

การตรวจจับสิ่งกีดขวางสำหรับเก้าอี้รถเข็นไฟฟ้าด้วยคอมพิวเตอร์วิทัศน์



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2564

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

Detection of Obstacles for Electric Wheelchair with Computer Vision



A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science

Department of Computer Engineering

FACULTY OF ENGINEERING

Chulalongkorn University

Academic Year 2021

Copyright of Chulalongkorn University

| | |
|---------------------------------|---|
| หัวข้อวิทยานิพนธ์ | การตรวจจับสิ่งกีดขวางสำหรับเก้าอี้รถเข็นไฟฟ้าด้วย คอมพิวเตอร์วิทัศน์ |
| โดย | น.ส.เพ็ญพิชชา พัฒนจิตรศิลป์ |
| สาขาวิชา | วิทยาศาสตร์คอมพิวเตอร์ |
| อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก | ศาสตราจารย์ ดร.ประภาส จงสกลิตย์วัฒนา |

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้หัวข้อวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่ง
ของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.สุพจน์ เตชวรสินสกุล)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.สุกรี สินธุภิญโญ)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ศาสตราจารย์ ดร.ประภาส จงสกลิตย์วัฒนา)

..... กรรมการภายนอกมหาวิทยาลัย
(ผู้ช่วยศาสตราจารย์ ดร.ธนัสินี เพ็ญตระกูล)

CHULALONGKORN UNIVERSITY

เพื่อพิชชา พัฒนจิตรศิลป์ : การตรวจจับสิ่งกีดขวางสำหรับเก้าอี้รถเข็นไฟฟ้าด้วย
คอมพิวเตอร์วิทัศน์. (Detection of Obstacles for Electric Wheelchair with
Computer Vision) อ.ที่ปรึกษาหลัก : ศ. ดร.ประภาส จงสถิตย์วัฒนา

งานวิจัยนี้นำเสนอระบบตรวจจับสิ่งกีดขวางขณะถอยหลังสำหรับเก้าอี้รถเข็นไฟฟ้าด้วย
เทคนิคคอมพิวเตอร์วิทัศน์ เพื่ออำนวยความสะดวกให้แก่ผู้พิการ และ ช่วยลดอุบัติเหตุที่อาจจะ
เกิดขึ้น ต้องมีการกำหนดระยะที่ต้องการให้ระบบทำการแจ้งเตือนเมื่อใกล้จะชนสิ่งกีดขวาง ซึ่งใช้
กล้องจากสมาร์ทโฟนไปติดตั้งไว้ด้านหลังเก้าอี้รถเข็น จากนั้นใช้อัลกอริทึม YOLOv3 ในการตรวจจับ
สิ่งกีดขวางประเภทวัตถุ (object detection) แต่เนื่องจาก YOLOv3 ไม่สามารถตรวจจับสิ่งกีด
ขวางได้ทุกประเภท ในงานวิจัยนี้จึงได้มีการพัฒนาอัลกอริทึมที่ใช้ในการตรวจจับสิ่งกีดขวางที่เป็น
เสา กำแพง หรือ ประตู เพิ่มเติมขึ้นมาด้วยวิธีการตรวจจับขอบภาพ (edge detection) เพื่อเพิ่ม
ประสิทธิภาพในการตรวจจับสิ่งกีดขวางให้กับระบบ เมื่อมีการใช้งานร่วมกันระหว่าง 2 อัลกอริทึม
ระบบจึงมีการเลือกใช้อัลกอริทึมในการตรวจจับสิ่งกีดขวางระหว่างวัตถุกับเส้นขอบ ซึ่งผลการ
ทดลองพบว่า ระบบสามารถเลือกใช้อัลกอริทึมในการตรวจจับสิ่งกีดขวางได้ถูกต้องถึง 80% และ
ทำการแจ้งเตือนก่อนชนได้แม่นยำสูงถึง 90% อีกทั้งระบบนี้ยังสามารถคำนวณเวลาก่อนชนได้ด้วย

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์
ปีการศึกษา 2564

ลายมือชื่อนิสิต
ลายมือชื่อ อ.ที่ปรึกษาหลัก

6370208221 : MAJOR COMPUTER SCIENCE

KEYWORD: Computer Vision, Obstacle Detection, Edge Detection, Electric
Wheelchair, Time to Collision

Phenphitcha Patthanajitsilp : Detection of Obstacles for Electric
Wheelchair with Computer Vision. Advisor: Prof. PRABHAS
CHONGSTITVATANA, Ph.D.

This research aims to present the detection system of an obstacle for electric wheelchair using computer vision in order to facilitate for disabled persons and reduce the possibilities of accidents. In this system, the distance threshold is set to alert when a wheelchair is approaching an obstacle. The alert system consists of the smartphone camera attached to the back of a wheelchair. The YOLOv3 model was used for object detection. The researcher has developed an algorithm to detect obstacles such as pillars, walls, or doors with edge detection method to enhance the detection efficiency of the system. The usage of two algorithms enables the system to choose the obstacle detection between objects and edge detection. The research found that the system can choose the algorithm to detect obstacles with an accuracy of up to 80%. Moreover, the experiment revealed that the system can alert warnings before collisions with an accuracy of up to 90% and, this system can also calculate the accurate time prior to the collision.

Field of Study: Computer Science

Student's Signature

Academic Year: 2021

Advisor's Signature

กิตติกรรมประกาศ

ขอขอบคุณทุนอุดหนุนการศึกษาระดับบัณฑิตศึกษาจาก บัณฑิตวิทยาลัยจุฬาลงกรณ์มหาวิทยาลัย เพื่อเฉลิมฉลองในโอกาสที่สมเด็จพระเทพรัตนราชสุดาฯ สยามบรมราชกุมารี ทรงเจริญพระชนมายุ 60 พรรษา

ขอขอบคุณ ศ.ดร. ประภาส จงสฤษดิ์วัฒนา สำหรับคำแนะนำในการทำงานวิจัยตั้งแต่การเลือกหัวข้อวิจัย ขั้นตอนการทำวิจัย รวมไปถึงการแนะนำแหล่งอ้างอิงข้อมูล ตลอดจนแก้ไขข้อบกพร่องต่าง ๆ เพื่อให้งานวิจัยนี้เสร็จสมบูรณ์

สุดท้ายนี้ขอขอบคุณครอบครัวที่คอยสนับสนุนและให้กำลังใจเสมอมา

เพ็ญพิชชา พัฒนจิตรศิลป์



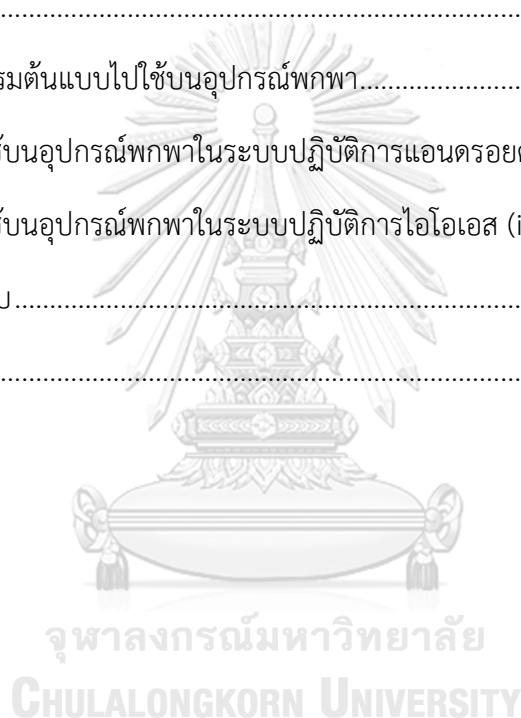
สารบัญ

| | หน้า |
|--|------|
| บทคัดย่อภาษาไทย..... | ค |
| บทคัดย่อภาษาอังกฤษ..... | ง |
| กิตติกรรมประกาศ..... | จ |
| สารบัญ..... | ฉ |
| สารบัญตาราง..... | ญ |
| สารบัญรูปภาพ..... | ฎ |
| บทที่ 1 บทนำ | 1 |
| 1.1 ความเป็นมาและความสำคัญของปัญหา | 1 |
| 1.2 วัตถุประสงค์ของการวิจัย..... | 1 |
| 1.3 ขอบเขตการวิจัย | 2 |
| 1.4 ประโยชน์ที่คาดว่าจะได้รับ..... | 2 |
| 1.5 ขั้นตอนการดำเนินงาน..... | 2 |
| 1.6 ผลงานวิจัยที่ตีพิมพ์..... | 3 |
| บทที่ 2 ทฤษฎีที่เกี่ยวข้อง | 4 |
| 2.1 การประมวลผลภาพ (Image Processing)..... | 4 |
| 2.1.1 วิธีที่ใช้ในการประมวลผลภาพ | 4 |
| 2.2 คอมพิวเตอร์วิทัศน์ (Computer Vision)..... | 5 |
| 2.3 ระบบปสี | 6 |
| 2.4 การกรองข้อมูลภาพ (Image Filtering)..... | 7 |
| 2.4.1 ประเภทของตัวกรอง | 8 |
| 2.4.2 การกรองโดยวิธีสังวัตนาการ (Convolution)..... | 10 |

| | | |
|---------|---|----|
| 2.5 | สังวัตนาการ (Convolution)..... | 10 |
| 2.5.1 | สังวัตนาการใน 1 มิติ..... | 12 |
| 2.5.2 | สังวัตนาการใน 2 มิติ..... | 12 |
| 2.6 | ตัวกรอง (Filter)..... | 13 |
| 2.7 | การแปลงภาพ (Image Transformation)..... | 13 |
| 2.7.1 | ประเภทการแปลง | 14 |
| 2.8 | Scale Invariant Features Transform (SIFT)..... | 14 |
| 2.8.1 | การหาปริภูมิค่าในมิติขนาดและระยะทาง (Scale-space extrema detection) | 15 |
| 2.8.2 | การกำหนดตำแหน่งจุดสนใจ (Keypoint localization)..... | 15 |
| 2.8.3 | การกำหนดทิศทางของจุดสนใจ (Orientation assignment)..... | 17 |
| 2.8.4 | การสร้างคำอธิบายลักษณะเด่นของภาพ (Keypoint descriptor)..... | 18 |
| 2.9 | การปรับเทียบกล้อง (Camera Calibration)..... | 19 |
| 2.9.1 | แบบจำลองกล้องรูเข็ม (Pinhole Camera Model)..... | 19 |
| 2.9.1.1 | พารามิเตอร์การปรับเทียบกล้องสำหรับกล้องรูเข็ม | 20 |
| 2.9.2 | การบิดเบือนในแนวรัศมี (Radial Distortion)..... | 22 |
| 2.9.2.1 | การบิดเบือนแบบบาร์เรล (Barrel Distortion)..... | 22 |
| 2.9.2.2 | การบิดเบือนแบบหมอนอิง (Pincushion Distortion)..... | 23 |
| 2.9.2.3 | การบิดเบือนแบบหนวด (Mustache Distortion) | 23 |
| 2.9.3 | การบิดเบือนในแนวสัมผัส (Tangential Distortion)..... | 24 |
| 2.10 | ออฟติคัลโฟลว์ (Optical Flow)..... | 24 |
| 2.11 | การตรวจจับวัตถุ (Object Detection) | 25 |
| 2.11.1 | Single-Shot Detector (SSD)..... | 25 |
| 2.11.2 | You Only Look Once (YOLO)..... | 26 |
| 2.12 | การตรวจจับขอบภาพ (Edge Detection)..... | 28 |

| | | |
|---------|---|----|
| 2.12.1 | ลาปลาเซียนเมทอด (Laplacian method)..... | 28 |
| 2.12.2 | เกรเดียนตเมทอด (Gradient method)..... | 28 |
| 2.12.3 | อัลกอริทึมตรวจจับขอบภาพแคนนี่ (Canny Edge Detection Algorithm)..... | 28 |
| บทที่ 3 | งานวิจัยที่เกี่ยวข้อง | 30 |
| บทที่ 4 | แนวคิดและวิธีการวิจัย | 34 |
| 4.1 | อุปกรณ์ที่ใช้ในงานวิจัย | 34 |
| 4.2 | กระบวนการทำงานของระบบ | 36 |
| 4.3 | การตรวจจับสิ่งกีดขวาง | 37 |
| 4.3.1 | การตรวจจับสิ่งกีดขวางประเภทวัตถุ | 37 |
| 4.3.2 | การตรวจจับสิ่งกีดขวางประเภทที่ไม่ใช่วัตถุ | 38 |
| 4.4 | การกำหนดระยะห่าง | 39 |
| 4.5 | การแจ้งเตือนการชน | 40 |
| 4.5.1 | ข้อความแจ้งเตือน Warning | 41 |
| 4.5.2 | ข้อความแจ้งเตือน Collision | 42 |
| 4.6 | การคำนวณเวลาก่อนชน | 43 |
| บทที่ 5 | การทดลองและผลการทดลอง | 45 |
| 5.1 | การนำเข้าข้อมูล | 45 |
| 5.2 | การทดลอง | 45 |
| 5.3 | วิธีการทดลองและการวัดผล | 47 |
| 5.3.1 | การเลือกใช้อัลกอริทึมสำหรับตรวจจับสิ่งกีดขวาง | 47 |
| 5.3.2 | การแจ้งเตือน | 49 |
| 5.3.3 | การคำนวณเวลาก่อนชน | 50 |
| 5.4 | ผลการทดลอง..... | 51 |
| 5.4.1 | การเลือกใช้อัลกอริทึมสำหรับตรวจจับสิ่งกีดขวาง | 51 |

| | |
|--|----|
| 5.4.2 การแจ้งเตือน | 52 |
| 5.4.3 การคำนวณเวลาก่อนชน | 53 |
| บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ | 55 |
| 6.1 สรุปผลการวิจัย..... | 55 |
| 6.2 ข้อเสนอแนะ | 56 |
| บรรณานุกรม..... | 58 |
| ภาคผนวก ก..... | 60 |
| 1. การนำโปรแกรมต้นแบบไปใช้บนอุปกรณ์พกพา..... | 60 |
| 1.1 นำไปใช้บนอุปกรณ์พกพาในระบบปฏิบัติการแอนดรอยด์ (Android OS) | 60 |
| 1.2 นำไปใช้บนอุปกรณ์พกพาในระบบปฏิบัติการไอโอเอส (iOS OS)..... | 60 |
| 2. การตั้งค่าระบบ..... | 61 |
| ประวัติผู้เขียน..... | 63 |



สารบัญตาราง

| | หน้า |
|---|------|
| ตารางที่ 1 ตัวอย่างตัวกรองที่สามารถทำได้โดยการสังวัตนาการกับรูปภาพ..... | 13 |
| ตารางที่ 2 เปรียบเทียบประสิทธิภาพของอัลกอริทึมตรวจจับวัตถุ..... | 32 |
| ตารางที่ 3 ผลการทดลองในการเลือกใช้อัลกอริทึมตรวจจับสิ่งกีดขวางของระบบ..... | 51 |
| ตารางที่ 4 ผลการทดลองการแจ้งเตือนเมื่อระบบตรวจพบสิ่งกีดขวาง..... | 52 |
| ตารางที่ 5 ผลการทดลองการคำนวณเวลาก่อนชน..... | 53 |
| ตารางที่ 6 ความแม่นยำในการตรวจจับสิ่งกีดขวาง การแสดงข้อความแจ้งเตือน และการคำนวณเวลาก่อนชน..... | 56 |



สารบัญรูปภาพ

| | หน้า |
|--|------|
| รูปที่ 1 วิดีทัศน์สตรีม (video stream)..... | 4 |
| รูปที่ 2 ระบบสี RGB..... | 6 |
| รูปที่ 3 ระบบสี CMYK | 6 |
| รูปที่ 4 ระบบสี HSB..... | 6 |
| รูปที่ 5 ระบบสี LAB..... | 7 |
| รูปที่ 6 ระบบสี Grayscale | 7 |
| รูปที่ 7 ระบบสี Bitmap..... | 7 |
| รูปที่ 8 ตัวกรองความถี่ต่ำผ่าน (Low-Pass Filter)..... | 8 |
| รูปที่ 9 สัจวัตนาการใน 2 มิติด้วยตัวกรองความถี่ต่ำผ่าน (Low-Pass Filter with 2D Convolution) | 8 |
| รูปที่ 10 ตัวกรองความถี่สูงผ่าน (High-Pass Filter)..... | 9 |
| รูปที่ 11 สัจวัตนาการใน 2 มิติด้วยตัวกรองความถี่สูงผ่าน (High-Pass Filter with 2D Convolution)..... | 9 |
| รูปที่ 12 ตัวกรองแถบความถี่ผ่าน (Band-Pass Filter)..... | 9 |
| รูปที่ 13 ตัวกรองหยุดแถบความถี่ (Band-Stop Filter)..... | 10 |
| รูปที่ 14 กราฟเส้นแสดงการสัจวัตนาการโดยการคูณตัวกรองกับแต่ละส่วน..... | 10 |
| รูปที่ 15 กราฟเส้นแสดงการสัจวัตนาการโดยการคูณตัวกรองกับแต่ละส่วนแบบกลับด้าน | 11 |
| รูปที่ 16 สัจวัตนาการใน 1 มิติ..... | 12 |
| รูปที่ 17 สัจวัตนาการใน 2 มิติ..... | 12 |
| รูปที่ 18 ขั้นตอนการทำงานของวิธีการเอสไอเอฟที | 15 |
| รูปที่ 19 การหา Difference of Gaussian ในแต่ละขั้น (Octave)..... | 16 |
| รูปที่ 20 จุด X เป็นจุดอ้างอิงเปรียบเทียบกับจุดรอบข้าง 26 จุด | 16 |

| | |
|--|----|
| รูปที่ 21 จุดสีเขียวน คือ subpixel เป็นจุดสูงสุดหรือต่ำสุด | 16 |
| รูปที่ 22 ขนาดและทิศทางของจุดสนใจ | 18 |
| รูปที่ 23 มุมมองรูเข็มและมุมมองฟิชอาย | 19 |
| รูปที่ 24 แบบจำลองกล้องรูเข็ม (Pinhole Camera Model) | 19 |
| รูปที่ 25 เมทริกซ์และพารามิเตอร์ที่เกี่ยวข้องกับแบบจำลองกล้องรูเข็ม | 20 |
| รูปที่ 26 การแปลงพารามิเตอร์ภายนอกไปเป็นพารามิเตอร์ภายในของแบบจำลองกล้องรูเข็ม | 20 |
| รูปที่ 27 การแปลงจากพิกัดโลก 3 มิติไปเป็นพิกัดภาพ 2 มิติของกล้องรูเข็ม | 21 |
| รูปที่ 28 พารามิเตอร์ภายนอกของกล้องรูเข็ม | 21 |
| รูปที่ 29 ความเอียงของพิกเซล | 21 |
| รูปที่ 30 การบิดเบือนในแนวรัศมีในรูปแบบต่าง ๆ | 22 |
| รูปที่ 31 การบิดเบือนแบบบาร์เรล | 22 |
| รูปที่ 32 การบิดเบือนแบบหมอนอิง | 23 |
| รูปที่ 33 การบิดเบือนแบบหนวด | 23 |
| รูปที่ 34 การบิดเบือนในแนวสัมผัส | 24 |
| รูปที่ 35 การไหลของแสงเทียบกับภาพต้นฉบับ | 24 |
| รูปที่ 36 การตรวจจับวัตถุเดี่ยวและหลายวัตถุในรูปภาพ | 25 |
| รูปที่ 37 โครงสร้างของ YOLO ที่เป็นหลายชั้นแบบ Neural Network โดย grid จะถี่ขึ้นเรื่อย ๆ จน ได้ผลลัพธ์ที่ดี | 26 |
| รูปที่ 38 การตรวจจับวัตถุวัตถุที่ทับซ้อนกันของ YOLO | 27 |
| รูปที่ 39 การเปรียบเทียบประสิทธิภาพของ YOLOv3 ซึ่งทำงานเร็วกว่าวิธีการตรวจจับแบบอื่น | 27 |
| รูปที่ 40 การตรวจจับขอบภาพด้วยอัลกอริทึม Canny | 29 |
| รูปที่ 41 ตัวอย่างอุปกรณ์ฮาร์ดแวร์ | 30 |
| รูปที่ 42 ช่วงของระยะในการส่งเสียงเตือน | 31 |
| รูปที่ 43 ติดตั้งกล้องและเลเซอร์ด้านหน้าเก้าอี้รถเข็น | 32 |

| | |
|--|----|
| รูปที่ 44 กล้อง iPhone XS Max | 35 |
| รูปที่ 45 ดึงกล้องสมาร์ทโฟนไว้ด้านหลังเก้าอี้รถเข็นไฟฟ้า | 35 |
| รูปที่ 46 ภาพรวมการทำงานของระบบ | 36 |
| รูปที่ 47 ตรวจสอบสิ่งกีดขวางประเภทวัตถุโดยใช้อัลกอริทึม YOLOv3 | 37 |
| รูปที่ 48 แปลงรูปภาพต้นฉบับให้เป็นภาพขอบ 3 รอบ | 38 |
| รูปที่ 49 ตรวจสอบขอบของสิ่งกีดขวางโดยใช้อัลกอริทึม Canny | 39 |
| รูปที่ 50 กำหนดระยะที่ต้องการให้ระบบแจ้งเตือน | 39 |
| รูปที่ 51 ระบบแสดงข้อความแจ้งเตือนก่อนชน ด้วยคำว่า Warning | 41 |
| รูปที่ 52 ระบบแสดงข้อความแจ้งเตือนก่อนชน ด้วยคำว่า Collision | 42 |
| รูปที่ 53 แสดงเวลาคงเหลือก่อนชน พร้อมกับ แสดงข้อความแจ้งเตือน | 44 |
| รูปที่ 54 ตัวอย่างรูปภาพบางส่วนจากวีดิทัศน์ที่มีสิ่งกีดขวางประเภทวัตถุ | 46 |
| รูปที่ 55 ตัวอย่างรูปภาพบางส่วนจากวีดิทัศน์ที่ไม่มีสิ่งกีดขวางประเภทที่ไม่ใช่วัตถุ | 46 |
| รูปที่ 56 ตัวอย่างรูปภาพบางส่วนจากวีดิทัศน์ที่ไม่มีสิ่งกีดขวาง | 47 |
| รูปที่ 57 กรอบสี่เหลี่ยมสีเขียวล้อมรอบสิ่งกีดขวางประเภทวัตถุ | 48 |
| รูปที่ 58 เส้นตรงสีเขียวที่ขอบของสิ่งกีดขวางประเภทที่ไม่ใช่วัตถุ | 49 |
| รูปที่ 59 เวลาที่ระบบคาดการณ์ไว้ล่วงหน้าว่าจะชน เทียบกับ เวลาคงเหลือก่อนที่วีดิทัศน์จะเล่นจบ | 50 |

บทที่ 1 บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

สังคมปัจจุบันผู้พิการและผู้สูงอายุมีอัตราการเพิ่มขึ้นอย่างต่อเนื่อง ซึ่งบุคคลเหล่านี้ล้วนแต่มีข้อจำกัดในการดำเนินชีวิตด้วยกันทั้งสิ้น จึงได้มีอุปกรณ์ช่วยเหลือที่แตกต่างกันออกไปในแต่ละประเภท

ในงานวิจัยนี้จะมุ่งเน้นไปที่กลุ่มผู้ใช้งานเก้าอี้รถเข็นไฟฟ้า เนื่องจากผู้ใช้ส่วนใหญ่จะมีข้อจำกัดทางด้านร่างกายและการเคลื่อนไหว ด้วยเหตุนี้เก้าอี้รถเข็นจึงเป็นอุปกรณ์ที่ช่วยให้ผู้พิการหรือผู้สูงอายุสามารถพึ่งพาตัวเองได้อย่างสะดวกมากยิ่งขึ้นและยังช่วยลดข้อจำกัดในการดำเนินชีวิตให้แก่พวกเขาได้อีกด้วย

โดยทั่วไปเก้าอี้รถเข็นสามารถแบ่งออกเป็น 2 ประเภทหลัก ได้แก่

1. เก้าอี้รถเข็นแบบใช้มือเข็น เหมาะสำหรับผู้ใช้งานที่มีความแข็งแรงพอสมควร เพราะต้องใช้แรงตัวเองในการขับเคลื่อน
2. เก้าอี้รถเข็นไฟฟ้า เหมาะสำหรับผู้ใช้งานที่ค่อนข้างมีข้อจำกัดทางด้านร่างกายและผลกะกำลัง เพราะผู้ใช้งานไม่จำเป็นจะต้องออกแรงเข็นเอง แต่จะขับเคลื่อนด้วยเครื่องยนต์ซึ่งต้องใช้แบตเตอรี่

แม้ว่าเก้าอี้รถเข็นจะช่วยให้ผู้ที่มีความบกพร่องทางด้านร่างกายสามารถเคลื่อนย้ายไปยังที่ต่าง ๆ ได้สะดวกขึ้น แต่กลับพบว่า บ่อยครั้งอุปสรรคของการใช้เก้าอี้รถเข็นมักจะเป็นเรื่องของการที่มองไม่เห็นสิ่งกีดขวาง โดยเฉพาะอย่างยิ่งเมื่อใช้งานโดยการเคลื่อนที่แบบถอยหลัง จะยิ่งทำให้มีความยากลำบากในการมองเห็นด้านหลัง ซึ่งอาจจะส่งผลให้เกิดอุบัติเหตุได้

