TAIPEI TECH 國立臺北科技大學

Mechanical Engineering Department

Capstone Course Summary Thesis

Autonomous Vehicle Integration with Deep SORT Image Tracking

Class/Student ID/Name：

Mechanical Engineering 4B/109810022/陳勁旗

Mechanical Engineering 4A /109810006/王姿伶

Mechanical Engineering 4B /109810021/楊其康

Mechanical Engineering 4B /109810031/許喆堯

Mechanical Engineering 4A/109810001/林辰燁

Supervisor：Professor 許志明

Signature or Stamp of the Supervisor：___________

Submission Date：中華民國 112 年 11 月 1 日

Final vision？ ■Yes □No

Title page

1. Thesis Title：Autonomous Vehicle Intergration With Deep Sort Image Tracking

2. Authors：林辰燁/王姿伶/楊其康/陳勁旗/許喆堯

3. Graphics Contributors：陳勁旗/許喆堯

   Experiment Record Keepers：林辰燁/王姿伶/楊其康

   Supervisor：Professor 許志明

4. Assisting Supervisors：陳劭穎(Graduate Student)、陳禹衡(Graduate Student)

5. Project Location：Room 525-4, Integrated Science Building/ Laboratory; Room 328-1, Integrated Science Building/ Computer Classroom

6. Project duration：March 11, 2023, to October 10, 2023

7. Related Project Names

8. Other Information about this Thesis

Abstract

Thesis Title：Autonomous Vehicle Intergration With Deep Sort Image Tracking

Page Count：Fifty Pages

Department: Department of Mechanical Engineering, National Taipei University of Technology

Graduation Time：Academic Year 113, Second Semester

Degree：Bachelor

Class/Student ID/Name：

Mechanical Engineering 4B/109810022/陳勁旗

Mechanical Engineering 4A /109810006/王姿伶

Mechanical Engineering 4B /109810021/楊其康

Mechanical Engineering 4B /109810031/許喆堯

Mechanical Engineering 4A /109810001/林辰燁

Supervisor：Professor 許志明

Keywords：Image recognition、Pure visual dynamic analysis、Self-propelled vehicle guidance technology

Abstract

With the rapid advancement of technology and the outbreak of the pandemic, people have become more conscious of the need for epidemic prevention. Nowadays, there is a growing emphasis on contactless consumer behavior. This article will introduce the use of pure visual AGV (Automated Guided Vehicle) guidance technology combined with Deep SORT (Simple Online and Realtime Tracking) to achieve the goal of autonomous shopping cart following. This allows shoppers to free their hands from pushing shopping carts. When it comes to the checkout process after selecting items, we utilize visual dynamic analysis to automatically recognize and tally the products.

Acknowledgments

The successful completion of this project owes much to the support and guidance of several individuals whom we would like to express our heartfelt gratitude to.

First and foremost, we extend our deepest thanks to our project supervisor, who not only provided us with valuable ideas and resources but also allowed us ample room for creative expression. During our meetings, the supervisor offered encouragement and constructive suggestions, consistently supporting our ambitious ideas and providing feasible solutions. Throughout the project development, there was a conducive environment for discussion, fostering opportunities for self-exploration and cultivating the necessary attitude for self-directed learning.

We would also like to acknowledge the guidance and assistance of senior students, Chen Shao-Ying and Chen Yu-Heng. In the initial stages of our project, we were relatively inexperienced, and the senior students played a crucial role in guiding us through the control of self-propelled vehicles and basic operations of Autoware. Their expertise paved the way for a smooth start to our project. Additionally, when faced with challenges such as the autonomous vehicle's inability to navigate, the seniors diligently assisted us in troubleshooting errors, ultimately leading to successful autonomous vehicle usage.

A special thanks is extended to each team member for their dedication and commitment to the project. We also express gratitude to the team leader for their effective leadership, fostering a collaborative and brainstorming environment that was essential for accomplishing our mission.

Finally, we appreciate the practical training provided by the school's project course, which equipped us with skills in teamwork, literature review, idea generation, and handling unexpected situations. The two semesters of this project have been a valuable learning experience for us all.

Table Of Content

Table

Figure

Chapter 1 Introduction

1.1 Preface

Due to the scarcity of energy and manpower, automation is imperative. However, the application of autonomous vehicles is limited by various constraints, making them less common in daily life. Therefore, this experiment aims to introduce autonomous vehicles and image recognition technology into our daily lives, with the hope of significantly reducing consumer shopping time and allowing for an enjoyable shopping experience without the need for pushing a shopping cart. The main focus of this research project is to learn the construction of map paths and positioning, enabling autonomous vehicles to possess self-driving capabilities. Additionally, the Deep Sort algorithm is employed to allow the vehicle to follow specified individuals, utilizing Re-identification (RE-id) technology to re-identify the original target. Subsequent iterations involve personnel testing, parameter adjustments, and other optimizations to achieve the best walking performance.

1.2 Motivation

There are various existing Automated Guided Vehicle (AGV) guidance technologies, including common methods such as magnetic tapes [1], lasers [2], QR-Code guidance [3], etc. In this paper, we utilize the less common pure visual guidance technology [4]. The advantage of pure visual AGV guidance is that it does not require additional transmitters. However, this also introduces challenges, such as interference in crowded environments, leading to the AGV losing track of the target and encountering tracking errors [5]. Furthermore, during tracking, abrupt turns by the main target can cause it to go out of view and disappear, requiring the user to retrack. To address these issues, this paper adopts the pure pursuit path-following algorithm [6] and the tracking algorithm proposed in [7] to ensure that the shopping cart can stably follow specified individuals without being affected by other individuals. Moreover, in line with the current trend of promoting contactless transactions, there are many computer vision-based checkout methods [8]. However, most methods only improve the checkout experience and do not address the various changes in the complete shopping process. This paper employs YOLOv8 for object recognition, training the model to mark and recognize products. By utilizing an RGB camera and algorithms developed in Python for dynamic sequence analysis, the system can automatically determine whether a product is being taken out or put in when it enters the shopping cart, facilitating a hands-free shopping experience.

Chapter 2　Development Environment and Equipment Introduction

2.1 Environment Setup

In this project, Ubuntu 18.04 is employed as the operating system, and ROS Melodic is utilized as the foundational communication framework for controlling the autonomous vehicle.

2.1.1 Ubuntu

The chosen development environment for this project is Ubuntu, a Linux distribution. The decision to use Ubuntu as the development environment is driven by several reasons:：

1. **Free and Open Source**：Ubuntu is a free and open-source operating system, eliminating the need for licensing fees. This allows us to utilize a complete set of development tools and environments without incurring additional costs.
2. **Extensive Software Repository**：Ubuntu Software Center provides a wide array of development tools and programming languages, including Python, Java, C++, and more. Installing required tools is straightforward, and updates can be easily managed and maintained.
3. **Stability and Reliability**：Ubuntu boasts high stability and reliability, ensuring that system crashes or instability concerns are minimized.
4. **Open Source Community**：Ubuntu benefits from a vast open-source development community where support can be sought, problems resolved, and experiences shared with other developers.
5. **Rich Development Tools**:Ubuntu offers a variety of development tools, including Integrated Development Environments (IDEs), editors, debuggers, and version control tools such as Git.

Figure 2.1 ubuntu18.04 Operating Interface [10]

2.1.2 ROS Introduction

ROS (Robot Operating System) is a computer operating system framework specifically designed for software development in robotics. It is an open-source platform installed on the Linux environment. It serves as an open-source operating system, providing functionalities similar to an operating system, including hardware abstraction, management of low-level drivers, execution of shared functionalities, inter-process message passing, and package management for program distribution. ROS also offers tools and libraries for acquiring, creating, writing, and executing programs for multi-machine fusion. ROS is responsible for facilitating communication and operations among various components of a robot. Taking autonomous mobile robots as an example, these robots need to receive external stimuli, process signals from sensors through programs, and convert them into command formats for robot movement. ROS plays the role of a communication bridge in this process, transmitting commands to specified locations.The basic architecture of ROS is primarily composed of nodes and topics. Nodes are responsible for sending messages and are termed publishers, while nodes receiving messages are called subscribers. Each node does not necessarily serve only one role; a node can simultaneously act as both a publisher and a subscriber.

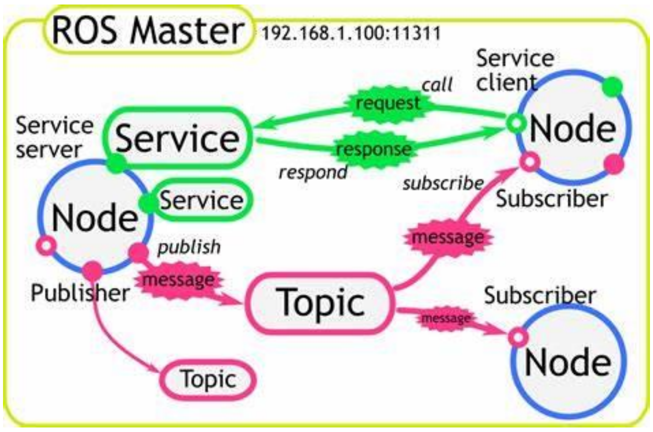

Figure 2.2 ROS system Node Signal Diagram [11]

- **Node**: A node is a process that performs computations. Robotic control systems typically consist of multiple nodes. For example, one node may control wheel movement, another may execute image recognition, another may handle localization, and yet another may be responsible for path planning.
- **Message**: Communication between nodes occurs through the exchange of messages. Each message adheres to a fixed data format. ROS supports primitive data structures such as integers, floats, booleans, strings, dictionaries, and lists. Messages can include arbitrarily nested structures and arrays.

- **Topic**: "Topic" is a crucial communication concept used to facilitate information exchange between robots and other systems. In ROS, a "topic" provides a simple and effective means to implement the publish-subscribe pattern for information dissemination, enabling different components to communicate asynchronously. A single "topic" may have multiple publishers and subscribers, and a single "node" can publish or subscribe to multiple "topics." Typically, publishers and subscribers are unaware of each other's existence. Each "node" has a name, and as long as they are of the correct type, any "node" can connect to the corresponding "topic" to send or receive messages. Information from "topics" can be sent or listened to by "nodes," which are referred to as "publishers" and "subscribers." This establishes the basis for message communication, allowing for a dynamic and flexible interaction between different components in the system.
- **Service**:The publish/subscribe model is an extremely flexible communication paradigm, but its many-to-many unidirectional transmission is not suitable for request/reply, which is often necessary in distributed systems. Request/reply is accomplished through services, defined by a pair of message structures: one for the request and one for the reply. Nodes offering services use names to provide these services, and clients utilize the service by sending a request message and waiting for a response.

2.2 Hardware Equipment Introduction

2.2.1 Agile X HUNTER

The HUNTER autonomous vehicle (Figure 2.3) is the chosen vehicle for our project. ， The high payload capacity of this vehicle, making it suitable for our project needs.

Driven Type	Rear-wheel drive
Steering Type	AGV
Maximum Speed	1.5m/s
Vehicle Weight	50kg
Payload	100kg
Minimum Turning Radius	1.7m

Table 1 Specifications of HUNTER Autonomous Vehicle Section

Figure 2. 1 HUNTER Autonomous Vehicle

2.2.2 Velodyne VLP-16

LiDAR (Light Detection and Ranging) utilizes optical sensing technology. LiDAR employs rotating pulses of laser light, forming a cylindrical coordinate axis, to measure various parameters of the target environment with high precision. In this project, we utilize the Velodyne VLP-16 LiDAR system (refer to Figure 2.4) within Autoware for recording maps and localization, enabling the vehicle to navigate autonomously.

Table 2　Velodyne VLP-16 規格

Number of Beams	16
Measurement Range	100 m
Range Accuracy	±30 mm
Vertical Field of View	±15.0°
Horizontal Field of View	360 °
Rotation Speed	5 Hz – 20 Hz

Figure 2. 2 Velodyne VLP-16

2.2.3 Intel L515

The Intel L515 is a solid-state LiDAR camera introduced by Intel, utilizing proprietary MEMS mirror scanning technology to achieve higher scanning efficiency. Its power consumption during depth scanning is below 3.5W, making it an energy-efficient LiDAR camera. Additionally, its built-in visual processor minimizes motion blur during runtime, capturing faster dynamic scenes. Due to its capabilities in depth measurement, power efficiency, and stable frame capture, the L515 was chosen as the visual sensor for the experiments in this paper. Partial specifications are listed in Table 3.

