แอพพลิเคชั่นสำหรับแปลภาษามือบนระบบปฏิบัติการแอนดรอยด์

SIGN LANGUAGE TRANSLATION APPLICATION ON ANDROID SYSTEM

นิสา คูส่ง
ภูริจิต วงศ์วิเศษกิจ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2557

แอพพลิเคชั่นสำหรับแปลภาษามือบนระบบปฏิบัติการแอนดรอยด์

SIGN LANGUAGE TRANSLATION APPLICATION ON ANDROID SYSTEM

นิสา คูส่ง

ภูริจิต วงศ์วิเศษกิจ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2557

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2557

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง แอพพลิเคชั่นสำหรับแปลภาษามือบนระบบปฏิบัติการแอนดรอยด์

SIGN LANGUAGE TRANSLATION APPLICATION ON ANDROID SYSTEM

ผู้จัดทำ

1. นางสาวนิสา คูส่ง รหัสนักศึกษา 54010721
2. นายภูริจิต วงศ์วิเศษกิจ รหัสนักศึกษา 54011007

อรฉัตร จิตต์โสภักตร์ อาจารย์ที่ปรึกษา

(รศ.ดร.อรฉัตร จิตต์โสภักตร์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอพพลิเคชั่นสำหรับแปลภาษามือ บนระบบปฏิบัติการแอนดรอยด์

นางสาวนิสา คูส่ง 54010721
นายภูริจิต วงศ์วิเศษกิจ 54011007
รศ.ดร. อรฉัตร จิตต์โสภักตร์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2557

บทคัดย่อ

การพัฒนาแอพพลิเคชั่นแปลภาษามือบนระบบปฏิบัติการแอนดรอยด์ มีวัตถุประสงค์เพื่อแก้ไขปัญหาการสื่อสารระหว่างผู้บกพร่องทางการได้ยินและบุคคลทั่วไป โดยพัฒนาบนระบบปฏิบัติการแอนดรอยด์เพื่อให้สะดวกแก่การพกพา และสามารถใช้งานได้ง่าย การแปลภาษามือจะใช้หลักการประมวลผลภาพในการแปล และใช้ภาษาจาวาในการพัฒนา โดยมี OpenCV Library เป็นส่วนเสริมในการช่วยการประมวลผลภาพ

ผลจากการพัฒนาแอพพลิเคชั่นสำหรับแปลภาษามือบนระบบปฏิบัติการแอนดรอยด์ สามารถตรวจจับมือ นิ้วมือได้ในระดับที่พอใจ และสามารถเพิ่มคำหรือ code ลงในฐานข้อมูลได้ และสามารถหาคำที่ต้องการ โดยแอพพลิเคชั่นจะมีคุณสมบัติในการแสดงคำแนะนำเพื่อเป็นตัวเลือกให้กับผู้ใช้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sign Language Translation Application on Android System

Ms. Nisa	Khusong	54010721
Mr. Purijit	Vongviseskij	54011007
Assoc.Prof.Dr. Orachat	Chitsobhuk	Advisor

Academic Year 2015

ABSTRACT

In this project, 'Sign Language Translation Application on Android System', it has been prepared with the objective to resolve the problem of communication between deaf people and normal people. It was created on Android system to make the application portable and convenient. This project uses Java Language as a main programming language and OpenCV library to perform image processing task.

The result of Sign Language Translation Application on Android System demonstrates good performance in detecting hand from complexity background, including fingertips position, naming finger, and tracking hand movement. For sign language translator, it can translate sign language to text and voice by using string matching algorithm to find the nearest word to make the translator part work effectively. The ThSL database can add and remove words and also can import and export database to another devices. Additionally, the system can provide alternative words for effective translation.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์เรื่อง แอพพลิเคชั่นแปลภาษามือบนระบบปฏิบัติการแอนดรอยด์ สามารถดำเนินงานได้เนื่องจากได้รับการสนับสนุนและช่วยเหลือด้านต่างๆ จากบุคคลต่อไปนี้

ขอขอบพระคุณ รศ.ดร. อรฉัตร จิตต์โสภักตร์ อาจารย์ที่ปรึกษาโครงงาน ที่คอยให้ความช่วยเหลือ ทั้งให้คำปรึกษา ชี้แนะเวลาเกิดปัญหาระหว่างการดำเนินโครงงาน รวมถึงการเสนอความคิดเห็นอันมีประโยชน์ในการพัฒนาโปรแกรมตั้งแต่เริ่มทำโครงงานจนจบ

ขอขอบพระคุณ คณาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่าน ที่ได้ให้ความรู้ ให้คำแนะนำและกำลังใจตลอดการศึกษาที่ผ่านมา

ขอขอบคุณทางสมาคมคนหูหนวกแห่งประเทศไทย และเพจภาษามือไทย Thai Sign Language ที่ได้ให้คำปรึกษา และให้ความร่วมมือในการให้ข้อมูลเกี่ยวกับภาษามือ

ขอขอบพระคุณ คุณพ่อ คุณแม่อันเป็นที่รักยิ่ง สำหรับความรักความห่วงใย ความดูแลเอาใจใส่และให้การสนับสนุนเสมอมา

ขอขอบคุณเพื่อนๆ ที่คอยช่วยเหลือ ผลักดันและให้กำลังใจในการทำโครงงานให้สำเร็จไปได้ด้วยดี

นิสา คูส่ง

ภูริจิต วงศ์วิเศษกิจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....VI

สารบัญรูป.....VII

บทที่ 1 บทนำ.....1

1.1 ความเป็นมาของปัญหา.....1

1.2 วัตถุประสงค์ของการศึกษา.....2

1.3 ขอบเขตของโครงงาน.....2

1.4 ประโยชน์ที่คาดว่าจะได้รับ.....2

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....3

2.1 ภาษามือ.....3

2.2 ภาษาจาวา.....4

2.3 ระบบปฏิบัติการแอนดรอยด์.....5

2.4 OpenCV.....6

2.5 Google Voice Typing.....6

2.6 Text to Speech Synthesis (SVOX).....6

2.7 การตรวจจับใบหน้า.....7

2.8 Haar-Like Features.....8

2.9 Morphology (Opening and Closing).....8

2.10 เทคนิคในการทำ Fingertips Tracking.....9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 3 การออกแบบและพัฒนาระบบ 11

3.1 ภาพรวมของโครงสร้างระบบ 11

3.2 Camera module 12

3.3 Detection module 15

3.4 User interface module 23

บทที่ 4 การทดลอง และผลการทดลอง 26

4.1 การทดลองแอพพลิเคชั่นส่วนจัดการบทสนทนา 26

บทที่ 5 บทสรุปและข้อเสนอแนะ 36

5.1 บทสรุป 36

5.2 ปัญหาและอุปสรรคต่างๆ 36

5.3 แนวทางในการพัฒนาต่อ 37

บรรณานุกรม 38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง หน้า

4.1 ความถูกต้องของการแปลงคำศัพท์จากเสียงเป็นข้อความ 27

4.2 ความถูกต้องของการแปลงประโยคจากเสียงเป็นข้อความ 28

4.3 ความถูกต้องของการเจอจุดปลายนิ้ว (ครั้งที่ 1) 30

4.4 ความถูกต้องของการเจอจุดปลายนิ้ว (ครั้งที่ 2) 30

4.5 สรุปค่าเฉลี่ยของความถูกต้องจากผลการทดลองของตาราง 4.2 และ 4.3 30

4.6 ความถูกต้องของการเจอจุดปลายนิ้วโดยมีปัจจัยคือแสงสว่าง (ครั้งที่ 1) 32

4.7 ความถูกต้องของการเจอจุดปลายนิ้วโดยมีปัจจัยคือแสงสว่าง (ครั้งที่ 2) 32

4.8 สรุปค่าเฉลี่ยของความถูกต้อง จากการทดลองของตารางที่ 4.6 และ 4.7 32

4.9 ความถูกต้องของการแปลภาษามือ (ตัวเลข) 33

4.10 ความถูกต้องของการแปลภาษามือ (ตัวอักษร) 34

4.11 ความถูกต้องของการแปลภาษามือ (คำศัพท์) 34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป		หน้า
2.1	แบบสะกดนิ้วมือไทย	4
2.2	การทำ opening และ closing	8
2.3	การทำ contour extraction	9
2.4	การทำ maximum inscribed circle and radius	9
2.5	convex hull และ convexity defects	10
2.6	convex hull และ convexity defects ของ contour ของรูปมือ	10
2.7	ผลจากการหาตำแหน่งของจุดปลายนิ้ว	10
3.1	ภาพรวมของโครงสร้างระบบ	11
3.2	การทำงานของ Camera Module	13
3.3	การทำงานของส่วน Calibration	13
3.4	การรวมกันของภาพ binary มือและพื้นหลัง	14
3.5	การทำงานของส่วน thresholding and binarization	15
3.6	การทำงานในส่วนของ Detection module	16
3.7	การหา min enclosing circle เพื่อนำมาใช้ในการตั้งค่า ROI	17
3.8	การหาจุด convex hull (จุดสีเขียว)	18
3.9	การหาจุด convexity defect	18
3.10	ภาพหลังจากการผ่านอัลกอริทึมเพื่อหาจุด defects บริเวณร่องนิ้ว	19
3.11	การหาจุดปลายนิ้ว	19
3.12	ภาพเมื่อผ่านอัลกอริทึมในการหาจุดปลายนิ้วมือ	20
3.13	การหา Bounding rectangle	21
3.14	การหาจุด centroid และพื้นที่วงกลมของฝ่ามือ	21
3.15	การทำงานของ Hand gesture module	22
3.16	การทำงานของส่วนเพิ่มและลบคำศัพท์	22
3.17	การทำงานส่วน App chat module	24
3.18	Chat display program	24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป		หน้า
4.1	ส่วนจัดการบทสนทนา	26
4.2	รูปสำหรับระบุฝั่งของบทสนทนา	26
4.3	ขนาดของมือที่ระยะต่างๆ	29
4.4	พื้นหลังในสภาวะมีแสงสว่างมาก	31
4.5	พื้นหลังในสภาวะมีแสงสว่างปานกลาง	31
4.6	พื้นหลังในสภาวะมีแสงสว่างน้อย	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของปัญหา

จำนวนผู้บกพร่องทางการได้ยินในประเทศไทย ในปีพุทธศักราช 2553 นั้น มีจำนวนประชากรประมาณ 700,000 คน ซึ่งถือว่าเป็นจำนวนที่ไม่มากหากเทียบกับจำนวนประชากรทั้งประเทศ โดยผู้บกพร่องทางการได้ยินจะใช้ภาษามือเป็นภาษาหลักในการสื่อสาร ในแต่ละประเทศได้มีการพัฒนาภาษามือที่แตกต่างกันออกไป โดยในประเทศไทยผู้มีความบกพร่องทางการได้ยินจะใช้ภาษามือไทยภายในกลุ่ม และใช้วิธีการเขียนกระดาษ หรือพิมพ์ข้อความในการสื่อสารกับบุคคลอื่นๆ

ปัญหาหนึ่งที่ผู้บกพร่องทางการได้ยินได้พบเจออยู่เป็นประจำ คือ การสื่อสารกับบุคคลทั่วไปซึ่งไม่สามารถเข้าใจภาษามือที่สื่อสารออกไปได้ นอกจากนี้ ผู้ที่มีความบกพร่องทางการได้ยินส่วนมาก มักจะบกพร่องทางการได้ยินตั้งแต่กำเนิด ทำให้การเรียนรู้ต่างๆ ทำได้ยากขึ้น ทั้งเรื่องสื่อการเรียนรู้ ที่ถูกจำกัดอยู่ในวงที่แคบลงและการเรียนรู้ร่วมกับบุคคลปกติที่ทำได้ยาก เนื่องจากไม่สามารถได้ยินเสียงของผู้บรรยายได้ ทำให้ผู้บกพร่องทางการได้ยินบางกลุ่มไม่สามารถเข้าใจประโยคหรือข้อความที่อยู่ในกระดาษได้ อีกทั้งไม่สามารถสื่อสารกับผู้อื่นด้วยการเขียนเพราะไวยากรณ์หรือคำของประโยคของภาษามือไม่ตรงกันกับภาษาไทยที่ใช้กันโดยทั่วไป (จากสถิติของรายงานสถานการณ์คนพิการในสังคมไทยปี 2550 พบว่า ผู้บกพร่องทางการได้ยิน มากกว่า 90 เปอร์เซ็นต์ ไม่ได้รับการศึกษา เนื่องจากไม่สามารถเรียนร่วมกับคนอื่นได้)

อีกปัญหาหนึ่งที่ยังเกิดขึ้นในปัจจุบันของผู้บกพร่องทางการได้ยิน คือ ล่ามภาษามือมีจำนวนขาดแคลน โดยในปี พุทธศักราช 2555 มีล่ามภาษามือทั้งหมด 500 คน ซึ่งหากเทียบเป็นจำนวนอัตราส่วนกันกับจำนวนผู้บกพร่องทางการได้ยินทั้งหมดแล้วถือว่ามีจำนวนน้อย ทำให้ค่าใช้จ่ายในการจ้างล่ามภาษามือมีราคาสูง หากเกิดกรณีที่ผู้บกพร่องทางการได้ยินจำเป็นต้องใช้ล่ามภาษามือ เช่น เกิดคดีความ ไปหาหมอ หรือติดต่อหน่วยงานต่างๆ จะต้องเสียค่าใช้จ่ายเป็นจำนวนมาก

การพัฒนาแอพพลิเคชั่นสำหรับแปลภาษามือบนระบบปฏิบัติการแอนดรอยด์ จะทำให้บุคคลทั่วไปสามารถเข้าใจสิ่งที่ผู้ที่มีความบกพร่องทางการได้ยินต้องการจะสื่อสารได้ และเนื่องจากแอพพลิเคชั่นถูกพัฒนาบนระบบปฏิบัติการแอนดรอยด์ซึ่งมีผู้ใช้ค่อนข้างแพร่หลาย จึงทำให้ผู้ที่มีความบกพร่องทางการได้ยินสามารถอยู่ร่วมกับบุคคลอื่นในสังคมได้ง่ายมากขึ้น อีกทั้งสามารถนำแอพพลิเคชั่นนี้ไปต่อยอดกับการใช้งานต่างๆ ได้ เช่น ด้านการศึกษา ด้านการแพทย์ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของการศึกษา

