แอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ

ANDROID APPLICATION FOR AUTOMATED WHEELCHAIR CONTROLLING

ศุภากร อริยมงคลชัย

สิทธิกร อริยมงคลชัย

ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ANDROID APPLICATION FOR AUTOMATED WHEELCHAIR CONTROLLING

SUPARKORN ARIYAMONGKOLCHAI

SITTIKORN ARIYAMONGKOLCHAI

A SPECIAL PROBLEM SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR

THE DEGREE OF BACHELOR OF SCIENCE (COMPUTER SCIENCE)

DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2017

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	แอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ ANDROID APPLICATION FOR AUTOMATED WHEELCHAIR CONTROLLING
ชื่อนักศึกษา	นายศุภากร อริยมงคลชัย รหัสนักศึกษา 57050336 นายสิทธิกร อริยมงคลชัย รหัสนักศึกษา 57050338
ปริญญา	วิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชา	วิทยาการคอมพิวเตอร์
ปีการศึกษา	2560
อาจารย์ที่ปรึกษา	ผศ.ดร.วรางคณา กิ้มปาน

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์) ประจำปีการศึกษา 2560

คณะกรรมการสอบ	ลายมือชื่อ
อ.ศังกรศรัณย์ ล่องชูผล ประธานกรรมการ	
ผศ.วิสันต์ ตั้งวงษ์เจริญ กรรมการ	
ผศ.ดร.วรางคณา กิ้มปาน อาจารย์ที่ปรึกษา	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
ลิขสิทธิ์ของคณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

หัวข้อปัญหาพิเศษ แอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ
ชื่อนักศึกษา นายศุภากร อริยมงคลชัย รหัสนักศึกษา 57050336
นายสิทธิกร อริยมงคลชัย รหัสนักศึกษา 57050338
ปริญญา วิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชา วิทยาการคอมพิวเตอร์
ปีการศึกษา 2560
อาจารย์ที่ปรึกษา ผศ.ดร.วรางคณา กิ้มปาน

บทคัดย่อ

บทความนี้นำเสนอแอนดรอยด์แอปพลิเคชันสำหรับควบคุมรถนั่งอัตโนมัติ เหมาะสำหรับผู้ที่พิการทางขา ซึ่งไม่สะดวกและไม่สามารถเคลื่อนไหวที่ขาได้นอกจากนั้นผู้สูงอายุทั่วไปก็สามารถใช้งานแอปพลิเคชันนี้ได้ ซึ่งแอปพลิเคชันที่สร้างขึ้นนี้จะใช้ควบคุมรถนั่งอัตโนมัติ สำหรับการใช้งานในโรงพยาบาล ใช้หลักการประมวลผลภาพ โดยใช้ป้ายบอกทางภายในโรงพยาบาลในการประมวลผลภาพเป็นแนวในการนำทางไปยังจุดมุ่งหมายของสถานที่ต่าง ๆ ที่แสดงในโรงพยาบาลได้โดยอัตโนมัติ อีกทั้งยังมีระบบที่ควบคุมด้วยมือ ที่ให้ผู้ใช้สามารถใช้ควบคุมรถนั่งอัตโนมัติด้วยตนเอง โดยทั้ง 2 การควบคุมจะมีเซ็นเซอร์ตรวจจับความปลอดภัย 2 รูปแบบ ซึ่งผู้ใช้รถนั่งเกิดใช้งานรถนั่งล้มจะมีการแจ้งเตือนไปยังผู้ดูแล และในขณะที่ผู้ใช้ควบคุมรถนั่งอัตโนมัติเคลื่อนที่จะมีเซ็นเซอร์ตรวจจับสิ่งกีดขวาง เมื่อพบสิ่งกีดขวางรถนั่งอัตโนมัติจะไม่สามารถเคลื่อนที่ไปยังทิศทางที่มีสิ่งกีดขวางได้ จึงทำให้รถนั่งมีความปลอดภัยและมีทิศทางถูกต้องในการนำทางไปยังจุดมุ่งหมายของผู้ใช้งาน

คำสำคัญ : รถนั่งอัตโนมัติ หลักการประมวลผลภาพ เซ็นเซอร์ แอนดรอยด์แอปพลิเคชัน โรงพยาบาล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	ANDROID APPLICATION FOR AUTOMATED WHEELCHAIR CONTROLLING
Students	Mr. Suparkorn Ariyamongkolchai Student ID 57050336
	Mr. Sittikorn Ariyamongkolchai Student ID 57050338
Degree	Bachelor of Science (Computer Science)
Department	Computer Science
Faculty	Science
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)
Academic Year	2017
Advisor	Asst.Prof.Dr.Warangkhana Kimpan

ABSTRACT

This special problem proposed the Android application for controlling automatic wheelchair. This application is convenient for legs disabled person, people who is not convenient about using legs, and elderly person. The application was developed to control the automatic wheelchair which will be used in hospitals. There are 2 modes to control the system: manual and automatic. In manual mode, the user can use the arrows which are left, right, forward and back to move the wheelchair to the destination. The user can also control the wheelchair in automatic mode by using the principle image processing via web camera and display in the Android application. The image processing will lead the wheelchair automatically by reading the guide posts in the hospital and it automatically moves the wheelchair to the destination. Moreover, there are 2 types of sensors to detect the safety. In case of wheelchair is falling, the application will send the notification to other persons. In case of detecting the barrier, the wheelchair will be automatically stopped.

Keywords : Automatic wheelchair, Image processing, Sensor, Android application, Hospital

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ในการจัดทำปัญหาพิเศษแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ ให้สำเร็จลุล่วงไปได้ด้วยดี คณะผู้จัดทำขอขอบพระคุณโครงการแข่งขันพัฒนาโปรแกรมคอมพิวเตอร์แห่งประเทศไทย ครั้งที่ 20 : NSC 2018 ในการให้ทุนสนับสนุนเพื่อนำไปพัฒนาต่อยอดให้โปรแกรมมีความสมบูรณ์และพร้อมใช้งานมากยิ่งขึ้น ขอขอบพระคุณ ผศ.ดร.วรางคณา กิ้มปาน เป็นอย่างสูง ซึ่งเป็นอาจารย์ที่ปรึกษาที่ให้คำแนะนำในการทำปัญหาพิเศษครั้งนี้ ให้สามารถแก้ไขปัญหาต่าง ๆ อีกทั้งยังช่วยตรวจสอบความถูกต้องของโครงการนี้ให้เป็นระเบียบมากยิ่งขึ้น

ขอขอบพระคุณ อ.ศังกรศรัณย์ ล่องชูผล ประธานกรรมการ และ ผศ.วิสันต์ ตั้งวงษ์เจริญ กรรมการในการสอบปัญหาพิเศษ ที่ให้คำแนะนำในการแก้ไขข้อบกพร่อง เพื่อให้ปัญหาพิเศษนี้สมบูรณ์ยิ่งขึ้น รวมถึงอาจารย์ในสาขาวิชาวิทยาการคอมพิวเตอร์ทุกท่าน ที่ได้สั่งสอนมาตลอดระยะเวลา 4 ปี

สุดท้ายนี้ ขอขอบพระคุณบิดามารดา และครอบครัว ที่ค่อยสนับสนุนและให้กำลังใจตลอดการดำเนินงานโครงการจนสำเร็จลุล่วงด้วยดี

สิทธิกร อริยมงคลชัย

ศุภากร อริยมงคลชัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย..... ก
บทคัดย่อภาษาอังกฤษ..... ข
กิตติกรรมประกาศ..... ค
สารบัญ..... ง
สารบัญตาราง..... ช
สารบัญรูป..... ซ

บทที่ 1 บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา..... 1
1.2 วัตถุประสงค์..... 2
1.3 ขอบเขตของปัญหาพิเศษ..... 2
1.4 ประโยชน์ที่คาดว่าจะได้รับ..... 3
1.5 ขั้นตอนการดำเนินงาน..... 3
1.6 อุปกรณ์ที่ใช้..... 3

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

2.1 โมดูลสื่อสารไร้สายบลูทูธ (Bluetooth Module)..... 5
2.2 บอร์ดราสเบอร์รี่พาย (Raspberry Pi Board)..... 6
2.3 บอร์ดขับมอเตอร์ไฟกระแสตรง (MDD10A)..... 8
2.4 เซนเซอร์วัดระยะ (JSN-SR04T)..... 10
2.5 รถเข็นวีลแชร์ (Wheelchair)..... 11
2.6 ภาษา Python..... 12
2.7 โปรแกรม Android Studio..... 13
2.8 พิการทางการเคลื่อนไหว..... 14
2.9 บอร์ดอาดูโน่ (Arduino Board)..... 16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.10 Firebase 18
2.11 SIFT (Scale Invariant Feature Transform) 20
2.12 เซนเซอร์วัดความเร่งบนแอนดรอยด์ (Android Sensor Accelerometer) 29

บทที่ 3 การวิเคราะห์และออกแบบระบบ

3.1 สถาปัตยกรรมของระบบ 34
3.2 การออกแบบระบบ 36
3.2.1 แผนภาพยูสเคส (Use Case Diagram) 36
3.2.2 แผนภาพแอคทิวิตี้ไดอะแกรม (Activity Diagram) 40
3.2.3 ขั้นตอนการทำงานของระบบ (Sequence Diagram) 44
3.2.4 แผนผังการทำงานของระบบ (Flow Chart) 51
3.3 ส่วนติดต่อประสานกับผู้ใช้ (User Interface) 53

บทที่ 4 การทดสอบแอปพลิเคชันและผลการทดสอบแอปพลิเคชัน

4.1 การทดสอบแอปพลิเคชัน SmartCar 58
4.1.1 อุปกรณ์ที่ใช้ทดสอบแอปพลิเคชัน 58
4.1.2 การทดสอบแอปพลิเคชัน 61
4.2 ผลการทดสอบแอปพลิเคชัน 71
4.2.1 การทดสอบการทำงานในการตรวจล้ม 71
4.2.2 การทดสอบความเร็วโดยเปลี่ยนแปลงน้ำหนัก 74
4.2.3 การทดสอบการทำงานในการตรวจสิ่งกีดขวาง 75
4.2.4 การทดสอบการติดตั้งป้าย 76
4.2.5 การตรวจสอบระยะในการประมวลภาพได้ 78
4.2.6 การทดสอบกับผู้ใช้จริง 79
4.3 ตัวอย่างโค้ดโปรแกรม 81
4.3.1 โค้ดในฝั่งของ Raspberry pi 81
4.3.2 โค้ดในฝั่งของ application 105

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

4.3.3 โค้ดในฝั่งของ Arduino..... 105
4.3.4 เป็นอัลกอริทึมในการประมวลผลภาพ..... 106

บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน..... 110
5.2 ปัญหาต่าง ๆ ที่พบระหว่างการพัฒนาแอปพลิเคชัน..... 110
5.3 ข้อจำกัดของระบบ..... 111
5.4 ข้อเสนอแนะ..... 111

เอกสารอ้างอิง..... 112
ภาคผนวก..... 113
ภาคผนวก ก คู่มือการใช้อย่างละเอียด..... 114
ภาคผนวก ข ผลงานที่ได้รับรางวัล..... 121

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
3.1 อธิบาย Use Case ควบคุมนั่ง	37
3.2 อธิบาย Use Case ควบคุมนั่งแบบอัตโนมัติ	37
3.3 อธิบาย Use Case โทรออกฉุกเฉินด้วยตนเอง	38
3.4 อธิบาย Use Case ตรวจการล้มจากรถนั่ง	38
3.5 อธิบาย Use Case ตรวจกันชนกับวัตถุด้านหน้าและด้านหลัง	39
3.6 อธิบาย Use Case ตั้งค่าความเร็วในการเคลื่อนที่โหมด Manual	39
3.7 อธิบาย Use Case ตั้งค่าเบอร์โทรผู้ดูแล	40
4.1 ทิศทางการล้มของรถนั่ง และค่าของแกนความเร่งที่เปลี่ยนไป	74
4.2 น้ำหนักส่งผลต่อความเร็ว	74
4.3 ระยะตรวจสิ่งกีดขวาง โดยระดับความเร็ว Standard	75
4.4 ระยะตรวจสิ่งกีดขวาง โดยระดับความเร็ว Medium	76
4.5 ระยะตรวจสิ่งกีดขวาง โดยระดับความเร็ว High	76
4.6 องศาในการติดตั้งป้ายประมวลผลภาพ	77
4.7 ระยะที่กล้องสามารถประมวลผลภาพ	78
4.8 ระยะห่างระหว่างป้ายกับตัวกล้องในการประมวลผล	79

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 Bluetooth Module	5
2.2 บอร์ด Raspberry Pi	6
2.3 ส่วนประกอบของ Raspberry Pi	7
2.4 พอร์ต Camera Serial Interface	7
2.5 บอร์ดขับมอเตอร์ไฟฟ้ากระแสตรง (MDD10A)	8
2.6 ส่วนประกอบของบอร์ด MDD10A	9
2.7 เซนเซอร์วัดระยะ (JSN-SR04T)	10
2.8 รถเข็นวีลแชร์ (Wheelchair)	12
2.9 โปรแกรม Android Studio	14
2.10 บอร์ด Arduino	17
2.11 ส่วนประกอบของบอร์ด Arduino	18
2.12 การหาปริภูมิค่าในมิติขนาดและระยะทางเสร็จจะได้ภาพที่มีความคมชัดต่ำ	21
2.13 การหา Difference of Gaussian	22
2.14 ผลลัพธ์จากการทำ Difference of Gaussian ในแต่ละขั้นของ Octave	23
2.15 จุด X ที่เป็นจุดอ้างอิงเปรียบเทียบกับจุดรอบข้าง 26 จุด	23
2.16 จุดสูงสุดหรือจุดต่ำสุดที่เป็น pixel	24
2.17 ผลลัพธ์จากการทำ Locate maxima/minima กับ 4 DoG images	25
2.18 ผลลัพธ์ที่ได้หลังจากทำการหา keypoint จากการทดสอบ Edge และ Contrast ที่เหมาะสม	26
2.19 การขนาดและทิศทางของ keypoint	27
2.20 ผลลัพธ์ของการหาตำแหน่งและทิศทางของการวางตัวของกล้องจากหนังสือที่ต้องการ (เส้นขอบสีน้ำเงิน)	29
2.21 บอร์ดที่อยู่ในตัวโทรศัพท์ Android Sensor Accelerometer	30
2.22 ทิศทาง 3 ทิศ ในการวัดความเร่ง สำหรับแกน XYZ	30
2.23 การวัดความเร่งในการเอียง	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
2.24 การเอียงไปทางแกน X และ Y ที่มีผลต่อ แกน Z	32
2.25 ค่าแกน XYZ วางเครื่องโทรศัพท์ในแนวหงาย หรือ คว่ำ	33
2.26 ค่าแกน XYZ วางเครื่องโทรศัพท์ในตะแคงซ้าย หรือ ขวา	33
2.27 ค่าแกน XYZ วางเครื่องโทรศัพท์ในแนวตั้งขึ้น หรือ ทิ่มหัวลง	33
3.1 สถาปัตยกรรมของระบบ	34
3.2 Use Case Diagram	36
3.3 แผนภาพ Activity Diagram การควบคุมการเคลื่อนที่รถนั่ง	38
3.4 แผนภาพ Activity Diagram การควบคุมเคลื่อนที่แบบอัตโนมัติ	38
3.5 แผนภาพ Activity Diagram การตรวจล้ม และโทรออกฉุกเฉิน	39
3.6 แผนผังการทำงานของรถนั่งควบคุมด้วยแอปพลิเคชันบนแอนดรอยด์	39
3.7 Sequence Diagram ของฟังก์ชันควบคุมการเคลื่อนที่รถนั่ง	40
3.8 Sequence Diagram ของฟังก์ชันตรวจจับการล้ม	48
3.9 Sequence Diagram ของฟังก์ชันควบคุมการเคลื่อนรถเข็นแบบอัตโนมัติ	50
3.10 Sequence Diagram ของฟังก์ชันกล้องแสดงผลภาพ	51
3.11 หน้าจอเข้าสู่ระบบ (HOME)	52
3.12 หน้าจอช่วยเหลือแนะนำการใช้งาน	53
3.13 หน้าจอเลือกระบบการทำงาน	54
3.14 หน้าจอการทำงาน AUTOMATIC	55
3.15 หน้าจอการทำงาน MANUAL	56
3.16 หน้าจอการทำงาน SETTING	57
4.1 วีลแชร์ที่ได้นำมาต่อยอด	58
4.2 ลักษณะ Camera Serial Interface	59
4.3 ลักษณะ JSN-SR04T Waterproof Ultrasonic Module	59
4.4 Smartphone Samsung galaxy S8	60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.5 โครงสร้างหน้าจอแอปพลิเคชัน SmartCar	61
4.6 หน้าจอแรกที่ให้ตั้งค่าความเร็วโดยอ้างอิงจากน้ำหนัก	62
4.7 หน้าจอเปิดบลูทูธ	62
4.8 หน้าจอหลังกดปุ่ม Paired Devices	63
4.9 หน้าจอโหมด Manual	63
4.10 รถนั่งเลี้ยวซ้าย	64
4.11 รถนั่งเลี้ยวขวา	64
4.12 รถนั่งเดินหน้า	65
4.13 รถนั่งถอยหลัง	65
4.14 ขณะที่ใช้โหมด Manual เพื่อใช้กล้องประมวลผลภาพ	66
4.15 เข้าสู่หน้าโหมด Auto	66
4.16 แสดงรูปกำลังเคลื่อนที่โหมด Auto	67
4.17 สถานะในการเคลื่อนที่ถึงตำแหน่งที่ต้องการเพื่อหยุด	67
4.18 แผนผังในการเคลื่อนที่โหมด Auto	68
4.19 แถบเครื่องมือ	68
4.20 หน้าจอปุ่ม setting	69
4.21 หน้าจอปุ่ม About	69
4.22 ทดสอบการตรวจล้ม	70
4.23 ทดสอบหยุดรถในขณะที่มีสิ่งกีดขวาง	70
4.24 ผลการรับค่าเฉลี่ยการทรงตัวแบบปกติ	71
4.25 ค่าที่ได้จากการจำลองการล้มกรณีรถนั่งล้มไปยังทิศทางด้านหน้า	72
4.26 ค่าที่ได้จากการจำลองการล้มกรณีรถนั่งล้มไปยังทิศทางด้านหลัง	72
4.27 ค่าที่ได้จากการจำลองการล้มกรณีรถนั่งล้มไปยังทิศทางด้านซ้าย	73
4.28 ค่าที่ได้จากการจำลองการล้มกรณีรถนั่งล้มไปยังทิศทางด้านขวา	73
4.29 ผลการทดลองการหยุดเมื่อเจอสิ่งกีดขวาง	75
4.30 ผลการทดลองติดตั้งองศาป้าย	76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.31 ผลการทดลองระยะการประมวลผลของกล้อง	77
4.32 ระยะห่างระหว่างป้ายและกล้องที่ประมวลผลภาพได้	78
4.33 ทดสอบการใช้งานรถนั่งอัตโนมัติกับผู้ใช้งานรายที่หนึ่ง	79
4.34 ทดสอบการใช้งานรถนั่งอัตโนมัติกับผู้ใช้งานรายที่สอง	80
ก.1 ไอคอนของแอปพลิเคชันบนสมาร์ทโฟน	114
ก.2 หน้าจอแรกให้กรอกน้ำหนักผู้ใช้งาน	114
ก.3 หน้าจออนุญาตเปิดบลูทูธ	115
ก.4 หน้าจอเชื่อมต่อบลูทูธ	115
ก.5 หน้าจอโหมด Manual	116
ก.6 หน้าจอแถบเมนู	116
ก.7 หน้าจอ Setting	117
ก.8 หน้าจอ Help	117
ก.9 การประมวลผลภาพเพื่อให้ปุ่ม Auto ใช้งานได้	118
ก.10 หน้าจอโหมด Auto	118
ก.11 หน้าจอกำลังเคลื่อนที่ในโหมด Auto	119
ก.12 สถานะบงบอกว่าถึงตำแหน่งที่ต้องการจะไปในโหมด Auto	119
ก.13 แสดงการโทรออกไปยังผู้ดูแล	120
ก.14 เมื่อมีสิ่งกีดขวาง	120
ข.1 ประกาศนียบัตรที่ได้รับ	121

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เทคโนโลยีมีบทบาทต่อการดำเนินชีวิตเป็นอย่างมาก ปัจจุบันจึงมีการให้ความสำคัญแก่ผู้พิการและผู้สูงอายุมากขึ้น การพัฒนาเทคโนโลยีเพื่อช่วยอำนวยความสะดวกให้กับผู้ที่ต้องการความสะดวกสบายในโรงพยาบาลจึงเป็นสิ่งสำคัญ จากการศึกษาพบว่าเทคโนโลยีสามารถนำมาประยุกต์ใช้เพื่อช่วยเหลือ เพิ่มความปลอดภัย และอำนวยความสะดวกในการเคลื่อนที่ไปยังตำแหน่งต่าง ๆ ในโรงพยาบาล

การนำเทคโนโลยีมาใช้เพื่อช่วยเหลืออำนวยความสะดวกให้กับผู้สูงอายุ ผู้พิการทางขา หรือผู้ที่ต้องการความสะดวกสบายในการเคลื่อนที่ไปยังจุดหมายที่ผู้ใช้ต้องการที่จะไป ผู้พัฒนาได้ทำการศึกษาและพัฒนาระบบแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ โดยมีการควบคุมการใช้งานผ่านบนระบบปฏิบัติการแอนดรอยด์ที่สามารถควบคุมการทำงานของรถวีลแชร์ธรรมดาได้ มีฟังก์ชันการทำงานหลัก 2 ฟังก์ชัน ได้แก่ ฟังก์ชันควบคุมโดยผู้ใช้งาน (Manual) โดยผู้ใช้งานสามารถควบคุมรถนั่งอัตโนมัติด้วยตนเองได้เพื่อให้รถนั่งอัตโนมัติเคลื่อนที่ไปในทิศทางที่ต้องการ และฟังก์ชันอัตโนมัติ (Automatic) โดยผู้ใช้สามารถเลือกสถานที่เป้าหมายเพื่อให้รถนั่งอัตโนมัติเคลื่อนที่ไปโดยอัตโนมัติ ระบบสามารถแจ้งเตือนผ่านแอปพลิเคชันเมื่อรถนั่งอัตโนมัติเกิดเอนเอียง สามารถร้องขอความช่วยเหลือและโทรออกฉุกเฉินไปยังผู้ดูแลภายในโรงพยาบาลนั้น ๆ และยังสามารถตรวจจับวัตถุหรือสิ่งกีดขวางเพื่อป้องกันการเกิดอุบัติเหตุ

โครงการปัญหาพิเศษนี้ได้นำเสนอแนวคิดและการพัฒนาระบบแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ รวมถึงการติดตั้งอุปกรณ์ที่รถวีลแชร์เพื่อให้สามารถควบคุมรถวีลแชร์ผ่านทางสมาร์ทโฟนได้ เพื่อขอความช่วยเหลือ เพิ่มความปลอดภัย และอำนวยความสะดวกให้กับผู้ที่ต้องการความสะดวกสบาย การพัฒนาระบบแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ มีการใช้ภาษาโปรแกรมมิ่ง (Programing Language) ได้แก่ จาวา (Java) ภาษาซี (C) และภาษาไพทอน (Python) ติดตั้งแอปพลิเคชันเพื่อใช้งานบนมือถือสมาร์ทโฟนที่ใช้ระบบปฏิบัติการแอนดรอยด์ ติดตั้งอุปกรณ์ต่าง ๆ ที่รถวีลแชร์ธรรมดาเพื่อให้สามารถควบคุมรถวีลแชร์ได้ผ่านสมาร์ทโฟน และนำราสเบอร์รี่พาย (Raspberry Pi) มาใช้เพื่อเพิ่มประสิทธิภาพในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์

1) พัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ที่ควบคุมการทำงานของรถนั่งอัตโนมัติได้
2) แอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์เลือกตำแหน่งเพื่อให้รถนั่งอัตโนมัติสามารถเคลื่อนที่โดยอัตโนมัติ ตามการประมวลผลภาพจากป้ายต่าง ๆในโรงพยาบาล
3) แอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์สามารถใช้ควบคุมการเคลื่อนที่ของรถนั่งอัตโนมัติโดยผู้ใช้สามารถบังคับทิศทางได้
4) เพื่ออำนวยความสะดวกสบายให้กับผู้ใช้รถนั่งอัตโนมัติในโรงพยาบาล โดยการใช้กล้องประมวลผลภาพ

1.3 ขอบเขตของปัญหาพิเศษ

ขอบเขตการดำเนินงานรถนั่งอัตโนมัติภายในโรงพยาบาลสำหรับผู้สูงอายุ หรือผู้พิการทางขาที่จำเป็นต้องใช้รถนั่งอัตโนมัติในการเคลื่อนที่ควบคุมผ่านแอนดรอยด์แอปพลิเคชัน

1) คุณสมบัติที่สามารถรองรับแอปพลิเคชัน
   - แอปพลิเคชันสามารถใช้งานในอุปกรณ์ที่รองรับระบบปฏิบัติการแอนดรอยด์ที่ควบคุมการทำงานของรถนั่งอัตโนมัติที่ใช้ภายในโรงพยาบาล
   - แอปพลิเคชันนี้ต้องใช้งานผ่านบลูทูธ
2) ลักษณะผู้ใช้งานรถนั่งอัตโนมัติที่เหมาะสม ได้แก่ บุคคลที่มีอายุตั้งแต่ 5 ปีขึ้นไปที่ต้องการความสะดวกสบาย ผู้ที่อวัยวะของร่างกายบางส่วนไม่สามารถทำงานได้ตามปกติ ผู้สูงอายุ หรือผู้ได้รับการกระทบกระเทือนทางขาทำให้เป็นผู้พิการ โดยมีอาการในลักษณะดังต่อไปนี้
   - สามารถขยับนิ้วมือได้
   - ผู้พิการทางขา
   - ผู้สูงอายุที่เคลื่อนไหวไม่สะดวก
   - ผู้ที่ต้องการความสะดวกสบาย
3) ลักษณะพื้นที่ที่เหมาะกับการใช้งานรถนั่งอัตโนมัติสำหรับใช้ภายในโรงพยาบาลที่มีลักษณะดังต่อไปนี้
   - พื้นโรงพยาบาลมีลักษณะเป็นพื้นเรียบไม่ขรุขระ
   - มีป้ายบอกตำแหน่งทุกที่มีการเลี้ยวโค้ง
   - กรณีที่มีทางลาดชันภายในโรงพยาบาล ทางลาดชันจะต้องมีความชันไม่เกิน 1:12 ความลาดยาวช่วงละไม่เกิน 6,000 มิลลิเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1) รถนั่งอัตโนมัติที่ควบคุมด้วยแอปพลิเคชันบนแอนดรอยด์ ช่วยอำนวยความสะดวกให้กับผู้ใช้รถนั่งอัตโนมัติที่เป็นบุคคลที่มีอายุตั้งแต่ 60 ปีขึ้นไปที่ต้องการความสะดวกสบาย ผู้สูงอายุ หรือผู้พิการทางขาในการเคลื่อนที่ไปในทิศทางต่าง ๆ ได้
2) เพิ่มความสะดวกบายให้แก่ผู้ใช้รถนั่งอัตโนมัติ ด้วยฟังก์ชันการเคลื่อนที่อัตโนมัติ

1.5 ขั้นตอนการดำเนินงาน

1) วางแผนการทำโครงงาน
2) ศึกษาการทำงานของอุปกรณ์เบื้องต้น การเชื่อมต่อสัญญาณระหว่างอุปกรณ์ Raspberry pi และโทรศัพท์มือถือ Smartphone
3) ศึกษาอุปกรณ์ที่ใช้ติดตั้งในตัวรถนั่งอัตโนมัติ
4) ศึกษาเกี่ยวกับการพัฒนาระบบแอนดรอยด์แอปพลิเคชัน
5) วิเคราะห์และออกระบบแอนดรอยด์แอปพลิเคชัน
6) ติดตั้งอุปกรณ์ต่าง ๆ ที่รถนั่ง
7) พัฒนาระบบแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติตามที่ออกแบบไว้
8) ทดสอบการทำงานของระบบแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ
9) สรุปผลและจัดทำเอกสาร

1.6 อุปกรณ์ที่ใช้

1) ฮาร์ดแวร์
    - รถวีลแชร์
    - มอเตอร์เกียร์ 24 vdc 250w 400rpm
    - Wall Adapter Power Supply - 5V DC 2A (USB Micro-B)
    - GPIO Extension Board for Raspberry Pi B+
    - GPIO Ribbon Cable for Raspberry Pi Model B+- (40 pins)
    - Micro sd card 13 gb Class 10
    - Raspberry Pi 2 Model 3 1GB
    - H-BRIDGE DRIVER DC MOTOR 80A
    - แบตเตอรี่แห้ง 12V 18AH จำนวน 2 ก้อน
    - สมาร์ทโฟน (Smartphone) จำนวน 1 เครื่อง
    - Bluetooth Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Cytron Dual Ch 10A DC Motor Driver Rev2
- JSN-SR04T Waterproof Ultrasonic Module 4 ชิ้น
- Accelerometers
- Camera Serial Interface

2) ซอฟต์แวร์

- Android Studio
- Adobe Photoshop CS6
- Firebase
- Arduino IDE
- Python 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

บทนี้เป็นการกล่าวถึงทฤษฎีที่เกี่ยวข้องกับการพัฒนาแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ เพื่อเป็นแนวทางในการศึกษาเกี่ยวกับการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ การใช้งานRaspberry pi กับโมดูลสื่อสารไร้สายบลูทูธ (Bluetooth module) หลักการสื่อสารระหว่าง Raspberry pi กับแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ผ่านสัญญาณบลูทูธ การควบคุมรถวีลแชร์ผ่านระบบปฏิบัติการแอนดรอยด์และพัฒนาแอปพลิเคชันบนมือถือเพื่อช่วยเหลือในการเคลื่อนที่โดยการควบคุมรถวีลแชร์

2.1 โมดูลสื่อสารไร้สายบลูทูธ (Bluetooth Module)

Bluetooth Module [1] โมดูล Bluetooth ทำให้ Arduino มีความสามารถสื่อสารแบบไร้สายโดยใช้ Bluetooth ใช้งานง่ายด้วยอินเตอร์เฟส Serial Port เขียนโปรแกรมง่าย เช่นเดียวกับการสั่งผ่าน Serial Port สามารถสั่งงานควบคุมจากอุปกรณ์ที่มี Bluetooth เช่น เครื่องคอมหรือโทรศัพท์มือถือที่มี Bluetooth แสดงดังรูปที่ 2.1

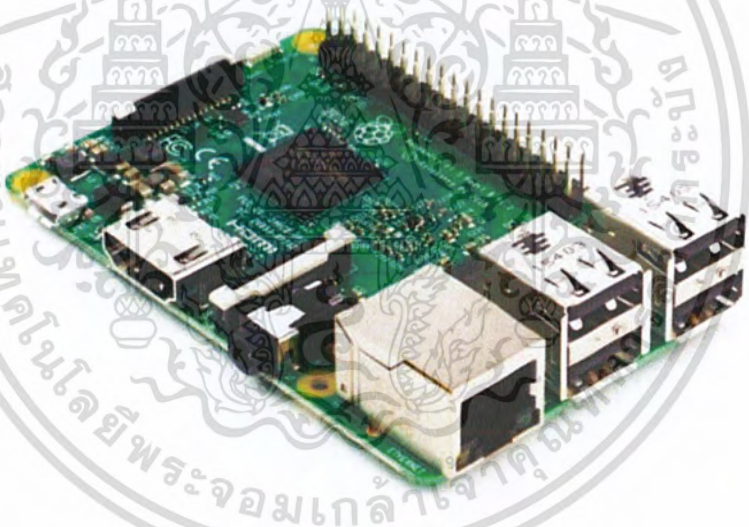
รูปที่ 2.1 Bluetooth Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 บอร์ดราสเบอร์รี่พาย (Raspberry Pi Board)

Raspberry Pi [2] คือบอร์ดคอมพิวเตอร์ขนาดเล็กที่สามารถเชื่อมต่อกับจอมอนิเตอร์ คีย์บอร์ด และเมาส์ได้ สามารถนำมาประยุกต์ใช้ในการทำโครงงานทางด้านอิเล็กทรอนิกส์ การเขียนโปรแกรม หรือเป็นเครื่องคอมพิวเตอร์ตั้งโต๊ะขนาดเล็ก ไม่ว่าจะเป็นการทำงาน Spreadsheet Word Processing ท่องอินเทอร์เน็ต ส่งอีเมล์ หรือเล่นเกมส์ อีกทั้งยังสามารถเล่นไฟล์วิดีโอความละเอียดสูง (High-Definition) ได้อีกด้วย

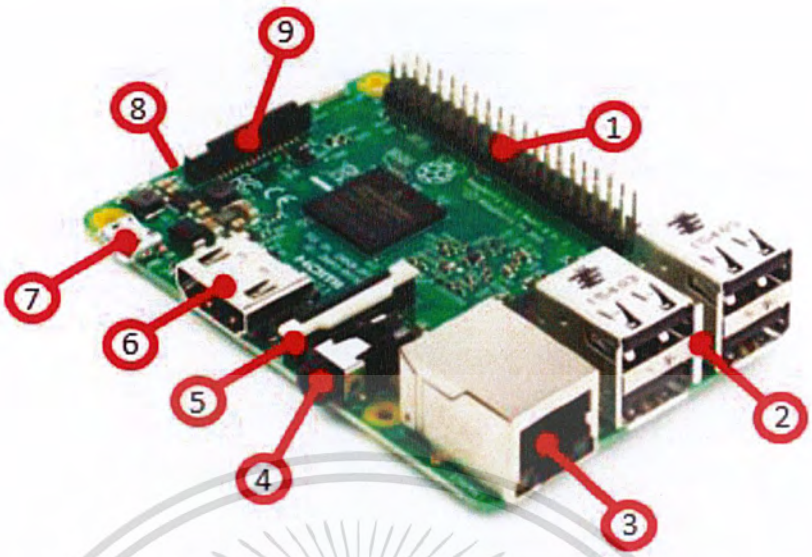
บอร์ด Raspberry Pi รองรับระบบปฏิบัติการลินุกซ์ (Linux Operating System) ได้หลายระบบ เช่น Raspbian (Debian) Pidora (Fedora) และ Arch Linux เป็นต้น โดยติดตั้งบน SD Card บอร์ด Raspberry Pi นี้ถูกออกแบบมาให้มี CPU GPU และ RAM อยู่ภายในชิปเดียวกัน มีจุดเชื่อมต่อ GPIO ให้ผู้ใช้สามารถนำไปใช้ร่วมกับอุปกรณ์อิเล็กทรอนิกส์อื่น ๆ ได้อีกด้วยบอร์ด Raspberry Pi แสดงดังรูปที่ 2.2



รูปที่ 2.2 บอร์ด Raspberry Pi

บอร์ด Raspberry Pi ปัจจุบันมีด้วยกัน 2 โมเดล คือ โมเดล A และโมเดล B ซึ่งทั้ง 2 โมเดล มีคุณสมบัติทางเทคนิคที่ใกล้เคียงกัน แตกต่างกันเพียงบางส่วน รายละเอียดส่วนประกอบของ Raspberry Pi แสดงดังรูปที่ 2.3

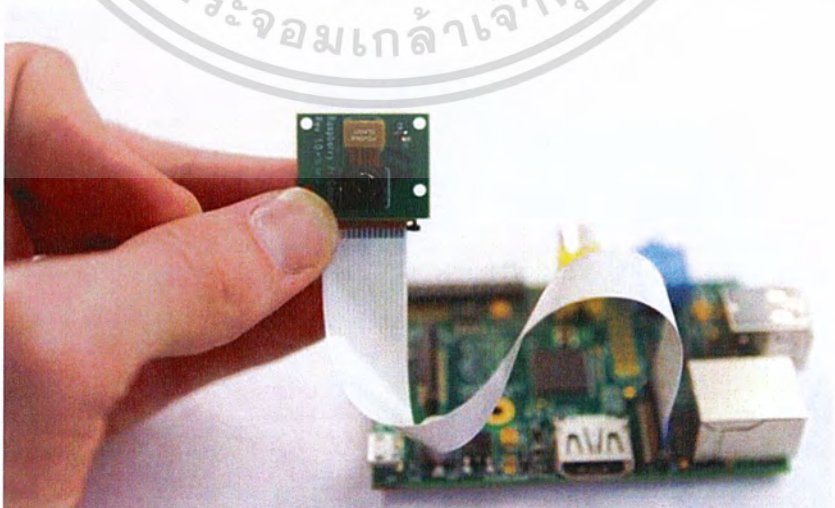
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ส่วนประกอบของ Raspberry Pi

จากรูปที่ 2.3 สามารถอธิบายรายละเอียดของส่วนประกอบได้ดังนี้

1) พอร์ต GPIO ซึ่งในโมเดล A และ B ทุก Pin จะเหมือนกัน แต่โมเดล B จะแตกต่างกัน
2) พอร์ต USB 2.0 จำนวน 2 พอร์ต
3) พอร์ต RJ-45 Ethernet LAN 10/100Mbps
4) ช่อง Audino/Video
5) พอร์ต CSI (Camera Serial Interface) สำหรับเชื่อมต่อโมดูลกล้องดังรูปที่ 2.4



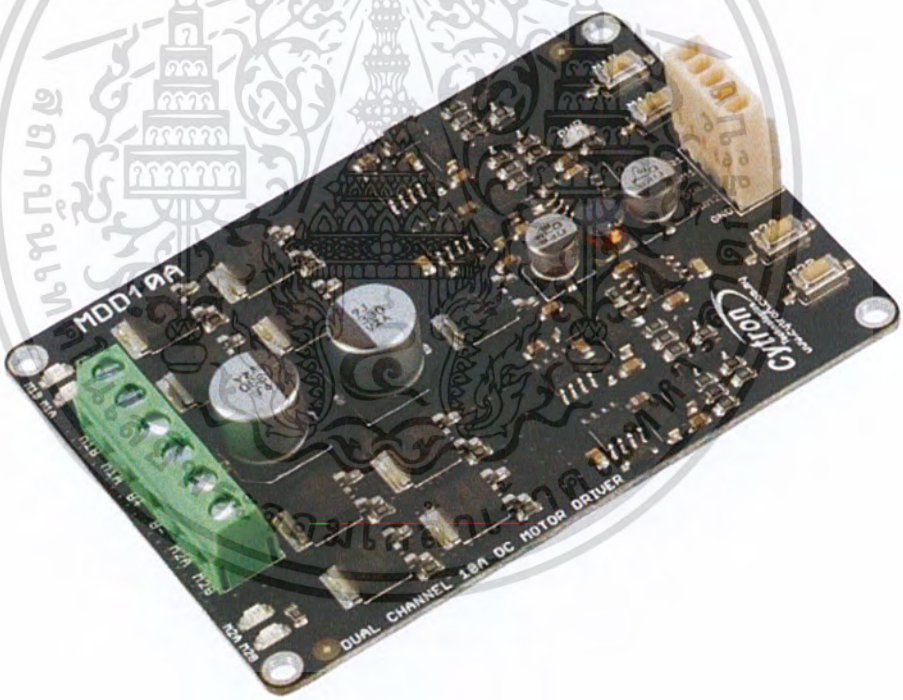
รูปที่ 2.4 พอร์ต Camera Serial Interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) พอร์ต HDMI สำหรับเชื่อมต่อสัญญาณภาพและเสียง
7) พอร์ต Micro USB Power สำหรับเป็นไฟเลี้ยงวงจรบอร์ด Raspberry Pi
8) ช่องเสียบ SD Card อยู่บริเวณด้านล่างของบอร์ด
9) พอร์ต DSI (Display Serial Interface) ใช้สำหรับต่อจอแสดงผล เช่น จอแสดงผลแบบ
TFT Touch Screen เป็นต้น

2.3 บอร์ดขับมอเตอร์ไฟฟ้ากระแสตรง (MDD10A)

บอร์ดขับมอเตอร์ [3] ไฟฟ้ากระแสตรง เป็นบอร์ดที่สามารถขับมอเตอร์ไฟฟ้ากระแสตรงได้สูงสุด 25V 10A ขับมอเตอร์ได้พร้อมกันถึง 2 ตัว และใช้งานควบคู่กับบอร์ดไมโครคอนโทรลเลอร์ได้ทุกตระกูล บอร์ดขับมอเตอร์ไฟฟ้ากระแสตรง แสดงดังรูปที่ 2.5



รูปที่ 2.5 บอร์ดขับมอเตอร์ไฟฟ้ากระแสตรง (MDD10A)

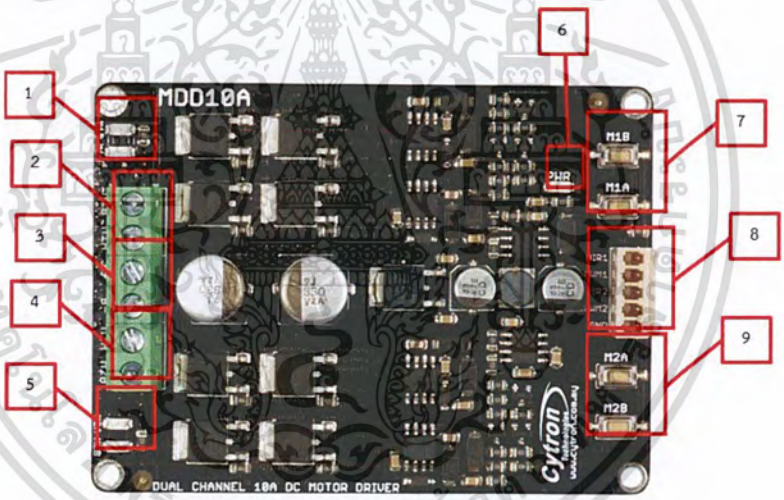
บอร์ดขับมอเตอร์ไฟฟ้ากระแสตรง มีคุณสมบัติทางเทคนิคดังต่อไปนี้

1) ควบคุมได้ 2 ทิศทางสำหรับมอเตอร์กระแสตรงชนิดแบบมีแปรงถ่าน (Brushed Motor)
2) รองรับมอเตอร์ที่ใช้แรงดันได้ในช่วง 5V ถึง 25V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ขับกระแสไฟต่อเนื่องได้สูงสุด 10A และขับกระแสไฟได้ถึงค่าสูงสุด (Peak) 30A ต่อเนื่องนานถึง 10 วินาที ในแต่ละช่อง
4) ใช้โซลิดคาปาซิเตอร์ เพื่อช่วยให้การตอบสนองที่รวดเร็ว และมีความคงทนช่วยให้มีอายุการใช้งานที่ยืนยาว
5) ใช้ NMOS H-Bridge โดยไม่จำเป็นต้องต่อ Heat sink ระบายความร้อน
6) ควบคุมความถี่ PWM (Pulse Width Modulation) สูงสุด 20kHz และรองรับการทำงาน PWM ทั้งแบบ Locked-Antiphase และ Sign-Magnitude
7) มีปุ่มกดสำหรับทดสอบการหมุนของมอเตอร์ได้ 2 ทิศทางในแต่ละช่อง เพื่อความรวดเร็วในการทดสอบ ไม่จำเป็นต้องต่อวงจรเพิ่มเติม หรือใช้ไมโครคอนโทรลเลอร์ควบคุม
8) ขนาดบอร์ด กว้าง 84.5 มม. และยาว 62 มม.

