Smart Patrol Robot
- Lost & Found
FYP Final Report

SUPERVISOR
Dr. TW Chim

PRESENTED BY
Cheung Ka Yuk Rio 3035565889

GROUP MEMBERS
Chan Pik Sui Briony 3035606657
Leung Joyce Hoi Ying 3035569720
Siu Choi Hei Daniel 3035566871

Date of submission: 18th April 2022
Abstract

The number of automated robot applications has been increasing in recent decades. The improvement in technology has reduced the cost and increased the efficiency and productivity of automated robots. Moreover, the advanced technology has widened the potential usage of robot to solve different kinds of problems. This project, in the cooperation of Hong Kong International Airport, would like to explore the usage of automated robot in the airport. It aims to implement a lost and found system with the help of robotic and AI technology. The project is progressing as scheduled. We have successfully implemented a lost and found system using the smart patrol robot. The performance of the system reached our expectation. And the future plan is suggested to further increase its accuracy and include more functionalities.
Acknowledgement

First of all, I am grateful to the Department of Computer Science, and our supervisor, Dr. TW Chim, for giving me an opportunity to work on the “Smart Airport Project”, and the support offered throughout the process.

Also, I would like to acknowledge the staff from Airport Authority Hong Kong, who has explained the current needs and the expectation from AAHK.

Moreover, I would like to acknowledge the technician supports from AAHK, who are going to carry a training session about the smart patrol robot.

I also appreciate the help from my teammates.

Lastly, I would like to show the gratitude to Miss Mable Choi for helping me regarding the report.
Table of contents

Abstract i
Acknowledgement ii
Table of contents i
List of Figures iii
List of Tables iv
Abbreviations v
1. Introduction 1
   1.1 Overview of Smart Airport 1
   1.2 Automated Robot in Airport 1
   1.3 Smart Patrol Robot 1
   1.4 Project Initiative 2
   1.5 Project Objective 3
   1.6 Expected Project Contribution 3
   1.7 Outline of the report 3
2. Methodology 4
   2.1 Introduction 4
   2.2 Overview 4
   2.3 Object Detection 4
   2.4 AI Model 5
   2.5.1 Lost Item Identification 6
   2.5.2 Tracking System 7
   2.5.3 Object Tracking 8
   2.5.4 Intersection over Union 9
   2.5.5 Image Similarity 11
   2.6.1 Database System Management System 12
   2.6.2 MongoDB 12
   2.7 Report Channel 13
   2.8 Summary 13
3. Experiment and Result 14
   3.1 Overview 14
   3.2 Comparison of AI Models 14
   3.2.1 Region-Proposal & Regression Based Model 15
   3.2.2 Architecture of YOLO 16
   3.2.3 Architecture of SSD 16
   3.2.4 Architecture of Faster R-CNN 17
   3.2.5 Comparison Result 17
   3.3 Patrol Robot Hardware 23
   3.3.1 Hikvision Camera 23
   3.3.2 Base Module 24
   3.4.1 Map Drawing 26
3.4.2 Checkpoints & Patrol Routes 27
3.5 Lost Item Identification 27
3.5.1 Checkpoints Checking 28
3.5.2 Patrol Path Checking 29
3.5.3 Database Update 30
3.5.4 Moving / Stationary Object 31
3.5.5 Image Similarity Accuracy 32
3.6.1 Telegram Channel 34
3.6.2 Log message 34
3.7 Custom YOLO Model Training 34
3.8 Other Work & Abandoned Design 36
3.8.1 Video / Image Mode 36
3.8.2 Usage of Different AI Model 37
3.8.3 Usage of GPU & TPU 37
4. Future Plans and Conclusion 37
   4.1 Overview 37
   4.2 Web Interface 37
   4.3 Improvement of 1st Class Detection 38
   4.4 Further Training on Custom YOLO Model 38
   4.5 Conclusion 39
References 40
Appendices 42
   Appendix A 42
   Appendix B 43
   Appendix C 44
   Appendix D 45
   Appendix E 48
   Appendix F 52
List of Figures

Figure 1. 1 Structure of a Smart Patrol Robot ......................................................... 2
Figure 1. 2 Different types of patrol robot ............................................................... 2

Figure 2. 1 Lost & Found System Architecture .......................................................... 4
Figure 2. 2 YOLO sample 1 .................................................................................... 5
Figure 2. 3 YOLO sample 2 ..................................................................................... 5
Figure 2. 4 Sample Scene in Airport ........................................................................ 6
Figure 2. 5 Checkpoints with object ID ..................................................................... 7
Figure 2. 6 first frame .............................................................................................. 8
Figure 2. 7 second frame .......................................................................................... 8
Figure 2. 8 clipped first frame .................................................................................. 9
Figure 2. 9 clipped second frame ............................................................................. 9
Figure 2. 10 second frame with suitcase gone .......................................................... 9
Figure 2. 11 second frame with different suitcase .................................................... 9
Figure 2. 12 Intersection of two boxes ..................................................................... 9
Figure 2. 13 Formula of IoU computation ................................................................. 10
Figure 2. 14 Sample ................................................................................................. 11
Figure 2. 15 Equation of SSIM calculation ............................................................... 11
Figure 2. 16 alert message ....................................................................................... 13

Figure 3. 1 Example of region-proposal based framework ........................................ 15
Figure 3. 2 Example of regression-based framework ............................................... 15
Figure 3. 3 Architecture of YOLO .......................................................................... 16
Figure 3. 4 Network Architecture of SSD ................................................................. 16
Figure 3. 5 Architecture of Faster R-CNN ............................................................... 17
Figure 3. 6 Sample Photo used in the tests ............................................................... 19
Figure 3. 7 Result Photo generated by YOLO-v5s .................................................... 20
Figure 3. 8 Result Photo generated by SSD512 ...................................................... 21
Figure 3. 9 Result Photo generated by Faster R-CNN ............................................. 22
Figure 3. 10 Camera of the robot .......................................................................... 23
Figure 3. 11 RGB channel view of top camera ...................................................... 24
Figure 3. 12 communication with base module ..................................................... 25
Figure 3. 13 decoding result ................................................................................. 25
Figure 3. 14 Map drawn with Robot on UNODOPO Navi Studio ......................... 26
Figure 3. 15 Checkpoints overview .................................................................... 27
Figure 3. 16 Trial run 1 result ............................................................................... 28
List of Tables

Table 2. 1 Rules about lost object determination ................................................. 7
Table 2. 2 Sample tracking record ........................................................................ 8

Table 3. 1 Comparison of AI Models ..................................................................... 18
Table 3. 2 Communication Protocol of the patrol robot ......................................... 24

