T.R.A.S.H.

Tracking Recyclables Automated Sorting Hand

Jayden Lumas, Tristan Koopman, Sayed Almousawi, Dylan Bowers

College of Engineering and Computer Science, University of Central Florida, Orlando, FL, 32826, USA

***Abstract* — This paper presents T.R.A.S.H., a robotic arm system for automated recycling. By integrating computer vision and spectroscopy, the system identifies and sorts recyclable materials with high accuracy. Using a Raspberry Pi and ESP32 microcontroller, it performs real-time detection and actuation without external computing. Designed for affordability and scalability, T.R.A.S.H. offers a practical solution for small recycling operations seeking automation in waste management.**

***Index Terms* — Automation, computer vision, machine learning, robotics, embedded systems, spectroscopy**

I. Introduction

Recycling is an essential part of reducing environmental impact and promoting sustainability, but the process is still heavily dependent on manual labor. This leads to challenges like high operational costs, labor shortages, and inconsistent sorting. While large-scale recycling centers have begun using robotic systems to help with sorting, these solutions are often expensive, bulky, and not practical for smaller operations such as schools, small businesses, or local municipalities.

To help close this gap, we designed T.R.A.S.H. (Tracking Recyclables Automated Sorting Hand), a low-cost, compact robotic arm system that can automatically detect, identify, and sort recyclable materials. It combines computer vision and spectroscopy to identify objects by both shape and material reflectance. The system is fully embedded, using a Raspberry Pi 5 and ESP32 microcontroller for real-time processing and control without the need for external computing resources.

This paper covers the full design and development process of T.R.A.S.H., including its electrical, mechanical, and software subsystems. We also discuss the testing results and challenges faced during development, showing how T.R.A.S.H. can be a practical and scalable solution for improving recycling accessibility and efficiency in smaller settings.

II. Engineering Requirements

The T.R.A.S.H. system is engineered to bring automation and precision to small-scale waste sorting applications. Each design choice was made with the goal of improving recyclability rates while keeping costs low and implementation simple. The following subsections outline the engineering requirements that guided the system's design.

A. Object Detection and Positioning

T.R.A.S.H. is capable of detecting recyclable items such as plastic bottles and aluminum cans using a combination of computer vision and optics. The system must identify an object's position within a workspace with at least 80% accuracy and locate up to five items simultaneously. The computer vision system also calculates the object's coordinates, which are then used to generate joint trajectories for the robotic arm.

B. Material Identification

In addition to object recognition, T.R.A.S.H. employs a reflection-based spectrometer using a linear CCD sensor to determine the material type. The system must correctly classify at least two types of materials (e.g., PET plastic and aluminum) with a minimum identification accuracy of 80%. Stretch goals include expanding detection to four or more materials and reaching 90% classification accuracy through machine learning.

C. Motion Precision and Payload

The robotic arm has six degrees of freedom (6DOF) and must be able to position the end effector with a tolerance of ±5 mm. The system must reliably lift items weighing up to 100 grams under the base requirement, with a stretch goal of handling 1 kg payloads. The end effector is designed to grip and place identified recyclables into the appropriate bin with minimal positional error.

D. Operational Speed

The system is designed for reasonable throughput in small-scale operations. The total time from detection to placement should not exceed 30 seconds per item. Improvements in classification algorithms and motor response aim to reduce this to 20 seconds for the advanced version and 10 seconds in optimal scenarios.

E. Power Consumption

T.R.A.S.H. is optimized for efficiency in power use. Under full load, the system must operate with a maximum power consumption below 350W, supplied via a standard

120VAC input. Power regulation is handled via a buck converter system that provides both 5V and 3.3V logic rails.

F. User Interface and Interaction

The system integrates a touchscreen display, offering a real-time view from the onboard camera and allowing users to override the sorting logic or view system status. The interface must respond within 1 second of touch input and provide visual confirmation of material classifications.

