B.E. Project Report

On

INTELLIGENT SYSTEM FOR HELMET DETECTION USING RASPBERRY PI

Submitted by

Mr. Athuljith MK	B120368570
Mr. Biren Patel	B120368591
Mr. Sourabh Pardeshi	B120368587
Mr. Vivien Rajguru	B120368600

Under the guidance of

Prof. N. S. More

Department of Information Technology

STES's, Smt. KashibaiNavale College of Engineering, Pune-41

Accredited by NACC

SAVITRIBAI PHULE PUNE UNIVERSITY

2016-2017

Sinhgad Technical Education Society,

Department of Information Technology

Smt. Kashibai Navale College of Engineering ,
Pune-411041

Date:

CERTIFICATE

This is to certify that,

Mr. Athuljith MK	B120368570
Mr. Biren Patel	B120368591
Mr. Sourabh Pardeshi	B120368587
Mr. Vivien Rajguru	B120368600

have successfully completed this project report entitled “**INTELLIGENT SYSTEM FOR HELMET DETECTION USING RASPBERRY PI**", under my guidance for partial fulfillment of the requirements of the degree of Bachelor of Engineering in Department of Information Technology of University of Pune during the academic year 2016-17.

Date:

Place: SKNCOE, Pune.

Prof. N. S. More	**Prof. R. H. Borhade**
Project Guide	Head of Department
Dr. S. N. Gujar	**(External Examiner)**
Project Coordinator	Savitribai Phule Pune University, Pune

ABSTRACT

Intelligent System for Helmet Detection using Raspberry Pi ensures helmet possession by a motorcyclist at all times by capturing a snapshot of the rider’s helmet using Pi Camera and confirming object detection by Haar cascading technique. The main idea behind the project is to reduce road fatalities among motorcyclists. Due to ignorance of riders for the Helmet compulsion law, every motorcyclist's safety has been compromised. Intelligent System for Helmet Detection solves this problem by leaving the rider no choice as the engine of the vehicle is operated through a single channel relay which is only closed after detection of the rider's helmet. An interactive LED display will alert the rider if the helmet is not detected after which the rider needs to ensure the possession of a helmet or else the System will display a warning message which will earn the rider a strike if it is ignored. Three such strikes will lead to the RTO penalizing the rider for breaking the law.

Keywords: *Intelligent System, Helmet, Raspberry pi, Haar Cascade, Motorcyclist*.

ACKNOWLEDGEMENT

We have taken efforts in this project, however, it would not have been possiblewithout the kind support and help of many individual and organizations, and wewould like to extend our sincere thanks to all of them.

We are highly indebted to **Prof. N. S. More** for his guidance and constant supervisionas well as for providing necessary information regarding the project & also forsupport in completing the project.

We are also obliged to the project coordinators Dr**. S. N. Gujar** and **Prof. Y. V. Wagh** and the facility of IT department for providing us with facilities to lab, it wouldhave been impossible for us to complete project without their valuable guidance andprompt coordination.

We would like to express our gratitude towards our head of department **Prof R. H. Borhade** (Information Technology) for their kind co-operation and encouragementwhich help us in completion of this project.

We are very much thankful to **Dr. K. R. Borole** Vice-principal, Smt. Kashibai Navale College of Engineering and **Dr. A. V. Deshpande** Principal, Smt. Kashibai Navale College of Engineering for providing all facilities and support to meet our projectrequirements.

Our thanks and appreciations also go to colleagues in developing the project and thepeople who have willingly helped us out with their abilities.

Athuljith MK (B120368570)

Biren Patel (B120368591)

Sourabh Pardeshi (B120368587)

Vivien Rajguru (B120368600)

List of Tables

List of Figures

List of Abbreviations

Abbreviations	Full form
DC	Direct Current
FCC	Federal Communications Commission
GPS	Global Positioning System
GPU	Graphic Processing Unit
HD	High Definition
OS	Operating System
RTO	Regional Transport Office
UML	Unified Modeling Language
XML	Extended Markup Language

Contents

Chapter 1
INTRODUCTION

1.1 MOTIVATION

Two-wheelers, the mode of transport which most Indians use, continue to be the most vulnerable to accidents. Indian roads were at their deadliest in 2015 claiming more than 16 lives every hour on average. Over 1.41 lakh people died in crashes, 3% more than the number of fatalities in 2014. Accidents involving two-wheelers and accounted for nearly half of the lives lost in road crashes. This led us to develop a system which would entitle the safety of the Motorcyclist and to contribute our innovative plan which could develop a plummet fall in rate of accidents. It will also ensure us that brain power in our country to develop by saving their lives.

1.2 PROBLEM STATEMENT

To design and develop a Smart Helmet for motorcyclists using Raspberry Pi and its modules which will ensure compulsory possession of a helmet with every motorcyclist. Scan the standard symbol using the Pi Camera module and verify if the helmet is present using image processing in Python.

1.3 BASIC CONCEPT

To make use of the Open computer vision library to process the buffered images captured through a raspberry pi camera module and highlight the detected object with a rectangle. After this process the python script will decide whether or not to provide access to the motorcyclist.

1.4 PROJECT OBJECTIVE

The system should be able to detect helmets in real time. The system should use an Haar-cascaded XML file to test the captured image or video. System should have a calibration of different helmets that are likely to be used by rider of motor cycle. System should detect attempts of failure and report data to RTO for further actions.

1.5 SCOPE OF PROJECT WORK

The scope of the project is to create an efficient system to detect the helmet on the rider's head. We will be using a raspberry pi as the processor which indeed will process the data. One of the applications will count the helmets wore. Another application will keep track on number of failure attempt.

1.6 APPLICATION

a) Smart Vehicle Detection Systems.
b) Helmet Detection at Traffic Signals.
c) Smart RTO Server Systems.
d) Helmet Detection at Ore Mines.

1.7 ORGANIZATION OF REPORT

Chapter 1: Introduction

It introduces the project's idea along with the motivation behind the project. Motivation from the growing traffic rush and the objectives of the projects and applications of project are discussed.

Chapter 2: Literature Survey

Here the research work already done for these problems and their proposed models are discussed. Different models were proposed.

Chapter 3: Project Statement

It includes the actual problem statement, goals and objectives of project. It also contains the various technologies used for development of project.

Chapter 4: System Requirement and Specification

It contains the purpose and scope of documentation, detailed system architecture as well as hardware and software requirements.

Chapter 5: System Design

Here all the design part and flow of system is discussed with the help of low level and high level diagrams with detailed description of each.

Chapter 6: Methodology

It introduces the algorithm which is actually used for implementation of the system. It also contains various test strategies and manual test cases of the system.

Chapter 2

LITERATURE SURVEY

2.1 INTRODUCTION

Two-wheelers, the mode of transport most Indians use, continue to be the most vulnerable to accidents. Indian roads were at their deadliest in 2015 claiming more than 16 lives every hour on average. Over 1.41 lakh people died in crashes, 3% more than the number of fatalities in 2014. Accidents involving two-wheelers and accounted for nearly half of the lives lost in road crashes. While 13,787 two-wheeler drivers were killed in crashes, 23,529 other people were killed in accidents involving these vehicles, while close to 1.4 lakh people were left injured in them.

The top five states - Uttar Pradesh, Tamil Nadu, Maharashtra, Karnataka and Rajasthan - accounted for over 40% of the fatalities. Among 53 mega cities, Delhi registered the highest number of fatalities at 2,199 and Maharashtra recorded 1,046 such deaths. Bhopal and Jaipur ranked third and fourth with the city roads claiming 1,015 and 844 lives respectively.Protective smart gear to increase road safety among motorcyclists. The idea is obtained after observing that there are nearly 2500 motorcycle accidents per day in Maharashtra itself.

With the use of image processing unit in Raspberry pi, the motorcycle can only be started if the rider possesses the helmet. A magnetic relay is attached to the raspberry pi circuit which detects if the standard symbol is identified or not. Since two wheelers are the most used transport means in India, this project has a wide application in the field of transport and vehicle management. Speeding and not wearing a helmet are the two most common reasons behind motorcycle fatalities. This project will increase rider safety and increase chance of life even though an accident occurs.

Table 2.1 - Statistics of Road Accidents [5]

Parameter	2014	2015	% change over previous year
Total Accidents in the country	4,89,400	5,01,423	2.5
Total number of Persons Killed in the country	1,39,671	1,46,133	4.6
Total number of Persons Injured in the country	4,93,474	5,00,279	1.4
Accident Severity*	28.5	29.1	2.1
* No. of persons killed per 100 accidents			

2.2 RELATED WORK DONE

A safety helmet is one of the most important items of personal protective equipment used by motorcycle riders primarily to protect the head against impact. Death of riders could have been avoided, if they had been wearing helmets, many of these deaths and disabling injuries would not have happened. Helmets won't prevent accidents, but they clearly will cut down on deaths and injuries when collisions occur.

