2019

An Assistive Device That Aids The Visiually Imparied People To The Crosswalk

Jiayue Wang

AN ASSISTIVE DEVICE THAT AIDS THE VISIUALLY IMPARIED PEOPLE
TO THE CROSSWALK

by

JIAYUE WANG

ABIDIN YILDIRIM, CHAIR
FRANKLIN R AMTHER
LEON JOLOLIAN
ARIE NAKHMANI

A THESIS

Submitted to the graduate faculty of The University of Alabama at Birmingham,
in partial fulfillment of the requirements for the degree of
Master of Science

BIRMINGHAM, ALABAMA

2019

AN ASSISTIVE DEVICE THAT AIDS THE VISIUALLY IMPARIED PEOPLE TO THE CROSSWALK

JIAYUE WANG

ELECTRICAL AND COMPUTER ENGINEERING

ABSTRACT

The visually impaired people mostly rely on a blind cane or specially trained the dogs (Guide dogs) to help them walk or cross the road safely. However, the white canes are for only avoiding obstacles on the roadway and training the guide dogs is a long-term process. Deficiency of reliable assistive tools, traffic accidents seem to be inevitable, especially to visually impaired or blind people. In order to address this problem and decrease the accidents of this nature, technologically more advanced and intelligent tools are necessary. This proposed thesis work will help visually impaired individuals, without affecting their daily activities and habits, improve their quality of life and their safety on the street and crosswalk areas. Since this solution will not require any changes in existing infrastructure, so it will not cost extra for the local governments or cities. This Thesis work aims to design, develop, and fabricate an assisting device, which will be capable of identifying and accurately interpret pedestrian traffic light to assist the visually impaired individuals by crossing the streets on crosswalk areas safely. Using a low cost, and off-the-shelve available camera module and an embedded computer, it will capture the surrounding image and process it to identify the traffic lights of interest. Then, according to the required direction of the crosswalk, the device will notify the person through the modulated vibration when the crosswalk is safe to cross. The vibration is produced by a mini DC vibration motor on the device. Besides, the wearability of the device, it will also be cost-effective solution and convenient in use.

Keywords: visually impaired, convolutional neural network, real-time classification, OpenCV

DEDICATION

I dedicated this master thesis to my advisors and my parents, who provide a great deal of guidance and support throughout my life and always had faith in my success.

ACKNOWLEDGEMENTS

I would like to thank all of the people that contributed in the process of completing my master's thesis.

First of all, I want to thank my advisor Dr. Abidin Yildirim. He provided the full support from the start of the thesis until finished completely. He is generous and patient to all the students. He gives the valued guidance.

Secondly, I want to thank my parents, they give me the chance to finish the master’s degree in the University of Alabama at Birmingham. I am always feeling gratitude.

Thirdly, I want to thank my committee member, Dr. Arie Nakhmani, Dr. Leon Jololian and Dr. Franklin Amther. They took their time to read my thesis and gave precious advices as well as help me to evaluate my thesis work.

Moreover, I would also like to thank my colleague and friends, especially my colleague Sahaj Patel, he shared good learning resources and gave worthy suggestions. I also want to thank to my friends who has encouraged me all the time.

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CNN	Convolutional Neural Network
DL	Deep learning
OS	Operating System
PTL	Pedestrian Traffic Lights
ReLU Layer	Rectified Linear Unit Layer

TABLE OF CONTENTS

CHAPTER 1

PROJECT INTRODUCTION AND BACKGROUND RESEARC

1.1 Introduction

In this thesis work, a framework for a wearable assistive device is developed to help visually impaired or blind people to recognize the condition of the Pedestrian Traffic Lights (PTL) while crossing the street. One hand, Crossing the street is inevitable in daily life for all the people and following the rule of PTL is necessary, because it help maintain the order between the vehicle and pedestrians. On the other hand, visually impaired people have difficulties by navigating within the daily traffic conditions. Therefore, in this thesis work, a device is proposed that can transform the visual PTL signal to the vibration signal that informs the visually impaired person about the condition of the crosswalk. The Machine Learning Algorithm is used, to create a stable image identifier (i.e. classifier) to recognize the traffic lights conditions. The Raspberry Pi (introduce in Chapter 2) is used to run the Machine Learning Algorithm and also to process the conditional output cases to control actuator. This thesis work consists of five chapters, which are briefly explained fallowing sections.

Chapter 1 is a brief introduction of the background, as well as the motivation of the thesis. The literature research is also covered in this chapter, including the current solutions. According to the literature survey, an analysis of the differentiations and shortcomings will also be given.

Chapter 2 discusses the hardware and the software components of the device.

Chapter 3 explains the device design. This includes also outlines the initial test prototype, and how to create the classifier model.

Chapter 4 discusses the training and test of the classifier for PTL, the chapter also includes the result of each method, training and testing accordingly.

Chapter 5 summarizes the thesis work, and discusses difficulties, as well as the future work.

1.2 Motivation and Problem Statement

The eye, which is one of the most important organs to humans, but with the change lifestyles of the society, for instance, using intensely computers, smartphones, watching TVs, etc., can lead eye diseases, which can result to visually impaired or even blind. ‘Braille’ [1] is a writing system that created to help visually impaired people to read. In ‘Braille’ (see Figure 1), it presents letters and number by the convex dot forming up different pattern

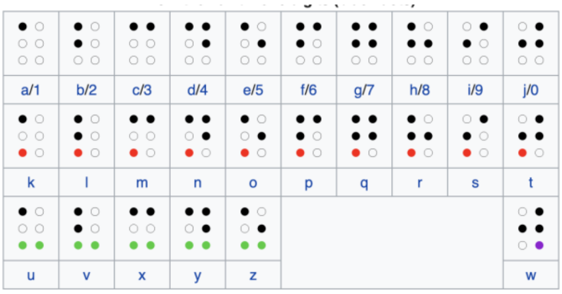


Figure 1. Braille

This writing system provide the ‘refreshable braille display’ to read the words for people have low vision condition. Today, more researchers trying to cure the diseases with advanced technology. Also new technological developments in engineering and materials science, allows to create convenient environment such as intelligent robot assistant and autonomous furniture to help visually impaired people in an indoor environment.

More and more visually impaired people having chances to join general public and having a job offer or decide to study for a better opportunity. As for outdoor activities, traffic accidents are unescapable, especially for visually impaired people. One of the motivational strengths of this thesis work was to help visually impaired people to avoid potential accidents while they are crossing the street intersections.

Visually impaired and blind people rely on the “white cane” (see Figure 2) [2] or “guide dog” [3] to help them walk or cross the road safely. However, they are not convenient and make the person dependent on them. Although, in some intersections all over the country, the sidewalk traffic lights have the audio warning, however, it is not available every street corner in a big city because of the limitation of the government funds. In this thesis work, an embedded computer-based device is proposed, which can help visually impaired people in convenient way. The device will help the visually impaired person by signaling the PTL conditions while he/she is waiting for the traffic lights' signal to cross the street.

Figure 2. White Cane

In this thesis work, the device will be attached to the ‘vest’, which is easy to wear and take off. Moreover, the device is stable because it will be near to the chest of the body. A series of intelligent devices have been invented to help visually impaired or blind people. Although some of them are producing excellent results, the affordability seems like a problem for most of the elderly users. In this thesis work, the system components are selected carefully to deliver an affordable product.

1.3 Background

1.3.1 Visually Impaired People

According to the World Health Organization, it is estimated that approximately 1.3 billion people are living on earth with some form of vision impairment. From 1.3 billion people, 188.5 million people have a mild vision impairment, 217 million have moderate to severe vision impairment, and 36 million people are blind. Also, note that more than 200 million people are aged 50 or above [4] in this category of visual impairment. Visual impairment influenced the people deeply not only in daily life but also on mental health. Caretaking of visually impaired people by having an assistant is both time and energy consuming. Another severe problem is that the visually impaired people can get hurt or even can die because of the traffic accident. The visually impaired people mostly rely on a white cane or specially trained assistive dog (i.e. guide dog) to help them walk or cross the street intersections safely. While the canes are for only avoiding the obstacles on the roadway, the guide dogs which are partially color blind and they cannot classify the color from the PTL.

1.3.2 Pedestrian Traffic Lights

Pedestrian traffic lights have been used for same purpose worldwide, however, they have different shapes and colors according to the local requirement [5]. Most of PTL has two colors: red and green. Red is always meaning ‘stop’ and green one means ‘keep going’, in the USA, the white color lights is always used to instead of the green lights. Besides, the pattern on the PTL is various but similar among different counties. The ‘walking man’ logo is always meaning keep going but the ‘hand’ on the PTL means stop. In some countries the ‘standing man’ logo also stand for stop. For example, Figure 3 shows the traditional United States' style PTL (stop). At the same time, Figure 4 presents the simplified structure of Chinese PTL. In this thesis work, will mainly cover the USA style of PTL. The differences among these pedestrian traffic lights require different datasets to create the training model, which is described in Chapter 3.

Figure 3. USA style PTL

Figure 4. Chinese Style PTL

1.3.3 Background Research

Walking without the partners who can help them is difficult for visually impaired people to walk an indoor or outdoor environment. Thus, initially, the guide blind dogs are trained to help the visually impaired people to check the obstacle and the taking the way, which would be the first solution for the visually impaired people in Figure 5. However, this kind of solution is not as reliable as when an exceptional situation or emergencies happens. Moreover, in the term of helping the visually impaired people across the street, the guide dogs are red–green color blind and incapable of interpreting street signs. The guide dogs are trained in the professional school, they will learn showing the directions, sitting with owner and finding certain objects (i.e. Stairs, Chairs, or Doors) for the owner. However, adapting a guide dog requires spending time on cooperate with each other. This will lead to difficulties in the case there is no reference.

Figure 5. Guide Dog[1]

The PTL with audio are compiled to help people (see Figure 6) when the different colors were shown the different frequency of buzzer would appear to help the visually impaired people. This is a more reliable design for people to cross the street. However, it is not possible for the government to set this kind of traffic light cover every intersection on the street as it will increase the budget from the government because of the amount of the pedestrian traffic light.

Figure 6. Pedestrian Traffic Light with Sound

[1] https://www.google.com/

Besides the two solutions (guide dog and audio PTL) analysis above, more research is related to use electronics and artificial intelligence to create the assistive device to help visually impaired people occurs. According to Elmannai et al., the authors divide of these devices in three categories as such: Vision enhancement, vision substitution, and vision replacement accordingly [6]. Visual enhancement requires the medical area as it needs to display the visual information directly through the visual cortex of the brain or an ocular nerve [7], thus it becomes the most complex technology among these three categories. Visual substitution is basically replacing the visual information directly to the audio or sense of touch. Vision replacement and substitution are similar; however, the difference between them is that the visual replacement aim to heighten the visual sense based on the functional glasses. This branch is helpful to the mild visually impaired people but not blind people. Figure 7 shows the basic category of the electronic devices for visually impaired people.

