REAL TIME IMAGE AND VOICE PROCESSING INPUT FOR ONLINE MULTIPLAYER COMPUTER GAMES

A Thesis

Submitted to the Department of Computer Science and Engineering

of

BRAC University

by

Syed Shaiyan Kamran Waliullah

Student ID: 09301019

Md. Risul Karim

Student ID: 09301002

In Partial Fulfillment of the

Requirements for the Degree

of

Bachelor of Science in Computer Science

December 2012

DECLARATION

I hereby declare that this thesis is based on the results found by myself. Materials of work found by other researcher are mentioned by reference. This thesis, neither in whole nor in part, has been previously submitted for any degree.

Signature of
Supervisor

Signature of
Author

ACKNOWLEDGMENTS

Special thanks to Insan Arafat for helping us with the idea and also Imran Farid for helping us in graphical elements. We would also like to thank Dr. Khalilur Rahman for helping us in our robotic project. We also would like to thank Nayak Amir Arup for helping us with OpenCV.

ABSTRACT

Real Time image processing and voice processing in computer gaming is a very recent topic as companies like Microsoft and Sony have researched and implemented this technology recently in their gaming consoles XBOX-360 and Play Station 3 respectively. As these consoles have high performance processor likes CELL or XENON with POWER (Performance Optimization with Enhanced RISC) PC Architecture unlike desktop computers which uses x86 and legacy architecture processor which is slower than console's processors. Gaming with "Natural User Interface" such as gestures and voice was the main focus of the research of this project which is named as "Project: Control Scheme Revolution". To materialize this challenge, several approaches were carried out. An open source game engine JMonkey Engine 3 (JME3) was customized with natural user interface. To measure the goodness of the system which is being developed by this project, Frame per Second (FPS) of the game was kept in account. All the approaches were taken to integrate different libraries to the core of the Game Engine ensuring a playable FPS for good user experience. Besides that, this research is also focused on implementing this system on the Multiplayer and Massively Online Multiplayer games as the participation of the more than one user in gaming produces an extra over head on the overall processing of the computer. As not all computers are powerful enough to support all of these systems at real time effortlessly, this research also focused on developing a system to render and stream a game from high performance computer to a moderate or low performance device such as note book or android device. Performance analysis of all of these approaches to develop a system which will bring the feeling of gaming in a console to the gaming in an ordinary computer is the main goal of this project.

TABLE OF CONTENT

Contents:
Page

Title I

Declaration II

Acknowledgements II

Abstract IV

Table of Contents V

List of Tables VII

List of Figures VIII

Chapter I : **Introduction**

1.1 Real-time image and Voice processing overview 1

1.2 System Integration Overview 2

1.3 Existing Research and Implementation 3

Chapter II: **Real-time image processing approach** 4

2.1 Overview of Technologies 4

2.2 Performance analysis of different libraries 5

2.2.1 Java Media Framework 2.2.1 5

2.2.2 Open Computer Vision 8

2.3 Consistency of the proposed Technology 9

Chapter III: **Real-time Speech recognition** 10

3.1 Speech recognition library 10

3.2 Performance analysis of proposed technology 11

Chapter IV: **Game Engine** 13

4.1 An overview of Existing Game Engines 13
4.2 Game Engine Criteria 13
4.3 Comparison among existing game engines and framework 13
4.3.1 XNA Framework 14
4.3.2 Unity 3D 14
4.3.3 Unreal Development Kit 14
4.3.4 JMonkey Engine 15
4.4 Effectiveness of the proposed game engine 15

Chapter V: **Networking** 17

5.1 Multiplayer System Overview 17
5.2 Spider monkey Network Library Integration 17
5.3 Network Features of the system 17
5.3.1 Multiplayer Network 17
5.3.2 Streaming for cloud based gaming 17

Chapter VI: **Natural User Interface** 19

6.1 **Image as Input** 19
6.1.1 Fist Detection 19
6.1.2 Face Detection 19
6.1.3 Gloves with LED Detection 19

6.2 **Algorithm for NUI** 20

6.2.1 Avatar Control 20

6.2.2 Face Detection Based Character Control 21

6.2.3 Multi-Player Flight Simulation 22

6.2.4 Mouse Cursor Controlled by Gesture 23

6.2.5 Different body part detection and collaboration 24

6.2.5.1 Relative distance calculation between face and right hand for Flight Simulation Game 24

6.2.5.2 Relative distance calculation between left and right hand for Flight Simulation Game 26

6.2.6 Rotation Detection 27

6.2.7 Animation Trigger 27

6.3 **Voice as Input** 28

6.3.1 Voice input Based User Interface 28

6.3.2 Voice as In-Game Input 28

6.3.3 Strategic AI control using voice 29

Chapter VII: **Performance analysis**

7.1 Test result on a computer 31

7.2 Practical implementation and Gaming Experience 32

Reference

LIST OF TABLES

Table Page

3.1 Computer Configuration 11

3.2 JMonkey Engine Blank Screen Frame Per Second 11

3.3 Sphinx Activated JMonkey Engine Blank Screen Frame Per Second 11

3.4 Sphinx Activated In-Game 12

4.1 Computer Configuration 16

4.2 Sphinx with OpenCV activated in-Game Frame Per Second 16

6.1 Sphinx Language Model for User Interface 28

6.2 Sphinx Language Model for Game Input 29

LIST OF FIGURES

Figure Page

1.1 System Integration Overview 2

1.2 Depth Sensing using Kinect and PS3 Move Wand 3

2.1 Example of Showing Face detection in OpenCV 8

5.1 Showing Multiplayer Using Real Time Image Processing 17

5.2 Showing state of a computer 18

5.3 Showing state of a computer in this laptop 18

6.1 Showing Face Detection in game 19

6.2 Showing Gloves Detection in game 20

6.3 Controlling Avatar using face detection 20

6.4 Face detection based character control and animation 22

6.5 Multiplayer Flight Simulation 23

6.6 Controlling mouse cursor using image processing 23

6.7 Face and Hand's relative distance as input 25

6.8 Relative distance of both hands as input in game 26

6.9 Rotating object by calculating depth 27

6.10 Voice command for Non Playable Character control 29

7.1 per second of the Flight Simulator Game 31