Figure 2. 3 Intel L515

Table 3 Intel L515 Specifications

Actual size (radius * thickness)	61mm*26mm
Depth Applicable Range	0.25m~9m
Depth map pixels	1024*768
RGB colormap pixels	1920*1080
update rate	30fps

Chapter 3 Autonomous Navigation

3.1 Autoware

3.1.1 Introduction

Autoware is an open-source autonomous driving software platform based on ROS [9], running on the Linux and ROS framework. Autoware provides numerous functionalities and technologies in the field of autonomous driving, including perception, decision-making, and control. Furthermore, Autoware utilizes localization and map data to enable the vehicle to understand its location and navigate on the map. It can also model the recorded path following using data collected from sensors such as LiDAR and radar, employing various methods. Autoware allows users to test autonomous driving systems in a virtual environment (Rviz) for subsequent analysis.

3.1.2 Structure

The operational mechanism of Autoware is primarily divied into the following stages: Sensing, Computing, and Actuation.

- **Sensing**: Receiving messages from various sensors.
- **Computing:** Making control logic decisions based on sensor data.
  - Perception : Perceiving the surrounding environment and state of the autonomous vehicle, used for subsequent decision-making.
  - Decision: Using perceptual information to determine the state of the autonomous vehicle (e.g., proceed along the path, emergency evasion, braking, etc.).
  - Planning the path involves two main components:
    - Mission Planner：Utilizing specific algorithms (e.g., shortest path algorithm) to determine a reference path from the starting point to the destination.
    - Motion Planner：Dynamically calculating a walking trajectory within a certain distance based on the current state, primarily used to control the autonomous vehicle to travel along the path determined by the Mission Planner or to avoid obstacles.
- **Actuation:** Obtaining control commands under the state of the autonomous vehicle.

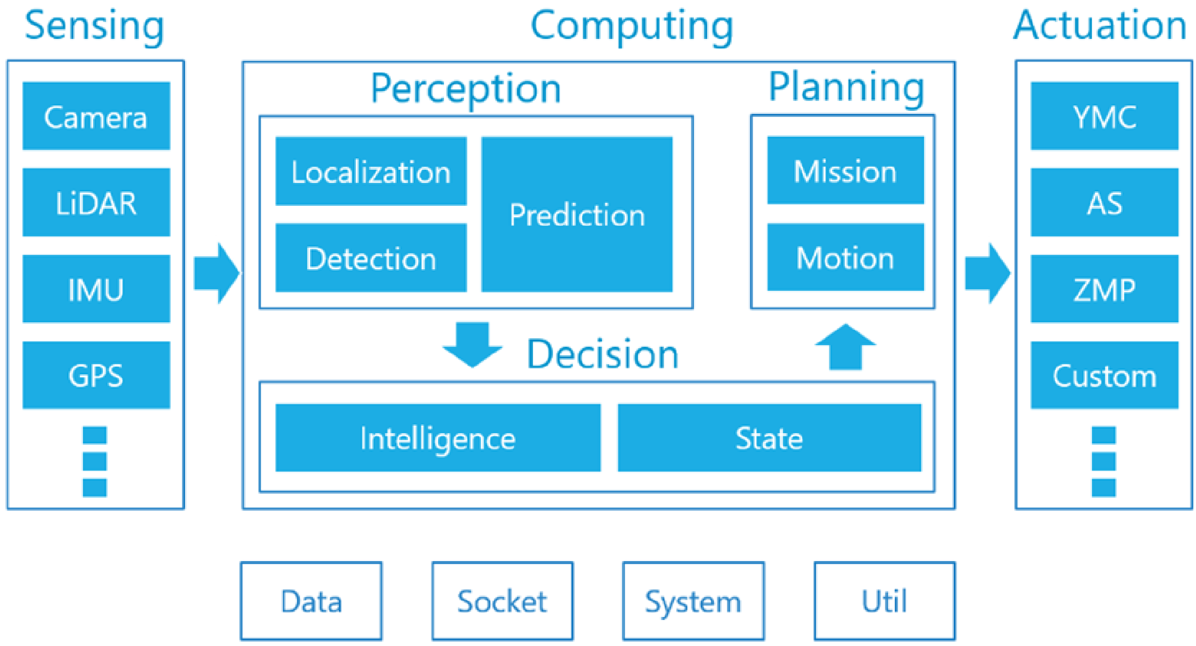

Figure 3. 1 Autoware structure diagram[12]

3.1.3 Experimental Process

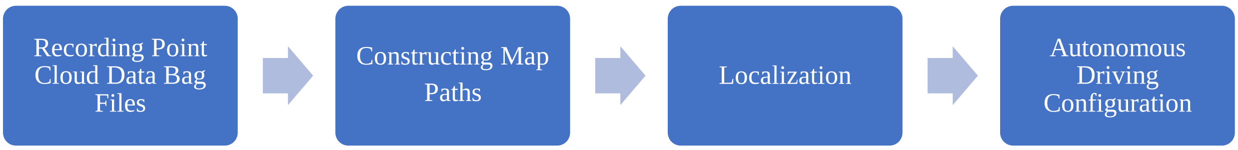

Figure 3. 2 Autonmous Driving Experiment Process Diagram

3.1.4 Modeling

Map file

The map file contains known information about the operation of the autonomous vehicle. Autoware primarily uses the following formats:

1. PCD (Point Cloud Data) Map

The PCD map provides data for the vehicle to perceive its surroundings. LiDAR scans the scene, and Normal Distributions Transform (NDT) is used for localization. Simultaneously, the scene objects are recorded in the form of a Point Cloud. The NDT_mapping process generates a PCD map file based on the car's GPS position and the point cloud obtained by LiDAR scanning.

2. Waypoint

The format for the path traveled by an autonomous vehicle, where the points on the path record speed information and direction. In the field of autonomous driving, waypoints are typically used to plan the vehicle's route, ensuring that the vehicle can reach its destination safely and efficiently. During operation, the autonomous vehicle dynamically generates waypoints based on the Vector map to perform avoidance maneuvers. Users record the movement trajectory from the previous driving as waypoints, and the next time waypoints are loaded, the vehicle can reproduce the previous walking path.

Perception

Lidar_localizer:

- Using Normal Distributions Transform (NDT) Algorithm。
- Principle: Utilizing real-time point cloud data from LiDAR, the NDT algorithm compares it with the recorded point cloud on the PCD map. Through Gaussian distribution estimation, the autonomous vehicle's position is determined.

Planning

1. Mission planner

- Navigation through vector map ：The Mission Planner plans based on the vector map, similar to navigation on Google Maps.
- Navigation through waypoints：If you already have a set of waypoint data, you can manually load it, and the vehicle will proceed based on this data

2. Motion planner

- The Motion Planner is responsible for planning short-term routes and controlling the vehicle's turning angle and speed to align with the path planned by the Mission Planner. In the Motion Planner, various planner algorithms, such as A* planner or lattice planner, are available. These are typically used to complement obstacle avoidance, traffic light detection, stop line detection, and other functionalities.

Chapter 4 Tracking Methods

4.1 YOLOv8

When it comes to YOLO object detection, in simple terms, it is a technology that enables a computer to quickly identify objects and their locations in images. The full name "You Only Look Once" means that the computer needs only one glance to complete object image recognition and localization, making YOLO highly efficient in the field of object detection.

The underlying principle of YOLO object detection is based on the concept of deep learning. Deep learning is a machine learning technique that mimics the operation of the human brain's neural networks, allowing computers to autonomously learn and recognize objects. In deep learning, the computer learns by using a large number of annotated images with object categories. After extensive training, the computer can recognize objects in new images.

YOLO object detection employs the methods of deep learning technology. It divides an image into many small grids and analyzes the possible objects and their positions in each grid. During the analysis, YOLO considers features such as the shape and color of objects and compares them with previously learned knowledge. Ultimately, it determines the types and positions of objects in the image (Figure 4.2). The advantage of YOLO object detection lies in its ability to achieve rapid identification and localization of objects in images in an extremely short time. This makes it highly valuable in applications that require real-time responsiveness, such as in autonomous driving and surveillance systems.

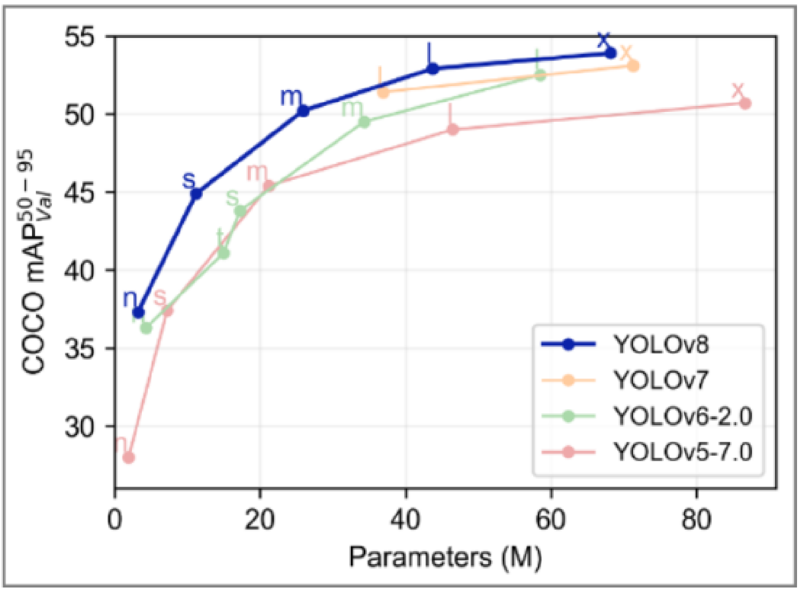

Figure 4. 1 Comparison of YOLOv8 Size with Previous Generations [16]

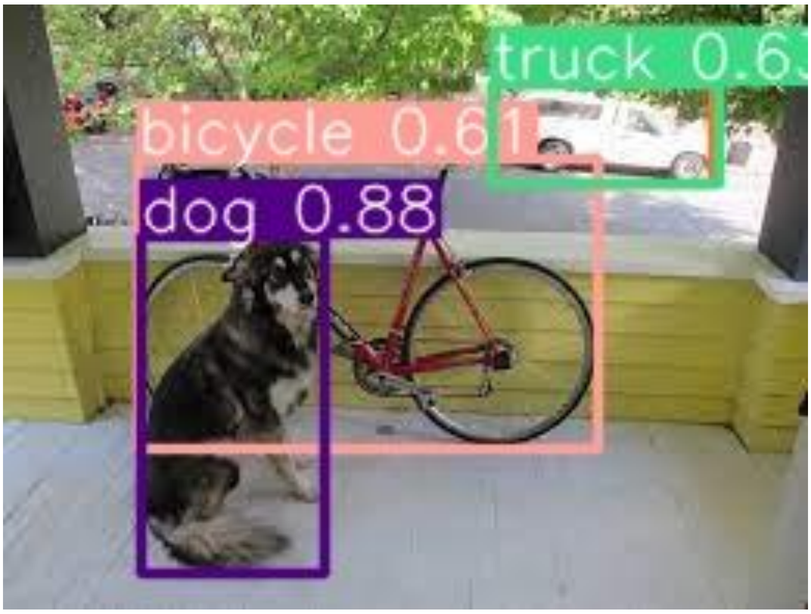

Figure 4. 2 Image Recognition Results using YOLOv8

The YOLO model family has continuously evolved over the years, with each version bringing improvements in performance, speed, and applications, from YOLO v1 in 2015 to YOLO v8 in 2023. The project at hand utilizes the latest YOLO v8 version, and based on the YOLO v8 documentation, the key improvements are outlined below:

1. Provides a framework capable of running models from all previous versions.
2. Introduces a completely new backbone network model.
3. Abandons the use of pre-defined detection boxes (anchor-free detection head).
4. Implements a new loss function.
5. CPU/GPU compatibility, resulting in a 60% reduction in training time, reduced parameter count, and more accurate predictions (mAP). See Figure 4.1 for a size comparison between YOLO v8 and previous generations.

4.2 Deep SORT

4.2.1 Deep SORT Introduction

Deep SORT is a widely used algorithm for Multi-Object Tracking (MOT), with its primary objective being the effective integration of object detection and target tracking. This approach initially employs an object detection algorithm, such as YOLO or Faster R-CNN, to identify objects in the video. Subsequently, the Deep SORT algorithm is utilized to achieve continuous tracking of these objects. During the tracking process, Deep SORT utilizes a Kalman filter to predict the motion trajectory of objects. Simultaneously, it employs appearance features (such as color and texture) for unique object identification, facilitating association across different frames. One of the core technologies of Deep SORT is the use of a deep learning-based Re-ID (Re-Identification) model. This model extracts appearance features to achieve unique identification and tracking of objects. The project utilizes the characteristics of the Re-ID (Re-Identification) model to track and re-identify specific individuals in unique situations. The model learns deep-level features, demonstrating robustness against interference like occlusion and changes in lighting conditions. It plays a crucial role in accurately re-identifying specific individuals in unique situations. Deep SORT has achieved widespread success in various applications, including traffic monitoring, personnel tracking, and pedestrian re-identification. Its main advantages lie in high precision and robustness, allowing effective tracking even in complex scenarios such as high-speed motion, target intersections, and occlusion. The primary workflow is depicted in Figure 4.3.