1) เพื่อศึกษาทดลองขั้นตอนวิธีการพัฒนาโปรแกรมสำหรับการประมวลผลภาพ
2) เพื่อศึกษาการพัฒนาโปรแกรมบนระบบปฏิบัติการแอนดรอยด์ ตั้งแต่เริ่มต้นวางแผนการทำงาน จนสามารถนำโปรแกรมมาใช้งานได้จริง
3) เพื่อพัฒนาระบบแอพพลิเคชั่นแปลภาษามือบนระบบปฏิบัติการแอนดรอยด์ สำหรับนำไปใช้ชีวิตประจำวันได้
4) เพื่อเป็นต้นแบบแอพพลิเคชั่นภาษามือ ในการนำไปพัฒนาต่อยอดในด้านต่างๆ ได้

1.3 ขอบเขตของโครงงาน

1) สามารถทำการวิเคราะห์และประมวลผลภาพจากการทำท่าทางตามภาษามือผ่านกล้องโทรศัพท์ มือถือ และนำมาแปลความหมายเป็นคำหรือประโยคได้
2) ระบบจะสามารถนำคำหรือประโยคที่ถูกแปลความหมายแล้ว นำมาแสดงผลผ่านทางข้อความ และเสียงบนโทรศัพท์มือถือได้
3) ระบบสามารถเพิ่มคำศัพท์ลงในฐานข้อมูลได้
4) ระบบสามารถรับข้อมูลเสียงเพื่อแปลงกลับเป็นข้อความได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1) ได้เรียนรู้ขั้นตอนการพัฒนาโปรแกรมจากการทำงาน
2) ได้แอพพลิเคชั่นแปลภาษามือที่เสร็จสมบูรณ์ และใช้ได้จริง
3) ผู้บกพร่องทางการได้ยินได้รับประโยชน์จากการพัฒนาแอพพลิเคชั่นแปลภาษามือ
4) สามารถนำแอพพลิเคชั่นไปพัฒนาต่อยอดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ภาษามือ

ภาษามือ (Sign Language) เป็นภาษาที่ใช้การสื่อสารด้วยมือ หรือร่างกาย หรือการใช้ริมฝีปากในการสื่อความหมายออกมาแทนการใช้เสียงพูด ภาษามือประกอบไปด้วยการทำท่ามือ, ตำแหน่งของมือ, การเคลื่อนไหวของลำตัว, จังหวะการทำท่ามือ และการแสดงออกของสีหน้า ในการใช้ภาษามือให้ถูกต้องจำเป็นต้องอาศัยองค์ประกอบเหล่านี้มาผสมผสานกัน เพื่อให้สามารถสื่อความหมายออกมาได้ชัดเจน หากองค์ประกอบเปลี่ยนไปอาจเปลี่ยนความหมายของสิ่งที่จะสื่อได้ เช่นเดียวกับการออกเสียงให้ถูกต้องในภาษาพูดที่จำเป็นต้องออกเสียงให้ชัดเจน

ภาษามือในแต่ละประเทศจะแตกต่างกันออกไป แต่ส่วนใหญ่จะได้รับอิทธิพลมาจากที่เดียวกัน คือ ภาษามืออเมริกัน (ASL: American Sign Language) โดยส่วนใหญ่ภาษามือของแต่ละประเทศจะถูกแบ่งออกเป็นภาษามือธรรมชาติ (ภาษามือตามท่าทาง) และภาษามือประดิษฐ์ (ภาษามือโดยการสะกดตัวอักษร) การเลือกใช้แต่ละประเภทจะขึ้นอยู่กับความสามารถและความรู้ด้านภาษามือของคู่สนทนาทั้งคู่ รวมถึงสถานการณ์ของการสนทนา ซึ่งผู้ใช้ภาษามือจะต้องปรับภาษามือที่ใช้ให้เหมาะสมกับเหตุการณ์

2.1.1 หลักสำคัญในการทำท่าภาษามือ

1) ภาษามือเป็นภาษาท่าทางซึ่งมีการเคลื่อนไหวของมือเป็นหลัก
2) ใช้กิริยา อาการ หน้าตาเป็นส่วนประกอบ เพื่อช่วยให้เกิดความเข้าใจ
3) ท่าภาษามือควรเป็นทำง่าย สะดวก มีความหมายใกล้เคียงธรรมชาติ ไม่มีหลายจังหวะ เหมาะสมกับหลักสรีระศาสตร์
4) ท่าภาษามือควรทำอย่างมีจังหวะ มีการเว้นระยะ ไม่ทำรวดเร็วหรือช้าเกินไป อยู่ในรัศมีที่สายตามองเห็นได้
5) การทำท่ามือเดียว หรือสองมือขึ้นอยู่กับระดับสายตาที่มองเห็น
6) การทำท่าภาษามือไม่มีการบังคับว่าต้องเป็นมือซ้ายหรือมือขวา ถ้าทำท่าสองมือ มือที่ถนัดเป็นมือที่เคลื่อนไหวส่วนอีกมือหนึ่งจะเป็นฐาน
7) ในการทำท่าภาษามือ การแสดงสีหน้า และการเคลื่อนไหวส่วนต่างๆ ของใบหน้า เช่น คิ้ว ปาก เป็นสิ่งสำคัญช่วยให้เข้าใจความหมายของภาษามือได้ชัดเจนขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 2.1 แบบสะกดนิ้วมือไทย

2.2 ภาษาจาวา

ภาษาจาวา (Java Language) เป็นภาษาโปรแกรมชั้นสูง (high-level programming language) และเป็นภาษาโปรแกรมเชิงวัตถุ ที่ถูกพัฒนาโดยบริษัท Sun Microsystems โดยมีจุดประสงค์เพื่อให้สามารถใช้งานได้ง่ายขึ้น และกำจัดข้อผิดพลาดต่างๆของโปรแกรมที่มักจะเกิดขึ้นบ่อยๆ รวมถึงสามารถใช้งานได้โดยไม่ขึ้นกับแพลตฟอร์ม ภาษาจาวามีคุณสมบัติดังนี้

1) Object Oriented ในภาษาจาวา ทุกอย่างจะถูกมองเป็นเชิงวัตถุเสมอ และสามารถขยายขอบเขตได้ง่าย เนื่องจากมีพื้นฐานอยู่บน Object model อยู่แล้ว
2) Platform independent ภาษาจาวามีข้อแตกต่างจากภาษาอื่นๆ เช่น ภาษา C หรือ C++ โดยภาษา จาวาจะถูก compile ให้อยู่ในรูปของ byte code และใช้ JVM เป็นตัวแปลความหมาย ทำให้ภาษาจาวาไม่ขึ้นกับแพลตฟอร์มใดๆ
3) Simple ภาษาจาวาถูกออกแบบมาเพื่อให้ง่ายต่อการเรียนรู้ หากผู้ใช้งานมีความรู้ความเข้าใจด้านการเขียนโปรแกรมเชิงวัตถุแล้ว
4) Secure ภาษาจาวามี feature ในการพัฒนาโปรแกรมให้ปราศจากไวรัส รวมถึงเทคนิคการระบุตัวตนโดยใช้การเข้ารหัสแบบ public-key

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ระบบปฏิบัติการแอนดรอยด์

ระบบปฏิบัติการแอนดรอยด์ (Android) เป็นระบบปฏิบัติการที่ได้รับการพัฒนามาจากระบบปฏิบัติการลินุกซ์ โดยเริ่มต้นถูกพัฒนาเพื่อนำมาใช้กับอุปกรณ์ที่ใช้จอสัมผัส เช่น สมาร์ทโฟน แท็ปแล็ต จนกระทั่งได้รับความนิยมและถูกนำไปพัฒนาต่อยอดกับอุปกรณ์อื่นๆ เช่น กล้องดิจิตอล สมาร์ททีวี นาฬิกา เป็นต้น แอนดรอยด์เป็นระบบปฏิบัติการ Open Source ซึ่งอนุญาตให้ผู้ผลิต นักพัฒนาและผู้ให้บริการเครือข่าย สามารถปรับแต่ง และนำออกจำหน่ายได้ ทำให้แอนดรอยด์กลายเป็นระบบปฏิบัติการที่แพร่หลาย มีการพัฒนาโปรแกรมขึ้นอย่างต่อเนื่อง และกลายเป็นหนึ่งในระบบปฏิบัติการที่มีผู้ใช้งานมากที่สุด

2.3.1 ส่วนติดต่อกับผู้ใช้งานของระบบ

ส่วนติดต่อกับผู้ใช้งานของระบบปฏิบัติการแอนดรอยด์ถูกออกแบบอยู่บนพื้นฐานของรูปแบบ direct manipulation ซึ่งจะใช้การสัมผัสที่สอดคล้องกับการกระทำในโลกความจริง เช่นการปัด, การแตะ, การกวาดนิ้ว รวมไปถึงการใช้นิ้วหมุนบนหน้าจอ

ระบบปฏิบัติการแอนดรอยด์จะใช้หน้าจอหลักในการนำทางไปทุกส่วนของอุปกรณ์ โดยส่วนหน้าจอหลักผู้ใช้งานจะสามารถนำไอคอน หรือเครื่องมือต่างๆ มาจัดเรียงได้ การติดตั้งโปรแกรมของระบบปฏิบัติการแอนดรอยด์จะใช้วิธีการติดตั้งผ่าน Google Play ซึ่งเป็นแหล่งรวบรวมโปรแกรมที่ถูกพัฒนาบนแอนดรอยด์ทั้งหมดเข้าไว้ด้วยกัน

ด้านบนของระบบปฏิบัติการแอนดรอยด์ จะเป็นแถบสถานะซึ่งแสดงถึงข้อมูลต่างๆ เกี่ยวกับอุปกรณ์ และ การเชื่อมต่อต่างๆ แถบสถานะสามารถดึงลงมาเพื่อที่จะแสดงการแจ้งเตือนบนหน้าจอเมื่อแอปพลิเคชั่นแจ้งเตือน หรือข้อมูลมีอัพเดท เช่นการได้รับข้อความใหม่ เป็นต้น

2.3.2 การจัดการหน่วยความจำ

เนื่องจากอุปกรณ์ที่ใช้ระบบปฏิบัติการแอนดรอยด์ส่วนใหญ่จะใช้แบตเตอรี่เป็นแหล่งพลังงานทำให้มีพลังงานจำกัด ระบบปฏิบัติการแอนดรอยด์จึงถูกออกแบบมาเพื่อจัดการหน่วยความจำโดยใช้พลังงานน้อยที่สุด โดยโปรแกรมใดที่ไม่ได้ใช้งานจะถูกเก็บไว้ในหน่วยความจำ ซึ่งจะทำให้ไม่มีการใช้พลังงานจากแบตเตอรี่จนกว่าโปรแกรมนั้นจะถูกเปิดขึ้นมาอีกครั้ง วิธีนี้จะทำให้ผู้ใช้งานไม่จำเป็นต้องปิดเปิดโปรแกรมใหม่ตั้งแต่เริ่มต้น และประหยัดพลังงาน

ในกรณีที่อุปกรณ์มีหน่วยความจำเหลือน้อย ระบบปฏิบัติการจะทำการปิดโปรแกรมและโปรเซสที่ไม่มีการทำงาน ซึ่งกระบวนการนี้จะทำให้ผู้ใช้งานไม่ต้องจัดการกับหน่วยความจำด้วยตัวเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 OpenCV

OpenCV (Open Source Computer Vision) เป็น library ที่รวบรวมฟังก์ชันในการเขียนโปรแกรมเกี่ยวกับคอมพิวเตอร์วิทัศน์ (Computer Vision) สำหรับนำไปพัฒนาโปรแกรมที่เกี่ยวข้องกับการประมวลผลภาพ (Image Processing), การเรียนรู้ของเครื่อง (Machine Learning) และความสามารถทางด้านการมองเห็นของคอมพิวเตอร์ (Computer Vision) ซึ่งถูกพัฒนาโดย Intel Russia research Center และเผยแพร่ภายใต้ลิขสิทธิ์ของ BSD สามารถนำไปใช้ในเชิงการศึกษาหรือเชิงธุรกิจได้โดยไม่เสียค่าใช้จ่าย โดยปัจจุบัน OpenCV สามารถทำงานได้กับภาษา C++, C, Python, MATLAB และ Java และทำงานได้กับระบบปฏิบัติการเกือบทั้งหมด

OpenCV ถูกพัฒนาขึ้นโดยมีเป้าหมายในการเพิ่มประสิทธิภาพในการคำนวณ โดยมุ่งเน้นไปยังการทำงานแบบ real-time โดยภายใน OpenCV มีอัลกอริทึมมากกว่า 2,500 อัลกอริทึม ซึ่งอัลกอริทึมเหล่านี้ สามารถนำมาตรวจจับและหาวัตถุที่ต้องการ ตรวจจับหน้า การติดตามวัตถุ การค้นหารูปจากฐานข้อมูลและการทำงานอื่นๆ อีกมากมาย

2.5 Google Voice Typing

Google Voice Typing เป็นบริการเสริมจาก Google ที่มาพร้อมกับ Google keyboard สำหรับใส่ข้อความผ่านการพูด แทนการพิมพ์ข้อความในระบบปฏิบัติการแอนดรอยด์ ซึ่งการจะใช้ Google Voice Typing จะต้องมีการเชื่อมต่อกับอินเทอร์เน็ต เพื่อส่งข้อมูลไปประมวลผลกับส่วน Google Voice Recognition Service

Google Voice Typing รองรับภาษาได้หลายภาษา ซึ่งรวมไปถึงภาษาไทย โดยเริ่มต้น Google Voice Typing จะรองรับได้แค่ภาษาอังกฤษเท่านั้น แต่ผู้ใช้งานสามารถเลือกโหลดภาษาที่ต้องการได้

2.6 Text to Speech Synthesis (SVOX)

Text to Speech Synthesis หรือการสังเคราะห์เสียง เป็นเทคโนโลยีในการสร้างเสียงขึ้นมาตามความต้องการ โดยทำงานร่วมกับส่วนการประมวลผลภาษา การสังเคราะห์เสียงสามารถทำได้โดยการนำเสียงที่อยู่ในฐานข้อมูลมาเรียงต่อกัน คุณภาพของ Speech Synthesizer จะวัดจากความเหมือนและความชัดเจนของเสียงสังเคราะห์เมื่อเทียบกับเสียงของมนุษย์

SVOX เป็นหนึ่งในผู้นำด้านการพัฒนาเสียงสังเคราะห์ให้กับอุปกรณ์เคลื่อนที่ และรถยนต์ต่างๆ SVOX ได้พัฒนาระบบ TTS (Text to Speech) ที่ใช้ในการช่วยอ่านออกเสียงบนมือถือ โดยได้รับความนิยมมากในปัจจุบัน เนื่องจากรองรับภาษามากกว่า 25 ภาษาทั่วโลก รวมทั้งภาษาไทยด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 การตรวจจับใบหน้า