ส่วนประกอบของบอร์ด MDD10A แสดงดังรูปที่ 2.6



รูปที่ 2.6 ส่วนประกอบของบอร์ด MDD10A

จากรูปที่ 2.6 แสดงส่วนประกอบของบอร์ด MDD10A โดยมีรายละเอียดของส่วนประกอบดังต่อไปนี้

1) หลอดไฟ LED สีแดงแสดงสถานะการทำงานของมอเตอร์ที่ช่อง 1
2) ช่องเสียบมอเตอร์ของช่อง 1 โดยเป็น Output Pin ของบอร์ด
3) ช่องเสียบแหล่งจ่ายไฟ
4) ช่องเสียบมอเตอร์ของช่อง 2 โดยเป็น Output Pin ของบอร์ด
5) หลอดไฟ LED สีแดงแสดงสถานะการทำงานของมอเตอร์ที่ช่อง 2
6) หลอดไฟ LED สีเขียวแสดงสถานะไฟที่จ่ายให้กับบอร์ด
7) ปุ่มทดสอบการควบคุมทิศทางการหมุนของมอเตอร์ช่อง 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

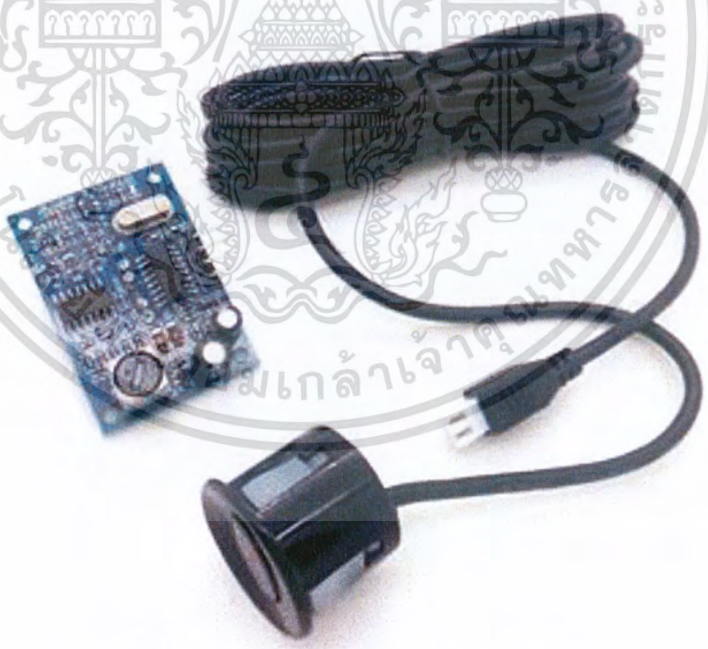
8) ช่องเสียบ Input Pin ใช้สำหรับควบคุมการทำงานด้วยบอร์ดไมโครคอนโทรลเลอร์

9) ปุ่มทดสอบการควบคุมทิศทางการหมุนของมอเตอร์ช่อง 2

วิธีการใช้งานสามารถใช้งานได้ 2 วิธี โดยวิธีแรก คือ ใช้โดยไม่ต่อวงจร หรือไมโครคอนโทรลเลอร์เพิ่มเติม ซึ่งวิธีนี้จะใช้สำหรับทดสอบการควบคุมทิศทางการหมุนของมอเตอร์ในเบื้องต้นเท่านั้น โดยสามารถต่อมอเตอร์เข้ากับบอร์ดและจ่ายไฟได้โดยตรง แล้วกดปุ่ม M1A, M1B หรือ M2A, M2B บนบอร์ดตามช่องที่ต่อมอเตอร์ไว้ สามารถใช้ควบคุมทิศทางการหมุนของมอเตอร์ได้ทันที วิธีที่สอง คือ ใช้ไมโครคอนโทรลเลอร์ควบคุมทิศทางและความเร็ว ซึ่งวิธีนี้จะเป็นวิธีที่นำไปใช้งานที่ต้องการควบคุมทั้งทิศทางและความเร็วของมอเตอร์ สามารถนำไปประยุกต์ใช้งานอื่น ๆ ได้ เช่น หุ่นยนต์ รถบังคับ เป็นต้น

2.4 เซนเซอร์วัดระยะ (JSN-SR04T)

Ultrasonic Module JSN-SR04T [4] คือ โมดูล Ultrasonic ซึ่งมีรูปแบบการใช้งานเช่นเดียวกับ โมดูล HS-S04 ใช้สำหรับตรวจจับวัตถุ และหาระยะห่างระหว่างตัว Sensor และวัตถุได้ในระยะ 25 เซนติเมตร ถึง 4.5 เมตร และสามารถป้องกันละอองน้ำได้แสดงดังรูปที่ 2.7



รูปที่ 2.7 เซนเซอร์วัดระยะ (JSN-SR04T)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 รถเข็นวีลแชร์ (Wheelchair)

โดยทั่วไป รถเข็นผู้ป่วย [5] รถเข็นผู้สูงอายุ รถเข็นผู้พิการ ที่พบเห็นตามสถานที่ต่าง ๆ และโรงพยาบาล คือ รถเข็นประเภทวีลแชร์ ซึ่งมีชื่อเรียกต่าง ๆ กันไม่ว่าจะเป็น รถเข็นรถเข็นธรรมดา รถเข็นนั่งรถเข็นวีลแชร์ หรือ รถวีลแชร์ ซึ่งทั้งหมดเป็นแบบเดียวกัน และมีลักษณะทั่วไปเหมือนกัน

รถเข็นวีลแชร์สามารถแบ่งย่อยได้เป็น 2 ประเภทคือ

1) Transport Wheelchair รถเข็นวีลแชร์ประเภทนี้จำเป็นต้องอาศัยคนช่วยเข็นเท่านั้น คนนั่งไม่สามารถเข็นเองได้ มักใช้ใน โรงพยาบาล เพื่อเคลื่อนย้ายคนไข้ รถเข็นวีลแชร์ประเภทนี้ จะมีล้อหลังขนาดเล็กขนาดเส้นผ่าศูนย์กลางตั้งแต่ประมาณ 8-14 นิ้ว ผู้นั่งจะไม่สามารถเอื้อมมือไปขยับล้อเพื่อเคลื่อนไหวรถเข็นด้วยตัวเองได้ ข้อดีคือน้ำหนักเบา ราคาค่อนข้างถูก

2) Manually Propelled Wheelchair รถเข็นวีลแชร์ประเภทนี้ผู้ใช้งานสามารถที่จะขยับรถเข็นเคลื่อนที่เองได้โดยจะต้องใช้แขนทั้งสองข้างในการช่วยหมุนล้อ และในเวลาที่ต้องการเบรกต้องใช้แขนทั้งสองข้างจับล้อเพื่อช่วยในการชะลอผ่อนความเร็วรถ รถเข็นวีลแชร์ประเภทนี้จะมีล้อหลังขนาดใหญ่ตั้งแต่ขนาดเส้นผ่าศูนย์กลางประมาณ 20-24 นิ้ว หรือ 51-61 ซ.ม. ข้อดีคือผู้ใช้งานสามารถบังคับควบคุมรถได้ด้วยตนเอง

ในปัจจุบันรถเข็นนั่งผู้ป่วยประเภทวีลแชร์ทั้ง 2 ประเภท จะมีทั้งแบบพับได้และพับไม่ได้ แต่สินค้ารุ่นหลัง ๆ ส่วนใหญ่มักจะออกแบบมาให้พับเก็บได้เพื่อความสะดวกในการเคลื่อนย้าย รวมถึงวัสดุของรถเข็นก็มีทั้งแบบที่เป็นเหล็ก อลูมิเนียม ซึ่งส่งผลต่อน้ำหนักรวมของรถเข็น การแบ่งประเภทรถเข็นวีลแชร์ข้างต้นเป็นเพียงการแบ่งประเภทหลัก ๆ เท่านั้น ในต่างประเทศ รถเข็นวีลแชร์ยังสามารถแบ่งเป็นประเภทย่อย ๆ ลงไปได้อีกตามลักษณะการใช้งานเช่น แบ่งเป็น รถเข็นเด็กสำหรับผู้พิการเด็ก รถเข็นสำหรับการเล่นกีฬา รถเข็นสำหรับใช้บนเส้นทางขรุขระหรือบนหาดทราย รถเข็นที่สามารถปรับยืนได้ รถเข็นปรับนอนได้ รถเข็นแบบนั่งอาบน้ำ รถเข็นแบบนั่งถ่าย และ อื่น ๆ อีกมากมายรถเข็นวีลแชร์แสดงดังรูปที่ 2.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.8 รถเข็นวีลแชร์ (Wheelchair)

2.6 ภาษา Python

Python [6] เป็นภาษาระดับสูงภาษาหนึ่ง เป็นภาษาโปรแกรมที่มีความสามารถสูง ถูกสร้างขึ้นในปีคริสต์ศักราช 1989 โดย Guido van Rossum ซึ่งถูกพัฒนาขึ้นมาโดยไม่ยึดติดกับแพลตฟอร์ม ดังนั้นภาษา Python จึงสามารถรันภาษาได้บนระบบปฏิบัติการที่หลากหลายไม่ว่าจะเป็น Unix, Linux, Windows NT, Windows 2000, Windows XP หรือแม้แต่ระบบ FreeBSD อีกอย่างหนึ่ง ภาษาตัวนี้เป็นภาษาที่มีลักษณะเป็น Open Source เช่นเดียวกับภาษา PHP

ในปัจจุบันภาษาที่ใช้ในการพัฒนาเว็บแอปพลิเคชันมีมากมาย อาทิเช่น ภาษา Perl, PHP, JAVA, ASP, Tcl หรือ Python เป็นต้น สำหรับภาษา Python นับว่ายังใหม่ในวงการพัฒนาโปรแกรมบนเว็บ แต่ด้วยข้อดีหลายประการของภาษา Python ทำให้มีผู้นิยมใช้ภาษานี้มากขึ้น โดยสามารถสรุปข้อดีของภาษา Python ได้ดังนี้

1) ง่ายต่อการเรียนรู้ โดยภาษา Python มีโครงสร้างของภาษาไม่ซับซ้อนเข้าใจง่าย ซึ่งโครงสร้างภาษา Python จะคล้ายกับภาษา C มาก เพราะภาษา Python สร้างขึ้นมาโดยใช้ภาษา C นอกจากนี้ตัวภาษามีความยืดหยุ่นสูงทำให้การจัดการกับงานด้านข้อความ และText File ได้เป็นอย่างดี

2) ไม่ต้องเสียค่าใช้จ่าย เพราะตัวแปลภาษา Python อยู่ภายใต้ลิขสิทธิ์ GNU

3) ใช้ได้กับหลายแพลตฟอร์ม แม้ในช่วงแรกภาษา Python จะถูกออกแบบให้มีการใช้งานกับระบบ Unix แต่ในปัจจุบันได้มีการพัฒนาตัวแปลภาษา Python ให้สามารถใช้กับ

เอกสารนี้ [illegible] ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบปฏิบัติการอื่น ๆ ได้ อาทิเช่น Linux, Windows 95/98/ME, Windows NT, Windows 2000 และ OS/2

4) ภาษา Python ถูกสร้างขึ้นโดยได้รวบรวมเอาส่วนดีของภาษาต่าง ๆ เข้ามาไว้ด้วยกัน อาทิเช่น ภาษา C, C++, Java และ Perl

5) ภาษา Python เป็นภาษาประเภท Server Side Script คือ การทำงานของภาษา Python จะทำงานด้านฝั่งเครือข่ายแล้วส่งผลลัพธ์กลับมายังผู้รับบริการทำให้มีความปลอดภัยสูง

6) ใช้พัฒนา Web Service โดยที่ภาษา Python สามารถนำมาพัฒนาเว็บเซอร์วิส รวมทั้งใช้บริหารการสร้างเว็บไซต์สำเร็จรูปที่เรียกว่า Content Management Framework (CMF) ตัวอย่าง CMF ที่มีชื่อเสียงมากและเบื้องหลังทำงานด้วย Python คือ Plone

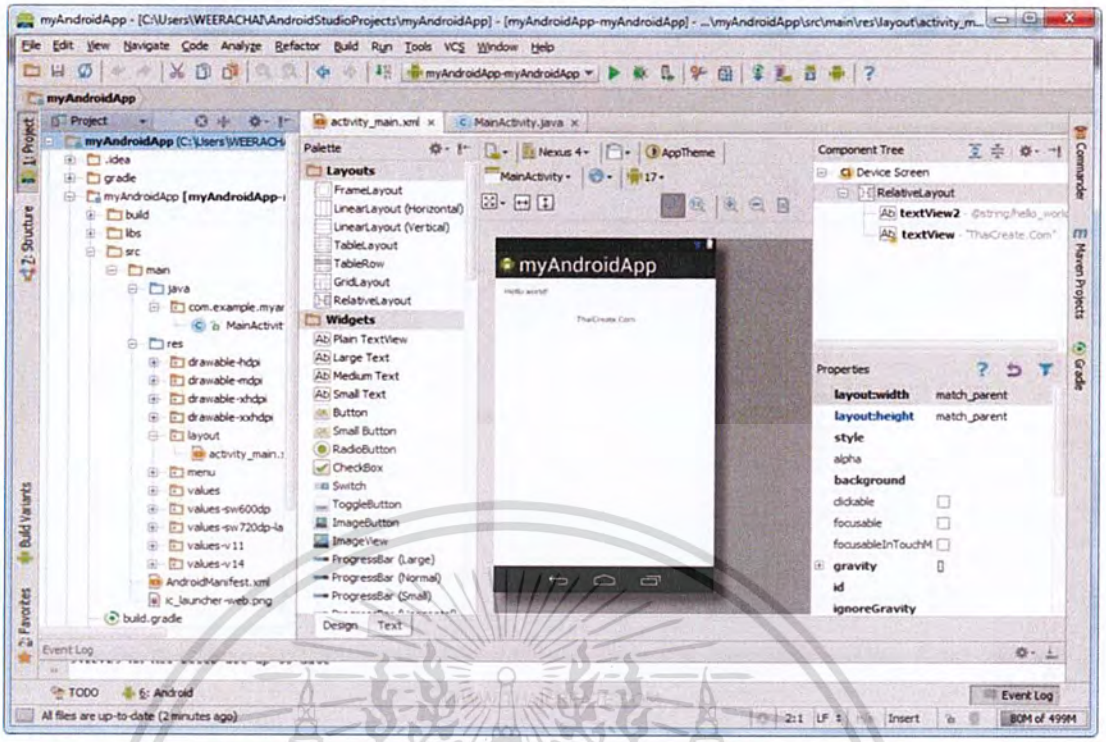
2.7 โปรแกรม Android Studio

Android Studio [7] เป็น IDE Tools จาก Google สำหรับพัฒนาโปรแกรมแอนดรอยด์โดยเฉพาะ มีการทำงานที่คล้ายกับ Eclipse และ Android ADT Plugin โดยวัตถุประสงค์ของการพัฒนา Android Studio คือ ต้องการพัฒนาเครื่องมือ IDE ที่สามารถพัฒนาแอปพลิเคชันบนแอนดรอยด์ ให้มีประสิทธิภาพมากขึ้น ทั้งด้านการออกแบบส่วนติดต่อกับผู้ใช้งาน (GUI) ที่ช่วยให้สามารถแสดงตัวอย่างมุมมองที่แตกต่างกันบน Smart Phone แต่ละรุ่นสามารถแสดงผลบางอย่างได้ทันทีโดยไม่ต้องทำการรันแอปพลิเคชันบน Emulator รวมทั้งยังแก้ไขปรับปรุงความเร็วของ Emulator ที่ยังเป็นปัญหาอยู่ในปัจจุบัน

การเขียนแอนดรอยด์บน Android Studio จะมีขั้นตอน 2 ขั้นตอน ขั้นตอนแรกได้แก่ การติดตั้ง Java SDK และขั้นตอนที่สองคือ การดาวน์โหลด Android Studio มาติดตั้ง จะสามารถใช้งานได้ทันทีโดยที่ไม่ต้องติดตั้ง Android ADT Plugin ซึ่งช่วยลดขั้นตอนการติดตั้งเครื่องมือต่าง ๆ ได้

ในปัจจุบัน Android Studio สามารถดาวน์โหลดเพื่อใช้งานบน Platform ต่าง ๆ ได้เกือบทุก ระบบปฏิบัติการเช่น Windows, Mac และ Linux โปรแกรม Android Studio แสดงดังรูปที่ 2.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 โปรแกรม Android Studio

2.8 ความพิการทางการเคลื่อนไหว

ความพิการทางการเคลื่อนไหว [8] หรือความพิการทางร่างกาย มีประกาศทางกระทรวงพัฒนาสังคมและความมั่นคงของมนุษย์เรื่อง ประเภทและหลักเกณฑ์ความพิการ ได้แบ่งความพิการทางการเคลื่อนไหวหรือร่างกายเป็น 2 กลุ่ม โดยมีรายละเอียดดังนี้

1) ความพิการทางการเคลื่อนไหว หมายถึง การที่บุคคลมีข้อจำกัดในการปฏิบัติกิจกรรมในชีวิตประจำวันหรือการเข้าไปมีส่วนร่วมในกิจกรรมทางสังคม ซึ่งเป็นผลมาจากการมีความบกพร่องหรือการสูญเสียความสามารถของอวัยวะในการเคลื่อนไหว ได้แก่ มือ เท้า แขน ขา อาจมาจากสาเหตุอัมพาต แขน ขา อ่อนแรง แขน ขาขาด หรือภาวะเจ็บป่วยเรื้อรังจนมีผลกระทบต่อการทำงานมือ เท้า แขน ขา

2) ความพิการทางร่างกาย หมายถึง การที่บุคคลมีข้อจำกัดในการปฏิบัติกิจกรรมในชีวิตประจำวันหรือการเข้าไปมีส่วนร่วมในกิจกรรมทางสังคม ซึ่งเป็นผลมาจากการมีความบกพร่องหรือความผิดปกติของศีรษะใบหน้า ลำตัว และภาพลักษณ์ภายนอกของร่างกายที่เห็นได้อย่างชัดเจน

หลักเกณฑ์ที่ใช้สำหรับการวินิจฉัยความพิการทางการเคลื่อนไหวหรือร่างกาย มีดังนี้

1) หลักเกณฑ์ความพิการ ได้แก่

- ความบกพร่องทางการเคลื่อนไหว ครอบคลุม ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แขนหรือขาอ่อนแรง จากสาเหตุต่าง ๆ เช่น อัมพาตจากโรคหลอดเลือดสมอง อุบัติเหตุทาง
- สมอง อุบัติเหตุไขสันหลัง มะเร็ง โรคติดเชื้อที่สมองหรือไขสันหลัง ภาวะผิดปกติทางสมองหรือไขสันหลังตั้งแต่กำเนิด
- แขน ขา ขาด จากสาเหตุต่าง ๆ เช่น เบาหวาน อุบัติเหตุ มะเร็ง แขนขาขาดหายตั้งแต่กำเนิด
- โรคข้อ หรือกลุ่มอาการปวด เช่น รูมาตอยด์ ข้อเข่าเสื่อม
- ภาวะเจ็บป่วยเรื้อรัง เหนื่อยง่าย จนมีผลกระทบต่อการปฏิบัติกิจกรรมในชีวิตประจำวัน เช่น โรคหัวใจ โรคไตวาย โรคปอด

- ความบกพร่องทางร่างกาย ครอบคลุมลักษณะ ดังนี้
  - ภาพลักษณ์ภายนอกร่างกายที่เห็นชัดเจน เช่น คนแคระที่มีขนาดลำตัวสั้นมากจนมีผลกระทบต่อการเข้าไปมีส่วนร่วมทางสังคม
  - ใบหน้า หรือศีรษะที่ผิดรูป เช่น หู ตา จมูก ปาก ที่ผิดรูป ผิดตำแหน่ง ผิดขนาด อาจจากสาเหตุอุบัติเหตุ ไฟไหม้ หรือโดนสารเคมี เช่น น้ำกรด มีผลกระทบต่อภาพลักษณ์ภายนอกอย่างรุนแรง
  - คอ หลัง ลำตัว ผิดปกติหรือผิดรูป เช่น กรณีหลังคด หรือผิดรูปอย่างรุนแรงที่เห็นเด่นชัด เมื่อดูด้วยตาเปล่า มีผลกระทบต่อภาพลักษณ์ภายนอกอย่างรุนแรง (กรณีนี้ต้องมีการตรวจประเมินว่าไม่สามารถจะทำการรักษาโดยการผ่าตัดได้แล้วหรือผู้ป่วยไม่ยินยอมผ่าตัด)

2) การวินิจฉัยคนพิการทางการเคลื่อนไหวหรือร่างกาย ดังนี้

- กรณีแขน ขาขาด วินิจฉัยได้ทันที
- กรณีอ่อนแรงของแขนหรือขา จากโรคหลอดเลือดสมอง โรคของสมอง อุบัติเหตุทางสมองหรือไขสันหลัง หรือโรคทางระบบประสาททั้งหมดต้องได้รับการรักษาและฟื้นฟูอย่างต่อเนื่องอย่างน้อย 3 เดือน หรืออยู่ในดุลยพินิจของแพทย์ เช่น ถ้าแพทย์ผู้นั้นมีความมั่นใจสามารถวิเคราะห์ได้ว่าเป็นความพิการที่ถาวรแล้วก็อาจออกเอกสารรับรองความพิการได้ก่อน
- กรณีอ่อนแรงของแขนหรือขา จากโรคเรื้อรังต่างๆ เช่น หัวใจล้มเหลว ไตวาย ที่อยู่ในระหว่างการรักษาและฟื้นฟูให้แพทย์ที่เป็นผู้รักษา หรือฟื้นฟูเท่านั้นเป็นผู้พิจารณาออกเอกสารรับรองความพิการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

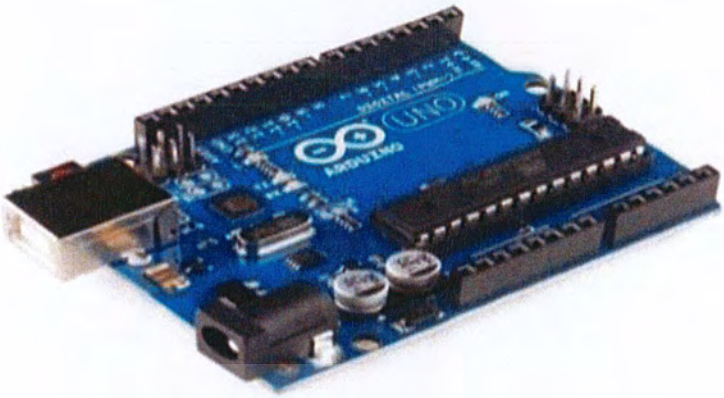
- กรณีที่ยังอยู่ในระหว่างการรักษาและฟื้นฟูโดยที่แพทย์ผู้รักษาฟื้นฟูวินิจฉัยแล้วว่าสามารถรักษาและฟื้นฟูได้แต่ผู้ป่วยปฏิเสธที่จะทำการรักษาและฟื้นฟูให้แพทย์ลงบันทึกไว้เป็นหลักฐานในเวชระเบียน และระบุในเอกสารรับรองความพิการ
- โรคลมชักที่ผู้ป่วยอยู่ระหว่างการรักษาแต่บางครั้งยังมีอาการชักจะประเมินว่าเป็นคนพิการได้ต้องมีภาวะแขน หรือขาอ่อนแรงตามเกณฑ์ที่กำหนดจึงจะพิจารณาว่ามีความบกพร่องทางการเคลื่อนไหว
- โรคที่ก่อให้เกิดความบกพร่องที่ผิวหนังบริเวณใบหน้าลำตัว หรือรยางค์แล้วมีผลต่อภาพลักษณ์ภายนอกที่เห็นด้วยตาเปล่าอย่างเด่นชัด หรือมีผลต่อการเคลื่อนไหวของข้อต่าง ๆ ตามร่างกาย เช่น โรคหนังแข็ง โรคสะเก็ดเงิน เด็กดักแด้ โรคเท้าแสนปม โรคตุ่มน้ำพองใส โรคด่างขาว คนเผือก เป็นต้น ให้ประเมินตามแนวทางการตรวจประเมินความบกพร่องทางการเคลื่อนไหว หรือร่างกาย

2.9 บอร์ดอาดูโน่ (Arduino Board)

Arduino [9] เป็นบอร์ดไมโครคอนโทรเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source มีการเปิดเผยข้อมูลทั้งด้านฮาร์ดแวร์และซอฟต์แวร์ บอร์ด Arduino ถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา และยังสามารถดัดแปลง เพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ดหรือโปรแกรมได้อีกด้วย

ความง่ายของบอร์ด Arduino ในการต่ออุปกรณ์เสริมต่าง ๆ คือผู้ใช้งานสามารถต่อวงจรอิเล็กทรอนิกส์จากภายนอกแล้วเชื่อมต่อเข้ามาที่ขา I/O ของบอร์ด หรือเพื่อความสะดวกสามารถเลือกต่อกับบอร์ดเสริม (Arduino Shield) ประเภทต่าง ๆ ยกตัวอย่างเช่น ArduinoXBee Shield, Arduino Music Shield, Arduino Relay Shield, Arduino Wireless Shield และArduino GPRS Shield เป็นต้น บอร์ด Arduino แสดงดังรูปที่ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



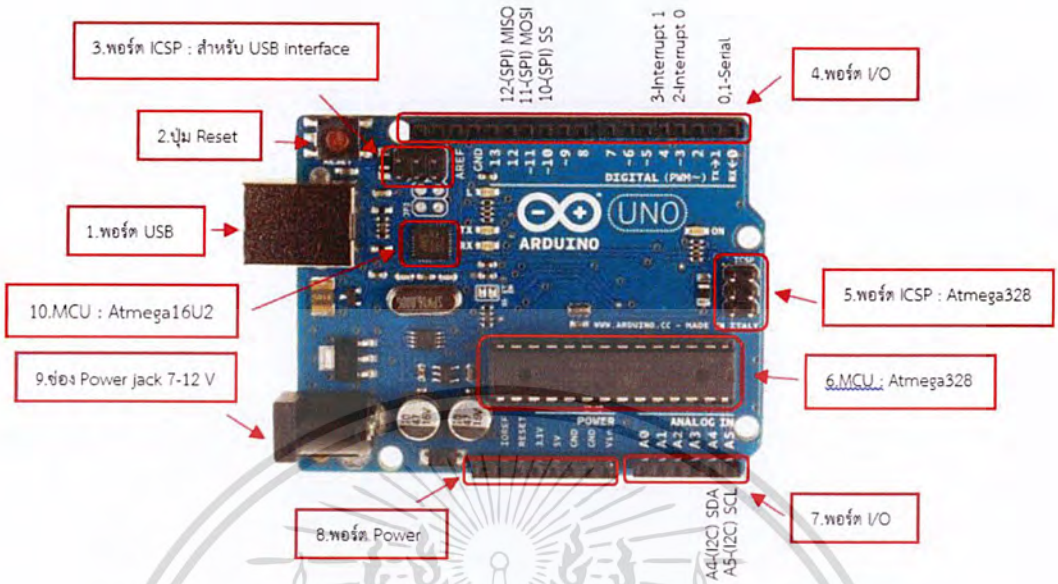
รูปที่ 2.10 บอร์ด Arduino

ส่วนประกอบของบอร์ด Arduino มีดังนี้

1) USBPort ใช้สำหรับต่อกับ Computer เพื่ออับโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด
2) Reset Button เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่
3) ICSP Port ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2
4) I/OPortDigital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้บาง Pin จะทำหน้าที่อื่น ๆ เพิ่มเติม
5) ICSP Port Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader
6) MCU Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino
7) I/OPort นอกจากจะเป็น Digital I/O แล้ว ยังเปลี่ยนเป็น ช่องรับสัญญาณอนาล็อกตั้งแต่ขา A0-A5
8) Power Port ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก
9) Power Jack: รับไฟจาก Adapter
10) MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนประกอบของบอร์ด Arduino แสดงดังรูปที่ 2.11



รูปที่ 2.11 ส่วนประกอบของบอร์ด Arduino

2.10 Firebase

Firebase[10] ถูกสร้างขึ้นจากคุณสมบัติเสริมว่านักพัฒนาสามารถผสมและจับคู่เพื่อให้พอดีกับความต้องการ บริษัทถูกก่อตั้งขึ้นในปีคริสต์ศักราช 2011 โดยมีแอนดรูลี และเจมส์ เทมปลิน เป็นผู้ก่อตั้ง สินค้าเริ่มต้นของ Firebase เป็นฐานข้อมูลเรียลไทม์ซึ่งมี API ที่ช่วยให้นักพัฒนาในการจัดเก็บและซิงค์ข้อมูลโดย Google Firebase 2.0 ในเวลาต่อมากูลเกิลได้ซื้อกิจการ Firebase และมีการพัฒนาจากบริการ backend ซึ่งเป็นการเก็บข้อมูลอย่างเดียว มาเป็นแพลตฟอร์มครบวงจรสำหรับนักพัฒนาแอปพลิเคชัน รองรับบริการเกือบทุกรูปแบบที่นักพัฒนาแอปพลิเคชันต้องใช้งาน บริการของ Firebase ได้แก่

- บริการวิเคราะห์ข้อมูล (Firebase Analytics) ดึงเทคโนโลยีมาจาก Google Analytics เปิดให้ใช้ฟรีแบบไม่จำกัดปริมาณข้อมูลใด ๆ
- ระบบส่งข้อความแจ้งเตือน (Firebase Cloud Messaging หรือ FCM) ซึ่งมีข้อดีคือสามารถใช้งานได้โดยไม่มีค่าใช้จ่ายและไม่จำกัดปริมาณข้อความ
- บริการพื้นที่เก็บข้อมูล (Firebase Storage) คือ บริการพื้นที่เก็บข้อมูลเก็บภาพวิดีโอ หรือไฟล์ขนาดใหญ่จากแอปพลิเคชันของผู้ใช้ สร้างอยู่บน Google Cloud Storage

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ตัวช่วยอัพเดตคอนฟิกของแอปพลิเคชัน (Firebase Remote Config) เป็นบริการสำหรับปรับแต่งค่าต่าง ๆ ในแอปพลิเคชันจากระยะไกล เช่น เกมที่อยากปรับสมดุลของเกมตลอดเวลา สามารถใช้ร่วมกับบริการวิเคราะห์ข้อมูล เพื่อกำหนดผู้ใช้งานแยกเป็นกลุ่ม ๆ ได้
- ตัวรายงานการแครชของแอปพลิเคชัน (Firebase Crash Reporting) รองรับการใช้งานบนระบบปฏิบัติการทั้ง ไอโอเอส (iOS) และแอนดรอยด์ (Android)
- บริการทดสอบแอปพลิเคชันบนฮาร์ดแวร์จริง(Firebase Test Lab for Android)
- Firebase Notifications เป็นคอนโซลสำหรับนักพัฒนา เพื่อยิงข้อความผ่าน FCM ไปยังผู้ใช้ สำหรับโฆษณาหรือกระตุ้นให้ผู้ใช้กลับมาเปิดแอปพลิเคชัน เช่น แจกของในเกม
- บริการ URL กลาง (Firebase Dynamic Links) สามารถชี้ทางไปยังเพจต่าง ๆ แปรผันตามอุปกรณ์หรือคุณสมบัติของผู้ใช้ เช่น แต่ละประเทศกดลิงก์เดียวกัน เข้าคนละเพจกัน
- ระบบเชิญเพื่อนมาใช้แอปพลิเคชัน (Firebase Invites) มีฟีเจอร์ referral
- Firebase App Indexing เปลี่ยนชื่อมาจาก Google App Indexing ที่ช่วยให้ Google Search ค้นเจอเนื้อหาภายในแอปพลิเคชัน

Firebase ครอบคลุมทุกการบริการสำหรับพัฒนา Realtime Application บริการเกือบทุกตัวของ Firebase ใช้งานได้โดยไม่เสียค่าใช้จ่ายและแบบไม่จำกัดปริมาณของหน่วยความจำ ยกเว้น Test Lab, Storage, Realtime Database และ Hosting ที่มีการเสียค่าใช้จ่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 SIFT (Scale Invariant Feature Transform)

นำมาใช้ในการวิเคราะห์เพื่อหาจุดเด่นของรูปภาพที่ได้รับเข้ามา โดยไม่ขึ้นกับขนาดหรือทิศทางของวัตถุในภาพ ซึ่งลักษณะเด่นที่ได้จากวัตถุลักษณะเดียวกันจะมีลักษณะที่คล้ายคลึงกัน โดยวิธีการของ SIFT[11] จะประกอบด้วย 4 ขั้นตอนที่สำคัญ ได้แก่

1) การหาปริภูมิค่าในมิติขนาดและระยะทาง (Scale-space Extrema Detection) จะเริ่มด้วยการหาลักษณะเด่นของภาพที่ไม่ขึ้นกับขนาดหรือทิศทาง โดยจะใช้วิธีการเบลอ (Blur) ภาพด้วยฟังก์ชันเกาส์เซียน (Gaussian Function) ไปในแต่ละขั้น (Octave) ซึ่งในแต่ละ Octave ก็จะมีรูปหลายๆระดับการ Blur ซึ่งมีขนาดรูปที่เท่ากัน โดยที่แต่ละระดับการ Blur จะ Blur จากระดับปกติแล้วค่อยๆเพิ่ม σ (Scale parameter) ซึ่งจะมีผลทำให้ภาพ Blur มากยิ่งขึ้น แล้วทำซ้ำกับ Octave ต่อไปเรื่อย ๆ โดย Octave ต่อไปก็จะมีขนาดของรูปเป็นครึ่งหนึ่งของ Octave เดิม การหาปริภูมิค่าในมิติขนาดและระยะทาง แสดงดังสมการที่ (2.1) และสมการที่ (2.2)

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{2.1}$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \tag{2.2}$$

โดยที่

L คือ ภาพที่ผ่านการ Blur

G คือ ตัวกรองแบบ Gaussian ที่มีขนาด σ

I คือ ภาพต้นฉบับ

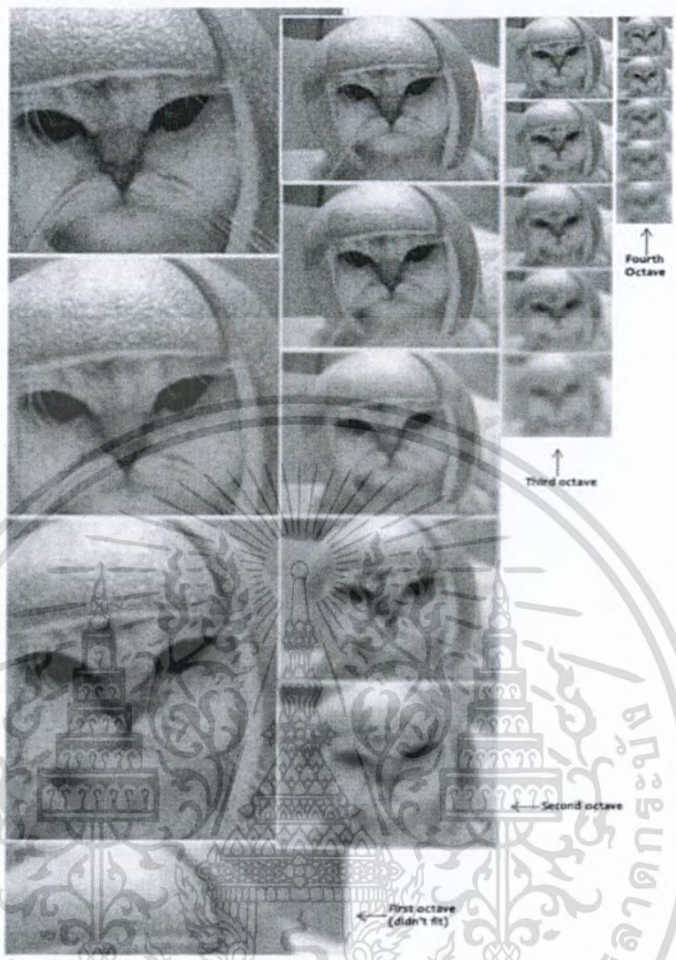
x,y คือ พิกัดบนภาพ

σ คือ ตัวแปรขนาดของการ Blur ยิ่งมีค่ามาก ยิ่ง Blur มาก

* คือ การคอนโวลูชั่นบนภาพ I โดยใช้ Gaussian blur G

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาปริภูมิค่าในมิติขนาด และระยะทางเสร็จจะได้ภาพที่มีความคมชัดต่ำ ดังรูปที่ 2.12



