INTRODUCING TANGIBLE OBJECTS INTO MOTION CONTROLLED GAMEPLAY USING MICROSOFT KINECT

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

L. GAMZE BOZGEYİKLİ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF MODELING AND SIMULATION

AUGUST 2012

INTRODUCING TANGIBLE OBJECTS INTO MOTION CONTROLLED GAMEPLAY USING MICROSOFT KINECT

Submitted by **L. GAMZE BOZGEYİKLİ** in partial fulfillment of the requirements for the degree of **Master of Science in Game Technologies, Middle East Technical University** by,

Prof. Dr. Nazife Baykal ____________________
Director, **Informatics Institute**

Assist. Prof. Dr. Hüseyin Hacıhabiboğlu ____________________
Head of Department, **Modeling and Simulation**

Assoc. Prof. Dr. Veysi İşler ____________________
Supervisor, **Computer Engineering**

Examining Committee Members:

Prof. Dr. Uğur Halıcı ____________________
Electrical and Electronics Engineeering, METU

Assoc. Prof. Dr. Veysi İşler ____________________
Supervisor, Computer Engineering, METU

Dr. Umut Durak ____________________
Roketsan

Assist. Prof. Dr. Hüseyin Hacıhabiboğlu ____________________
Modeling and Simulation, METU

Dr. Erdal Yılmaz ____________________
Sobee Studios

Date: August 29, 2012

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this wok.

Name, Last name : L. Gamze Bozgeyikli

Signature :

ABSTRACT

INTRODUCING TANGIBLE OBJECTS INTO MOTION CONTROLLED GAMEPLAY USING MICROSOFT KINECT

Bozgeyikli, L. Gamze
M.Sc., Department of Modeling and Simulation
Supervisor: Assoc. Prof. Dr. Veysi İşler

August 2012, 137 pages

Recent years have witnessed great improvements in ways of game controlling yielding to higher level of interaction. Release of motion controller devices radically changed the conventional ways of interaction that have been used for controlling games so far, also giving developers the opportunity of exploring various new possible ways of interaction. One of these off the shelf tools, Microsoft Kinect for Xbox 360, recognizes motions of the players as game controlling inputs. Although touchless interaction is perceived to be attractive, games that mimic real life activities such as table tennis, sword fighting, baseball and golf may benefit from the player's holding a tangible object to get more involved into game, sensing the actions deeply. In this thesis, a tangible gameplay interaction method that senses whether or not the player holds an object in the hand; if so, detects its dimensions and incorporates the hand-held object into gameplay by projecting motions of the player accordingly, is developed using Microsoft Kinect for Xbox 360. Developed algorithm is implemented on an experimental game and a user study is performed which revealed that an improved gameplay with more natural and accurate motion controlling yielding to new possible actions is achieved with the developed system.

Keywords: Tactile, Tangible Interaction, Motion Tracking, Kinect, Motion Controlled Video Games

ÖZ

FİZİKSEL NESNELERİN HAREKETLİ OYUN KONTROLÜ KAPSAMINA MICROSOFT KINECT KULLANILARAK DAHİL EDİLMESİ

Bozgeyikli, L. Gamze
Yüksek Lisans, Modelleme ve Simülasyon Bölümü
Tez Yöneticisi: Yard. Doç. Dr. Veysi İşler

Ağustos 2012, 137 sayfa

Geçtiğimiz yıllarda oyun kontrolü alanında daha fazla etkileşime olanak sağlayan büyük gelişmeler yaşanmıştır. Hareket kontrol kumandalarının kullanıcılara sunulması süregelen alışılmış oyun etkileşiminde köklü bir değişikliğe yol açmış ve geliştiriciler için pekçok yeni etkleşim yolunu inceleme ve araştırma olanağını beraberinde getirmiştir. Oyun kontrolü alanında son teknolojiye sahip bu cihazlardan biri olan Microsoft Kinect for Xbox 360, kullanıcıların hareketlerini kontrol girdisi olarak algılayabilmektedir. Temassız etkileşimin ilgi çekici ve cazip olduğu düşünülse de, gerçek hayattaki fiziksel aktiviteleri taklit eden masa tenisi, eskrim, beyzbol ve golf gibi bazı oyunlar için kullanıcının elinde fiziksel bir nesne taşıması, oyuna daha çok dahil olması ve hareketleri daha derinden hissetmesi açsından daha yararlı olabilir. Bu tez dahilinde, Microsoft Kinect for Xbox 360 kullanılarak, oyuncunun elinde fiziksel bir nesne taşıyıp taşımadığını algılayan, eğer taşıyorsa nesnenin boyutlarını algıladıktan sonra elde tutulan nesneyi kullanıcının hareketlerini yansıtarak oyuna dahil eden bir fiziksel oynanış yöntemi geliştirilmiştir. Geliştirilen yöntem, deneysel bir oyun üzerinde uygulanmış ve etkileri ölçmek için bir kullanıcı çalışması gerçekleştirilmiştir. Sonuçlar, geliştirilen yöntemin kullanıcılara daha doğal ve hassas kontrol olanağı ile yeni olası hareketler sunarak daha iyi bir oynanış şekli sağladığını göstermektedir.

Anahtar Kelimeler: Dokunsal, Dokunsal Etkileşim, Hareket Algılama, Kinect, Hareket Kontrollü Video Oyunları

To the awesomest man of the universe; my adorable husband Can…

ACKNOWLEDGEMENTS

This thesis would literally never ever have been complete without my stunning husband Can Bozgeyikli. I am deeply grateful for all the meaning, charm and inspiration you fulfilled my life with. I feel so lucky for living the life beyond my imagination in our wonderland, celebrating every day. I am thankful for your endless support, help and all the laughter we shared through those sleepless nights at the lab, and of course our fleeting run offs to World of Warcraft, almost fleeting. Whatever we do, we do with joy; this was the same for our theses, it is laughter and joy I remember the most instead of difficulties of preparing a whole thesis, everything is just so easy and fabulous with -and only with- you, I love you…

We are so lucky for having two lovely mothers and fathers, feeling support of them always with us. Many thanks to our mothers Mihriban Bozgeyikli and Oya Yılmaz and our fathers Halil Yılmaz and Haluk Bozgeyikli (mentioned in alphabetical name order since they are indistinguishable to us) for unconditionally supporting us all the time with their care and love, and cheering us up with amusing phone call breaks during our busy studying periods, always reminding us not to work so much! We would also like to thank a lot to our cute sister and brother, Ozlem Yılmaz and Arda Bozgeyikli (lady is first) for always standing by us and lighting up our worlds with their sweet beings. And of course, for all the games we shared together!

We, as me and my husband, would like to express our gratitude and endless thanks to our advisor Veysi İşler, who always supported us throughout our graduate education, much more than an instructor, as a mentor who we admire and appreciate a lot. We were so lucky of being able to laugh with our thesis advisor instead of working in a stressful atmosphere, which we strongly believe has reflected to our works also. Our sound computer graphics and animation foundations are his invaluable gift to us. We are indebted to him for the tremendous knowledge he taught us and we believe that in every successful work of us, there will certainly be a remarkable trace of him, even if he will not be physically involved in, through the great teachings we were able to learn from him not only on game technologies but also on life. And we thank to him most importantly for believing in us, always saying "You can do it." by meaning it. It is a great privilege to have an advisor who really listens to us and makes us feel important among his busy schedule. We are so proud of and lucky for being students of him, and it is always a huge motivation for us, not to let him down. We would also like to thank him a lot for the generous physical equipment he provided us to conduct our researches and theses in GATE-R lab at MODSIMMER.

We would like to express sincere thanks to our warm hearted and kind instructor Erdal Yılmaz for all the valuable knowledge he taught us, both academic and industrial, and making us feel that he is always there ready for help in case we needed.

We would also like to thank a lot to our tender and kindhearted instructor Uğur Halıcı for coloring up our lives with the graceful aesthetic vision of her and for the advanced C++ we were gratefully able to learn from her.

We would like to express many thanks to our affectionate instructor Tolga Tem for making us more aware of the melodies hidden in life and sounding up the music, also for his gentle personality whom we enjoy working with a lot.

We would also like to thank a lot to our friendly instructor Umut Durak for the valuable massive amount of game physics we learnt from him at encouraging, vivid and enjoyable class sessions.

We would also like to express many thanks to our sincere instructor Hüseyin Hacıhabiboğlu for the game audio programming we were luckily able to catch from him by auditing his courses.

Last but not least, we would like to express our thanks to MODSIMMER (Modeling and Simulation Research Center) for all the equipment and facility we benefited during our theses studies.

PHD Comics (http://www.phdcomics.com) is also worth mentioning with all the entertaining stuff we laughed a lot and often felt familiar. It would be a nice reminder to follower graduate students to conclude with the amusing -but unquestionably true- PHD Comics modification of Newton's Third Law: "For every action towards graduation there is an equal and opposite distraction.".

TABLE OF CONTENTS

ABSTRACT iv

ÖZ vi

DEDICATION viii

ACKNOWLEDGEMENTS ix

TABLE OF CONTENTS xiii

LIST OF FIGURES xvi

LIST OF ABBREVIATIONS xx

CHAPTER

1. INTRODUCTION 1
2. RELEVANT WORK AND BACKGROUND 5
   - 2.1. Tangible Interaction 5
     - 2.1.1. Physicality 6
     - 2.1.2. Children Development 7
     - 2.1.3. Naturalness and Intuitiveness 8
     - 2.1.4. Realism 9
     - 2.1.5. Fun 9

2.2. Drawbacks of Touchless Game Controlling 10
2.2.1. Lack of Haptic Feedback 11
2.2.2. Unfamiliarity 11
2.2.3. Form... 13
2.2.4. Bias 14
2.3. Tangible Games 16
2.3.1. Games from the Academia.. 16
2.3.2. Commercial Games 18
2.4. Motion Sensing Game Console Peripherals.................................... 21
2.4.1. Nintendo® Wii™ 22
2.4.2. Sony® PlayStation Move™ 26
2.4.3. Microsoft® Kinect™ 28
3. PROPOSED APPROACH 37
3.1. Problem Statement 38
3.2. Object Detection Algorithm 38
3.3. Technical Improvements 59
3.3.1. Object Recognition.. 60
3.3.2. Indirect Rolling Motion Recognition 61
3.4. Limitations 63
3.5. Implementation.. 63
3.5.1. Finding First Point on the Object 65
3.5.2. Finding Second Point on the Object 69
3.5.3. Finding Direction of the Object.. 72
3.5.4. Finding Height of the Object 72
4. EVALUATION 74
4.1. Experimental Game Design 74

4.1.1. Control Version of the Experimental Game 78
4.1.2. Experiment Version of the Experimental Game 80
4.2. Participant Information 82
4.3. Experiment Procedure 84
4.4. Results 86
4.4.1. Questionnaire Results 86
4.4.2. Subjective Mental Effort Questionnaire 101
4.4.3. Quantitative Data 104
4.4.4. User Preference 108
4.4.5. User Reviews 109
4.5. Discussion 111
5. CONCLUSION AND FUTURE WORK 115
5.1. Conclusion 115
5.2. Future Work 116

REFERENCES 117

APPENDICES

A. PARTICIPANT INFORMATION FORM 126
B. CONTROL GAME QUESTIONNAIRE 127
C. EXPERIMENT GAME QUESTIONNAIRE 130
D. CONTROL GAME SMEQ 133
E. EXPERIMENT GAME SMEQ 134
F. CONTROL GAME USER EXPERIENCE STATEMENT 135
G. EXPERIMENT GAME USER EXPERIENCE STATEMENT 136
H. USER PREFERENCE FORM 137

LIST OF FIGURES

Figure 2.1 Nintendo Wii Motion Sensing Video Game Console 23

Figure 2.2 Nintendo Wii Motion Sensor Bar 24

Figure 2.3 Nintendo Wii Motion Controllers 25

Figure 2.4 Nintendo Wii MotionPlus Expansion of Wii-mote 25

Figure 2.5 PlayStation Move Motion Controller 26

Figure 2.6 PlayStation Move Navigation Controller 27

Figure 2.7 Microsoft Kinect for Xbox360 28

Figure 2.8 IR Rays Cast by Microsoft Kinect for Xbox360 29

Figure 2.9 Infrared Rays Scattered from Depth Camera 30

Figure 2.10 Depth Image Stream and Skeleton Tracked by Kinect 31

Figure 2.11 Joint Distribution of the Skeleton Tracked by Kinect................. 32

Figure 2.12 Hand-Held Object Ignored by Current Skeleton Tracking Scope of Microsoft Kinect 34

Figure 2.13 "Kinect Sports Pack" that is produced by Intec - Image Adopted from [40] 35

Figure 3.1 Global and Local Coordinates 40

Figure 3.2 Visualization of Kinect Field of View with Player Illustrations Standing at Different Distances from the Sensor 44

Figure 3.3 Player Occupying Different Proportions of the Screen while Standing at Different Distances from the Sensor 45

Figure 3.4 Finding Two Edge Points of the Hand-Held Object with Linear Searches 50

Figure 3.5 First Point of the Object P1 is Determined 51

Figure 3.6 A Visualized Summary of Algorithm Steps 9 to 13 53

Figure 3.7 Misinterpreted Object Direction Resulted from Using One Point on the Object and Hand Joint Data to Find the Direction 55

