ 4

### ผลการทดลอง

การทดลองของอัลกอริทึมทั้งสามอัลกอริทึมในหัวข้อ 3.2.1 3.2.2 และ 3.2.3 ดังอธิบายดังนี้

#### 4.1 ผลการทดลองของอัลกอริทึมรู้จำท่ามือแบบสถิติสำหรับผู้สูงอายุโดยใช้กฎการจำแนก

การทดลองนี้ทำการวิเคราะห์ท่ามือแบบภาพนิ่งจำนวน 6 ท่าภายใต้สภาพแวดล้อมปิด โดยติดตั้งกล้องให้อยู่ห่างจากท่ามือเป็นระยะห่างเท่ากับ 30 เซนติเมตร โดยใช้ฉากหลังสีขาวและไม่มีสิ่งกีดขวาง แสงสว่างมาจากหลอดไฟฟลูออเรสเซนต์ภายในห้องทดลองซึ่งมีระยะห่างจากระดับอ้างอิงหรือบริเวณมือที่ทำการทดลองอยู่ที่ 2 เมตร ผลลัพธ์ที่ได้จากการทดลองแสดงดังรูปที่ 4.1

จากรูปที่ 4.1 พบว่าระบบรู้จำท่ามือแบบสถิติให้ผลลัพธ์ที่ดีสำหรับการตรวจจับจุดกลางฝ่ามือ เส้นขอบของมือ และจุดยอดมูมปลายนิ้ว และสามารถจำแนกและแสดงความหมายของท่ามือทั้ง 6 ท่าได้ถูกต้อง โดยความหมายท่ามือกำหนดตามคำสั่งทั่วไปและความต้องการพื้นฐานในชีวิตประจำวัน ทั้งนี้ท่ามือและความหมายต้องขึ้นอยู่กับบริบทกลางกันระหว่างผู้สูงอายุและผู้ดูแล ความหมายของท่ามือที่แสดงบนภาพผลลัพธ์มีดังนี้

ท่ากำมือ หมายถึง หยุด

ท่าชูนิ้วชี้ หมายถึง ห้องน้ำ

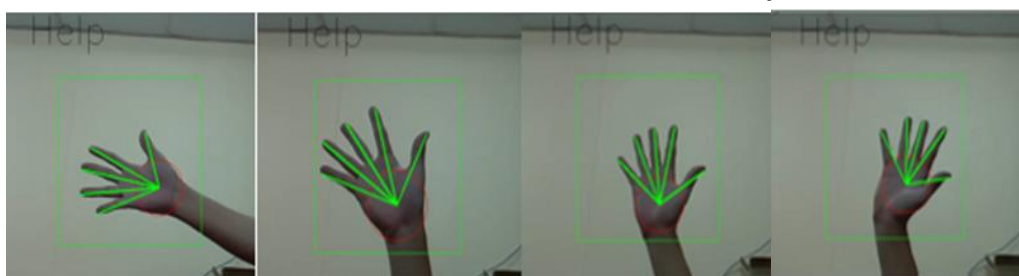
ท่าชูนิ้วชี้และนิ้วกลาง หมายถึง โอเค

ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง หมายถึง อาหาร

ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย หมายถึง น้ำ

ท่าแบมือ หมายถึง ช่วยด้วย

นอกจากนี้ระบบยังสามารถตรวจจับท่ามือได้ในองศาที่ต่างกันได้ในระดับดี อย่างไรก็ตาม เพื่อให้ระบบสามารถประมวลผลวิเคราะห์ท่ามือได้ ผู้ใช้งานจะต้องแสดงท่ามือภายในกรอบสี่เหลี่ยมและระยะที่กำหนดไว้เท่านั้น ภาพการทดสอบท่ามือในองศาต่าง ๆ แสดงในรูปที่ 4.1



(ก) 65 องศา

(ข) 45 องศา

(ค) 0 องศา

(ง) -45 องศา

รูปที่ 4.1 ผลลัพธ์การจำแนกท่าแบมือแบบภาพนิ่งในมุมต่าง ๆ



รูปที่ 4.2 แสดงผลลัพธ์ทำมือจากข้างซ้ายไปข้างขวาจำนวน 6 ท่าในองศาที่แตกต่างกัน

#### 4.2 ผลการทดลองของอัลกอริทึมสำหรับผู้สูงอายุโดยใช้ Optical Flow และ SURF

อัลกอริทึมนี้ประมวลผลโดยใช้วิธีการไหลของแสง การวิเคราะห์หีบไลบ และวิธี SURF ซึ่งเป็นวิธีที่ให้ความรวดเร็วในการประมวลผล มีความซับซ้อนน้อย และสะดวกต่อการประยุกต์ใช้งาน จากผลการทดลองพบว่าระบบสามารถจำแนกท่ามือจำนวน 4 ท่าได้ดี ได้แก่ ท่ากำมือ ท่าชูนิ้วชี้ ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย และท่าแบมือ

ผู้พัฒนาทำการหาค่าความแม่นยำที่ระบบสามารถแสดงความหมายท่ามือได้ถูกต้องสอดคล้องกับกฎการจำแนกท่ามือที่กำหนด การหาค่าความแม่นยำสามารถหาได้ดังนี้

$$\text{accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \quad (4.1)$$

เมื่อ True Positive (TP) คือ จำนวนภาพที่จำแนกท่ามือถูกต้องตามท่ามือที่แสดง

False Positive (FP) คือ จำนวนภาพที่จำแนกเป็นท่ามือทั้งที่ไม่มีท่ามือ

True Negative (TN) คือ จำนวนภาพที่ไม่มีท่ามือ

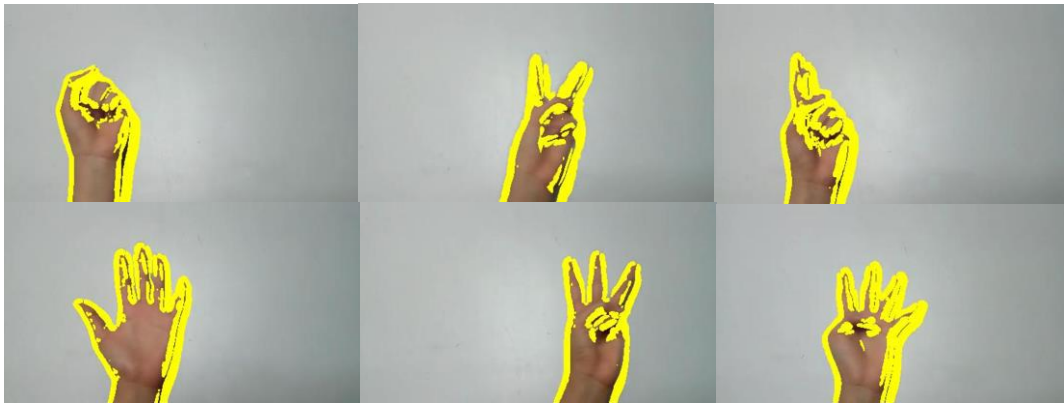
False Negative (FN) คือ จำนวนภาพที่จำแนกท่ามือผิด

จากสมการข้างต้นนำมาใช้หาค่าความแม่นยำของจำนวนภาพที่จำแนกท่ามือถูกต้องตามท่ามือที่แสดงและหาค่าความผิดพลาดของจำนวนภาพที่จำแนกท่ามือผิด ผลลัพธ์แสดงในตารางที่ 4.1 ดังนี้

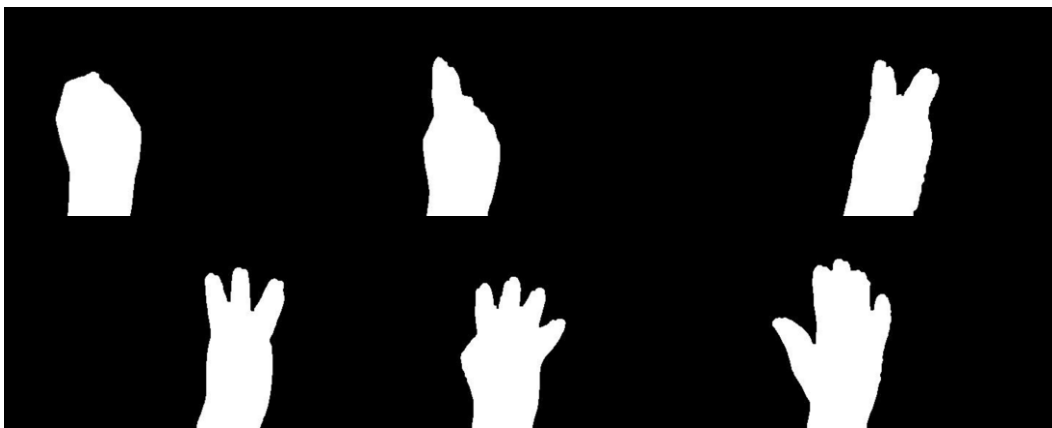
ตารางที่ 4.1 ผลการทดลองของการจำแนกท่ามือแบบเคลื่อนไหว

ท่ามือ	การจำแนกท่ามือ			
	จำนวนเฟรม	เฟรมที่จำแนกได้	ค่าความแม่นยำ (%)	ค่าความผิดพลาด (%)
กำมือ	300	300	100	0
ชูนิ้วชี้ (ยา)	310	290	93.55	6.45
ชูนิ้วชี้และนิ้วกลาง	340	210	61.76	38.24
ชูนิ้วชี้ นิ้วกลาง และนิ้วนาง	420	230	54.76	45.24
ชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย	410	360	87.8	12.2
ท่าแบมือ	420	389	92.62	7.38
<b>รวม</b>	<b>2,200</b>	<b>1,779</b>	<b>80.86</b>	<b>19.14</b>

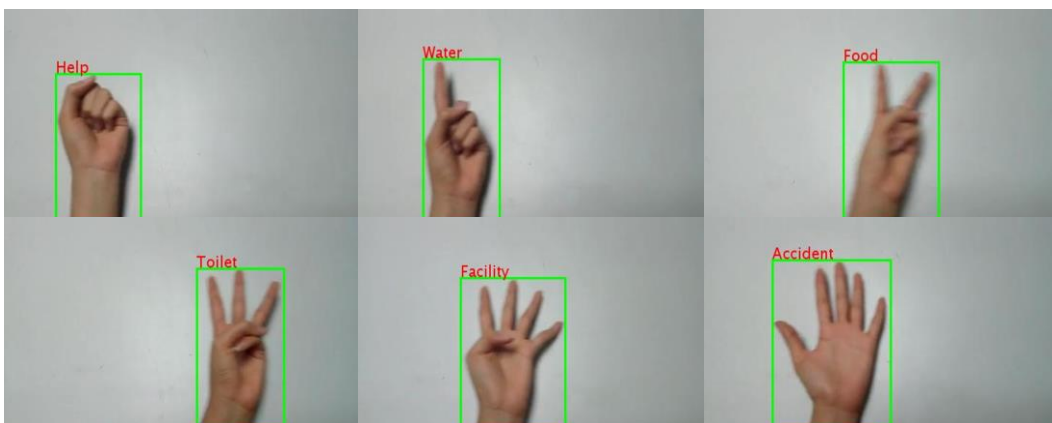
โดยภาพรวมแล้วค่าความแม่นยำของการจำแนกท่ามือเคลื่อนไหวจำนวน 6 ท่ามีค่าเท่ากับ 80.86 % จากทั้งหมด 2,200 เฟรม โดยท่ากำมือจะมีความแม่นยำมากที่สุดเท่ากับ 100 เปอร์เซ็นต์ สำหรับท่าชูนิ้วชี้และนิ้วกลาง และท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง มีค่าความแม่นยำต่ำเท่ากับ 61.76 เปอร์เซ็นต์ และ 54.76 เปอร์เซ็นต์ เนื่องจากนิ้วมือของทั้งสองท่ามือมีความใกล้เคียงกัน โดยภาพผลการทดลองของอัลกอริทึมนี้แสดงในรูปที่ 4.3-4.5 ดังนี้



รูปที่ 4.3 เวกเตอร์เคลื่อนที่จากการใช้วิธีการไหลของแสง



รูปที่ 4.4 การปรับปรุงโครงร่างภาพท่ามือ



รูปที่ 4.5 ผลลัพธ์จากการใช้การวิเคราะห์หับล้อม การไหลของแสง และ SURF

#### 4.3 ผลการทดลองของอัลกอริทึมที่นำเสนอสำหรับระบบรู้จำท่าบนพื้นฐานการมองเห็นสำหรับผู้สูงอายุ

จากผลการทดลองระบบในหัวข้อ 4.2 และ 4.3 พบว่ายังมีข้อจำกัดในการทดลองเป็นจำนวนมาก อาทิ ต้องทำการทดลองภายในสภาพแวดล้อมปิด ระยะห่างระหว่างกล้องและท่ามือที่สั้น การแสดงท่ามือในบริเวณที่จำกัด ท่ามือมีจำนวนน้อย และการที่ระบบไม่สามารถประมวลผลท่ามือแบบสถิติพร้อมกับท่ามือแบบเคลื่อนไหว เป็นต้น ดังนั้นเพื่อแก้ไขปัญหามาและปรับปรุงอัลกอริทึมให้มีประสิทธิภาพมากยิ่งขึ้น ผู้พัฒนาจึงทำการทดลองอัลกอริทึมรู้จำท่ามือบนพื้นฐานการรับรู้ภาพสำหรับผู้สูงอายุในสภาพแวดล้อม ตำแหน่ง ระยะ องศา และมุมมองที่แตกต่างกัน สำหรับการทดลองของระบบรู้จำท่ามือบนพื้นฐานการรับรู้ภาพสำหรับผู้สูงอายุนั้นจะแบ่งออกเป็น 6 ขั้นตอนหลัก โดยขั้นตอนท่ามือขณะอยู่นิ่งและเคลื่อนไหวในตำแหน่งต่างกัน แบ่งแต่ละขั้นตอนออกเป็น 2 ขั้นตอนย่อย ได้แก่

- 4.3.1. การทดลองท่ามือขณะอยู่นิ่ง
  - 4.3.1.1 การทดลองท่ามือขณะอยู่นิ่งบนฉากหลังแบบเรียบ
  - 4.3.1.2 การทดลองท่ามือขณะอยู่นิ่งบนฉากหลังแบบซับซ้อน
- 4.3.2. การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกัน
  - 4.3.2.1 การทดลองท่ามือเคลื่อนไหวบนฉากหลังแบบเรียบ
  - 4.3.2.2 การทดลองท่ามือเคลื่อนไหวบนฉากหลังแบบซับซ้อน
- 4.3.3. การทดลองท่ามือเคลื่อนไหวในองศาที่แตกต่างกัน
- 4.3.4. การทดลองท่ามือเคลื่อนไหวในมุมมองที่แตกต่างกัน
- 4.3.5. การทดลองท่ามือเคลื่อนไหวในระยะที่แตกต่างกัน
- 4.3.6. การทดลองการส่งข้อความแจ้งเตือนบนสมาร์ตโฟน

##### 4.3.1 การทดลองท่ามือขณะอยู่นิ่ง

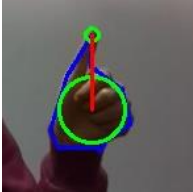
###### 4.3.1.1 การทดลองท่ามือขณะอยู่นิ่งบนฉากหลังแบบเรียบ

ในขั้นตอนแรกของการทดสอบนั้น ผู้พัฒนาทำการทดสอบท่ามือขณะอยู่นิ่งกับพื้นหลังที่ไม่มีสิ่งกีดขวาง โดยทดสอบท่ามือทั้งหมด 8 ท่ามือ ได้แก่ ท่ากำมือ (หยุด) ท่าชูนิ้วชี้ (ยา) ท่าชูนิ้วชี้และนิ้วกลาง (น้ำ) ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง (หิว) ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย (ห้องน้ำ) ท่าแบมือ (ช่วยด้วย) ท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย (รักคุณ) และท่าชูนิ้วโป้ง (ดี) กำหนดการทดสอบให้ท่ามือละ 300 เฟรม ผลลัพธ์ที่ได้แสดงตารางที่ 4.2



ตารางที่ 4.2 การทดลองท่ามือขณะอยู่นิ่งบนฉากหลังแบบเรียบ

ท่ามือ	รูปแสดงท่ามือ	จำนวนเฟรม	จำนวนเฟรมที่จำแนกได้ (TP)	จำนวนเฟรมที่จำแนกผิด (FN)	ค่าความแม่นยำ (%)
กำมือ (หยุด)		300	300	0	100
ชูนิ้วชี้ (ยา)		300	300	0	100
ชูนิ้วชี้และนิ้วกลาง (น้ำ)		300	300	0	100
ชูนิ้วชี้ นิ้วกลาง และนิ้วนาง (หิว)		300	300	0	100
ชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย (ห้องน้ำ)		300	300	0	100
แบมือ (ช่วยด้วย)		300	300	0	100
ชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย (รักคุณ)		300	300	0	100

ท่ามือ	รูปแสดงท่ามือ	จำนวนเฟรม	จำนวนเฟรมที่จำแนกได้ (TP)	จำนวนเฟรมที่จำแนกผิด (FN)	ค่าความแม่นยำ (%)
ชูนิ้วโป้ง (ดี)		300	300	0	100
รวม		2,400	2,400	0	100

