



โรงรถอัจฉริยะ

Smart Garage

นายณัฏฐิ์ ปทุมมา

Nantich Patumma

นายอภิพล บุญให้เจริญ

Apipol Boonhaicharoen

นางสาวปฐิ์ปรัชญ์ มานมุงศิลป์

Pulipach Manmungsin

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์

มหาวิทยาลัยศรีนครินทรวิโรฒ ปีการศึกษา พ.ศ.2563



คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ

ชื่อหัวข้อโครงการ โรงรถอัจฉริยะ
Smart Garage

นิสิต	นายนันทิช	ปทุมมา	60102010337
	นายอภิพล	บุญให้เจริญ	60102010350
	นางสาวปฐิปรัชญ์	มานมุงศิลป์	60102010569

ปริญญา	วิทยาศาสตร์บัณฑิต (วท.บ.)
ภาควิชา	วิทยาการคอมพิวเตอร์
อาจารย์ที่ปรึกษาโครงการ	อ.ดร. โสภณ มงคลลักษณ์

ลงชื่อ.....

(อ.ดร. โสภณ มงคลลักษณ์)

อาจารย์ที่ปรึกษาโครงการ

บทคัดย่อ

ในปัจจุบันเทคโนโลยีต่างๆได้เข้ามามีบทบาทในการดำรงชีวิตของมนุษย์อย่างแพร่หลาย เพื่อช่วยอำนวยความสะดวกได้ในหลายๆด้าน เพิ่มประสิทธิภาพในการทำงาน และเสริมสร้างความปลอดภัยให้กับมนุษย์ รวมทั้งการนำเทคโนโลยีสมัยใหม่มาประยุกต์ใช้ร่วมกับสิ่งต่างๆ หรือที่อยู่อาศัย ไม่ว่าจะเป็นภายในบ้าน รอบๆที่อยู่อาศัย หรือที่จอดรถของสถานที่นั้นๆ บทความนี้จะนำเสนอการพัฒนาระบบความปลอดภัยของโรงจอดรถในบ้านพักอาศัย ด้วยเทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งมาในรูปแบบของโรงรถจำลอง โดยระบบสามารถตรวจจับป้ายทะเบียนด้วยกล้องขนาดเล็ก เมื่อรถเข้าใกล้ในระยะที่กำหนด และส่งภาพที่ได้ไปตรวจสอบป้ายทะเบียนที่บันทึกไว้ล่วงหน้า หากเลขทะเบียนตรงกับที่ลงทะเบียนไว้ ประตูโรงรถจะเปิดอัตโนมัติ ภายในโรงรถมีเซ็นเซอร์ช่วยให้สามารถจอดรถยนต์ในตำแหน่งที่เหมาะสมและลดอุบัติเหตุจากการจอดรถ โดยสามารถควบคุมและตรวจสอบการทำงานของโรงรถได้ผ่านแอปพลิเคชันผ่านโทรศัพท์มือถือบนแพลตฟอร์มแอนดรอยด์ นอกจากนั้นมีการศึกษาเปรียบเทียบประสิทธิภาพการตรวจจับป้ายทะเบียนระหว่างการประมวลผลด้วยภาพที่จากกล้องตัวเดียวกับการประมวลผลจากภาพที่ได้จากกล้อง 2 ตัว โครงการนี้จะนำเสนอถึงการเพิ่มระดับความปลอดภัยโดยการใช้อินเทอร์เน็ตของสรรพสิ่งซึ่งมีราคาไม่สูง และง่ายต่อการติดตั้งเพื่อใช้งาน

Abstract

Internet of things (IOT) technology has become a part of human daily life. It can facilitate, increase work efficiency, increase work efficiency and improve security for humans. Including how we applied these modern technologies on things and people's houses. Whether inside the house ,around the residence ,a parking lot or even a garage. In this article we present an improvement of the household garage security system using IoT technology. We present the system with a model. The system can detect a car license plate. When the car is at a specific distance, a small camera will capture a picture of the license plate and pass it to the main system to analyze and compare with the registered license plate. If the license plate number matches the registered plate number, the door will be opened. There are sensors in the garage that will help the driver park the car in the right position. The door will be closed automatically. Users can also control and monitor the working process through a mobile application on android platform. Moreover, we study the performance of using one camera and two cameras for analyzing the license plate. This project reveals an enhanced level of security by using the Internet of Things which is affordable and easy to implement.

กิตติกรรมประกาศ

ขอขอบคุณภาควิชาวิทยาการคอมพิวเตอร์ สำหรับเครื่องมือ เครื่องใช้ และอุปกรณ์ต่างๆ ที่ใช้ในการดำเนินงานในครั้งนี้ รวมถึงสถานที่ที่ใช้ในการทำงาน ใช้ในการทดสอบระบบ และขอขอบพระคุณอาจารย์ประจำภาควิชา วิทยาการคอมพิวเตอร์ทุกท่าน ที่ได้ให้ความรู้ ให้คำปรึกษาและการช่วยเหลือ ให้ความสนับสนุนในการดำเนิน โครงการในครั้งนี้จนสามารถจัดทำโครงการได้ดีและมีประสิทธิภาพมากยิ่งขึ้น

ขอขอบคุณมหาวิทยาลัยศรีนครินทรวิโรฒ ที่อำนวยความสะดวกในเรื่องของสถานที่สำหรับดำเนินโครงการ และเป็นแหล่งศึกษาหาความรู้เพิ่มเติมเพื่อนำมาความรู้ที่ได้จากการศึกษามาพัฒนาระบบงาน และปรับปรุงระบบงานให้ดียิ่งขึ้น

ขอขอบพระคุณ อ.ดร.โสภณ มงคลลักษณ์ ที่คอยสนับสนุนในการดำเนินโครงการทุกๆ เรื่อง ตลอดจนเสร็จสิ้นโครงการ

สุดท้ายนี้ ขอขอบพระคุณทุกท่านที่มีส่วนเกี่ยวข้องในการดำเนินโครงการในครั้งนี้ ทั้งที่คอยให้คำแนะนำ ให้คำปรึกษา และให้การช่วยเหลือในเรื่องต่างๆ จนโครงการสำเร็จลุล่วงได้ด้วยดี คณะผู้จัดทำหวังว่าโครงการโรงรถอัจฉริยะ (Smart Garage) จะเป็นประโยชน์ต่อการศึกษา มีส่วนช่วยในการส่งเสริมการใช้เทคโนโลยีในปัจจุบันเพื่อให้เกิดประโยชน์สูงสุด และสามารถนำไปพัฒนาต่อยอดในอนาคตได้ดียิ่งๆ ขึ้นไป

คณะผู้จัดทำ

สารบัญ

บทที่ 1 บทนำ	9
1.1 ที่มาและความสำคัญ	9
1.2 วัตถุประสงค์ของโครงการ	10
1.3 ขอบเขตของงานวิจัย	11
1.4 ประโยชน์ที่คาดว่าจะได้รับ	11
บทที่ 2 องค์ความรู้ที่เกี่ยวข้อง	13
2.1 ทฤษฎีและแนวคิดที่ใช้	13
2.2 งานวิจัยที่เกี่ยวข้อง	17
บทที่ 3 วิธีการดำเนินโครงการ	18
3.1 วิธีการดำเนินงาน	18
3.2 แผนการดำเนินงานตลอดโครงการ	19
3.3 อุปกรณ์และเครื่องมือที่ใช้	21
3.4 ชุดข้อมูล (Dataset set) ที่ใช้ในโครงการ	22
3.5 การออกแบบระบบ	23
บทที่ 4 ผลการดำเนินโครงการ	24
บทที่ 5 สรุปผล อภิปรายผล และข้อเสนอแนะ	31
5.1 สรุปผลการศึกษาค้นคว้า	31
5.2 อภิปรายผล	32
5.3 ข้อเสนอแนะ	32
บรรณานุกรม	33