Figure 3.8 Player Leans the Sword Forth and Back 56

Figure 3.9 A Visualized Summary of Algorithm Steps 14 to 16 58

Figure 3.10 Player Carrying Different Objects at Hand 61

Figure 3.11 Player Rotates the Sword in Rolling Axis of Motion 62

Figure 3.12 Variable Initialization, Arm Direction Determination and Right Hand Joint Data Gathering 65

Figure 3.13 Depth Comparison Search Performed to Detect the First Edge Point of the Object 66

Figure 3.14 Depth Detection Searches Performed with Decreased Radii 67

Figure 3.15 First Point Lying on the Object is Determined 68

Figure 3.16 Radius Assignment of Second Search Circle and Variable Initialization 69

Figure 3.17 Depth Comparison Search Performed to Detect the Second Edge Point of the Object 70

Figure 3.18 Second Point Lying on the Object is Determined 71

Figure 3.19 Direction of the Object is Found 72

Figure 3.20 Height of the Object is Found 73

Figure 4.1 An Ingame Screenshot from the Designed Game Hoopla 77

Figure 4.2 Instructions Screen of the Control Game which Presents the Player Required Actions to Play the Game and the Winning Condition 79

Figure 4.3 Player Playing the Control Game 80

Figure 4.4 Instructions Screen of the Experiment Game which Presents the Player Required Actions to Play the Game and the Winning Condition 81

Figure 4.5 Player Playing the Experiment Game 82

Figure 4.6 SMEQ Chart 102

Figure 4.7 SMEQ Histograms for Version 1 and Version 2 103

Figure 4.8 SMEQ Distributions for Version 1 and Version 2 103

Figure 4.9 Number of Fails Histograms 105

Figure 4.10 Completion Time Histograms 106

Figure 4.11 Completion Time Distributions 106

Figure 4.12 Average Success Times Histograms 107

Figure 4.13 Preference Score Distributions 108

LIST OF ABBREVIATIONS

2D	Two Dimensional
3D	Three Dimensional
CEGEQ	Core Elements of the Gaming Experience Questionnaire
CIT	Critical Incidents Technique
$\mathbf{DJ_{RE}}$	Position of Right Elbow Joint in Depth Coordinates
$\mathbf{DJ_{RH}}$	Position of Right Hand Joint in Depth Coordinates
$\vec{\mathbf{d}}_{\mathbf{o}}$	Direction of the Object
$\mathbf{d_{RH}}$	Depth of the Right Hand Joint
$\mathbf{DT_o}$	Position of Tip Point of the Object in Depth Coordinates
e	Edge Point of the Object
fps	Frames per Second
HDMI	High Definition Multimedia Interface
$\mathbf{h_o}$	Height of the Object

IR	Infrared
$\mathbf{P_1}$	First Point on the Object
$\mathbf{P_2}$	Second Point on the Object
RGB	Red-Green-Blue
$\mathbf{r_{n,m}}$	Search Circle Radius of n^{th} Iteration's m^{th} Search Set
SDK	Software Development Kit
SMEQ	Subjective Mental Effort Questionnaire
$\mathbf{WT_o}$	Position of Tip Point of the Object in World Coordinates
$\mathbf{WJ_{RH}}$	Position of Right Hand Joint in 3D World Coordinates
ε	Depth Difference Threshold Constant

CHAPTER 1

INTRODUCTION

Video games keep on spreading through people's lives regardless of their age. Besides improved graphics and sound, recent years have witnessed great improvements in game controlling, which yielded to higher level of interaction possibilities with games. Release of Nintendo Wii [1] in 2006 followed by releases of Sony PlayStation Move [2] and Microsoft Kinect for Xbox 360 [3] in 2010 radically changed the conventional ways of interaction that have been used for controlling games so far, bringing along the opportunity of exploring various new possible ways of interaction for developers and researchers. Advanced motion sensing capabilities of these off the shelf game consoles made it easier to employ natural ways of interaction with digital devices in many areas, but at the very most in game technologies; by employing gesture and body movements as game controlling inputs rather than using conventional game pads as controllers. With these novel improvements, nowadays players can enjoy more realistic gameplay interactions, in which they are fully involved more actively than pressing on some buttons at a sitting posture.

Both Nintendo Wii and Sony PlayStation Move video game consoles involve specially designed tangible game controllers that are used by holding at hand and performing various movements such as swinging, tilting and lining for motion controlling. On the other hand, Microsoft Kinect for Xbox 360 requires no controllers to be held by the player for motion controlling, it rather senses the player's actions as game inputs, turning the player into the controller and presenting a more immersive and attractive new way of gameplay.

With the release of Microsoft Kinect for Xbox 360, various compatible commercial video games have also been released in which the player uses his/her own body movements and/or gestures to control the game. It is stated that Kinect sensors reached 10 million sales and standalone Kinect games reached more than 10 million sales worldwide as of 2011 [4]. These sales figures indicate a large popularity of both Kinect and specially designed games among players and may be interpreted as an evidence for the new way of controlling's being attractive.

Although it may be perceived by the players to be funny and attractive to control games touchlessly without using any hand-held controllers, it is not known for sure that a gameplay of controlling without any tangible objects is an improved one over controlling with tangible objects yet and this reveals a new area of exploration in video game controlling with motion sensing technologies. In the motion tracking video game systems till Microsoft Kinect for Xbox 360, the motion controller also served as a tangible object representing the virtual object that is being interacted in the video game. But Microsoft Kinect for Xbox 360 came with a unique motion sensing technology that is capable of tracking the player's motions without any additional equipped hand-held device involving accelerometers or optical sensors. This also brought along the lacking that absence of any hand-held tangible object causes. The question then arises: Does touchless control without holding any tangible object always yield to the best possible gameplay? In many games, although touchless motion controlling intuitively seems to be well suited and attractive, some games that mimic real life physical activities such as table tennis, sword fighting, baseball and golf may benefit from a tangible object that is held by the player, providing more involvement into the game, sensing the actions deeply. Wiebe et al. state that people tend to feel an experience more real if the touching sense is involved into it [5]. A familiar everyday example of the situation can be given as a person's grabbing an object resembling a microphone while singing for pleasure to feel the experience deeper.

In this thesis, an improved gameplay is aimed to achieve via introducing hand-held tangible objects into motion controlled gameplay using Microsoft Kinect for Xbox 360. With the developed algorithm, tangible object is detected and movements of the user performed with the handheld object are projected into gameplay. To evaluate effects of the developed interaction system on player experience, an experimental game is developed with two controlling versions. A user study is then performed and the results are analyzed with appropriate statistical methods. Results of the conducted user study revealed that tangible motion controlling provided the players a more realistic and natural experience with the ease of mental motion mapping.

Outline of the thesis can be summarized as follows:

- Chapter 2 first provides relevant work in the area of tangible interaction followed by drawbacks of touchless motion controlling. Then, motion controlled game examples from academia and game industry are provided. The chapter is concluded with the discussion of current motion sensing game console peripherals.
- Chapter 3 presents the proposed approach. First, problem definition is stated. Then, object detection algorithm is discussed in detail. Technical improvements achieved by the developed algorithm are presented next. Limitations of the algorithm are discussed and the chapter is concluded with the implementation of the algorithm that is supported with code excerpts provided.
- Chapter 4 includes evaluation of the developed motion controlling method. Experimental game design is presented at the beginning of the chapter for two controlling versions. Information of the participants that took part in the user study is mentioned then. Experiment methodology is discussed which is followed by the results of the experiment and relevant discussion.
- Chapter 5 presents conclusion of the thesis study and future work implications.

CHAPTER 2

RELEVANT WORK AND BACKGROUND

Motion controlling for video games is an immature area that is being evolved with present researches, but there is a vast of previous works on physical world tangible interaction that discuss different aspects of the phenomenon. In the following subsections, previous works that are related to tangible interaction and drawbacks of touchless controlling are discussed as an explanation of why this thesis is necessary and important, followed by background information that may help the reader in understanding the algorithms developed in the following chapter to introduce tangible objects into motion controlling for an improved player experience are presented.

2.1. Tangible Interaction

Tangible interaction is not a new concept since touching is an indispensable sense of human being since the very first existence, but it is a recent area of research in the digital entertainment field which is attracting a lot of attention with the new possibilities that are presented by the novel enhancements of motion sensing devices. There are many previous works that emphasize the importance of tactile interaction both in the area of human computer interaction and natural sciences. Among a broad range of previous works, in the scope of this thesis, in the following subsections, studies that are related to importance and effects of tangible interaction are mentioned in five categories of Physicality, Children Development, Naturalness and Intuitiveness, Reality and Fun.

2.1.1. Physicality

There are various previous works emphasizing the importance of physicality in projecting desired real world actions into virtual word. Yao et al. criticize that majority of video games are using mouse, keyboard or touch screen as interaction instruments, causing a lack of the physical metaphors which can be useful for the player while manipulating the virtual in game objects [6]. Cheok et al. state that physical interaction provides a psychological advantage to players and helps them to enter into virtual world from real world in an effective and easier way [7]. The authors emphasize that it is helpful for the players to keep a mental model of the real world actions that are to be performed in virtual environments, providing an easier understanding. The authors also state importance of tangible interaction as "Physical interaction thus should be a fundamental element for the next-generation entertainment". From a similar point of view, Hornecker expresses the importance of physicality in interaction with this statement: "Physicality is a central aspect of embodied interaction. We are incarnate, physical beings that live in a physical world." [8]. The author suggests that humans are not abstract beings and are inherently suitable for physical interaction.

Lok et al. conducted a study in which they measured the effect of handling real objects on user performance in cognitive tasks [9]. In the study, users performed the same tasks with real objects and virtual representations, results indicating that interaction with real objects positively affects the performance as compared to interacting with virtual objects. The authors state that real object interaction improves the user performance in tasks of spatial cognition, resulting in performance measures that are similar to the ones that are achieved in real world conditions of the same activity. They also suggest that motion constraints and tactile feedback provided by using real objects help the user in the interaction, making it more realistic, closer to real world interactions instead of virtual ones, which may cause the users to associate interaction mechanics incorrectly.

2.1.2. Children Development

Physical interaction is stated to be healthy for a child's growth by many previous works that are focused in the area of children development. Yao et al. state that "It is also known that children develop many skills by interacting with the physical world but most of these potential skills are not employed by video games." [6]. In Spina-Caza's work children are stated to be tactile learners, and it is explained that during video game design children's possible benefits from tactile learning are usually ignored and this lacking can be improved by employing new approaches in tangible design [10].

2.1.3. Naturalness and Intuitiveness

Many previous works assert that physical interaction is natural and intuitive, which are desirable features of video game controlling. Wachs et al. state that intuitiveness is an important attribute of entertainment interfaces and games that mimic real human motions such as motion controlled sport games may be specified as intuitive [11]. On the other hand, Thomas asserts that vision and hearing are not sufficient senses for future's interaction [12]. The author suggests that new devices will be employed that support more natural interaction forms including touching. From an alternative point of view, Hornecker states that the boundary between digital and physical world gets transparent as more natural physical interaction forms are achieved [8].

2.1.4. Realism

Tactile sense provides a more real experience to humans than abstract illusions. In the work of Wiebe et al., touching is specified to be an active discovery sense by the authors and it is stated that "Something that is touched is perceived more real than something that is seen" [5]. A realistic interaction may provide the players a more comfortable experience as compared to abstract interaction.

2.1.5. Fun

Physical interaction is considered to be bringing fun into game controlling by many previous works. Saffer states that a tennis game that is designed to be motion controlled with real tennis actions performed is more entertaining than a tennis game that is designed to be controlled with pressing keyboard buttons [13]. The author explains that the player's seeing the mirrored actions of himself is also a fun factor rather than pressing a button and watching some other virtual exaggerated action that is performed on screen. The author mentions the inviting aspect of motion interaction as follows: "Gestural systems encourage play and exploration of a system by providing a more hands-on (sometimes literally hands-on) experience." Hornecker states that new genres of games are emerged with the improved motion sensing technology of off the shelf devices such as Nintendo Wii, Microsoft Kinect and Sony PlayStation

Move that could not be played enjoyably with traditional input devices of keyboard and mouse such as yoga [8]. The author states that "We are just re-discovering that physical interaction may increase both usability and enjoyment.". Zaman et al. assert that increasing studies on the effect of tangible interaction is due to physical manipulation's being embodied, providing a more natural and funnier way of interaction [14].

2.2. Drawbacks of Touchless Game Controlling

Although motion controlling without using any tangible controllers may fit into some game styles attractively as an effective interaction method, there may be lacking encountered in some games that simulate activities which are performed with tangible objects in real life such as sword fighting, golf and tennis.

LaViola and Keefe classify spatial interfaces into two as real and magical [15]. Real interfaces aim at imitating real world interactions such as swinging a tennis racket. Magical interfaces are constructed with imagination and involve unreal interactions such as casting a spell. Touchless interaction may suit to magical interfaces where known movements do not mean so much, but there may arise some problems with real interfaces which mimic real interactions performed using physical objects.

In this subsection, drawbacks of touchless and specially equipped motion controller using video game controlling are discussed from different aspects under relevant categories of lack of haptic feedback, unfamiliarity, form and bias.