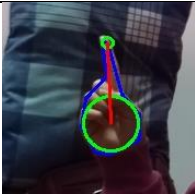
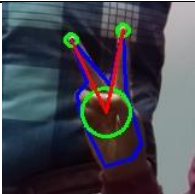


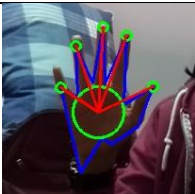
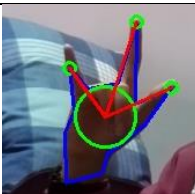
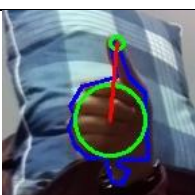
จากตารางที่ 4.2 พบว่าเมื่อแสดงท่ามือขณะอยู่เบื้องบนพื้นหลังที่ไม่มีสิ่งกีดขวาง ระบบให้ค่าความแม่นยำในการจำแนกความหมายในระดับดีมาก คือเท่ากับ 100 เปอร์เซ็นต์สำหรับท่ามือ 8 ท่า จากจำนวนเฟรมทั้งหมด 2,400 เฟรม สำหรับการทดสอบท่าโบกมือนั้นจะต้องมีการเคลื่อนที่ท่าแบมือ ดังนั้นจึงทำการทดสอบท่าโบกมือในหัวข้อที่ 4.3.2 ดังจะได้อธิบายต่อไป

#### 4.3.1.2 การทดลองท่ามือขณะอยู่เบื้องบนฉากหลังแบบซับซ้อน

เพื่อทดสอบความแม่นยำของอัลกอริทึม ผู้พัฒนาทำการทดสอบกับพื้นหลังที่มีสิ่งกีดขวาง โดยทดสอบท่ามือทั้งหมด 8 ท่ามือ ได้แก่ ท่ากำมือ (หยุด) ท่าชูนิ้วชี้ (ยา) ท่าชูนิ้วชี้และนิ้วกลาง (น้ำ) ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง (หิว) ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย (ห้องน้ำ) ท่าแบมือ (ช่วยด้วย) ท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย (รักคุณ) และท่าชูนิ้วโป้ง (ดี) กำหนดการทดสอบให้ท่ามือละ 300 เฟรม เช่นเดียวกับการทดสอบในหัวข้อ 4.3.1.1 ผลลัพธ์ที่ได้แสดงตารางที่ 4.3

ตารางที่ 4.3 การทดลองท่ามือขณะอยู่เบื้องบนฉากหลังแบบซับซ้อน

ท่ามือ	รูปแสดงท่ามือ	จำนวนเฟรม	จำนวนเฟรมที่จำแนกได้ (TP)	จำนวนเฟรมที่จำแนกผิด (FN)	ค่าความแม่นยำ (%)
กำมือ		300	300	0	100

ท่ามือ	รูปแสดงท่ามือ	จำนวนเฟรม	จำนวนเฟรมที่จำแนกได้ (TP)	จำนวนเฟรมที่จำแนกผิด (FN)	ค่าความแม่นยำ (%)
ชูนิ้วชี้		300	300	0	100
ชูนิ้วชี้และนิ้วกลาง		300	300	0	100
ชูนิ้วชี้ นิ้วกลางและนิ้วนาง		300	299	1	99.67
ชูนิ้วชี้ นิ้วกลางนิ้วนางและนิ้วก้อย		300	299	1	99.67
แบมือ		300	297	3	99
ชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย		300	299	1	99.67
ชูนิ้วโป้ง		300	300	0	100
<b>รวม</b>		<b>2,400</b>	<b>2,394</b>	<b>6</b>	<b>99.75</b>

จากตารางที่ 4.3 พบว่าเมื่อแสดงท่ามือขณะอยู่บนพื้นหลังแบบซับซ้อน พบว่าท่าที่มีความซับซ้อนน้อยหรือมีจำนวนนิ้วที่แสดงน้อยจะให้ค่าความแม่นยำในการจำแนกความหมายสูง โดยท่ากำมือ ท่าชูนิ้วชี้ ท่าชูนิ้วชี้และนิ้วกลาง และท่าชูนิ้วโป้ง มีค่าความแม่นยำในการจำแนกเท่ากับ 100 เปอร์เซ็นต์ สำหรับค่าความแม่นยำรองลงมา ได้แก่ ท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย ท่าชูนิ้วชี้ นิ้วกลาง และนิ้วนาง และท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย เท่ากับ 99.67 เปอร์เซ็นต์ โดยท่าแบมือมีความแม่นยำน้อยที่สุดเท่ากับ 99 เปอร์เซ็นต์ ดังนั้นจะเห็นว่ายังมีจำนวนนิ้วน้อยและระยะห่างของแต่ละนิ้วมากระบบยิ่งให้ค่าความแม่นยำดี ดังเห็นจากท่าแบมือเนื่องจากเป็นท่าที่มีความซับซ้อนมากที่สุด คือมีจำนวนนิ้วที่ชูขึ้นมากที่สุดและระยะห่างระหว่างนิ้วแต่ละนิ้วน้อยนั่นเอง

#### 4.3.2 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกัน

สำหรับการตรวจสอบค่าความแม่นยำของท่ามือเคลื่อนไหวในตำแหน่งต่าง ๆ บนฉากหลังแบบเรียบและฉากหลังแบบซับซ้อนนั้น สามารถแบ่งออกได้ 4 ทิศทาง ได้แก่ ขึ้น ลง ซ้าย และขวา ค่าเปอร์เซ็นต์ความแม่นยำแสดงในตารางที่ 4.4 ดังนี้

##### 4.3.2.1 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนฉากหลังแบบเรียบ

ทำการทดสอบท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกัน ได้แก่ การเคลื่อนที่ขึ้น ลง ซ้าย และขวาบนพื้นหลังแบบเรียบ โดยทดสอบทั้งหมด 8 ท่ามือ ได้แก่ ท่ากำมือ (หยุด) ท่าชูนิ้วชี้ (ยา) ท่าชูนิ้วชี้และนิ้วกลาง (น้ำ) ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง (หิว) ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย (ห้องน้ำ) ท่าโบกมือ (ปฏิเสธ) ท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย (รักคุณ) และท่าชูนิ้วโป้ง (ดี) กำหนดการทดสอบให้ท่ามือละ 100 เฟรม ค่าความแม่นยำของท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนฉากหลังแบบเรียบแสดงตารางที่ 4.4

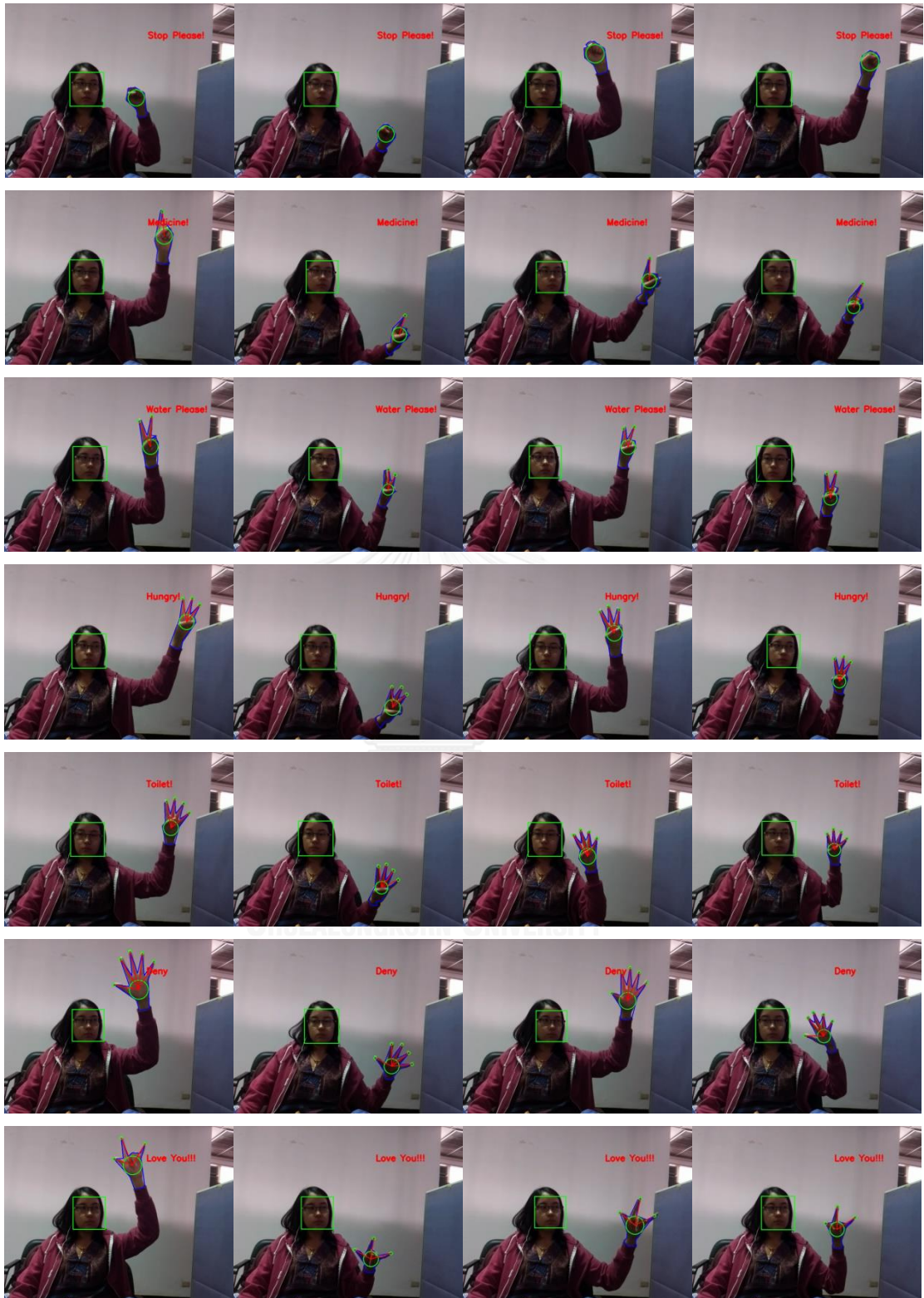
ตารางที่ 4.4 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนฉากหลังแบบเรียบ

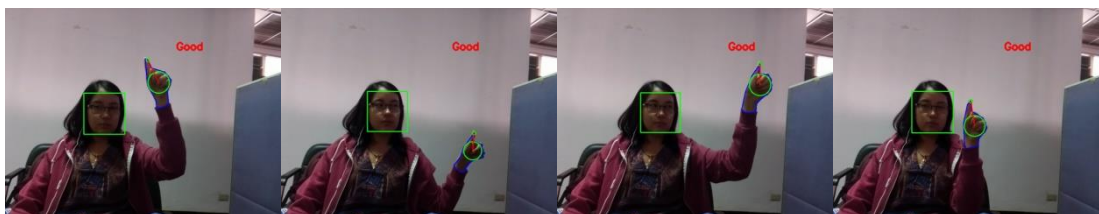
ท่ามือ	ค่าความแม่นยำของทิศทางการเคลื่อนที่ (%)				ค่าความแม่นยำรวม (%)
	ขึ้น	ลง	ซ้าย	ขวา	
กำมือ	100	100	100	100	100
ชูนิ้วชี้	99	100	100	100	99.75
ชูนิ้วชี้และนิ้วกลาง	100	99	100	100	99.75

ท่ามือ	ค่าความแม่นยำของทิศทางการเคลื่อนที่ (%)				ค่าความแม่นยำรวม (%)
	ขึ้น	ลง	ซ้าย	ขวา	
ชูนิ้วชี้ นิ้วกลางและนิ้วนาง	100	100	100	100	100
ชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย	100	100	100	100	100
*โบกมือ	100	100	99	98	99.25
ชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย	99	100	100	100	99.75
ชูนิ้วโป้ง	100	100	99	98	99.25
<b>รวม</b>	<b>99.75</b>	<b>99.88</b>	<b>99.63</b>	<b>99.5</b>	<b>99.69</b>

จากตารางที่ 4.4 พบว่าท่ากำมือยังคงให้ค่าความแม่นยำที่สูงที่สุดคือเท่ากับ 100 เปอร์เซ็นต์ สำหรับการทดสอบบนพื้นหลังที่ไม่มี ความซับซ้อน ค่าความแม่นยำรองลงมาเท่ากับ 99.75 เปอร์เซ็นต์ โดยความผิดพลาดที่เกิดขึ้น คือ เกิดจากการเคลื่อนที่ขึ้น เนื่องจากสภาพแสดงตำแหน่งบนจะแตกต่างจากระดับอ้างอิงที่ทดลอง ดังนั้นเมื่อเคลื่อนที่ขึ้นสภาพแสงจึงเปลี่ยนไป ผลที่ได้คือความยาวของนิ้วชี้สั้นลง สำหรับท่าชูนิ้วชี้ นอกจากนี้ระยะห่างระหว่างนิ้วถือเป็นปัจจัยสำคัญสำหรับท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย ซึ่งเมื่อเคลื่อนที่ขึ้นแล้วท่ามือไม่หันเข้าหากล้องทำให้นิ้วโป้งและนิ้วชี้ใกล้กัน ระบบจึงตีความเป็นท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง สำหรับการเคลื่อนที่ลงในท่าชูนิ้วชี้และนิ้วกลาง ระบบจะตรวจจับความยาวของนิ้วนางที่ไหลขึ้นมาในบางเฟรม จึงตีความเป็นท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง เช่นกัน

สำหรับท่าที่มีค่าความแม่นยำน้อยที่สุดคือท่าโบกมือและท่าชูนิ้วโป้ง เท่ากับ 99.25 เปอร์เซ็นต์ ค่าความผิดพลาดของท่าโบกมือนั้นเกิดจากสภาพแสงและการเคลื่อนที่ โดยเฉพาะการเคลื่อนมือไปทางขวาหรือซ้าย โดยสภาพแสงมีผลทำให้ความยาวของนิ้วสั้นลง ทำให้ปลายนิ้วก้อยและนิ้วโป้งไม่สามารถตรวจจับได้ ระบบจึงตีความหมายของท่ามือผิด ซึ่งทำให้ต้องเริ่มการจำแนกใหม่เป็นความหมายว่า “ช่วยด้วย” แล้วจึงให้ความหมายว่า “ปฏิเสธ” ตามเงื่อนไขที่กำหนด สำหรับความผิดพลาดของท่าชูนิ้วโป้งเกิดจากการเคลื่อนที่ โดยเฉพาะการเคลื่อนมือไปทางขวาหรือซ้าย เพราะจะทำให้มือไม่หันหน้าเข้ากล้อง ทำให้บางเฟรมภาพพบว่าความยาวของนิ้วมีมากขึ้นจากมุมของมือที่เปลี่ยนไปขณะทำการทดลอง ซึ่งส่งผลต่อขั้นตอนการลบพื้นหลังที่ให้ค่าฉากหน้ามีค่าเท่ากับ 1 หรือ พิกเซลสีขาวมากขึ้น ระบบจึงตรวจจับเป็นท่าชูนิ้วชี้ นั่นเอง ภาพผลลัพธ์ที่ได้จากการทดลองแสดงดังนี้





รูปที่ 4.6 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนฉากหลังแบบเรียบ

#### 4.3.2.2 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนฉากหลังแบบซับซ้อน

เพื่อทดสอบความแม่นยำของอัลกอริทึม ผู้พัฒนาทำการทดสอบท่ามือขณะเคลื่อนไหวในตำแหน่งที่แตกต่างกัน ได้แก่ การเคลื่อนที่ขึ้น ลง ซ้าย และขวาบนพื้นหลังแบบซับซ้อน โดยทดสอบท่ามือทั้งหมด 8 ท่ามือ กำหนดการทดสอบให้ท่ามือละ 100 เฟรม เช่นเดียวกับการทดสอบในหัวข้อ

#### 4.3.2.1 ผลลัพธ์ที่ได้แสดงตารางที่ 4.5

ตารางที่ 4.5 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนพื้นหลังแบบซับซ้อน

ท่ามือ	ค่าความแม่นยำของทิศทางการเคลื่อนที่ (%)				ค่าความแม่นยำรวม (%)
	ขึ้น	ลง	ซ้าย	ขวา	
กำมือ	100	100	100	100	100
ชูนิ้วชี้	99	99	100	99	99.25
ชูนิ้วชี้และนิ้วกลาง	100	97	99	100	99
ชูนิ้วชี้ นิ้วกลางและนิ้วนาง	98	100	99	100	99.25
ชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย	99	100	99	98	98.5
*โบกมือ	99	100	97	97	98.25
ชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย	98	100	100	99	99.25
ชูนิ้วโป้ง	99	100	97	98	98.5
<b>รวม</b>	<b>99</b>	<b>99.5</b>	<b>98.88</b>	<b>98.88</b>	<b>99</b>

