University of Glasgow

电子科技大学格拉斯哥学院
Glasgow College, UESTC

UESTC 3010
Team Design Project and Skills
Final Report

#34 Team Friday

CUI Sijie	2510762	2019190505010
FENG Zihao	2510986	2019190605011
HE Renjie	2510987	2019190605012
LI Sipei	2510754	2019190505002
WU Haoyang	2510778	2019190505027
XU Ben	2510780	2019190505029
YAN Zecheng	2510783	2019190505032
YANG Pei	2510784	2019190505033
YUE Haiyi	2510758	2019190505006
ZHENG Xiaochen	2511012	2019190605037

[in order of Chinese Pinyin]

Abstract

This team design project presents an intelligent rover design capable of automatically completing six tasks on East Lake patios, UESTC, within the budget of 1,000 RMB.

Our designed rover is equipped with a mbed Nucleo-L432KC as the central controller, an OpenMV camera and four ultrasonic detectors as its sensors, and two robotic arms for completing the tasks. This rover features its customised PCB design, practical deck and mast design, a Bluetooth receiver with GUI, and AI-based gravel recognition algorithms. With the help of a highly-sensitive closed-loop feedback system and finetuned motors, our rover travelled smoothly on patio one and completed all tasks within 200 seconds. Another ultrasonic-based robust feedback system, together with a classic yet advanced shape detection algorithm, ensured the rover finished all tasks on patio two within 240 seconds. The total budget of this project was 998.84 RMB.

For the convenience of individual assessment, Table 7 summarises each team member's contribution. Also, note that in this report, individual contributions have been noted after the heading of the lowest possible level.

Acknowledgement

The authors wish to thank all those who have helped us with **Team Design Project and Skills**.

First, thanks to **Prof. Wasim Ahmed, Prof. Abdullah Al-Khalidi, Prof. Oluwakayode Onireti** and all the other supervisors who had set up this course. The elaborately designed tasks and meticulous instructions made the project significant and inspiring to conduct. Our problem-solving skills, cooperation techniques, and professional knowledge were significantly enhanced by designing a multifunctional rover.

Sincere gratitude should also go to all our learned and warm-hearted **teaching assistants** who have greatly helped us, ranging from technical guidance to task demand clarification.

Finally, our cordial thanks also go to all of our **team members**. Everyone was diligent and assiduous in completing their missions. We are all proud of the achievements and joint efforts we made in the past three months.

Content

List of Tables

List of Figures

1

Task Analysis

1.1 Tasks in Patio 1

For patio 1, the rover will first track a lane which has a different texture and colour from its surrounding ground. After that, the rover should turn right and pass through a bridge, which has a ramp of 14 degrees. The car will turn right and start tracking again once it leaves the bridge. The final task is to pass through a given gate which has a height of 0.5 m and a width of 0.5 m.

Given the tasks, it is quite clear that the core function the rover should complete in this patio is to track the specific lane. The second significant function of the patio is to find the place to make a turn.

1.2 Tasks in Patio 2

For patio 2, the first task the rover should complete is to find a 10 cm x 10 cm card and detect the shape of the pattern on the card. The next step is decided by the detected shape, which can be a triangle, rectangular or circular. For each case, the rover will run forward to a specific vertex of the diamond and pull down a sign. Once the task is completed, the rover should automatically go to a specific place to release a ball to a basket. After that, the rover will go to the appointed place to send a message containing the required information to the computer using Bluetooth with a frequency 433Mhz. Finally, the car will go to the terminal and complete its work on patio 2.

The key actions that should be completed for patio 2 include detecting the shape of the given card, releasing the carried ball to the given basket and sending a message using Bluetooth. Another important point to be considered is how to guide the rover to the specific place for releasing the ball and sending the message. To guide the rover to move between places of different tasks, the route is designed and shown in Figure 1.

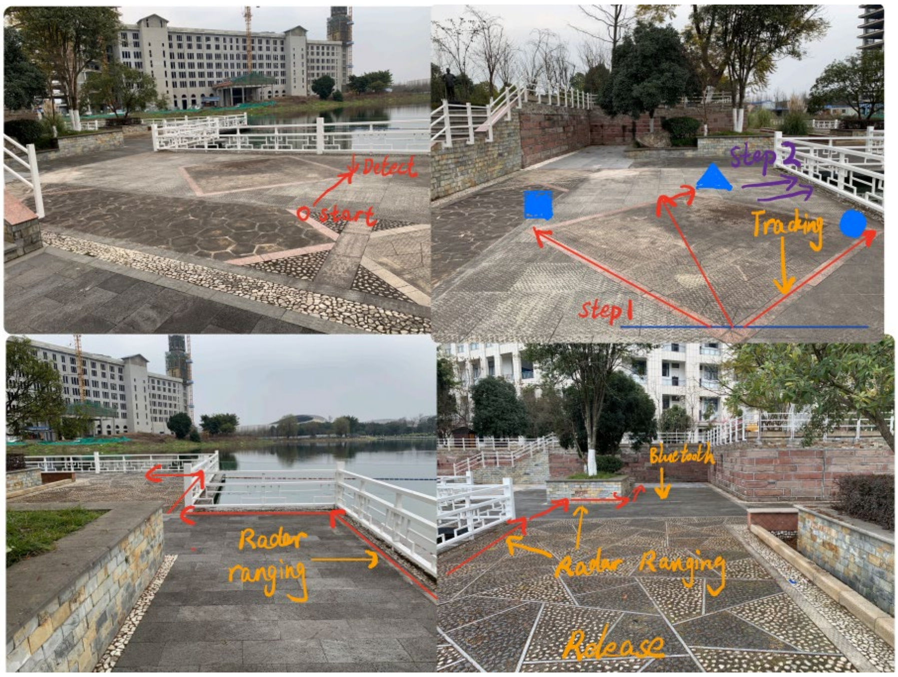


Figure 1. Expected moving route for patio 2.

1.3 Report Structure

The rest of this report is organised as follows: We will begin by introducing our rover's system-level design in Section 2. Then in Section 3, we will cover the technical content of our submodules that appeared in our final design. Section 4 will introduce those designs and submodules developed for redundancy that did not eventually appear in our final design. System integration will be introduced in Section 5, followed by field test results and discussion in the same Section. Finally, Section 6 concludes this Team Design Project. External links of codes and project files, together with our financial record, will be provided in appendices.

Blank Page

2

Top System Design

2.1 System Design

The top-level system design is shown in Figure 2. It contains three main subsystems, the communication system, intelligent control system and ball releasing system. The communication system is designed to send a message for Task 3 of Patio 2 while the ball releasing system is designed to release the tennis ball into the basket for Task 2 of Patio 1. The intelligent control system has the ability to decide the rover's motion using its distance from the surroundings, visual information and speed information. The main sensors are a camera, 4 ultrasonic radars and encoders on the motors. The Speed control is a PID control and the others apply P control algorithm.

To achieve the function required, a detailed design of the hardware is shown in Figure 3. In our development of software, the structure of the software is specifically designed so that functions can be developed individually and will not affect each other. The detailed structure of the software is shown in Figure 4.

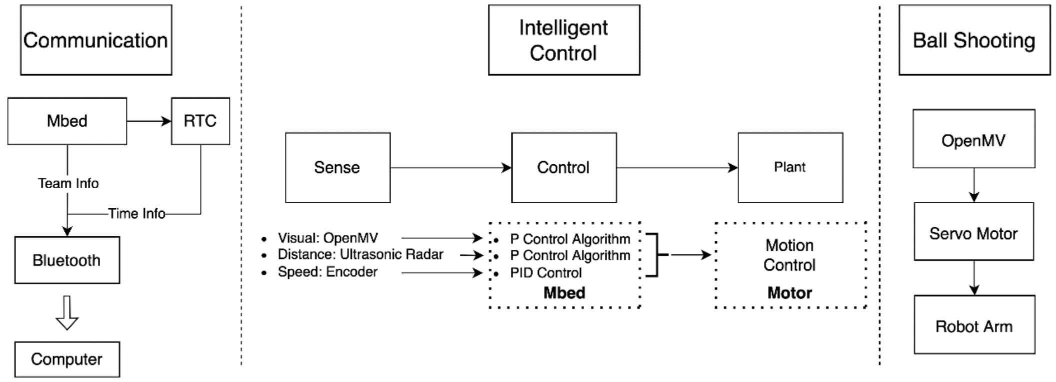


Figure 2. Top-level system structure for the rover.

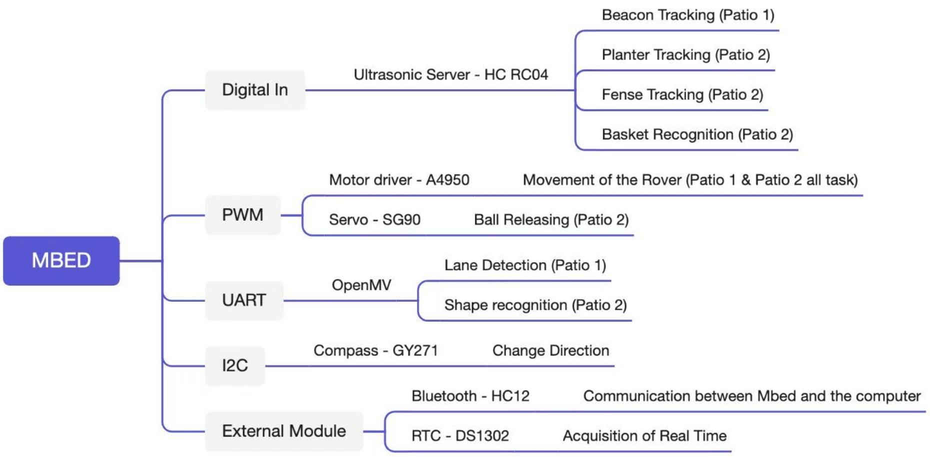


Figure 3. Structure of the hardware.

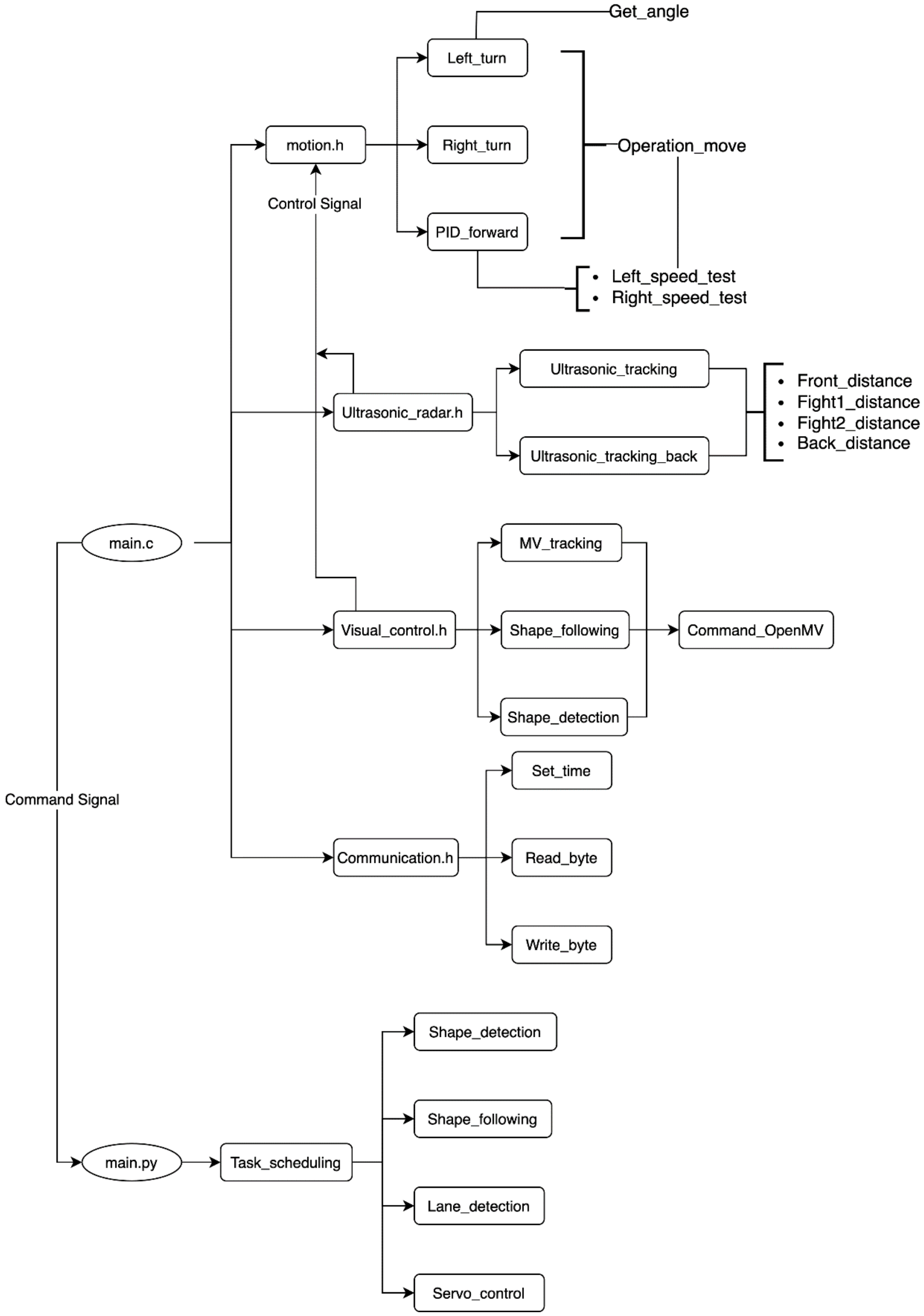


Figure 4. Structure of the software.

2.2 Task Breakdown

We divide the working system into 6 technical systems, including the ultrasonic radar system, driver system, computer vision system, Bluetooth communication system, ball releasing system and coordination between MBED and OpenMV. The detailed functions are introduced.

2.2.1 Ultrasonic Radar System

The radar range should be realized. The system should be able to accept the Radar ranging distance from 3 directions: the front side, the left side and the right side. Specifically, the following functions should be realized:

Table 1. The function definition for ultrasonic radar ranging.

Function Name	Input	Output	Notes
Radar ranging	(int) Direction	(float) Distance	The main function can obtain distance information.
Radar tracking	(int) Input 1: Direction (float) Input 2, 3: Tracking range	(int) Quantized deviation	Input 2 is the minimum distance and input 3 is the maximum distance.

2.2.2 Driver System

The driver system should facilitate the moving motion of the rover. The rover will be able to move forward, stop and make a turn-based on the driver system. The functions should be organized as the follows:

Table 2. The function definition for driver system.

Function Name	Input	Output	Notes
Forward			The rover moves forward
Stop			Rover stops running
Turn 45°	(int) left or right		If a 90° or 135° turning is required, the function will be called for many times
Adjust	(int)		The input shows the magnitude of the adjustment

The functions have no output and they are only required to accept information and decide how the motors work.

2.2.3 Computer Vision System

The main functions that need to be completed include:

Table 3. The function definition for computer vision system.

Function Name	Input	Output	Notes
Tracking		(int)	The output indicates the deviation of the rover in tracking
Shape detection		(int)	0 for None, 1 for circular, 2 for triangle and 3 for rectangular

Shape follows		(int)	output= 0, left turn
			output= 1, forward
			output= 2, right turn

Tracking		(int)	The output indicates the deviation of the rover in tracking

2.2.4 Bluetooth Communication System

A Bluetooth HC-12 will be applied to the system to support communication between the rover and the computer. A UART port will be used to connect the OpenMV and the Bluetooth chip. The system is designed to send a message to the computer when it is called by the main function.

Table 4. The function definition for Bluetooth communication.

Function Name	Input	Output	Notes
Send message		1/0	If the message is accepted by the computer successfully, a 1 is returned; else, the function should try it again until successful.

2.2.5 Tennis Releasing System

A mechanical device needs to be designed to release the ball to the basket. One function should be written in this module.

Table 5. The function definition for ball releasing.

Function Name	Input	Output	Notes
Release			When called, should release the ball

2.2.6 Coordination between MCU and OpenMV

The system will support a connection using UART between OpenMV and MBED. The MBED and

OpenMV should follow the master-slave relationship. The MBED should be able to control OpenMV to act the required functions. For example, when the main function in MBED decided that the next step is to do tracking. The Tracking function in the OpenMV should be called. Once the task is completed, a command should be sent to OpenMV to close the function Tracking. The OpenMV should also be able to send messages to MBED.

Table 6. The function definition for master-slave control between MBED and OpenMV.

Function Name	Input	Output	Notes
Order	Order from MBED	Results of Operation from OpenMV	Once the OpenMV received an Order from MBED, it will start working and return the corresponding results to MBED.

In this section, we present the submodules that appeared in our final design.

3

Submodule Design

3.0 Individual Contributions

For the convenience of individual assessment, we summarised our contributions individually in Table 7.

Table 7. Individual contributions summarised by each student. This table is summarised here for the convenience of individual assessment. Names are listed in order of Chinese Pinyin.

Name and UoG Matric	Contributions
CUI Sijie 2510762	- Designed and constructed the Adaptive Dynamic System (ADC) - Designed, constructed and tested the fence tracking system based on P-P Tracking System (P2TS) - Designed and constructed Basket Docking System (BDS) - Constructed the powertrain and physical structure of the rover - Coordinated dynamic control and sensing part and liaised with visual and communication part
FENG Zihao 2510986	- Designed a sweeping system for removing obstacles - Implemented the stable communication between OpenMV and STM32L432KC - Did field tests and adjusted the parameters for the dynamic system - Recorded and managed the manufacture and operation expenses
HE Renjie 2510987	- Designed a communication scheme between the rover and the PC - Developed an interface between HC-12 and MBED to complete the wireless transmission task - Developed a serial debugger on the PC side to meet specified requirements - Assisted in overall system debugging and field testing
LI Sipei 2510754	- Achieved short and long distance recognition using AprilTag - Designed a tailored beacon - Processed dataset for line tracking - Tested the feasibility of visual auxiliary system for shape detection - Assisted group meetings - Responsible for Gantt Chart - Report synthesis
WU Haoyang 2510778	- Designed a shape detection framework - Proposed a shape localisation and recognition algorithm based on ROA and CFF - Advised on the structure an implementation of code
XU Ben 2510780	- Debugged the overall system - Developed the main function - Optimised the feedback systems - Designed the master-slave control of MBED and OpenMV

Name and UoG Matric	Contributions
YAN Zecheng 2510783	- Constructed the rover - Adjusted PID parameters - Conducted the preliminary soldering work of the PCB version - Tested the accuracy and precision of ultrasonic module
YANG Pei 2510784	- Designed a feedback control system for Patio 1 gravel tracing - Developed a novel AI-based gravel recognition algorithm and implemented on OpenMV - Proposed an ultrasonic angular proportional controlling algorithm - Performed magnetometer calibration based on least square fitting and matrix factorisation - Organised group regular meetings - Organised and synthesised group assessed materials - Developed visual identity guidelines for assessed materials
YUE Haiyi 2510758	- Designed the ball-releasing device - Developed a CV-based algorithm to detect the basket - Developed a CV-based algorithm for the transition from Task 1 to Task 2 in Patio 2 - Engaged in the dataset processing for Patio 1
ZHENG Xiaochen 2511012	- Designed the motherboard PCB - Designed the rover’s structure - Contacted with the manufacturers - Helped with assembly and testing

3.1 Rover Structure

The structure includes the skeleton of the rover, what it was like, how it was designed, and how it was built.

3.1.1 Function Division (ZHENG Xiaochen)

First, we identified what components were given and what needs to be designed.

Things we bought directly from the suppliers:

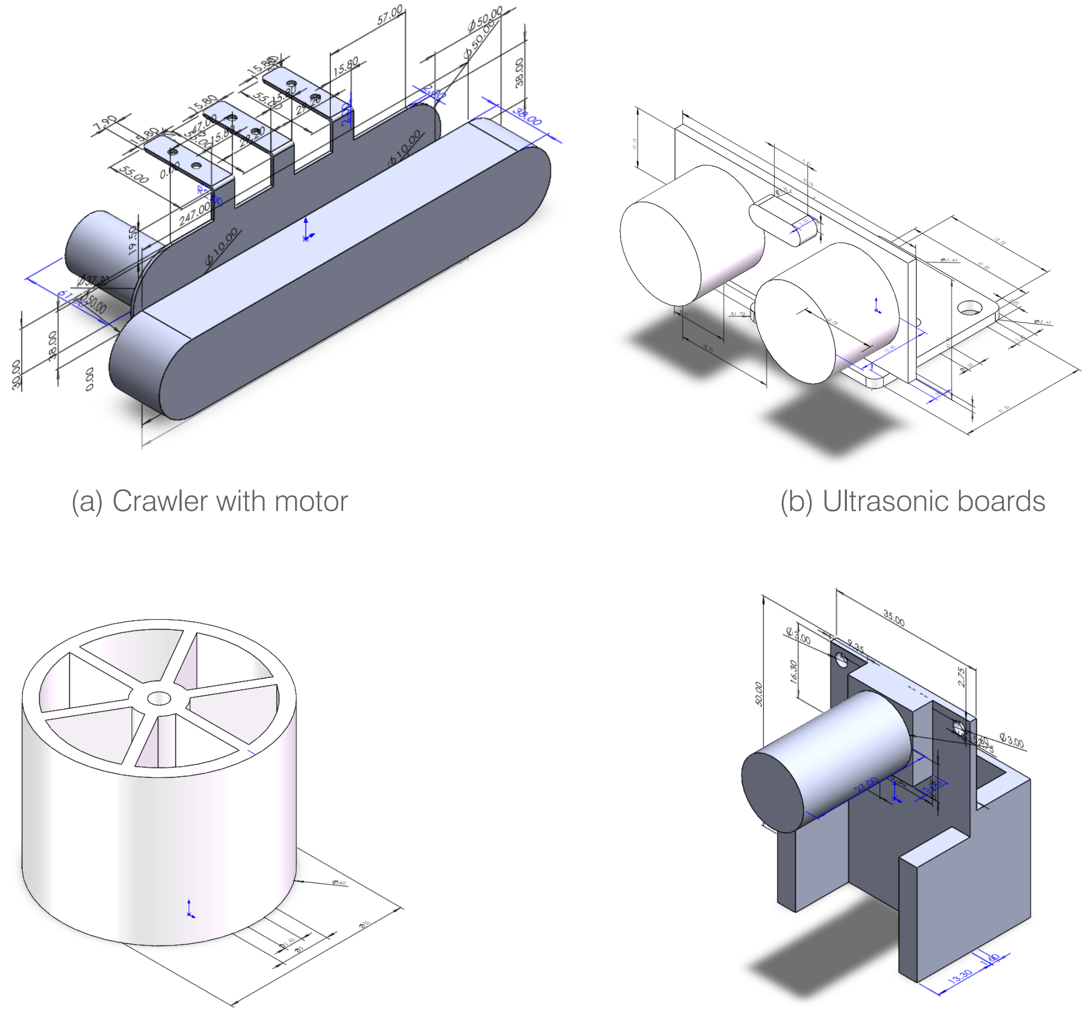
- Crawler with motor
- Poles
- Ultrasonic boards
- Guide wheel
- OpenMV board
- SG90 servo

Things we needed to design ourselves:

- Chassis (A drilled board)
- PCB mounting board
- OpenMV mounting bracket
- Battery pack
- Ball releasing apparatus

A drilled solid board is mounted with the crawler to form the base of the rover. The brain of the rover, the motherboard PCB together with other functional components are mounted on this base to equip the rover. The position of the components needs to be carefully designed, tested and modified to ensure they work properly.

Computer-aided design (CAD) tools were utilized to optimize our design process and avoid manufacturing failures. The software chosen was Solidworks 2021 for its powerful features.
The dimensions of the already existing parts were well measured and modelled as a reference for custom designs. Their respective 3-D models built are shown in Figure 5.

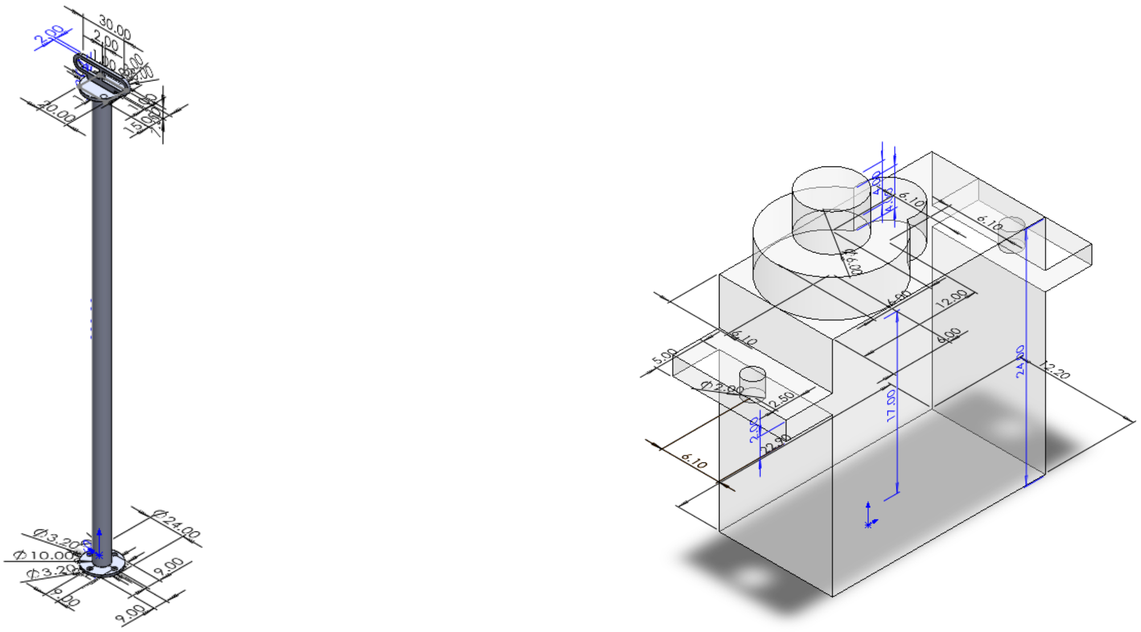


(a) Crawler with motor

(b) Ultrasonic boards

(c) Guide wheel

(d) OpenMV board



(e) Poles

(f) SG90 servo

Figure 5. 3-D models.

With the model of the given components, we could design our structural components. The key to the design was to leave the correct size and position of the screw holes so that these structures could be fitted to our custom skeleton.

3.1.1.1 Chassis Board

First, the solid board as the chassis is designed. Some specifications were collected from prototyping.

- Shark's head-shaped front end, with guide wheels on both sides.
- Curved front, just the right shape to fit the bin.
- Motherboard at the middle.
- A battery case behind the motherboard.
- A pole to sustain the compass board, at the left rear of the rover.
- The two crawlers were placed as close as possible.
- Not too large, with each edge just over the crawler.
- The OpenMV is hung looking ahead at the front with two positions available: 1. 30cm above the ground, pitch angle 60 degrees; 2. 14cm above the ground, pitch angle 7.5 degrees.
- Ball releasing apparatus at the front.
- Four ultrasonic boards are placed directly on the chassis board. One at the front, one at the end, and two facing right. No obstruction in front of the ultrasound.
- The distance between the right two ultrasound modules is a definite number between 15 and 20 cm.
- The mounting holes for the crawlers are M5, for the rest of the components are M3.

Considering these specifications, the chassis board was designed as follows.

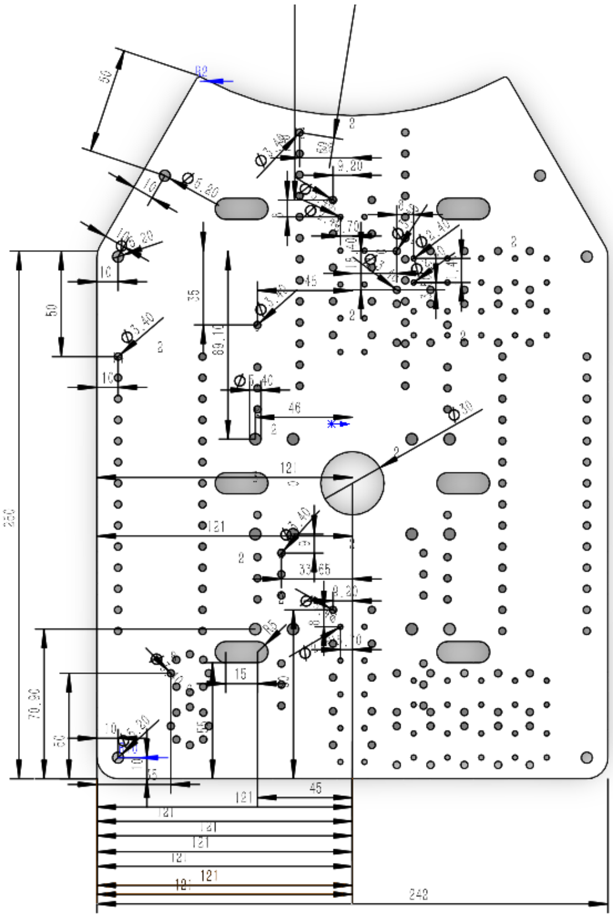


Figure 6. The final design of the chassis board.

Extra holes are prepared on the board in the matrix in case of modifications. An adequate number of evenly distributed holes provide sufficient mounting pivots for the components while ensuring the mechanical properties of the board.

Assemble the crawlers and the chassis board, and we can get a rough preview of the rover's chassis in Solidworks. The distance between the chassis board and the ground is approximately 86.5mm.

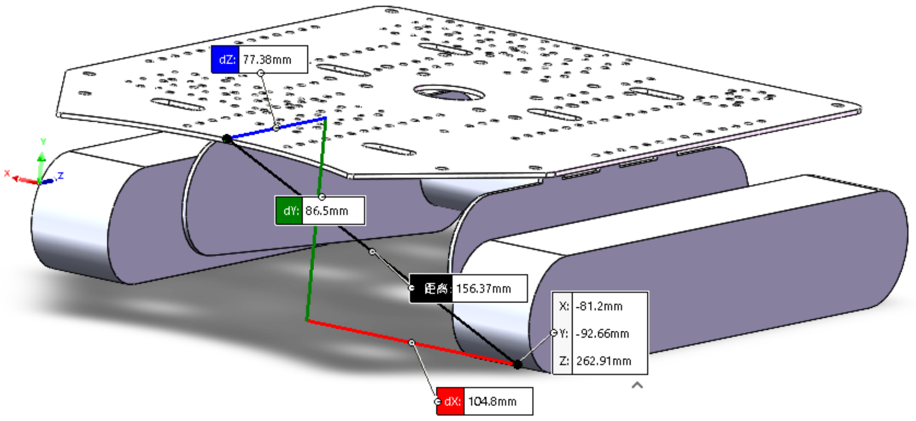


Figure 7. Rover chassis.

3.1.1.2 PCB Mounting Board

Although the PCB itself is a platform, we need some support plates to clamp and hold the PCB like a hamburger to put some accessories on it. The specifications for this board are listed below.

Mounting holes for PCB, 90mm x 90mm square

Two mounting holes for the voltmeter, spacing 33mm

Mounting holes and air holes for the fan. The fan should be placed above the chip A4950.

Other peripheral boards glued to the board (if necessary)

Considering these specifications, the PCB mounting board was designed as shown below.

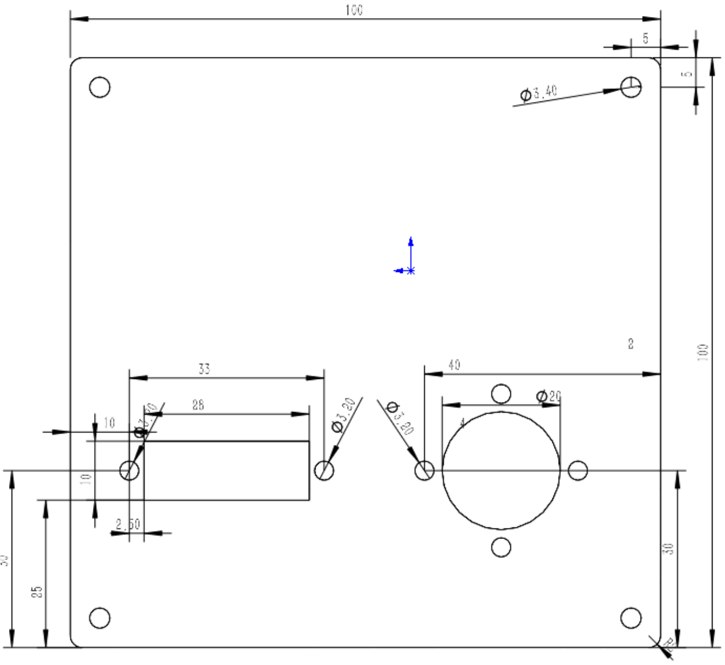


Figure 8. The final design of the PCB mounting board.

3.1.1.3 OpenMV Mounting Bracket

In patio 1, the OpenMV is required to be installed 14cm above the ground. Thus, we need a frame to provide special-orientation mounting holes for the OpenMV board and to raise the camera to approximately 14cm. The design of this bracket is based on the idea of using separated acrylic pieces in a mortise-and-tenon construction and reinforced with glue to create an arch-shaped three-dimensional bracket. The design is shown below.

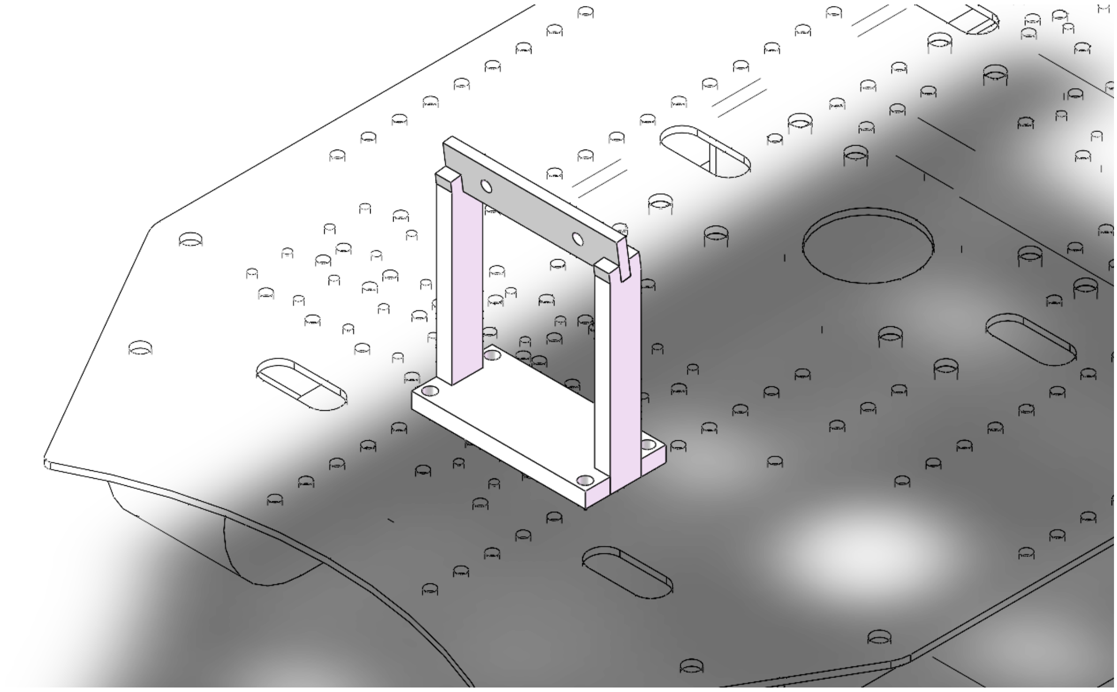


Figure 9. The final design of the OpenMV bracket.