จากปัญหาดังกล่าว งานวิจัยนี้จึงต้องการนำเสนอระบบตรวจจับสิ่งกีดขวางสำหรับเก้าอี้รถเข็นไฟฟ้าด้วยคอมพิวเตอร์วิทัศน์ เพื่ออำนวยความสะดวก ลดข้อจำกัดในการใช้งาน และ ลดอุบัติเหตุที่อาจจะเกิดขึ้นได้ให้แก่ผู้ใช้เก้าอี้รถเข็นไฟฟ้า

1.2 วัตถุประสงค์ของการวิจัย

เพื่ออำนวยความสะดวกให้กับผู้ใช้เก้าอี้รถเข็นไฟฟ้า โดยระบบทำการแจ้งเตือนว่ามีสิ่งกีดขวางอยู่ด้านหลัง

เนื่องจากโดยทั่วไปแล้วผู้ที่มีความบกพร่องทางด้านร่างกายจะไม่สะดวกในการขยับร่างกาย จึงทำให้ไม่สามารถหันไปมองเห็นสิ่งกีดขวางที่อยู่ด้านหลัง ดังนั้นในงานวิจัยนี้จึงได้นำเอาความรู้ในเรื่องของการทำให้คอมพิวเตอร์สามารถมองเห็นหรือตรวจจับวัตถุได้มาประยุกต์ใช้ เพื่อช่วยให้ผู้พิการ

สามารถใช้แก้อักรถเซ็นไฟฟ้าได้อย่างสะดวกและปลอดภัยมากยิ่งขึ้น อีกทั้งยังช่วยลดอุบัติเหตุที่อาจจะเกิดขึ้นได้

1.3 ขอบเขตการวิจัย

- งานวิจัยนี้มุ่งเน้นไปที่การใช้งานแก้อักรถเซ็นไฟฟ้าในสภาวะแวดล้อมบนพื้นราบเท่านั้น
- งานวิจัยนี้ทดลองกับวัตถุที่เป็นสิ่งกีดขวางโดยไม่มีเคลื่อนที่
- งานวิจัยนี้ใช้กล้องสมาร์ทโฟน (smartphone) เป็นอุปกรณ์ในการตรวจจับสิ่งกีดขวาง
- งานวิจัยนี้นำเข้าข้อมูลวีดิทัศน์ ซึ่งถ่ายด้วยกล้องสมาร์ทโฟน โดยจำลองการใช้งานแก้อักรถเซ็นไฟฟ้าแบบเสมือนจริง
- ทดสอบระบบที่เป็นการรวมกันของ 2 แบบจำลองกับวีดิทัศน์
- งานวิจัยนี้ได้ผลลัพธ์เป็นข้อความแจ้งเตือน 2 รูปแบบ ได้แก่ warning และ collision พร้อมกับคำนวณเวลาก่อนชนสิ่งกีดขวางในหน่วยวินาที
- เป้าหมายของงานวิจัยนี้ คาดหวังให้ระบบสามารถแจ้งเตือนก่อนชนสิ่งกีดขวางได้ โดยมีความแม่นยำอยู่ที่ 80%

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- สามารถทราบได้ว่าจะไปชนกับสิ่งกีดขวางเมื่อใด
- สามารถแจ้งเตือนล่วงหน้าได้ว่ากำลังจะไปชนกับสิ่งกีดขวาง
- เพื่อลดโอกาสเกิดอุบัติเหตุจากการที่มองไม่เห็นสิ่งกีดขวาง
- เพื่ออำนวยความสะดวกให้กับผู้พิการ หรือ ผู้สูงอายุที่ใช้แก้อักรถเซ็นไฟฟ้า ให้สามารถดำเนินชีวิตได้อย่างสะดวกมากยิ่งขึ้น
- สามารถนำไปพัฒนาต่อยอด ประยุกต์ใช้กับอุปกรณ์อำนวยความสะดวกประเภทอื่น ๆ เพื่อให้เกิดประโยชน์แก่ผู้ที่มีความบกพร่องในด้านนั้น ๆ ต่อไป

1.5 ขั้นตอนการดำเนินงาน

- ศึกษาพฤติกรรมการใช้งานแก้อักรถเซ็นไฟฟ้า และ วิเคราะห์สาเหตุของปัญหาที่เกิดขึ้น
- ศึกษาทฤษฎีที่เกี่ยวข้องกับการทำระบบตรวจจับสิ่งกีดขวางและการประมวลผลภาพ
- ศึกษากล้องของสมาร์ทโฟน โดยการถ่ายเป็นวีดิทัศน์มาทำการทดลองประมวลผลภาพเบื้องต้น เพื่อวัดคุณภาพกล้อง
- เปรียบเทียบประสิทธิภาพและความแม่นยำของแบบจำลองในการตรวจจับสิ่งกีดขวางและการตรวจจับขอบภาพของวัตถุ

- เลือกแบบจำลองในการตรวจจับสิ่งกีดขวางและการตรวจจับขอบภาพของวัตถุที่เหมาะสมกับการนำมาใช้ในระบบ
- พัฒนาระบบในส่วนของการตรวจจับสิ่งกีดขวางประเภทวัตถุ
- พัฒนาระบบในส่วนของการตรวจจับขอบภาพของสิ่งกีดขวางที่ไม่ใช่วัตถุ
- พัฒนาระบบในส่วนของการคำนวณเวลาก่อนชนสิ่งกีดขวาง
- ทดสอบระบบ และ ตรวจสอบความถูกต้องแม่นยำ
- ปรับปรุงประสิทธิภาพของระบบ
- สรุปผลการวิจัย
- เรียบเรียงบทความทางวิชาการ
- จัดทำวิทยานิพนธ์

1.6 ผลงานวิจัยที่ตีพิมพ์

“Obstacles Detection for Electric Wheelchair with Computer Vision” โดย เพ็ญพิชชา พัฒนจิตรศิลป์ และ ศ.ดร. ประภาส จงสฤษดิ์วัฒนา ในงานประชุมวิชาการนานาชาติ Knowledge and Smart Technology ครั้งที่ 14 (The 2022-14th International Conference on Knowledge and Smart Technology : KST)

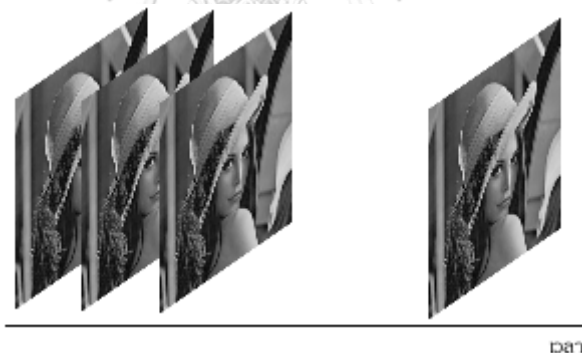
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

งานวิจัยนี้ได้นำทฤษฎีที่เกี่ยวข้องกับการประมวลผลภาพดิจิทัล (Digital Image Processing) และ วิธีการค้นหาวัตถุในรูปภาพ มาประยุกต์ใช้ในขั้นตอนต่าง ๆ เพื่อพัฒนาระบบการในส่วนของการตรวจจับสิ่งกีดขวางสำหรับเก้าอี้รถเข็นไฟฟ้า

2.1 การประมวลผลภาพ (Image Processing)

การประมวลผลภาพเป็นการประยุกต์ใช้งานการประมวลผลสัญญาณ บนสัญญาณ 2 มิติ เช่น ภาพนิ่ง (ภาพถ่าย) หรือ ภาพวิดีโอ และ รวมถึงสัญญาณ 2 มิติอื่น ๆ ที่ไม่ใช่ภาพด้วย

การประมวลผลภาพดิจิทัลเป็นสาขาที่กล่าวถึงเทคนิค ขั้นตอน และ วิธีการต่าง ๆ รวมถึง อัลกอริทึมที่ใช้การประมวลผลภาพที่อยู่ในรูปแบบดิจิทัล ภาพในที่นี้หมายถึงสัญญาณดิจิทัลใน 2 มิติอื่น ๆ โดยทั่วไปคำนี้มักจะครอบคลุมถึงสัญญาณวิดีโอ หรือ ภาพเคลื่อนไหว ซึ่งเป็นชุดของภาพนิ่ง เรียกว่า เฟรม (frame) หลาย ๆ ภาพต่อกันไปตามเวลา ซึ่งก็คือสัญญาณ 3 มิติ เมื่อนับเวลาเป็นมิติที่ 3 หรือ อาจจะครอบคลุมถึงสัญญาณ 3 มิติอื่น ๆ เช่น ภาพ 3 มิติทางการแพทย์ หรือ อาจจะมากกว่า เช่น ภาพ 3 มิติ และ หลายชนิด (multimodal image)



รูปที่ 1 วิดีทัศน์สตรีม (video stream)

2.1.1 วิธีที่ใช้ในการประมวลผลภาพ

1. ปรับปรุงคุณภาพของภาพ (Image Enhancement) เป็นกระบวนการในการแปลงข้อมูลภาพตัวเลข เพื่อที่จะสร้างภาพที่เน้นรายละเอียดที่ต้องการ หรือ ปรับพิกเซลของโทนแสงที่ต้องการของภาพ
2. การกรองภาพหรือการกำจัดสัญญาณรบกวนออกจากภาพ (Image Filters) คือ การนำภาพไปผ่านตัวกรองสัญญาณ เพื่อให้ได้ภาพผลลัพธ์ที่มีคุณสมบัติแตกต่างจากภาพเริ่มต้น

3. การซ้อนทับภาพ (Image Registration) เป็นวิธีการนำข้อมูลของสองภาพหรือมากกว่ามารวมกัน เพื่อให้เกิดภาพใหม่ที่มีข้อมูลภาพสมบูรณ์มากขึ้น
4. การคืนสภาพของภาพ (Image Restoration) คือ การทำให้ภาพคืนสู่สภาพเดิม หรือการปรับปรุงภาพให้เหมาะสมกับการมองเห็น
5. การแบ่งส่วนภาพ (Image Segmentation) เป็นวิธีการแบ่งส่วนใดส่วนหนึ่งของภาพที่สนใจออกมาจากภาพ
6. การหาขอบภาพในวัตถุ (Edge Detection) จะใช้หลักการหาความชันของความเข้มสี (intensity) เนื่องจากที่ขอบรูปจะเป็นบริเวณมีความแตกต่างของสีมาก
7. การบีบอัดภาพ (Image Compression)
8. การสร้างภาพ 3 มิติ (3D Image Reconstruction)

2.2 คอมพิวเตอร์วิทัศน์ (Computer Vision)

คอมพิวเตอร์วิทัศน์เป็นศาสตร์ที่ใช้ในการทำให้คอมพิวเตอร์สามารถเข้าใจภาพ หรือ วิดีทัศน์ในรูปแบบเดียวกับระบบการมองเห็นของมนุษย์

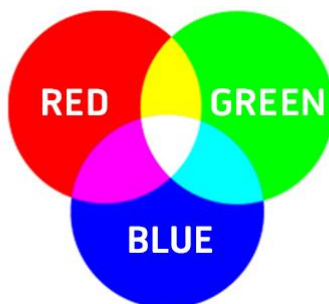
คอมพิวเตอร์วิทัศน์สามารถนำไปใช้งานได้หลายสถานการณ์ ดังนี้

1. Image segmentation คือ การแยกส่วนของภาพออกเป็นหลาย ๆ ส่วน หรือ ชิ้นองค์ประกอบย่อย ๆ เพื่อพิจารณาแยกส่วนกัน
2. Object detection คือ การตรวจหาวัตถุแบบเฉพาะเจาะจงในภาพแต่ละภาพ ซึ่งมีการทำงานในระดับสูงที่สามารถระบุวัตถุหลายชิ้นในภาพเดียวกันได้ เพื่อสร้างกล่องสำหรับการพิจารณา และ ระบุวัตถุทุกชิ้นที่อยู่ในพื้นที่กล่องแต่ละกล่องที่กำหนดขึ้น
3. Facial recognition คือ การจดจำใบหน้า เป็นรูปแบบการระบุวัตถุชิ้นสูงที่ไม่ได้ทำแค่การระบุว่ามิใช่ใบหน้าของมนุษย์อยู่ในภาพเท่านั้น แต่ยังสามารถแยกแยะแต่ละบุคคลออกจากกัน และ ระบุบุคคลที่เจาะจงได้
4. Edge detection คือ การระบุหาขอบ มุมของวัตถุ หรือ ภาพทิวทัศน์ เพื่อให้ทราบได้ง่ายขึ้นว่าองค์ประกอบในภาพมีสิ่งใดบ้าง
5. Pattern detection คือ การระบุวัตถุจากรูปทรง สี หรือ สิ่งบ่งชี้ต่าง ๆ ที่พบในภาพที่เป็นรูปแบบเดียวกันซ้ำ ๆ สำหรับวัตถุประเภทนั้น ๆ
6. Image classification คือ การจัดกลุ่มภาพออกเป็นหมวดหมู่ต่าง ๆ
7. Feature matching คือ การตรวจหารูปแบบ (pattern detection) ที่ระบุจุดที่เหมือนหรือคล้ายคลึงกันในภาพต่าง ๆ เพื่อจัดหมวดหมู่ให้แก่วัตถุและภาพเหล่านั้น

2.3 ระบบสี

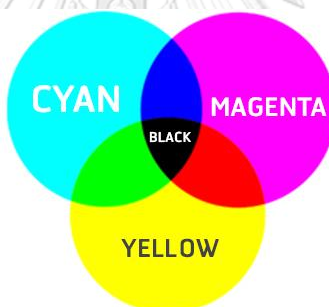
โดยทั่วไปสีในธรรมชาติและสีที่สร้างขึ้นจะมีรูปแบบการมองเห็นของสีที่แตกต่างกัน ซึ่งรูปแบบการมองเห็นสีที่ใช้ในงานด้านกราฟิกนั้น มีอยู่ด้วยกันหลายแบบ ได้แก่

1. RGB (Red, Green, Blue) เป็นระบบสีตามหลักการแสดงสีของเครื่องคอมพิวเตอร์



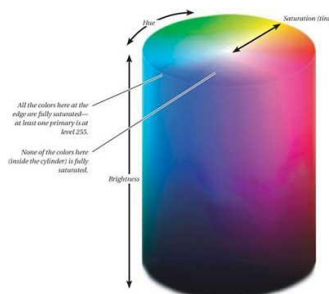
รูปที่ 2 ระบบสี RGB

2. CMYK (Cyan, Magenta, Yellow, Black) เป็นระบบสีตามหลักการแสดงสีของเครื่องพิมพ์



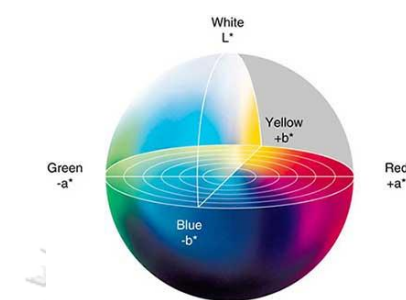
รูปที่ 3 ระบบสี CMYK

3. HSB (Hue, Saturation, Brightness) ตามหลักการมองเห็นสีของสายตามนุษย์



รูปที่ 4 ระบบสี HSB

4. LAB ตามหลักการแสดงสีที่ไม่ขึ้นอยู่กับอุปกรณ์ใด ๆ สามารถใช้กับสีที่เกิดจากอุปกรณ์ทุกชนิด ไม่ว่าจะเป็นจอคอมพิวเตอร์หรือเครื่องพิมพ์ ส่วนประกอบของโหมดสีนี้ได้แก่ L หรือ Luminance เป็นการกำหนดความสว่าง ซึ่งมีค่าตั้งแต่ 0 ถึง 100 ถ้ากำหนดที่ 0 จะเป็นสีดำ ถ้ากำหนดที่ 100 จะเป็นสีขาว
- A เป็นค่าของสีที่ไล่จากสีเขียวไปสีแดง
- B เป็นค่าของสีที่ไล่จากสีน้ำเงินไปสีเหลือง



รูปที่ 5 ระบบสี LAB

5. Grayscale มีเพียง 2 สี คือ สีขาวและสีดำ มีระดับความเข้มของสีดำ 255 ระดับ รวมกับสีขาวอีก 1 สี ดังนั้นจะมีทั้งหมด 256 สี



รูปที่ 6 ระบบสี Grayscale

6. Bitmap ประกอบด้วยสี 2 สี คือ สีขาวและสีดำ บางครั้งเรียกว่า ภาพแบบ 1 บิต ซึ่งแต่ละพิกเซลในภาพจะเป็นได้เพียงสีขาวหรือสีดำเท่านั้น



รูปที่ 7 ระบบสี Bitmap

สำหรับในงานวิจัยนี้จะใช้เป็นการแปลงจากภาพสีให้เป็นภาพขาวดำ หรือ ปรับให้เป็น grayscale นั้นเอง เพื่อนำไปทำการตรวจจับขอบ (edge detection) ของสิ่งกีดขวางต่อไป

2.4 การกรองข้อมูลภาพ (Image Filtering)

การกรองข้อมูลภาพเป็นการนำภาพไปผ่านตัวกรองสัญญาณ เพื่อให้ได้ภาพผลลัพธ์ออกมา โดยภาพผลลัพธ์ที่ได้จะมีคุณสมบัติแตกต่างจากภาพเริ่มต้น

วัตถุประสงค์ของการกรองข้อมูลภาพ คือ การเน้น (enhance) หรือ ลดทอน (attenuate) คุณสมบัติบางประการของภาพ เพื่อให้ได้ภาพที่มีคุณสมบัติตามต้องการ

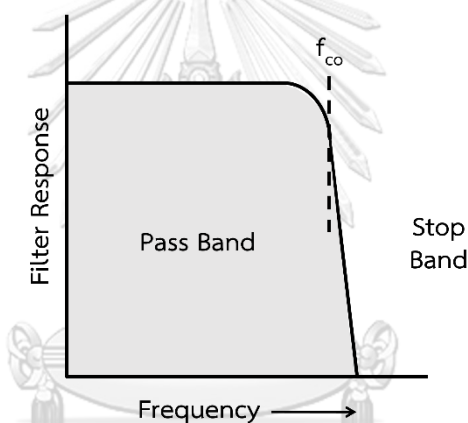
ตัวกรอง คือ ระบบหนึ่งซึ่งรับสัญญาณเข้า (input) ประมวลผลสัญญาณ (process) และ ส่งสัญญาณออก (output)

ในการกรองข้อมูลภาพ จะพิจารณาว่าภาพคือสัญญาณ 2 มิติที่ประกอบขึ้นจากสัญญาณความถี่ต่าง ๆ ผสมกันอยู่ในสัดส่วนที่ต่างกัน การออกแบบตัวกรองจึงเป็นการกำหนดว่า ต้องการกำจัดสัญญาณความถี่ใดออกไป หรือ ต้องการเลือกสัญญาณความถี่ใด

2.4.1 ประเภทของตัวกรอง

ตัวกรองแบ่งออกเป็น 4 ประเภท ตามลักษณะความถี่ ดังนี้

1. ตัวกรองความถี่ต่ำผ่าน (Low-Pass Filter)



รูปที่ 8 ตัวกรองความถี่ต่ำผ่าน (Low-Pass Filter)

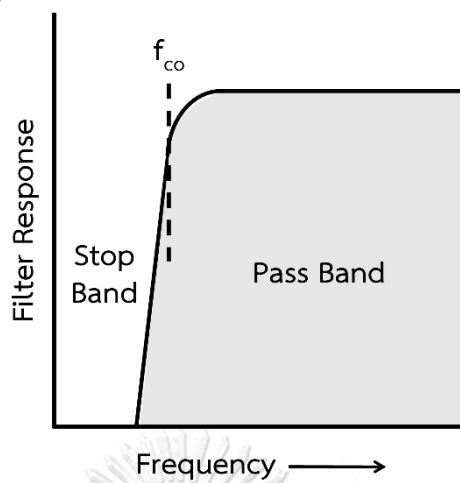


Source Image

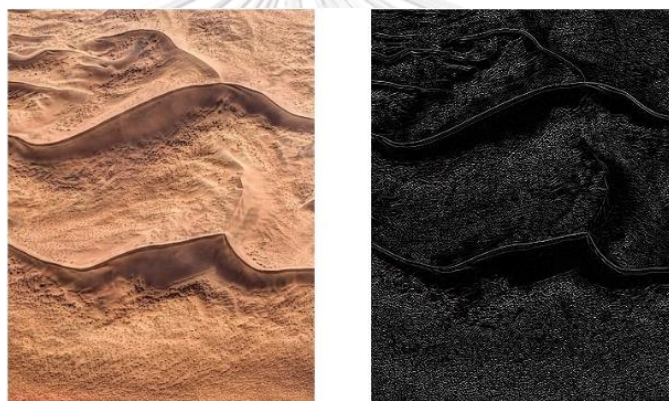
Result Image

รูปที่ 9 สังวัตนาการใน 2 มิติด้วยตัวกรองความถี่ต่ำผ่าน (Low-Pass Filter with 2D Convolution)

2. ตัวกรองความถี่สูงผ่าน (High-Pass Filter)



รูปที่ 10 ตัวกรองความถี่สูงผ่าน (High-Pass Filter)



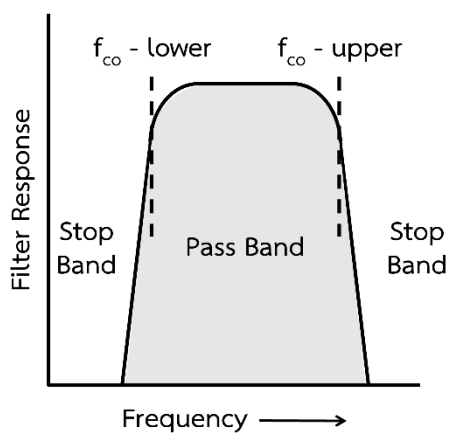
Source Image

Result Image

รูปที่ 11 สัจวัตนาการใน 2 มิติด้วยตัวกรองความถี่สูงผ่าน (High-Pass Filter with 2D Convolution)

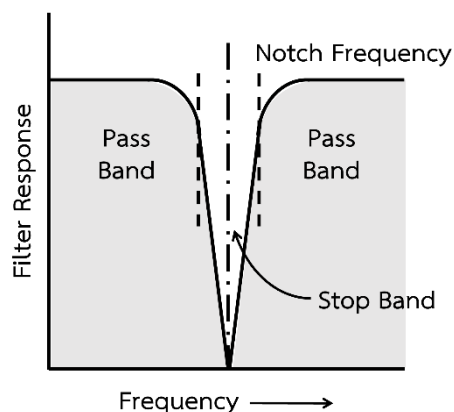
CHULALONGKORN UNIVERSITY

3. ตัวกรองแถบความถี่ผ่าน (Band-Pass Filter)



รูปที่ 12 ตัวกรองแถบความถี่ผ่าน (Band-Pass Filter)

4. ตัวกรองหยุดแถบความถี่ (Band-Stop Filter)



รูปที่ 13 ตัวกรองหยุดแถบความถี่ (Band-Stop Filter)

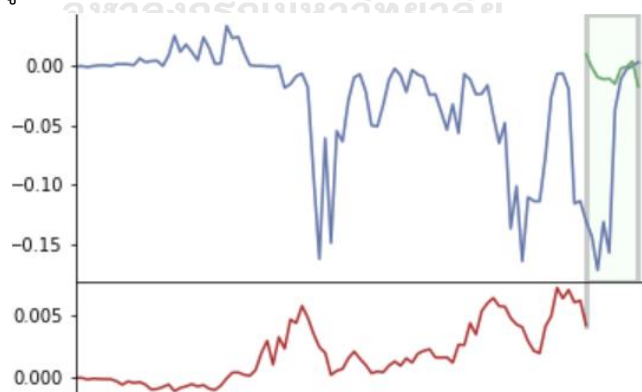
2.4.2 การกรองโดยวิธีสังวัตนาการ (Convolution)

ในการประมวลผลภาพ สังวัตนาการ คือ การกระทำกันระหว่างเทมเพลต (template) กับภาพ

เทมเพลต คือ เมตริกซ์ขนาด $n \times m$ ของชุดตัวเลขที่จะนำไปซ้อนทับกับรูปภาพ ณ ตำแหน่งต่าง ๆ เพื่อหาผลลัพธ์ของสังวัตนาการ

2.5 สังวัตนาการ (Convolution)

สังวัตนาการเป็นการดำเนินการทางคณิตศาสตร์ใน 2 ฟังก์ชัน โดยสร้างฟังก์ชันที่ 3 ที่แสดงว่ารูปร่างของตัวหนึ่งถูกตัดแปลงโดยอีกตัวหนึ่ง



รูปที่ 14 กราฟเส้นแสดงการสังวัตนาการโดยการคูณตัวกรองกับแต่ละส่วน

จากรูปที่ 14 เส้นสีเขียว คือ ตัวกรองที่ไล่คูณกับแต่ละส่วนไปเรื่อย ๆ จากนั้นนำผลคูณทุกช่องมาบวกกัน จนได้ผลลัพธ์เป็นกราฟเส้นสีแดง ซึ่งแบบไม่กลับด้านนี้เรียกว่า สหสัมพันธ์ไขว้ (cross-correlation) สามารถเขียนเป็นสมการทางคณิตศาสตร์ได้ดังนี้