จากตารางที่ 4.5 พบว่าท่ากำมือยังคงให้ค่าความแม่นยำที่สูงที่สุดคือเท่ากับ 100 เปอร์เซ็นต์ สำหรับการทดสอบบนพื้นหลังที่มีความซับซ้อน ค่าความแม่นยำรองลงมาเท่ากับ 99.25 เปอร์เซ็นต์ โดยความผิดพลาดที่เกิดขึ้นมากที่สุด คือ การเคลื่อนที่ทางขวาและทางซ้าย โดยความผิดพลาดที่เกิดขึ้นเกิดจากการเคลื่อนที่ท่ามือที่เร็วเกินไปและการเปลี่ยนของแสง สำหรับค่าความแม่นยำรองลงมา คือ การเคลื่อนที่ขึ้นและลง ซึ่งเกิดจากสภาพแสงเช่นกัน โดยในห้องทดลองจะเห็นว่ายิ่งเคลื่อนที่ขึ้นสูงจะยังมีแสงสว่างมากขึ้นเพราะเข้าใกล้หลอดไฟ ดังแสดงในรูปที่ 4.7 ซึ่งแสงที่มากไปก็ส่งผลให้เกิดความผิดพลาดในการตรวจจับสีผิว สำหรับท่าที่ให้ความแม่นยำน้อยที่สุด คือ ท่าโบกมือ











รูปที่ 4.7 การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกันบนฉากหลังแบบซับซ้อน

#### 4.3.3 การทดลองท่ามือเคลื่อนไหวในองศาที่แตกต่างกัน

การทดลองนี้ทำการทดสอบความแม่นยำของท่ามือเคลื่อนไหวในองศาที่ต่างกันของอัลกอริทึมบนพื้นหลังแบบเรียบ แบ่งออกเป็น 7 องศา โดยกำหนดให้องศาเมื่อท่ามืออยู่ตรงกลางเป็นองศาอ้างอิงมีค่าเท่ากับ 0 องศา เมื่อท่ามือเอียงทางขวาให้มีค่าเป็นบวก (+) ได้แก่ 20 45 และ 65 องศา และเมื่อท่ามือเอียงทางซ้ายให้มีค่าเป็นลบ (-) ได้แก่ -20 -45 และ -65 องศา โดยทดสอบกับท่ามือทั้งหมด 8 ท่ามือ กำหนดการทดสอบให้ท่ามือละ 100 เฟรม เช่นเดียวกับการทดสอบในหัวข้อ

##### 4.3.2.1 ค่าความแม่นยำของท่ามือเคลื่อนไหวในองศาที่ต่างกันแสดงตารางที่ 4.6

ตารางที่ 4.6 การทดลองท่ามือเคลื่อนไหวในองศาที่ต่างกัน

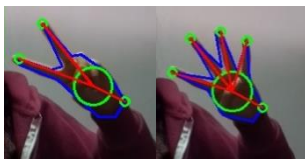
ท่ามือ	ค่าความแม่นยำขององศาของมือ (%)							ค่าความแม่นยำรวม (%)
	-65	-45	-20	0	20	45	65	
กำมือ	100	100	100	100	100	100	100	100
ชูนิ้วชี้	100	100	100	100	100	100	100	100
ชูนิ้วชี้และนิ้วกลาง	97	100	100	100	100	100	100	99.57
ชูนิ้วชี้ นิ้วกลางและนิ้วนาง	100	100	100	100	100	100	98	99.71
ชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย	98	100	100	100	100	100	100	99.71
แบมือ	97	100	100	100	100	100	96	99
ชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย	100	100	100	100	100	100	100	100
ชูนิ้วโป้ง	100	100	100	100	100	100	100	100
<b>รวม</b>	<b>99</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>99.25</b>	<b>99.75</b>

จากการทดลองพบว่าระบบสามารถจำแนกท่ามือได้กับองศาต่าง ๆ อย่างหลากหลายในระดับดีมาก โดยองศาที่ -45 -20 0 20 และ 45 มีค่าความแม่นยำเท่ากับ 100 เปอร์เซ็นต์ในทุกท่ามือ ภาพผลลัพธ์การทดสอบความแม่นยำในการเคลื่อนที่ในองศาที่ต่างกันของอัลกอริทึมบนพื้นหลังแบบเรียบแสดงดังรูปที่ 4.8 ทั้งนี้ค่าความผิดพลาดในการจำแนกท่ามือที่องศา -65 และ 65 เกิดขึ้นกับ 4 ท่ามือที่มีจำนวนนิ้วที่ชูขึ้นมาก ได้แก่ ท่าชูนิ้วชี้และนิ้วกลาง และท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและ

นิ้วก้อยที่องศา -65 ทำชูนิ้วชี้ นิ้วกลางและนิ้วนางที่องศา 65 และท่าแบมือที่องศา -65 และ 65 โดยที่ความผิดพลาดเกิดจากระบบจำแนกระยะทางความยาวจากจุดกลางฝ่ามือไปยังจุดคอนเวกซ์ฮัล บริเวณข้อมืองดแสดงในรูปที่ 4.9 ว่าเป็นนิ้วมือด้วย ดังนั้นระบบจึงตีความจำนวนนิ้วเกินมา 1 จำนวน



รูปที่ 4.8 แสดงองศาที่แตกต่างกันของท่ามือแต่ละท่า



รูปที่ 4.9 แสดงผลลัพธ์ที่ระบบจำแนกผิดพลาด

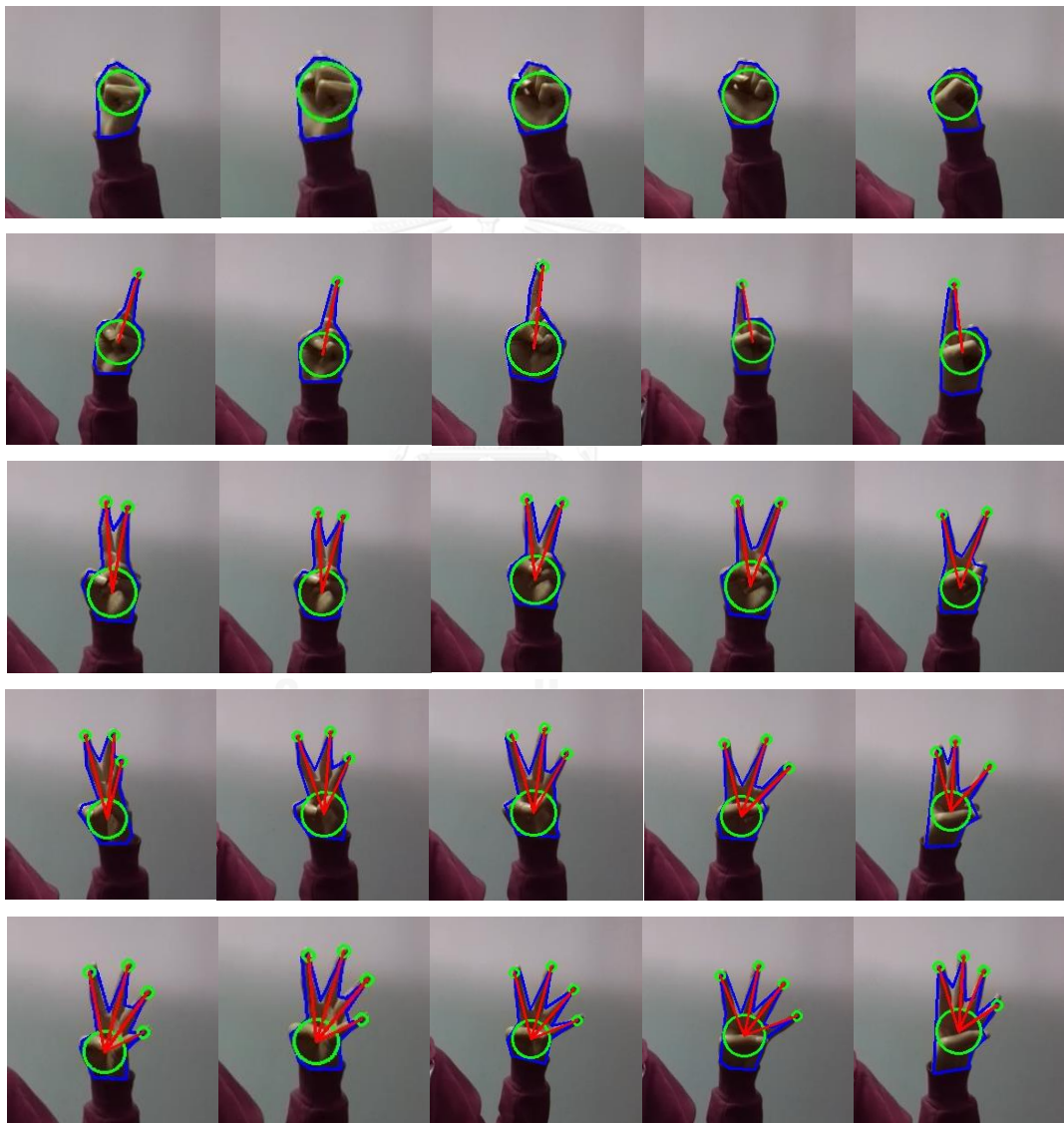
#### 4.3.4 การทดลองท่ามือเคลื่อนไหวในมุมมองที่แตกต่างกัน

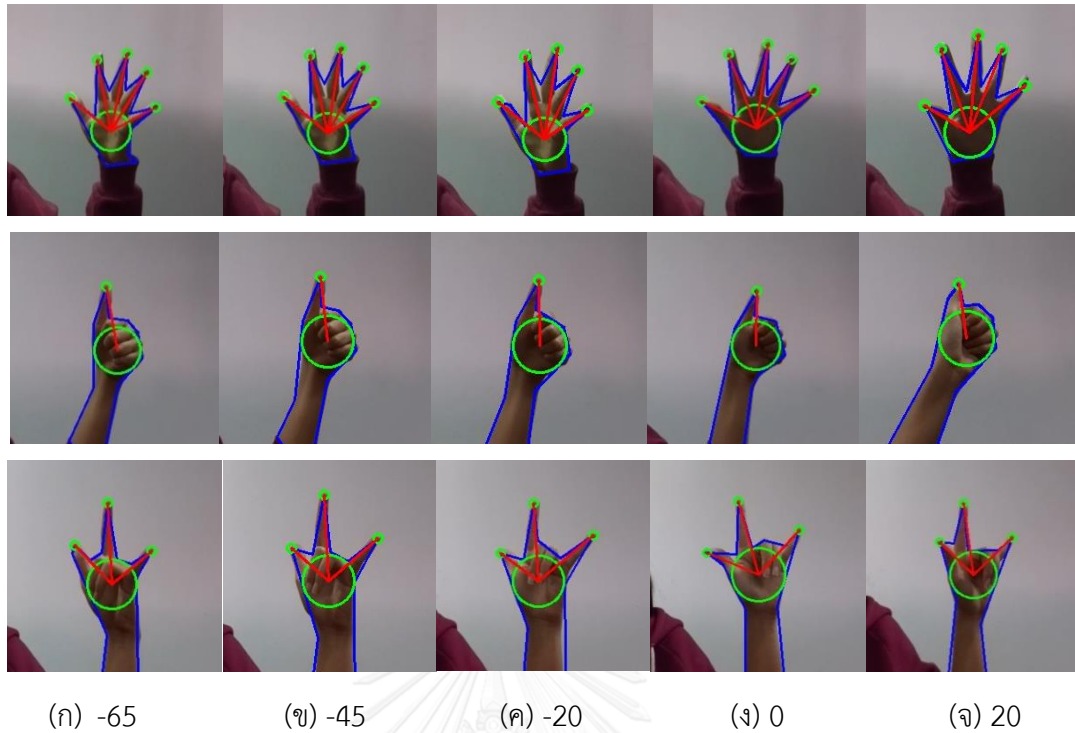
มุมมองในการแสดงท่ามือจัดเป็นปัญหาที่สำคัญสำหรับวิธีการรับรู้ภาพคอมพิวเตอร์ (Computer vision-based method) ดังนั้นผู้พัฒนาจึงทำการทดสอบการเคลื่อนไหวที่ในมุมมองที่ต่างกันของท่ามือ กำหนดมือขวาเป็นข้างที่ใช้แสดงท่ามือ เมื่อท่ามืออยู่ตรงกลางให้เป็นองศาอ้างอิงมีค่าเท่ากับ 0 องศา เมื่อท่ามือหมุนทวนเข็มนาฬิกามีค่าเป็นบวก (-) ได้แก่ -20 -45 และ -65 องศา และเมื่อท่ามือหมุนตามเข็มนาฬิกาให้มีค่าเป็นลบ (+) ได้แก่ 20 องศา ค่าความแม่นยำของท่ามือเคลื่อนไหวในมุมมองที่แตกต่างกันแสดงในตารางที่ 4.7

ตารางที่ 4.7 การทดลองท่ามือขณะเคลื่อนไหวที่ในมุมมองที่แตกต่างกัน

ท่ามือ	ค่าความแม่นยำขององศาของมือ (%)					ค่าความแม่นยำรวม (%)
	-65	-45	-20	0	20	
กำมือ	100	100	100	100	100	100
ชูนิ้วชี้	100	100	100	100	100	100
ชูนิ้วชี้และนิ้วกลาง	100	99	100	100	100	99
ชูนิ้วชี้ นิ้วกลางและนิ้วนาง	99	100	100	100	100	99
ชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย	100	100	100	100	96	99.2
แบมือ	100	100	100	100	92	98.4
ชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย	100	100	100	100	100	100
ชูนิ้วโป้ง	100	100	100	100	100	100
รวม	99.88	99.88	100	100	98.5	99.65

จากการทดลองพบว่าการหมุนมือในแต่ละท่ามือนั้นจะต้องให้ระบบเห็นจุดปลายนิ้วมือของท่ามือนั้น ๆ ให้ครบทุกนิ้วดังแสดงในรูปที่ 4.10 โดยเฉพาะเมื่อมีการหมุนมือในองศาที่มากขึ้น อาทิ องศาที่ -45 และ -65 เนื่องจากวิธีการคอนเวกซ์ฮัลจะพิจารณาบริเวณจุดปลายนิ้วมือ จึงต้องแสดงจุดปลายนิ้วมือให้ชัดเจน ซึ่งจากตารางที่ 4.7 แสดงให้เห็นว่าองศาที่ให้ค่าความผิดพลาด ได้แก่ 20 -45 และ -65 องศา โดยท่ามือที่ระบบจำแนกความหมายผิด ได้แก่ ท่าชูนิ้วชี้และนิ้วกลางที่ -45 องศา ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนางที่ -65 องศา ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย และท่าแบมือที่ 20 องศา โดยท่าแบมือที่ยังคงเป็นท่าที่มีค่าความแม่นยำน้อยที่สุด





รูปที่ 4.10 แสดงท่ามือเคลื่อนไหวในมุมมองที่แตกต่างกัน

#### 4.3.5 การทดลองท่ามือเคลื่อนไหวในระยะที่แตกต่างกัน

ระยะห่างระหว่างท่ามือและกล้องเป็นอีกหนึ่งปัญหาสำคัญสำหรับระบบการรู้จำท่ามือ ดังนั้นผู้พัฒนาจึงทำการทดสอบการเคลื่อนไหวในระยะที่ต่างกันของท่ามือ กำหนดมือขวาเป็นข้างที่ใช้แสดงท่ามือ โดยแบ่งระยะห่างระหว่างกล้องและท่ามือออกเป็น 2 ช่วง ได้แก่ ช่วงระยะ 0.5 - 1 เมตรและระยะ 1 - 1.5 เมตร ค่าความแม่นยำของท่ามือเคลื่อนไหวในระยะที่แตกต่างกันแสดงในตารางที่ 4.8 ดังนี้