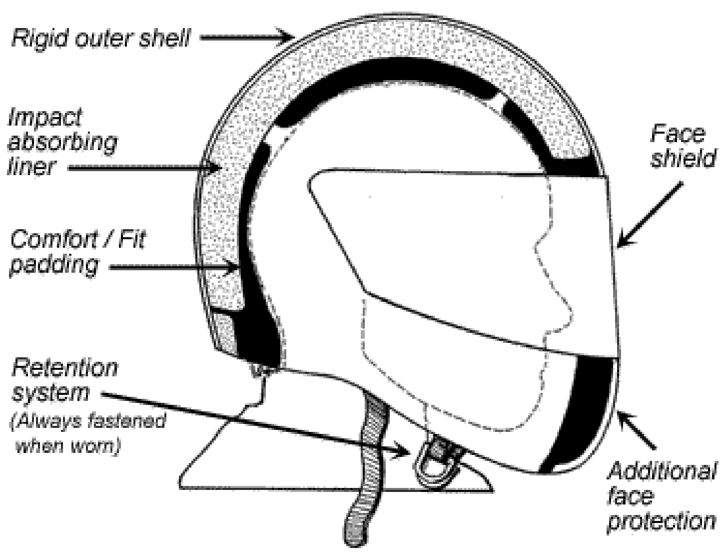


Fig. 2.1Components of the Helmet and Their Roles [3]

Typically, a helmet has the following components:

- **The Shell:** the shell of a helmet is an injection molded thermoplastic or a pressure molded thermo set that is reinforced with glass fibers or made of fiber glass.
  - **It absorbs energy in an impact:** the shell bends when the helmet is impacted and the underlying foam deforms. At moderate speeds the shell can take one-third of the impact energy.
  - **It distributes local forces from an impact:** rigid objects like stone or a projecting beam can cause a skull fracture at low forces, the shell acts to distribute the force of such impact eliminating the risk of penetration.
  - **It allows sliding on road surfaces:** the shell being rigid and having a convex shape allows the helmet to slide along a road surface without there being an excessive force.
  - **It protects the face and temples:** full-face helmet is beneficial in protecting the face and jaw. The chin bar of such helmets contain rigid foam to absorb energy for direct blows on the chin, prevent facial bone fractures and prevent the lower part of the forehead and temple being struck.

- **The Protective Padding:** this is a molding of polystyrene beads or polyurethane foam. It provides a stopping distance for the head. The foam can compress by 90% during an impact, although it recovers partially afterwards. But this helps increase the stopping distance thus reducing the peak deceleration of the head. It also protects as much as possible of the head. In addition to this there is a layer of comfort padding to provide comfort to the wearer.

- **Proper Strapping System:** It is essential to wear a well-fitting helmet for the effective working of chinstrap system. To test if the helmet fits your head properly, tightly fasten the chinstrap and then pull helmet off forward by gripping the rear and then pulling. The strap must be threaded correctly so that the buckle locks the strap when it is pulled from the chin side. The strap must be pulled as tight as is bearable under the chin.

2.3 STATE OF THE ART

A. Head/helmet mounted passive and active infrared imaging system with parallax

A passive/active infrared imaging system apparatus for mounting on a head/helmet includes a passive infrared camera Head Pack having a removable narrow band filter cover, an objective lens, an interface board, and a display unit such a liquid crystal display (LCD), with forward/back, up/down, and tilt adjustment functions fitting any mask, mounted in the front of said head/helmet for converting infrared light images into electronic signals. An electronic unit coupled between the UFPA of the infrared camera and the display unit, includes a controller for processing video signals from the infrared camera and supplying them to the display unit. The electronic circuit includes a wireless video & audio transceiver, a piezoelectric microphone, a voice controller, and a neural network pattern recognition chip.

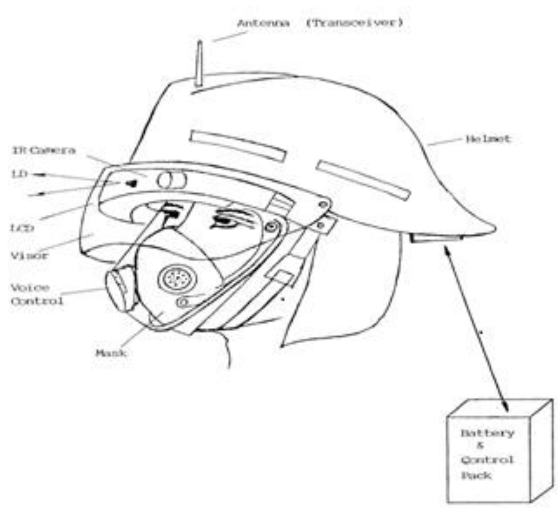


Fig 2.2 Infrared Helmet [5]

B. Skully Smart Helmet

The Helmet features heads up display which is transparent in nature giving us the display of rear mirror on gear shield screen. They do have taken care of situational awareness with help technology of GPS navigation and Bluetooth connectivity with mobile for communication the below could give you an idea about market creativity.

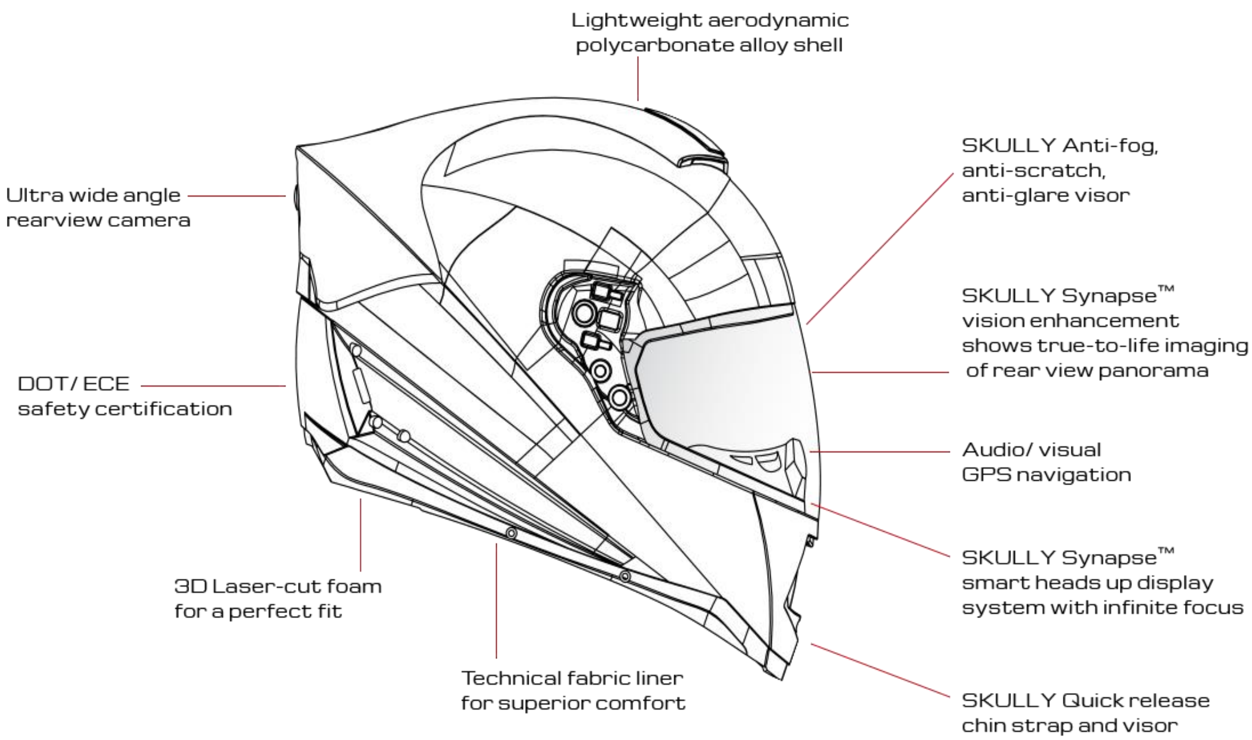


Fig 2.3 Skully Smart Helmet [4]

2.4 EXISTING METHODOLOGY

A. Traditional Concept

Generally, population in India is using helmet while driving their motor-cycle. But as due to lack in laws and justice department in India has not made the helmet compulsion in India Riding a motor-cycle in India is way too risky. Accident cases are also increasing due to it.

B. Smart Helmet

The Smart Helmet comes pre-wired and ready to go with built-in Bluetooth capabilities and controls, so users can enjoy all features of organization communication devices without the hassle of installing a headset. Chat with your buddies over the intercom system or take calls directly through your helmet with the built-in microphone, and hear your music louder and clearer built-in speakers to boost the audio level while you’re out on the open road.

Chapter 3
PROPOSED METHODOLOGY

3.1 AIM

The aim is to provide compulsion for rider to wear helmet while riding a motor-cycle. Further penalizing the law breakers by generating the fine receipt in collaboration with R.T.O. of India.

3.2 DESIGN GOALS AND OBJECTIVES

The proposed approach of a Smart Helmet for motorcyclists using Raspberry Pi and its modules which will ensure compulsory possession of a helmet with every motorcyclist. Scanning the object (helmet) using the Pi Camera module and verify if the helmet is present using image processing in Python. The model will also help R.T.O. by providing the information of rider who had challenged the system to trick by removing the helmet during its ride as system ensures periodic check on rider.

3.3 ADVANTAGES

1. The project is expected to improve safety and reduce accidents, especially fatal to the motorcyclist.
2. System with low cost and less complexity.
3. Reduce workload of Traffic Policemen.

Chapter 4
SYSTEM REQUIREMENTS

4.1 SOFTWARE REQUIREMENTS

1. Python

Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.

2. Linux (Raspbian OS)

Raspbian is the Foundation's official supported operating system. Raspbian comes pre-installed with plenty of software for education, programming and general use. It has Python, Scratch, Sonic Pi, Java, Mathematical capabilities and more.

3. OpenCV

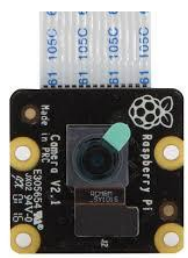
OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel, it was later supported by Willow Garage and is now maintained by Itseez. The library is cross-platform and free for use under the open-source BSD license. OpenCV is written in C++ and its primary interface is in C++, but it still retains a less comprehensive though extensive older C interface. Wrappers in other languages such as C#, Perl, Haskell and Ruby have been developed to encourage adoption by a wider audience.