รูปที่ 2.12 การหาปริภูมิค่าในมิติขนาดและระยะทางเสร็จจะได้ภาพที่มีความคมชัดต่ำ

จากรูปที่ 2.12 อธิบายได้ว่า การใช้ σ ที่แตกต่างกันไปในแนวตั้งซึ่งได้ผลลัพธ์คือภาพที่มีความ Blur มากขึ้น และการขยับ Octave ขึ้นไปทีละขั้นโดยมีขนาดรูปภาพเป็นครึ่งหนึ่งของ Octave ก่อนหน้า ซึ่งในภาพนี้ได้แสดง 4 Octave โดยแต่ละ Octave มี 5 รูปที่เหมือนกันแต่ต่างกันที่ขนาดของการ Blur

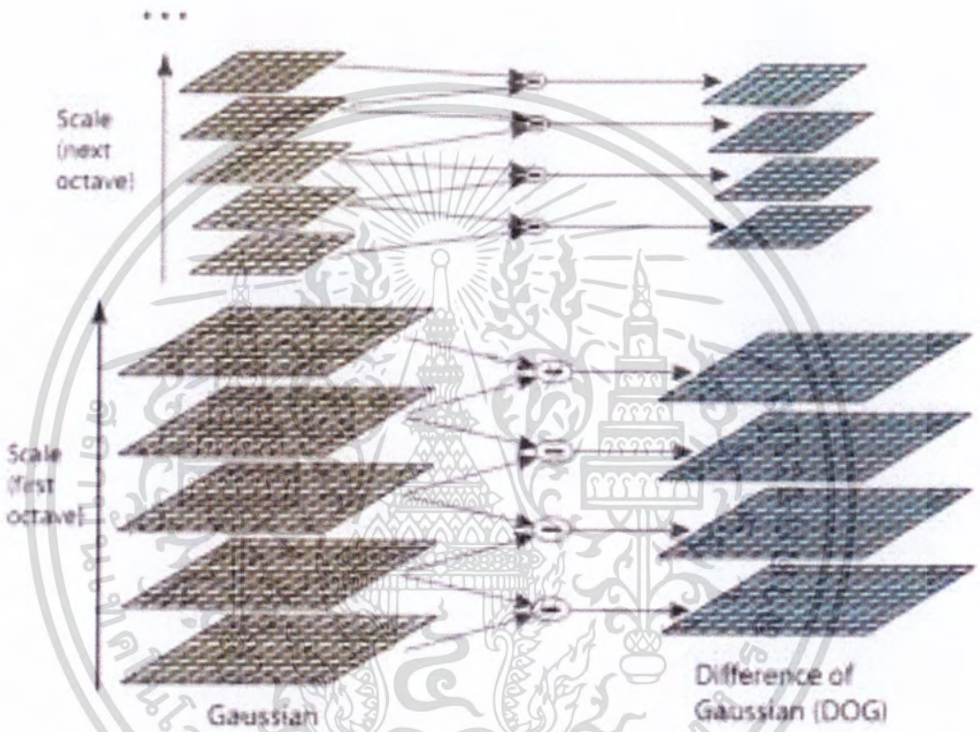
2) การกำหนดตำแหน่งจุดสนใจ (Keypoint localization) จากการหาปริภูมิค่าในมิติขนาดและระยะทางในขั้นตอนก่อนหน้า จะนำมาหาจุดที่น่าจะเป็นลักษณะสำคัญของภาพ โดยจะทำการจับคู่ภาพของแต่ละOctave มาหาความต่างของแต่ละผลลัพธ์ที่ได้จากการBlur (Difference of Gaussian: DoG) โดยจะทำทุกภาพในแต่ละ Octave และทำซ้ำจนครบทุก Octave การกำหนดตำแหน่งจุดสนใจ แสดงดังสมการที่ (2.3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$D(x,y,\sigma) = \big(G(x,y,k\sigma) - G(x,y,\sigma)\big) * I(x,y)$$

$$= L(x,y,k\sigma) - L(x,y,\sigma) \qquad (2.3)$$

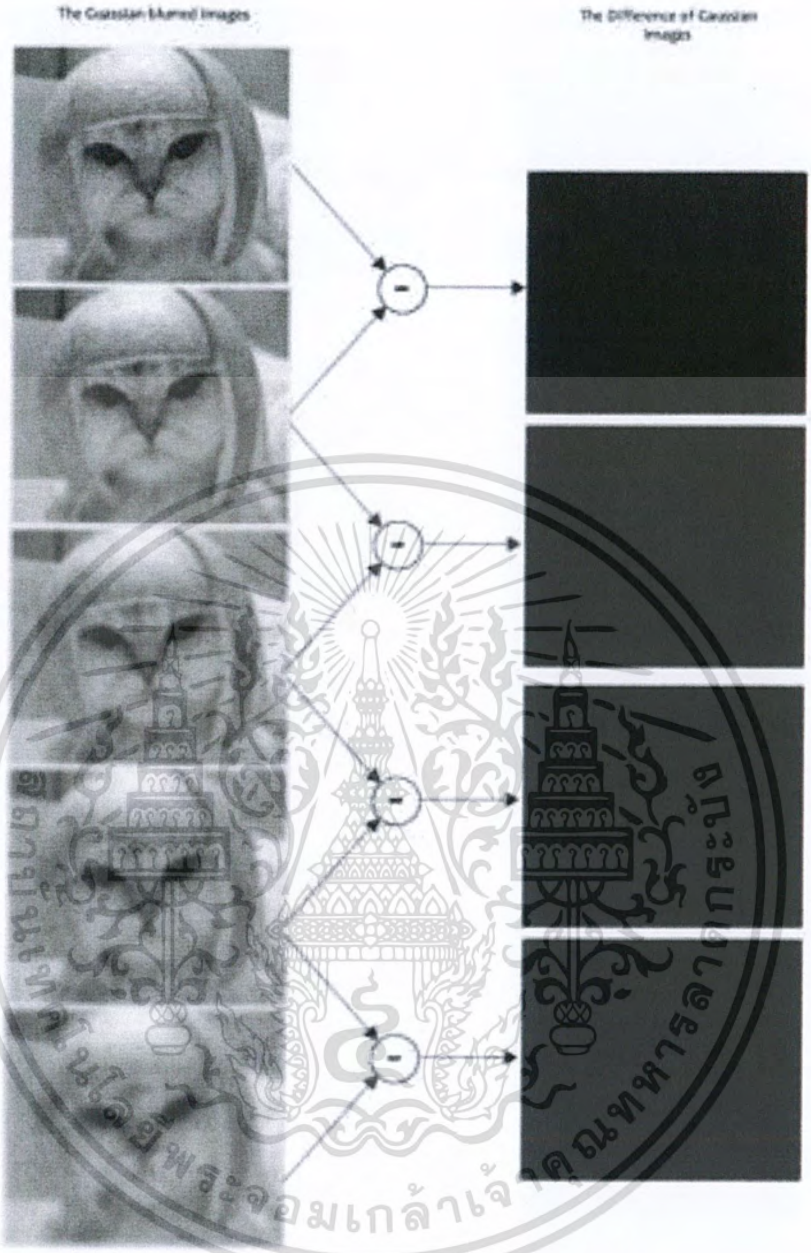
การหา Difference of Gaussian ในแต่ละขั้นของ Octave แสดงดังรูปที่ 2.13



รูปที่ 2.13 การหา Difference of Gaussian

การหา Difference of Gaussian ในแต่ละขั้นของ Octave จะได้ภาพเป็นเงา แสดงดังรูปที่ 2.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



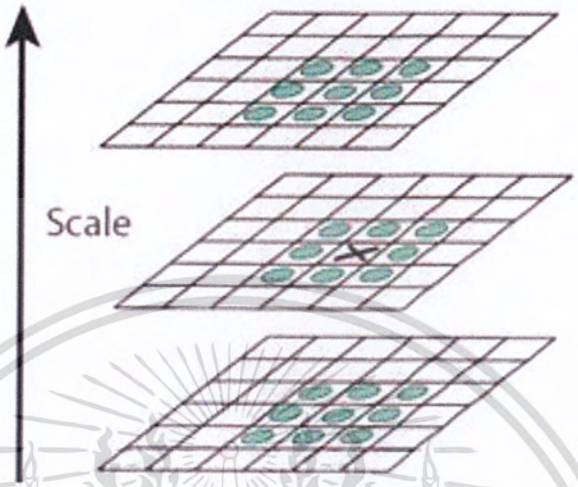
รูปที่ 2.14 ผลลัพธ์จากการทำ Difference of Gaussian ในแต่ละขั้นของ Octave

การหาจุดสำคัญ (Keypoints) ซึ่งจะถูกแบ่งออกเป็น 2 ส่วน คือ การประมาณหาค่าจุดภาพที่มีค่าสูงสุดหรือต่ำสุดแบบหยาบๆ และ การใช้สมการของ Taylor เพื่อทำการประมาณหา subpixel โดยมีรายละเอียดดังนี้

- การประมาณหาค่าจุดภาพ (Pixel) ที่มีค่าสูงสุดหรือต่ำสุดแบบหยาบๆ (Locate maxima/minima in DoG images) เป็นการประมาณหาค่าจุดภาพ (Pixel) จุดที่มีค่าสูงสุดหรือต่ำสุดเป็นจุดที่อยู่ระหว่างจุดภาพ เมื่อทำการเปรียบเทียบกับจุดรอบบ้านรอบข้าง 26 จุด ซึ่ง

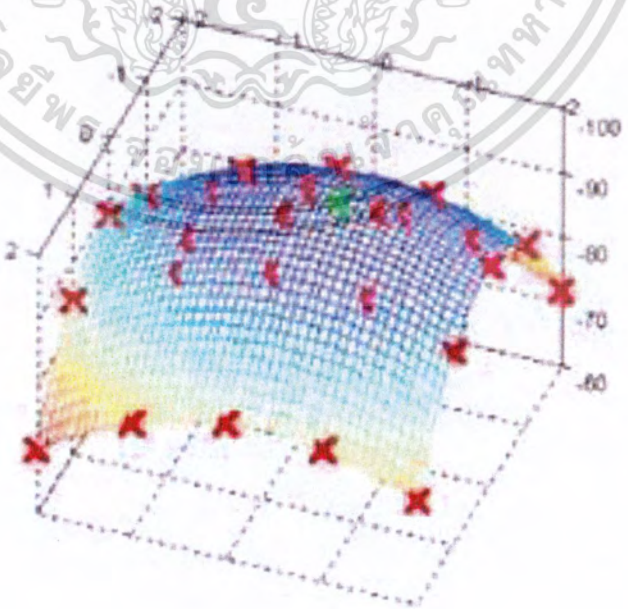
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

แบ่งเป็น 8 จุดที่อยู่ติดกับจุดที่พิจารณา และ 9 จุดบนและล่างที่อยู่ติดกับจุดที่พิจารณา แสดงดังรูปที่ 2.15



รูปที่ 2.15 จุด X ที่เป็นจุดอ้างอิงเปรียบเทียบกับจุดรอบข้าง 26 จุด

จุดสีแดงที่เป็นจุดสูงสุดหรือต่ำสุดที่เป็น pixel แต่จุดที่สูงสุดจริงๆ คือ จุดสีเขียว ซึ่งเป็น subpixel แสดงดังรูปที่ 2.16



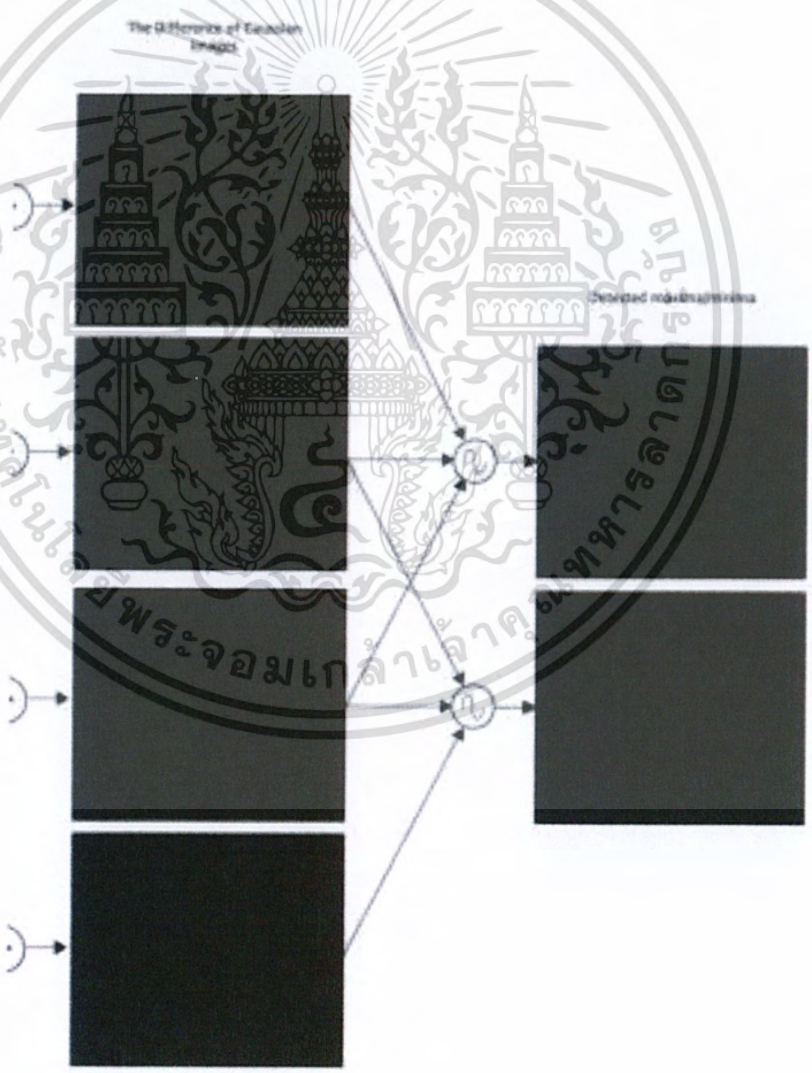
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.16 จุดสูงสุดหรือต่ำสุดที่เป็น pixel

- Find subpixel maxima/minima ใช้สมการของ Taylor เพื่อทำการประมาณหา subpixel ที่เป็นจุดที่มีค่าสูงสุดหรือต่ำสุดอย่างแท้จริง จาก pixel ที่ได้เลือกมาจากขั้นตอนก่อนหน้า แสดงดังสมการที่ (2.4)

$$D(x) = D + \frac{\partial D^T}{\partial x}x + \frac{1}{2}x^T\frac{\partial^2 D}{\partial x^2}x \qquad (2.4)$$

ผลลัพธ์จากการทำ Locate maxima/minima กับ 4 DoG images แสดงดังรูปที่ 2.17



รูปที่ 2.17 การทำ Locate maxima/minima กับ 4 DoG images

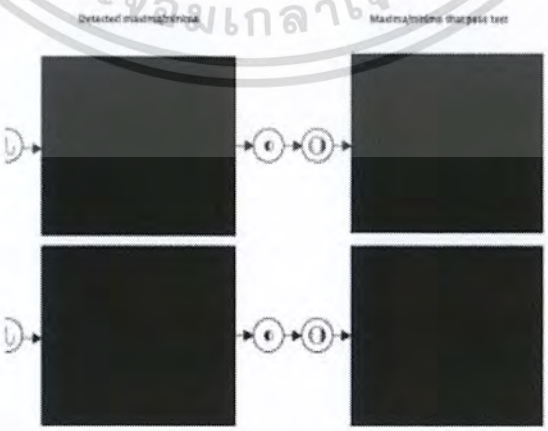
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากขั้นตอนที่ผ่านมาจะได้ Keypoints จำนวนมาก ซึ่งบางจุดจะวางเรียงตัวอยู่บนเส้นขอบ (Edge) หรือบางจุดมีความคมชัด (Contrast) โดยจุดเหล่านี้จะเป็นจุดที่จะทำการคัดออก เพื่อจะได้ผลลัพธ์สุดท้ายที่แต่ละจุดจะมีความเหมาะสมที่จะเป็น Keypoint มากที่สุด ประกอบด้วย 2 ขั้นตอน คือ Removing low contrast features และ Removing edges โดยมีรายละเอียดดังนี้

- Removing low contrast features ทำการหาค่าความเข้ม (Intensity) ของจุดที่เป็น subpixel ซึ่งได้ประมาณมาจากขั้นตอนที่ผ่านมา นำมาเปรียบเทียบกับค่าที่กำหนดค่าหนึ่งโดยถ้ามีค่าน้อยกว่าก็จะทำการคัดจุดนั้นออกจาก Keypoints

- Removing edges ทำการคำนวณ gradients 2 ตัว ของจุดที่เป็น keypoint ซึ่งตั้งฉากกันและกัน โดยสามารถสรุปความสัมพันธ์บนภาพจากจุดทั้ง 2 ได้เป็น 3 แบบ คือ พื้นที่ราบ ขอบ และ มุม โดยมีรายละเอียดดังนี้
  - พื้นที่ราบ (Flat region) จะมี gradient ที่มีค่าน้อยทั้งคู่
  - ขอบ (Edge) จะมี gradient ที่มีค่ามากสำหรับตัวที่ตั้งฉากกับขอบ และมี gradient ที่มีค่าน้อยสำหรับตัวที่ขนานไปกับขอบ
  - มุม (Corner) จะมี gradient ที่มีค่ามากทั้งคู่

โดยจะพิจารณา keypoint ที่มีลักษณะเป็นมุมเป็นหลัก ซึ่งถ้ามีค่าของ gradient ที่มากพอทั้งคู่ จะนิยามให้เป็น keypoint แต่ถ้าไม่มากพอก็จะถูกตัดทิ้ง และในขั้นตอนการพิจารณาหา keypoint ที่เป็นมุมนี้ สามารถใช้ Hessian Matrix เข้ามา เพื่อช่วยในการหาว่าเป็นมุมหรือไม่ ได้ดียิ่งขึ้น แสดงดังรูปที่ 2.18



รูปที่ 2.18 ผลลัพธ์ที่ได้หลังจากทำการหา keypoint จากการทดสอบ Edge และ Contrast ที่เหมาะสม

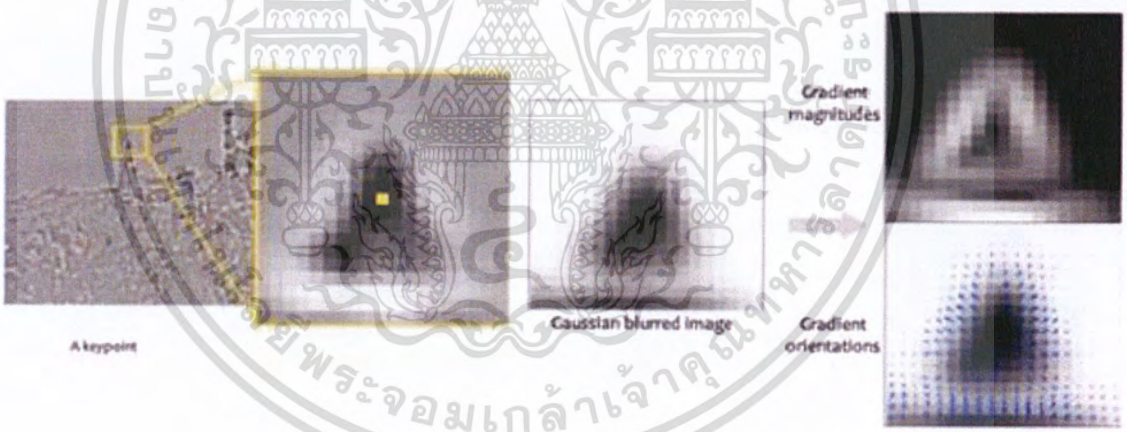
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) การกำหนดทิศทางของจุดสนใจ (Orientation Assignment) ทำการเก็บรวบรวมขนาด m(x,y) และ ทิศทาง Θ(x,y) ของ gradient ของบริเวณรอบๆ keypoints เพื่อที่จะกำหนดทิศทางให้ keypoint นั้นๆ แสดงสมการดังที่ (2.5) และ (2.6)

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \quad (2.5)$$

$$\theta(x,y) = \tan^{-1}\left(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1<y)}\right) \quad (2.6)$$

กำหนดขนาดและทิศทางเสร็จจะได้ภาพ แสดงดังรูปที่ 2.19



รูปที่ 2.19 การกำหนดขนาดและทิศทางของ keypoint

จากนั้นจะนำเอาขนาดและทิศทางของ gradient ของ pixel รอบๆ keypoint มาทำการสร้างภาพแท่งความถี่ (Histogram) ซึ่งจะมีแกน x ช่วงขององศา และแกน y เป็นขนาดของ gradient ของ pixel นั้นๆรวมกับ pixel อื่นๆที่แบ่งแยกตามแต่ละช่องของแกน x เดียวกัน ต่อมาก็จะทำการเพิ่ม keypoint โดยมีเงื่อนไขว่า ถ้า keypoint ใดมี peak ของ histogram ที่มีขนาดมากกว่า 80%ของ peakที่สูงสุด ก็จะแบ่งออกเป็น keypoint ใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) การสร้างคำอธิบายลักษณะเด่นของภาพ (Keypoint descriptor)

ทำการสร้าง 16x16 window (window คือ บริเวณที่จะทำการเก็บข้อมูลทิศทาง ซึ่งจะมีขนาดคือ 1.5*σ) รอบ keypoint และทำการแบ่งออกเป็น 4x4 windows ทั้งหมด 16 ชุด โดยในแต่ละชุดจะทำการคำนวณหาขนาดและทิศทางของ gradient แล้วนำมาสร้างเป็น histogram ที่มีขนาด 8bin (แกน x แบ่ง 8 ช่วง ช่วงละ 45 องศา) โดยที่ขนาดของ histogram ในแต่ละส่วนจะขึ้นกับ ขนาด*ตัวถ่วงน้ำหนัก (ระยะทางที่ห่างจาก keypoint) ดังนั้นเมื่อทำการคำนวณเสร็จสิ้น ผลลัพธ์ที่ได้คือ 4x4x8= 128 ซึ่งจะทำการ normalize ต่อ และจะใช้เป็น feature vector ของแต่ละ keypoint นั้นเอง

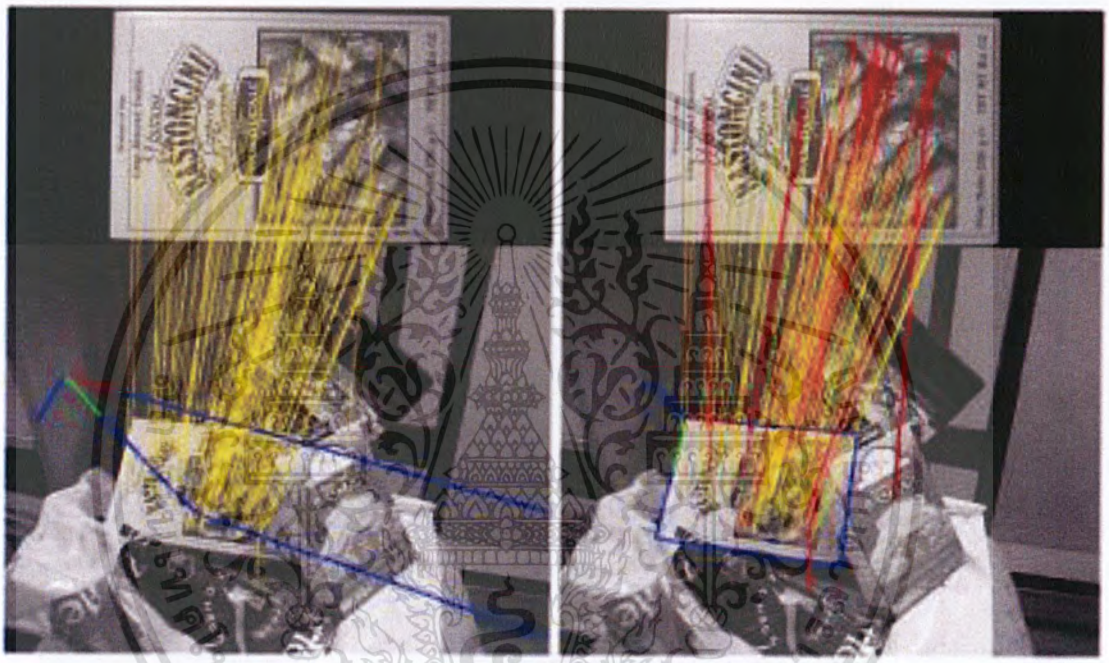
- Pose Estimation เป็นกระบวนที่นำมาใช้เพื่อทำการหาตำแหน่งและทิศทางการวางตัวของกล้อง โดยจะนำความสัมพันธ์ของจุดที่จะใช้เป็น feature point หรือ key point บนภาพถ่ายสองมิติของวัตถุชิ้นเดียวกันจากหลายๆมุมมองมาช่วยในการวิเคราะห์เพื่อคำนวณตำแหน่งและทิศทางได้อย่างถูกต้องและมีความใกล้เคียงมากที่สุด โดยในการคำนวณโมเดลของกล้องนี้ จะมีพารามิเตอร์ที่สนใจ 2 ตัวด้วยกัน ได้แก่
- Internal parameter จะแทนด้วย $K \in R^{3*3}$ โดยค่า K จะขึ้นอยู่กับคุณสมบัติของตัวกล้องโดยตรง ได้แก่ ความยาวโฟกัส คุณสมบัติของเลนส์ ซึ่งค่า K สามารถประมาณได้ด้วยการใช้เทคนิค Camera Calibration (การปรับมาตรฐานของภาพที่ได้จากกล้อง เพื่อให้ได้ภาพที่ไม่บิดเบือน) โดยจะนำมาใช้ในการเปลี่ยนจากพิกัดสามมิติของกล้องให้กลายเป็นพิกัดสองมิติบนภาพ
- External parameter จะแทนด้วย $[R|t] \epsilon R^{3*4}$ โดยค่า R จะแทนการหมุน และค่า t จะแทนการเลื่อนขนาน โดยค่าทั้งสองจะใช้แทนการเปลี่ยนจากตำแหน่งของกล้องบนพิกัดสามมิติในโลกจริงมายังพิกัดของกล้องในระบบ ซึ่งสามารถสรุปเป็นสมการของการเปลี่ยนจากพิกัดสามมิติในโลกจริง $(P \epsilon R^4)$ ให้พิกัดสองมิติบนภาพ $(P \epsilon R^3)$ แสดงสมการดังที่ (2.7)

$$p = K[R|t]P \text{ หรือ } sm' = A[R|t]M' \qquad (2.7)$$

ซึ่งจากความสัมพันธ์ของจุดที่จะใช้เป็น feature point หรือ key point นี่เอง จะใช้ความสอดคล้องของจุดเหล่านี้ $(c = \{P_i \leftrightarrow p_i\})$ โดยใช้สมการข้างต้น เพื่อทำการประมาณการตำแหน่งและทิศทางของกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยทั่วไปแล้ววิธีการหาจุดที่จะเป็น feature point หรือ key point นั้น จะมีด้วยกันหลากหลายวิธี โดยวิธีที่ค่อนข้างเป็นที่นิยม คือ SIFT และ Randomized Tree และเมื่อทำการหาความสอดคล้องได้จนครบทุกจุดในภาพแล้ว ก็อาจพบว่ามีจุดบางจุดที่เป็นจุดที่อาจจะอยู่นอกความสนใจในการพิจารณา ซึ่งมีผลทำให้เกิดความผิดพลาดในกระบวนการประมาณ ต้องทำการวิเคราะห์เพื่อคัดจุดที่อยู่นอกความสนใจออก โดยใช้วิธีการ คือ RANSAC (Random Sample Consensus) ซึ่งเป็นวิธีการทำซ้ำ (Iterative Method) จนกว่าจะได้ผลลัพธ์ที่มีความเป็นไปได้สูงสุดขึ้นกับจำนวนรอบการทำซ้ำนั่นเอง แสดงดังรูปที่ 2.20

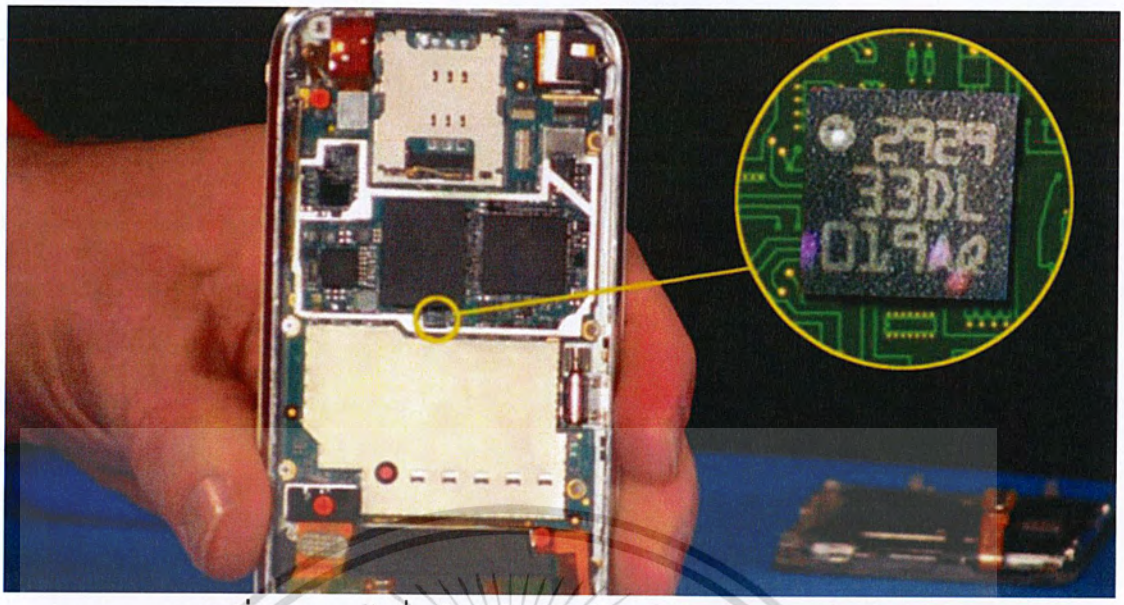


รูปที่ 2.20 ผลลัพธ์ของการหาตำแหน่งและทิศทางของการวางตัวของกล้องจากหนังสือที่ต้องการ

2.12 เซนเซอร์วัดความเร่งบนแอนดรอยด์ (Android Sensor Accelerometer)

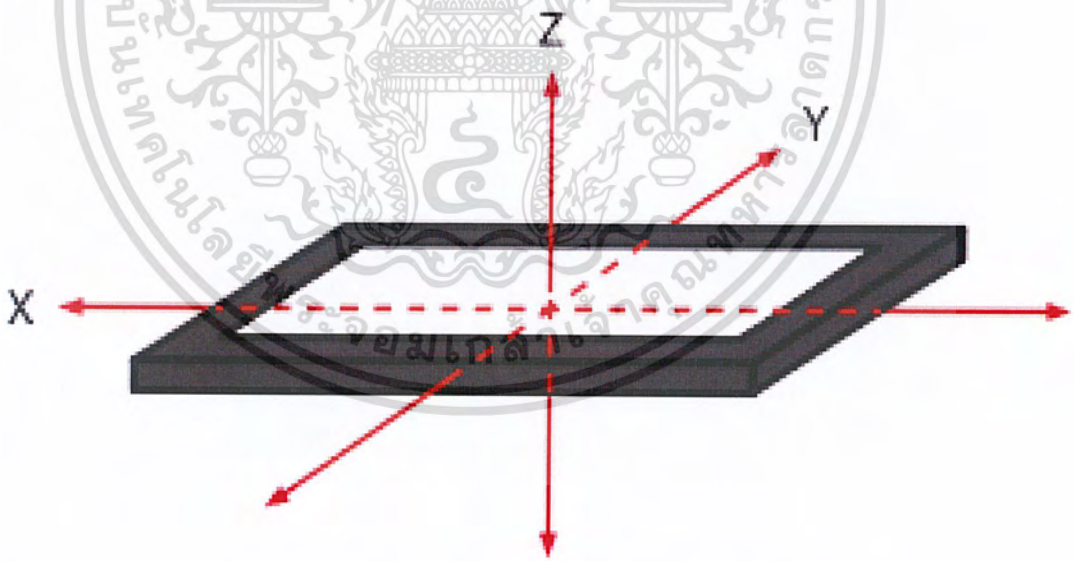
Accelerometer เป็นเซนเซอร์พื้นฐานของอุปกรณ์แอนดรอยด์กล่าวคือ แอนดรอยด์ทุกๆรุ่นจะต้องมีเซนเซอร์นี้เป็น อีกทั้งยังเป็นเซนเซอร์ที่ผู้ใช้งานส่วนใหญ่ใช้งานกัน บอร์ดที่อยู่ในตัวโทรศัพท์ Android Sensor Accelerometer แสดงดังรูปที่ 2.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 บอร์ดที่อยู่ในตัวโทรศัพท์ Android Sensor Accelerometer

Accelerometer[12] จะวัดความเร่งในการเอียงเครื่องทั้ง 3 ทิศ สำหรับแกน XYZ บนอุปกรณ์แอนดรอยด์ใดๆจะมีตำแหน่ง ดังรูปที่ 2.22



รูปที่ 2.22 ทิศทาง 3 ทิศ ในการวัดความเร่ง สำหรับแกน XYZ

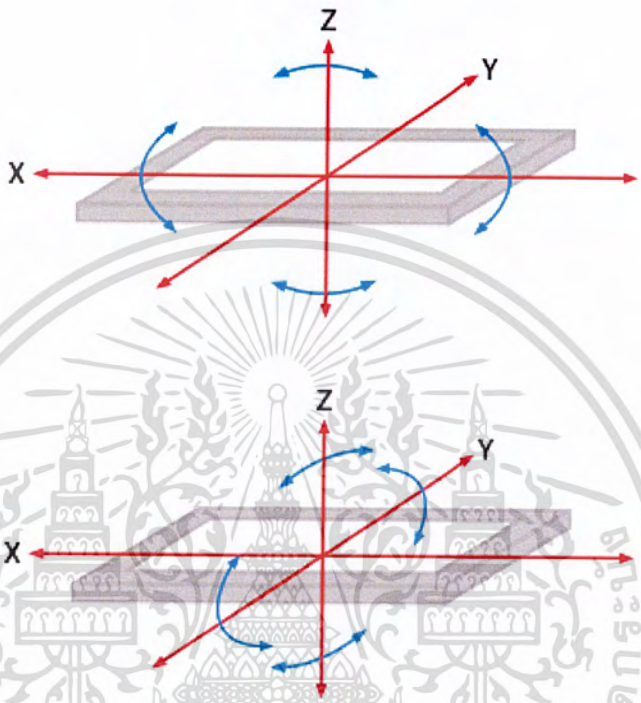
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวัดความเร่งในการเอียงก็คือการเอียงในแต่ละทิศที่จะมีลักษณะการเอียงในทิศทางต่างๆ ดังรูปที่ 2.23

รูปที่ 2.23 การวัดความเร่งในการเอียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แกน X และ Y จะมีแค่ขึ้นลงเท่านั้น แต่แกน Z จะพิเศษกว่าคือมีทั้งสองแกนที่เคลื่อนที่ ดังนั้น เวลาที่เอียงไปทางแกน X แกน Z ก็จะเอียงด้วย และเมื่อเอียงไปทางแกน Y แกน Z ก็จะเอียงตาม ดังรูปที่ 2.24

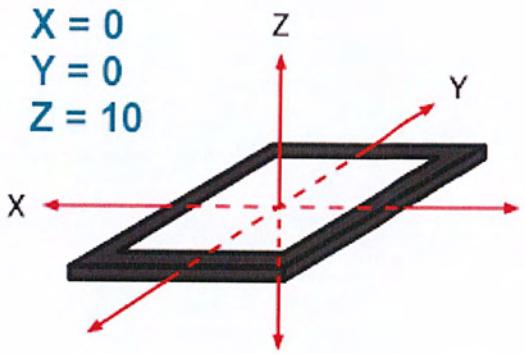


รูปที่ 2.24 การเอียงไปทางแกน X และ Y ที่มีผลต่อ แกน Z

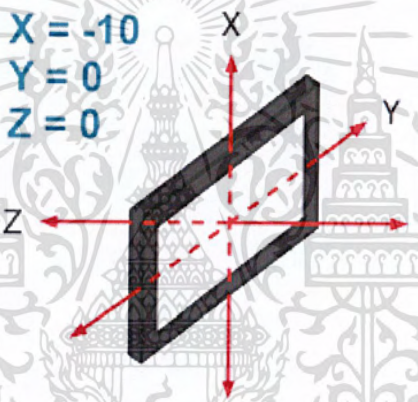
Accelerometer จะวัดความเร่งในแต่ละแกน อธิบายได้ว่า เวลาที่เครื่องอยู่นิ่ง ๆ ไม่มีการเคลื่อนไหว ค่าแต่ละแกนจะมีค่าเป็น 0 แต่เนื่องจากมีแรงโน้มถ่วงของโลก ดังนั้นจึงมีบางค่าจาก Accelerometer ที่ไม่เท่ากับ 0 ยกตัวอย่างเช่น ขณะที่ไม่มีเคลื่อนที่ ตั้งเครื่องให้แกน Z ตั้งฉากกับพื้นโลก แกน X และ Y จะเป็น 0 โดยที่แกน Z จะมีค่าไม่เท่ากับ 0 เนื่องจากมีแรงโน้มถ่วงของโลกกระทำอยู่ ดังนั้นค่าที่ได้จากแกน Z จะเป็น 9.8 m/s^2 ซึ่งเป็นค่าในอุดมคติ และไม่เท่ากับ 9.8 ตลอดเวลา ค่าจะเปลี่ยนแปลงไม่คงที่ โดยค่าเฉลี่ยจะอยู่ที่ 9.8 หรือบางกรณีจะมีประมาณค่านี้ ให้มีค่าเท่ากับ 10 เพื่อให้ง่ายต่อการนำไปคำนวณ

ค่าแกน XYZ จะเป็นไปตามทิศทางของโทรศัพท์มือถือ ดังรูปที่ 2.25 ถึงรูปที่2.27

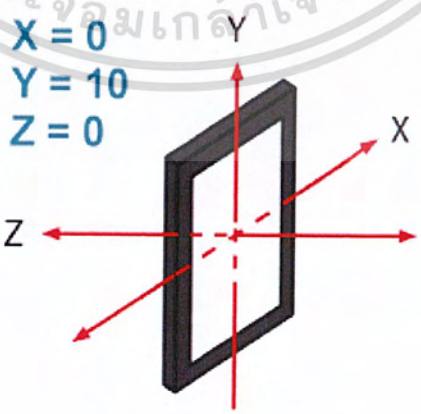
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.25 ค่าแกน XYZ วางเครื่องโทรศัพท์ในแนวหงาย หรือคว่ำ



รูปที่ 2.26 ค่าแกน XYZ วางเครื่องโทรศัพท์ตะแคงซ้าย หรือขวา



รูปที่ 2.27 ค่าแกน XYZ วางเครื่องโทรศัพท์ในแนวตั้งขึ้น หรือทิ่มหัวลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