การตรวจจับใบหน้า (Face Detection) เป็นผลจากการพัฒนาการวิจัยของระบบการมองของคอมพิวเตอร์ (Computer Vision) อย่างต่อเนื่อง โดยวิธีพื้นฐานในการตรวจจับใบหน้าสามารถแยกได้ตามวิธีการเป็น 4 กลุ่ม [อ้างอิง] ดังนี้

2.7.1 วิธีการ Knowledge-based

วิธีการ Knowledge-based เป็นวิธีการในการสังเกตใบหน้าจากความเข้มของพิกเซลทั้งแนวตั้ง และแนวนอน โดยมีการกำหนดกฎเกณฑ์อย่างชัดเจน ว่ารูปที่มีใบหน้าจะต้องประกอบด้วยจุดสังเกตอะไรบ้าง และแต่ละจุดสังเกตจะต้องมีระยะห่างตามความสัมพันธ์ที่ได้ตั้งไว้ ซึ่งข้อเสียของวิธีการนี้ คือการสามารถตรวจจับใบหน้าได้ยากมาก เพราะมีการกำหนดกฎที่ค่อนข้างตายตัว ทำให้ไม่มีใบหน้าผ่านเกณฑ์ที่กำหนดไว้

2.7.2 วิธีการ Feature invariant

วิธีการ Feature invariant ใช้สำหรับการค้นหาลักษณะเด่นที่แตกต่างกันของใบหน้าโดยการตรวจจับขอบ (Edge Detection) แล้วนำมาวิเคราะห์เชิงสถิติ ซึ่งวิธีการนี้มีข้อเสียคือ สภาพของแสง เงา รวมถึงการถูกบดบัง จะทำให้ไม่สามารถตรวจจับใบหน้าได้

2.7.3 วิธีการ Template Matching

วิธีการ Template Matching เป็นการกำหนดใบหน้ารูปแบบมาตรฐาน แล้วนำภาพที่ต้องการตรวจหาใบหน้ามาหาค่าและความสัมพันธ์ต่างๆ เพื่อทำการเปรียบเทียบ มีข้อเสียคือประสิทธิภาพในการตรวจจับยังไม่ดีเท่าที่ควร แต่เป็นวิธีการที่ง่าย

2.7.4 วิธีการ Appearance – based

วิธีการ Appearance – based เป็นวิธีการที่อยู่บนพื้นฐานของการวิเคราะห์เชิงสถิติและการเรียนรู้ของเครื่อง (Machine Learning) โดยการใช้โมเดลที่ได้จากการเรียนรู้กลุ่มภาพตัวอย่างในการหาคุณลักษณะความสัมพันธ์ของใบหน้า ซึ่งวิธีการนี้เป็นวิธีที่ได้รับการยอมรับมากที่สุด เพราะสามารถตรวจจับได้อย่างรวดเร็วและมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 Haar-Like Features

Haar-Like Features เป็นวิธีการหนึ่งที่ถูกนำมาใช้ในการตรวจจับวัตถุโดย Paul viola และ Michael J. Jones ซึ่งนำแนวคิดของ Haar wavelets มาใช้ และเรียกวิธีการนี้ว่า Haar-Like Features

การทำงานของ Haar-Like Features ทำงาน โดยการพิจารณาพื้นที่สี่เหลี่ยมที่อยู่ติดกันที่ตำแหน่งภายในหน้าต่าง จากนั้นรวมค่าความเข้มของสีในแต่ละพื้นที่และคำนวณความแตกต่างระหว่างผลลัพธ์เหล่านี้ จากนั้นความแตกต่างที่ได้มาใช้แยกแยะส่วนต่างๆ ของภาพต่อไป โดยข้อดีของ Haar-Like Features คือการคำนวณที่เร็ว และเวลาในการคำนวณค่อนข้างคงที่ แม้ขนาดหน้าต่างจะแตกต่างกัน

2.9 Morphology (Opening and Closing)

Morphology เป็นกระบวนการในการประมวลผลภาพที่ถูกนำมาใช้ในการแยกคุณสมบัติของภาพออกมา เช่น contour, skeletons และ convex hulls เป็นต้น และสามารถนำมาใช้กับการทำ morphological filtering, thickening หรือ thinning ได้ โดยการทำงานหลักของ morphology ประกอบไปด้วย ไปด้วย dilation (การขยายภาพ), erosion (การย่อภาพ), closure (การปิดช่องโหว่ภายในภาพ) และ disconnection (การลด noise ภายในภาพ)

การทำ opening และ closing เป็นเทคนิคที่มีประโยชน์อย่างมากในการจัดการกับภาพที่มีสิ่งรบกวนเยอะ หรือพิกเซลที่มีค่าผิดแปลกไป ซึ่งปกติแล้ววิธีการดังกล่าวสามารถสรุปการเลือกใช้ได้ดังนี้ ถ้าต้องการที่จะทำให้ขอบขอบรูปภาพมีความต่อเนื่องกันให้ใช้ opening แต่ถ้าอยากให้ภาพนั้นไม่มีช่องโหว่หรือไม่มีรูปให้ใช้ closing

ก) ข) ค)

รูป 2.2 การทำ opening และ closing

ก) รูปต้นฉบับ

ข) รูปต้นฉบับเมื่อทำ Opening

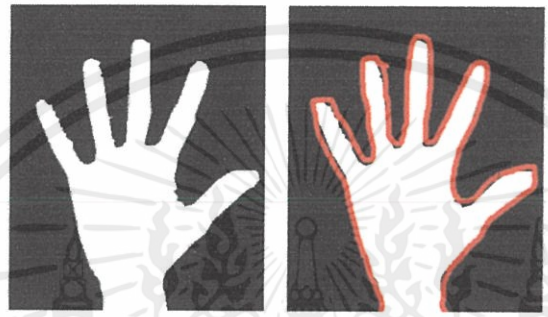
ค) รูปต้นฉบับเมื่อทำ Closing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 เทคนิคในการทำ Fingertips Tracking

เทคนิคในการทำ Fingertips tracking เป็นเทคนิคที่ใช้ในการหาตำแหน่งของปลายนิ้วมือ โดยเทคนิคนี้จะใช้เทคนิคของ Convexity defects และ convex hull มาช่วยในการหาและระบุตำแหน่งของนิ้วมือ โดยมีขั้นตอนในการทำงานดังนี้

1) นำภาพมือที่เป็นภาพแบบ binary มาทำ contour extraction เพื่อให้ได้ตำแหน่งของบริเวณที่มีความแตกต่างเกิดขึ้น (รูป 2.3)



ก) ข)

รูป 2.3 การทำ contour extraction

ก) ภาพ binary ของมือ

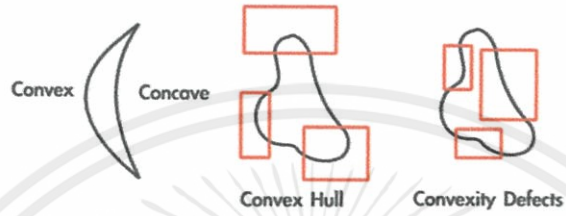
ข) ภาพ binary ของมือหลังจากทำ contour extraction

2) นำภาพที่ผ่านการทำ contour extraction มาหาจุดกึ่งกลางของมือ โดยใช้วิธี maximum inscribed circle and radius (รูป 2.4)

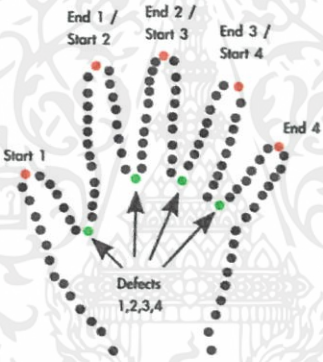
รูป 2.4 การทำ maximum inscribed circle and radius

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ทำการหาจุดปลายนิ้วโดยใช้เทคนิค Convex hull and convexity defects extraction ซึ่งเป็นขั้นตอนสำคัญที่ใช้ในการหาจุดปลายนิ้วของนิ้วต่างๆ โดยที่ convex hull จะนำมาใช้เพื่อหาจุดที่เว้าออก และใช้ convexity defects เพื่อหาจุดที่เว้าเข้า ดังรูป 2.5 เมื่อทำการใช้ฟังก์ชันทั้งสองแล้วจะได้จุด convex และ defects ดังรูป 2.6 โดยสีแดงคือจุด convex hull ส่วนจุดสีเขียนคือ จุด convexity defects

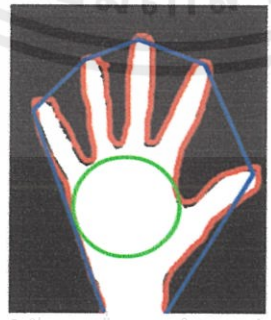


รูป 2.5 convex hull และ convexity defects



รูป 2.6 convex hull และ convexity defects ของ contour ของรูปมือ

4) หลังจากที่ทำทั้งสามขั้นตอนแล้วผลลัพธ์ที่ได้เป็นดังนี้ (รูป 2.7)



รูป 2.7 ผลจากการหาตำแหน่งของจุดปลายนิ้ว

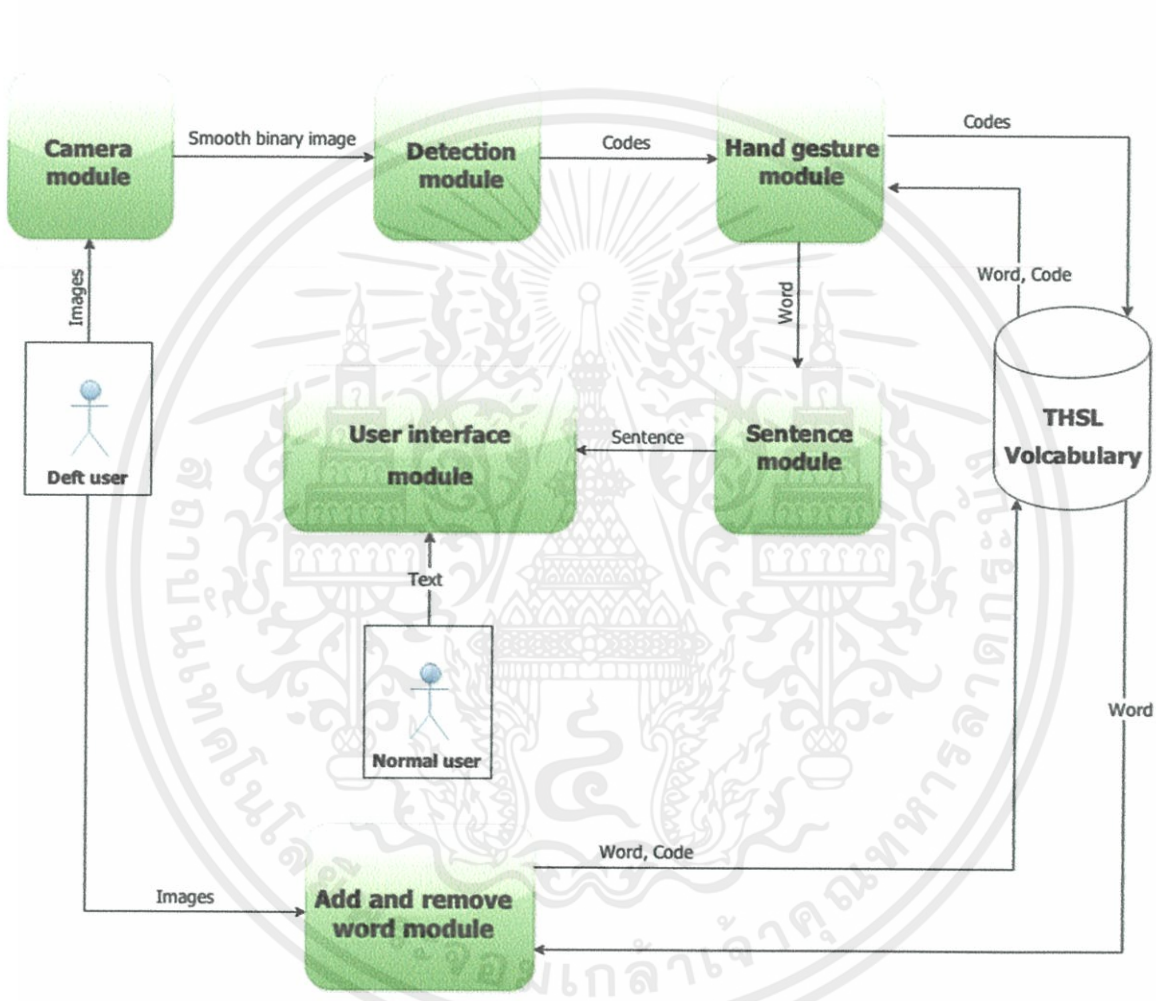
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและพัฒนาระบบ

3.1 ภาพรวมของโครงสร้างระบบ

ในโครงสร้างของระบบจะแบ่งส่วนการทำงานออกเป็น 6 ส่วนหลักๆ ได้แก่



รูป 3.1 ภาพรวมของโครงสร้างระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1 Camera module

ส่วนการทำงานของ Camera module เป็นส่วนที่ทำการติดต่อกับกล้องของแอนดรอยด์ โดยทำการจับภาพวีดีโอแล้วนำมาแยกออกเป็นเฟรม จากนั้นนำเฟรมเหล่านั้นมาทำการประมวลผลภาพเพื่อให้ได้เป็นรูปภาพ binary ซึ่งแสดงถึงลักษณะของมือ

3.1.2 Detection module

ส่วนการทำงานของ Detection module เป็นส่วนที่ทำหน้าที่ตรวจจับและหาลักษณะของมือออกมา เช่น ลักษณะของมือ นิ้วมือ จุดกึ่งกลางมือ เป็นต้น หลังจากนั้นนำผลลัพธ์ที่ได้แปลงออกมาเป็นโค้ด โดยมีหลักการการทำงานคือ เมื่อได้รูป binary จากส่วน Camera module ก็จะนำรูปที่ได้นั้นมาหาลักษณะของมือที่ได้กล่าวไปข้างต้น โดยใช้กระบวนการทางการประมวลผลภาพ เช่น Contour Polygon Approximation เป็นต้น

3.1.3 Hand gesture module

ส่วนการทำงานของ Hand gesture module เป็นส่วนที่นำโค้ดที่ได้จาก Detection module มาทำการค้นหาความน่าจะเป็นของคำนั้นๆ โดยหลักการคือ นำโค้ดมาทำการหาความน่าจะเป็นของท่าทางของคำโดยทำการค้นหาลงในฐานข้อมูลที่ได้ถูกสร้างเป็นแบบโครงสร้างของต้นไม้ หลังจากนั้นทำการส่งผลลัพธ์ที่ได้ส่งไปยัง Word processing module

