# ELEC3838 - Integrated Design Project


## AT-RS

(Automatic Tracker - Rubbish Scavenger)

| Group Member | Program | UID |
|---|---|---|
| Chen, Yiyang | ElecE | 3035953701 |
| Yang, Ziling | CE | 3036105911 |
| Zhang, Shengce | CE | 3035952721 |
| Zhang, Yilei | CE | 3035947738 |

## Part I. Project Abstract

Our project is called: "Automatic Tracker: Rubbish Scavenger," An auto-tracking, self-determined rubbish cleaner. The vehicle can roam and track its traits, performing a chain of actions that includes scanning, locating, and collecting trash. The whole system operates under the link between a computer and the vehicle. Through this link, the computer can sketch the environment and calculate both the position and posture of the vehicle, giving the car instructions on movements.

The vehicle is constructed with 4 modules. The LiDAR module is used to outline the surrounding environment. The AI recognition module lets the car label the spitball the camera captured, enabling it to find the paper ball we wish to throw away. The Robotic Arm module is responsible for grabbing the object and throwing it into the Smart Bin. The Smart Bin should be able to open up automatically when the arm holds the target and when someone's hand is close to the bin.

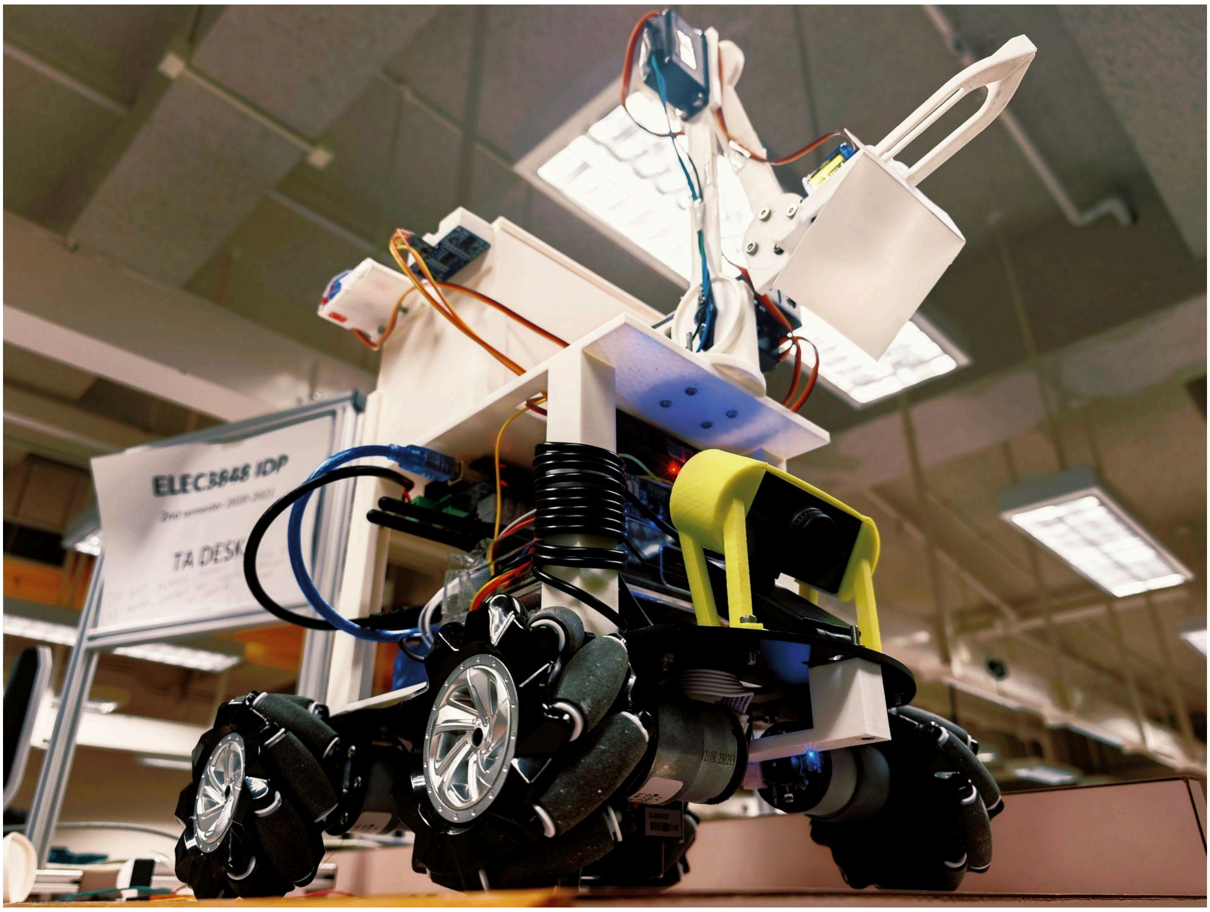



Fig.1.1. Picture of the whole vehicle

## Part II. System Description

### System Architecture

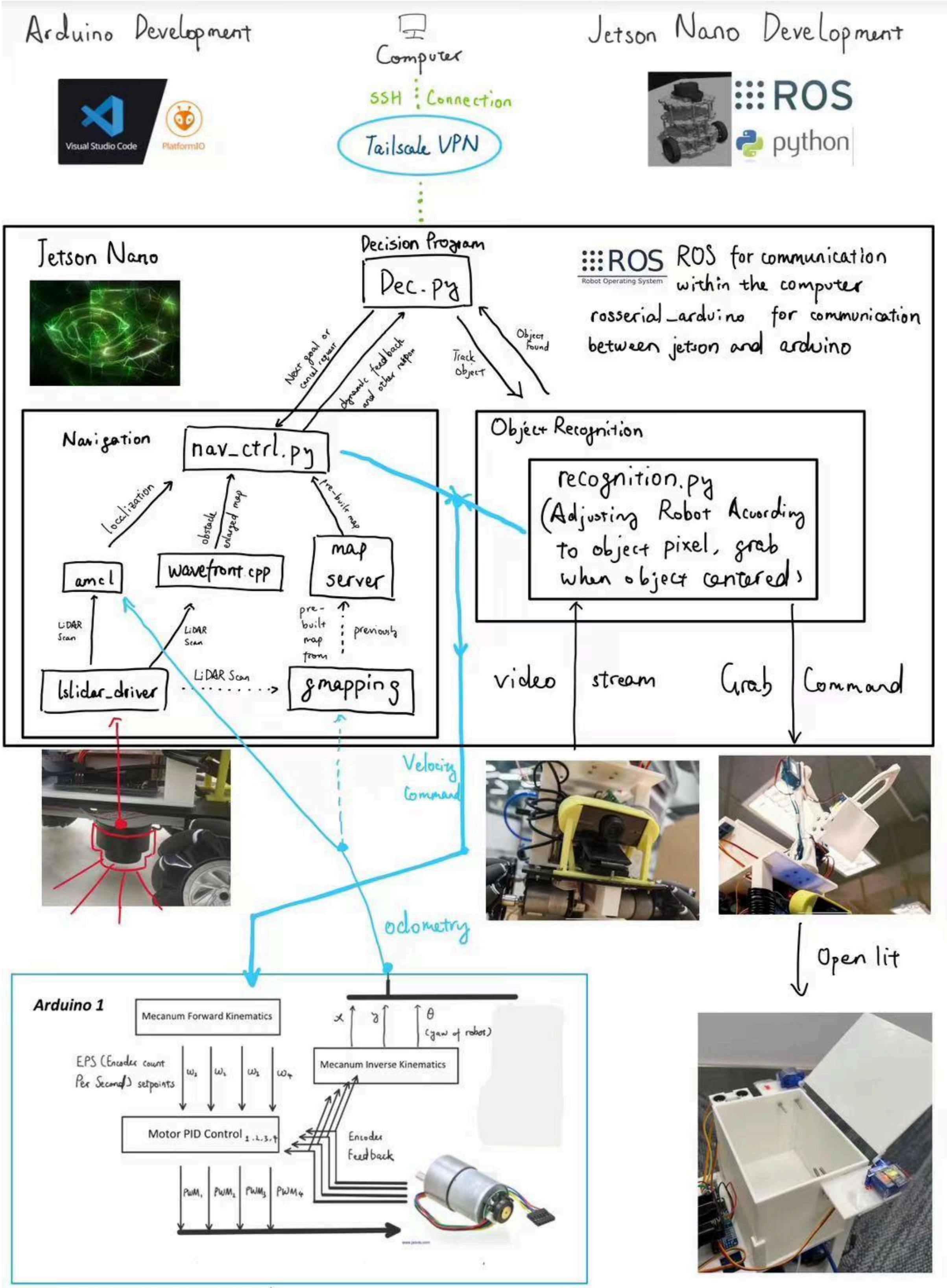
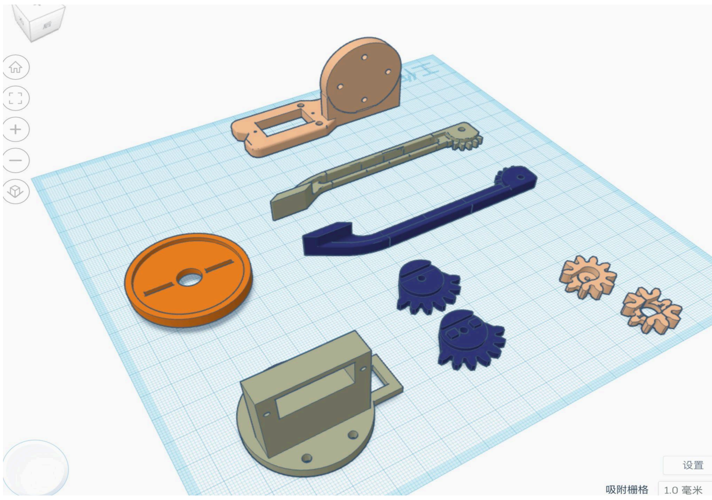