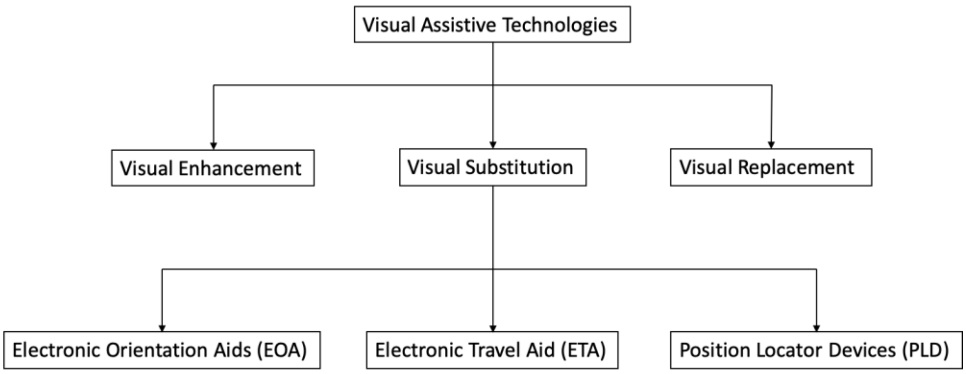


Figure 7. Visual Assistive Technologies and Subcategories

Based on the explanation above, vision substitution is also divided into many subcategories, which are designed more specified applications. Electronic Orientation Aid (EOA), mainly work on the path detecting or produce the path. Second one is

Electronic Travel Aid (ETA), these devices working on obstacle detecting and distance calculation. The third one is Position Locator Devices (PLD), among these devices, working mainly rely on the GPS (Global Position System), which are used to make the estimation of users' position. Besides, these three categories have a more detailed branches based on time analysis, in-door/out-door use, distance range, and various object type in the environment. For example, some devices can be used outdoor some can be only used indoor environments, and users may need different assistive as they use in daylight or night.

Smart Cane is a portal device embedded with a sensor system, it consists of two parts; one is the detection part, and the other one is a voice feedback part [8]. They use ultrasound sensors, servo motors, and a fuzzy controller to detect the object while using it. Besides, a buzzer and a vibration output are used to guide the user. Figure 8 gives a demonstration of the smart cane.

Figure 8. Prototype of the Smart Cane

Figure 9 shows another design called "A Path Force Feedback Belt" [9]. The whole device is designed in the shape of a belt that can be worn by the user. Two cameras are on the both sides of the belt are used to record stationary and nonstationary objects in the environment. Each circle on the belt is the vibration actuator that is used to provide a feedback to the user. After the creating the 3D model based on the video captured by the camera module, the device can detect the walls and other obstacles. Then, users can navigate according to the vibrators signals that are controlled by 3D model result of image processing.

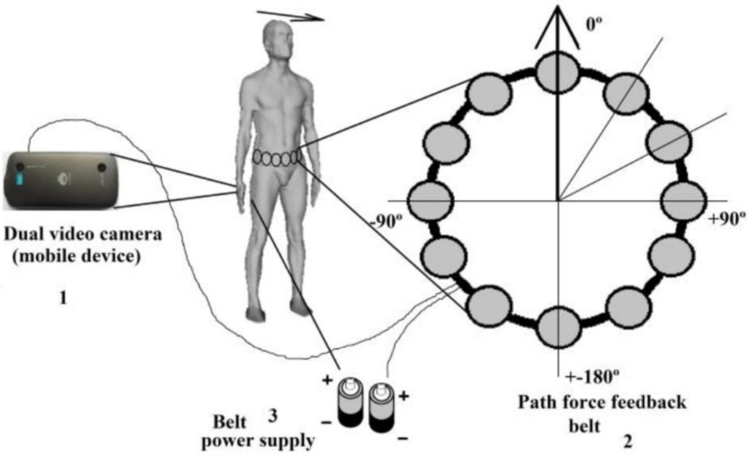


Figure 9. Design of the A Path Force Feedback Belt

Figure 10 shows one design from Adrien Brilhault, et al., which is a smart assistive device for visually impaired people. It depends on the adapted Geographical Information System (GIS) and the fusion of the Global Positioning System (GPS) as well as vision-based positioning [10]. The head-mounted device is consisting of two cameras as the input to the device to analyze the video and send the information to the portable computer, which is called the "head tracking system." A magnetic compass and gyroscopes are also included in the system. When the user started to walk on the street, a certain algorithm (SpikeNet) is used to recognize the targets. Then the algorithms return the coordinates of the targets. The GIS has the database which contains four classes: transportation, buildings, land use, other objects. The input information will be processed and distributed to the different classes of a database in order to have a better localization. Thus, with the adapted GIS and on the fusion of GPS and the camera input, encourage the user has better mobility through the audio feedback from the device.

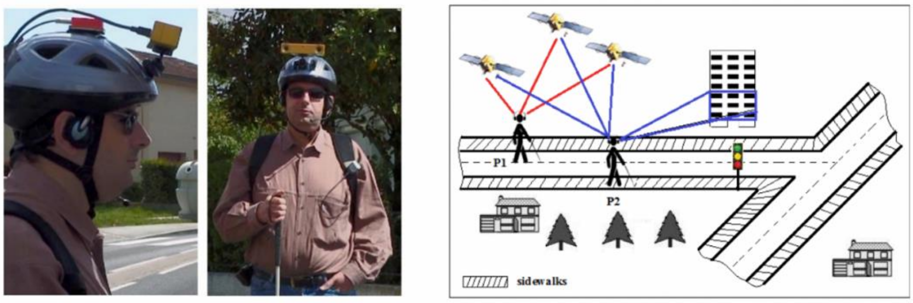


Figure 10. The Fusion of Artificial Vision and GPS to Improve Blind Pedestrian Positioning

These solutions are represented in the category of visual substitution. Some of are the sensor-based the assistive device and some are the GPS based, by the comparing the audio PTL and the blind guide dog, these devices are service for the single user which is more reliable and efficient. According to these current solutions, the thesis work will have an assistive device that is based on the deep learning algorithm to recognize the PTL, which is feasible with low budget also robust system for visually impaired people.

CHAPTER 2

HARDWARE AND SOFTWARE REQUIREMENTS

2.1 Introduction

Chapter two contains the introduction and the explanation of the hardware and software components. Based on that, five hardware components are given. The Raspberry Pi is the Center Process Unit (CPU), which is working as the master device to receive the input images and make the classification based on the input and give the output signal. The camera is the only channel for getting the image for the ambient. The Bluetooth module HC-05 completes the wireless communication between the Raspberry Pi and the Arduino Nano board. Moreover, the Arduino Nano is the controller to give the commands to the vibration motor. The mini vibration motor is used to give the output signal to the users.

Besides, the software components are also included in this chapter. Python will be the programming environment for creating the classifier module. Deep Learning is also introduced to create the classifier based on the Convolutional Neural Network (CNN). A dataset is created to training the model. Some of the libraries in the Python is imported to preprocessing the image and create the architecture of the CNN, etc.

2.2 Hardware

2.2.1 Raspberry Pi as Center Processing Unit

The Raspberry Pi is a microcomputer that can be carried mobile and it consumes less power. The Raspberry Pi 3 model B version is used in this thesis work, which is shown in Figure 11. It consists of Quad Core 1.2GHz Broadcom BCM2837 64bit CPU and 1GB RAM [11]. Additionally, it also consists of Bluetooth Low Energy (BLE) module. A Micro SD port reads the SD card with the Raspbian operating system on it.

Raspberry Pi also consist of 40 pins extended GPIO for external interfacing sensor with different protocols such as SPI and I2C, including Inputs and Outputs pins. The assembling of the Raspberry Pi is simple, a display connects with Raspberry Pi board through the HDMI port. The mouse, keyboard, and a mobile WIFI module will be connected with the board through the USB port. These components are used in setting up the operating system in Raspberry Pi, running the Python code, and observe the output result from the classifier. In practical use, the USB camera is the only device that is connected with Raspberry Pi to obtain the real-time image from the environment. As a note, the WIFI will be used only during the training phase of the classifier.

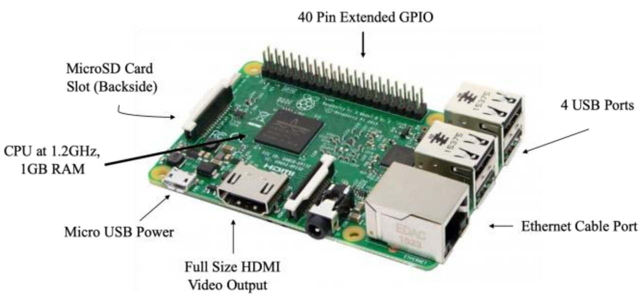


Figure 11. Raspberry Pi 3 Model B

The Raspbian Operating System (OS) is based on the Linux OS. It can work smoothly on the Raspberry Pi. The version of the Raspbian in this thesis is Raspbian Buster, which is released in September 2019. The Kernel version is 4.19 and the size is 2541MB. In the Raspbian system, the build-in Bluetooth module can enable directly through the terminal or through the system setting. This brings the convenience to later use for connecting to the HC-05 on the Arduino Nano board. The new system '*Bust*er' is safer compared with the former version of the Raspbian system. Besides, the system also contains the new programming environment for Python, '*Thonny*.'

2.2.2 Camera Module

The camera (see Figure 12) in this thesis work has the size with 15cm ´ 7.6cm ´ 20cm. It is suitable to put in the pocket on the vest. It has a high resolution of up to 5 megapixels and 720 HD on video recording, as well as the automatic light correction [12]. The input image size of the camera is 1280x720 pixels. The weight of the camera is 200g, which is satisfied for the user to carry it during the working time. The camera has a USB connector, can plug into the Raspberry Pi through the build-in port. The price of the camera varies between 20 $ and 25 $ for each, which helps to make the device affordable.

Figure 12. Camera

2.2.3 Arduino Nano

The Arduino Nano (see Figure 13) is a small and breadboard-friendly board based on the ATmega328P (Arduino Nano 3.x) micro-controller with a total board size of 1.85 cm x 4.31 cm [13]. It has the same working principle as the Arduino Uno but with different packages embedded in it. Nano contains 14 Digital I/O Pins, of which 6 are PWM Output pins and 8 Analog input pins. Besides, the operating voltage of Nano is 5V DC. In thesis work, the Nano is connected to an alerting system which is placed on individuals' hand for PTL indication. The Nano is connected with other several components such as Bluetooth module, Mini vibration motor, and battery power.