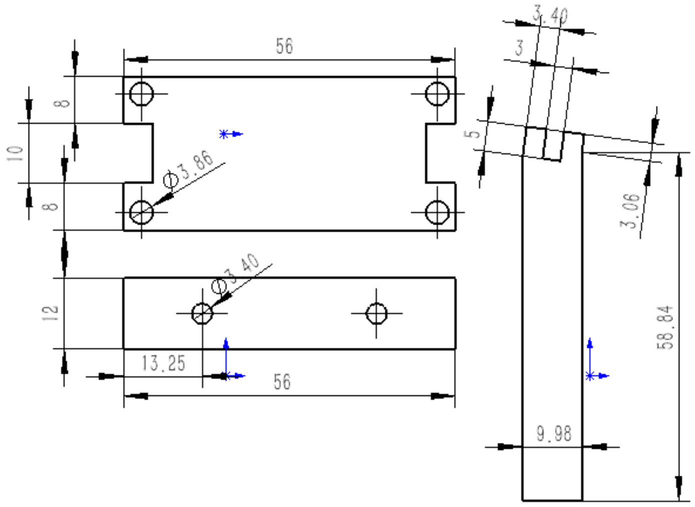


Figure 10. Section of the parts.

3.1.1.4 Battery Pack

As the battery is a smooth cube and does not provide a mounting method on its own to attach itself to the car, we needed to design a battery pack that could be fixed to the car to place the battery inside. The design of the battery pack is based on the same idea as the OpenMV bracket, with several specially designed acrylic panels that are joint together in a mortise and tenon configuration and glued in place to form a single box structure. The design is shown below.

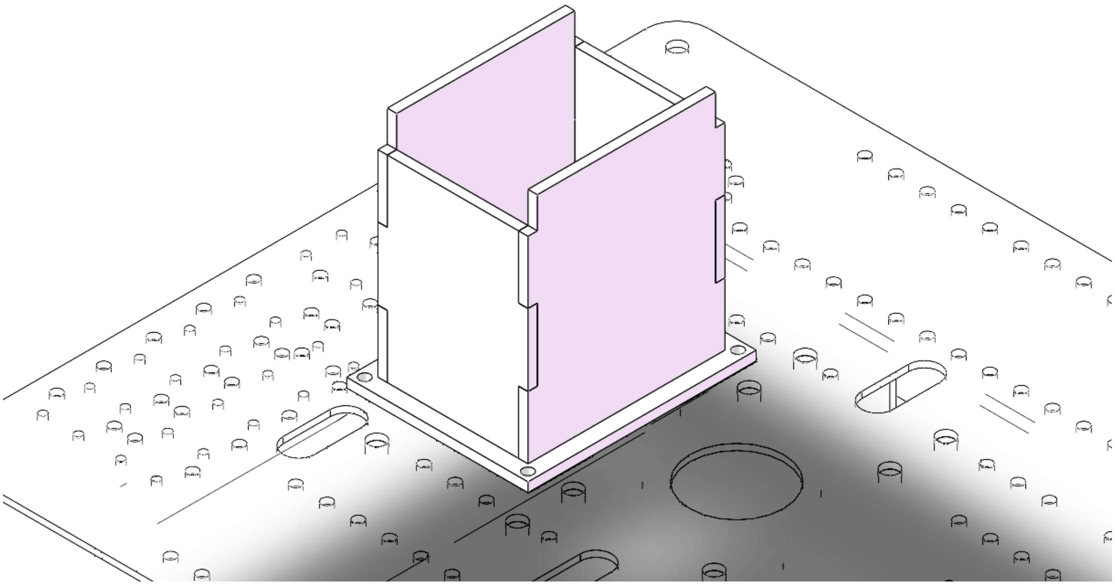


Figure 11. The final design of the battery stand.

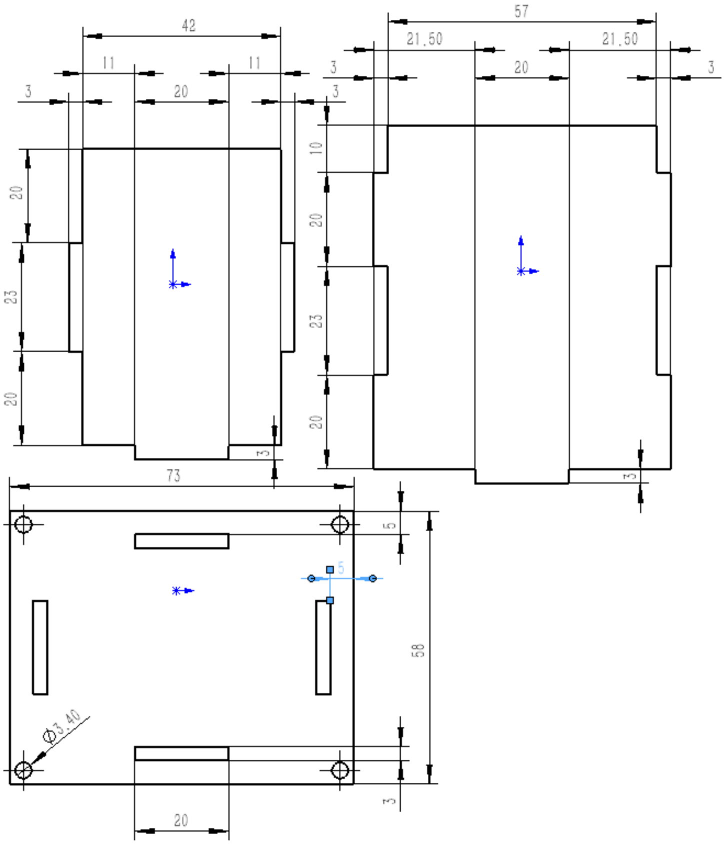


Figure 12. Sections of the parts.

3.1.1.5 Ball Releasing Apparatus

The content will be discussed in Section 3.6.

3.1.2 Fabrication

3.1.2.1 Chassis Board (ZHENG Xiaochen)

The material for the chassis board was carbon fibre. For material selection, the rover's weight was a significant consideration because if the rover is heavy, the two motors cannot drive the rover. By comparing the densities and mechanical properties of some common plates, we chose carbon fibre for its lightweight and high strength.

Table 8. Comparison of materials to be selected.

Material	Density (g/cm-3)	Tensile strength (MPa)	Additional attributes
Acrylic	1.2	~50	Cheap, Wear-resistant, Easy processing
Carbon fibre	~1.5	3400	Corrosion-resistant, Fatigue resistant
Aluminium alloy	~2.63	480	Corrosion-resistant
Stainless steel	7.93	515	Corrosion-resistant, High hardness
Brass	~8.50	330	Cheap, High hardness

Our carbon fibre board was ordered online.

3.1.2.2 Other Boards (ZHENG Xiaochen)

The material is chosen for the PCB mounting board, OpenMV bracket, battery pack, and other self-made structures were acrylic boards featuring low price and satisfactory strength. They were easy-accessible materials that are a wonderful alternative to 3D-printed plastics. With simple laser cutting processes and assembly, they can be made into a variety of 3-D structures.

All our acrylic boards were ordered online.

3.1.2.3 Assembly (ZHENG Xiaochen, CUI Sijie, YAN Zecheng, XU Ben)

Assembly was a significant step in making the rover. We applied screws and nuts, copper cylinders, glue, adhesives, and cord for the jointing and fastening of discrete pieces. The mounting positions of the components have been carefully designed, while special assembly methodologies adapted to actual situations, with the necessary reinforcement to ensure that the rover has sufficient structural strength.

The materials used for assembly are listed below.

- M2 screws and nuts
- M3 screws and nuts
- M5 screws and nuts
- M2 brass pillars

- M3 brass cylinders
- Cyanoacrylate glue
- Hot melt adhesive
- Cord
- Solder

3.2 Moving System

The moving system includes the powertrain and the Adaptive dynamic system (ADS).

3.2.1 Power Train (CUI Sijie & YAN Zecheng)

The powertrain includes the motor controller, motor driver, and motor, which will power the rover to move forward, steer and climb ramps.

(1) Powertrain hardware configuration

Two MG513P60_12V motors are used to drive the rover, and these two motors are controlled by the A4950 driver chip. The output power of the driver chip is determined by the duty cycle of the PWM signal input to the driver chip. MBED (L432KC) is used to output PWM signals to control the driver chip, that is, to control the motor speed.

(2) Powertrain performance

The motor speed under different PWM signals is shown in Figure 13.

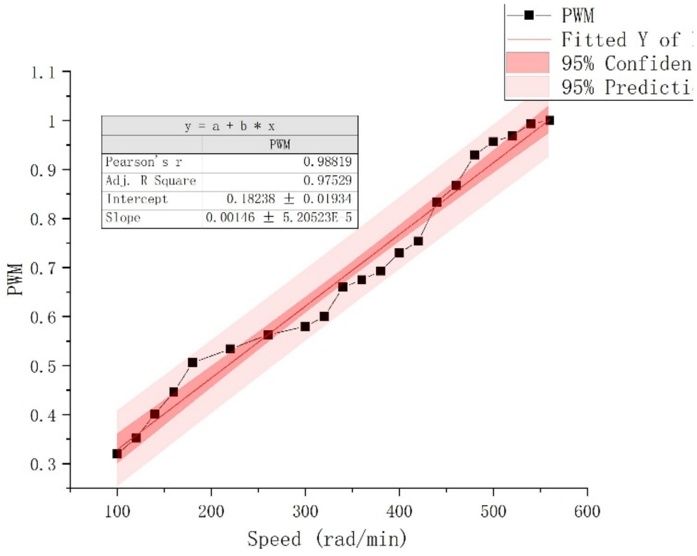


Figure 13. Relationship between speed and PWM. Note that this speed curve is obtained when the battery is fully charged.

Given that the speed curve is obtained when the battery is fully charged, and in field tests, we found that the speed curve changes as the battery power decreases, so controlling the speed by the PWM duty cycle alone cannot achieve high system stability. Therefore, the Adaptive Dynamic System will be introduced to control the speed which will be discussed in 3.2.2. Another performance of the rover's powertrain is given in Table 9.

Table 9. Performance of the rover's powertrain. (Note: (1) The minimum turning radius is the radius measured when one of the tracks is stationary and the other is advancing at maximum speed. (2) The maximum adjustment frequency is measured by the shortest adjustment time during which the rover can be seen to swing significantly when the speed difference between the two tracks is set to ±50%. (3) Test the maximum speed of the rover when moving forward on a concrete road. (4) The maximum climb angle is measured on a ramp with a dry wood surface.)

Test items	Performance
Minimal Turning radius	0.08cm
Maximal adjusting frequency	10Hz
Maximal speed	1.2m/s
Maximal climbing angle	35°

Table 9 demonstrate the feasibility of using this power train to finish the 6 tasks in the following aspect:

- The maximal speed allows the rover to finish both tasks in 5 minutes.
- The maximal climbing angle allows the rover to climb the bridge in patio 1 task 2 whose angle is about 14°.
- The minimum turning radius allows the rover to successfully handle the sharp turn of patio 1 task 1 whose radius of curvature is about 0.5m.
- The maximal adjusting frequency allows the rover to adjust its speed for both tracks frequently.

3.2.2 Adaptive Dynamic System

3.2.2.1 Reference PWM Determination (CUI Sijie)

During testing, battery power was found to affect the rover's ability to move in a straight line and the accuracy of its steering. Therefore, an adaptive method is used to modify the reference PWM. The method is to obtain a group of optimized PWM signals by ensuring that the speed of the left track is equal to the speed of the right track, and this group of PWM will be used as a set of initial values for PID control. In addition, the steering function will also use this set of values to ensure that both tracks rotate at the same speed so that the centre of rotation does not shift during steering.

The principle of this method is to fix the PWM signal output to the left track's motor at 0.5, and then change the PWM signal output to the right track's motor until the speed difference between the left and right crawler is less than 2%, and this state needs to be maintained for 5 seconds to satisfy the convergence condition.

The working principle of this adaptive reference PWM determination is shown in Figure 14.

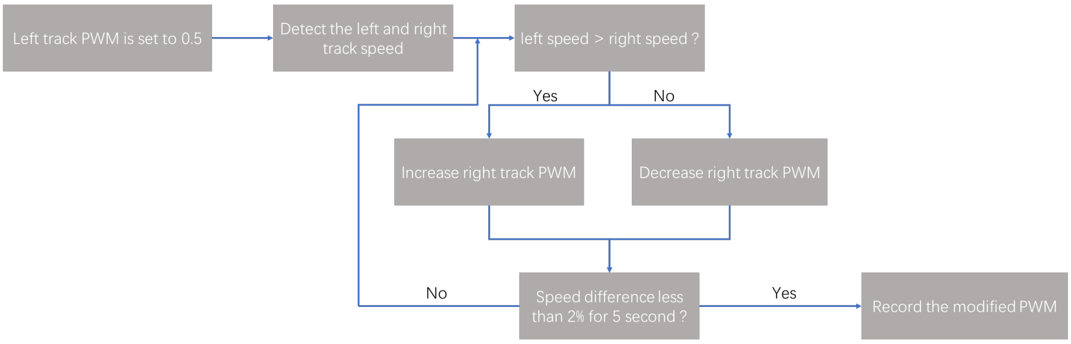

Left track PWM is set to 0.5
Detect the left and right track speed
left speed > right speed ?
Yes
No
Increase right track PWM
Decrease right track PWM
No
Speed difference less than 2% for 5 second ?
Yes
Record the modified PWM

Figure 14. Reference PWM determination working mechanism.

Every time before the rover test, the rover will independently adjust the reference PWM value to achieve the best speed matching state under the current power and the result of this will be included in the result in PID control part and steering part.

3.2.2.2 PID Speed Dynamic System (CUI Sijie & YAN Zecheng)

(1) Background and Objectives (CUI Sijie)

Given that the PWM duty cycle-speed curve will change with time, it is unstable to control the speed of the track by PWM signal. Besides, when applying 8 Volts (0.75 for PWM Duty Cycle) to both motors, it is found that the rotating speed is different for both motors, and the speed difference could be up to 30%. This is because the friction force varied for each gear, shaft or junction. Therefore, PID speed control is introduced because the PID can precisely control the motor speed by dynamically controlling the output power of each motor driver, thereby improving the performance of the powertrain.

(2) PID Speed Controlling Algorithm (CUI Sijie)

The reference speed will be set by the program, the hall sensor (motor encoder) will return the real-time rotation speed of the shaft, and the MBED as the controller will calculate the difference between the reference speed and the actual speed, which is considered as the speed between the real speed and the reference speed error. The controller will process the speed error (PID calculation) to get the modified PWM signal. The motor driver will output the corresponding control voltage to the motor based on the modified PWM signal. The shaft speed gradually approaches the set reference speed with the help of PID. The control mechanism of PID in the rover is shown in Figure 15.

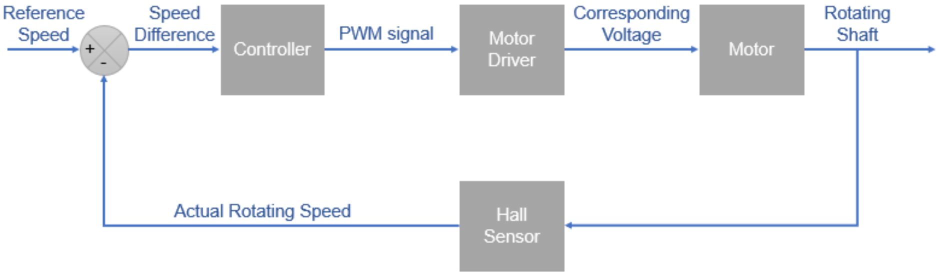


Figure 15. The control mechanism of PID control.

The expression for the PID control can be expressed as

$$PWM_{current} = PWM_{previous} + K_p \times \left(V_{reference} - V_{actual}\right) + K_D \times (V_{Error} - V_{lastError}) + K_I \times \left(\sum_{n=0}^{current} V_{error}\right)$$

In the above equation, P control is to increase the PWM signal if the current speed is less than the reference speed and vice versa. D control is to help the system reach the set speed as soon as possible by reduce the oscillation. I control is to eliminate the static error of the system.

(3) PID Speed Controlling Algorithm (YAN Zecheng)

The debugging of PID parameters is a comprehensive process in which each parameter affects each other. Many attempts in the actual debugging process are very important and necessary. In adjusting process, there were three parameters $K_p,\ K_i$ and K_d should be adjusted. It is found PID algorithm would have an excellent performance if the three parameters were adjusted separately. As a result, in the adjusting process, the parameter K_p at the beginning was adjusted and then $K_i\ and\ K_d$. Eventually, some small changes of the values of three parameters were conducted.

Parameter K_p reflected the magnitude of data volatility. The larger the value of K_p is, the larger the fluctuation of the data will be. In real practice, the value of K_p is needed, which ensures the data reach the reference value in the shortest time. Table 10 would show some of the results of the adjusting process.

Table 10. Value of K_p & Time of data reaching reference value.

K_p	Time (s)
0.1	15
0.01	12
0.001	8
0.0001	4
0.00001	6
0.0005	3.8
0.0006	4.2
0.0007	4.3
0.0008	4.3
0.0009	4.4

After comparing the time that the data reaches the reference value, it is found that the parameter should be between 0.00001 to 0.0001.

To adjust the second parameter K_d, K_p is set to 0.0004, 0.0005 and 0.0006 as its initial values to observe how the PID algorithm behaved (the time of reaching the reference value). Testing results are demonstrated in Table 11.

Table 11. Values of K_p and K_d & Time of data reaching reference value.

K_p	K_d	Time (s)
0.0004	0.0001	4.0
0.0004	0.00001	3.8
0.0004	0.000001	4.1
0.0005	0.00001	4.1
0.0006	0.00001	4.0
0.0004	0.00002	4.0
0.0005	0.00002	4.0
0.0006	0.00002	4.0
0.0004	0.00003	3.8
0.0005	0.00003	4.3
0.0006	0.00003	4.2
0.0004	0.00004	3.8
0.0005	0.00004	4.0
0.0006	0.00004	4.3

After adjusting, it is found that the appropriate interval is between 0.000001-0.00001. When setting the value of K_d to one-tenth of K_p, time of data reaching the reference value would be decreased.

Finally, parameter K_i is adjusted. It is found that the value of the K_i had an impact on the final stable number. The greater the value of K_i was, the greater the final value of data would be. For example, if the reference value was 300, the final value of data would be 310 when K_i was equal to 0.000003, and the final value of data would be 315 when K_i was equal to 0.000005. Experiment results are shown in Table 12.

Table 12. Values of K_p, K_d and K_i and time of data reaching reference value.

K_p	K_d	K_i	Time (s)
0.0005	0.00005	0.00000045	3.8
0.0005	0.00005	0.00000055	4.5
0.0005	0.00005	0.00000065	7.2
0.0009	0.00009	0.00000045	3.5
0.0009	0.00009	0.00000055	5.0
0.0009	0.00009	0.00000065	6.8

Finally, two sets of parameters were obtained. Compared with the first set of parameters, the second value has greater fluctuations but the corresponding reaching time would be decreased. Finally, we chose the second set of parameters $K_p = 0.0009$, $K_d = 0.00009$ and $K_i = 0.00000045$.

(4) PID Speed Control System Performance (YAN Zecheng)

Table 13 and Figure 16 will illustrate the performance of the PID speed control system. We use the final deviation distance of the rover from the set 1m, 2m, and 5m routes as the standard to test its performance

Table 13. Performance of chosen PID speed control system.

Speed of motor (rad/s)	Deviations for 1m route (cm)	Deviations for 2m route (cm)	Deviations for 5m route (cm)
100	1	5	13
150	3	5	13
200	2	7	14
250	3	6	16
300	3	8	18
350	2	7	18
400	3	9	20

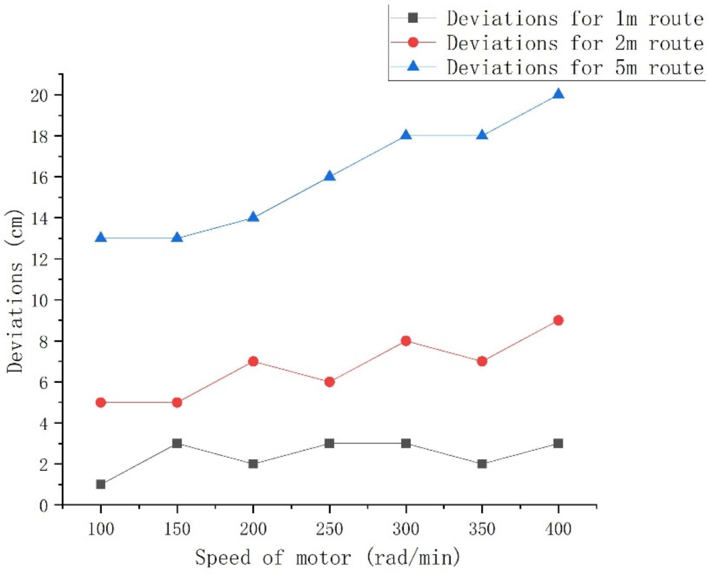


Figure 16. Performance of PID control system in three conditions.

3.2.3 Steering System (CUI Sijie)

The steering system needs a highly stable and accurate performance to complete both patios. Timer and compass had been proposed to perform the steer. However, the accuracy of steering by using a timer is highly dependent on the battery power, which will be inaccurate when the battery power is low. For the compass, the ambient electromagnetic field would greatly affect the stability of the steering system. However, by using the encoder, stability and accuracy can be promised.

The encoder could accurately measure the rotation speed which is independent of the ambient environment and battery power. By controlling the two tracks to turn the same number of turns in opposite directions, the rover can steer accurately.

Table 14. Relationship between number of turns and steering angle and steering accuracy.

Number of turns (s)	Steering angle (degree)	Steering accuracy (degree)
900	45	±5
1300	90	±5
1700	135	±10
2100	180	±10
1500	270	±10

The accuracy is acceptable and the stability is very high.

3.3 Gravel Path Tracking Feedback Control System

This module aims at creating a control system for the rover to recognise and travel along the gravel. This system is designed to be a digital closed-loop system so that the rover can constantly correct its offset from the centre of the road. The system mainly consists of three submodules: a controller, the rover motors, and a gravel recognition sensor. This sensor includes an ensemble of two recognition algorithms for both the dark (natural) gravel and the bright (painted) gravel. This system is implemented across different hardware, including the mbed controller, the OpenMV camera, the motors, and the interfaces. Figure YP-1 shows the block diagram representation of the system.

3.3.1 Objectives and System-Level Design (YANG Pei)

This module aims at creating a control system for the rover to recognise and travel along the gravel. This system is designed to be a digital closed-loop system so that the rover can constantly correct its offset from the centre of the road. The system mainly consists of three submodules: a controller, the rover motors, and a gravel recognition sensor. This sensor includes an ensemble of two recognition algorithms for both the dark (natural) gravel and the bright (painted) gravel. This system is implemented across different hardware, including the mbed controller, the OpenMV camera, the motors, and the interfaces. Figure 17 shows the block diagram representation of the system.

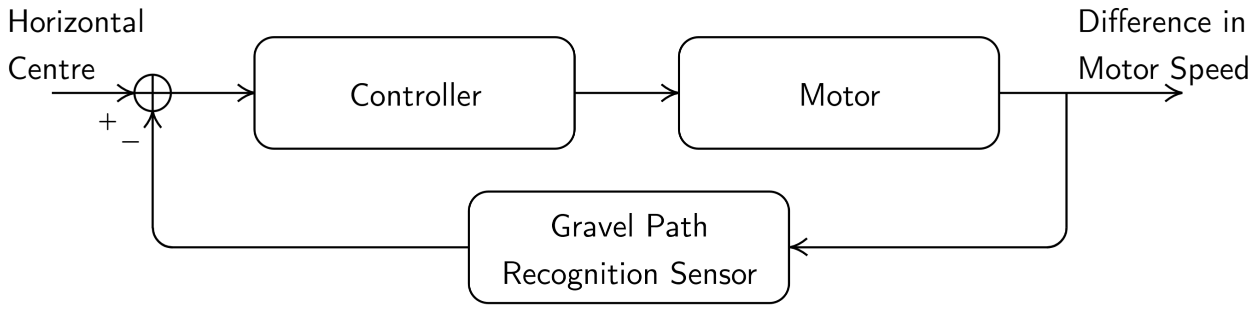


Figure 17. Block diagram of the gravel path tracking feedback control system.

Principle of the system. When this system functions, the sensor first captures an image of the gravel section in front and performs pixel-level semantic labelling of the gravel. Based on this information, the controller then computes the offset of the rover from the centre of the road. The offset value is then projected to the difference between the speed of two motors so that the rover can make a turn while travelling forward. This system will function under the command of the mbed central controller.

Analysis of the main challenge. The gravel path on Patio 1 can be divided into two sections based on the colour (or brightness): the dark section in Patio 1 Task 1 and the bright section in Patio 1 Task 3. Figure 18 is an image of the dark road section, showing that the difference between the gravel and the background is similar in colour and texture. Therefore, the recognition of the dark gravel section would be challenging.

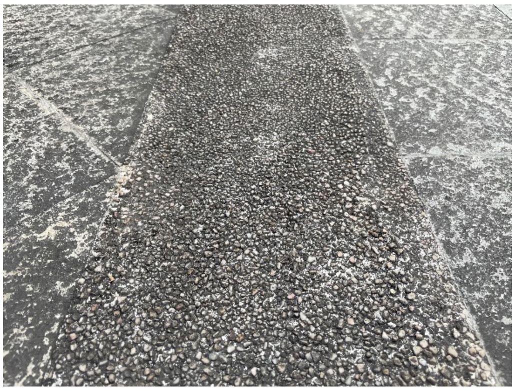


Figure 18. Image of the dark section of the gravel captured on an iPhone 12 mini with default settings. The gravel is similar in both colour and texture with both sides of background.

Referring to the block diagram in Figure 17, the design of the gravel recognition sensor will be introduced in Section 3.3.2 for the dark section and Section 3.3.3 for the bright section. Section 3.3.4 will introduce the design of the controller algorithm, while Section 3.3.5 will introduce how the signals control the motor speed.

3.3.2 Gravel Recognition – Dark Section

3.3.2.1 Objectives and Technical Route (YANG Pei)

The dark gravel recognition algorithm aims to semantically distinguish the pixels on and off the gravel on a captured image by applying image processing techniques. The essence of this technique is a 5×5 application-specific convolutional kernel, which we obtained through deep learning techniques.

This design's idea is: Achievements in Computer Vision have greatly benefited Deep Learning development. Can we be benefited backwards?

We began with training a shallow FCN network [1] that performs semantic segmentation on the gravel. By exploiting the features of this network, we then obtained a 5×5 convolutional filter capable of extracting a number of feature points on the gravel. We then implemented the algorithm on the OpenMV and performed field tests to finetune the algorithm.

3.3.2.2 Dataset and Preprocessing (LI Sipei and YUE Haiyi)

We collected 118 images of size 320×240 using an OpenMV H7 Plus Camera with default settings. The pitch of view ranged from -15° to 75°, and the camera was 20-45cm above the patio surface. A corresponding binary semantic ground-truth mask was then manually annotated in Adobe Photoshop using "Polygonal Lasso" for each collected image. All the pixels contained by the lane were labelled in white (RGB #FFFFFF), while the rest background pixels were labelled in black (RGB #000000). Figure 19 shows an example image in the dataset with its ground-truth mask.

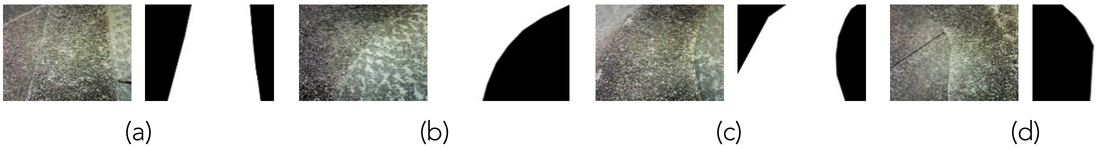


Figure 19. Image collected by an OpenMV camera with its annotated ground-truth mask.

Once the dataset was annotated properly, 294 images were extracted to form a 147-image training set and a 147-image test set. Preprocessing of the images includes resizing to 384×288, converting to grayscale, and normalising the pixel values.

3.3.2.3 Constructing Auxiliary Network Model (YANG Pei)

The auxiliary FCN network was designed to be shallow, with only seven single-layer convolutions. Figure 20 shows the structure of this network.

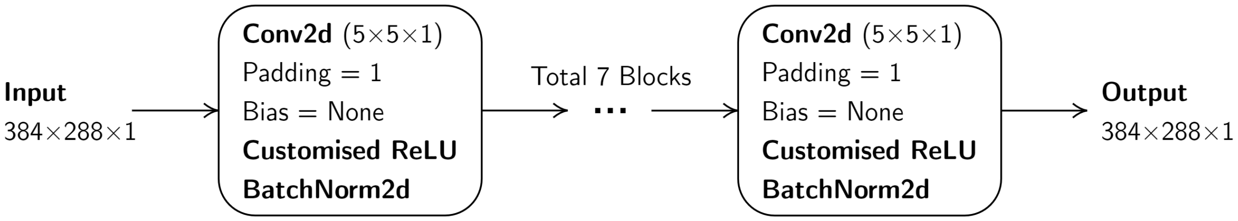


Figure 20. Structure of the shallow FCN. This network includes a total of seven convolutional blocks that share the same structure.

Customised activation layer. Since OpenMV built-in functions will automatically truncate all negative values after each arithmetic calculation, to adapt our network to this feature, the activation function after each convolutional layer was exclusively customised to be

$$f(x) = \mathrm{Clip}\{x, 0, 1\}$$

which was realised by cascading two ReLU layers computing $ReLU(x)$ and $ReLU(1-x)$.

Convolutional layers. Due to the same automatic truncation problem, we removed the bias of each convolution layer to prevent possible future technical obstacles regarding OpenMV implementation.

Loss function. The loss function was determined to be a weighted combination of Binary Cross Entropy (BCE) loss and Dice loss,

$$\mathbb{L} = \frac{3}{4}\mathbb{L}_{BCE} + \frac{1}{4}\mathbb{L}_{Dice}$$

For clarification, Dice loss is given by

$$\mathbb{L}_{Dice} = 1 - \frac{2|X \cap Y|}{|X| + |Y|}$$

where X and Y are two images. $|X \cap Y|$ represents the number of true positive and true negative pixels while $|X| + |Y|$ represents the total number of pixels in X and Y.

Comparison studies. Note that this network structure stands out from the 25 network models we have trained based on segmentation performance. Please refer to our lab notebook for the performance of other models.

3.3.2.4 Network Training and Performance (YANG Pei)

Once the network was constructed, it was trained on an Nvidia RTX 3060 GPU for 100 epochs at a learning rate of 0.01.

Each epoch ends with a validation, during which the validation loss was recorded, and the temporary model was archived. Once the training was completed, we selected the model with the lowest validation loss as the trained model. Note that the performance of all models is available on request.

3.3.2.5 Exploiting Computational Features from the Auxiliary Model (YANG Pei)

We visualised each network layer by manually forward propagating the entire dataset through the layers one by one to exploit useful computational features.
Figure 21 visualises the original and filtered images before the first, second, and third Batch Normalisation layers.

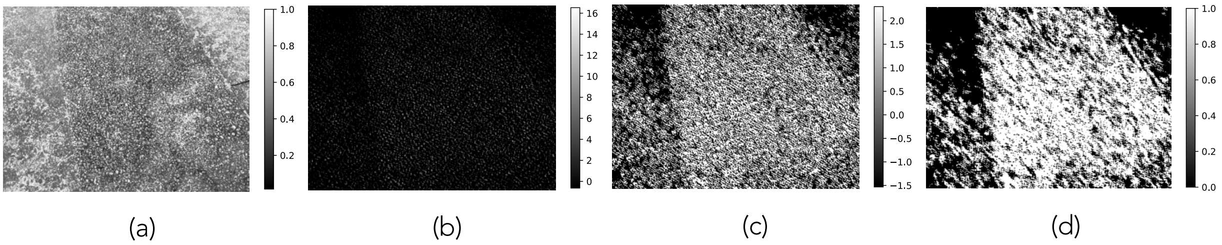


Figure 21. Visualisaiton of (a) an input image propagating into the network before (b) the first BN layer, (c) the second BN layer.

We observed that the first convolution layer had already successfully differentiated the texture by generating sparsely distributed bright dots on the gravel, which is similar to the result but more effective than a Laplace edge detector. As the image continued to propagate through deeper layers,

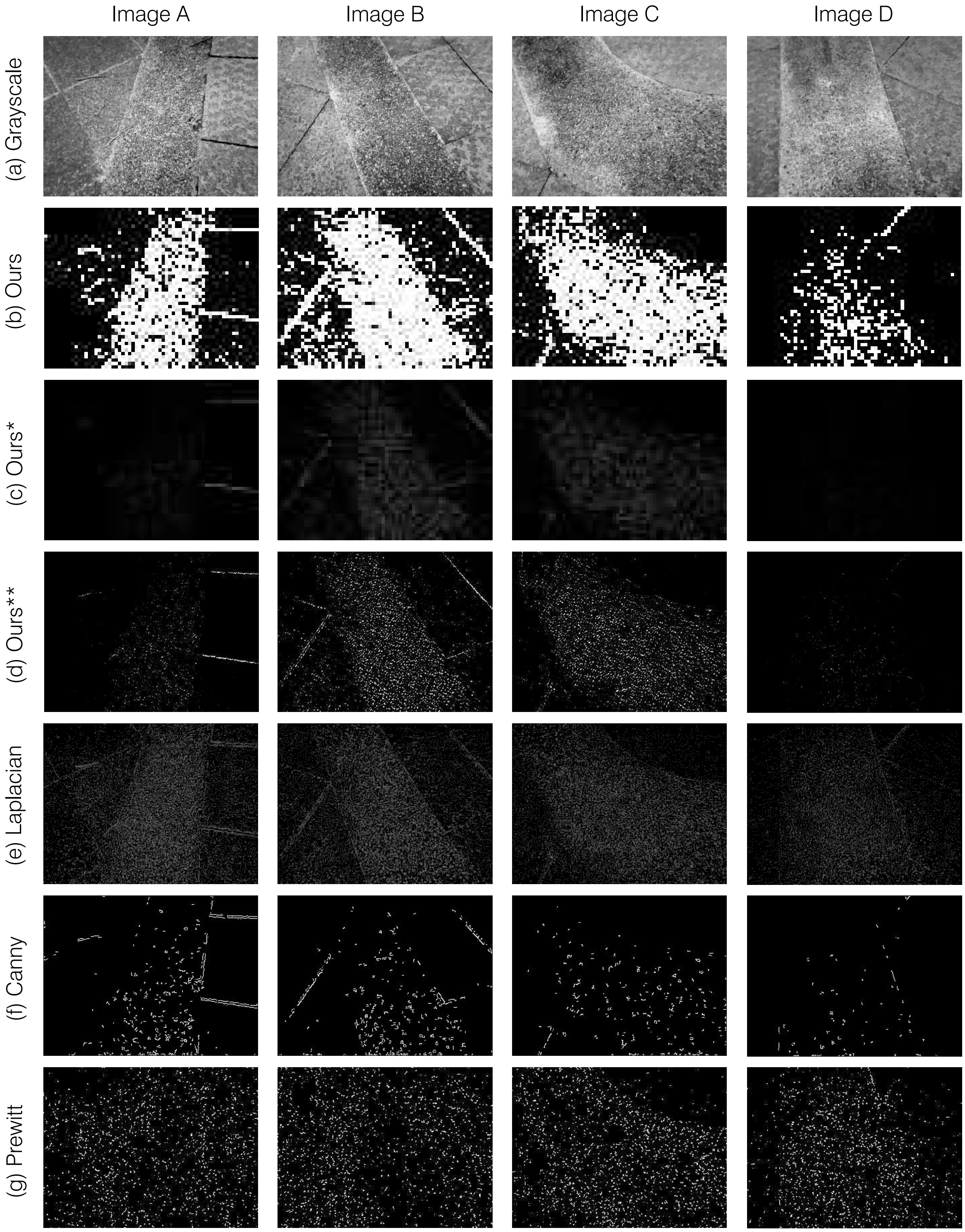


Figure 22. Comparison of results on (a) grayscale, (b) our proposed method, (c) our proposed method without binary thresholding, (d) our proposed method without low-pass filtering, (e) Laplacian edge detection with brightness increased by 40% and contrast increased by 40%, (f) Canny edge detection with lower threshold of 10 and upper threshold of 240, and (g) Prewitt edge detection. All images were processed on an OpenMV and captured from the OpenMV cache buffer. Images in group (d) shows better differentiation between the gravel and the background than images in (e), (f), and (g). Qualitatively, images in group (b), compared to group (d), significantly increase the true positives while maintaining the proportion of false positives.

the convolutional filters helped spread the bright dots more uniformly over the entire gravel section.