1. Detector obtains the detection bounding box at time t, updating the track set.
2. Kalman filter predicts the bounding box at time t + 1 based on the previous state.
3. Detector obtains the detection at time t + 1.
4. Hungarian algorithm is used to match the detection and prediction bounding boxes at time t + 1, updating the track set.

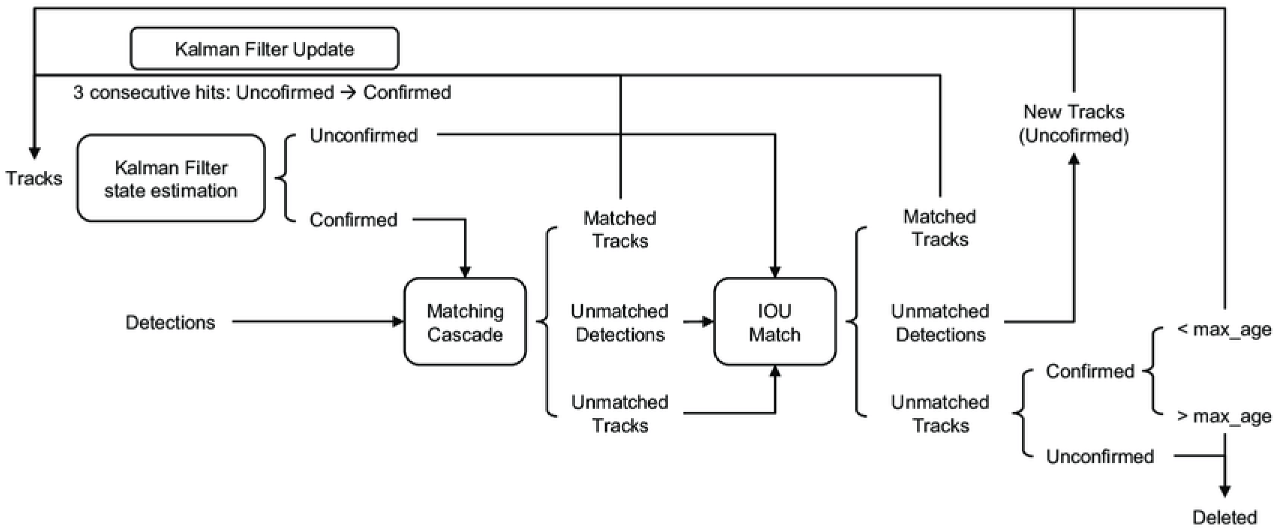

Figure 4. 3 Deep SORT Tracking Process Diagram [13]

4.2.2 Kalman filter

The Kalman filter is a mathematical method used to estimate the state of a dynamic system, particularly suitable for systems subject to measurement noise and motion model noise interference. Its core idea lies in balancing past estimates and current measurements to provide the optimal estimate of the system's state.

Due to its ability to provide optimal estimates, the Kalman filter finds widespread applications in various fields, including aircraft navigation, finance, and robot control. In this project, its estimation characteristics are utilized to infer the path of the next frame. The Kalman filter enhances system accuracy and stability, enabling precise state estimation in real-time applications.

The Kalman filter algorithm consists of two main processes: prediction and update. For the Deep SORT algorithm, it defines the motion state of the target as a vector containing the object's screen position (x, y), the object's bounding box dimensions (w, h), and the time-varying rates (x, y, w, h) represented by eight normal distribution vectors.

Prediction Process: When the target moves, it uses the parameters from the previous frame, such as the target box size and velocity, to predict the current frame's target box position and velocity.

Update Process: Through linearly combining the predicted values and observed values with weighted factors, the current system's predicted state is obtained. Both predicted and observed values are in the form of normal distributions, achieving the optimal estimate of the system's state. The overall process of Deep SORT can be referred to in Figure 4.4.

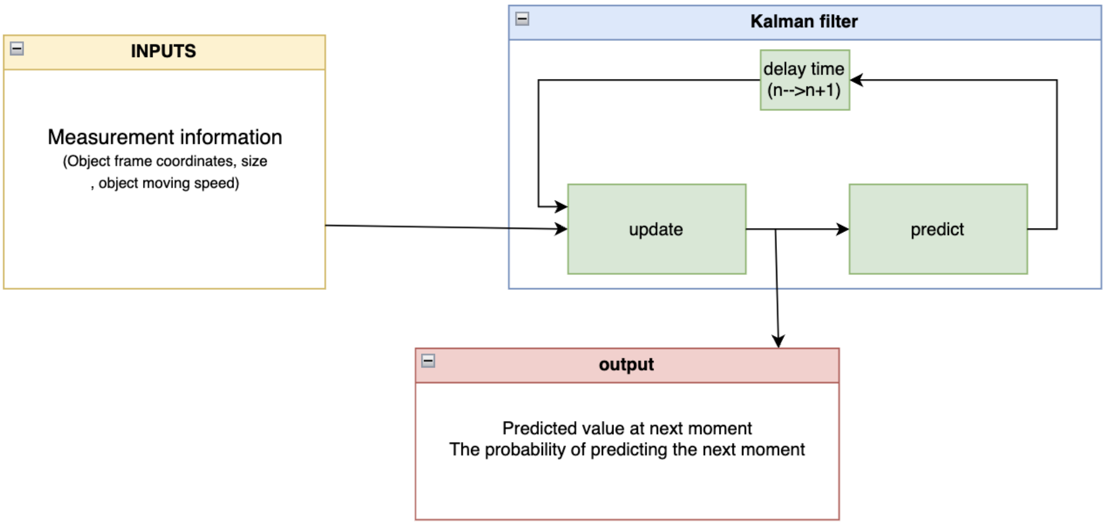

Figure 4. 4 Use of Kalman Filter in Deep SORT

4.2.3 Hungarian Algorithm

The Hungarian algorithm is a graph theory algorithm used to solve bipartite matching problems, as illustrated in Figure 4.5.

In Deep SORT, the Hungarian algorithm is employed to compare tracking boxes from the previous frame with detection boxes in the current frame. The algorithm calculates the cost matrix through appearance information, Mahalanobis distance, or Intersection over Union

(IOU). This is done to find the optimal assignment of numbers, aiding in associating tracking boxes with detection boxes. ◦

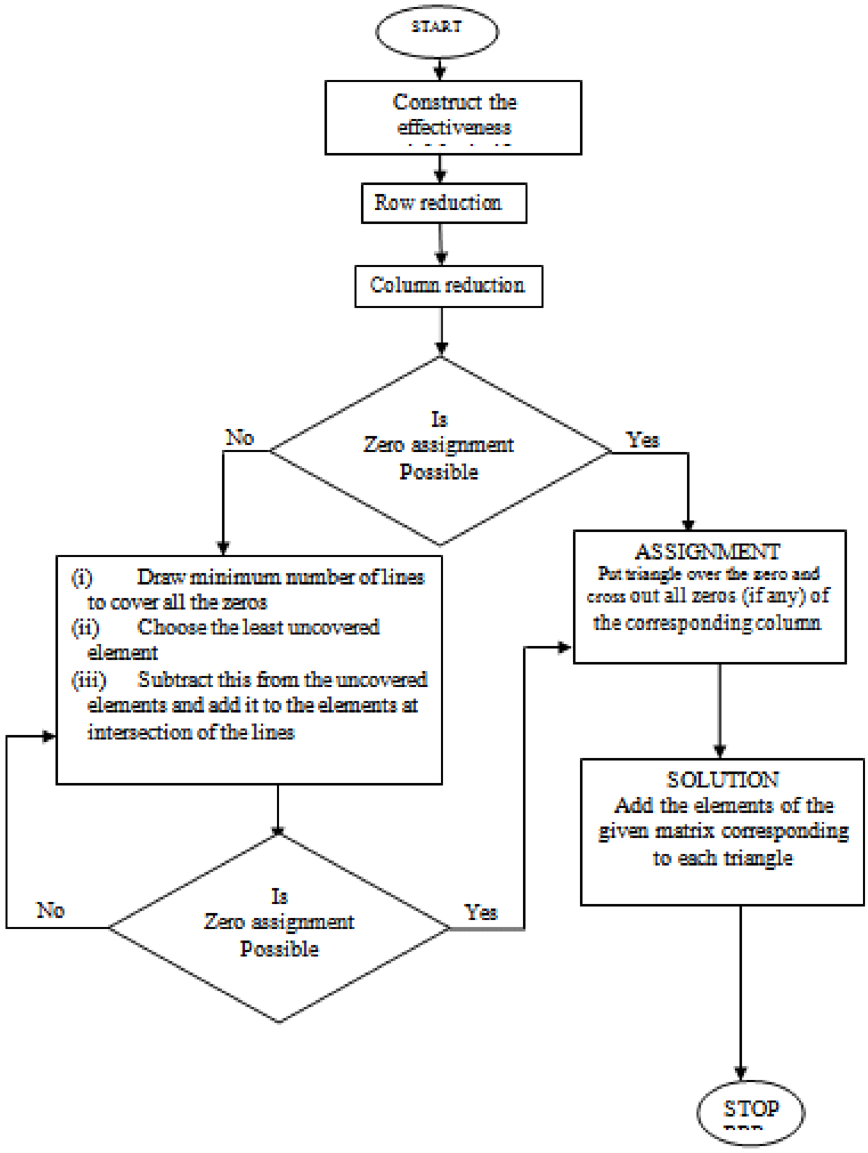

Figure 4. 5 Brief Process Diagram of Hungarian Algorithm

Chapter 5 Path Planning

5.1 Carrot chasing Algorithm

Carrot Tracking describes an algorithm that involves constructing virtual target points (VTP, Virtual Target Point) along a path. The target point is referred to as the "Carrot," and the autonomous vehicle, moves towards this target point. The vehicle's movement path gradually converges towards the target route, resembling an asymptotic line

The principle of the Carrot Tracking algorithm is illustrated in the diagram below, where multiple nodes are established on a two-dimensional plane. The nodes are connected with straight lines to obtain the shortest path, allowing the autonomous vehicle to follow this path and achieve the goal of the algorithm's path planning.

The movement of the autonomous vehicle can be simplified to a two-dimensional plane, and the model can be represented as shown in Figure 5.1 In this figure, the coordinates (x1, y1) represent the starting point of the autonomous vehicle, while (x2, y2) represent the user's coordinates. The objective is to move the autonomous vehicle to the user's location, where "h" represents the depth distance between the vehicle and the user, "w" represents the horizontal distance, and "safe_dis" defines the safety distance the vehicle needs to maintain.

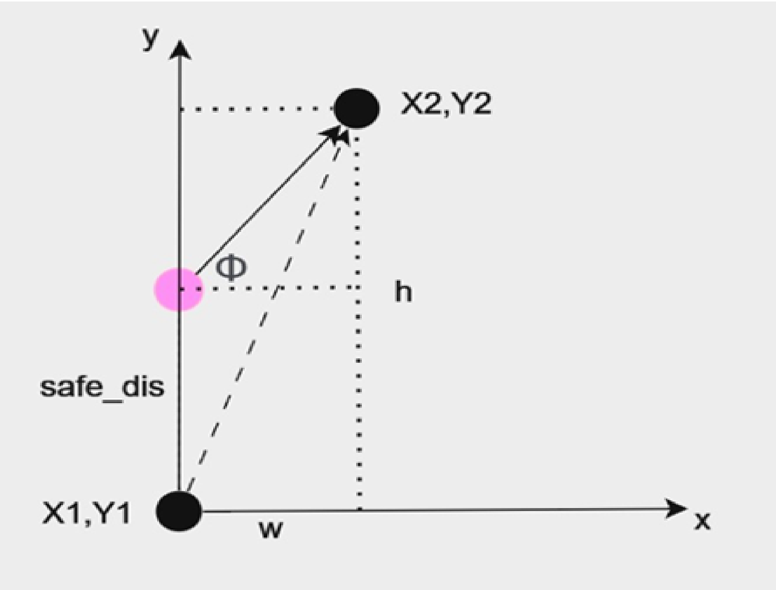

圖 5. 1 Geometric Relationship Diagram of Carrot Chasing to Nodes

The above definitions lead to the geometric derivation of the following formula:

$$\phi = tan^{-1}\left(\frac{h - safe\_dis}{w}\right) \tag{5.1}$$

5.2 Pure Pursuit Algorithm

The Pure Pursuit algorithm, also known as the "pure tracking algorithm," was initially conceptualized by Richard Wallace in 1985. Its primary application was to compute the necessary steering angles for a robot to return to a target path. The concrete application of the algorithm was later presented by Craig Coulter in 1992.

The foundation of the Pure Pursuit algorithm is primarily built upon the Ackermann geometric steering model and the two-dimensional bicycle model. The core concept involves calculating the turning angle between the vehicle's position and a "target point" using parameters such as the vehicle's forward visibility, turning radius, and steering angle. Subsequently, the vehicle follows an arc trajectory to achieve the tracking objective.