III. System Components

The T.R.A.S.H. system comprises several integrated subsystems that work together to identify, classify, and sort recyclable materials. These components were selected for their cost-efficiency, performance, and compatibility with the system's goals. The following subsections outline the major hardware elements.

A. Microcontrollers

Two microcontrollers were employed in the system: the ESP32-S3-WROOM-1-N8 and the STM32F401. The ESP32-S3 serves as the primary control unit for the robotic arm, interfacing with the stepper and servo motors. It offers ample computational power and supports a wide array of peripherals, including multiple UART and I2C channels. The ESP32 also provides built-in Wi-Fi and Bluetooth, which, while not central to the T.R.A.S.H. functionality, allow for future wireless expansion.

The STM32F401, a 32-bit ARM Cortex-M4 MCU, was chosen to drive the spectrometer's linear CCD sensor (TCD1304DG). This microcontroller was selected for its multiple advanced timers, DMA support, and high-speed ADC, which are critical for generating precise timing signals and capturing CCD output. The STM32's hardware IDE van be used to configure and meet the timing constraints of the linear CCD Sensor.

B. Single Board Computer

A central component of the system's electrical architecture is the Raspberry Pi 5, which serves as the primary processing unit for both computer vision and system coordination. Equipped with 8 GB of RAM and significantly improved CPU and GPU performance over the Raspberry Pi 4, the Pi 5 can efficiently manage high-throughput tasks without requiring external accelerators like the Coral TPU. It also runs the spectrometer control scripts, allowing it to process spectral data in parallel with image inference. Furthermore, the Raspberry Pi 5 handles UART communication with the motor controller, sending positional and classification data to orchestrate robotic arm movement. Its compatibility with the Arducam Camera Module 3 through the native CSI interface eliminates the need for USB webcams, reduces latency, and enhances visual performance. Altogether, the Pi 5's robust processing capabilities and multi-role functionality make it an ideal choice for tightly integrated embedded systems like T.R.A.S.H.

C. Motors and Drivers

T.R.A.S.H. utilizes a combination of NEMA 17 stepper motors and high-torque servo motors. The NEMA 17 motors are paired with MKS Servo42C smart drivers, which feature closed-loop feedback via onboard encoders, allowing for precise joint control. Servo motors were used for joints that doesn't need to handle heavy weight (end effector joints). This hybrid configuration provides a balance of strength and responsiveness across the robotic arm's six degrees of freedom while keeping costs low.

D. Spectroscopy Sensor

For material classification, the system integrates the TCD1304DG linear CCD sensor, driven by the STM32F4. This sensor enables reflection-based spectral analysis of materials by capturing the light intensity across a wavelength range of 400–1000 nm. Paired with a collimated LED light source and diffraction optics, the CCD can distinguish between common recyclables such as aluminum, PET plastic, and glass, based on their spectral fingerprints.

E. Camera

The system uses the Raspberry Pi Camera Module V3, which provides a 12MP image sensor and hardware autofocus. The camera captures real-time images of the workspace for object localization and shape detection. Image data is processed by the Raspberry Pi using a YOLO-based neural network trained to detect recyclable items.

F. Power System

The power subsystem includes a 12V AC-DC supply, two LM2678 buck converters to generate 5V, and AMS1117 two LDO regulators for 3.3V logic levels. Additionally there is a 4V LDO used to power the CCD sensor only. This Design Ensures stable voltage rails for motors, logic circuits, and sensors while minimizing power loss, noise interference and also providing enough current for our system components.

IV. System Overview

The Tracking Recyclables Automated Sorting Hand (TRASH) is a semi-autonomous robotic system developed to identify and sort recyclable materials. The system integrates a 6-DOF robotic arm, a camera, and a spectrometer to perform object detection, material classification, and pick-and-place operations. As items are presented, the vision system locates the object, and the spectrometer evaluates its material properties. A decision-making algorithm determines whether the object is recyclable. If it is, the system computes a trajectory and commands the robotic arm to deposit the object into the appropriate bin.