Figure 3. 17 Report message (2nd class object) .................................................... 29
Figure 3. 18 Report message (1st class object) ..................................................... 30
Figure 3. 19 Database before/after update ............................................................. 30
Figure 3. 20 Determine the movement of the ball by IoU ........................................ 31
Figure 3. 21 Image similarity result 1 ...................................................................... 32
Figure 3. 22 Image similarity result 2 ...................................................................... 33
Figure 3. 23 Message log example ......................................................................... 34
Figure 3. 24 Yolov5s versus custom model example 1 .......................................... 35
Figure 3. 25 Yolov5s versus custom model example 2 .......................................... 35
Figure 3. 26 Yolov5s versus custom model example 3 .......................................... 36

Figure 4. 1 Example of a detection result ............................................................... 38
<table>
<thead>
<tr>
<th>Abbreviations</th>
<th>Acronyms</th>
</tr>
</thead>
<tbody>
<tr>
<td>AA / AAHK</td>
<td>Airport Authority Hong Kong</td>
</tr>
<tr>
<td>HKIA</td>
<td>Hong Kong International Airport</td>
</tr>
<tr>
<td>HKU</td>
<td>The University of Hong Kong</td>
</tr>
<tr>
<td>Innowing</td>
<td>The Tam Wing Fan Innovation Wing</td>
</tr>
<tr>
<td>T1</td>
<td>Terminal 1</td>
</tr>
<tr>
<td>YOLO</td>
<td>You Only Look Once</td>
</tr>
<tr>
<td>SSD</td>
<td>Single Shot MultiBox Detector</td>
</tr>
<tr>
<td>R-CNN</td>
<td>Region-based Convolutional Neural Network</td>
</tr>
<tr>
<td>RPN</td>
<td>Region Proposal Network</td>
</tr>
<tr>
<td>FPS</td>
<td>Frame Per Second</td>
</tr>
<tr>
<td>mAP</td>
<td>Mean Average Precision</td>
</tr>
<tr>
<td>IoU</td>
<td>Intersection over Union</td>
</tr>
<tr>
<td>SSIM</td>
<td>Structural Similarity Index Measure</td>
</tr>
<tr>
<td>DBMS</td>
<td>Database Management System</td>
</tr>
<tr>
<td>GPU</td>
<td>Graphics Processing Unit</td>
</tr>
<tr>
<td>TPU</td>
<td>Tensor Processing Unit</td>
</tr>
</tbody>
</table>
1. Introduction

1.1 Overview of Smart Airport

Smart Airports refers to airports that rely on the use of connected technologies [1]. Due to the rapid growth of modern technology. Many traditional works of the airports are replaced by some advanced technology to achieve three major goals —— reduce cost by saving manpower, improve efficiency and enhance user experience. Automated robot is regarded as one of the major development fields in airports all over the world. The number of robots inside airports has increased significantly in recent decades [2].

1.2 Automated Robot in Airport

Automated robots in airport are specialized to serve different purposes. The most common type of robots found in airport is the cleaning robot. More advanced airports may possess robots with facial recognition function, thermal detection function etc. Moreover, some robots are built in with strong artificial intelligence to support instant interaction with airport users. Singapore, South Korea, and Japan are some examples with high robot density (See Appendix A for further detail). Among different kinds of robots possessed by HKIA, two smart patrol robots are given to HKU to explore their potential application.

1.3 Smart Patrol Robot

Currently, there are four smart patrol robots working in the airport area. Their major duty is to collect some environmental data including lighting condition, temperature, humidity, Wi-Fi signal, iBeacon signal in the airport.
Figure 1.1 above shows the basic structure of a patrol robot and Figure 1.2 shows different models of patrol robot [3]. For example, in Figure 1.2, the middle one with locker is designed to transport goods and the righter most one is designed to collect/deliver tray in the restaurant. In general, the base module of the robot is responsible for navigation while the top module is customized based on the job duty of the robot. (See Appendix B for more hardware detail of the robot) For the tradition patrol robot, several checkpoints are preassigned to it and the robot would travel to each checkpoint to collect the environmental data. This project would like to extend the functionality of the smart patrol robot such that it could do more other than collecting data. As visitors have a tendency to misplace their personal belongings in the terminal, a robot with the capability of lost item classification, could be a solution.

1.4 Project Initiative

Being one of the busiest airports in the world, HKIA has a very highest visitor flowrate. It is not an easy task to monitor such a large population and their personal belongings. Currently, HKIA detect lost item through patrols and report from visitors [4]. However, it is not obviously to
recognize a lost item among thousands of items, especially when the area covered by the airport is very large. Since the patrol robots keep patrolling the airport continuously, building a lost item detection function inside the robot can help to identify lost item as early as possible. Furthermore, this lost and found system may help to recognize suspicious item as well since both of them have some similar features. Hence, the idea of lost and found system is initiated and eventually become the topic of this smart airport project.

1.5 Project Objective

In this project, we aim to implement a lost item identification function in the smart patrol robot to improve the effectiveness of recognizing unclaimed/lost items in the airport. The scope of the project includes three parts:

(1) identify object type in the image captured by the built-in camera,
(2) classify if the detected object should be considered as a lost item,
(3) implement an effective system for the proper handling of the lost items.

1.6 Expected Project Contribution

It is hoped that this project can reduce the manpower required for patrolling in T1. Also, early detection of unclaimed objects can help prevent crime such as theft and long-time exposure of illegal objects. At the same time, it can increase the chance for airport users getting back their lost belongings securely as well as improving their user experiences. Last but not least, the success of this project will show the potential ability for robots performing different tasks in order to promote the application of automated robots.

1.7 Outline of the report

The report is structured into four chapters. The first chapter gives an introduction of the project, including the background, project initiative, objective, and the expected contribution.

Chapter two illustrates the methodology used in this project, including the expected work flow of the product and some illustration of the used technique such as AI model, lost item identification, IoU & image similarity and report channel.

Chapter three presents the experiment and the corresponding results, including the comparison result of different AI models, lost item identification algorithm, robot set up, system performance and some other work and abandoned design in this project.

Chapter four concludes the report and describe the future plan of the project.
2. Methodology

2.1 Introduction

This chapter presents the technology and method this project will use to build the system. In this project, the robot has to detect lost items and report them to the lost and found system for further handling.

2.2 Overview

![Lost & Found System Architecture](image)

*Figure 2.1 Lost & Found System Architecture*

Figure 2.1 shows the overall structure of the lost & found system. The patrol robot would capture images by a built-in camera and perform object detection with the help of AI model. The base module would perform pathing and positioning. After that, the lost item classification algorithm would be applied to determine whether or not the target object should be reported through the alert channel. By combining all the components, a lost & found system is built up. Different components of the system will be illustrated one by one in this report.