The mathematical derivation of the algorithm is based on the Ackermann geometric steering model, as outlined below.

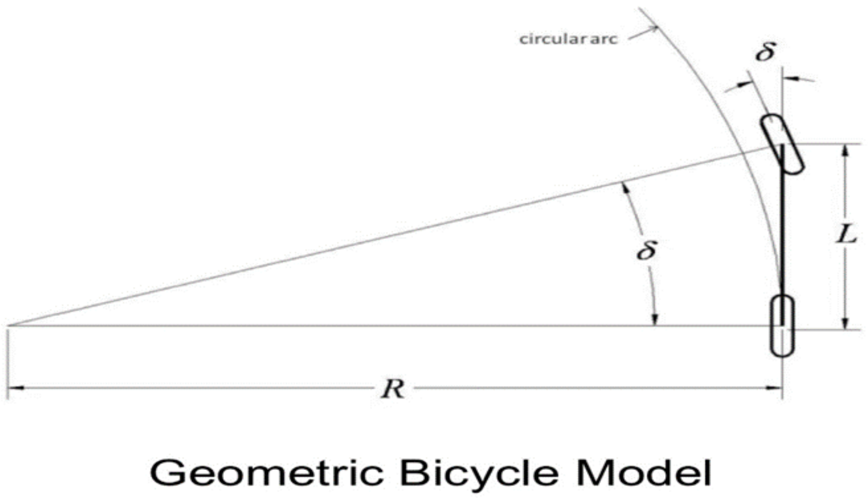

Figure 5. 2 Ackermann Geometric Bicycle Model [14]

We can deduce from Figure 5.1 the relationships among the following parameters:

L：wheelbase

R：turning radius

Φ：steering angle

These relationships can be expressed as：

$$tan(\delta) = \frac{L}{R} \tag{5.2}$$

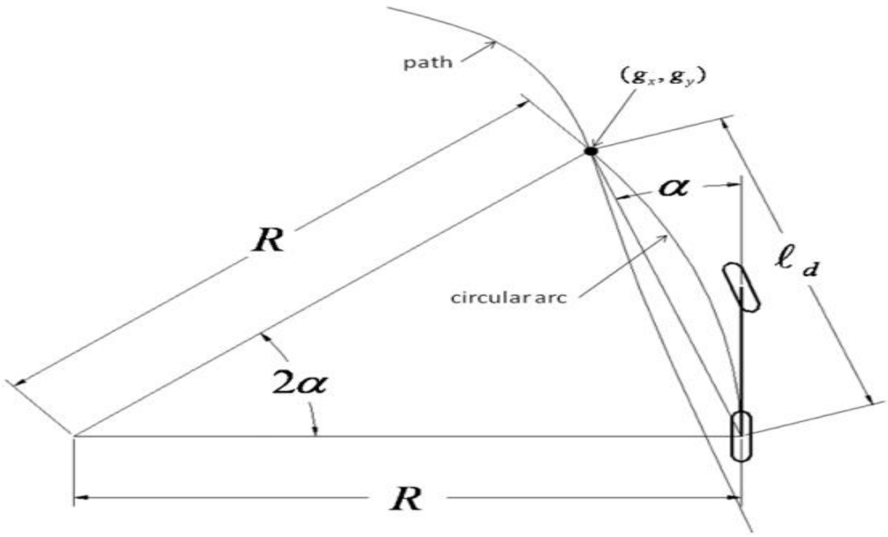

Figure 5. 3 Geometric Relationship Diagram of Pure Pursuit and Target Nodes [15]

In Figure 5.3, l_d represents the forward distance, α signifies the angle between the target point and the forward direction, and (g_x, g_y) denotes the coordinates of the target point. The derivation process for the circular arc curvature (κ) is outlined as follows:

$$\frac{\mathrm{l_d}}{\sin(2\alpha)} = \frac{R}{sin\left(\frac{\pi}{2} - \alpha\right)} \rightarrow \kappa = 2\frac{2Lsin(\alpha)}{l_d}, \tag{5.3}$$

After rearranging Equation (5.2), we obtain the following:

$$\delta = \tan^{-1}(\kappa L)\,. \tag{5.4}$$

Combining the above equations, we derive:

$$\delta(\mathrm{t}) = \tan^{-1}(\frac{2Lsin(\alpha(t))}{l_d}) \tag{5.5}$$

Additionally, we define the lateral error as:

$$\mathrm{e} = \mathrm{l_d}\sin(\alpha)\,. \tag{5.6}$$

When the steering angle is minimal, the lateral error can be approximated as:

$$\delta = \frac{2L}{l_d}\sin{(\alpha)}, \tag{5.7}$$

The determined forward distance ℓd，allows us to apply the above formulas to calculate the steering angle δ, essential for updating the vehicle's motion state in the context of path planning.

Chapter 6 Checkout System Design

6.1 Introduction

Traditional checkout processes involve scanning product barcodes and manually entering product names or prices, which can be time-consuming, especially during busy periods. However, using image recognition technology can expedite the checkout process, and in some cases, can be omitted. Additionally, it contributes to reducing physical contact, particularly in the current era of the pandemic.

The current method for checkout using image recognition often involves customers placing items in front of the cashier or shopping cart camera, allowing the camera to capture the current view inside the cart for identification. However, this approach has its limitations. For instance, if a customer places an item and later regrets it, wanting to remove it, or if a product is obscured by other items, the accuracy of the checkout system may be compromised. Moreover, some customers may intentionally block items with their hands to prevent the camera from accurately reading them, potentially leading to significant losses for the store.

To address these issues, we aim to develop a system that can instantly and directly recognize purchasing behavior, ensuring the consistency between the items in the shopping cart and the products the consumer intends to purchase. This system will enhance the accuracy and efficiency of the checkout process by analyzing each frame of the video to determine the placement and removal of items.

6.2 System Structure

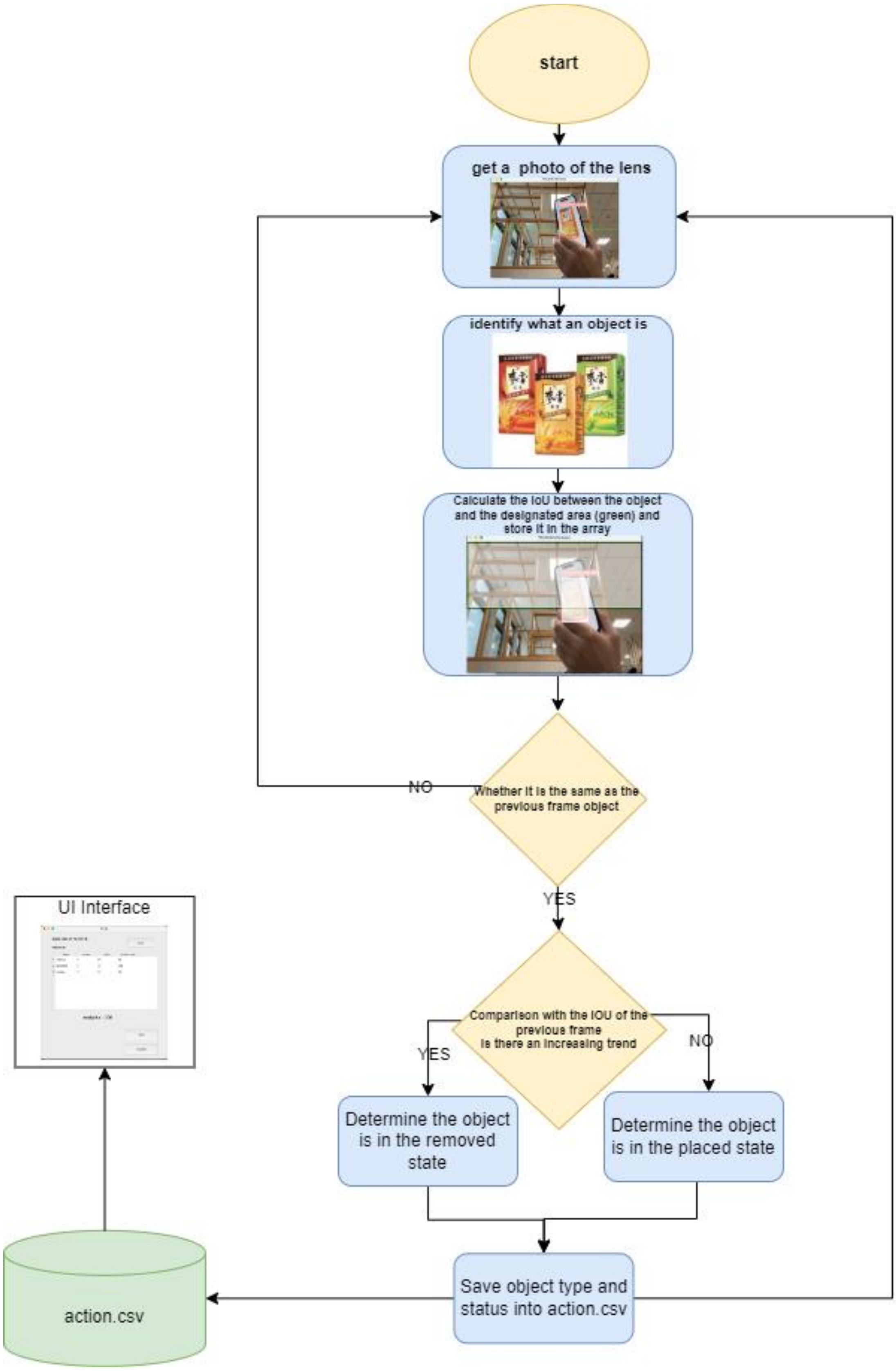

Figure 6. 1 Checkout System Process Diagram

6.3 IoU (Intersection over Union)

IoU (Intersection over Union) is a standard used to measure the accuracy of object detection algorithms, commonly employed in algorithms such as YOLO. It is calculated by taking the ratio of the intersection area to the union area of the predicted object region by the object detection algorithm and the actual object location, as illustrated in Figure 6.2. The mathematical representation is given by equation (6.1).

$$\mathrm{IoU} = \frac{\text{area of intersect}}{\text{area of union}} \tag{6.1}$$

Due to the characteristic of IoU varying with the movement and size of objects, we utilize this concept to enhance the fundamental method for detecting dynamic trends in objects. Specifically, we establish a fixed region and calculate the Intersection over Union (IoU) between the bounding box of the detected object and the fixed region. By monitoring the fluctuations in IoU values, we can infer the trend of the object, thereby predicting whether the object is being extracted or inserted.

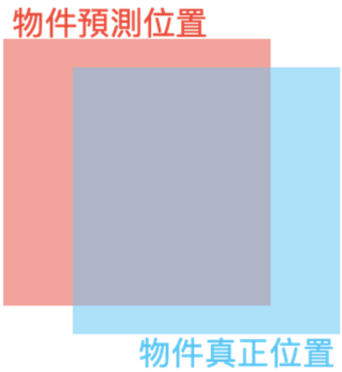

Figure 6. 2 IOU

6.4 Queue

A queue is a data structure with the characteristic of First-In-First-Out (FIFO). This means that when the queue is full and there is a need to add another element, the queue removes the earliest added element, shifts the memory positions of all remaining elements forward by one, and then inserts the new element.

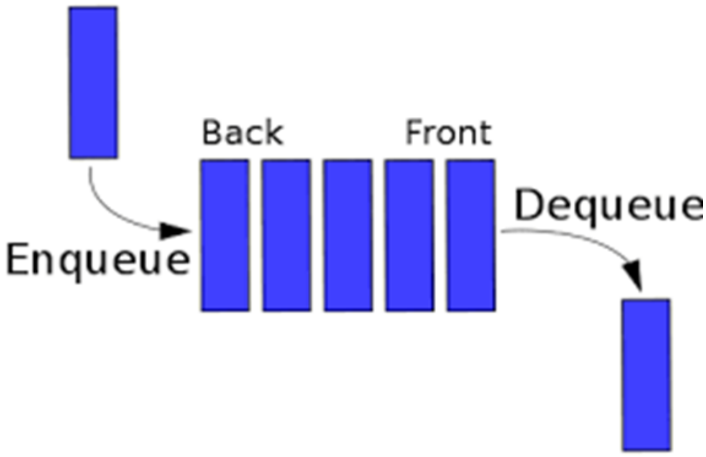

Figure 6. 3 Queue Data Structure Illustration

6.5 Alogorithms

Primary Algorithm

Through the detection in each frame, we delineate a region and calculate the Intersection over Union (IoU) between the moving object and the designated area. Utilizing time series analysis, we discern trends. When we observe a gradual decrease in IOU values, we infer that the object is in the process of being inserted. Conversely, an increase in IoU values indicates that the object is being taken out. With time series analysis, we can intuitively determine the dynamic trajectory of an object, whether it is being extracted or inserted. This approach contrasts with many current image recognition-based checkout systems that are primarily designed for static images.