ตารางที่ 4.8 การทดลองท่ามือเคลื่อนไหวในระยะที่แตกต่างกัน

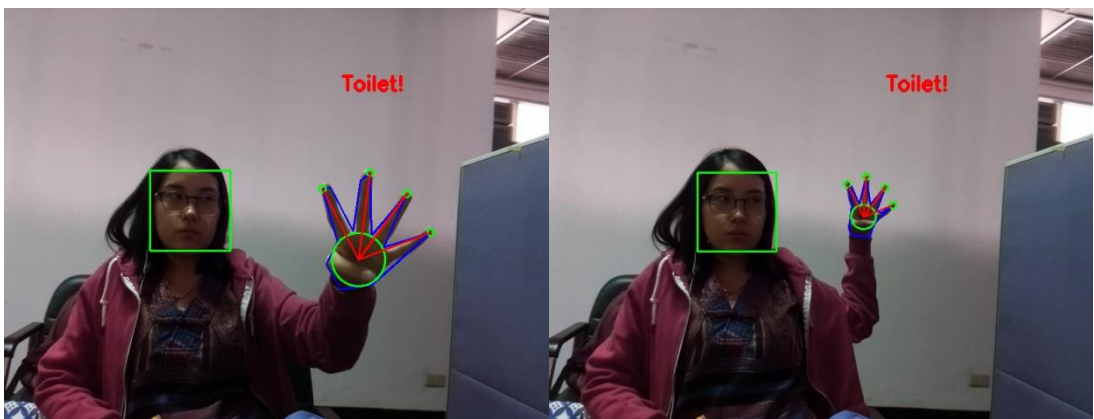
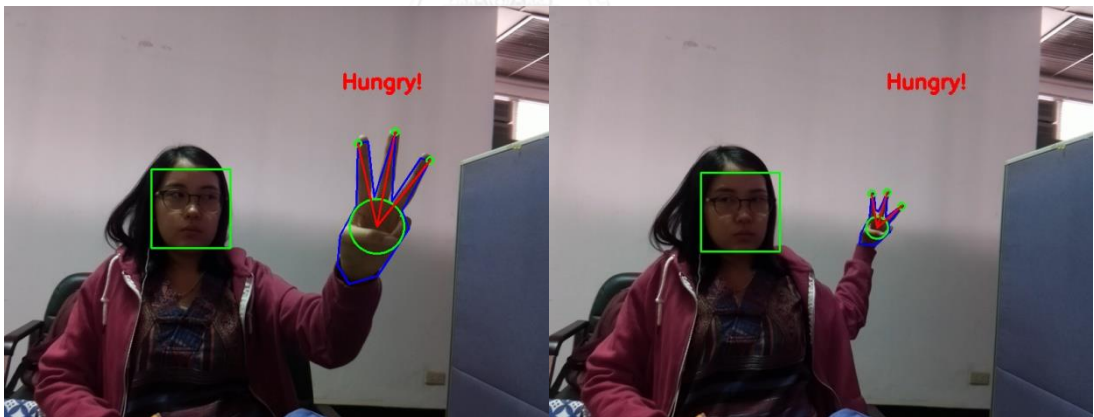
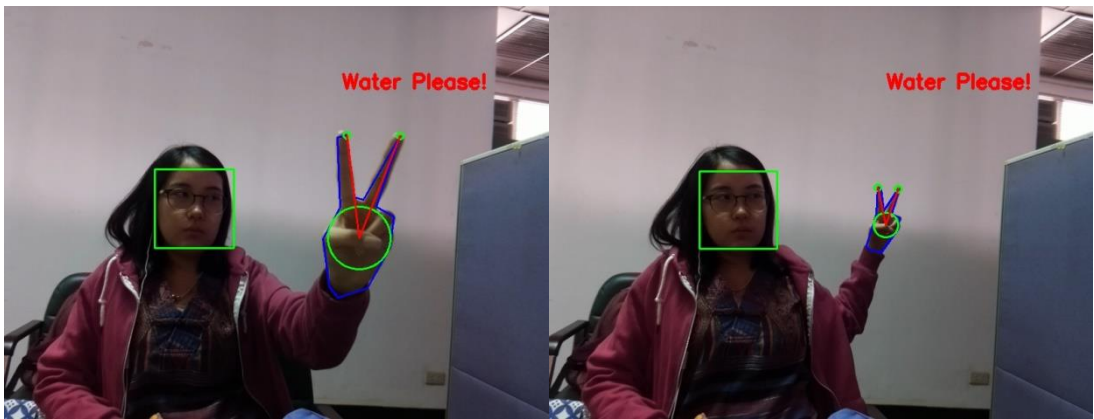
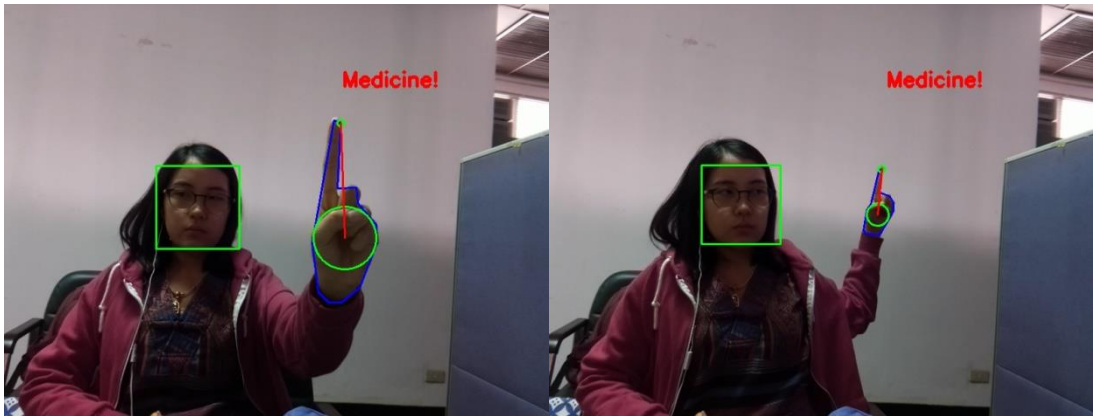
ท่ามือ	ค่าความแม่นยำของระยะห่างระหว่างกล้องและท่ามือ(%)		ค่าความแม่นยำรวม(%)
	0.5 - 1 เมตร	1 - 1.5 เมตร	0.5-1.5 เมตร
กำมือ	100	100	100
ชูนิ้วชี้	100	100	100
ชูนิ้วชี้และนิ้วกลาง	100	97	98.5

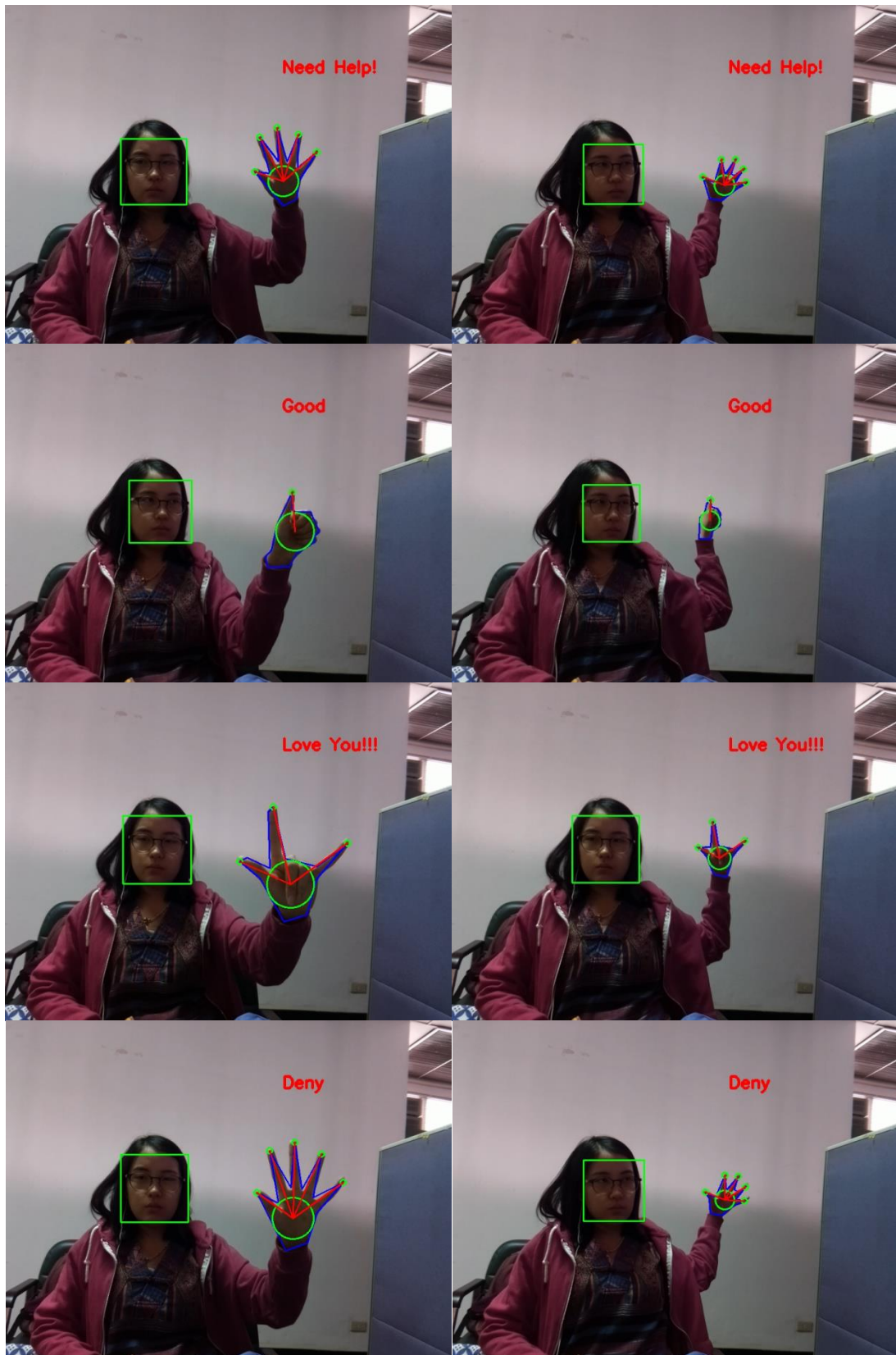


ท่ามือ	ค่าความแม่นยำของระยะห่างระหว่างกล้องและท่ามือ(%)		ค่าความแม่นยำรวม(%)
	0.5 - 1 เมตร	1 - 1.5 เมตร	0.5-1.5 เมตร
ชูนิ้วชี้ นิ้วกลางและนิ้วนาง	100	98	99
ชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย	100	97	98.5
แบมือ	100	92	96
ชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย	100	98	99
ชูนิ้วโป้ง	100	86	93
รวม	100	96	98

จากการทดลองพบว่าที่ระยะห่าง 0.5 - 1 เมตรนั้นระบบสามารถจำแนกท่ามือทุกท่ามือได้ในระดับดีมาก มีค่าความแม่นยำเท่ากับ 100 เปอร์เซ็นต์ สำหรับระยะห่าง 1 - 1.5 เมตร นั้นระบบมีความผิดพลาดตั้งแต่ท่าชูนิ้วชี้และนิ้วกลางไปจนถึงท่าชูนิ้วโป้ง ซึ่งโดยส่วนมากค่าความผิดพลาดจะเกิดที่ระยะห่างประมาณ 1.5 เมตร ทั้งนี้เกิดจาก 2 ปัจจัยหลัก ได้แก่ ค่าความละเอียดของกล้อง และสภาพแสง ซึ่งปัญหาสภาพแสงจะส่งผลกระทบต่อการทำงานของระบบทำให้ระบบไม่สามารถตรวจจับนิ้วมือบางนิ้วได้เช่นเดียวกับผลลัพธ์ที่ได้จากการทดลององศาที่แตกต่างกัน







รูปที่ 4.11 แสดงท่ามือเคลื่อนไหวในระยะเวลาที่แตกต่างกัน

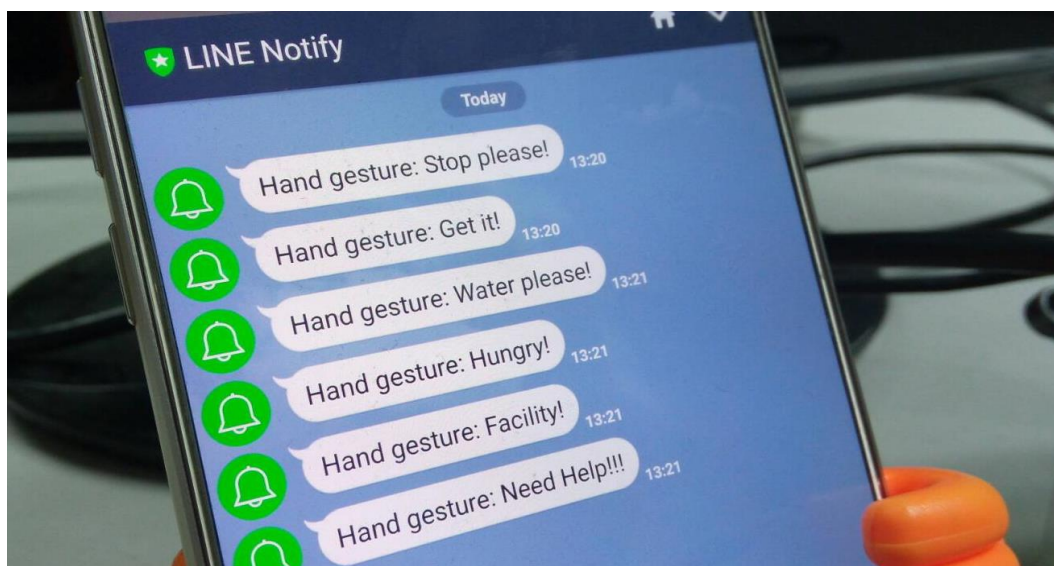
#### 4.3.6 การทดลองการส่งข้อความแจ้งเตือนบนสมาร์ตโฟน

เมื่อระบบประมวลผลในการจำแนกท่ามือแล้วระบบจะทำการส่งข้อความไปยังสมาร์ตโฟนผ่านทางแอปพลิเคชันไลน์ โดยทำการทดลองกับท่ามือทั้งหมด 9 ท่ามือ ได้แก่ ท่ากำมือ (หยุด) ท่าชูนิ้วชี้ (ยา) ท่าชูนิ้วชี้และนิ้วกลาง (น้ำ) ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง (หิว) ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย (ห้องน้ำ) ท่าแบมือ (ช่วยด้วย) ท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย (รักคุณ) ท่าชูนิ้วโป้ง (ดี) และท่าโบกมือ (ปฏิเสธ)

ตารางที่ 4.9 การทดลองการส่งข้อความแจ้งเตือนบนสมาร์ตโฟน

ท่ามือ	จำนวนครั้ง	จำนวนครั้งที่ส่งถูกต้อง	จำนวนครั้งที่ส่งผิด	ค่าความแม่นยำ (%)
กำมือ (หยุด)	30	30	0	100
ชูนิ้วชี้ (ยา)	30	30	0	100
ชูนิ้วชี้และนิ้วกลาง (น้ำ)	30	30	0	100
ชูนิ้วชี้ นิ้วกลางและนิ้วนาง (หิว)	30	30	0	100
ชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย (ห้องน้ำ)	30	30	0	100
แบมือ (ช่วยด้วย)	30	30	0	100
ชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย (รักคุณ)	30	30	0	100
โบกมือ (ปฏิเสธ)	30	30	0	100
<b>รวม</b>	<b>30</b>	<b>30</b>	<b>0</b>	<b>100</b>

จากการทดลองพบว่าตั้งนั้นจากการทดลองพบว่าระบบสามารถส่งข้อความแจ้งเตือนได้ถูกต้องตรงกับความหมายท่ามือที่ผ่านการจำแนกแล้วในระดับดีมากเท่ากับ 100 เปอร์เซ็นต์ ตัวอย่างข้อความแจ้งเตือนแสดงในรูปที่ 4.12



รูปที่ 4.12 ตัวอย่างการส่งข้อความแจ้งเตือนบนสมาร์ทโฟน

#### 4.4 การทดสอบกับผู้สูงอายุ

การทดสอบกับผู้สูงอายุที่มีอายุตั้งแต่ 60 ปีขึ้นไปเพื่อตรวจสอบว่าทำมือใดที่ผู้สูงอายุสามารถทำได้โดยง่าย โดยทำการทดสอบในสภาพแวดล้อมปิด แสดงสว่างมาจากหลอดไฟฟลูออเรสเซนต์ห่างจากบริเวณอ้างอิงที่ทำการทดสอบประมาณ 2.6 เมตร ตั้งกล้องให้หันหน้าตรงมาทางผู้ทดสอบ กำหนดการทดสอบทั้งหมด 10 คน คณะเพศหญิง (ญ) และเพศชาย (ช) แบ่งเป็น 3 ช่วงอายุ ดังนี้ กลุ่มผู้สูงอายุในช่วงอายุ 60-70 ปี 71-80 ปี และ 81-90 ปี โดยเกณฑ์การประเมินผลแบ่งออกเป็น 5 เกณฑ์ ได้แก่

ทำได้ในระดับดีมาก คือ 5

ทำได้ในระดับดี คือ 4

ทำได้ในระดับปานกลาง คือ 3

ทำได้ในระดับน้อย คือ 2

ทำได้ในระดับน้อยมาก คือ 1

ทำไม่ได้เลย คือ 0

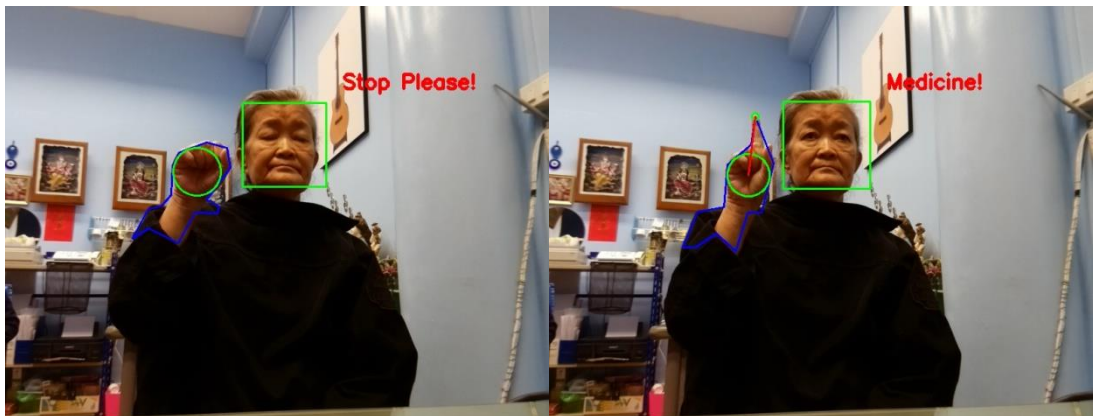
ทั้งนี้ค่าคะแนนสูงสุดเท่ากับ 5 หมายถึงผู้สูงอายุสามารถแสดงท่ามือได้อย่างถูกต้องตามท่ามือที่กำหนด คือสามารถกางนิ้วให้ห่างออกจากกันได้ในการทำมือที่มีการชูมากกว่าหนึ่งนิ้วขึ้นไป และสามารถแสดงเฉพาะนิ้วที่ต้องการเท่านั้น และหากเป็นท่าโบกมือก็สามารถโบกมือให้เข้าได้ แต่หาก