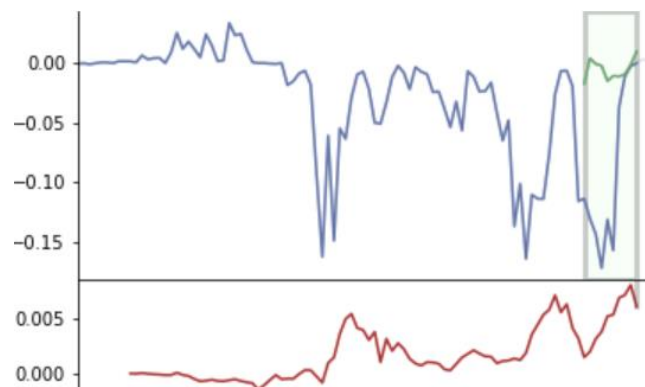
$$(x * k)_m = \sum_n x_n k_{m+n} \quad \dots (1)$$

โดยที่

x คือ อาร์เรย์ (array) ค่าที่ป้อนเข้า

y คือ ตัวกรอง

อย่างไรก็ตาม ด้วยนิยามของสังวัตนาการแล้ว ตัวกรองจะต้องกลับด้าน ซึ่งจะได้ผลลัพธ์ดังรูปที่ 15



รูปที่ 15 กราฟเส้นแสดงการสังวัตนาการโดยการคูณตัวกรองกับแต่ละส่วนแบบกลับด้าน

สามารถเขียนเป็นสมการทางคณิตศาสตร์ได้ดังนี้

$$(x * k)_m = \sum_n x_n k_{m-n} \quad \dots (2)$$

สำหรับกรณีที่ x และ k เป็นฟังก์ชันที่มีค่าแบบต่อเนื่อง สามารถเขียนอยู่ในรูปของสมการปริพันธ์ (Integral) ได้ดังนี้

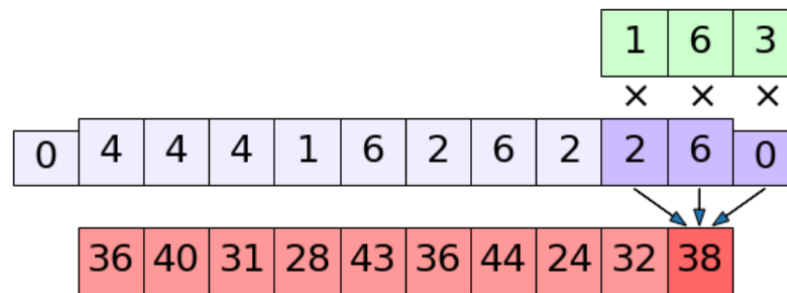
$$(x * k)(t) = \int x(\tau)k(t - \tau)d\tau \quad \dots (3)$$

$$(x * k)(t) = \int x(\tau)k(t + \tau)d\tau$$

โดยปกติแล้ว เมื่อทำสังวัตนาการจะได้ผลลัพธ์ออกมาเป็นอาร์เรย์ที่มีขนาดเล็กลง โดยที่สังวัตนาการสามารถทำได้หลายมิติ ในที่นี้จะอธิบายถึงการทำสังวัตนาการใน 1 มิติ และ 2 มิติ

2.5.1 สัจวัตนาการใน 1 มิติ

จากที่กล่าวไปในข้างต้นว่า การทำสัจวัตนาการจะทำให้ได้ผลลัพธ์ที่มีขนาดเล็กลง ดังนั้นหากต้องการให้ได้ขนาดของผลลัพธ์ที่ไม่มีการเปลี่ยนแปลง โดยทั่วไปจะทำโดยการเพิ่มขอบ (padding) ก่อนที่จะทำสัจวัตนาการ ดังรูปที่ 16



รูปที่ 16 สัจวัตนาการใน 1 มิติ

2.5.2 สัจวัตนาการใน 2 มิติ

การทำสัจวัตนาการใน 2 มิติ อาเรย์ป้อนเข้าและตัวกรองจะเป็น 2 มิติ ผลลัพธ์ที่ได้ก็จะได้เป็น 2 มิติ โดยส่วนมากนิยมนำมาใช้ในเรื่องของการจัดการกับรูปภาพ ดังรูปที่ 17





| | | | | | |
|---|---|---|-----------------|-----------------|-----------------|
| 7 | 7 | 7 | 4 | 0 | 4 |
| 1 | 5 | 2 | 2 | 7 | 1 |
| 4 | 0 | 0 | 1 _{x1} | 0 _{x0} | 4 _{x0} |
| 1 | 3 | 5 | 5 _{x3} | 3 _{x1} | 1 _{x6} |
| 2 | 7 | 7 | 2 _{x5} | 2 _{x2} | 9 _{x2} |

| | | | |
|----|----|----|----|
| 47 | 38 | 59 | 36 |
| 34 | 46 | 44 | 62 |
| 78 | 97 | 81 | 57 |

รูปที่ 17 สัจวัตนาการใน 2 มิติ

2.6 ตัวกรอง (Filter)

ในการประมวลผลภาพ ตัวกรอง เมทริกซ์สังวัตนาการ (convolution matrix) หรือ มาสก์ (mask) เป็นเมทริกซ์ขนาดเล็ก ใช้สำหรับการเบลอ (blurring), การเพิ่มความคมชัด (sharpening), การทำให้งู๋นขึ้น (embossing), การตรวจจับขอบ (edge detection) และ อื่น ๆ

| ตัวดำเนินการ | ตัวกรอง | รูปภาพผลลัพธ์ |
|----------------|---|---|
| Identity | $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ |  |
| Edge detection | $\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$ |  |
| | $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ | |
| | $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$ | |
| Sharpen | $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ |  |
| Blur | $\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ |  |

ตารางที่ 1 ตัวอย่างตัวกรองที่สามารถทำได้โดยการสังวัตนาการกับรูปภาพ

2.7 การแปลงภาพ (Image Transformation)

การแปลงเป็นฟังก์ชันหรือตัวดำเนินการ (operator) ที่รับรูปภาพเป็นข้อมูลนำเข้า (input) และ สร้างรูปภาพเป็นข้อมูลนำออก (output)

การแปลงภาพสามารถพิจารณาได้จากสมการนี้

$$G(x, y) = T(f(x, y)) \quad \dots (4)$$

โดยที่

$f(x, y)$ คือ รูปภาพนำเข้า (input) ที่ต้องใช้ฟังก์ชันการแปลง

$G(x, y)$ คือ รูปภาพผลลัพธ์ (output) หรือ รูปภาพที่ผ่านการประมวลผลแล้ว

T คือ ฟังก์ชันการแปลง

ความสัมพันธ์ระหว่างรูปภาพนำเข้า (input) และ รูปภาพผลลัพธ์ (output) นี้ สามารถแสดงได้ดังสมการนี้

$$s = T(r) \quad \dots (5)$$

โดยที่

r คือ ค่าพิกเซลหรือระดับสีเทาของ $f(x, y)$ ณ จุดใด ๆ

s คือ ค่าพิกเซลหรือระดับสีเทาของ $g(x, y)$ ณ จุดใด ๆ

2.7.1 ประเภทการแปลง

การแปลงมี 2 ประเภท ดังนี้

1. การแปลงแบบฟูรีเยร์ (Fourier Transform) ใช้สำหรับการตรวจจับขอบภาพ (edge detection)
2. การแปลงแบบโคไซน์ไม่ต่อเนื่อง (Discrete Cosine Transform : DCT) ใช้สำหรับการบีบอัดภาพ

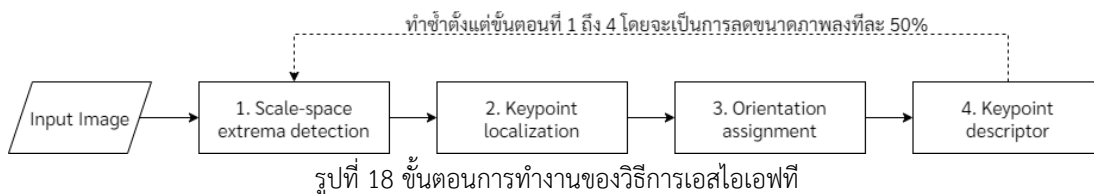
สำหรับในงานวิจัยนี้จะใช้การแปลงแบบฟูรีเยร์เป็นหลัก เพราะนำมาใช้สำหรับทำการตรวจจับขอบ (edge detection) ของวัตถุที่อยู่ในรูปภาพ

2.8 Scale Invariant Features Transform (SIFT)

เอสไอเอฟที (SIFT) เป็นวิธีการหนึ่งในคอมพิวเตอร์วิทัศน์ สำหรับคำนวณหาจุดสนใจ (Keypoint) ในรูปภาพหนึ่ง ๆ และ คำนวณหาคุณลักษณะของจุดสนใจหนึ่ง ๆ ที่ทำได้ โดยการเอาจุดสนใจในรูปที่ไม่ได้ขึ้นอยู่กับมาตราส่วน (scale) การกำหนดทิศทาง ตำแหน่ง มุมมอง แสงสว่าง หรือเงาของวัตถุในภาพ ซึ่งสามารถนำมาใช้ในการวิเคราะห์ เพื่อหาจุดเด่นของรูปภาพที่ได้รับเข้ามา ซึ่งลักษณะเด่นที่ได้จากวัตถุลักษณะเดียวกันจะมีลักษณะที่คล้ายคลึงกัน

โดยวิธีการของ SIFT จะประกอบด้วย 4 ขั้นตอนที่สำคัญ [1] [2] [3] ได้แก่

1. การหาปริภูมิค่าในมิติขนาดและระยะทาง (Scale-space extrema detection)
2. การกำหนดตำแหน่งจุดสนใจ (Keypoint localization)
3. การกำหนดทิศทางของจุดสนใจ (Orientation assignment)
4. การสร้างคำอธิบายลักษณะเด่นของภาพ (Keypoint descriptor)



2.8.1 การหาปริภูมิค่าในมิติขนาดและระยะทาง (Scale-space extrema detection)

เป็นการหาลักษณะเด่นของภาพที่ไม่ขึ้นกับขนาดหรือทิศทาง โดยจะใช้วิธีการเบลอภาพ (blur) ด้วยฟังก์ชันเกาส์เซียน (Gaussian function) ไปในแต่ละชั้น (octave) ซึ่งในแต่ละชั้นก็จะมีรูปหลาย ๆ ระดับการเบลอ ซึ่งมีขนาดรูปที่เท่ากัน โดยที่แต่ละระดับการเบลอจะเบลอจากระดับปกติแล้วค่อย ๆ เพิ่ม σ (Scale parameter) ซึ่งจะมีผลทำให้ภาพเบลอมากยิ่งขึ้น แล้วทำซ้ำกับขั้นต่อไปเรื่อย ๆ โดยขั้นต่อไปก็จะมีขนาดของรูปเป็นครึ่งหนึ่งของขั้นก่อนหน้า

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad \dots (6)$$

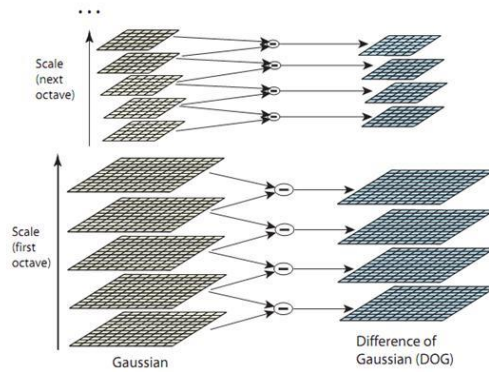
$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad \dots (7)$$

โดยที่

- L คือ ภาพที่ผ่านการเบลอ
- G คือ ตัวกรองแบบเกาส์เซียน (Gaussian) ที่มีขนาด σ
- I คือ ภาพต้นฉบับ
- x, y คือ พิกัดบนภาพ
- σ คือ ตัวแปรขนาดของการเบลอ ยิ่งมีค่ามาก ยิ่งเบลอมาก
- * คือ การสังวัตนาการ (convolution) บนภาพ I โดยใช้ Gaussian blur G

2.8.2 การกำหนดตำแหน่งจุดสนใจ (Keypoint localization)

จากการหาปริภูมิค่าในมิติขนาดและระยะทางในขั้นตอนก่อนหน้า จะนำมาหาจุดที่น่าจะเป็นลักษณะสำคัญของภาพ โดยจะทำการจับคู่ภาพของแต่ละชั้นมาหาความต่างของแต่ละผลลัพธ์ที่ได้จากการเบลอ (Difference of Gaussian : DoG) โดยจะทำทุกภาพในแต่ละชั้นและทำซ้ำจนครบทุกชั้น

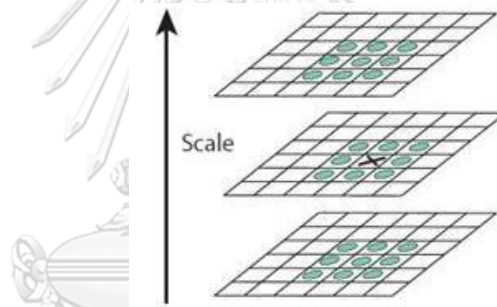


รูปที่ 19 การหา Difference of Gaussian ในแต่ละชั้น (Octave)

การหาจุดสนใจ แบ่งออกเป็น 2 ส่วน คือ

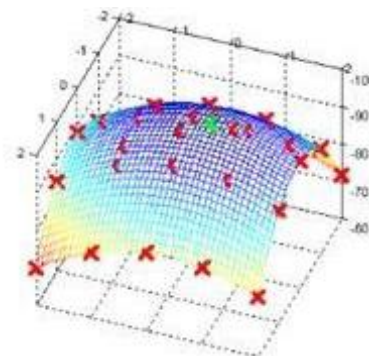
1. Locate maxima/minima in DoG images

เป็นการประมาณหาพิกเซลที่มีค่าสูงสุดหรือต่ำสุดแบบคร่าว ๆ เมื่อทำการเปรียบเทียบกับจุดรอบข้าง 26 จุด ดังรูปที่ 20



รูปที่ 20 จุด X เป็นจุดอ้างอิงเปรียบเทียบกับจุดรอบข้าง 26 จุด

จุฬาลงกรณ์มหาวิทยาลัย
CHULALON



รูปที่ 21 จุดสีเขียว คือ subpixel เป็นจุดสูงสุดหรือต่ำสุด

2. Find subpixel maxima/minima

จะใช้สมการของเทย์เลอร์ (Taylor) เพื่อทำการประมาณหาพิกเซลย่อย (subpixel) ที่เป็นจุดที่มีค่าสูงสุดหรือต่ำสุดอย่างแท้จริง จากพิกเซลที่ได้เลือกมาจากขั้นตอนก่อนหน้า

$$D(x) = D + \frac{\partial D}{\partial x} + \frac{1}{2}x^T \frac{\partial^2 D}{\partial x^2} x \quad \dots \quad (8)$$

จากขั้นตอนที่ผ่านมาจะได้จุดสนใจจำนวนมาก ซึ่งบางจุดจะวางเรียงตัวอยู่บนเส้นขอบ (edge) หรือ บางจุดมีความคมชัด (contrast) โดยจุดเหล่านี้จะเป็นจุดที่จะทำการตัดออก เพื่อให้ได้ผลลัพธ์สุดท้ายที่แต่ละจุดจะมีความเหมาะสมที่จะเป็นจุดสนใจมากที่สุด ประกอบด้วย 2 ขั้นตอน คือ

1. การลบคุณสมบัติความคมชัดต่ำ (Removing low contrast features)

ทำการหาค่าความเข้ม (intensity) ของจุดที่เป็นพิกเซลย่อย ซึ่งเป็นการประมาณมาจากขั้นตอนก่อนหน้า นำมาเปรียบเทียบกับค่าที่กำหนดค่าหนึ่ง ถ้าหากมีค่าน้อยกว่าจะทำการตัดจุดนั้นออกจากจุดสนใจ

2. การลบเส้นขอบ (Removing edges)

ทำการคำนวณการไล่ระดับสี (gradient) 2 ตัวของจุดที่เป็นจุดสนใจ ซึ่งตั้งฉากกันและกัน โดยสามารถสรุปความสัมพันธ์บนภาพจากจุดทั้ง 2 ได้ 3 แบบ คือ

1. พื้นที่ราบ (flat region) จะมีการไล่ระดับสีที่มีค่าน้อยทั้งคู่

2. ขอบ (edge) จะมีการไล่ระดับสีที่มีค่ามากสำหรับตัวที่ตั้งฉากกับขอบ และ มีการไล่ระดับสีที่มีค่าน้อยสำหรับตัวที่ขนานไปกับ

ขอบ

3. มุม (corner) จะมีการไล่ระดับสีที่มีค่ามากทั้งคู่

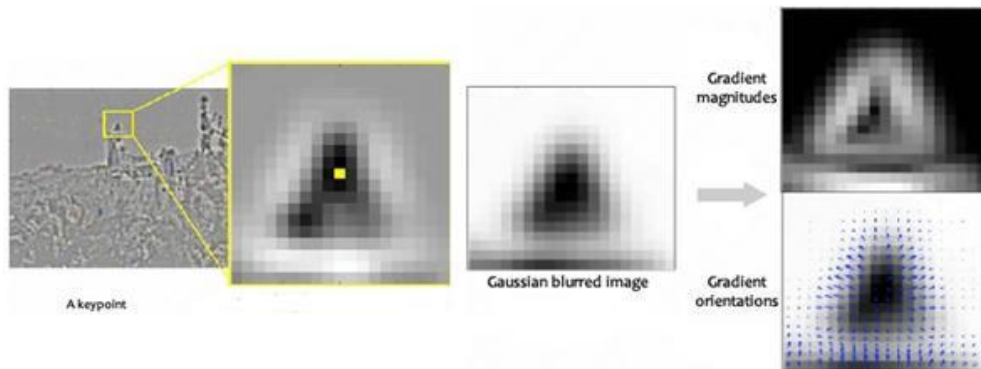
โดยจะพิจารณาจุดสนใจที่มีลักษณะเป็นมุมเป็นหลัก ซึ่งถ้าหากมีค่าของการไล่ระดับสีที่มากพอทั้งคู่ จะนิยามให้เป็นจุดสนใจ แต่ถ้าหากไม่มากพอจะถูกตัดทิ้ง

2.8.3 การกำหนดทิศทางของจุดสนใจ (Orientation assignment)

ทำการเก็บรวบรวมขนาด $m(x,y)$ และ ทิศทาง $\Theta(x,y)$ ของการไล่ระดับสีของบริเวณรอบ ๆ จุดสนใจ เพื่อที่จะกำหนดทิศทางให้จุดสนใจนั้น ๆ

$$m(x, y, \sigma) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad \dots (9)$$

$$\theta(x, y, \sigma) = \tan^{-1} \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \quad \dots (10)$$



รูปที่ 22 ขนาดและทิศทางของจุดสนใจ¹

จากนั้นนำเอาขนาดและทิศทางของการไล่ระดับสีของพิกเซลรอบ ๆ จุดสนใจมาทำการสร้างภาพแท่งความถี่ (histogram) ซึ่งจะมีแกน x ช่วงขององศา และ แกน y เป็นขนาดของการไล่ระดับสีของพิกเซลนั้น ๆ รวมกับพิกเซลอื่น ๆ ที่แบ่งแยกตามแต่ละช่องของแกน x เดียวกัน ต่อมาจะทำการเพิ่มจุดสนใจ โดยมีเงื่อนไขว่า ถ้าจุดสนใจใดมีจุดสูงสุด (peak) ของภาพแท่งความถี่ที่มีขนาดมากกว่า 80% ของจุดสูงสุดก็จะแบ่งออกเป็นจุดสนใจใหม่

2.8.4 การสร้างคำอธิบายลักษณะเด่นของภาพ (Keypoint descriptor)

ทำการสร้าง 16×16 window (window คือ บริเวณที่จะทำการเก็บข้อมูลทิศทาง ซึ่งมีขนาด คือ $1.5 \cdot \sigma$) รอบจุดสนใจ และ ทำการแบ่งออกเป็น 4×4 windows ทั้งหมด 16 ชุด โดยในแต่ละชุดจะทำการคำนวณหาขนาดและทิศทางของการไล่ระดับสี แล้วนำมาสร้างเป็นภาพแท่งความถี่ที่มีขนาด 8 bin (แกน x แบ่ง 8 ช่วง ช่วงละ 45 องศา) โดยที่ขนาดของภาพแท่งความถี่ในแต่ละส่วนจะขึ้นกับขนาดคุณตัวถ่วงน้ำหนัก (ระยะทางที่ห่างจากจุดสนใจ) ดังนั้นเมื่อทำการคำนวณเสร็จสิ้นผลลัพธ์ที่ได้คือ $4 \times 4 \times 8 = 128$ ซึ่งจะทำการ normalize ต่อและจะใช้เป็น feature vector ของแต่ละจุดสนใจ

¹ ที่มา : <https://aishack.in/tutorials/sift-scale-invariant-feature-transform-keypoint-orientation/#:~:text=To%20assign%20an%20orientation%20we,is%20assigned%20to%20the%20keypoint.>

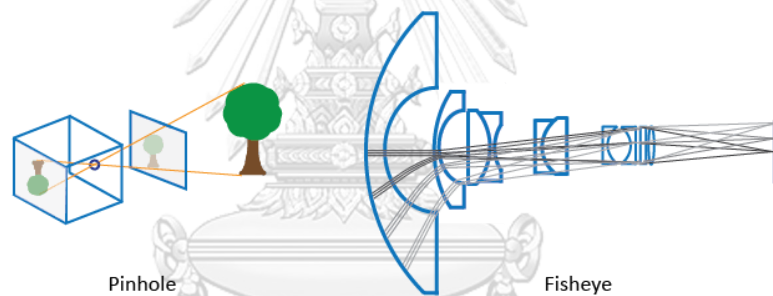
2.9 การปรับเทียบกล้อง (Camera Calibration)

การปรับเทียบกล้องเป็นกระบวนการประมาณค่าพารามิเตอร์ของกล้องโดยใช้ภาพที่มีรูปแบบการปรับเทียบ (calibration pattern)

ในการประมาณค่าพารามิเตอร์ของกล้อง ต้องมีจุดโลก 3 มิติ และ จุดภาพ 2 มิติที่สอดคล้องกัน ซึ่งสามารถทำได้โดยใช้รูปภาพหลายรูปของรูปแบบการปรับเทียบ เช่น กระจาดนหมากกรุก อีกทั้งยังสามารถแก้ไขพารามิเตอร์ของกล้องได้ หลังจากปรับเทียบกล้องแล้ว ในการประเมินความถูกต้องของพารามิเตอร์โดยประมาณสามารถทำได้ดังนี้

1. พล็อต (plot) ตำแหน่งสัมพัทธ์ (relative locations) ของกล้อง และ รูปแบบการปรับเทียบ
2. คำนวณข้อผิดพลาดในการฉายซ้ำ
3. คำนวณข้อผิดพลาดในการประมาณค่าพารามิเตอร์

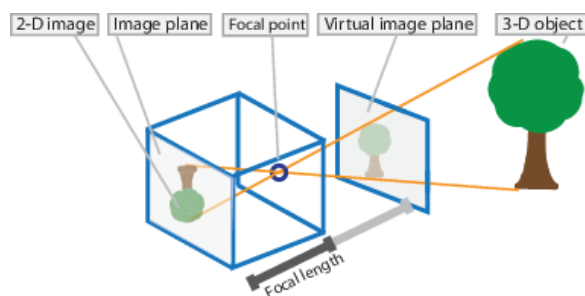
เครื่องมือที่นำมาใช้ส่วนมากจะเป็นอัลกอริทึมสำหรับการปรับเทียบแบบจำลองกล้องรูเข็ม (pinhole camera model) และ แบบจำลองกล้องฟิชอาย (fisheye camera model)



รูปที่ 23 มุมมองรูเข็มและมุมมองฟิชอาย

2.9.1 แบบจำลองกล้องรูเข็ม (Pinhole Camera Model)

กล้องรูเข็มเป็นกล้องธรรมดาที่ไม่มีเลนส์และมีรูรับแสงขนาดเล็กเพียงช่องเดียว รังสีของแสงจะลอดผ่านรูรับแสงและฉายภาพกลับด้านที่ฝั่งตรงข้ามของกล้อง โดยคิดว่าระนาบภาพเสมือนอยู่หน้ากล้อง และมีภาพตรงของฉาก



รูปที่ 24 แบบจำลองกล้องรูเข็ม (Pinhole Camera Model)

พารามิเตอร์กล้องรูเข็มจะแสดงในเมทริกซ์ 4×3 เรียกว่า เมทริกซ์กล้อง (camera matrix) เมทริกซ์นี้จับคู่ฉากโลก 3 มิติในระนาบของภาพ อัลกอริทึมการปรับเทียบจะคำนวณเมทริกซ์ของกล้องโดยใช้พารามิเตอร์ภายนอก (extrinsic) และ ภายใน (intrinsic) พารามิเตอร์ภายนอกแสดงถึงตำแหน่งของกล้องในฉาก 3 มิติ และ พารามิเตอร์ภายในแสดงถึงจุดศูนย์กลางของเลนส์ (optical center) และความยาวโฟกัสของกล้อง

$$w \underbrace{[x \ y \ 1]}_{\text{Image points}} = \underbrace{[X \ Y \ Z \ 1]}_{\text{World points}} P$$