4.2 HARDWARE REQUIREMENTS

1. Raspberry Pi 3

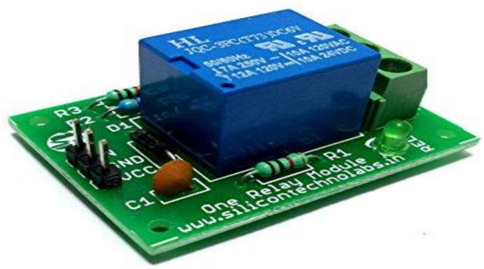
The Raspberry Pi 3 Model B features a quad-core 64-bit ARM Cortex A53 clocked at 1.2 GHz. This puts the Pi 3 roughly 50% faster than the Pi 2. Compared to the Pi 2, the RAM remains the same – 1GB of LPDDR2-900 SDRAM, and the graphics capabilities, provided by the Video Core IV GPU, are the same as they ever were. As the leaked FCC docs will tell you, the Pi 3 now includes on-board 802.11n WiFi and Bluetooth 4.0. WiFi, wireless keyboards, and wireless mice now work out of the box.

2. Raspberry Pi Camera Module



The camera module utilizes the dedicated CSi interface, which is located behind the Ethernet port on the Raspberry Pi .The Raspberry Pi Camera Module has a 8MP CMOS camera with a fixed focus lens that is capable of capturing still images as well as high definition video. Stills are captured at a full HD resolution while video is supported at 1080p at 30 FPS, 720p at 60 FPS and 640x480 at 60 or 90 FPS. . The camera is supported in the latest version of Raspbian, Raspberry Pi's preferred operating system.

3. Relay



A relay is an electrically operated switch. Many relays use a electromagnet to mechanically operate a switch. Relays are used where it is necessary to control a circuit by a low power signal, or where several circuits must be controlled by one signal. A type of relay that can handle the high power required to directly control an electric motor or other loads is called a contactor.

4. DC motor

A DC motor is any of a class of electrical machines that converts direct current electrical power into mechanical power. The most common types relay on the forces produced by magnetic fields. Nearly all types of DC motors have some internal mechanism, either electromechanical or electronic, to periodically change the direction of current flow in part of the motor. Most types produce rotary motion; a linear motor directly produces force and motion in a straight line.

4.3 SYSTEM ARCHITECTURE

Architectural Diagram is a graphical representation of the concepts, their principles, elements and components that are part of an architecture.

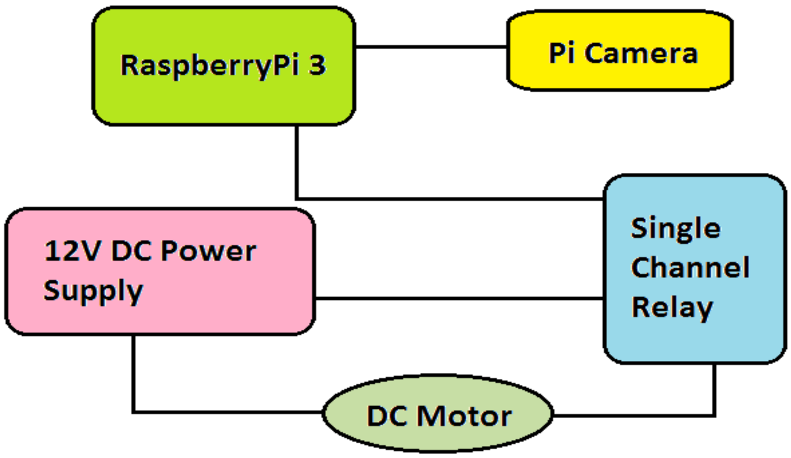


Fig. 4.1 System Architecture

The fig. 4.6 depicts the working of proposed methodology in brief. Raspberry Pi captures an image with the help of pi camera then process it to generate the result for detection of helmet .Resulting in giving access to rider to ride a motorcycle.

4.4 ESTIMATED COST

Table 4.1 - Total cost of project

Sr. No.	Product	Quantity*Price	Total Price
1.	Raspberry Pi 3	1*2800	2800
2.	Raspberry Pi NOIR camera	1*2200	2200
3.	Relay	1*100	100
		Total Cost	**5100**

Hence estimated total cost of the project would be around 5 thousand.

Chapter 5
SYSTEM DESIGN

5.1 DFD LEVEL-0 DIAGRAM

Level 0 DFD must balance with the context diagram it describes. Input going into a process is different from outputs leaving the process

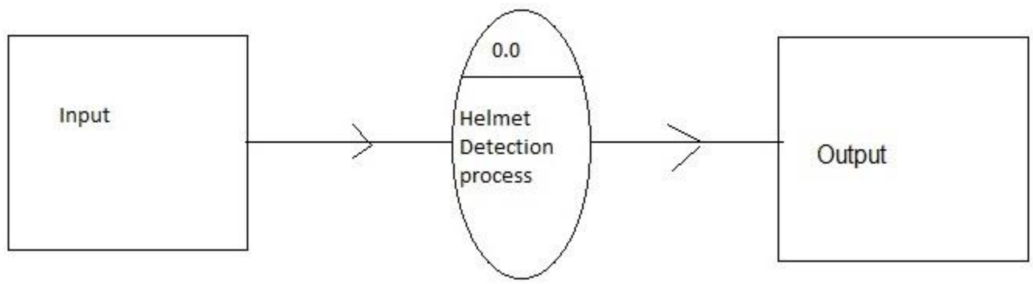


Fig. 5.1 DFD level-0 Diagram

The figure 5.1 describes, the DFD level 0 of Helmet detection system in which input is given to the system and required output is given by the system.

5.2 DFD LEVEL-1 DIAGRAM

Level 1 Data Flow Diagram. This highlights the main functions carried out by the system. As a rule, we try to describe the system using between two and seven functions - two being a simple system and seven being a complicated system. This enables us to keep the model manageable on screen or paper.

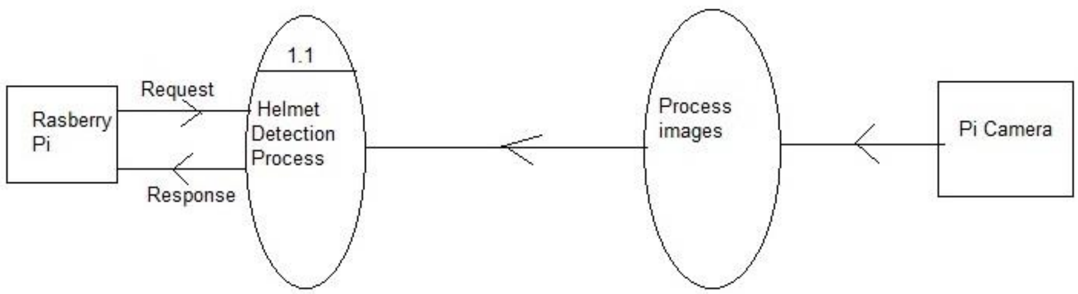


Fig. 5.2 DFD level-1 Diagram

The figure 5.2 describes, the DFD level 1 of Helmet detection system is a request and response type of process between the main system.

5.3 CLASS DIAGRAM

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods).

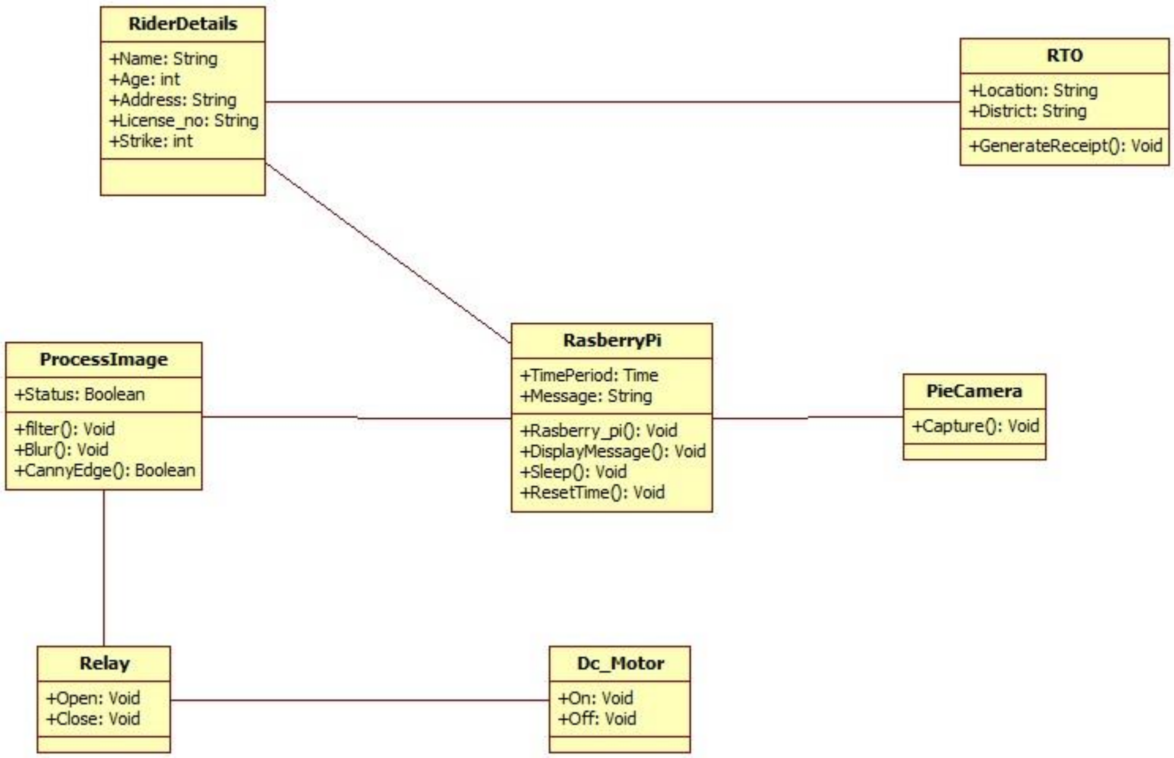


Fig. 5.3 Class Diagram

In fig. 5.3, several system classes have been introduced.

