Autonomous Litter Detection and Recovery System

Malek Hammoud[a]

[a]*London Central Secondary School, London, Ontario, Canada*

Abstract

The issue of public littering and waste disposal is a major environmental, visual, and societal concern. Urban waste management often relies on manual cleanup, which is time-consuming and challenging. This paper proposes a low-cost proof-of-concept solution designed to transform urban cleanup through automated waste detection and collection. The system leverages computer vision and robotics to accurately identify and retrieve litter from public spaces. It is built using a Raspberry Pi for processing, ROS and microcontrollers for robotic control, and an array of sensors—including magnetometers and distance sensors—for precise navigation and obstacle avoidance. This concept demonstrates the potential for autonomous drone solutions to streamline urban waste management, reduce manual labor, and foster cleaner, more sustainable environments.

Keywords: ROS, Microcontrollers, Computer vision, Sustainability, Littering

1. Introduction

Recent advancements in drone technology have led to many unfulfilled opportunities in the field of Autonomous robotics. Robotics and the outdoors are often overlooked fields. There are still so many problems to be solved in this field of work. One of those problems is litter. Litter gets into the waterways and poisons drinking water. It also washes up into the sea where it gets broken down and then ingested by animals. Human tissues have been shown to have microplastics in them. All of these problems caused by litter can be mitigated if a widespread litter cleanup is done. Studies have shown that a clean environment will reduce the likelihood of a person littering. This proves that cleaning up litter does work; we just need to put in the effort once. Human labour is very expensive, and an autonomous litter cleanup process is required. This paper will be covering a technological advancement in how we think about drones and robotics. This paper covers the design and implementation of an outdoor drone robotics system that will be used to identify and pick up litter. Although there were many projects focused on robot systems designed to clean litter, and drone systems designed to find litter, they were never put together into one system until now.

2. Methodology

The design process began with a thorough analysis of system requirements, leading to the formulation of a robust architectural framework. The system requirements were as follows:

- Must be safe to be used in public spaces
- Efficient detection of litter
- No human intervention
- Picking up litter and storing it on board

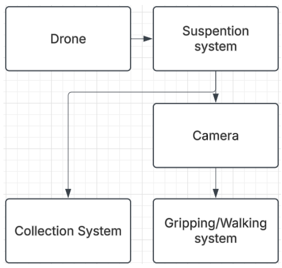


Figure 1: The hardware setup

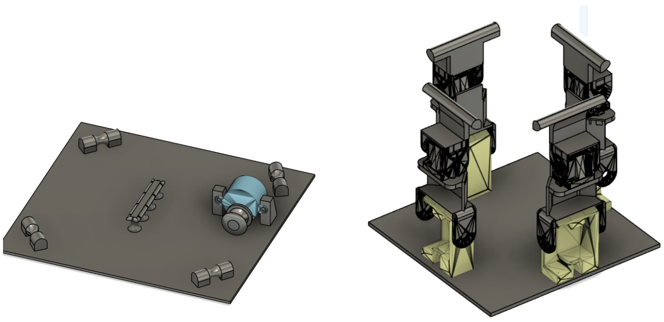
Based on those system requirements, I started planning the design on paper. An autonomous drone system was chosen as it's the fastest and most efficient. However, the spinning blades on a drone are not safe. To combat this issue, a drop-down approach was chosen. The drone can now be 10–15 m above the ground, while a robotic system is dropped down to pick up the litter. This approach also avoids the problem of having the litter blown away by the wind. The robotic system is a walking gripper, able to walk and grip items. To collect litter, a separate collection system is used. Refer to Figure 1 for a visual on how everything fits together.

There are 4 parts to this system:

- The drone.
- The Gripper
- The Suspension system
- The Collection system

The drone was put together using the F450 drone frame. Refer to the materials chart for more details on the components.

The gripper was built using servo motors. It was assembled using small screws.



