**ROS2 Object Detection and Tracking in field Autonomous Vehicles**  
การทำ Objective detection ด้วย ROS2 สำหรับการใช้งานรถยนต์อัตโนมัติ

Weerapat Supapornopas 63340500061

# Problem Statement and Introduction

การใช้งานระบบการติดตามการเคลื่อนไหวของวัตถุเป็นสิ่งสำคัญในหลายแอปพลิเคชัน เช่น การติดตามการเคลื่อนไหวของรถยนต์ หรือการติดตามการเคลื่อนไหวของผู้คนในระบบขับขี่ไร้คนขับ โดยผ่านการใช้ Computer Vision ซึ่งสามารถช่วยในการระบุตำแหน่งและชนิดของวัตถุได้ อย่างไรก็ตามการระบุตำแหน่งและชนิดของวัตถุยังไม่ได้รับความพัฒนาเท่าที่ควรใน ROS 2 (Robot Operating System 2) โดยที่ ROS2 ถูกการใช้งานอย่างกว้างขวางในงานที่เกี่ยวข้องกับหุ่นยนต์และระบบอัตโนมัติ

ดังนั้นเพื่อเพิ่ม platform ของระบบการตรวจจับวัตถุ จึงเลือกการทำการ implement Object detection มาใช้กันกับ Ros2 เพื่อให้สามารถระบุตำแหน่งและชนิดของวัตถุใน ROS 2 เพื่อไปพัฒนาในระบบรถยนต์ไร้ขนคับ เพื่อให้สามารถระบุตำแหน่งและชนิดของวัตถุได้อย่างแม่นยำและสามารถใช้งานควบคู้กับ Framework ที่มีอยู่มากมายใน ROS 2

# Background Study

1. ROS2



ROS2 หรือ Robot Operating System 2 เป็น framework ที่ถูกออกแบบมาเพื่อใช้ในการพัฒนาระบบหุ่นยนต์และระบบอัตโนมัติ โดยมีความยืดหยุ่นในการทำงานเป็นจุดเด่นที่สำคัญของ ROS2 ซึ่งสามารถทำงานร่วมกับหลายฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในหุ่นยนต์ได้ นอกจากนี้ ROS2 ยังมีความสามารถในการสื่อสารและจัดการข้อมูลในระบบหุ่นยนต์ได้อย่างมีประสิทธิภาพ

ROS2 มีโครงสร้างแบ่งออกเป็น module ที่เรียกว่า nodes ซึ่งทำให้สามารถทำงานแบบ distributed ได้ นั่นคือการทำงานใน ROS2 จะเป็นการแบ่งงานออกเป็นหลายๆ ส่วน และทำงานในแต่ละส่วนนั้นสามารถทำงานบนคอมพิวเตอร์หรืออุปกรณ์ต่างๆ ที่เชื่อมต่อกันผ่านทางเครือข่ายได้ โดยมีการใช้โปรโตคอล ROS2 ในการแลกเปลี่ยนข้อมูลระหว่าง nodes

นอกจากนี้ ROS2 ยังมีความยืดหยุ่นในการทำงานภายใน node ด้วยการใช้ตัวกลางที่เรียกว่า ROS2 middleware ซึ่งช่วยในการจัดการการสื่อสารและข้อมูลในระบบ โดย ROS2 middleware จะช่วยในการเชื่อมต่อหรือเปลี่ยนแปลงฮาร์ดแวร์และซอฟต์แวร์ต่างๆ ที่ใช้ในหุ่นยนต์ได้อย่างมีประสิทธิภาพ นอกจากนี้ ROS2 ยังสามารถรองรับการใช้งานหลายภาษาและรองรับการทำงานบนหลายแพลตฟอร์ม ซึ่งทำให้ ROS2 เป็นเครื่องมือที่ใช้งานได้หลากหลายและมีความยืดหยุ่นสูงในการพัฒนาระบบหุ่นยนต์และระบบอัตโนมัติ

นอกจากนี้ ROS2 ยังมีการพัฒนาต่อยอดอยู่เรื่อยๆ โดยมีการเพิ่มฟีเจอร์ใหม่ๆ เพื่อตอบสนองความต้องการของผู้ใช้งานในสายงานหุ่นยนต์และระบบอัตโนมัติ โดย ROS2 ยังมีการใช้งานในโครงการหลายๆ โครงการที่เกี่ยวข้องกับหุ่นยนต์และระบบอัตโนมัติ เช่น โครงการ ROS-Industrial ที่ใช้งานในการพัฒนาระบบการผลิตและจัดการโรงงาน นอกจากนี้ยังมีการใช้งาน ROS2 ในการพัฒนาหุ่นยนต์ที่ใช้ในงานวิจัยและการศึกษาด้วยเช่นกัน