2.3 Object Detection

Object detection is a computer vision technique used for identifying objects in images or videos [5]. When humans are given an image, people, animals, objects, scenes, and other visual details within the image can be spotted and classified easily. But it is hard to know how the
classification is performed by a human. Similarly, it is hard to define a clear set of rules for the computer to recognize objects inside an image. Hence, object detection is the key output of combining deep learning and machine learning algorithms. Both algorithms require a huge amount of data sets and a long period of time to train the system. Instead of building a new object recognition system, this project will make use of some AI models which are available online to serve the purpose.

2.4 AI Model

AI model is a software program that has been trained on a set of data to perform specific tasks such as voice recognition, pattern recognition, object detection etc. [6]. In this project, object detection is the target function required. Currently, there have been plenty of AI models serving this purpose such as YOLO, Amazon Rekognition, RetinaNet, Mask R-CNN etc. In this project, YOLO will be used to perform object detection.
Figures 2.2 & 2.3 are examples of YOLO object detection application [7]. Given an image, the AI model can return the object type with the corresponding probability. After recognizing the object type successfully, the robot will be able to perform lost item identification.

2.5.1 Lost Item Identification

In order to construct a robust and efficient lost and found system, it is necessary to define what is a “lost item” and prepare a complete set of rules for the robot to make the correct judgement. For example, object type and the distance between the object and the surrounding people are some factors that the robot need to take into consideration when making judgment.

![Image of airport scene with highlighted objects](image)

Figure 2.4 Sample Scene in Airport

To illustrate the idea, Figure 2.4 shows a sample scene with three highlighted objects a, b (a mobile phone), and c. To compare object a and b, it is obvious that only object b should be considered as a lost item although both of them have a similar distance with the nearby person. It is because a mobile phone should not appear on the ground in normal circumstance. However, for object c, even if there are no people nearby, it is not sufficient to affirm that it must be a lost item because it is very common for a baggage to appear alone for a short moment in the airport area (e.g., owner went to washroom). Hence, the robot should keep track of the baggage for a period of time for identification. Therefore, different rules should be applied to different types of objects when making judgement.
Table 2.1 Rules about lost object determination

<table>
<thead>
<tr>
<th>Object type</th>
<th>1st class object</th>
<th>2nd class object</th>
<th>3rd class object</th>
</tr>
</thead>
<tbody>
<tr>
<td>trigger rule</td>
<td>seen on floor</td>
<td>unmoved for long period</td>
<td>never report</td>
</tr>
<tr>
<td>determination time</td>
<td>instant</td>
<td>long</td>
<td>//</td>
</tr>
<tr>
<td>example</td>
<td>phone, wallet</td>
<td>luggage, backpack</td>
<td>chair, trolley</td>
</tr>
</tbody>
</table>

Table 2.1 above shows a set of simplified rules for the identification. Every time when the robot performs object detection, the system has to classify the object into one of the classes mentioned above and apply different rules on them for analysis. For example, if the object has been classified into 1st class, and it is currently on the floor, this object will be directly reported to the “report channel” (Further elaboration in Section 2.7). If it is 3rd class object, the robot will simply ignore it. But for the 2nd class object, as mentioned above, sometimes it is not sufficient to make judgement instantly, therefore, it is necessary to have an efficient way to keep track of the status of the detected object.

2.5.2 Tracking System

The tracking system is responsible for tracking the status of 2nd class objects that the robot has detected. When the robot reaches a checkpoint, it will revolve and look for any kinds of 2nd class objects and store its relative location with respect to the checkpoint. Each checkpoint area will be divided into several sectors and each sector refers to 1 direction from the checkpoint. The robot will investigate the sectors one after another. It is because the robot has to focus on those stationary 2nd class objects. However, it is hardly achieved if the robot itself is moving and the camera is shifting. In the following example, each checkpoint is divided into 4 sectors.

![Figure 2.5 Checkpoints with object ID](image)

Figure 2.5 shows an example of how the robot keep track of the objects. Suppose the robot is assigned with 3 checkpoints (A, B, C). The robot will move from A to B, B to C, C to A and A
to B repeatedly. When it reaches any checkpoint, the robots will look for the potential lost items nearby (1, 2, 3, 4 in the figure refer to the 2nd class objects being tracked).

Table 2.2 Sample tracking record

<table>
<thead>
<tr>
<th>object ID</th>
<th>Checkpoint</th>
<th>Bounding Box (X-min, Y-min, X-max, Y-max)</th>
<th>First Found Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A-0</td>
<td>(1000, 400, 1150, 600)</td>
<td>14:32:40</td>
</tr>
<tr>
<td>2</td>
<td>A-3</td>
<td>(200, 350, 300, 500)</td>
<td>14:32:45</td>
</tr>
<tr>
<td>3</td>
<td>B-1</td>
<td>(900, 250, 1000, 325)</td>
<td>14:34:20</td>
</tr>
<tr>
<td>4</td>
<td>C-0</td>
<td>(550, 350, 700, 500)</td>
<td>14:36:35</td>
</tr>
</tbody>
</table>

Table 2.2 shows a simplified version of the tracking record. The robot will try to match the nearby objects with the existing tracking record according to the location and other information. “Bounding Box” indicates the pixel coordinate of the object inside the image captured by the camera (example in Figure 2.6 and Figure 2.7). If there is no corresponding match (i.e., the object has been taken away by the owner), the object record will be removed from the tracking system. On the other hand, if there is a match and the time being counted from “First Found Time” has exceeded the time threshold, the object will be identified as a lost item and reported to the reporting channel. Section 2.5.3 illustrates how the robot can keep track of an object.

2.5.3 Object Tracking

Suppose the robot reaches one of the checkpoints and faces towards one direction. The robots would identify those stationary 2nd class objects captured by the camera (shown in Figure 2.6). Assume it is the first time that the robot sees the highlighted suitcase. It would store the data of the suitcase and continue patrol. After a while, when the robots come back to this checkpoint, it detects a suitcase appeared in this sector again (shown in Figure 2.7). Then, Intersection over Union (IoU) of the bounding box would be calculated to determine if two suitcases are at the same position (Detail of IoU calculation will be discussed in Section 2.5.4). After that, the robot should calculate the image similarity to determine if two suitcases are the same one.
As shown in Figure 2.8 and Figure 2.9, the robot would clip the target items from the images and compare their similarity. Since two images are taken from the same location and direction, the similarity should be high if two suitcases are the same one. Hence, the robot can now keep track of the candidate items. When the time duration of the candidate exceeded the preset threshold, the robot would send an alert message in the report channel. On the other hand, if the candidate item is no longer available (i.e., Figure 2.10 and Figure 2.11), the robot would update its database.