5.4 ADVANCED CLASS DIAGRAM

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

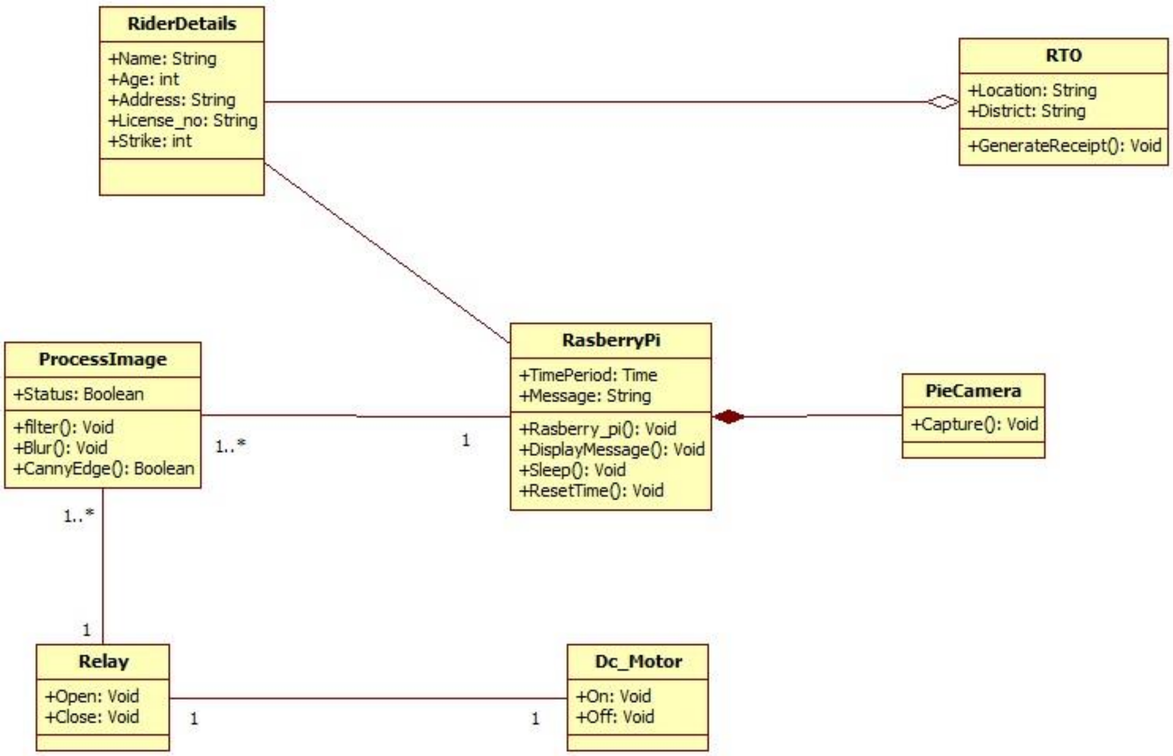


Fig.5.4 Advanced Class Diagram

In the fig 5.4, hardware devices such as Raspberry Pi, Pi camera, DC motor, etc. is connected to the vehicle where in Raspberry Pi has all information about its components.

5.5 USE CASE DIAGRAM

A use case is a list of actions or event steps, typically defining the interactions between an actor and a system, to achieve a goal. The actor can be a human or other external system.

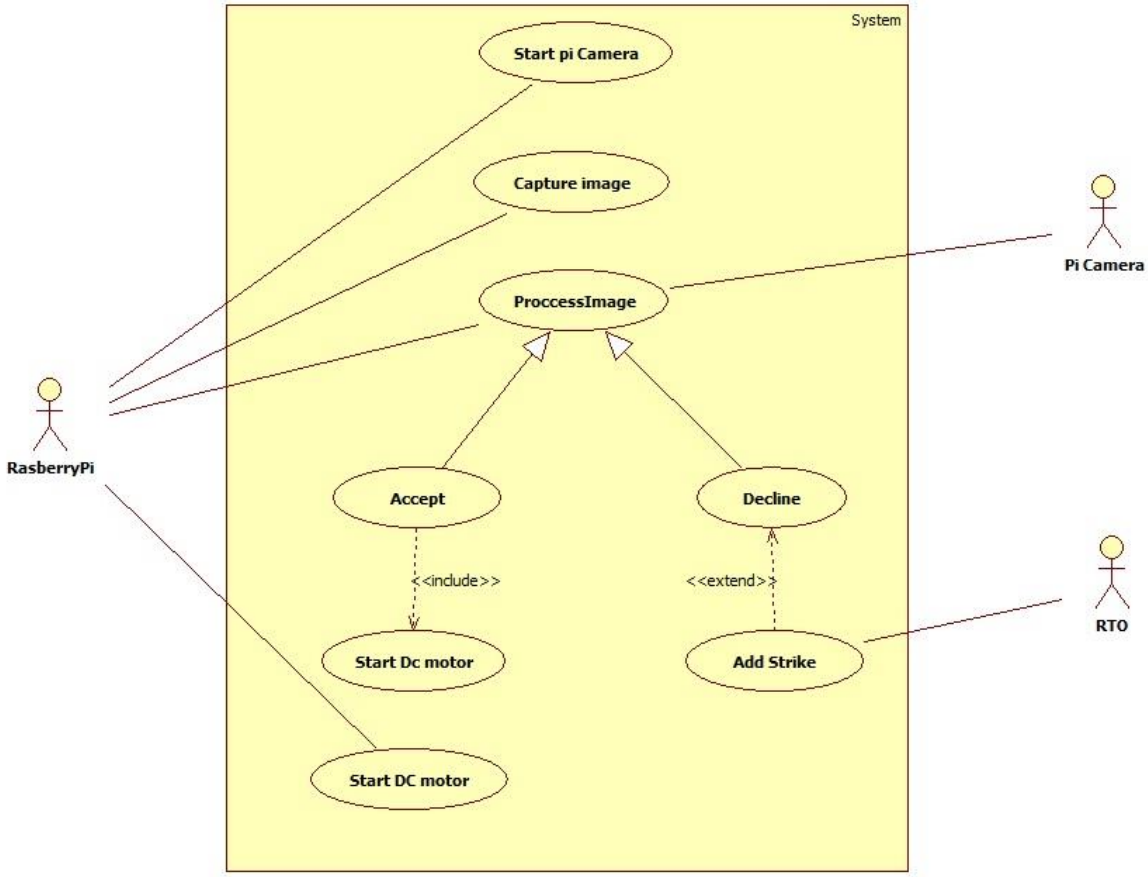


Fig. 5.5 Use-Case Diagram

The fig. 5.5 shows the various activities that a user and system can perform. The Raspberry pi can do various activities such as starting Pi camera, capturing mages, processing image and starting DC motor. As soon as the Raspberry Pi detects the helmet it closes relay to give access to rider otherwise direct strikes to RTO.