3.1.4 Sentence module

ส่วนการทำงานของ Sentence module จะทำหน้าที่นำคำที่ได้จากส่วน Hand gesture module มาทำการจัดเรียงให้เป็นประโยค และทำการส่งผลลัพธ์ไปยัง User interface module

3.1.5 User interface module

ส่วนการทำงานของ User interface module เป็นส่วนที่รับอินพุทจากคนหูหนวกและคนทั่วไปมาประมวล โดยทำเป็นเหมือนการคุยติดต่อกันในรูปแบบของโปรแกรมแชท

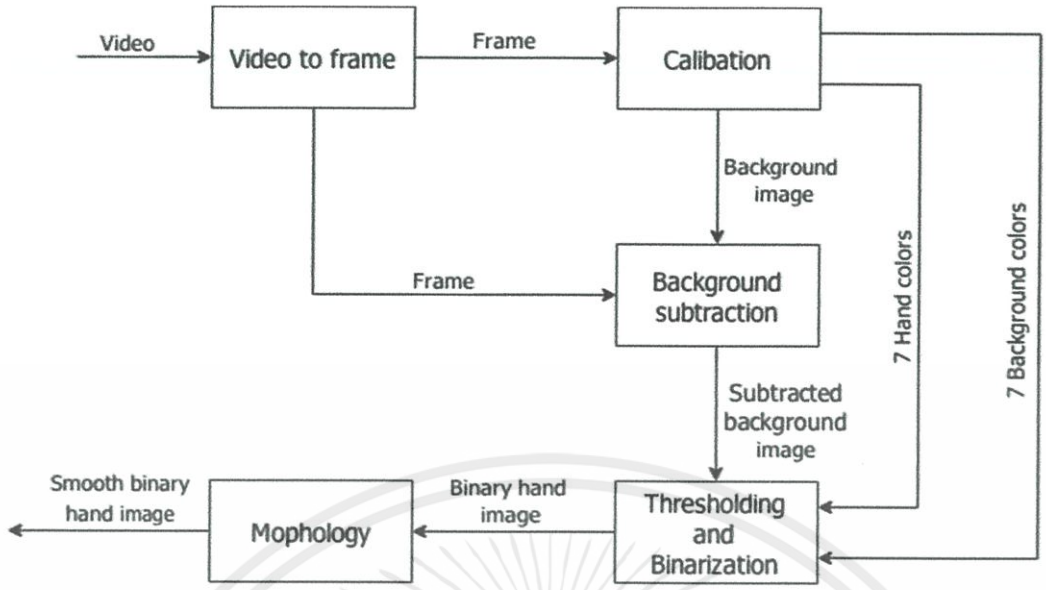
3.1.6 Add and remove word module

ส่วนการทำงานของ Add and remove word module เป็นส่วนที่ทำหน้าที่ในการเพิ่มหรือลบคำศัพท์ภาษามือในฐานข้อมูลที่มือชื่อว่า ThSL Vocabulary

3.2 Camera module

ในส่วนการทำงานนี้ เริ่มต้นจาการรับวีดีโอจากกล้องของมือถือแอนดรอยด์ หลังจากนั้นนำวีดีโอ ที่ได้นำมาแบ่งออกเป็นเฟรม และนำเฟรมที่ได้มาทำการประมวลผลภาพ ซึ่งจะอธิบายรายละเอียด ในหัวข้อ 3.2.1 ถึง 3.2.5 ดังนี้

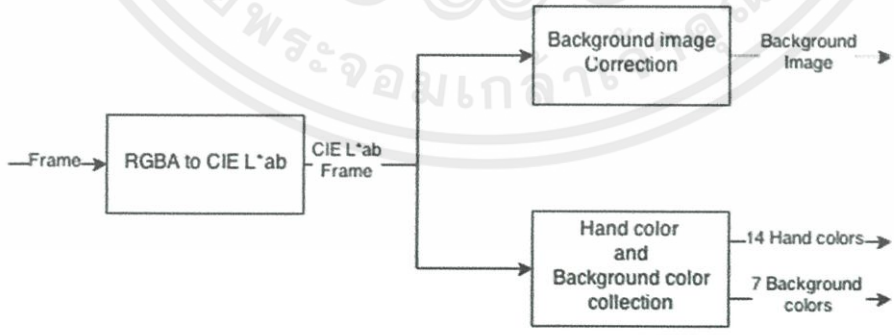
เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.2 การทำงานของ Camera Module

3.2.1 Calibration

ในส่วน calibration เป็นส่วนที่ช่วยทำให้การหาลักษณะของมือนั้นมีประสิทธิภาพมากขึ้น โดยมีหลักการทำงาน (รูป 3.3) คือ ขั้นตอนแรกจะนำเฟรมที่ได้ซึ่งอยู่ใน color space แบบ RGBA นำมาเปลี่ยนเป็น CIE L*ab เนื่องจากมีประสิทธิภาพอย่างมากในการหาสีของมือที่มีปัจจัยของแสงเข้ามาเกี่ยวข้อง จากนั้นนำเฟรมที่ได้มาทำการวิเคราะห์และตรวจจับภาพพื้นหลัง เพื่อที่จะนำเอาเฟรมนี้เป็นเฟรมต้นแบบและนำไปใช้ในส่วนของ background subtraction และนำมาเก็บรายละเอียดสีของพื้นหลัง และสีของมือทั้งสองข้าง โดยสีของพื้นหลังจะเก็บมาทั้งหมด 7 จุด ส่วนสีของมือนั้นจะเก็บทั้งหมด 14 จุด เพื่อนำจุดที่ได้เหล่านี้ไปทำการประมวลผลและหาค่าเฉลี่ยของสี



รูป 3.3 การทำงานของส่วน Calibration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 Background subtraction

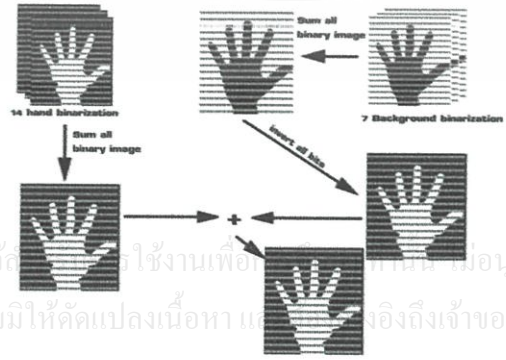
ส่วนของ Background subtraction จะทำการตัดภาพพื้นหลังออกมาจากรูปภาพโดยการหาความแตกต่างระหว่างภาพพื้นหลังที่ได้จากข้อ 3.2.1 และเฟรมปัจจุบันที่รับเข้ามา ซึ่งจะทำให้การตรวจจับลักษณะของมือนั้นมีประสิทธิภาพมากยิ่งขึ้น โดยในส่วนของ Background subtraction จะใช้โมเดล MOG2 ซึ่งอยู่ในไลบารี่ของ OpenCV

3.2.3 Thresholding and binarization

เป็นส่วนที่นำภาพจากส่วนของ Background subtraction มาทำการหารูปร่างลักษณะของมือ โดยภาพผลลัพธ์ที่ได้จะเป็นภาพของมือซึ่งเป็นภาพแบบไบนารี่ ส่วนหลักการทำงานในส่วนของ thresholding and Binarization มีดังนี้

ในขั้นตอนแรกจะทำการนำรูปที่ได้จากส่วนของ Background subtraction มาแยกออกเป็น 3 ช่องสีคือ C I และ E หลังจากนั้นนำสีของมือ 14 จุด และ สีของพื้นหลัง 7 จุดมาทำการประมวลผล โดยก่อนที่จะนำไปประมวลผลจะนำสีของมือทั้ง 14 จุด และสีของพื้นหลังทั้ง 7 จุด ไปเข้าในส่วนของ Hand color bounding และ Background color bounding เพื่อให้ได้ช่วงของค่าสีสำหรับการทำ thresholding

ในส่วนของ Thresholding and Binarization จะนำช่วงสีที่ได้จากส่วนของ Hand color bounding และ Background color bounding มาทำการกำหนดเงื่อนไขว่าส่วนใดของภาพควรเป็น 0 หรือ 1 ถ้าค่าสีอยู่ในช่วงที่กำหนดไว้ก็จะเปลี่ยนให้เป็น 1 นอกนั้นเป็น 0 ตามรูป (เดี๋ยวมาทำ) โดยการทำ Thresholding and Binarization ของสีมือจะทำทั้งหมด 14 ครั้ง และสีของพื้นหลังจะทำทั้งหมด 7 ครั้ง โดยแต่ละครั้งจะใช้ช่วงสีที่แตกต่างกัน จากนั้นนำภาพจากการทำ Thresholding and Binarization มารวมกัน โดยนำภาพของมือทั้ง 14 ภาพมารวมกันให้เป็นรูปเดียว (รูป 3.4) และสีของพื้นหลังทั้ง 7 ภาพมารวมกันให้เป็นรูปเดียว (รูป 3.4) แต่ในส่วนของภาพพื้นหลังจะทำการกลับบิทในภาพทั้งหมด ในขั้นตอนสุดท้ายจะนำภาพของมือและพื้นหลัง มาทำการหาความเหมือนโดยใช้ and operation ซึ่งจะได้เป็นภาพผลลัพธ์ของมือซึ่งอยู่ในรูปแบบ binary

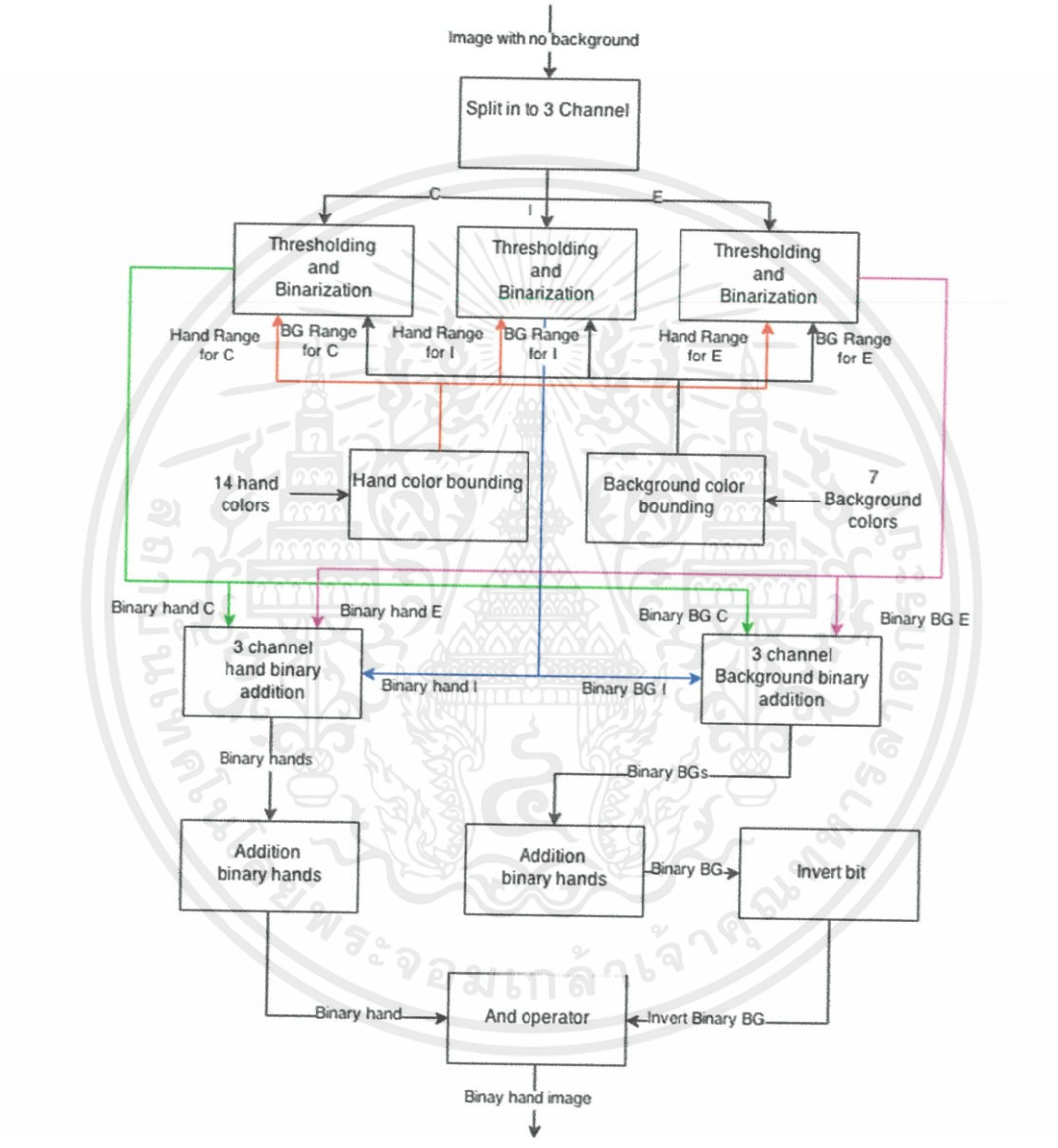


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 3.4 การรวมกันของภาพ binary มือและพื้นหลัง

3.2.4 Morphology

ขั้นตอนนี้จะสำคัญทำการลบสิ่งรบกวนออกจากภาพ โดยการใช้ morphology ประเภทOpening and closing เพื่อที่ทำจะให้ภาพที่ได้ไม่มีสิ่งรบกวน(noise) อีกทั้งยังช่วยทำการเติมเต็มรูปร่างของมือให้ชัดเจนและมีความต่อเนื่องกันมากขึ้น