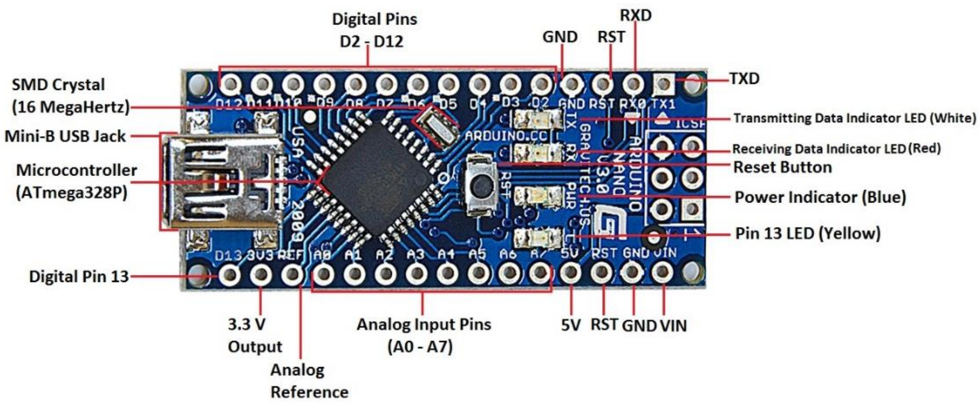


Figure 13. Arduino Nano Board

2.2.4 Bluetooth Module

HC-05 is the Bluetooth Module (Fig17) is used in this thesis work. It has a CSR BC417 mainstream Bluetooth chip with an operating voltage of 3.6V DC to 6V DC, respectively [14]. This module can work both; as a slave or a master device. Before using the HC-05 Bluetooth module, it is necessary to set the HC-05 as the slave device by using the AT command. The AT command in the HC-05 Bluetooth module is used to check and change the status (version, name, etc.) of the HC-05. The modification of the default mode can be changed in the AT command mode.

To enter the AT command mode, one needs to hold the reset bottom and turn on the power supply. To check if the Bluetooth module is serving as a slave device, it needs to send the ‘AT+ROLE?’ command to the serial port. If the result is ‘0’, it means that the device is in the slave mode. Otherwise, it is in master mode. In this thesis work, the Bluetooth module is operated as a slave. The complete AT commands and their detailed explanation can be found in the HC-05 user manual.

Figure 14 shows the wiring connections of the HC-05 module. The HC-05 module has a connector with five pins and a reset button. The TXD pin of the Bluetooth module is connected to the RXD pin on the Arduino board. The RXD pin of the Bluetooth module is connected with TXD on the Arduino Board. The enable pin is used to enter the AT command mode by utilizing the reset button. The enable pin is activated by connecting to the power rail (i.e., 5V) momentarily when entering to the AT command mode, otherwise, it will leave open while using normal operation. The Build-in LED is used to indicate the working mode of this Bluetooth module. If it is blinking with high frequency, it means waiting for pairing with another Bluetooth module. The led blinks with the three seconds interval mean that it is in the AT command mode. If the led blinks once per second means that it is paired successfully.

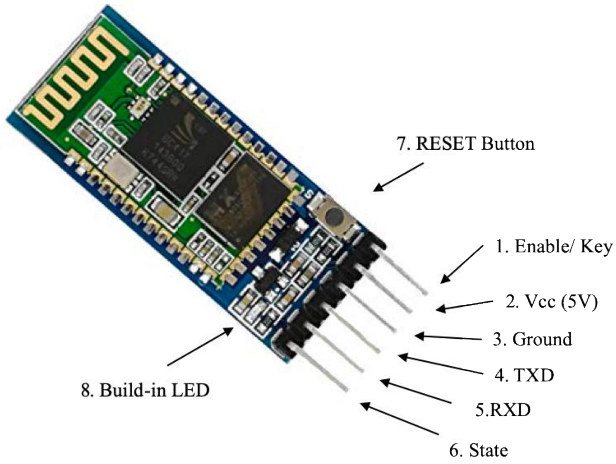


Figure 14. HC-05 Bluetooth Module

2.2.5 Mini Vibration Motor

The mini vibration motor is used as an actuator in the wristband. It can vibrate in different frequencies based on the setting in Arduino Nano. It has a diameter of 6mm and a length of 17mm [15]. It requires the voltage of 3V DC and speeds up to 1100 RPM. The weight of the vibration motor is 3g, which is acceptable to be a component on the wristband for a visually impaired person. An example of the mini vibration motor that is used in this thesis work is shown in Figure 15.

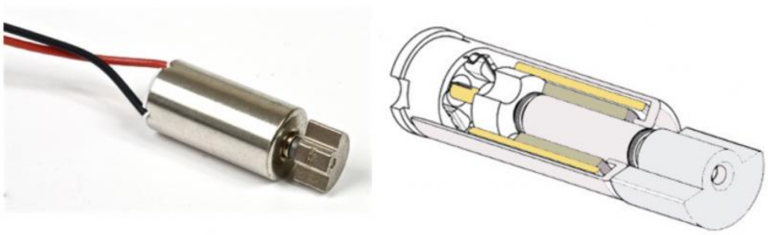


Figure 15. Vibration Motor

2.3 Software

2.3.1 Programming language

In this thesis, the Python programming language is used. The Python programming language is a high-level programming language and its open-source software. The Python language easy for beginners to learn. Currently, the majority task of Artificial Intelligence is executed with the help of the Python language.

2.3.2 Python Libraries

The first library used in Python is '*Numpy*' (Numeric of Python) [16]. In this thesis work, it is used to store the images as number values in the array. These values in the array are used to feed into the CNN. The '*matplotlib. pyplot*' library [17] is used to create a figure and create a plotting area, as well as the plot the lines. In this thesis work, the images are preprocessed in size, the *matplotlib. pyplot* is used to check the size of the images. Besides, the figures of comparing the loss and accuracy of the classifier are also plotted through this library. The '*glob*' library is used to read the path of the file. '*Scikit-learn*' is a library for data mining and data analysis [18], it is used to shuffle all the images in the dataset and split it into to subsets for training classifiers and testing classifier.

OpenCV (Open Source Computer Vision) is an advanced library for computer vision as well as machine learning [19]. It is developed based on C++. OpenCV is an excellent option to process images and video frames. It will be easy to compile applications, such as face recognition and object tracking. It contains about 2,500 algorithms for various image processing applications.

In this thesis work, OpenCV is used first to prepare the images in the dataset. All the images initially have different sizes and BGR (blue, green, and red) color mode. To read the images successfully, a uniform format of the images is essential. The default BGR mode needs to be converted into the RGB mode to process by OpenCV.

After the camera is activated by OpenCV to capture the real-time video, the OpenCV converts the video into frames to resize them and to change the color mode of the images, as mentioned before.

The '*Keras*' is another main library used in Python. Keras works as an essential tool in Python for creating the model [20]. This is a deep-learning framework for Python, and it provides the researcher a more convenient and more accessible way to create the deep-learning model. It contains built-in support for CNN for computer vision.

2.3.3 Dataset

The dataset is a crucial part of the deep learning model. The efficiency of the Deep learning model determines the training dataset. The cell phone video was recorded from the surrounding environment of the University of Alabama at Birmingham by walking on the street. These videos were converted into image frames with the help of '*Adapter*'. '*Adapter*' is a free software to convert the video to the image [21]. It supports the conversion in a batch and chooses the time interval based on the video.

After converting video to images, a part of the images in the dataset were rotated and flipped horizontally in order to increase the size of the dataset. This process is achieved by the built-in software in Mac, 'Automator'. In the Automator working environment, the 'Workflow' is created to processed multiple images in one time, choose the images that need the rotation and flip horizontally and save it as the new images. In this way, a series of new images are added into the dataset. This dataset is used to train the model on CNN for predicting target classes (category). The whole dataset is divided into two parts, which is train and test dataset with a ratio of 70% to 30% accordingly. The dataset in this thesis work contains three categories, which are ‘red PTL,’ ‘white PTL,’ and ‘background,’ respectively. The background category has no images with PTL. The background class is also included to distinguish the case if the visually impaired person is walking on the street without any PTL.

CHAPTER 3

DEVICE DESIGN

3.1 Overview

Chapter 3 contains the complete design of the device. The whole device is divided into two parts: The first part is the ‘vest’, which can be worn and taken off easily. The second part is the wristband with a vibration actuator. The communication between the main computer and wristband is also explained in this chapter. Also, a brief introduction is presented to explain how to create a classifier model based on the Deep Learning (DL), and the Convolutional Neural Network (CNN) architecture.

3.2 Hardware Design

This Master's Thesis work delivers initially proposed features. First, it is a wearable device that will be attached to the chest of the person with a specially designed vest with easy access to the required functions. A camera is exposed from the front pocket where the device safely placed. Raspberry pi work as the Center Processing Unit (CPU) to process the captured image and deliver the feedback over Bluetooth to the person's wristband.

The wristband, which is a second significant feature that provides convenience and comfort for the user. An operating actuator device receives the commands from the Raspberry Pi and executes wirelessly, which is integrated into a wristband. A Bluetooth module in the wristband act as a slave device and communicate with the

Raspberry Pi. It will receive the commands from Raspberry Pi and delivers the appropriate output signals. These output signals activate a vibration motor with a different frequency.

Finally, achieving an overall small mechanical size was one important goal. Using Raspberry Pi as the central processor, the design demonstrates not only powerful aid for visually impaired peoples but also a compact device that is usually a basic preference in terms of portability. Without a high battery pack, this design has a small mechanical size of 100 mm x 70 mm x 40 mm, which is a little bit larger than a standard Raspberry Pi Printed Circuit Board (PCB).

Initially, the position of the camera was proposed to attach to a helmet that the person can wear. However, after reviewing the design requirements, it was apparent that for better stability the attaching the device with a camera on a vest that can be worn by the person who needs it.

3.2.1 Prototype of the Device

According to the explanation above, a front pocket will on the vest (see Figure 16), which is the residence for the Camera and the Raspberry Pi. The camera will be exposed to capture the real-time image from the environment. Three buckles will be fixed on two sides to make convenience for users adjusting depending on the users' size requirements, due to the working time of the device (assume eight hours per day), the material will be considered in this thesis work. Nylon or fabric can be considered as a suitable material in design. Two batteries are used to provide the power to the Raspberry Pi and the wristband separately.

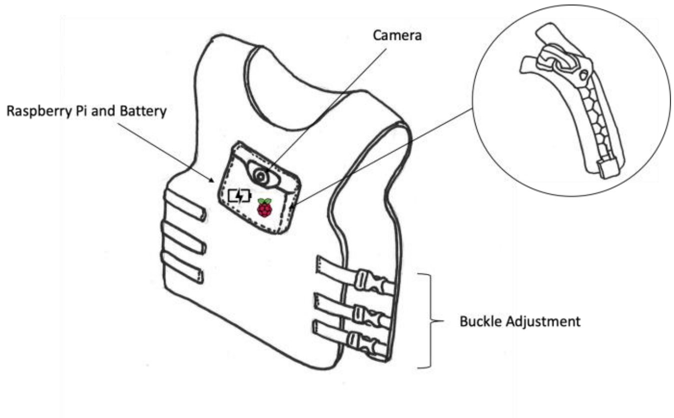


Figure 16. Prototype of the Vest

The design of the wristband is shown in Figure 17, the length of the band is adjustable. Because of the weight of the wristband, the high-performance Fluor elastomer material would be a good choice.