2.2.1. Lack of Haptic Feedback

Haptic is a term that can be defined as being related to the touching sense. Physical objects give the user haptic feedback, making it easier to coordinate and project their spatial actions since human being is more close to physical than virtual by nature [8]. In the absence of any physical feedback, coordinating and projecting desired actions become more difficult for the players which can be considered as a drawback that causes distraction.

2.2.2. Unfamiliarity

Controllers of the video game systems that enable motion tracking such as Wii-mote and PlayStation Move motion controller are specially designed to be having electronic equipment inside to provide motion detection; hence are expensive compared to the price of daily used plastic toys of children. Also, these controllers are not familiar objects to the users at first and may need time for the user to get accustomed to them. Familiar objects that are used for motion controlling such as a toy sword or racquet may increase the level of confidence of the player while playing motion controlled games rather than using unfamiliar and relatively expensive controllers which can arise a concern to the player for damaging or breaking it.

Players may enjoy playing a sword fight game with an inexpensive plastic toy more than using a specially equipped expensive remote controller attachment without caring for the controller's well shape. Moreover, it may be more familiar for the user to play the game with an accustomed object, maybe the player's favorite one, instead of specially equipped tangible controllers. This familiarity adds richness to the games since it provides a variety of controlling the same game with different objects which yield to different experiences and it may enable especially children to improve their imagination and internalize the game perceiving it to be more engaging. It may nurture children's imaginative power being able to play video games with their belonged toys, inventing different scenarios. It also provides richness in gameplay with different consequences such as playing a game with a toy sword or plastic beverage bottle. As the object's dimensions are incorporated into the game, even different sized objects may provide different gameplays with different levels of enjoyment. An example can be given as playing table tennis with a standard table tennis paddle and much bigger tennis racket. With the bigger tennis racket it should be more challenging to play table tennis, it may be easier to catch the ball since the big racket will occupy a considerable portion of the acceptable area in which the ball may fall but on the other hand it may be more difficult to send the ball in the desired direction with a big racket since the player may not move and rotate it as rapidly as he does with the small paddle. But it may provide more entertainment to the player, being able to play an exaggerated version of a classic sport simulation with just changing the object in his hand and seeing the effects of the changed object on the game instantly.

2.2.3. Form

Specially equipped motion controllers have unique forms of their own. There are previous works that emphasize the importance of the used physical object's form being as descriptive as possible to help the users understand how they would interact using it. LaViola and Keefe state that forms of the tangible objects give the user intuition about how to interact with them [15]. The authors mention that it is important to choose the form of a new developed real world object controlling tool to be as similar as possible to its virtual duplicate since it will provide directness and familiarity yielding to ease of use. The authors assert that forms of the physical tools tell the user about how to use them. Bowman et al. state that graphical drawing tablets that are designed as traditional sheet and pen is a representative example of familiar design that gives the user a lot of clues about how to use it [16].

It may increase the confidence of the users and provide a more explanative and easier way of interaction to enable them using similar real world objects to their virtual duplicates as game controllers.

Moreover, form of the tangible object may affect gameplay positively, adding variety and richness. An example can be given as golf, which is played with different sized clubs in real life. A game mimicking real life golf sports activity would be more realistic when played with different sized objects since they will exert different forces to the ball as they are swung. For example, in the commercial golf game in Kinect Sports Season 2 [17], the player changes the virtual golf club the avatar holds with voice command, but always does the same swinging motion since he does not hold any physical object but pretending to do so, which degrades the realism since swinging a long golf club and a short one may have different mechanics.

2.2.4. Bias

Video game technologies pioneer Nolan Bushnell states that a good game should be easy to play yet hard to master [18]. Lack of a hand-held physical object in the form of full body motion controlling without using any controllers may lead to a biased gameplay. To give an example, a sword fighting game that is controlled by swinging the hand without holding nothing gives the player the chance of moving his/her hands very rapidly to achieve high scores easily, theoretically mastering the game without facing any challenges which may yield to boredom. This may result in the game's being perceived as being too easy to be worth playing by the player. On the other hand, holding an object that has some weight and volume, not necessarily to be heavy, restricts the player from making biased rapid moves. This constitutes another basis for considering the improvement that may be achieved by using a tangible object in motion based game controlling. In an interview made with the executive producer of the commercial Kinect game Kinect Star Wars during E3 Electronic Entertainment Expo, the interviewer doubts on the lacking of the weight sense caused by touchless controlling and questions how the developers achieved restricting biased rapid and/or very small moves of the players while controlling the game which may cause boredom and result in losing the player [19]. Jerauld states that there exists no special algorithm or method to sense the weight applied to their game, which means whether the player makes a small movement or a big one, the virtual avatar will perform the same action, but they tried to direct the player towards making bigger moves with a rewarding system of fancy sparkles and some attractive graphical effects that only appear as the player performs a big move. The player's holding a tangible object may help eliminate the lack of weight problem yielding to a more realistic, challenging and engaging gameplay rather than a biased and boring one.

Considering possible drawbacks of motion controlling without holding any tangible objects or holding specially equipped motion controllers that are explained so far, it may be inferred that giving players the option of holding self-selected objects in their hands during gameplay may presumably improve the gaming experience by providing various advantages that are mentioned so far.

2.3. Tangible Games

There have been various previous works on experimental games and applications of tangible motion sensing technologies both in the field of game technologies and other areas of arts and sciences that are accomplished recently. In the following subsections, some of these previous works that are related with the scope of this thesis are presented in two categories of academic games and commercial games.

2.3.1. Games from the Academia

Many previous academic works that are developed for research purposes use existing physical objects as tangible interfaces with different interaction approaches employed.

Harfield et al. employed a distributed multi player tangible gaming technology to develop a traditional tug of war game for children digitally with the usage of a spring and a rotation sensor for the measurement of the forces applied to a tangible rope by the players [20]. The work is an effective example of combining physical interaction with digital gaming and stated to be the first example of the authors' term "distributed tangible technologies".

As another example, Ishii et al. presents a ping pong gaming interface, which they term as “reactive ping-pong table” that enables the players use standard tangible racket sets as controllers and full body motions as the controlling inputs [21]. In the game, a reactive ping pong table is achieved using sound based ball tracking and familiar yet attractive game playing is provided with different graphical design patterns projected onto table that change with the rhythm and style of the game such as water ripples and spots. The work provides a significant example of using familiar tangible objects as video game controllers but as a drawback, the gaming interface requires a specially equipped digital table which is neither easy nor inexpensive to obtain.

In their work, Yao et al. introduce two developed games that are called Multi-Jump and Multi-Fly in an effort of extending video games with tangible interfaces to multiplayer social gaming environments, in which the players experience both physical activities and social interactions [6, 22]. Multi-Jump is a digital jumping rope game and Multi-Fly is a digital kite flying game. In both games, rope is used as both the game and the interface controller that serves as a tangible interface with the usage of special equipped hardware employed. In Multi-Jump, rope is twirled by two players and the motion is traced using an accelerometer located in the handle and in Multi-Fly, rope is used to control the kite’s position with the usage of a constant force spring and potentiometer. The games constitute notable examples of the combination of video games and physical activities that incorporate tactile sensing.

Ryokai et al. introduce a digital drawing tool especially for children to improve their cognitive development, which allows them to interact with their environments and picking up colors, textures and movements of common objects with the usage of a small video camera that has lights and touch sensors embedded in it [23]. The work is an effective example of combining tangible interfaces with digital art but as a drawback, it requires a specially equipped tool to operate.

Although all of these examples has their own merits and gave valuable contributions to the field; as a common drawback, to use tangible objects as the game controller these systems need special equipment such as accelerometers, gyroscopes, projectors and cameras that can be burdensome and costly for the player to obtain.

2.3.2. Commercial Games

There are various commercial Kinect games that use body motions of the player as controlling inputs such as sport simulations of Kinect Sports [24] and Kinect Sports Season Two [17]; action adventure genre games of Harry Potter and the Deathly Hallows™ - Part 1 [25] and Kinect Star Wars [26]; and arcade style game Fruit Ninja Kinect [27]. These games are controlled with only body motions without any tangible controllers involved but are considered to be suitable for controlling with tangible objects according to the scope of this work since the player simulates sports that are played with tangible objects in real life such as swinging a golf club, playing tennis with a racket; and performs some special actions with a virtual wand, a virtual light saber or a virtual ninja sword. All of these movements mimic some physical actions without using any tangible object held and may be exposed to the drawbacks of touchless gameplay interaction lacking a physical object that will be mentioned in the upcoming Section 3.1. For example in the game Harry Potter and the Deathly Hallows™ - Part 1, lack of any physical wand held by the player during gameplay degrades realism since some unnatural moves are assigned to the player instead of a natural interaction that would be achieved by using a tangible object as the virtual character holds. The same situation occurs in the game Kinect Star Wars in which the virtual character carries a light saber but the player controls it by performing large moves without holding anything. Besides, these games and many more may benefit from an alternative controlling style of optionally using tangible objects with body movements.

Angry Birds is a popular puzzle genre game that is created by Rovio [28]. The game is stated to reach 1 billion total sales for different platforms as of May 2012 by Vesterbacka [29]. The game is mainly designed for touchscreen devices but there have emerged various haptic additions to the game developed by different companies or individuals.

Matsui and Spitz created a tangible slingshot controller for PC called “Super Angry Birds” which attracted a lot of attention [30]. The controller provides accurate controlling of the virtual in game slingshot, providing force feedback with the special hardware inside such as motorized fader, electric motor and board. The haptic controller is stated to add realism and richness to the game controlling by physicality that keyboard or touchscreen is not capable of.

Another realistic adaptation of the game is made by T-Mobile, a mobile communication company [31]. The company created a real life version of the Angry Birds that is controlled via a smartphone device [32]. The player aims the slingshot using touchscreen and when he/she releases his/her finger, a physical big bird is launched towards the pre-built real world level that is constructed with real physical objects of wood bars and pigs. The work attracted a lot of attention and lots of people queued enthusiastically to try the real life version of the game.

Another tangible adaptation of the game is the real toy version called "Angry Birds Knock on Wood" [33] that is produced by a toy company called Mattel. The tangible toy kit involves plastic models of the game objects and provides real collisions and interaction. The toy targets especially children and may improve their imagination and learning with self-constructed levels and physical interaction it provides.

All of these adaptations are towards more realism and may be interpreted as people look for more real interactions and gameplay that extends over tangible toy kits of the video games involving physical versions of game objects.

2.4. Motion Sensing Game Console Peripherals

There are recent off the shelf motion sensing video game consoles that initiated a new era of video game interaction. Milestones of the video game motion controlling can be stated as Nintendo Wii, Sony PlayStation Move and Microsoft Kinect for Xbox 360 chronologically. There are following remarkable motion controlling endeavors also such as Asus Wavi Xtion that enables motion controlled gaming and PC control with navigation, having an underlying technology similar to Kinect [34]. Instead of being a video game console accessory, the recently developed device is standalone and operates with wireless High Definition Multimedia Interface (HDMI) technology.

There are smart televisions that provide motion controlling such as Samsung SmartTV [35] which enables motion control for navigation and gaming. Most of the smart phones already include special orientation detecting equipment such as accelerometers and gyroscopes that enable game playing with motion controlling such as leaning the device to sides during gameplay to make the character move in the leaned direction. There are recent advanced endeavors trying to incorporate touchless motion controlling into smart phones to enable controlling mobile phones with body movements such as starting a phone call with a hand swipe and playing mobile games with body motions [36].

All of these advancements indicate that motion controlling will continue pervading with a broad range of usage area in the future.

In this subsection, novel motion controlling devices Nintendo Wii, Sony PlayStation Move and Microsoft Kinect for Xbox 360 are presented with their important features and brief information on their working principles.

2.4.1. Nintendo® Wii™

Wii is the motion sensing video game console that is released by Nintendo in November 2006 [1]. The console can be seen in Figure 2.1.

Figure 2.1: Nintendo Wii Motion Sensing Video Game Console

The system involves a specially designed tangible motion controller, which is called Wii-mote that enables the user to control games via performing motions such as swinging, tilting and lining in front of the stationary sensor bar component that is placed on top of or at the bottom of the TV. Motion sensor bar of Nintendo Wii can be seen in Figure 2.2.

Figure 2.2: Nintendo Wii Motion Sensor Bar

The console is considered to revolutionize the way that video games are played by incorporating more parts of the body other than fingers into the gameplay with various actions performed by the player. Motion control is achieved via Wii-mote, in which an accelerometer exists that gives the linear acceleration of the player in three axes [37, 38]. Position of the player is tracked by the motion sensor bar, which sends infrared (IR) light to the camera that is located at the tip of Wii-mote with the aim of detecting in-between distance. An additional wired motion controller, which is called Nunchuck, may be plugged into the bottom of Wii-mote to be used in tandem at the accompanying hand. Nunchuck also hosts an accelerometer inside that gives the linear acceleration of the player in three axes, as in Wii-mote. Both motion controllers of Nintendo Wii video game console can be seen in Figure 2.3. At the right side of the figure, Wii-mote is presented and at the left side, Nunchuck that is plugged into Wii-mote can be seen.