5.6 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

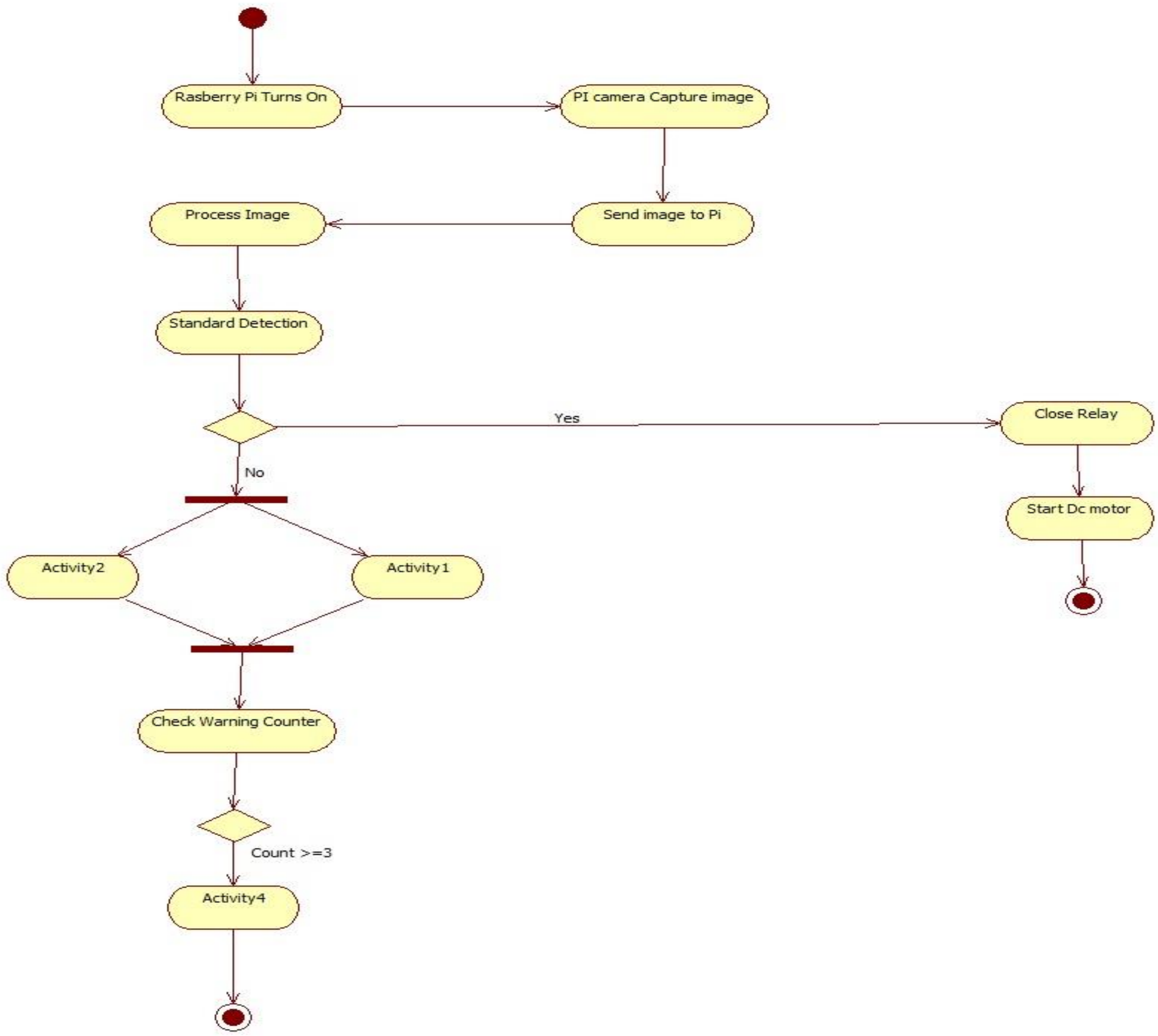


Fig. 5.6 Activity Diagram

The fig 5.6 shows the various activities involved in the system. Raspberry Pi camera first captures the image then process image to verify the detection of helmet. In case of failure it does give a warning message and also set the timer. After three such strikes of failure it does inform RTO to generate the defaulters receipt. Success in detection gives access to ride the vehicle.

5.7 SEQUENCE DIAGRAM

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

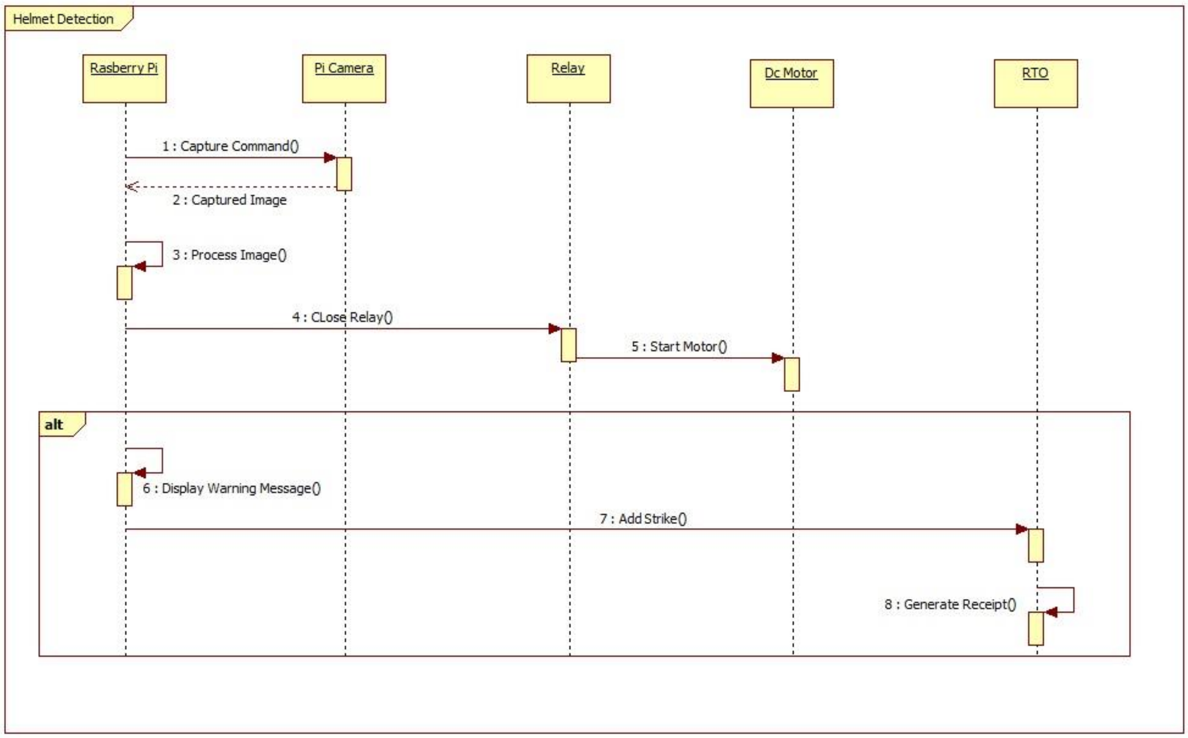


Fig. 5.7 Sequence Diagram

The fig. 5.7 shows the interactions between different objects arranged in a time sequence. The Pi camera initially captures the image or video and then process the image. If the detection of helmet occurs it does give signal to the relay to be close so that motor starts. In case of failure in detection of helmet it does display the warning message to the user also it does add strikes. Three such strikes of the user may result him to pay the fine as RTO generates a receipt.

5.8 COMPONENT DIAGRAM

Component diagrams are used to visualize the organization and relationships among components in a system

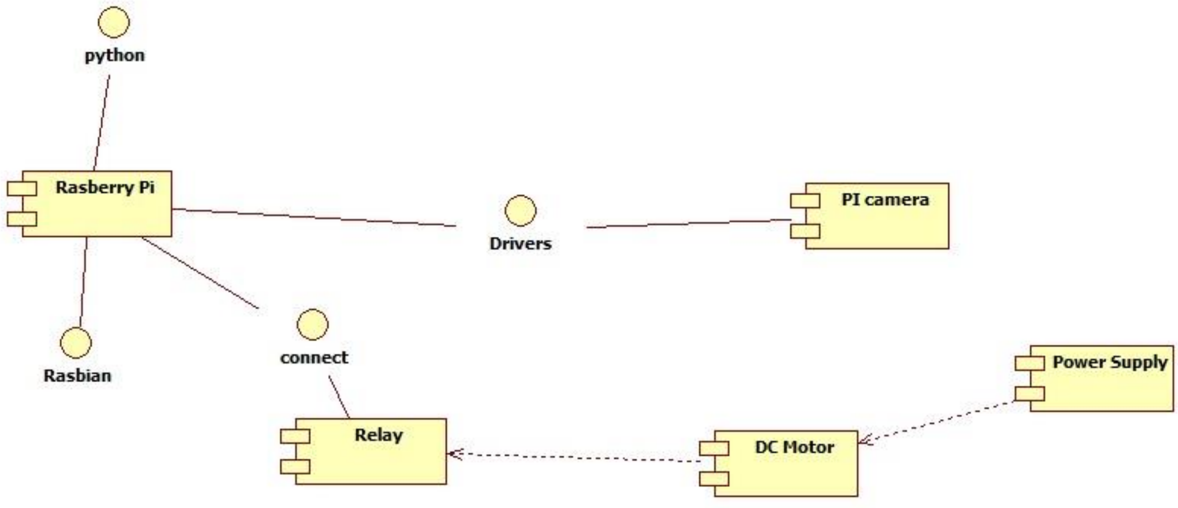


Fig. 5.8 Component Diagram

The figure 5.8 describes the different components like Raspberry component, Pi Camera component, Power Supply component, Dc motor component, Relay component. Raspberry component is connected to Python and Raspbian interface which is a required interface. Power Supply component is connected to Dc motor which is connected to Relay component.

5.9 DEPLOYMENT DIAGRAM

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.

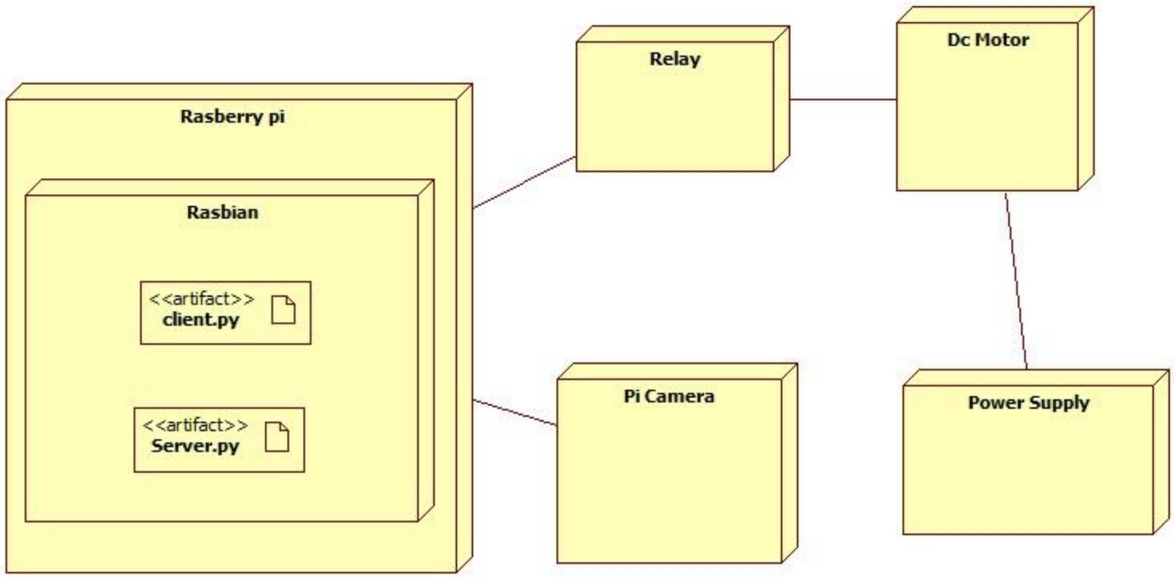


Fig. 5.9 Deployment Diagram

The figure 5.9 Consist of six nodes namely Raspberry pi, Relay, Dc motor, Pi Camera, Power Supply, Raspbian. Node Raspbian consists of two artifacts client.py, server.py. Artifacts are the executable files of the system.