Time Series Analysis

Time series analysis is a statistical technique used for handling time series data and trend analysis. Time series data involves measurements taken at regular intervals or collected at specific time intervals. In contrast, the algorithms for image recognition assess the content of each frame independently rather than making continuous judgments. Therefore, we can apply time series analysis by combining the data from each frame with its corresponding timestamp, creating time-related data. We organize the data from each frame in chronological order and store it in an array. Since the processing speed for each frame's time is nearly constant, it can be treated as evenly spaced data for analysis.

Repetitive Judgment Filtering

Due to the continuous nature of the insertion and extraction actions, the discrete sampling in each frame can lead to the discretization of these actions, resulting in repetitive judgments and an inaccurate count of insertions and extractions. The solution involves leveraging properties similar to those of a queue. We categorize each occurrence of judging an extraction (insertion) into an array. Subsequently, we reintegrate the discrete actions with their respective timestamps. Therefore, identifying the breakpoints of continuous actions allows us to determine whether an action has already been counted accurately.

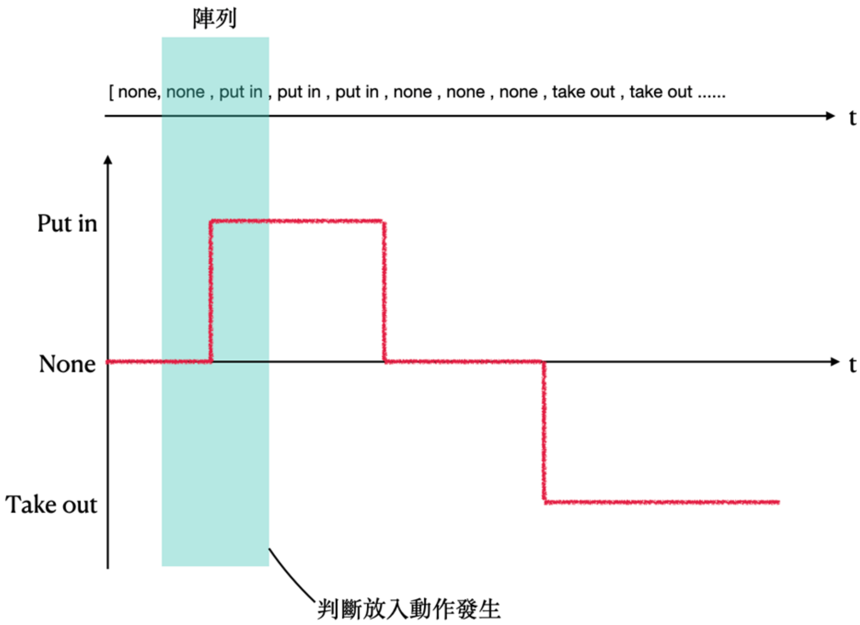

Figure 6. 4 System Judgement Illustration when Insert Action Occurs

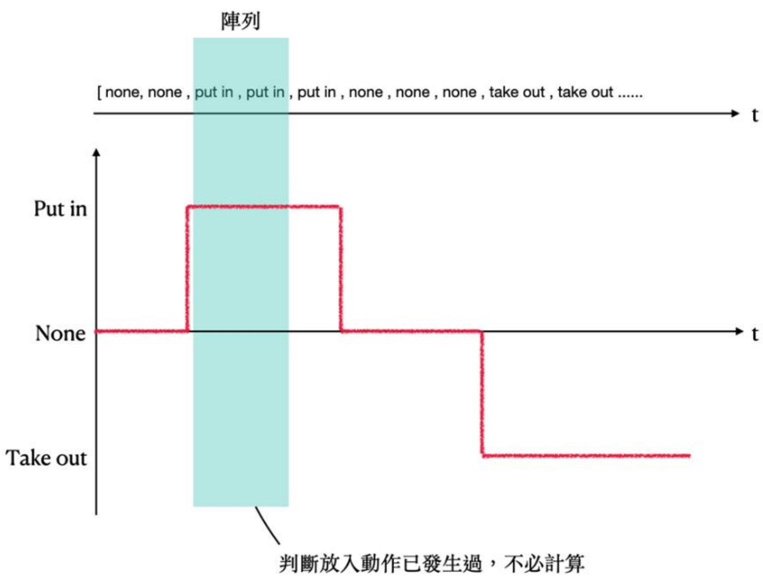

Figure 6. 5 System Judgement Illustration when Insert Action Occurs

6.6 UI Design

1. Login Functionality
2. Display of User Name
3. Presentation of Items Added to the Shopping Cart for User Confirmation
4. Computation of Total Price
5. Payment Button
6. Cancel Purchase Button

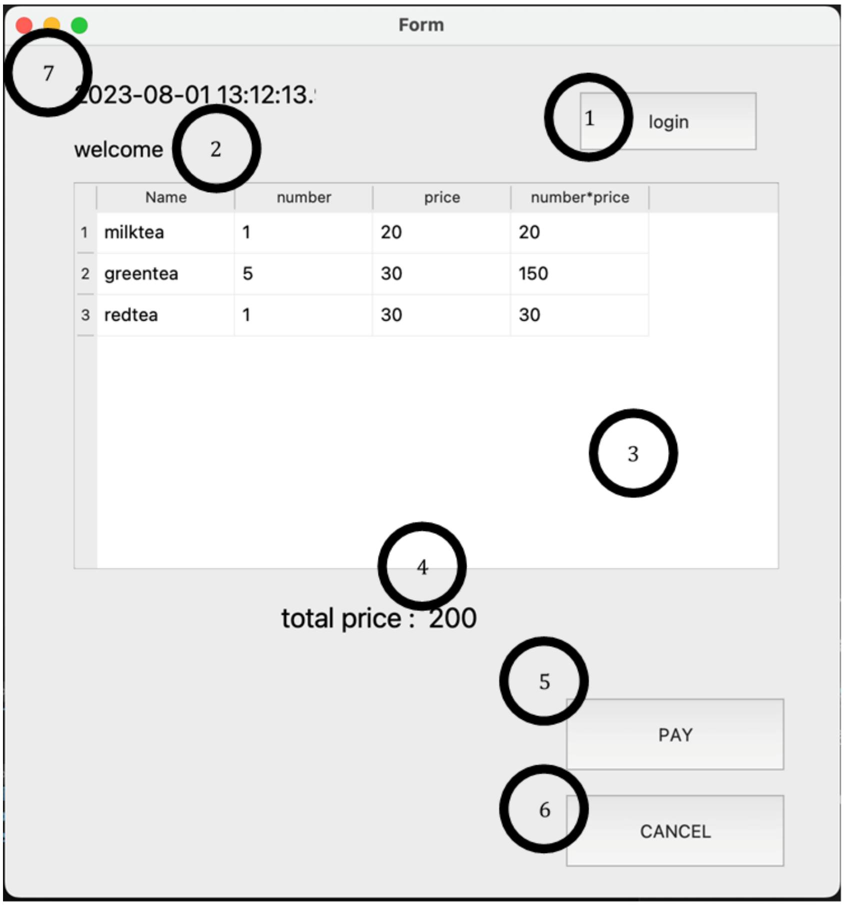

Figure 6. 6 Checkout UI interface

Chapter 7 Experimental Results

7.1 Experimental Categories

Taking into account the concepts and techniques discussed above, this project will divide the Automatic Shopping Cart Experiment into three main experiments:

- Automatic Return
- Following Functionality
- Automatic Checkout Systems

7.2 Automatic Return (Autoware)

This project utilizes the path recording feature of Autoware. The envisioned approach involves pre-recording the path from the cart borrowing entrance to the parking area. When we have completed our shopping and intend to return the cart by entering the parking area, the shopping cart will autonomously navigate back along the pre-recorded path to the cart borrowing entrance. This aims to achieve the ultimate goal of true hands-free operation.

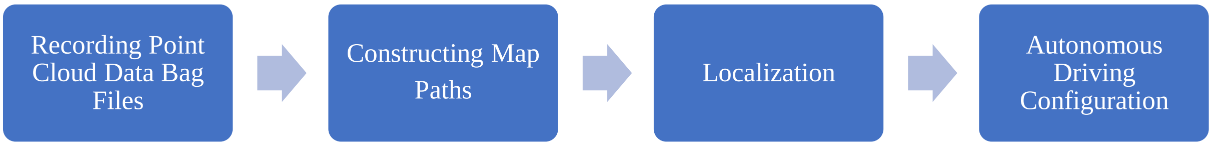

Figure 7. 1 Experiment Process Diagram

During the testing of the auto-retrieval system, we observed that the system could follow the recorded path and simultaneously eliminate any disturbances that occurred during recording, ensuring stability during the automatic return process. Autoware demonstrated the capability to autonomously estimate path planning within a certain distance, successfully completing the retrieval task.However, in the presence of obstacles, Autoware was unable to react immediately and lacked dynamic obstacle avoidance capabilities. This limitation will be the focus of further in-depth research and improvement in future work.

7.3 Following Functionality

In order to design a system that can stably track a target without interference, we have developed the following flowchart for the tracking system, as shown in Figure 7.2. Additionally, experiments have been conducted to evaluate the performance of both image tracking and actual tracking.

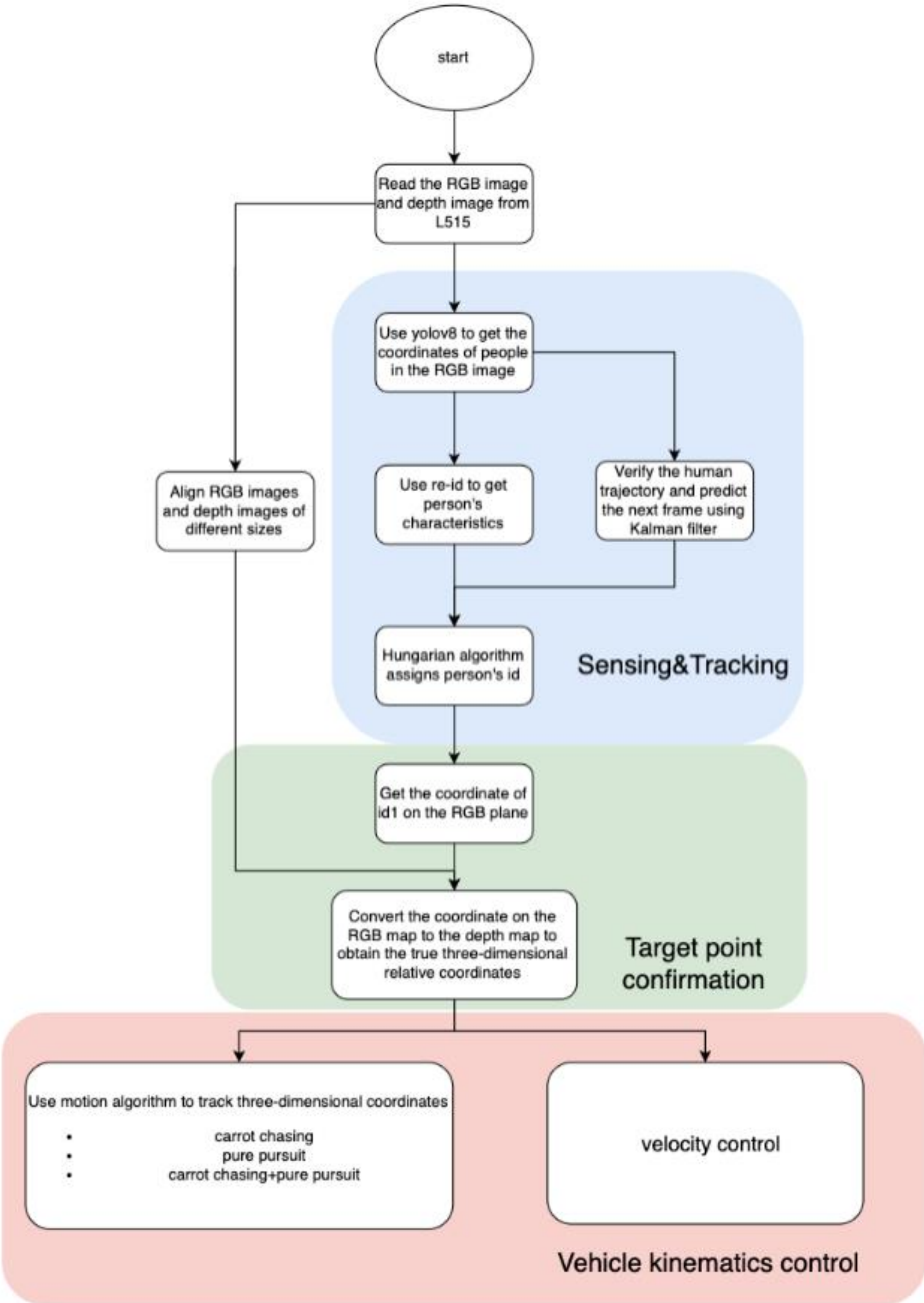

Figure 7. 2 Autonomous Following System Process Diagram

7.3.Image Tracking

To assess the performance of Deep SORT in tracking, we conducted two experiments:

1. Stability in the Presence of Multiple Target Interference
2. Ability to Recover a Target after Being Occluded

To test the tracking system's continuous tracking performance in scenarios with multiple people, such as malls and hospitals, we conducted an experiment. In this experiment, we initially fixed the camera in one position, designated a primary test subject, and introduced multiple distractors—individuals who continuously moved around and obstructed the primary test subject. Subsequently, we observed whether the system could consistently track the primary test subject even after 10 seconds. The following are the experimental results:

Table 4 System Tracking Accuracy under Target Interference Conditions

The number of people within the field of view.	1	2	3	4
Whether it is tracking the correct person.	o	o	o	o

Based on the experimental results, it was determined that the Deep SORT tracking algorithm can consistently track the same target in an environment with more than one person, remaining robust against interference. Moreover, given the presence of multiple corners and blind spots in indoor environments, target tracking might be temporarily interrupted due to occlusion. To test the permissible duration of disappearance, we initially fixed the camera in one position and designated a primary test subject as the main tracking target. The primary test subject temporarily disappeared from the frame after 30 seconds. Simultaneously, we assessed whether the appearance of other distractors during the disappearance period would lead to misjudgments. The experimental results regarding the accurate recovery of the target when it disappears are presented in Table 5.