รูปที่ 4.13 ผู้สูงอายุหญิงวัย 60 ปี







รูปที่ 4.14 ผู้สูงอายุหญิงวัย 70 ปี





รูปที่ 4.15 ผู้สูงอายุชายวัย 84 ปี

จากผลการทดลองพบว่าผู้สูงอายุสามารถทำท่ามือที่ไม่มี ความซับซ้อนได้ดี ได้แก่ ท่ากำมือ ท่าชู นิ้วชี้ และท่าชูนิ้วโป้ง โดยท่าชูนิ้วโป้งเป็นท่าที่ผู้สูงอายุชอบมากที่สุด สำหรับท่าที่มีความซับซ้อนมากที่สุดสำหรับผู้สูงอายุคือท่ารักคุณ โดยผู้สูงอายุส่วนใหญ่ใช้เวลาในการจัดนิ้วกลางและนิ้วนางให้ตกลง ปัญหาที่พบคือ นิ้วแข็ง ไม่สามารถเก็บนิ้วได้หมด ทั้งนี้อาการนิ้วแข็งขึ้นอยู่กับผู้สูงอายุเป็นรายบุคคล ไป ดังเช่น ผู้สูงอายุหญิงวัย 70 ปีทำท่ารักคุณได้ในระดับน้อยมาก ในขณะที่ผู้สูงอายุชายวัย 84 ปี สามารถทำท่ารักคุณได้ในทันที ส่วนท่ามืออื่น ๆ ที่มีการแสดงนิ้วขึ้นยกเว้นท่าโบกมือมาจัดอยู่ในระดับ ดี ผู้สูงอายุสามารถแสดงท่าชูสองนิ้ว สามนิ้ว สี่นิ้ว และท่าแบมือได้ ทั้งนี้สำหรับท่าชูสองนิ้วและสาม นิ้วพบว่าผู้สูงอายุบางรายชูนิ้วมือติดกัน ทำให้เกิดความผิดพลาดในการตรวจจับในบางเฟรม ท่าโบก มือเป็นท่าที่ผู้สูงอายุสามารถแสดงท่ามือได้ถูกต้องแต่เคลื่อนที่เร็วเกินไป จึงทำให้ระบบตรวจจับไม่ได้ เนื่องจากท่ายีระบบใช้เวลาในการประมวลผลนานกว่าท่ามืออื่น ๆ เพราะต้องคำนวณระยะห่างต่อ เฟรม ดังนั้นสำหรับท่าโบกมือระบบจะสามารถจำแนกท่ามือได้ถูกต้องเมื่อผู้สูงอายุเคลื่อนที่ช้า ๆ อย่างต่อเนื่องเท่านั้น

#### 4.5 การเปรียบเทียบค่าความแม่นยำของการจำแนกท่ามือ

ทำการเปรียบเทียบค่าความแม่นยำในการจำแนกท่ามือขณะอยู่นิ่งบนพื้นหลังแบบไม่ซับซ้อนของผลที่ได้จากอัลกอริธึมที่นำเสนอในหัวข้อ 3.2.3 กับอัลกอริธึมในหัวข้อ 3.2.2 และอัลกอริธึมที่เสนอโดย [30] โดยใช้การค้นหาเส้นรอบขอบและ HOG โดยใช้ชั้นการเรียนรู้ 27 ชั้น การเปรียบเทียบค่าความแม่นยำในการจำแนกท่ามือของระบบแสดงในตารางที่ 4.11

ตารางที่ 4.11 การเปรียบเทียบค่าความแม่นยำในการจำแนกท่ามือขณะอยู่นิ่งบนพื้นหลังแบบไม่ซับซ้อน

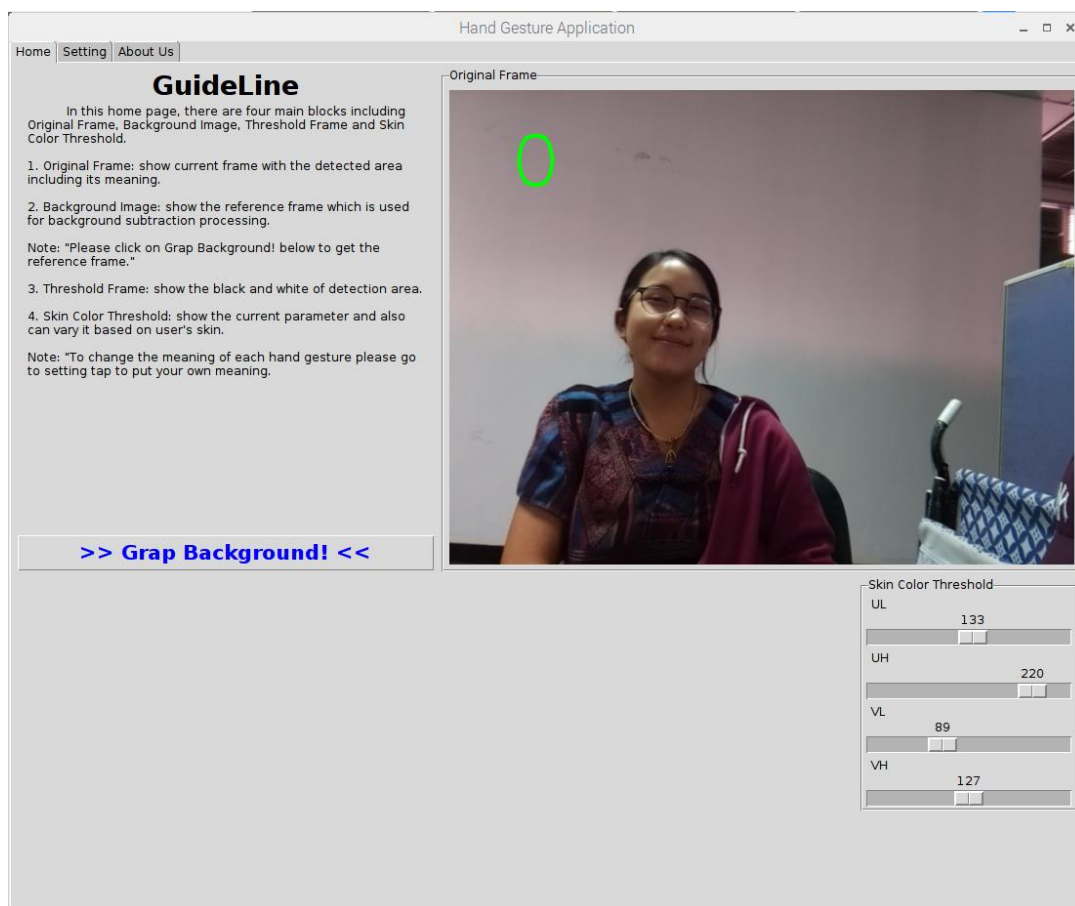
ท่ามือ	ค่าความแม่นยำ (%)		
	[30]	ผลการทดลอง อัลกอริธึมในหัวข้อ 3.2.2	ผลการทดลอง อัลกอริธึมที่นำเสนอ ในหัวข้อ 3.2.3
กำมือ	99.8	100	100
ชูนิ้วชี้	99.61	93.55	100
ชูนิ้วชี้และนิ้วกลาง	100	61.76	100
ชูนิ้วชี้ นิ้วกลางและนิ้วนาง	-	54.76	100
ชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อย	-	87.8	100
แบมือ	-	92.62	100
ชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย	-	-	100
ท่าชูนิ้วโป้ง	-	-	100
<b>รวม</b>	<b>99.8</b>	<b>80.86</b>	<b>100</b>

จากตารางที่ 4.11 พบว่าในการทดลองบนฉากหลังแบบเรียบนั้นอัลกอริธึมที่นำเสนอในหัวข้อ 3.2.3 มีค่าความแม่นยำสูงกว่าอัลกอริธึมในหัวข้อ 3.2.2 ที่มีค่าเท่ากับ 80.86 เปอร์เซ็นต์ และอัลกอริธึม [30] ที่มีค่าเท่ากับ 99.8 เปอร์เซ็นต์ แสดงว่าอัลกอริธึมที่นำเสนอมีประสิทธิภาพในการจำแนกท่ามือมากกว่า

#### 4.4 การออกแบบหน้าจออินเตอร์เฟซ

การทำงานของระบบได้รับการออกแบบหน้าจออินเตอร์เฟซ (Graphical User Interface: GUI) เพื่อแสดงท่ามือให้ผู้สูงอายุได้รับรู้การแสดงท่ามือของตน อีกทั้งยังสามารถตั้งค่าความหมายที่ต้องการให้แต่ละท่ามือได้ด้วยตนเองผ่านทางหน้าจออินเตอร์เฟซ หน้าจออินเตอร์เฟซแบ่งออกเป็น 3 แถบเมนูบาร์ ได้แก่ แถบหน้าหลัก (Home) แถบตั้งค่า (Settings) และแถบเกี่ยวกับเรา (About us) โดยมีรายละเอียดการใช้งานดังนี้

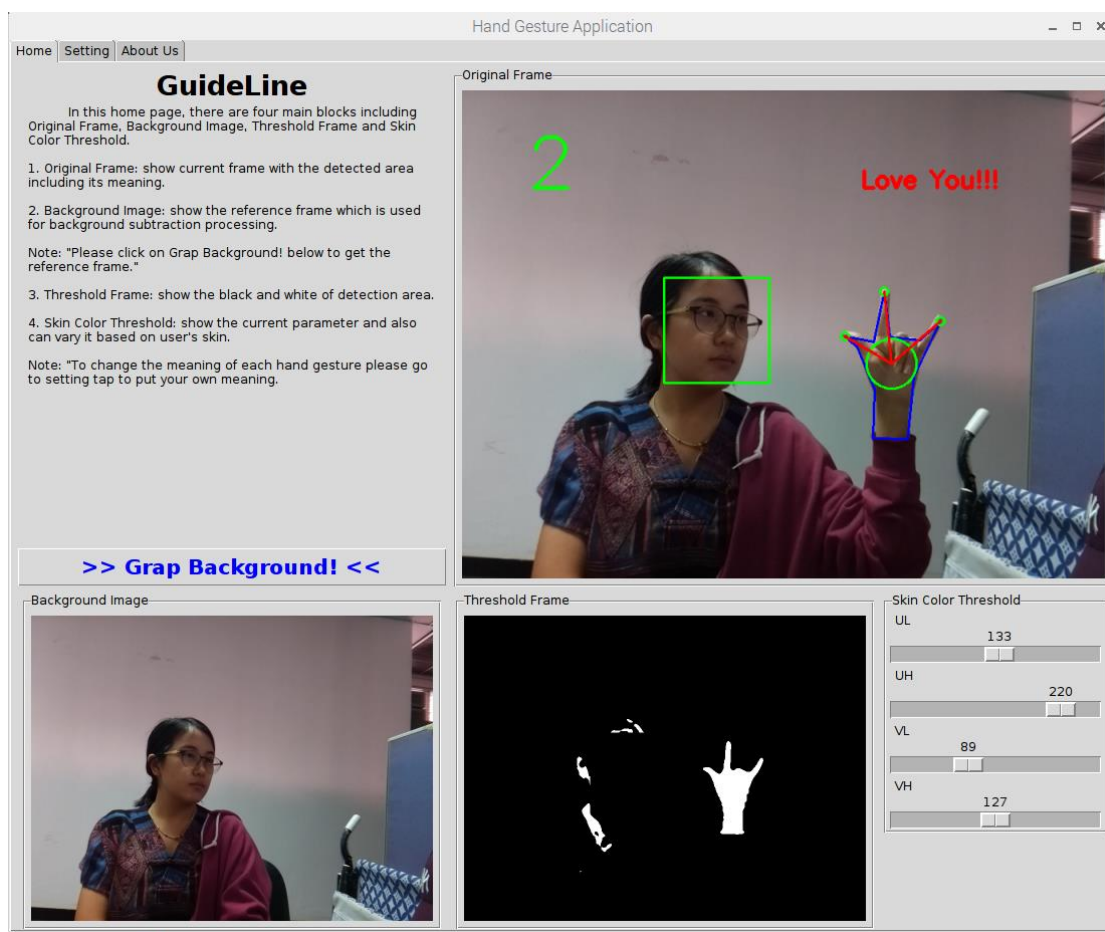
ในแถบหน้าหลัก (Home) จะมีคำแนะนำ (guideline) ในการทำงานอยู่ทางด้านซ้ายมือบน ดังแสดงในรูปที่ 4.13



รูปที่ 4.16 แสดงแถบหน้าหลักของหน้าจออินเตอร์เฟซของระบบก่อนการจับภาพฉากหลัง

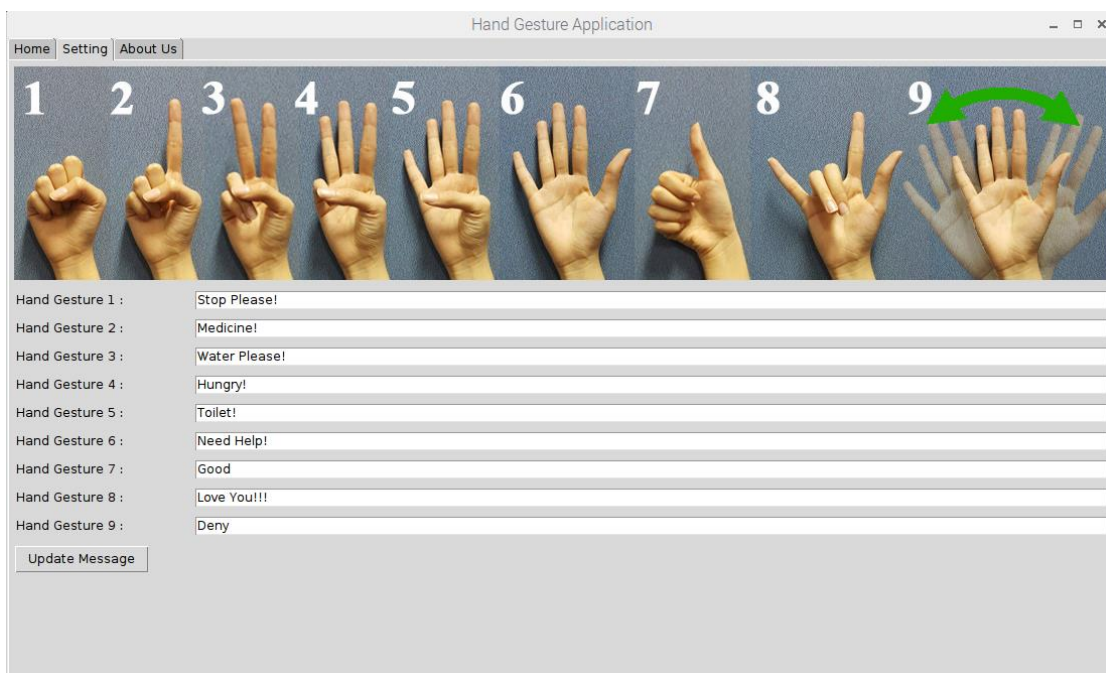
เมื่อผู้ใช้งานต้องการเริ่มต้นการทำงานของระบบ ผู้ใช้งานสามารถเริ่มต้นการทำงานได้โดยการกดปุ่ม Grab background ซึ่งระบบจะ capture ภาพนั้น ๆ เป็นภาพพื้นหลัง โดยจะแสดงที่หน้าต่างด้านขวาบน หลังจากนั้นผู้ใช้งานสามารถแสดงท่ามือได้อย่างอิสระ โดยสามารถปรับค่าเทรชโฮลด์ให้เหมาะสมได้ กำหนดให้สามารถปรับช่วงของค่าเทรชโฮลด์ของการตรวจจับสีผิวของค้ประกอบสี

U และ V ในช่วง 0-255 สำหรับองค์ประกอบ U กำหนดให้ UL คือ ค่าเทรชโฮลด์ขั้นต่ำขององค์ประกอบ U มีค่า Default เท่ากับ 133 และ UH คือ ค่าเทรชโฮลด์ขั้นสูงขององค์ประกอบ U มีค่า Default เท่ากับ 220 ในทางเดียวกันสำหรับองค์ประกอบ V ให้ VL คือ ค่าเทรชโฮลด์ขั้นต่ำขององค์ประกอบ V มีค่า Default เท่ากับ 89 และ VH คือ ค่าเทรชโฮลด์ขั้นต่ำขององค์ประกอบ V มีค่า Default เท่ากับ 127 ดังแสดงในรูปที่ 4.14