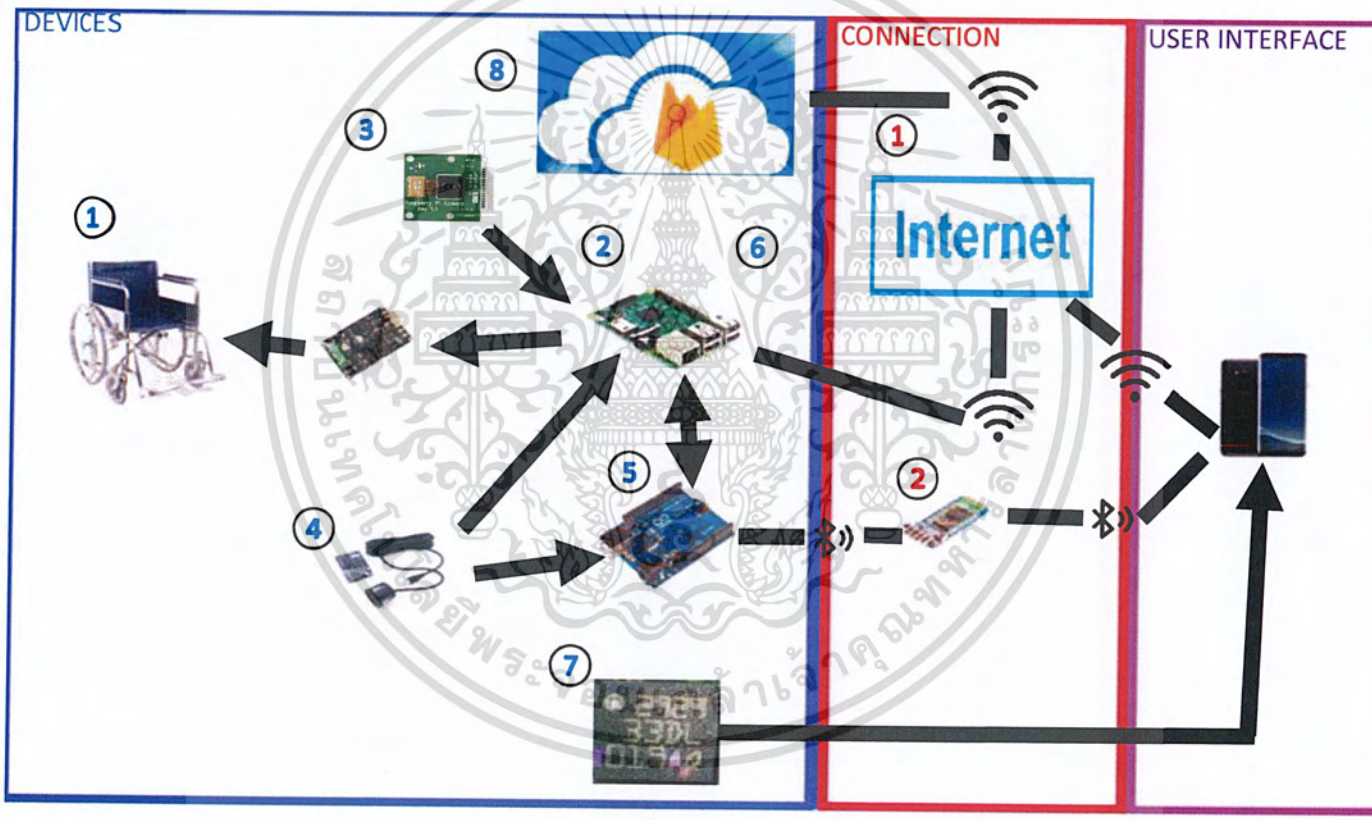
บทที่ 3

การวิเคราะห์และออกแบบระบบ

การวิเคราะห์และออกแบบระบบในปัญหาพิเศษมีรายละเอียดดังต่อไปนี้

3.1 สถาปัตยกรรมของระบบ

ระบบมีการใช้งานร่วมกันระหว่างอุปกรณ์สมาร์ทโฟนระบบปฏิบัติการแอนดรอยด์ และรถนั่งโดยสถาปัตยกรรมของระบบมีลักษณะดังรูปที่ 3.1



รูปที่ 3.1 สถาปัตยกรรมของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างและส่วนประกอบต่าง ๆ มีรายละเอียดดังต่อไปนี้

3.1.1 Devices

1) รถนั่ง: ทำหน้าที่ในการเคลื่อนที่ตามคำสั่งที่ได้รับ
2) Control motor: เป็นตัวควบคุมมอเตอร์โดยค่าที่ได้รับจาก Raspberry pi
3) Camera Serial Interface: เชื่อมต่อกับ Raspberry Pi ทำหน้าที่ในการจับภาพด้านหน้าเพื่อนำภาพไปประมวลผลแบบเรียลไทม์ และได้ตัวแปรในการประมวลผลภาพส่งไปยัง Raspberry pi
4) เซนเซอร์วัดระยะ: เป็นตัววัดระยะระหว่างวัตถุเพื่อส่งค่าไปยัง Raspberry pi และ Arduino เพื่อป้องกันอุบัติเหตุ
5) Arduino: ทำหน้าที่รับค่าจาก Accelerometer และส่งคำสั่งส่งเสียงเตือน โดยส่งผ่าน Bluetooth Module
6) Raspberry pi: ทำหน้าที่เสมือน Central Processing Unit (CPU) เป็นตัวกลางระหว่างผู้ใช้กับรถนั่ง ใช้ในการควบคุมการทำงานของรถนั่ง เชื่อมต่อกับ Bluetooth Dongle เพื่อส่งสัญญาณควบคุมรถเข็น และจะเป็น Raspberry Pi ใช้ กล้องเป็นตัวประมวลผลภาพ ซึ่งทำหน้าที่ตรวจจับภาพเพื่อส่งค่าไปยัง Arduino
7) Android sensor accelerometer: เซนเซอร์วัดความเร่งทำหน้าที่ในการตรวจสอบสถานการณ์ล้มของรถนั่งโดยส่งค่าที่ได้ให้สมาร์ทโฟนทำการโทรออกฉุกเฉิน
8) Firebase: ทำหน้าที่เป็นตัวเก็บข้อมูลในการขับเคลื่อนอัตโนมัติไปยังสถานที่ต่าง ๆ

3.1.2 Connection

1) Internet: ทั้งสมาร์ทโฟน และ Raspberry pi ต้องเชื่อมต่อผ่านอินเทอร์เน็ตเดียวกัน เพื่อให้สมาร์ทโฟน และราสเบอร์รี่พายสามารถติดต่อสื่อสารกันภายในวง LAN ได้
2) Bluetooth Module: ทำหน้าที่เชื่อมต่อระหว่าง Arduino กับสมาร์ทโฟน

3.1.3 User interface

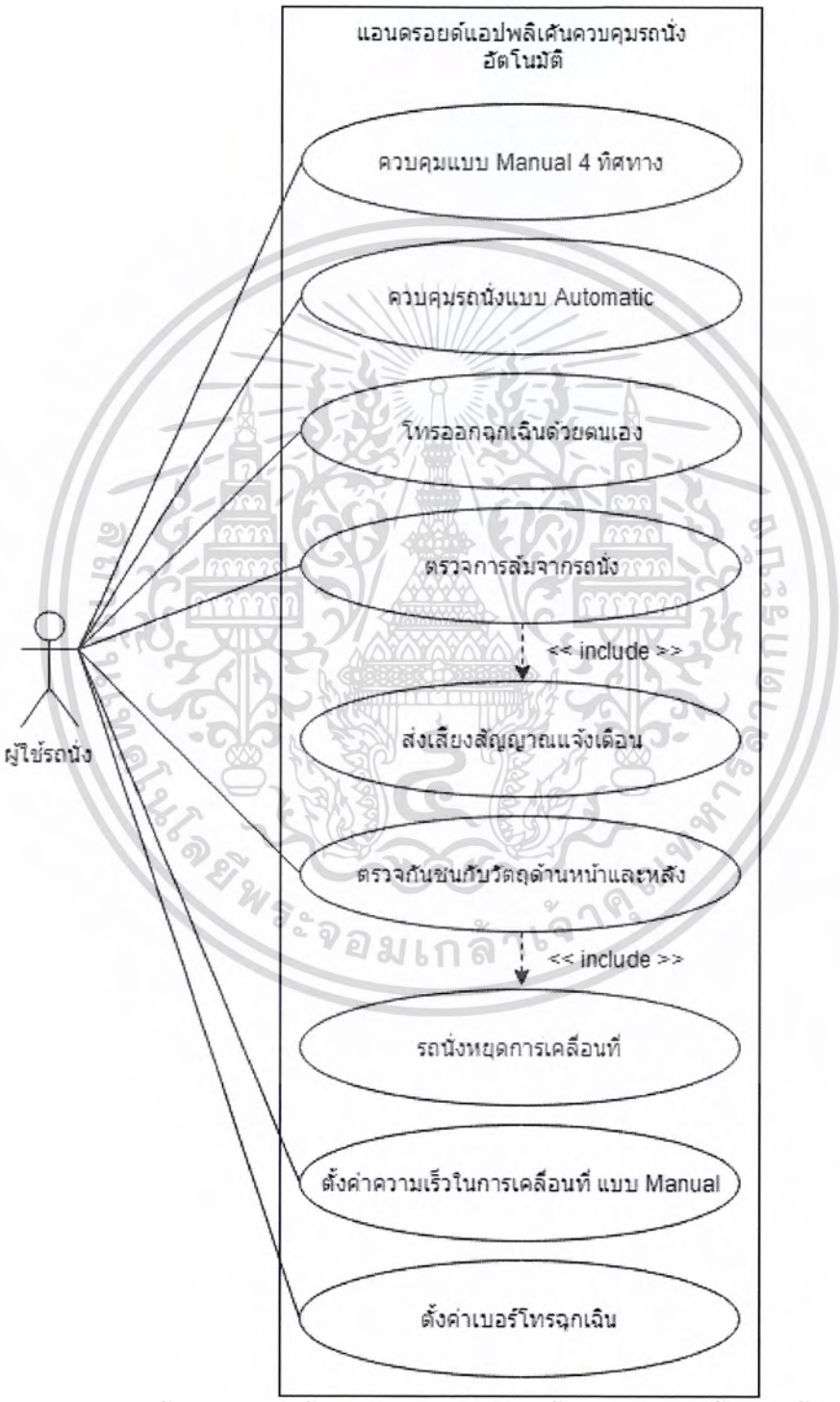
Smartphone: ทำหน้าที่ส่งคำสั่งควบคุมรถนั่งผ่านทางอินเทอร์เน็ตไปยัง Raspberry Pi

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบระบบ

3.2.1 แผนภาพยูสเคส (Use Case Diagram)

แผนภาพยูสเคสของแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ แสดงดังรูปที่ 3.2



รูปที่ 3.2 Use Case Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางแสดงคำอธิบายของผู้ใช้ระบบ (Use Case Description) แสดงได้ดังตารางที่ 3.1 ถึง ตารางที่ 3.7

ตารางที่ 3.1 อธิบาย Use Case ควบคุมแบบ Manual 4 ทิศทาง

หัวข้อ	คำอธิบาย
Use Case Name:	ควบคุมแบบ Manual 4 ทิศทาง
Use-Case ID:	01
Priority:	High
Actor:	user
Description:	สำหรับการควบคุมรถนั่ง 4 ทิศทาง
Pre-Conditions:	เชื่อมต่อ Bluetooth
Trigger:	กดปุ่ม ON และ เลือกทิศทาง
Post-Conditions:	รถนั่งสามารถเคลื่อนที่ตามทิศทางที่เลือก
Flow:	1) กรอกข้อมูลน้ำหนักผู้ใช้ 2) เชื่อมต่อ Bluetooth 3) เลือกทิศทางที่ต้องการจะเคลื่อนที่

ตารางที่ 3.2 อธิบาย Use Case ควบคุมรถนั่งแบบ Automatic

หัวข้อ	คำอธิบาย
Use Case Name:	ควบคุมรถนั่งแบบ Automatic
Use-Case ID:	02
Priority:	High
Actor:	user
Description:	สำหรับเลือกตำแหน่งในโรงพยาบาลเพื่อให้รถนั่งเคลื่อนที่ไปยังตำแหน่งนั้น ๆ
Pre-Conditions:	เชื่อมต่อ Internet
Trigger:	เลือกปุ่มตำแหน่งที่ต้องการจะไป
Post-Conditions:	รถนั่งสามารถเคลื่อนที่อัตโนมัติไปตามตำแหน่งที่เลือก
Flow:	1) กรอกข้อมูลน้ำหนักผู้ใช้ 2) เชื่อมต่อ Bluetooth 3) เลือกทิศทางที่ต้องการจะเคลื่อนที่เพื่อประมวลผลป้าย 4) เลือกปุ่ม Go Auto

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 อธิบาย Use Case โทรออกฉุกเฉินด้วยตนเอง

หัวข้อ	คำอธิบาย
Use Case Name:	โทรออกฉุกเฉินด้วยตนเอง
Use-Case ID:	03
Priority:	High
Actor:	user
Description:	สำหรับเลือกปุ่มโทรฉุกเฉินในกรณีที่เกิดเหตุการณ์ฉุกเฉิน
Pre-Conditions:	เลือกปุ่ม Setting เพื่อทำการตั้งค่าเบอร์โทรที่จะต้องโทรฉุกเฉิน
Trigger:	เมื่อเกิดเหตุการณ์ฉุกเฉิน
Post-Conditions:	รถนั่งสามารถเคลื่อนที่อัตโนมัติไปตามตำแหน่งที่เลือก
Flow:	1) กรอกข้อมูลน้ำหนักผู้ใช้งาน 2) เชื่อมต่อ Bluetooth 3) เลือกปุ่มรูปโทรศัพท์สีแดงเพื่อโทรฉุกเฉิน

ตารางที่ 3.4 อธิบาย Use Case ตรวจการล้มจากรถนั่ง

หัวข้อ	คำอธิบาย
Use Case Name:	ตรวจการล้มจากรถนั่ง
Use-Case ID:	04
Priority:	High
Actor:	user
Description:	สำหรับโทรฉุกเฉินในกรณีที่เกิดอุบัติเหตุรถนั่งล้ม
Pre-Conditions:	เลือกปุ่ม Setting เพื่อทำการตั้งค่าเบอร์โทรที่จะต้องโทรฉุกเฉิน
Trigger:	เมื่อเกิดเหตุอุบัติเหตุรถนั่งล้ม
Post-Conditions:	โทรออกฉุกเฉินเมื่อเกิดอุบัติเหตุรถนั่งล้ม
Flow:	1) กรอกข้อมูลน้ำหนักผู้ใช้งาน 2) เชื่อมต่อ Bluetooth 3) ตรวจล้มหน้าManual

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 อธิบาย Use Case ตรวจกันชนกับวัตถุด้านหน้าและด้านหลัง

หัวข้อ	คำอธิบาย
Use Case Name:	ตรวจกันชนกับวัตถุด้านหน้าและด้านหลัง
Use-Case ID:	05
Priority:	High
Actor:	user
Description:	สำหรับกันชนในโหมด Manual และ Auto
Pre-Conditions:	เชื่อมต่อ Bluetooth
Trigger:	เมื่อมีสิ่งกีดขวาง
Post-Conditions:	หยุดรถนั่งเมื่อมีสิ่งกีดขวาง
Flow:	1) กรอกข้อมูลน้ำหนักผู้ใช้งาน 2) เชื่อมต่อ Bluetooth 3) ตรวจล้มหน้า Manual

ตารางที่ 3.6 อธิบาย Use Case ตั้งค่าความเร็วในการเคลื่อนที่โหมด Manual

หัวข้อ	คำอธิบาย
Use Case Name:	ตั้งค่าความเร็วในการเคลื่อนที่โหมด Manual
Use-Case ID:	06
Priority:	High
Actor:	user
Description:	สำหรับผู้ใช้ที่ต้องการความเร็วในการเคลื่อนที่โหมด Manual เพิ่ม
Pre-Conditions:	เชื่อมต่อ Bluetooth
Trigger:	เมื่อผู้ใช้ต้องการเพิ่มความเร็ว
Post-Conditions:	ผู้ใช้ต้องการเพิ่มความเร็ว
Flow:	1) กรอกข้อมูลน้ำหนักผู้ใช้งาน 2) เชื่อมต่อ Bluetooth 3) เลือกปุ่ม แฮมเบอร์เกอร์ 4) เลือกปุ่ม Setting 5) เลือก ความเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.7 อธิบาย Use Case ตั้งค่าเบอร์โทรผู้ดูแล

หัวข้อ	คำอธิบาย
Use Case Name:	ตั้งค่าเบอร์โทรผู้ดูแล
Use-Case ID:	07
Priority:	High
Actor:	user
Description:	สำหรับโทรฉุกเฉิน และตรวจล้ม
Pre-Conditions:	เชื่อมต่อ Bluetooth
Trigger:	ผู้ใช้ต้องการเปลี่ยนเบอร์ผู้ดูแล
Post-Conditions:	เปลี่ยนเบอร์ผู้ดูแล
Flow:	1) กรอกข้อมูลน้ำหนักผู้ใช้งาน 2) เชื่อมต่อ Bluetooth 3) เลือกปุ่ม Setting 4) กรอกเบอร์ผู้ดูแล

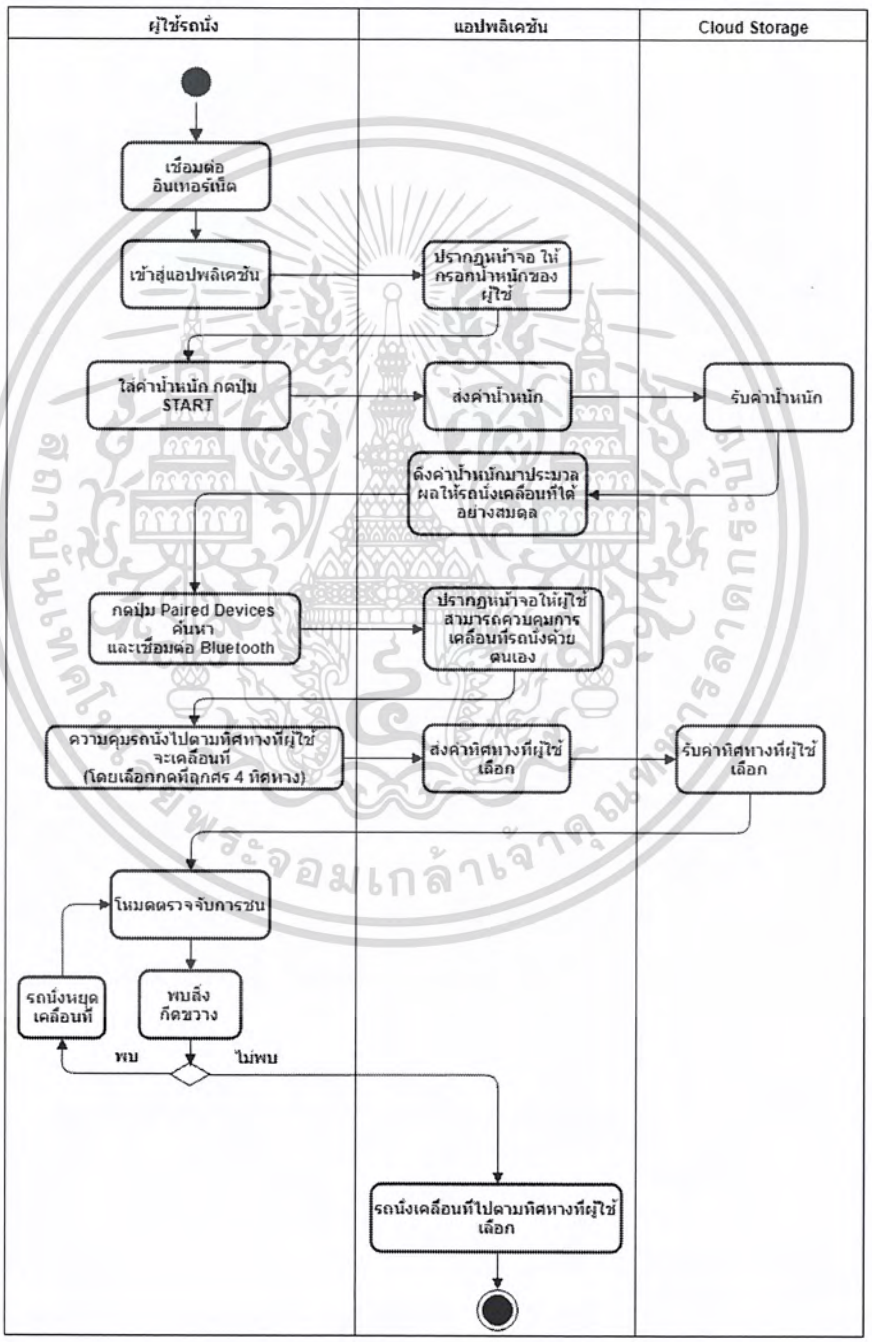
3.2.2 แผนภาพแอคทิวิตี้ไดอะแกรม (Activity Diagram)

เป็นแผนภาพแสดงขั้นตอนการทำงาน Use Case แต่ละขั้นตอนของรถนั่งภายในโรงพยาบาลสำหรับผู้สูงอายุและผู้พิการทางขาควบคุมผ่านแอนดรอยด์แอปพลิเคชัน มีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) การควบคุมการเคลื่อนที่รถนั่ง

แผนภาพ Activity Diagram แสดงขั้นตอนการเข้าสู่ระบบ และควบคุมรถนั่งผ่านแอปพลิเคชัน SmartCar โดยผู้ใช้รถนั่งต้องทำการเชื่อมต่อกับอินเทอร์เน็ตแล้วเข้าสู่แอปพลิเคชันใส่ค่าน้ำหนักของผู้ใช้งานเพื่ออ้างอิงความเร็วเริ่มต้นแล้วเลิกปุ่ม START และทำการเชื่อมต่อ Bluetooth จึงจะสามารถใช้งานโหมด ควบคุม 4 ทิศทางได้ แสดงดังรูปที่ 3.3

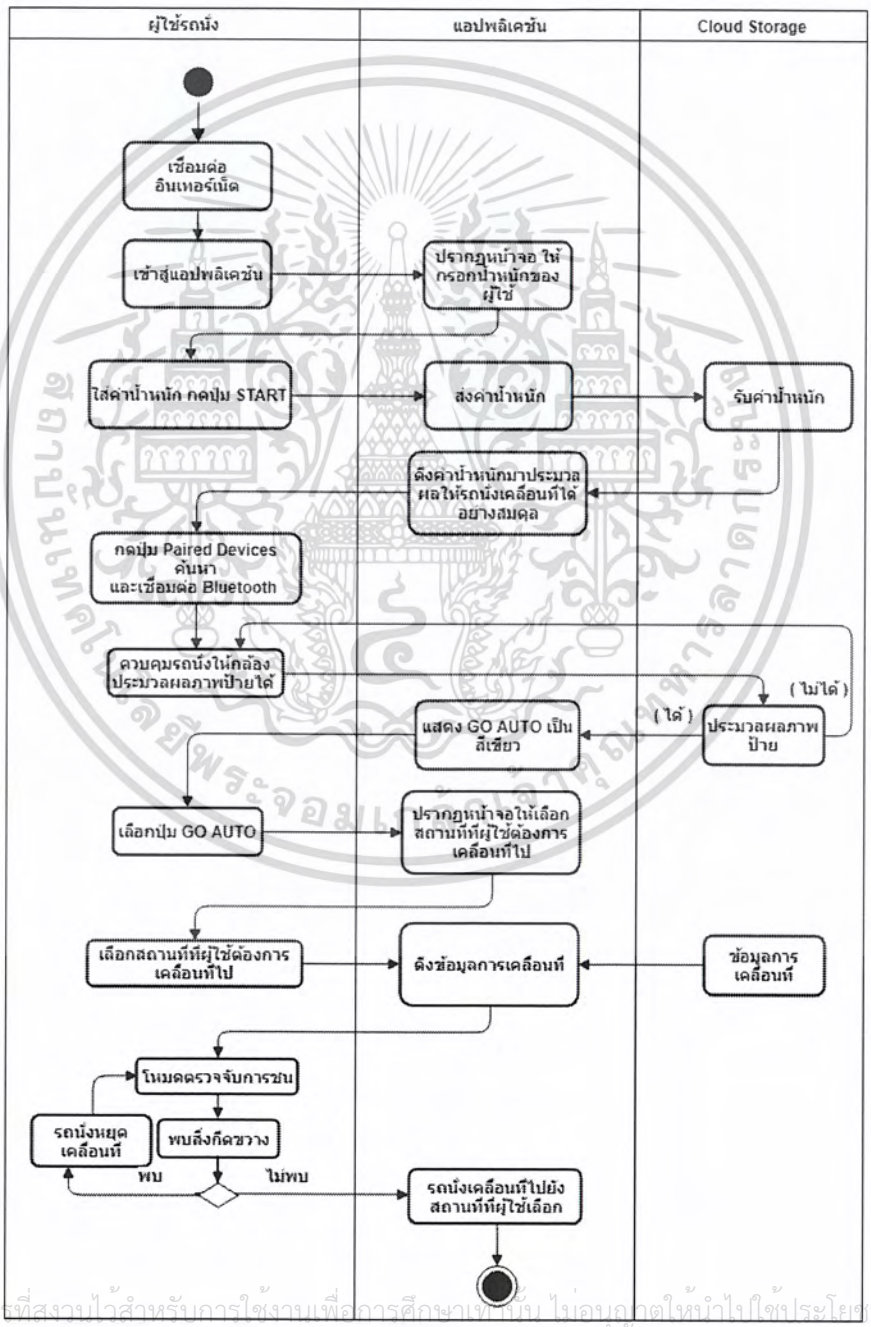


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.3 แผนภาพ Activity Diagram การควบคุมการเคลื่อนที่รถนั่ง

2) การควบคุมการเคลื่อนที่แบบอัตโนมัติ

แผนภาพ Activity Diagram แสดงการควบคุมรถนั่งแบบอัตโนมัติทำการเชื่อมต่อกับอินเทอร์เน็ตแล้วเข้าสู่แอปพลิเคชันใส่ค่าน้ำหนักของผู้ใช่งานเพื่ออ้างอิงความเร็วเริ่มต้นแล้วเลิกปุ่ม START และทำการเชื่อมต่อ Bluetooth และทำการควบคุมโหมด Manual ควบคุมทิศทางไปยังป้ายเพื่อประมวลผลภาพ และกดปุ่ม Go AUTO เพื่อทำการเลือกตำแหน่งที่ผุ้ใช้ต้องการจะไปแสดงดังรูปที่ 3.4

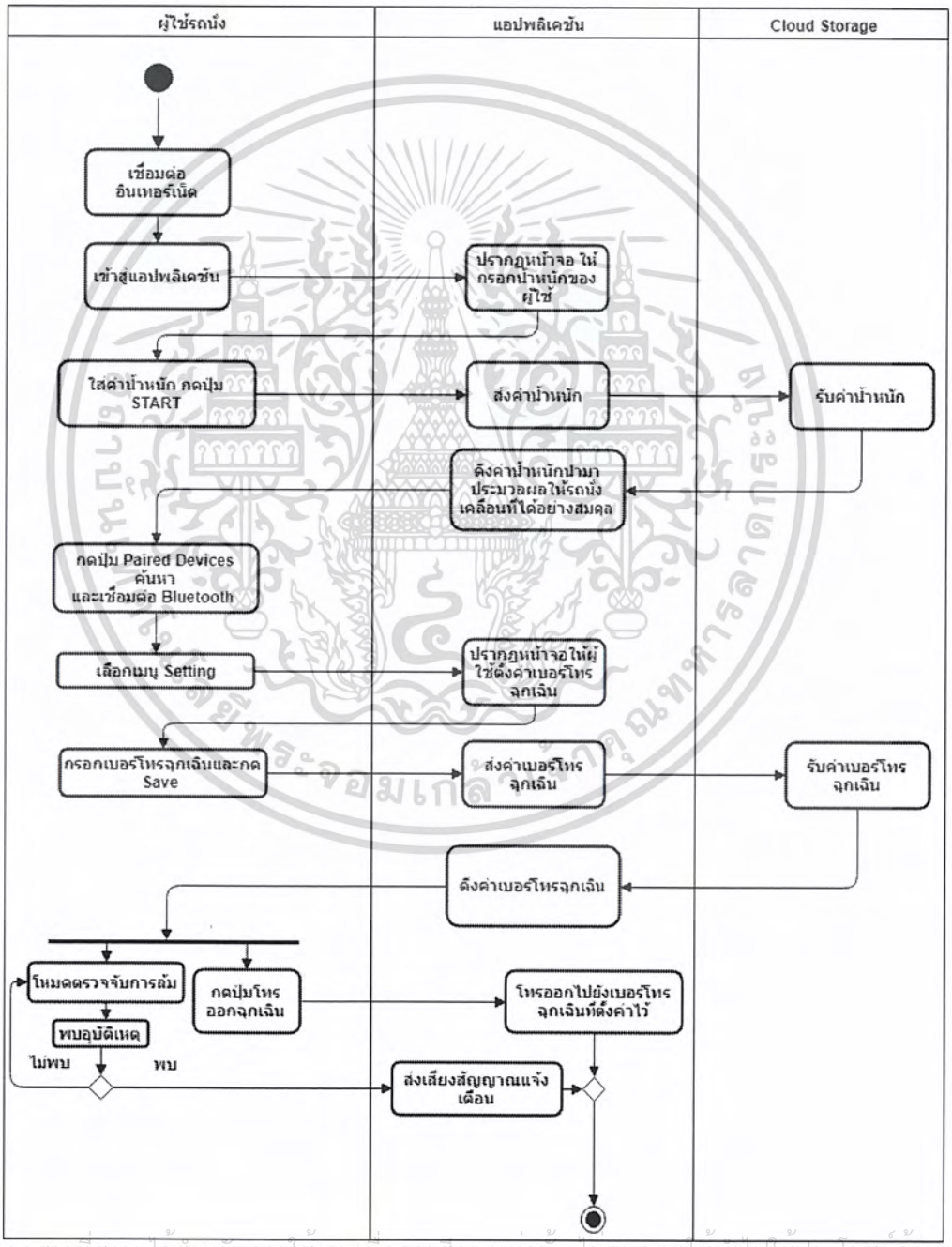


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.4 แผนภาพ Activity Diagram การควบคุมเคลื่อนที่แบบอัตโนมัติ

3) การตรวจล้ม และโทรออกฉุกเฉิน

แผนภาพ Activity Diagram แสดงการตรวจจับการล้มรถนั่ง และการโทรออกฉุกเฉิน โดยผู้ใช้งานเชื่อมต่ออินเทอร์เน็ตเข้าสู่แอปพลิเคชันแล้วใส่ค่าน้ำหนักของผู้ใช้งานแล้วจึงเลือกปุ่ม START และทำการเชื่อมต่อ Bluetooth แล้วเลือกปุ่ม แฮมเบอร์เกอร์ เพื่อนำแถบเมนูออกมาแล้วเลือกปุ่ม Setting เพื่อเพิ่มเบอร์ผู้ดูแลจึงจะสามารถใช้ระบบโทรออกฉุกเฉินและตรวจล้มได้ดังรูปที่ 3.5



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น

รูปที่ 3.5 แผนภาพ Activity Diagram การตรวจล้ม และโทรออกฉุกเฉิน

ที่มีการนำไปใช้

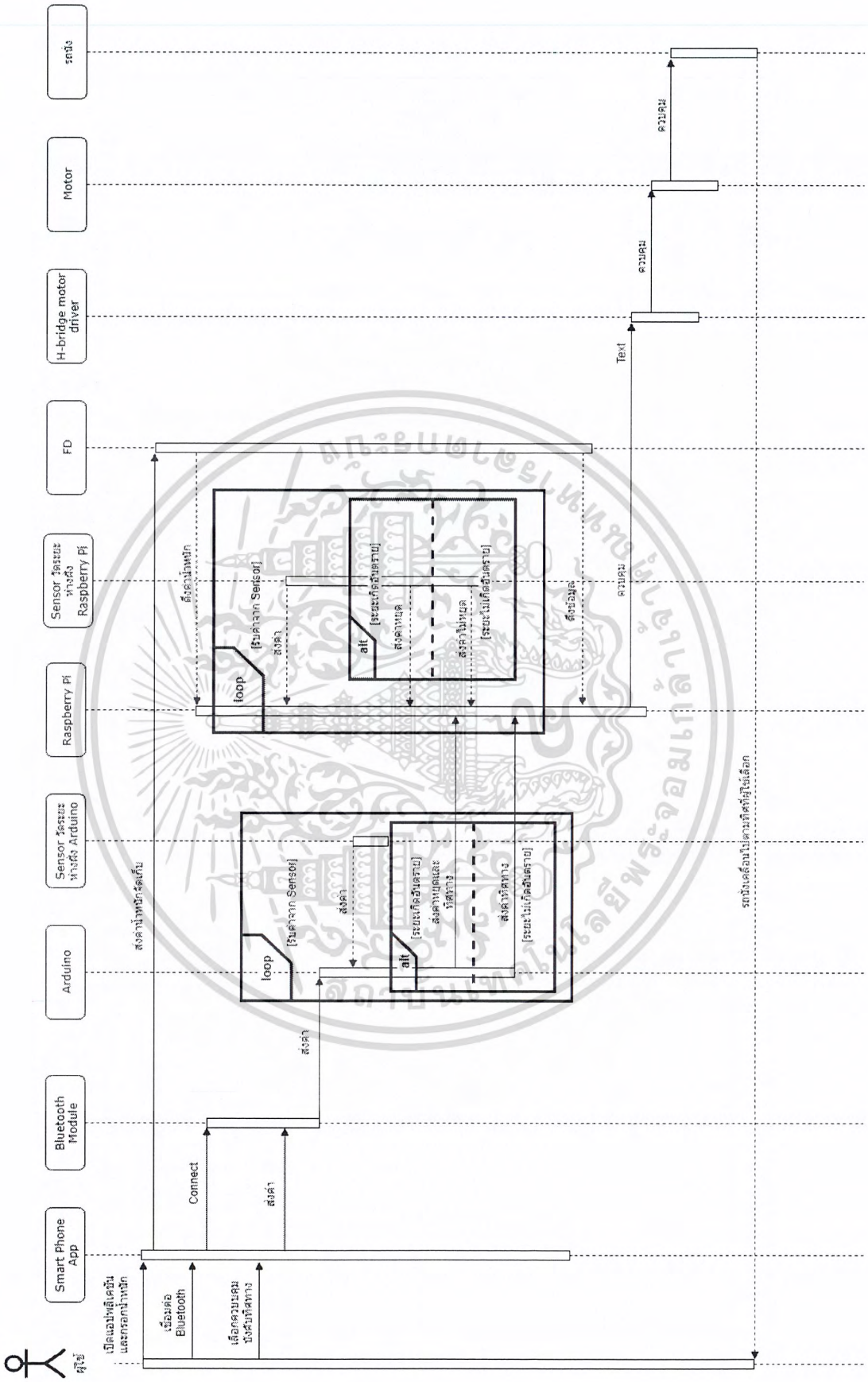
3.2.3 ขั้นตอนการทำงานของระบบ (Sequence Diagram)

ขั้นตอนการทำงานของระบบแบ่งออกเป็น 4 ฟังก์ชัน ได้แก่

1) การควบคุมการเคลื่อนที่รถนั่ง

การควบคุมการเคลื่อนที่รถนั่งโดยใช้แอปพลิเคชันในสมาร์ทโฟน ระบบปฏิบัติการแอนดรอยด์ มีขั้นตอนการทำงานโดยเริ่มจากการเปิดสวิตช์ของรถนั่ง และสั่งการเคลื่อนที่ผ่านแอปพลิเคชันบนสมาร์ทโฟนโดยทำการเขียนคำสั่งไว้บน Arduino รอรับคำสั่งจากสมาร์ทโฟน เมื่อได้รับคำสั่ง Raspberry Pi จะจ่ายกระแสไฟให้ H-Bridge เพื่อขับเคลื่อนมอเตอร์ แสดงใน Sequence Diagram ดังรูปที่ 3.6

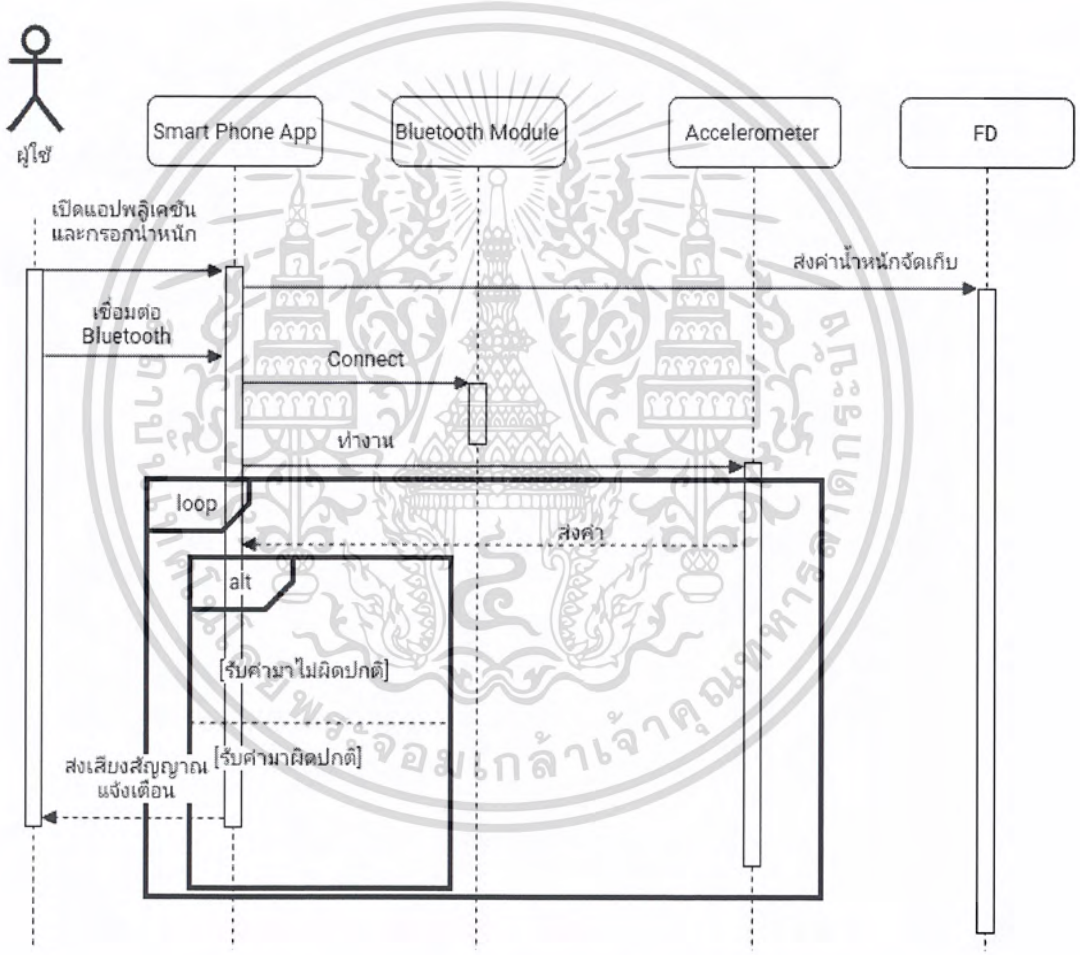
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 Sequence Diagram ของฟังก์ชันควบคุมการเคลื่อนที่รถนั่ง