2.5.4 Intersection over Union

![Figure 2.12 Intersection of two boxes](image)
In order to speed up the computation and increase the accuracy, the robot compute the image similarity only when the IoU of 2 bounding boxes exceeds the threshold. IoU is a term used to describe the extent of overlap of two boxes. Greater the region of overlap, greater the IoU is [8]. Figure 2.12 shows an example of two intersected boxes. In this case, two boxes would be the bounding boxes of the objects inside the image. Since two images are captured at the same position and orientation. The stationary object inside two images should also appear at a very similar position. Therefore, the intersection between two bounding boxes can be used to determine if two objects are the same one.

\[
\text{IOU} = \frac{\text{Area of Intersection of two boxes}}{\text{Area of Union of two boxes}}
\]

Figure 2.13 illustrates how the IoU is computed. However, only IoU itself is not sufficient to determine if two objects are the same one. Because it is possible to have two different objects placed at the same position in different time slot. Therefore, after the IoU is computed to be greater than the threshold, the robot would further compute the image similarity to guarantee the result.
2.5.5 Image Similarity

Figure 2.14 shows a sample scene with two different suitcases appear at the same position. IoU of two bounding boxes are greater than the threshold and image similarity can be used to distinguish the difference. In this project, SSIM (structural similarity index measure), d-Hash, w-Hash are used.

Figure 2.15 shows how SSIM is calculated. \( x \) and \( y \) are two images with common size \( N \times N \). The SSIM index is used for measuring the similarity between two images. The SSIM predicts image quality based on an initial uncompressed or distortion-free image as reference. It tells how far away an image is from its original reference image more aligned with the human perceptual

\[
SSIM(x, y) = \frac{(2\mu_x \mu_y + c_1)(2\sigma_{xy} + c_2)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)}
\]

with:
- \( \mu_x \) the average of \( x \);
- \( \mu_y \) the average of \( y \);
- \( \sigma_x^2 \) the variance of \( x \);
- \( \sigma_y^2 \) the variance of \( y \);
- \( \sigma_{xy} \) the covariance of \( x \) and \( y \);
- \( c_1 = (k_1 L)^2 \), \( c_2 = (k_2 L)^2 \) two variables to stabilize the division with weak denominator;
- \( L \) the dynamic range of the pixel-values (typically this is \( 2^{\text{bits per pixel}} - 1 \));
- \( k_1 = 0.01 \) and \( k_2 = 0.03 \) by default.

Figure 2. 15 Equation of SSIM calculation
system. SSIM is designed to improve on traditional methods such as peak signal-to-noise ratio (PSNR) and mean squared error [9].

D-hash and w-Hash are the technique of using image hashing to compute the image similarity. The hashing value of two similar images would output a result close to 1.

### 2.6.1 Database System Management System

A database is a large collection of inter-related data. A Database Management System (DBMS) includes one or more than one database(s) and a set of software programs that store, access, and run queries on data. Users can create, read, update, and delete data in the database through the DBMS [10]. In this project, a DBMS is required to manage the data of the candidate items such as the object ID, bounding box coordinate, time duration etc. In general, there are two types of databases. Relational-database and non-relational database. This project would use MongoDB (Non-relational) instead of MySQL (Relational).

### 2.6.2 MongoDB

The reason of using MongoDB is because the relation between different datasets is very insignificant in this project. There is not much benefit from using a relational database (MySQL). In contrast, there are several advantages from using a non-relational database (MongoDB). One of the advantages is that MongoDB is more flexible than MySQL. In MySQL, attributes are fixed and enforced. All the data from the same table has to follow the same set of formats. On the other hand, MongoDB allows best-suited data type storage. Each object can have different fields which make implementation easier and faster.

Moreover, MongoDB is more efficient for large data sets since all the data is stored in one document. This can speed up the processing time.
2.7 Report Channel

Since the robot is not responsible of picking up the lost item, what it does is to alert the airport’s staffs to handle the target item. Therefore, a report channel is needed between the robot and the staff on duty. The report channel in this project would be established using Telegram. After the robot detects a lost item, it would package the data of the target item and send a message with a photo to the report channel (shown in Figure 2.16). Hence, the airport staff can handle the lost item accordingly.

2.8 Summary

If the robot works to expectation, it will be capable of patrolling the designated area and be able to recognize objects such as baggage, wallet, phone etc. It will only store the data of the item which has a reasonably high probability of being a lost item. To achieve this, AI model will be used to analyze the images captured by the camera. Lost item identification can take place to determine which item should be classified as a “lost item” and reported. If there is an identified lost item, the robot would document the record and notify the staff. This helps the identification of lost items in the airport as early as possible.
3. Experiment and Result

3.1 Overview

This chapter presents the progress and result of the project. First, the comparison of different AI models would be demonstrated. Then the hardware part of the robot including camera and base module would be illustrated. After that, the performance of the lost item identification algorithm and the report channel would be evaluated. Finally, other work being done and some of the abandoned designed would be discussed.

3.2 Comparison of AI Models

As mentioned in Section 2.4, AI model plays an important role in this project. Some researches and experiments have been done so as to find the model that is the most suitable to this project. Currently, there are two types of detection framework being used for object detection, region-proposal based and regression based.
3.2.1 Region-Proposal & Regression Based Model

Since an image can consist of multiple different objects, it is not feasible to pass the image into the neural network directly. Therefore, region-proposal based framework (e.g., R-CNN) first extracts the potential regions using selective search. It means recursively combining the smaller similar regions into larger ones [11]. After that, each proposed region would be analyzed by a neural network to identify the object type inside the region.

![Figure 3.1 Example of region-proposal based framework](image1.png)

Figure 3.1 shows the process of region extraction. Depends on the image, the number of proposed regions can be very large. Therefore, long computation time is required for the region-proposal based method.

![Figure 3.2 Example of regression-based framework](image2.png)

On the other hand, regression-based framework (e.g., YOLO & SSD) adopts an opposite approach. As shown in Figure 3.2, regression-based framework first divides the image into constant number of boxes and passes them into the neural networks. Since the number of boxes are constant, this method is much faster than the previous one.
To decide which of the AI models should be used in this project, there are several factors taken into the consideration such as the difficulty of use, computational speed, accuracy of the model. YOLO, SSD, Faster R-CNN have been studied and compared in this project.

