**Boston University Electrical & Computer Engineering**

**Boston University Electrical & Computer Engineering**

**EC463 Senior Design Project**

**First Prototype Testing Plan**

Ecobin

***W*h**

1*972:*

**1*. 7*2**

**by**

**Team 9** Ecobin

Team Members

Aditya Wikara adwikara@bu.edu **Hayato Nakamur**a hayaton@bu.edu

Kevin Sinaga kfsinaga@bu.edu **Shreenidhi Jayara**m shreej*@*bu.edu

Charles Thao cthao19@bu.edu

**Required Materials**

Hardware:

**• Raspberry** Pi 3 B+ (with 32GB SanDisk SDHC Class 10 card)

**• Raspberry Pi Camera Module v2**

**• PIR Motion Sensor**

• LED Strips (12V, 0.9W)

**• LED diodes (green, red, white)**

• 12W Power Adapter (Pi3)

**• 12DC Power Supply (LED Strips)**

• Transistor (BJT PN2*22*2*A)*

**Software:**

• Python 3 scripts:

o **Capture photos in JPEG format o Object recognition**

**Tensorflow**

**Keras**

Numpy, Scipy

o

Signal handling

. Raspi GPIO

**Set Up**

**The equipments and setup are divided into two parts: the Raspberry Pi with the camera** for hardware; image capturing and image recognition Python scripts for software. A computer acts as processing cloud server. Peripherals such as the motion sensor and the Pi ca**mera are** attached to the Raspberry Pi. If the Passive Infrared Motion Sensor (PIR) detects an object, it **transmits** a 3.3V signal to the Raspberry Pi via the Pi's GPIO pins. This trigger signals the Pi to **turn on th**e LED strip to illuminate the bin, and captur**e an image using the camera. Afte**r that, the Pi will forw**ard the image to a simulat**ed server. Within the simulated server, a Python script, powered by the Keras API and the pre-trained VGG model, identifies object in the image. (Keras **is an open source high level neural netw**ork library which al**lows for efficient and fast** prototyping and experimentation.) The server then communicates with the Pi and then a signal is sent to the LED which in turn lights up green(if recyclable) and red(if trash).

**Pre-testing Setup Procedure:**

**Server side:**

1. Enable ssh connectivity with the Raspberry Pi with the command *ssh pi@\*ip\_address\** 2. Enable sftp connectivity with the Raspberry Pi with the command *sftp pi@\*ip\_address\**

Make sure that the s**ftp environment is in the correct working directory.**

(ie*. cd Desktop)* 3. Run the python script, *keras\_test\_individual.py* Raspberry Pi side:

1. Ensure that the background is clear 2. Run the python script, *ecobin.py*

**Cardboard Bax lined with white paper for good amera background**

**12 V Power Supply**

**DO!**

1.#B

11.

1 +

**FITIL**

**.** UIT

1

**.**

+ I.

it. IIIIPITI

I

.

**+ITUTE** - IIIIITUITILLE! **INITIOII.**

**\***

.

**..**

bitti II.WIL

**.** ..

**PI Camera**

**Waste (Plastic Bottle)**

.

**0271**

..

**.**

III.

R

**PAN**

-

O

**.**

**MEN**

**IPPEN**

U

**.**

**5 V P*ow*er Supply**

**PIR M*o*tion Sensor**

**Breadboard with Red and Green LEO**

**Raspberry PI 3 B+**

***Figure 1: Illustration of Setup and Process Flow***

**Testing Procedure:**

1. Place an object on t**he detection platform(cardboard box), and wave hand over PIR**

**motion sensor. 2. Image of trash is captured by the Raspi camera 3. The image is processed in t**he simulated server, in this c**ase, a computer.** 4. [Computer] - Through sftp, enter the follow**ing commands**

*i get trash.jpeg* Sends objec**t image to the computer from the pi**

***ii. put code.txt* Sends detection results to the pi from the computer 5. The computer sends the results back to th**e Pi, the Pi sh**ows the result through the**

**red(trash) and green(re**cyclable) LEDs.

**ICO**

**Measurable Criteria**

The criteria for successful running and output is as follows:

I. The Raspberry Pi should successfully capture an image and output name of the captured

**image on to the terminal** 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. The object is able to be identified and sorted into these two overarching **classification because we categoris**ed common objects into two separate lists and once the

object is detected and identified, it searches the list to match the object to a classification. III. If there's motion above the PIR motion sensor, the Raspberry Pi should successfully take

a photo of the object. IV. The Raspberry Pi should successfully classify whether an object is recyclable with 75%

OVE

**uce**

**accuracy.**

**Score Sheet**

**Correct? (Y*/*N)**

*7*

*(2)*

**Object**

**Category** I Plastic bottle (1) Recyclable CRO

Plastic bottle (2) Recyclable (R)

*Apple o*

Trash (T | Apple (2) - Trash (T)

Deformed paper (1)| Trash (7) Deformed paper (2) Trash CD orange (1) Trash (7) *ora*nge *(2*)

Trash C) I plastic bottle (3) | Reaclable (R) I plastic bottle 14 Recyclable (R)

**Result-**

***%***

**Hardware Pinout**

**Pi3 Pin *#***

**Usage*/*Description**

**5V Pow**er -> PIR Vcc

**Ground -> PIR GND**

**GPIO 1*7* -> PIR Out**

**GPIO2 -> White LED diode**

**GPIO3 -> Red LED diode**

**GPIO 4 -> Green LED diode**

**GPIO 27 -> LED strips**