Based on this observation, we extracted the first convolutional layer as it helps distinguish the gravel from the background. The kernel is given by

$$K = \frac{1}{10000}\begin{bmatrix} -876 & -978 & -276 & 2459 & 1258 \\ -23 & -2564 & -2519 & 97 & 1411 \\ 2527 & -88 & 16 & 1429 & -2016 \\ 648 & -506 & -816 & 1628 & -619 \\ -348 & 899 & -138 & -1040 & -677 \end{bmatrix}.$$

Additionally, copyright information has been embedded in the weight of this convolutional kernel, in case of possible issues.

3.3.2.6 Noise-Reduction Filter Design (YANG Pei)

Figure 21 (d) shows a considerable number of false positives, which could severely reduce the accuracy of gravel recognition and potentially cause the rover to run off-track. We believe this is due to the high similarity in texture between the texture of the gravel and other parts of the patio surface. Figure 21 (c) shows that the false positive pixels distribute more sparsely than the true positive pixels, implying that the false positives occupy a higher frequency band. Therefore, a low pass filter could help reduce the false positives.

Typical low pass filters include the moving-average filter [2], given by

$$y[m,n] = \frac{1}{N^2} \sum_{l=m-N+1}^{m} \sum_{k=n-N+1}^{n} x[l,k]$$

Increasing the step of moving to the order of the kernel simplifies a moving-average process into a mean-pooling operation.

Finally, time-domain binary thresholding was applied to distinguish the gravel from the background. We determined the threshold at 20 through repeated field tests, as this value balances well between false positives and true negatives.

3.3.2.7 Algorithm Formulation (YANG Pei)

The dark gravel path recognition algorithm is given by

$$y = \mathrm{MeanPooling}\left(\frac{1}{10000}\begin{bmatrix} -876 & -978 & -276 & 2459 & 1258 \\ -23 & -2564 & -2519 & 97 & 1411 \\ 2527 & -88 & 16 & 1429 & -2016 \\ 648 & -506 & -816 & 1628 & -619 \\ -348 & 899 & -138 & -1040 & -677 \end{bmatrix} \circledast x \right) > 20,$$

where $\circledast$ is a discrete-convolution operator and $>$ is a comparison operator.

3.3.2.8 Qualitative Analysis (YANG Pei)

Comparison with other methods. Figure 6 compared our proposed method in gravel recognition with several general-purpose edge detection algorithms. Results in Figure 6 show that both Canny and Prewitt have poor performance in performing this task. While the Laplacian filter generates clustered dots in the gravel area, it has many more false positives than our method. The results show that our application-specific image processing technique outperforms several general-purpose ones.

Verification of the effectiveness of low-pass filtering. Figure 22 shows that low-pass filtering and thresholding marginally increase the false-positive rate but significantly increase the true-positive rate, which leads to a better differentiation between the foreground patio and background ground surface. Therefore, low-pass filtering is considered highly effective.

3.3.3 Gravel Recognition – Bright Section (YANG Pei)

Similar to the purpose of recognising the dark gravel section, this algorithm aims at semantically distinguishing the gravel from other parts of the patio surface. Figure 23 includes images of the bright gravel, which shows a clear brightness distinction between the gravel and its sides. We, therefore, performed brightness thresholding on grayscale captured images of size 80×60.

We conducted repeated field tests and set the threshold at 180 to best balance the false positive and true negative rates. Hence the algorithm is given by

$$y = x > 180.$$

Figure 23 shows results of bright gravel recognition.

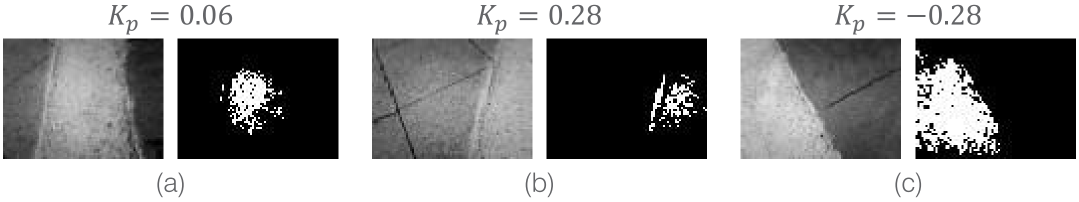


Figure 23. Results of bright gravel recognition with corresponding proportional gain.

3.3.4 Controller Algorithm (YANG Pei)

We adopted proportional control for simplicity. A normalised proportional gain is acceptable since the proportional gain would eventually be mapped to the difference in PWM mark-to-space ratios that create the difference in motor speed. The controller algorithm, therefore, simplifies into a proportional gain describing to what extent the rover is away from the centre of the gravel.

In our design, we anchored the proportional gain K_p with the horizontal expectation of an input segmentation mask X. Suppose x_{ij} is an element at location (m, n) in the two-dimensional segmentation mask X of size $a \times b$. With N representing the number of positive pixels, the proportional gain is given by

$$K_p = E(m) = \frac{1}{N} \sum_{\text{All positives}} i \times x_{ij}.$$

To accelerate the computation, we performed matrix operation to compute K_p, which is given by

$$K_p = \frac{1}{\sum x_{ij}} X^{\top} \begin{bmatrix} 0 & 1 & \cdots & a-1 \\ 0 & 1 & \cdots & a-1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \cdots & a-1 \end{bmatrix}.$$

Figure 24 shows the results of the controller algorithm on the dark gravel, while Figure 23 shows the results on the bright gravel. The K_p follows the intuition that a negative value is generated when the gravel deviates to the left side of the view, while a positive value is generated vice versa.

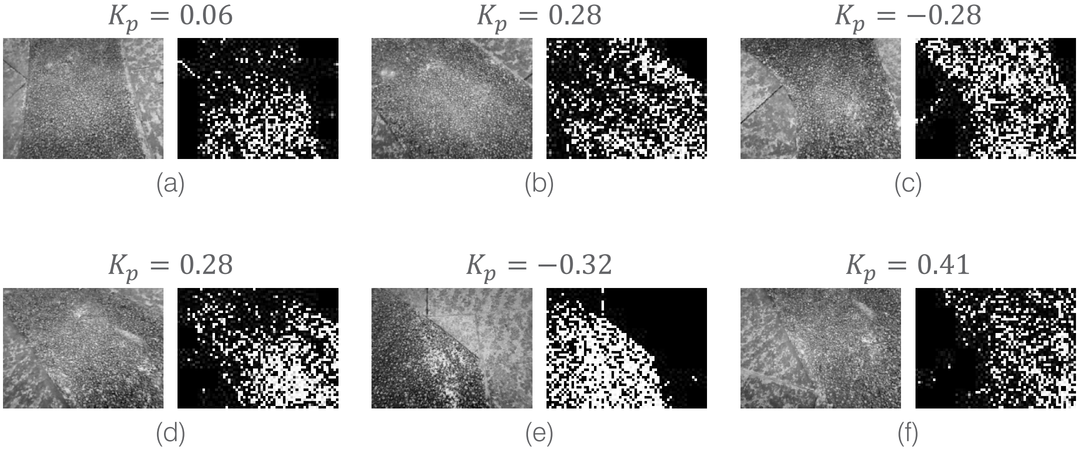

$K_p = 0.06$ (a)
$K_p = 0.28$ (b)
$K_p = -0.28$ (c)
$K_p = 0.28$ (d)
$K_p = -0.32$ (e)
$K_p = 0.41$ (f)

Figure 24. Results of dark gravel recognition with corresponding proportional gain.

3.4 Shape Detection and Recognition

3.4.1 Overview (WU Haoyang)

Patio 2 task 1 is one of the most challenging tasks in TDPS, in which the rover needs to locate and identify contours printed on standing boards and hit the boards with the same shape. Many previous methods can be used to detect and identify contours [3], [4], but this mission becomes extremely difficult in a real situation because of intricate illumination conditions and interferences from background items. To this end, we designed a feature-driven de-noising algorithm, which worked reliably in practical conditions. Three shape matching algorithms were proposed throughout the project but only one based on Ratio-of-Area (RoA) and Central Fill Factor (CFF) was proved stable and precise enough to accomplish this task. We also developed a Visual Auxiliary System (VAS) based on RoA to fix the trajectory of the rover.

3.4.2 Implementation

The shape-matching algorithm consists of four stages, including image preprocessing, target localization, pattern matching and result de-noising & encoding. Background de-noising is applied throughout target localization to improve the accuracy of detection. The VAS shares the output of target localization with shape detection but provides guidance on expected direction rather than the type of shape. The flow chart of our algorithm is shown in Figure 25.

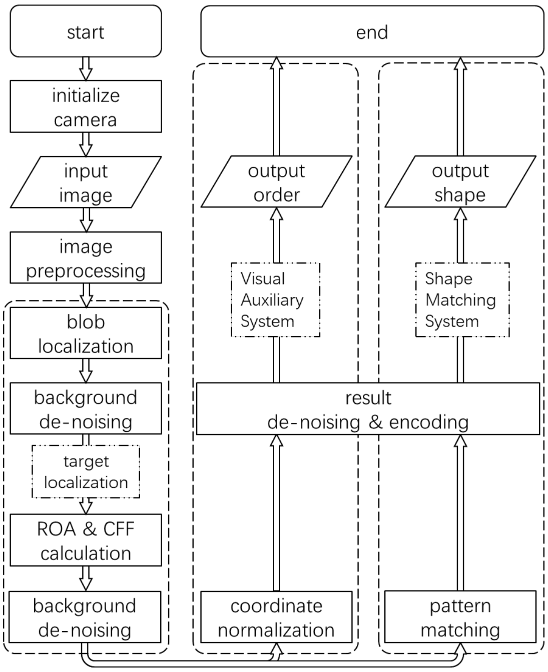


Figure 25. Algorithm flow chart.

A. Image Preprocessing (WU Haoyang)

Image preprocessing is a commonly used approach in computer vision tasks. The application of specific algorithms and filters, such as histogram equalization [5], Gaussian filter [6] and Laplacian edge detection [7] can greatly decrease the complexity of images while emphasizing the desired feature. In stage one, we first convert the image to greyscale (Figure 26 b) to reduce the redundant colour information as well as alleviating the pressure on computation. After that binarization is used to raise the contrast to emphasize the black contours printed on white paper (Figure 26 c). Finally, the Laplacian edge detection is applied to extract lines from the image (Figure 26 d), which is one of the building blocks of contours.

B. Target Localization (WU Haoyang)

One of the toughest parts of this task is to recognize the desired contours from countless disturbing items in the background, such as shadows, dark leaves, and humans in black and windows. This requires the algorithm to be of high accuracy and efficiency. To solve this problem, we proposed RoA estimation, which is given by

$$RoA = \frac{\sum_{all\ P}(P < threshold)}{N}$$

where P is the grey value of pixels and N is the number of pixels. By detecting the dark blobs and applying statistics on pixels inside the circumscribed square, we can calculate the RoA of these blobs.

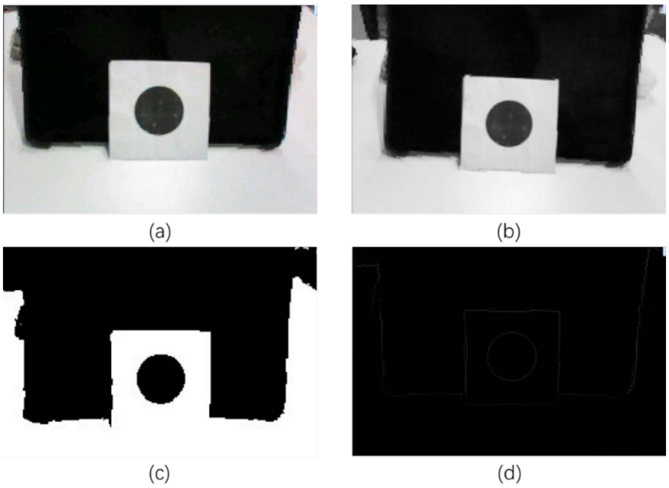


Figure 26. (a) Input image. (b) Input image in grayscale. (c) Input image after binarization. (d) Input image after edge detection.

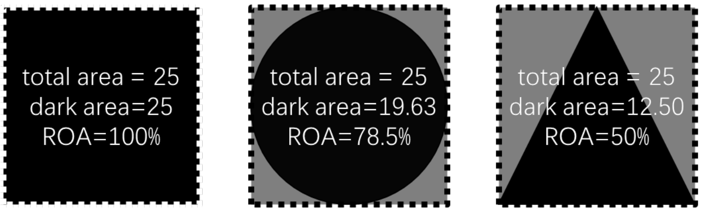


Figure 27. Characteristic ROA of rectangle, circle and triangle.

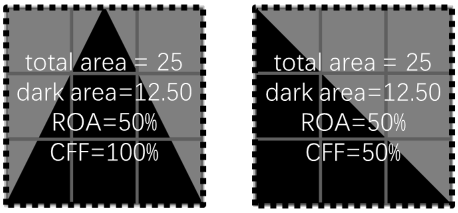


Figure 28. Two contours have the same ROA, but the right one is not a valid target. Hence, CFF is used to distinguish them.

As the size of the contours and the paper is fixed, the ratio of dark area on the paper should be constant, which are 100%, 78.5% and 50% respectively according to the specifications.

This method is simple, efficient and easy to implement. It is especially robust in triangle and circle detection because the RoA is consistent in terms of rotation, shift and slight perspective transform. By detecting dark blobs in the image, we can pick Regions-of-Interest (RoI) from the original image. This method is further developed as we discovered its drawbacks in experiments. The problem is that RoA only restricts the number of pixels while neglecting the distribution of them, which led to misrecognition occasionally. In this case, another feature shared by these contours was introduced to improve the algorithm. As the contours are placed in the middle of the paper, the central part of the paper should always be 100% black. In this case, we split the RoI into a 3*3 grid (Figure 28 and Figure 29) and calculate the ratio of dark pixels in the central block, which is called the "Central Fill Factor" (CFF). Only those with RoA larger than 0.5 and CFF of 1.0 will be recognized as valid patterns (Figure 28).

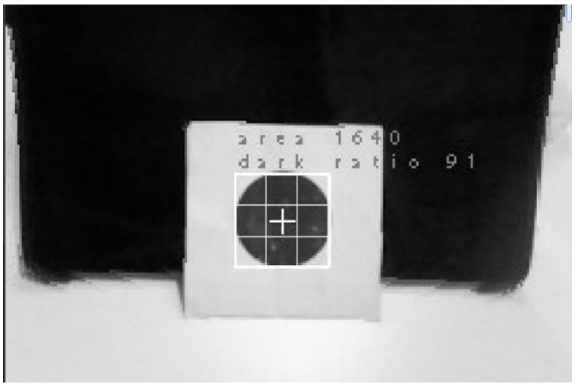
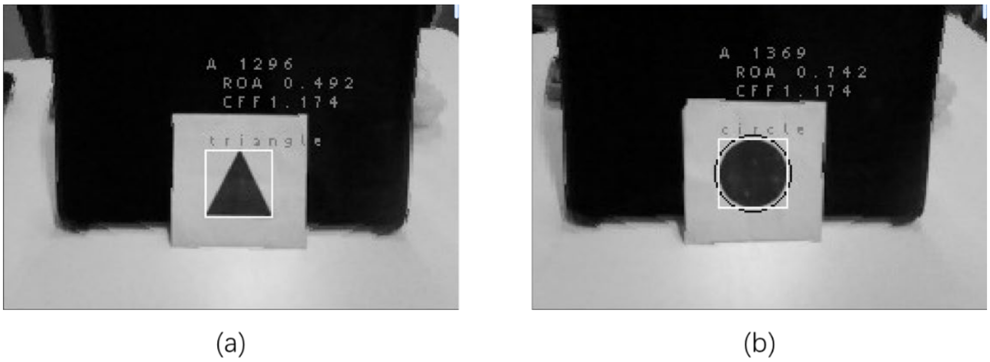


Figure 29. CFF estimation in target localization. The upper number is the area of the contour. The lower number is the CFF of the contour.

C. Pattern Matching (WU Haoyang)

Traditional methods of pattern matching focus on finding features that are mathematically consistent after scaling or geometric transforms. These approaches provide stable and accurate evidence for pattern matching but come with high computation and memory consumption, which is not usually applicable to embedded systems. In this case, we chose to reuse the features in previous stages. By comparing input RoA with the three eigenvalues shown in stage two, we can distinguish these contours (Figure 27 and Figure 30). In practical implementation, these eigenvalues are expanded to three characteristic ranges to adapt perspective transformation and improve the robustness of the algorithm. Experiments showed that 50% worked well with triangles, but 10% and 78.5% are sometimes indistinguishable for circles and rectangles. So, we inserted a circle detection function based on Hugh transform before the detection of other patterns. Even though we have implemented several techniques to reduce misrecognition, we cannot achieve 100% accuracy in the task. Hence statistical control is applied to the output of the algorithm, which is introduced in 4.3.2 E. With the combined effort of all these optimizations, we finally carried out our algorithm for patio 2 task 1.



(a) (b)

Figure 30. The area, ROA, CFF and recognition result from two contours.

D. Visual Auxiliary System (XU Ben)

The VAS was developed based on the second stage of shape detection. It is difficult to make sure that the rover can move in a straight line only with the PID feedback system. To make sure that the rover will move to the position of the paper, a shape-based navigation system is designed. The camera will keep detecting the shape. If there is no shape detected in the captured figure, the rover will move in a straight line by the PID feedback system. If the shape is detected, the x-axis of the centre of the shape will be used to navigate the rover. The rover will move to the left If the detected shape is on the left of the figure and vice versa. A PWM base left turn and right turn is applied.

The transformation of the left-right deviation to PWM output is shown in Table 15. The deviation is obtained by calculating the average abscissa X of the pixels of the shape. The PWM output will be (0.4, 0.85) if X<0.4, which means the rover will make a left turn. In this way, the rover can keep adjusting its direction until it achieves the position of the shape.

Table 15. Transformation of left-right deviation to PWM output.

Deviation	Non-detected	X<0.4	0.4<X<0.6	X>0.6
PWM	(0.5,0.5)	(0.4,0.85)	(0.5,0.5)	(0.85,0.4)

To determine the position to stop, the image size of the shape is used. According to the experiment results, the rover will be around 30cm away from the shape with an image size of 1500. In addition, the accuracy of the shape detection is very high at this distance. As a result, the rover will stop once it detects the size of the shape is greater than 1500. After that, it will determine if it is triangular, rectangular or circular.

E. Background Denoising (WU Haoyang)

The outputs of the shape detection and tracking algorithm are mostly correct but far from flawless. However, the accuracy of identification and VAS should reach 100% or the failure in one task will influence all successive tasks. In this case, we designed a result de-noising algorithm, in which the nearest 10 predictions are stored in a queue and the most frequent result becomes the system output. It is proved highly reliable in both shape detection and VAS. We also designed an encoding paradigm to make the prediction and guidance of the algorithm comprehensible and coding-friendly. Results initially represented by float numbers are converted to numerical instructions in accordance with the requirement of the system designer, which has saliently reduced coding and debugging expenditure.

3.4.3 Runtime Testing

A. Sound Region (LI Sipei)

The system testing was conducted to obtain the valid distance and angle for the Visual Auxiliary System. This process includes two phases, indoor testing and outdoor testing.

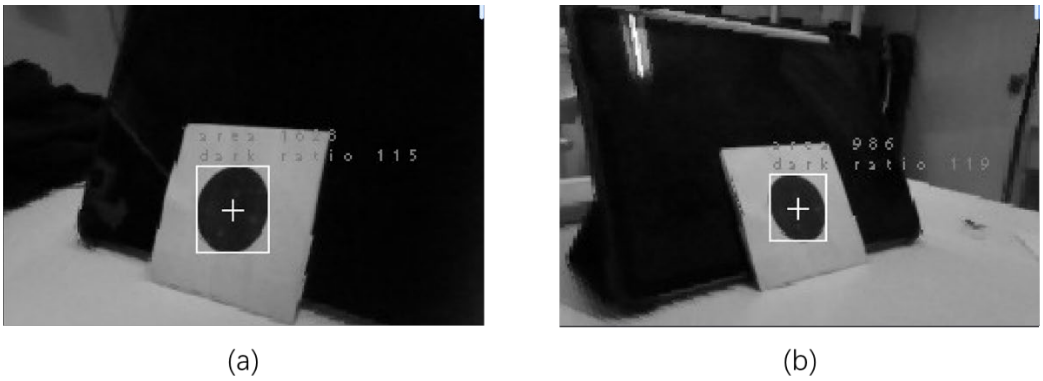
For indoor testing, figures were demonstrated by iPad, while for outdoor testing, field testing was conducted. The results are displayed in Table 16

Table 16. Results for indoor and outdoor testing.

Angle (°)	Indoor Testing Result	Outdoor Testing Result
-90	fail	fail
-85	fail	fail
-80	fail	fail
-75	fail	fail
-70	fail	fail
-65	fail	fail
-60	fail	fail
-55	fail	fail
-50	fail	fail
-45	occasionally fail	fail
-40	acceptable	occasionally fail
-35	acceptable	occasionally fail
-30	acceptable	acceptable
-25	acceptable	acceptable
-20	acceptable	acceptable
-15	acceptable	acceptable
-10	acceptable	acceptable
-5	acceptable	acceptable
0	acceptable	acceptable
5	acceptable	acceptable
10	acceptable	acceptable
15	acceptable	acceptable
20	acceptable	acceptable
25	acceptable	acceptable
30	acceptable	occasionally fail
35	acceptable	occasionally fail
40	occasionally fail	fail
45	occasionally fail	fail
50	fail	fail
55	fail	fail
60	fail	fail
65	fail	fail
70	fail	fail

Angle (°)	Indoor Testing Result	Outdoor Testing Result
75	fail	fail
80	fail	fail
85	fail	fail
90	fail	fail
-90	fail	fail
-85	fail	fail

Indoor Testing Result Analysis. The above results show quite satisfying performance. It is concluded that in the rough region from -40° to 35°, the Visual Auxiliary System could work properly. When the angles are -45°, 40° and 45°, the system would face occasional failure. In addition, the system cannot work beyond the region mentioned about.



(a)

(b)

Figure 31. Recognition under severe perspective distortion. (a) Recognition at 35° on the right. (b) Recognition at -40° on the left.

Outdoor Testing Result Analysis. The above results are not as good as indoor testing. It is concluded that the acceptable range is within -30° to 25°. For angles -40°, -35°, 30°and 35° the system would face failure occasionally.

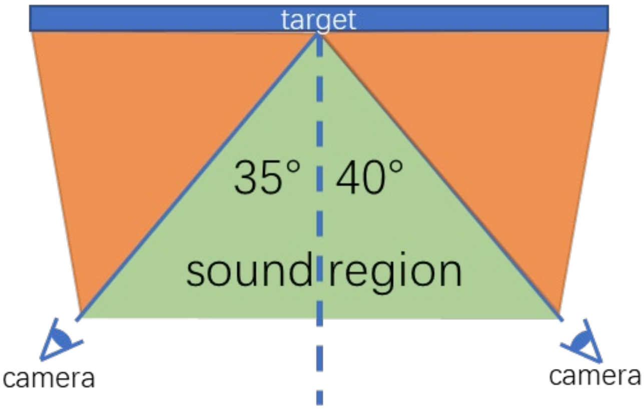


Figure 32. Sound region of the algorithm.

Compared to the results from the two testing conditions, indoor testing has a better performance than outdoor testing since the figures placed on the iPad are clearer, enabling the system to achieve the task more accurately.

B. Illumination (WU Haoyang)

The VAS was tested in 18 runs under three illumination conditions, including cloudy, sunny-normal and sunny-scorching. The performance of the rover was scored in four levels: good, acceptable, improper direction and yawing, based on the trajectory of the rover under guidance. Good means the rover was perpendicular to the standing board. Acceptable indicates that the rover was not perpendicular to the board but still in the sound region. While improper direction means the rover was heading towards the board but the outside sound region and yawing is a total loss of target.

Results from Table 17 show that our method worked well in cloudy and sunny-normal conditions with no yawing and a few improper directions while achieving only 66.67% acceptable and above runs in sunny-scorching conditions. The VAS performed best under sunny-normal weather, with all good results except for one acceptable. The worst results were recorded under sunny-scorching, in which the trajectory of rover diverse went to the extreme. 33.3% of runs failed the task and no improper direction occurred. This result is beyond our expectation that better illumination conditions could improve the accuracy of the algorithm. It is probably because when sunlight exceeds a certain threshold, excessive sunlight is reflected by the black ink, making the target region brighter than the configured grey value so that the algorithm cannot identify dark pixels. A later improvement was made to extend the sensitive range of detection in case of extreme illumination.

Table 17. Test results of VAS in various illumination conditions.

	Cloudy	Sunny-normal	Sunny-scorching
Good	15	17	10
Acceptable	2	1	2
Improper direction	1	0	0
Yawing	0	0	6

3.4.4 Conclusions (WU Haoyang)

We proposed a simple, efficient and robust framework to accomplish task 1 of patio 2. The number of dark pixels inside the RoI was used to calculate RoA and CFF, the characteristic values of patterns, are used for pattern matching and de-noising. We also applied a statistical control to the predictions of the model to ensure accuracy. Result encoding was used to construct a code-friendly and efficient system. Our main contribution is designing a feature-driven, effective de-noising algorithm without increasing computation. During runtime testing, the system is evaluated in terms of sound region and illumination conditions. Results showed that this module is efficient and reliable on most occasions. It achieved 100% and 94.4% acceptable runs under normal sunny and cloudy conditions, respectively, which satisfies the design specification.

3.5 Fence Tracing Feedback System in Patio 2

3.5.1 Objectives (CUI Sijie)

The objectives of this feedback system are to complete the transition between task 1 and task 2 (Figure 33, orange line), task and task 3 (Figure 33, green line) without using any beacon.

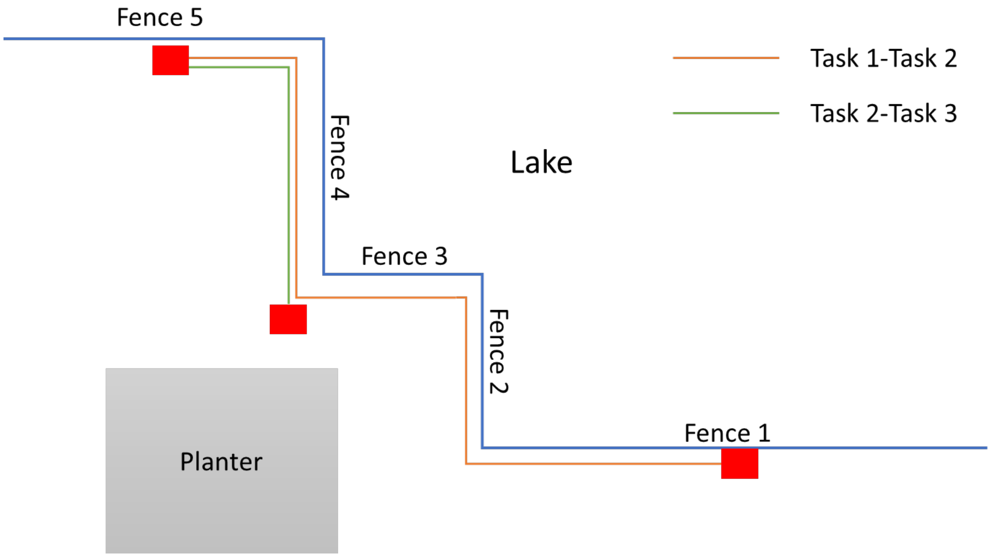


Figure 33. The transition route between tasks 1 & 2, 2 &3 in patio 2.

3.5.2 Fence Tracking Feedback System Working Mechanism (CUI Sijie)

Without using any beacon, we decide to track the fence by ultrasonic sensor because the position of the 5 fences (Figure 33) is fixed and this could improve the stability of the rover.
Looking at the planned path for the rover for transiting from task 1 to task 2 (Figure 33), the following working mechanism (Figure 34) could ensure that the rover moves from fence 1 to fence 5.

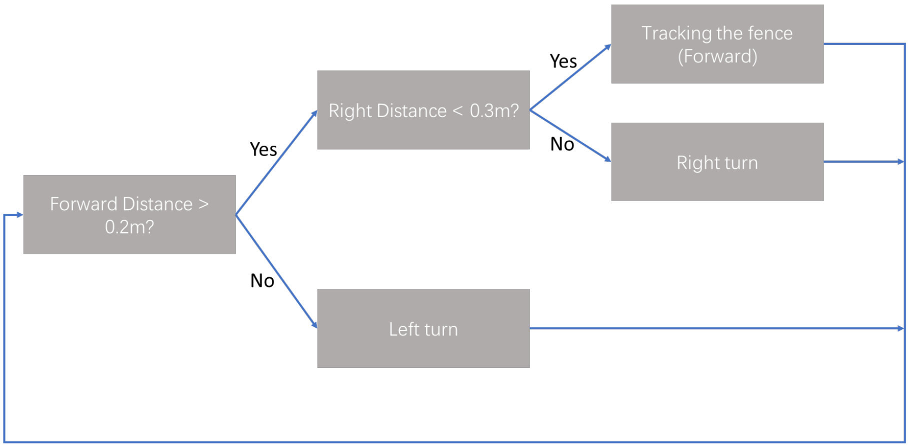


Figure 34. The fence tracking flow chart for transition between tasks 1&2

The working mechanism for the rover from task 2 to task 3 (along the green line in Figure 33) could be concluded as follow (Figure 35).

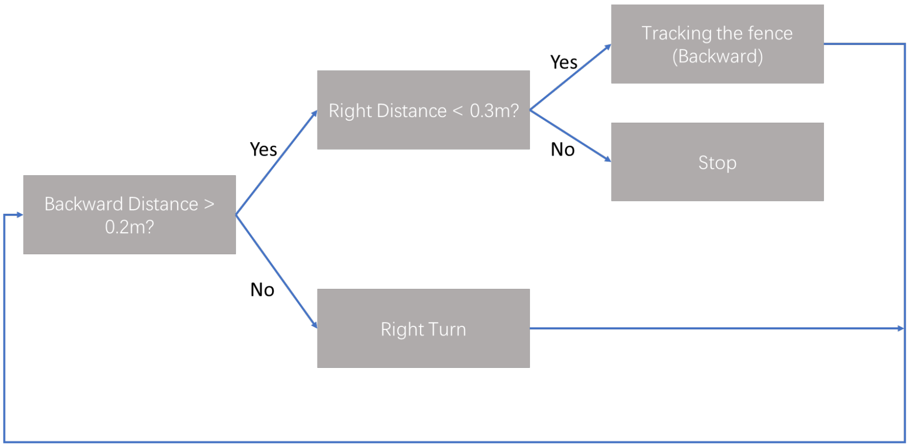


Figure 35. The fence tracking flow chart for transition between tasks 2&3.

3.5.3 Ultrasonic Sensor Fence Tracking Module

3.5.3.1 The Ultrasonic Sensors and Performance (CUI Sijie)

HC-SR04 is used and its performance of it when detecting the fence is given in Table 18.

Table 18. Performance of the ultrasonic sensor.

Test items	Performance
Maximal detecting distance	0.47m
Minimal detecting distance	0.02m
Sampling frequency	More than 50Hz
Detecting accuracy	Less than 1cm
Maximal incidence angle	27°

Table 18 verifies the feasibility of the rover to detect the distance between the rover and the fence in 3 aspects: the detecting range, the detecting accuracy and the detecting frequency.

- The reference distance between the rover and the fence is set to 15cm, which is within the detection range. In addition, the detection range of 2cm to 47cm also guarantees a safety margin of -13cm (right side) to +32cm (left side) for the rover to deviate from the reference distance.
- The detecting accuracy is 1cm, which is enough for controlling the distance.
- The sampling frequency is 50Hz, which allows the rover to adjust its yaw angle and deviation distance from the reference distance frequently.

3.5.3.2 Ultrasonic Sensors Installation Layout (CUI Sijie)

4 ultrasonic sensors are installed in the rover (Figure 36).

One ultrasonic sensor is installed at the front to determine the distance between the rover and the front barrier. Two ultrasonic sensors are installed on the right to track the fence. One ultrasonic sensor is installed in the back to determine the distance between the rover and the back barrier.

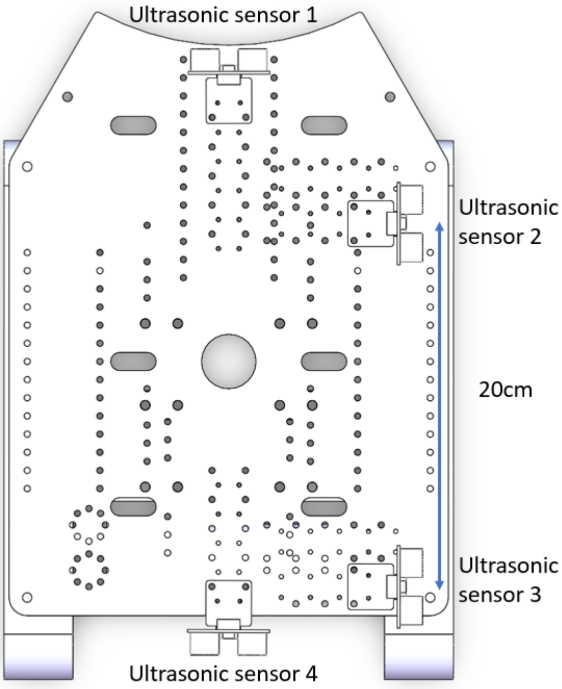


Figure 36. The ultrasonic sensor installation layout.

3.5.3.3 P-P Control Mechanism for Controlling the Distance (CUI Sijie)

When tracking the fence when the right distance is less than 0.3m, two ultrasonic sensors are used and the control mechanism is shown in Figure 36.

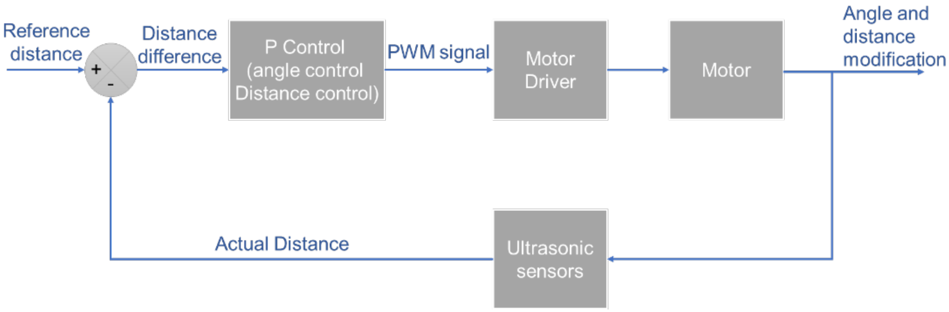


Figure 37. The fence tracking flow chart for transition between task 2&3.