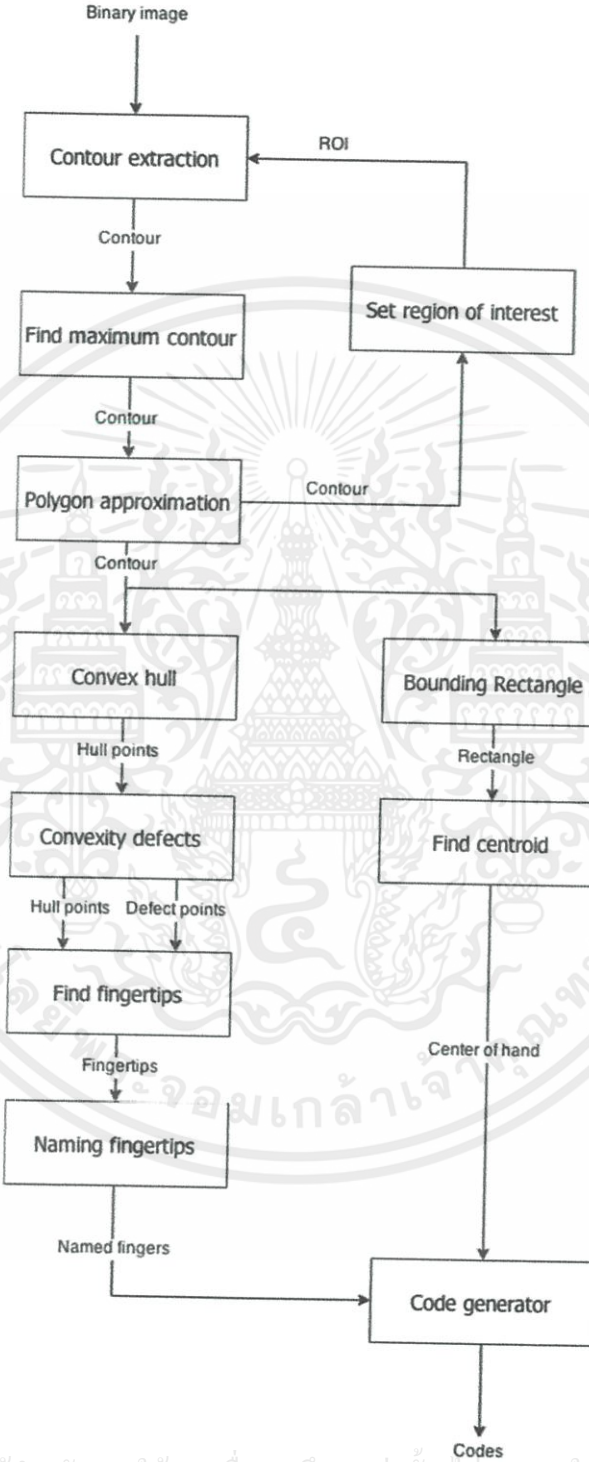
รูป 3.5 การทำงานของส่วน thresholding and binarization

3.3 Detection module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่ได้ภาพไบนารี่ของมือจาก camera module ในส่วนของ Detection module (รูป 3.6) จะทำการหารายละเอียดของมือจากภาพที่ได้มา โดยรายละเอียดที่จะทำการค้นหา ได้แก่ ตำแหน่งของมือ จุดกึ่งกลางมือ ปลายนิ้วมือและการตั้งชื่อนิ้วมือ ระยะห่างระหว่างนิ้วมือ และระยะห่าง

ระหว่างปลายนิ้วมือถึงจุดกึ่งกลางมือ หลังจากนั้นนำรายละเอียดที่ได้มาแปลงเป็นโค้ด เพื่อนำไปคำนวณในส่วนการทำงานต่อไป



รูป 3.6 การทำงานในส่วนของ Detection module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1 Contour extraction Find maximum contour and polygon approximation

จากภาพ binary ที่ได้จาก Camera module ในภาพไบนารี่นั้นจะประกอบไปด้วย white blob และ back blob ซึ่ง white blob ก็คือภาพของมือ โดยส่วนการทำงานนี้จะทำการหา contour ทั้งหมดในภาพ ซึ่งจะเลือก contour ที่ใหญ่ที่สุดมาใช้ในการคำนวณ สาเหตุที่ต้องทำการหา contour ที่ใหญ่ที่สุดเนื่องจากบางครั้งภาพที่เราได้อาจมีสิ่งรบกวนปนเข้ามาด้วย ทำให้เวลาทำ contour extraction จะได้ contour ออกมาจำนวนหลาย contour และยังสามารถลบสิ่งรบกวนออกไปจากภาพได้อีกด้วย หลังจากนั้นนำ contour ไปลดจำนวนของ vertices เพื่อให้ได้ contour ที่มีรูปร่างของมือชัดมากยิ่งขึ้น โดยใช้วิธี polygon approximation การทำ polygon approximation นอกจากจะทำให้ contour ที่ได้มีรูปร่างที่ชัดเจนแล้ว ยังช่วยให้ลดขนาดของการประมวลผลได้ด้วย

3.3.2 Setting Region of Interest (ROI) and finding min enclosing circle

ส่วนการทำงานนี้เราจะใช้ค่ารัศมีที่หาได้จากข้อ 3.3.1 หลังจากนั้นนำมาคูณด้วย 3.5 และทำการกำหนด ROI คือทำการตั้งค่าจุดที่สนใจ ซึ่งการทำเช่นนี้จะทำให้เราสามารถลบส่วนที่เราไม่สนใจออกได้เช่น แขน เป็นต้น หลังจากที่กำหนดส่วนที่สนใจได้แล้ว ก็จะนำส่วนที่สนใจนี้ไปทำในข้อ 3.3.1 อีกครั้ง เพื่อให้ได้ contour ที่มีรูปร่างที่เป็นมือ

รูป 3.7 การหา min enclosing circle เพื่อนำมาใช้ในการตั้งค่า ROI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 Convex hull, convexity defects extraction, and find fingertips

ส่วนนี้เป็นอัลกอริทึมที่สำคัญสำหรับใช้ในการหาจุดปลายนิ้ว และง่ามนิ้ว โดยที่ convex hull จะนำมาใช้เพื่อหาจุดที่เว้าออก ดังรูป 3.8 และใช้ convexity defects เพื่อหาจุดที่เว้าเข้า ดังรูป 3.9

รูป 3.8 การหาจุด convex hull (จุดสีเขียว)

รูป 3.9 การหาจุด convexity defect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

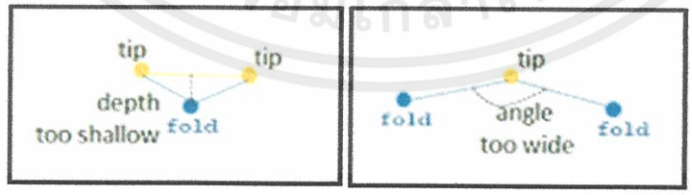
หลังจากนั้นจะนำจุดที่ได้จากการทำ convex hull และ convex defects มาเข้าอัลกอริทึมเพื่อคัดกรองว่าส่วนไหนเป็นจุดปลายนิ้ว และเป็นง่ามนิ้วมือจริงๆ โดยอัลกอริทึมมีการทำงานดังนี้

ขั้นตอนแรก ทำการกรองจุดง่ามนิ้ว โดยการนำจุดที่เป็น Defects ทั้งหมดมาทำการวนหาค่าโดยมีเงื่อนไขดังนี้ ถ้า $Defect_N$ โดยที่ N%4 เท่ากับ 2 (เป็นค่าที่ได้จากการทดลอง) จะพิจารณาให้จุดนั้นเป็นจุดง่ามนิ้ว ซึ่งจะได้ผลลัพธ์ดังรูป 3.10

ขั้นตอนที่สอง ในขั้นตอนนี้จะเป็นการหาจุดปลายนิ้ว โดยจะเป็นการทดสอบว่าจุดระหว่าง convex hull และค่า defect ที่ได้จากขั้นตอนแรก หลังจากนั้นตรวจสอบว่าความลึกถึงเกณฑ์ที่กำหนดไว้หรือไม่ ถ้าถึงให้ไปทำขั้นตอนที่ 3 แต่ถ้าไม่ถึงให้กลับไปทำในขั้นตอนที่สองโดยเลือกจุดถัดไป

ขั้นตอนที่สาม เป็นขั้นตอนในการหาขนาดของมุมระหว่างจุด convex hull ไปยังจุดด้านซ้าย และด้านขวา โดยพิจารณาว่ามุมมีความกว้างเกินกว่าค่าที่ได้กำหนดไว้หรือไม่ ถ้าไม่เกิดแสดงว่าจุดที่พิจารณาดังกล่าวเป็นจุดปลายนิ้วมือ ดังรูป 3.12

รูป 3.10 ภาพหลังจากการผ่านอัลกอริทึมเพื่อหาจุด defects บริเวณร่องนิ้ว



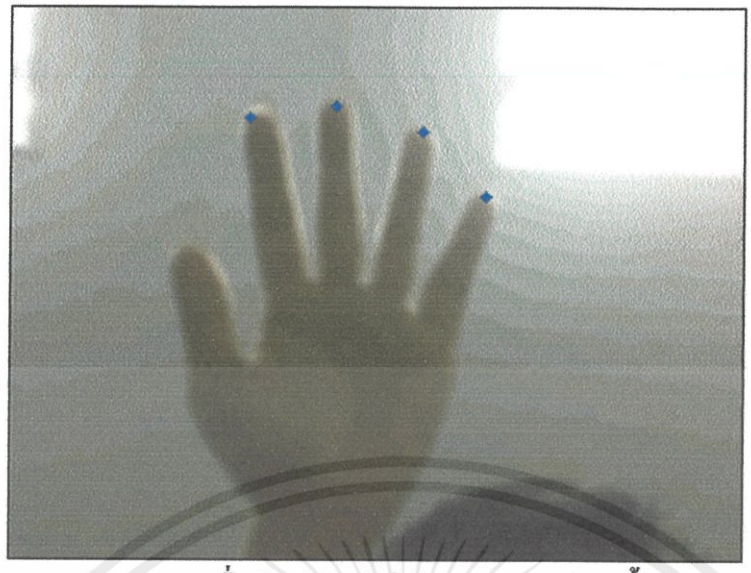
ก) ข)

รูป 3.11 การหาจุดปลายนิ้ว

ก) การหาความลึกระหว่างจุดกึ่งกลางปลายนิ้วถึงง่ามนิ้ว

ข) การหาองศาของมุมระหว่างจุดปลายนิ้วถึงจุดง่ามนิ้วทั้งสอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.12 ภาพเมื่อผ่านอัลกอริทึมในการหาจุดปลายนิ้วมือ

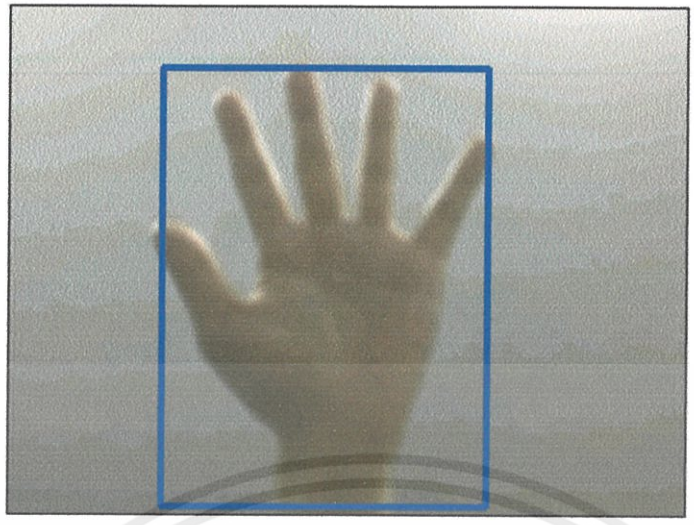
3.3.4 Naming fingertips

ในส่วนการทำงานนี้จะนำจุดปลายนิ้วที่หาจากส่วนที่ 3.3.3 มาทำการตั้งชื่อจุดปลายนิ้ว โดยมีหลักการดังนี้ ขั้นตอนแรก ถ้าเจอนิ้วครบทุก 5 นิ้ว ให้ทำการตั้งนิ้วทั้งหมดตามลำดับของนิ้ว คือ โป้ง ชี้ กลาง นาง และก้อย แต่ถ้าเจอนิ้วไม่ครบ 5 นิ้ว ให้ทำดังขั้นตอนต่อไปนี้ ขั้นตอนที่สองทำการค้นหานิ้วโป้งและนิ้วก้อยก่อน โดยการหานิ้วโป้งจะมีหลักการคือ ดูว่าจุดปลายนิ้วอยู่บริเวณซ้ายสุดของภาพหรือไม่ ถ้าใช่แสดงว่านิ้วที่พิจารณาอยู่คือนิ้วโป้ง ส่วนนิ้วก้อยมีหลักการในการพิจารณาคล้ายๆกับนิ้วโป้ง แต่ให้พิจารณาว่านิ้วอยู่ด้านขวาสุดใช้หรือไม่ จากนั้นทำการกำปลายนิ้วที่อยู่ ณ ตำแหน่งจุดกึ่งกลางของมือให้เป็นนิ้วกลาง นิ้วที่อยู่ด้านซ้ายของนิ้วกลางให้เป็นนิ้วชี้ และด้านขวาให้เป็นนิ้วกลาง

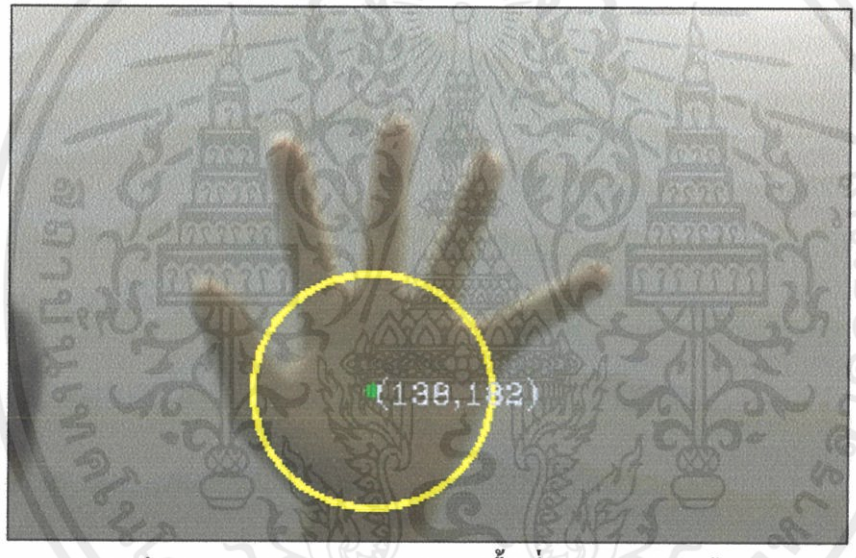
3.3.5 Bounding rectangle and find centroid

ในส่วนการทำงานนี้จะเป็นการหาพื้นที่สี่เหลี่ยมที่เล็กที่สุดที่สามารถครอบคลุมบริเวณของ contour ทั้งหมดได้ (Bounding rectangle) ตามรูป *** หลังจากหานั้นจะจำค่า Bounding rectangle ดังกล่าว มาทำการหาจุดกึ่งกลางมวลของมือซึ่งจะได้ผลลัพธ์ตามรูป *** และสามารถนำจุดกึ่งกลางของมือไปหาพื้นที่วงกลมของฝ่ามือได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.13 การหา Bounding rectangle



