A ROBOTIC SYSTEM FOR AUTOMATIC IDENTIFICATION AND COLLECTION OF RECYCLABLE PLASTIC BOTTLES

by

Uzma Ahmed Din

A Thesis Presented to the Faculty of the
American University of Sharjah
College of Engineering
in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science in
Electrical Engineering

Sharjah, United Arab Emirates

May 2020

Declaration of Authorship

I declare that this thesis is my own work and, to the best of my knowledge and belief, it does not contain material published or written by a third party, except where permission has been obtained and/or appropriately cited through full and accurate referencing.

Signature  …….

Date……13-05-2020..……

Approval Signatures

We, the undersigned, approve the Master's Thesis of Uzma Ahmed Din

Thesis Title: A Robotic System for Automatic Identification and Collection of Recyclable Plastic Bottles

Date of Defense: 07-May-2020

Name, Title and Affiliation	Signature
Dr. Shayok Mukhopadhyay Assistant Professor, Department of Electrical Engineering Thesis Advisor	
Dr. Usman Tariq Assistant Professor, Department of Electrical Engineering Thesis Co-Advisor	
Dr. Hasan Mir Professor, Department of Electrical Engineering Thesis Committee Member	
Dr. Lotfi Romdhane Professor, Department of Mechanical Engineering Thesis Committee Member	
Dr. Nasser Qaddoumi Head Department of Electrical Engineering	
Dr. Lotfi Romdhane Associate Dean for Graduate Studies and Research College of Engineering	
Dr. Sirin Tekinay Dean College of Engineering	
Dr. Mohamed El-Tarhuni Vice Provost for Graduate Studies Office of Graduate Studies	

Acknowledgements

Firstly, I would also like to thank my beloved family, without whose help and support I would not have been able to reach this far.

I would like to extend my gratitude to my advisor Dr. Shayok Mukhopadhyay and co-advisor Dr. Usman Tariq, for their guidance, patience, and help. My thanks also go to the Electrical Engineering program for giving me the opportunity to do my Masters's degree here at American University of Sharjah. Special thanks to all faculty members who taught me during my time here. I would also thank all my friends with whom I spent unforgettable times: Koshish Koirala, Danial Waleed, Ali Tameemi, and all the others.

Finally, I would like to thank my committee members, Dr. Lotfi Romdhane and Dr. Hasan Mir, for the effort and time they dedicated to reading my proposal and thesis reports, and for their valuable feedback. Also, I would like to acknowledge AUS for providing me with the assistantship during study.

Dedication

To my friends and family . . .

Abstract

Plastics have become a cornerstone of modern life, but they are also hazardous for the environment. Manually collecting and sorting such recyclable plastic waste from a mix of other refuse is tedious work, and has accompanying health hazards. This project develops a prototype autonomous robot that can identify, and collect plastic bottles from other waste, and return to recycling stations. The focus of this prototype is on the recognition and separation of a particular type of plastic bottle commonly used for packaging drinking water. In this work, the robot navigates to a general location where such plastic waste is expected to be found via GPS based navigation. Then, the robot recognizes recyclable plastic bottles, and computes the coordinates of the plastic bottle with respect to the robot, using a stereo camera. Finally, using a robotic arm attached to it, the robot collects the plastic bottle. This thesis does not focus on developing an appropriate gripper to pick up a bottle as it is out of the scope of the current work. To aid the picking process, the end-effector and bottles are fitted with small magnetic attachments. Moreover, the robot developed is 80% and 67% successful, in indoor and outdoor testing respectively. The future versions of the proposed robot may be advanced to segregate the collected plastic bottle waste into appropriate bins located at a recycling station.

Keywords: ***Plastic bottle recycling, autonomous robot, neural networks, computer vision.***

Table of Contents

List of Figures

List of Tables

List of Abbreviations

CCD Charged Coupled Device

DH Denavit-Hartenberg

FLDA Fisher's Linear Discriminant Analysis

HDPE High-Density Polyethylene

IMU Inertial Measurement Unit

KPCA Kernel Principal Component Analysis

LDA Linear Discriminant Analysis

PCA Principal Component Analysis

PET Polyethylene Terephthalate

PP Polypropylene

R-CNN Region-CNN

RoI region of interest

ROS Robot Operating System

SSD Single Shot Multibox Detector

SVD Singular Value Decomposition

SVM Support Vector Machine

YOLOv2 You only look once version 2

Chapter 1: Introduction

Plastic has become a vital part of our lives. Plastics are light and relatively strong, which makes them useful for many purposes. They used not only for food packages but also for packing non-food items. Plastic is also widely used in the transportation industry [1]. The sports industry is also heavily dependent on plastic. Most sports equipment and clothing are crafted from plastics because plastics make the product lighter, waterproof, and long-lasting. Plastic also plays an important role in the medical sector; it is very essential for making common tools like syringes. However, plastic waste is very dangerous for the environment. In addition, it is also dangerous for human, and animal health.

It can take 450 years or more for a plastic bottle to decay [2]. Plastic bottles have become a major public waste nuisance. Water bottles are made of Polyethylene Terephthalate (PET) plastic, which after a long period of time, breaks down into smaller pieces of plastic. These plastic pieces get absorbed into the sea via running water sources, or get embedded into the land and stay embedded for years. Land and sea animals end up consuming such plastic pieces which makes them sick and weak. If humans consume them inadvertently, the toxic effects can be very harmful to our health [3]. This increases the importance of reprocessing and reusing plastics. Segregating the different plastics autonomously is a challenging problem. This is because of the variety of plastics available, and also due to challenges presented by autonomous identification. So, this work focuses on specifically identifying and segregating plastic bottles from a mix of other items. The particular type of plastic bottles used in this work is the one commonly used for packaging drinking water. Manually sorting such recyclable plastic bottles from the trash can be labor-intensive, and poses health concerns. Thus, this work proposes the development of a robot that autonomously navigates to known locations and scans the vicinity for recyclable plastic bottles. It then uses neural networks to identify plastic bottles and collect them using a robotic arm. In future updated versions the robot may be advanced to segregate the waste collected into bins. This work has the potential to promote and achieve in the future, a cleaner environment by creating

automated trash segregating, and recycling robots. In the following section, a review of some literature related to the topics of this work is presented.

1.1. Literature Review

There have been many works along the line of detecting plastics using computer vision. For example, the authors in [4] detect plastic bottles using unmanned aerial vehicles (UAVs). They take pictures of plastic bottles in eight different backgrounds and create a large data set of images, called UAV-BD. This dataset is used to train a convolutional neural network (CNN). This CNN detects objects which are oriented either vertically or horizontally, with respect to the x-axis of the camera image. Furthermore, [5] compared Faster Region-CNN (R-CNN), Single Shot Multibox Detector (SSD) and You only look once (YOLOv2) methods for object detection. The average precision values of Faster R-CNN, SSD, and YOLOv2 90.3%, 90.1%, and 77.4% respectively on the UAV-BD dataset. For detecting objects which are not oriented perfectly vertically or horizontally with respect to the x-axis of the camera image, [4] used Rotation Region Proposal Networks (RRPNs).

In [5], a two-stage plastic bottle sorting system was presented. Firstly, feature extraction was done using Near Infrared (NIR) Imaging and then, for the classification process, linear, quadratic and diag-quadratic classifiers were used. This process of classifying plastic bottles was 94.1% accurate for clear and opaque plastic bottles. In the second stage, they used a Charged Coupled Device (CCD) camera to classify plastic bottles based on their colors. This process was 92% accurate for clear bottles and 96% accurate for opaque bottles. Therefore, the entire system had 83.5% accurate rate of detection.

In another study [6], the researchers used a Support Vector Machine (SVM) to differentiate between PET and non-PET plastic bottles. They used color moment and structural edge-based features. The identification process using SVM resulted in more than 90% accuracy.

In article [7], the identification of the objects is started after the feature extraction is completed. The plastic bottle was extracted from the background in the images

using Otsus threshold method before the feature extraction process. For the feature extraction, Principal Component Analysis (PCA), Kernel Principal Component Analysis (KPCA), Fisher's Linear Discriminant Analysis (FLDA), Singular Value Decomposition (SVD), and Laplacian Eigenmaps (LEMAP) were used. In the experiment, the authors in [7] used three types of plastic bottles which were made of PET, High-Density Polyethylene (HDPE) and Polypropylene (PP). In the first process, the plastic bottles were categorized as PET and non-PET with the help of SVM classification which was 96% accurate. In the second process, the non-PET bottles were classified as HPDE or PP with 92% classification accuracy. It was observed in this experiment that Linear Discriminant Analysis (LDA) based features gave better accuracy than PCA, KPCA, FLDA, SVD, and LEMAP based feature extraction.

In another experiment [8], samples of flowery, dendritic, and feathery patterns with 200 by 200 pixel size were used as a dataset. These three patterns were named as class A, B and C respectively. Twenty five training datasets from each class were taken. These samples were preprocessed with the help of PCA for dimensionality reduction. Then using this dataset the artificial neural network was trained and tested. The results show that the system could identify with 75%, 87.50% and 77.78% accuracy, samples in classes A, B and C respectively. This system classified 25% samples of class A as samples of class B and distinguished 12.5% samples of class B as of class A. Thus, the system had an overall accuracy of 80.09%. The misclassification was primarily due to the similarity between class A and class B.

The work in [9] uses four different types of neural networks to detect plastic debris underwater. Open and available databases [10] of all the debris found underwater were used for training their neural network architectures. The YOLOv2, Tiny-YOLO, Faster R-CNN, and SSD networks were trained for plastic debris detection, and the results were compared based on accuracy and processing time. Parameters like mean Average Precision (mAP) and Intersection over Union (IoU) were used to evaluate the network and models [9]. The average precision percentage in detecting plastic using YOLOv2, Tiny-YOLO, Faster R-CNN, and SSD were 82.3, 70.3, 83.3 and 69.8 respectively. Thus, faster R-CNN was more accurate compared to the other networks in [9]. While the above paragraphs surveyed results of neural network based classification, the

following paragraphs are geared mostly towards reviewing autonomous robotic navigation. A robot prototype presented in [11] collects floating waste from oceans, ponds or canals. This design is created using two propellers that are controlled by a Bluetooth based application and a conveyor belt that collects the waste in the trash box. But, this was not tested in a real environment and the motion of the robot was operated from a distance.

In [12] authors used forward-looking sonar (FLS) on an autonomous underwater vehicle with a trained deep convolutional neural network to detect debris in the ocean. The authors in [12] reported 80 percent accuracy in classifying marine debris. Marine data, which were the images captured inside a water tank containing different types of debris, like cans, valves, bottles, and cartons were used for testing. There were six classes, where class 0, class 1, class 2, class 3, class 4, class 5, and class 6 had images of metal objects, glass objects, cardboard, rubber, plastic materials, crops in the background respectively. This resulted in 80.8% accurate detection with a binary sensor and 70.8% accuracy with a multi-class sensor. But the system did not use real environmental data and as mentioned above, was only evaluated in a tank.

In another experiment [13], initial results on an autonomous indoor garbage detection and collection system using a robot was presented. It used two ultrasonic sensors for identifying the object location and the object of interest. In another research [14], a design of an indoor automated robot was presented. The robot collected waste from dustbins which are full. With the help of sensors that were fixed in the dustbins, the robot was notified when a dustbin is full. The robot then moved to the full dustbin and collected the waste from it.

In [15], the authors designed a road garbage cleaning vehicle that consisted of a power module, robotic arm, camera, and GPS module mounted on a vehicle. They tested identifying the waste using a camera and picking the waste by a robotic arm. However, [15] did not present any tests with the complete integrated system.

The following paragraph explores manual plastic segregation techniques, and presents the motivation for using neural network based techniques onboard a robot, for the work in this thesis.In addition to automated approaches, there are several manual approaches for separating plastics. These include color-based classification, electrostatic

separation, gravity segregation, froth flotation, and near-infrared ray techniques [16,17]. The gravity-based and froth flotation based separation techniques are costlier than other methods [17]. Likewise, the separation techniques involving lasers, X-rays, and spectrometers, have expensive experimental setups but the accuracy of segregation of plastics is better. Thus, computer vision and machine learning-based approaches appear to outshine other approaches for the segregation of plastic bottles, in terms of cost, accuracy, and ease of setting up and implementing a segregation technique. Hence, this work proposes the use of neural networks based approaches to identify plastic bottles and develop a complete system prototype.

1.2. Motivation

The motivation behind this thesis is to use robotics to help the environment be waste-free. Also, robotics can be used in areas where it is risky for humans to reach and clean the environment. The major components of the developed autonomous robotic system for collecting recyclable plastic bottles are a Kobuki robot base, stereo camera, and a robotic arm. The remainder of this thesis is organized as follows. Chapter two provides background information related to the developed robotic system. Chapter three describes the details of the developed robotic system for plastic bottle collection. Chapter four presents the experimental results. Chapter five concludes the thesis and discusses future work.

Chapter 2: Background

This chapter presents some preliminary information required for this work. This information is related to the robot operating system (ROS), computer vision, object detection with help of neural networks and forward, inverse kinematics of a robotic arm. The developed robot can detect and pick the plastic bottles from the waste autonomously. To achieve this, the robot works as follows. Firstly, to move the robot to the garbage collection point, path planning with known poses is used. Secondly, computer vision algorithms are used to detect the plastic bottles from the waste. And the arm located on the robot picks the plastic bottles. The discussion of various components is as follows.

2.1. Navigation using ROS

ROS is an open-source program, which has helped researchers as well as consumers operate robots. Another advantage of using ROS is that one can reuse libraries already created by other users and modify them according to requirements. The peer-to-peer topology of ROS makes network connectivity easier [18]. ROS is considered to be language-neutral as different programming languages can be mixed and modified to get the desired result. ROS has many tools to manage complicated tasks efficiently.

To use ROS, it is very important to get familiar with the basic terms used in the ROS which are nodes, messages, topics, and services. Nodes are processes that do computational work. A system comprises of many nodes and each node communicates with another node using messages. A message can be an integer, floating-point, Boolean or any other primitive type of data. The mechanism is such that when a node sends a message, it publishes the message to a specific topic. The node can use a subscribe/publish system to identify and connect to the desired topic. Multiple publishers and subscribers can be running for the same topic simultaneously, and one node can publish and/or subscribe to multiple topics. Another terminology of ROS is bags, which are very essential for storing ROS message data [19]. The ROS Master is very important as it assigns names and makes it possible for nodes to find other nodes and pass messages. The

Master includes the Parameter Server which stores data in the main location by key. The Master is responsible for the name service and saves information about the topics and services registration.

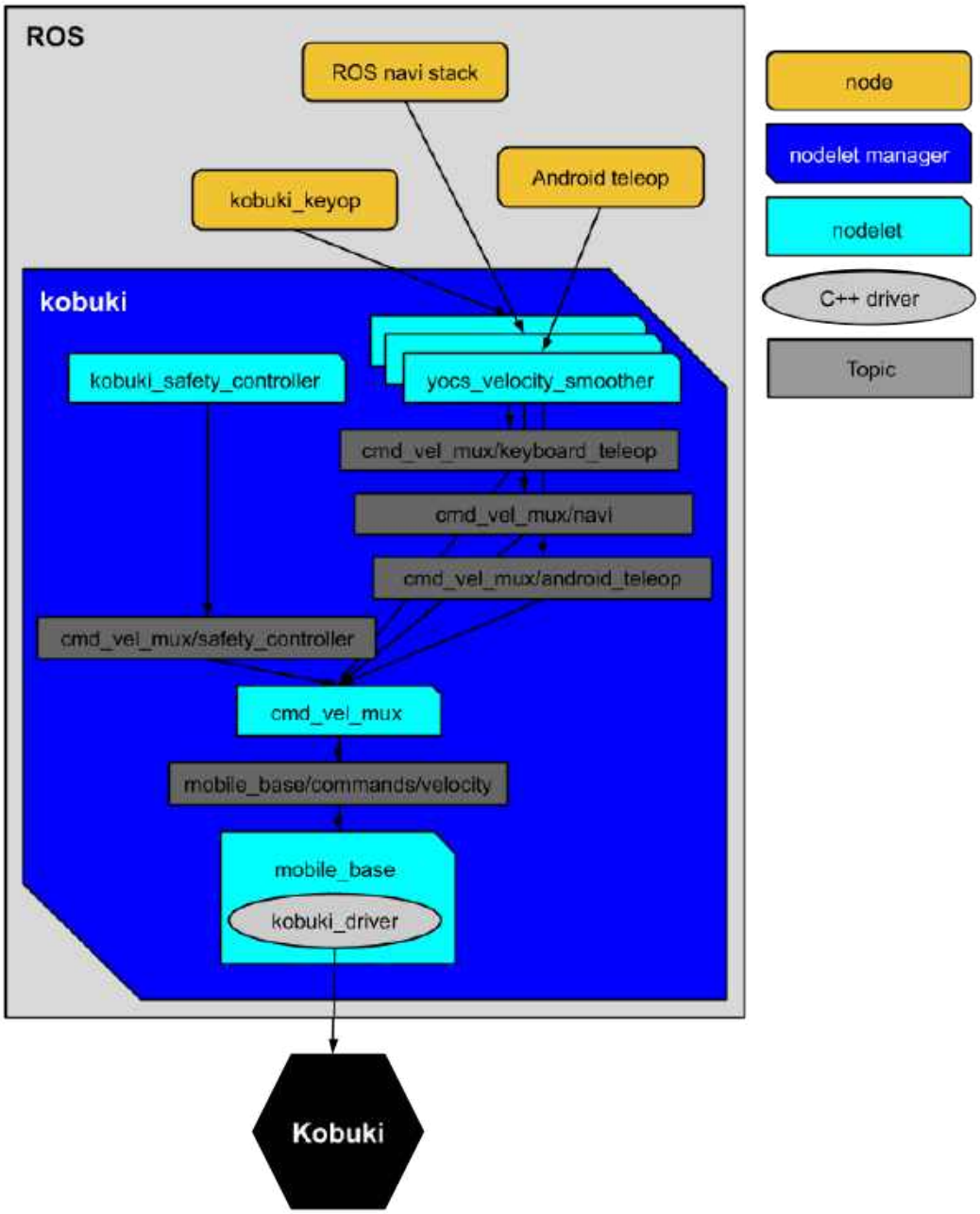

Figure 1: Kobuki's ROS package architecture [20].