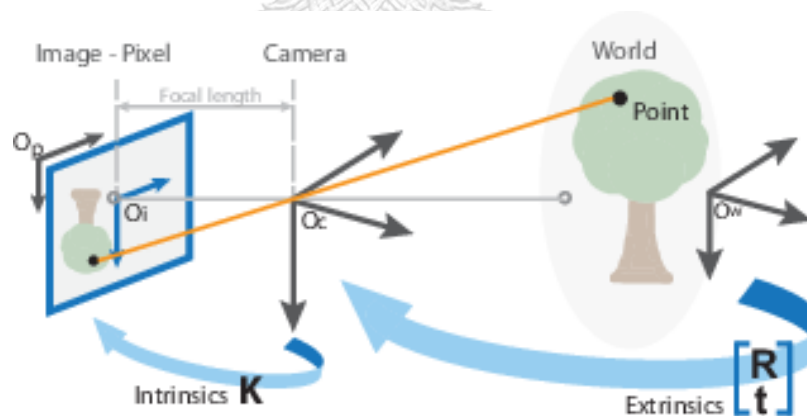
Scale factor Image points World points

$$P = \begin{bmatrix} R \\ t \end{bmatrix} K$$

Camera matrix Extrinsic Intrinsic matrix
Rotation and translation

รูปที่ 25 เมทริกซ์และพารามิเตอร์ที่เกี่ยวข้องกับแบบจำลองกล้องรูเข็ม

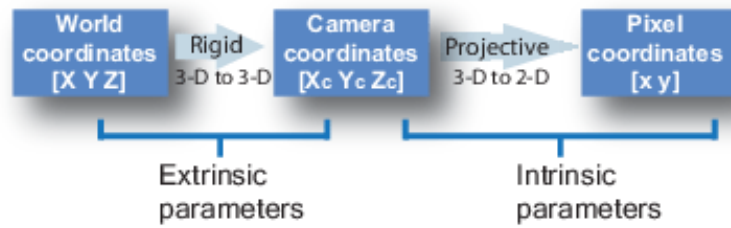
จุดโลกจะถูกแปลงเป็นพิกัดของกล้องโดยใช้พารามิเตอร์ภายนอก จากนั้นพิกัดของกล้องจะจับคู่กับระนาบภาพโดยใช้พารามิเตอร์ภายใน



รูปที่ 26 การแปลงพารามิเตอร์ภายนอกไปเป็นพารามิเตอร์ภายในของแบบจำลองกล้องรูเข็ม

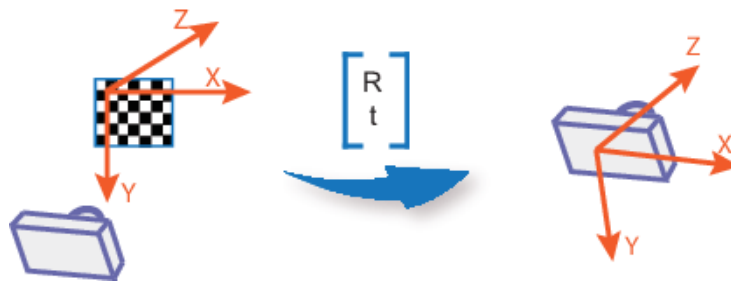
2.9.1.1 พารามิเตอร์การปรับเทียบกล้องสำหรับกล้องรูเข็ม

อัลกอริทึมการปรับเทียบจะคำนวณเมทริกซ์ของกล้องโดยใช้พารามิเตอร์ภายนอกและภายใน พารามิเตอร์ภายนอกแสดงถึงการเปลี่ยนแปลงที่จากระบบพิกัดโลก 3 มิติไปเป็นระบบพิกัดของกล้อง 3 มิติ และ พารามิเตอร์ภายในแสดงถึงการแปลงการฉายภาพจากพิกัดของกล้อง 3 มิติไปเป็นพิกัดภาพ 2 มิติ



รูปที่ 27 การแปลงจากพิกัดโลก 3 มิติไปเป็นพิกัดภาพ 2 มิติของกล้องรูเข็ม

พารามิเตอร์ภายนอก ประกอบด้วย การหมุน (rotation : R) และ แปล (translation : t) ซึ่งที่มาของระบบพิกัดของกล้องอยู่ที่จุดศูนย์กลางของเลนส์ และ แกน x แกน y ใช้กำหนดระนาบภาพ



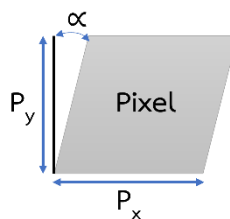
รูปที่ 28 พารามิเตอร์ภายนอกของกล้องรูเข็ม

พารามิเตอร์ภายใน ประกอบด้วย ความยาวโฟกัส จุดศูนย์กลางของเลนส์ หรือเรียกว่า จุดสำคัญ (principal point) และ ค่าสัมประสิทธิ์ความเอียง (skew coefficient)

เมทริกซ์ภายในของกล้องถูกกำหนดเป็น K ได้ดังนี้

$$K = \begin{bmatrix} f_x & 0 & 0 \\ s & f_y & 0 \\ c_x & c_y & 1 \end{bmatrix}$$

ความเอียงของพิกเซลถูกกำหนดได้ดังนี้



รูปที่ 29 ความเอียงของพิกเซล

โดยที่

$[c_x, c_y]$ คือ จุดศูนย์กลางของเลนส์ (จุดสำคัญ) ในหน่วยพิกเซล

(f_x, f_y) คือ ความยาวโฟกัสในหน่วยพิกเซล

F คือ ความยาวโฟกัสในหน่วยโลก

(P_x, P_y) คือ ขนาดของพิกเซลในหน่วยโลก

s คือ ค่าสัมประสิทธิ์ความเอียง ซึ่งไม่ใช่ศูนย์ หากแกนภาพไม่ตั้งฉาก

$$f_x = \frac{F}{P_x} \quad \dots (11)$$

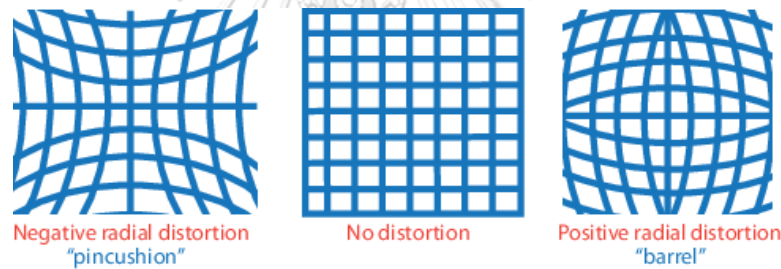
$$f_y = \frac{F}{P_y} \quad \dots (12)$$

$$s = f_x \tan \alpha \quad \dots (13)$$

2.9.2 การบิดเบือนในแนวรัศมี (Radial Distortion)

การบิดเบือนในแนวรัศมีเกิดจากที่เลนส์ฉายภาพเสมือนจริง โดยการที่แสงเข้าทางด้านหน้า และออกทางด้านหลังของเลนส์

การบิดเบือนบางครั้งเรียกว่า การบิดเบี้ยวทางเรขาคณิต ซึ่งส่งผลต่อการแสดงรูปทรงเรขาคณิต เมื่อรังสีของแสงโค้งงอใกล้ขอบเลนส์มากกว่าที่จุดศูนย์กลางของเลนส์ ดังนั้นหากเลนส์ยิ่งเล็ก ความเพี้ยนจะยิ่งมาก

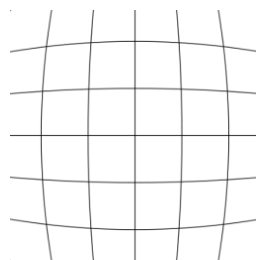


รูปที่ 30 การบิดเบือนในแนวรัศมีในรูปแบบต่าง ๆ

จุฬาลงกรณ์มหาวิทยาลัย

2.9.2.1 การบิดเบือนแบบบาร์เรล (Barrel Distortion)

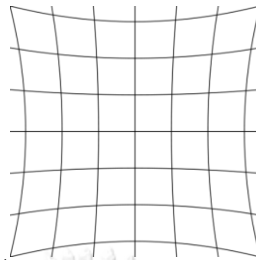
การบิดเบือนแบบบาร์เรลเป็นความผิดเพี้ยนทางเรขาคณิตที่เส้นตรงปรากฏเป็นเส้นโค้งออก จากจุดศูนย์กลางของภาพ ในลักษณะที่ขวมปองคล้ายถังเบียร์ ซึ่งเกิดจากการขยายของภาพที่ ศูนย์กลางของเลนส์มากกว่าที่ขอบ โดยส่วนมากการบิดเบือนในลักษณะนี้จะพบได้ในเลนส์ฟิชอาย



รูปที่ 31 การบิดเบือนแบบบาร์เรล

2.9.2.2 การบิดเบือนแบบหมอนอิง (Pincushion Distortion)

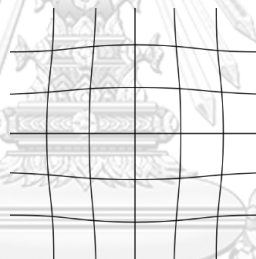
การบิดเบือนแบบหมอนอิงเป็นความผิดเพี้ยนทางเรขาคณิตที่เส้นตรงปรากฏเป็นเส้นโค้งเข้าหาจุดศูนย์กลางของภาพ ทำให้มีลักษณะคล้ายกับเข็มหมุดพุ่งเข้ามาปักหมอนอิง เกิดจากการขยายของภาพที่ขอบเลนส์มากกว่าจุดศูนย์กลาง



รูปที่ 32 การบิดเบือนแบบหมอนอิง

2.9.2.3 การบิดเบือนแบบหนวด (Mustache Distortion)

การบิดเบือนแบบหนวดเป็นความผิดเพี้ยนทางเรขาคณิตที่จะแสดงการบิดเบือนแบบบาร์เรลใกล้กับศูนย์กลางของเลนส์ แล้วค่อย ๆ เปลี่ยนเป็นการบิดเบือนแบบหมอนอิงเมื่อใกล้กับขอบเลนส์



รูปที่ 33 การบิดเบือนแบบหนวด

จุฬาลงกรณ์มหาวิทยาลัย

ค่าสัมประสิทธิ์การบิดเบือนในแนวรัศมี แสดงจุดที่บิดเบือนเป็น $(x_{\text{distorted}}, y_{\text{distorted}})$ ซึ่งสามารถคำนวณได้ตามสมการที่ 16 และ 17

$$x_{\text{distorted}} = x(1 + k_1 * r^2 + k_2 * r^4 + k_3 * r^6) \quad \dots (16)$$

$$y_{\text{distorted}} = y(1 + k_1 * r^2 + k_2 * r^4 + k_3 * r^6) \quad \dots (17)$$

โดยที่

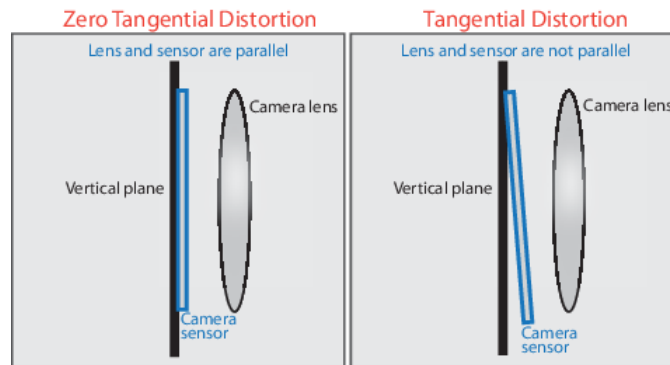
x, y คือ ตำแหน่งพิกเซลที่ไม่บิดเบี้ยว x และ y อยู่ในพิกัดภาพปกติ ซึ่งพิกัดภาพปกติจะคำนวณได้จากพิกัดพิกเซลโดยการแปลงเป็นจุดศูนย์กลางของเลนส์ และหารด้วยความยาวโฟกัสเป็นพิกเซล ดังนั้น x และ y จึงไม่มีมิติ

k_1, k_2, k_3 คือ ค่าสัมประสิทธิ์การบิดเบือนในแนวรัศมีของเลนส์

$$r^2 = x^2 + y^2 \quad \dots (18)$$

2.9.3 การบิดเบือนในแนวสัมผัส (Tangential Distortion)

การบิดเบือนในแนวสัมผัสเกิดขึ้นเมื่อเลนส์ไม่ขนานกับระนาบภาพ



รูปที่ 34 การบิดเบือนในแนวสัมผัส

ค่าสัมประสิทธิ์การบิดเบือนในแนวสัมผัส แสดงจุดที่บิดเบือนเป็น $(x_{\text{distorted}}, y_{\text{distorted}})$ ซึ่งสามารถคำนวณได้ตาม 2 สมการนี้

$$x_{\text{distorted}} = x + [(2 * p_1 * x * y) + p_2(r^2 + 2x^2)] \quad \dots (19)$$

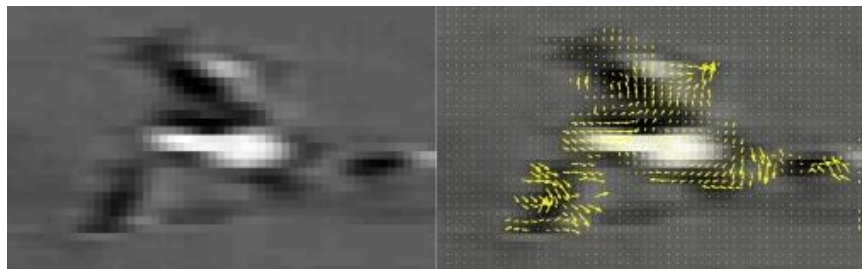
$$y_{\text{distorted}} = y + [p_1(r^2 + 2y^2) + (2 * p_2 * x * y)] \quad \dots (20)$$

โดยที่

p_1, p_2 คือ ค่าสัมประสิทธิ์การบิดเบือนในแนวสัมผัสของเลนส์

2.10 ออพติคัลโฟลว์ (Optical Flow)

ออพติคัลโฟลว์เป็นหนึ่งวิธีการตรวจจับการเคลื่อนที่ของวัตถุภายในภาพ โดยจะทำการวิเคราะห์จากการเปลี่ยนแปลงความเข้มของแสงภายในภาพไปที่ลำดับภาพ (image sequence) ซึ่งประกอบด้วย ขนาด และ ทิศทางของความเข้มแสงที่เปลี่ยนตำแหน่งไป ดังนั้นวัตถุที่เปลี่ยนตำแหน่งไปจะยังคงความเข้มของแสงเท่าเดิม ทำให้ผลของการไหลของแสงของวัตถุที่เปลี่ยนตำแหน่งไปนั้น จะมีขนาดและทิศทางที่ใกล้เคียงกัน



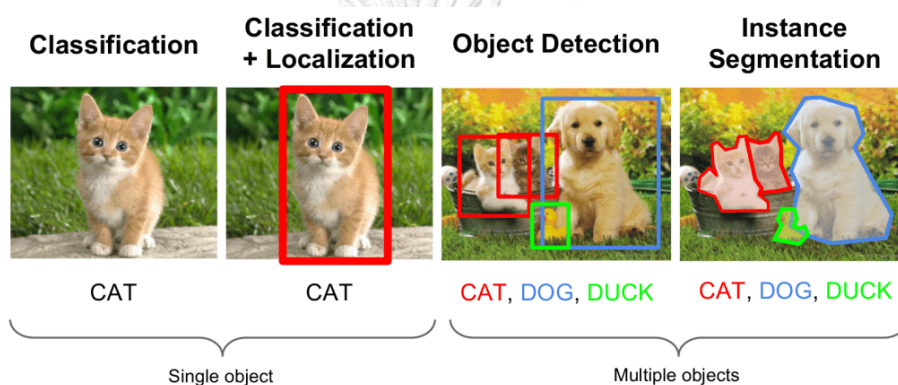
รูปที่ 35 การไหลของแสงเทียบกับภาพต้นฉบับ¹

¹ ที่มา : Rajamangala University of Technology Isan Image Processing & Computer Vision Optical Flow Motion Page 3

2.11 การตรวจจับวัตถุ (Object Detection)

การตรวจจับวัตถุ คือ เทคโนโลยีทางคอมพิวเตอร์ ซึ่งมีหลักการเกี่ยวกับคอมพิวเตอร์วิทัศน์ (Computer Vision) และ การประมวลผลภาพ (Image Processing) ที่ใช้ในงานด้านปัญญาประดิษฐ์ (Artificial Intelligence) สำหรับตรวจจับวัตถุชนิดที่กำหนด เช่น มนุษย์ รถยนต์ อาคาร ที่อยู่ในรูปภาพ หรือ วิดีทัศน์

การตรวจจับวัตถุในรูปภาพ สามารถเจาะลึกลงไปได้หลายแขนง เช่น การตรวจจับใบหน้ามนุษย์ (Face Detection) หรือ การตรวจจับคนเดินถนน (Pedestrian Detection) ซึ่งสามารถนำไปประยุกต์ใช้กับงานได้หลากหลายด้าน เช่น รักษาความปลอดภัย และ รถยนต์ไร้คนขับ เป็นต้น



รูปที่ 36 การตรวจจับวัตถุเดียวและหลายวัตถุในรูปภาพ ¹

ในงานวิจัยนี้ได้ทำการเปรียบเทียบประสิทธิภาพของแบบจำลองที่นำมาใช้ในการตรวจจับวัตถุในรูปภาพดังนี้

2.11.1 Single-Shot Detector (SSD)

SSD คือ แบบจำลองสำหรับการตรวจจับวัตถุที่จะตรวจหา และ จำแนกวัตถุทั้งหมดทุกชิ้นที่อยู่ภายในภาพ ซึ่งมีบล็อกขนาด 38×38 และ 76×76 จะรับได้เฉพาะความละเอียดสูง

แบบจำลองนี้จะตรวจจับวัตถุจำนวน 90 classes ตามใน COCO dataset ซึ่งเป็นชุดข้อมูลขนาดใหญ่

SSD ถูกออกแบบมาสำหรับการตรวจจับวัตถุแบบเรียลไทม์ ใช้ R-CNN (Region Based Convolutional Neural Networks) ที่เร็วขึ้น ในการสร้างขอบเขตเพื่อจัดประเภทวัตถุ ถือว่าเป็น

¹ ที่มา : Stanford University 2016 winter lectures CS231n Fei-Fei Li & Andrej Karpathy & Justin Johnson

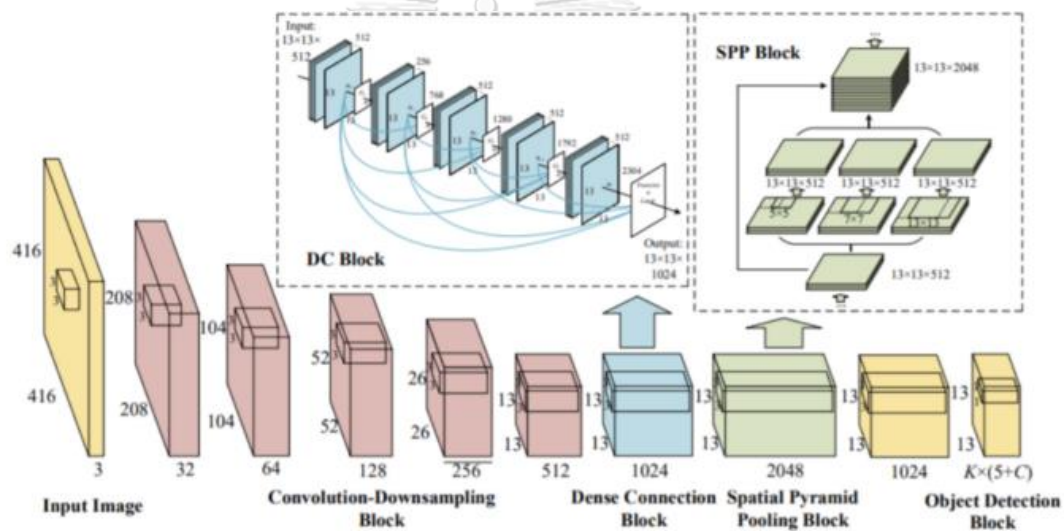
จุดเริ่มต้นของความแม่นยำ แต่กระบวนการทำงานทั้งหมดจะอยู่ที่ 7 เฟรมต่อวินาที ต่ำกว่าความต้องการของการประมวลผลแบบเรียลไทม์

ต่อมา SSD ได้ปรับปรุงกระบวนการให้เร็วสูงขึ้น โดยใช้ภาพที่มีความละเอียดต่ำ ทำให้ได้ความเร็วในการประมวลผลแบบเรียลไทม์และความแม่นยำเฉลี่ยของการคาดการณ์เพิ่มขึ้น

2.11.2 You Only Look Once (YOLO)

YOLO คือ แบบจำลองสำหรับการตรวจจับวัตถุแบบเรียลไทม์ (Real-time Object Detection Model) [4] ที่มีความโดดเด่นเรื่องความเร็วและความถูกต้อง

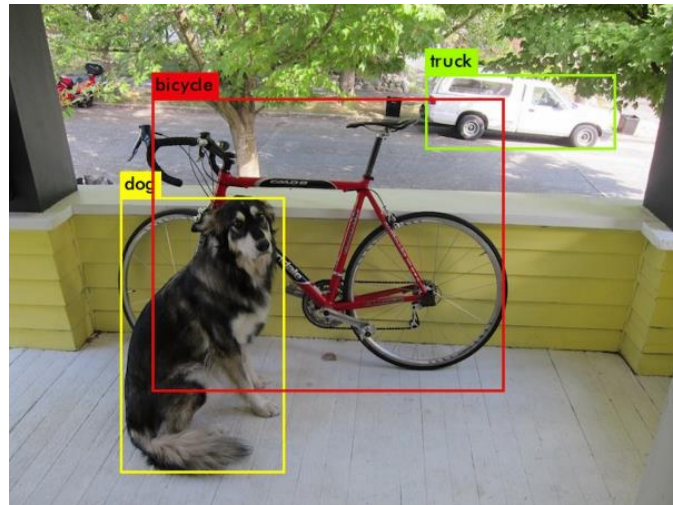
ความโดดเด่นของ YOLO คือ สามารถตรวจจับได้แม้กระทั่งวัตถุที่ซ้อนกัน โดยมีโครงสร้างที่ค่อนข้างซับซ้อนของ grid ในแต่ละชั้น (Layers) ที่เล็กลงเรื่อย ๆ



รูปที่ 37 โครงสร้างของ YOLO ที่เป็นหลายชั้นแบบ Neural Network โดย grid จะถี่ขึ้นเรื่อย ๆ จนได้ผลลัพธ์ที่ดี¹

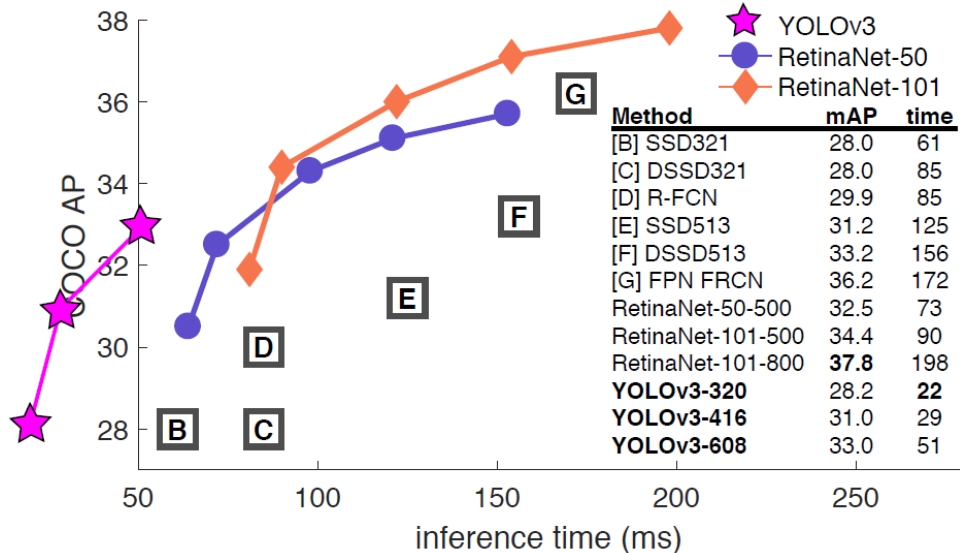
หลักการของ YOLO คือ ถ้ามีสุนัข จักรยาน และ รถบรรทุกอยู่ในรูปเดียวกัน (ดังรูปที่ 38) แบบจำลองจะพยายามตีกรอบวัตถุ (rectangle object) เหล่านั้นไว้ โดยหาจุดกึ่งกลางของแต่ละวัตถุ แล้วค่อยครอบกล่อง (box) เอาไว้ แล้วบอกว่าสิ่งนั้นคืออะไร โดยมีแบบจำลองพื้นฐานอยู่ประมาณ 80 classes ที่ถูกเทรนเอาไว้ และสามารถบอกได้ว่าความน่าจะเป็นมีเท่าไรจาก classes ที่มี

