ACKNOWLEDGEMENT

We take this opportunity to express our heartfelt gratitude and appreciation to all those who provided us the support and encouragement to complete this project. Without their contributions, inputs and suggestions, we would not have succeeded in developing the idea and completing the project. We record our indebtedness to NMAM Institute of Technology for giving us a platform to learn and also initiate our project.

Our heartfelt thanks to our esteemed guide and mentor, Ms. Swathi Pai M., Assistant Professor, Department of CSE, for her valuable advice, endless support and motivation, constantly throughout.

We would like to thank Dr. Niranjan Chiplunkar, Principal, NMAMIT and Dr. Udaya Kumar Reddy, Head, Department of CSE for their consistent support and providing us this opportunity to do the project.

We would like to thank our college, NMAMIT for providing us with facilities such as infrastructure, high performance computers and laboratories for carrying out our work.

We would also like to thank all the teaching and non-teaching staffs of Department of CSE whose support motivated us to complete the project. A heartfelt thanks to our parents and our families for their undeviating solace. We express our deep sense of gratitude to all our batch mates who have invariably contributed to our project with their inputs and suggestions. Our unflinching gratitude to everyone who has directly or indirectly contributed to the project.

Rithika Chowta (4NM14CS135)
Priya Shetty (4NM14CS117)
Sharan Preetha Noronha (4NM14CS146)

ABSTRACT

Blind curves are one of the leading causes of road accidents. Vehicles speeding along a curve are not aware of the presence of vehicles coming from the other direction. Here, a system is proposed to alert drivers going around a blind curve to the presence of oncoming vehicles. 2 poles are erected on either side of the curve, bearing cameras, red and green LED lights and piezoelectric buzzers. They are connected to a Raspberry Pi. The live video feed from the cameras is processed to detect the presence of vehicles. If vehicles are approaching on both sides of the curve, the buzzers and red lights are activated, thus alerting the drivers of the vehicle to slow down. Then the green LED is activated on one side to allow one vehicle to move forward. After it passes, the other vehicle is allowed to move by activating the green LED on the other side. Red LED is reactivated on the previous side, to stop any vehicles that were behind the first vehicle. After all vehicles pass, all LEDs are deactivated.

CONTENTS

Title page i

Certificate ii

Acknowledgement iii

Abstract iv

Table of Contents v

List of Figures viii

CHAPTER 1 - INTRODUCTION 1

1.1 Overview 2

1.2 Problem Statement 2

1.3 Study Areas 2

1.4 Objective 3

1.5 Methodology 3

1.6 Organization of the report 4

CHAPTER 2 - LITERATURE SURVEY 5

2.1 Existing System 5

2.2. Proposed System 7

CHAPTER 3 - SYSTEM ANALYSIS AND REQUIREMENTS 9

3.1 System Analysis 9

3.1.1 Relevance of platform 9

3.1.2 Relevance of programming language 9

3.2 Requirements Analysis 10

3.2.1 Scope and Boundary 10

3.2.2 Assumptions and Dependencies 10

3.3 Functional Requirements 10

3.3.1 Software Requirements 10

3.3.2 Hardware Requirements 10

CHAPTER 4 - HARDWARE APPROACH 15

4.1 Physical Setup of Raspberry Pi 15

4.2 GPIO Alarm Sub-System Circuitry 16

CHAPTER 5 - SOFTWARE APPROACH 19

5.1 OpenCV 19

5.2 Cascade of Boosted Classifiers 19

5.3 LBP Features 21

CHAPTER 6 - SYSTEM DESIGN 24

CHAPTER 7 - IMPLEMENTATION 26

7.1 Training of Cascade Classifier 26

7.2 Detecting the Vehicles 36

7.3 GPIO Alarm Module 38

CHAPTER 8 - RESULTS AND DISCUSSION 44

8.1 Results 44

8.2 Discussion 49

CHAPTER 9 - CONCLUSION AND FUTURE WORK 51

9.1 Conclusion 51

9.2 Future work 51

REFERENCES 52

LIST OF FIGURES

Figure 4.1 Connection setup of Pi 15

Figure 4.2 Pin diagram of Raspberry Pi, with USB ports facing downwards 16

Figure 4.3 Circuit diagram for GPIO alarm subsystem 17

Figure 4.4 Breadboard 1 18

Figure 4.5 Breadboard 2 18

Figure 5.1 Thresholding of 8-pixel neighborhood around center pixel 21

Figure 5.2 Converting binary neighborhood of center pixel into decimal form 22

Figure 5.3 The calculated LBP value is then stored in an output array with the same width and height as the original image 22

Figure 5.4 LBP representation of an image 23

Figure 6.1 Block diagram of the system 24

Figure 6.2 Activity diagram of the system 25

Figure 7.1 “neg” folder with all negative images 28

Figure 7.2 bg.txt 29

Figure 7.3 Cropping a positive image 30

Figure 7.4 Positive images 30

Figure 7.5 Sample creation command 31

Figure 7.6 Samples and annotation file generated 32

Figure 7.7 info.lst contents 32

Figure 7.8 Creating vector file from samples in info1 34

Figure 7.9 Training began 36

Figure 8.1 Output from the monitor when both the cars got detected 44

Figure 8.2 Both the red LEDs switched on upon detection 45

Figure 8.3 Both the vehicles stop when both red LEDs are on 46

Figure 8.4 Right side green LED on and left red LED on 47

Figure 8.5 Left green LED is on and right red LED is on 48

Figure 8.6 LEDs on the both of the sides of the curve are off 49

Figure 8.7 LBP features detected for test image in stage 2 of cascade classifier 50

Figure 8.8 LBP features detected for test image in stage 9 of cascade classifier 50