(a) The Suspension System (b) The Gripping System

Figure 2: Fusion 360 Designs

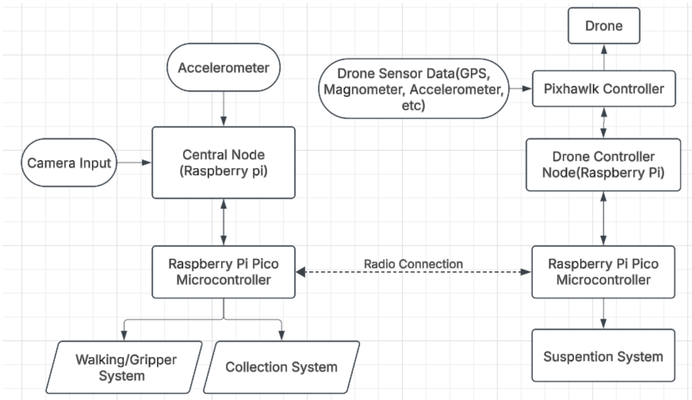


Figure 3: Node Graph

The suspension system is powered by a DC motor with an encoder. Pulley ball bearings were put in a certain structure to achieve the lowering motion.

The collection system is still in development, however the proposal is to build it using a stepper motor and pulleys. A small bag is attached to a string to open and close the bag. Tension would be powered by gravity.

Figure 2 shows the designs for the gripping and suspension system. They were designed in Fusion 360 and printed on an Ender 3 V3 3D printer.

For the software, ROS2 was used for its structured and modular framework. ROS: an open-source Robot Operating System is a system designed to handle complex interactions between software and hardware. A central node system was used. A powerful computer is the central node; this communicates with all the other nodes. This centralized approach allows for easier processing of all the information, especially when more intelligent algorithms are needed. This way we can reduce the overall cost of the system by having only one powerful computer, rather than four. The following diagram depicts the structure of the nodes:

Microcontrollers were used, for several reasons. It allows offloading processing from the Raspberry Pi. The other reason was that the microcontrollers are better designed to handle faster, lower-level communication with electronic circuits. Although it can be done with just the Raspberry Pi, certain things like PWM are way better if they are controlled through lower-level hardware (aka hardware PWM). Sensors, magnetometers, and GPS were used for the.

2.1. *The Drone Software*

The drones used ArduPilot, which is a drone controller software used for controlling the drone's stability, direction, speed, etc. This runs on the Pixhawk controller. The PID algorithms come with ArduPilot, so it's very easy to get started with DIY Drones. However, the Pixhawk controller is not powerful enough to handle running computer vision algorithms like YOLO. The Pixhawk is designed to excel at drone control, but nothing else. To solve this issue, the Raspberry Pi 4 was introduced. The Raspberry Pi 4 is a powerful single-board computer that can handle computer vision tasks. I connected the Raspberry Pi 4 and the Pixhawk using a UART connection. A UART connection is a way for two devices to talk to each other. There is an input and output; the Raspberry Pi can both send and receive data from the Pixhawk. The software I used to handle that communication was PIMAVPROXY and PIDRONEKIT. PIMAVPROXY was used as the main connection software. This handles all the computation required for proper communication. However, communication alone is not enough; this is when PIDRONEKIT is introduced. PIDRONEKIT is a Python package that allows the Raspberry Pi to send commands to the Pixhawk using Python. Both software packages are equally important and were essential to this project.

2.2. *Walking Algorithm*

The walking algorithm is simple. A GUI input was used to allow me to properly find the most efficient process I can use to walk and pick up litter. This was done using Tkinter. The input node ran Tkinter, got input from the GUI, and passed it to the output node which controlled the motors. Once I understood how I could control the joints to properly walk and pick up stuff, I recorded the joint angles and stored them in a list. Then, to walk, the algorithm would simply repeat that pattern. For example, if the starting position of the angle values is [90,90,90,90] and to lift one leg it's [45,90,90,90]. The algorithm would simply loop through all the values between 90 and 45 in an amount of time specified to achieve that motion.

2.3. *The Computer Vision Algorithm*

Good computer vision requires a good dataset. The photos in that dataset were taken by a DJI drone. This ensures that the pictures are relevant to the actual use case: my DIY drone. I made sure that the dataset pictures were exactly the same by comparing pictures taken by my drone and the dataset. They matched perfectly. YOLOv5 is designed to be fast, accurate, and easy to use, making it an excellent choice for a wide range of object detection, instance segmentation and image classification tasks. (Jocher, 2020) Training was done on my laptop (an Acer Swift 3) and took 4 hours. No data augmentation or processing was required on this dataset.