Figure 2.3: Nintendo Wii Motion Controllers Left: Nunchuck Right: Wii-mote

In 2009, Nintendo released an expansion to the Wii-mote, which is called Wii MotionPlus that is pluggable to the bottom of the standard Wii-mote controller. Wii MotionPlus hosts a gyroscope inside that provides three dimensional (3D) angular motion data [15]. Wii MotionPlus can be seen in Figure 2.4.

Figure 2.4: Nintendo Wii MotionPlus Expansion of Wii-mote

2.4.2. Sony® PlayStation Move™

PlayStation Move is the motion sensing video game add-on of PlayStation 3 that is released by Sony Computer Entertainment in September 2010 [2]. The device involves a specially designed hand-held tangible controller, which is called PlayStation Move motion controller that enables the user to control games via performing actions such as swinging and tilting in front of the PlayStation eye camera that is placed on top of or at the bottom of the television. PlayStation Move motion controller can be seen below in Figure 2.5.

Figure 2.5: PlayStation Move Motion Controller

There exists an orb at the tip of the PlayStation Move motion controller as can be seen in Figure 2.5 that is able to glow in different colors, which are chosen dynamically by the system as being unique considering the environmental background colors, to enable color tracking for position determination [15]. PlayStation Move motion controller hosts an accelerometer and a gyroscope inside that gives the linear acceleration of the player in three axes and provides three dimensional angular motion data respectively. An additional wireless motion controller, which is called PlayStation Move navigation controller, may be used in tandem at the accompanying hand to give directional or action commands via buttons pressed and the analog stick that is rotated. PlayStation Move navigation controller can be seen below in Figure 2.6.

Figure 2.6: PlayStation Move Navigation Controller

2.4.3. Microsoft® Kinect™

Kinect is the motion sensing video game add-on of Xbox360 console that is released by Microsoft in November 2010 [3]. Microsoft Kinect for Xbox360 can be seen below in Figure 2.7.

Figure 2.7: Microsoft Kinect for Xbox360

It is stated by Microsoft that the device changed the way players control games, since it responds to body movements of the players as game inputs rather than requiring the usage of conventional tangible controllers [3]. The experience is also unofficially termed by the company as full body gaming since Kinect tracks body movements of the players and enables them to control games without using any tangible hand-held controllers, in other words turning the players into controllers. Kinect hosts an internal red green blue (RGB) camera, a depth sensor and four microphones inside [15]. Depth sensor sends out IR rays and reads them back to detect 3D depth data of the 640x480 pixels scene. IR rays that are cast by Kinect can be seen below in Figure 2.8 which presents a photograph that is taken using the night vision setting of a digital camera.

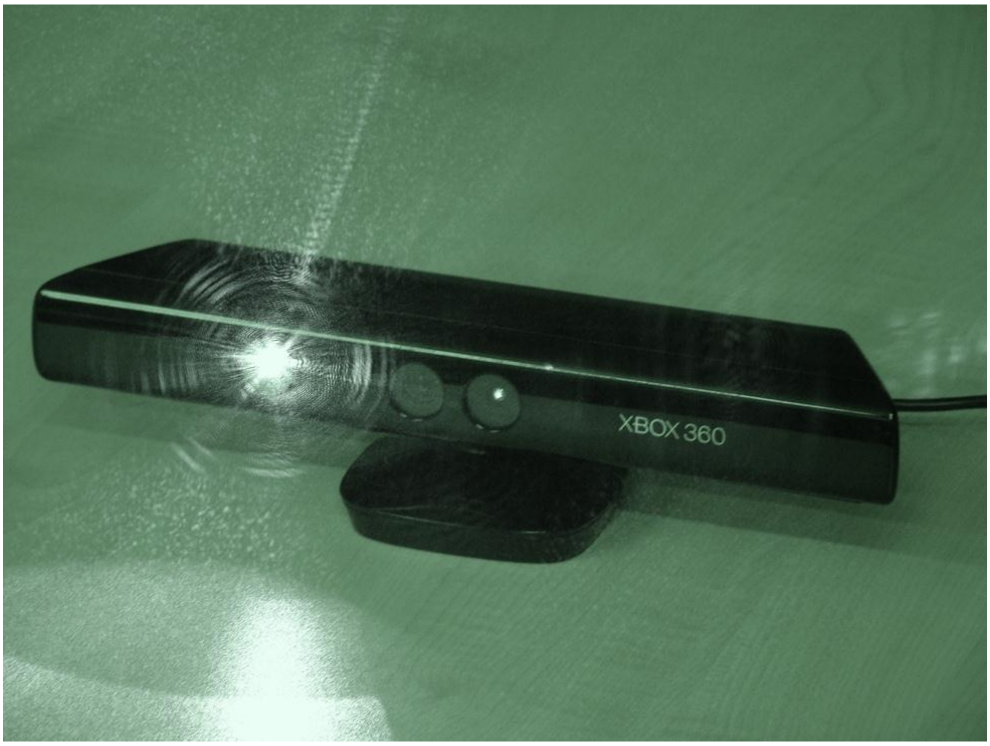


Figure 2.8: IR Rays Cast by Microsoft Kinect for Xbox360

The IR rays that are cast by the device hits to people and/or the objects in the environment and enables depth recognition via measuring the distance from the sensor to the hit point. Infrared rays that are scattered from the depth camera of Kinect impinging of the arm can be seen below in Figure 2.9.

Figure 2.9: Infrared Rays Scattered from Depth Camera Impinging on the Arm

Since the device senses depth information via IR rays, it is able to operate successfully at dark environments, but sunlight is interference to Kinect that affects its understanding of depth information negatively.

Besides recognizing 3D depth information, Kinect is also able to track user skeleton with the help of an algorithm that works at the background. Traced skeleton of a user sketched over depth image stream, provided by Kinect Explorer which is a sample application of the Kinect Software Development Kit (SDK) of Microsoft [3] is presented below in Figure 2.10.

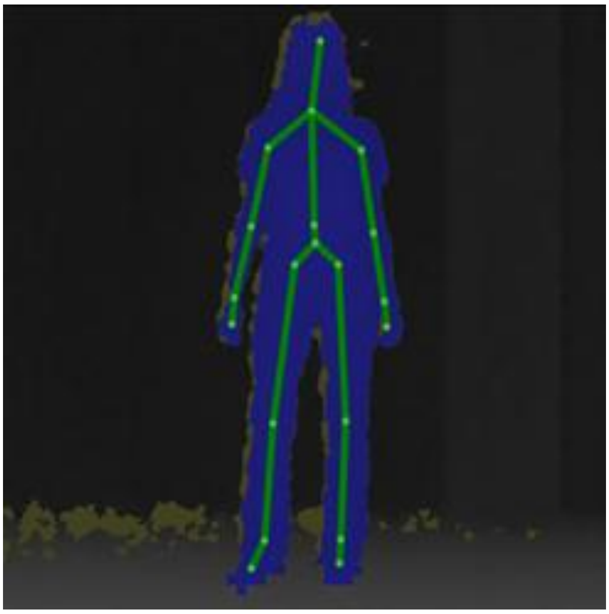


Figure 2.10: Depth Image Stream and Skeleton Tracked by Kinect

As it can be realized from the green circles at the joint intersections in Figure 2.10, Kinect provides positional data of the 20 body joints. These joints with their self-explanatory labels are presented in Figure 2.11.

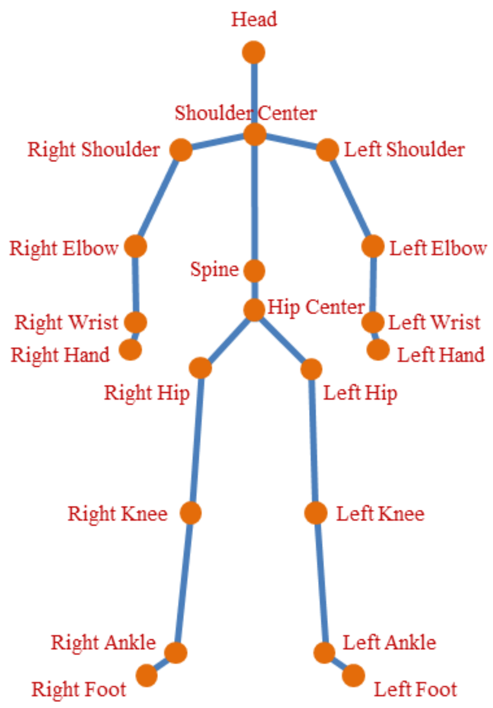


Figure 2.11: Joint Distribution of the Skeleton Tracked by Kinect

Background working algorithm of the device that enables skeleton tracking is discussed thoroughly in a research conducted by Shotton et al. that introduces a new accurate real time method of predicting body joints' 3D positions from only a depth image without using any temporal information [39]. The work is stated by the authors to be used in the underlying core skeleton tracking technology of Microsoft Kinect for Xbox 360. In the research, every pixel on an input depth image is classified as a body part with an assigned probability using pre-built decision trees. Pixels in these body parts are then weighted to decide joint positions to be proposed and body parts are guessed. Research provides a sound basis in understanding the underlying technical working principle of Microsoft Kinect for Xbox 360 on skeleton tracking.

In the currently existing motion tracking pipeline of Kinect, as the players grab something in their hands, skeleton recognition do not consider the hand-held object since the underlying skeleton tracking technology is not developed with this consideration. As Kinect detects a player in front of it, the underlying algorithm compares the human with thousands of pre-generated training images involving real human poses and the skeleton is determined from the nearest image. Since the images are generated using human figures, an anomaly of carrying 30 centimeters length club in hand is ignored by the algorithm, and the best fitting human structure is assigned, ignoring the objects. Tracked skeletons ignoring the presence of a hand-held object of plastic toy sword by the current Kinect motion tracking algorithm can be seen in Figure 2.12, which is generated using sample application Kinect Explorer of Kinect's official SDK [3].

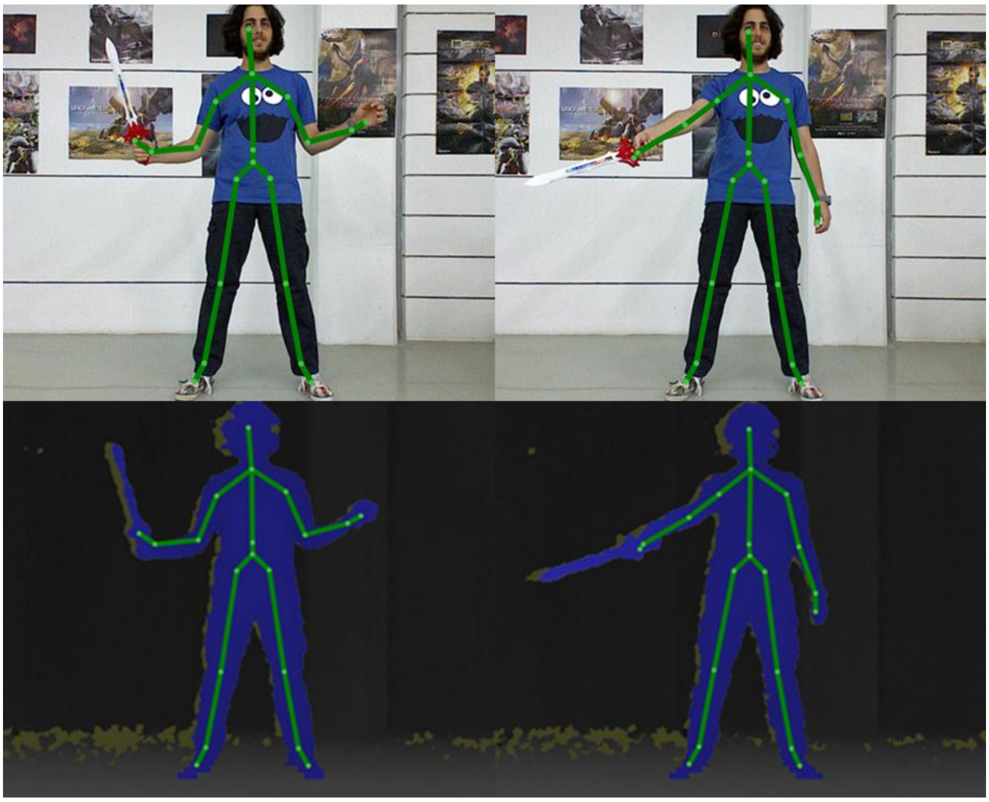


Figure 2.12: Hand-Held Object Ignored by Current Skeleton Tracking Scope of Microsoft Kinect

A recent endeavor to enable the players play Kinect games with hand-held objects is made by a commercial company named Intec [40]. An accessory pack, called Kinect Sports Pack is sold by the company as an effort to fill the lacking caused by the absence of feeling any tangible objects at hand while controlling Kinect games touchlessly. The commercial pack can be seen in Figure 2.13. The pack consists of plastic objects of a driving wheel, a golf club, a baseball bat, a tennis racket and a sword.

Figure 2.13: “Kinect Sports Pack” that is produced by Intec

- Image Adopted from [40]

Advantage of the commercial accessory pack is its being cheap, since it’s composed of foam and plastic, without involving any special electronic equipment inside. The disadvantage is that there is no software adjustment made for Kinect to understand the product that is sold, so it is no different than bare hand playing in the aspect of gameplay, since Kinect does not recognize hand-held objects. Pack sales resulted in some negative user reviews about frequently lost hand tracking and sensing, misbalanced inconsistent behaviors of the objects, and problems on projecting desired actions using the objects into games which are predictable complaints considering the fact that Kinect is not designed to understand any hand-held object but is designed for controllerless motion controlling [41]. On the other hand, users stated that it was a better and more realistic experience to actually hold something than barely using their hands for controlling. A user stated that he enjoyed playing Fruit Ninja Kinect by holding a physical sword more than playing bare hands despite the problems he encountered in reflecting his motions into the game while holding the sword.

