

Intelligent Trash Can

Anurag Andhare
aandhar2@binghamton.edu
Binghamton University

Priti Wakodikar
pwakodi1@binghamton.edu
Binghamton University

ABSTRACT

Recycling is a difficult task. Different cities, counties, states have a different set of rules to define recyclable and non-recyclable items. Hence a system developed to collect recyclable items in one city can not be used in another, and this leads to the contamination of the recyclable waste. Contamination occurs in recycling when non-recyclable items are placed in recycling [1]. Contamination of recyclable waste starts at home. If the waste is properly separated at home, then we can eliminate the problem of contamination, and hence recycling can be improved. As recycling rules differ from city to city, we need a system that can be quickly adapted to the rules of a particular city. To address this issue we created an Intelligent Trash Can. This Intelligent Trash Can helps us to sort the trash at home to reduce contamination. The Intelligent Trash Can makes use of Machine Learning to categorize waste materials.

1 INTRODUCTION

Recycling is done to create products by reusing waste material. The quality of the products depends on the purity of the waste material. If the waste material is contaminated then that leads to producing inferior quality products. Contamination can include anything from food waste to plastic bags. [1] Different recycling systems allow for different materials to be recycled and consider different materials as contaminants. Generally, this contamination starts at home. Due to the long and incomplete list of recyclable and non-recyclable items, we mix recyclable items with non-recyclable items and dump them into the same trash can. When this trash goes to the recycling plant they can not recycle it properly because the trash now has non-recyclable items in it. For example, if a paper mill uses contaminated waste paper as raw material then this leads to the creation of a very poor quality new paper. Therefore, avoiding this contamination becomes the first step in the recycling process.

2 MOTIVATION

To prevent contamination it is necessary to be informed about the materials recycling programs accept. Respective city officials provide the list of recyclable and non-recyclable items to its citizens. But these lists are long and complicated, that people avoid reading them and hence do not follow the guidelines set for proper recycling.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference'17, July 2017, Washington, DC, USA

© 2020 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...\$15.00

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

Therefore, we decided to build a system that will help the user to sort the trash as recyclable and non-recyclable at home.

Machine learning is a great concept that can be used to solve many challenging life problems. Especially, its application in image processing has produced remarkable results. As our project involves image processing, we decided to make use of a machine learning for our project to train a model to classify the object as either recyclable or non-recyclable.

3 SYSTEM DESIGN

To classify the recyclable and non-recyclable waste items, we trained a machine learning model. We used a total of 10000 images to train our model. Figure 1 represents design of our system. It gives an idea that how the components are connected and how they work together. Once training is done the model is deployed on the Raspberry Pi board. [3] As the ML model is compiled using TensorFlow Lite [15], it can not be executed by the Raspberry Pi board alone. Therefore, to run this model on the Raspberry Pi [3] board we connected the Coral USB accelerator [7] to it. This enables quick inference using the ML model. The Coral USB accelerator [7] provides high-speed machine learning inferencing while consuming very low power. The Pi camera [4] is used to take the video stream of the objects. The object from the video stream is then classified by the ML model as either recyclable or non-recyclable. Based on the inference, the appropriate servo motor [10] connected to a trash can lid is rotated. This opens the trash can lid. The lid is kept open for 3 seconds and then the motor is rotated again in reverse angle to close the lid. We also used Ultrasonic sensors [13] to detect the level of garbage in the trash can. To indicate that the trash can is full, we used a LED mounted on the lid of the trash can. When the trash can is full, the corresponding LED on the lid will glow. The trash can lid will not open to taking any garbage in while the LED is glowing. This prevents the overfilling of the trash can. Once the trash can is empty, the LED will be turned off, and the lid will operate normally.

Figure 2 highlights how the control flows through the system.