Table 5 Precision of Target Rediscovery when Lost

Whether there is interference/disappearance time.	10s	30s	50s	60s	70s
Yes	o	o	o	x	x
No	o	o	o	o	x

7.3.2 Implementation of Following

After incorporating Deep SORT to ensure accurate tracking of the same target, to achieve a following effect, we utilized the Pure Pursuit path tracking algorithm. However, due to the potential for oscillations in smaller spatial environments caused by insufficient lookahead distance, we made slight modifications. We tested manual control, Pure Pursuit, and the modified angular velocity and acceleration variables using the built-in IMU in the iPhone 14 Pro. The relationships between these variables and time were plotted, along with the path trajectories. The red-boxed positions in the graph indicate turning points.

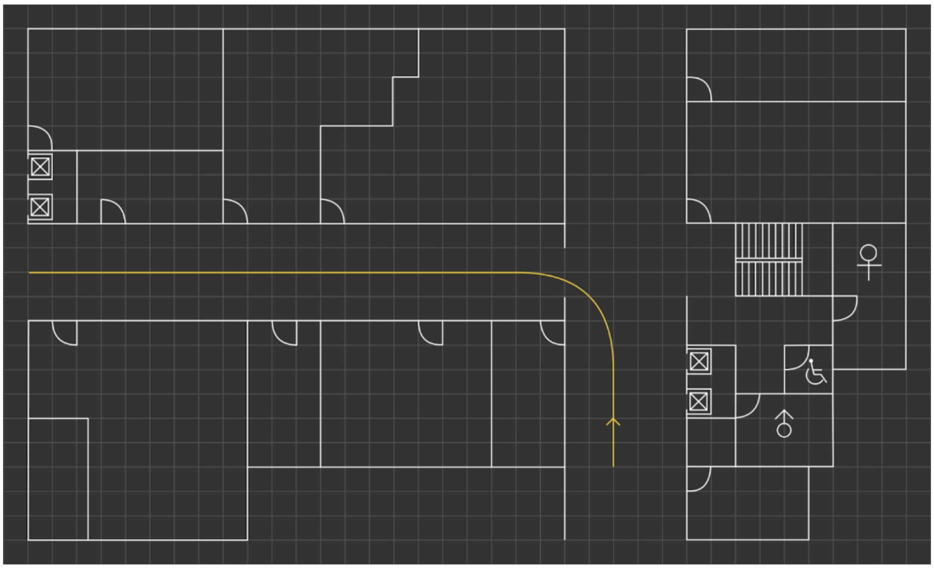

Figure 7. 3 Experimental Planning Route

Firstly, using manual control of the vehicle along the planned path depicted in Figure 7.3, we recorded the angular velocity graph of the vehicle under manual control, as shown in Figure 7.4. This graph illustrates the recorded vehicle angular velocity under manual control. During testing, a slight right deviation of 5 degrees was observed in the original front wheel steering angle, and efforts were made to maintain stability during manual control. However, this led to minor oscillations in the straight-line control portion.

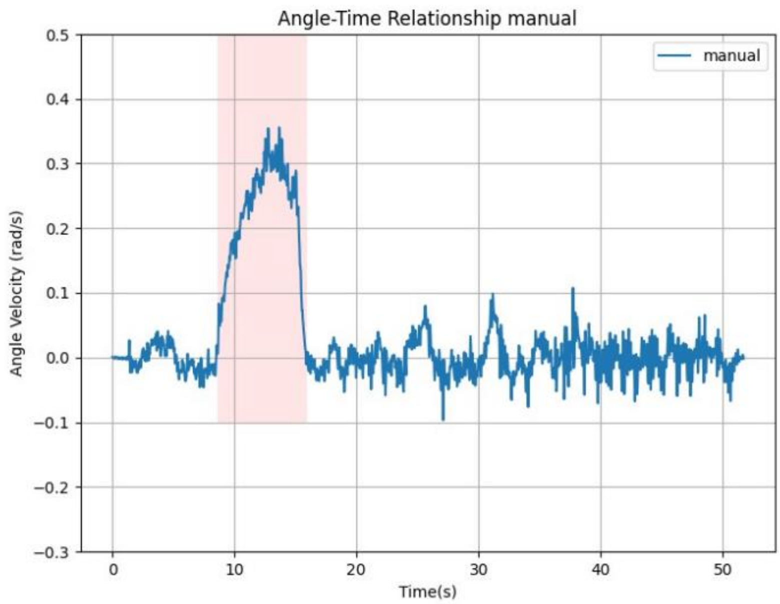

Figure 7. 4 Relationship between Angular Velocity and Time during Manual Control

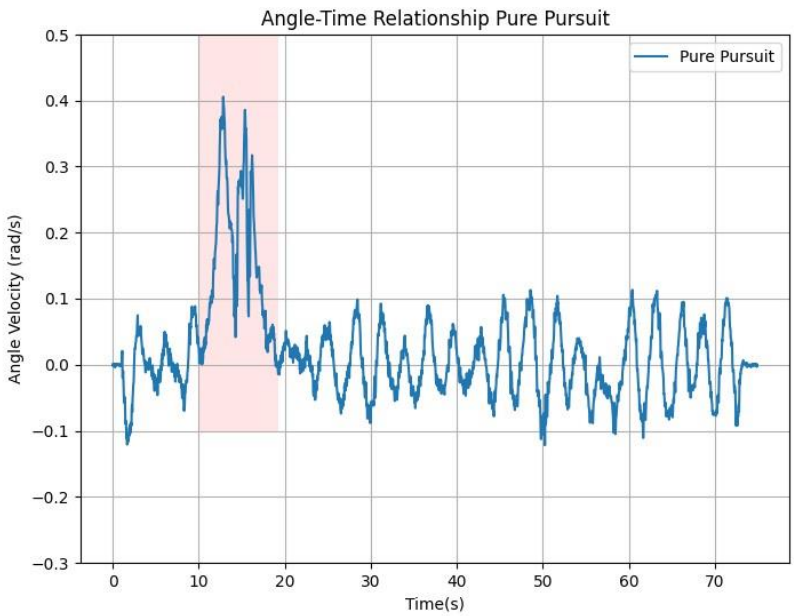

Figure 7. 5 Relationship between Angular Velocity and Time using Pure Pursuit to Follow the User

Figure 7.5 illustrates the angular velocity profile recorded for the vehicle utilizing the Pure Pursuit following method. In comparison to the peak values observed during manual operation, Pure Pursuit exhibits a peak value approximately 0.05 (rad/s) higher during turning. Although significant oscillations occur during turning to ensure successful navigation, particularly in scenarios where this following method is applied, challenges arise during straight-line following. Due to spatial constraints limiting the forward visibility, the system's sensitivity to target tracking becomes excessively high, resulting in substantial oscillations and hindering convergence.

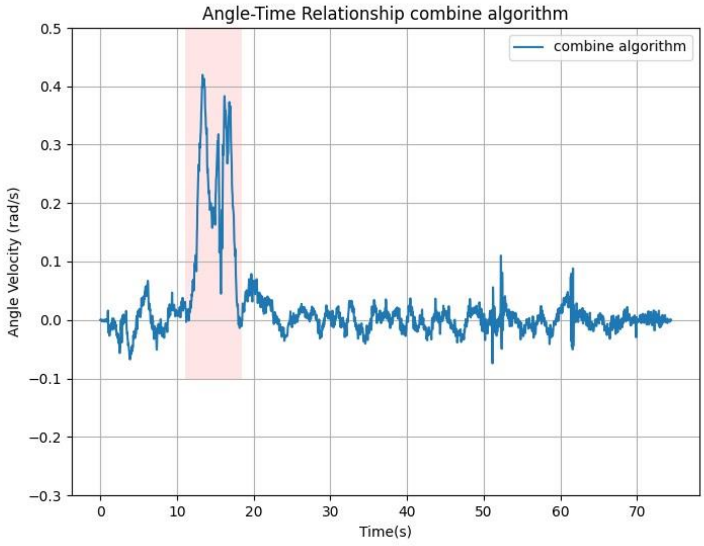

Figure7. 6 Relationship between Angular Velocity and Time using Improved Pure Pursuit to Follow the User

Figure 7.6 depicts the modified following method obtained by integrating Pure Pursuit with Carrot Chasing. The process diagram of the revised following method effectively preserves the turning advantages of Pure Pursuit while eliminating the oscillations inherent in Pure Pursuit during straight-line following. In comparison to manual operation, the revised following method exhibits reduced oscillations during straight-line movement. Consequently, the revised following method addresses the need for manual correction of wheel angle deviations observed in manual control.

Furthermore, we have generated trajectory plots for different following methods, as illustrated in Figure 7.7.

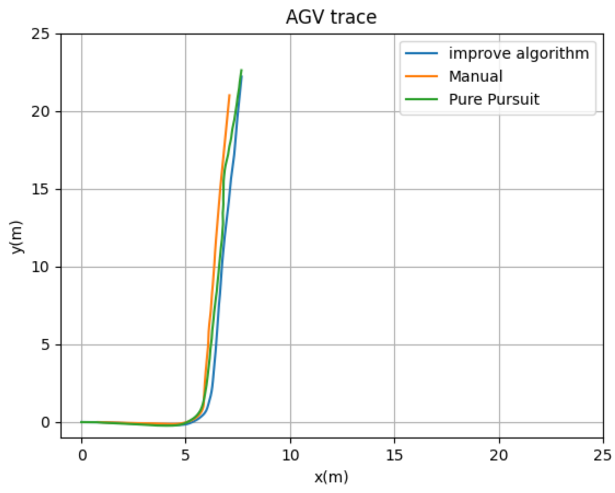

Figure 7. 7 Path Following by Different Methods

Under the Pure Pursuit following method, the vehicle's trajectory deviates from a straight line and exhibits an unstable curve. This is confirmed by the angular velocity plot of the vehicle under Pure Pursuit, showing significant oscillations during straight-line following, and the path is no longer a straight line. In contrast, the revised following method achieves a nearly identical straight-line path to manual operation, addressing the oscillation issue observed in the Pure Pursuit method and achieving effective straight-line following. ◦

7.4 Automatic Checkout System

In order to achieve the checkout functionality through the integration of image recognition and algorithms, we collected a total of three hundred images of relevant products. These images were divided into a training set and a control group in an 8:2 ratio. We utilized the YOLOv8 model for foundational training, creating a model capable of recognizing various products to serve as the image recognition basis for dynamic algorithms.

Based on this model's ability to recognize product items, we conducted experiments involving 100 instances of placing and retrieving actions for identification. The objects were tested for detection and action determination at a frequency of 1Hz. Through these tests, we obtained the following accuracy rates for action identification under stable lighting conditions:

- Placement: 97%
- Retrieval: 96%

Chapter 8 Reaults and Discussion

In this project, we successfully integrated self-driving cars with the Deep SORT image tracking technology, aiming to achieve efficient automatic tracking and re-identification of targets.

The utilization of the Deep SORT algorithm greatly enhanced the stability of our purely visual image recognition-based target tracking. By combining self-driving cars with Deep SORT technology, we were able to consistently track targets in dynamic environments. This integration holds promising future developments, particularly in applications such as shopping carts and security surveillance. Whether faced with scenarios featuring multiple individuals or changing backgrounds, our system maintains the ability to continuously track targets, ensuring robust tracking performance in the presence of interference or momentary disappearance of the target.