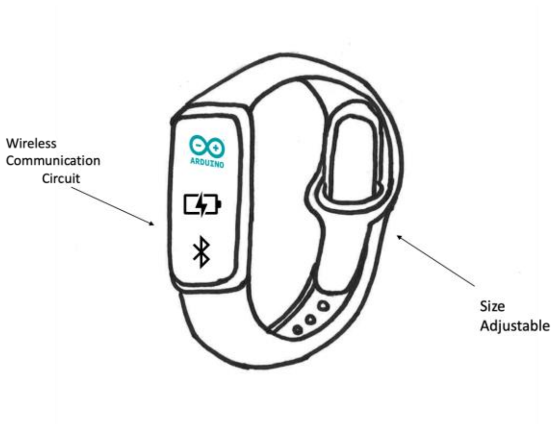


Figure 17. Prototype of the Wristband

The wristband is consisting of an Arduino Nano, a vibration motor, and a Bluetooth Module. The wristband will communicate with the Raspberry Pi through Bluetooth on the Arduino Nano board, which is shown in Figure 18. Once the Raspberry Pi obtains the image from the camera and processes it, it predicts the category of the image, which depends on the input. This prediction result is presented to the users with different frequencies via vibrator. Due to mechanical size and available functions, the Arduino Nano was a good choice to control the vibrator. The Arduino Nano operates as the slave device while the Raspberry Pi acts as the master device (see in Figure 18).

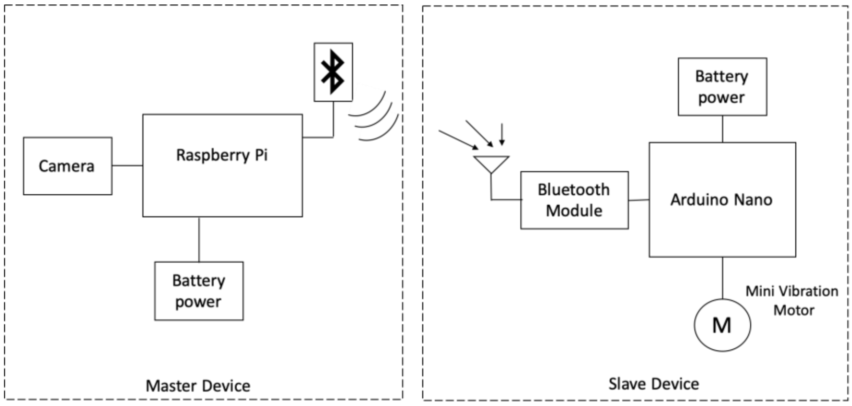


Figure 18. Wireless Communication of Device

3.3 Software Design

3.3.1 Computer Vision

In the computer vision, the aim is that the computers interpret what is presented in an image. For instance, the human recognizes the image based on the pre-existing database that is accumulated throughout the life cycle. However, the machine or computer is not able to process the data as a human does in his/her brain. Because of computer organizes the image data as a series of matrixes that made of the pixels and these pixels are color-coded [22]. Furthermore, these pixels are stored at a precise location in computer memory.

An image data is a stream of row numbers from an image sensor, and typically it is a grid or an array of color intensities of BGR (Blue, Green, and Red, respectively). Compare to human vision, the computer vision has several limitations, such as image sensors, lens limitation, viewpoint variations, and issues of scaling. Generally, images in the computer are stored in multi-dimension Arrays (see Figure 19).

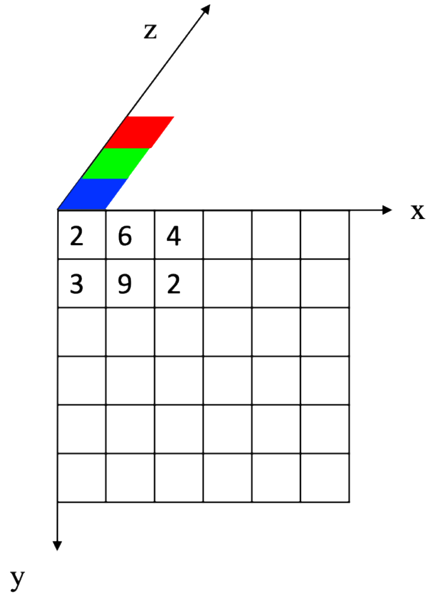


Figure 19. Image representation in Computer Vision

3.3.2 Deep Learning

Deep Learning (DL) is a machine learning technique that enables a machine to learn complicated patterns or representations in the presented data. For instance, assume that is an untrained brain, and thousands of similar images but with minor differences are presented to the brain. After 'training' the brain with these images, it can distinguish the differences among these images automatically. Therefore, DL takes advantage of the ability to understand the complicated environment and non-linear process dataset.

The Neural Network (NN) plays a vital role in deep learning, including binary classification, multiclass classification, and scaler regression [23]. These are the three most common application which is used in artificial intelligence and finance market (scaler regression). The layers of the NN are the building blocks of the deep learning data-processing module, which are also the filter of the input data. Most of the DL consists of multiple hidden layers that are connected to each other and resolve to a simple layer. This interconnected hidden layer architecture implements a form of progressive data distillation. In other words, the DL process is like a strainer for data that it cleans the data progressively. Briefly to say that the NN takes the input data to predict the output.

3.3.3 Convolutional Neural Network

The color image contains depth, length, and channel value, e.g., BGR. As a result, it increases the input parameters of DL, that in turn, increasing the input value will lead to an increase in training time. The DL algorithms can be accomplished with CNN or NN, according to the type of the input data. However, in comparison, CNN is more efficient than NN as the input of the DL is an image. In addition, CNN can scale the image data faster [24]. The DL algorithms that use CNN receives the image as input and passing it through a series of hidden layers to predict results on the output layer (see Figure 20).

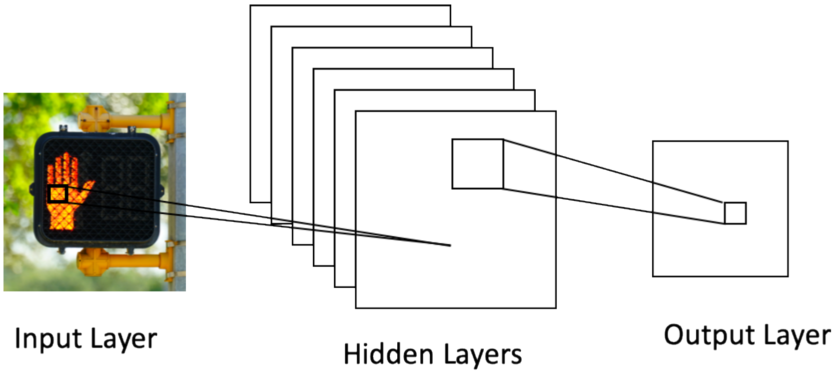


Figure 20. General Shape of CNN

The hidden layers have different functions to process the images to find the features of the input image (e.g., edge, color, and shapes). In the CNN architecture, the following layers are used: Convolution Layers, ReLU (Rectified Linear Unit) Layer, Pooling Layer, and Dense Layer (Fully Connected Layer) [24].

3.3.4 Convolutional Layer

Convolution is a mathematic term to describe the process of data filtering [25]. In this thesis, however, it is used to create the feature map for each following convolutional layer. It is an act of using a filter (kernel) that is applied to the input. In the CNN, convolution is executed by sliding the filter mask (kernel) over the input image. The filter mask has different weight values. This sliding process is simple math matrix dot production. Figure 21 explains the convolutional filtering process. This result of the sum is the first cell location (x_1, y_1) in the feature map. The output from the convolutional layer is called a feature map, which is used to the next layer.

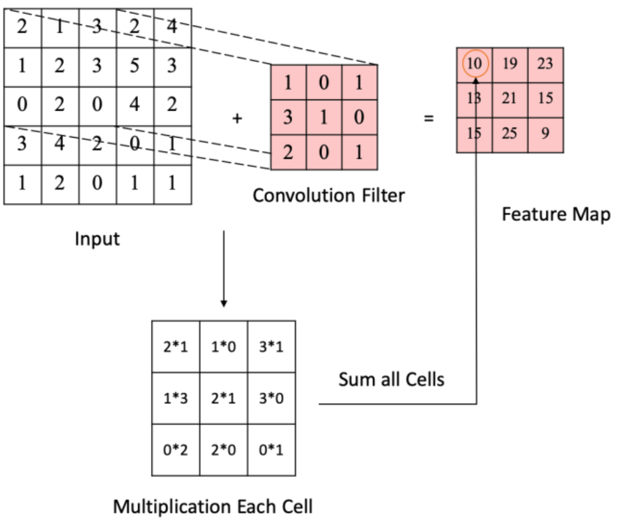


Figure 21. Sliding Process of Convolutional Filters

Figure 21 shows only the principles of the process, however, in this thesis work the input parameters is much larger than the example, but the calculation principle remains the same.

3.3.5 ReLU (Rectified Linear Unit) Activation Function

This is an activation function that is used to implement non-linear data to the model [25]. In most cases, the input layer is non-linear, on the other hand, the convolution process is linear. Therefore, the ReLU activate function is used to linearization. In the process of classification, the input data is always non-linear. As a result, that is one of the reasons to use the DL algorithms to make a classification. In this activation function, all the negative numbers assign to '0', and pass all the positive numbers (see in Figure 22).

$$\begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline 3 & -1 & 0 \\ \hline -2 & 0 & 1 \\ \hline \end{array} \xrightarrow[\text{ReLU Function}]{\text{Apply}} \begin{array}{|c|c|c|}\hline 0 & 0 & 1 \\ \hline 3 & 0 & 0 \\ \hline 0 & 0 & 1 \\ \hline \end{array}$$

Figure 22. Apply ReLU Function

3.3.6 Pooling Layer

The Pooling, also known as subsampling or downsampling [25], which is a simple process where it reduces the size or dimensionality of the feature map. In other words, the purpose of this reduction is to reduce the number of parameters needed to train, at the same time, avoiding losing the essential features of the image.

There are three types of Pooling, Max, Average, and Sum accordingly. The Max Pooling Layer is used in this thesis work because it is the most common type of

Pooling Layer (see Figure 23). Average and Sum Layer are excluded because they cannot maintain the essential parameters for the classification.

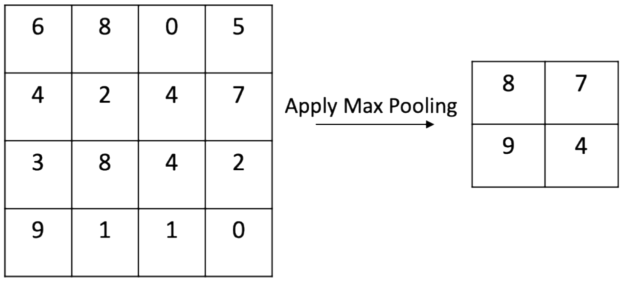


Figure 23. Apply Max Pooling Layer

According to Figure 23, the pooling has the effect of reducing the dimensionality (with and height) of the previous layer by half (4x4 to 2x2) and thus removing 75% of the parameters from the previous layer.