3.2.2 Architecture of YOLO

Figure 3.3 Architecture of YOLO

Figure 3.3 shows the architecture of YOLOv5. The YOLO network consists of three main pieces including backbone, neck and head. Backbone is a convolutional neural network that aggregates and forms image features at different granularities. Neck is a series of layers to mix and combine image features to pass them forward to prediction. And head consumes features from the neck and takes box and class prediction steps. The YOLO model is the first object detector to connect the procedure of predicting bounding boxes with class labels in an end-to-end differentiable network [12].

3.2.3 Architecture of SSD

Figure 3.4 Network Architecture of SSD
SSD extracts feature maps and applies convolution filters in object detection. As shown in Figure 3.4, the feature maps are first extracted using VGG-16, which is a convolutional neural network model for object classification. Boundary boxes are used to bound the objects in the image. Each object is then detected using the Conv4_3 layer, where a total of 5776 predictions are made by separating the layer into cells and making 4 predictions for each cell. In addition, small convolution filters are used to calculate the location and class scores instead of a region proposal network. Predictions are made by applying the filters to each cell in the feature maps [13].

3.2.4 Architecture of Faster R-CNN

![Figure 3.5 Architecture of Faster R-CNN](image)

Figure 3.5 illustrates the architecture of Faster R-CNN. The RPN module in Faster R-CNN generates region proposals. The module applies the concept of neural networks, guiding the Fast R-CNN module to locate the objects in the image. It generates feature maps from convolutional layers shared with the Fast R-CNN module. Then, region proposals are generated from feature maps of the previous shared convolution layer. Each proposal is further separated into anchor boxes, which the number of boxes varies in the scale of the aspect ratio. Feature vectors are extracted from the regions with the use of RoI Pooling. In addition, the vectors are passed to the Fast R-CNN module for object classification [14].

3.2.5 Comparison Result

Three models mentioned above were tested and examined under two parameters, accuracy, and processing speed. In terms of accuracy, the mean average precision (mAP) of each model is
considered. The mAP is evaluated by the mean accuracy in detecting different types of items. High mAP indicates that the model is accurate in detecting objects.

Moreover, the inference time or the frame per second (FPS) rate of the models demonstrates its processing speed, where the inference time is the reciprocal of the FPS rate. Since the models are used for real-time detection, the processing time required affects the efficiency of object identification. Short inference time or high FPS rate shows the efficiency of the model in object recognition.

Table 3.1 Comparison of AI Models [13] [14] [15].

<table>
<thead>
<tr>
<th>Model</th>
<th>mAP(%)</th>
<th>FPS</th>
<th>Inference Time (s/image)</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLO-v5s</td>
<td>56.0</td>
<td>455</td>
<td>0.0022</td>
</tr>
<tr>
<td>SSD512</td>
<td>76.9</td>
<td>19</td>
<td>0.0526</td>
</tr>
<tr>
<td>Faster R-CNN</td>
<td>76.4</td>
<td>5</td>
<td>0.2</td>
</tr>
</tbody>
</table>

Table 3.1 illustrates the accuracy and the processing speed of each model. According to the table, YOLO-v5s has the fastest processing speed among the three models for its short inference time. On the other hand, SSD512 and Faster R-CNN are more accurate in object detection since their mAP is higher than that of YOLOv5s. Considering the findings from the table, it is concluded that YOLOv5s is a better choice in terms of speed while SSD and Faster R-CNN are better choices in terms of accuracy.

YOLO-v5s, SSD512 and Faster R-CNN were tested for object detection. One of the sample photos (As shown in Figure 3.6) would be discussed in this section. The middle part of the photo is relatively crowded which can test the models’ ability in object detection.
Figure 3. Figure 3.6 and Figure 3.7, and Figure 3.9 show the result photo generated by YOLOv5s, SSD512 and Faster R-CNN respectively. The objects detected by the model will be marked by bounding boxes. A bounding box is usually used to locate an object spatially. It is a rectangular box that bounds the upper-left corner and the lower-right corner of the object [16]. Some of the results would include the confidence score to the results. The confidence score refers to the probability that the result generated by the model is correct and satisfactory [17].
As shown in Figure 3.7, YOLO-v5s gave accurate predictions in the crowded area where the targeted objects (such as suitcases and handbags) and the people were identified. It performed well on identifying targeted objects. This indicates that the YOLO model has a high accuracy in object classification of targeted objects.
Figure 3.8 illustrates the performance of SSD512 in object detection. Similar to YOLO-v5s, the model detected people and the targeted objects appearing in the sample photo accurately even in the crowded environment. However, it could not identify surrounded objects. For instance, neither the large sign nor the environment was detected. Compared with the performance of YOLO-v5s, SSD detected fewer surrounding objects such as the TV appeared in the photo. It shows that SSD512 has the ability of detecting targeted objects in photos precisely though its performance is not as good as that of YOLO-v5s.
Figure 3.9 shows the performance of Faster R-CNN in object detection. According to the figure, the model identified target objects and surrounding objects precisely. It recognized the whole scene as the airport with ceiling and floor was identified. Moreover, the objects were recognized in detail. Small objects like shirts, and the gender of the people were detected. Nevertheless, the model is not specific in recognizing objects in crowded areas. The group of people was detected as a group instead of each person one by one. This illustrates the ability of the model in detailed identification and its performance in identifying objects.

As mentioned in the previous section, YOLO-v5s has the highest FPS rate among the three models which illustrates its fast-processing speed in object detection. SSD512 and Faster R-CNN process the photos much slower than YOLO-v5s. Therefore, it is expected that YOLOv5s is able to generate results at the fastest speed. Considering the accuracy of the results, all of the 3 models give precise results. Among the 3 models, Faster R-CNN has the highest accuracy for detecting the details in the photo precisely. Despite the relatively high accuracy of Faster R-CNN in identifying all the objects in the scene, it requires a long processing time due to the long inference time of the model. Based on the results, it is concluded that YOLO-v5 would be used for object detection due to its fast and precise performance.
3.3 Patrol Robot Hardware

The Patrol Robot is the main body of the patrol robot. In order to make the system available, each component has to perform its own task properly. This section will introduce the camera used by the robot, and how the system manages to communicate with the base model of the patrol robot.

3.3.1 Hikvision Camera

![Figure 3.10 Camera of the robot](image)

In total, there are two cameras installed on the patrol robot, top-camera (thermal camera) and front-camera. The original plan would be using the front-camera to perform image capturing. However, the password used to access the front-camera is lost, while resetting the password could be troublesome and there is a risk to ruined the internal structure of the robot (reset password require uninstalling the camera from the robot). Hence, top camera is used instead. Figure 3.10 shows the current setting of the robot. The top camera is fixed at the top position and enforced to look frontward.