In Figure 37, the MBED would control the motor dynamically based on the data collected by the ultrasonic sensors. This system uses two proportional controls, one for the yaw angle of the rover and the other for the distance between the rover and the fence. Although the method introduced in 3.5.3.6 can theoretically adjust the distance and angle at the same time, it is found in the field test that the effect of adjusting the distance does not satisfy our expectation, so another distance control is added to improve the distance adjustment performance. Two proportional controls are executed in parallel, which ensures the real-time performance of system adjustment to the greatest extent.

3.5.3.4 Yaw Angle Calculation (CUI Sijie)

To apply the distance P control and angle P control, the yaw angle has to be obtained at first. Based on CUI Sijie-7, The Yaw angle could be expressed by

$$\theta = \arctan\left(\frac{right1 - right2}{\Delta l}\right)$$

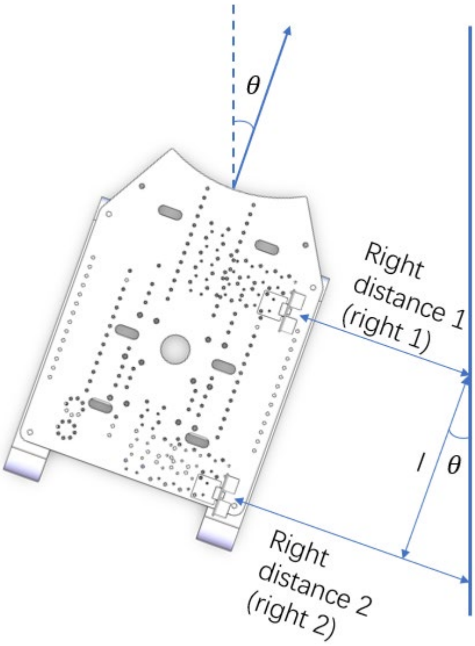


Figure 38. Yaw angle when tracking the fence.

3.5.3.5 Distance P Control (CUI Sijie)

The distance control is to control the distance between the rover and the fence. The distance between the rover and the fence can be expressed by

$$current\ distance = \frac{right1 + right2}{2} \times \cos(\theta)$$

When the rover is advancing and the current distance is larger than the reference distance, the rover should move forward right, indicating the speed of the left track should increase while the speed of the right track should decrease and vice versa. This can be realized by the following P control equation:

$$PWM_{left\_previous} = PWM_{left\_current} + Kp \times (current\ distance - reference\ distance)$$
$$PWM_{right\_previous} = PWM_{right\_current} - Kp \times (current\ distance - reference\ distance)$$

Similarly, when the rover is moving backwards, the P control for distance can be realized by

$$PWM_{left\_previous} = PWM_{left\_current} - Kp \times (current\ distance - reference\ distance)$$
$$PWM_{right\_previous} = PWM_{right\_current} + Kp \times (current\ distance - reference\ distance)$$

3.5.3.6 Angle P Control (YANG Pei)

A. Problem Definition

Suppose a rover will trace and travel along an infinitely long straight line. The reference axis here is with a direction that indicates where the rover travels, and the rover is considered as a point (with no length or beam) assigned with a direction that indicates its heading. Also, suppose the rover can only travel forward, turn, or turn while travelling forward.

B. Analysis of Discrete Cases

We now consider the following four cases and analyse what manoeuvres the rover can perform to get back to the straight line in each case. Figure 39 shows the four cases.

Case 1: If the rover is on or near the reference axis with only a tiny mismatch in heading, then only a minor adjustment is needed to align with the axis.

Case 2: If the rover is on or near the reference axis but with a large mismatch in heading, then the rover needs immediately make a large turn before it goes off-track.

Case 3: If the rover is travelling distant from and parallel with the reference axis, a turn towards the reference axis is necessary to bring the rover back to the axis.

Case 4: If the rover is travelling away from but heading towards the reference axis, adjustments are unnecessary as the rover would eventually return to the axis.

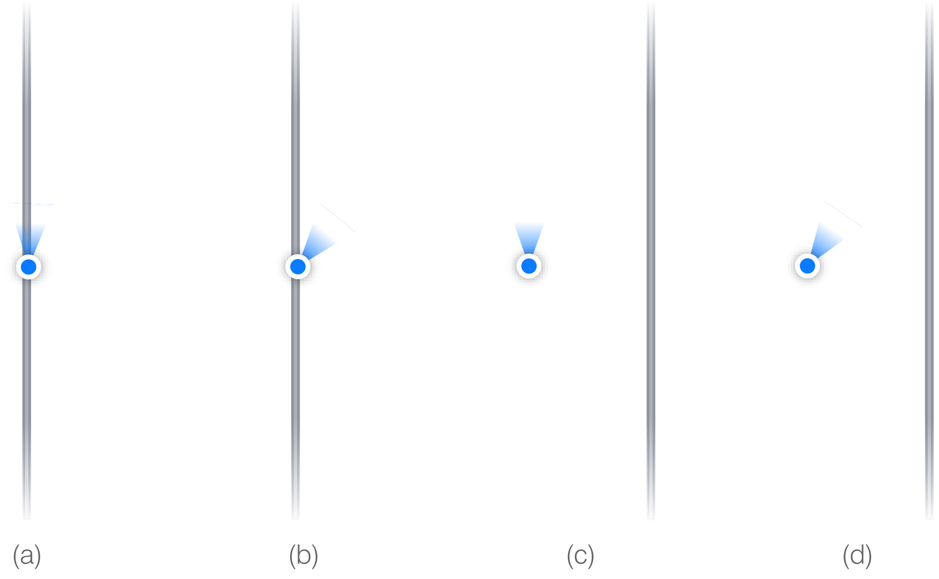


Figure 39. Analysis of discrete cases of the rover when tracing a reference axis.

Comparisons between Case 1 and 2, 3 and 4 show that whether the rover should make a turn or not is not directly related to its distance from the axis but instead related to the difference between its current heading and one desired heading, which guides the rover back to the axis.

Comparisons between Case 1 and 3, 2 and 4 imply an inversely proportional relationship between the desired heading and the rover's distance to the axis.

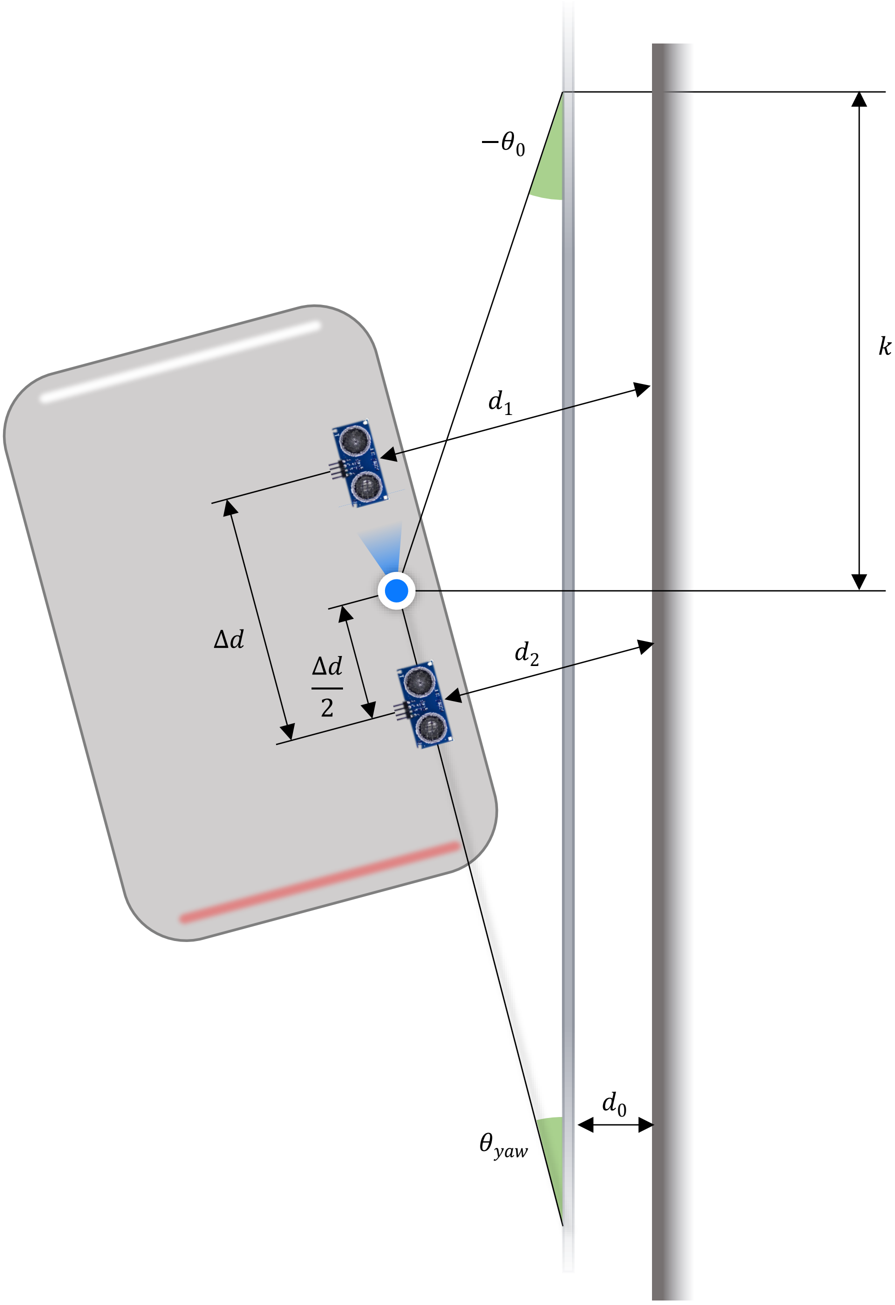


Figure 40. Geometric illustration of the physical quantities in the proposed angular control algorithm.

C. Algorithm Formulation

Suppose we wish the rover to travel along a reference axis whose distance to the lakeside railing on the right is d_0. The front ultrasonic sensor returns a distance of d_1, while the rear sensor returns a distance of d_2.

Suppose there is a difference in the heading of angle θ_{yaw} between the rover and the reference axis. Figure 40 displays the physical quantities geometrically. Hence the average distance from the rover to the reference axis is given by

$$d_{avg} = \cos\frac{d_1 + d_2}{2}.$$

These distances can also determine the mismatch of heading θ_{yaw} between the rover and the reference.

$$\theta_{yaw} = \arctan\frac{d_2 - d_1}{\Delta d}$$

If we wish the rover to return to and intersect with the reference axis at a distance of k, the yaw direction of the rover should now have an angle θ_0 with the negative direction of the reference axis. The following geometric relationship gives this angle.

$$\theta_0 = \arctan\frac{d - d_0}{k}$$

Therefore, the rover needs a turn of

$$\theta_{turn} = \theta_{yaw} - \theta_0.$$

Normalising this value by dividing π yields an angular proportional gain from -1 to 1. This would be used to control the difference in motor speed.

3.5.3.7 Ultrasonic Sensor Fence Tracking Module Performance (CUI Sijie)

A. Results

By adjusting the coefficient, the performance of the ultrasonic sensor fence tracking module is shown in Table 19 and Table 20.

Table 19. Ultrasonic sensor fence tracking module performance (moving forward).

Angle (degree)	Distance (cm)	Convergence time (s)	Convergence distance (m)
0	+10	3.5	0.7
0	+20	5.6	1.3
0	-10	1.7	0.3
0	-20	4.0	0.7
+10	0	0.9	0.2
+20	0	1.5	0.3
-10	0	1.1	0.2
-20	0	1.3	0.4

Table 20. Ultrasonic sensor fence tracking module performance (moving backwards).

Angle (degree)	Distance (cm)	Convergence time (s)	Convergence distance (m)
0	+10	3.7	0.8
0	+20	5.6	1.2
0	-10	2.0	0.5
0	-20	4.1	0.7
+10	0	1.0	0.2
+20	0	1.3	0.4
-10	0	1.0	0.2
-20	0	1.5	0.5

B. Discussion

The performance is acceptable for tracking the fence and the field test indicates that once the rover has arrived at the convergence state, the performance for tracking the fence is very stable. However, when the initial declination angle is greater than 25°, the rover cannot correct its yaw angle and the distance between the rover and the fence. And this is limited by the ultrasonic sensor's inherent characteristics (the maximal incidence angle listed in Table 18).

3.5.4 Steering System and its Performance (CUI Sijie)

Considering that the Ultrasonic sensor fence tracking module has a short convergence time and short convergence distance, there are fewer requirements for the steering system. However, the difference between steering angle and target angle should be less than 20 degrees when considering the fence tracking system limitation (the initial declination angle should be less than 25 degrees).

In the transition from task 1 to 2 and task 2 to 3, only 90 degrees steering is needed, and the accuracy for steering 90 degrees is less than ±10° which is shown in Table 14, therefore, the accuracy is acceptable.

3.5.5 System Performance (CUI Sijie)

During the tracking fence process, the rover can maintain the distance between the rover and the fence less than 20cm, which ensures the stability of the transition between tasks. During 10 tests, the rover successfully navigated from task 1 to task 2, and from task 2 to task 3 for 9 times. The rover failed to transit tasks once. This was due to the unevenness of the cobblestone road that caused the trolley to tilt, so that the ultrasonic sensor detects the wrong distance, causing the car to fail to transit to mission 2.

3.6 Basket Docking and Ball Releasing System

3.6.1 Objectives (CUI Sijie and YUE Haiyi)

At fence 5, the rover needs to dock with the basket and then release the ball to the basket (Figure 41).

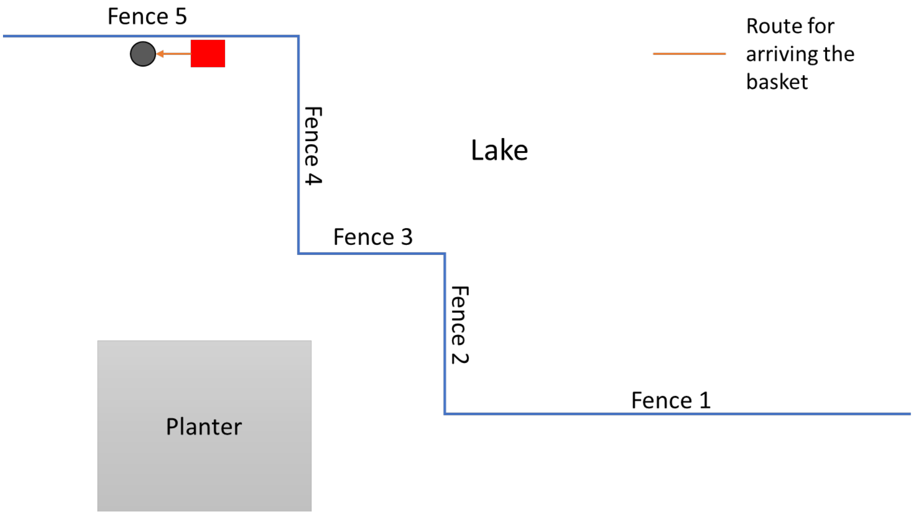


Figure 41. The route for docking the basket.

3.6.2 Working Mechanism (CUI Sijie)

The working mechanism of the basket docking and ball releasing system (along the orange line) is shown in Figure 42. Two judgements would ensure the rover successfully dock with the rover and release the ball.

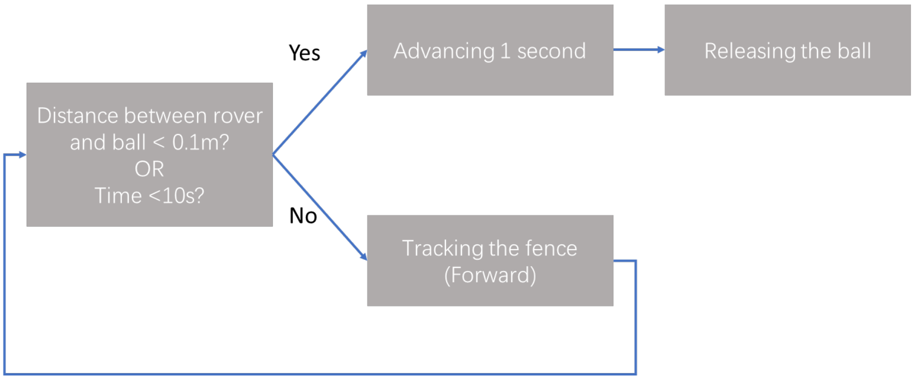

Distance between rover and ball < 0.1m?
OR
Time <10s?
Yes
Advancing 1 second
Releasing the ball
No
Tracking the fence (Forward)

Figure 42. The working mechanism of basket docking and ball releasing system.

3.6.3 Basket Docking System (CUI Sijie)

3.6.3.1 Docking Module

The docking module (the curved design on the front of the rover, Figure 43) will ensure that the basket will eventually successfully dock to the centerline of the rover, even if it is not aligned with the centerline at the beginning. The acceptable deviation between the rover centerline and the basket is ±7cm (Figure 43), so as long as the error of the ultrasonic tracking module is less than ±7cm (in fact, in 3.5.3, our ultrasonic tracking module's error is less than ±3.5cm), the rover can be docked with the basket.

In addition, the rover can push the basket forward for more than 20m without falling down and without deviating from the centerline of the rover (in fact, the distance should be much greater than 20m. In our field test, due to the size of the field, we only measured the case for 20m) and this will provide feasibility for our following working mechanism.

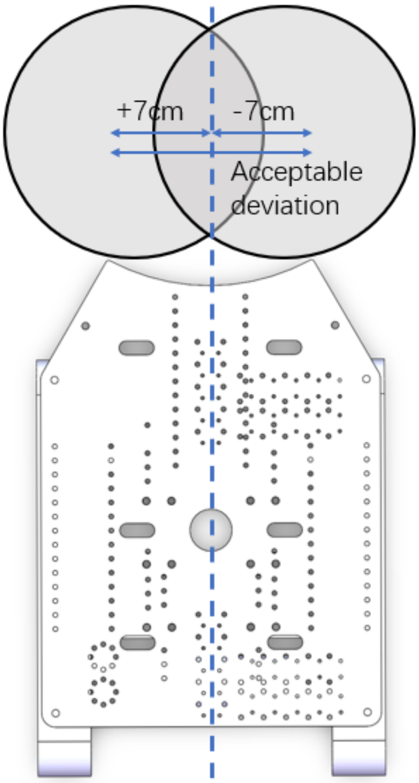


Figure 43. The docking module and acceptable deviation visualization

3.6.3.2 Basket Detecting Module

A. Distance (between the rover and basket) detection module

The distance between the rover and the ball is obtained by using the ultrasonic sensor 1 (Figure 36). The ultrasonic sensor's performance for detecting the basket is given in Table 21.

Table 21. Ultrasonic sensor's performance for detecting the basket.

Test items	Performance
Maximal detecting distance	15cm
Minimal detecting distance	7cm
Detecting rate	56%

The detecting rate is 56%, indicating that the system is not stable by only using the ultrasonic module to detect the distance, therefore, the timer is used to be an auxiliary plan for docking the basket.
In addition, considering that the ultrasonic sensor detects the basket at a detection distance of 7cm~15cm, to ensure that the rover is in full contact with the basket, the rover will continue to move forward for 1 second (that is, about 30cm) to fully contact with the basket.

B. Auxiliary timer module

Given that the rover can push the basket forward more than 20m without the basket falling off or deviating from the centerline of the rover, a timer can be used. Because the distance between the basket and the fence 4 is fixed (less than 4m), therefore, the timer is set to 10s (corresponding to a movement of 5m). If the ultrasonic sensor has not detected the basket after 10 seconds, the system will automatically consider that the rover has docked with the basket successfully and begin the ball releasing phase.

3.6.3.3 Ultrasonic Sensor Fence Tracking Module

This module utilizes the same module introduced in 3.5.3, and the module performance is given in 3.5.3.7.

3.6.3.4 Basket Docking System Performance

50 trials had been made to verify the validity of this system and the rover docked successfully with the rover for all the 50 trials.

3.6.4 Ball Releasing Module (YUE Haiyi)

3.6.4.1 Structure and Components

Once the rover has reached the ball-releasing site, the ball would be pitched into the basket from the rover immediately. In this process, a pitching device is needed. To construct a ball-releasing device, a 26.5 cm long iron bar, a 15 cm long chopstick, a piece of thread, and a piece of the tiny board were needed. A servo connected with the mainboard was necessary to control the movement of the ball. The schematic of the pitching system is shown below in Figure 44.

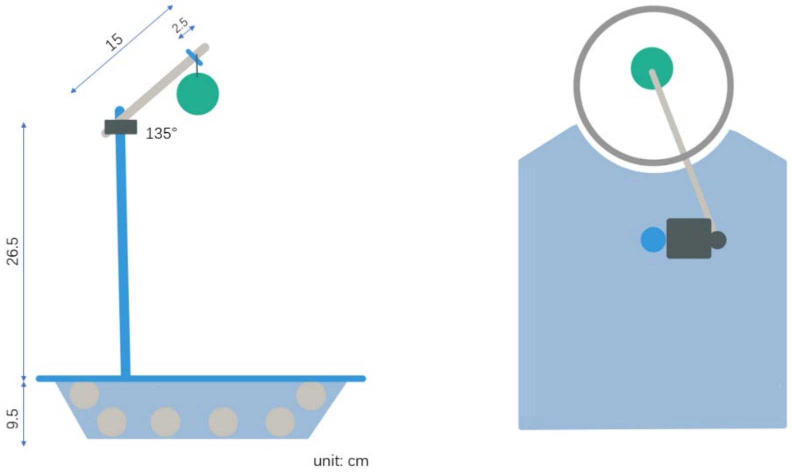


Figure 44. Schematic of the pitching device.

3.6.4.2 Working Mechanism

The chopstick would be perpendicular to the surface when there is no load in the end. Once the ball was attached to the chopstick by a piece of thread, the angle between the iron bar and the chopstick would be 135°. Based on the basket docking system, when the rover has located the basket and the servo receives a command of rotation, it would drive the chopstick to turn over, thus the ball would be forced to roll and fall into the basket.

3.6.4.3 Results

50 trials had been made to verify the validity of this system. The ball could fall into the basket successfully in 49 trials, the only failure was owing to human error. The average drop point is 18.2 cm from the edge of the rover frame.

3.6.5 Basket Docking and Ball Releasing System Performance (CUI Sijie and YUE Haiyi)

The system could help the rover successfully dock with the basket and then release the ball into the basket and the stability is very high because in only 1 trial the ball has failed to be released into the basket due to human error over 50 trials.

3.7 Sweeping System (FENG Zihao)

3.7.1 Task Analysis

In task 1 of Patio 2, according to the rules, the rover should move to the same shape after recognizing the shape at the beginning. When a triangle is determined initially, the rover should move straight to follow the distal triangle based on the visual navigation system.

In the field test, when the rover is programmed to go straight directly after recognizing the triangle, the standing sign will influence both the dynamic system and the visual navigation system. As Figure 45 shows, if the standing sign is knocked out, it is likely to be run over by the track on one side, causing the whole body to tilt. Moreover, the standing sign in front of the lens can cause the visual system to work abnormally as the vision system cannot see and follow the shape at the far end. Therefore, a sweeping system is required to sweep the standing sign to the sides after the triangle is recognized. Moreover, since the bracket does not influence other tasks, it will be set up close to the body, which means a specified angle of the servo motor should be selected and remained during the rest of the tasks.

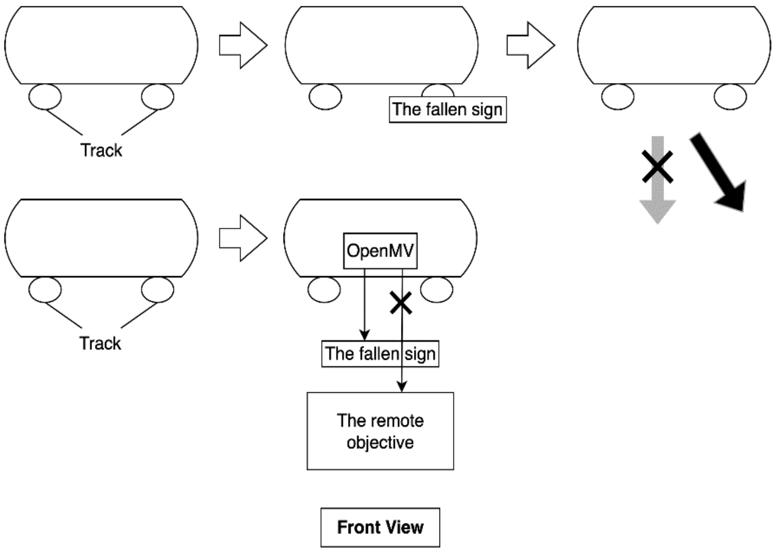


Figure 45. The potential problems caused by the standing sign.

3.7.2 Working Principle

As Figure 46 shows, the sweeping system consists of a servo motor and a bracket. The servo motor is placed under the support plate, and it is connected to a bracket through a rotating shaft.

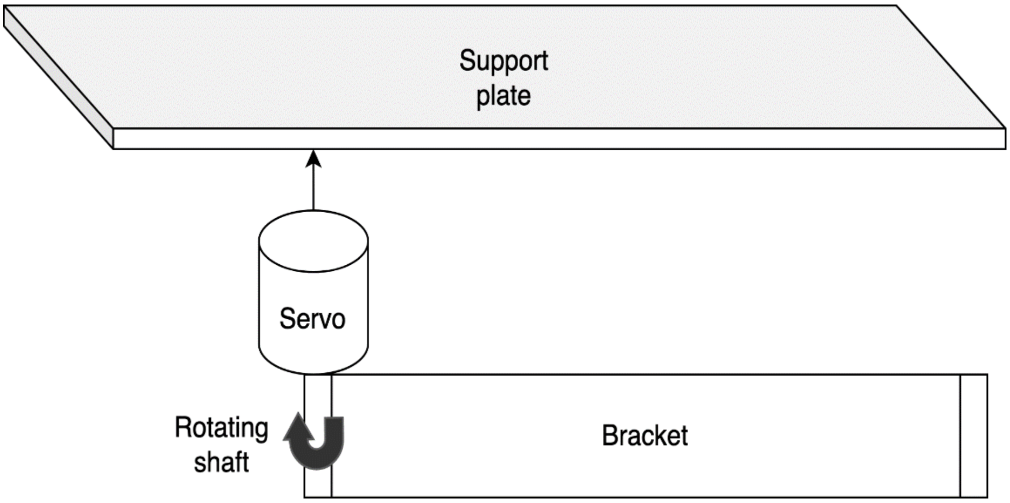


Figure 46. The structure diagram of the sweeping system.

After the triangle is recognized, the sweeping system will start up. The rotating shaft of the servo drives the bracket back and forth in the direction of the arrow for some time, sweeping the standing signs away.

3.7.3 Implementation

The sweeping system is controlled by the OpenMV board, which uses MicroPython. Therefore, the Servo library can be imported directly to adjust the rotation of the rotating shaft. During the sweeping task, the servo motor is set to rotate to another angle, and then wait for some time. After that, it will turn back to the starting angle. The whole process will be repeated a certain number of times. Besides, as the rotation speed of the servo motor cannot be controlled, the angle of rotation, the waiting time and the number of rotations is continuously adjusted to achieve maximum sweeping efficiency.

3.7.4 Result

In the field test, once certain shapes are recognized and the sweeping system is turned on, the standing sign will be swept away and the rover will move to the corresponding sign without any influence, as Figure 47 shows.

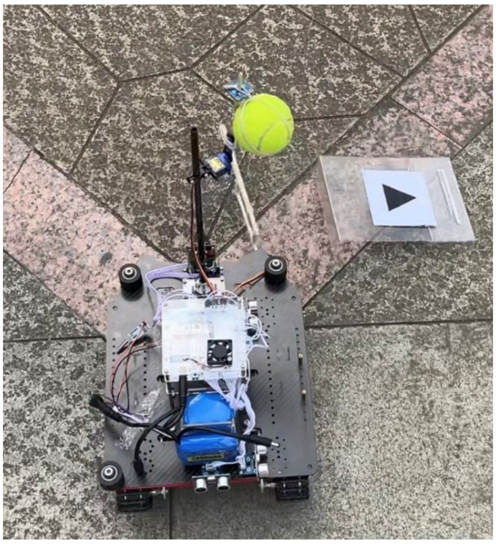


Figure 47. The operation result of the sweeping system.

3.8 Wireless Communication Task (HE Renjie)

3.8.1 Problem Analysis

In patio 2 task 3, the rover is required to stop at a specified area in the planter area and transmit a message to a laptop. It should then continue to move after acknowledging the transmission. In this task, our major work is to design the interface between HC-12 and the robot and laptop, rather than the specific data protocol and transmission channel since the task requests the implementation to use HC-12 wireless transceiver at 433MHz, which supports transparent transmission. Therefore, based on the task requirement, this task can be divided into three parts:

1. An interface called by the main board to complete the transmission
2. A platform on the laptop for sending and receiving messages, pattern selection, and information display
3. The acquisition of the time information

3.8.2 Design Specification

A. System Schematic

The system schematic for the final design is shown in Figure 48. The modules in the figure are designed with two sets of solutions. The final solution for the communication task is shown in the figure, along with a list of the remaining unselected solutions. The specific design, analysis, and solution selection are described next.

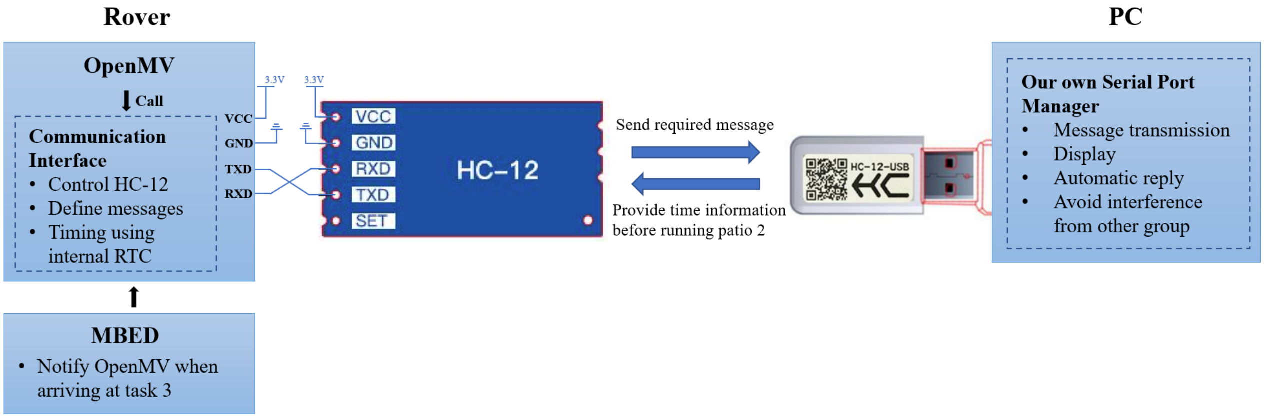


Figure 48. Communication system schematic.

B. Transmission Sequence Diagram

The design of the sequence diagram for wireless transmission is shown in Figure 49.The transmission is divided into four phases. The test phase is to transmit a simple message to ensure the module is working. The calibration phases send a time offset to the rover. These two steps are done before officially beginning patio 2. Then task 3 is carried out after waiting for other tasks to complete.

The technical details of the system schematic and the transmission sequence diagram are described and discussed in the following contents.

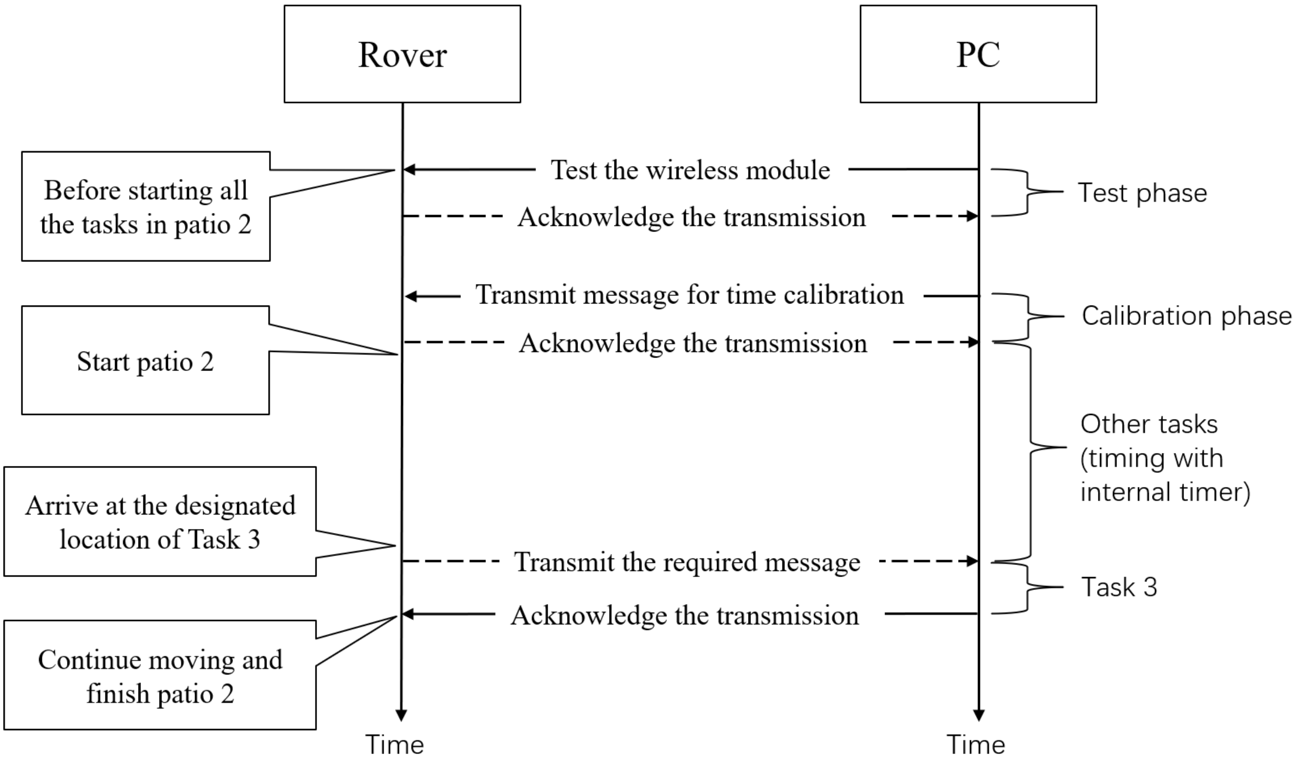

Rover
PC
Before starting all the tasks in patio 2
Test the wireless module
Acknowledge the transmission
Test phase
Start patio 2
Transmit message for time calibration
Acknowledge the transmission
Calibration phase
Other tasks (timing with internal timer)
Arrive at the designated location of Task 3
Transmit the required message
Task 3
Acknowledge the transmission
Continue moving and finish patio 2
Time
Time

Figure 49. Transmission sequence diagram.

3.8.3 Rover-side Communication Interface

HC-12 module can communicate with other systems by simply using UART according to the datasheet [8]. Thus, the rover-side interface is a simple callable function that takes message strings as the input and returns the acknowledgement state. If the rover receives acknowledgement of the transmission, this function returns a value to tell the rover to move on.

