VISION-BASED REAL-TIME TRAFFIC CONTROL USING ARTIFICIAL NEURAL NETWORK ON GENERAL-PURPOSE EMBEDDED HARDWARE

Hithanadura Kavindu Gimhan Zoysa

(188470C)

Dissertation submitted in partial fulfillment of the requirements for the degree Master of Science in Electronic and Automation

Department of Electronic and Telecommunication Engineering

University of Moratuwa

Sri Lanka

August 2020

DECLARATION

I declare that this is my own work and this dissertation does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: Date:

The above candidate has carried out research for the Masters Dissertation under my supervision.

Name of the supervisor: Prof. Rohan Munasinghe

Signature of the Supervisor: Date:

Abstract

In urban cities, traffic management of intersections is a substantially challenging problem. In appropriate traffic control leads to waste of fuel, time, and productivity of nations. Though the traffic signals are used to control traffic, it often causes problems due to the pre-programmed timing being not appropriate for the actual traffic intensity at the intersection. Traffic intensity determination based on statistical methods only gives the average intensities expected at any given time. However, to control traffic effectively, the knowledge of real-time traffic intensity is a must-have. In this project, vision-based technology and artificial intelligence (AI) are used to estimate traffic in real-time and control the traffic in order to reduce the traffic congestion. General -purpose electronic hardware has been used for in-situ image processing based on edge-detection methods. A Neural Network (NN) was trained to infer traffic intensity in each image in real-time using a scale of 1(very low) to 5 (very high). A Trained AI unit, which takes approximately 4 seconds to process each image and estimate traffic intensity was tested on the road where it recorded a 90% acceptance rate. In order to control the traffic, a ratio-based method and a reinforcement learning (RL)-based method was used. The performance of these methods are compared with a pre-programmed traffic controller.

***Keywords*-traffic sensing, traffic control, neural network, reinforcement learning**

ACKNOWLEDGMENTS

First, I wish to express my sincere appreciation to my supervisor, Professor Rohan Munasinghe. I was privileged to work under his guidance. I thank him for his support, encouragement and patience made my success reality. Without his persistent help, the goal of this project would not have been realized.

Then, I would like to extend my gratitude to Doctor Chamira Edussooriya for his guidance and the support I had over the past two years as the coordinator of my MSc program. Also, my appreciation goes to the staff members and all of my friends who participated in the MSc program in the Department of Electronic and Telecommunication Engineering for their invaluable support.

This work was supported by World Bank AHEAD ITS project and Innovation Quotient Pvt Ltd.

TABLE OF CONTENTS

Declaration i

Abstract ii

Acknowledgments iii

Table of Contents vi

List of Figures viii

List of Tables ix

List of abbreviations x

List of Appendices xi

1 INTRODUCTION 1

2 DESIGN OF TRAFFIC INTENSITY ESTIMATION 5

2.1 Overall design 5

2.2 Edge Detection for traffic intensity estimation 7

2.3 Estimation of traffic using a neural network 13

3 TEST TRAFFIC INTENSITY ESTIMATION METHOD 20

3.1 Traffic intensity Estimation using sidebar images 20

3.2 Traffic intensity estimation using crossbar images 20

4 DESIGN OF TRAFFIC CONTROLLER 27

4.1 Simulator 27

4.2 Evaluators 28

4.3 Basic Configurations 28

4.4 Real world configurations 30

4.5 Ratio-based traffic controller 31

4.5.1 Methodology 31

4.5.2 Results 31

4.5.3 Discussion 31

4.6 RL-based traffic controller 32

4.6.1 Methodology 32

4.6.2 Results 33

4.6.3 Discussion 33

5 CONCLUSIONS AND RECOMMENDATIONS 37

Bibliography **40**

LIST OF FIGURES

2.1 Overall design of traffic intensity estimation 6

2.2 Sidebar image and crossbar image 7

2.3 Three zones of the sidebar image and crossbar image 8

2.4 Edges count 9

2.5 Lane marks that are detected as edges 10

2.6 Zero traffic image 11

2.7 An image after removing erroneous edges 12

2.8 Flow of detecting and counting the edges 12

2.9 Example images to explain the annotation criteria 14

2.10 NN used for real-time traffic intensity estimation 14

2.11 ReLU activation function 16

2.12 Flow of training the NN 17

3.1 Traffic intensity estimation using sidebar images 21

3.2 Traffic intensity estimation using crossbar images 24

4.1 The two way isolated intersection 29

4.2 Map of the junction used to obtain phase timing 30

4.3 Real world phases 30

4.4 Performance of ratio-based controller 34

4.5 Flow of the RL-based controller 35

4.6 Performance of RL-based controller 36

LIST OF TABLES

2.1 Criterion to Remove Erroneous Pixels 8

2.2 Traffic intensity of sidebar image annotated by an expert 18

2.3 Traffic intensity of crossbar image annotated by an expert 19

3.1 Accuracy of the sidebar estimation 21

3.2 Compare expected values and estimated values for sidebar images 22

3.3 Accuracy of the crossbar estimation 24

3.4 Compare expected values and estimated values for crossbar imaeges 25

LIST OF ABBREVIATIONS

Abbreviation	Description
AI	Artificial Intelligence
NN	Neural Network
RL	Reinforcement Learning
GPS	Global Positioning System
YOLO	you-only-look-once
ReLU	Rectified Linear Unit
SUMO	Simulation of Urban Mobility
TraCI	Traffic Control Interface

LIST OF APPENDICES

Appendix	Description	Page
RELATED CODES	Python codes related to performance	40