

ACCURATE LOCATION DETECTION SYSTEM FOR AUGMENTED REALITY APPLICATION

Dewagirige Chamila Kanchana Rathnayake

(168262X)

M.Sc. in Computer Science

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

March 2020

ACCURATE LOCATION DETECTION SYSTEM FOR AUGMENTED REALITY APPLICATION

Dewagirige Chamila Kanchana Rathnayake

(168262X)

M.Sc. in Computer Science

This dissertation submitted in partial fulfillment of the requirements for the Degree of MSc
in Computer Science specializing in Software Architecture

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

March 2020

DECLARATION

I declare that this is my own work and this dissertation does not incorporate without acknowledgement any material previously submitted for 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:

Name: Dewagirige Chamila Kanchana Rathnayake.

The supervisor/s should certify the thesis/dissertation with the following declaration.

I certify that the declaration above by the candidate is true to the best of my knowledge and that this report is acceptable for evaluation for the draft MSc Thesis.

Signature of the supervisor:

Date:

Name: Dr. Indika perera

ACKNOWLEDGEMENTS

I would like to express gratitude to my supervisor, Dr. Indika Perera, for his invaluable support by providing relevant knowledge, materials, advice, supervision and useful suggestions throughout this research work. His expertise and continuous guidance enabled me to complete my work successfully. Further I would like to thank Amodth Jayawardena for providing valuable resources and advise in the area of this research.

Finally, I would like to thank the academic and non-academic staff of Department of Computer Science and Engineering of the University of Moratuwa.

ABSTRACT

The proposed research has been done as an extended version of the research named “A Framework for Mixed Reality Application Development: A Case Study on Yapahuwa Archaeological Site” which was done by Amodth Jayawardena. The above research implemented a proper framework for developing augmented reality applications. The researcher has planned to implement a working model of the above project in Yapahuwa Archaeological site. However, still, there is a problem in implementing this project in real-life situations. That is because of non-accurate location data. Although the technology today has improved, GPS technology still doesn't accurately offer the exact location data. Therefore, when a user tries to use the above-mentioned AR application on his/her mobile, the virtual graphics generated for that location may get inaccurate because of the slight GPS location data fluctuations. However, we can improve the accuracy of the inbuilt GPS data of a mobile phone using third party devices, which are expensive. The proposed solution implemented in this is based on giving a solution to find the most accurate geographic locations for augmented reality applications. Since improving location data accuracy using third party devices is not practical in an implementation environment like Yapahuwa this research paper focused on getting the exact location using image processing technology and mobile phone's sensor data (when needed). An image file system is maintained separately for implementing the system. As an example, when the research is implemented in the real world it can identify and generate the graphic image to the relevant location when a user aims his/her mobile phone to an archaeological place. The technology used in the research can be implemented in many other industries for purposes like Google Maps in the future.

TABLE OF CONTENTS

DECLARATION	i
ACKNOWLEDGEMENTS	ii
ABSTRACT	iii
LIST OF FIGURES	vi
LIST OF ABBREVIATIONS	viii
CHAPTER 1	1
INTRODUCTION	1
1.1 Approaching to the Research on Place Detection	2
1.2 Geolocation Detection	3
1.3 Problem	3
1.4 Objective	4
CHAPTER 2	5
LITERATURE REVIEW	5
2.1 A Brief Introduction to Research Studies Carried Out	6
2.2 Visual Object Recognition	12
2.2.1 Visual Object Recognition in the Context of Mobile Vision Services	14
2.2.2 Object Recognition from Natural Features on a Mobile Phone	18
2.2.3 Using Camera-Equipped Mobile Phones	21
2.2.4 Server-Client Object Recognition for AR	22
2.2.5 Various Building Detection Using Lidar Data	25
2.2.6 Classification of a Person Picture and Scenery	28
2.2.7 Detecting Objects Using Colours	29
2.2.8 Tensorflow Object Detection API	30
CHAPTER 3	32