¹ ที่มา : <https://grassrootengineer.medium.com/yolo-คืออะไร-ในงาน-object-detection-83b3a7461114>



รูปที่ 38 การตรวจจับวัตถุที่ทับซ้อนกันของ YOLO

ปัจจุบัน YOLO มีหลายแบบจำลองให้เลือกใช้ และ ในแต่ละเวอร์ชัน (version) มีโครงสร้างที่ไม่เหมือนกัน YOLOv3 รวดเร็วและแม่นยำในแง่ของความเที่ยงตรงเฉลี่ย (mean average precision) และ ค่าจุดตัดเหนือสหภาพ (intersection over union) ซึ่งทำงานได้เร็วกว่าวิธีการตรวจจับอื่น ๆ อย่างเห็นได้ชัดจากการเปรียบเทียบประสิทธิภาพที่เท่ากัน ดังรูปที่ 39



รูปที่ 39 การเปรียบเทียบประสิทธิภาพของ YOLOv3 ซึ่งทำงานเร็วกว่าวิธีการตรวจจับแบบอื่น ¹

จากการเปรียบเทียบประสิทธิภาพของทั้ง 2 แบบจำลอง พบว่า YOLOv3 มีความเหมาะสมในการนำมาใช้กับงานวิจัยนี้ในส่วนของ การตรวจจับสิ่งกีดขวางประเภทวัตถุที่สุด

¹ ที่มา : <https://jonathan-hui.medium.com/object-detection-speed-and-accuracy-comparison-faster-r-cnn-r-fcn-ssd-and-yolo-5425656ae359>

2.12 การตรวจจับขอบภาพ (Edge Detection)

การตรวจจับขอบภาพเป็นการหาเส้นรอบวัตถุที่อยู่ในภาพ [5] เมื่อทราบเส้นรอบวัตถุจะสามารถคำนวณหาพื้นที่ ขนาด หรือ รู้จำนวนของวัตถุนั้นได้ อย่างไรก็ตาม การหาขอบภาพที่ถูกต้องสมบูรณ์นั้นต้องอาศัยความแตกต่างของความเข้มของแสงจากจุดหนึ่งไปยังอีกจุดหนึ่ง คือ ความแตกต่างระหว่างพื้นหน้ากับพื้นหลัง หากความต่างนี้มีค่ามากจะทำให้เห็นขอบภาพได้อย่างชัดเจน แต่ถ้าความต่างนี้มีค่าน้อยจะทำให้เห็นขอบภาพได้ไม่ชัดเจน

วิธีการหาขอบภาพนั้นมีด้วยกันหลายวิธี ซึ่งสามารถแบ่งได้เป็น 2 กลุ่มหลัก คือ ลาปลาเซียนเมทรูด (Laplacian method) และ เกรเดียนตเมทรูด (Gradient method) โดยในแต่ละวิธีมีรายละเอียดดังต่อไปนี้

2.12.1 ลาปลาเซียนเมทรูด (Laplacian method)

เป็นการหาขอบภาพโดยใช้ออนุพันธ์อันดับ 2 โดยใช้จุดที่ค่า y เป็น 0 (Zerocrossing) ซึ่งวิธีนี้จะใช้เวลาในการคำนวณมากกว่าเกรเดียนตเมทรูด ตัวอย่างวิธีการหาขอบภาพของกลุ่มนี้ เช่น Laplacian of Gaussian และ Marrs-Hildreth เป็นต้น

2.12.2 เกรเดียนตเมทรูด (Gradient method)

วิธีนี้จะหาขอบภาพโดยการหาจุดต่ำสุดและจุดสูงสุดในรูปของอนุพันธ์อันดับหนึ่งของภาพ โดยจุดที่เป็นขอบจะอยู่ในส่วนที่มีค่ามากกว่าจุดเริ่มเปลี่ยน (threshold) จึงทำให้เส้นขอบที่ได้มีลักษณะหนา ตัวอย่างวิธีการหาขอบภาพของกลุ่มนี้ เช่น Roberts, Prewitt, Sobel และ Canny เป็นต้น

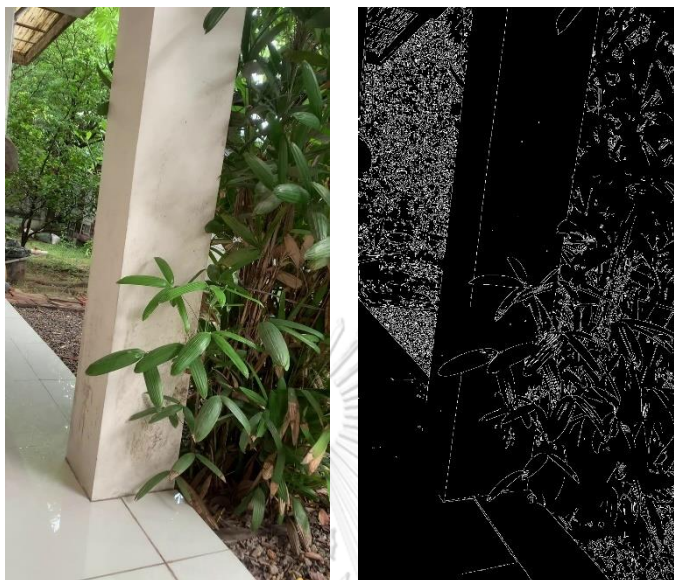
2.12.3 อัลกอริทึมตรวจจับขอบภาพแคณี่ (Canny Edge Detection Algorithm)

Canny จะทำการตรวจจับขอบภาพโดยอาศัยหลักการของเกรเดียนตภายในภาพ ซึ่งใช้การหาอนุพันธ์ของตัวกรองเกาส์เซียน (Gaussian filter) โดยมีการกำหนดใช้ threshold 2 ค่าในการตรวจจับขอบภาพคมชัดมาก และ ขอบภาพคมชัดน้อยที่เชื่อมต่อกับขอบภาพคมชัดมาก [6]

Canny สามารถลดสัญญาณรบกวนภาพได้เป็นอย่างดี และ ยังสามารถตรวจจับขอบภาพที่มีความคมชัดน้อยได้เป็นอย่างดี อีกทั้ง Canny ยังสามารถระบุ threshold เพื่อความไวของอัลกอริทึม โดยหากกำหนดให้ threshold มีค่าสูง ภาพที่ได้จะไม่ตรวจจับขอบภาพที่ไม่คมชัด

การทำงานของ การตรวจจับขอบภาพ Canny เริ่มต้นจากการปรับภาพให้เรียบ (smoothing) ด้วยตัวกรองเกาส์เซียน เพื่อกำจัดสัญญาณรบกวน หลังจากนั้นคำนวณค่าขนาด

(magnitude) และ ทิศทาง (orientation) ของเกรเดียนต์ โดยใช้การหาอนุพันธ์อันดับหนึ่ง ถัดมาใช้ non-maxima suppression กับ gradient magnitude เพื่อทำให้ได้ขอบที่บางลง และ ขั้นตอนสุดท้ายใช้ double thresholding algorithm เพื่อระบุพิกเซลที่เป็นขอบและช่วยเชื่อมต่อขอบภาพ



รูปที่ 40 การตรวจจับขอบภาพด้วยอัลกอริทึม Canny

สำหรับในงานวิจัยนี้จะใช้การตรวจจับขอบภาพด้วยอัลกอริทึม Canny นำมาใช้ในการตรวจจับสิ่งกีดขวางประเภทที่ไม่ใช่วัตถุ เช่น เสา กำแพง หรือ ประตู เป็นต้น โดยจะทำการหาเส้นตรงที่ยาวที่สุดในภาพมาคำนวณ

บทที่ 3 งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์นี้ เนื้อหาหลักจะเกี่ยวกับอุปกรณ์ช่วยเหลือผู้พิการ ซึ่งจะเน้นไปในเรื่องของการตรวจจับสิ่งกีดขวาง เพื่ออำนวยความสะดวกและลดโอกาสเกิดอุบัติเหตุให้กับผู้ใช้งาน โดยในแต่ละงานวิจัยนั้นมีความแตกต่างกันตรงการเลือกใช้อัลกอริทึมและเครื่องมือที่นำมาใช้ในการตรวจจับสิ่งกีดขวาง บางงานวิจัยใช้อุปกรณ์ฮาร์ดแวร์ในการตรวจจับ บางงานวิจัยใช้วิธีการทางคอมพิวเตอร์วิทัศน์ในการตรวจจับ ซึ่งงานวิจัยที่ได้ยกมานั้นแสดงให้เห็นถึงความหลากหลายทางแนวคิดและวิธีการสำหรับนำมาทำการตรวจจับสิ่งกีดขวาง เพื่อจะได้นำเอาวิธีการเหล่านั้นมาประยุกต์ใช้ ทำให้ระบบสามารถใช้งานได้กับสภาพแวดล้อมที่หลากหลาย โดยให้มีข้อจำกัดในด้านการใช้งานน้อยที่สุด และ ทำให้ระบบมีประสิทธิภาพมากที่สุด เพื่อก่อให้เกิดประโยชน์สูงสุดกับผู้ใช้งาน

อุปกรณ์ช่วยเหลือผู้พิการยังคงถูกนำมาพัฒนาและประยุกต์ใช้ร่วมกับเทคโนโลยีที่มีความก้าวหน้าขึ้นในแต่ละยุคสมัยอย่างต่อเนื่อง เพื่อเพิ่มประสิทธิภาพการใช้งาน อำนวยความสะดวกให้แก่ผู้ใช้งาน และ ลดปัญหาหรือข้อบกพร่องอันเกิดจากการใช้งานต่าง ๆ โดยในแต่ละอุปกรณ์ถูกนำมาพัฒนาด้วยเทคโนโลยีที่แตกต่างกันไป เพื่อให้สอดคล้องกับปัญหาที่พบในแต่ละการใช้งาน

งานวิจัย [7] เกี่ยวข้องกับเก้าอี้รถเข็นไฟฟ้าที่มีระบบเซนเซอร์ถอยหลังอัตโนมัติ ผู้วิจัยได้นำเอาอุปกรณ์ฮาร์ดแวร์สำเร็จรูปมาใช้ ได้แก่ ไมโครคอนโทรลเลอร์ (microcontroller) เซนเซอร์วัดระยะอัลตราโซนิก (ultrasonic sensor) และ อุปกรณ์ไฟฟ้าสำหรับส่งเสียงสัญญาณเตือน (buzzer) โดยได้มีการกำหนดช่วงของระยะไว้ว่า ถ้าอยู่ในช่วงที่กำหนดให้ส่งเสียงเตือนออกมา ซึ่งอุปกรณ์เหล่านี้สามารถนำไปใช้ประมาณระยะห่างระหว่างวิลแชร์กับสิ่งกีดขวางได้อย่างมีประสิทธิภาพและใช้งบประมาณต่ำ



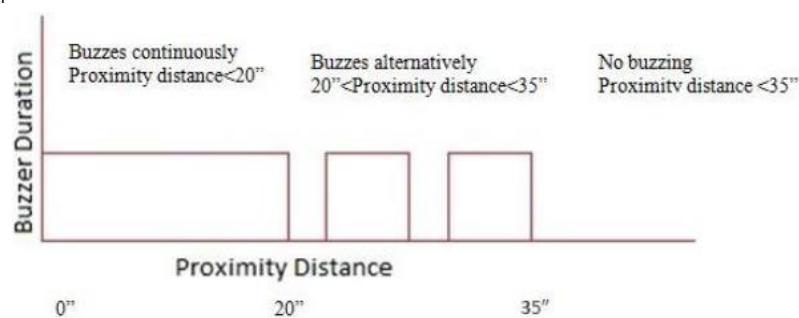
microcontroller

ultrasonic sensor

buzzer

รูปที่ 41 ตัวอย่างอุปกรณ์ฮาร์ดแวร์

งานวิจัยของเขาอาจจะไม่สามารถนำมาปรับใช้กับงานวิจัยนี้ได้โดยตรง เนื่องจากผู้วิจัยเน้นไปทางใช้อุปกรณ์ฮาร์ดแวร์เป็นหลัก แต่ได้มีการนำเอาแนวคิดในส่วนของกาหนดช่วงของการแจ้งเตือนมาประยุกต์ใช้กับงานวิจัยนี้ได้



รูปที่ 42 ช่วงของระยะในการส่งเสียงเตือน

งานวิจัยต่อมา [8] ผู้วิจัยใช้เซ็นเซอร์วัดระยะอัลตราโซนิกร่วมกับอุปกรณ์ช่วยเดิน (walker) สำหรับผู้ที่มีความบกพร่องทางการมองเห็นและการเคลื่อนไหว โดยงานวิจัยของเขาได้ศึกษาเพิ่มเติมพบว่า การตรวจจับสิ่งกีดขวางด้วยกล้องมีความแม่นยำมากกว่าการใช้เซ็นเซอร์วัดระยะอัลตราโซนิก แต่หากใช้งานในสภาพแวดล้อมที่มีแสงน้อยหรือในเวลากลางคืน ประสิทธิภาพในการมองเห็นสิ่งกีดขวางจะลดลง

จากการศึกษาประสิทธิภาพความแม่นยำของอุปกรณ์ที่จะนำมาใช้ตรวจจับสิ่งกีดขวางในงานวิจัยของเขา ถือเป็นอีกหนึ่งตัวช่วยที่สำคัญในงานวิจัยนี้เลือกใช้กล้องสมาร์ทโฟนในการนำมาใช้เป็นอุปกรณ์ในการตรวจจับสิ่งกีดขวาง

งานวิจัย [9] ได้พัฒนารองเท้าที่มีเซ็นเซอร์และแอกชูเอเตอร์ (sensors and actuators) ฝังอยู่ สำหรับนำทางให้ผู้พิการทางสายตาสามารถเดินทางได้สะดวกมากขึ้น ช่วยตรวจจับวัตถุต่าง ๆ โดยรอบได้อย่างมีประสิทธิภาพ และ มีการแจ้งเตือนผู้ใช้งานอย่างเหมาะสมผ่านสมาร์ทโฟน อีกทั้งงานวิจัยของเขาได้ทำการเปรียบเทียบประสิทธิภาพของอัลกอริทึมที่ใช้ในการตรวจจับวัตถุระหว่าง YOLOv3 กับอีก 5 โมเดล ได้แก่ Faster R-CNN, Fast YOLO, YOLOv1, YOLOv2 และ SSD500 พบว่า YOLOv3 มีประสิทธิภาพดีที่สุด ถึงแม้ YOLOv3 จะมีประสิทธิภาพดีที่สุด

จากการเปรียบเทียบประสิทธิภาพของอัลกอริทึมในงานวิจัยของเขา ช่วยให้ตัดสินใจเลือกใช้ อัลกอริทึม YOLOv3 มาใช้ในส่วนของการตรวจจับสิ่งกีดขวางที่เป็นประเภทวัตถุสำหรับงานวิจัยนี้

| Model Name | mAP |
|--------------|------|
| Faster R-CNN | 73.2 |
| Fast YOLO | 52.7 |
| YOLOv1 | 66.4 |

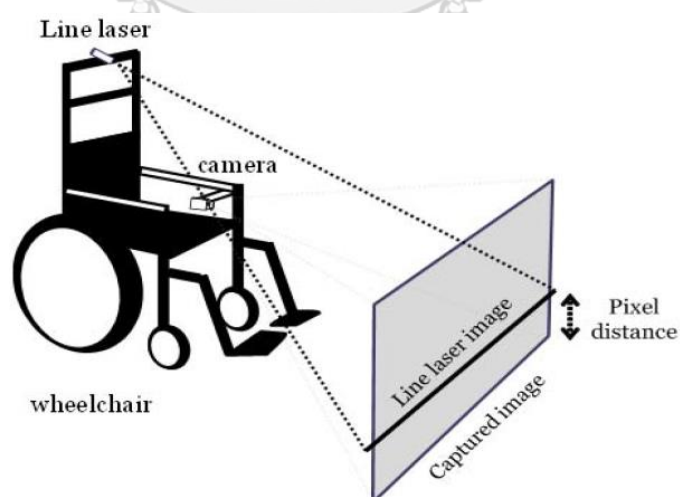
| Model Name | mAP |
|------------|------|
| YOLOv2 | 78.6 |
| YOLOv3 | 82.9 |
| SSD500 | 76.8 |

ตารางที่ 2 เปรียบเทียบประสิทธิภาพของอัลกอริทึมตรวจจับวัตถุ

งานวิจัย [10] ได้พัฒนาระบบหลีกเลี่ยงสิ่งกีดขวางแบบเรียลไทม์ (real time) สำหรับเก้าอี้รถเข็นไฟฟ้า โดยติดกล้องสามมิติ (stereo camera) ไว้ด้านหน้าเก้าอี้รถเข็น และ ใช้วิธีการปรับเทียบกล้อง (camera calibration) จากมุมมองแบบสามมิติให้กลายเป็นมุมมองแบบรูเข็ม (pinhole image model) เพื่อคำนวณค่าความต่างโดยใช้วิธีสหสัมพันธ์ผลรวมของผลต่างสัมบูรณ์ (the Sum of Absolute Differences (SAD) correlation method) สำหรับการตรวจจับจุดสังเกต (landmarks) นอกจากนี้ยังใช้เพื่อประมาณระยะทางการเคลื่อนที่ ซึ่งวิธีการนี้มักนำไปใช้ในการนำทางและการทำแผนที่ของหุ่นยนต์ โดยผลการทดลองพบว่าสามารถนำไปใช้กับสภาพแวดล้อมจริงได้

ได้นำเอาหลักการคำนวณผลต่างสัมบูรณ์ (SAD) ในงานวิจัยของเขา มาประยุกต์ใช้ในส่วนของการคำนวณเวลาก่อนชนสำหรับงานวิจัยนี้

งานวิจัย [11] ได้พัฒนาเก้าอี้รถเข็นอัจฉริยะเช่นกัน แต่มีระบบตรวจจับสิ่งกีดขวางและอัลกอริทึมตรวจจับมนุษย์ ด้วยวิธีการพื้นฐานของคอมพิวเตอร์วิทัศน์ โดยมีการติดตั้งกล้องและเลเซอร์ (laser) ไว้ด้านหน้าเก้าอี้รถเข็น เพื่อใช้ประมาณระยะทางและตำแหน่งของสิ่งกีดขวาง



รูปที่ 43 ติดตั้งกล้องและเลเซอร์ด้านหน้าเก้าอี้รถเข็น

งานวิจัยนี้ได้มีการนำเอาแนวคิดในส่วนของการใช้เส้นมาใช้เป็นตัวกำหนดระยะห่างของสิ่งกีดขวางที่จะต้องทำการแจ้งเตือน

งานวิจัย [12] ได้นำเสนอระบบป้องกันการชนกัน เพื่อการนำทางอัตโนมัติ โดยใช้กล้องธรรมดาเป็นเซ็นเซอร์หลัก และ ใช้วิธีการของออปติคัลโฟลว์ (optical flow) ในการตรวจจับสิ่งกีดขวาง ซึ่งดูจากกลุ่มพิกเซลที่มีสีใกล้เคียงกันของวัตถุเคลื่อนที่ไปในทิศทางเดียวกัน

ในงานวิจัยนี้ได้มีการนำเอาหลักการของ optical flow มาปรับใช้ในส่วนของการหาทิศทางเคลื่อนที่ด้วยเช่นกัน

จากงานวิจัยในข้างต้นอาจกล่าวได้ว่า ผู้วิจัยส่วนใหญ่เล็งเห็นถึงปัญหาไปในทิศทางเดียวกัน นั่นคืออุปสรรคอันเกิดจากการมองไม่เห็นสิ่งกีดขวาง ส่งผลให้อุปกรณ์ช่วยเหลือเหล่านั้นไม่สามารถผ่านไปได้ หรือ อาจส่งผลให้เกิดอุบัติเหตุได้ จึงได้มีงานวิจัยที่มีเนื้อหาเกี่ยวกับการตรวจจับหรือหลบหลีกสิ่งกีดขวางอยู่มากพอสมควร แต่งานวิจัยส่วนใหญ่จะเน้นเป็นการตรวจจับสิ่งกีดขวางที่อยู่ทางด้านหน้ามากกว่าด้านหลัง ซึ่งในความเป็นจริงแล้ว การใช้งานเก้าอี้รถเข็นผู้ใช้งานจะมองไปด้านหน้าได้สะดวกกว่าการหันไปมองด้านหลัง ซึ่งทำให้งานวิจัยนี้เน้นไปที่การตรวจจับสิ่งกีดขวางที่อยู่ด้านหลัง แต่ด้วยวิธีการที่ในงานวิจัยนี้ใช้สามารถนำไปใช้ตรวจจับสิ่งกีดขวางได้ทั้งด้านหน้าและด้านหลัง อีกทั้งยังสามารถนำไปประยุกต์ใช้ร่วมกับหลากหลายอุปกรณ์ช่วยเหลือ เพื่อลดข้อจำกัดให้กับผู้ใช้งานให้สามารถใช้งานได้สะดวกมากยิ่งขึ้น

บทที่ 4 แนวคิดและวิธีการวิจัย

งานวิจัยนี้เป็นการพัฒนาระบบตรวจจับสิ่งกีดขวางสำหรับเก้าอี้รถเข็นไฟฟ้า จึงมีแนวคิดที่จะทำการแจ้งเตือนผู้ใช้งานให้ทราบว่ามีสิ่งกีดขวางอยู่ด้านหลัง โดยใช้กล้องสมาร์ทโฟนเป็นอุปกรณ์ในการตรวจจับสิ่งกีดขวาง ในส่วนของการตรวจจับสิ่งกีดขวางนั้นนำอัลกอริทึมตรวจจับวัตถุ (object detection algorithm) มาใช้ แต่เนื่องจากการทดลองเบื้องต้นพบว่า หากการใช้แค่อัลกอริทึมตรวจจับวัตถุเพียงอย่างเดียว จะพบว่าระบบไม่สามารถตรวจจับสิ่งกีดขวางประเภทที่ไม่ใช้วัตถุ เช่น เส้า กำแพง หรือ ประตู เป็นต้นได้ จึงได้มีการนำเอาอัลกอริทึมตรวจจับขอบภาพ (edge detection algorithm) มาใช้ร่วมด้วย เพื่อทำการหาขอบของสิ่งกีดขวางนั้น แต่ยังมีแนวคิดว่า ถ้าหากระบบทำการตรวจจับสิ่งกีดขวางได้เพียงอย่างเดียวก็อาจจะยังอำนวยความสะดวกให้กับผู้ใช้งานได้ไม่มากเท่าที่ควร จึงได้เพิ่มในส่วนของการคำนวณเวลาก่อนชนเข้าไปในระบบด้วย เพื่อให้ผู้ใช้งานสามารถทราบได้ว่าถ้าหากชนเคลื่อนเก้าอี้รถเข็นด้วยความเร็วเท่าเดิมจะเหลือเวลาอีกเท่าใดที่จะไปชนสิ่งกีดขวางนั้น

4.1 อุปกรณ์ที่ใช้ในงานวิจัย

เนื่องจากเป็นระบบตรวจจับสิ่งกีดขวาง ดังนั้นจึงต้องมีอุปกรณ์ที่นำมาใช้ในการตรวจจับ ซึ่งในงานวิจัยนี้ได้เลือกใช้กล้องสมาร์ทโฟนมาเป็นอุปกรณ์ในการตรวจจับ เพราะปัจจุบัน [13] เทคโนโลยีกล้องสมาร์ทโฟนนั้นให้ความละเอียดภาพมากพอที่จะนำมาใช้กับงานด้านคอมพิวเตอร์วิทัศน์ได้

สำหรับอุปกรณ์ที่เลือกมาใช้เป็นกล้อง คือ ไอโฟน เอ็กซ์เอส แมกซ์ (iPhone XS Max) ซึ่งมีคุณสมบัติของกล้อง ดังนี้

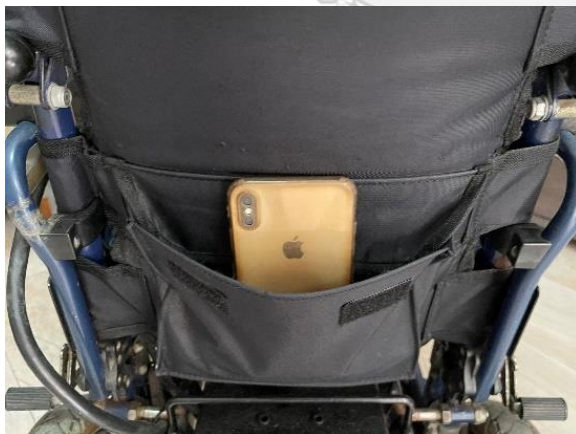
- กล้องคู่ด้านหลังมีความละเอียดอยู่ที่ตัวละ 12 ล้านพิกเซล
- มุมกว้าง: รัรับแสงขนาด $f/1.8$
- ซุมออปติคัล 2 เท่า และ ซุมดิจิทัลได้สูงสุด 10 เท่า
- ระบบป้องกันภาพสั่นไหวคู่แบบออปติคัล
- ชุดเลนส์ 6 ชั้น
- เซ็นเซอร์รับแสงด้วยส่วนหลัง
- อดโต้โฟกัสพร้อมโฟกัสพิกเซล
- การควบคุมค่าแสง
- ระบบป้องกันภาพสั่นไหวอัตโนมัติ
- รูปแบบไฟล์ภาพที่บันทึก: HEIF และ JPEG