2) การตรวจจับการล้ม

การตรวจจับการล้มโดยใช้เซนเซอร์วัดความเร่งมีขั้นตอนการทำงานโดยเริ่มจากเปิดสวิตช์รถนั่ง ส่วน Raspberry Pi ทำหน้าที่เชื่อมต่อเพื่อจ่ายกระแสไฟฟ้าให้ Arduino ซึ่ง Arduino ทำการเชื่อมต่อโดยตรงกับวัดความเร่งในโทรศัพท์ ซึ่งจะทำการวัดค่าแกนของรถนั่งตลอดเวลา ในกรณีที่รถนั่งล้ม Arduino จะส่งการแจ้งเตือนไปที่แอปพลิเคชันบนสมาร์ทโฟนผ่านบลูทูธ เพื่อทำการโทรออกไปยังหมายเลขฉุกเฉินที่ตั้งค่าไว้เพื่อรับความช่วยเหลือต่อไปแสดงใน Sequence Diagram ดังรูปที่ 3.7



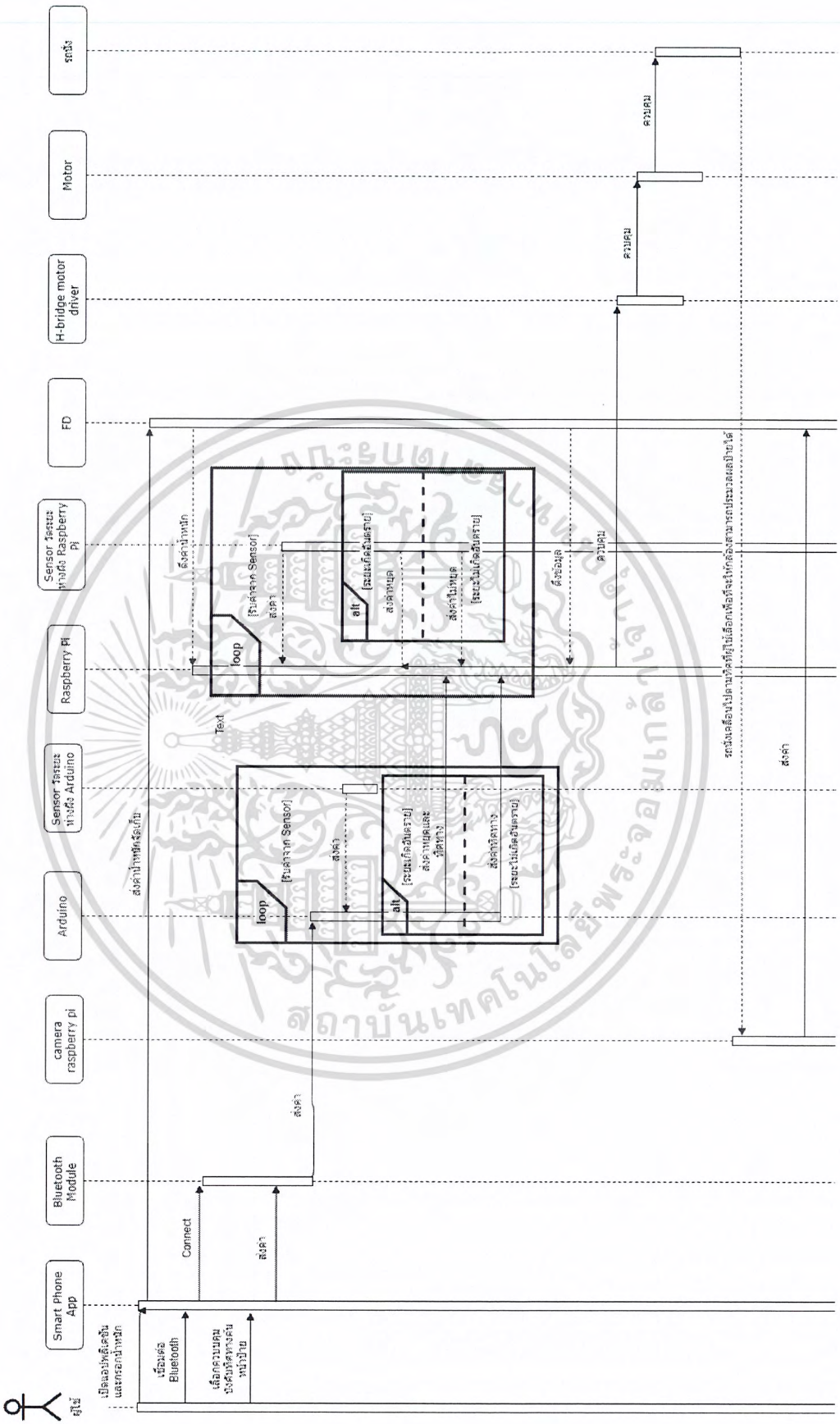
รูปที่ 3.7 Sequence Diagram ของฟังก์ชันตรวจจับการล้ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

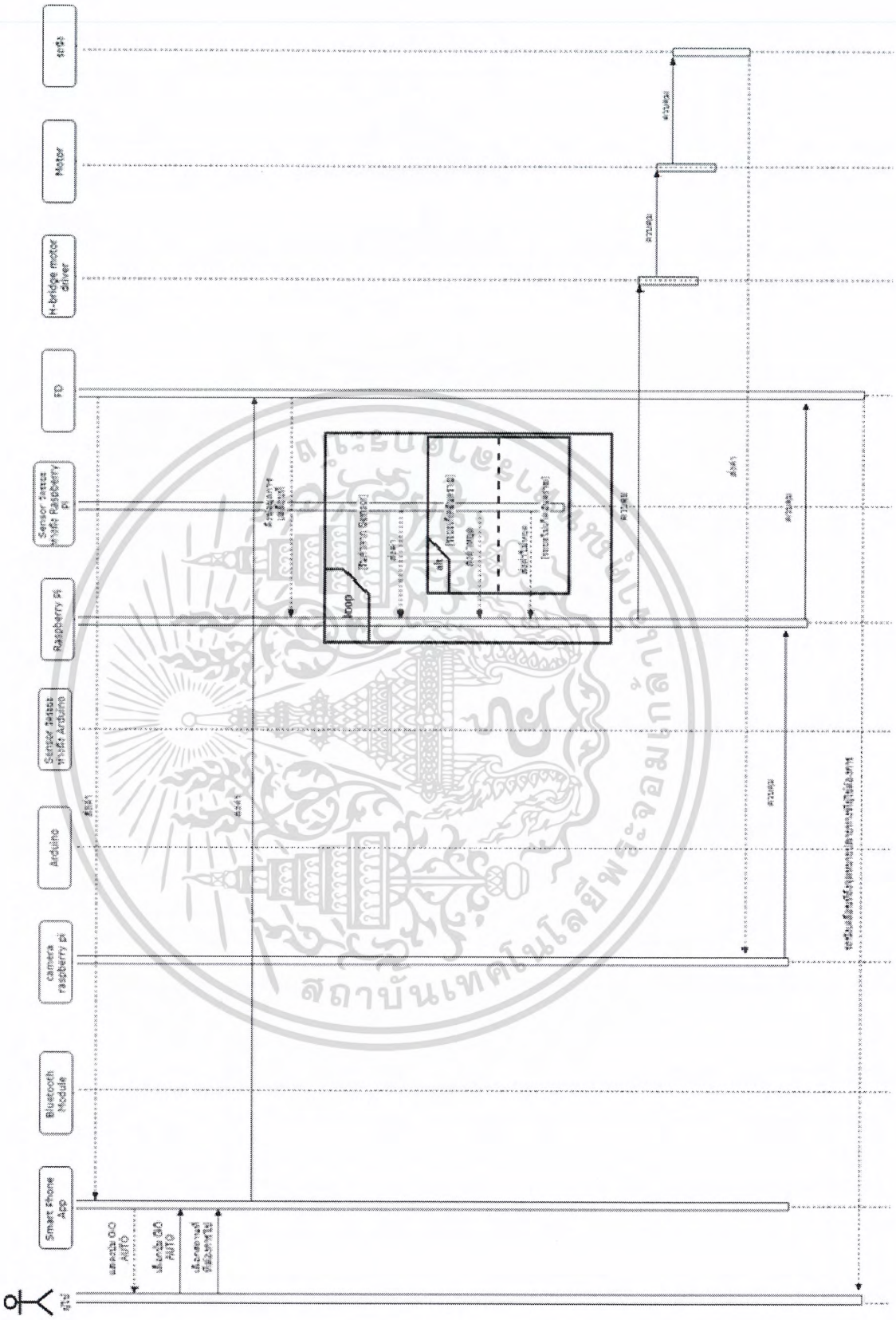
3) การควบคุมการเคลื่อนรถเข็นแบบอัตโนมัติ

การควบคุมการเคลื่อนที่รถนั่งแบบอัตโนมัติโดยใช้แอปพลิเคชันในสมาร์ทโฟน ระบบปฏิบัติการแอนดรอยด์ มีขั้นตอนการทำงานโดยเริ่มจากการเปิดสวิตช์ของรถนั่ง และเลือกสถานที่ที่จะไปผ่านแอปพลิเคชันบนสมาร์ทโฟนโดยทำการเขียนคำสั่งไว้บน Arduino รอรับคำสั่งจากสมาร์ทโฟน เมื่อได้รับคำสั่ง Raspberry Pi จะจ่ายกระแสไฟให้ H-Bridge เพื่อขับเคลื่อนมอเตอร์โดยใช้การประมวลผลภาพสร้างปุ่มขึ้นมา แสดงใน Sequence Diagram ดังรูปที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



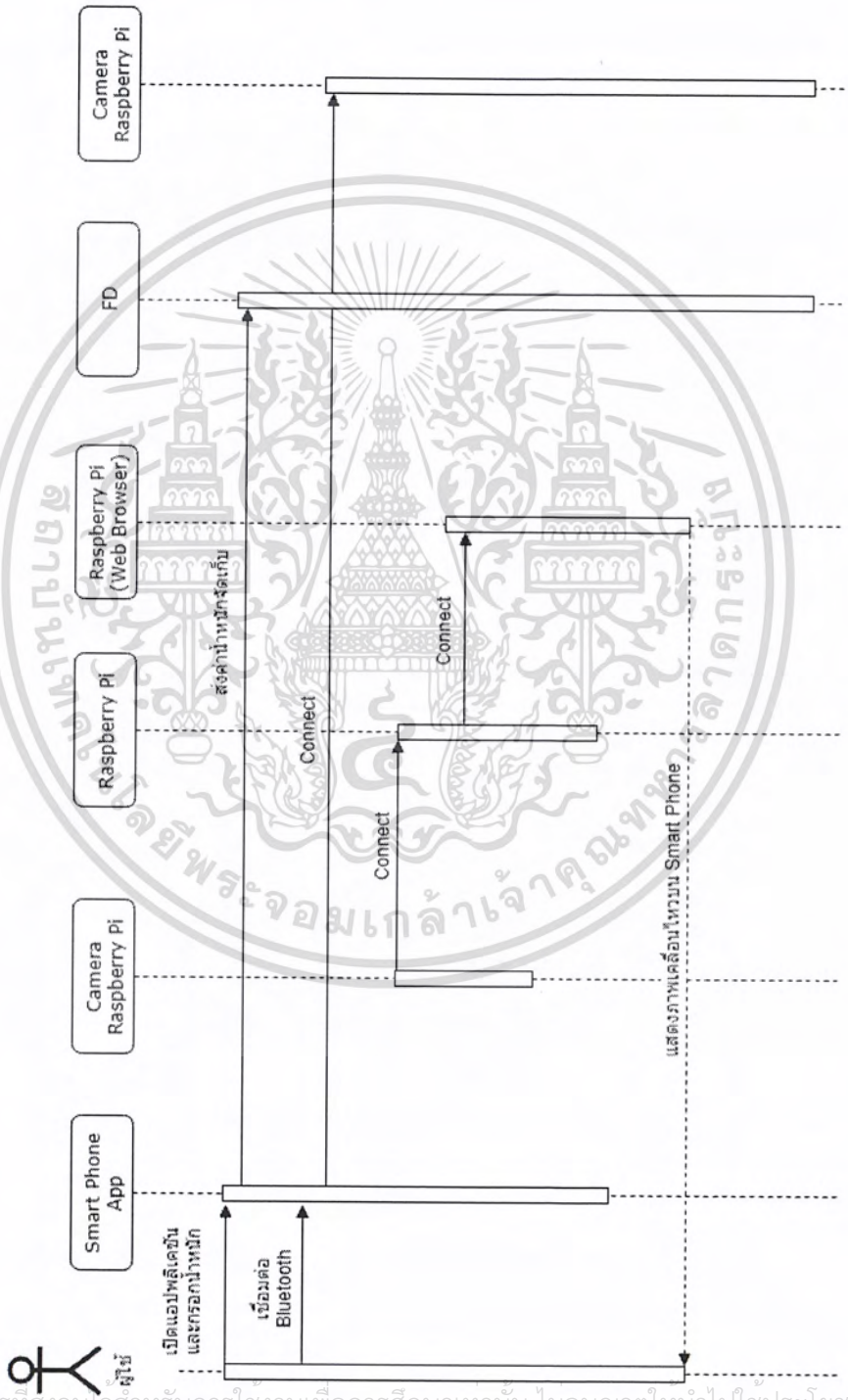
รูปที่ 3.8 Sequence Diagram ของฟังก์ชันควบคุมการเคลื่อนรถเข็นแบบอัตโนมัติ



รูปที่ 3.8 (ต่อ) Sequence Diagram ของฟังก์ชันควบคุมการเคลื่อนรถเข็นแบบอัตโนมัติ

4) ฟังก์ชันกล้องแสดงผลภาพ

การแสดงผลภาพจะต้องเชื่อมต่อ Raspberry pi เพื่อส่งภาพขึ้นเว็บเบราว์เซอร์ และแอปพลิเคชันสามารถเรียกดูผ่านเว็บเบราว์เซอร์ได้จึงทำให้ผู้ใช้สามารถเห็นภาพที่ปรากฏในแอปพลิเคชันได้ แสดงใน Sequence Diagram ดังรูปที่ 3.9

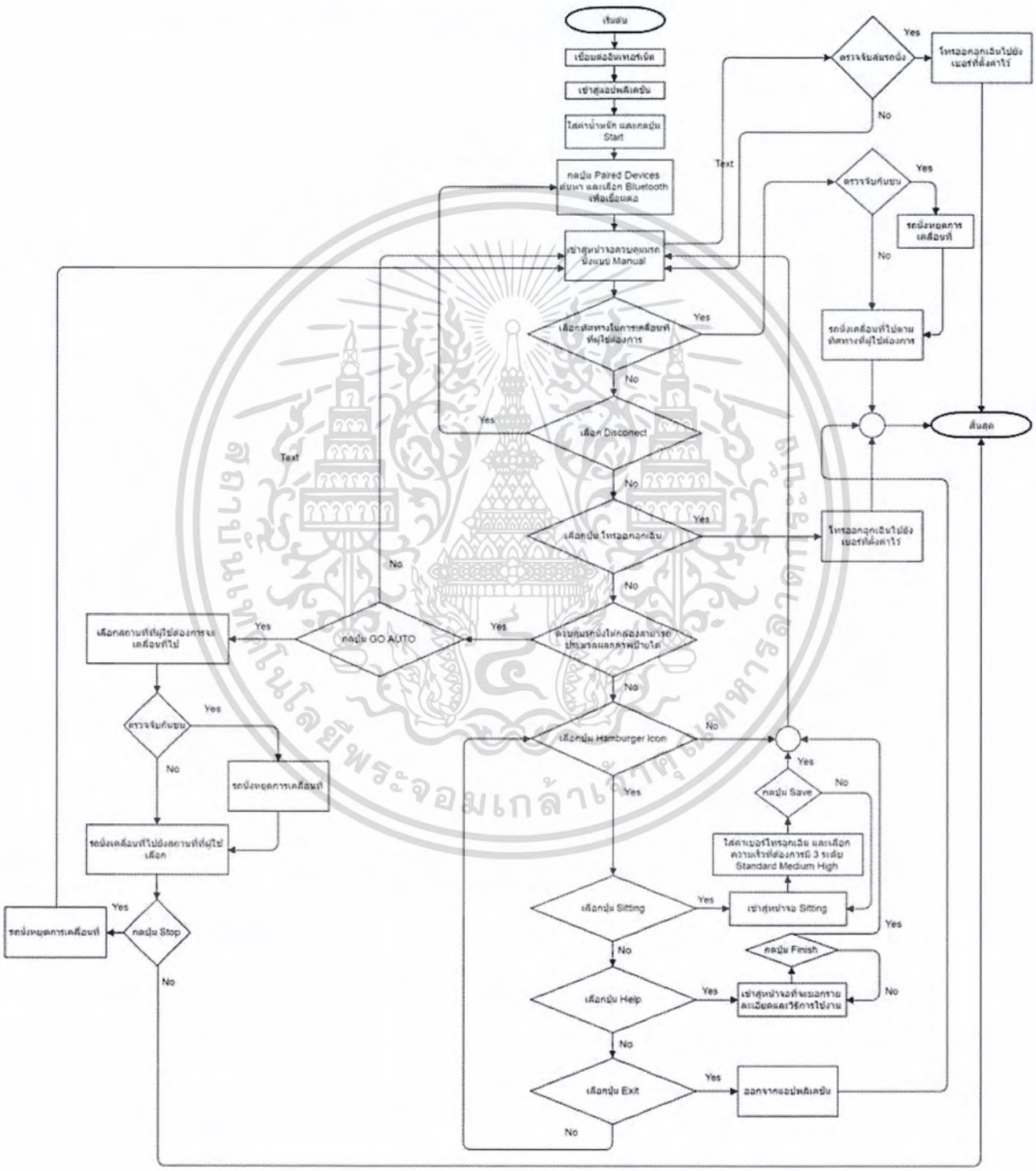


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.9 Sequence Diagram ของฟังก์ชันกล้องแสดงผลภาพ

3.2.4 แผนผังการทำงานของระบบ (Flow Chart)

เป็นแผนผังการทำงานของระบบ Flow Chart ของระบบทั้งหมดและฟังก์ชันต่าง ๆ ประกอบด้วยฟังก์ชันควบคุมการเคลื่อนที่รถนั่ง ฟังก์ชันการตรวจล้ม ฟังก์ชันควบคุมการเคลื่อนที่แบบอัตโนมัติและโทรออกฉุกเฉิน แสดงดังรูปที่ดังรูปที่ 3.10



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.10 แผนผังการทำงานของรถนั่งควบคุมด้วยแอปพลิเคชันบนแอนดรอยด์

3.3 ส่วนติดต่อประสานกับผู้ใช้ (User Interface)

การออกแบบส่วนติดต่อประสานงานกับผู้ใช้แบ่งออกเป็น 2 ส่วน คือการติดต่อผ่านแอปพลิเคชัน และการติดต่อผ่านรถนั่ง

การติดต่อผ่านแอปพลิเคชันกับผู้ใช้มีส่วนที่จำเป็นสำหรับการใช้งาน โดยหน้าจอหลักแสดงดังรูปที่ 3.11

รูปที่ 3.11 หน้าจอเข้าสู่ระบบ HOME

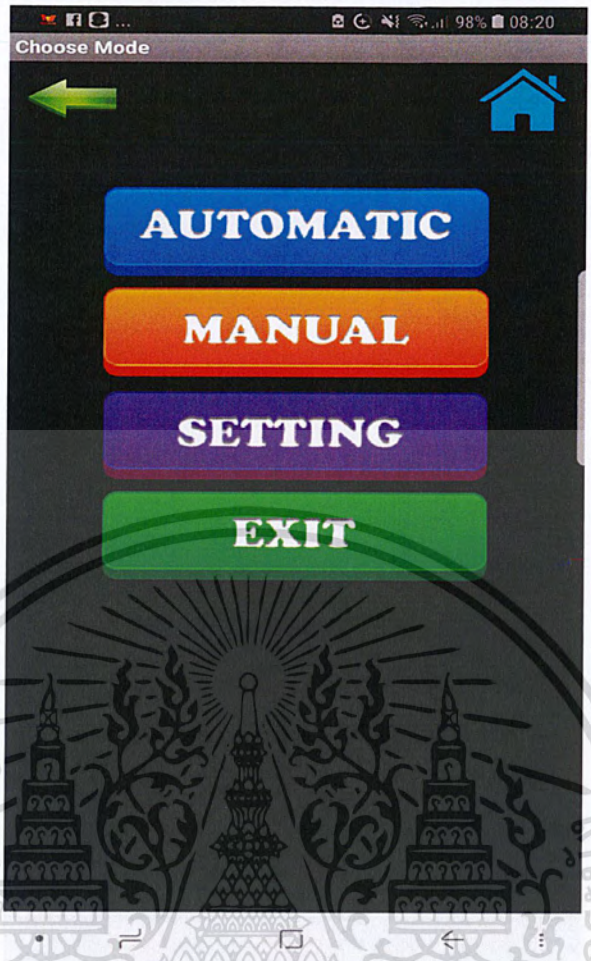
จากรูปที่ 3.11 เมื่อผู้ใช้งานเข้าสู่แอปพลิเคชันให้กดปุ่ม START จะเป็นการเริ่มการทำงานของโปรแกรม โดยสามารถกดปุ่มเครื่องหมายคำถาม ด้านขวาบน เมื่อต้องการคำอธิบายการทำงานของแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.12 หน้าจอช่วยเหลือแนะนำการใช้งาน

จากรูปที่ 3.12 จะอธิบายหลักการทำงานทั้งหมดของระบบให้ผู้ใช้เข้าใจได้ง่ายที่สุดพร้อมระบุ version ปัจจุบันที่ทำการอัพเดท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 หน้าจอเลือกระบบการทำงาน

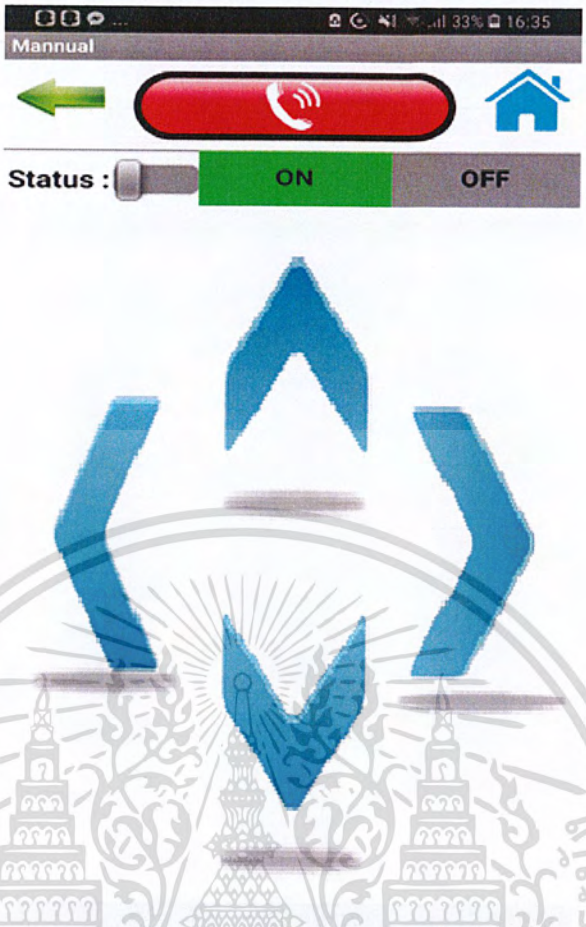
จากรูปที่ 3.13 ผู้ใช้สามารถเลือกฟังก์ชันการทำงานต่าง ๆ ได้จากหน้านี้ โดยเมนู AUTOMATIC คือเลือกใช้ฟังก์ชันการนำทางอัตโนมัติ, เมนู MANUAL คือเลือกใช้ฟังก์ชันควบคุมรถนั่งด้วยตนเอง, เมนู SETTING คือเมนูสำหรับทำการตั้งค่าแอปพลิเคชัน, เมนู EXIT คือเมนูสำหรับออกจากแอปพลิเคชัน, เมนู HOME คือ เมนูสำหรับกลับไปหน้า HOME และเมนู BACK คือเมนูสำหรับย้อนกลับไปหน้าก่อนหน้านี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.14 หน้าจอการทำงาน AUTOMATIC

จากรูปที่ 3.14 ผู้ใช้สามารถเลือกสถานที่เป้าหมายที่ต้องการจะไป โดยทำการเลือกปุ่มที่แสดงบนหน้าจอที่ระบุชื่อของสถานที่นั้น ๆ รถจะเคลื่อนที่ไปยังจุดหมายที่ต้องการจะไป หากเกิดอุบัติเหตุ ณ ขณะที่รถเคลื่อนที่รถจะทำการโทรออกฉุกเฉินอัตโนมัติ หรือผู้ใช้สามารถโทรออกฉุกเฉินด้วยตนเองได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 หน้าจอการทำงาน MANUAL

จากรูปที่ 3.15 แสดงหน้าจอการทำงานMANUAL จะมีปุ่มลูกศรซ้าย ขวา ขึ้น และ ลง เพื่อบังคับรถไปในทิศทางด้านซ้าย ด้านขวา เดินหน้า และถอยหลัง ตามลำดับ โดยการใช้งานปุ่มเหล่านี้จะต้องทำการกดปุ่ม ON ก่อน เพื่อให้รถนั่งอัตโนมัติเปลี่ยนสถานะเป็นพร้อมใช้งาน หรือเลือก OFF หากต้องการเปลี่ยนสถานะเป็นหยุดใช้งานรถนั่งอัตโนมัติ เพื่อป้องกันการสัมผัสปุ่มบังคับการเคลื่อนที่โดยไม่ได้ตั้งใจของผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.16 หน้าจอการทำงาน SETTING

จากรูปที่ 3.16 แสดงหน้าจอการตั้งค่าหมายเลขปลายทางเพื่อการโทรออกกรณีฉุกเฉิน กรณีที่รถนั่งล้ม หรือกรณีที่ผู้ใช้ต้องโทรออกเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดสอบและผลการทดสอบแอปพลิเคชัน

ในบทนี้จะกล่าวถึงการทดสอบและผลการทดสอบแอปพลิเคชันควบคุมรถนั่งอัตโนมัติ และอุปกรณ์ที่ใช้ทดสอบแอปพลิเคชัน

4.1 การทดสอบแอปพลิเคชัน SmartCar

4.1.1 อุปกรณ์ที่ใช้ทดสอบแอปพลิเคชัน

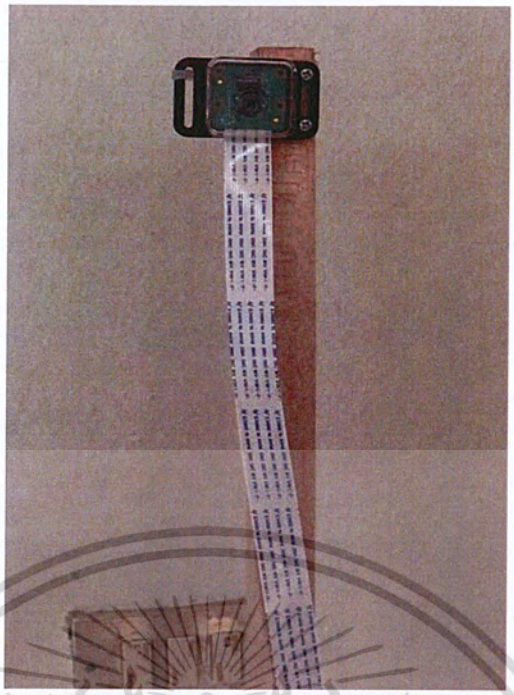
ในการทดสอบแอปพลิเคชัน SmartCar ใช้อุปกรณ์ในการทดสอบแอปพลิเคชันดังนี้

- รถวีลแชร์ (Wheelchair) รถนั่งที่นำมาใช้ในการทดสอบระบบ ต้องเป็นรถที่ดัดแปลงโดยมีการติดตั้งอุปกรณ์ต่าง ๆ เพื่อให้สามารถเคลื่อนที่ได้โดยอุปกรณ์เหล่านั้น ผู้พัฒนาได้นำรถวีลแชร์ที่มีการดัดแปลง มาพัฒนาต่อยอดตามวัตถุประสงค์การใช้งาน คือ การใช้งานสำหรับผู้สูงอายุ หรือผู้ที่จำเป็นต้องใช้รถนั่งอัตโนมัติภายในโรงพยาบาล ดังรูปที่ 4.1

รูปที่ 4.1 วีลแชร์ที่มีการดัดแปลง และนำมาพัฒนาต่อยอดตามวัตถุประสงค์การใช้งาน

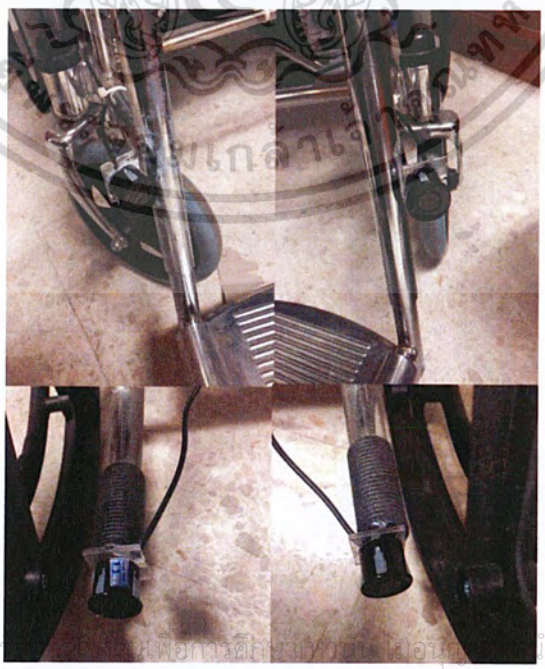
- กล้องตรวจจับภาพ (Camera Serial Interface) คือกล้องที่ใช้สำหรับตรวจจับภาพเพื่อนำไปประมวลผลและแสดงผลภาพผ่านทางหน้าจอสมาร์ทโฟน ในการติดกล้อง ต้องติดตัวกล้องให้มีองศาที่พอดีกับป้ายในการเคลื่อนที่ผ่านป้าย และปรับกล้องซูมเพื่อให้ภาพอยู่กึ่งกลางระหว่างช่วงหลังผู้ใช้งาน ดังรูปที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 ลักษณะ Camera Serial Interface

- ตัววัดระยะห่างระหว่างวัตถุ (JSN-SR04T Waterproof Ultrasonic Module) เป็นตัววัดระยะห่างระหว่างวัตถุกับรถวีลแชร์หรืออุปกรณ์ที่ติดตั้งอยู่ ผู้พัฒนาใช้ตัววัดระยะห่างระหว่างวัตถุทั้งหมด 4 บอร์ด โดยติดตั้งไว้ที่ตำแหน่งหน้ารถวีลแชร์ทางด้านซ้ายและขวา และท้ายรถวีลแชร์ทางด้านซ้ายและขวา เพื่อให้สามารถตรวจวัดระยะห่างได้ในทุกทิศทาง สามารถเพิ่มความปลอดภัยในการเคลื่อนที่ของรถวีลแชร์เมื่อพบสิ่งกีดขวาง ดังรูปที่ 4.3



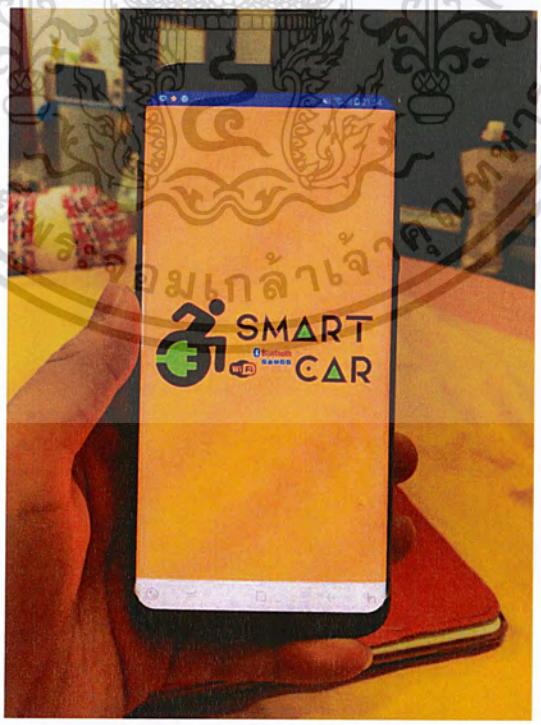
รูปที่ 4.3 ลักษณะ JSN-SR04T Waterproof Ultrasonic Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สมาร์ทโฟน (Smartphone) โทรศัพท์มือถือที่ใช้ระบบปฏิบัติการแอนดรอยด์ ผู้พัฒนาจะทำการติดตั้งแอปพลิเคชัน SmartCar บนสมาร์ทโฟนเพื่อใช้ในการควบคุมรถวีลแชร์ โดยสมาร์ทโฟนจะต้องมีคุณสมบัติดังนี้

  - หน้าจอแสดงผล
    - หน้าจอแสดงผลกว้าง 5.8 นิ้ว
    - ความละเอียด 1440x2960 พิกเซล
    - หน้าจอสัมผัสแบบ Multi-Touch
  - ระบบปฏิบัติการ หน่วยประมวลผล และหน่วยความจำ
    - CPU Exynos 8895 Octa Core ความเร็ว 2.3 GHz
    - RAM 4 GB
      ระบบปฏิบัติการแอนดรอยด์เวอร์ชั่น 5.5 ขึ้นไป
    - หน่วยความจำภายใน 16 GB หรือ 64 GB
  - ระบบการเชื่อมต่อ
    - รองรับการเชื่อมต่อ Wi-Fi, Bluetooth, USB, NFC
    - มีระบบ GPS และ A-GPS ในตัว

ตัวอย่างสมาร์ทโฟนที่มีการติดตั้งแอปพลิเคชัน SmartCar และมีคุณสมบัติตามที่กล่าวมาข้างต้น แสดงดังรูปที่ 4.4

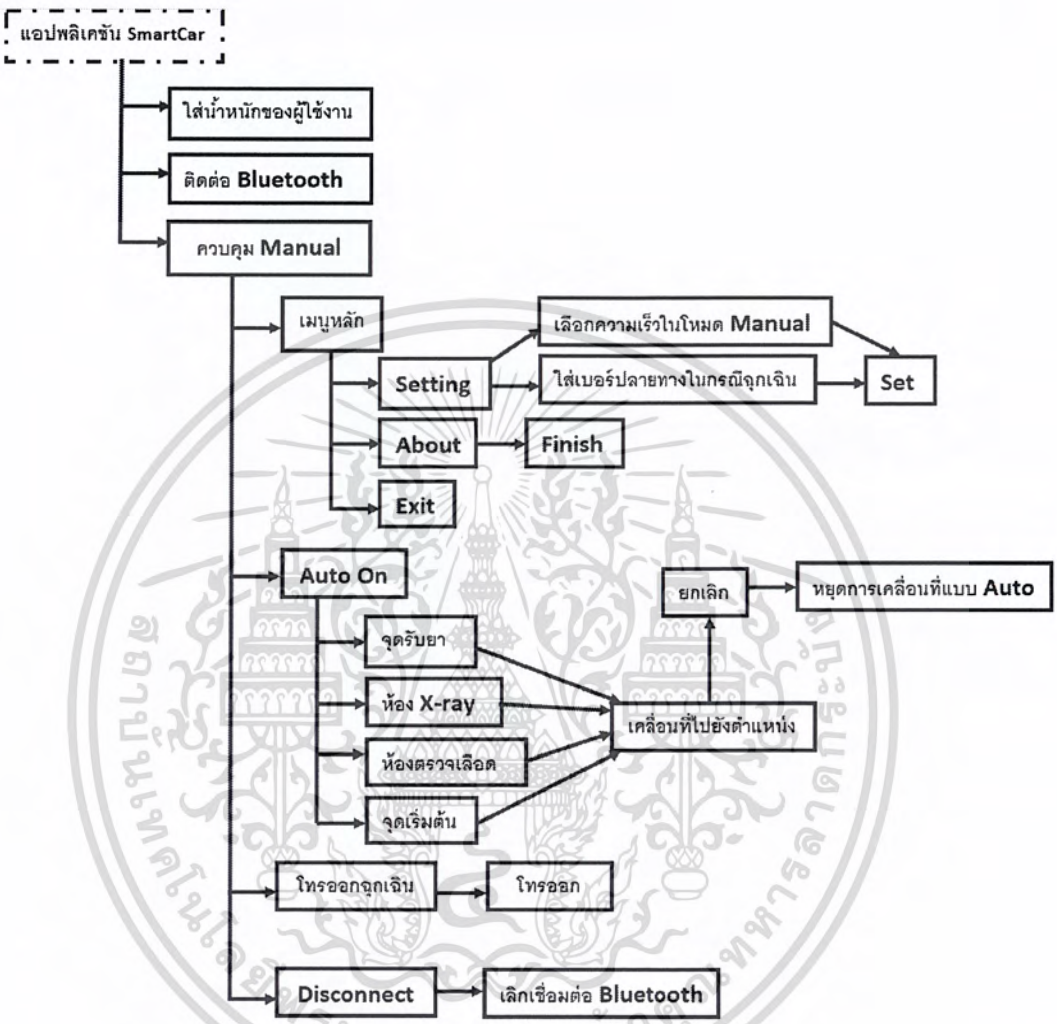


รูปที่ 4.4 สมาร์ทโฟนที่มีการติดตั้งแอปพลิเคชัน SmartCar

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 การทดสอบแอปพลิเคชัน

แอปพลิเคชันควบคุมรถนั่งอัตโนมัตินผ่านแอนดรอยด์แอปพลิเคชัน (แอปพลิเคชัน SmartCar) ประกอบไปด้วยโครงสร้างหน้าจอของโปรแกรม ดังรูปที่ 4.5



รูปที่ 4.5 โครงสร้างการใช้งานแอปพลิเคชัน SmartCar

จากรูปที่ 4.5 แสดงเห็นถึงโครงสร้างหน้าจอแอปพลิเคชัน SmartCar ซึ่งเป็นโครงสร้างที่แบ่งตามฟังก์ชันการใช้งานของแอปพลิเคชัน มีลำดับการแสดงผลของหน้าจอโดยเริ่มตั้งแต่หน้าจอการใส่น้ำหนักของผู้ใช้งาน หน้าจอการติดต่อบลูทูธ หน้าจอการควบคุม และหน้าจออื่นๆ ซึ่งจะแสดงผลตามที่ผู้ใช้ได้ทำการเลือกในแต่ละเมนู มีรายละเอียดของหน้าจอต่าง ๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) หน้าจอการตั้งค่าความเร็ว เมื่อเข้าสู่แอปพลิเคชัน ผู้ใช้งานจะต้องทำการระบุข้อมูลน้ำหนัก โดยระบบจะใช้ข้อมูลน้ำหนักในการกำหนดความเร็วในการเคลื่อนที่ของรถนั่ง และเป็นหน้าจอแรกสำหรับการเริ่มต้นการใช้งานการควบคุมรถนั่ง ดังรูปที่ 4.6

รูปที่ 4.6 หน้าจอแรกที่ให้ตั้งค่าความเร็วโดยอ้างอิงจากน้ำหนัก

2) หน้าจอการเชื่อมต่อบลูทูธ หลังจากกดปุ่ม START ในหน้าจอการตั้งค่าความเร็วแล้ว จะเข้าสู่หน้าการเชื่อมต่อบลูทูธ โดยภายในหน้านี้ ผู้ใช้จะพบกล่องข้อความแจ้งเตือนเพื่อให้ผู้ใช้ทำการเชื่อมต่อบลูทูธระหว่างสมาร์ทโฟนและรถนั่งอัตโนมัติ ดังรูปที่ 4.7