3.3.7 Dense Layer (Fully Connect Layer) and Softmax Activation Function

In the CNN architecture, the Dense Layer is placed after the Convolutional and Maxpooling Layers to connect every input parameter with the output [25]. During the classification task, the number of the outputs depends on the predicted output number; for instance, in this work, three classes lead to the three outputs through the fully connected layers. In this process, the Softmax activation function is used to produce the output of each category (classes or predicted outputs) on the bases of its probabilistic distributions. In other words, the output of probabilistic distributions of Softmax function is in the range of $0 < x < 1$, where x representing the output. Therefore, the sum of the three outputs is always 1. Sigmoid is another activation function that has a similar role. However, Softmax mostly works on the multiple-class classification, which was required for this thesis work. Note that, in the binary classification task, usually the Sigmoid activation function is used. Figure 24 shows the main layers and the activation functions used to build a CNN model.

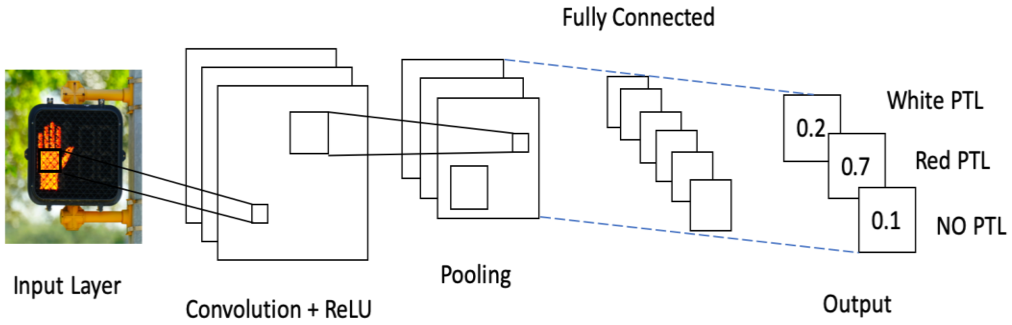


Figure 24. Basic Structure of a CNN model

Figure 25 shows the CNN model that is designed in this thesis work. The results in the table are based on the training loss and training accuracy value (refer the Chapter 4).

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d_1 (MaxPooling2	(None, 74, 74, 32)	0
conv2d_2 (Conv2D)	(None, 72, 72, 64)	18496
max_pooling2d_2 (MaxPooling2	(None, 36, 36, 64)	0
conv2d_3 (Conv2D)	(None, 34, 34, 128)	73856
max_pooling2d_3 (MaxPooling2	(None, 17, 17, 128)	0
conv2d_4 (Conv2D)	(None, 15, 15, 128)	147584
max_pooling2d_4 (MaxPooling2	(None, 7, 7, 128)	0
flatten_1 (Flatten)	(None, 6272)	0
dense_1 (Dense)	(None, 512)	3211776
dense_2 (Dense)	(None, 512)	262656
dense_3 (Dense)	(None, 3)	1539

Total params: 3,716,803
Trainable params: 3,716,803
Non-trainable params: 0

Figure 25. The Model Summary

CHAPTER 4

REAL-TIME PEDESTRIAN TRAFFIC LIGHT CLASSIFICATION

4.1 Introduction

The proposed real-time pedestrian traffic lights classification design is simplified in the flowchart, Figure 26. For proper use, the user should wear the vest and the wristband and turn on the device. Three buckles will be fixed on two sides to adjust the tightness of the vest depends on the users' size requirements. The front pocket of the vest, which is the residence for the Raspberry Pi and Camera. Therefore, the camera can continuously capture the image from the environment (the street view) conveniently. The input image from the camera is classified through the CNN model and then predict the result. According to the classification result, the actuator produces signals to the users.

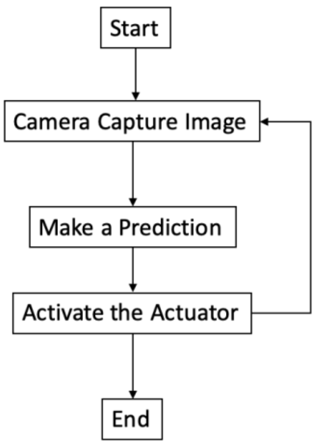


Figure 26. Flow Chart of Real-Time PTL Classification

4.2 Training the Pedestrian Traffic Lights Classifier

To train the PTL classifier, first, the CNN model is trained, and the then weights for the model are tuned. In the first step, the images are labeled manually according to the contents, red PTL, white PTL or no PTL (e.g., background) accordingly. As a result, three classes are created from the raw images. The next step of the training is to feed the individual dataset for each class into the preprocessing stage. This is required to resize and normalize the raw images. In the preprocessing step, the size of random input images needs to be changed to the fixed pixel size: 150 ´ 150 ´ 3 (Length ´ Width ´ Depth). Next, to normalize all resized images, each pixel of images is divided by 255, which results in pixel values between 0 and 1 (see Figure 27).

```
x_red=[]
y_red=[]
for i in range(len(red_light)):
    red=0
    im=cv2.imread(red_light[i])
    im = cv2.resize(im, dsize=(150, 150), interpolation=cv2.INTER_CUBIC)
    im = cv2.cvtColor(im, cv2.COLOR_BGR2RGB)
    im = im/255.
    x_red.append(im)
    y_red.append(red)

x_red=np.asarray(x_red)
```

Figure 27. Image Resize and Normalization

Then all preprocessed images with their labels are combined and shuffled randomly. Next, the combined dataset is split into two separate groups of images that are called training dataset and test dataset. The ratio of splitting is 70 percentage training and 30 percentage of the testing dataset, respectively. Additionally, the training dataset is divided into two parts to obtain a training dataset and validation dataset accordingly. Figure 28 shows the python programming code to create the training and testing dataset.

```
X, y = shuffle(x_data, y_data,random_state=0)
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=.3, random_state=47)
```

Figure 28. Dataset Shuffle and Split

Next, the training dataset is fed into a dummy CNN model in order to evaluate the model overfitting and to estimate the weight of individual layers for 20 epochs. The input of the CNN is feed with a batch (bundle) size of 50 images are taken from the training dataset. After, by visualizing the training and validation losses and accuracy, which is shown in Figure 29 and Figure 30; the final epochs are set for the main CNN model.

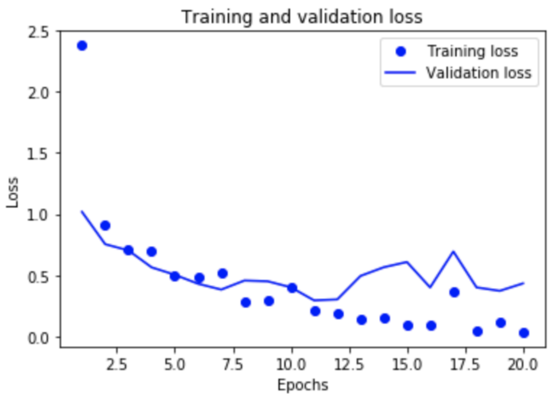


Figure 29. Training and Validation loss

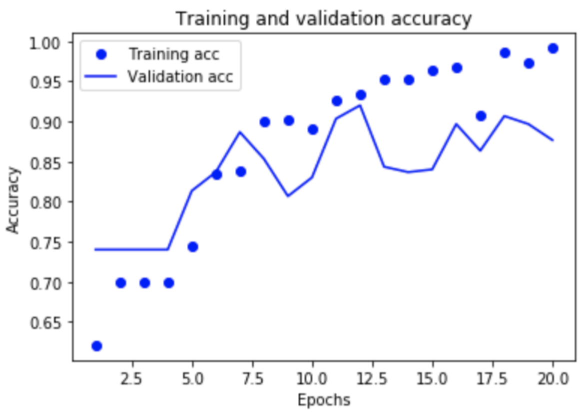


Figure 30. Training and Validation accuracy

Based on the result of the training and validation losses, and accuracy, the final CNN model and dummy CNN model architecture remain the same. The dummy classifier structure is modified and reduce from three hidden layers to two hidden layers to make the main model functions better. Next, the whole training dataset is fed into the final CNN model for training with 20 epochs. After training the CNN model, the test dataset is fed into for predicting the accuracy and for validating the output. Figure 31 shows the python code shows the loss and accuracy. Note that the loss is ~ 0.213 and the accuracy is ~ 94.6 %.

```
In [32]: results

Out[32]: [0.2128344473112883, 0.946472028746222]
```

Figure 31. Loss and Accuracy

After getting the PTL classifier model, the testing dataset is used to predict results to see the classifier's performance (see Figure 32). The three numbers (0, 1, 2) in Figure 32 represent three categories of the classifier. In the figure, a total 548 images are predicted in the testing dataset.

```
In [36]: print(x_predict1)

[1 2 0 0 0 1 2 0 2 0 0 0 0 0 0 0 0 0 2 0 0 0 2 2 1 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 2 0 0 0 0 2 2 0 0 0 0 0 2 0 2 0 0 0 1 0
 0 0 1 0 0 2 2 2 2 0 2 0 0 1 0 0 0 0 0 0 0 1 0 2 0 1 2 0 2 1 0 1 0 0 0 0 2
 1 0 1 0 0 0 0 2 0 2 2 2 0 2 2 0 2 0 0 0 1 0 0 0 0 0 0 2 1 1 2 0 2 0 0 0 1
 0 0 1 0 0 1 2 0 0 2 1 1 2 1 0 0 0 0 0 1 1 0 0 0 0 1 0 0 0 2 1 0 0 0 1 0 2
 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 2 0 0 0 0 0 0 2 2 0 0 2 2 2 0 2 0 0 2
 1 1 0 0 1 0 0 0 0 0 0 2 2 2 0 0 2 0 0 0 1 2 0 1 2 0 0 0 0 0 1 2 2 2 0 2 1
 2 0 0 0 0 0 2 2 0 2 0 1 0 0 2 2 0 0 0 2 0 0 1 1 2 0 2 2 0 0 2 0 2 0 0 2 2
 0 0 0 0 1 0 0 0 0 1 2 0 0 0 0 0 0 0 1 2 2 0 0 0 0 1 2 2 0 2 0 0 0 0 0 1 0
 0 0 0 0 0 0 2 2 0 2 0 2 0 0 0 0 2 0 2 0 0 0 0 0 0 2 2 0 0 0 0 1 0 0 0 0 2
 1 0 1 1 1 0 0 0 2 0 0 0 0 1 2 0 2 1 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 2 0
 2 0 0 0 0 0 0 0 2 1 0 0 0 0 0 0 0 0 2 0 2 1 2 1 2 1 2 0 1 0 0 1 2 0 2 0 2
 2 2 2 0 0 0 2 2 2 2 0 0 2 0 0 0 2 0 1 0 2 2 0 0 2 0 0 1 1 0 1 0 0 2 1 2 1
 0 1 0 0 0 0 1 2 0 1 2 2 1 0 0 2 2 0 0 0 0 0 0 0 0 1 0 0 2 0 0 0 2 1 0 2 1
 2 0 0 1 0 0 0 2 0 2 0 0 0 2 0 2 0 0 0 0 0 0 0 2 0 0 0 0 0 0]
```