รูป 3.14 การหาจุด centroid และพื้นที่วงกลมของฝ่ามือ

3.3.6 Code generator

ในส่วนนี้จะทำการนำส่วนต่างๆที่เราหาได้มาคำนวณเป็นโค้ด ผลลัพธ์ที่ได้จากการประมวลผลภาพจะถูกนำมาจัดให้อยู่ในรูปของ code โดยภาพแต่ละเฟรมจะสามารถสร้าง code ได้1 code ซึ่งตัวโครงสร้างของ code จะถูกแบ่งออกเป็น 3 ส่วน โดยมีรูปแบบดังนี้

$$A_1 - A_2 - A_3 - A_4 - A_5_B_1 - \ldots - B_n_C_1C_2$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

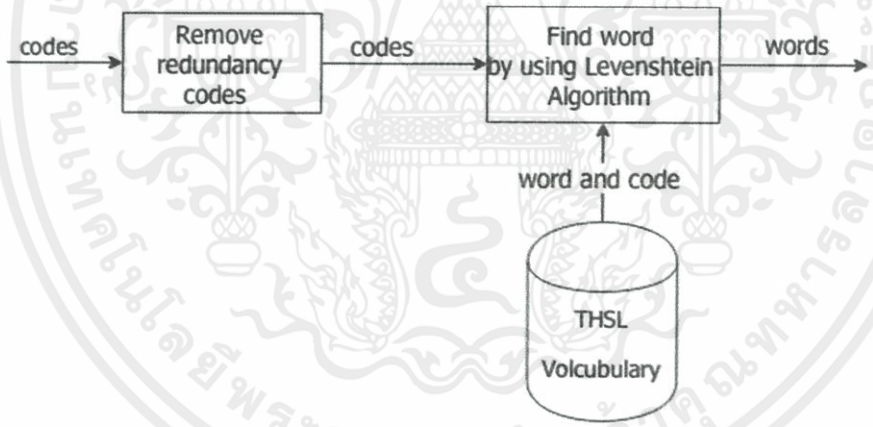
ส่วน A เป็นส่วน code ที่ระบุว่าเจอปลายนิ้วมือหรือไม่ โดยค่าที่กำหนดจะมีค่าเท่ากับ 0 (หาปลายนิ้วไม่เจอ) และ 1 (หาปลายนิ้วเจอ) โดย code จะเรียงจากปลายนิ้วมือจากนิ้วหัวแม่โป้งจนถึงนิ้วก้อยตามลำดับ (A_1-A_5)

ส่วน B เป็นส่วน code ที่เก็บค่าระยะห่างระหว่างปลายนิ้วมือ ซึ่งจำนวนของค่า B จะแตกต่างกันไปตามจำนวนปลายนิ้วมือที่เจอ (ส่วน A) โดยค่ามากสุดที่มีได้คือ 4 (กรณีเจอปลายนิ้วครบ) ตัวอย่างเช่น หากผลลัพธ์ของ code ที่ได้เจอนิ้วเพียงสามนิ้ว ค่าของ B ก็จะมีแค่เพียง 2 ค่า เป็นต้น

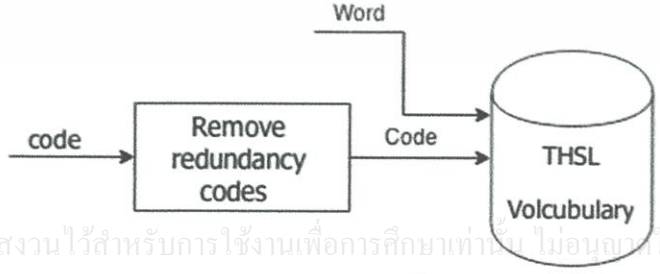
ส่วน C เป็นส่วน code ที่ระบุทิศทางการเคลื่อนไหวของนิ้วมือ โดยค่า C1 จะเป็นส่วนระบุการเคลื่อนไหวขึ้นลง มีค่าที่เป็นไปได้เท่ากับ U(up) หรือ D(down) หรือ N(none) และส่วนของค่า C2 เป็นการระบุการเคลื่อนไหวซ้ายขวา มีค่าที่เป็นไปได้คือ R(right) หรือ L(left) หรือ N(none)

3.3.7 Hand Gesture module

ส่วนนี้จะทำหน้าที่ในการคัดกรองและจัดเก็บ code ภาษามือลงในฐานข้อมูล ตลอดจนดึง code จากฐานข้อมูลมาประมวลผลเพื่อหาความหมาย



รูป 3.15 การทำงานของ Hand gesture module



รูป 3.16 การทำงานของส่วนเพิ่มและลบคำศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเอกสารทุกครั้งที่มีการนำไปใช้

3.3.8 ส่วนการเก็บข้อมูล

เมื่อได้รับ code จากส่วนของ code generator code ที่เข้ามาจะถูกนำมาพิจารณาความซ้ำซ้อน โดยใช้การเทียบ code ลำดับที่ n และลำดับที่ n+1 ว่ามีความแตกต่างมากกว่าค่าที่กำหนดไว้หรือไม่ หากน้อยกว่าค่าที่กำหนด code ลำดับที่ n+1 จะถูกคัดทิ้ง

ตัวอย่างการคัด code ที่มีความซ้ำซ้อนกัน โดยมีค่ากำหนดไว้เท่ากับ 0,3,0 หมายความว่า ส่วน A และส่วน C ของ code ลำดับที่ n และ n+1 จะต้องเหมือนกันทุกตัวอักษร และแต่ละค่าตัวเลขของส่วน B จะมีความคลาดเคลื่อนได้ไม่เกิน +- 3 code ลำดับที่ n+1 จึงจะไม่ถูกคัดทิ้ง

จาก code ตัวอย่าง จะเห็นว่า ส่วน A และส่วน C ของทั้งสอง code เหมือนกันทุกตัวอักษร แต่ในส่วน B ค่าตัวเลขมีความแตกต่างเกินเกณฑ์ที่กำหนด ดังนั้น code ที่สองจะต้องถูกตัดทิ้ง

1-1-0-0-1_46-10_NN	1-1-0-0-1_12-43_NN

เมื่อได้ผลลัพธ์ทั้งหมด code จะถูกนำมาต่อกันโดยใช้เครื่องหมาย "|" เป็นตัวแบ่งส่วนของ code และจัดเก็บลงฐานข้อมูลเพื่อรอการค้นหาต่อไป

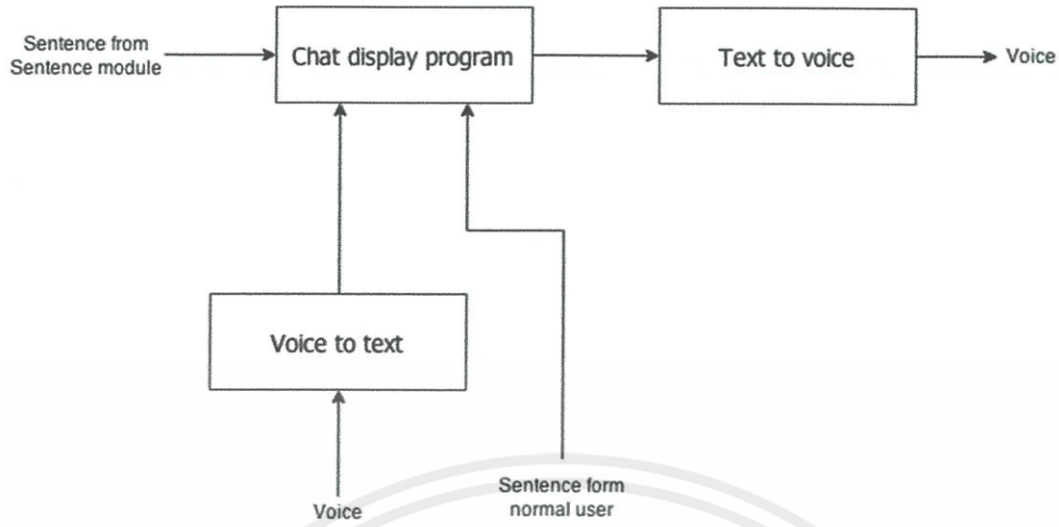
3.3.9 ส่วนการค้นหาข้อมูล

ในการค้นหา code ที่ใกล้เคียงกันมากที่สุด ผู้จัดทำได้เลือกใช้อัลกอริทึม Levenshtein Distance ในการทำ string matching ซึ่งใช้หลักการการให้น้ำหนักค่าความใกล้เคียงจากการเพิ่ม ลบ แทนที่ตัวอักษร ในการค้นหา ผู้จัดทำได้ทำการเรียงลำดับ code ที่ใกล้เคียงกับ input code มากที่สุด 6 ลำดับ เพื่อใช้สำหรับเป็นคำแปล และคำใกล้เคียง

3.4 User interface module

ส่วน User Interface module เป็นส่วนเชื่อมต่อระหว่างผู้ใช้งานกับฟังก์ชั่นที่อยู่ในโปรแกรมทั้งหมด ซึ่งสามารถแบ่งเป็น 3 ส่วนหลัก ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.17 การทำงานส่วน App chat module

3.4.1 ส่วน Chat display program

เป็นส่วนหน้าจอการพูดคุยกันระหว่างผู้บกพร่องทางการได้ยิน และบุคคลธรรมดา ประกอบไปด้วยส่วนต่างๆ คือ

1) ส่วนกล่องข้อความแชท
2) ส่วนพิมพ์ข้อความของบุคคลทั่วไป
3) ส่วนส่งข้อความโดยใช้เสียง
4) ส่วนเชื่อมต่อไปยังหน้าต่างแปลภาษามือ

รูป 3.18 Chat display program

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2 Text to voice

ส่วนของ text to voice เป็นส่วนที่จะนำข้อความที่เป็นผลลัพธ์ให้อยู่ในรูปของเสียง โดยใช้ API ของ SVOX ในการแปลงข้อความให้เป็นเสียง

3.4.3 Voice to text

ส่วนของ voice to text เป็นส่วนการรับข้อมูลในรูปแบบเสียงเพื่อนำมาแสดงให้อยู่ในรูปของข้อความ โดยใช้ Google voice typing ซึ่งเป็นบริการเสริมจาก Google Keyboard (ในส่วนนี้ จะสามารถใช้ได้เฉพาะอุปกรณ์ที่มีระบบปฏิบัติการแอนดรอยด์ เวอร์ชั่น 4.1 (Jelly bean) ขึ้นไป และมีการเชื่อมต่อกับอินเทอร์เน็ต)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลอง และผลการทดลอง

4.1 การทดลองแอพพลิเคชั่นส่วนจัดการบทสนทนา

ส่วนจัดการบทสนทนาประกอบไปด้วยส่วนแสดงบทสนทนา ส่วนการป้อนบทสนทนาด้วยข้อความและเสียง และส่วนแปลภาษามือ โดยส่วนแสดงบทสนทนาจะแบ่งออกเป็น 2 ฝั่ง คือ ฝั่งผู้บกพร่องทางการได้ยิน(ฝั่งซ้าย) และ ฝั่งบุคคลธรรมดา(ฝั่งขวา)

รูป 4.1 ส่วนจัดการบทสนทนา

ก) ข)

รูป 4.2 รูปสำหรับระบุฝั่งของบทสนทนา

ก) ผู้บกพร่องทางการได้ยิน

ข) บุคคลทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1 การใช้งานเพื่อสื่อสารของคนปกติ

การใช้งานเพื่อการสื่อสารของคนปกติ ได้ทำการทดลองหาความถูกต้องของการแปลงเสียงเป็นข้อความ โดยแบ่งออกเป็นสองส่วน ดังนี้

1) <u>การทดลองส่วนที่ 1</u> ทำการสุ่มคำศัพท์ 10 คำ มาทดลองการแปลงเสียงเป็นข้อความจำนวน 5 ครั้ง โดยกำหนดให้เสียงคนผู้พูดเป็นคนเดียวกัน และมีความเร็วในการพูดเท่ากัน จากนั้นนำผลลัพธ์ที่ได้ออกมาคำนวณหาความถูกต้องจาก

$$\frac{\text{จำนวนครั้งที่ผลลัพธ์ถูกต้อง}}{\text{จำนวนครั้งที่ทำการทดลอง}} \times 100 \qquad (4.1)$$

2) <u>การทดลองส่วนที่ 2</u> ทำการทดลองเช่นเดียวกับการทดลองส่วนที่ 1 และเปลี่ยนจากการสุ่มคำศัพท์เป็นการสุ่มประโยค 10 ประโยค และนำมาคำนวณหาเปอร์เซ็นต์ความถูกต้อง

ตาราง 4.1 ความถูกต้องของการแปลงคำศัพท์จากเสียงเป็นข้อความ

คำศัพท์	ผลลัพธ์ที่ได้					เปอร์เซ็นต์ความถูกต้อง
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4	ครั้งที่ 5	
สวัสดี	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
ขอบคุณ	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
พ่อ	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
แม่	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
พี่น้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
ต้องการ	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
สวยงาม	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
กรณี	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
ตัวอักษร	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
กฤษณา	ผิด	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	80
ทุกขลาภ	ถูกต้อง	ถูกต้อง	ผิด	ถูกต้อง	ถูกต้อง	80

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.2 ความถูกต้องของการแปลงประโยคจากเสียงเป็นข้อความ

คำศัพท์	ผลลัพธ์ที่ได้					เปอร์เซ็นต์ความถูกต้อง
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4	ครั้งที่ 5	
สวัสดี เป็นอย่างไรบ้าง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
วันนี้ฉันไปตลาดกับแม่	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
พ่อไปวิ่งแต่เช้า	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
แม่ร้องเพลง	ผิด	ผิด	ถูกต้อง	ถูกต้อง	ถูกต้อง	60
พี่น้องรักกัน	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
ต้องการเงิน	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
เก้าอี้สวยงาม	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
วันนี้เป็นวันหยุด	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
เขียนโปรแกรม	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
ยายกินลำไยน้ำลายยายไหลย้อย	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	ถูกต้อง	100
ทุกขลาภเป็นสิ่งไม่จีรัง	ผิด	ผิด	ถูกต้อง	ผิด	ถูกต้อง	40

จากผลการทดลองในตาราง 4.1 และตาราง 4.2 สามารถสรุปได้ว่า การแปลงเสียงเป็นข้อความทำได้ค่อนข้างจะมีความถูกต้องเสมอ สามารถแปลคำทั่วไปได้ถูกต้อง และรองรับชื่อเฉพาะ เช่น ชื่อคน ชื่อสถานที่ เป็นต้น ยกเว้นในกรณีที่ผู้พูดพูดไม่ชัดเจน หรือคำที่ใช้มีเสียงกำกวมที่สื่อได้สองความหมาย