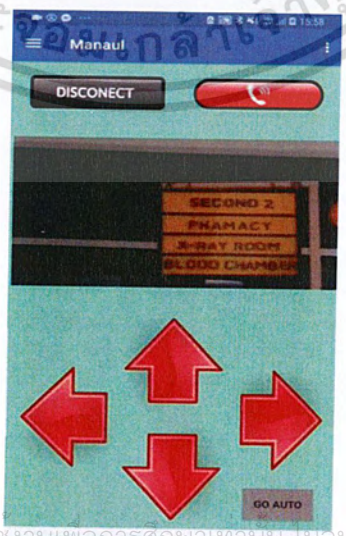
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.7 หน้าจอเปิดบลูทูธ

ในการค้นหาสัญญาณบลูทูธ ผู้ใช้จะต้องทำการกดปุ่ม Paired Devices เพื่อทำการค้นหาสัญญาณ ซึ่งหน้าจอจะแสดงผล ดังรูปที่ 4.8

รูปที่ 4.8 หน้าจอหลังกดปุ่ม Paired Devices

3) หน้าจอควบคุมรถนั่งอัตโนมัติแบบ Manual เมื่อผู้ใช้ทำการเชื่อมต่อบลูทูธเรียบร้อยแล้ว จะแสดงหน้าจอนี้ขึ้นมาโดยภายในหน้าจอนี้ จะประกอบไปด้วยฟังก์ชันการแสดงผลภาพจากกล้องที่ติดตั้งไว้ที่รถวีลแชร์ ปุ่มควบคุมทิศทาง ปุ่มเปลี่ยนการควบคุมเป็นแบบอัตโนมัติ และปุ่มโทรออกฉุกเฉิน ดังรูปที่ 4.9



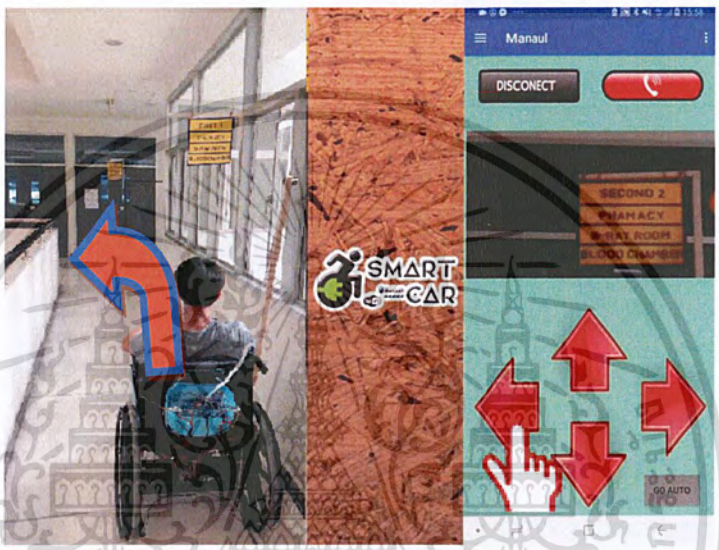
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.9 หน้าจอโหมด Manual

รายละเอียดของแต่ละฟังก์ชันภายในหน้าจอควบคุมรถนั่งอัตโนมัติแบบ Manual ได้แก่ การใช้โหมดควบคุมโดยผู้ใช้ (Manual) และโหมดควบคุมแบบอัติโนมัติ (Automatic) โดยมีรายละเอียดดังนี้

- การใช้โหมดควบคุมโดยผู้ใช้งาน(Manual) เป็นโหมดที่ผู้ใช้สามารถควบคุมการเคลื่อนที่ของรถนั่งได้ด้วยตนเอง มีรายละเอียดขั้นตอนการใช้งาน ดังนี้

  - เมื่อกดปุ่มควบคุมลูกศรทางซ้าย รถนั่งจะเคลื่อนที่ไปในทิศทางด้านซ้าย ดังรูปที่ 4.10



รูปที่ 4.10 รถนั่งเลี้ยวซ้าย

  - เมื่อเลือกปุ่มควบคุมลูกศรทางขวา รถนั่งจะเคลื่อนที่ไปในทิศทางด้านขวา ดังรูปที่ 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.11 รถนั่งเลี้ยวขวา

- เมื่อเลือกปุ่มควบคุมลูกศรทางด้านหน้า รถนั่งจะเคลื่อนที่ไปในทิศทางด้านหน้า ดังรูปที่ 4.12

รูปที่ 4.12 รถนั่งเดินหน้า

- เมื่อเลือกปุ่มควบคุมลูกศรทางด้านหลัง รถนั่งจะเคลื่อนที่ไปในทิศทางถอยหลัง พร้อมมีภาพเคลื่อนไหวจากทางด้านหลังแสดงบนแอปพลิเคชัน ดังรูปที่ 4.13

รูปที่ 4.13 รถนั่งถอยหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ในการใช้โหมดควบคุมแบบอัติโนมัติ (Automatic)

- เมื่อผู้ใช้ต้องการใช้โหมดควบคุมแบบอัติโนมัติ ต้องทำการเคลื่อนที่รถนั่งอัตโนมัติด้วยโหมดควบคุมโดยผู้ใช้งาน เพื่อไปยังป้ายบอกทางตามสถานที่ต่าง ๆ โดยป้ายจะต้องมีการระบุชื่อสถานที่ที่ผู้ใช้ต้องการจะไป หากผู้ใช้สังเกตภาพบนสมาร์ทโฟนที่ได้จากการจับภาพของกล้องจะเห็นการประมวลผลภาพเพื่ออ่านข้อความที่อยู่บนที่ประมวลผลป้ายเป็นลักษณะของกรอบสี่เหลี่ยมที่อยู่รอบข้อความในขณะที่ใช้โหมดควบคุมโดยผู้ใช้งาน ดังรูปที่ 4.14

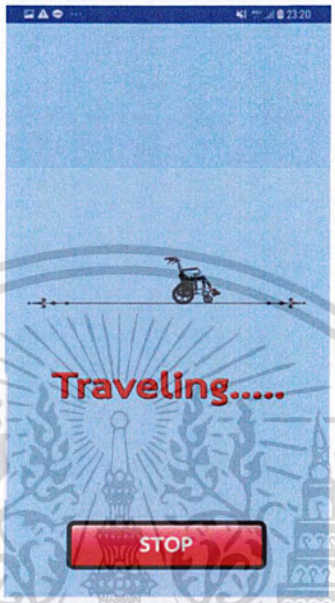
รูปที่ 4.14 การเปลี่ยนการใช้งานเป็นโหมดอัตโนมัติ

- หากมีกรอบสีเขียวขึ้นที่ข้อความในภาพที่ได้จากการประมวลผลภาพ หมายถึงผู้ใช้สามารถไปยังสถานที่นั้นได้ โดยผู้ใช้จะต้องทำการกดปุ่ม Go Auto เพื่อไปยังสถานที่เป้าหมาย ดังรูปที่ 4.15

รูปที่ 4.15 การเข้าสู่หน้าโหมดควบคุมรถนั่งแบบอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- หน้าจอแสดงสถานนะการเคลื่อนที่ของรถนั่ง ในหน้าจอนี้จะแสดงสถานะว่ารถกำลังเคลื่อนที่โดยอัตโนมัติ ขณะที่รถนั่งอัตโนมัติกำลังเคลื่อนที่ หากผู้ใช้ต้องการจะทำการหยุดการเคลื่อนไหวของรถนั่ง ผู้ใช้สามารถกดปุ่ม Stop เพื่อทำการหยุดการเคลื่อนที่ได้ ดังรูปที่ 4.16



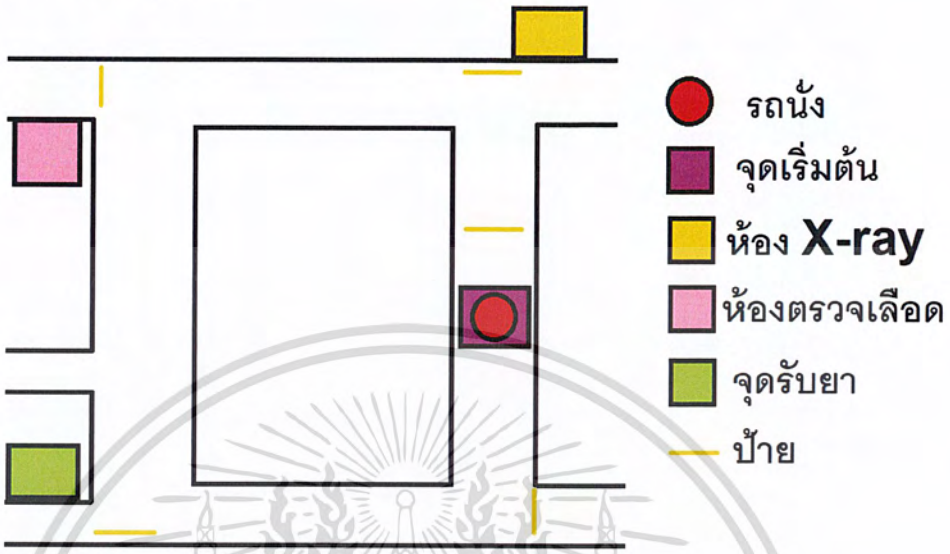
รูปที่ 4.16 หน้าจอแสดงสถานะการเคลื่อนที่ของรถนั่ง

- หน้าจอแสดงการแจ้งเตือนเมื่อรถเคลื่อนที่ถึงสถานที่เป้าหมายโดยอัตโนมัติ เมื่อรถเคลื่อนที่ไปถึงสถานที่เป้าหมายเรียบร้อยแล้ว จะมีการแสดงกล่องข้อความแจ้งเตือนผ่านทางหน้าจอ เพื่อแจ้งให้ผู้ใช้งานทราบ ดังรูปที่ 4.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น รวมถึงไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้งที่มีการนำไปใช้

รูปที่ 4.17 หน้าจอแสดงการแจ้งเตือนเมื่อรถเคลื่อนที่ถึงสถานที่เป้าหมายโดยอัตโนมัติ

- เมื่อผู้ใช้ต้องการไปยังถสานที่ที่ไม่มีระบุภายในป้าย ผู้ใช้สามารถเปลี่ยนจากโหมดควบคุมแบบอัติโนมัติ เป็นโหมดควบคุมโดยผู้ใช้งานได้ เพื่อควบคุมรถนั่งไปยังสถานที่นั้น ๆ โดยมีแผนผังดังรูปที่ 4.18



รูปที่ 4.18 แผนผังในการเคลื่อนที่โหมด Auto

- แถบเครื่องมือ แบ่งออกเป็น 2 เมนู ได้แก่ การตั้งค่า (Setting) และการช่วยเหลือ (Help) โดยแถบเครื่องนี้จะแสดงผลทางหน้าจอดังรูปที่ 4.19

รูปที่ 4.19 แถบเครื่องมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของเมนูการตั้งค่า และการช่วยเหลือ มีดังต่อไปนี้

- การตั้งค่า (Setting) ผู้ใช้งานสามารถตั้งค่าเบอร์โทรศัพท์ที่จะทำการโทรออกฉุกเฉิน หรือตั้งค่าความเร็วในการเคลื่อนที่รถนั่งอัตโนมัติในโหมดควบคุมโดยผู้ใช้งาน โดยแสดงผลหน้าจอ ดังรูปที่ 4.20

รูปที่ 4.20 หน้าจอการตั้งค่า

- การช่วยเหลือ (Help) เมื่อเข้าสู่เมนูนี้ จะแสดงการสอนวิธีใช้แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ ดังรูปที่ 4.21

รูปที่ 4.21 หน้าจอปุ่ม About

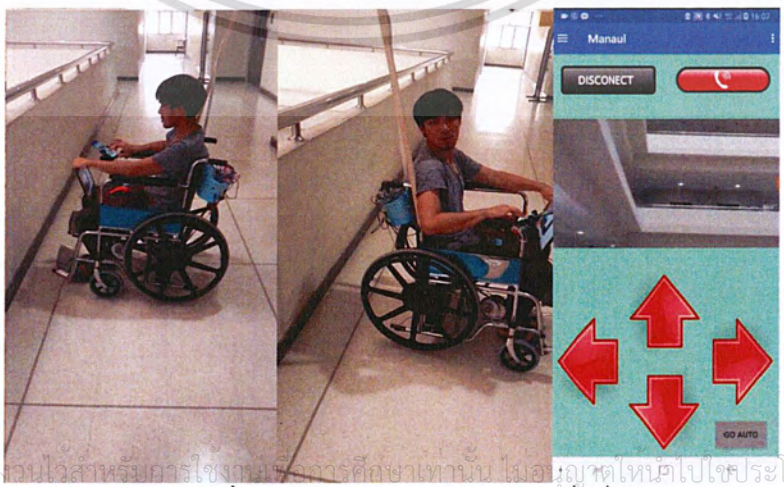
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) การตรวจล้ม และกันชนด้านหน้าและด้านหลังได้ทั้ง 2 โหมด

- ในการตรวจล้มใช้ตัววัดความเร่ง แกน X,Y ในตัวของโทรศัพท์ในการตรวจล้ม ระบบตรวจล้มสามารถตรวจล้มได้ทั้ง 2 โหมด จะทำการส่งเสียสัญญานเพื่อแจ้งเตือนให้ผู้ใช้ทราบว่ารถกำลังจะล้ม ดังรูปที่ 4.22

รูปที่ 4.22 ทดสอบการตรวจล้ม

- ตรวจระยะห่างระหว่างวัตถุเพื่อป้องกันการชน ด้านหน้าละด้านหลัง ทั้ง 2 โหมด แสดงดังรูปที่ 4.23



รูปที่ 4.23 ทดสอบหยุดรถในขณะที่มีสิ่งกีดขวาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ผลการทดสอบแอปพลิเคชัน

ในการพัฒนารถนั่งอัตโนมัติบนระบบแอนดรอยด์แอปพลิเคชัน โดยในการทดสอบแบ่งออกเป็น 2 ส่วน ดังนี้

4.2.1 การทดสอบการทำงานในการตรวจล้ม

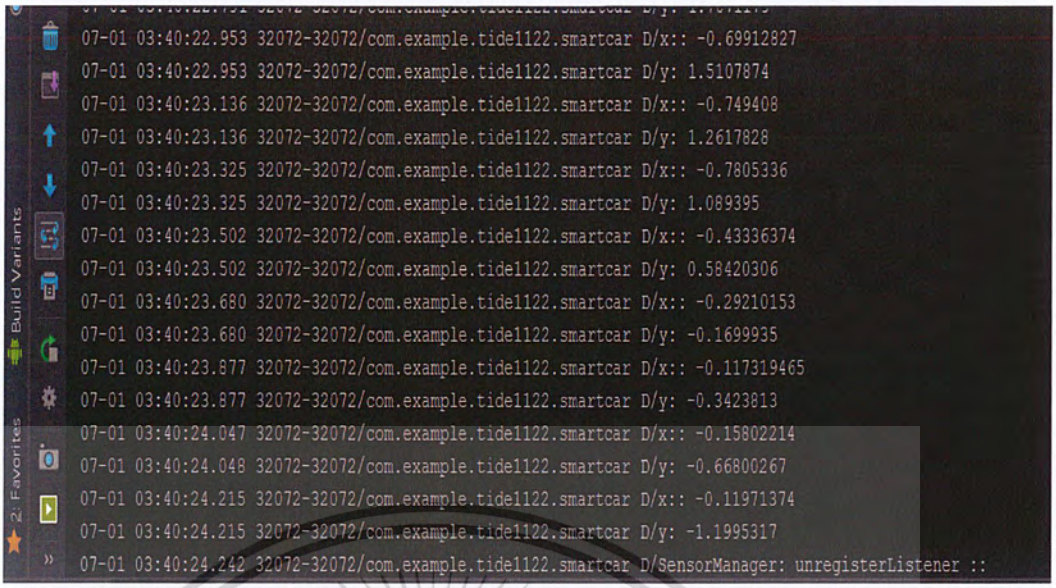
ลักษณะการตรวจล้มของรถนั่งได้นำตัววัดความเร่งในโทรศัพท์มือถือเข้ามาใช้ เพื่อตรวจจับลักษณะการทรงตัวของรถนั่ง เช่น การล้มของรถนั่งนั้นต้องใช้ค่าความเร่งเข้ามาช่วยในการคำนวณเพิ่มขึ้นด้วย โดยเปรียบเทียบกับค่ามาตรฐานปกติ หากความเร่งในขณะนั้นมีมากเกินค่าเฉลี่ยที่กำหนดไว้แสดงว่ารถนั่งอาจล้ม ผลของการรับค่าเฉลี่ยการทรงตัว แสดงดังรูปที่ 4.24

รูปที่ 4.24 ผลการรับค่าเฉลี่ยการทรงตัวแบบปกติ

จากรูปที่ 4.24 ได้แสดงข้อมูลค่าที่วัดได้จากการเคลื่อนไหวของโทรศัพท์มือถือ ที่ติดตั้งอยู่กับรถนั่ง โดยจะแบ่งออกเป็นแกน X แกน Y ตามลำดับ เมื่อใดก็ตามที่ผู้ใช้อุปกรณ์เฝ้าระวังมีการเคลื่อนไหว ค่าที่ถูกใส่ลงในตารางจะเพิ่มขึ้นและลดลงตามทิศทางของการเคลื่อนไหวนั้น ๆ ซึ่งในส่วนนี้ ทางผู้จัดทำได้มุ่งเน้นไปยังการวัดค่าความเร่ง ในแต่ละแกน โดยตัวเลขในคอลัมน์แรกแสดงค่าแกน X ส่วนตัวเลขในคอลัมน์ 2 จะแสดงค่าแกน Y

ผู้พัฒนาได้ทำการทดลอง โดยจำลองการล้มของรถนั่ง ในลักษณะต่าง ๆตามแกนที่กำหนดไว้ ซึ่งผลที่ออกมาจะมีค่าของแกนใดแกนหนึ่งเปลี่ยนไป ในขณะที่ผู้ทดลองทำการจำลองการล้ม แสดงผลลัพธ์จากการจำลองการล้มในทิศทางต่าง ๆ ดังรูปที่ 4.25 ถึงรูปที่ 4.28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.25 ค่าที่ได้จากการจำลองการล้มกรณีรถนั่งล้มไปยังทิศทางด้านหน้า

รูปที่ 4.26 ค่าที่ได้จากการจำลองการล้มกรณีรถนั่งล้มไปยังทิศทางด้านหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
07-01 03:36:47.659 31174-31174/com.example.tide1122.smartcar D/y: 5.4852834
07-01 03:36:47.852 31174-31174/com.example.tide1122.smartcar D/x:: 1.7430321
07-01 03:36:47.852 31174-31174/com.example.tide1122.smartcar D/y: 5.2650104
07-01 03:36:48.039 31174-31174/com.example.tide1122.smartcar D/x:: 3.5890179
07-01 03:36:48.039 31174-31174/com.example.tide1122.smartcar D/y: 4.9537544
07-01 03:36:48.212 31174-31174/com.example.tide1122.smartcar D/x:: 3.677606
07-01 03:36:48.212 31174-31174/com.example.tide1122.smartcar D/y: 5.1620564
07-01 03:36:48.383 31174-31174/com.example.tide1122.smartcar D/x:: 4.3791285
07-01 03:36:48.383 31174-31174/com.example.tide1122.smartcar D/y: 5.1620564
07-01 03:36:48.568 31174-31174/com.example.tide1122.smartcar D/x:: 4.886715
07-01 03:36:48.568 31174-31174/com.example.tide1122.smartcar D/y: 5.03516
07-01 03:36:48.746 31174-31174/com.example.tide1122.smartcar D/x:: 5.1452966
07-01 03:36:48.746 31174-31174/com.example.tide1122.smartcar D/y: 4.929812
07-01 03:36:48.920 31174-31174/com.example.tide1122.smartcar D/x:: 5.6002088
07-01 03:36:48.921 31174-31174/com.example.tide1122.smartcar D/y: 5.05192
07-01 03:36:49.116 31174-31174/com.example.tide1122.smartcar D/x:: 6.3903193
07-01 03:36:49.116 31174-31174/com.example.tide1122.smartcar D/y: 4.790944
```

รูปที่ 4.27 ค่าที่ได้จากการจำลองการล้มกรณีรถนั่งล้มไปยังทิศทางด้านซ้าย

รูปที่ 4.28 ค่าที่ได้จากการจำลองการล้มกรณีรถนั่งล้มไปยังทิศทางด้านขวา

จากรูปที่ 4.24 ถึงรูปที่ 4.28 เป็นผลลัพธ์ที่ได้เมื่อผู้ทดลองทำการจำลองการล้มของรถนั่ง ซึ่งจะสังเกตเห็นได้ว่า ค่าที่ได้นั้นจะต่างกันมากจนสังเกตได้ ซึ่งทำให้สามารถจำแนกการล้มได้ตามแกนความเร่งต่าง ๆ โดยสามารถจำแนกทิศทางการล้มของรถนั่งกับค่าแกนที่เปลี่ยนไปได้ ดังตารางที่ 4.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 ทิศทางการล้มของรถนั่ง และค่าของแกนความเร่งที่เปลี่ยนไป

ทิศทางการล้มของรถนั่ง	ค่าของแกนความเร่งที่เปลี่ยนไป	
	แกน X	แกน Y
ด้านหน้า	-	น้อยกว่า - 1
ด้านหลัง	-	มากกว่า 10
ด้านซ้าย	มากกว่า 6	-
ด้านขวา	น้อยกว่า - 6	-

จากตารางที่ 4.1 จะแสดงค่าของแกนความเร่งที่เปลี่ยนไปเมื่อผู้ทดลองจำลองการล้มของรถนั่งในทิศทางต่าง ๆ เช่น เมื่อค่าแกน Y ลดลงจนเหลือค่ามากกว่า 10 จะส่งเสียงสัญญานเพื่อเตือนว่ารถจะล้มให้หยุดการเคลื่อนที่

4.2.2 การทดสอบความเร็วโดยเปลี่ยนแปลงน้ำหนัก

ลักษณะความเร็วจะมีผลต่อน้ำหนัก และจะทราบค่า duty cycle ในการเลือกความเร็วให้เหมาะสมกับน้ำหนักได้ ในการทำสอบความเร็ว ผู้พัฒนาได้ทำการทดสอบกับผู้ใช้ที่มีน้ำหนักแตกต่างกัน โดยผลการทดสอบแสดงดังตารางที่ 4.2

ตารางที่ 4.2 น้ำหนักส่งผลต่อความเร็ว

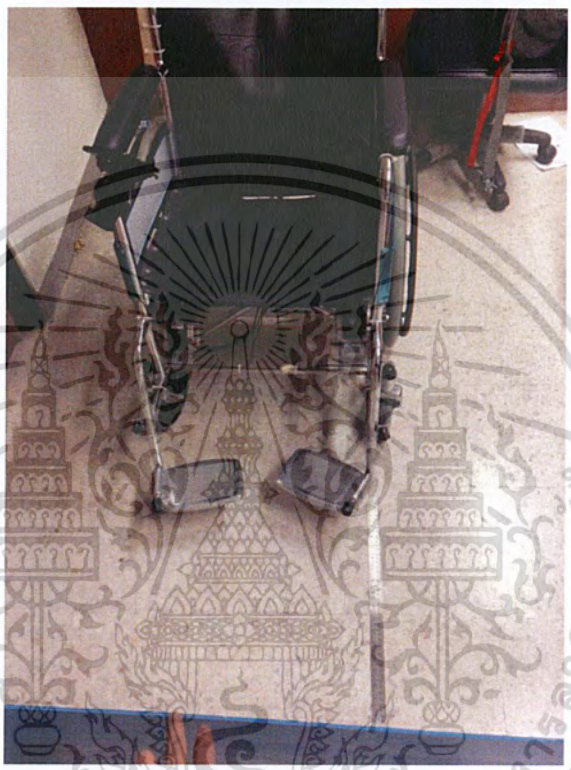
ค่า duty cycle	น้ำหนัก				
	20 kg	40 kg	60 kg	80 kg	100 kg
15	พอดี	พอดี	พอดี	ไม่พอดี	ไม่พอดี
20	ไม่พอดี	ไม่พอดี	พอดี	พอดี	ไม่พอดี
25	ไม่พอดี	ไม่พอดี	พอดี	พอดี	พอดี
30	ไม่พอดี	ไม่พอดี	ไม่พอดี	พอดี	พอดี

จากตารางที่ 4.2 จะแสดงให้เห็นน้ำหนักที่ 20-60 kg ค่า Duty cycle ที่ 15 จึงจะพอดีต่อการเคลื่องที่ ดังนั้นผู้พัฒนาจึงทำการตั้งค่าความเร็ว โดยระดับความเร็ว Standard ให้เป็นช่วงระหว่าง 20 ถึง 60 Kg เท่ากับ 15 Duty cycle และทำการตั้งค่าอีก 2 ระดับ โดยคำนวณให้ Medium มีค่าความเร็วเป็น 2 เท่า และ High มีค่าความเร็วเป็น 3 เท่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3 การทดสอบการทำงานในการตรวจสิ่งกีดขวาง

ลักษณะการตรวจสิ่งกีดขวางของรถนั่งได้นำตัววัดระยะห่างระหว่างวัตถุ 4 บอร์ด เข้ามาใช้ เพื่อตรวจจับสิ่งกีดขวางด้านหน้าและด้านหลังของรถนั่ง เช่น ในขณะเคลื่อนที่ในโหมด Manual จะทำการหยุดและไม่สามารถเลือกลูกศรในแอปพลิเคชันในทิศทางที่มีสิ่งกีดขวางได้ และ ในโหมด Auto หยุดเมื่อสิ่งกีดขวางผ่านไปจะทำงานต่อจนถึงตำแหน่งที่เลือกไว้ แสดงดังรูปที่ 4.29



รูปที่ 4.29 ผลการทดลองการหยุดเมื่อเจอสิ่งกีดขวาง

ตารางที่ 4.3 ระยะตรวจสิ่งกีดขวาง โดยระดับความเร็ว Standard

รอบในการ Test	ระยะหยุดเมื่อเจอสิ่งกีดขวาง			
	25 cm.	50 cm.	75 cm.	100 cm.
1	ไม่พอดี	พอดี	พอดี	ไม่พอดี
2	ไม่พอดี	พอดี	ไม่พอดี	ไม่พอดี
3	ไม่พอดี	พอดี	พอดี	ไม่พอดี
4	ไม่พอดี	พอดี	ไม่พอดี	ไม่พอดี
5	ไม่พอดี	พอดี	ไม่พอดี	ไม่พอดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางที่ 4.3 แสดงข้อมูลการหยุดรถนั่งอัตโนมัติ ที่หยุดได้ระยะพอดีและไม่เกิดอุบัติเหตุ ในระดับความเร็วมาตรฐาน ตั้งค่าการหยุดได้ที่ 50 เซนติเมตร

ตารางที่ 4.4 ระยะตรวจสิ่งกีดขวาง โดยระดับความเร็ว Medium

รอบในการ Test	ระยะหยุดเมื่อเจอสิ่งกีดขวาง			
	25 cm.	50 cm.	75 cm.	100 cm.
1	ไม่พอดี	พอดี	ไม่พอดี	ไม่พอดี
2	ไม่พอดี	ไม่พอดี	ไม่พอดี	ไม่พอดี
3	ไม่พอดี	พอดี	พอดี	ไม่พอดี
4	ไม่พอดี	ไม่พอดี	พอดี	ไม่พอดี
5	ไม่พอดี	พอดี	ไม่พอดี	ไม่พอดี

จากตารางที่ 4.4 แสดงข้อมูลการหยุดรถนั่งอัตโนมัติ ที่หยุดได้ระยะพอดีและไม่เกิดอุบัติเหตุ ในระดับความเร็วมาตรฐาน ตั้งค่าการหยุดได้ที่ 60 เซนติเมตร

ตารางที่ 4.5 ระยะตรวจสิ่งกีดขวาง โดยระดับความเร็ว High

รอบในการ Test	ระยะหยุดเมื่อเจอสิ่งกีดขวาง			
	25 cm.	50 cm.	75 cm.	100 cm.
1	ไม่พอดี	ไม่พอดี	พอดี	ไม่พอดี
2	ไม่พอดี	ไม่พอดี	พอดี	ไม่พอดี
3	ไม่พอดี	ไม่พอดี	พอดี	ไม่พอดี
4	ไม่พอดี	ไม่พอดี	พอดี	ไม่พอดี
5	ไม่พอดี	ไม่พอดี	ไม่พอดี	ไม่พอดี

จากตารางที่ 4.5 แสดงข้อมูลการหยุดรถนั่งอัตโนมัติ ที่หยุดได้ระยะพอดีและไม่เกิดอุบัติเหตุ ในระดับความเร็วมาตรฐาน ตั้งค่าการหยุดได้ที่ 70 เซนติเมตร

4.2.4 การทดสอบการติดตั้งป้าย

การทดสอบการติดตั้งป้าย ทดสอบโดยการเคลื่อนที่รถนั่งอัตโนมัติไปยังตำแหน่งที่ติดตั้งป้ายไว้ โดยสังเกตการจับภาพของกล้องเพื่อทดสอบขนาดขององศา และความสูงของป้ายที่สามารถประมวลผลภาพได้ แสดงการทดสอบ ดังรูปที่ 4.30 ถึงรูปที่ 4.31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.30 ผลการทดลองติดตั้งองศาป้าย

ตารางที่ 4.6 องศาในการติดตั้งป้ายในการประมวลผลภาพ

รอบในการ Test	องศาในการติดตั้งป้าย			
	40 องศา	60 องศา	80 องศา	100 องศา
1	ไม่ได้	ได้	ไม่ได้	ไม่ได้
2	ไม่ได้	ได้	ได้	ไม่ได้
3	ไม่ได้	ได้	ไม่ได้	ไม่ได้
4	ไม่ได้	ได้	ไม่ได้	ไม่ได้
5	ไม่ได้	ได้	ได้	ไม่ได้

จากตารางที่ 4.6 แสดงข้อมูลองศาในการติดตั้งป้าย เพื่อการประมวลผลภาพของกล้อง ช่วงระหว่าง 60-75 องศา กล้องจะสามารถประมวลผลภาพได้

รูปที่ 4.31 ผลการทดลองระยะการประมวลผลของกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.7 ระยะที่กล้องสามารถประมวลผลภาพ

รอบในการ Test	ระยะหยุดเมื่อเจอสิ่งกีดขวาง			
	170 cm.	190 cm.	210 cm.	230 cm.
1	ไม่พอดี	ไม่พอดี	พอดี	ไม่พอดี
2	ไม่พอดี	พอดี	ไม่พอดี	ไม่พอดี
3	ไม่พอดี	พอดี	พอดี	ไม่พอดี
4	ไม่พอดี	พอดี	ไม่พอดี	ไม่พอดี
5	ไม่พอดี	ไม่พอดี	ไม่พอดี	ไม่พอดี

จากตารางที่ 4.7 จะแสดงให้เห็นได้ว่าความสูงในการติดตั้งป้ายในการประมวลผลภาพ ต้องอยู่ในช่วง 195 – 205 cm. จึงจะสามารถประมวลผลภาพได้

4.2.5 การทดสอบระยะในการประมวลผลภาพ

ในการทดสอบระยะของการตรวจจับภาพและประมวลผลภาพของกล้องที่ติดตั้งบนรถนั่งอัตโนมัติ ผู้พัฒนาได้ทำการทดสอบทั้งหมด 5 ครั้ง โดยทำการเคลื่อนที่รถนั่งอัตโนมัติและเมื่อรถสามารถจับภาพจากป้ายและประมวลผลได้แล้วจึงทำการหยุดรถและวัดระยะระหว่างกล้องกับป้าย ตัวอย่างการทดสอบระยะในการประมวลผลภาพ แสดงดังรูปที่ 4.32

รูปที่ 4.32 ระยะห่างระหว่างป้ายและกล้องที่ประมวลผลภาพได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.8 ระยะห่างระหว่างป้ายกับตัวกล้องในการประมวลผล

รอบในการ Test	ลำดับของป้าย				
	1	2	3	4	5
1	120 cm.	90 cm.	105 cm.	104 cm.	113 cm.
2	110 cm.	120 cm.	101 cm.	120 cm.	116 cm.
3	111 cm.	115 cm.	111 cm.	108 cm.	96 cm.
4	100 cm.	121 cm.	109 cm.	105 cm.	120 cm.
5	110 cm.	109 cm.	120 cm.	112 cm.	100 cm.

จากตารางที่ 4.8 จะแสดงให้เห็นว่าระยะการประมวลผลภาพระหว่างป้ายกับกล้องต้องความความห่างจากกันไม่เกิน 120 cm

4.2.6 การทดสอบกับผู้ใช้จริง

ในการทดสอบกับผู้ใช้จริง ผู้พัฒนาเลือกผู้ใช้ที่มีอายุในช่วงระหว่าง 40-60 ปี เพื่อทำการทดสอบกับระบบ โดยให้ทดสอบฟังก์ชันการใช้งานทุกฟังก์ชันเพื่อเก็บข้อมูลและทำการสรุปผลการใช้งานของผู้ใช้จริง และนำไปแก้ไขเพื่อให้ระบบมีประสิทธิภาพและมีความปลอดภัย เหมาะสมต่อผู้ใช้มากที่สุด แสดงดังรูปที่ 4.33 ถึงรูปที่ 4.34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.33 ทดสอบการใช้งานรถนั่งอัตโนมัติกับผู้ใช้งานรายที่หนึ่ง

รูปที่ 4.34 ทดสอบการใช้งานรถนั่งอัตโนมัติกับผู้ใช้งานรายที่สอง

ในการทดสอบรถนั่งอัตโนมัติกับผู้ใช้งาน ได้มีการทดสอบโดยให้ผู้ใช้ทำลองฟังก์ชันต่าง ๆ ของรถนั่ง ได้แก่ โหมด Manual โหมด Automatic ระบบตรวจล้ม และระบบหยุดอัตโนมัติเมื่อเจอสิ่งกีดขวาง จากนั้นได้ทำการประเมินความพึงพอใจของผู้ใช้งานได้ผลสรุปว่า โหมด Manual อยู่ในระดับดี โหมด Automatic อยู่ในระดับพอใช้ ระบบตรวจล้ม อยู่ในระดับดี ระบบหยุดอัตโนมัติเมื่อเจอสิ่งกีดขวาง อยู่ในระดับดีมาก โดยผู้ใช้มีข้อเสนอแนะว่า ควรปรับปรุงรถนั่งให้ควบคุมทิศทางและประมวลผลภาพได้ real time มากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ตัวอย่างโค้ดโปรแกรม

4.3.1 โค้ดในฝั่งของ Raspberry pi

```
# method นี้เป็นส่วนของการ ใช้กล้องจับภาพเพื่อส่งภาพขึ้นไปยังเว็บเบราเซอร์เพื่อให้แอปพลิเคชัน
สามารถเรียกดูแบบ real time
class VideoCamera(object):
    def __init__(self):
        self.i = 0
    def __del__(self):
        print("555555")
        rawCapture.truncate(0)
        Modul3()
    def get_frame(self):
        for frame in camera.capture_continuous(rawCapture, format="bgr",
use_video_port=True):
            image = frame.array
            QueryImgBGR=image
            QueryImg=cv2.cvtColor(QueryImgBGR,cv2.COLOR_RGB2GRAY)
            queryKP,queryDesc=detector.detectAndCompute(QueryImg,None)
            matches=flann.knnMatch(queryDesc,trainDesc1,k=2)
            goodMatch=[]
            for m,n in matches:
                if(m.distance<0.75*n.distance):
                    goodMatch.append(m)

            if(len(goodMatch)>MIN_MATCH_COUNT):
                tp=[]
                qp=[]
                for m in goodMatch:
                    tp.append(trainKP1[m.trainIdx].pt)
                    qp.append(queryKP[m.queryIdx].pt)
                tp,qp=np.float32((tp,qp))
                H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        h,w=trainImg1.shape
        trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
        queryBorder=cv2.perspectiveTransform(trainBorder,H)
        cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(0,0,255),5)
        db.child("").update({"zone": "zone1"})
        db.child("").update({"state": "1"})
        self.i=0

    else:
        self.i = self.i+1
        if(self.i==15):
            db.child("").update({"state": "0"})
            self.i=0

    matches=flann.knnMatch(queryDesc,trainDesc2,k=2)
    goodMatch=[]
    for m,n in matches:
        if(m.distance<0.75*n.distance):
            goodMatch.append(m)
    if(len(goodMatch)>MIN_MATCH_COUNT):
        tp=[]
        qp=[]
        for m in goodMatch:
            tp.append(trainKP2[m.trainIdx].pt)
            qp.append(queryKP[m.queryIdx].pt)
        tp,qp=np.float32((tp,qp))
        H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
        h,w=trainImg2.shape
        trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
        queryBorder=cv2.perspectiveTransform(trainBorder,H)
        cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(0,255,0),5)
    matches=flann.knnMatch(queryDesc,trainDesc3,k=2)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
goodMatch=[]
for m,n in matches:
    if(m.distance<0.75*n.distance):
            goodMatch.append(m)
if(len(goodMatch)>MIN_MATCH_COUNT):
    tp=[]
    qp=[]
    for m in goodMatch:
        tp.append(trainKP3[m.trainIdx].pt)
        qp.append(queryKP[m.queryIdx].pt)
    tp,qp=np.float32((tp,qp))
    H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
    h,w=trainImg3.shape
    trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
    queryBorder=cv2.perspectiveTransform(trainBorder,H)
    cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(255,0,0),5)

matches=flann.knnMatch(queryDesc,trainDesc4,k=2)
goodMatch=[]
for m,n in matches:
    if(m.distance<0.75*n.distance):
            goodMatch.append(m)
if(len(goodMatch)>MIN_MATCH_COUNT):
    tp=[]
    qp=[]
    for m in goodMatch:
        tp.append(trainKP4[m.trainIdx].pt)
        qp.append(queryKP[m.queryIdx].pt)
    tp,qp=np.float32((tp,qp))
    H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
    h,w=trainImg4.shape
    trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
    queryBorder=cv2.perspectiveTransform(trainBorder,H)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(255,255,0),5)

    matches=flann.knnMatch(queryDesc,trainDesc5,k=2)
    goodMatch=[]
    for m,n in matches:
        if(m.distance<0.75*n.distance):
            goodMatch.append(m)
    if(len(goodMatch)>MIN_MATCH_COUNT):
        tp=[]
        qp=[]
        for m in goodMatch:
            tp.append(trainKP5[m.trainIdx].pt)
            qp.append(queryKP[m.queryIdx].pt)
        tp,qp=np.float32((tp,qp))
        H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
        h,w=trainImg5.shape
        trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
        queryBorder=cv2.perspectiveTransform(trainBorder,H)
        cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(0,255,255),5)
        db.child("").update({"zone": "zone2"})
        db.child("").update({"state": "1"})
        self.i=0
    else:
        self.i = self.i+1
        if(self.i==15):
            db.child("").update({"state": "0"})
            self.i=0


    matches=flann.knnMatch(queryDesc,trainDesc6,k=2)
    goodMatch=[]
    for m,n in matches:
        if(m.distance<0.75*n.distance):
            goodMatch.append(m)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(len(goodMatch)>MIN_MATCH_COUNT):
    tp=[]
    qp=[]
    for m in goodMatch:
        tp.append(trainKP6[m.trainIdx].pt)
        qp.append(queryKP[m.queryIdx].pt)
    tp,qp=np.float32((tp,qp))
    H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
    h,w=trainImg6.shape
    trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
    queryBorder=cv2.perspectiveTransform(trainBorder,H)
    cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(255,0,255),5)
    db.child("").update({"zone": "zone3"})
    db.child("").update({"state": "1"})
    self.i=0
else:
    self.i = self.i+1
    if(self.i==15):
        db.child("").update({"state": "0"})
        self.i=0


matches=flann.knnMatch(queryDesc,trainDesc7,k=2)
goodMatch=[]
for m,n in matches:
    if(m.distance<0.75*n.distance):
        goodMatch.append(m)
if(len(goodMatch)>MIN_MATCH_COUNT):
    tp=[]
    qp=[]
    for m in goodMatch:
        tp.append(trainKP7[m.trainIdx].pt)
        qp.append(queryKP[m.queryIdx].pt)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        tp,qp=np.float32((tp,qp))
        H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
        h,w=trainImg7.shape
        trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
        queryBorder=cv2.perspectiveTransform(trainBorder,H)
        cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(255,150,0),5)
        db.child("").update({"zone": "zone4"})
        db.child("").update({"state": "1"})
        self.i=0
    else:
        self.i = self.i+1
        if(self.i==15):
            db.child("").update({"state": "0"})
            self.i=0
    matches=flann.knnMatch(queryDesc,trainDesc8,k=2)
    goodMatch=[]
    for m,n in matches:
        if(m.distance<0.75*n.distance):
            goodMatch.append(m)
    if(len(goodMatch)>MIN_MATCH_COUNT):
        tp=[]
        qp=[]
        for m in goodMatch:
            tp.append(trainKP8[m.trainIdx].pt)
            qp.append(queryKP[m.queryIdx].pt)
        tp,qp=np.float32((tp,qp))
        H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
        h,w=trainImg8.shape
        trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
        queryBorder=cv2.perspectiveTransform(trainBorder,H)
        cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(255,180,255),5)
        db.child("").update({"zone": "zone5"})
        db.child("").update({"state": "1"})
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            self.i=0
        else:
            self.i = self.i+1
            if(self.i==15):
                db.child("").update({"state": "0"})
                self.i=0
        rawCapture.truncate(0)
        ret, jpeg = cv2.imencode('.jpg', QueryImgBGR)
        return jpeg.tobytes()