For navigation using the Kobuki robot base in this work, the Kobuki_node package from [21] is used. This package provides access to turning on/off the Kobuki robots

motors and allows getting readings from wheel encoders and the gyro. As shown in Figure 1, the Kobuki has nodelets that allow multiple programmed files to run simultaneously [20].

For outdoor navigation, a GPS module is used to get the latitude and longitude of the robot's position. The latitude and longitude are extracted from the NMEA format data received from GPS readings. Also, a magnetometer from in an inertial measurement unit (IMU) measures the orientation of the robot.

2.2. Computer Vision

To detect and find the location of a plastic bottle with respect to the center of a camera module, computer vision techniques are used. This is a two-step process. Firstly, R-CNNs are being used in this work to detect plastic bottles. Secondly, a depth map of the image is created with the help of images taken with a stereo camera.

Both of the above processes provide the Cartesian coordinates of the plastic bottle in 3D space. To train the R-CNN used for plastic bottle detection, the transfer learning technique is used. In transfer learning, a CNN which is pre-trained on a large set of images is fine-tuned, with inputs of plastic bottle images to be detected, and corresponding labels. This method reduces the training time and size of the training data set [22] compared to training a model from scratch.

A CNN can be thought of as a feed-forward network that involves convolutional operations with learned filters. It is commonly used to evaluate images or videos in machine learning and computer vision. A typical CNN may have four kinds of layers: convolutional layer, rectified linear unit (ReLU) layer, pooling layer and fully connected (FC) layer [23]. The convolutional layer is used to carry out convolutions and is the first layer after the input layer whereas, the ReLU layer comes after the convolutional layer and introduces non-linearities in the network. The pooling layer is then used to downsample the data in the network to reduce the processing complexity. These layers may be replicated and are then followed by FC layers. Finally, the last layer assigns scores that determine the probability of the input belonging to each class. One of the pioneering models of CNN was LeNet [24]. Recently more complex variants such as

AlexNet [25], ZFNet [26], GoogLeNet [27], VGGNet [28] and ResNet [29] have been developed.

However, CNN is mostly used for classification and identifications of objects in an image. That's why region-based CNN (R-CNN) is used in this thesis. It works with images having multiple objects and determines the location of the target object [30]. R-CNN is used in applications where detection of the object, segmentation of an image and categorization of an image is required. The R-CNN uses selective search to recognize a region of interest (RoI) from the labeled images and it differentiates between the foreground and background using CNN [31].

Figure 2 shows the architecture of R-CNN. There are three steps to create this architecture which are listed as:

1. Possible objects are browsed from the input using selective search. This returns around 2000 proposed regions [31].

2. These proposed regions are tested using CNN.

3. The outputs of the CNN go through a support vector machine that identifies parts of items in an image. Then a fit bounding box is created around the identified object using linear regression.

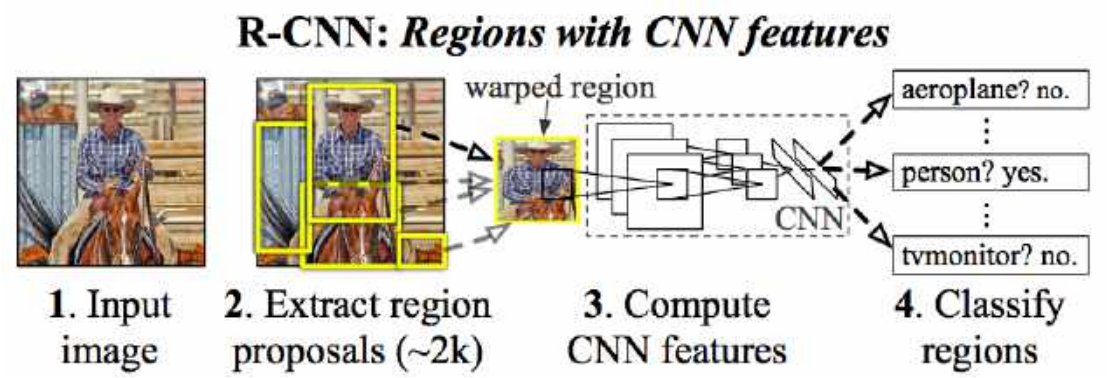

Figure 2: Architecture of R-CNN [30].

To train a R-CNN for plastic bottles with less training images, transfer learning is used. In this method, a CNN which is pre-trained on a large set of images is fine-

tuned with inputs of labeled plastic bottle images that are supposed to be detected. This method reduces the training time and size of the training data set [22] compared to training a CNN from scratch.

After detecting the object, its centroid is found and the average depth around the centroid is computed. This is achieved by computing a disparity map using the left and right images from a stereo camera [32]. Hence, this gives the coordinates of the plastic bottles in the 3D space.

2.3. Robotics Arm Kinematics

A robotic arm is used in this work for picking up plastic bottles once they have been detected. Note that the work in this thesis does not focus on the appropriate gripper development, or the gripping technique to be used to pick up a bottle. This is because gripper development is a separate area of research by itself, and is very far out of the scope of the current work. So the end-effector and bottles to be picked up in this work are magnetized with small magnetic attachments, which aid the picking up process. Therefore, to use a robotic arm for the purpose of picking up plastic bottles, the robot arm model needs to be known.

A robotic arm is made of multiple links which are connected by joints. These rigid bodies are fixed at the base and the last link is connected to an end-effector which is mostly the wrist. Each joint is connected to a motor. To understand the kinematics of the robotic arm, knowledge of transformation from one frame to another is essential. Every joint in the robotic arm has a frame which has a certain orientation and position. To pick and place any object using the end-effector, the initial orientation and position of the end-effector should be known. This pose of the wrist should be represented in terms of the base reference frame. This coordinate transformation can be done by using Euler angles, RPY (Roll, Pitch, and Yaw) Angles, Axis-angle Representation and Quaternions. The Euler angles based approach involves a series of rotations.

There are open chain manipulator where a single joint is joined by a link and closed chain manipulator where every link is attached to adjoining links by joints. In this thesis, an open-chain manipulator was used. It has n joints, $n+1$ links and the link

fixed to the ground in this manipulator is Link 0. Each link has its respective frame from Link 0 to Link n. The position and orientation of Frame n can be expressed in Frame 0 using Eq. (1), where each ${}^{n-1}_{n}A$ related the current joint's orientation to the previous joint. Eq. (2) is used to express the pose of end-effector to the Frame n and then Frame n's position and orientation to Frame of the base.

$$ {}^{0}_{n}T(q) = {}^{0}_{1}A\,{}^{1}_{2}A \ldots {}^{n-1}_{n}A \tag{1} $$

$$ {}^{e}_{b}T(q) = {}^{b}_{0}T\,{}^{0}_{n}T(q)\,{}^{n}_{e}T \tag{2} $$

As shown in Figure 3, Link $i-1$ and Link i are connected by Axis i.

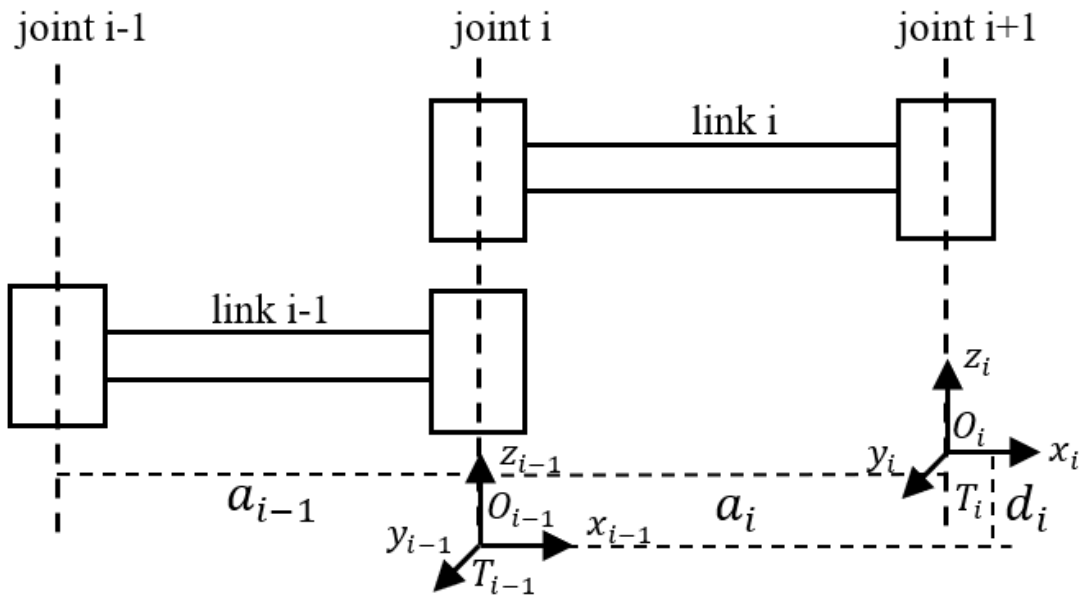

Figure 3: Denavit-Hartenberg notation.

There are rules to establish a Frame on a link i based on the Denavit-Hartenberg (DH) convention, which are as follows:

- The z_i axis should be selected along the Joint $i+1$ axis.
- The origin O_i should be at the intersection of z_i and the normal vector between z_{i-1} and z_i axes.

- The x_i axis should be selected as the normal vector between z_{i-1} and z_i axes going in the direction from Joint i to $i+1$.
- The y_i axis is selected using the right-hand rule.

After the frames are assigned to the links, the DH parameters such as link length, twist angle, joint angle and link offset represented as a_{i-1}, α_{i-1}, θi and d_i respectively, are used to represent the position and orientation of Frame i in terms of Frame $i-1$.

As shown in Figure 3, link length, a_{i-1}, is the length between z_{i-1} and z_i axes, measured along the x_i axis. The twist angle, α_{i-1}, is the angle between the z_{i-1} and z_i axes, measured about the x_i axis. Joint angle, θ_i, is angle between x_{i-1} and x_i axes, measured along the z_{i-1} axis. Link offset, d_i, is the distance between x_{i-1} and x_i axes, measured along the z_{i-1} axis from the origin of frame T_{i-1} [33].

The remaining parameters, α_{i-1} and a_{i-1} are based on the Link i configuration between the joints and are mostly constant. θ_i and d_i are the parameters that constantly change. d_i is variable when the Joint i is prismatic, which is a sliding joint, whereas, θ_i is variable when the Joint i is revolute, which is a rotating joint, [33].

A transformation matrix between frame T_{i-1} and frame T_i expresses the coordinates of joint i with respect to the previous frame. This is done by using Eq. (3) with known DH parameters. A product of each transformation matrix for individual links from the base of the robot arm to the last link joining the end-effector using Eq. (4), can give the complete system transformation matrix. This overall matrix is represented as ${}^0_nT$, which expresses the frame of the end-effector in terms of the frame of the robot arm base. This process of getting the position and orientation of the end-effector for given joint angles using Eq. (4), is called forward kinematics.

$$ {}^{i-1}_{i}A = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3} $$

$$ {}^0_nT = {}^0_1A^1_2A\ldots{}^{n-1}_nA = K(q) \tag{4} $$

The procedure for developing forward kinematics of a system, as mentioned in [34] is as follows:

1. Number the axes of each joints starting as z_0 to z_{n-1} where n is the number of joints.

2. Select the origin of axis z_0 as Frame 0. x_0 and y_0 are assigned based on right-hand frame rule. Mostly Frame 0 is selected as the base frame.

Repeat steps 3 to 5 for joints from $i = 1$ to $n - 1$ as mentioned in [34].

3. The origin O_i is located at the intersection of common perpendicular to z_{i-1} and z_i axes and z_i. If the Joint i is revolute, the O_i is located such that d_i is zero and if the Joint i is prismatic, the O_i is located based on the mechanical range of the joint.

4. Assign x_i axis from Joint i to $i+1$ direction with the common perpendicular between z_{i-1} and z_i.

5. Select y_i axis based on right-hand frame rule.

The following steps are from [34] to complete the process:

6. If the Joint n is revolute, align z_n with z_{n-1}. If the Joint n is prismatic, align z_n randomly. Then assign x_n using step 4.

7. Create table as shown in Table 4 for a_{i-1}, α_{i-1}, θ_i and d_i parameters for $i = 1$ to n.

8. Based on values from step 7, create transformation matrix $A_i^{i-1}(q_i)$ for $i = 1$ to n.

9. Calculate the pose of Frame n to 0 using Eq. (3).

10. Calculate the direct kinematics using Eq. (4).

However, for the robotic arm to move to a known position and orientation, inverse kinematics is used. For any reachable location, the inverse kinematic model returns the joint angles for the robotic arms motors.

Inverse kinematics is complicated due to the following factors:

- Due to the non-linearity of the problem, the solution does not always exist.
- There is a possibility of many solutions.
- There can be no solution available, especially if the pose of the end-effector is outside of the manipulator's workspace.

For the bottle collection system, a robotic arm with five degrees of freedom is used. The end-effector is magnetized. The reason behind using a magnet is to demonstrate the working of the overall prototype plastic bottle collection robotic system. This is justified from the perspective of the scope of this work, because developing a gripper that can grasp a bottle laying on the ground in any orientation is a separate problem of research. The focus of this work is not on developing grippers for a robotic arm, but on developing a robotic system that uses GPS coordinates to navigate to a known location where recyclable plastics are expected to be found (e.g. near a recyclable waste bin). And, then using RCNN based object detection to identify plastic bottles, and further use stereo vision for localization of the plastic bottle relative to the robot. The next chapter provides the details of the robotic system developed in this work.

Chapter 3: Developed Robotic System for Plastic Bottle Collection

This chapter presents the complete hardware and software structure for the developed robot.

3.1. Hardware

As shown in Figure 4, the hardware consists of a Kobuki base, a Raspberry Pi 3, a stereo camera, a robotic arm, a laptop, a GPS and an IMU. The developed robot has three main modules, they are i) the navigation control module, ii) the computer vision module and iii) the arm kinematics module. The navigation control module consists of a Raspberry Pi 3, GPS and IMU. The computer vision module consists of a stereo camera and a laptop computer. The arm kinematics module is a robotic arm. The stereo camera is mounted on the Kobuki base. The robotic arm is mounted above the stereo camera holder as illustrated in Figure 4. The laptop is kept on a Plexiglass sheet which is held by metallic rods attached to the Kobuki base. On the edge of the Plexiglass sheet, a GPS is attached. An IMU is mounted on a holder which is attached at the center of the laptop. This prevents the IMU from being affected by any stray magnetic fields produced by the equipment below the Plexiglass sheet.

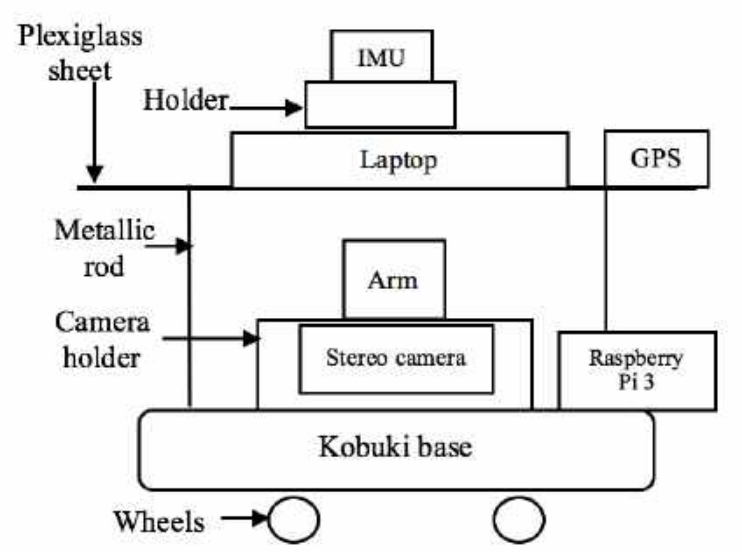

Figure 4: Schematic diagram showing hardware components of the developed robot.

3.2. Software Architecture

Figure 5 shows the software system schematics for the developed robotic system. The software system has two main modules which are the navigation control module and computer vision and arm kinematics (CVAK) module. The software used for the navigation control module is ROS and the CVAK module is programmed in MATLAB. The Raspberry Pi 3 in the navigation control module receives the GPS and IMU readings continuously.

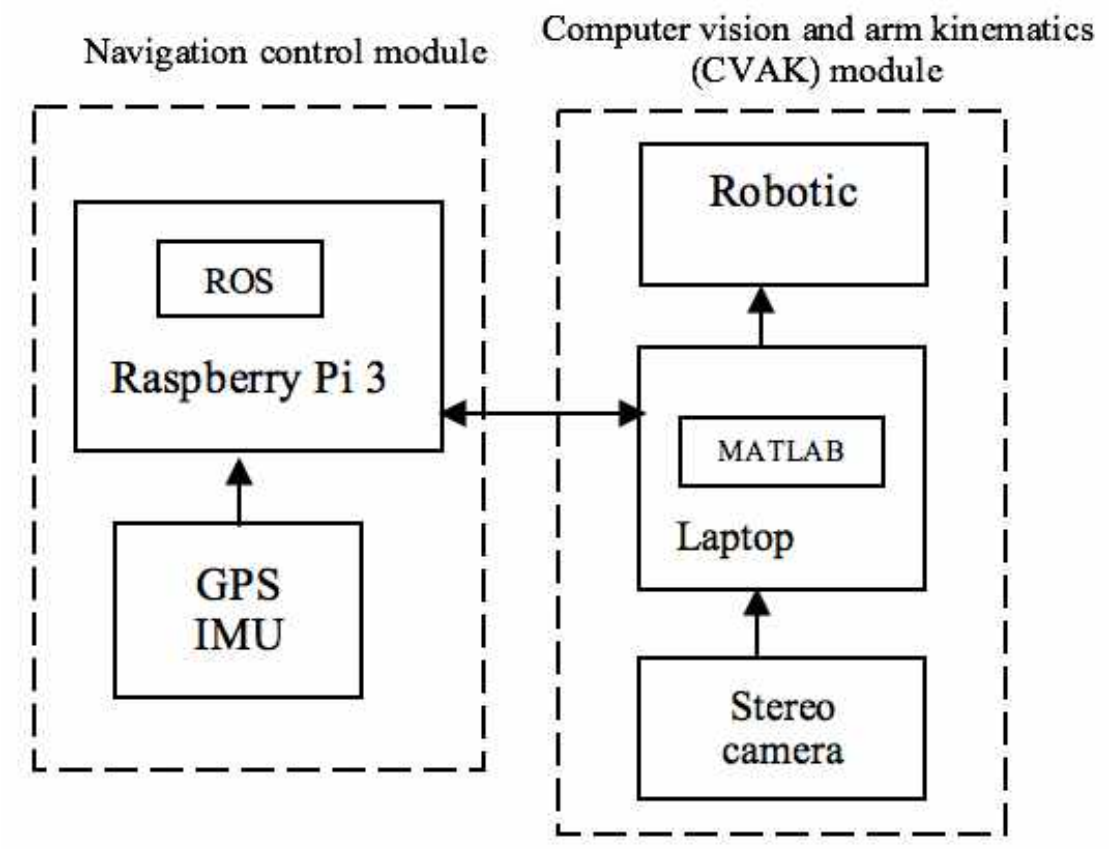

Figure 5: Schematic diagram showing software components of the developed robot.