สารบัญรูปภาพ

รูปที่ 1 หลักการการทำงาน OpenCV	13
รูปที่ 2 ตัวอย่างการทำงาน Perspective Transformation	14
รูปที่ 3 ตัวอย่างการใช้งาน Tesseract	14
รูปที่ 4 อุปกรณ์ ESP32-CAM	15
รูปที่ 5 อุปกรณ์ ESP8266	16
รูปที่ 6 Android Studio	16
รูปที่ 7 หลักการการทำงานของงานวิจัย	17
รูปที่ 8 หลักการการทำงานของโครงงาน	23
รูปที่ 9 การเชื่อมต่ออุปกรณ์	24
รูปที่ 10 การบันทึกรูปภาพและส่ง MQTT message	24
รูปที่ 11 การรับ message และทำงานต่อ	25
รูปที่ 12 ตัวอย่างการรูปตัวอย่างหลังเปลี่ยน Module กล้อง esp32-cam	25
รูปที่ 13 ตัวอย่างการตรวจจับวัตถุที่เป็นรถจากรูปตัวอย่าง	25
รูปที่ 14 ตัวอย่างการตรวจจับป้ายจากรูปตัวอย่าง	26
รูปที่ 15 ตัวอย่างการแบ่งเลขทะเบียนและจังหวัดของป้ายทะเบียนรถแล้ว	26
รูปที่ 16 ตัวอย่างการแปลงอักษรจากรูปภาพหลังตรวจจับทะเบียนแล้ว	26
รูปที่ 17 รูปแบบจำลองโรงรถ	27
รูปที่ 18 ตัวอย่างการทำงานการเปิดประตูโรงรถ	27
รูปที่ 19 ตัวอย่างการทดสอบส่งข้อความแจ้งเตือนผ่านไลน์	27
รูปที่ 20 ตัวอย่างการทดสอบการตรวจจับป้ายทะเบียน และผลลัพธ์บน Terminal	28
รูปที่ 21 แบบจำลองโรงรถ	28
รูปที่ 22 ตัวอย่างผลลัพธ์ของภาพที่ถ่ายได้จากกล้อง 2 ตัว	29
รูปที่ 23 ตัวอย่างการทำงานบนแอปพลิเคชัน Smart Garage	30

สารบัญตาราง

ตารางที่ 1 แผนการดำเนินงานตลอดโครงการ

19



บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

เนื่องจากปัจจุบันนี้เทคโนโลยีต่างๆ มีความก้าวหน้าและเข้ามามีบทบาทต่อการใช้ชีวิตประจำวันของมนุษย์เป็นอย่างมาก โดยเฉพาะในตอนนี้ Internet of Things (IoT) ได้รับความสนใจอย่างมาก คือ การที่อุปกรณ์ต่าง ๆ สิ่งต่าง ๆ ได้ถูกเชื่อมโยงทุกอย่างสู่โลกอินเทอร์เน็ต ให้อุปกรณ์นั้นสามารถรับ-ส่งข้อมูล ซึ่งส่งผลมนุษย์สามารถสั่งการควบคุมการใช้งานอุปกรณ์ต่าง ๆ ผ่านทางเครือข่ายอินเทอร์เน็ต ทำให้เพิ่มความสะดวกสบาย และเพิ่มในเรื่องความปลอดภัยต่อผู้ใช้งาน ตามจุดประสงค์ของแต่ละอุปกรณ์ในแต่ละการใช้งานอีกด้วย ตัวอย่างเช่น กล้องวงจรปิด ที่สามารถติดในบ้านเพื่อตรวจจับสิ่งผิดปกติที่อยู่ในบ้าน สามารถจับความผิดปกติและส่งข้อมูลไปยังสมาร์ตโฟนของเจ้าของได้ บ้านอัจฉริยะ โดยเราสามารถควบคุมการใช้งานอุปกรณ์อิเล็กทรอนิกส์ในบ้านได้โดยใช้เพียงสมาร์ตโฟน หรือ อุปกรณ์สื่อสารที่ทำการติดตั้งไว้ โดยตัวอย่างของสิ่งที่สามารถควบคุมได้คือ การเปิด/ปิด ไฟในแต่ละห้อง การตั้งค่าการล็อคประตู ตรวจสอบสิ่งของในตู้เย็น เป็นต้น แต่อุปกรณ์ที่เป็นIoTก็ยังมีจุดอ่อนในเรื่องของด้านความปลอดภัย เพราะเป็นอุปกรณ์ที่มีการเชื่อมต่อทุกอย่างเข้าด้วยกันผ่านทางอินเทอร์เน็ตหากสามารถทำการเจาะเข้าระบบได้ส่วนหนึ่งก็อาจจะทำให้เกิดความเสียหายต่อส่วนอื่นๆที่เกี่ยวข้องกันได้หรือหากขาดการเชื่อมต่ออินเทอร์เน็ตก็จะทำให้เราไม่สามารถสั่งงานอุปกรณ์ได้และอาจเกิดการทำงานผิดพลาดของอุปกรณ์ที่เกิดขึ้นจากการเขียนคำสั่งไม่รัดกุมหรือข้อจำกัดของตัวอุปกรณ์ที่เป็นในส่วนของความจำซึ่งหากต้องการหน่วยความจำที่มีประสิทธิภาพมาก ๆ ก็จะมีค่าใช้จ่ายที่สูงขึ้นตามเช่นกันซึ่งสิ่งต่างๆเหล่านี้เป็นสิ่งที่ควรตระหนักถึงเพราะอาจจะทำให้เกิดผลเสียต่อผู้ใช้งานไม่ว่าจะเป็นต่อบุคคลหรือองค์กรในหลายๆด้าน เช่น ด้านร่างกาย ด้านทรัพย์สิน เป็นต้น

ในประเทศไทยมีบ้านที่หลากหลายรูปแบบและในแต่ละแบบจะมีโรงรถอยู่ด้วยซึ่งโรงรถในประเทศไทยปัจจุบันนี้เป็นเพียงหลังคาหรือเพียงกันสาดยื่นออกมาจากตัวบ้านเท่านั้นเพื่อเป็นส่วนสำหรับไว้จอดรถและป้องกันรถไม่ให้โดนลมหรือฝนโดยตรงเมื่อจอดไว้ที่บ้าน แต่ไม่มีการป้องกันความเสียหายของรถจาก การโจรกรรมจากผู้ไม่ประสงค์ดี ซึ่งอาจทำให้เกิดความเสียหายต่อรถหรือเจ้าของรถได้ เช่น การดักทำร้ายในขณะที่เจ้าของบ้านลงมาเปิดประตูหน้าบ้านเพื่อเอารถเข้าจอด การขโมยรถที่จอดอยู่ภายในบ้าน เป็นต้น หรือเหตุการณ์ที่เกิดขึ้นโดยไม่ได้ตั้งใจจากสถานการณ์ต่าง ๆ เช่น การถอยรถชนกับ

กำแพงโรงรถ ทำให้เกิดความเสียหายต่อรถ การจอดรถไว้หน้าประตูอาจมีรถคันอื่นที่ขับมาอย่างประมาท มาเฉี่ยวชน เป็นต้น

โดยทางผู้จัดทำได้ให้ความสนใจในเรื่องการทำ smart garage หรือ โรงรถอัจฉริยะ ซึ่งเป็นการนำเทคโนโลยีที่เกี่ยวกับ IoT มาต่อยอดการใช้งานร่วมกับโรงรถ ให้มีระบบรัดกุมต่อการใช้งาน โดยสามารถตรวจสอบเลขทะเบียนรถเพื่อ เปิด/ปิด ประตูอัตโนมัติ โดยทำตามเงื่อนไขที่ตั้งไว้ ตรวจสอบระยะเวลาการห่างระหว่างรถกับกำแพงโดยสามารถใส่ข้อมูลที่ต้องใช้ ออกคำสั่งควบคุมได้ผ่านแอปพลิเคชันบนมือถือ โดยจะมีการให้กรอกข้อมูลเพื่อให้ระบบสามารถนำข้อมูลไปใช้ในการดำเนินการของระบบ และมีการตั้งค่าระบบ ให้มีความปลอดภัยต่อผู้ใช้งานเมื่อไม่ได้ทำการเชื่อมต่ออินเทอร์เน็ตให้เหมาะสมด้วยการตั้งค่าพื้นฐาน หรือ ให้ผู้ใช้งานสามารถเข้ามาปรับใช้ตามความต้องการได้ เพื่อก่อให้เกิดความสะดวกสบาย และความปลอดภัยแก่ผู้ใช้งานและช่วยให้ประหยัดเวลาในการใช้งานอีกด้วย