A. Use of OpenMV as the control board of HC-12

During the early development phase, MBED was used as the control board for HC-12. However, during the overall test of the whole rover, only one of NUCLEO-L432KC's three UARTs was found to be functional. Through an extensive collection of data, this result was confirmed by the official MBED pin mapping definition. This leads to an insufficient number of pins on MBED. Therefore, we use OpenMV as the control board of HC-12.

B. HC-12 Mode Configuration

HC-12 has four different transparent transmission modes and 127 communication channels which have to be configured according to our requirements. The default CH001(433.4MHz) channel is used according to the task requirement. As for the transmission mode, the FU3 mode is chosen, because the message to be sent has a length of 161 bytes, which exceeds the applicable data volume of FU2 and FU4 modes. FU1 is more power-saving. But it is chosen to work at full speed as their idle current does not differ much.

3.8.4 PC-Side Wireless Transmission Platform

A. Hardware Interface

On the PC side, a hardware interface would be needed between HC-12 and the laptop. Based on preliminary investigations, two options were found: CH340 which supports USB-to-TTL conversion, and HC-T test frame which provides an HC-12-to-USB interface. HC-T is chosen for the final design as it is specifically designed for HC-12 and is easier to use. It is shown in the following figure:

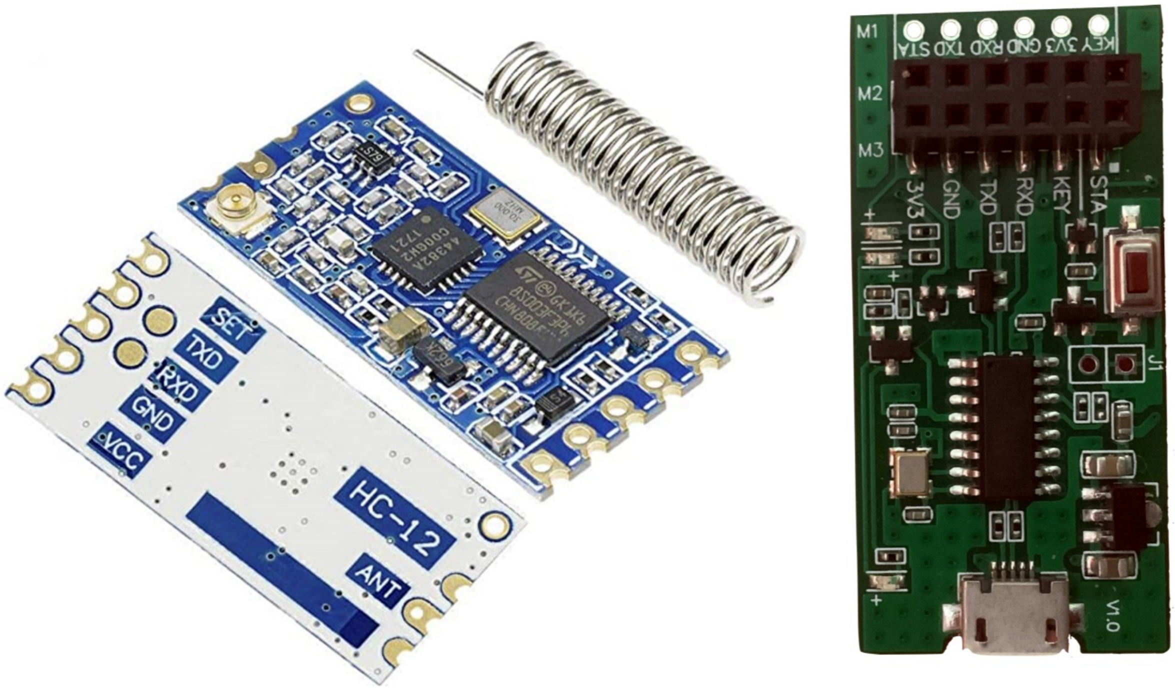


Figure 50. (left) HC-12; (right) HC-T which is used to provide an HC-12-to-USB interface.

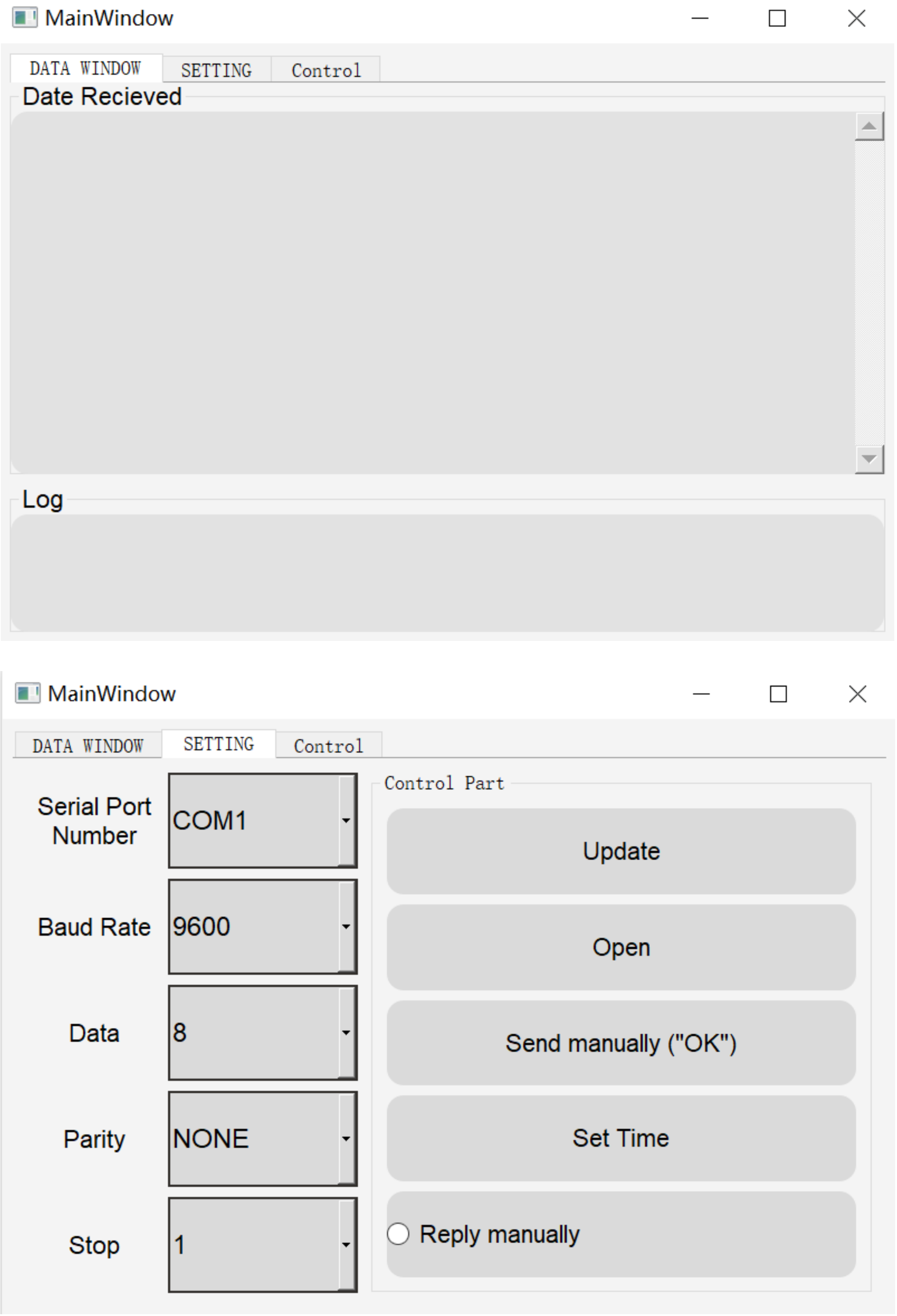
B. Development of Our Serial Port Manager

It should be noted that a serial port manager would be needed, and should consider the following problems:

- Monitor the port, transmit or receive data and display it on the screen of the PC.
- Automatically send an acknowledgement to the rover after receiving the correct message.
- Congested channel and Interference.

It should be noted that 433MHz is a free band that does not require a license, hence it is used in many common products. Especially considering that the patio is likely to have interference from other groups, and thus the band is congested and has a lot of interference. It would be good if this interference can be filtered by the platform.

Though there are many open-source or free serial port managers for PC, they cannot handle the second and the third problem listed above, and thus do not satisfy our needs here. Therefore, our version serial port was developed according to the requirements above. It is written in Qt Designer using C++ and has a user interface as shown in the following:



Serial Port Number COM1

Baud Rate 9600

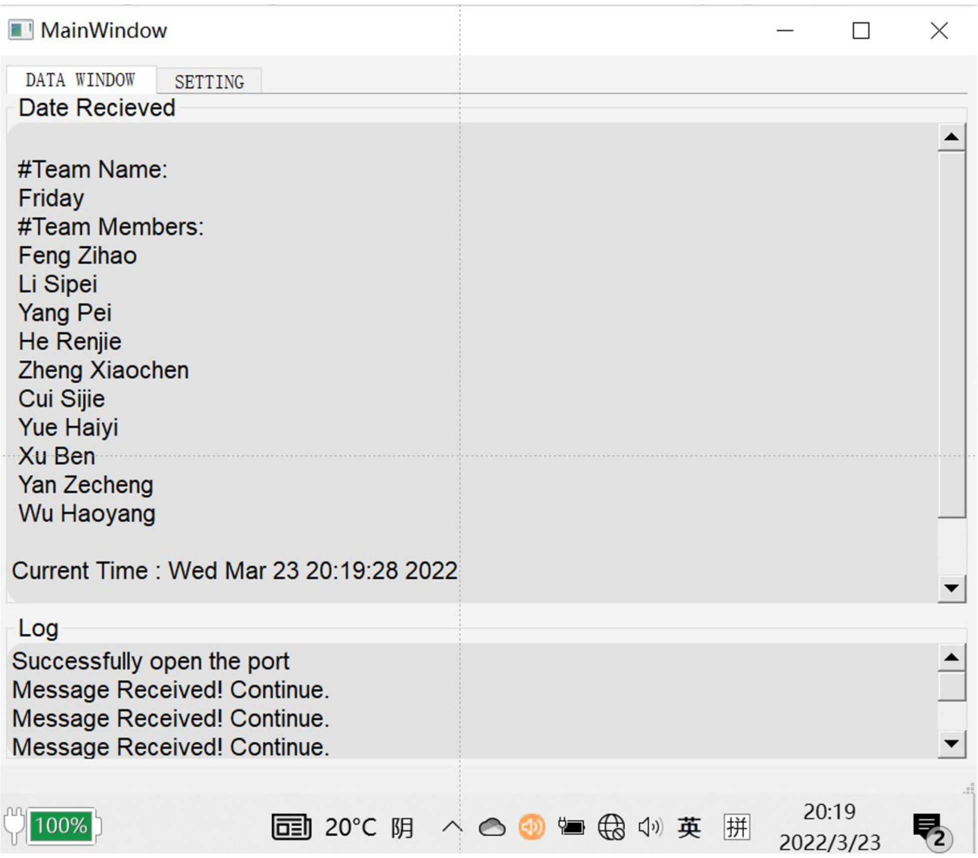
Data 8

Parity NONE

Stop 1

Figure 51. Our own version of serial port manager for wireless transmission between PC and the rover. (upper) displaying page; (lower) controlling page.

In the "Data Window", our received messages are shown in the "Data Received" zone. Other control messages such as acknowledgement are shown in the "Log" zone. The "Setting Window" contains all the tabs to configure the serial port and control components of the “Data Window”. A successful transmission example is shown in the following figure:



Date Recieved

```
#Team Name:
Friday
#Team Members:
Feng Zihao
Li Sipei
Yang Pei
He Renjie
Zheng Xiaochen
Cui Sijie
Yue Haiyi
Xu Ben
Yan Zecheng
Wu Haoyang

Current Time : Wed Mar 23 20:19:28 2022
```

Log

```
Successfully open the port
Message Received! Continue.
Message Received! Continue.
Message Received! Continue.
```

20:19
2022/3/23

Figure 52. An example of a successful transmission.

Another concern here is how to minimize the risks of being interfered with by other teams’ messages. This is worth doing because during the field test we found that interfering messages from other groups caused the messages to be mixed. Thus, the platform is programmed to examine the received message and search for a predefined pattern so that other messages which do not contain the pattern will be filtered.

An experiment was designed and carried out to determine how HC-12 transmits strings. The details of this experiment are recorded in the notebook. The results show that HC-12 sends 17 bytes at one time. A prefix which is a short string that is unlikely to be sent by others is added to each message. Whether messages have these prefixes will determine whether they are filtered. In our design, the prefix is chosen to be “Friday”.

3.8.5 The Acquisition of Time Information

The message sent to the PC is required to include time information. Two plans are raised for the acquisition of time information:

- Plan A is to add an external real-time clock (RTC) module to store the calibrated time information and do the timing.

- Plan B is to calibrate the time before the rover begins patio 2 tasks and use the internal RTC of the main board to calculate the period.

The advantage of plan A is that the external RTC has a backup battery, so we only need to calibrate the time once and it will be stored even if the power of the rover is cut off. Its disadvantage is it will take three I/O ports on the MBED which could be used by other modules. Plan B saves us I/O ports but we need to calibrate the time before every trial of patio 2 tasks. Considering factors such as reserving MBED ports for other more important modules and saving budget, plan B is adopted.

For Plan B, the time to be sent in patio 2 task 3 can be written as:

$$T_{sent} = T_{cali} + (T_{internal2} - T_{internal1})$$

, where T_{cali} is the time information we send before starting patio 2 tasks. $T_{internal1}$ is the time recorded using the internal RTC of the controller at the beginning of patio 2. $T_{internal2}$ is recorded using the internal RTC at the beginning of patio 2 task3. Before sending, T_{sent} is converted from total seconds to standard calendar time.

3.8.6 Results and Discussion

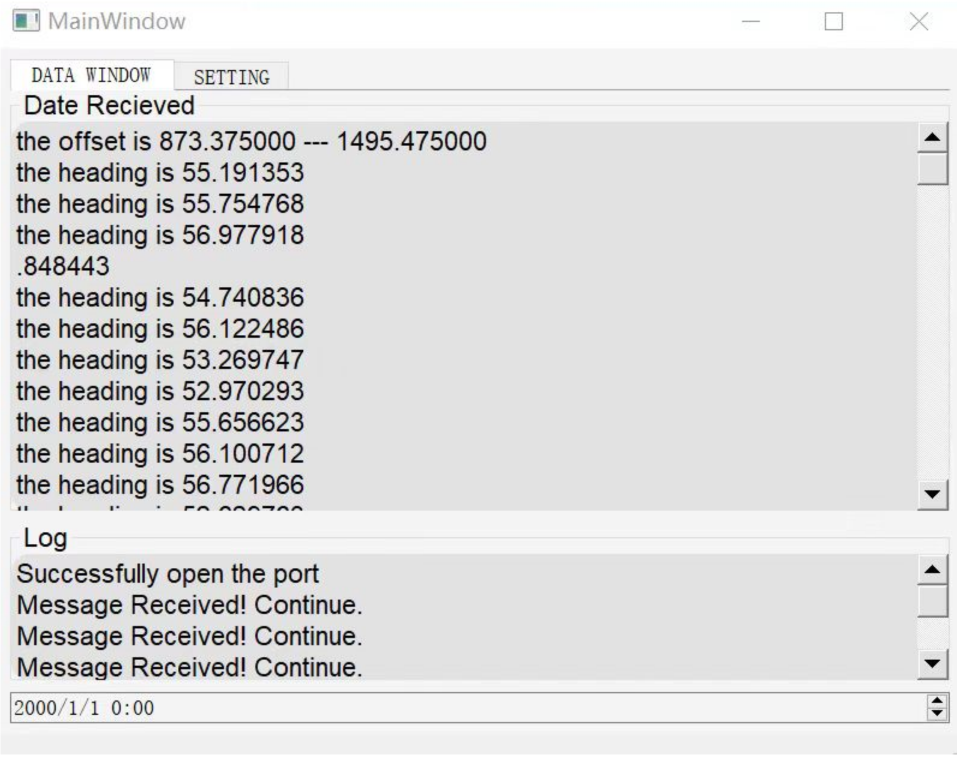
A. Results

Over 50 field tests have been carried out to test the performance of the rover in patio 2. The wireless transmission task of the final design works steadily and can complete tasks successfully. The transmission contents and acknowledgement are transmitted properly, information from other groups in the same channel can be filtered by our serial port assistant, and the time information is accurate.

B. Discussion

During the designing phase, every sub-task in the wireless communication task is brought with two plans. This greatly facilitates the flexibility to adjust strategies and make technical judgments and modifications. The problem encountered is that sometimes the DuPont wire connection between HC-12 and the PCB board is not stable and causes transmission failure. To tackle this problem, a hot melt adhesive is used to securely hold the connection between the communication module and the MBED.

Besides, it is worth mentioning that this platform could not only complete task 3, but also help us to monitor certain parameters when the rover is running in a wireless approach. For example, we can transmit the parameters we care about to the PC via HC-12 and then use this platform to detect the rover status. An example of using this platform to debug other modules is shown in Figure 53. This provides wireless monitoring for other modules, which is very useful and can facilitate our debugging.



```
the offset is 873.375000 --- 1495.475000
the heading is 55.191353
the heading is 55.754768
the heading is 56.977918
.848443
the heading is 54.740836
the heading is 56.122486
the heading is 53.269747
the heading is 52.970293
the heading is 55.656623
the heading is 56.100712
the heading is 56.771966
```

```
Successfully open the port
Message Received! Continue.
Message Received! Continue.
Message Received! Continue.
```

2000/1/1 0:00

Figure 53. An example of using our wireless transmission platform to monitor the parameters of the compass module.

3.9 Inter-Board Communication (FENG Zihao)

3.9.1 Problem Analysis

To achieve high driver compatibility and guarantee an adequate number of interfaces (Pins), in addition to the OpenMV board, our group used an STM32L432KC board to control mechanical devices such as the power system. Therefore, efficient and reliable communication between the OpenMV board controlling the vision system and the stm32 board controlling the remaining systems is essential. Among the widely used inter-board communication protocols such as UART, I2C, and SPI, we choose UART (universal asynchronous receiver-transmitter) as our communication protocol for its ease of operation, highly detailed document and no clock needed. Moreover, both the OpenMV board and STM32L432KC initially support UART protocol, so we do not need to set up additional drivers.

3.9.2 Working Principle

3.9.2.1 Interface

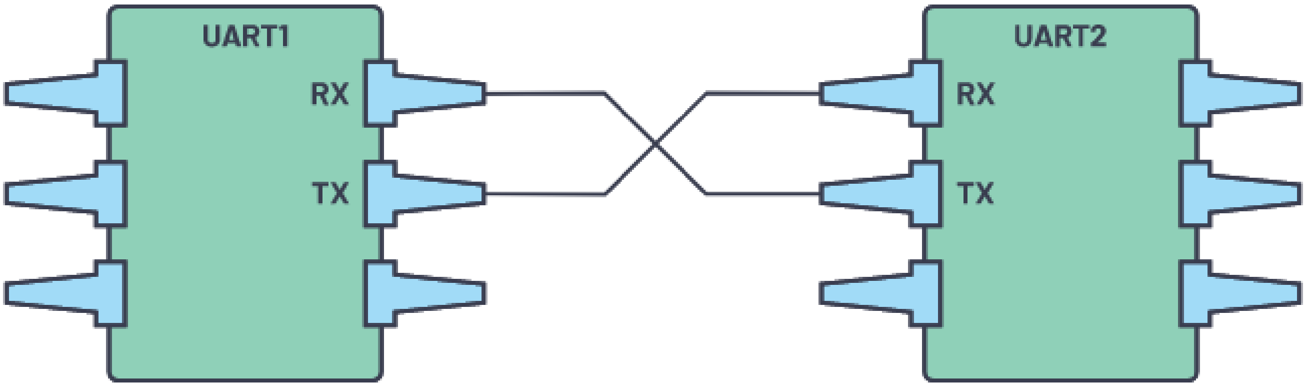


Figure 54. The UART interface.

As Figure 54 shows, each UART device has two signals, Transmitter(TX) and Receiver(RX), used to transmit and receive data intended for serial communication, respectively. The UART interface delivers data asynchronously and does not require a clock signal to synchronize the transmitter and receiver devices. The transmitter generates a bitstream based on its clock signal. At the same time, the receiver samples the incoming data using its internal clock signal. Having the same baud rate on both boards manages the point of synchronization [9].

3.9.2.2 Data Transmission

The manner of transmission in UART is in the form of a packet. The component that links the transmitter and receiver generates serial packets and manages the actual hardware lines. The start bit, data frame, parity bit, and stop bits make up a packet.

A. Start Bit

When there is no data transfer, the UART transmission line is retained at high voltage. The voltage is pushed from high to low for one clock cycle to start the data transmission. This transition serves as the beginning. When the receiving UART detects a high to low voltage change, it starts reading the data frame at the baud rate's frequency.

B. Data Frame

The data frame contains the data being transferred, which is commonly 5 to 8 bits long.

C. Parity

The receiving UART uses the parity bit to determine if any data has changed during transmission. Electromagnetic radiation, mismatched baud rates, and long-distance data transmissions can all alter bits.

After reading the data frame, the receiving UART counts the number of bits with a value of 1 and determines whether the total is even or odd. If the parity bit is 0 (even parity), the data frame's 1 or logic-high bit should add up to an even amount and if the parity bit is 1 (odd parity), the data frame's 1 bit or logic highs should add up to an odd number, which may be used to evaluate whether the data received is incorrect.

D. Stop Bits

The transmitting UART drives the data transmission line from a low voltage to a high voltage for 1 to 2 bit(s) to signify the end of the data packet.

3.9.3 Implementation and Test

As both the OpenMV boards and STM32L432KC support UART protocol, UART function can be directly applied. Before the transmission start, several parameters should be set to be the same. The most significant one is the baud rate, at which information is transferred in a communication channel. In our rover, the baud rate of the two boards is set to 9600. In addition, the number of bits per character is set to be 8. Besides, as inter-board communication is achieved by wired transmission, it is not essential to apply the parity bit.

After the preset process, the next step is to choose the UART pins according to the document. In our group, (TX, RX), (P4, P5) for the OpenMV board and (D1, D0) for STM32L432KC are chosen for UART transmission respectively asFigure 55 shows.

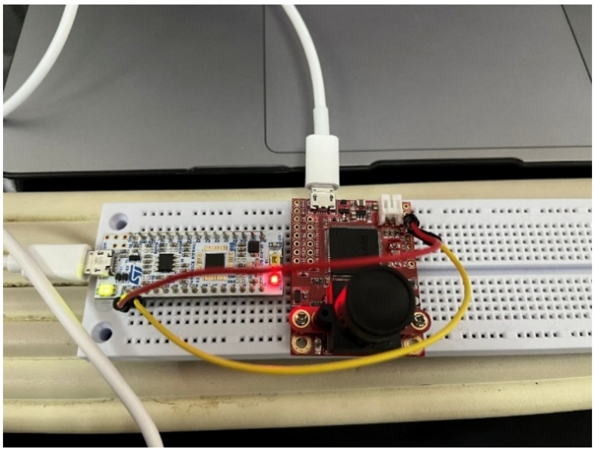
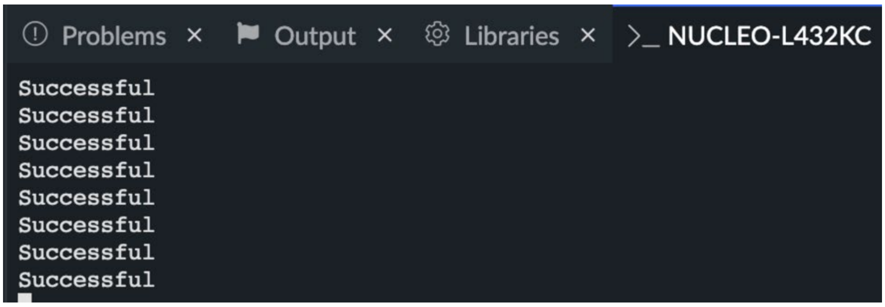


Figure 55. The hardware connection image of the two boards.

To test the stability of UART transmission, STM32L432KC is connected to a computer through the micro USB port. As Figure 56 shows, a test program is set to print a message "Successful" to the computer terminal after STM32L432KC transmits signals to the OpenMV board and the OpenMV board transmits signals to STM32L432KC back, as this condition can simulate the signal transmission during the actual operation as much as possible.



```
Successful
Successful
Successful
Successful
Successful
Successful
Successful
Successful
```

Figure 56. The message printed to the terminal.

3.9.4 Results and Discussion

When testing UART transmitting on a breadboard, the stability of direct data transmitting is unsatisfactory. Therefore, a channel coding algorithm to optimize transmission reliability is required. As Table 22 shows, by increasing the redundancy of information, the stability of transmission is significantly improved.

Table 22. Comparison of the results of direct transmitting and using channel coding.

	Direct Transmitting	Using Channel Coding
Success Rate	39.5%	80%

When the wires are integrated into the PCB, it is surprising that the success rate of the transmission is increased to 100%, which means that the channel coding algorithm is no longer needed, as it would affect the transmission efficiency.

3.10 Circuit

Since the rover contains several modules working as a whole, a motherboard is designed for inter-board connection.

3.10.1 Circuit Connection Guidelines

3.10.1.1MBED Connection (CUI Sijie)

The following components are connected to the MBED.

- 4 ultrasonic sensors
- 2 motor drivers
- 2 motor encoders
- OpenMV

These components connect with MBED based on the following guidelines (Table 23).

Table 23. Guideline for MBED connection.

Pin	Type of pin	Pin on the MBED
Ultrasonic sensor 1, 2, 3, 4 trigger pin	Digital out	PB_3
Ultrasonic sensor 1 echo pin	Digital in	PA_1
Ultrasonic sensor 2 echo pin	Digital in	PA_0
Ultrasonic sensor 3 echo pin	Digital in	PA_3
Ultrasonic sensor 4 echo pin	Digital in	PA_4
Right motor driver pin 1	PWM out	PA_11
Right motor driver pin 2	PWM out	PA_8
Left motor driver pin 1	PWM out	PB_6
Left motor driver pin 2	PWM out	PB_0
Right motor encoder phase A	Digital in	PB_5
Right motor encoder phase B	Digital in	PA_12
Communication between MBED and OpenMV TX	UART	PA_9
Communication between MBED and OpenMV RX	UART	PA_10

3.10.1.2OpenMV Connection (CUI Sijie)

The following components are connected to the MBED

- Bluetooth module
- Servo

These components connection with OpenMV based on the following guidelines (Table 24).

Table 24. Guideline for OpenMV connection.

Pin	Type of pin	Pin on the MBED
Communication between MBED and OpenMV TX	Digital out	P4
Communication between MBED and OpenMV TX	Digital in	P5
Servo	PWM out	P8
Bluetooth Module TX	UART	P0
Bluetooth Module RX	UART	P1

3.10.1.3OpenMV Connection (CUI Sijie and ZHENG Xiaochen)

The following are specifications collated.

- DC powering plug Φ5.5mm/2.5mm
- 12V battery input
- 12V, 5V, 3.3V power supply
- Blown fuse
- LED
- Power switch

- Reset switch
- Voltmeter (Connected to 12V)
- 5V Mbed powering
- 5V OpenMV powering
- 5V fan powering
- 2 motor connections (6 pins, 12V)
- 4 ultrasonic connections (4 pins, 5V)
- 1 OpenMV connection (2 pins)
- 1 Bluetooth connection (5 pins, 3.3V)
- 1 RTC connection (5 pins, 3.3V)
- 1 compass connection (5 pins, 3.3V)
- 2 servo connection (4 pins)
- Size: less than 100mm x 100mm
- Tapped holes at the corner
- Heat sink
- Grounding

3.10.2 Circuit Schematic Design (ZHENG Xiaochen)

The circuit board was divided into four parts: power and buck, motor driver, Mbed peripherals, and OpenMV.

3.10.2.1Power and Buck

Through previous measurements of the maximum load current of the power supply, a 3A DC-DC bulk converter TD1587, a voltage regulator to convert 5V to 3.3V for low-power devices' powering was chosen. The module is AMS1117-3.3, a popular and reliable solution for 5V-to-3.3V conversion.

A. TD1583

The TD1583 is a 380 KHz fixed frequency monolithic step-down switch mode regulator with a built-in internal Power MOSFET. It achieves a 3A continuous output current over a wide input supply range with excellent load and line regulation [10]. Pin description of the device is shown below.

Table 25. TD1587 pin description [10].

Pin Number	Pin Name	Description
1, 6, 8	NC	Not connect.
2	Vin	Supply voltage input pin. Support 3.6V to 28V DC voltage. Need a large decoupling capacitor to eliminate noise.
3	SW	Power switch output pin.
4	GND	Ground pin.
5	FB	Feedback pin. Through an external resistor divider network, FB senses the output voltage and regulates it. The feedback threshold voltage is 1.222V.

Pin Number	Pin Name	Description
7	EN	Enable pin. High to turn on, low to turn off.

A typical application circuit for 5V/3A output given by the device’s datasheet is shown below.

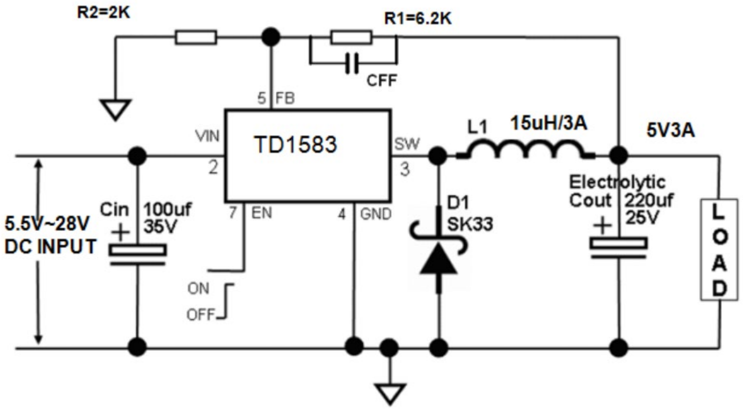


Figure 57. TD1583 Typical Application Circuit @ 5V/3A [10].

From the datasheet, we got the following information:

- The feedforward capacitor CFF is unnecessary for output voltage less than 8V applications.
- Output voltage: $V_{OUT} = 1.222 \times (R1 + R2)/R2$
- Inductance value: $L = (V_{OUT}) \times (V_{IN} - V_{OUT})/V_{IN} \times f \times \Delta I$, where ΔI the peak-to-peak ripple current in the inductor to be approximately 30% of the maximum load current.
- Input capacitor: a smaller high-quality ceramic 0.1μF capacitor placed closer to the IC and a larger tantalum or electrolytic capacitor placed further away.
- Output capacitor affects the output voltage ripple. The ripple is allergic to ESR. For ceramic capacitors,$V_{RIPPLE} \cong 1.4 \times VIN \times (f_{LC}/f_{SW})^2$ while for tantalum or low-ESR electrolytic capacitors, $V_{RIPPLE} \cong \Delta I \times R_{ESR}$.

Concerning the above requirements, we determine the passive devices’ parameters as shown in Table 26.

Table 26. Component selection.

R1	6.2K
R2	2K
L1	15μH/3A
D1	SK33
Cin	100μF ceramic // 220μF electrolytic
Cout	100μF ceramic // 220μF electrolytic

B. AMS1117-3.3

The AMS1117-3.3 is a fixed regulator to 3.3V that is easy to use and is protected against short circuits and thermal overloads. It can be used for 5V-to-3.300V step-down and line voltage regulation to a typical value of 0.5mV under the condition of $1.5V \leq (V_{IN} - V_{OUT}) \leq 12V$ [11].

A typical application circuit is given below.

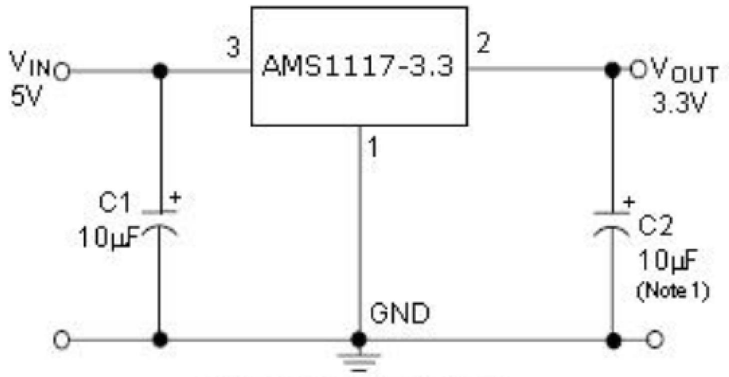


Figure 58. AMS1117-3.3 Typical Application Circuit [11].

We connected our circuit as above. The passive devices' parameters are determined as follows:

Table 27. Component selection.

C1	10uF
C2	10uF
C3 (output filter capacitor)	100uF

3.10.2.2Motor Driver – A4950

Designed for PWM control of DC motors, the A4950 is capable of peak output currents to ±3.5 A and operating voltages to 40 V, with input terminals provided for use in controlling the speed and direction of a DC motor with externally applied PWM control signals [12]. The A4950 act as an exciter for the motor's control signal from the Mbed, and provides sufficient current to keep the motor rotating powerfully.

There are eight I/O ports on the chip. Following is the pin description.

Table 28. A4950 terminal list table [12].

Pin Number	Pin Name	Description
1	GND	Ground
2	IN2	Logic input 2
3	IN1	Logic input 1
4	VREF	Analog input
5	VBB	Load supply voltage

Pin Number	Pin Name	Description
6	OUT1	DMOS full-bridge output 1
7	LSS	Sense signal connection
8	OUT2	DMOS full-bridge output 2
9	PAD	Thermal dissipation pad connected to GND

For passive device selection, from the datasheet, we got the following information:

- The maximum value of current limiting is set by the selection of small-value sensor resistor RS and the voltage at the VREF pin. The transconductance function is set by ITripMAX = VREF/10 * RS. The maximum voltage on the LSS pin should be ±500 mV at maximum load.
- The load supply pin, VBB, should be decoupled with an electrolytic capacitor (typically 100 uF) in parallel with a lower-valued ceramic capacitor placed as close as practicable to the device.

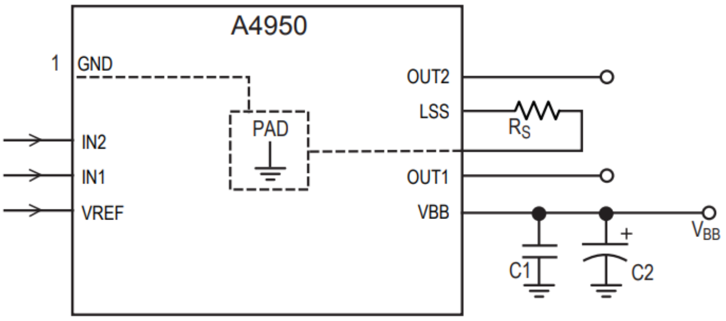


Figure 59. A4950 Application Circuit [12].

According to the datasheet, the passive devices' parameters are determined as follows:

Table 29. Component selection.

RS	250mΩ	2512, 1 W, 1% or better, carbon film chip resistor
C1	0.22uF	X5R minimum, 50 V or greater
C2	100uF	Electrolytic, 50 V or greater

3.10.2.3Mbed Peripherals

The Mbed peripherals are components and boards connected with the Mbed such as switches, motors, ultrasonics, Bluetooth, RTC, and compass. The Mbed is connected to the board via pin headers. While the peripherals are connected to the board via wire-to-board connectors. The interconnections are made on the motherboard between the corresponding pads following the hierarchy shown in Table 23. The following table discusses the specific requirements for connections.