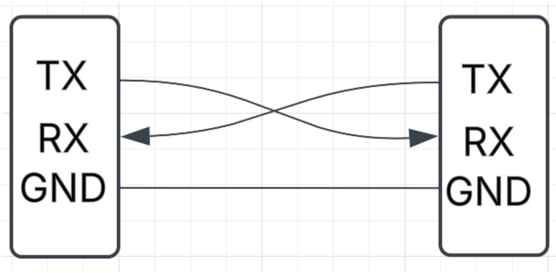


Figure 4: UART connection between Pixhawk and RPI

Weight (in meters)	250 Grams Oscillation	500 Grams Oscillation
0.2M	Minimal	Minimal
2M	Manageable	(not tested)
4M	Too much	(not tested)

Table 1: Load testing results

3. Results

3.1. The Drone/Hardware

Testing was done to validate system performance under various operating conditions. The system was first tested indoors, but to ensure the highest level test, certain systems were tested outdoors with varying conditions.

The drone was tested outdoors with varying load. The results were surprising. The drone was able to achieve an incredibly high amount of load; testing was stopped after a certain threshold to prevent damage to the system.

In Table 1 are the results for varying loads and lengths of lines:

The GPS was also tested. Loitering and landing were very stable. On average, the GPS was accurate to 1 meter.

3.2. The Walking/Gripping Algorithm

The walking algorithm's results were not promising. The algorithm was able to achieve a walking speed of 3 cm/second. More could have been done to increase this speed. However, it's important to note that the gripper was incredibly stable the entire time; there were no cases where the algorithm caused the robot to tip over.

3.3. Computer Vision Results

The computer vision results were very promising. An accuracy of 90% was achieved. It was also tested in horrible video conditions (without a gimbal) and was able to detect litter in an input video without any problems.

4. Discussion

The load testing of the drone resulted in interesting results. The drone was able to properly handle any load up to half a kilogram without any problems. Testing was stopped to avoid any problems with the hardware. More testing needs to be done;

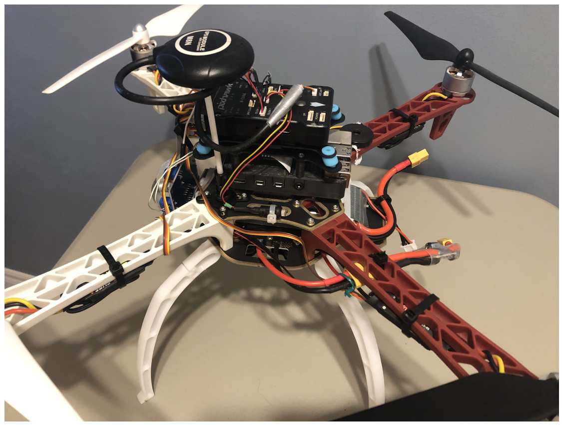


Figure 5: The drone

however, these tests show that the actual load a drone is capable of is very different from the specified load. However, it is always safer to stay within the specified max load as the motors can easily heat up and break in extreme load conditions.

The GPS was also tested; the results show how a cheap GPS can compete with industry standards. Most of the GPS's on the market are overvalued; there are cheap GPS's that can do the job perfectly well.

The walking algorithm's results were less satisfactory; the speed of 3 cm/second is just too slow. However, it is important to note that to achieve great accuracy and stability, one must sacrifice speed. A reinforcement learning approach must be investigated to potentially discover better ways of walking on this hardware automatically.

The computer vision received promising results. The results show that CV on drones to detect litter is very much possible. However, a variety of distances is not possible. More needs to be done with regard to creating multiple models at different heights and running each of those models at said heights. This approach will allow the drone to adapt on the fly, and always be able to detect the litter.

5. Conclusions

In summary, this paper covered the development of the drone and robotics system that is capable of detecting and locating litter autonomously. Future work is needed to optimize various subsystems in this project; however, results were promising and a system like this can be built. This system will one day keep our highways, parks and streets clean.

Acknowledgements

Thank you to Mateusz Piechocki for his insights on my CV algorithm.

Thank you to friends and family who were cheering me on the sidelines.