# method นี้เป็นส่วนของส่งไฟล์ภาพไปยังเว็บเบราเซอร์ เพื่อให้แอปพลิเคชัน สามารถเรียกดูแบบ
real time

def Modul1Modul3():
    print("-severcamera-")
    app = Flask(__name__)
    stop_run = False
    print("11")
    @app.route('/')
    def index():
        print("index")
        return render_template('index.html')
    def gen(camera):
        print("gen")
        while True:
            frame = camera.get_frame()
            yield (b'--frame\r\n'
                   b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n\r\n')


    @app.route('/video_feed')
    def video_feed():
        print("video_feed")
        return Response(gen(VideoCamera()),
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    mimetype='multipart/x-mixed-replace; boundary=frame')
    print("__name__")
    app.run(host='192.168.43.185',port = 5000, debug=False)
```

method คือส่วนของการทำงาน เพื่อควบคุมแบบ Manual โดยการส่งค่าผ่าน serial arduino แบบ UART

```
def Modul2():
    while True:
        testcheckonoff = db.child("/testonoff").get()
        if(testcheckonoff.val() == "off"):
            print("Wait")
        ser = serial.Serial('/dev/ttyACM0',9600)
        try:
            data=ser.readline()
            sdata=data.split("\t")
            x = sdata[0]
            sta = sdata[1]
        except:
            print("error")
        gpio.output(TRIG1, True)
        gpio.output(TRIG1, False)
        start1 = time.time()
        while gpio.input(ECHO1)==0:
            start1 = time.time()
        while gpio.input(ECHO1)==1:
            stop1 = time.time()
        measuredTime1 = stop1 - start1
        distanceBothWays1 = measuredTime1 * 33112
        distance1 = distanceBothWays1 / 2

        gpio.output(TRIG, True)
        gpio.output(TRIG, False)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
start = time.time()
while gpio.input(ECHO) == 0:
    start = time.time()
while gpio.input(ECHO) == 1:
    stop = time.time()
measuredTime = stop - start
distanceBothWays = measuredTime * 33112
distance = distanceBothWays / 2
sl=(int)(distance1)
sr=(int)(distance)
if x == "u":
    if s=='1':
        gpio.output(PWM_1,gpio.LOW)
        gpio.output(PWM_2,gpio.LOW)
    elif s=='0':
        gpio.output(DIR_1,gpio.HIGH)
        gpio.output(DIR_2,gpio.HIGH)
        gpio.output(PWM_1,gpio.HIGH)
        gpio.output(PWM_2,gpio.HIGH)
    if (sl >= 20 and sl <= 80) or (sr >= 20 and sr <= 80):
        s='1'
if x == "d":
    if s=='1':
        gpio.output(PWM_1,gpio.LOW)
        gpio.output(PWM_2,gpio.LOW)
    elif s=='0':
        gpio.output(DIR_1,gpio.LOW)
        gpio.output(DIR_2,gpio.LOW)
        gpio.output(PWM_1,gpio.HIGH)
        gpio.output(PWM_2,gpio.HIGH)
    if sta == "y":
        s='1'
if x == "l":
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            gpio.output(DIR_1,gpio.HIGH)
            gpio.output(DIR_2,gpio.LOW)
            gpio.output(PWM_1,gpio.HIGH)
            gpio.output(PWM_2,gpio.HIGH)
        if x == "r":
            gpio.output(DIR_1,gpio.LOW)
            gpio.output(DIR_2,gpio.HIGH)
            gpio.output(PWM_1,gpio.HIGH)
            gpio.output(PWM_2,gpio.HIGH)
        if x == "q":
            gpio.output(PWM_1,gpio.LOW)
            gpio.output(PWM_2,gpio.LOW)
            s = '0'
    gpio.cleaup
```

method ที่ควบคุมการเคลื่อนที่แบบ Automatic โดยจะดึงข้อมูลจาก Firebase มาทำการตรวจสอบและประมวลผลการเคลื่อนที่โดยการจับภาพผ่านกล้อง Raspberry pi

```
def Modul3():
    checkro = 0
    print("-Auto-")
    print("1")
    while True:
        autostatus = db.child("/autostatus").get()
        print("autostatus = "+autostatus.val())
        while autostatus.val()!="0":
            autostatus = db.child("/autostatus").get()
            if autostatus.val()=="1":
                print("this")
                i=1
                place = db.child("/place").get()
                zone = db.child("/zone").get()
                print(place.val())
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            print(zone.val())
            if(place.val()=="phamacy" or place.val()=="blood" or place.val()=="start" or
place.val()=="x-ray"):
                n = db.child("/whatzone/"+zone.val()+"/"+place.val()+"/n").get()
                nn=n.val()
                for i in range(n.val()):
                    x1="0"
                    num = i+1
                    print("i="+str(num)+"   n="+str(nn))
                    step                                                    =
db.child("/whatzone/"+zone.val()+"/"+place.val()+"/step"+str(num)).get()
                    if(step.val()=="up"):
                        print("up")
                        gpio.output(DIR_1,gpio.HIGH)
                        gpio.output(DIR_2,gpio.HIGH)
                        gpio.output(PWM_1,gpio.HIGH)
                        gpio.output(PWM_2,gpio.HIGH)
                        time.sleep(1)
                    if(step.val()=="left"):
                        print("left")
                        gpio.output(DIR_1,gpio.HIGH)
                        gpio.output(DIR_2,gpio.LOW)
                        gpio.output(PWM_1,gpio.HIGH)
                        gpio.output(PWM_2,gpio.HIGH)
                        time.sleep(1)
                        print("up")
                        gpio.output(DIR_1,gpio.HIGH)
                        gpio.output(DIR_2,gpio.HIGH)
                        gpio.output(PWM_1,gpio.HIGH)
                        gpio.output(PWM_2,gpio.HIGH)
                        time.sleep(1)
                    for frame in camera.capture_continuous(rawCapture, format="bgr",
use_video_port=True):
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
image = frame.array
QueryImgBGR=image
QueryImg=cv2.cvtColor(QueryImgBGR,cv2.COLOR_RGB2GRAY)
queryKP,queryDesc=detector.detectAndCompute(QueryImg,None)
matches=flann.knnMatch(queryDesc,trainDesc1,k=2)
goodMatch=[]
for m,n in matches:
    if(m.distance<0.75*n.distance):
        goodMatch.append(m)
if(len(goodMatch)>MIN_MATCH_COUNT):
    tp=[]
    qp=[]
    for m in goodMatch:
        tp.append(trainKP1[m.trainIdx].pt)
        qp.append(queryKP[m.queryIdx].pt)
    tp,qp=np.float32((tp,qp))
    H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
    h,w=trainImg1.shape
    trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
    queryBorder=cv2.perspectiveTransform(trainBorder,H)

cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(0,0,255),5)
    if(x1=="0"):
        x1="1"
        checkro=1


matches=flann.knnMatch(queryDesc,trainDesc2,k=2)
goodMatch=[]
for m,n in matches:
    if(m.distance<0.75*n.distance):
        goodMatch.append(m)
if(len(goodMatch)>MIN_MATCH_COUNT):
    tp=[]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                qp=[]
                for m in goodMatch:
                    tp.append(trainKP2[m.trainIdx].pt)
                    qp.append(queryKP[m.queryIdx].pt)
                tp,qp=np.float32((tp,qp))
                H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
                h,w=trainImg2.shape
                trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
                queryBorder=cv2.perspectiveTransform(trainBorder,H)

cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(0,255,0),5)
                if(x1=="0"):
                    x1="1"
                    checkro=1

                matches=flann.knnMatch(queryDesc,trainDesc3,k=2)
                goodMatch=[]
                for m,n in matches:
                    if(m.distance<0.75*n.distance):
                        goodMatch.append(m)
                if(len(goodMatch)>MIN_MATCH_COUNT):
                    tp=[]
                    qp=[]
                    for m in goodMatch:
                        tp.append(trainKP3[m.trainIdx].pt)
                        qp.append(queryKP[m.queryIdx].pt)
                    tp,qp=np.float32((tp,qp))
                    H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
                    h,w=trainImg3.shape
                    trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
                    queryBorder=cv2.perspectiveTransform(trainBorder,H)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(255,0,0),5)
```

```
            if(x1=="0"):
                x1="1"
                checkro=1

            matches=flann.knnMatch(queryDesc,trainDesc4,k=2)
            goodMatch=[]
            for m,n in matches:
                if(m.distance<0.75*n.distance):
                        goodMatch.append(m)
            if(len(goodMatch)>MIN_MATCH_COUNT):
                tp=[]
                qp=[]
                for m in goodMatch:
                    tp.append(trainKP4[m.trainIdx].pt)
                    qp.append(queryKP[m.queryIdx].pt)
                tp,qp=np.float32((tp,qp))
                H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
                h,w=trainImg4.shape
                trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
                queryBorder=cv2.perspectiveTransform(trainBorder,H)

cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(255,255,0),5)
                if(x1=="0"):
                    x1="1"
                    checkro=1

            matches=flann.knnMatch(queryDesc,trainDesc5,k=2)
            goodMatch=[]
            for m,n in matches:
                if(m.distance<0.75*n.distance):
                    goodMatch.append(m)
            if(len(goodMatch)>MIN_MATCH_COUNT):
                tp=[]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                qp=[]
                for m in goodMatch:
                    tp.append(trainKP5[m.trainIdx].pt)
                    qp.append(queryKP[m.queryIdx].pt)
                tp,qp=np.float32((tp,qp))
                H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
                h,w=trainImg5.shape
                trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
                queryBorder=cv2.perspectiveTransform(trainBorder,H)

cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(0,255,255),5)
                if(x1=="0"):
                    x1="1"
                    checkro=1

            matches=flann.knnMatch(queryDesc,trainDesc6,k=2)
            goodMatch=[]
            for m,n in matches:
                if(m.distance<0.75*n.distance):
                    goodMatch.append(m)
            if(len(goodMatch)>MIN_MATCH_COUNT):
                tp=[]
                qp=[]
                for m in goodMatch:
                    tp.append(trainKP6[m.trainIdx].pt)
                    qp.append(queryKP[m.queryIdx].pt)
                tp,qp=np.float32((tp,qp))
                H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
                h,w=trainImg6.shape
                trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
                queryBorder=cv2.perspectiveTransform(trainBorder,H)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(255,0,255),5)
```

```
            if(x1=="0"):
                x1="1"
                checkro=1

        matches=flann.knnMatch(queryDesc,trainDesc7,k=2)
        goodMatch=[]
        for m,n in matches:
            if(m.distance<0.75*n.distance):
                goodMatch.append(m)
        if(len(goodMatch)>MIN_MATCH_COUNT):
            tp=[]
            qp=[]
            for m in goodMatch:
                tp.append(trainKP7[m.trainIdx].pt)
                qp.append(queryKP[m.queryIdx].pt)
            tp,qp=np.float32((tp,qp))
            H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
            h,w=trainImg7.shape
            trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
            queryBorder=cv2.perspectiveTransform(trainBorder,H)

cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(255,150,0),5)
            if(x1=="0"):
                x1="1"
                checkro=1

        matches=flann.knnMatch(queryDesc,trainDesc8,k=2)
        goodMatch=[]
        for m,n in matches:
            if(m.distance<0.75*n.distance):
                goodMatch.append(m)
        if(len(goodMatch)>MIN_MATCH_COUNT):
            tp=[]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                qp=[]
                for m in goodMatch:
                    tp.append(trainKP8[m.trainIdx].pt)
                    qp.append(queryKP[m.queryIdx].pt)
                tp,qp=np.float32((tp,qp))
                H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
                h,w=trainImg8.shape
                trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])
                queryBorder=cv2.perspectiveTransform(trainBorder,H)

cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(255,180,255),5)
                if(x1=="0"):
                    x1="1"
                    checkro=1
                rawCapture.truncate(0)
                if((i+1)==nn):
                    db.child("").update({"autostatus": "0"})
                    db.child("").update({"place": "0"})
                    db.child("").update({"zone": "0"})
                if(checkro==1):
                    print("stop")
                    gpio.output(PWM_1,gpio.LOW)
                    gpio.output(PWM_2,gpio.LOW)
                    time.sleep(1)
                    x1="0"
                    checkro = 0
                    break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 โค้ดในฝั่งของแอปพลิเคชัน

```
# เป็นส่วนการส่งค่าขึ้น Firebase จากการเลือกสถานที่โดยผู้ใช้งาน เพื่อที่จะทำการเคลื่อนที่แบบ
Automatic
b1 = (Button)findViewById(R.id.b1);
        b2 = (Button)findViewById(R.id.b2);
        b3 = (Button)findViewById(R.id.b3);
        b4 = (Button)findViewById(R.id.b4);
        /*b5 = (Button)findViewById(R.id.b5);
        b6 = (Button)findViewById(R.id.b6);
        b7 = (Button)findViewById(R.id.b7);
        b8 = (Button)findViewById(R.id.b8);
        b9 = (Button)findViewById(R.id.b9);
        b10 = (Button)findViewById(R.id.b10);*/
        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Map<String, Object> value = new HashMap<String, Object>();
                Map<String, Object> value2 = new HashMap<String, Object>();
                value.put("place","phamacy");
                num.updateChildren(value);
                value2.put("autostatus","1");
                num.updateChildren(value2);
                Intent p1 = new Intent(auto.this,Traveling.class);
                startActivity(p1);
            }
        });
        b2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Map<String, Object> value = new HashMap<String, Object>();
                Map<String, Object> value2 = new HashMap<String, Object>();
                value.put("place","x-ray");
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            num.updateChildren(value);
            value2.put("autostatus","1");
            num.updateChildren(value2);
            Intent p1 = new Intent(auto.this,Traveling.class);
            startActivity(p1);
        }
    });
    b3.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Map<String, Object> value = new HashMap<String, Object>();
            Map<String, Object> value2 = new HashMap<String, Object>();
            value.put("place","blood");
            num.updateChildren(value);
            value2.put("autostatus","1");
            num.updateChildren(value2);
            Intent p1 = new Intent(auto.this,Traveling.class);
            startActivity(p1);
        }
    });
    b4.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Map<String, Object> value = new HashMap<String, Object>();
            Map<String, Object> value2 = new HashMap<String, Object>();
            value.put("place","start");
            num.updateChildren(value);
            value2.put("autostatus","1");
            num.updateChildren(value2);
            Intent p1 = new Intent(auto.this,Traveling.class);
            startActivity(p1);
        }
    });
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#เป็นส่วนของการรับค่าจาก Firebase ในการตรวจสอบว่ากล้องมีการจับภาพได้หรือไม่ได้ และรับค่าเบอร์โทรที่ได้ถูกตั้งค่าไว้

```
num.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
           Map map = (Map)dataSnapshot.getValue();
           String value = String.valueOf(map.get("number"));
           Map map1 = (Map)dataSnapshot.getValue();
           String value1 = String.valueOf(map1.get("state"));
           //String value1 = String.valueOf(map.get("state"));
           //Log.d("123",value1);
           ber = value;
           stcolor = value1;
           Log.d("1",stcolor);
           if(stcolor.equals("1"))
           {
              Log.d("1",stcolor);
              t.setBackgroundColor(Color.GREEN);
              Map<String, Object> value3 = new HashMap<String, Object>();
              Map<String, Object> value2 = new HashMap<String, Object>();
              value3.put("chackauto","on");
              num.updateChildren(value3);
              value2.put("testonoff","off");
              num.updateChildren(value2);
           }
           else
           {
              t.setBackgroundColor(Color.GRAY);
              Map<String, Object> value3 = new HashMap<String, Object>();
              Map<String, Object> value2 = new HashMap<String, Object>();
              value3.put("chackauto","off");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            num.updateChildren(value3);
            value2.put("testonoff","on");
            num.updateChildren(value2);
        }
        //ber1 = value1;
    }
```

เป็นส่วนการส่งค่าผ่าน Bluetooth เพื่อที่จะให้ผู้ใช้สามารถกดเลือกทิศที่ผู้ใช้ต้องการ

```
public void Disconnect ()
  {
    if (btSocket!=null) //If the btSocket is busy
    {
      try
      {
        btSocket.close(); //close connection
      }
      catch (IOException e)
      { msg("Error");}
    }
    finish(); //return to the first layout

  }
private void upset()
  {
    if (btSocket!=null)
    {
      try
      {
        btSocket.getOutputStream().write("u".toString().getBytes());
      }
      catch (IOException e)
      {
        msg("Error");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            }
        }
    }
private void downset()
    {
        if (btSocket!=null)
        {
            try
            {
                btSocket.getOutputStream().write("d".toString().getBytes());
            }
            catch (IOException e)
            {
                msg("Error");
            }
        }
    }
private void leftset()
    {
        if (btSocket!=null)
        {
            try
            {
                btSocket.getOutputStream().write("l".toString().getBytes());
            }
            catch (IOException e)
            {
                msg("Error");
            }
        }
    }
private void rightset()
    {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if (btSocket!=null)
        {
            try
            {
                btSocket.getOutputStream().write("r".toString().getBytes());
            }
            catch (IOException e)
            {
                msg("Error");
            }
        }
    }
```

เป็นส่วนของการตรวจล้มของรถนั่ง ถ้ามีการล้มเกิดขึ้นจะทำการโทรออกโดยจะเช็คจากแกน X แกน Y ที่ได้จาก accelerometer ภายใน Smartphone

```
public void onSensorChanged(SensorEvent event) {
        float x = event.values[0];
        float y = event.values[1];
        float z = event.values[2];
        Log.d("x:", String.valueOf(x));
        Log.d("y", String.valueOf(y));
        if(x<=-6||x>=6||y<=-1||y>=10)
        {
            Intent callIntent = new Intent(Intent.ACTION_CALL);
            callIntent.setData(Uri.parse("tel:"+ber));
            startActivity(callIntent);
        }
    }
```

เป็นส่วนของการตั้งค่าเบอร์โทรออกฉุกเฉิน โดยจะส่งค่าเบอร์โทรออกฉุกเฉินที่ผู้ใช้ได้กรอกข้อมูลไปจัดเก็บที่ Firebase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        x = (Button)findViewById(R.id.button1);
        x.setOnClickListener(new View.OnClickListener() {
```

```
        @Override
        public void onClick(View v) {
            String tmp = mphoneNumber.getText().toString();
            Map<String, Object> value = new HashMap<String, Object>();
            value.put("number",tmp);
            num.updateChildren(value);
            Intent p1 = new Intent(setting.this,page1.class);

            //Change the activity.
            p1.putExtra(EXTRA_ADDRESS, "20:16:11:14:23:44"); //this will be received at
ledControl (class) Activity
            startActivity(p1);

        }
    });
```

เป็นส่วนที่ เมื่อผู้ใช้กดโทรออกฉุกเฉิน จะทำการโทรออกฉุกเฉินไปยังเบอร์ที่ได้ตั้งค่าไว้

```
b = (Button) findViewById(R.id.call);
    b.setOnClickListener(new View.OnClickListener() {
        @SuppressLint("MissingPermission")
        @Override
        public void onClick(View v) {
            Intent callIntent = new Intent(Intent.ACTION_CALL);
            callIntent.setData(Uri.parse("tel:"+ber));
            startActivity(callIntent);
        }
    });
    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);
```

เป็นส่วนที่ เมื่อผู้ใช้กรอกค่าน้ำหนักระบบจะส่งค่าขึ้น Firebase เพื่อนำไปคำนวณความเร่งในการขับเคลื่อนมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    b1.setOnClickListener(new View.OnClickListener() {
        @Override
```

```
        public void onClick(View v) {
            sUsername = weightEditText.getText().toString();
            Log.d("dddddd",sUsername);

            if (sUsername.matches("")) {
            Toast.makeText(weight.this, "กรุณากรอกข้อมูลให้ครบ",
Toast.LENGTH_LONG).show();
        }
        else {
            String tmp = weightEditText.getText().toString();
            Map<String, Object> value = new HashMap<String, Object>();
            value.put("weight",tmp);
            num.updateChildren(value);
            Map<String, Object> value1 = new HashMap<String, Object>();
            value1.put("setweight","1");
            num.updateChildren(value1);
            gifImageView.setBackgroundResource(R.drawable.giffiv);
            new Timer().schedule(
                new TimerTask() {
                    @Override
                    public void run() {
                        Intent p1 = new Intent(weight.this, Devicelist.class);
                        startActivity(p1);
                    }
                },
                250
            );
            }
        }
    });
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 โค้ดในฝั่งของ Arduino

```
# เป็นส่วนของการที่จะรับข้อมูลผ่าน Bluetooth จากแอปพลิเคชันที่ได้ส่งมา และข้อมูลที่ได้จาก sensor เพื่อนำข้อมูลส่งต่อไปยัง Raspberry pi ผ่าน USB แบบ UART
void loop()
{
 if(Serial.available() > 0)
 {
   data = Serial.read();
 }
 if((sonar.ping_cm() >= 20 && sonar.ping_cm() <= 80)||(sonar1.ping_cm() >= 20 && sonar1.ping_cm() <= 80))
 {
     Serial.print(data);
     Serial.print("\t");
     Serial.print("y");
     Serial.println("\t");
 }
 else
 {
   Serial.print(data);
   Serial.print("\t");
;    Serial.print("n");
   Serial.println("\t");
 }
   delay(130);
}
```