คณะผู้จัดทำมีความประสงค์ที่จะจัดทำโครงการ โรงรถอัจฉริยะ ที่เป็นการทำให้โรงรถสามารถเชื่อมต่อกับอินเทอร์เน็ตแล้วทำให้ผู้ใช้งานสามารถทำการควบคุมผ่านทางแอปพลิเคชันมือถือบนแพลตฟอร์มแอนดรอยด์ได้ ซึ่งจะก่อให้เกิดประโยชน์ต่อผู้ใช้งานในด้านความสะดวกสบาย โดยผู้ใช้งานไม่ต้องลงมาเปิดประตูด้วยตนเอง ป้องกันการเกิดความเสียหายของรถจากการชนกำแพงขณะกำลังจอดรถ และด้านความปลอดภัยจากภัยต่างๆซึ่งอาจก่อให้เกิดความเสียหายต่อบุคคลหรือทรัพย์สินในบริเวณที่พักอาศัยโดยจะมีการแจ้งเตือนจากระบบหากมีการตรวจจับได้ถึงความผิดปกติ และทางคณะผู้จัดทำจะสร้างระบบให้มีประสิทธิภาพโดยการทำการเปรียบเทียบการใช้งานทั้งในด้านของระบบ และตัวอุปกรณ์ ตามความเหมาะสม

1.2 วัตถุประสงค์ของโครงการ

1.2.1 เพื่อเพิ่มความสะดวกสบายในการใช้งานโรงรถโดยที่ประตูโรงรถและหลอดไฟจะเปิดเองอัตโนมัติเมื่อรถมาจอดถึงหน้าโรงรถ

1.2.2 เพื่อเพิ่มความปลอดภัยของการใช้งานโรงรถเมื่อมีรถที่ป้ายทะเบียนรถไม่ตรงกับป้ายทะเบียนที่บันทึกไว้ มาจอดหน้าโรงรถจะส่งรูปรถคันนั้นกลับไปให้เจ้าของบ้าน

1.2.3 เพื่อป้องกันการเปิดประตูโรงรถทิ้งไว้เมื่อจอดรถเสร็จประตูโรงรถจะปิดเองอัตโนมัติ

1.2.4 เพื่อเพิ่มความปลอดภัยจากการถอยรถจอดเมื่อรถจอดได้ตำแหน่งที่เหมาะสมจะมีสัญญาณเสียงแจ้งเตือนให้ผู้ขับ

1.2.6 เพื่อศึกษาเพิ่มเติมเกี่ยวกับการใช้งานระหว่าง Computer Vision กับ อุปกรณ์ IoT

1.2.7 เพื่อเพิ่มคุณภาพของรูปภาพที่จะนำไปใช้ในการวิเคราะห์ป้ายทะเบียน เพื่อให้เกิดประสิทธิภาพที่ดียิ่งขึ้น

1.2.8 เพื่อเพิ่มความสะดวกสบายในการใช้งานโรงรถโดยจะสามารถควบคุมการทำงานของโรงรถได้ผ่านทางแอปพลิเคชันมือถือ

1.2.9 เพื่อเพิ่มประสิทธิภาพในการตรวจจับ ให้สามารถยืนยันได้ว่าป้ายทะเบียนที่สามารถตรวจจับได้ในขณะนั้นเป็นรถจริงๆ ไม่ใช่แค่ตัวป้ายทะเบียน เพื่อป้องกันการปลอมแปลงป้ายทะเบียนรถ

1.3 ขอบเขตของโครงการ

1.3.1 ใช้การตรวจจับป้ายทะเบียนรถจากกล้องเพื่อยืนยันตัวตนของรถ

1.3.2 ใช้ภาพป้ายทะเบียนตัวอย่างเพื่อเพิ่มประสิทธิภาพในการตรวจจับป้ายทะเบียน

1.3.3 ใช้ ESP8266 ในการสั่งการทำงานระหว่างอุปกรณ์ IoT

1.3.4 สามารถส่งภาพกลับไปหาผู้ใช้งานได้เมื่อมีรถคันอื่นมาจอดหน้าโรงรถ

1.3.5 สามารถควบคุมการใช้งานประตูโรงรถผ่าน Freeboard บน NETPIE ได้

1.3.6 สามารถใช้งานกล้อง ESP32-CAM จำนวน 2 ตัวร่วมกันเพื่อเพิ่มประสิทธิภาพในการจับภาพการนำภาพไปวิเคราะห์ในลำดับถัดไป

1.3.7 สามารถควบคุมการทำงานของโรงรถได้ผ่านทางแอปพลิเคชันมือถือ

1.3.8 สามารถตรวจจับและยืนยันได้ว่าวัตถุที่ได้รับการตรวจจับนั้นเป็นรถจริงๆหรือไม่

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 สามารถเพิ่มความสะดวกในการใช้งานโรงรถ โดยที่ประตูโรงรถและไฟจะสามารถเปิดได้เองอัตโนมัติ

1.4.2 สามารถเพิ่มความปลอดภัยของการใช้งานโรงรถในกรณีที่มียานพาหนะที่ป้ายทะเบียนรถไม่ตรงกับป้ายทะเบียนที่บันทึกไว้มาจอดหน้าโรงรถ

1.4.3 สามารถป้องกันการเปิดประตูโรงรถทิ้งไว้ โดยประตูโรงรถจะสามารถปิดได้เองอัตโนมัติ

1.4.4 สามารถเพิ่มความปลอดภัยจากการจอดรถ

1.4.5 สามารถนำข้อมูลเกี่ยวกับการใช้งาน Computer Vision และอุปกรณ์ IOT และสิ่งที่ได้รับในโครงการไปประยุกต์ใช้ในอนาคตได้อย่างมีประสิทธิภาพ

1.4.6 สามารถเพิ่มคุณภาพของรูปภาพ เพื่อนำไปใช้ในการวิเคราะห์ป้ายทะเบียนที่มีประสิทธิภาพที่ดียิ่งขึ้น

1.4.7 สามารถเพิ่มความสะดวกสบายในการใช้งานโรงรถ เมื่อมีการควบคุมการทำงานของโรงรถได้ผ่านทางแอปพลิเคชันมือถือ

1.4.8 สามารถเพิ่มประสิทธิภาพในการตรวจจับ ให้สามารถยืนยันได้ว่าป้ายทะเบียนที่สามารถตรวจจับได้ในขณะนั้นเป็นรถจริงๆ เพื่อป้องกันการฉ้อโกงปลอมแปลงป้ายทะเบียนรถ



บทที่ 2

องค์ความรู้ที่เกี่ยวข้อง

2.1 ทฤษฎีและแนวคิดที่ใช้

2.1.1 Open source Computer Vision Library(OpenCV)

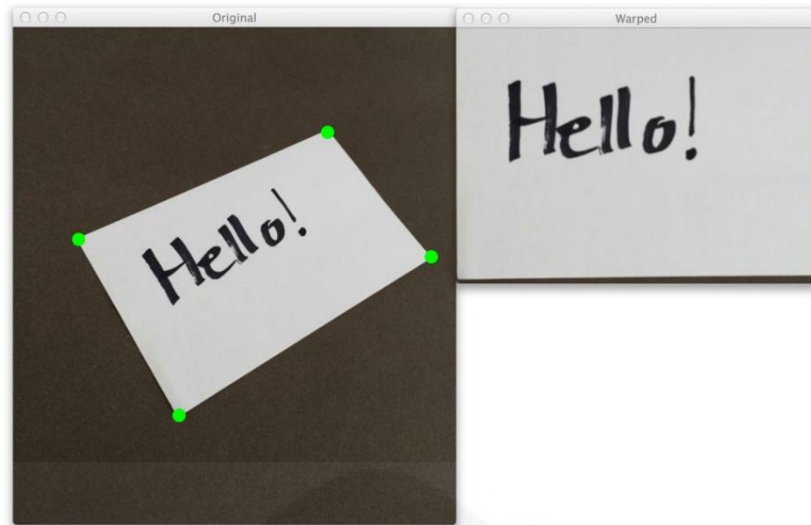
OpenCV เป็น Library สำหรับเขียนโปรแกรมหรือพัฒนาซอฟต์แวร์ให้สามารถประมวลผลภาพและวิดีโอได้โดยส่วนใหญ่จะมุ่งเป้าไปที่การแสดงผลด้วยคอมพิวเตอร์แบบเรียลไทม์ สามารถใช้ในการพัฒนาระบบ Open Source ที่นำมาใช้สร้าง Machine Learning หรือ AI ในการจำแนกวัตถุ หรือ การจดจำใบหน้า และไลบรารีนี้มีภาษาที่ใช้ในการพัฒนาคือ C++ และยังรองรับภาษา Python ,Java และ MathLab และใช้OpenCV ในการตรวจจับสิ่งต่างๆที่เราต้องการจาก Model ที่เราสร้างซึ่งเป็นไฟล์ XML ที่ประกอบไปด้วยภาพตัวอย่างของสิ่งนั้นๆจำนวนมากและหลักการทำงานของOpenCV เป็นตามรูปที่ 9.1



