The PPT shows the basic framework of patio2. It is now up to me to describe the main problems encountered in patio2 and their solutions.

click

1. 识别箭头

In the recognition in task1, the success rate of recognition is not satisfactory and the error information may transmitted to the MCU.

click

In order to make OpenMV identify and communicate correctly with the MCU, we improve the gray-scale graphics interception algorithm to binary algorithm for recognition, which greatly improves the accuracy.

click

We then set a delay in OpenMV to ensure the Openmv doesn’t start working until the robot reaches the arrow. Besides, we extend the time standing in front of the arrow. This allows OpenMV to perform recognitions for several times, and we take the one with most occurrences among them as the result. This gives us a 100% success rate for arrow recognition.

click

2. 巡线问题

Then there are the problems during tracing.

click

2.1 走直线

In field tests, we found that the robot will deviate from the straight direction after a period. To solve this problem, we use an accelerometer to assist moving straight.

Click

This is achieved using the following algorithm.

a: The orientation coordinate value obtained by the accelerometer when it is not moving.

yaw1: The real-time orientation coordinates while it is moving forward.

Coeff: The speed difference between the two tracks.

In the code, Coeff needs to be applied to the duty cycle of the motor, so the order of magnitude needs to be set under 0.01. At the same time, in order to ensure that the robot will not turn around in place because of the large speed difference between the tracks, the value of Coeff should below 0.1. So the Coeff is calculated as follows.

In the field tests, first the car is set to stay still for a short period to obtain the initial direction and stored in a. Then, the car starts to move, and the rotation speed of the two tracks is adjusted. If it is deviated to the left of the direction of a, coeff is calculated and is added to the duty cycle of the left motor, subtracted from the duty cycle of the right motor. As a result, the left track will rotate faster and the right track will rotate slower, so the return of the robot from yaw1 to a can be realized. Vice versa.

click

2.2 走石子路

Surprisingly, the above algorithm worked perfectly when the robot is moving on the stone road. This allows it to smoothly pass the stone area and reach the bin. Moreover, after the completion of task2, the robot doesn’t need to move on the narrow flat beside the fence. Instead, it can directly move to the communication area through the stone road for task3.

2.3 沿栏杆巡线

In advancing parallel to the fence, we also used the similar algorithm as above, the difference of the orientation coordinates was changed to the difference of the distance data obtained by the two ultrasonics on the right side.