METHODOLOGY	32
3.1 Important Considerations When Implementing the Solution	33
3.2 Key Features the Implemented System Contains	33
3.3 Implementation of a Geolocation Detection System for Mobile	35
3.3.1 Image Recognition Technology	35
3.3.2 Use of Mobile Phone’s Inbuilt Sensors	38
3.4 Summary	39
CHAPTER 4	40
SOLUTION ARCHITECTURE AND IMPLEMENTATION	40
4.1 Image Recognition Technology (Server Implementation)	43
4.2 Use of Mobile Phone’s Inbuilt Sensors (Client Implementation)	46
CHAPTER 5	51
EVALUATION	51
6.1 Evaluation	52
6.2 Hardware Specifications	52
6.3 Evaluation process Steps and Results	52
6.4 Sample images of the developed system	58
CHAPTER 6	59
CONCLUSTION	59
7.1 Conclusion	60
7.2 Limitations	61
7.3 Future Work	62
7.4 REFERENCES	64
7.5 Appendix A - Before System Evaluation Questionnaire	70
7.6 Appendix B – Functionality Evaluation	71
7.7 Appendix C – Sample Server Side Implementation	73
7.8 Appendix D – Sample Client Side Implementation (IOS SwiftUI)	75

LIST OF FIGURES

Figure 1 – Pokmon Go	7
Figure 2 - Project Tango	7
Figure 3 - Pokemon GO Augmented Reality	8
Figure 4 - Google Street View	9
Figure 5- Bluetooth Beacon	11
Figure 6 - Identifying objects using YOLO Framework	13
Figure 7 - How YOLO Identifies an Object	14
Figure 8 - 3D visual perception for self-driving cars	16
Figure 9 - Real time edge detection	16
Figure 10 - Client-server architecture for object awareness.	17
Figure 11 - Concept for recognition from informative local descriptors.	17
Figure 12 - related to [8]	18
Figure 13 - User interacting with an interactive Poster.	19
Figure 14 - Overview of the recognition pipeline	19
Figure 15 - The poster in first task evaluation (left).	20
Figure 16 - ASUS smartphone running the used prototype	20
Figure 17 - Components of the visual code and code coordinate system.	21
Figure 18 - Application screenshots: Visual Code Recognizer, Dialer, and Profile	21
Figure 19 - code, frame, checkboxes, and canvas fields recognized.	22
Figure 20 - Screenshot of mobile augmented reality application.	23
Figure 21 - The mobile phone sends recognition requests to an external service.	24
Figure 22 - Top: Manually annotated test sequence.	24
Figure 23 - Refinement of the classification result	26
Figure 24 - Workflow of building detection.	27
Figure 25 - How BGR is formed	29

Figure 26 – Tensor-flow Object Detection	31
Figure 27 – How FAST algorithm works	36
Figure 28 – ORB multi-scale image pyramid.	37
Figure 29 – Binary Feature Vector used in BRIEF.	37
Figure 30 – Accelerometer Sensor (iPhone 6s)	38
Figure 31 – Three Tier Architecture.	41
Figure 32 – Server side code part 1	44
Figure 33 – Server side code part 2	45
Figure 34 – Accelerometer Implementation of Client	47
Figure 35 - Camera View Code Section 1	48
Figure 36 - Camera View Code Section 2	48
Figure 37 - Camera View Code Section 3	49
Figure 38 - REST API Implementation (Client)	50
Figure 39 – Server Image Identification Accuracy	54
Figure 40 – Scenery Accuracy	56
Figure 41 - Scenery Accuracy (Outdoor Only)	57
Figure 42 - Sample images of the implemented system	58

LIST OF ABBREVIATIONS

Abbreviation	Description
GPS	Global Positioning System
CPU	Central Processing Unit
VT	Vocabulary Tree
DB	Database
SIFT	Scale-invariant feature transform
AR	Augmented Reality
ORB	Oriented FAST and Rotated BRIEF Algorithm
SURF	Speeded up Robust Features Algorithm
FAST	Features from Accelerated and Segments Test