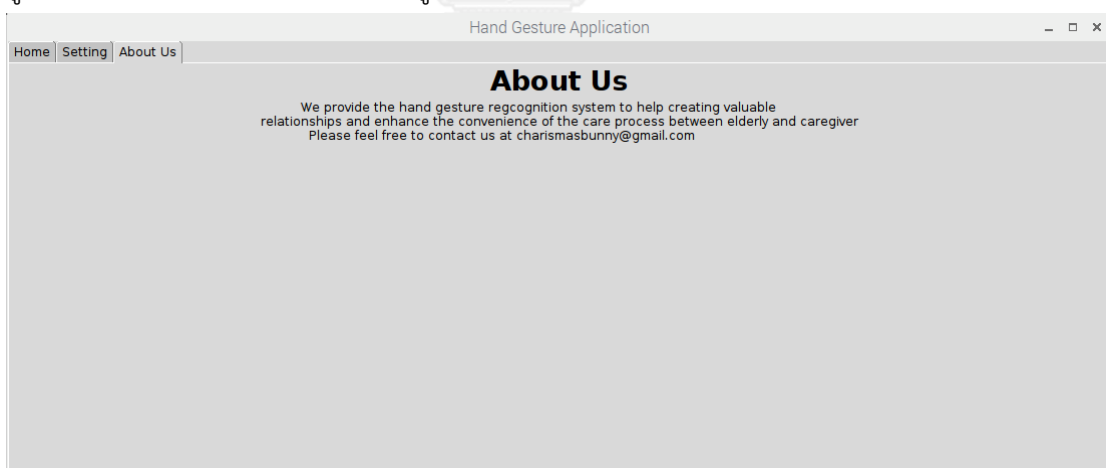
รูปที่ 4.17 แสดงแถบหน้าหลักของหน้าจ่อินเตอร์เฟซของระบบหลังการจับภาพฉากหลัง

สำหรับแถบที่สอง คือ แถบการตั้งค่า (Settings) ซึ่งจะแสดงท่ามือทั้งหมดของระบบและอนุญาตให้ผู้ใช้สามารถปรับเปลี่ยน แก๊ซ และ/หรือ กำหนดความหมายใหม่ได้ เมื่อผู้ใช้พิมพ์ข้อความที่ต้องการลงในช่องสีขาวแล้วเรียบร้อยให้กดปุ่มอัปเดตข้อความ (Update message) เมื่อกดปุ่มเรียบร้อยแล้วระบบจะทำการอัปเดตข้อความโดยอัตโนมัติ ซึ่งเมื่อผู้ใช้คลิกกลับมาที่แถบหน้าหลักและแสดงท่ามือที่ได้รับการเปลี่ยนความหมาย ระบบจะแสดงข้อความใหม่ในทันทีโดยที่ไม่ต้องเริ่มรันโปรแกรมใหม่ แถบหน้าการตั้งค่าแสดงดังรูปที่ 4.15



รูปที่ 4.18 แสดงแถบการตั้งค่าของหน้าจ่อินเตอร์เฟซของระบบ

สำหรับแถบสุดท้าย คือแถบเกี่ยวกับเรา ผู้พัฒนาได้ใส่ข้อมูลการติดต่อเพื่อเป็นประโยชน์ต่อผู้ใช้งาน แถบหน้าเกี่ยวกับเราแสดงในรูปที่ 4.16



รูปที่ 4.19 แสดงแถบเกี่ยวกับเราของหน้าจ่อินเตอร์เฟซของระบบ

## บทที่ 5

### สรุป อภิปรายผล และข้อเสนอแนะ

งานวิจัยนี้นำเสนอการพัฒนาอัลกอริทึมของระบบรู้จำท่าทางมือแบบเคลื่อนไหวมาประยุกต์ใช้กับระบบสมองกลฝังตัวโดยใช้บอร์ด Raspberry pi 3 Model B เพื่อพัฒนาการตรวจจับและจำแนกท่าทางมือ ผู้พัฒนาทำการพัฒนาอัลกอริทึมบนพื้นฐานการรับรู้สำหรับผู้สูงอายุทั้งหมด 3 อัลกอริทึม รวมถึงการอภิปรายผล และข้อเสนอแนะดังที่ได้สรุปไว้ในหัวข้อ 5.1 5.2 และ 5.3 ดังนี้

#### 5.1 สรุปผลการทดลอง

##### 5.1.1 อัลกอริทึมรู้จำท่าทางมือแบบสถิติสำหรับผู้สูงอายุโดยใช้กฎการจำแนก

จากผลการทดลองระบบรู้จำท่าทางมือแบบสถิติสำหรับผู้สูงอายุโดยใช้กฎการจำแนกในหัวข้อ 3.2.1 แสดงให้เห็นว่าระบบสามารถตรวจจับและจำแนกท่าทางมือตามความหมายที่กำหนดไว้จำนวน 6 ท่า ได้แก่ ท่ากำมือหมายความว่า “หยุด” ท่าชูนิ้วชี้หมายความว่า “ห้องน้ำ” ท่าชูนิ้วชี้และนิ้วกลางหมายความว่า “ตกลง” ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนางหมายความว่า “อาหาร” ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อยหมายความว่า “น้ำ” และท่าแบมือหมายความว่า “ช่วยด้วย” ในความหมายและองค์ประกอบที่แตกต่างกันได้ ระบบทำงานได้บนพื้นหลังใดๆที่ไม่มีวัตถุหรือสิ่งกีดขวางโดยรับภาพจากกล้องที่ติดตั้งในแนวตั้ง และทำการตรวจจับและจำแนกท่าทางมือตามกฎการจำแนก

อัลกอริทึมนี้ใช้วิธีการหาเส้นขอบสำหรับตรวจจับเส้นขอบของมือ คอนเวกซ์ฮัลสำหรับหาจุดปลายนิ้วมือ จุดลึกสุดหรือ Convexity defects สำหรับหาจุดลึกสุดในคอนเวกซ์ฮัลหรือจุดลึกสุดที่อยู่ระหว่างนิ้ว การหาโมเมนต์ภาพเพื่อหาจุดกลางฝ่ามือ และการใช้กฎการจำแนกสำหรับการจำแนกท่าทางมือ ซึ่งพบว่าการใช้กฎการจำแนกให้ผลลัพธ์ที่แม่นยำ สะดวกต่อการใช้งาน และรวดเร็ว แต่การหาจุดลึกสุดหรือ Convexity defects ส่งผลเสียต่อการแสดงท่ามือในมุมมองที่ซับซ้อน (Occlusions) ซึ่งเมื่อระบบไม่สามารถตรวจจับจุดลึกสุดได้ ระบบจะไม่สามารถตรวจจับจำนวนนิ้วมือได้เช่นกัน นอกจากนี้ในอัลกอริทึมนี้ยังมีการทำกระบวนการประมวลผลเบื้องต้นน้อย (Pre-processing) และยังไม่ดีเท่าที่ควร ทำให้ระบบยังมีปัญหาเรื่องสภาพแสง ระยะห่างระหว่างกล้องและมือ ความอิสระในการแสดงท่ามือเนื่องจากต้องแสดงท่ามือภายในกรอบสี่เหลี่ยมที่กำหนด รวมถึงจะต้องแสดงเฉพาะท่ามือให้กล้องเท่านั้น เพราะระบบไม่สามารถจำแนกท่ามือได้หากมีใบหน้าหรือลำตัวเข้ามาในฉากหน้า

### 5.1.2 อัลกอริธึมรู้จำท่ามือแบบสถิติสำหรับผู้สูงอายุโดยใช้โดยใช้ Optical Flow และ SURF

จากผลการทดลองอัลกอริธึมรู้จำท่ามือสำหรับผู้สูงอายุโดยใช้ Optical Flow และ SURF ในหัวข้อ 3.2.1 แสดงให้เห็นว่าระบบสามารถตรวจจับ ติดตาม และจำแนกท่ามือแบบเคลื่อนไหวตามความหมายที่กำหนดไว้จำนวน 6 ท่า ได้แก่ ท่ากำมือหมายความว่า "ช่วยด้วย" ท่าชูนิ้วชี้หมายความว่า "น้ำ" ท่าชูนิ้วชี้และนิ้วกลางหมายความว่า "อาหาร" ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนางหมายความว่า "ห้องน้ำ" ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อยหมายความว่า "อำนวยความสะดวก" และท่าแบมือหมายความว่า "อุบัติเหตุ" ได้ในระดับดีแม้ว่าท่ามือจะเบลอเนื่องจากการเคลื่อนไหว โดยมีความแม่นยำโดยรวมเท่ากับเท่ากับ 80.86 เปอร์เซ็นต์ จากทั้งหมด 2,200 เฟรม โดยท่ากำมือจะมีความแม่นยำมากที่สุดเท่ากับ 100 เปอร์เซ็นต์ เนื่องจากรูปร่างของท่ากำมือมีความซับซ้อนน้อยที่สุดสำหรับท่าชูนิ้วชี้และนิ้วกลาง และท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง มีค่าความแม่นยำต่ำเท่ากับ 61.76 เปอร์เซ็นต์ และ 54.76 เปอร์เซ็นต์ เนื่องจากนิ้วมือของทั้งสองท่ามือมีความใกล้เคียงกัน อย่างไรก็ตามระบบแสดงค่าความแม่นยำสำหรับท่าชูนิ้วชี้เท่ากับ 93.55 เปอร์เซ็นต์ ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อยเท่ากับ 87.8 เปอร์เซ็นต์และท่าแบมือเท่ากับ 92.62 เปอร์เซ็นต์

สำหรับวิธีหลักที่ใช้ในอัลกอริธึม คือ การไหลของแสง (optical flow) สำหรับตรวจจับเวกเตอร์เคลื่อนที่ของท่ามือเคลื่อนไหว การวิเคราะห์blob (blob analysis) สำหรับตรวจจับและติดตามท่ามือ และ SURF สำหรับการจำแนกท่ามือ อย่างไรก็ตามค่าความผิดพลาดที่เกิดขึ้นเกิดจากเงาของมือขณะที่เคลื่อนที่บนฉากหลัง รวมถึงกระบวนการประมวลผลเบื้องต้นน้อย (Pre-processing) ทำให้ระบบยังมีปัญหาเรื่องสภาพแสงและเงาซึ่งเป็นสัญญาณรบกวนของระบบ นอกจากนี้ยังมีปัญหาเกี่ยวกับการใช้การไหลของแสง นั่นคือ ระบบจะให้ค่าความผิดพลาด เมื่อหยุดการเคลื่อนที่ของท่ามือ ทำให้ต้องแสดงการเคลื่อนที่ท่ามือโดยตลอดการทดลอง และการแสดงท่ามือจะต้องแสดงเฉพาะท่ามือให้กล้องเท่านั้น เพราะระบบไม่สามารถจำแนกท่ามือได้หากมีใบหน้าหรือลำตัวเข้ามาในฉากหน้าเหมือนกับปัญหาของอัลกอริธึมรู้จำท่ามือแบบสถิติสำหรับผู้สูงอายุที่ใช้กฎการจำแนก

### 5.1.3 อัลกอริทึมที่นำเสนอสำหรับระบบรู้จำท่าบนพื้นฐานการมองเห็นสำหรับผู้สูงอายุ

จากผลการทดลองอัลกอริธึมที่นำเสนอแสดงให้เห็นว่าระบบสามารถตรวจจับ ติดตาม และจำแนกท่ามือแบบเคลื่อนไหวตามความหมายที่กำหนดไว้จำนวน 9 ท่า ได้แก่ ท่ากำมือ คือ "หยุด" ท่าชูนิ้วชี้ คือ "ยา" ท่าชูนิ้วชี้และนิ้วกลาง คือ "น้ำ" ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง คือ "หิว" ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย คือ "ห้องน้ำ" ท่าแบมือ คือ "ช่วยด้วย" ท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย คือ "รักคุณ" ท่าชูนิ้วโป้ง คือ "ดี" และท่าโบกมือ คือ "ปฏิเสธ" ได้ในระดับดีมาก โดย



ผู้พัฒนาออกแบบให้ผู้ใช้สามารถสั่งการใช้งานด้วยมือข้างที่ถนัด เพื่อให้เหมาะสมต่อการใช้งานจริง สำหรับผู้สูงอายุ ดังนั้นจึงสามารถสั่งการด้วยมือขวาหรือซ้ายก็ได้ นอกจากนี้ยังสามารถปรับเปลี่ยนความหมายท่ามือ และปรับค่าเทรซโฮลต์ให้เหมาะสมกับสภาพแสงหรือสภาพแวดล้อมที่เปลี่ยนไปได้ ตามที่ต้องการผ่านทางหน้าจอสัมผัสที่ออกแบบมาเพื่ออำนวยความสะดวก โดยจะแสดงท่ามือพร้อมความหมายให้ผู้ใช้งานเห็น อีกทั้งยังเป็นการตรวจสอบว่าท่ามือที่ระบบแสดงให้เห็นนั้นเป็นไปตามที่ต้องการหรือไม่ สำหรับความสามารถของระบบ พบว่าระบบสามารถตรวจจับและจำแนกท่าทางมือแบบเคลื่อนไหวได้แม้ฉากหลังมีความซับซ้อน สามารถเคลื่อนที่ท่ามือในตำแหน่งต่าง ๆ อาทิ ขึ้น ลง ซ้าย ขวา หรือเคลื่อนไปรูปวงกลม สีเหลี่ยม เป็นต้น นอกจากนี้ยังสามารถเอียงมือและหมุนมือในองศาต่าง ๆ สูงสุดที่ 65 องศา และเปลี่ยนระยะได้สูงสุดที่ 1.5 เมตร และสามารถส่งข้อความแจ้งเตือนไปยังสมาร์ตโฟนได้ทันที

สำหรับอัลกอริทึมที่ใช้ในระบบนี้ ผู้พัฒนานำข้อดีจากการใช้วิธีการต่าง ๆ ที่ใช้ในอัลกอริทึมในหัวข้อ 3.2.1 และ 3.2.2 มาใช้กับระบบ โดยพบว่าควรมีการใช้วิธีการประมวลผลเบื้องต้นในขั้นตอนแรก ๆ เพื่อลดสัญญาณรบกวนและปรับภาพให้เรียบเนียนขึ้น อาทิ การใช้ตัวกรองเกาส์เซียน การทำเทรซโฮลต์ในแต่ละองค์ประกอบสี แพลงโมเดลสี เป็นต้น จากนั้นจึงนำวิธีการลบพื้นหลังซึ่งให้ผลลัพธ์ที่ดีมาใช้ในการหามือเพื่อแยกออกเป็นส่วนของฉากหน้า แล้วใช้วิธีการตรวจจับใบหน้าเพื่อลบใบหน้าออกจากฉากหน้า การตรวจจับสีผิวเพื่อแยกออกจากเสื้อผ้าและองค์ประกอบอื่น ๆ จากนั้นจึงหาเส้นขอบของมือ คอนเวกซ์ฮัลล์สำหรับหาจุดปลายนิ้วมือ หาวงกลมสูงสุดภายในเพื่อหาจุดกลางฝ่ามือ และการใช้กฎการจำแนกสำหรับการจำแนกท่ามือที่สะดวกและรวดเร็ว

โดยในการทดลองของระบบนี้กำหนดให้แสงสว่างจากหลอดไฟลูออเรสเซนต์ ติดตั้งห่างจากบริเวณอ้างอิงที่แสดงท่ามือที่ระยะประมาณ 3 เมตร จากผลการทดลองพบว่าเมื่อทดลองท่ามือขณะอยู่หนึ่งบนพื้นหลังแบบเรียบนั้น ระบบให้ค่าความแม่นยำสูงสุดเท่ากับ 100 เปอร์เซ็นต์ เมื่อทดลองท่ามือขณะอยู่หนึ่งบนพื้นหลังซับซ้อนระบบให้ความแม่นยำเท่ากับ 99.75 เปอร์เซ็นต์ แต่หากทดลองท่ามือเคลื่อนไหวในตำแหน่งต่าง ๆ บนพื้นหลังแบบเรียบและซับซ้อนนั้น ระบบให้ค่าความแม่นยำเท่ากับ 99.69 เปอร์เซ็นต์ และ 99 เปอร์เซ็นต์ตามลำดับ สำหรับการทดสอบท่ามือบนพื้นหลังแบบเรียบ เมื่อทดสอบท่ามือเคลื่อนไหวในองศาต่าง ๆ กันตั้งแต่ -65 ถึง 65 พบว่าค่าความแม่นยำของระบบเท่ากับ 99.75 เปอร์เซ็นต์ เมื่อทดสอบท่ามือเคลื่อนไหวในมุมมองตั้งแต่ 20 ถึง -65 พบว่าค่าความแม่นยำของระบบเท่ากับ 99.65 เปอร์เซ็นต์ และเมื่อทดสอบท่ามือเคลื่อนไหวในระยะ 0.5 เมตร ถึง 1.5 เมตร พบว่าค่าความแม่นยำของระบบเท่ากับ 98 เปอร์เซ็นต์ เมื่อทดสอบการส่งข้อความแจ้งเตือนบนสมาร์ตโฟนพบว่าค่าความแม่นยำของระบบเท่ากับ 100 เปอร์เซ็นต์ ระบบมีประสิทธิภาพในการจำแนกท่ามือโดยรวมเท่ากับ 99.48 เปอร์เซ็นต์