- บันทึกวิดีโอระดับ HD 1080p ที่ 30 เฟรมต่อวินาที (FPS)



รูปที่ 44 กล้อง iPhone XS Max

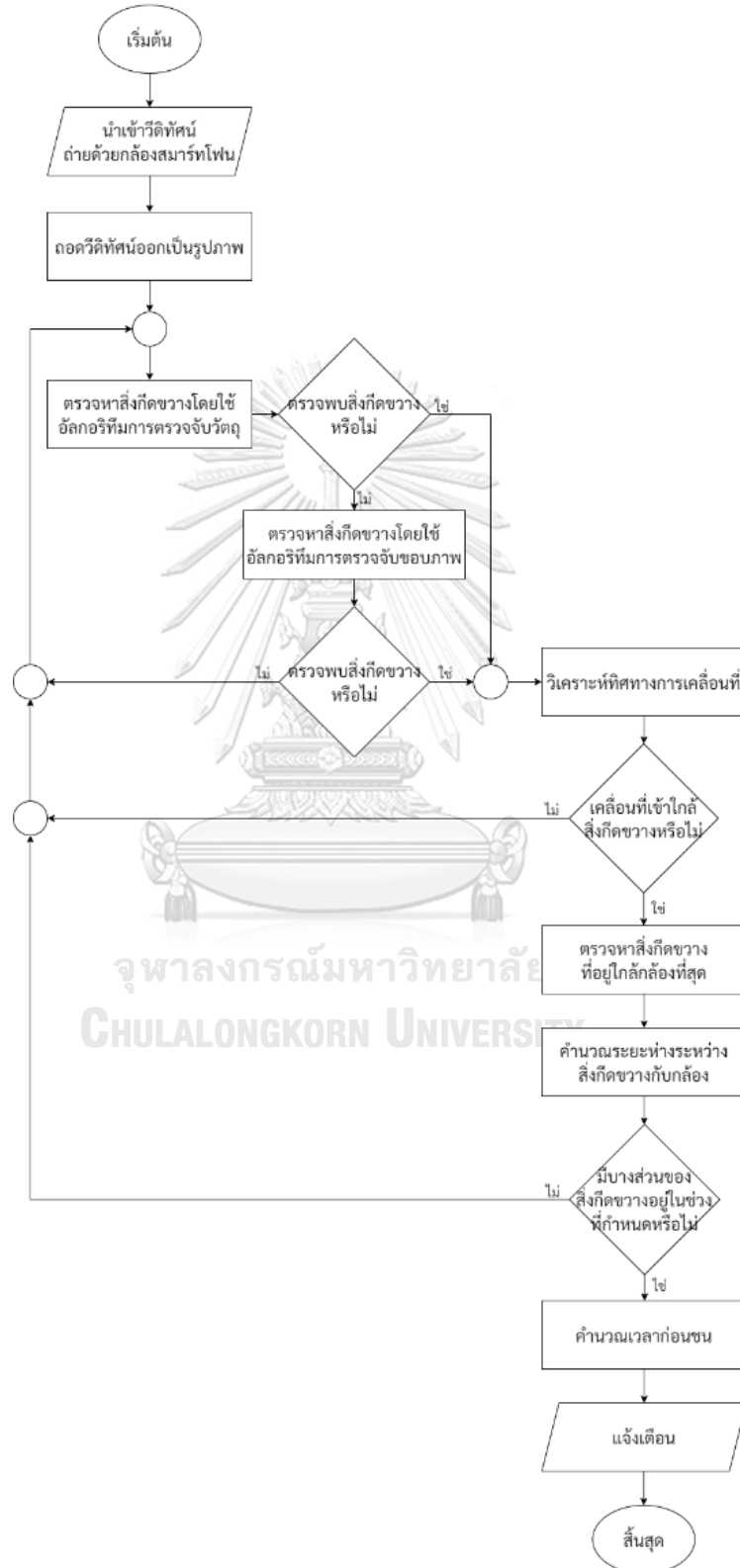
จากนั้นนำกล้องสมาร์ทโฟนไปติดไว้ที่ด้านหลังเก้าอี้รถเข็นไฟฟ้า ดังรูปที่ 45



รูปที่ 45 ติดกล้องสมาร์ทโฟนไว้ด้านหลังเก้าอี้รถเข็นไฟฟ้า

4.2 กระบวนการทำงานของระบบ

งานวิจัยนี้ได้พัฒนาระบบตรวจจับสิ่งกีดขวาง ซึ่งมีกระบวนการทำงานดังนี้



รูปที่ 46 ภาพรวมการทำงานของระบบ

นำเข้าข้อมูลวีดีทัศน์ ถอดวีดีทัศน์ออกมาเป็นรูปภาพทีละรูป จากนั้นทำการตรวจจับวัตถุที่อยู่ในรูปนั้น ๆ ซึ่งระบบจะเริ่มใช้อัลกอริทึมตรวจจับวัตถุก่อน หากตรวจจับแล้วไม่พบวัตถุใด ๆ ระบบจะเปลี่ยนไปใช้อัลกอริทึมตรวจจับขอบภาพแทน จากนั้นระบบจะทำการวิเคราะห์ทิศทางเคลื่อนที่ โดยตรวจสอบจากขนาดของวัตถุ หากวัตถุชนิดเดียวกันในรูปถัดไปมีขนาดใหญ่ขึ้น แสดงว่าเก้าอี้รถเข็นกำลังเคลื่อนที่เข้าไปใกล้วัตถุมากขึ้น ต่อมาระบบจะทำการคำนวณหาระยะห่างระหว่างเก้าอี้รถเข็นไปยังแต่ละสิ่งกีดขวางที่ตรวจพบได้จากในรูปนั้น ๆ แล้วเลือกสิ่งกีดขวางที่อยู่ใกล้ที่สุด เมื่อได้สิ่งกีดขวางที่ใกล้ที่สุดแล้ว ระบบจะทำการตรวจสอบว่า ถ้าหากมีบางส่วนของสิ่งกีดขวางอยู่ในช่วงที่กำหนด ระบบจะแสดงข้อความแจ้งเตือน พร้อมกับคำนวณเวลาก่อนชนออกไปให้ผู้ใช้งานทราบ

4.3 การตรวจจับสิ่งกีดขวาง

การตรวจจับสิ่งกีดขวางถือเป็นส่วนที่สำคัญที่สุดในงานวิจัยนี้ ในเบื้องต้นผู้วิจัยมีความคิดที่จะใช้อัลกอริทึมตรวจจับวัตถุเพียงอย่างเดียว แต่เมื่อได้ทดสอบแล้วพบว่า อัลกอริทึมตรวจจับวัตถุเพียงอย่างเดียวไม่สามารถตรวจจับสิ่งกีดขวางได้ทุกประเภท ทางผู้วิจัยจึงได้มีแนวคิดที่จะนำเอาอัลกอริทึมที่ใช้สำหรับตรวจจับขอบภาพมาประยุกต์ใช้ร่วมด้วย เพื่อให้ระบบมีประสิทธิภาพมากพอที่จะสามารถนำไปประยุกต์ใช้ได้จริงกับเก้าอี้รถเข็นไฟฟ้า เพราะฉะนั้นในส่วนของการตรวจจับสิ่งกีดขวาง จะแบ่งออกเป็น 2 ระบบย่อยดังนี้

4.3.1 การตรวจจับสิ่งกีดขวางประเภทวัตถุ

ในงานวิจัยนี้จะใช้อัลกอริทึม YOLOV3 [14] ในการตรวจจับสิ่งกีดขวางเป็นหลัก เพราะมีความแม่นยำในการตรวจจับวัตถุค่อนข้างสูง อีกทั้งยังสามารถตรวจจับวัตถุที่ทับซ้อนกันได้ หากอัลกอริทึมนี้สามารถตรวจจับสิ่งกีดขวางได้ ระบบจะทำการหาสิ่งกีดขวางที่อยู่ใกล้กับเก้าอี้รถเข็นไฟฟ้ามากที่สุด (วัตถุที่อยู่ในกรอบสีเขียว ดังรูปที่ 47) เพื่อจะนำไปคำนวณเวลาก่อนชน และ ทำการแจ้งเตือนต่อไป

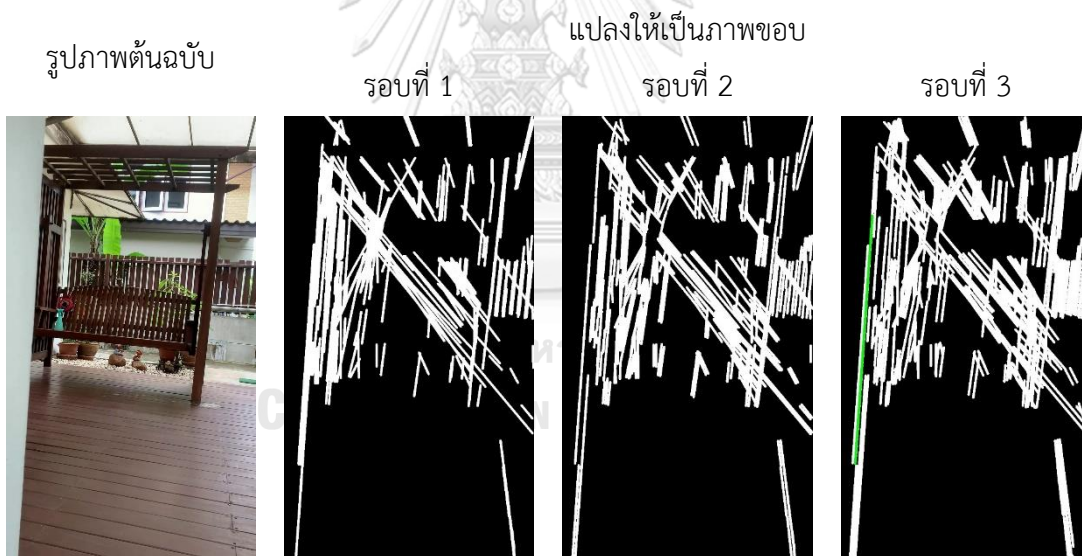


รูปที่ 47 ตรวจจับสิ่งกีดขวางประเภทวัตถุโดยใช้อัลกอริทึม YOLOV3

แม้ว่าอัลกอริทึม YOLOv3 จะมีประสิทธิภาพในการตรวจจับค่อนข้างสูง แต่กลับพบข้อจำกัดคือ อัลกอริทึมนี้ไม่สามารถตรวจจับสิ่งกีดขวางได้ทุกประเภท ดังนั้นผู้วิจัยจึงมีความคิดว่า ถ้าหากจะนำอัลกอริทึมนี้ไปใช้เพียงอย่างเดียวคงจะไม่เพียงพอ ในกรณีที่ใช้งานจริงกับแก้อักรเซ็นไฟฟ้า แล้วไปเจอสิ่งกีดขวางที่มีขนาดใหญ่ อาจจะทำให้ระบบไม่สามารถแจ้งเตือนผู้ใช้งานได้ ส่งผลให้ประสิทธิภาพในการตรวจจับสิ่งกีดขวางของระบบอาจจะไม่ดีเท่าที่ควร เพราะฉะนั้นผู้วิจัยจึงได้พัฒนาในส่วนของการตรวจจับสิ่งกีดขวางประเภทที่ไม่ใช่วัตถุเพิ่มขึ้นมา เพื่อนำมาใช้งานร่วมกัน

4.3.2 การตรวจจับสิ่งกีดขวางประเภทที่ไม่ใช่วัตถุ

การตรวจจับสิ่งกีดขวางประเภทที่ไม่ใช่วัตถุ ในที่นี้หมายถึง สิ่งกีดขวางที่มีขนาดใหญ่ หรือ สิ่งกีดขวางที่ไม่สามารถมองเห็นได้หมดในเฟรมเดียว เช่น เสา กำแพง หรือ ประตู เป็นต้น ในส่วนนี้จะใช้อัลกอริทึมในการตรวจจับขอบภาพ คือ Canny ซึ่งจะทำการแปลงจากภาพต้นฉบับให้เป็นภาพขอบ (edge image) โดยระบบจะทำการแปลง 3 รอบ เพื่อให้ได้เส้นขอบที่ชัดเจนขึ้น



รูปที่ 48 แปลงรูปภาพต้นฉบับให้เป็นภาพขอบ 3 รอบ

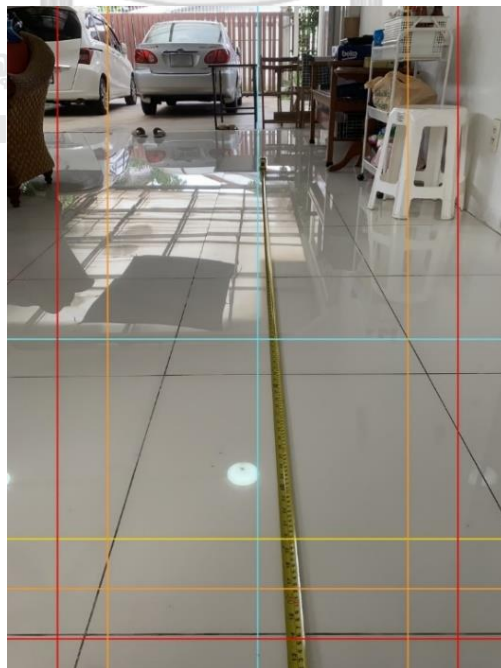
จากนั้นจะทำการหาเส้นที่ยาวที่สุดจากภาพขอบ หากตรวจพบว่าเส้นขอบนั้นอยู่ในช่วงที่กำหนด ระบบจะถือว่ามีส่วนของสิ่งกีดขวางอยู่ในช่วงที่กำหนด จึงจะทำการแจ้งเตือนต่อไป



รูปที่ 49 ตรวจจับขอบของสิ่งกีดขวางโดยใช้อัลกอริทึม Canny

4.4 การกำหนดระยะห่าง

ในการคำนวณหาระยะห่างระหว่างเก้าอี้รถเข็นกับสิ่งกีดขวาง เริ่มจากกำหนดระยะห่างที่ต้องการให้ระบบทำการแจ้งเตือนก่อนที่จะชน โดยในงานวิจัยนี้จะเป็นการวัดระยะจริงจากเก้าอี้รถเข็น



รูปที่ 50 กำหนดระยะที่ต้องการให้ระบบแจ้งเตือน

จากรูปที่ 50 จะเห็นว่า มีการใช้เส้นแนวตั้งและแนวนอน และ มีการกำหนดระยะที่จะใช้เป็นเกณฑ์ในการแจ้งเตือนไว้ตามสีเส้น โดยมีรายละเอียดดังนี้

- **เส้นแนวตั้ง** ใช้กับสิ่งกีดขวางประเภทที่ไม่ใช้วัตถุที่อัลกอริทึมตรวจจับขอบภาพสามารถตรวจจับได้ ในกรณีที่ขอบของสิ่งกีดขวางอยู่ในบริเวณฝั่งซ้ายของรูป ระบบจะใช้เส้นแนวตั้งที่อยู่ทางฝั่งซ้ายเป็นเกณฑ์ในการแจ้งเตือน ในทำนองเดียวกัน หากขอบของสิ่งกีดขวางอยู่ในบริเวณฝั่งขวาของรูป ระบบก็จะใช้เส้นแนวตั้งที่อยู่ทางฝั่งขวาเป็นเกณฑ์ในการแจ้งเตือนแทน
- **เส้นแนวนอน** ใช้กับสิ่งกีดขวางประเภทวัตถุที่อัลกอริทึมตรวจจับวัตถุสามารถตรวจจับได้
- **เส้นสีฟ้า** ใช้คัดกรองสิ่งกีดขวางที่สนใจ ซึ่งระบบจะสนใจเฉพาะสิ่งกีดขวางที่อยู่ต่ำกว่าเส้นสีฟ้าลงมา
- **เส้นสีเหลือง** ใช้ตรวจสอบว่าสิ่งกีดขวางอยู่ต่ำกว่าเส้นสีเหลืองหรือไม่ หากต่ำกว่าระบบจะเริ่มทำการคำนวณเวลาที่เหลือก่อนที่จะไปชนสิ่งกีดขวางนั้น และ แสดงข้อความแจ้งเตือน
- **เส้นสีส้ม** ใช้เป็นเกณฑ์ในการแจ้งเตือนก่อนที่จะไปชนกับสิ่งกีดขวางนั้น
- **เส้นสีแดง** ใช้เป็นเกณฑ์ในการแจ้งเตือนว่ามีโอกาสสูงมากที่จะไปชนกับสิ่งกีดขวางนั้น แล้วอาจจะเกิดอันตรายได้

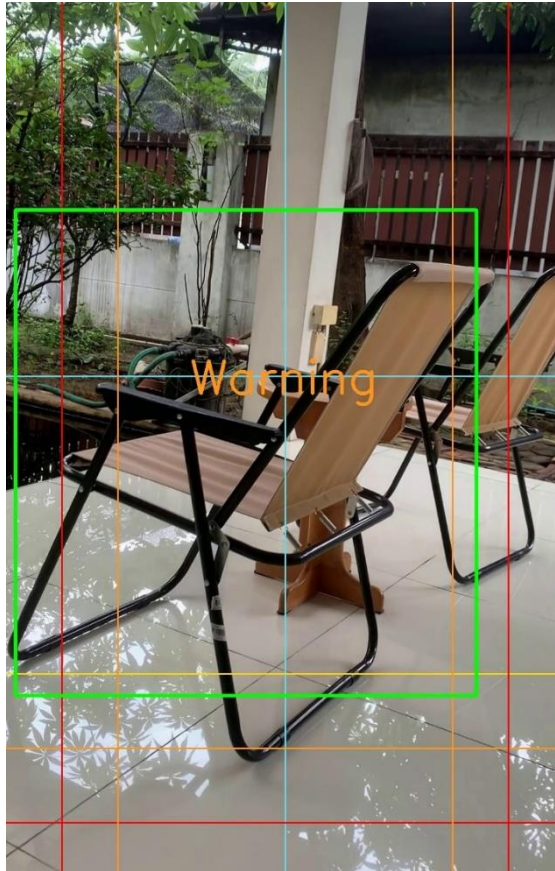
เมื่อกำหนดระยะให้กับแต่ละเส้นแล้ว ระบบจะสนใจเฉพาะสิ่งกีดขวางที่อยู่ใกล้ที่สุดเท่านั้น แล้วตรวจสอบว่า ถ้ามีบางส่วนของสิ่งกีดขวางอยู่ในช่วงที่กำหนด ระบบก็จะทำการแจ้งเตือนออกไป

4.5 การแจ้งเตือนการชน

หลังจากที่ระบบสามารถตรวจจับสิ่งกีดขวางได้แล้วนั้น การแจ้งเตือนจึงเป็นอีกส่วนที่สำคัญ โดยเฉพาะอย่างยิ่งการทำให้ระบบสามารถแจ้งเตือนได้ล่วงหน้า จะช่วยให้ผู้ใช้งานสามารถทราบได้ก่อน เพื่อลดโอกาสเกิดอุบัติเหตุ หรือ ช่วยให้ผู้ใช้งานสามารถหลบหลีกสิ่งกีดขวางได้ทันเวลา

สำหรับการแจ้งเตือนจะแสดงออกมาในรูปแบบของข้อความแจ้งเตือน โดยแบ่งเป็น 2 คำ ได้แก่ Warning (เตือน) และ Collision (ชน)

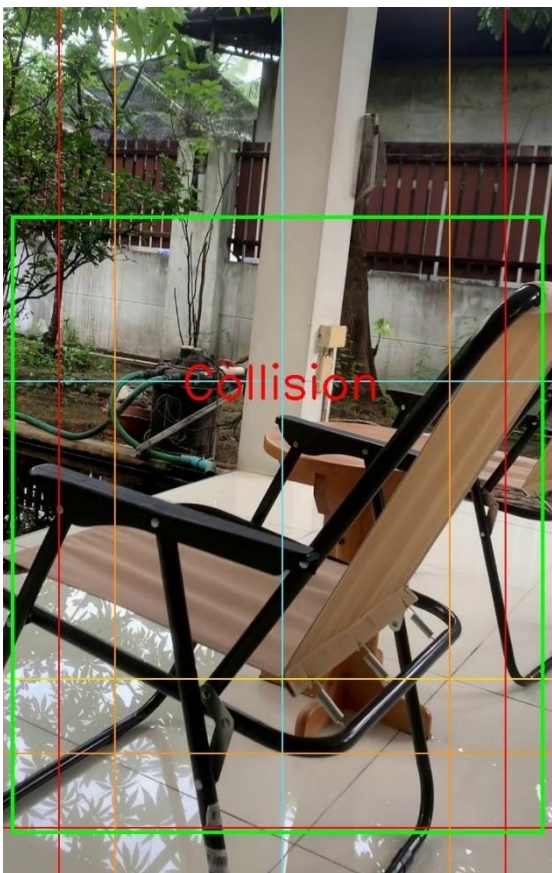
4.5.1 ข้อความแจ้งเตือน Warning



รูปที่ 51 ระบบแสดงข้อความแจ้งเตือนก่อนชน ด้วยคำว่า Warning

หากระบบปรากฏข้อความแจ้งเตือน คำว่า Warning หมายถึง ระบบตรวจจับได้ว่ามีบางส่วน
ของสิ่งกีดขวางอยู่ในช่วงตั้งแต่เส้นสีเหลืองจนถึงเส้นสีแดง ให้ผู้ใช้งานลดความเร็วในการขับเคลื่อน
หรือ ยังพอมีเวลาให้ผู้ใช้งานสามารถหลบหลีกสิ่งกีดขวางได้ทันก่อนที่จะชน

4.5.2 ข้อความแจ้งเตือน Collision



รูปที่ 52 ระบบแสดงข้อความแจ้งเตือนก่อนชน ด้วยคำว่า Collision

หากระบบปรากฏข้อความแจ้งเตือน คำว่า Collision นั้นหมายความว่า แก้อัศจรรย์เซ็นไฟฟ้ามีการเคลื่อนที่เข้าไปใกล้สิ่งกีดขวางนั้นมากขึ้นเรื่อย ๆ และ มีโอกาสสูงมากที่จะเข้าไปชนกับสิ่งกีดขวางนั้น ซึ่งระบบจะทำการตรวจสอบว่ามีบางส่วนของสิ่งกีดขวางอยู่ต่ำกว่าเส้นสีแดงลงมาหรือไม่ หากอยู่ต่ำกว่า ระบบจะเปลี่ยนการแสดงผลข้อความแจ้งเตือนทันที

จะเห็นว่า ข้อความแจ้งเตือนทั้ง 2 รูปแบบนั้น จะใช้สีของข้อความที่แตกต่างกัน เพื่อแสดงให้เห็นถึงระดับของความอันตราย และ ทำให้ผู้ใช้งานเข้าใจความหมายของการแจ้งเตือนได้ง่ายขึ้น

4.6 การคำนวณเวลาก่อนชน

ในส่วนของ การคำนวณเวลาก่อนชน จะช่วยให้ผู้ใช้งานสามารถทราบได้ว่าเหลือเวลาอีกเท่าใดที่จะไปชนสิ่งกีดขวางนั้น โดยระบบจะอาศัยจากการตรวจสอบว่าสิ่งกีดขวางมีการเคลื่อนที่เข้าใกล้กล้องที่ละกี่พิกเซล และ จะนำขนาดของสิ่งกีดขวางนั้นมาหาว่า เหลืออีกกี่พิกเซลจึงจะถึงระยะที่กำหนดไว้ว่าจะชน (ตามเส้นสีแดง) ซึ่งสามารถคำนวณได้ตามสมการที่ 21

$$\text{Gap to Collision} = |a - b| \quad \dots (21)$$

โดยที่

- a คือ พิกัดในแกน y ของด้านล่างของกรอบที่ล้อมรอบสิ่งกีดขวาง ณ รูปปัจจุบัน
- b คือ พิกัดในแกน y ของเส้นสีแดง

จากนั้นคำนวณหาค่าเฉลี่ยในการเคลื่อนที่ในแต่ละพิกเซลว่าใช้เวลาผ่านไปประมาณกี่รูป ซึ่งสามารถคำนวณได้จากสมการที่ 22

$$\text{Average Frame} = \frac{|a - c|}{d} \quad \dots (22)$$

โดยที่

- a คือ พิกัดในแกน y ของด้านล่างของกรอบที่ล้อมรอบสิ่งกีดขวาง ณ รูปปัจจุบัน
- c คือ พิกัดในแกน y ของด้านล่างของกรอบที่ล้อมรอบสิ่งกีดขวาง ณ รูปแรกที่ระบบเริ่มแจ้งเตือน
- d คือ จำนวนรูป ตั้งแต่เริ่มแจ้งเตือนในรูปแรกจนถึงรูปปัจจุบัน