.

Chapter 6
METHODOLOGY

6.1 ALGORITHMS/TECHNIQUES USED FOR IMPLEMENTATION

Using boosted cascades of Haar-like features, which was Proposed by [Viola, Jones 2001]. Implementation available in OpenCV

- Haar like feature

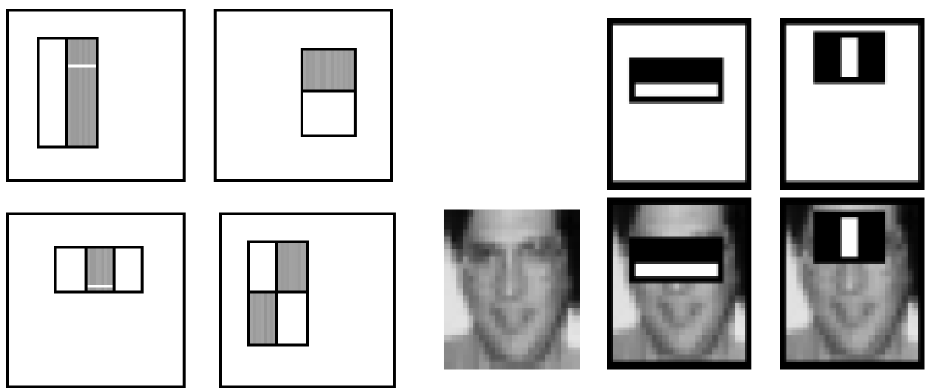


Fig. 6.1Haar like feature [8]

1) Feature = w1 x RecSum(r1) + w2 x RecSum(r2)
2) Weights can be positive or negative
3) Weights are directly proportional to the area
4) Calculated at every point and scale

- Attentional Cascade

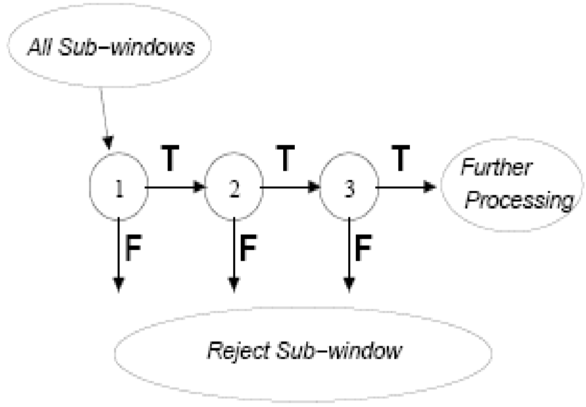


Fig. 6.2 Attentional Cascade [8]

1) Initial stages have less features (faster computation)
2) More time spent on evaluating more promising sub-windows

- Cascade Creation - Walkthrough

Input:

– f = Maximum acceptable false positive rate per layer (0.5)

– d = Minimum acceptable detection rate per layer (0.995)

– Ftarget = Target overall false positive rate

• Or maximum number of stages in the cascade

• For nStages = 14, Ftarget = f nStages = 6.1 e-5

– P = Set of positive examples

• 200 distorted versions of a synthetic image

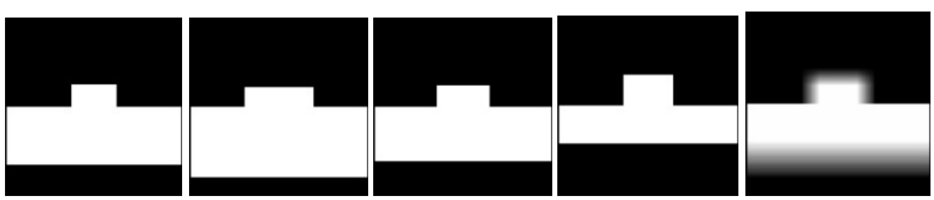


Fig. 6.3 Synthetic Image [8]

–N = Set of negative examples

• 100 images from BACKGROUND_Google category of Caltech 101 dataset

Fig. 6.4 Background Negatives [8]

- Algorithm for Creation of Cascade

```
F0 = 1
i = 0
while Fi >Ftarget and i<nStages
    i = i + 1
    Train Classifier for stage i
        Initialize Weights
        Normalize Weights
        Pick the (next) best weak classifier
        Update Weights
        Evaluate fi
        if fi > f
        go back to Normalize Weights
        Combine weak classifiers to form the strong stage classifier
        Evaluate Fi
```

6.2 TEST STRATEGY

Here,

i. Fi = False alarm rate of the cascade with i stages

ii. Initialize Weights

Weight for each
positive sample 0.5/m
negative sample 0.5/n

m – number of positive samples (200)
n – number of negative samples (100)

iii. Normalize Weights and Picking out best weak classifier

The one with minimum error

$$\in_t = \min_{f,p,\theta} \sum_t w_t |h(x_t, f, p, \theta) - y_t|$$

weight normalization formula [8]

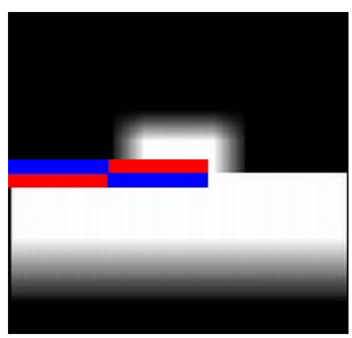


Fig. 6.5 Minimum Error [8]

$\in_t = 0.005$

iv. Error Minimization Process

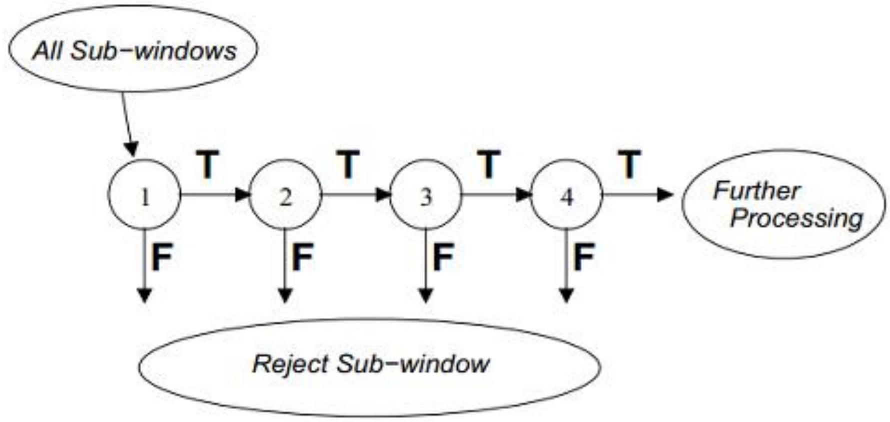


Fig. 6.6 Error minimization flowchart [8]

Table 6.1 - Error minimization flowchart parameters [8]

F T+: Total sum of weights of positive examples	e1 = S+ + (T-- S-)
i g T-: Total sum of weights of negative examples	e2 = S- + (T+ - S+)
. S+: Total sum of weights of positive examples below the current one	
4 S-: Total sum of weights of negative examples below the current one .	**e = min(e1, e2)**

v. Update Weights

$$W_{t+,l} = W_{t,l}\beta_t^{1-e_t}$$

weight update formula [8]

ei = 0, if example xi is classified correctly
ei = 1 , otherwise

$$\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$$

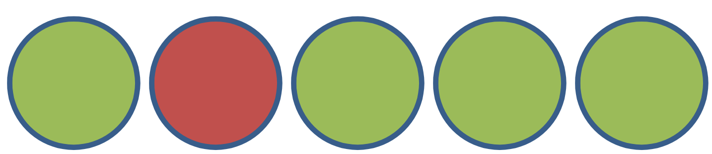


Fig. 6.7 Weight Updating [8]

vi. Evaluate fi

fi= number of negative samples that were detected by this stage/ total number of negative samples
= 1/100

vii. Combine weak classifiers to form the strong stage classifier

$$C(x) = \begin{cases} 1, \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0, otherwise \end{cases}$$

$$\beta_t = \frac{\epsilon_t}{1 - \epsilon_t} \alpha_t = \log \frac{1}{\beta_t}$$

Weight is inversely proportional to the training error [8]

1. **Paper**
   Decrease threshold until the classifier has a detection rate of at least d
2. **OpenCV**
   a) For each positive sample, find the weighted sum of all features
   b) Sort these values
   c) Set threshold = sorted_values[(1-d) * |P|]

6.3 RESULT AND ANALYSIS

- Resulting Cascade

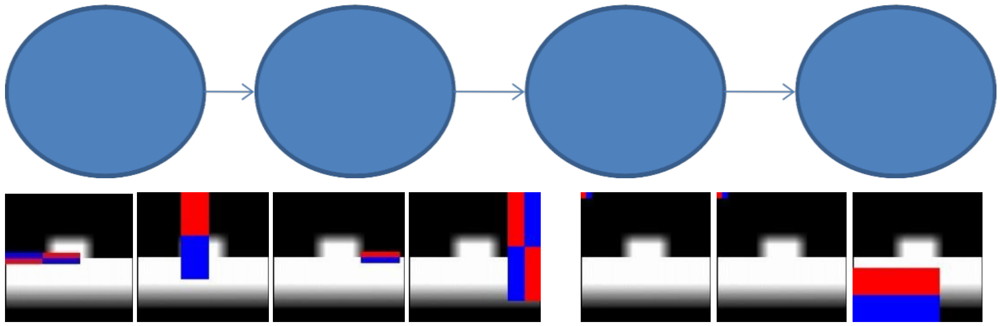


Fig. 6.8 Result Cascade Stages [8]