รูปที่ 1 หลักการทำงานของ OpenCV

2.1.2 Perspective Transformation

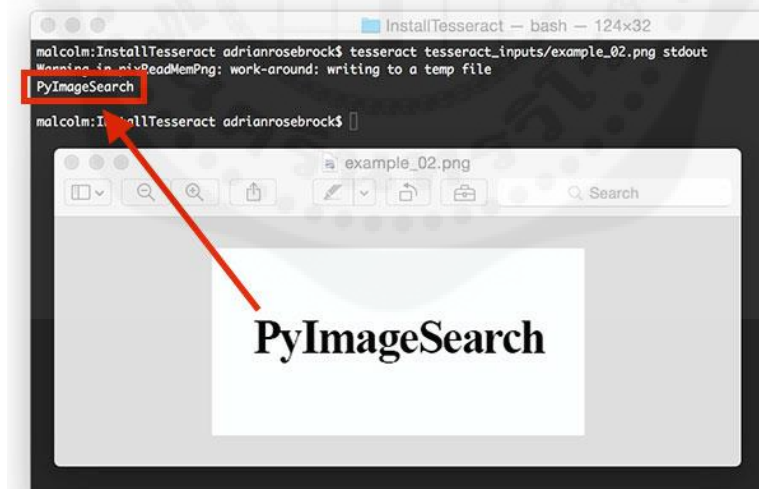
การเปลี่ยนมุมมอง Perspective Transformation การเปลี่ยนแปลงมุมมอง จำเป็นต้องมีการเปลี่ยนแปลงขนาดตารางเมทริกซ์ 3×3 โดยเส้นตรงจะยังคงอยู่ที่เดิม แม้จะเปลี่ยนแปลงแล้ว เพื่อหาเมทริกซ์การเปลี่ยนแปลงครั้งนี้คุณจำเป็นต้องมี 4 จุดสำหรับภาพที่นำเข้ามา โดยหาจุดที่สอดคล้องกัน



รูปที่ 2 ตัวอย่างการทำงาน Perspective Transformation

2.1.3 Tesseract OCR

Tesseract เป็นเครื่องมือหนึ่งที่ใช้ในการรู้จำตัวอักษรหรือ optical character recognition (OCR) ซึ่งมีอยู่ภายใต้ใบอนุญาต Apache 2.0 สามารถใช้ได้โดยตรงหรือ ใช้ API เพื่อดึงข้อความที่พิมพ์ออกมาจากรูปภาพ รองรับภาษาที่หลากหลายรวมถึงภาษาไทยด้วย



รูปที่ 3 ตัวอย่างการใช้งาน Tesseract

2.1.4 Levenshtein Distance และ Text Similarity

เป็นขั้นตอนการวัดค่าความต่างกันของอักขระสองชุด ระหว่างชุดแรกที่เป็นต้นแบบ และ ชุดที่สองที่เป็นชุดเปรียบเทียบ โดยค่าความต่างกันจะวัดจากจำนวนของการที่จะต้องทำการตัดออก แทรก และ แทนที่อักขระในชุดที่นำมาเปรียบเทียบกับจนกระทั่งมีลักษณะเหมือนชุดอักขระที่เป็นต้นแบบทุกประการ

2.1.5 ESP32-CAM

เป็นโมดูลกล้องขนาดเล็กมาก ที่มาพร้อมกับชิพ ESP32-S และสามารถใช้งาน ESP32-CAM เป็นกล้องวงจรปิด ที่สามารถดูภาพแบบ real time ผ่านโทรศัพท์มือถือหรือทางหน้าเว็บเบราว์เซอร์ได้สามารถดูตัวอย่างของ ESP32-CAM ตามรูปที่ 9.4.1 และ ESP32-CAM นั้นเป็นตัวส่งภาพวิดีโอแบบ real time เพื่อนำไปวิเคราะห์และประมวลผลภาพว่าภาพที่ได้นั้นตรงกับที่ลงทะเบียนไว้หรือไม่



รูปที่ 4 อุปกรณ์ ESP32-CAM

2.1.6 ESP8266

ESP8266 เป็นชื่อเรียกของชิพของโมดูล ESP8266 สำหรับติดต่อสื่อสารบนมาตรฐาน WiFi ทำงานที่แรงดันไฟฟ้า 3.0-3.6V ทำงานใช้กระแสโดยเฉลี่ย 80mA รองรับคำสั่ง deep sleep ในการประหยัดพลังงาน ใช้กระแสต่ำกว่า 10 ไมโครแอมป์ สามารถ wake up กลับมาส่งข้อมูลใช้เวลาไม่น้อยกว่า 2 มิลลิวินาที ภายในมี Low power MCU 32bit ทำให้เราเขียนโปรแกรมสั่งงานได้ มีวงจร analog digital converter ทำให้สามารถอ่านค่าจาก analog ได้ความละเอียด 10bit ทำงานได้ที่อุณหภูมิ -40 ถึง 125 องศาเซลเซียส

ESP8266 ติดต่อกับ Wi-Fi แบบ Serial สามารถเขียนโปรแกรมลงไปในชิพ โดยใช้ Arduino IDE ได้ ทำให้การเขียนโปรแกรมและใช้งานเป็นเรื่องง่าย คล้ายกับการใช้ Arduino แน่นอนว่าสามารถติดต่ออุปกรณ์อื่น ๆ เซนเซอร์ ต่าง ๆ แบบสแตนด์ Arduino ถ้ามีพื้นฐาน Arduino อยู่แล้ว ก็จะเข้าใจและใช้งานได้รวดเร็ว



รูปที่ 5 อุปกรณ์ ESP8266

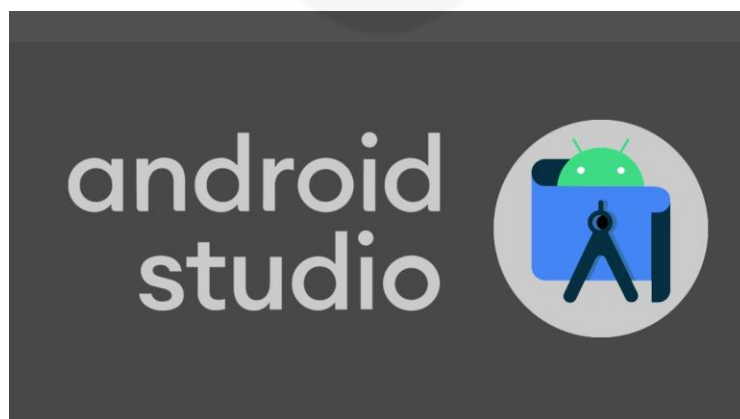
2.1.9 ImageAI

เป็น Open-source python library ที่สร้างขึ้นเพื่อช่วยให้นักพัฒนาสามารถสร้างแอปพลิเคชันและระบบที่มีความสามารถในการเรียนรู้เชิงลึกและคอมพิวเตอร์วิชั่นในตัวโดยใช้โค้ดที่เรียบง่าย

ปัจจุบัน ImageAI รองรับการทำนายภาพและการ train โดยใช้อัลกอริธึม Machine Learning 4 แบบที่ได้รับการ train บนชุดข้อมูล ImageNet-1000 ImageAI ยังรองรับการตรวจจับวัตถุ การตรวจจับวีดีโอและการติดตามวัตถุโดยใช้ RetinaNet, YOLOv3 และ TinyYOLOv3 ที่ได้รับการ train เกี่ยวกับชุดข้อมูล COCO

2.1.9 Android Studio

เป็นเครื่องมือไว้สำหรับพัฒนาโปรแกรม Android โดยเฉพาะ โดยวัตถุประสงค์ของ Android Studio คือต้องการพัฒนาเครื่องมือ IDE ที่สามารถพัฒนา App บน Android ให้มีประสิทธิภาพมากขึ้น ทั้งด้านการออกแบบ GUI ที่ช่วยให้สามารถ Preview ตัว App มุมมองที่แตกต่างกัน บน Smart Phone แต่ละรุ่น สามารถแสดงผลบางอย่างได้ทันที โดยไม่ต้องรัน App บน Emulator รวมทั้งยังแก้ไขปรับปรุงในเรื่องของความเร็วของ Emulator ที่ยังเจอ ปัญหาน้อยอยู่ในปัจจุบัน