Fig.2.1. The outline of our vehicle's system architecture.

## 1. 3D printed body and robotic arm

The body and robotic arm of our robot are constructed using 3D printing technology, which allows for rapid prototyping and custom designs tailored to the specific needs of our trash collection system.

The robotic arm motor is connected to the Arduino through PCA9685, which is a 16-channel PWM output plate based on I2C communication. The power of the motor is provided by an external power supply directly connected to the PCA9685. As there are spare channels that can allow to control of servo motors is not been used, the robotic arm can be designed to be another type that could have more DOF.



F.2.2. The 3d model of robotic arm components.

Fig.2.3. The robotic arm picking up an object

2. **Chassis Control and Communication with Jetson Nano**

Kinematics and Control: Our Arduino controller performs both forward and inverse kinematics calculations. This enables precise control over robot velocity and accurate odometry calculations.

PID Controller: The wheel speeds calculated through forward kinematics are input to a PID controller. A low-pass filter smooths the feedback, ensuring robust control of wheel speed. This setup allows for precise manipulation of the robot's velocity and pose. At a higher level, engineers can simply manage the robot's x and y linear velocities, along with its angular velocity.

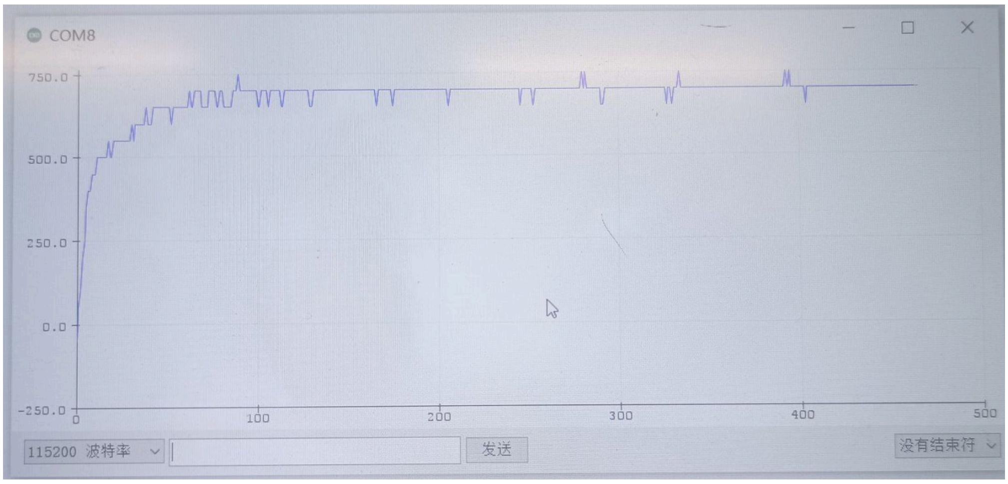



Fig.2.4. The serial plot of PID for one motor.