Figure 32. Predict Result on Test Dataset

Figure 33 shows the python programming code to save the weights on the disk for later use. Besides, Figure 34 shows the block diagram of the training PTL classification.

```python
# serialize model to JSON
model_json = model.to_json()
with open("trafficlightmodel2.json", "w") as json_file:
    json_file.write(model_json)
# serialize weights to HDF5
model.save_weights("trafficlightmodel2.h5")
print("Saved model to disk")
```

```
Saved model to disk
```

Figure 33. Save the Model and Weights

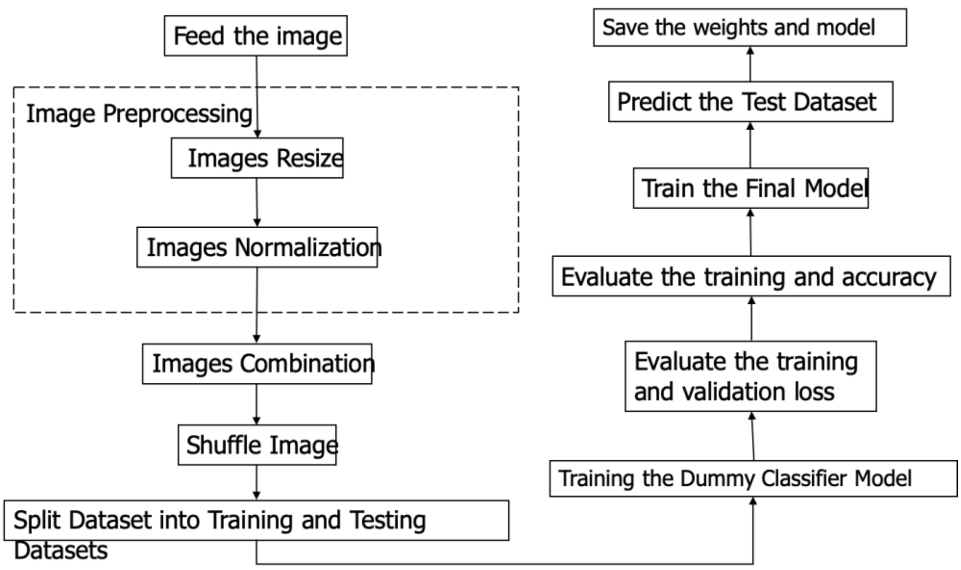


Figure 34. Flowchart of Training Method

4.3 Testing the Pedestrian Traffic Lights Classifier

The first stage is getting a real-time image for testing the PTL classifier. After the camera is activated, the real-time video is recorded and convert to the individual frames. Each frame goes to the preprocessing step before feeding into the classifier to predict the result. The frame size is changed to the $150 \times 150 \times 3$ and also normalized by divided by 255. Then the frames are fed into the final CNN classifier model to make prediction results. The results may have three classes are shown by numbers, 0, 1, 2, accordingly. The number '*0*' means red, '*1*' means white, and '*2*' means background (Table1).

Table 1 Explanation of the Classifier Result

Classes (Categories)	Meaning
0	Red pedestrian traffic lights
1	White pedestrian traffic lights
2	No pedestrian traffic lights

After obtaining the prediction result, the Raspberry Pi sends the result to the slave device, i.e. Arduino Nano through the Bluetooth. The Figure 35 shows the python programming code to connect Bluetooth.

```
bd_addr = "98:D3:32:10:7A:44"
port = 1
sock = bluetooth.BluetoothSocket(bluetooth.RFCOMM)
sock.connect((bd_addr,port))
```

Figure 35. Connecting the Bluetooth Module

Arduino receives the signal from the master device, and depends on the value, the actuator (vibration motor) vibrates with different frequencies. For instance, if the value ‘*0*’, vibrates two times with 3 seconds interval, if the value is ‘*1*’, then it vibrates once with 5 seconds interval, and if the value is ‘*2*’, then the vibration motor keeps silence (Table 2).

Table 2 Explanation of the Vibration Frequency

Result	Vibration frequency
0	Two times vibration
1	One-time vibration
2	Silence

The Figure 36 shows the flowchart of testing the real-time PTL classifier.

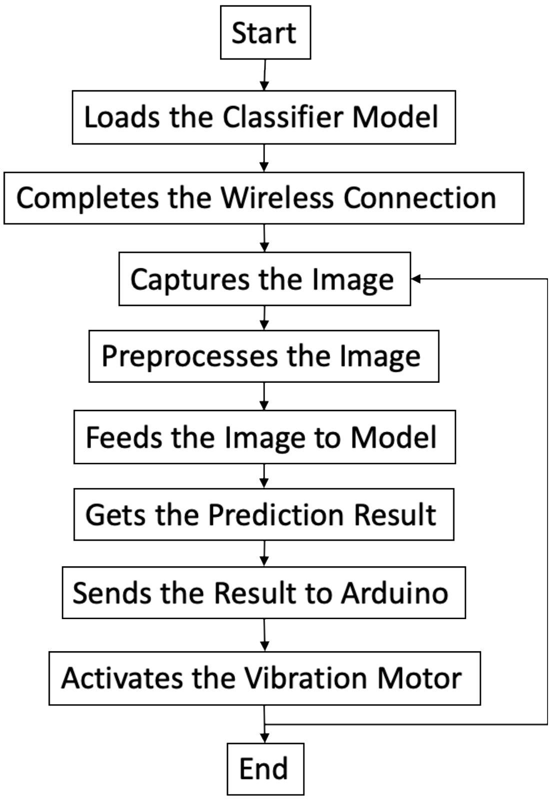


Figure 36. Flowchart of Real-Time PTL Classifier Testing

4.3.1 Pedestrian Traffic Light Simulation

To optimize and make the testing reliable, the experiments need to repeat several times in the lab. For that purpose, the test tools are manufactured to simulate the outdoor PTL. Additionally, a display is required to observe the camera input and examine the prediction result regarding the input image. For the lab simulation, several PTL is captured by the camera in advance. These images were not belonging to any dataset that is used to train the final CNN classifier. In other words, these images were totally new images capture for solely testing in the lab environment. In order to have a reliable simulation, these images are selected to represent two different conditions of PTL. Depends on streets' shapes and sizes, the test images are taken in different distances, i.e., near or close. These test images also have diverse backgrounds and contain both red PTL and white PTL.

Next, the test images are color printed separately and glued on the cardboard with the same size as the image. To hold the cardboard and able to change the picture swiftly, a PVC pipe is used with a diameter of 20mm and 300mm length. During the experiment, the 'PTL' is held in front of the camera so that the hand will not be in the frame. This was necessary to achieve an original street scene for a PTL as much as possible. Figure 37 shows one of the test tools. The detailed experimenting images are shown in Appendix C.

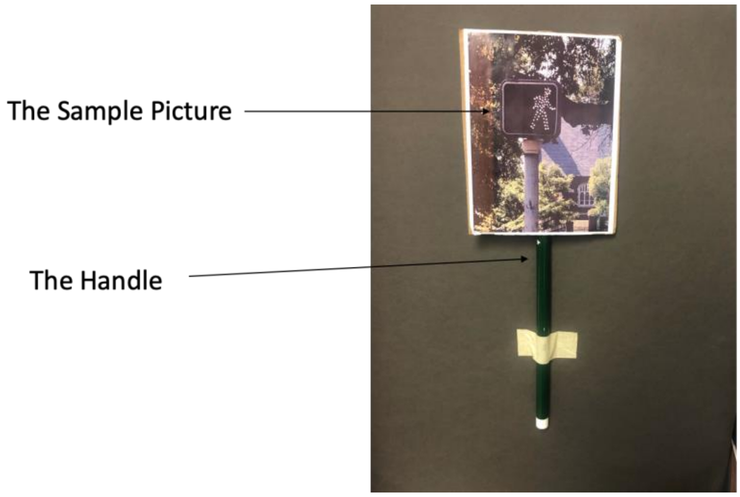


Figure 37. Test Tool in the Experiment

4.4 Display Outputs

The programming of the classifier is written in Python. Besides Python on Raspberry Pi, Arduino sketch is also used to program wireless communication with the host (Raspberry Pi) and the slave (Arduino Nano). In order to run classifier, the terminal option on the Raspberry Pi is used. Programming in Python and executing the program, the OpenCV environment is created on Raspbian Operating System. Figure 38 shows the OpenCV environment on Raspberry Pi.

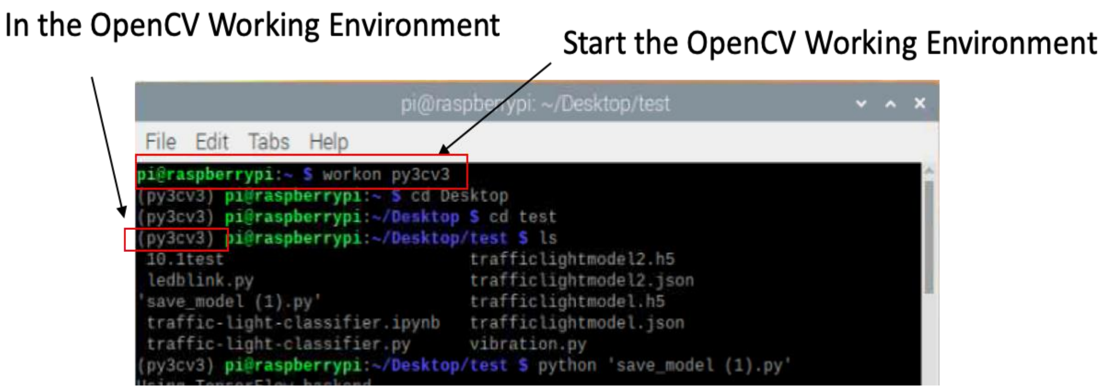


Figure 38. Start OpenCV Working Environment

Figure 39 shows the typical lab environment that the classifier was tested. On the right-side window without PTL, and on the left side, the real-time result of the classifier is shown. On the left side, window shows that the classifier did not identify PTL (indicate with ‘*2*’ as an output) on the camera image (right window).