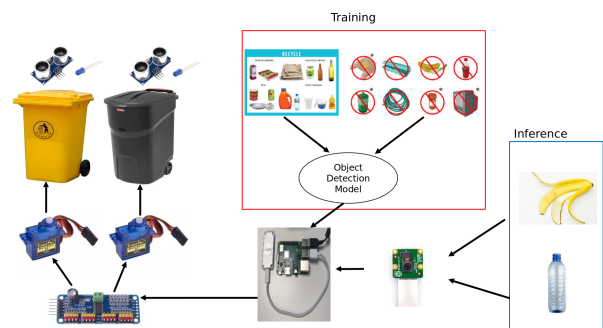


Figure 1: System Design

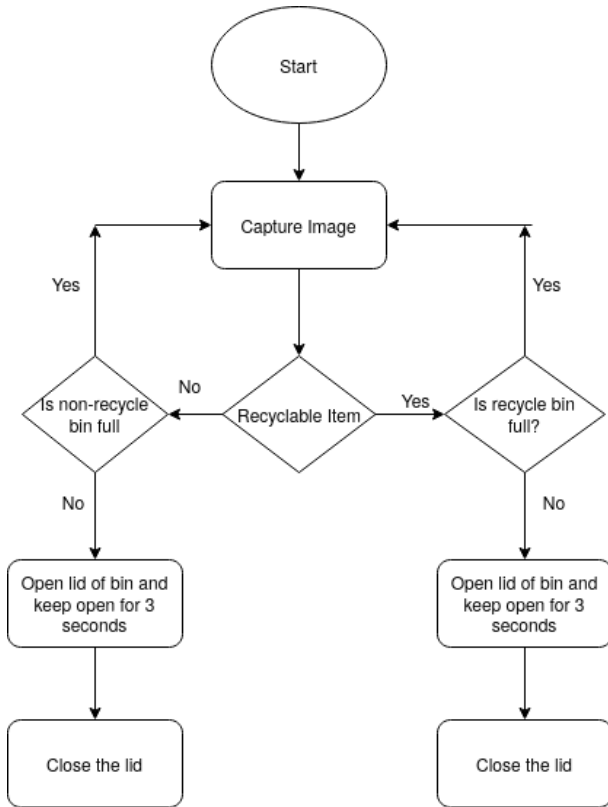


Figure 2: Control Flow

4 IMPLEMENTATION

Implementation of our project started with collecting images for the training of the machine learning model. We collected images from online datasets and also clicked pictures that were not available in the dataset. We used around 6500 images for training. We then categorized those images as recyclable and non-recyclable. We also used a third category named "others". We trained our model on a tool called Teachable Machines. [14] We used this tool because it enabled us to directly export the trained model as a TensorFlow Lite [15] model.



Figure 3: Project setup

Controlling the servo motor [10] from Raspberry Pi [3] was our next step. We tried several different libraries and found that

ServoKit [6] is the best suitable one. This library also allowed us to provide a delay in the operation to hold the rotation of servos.

Then we wanted to measure the level of trash in a trash can. For that, we used Ultrasonic sensors [13]. To operate these sensors we used the hcsr04 library. [9] We mounted these sensors on the lid of trash cans. If the trash can is full, the LED mounted on the lid will start glowing. The trash can lid will not open while LED is ON.

Figure 3 shows the complete setup of the project. In figure 4 you can see the Ultrasonic sensor, LED, and Servo motor mounted on the trash can lid.