1. Open-CV

Icon

Description automatically generated

OpenCV (Open Source Computer Vision Library) เป็นไลบรารี Open souce เพื่อการประมวลผลภาพและวิดีโอ รวมถึงฟังก์ชันการทำความเข้าใจภาพ เช่น การตรวจจับวัตถุ การจัดการกับภาพและวิดีโอ การแยกแยะวัตถุ การหาและวิเคราะห์ลักษณะของภาพ และอื่นๆ OpenCV

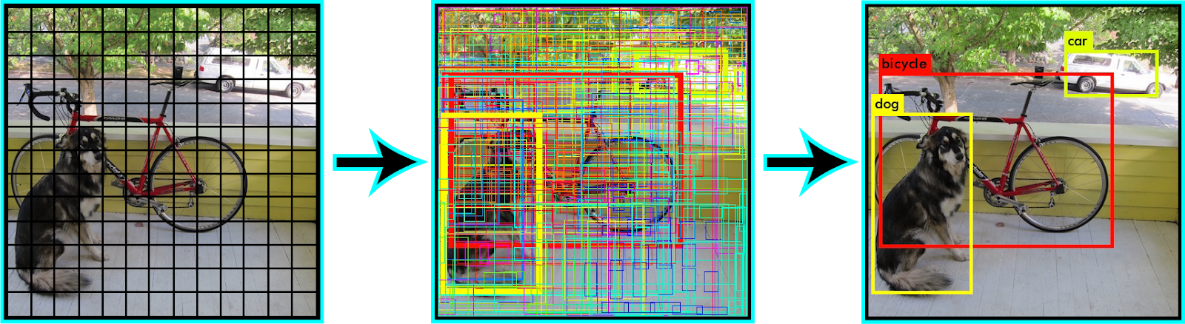
ถูกพัฒนาโดยใช้ภาษา C++ และสามารถรองรับการใช้งานบนหลายแพลตฟอร์ม เช่น Windows, Linux, macOS, iOS, Android และ Raspberry Pi เป็นต้น และยังสามารถเชื่อมต่อกับกล้องและอุปกรณ์อื่นๆ เพื่อดึงภาพมาใช้งานในการประมวลผลภาพได้อย่างง่าย

สำหรับฟังก์ชันการประมวลผลภาพที่ OpenCV สามารถทำได้ สามารถแบ่งได้เป็นหลายๆ ประเภท เช่น

* การจัดการกับภาพ : เช่น การปรับขนาดภาพ, การปรับความคมชัดภาพ, การตัดภาพ, การแปลงรูปแบบสี
* การตรวจจับวัตถุ : เช่น การตรวจจับใบหน้า, การตรวจจับตำแหน่งและขนาดของวัตถุ, การตรวจจับคน
* การวิเคราะห์ภาพ : เช่น การวิเคราะห์รูปแบบการเคลื่อนไหวของวัตถุ, การวิเคราะห์ภาพการแสดงอารมณ์, การวิเคราะห์ภาพสไตล์เสื้อผ้า

นอกจากนี้ OpenCV ยังสามารถเชื่อมต่อกับกล้องและอุปกรณ์อื่นๆ ได้อย่างสะดวกสบาย เพื่อดึงภาพมาใช้งานในการประมวลผลภาพได้อย่างง่ายดาย ทำให้สามารถนำไปประยุกต์ใช้งานในหลากหลายอุตสาหกรรมได้ เช่น หุ่นยนต์, การวิเคราะห์ภาพแพทย์, การตรวจสอบความถูกต้องของผลิตภัณฑ์, การตรวจสอบคุณภาพ, และอื่นๆ

1. YOLO



YOLO (You Only Look Once) เป็นโมเดลการตรวจจับวัตถุ (object detection) ที่ออกแบบมาให้มีประสิทธิภาพสูงและความเร็วสูงในการทำงาน เป็นโมเดลที่นิยมใช้ในงาน Computer Vision โดยเฉพาะในงานตรวจจับวัตถุในวิดีโอ (video object detection)

YOLO มีจุดเด่นคือการทำงานโดยการแบ่งภาพเป็นเส้นตารางหรือตาราง (grid cell) แล้วสร้าง bounding box โดยใช้โมเดล regression เพื่อหาพารามิเตอร์ของกล่องขอบเขตนั้น พร้อมทั้งกำหนดค่าความเชื่อมั่น (confidence score) ว่าวัตถุที่ตรวจจับเป็นวัตถุประเภทใด และตำแหน่งของวัตถุนั้นอยู่ที่ใดในภาพ