Suppose (x_1, y_1) represents the commanded GPS goal location, and (x_s, y_s) represents the robots location at the start of GPS navigation, and the robots current location respectively. Let θ_s be the angle given by $tan^{-1}(\frac{y_1-y_s}{x_1-x_s})$, and similarly let θ represents the angle given by $tan^{-1}(\frac{y_1-y}{x_1-x})$. At every time step, the robot receives the GPS and

IMU readings which provide the coordinates of the robot (x, y) and also the robots current heading. All of the above-mentioned quantities are with respect to a common inertial reference frame. Then at each time step the GPS navigation code checks to see if the angle θ is within ± 5 degrees of the desired heading angle θ_s. If yes, the robot simply moves straight along its current heading. Or else, the code issues appropriate commands to the wheel motors such that the robot turns until the difference between θ and θ_s is less than ± 5 degrees. Once this is achieved, then the robot moves straight ahead as described above, until it reaches its commanded GPS goal location. To exit the GPS based navigation, and pass control to image-based navigation, the distance of the GPS goal location to the robots current position is compared at every time step to a preset threshold around 1.5 m. Once this threshold is reached, the robot stops. The information below relates to image-based navigation.

The motion control algorithm written in Python operates under the ROS framework. The images from the stereo camera are used for detection and 3D mapping of the plastic bottle which are processed in MATLAB on the laptop shown in Figures 4 and 5. A MATLAB program is developed to find the Cartesian coordinates of the identified plastic bottle w.r.t the center of the robots stereo camera. This information is then used by the navigation module to switch from GPS based navigation to image-based navigation for completing the task of moving to a plastic bottle. Once the robot reaches the bottle, then certain MATLAB commands are issued to the robotic arm to pick up the plastic bottle. The details of each module are discussed in the next chapter.

3.3. Methodology

This section discusses navigation, computer vision, and robotic arm kinematics modules in detail. It also represents the developed robot and its main components. Figure 6 shows the real-life hardware assembly based on the schematic presented in Figure 4. After assembly of the entire hardware, as shown above, the frames and DH matrix parameters, which are link length, twist angle, joint angle and link offset, are identified for the 5 degrees of freedom (DOF) manipulator. Moreover, calibration is done for each servo motor in the robotic arm separately to ensure that the joint angles

and corresponding pulse width commands required for accurate control are correctly known. Likewise, the onboard stereo camera is also calibrated using MATLABs Stereo Camera Calibration tool. The following section presents the results of experimentation using the above developed robotic plastic bottle collection system.

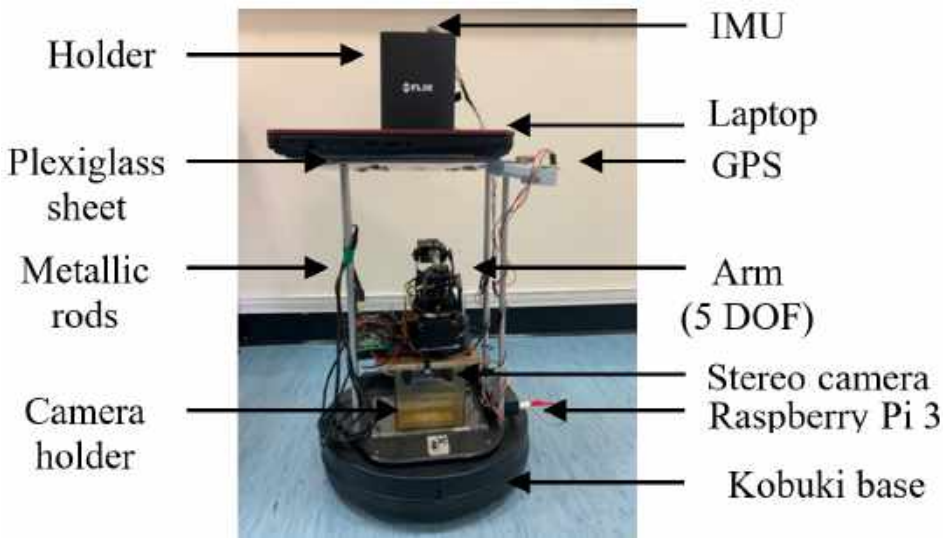

Figure 6: The developed robot.

3.3.1. Navigation module. For the navigation process, GPS and IMU are inetgrated with the Kobuki base. The GPS and IMU readings are continuously read and based on those reading the Kobuki motors are controlled as discussed in section 2.1. These latitude and longitude readings from the GPS were converted to Cartesian x and y coordinates using Eq. (5) and Eq. (6), where Lon_c, Lon_o, Lat_c, and Lat_o are current longitude, origin longitude, current latitude, and origin latitude respectively.

$$x = (Lon_c - Lon_o) \times 3.14 \times 63710 \times cos(Lat_c \times (3.14/180)) \tag{5}$$

$$y = (Lat_c - Lat_o) \times 6371000 \times (3.14/180) \tag{6}$$

To test the navigation module, ten tests were conducted in different outdoor locations. Figure 7 represents one of the tracks of the GPS based navigation outside the main building of the university with 2 meters stopping threshold value from the

final GPS given goal location. The blue line in Figure 7 is the actual navigation route, whereas the red line is expected navigation route. Figure 8 represents one of the tracks of the GPS based navigation outside the main building of the university with 1 meter stopping threshold value from the final GPS given goal location.

Figure 7: Preliminary testing of navigation module with 2 meters stopping threshold.

Figure 8: Preliminary testing of navigation module with 1 meter stopping threshold.

Figure 9 shows the GPS based navigation for two points in the parking area of university with 1 meter stopping threshold value from the both GPS points. The problem faced during the two GPS points test was that it took longer than expected to complete the test due to ± 5 meters error in GPS readings.

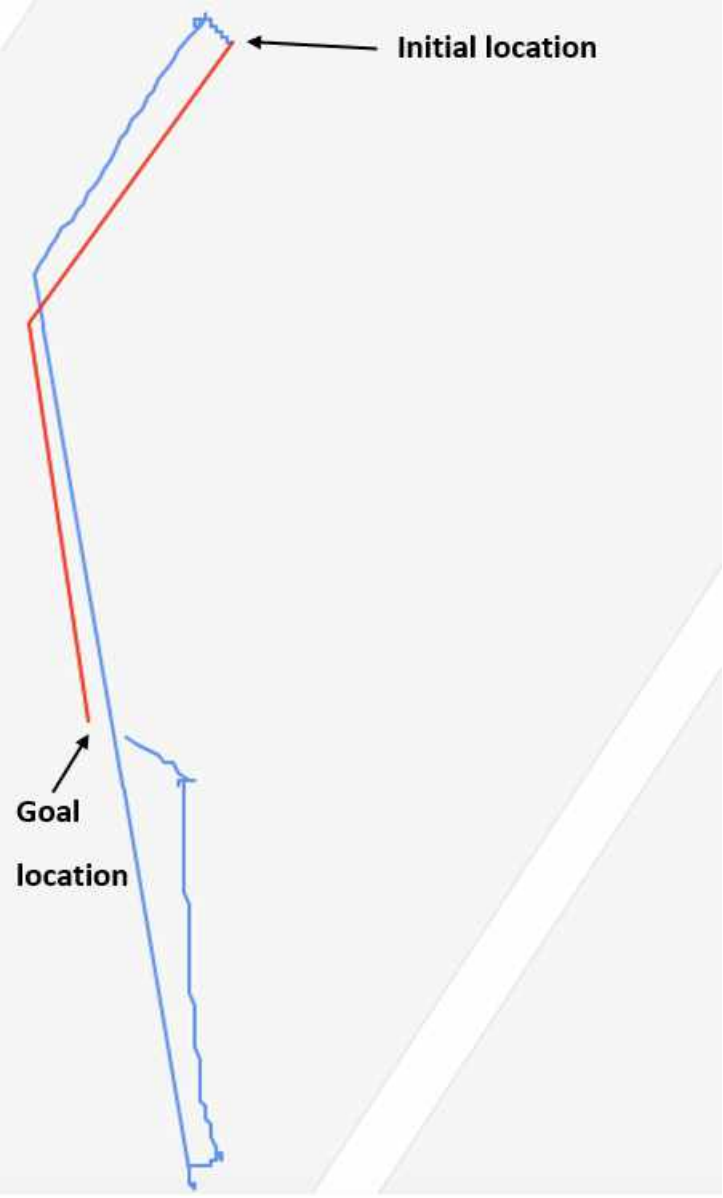

Figure 9: Preliminary testing of navigation module with two GPS points and 1 meter stopping threshold.

Furthermore, Figures 38 to 44 are the remaining preliminary tests to verify navigation module which can be found in Appendix.

Moreover, Table 1 contains the latitude and longitude of each test conducted to verify the working of the navigation module individually.

Table 1: Navigation module preliminary tests' initial and goal locations.

Outdoor test number	Initial latitude (degrees)	Initial longitude (degrees)	Goal latitude (degrees)	Goal longitude (degrees)	Distance (meters)
A1	25.0092222	55.007187	25.009323	55.0072628	13.57
A2	25.0092923	55.0072097	25.009323	55.0072628	6.35
A3	25.0092792	55.0072137	25.009323	55.0072628	6.95
A4	25.0092607	55.0072293	25.009323	55.0072628	7.71
A5	25.309293	55.490544	25.3093702	55.490608	10.74
A6	25.3093153	55.4905207	25.3093702	55.490608	10.69
A7	25.3093493	55.4906753	25.309394	55.4906573	5.29
A8	25.3092953	55.4905768	25.3092548	55.4905525	5.12
A9	25.3091063	55.4905692	25.3092012	55.4905947	10.85
A10	25.305981	55.4907145	25.3058673	55.4906882	12.91

Table 2 demonstrates the expected stopping distance of the robot from the goal location, the actual stopping distance of the robot from the goal location, and the distance error. Therefore, the average mean distance error of Table 2 is 0.09 meters. Thus, the navigation module is verified to be working properly.

Table 2: Navigation module preliminary tests.

Outdoor test number	Expected stop distance (meters)	Actual stop distance (meters)	Distance error (meters)
A1	2	1.93	0.07
A2	2	1.84	0.16
A3	2	1.89	0.11
A4	1	0.91	0.09
A5	1	0.88	0.12
A6	1	1	0
A7	1	0.97	0.03
A8	1	0.98	0.02
A9	1	0.87	0.13
A10	1	0.83	0.17

3.3.2. Computer vision module. The computer vision module is created using a two-step process. The first step is detecting the plastic bottle from the image and then finding the 3D coordinate values of the detected plastic bottle.

The plastic bottle detection process is completed using two steps. Firstly, the CNN is trained on an existing online image data set. And then, using transfer learning techniques, the pre-trained network's weights are updated using a few labeled images of plastic bottles.

We used a CIFAR-10 image data set to train our initial model. CIFAR-10 set consists of 10 different categories where each has 6,000 32x32 pixels size images. These 10 classes are of airplanes, birds, cars, cats, dogs, deer, horses, trucks, ships and frogs. Therefore, there are a total of 60,000 color images, out of which 50,000 images are used for training and 10,000 images are used for testing [35]. Using the image set of CIFAR-10, a CNN is trained with 50,000 images. This pre-trained network is then used to train for detecting plastic bottles using 80 training images. This way, we were able to train for the new class of plastic bottles with far fewer images.

The CNN used in this thesis consists of three main types of layers with a total of 15 layers, as shown in Table 3 and Figure 10. The first is the input layer which is created for 32x32 pixels size images because the existing online dataset CIFAR-10 is used for training which has images of size 32x32 pixels. Then we have a combination of convolutional layers, rectified linear units, and pooling layers. In the convolutional layers, the weights of filters are updated during training. The use of the ReLU layer is to introduce non-linearities to learn richer and more complex representations. The pooling layer is used throughout the network to downsample the information. We used three sets of convolutional, ReLu, and max-pooling layers, consecutively. The convolutional layer has 2-pixel padding to keep the output of the convolution the same as that of the input. The pooling is done on a 3x3 pixels neighborhood with strides of 2 pixels, such that 32x32 pixels are downsampled to 15x15 pixels dimensions. The final section of the network contains fully connected layers and a softmax loss activation function. We had one fully connected layer with 64 outputs. This is then followed with another fully connected layer with 10 nodes which decides the class of the input image. A

probabilistic value to an input image for each class is assigned at the fully connected layer output using a softmax function.

Table 3: Details of CNN layers.

	Name	Type	Activations	Learnables
1	imageinput 32x32x3 images with 'zerocenter' normalization	Image Input	32x32x3	-
2	conv 32 5x5x3 convolutions with stride [1 1] and padding [2 2 2 2]	Convolution	32x32x32	Weights 5x5x3x32 Bias 1x1x32
3	relu ReLU	ReLU	32x32x32	-
4	maxpool 3x3 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	15x15x32	-
5	conv_1 32 5x5x32 convolutions with stride [1 1] and padding [2 2 2 2]	Convolution	15x15x32	Weights 5x5x32x32 Bias 1x1x32
6	relu_1 ReLU	ReLU	15x15x32	-
7	maxpool_1 3x3 max pooing with stride [2 2] and padding [0 0 0 0]	Max Pooling	7x7x32	-
8	conv_2 64 5x3x32 convolutions with stride [1 1] and padding [2 2 2 2]	Convolution	7x7x64	Weights 5x5x32x64 Bias 1x1x64
9	relu_2 ReLU	ReLU	7x7x64	-
10	maxpool_2 3x3 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	3x3x64	-
11	fc 64 fully connected layer	Fully Connected	1x1x64	Weights 64x576 Bias 64x1
12	relu_3 ReLU	ReLU	1x1x64	-
13	rcnnFC 2 fully connected layer	Fully Connected	1x1x2	Weights 2x64 Bias 2x1
14	rcnn Softmaxsoftmax	Softmax	1x1x2	-
15	rcnnClassification crossentropyex with classes 'PlasticBottle' and 'Background'	Classification Output	-	-

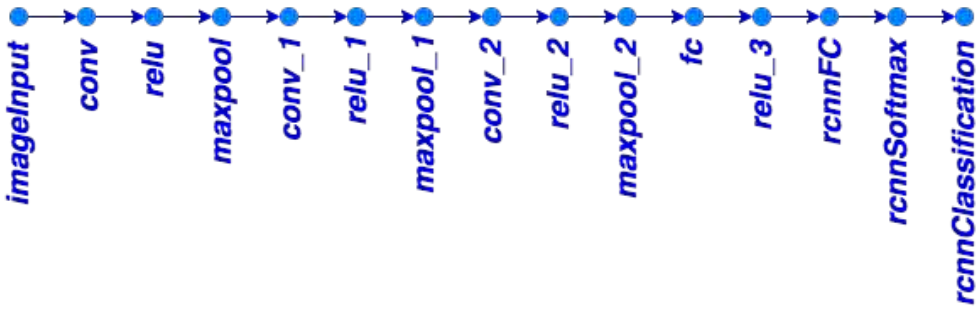

Figure 10: Layers of CNN.

The weights of the network are set using a random distribution with a 0.0001 standard deviation. This CNN is trained on CIFAR-10 with Stochastic Gradient De-

scent with Momentum (SGDM) training algorithm using 0.001 initial learning rate. Also, this initial learning rate decreases each time after 8 epochs are completed, which means the complete dataset has passed through the entire neural network process eight times during the testing. The maximum number of epochs was 40. After the training is completed, the CNN is verified by checking the filters' weights of the first layer as seen in Figure 11. The weights of the neural network changes during the training process. These weights are the filter of the first layer of the network, each filter is used to extract some edge of arch like features of the training dataset, plastic bottles similar to Figure 12. Since the network architecture was limited to having only 32 filters, the 32 different features of the training dataset are illustrated in Figure 11. Furthermore, once the visual filters' weights of the first layer have some edge like structure as observed in Figure 11, the network does not need more training. Therefore, the CNN is ready to be tested on the test data.

Figure 11: First layer filters structure.

For the second part of creating R-CNN, ground truth data for the plastic bottle is collected. Around 80 images of the plastic bottles in the different backgrounds are taken. The location of the plastic bottle, which is the RoI, in the images is then labeled using a labeling tool in MATLAB as shown in Figure 12. This ground truth data is then used to further train the pre-trained model for object detection.

Figure 12: RoI labelled Plastic Bottle in an image.

The original CNN for CIFAR-10 set is used to train R-CNN for the Plastic Bottle detection. This network classifies images into two categories: Plastic Bottle and general

items. Using the data from the ground truth table, the weights of the input network are tuned at the time of training. There are positive and negative training samples which are used during the training. For this thesis, the range of positive overlap was set from 0.5 to 1.0 and negative overlap has range from 0 to 0.3. When the image is being tested using this R-CNN, pixel locations of the bounding box in the image containing plastic bottles are returned along with a score that determines the confidence of detecting the plastic bottle.