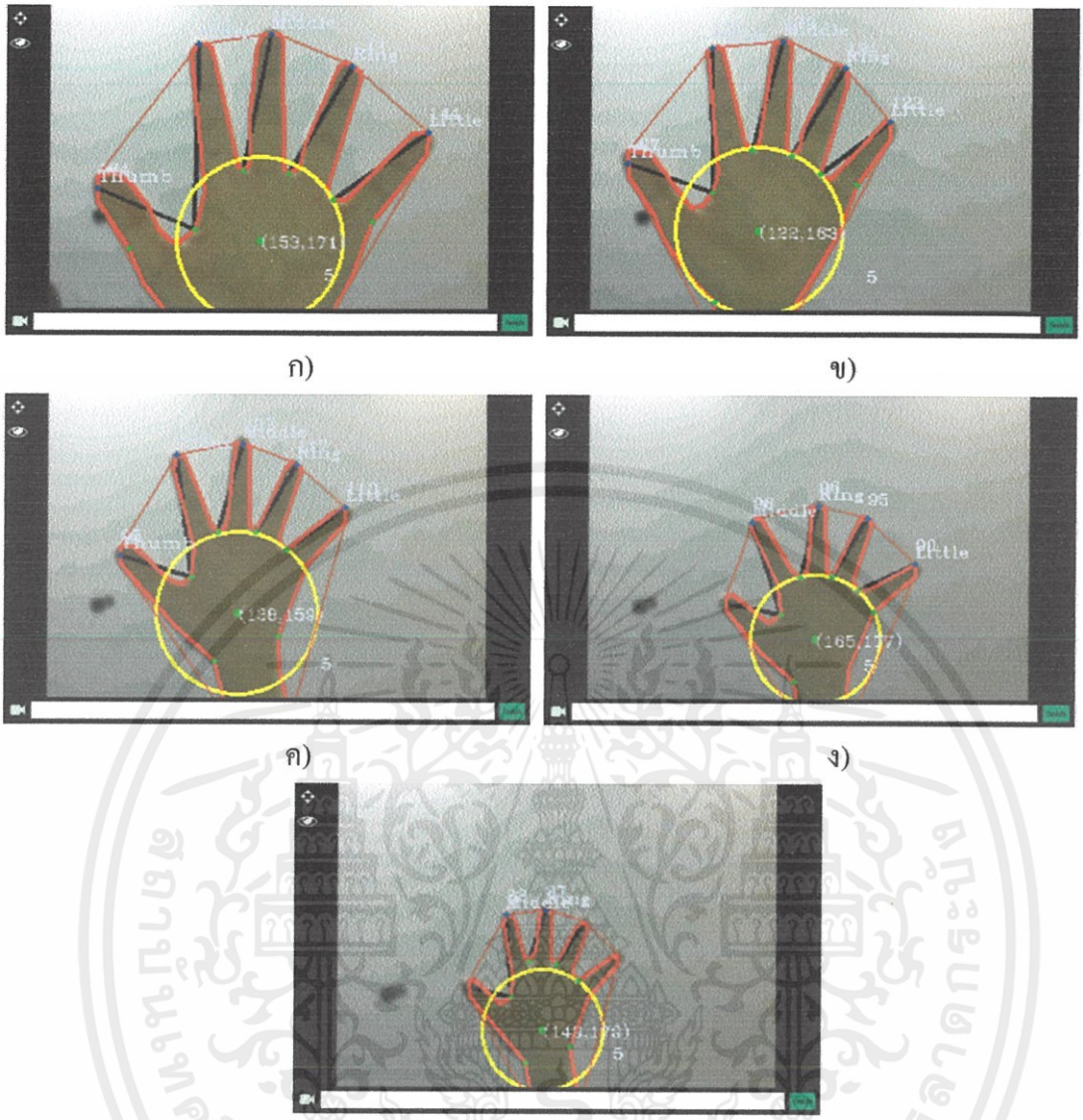
4.1.2 การใช้งานเพื่อการสื่อสารของคนหูหนวก

ส่วนการใช้งานเพื่อการสื่อสารของคนหูหนวก ได้แบ่งการทดลองออกเป็นการหาความถูกต้องของการเจอจุดปลายนิ้ว และการทดลองหาความถูกต้องในการระบุชื่อของนิ้วมือ

4.1.2.1 การทดลองหาความถูกต้องของการเจอจุดปลายนิ้วโดยมีปัจจัยคือระยะห่าง

การทดลองหาความถูกต้องของการเจอจุดปลายนิ้วส่วนนี้ มีปัจจัยคือระยะห่างระหว่างมือและอุปกรณ์เป็นตัวเปรียบเทียบ โดยแบ่งระยะห่างออกเป็น 20, 25, 30, 35 และ 40 เซนติเมตร โดยใช้ท่าทางของมือที่ตำแหน่งเดิมและมืออยู่ในลักษณะเดิมทุกครั้ง แล้วคำนวณค่าความถูกต้องโดยนำจำนวนนิ้วที่ตรวจเจอหารด้วยจำนวนนิ้วทั้งหมด

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ก)

ข)

ค)

ง)

จ)

รูป 4.3 ขนาดของมือที่ระยะต่างๆ

ก) 20 เซนติเมตร

ข) 25 เซนติเมตร

ค) 30 เซนติเมตร

ง) 35 เซนติเมตร

จ) 40 เซนติเมตร

ทำการทดลองนี้จะทำทั้งหมด 2 ครั้ง โดยในแต่ละครั้งจะคำนวณค่าความถูกต้องแล้วนำค่าความถูกต้องของทั้ง 2 ครั้งมาหาค่าเฉลี่ย ซึ่งการหาค่าความถูกต้องแต่ละครั้งหาได้จาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\left(\frac{\text{จำนวนนิ้วที่หาเจอ}}{\text{จำนวนนิ้วทั้งหมด}} + \frac{\text{จำนวนนิ้วที่ตรวจเจอแล้วถูกต้อง}}{\text{จำนวนนิ้วที่ตรวจเจอทั้งหมด}}\right) /2 \times 100 \qquad (4.2)$$

ตาราง 4.3 ความถูกต้องของการเจอจุดปลายนิ้วโดยมีปัจจัยคือระยะห่าง (ครั้งที่ 1)

ระยะห่าง	จำนวนนิ้วที่หาเจอ	นิ้วที่ตรวจหาเจอ					ค่าความถูกต้อง
		โป้ง	ชี้	กลาง	นาง	ก้อย	
20 cm	5	โป้ง	ชี้	กลาง	นาง	ก้อย	100
25 cm	5	โป้ง	ชี้	กลาง	นาง	ก้อย	100
30 cm	5	โป้ง	ชี้	กลาง	นาง	ก้อย	100
35 cm	2	-	กลาง	นาง	-	-	20
40 cm	2	-	กลาง	นาง	-	-	20

ตาราง 4.4 ความถูกต้องของการเจอจุดปลายนิ้วโดยมีปัจจัยคือระยะห่าง (ครั้งที่ 2)

ระยะห่าง	จำนวนนิ้วที่หาเจอ	นิ้วที่ตรวจหาเจอ					ค่าความถูกต้อง
		โป้ง	ชี้	กลาง	นาง	ก้อย	
20 cm	5	โป้ง	ชี้	กลาง	นาง	ก้อย	100
25 cm	5	โป้ง	ชี้	กลาง	นาง	ก้อย	100
30 cm	4	-	ชี้	กลาง	นาง	ก้อย	90
35 cm	3	-	กลาง	นาง	-	ก้อย	46.67
40 cm	2	-	กลาง	นาง	-	-	20

ตาราง 4.5 สรุปค่าเฉลี่ยของความถูกต้องจากผลการทดลองของตาราง 4.2 และ 4.3

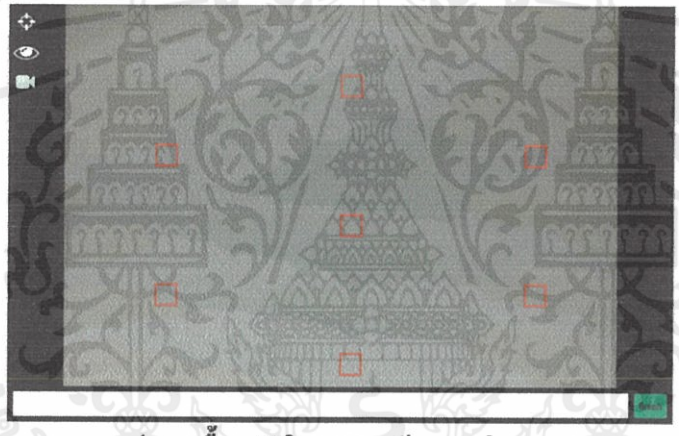
ระยะห่าง	จำนวนนิ้วที่หาเจอ	นิ้วที่ตรวจหาเจอ					เปอร์เซ็นต์ความถูกต้อง
		โป้ง	ชี้	กลาง	นาง	ก้อย	
20 cm	5	โป้ง	ชี้	กลาง	นาง	ก้อย	100
25 cm	5	โป้ง	ชี้	กลาง	นาง	ก้อย	100
30 cm	4.5	โป้ง, -	ชี้	กลาง	นาง	ก้อย	95
35 cm	2.5	-	กลาง	นาง	-	ก้อย, -	33.34
40 cm	2	-	กลาง	นาง	-	-	20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

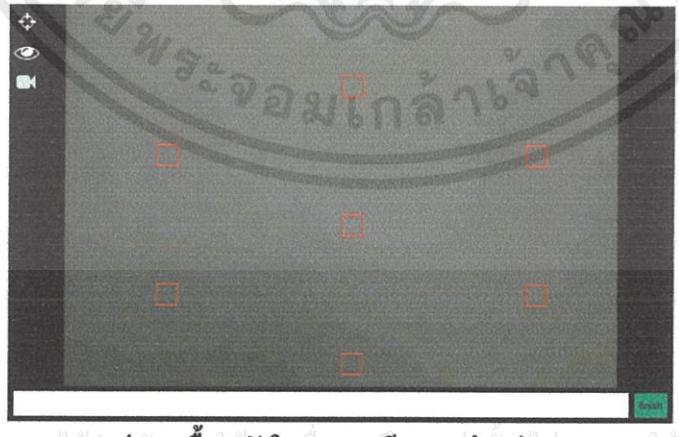
จากการทดลองหาความถูกต้องของการเจอจุดปลายนิ้ว โดยมีปัจจัยคือระยะห่างระหว่างมือและอุปกรณ์เป็นตัวเปรียบเทียบโดยมีปัจจัยคือแสงสว่าง สรุปผลได้ดังนี้ ระยะห่างระหว่างมือถึงตัวอุปกรณ์ที่ดีที่สุดคือ ที่ระยะ 20 – 25 เซนติเมตร โดยจะเริ่มมีความผิดเพี้ยนเมื่อระยะเกินกว่า 30 เซนติเมตรขึ้นไป

4.1.2.2 การทดลองหาความถูกต้องของการเจอจุดปลายนิ้วโดยมีปัจจัยคือแสงสว่าง

การทดลองหาความถูกต้องของการเจอจุดปลายนิ้วโดยมีปัจจัยคือแสงสว่าง จะทำการทดลองทั้งหมด 2 ครั้ง โดยในแต่ละครั้งจะแบ่งการทดลองออกเป็น 3 กรณี คือ กรณีมีแสงสว่างมาก กรณีมีแสงสว่างปานกลาง และกรณีมีแสงสว่างน้อย และจะคำนวณค่าความถูกต้องแล้วนำค่าความถูกต้องของทั้ง 2 ครั้งมาหาค่าเฉลี่ย ซึ่งการหาค่าความถูกต้องแต่ละครั้งหาได้จากสมการ 4.2

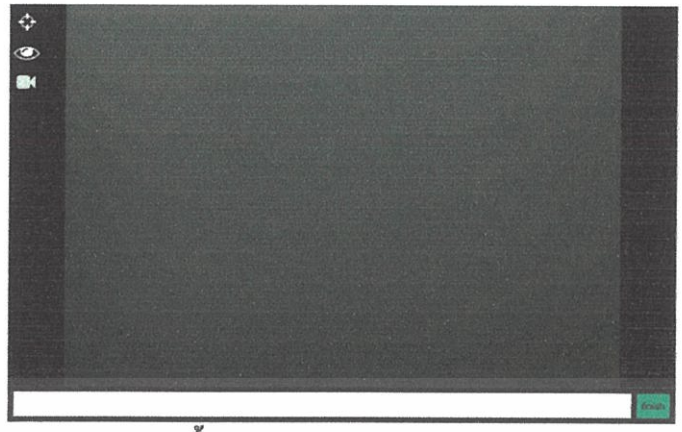


รูป 4.4 พื้นหลังในสภาวะมีแสงสว่างมาก



รูป 4.5 พื้นหลังในสภาวะมีแสงสว่างปานกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.6 พื้นหลังในสภาวะมีแสงสว่างน้อย

ตารางที่ 4.6 ความถูกต้องของการเจอจุดปลายนิ้วโดยมีปัจจัยคือแสงสว่าง (ครั้งที่ 1)

ความสว่าง	จำนวนนิ้วที่หาเจอ	นิ้วที่ตรวจหาเจอ					ค่าความถูกต้อง
		โป้ง	ชี้	กลาง	นาง	ก้อย	
มาก	5	โป้ง	ชี้	กลาง	นาง	ก้อย	100
ปานกลาง	3	-	กลาง	นาง	-	ก้อย	46.67
น้อย	3	-	กลาง	นาง	-	ก้อย	46.67

ตารางที่ 4.7 ความถูกต้องของการเจอจุดปลายนิ้วโดยมีปัจจัยคือแสงสว่าง (ครั้งที่ 2)

ความสว่าง	จำนวนนิ้วที่หาเจอ	นิ้วที่ตรวจหาเจอ					ค่าความถูกต้อง
		โป้ง	ชี้	กลาง	นาง	ก้อย	
มาก	5	โป้ง	ชี้	กลาง	นาง	ก้อย	100
ปานกลาง	3	-	กลาง	นาง	-	ก้อย	46.67
น้อย	2	-	กลาง	นาง	-	-	20

ตารางที่ 4.8 สรุปค่าเฉลี่ยของความถูกต้อง จากการทดลองของตารางที่ 4.6 และ 4.7

ความสว่าง	ค่าเฉลี่ยจำนวนนิ้วที่หาเจอ	นิ้วที่ตรวจหาเจอ					ความถูกต้อง
		โป้ง	ชี้	กลาง	นาง	ก้อย	
มาก	5	โป้ง	ชี้	กลาง	นาง	ก้อย	100
ปานกลาง	3	-	กลาง	นาง	-	ก้อย	46.67
น้อย	2.5	-	กลาง	นาง	-	- , ก้อย	33.34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2.3 การทดลองหาความถูกต้องของการแปลภาษามือ