Connection Type	Requirements
Reset switch	10k pull-up resistor and 100uF decoupling capacitor for hardware denouncement

Connection Type	Requirements
Power to ultrasonic	100uF filter capacitor from 5V to ground

3.10.2.4OpenMV

The OpenMV demo board is also connected to the motherboard via a wire-to-board connector. Two separate connectors are used for powering and inter-board communication.

3.10.2.5Final Circuit Schematic

The schematic was drawn using LCEDA.

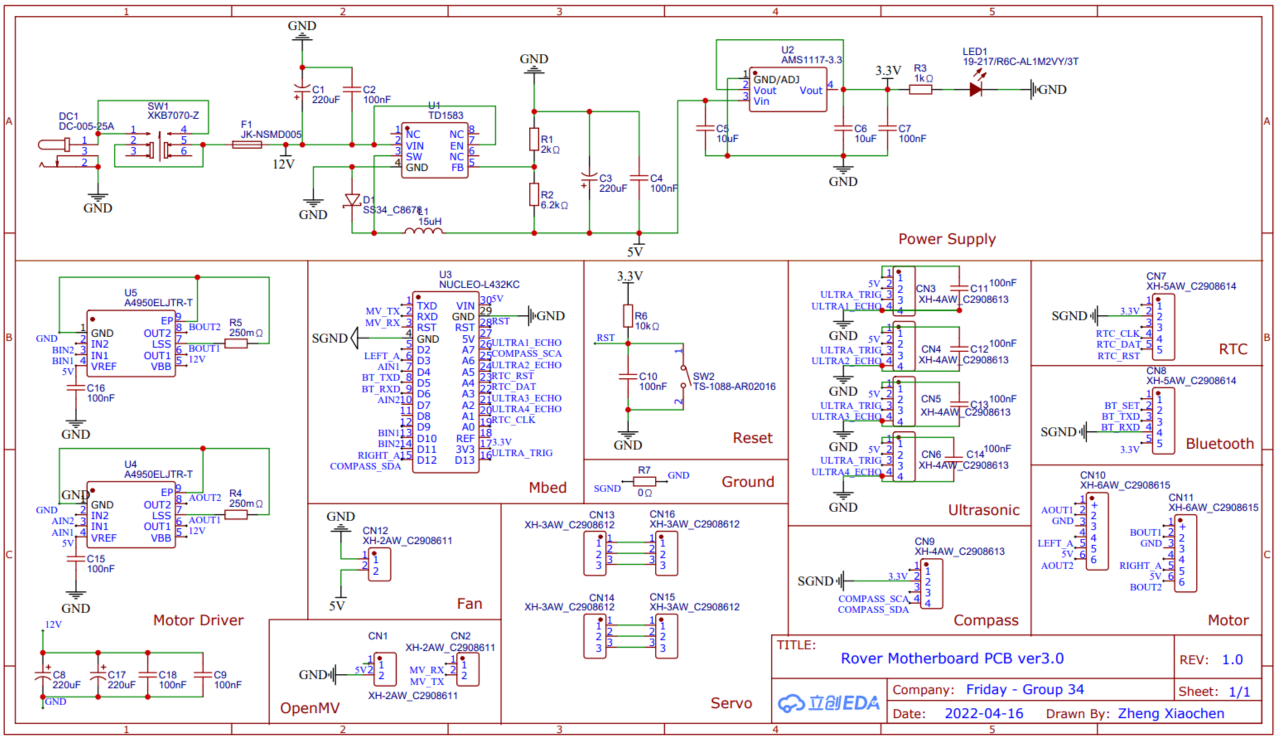


Figure 60. Final circuit schematic.

3.10.3 PCB Layout Design (ZHENG Xiaochen)

3.10.3.1Considerations

From the datasheets of devices used, we get the following information.

- The ceramic decoupling capacitor should be connected to the pin as close as possible.
- Make the inductor loop as small as possible.
- A high pulsing current will pass through the TD1583 buck regulator circuit when the power supply is working. Thus, a ground plane is recommended to connect the input capacitor, output capacitor, and GND pin of the TD1583.
- A star ground is needed to isolate the signal ground and the power ground (large current ground).

- Distributed power to corresponding ports through powering wires on the outer ring of the board, protected by ground plane and far away from high-frequency signals.
- Grounding the sensitive signal with a low-resistance and interference-free path.
- Keep sensitive signal traces away from the large current pins.
- Pour a maximized copper area to all the GND pins and power supply pins to help thermal dissipation.
- The extra connecting pad of A4950 should be soldered onto the ground copper to help thermal dissipation.
- The signal line for OpenMV, ultrasonic, RTC, and compass communication (~100kbps) should be as short as possible for signal integrity.
- Minimise the number of vias for signal integrity.

A sample layout is also given for A4950.

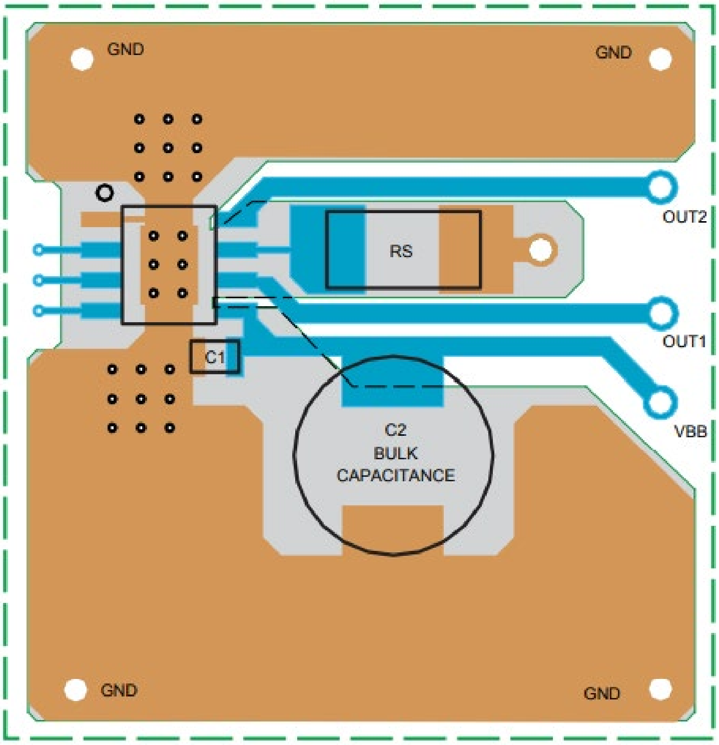


Figure 61. A4950 sample layout [A4950 datasheet].

3.10.3.2Placing

Specifications for PCB placing include:

- Top ports (Facing the rover back): Servo
- Left ports: OpenMV, Bluetooth, RTC, compass
- Right ports: ultrasonic
- Bottom ports (Facing the rover front): DC plug, power switch, motor
- All the ports and switches should be on the edge for easy attachment.
- The bulk circuit should be close to the DC plug to minimise the wire length.
- Sensitive signal components should be placed far from the large current components.

- Board size: 97.4mm x 97.4mm, with the centre of tapped holes distant at 90mm.

The final placing is shown below.

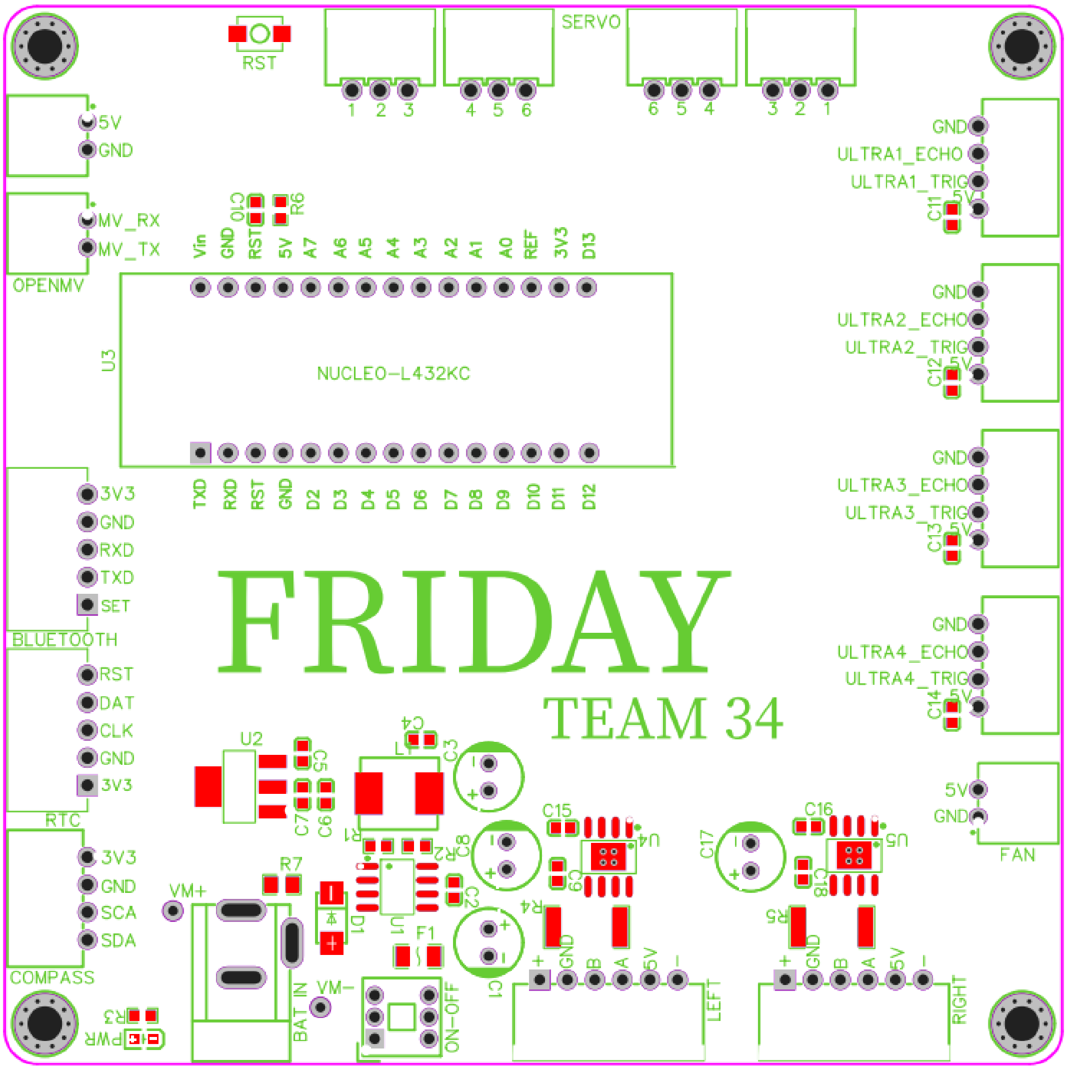


Figure 62. Components placing.

3.10.3.3PCB Routing

The PCB routing follows the following design rules.

- 45-degree line or circular arc (if necessary)
- Wire width: signal 20mil, 3.3v 20mil, 5V 40mil, 12V 70mil
- Vias: three sizes used, 24.016mil/12mil, 48mil/24mil, 64mil/32mil
- No acute angle.
- Teardrops applied.
- Line space larger than 3x times wire width to avoid RF interference.
- DRC rules: wire width > 10mil, line space > 5.984mil, hole diameter 24.016mil/12.008mil
- Minimised vias
- Minimised line width

- Star grounding. Avoid poor grounding loops.
- Polygon pours on heated areas for thermal dissipation.
- Remove tips on polygon pour to avoid antennas.

The final routing is shown in Figure 63 and Figure 64.

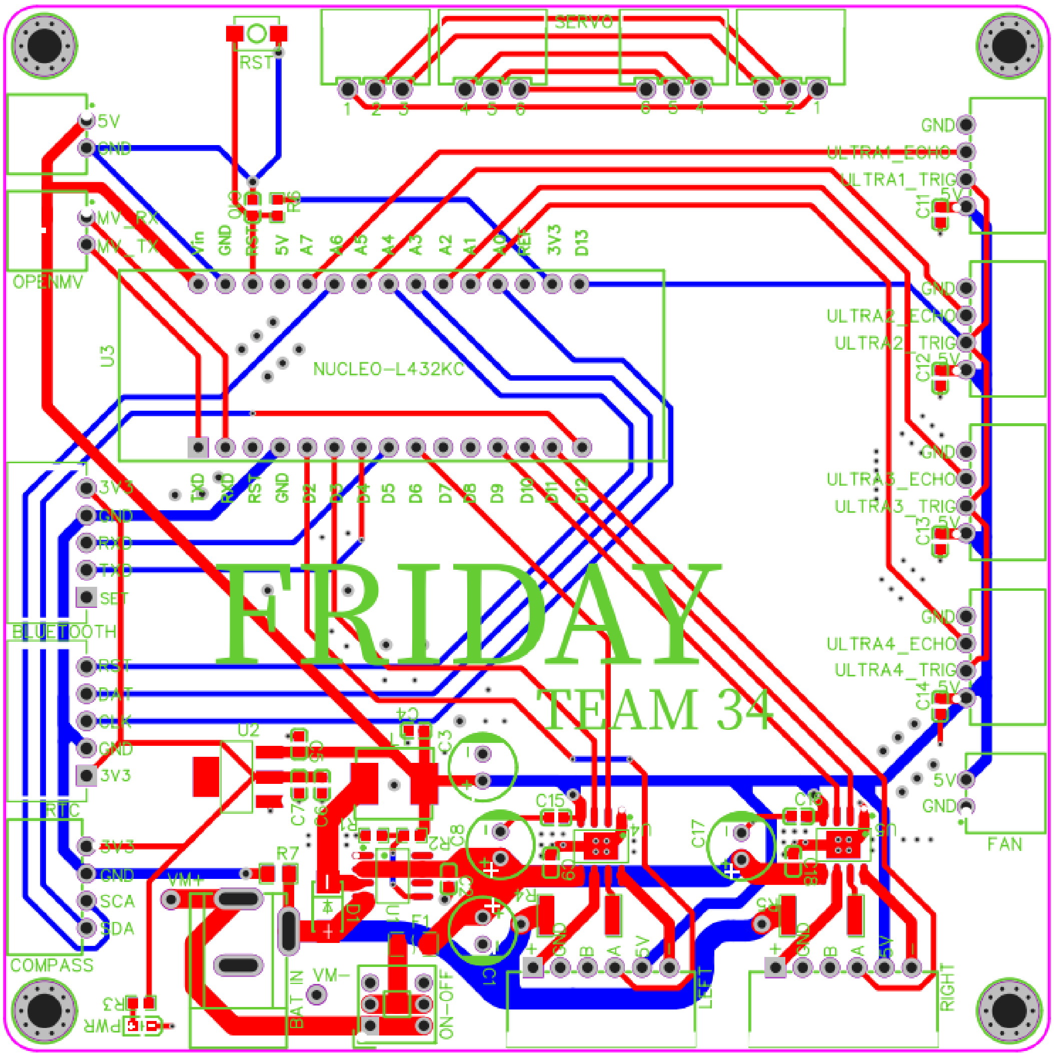


Figure 63. Routing (without polygon pour).

3.10.4 PCB Manufacture (ZHENG Xiaochen)

The manufacturer chosen was JLCPCB, which is a leading global PCB manufacturer founded in Shenzhen, featuring high-quality, cost-effective, and fast-deliver PCB manufacturing for corporations and engineers.

The process requirements selected were 2-layer 1.6mm FR4 substrate and 1 oz finished outer layer copper because it was the cheapest option for a high-standard prototype. A white solder mask colour is selected for aesthetic.

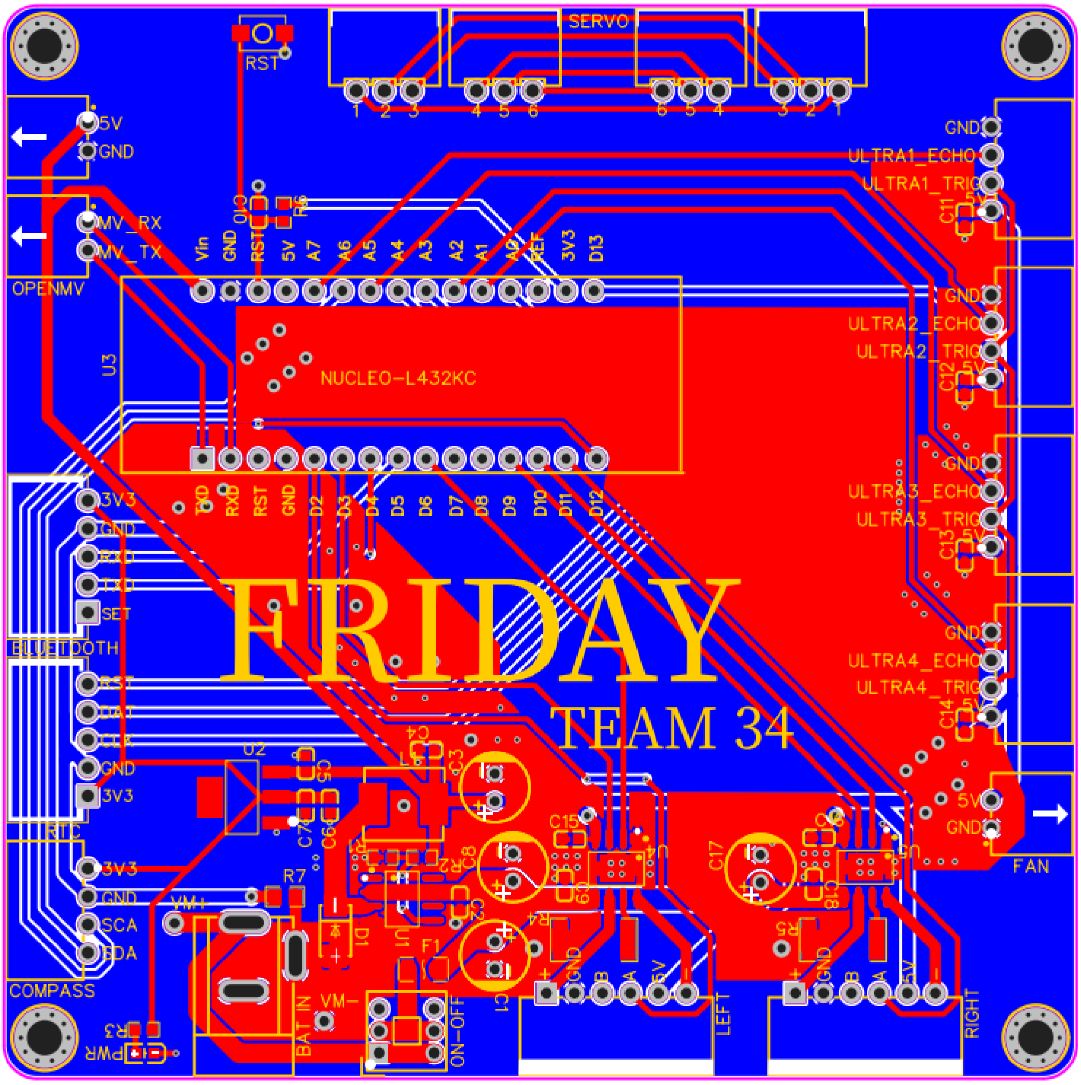


Figure 64. Routing (with polygon pour, final version for fabrication).

3.10.5 PCB Assembly (ZHENG Xiaochen)

The PCB Assembly is done by ZHENG Xiaochen manually. Although it is challenging of mounting as small as 0603 and SOIC-8 packaging devices, Xiaochen is experienced with it and finished with ease.

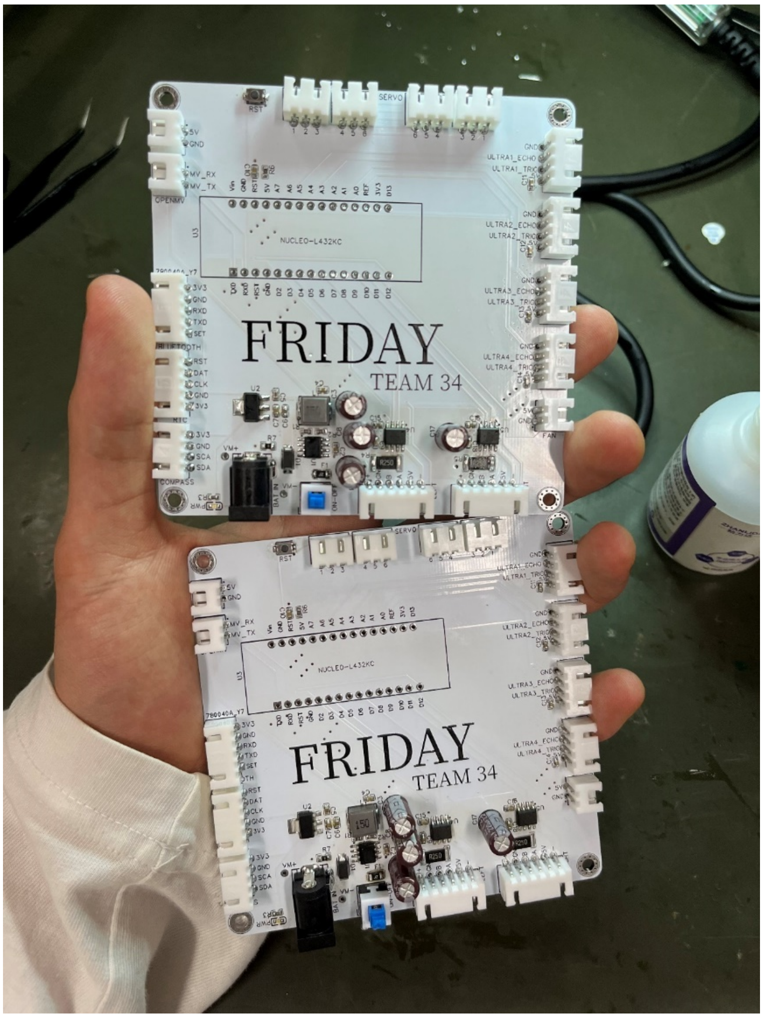


Figure 65. The assembled motherboard PCB.

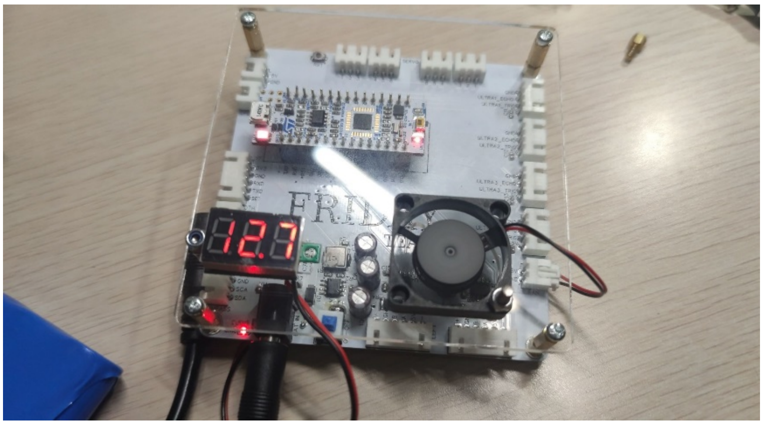


Figure 66. The working motherboard PCB.

In this section, we present our designs for redundancy. Although these submodules did not make it into our “Plan A”, they are essential as they act as our “safety net”.

4

Backup Module Design

4.1 Color Blob Identification (YUE Haiyi)

4.1.1 Objectives

The Color Blob Identification algorithm is applied in Patio 2, when the rover has finished shape detection and is on its way to the ball-releasing site. During this process, there exists a site where the ultrasonic radar system failed to work. As a result, a red edge detection module based on Color Blob Identification was designed to achieve the goal of sheering off.

4.1.2 Working Mechanism

To fulfil the task of red edge detection, the function "find_blobs" from module "image" was selected, to find all blobs in the image and returns a list of blob objects which describe each blob. By testing and adjusting the parameters of the target blob, the red edge could be detected and framed by a rectangular shape. In the meantime, the rover could receive a command of sheering at a proper position.

4.1.2.1 Determine the Threshold Value

The target image was selected on the IDE frame buffer of "Threshold Value Editor", then the image was automatically transferred to a binary image (Figure 67), where the white points stand for the pixels that needed to be tracked.

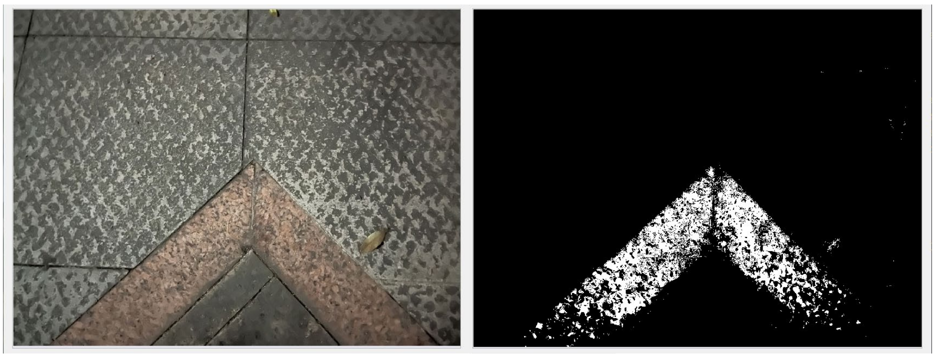


Figure 67. The binary image of the field.

Based on the statistic image pictured on-site, the six parameters of the Threshold Value were generated by the Threshold Value Editor, and recorded as (76, 47, 3, 127, -26, 127).

4.1.2.2 Detect the Target Blob

If any blob detected by the camera has a set of LAB values that could match with the LAB threshold value of the red edge, then the blob would be circumscribed by a rectangular frame. The direction of the rover was adjusted automatically until the width of the rectangular frame reaches its minimum value.

4.1.2.3 Return the Reference Point

Once the match was completed, values of center of inertia and width of red edge would be returned. Based on this reference point, the command of sheering off would be received by the mainboard.

4.1.3 Results and Discussion

4.1.3.1 Direction of the Rover

Due to certain environmental factors, the setting parameters appeared to be inaccurate in practical tests. After repeated trials, the six parameters of the Threshold Value were measured to be: (0, 85, 5, 107, -6, 19). The results of detection are shown below in Figure 68.

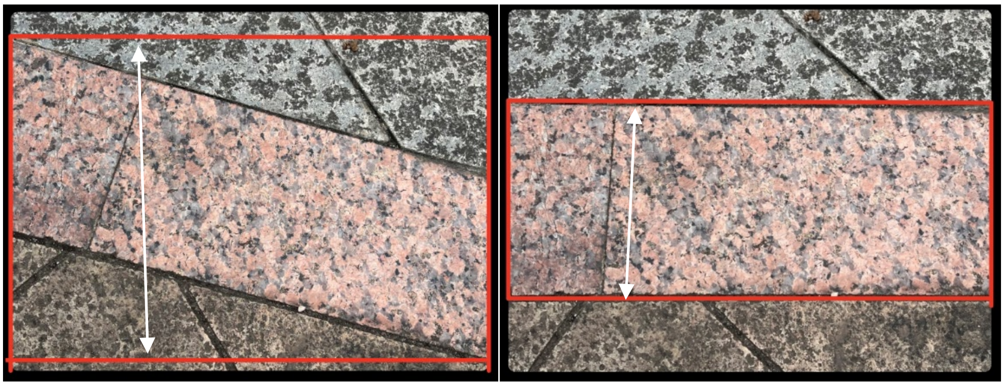


Figure 68. The results of the color blob identification.

In the first figure, the rover was not perpendicular to the red edge, thus the width is wider than it is in the second figure. The direction of the rover in the second figure appeared to be correct. To measure the direction in quantization, an index called Relative Width is defined, describing the ratio of the actual width (d) with the width when the rover is perpendicular to the red edge (d_0):

$$RW = \frac{d}{d_0}$$

The results of several trials are listed below in Table 30.

Table 30. The direction of the rover measured by Relative Width.

Angle Offset (°, right as +)	Relative Width
2	1.01
-3	1.01
15	1.34
-18	1.39
32	1.67
-29	1.66

The direction could be considered acceptable when the direction angle is smaller than 5°, where the Relative Width equals 1.12. By adjusting the direction repeatedly, the rover could limit the Relative Width by 1.12, thus the direction of the rover is correct.

4.1.3.2 Height of the Camera

The height of the camera could affect the angle and scope of images. Three heights were examined in the field test, shown below in Figure 69.

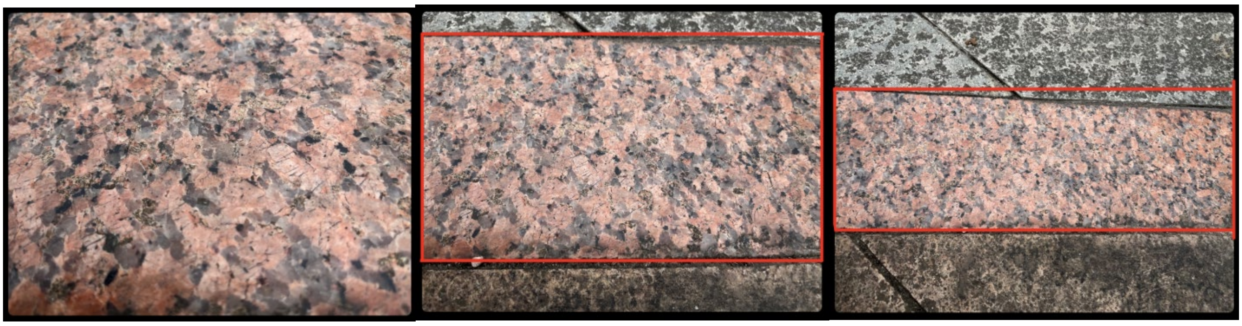


Figure 69. OpenMV installed at a height of 5cm, 15cm and 25cm.

The performance of detection varied by the height of the camera could also be measured by the index Relative Width, as shown below in Table 31. Under such circumstances, Relative Width was defined to be the ratio of the frame width (d) with the width of the screen (D):

$$RW = \frac{d}{D} \times 100\%$$

Table 31. The height of the camera of the rover measured by Relative Width.

Height of the Camera (cm)	Relative Width (%)
5	Out of range
10	96
15	79
20	56
25	43
30	22

To have a more accurate detecting result, there should be enough margin-left on each side of the frame. As a result, the ideal Relative Width should be less than 70%, which means the height of the camera should be no lower than approximately 18 cm. After on-site experiments, the height of the OpenMV instalment was determined to be 25 cm. However, since the height of the rover must meet the conditions for other tasks, it was reduced to 15 cm in the end.

4.1.4 Conclusion

In the “Color Blob Detection” module, an algorithm based on Computer Vision was designed to identify the rhombic red edge on the field, and to determine the position where the rover needs to sheer off. The direction and reference point of the rover could be measured by the width of the frame, and the height of the OpenMV should be greater than 15 cm.

4.2 Basket Detection (YUE Haiyi)

4.2.1 Objective

The Basket Detection algorithm was designed for task 2, in Patio 2, where the rover performs the ball-releasing task. The rover ought to locate the position of the reticular basket and head to it. The basket is made of reticular iron, which means it is difficult to detect it by normal target identification methods. Taking advantage of its meshy property, it could be sharpened to multiple pixel points. To sharpen the image of the reticular basket, the computer vision algorithms were applied using Laplacian transform.

4.2.2 Working Mechanism

4.2.2.1 Convolution and Laplace Transform

Laplacian operator is an image neighborhood enhancement algorithm derived by second order differentiation, which is used for image sharpening. For a continuous binary function f(x,y), its Laplace's operation is defined as:

$$\nabla^2 = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

For digital images, it could be represented in convolution form as:

$$g(i,j) = \sum_{r=-k}^{k} \sum_{s=-l}^{l} f(i-r, j-s) H(r,s), i,j = 0,1,2 \dots, N-1$$

When k=l=1, H (r, s) could be shown as a matrix:

$$H1 = \begin{Bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{Bmatrix}$$

When the gray level of the centre pixel in the neighbourhood is lower than the average gray level of other pixels in the neighbourhood, the gray level of the centre pixel should be further reduced. As for higher ones, the gray level of the centre pixel is further improved. In this way, the image could be sharpened. In this project, two kernel matrices were used to perform the convolution:

$$K1 = \begin{Bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{Bmatrix}$$

$$K2 = \begin{Bmatrix} -2 & -1 & 0 \\ -1 & 1 & 1 \\ 0 & 1 & 2 \end{Bmatrix}$$

After convolution, the image was processed with the function "laplacian(2)" to be sharpened. The sharpened image is composed of multiple white pixel points, where the sum of them could be analyzed and the centre of the sum could be returned.

4.2.2.2 Return the Centre Point

To return to the center reference point of the basket, it is necessary to convert the sharpened image to a gray-scale map first. The centre could be represented by the ratio of the sum of all the x-coordinates of the white points and the number of points. The rover was designed to change its direction according to the reference point of the centre. It would move towards the centre until the centre falls in the middle of the screen.

The flowchart of this process is shown below in Figure 70.

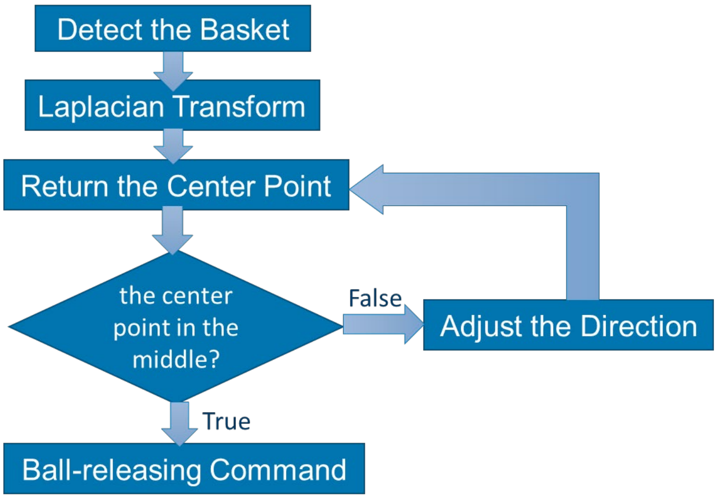


Figure 70. Flowchart of basket detection.

4.2.3 Results and Discussion

The image sharpened using Laplacian equator is shown below in Figure 71.

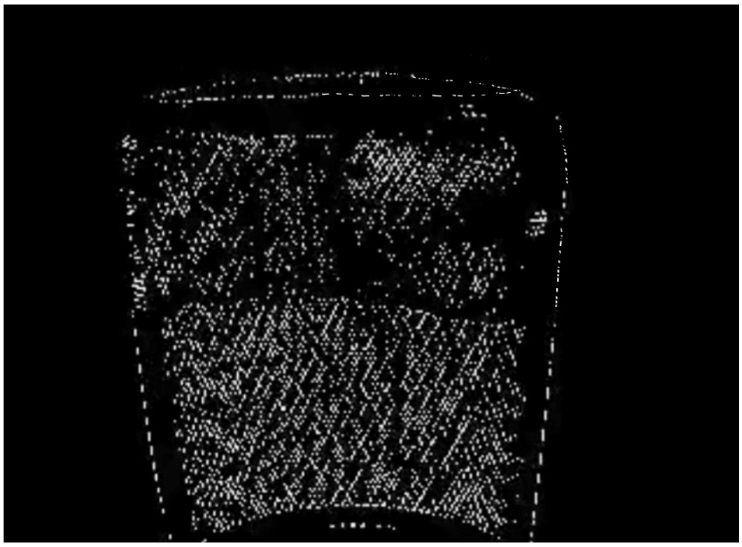


Figure 71. Image sharpened by Laplacian equator.