รูปที่ 6 Android Studio

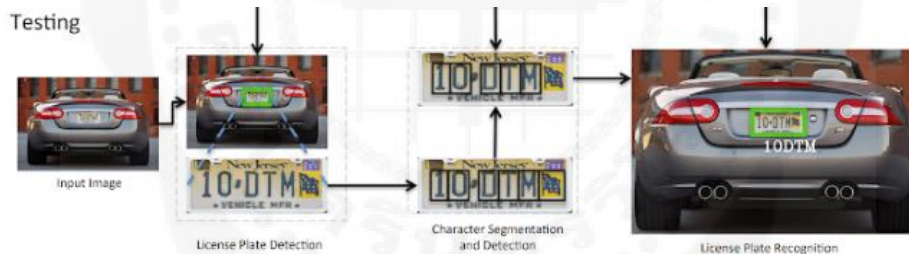
2.1.10 test.mosquitto.org

MQTT เป็นเพียง Protocol สื่อสาร ในการใช้งานจริง จำเป็นต้องมี Software และ Hardware มารองรับ MOSQUITTO เป็น Open source Message Broker ที่สนับสนุน MQTT แต่ในบทความนี้จะใช้ Broker ที่ทาง MOSQUITTO เตรียมไว้ให้เพื่อในการเรียนรู้และทดสอบโดยไม่เสียค่าใช้จ่ายซึ่งมี URL คือ test.mosquitto.org

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 งานวิจัยเรื่อง “License Plate Detection and Recognition Using Deeply Learned Convolutional Neural Networks”

ซึ่งจะนำเสนอถึง end-to-end system ที่จะใช้สำหรับการตรวจจับและการจดจำป้ายทะเบียนรถ โดยจะแนะนำตัว pipeline ใหม่ที่มีลำดับของ Convolutional Neural Networks (CNN) ในการทำงานเพื่อที่จะระบุถึง task ของการตรวจจับและการจดจำ และจะแสดงให้เห็นว่า pipeline ดังกล่าวมีประสิทธิภาพเหนือกว่าทางออกเชิงพาณิชย์ที่ล้ำสมัยซึ่งเป็นชุดข้อมูลมาตรฐานจากรูป 9.1 จะเป็นหลักการการทำงานในการตรวจสอบป้ายทะเบียนรถ



รูปที่ 7 หลักการการทำงานของงานวิจัย

2.2.2 งานวิจัยเรื่อง “OpenCV for Computer Vision Applications”

ซึ่งจะกล่าวว่า OpenCV มีไว้เพื่อทำงานเกี่ยวกับ Image processing เป็นหลัก โดยจะมีเครื่องมือต่างๆเพื่อแก้ไขปัญหาในเรื่องของ Computer vision และกล่าวถึงการใช้งาน OpenCV บน แอปพลิเคชัน โดยมีการอธิบายเทคนิคต่างๆของการทำ image processing และมีการแสดงให้เห็นถึงตัวอย่างการทำงาน การคำนวณ การเปรียบเทียบ และผลลัพธ์การทำงาน ให้เห็นภาพอย่างชัดเจน

บทที่ 3

วิธีการดำเนินโครงการ

3.1 วิธีการดำเนินงาน

3.1.1 เริ่มต้นวางแผนโครงการ

- 3.1.1.1 เลือกหัวข้อและปรึกษากับอาจารย์ที่ปรึกษา
- 3.1.1.2 ศึกษาค้นคว้าเอกสารและงานวิจัยที่เกี่ยวข้อง
- 3.1.1.3 เลือกเครื่องมือที่จะใช้ในโครงการ
- 3.1.1.4 ศึกษาข้อมูลเกี่ยวกับการใช้งาน Computer Vision
- 3.1.1.5 กำหนดรายละเอียดของโครงการ

3.1.2 การวิเคราะห์ระบบ

- 3.1.2.1 ศึกษาวิธีใช้งาน Computer Vision กับอุปกรณ์ IOT เพื่อมาประยุกต์ใช้กับการประมวลผลจากกล้องที่ใช้งาน
- 3.1.2.2 ศึกษาวิธีการใช้งาน Raspberry ในการเชื่อมต่อกับกล้องวงจรปิดหรือกล้อง ESP32-CAM
- 3.1.2.3 ทดสอบการทำงานของกล้องวงจรปิดหรือกล้อง ESP32-CAM, Raspberry Pi, เซนเซอร์ และอุปกรณ์ IOT
- 3.1.2.4 ทดลอง Train และ Test model

3.1.3 การออกแบบระบบ

- 3.1.3.1 ออกแบบโรงรถ
- 3.1.3.2 ออกแบบระบบการทำงานของโรงรถ

3.1.4 การพัฒนาและติดตั้งระบบ

- 3.1.4.1 เชื่อมต่อ Raspberry Pi กับกล้อง หลอดไฟ และเซนเซอร์ต่างๆ
- 3.1.4.2 สร้างแอปพลิเคชันสำหรับสื่อสารรับ-ส่งข้อมูลระหว่างอุปกรณ์และผู้ใช้
- 3.1.4.3 สร้างแบบจำลองสำหรับใช้ทดสอบการทำงานของโครงการ
- 3.1.4.4 ทดสอบการทำงานของโครงการ
- 3.1.4.5 ทำการเปรียบเทียบประสิทธิภาพของการใช้งานแต่ละแบบ
- 3.1.4.6 ทดสอบการทำงานของอุปกรณ์เมื่อมีการใช้งานกล้อง ESP32-CAM 2 ตัว
- 3.1.4.7 อัปเดตโมเดลสำหรับวิเคราะห์ป้ายทะเบียน
- 3.1.4.8 พัฒนาแอปพลิเคชันสำหรับควบคุมการทำงานของโรงรถ

3.1.5 สรุปผลการวิจัย เสนอแนะ และจัดทำรูปเล่ม

[illegible]

2.2 ศึกษาวิธีการใช้งาน Raspberry ในการเชื่อมต่อกับกล้องวงจรปิดหรือกล้อง ESP32-CAM												
2.3 ทดสอบการทำงานของกล้องวงจรปิดหรือกล้อง ESP32-CAM, Raspberry Pi, เซนเซอร์ และอุปกรณ์ IOT												
2.4 ทดลอง Train และ Test model												
3.การออกแบบระบบ												
3.1 ออกแบบโรงรถ												
3.2 ออกแบบระบบการทำงานของโรงรถ												
4.การพัฒนาและติดตั้งระบบ												
4.1 เชื่อมต่อ Raspberry Pi กับกล้อง หลอดไฟ และเซนเซอร์ต่างๆ												
4.2 สร้างแอปพลิเคชันสำหรับสื่อสารรับ-ส่ง ข้อมูลระหว่างอุปกรณ์และผู้ใช้												
4.3 สร้างแบบจำลองสำหรับใช้ทดสอบการทำงาน ของโครงการ												
4.4 ทดสอบการทำงานของโครงงาน												
4.5 ทำการเปรียบเทียบประสิทธิภาพของการ ใช้งานแต่ละแบบ												

4.6 ทดสอบการทำงานของอุปกรณ์เมื่อมีการใช้งานกล้อง ESP32-CAM พร้อมกัน 2 ตัว												
4.7 อัปเดตโมเดลสำหรับวิเคราะห์ป้ายทะเบียน												
4.8 พัฒนาแอปพลิเคชันสำหรับควบคุมการทำงานของโรงรถ												
4.9 ทดสอบการทำงานของแอปพลิเคชันเมื่อใช้งานร่วมกับแบบจำลอง												
4.10 หาข้อผิดพลาดและแก้ไขปรับปรุง												
4.11 จัดทำเอกสารการติดตั้งระบบและคู่มือการใช้งาน												
5.สรุปผลการวิจัย เสนอแนะ และจัดทำรูปเล่ม												