However, despite the commendable results of our project, there are still challenges to overcome. Achieving a fully integrated smart store requires a focus on performance optimization, including refining algorithms and hardware to further enhance the system's performance and efficiency. Additionally, attention must be given to addressing privacy and security concerns to ensure the system's safety and the comprehensive protection of user privacy.

Ultimately, the application of self-driving cars combined with Deep SORT image tracking technology provides a promising solution for achieving automatic tracking and reidentification of targets, especially in the context of shopping cart applications. Through continuous improvement and optimization, this technology has the potential to play a broader role in various application domains, enhancing efficiency, convenience, and security. This research project serves as a valuable reference for the future development of self-driving car technology and holds the promise of bringing more convenience and benefits to society.

References

[1] Jiang, Z., Xu, Y. & Sun, L. “A Novel Path Tracking Controller for Magnetic Guided AGVs,” *33rd Chinese Control and Decision Conference*, Kunming, China, 2021, pp.3292-3296.

[2] Xu, H., Xia, J., Yuan, Z., & Cao, P. “Design and Implementation of Differential Drive AGV Based on Laser Guidance,” *3rd International Conference on Robotics and Automation Sciences*, Wuhan, China, 2019, pp.112-117.

[3] Li, W., Zhang, N., Yao, Y., Zhan, H., & Ding, Y. “Research on Vision Navigation and Positioning of AGV Terminal Based on QR Code,” *IEEE 9th Joint International Information Technology and Artificial Intelligence Conference*, Chongqing, China, 2020, pp.845-848.

[4] Kuang, J. M., & Liu, M. “Stereo Vision based Moving Target Tracking for Omni-directional Mobile Robots in 2D Space,” *International Conference on In formation Acquisition*, Seogwipo, Korea (South), 2007, pp.602-607.

[5] Yoo, H., Kim, D., Sohn, J., Lee, K., & Kim, C. “Development of a Worker-Following Robot System: Worker Position Estimation and Motion Control und er Measurement Uncertainty,” *Machines*, 11, 366.

[6] Coulter, R. Craig. “Implementation of the pure pursuit path tracking algorithm,” Carnegie Mellon University, The Robotics Institute, Pittsburgh, Hull, USA, 1992.

[7] Wojke, N., Bewley, A., & Paulus, D. “Simple online and realtime tracking with a deep association metric,” *IEEE international conference on image processing*, 2007

[8] Bukhari, S. T., et al. “A vision-based automatic retail checkout system ,” University of Engineering and Technology, Lahore (Main Campus), Pakistan, 2021.

[9] Carballo, A., Wong, D., Ninomiya, Y., Kato, S., & Takeda, K. “Training Engineers in Autonomous Driving Technologies using Autoware,” *IEEE Intelligent Transportation Systems Conference*, 2019, pp.3347-3354.

[10] Minecraftgamerpc69.(2008).Ubuntu18.04LTS BionicBeaver. https://en.wikipedia.org/wiki/Ubuntu_version_history

[11] iT 邦幫忙[ROS#2]系統架構及重要觀念介紹 https://ithelp.ithome.com.tw/articles/10216423

[12] The Autoware Foundation.(2022). https://github.com/autowarefoundation/autoware

[13] Avola D, Cinque L, Diko A, Fagioli A, Foresti GL, Mecca A, Pannone D, Piciarelli C."MS-Faster R-CNN: Multi-Stream Backbone for Improved Faster R-CNN Object Detection and Aerial Tracking from UAV Images," Department of Computer Science, Sapienza University, Rome, Italy, 2021.

[14] Dhargawe, S.D., Maske, M., Kailaswar, S., & Pusdekar, P. (2016). "Design and Implementation of Automatic Steering Control. International journal of engineering research and technology," *International Journal of Engineering Research & Technology*, 2016.

[15] Ren Y, Wang J, Zheng X, Zhao Q, Ma J, Zhao L. "Research on multidimensional evaluation of tracking control strategies for self-driving vehicles," Advances in Mechanical Engineering, 2020.

[16] Jocher, G., Chaurasia, A., & Qiu, J. (2023). YOLO by Ultralytics (Version 8.0.0). https://github.com/ultralytics/ultralytics

Appendix

1、Gantt Chart

Gantt Chart：

Months Tasks	No. 1	No. 2	No. 3	No. 4	No. 5	No. 6	No. 7	No. 8	No. 9	No. 10	No. 11	備註
Project direction discussion/individual divergent research												
Define functionalities in self-driving car												
Install and familiarize with Linux system Ubuntu and ROS												
Learn ROS system and AutoWare software interface and operation												
Map recording and self-driving practice												
Checkout UI interface design												
Analysis ans application of Publisher &Subscriber												
Control using Python and ROS instead of AutoWare												
Research and implement tracking algorithms												
Improve difficulties in autonomous car tracking and turning												
Add functionality to relock the target after target loss												
Stability testing and variable analysis of the tracking system												
Final project												
Project progress cumulative percentage %	8.3	16.6	24.9	33.2	41.5	50	58.3	66.6	83.3	91.6	100	

2、Project Meeting Records

國立臺北科技大學機械系大學部實務專題會議記錄表

<table>
<tr><td>指導老師</td><td>許志明</td><td>日期</td><td>111 年 09 月 18 日</td><td>次數</td><td>第 1 次</td></tr>
<tr><td colspan="4">參與討論組員姓名以及學號(可打字或親簽)</td><td rowspan="2">討論地點</td><td rowspan="2">綜科 525-4</td></tr>
<tr><td colspan="4">林辰燁 109810001
王姿伶 109810006
楊其康 109810021
陳勁旗 109810022
許喆堯 109810031</td></tr>
<tr><td colspan="3">本次討論議題：
1. 討論研究方向-主要從無人機跟自駕車討論
2. 訂定專題主題-老人照護、機械手臂等等
3. 專題功能可行性-避障、追蹤</td><td colspan="3">前次會議議題處理進度：
想出四個專題主題:
1. 無人機工具人
2. 無人機導遊
3. 無人機 Uber eats
4. 灑農藥無人機</td></tr>
<tr><td colspan="6">本次會議結論：
1.由於實驗室鑽研無人機的學長不多，並且考量無人機的控制難度，最後決定做自駕車當作這次專題的主軸。
2.專題主題的訂定方向尚未明朗，打算先熟悉基本機器人作業系統(ROS)以及自駕車開源軟體(Autoware)，滾動式進行發想及修正。</td></tr>
<tr><td>指導老師簽名</td><td colspan="2"></td><td colspan="2">填表紀錄者簽名</td><td></td></tr>
<tr><td colspan="6">注意事項:
1. 本表的目的是有效的在專題製作時，幫助全組掌握分工與進度。
2. 無任何具體討論內容、結論時不需填寫紀錄表。
3. 本表填完及簽名完建議將其立即掃描為 jpeg 檔形式，以利製作完整報告時置入其中。</td></tr>
</table>

國立臺北科技大學機械系大學部實務專題會議記錄表

<table>
<tr><td>指導老師</td><td>許志明</td><td>日期</td><td>112 年 03 月 02 日</td><td>次數</td><td>第 2 次</td></tr>
<tr><td colspan="4">參與討論組員姓名以及學號(可打字或親簽)</td><td rowspan="2">討論地點</td><td rowspan="2">綜科 525-4</td></tr>
<tr><td colspan="4">林辰燁 109810001
王姿伶 109810006
楊其康 109810021
陳勁旗 109810022
許喆堯 109810031</td></tr>
<tr><td colspan="3">本次討論議題：
a.訂定自駕車預想功能及可行的實踐方法
1. 跟隨使用者
2. 自動避障
b.實際遙控自駕車，手動及自駕進行練習。
c.和學長學習 Autoware 軟體基礎操作</td><td colspan="3">前次會議議題處理進度：
1. 安裝 Linux 系統中的 Ubuntu 及 ROS，並熟悉其工作環境
2. 安裝 Autoware，並理解其中核心演算法之基本原理。</td></tr>
<tr><td colspan="6">本次會議結論：
1.為更加熟悉 Autoware，每次專題全組皆要至實驗室外進行基本車子操作，皆須實際跑一次 Autoware 基本實驗流程-包括建構地圖路徑、定位及自駕設定。
2.各自練習基本 ROS code 及架構學習，於下次會議前對 ROS 中的節點(Nodes)、Master、Topic 等有基本理解。</td></tr>
<tr><td colspan="2">指導老師簽名</td><td colspan="2"></td><td>填表紀錄者簽名</td><td></td></tr>
<tr><td colspan="6">注意事項:
1.本表的目的是有效的在專題製作時，幫助全組掌握分工與進度。
2.無任何具體討論內容、結論時不需填寫紀錄表。
3.本表填完及簽名完建議將其立即掃描為 jpeg 檔形式，以利製作完整報告時置入其中。</td></tr>
</table>

國立臺北科技大學機械系大學部實務專題會議記錄表

<table>
<tr><td>指導老師</td><td>許志明</td><td>日期</td><td>112 年 03 月 30 日</td><td>次數</td><td>第 3 次</td></tr>
<tr><td colspan="4">參與討論組員姓名以及學號(可打字或親簽)</td><td rowspan="2">討論地點</td><td rowspan="2">綜科 525-4</td></tr>
<tr><td colspan="4">林辰燁 109810001
王姿伶 109810006
楊其康 109810021
陳勁旗 109810022
許喆堯 109810031</td></tr>
<tr><td colspan="3">本次討論議題：
1. 向教授報告目前想法跟進度，且正式規劃進度甘特圖。
2. 實現錄製地圖讓自駕車自駕到指定位置，確定車子返航功能。</td><td colspan="3">前次會議議題處理進度：
1. 熟悉 ROS 程式環境且皆練習寫 publisher 以及 subsciber 。
2. 利用程式來操控 ROS 之小烏龜，以利模擬未來控制車子。</td></tr>
<tr><td colspan="6">本次會議結論：
1.實際運行時發現利用 Autoware 控制自駕車能做出的變化有限，侷限性太高了僅能做點到點的控制，因此希望能找到方法跳脫 Autoware 的框架。
2.確定錄製地圖自駕功能後，開始往相關情景進行發想討論。</td></tr>
<tr><td colspan="2">指導老師簽名</td><td colspan="2"></td><td>填表紀錄者簽名</td><td></td></tr>
<tr><td colspan="6">注意事項:
1.本表的目的是有效的在專題製作時，幫助全組掌握分工與進度。
2.無任何具體討論內容、結論時不需填寫紀錄表。
3.本表填完及簽名完建議將其立即掃描為 jpeg 檔形式，以利製作完整報告時置入其中。</td></tr>
</table>

國立臺北科技大學機械系大學部實務專題會議記錄表

<table>
<tr><td>指導老師</td><td>許志明</td><td>日期</td><td>112 年 5 月 11 日</td><td>次數</td><td>第 4 次</td></tr>
<tr><td colspan="4">參與討論組員姓名以及學號(可打字或親簽)</td><td rowspan="2">討論
地點</td><td rowspan="2">綜科 525-4</td></tr>
<tr><td colspan="4">林辰燁 109810001
王姿伶 109810006
楊其康 109810021
陳勁旗 109810022
許喆堯 109810031</td></tr>
<tr><td colspan="3">本次討論議題：
1. 確立專題情境:大賣場自動跟隨車
2. 欲使用 yolo v8 影像辨識功能，可以將物品放入的同時結帳。
3. 深度相機的使用方法</td><td colspan="3">前次會議議題處理進度：
1. 成功跳脫 Autoware 框架，用 ROS 加 python 自編程式以驅動自駕車。
2. 進行情景模擬討論，與教授討論確定主題方向。</td></tr>
<tr><td colspan="6">本次會議結論：
1. 使用深度相機結合 yolov8 的姿態辨識，得出使用者跟載具的相對位置。
2. 嘗試用時間序列分析結合影像辨識，期望能完成動態結帳系統。</td></tr>
<tr><td colspan="2">指導老師簽名</td><td colspan="2"></td><td>填表紀錄者
簽名</td><td></td></tr>
<tr><td colspan="6">注意事項:
1.本表的目的是有效的在專題製作時，幫助全組掌握分工與進度。
2.無任何具體討論內容、結論時不需填寫紀錄表。
3.本表填完及簽名完建議將其立即掃描為 jpeg 檔形式，以利製作完整報告時置入其中。</td></tr>
</table>