The x-axis reference point of the center was returned. In practical field tests, due to the factors of lighting, environmental condition and camera height, the clearness of the sharpened image of the basket varied.

To determine the point where the rover should stop travelling and start to release the ball, the dynamic distance from the basket to the rover should be monitored. This distance could be represented by an index measuring the diameter of the basket shown on the screen, defined as Relative Diameter, where d represents the diameter measured and L stands for the length of the screen:

$$RD = \frac{d}{L} \times 100\%$$

The results of several trials are listed below in Table 32.

Table 32. The distance from the rover to the basket measured by Relative Diameter.

Distance (m)	Relative Diameter (%)
1	27
0.5	56
0.3	70
0.2	91
0.1	Out of range

Considering the appropriate distance for the ball-releasing device was less than 0.1m, where the Relative Diameter was not applicable for the measurement, some modifications are needed, and a reference point is needed to be clarified. After repeated trials, 0.3m (70%) was chosen to be the reference point. It was determined that the rover would travel 0.3 m more when the shown Relative Diameter was 70%.

4.2.4 Conclusion

In the “Basket Detection” module, an algorithm based on Computer Vision was designed to identify the position of the basket where the ball needs to be pitched, and to determine the position where the rover stops and pitches the ball. The direction of the rover could be measured by the center point, and the distance between the rover and the basket was represented by the Relative Diameter. When the distance is 0.3m, the rover is prepared to move a little distance and pitch the ball.

4.3 AprilTag (LI Sipei)

4.3.1 Initial Problem Analysis

For Patio1, the roughness of the bridge makes the rover hard to go straight without any assistance. Therefore, our group decided to use a beacon to make sure that the rover could cross the bridge successfully. The main idea was that the rover would be able to recognize the beacon in the center of its view as it passes across the bridge. The maximum detection distance would be about 1 meter.

For Patio2, after the rover releases the ball, the rover would go to the area to send certain message. During that path, there might be chances that the rover cannot go to that area without any beacon guidance. Therefore, a beacon would be put on the planter to instruct the rover. When the recognized distance is within about 50 centimetres, the rover is expected to turn right. The maximum distance would be 2 meters.

4.3.2 Why AprilTag

AprilTag is used as a Beacon for several reasons. First, AprilTag contains more information than just a box as a beacon, therefore, achieving more tasks when necessary. Second, AprilTag is of great convenience for its small payloads and it can be downloaded online and generated via a printer [13]. Third, AprilTag system has high sensitivity and accurate detection abilities. It could give the precise position of the camera, including distance and angles in x, y, z directions of a Cartesian Coordinate [13]. It is better than ultrasound detection for the reason that AprilTag could give the corresponding angles. Compared with QR codes, AprilTag could be detected under longer-distant, poor-lightening conditions, enabling it to handle various types of image distortion problems [14].

4.3.3 Methodology

4.3.3.1 Type of AprilTag

Through many tests and comparison, we finally choose TAG36H11 for the reason that it would achieve high accuracy recognition with a sufficiently long distance.

4.3.3.2 Methodology

The diagonal length of the AprilTag image is calculated by the measured side length of AprilTag. Then the actual distance between the AprilTag and OpenMV could be determined by diagonal image length and the focal length of OpenMV.

4.3.3.3 Turn Right Instruction

The rover is expected to turn right when the actual distance to the AprilTag is within 50 centimers in Patio2. A "turn_bool" is used in the code to achieve this task. When AprilTag is detected and the actual distance is calculated within 50, turn_bool would return true to perform the task. In case of turning multiple times, it is set that once the rover has turned right, turn_bool would remain false. Before the formal demonstration, turning distance, time and duration should be measured in advance.

4.3.3.4 Distance Recognition

For distance recognition, the maximum recognition distances for two patios are far beyond the tested 4cm×4cm AprilTag, which is 0.58m. Therefore, the beacon is designed to have a column of AprilTag ranging from 20cm×20cm to 4cm×4cm, as demonstrated in Figure 72. It also takes the view and height of OpenMV into consideration. From the test results, OpenMV can detect multiple AprilTags simultaneously. The maximum detection distances for the expected AprilTags are shown in Table 33.

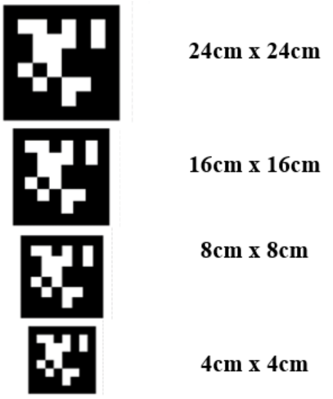


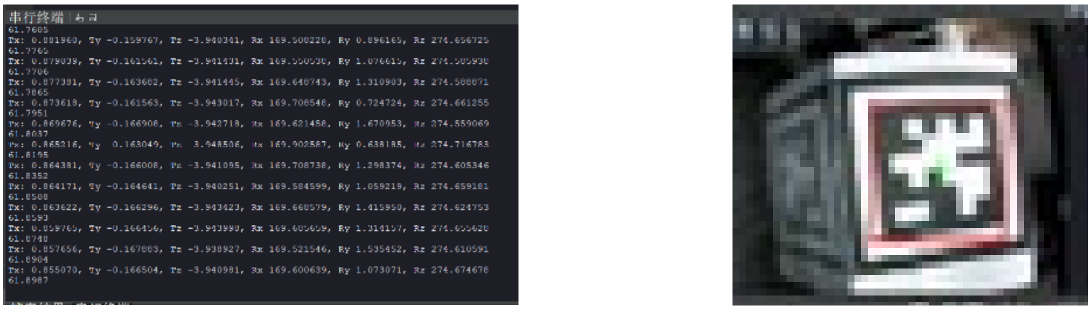
Figure 72. Demonstration of the beacon.

Table 33. Maximum distance.

Measurement	Maximum detection distance (m)
20cm×20cm	2.18
16cm×16cm	1.82
8cm×8cm	1.14
4cm×4cm	0.54

4.3.4 Simulation Result

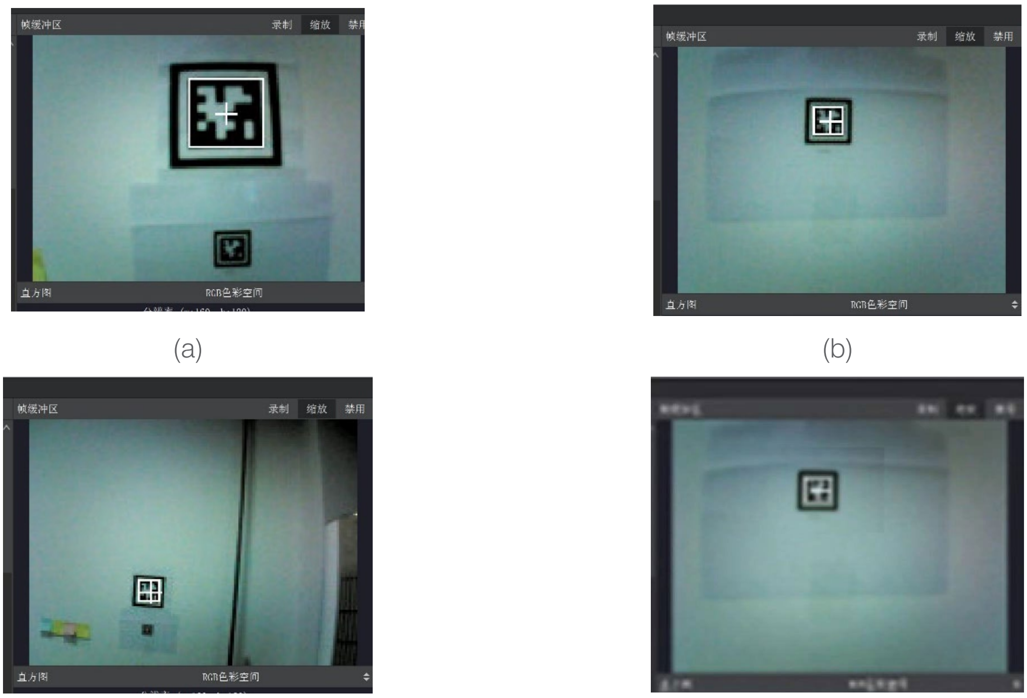
Figure 73 demonstrated that OpenMV has successfully recognized AprilTag with a rectangular and cross drawn. In addition, the 6 DOF positions has accurately measured within an error of 3%.



```
61.7605
Tx: 0.881960, Ty -0.159767, Tz -3.940341, Rx 169.508228, Ry 0.896165, Rz 274.656725
61.7765
Tx: 0.879039, Ty -0.161561, Tz -3.941431, Rx 169.550530, Ry 1.076615, Rz 274.505930
61.7706
Tx: 0.877381, Ty -0.163682, Tz -3.941445, Rx 169.648743, Ry 1.310903, Rz 274.588871
61.7865
Tx: 0.873618, Ty -0.161563, Tz -3.943017, Rx 169.708548, Ry 0.724724, Rz 274.661255
61.7951
Tx: 0.869676, Ty -0.166908, Tz -3.942718, Rx 169.621458, Ry 1.670953, Rz 274.559069
61.8037
Tx: 0.865216, Ty -0.163049, Tz -3.948506, Rx 169.902587, Ry 0.638185, Rz 274.716783
61.8195
Tx: 0.864381, Ty -0.166008, Tz -3.941095, Rx 169.708738, Ry 1.298374, Rz 274.605346
61.8352
Tx: 0.864171, Ty -0.164641, Tz -3.940251, Rx 169.584599, Ry 1.059219, Rz 274.659181
61.8508
Tx: 0.863622, Ty -0.166296, Tz -3.943423, Rx 169.668579, Ry 1.415950, Rz 274.624753
61.8593
Tx: 0.859765, Ty -0.166456, Tz -3.943990, Rx 169.605659, Ry 1.314157, Rz 274.655620
61.8748
Tx: 0.857656, Ty -0.167083, Tz -3.938927, Rx 169.521546, Ry 1.535452, Rz 274.610591
61.8904
Tx: 0.855070, Ty -0.166504, Tz -3.940981, Rx 169.600639, Ry 1.073071, Rz 274.674678
61.8987
```

(a) (b)

Figure 73. (a) 3D position result, (b) AprilTag recognition.



(a) (b)

(c) (d)

Figure 74. AprilTag Recognition of size 20cm×20cm, 16cm×16cm, 8cm×8cm and 4cm×4cm.

4.3.5 Discussion

4.3.5.1 What has been done

In this project, I have used AprilTag as beacon to solve problems in Patio1 and Patio2. For Patio1, the beacon helps the rover to cross the bridge. In Patio2, the beacon instructs the rover to the specified area to send messages. In design, AprilTag TAG36H11 is used, and a column of tags with proper height is designed as beacon to solve the problems of remote recognition and close recognition. Finally, 2.2-meter distance recognition has been achieved. By field test, the rover can tract the expected route with the help of AprilTags and could achieve tasks with high accuracy. However, since AprilTag has been canceled from the final design, further tests are not conducted.

4.3.5.2 Why not AprilTag

However, we didn't apply AprilTag to the final rover is because in order to accomplish shape detection task, OpenMV has to be placed in a depression angle. Therefore, it is not able to achieve long-distance detection for AprilTag. Furthermore, taking simplicity in design for consideration, AprilTag would make the whole system more complex. It might cause more unexpected results with other part of the codes consisting together. Therefore, ultrasound detection is used for both two patios. In patio1, a cardboard beacon is constructed and placed beside the bridge. The rover could detect the distance using ultrasound detection. In patio2, the rover could backtrack to the edge and then go straight to the planter.

4.4 Channel Coding for UART (FENG Zihao)

4.4.1 Task Analysis

When performing UART tests on a breadboard, the success rate is far from satisfaction due to the instability of the hardware. Because of the random nature of the hardware problem, one of the effective methods to improve the success rate is applying channel coding.

4.4.2 Working Principle and Implementation

Essentially, channel coding improves transmission reliability by adding redundancy. STM32L432KC, the sender, is set to send the preset character 10 times repeatedly, while the receiver is set to receive 8 characters as there will be losses during the transmission. As Figure 75 shows, after 8 characters are received, they will be stored in a character list. Then, a counter is set to count the number of occurrences of each character in the list, and then determine the most frequent character according to their occurrences. The most frequent character is then considered as the correct message from STM32L432KC.

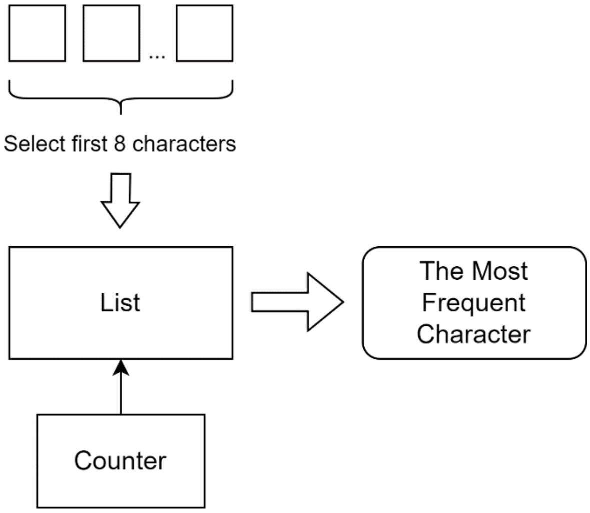


Figure 75. The working principle of the channel coding algorithm.

4.4.3 Results

Table 34. Comparison of the results of direct transmitting and using channel coding. shows the comparison of the success rates of direct transmission and the use of channel coding algorithms. When testing UART on a breadboard, the success rate of direct transmission is less than 40%, this stability is mainly due to the instability of hardware such as wire connections. After the channel coding algorithm is applied, there is a significant improvement in the success rate.

Table 34. Comparison of the results of direct transmitting and using channel coding.

	Direct Transmitting	Using Channel Coding
Success Rate	39.5%	95%
Stability	Low	High

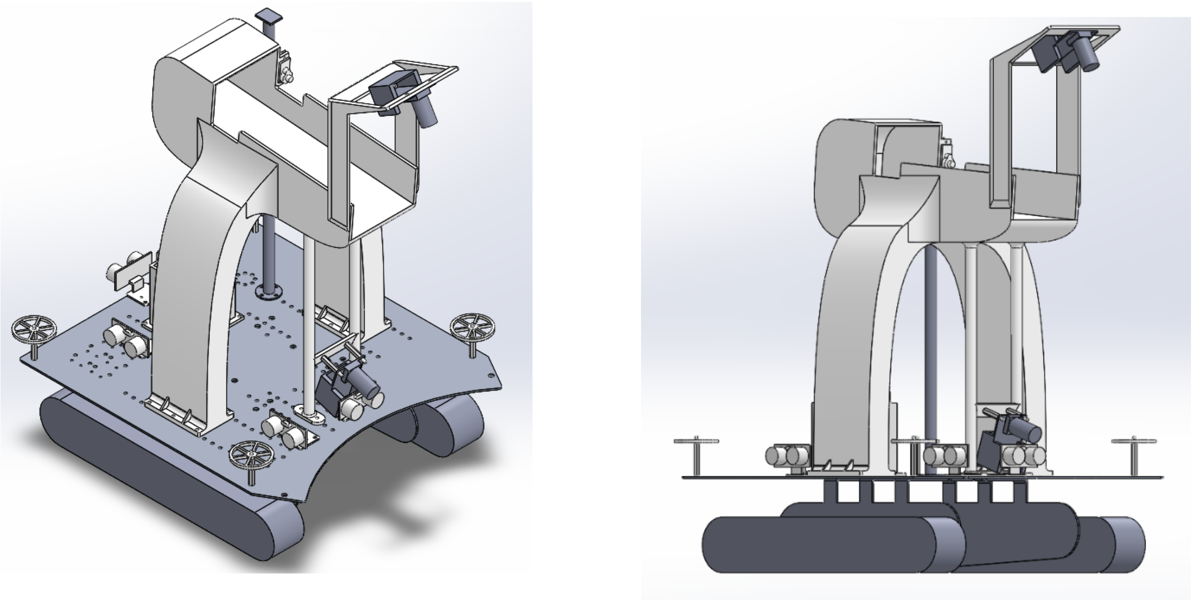
4.5 Ball Releasing Apparatus – A Chute (ZHENG Xiaochen)

Before the ball releasing apparatus with swing arm construction previously introduced in 3.6 was designed, a chute-shaped apparatus was earlier proposed, designed, made, and tested. The principle of this device is that the ball is placed on an elevated chute, blocked by a gate. When the rover moves in front of the bin, the gate is raised by a servo and the ball rolls down the chute by gravity and eventually falls into the bin. This design proved to be functionally effective but did not fit with our rover, so we eventually abandoned this option. The detailed reasons will be discussed in the next topic.

4.6 One-Piece Moulded Frame (ZHENG Xiaochen)

4.6.1 Design

As the chute was sustained above the bin, specifically, 30cm above the ground, the chute and its supportive structure became the bulk of the rover. The design concept was that the structure can also provide contact holes for other components, such as OpenMV. This one-piece structure was designed and modelled in Solidworks. A preview is shown below.



(a) (b)

Figure 76. Preview of the designed one-piece frame.

As shown in the figure, the structure achieves the function of releasing the ball, and provides two installation positions for the OpenMV. Since the structure was modelled as a whole, with dedicated curves, the only fabrication method is 3-D printing. However, the overhead for 3-D printing of this structure exceeded 1,000 RMB, so we dropped this option. Then we turned to another option, using stacked and pieced acrylic panels to create a similar structure. The 3-D model preview and was shown below.

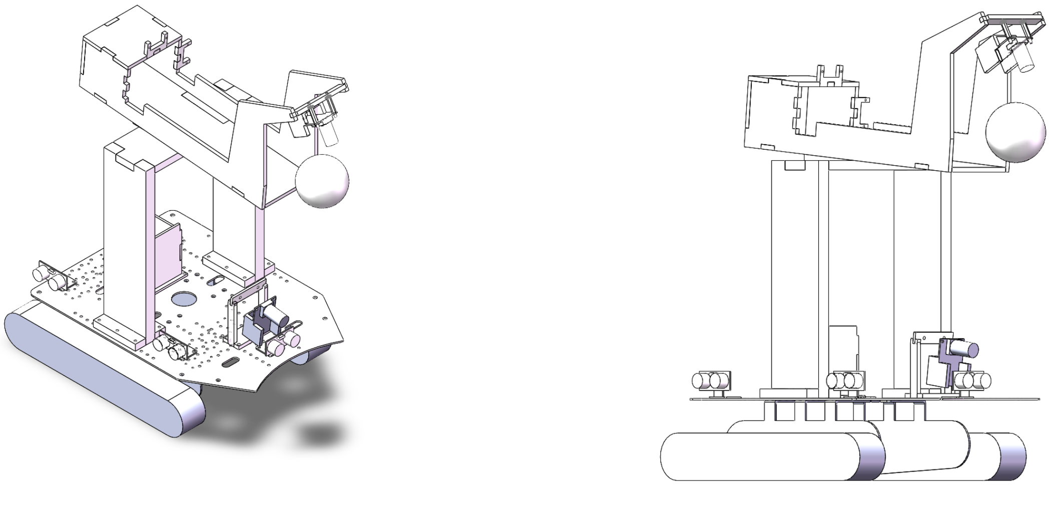


Figure 77. Preview of the designed acrylic-composed frame.

The drawings for fabrication are provided in the appendix.

4.6.2 Discussion

We prototyped and assembled the structure in the field.

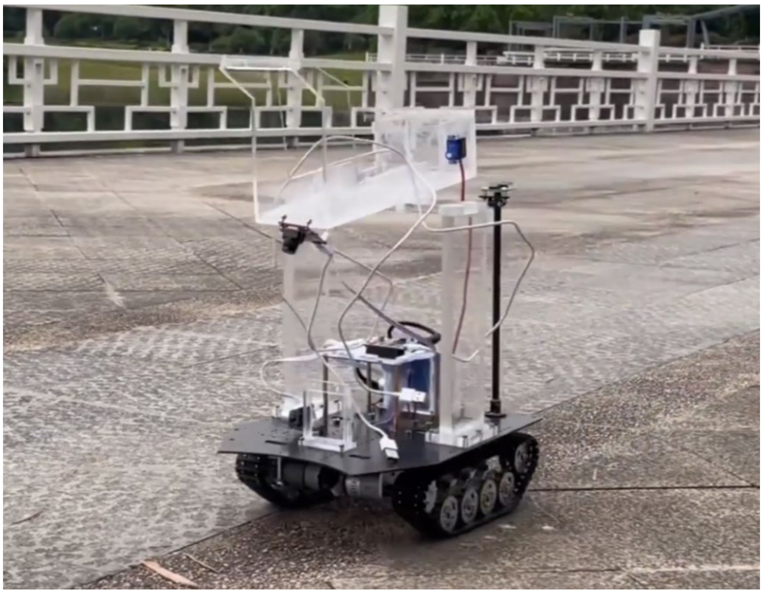


Figure 78. Rover with acrylic-composed frame.

We carried out tests on the operation of the rover and the ball releasing function. The ball releasing function worked well, while the rover could ride smoothly on flat ground. However, the centre of gravity was so high that the car would tip forward when getting on and off the bridge and would still tip over even with the added weight on the bottom. Thus, we abandoned this design.

4.7 Edge Detection-Based Dark Gravel Recognition Algorithm (YANG Pei)

This algorithm adopts an edge-detection approach toward the differentiation of the gravel path from other parts of the patio surface. Due to the gravel path being rich in texture, an edge-detector could extract denser textures from within the gravel region.

For an input image, this algorithm proposes to perform a sequence of operations listed in Table 35.

Table 35. Plan B of the gravel recognition algorithm.

	Operation	Purpose
1	3×3 median pooling	Salt-and-paper denoising
2	Canny filtering (with lower and upper threshold equals the 5th and 95th percentile of the pixel values)	Edge detection
3	4×4 midpoint pooling	Low-pass filtering
4	7×7 gaussian blurring	Low-pass filtering
5	Binary thresholding (with lower threshold equals the 10th percentile and no upper bound)	

Although acceptable, field tests showed that qualitatively, this algorithm was more susceptible to the pattern on the gravel's left side than our Plan A. Another disadvantage was that this algorithm has an average processing speed of around 2fps, while our Plan A reaches up to 10fps. Hence this algorithm was eventually not adopted.

5

System Integration, Results and Discussion

5.1 Design of Working Flow (XU Ben’s Individual Section)

Patio 1: The rover can complete all the tasks and transitions using the functions introduced in the Individual report section. The working flow of Patio 1 is as shown in Figure 79. The rover will first start tracking along with the road until the ultrasonic radar detects the beacon by measuring the distance. After that, the rover will turn right to 90°. To make sure that the rover can drive to the bridge and not drop out of it, ultrasonic tracking is applied and a beacon is used as the reference. Once the rover passes the bridge, it will move forward until it detects the second beacon on the other side of the bridge. After that, the rover will turn left to track along the last part of the path until it detects the gate by measuring the distance.

Patio 2: The working flow of patio 2 is much more complicated than that of patio 1 because the transition of tasks is more challenging. The working processes of the three tasks and their transitions are shown in Figure 80, Figure 81 and Figure 82.

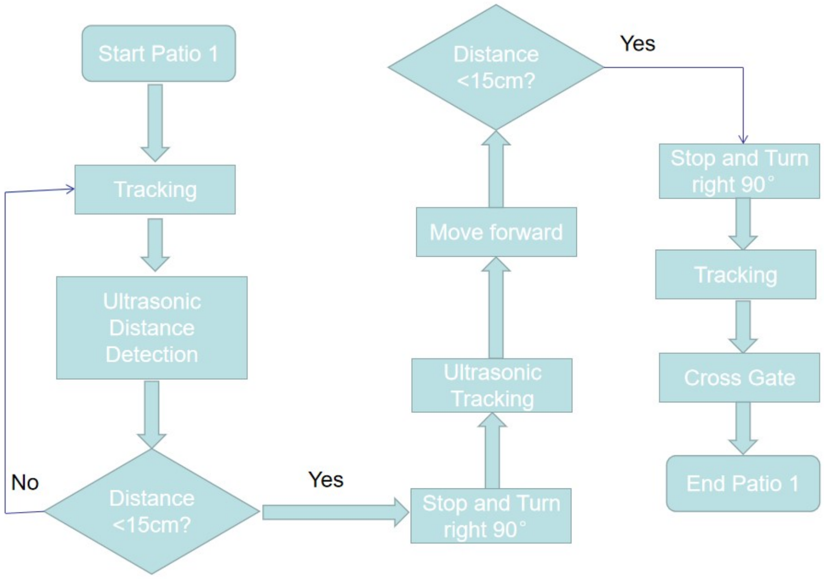


Figure 79. Working flow of Patio 1.

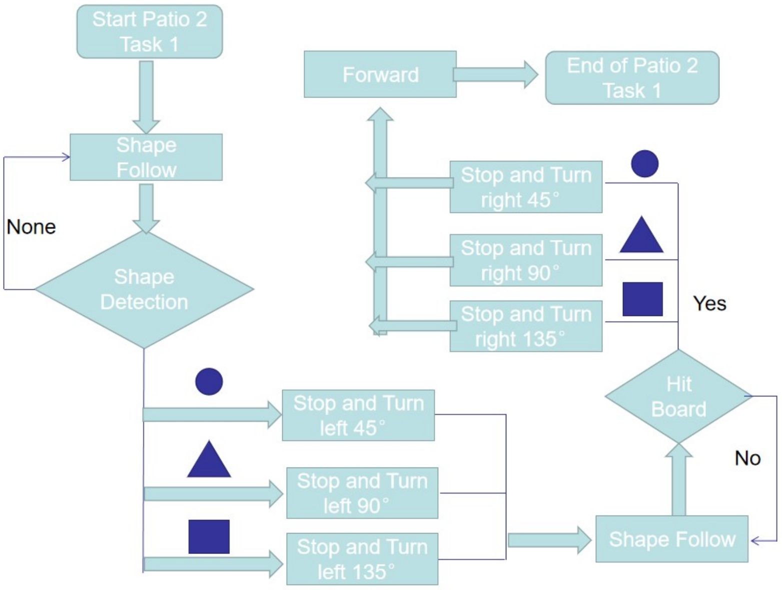


Figure 80. Working flow of Task 1, Patio 2.

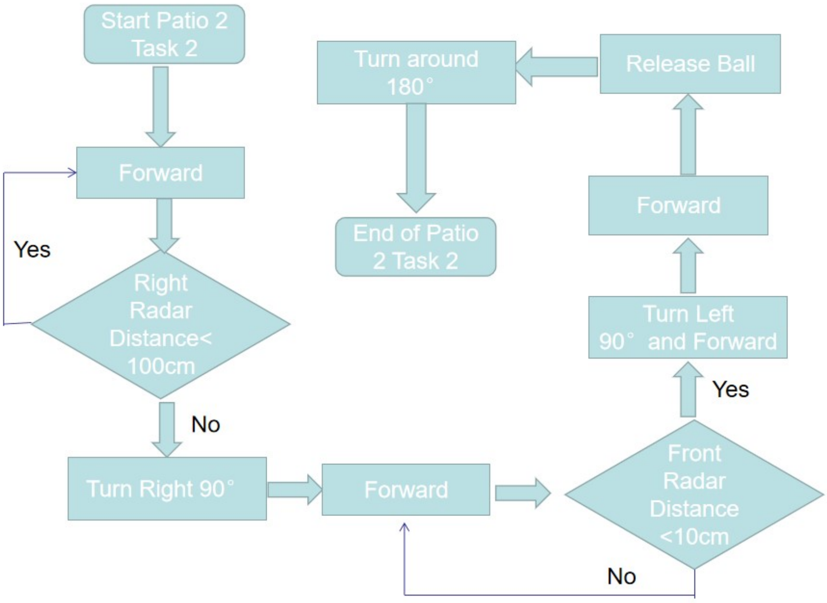


Figure 81. Working flow of Task 2, Patio 2.

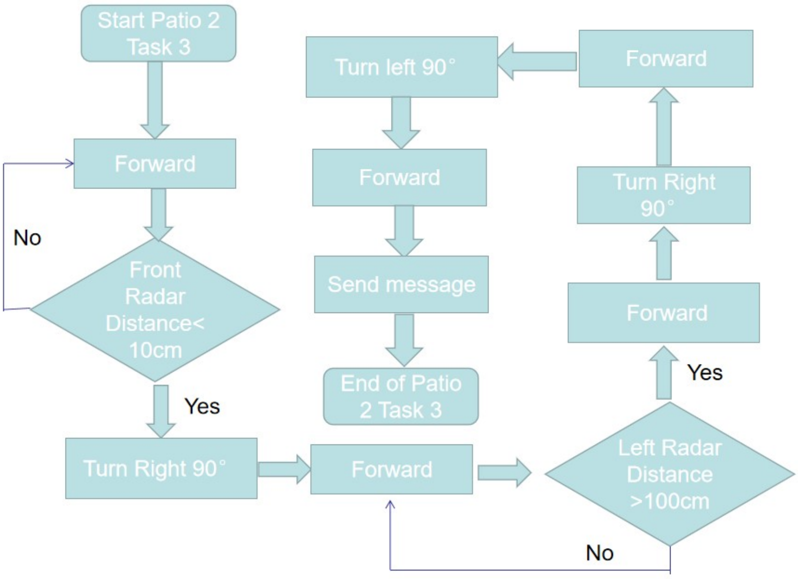


Figure 82. Working flow of Task 3, Patio 2.

5.2 Integration and Optimization

5.2.1 Coordination of MBED and OpenMV (XU Ben's Individual Section)

The feedback control system can take visual signals using the camera on the OpenMV. The signal is then analyzed by a task-specific algorithm. In Task 1 of Patio 1, the road surface with a special texture will be identified. The signal

The project is to design an intelligent rover that can complete the tasks in Patio 1 and Patio 2 automatically. As a result, the sub-systems should be integrated. Otherwise, the rover will not be able to react to the real-time event. The system consists of three major sub-systems, including the intelligent control system, communication system and releasing system. Limited to the performance and number of I/O ports of the MBED, the visual feedback system, communication system and releasing system are mounted on the OpenMV. As a result, the interconnection between the MBED and OpenMV should be realized to integrate the systems.

UART is applied to realize the connection between the MBED and OpenMV. The communication is organized as a master-slave communication mechanism while the MBED is the main controller and OpenMV is the slave. Functions are packaged in the OpenMV as shown in Table 36. MBED will send the corresponding command code to the OpenMV and then wait for the return value to decide the next action.

Table 36. Command-reacting table.

Function name	Command Received	Return Value
Tracking_dark ()	1	0-8 (Degree of Left-right deviation)

Function name	Command Received	Return Value
Tracking_bright ()	2	0-8 (Degree of Left-right deviation)
Shape_follow ()	3	0(Straight forward) 1(Right turn) 2(Left turn)
Shape_detection ()	4	0(None) 1(Triangle) 2(Rectangle) 3(Circle)
Ball_releasing ()	5	1 (Success)
Communication ()	6	1 (Success)
Wrong_command ()	Others	‘a’ (Tell the MBED to resend)

5.2.2 Integration of Visual Feedback Control System (XU Ben’s Individual Section)

The feedback control system can take visual signals using the camera on the OpenMV. The signal is then analyzed by a task-specific algorithm.

In Task 1 of Patio 1, the road surface with a special texture will be identified. The tracking algorithm will return a floating-point value between 0 and 1. The major problems solved in the integration and field test of the tracking algorithm are summarized as follows.

A. Determination of the height and angle of the camera

Patio 1: At the beginning of the design, the camera is expected to be placed as high as possible since this can help the camera capture more information. However, in the field test, it turns out that the height of the camera should be around 30 cm from the ground. A comparison of the visions captured at 30cm and 45cm is shown in Figure 83. If the camera is placed higher, its field of vision will be too large. As a result, more stains around the lane will be included in the video and the input signal to the algorithm has a lower signal noise ratio (SNR). In addition, a higher position can also cause the loss of the details of the road texture since the quality of the vision is limited to QVGA (320 x 240).

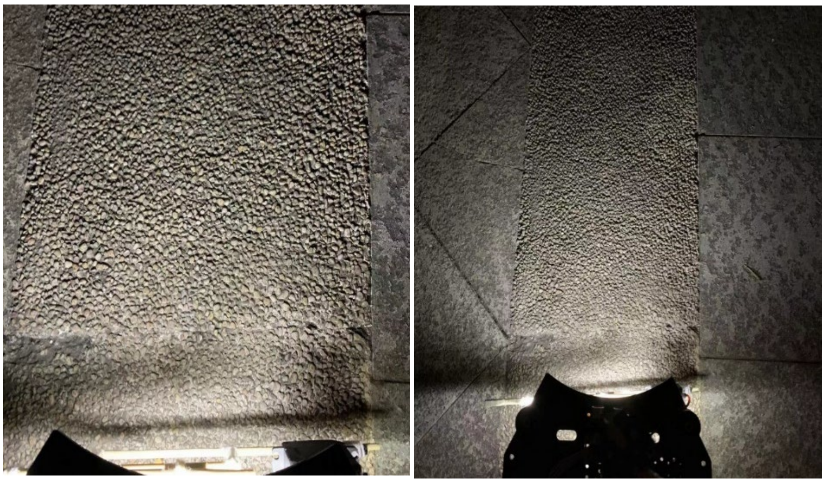


Figure 83. Photos captured at 30cm and 45cm.

The angle between the camera and the vertical is originally designed to be 30°, however, after the field test, the angle is adjusted to 15°. The view difference caused by angles is shown in Figure 84. Since the camera is not heading to the ground vertically, the y-axis (vertical axis) of the camera is compressed and this distorts the vision. A longer distance can also affect the performance of the algorithm.

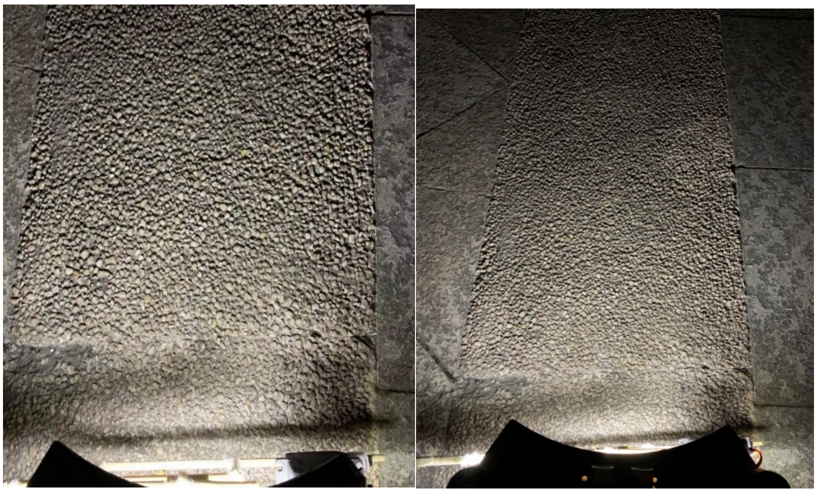


Figure 84. The situation of a camera mounted with an angle of 30° and 15°.

Patio 2: The camera is applied to detect the position and the shape of a given paper. To capture a figure with high quality, the height of the camera should be the same as the shape on the paper and the incident angle should be vertical to the paper.

B. Offset of the x-axis of OpenMV

OpenMV is applied to identify the position of the shape as well as the position of the road. As a result, it is important to make sure that it can correctly determine the centre of the image. In the field test, it is found that the camera of OpenMV deviates to the right, and the x-coordinate of the real centre is 0.57. This can cause the rover to keep moving on the right side of the correct path. To solve the problem, an offset of -0.07 is applied to the x to neutralize the impact.

