Boston University
Electrical & Computer Engineering
EC464 Senior Design Project

Prototype Testing Plan

Ecobin

by

Team 9
Ecobin

Team Members

Aditya Wikara adwikara@bu.edu
Hayato Nakamura hayaton@bu.edu
Kevin Sinaga kfsinaga@bu.edu
Shreenidhi Jayaram shreej@bu.edu
Charles Thao cthao19@bu.edu

Equipment

The required equipment for this prototype testing could be divided hardware, software, and any mechanical components. The ECE team provides the software and hardware while the ME team build any mechanical parts for the prototype.

Hardware

- Raspberry Pi 3 B+ (with 32GB SanDisk SDHC Class 10 card)
- Raspberry Pi Camera Module v2
- Camera Lens
- PIR Motion Sensor
- Ultrasonic Sensor HC-SR04
- LCD Screen - 3.5 inch
- LED Strips (12V, 0.9W)
- LED diodes (green, red, white)
- 12W Power Adapter (Pi3)
- 12DC Power Supply (LED Strips)
- Transistor (BJT PN2222A)

Software

- Object Classification (Machine Learning)
- Web Scraper (Collect datasets from Google Images)
- MongoDB (database)
- Python Scripts for sensors and overall Ecobin functionality
- Dependencies
    - Pymongo, base64, numpy, PIL, keras/tensorflow

Mechanical

- Object Detection Platform
- Ecobin Housing
- Camera Mount/Stand
- Camera Lens Mount (3D printed)

Description of Equipment & Set Up

For this prototype, there are several additional equipment for both hardware and software. For hardware, there is a new PIR motion sensor, an ultrasonic sensor, LED strips, and camera lens. The new PIR motion sensor has more consistent results compared to the previous one. An ultrasonic sensor has been configured to measure distance, which will be used to measure the capacity of the trash bin. The LED strips will illuminate the enclosed Ecobin system. Lastly, the camera lens minimizes the distance required to take a picture, hence saving a lot of space.

In terms of software, we applied transfer learning and fine tuning to the Keras pre-trained VGG16 model. VGG16 is a general object classifier, trained on Imagenet and can output 1000 classes. We modified it into an object classifier with 12 classes: Metal Can, Bottle, Fork, Spoon, Cup, Glass, Orange, Apple, Banana , Leafy Greens, Paper and Plastic Containers. Each class consists of roughly 400 images which we scraped from Google images or collected from ImageNet. This new model will be trained and run on the Shared Computing Cluster at Boston University. The current configuration includes modifying the last 4 layers of VGG16 to a Flatten layer, a Dense layer with 4096 nodes, DropOut layer with rate 0.8 and an output layer of 12 nodes (for 12 classes).

In addition, the Ecobin system is now also interfaced with MongoDB. The RasPi stores data in the MongoDB database, the Keras model retrieves this data and runs the sorting algorithm. This allows the RasPi to securely communicate with the cloud, and later also to the mobile application. Additionally, MongoDB is not stored locally, allowing access to the database from any device, creating flexibility.

In terms of the mechanical components, the Mechanical Engineering team built a testing environment for the Ecobin. Previously, all of the testing was done under non ideal conditions, meaning the camera and object was not inside an enclosed space, which would have affected the quality of the picture. Therefore, the ME team build an object detection platform, an enclosing box, a mount for the camera lens, and a mount for the camera itself. With this, we are able to take a picture of the object while it is being illuminated by the LED strip.

The general setup will be similar to previous prototype testing. The RasPi functions as the microcontroller of the Ecobin. The motion sensor detects motion and triggers the camera to take a picture of the object inside the enclosed box. The RasPi then communicates with

MongoDB and sends the picture to the database, where it can be retrieved by the Keras model. The picture is processed and classified. Once it is classified as trash or recyclable, the data is saved back in MongoDB, and sent back to the RasPi. The RasPi then drives this signal to the red or green LED light depending whether its trash or recyclable respectively.

Pre-testing Setup Procedure:

Software

1. The RasPi has internet connection.
2. Make sure the model is trained with the desired classes and the HDF5 file is ready.
3. Make sure the computer/SCC has all the dependencies.
4. Mongolab (Mlab3) database set up to communicate from raspberry pi and SCC ML database
5. The RasPi is interfaced with the API and the SCC

Hardware

1. Plug in all power supply.
2. Test if all sensors are working properly.
3. Attach LED strip into the prototype.
4. Run the python script, *ecobin.py.*

Testing Procedure:

1. Place an object on the detection platform(cardboard box), and wave hand over PIR motion sensor.
2. LED strip turns on
3. Image of trash is captured by the Raspi camera inside the enclosed space
4. RasPi connects to the pymongo database, in which the image is processed and classified as trash or recyclable in the SCC.
5. The RasPi receives information from the API (pymongo database), whether the object is ‘trash’ or ‘recyclable’.
6. The RasPi outputs a signal towards LEDs, which represents the motor.
7. The LCD screen outputs the taken image, the object’s class, and the confidence level(probability)

8. This process will be repeated 10 times with equal number of trash and recyclable objects. This data will be recorded in the score sheet, where the accuracy could then be calculated.