Odometry and Localization: Inverse kinematics, based on encoder counts, accurately track the distance traveled by the robot. While odometry alone doesn't correct movement errors, it serves as input to the Adaptive Monte Carlo Localization (AMCL) package, which provides error-correcting localization.

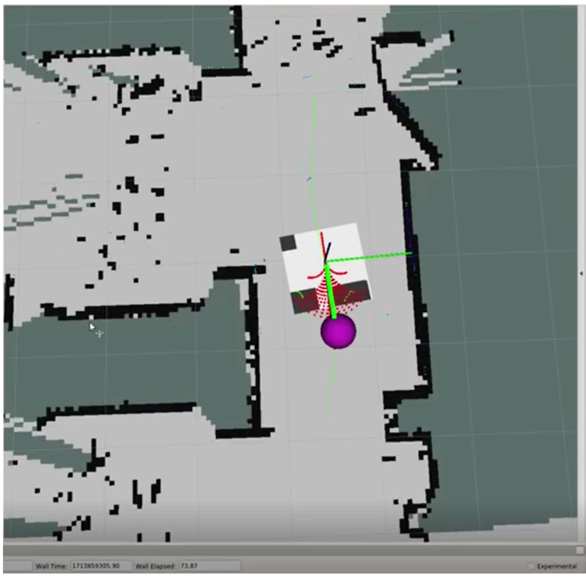
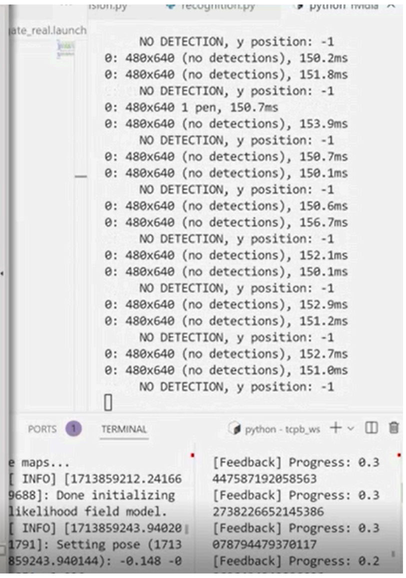
Communication: Inter-device communication is facilitated by the rosserial_arduino package([rosserial_arduino - ROS Wiki]), which significantly simplifies the integration and debugging of hardware communications.

3. **Localization, Path Planning, and Obstacle Avoidance**

AMCL Integration: Odometry data from the Arduino is transferred to the Jetson Nano and enhanced with AMCL([amcl - ROS Wiki]), using additional LiDAR data for correcting pose estimations.

Path Planning with RRT*: Although RRT* is not typically the most efficient path-planning algorithm for mobile robots due to its tendency to plan non-linear paths, this characteristic is advantageous for our patrol bot. The robot's ability to traverse non-linear paths allows it to cover more area and detect more trash effectively.

Obstacle Avoidance: This is managed through a trajectory rollout approach. The robot stores several potential velocities and extends the trajectory for each during the motion planning phase. It assesses whether any trajectory encounters obstacles or aids in progressing toward a goal. The robot selects the velocity associated with the most favorable trajectory, as determined by a predefined cost function.


(left) Fig.2.5. The mapping of the environment. The x-y-z plane represents the coordinates of the vehicle. The green line represents the path of the vehicle (both past and future).

(right) Fig.2.6. The log of the vehicle. Listing the location, time, and detecting details.

## 4. Object Detection and Recognition

The vehicle uses a camera to monitor the environment in front of it. It uses a YOLO real-time object detection program and a custom-trained model to recognize garbage. The model is trained with more than 200 photos and 300 epochs. Once the object detection program detects garbage with a high confidence score, it calculates the garbage's center point using its bounding box coordinates. After that, the program compares the garbage's center point to the center point of the camera's view and steers the vehicle to the right in front of the garbage. The program then signals the robotic arm to pick up the garbage.

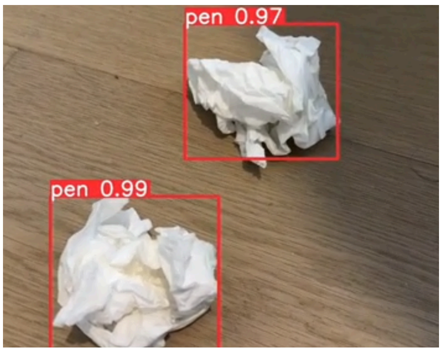



Fig.2.7. Objects with AI labeling. ("pen" is the code used for training and anti-plagiarism.)