3.3 อุปกรณ์และเครื่องมือที่ใช้

3.3.1 ฮาร์ดแวร์

3.3.1.1 คอมพิวเตอร์

3.3.1.2 Torque Motor

3.3.1.3 แผ่นอะคริลิก

3.3.1.4 ESP32-CAM

3.3.1.5 ESP8266

3.3.1.6 Sensor วัดระยะ

3.3.1.7 หลอดไฟ

3.3.2 ซอฟต์แวร์

3.3.1.1 Jupyter Notebook

3.3.1.2 Colab Notebook

3.3.1.3 Arduino

3.3.1.4 Android Studio

3.4 ชุดข้อมูล (Dataset set) ที่ใช้ในโครงงาน

เป็นโมเดลที่ถูก Train มาแล้วของ Tesseract ที่มีการระบุรายละเอียดคร่าวๆเกี่ยวกับชุดข้อมูลว่า ในภาษาละตินใช้ข้อมูลข้อความที่เป็นบรรทัด ประมาณ 400000 บรรทัด และมีรูปแบบฟอนต์ ประมาณ 4500 ฟอนต์ โดยในภาษาอื่น ๆ ชุดข้อมูลที่ใช้ในการ Train มีจำนวนที่น้อยกว่าที่กล่าว มาข้างต้น แต่ก็มีปริมาณที่ใกล้เคียง โดยใช้โมเดล tha.traineddata โดยเลือกใช้ตัว tessdata_best ซึ่งมีผลลัพธ์ที่ดีที่สุดจากข้อมูลการประเมินของ Google เข้ากว่าตัวธรรมดา

3.5 การออกแบบระบบ

ในส่วนแรกนั้นคือการใช้กล้อง ESP32-CAM บันทึกภาพหน้ารถ เมื่อมีรถหรือวัตถุเข้าใกล้เซ็นเซอร์ในระยะที่กำหนด

ในส่วนที่สองนั้นจะรับภาพจากกล้อง ESP32-CAM มาทำการประมวลผลภาพและทำการแยกองค์ประกอบของภาพเพื่อตรวจสอบรถตามด้วยป้ายทะเบียนรถตรงกับที่ลงทะเบียนไว้หรือไม่

ในส่วนที่สามจะแยกออกเป็นสองส่วนคือส่วนแรกป้ายทะเบียนรถตรงกับที่ลงทะเบียนไว้ก็จะทำการเปิดประตูโรงรถและปิดเมื่อรถจอดอยู่ในตำแหน่งที่เหมาะสม ในส่วนที่สอง คือป้ายทะเบียนรถไม่ตรงกับที่ลงทะเบียนไว้จะทำการแจ้งเตือนไปให้ผู้รู้และภาพของรถที่มาจอด

ในส่วนที่สี่ จะเป็นส่วนของแอปพลิเคชัน Smart Garage ที่ใช้ในการสั่งการทำงานของโรงรถไม่ว่าจะเป็นเปิดปิดประตูหรือลงทะเบียนป้ายทะเบียนรถที่จะไว้ใช้ตรวจสอบ



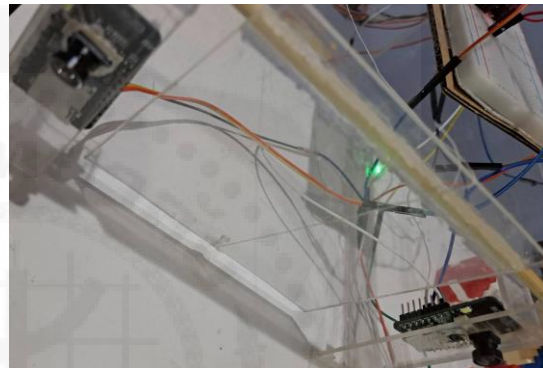
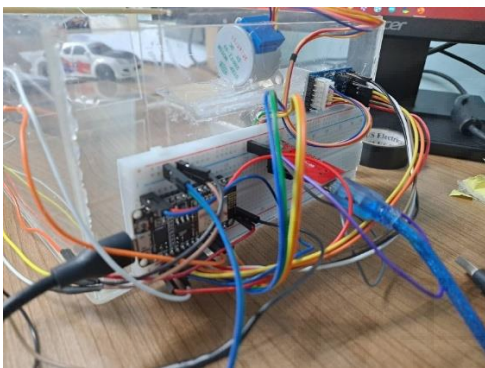
รูปที่ 8 หลักการทำงานของโครงการ

บทที่ 4

ผลการดำเนินโครงการ

4.1 กล้อง ESP32-Cam สามารถจับภาพและส่งข้อมูลภาพขึ้นไปแสดงผลบนเว็บได้

4.2 กล้อง ESP32-Cam สามารถใช้งานร่วมกับ ESP8266 เพื่อใช้ในการเชื่อมต่อและใช้งานอุปกรณ์ร่วมกันกับเซนเซอร์ต่างๆ



รูปที่ 9 การเชื่อมต่ออุปกรณ์

4.3 กล้อง ESP32-Cam สามารถติดต่อสื่อสารกับ python ได้

4.3.1 กล้อง ESP32-Cam สามารถส่ง MQTT message ไปให้ python รับได้

4.3.2 เมื่อ python ได้รับ MQTT message จากอุปกรณ์แล้วจะทำการเซฟภาพจากเว็บ

```
links = []
for img in imgs:
    link = img.get('src')
    if 'http://' not in link :
        link = url + link
    links.append(link)

for i in range(len(links)):
    filename = 'img{}.png'.format(i)
    urllib.request.urlretrieve(links[i],filename)

    print("Save Image")

    client.publish("Opendoor","open")

    print("Opendoor")

client = mqtt.Client()
client.on_connect = on_connect
client.on_message = on_message

client.connect("broker.mqtt-dashboard.com", 1883, 60)
client.loop_forever()

... Connected with result code 0
Save Image
Opendoor
```

รูปที่ 10 การบันทึกรูปภาพและส่ง MQTT message

4.3.3 python ส่ง MQTT message กลับไปที่กล้อง ESP32-Cam เพื่อให้ไค์ดบน Arduino สั่งให้อุปกรณ์
ตัวอื่นทำงานต่อ

```
Publish message: detect  
Message arrived [Opendoor] open
```

☒ Autoscroll ☐ Show timestamp

Newline

115200 baud

Clear output

รูปที่ 11 การรับ message และทำงานต่อ

4.4 เปลี่ยน Module กล้อง esp32-cam เป็นกล้องเลนส์กว้าง 160 องศา 2 ล้านพิกเซลเพื่อเพิ่ม
ความคมชัดและประสิทธิภาพการตรวจจับป้ายทะเบียน



รูปที่ 12 ตัวอย่างการรูปตัวอย่างหลังเปลี่ยน Module กล้อง esp32-cam

4.5 ตรวจสอบป้ายทะเบียนจากรูปถ่ายกล้องเดี่ยวและแปลงตัวอักษรจากทะเบียน

ตรวจสอบว่าในรูปภาพสามารถตรวจจับหรือไม่จากรูปภาพที่ถ่ายจากกล้อง ESP32-CAM



รูปที่ 13 ตัวอย่างการตรวจจับวัตถุที่เป็นรถจากรูปตัวอย่าง

4.5.1 หากตรวจพบรถจะทำการตรวจจับป้ายทะเบียนรถ



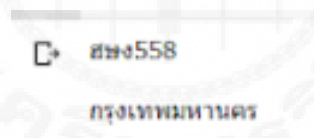
รูปที่ 14 ตัวอย่างการตรวจจับป้ายจากรูปตัวอย่าง

4.5.2 ทำการแบ่งส่วนที่เป็นเลขทะเบียนและจังหวัดของป้ายทะเบียนรถ



รูปที่ 15 ตัวอย่างการแบ่งเลขทะเบียนและจังหวัดของป้ายทะเบียนรถแล้ว

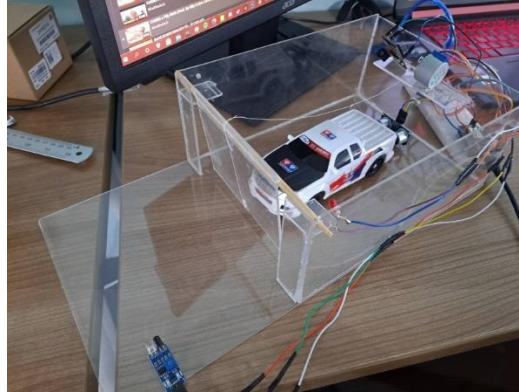
4.5.3 แปลงตัวอักษรจากรูปภาพที่ตรวจจับมาได้ โดยใช้โมเดลของ Tesseract