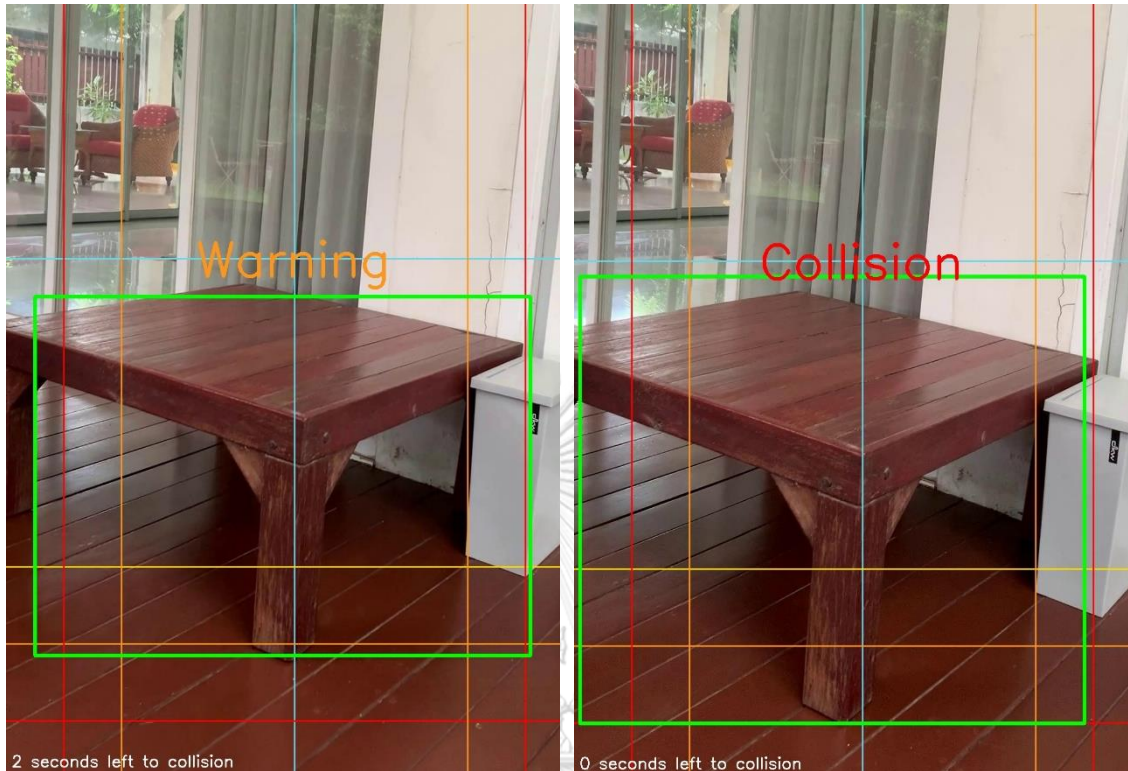
เมื่อทราบค่าผลต่างของพิกเซลที่กำหนดไว้ว่าจะชน (จากสมการที่ 21) และ ค่าเฉลี่ยเฟรมในการขยับต่อ 1 พิกเซล (จากสมการที่ 22) แล้ว สามารถคำนวณเวลาที่เก๊า์รธเซ็นไฟฟ้าจะเคลื่อนที่เข้าไปชนสิ่งกีดขวางได้ตามสมการที่ 23

$$\text{Time to Collision} = \left| \tan \left(\frac{\text{Gap to Collision} \times \text{Average Frame}}{\text{FPS}} \right) \right| \quad \dots (23)$$

โดยที่

- Gap to Collision คือ ค่าที่คำนวณได้จากสมการที่ 21
- Average Frame คือ ค่าที่คำนวณได้จากสมการที่ 22
- FPS คือ อัตราการแสดงผลรูปภาพใน 1 วินาที (frame rate) ของกล้องที่ใช้ตรวจจับสิ่งกีดขวาง

หลังจากที่คำนวณเวลาก่อนชน (ตามสมการที่ 23) แล้ว ระบบจะแสดงเวลาคงเหลือก่อนที่จะไปชนสิ่งกีดขวางนั้นในหน่วยวินาที พร้อมกับ แสดงข้อความแจ้งเตือนการชน ดังรูปที่ 53



รูปที่ 53 แสดงเวลาคงเหลือก่อนชน พร้อมกับ แสดงข้อความแจ้งเตือน

บทที่ 5 การทดลองและผลการทดลอง

งานวิจัยนี้นำเสนอระบบตรวจจับสิ่งกีดขวางสำหรับแก้อัลดเซ็นไฟฟ้า โดยมีกระบวนการตรวจจับสิ่งกีดขวาง เพื่อค้นหาสิ่งกีดขวางที่อยู่ใกล้แก้อัลดเซ็นไฟฟ้ามากที่สุด การกำหนดระยะในการแจ้งเตือนเป็นการกำหนดระยะห่างระหว่างสิ่งกีดขวางกับแก้อัลดเซ็นไฟฟ้า เพื่อให้ระบบสามารถแจ้งเตือนได้ล่วงหน้าก่อนชน หากพบว่าระยะห่างนั้นน้อยกว่าระยะที่กำหนด การคำนวณเวลาก่อนชนเป็นการคาดการณ์ว่าหากผู้ใช้งานยังคงขับเคลื่อนแก้อัลดเซ็นไฟฟ้าด้วยความเร็วเท่าเดิม จะเหลือเวลาอีกเท่าใดจึงจะไปชนกับสิ่งกีดขวางนั้น สำหรับการตรวจจับสิ่งกีดขวางเป็นการประยุกต์ใช้วิธีการประมวลผลภาพดิจิทัล การตรวจจับวัตถุ และการตรวจจับขอบภาพมาใช้ร่วมกัน

โปรแกรมต้นแบบที่ใช้ในการทดลอง พัฒนาขึ้นด้วยโปรแกรมวิซวลสตูดิโอไอโค้ด (Visual Studio Code) เขียนด้วยภาษาโปรแกรมไพทอน (Python) และ ใช้เครื่องคอมพิวเตอร์ Intel(R) Core(TM) i7 Gen 8th ความเร็ว 1.8 GHz หน่วยความจำ 16 GB ในการประมวลผล

5.1 การนำเข้าข้อมูล

ในการทดลองนี้จะนำเข้าวิดีโอจำนวน 20 คลิป ซึ่งถ่ายโดยจำลองการใช้งานจริงกับสภาพแวดล้อมภายในอาคาร ประกอบด้วย วิดีทัศน์ที่มีสิ่งกีดขวางประเภทวัตถุ วิดีทัศน์ที่มีสิ่งกีดขวางประเภทที่ไม่ใช่วัตถุ เช่น เสา กำแพง หรือ ประตู และ วิดีทัศน์ที่ไม่มีสิ่งกีดขวางใดเลย โดยวิดีโอทั้ง 20 คลิป ถูกถ่ายด้วยกล้องโทรศัพท์มือถือไอโฟน เอ็กซ์เอส แมกซ์ (iPhone XS Max) จะใช้เฉพาะกล้องคู่ด้านหลังมีความละเอียดอยู่ที่ตัวละ 12 ล้านพิกเซล มีขนาดรูปภาพ 1080 x 1920 พิกเซล และ มีอัตราการแสดงผลภาพอยู่ที่ 30 เฟรม ต่อ 1 วินาที

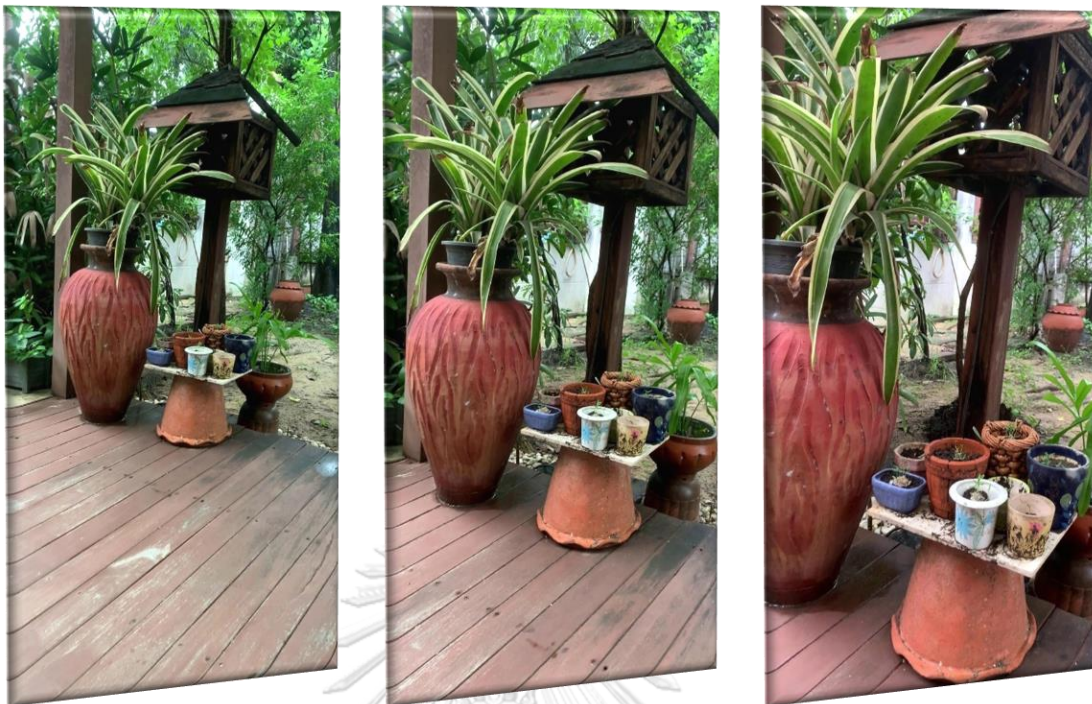
5.2 การทดลอง

ในงานวิจัยนี้จะแบ่งการทดลองออกเป็น 3 ส่วนหลัก ได้แก่

1. การเลือกใช้อัลกอริทึมสำหรับตรวจจับสิ่งกีดขวาง
2. การแจ้งเตือน
3. การคำนวณเวลาก่อนชน

โดยการทดลองทั้ง 3 ส่วน จะใช้วิดีโอชุดเดียวกัน ซึ่งสามารถแบ่งได้เป็น 3 รูปแบบ ได้แก่

1. วิดีทัศน์ที่มีสิ่งกีดขวางประเภทวัตถุ
2. วิดีทัศน์ที่มีสิ่งกีดขวางประเภทที่ไม่ใช่วัตถุ เช่น เสา กำแพง หรือ ประตู
3. วิดีทัศน์ที่ไม่มีสิ่งกีดขวาง



เวลา

รูปที่ 54 ตัวอย่างรูปภาพบางส่วนจากวิดีโอที่มัลลิ่งกิดขวางประเภทวัตถุ



เวลา

รูปที่ 55 ตัวอย่างรูปภาพบางส่วนจากวิดีโอที่มัลลิ่งกิดขวางประเภทที่ไม่ใช่วัตถุ



เวลา

รูปที่ 56 ตัวอย่างรูปภาพบางส่วนจากวิดีโอที่คนที่ไม่มีความคิดวางแผน

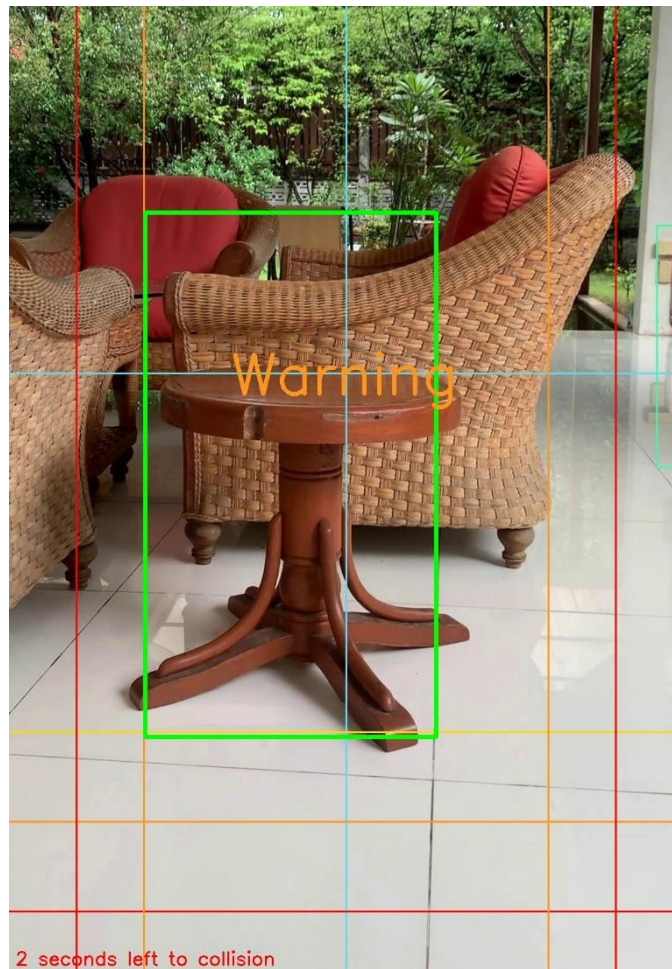
5.3 วิธีการทดลองและการวัดผล

ในงานวิจัยนี้จะแบ่งการทดลองออกเป็น 3 ส่วน ดังนี้

5.3.1 การเลือกใช้อัลกอริทึมสำหรับตรวจจับสิ่งกีดขวาง

ในส่วนของการตรวจจับสิ่งกีดขวาง เนื่องจากโปรแกรมต้นแบบที่ได้พัฒนาขึ้นมา นั้น เป็นการรวมกันของ 2 อัลกอริทึม คือ อัลกอริทึมสำหรับตรวจจับวัตถุ และ อัลกอริทึมสำหรับตรวจจับขอบภาพ เพราะฉะนั้นในส่วนนี้จะทดลองโดยการนำวิดีโอที่คนทั้ง 3 ประเภทเข้าโปรแกรมครั้งละ 1 คลิป จากนั้นทำการตรวจสอบว่าระบบสามารถเลือกอัลกอริทึมที่จะใช้ในการตรวจจับสิ่งกีดขวางได้ถูกต้องตามประเภทของสิ่งกีดขวางหรือไม่

ในกรณีที่ระบบเลือกใช้อัลกอริทึมสำหรับตรวจจับวัตถุ หลังจากโปรแกรมทำการประมวลผลเสร็จเรียบร้อยแล้ว ผลลัพธ์ที่ได้ในแต่ละรูปภาพ จะปรากฏกรอบสี่เหลี่ยมสีเขียวล้อมรอบสิ่งกีดขวางที่ระบบสามารถตรวจจับได้



รูปที่ 57 กรอบสี่เหลี่ยมสีเขียวล้อมรอบสิ่งกีดขวางประเภทวัตถุ

ในกรณีที่ระบบเลือกใช้อัลกอริทึมสำหรับตรวจจับขอบภาพ หลังจากโปรแกรมทำการประมวลผลเสร็จเรียบร้อยแล้ว ผลลัพธ์ที่ได้ในแต่ละรูปภาพ จะปรากฏเส้นตรงสีเขียว ซึ่งเป็นขอบของสิ่งกีดขวางที่ระบบสามารถตรวจจับได้



รูปที่ 58 เส้นตรงสีเขียวที่ขอบของสิ่งกีดขวางประเภทที่ไม่ใช้วัตถุ

5.3.2 การแจ้งเตือน

การวัดผลในส่วนนี้ จะดูจากประเภทของสิ่งกีดขวางเป็นหลัก โดยจะแยกออกเป็น 2 กรณี
ดังนี้

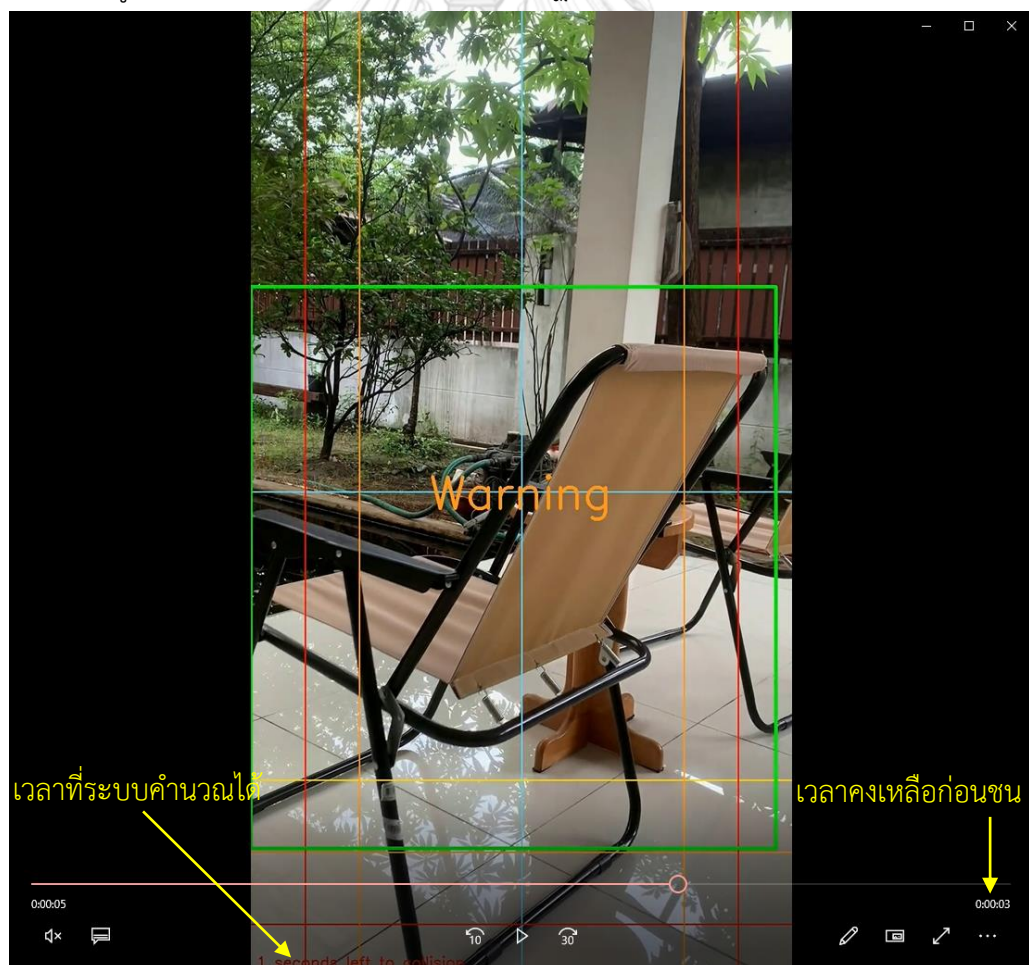
1. กรณีที่สิ่งกีดขวางเป็นวัตถุ ระบบจะใช้เส้นระยะในแนวนอนเป็นเกณฑ์เท่านั้น ซึ่งระบบจะแจ้งเตือนก็ต่อเมื่อมีบางส่วน of สิ่งกีดขวางอยู่ต่ำกว่าเส้นระยะนั้น ๆ ลงมา โดยจะแบ่งระดับความเสี่ยงด้วยสีของเส้น คือ หากมีบางส่วน of สิ่งกีดขวางอยู่ต่ำกว่าเส้นสีแดง ลงมา ระบบจะแสดงข้อความแจ้งเตือน ด้วยคำว่า Collision แต่ถ้าหากมีบางส่วน of สิ่งกีดขวางอยู่ในช่วงตั้งแต่เส้นสีเหลืองจนถึงเส้นสีแดง ระบบจะแสดงข้อความแจ้งเตือน ด้วยคำว่า Warning

- กรณีที่สิ่งกีดขวางที่ไม่ใช่วัตถุ ระบบจะใช้เส้นระยะทั้งแนวตั้งและแนวนอนเป็นเกณฑ์ หากเส้นตรงซึ่งเป็นขอบของสิ่งกีดขวางอยู่ในช่วงสีแดง ระบบจะแสดงข้อความแจ้งเตือน คำว่า Collision แต่ถ้าหากเส้นตรงอยู่ในช่วงสีส้ม ระบบจะแสดงข้อความแจ้งเตือน คำว่า Warning

5.3.3 การคำนวณเวลาก่อนชน

เวลาก่อนชนที่ระบบคำนวณได้ ในส่วนของการวัดผลจะตรวจสอบเฉพาะที่ระบบมีการแจ้งเตือน Warning ว่า เวลาที่คำนวณออกมา ณ ขณะนั้นมีความใกล้เคียงกับเวลาคงเหลือก่อนที่จะแจ้งเตือนว่า Collision มากน้อยเพียงใด

การทดลองในส่วนนี้จะใช้วิธีการนำวีดิทัศน์ที่ผ่านการประมวลผลในส่วนของตรวจจับสิ่งกีดขวางแล้ว ไปเปิดบนโปรแกรมเล่นวีดิโอ จากนั้นดูเวลาที่ระบบทำการคำนวณไว้ล่วงหน้าว่าจะชนว่า เหลือเวลาอีกเท่าใด เทียบกับ เวลาคงเหลือของวีดิทัศน์ก่อนที่จะเล่นจบ ซึ่งสามารถดูได้จากโปรแกรมเล่นวีดิโอ (ดังรูปที่ 59) โดยวีดิทัศน์จะตัดจบเมื่อปรากฏข้อความแจ้งเตือน คำว่า Collision



รูปที่ 59 เวลาที่ระบบคาดการณ์ไว้ล่วงหน้าว่าจะชน เทียบกับ เวลาคงเหลือก่อนที่วีดิทัศน์จะเล่นจบ

จากรูปที่ 59 จะเห็นว่า เวลาที่ระบบคาดการณ์ไว้ล่วงหน้าว่าจะชน คือ อีก 1 วินาที ในขณะที่ วิดีทัศน์นี้จะเล่นจบในอีก 3 วินาที จากตัวอย่างนี้ถือว่าระบบสามารถคำนวณเวลาก่อนจนได้ แต่ยังคง มีความคลาดเคลื่อนอยู่

5.4 ผลการทดลอง

5.4.1 การเลือกใช้อัลกอริทึมสำหรับตรวจจับสิ่งกีดขวาง

| วีดิทัศน์ | อัลกอริทึมในการตรวจจับสิ่งกีดขวาง | |
|--------------------|-----------------------------------|---------------|
| | ควรเลือกใช้ | ระบบเลือกใช้ |
| คลิปที่ 1 | ตรวจจับวัตถุ | ตรวจจับวัตถุ |
| คลิปที่ 2 | ตรวจจับวัตถุ | ตรวจจับวัตถุ |
| คลิปที่ 3 | ตรวจจับวัตถุ | ตรวจจับขอบภาพ |
| คลิปที่ 4 | ตรวจจับขอบภาพ | ตรวจจับวัตถุ |
| คลิปที่ 5 | ตรวจจับขอบภาพ | ตรวจจับขอบภาพ |
| คลิปที่ 6 | ตรวจจับวัตถุ | ตรวจจับวัตถุ |
| คลิปที่ 7 | ตรวจจับวัตถุ | ตรวจจับวัตถุ |
| คลิปที่ 8 | ตรวจจับขอบภาพ | - |
| คลิปที่ 9 | ตรวจจับวัตถุ | ตรวจจับวัตถุ |
| คลิปที่ 10 | ตรวจจับวัตถุ | ตรวจจับวัตถุ |
| คลิปที่ 11 | ตรวจจับขอบภาพ | ตรวจจับขอบภาพ |
| คลิปที่ 12 | ตรวจจับวัตถุ | ตรวจจับวัตถุ |
| คลิปที่ 13 | ตรวจจับวัตถุ | ตรวจจับวัตถุ |
| คลิปที่ 14 | - | - |
| คลิปที่ 15 | ตรวจจับขอบภาพ | ตรวจจับขอบภาพ |
| คลิปที่ 16 | ตรวจจับวัตถุ | ตรวจจับวัตถุ |
| คลิปที่ 17 | ตรวจจับวัตถุ | ตรวจจับวัตถุ |
| คลิปที่ 18 | ตรวจจับวัตถุ | ตรวจจับวัตถุ |
| คลิปที่ 19 | ตรวจจับขอบภาพ | ตรวจจับขอบภาพ |
| คลิปที่ 20 | ตรวจจับวัตถุ | - |
| ความถูกต้อง | 16/20 | |

ตารางที่ 3 ผลการทดลองในการเลือกใช้อัลกอริทึมตรวจจับสิ่งกีดขวางของระบบ

ผลการทดลองในตารางที่ 3 จะเห็นว่า ระบบสามารถเลือกใช้อัลกอริทึมในการตรวจจับสิ่งกีดขวางได้ถูกต้อง 16 คลิป จากทั้งหมด 20 คลิป คิดเป็นร้อยละ 80 รวมไปถึงระบบสามารถแยกได้ว่าในคลิปนั้น ๆ มีสิ่งกีดขวางหรือไม่ หากไม่พบสิ่งกีดขวางระบบจะไม่ปรากฏผลการตรวจจับ ดังเช่นในคลิปที่ 14