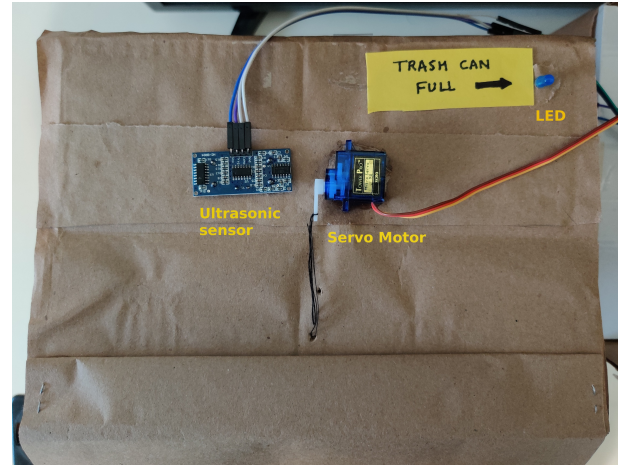


Figure 4: Top view of a trash can lid

5 CHALLENGES

During the development of this project, we faced many challenges. Most prominent was the challenge with the machine learning model. The first machine learning model that we designed was able to categorize the objects into recyclable and non-recyclable categories. But as we were using a video stream, that led to continuous inferencing, and the lid of the dustbin was opening without even giving any input. Therefore, we decided to use three categories for inferencing. Recyclable, non-recyclable, and others. The third category helped us to make the inferencing more accurate and the problem with the lid was solved.

As the servo motor [10] is an electromechanical device, it can not match its operating speed with the signals received from Raspberry Pi. Therefore we had to calculate the delay to be used while operating the servo motor. We stopped the signals coming from Raspberry Pi to let the servo motor complete its rotation.

The quality of the images used during training impacts the inference quality. As the lighting condition in which the images were taken plays an important role, we clicked images of objects in low light, bright light, white light, and yellow light conditions. This strategy made our model more accurate.

6 HARDWARE

Selecting suitable hardware for this project was critical for getting accurate results. We wanted to make our project portable and small enough to store and move around comfortably. Therefore we decided to use a single board computer. Next, we wanted a board that

is capable of running a machine learning model and doing quick inferencing. Because if inferencing time is more then, user quality of experience will be poor. To make inferencing faster, we used a Coral Edge TPU coprocessor [7]. It enhances the inferencing speed while keeping the power consumption low.

Following is the list of hardware we used for this project:

- (1) Raspberry Pi 3 [3]
- (2) Coral USB accelerator [7]
- (3) Raspberry Pi camera module V2 [4]
- (4) SG90 9G Micro Servo Motors [10]
- (5) PCA9685 Module Servo Motor Driver [12]
- (6) HC-SR04 Ultrasonic sensors [13]
- (7) LEDs
- (8) Resistors

(1) Raspberry Pi 3

Given that the Raspberry Pi 3 [3] is a powerful single-board computer with GPIO pins, it was our first choice for the project. GPIO pins allowed us to connect all the sensors and other devices used in the project to the Raspberry Pi. With support from the USB accelerator, it runs our program as expected.

(2) Coral USB accelerator

The Coral USB Accelerator [7] adds an Edge TPU coprocessor to the system, enabling high-speed machine learning inferencing. We just connected it to the Raspberry Pi's USB port and the board was ready to do the high-speed inferencing.

(3) Raspberry Pi camera module V2

The Raspberry Pi camera module V2 [4] has a Sony IMX219 8-megapixel sensor. This camera is capable of taking high definition pictures and videos. We used it to take a video stream and then object detection was done on that stream.

(4) SG90 9G Micro Servo Motors

We needed a device that will enable us to open and close the lid of the trash can. The Servo motor [10] can be rotated through specific angles as needed. And hence we decided to connect to the lid of the trash can to open and close them.

(5) PCA9685 Module Servo Motor Driver

Servo motors need 6V to operate. If we connect the servo motors directly to the Raspberry Pi then there is a possibility of brownout. Hence we used a servo motor driver [12] to connect the servo motors.

(6) HC-SR04 Ultrasonic sensors

To detect the level of garbage in the trash can we used an ultrasonic sensor [13]. The sensor head emits an ultrasonic wave and receives the wave reflected from the target. Ultrasonic Sensors measure the distance to the target by measuring the time between the emission and reception.

(7) LEDs

We connected LEDs to the lid of each trash can. The LED indicates the trash level in the can. If the trash can is full, the LED turns ON. It is OFF otherwise.

(8) Resistors

These are required to connect ultrasonic sensors and LEDs to the Raspberry Pi.