Graphical Representation

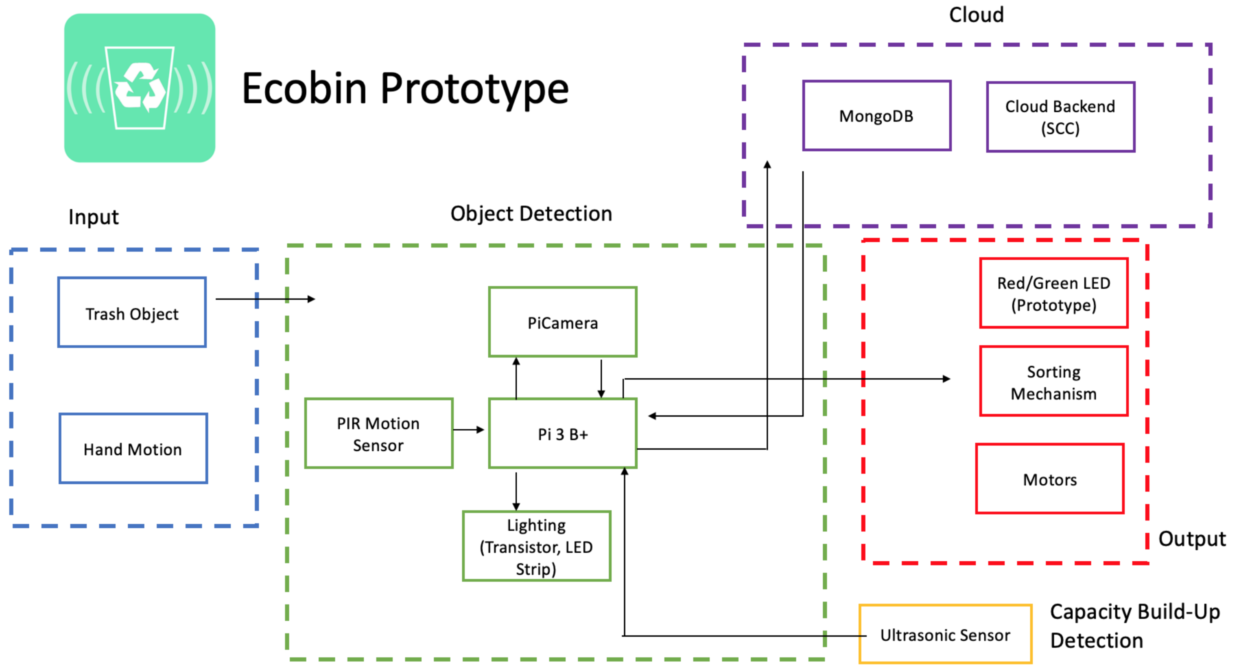

Figure 1: Block diagram of the Ecobin prototype. For this prototype, the RasPi is interfaced with MongoDB