If (maximum false alarm rate) is increased from 0.5 to 0.7, a cascade with only the first two stages is created

6.3.1 PYTHON SCRIPTS

Helmet Detection Script

```
import sys
sys.path.append('/usr/local/lib/python3.4/site-packages')

frompicamera.array import PiRGBArray
frompicamera import PiCamera
import time
import cv2
importargparse

# initialize the camera and grab a reference to the raw camera capture
camera = PiCamera()
camera.resolution = (640, 480)
camera.framerate = 32
rawCapture = PiRGBArray(camera, size=(640, 480))

# allow the camera to warmup
time.sleep(0.1)

# capture frames from the camera
for frame in camera.capture_continuous(rawCapture, format="bgr", use_video_port=True):
        # grab the raw NumPy array representing the image - this array
        # will be 3D, representing the width, height, and # of channels
        image = frame.array
        gray = cv2.cvtColor(image,cv2.COLOR_BGR2GRAY)
        faceCascade=
cv2.CascadeClassifier('/home/pi/project_workspace/New_helmet_haar.xml')
```

```
    faceRects=faceCascade.detectMultiScale(gray,scaleFactor=1.2,minNeighbors=5,minSiz
e=(30,30))
    for (fX,fY,fW,fH) in faceRects:
       cv2.rectangle(image,(fX,fY),(fX+fW,fY+fH),(0,255,0),2)
    # show the frame
    cv2.imshow("Frame",image)
    key = cv2.waitKey(1) & 0xFF

    # clear the stream in preparation for the next frame
    rawCapture.truncate(0)

    # if the `q` key was pressed, break from the loop
    if key == ord("q"):
            break
```

6.3.2 SCREENSHOTS

After running the helmet detection Python script, a new frame buffers a video which highlights the helmet with a rectangle.

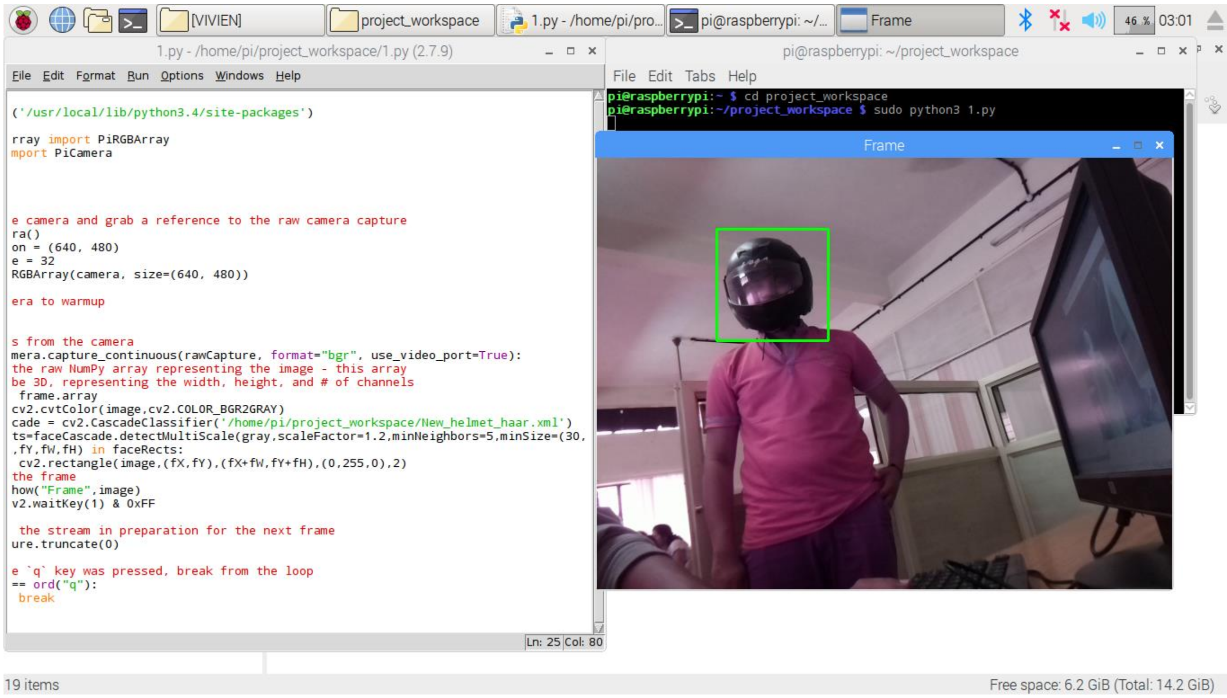


Fig. 6.9 Detected Object Frame

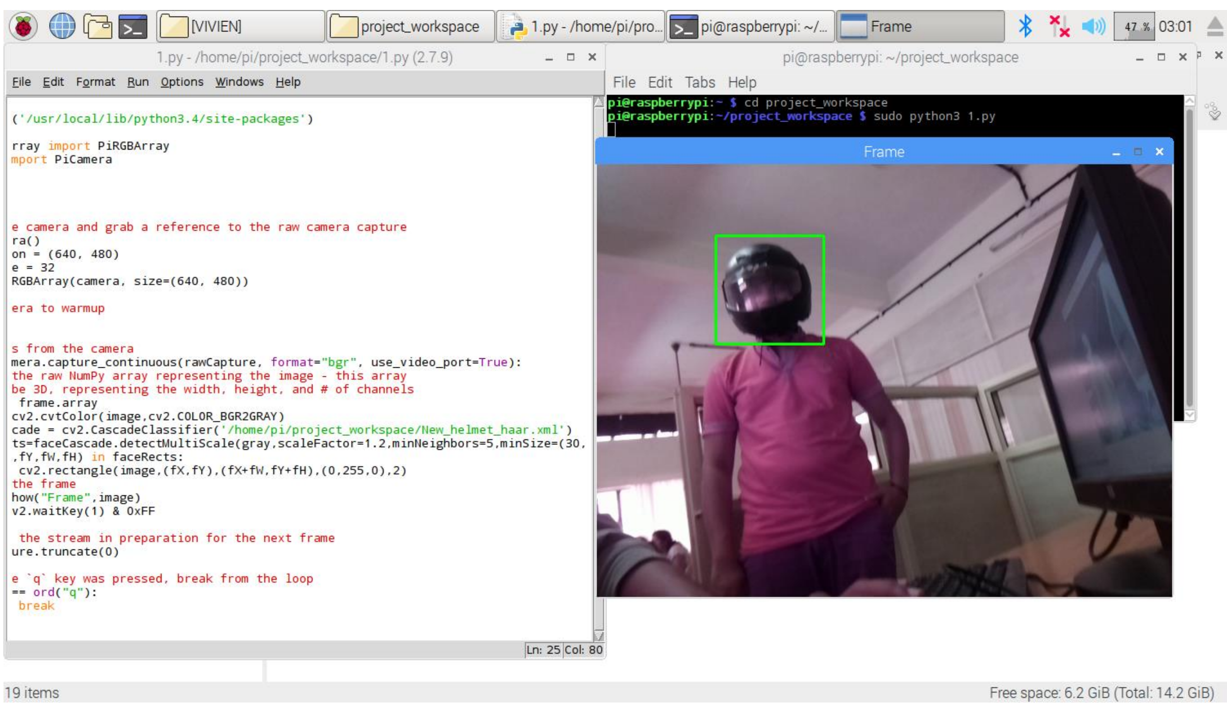


Fig. 6.10 Detected Object Frame 2

The Haar cascade training works in 15 stages, more stages makes the object detection stages more accurate. This picture depicts the stage 1 of the cascade creation process.