Since there is no software adjustment done, even if the user is able to play Kinect games with these accessories, motions of the objects are not recognized by current system, Kinect continues to track only human skeleton motions whether he is holding an object or not. Since the system ignores any held objects, dimensions of these objects are also not recognized in the current motion tracking scope of the tool, again making it no different than playing games bare hand.

Despite the negative user reviews, these reviews on the other hand may be interpreted as a potential for the contribution of object recognition and gameplay adjustment according to the recognized object's dimensions to the Kinect gaming experience.

CHAPTER 3

PROPOSED APPROACH

This chapter presents the proposed approach that is developed, implemented and evaluated throughout the thesis to achieve an improved motion controlled video game interaction method with tangible objects using Microsoft Kinect. Considering the drawbacks of touchless interaction that are presented in the previous chapter as a motivation, handheld tangible objects are introduced into motion controlling in this thesis to achieve an improved gameplay.

First, problem statement is presented that defines and clarifies the problem which this thesis aims at solving. Then, object detection algorithm that is developed to include handheld tangible objects into gameplay is explained step by step to give the reader a clear understanding of the developed method. Afterwards, technical improvements that are achieved by the developed algorithm are presented. Limitations of the developed algorithm are then discussed and the chapter concludes with the implementation of the algorithm supported with provided code excerpts. In the following chapter, evaluation of the developed algorithm to measure its effects on gameplay experience is presented.

3.1. Problem Statement

Problem to be solved in this thesis can be identified as: Introducing hand-held tangible objects into the scope of motion tracking using Microsoft Kinect to provide an improved gaming experience by including touching sense in motion controlled video game interaction.

3.2. Object Detection Algorithm

In this subsection, the algorithm that is employed to detect the presence of an object in the player's hand and to incorporate the detected object into motion controlled gameplay is explained step by step in detail. Skeleton of the player is recognized using data gathered from Microsoft Kinect for Windows SDK, then it is checked whether the player holds an object in the hand or not; if so, motions of the player that are performed using the hand-held object are incorporated into gameplay following the steps that are explained below.

The global coordinates and local coordinates of the hand-held object that are used throughout the algorithm are illustrated in Figure 3.1. In the figure, a representative player is holding an object in his right hand and local coordinates of the carried object are shown in green on itself. In this coordinate system, local x axis is pointing towards tip of the object, local y axis is pointing towards the right hand palm and local z axis is pointing towards bottom part of the fingers of the grasped right hand. Global coordinates are shown in red, at the bottom side of the figure. In this coordinate system, global x axis is pointing towards right of the player, global y axis is pointing towards top of the player and global z axis is pointing towards front of the player.

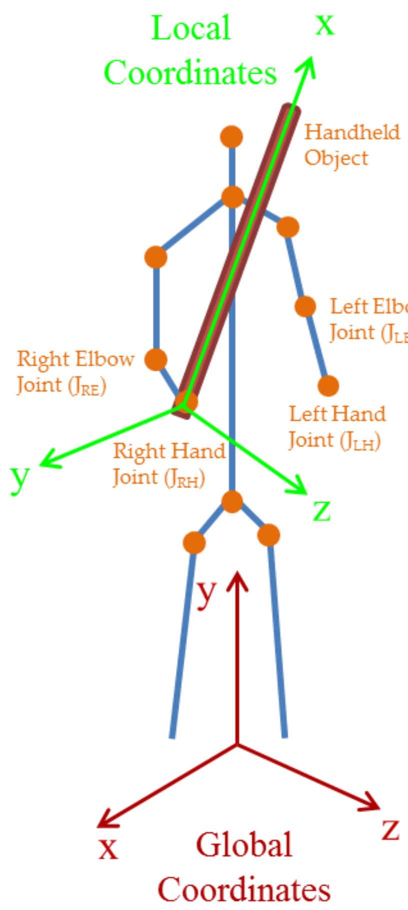


Figure 3.1: Global and Local Coordinates that are Used Throughout the Algorithm

Step 1: Presence of the player in front of Kinect is detected using Microsoft Kinect for Windows SDK.

Step 2: Position of the right hand joint in 3D world coordinates (WJ_{RH}) is taken from the skeletal data that is provided by Microsoft Kinect for Windows SDK.

Step 3: WJ_{RH} is converted from 3D world coordinates to two dimensional (2D) depth coordinates (DJ_{RH}) for being able to work with the 2D depth data stream provided by Microsoft Kinect for Windows SDK as from this step.

Step 4: Depth data of the right hand joint (d_{RH}) is found by gathering the 2D depth value corresponding to DJ_{RH} from depth stream provided by Microsoft Kinect for Windows SDK.

Step 5: A circle having radius $r_{n,m}$ where n is the number of iteration in the search set and m is the search set number of possible values 1 and 2, is initialized by Equation 3.1 that is stated below to be able to perform a linear search on the pixels that lies on the circle, looking for the depth value.

$$r_{1,1} = 134.43 - 0.029d_{RH} \qquad \text{(Equation 3.1)}$$

where

$r_{1,1}$ = Radius of the first search set's first iteration circle in pixels

d_{RH} = Depth data of the right hand joint in millimeters

Equation 3.1 is derived from Equation 3.2 that is presented below.

$$r(d) = \frac{r(d_{max}) - r(d_{min})}{d_{max} - d_{min}} (d - d_{min}) + r(d_{min}) \qquad \text{(Equation 3.2)}$$

where

$r(d_{max}) = lr\ (Sh\ /\ l_{max})$

$r(d_{min}) = lr\ (Sh\ /\ l_{min})$

where

$l_{max} = Tan\ (V/2)2d_{max}$

$l_{min} = Tan\ (V/2)2d_{min}$

where

d = Depth of the center point on which search circle will be aligned in millimeters

$r(d)$ = Search circle radius for depth d in pixels

d_{max} = Maximum depth range of the sensor in millimeters

d_{min} = Minimum depth range of the sensor in millimeters

$r(d_{max})$ = Search circle radius for d_{max} in pixels

$r(d_{min})$ = Search circle radius for d_{min} in pixels

lr = Desired real world search circle radius in millimeters

Sh = Sensor resolution height in pixels

l_{max} = Height seen by the sensor at d_{max} in millimeters

l_{min} = Height seen by the sensor at d_{min} in millimeters

V = Vertical angle of Kinect vision in degrees

To provide a better understanding of how Equation 3.1 is derived from Equation 3.2, a detailed explanation is provided as follows. Angle of Kinect vision is 43.5 degrees in vertical axis and the device is stated to have 1.2 meters to 3.5 meters depth range [42]. In the algorithm, it is assumed that the object that is held is greater than 100 millimeters in length since smaller length objects may interfere with the hand joint data and yield to unreliable results. Such small sized objects will not serve as extensions to body joints hence not providing additional movement capability to the player, so it will not be different than the current motion controlling whether the arm is swung with the hand in the form of a fist or carrying a small object at hand. If this assumption is to be removed, following calculations need to be adjusted accordingly.

An initial guess for the circle radius is determined as 200 millimeters in real world, which is found by optimization. But this radius has to be adjusted according to the distance of the user to Kinect since the device's camera view is perspective. Camera view of Kinect is illustrated in Figure 3.2 below. From the figure it can be observed that as the player stands nearer to Kinect, he occupies more area of the camera's field of view and as he moves away, he occupies relatively less area, since the camera has a perspective view.

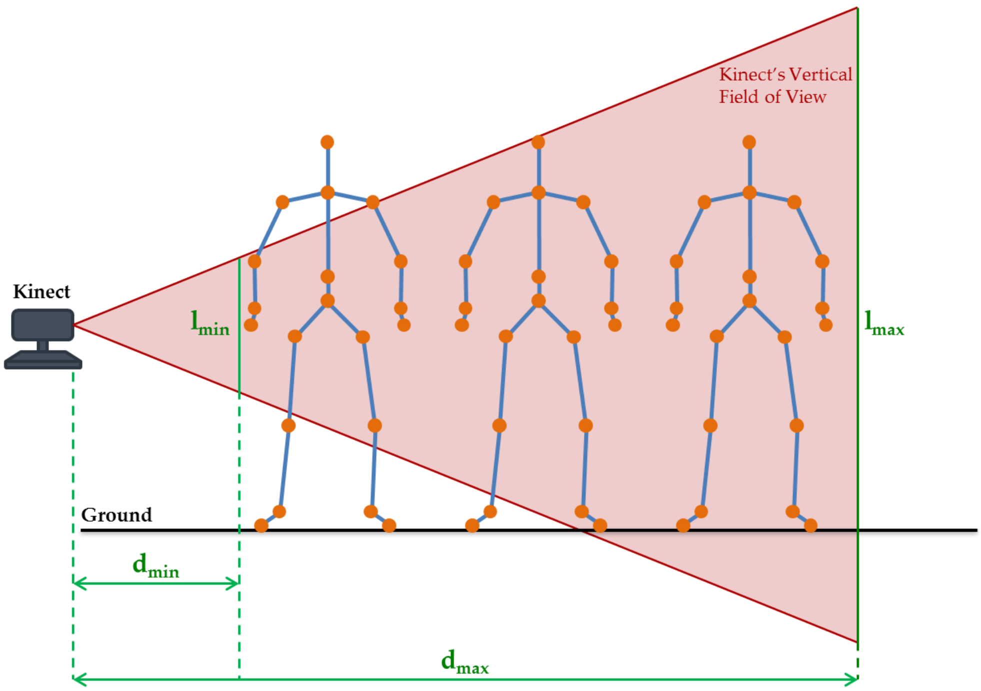


Figure 3.2: Visualization of Kinect Field of View with Player Illustrations Standing at Different Distances from the Sensor

The further the player is detected to be standing from the sensor, the smaller the diameter of the circle has to be assigned and controversially the nearer the player is detected to be standing from the sensor, the bigger the diameter of the circle has to be assigned proportionally to prevent missing the presence of an object that would lay inside a large diameter circle without being detected in the depth search. The situation can be observed in Figure 3.3, in which the player stands at different distances from Kinect. Upper side of the figure shows RGB images captured by the color camera of Kinect and bottom side shows depth stream images. At the leftmost side of the figure, the player stands nearest to the sensor and he moves back gradually in the following shots rightwards. In the upper RGB images, there exist red circles representing the search circles of the same diameter. As the player gets farther from the sensor, red circles miss the presence of the object at hand; on the other hand there exist green circles of shrinking diameter in the RGB image which intersect with the carried object in all four conditions. In order to detect the carried object accurately from different distances, diameter of the circle needs to be adjusted in correlation with the player's distance from the sensor.

Figure 3.3: Player Occupying Different Proportions of the Screen while Standing at Different Distances from the Sensor

Maximum height that can be seen by Kinect camera at the near depth edge limit (d_{min}) which is found to be 1200mm from Kinect's technical capabilities, is found as follows:

1200mm × Tan (21.75 degrees) × 2 = 957mm

Since Kinect's screen resolution is 640×480 pixels, as 957mm refers to 480 pixels height of depth resolution, height pixels that a 200mm object occupies at the near edge limit of Kinect camera is found to be 200mm × 480 pixels ⁄ 957mm = 100 pixels, giving the relation between depth and search circle radius r(1200mm) = 100 pixels.

Maximum height that can be seen by Kinect camera at the far depth edge limit (d_{max}) which is 3500mm from Kinect's technical capabilities, is found as follows:

3500mm × Tan (21.75 degrees) × 2 = 2792mm

As 2792mm refers to 480 pixels height of depth resolution since screen resolution of Kinect is 640×480 pixels, height pixels that a 200mm object occupies at the far edge limit of Kinect camera is found to be 200mm × 480 pixels ⁄

2792mm = 34 pixels, giving the relation between depth and search circle radius r(3500mm) = 34 pixels.

Using the previously found two relations r(1200mm) = 100 pixels and r(3500mm) = 34 pixels, correlation between depth and search circle radius is found to be as follows in the Equation 3.3.

$$r(d) = 134.43 - 0.029 \times d \qquad \text{(Equation 3.3)}$$

where

d = Depth of the center point on which search circle will be aligned in millimeters

r(d) = Search circle radius for depth d in pixels

Step 6: Calculations are done for a maximum of 60 degrees lean of the sword in local z axis. In the iteration, if a depth value between $d_{RH} \pm r_{1,1} / \tan 30$ is detected, it is understood that an additional object is present in the hand of the user, the algorithm processes to Step 7. If no depth value between the targeted range is detected, $r_{1,1}$ is updated with the following formula presented in Equation 3.4 where n is the number of iteration in the search set.

$$r_{n+1,1} = r_{n,1} \times 0.9 \qquad \text{(Equation 3.4)}$$

where

$r_{n+1,1}$ = Radius of the first search set's (n+1)th iteration circle in pixels

$r_{n,1}$ = Radius of the first search set's nth iteration circle in pixels

And following iterations are performed until the first edge point on the object $e_{1,1}$ is detected or $r_{n,1} < r_{1,1} \times 0.5$ since hand data is beginning to interfere as the initial radius is shrunk more than 0.5 of the initial value.