國立臺北科技大學機械系大學部實務專題會議記錄表

<table>
<tr><td>指導老師</td><td>許志明</td><td>日期</td><td>112 年 06 月 08 日</td><td>次數</td><td>第 5 次</td></tr>
<tr><td colspan="4">參與討論組員姓名以及學號(可打字或親簽)</td><td rowspan="2">討論
地點</td><td rowspan="2">綜科 525-4</td></tr>
<tr><td colspan="4">林辰燁 109810001
王姿伶 109810006
楊其康 109810021
陳勁旗 109810022
許喆堯 109810031</td></tr>
<tr><td colspan="3">本次討論議題：
1. 跟隨方法決定與討論-pure pursuit 和 follow the carrot
2. 設計結帳 IU 介面功能-欲使用 PyQt5</td><td colspan="3">前次會議議題處理進度
1. 完成動態結帳系統，達到商品取出及放入的動態辨識。
2. 使用 yolo 神經網路訓練商品模型(麥香奶紅綠茶)，並用 RGB 相機測試結帳程式。</td></tr>
<tr><td colspan="6">本次會議結論：
1. 於實作時測試 pure pursuit 和 follow the carrot 實際效果
2. 決定 UI 介面有登錄、結帳取消等功能
3. 各自嘗試追蹤系統的撰寫</td></tr>
<tr><td colspan="2">指導老師簽名</td><td colspan="2"></td><td>填表紀錄者簽名</td><td></td></tr>
<tr><td colspan="6">注意事項:
1.本表的目的是有效的在專題製作時，幫助全組掌握分工與進度。
2.無任何具體討論內容、結論時不需填寫紀錄表。
3.本表填完及簽名完建議將其立即掃描為 jpeg 檔形式，以利製作完整報告時置入其中。</td></tr>
</table>

國立臺北科技大學機械系大學部實務專題會議記錄表

<table>
<tr><td>指導老師</td><td>許志明</td><td>日期</td><td>112 年 07 月 05 日</td><td>次數</td><td>第 6 次</td></tr>
<tr><td colspan="4">參與討論組員姓名以及學號(可打字或親簽)
林辰燁 109810001
王姿伶 109810006
楊其康 109810021
陳勁旗 109810022
許喆堯 109810031</td><td>討論地點</td><td>綜科 328</td></tr>
<tr><td colspan="3">本次討論議題：
1. 發現追蹤時會有跟錯人的問題，須找出方法解決。
2. 僅能進行直線跟隨，轉彎有困難，且無法鎖定特定目標。
3. 欲嘗試回授概念、比例控制調整車速。</td><td colspan="3">前次會議議題處理進度
1. 實際測試 pure pursuit 和 follow the carrot 效能，決定使用 pure pursuit
2. UI 介面初版設計完成</td></tr>
<tr><td colspan="6">本次會議結論：
1. 尋找能夠鎖定特定目標的方法
2. 實測時，加上各種參數的調控以觀察變因
3. 下次實作決定嘗試採用 openCV 追蹤</td></tr>
<tr><td colspan="2">指導老師簽名</td><td></td><td colspan="2">填表紀錄者簽名</td><td></td></tr>
<tr><td colspan="6">注意事項:
1.本表的目的是有效的在專題製作時，幫助全組掌握分工與進度。
2.無任何具體討論內容、結論時不需填寫紀錄表。
3.本表填完及簽名完建議將其立即掃描為 jpeg 檔形式，以利製作完整報告時置入其中。</td></tr>
</table>

國立臺北科技大學機械系大學部實務專題會議記錄表

<table>
<tr><td>指導老師</td><td>許志明</td><td>日期</td><td>112 年 07 月 19 日</td><td>次數</td><td>第 7 次</td></tr>
<tr><td colspan="4">參與討論組員姓名以及學號(可打字或親簽)</td><td rowspan="2">討論地點</td><td rowspan="2">綜科 328</td></tr>
<tr><td colspan="4">林辰燁 109810001
王姿伶 109810006
楊其康 109810021
陳勁旗 109810022
許喆堯 109810031</td></tr>
<tr><td colspan="3">本次討論議題：
1. 實作時發現轉彎方向錯誤，人往左但車子往右。
2. 須找出目標消失時，能重新辨識出目標的方法</td><td colspan="3">前次會議議題處理進度
1. 嘗試採用 Deep SORT 以解決追蹤錯誤問題。
2. 實作時調整轉彎比例以達轉彎效果
3. 調整 Kp 值，為使車子接近人時速度不過快。</td></tr>
<tr><td colspan="6">本次會議結論：
1. 各自研究 dancetrack、RE-id
2. 實作進行轉彎、車速等穩定度測試
3. 開始進行專題相關資料之收集整理</td></tr>
<tr><td colspan="2">指導老師簽名</td><td></td><td colspan="2">填表紀錄者簽名</td><td></td></tr>
<tr><td colspan="6">注意事項:
1.本表的目的是有效的在專題製作時，幫助全組掌握分工與進度。
2.無任何具體討論內容、結論時不需填寫紀錄表。
3.本表填完及簽名完建議將其立即掃描為 jpeg 檔形式，以利製作完整報告時置入其中。</td></tr>
</table>

國立臺北科技大學機械系大學部實務專題會議記錄表

<table>
<tr><td>指導老師</td><td>許志明</td><td>日期</td><td>112 年 08 月 10 日</td><td>次數</td><td>第 8 次</td></tr>
<tr><td colspan="4">參與討論組員姓名以及學號(可打字或親簽)</td><td rowspan="2">討論地點</td><td rowspan="2">綜科 328</td></tr>
<tr><td colspan="4">林辰燁 109810001
王姿伶 109810006
楊其康 109810021
陳勁旗 109810022
許喆堯 109810031</td></tr>
<tr><td colspan="3">本次討論議題：
1. 進行追隨系統穩定度之測試
2. 訂定實驗流程以及變因分析-光害、移動方式等
3. 向教授進行完整進度彙報</td><td colspan="3">前次會議議題處理進度
1. 嘗試使用 RE-id 來解決目標消失後能重新辨識追蹤的問題。
2. 在控制轉彎的程式增加對應的正負號進行方向修正。
3. 以 dancetrack 來測試追蹤演算法優劣</td></tr>
<tr><td colspan="6">本次會議結論：
1. 實測時有發現自身硬體算力不夠的問題，有需更新設備或是調整程式的考量
2. 下次實作欲進行路線規劃等設計，實測車子追隨人之路徑精準度。</td></tr>
<tr><td colspan="2">指導老師簽名</td><td></td><td colspan="2">填表紀錄者簽名</td><td></td></tr>
<tr><td colspan="6">注意事項:
1.本表的目的是有效的在專題製作時，幫助全組掌握分工與進度。
2.無任何具體討論內容、結論時不需填寫紀錄表。
3.本表填完及簽名完建議將其立即掃描為 jpeg 檔形式，以利製作完整報告時置入其中。</td></tr>
</table>

國立臺北科技大學機械系大學部實務專題會議記錄表

<table>
<tr><td>指導老師</td><td>許志明</td><td>日期</td><td>112 年 08 月 24 日</td><td>次數</td><td>第 9 次</td></tr>
<tr><td colspan="4">參與討論組員姓名以及學號(可打字或親簽)</td><td rowspan="2">討論
地點</td><td rowspan="2">綜科 328</td></tr>
<tr><td colspan="4">林辰燁 109810001
王姿伶 109810006
楊其康 109810021
陳勁旗 109810022
許喆堯 109810031</td></tr>
<tr><td colspan="3">本次討論議題：
1. 我們懷疑光害(室內、室外)是一個主要影響車子晃動的因素，因此會下次要加以實驗。
2. 更新硬體設備-MSI 顯卡 GTS960M，測試這樣算力是否夠。</td><td colspan="3">前次會議議題處理進度
1. 經由測試，發現 purepursuit 轉彎效果比較好、carrot chasing 直走效果比較好，因此我們將這兩個演算法結合，並加以測試車子擺盪幅度。</td></tr>
<tr><td colspan="6">本次會議結論：
1. 開始處理數據並分析，各自分配專題報告部分。
2. 找出車子穩定之最佳演算法，將 purepursuit+carrot chasing 結合。</td></tr>
<tr><td>指導老師簽名</td><td colspan="2"></td><td>填表紀錄者簽名</td><td colspan="2"></td></tr>
<tr><td colspan="6">注意事項:
1.本表的目的是有效的在專題製作時，幫助全組掌握分工與進度。
2.無任何具體討論內容、結論時不需填寫紀錄表。
3.本表填完及簽名完建議將其立即掃描為 jpeg 檔形式，以利製作完整報告時置入其中。</td></tr>
</table>

國立臺北科技大學機械系大學部實務專題會議記錄表

<table>
<tr><td>指導老師</td><td>許志明</td><td>日期</td><td>112 年 09 月 14 日</td><td>次數</td><td>第 10 次</td></tr>
<tr><td colspan="4">參與討論組員姓名以及學號(可打字或親簽)</td><td rowspan="2">討論地點</td><td rowspan="2">綜科 328</td></tr>
<tr><td colspan="4">林辰燁 109810001
王姿伶 109810006
楊其康 109810021
陳勁旗 109810022
許喆堯 109810031</td></tr>
<tr><td colspan="3">本次討論議題：
1. 因應室內室外問題，我們將車子搬至戶外試驗，然而車子明顯不穩，因此我們懷疑是光害干擾感測器，造成電腦控車不穩。
2. 專題精修程度，以及文案內容。</td><td colspan="3">前次會議議題處理進度
1. 實作時我們正式從室內搬至室外運行，發現室外實作，車子移動較不穩定。</td></tr>
<tr><td colspan="6">本次會議結論：
1. 使用新的硬體設備測試，發現其效能沒有原本的好，因此決定仍用原本的裝置進行追蹤。
2. 專題實作能精修的部分目前已告一段落，因此我們決議專題目前可以設一個中止點，開始著手文案部分，未來若要補強改善會隨時進行。</td></tr>
<tr><td colspan="2">指導老師簽名</td><td></td><td colspan="2">填表紀錄者簽名</td><td></td></tr>
<tr><td colspan="6">注意事項:
1.本表的目的是有效的在專題製作時，幫助全組掌握分工與進度。
2.無任何具體討論內容、結論時不需填寫紀錄表。
3.本表填完及簽名完建議將其立即掃描為 jpeg 檔形式，以利製作完整報告時置入其中。</td></tr>
</table>

National Taipei University of Technology Department of Mechanical Engineering Capstone Summary Course Students' Out-of-School Exhibition Visit Experience Record and Proof

Visit to the exhibition:

<table>
<tr><td>Full Name of the Exhibition</td><td colspan="3">Taipei International Machine Tool Show</td></tr>
<tr><td>Visit Date</td><td>March 10, 112th Year</td><td>Exhibition Venue or Hall Name</td><td>Taipei Nangang Exhibition Hall 1, 2, Taipei World Trade Center Hall 1</td></tr>
<tr><td>Class/Student ID/Name</td><td colspan="3">Mechanical Engineering 4A/109810001/林辰燁
Mechanical Engineering 4A/109810006/王姿伶
Mechanical Engineering 4B/109810021/楊其康
Mechanical Engineering 4B/109810022/陳勁旗
Mechanical Engineering 4B/109810031/許喆堯</td></tr>
<tr><td>Title of our project</td><td colspan="3">Smart Shopping cart</td></tr>
</table>

Summary of Exhibition Visit

<table>
<tr><td colspan="2">Description of Exhibition Contemt：</td></tr>
<tr><td>Organizer or Co-organizer</td><td>Taiwan External Trade Development Council (TAITRA)
Taiwan Association of Machinery Industry (TAMI)</td></tr>
<tr><td>Nature of Exhibition Content: (Select or specify)</td><td>☑Insustrial、□Household、□Defense、□Medical、☑Automotive、□3C、□Other:_________</td></tr>
<tr><td>Relvence to Our Project (select)</td><td>☑Highly relevent、□Partially relevant、□Not relevant</td></tr>
<tr><td colspan="2">Visit impressions and Thoughts：</td></tr>
<tr><td colspan="2">During the exhibition visit to Nangang and World Trade Center, various types of robotic arms and self-driving cars were on display. We were particularly intrigued by a self-driving forklift that automatically detected and lifted a cardboard box, providing us with a unanimous consensus on our project goals. After exploring many exhibitors, we observed that although companies produced similar items such as flexible robotic arms and precise CNC machinery, they all aimed to combine these elements with technology to optimize,</td></tr>
</table>

make them intelligent, and enhance precision. This experience provided us with more technological insights into the control of our project - the autonomous vehicle.
Substantial Gains for My Practical Project
1 Impact Recognition: The use of depth cameras in the project has been more versatile. 2. Autonomous Vehicle: Seeing many practical scenarios of self-driving cars enabled us to conduct deeper research and learning about self-constructed systems and ROS robot development systems. 3. Increased awareness of the industry, using practicality and feasibility as the primary premises for implementing the project, and adding creativity in the process.

Attach the DM image or scanned file of the exhibition event.

A group photo of all project members during the on-site visit.

(If it's not possible for the entire group to visit together, please provide individual photos as evidence.)