การตรวจหาความถูกต้องของการแปลภาษามือ จะแบ่งออกเป็นการตรวจหาตัวเลข ตัวอักษร และคำศัพท์ ดังนี้

1) ความถูกต้องในการแปลตัวเลข ทดลองโดยนำตัวเลข 1-9 มาทำท่าภาษามือแล้วใช้แอพพลิเคชั่นแปล จากนั้นดูว่าผลลัพธ์ที่ได้ตรงตามความต้องการหรือไม่ โดยแต่ละตัวเลขจะทำการทดลองทั้งหมด 3 ครั้ง จากนั้น นำผลลัพธ์ที่ได้มาคำนวณเป็นเปอร์เซ็นต์ความถูกต้องโดยคิดจากจำนวนครั้งที่ถูกต้องหารด้วยจำนวนครั้งที่ทดลองทั้งหมด
2) ความถูกต้องในการแปลตัวอักษร ทำการทดลองเช่นเดียวกับข้อ 1) โดยเปลี่ยนจากตัวเลขเป็นตัวอักษรจำนวน 9 ตัวอักษร แล้วนำผลลัพธ์มาคำนวณหาเปอร์เซ็นต์ความถูกต้อง
3) ความถูกต้องในการแปลคำศัพท์ ทำการทดลองเช่นเดียวกับข้อ 1) โดยเปลี่ยนจากตัวเลขเป็นคำศัพท์จำนวน 5 คำศัพท์ แล้วนำผลลัพธ์มากคำนวณหาเปอร์เซ็นต์ความถูกต้อง

ตาราง 4.9 ความถูกต้องของการแปลภาษามือ (ตัวเลข)

ตัวเลข	ผลลัพธ์การแปล			เปอร์เซ็นต์ความถูกต้อง	หมายเหตุ
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3		
1	1	1	1	100	-
2	ก	2	2	66	เจอในคำใกล้เคียง
3	3	3	ว	66	เจอในคำใกล้เคียง
4	4	4	4	100	-
5	5	5	5	100	-
6	6	6	ย	66	เจอในคำใกล้เคียง
7	3	3	7	33	เจอในคำใกล้เคียง
8	ว	8	8	66	เจอในคำใกล้เคียง
9	ฟ	9	9	66	เจอในคำใกล้เคียง
ค่าเฉลี่ยเปอร์เซ็นต์ความถูกต้อง				73.7	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.10 ความถูกต้องของการแปลภาษามือ (ตัวอักษร)

ตัวอักษร	ผลลัพธ์การแปล			เปอร์เซ็นต์ความถูกต้อง	หมายเหตุ
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3		
ก	ก	ก	2	66	เจอในคำใกล้เคียง
ข	ข	ข	ฬ	66	เจอในคำใกล้เคียง
ฆ	ฆ	ฆ	ฆ	100	
จ	ค	ศ	จ	33	เจอในคำใกล้เคียง
ช	ช	ธ	ช	66	เจอในคำใกล้เคียง
ฎ	ฒ	ฎ	ฑ	33	เจอในคำใกล้เคียง
ฐ	ฐ	ฐ	ฐ	100	
ถ	ถ	1	ถ	66	เจอในคำใกล้เคียง
ล	ล	ล	ก	66	เจอในคำใกล้เคียง
ค่าเฉลี่ยเปอร์เซ็นต์ความถูกต้อง				66.2	

ตาราง 4.11 ความถูกต้องของการแปลภาษามือ (คำศัพท์)

คำศัพท์	ผลลัพธ์การแปล			เปอร์เซ็นต์ความถูกต้อง	หมายเหตุ
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3		
ผู้หญิง	ผู้หญิง	ช	ช	33	เจอในคำใกล้เคียง
อา	อา	อา	อา	100	
แฟน	2	2	ฆ	0	เจอในคำใกล้เคียง
แม่	แม่	ผอม	ผอม	33	เจอในคำใกล้เคียง
รัก	รัก	รัก	รัก	100	
ค่าเฉลี่ยเปอร์เซ็นต์ความถูกต้อง				53.2	

จากการทดลอง จะพบว่า การแปลภาษามือที่เป็นตัวเลขจะมีเปอร์เซ็นต์ความถูกต้องเยอะที่สุด โดยมีเปอร์เซ็นต์ความถูกต้องถึง 73.7 เปอร์เซ็นต์ เนื่องจากท่าทางในการทำตัวเลข เป็นท่าทางที่สามารถทำได้ง่ายและไม่ซับซ้อน สามารถแปลได้ง่าย

ส่วนความถูกต้องในการแปลตัวอักษร มีเปอร์เซ็นต์ความถูกต้องเท่ากับ 66.2 เปอร์เซ็นต์ เนื่องจากท่าทางของการทำตัวอักษรนั้นแม้จะไม่ซับซ้อนมาก แต่มีตัวอักษรจำนวนหนึ่งที่มีท่าทางคล้ายกัน ทำให้ผลลัพธ์การแปลอาจจะไม่ถูกต้องในบางครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้นำไปใช้

ส่วนความถูกต้องของการแปลคำศัพท์ มีเปอร์เซ็นต์ความถูกต้องเท่ากับ 53.2 เปอร์เซ็นต์ โดยผลลัพธ์จะขึ้นอยู่กับความซับซ้อนของท่าทางคำศัพท์ หากมีความซับซ้อนมาก อาจทำให้ผลลัพธ์ที่ได้ออกมาไม่ตรง เช่นเดียวกับจำนวนคำศัพท์ในฐานข้อมูล หากจำนวนคำศัพท์ในฐานข้อมูลมีจำนวนมาก ทำให้มีโอกาสที่คำศัพท์จะมีโอกาสคล้ายกันเพิ่มขึ้น (เนื่องจากคุณลักษณะในการนำมาใช้แยกแยะคำศัพท์ยังน้อย ทำให้มีโอกาสเกิดกรณีซ้ำกันเยอะ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุปและข้อเสนอแนะ

5.1 บทสรุป

แอพพลิเคชั่นแปลภาษามือบนระบบปฏิบัติการแอนดรอยด์ เป็นแอพพลิเคชั่นที่ออกแบบมาสำหรับเป็นเหมือนเครื่องมือแปลภาษาส่วนตัวของคนหูหนวก เพื่อให้คนหูหนวกสามารถสื่อสารกับคนทั่วไปได้สะดวกขึ้น โดยแอพพลิเคชั่นจะรองรับเฉพาะแอนดรอยด์เวอร์ชั่นตั้งแต่ 4.1 ขึ้นไป และต้องมีกล้องหน้าเพื่อใช้ในการตรวจจับท่าทางของภาษามือ รวมถึงกล้องที่ใช้ต้องรองรับขนาดของภาพ 320 x 240 pixel เป็นอย่างน้อย

5.2 ปัญหาและอุปสรรคต่างๆ

ในการพัฒนาแอพพลิเคชั่นแปลภาษามือบนระบบปฏิบัติการแอนดรอยด์ มีอุปสรรคและปัญหาดังนี้

5.2.1 ฐานข้อมูล

ในการจัดเก็บคำและแปลข้อมูลให้เป็นคำ ผู้พัฒนาได้พบปัญหาในการออกแบบฐานข้อมูล เนื่องจากการทำภาษามือแต่ละท่านั้น จำเป็นต้องใช้ทั้งท่าทาง และหน้าตา รวมถึงตำแหน่งของมือในการจะบอกถึงความหมายของคำได้ ฐานข้อมูลจึงต้องออกแบบเพื่อรองรับตัวแปรเหล่านี้ให้ได้ทั้งหมด

นอกจากนี้ เรื่องความเร็วในการทำท่าทาง ก็มีผลต่อการแปลเช่นกัน เพราะหากท่าทางที่ได้รับเข้ามา มีความเร็วในการทำท่าทางมากกว่าตอนจัดเก็บลงฐานข้อมูล ทำให้ลำดับท่าที่ตรวจจับได้จะไม่ตรงกับฐานข้อมูลไม่ตรงกัน ซึ่งส่วนนี้ ผู้พัฒนาจึงต้องคิดวิธีการตรวจสอบความแตกต่างระหว่างท่าทาง เพื่อให้ฐานข้อมูลเก็บข้อมูลเฉพาะท่าหลักๆ เท่านั้น

5.2.2 ภาษามือ

เนื่องจากภาษามือของไทย ยังไม่มีการเรียบเรียงหรือแหล่งเรียนรู้ที่มากนัก ทำให้ผู้พัฒนาศึกษาเกี่ยวกับภาษามือได้ยาก และอาจเกิดการประมวลผลลักษณะของภาษามือผิดได้

ภาษามือ เป็นภาษาที่ไวยากรณ์แตกต่างจากภาษาไทยทั่วไป ซึ่งการจะทำความเข้าใจได้นั้น จำเป็นต้องใช้ความคุ้นเคยในการใช้ภาษามือพอสมควร และคำบางคำมีได้หลายท่าทาง การจะทำให้แอพพลิเคชั่นสามารถแปลภาษามือได้นั้น จึงเป็นเรื่องที่ท้าทายสำหรับผู้พัฒนาเป็นอย่างมาก ทั้งเรื่องท่าทางที่การประมวลภาพสามารถจับได้ยาก และความหลากหลายของภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.3 ข้อจำกัดการประมวลผลภาพ

เนื่องเป็นแอพพลิเคชั่นสำหรับใช้งานบนโทรศัพท์ประเภทสมาร์ทโฟน ทำให้มีข้อจำกัดทางด้านทางด้านประสิทธิภาพ เช่น ความเร็วในการใช้งาน การประมวลผลรูปภาพที่ช้าทำเกิดข้อผิดพลาดในการแปลภาษามือ และการหยุดทำงานของแอพพลิเคชั่น เป็นต้น

เซนเซอร์(กล้องโทรศัพท์)ที่นำมาใช้ในการแปลภาษามือ ไม่มีเซนเซอร์ตรวจจับความลึกความตื้นของเฟรม ทำให้ไม่สามารถระบุหรือแปลภาษามือที่มีท่าทางเคลื่อนที่ไปด้านหน้าและด้านหลัง เช่นคำว่า “รถเมล์” เป็นต้น ฟังก์ชั่นที่ต้องการใช้งานในไลบารี่ของ OpenCV ที่ใช้สำหรับพัฒนาบนแอนดรอยด์ไม่รองรับ เช่น Inscribed Circle เป็นต้น ทำให้ผู้พัฒนาต้องทำการเขียนฟังก์ชั่นขึ้นมาเอง หรือหาฟังก์ชั่นอื่นมาใช้ทดแทนข้อจำกัดการประมวลผลภาพ

5.3 แนวทางในการพัฒนาต่อ

แนวคิดของการพัฒนาแอพพลิเคชั่นแปลภาษามือนี้ เป็นการพัฒนาแอพพลิเคชั่นโดยผู้ใช้สามารถเพิ่มคำเข้าไปในระบบได้ โดยไม่ขึ้นอยู่กับภาษามือประเทศใด ซึ่งหากประสบความสำเร็จจะทำให้เกิดกลุ่มของผู้บกพร่องทางการได้ยินที่มีการแชร์ฐานข้อมูลจากแอพพลิเคชั่น และมีการใช้ภาษามือที่ตรงกันมากขึ้น จนสามารถรวมภาษามือที่แตกต่างไปตามภาคท้องถิ่นต่างๆ ให้เป็นภาษาเดียวกันได้

และการที่แอพพลิเคชั่นนี้ไม่ขึ้นอยู่กับภาษามือประเทศใด ในอนาคตจึงสามารถนำไปต่อยอดให้แอพพลิเคชั่นสามารถรองรับภาษามือหลายประเทศได้

นอกจากนี้ เมื่อแอพพลิเคชั่นเสร็จสมบูรณ์ จะสามารถนำไปปรับใช้กับงานด้านอื่นๆ ต่อไปได้ เช่น นำไปติดตั้งตามที่ทำการต่างๆ เพื่อให้สามารถรองรับการใช้บริการของผู้บกพร่องทางการได้ยินได้ หรืออาจจะพัฒนาเป็นอุปกรณ์ช่วยในการเรียนรู้เพื่อให้ผู้บกพร่องทางการได้ยินสามารถเรียนรู้ได้เร็วขึ้น หรือสามารถนำมาพัฒนาเป็นฟีเจอร์เสริมในแอพพลิเคชั่นสื่อสารทางมือถือ เช่น Line, WeChat ได้เช่นกัน

ทั้งนี้ ผู้พัฒนามีความคิดที่จะทำให้แอพพลิเคชั่นนี้ สามารถนำไปพัฒนาต่อโดยบุคคลอื่นๆ ได้ โดยการพยายามแบ่งฟังก์ชันภายในแอพพลิเคชั่นให้ออกเป็นโมดูลตามที่ได้กล่าวไว้แล้วในบทที่ 3 เพื่อให้ผู้ที่ต้องการพัฒนาต่อสามารถศึกษาระบบให้เข้าใจและพัฒนาได้ง่ายขึ้น อีกทั้งยังให้ผู้ที่ต้องการนำระบบบางส่วนไปพัฒนา สามารถที่จะนำโมดูลเฉพาะส่วนที่ต้องการนำไปพัฒนาเพิ่มเติมต่อได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

Hui-Shyong Yeo, Byung-Gook Lee, Hyotaek Lim. 2013. "Hand tracking and gesture recognition system for human-computer interaction using low-cost hardware," **Springer US.** 74(8) : pp 2687-2715.

University of California, Santa Barbara. 2014. **Color-Based Hand Gesture Recognition on Android.** [Online].
Available : http://eaglesky.github.io/blog/2014/03/27/color-based-hand-gesture-recognition-on-android/.

Chen, Qing. 2008. **Real-time vision-based hand tracking and gesture recognition.** [Online].
Available : http://www.ruor.uottawa.ca/handle/10393/29518.

Amiraj Dhawan, Vipul Honrao. 2013. "Implementation of Hand Detection based Techniques for Human Computer Interaction," **International Journal of Computer Applications.** 72(17)

Itseez. 2013. **OpenCV Tutorials.** [Online].
Avaliable : http://docs.opencv.org/doc/tutorials/tutorials.html.

James Elsey. 2011. **Android; How to implement voice recognition, a nice easy tutorial.** [Online].
Avaliable : http://www.jameselsey.co.uk/blogs/techblog/android-how-to-implement-voice-recognition-a-nice-easy-tutorial.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้