## 5.2 อภิปรายผล

จากการทดลองการใช้งานระบบ พบว่าระบบนี้มีข้อจำกัดอยู่ 3 ประการ นั่นคือ 1) ผู้ใช้งานต้องหลีกเลี่ยงฉากหลังที่มีสีใกล้เคียงกับสีผิว 2) การสวมใส่เสื้อแขนยาว และ 3) การใช้ระบบในที่ที่มีแสงสว่างเพียงพอ โดยระบบนี้ไม่สามารถทดสอบบนฉากหลังที่มีสีเหมือนสีผิวมนุษย์ อาทิ โทนสีเหลือง ส้มอ่อน ทอง น้ำตาลอ่อน เป็นต้น รวมถึงหลีกเลี่ยงการใส่เสื้อผ้าที่มีสีดังกล่าว และผู้ใช้งานจะต้องสวมใส่เสื้อแขนยาวที่คลุมบริเวณข้อศอกเพื่อให้ระบบพิจารณาเฉพาะผิวของมือเท่านั้น เนื่องจากระบบนี้ใช้อัลกอริทึมในการตรวจจับสีผิวและการหาวงกลมที่ใหญ่ที่สุดภายในเส้นขอบ ซึ่งบริเวณข้อศอกมีขนาดใหญ่ใกล้เคียงกับบริเวณฝ่ามือ ทำให้ระบบสามารถตรวจจับผิวของบริเวณข้อศอกเป็นวงกลมของฝ่ามือได้และจะทำการให้ตรวจจับมือและจำแนกท่ามือผิด ดังนั้นเพื่อให้ระบบจำแนกท่ามือได้อย่างมีประสิทธิภาพ ผู้ใช้งานจะต้องสวมใส่เสื้อแขนยาวในการใช้งาน หรือในกรณีไม่สวมใส่เสื้อแขนยาว ผู้ใช้งานก็ต้องไม่แสดงผิวบริเวณข้อศอกต่อหน้ากล้อง รวมถึงการใช้ระบบในพื้นที่ที่มีแสงสว่างเพียงพอโดยเฉพาะการใช้กับพื้นหลังแบบซับซ้อน เพื่อให้เกิดความแตกต่างของสีของท่ามือและฉากหลังที่ชัดเจน ซึ่งจะทำให้ระบบตรวจจับและจำแนกท่ามือได้อย่างถูกต้อง

ในการทดสอบระบบได้ทำการแบ่งการทดลองออกเป็น 6 การทดลองหลัก ได้แก่ 1) การทดลองท่ามือขณะอยู่นิ่ง 2) การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่ต่างกัน 3) การทดลองท่ามือเคลื่อนไหวในองศาที่ต่างกัน 4) การทดลองท่ามือเคลื่อนไหวในมุมมองที่ต่างกัน 5) การทดลองท่ามือเคลื่อนไหวในระยะที่ต่างกัน และ 6) การทดลองการส่งข้อความแจ้งเตือนบนสมาร์ทโฟน โดยที่การทดลองที่ 1) และ 2) สามารถแบ่งแต่ละการทดลองออกเป็น 2 การทดลองย่อย คือ การทดลองบนพื้นหลังเรียบและซับซ้อน ดังนั้นการทดสอบระบบมีทั้งหมด 8 เงื่อนไข สามารถอภิปรายได้ดังนี้

### 5.2.1 ท่ามือขณะอยู่นิ่ง

การทดลองท่ามือขณะอยู่นิ่งแบ่งออกเป็น 2 การทดลอง ได้แก่ การทดลองบนฉากหลังแบบเรียบและแบบซับซ้อน โดยทดสอบท่ามือทั้งหมด 8 ท่ามือในทั้งสองการทดลอง ได้แก่ ท่ากำมือ คือ “หยุด” ท่าชูนิ้วชี้ คือ “ยา” ท่าชูนิ้วชี้และนิ้วกลาง คือ “น้ำ” ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง คือ “หิว” ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย คือ “ห้องน้ำ” ท่าแบมือ คือ “ช่วยด้วย” ท่าชูนิ้วโป่ง นิ้วชี้และนิ้วก้อย คือ “รักคุณ” และท่าชูนิ้วโป่ง คือ “ดี” โดยจะยกเว้นการทดสอบท่าโบกมือในหัวข้อนี้ เนื่องจากต้องมีการเคลื่อนที่ของมือ กำหนดการทดสอบให้ท่ามือละ 300 เฟรม

สำหรับการทดลองบนพื้นหลังแบบเรียบพบว่าระบบให้ค่าความแม่นยำในการจำแนกความหมายในระดับดีมาก คือเท่ากับ 100 เปอร์เซ็นต์สำหรับท่ามือ 8 ท่า จากจำนวนเฟรมทั้งหมด 2,400 เฟรม สำหรับการทดลองบนพื้นหลังแบบซับซ้อนพบว่าท่าที่มีความซับซ้อนน้อยหรือมีจำนวน

นิ้วที่แสดงน้อยจะให้ค่าความแม่นยำในการจำแนกความหมายสูง โดยทำกำมือ ทำชุนิ้วชี้ ทำชุนิ้วชี้และนิ้วกลาง และทำชุนิ้วโป้ง มีค่าความแม่นยำในการจำแนกเท่ากับ 100 เปอร์เซ็นต์ สำหรับค่าความแม่นยำรองลงมา ได้แก่ ทำชุนิ้วโป้ง นิ้วชี้และนิ้วก้อย ทำชุนิ้วชี้ นิ้วกลางและนิ้วนาง และทำชุนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย เท่ากับ 99.67 เปอร์เซ็นต์ โดยทำแบมือมีความแม่นยำน้อยที่สุดเท่ากับ 99 เปอร์เซ็นต์ ดังนั้นจะเห็นว่ายังมีจำนวนนิ้วน้อยและระยะห่างของแต่ละนิ้วมากระบบยิ่งให้ค่าความแม่นยำดี ดังเห็นจากทำแบมือเนื่องจากเป็นท่าที่มีความซับซ้อนมากที่สุด คือมีจำนวนนิ้วที่ชูขึ้นมากที่สุดและระยะห่างระหว่างนิ้วแต่ละนิ้วน้อยนั่นเอง

นอกจากนี้ปัจจัยที่ส่งผลต่อค่าความแม่นยำของระบบ คือ สภาพแสงและการเคลื่อนที่ สำหรับการทดลองบนพื้นหลังแบบเรียบและแบบซับซ้อน แต่ในการทดลองบนพื้นหลังซับซ้อนจะต้องระวังสีที่ฉากหลังมากขึ้น โดยโทนสีผิวยังคงเป็นสีต้องห้ามสำหรับระบบนี้ นอกจากนี้ในการทดสอบกับฉากหลังซับซ้อนจะต้องมีแสงสว่างเพียงพอ โดยหากทดสอบในบริเวณแสงน้อยจะทำให้ขั้นตอนการลบพื้นหลังให้ค่าเท่ากับ 0 นั่นคือพิกเซลสีดำ หมายความว่าระบบไม่สามารถตรวจจับนิ้วมือได้นั่นเอง ดังนั้นความหมายที่ได้จะผิดไป ตัวอย่างเช่น แสดงทำแบมือ แต่ระบบไม่สามารถตรวจจับนิ้วโป้งและนิ้วก้อยได้ ทำให้ระบบจำแนกเป็นท่ามือชุนิ้วชี้ นิ้วกลาง และนิ้วก้อยแทน

## 5.2.2 ท่ามือเคลื่อนไหวในตำแหน่งที่ต่างกัน

การทดลองท่ามือเคลื่อนไหวในตำแหน่งที่ต่างกันแบ่งออกเป็น 2 การทดลอง ได้แก่ การทดลองบนฉากหลังแบบเรียบและแบบซับซ้อน โดยทั้งสองการทดลองจะทำการทดลองท่ามือเคลื่อนไหวในตำแหน่งที่แตกต่างกัน ได้แก่ การเคลื่อนที่ขึ้น ลง ซ้าย และขวาบนพื้นหลังแบบเรียบและทั้งหมด 8 ท่ามือ ได้แก่ ทำกำมือ คือ “หยุด” ทำชุนิ้วชี้ คือ “ยา” ทำชุนิ้วชี้และนิ้วกลาง คือ “น้ำ” ทำชุนิ้วชี้ นิ้วกลางและนิ้วนาง คือ “หิว” ทำชุนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย คือ “หื่อน้ำ” ทำแบมือ คือ “ช่วยด้วย” ทำชุนิ้วโป้ง นิ้วชี้และนิ้วก้อย คือ “รักคุณ” และทำชุนิ้วโป้ง คือ “ดี” และท่าโบกมือ “ปฏิเสธ” กำหนดการทดลองให้ท่ามือละ 100 เฟรม

จากการทดลองพบว่าท่ากำมือยังคงให้ค่าความแม่นยำที่สูงที่สุดคือเท่ากับ 100 เปอร์เซ็นต์ สำหรับทั้งการทดสอบบนพื้นหลังแบบเรียบและแบบซับซ้อน ค่าความแม่นยำรองลงมาเท่ากับ 99.75 เปอร์เซ็นต์สำหรับการทดสอบบนพื้นหลังแบบเรียบ และเท่ากับ 99.25 เปอร์เซ็นต์สำหรับการทดสอบบนพื้นหลังแบบซับซ้อน โดยค่าความผิดพลาดเกิดจากสภาพแสงและการเคลื่อนที่ โดยเฉพาะการเคลื่อนที่ทางขวาและซ้าย รองลงมาคือการเคลื่อนที่ขึ้น สามารถสรุปปัจจัยการเกิดค่าความผิดพลาดตามทิศทางการเคลื่อนที่ได้ดังนี้

- ค่าความผิดพลาดที่เกิดจากการเคลื่อนที่ขึ้น เกิดจากสภาพแสดงด้านบนจะแตกต่างจากระดับอ้างอิงที่ทดลอง โดยในห้องทดลองจะเห็นว่ายิ่งเคลื่อนที่ขึ้นสูงจะยังมีแสงสว่างมากขึ้นเพราะเข้าใกล้หลอดไฟ ซึ่งแสงที่มากไปก็ส่งผลให้เกิดความผิดพลาดในการตรวจจับสีผิว ผลที่ได้คือความยาวของนิ้วจะสั้นลง ระบบจึงไม่สามารถตรวจจับนิ้วมือได้ครบทุกนิ้ว โดยเฉพาะนิ้วก้อยและนิ้วโป้งที่สั้นกว่านิ้วอื่น ๆ นอกจากนี้ระยะห่างระหว่างนิ้วถือเป็นปัจจัยสำคัญสำหรับท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย ซึ่งเมื่อเคลื่อนที่ขึ้นแล้วท่ามือไม่หันเข้าหน้ากล้องจึงทำให้นิ้วโป้งและนิ้วชี้ใกล้กัน ระบบจึงตีความเป็นท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง
- ค่าความผิดพลาดที่เกิดจากการเคลื่อนที่ลงที่จัดว่าเป็นการเคลื่อนที่ที่ให้ค่าความแม่นยำมากที่สุด อาทิ ในท่าชูนิ้วชี้และนิ้วกลาง เกิดจากการที่ระบบตรวจจับความยาวของนิ้วนางที่โผล่ขึ้นมาในบางเฟรม จึงตีความเป็นท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง
- ค่าความผิดพลาดของการเคลื่อนมือไปทางขวาหรือซ้าย เกิดจากการเคลื่อนที่ท่ามือที่เร็วเกินไปส่งผลต่อความแม่นยำของระบบ และการเปลี่ยนของสภาพแสงที่ทำให้บริเวณนิ้วมือนหายไปในขั้นตอนการตรวจจับสีผิวโดยเฉพาะนิ้วก้อย เพราะเป็นนิ้วที่บางที่สุดและสั้นที่สุด ดังนั้นจึงส่งผลให้ระบบจำแนกท่ามือผิด โดยท่าที่มีค่าความแม่นยำน้อย คือท่าโบกมือ ซึ่งเป็นท่าที่มีความซับซ้อนมากที่สุด จากความผิดพลาดทำให้ท่ามือนี้กลับไปแสดงความหมายว่าช่วยด้วย และต้องใช้เวลาในการเคลื่อนที่เพื่อให้กลับมาเป็นความหมาย ปฏิเสธ อีกครั้ง นอกจากท่าโบกมือแล้ว การเคลื่อนที่ทางขวาหรือซ้ายยังส่งผลต่อท่าชูนิ้วโป้ง เนื่องจากท่ามือไม่หันหน้าเข้ากล้อง จึงพบว่านิ้วมีความยาวมากขึ้นจากมุมของมือที่เปลี่ยนไปขณะทำการทดลอง ส่งผลต่อขั้นตอนการลบพื้นหลังที่ให้ค่าฉากหน้ามีค่าเท่ากับ 1 หรือพิกเซลสีขาวมากขึ้น ระบบจึงตรวจจับเป็นท่าชูนิ้วชี้ขึ้นนั่นเอง

### 5.2.3 ท่ามือเคลื่อนไหวในองศาที่ต่างกัน

การทดลองท่ามือเคลื่อนไหวในองศาที่ต่างกันบนพื้นหลังแบบเรียบ แบ่งออกเป็น 7 องศา โดยกำหนดให้องศาเมื่อท่ามืออยู่ตรงกลางเป็นองศาอ้างอิงมีค่าเท่ากับ 0 องศา เมื่อท่ามือเอียงทางขวาให้มีค่าเป็นบวก (+) ได้แก่ 20 45 และ 65 องศา และเมื่อท่ามือเอียงทางซ้ายให้มีค่าเป็นลบ (-) ได้แก่ -20 -45 และ -65 องศา โดยทดสอบกับท่ามือทั้งหมด 8 ท่ามือ ได้แก่ ท่ากำมือ คือ “หยุด” ท่าชูนิ้วชี้ คือ “ยา” ท่าชูนิ้วชี้และนิ้วกลาง คือ “น้ำ” ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง คือ “หิว” ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย คือ “ห้องน้ำ” ท่าแบมือ คือ “ช่วยด้วย” ท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย คือ “รักคุณ” และท่าชูนิ้วโป้ง คือ “ดี” กำหนดการทดลองให้ท่ามือละ 100 เฟรม

จากการทดลองพบว่าระบบสามารถจำแนกท่ามือได้กับองศาต่าง ๆ อย่างหลากหลายในระดับดีมาก โดยองศาที่ -45 -20 0 20 และ 45 มีค่าความแม่นยำเท่ากับ 100 เปอร์เซ็นต์ในทุกท่ามือ ภาพผลลัพธ์การทดสอบความแม่นยำในการเคลื่อนที่ในองศาที่ต่างกันของอัลกอริทึมบนพื้นหลังแบบเรียบแสดงดังรูปที่ 4.8 ทั้งนี้ค่าความผิดพลาดในการจำแนกท่ามือที่องศา -65 และ 65 เกิดขึ้นกับ 4 ท่ามือที่มีจำนวนนิ้วที่ชูขึ้นมาก ได้แก่ ท่าชูนิ้วชี้และนิ้วกลาง และท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อยที่องศา -65 ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนางที่องศา 65 และท่าแบมือที่องศา -65 และ 65

โดยที่ความผิดพลาดเกิดจากระบบจำแนกระยะห่างความยาวจากจุดกลางฝ่ามือไปยังจุดคอนเวกซ์ฮัลลบริเวณข้อมือดังแสดงในรูปที่ 4.9 ว่าเป็นนิ้วมือด้วย ดังนั้นระบบจึงตีความจำนวนนิ้วเกินมา 1 จำนวน ซึ่งปัญหานี้สามารถแก้ไขได้โดยการแสดงผิวบริเวณแขนให้ยาวขึ้น ซึ่งเมื่อระยะบริเวณแขนยาวมากกว่าค่าระยะห่างที่กำหนด ระบบจะไม่พิจารณาจุดบริเวณข้อมือว่าเป็นจุดปลายนิ้วมืออีกต่อไป ทั้งนี้จะต้องระวังไม่ให้ความยาวของช่องแขนมากเกินไปจนไปถึงช่วงข้อศอก โดยระยะห่างที่เหมาะสมจากข้อมือลงมาที่ช่วงแขนอยู่ที่ 10 เซนติเมตร

#### 5.2.4 ท่ามือเคลื่อนไหวในมุมมองที่ต่างกัน

มุมมองในการแสดงท่ามือจัดเป็นปัญหาที่สำคัญสำหรับวิธีการรับรู้ภาพคอมพิวเตอร์ (Computer vision-based method) ดังนั้นผู้พัฒนาจึงทำการทดสอบการเคลื่อนที่ในมุมมองที่ต่างกันของท่ามือ กำหนดมือขวาเป็นข้างที่ใช้แสดงท่ามือ เมื่อท่ามืออยู่ตรงกลางให้เป็นองศาอ้างอิงมีค่าเท่ากับ 0 องศา เมื่อท่ามือหมุนทวนเข็มนาฬิกาให้ค่าเป็นบวก (-) ได้แก่ -20 -45 และ -65 องศา และเมื่อท่ามือหมุนตามเข็มนาฬิกาให้ค่าเป็นลบ (+) ได้แก่ 20 องศา โดยทดสอบกับท่ามือทั้งหมด 8 ท่ามือ ได้แก่ ท่ากำมือ คือ “หยุด” ท่าชูนิ้วชี้ คือ “ยา” ท่าชูนิ้วชี้และนิ้วกลาง คือ “น้ำ” ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง คือ “หิว” ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย คือ “หึ่งน้ำ” ท่าแบมือ คือ “ช่วยด้วย” ท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย คือ “รักคุณ” และท่าชูนิ้วโป้ง คือ “ดี” กำหนดการทดลองให้ท่ามือละ 100 เฟรม