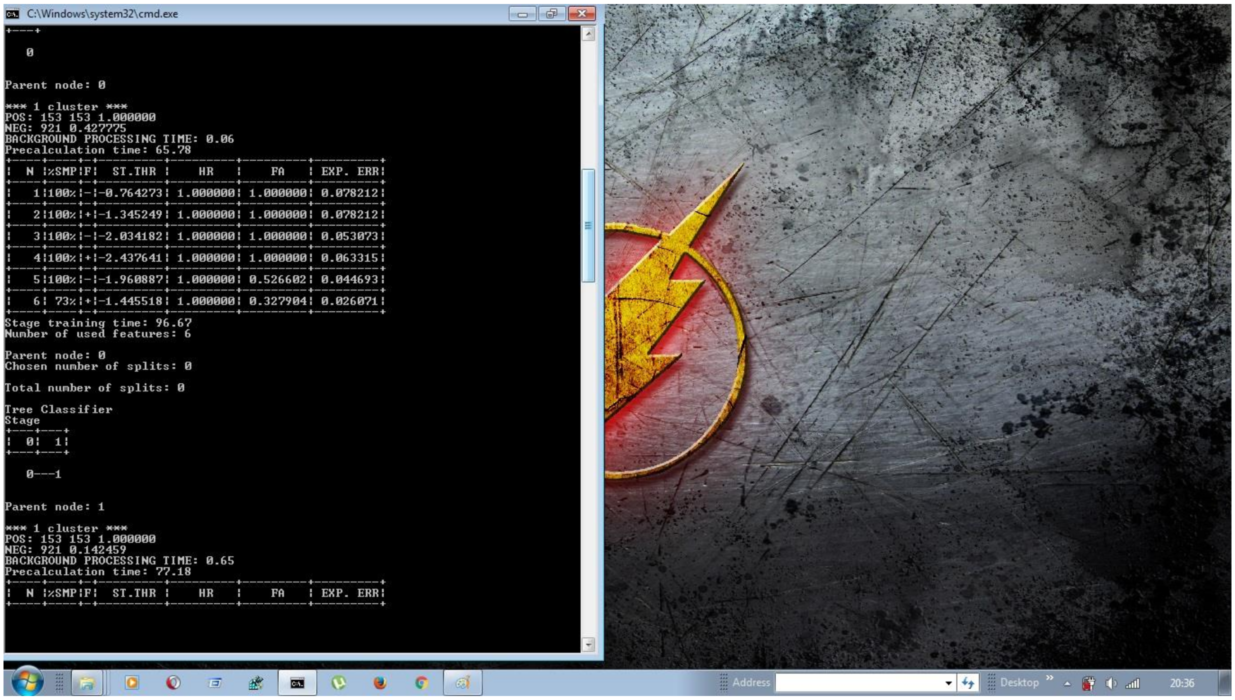


Fig. 6.11 Haar Cascade training stage 1

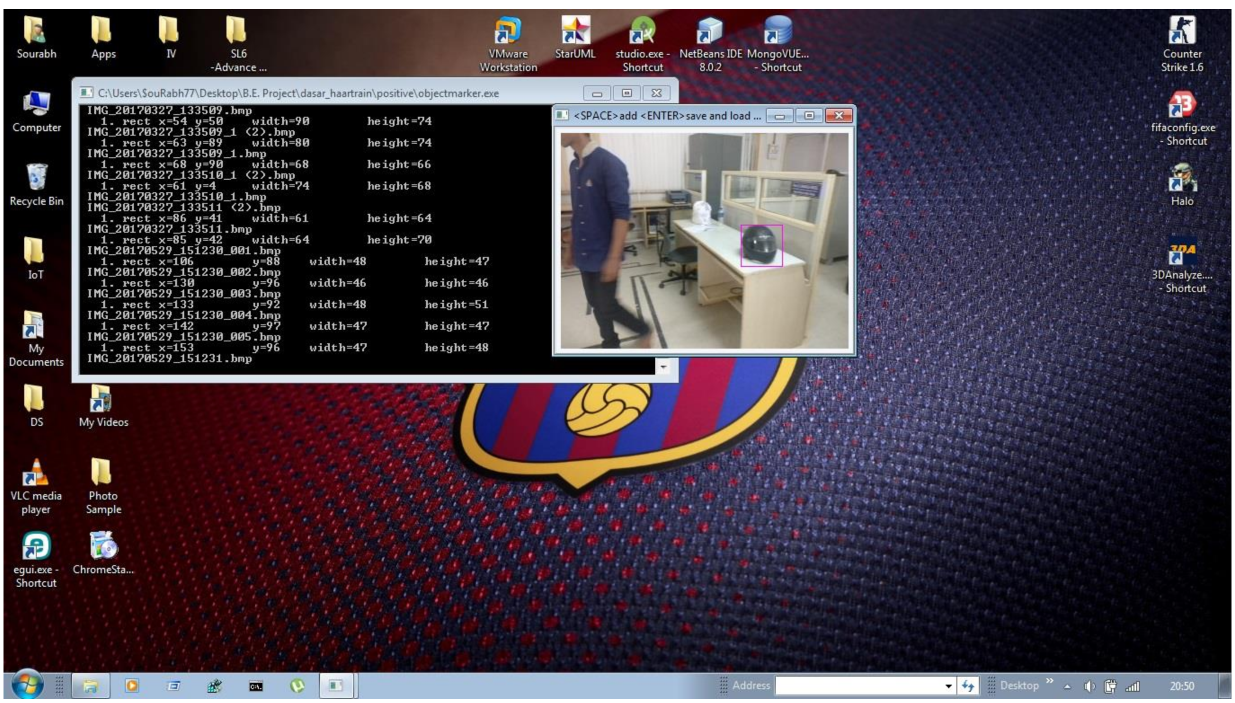


Fig. 6.12 Adding object parameters to cascade 1

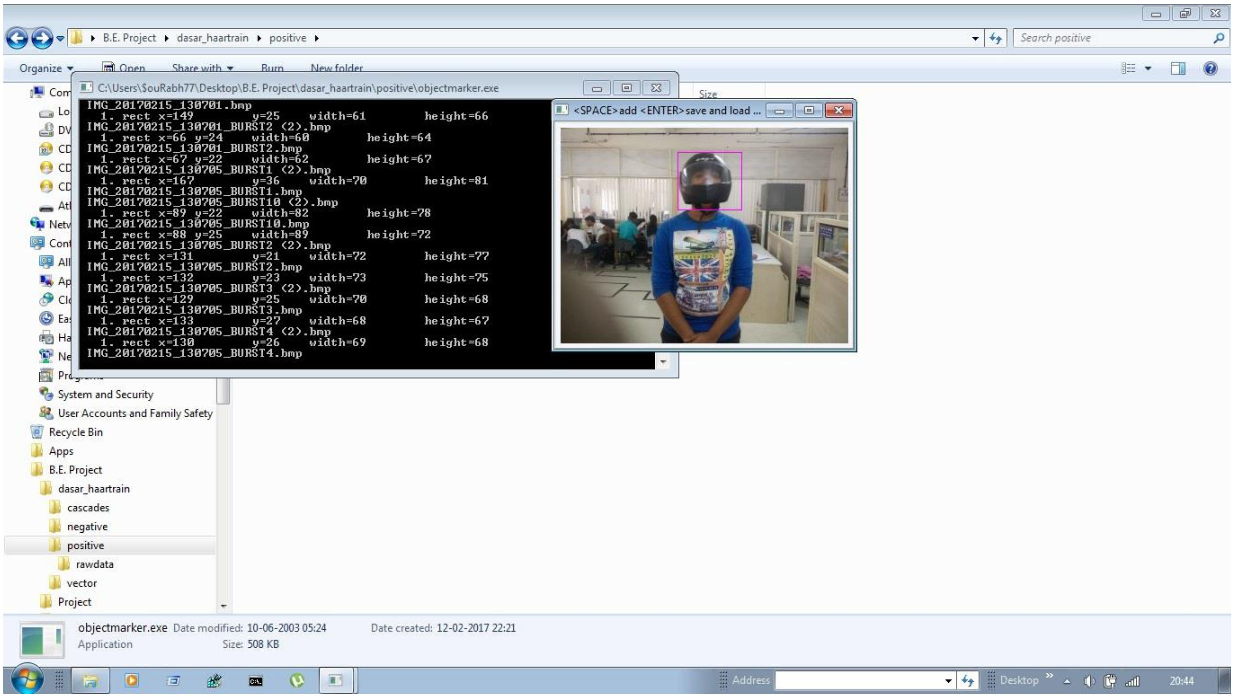


Fig. 6.13 Adding object parameters to cascade 2

Chapter 7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

The scope of this project is a motorcycle rider that they care about their safety while riding. As we know, the motorcycle riders are now less concerned about their safety while riding, then the creation of this helmet safety rates can be increased and rate of road accidents can be reduced. The accident rates for motorcyclists are increasing from year to year, a Smart Helmet for Motorcyclist using Raspberry Pi which in future will inspire safety features for motorcyclists.

7.2 FUTURE SCOPE

In future work the system will be enable feature like key elimination as the helmet itself would become a key when it comes in contact with vehicle. Also with use of pneumatic sensors we would make pressure on visor (chin strap).

7.3 DEVELOPER'S COMMENTS

This project focuses on rider safety by ensuring the possession of a helmet by the motorcyclist at all times. Open Computer Vision (Open CV) libraries provide the necessary services for the image processing tasks and enable the system to detect the helmet. This prototype can be established on any vehicle as it takes a minimal amount of space.

REFERENCES

[1] PG Hartwell, JA Brug - US Patent 6,798,392, 2004 - Google Patents

[2] Zhang Y., Lu Y., Wu H., Wen C., Ge C. (2016) Face Occlusion Detection Using Cascaded Convolutional Neural Network. In: You Z. et al. (eds) Biometric Recognition. CCBR 2016. Lecture Notes in Computer Science, vol 9967. Springer, Cham

[3] KC Hobby, B Gowing, DP Matt - US Patent App. 13/658,793, 2012 - Google Patents

[4] RattapoomWaranusast, NannaphatBundon, VasanTimtong and ChainaronTangnoi,"Machine Vision Techniques for Motorcycle Safety Helmet Detection," 2013,28th International Conference on Image and Vision Computing New Zealand

[5] Manjesh N, Prof. SudarshanRaj,"Smart Helmet Using GSM &GPS Technology for Accident Detection and Reporting System," International Journal of Electrical and Electronics Research ISSN 2348-6988 (online) Vol. 2, Issue 4

[6] Ashvini M., Revathi G., Yogameena B., Saravanaperumaal S. (2017) View Invariant Motorcycle Detection for Helmet Wear Analysis in Intelligent Traffic Surveillance. In: Raman B., Kumar S., Roy P., Sen D. (eds) Proceedings of International Conference on Computer Vision and Image Processing. Advances in Intelligent Systems and Computing, vol 460. Springer, Singapore

[7] Mohd Khairul, Afiq Mohd Rasli, Nina Korlina Madzhi, Juliana Johari, "Smart Helmet Sensors for Accident Prevention," 2013 International Conference on Electrical Electronics and System Engineering.

[8] Paul Viola Michael J. Jones ,"Robust Real-Time Face Detection", International Journal of Computer Vision ,Volume 57 Issue 2 May 2004 ,United States.