The project workload is divided into four main subsystems: mechanical, electrical, software, and sensing. The mechanical subsystem includes the arm structure and a 3D-printed end-effector. The electrical subsystem manages motor drivers, microcontrollers, and power delivery. The software subsystem, initially developed in ROS2, handles perception, motion planning, and motor control. Finally, the sensing subsystem integrates OpenCV-based computer vision with real-time spectrometry data for classification. These components communicate through a centralized ROS2 framework, allowing for modular development and future scalability. The TRASH system demonstrates a practical application of robotics in waste management, aiming to reduce manual labor and improve recycling throughput.

Figures 1 and 2 show the high-level hardware and software block diagrams of the TRASH system, highlighting how components are connected and how data flows between subsystems.

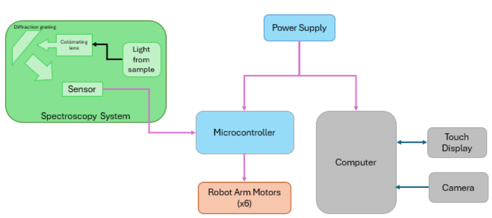

Figure 1: System hardware diagram.

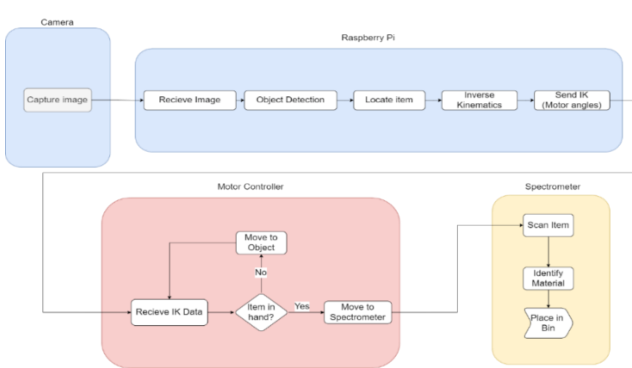

Figure 2: High-level Software Block Diagram

V. Mechanical System

The TRASH system utilizes a 6-degree-of-freedom robotic arm composed entirely of FDM 3D-printed components, with the exception of one high-stress joint connector. The base of the arm is actuated by a NEMA stepper motor connected to a custom gear train, enabling full rotation of the arm around the vertical axis. Mounted on the final gear of this train is the first joint motor, which actuates the first arm link via a high-torque cycloidal drive. This drive provides high torque and minimal backlash, enabling stable, precise motion of the shoulder joint. The third motor actuates the second link directly and has presented a notable mechanical failure point in testing: the 3D-printed connector transferring torque from this motor to the link repeatedly deformed under load. To resolve this, a custom-machined aluminum replacement is being fabricated to ensure durability and long-term reliability.

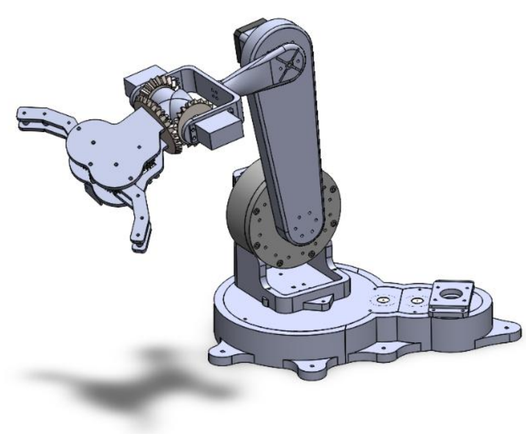

Figure 3: Robot Arm Design

The distal section of the arm incorporates a differential wrist, driven by the fourth and fifth servo motors, which allows for simultaneous and decoupled control of tilt and twist at the end effector. The end effector is mounted to the bevel gear output from the wrist and is actuated by a sixth servo motor. This gripper uses a gear-based mechanism with one central drive gear and two idler gears connected to shaped, pivoting links. The link design enables conformance to a variety of can sizes, ensuring secure and adaptive grasping. With the exception of the critical aluminum connector, all structural and functional mechanical components—including links, joints, and the gripper—are 3D printed. The entire assembly is mounted to a stable wooden baseplate for integration with the electrical and sensing subsystems.