Appendix A. Materials List

Material	Description
Acer Swift 3 Laptop	Laptop for development, design, and control tasks
Raspberry Pi 4 (2GB)	Single-board computer serving as the main processing unit
Raspberry Pi Pico	Microcontroller board for handling peripheral controls
Raspberry Pi Camera Module	Camera for image capture and computer vision tasks
6V 1A Ni-Mh Battery	Rechargeable battery for low-power applications
12V 60C LiPo Battery	High-discharge battery for power-demanding systems
Relay Module	Electrically controlled switch for circuit isolation
6V DC Motor Metal TT 1:90	Motor for mechanical actuation in low voltage applications
Micro Servo Motor (SG90)	Compact servo for precise, small-range motion control
Large Servo Motor (MG946R)	High-torque servo motor ideal for robust motion control tasks
F450 Quadcopter Frame	Drone frame for constructing a quadcopter platform
4 x 9450 Propellers	Propellers designed to provide lift for the quadcopter
4 x 2212 920kv Motors	Brushless motors for driving the drone's propellers
4 x 30A Simonk ESC	Electronic speed controllers for regulating motor speed
Pixhawk 2.4.8 Flight Controller	Advanced controller for autonomous flight management
M8N GPS Module	GPS receiver for navigation and positioning
Power Modules	Units for voltage regulation and power distribution
Breadboards	Prototyping boards for assembling temporary circuits
Electrical Wires	Dupont wires and Copper Flexible 22 gauge wires
Cardboard	Lightweight material for prototyping and support
3D Printed Parts	Custom components produced via 3D printing
Metal Bolts, Nuts, Washers, Brackets	Fasteners and hardware for structural assembly
Ender 3 V3 3D Printer	3D printer used for fabricating prototypes and parts
General Tools	Essential hand tools like pliers, tape, screws, screwdrivers, etc.

Table A.2: Materials List

References

Amatucci, L., Turrisi, G., Bratta, A., Barasuol, V., Semini, C., 2024. Vero: A vacuum-cleaner-equipped quadruped robot for efficient litter removal. Journal of Field Robotics 41, 1829–1842. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.22350, doi:https://doi.org/10.1002/rob.22350, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.22350.

Brady Jeong, Amber Lindberg, G.R.M.T., n.d. Marea Verde Proposal. Technical Report. Worcester Polytechnic Institute. URL: https://web.cs.wpi.edu/~rek/Projects/Marea-Verde_Proposal.pdf. accessed: 2025-03-02.

Cleaners", T.S., . "bebot - beach cleaning robot". URL: https://searial-cleaners.com/our-cleaners/bebot-the-beach-cleaner/.

CNN, 2021. Drone technology aids in combating plastic pollution. CNN URL: https://www.cnn.com/2021/06/23/europe/ellipsis-drone-plastic-pollution-c2e-spc-intl/index.html. article published on June 23, 2021.

Elijah Almanzor1, Nzebo Richard Anvo1, T.G.T.F.I.I., 2022. Autonomous detection and sorting of litter using deep learning and soft robotic grippers. Frontiers in Robotics and AI URL: https://www.frontiersin.org/journals/robotics-and-ai/articles/10.3389/frobt.2022.1064853/full, doi:10.3389/frobt.2022.1064853.

Hiromichi Yamamoto, Rana Abdel Sattar, J.W.L.F.M., 2016. Operation 2nd Place: Rakas. Technical Report. Transportation Research Board. URL: https://onlinepubs.trb.org/onlinepubs/acrp/docs/UniversityDesign/pdfs/CompetitionWinners/2016-Operation_2nd-Place_Rakas.pdf. competition document.

Jocher, G., 2020. Yolov5 by ultralytics. URL: https://github.com/ultralytics/yolov5, doi:10.5281/zenodo.3908559.

Knight, A., 2016. Drone-robot technology aiding litter capturing efforts along great lakes waterfronts. Tribune-Star URL: https://www.tribstar.com/news/local_news/drone-robot-technology-aiding-litter-capturing-efforts-along-great-lakes-waterfronts/article_b063e222-2a71-11ef-86cc-47fd7102874a.html. accessed: 2025-03-02.

Kraft, M., Piechocki, M., Ptak, B., Walas, K., 2021. Autonomous, onboard vision-based trash and litter detection in low altitude aerial images collected by an unmanned aerial vehicle. Remote Sensing 13. URL: https://www.mdpi.com/2072-4292/13/5/965, doi:10.3390/rs13050965.

Team, T.R.D., n.d. Ros 2 concepts: Basic. URL: https://docs.ros.org/en/iron/Concepts/Basic.html. accessed from ROS 2 Iron documentation.

The Washington Post BrandStudio, . Tackling trash in our nation8217;s green spaces. URL: https://www.washingtonpost.com/sf/brand-connect/subaru/tackling-trash-in-our-nations-green-spaces/.

Works, W.P., n.d. Who Litters and Why: People Litter. Technical Report. Wycokck Public Works. URL: https://www.wycokck.org/files/assets/public/v/1/public-works/documents/who-litters-and-why-people-litter.pdf. accessed: 2025-03-02.