Top camera is a Hikvision thermal camera. There are two channels provided by the camera, IR (Infra-Red) channel and RGB (Red, Green, Blue) channel. In this project, the thermal detection function is not used. Currently, only RGB channel is used to capture the RGB image for performing object detection. Figure 3.11 below shows an example of the RGB channel view.
3.3.2 Base Module

<table>
<thead>
<tr>
<th>Communication Interface</th>
<th>Baud Rate</th>
<th>Parity bit</th>
<th>Data Bit</th>
<th>Stop Bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS-232</td>
<td>115200bps</td>
<td>None</td>
<td>8</td>
<td>1</td>
</tr>
</tbody>
</table>

Since the robot is required to patrol in the designed area, and report to airport staff about the location of the lost item. Pathing and positioning are the important tasks in this project. To do that, the program need to communicate with the base module of the patrol robot. Table 3.2 shows the communication protocol used by the base module. A command with 28 bytes can be sent to the robot and a message with 30 bytes would be returned.
Figure 3.12 communication with base module

Figure 3.12 shows an example of the communication. A 28 bytes command is sent to get the current location of the robot. And the base module returns a 30 bytes message with the location stored inside. In order to get the actual value, the program need to decode the returned message according to the protocol (See Appendix D for more detail about the instructions that have been used in the project).

Figure 3.13 decoding result

Figure 3.13 shows the result of decoded message which refers to the current position of the robot.
3.4.1 Map Drawing

In order to test the robot, a test scene is required. The Tam Wing Fan Innovation Wing (Innowing) is chosen to be the test scene in this project.

![Figure 3.14 Map drawn with Robot on UNODOPO Navi Studio](image)

Figure 3.14 shows the map in innovation lab. Several map drawing sessions were conducted by the team at the open area of Innowing. UNODOPO Navi Studio was used to draw maps and plan patrolling routes. A smart patrol robot was controlled by a controller to scan the surroundings with lidar. Lidar, which stands for Light Detection and Ranging, measures variable distances with light in the form of a pulsed laser to detect objects around. A lidar system is in the base module of the robot. The scattered blue rays represent the scanning result of the lidar. The robot follows the route correctly after being instructed. The map would be used for testing the ability of the robot in patrolling and running the YOLO model and the classification algorithm.
3.4.2 Checkpoints & Patrol Routes

![Figure 3.15 Checkpoints overview](image)

*Figure 3.15* indicates the checkpoint location and direction in the patrol path. The robot would patrol according to the sequence A0, A1, B0, B1 repeatedly. In the actual application, a guide map similar to *Figure 3.15* will be given to the airport staff. Alert message would include the checkpoint information as the location of the detected item. Hence, the airport staff can refer to the guide map and look for the detected item accordingly. (Appendix E includes the location and camera view of each [checkpoint-direction])

3.5 Lost Item Identification

As mentioned in Section 2.5.1, the candidates are divided into 2 classes. In the code of lost item identification, main program is responsible for detecting 2nd class object (e.g., suitcase) at checkpoints and a thread is used to detect 1st class object (e.g., phone).
3.5.1 Checkpoints Checking

Figure 3.16 shows the result of trial run 1. In this trial, the thread was set to be inactive. The robot only performed lost item identification at the checkpoints. In Figure 3.16, only the result at checkpoint A-0 is shown. The database was empty before the trial started (i.e., pic 1). Therefore, when the robot first reached checkpoint A-0 (pic 2), it detected a new suitcase (green box) and updated its database. After a while, the robot came back to A-0 again (pic 3). It matched the current finding with the existing database. The robot then performed IoU and image similarity calculation (mentioned in Section 2.5.4 & 2.5.5). Since the suitcase fulfilled the criteria, the robot decided to report the item (red box). The result of the trial is consistent with the original expectation.
Figure 3.17 shows the result in Telegram. The robot sent a message and the first image (pic 2) of the object through Telegram. Hence, the airport staff can refer to the information provided to handle to lost item.

3.5.2 Patrol Path Checking

Apart from the main program, there is a thread which is responsible for checking the 1st class object all the time. The algorithm is similar to the main program but the thread would not keep track of the candidates. It would determine whether or not report the item at the first time it sees the objects. Figure 3.18 below shows the message of reporting a cellphone on the ground.
3.5.3 Database Update

Figure 3. 19 Database before/after update
Figure 3.19 shows an example of database update about the detected objects at checkpoint A-0 (red box). The orange box shows the information about the detected objects including the bounding box information (x-min, y-min, x-max, y-max), YOLO detection confidence, object class (name), object id, time of first detection, image path (cropped image used to calculate the image similarity), full image path (image being sent in alert message). After the robot has completed checking, it would update the information of the corresponding checkpoint. For example, if an object disappears, or it is reported, its information in the database would be deleted (shown in DB 2).

3.5.4 Moving / Stationary Object

As mentioned in Section 2.5.4, IoU is calculated in order to keep track of the object. It can be used to determine the status of the detected object (stationary or moving).

![Diagram](image)

Figure 3. 20 Determine the movement of the ball by IoU

Figure 3.20 shows an example of how IoU can be used to determine object status. Assume the status of the blue ball (in frame 1) would like to be determined. Several consecutive frames can be taken (in this example, only 2 frames are taken). Then, the bounding box of the ball is located. If the ball is stationary, IoU of two bounding boxes should be very close to 1. In contrast, if the ball is moving, IoU would decrease significantly. If an object is determined to be a moving object, the robot would not take this object into account since a moving object cannot be a lost item at the same time (e.g., a moving suitcase pulled by a passenger). In fact, this technique is not restricted to consecutive frames. The robot also uses this technique to keep track of the
candidates at the checkpoints. However, the threshold being used would be slightly smaller since the location of the robot may not be exactly the same when it reaches a certain checkpoint.

3.5.5 Image Similarity Accuracy

As mentioned in Section 2.5.5, a large IoU is not sufficient to guarantee that two objects are the same one. It is possible to have two different objects at the same position at different time intervals. In this project, there are 3 algorithms being used to calculate the image similarity which are SSIM, d-hash and w-hash.

Figure 3.21 shows how image similarity can be used to distinguish two different objects. In the example, “img1” and “img2” are different images refer to same suitcase while “img3” is referring a different suitcase. As shown in the result, if the threshold is set to be 0.8, using either 1 method can achieve the correct result. However, the approach of using only 1 method has a great defect which will be illustrated in the following example.
Figure 3.22 Image similarity result 2

Figure 3.22 shows another example with 2 different suitcases. Similar to Figure 3.21, “img4” and “img5” refer to the same suitcase and “img6” refer to a different suitcase. In the example above, it is noticed that “whash” is giving a very similar result when comparing “img4” versus “img5” and “img5” versus “img6”. This result is not desired. In the ideal case, \( \text{sim}(\text{img4}, \text{img5}) \) should output a value close to 1 while \( \text{sim}(\text{img4}, \text{img5}) \) should output a value far from 1. Therefore, it shows that w-hash may raise false positive result sometimes.