I. SPECTROMETRY SYSTEM

The purpose of a Spectrometer in the project is for material identification. In a real worth application of TRASH. Item will most often be deformed uniquely. A spectrometer is not impacted by such deformities unlike computer vision.

The most robust method for spectroscopy for our application is Laser Inducted Breakdown Spectroscopy LIBS. For a laser to create plasma for most materials requires a nano second pulse laser. Due to the cost of such a laser system as well as safety concerns. I decided not to move forward with a LIBS based system. The system I have decided to develop is the reflection-based spectroscopy system.

Reflection-based Spectroscopy is a technique that analyzes the spectrum of light reflected from a sample to determine its properties. Typically, a broadband light source is incident on the sample. However, to have a more modern light source. An array of light emitting diodes were used to over the visible spectrum 400 nm to 700nm. The reflected light interacts with the samples' surface some wavelengths will be absorbed while others are reflected dependent on the material properties. For spectral analysis the reflected light is collected, and the intensity of the light is measured based on wavelengths. The spectral data provided a fingerprint of the sample. Data Interpretation provides a way to classify the spectrum of an unknow sample to reference spectra.

Now more details about the final assembled design. For the Original design I had an entrance slit of 10 micrometers; however, not enough light would make it into the system. To ensure enough incident light would make it into the system we started to use a ¼ inch circular entrance aperture. The new aperture allows enough light to enter the system.

For the final design I chose 2 achromatic lenses. The first lenes 1" diameter with 80mm focal distance. The second lens is 2" diameter with 100mm focal distance. The resulting system magnification is 0.8. During the original design phase I was trying to do a full system design with just B7K lenses; however, the difference in lateral shift do to the difference in color. Lead to a defocused image across the spectrum. That's why the final lens chose achromatic lenses.

II. ELECTRICAL AND HARDWARE DESIGN

The electrical and hardware design of the TRASH system brings together the subsystems responsible for power delivery, motor control, sensing, and system communication. These components are integrated through a custom PCB, ensuring reliable operation. An overview of the full system can be seen in the hardware design summary (Figure 4) which highlights the power and data connections between all subsystems.

A. Power Subsystem

The power subsystem manages voltage regulation and distribution throughout the system. A 120V AC wall input is converted to 12V using a commercial AC-DC converter. A buck converter LM2678 then steps this down to 5V for motors and servos, while a linear regulator AMS1117 produces a clean 3.3V rail for microcontrollers and a LT1761 produces 4V for the CCD sensor.

B. Motor Control Subsystem

The ESP32-S3 handles all motor-related tasks. It communicates with MKS Servo42C stepper drivers via UART2 and controls servos through a PCA9685 PWM driver over I2C. The Raspberry Pi sends high-level joint commands to the ESP32 over UART1.

C. Spectrometer Subsystem

This subsystem is controlled by an STM32F401 dev board, which drives the TCD1304 linear CCD using multiple timers and collects analog data using its ADC with DMA. The STM32 operates on a dedicated 3.3V rail and transmits captured data to the Raspberry Pi over USB (CDC). The TCD1304 sensor is driven by a specific driving and amplification circuit that is provided in the datasheet.

Additionally an LED circuit was added later on during the project to provide a light source for the spectrometer system which is needed to capture reflected light off the materials. A custom circuit is designed to integrate specific LEDS to cover the wavelength required followed by a MOSFET acting as a switch to control the LEDs using the microcontroller.

D. Control and Communication Subsystem

A Raspberry Pi 5 acts as the system's central processor. It handles image-based object detection and material classification, then sends movement commands to the ESP32 and receives spectral data from the STM32. This separation between high-level processing and low-level actuation allows for better modularity and task specialization.