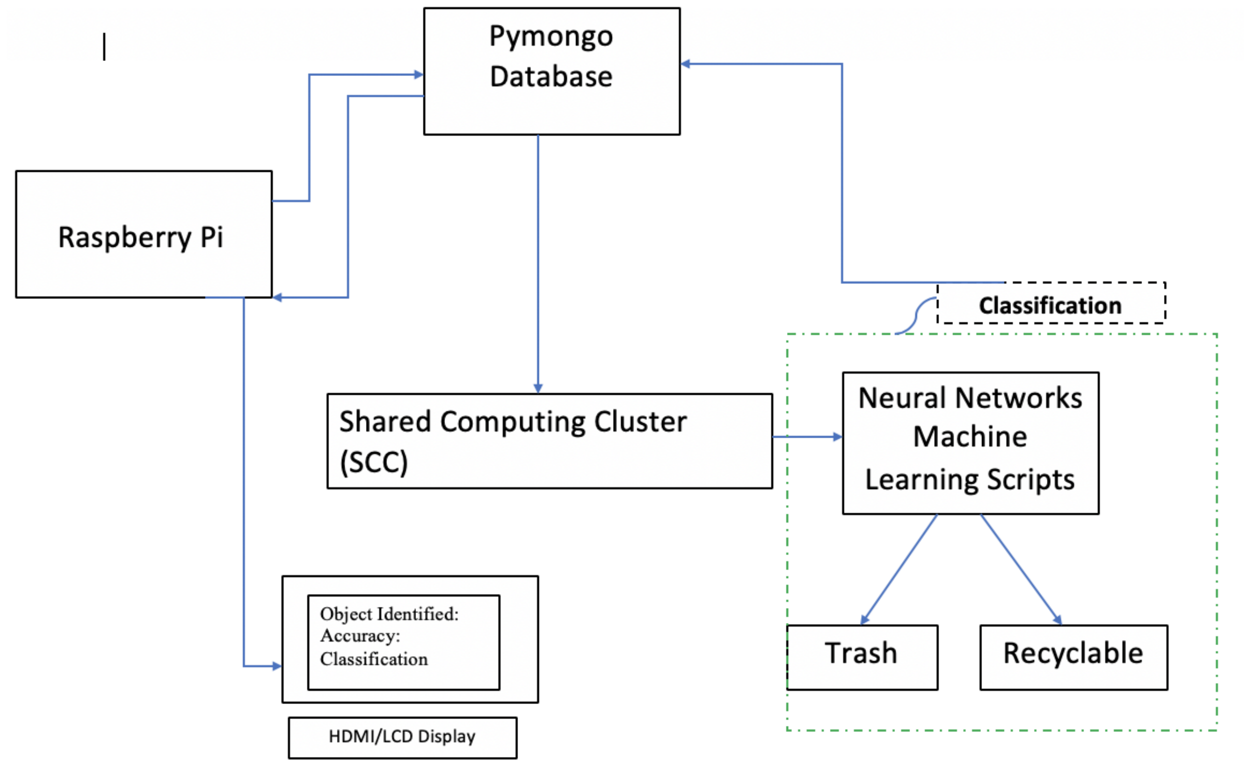

Figure 2: Software block process diagram between the raspberry Pi , PyMongo Database and Shared Computing Cluster

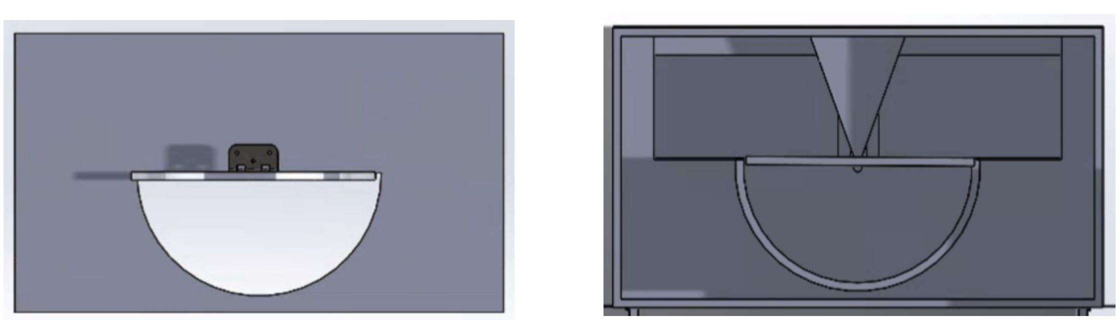

Figure 3: 2D representation of the Ecobin design (the classifier part). This is a top view. The actual design will have the same dimensions with the prototype but with different material.

Measurable Criteria

The criteria for successful running and output is as follows:

I. The Raspberry Pi should successfully capture an image and output whether it is trash or recyclable, as well as the confidence level (probability).

II. On the breadboard, there is a Red LED and a Green LED. The Red LED should light up if the system detects "Trash" and the Green LED will light up if it detects a "recyclable" object.

III. The object detection should be done in an enclosed space to make the testing environment ideal. This way, the picture will be taken in the dark where the object is illuminated by the LED strip. This is the type of picture that the Ecobin will receive for the final product.

IV. The RasPi must be able to communicate to mongoDB server and the API in order to process the picture in the cloud/SCC. This process is deemed successful when the handshaking is completed in < 2 seconds.

V. If there's motion above the PIR motion sensor, the Raspberry Pi should successfully take a photo of the object. The LED strip should also turn on when motion is detected, and turned off after the picture has been taken.

VI. Successfully interface the LCD screen with the RasPi, providing a UI for the results of the classification.

VII. The Raspberry Pi should successfully classify whether an object is recyclable with **75%** accuracy

VIII. Both encoding and decoding the 5Mp raspberry pi image in base64 binary format should be successful, with the resulting image being identical to the original image.

Score Sheet

Object	Category	Correct? (Y/N)
Metal Can	**Recyclable**	
Red Cup	**Recyclable**	
Green Cup	**Recyclable**	
Metal Spoon	**Recyclable**	
Metal Fork	**Recyclable**	
Transparent Cup	**Recyclable**	
Banana	**Trash**	
Orange	**Trash**	
Apple	**Trash**	
Leafy Green	**Trash**	
Result →		%

Hardware Pinout

RasPi Pin #	Usage/Description
2	5V Power -> PIR VCC, HC-SR04 VCC
6	Ground
7	GPIO 04 -> PIR Output
13	GPIO 27 -> HC-SR04 Trigger
16	GPIO 23 -> HC-SR04 Echo
11	GPIO 17 -> LED strips Output