5.4.2 การแจ้งเตือน

| วีดิทัศน์ | ประเภทสิ่งกีดขวาง | ระบบสามารถแจ้งเตือนได้ |
|--------------------|-------------------|------------------------|
| คลิปที่ 1 | วัตถุ | ใช่ |
| คลิปที่ 2 | วัตถุ | ใช่ |
| คลิปที่ 3 | วัตถุ | ใช่ |
| คลิปที่ 4 | เสา | ใช่ |
| คลิปที่ 5 | ประตู | ใช่ |
| คลิปที่ 6 | วัตถุ | ใช่ |
| คลิปที่ 7 | วัตถุ | ใช่ |
| คลิปที่ 8 | เสา | ไม่ใช่ |
| คลิปที่ 9 | วัตถุ | ใช่ |
| คลิปที่ 10 | เสา | ใช่ |
| คลิปที่ 11 | เสา | ใช่ |
| คลิปที่ 12 | วัตถุ | ใช่ |
| คลิปที่ 13 | วัตถุ | ใช่ |
| คลิปที่ 14 | - | - |
| คลิปที่ 15 | ประตู | ใช่ |
| คลิปที่ 16 | วัตถุ | ใช่ |
| คลิปที่ 17 | วัตถุ | ใช่ |
| คลิปที่ 18 | วัตถุ | ใช่ |
| คลิปที่ 19 | วัตถุ | ใช่ |
| คลิปที่ 20 | วัตถุ | ไม่ใช่ |
| ความถูกต้อง | | 18/20 |

ตารางที่ 4 ผลการทดลองการแจ้งเตือนเมื่อระบบตรวจพบสิ่งกีดขวาง

ในตารางที่ 4 จะเห็นว่า ระบบสามารถแสดงข้อความแจ้งเตือนก่อนชนได้ แม้ว่าสิ่งกีดขวางจะต่างชนิดกัน โดยผลการทดลองบ่งชี้ว่า ระบบสามารถแจ้งเตือนได้ถูกต้องถึง 18 คลิป จากทั้งหมด 20 คลิป คิดเป็นร้อยละ 90 ซึ่งถือว่าในส่วนของการทำงานแจ้งเตือนนั้นมีความแม่นยำสูง

5.4.3 การคำนวณเวลาก่อนชน

| วิธีทัศน | ประเภทสิ่งกีดขวาง | เวลาคงเหลือก่อนชน (วินาที) | | ความคลาดเคลื่อน (วินาที) |
|------------|-------------------|----------------------------|-----------|--------------------------|
| | | เวลาจริง | ระบบคำนวณ | |
| คลิปที่ 1 | วัตถุ | 2 | 1 | + 1 |
| คลิปที่ 2 | วัตถุ | 2 | 3 | - 1 |
| คลิปที่ 3 | วัตถุ | 2 | 2 | 0 |
| คลิปที่ 4 | เสา | 2 | - | - |
| คลิปที่ 5 | ประตู | 2 | - | - |
| คลิปที่ 6 | วัตถุ | 1 | - | - |
| คลิปที่ 7 | วัตถุ | 1 | 1 | 0 |
| คลิปที่ 8 | เสา | 2 | - | - |
| คลิปที่ 9 | วัตถุ | 1 | 1 | 0 |
| คลิปที่ 10 | เสา | 2 | - | - |
| คลิปที่ 11 | เสา | 1 | - | - |
| คลิปที่ 12 | วัตถุ | 1 | 1 | 0 |
| คลิปที่ 13 | วัตถุ | 3 | - | - |
| คลิปที่ 14 | - | - | - | - |
| คลิปที่ 15 | ประตู | 1 | - | - |
| คลิปที่ 16 | วัตถุ | 0 | 0 | 0 |
| คลิปที่ 17 | วัตถุ | 0 | 1 | - 1 |
| คลิปที่ 18 | วัตถุ | 1 | 1 | 0 |
| คลิปที่ 19 | วัตถุ | 1 | 1 | 0 |
| คลิปที่ 20 | วัตถุ | 3 | - | - |

ตารางที่ 5 ผลการทดลองการคำนวณเวลาก่อนชน

ผลการทดลองการคำนวณเวลาก่อนชนในตารางที่ 5 พบว่า มีอยู่หลายสถานการณ์ที่ระบบไม่สามารถคำนวณหาเวลาคงเหลือก่อนที่จะไปชนสิ่งกีดขวางได้ (ช่องที่แสดง -) ซึ่งอาจจะเกิดได้จากหลายปัจจัย เช่น ความเร็วในการเคลื่อนที่ไม่คงที่ ระยะที่กำหนดให้ระบบเริ่มคำนวณเวลาก่อนชนอาจจะใกล้เกินไป หรือ เป็นสิ่งกีดขวางประเภทที่ไม่ใช้วัตถุ จึงทำให้ระบบไม่สามารถคำนวณเวลาได้จากเส้นขอบที่อยู่ด้านข้าง เป็นต้น

ซึ่งในส่วนของการกำหนดระยะ สามารถปรับตั้งค่าให้กับโปรแกรมได้

แม้ว่าในส่วนของการคำนวณเวลาก่อนชนจะยังไม่เสถียรเท่าที่ควร แต่หากดูจากเวลาที่ระบบสามารถคำนวณได้ จะเห็นว่า ความคลาดเคลื่อนของเวลาบวกลบ (+/-) เพียง 1 วินาทีเท่านั้น แสดงให้เห็นว่า สูตรการคำนวณเวลาก่อนชนที่ได้พัฒนาขึ้นมานั้น (ตามสมการที่ 21-23) สามารถนำไปประยุกต์ใช้ได้



บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงการสรุปผลการวิจัยการตรวจจับสิ่งกีดขวางสำหรับเก้าอี้รถเข็นไฟฟ้า รวมถึงข้อเสนอแนะต่าง ๆ เพื่อเป็นแนวทางในการพัฒนางานวิจัยต่อไปในอนาคต

6.1 สรุปผลการวิจัย

งานวิจัยนี้เป็นการพัฒนาโปรแกรมตรวจจับสิ่งกีดขวางสำหรับเก้าอี้รถเข็นไฟฟ้า โดยข้อมูลนำเข้าเป็นวีดิทัศน์จำนวน 20 คลิป ถ่ายในสภาพแวดล้อมการใช้งานภายในอาคาร ซึ่งประกอบไปด้วยวีดิทัศน์ที่มีสิ่งกีดขวางเป็นวัตถุ วีดิทัศน์ที่มีสิ่งกีดขวางประเภทที่ไม่ใช่วัตถุ เช่น เสาค้ำพวง หรือ ประตู และ วีดิทัศน์ที่ไม่มีสิ่งกีดขวาง โดยค่าความแม่นยำแสดงรายละเอียดในตารางที่ 6

| วีดิทัศน์ | สิ่งกีดขวาง | อัลกอริทึมตรวจจับสิ่งกีดขวาง | | ระบบแจ้งเตือนได้ | ความคลาดเคลื่อนการคำนวณเวลาก่อนชน (วินาที) |
|------------|-------------|------------------------------|---------------|------------------|--|
| | | ควรเลือกใช้ | ระบบเลือกใช้ | | |
| คลิปที่ 1 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับวัตถุ | ใช่ | + 1 |
| คลิปที่ 2 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับวัตถุ | ใช่ | - 1 |
| คลิปที่ 3 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับขอบภาพ | ใช่ | 0 |
| คลิปที่ 4 | เสาค้ำพวง | ตรวจจับขอบภาพ | ตรวจจับวัตถุ | ใช่ | - |
| คลิปที่ 5 | ประตู | ตรวจจับขอบภาพ | ตรวจจับขอบภาพ | ใช่ | - |
| คลิปที่ 6 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับวัตถุ | ใช่ | - |
| คลิปที่ 7 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับวัตถุ | ใช่ | 0 |
| คลิปที่ 8 | เสาค้ำพวง | ตรวจจับขอบภาพ | - | ไม่ใช่ | - |
| คลิปที่ 9 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับวัตถุ | ใช่ | 0 |
| คลิปที่ 10 | เสาค้ำพวง | ตรวจจับขอบภาพ | ตรวจจับวัตถุ | ใช่ | - |
| คลิปที่ 11 | เสาค้ำพวง | ตรวจจับขอบภาพ | ตรวจจับขอบภาพ | ใช่ | - |
| คลิปที่ 12 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับวัตถุ | ใช่ | 0 |
| คลิปที่ 13 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับวัตถุ | ใช่ | - |
| คลิปที่ 14 | - | - | - | - | - |
| คลิปที่ 15 | ประตู | ตรวจจับขอบภาพ | ตรวจจับขอบภาพ | ใช่ | - |
| คลิปที่ 16 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับวัตถุ | ใช่ | 0 |

| วีดิทัศน์ | สิ่งกีดขวาง | อัลกอริทึมตรวจจับสิ่งกีดขวาง | | ระบบ แจ้งเตือนได้ | ความคลาดเคลื่อน การคำนวณเวลา ก่อนชน (วินาที) |
|------------|-------------|------------------------------|---------------|----------------------|---|
| | | ควรเลือกใช้ | ระบบเลือกใช้ | | |
| คลิปที่ 17 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับวัตถุ | ใช่ | - 1 |
| คลิปที่ 18 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับวัตถุ | ใช่ | 0 |
| คลิปที่ 19 | วัตถุ | ตรวจจับวัตถุ | ตรวจจับขอบภาพ | ใช่ | 0 |
| คลิปที่ 20 | วัตถุ | ตรวจจับวัตถุ | - | ไม่ใช่ | - |
| ความแม่นยำ | | ร้อยละ 80 | | ร้อยละ 90 | +/- 1 |

ตารางที่ 6 ความแม่นยำในการตรวจจับสิ่งกีดขวาง การแสดงข้อความแจ้งเตือน และ การคำนวณเวลาก่อนชน

ผลการวิจัยในครั้งนี้สามารถสรุปได้ว่า โปรแกรมที่พัฒนาขึ้นมาสามารถนำไปใช้ในการตรวจจับสิ่งกีดขวางได้ เนื่องจากมีความแม่นยำสูงถึง 80% มีการแจ้งเตือนผู้ใช้เก้าอี้รถเข็นไฟฟ้าให้หลบเลี่ยงสิ่งกีดขวางได้ก่อนชน ความแม่นยำสูงถึง 90% และ มีการคำนวณเวลาก่อนชนให้ล่วงหน้าที่มีความคลาดเคลื่อนเพียง 1 วินาที เพื่อลดโอกาสเกิดอุบัติเหตุที่อาจจะเกิดขึ้นได้และอำนวยความสะดวกให้แก่ผู้ใช้งาน

6.2 ข้อเสนอแนะ

ในการวิจัยขั้นถัดไป อาจจะพัฒนาเพิ่มเติมในส่วนของการคำนวณเวลาก่อนชนให้สามารถคำนวณได้กับหลากหลายประเภทสิ่งกีดขวางมากยิ่งขึ้น หรือ จะเป็นการนำโปรแกรมต้นแบบที่ได้พัฒนาขึ้นในงานวิจัยนี้ไปเชื่อมต่อกับเก้าอี้รถเข็นไฟฟ้าให้มีระบบหยุดรถอัตโนมัติเมื่อตรวจพบสิ่งกีดขวางอยู่ในระยะที่กำหนด หรือ พัฒนาแอปพลิเคชันสำหรับแสดงข้อความแจ้งเตือนและเวลาใกล้ชนบนอุปกรณ์พกพา ทำให้ผู้ใช้งานสามารถใช้งานได้สะดวกยิ่งขึ้นและช่วยลดโอกาสเกิดอุบัติเหตุ หรือ อาจจะนำโปรแกรมนี้ไปพัฒนาต่อยอดประยุกต์ใช้ร่วมกับอุปกรณ์ช่วยเหลือผู้พิการประเภทอื่น ๆ ต่อไป

หากต้องการนำโปรแกรมตรวจจับสิ่งกีดขวางนี้ไปใช้อุปกรณ์ช่วยเหลือผู้พิการ ต้องคำนึงถึงขนาดของอุปกรณ์ด้วย เนื่องจากในระบบนี้มีการกำหนดเส้นที่ใช้เป็นเกณฑ์ในการแจ้งเตือน ดังนั้นความกว้างของอุปกรณ์จึงมีผลต่อการกำหนดเส้นแจ้งเตือนด้านข้าง ถ้าอุปกรณ์มีขนาดกว้าง ควรจะต้องกำหนดให้เส้นด้านข้างอยู่ชิดกับขอบรูปภาพ แต่หากอุปกรณ์มีขนาดไม่กว้างมาก สามารถกำหนดให้เส้นด้านข้างห่างจากขอบรูปภาพได้ตามความเหมาะสมของการใช้งาน

สำหรับงบประมาณที่ใช้ในระบบ สามารถแบ่งตามรูปแบบการนำไปใช้งานได้ 2 กรณี ดังนี้

1. กรณีใช้งานบนโทรศัพท์มือถือ จะต้องติดตั้งแอปพลิเคชันที่พัฒนาขึ้นสำหรับตรวจจับสิ่งกีดขวางโดยเฉพาะ แล้วเปิดโหมดถ่ายภาพ และ ใช้การแจ้งเตือนด้วยเสียง 2 ระดับ ซึ่งมีค่าใช้จ่ายโดยประมาณ ดังนี้
 - โทรศัพท์ 1 เครื่อง ราคาประมาณ 10,000 - 20,000 บาท
2. กรณีใช้งานบนเครื่องคอมพิวเตอร์ขนาดเล็ก จะต้องมิกกล้อง 1 ตัว ติดไว้ที่ด้านหลังเก้าอี้รถเข็นไฟฟ้า ต่อสาย USB จากกล้องไปที่เครื่องคอมพิวเตอร์ขนาดเล็ก และ ติดตั้งโปรแกรมไพทอนลงในเครื่องคอมพิวเตอร์ขนาดเล็ก จึงจะสามารถนำโปรแกรมต้นแบบนี้ไปทำงานบนเครื่องคอมพิวเตอร์นั้นได้ ซึ่งมีค่าใช้จ่ายโดยประมาณ ดังนี้
 - กล้องเว็บแคม (Webcam) 1 ตัว ราคาประมาณ 500 - 7,000 บาท
 - สายยูเอสบี (USB) 1 เส้น ราคาประมาณ 40 - 200 บาท
 - เครื่องคอมพิวเตอร์ขนาดเล็ก หรือ ราสเบอร์รี่พาย (Raspberry Pi) 1 ตัว
ราคาประมาณ 1,000 - 5,000 บาท

บรรณานุกรม

- [1] Lowe, D.G. *Object recognition from local scale-invariant features*. in *Proceedings of the Seventh IEEE International Conference on Computer Vision*. 1999.
- [2] Lowe, D.G., *Distinctive Image Features from Scale-Invariant Keypoints*. *International Journal of Computer Vision*, 2004. 60(2): p. 91-110.
- [3] Zhang, Z., *A flexible new technique for camera calibration*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000. 22(11): p. 1330-1334.
- [4] Redmon, J., et al. *You Only Look Once: Unified, Real-Time Object Detection*. in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016
- [5] Canny, J., *A Computational Approach to Edge Detection*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1986. **PAMI-8**(6): p. 679-698.
- [6] EL-Khamy, S.E., M. Lotfy, and N. El-Yamany. *A modified fuzzy Sobel edge detector*. in *Proceedings of the Seventeenth National Radio Science Conference. 17th NRSC'2000 (IEEE Cat. No.00EX396)*. 2000.
- [7] Surendran, S. and G. Ganesan, *Intelligent Reverse Sensor for Electric Wheelchair*. Vol. 5. 2018.
- [8] Mostofa, N., et al., *A Smart Walker for People with Both Visual and Mobility Impairment*. *Journal of King Fahd University of Petroleum & Technology*, 2021. 21(10).
- [9] Rao, S. and V.M. Singh. *Computer Vision and IoT Based Smart System for Visually Impaired People*. in *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*. 2021.
- [10] Nguyen, T.H., et al. *Real-Time Obstacle Detection for an Autonomous Wheelchair Using Stereoscopic Cameras*. in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. 2007.
- [11] Utaminingrum, F., et al. *Development of computer vision based obstacle detection and human tracking on smart wheelchair for disabled patient*. in *2017 5th International Symposium on Computational and Business Intelligence (ISCBI)*. 2017.

- [12] Urieva, N., et al. *Collision Detection and Avoidance using Optical Flow for Multicopter UAVs*. in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*. 2020.
- [13] Tapu, R., et al. *A Smartphone-Based Obstacle Detection and Classification System for Assisting Visually Impaired People*. in *2013 IEEE International Conference on Computer Vision Workshops*. 2013.
- [14] Redmon, J. and A. Farhadi, *YOLOv3: An Incremental Improvement*. 2018.



ภาคผนวก ก

1. การนำโปรแกรมต้นแบบไปใช้บนอุปกรณ์พกพา

เนื่องจากโปรแกรมต้นแบบนี้ถูกพัฒนาด้วยภาษา Python ดังนั้นหากจะนำไปใช้กับอุปกรณ์พกพา จะต้องทำการแปลงโค้ดให้อยู่ในรูปแบบที่แต่ละระบบปฏิบัติการรองรับ

1.1 นำไปใช้บนอุปกรณ์พกพาในระบบปฏิบัติการแอนดรอยด์ (Android OS)

ทำการแปลงโค้ดพื้นฐานจากภาษา Python ไปเป็นภาษา C, C++, Kotlin, XML Schema หรือ Java ซึ่งเป็นภาษาที่ระบบปฏิบัติการแอนดรอยด์รองรับ จากนั้นในส่วนของอัลกอริทึมที่ใช้ในการตรวจจับวัตถุในงานวิจัยนี้ใช้ YOLOv3 object detection และ อัลกอริทึมที่ใช้ในการตรวจจับตรวจจับขอบภาพใช้ Canny edge detection ซึ่งทั้ง 2 อัลกอริทึมนี้ต่างมี library สำหรับระบบปฏิบัติการแอนดรอยด์ เพื่อให้สามารถนำไปสร้างเป็นแอปพลิเคชันต่อไปได้

YOLOv3 for android : https://github.com/huuuuusy/YOLOv3-on-Android/blob/master/Readme_English.md

Canny edge detection for android : <https://zaven.co/blog/opencv-advanced-android-development-edge-detection>

1.2 นำไปใช้บนอุปกรณ์พกพาในระบบปฏิบัติการไอโอเอส (iOS OS)

ทำการแปลงโค้ดพื้นฐานจากภาษา Python ไปเป็นภาษา C, C++, Objective-C หรือ Java ซึ่งเป็นภาษาที่ระบบปฏิบัติการไอโอเอสรองรับ จากนั้นในส่วนของอัลกอริทึมที่ใช้ในการตรวจจับวัตถุ และตรวจจับขอบภาพ ให้ไปใช้ library ของ YOLOv3 object detection และ Canny edge detection ที่มีไว้สำหรับให้ระบบปฏิบัติการไอโอเอสเรียกใช้

YOLOv3 for iOS : <https://github.com/Ma-Dan/YOLOv3-CoreML>

Canny edge detection for iOS : <http://a2dr.blogspot.com/2014/12/canny-edge-detection.html>

ในกรณีที่ไม่ต้องการแปลงโค้ดพื้นฐานจากภาษา Python ไปเป็นภาษาที่แต่ละระบบปฏิบัติการรองรับ สามารถทำได้โดยการนำโปรแกรมต้นแบบนี้ไป run บน Raspberry Pi ซึ่งจะต้องทำการติดตั้ง Python ลงใน Raspberry Pi สามารถดูวิธีการติดตั้งได้จาก

<https://medium.com/@krittidejkeng/ติดตั้ง-python-3-8-บน-raspberry-pi-raspbian-460ab1e931f0>

2. การตั้งค่าระบบ

โปรแกรมต้นแบบที่ได้พัฒนาขึ้นมาในงานวิจัยนี้ได้มีการกำหนดค่าต่าง ๆ เพื่อใช้สำหรับให้ระบบทำการแจ้งเตือนเมื่อตรวจพบสิ่งกีดขวางในระยะที่กำหนด และ ใช้ในการคำนวณเวลาก่อนชนสิ่งกีดขวาง โดยการกำหนดค่าต่าง ๆ มีดังนี้

| ตัวแปร | ความหมาย | ค่าตั้งต้น | หน่วย |
|----------------------------|---|------------|-----------------|
| FPS | จำนวนรูปภาพต่อ 1 วินาทีของกล้องที่นำมาใช้ในการตรวจจับสิ่งกีดขวาง | 30 | รูปภาพต่อวินาที |
| SLOPE_VERTICAL_ABS | ค่าความชันสัมบูรณ์ของเส้นตรง ใช้สำหรับคัดกรองเส้นตรงที่ยาวที่สุด และมีความชันมากที่สุด | 2.5 | |
| SLOPE_THRESHOLD_VERTICAL | ค่าความชันของเส้นตรงในแนวตั้ง ใช้สำหรับคัดกรองเส้นตรงที่สนใจที่ได้จากการตรวจจับขอบภาพของสิ่งกีดขวาง | 0.5 | |
| SLOPE_THRESHOLD_HORIZONTAL | ค่าความชันของเส้นตรงในแนวนอน ใช้สำหรับคัดกรองเส้นตรงที่สนใจที่ได้จากการตรวจจับขอบภาพของสิ่งกีดขวาง | 0.5 | |
| CONFIDENCE_THRESHOLD | ค่าความมั่นใจที่ได้จากการตรวจหาวัตถุของอัลกอริทึม YOLOv3 ใช้คัดกรองวัตถุที่สนใจ ซึ่งจะต้องมีค่าตั้งแต่ค่าที่กำหนดขึ้นไป | 0.3 | อัตราร้อยละ |
| OBJECT_WIDTH_THRESHOLD | ความกว้างต่ำสุดของวัตถุที่สนใจ ใช้คัดกรองวัตถุซึ่งเป็นสิ่งกีดขวางที่ขนาดเริ่มส่งผลให้เกิดอุบัติเหตุได้ | 150 | พิกเซล |
| GAP_HEIGHT_THRESHOLD | ความสูงต่ำสุดของสิ่งกีดขวางที่สนใจ ใช้ได้ทั้งกรณีที่เป็นวัตถุหรือเส้นตรงที่ได้จากการตรวจจับขอบภาพ | 10 | พิกเซล |

| ตัวแปร | ความหมาย | ค่าตั้งต้น | หน่วย |
|--------------------------|--|-------------|-------------|
| WARNING_THRESHOLD | ค่าที่ใช้เป็นเกณฑ์ในการแจ้งเตือนก่อนชนสำหรับเส้นแนวนอนด้านล่าง ซึ่งจะคิดเป็นอัตราร้อยละของขนาดของรูปภาพ โดยวัดจากขอบของรูปภาพหัดใกล้เข้ามาตามค่าอัตราร้อยละที่กำหนดไว้ | 0.2 | อัตราร้อยละ |
| COLLISION_THRESHOLD | ค่าที่ใช้เป็นเกณฑ์ในการแจ้งเตือนเมื่อชนสำหรับเส้นแนวนอนด้านล่าง ซึ่งจะคิดเป็นอัตราร้อยละของขนาดของรูปภาพ โดยวัดจากขอบของรูปภาพหัดใกล้เข้ามาตามค่าอัตราร้อยละที่กำหนดไว้ | 0.05 | อัตราร้อยละ |
| SIDE_WARNING_THRESHOLD | ค่าที่ใช้เป็นเกณฑ์ในการแจ้งเตือนก่อนชนสำหรับเส้นแนวตั้งด้านข้าง ซึ่งจะคิดเป็นอัตราร้อยละของขนาดของรูปภาพ โดยวัดจากขอบของรูปภาพหัดใกล้เข้ามาตามค่าอัตราร้อยละที่กำหนดไว้ | 0.2 | อัตราร้อยละ |
| SIDE_COLLISION_THRESHOLD | ค่าที่ใช้เป็นเกณฑ์ในการแจ้งเตือนเมื่อชนสำหรับเส้นแนวตั้งด้านข้าง ซึ่งจะคิดเป็นอัตราร้อยละของขนาดของรูปภาพ โดยวัดจากขอบของรูปภาพหัดใกล้เข้ามาตามค่าอัตราร้อยละที่กำหนดไว้ | 0.1 | อัตราร้อยละ |
| TEXT_WARNING | ข้อความแจ้งเตือนก่อนชน | 'Warning' | |
| TEXT_COLLISION | ข้อความแจ้งเตือนเมื่อชน | 'Collision' | |

ประวัติผู้เขียน

| | |
|-------------------|---|
| ชื่อ-สกุล | เพ็ญพิชชา พัฒนจิตรศิลป์ |
| วัน เดือน ปี เกิด | 19 พฤศจิกายน 2536 |
| สถานที่เกิด | กรุงเทพมหานคร |
| วุฒิการศึกษา | ปริญญาตรี ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี จบการศึกษาปี 2560 |
| ที่อยู่ปัจจุบัน | 508/7-8 ซอย เจริญกรุง 109 ถนน เจริญกรุง แขวง บางคอแหลม เขต บางคอแหลม กรุงเทพมหานคร 10120 |
| ผลงานตีพิมพ์ | “Obstacles Detection for Electric Wheelchair with Computer Vision” โดย เพ็ญพิชชา พัฒนจิตรศิลป์ และ ศ.ดร. ประภาส จงสฤษดิ์ วัฒนา ในงานประชุมวิชาการนานาชาติ Knowledge and Smart Technology ครั้งที่ 14 (The 2022-14th International Conference on Knowledge and Smart Technology : KST) |