The original CNN for CIFAR-10 set is used to train R-CNN for the Plastic Bottle detection. This network classifies images into two categories: Plastic Bottle and general items. Using the data from the ground truth table, the weights of the input network are tuned at the time of training. There are positive and negative training samples which are used during the training. For this thesis, the range of positive overlap was set from 0.5 to 1.0 and negative overlap has range from 0 to 0.3. When the image is being tested using this R-CNN, pixel locations of the bounding box in the image containing plastic bottles are returned along with a score that determines the confidence of detecting the plastic bottle.

After identifying the location of plastic bottles in the images, we needed to determine how far are the plastic bottles are from the robot. This is achieved by using a stereo camera, where two cameras were placed 0.4 meters distance apart. To start this process, the stereo camera is calibrated. There are two cameras in the stereo system, the calibration process measures the pose of the second camera with respect to the first camera. The calibration process is as follows:

1. A checkerboard image, which is not in square shape, is printed. This pattern should have an even number of squares on one side and an odd number of squares on the other side. One side of the pattern should have black boxes in the corners whereas the other side should have white boxes in the corner. The longer side of the image should be in the direction of x as shown in Figure 13. This helps to recognize the pattern's orientation. We fix this printed checkerboard square pattern on a flat surface for accurate calibration. We follow the calibration instructions from [36].

2. We then set up the camera such that the patterns in Figure 13 do not go out of focus and do not zoom in or out as this may affect the focal length.

3. We take more than 10 images for better calibration. We also take the image of the checkerboard around the same distance at which the object to be detected is expected to, be from the camera. Moreover, we do not tilt the checkerboard more than 45 degrees with respect to the camera's vertical axis. The whole checkerboard is required to appear in the captured image. Also, the x-axis and y-axis of the checkerboard does not have to be aligned with the robot's camera.

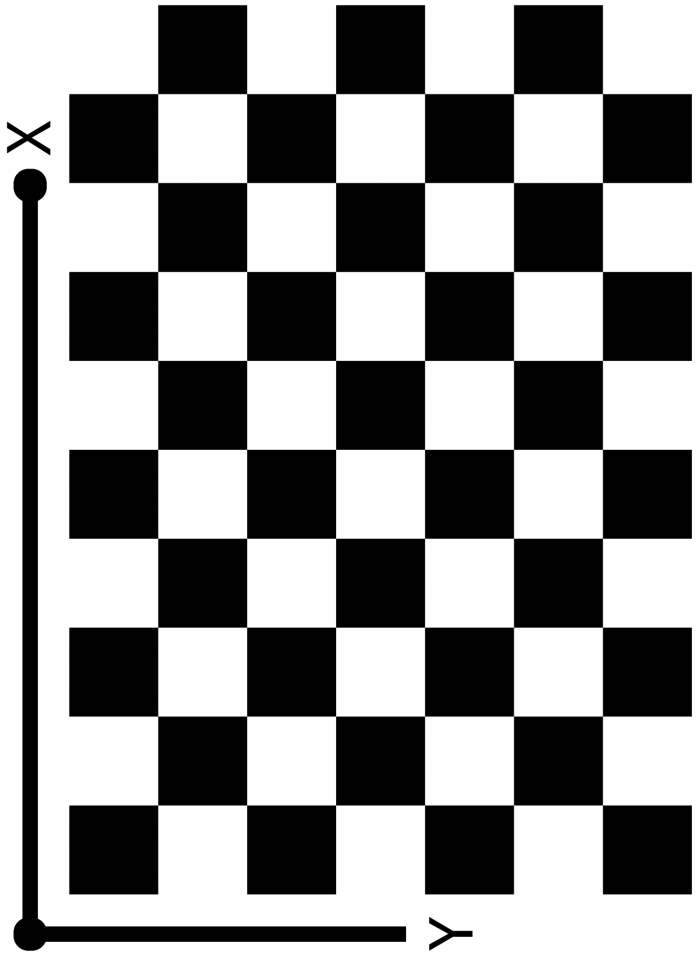

Figure 13: Checkerboard pattern for stereo camera's calibration.

After completing the calibration process using a "StereoCameraCalibrator" app in MATLAB, the camera extrinsic features can be seen in Figure 14. The rectangular boxes in Figure 14 represent different pose of checkerboard pictures that were used for calibration, whereas the blue and red camera like shapes are the stereo camera. The extrinsic features convert the 3D coordinates of the real world to the 3D coordinates of a camera.

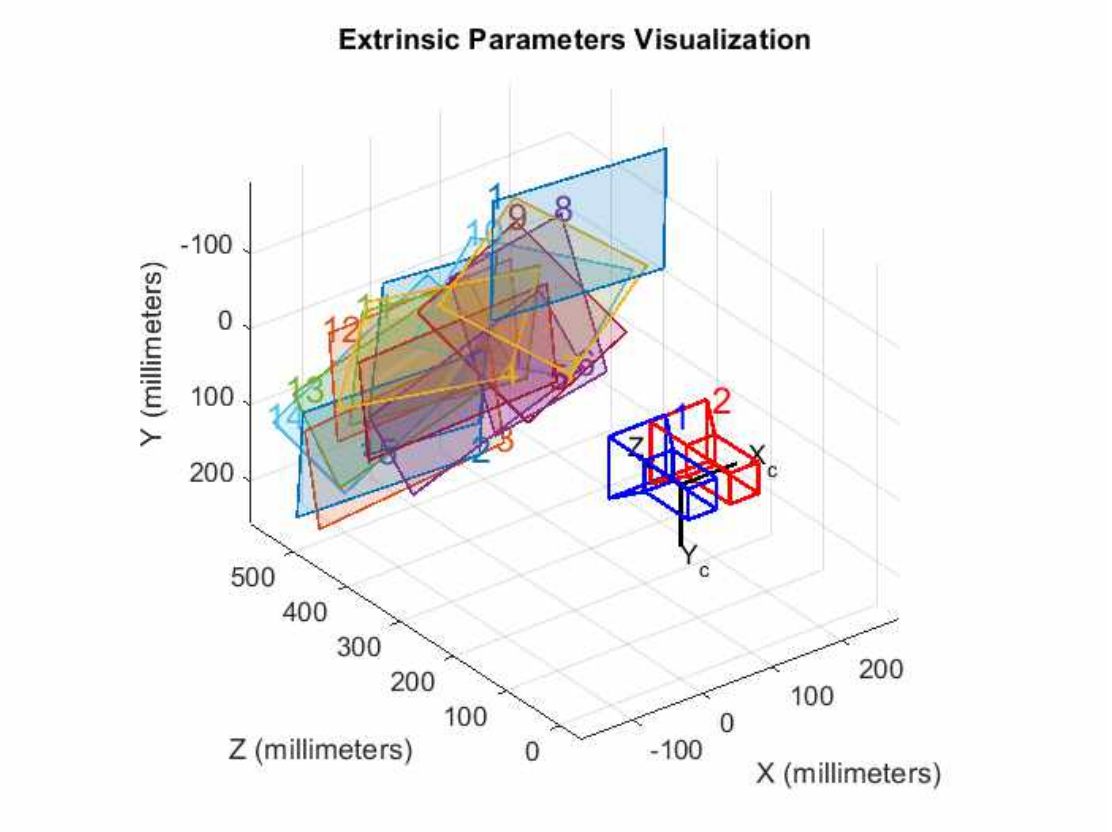

Figure 14: Extrinsic parameters of the stereo camera.

In order to create a 3D image of the detected plastic bottle, the images from the stereo camera are split into left and right images. These images are then rectified to have lines that are epipolar and aligned in a row. This helps to create a disparity map easier. These rectified images in anaglyph give 3D effect when seen with the red-cyan glasses as seen in Figure 15.

The disparity map which is shown in Figure 16 is then used to construct a point cloud as shown in Figure 17. This point of cloud represents each pixel of the disparity map to a 3D real-life value.

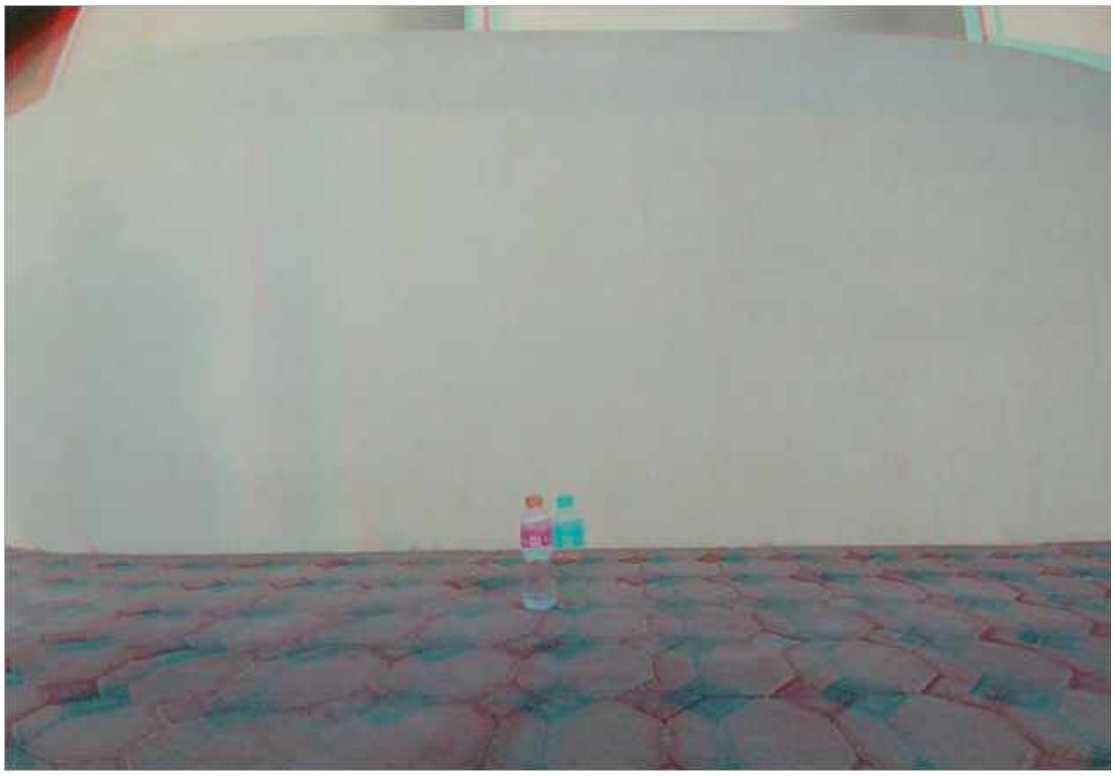

Figure 15: Rectified images in red-cyan anaglyph.

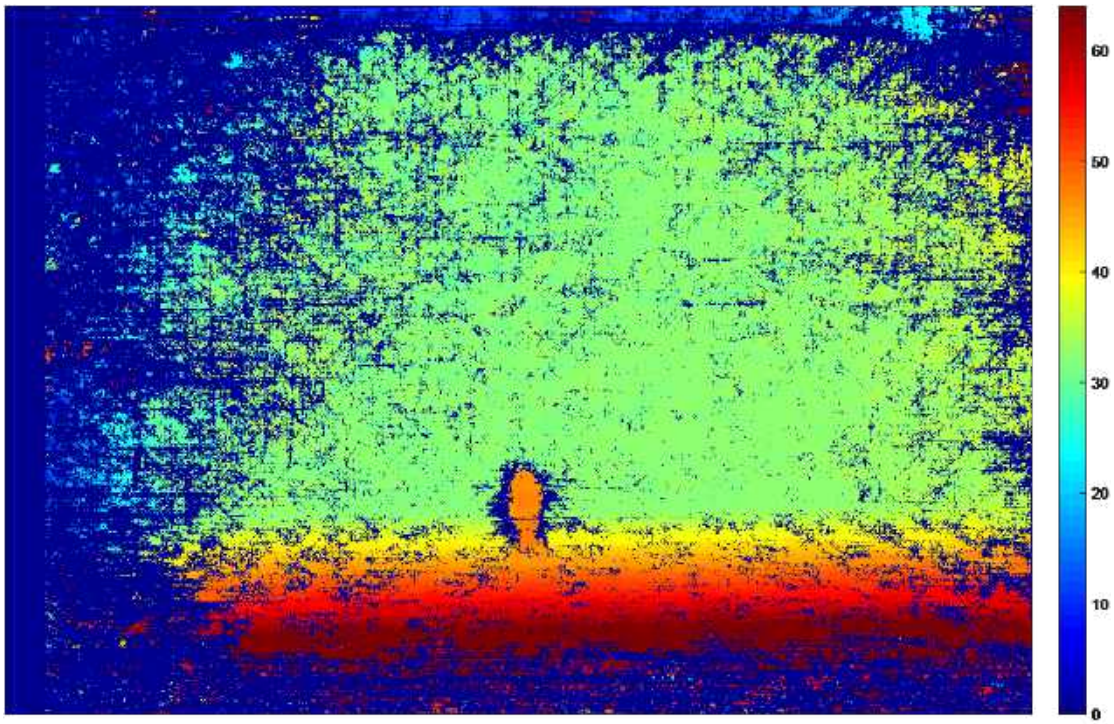

Figure 16: Disparity map.

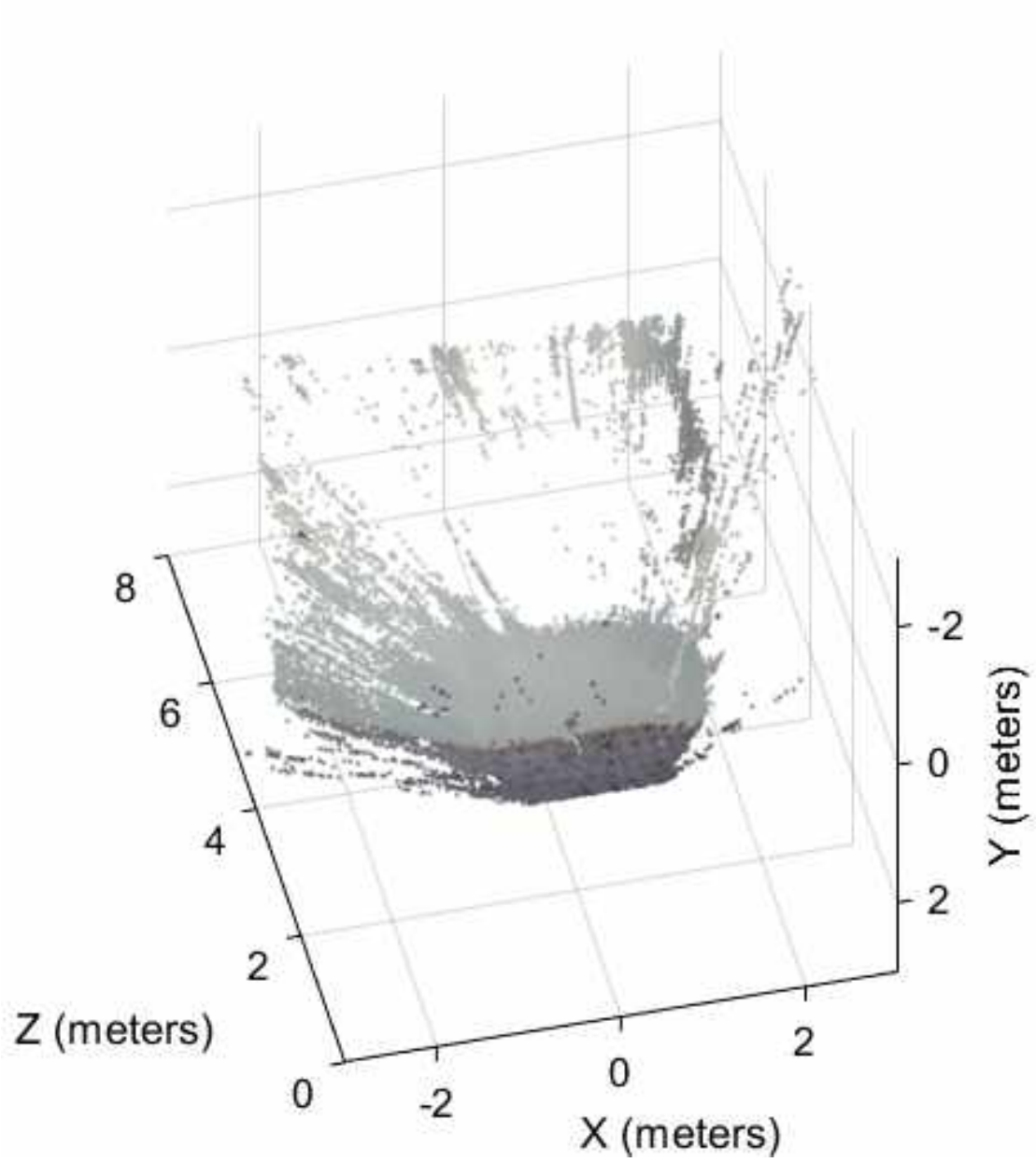

Figure 17: Reconstructed 3D values of a scene.

Therefore, after the bounding box containing the plastic bottle in a test image is detected as in Figure 18, the pixel value of the center of the box is determined. After that using the disparity map, the distance of the bottle from the camera is calculated in meters which is illustrated in Figure 19. Thus, the actual distance from the plastic bottle during this computer vision test was 1.12 meters, whereas the detected distance of the plastic bottle from the camera was 1.10 meters. Therefore, there was 2 cm error in this preliminary test.

Figure 18: The detected plastic bottle.

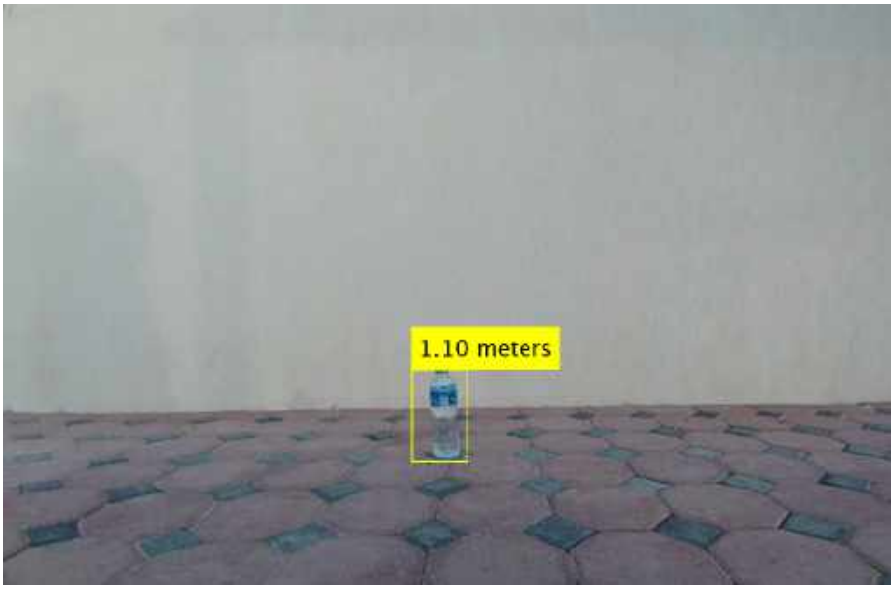

Figure 19: The depth of the plastic bottle from the camera.