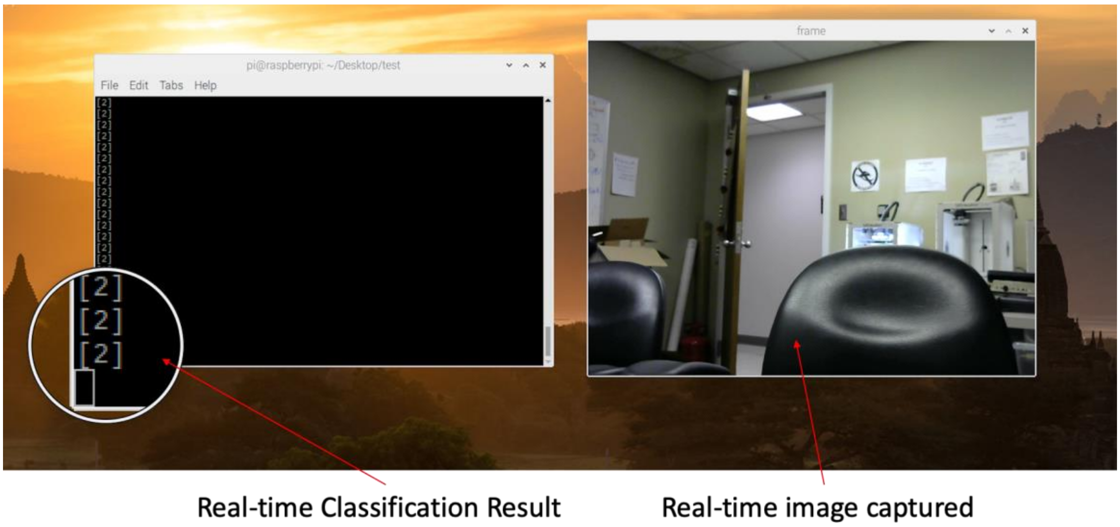


Figure 39. Real-Time Classification of Background-No PTL

Next, the simulator image with red PTL is presented to the camera (see Figure 40). The classifier is identified the red PTL and sends the signal to the confirm with number ‘*0*’. The left window terminal shows the prediction result. This result is sent to the Arduino Nano to give vibration feedback to the users, i.e., visually impaired persons. Similarly, in Figure 41, it shows the case if the classifier identifies valid white PTL in the image, the classifier sends the number ‘*1*’ to the terminal to send the signal to a wristband on the user’s arm.

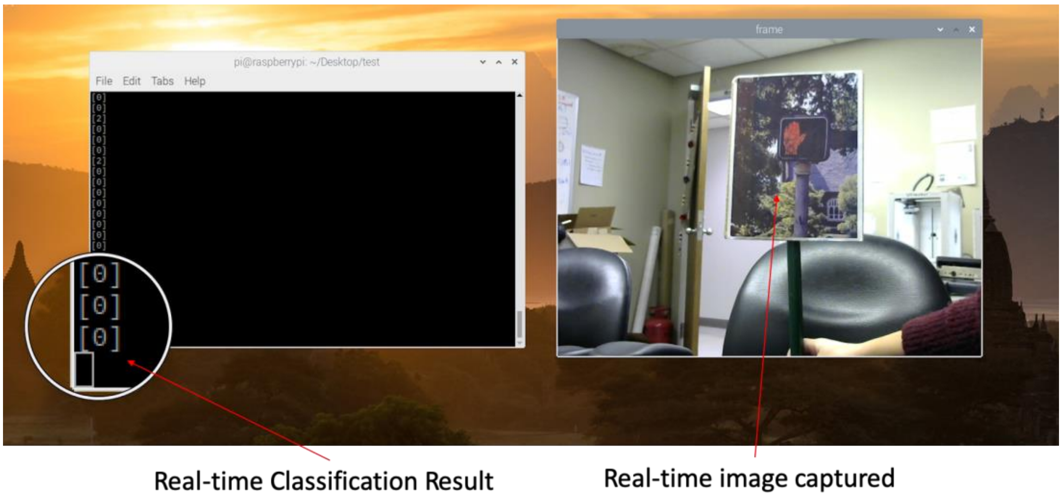


Figure 40. Real-Time Classification of Red PTL

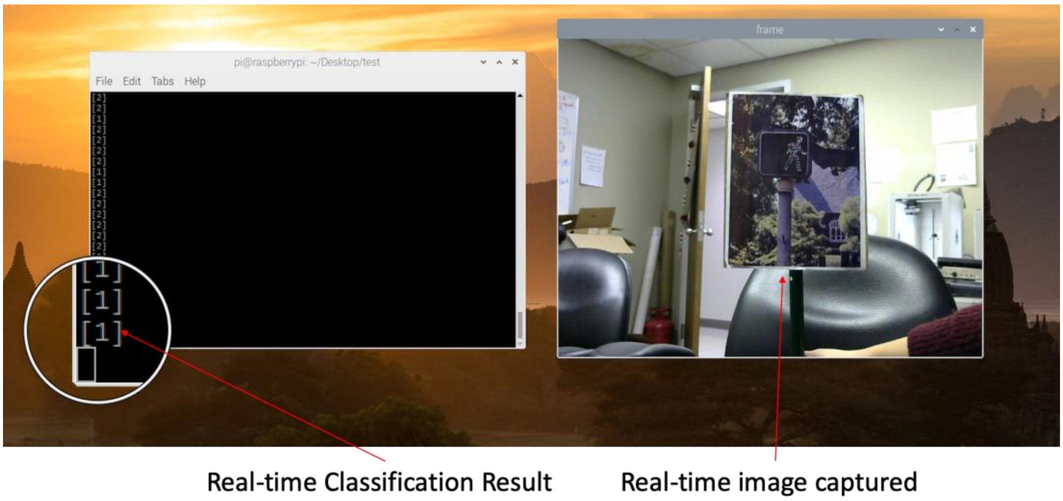


Figure 41. Real-Time Classification of White PTL

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

5.1 Summary

In this thesis work, the input signal from the camera capturing is the visual signal, and the results are shown through the different frequencies of the vibration. Thus, the output can only be described in the text form in this thesis. Later a video demonstration will be attached to this thesis work. According to the demonstration of the result in chapter four, a wearable device that can recognize the PTL and give feedback to the users is completed. This device departs the display showing and the Internet using during the working time. Users can 'wear' this device when they are walking with the accuracy is in the range of 94% to 97%.

5.2 Difficulties and Solutions

5.2.1 Raspberry Pi system installing

In the progress of downloading the Raspberry Pi, the system is downloaded in the form of the zip file, and the whole zip file should be transferred to the SD card and then plug into the Raspberry Pi board to initiate the system. When the first time to transfer the file to the SD card, the file is copied and pasted on the SD card directly, which is unaccessable to be read on the board. The reason for that is the file is the image of the operating system. It can not be copied and pasted to the SD card. Instead, a certain software (balenaEtcher) is needed to write the image into the SD card (see Figure 42). This software is both workable on the Mac and Windows system.

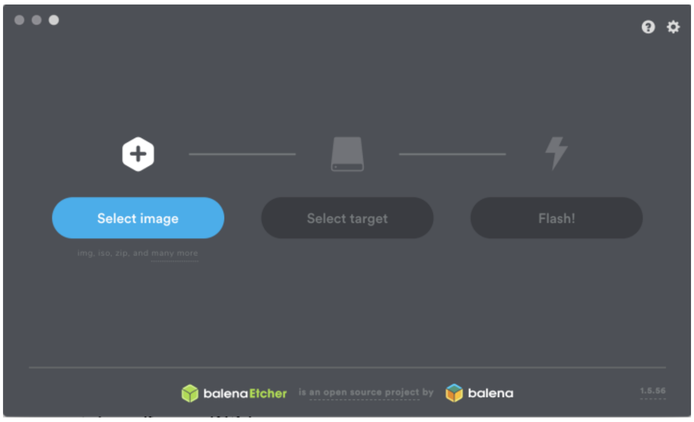


Figure 42. Software Used to Write the Image to SD Card

5.2.2 OpenCV installed on the Raspberry Pi

OpenCV worked as a tool to process the image, and camera operation is the first tool package that needs to be installed on the Raspberry after setting up the system while installing OpenCV is not enough to install the OpenCV only. The first step is to expand the filesystem include all the free space on the SD card. Secondly, create the dependencies, which means some input/output packages for reading the image in a different format (for example, JPEG, PNG, TIFF, etc.) as well as the packages for input/output the video. Besides, some packages for displaying the images and videos. Thirdly, download the OpenCV source code. In this step, either based on the python2.7 or python3 can be installed. Fourthly, creating the Python virtual environment is optional to download. First, it’s important to understand that a virtual environment is a special tool used to keep the dependencies required by different projects in separate places by creating isolated, independent Python environments for each of them. In Figure 43 below shows how the OpenCV in virtual environment work.

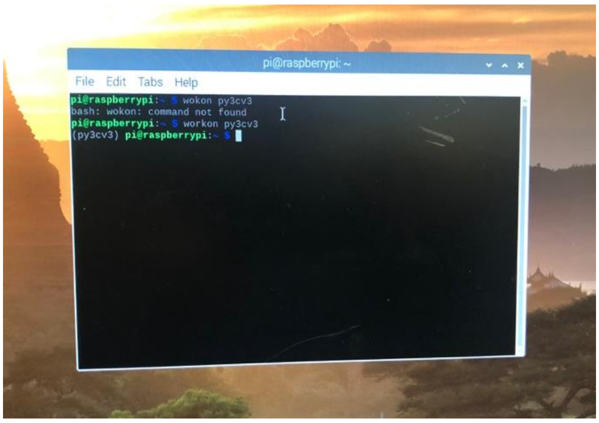


Figure 43. Work in the OpenCV Virtual Environment

5.2.3 Permission Denied While Installing the Raspberry Pi

While installing the package in the Raspbian System, there is an error: Permission denied. This may lead the system cannot open a certain file. This may because the system only executes the order from the original user. For this problem, some little change needs to add after the command. Add '- - user 'after the command that means told the system is the superuser is executing the command (see Figure 44.).

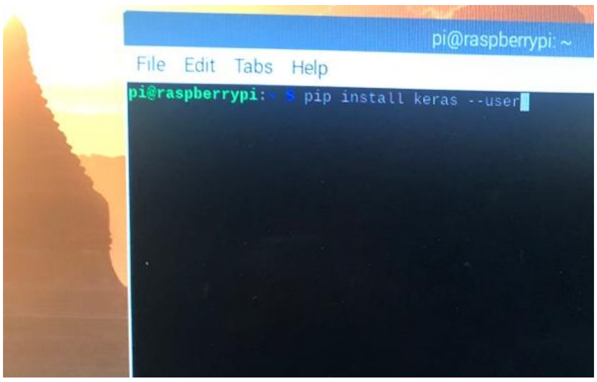


Figure 44. Command as the Original User

5.2.4 Image Classification with Background

Initially, the classifier was made as a binary classifier (two objects), the binary classification is workable on the prediction of the PTL; it can only give the result. However, in this thesis project a real-time prediction is needed, thus, the background is also needed to be included in the project because while the device is starting to work, when there are no traffic lights the device should not give any vibration feedback to the user, so the third class is included in the classifier to optimize the design.

5.3 Future Work

5.3.1 Size Improvement of the Wristband

Consider the using time of the device and meet the requirements of the size and weight limitation of the Bluetooth wristband. It is necessary to have a smaller proportion of the microcontroller, and the Bluetooth should be contained. For the later work, ‘Beetle BLE’ can be one option as a component as the smallest Arduino board (28.8mm X 33.1mm) (see Figure 45). Besides, it contains the Bluetooth module on the board, which is another advantage of this board. It works with a 5v power supply, four digital pins, and four analog pins on the board. This is another good option after the Arduino Nano board. More spaces will be saved on the wristband. The price of the Beetle BLE is ranging between 15$ and 18$.