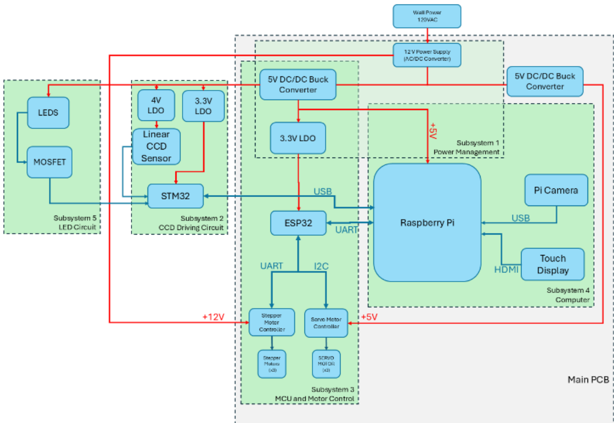

Figure 4: Hardware Design Summary Block Diagram

III. Software Architecture

The software architecture of the T.R.A.S.H. system is organized into three coordinated stages that reflect the system's perception, control, and sorting operations. The Raspberry Pi 5 serves as the central processor, running all core programs responsible for computer vision, inverse kinematics, spectrometer control, and system-wide coordination. The process begins with the Arducam Camera Module 3, which captures a live image of the workspace. This image is then processed by the Raspberry Pi using a custom-trained YOLOv8 object detection model to locate recyclable items and determine their position within the frame. The system then calculates the required inverse kinematics to determine motor angles needed for reaching the detected object.

Once the inverse kinematic data is calculated, it is transmitted via UART to the ESP32-based motor controller, which manages all low-level motion control. The robotic arm moves to the identified object, attempts a grasp, and confirms whether the object is securely held. If the grasp fails, the system loops back and reattempts the pick-up. Upon successful grasping, the arm transitions to the spectrometer station, where the object is scanned to confirm or correct the material classification previously inferred by the vision model. This layered sensing approach ensures more accurate identification, especially in visually ambiguous cases.

Following spectrometer analysis, the final material type is sent back to the Raspberry Pi, which determines the correct bin destination for the item. A new command is sent to the motor controller with updated coordinates, prompting the arm to move and release the object into the appropriate bin. The overall system structure, as visualized in the accompanying architectural diagram, demonstrates clear modularity between vision, control, and material analysis, all while maintaining tightly integrated communication loops. This design allows T.R.A.S.H. to operate autonomously, efficiently coordinating perception and motion in a closed-loop manner.

IV. Computer Vision

The T.R.A.S.H. system utilizes a custom-built computer vision pipeline to autonomously identify, classify, and localize recyclable materials in real time. This pipeline is central to the robot's ability to operate without human guidance and make informed sorting decisions. To support edge inference, a Raspberry Pi 5 was selected as the primary onboard processor. Its enhanced CPU, GPU, and RAM capabilities allow it to efficiently handle modern machine learning workloads, eliminating the need for external accelerators and making it ideal for embedded vision tasks.

The object detection model is based on YOLOv11 (You Only Look Once), the latest version from Ultralytics, known for its speed and improved accuracy. The model was trained to detect five recyclable categories: plastic, glass, metal, cardboard, and paper. To build a high-quality dataset, over 3,000 labeled images were collected, combining samples from the TrashNet dataset with custom photos taken using the Arducam Camera Module 3. These images include a mix of isolated and cluttered scenes captured in diverse lighting conditions and from multiple perspectives. This variability allowed the model to generalize effectively across real-world environments (see Figure 11).

Figure 5: Sample training batch used to teach the YOLOv11 model, showing labeled recyclable items.

Figure 6: Validation set (left) and corresponding model predictions (right), showing accurate classification of various recyclable materials.

To validate performance, labeled validation sets were created from previously unseen images and used to assess model accuracy during training (see Figure 6Figure). These validation batches included real-world challenges such as distorted cans, reflective surfaces, and overlapping items. The model demonstrated high detection accuracy, especially for metal and paper, and this visual feedback was instrumental in fine-tuning hyperparameters and class balancing during the training cycle.