4.3.4 อัลกอริทึมในการประมวลผลภาพ

```
import cv2
import numpy as np
MIN_MATCH_COUNT=50
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
# setup การทำงานของกล้อง
camera = PiCamera()
camera.resolution = (240, 180)
camera.framerate = 80
rawCapture = PiRGBArray(camera, size=(240, 180))

# เริ่มต้นเครื่องตรวจจับ SIFT
detector=cv2.SIFT()

# พารามิเตอร์ FLANN
FLANN_INDEX_KDITREE=0

# flannParam วิธีการสร้างโครงสร้างการค้นหาเพื่อนบ้าน ใช้วิธีดัชนีที่สร้างขึ้นที่จะประกอบด้วยชุด
ต้นไม้ kd แบบสุ่มที่จะค้นหาแบบขนาน

#search_params = dict (checks = 50)   # หรือส่งผ่านพจนานุกรมเปล่า

#SearchParams ระบุจำนวนครั้งที่ต้นไม้ในดัชนีควรผ่านการสำรวจซ้ำ ค่าที่สูงขึ้นจะให้ความแม่นยำ
ที่ดีขึ้น แต่ต้องใช้เวลามากขึ้น
flannParam=dict(algorithm=FLANN_INDEX_KDITREE,tree=5)
flann=cv2.FlannBasedMatcher(flannParam,{})

# ไฟล์ภาพที่ถูกโหลดเพื่อเอามาเปรียบเทียบกับภาพที่ได้จากกล่อง Pi
trainImg1=cv2.imread("sum/pzone1.jpg",0)

# ค้นหา keypoints และ descriptors ที่มี SIFT ของ ไฟล์ภาพ
trainKP1, trainDesc1=detector.detectAndCompute(trainImg1, None)

# วนลูปเพื่อทำการใช้การถ่ายภาพแบบVideoรับเป็น frame
for frame in camera.capture_continuous(rawCapture, format="bgr",
use_video_port=True):
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    # แปลงภาพที่ได้จากกล้องเป็น array
```

```
image = frame.array
QueryImgBGR=image

# ปรับภาพที่ได้จากกล้อง Raspberry Pi เป็นสีเทา
QueryImg=cv2.cvtColor(QueryImgBGR, cv2.COLOR_RGB2GRAY)

# ค้นหา keypoints และ descriptors ที่มี SIFT ของ ภาพได้จากกล้อง Raspberry Pi ที่ถูกปรับเป็นสีเทาแล้ว
queryKP,queryDesc=detector.detectAndCompute(QueryImg,None)

# จับคู่คุณลักษณะ flann เพื่อให้ตรงกับคุณสมบัติในภาพทั้งสองภาพ และเก็บผลลัพธ์ที่ตรงกันไว้ในตัวแปรที่ matches
matches=flann.knnMatch(queryDesc,trainDesc1,k=2)
goodMatch=[]
# กรองการจับคู่เพื่อหลีกเลี่ยงการจับคู่แบบเท็จ
# ตรวจสอบพิสัยจากเพื่อนบ้านที่มีค่าเหมาะสมที่สุด m และเพื่อนบ้านที่ใกล้เคียงที่สุด n และผู้พัฒนากำลังพิจารณาการจับคู่เป็นการจับคู่ที่ดีถ้าพิสัยจากจุด "m" น้อยลง 70% ของพิสัยที่จุด "n" และต่อท้ายชี้ไปที่ "goodMatch"
for m,n in matches:
    if(m.distance<0.75*n.distance):
        goodMatch.append(m)

#นอกจากนี้ผู้พัฒนาต้องตรวจสอบให้แน่ใจว่ามีการจับคู่คุณลักษณะเพียงพอสำหรับการเรียกการจับคู่เหล่านี้เนื่องจากจะกำหนดเกณฑ์ "MIN_MATCH_COUNT" และหากจำนวนการจับคู่สูงกว่าค่านั้น ต้องพิจารณาว่า การจับคู่ ดังนั้นในโค้ดด้านบน ผู้พัฒนาต้องตรวจสอบก่อนว่าหมายเลขของคุณสมบัติที่ตรงกันมีจำนวนมากกว่าเกณฑ์ขั้นต่ำ
if(len(goodMatch)>MIN_MATCH_COUNT):
    tp=[]
    qp=[]

    #สร้างช่องว่างเปล่าเพื่อรับพิกัดของคุณลักษณะที่ตรงกันจากภาพที่ได้จากไฟล์รวมทั้งจากรูปแบบข้อความค้นหาและแปลงรายการนั้นเป็น numpy
    for m in goodMatch:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                tp.append(trainKP1[m.trainIdx].pt)
                qp.append(queryKP[m.queryIdx].pt)
            tp,qp=np.float32((tp,qp))

            # cv2.findHomography (tp, qp, cv2.RANSAC, 3.0) เพื่อหาค่าคงที่ในการแปลจุดจากจุดภาพที่ได้จากไฟล์ ไปยังจุดภาพที่ได้จากกล้อง
            H,status=cv2.findHomography(tp,qp,cv2.RANSAC,3.0)
            h,w=trainImg1.shape

            # การวาดเส้นขอบรอบวัตถุเพื่อรับพิกัดของมุมขอบจากภาพการได้จากไฟล์ซึ่งเป็น (0,0), (0, h-1), (w-1, h-1) , (w-1,0) โดยที่ h, w คือความสูงและความกว้างของภาพที่ได้จากไฟล์
            trainBorder=np.float32([[[0,0],[0,h-1],[w-1,h-1],[w-1,0]]])

            # ใช้ค่าคงที่การแปลง "H" ที่ได้รับจากก่อนหน้านี้ จะแปลพิกัดจากภาพการที่ได้จากไฟล์ไปเป็นรูปแบบข้อความค้นหา
            queryBorder=cv2.perspectiveTransform(trainBorder,H)

            # "cv2.polylines ()" เพื่อวาดเส้นขอบในรูปแบบข้อความค้นหา
            cv2.polylines(QueryImgBGR,[np.int32(queryBorder)],True,(0,0,255),5)
            db.child("").update({"zone": "zone1"})
            db.child("").update({"state": "1"})
            self.i=0

        # ถ้าจำนวนของคุณสมบัติน้อยกว่าที่การนับขั้นต่ำนับแล้วพิมพ์
len(goodMatch),MIN_MATCH_COUNT
        else:
            self.i = self.i+1
            if(self.i==15):
                db.child("").update({"state": "0"})
                self.i=0
        rawCapture.truncate(0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการดำเนินงานและข้อเสนอแนะ

ในการพัฒนาแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ ผู้พัฒนาได้พบกับปัญหาต่าง ๆ ในระหว่างการพัฒนา ซึ่งได้แก้ปัญหา พัฒนาระบบจนเสร็จสมบูรณ์ และได้ผลสรุปของการพัฒนาแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติดังนี้

5.1 สรุปผลการดำเนินงาน

แอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ ถูกพัฒนาขึ้นโดยมีจุดประสงค์เพื่อเพิ่มความสะดวกสบายให้กับผู้ใช้งานรถนั่งอัตโนมัติ ผู้ใช้สามารถควบคุมรถนั่งผ่านแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ โดยมีฟังก์ชันเพิ่มความปลอดภัยในการใช้งาน ได้แก่ ฟังก์ชันหยุดการเคลื่อนที่เมื่อสิ่งกีดขวางหรือหยุดตามความต้องการของผู้ใช้ และฟังก์ชันแจ้งเตือนเมื่อรถล้ม โดยจะทำการโทรออกไปยังเบอร์ผู้ดูแลภายในโรงพยาบาลเมื่อเกิดเหตุฉุกเฉิน ผู้ใช้งานรถนั่งสามารถควบคุมการเคลื่อนที่ของรถนั่งไปในทิศทางต่าง ๆ หรือเลือกสถานที่ที่ต้องการจะไปผ่านแอปพลิเคชันเพื่อให้รถเคลื่อนที่โดยอัตโนมัติ โดยใช้การหลักการประมวลผลภาพดิจิทัลเพื่อจับภาพป้ายที่ติดตั้งไว้ภายในโรงพยาบาล

แอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ ที่ใช้ควบคุมรถนั่งถูกพัฒนาขึ้นโดยใช้ภาษา Python, JAVA,PHP และ HTML ติดตั้งการใช้งานบนระบบปฏิบัติการแอนดรอยด์ ผู้พัฒนาได้ใช้เทคโนโลยีเซนเซอร์วัดความเร่งที่มีในตัวโทรศัพท์ (Accelerometer) ในการตรวจจับการล้มของรถนั่งและเซนเซอร์ตรวจจับระยะห่างระหว่างวัตถุเพื่อความปลอดภัยในการใช้งาน มีการใช้ Raspberry Pi และ Arduino ในระบบเพื่อเพิ่มประสิทธิภาพในการทำงาน และใช้หลักการประมวลผลภาพดิจิทัลเพื่อทำการตรวจจับภาพป้ายตามจุดต่าง ๆ ภายในโรงพยาบาลเพื่อการเคลื่อนที่ไปยังจุดต่างๆโดยอัตโนมัติ รถนั่งอัตโนมัติเหมาะสำหรับผู้สูงอายุ และผู้พิการทางขา ที่ต้องการความสะดวกสบาย

5.2 ปัญหาที่พบระหว่างการพัฒนาแอปพลิเคชัน

1) ในการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ ผู้พัฒนาต้องทำความคุ้นเคยกับเครื่องมือที่ใช้พัฒนา เรียนรู้พื้นฐานของการพัฒนาแอปพลิเคชัน และต้องให้เวลาในทดลองมากๆ
2) เนื่องจากมีการใช้ Raspberry Pi และ Arduino มาใช้ในการพัฒนาด้วย ผู้พัฒนาจึงต้องทำการศึกษาพื้นฐานการใช้อุปกรณ์เหล่านี้ด้วย
3) การประมวลผลภาพที่ไม่ชัดเจน เนื่องจาก แสง เงา สิ่งบทบังในขณะประมวลผลภาพ
4) Raspberry Pi มีทรัพยากรไม่พอในการประมวลผลภาพทำให้เกิดภาพที่ดีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) น้ำหนักของผู้ใช้งานมีผลต่อทำให้ความเร็วของรถนั่ง

5.3 ข้อจำกัดของระบบ

1) ในการประมวลผลภาพจำเป็นต้องมีแสงช่วยให้เห็นภาพที่ชัด ไม่ควรให้ป้ายอยู่ในที่ไม่มีแสงหรือเงาสะท้อน
2) ในการใช้ฟังก์ชันเคลื่อนที่อัตโนมัติ จำเป็นต้องเป็นเส้นทางวันเวย์ เท่านั้น
3) ไม่สามารถเพิ่มเส้นทางได้ในฟังก์ชันเคลื่อนที่อัตโนมัติได้
4) ไม่สามารถจับโทรศัพท์ที่ใช้แอปพลิเคชันขึ้นมา จากแท่นวางได้เนื่องจาก มีการใช้เทคโนโลยีเซนเซอร์วัดความเร่งที่มีในตัวโทรศัพท์ (Accelerometer) ในการตรวจจับการล้มของรถนั่ง
5) ในขณะเคลื่อนที่อัตโนมัติจำกัดความเร็วเนื่องจากกล้องที่มีความละเอียดต่ำและทรัพยากร ของ Raspberry Pi ไม่พอต่อการประมวลผลภาพทำให้เกิดภาพที่ดีเลย์

5.4 ข้อเสนอแนะ

1) พัฒนาแอปพลิเคชันให้สามารถเพิ่มเส้นทางได้ โดยใช้การกรอกข้อมูลลงในแอปพลิเคชันเพื่อเพิ่มเส้นทางไปยังฐานข้อมูล
2) พัฒนาแอปพลิเคชันให้สามารถไปและกลับได้ โดยที่ไม่จำเป็นต้องใช้เส้นทางที่เป็นทางเดียว
3) พัฒนาแอปพลิเคชันให้รองรับระบบปฏิบัติการ iOS
4) ในการประมวลผลภาพโดยไม่ให้เกิดการดีเลย์ ต้องใช้บอร์ดคอมพิวเตอร์ขนาดเล็กที่มีทรัพยากรที่เพียงพอต่อการประมวลผลภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] 2017.**การใช้งาน Bluetooth module HC05 แบบละเอียด**.สืบค้น 15 กันยายน. 2558.[Online].Available : https://www.ioxhop.com

[2] 2012.**บทความการพัฒนาโปรแกรมบนRaspberry pi**,15 กันยายน.2558.[Online],Available : http://www.thaieasyelec.com

[3] 2012.**บอร์ดขับมอเตอร์ไฟกระแสตรง2ช่อง,**16 กันยายน.2558.[Online],Available : http://www.thaieasyelec.com

[4] 2012.**Ultrasonic Module JSN-SR04T,**17 กันยายน.2558.[Online],Available : http://www.thaieasyelec.com

[5] 2013.**รถเซ็นวีลแชร์,**17 กันยายน.2558.[Online],Available : http://www.themobilitythailand.com

[6] 2017.**Python,**18 กันยายน.2558.[Online],Available : http://www.mindphp.com

[7] 2015.**โปรแกรม Android Studio,**18 กันยายน.2558.[Online],Available : http://www.thaicreate.com

[8] 2016.**พิการทางการเคลื่อนไหว,**19 กันยายน.2558.[Online],Available : http://202.151.176.107:8080/public/blog.do?cmd=goView&id=17

[9] 2016. **Arduino,**20 กันยายน.2558.[Online],Available : http://www.thaieasyelec.com

[10] 2016. **Firebase,**20 กันยายน.2558.[Online],Available : https://firebase.google.com/

[11] 2012.**SiFT,**16 กันยายน.2558.[Online],Available : https://endcodes.wordpress.com/2014/06/02/sift-scale-invariant-feature-transform/

[12] 2016.**Android Sensor Accelerometer,**19 กันยายน.2558.[Online],Available : http://www.akexorcist.com/2013/03/android-code-accelerometer.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

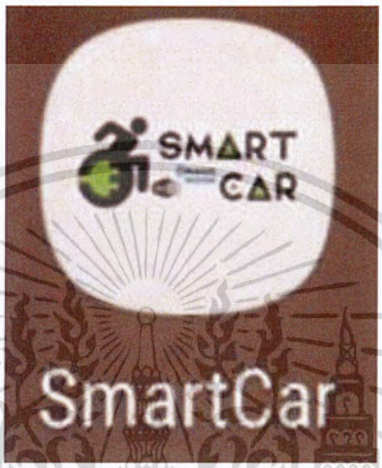
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

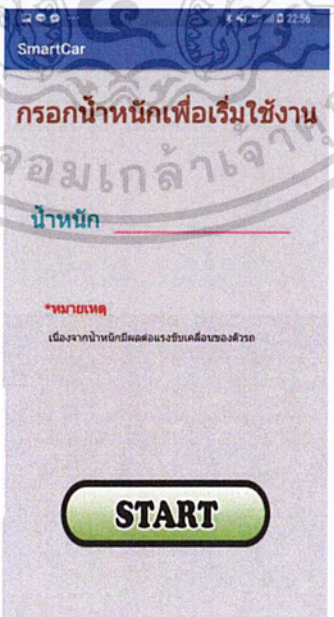
คู่มือการใช้งานอย่างละเอียด

1) เมื่อติดตั้งเสร็จเรียบร้อย ผู้ใช้งานสามารถดาวน์โหลดแอปพลิเคชันมาใช้งานได้ทันทีแสดงดังรูปที่ ก.1



รูปที่ ก.1 ไอคอนของแอปพลิเคชันบนสมาร์ทโฟน

2) เมื่อเปิดแอปพลิเคชันขึ้นมาแล้วจะปรากฏหน้าจอแรกของแอปพลิเคชันให้กรอกน้ำหนักเพื่อใช้น้ำหนักอ้างอิงกับความเร็วเริ่มต้นในการเคลื่อนที่ทั้ง 2 โหมด แสดงดังรูปที่ ก.2



รูปที่ ก.2 หน้าจอแรกให้กรอกน้ำหนักผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) หลังจากกดปุ่ม START แล้วจะต้องทำการเชื่อมต่อบลูทูธโดยเลือกปุ่ม “อนุญาต” เพื่อเปิดบลูทูธ ดังรูปที่ ก.3

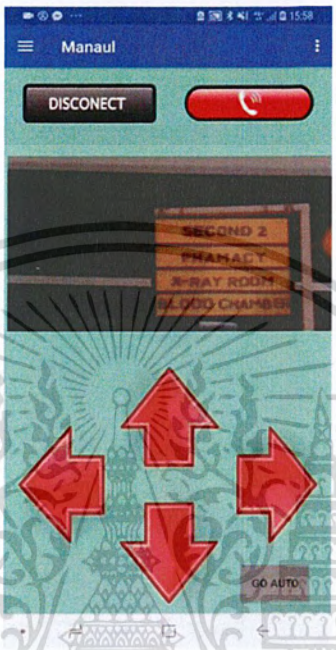
รูปที่ ก.3 หน้าจออนุญาตเปิดบลูทูธ

4) การค้นหาสัญญาณบลูทูธโดยการเลือก Paired Devices เพื่อทำการค้นหาสัญญาณ ดังรูปที่ ก.4

รูปที่ ก.4 หน้าจอเชื่อมต่อบลูทูธ

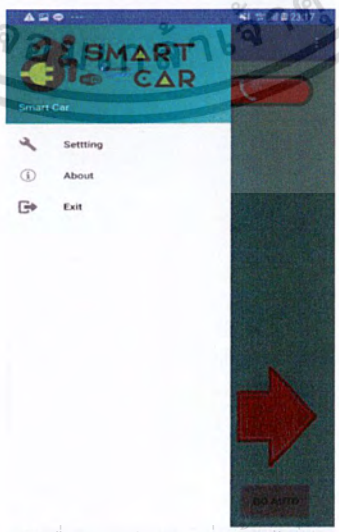
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) เป็นหน้าจอที่หลังเชื่อมต่อบลูทูธได้แล้ว จะสามารถใช้กล้องประมวลผลภาพได้ เลือกเมนูได้ และสามารถใช้โหมด Manual ได้ ในโหมดนี้สามารถ เลือกลูกศรเคลื่อนที่ไปได้ 4 ทิศทาง ได้แก่ เดินหน้า ถอยหลัง หันซ้าย และหันขวา ดังรูปที่ ก.5



รูปที่ ก.5 หน้าจอโหมด Manual

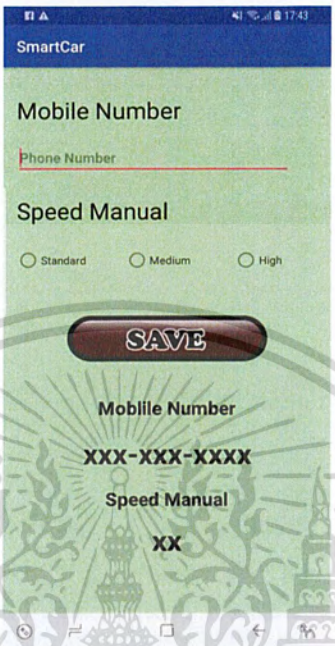
6) ก่อนใช้งานทุกโหมดจำเป็นต้องเข้า ปุ่ม icon Hamburger เพื่อเลือกปุ่ม Setting เพื่อตั้งค่า และเลือกปุ่ม Help เพื่อทำการเรียนรู้ก่อนใช้งาน ดังรูปที่ ก.6



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ ก.6 หน้าจอแถบเมนู

7) หลักจากเลือกปุ่ม Setting เพื่อตั้งค่าเบอร์โทรฉุกเฉิน ตั้งค่าความเร็วในโหมด Manual สำหรับคนที่ต้องการความเร็วที่มากกว่าระบบตั้งไว้ในการอ้างอิงจากน้ำหนัก ดังรูปที่ ก.7



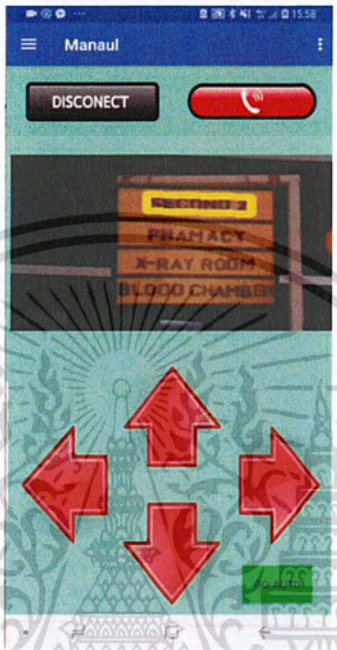
รูปที่ ก.7 หน้าจอ Setting

8) หลักจากเลือกปุ่ม Help เพื่ออ่านแนะนำวิธีการใช้งาน ดังรูปที่ ก.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ ก.8 หน้าจอ Help

9) หลังจากตั้งค่าและอ่านคำแนะนำการใช้งานเสร็จเรียบร้อย ให้กลับมาที่หน้าจอโหมด Manual และทำการเคลื่อนที่ให้ภาพประมวลผล เพื่อทำการใช้งานโหมด Manual ได้ ดังรูปที่ ก.9



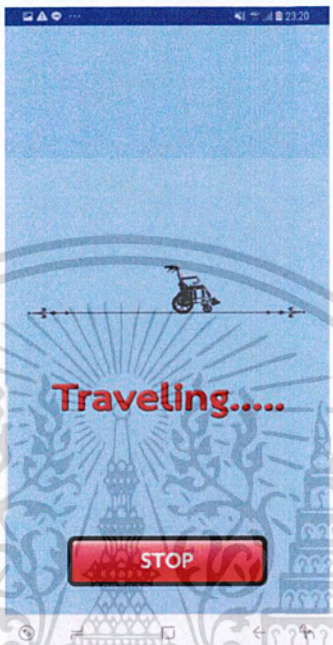
รูปที่ ก.9 การประมวลผลภาพเพื่อให้ปุ่ม Auto ใช้งานได้

10) หลังจากเลือกปุ่ม GO AUTO จะมีปุ่มแต่ละตำแหน่งให้เลือกเพื่อทำการเคลื่อนที่ไปยังเป้าหมายที่ผู้ใช้ต้องการจะไป ดังรูปที่ ก.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ ก.10 หน้าจอโหมด Auto

11) หลังจากเลือกตำแหน่งที่ต้องการแล้ว รถนั่งจะทำการเคลื่อนที่ไปยังตำแหน่งที่ต้องการจะไป และในขณะที่รถนั่งกำลังเคลื่อนที่อยู่สามารถหยุดรถนั่งเพื่อกลับไปประมวลผลภาพใหม่ ดังรูปที่ ก.11



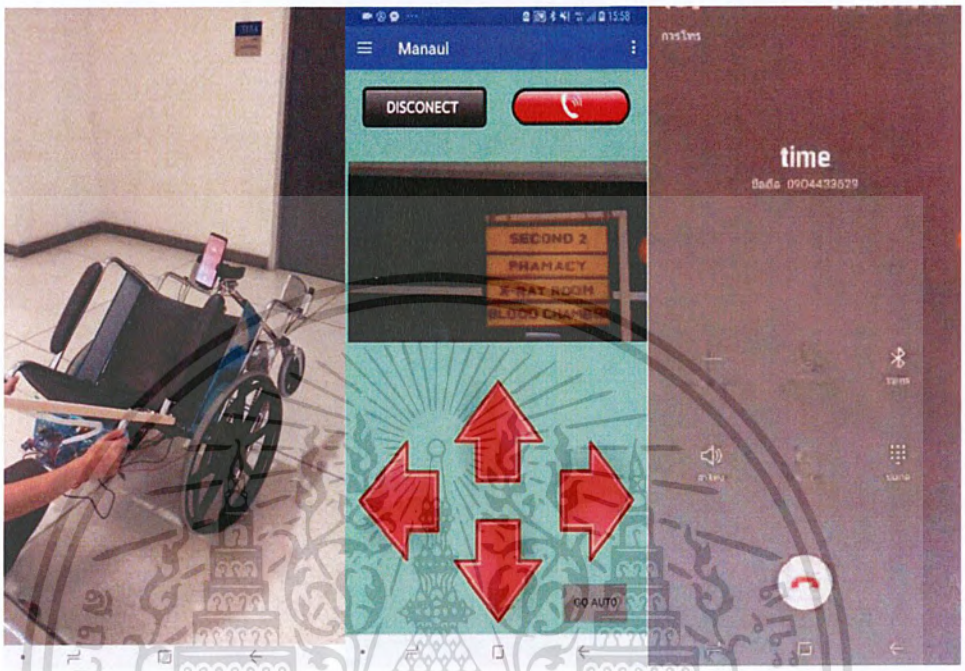
รูปที่ ก.11 หน้าจอกำลังเคลื่อนที่ในโหมด Auto

เมื่อเคลื่อนที่ถึงตำแหน่งที่ผู้ใช้ต้องการจะไป รถนั่งจะหยุดและขึ้นสถานะในแอปพลิเคชัน ว่า “ถึงจุดหมายปลายทางของท่านแล้ว” ดังรูปที่ ก.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

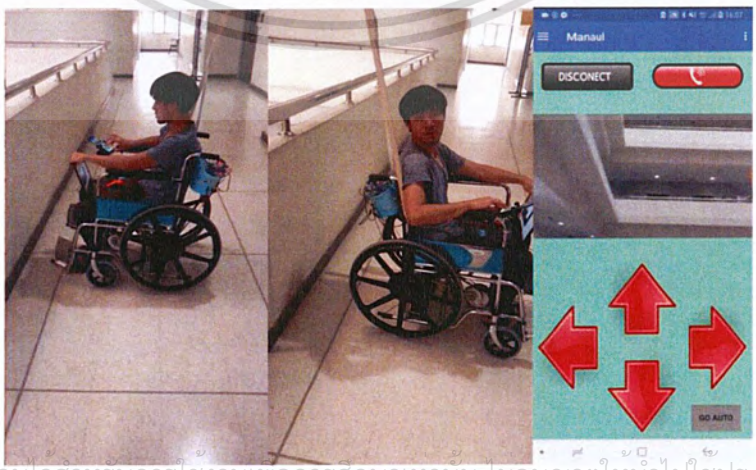
รูปที่ ก.12 สถานะบอกว่าถึงตำแหน่งที่ต้องการจะไปในโหมด Auto

12) เมื่อเกิดเหตุการณ์รถนั่งล้มจะมีการโทรออกอัตโนมัติ หรือถ้าผู้ใช้เกิดเหตุการณ์มึนงงในขณะที่ใช้โหมด Manual สามารถโทรแจ้งโดยกดปุ่ม โทรสีแดง ดังรูปที่ ก.13



รูปที่ ก.13 การโทรออกไปยังผู้ดูแล

13) เมื่อมีสิ่งกีดขวางในขณะเคลื่อนที่ในโหมด Manual รถจะหยุดและไม่สามารถเลือกลูกศรในแอปพลิเคชันในทิศทางที่มีสิ่งกีดขวางได้ และในโหมด Auto หยุดเมื่อสิ่งกีดขวางผ่านไปจะทำงานต่อจนถึงตำแหน่งที่เลือกไว้ ดังรูปที่ ก.14



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ ก.14 เมื่อมีสิ่งกีดขวาง

ภาคผนวก ข

ผลงานที่ได้รับรางวัล

ข.1 การประชุมวิชาการระดับปริญญาตรีด้านคอมพิวเตอร์ระดับภูมิภาคอาเซียน 2018 (The ASEAN Undergraduate Conference in Computing: AUC2 2018)

1) ส่งเข้าแข่งขันในประเภทการนำเสนอด้วยวาจา (Oral Presentation) ผลงานที่ได้รับรางวัล Good Paper Award รางวัลที่ได้รับ แสดงดังรูปที่ ข.1

The 6th ASEAN Undergraduate Conference in Computing (AUC² 2018)

Certificate of Participation

presented to

ศุภากร อริยมงคลชัย, สิทธิกร อริยมงคลชัย, กษิดิศ แสงสีทอง, ณพชร จีงธนโรจน์ และ วรางคณา กิ้มปาน

for a contribution in the field of computing entitled

รถนั่งอัตโนมัติสำหรับผู้สูงอายุควบคุมโดยแอนดรอยด์แอปพลิเคชัน

held on 23rd – 25th March 2018 at Faculty of Science,
King Mongkut's Institute of Technology Ladkrabang, Thailand

Assoc. Prof. Dr. Ittipol Jangchud
General Chair of AUC² 2018

รูปที่ ข.1 ประกาศนียบัตรที่ได้รับ

2) ผลงานที่น้าไปแข่งขัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถนั่งอัตโนมัติสำหรับผู้สูงอายุควบคุมโดยแอนดรอยด์แอปพลิเคชัน

AUTOMATIC WHEELCHAIR FOR ELDERLY CONTROLLED BY ANDROID APPLICATION

ศุภากร อริยมงคลชัย สิทธิกร อริยมงคลชัย กษิดิศ แสงสีทอง ณพชร จีงธนโรจจน์ และวรางคณา กิ้มปาน
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
Emails: tide1122@hotmail.com, time8899@hotmail.com, kasidit.t1995@gmail.com, napatchara.c@gmail.com

บทคัดย่อ

บทความนี้นำเสนอแอนดรอยด์แอปพลิเคชันสำหรับควบคุมรถนั่งอัตโนมัติ เหมาะสำหรับผู้ที่พิการทางขา ซึ่งไม่สะดวกและไม่สามารถเคลื่อนไหวที่ขาได้นอกจากนั้นผู้สูงอายุทั่วไปก็สามารถใช้งานแอปพลิเคชันนี้ได้ ซึ่งแอปพลิเคชันที่สร้างขึ้นนี้จะใช้ควบคุมรถนั่งอัตโนมัติ สำหรับการใช้งานในโรงพยาบาล ใช้หลักการประมวลผลภาพ โดยใช้ป้ายบอกทางภายในโรงพยาบาลในการประมวลผลภาพเป็นแนวในการนำทางไปยังจุดมุ่งหมายของสถานที่ต่าง ๆ ที่แสดงในโรงพยาบาลได้โดยอัตโนมัติ อีกทั้งยังมีระบบที่ควบคุมด้วยมือ ที่ให้ผู้ใช้สามารถใช้ควบคุมรถนั่งอัตโนมัติด้วยตนเอง โดยทั้ง 2 การควบคุมจะมีเซ็นเซอร์ตรวจจับความปลอดภัย 2 รูปแบบ ซึ่งผู้ใช้รถนั่งเกิดใช้งานรถนั่งล้มจะมีการแจ้งเตือนไปยังผู้ดูแล และในขณะที่ผู้ใช้ควบคุมรถนั่งอัตโนมัติเคลื่อนที่จะมีเซ็นเซอร์ตรวจจับสิ่งกีดขวางเมื่อเจอสิ่งกีดขวางรถนั่งอัตโนมัติจะไม่สามารถเคลื่อนที่ไปยังทิศทางที่มีสิ่งกีดขวางได้ จึงทำให้รถนั่งมีความปลอดภัยและมีทิศทางถูกต้องในการนำทางไปยังจุดมุ่งหมายของผู้ใช้งาน

ABSTRACT

This paper proposed the Android application for controlling automatic wheelchair. This application is convenient for legs disabled person or who is not convenient about using legs and also elderly person. The application was developed to control the automatic chair which will be used in hospitals. There are 2 modes to control the system: manual and automatic. In manual mode, the user can use the arrows which are left, right, forward, and back to move the wheelchair to the destination. The user can also control the wheelchair in automatic mode by using the principle image processing via web camera and display in the Android application. The image processing will lead the wheelchair automatically by reading the guide posts in the hospital and it automatically moves the wheelchair to the destination. Moreover, there are 2 types of sensors to detect the safety. In case of wheelchair is falling, the application will send the notification to other persons. In case of detecting the barrier the wheelchair will be automatically stopped.

คำสำคัญ— รถนั่งอัตโนมัติ;หลักการประมวลผลภาพ;เซ็นเซอร์; แอนดรอยด์แอปพลิเคชัน;โรงพยาบาล

1. บทนำ

ปัจจุบันเทคโนโลยีมีบทบาทต่อการดำเนินชีวิตเป็นอย่างมาก และเนื่องจากประเทศไทยกำลังก้าวเข้าสู่สังคมผู้สูงอายุ อีกทั้งยังมีการให้ความสำคัญแก่ผู้พิการมากขึ้น การพัฒนาเทคโนโลยีเพื่อช่วยอำนวยความสะดวกให้กับผู้สูงอายุและผู้พิการทางขาในสถานที่ต่าง ๆ เช่น โรงพยาบาล เป็นต้น จึงเป็นสิ่งสำคัญ จากการศึกษาพบว่าเทคโนโลยีสามารถนำมาประยุกต์ใช้เพื่อช่วยเหลือ เพิ่มความปลอดภัย และอำนวยความสะดวกในการเคลื่อนที่ไปยังตำแหน่งต่างๆ ภายในโรงพยาบาลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเอกสารนี้ทุกครั้งที่มีการนำไปใช้

ผู้พัฒนาได้ทำการศึกษาและพัฒนาระบบแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ โดยมีการควบคุมการใช้งานผ่านบนระบบปฏิบัติการแอนดรอยด์ที่สามารถควบคุมการทำงานของรถวีลแชร์ธรรมดาได้ มีฟังก์ชันการทำงานหลัก 2 ฟังก์ชัน ได้แก่ ฟังก์ชันควบคุมโดยผู้ใช้งาน (Manual) โดยผู้ใช้งานสามารถควบคุมรถนั่งอัตโนมัติด้วยตนเองได้เพื่อให้รถนั่งอัตโนมัติเคลื่อนที่ไปในทิศทางที่ต้องการ และฟังก์ชันอัตโนมัติ (Automatic) โดยผู้ใช้สามารถเลือกสถานที่เป้าหมายเพื่อให้ รถนั่งอัตโนมัติเคลื่อนที่ไปโดยอัตโนมัติ ระบบสามารถแจ้งเตือนผ่านแอปพลิเคชันเมื่อรถนั่งอัตโนมัติเกิดอุบัติเหตุพลิกคว่ำเพื่อร้องขอความช่วยเหลือกับผู้ดูแลในสถานที่นั้น ๆ และสามารถตรวจจับวัตถุเพื่อป้องกันการชน

2. ทฤษฎีที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้องกับการพัฒนาแอนดรอยด์แอปพลิเคชันควบคุมรถนั่งอัตโนมัติ ได้แก่ ทฤษฎีการประมวลผลภาพ, Android Studio, ภาษา Python, Arduino IDE และ SQLite Database Browser โดยมีรายละเอียดดังต่อไปนี้

2.1 ทฤษฎีการประมวลผลภาพ (Image Processing)

ทฤษฎีการประมวลผลภาพ [1] คือการเรียกใช้ขั้นตอน หรือกรรมวิธีใด ๆ มากระทำกับภาพโดยมีวัตถุประสงค์ เพื่อปรับปรุงคุณภาพของภาพ และให้ได้ภาพใหม่ที่มีคุณสมบัติตามความต้องการ เช่นการกำจัดสัญญาณรบกวน การจำแนกภาพ การหาขอบภาพ การเปลี่ยนแปลงภาพ การทำให้ภาพคมชัด เป็นต้น โดยการประมวลผลในระดับต่ำจะใช้ค่าความสว่าง หรือระดับความเข้มของจุดภาพโดยตรง ซึ่งถือว่าเป็นกระบวนการเบื้องต้นที่มีความสำคัญมาก ความคมชัดหรือการประหยัดพื้นที่ ในการเก็บข้อมูล หรือใช้สำหรับการประมวลผลระดับสูง เช่น การจดจำรูปร่างลักษณะให้ได้อย่างแม่นยำ โดยทั่วไปแล้ววัตถุประสงค์ของการประมวลผลภาพ (Image Processing) คือ

1) **Image Processing (Image In Image Out)** เป็นขั้นตอนหรือวิธีที่มีการใช้กระบวนการทางการประมวลผลภาพ เพื่อทำให้ได้ภาพออกมาอย่างสมบูรณ์ เช่น การตกแต่งภาพด้วยโปรแกรม Photoshop เป็นต้น
2) **Image Analysis (Image In Measurements Out)** เป็นขั้นตอนหรือวิธีที่มีการใช้กระบวนการทางการประมวลผลภาพ เพื่อทำให้ได้ค่าการวัดออกมา เช่นการวัดขนาดในงานอุตสาหกรรมเป็นต้น
3) **Image Understanding (Image In High-Level Description Out)** เป็นขั้นตอนหรือวิธีที่มีการใช้กระบวนการทางการประมวลผลภาพ เพื่อให้ได้ผลลัพธ์ออกมาเป็นความหมาย และความหมายนั้นสามารถที่จะทำให้เกิดประโยชน์ สามารถนำมาประยุกต์ใช้ต่อได้ ตัวอย่างเช่น การรู้จำหรือการจดจำตัวอักษร (Optical Character Recognition : OCR)

โดยรูปแบบในการประมวลผลภาพสามารถแบ่งออกได้เป็น 2 ระดับคือ การประมวลผลภาพในระดับต่ำ (Low-Level Image Processing) และการประมวลผลภาพในระดับสูง (High-Level Image Processing)

2.2 Python Language

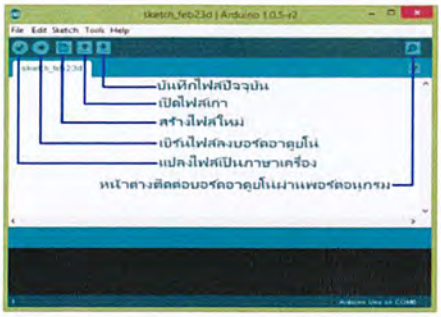
Python [2] เป็นภาษาระดับสูงภาษาที่เป็น Open Source หนึ่ง ที่มีความสามารถสูงถูกสร้างขึ้นในปี 1989 โดย Guido van Rossum ซึ่งถูกพัฒนาขึ้นมาโดยไม่ยึดติดกับแพลตฟอร์ม กล่าวคือสามารถรันภาษา Python ได้ทั้งบนระบบ Unix, Linux, Windows NT, Windows 2000, Windows XP หรือแม้แต่ระบบ FreeBSD

ซึ่งในปัจจุบันภาษาที่ใช้ในการพัฒนาเว็บแอพพลิเคชันมีมากมาย อาทิเช่นภาษาPerl, PHP, JAVA, ASP, Tcl, Python เป็นต้น สำหรับภาษา Python นับว่ายังใหม่ในวงการพัฒนาโปรแกรมบนเว็บ แต่ด้วยข้อดีหลายประการของภาษา Python ทำให้มีผู้นิยมใช้มากขึ้น

2.3 Arduino IDE

Arduino IDE [3] เป็นระบบเปิด (Open Source) ทั้งฮาร์ดแวร์และซอฟต์แวร์ ดังนั้นในการเขียนโปรแกรมควบคุมการทำงานของบอร์ด Arduino ที่พัฒนาด้วยโปรแกรมนี้คือ Arduino IDE (Arduino Integrated Development Environment: ID) แสดงดังรูปที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1. ตัวอย่างโปรแกรม Arduino IDE

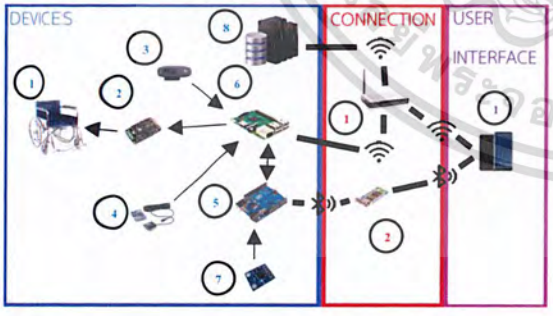
การใช้งาน Arduino IDE มีขั้นตอนดังนี้

1) ตั้งค่าบอร์ดให้ตรงกับบอร์ดที่ใช้งาน
2) ตั้งค่าพอร์ตเชื่อมต่อกับบอร์ด (กรณีต่อบอร์ดจริง)
3) ตั้งค่าชนิดการโปรแกรม (กรณีต่อบอร์ดจริง)
4) เขียนโปรแกรม
5) คลิกปุ่มแปลงไฟล์เป็นภาษาเครื่อง คลิกปุ่มเบิร์นไฟล์ลงบอร์ด (กรณีต่อบอร์ดจริง)

3. การวิเคราะห์และออกแบบระบบ

3.1 โครงสร้างสถาปัตยกรรมของระบบ

ระบบมีการใช้งานร่วมกันระหว่างอุปกรณ์สมาร์ทโฟนระบบปฏิบัติการแอนดรอยด์ และรถนั่งโดยสถาปัตยกรรมของระบบมีลักษณะ ดังรูปที่ 2



รูปที่ 2. โครงสร้างสถาปัตยกรรมของระบบ

จากรูปที่ 2 สามารถอธิบายรายละเอียดได้ดังนี้

3.1.1 Devices

1) รถนั่ง : ทำหน้าที่ในการเคลื่อนที่ตามคำสั่งที่ได้รับ
2) Control motor : เป็นตัวควบคุมมอเตอร์โดยค่าที่ได้รับจาก Raspberry pi
3) Camera Module : เชื่อมต่อกับ Raspberry Pi ทำหน้าที่ในการจับภาพด้านหน้าเพื่อนำภาพไปประมวลผลแบบเรียลไทม์ และได้ตัวแปลในการประมลผลภาพส่งไปยัง Raspberry pi
4) เซนเซอร์วัดระยะ : เป็นตัววัดระยะระหว่างวัตถุเพื่อส่งค่าไปยัง Raspberry pi เพื่อป้องกันอุบัติเหตุ
5) Arduino : ทำหน้าที่รับค่าจาก Accelerometer และส่งคำสั่งเพื่อให้โทรศัพท์โทรออก โดยส่งผ่าน Bluetooth Module
6) Raspberry pi : ทำหน้าที่เสมือน Central Processing Unit (CPU) เป็นตัวกลางระหว่างผู้ใช้กับรถนั่งใช้ในการควบคุมการทำงานของรถนั่ง เชื่อมต่อกับ Bluetooth Dongle เพื่อส่งสัญญาณควบคุมรถเข็น และจะเป็น Raspberry Pi ใช้กล้องเป็นตัวประมวลผลภาพ : ทำหน้าที่ตรวจจับภาพเพื่อส่งค่าไปยัง Arduino
7) Accelerometer : เซนเซอร์วัดความเร่งทำหน้าที่ในการตรวจสอบสถานการณ์ล้มของรถนั่งโดยส่งค่าที่ได้ให้ Arduino กรณีที่มีแกนใดผิดปกติจะส่งคำสั่งไปให้สมาร์ทโฟนทำการโทรออกฉุกเฉิน
8) Server : ทำหน้าที่เป็นตัวเก็บข้อมูลในการขับเคลื่อนอัตโนมัติไปยังสถานที่ต่าง ๆ

3.1.2 Connection

1) Router : ทั้งสมาร์ทโฟน และราสเบอร์รี่พายต้องเชื่อมต่อผ่านเร้าท์เตอร์เดียวกันเพื่อให้สมาร์ทโฟนและราสเบอร์รี่พายสามารถติดต่อสื่อสารกันภายในวง LAN ได้
2) Bluetooth Module : ทำหน้าที่เชื่อมต่อระหว่าง Arduino กับสมาร์ทโฟน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

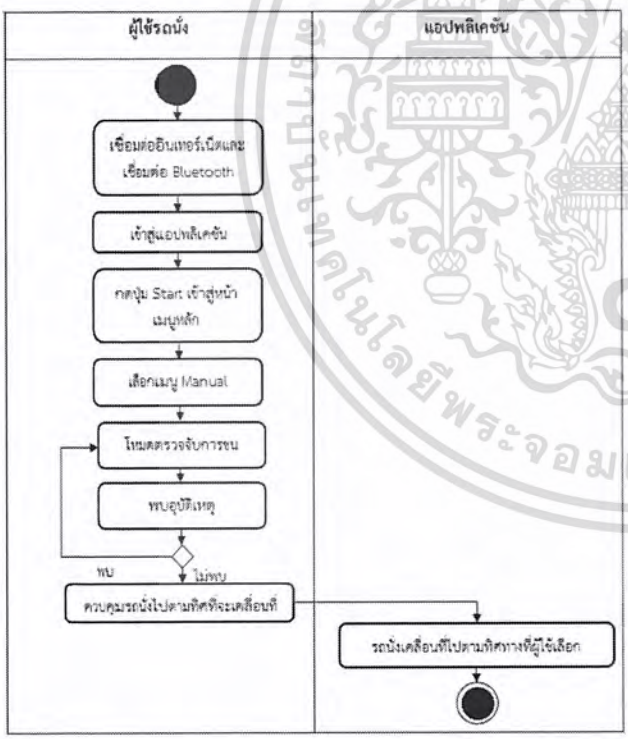
3.1.3 User interface

1) Smartphone : ทำหน้าที่ส่งคำสั่งควบคุมรถนั่งผ่านทางอินเทอร์เน็ตไปยัง Raspberry Pi

3.2 แผนภาพการทำงานของระบบ

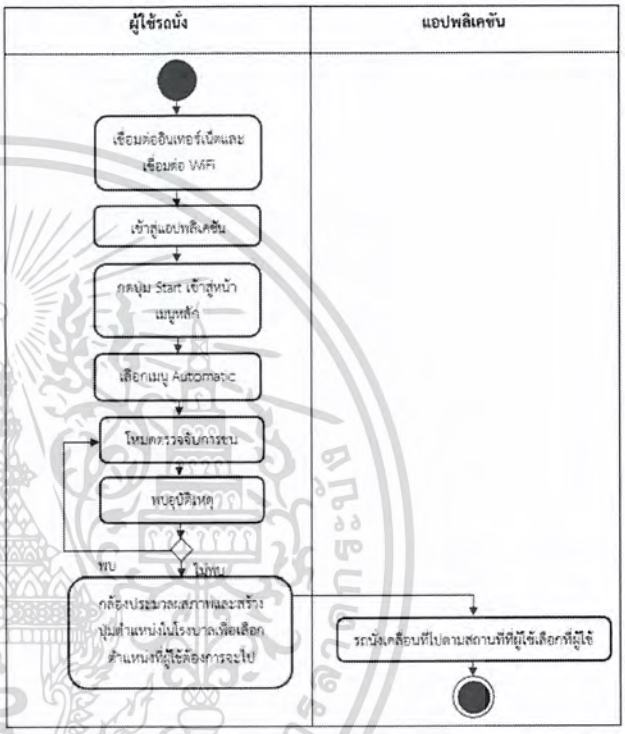
ระบบจะแบ่งการทำงานเป็น 3 ฟังก์ชันคือ การควบคุมการเคลื่อนที่รถนั่งด้วยตนเอง การควบคุมการเคลื่อนที่รถนั่งแบบอัตโนมัติ และการตรวจล้มและโทรออกฉุกเฉินดังนี้

1) การควบคุมการเคลื่อนที่รถนั่งด้วยตนเองผ่านแอปพลิเคชัน SmartCar โดยผู้ใช้รถนั่งต้องทำการเชื่อมต่อกับอินเทอร์เน็ตและเชื่อมต่อ Bluetoothก่อนจึงจะสามารถใช้งานแอปพลิเคชันได้ เมื่อเข้าสู่แอปพลิเคชันจะเปิดปุ่ม Start พอแตะเข้าไปแล้วจะพบกับเมนูสำหรับควบคุมการเคลื่อนที่ที่มีปุ่มชื่อว่า Manual และ สามารถควบคุมการเคลื่อนที่ของรถนั่งผ่านเมนูนี้ได้ทันที แสดงการทำงานดังรูปที่ 3



รูปที่ 3. แผนภาพ Activity Diagram การควบคุมการเคลื่อนที่รถนั่ง

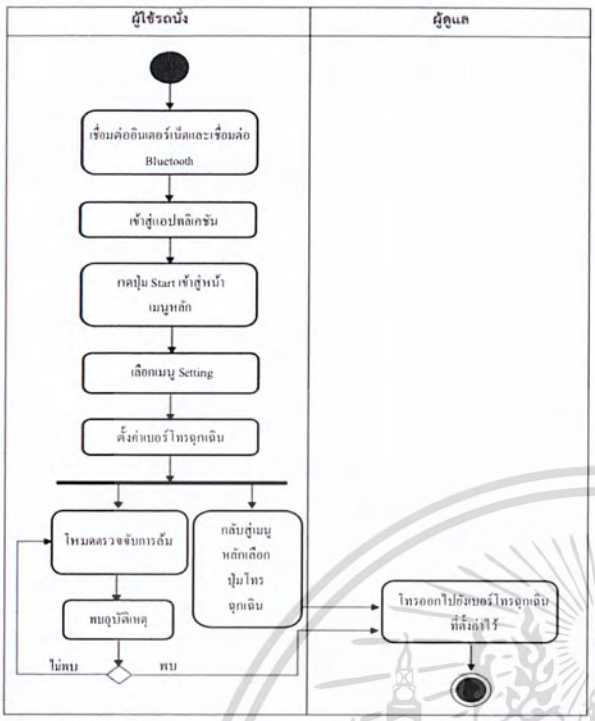
2) การควบคุมรถนั่งแบบอัตโนมัติผ่านแอปพลิเคชัน SmartCar โดยผู้ใช้รถนั่งต้องทำการเชื่อมต่อกับอินเทอร์เน็ตและเชื่อมต่อ Wifi ก่อนจึงจะสามารถใช้งานแอปพลิเคชันได้ เมื่อเข้าสู่แอปพลิเคชันแล้วจะพบกับปุ่ม Start เพื่อเข้าสู่การทำงานหน้าจอหลัก โดยในหน้าจอหลักจะพบปุ่ม Automatic สำหรับควบคุมการเคลื่อนที่ของรถนั่งแบบอัตโนมัติ โดยการใช้กล้องหน้าสำหรับการประมวลผลภาพเพื่อจะสร้างขึ้นบนหน้าจอและสามารถเลือกปุ่มกดไปยังตำแหน่งต่าง ๆ ในโรงพยาบาล ได้ควบคุมรถนั่งผ่านแอปพลิเคชัน SmartCar แสดงการทำงานได้ดังรูปที่ 4



รูปที่ 4. แผนภาพ Activity Diagram การควบคุมเคลื่อนที่แบบอัตโนมัติ

3) การตรวจจับการล้มรถนั่ง และการโทรออกฉุกเฉิน โดยผู้ใช้งานหลังจากเข้าสู่แอปพลิเคชันเจอปุ่ม Start เพื่อเข้าสู่การทำงานและเลือกเมนู Setting เพื่อทำการตั้งค่าเบอร์โทรออกฉุกเฉินเมื่อทำการตั้งค่าเสร็จระบบจะให้ทำการเชื่อมต่อบลูทูธเพื่อใช้ในการโทรออกฉุกเฉิน เมื่อรถนั่งล้มระบบจะทำการโทรออกอัตโนมัติไปยังเบอร์โทรฉุกเฉินที่ตั้งค่าไว้ หรือผู้ใช้สามารถกดปุ่มโทรออกฉุกเฉินด้วยตนเองได้ โดยปุ่มโทรออกฉุกเฉินจะแสดงในเมนูควบคุมการเคลื่อนที่รถนั่งแผนภาพการตรวจล้ม และโทรออกฉุกเฉินแสดงดังรูปที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

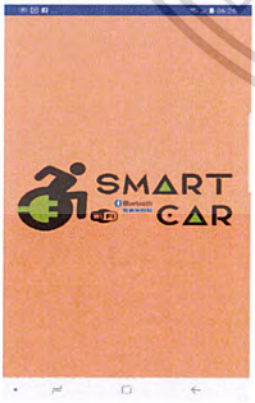


รูปที่ 5. แผนภาพ Activity Diagram การตรวจล้ม และโทรออกฉุกเฉิน

4. ผลการทดลอง

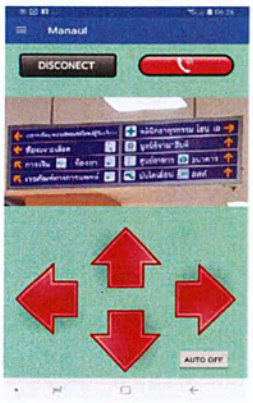
การออกแบบส่วนติดต่อประสานงานกับผู้ใช้แบ่งออกเป็น 2 ส่วน คือการติดต่อผ่านแอปพลิเคชัน และการติดต่อผ่านรถนั่ง มีรายละเอียดดังนี้

การติดต่อผ่านแอปพลิเคชันกับผู้ใช้มีส่วนที่จำเป็นสำหรับการใช้งาน โดยหน้าจอหลักแสดงดังรูปที่ 6



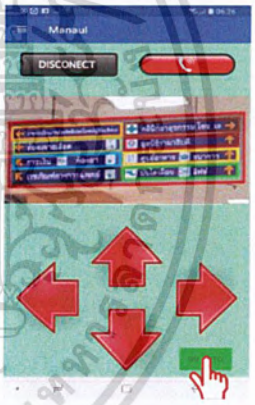
รูปที่ 6. หน้าจอเข้าสู่ระบบ

จากรูปที่ 6 เป็นการเปิดแอปพลิเคชันในหน้าแรกเพื่อเข้าสู่การใช้แอพพลิเคชัน



รูปที่ 7. หน้าจอโหมด Manual

จากรูปที่ 7 แสดงหน้าจอ Manual โดยหน้าจอนี้สามารถใช้ควบคุมได้ 4 ทิศทางโดยต้องกดปุ่ม DISCONECT เพื่อเชื่อต่อ Bluetooth จึงจะสามารถควบคุมได้ มีโทรออกฉุกเฉินและจะมีปุ่ม AUTO มุมขวาล่าง แสดงดังรูปที่ 8



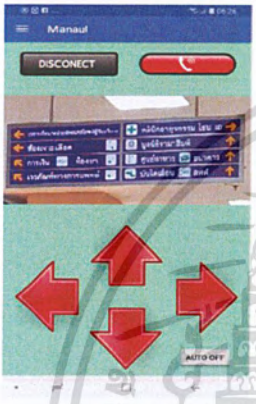
(ก)

(ข)

รูปที่ 8. (ก) หน้าจอพร้อมใช้งาน (ข) หน้าจอเลือกสถานที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 8 แสดงหน้าจอการทำงานโหมด Automatic เมื่อทำการเคลื่อนที่โดยในโหมด Manual ไปยังที่กล้องจับตำแหน่งป้ายได้ จะขึ้นกรอบสี่จับเป็น Pixel และ สร้างปุ่ม Button แต่ละตำแหน่งเพื่อเลือกไปยังตำแหน่งนั้นในโหมด Auto 8(ก) แสดงหน้าจอพร้อมใช้งานเมื่อได้ทำการเคลื่อนไปยังตำแหน่งที่จับภาพมีกรอบขึ้นแล้ว ปุ่มมุมด้านขวาล่างจะเป็นสีเขียวเพื่อให้รู้พร้อมใช้งาน และรูปที่ 8(ข) แสดงหน้าจอแอพลิเคชันหลังจากกดปุ่มสีเขียวเพื่อเข้าสู่หน้าจอเลือกที่ต้องการต่อไป



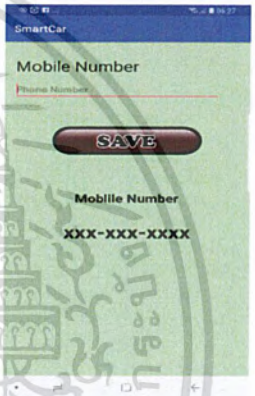
(ก)

(ข)

รูปที่ 9. หน้าจอการเข้าสู่เมนู Setting (ก) ตำแหน่งการกด (ข) หน้าจอแถบเครื่องมือ

จากรูปที่ 9 9(ก) เป็นการเลือกปุ่มแฮมเบอร์เกอร์ 9(ข) เป็นแถบเครื่องมือ

(ก)



(ข)

รูปที่ 10. (ก) ตำแหน่งการกด (ข) หน้าจอตั้งค่าเบอร์โทรผู้ดูแล

จากรูปที่ 10 10(ก) เลือกปุ่มเครื่องมือ Setting 10(ข) เป็นการ Setting เบอร์ผู้ดูแล

5. บทสรุป

แอนดรอยด์แอปพลิเคชันที่ใช้ควบคุมรถนั่งอัตโนมัติต้นแบบถูกพัฒนาขึ้นโดยใช้ภาษา C ติดตั้งใช้งานบนระบบปฏิบัติการแอนดรอยด์ และใช้บลูทูธในการเชื่อมต่อสัญญาณกันระหว่างสมาร์ทโฟนกับรถนั่งอัตโนมัติต้นแบบ ซึ่งสามารถนำแนวคิดรถนั่งอัตโนมัติต้นแบบมาใช้กับรถวีลแชร์ทั่วไปภายในบ้านเพื่อใช้อำนวยความสะดวกกับผู้พิการทางขา หรือ ผู้สูงอายุต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้