3.3.3. Robotic arm kinematics module. For the robotic arm, a robotic arm produced by Lynxmotion with five degrees of freedom was used. After assembling the hardware of the arm, the DH parameters for the 5 DOF manipulator were identified. Figure 20 and Figure 21 show each joint and the assigned frames of reference for each

joint respectively. The fifth motor can be used to grab bottles in different orientations by utilizing an appropriate end-effector.

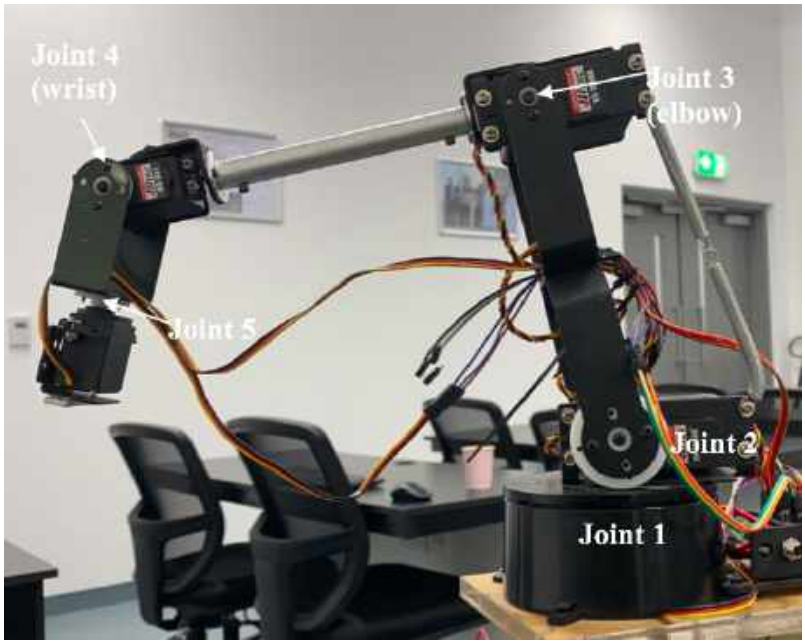

Figure 20: Labelled assembled robotic arm.

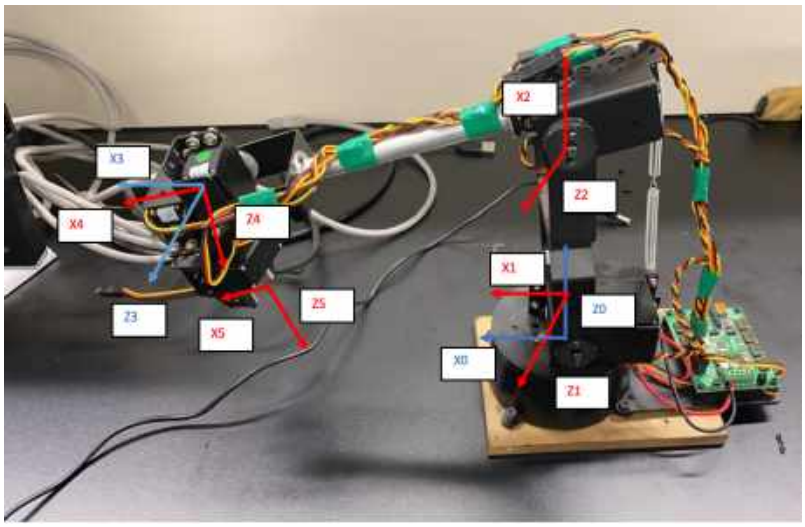

Figure 21: Assembled robotic arm.

Here, the DH convention was followed to set the x and z axes for each joint frame. The parameters, shown in Table 4, were obtained by considering the twist be-

tween each z axis for α, measuring the lengths for d_i and a_i, and the joint angles are free to change.

Table 4: DH parameters for each joint frame of the 5 DOF arm robot.

Joint i	Θ_i	d_i	a_i	α_i
1	Θ_1	0.07	0	$-\frac{\pi}{2}$
2	Θ_2	0	0.147	0
3	Θ_3	0	0.185	0
4	Θ_4	0	0	$\frac{\pi}{2}$
5	Θ_5	0.152	0	0

The robot with DH parameters defined in Table 4 was simulated in MATLAB using Peter Corke's Robotics Toolbox [37] as shown in Figure 22. This was done to verify if the simulated robotic arm was similar to the actual robotic arm. Limits were defined for the joint angles to restrict the angles to $\Theta_i \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ for all i and offsets for joints 2, 3, and 4 are defined as $-\frac{\pi}{2}$, $-\frac{\pi}{2}$ and $-\frac{\pi}{2}$ respectively for the arm to be in the configuration shown in Figure 22 at the home position, that is when the angles of all the joints are zero. To make sure the angles of all joints were zero, an offset was introduced.

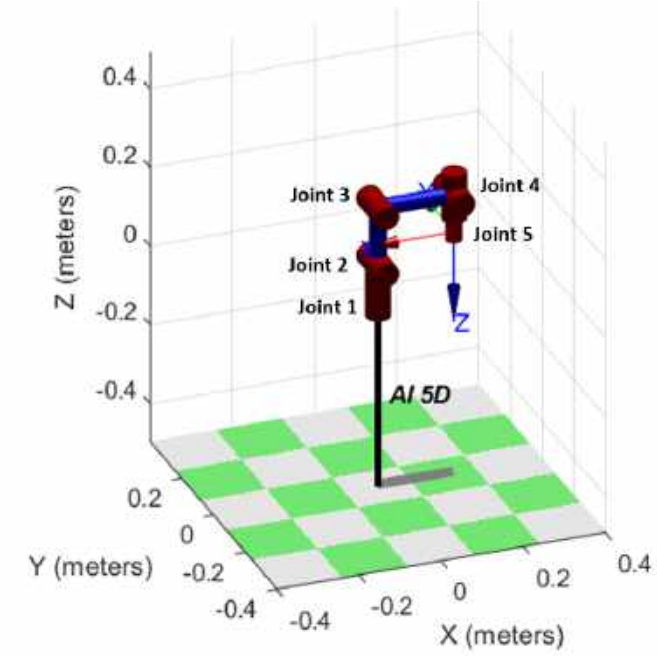

Figure 22: Simulated robotic arm [37].

3.3.4. Operation of the developed plastic bottle collection robot. Figure 23 briefly explains the complete operational process of the developed robot.

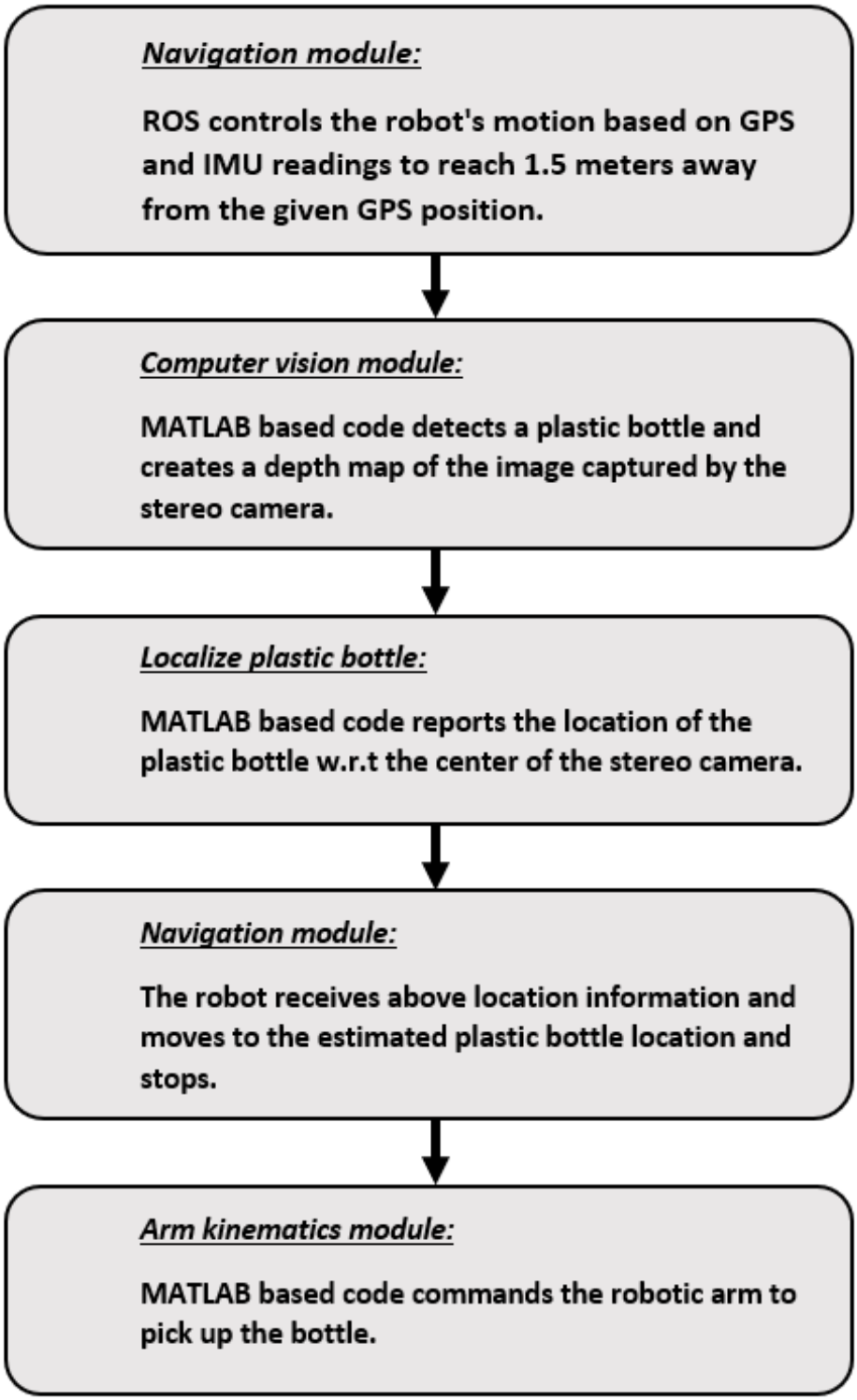

Figure 23: Schematic of the general operational process of the developed robot.

Firstly, if the test is outdoors, the GPS based navigation module is activated. Then, as shown above, the Kobuki base moves until it is 1.5 meters away from the given GPS location. Further, the computer vision module is used to detect and estimate the distance of the plastic bottle from the center of the stereo camera. The general process includes two steps, which are as described in Section 2.2. However, the detailed steps are as follows. In the beginning, a plastic bottle is detected by the R-CNN and the pixel location of the center of the box bounding the detected plastic bottle is saved. Then a lookup operation is performed at the above saved pixel location, within the depth map generated using the image of the scene containing the detected plastic bottle. This gives the distance of the plastic bottle from the center of the camera. Knowing the camera placement w.r.t the center of the Kobuki base, the robot is made to move straight ahead, till the robot reaches 0.1 meters away from the estimated location of the plastic bottle straight ahead. Because the motion executed at this phase is simply a straight line motion, encoder readings are used to determine when the robot should stop moving.

Once, the robot reaches the desired location 0.1m away from the detected plastic bottle, code in MATLAB commands the robotic arm to pick the plastic bottle up. It is also worth noting that the length of the arm (elbow to wrist) as seen in Figure 21 is 15 cm. It is also noticed that as long as the bottle is in the field of view of the stereo camera when the bottle is 1.5 m away from the center of the robot, moving straight ahead following the last known orientation of the robot at the end of GPS navigation, vision-based guidance successfully leads the robot close enough to the bottle so that the arm can pick it up. In case, GPS navigation leaves the robot 1.5 m away from the bottle and orientated such that the bottle is not in the view of the stereo camera then it is very easy to update the software to include code that rotates the robot around its position until the bottle is visible, identified, its distance from the center of the camera detected, and then the robot can simply move straight ahead.

Chapter 4: Experimental Results

In this chapter, indoor and outdoor experiments and their results are presented. Before conducting the experiments, the arm and the plastic bottle are attached with small piece of magnet as shown in Figure 24.

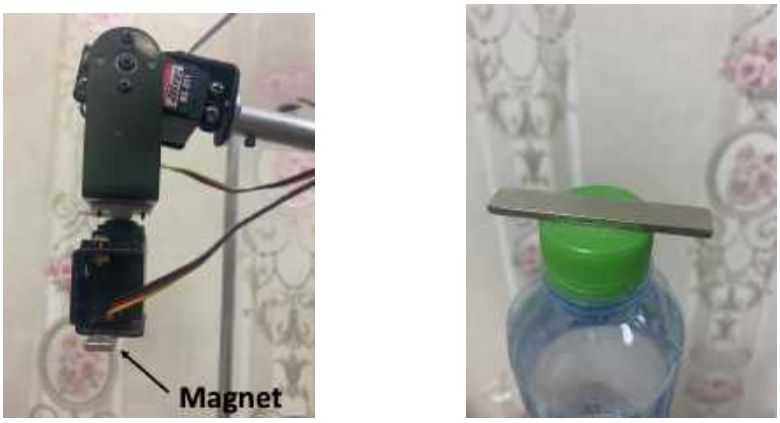

Figure 24: Magnetized arm and bottle cap.

4.1. Indoor testing

Figures 25, 26 and 27 are related to an indoor test, where a bottle is seen next to a trash can, along with some other recyclable plastic items.

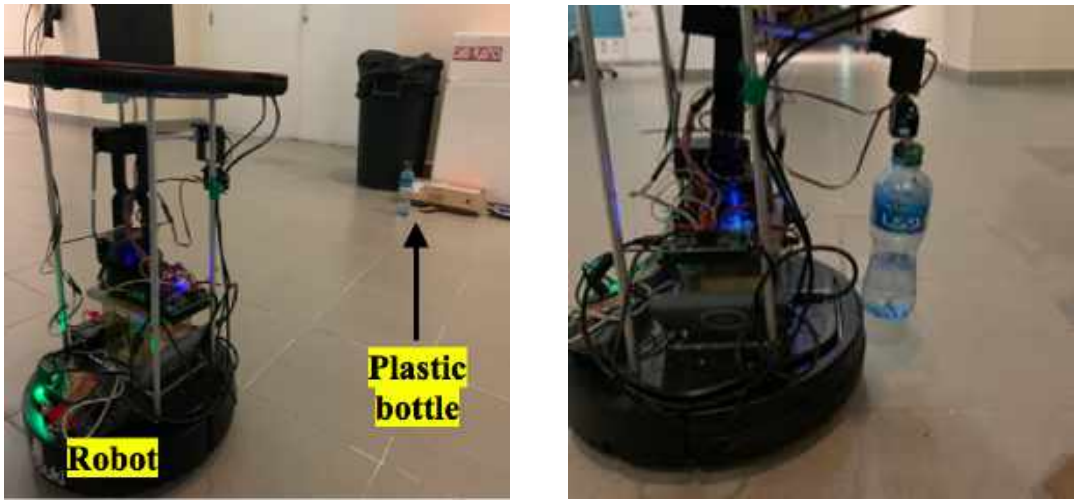

Figure 25: (a) Left: Initial position of the developed robot away from plastic bottle indoors (b) Right: Final stage of indoor testing showing successful bottle pickup.

Figures 26 and 27 show the results obtained from the computer vision module. Figure 26 is the disparity map for the indoor Test 1 and Figure 27 illustrates the estimated distance of the plastic bottle from the center of the camera, after accounting for the fixed offset errors.

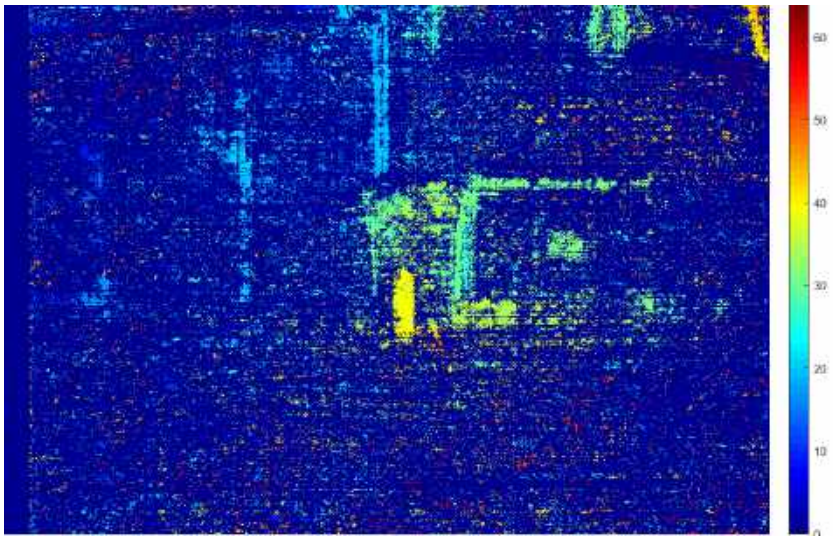

Figure 26: Disparity Map.

Figure 27: Detected plastic bottle during the indoor test shown in Figure 25.

As seen in the left image in Figure 25, the plastic bottle is 1.45m away from the robot and 1.53m away from the center of the camera. The right image in Figure 25

shows the bottle is picked up using a magnetized arm. The distance of the bottle obtained from the computer vision processing module is not very accurate as there is an error of 0.22 m. The computer vision module estimated the plastic bottle to be 1.31 m away as shown in Figure 27 whereas, the actual distance is 1.53 m. However, this discrepancy turns out to be a constant offset, hence this error was fixed by adding an offset of 22 cm to the detected position for remaining tests. The robotic arm also has small errors in the joint angles, i.e. the actual angle is not exactly equal to the commanded angle, however, this does not affect the picking up process because the magnetized arm picks the bottle if it is in the vicinity of the magnetized bottle cap. Thus it is clear from the above figures that the preliminary indoor test was successful.