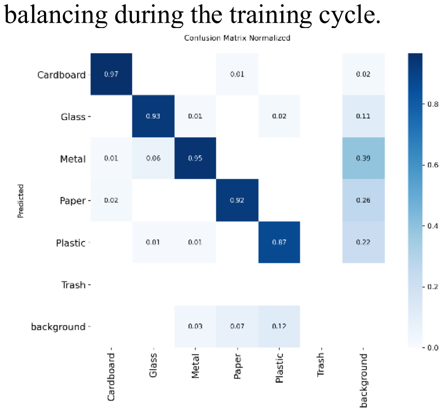

Figure 7: Normalized confusion matrix showing strong classification accuracy across all five material classes.

A normalized confusion matrix also generated post-training to evaluate class-wise performance (see Figure 7). The results indicate strong accuracy across all classes, with cardboard (97%), metal (95%), glass (93%), paper (92%), and plastic (87%). While plastic showed slightly lower accuracy due to overlapping visual features with background elements, overall system performance remained robust and well-balanced. This evaluation confirmed that the model was sufficiently trained for deployment in a real-time, embedded system.

To further increase reliability, a Canny edge detection fallback is triggered whenever the YOLOv11 model returns a classification with confidence below 80%. This helps ensure that occluded or ambiguous objects are still detectable based on shape rather than class features. Once an object is confidently detected and localized, its spatial coordinates are passed to an ESP32 motor controller, which calculates the necessary inverse kinematics and executes the movement sequence. This fully integrated, vision-guided control loop allows the system to perform autonomous sorting without external supervision or manual intervention.

V. PCB Design

The PCB design for the TRASH system consists of three custom boards: the main control PCB, the CCD/spectrometer PCB, and the LED ring PCB. Separating the system into multiple boards helped improve signal integrity, move subsystems freely, and simplify testing and integration. All PCBs were designed in KiCad and manufactured after drawing the schematics and testing on breadboard prototypes.

A. Main PCB

The main control PCB acts as the central hub for power regulation, motor control, and system communication. It includes the ESP32-S3 microcontroller, power input connectors, step-down regulators (LM2678 for 5V and AMS1117 for 3.3V), UART and I2C ports for motor drivers and sensors, and connectors for interfacing with the Raspberry Pi and spectrometer board.

The board layout was designed while keeping current handling capabilities in mind. High-current traces for motors were routed with wider widths and kept short, while the 3.3V logic and ADC sensitive sections were placed away from switching regulators and high-power outputs. All regulators are placed on daughter boards to provide modularity and lower replacement costs if a regulator fails. All external connections use JST connectors and terminal blocks to make setup easy.

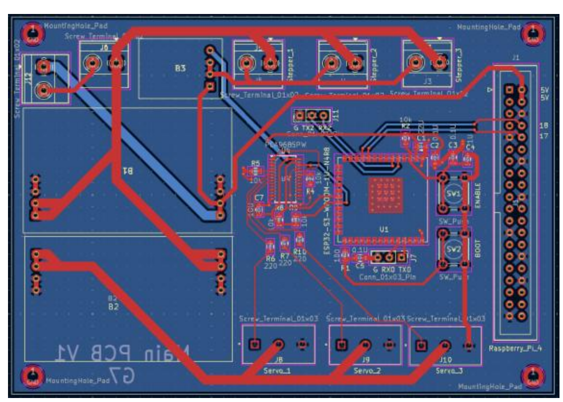

Figure 8: Main PCB Layout

B. CCD Spectrometer PCB

The CCD sensor board interfaces with the TCD1304 linear image sensor and includes all supporting components required for operation. An STM32F401 microcontroller is mounted on the back side of the board and generates precise timing signals using internal timers. It also reads the analog signal using its ADC and transmits the data to the Raspberry Pi over USB.

To ensure signal integrity, the board uses a dedicated 3.3V and 4V regulators and has carefully routed traces for analog inputs and CCD clocks. A 5V input connector powers the board, and an additional 3-pin jst connector provides GPIO and power output to the LED ring board.