7 SOFTWARE

For our project, the software is equally important as hardware. As we are using Edge TPU [7], we used a compatible machine learning framework, TensorFlow Lite [15]. Also, to operate servo motors [10], ultrasonic sensors [13], and LEDs we used respective libraries available from Adfruit [6][9][8]. All the programs were written using Python programming language.

Following is the list of software we used for this project:

- (1) TensorFlow Lite [15]
- (2) Opencv [11]
- (3) ServoKit library [6]
- (4) Adafruit hcsr04 library [9]
- (5) Board library [5]
- (6) Digitalio library [8]
- (7) Python [2]

(1) TensorFlow Lite

TensorFlow Lite [15] is a set of tools to help developers run TensorFlow models on mobile, embedded, and IoT devices. It enables on-device machine learning inference with low latency and small binary size. The Coral devices can run this framework and integrates well with it. It provides all support for training and inferencing using a machine learning model at edge devices.

(2) Opencv

OpenCV [11] is a library of programming functions mainly aimed at real-time computer vision. We are using it to capture the video stream from the Pi camera [4].

(3) ServoKit

This library provides all the functions necessary to operate a servo motor [10] from Raspberry Pi [3].

(4) Adafruit hcsr04 library

We used this library to operate Ultrasonic sensors [13]. We can directly get the distance of an object from the sensor using this library.

(5) Board library

We operated GPIO pins of Raspberry Pi [3] using this library

(6) Digitalio library

We used this library to operate LEDs

(7) Python

Complete program was written in Python Programming language [2].

8 CONCLUSION

Sorting trash is a tedious process. A long and incomplete list of recyclable and non-recyclable items makes this task more difficult. Due to this difficulty people tend to throw all items in the same trash can. This leads to contamination and hence recycling becomes costly and sometimes impossible. Our project solves this problem with the help of Machine Learning. Our trained machine learning model can sort the trash into recyclable and non-recyclable items and hence prevents contamination. The garbage level detection also prevents overfilling of the trash can. This system can be used in any city by just training a new machine learning model according to the list of the recyclable and non-recyclable items of that city. Our project, therefore, saves time of people at home, time and money of recycling centers, and most importantly it helps to save our precious environment.

REFERENCES

- [1] 2013. Recycling bin contamination. <http://sustainability.umich.edu/environ211/recycling-bin-contamination>. (2013).
- [2] 2016. Python 3.6. <https://python.org>. (2016).
- [3] 2016. Raspberry Pi 3. <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/?resellerType=home>. (2016).
- [4] 2016. The Raspberry Pi Camera Module v2. <https://www.raspberrypi.org/products/camera-module-v2/?resellerType=home>. (2016).
- [5] 2018. Board, Circuitpython. <https://circuitpython.readthedocs.io/en/6.0.x/shared-bindings/board/index.html?highlight=board>. (2018).
- [6] 2018. Servokit Library. <https://circuitpython.readthedocs.io/projects/servokit/en/latest/>. (2018).
- [7] 2019. Coral USB Accelerator. <https://coral.ai/products/accelerator/>. (2019).
- [8] 2019. DigitalIO. <https://circuitpython.readthedocs.io/en/6.0.x/shared-bindings/digitalio/#digitalio.DigitalInOut>. (2019).
- [9] 2019. HC-SR04 Library. <https://circuitpython.readthedocs.io/projects/hcsr04/en/latest/api.html>. (2019).
- [10] 2019. Micro Servo Digital SG90. <http://www.towerpro.com.tw/product/sg90-7/>. (2019).
- [11] 2019. OpenCV. <https://opencv.org/>. (2019).
- [12] 2019. PCA9685 16-Channel 12-bit PWM/Servo Driver. <https://www.adafruit.com/product/815>. (2019).
- [13] 2019. Ultrasonic Sensor HC-SR04. <https://www.adafruit.com/product/3942>. (2019).
- [14] 2020. Teachable Machines. <https://teachablemachine.withgoogle.com/>. (2020).
- [15] 2020. TensorFlow Lite. <https://www.tensorflow.org/lite>. (2020).