## 5. Smart Bin

To fulfill both the requirement of automatic cleaning and daily usage, the smart bin is designed as an integration of signal control and smart detection. Two motors are merged into the bin using 3D printing to control it. The bin is controlled by the Arduino Mega 2560 board that controls the robotic arm, creating a small system that can increase its reliability and soundness.

Signal Control: When the robotic arm grabs the targeted trash, it will send a signal to the bin, which will open the bin. Similarly, when the arm drops the trash, a signal will be sent to close the bin.

Smart Detection[1]: This design is aimed at allowing users to throw their trash without the collection of the vehicle. There is an ultrasonic detector on one side of the bin, and a customized threshold will trigger the bin when it detects incoming trash or hand. The bin's cover is set to remain open for a while, and it will continually detect the user, so the bin will not close while users are still using the bin.

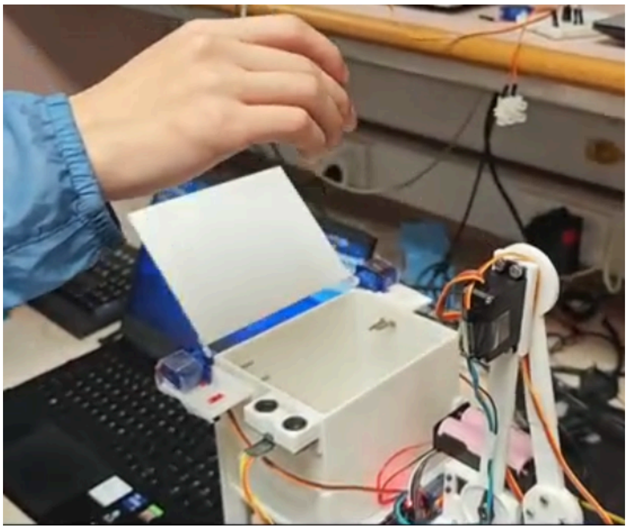


Fig.2.8. The bin automatically open when hands are close to the detector.

[1] Due to the limitation of funds, the smart design is fulfilled using lab provided ultrasonic sensor. Although the system might meet some inefficiency, it still performs with a high accuracy.

## Part III. Changes to Proposal

In our project design, the proposal is written in parallel with the feasibility test of our design. Before we merge our ideas into a project proposal, we spend time listing what we are strong at, discussing the parts we are interested in, and considering the targets of this project. While we are writing the proposal, we speak out about our ideas, we spend time researching relative fields, and we consider the risks and challenges we could face on each kind of design. After we finish the proposal, we weigh the pros and cons of the design, we review the proposed design to correct it, and we are always passionate about adding new ideas to the design.

However, we changed the project name from the previous name “Trash Collection Patrol Bot”.

## Part IV. Reflection of Project

When taking an overall perspective, the project emphasized a point for all engineers: to make creations under constraints. From the start of our project design, we start facing the limitation of the size and shape of the vehicle. The other constraint is the funding, if we wish to achieve a certain function, we can either choose to use our limited funding or to use the lab provided devices, which are lack of efficiency and have multiple uncertainties. We have to find a balance between our wishes and the actual situation, weigh the pros and cons of the use of different devices, and make compensations from other perspectives.

One of the trouble is that PCA9685 connection cause a current recoil, which will burn the MEGA2560 board and influence the Jetson nano, the problem is solved by connecting a external power supply to PCA9685 while disconnect the port that supplies power to the servo motor. This problem is covered by manual instruction.

Due to the insufficient hardness of the 3d printed material, the full expansion of the robotic arm is fairly length, and the torque of servo motor is not to strong, so it will cause vibration during the operation of the robotic arm, we solve this by extending the overall working time of the robotic arm in order to decline the angular momentum so that the stop of robotic arm when the motor is turn into a certain angle will not bring a huge vibration to the robotic arm and car platform.

## Part V. Demonstration Video

The link of the demonstration video:

https://1drv.ms/f/s!As8KODtqeUsvijHGKQyp-fmgi7iz?e=4LfqFo