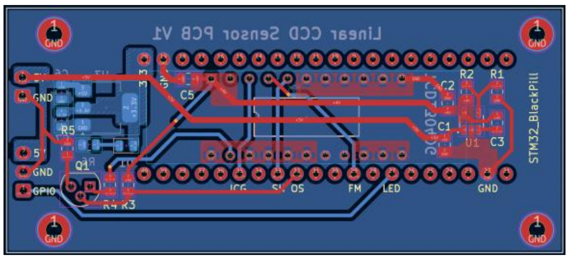

Figure 9:Linear CCD Sensor PCB Layout

C. LED Ring Light PCB

The LED ring board contains 8 pairs of specific wavelengths LEDs arranged in a circular pattern around a central cutout designed to align with the CCD sensor's optical path. This board provides uniform illumination of objects during reflectance analysis. The LEDs are controlled using a MOSFET driven by GPIO from the STM32 on the sensor board.

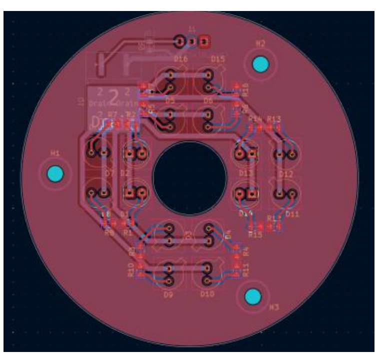

Figure 10:LED PCB Layout

VI. Testing and Results

A. Computer Vision Testing

To evaluate the object detection system, we trained a YOLOv8 model on over 3,000 labeled images covering five recyclable classes: plastic, glass, metal, cardboard, and paper. The model showed strong accuracy during testing, especially for metal and cardboard. Validation tests confirmed robust generalization, even in cluttered or low-light scenes. When deployed on the Raspberry Pi 5, the system ran in real-time with minimal delay, successfully detecting and localizing objects in the workspace. Overall, the vision system met performance expectations and was ready for integration with the robotic arm.

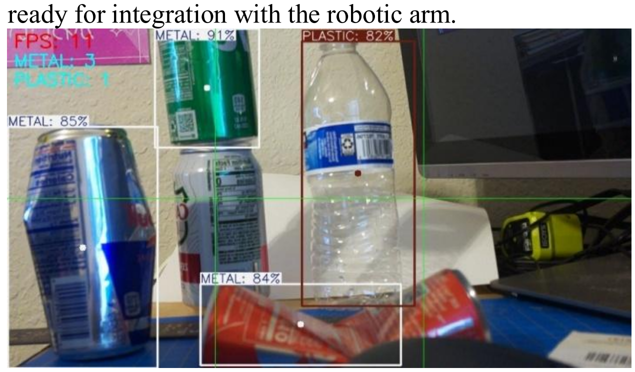

Figure 11: live object detection results using the deployed YOLOv11 model on Raspberry Pi, confirming real-time performance and high-confidence classification.

B. Motor Testing

Servo motor testing began with basic PWM signals from the ESP32, later upgraded to PCA9685 for smoother multi-servo control. We initially faced microcontroller resets due to power drops, which were fixed by adding a separate buck converter for the servo rail.

Stepper motors were tested using MKS Servo42C drivers over UART. The manufacturer's GUI was used to verify direction and speed, while custom ESP32 firmware handled angle and speed commands. Encoder feedback confirmed accurate motion, and PID-based holding worked well under load.

We also built a simple web interface for controlling joint angles and viewing motor status, which made testing all six joints more efficient.

C. Linear CCD Sensor Testing

The CCD spectrometer subsystem was first prototyped using a breadboard setup. The STM32 MCU generated the timing signals using its hardware timers. Signal amplification for the analog output was handled using a 2N3906 transistor circuit. Oscilloscope measurements confirmed synchronization between the analog output and ADC sampling, validating proper CCD operation.

The spectrometer firmware utilized DMA with the STM32 ADC to efficiently capture CCD output. Python-based plotting tools were created to visualize the spectral data and evaluate signal clarity. By adjusting sampling timing and gain, clean reflectance curves were captured under different lighting conditions. Figure 12 demonstrates consistent output across the sensor.