Figure 45. Beetle BLE

5.3.2 Extreme Weather Condition

Extreme weather could influence the accuracy of the result. For instance, on a rainy day, the camera might be covered by water drops, which will lead to a distortion of the image. Such distorted images will result in misinterpretation of the image and deliver false outputs that might be fatal results. In the same manner, heavy snow can cover the objects on the street so that it may increase the load of the device because too many similar objects need to be classified. Although, an all-weather device could be considered even it would increase the cost.

5.3.3 Wider Application Based on Different Dataset

In this thesis work, an assistive device for visually impaired people have been discussed. The result of this work can be the basis for developing more advanced and affordable devices. For instance, with a more extensive and diverse image dataset, not only the traffic lights can be classified, but also a variety of vehicles, other traffic signs, and various store signs can be identified. In order to achieve that, a larger and diverse dataset needs to be prepared and utilized with this framework. Such a solution for visually impaired people will offer more convenience and secure social integration to the society.

5.3.4 Diverse Transformations of Output Signals

Another improvement can be made on the signaling of the traffic condition: For instance, the feedback to the user may not be limited to vibration, but also using more advanced applications such as “Siri” to inform the user with more detailed information. Such an application would eliminate the long training phase for the device. In such a case, a Bluetooth earphone, for instance, could be used to listen to the device conveniently.

Using the “Siri” type application will also reduce the confusion with vibration, which can be difficult by elderly persons to distinguish the meanings of different vibration frequencies. Notably, some elderly persons may be confused with it, because they may forget the meaning of the different frequency of the vibrator.

LIST OF REFERENCE

[1] Braille. (2019, November 5). Retrieved from: https://en.wikipedia.org/wiki/Braille

[2] White cane. (2019, October 15). Retrieved from: https://en.wikipedia.org/wiki/White_cane.

[3] Baker, A. P. (2004). Liakhoff, Nicolai [Nikolay Liakhov] (1897–1962), pioneer of guide dog training. Oxford Dictionary of National Biography. doi: 10.1093/ref:odnb/71948

[4] Global data on visual impairment. (2017, December 8). Retrieved from https://www.who.int/blindness/publications/globaldata/en/.

[5] Traffic light. (2019, November 14). Retrieved from: https://en.wikipedia.org/wiki/Traffic_light.

[6] Elmannai, W., & Elleithy, K. (2017). Sensor-Based Assistive Devices for Visually-Impaired People: Current Status, Challenges, and Future Directions. Sensors, 17(3), 565. doi: 10.3390/s17030565

[7] Dakopoulos, D.; Bourbakis, N.G. Wearable obstacle avoidance electronic travel aids for blind: A survey. IEEE Trans. Syst. Man Cybern. Part C 2010, 40, 25–35.

[8] Wahab, A.; Helmy, M.; Talib, A.A.; Kadir, H.A.; Johari, A.; Noraziah, A.; Sidek, R.M.; Mutalib, A.A. Smart Cane: Assistive Care for Visually-impaired People. Int. J. Comput. Sci. Issues 2011, 8, 4

[9] Oliveira, J. F. (2013). The path force feedback belt. 2013 8th International Conference on Information Technology in Asia (CITA). doi: 10.1109/cita.2013.6637564

[10] Brilhault, A., Kammoun, S., Gutierrez, O., Truillet, P., & Jouffrais, C. (2011). Fusion of Artificial Vision and GPS to Improve Blind Pedestrian Positioning. 2011 4th IFIP International Conference on New Technologies, Mobility and Security. doi: 10.1109/ntms.2011.5721061

[11] Teach, Learn, and Make with Raspberry Pi – Raspberry Pi. (2019). Retrieved from http://raspberrypi.org/.

[12] Webcams for Video Conferencing and Video Calling. (2019). Retrieved from: https://www.logitech.com/en-us/video/webcams.

[13] ArduinoNano. (2019). Retrieved from: https://www.arduino.cc/en/Guide/ArduinoNano.

[14] HC-05 Bluetooth Module Pinout, Specifications, Default Settings, Replacements & Datasheet. (2019). Retrieved from https://components101.com/wireless/hc-05-bluetooth-module.

[15] Miniature Vibrating Motors - RobotShop. (2019). Retrieved from https://www.robotshop.com/en/miniature-vibrating-motors.html.

[16] NumPy. (2019.). Retrieved from https://numpy.org/.

[17] pyplot. (2019). Retrieved from https://matplotlib.org/api/pyplot_api.html.

[18] Scikitlearn. (2019). Retrieved from https://scikit-learn.org/stable/index.html.

[19] OpenCV. (2019, October 14). Retrieved from https://opencv.org/.

[20] Keras: The Python Deep Learning library. (n.d.). Retrieved from https://keras.io/.

[21] Macroplant. (n.d.). Adapter Free Image and Video Converter for Mac and PC. Retrieved from https://macroplant.com/adapter.

[22] Batchelor, B. G. (2012). Machine vision handbook. London: Springer.

[23] Ketkar, N. (2017). Introduction to Deep Learning. *Deep Learning with Python*, 1–5. doi: 10.1007/978-1-4842-2766-4_1

[24] Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, *9*(4), 611–629. doi: 10.1007/s13244-018-0639-9

[25] Liu, L., Shen, C., & Hengel, A. V. D. (2015). The treasure beneath convolutional layers: Cross-convolutional-layer pooling for image classification. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. doi: 10.1109/cvpr.2015.7299107

[26] Chollet François. (2018). Deep learning with Python. Shelter Island, NY: Manning Publications Co.

[27] Bai, J., Lian, S., Liu, Z., Wang, K., & Liu, D. (2017). Smart guiding glasses for visually impaired people in indoor environment. IEEE Transactions on Consumer Electronics, 63(3), 258–266. doi: 10.1109/tce.2017.014980

[28] Cengil, E., Cinar, A., & Guler, Z. (2017). A GPU-based convolutional neural network approach for image classification. 2017 International Artificial Intelligence and Data Processing Symposium (IDAP). doi: 10.1109/idap.2017.8090194

[29] Charette, R. D., & Nashashibi, F. (2009). Traffic light recognition using image processing compared to learning processes. 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems. doi: 10.1109/iros.2009.5353941

[30] Lindner, F., Kressel, U., & Kaelberer, S. (n.d.). Robust recognition of traffic signals. IEEE Intelligent Vehicles Symposium, 2004. doi:10.1109/ivs.2004.1336354

[31] Wang, C., Jin, T., Yang, M., & Wang, B. (2011). Robust and Real-Time Traffic Lights Recognition in Complex Urban Environments. International Journal of Computational Intelligence Systems, 4(6), 1383. doi: 10.2991/ijcis.2011.4.6.32

[32] Zheng, Z., Liu, W., Ruby, R., Zou, Y., & Wu, K. (2017). ABAid: Navigation Aid for Blind People Using Acoustic Signal. 2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS). doi: 10.1109/mass.2017.37

APPENDIX

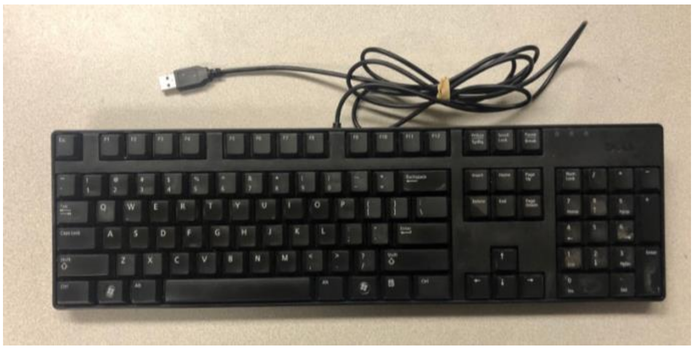
A. Summary of the PTL Worldwide

Country	PTL Type	Image Illustrate
USA	1. Green, blue, or white walking person or "walk": cross with caution (pedestrians have the right of way; motorists turning left or right must yield to pedestrians). 2. Flashing red or orange stop hand or "don't walk": do not start crossing but continue if already in the middle of the intersection. 3. Red or orange stop hand or "don't walk": do not enter the intersection.	
C H I N A	1.Green: safe to cross. 2. Red: do not cross. 3. Yellow (solid, after green, before red): continue to cross only if unable to stop safely.	
J A P A N	1. Blue: safe to cross 2. Yellow: cross only if it is safe 3. Red: do not cross	

E U R O P E	1. Green: safe to cross. 2. Yellow or orange: continue to cross only if unable to stop safely. 3. Flashing yellow or orange: cross with caution (often used when lights are out of order or shut down). 4. Red: do not cross.	
G E R M A N Y	1.Green: safe to cross. 2. Orange: continue to cross only if unable to stop safely. 3. Flashing orange: cross with caution, obey signage (used when lights are out of order or shut down). 4. Red: do not cross. 5. Red and orange: do not cross, prepare for green.	
The United Kingdom and Ireland	1. A still image of a green walking person: cross the road 2. Flashing green walking person: continue to cross if already on the crossing but do not start to cross 3. Red standing man: do not cross/do not start to cross	

Table3. Summary of the PTL Worldwide [5]

B. Raspberry Pi Board Accessories

Name	Figure
S D C A R D	
K E Y B O A R D	
D I S P L A Y	

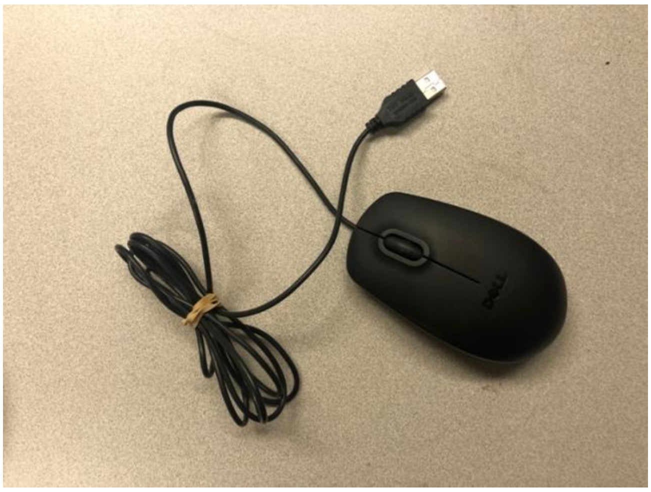
M O U S E	
WIFI Module For Raspberry Pi	

Table 4. Raspberry Pi Board Accessories

C. Sample Picture for Real-time PTL Classification

Type	Sample Picture
Red PTL Smallest	
Red PTL Medium	

Red PTL Largest	
White PTL Smallest	

White PTL Medium	
White PTL Largest	

Table 5. Sample Pictures