รูปที่ 16 ตัวอย่างการแปลงอักษรจากรูปภาพหลังตรวจจับทะเบียนแล้ว

4.6 ทำแบบจำลองโรงรถสำหรับทดสอบการทำงานของโครงงาน

4.6.1 ใช้แผ่นอะคริลิกใสสร้างเป็นห้องและติดตั้งอุปกรณ์เข้ากับแบบจำลอง



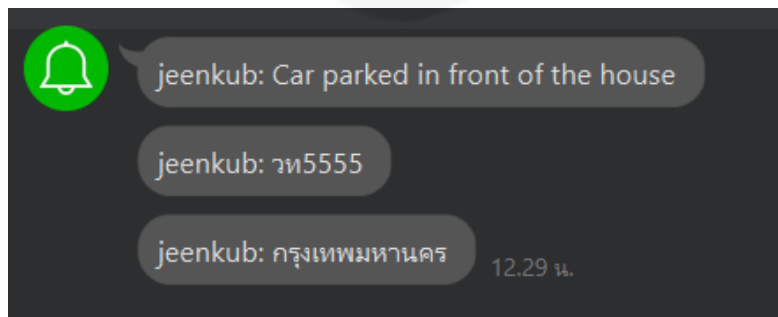
รูปที่ 17 รูปแบบจำลองโรงรถ

4.6.3 ทดสอบการทำงานของประตูโรงรถโดยการใช้อุปกรณ์ดึงเส้นเอ็นที่ผูกไว้กับประตู



รูปที่ 18 ตัวอย่างการทำงานการเปิดประตูโรงรถ

4.6.4 ทดสอบการส่งข้อความแจ้งเตือนผ่านไลน์เมื่อมีรถมาจอด



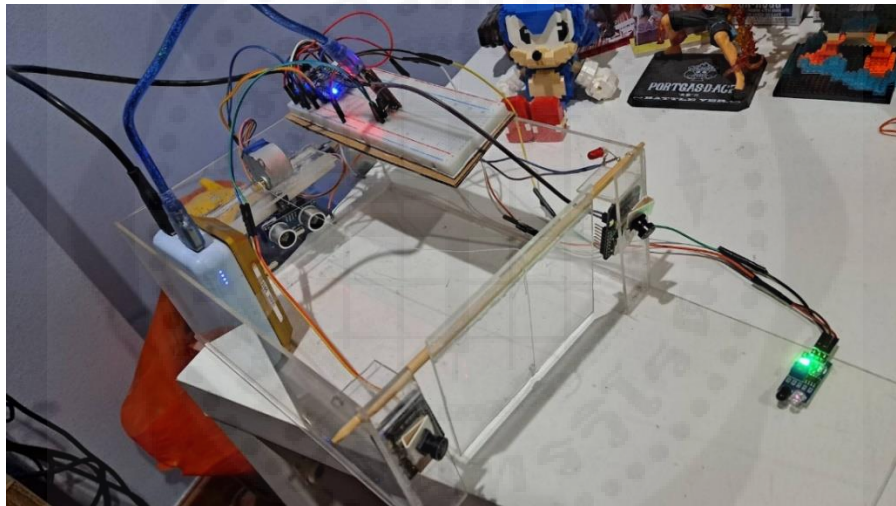
รูปที่ 19 ตัวอย่างการทดสอบส่งข้อความแจ้งเตือนผ่านไลน์

4.6.5 ทดลองนำป้ายทะเบียน มาตรวจจับและเมื่อป้ายทะเบียนถูกต้องให้มอเตอร์ทำงานเพื่อเปิดประตู



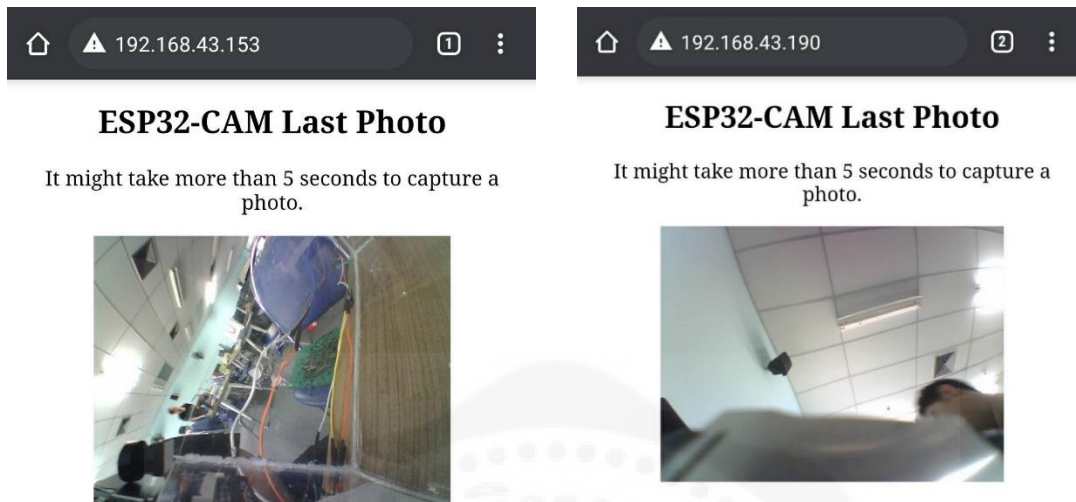
รูปที่ 20 ตัวอย่างการทดสอบการตรวจจับป้ายทะเบียน และผลลัพธ์บน Terminal

4.6.6 ปรับแต่งรูปแบบของแบบจำลอง สำหรับใช้งานกล้อง esp32-cam 2 ตัว



รูปที่ 21 แบบจำลองโรงรถ

4.7 เพิ่มการใช้งานกล้อง esp32-cam เป็นสองตัว โดยให้แสดงผลผ่านคนละ IP



รูปที่ 22 ตัวอย่างผลลัพธ์ของภาพที่ถ่ายได้จากกล้อง 2 ตัว

เปรียบเทียบการทำงาน ระหว่าง ใช้กล้อง 1 ตัว และ 2 ตัว จากการทำการทดลอง 1 ครั้ง

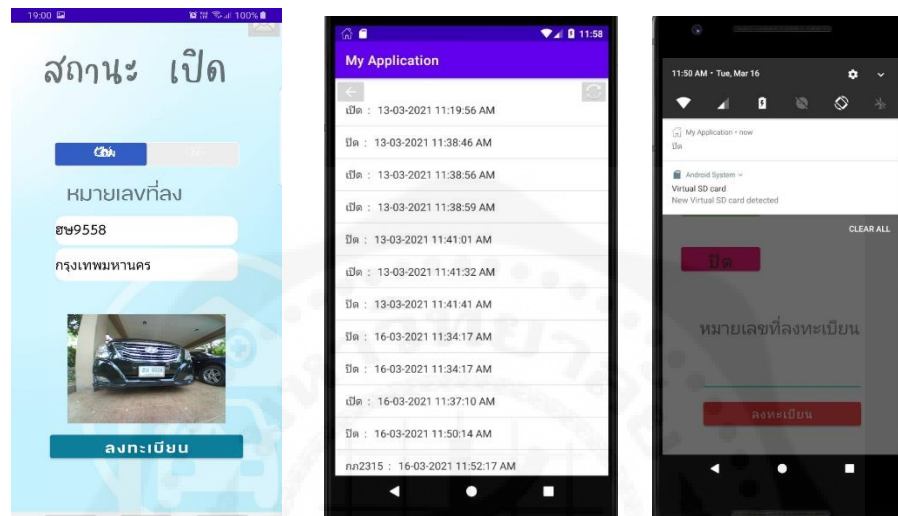
จำนวน	เวลาที่ใช้ทำงาน	ความถูกต้อง
1	10.50283	ไม่ตรง
2	22.7035	ไม่ตรง

4.8 จัดทำแอปพลิเคชันมือถือผ่าน Android Studio สำหรับควบคุม และลงทะเบียนเลขทะเบียนรถ

4.8.1 พัฒนาแอปพลิเคชันให้สามารถลงทะเบียนได้ทั้งเลขทะเบียนและจังหวัด

4.8.2 เพิ่มให้สามารถ Upload รูปรถพร้อมป้ายทะเบียน

4.8.3 เพิ่ม Function ให้สามารถควบคุมการทำงานของประตูโรงรถได้ผ่านแอปพลิเคชัน Smart Garage