In this search, forearm of the player (from elbow joint to wrist joint) is ignored since it will have an irregular depth data compared to the rest of the background scene that would be detected in the depth search and misinterpreted as an object's presence otherwise. This is ensured by performing a check in which ± 15 degrees of the slope of 2D points DJ_{RE} and DJ_{RH} is ignored in the search, since this interval contains the forearm of the player.

Step 7: After $e_{1,1}$ is found, a linear search is performed on the circle in which depth value of the current pixel is compared with the previous one to detect a drop in the depth meaning that the opposite edge of the object in local y axis

$e_{1,2}$ is reached. The iteration continues as long as the condition $d_{P1} - \varepsilon < d_{x,y} < d_{P1} + \varepsilon$ is satisfied, where ε is a constant representing the threshold after which the depth difference indicates that the object has ended. In this thesis, ε is chosen to be 50 which is found by optimization, assuming that depth difference between two consecutive pixels on the object cannot be more that 50mm in any orientation of the object. This constant may be chosen to be different than 50 for other usage purposes, keeping in mind that as ε is decreased, there may be false alarms of depth fall as the object is held rotated in local x axis and as ε is increased, there may be false alarms of depth fall as the player holds his hand carrying the object at a nearer distance to his body than the chosen threshold.

A summary of the algorithm steps that are performed so far is illustrated in Figure 3.4. In the figure, three iterations are performed to detect the presence of the object. In the first and second iterations, the object is not detected since it is held at a skewed orientation at its local z axis. In the third iteration at which the search circle has a smaller radius, the object is recognized and two edge points $e_{1,1}$ and $e_{1,2}$ are detected.

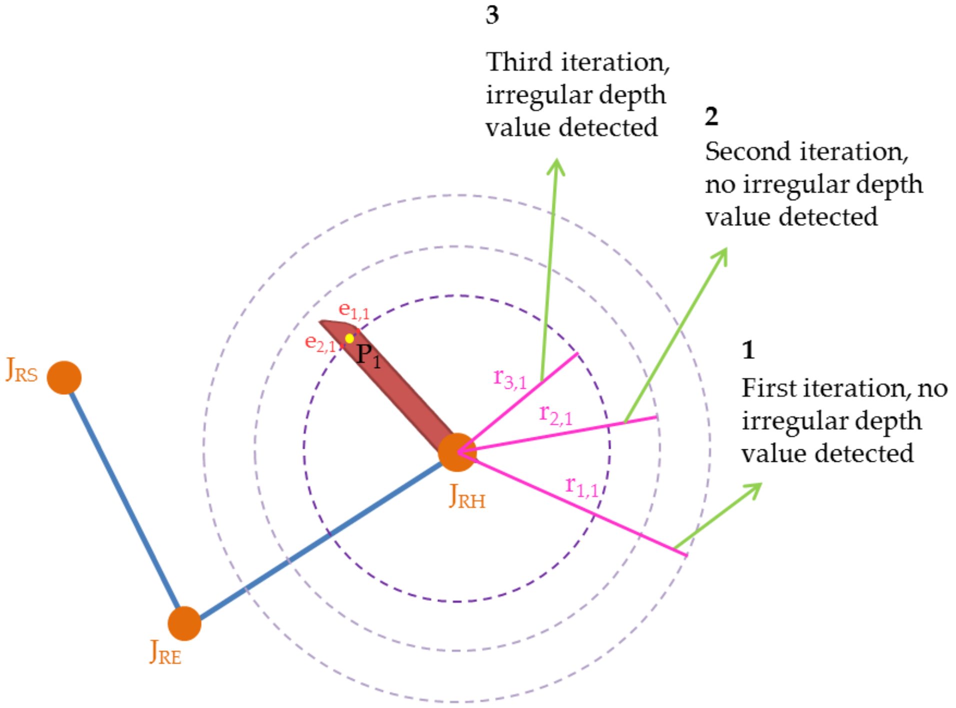


Figure 3.4: Finding Two Edge Points of the Hand-Held Object with Linear Searches

Step 8: Midpoint of the two edge points of the object $e_{1,1}$ and $e_{1,2}$ is converted to world coordinates and stored as the first point on the object (P_1).

Determination process of P_1 is illustrated in summary in Figure 3.5. In the figure three levels are illustrated, showing the linear search performed on the circle to detect any irregular depth values while ignoring body parts, and assigning the first point on the object as the midpoint of object's two edge points.

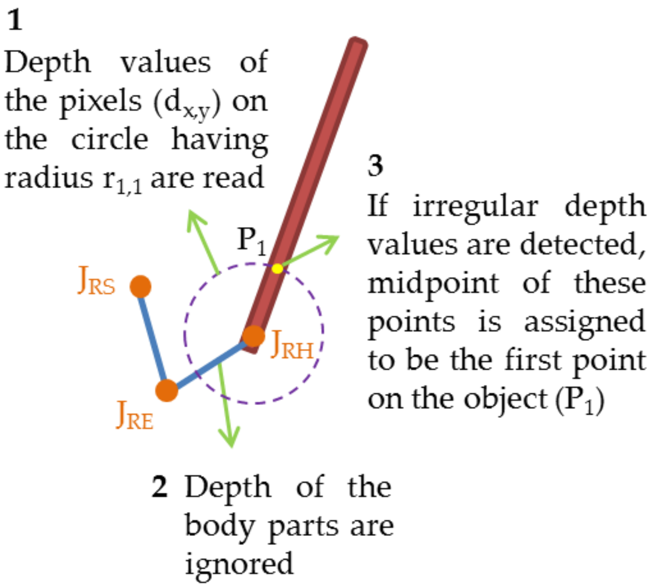


Figure 3.5: First Point of the Object P_1 is Determined

Step 9: A second circle having radius r_2 determined by the formula stated below in Equation 3.5 is defined to be able to perform a depth value search on the pixels lying on it. In Equation 3.5, n is the number of search iteration on which $e_{1,1}$ was found in Step 6.

$$r_2 = r_{n,1} \times 0.66 \quad \text{(Equation 3.5)}$$

where

r_2 = Radius of the second search set's circle in pixels

$r_{n,1}$ = Radius of the first search set's n^{th} iteration circle in pixels

Depth of the body parts are ignored in this second search too, similar to being explained in Step 6.

Step 10: Procedure that is explained in Step 6 is performed for the circle having radius r_2 to find $e_{2,1}$ only with the difference that no iteration is performed since a smaller radius circle is checked in this second search, hence there is no need to shrink the diameter gradually since the search is performed in an area that is essentially smaller than the area that already contains a point belonging to the object inside. It can be stated that if a part of the hand-held object lies on the circle having radius r_1, then some part of the object is expected to lie on the circle having radius r_2, where $r_2 < r_1$, assuming that the hand-held tangible object is continuous, rigid and one parted.

Step 11: Procedure that is explained in Step 7 is performed to find the second edge point of the second search set on the object $e_{2,2}$.

Step 12: Procedure that is explained in Step 8 is performed with $e_{2,1}$ and $e_{2,2}$ to assign the second point lying on the object (P_2).

Step 13: Direction of the object ($\vec{d_o}$) is found using P_1 and P_2, as being the 2D vector that is defined from P_2 to P_1.

Algorithm steps from 9 to 13 are summarized in Figure 3.6 to provide a clear understanding. In the figure, three levels are presented first one illustrating the second search that is performed on the circle having radius r_2, second one showing the assignment of the second point on the object and third one illustrating the determination of the object direction.

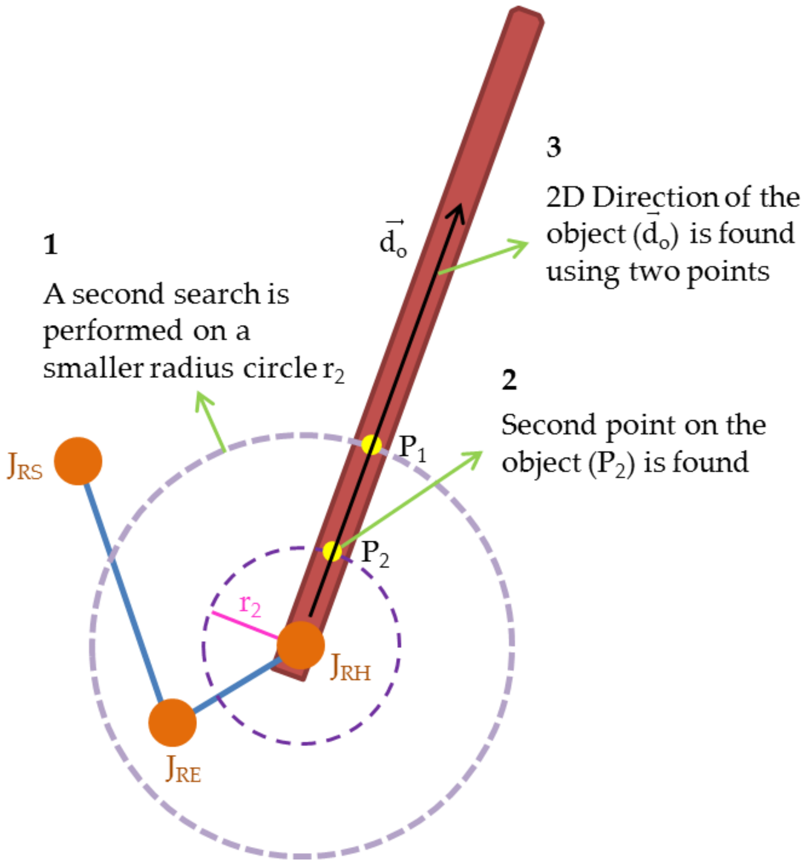


Figure 3.6: A Visualized Summary of Algorithm Steps 9 to 13

As an option, instead of finding two points on the object to determine its direction, one point on the object and the already known hand joint data could have been used to determine the object's orientation but it yielded unreliable results and for the sake of accuracy, two points on the object are preferred to be found in this thesis. Hand joint orientation is not always 100% accurately determined by Kinect SDK's skeletal tracking since body joint positions are predicted using trained data and may be given slightly different from their actual locations due to the possible errors in these predictions. So, to achieve an accurate recognition, finding two points on the object is more effective although it incurs some calculation costs. The situation is illustrated in Figure 3.7. In the figure, if the hand joint data is given to be at a slightly different position that the actual, object direction can be misinterpreted dramatically that would affect whole motion recognition negatively. Using two points on the object to determine the direction eliminates this possible inaccuracy.

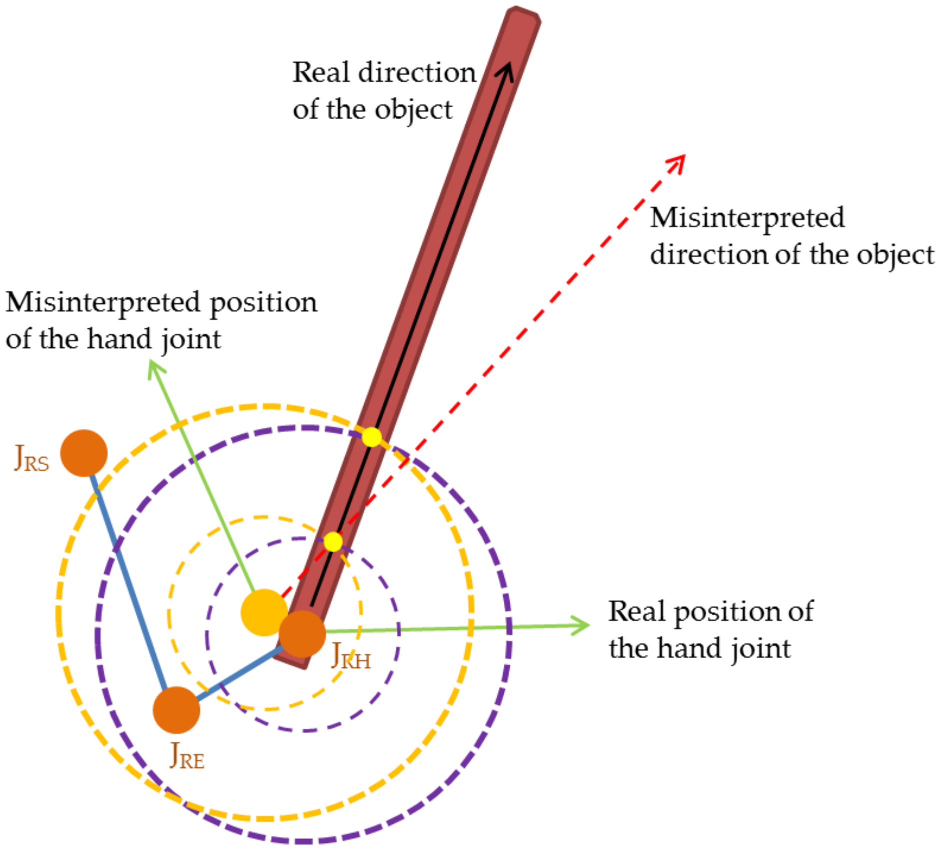


Figure 3.7: Misinterpreted Object Direction Resulted from Using One Point on the Object and Hand Joint Data to Find the Direction

Accuracy of the object direction is important since it may cause a serious degrade in motion recognition of the object if not detected accurately. The player may hold the object in different orientations that are needed to be projected into gameplay for the sake of realism and richness of possible movements. In Figure 3.8, the player holds a plastic toy sword at his hand which is leaned to front in global x axis, held perpendicularly and leaned to back in global x axis consecutively from left to right. Upper side of the figure presents RGB images and lower side of the figure presents depth images on which a point lying on the object and tip point are shown with green dots and direction of the object is shown with a red line.