This indoor test was repeated 10 times to check its success rate which is shown in Table 5. Thus, the successful pickup using this robot in the indoor scenario was 80%.

Table 5: Indoor tests' success rate.

Indoor test number	Succeeded
1	Yes
2	Yes
3	Yes
4	Yes
5	No
6	No
7	Yes
8	Yes
9	Yes
10	Yes

Furthermore, the actual distance of the plastic bottle from the camera, the estimated distance of the plastic bottle from the camera, and the distance error for each 10 indoor test is illustrated in Table 6. Also, the average mean distance error for the indoor tests is 0.253 meters. Figure 45, and Figure 46 are some other indoor tests in different setup environment, whereas Figure 47, and Figure 48 are images of some of the remaining indoor tests in another experimental setup location. These images of indoor tests can be found in the Appendix of the thesis.

Table 6: Indoor tests' details.

Indoor test number	Measured distance (meters)	Detected distance (meters)	Distance error (meters)
1	1.7	1.41	0.29
2	1.7	1.43	0.27
3	1.7	1.41	0.29
4	1.7	1.39	0.31
5	1.7	1.4	0.3
6	1.7	1.44	0.26
7	1.7	1.45	0.25
8	1.7	1.4	0.3
9	1.45	1.33	0.12
10	1.45	1.31	0.14

4.2. Outdoor testing

Figures 28, 29 and 30 show the results of an outdoor test where the robot is initially guided by GPS based navigation to a location where recyclable waste is expected, and the final stage of navigation is performed using image processing.

Figure 28: (a) Left: Initial position of the developed robot away from plastic bottle outdoors (b) Right: Final stage of outdoor testing showing successful bottle pickup.

The left and right images in Figure 28 show the initial and final stage of plastic bottle collection respectively. Figure 29 is the disparity map used to detect the depth of the plastic bottle ahead of the robot, and Figure 30 shows the estimated distance of the plastic bottle from the center of the camera once the robot reaches a given GPS point near an area where recyclable plastic waste is expected to be found.

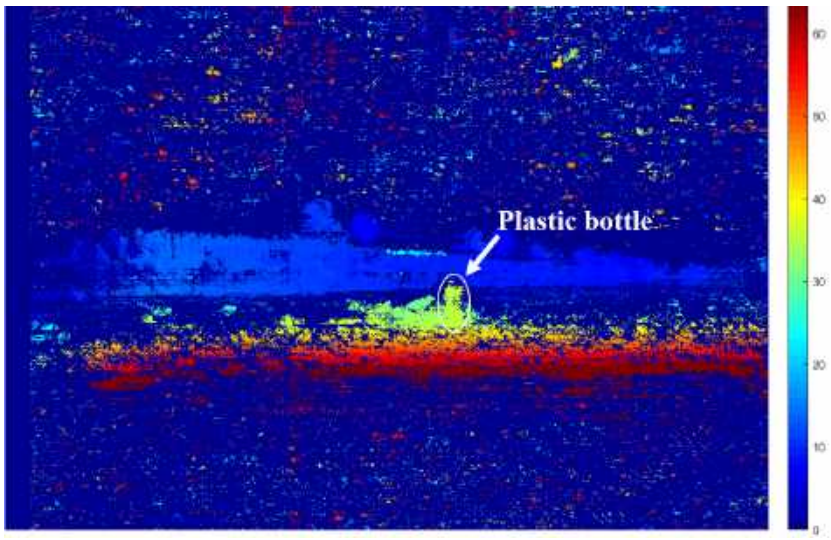

Figure 29: Disparity Map for outdoor Test B1.

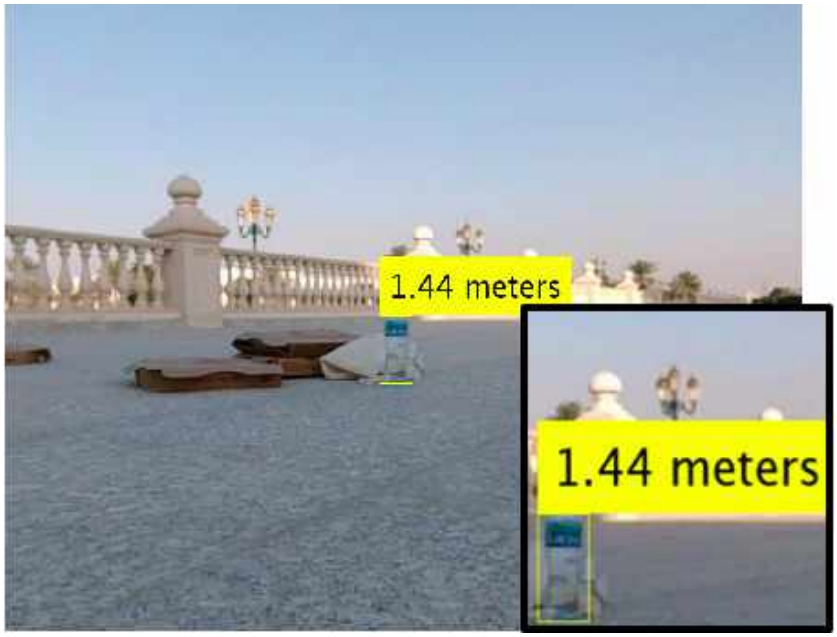

Figure 30: Detected plastic bottle.

Figures 31 and 32 summarize the results of two outdoor navigation tests on the robot, Test B1 and Test B2 respectively. The green graph represents the position of the robot during GPS based motion control to follow the assigned path shown in blue, and the orange graph represents the stereo camera based motion.

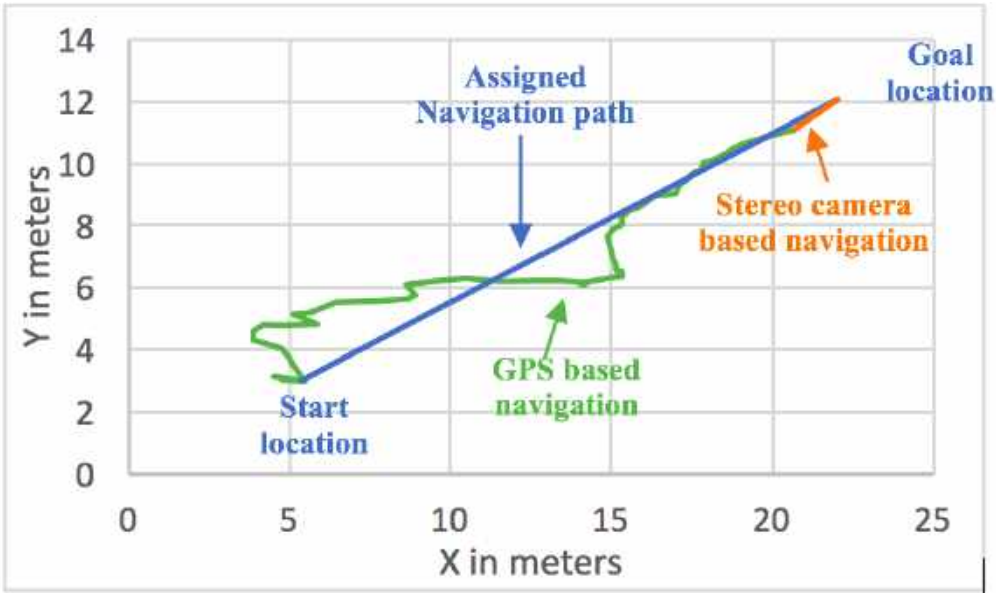

Figure 31: Navigation map of Test B1.

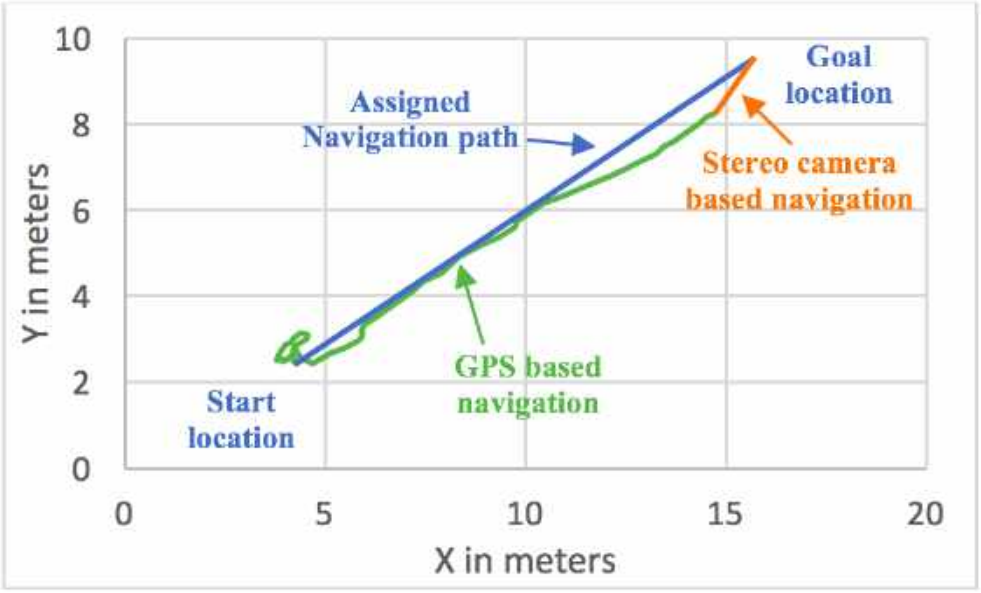

Figure 32: Navigation map of Test B2.

The results of navigation based on the computer vision for Test B1 and Test B2 are zoomed and shown in Figure 33 and Figure 34 respectively for clarity.

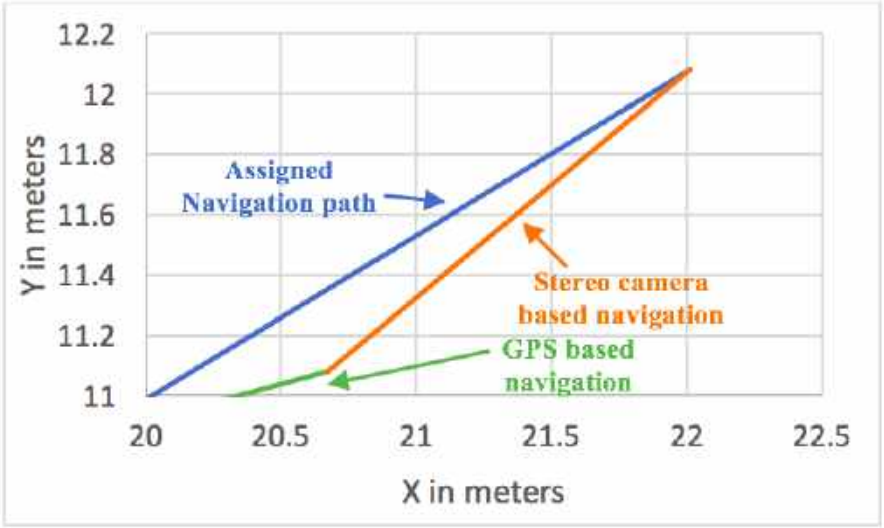

Figure 33: Navigation based on computer vision, for outdoor Test B1.

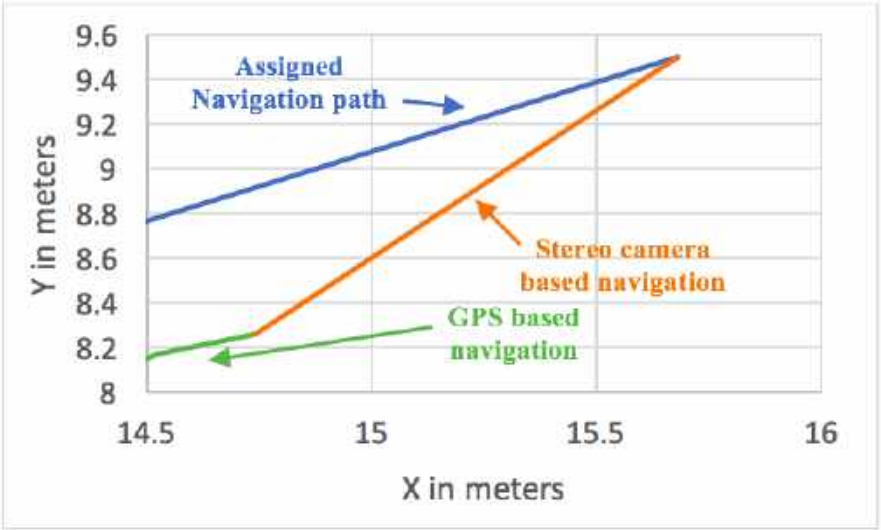

Figure 34: Navigation based on computer vision, for outdoor Test B2.

As visible in Figures 33 and 34 during both tests, the GPS navigation does not end with the bottle straight ahead of the robot, but as long as the camera can see the

bottle, the image processing based navigation module is able to guide the robot to the bottle. Also, in each case, the GPS based navigation goal is assigned to be about 1.5 m away from the position of a plastic bottle.

Figures 35 and 36 are the distance of the robot from the plastic bottle as it performs GPS based navigation, before the robot starts the camera based navigation for Test B1 and Test B2 respectively.

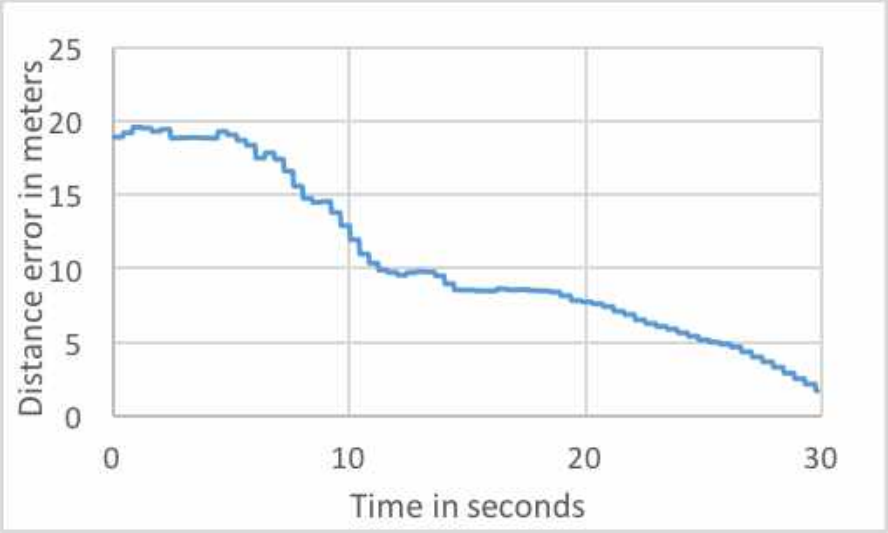

Figure 35: Distance error for outdoor Test B1.

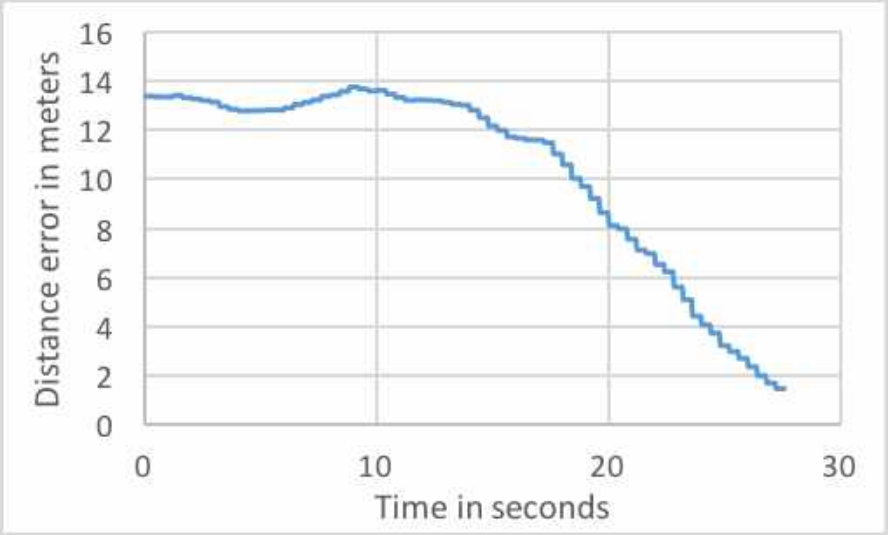

Figure 36: Distance error for outdoor Test B2.

This outdoor test was repeated 6 times to check its success rate which is shown in Table 7. Thus, the successful pickup using the developed robot in an outdoor setting was 67%.

Table 7: Outdoor tests' success.

Outdoor test number	Succeeded
B1	Yes
B2	Yes
B3	Yes
B4	No
B5	Yes
B6	No

Table 8 contains initial and final location of the developed robot in six outdoor tests. The location for all the outdoor tests was outside the library of American University of Sharjah, Sharjah, United Arab Emirates.

Table 8: Outdoor tests initial and goal latitude and longitude.

Outdoor test number	Initial latitude (degrees)	Initial longitude (degrees)	Goal latitude (degrees)	Goal longitude (degrees)	Distance (meters)
B1	25.3119543	55.4918777	25.3120357	55.4919697	12.94
B2	25.3119078	55.4919067	25.3120357	55.4919697	15.57
B3	25.3119185	55.4919	25.3120357	55.4919697	14.8
B4	25.3119552	55.491873	25.3120357	55.4919697	13.21
B5	25.311907	55.491913	25.3120357	55.4919697	15.4
B6	25.3119545	55.4918758	25.3120357	55.4919697	13.06

Moreover, Table 9 refers to image based navigation and displays actual distance of the plastic bottle from the developed robot, the estimated distance of the plastic bottle from the developed robot using navigation module, and the distance error. The '-' in Table 9 represents that Test B4, and B6 were not successful as the computer vision module failed to recognize the plastic bottle. The average mean distance error of stereo camera based navigation is 0.583 meters. The main reason for less success of the

outdoor tests was the inaccuracy of the GPS reading as the GPS had a ± 5 meters error in the readings. To avoid this a more accurate GPS can be used.