รูปที่ 23 ตัวอย่างการทำงานบนแอปพลิเคชัน Smart Garage

บทที่ 5

สรุปผล อภิปรายผล และข้อเสนอแนะ

5.1 สรุปผลการศึกษาค้นคว้า

จากการศึกษาค้นคว้าการใช้งานอุปกรณ์ต่างๆร่วมกับแอปพลิเคชัน

ในส่วนของการใช้งานอุปกรณ์ IoT พบว่าอุปกรณ์และเซ็นเซอร์ต่างๆสามารถใช้งานร่วมกันได้สำเร็จ มีการติดต่อสื่อสารกันระหว่างอุปกรณ์ทั้งทางกายภาพ และผ่านเครือข่ายสัญญาณหรือ Wifi กล้อง ESP32-CAM สามารถใช้ถ่ายภาพได้สำเร็จและสามารถทำงานได้อย่างต่อเนื่อง เมื่อนำภาพที่ถ่ายไว้ไปตรวจจับและวิเคราะห์ป้ายทะเบียนสำเร็จ ก็สามารถสั่งให้อุปกรณ์ตัวอื่นทำงานต่อในลำดับถัดไปได้ และยังสามารถใช้งาน ESP32-CAM จำนวน 2 ตัวร่วมกันได้

ในส่วนของการตรวจจับป้ายทะเบียนรถ สามารถแยกแยะได้ว่าวัตถุที่อยู่ในรัศมีการตรวจจับนั้นเป็นรถจริงแล้วจึงจะนำไปตรวจจับป้ายทะเบียนในลำดับถัดไป ซึ่งสามารถตรวจจับได้จากภาพที่ได้มาทำการหาขอบของภาพที่เป็นส่วนป้ายทะเบียนรถจากรูปภาพ และนำรูปป้ายทะเบียนที่จับได้มาทำการบิดภาพให้เป็นทรงสี่เหลี่ยมเพื่อให้ง่ายต่อการแบ่งส่วนที่เป็นเลขทะเบียนกับจังหวัดของป้ายทะเบียนรถนั้น แล้วทำการแปลงตัวอักษรจากรูปภาพให้กลายเป็นตัวอักษรออกมาโดยใช้ Tesseract โดยสามารถแปลงออกมาได้ตรงกับตัวอักษรที่อยู่บนป้าย และสามารถประมวลผลความถูกต้องของป้ายทะเบียนเพื่อนำไปทำงานต่อบน Arduino IDE และแจ้งเตือนบนแอปพลิเคชันได้

และในส่วนของการทำงานบนแอปพลิเคชันมีการเพิ่มระบบสำหรับลงทะเบียนป้ายทะเบียนรถของผู้ใช้ พร้อมกับการอัปเดตฐานข้อมูลของผู้ใช้เพื่อนำไปใช้ในการวิเคราะห์ในแต่ละครั้ง และยังสามารถควบคุมการทำงานของประตูโรงรถได้ผ่านทางแอปพลิเคชันโดยตรง

5.2 อภิปรายผล

จากการดำเนินโครงการในส่วนของการใช้งานอุปกรณ์ IoT อุปกรณ์ต่างๆสามารถทำงานร่วมกันได้ แต่อาจเกิดปัญหาในกรณีที่จ่ายไฟให้อุปกรณ์ไม่พอ ซึ่งสามารถเกิดจากมีการใช้อุปกรณ์มากกว่า 2 ตัวขึ้นไป อาจมีตัวใดตัวหนึ่งที่ใช้มากกว่าหรืออาจจะเท่ากันก็ได้ แต่แหล่งจ่ายไฟอาจมีปริมาณไฟไม่เพียงพอที่จะครอบคลุมอุปกรณ์ได้ทั้งหมด ซึ่งอาจทำให้ผลลัพธ์การทำงานของอุปกรณ์บางตัวผิดไปจากที่ควรเป็น หรืออาจไม่ทำงาน เราจึงเพิ่มแหล่งจ่ายไฟด้วยแบตเตอรี่สำรอง (Powerbank) และแบ่งการใช้งานของอุปกรณ์ต่างๆให้เหมาะสมกับแหล่งจ่ายไฟ ทำให้อุปกรณ์แต่ละตัวสามารถทำงานได้อย่างปกติ

ในส่วนของการพัฒนาโมเดลในการตรวจจับและวิเคราะห์ป้ายทะเบียน จะพบปัญหาว่าการตรวจจับป้ายจากภาพตัวอย่างนั้นจะมีปัญหาในบางรูป เนื่องจากการตรวจจับในแต่ละครั้งจะตรวจจับขอบหรือกรอบของป้ายทะเบียน ซึ่งในกรณีที่บางตัวอย่างมีองค์ประกอบอื่นที่มีขอบหรือมองเห็นเป็นรูปทรงอย่างชัดเจนจะทำให้โมเดลไปตรวจจับสิ่งนั้นแทน เช่น ป้ายจราจร ป้ายโฆษณา เป็นต้น และในส่วนของการวิเคราะห์ป้ายทะเบียน เนื่องจากเราทำการการวิเคราะห์ป้ายทะเบียนโดยการใช้ Tesseract สามารถแปลงตัวอักษรภาษาไทยออกมาได้หากรูปภาพมีความชัดเจน และมีขนาดที่พอดี แต่หากมีขนาดเล็กเกินไปหรือภาพชัดเจนไม่มากพอ อาจส่งผลให้การแปลงรูปเป็นภาษานั้นมีความผิดพลาดได้

5.3 ข้อเสนอแนะ

ถ้านำไปพัฒนาต่อ การเปลี่ยนกล้องให้เป็นรูปแบบที่ดีกว่าจะเพิ่มประสิทธิภาพการทำงานของโครงงานได้เป็นอย่างมาก และสามารถปรับเปลี่ยนรูปแบบการทำงานของโครงงานให้อยู่ในรูปแบบที่ผู้ใช้สามารถใช้งานได้ง่ายขึ้น สามารถนำไปพัฒนาเพื่อเพิ่มฟังก์ชันต่างๆบนแอปพลิเคชันได้ พัฒนาโมเดลเพื่อใช้ในการแสดงผล และพัฒนาประสิทธิภาพของโมเดลให้ทำงานได้เร็วขึ้นโดยการปรับเปลี่ยน source code หรือใช้โมเดลที่จะสามารถทำให้ตรวจจับได้เร็วยิ่งขึ้น

บรรณานุกรม

- Lachi Agnew. How to use OpenCV Computer Vision [ออนไลน์]. 2019, <https://medium.com/seed-digital/lets-sort-some-m-ms-using-computer-vision-c27cb5e96d83> [28 พฤษภาคม 2563]
- Syed Zain Masood, Guang Shu, Afshin Dehghan and Enrique G. Ortiz. License Plate Detection and Recognition Using Deeply Learned Convolutional Neural Networks [ออนไลน์]. 2017, <https://paperswithcode.com/paper/license-plate-detection-and-recognition-using> [28 พฤษภาคม 2563]
- Naveenkumar Mahamkali. OpenCV for Computer Vision Applications [ออนไลน์]. 2015 , https://www.researchgate.net/publication/301590571_OpenCV_for_Computer_Vision_Applications [30 พฤษภาคม 2563]
- Vladimir Vujovic, Branko Perisic, Nikola Davidovic. Raspberry Pi as Internet of Things hardware: Performances and Constraints [ออนไลน์]. 2014, https://www.researchgate.net/publication/272175660_Raspberry_Pi_as_Internet_of_Things_hardware_Performances_and_Constraints [30 พฤษภาคม 2563]
- Akshay Sinha. Getting Started with image Processing using Python [ออนไลน์]. 2019, <https://medium.com/analytics-vidhya/getting-started-with-image-processing-using-python-c027fbdd4608> [11 มิถุนายน 2563]
- Edwin Nyawoli. Tesseract OCR [ออนไลน์]. 2020, <https://github.com/tesseract-ocr/tesseract/blob/master/README.md> [10 กรกฎาคม 2020]
- Frank Hofmann. Levenstein Distance and Text Similarity in Python [ออนไลน์]. 2018, <https://stackabuse.com/levenshtein-distance-and-text-similarity-in-python/> [18 ม.ค. 2561]
- Geometric Transformation of Image [ออนไลน์]. 2016, [Geometric Transformations of Images - Morange Wiki](#) [3 สิงหาคม 2559]
- Moses Olafenwa, John Olafenwa. Official English Documentation for ImageAI [ออนไลน์]. 2021, <https://imageai.readthedocs.io/en/latest/> [5 มกราคม 2564]
- Trained data files for Version 4.00+ (tessdoc) [ออนไลน์], 2017, <https://tesseract-ocr.github.io/tessdoc/Data-Files.html> [กันยายน 2560]