Figure 3.8: Player Leans the Sword Forth and Back Left: Player Leans the Sword Forth Middle: Player Holds the Sword Perpendicularly Right: Player Leans the Sword Back

Step 14: A linear search is performed in $\vec{d_o}$ to find the tip point of the object in depth coordinates (DT_o), in which depth value of the searched pixel is compared with the previous one to detect a drop in the depth value meaning that the tip point of the object is reached. The iteration continues as long as the condition $d_{P1} - \varepsilon < d_{x,y} < d_{P1} + \varepsilon$ is satisfied, where ε is chosen to be 50 as explained in Step 7.

The reason that linear search is implemented is that the depth distribution of the object can also be found using this method, which can be used as an estimate for the object's shape that can be incorporated into gameplay.

Step 15: DT_o is converted to 3D world coordinates (WT_o) from 2D depth coordinates.

Step 16: Height of the object (h_o) is found by using WJ_{RH} and WT_o using the formula presented below in Equation 3.6.

$$h_o = | W_{T0} - WJ_{RH} | \qquad \text{(Equation 3.6)}$$

where

h_o = Height of the object

W_{T0} = Tip point of the object in 3D world coordinates

WJ_{RH} = Position of the right hand joint in 3D world coordinates

Algorithm steps 14 to 16 are illustrated in the following Figure 3.9. In the figure, there are three steps, first of which illustrates the binary search that is performed in the object's direction, second one shows the tip point of the object that is found with the binary search performed and the third one illustrating the height of the object being determined.

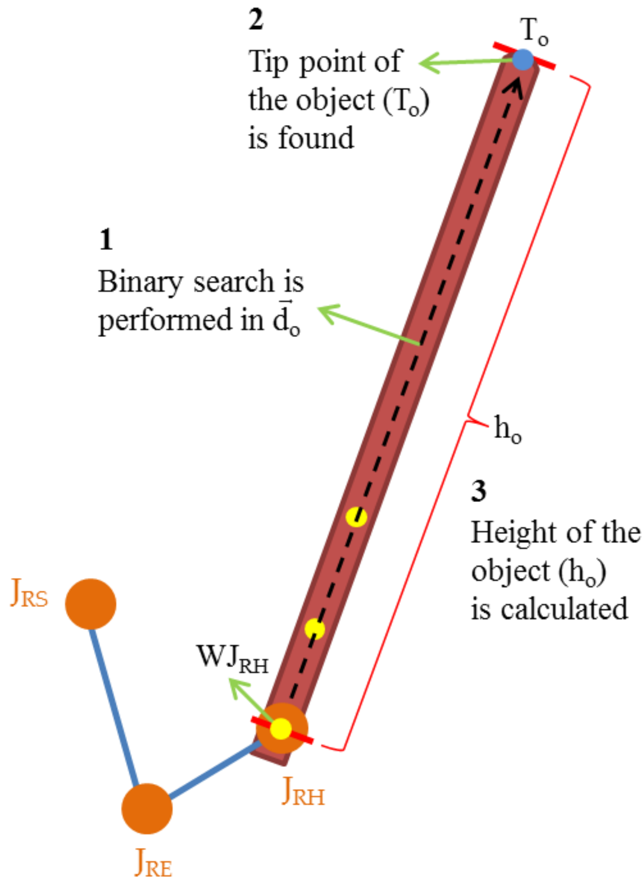


Figure 3.9: A Visualized Summary of Algorithm Steps 14 to 16

Step 17: This algorithm is performed at 30 frames per second (fps), synchronous with Kinect's own fps and the object is incorporated into the gameplay via placing the virtual sword to the hand joint of the avatar of the player in the detected direction and projecting every movement of the sword into gameplay.

The algorithm that is explained above is for the right hand tracking of the hand-held object. To detect the left hand, all steps should be performed the same except for taking left hand's data from Microsoft Kinect for Windows SDK and performing the analyses considering the left hand instead of the right hand. If the algorithm is desired to be employed to detect multiple objects that are held at both hands of the player, it needs to be run consecutively for each hand.

3.3. Technical Improvements

In this subsection, technical improvements that are achieved by the developed algorithm which enables motion controlling with object recognition are presented and discussed under relevant subsections.

3.3.1. Object Recognition

The developed algorithm provides hand-held object recognition using Microsoft Kinect with its orientation and dimensions, which is ready to be incorporated into gameplay. This is the major improvement that is achieved since tangible interaction has many advantages that are mentioned in Section 2.2. The algorithm is able to detect different objects since it is independent of any specific one. Different object the player is handing that are detected by the algorithm are presented in Figure 3.10. Upper side of the figure shows RGB images and bottom side shows depth images in which a point detected on the object P_2 and the tip point DT_o in depth coordinates are shown with green dots, and the direction is shown with a red line. At the left side of the figure, the player is holding a gray plastic toy sword. At the middle of the figure, the player is holding a milk bottle in his hand. At the left side of the figure, the player is holding another plastic toy sword which has different form and color than the one at the left side. The algorithm successfully detects these objects and determines their orientation.

Figure 3.10: Player Carrying Different Objects at Hand Left: Player is Holding a Plastic Toy Sword Middle: Player is Holding a Milk Bottle Right: Player is Holding another Plastic Toy Sword of Different Form

3.3.2. Indirect Rolling Motion Recognition

An additional improvement that is achieved by the developed algorithm is the addition of rolling motion recognition of the hand as an object is carried. Skeleton tracking support of Kinect is limited in rolling axis of rotation meaning that rolling actions of the limbs are not recognized by the sensors [43]. Although rolling motion of body joints are not recognized by the developed algorithm, recognition of rolling action of the wrist is made possible if the user holds an object since the object will serve as an additional joint added to the skeleton. An example is shown in Figure 3.11, in which the player holds a plastic toy sword at his hand rotated in rolling axis to his right side in global z axis, held perpendicularly and rotated in rolling axis to his left side in global z axis consecutively from left to right. Upper side of the figure shows RGB images and bottom side shows depth images in which a point detected on the object P_2 and the tip point DT_o in depth coordinates are shown with green dots, and the direction is shown with a red line. If the player did not hold any object in his hand and performed the same actions shown in the figure, the rotational motion would not be detected by the current motion tracking provided by Kinect, but made possible to be recognized with the developed algorithm as the player holds an object in the hand.

Figure 3.11: Player Rotates the Sword in Rolling Axis of Motion Left: Player Leans the Sword Right in Global z Axis Middle: Player Holds the Sword Perpendicularly Right: Player Leans the Sword Left in Global z Axis

3.4. Limitations

Limitations of the developed algorithm that should be considered during employment of the proposed method are discussed in this subsection.

Implementation is done for the right hand interaction but using the developed algorithm, it is possible to recognize the hand-held object in the left hand either by changing the right hand data with left hand. Although object dimension is recognized with the algorithm developed, gameplay is not adjusted according to the object dimensions currently, whether the player uses a 20 centimeters length toy of 50 centimeters, virtual sword in the game makes the same movements. The object that will be used for interaction should not be too thin since Kinect may not able to recognize it if depth rays that are sent miss it. The tangible object should not be transparent; otherwise Kinect will not recognize its presence since depth rays that are cast will pass through it. It is assumed that the object held is greater than 100 millimeters in length since smaller length objects do not provide considerable extensions to the body and it is not much different that playing with bare hands. Rotation of the hand-held sword in its local x axis is not recognized with the developed algorithm as a limitation.

3.5. Implementation

In this subsection, implementation of the developed algorithm in an experimental game is presented with the discussion of game design, explaining the aspects that are paid special attention during design process and providing a neat game overview with screenshots and images, and game development presenting code excerpts from the characteristic operations of the algorithm to provide a clear understanding of the implementation.

To implement the game that is designed, Unity3 game engine is used with C# scripting language [44]. Kinect data is obtained from Microsoft Kinect for Windows SDK. But between the two development mediums, there exist a communication incompatibility; Microsoft Kinect for Windows SDK requires .NET Framework 4.0 [45] while Unity3 supports up to .NET 3.5 [46]. To overcome this communication problem, Zigfu for Unity3D is used as a wrapper which connects Unity3 with Microsoft Kinect for Windows SDK and enables data gathering [47]. As the hardware, a notebook computer of Toshiba Satellite L750 is used of having Intel Core i7-2670QM CPU @ 2.20 GHz, NVIDIA GeForce GT 525M GPU and 4 GB RAM. The implemented game runs interactively at 60 fps on the computer.

In the following subsections, code excerpts of the important characteristic operations of the algorithm are presented with their explanatory comments for further understanding of the implementation. All code excerpts that are presented throughout the thesis are written in C# programming language.

3.5.1. Finding First Point on the Object

Code excerpts that are presented in this subsection performs as a whole finding the first point that lies on the hand-held object corresponding to Step 5, Step 6, Step 7 and Step 8 of the object detection algorithm that is discussed in Section 3.2.

Code excerpt that is presented below in Figure 3.12 performs variable initializations, arm direction determination to be excluded from depth detection search later on and right hand joint's position data gathering from Microsoft Kinect for Windows SDK.

```
//Initialization of the variables
first_point_found = false;
second_point_found = false;
first_angle1 = 0;
second_angle1 = 0;

//Determination of the arm direction
arm_direction = Mathf.Atan2 ( RightElbowPixel.y -  RightHandPixel.y,
            RightElbowPixel.x - RightHandPixel.x ) * ( 180.0 / Mathf.PI );

//Gathering right hand data
RightHandDepth = Depth.data[RightHandIndex];
```

Figure 3.12: Variable Initialization, Arm Direction Determination and Right Hand Joint Data Gathering

Code excerpt that is presented below in Figure 3.13 performs guessing first assigned radius of the search circle and comparing depth values of the pixels lying on the circle with the depth value of hand joint to detect the first edge point lying on the object. Pixels lying on the arm direction that is determined previously in Figure 3.12 are excluded from the search.

```
//First guess for the radius of the first search circle
radius1 = ( 134.43f - ( 0.029f ) * RightHandDepth );
int radius_shrink = 0;
for ( int i = 0; i < 360; i++ )
{
/*x and y of the candidate pixel is converted into index representation to find the depth
value in the depth data array of size [640x480]*/
CheckPixel.x = RightHandPixel.x + (radius1*Mathf.Cos(i*Mathf.PI/180.0f));
CheckPixel.y = RightHandPixel.y + (radius1*Mathf.Sin(i*Mathf.PI/180.0f));
CheckPixelIndex = ((Depth.yres-CheckPixel.y)*Depth.xres) + (Depth.xres-CheckPixel.x);
if ( first_point_found == false )
{
//If checked pixel does not lay on the arm of the player, enter inside the below statement
if ( i > arm_direction + 15 || i < arm_direction - 15 )
{
//Depth of the pixel is compared to depth of the hand with a threshold
if ( Depth.data[CheckPixelIndex] < RightHandDepth + depth_variant &&
Depth.data[CheckPixelIndex] > RightHandDepth - depth_variant )
{
//If the depth lies between the thresholds, angle of the first edge point is found
first_angle1 = i;
first_point_found = true;
} }
```

Figure 3.13: Depth Comparison Search Performed to Detect the First Edge Point of the Object

Code excerpt that is presented below in Figure 3.14 performs radius decrease to make the searches continue. The search continues till the radius is shrunk 50% of its original value, which corresponds to 7 iterations performed at most. If the object is not found in 7 iterations, it is concluded that the player does not carry any objects at hand.

```
//If no object is detected, enter inside the below statement
if ( i == 359 )
{
/*Radius is shrunk by 10% until it reaches %50 of the original, corresponding to 7
iterations*/
if ( radius_shrink < 7 )
{
radius1 = radius1 * 0.9f;
i = 0;
radius_shrink++;
}
//If no object is found in 7 iterations, no object is detected at all
else
{
NoSword();
} } }
```

Figure 3.14: Depth Detection Searches Performed with Decreased Radii

Code excerpt that is presented in Figure 3.15 performs finding the opposite edge point of the object. First point on the object is then assigned to be the midpoint of the two previously found edge points that is converted to world coordinates.

```
/*If the first angle is found, circle arc is continued until a drop in depth is examined,
which means the second angle is also found*/
else
{
if ( second_point_found == false)
{
if ( LabelMap.data[CheckPixelIndex] <= 0 || Depth.data[CheckPixelIndex] >
RightHandDepth+depth_variant || Depth.data[CheckPixelIndex] < RightHandDepth-
depth_variant )
{
//If a drop in depth is detected, second point's angle is also found
second_angle1 = i;
second_point_found = true;
} } } }
//Mid angle is found by taking the average
average_angle1 = ( first_angle1 + second_angle1 ) / 2.0f;
/*Using the hand's pixel data and the mid angle, first point lying on the object is found
in 2D depth coordinates*/
Vector2 P1_Depth = new Vector3 ( RightHandPixel.x +
                (radius1*Mathf.Cos(average_angle1*Mathf.PI/180.0f)) ,
                RightHandPixel.y +
                (radius1*Mathf.Sin(average_angle1*Mathf.PI/180.0f)));

//First point on the object is converted to depth array representation to get its depth
value
int P1_Index = ((Depth.yres-P1_Depth.y)*Depth.xres) + (Depth.xres-P1_Depth.x);

/*Depth data is converted into 3D world coordinates to find position of the point in
real world space*/
P1_World = Depth.ConvertImageToWorldSpace ( new Vector3( P1_Depth.x,
        P1_Depth.y, Depth.data[P1_Index] ));
```

Figure 3.15: First Point Lying on the Object is Determined

3.5.2. Finding Second Point on the Object

Code excerpts that are presented below performs as a whole finding the second point that lies on the hand-held object corresponding to Step 9, Step 10, Step 11 and Step 12 of the object detection algorithm that is discussed in Section 3.2.