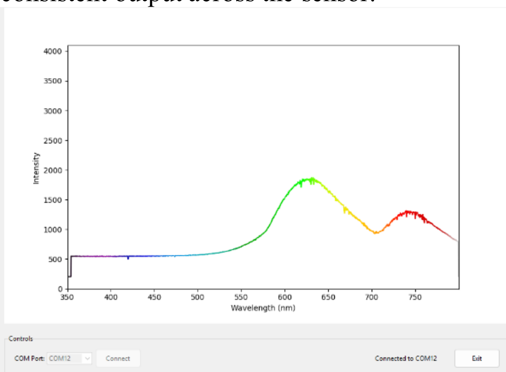

Figure 12: CCD Sensor Serial Plot

D. *Spectrometer testing*

To evaluate material classification, a sample library was collected using a StellarNet Blue-Wave Miniature Spectrometer. A basic test program was developed to classify these samples, achieving an 80% identification accuracy. This helped highlight the importance of how the reference library is built and how it directly impacts classification reliability. The next step is to integrate live data from the onboard sensor and compare it against this library for real-time identification.

The spectrometer hardware, including the optics, sensor, and LED array, was successfully integrated thanks to close collaboration between the mechanical and photonics teams.

VII. Challenges and Lessons Learned

One of the major mechanical challenges encountered during development was the initial attempt to implement a tendon-driven actuation system with the motors positioned near the base of the robot to reduce weight. The design used carbon fiber rods for the arm links and routed tendons through pulleys to transmit torque to remote joints. However, we faced critical structural issues mounting the rods to the axle. A more critical issue arose with the tendons themselves. When using metal cables, the cable was unable to conform around the pulley and would uncoil or resist tensioning. When using fishing line, we experienced excessive stretching when applying any meaningful loading. This resulted in poor positional accuracy and unusable arm behavior. Due to these limitations, we pivoted to a direct-drive actuation method with the motors mounted at each joint, significantly improving mechanical stability and precision.

Another major challenge was software compatibility with ROS2, which we initially planned to use for communication between system components. However, we ran into hardware issues, especially with the Raspberry Pi Camera, and found that the Coral Edge TPU required outdated Python versions incompatible with newer camera libraries. These conflicts made it difficult to use ROS2 effectively while keeping other dependencies up to date. MoveIt2 integration was also unstable with our custom robot model. In the end, we decided to set ROS2 aside as a future upgrade rather than a core part of the system. Early testing would have helped catch these limitations sooner.

VIII. Conclusion

In summary, the goal of this project was to apply our engineering skills toward the design, development, and testing of an automated robotic sorting system for recyclables. The T.R.A.S.H. system uses a Raspberry Pi and ESP32 to handle real-time object detection, motor control, and material classification. A combination of a custom-trained vision model and a reflection-based spectrometer allows the system to identify and sort recyclable items with good accuracy. Throughout the project, we designed and built custom PCBs, implemented firmware for multiple microcontrollers, and developed a working prototype that performs as intended.

Acknowledgement

We thank Dr. Wei for his guidance and Dr. Weeks for his support during testing and PCB assembly. We also appreciate the resources provided by UCF's College of Engineering and Computer Science.

Engineers

Sayed Almousawi: Electrical Engineer
Dylan Bowers: Photonics Engineer
Jayden Lumas: Computer Engineer
Tristan Coopman: Mechanical Engineer

References

[1] W. H. Cantrell, "Tuning analysis for the high-*Q* class-E power amplifier," *IEEE Trans. Microwave Theory & Tech.*, vol. 48, no. 12, pp. 2397-2402, December 2000.

[2] W. H. Cantrell, and W. A. Davis, "Amplitude modulator utilizing a high-Q class-E DC-DC converter," *2003 IEEE MTT-S Int. Microwave Symp. Dig.*, vol. 3, pp. 1721-1724, June 2003