โดย YOLO แบ่งพื้นที่ของภาพเป็นส่วนเล็ก ๆ แต่มีจำนวนมาก แล้วทำการตรวจจับวัตถุบนแต่ละพื้นที่ดังกล่าว ดังนั้นการทำงานของ YOLO จึงมีความเร็วสูงและสามารถตรวจจับวัตถุได้หลายวัตถุในภาพเดียว อีกทั้ง YOLO ยังสามารถจำแนกวัตถุที่อยู่ใกล้เคียงกันได้ดีเนื่องจากใช้เทคนิค non-max suppression เพื่อลดจำนวน bounding box ที่ไม่จำเป็นออกจากภาพ นอกจากนี้ YOLO ยังสามารถใช้งานได้ง่ายและมีการตั้งค่าที่หลากหลายให้กับผู้ใช้งาน

1. R-CNN

Diagram

Description automatically generated

R-CNN (Region-based Convolutional Neural Network) เป็นโมเดลที่ออกแบบมาเพื่อใช้ในการตรวจจับวัตถุในรูปภาพ โดยใช้หลักการของ Deep Learning และ Convolutional Neural Network (CNN) เป็นหลัก โดย R-CNN นั้นเป็นการสร้างสองส่วนหลักๆ คือการทำ feature extraction จากรูปภาพและการคัดเลือกพื้นที่ที่มีวัตถุอยู่ (region proposal) เพื่อนำไปใช้ในการจำแนกวัตถุ

ส่วนการสกัดคุณลักษณะของ R-CNN นั้นใช้โมเดล CNN เพื่อสกัดคุณลักษณะจากรูปภาพ โดยเราสามารถใช้โมเดล CNN ที่ได้รับการสร้างแล้ว (pre-trained model) เช่น VGG16 หรือ ResNet50 มาใช้งานได้เลย แต่ใน R-CNN เราจะต้องนำโมเดล CNN มาทำงานกับแต่ละพื้นที่ที่ต้องการตรวจจับวัตถุ ซึ่งใน R-CNN เราจะนำภาพที่ต้องการตรวจจับวัตถุมาแบ่งออกเป็นพื้นที่หลายๆ พื้นที่โดยใช้วิธีการสร้างพื้นที่เหล่านี้โดยใช้โปรแกรม Selective Search ซึ่งเป็นวิธีการคัดเลือกพื้นที่โดยใช้การรวมเพื่อเลือกส่วนที่น่าสนใจ

# Conceptual Design

ศึกษาตัว Model object detection ที่ใช้งานกันในปัจจุบันแล้วทำการไปเทสในท้องถนนเองเผื่อดูประสิทธิภาพของแต่ล่ะโมเดลและทำการ Implement ลงบน ROS2 เผื่อใช้ในการพัฒนาต่อไปกับระบบไร้คนขับ

1. ROS2 (Robot Operating System 2)
2. Software components:
   * Open-CV
   * PyTorch
   * YOLOV7
   * R-CNN

# Team, Task, Process

| **Mildstone** | **Task** | **Deliver Date** |
| --- | --- | --- |
| **Scrum 2** |  | **22/03/2023** |
| - ศึกษา performace ของแต่ละ model YOLOV7 และ R-CNN | 17/03/2023 |
| - เริ่มศึกษาการใช้งาน R-CNN ,YOLO V7 | 21/03/2023 |
| **Scrum 3** |  | **19/04/2023** |
| - ทำให้ ทั้ง 2 model นั้นสามารถทำงานได้ real time คอมพิวเตอร์ | 31/03/2023 |
| - เตรียม data set เองมาเทสตัว model | 07/04/2023 |
| - เลือก model สำหรับทำใน ROS2 | 18/04/2023 |
| **Scrum 4** |  | **10/05/2023** |
| - design system architecture ของแต่ล่ะ Node ใน ROS2 | 25/04/2023 |
| - ทำให้ model สามารถรันบน ROS2 ได้ | 02/05/2023 |
| - เก็บผลจากข้อมูลจริง | 09/05/2023 |
| **Final presentation** |  | **24/05/2023** |
| - final report | 16/05/2023 |
| - presentation | 23/05/2023 |

# Learning Plan

1. เรียนรู้การใช้งาน โมเดลในการตรวจจับวัตถุ โดย YOLO V7 และ R-Cnn ผ่าน Tutorial และ guide ใน Internet
2. ลองสร้าง data set ขึ้นมาเองแล้วนำไปทดสอบกับโมเดลทั้งสองเพื่อดู ประสิทธิภาพและ Accuarcy
3. นำ Model ที่เลือหมาปรับใช้กับ Ros2

# Process

* ศึกษา performace ของแต่ละ model YOLOV7 และ R-CNN
* ทำให้ ทั้ง 2 model นั้นสามารถทำงานได้ real time คอมพิวเตอร์
* เตรียม data set เองมาเทสตัว model และเลือก model สำหรับนำไปใช้ในระบบ ROS2
* design system architecture ของแต่ล่ะ Node ใน ROS2
* เทสตัว model ในRos2 แบบ realtime