In fact, all 3 methods are not perfect. It is true that given two similar images, each method would output a high value. However, it is not guarantee that, a high value must come from two similar images. Two images referring different suitcases could still result in high value as shown in the example above (w-hash output 0.867 when comparing “img5” & “img6”). Therefore, using only one method is not safe. In order to minimize the number of false positive, the program would use 3 methods altogether. The program would return true only when the results from all 3 methods have exceeded the threshold. In order words, to get a false positive, all 3 methods need to return unreasonable values which is a rare case. By doing so, the accuracy of the algorithm is increased with the tradeoff of time complexity since the program need to compute 3 values instead of 1.
3.6.1 Telegram Channel

As mentioned in Section 2.7, a report channel is required in the lost & found system. A report channel is established between the patrol robot and Telegram by using Telethon API. The robot can send message through Telegram to raise message notification.

The Telegram channel is a private channel. Only the invited member can join the channel. This allows access control of the information provided by the robot. Only the robot can publish messages in the channel. The message can be deleted automatically after 7 days to reserve the memory space.

The report channel is working probably. The robot is able to connect to the Telegram account and publish images and messages in an appropriate manner. Figure 3.17 and Figure 3.18 shows the examples of the Telegram messages.

3.6.2 Log message

In the trial runs, the messages need to be logged in a log file before sending to Telegram. This is because the robot does not possess a valid SIM card. It cannot access to the internet since it need to connect with the base module. Therefore, during the trial run, the alert messages are temporarily saved in a text file (See Figure 3.23 below). However, in actual application, this problem can be neglected since the patrol robot can be assumed to have a valid SIM card.

![Message log example](image)

Figure 3.23 Message log example

3.7 Custom YOLO Model Training

The current lost & found system is using YOLO-v5s model. It is one of the pretrained model provided by YOLO. However, many classes in this pretrained model is actually irrelevant to this project. For example, the detection of chair, table, car etc. is unnecessary for the lost & found system. Hence, a custom model is trained which aims to maximize the detection accuracy and efficiency of the target object such as suitcase, backpack, handbag, cell phone, wallet etc.
Figure 3.24 shows an example of comparing the result of using yolov5s and the custom model. As shown in the figure, the default model (yolov5s) detects some irrelevant objects such as tv and clock. On the other hand, the custom model only focuses on the target objects (suitcase). By excluding the irrelevant objects, not only the detection accuracy and efficiency can be increased, it also makes the resultant image clearer for the staff to locate the target items.

One more different between yolov5s and the custom model is that yolov5s model does not include “wallet” (one of the 1st class objects) in its training dataset. Figure 3.25 shows another example of detecting a batch of wallets. Yolov5s cannot recognize the wallet correctly while the custom model is able to detects wallet shown in the image. Although it often misclassifies the wallet as a cell phone, it is a great improvement compared to the default model. In fact, the system can still make notification correctly even if it misrecognizes the wallet as a cell phone since cell phone is also one of the 1st class objects. There would just be a minor error in the alert message (“wallet” replaced by “cell phone”).
However, the accuracy of the custom model is sometimes lower than the default model. As shown in Figure 3.26, the default model is having 0.82 confidence on the suitcase, while the custom model has made a wrong prediction. This may due to the custom model is trained by a dataset with less data (See Appendix F for more detail of the training result of the custom model) while the default model is trained by a large dataset.

3.8 Other Work & Abandoned Design

Throughout the project, there are some other works being done but finally are not included in the current system. There are also some of the ideas which have been abandoned as the project progresses. This section would like to discuss some of them and explain the reason to exclude them from the project.

3.8.1 Video / Image Mode

As mentioned above, 1\textsuperscript{st} class and 2\textsuperscript{nd} class objects are detected in two different approaches (1\textsuperscript{st} class objects are detected throughout patrol, 2\textsuperscript{nd} class objects are detected at checkpoints). Therefore, in the initial planning, the robot would perform object detection on video stream during patrol and perform object detection on the images at the checkpoint.

However, since the speed of the robot is not high (max speed: 1 m/s), high FPS is not required for checking the environment. For example, setting FPS to 1 can already capture most of the candidate items during the patrol. Therefore, video mode is removed from the lost item detection system to reduce the computational time and increase the efficiency.

In the experiment, the robot can still recognize the lost item on the patrol path successfully using image mode (FPS = 1). This implies that the decision of abandoning video mode is reasonable.
3.8.2 Usage of Different AI Model

In the current system, YOLO is the only model being used for object detection. However, in the initial planning, different models would be used in different section. The main purpose of doing so is to utilize the advantages of different models. In the original hypothesis, YOLO & SSD model would run faster but with a lower accuracy, while Faster R-CNN would be the opposite. Hence, the model with high accuracy would be used to detect objects at checkpoint since more time is given while the model with faster speed would be used to detect objects during patrol to achieve real time detection. This approach can make good use of both high accuracy and fast processing time at the same time.

This design is eventually abandoned due to two reasons. The first reason is mentioned in Section 3.7.1, that is, a low FPS can still achieve a satisfactory result. Hence, reduce FPS can solve the concern of processing time. The second reason is due to the performance of YOLO is better than the original expectation. Therefore, YOLO become the only model left in the system.

3.8.3 Usage of GPU & TPU

The usage of GPU (Graphics Processing Unit) and TPU Tensor Processing Unit have been considered in the middle stage of the progress. Both GPU and TPU are the hardware which can facilitate the processing time of performing object detection. However, this idea is not adopted at the end due to the power consumption concern and the approach of reducing FPS. After reducing the FPS and image resolution, simply using CPU is already capable of performing the task and therefore GPU & TPU are no longer needed.

4. Future Plans and Conclusion

4.1 Overview

Due to time limitation, the system is not yet fully developed. There are still some improvements and additional function which can be included. This is the final chapter to present the future plan such as web interface, improvement about item detection and training of YOLO network. Finally, a conclusion will be given to summarize the whole project.

4.2 Web Interface

Currently, the system stops tracking of the lost item after it has been reported. In the future, a web interface would be implemented to manage the reported items. The web interface would be connected to the database which stores the information of the reported items. Airport staffs can
provide feedback through the web interface. For example, has the item been picked up? Is the
detection valid? Is there any remake on the item? The information can be used to further improve
the detection system. In addition, the web interface can provide a more comprehensive view of
the lost item collections which can facilitate the management of the reported items.