5.2.3 Integration of Distance Feedback Control System (XU Ben's Individual Section)

The rover should be able to complete the transitions among the tasks automatically. The main problems appear in the transition between Task 1 and Task 2 in Patio 2. In our design, the rover should be able to know its position based on its distance from the surroundings. Based on the distance information, the rover will track along the fence to achieve the position to release the ball. Four ultrasonic radars are used to detect the distance from the surroundings. The algorithms for the tracking are designed based on the mean average distance as well as the calculated angle.

5.2.4 Integration of Speed Feedback Control System (XU Ben's Individual Section)

A man-controlled rover can move in a straight line easily because human acts as a feedback system and can correct the direction of the rover with a controller. However, it is very difficult to achieve it. The major problems include the mismatched motors, deviation of center of gravity, and difference in friction between two caterpillars. Therefore, a speed feedback control system was designed.

The system applied a PID controller using the rotating speed of the motors. The three parameters Kp, Ki and Kd are adjusted so that the system is near the critical damping point. However, the reference speeds of the two motors are different due to the unbalanced motors and caterpillars. The reference speeds can affect the straight-line driving ability in a steady state. After the experiments, the reference speed for the left motor is modified to be 301 while the reference speed for the right one is set as 295.

5.2.5 Stability of the Hardware Connection (CUI Sijie's Individual Section)

The rover should pass through various types of roads and bridges, which means the hardware connection should be secure and stable to prevent possible failures. A PCB has been designed to integrate the electronic devices as much as possible. However, there are still many components that cannot be fixed on the PCB, such as the OpenMV, ultrasonic radars and releasing system.

At the beginning of the field test, the peripherals are connected to the I/O ports on the PCB using Dupont thread. However, it turns out that the connection is very unstable and can cause bad contact. It can affect the performance of the UART communication between MBED and OpenMV heavily. To solve the problem, we weld all the important connecting wires to the ports of the PCB. In addition, a hot melt adhesive is used to secure the wire welding joint. In this way, all the components are fixed. The drawback is it is difficult to make any change to the system. However, considering that all the tasks can be completed by the current system, it is not necessary to replace any components unless they break down. We believe the stability of the system is much more important than the flexibility to modify in our final design.

5.3 Field Test Results and Discussion

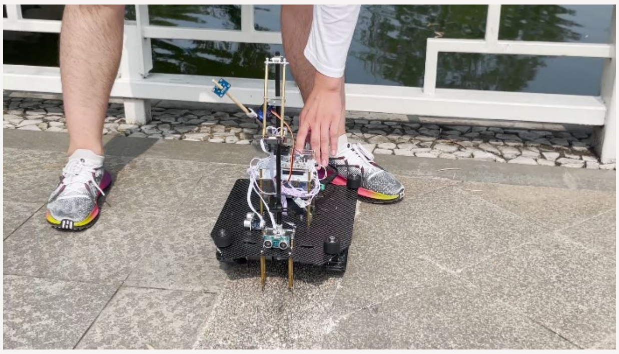
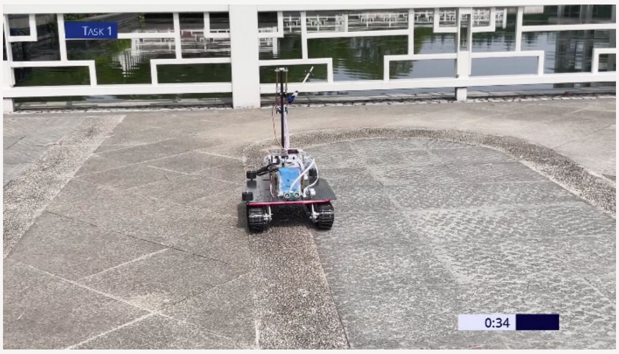
5.3.1 Patio 1

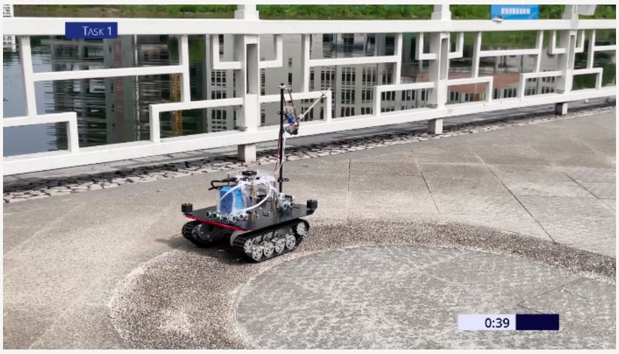
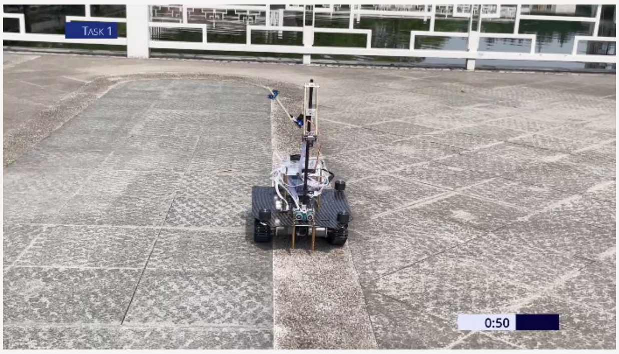
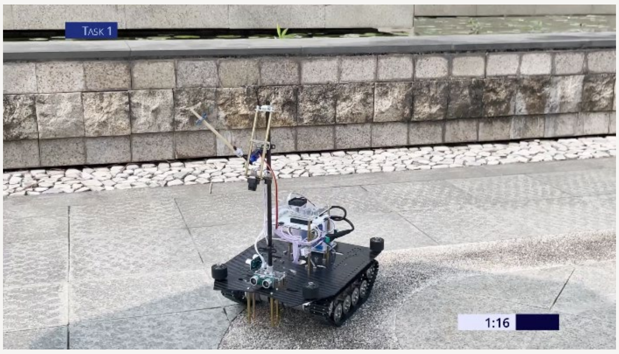
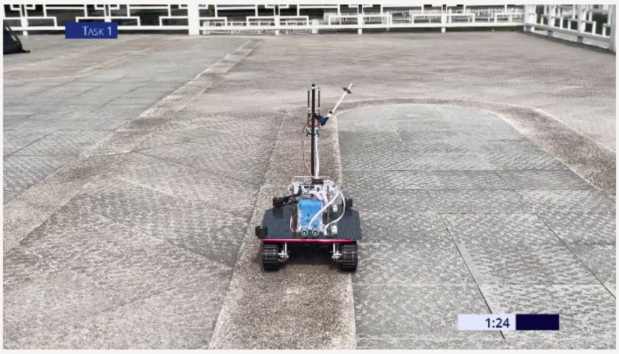
Field test results and discussions have been organised and presented in Table 37.

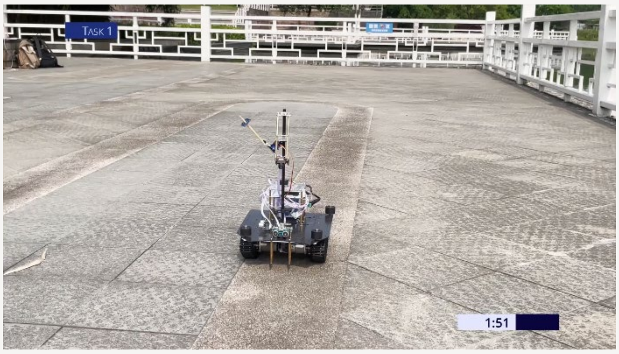
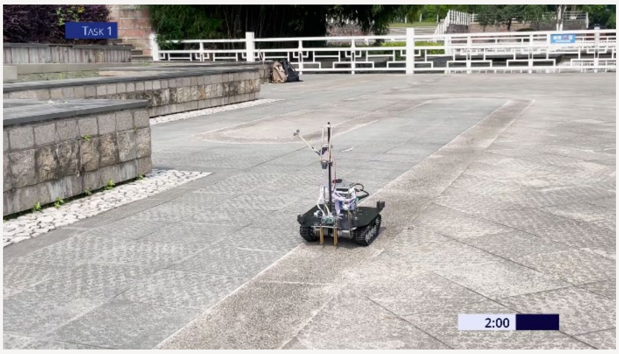
5.3.2 Patio 2

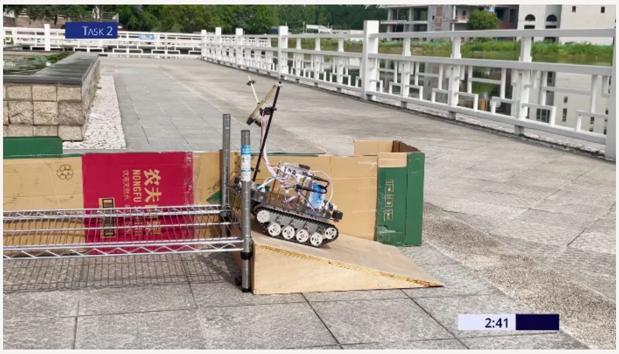
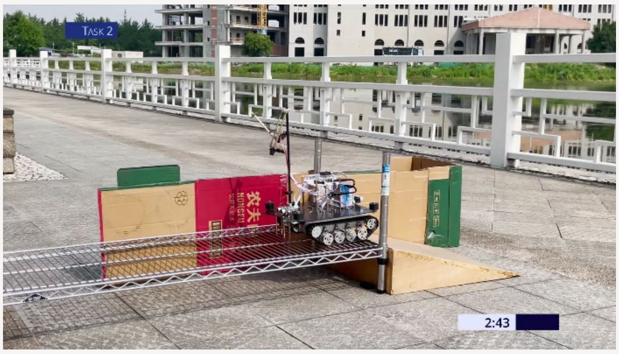
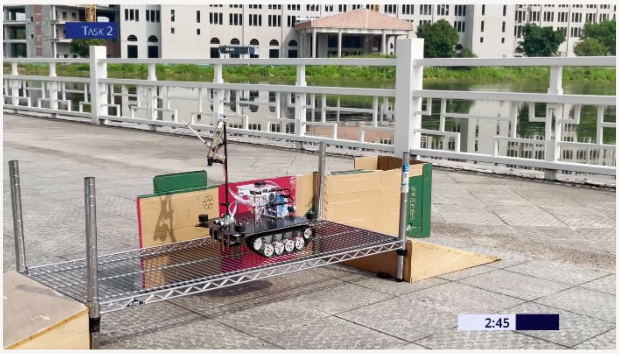
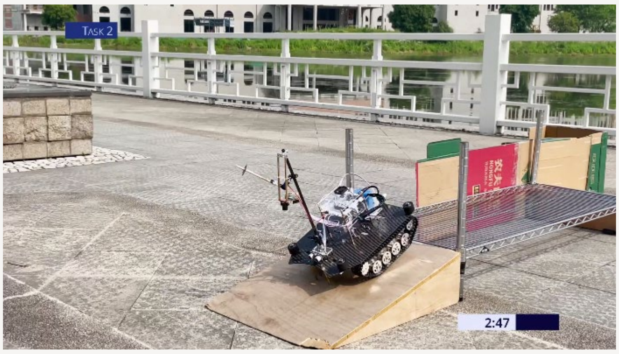
Field test results and discussions have been organized and presented in Table 38.

Table 37. Patio 1 field test results and discussion.

Results	Discussion of Results
 Figure 85. The rover on Patio 1 at start.	**Start:** The rover was placed in the middle of the lane and began travelling three seconds after it was powered on.
 Figure 86. The rover on Patio 1 at 0:15.	**Task 1 Straight Lane 1:** The rover travelled straight with mild adjustments.
 Figure 87. The rover on Patio 1 at 0:20.	**Task 1 Curve 1:** The rover made a rapid turn. This is due to the large pitch angle of the OpenMV camera, which limits the forward perception range of the rover. When the rover detects the curve, it should turn sharply to prevent travelling off-track. Note that the large pitch angle was a trade-off, as a large angle significantly improves the gravel recognition accuracy.
 Figure 88. The rover on Patio 1 at 0:34.	**Task 1 Straight Lane 2:** The rover travelled very straight with hardly perceptible corrections.

Results	Discussion of Results
 Figure 89. The rover on Patio 1 at 0:39.	Task 2 Curve 2: The rover travelled almost along the central axis throughout the curve.
 Figure 90. The rover on Patio 1 at 0:50.	Task 1 Straight Lane 3: The rover travelled straight with mild corrections and a small constant offset to the right of the central axis during the first half of the long straight section. The offset could be attributed to the texture of the patio surface on the right. This corresponds with the analysis in Section 3.3.2.6 that the similarity in texture could increase false positives and reduce recognition accuracy.
 Figure 91. The rover on Patio 1 at 1:16.	Task 1 Curve 3: The rover slightly travelled out of the curve but made a sharp turn to correct back into the lane. This is due to the outer border of the gravel being a 90° arc instead of a 180° arc. The missing 90° implies that theoretically, a 45° in-place turn is required when entering and exiting the curve section.
 Figure 92. The rover on Patio 1 at 1:24.	Task 1 Straight Lane 4: The rover travelled straight with slight oscillations of heading. The oscillation is the transient response of the previous sharp turn made at the curve.

Results	Discussion of Results
 Figure 93. The rover on Patio 1 at 2:00.	**Task 1 Curve 4:** The rover travelled through the curve perfectly.
 Figure 94. The rover on Patio 1 at 1:51.	**Task 1 Straight Lane 5:** The rover travelled straight with mild oscillations throughout this straight section due to the similarity in texture between the gravel and the patio surface.
 Figure 95. The rover on Patio 1 at 2:00.	**Technical Timeout 1:** The rover held short on the gravel after two minutes of travelling, indicating that the right ultrasonic sensors were on standby, ready to detect the beacon.
 Figure 96. The rover on Patio 1 at 2:38..	**Task 2 90° Turn 1:** The rover performed an in-place 90° turn to the right and aligned with the bridge's central axis.

Results	Discussion of Results
 Figure 97. The rover on Patio 1 at 2:41.	Task 2 Climbing the Slope: The rover climbed up the slope with a slow but steady attitude.
 Figure 98. The rover on Patio 1 at 2:43.	Technical Timeout 2: The rover held short after climbing up the slope, indicating that the right ultrasonic sensors were turned off and the front ultrasonic sensor was turned on.
 Figure 99. The rover on Patio 1 at 2:45.	Task 2 Crossing the Bridge: The rover travelled on the bridge fast. The rover's tracks stuck into the metal bars on the bridge, helping the rover align its heading.
 Figure 100. The rover on Patio 1 at 2:47.	Task 2 Descending the Slope: The rover descended the slope fast. The supporting metal bars mounted below the front bumper of the rover touched the ground, which reduced severe bumping of the rover.

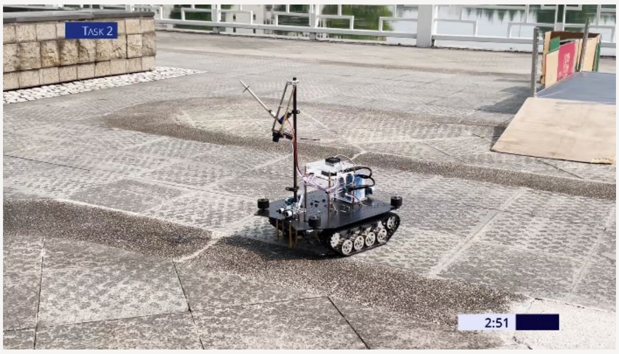
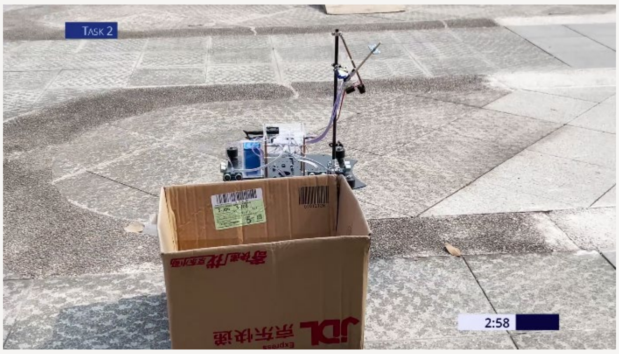
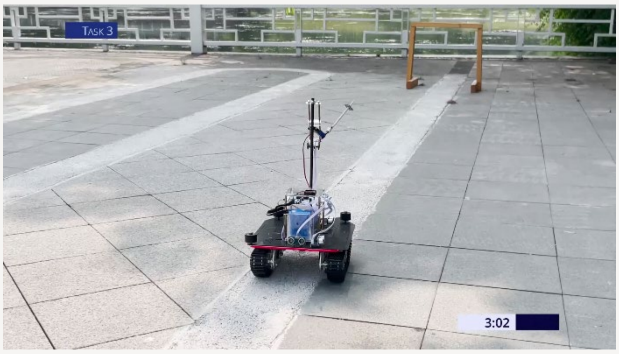
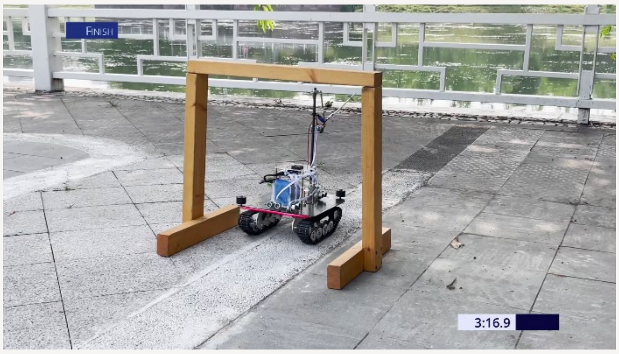
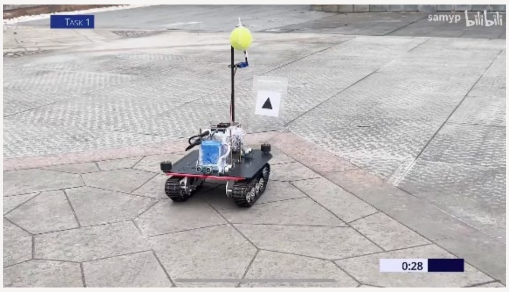
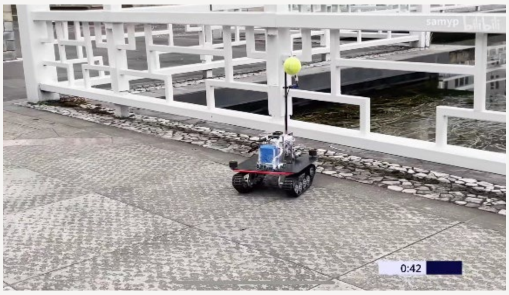
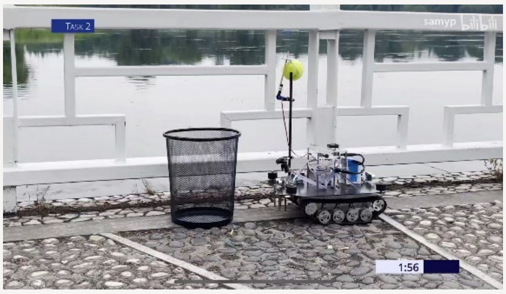
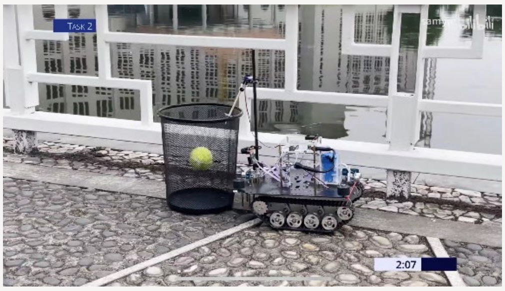
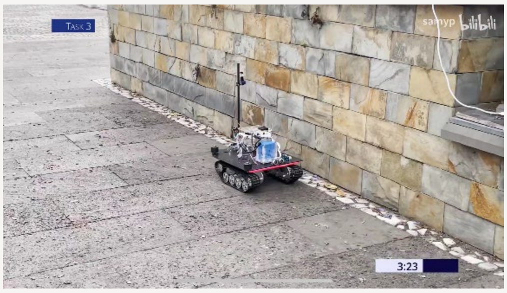
Results	Discussion of Results
 Figure 101. The rover on Patio 1 at 2:51.	Task 2 Travelling Forward: After descending the slope, the rover deviated from the central axis to the right. This was due to no feedback control system being activated at this time.
 Figure 102. The rover on Patio 1 at 2:58.	Task 2 90° Turn 2: After detecting the beacon placed in front of its estimated path, the rover made a precise 90° turn to the left and then travelled forward, with approximately 10° error in heading.
 Figure 103. The rover on Patio 1 at 3:02.	Task 3: After entering the white gravel section, the rover rapidly corrected its heading and started tracking the gravel. The rover travelled forward with an offset to the left of the central axis, possibly due to the brighter patio surface on the left that caused false-positive thresholding results.
 Figure 104. The rover on Patio 1 at the finish point.	Finish: The rover stopped short after passing through the bridge, finishing all three tasks in Patio 1 within 3:16.9.

Table 38. Patio 2 field test results and discussion.

Results	Discussion of Results
 Figure 105. The rover on Patio 2 at the start.	**Start**: The rover was placed in the middle of the field, where the starting line was located.
 Figure 106. The rover on Patio 2 at 0:10.	**Task 1 Shape Recognition:** The rover would recognize the shape on the standing board.
 Figure 107. The rover on Patio 2 at 0:28.	**Task 1 Target Tracking:** The rover achieved successful tracking in a straight line. The motor control was first designed as an open-loop system, leading to frequent yaw of the rover. To make the rover travel straight, a negative feedback system was added. It was verified to be an effective way to eliminate offsets.
 Figure 108. The rover on Patio 2 at 0:33.	**Task 1 Strike of Target Board:** The target standing board was successfully stroke by the rover. The rover was designed to determine its position relative to the target board based on computer vision, to make the system more stable and more flexible for direction correction. During multiple field tests, the rover successfully stroked the target board as expected, even when the rover understeered or oversteered.
 Figure 109. The rover on Patio 2 at 0:42.	**The transition from Task 1 to Task 2:** The rover completes the transition part by ultrasonic positioning. Since the height of the fences is ragged, the position of the ultrasonic was of vital importance. During the process of travelling along the fences, the rover showed a stable performance every time. This was because the valid range of the ultrasound was measured in advance, at a height of 11.5-12.5cm and

Results	Discussion of Results
	our ultrasound was eventually placed at a height of 12cm.
 Figure 110. The rover on Patio 2 at 1:56.	Task 2 Basket Tracking: The curved design on the front of the rover was used to prevent the mismatching between the rover and the basket, which may cause the ball to be dropped out of the basket. The curve was proved in the field tests to be conducive, that the basket could be accurately located in front of the rover. As a result, the pitching system was very stable.
 Figure 111. The rover on Patio 2 at 2:07.	Task 2 Ball Releasing: Controlled by the servo, the ball was successfully pitched in the basket.
 Figure 112. The rover on Patio 2 at 3:23.	Task 3 Bluetooth: When the rover reached the target point as expected, it would send a signal through Bluetooth. To prevent signal loss, the rover was designed to stall in the task area, permitting the Bluetooth to have enough time to continue to send signals until it successfully receives the return signal ACK from the computer. Next, the rover would leave this zone and travel to the finish line.

6

Conclusions

In the course Team Design Project and Skills (TDPS), we have designed, assembled and tested an intelligent rover to accomplish six tasks on two patios. These challenging tasks range from sematic segmentation to signal to a process, during which computation efficiency and accuracy are required.

Six modules were constructed and mounted on the rover, including the main framework, a dynamic pack, a visual system, ultrasonic sensors, a tennis pitcher and Bluetooth. These modules cooperated substantially to provide a reliable and strong hardware platform for the software.

The control system is composed of innovative, effective and efficient algorithms. We proposed a novel method to segment and tracked the path in patio 1 using much less computation. In patio 2, simple yet robust shape detection and recognition algorithms were developed to recognize the pattern on standing boards as well as provide visual alignment for the rover. A complicated but resultful feedback control system was designed to control, calibrate and monitor the dynamic pack. It can also process, analyze and react to the data flow from the sensors and the visual system. As a result, our design stood out from 52 others and finished strong in the final test.

Teamwork is our core competitiveness. We established the team early in Year 3 and worked intimately throughout the project. To the best of our knowledge, we were the first team to deliver a well-constructed final design that could finish all tasks without human intervention. The influence of our project is also remarkable. We published our demonstration on social media and have received over 5000 view counts within 20 days, mostly from our colleagues and previous students.

In conclusion, we successfully delivered an innovative, reliable and efficient design with the help of early planning, intimate teamwork and incredible ideas. We are proud of what we have achieved and gone through, from which all of us learnt about the delicacy of hardware design, skills of software design, and the importance of leadership and teamwork skills.

References

[1] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 3431–3440. doi: 10.1109/CVPR.2015.7298965.

[2] M. J. Shafiee, B. Chywl, F. Li, and A. Wong, “Fast YOLO: A Fast You Only Look Once System for Real-time Embedded Object Detection in Video,” Sep. 2017, [Online]. Available: http://arxiv.org/abs/1709.05943

[3] S. Suzuki and K. Be, “Topological structural analysis of digitized binary images by border following,” *Comput. Vision, Graph. Image Process.*, vol. 30, no. 1, pp. 32–46, Apr. 1985, doi: 10.1016/0734-189X(85)90016-7.

[4] P. K. Sinha, C. F.Y., and R. E. N. Horne, “Recognition and location of shapes in the Hough parameter space,” *IEE Colloq. Hough Transform.*, pp. 11/1-11/4, 1993.

[5] S. Patel and M. Goswami, “Comparative analysis of Histogram Equalization techniques,” in *2014 International Conference on Contemporary Computing and Informatics (IC3I)*, Nov. 2014, pp. 167–168. doi: 10.1109/IC3I.2014.7019808.

[6] M. Vemula, Mó. F. Bugallo, and P. M. Djuric, “Performance Comparison of Gaussian-Based Filters Using Information Measures,” *IEEE Signal Process. Lett.*, vol. 14, no. 12, pp. 1020–1023, Dec. 2007, doi: 10.1109/LSP.2007.906214.

[7] Xin Wang, “Laplacian Operator-Based Edge Detectors,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 886–890, May 2007, doi: 10.1109/TPAMI.2007.1027.

[8] L. Guangzhou HC Information Technology Co., “HC-12 Wireless RF UART Communication Module V2.6 User Manual”.

[9] “UART: A Hardware Communication Protocol Understanding Universal Asynchronous Receiver/Transmitter | Analog Devices.” https://www.analog.com/en/analog-dialogue/articles/uart-a-hardware-communication-protocol.html (accessed Apr. 04, 2022).

[10] T. S. Limited, “3A 380KHz 28V PWM Buck DC/DC Converter TD1583”.

[11] I. Advanced Monolithic Systems, “AMS1117 1A LOW DROPOUT VOLTAGE REGULATOR”.

[12] *A4950 Motor Drive Module Manual*, 1st ed. Dongguan, Guangdong, P.R.C.: Dongguan Wheeltec Intelligent Technology, 2020.

[13] J. Wang and E. Olson, “AprilTag 2: Efficient and robust fiducial detection,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2016, pp. 4193–4198. doi: 10.1109/IROS.2016.7759617.

[14] E. Olson, “AprilTag: A robust and flexible visual fiducial system,” in *2011 IEEE International Conference on Robotics and Automation*, May 2011, pp. 3400–3407. doi: 10.1109/ICRA.2011.5979561.

Blank Page

Appendix A: External Links

1 LI Sipei’s Repository

https://gist.github.com/AmandaLiSP/8820004257b8dd0330ce7026a82b03dc

2 FENG Zihao’s Repository

https://gist.github.com/ZihaoFeng2001/c7d64f2140fee737e15ec7e298407a51

https://gist.github.com/ZihaoFeng2001/10e08c2492c9ba1a35321d930fd9a836

3 WU Haoyang’s Repository

https://gist.github.com/3rd-Musketeer/f7d428b5226598adc3e0b4ce3ddfe3f3

4 ZHENG Xiaochen’s Repository

https://github.com/y4kuyaku/TDPS_Group34

5 YANG Pei’s Repository

https://pan.baidu.com/s/1SWhOWuTf83fLvHy_tz7NDQ

Password: 1wec

6 Video Demonstrations

Patio 1 Task 1: https://b23.tv/ThVSb7i

Patio 1 Task 1-3: https://b23.tv/By8MXt8

Patio 1 Assessed Demonstration: https://b23.tv/WcSYbiV

Patio 2 Task 1-3: https://b23.tv/x3Bn10x

Patio 2 Assessed Demonstration: https://b23.tv/K6yyRFX

7 CUI Sijie’s Code

Patio 1: https://pastebin.ubuntu.com/p/DPdhTzFpBh/

Patio 2: https://pastebin.ubuntu.com/p/hvsfJMQYnt/

Blank Page

Appendix B: Financial Record

Financial record by FENG Zihao.

Category	Item	Unit Price (¥)	Number	Cost (¥)
Control	OpenMV	470.00	1	470.00
	STM32LKC	80.00	1	80.00
Dynamic	Ultrasonic Sensor	4.00	4	16.00
	Ultrasonic Sensor Holder	4.00	4	16.00
	Guid Wheel	7.70	3	23.10
	Rover Bracket	28.00	1	28.00
	Tracks and Frame	299.00	1	299.00
	Servo Motor	5.00	2	10.00
Communication	Bluetooth Module	18.80	1	18.80
PCB (with components)				22.94
Other Components				15.00
			Total	998.84

Blank Page

Appendix C: Writing Guidelines

This guideline was developed by YANG Pei under the guidance of University of Glasgow Branding Toolkit.

1 Lab Notebook Guidelines

This guideline provides a recommended template for lab notebook writing. For the purpose of assessment, it would be convenient to begin your lab notebook with:

Lab Report

Title:	Lorem Ipsum Dolor
Date:	5 March 2022
Written by:	**Read and understood by:**

For notebooks written in Microsoft Word, it is recommended that the logo of both University of Glasgow and Glasgow College, UESTC be presented on the top of each page.

University of Glasgow
VIA VERITAS VITA
University of Glasgow
2013
Glasgow College, UESTC
电子科技大学
格拉斯哥学院
Glasgow College, UESTC

For notebooks written in Microsoft Word, after you have finished writing, it is recommended to change the font into Arial or Times New Roman. A consistent typeface would be friendly to readers.

2 Final Report Visual Identity Guidelines

2.1 Headings

Level 1 Heading

Adobe Garamond, 80pts index, 42pts text, white

1

Lorem Ipsum Dolor

Level 2 Heading

Swiss 721, 16pts, 0.74cm indent, UoG Blue

1.1 Lorem Ipsum Dolor (0.74cm indent)

Level 3 Heading

Swiss 721, 14pts, 0.74cm indent, UoG Blue

1.1.1 Lorem Ipsum Dolor (0.74cm indent)

Level 4 Heading

Swiss 721 Light, 12pts, 0.25 lines before, 0.25 lines after, 0.74cm indent, UoG Blue

1.1.1.1 Lorem Ipsum Dolor (0.74cm indent)

Normal

Swiss 721 Light, 11 pts, dark gray

Lorem Ipsum Dolor Sit Amet

2.2 Colours

University of Glasgow Blue

R: 0, G: 53, B: 104, Pantone 2955 C/U
As the colour of backgrounds and captions

Dark Gray

R: 99, G: 100, B: 103, Pantone 10452 C
As the colour of normal texts

University of Glasgow Stone

R: 127, G: 116, B: 103
Pantone 403 C/U
As the colour of table borders

40%
University Glasgow Stone

As the colour of table backgrounds

20%
University of Glasgow Stone

As the colour of table backgrounds

2.3 Tables and Figures

Table 1. Example table. Use Swiss 721 Light, 11pts for table captions. Captions should be full sentences.

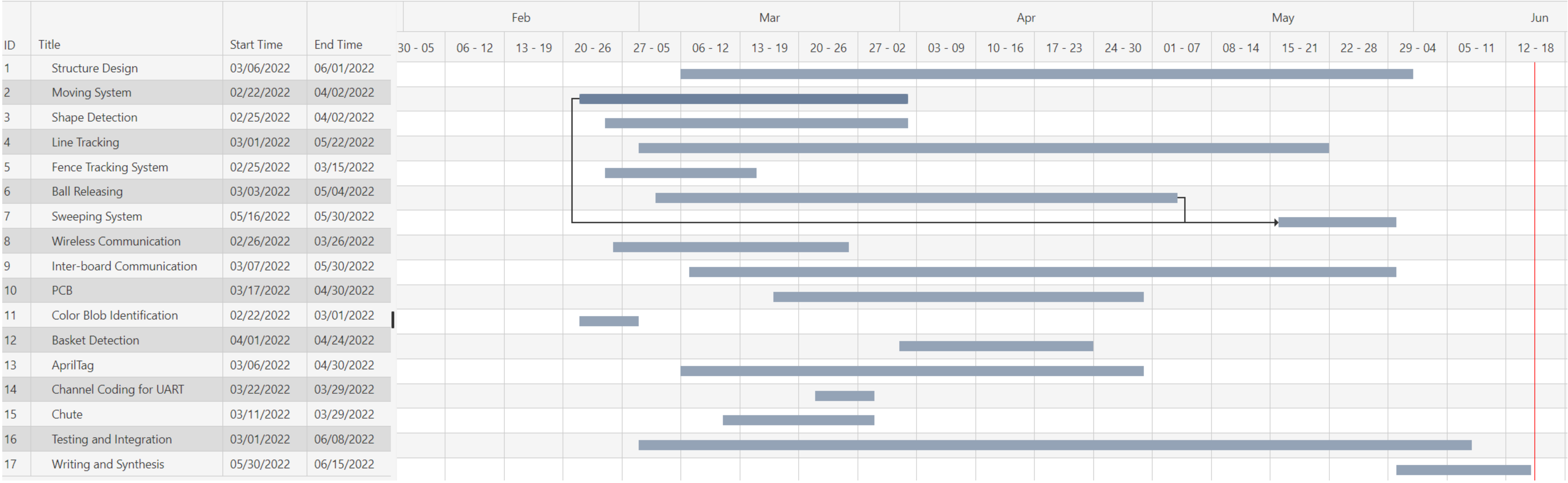
Title	Swiss 721 Light, 10pts, left aligned	Lorem Ipsum Dolor
Entries	Swiss 721 Light, 10pts or smaller, dark gray	Lorem ipsum dolor

Figure 1. Example image. Use Swiss 721 Light, 11pts for figure captions. Captions should be full sentences.

Blank Page

Appendix D: Gantt Chart

Gantt Chart developed by LI Sipei.



ID	Title	Start Time	End Time
1	Structure Design	03/06/2022	06/01/2022
2	Moving System	02/22/2022	04/02/2022
3	Shape Detection	02/25/2022	04/02/2022
4	Line Tracking	03/01/2022	05/22/2022
5	Fence Tracking System	02/25/2022	03/15/2022
6	Ball Releasing	03/03/2022	05/04/2022
7	Sweeping System	05/16/2022	05/30/2022
8	Wireless Communication	02/26/2022	03/26/2022
9	Inter-board Communication	03/07/2022	05/30/2022
10	PCB	03/17/2022	04/30/2022
11	Color Blob Identification	02/22/2022	03/01/2022
12	Basket Detection	04/01/2022	04/24/2022
13	AprilTag	03/06/2022	04/30/2022
14	Channel Coding for UART	03/22/2022	03/29/2022
15	Chute	03/11/2022	03/29/2022
16	Testing and Integration	03/01/2022	06/08/2022
17	Writing and Synthesis	05/30/2022	06/15/2022

Blank Page

TDPS 2022

#34 Team Friday