Table 9: Outdoor tests' computer vision navigation analysis.

Outdoor test number	Actual distance from bottle (meters)	Estimated distance (meters)	Distance error (meters)
B1	1.49	1.25	0.24
B2	1.45	1.51	0.06
B3	1.48	1.64	0.16
B4	-	-	-
B5	1.48	1.44	0.04
B6	-	-	-

Figure 37 also shows the results of four other outdoor tests detecting and computing depth of the plastic bottles in different orientations. Also, Figure 49 and Figure 50 show the remaining outdoor tests which can be found in the Appendix.

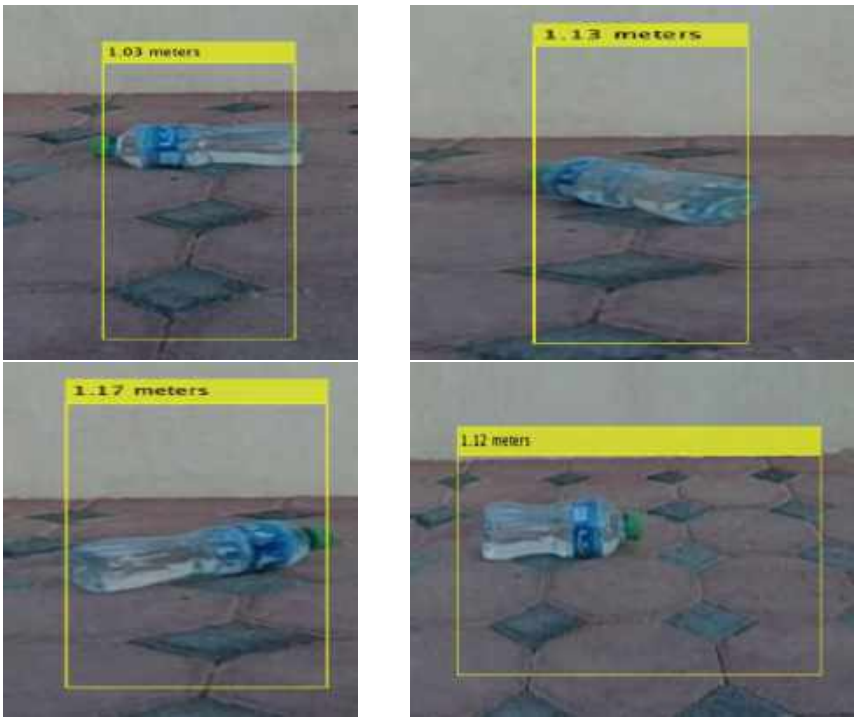

Figure 37: Detection of the plastic bottle's depth in different orientation.

Chapter 5: Conclusion

To conclude, this work has presented an autonomous robotic system for automated identification and collection of plastic bottles. Also, results of several indoor and outdoor tests have been presented, where the robot was able to successfully identify and locate recyclable plastic bottles, and then pick it up. Here, the GPS location of a point where recyclable waste is expected to be found is assumed to be known beforehand. Then the Kobuki robot uses GPS based navigation, followed by the computer vision-based navigation to identify, locate and reach a plastic bottle among the recyclable waste. The computer vision module also detects plastic bottles in different orientations. Once the plastic bottle is identified, the arm kinematics module makes the arm move to the location of the plastic bottle and picks it up. The importance of this work is that it can reach and pick plastic bottles in areas where humans may not easily be able to go, or operate in conditions unfavorable for humans e.g. extreme heat as is common in the UAE. In future, a further developed prototype can be used to help clean plastics from the natural habitat of animals to ensure they do not swallow the plastic pieces and get severely infected. This prototype may help achieve upon development, a final goal towards a solution that can reduce filling landfills with otherwise recyclable plastic waste material, that may go unnoticed or not collected for recycling by humans. Hence, keeping the providing hopefully for a sustainable and clean environment for better life of humans and animals. In future work, various types of plastic bottles can be detected by training a CNN with different types of plastic bottles having various shapes, sizes, and colours. Furthermore, these bottles can, be separated by type into bins, and then later be transported to a recycling station automatically. This may help increase the efficiency of the overall recycling process from collection to generation of recycled material.

References

[1] L. Adane and D. Muleta, "Survey on the usage of plastic bags, their disposal and adverse impacts on environment: A case study in Jimma City, Southwestern Ethiopia," *Journal of Toxicology and Environmental Health Sciences*, vol. 3, no. 8, pp. 234–248, 2011.

[2] T. Opeolu Olukunle, "Plastic waste awareness and practices among young environmentalists in the faculty of environmental sciences." Master's thesis, University of Lagos, Akoka, Yaba, Lagos state, 5 2019.

[3] S. Karbalaei, P. Hanachi, T. R. Walker, and M. Cole, "Occurrence, sources, human health impacts and mitigation of microplastic pollution," *Environmental Science and Pollution Research*, vol. 25, no. 36, pp. 36 046–36 063, 2018.

[4] J. Wang, W. Guo, T. Pan, H. Yu, L. Duan, and W. Yang, "Bottle detection in the wild using low-altitude unmanned aerial vehicles," in *2018 21st International Conference on Information Fusion (FUSION)*. Cambridge, UK: IEEE, July 2018, pp. 439–444.

[5] Y. Tachwali, Y. Al-Assaf, and A. Al-Ali, "Automatic multistage classification system for plastic bottles recycling," *Resources, Conservation and Recycling*, vol. 52, no. 2, pp. 266–285, 2007.

[6] S. Shahbudin, A. Hussain, D. A. Wahab, M. Marzuki, and S. Ramli, "Support vector machines for automated classification of plastic bottles," in *6th International Colloquium on Signal Processing and Its Applications (CSPA)*, Mallaca City, Malaysia, 2010, pp. 1–5.

[7] K. Ozkan, S. Ergin, S. Isik, and I. Isikli, "A new classification scheme of plastic wastes based upon recycling labels," *Waste Management*, vol. 35, pp. 29–35, Jan. 2015.

[8] V. Singh and R. Mishra, "Developing a machine vision system for spangle classification using image processing and artificial neural network," *Surface and Coatings Technology*, vol. 201, no. 6, pp. 2813–2817, 2006.

[9] M. Fulton, J. Hong, M. J. Islam, and J. Sattar, "Robotic detection of marine litter using deep visual detection models," in *2019 International Conference on Robotics and Automation (ICRA)*, Montreal, QC, Canada, May 2019, pp. 5752–5758.

[10] "Deep-sea Debris Database," library Catalog: www.godac.jamstec.go.jp. [Online]. Available: http://www.godac.jamstec.go.jp/catalog/dsdebris/e/ [Accessed: 2020-03-06].

[11] A. Akib, F. Tasnim, D. Biswas, M. B. Hashem, K. Rahman, A. Bhattacharjee, and S. A. Fattah, "Unmanned floating waste collecting robot," in *TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON)*, 2019, pp. 2645–2650.

[12] M. Valdenegro, "Submerged marine debris detection with autonomous underwater vehicles," in *2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA)*, Kollam, India, Dec. 2016, pp. 1–7.

[13] S. Kulkarni and S. Junghare, "Robot based indoor autonomous trash detection algorithm using ultrasonic sensors," in *2013 International Conference on Control, Automation, Robotics and Embedded Systems (CARE)*, Jabalpur, India, Dec. 2013, pp. 1–5.

[14] P. Ravindhiran, P. Gopal, S. J. Gladwin, and R. Rajavel, "Automated indoor waste management system employing wavefront algorithm and received signal strength indicator values-based mobile robot," in *2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, Dhaka, Bangladesh, 2017, pp. 284–289.

[15] L. Cui, T. Zhang, X. Yin, and Y. Qi, "Road garbage cleaning device based on zigbee gateway and image recognition," in *2019 Chinese Control And Decision Conference (CCDC)*, Nanchang, China, 2019, pp. 5603–5607.

[16] R. Pascoe, "Development of a method for separation of pvc and pet using flame treatment and flotation," *Minerals Engineering - MINER ENG*, vol. 16, pp. 1205–1212, 11 2003.

[17] C.-H. Park, H.-S. Jeon, H.-S. Yu, O.-H. Han, and J.-K. Park, "Application of Electrostatic Separation to the Recycling of Plastic Wastes: Separation of PVC, PET, and ABS," *Environmental Science & Technology*, vol. 42, no. 1, pp. 249–255, Jan. 2008, publisher: American Chemical Society.

[18] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. C. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," *ICRA Workshop on Open Source Software*, 1 2009.

[19] "ROS/Concepts - ROS Wiki." [Online]. Available: http://wiki.ros.org/ROS/Concepts [Accessed: 2020-03-06].

[20] "kobuki/Tutorials/Kobuki's Control System - ROS Wiki." [Online]. Available: http://wiki.ros.org/kobuki/Tutorials/Kobuki's%20Control%20System [Accessed: 2020-04-29].

[21] "kobuki_node - ROS Wiki." [Online]. Available: http://wiki.ros.org/kobuki_node?distro=kinetic [Accessed: 2020-03-06].

[22] B. Zhao, B. Huang, and Y. Zhong, "Transfer Learning With Fully Pretrained Deep Convolution Networks for Land-Use Classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 9, pp. 1436–1440, Sep. 2017.

[23] P. Bharati and A. Pramanik, "Deep Learning TechniquesR-CNN to Mask R-CNN: A Survey," in *Computational Intelligence in Pattern Recognition*, ser. Advances in Intelligent Systems and Computing, A. K. Das, J. Nayak, B. Naik, S. K. Pati, and D. Pelusi, Eds. Singapore: Springer, 2020, pp. 657–668.

[24] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[26] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," in *Computer Vision ECCV 2014*. Cham: Springer International Publishing, 2014, pp. 818–833.

[27] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, 2015, pp. 1–9.

[28] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv:1409.1556 [cs]*, Apr. 2015. [Online]. Available: http://arxiv.org/abs/1409.1556 [Accessed: 2020-03-30].

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016, pp. 770–778.

[30] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, 2014, pp. 580–587.

[31] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[32] "Depth estimation from stereo video." [Online]. Available: https://www.mathworks.com/help/vision/examples/depth-estimation-from-stereo-video.html

[33] P. Corke, *Robotics, Vision and Control: Fundamental Algorithms In MATLAB Second, Completely Revised, Extended And Updated Edition*, 2nd ed., ser. Springer Tracts in Advanced Robotics. Springer International Publishing, 2017.

[34] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer Science & Business Media, Nov. 2008.

[35] "CIFAR-10 and CIFAR-100 datasets." [Online]. Available: https://www.cs.toronto.edu/~kriz/cifar.html [Accessed: 2020-03-27].

[36] "Camera calibrator." [Online]. Available: https://www.mathworks.com/help/vision/ug/stereo-camera-calibrator-app.html [Accessed: 2020-04-20].

[37] “Robotics toolbox for matlab.” [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/68542-robotics-toolbox-for-matlab [Accessed: 2020-04-29].

Appendix: Detailed Experimental Results

Navigation Module Testing

The preliminary tests of navigation module are shown in Figures 38 to 44. The blue line in the figures is the actual navigation path on the google map, whereas the red line is expected navigation path on the google map.

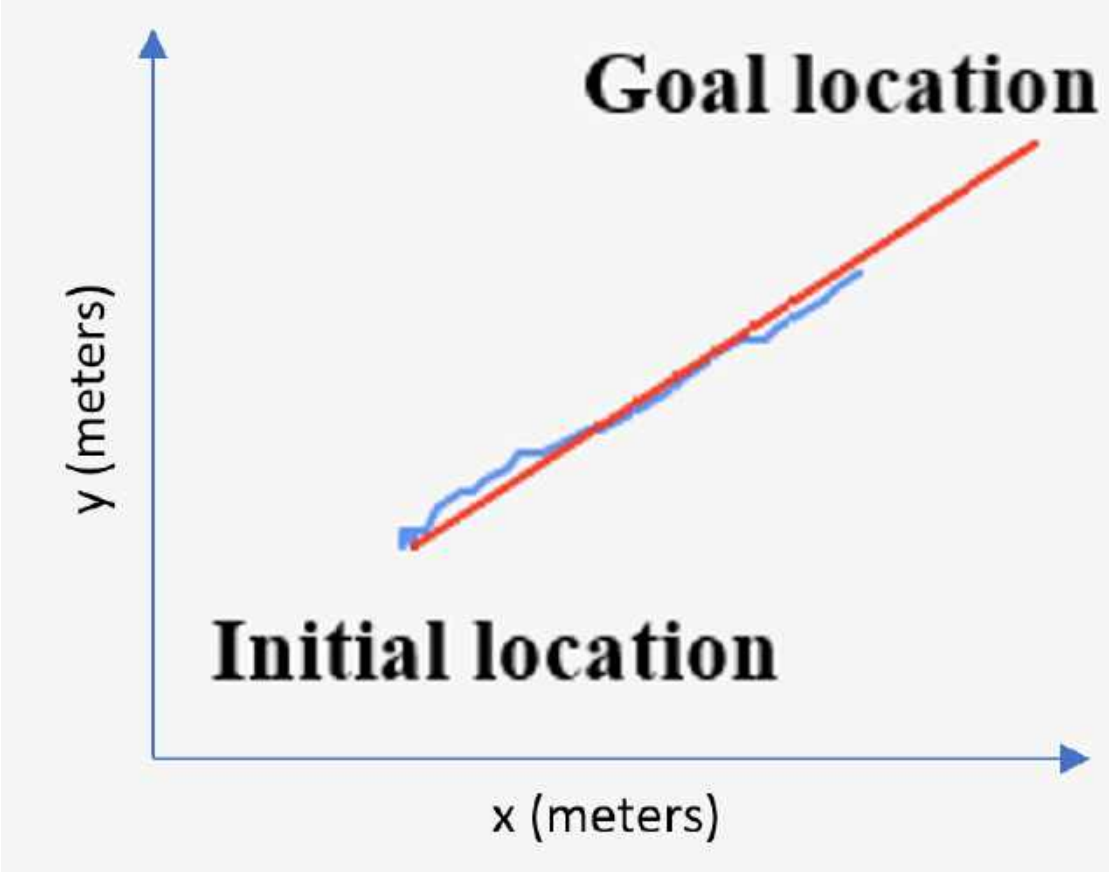

Figure 38: Preliminary navigation module Test A2.

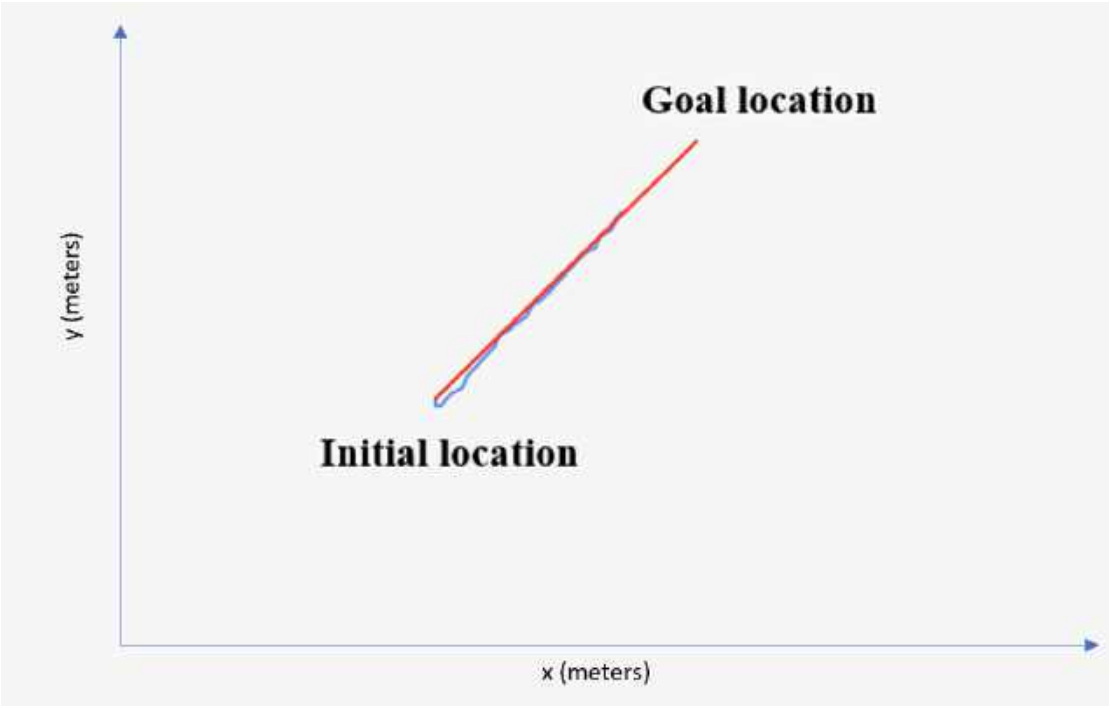

Figure 39: Preliminary navigation module Test A3.