จากการทดลองพบว่าการหมุนมือในแต่ละท่ามือนั้นจะต้องให้ระบบเห็นจุดปลายนิ้วมือของท่ามือนั้น ๆ ให้ครบทุกนิ้ว โดยเฉพาะเมื่อมีการหมุนมือในองศาที่มากขึ้น อาทิ องศาที่ -45 และ -65 เนื่องจากวิธีการคอนเวกซ์ฮัลลจะพิจารณาบริเวณจุดปลายนิ้วมือ จึงต้องแสดงจุดปลายนิ้วมือให้ชัดเจน จากผลการทดลองแสดงให้เห็นว่าองศาที่ให้ค่าความผิดพลาด ได้แก่ องศาที่ 20 -45 และ -65 โดยท่ามือที่ระบบจำแนกความหมายผิด ได้แก่ ท่าชูนิ้วชี้และนิ้วกลางที่ -45 องศา ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนางที่ -65 องศา ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย และท่าแบมือที่ 20 องศา โดยท่าแบมือที่ยังคงเป็นท่าที่มีค่าความแม่นยำน้อยที่สุด

ทั้งนี้สังเกตได้ว่าที่ 20 องศาจะมีค่าความผิดพลาดเฉพาะกับท่าที่มีจำนวนนิ้วมือมากเท่านั้น ทั้งนี้เนื่องจากการทดลองด้วยมือข้างขวา ทำให้การหมุนมือในทิศทางตามเข็มนาฬิกาทำได้ยาก ต่างกับการหมุนมือในทิศทางทวนเข็มนาฬิกาที่ทำได้ง่ายกว่ามาก ดังนั้นในบางเฟรมภาพระบบจึงไม่สามารถตรวจจับนิ้วก้อยที่มีขนาดเล็กและอยู่ไกลที่สุดได้ในองศาที่ 20 แต่ระบบสามารถตรวจจับนิ้วก้อยได้อย่างชัดเจนในองศาที่ 65 เนื่องจากนิ้วก้อยหันเข้าหากล้องนั่นเอง

### 5.2.5 ท่ามือเคลื่อนไหวในระยะที่ต่างกัน

การทดสอบการเคลื่อนไหวในระยะที่ต่างกันของท่ามือ ทำการแบ่งระยะห่างระหว่างกล้องและท่ามือออกเป็น 2 ช่วง ได้แก่ ช่วงระยะ 0.5 - 1 เมตรและระยะ 1 - 1.5 เมตร จากการทดลองพบว่าที่ระยะห่าง 0.5 - 1 เมตรนั้นระบบสามารถจำแนกท่ามือทุกท่ามือได้ในระดับดีมาก มีค่าความแม่นยำเท่ากับ 100 เปอร์เซ็นต์ สำหรับระยะห่าง 1 - 1.5 เมตร นั้นระบบมีค่าความผิดพลาดตั้งแต่ท่าชูนิ้วชี้และนิ้วกลางไปจนถึงท่าชูนิ้วโป้ง ซึ่งโดยส่วนมากค่าความผิดพลาดจะเกิดที่ระยะห่างประมาณ 1.5 เมตร ทั้งนี้เกิดจาก 2 ปัจจัยหลัก ได้แก่ ค่าความละเอียดของกล้อง และสภาพแสง ซึ่งปัญหาสภาพแสงจะส่งผลกระทบต่อตรวจจับสีผิวทำให้ระบบไม่สามารถตรวจจับนิ้วมือบางนิ้วได้เช่นเดียวกับผลลัพธ์ที่ได้จากการทดลององศาที่แตกต่างกัน

สำหรับความละเอียดของกล้อง หากเพิ่มค่าความละเอียดของกล้องจะทำให้ได้พิกเซลของภาพมากขึ้น ระบบจะสามารถตรวจจับและจำแนกท่ามือในระยะไกลได้อย่างถูกต้อง โดยถ้าตั้งความละเอียดเท่ากับ  $1024 \times 768$  ระบบจะสามารถจำแนกท่ามือได้ในระยะห่างสูงสุด 2 เมตร อย่างไรก็ตามการเพิ่มความละเอียดให้กล้องจะทำให้การทำงานของบอร์ดช้าลง ดังนั้นผู้พัฒนาจึงเลือกใช้ความละเอียดที่  $800 \times 608$  ซึ่งเป็นค่าที่เหมาะสมกับการรับภาพของกล้อง Pi เพื่อให้ระบบทำงานได้รวดเร็ว เหมาะกับการประมวลผลแบบ Real-time โดยสรุปแล้วค่าความแม่นยำโดยรวมของระบบในระยะห่าง 0.5-1.5 เมตร มีค่าในระดับดี เท่ากับ 98 เปอร์เซ็นต์

### 5.2.6 การส่งข้อความแจ้งเตือนบนสมาร์ตโฟน

การทดลองการส่งข้อความไปยังสมาร์ตโฟนผ่านทางแอปพลิเคชันไลน์ทำการทดลองกับท่ามือทั้งหมด 9 ท่ามือ ได้แก่ ท่ากำมือ (หยุด) ท่าชูนิ้วชี้ (ยา) ท่าชูนิ้วชี้และนิ้วกลาง (น้ำ) ท่าชูนิ้วชี้ นิ้วกลางและนิ้วนาง (หิว) ท่าชูนิ้วชี้ นิ้วกลาง นิ้วนางและนิ้วก้อย (ห้องน้ำ) ท่าแบมือ (ช่วยด้วย) ท่าชูนิ้วโป้ง นิ้วชี้และนิ้วก้อย (รักคุณ) ท่าชูนิ้วโป้ง (ดี) และท่าโบกมือ (ปฏิเสธ) ในแต่ละเงื่อนไข ซึ่งการทำงานของระบบคือ ระบบทำการส่งข้อความแจ้งเตือนไปยังสมาร์ตโฟนเมื่อระบบเสร็จสิ้นขั้นตอนการจำแนกท่ามือและแสดงข้อความบนหน้าจอบนอินเทอร์เน็ตเฟสแล้วเท่านั้น ฉะนั้นค่าความถูกต้องของระบบ

จะเป็นไปตามค่าความถูกต้องของการจำแนกในแต่ละเงื่อนไขนั่นเอง ดังนั้นจากการทดลองพบว่าระบบสามารถส่งข้อความแจ้งเตือนได้ถูกต้องตรงกับความหมายท่ามือที่ผ่านการจำแนกแล้วในระดับดีมากเท่ากับ 100 เปอร์เซนต์

### 5.2.7 การเปรียบเทียบค่าความแม่นยำของการจำแนกท่ามือ

การเปรียบเทียบค่าความแม่นยำในการจำแนกท่ามือแบ่งออกเป็น 3 ประเภท ได้แก่ 1) การเปรียบเทียบค่าความแม่นยำในการจำแนกท่ามือขณะอยู่นิ่งบนพื้นหลังแบบเรียบ 2) การเปรียบเทียบค่าความแม่นยำในการจำแนกท่ามือขณะอยู่นิ่งบนพื้นหลังแบบซับซ้อน และ 3) การเปรียบเทียบค่าความแม่นยำในการจำแนกท่ามือเคลื่อนไหวบนพื้นหลังแบบซับซ้อน ดังนี้

จากการเปรียบเทียบค่าความแม่นยำในการจำแนกท่ามือขณะอยู่นิ่งบนพื้นหลังแบบเรียบของผลการทดลองที่ได้จากอัลกอริธึมที่นำเสนอในหัวข้อ 3.2.3 กับอัลกอริธึมในหัวข้อ 3.2.2 และอัลกอริธึมที่เสนอโดย [30] ที่ใช้การค้นหาเส้นรอบขอบและ HOG โดยใช้ชั้นการเรียนรู้ 27 ชั้น พบว่าผลลัพธ์ของอัลกอริธึมที่นำเสนอมีค่าเท่ากับ 100 เปอร์เซนต์ ซึ่งมีความแม่นยำสูงกว่าอัลกอริธึมในหัวข้อ 3.2.2 ที่มีค่าเท่ากับ 80.86 เปอร์เซนต์ และอัลกอริธึม [30] ที่มีค่าเท่ากับ 99.8 เปอร์เซนต์ ดังนั้นอัลกอริธึมที่เสนอมีประสิทธิภาพในการจำแนกท่ามือมากกว่าเมื่อท่ามืออยู่นิ่งบนพื้นหลังแบบเรียบ

### 5.3 ข้อเสนอแนะ

งานวิจัยชิ้นนี้สามารถพัฒนาเพิ่มเติมเพื่อให้เกิดประสิทธิภาพสูงสุดในการนำไปใช้งานจริงได้อาทิ การปรับปรุงอัลกอริธึมให้สามารถทำงานได้บนพื้นหลังซับซ้อนแม้ในสภาพแสงน้อย การสั่งการให้เครื่องพักการทำงานเพื่อประหยัดพลังงาน การเพิ่มท่ามือตามความต้องการของผู้ใช้งาน การแยกความแตกต่างระหว่างฝ่ามือและข้อศอกเพื่อป้องกันไม่ให้ระบบตรวจจับผิดพลาด การออกแบบแพ็คเกจเพื่อใช้ติดตั้งในสถานที่จริง เป็นต้น

## รายการอ้างอิง

- [1] C. Acharya, H. Thimbleby, and P. Oladimeji, "Human computer interaction and medical devices," in *Proceedings of the 24th BCS Interaction Specialist Group Conference*, 2010, pp. 168-176: British Computer Society.
- [2] J. M. Ortman, V. A. Velkoff, and H. Hogan, "An aging nation: the older population in the United States," *Washington, DC: US Census Bureau*, pp. 25-1140, 2014.
- [3] A. Chaudhary and J. L. Raheja, "A health monitoring system for elder and sick persons," *arXiv preprint arXiv:1304.4652*, 2013.
- [4] S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer interaction: a survey," *Artificial Intelligence Review*, vol. 43, no. 1, pp. 1-54, 2015.
- [5] A. Nishikawa *et al.*, "FAce MOUSe: A novel human-machine interface for controlling the position of a laparoscope," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 5, pp. 825-841, 2003.
- [6] P. N. Puongwarin. (2016, Mar. 15). *Parkinson's disease with the elderly*. Available: <http://www.si.mahidol.ac.th/sidoctor/e-pl/articledetail.asp?id=112>
- [7] S. Sawasdee and S. Pumrin, "Elderly care notification system using hand posture recognition," in *Digital Information and Communication Technology and it's Applications (DICTAP), 2014 Fourth International Conference on*, 2014, pp. 304-309: IEEE.
- [8] E. Ohn-Bar and M. M. Trivedi, "Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations," *IEEE transactions on intelligent transportation systems*, vol. 15, no. 6, pp. 2368-2377, 2014.
- [9] K. Liu and N. Kehtarnavaz, "Real-time robust vision-based hand gesture recognition using stereo images," *Journal of Real-Time Image Processing*, vol. 11, no. 1, pp. 201-209, 2016.



- [10] C. Shan, T. Tan, and Y. Wei, "Real-time hand tracking using a mean shift embedded particle filter," *Pattern recognition*, vol. 40, no. 7, pp. 1958-1970, 2007.
- [11] C.-C. Hsieh and D.-H. Liou, "Novel Haar features for real-time hand gesture recognition using SVM," *Journal of Real-Time Image Processing*, vol. 10, no. 2, pp. 357-370, 2015.
- [12] H.-I. Suk, B.-K. Sin, and S.-W. Lee, "Hand gesture recognition based on dynamic Bayesian network framework," *Pattern Recognition*, vol. 43, no. 9, pp. 3059-3072, 2010.
- [13] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital Image Processing Using MATLAB®*. McGraw Hill Education, 2010.
- [14] A. K. Jain, *Fundamentals of digital image processing*. Prentice-Hall, Inc., 1989, p. 569.
- [15] M. Sonka, V. Hlavac, and R. Boyle, *Image processing, analysis, and machine vision*. Cengage Learning, 2014.
- [16] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137-154, 2004.
- [17] M. Piccardi, "Background subtraction techniques: a review," in *Systems, man and cybernetics, 2004 IEEE international conference on*, 2004, vol. 4, pp. 3099-3104: IEEE.
- [18] K. Yadav and J. Bhattacharya, "Real-Time Hand Gesture Detection and Recognition for Human Computer Interaction," in *Intelligent Systems Technologies and Applications*: Springer, 2016, pp. 559-567.
- [19] H.-S. Yeo, B.-G. Lee, and H. Lim, "Hand tracking and gesture recognition system for human-computer interaction using low-cost hardware," *Multimedia Tools and Applications*, vol. 74, no. 8, pp. 2687-2715, 2015.
- [20] B. K. Horn and B. G. Schunck, "Determining optical flow," *Artificial intelligence*, vol. 17, no. 1-3, pp. 185-203, 1981.
- [21] S. Lu, G. Tsechpenakis, D. N. Metaxas, M. L. Jensen, and J. Kruse, "Blob analysis of the head and hands: A method for deception detection," in

- System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on*, 2005, pp. 20c-20c: IEEE.
- [22] M. T. Goodrich and R. Tamassia, *Algorithm design: foundation, analysis and internet examples*. John Wiley & Sons, 2006.
- [23] M. M. Youssef, "Hull convexity defect features for human action recognition," University of Dayton, 2011.
- [24] M. M. Deza and E. Deza, "Encyclopedia of distances," in *Encyclopedia of Distances*: Springer, 2009, pp. 1-583.
- [25] A. M. Andrew, "An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods by Nello Christianini and John Shawe-Taylor," Cambridge University Press, Cambridge, 2000, xiii+ 189 pp., ISBN 0-521-78019-5 (Hbk, £ 27.50)," ed: Cambridge Univ Press, 2000.
- [26] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346-359, 2008.
- [27] (1 May). *RASPBERRY PI 3 MODEL B*. Available: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
- [28] (2017, 1 May). *Mean, Median, Mode, and Range*. Available: <http://www.purplemath.com/modules/meanmode.htm>
- [29] H. Dmitriv and R. Kaminsky, "Two Algorithms Median Filtering to Identify the Time Series Trend," in *Advances in Intelligent Systems and Computing*: Springer, 2017, pp. 283-292.
- [30] S. Sawasdee, "Elderly Care Notification System Using Hand Posture Recognition," Master of Engineering Program in Electrical Engineering, Department of Electrical Engineering, Chulalongkorn University, Chulalongkorn University, 2014.



ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

## ประวัติผู้เขียนวิทยานิพนธ์

ฐิตาภรณ์ กนกรัตนา เกิดวันที่ 5 กรกฎาคม พ.ศ. 2535 จบการศึกษาระดับปริญญาตรี จากภาควิชาเทคโนโลยีมีเดีย (วท.บ.) มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ด้วยเกียรตินิยม อันดับ 1 เหรียญทอง ปัจจุบันศึกษาระดับปริญญาโท ภาควิชาวิศวกรรมศาสตร์ คณะวิศวกรรมศาสตร์ (วศ.ม.) จุฬาลงกรณ์มหาวิทยาลัย เคยรับรางวัลจากการนำเสนอผลงานวิจัยในการประชุมวิชาการระดับนานาชาติ Best Paper Award 3 ครั้ง ได้แก่ การประชุมวิชาการระดับนานาชาติ the Eleventh International Symposium on Natural Language Processing (SNLP2016) การประชุมวิชาการระดับนานาชาติ the 7th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2013) และการประชุมวิชาการระดับนานาชาติ PSNR (ICACME' 2012) และรับรางวัลรองชนะเลิศอันดับ 1 จากการแข่งขัน 3 minute thesis โดยจุฬาลงกรณ์มหาวิทยาลัยร่วมกับมหาวิทยาลัยควีนส์แลนด์ (University of Queensland) งานวิจัยที่สนใจ ได้แก่ การประมวลผลภาพและคอมพิวเตอร์วิทัศน์

บทความทางวิชาการที่ได้รับการเผยแพร่

1. T. Ganokratanaa and S. Pumrin, "The vision-based hand gesture recognition using blob analysis," in Digital Arts, Media and Technology (ICDAMT), International Conference on, 2017, pp. 336-341: IEEE.
2. T. Ganokratanaa and S. Pumrin, "The Algorithm of Static Hand Gesture Recognition with Knowledge Based for the Elderly," Advances in Intelligent Systems and Computing, 2016.

วารสารวิชาการระดับนานาชาติที่ได้รับการยอมรับให้ตีพิมพ์

1. T. Ganokratanaa and S. Pumrin, "The Elderly People Dynamic Hand Gesture Recognition Algorithm Using Optical Flow and SURF," Chiang Mai University Journal of Natural Sciences, 2017.
2. T. Ganokratanaa and S. Pumrin, "Hand Gesture Recognition Algorithm for Smart Cities based on Wireless Sensor," International Journal of Online Engineering, 2017.