4.3 Improvement of 1\textsuperscript{st} Class Detection

![Figure 4.1 Example of a detection result](image)

Figure 4.1 shows an example of a detection result including a cell phone. In the above case, it is
obviously that the cell phone should not be classified as a lost item. A simple solution of solving
this problem is to divide the image into upper part and bottom part (as shown in Figure 4.1). Since
the objective is to detect 1\textsuperscript{st} class object appeared on the ground, pass only the bottom part
of the image into YOLO network can avoid false alert. Other than that, the system can be further
improved by analyzing the status of the 1\textsuperscript{st} class object such as whether or not the item is held by
a person. This may require the technique of action understanding in video/picture which can be
included in the future to further increase the detection accuracy.

4.4 Further Training on Custom YOLO Model

As mentioned in Section 3.7, there are still a lot for the custom model to improve. Therefore,
further training on the custom YOLO model would be one of the tasks in the future plan.
Including prepare the corresponding training dataset and train the custom model with different
hyperparameter to find the best model.
4.5 Conclusion

To conclude, this project managed to build a lost and found system for HKIA with the use of smart patrol robot. The goal of this project is to save manpower and increase the efficiency of finding lost item in the airport. The overall result of the project is satisfactory. This project has shown the potential ability of using computer vision and robotic technique to detect lost item in the environment. With the improvement mentioned in the future plan, it is believed that the lost & find system is a workable idea which can be applied to create a smart airport.
References


Appendices

Appendix A

Robot density of different countries

Robot density in the manufacturing industry 2019

Source: International Federation of Robotics
Appendix B

External hardware of the patrol robot
Appendix C

Accuracy and FPS Comparison

![Accuracy and FPS Comparison Chart]

---

Frames per second

---

Accuracy comparison
二、通信格式

通信的具体格式为4字节的16进制数据，通常发送给机器人的命令为28位字节数据，机器人返回30位字节。

例：55 77 CMD NUM Data1 Data2 Data3 Data4 ……

数据解析：

<table>
<thead>
<tr>
<th>数据位</th>
<th>详情</th>
</tr>
</thead>
<tbody>
<tr>
<td>55 77</td>
<td>固定位</td>
</tr>
<tr>
<td>CMD</td>
<td>01：控制；02：信息查询；03：数据返回</td>
</tr>
<tr>
<td>NUM</td>
<td>CMD数据位的子命令数据位</td>
</tr>
<tr>
<td>Data1～Data24</td>
<td>具体数据内容</td>
</tr>
</tbody>
</table>

三、通信内容

根据CMD值的不同，可以把串口信息分为三大类：机器人控制、信息查询、数据返回。

1. 机器人控制（CMD=0x01）

根据NUM值的不同，可以再次把指令细分下去。
2 信息查询（CMD=0x02）

2.1 获取位置信息（NUM=0x01）

发送指令

55 77 02 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

返回信息：

55 77 03 01 56 B0 80 3E E3 09 F6 3B 36 B3 8E BA FE FF 7F 3F 00 00 00 00 00 00 00 00

解析：当前的数据转换坐标为：x=0.2513453；y=0；a=0；w=0.99999

坐标转换表

<table>
<thead>
<tr>
<th>字节数</th>
<th>16 进制</th>
<th>10 进制</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>56 B0 80 3E</td>
<td>0.2513453</td>
</tr>
<tr>
<td>4</td>
<td>E3 09 F6 3B</td>
<td>0.00000000000163 约等于 0</td>
</tr>
<tr>
<td>4</td>
<td>36 B3 8E BA</td>
<td>-0.001088715 约等于 0</td>
</tr>
<tr>
<td>4</td>
<td>FE FF 7F 3F</td>
<td>0.99999</td>
</tr>
</tbody>
</table>

3.信息返回（CMD=0x03）

3.1 返回位置信息（NUM=0x01）

55 77 03 01 56 B0 80 3E E3 09 F6 3B 36 B3 8E BA FE FF 7F 3F 00 00 00 00 00 00 00 00

说明 | 字节数 | 16 进制  | 10 进制  |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>X 坐标值</td>
<td>4</td>
<td>56 B0 80 3E</td>
<td>0.2513453</td>
</tr>
<tr>
<td>Y 坐标值</td>
<td>4</td>
<td>E3 09 F6 3B</td>
<td>0.00000000000163 约等于 0</td>
</tr>
<tr>
<td>A 值</td>
<td>4</td>
<td>36 B3 8E BA</td>
<td>-0.001088715 约等于 0</td>
</tr>
<tr>
<td>W 值</td>
<td>4</td>
<td>FE FF 7F 3F</td>
<td>0.99999</td>
</tr>
</tbody>
</table>
3.3 返回机器人导航状态 (NUM=0x03)

55 77 03 03 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00

解析：机器人成功到达导航目标

数据解析表

<table>
<thead>
<tr>
<th>数据位</th>
<th>数值</th>
<th>说明</th>
</tr>
</thead>
<tbody>
<tr>
<td>55 77</td>
<td>55 77</td>
<td>固定位</td>
</tr>
<tr>
<td>CMD</td>
<td>03</td>
<td>机器人返回信息</td>
</tr>
<tr>
<td>NUM</td>
<td>03</td>
<td>导航运动状态</td>
</tr>
<tr>
<td>Data1</td>
<td>0/1</td>
<td>发送目标点给机器，机器开始执行任务</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>发送目标点给机器，机器成功到达后返回</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>发送目标点给机器，机器判定无法达到立即返回</td>
</tr>
<tr>
<td></td>
<td>2, 5-8</td>
<td>发送目标点给机器，机器接收到取消或判定无法到达后发出任务取消</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>发送目标点给机器，机器判定目标丢失后发出</td>
</tr>
</tbody>
</table>

Data1 (0----F) 表示不同的状态，如：data1=3 表示成功到达目标位置

备注：

X 与 Y 的单位为米 (m);

角度值需要通过 A 和 W 之间的关系式进行计算：

A 和 W 关系式：

\[ Th = \text{atan2}(2^*A^*W;W^*W-A^*A); \]

\[ A=\sin(Th/2); W=\cos(Th/2); \]

\(-\pi < Th <= \pi, Th \) 单位是弧度；

弧度转角度公式：180/π*弧度；

角度转弧度公式：π/180*角度。
Appendix E

Location and View of each [Checkpoint-Direction]

A-0 location & camera view
A-1 location & camera view
B-0 location & camera view
B-1 location & camera view
Appendix F

Result of custom YOLO model training