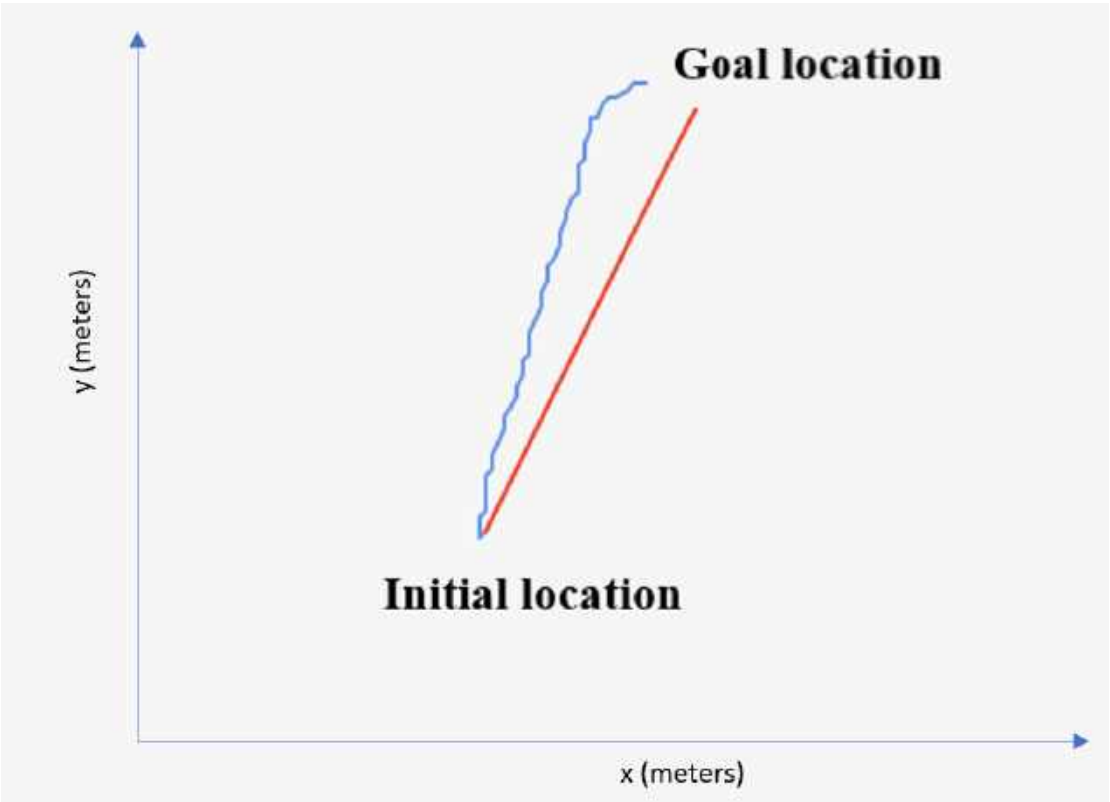

Figure 40: Preliminary navigation module Test A4.

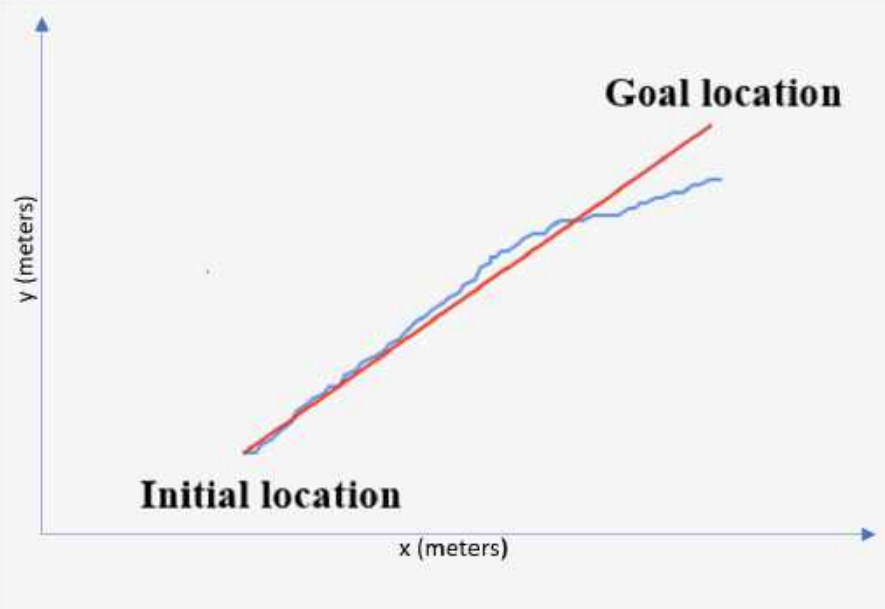

Figure 41: Preliminary navigation module Test A6.

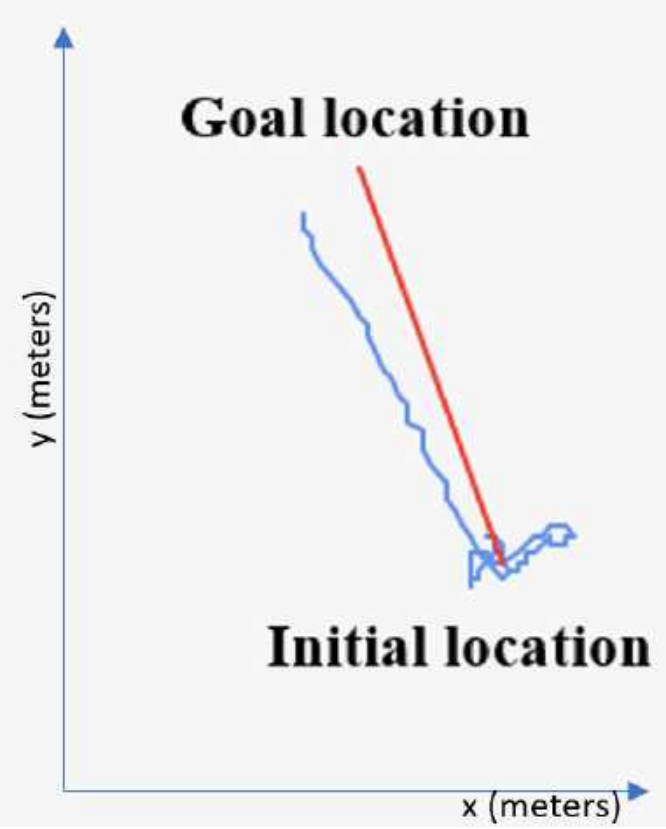

Figure 42: Preliminary navigation module Test A7.

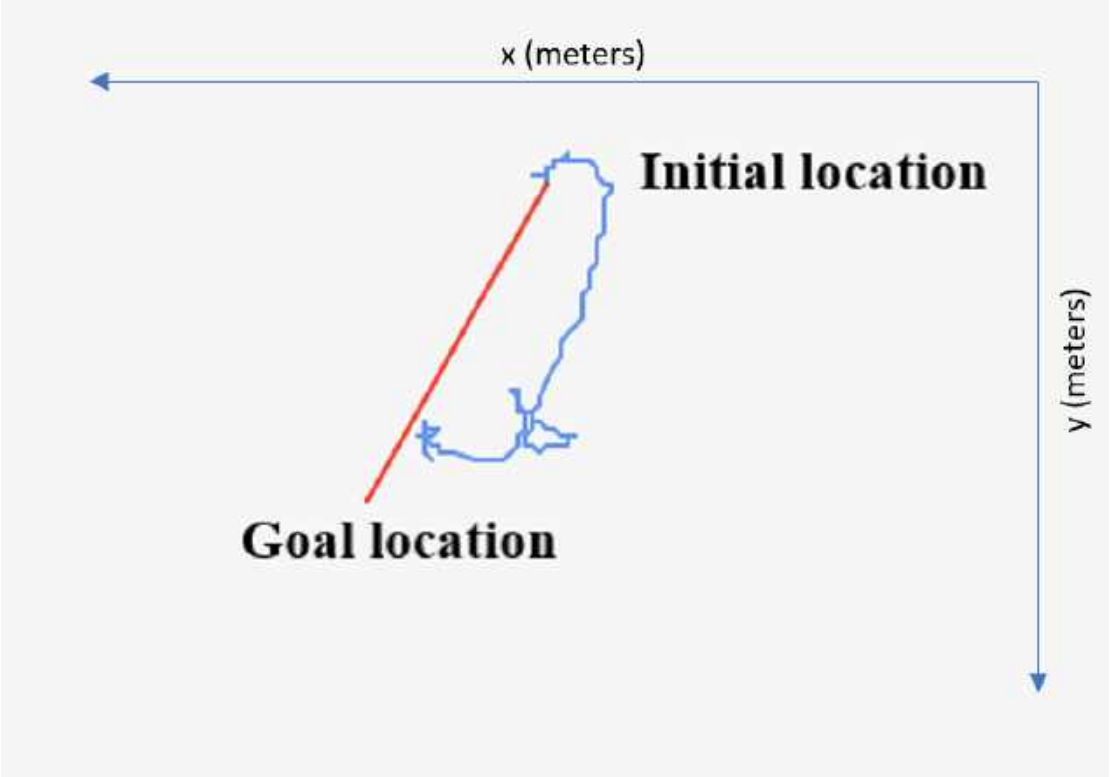

Figure 43: Preliminary navigation module Test A8.

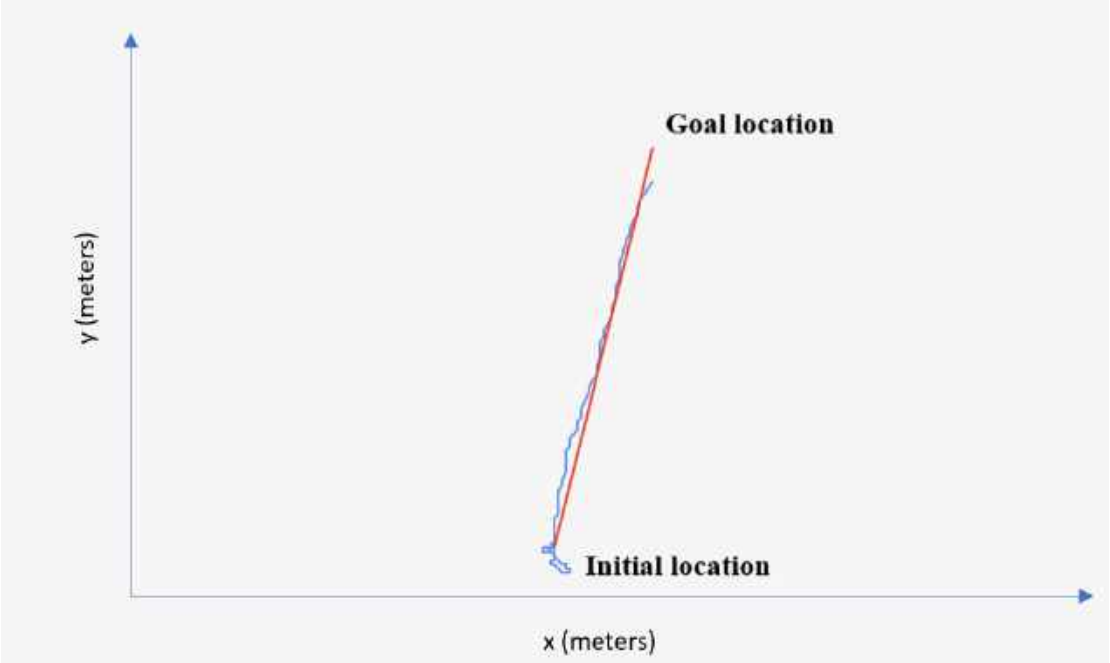

Figure 44: Preliminary navigation module Test A9.

Indoor Testing

The indoor tests are shown in Figures 45 to 48.

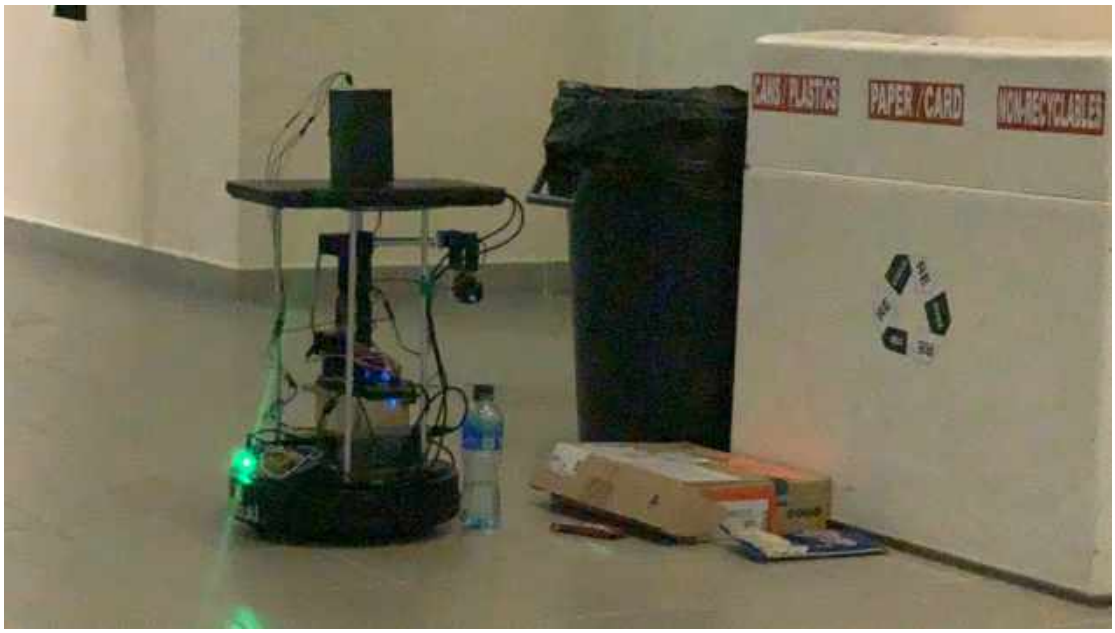

Figure 45: The developed robot before picking up the bottle in indoor Test 2.

Figure 46: The developed robot picked up the bottle in indoor Test 2.

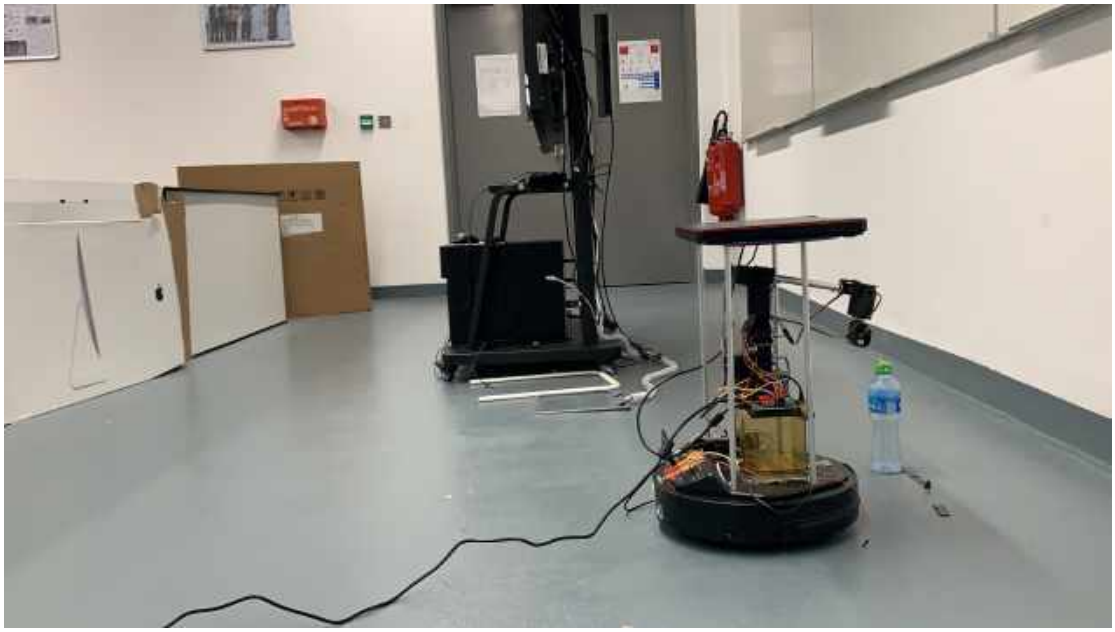

Figure 47: The developed robot before picking up the bottle in indoor Test 4.

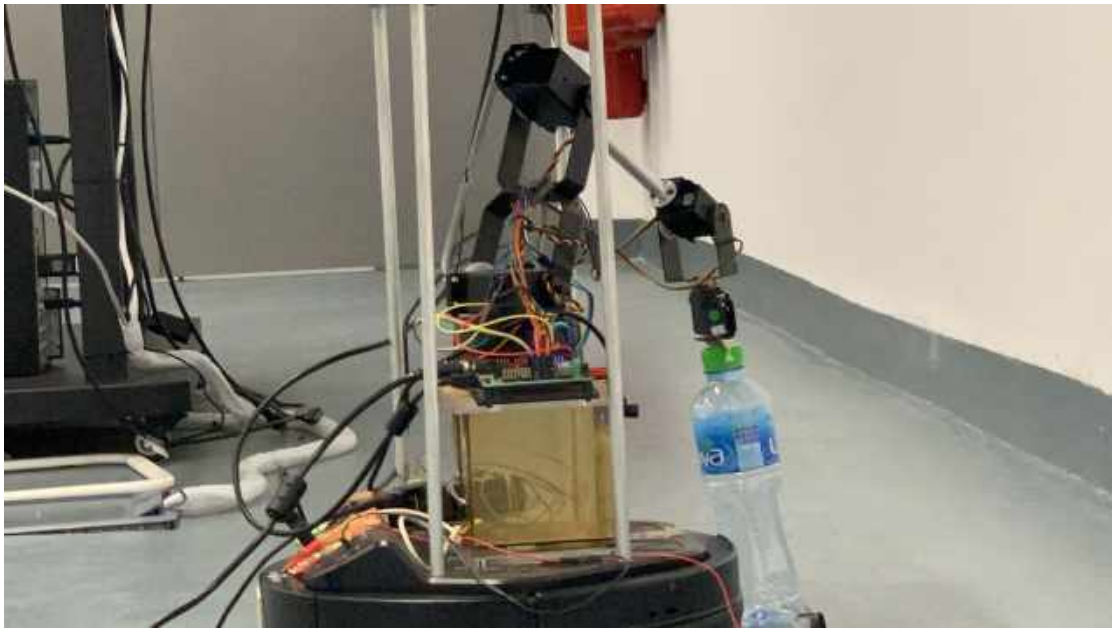

Figure 48: The developed robot after picking up the bottle in indoor Test 4.

Outdoor Testing

Figure 49 and Figure 50 are the outdoor tests.

Figure 49: The experimental setup before starting the outdoor Test B3.

Figure 50: The developed robot after picking up the bottle in outdoor Test B3.

Vita

Uzma Ahmed Din was born in Ajman, United Arab Emirates. She received a Merit Scholarship to the American University of Sharjah in Sharjah, United Arab Emirates, from which she graduated magna cum laude in 2017. Her degree was Bachelor of Science in Electrical Engineering with minor in Computer Engineering. In 2018, Ms. Uzma began a Masters program in Electrical Engineering at the American University of Sharjah. Ms. Uzma is also a member of the Institute of Electrical and Electronics Engineers.