Code excerpt that is presented below in Figure 3.16 performs assigning radius of the second search circle as 2/3 of the first search circle's radius and variable initialization.

```
//Radius of the second search circle is initialized as 2/3 of the first radius
radius2 = radius1 * 2.0f / 3.0f;

//Variables are initialized
first_point_found = false;
second_point_found = false;
first_angle2 = 0;
second_angle2 = 0;
```

Figure 3.16: Radius Assignment of Second Search Circle and Variable Initialization

Code excerpt that is presented below in Figure 3.17 performs comparing depth values of the pixels lying on the second search circle with the depth value of hand joint to detect the second edge point lying on the object. Pixels lying on the arm direction that is determined previously in Figure 3.12 are again excluded from the search.

```
for ( int i = 0; i < 360; i ++ )
{
/*x and y of the candidate pixel is converted into index representation to find
the depth value in the depth data array of size [640x480]*/
CheckPixel.x = RightHandPixel.x + (radius2*Mathf.Cos(i*Mathf.PI/180.0f));
CheckPixel.y = RightHandPixel.y + (radius2*Mathf.Sin(i*Mathf.PI/180.0f));
CheckPixelIndex = (((int)Depth.yres - (int)CheckPixel.y) * Depth.xres) +
                        ((int)Depth.xres - (int)CheckPixel.x);
if ( first_point_found == false )
{
/*If the checked pixel does not lay on the arm of the player, enter inside the
below statement*/
if ( i > arm_direction + 15 || i < arm_direction - 15 )
{
//Depth of the pixel is compared to depth of the hand with a threshold
if ( Depth.data[CheckPixelIndex] < RightHandDepth + depth_variant &&
Depth.data[CheckPixelIndex] > RightHandDepth - depth_variant )
{
/*If the depth lies between the thresholds, angle of the second edge point is
found*/
first_angle2 = i;
first_point_found = true;
} } }
```

Figure 3.17: Depth Comparison Search Performed to Detect the Second Edge Point of the Object

Code excerpt that is presented below in Figure 3.18 performs finding the opposite edge point of the object. Second point on the object is then assigned to be the midpoint of the two previously found edge points that is converted to world coordinates.

```
/*If the first angle is found, circle arc is continued until a drop in depth is
examined, which means the second angle is also found*/
else
{
if ( second_point_found == false )
{
if (Depth.data[CheckPixelIndex] > RightHandDepth+depth_variant ||
Depth.data[CheckPixelIndex] < RightHandDepth-depth_variant )
{
//If a drop in depth is detected, second point's angle is also found
second_angle2 = i;
second_point_found = true;
} } } }
//Mid angle is found by taking the average
average_angle2 = ( first_angle2 + second_angle2 ) / 2.0f;
/*Using the hand's pixel data and the mid angle, second point lying on the
object is found in 2D depth coordinates*/
Vector2 P2_Depth = new Vector3 (RightHandPixel.x +
   (radius2*Mathf.Cos(average_angle2*Mathf.PI/180.0f)) , RightHandPixel.y
   + (radius2*Mathf.Sin(average_angle2*Mathf.PI/180.0f)));
/*Second point on the object is converted to depth array representation to get
its depth value*/
int P2_Index = ((Depth.yres-P2_Depth.y)*Depth.xres)+(Depth.xres-
P2_Depth.x);
/*Depth data is converted into 3D world coordinates to find the position of
the point in real world space*/
P2_World = Depth.ConvertImageToWorldSpace ( new Vector3 (P2_Depth.x,
   P2_Depth.y, Depth.data[P2_Index] ) );
```

Figure 3.18: Second Point Lying on the Object is Determined

3.5.3. Finding Direction of the Object

Code excerpt that is presented below in Figure 3.19 performs finding direction of the hand-held object that corresponds to Step 13 of the object detection algorithm that is discussed in Section 3.2.

```
/*Direction of the object is found using data of previously found two points in
world coordinates*/
Vector3 sword_direction = P2_World - P1_World;

/*Orientation of the object in real world is determined as a quaternion to project
the actions of the player performed with the sword into gameplay*/
Quaternion current_orientation = Quaternion.LookRotation ( sword_direction,
                               Vector3.forward );
```

Figure 3.19: Direction of the Object is Found

3.5.4. Finding Height of the Object

Code excerpt that is presented in Figure 3.20 performs finding height of the hand-held object that corresponds to Step 14, Step 15 and Step 16 of the object detection algorithm that is discussed in Section 3.2.

```
//Slope of the object in depth coordinates is found
CheckPixel = P2_Depth;
CheckPixelIndex = P2_Index;
CheckPixelDepth = Depth.data[P2_Index];
float slope_radian = Mathf.Atan2 ( P2_Depth.y - P1_Depth.y , P2_Depth.x - P1_Depth.x );
/*As long as the depth value of the candidate pixel lies between the threshold as
compared to the previous candidate, the search continues*/
while ( ( Depth.data[CheckPixelIndex] < CheckPixelDepth + epsilon) && (
Depth.data[CheckPixelIndex] > CheckPixelDepth – epsilon ) )
{
CheckPixelDepth = Depth.data[CheckPixelIndex];
CheckPixel.x += 2 * Mathf.Cos(slope);
CheckPixel.y += 2 * Mathf.Sin(slope);
CheckPixelIndex = (((int)Depth.yres - (int)CheckPixel.y) * Depth.xres) + ((int)Depth.xres -
        (int)CheckPixel.x);
}
CheckPixel.x -= 2 * Mathf.Cos(slope);
CheckPixel.y -= 2 * Mathf.Sin(slope);
//Pixel corresponding to tip of the object is assigned
TipPixel = CheckPixel;
//Pixel corresponding to tip of the object is converted into 3D world coordinates
TipWorld = ConvertImageToWorldSpace ( new Vector3(CheckPixel.x, CheckPixel.y,
      CheckPixelDepth));
```

Figure 3.20: Height of the Object is Found

CHAPTER 4

EVALUATION

This chapter presents evaluation of the developed algorithm that introduces handheld tangible objects into motion controlling using Microsoft Kinect to measure its outcomes on gameplay experience. First, design of an experimental game is presented emphasizing the features that are given importance to be able to perform a reliable evaluation. Then, information of the participant who took part in the user study is presented. Afterwards, experiment procedure is discussed that is followed by results of the study that are analyzed statistically. The chapter concludes with the discussion of the results.

4.1. Experimental Game Design

An experimental game is designed to measure effects of the developed motion recognition system on player experience. During design, it is aimed to create a motion controlled game that is self-expressive without involving complicated mechanics which may result in player confusion, familiar to users and provide the opportunity of measuring effects of the introduced tangible object into motion controlling by being suitable for playing both by using only body movements and by using a hand-held object. On the other hand it is intentionally avoided to develop a well-known sports game to avoid bias of people's prior knowledge of how to use the sports equipment such as a racket or a baseball bat, which may affect the results unfairly. Considering all, a traditional game of hoopla, which originally consists of catching thrown rings with hand-held sticks or in its modern better known version in the form of a children toy, throwing rings onto stable sticks, is found to be appropriate for the scope of this thesis since the game is self-expressive, familiar to players and do not involve so many different variables in it. The game provides healthy measurement of the effects of the developed algorithm since it can both be played using only body movements with touchless interaction and performing motions using a hand-held object.

In the game, player tries to catch the disks that are thrown from a distance with the sword he is holding at hand. To enable comparison of the current and developed controlling styles, two versions of the game is designed one of which is controlled using only body movements and one controlled using hand-held tangible object. Version 1 is designed to be the control game in which only body movements are used for controlling the game. Version 2 is designed to be the experiment game in which tangible object is used for motion controlling.

Gameplay is tried to be kept simple, concentrating on the interaction style without imposing complex mechanics on the player he needs to struggle with while playing the game. Design is kept simple, disks are thrown from a distance at every 3 seconds and the player tries to catch the disks by controlling the virtual sword on the screen. The player is allowed to move in 3D space, moving forwards, backwards and sides to catch the disks that are thrown, since motion controlling is the focus of the study. No avatar is used in the game intentionally to keep the user's attention on the inputs that are projected onto sword and to avoid from any distraction that may be caused by the existence of an avatar hence being able to measure the user experience effectively. All of the disks are programmed to fall in the game area, being possible to get caught by the user. To help the player concentrate on the game matching performed actions easily with their mirroring on screen without performing any spatial projections in his mind, first person viewpoint is used.

An ingame screenshot is shown in Figure 4.1. In the figure, a sword can be seen which the player controls to catch the red disks that are thrown. Blue line on the ground exists as a visual clue for the player indicating that disks will fall on a close distance to it. The player earns a score point as he catches a disk with the sword and the score is shown to the player with a plain HUD on the upper right side of the screen to avoid distraction.

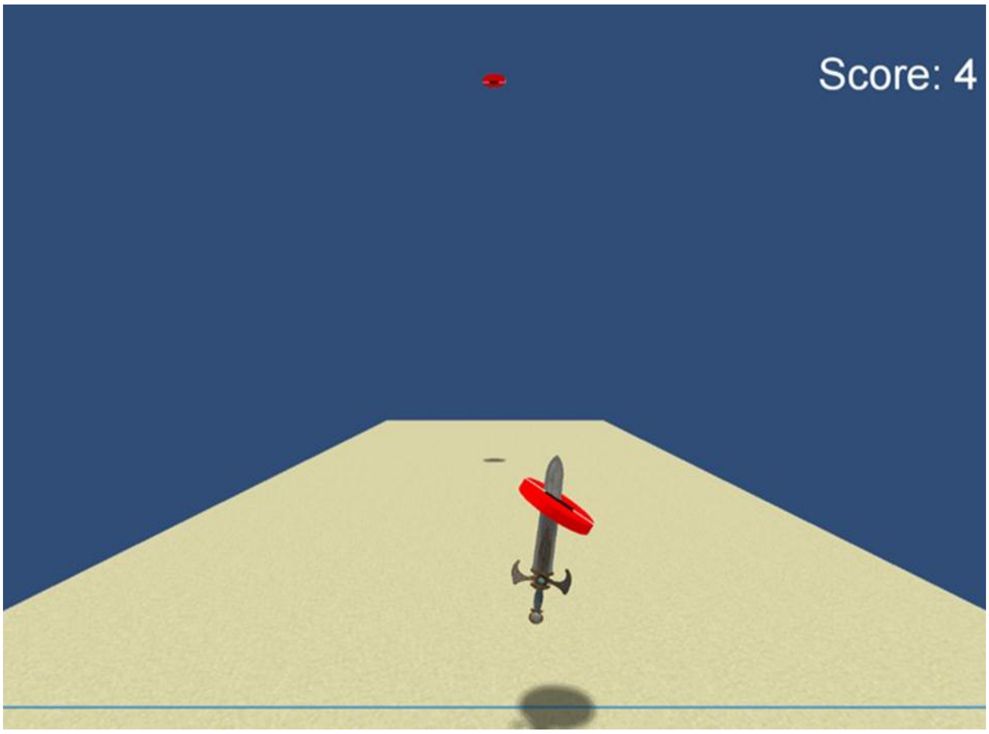


Figure 4.1: An Ingame Screenshot from the Designed Game Hoopla

There are two approaches to three dimensional spatial interaction projection: isomorphism and non-isomorphism [15]. Isomorphism refers to reflecting exactly the same amount of placement that is read from the motion controller to the virtual world. Non-isomorphism refers to scaling the amount of placement that is read when reflecting it to virtual world. Although non-isomorphism is a powerful approach of amplifying user motions to enable a broader range of control, it can bring with a biased gameplay of swinging the hand back and forth rapidly to defeat a challenging boss in seconds, wiping away the fun of video games. To avoid the bias effect of non-isomorphism in the user study, the game is designed to be isomorphic in terms of spatial interaction; player's actions are projected into the game with a ratio of one to one.

4.1.1. Control Version of the Experimental Game

Control game is the first version of the game that is controlled by using only body movements. Controlling gestures of the game are designed to be easy to perform and as meaningful and self-expressive as possible to avoid from any bias that a badly designed gesture would create in comparison of the two versions. Considering these, the player is assumed to be carrying an imaginary sword in his hand that is always perpendicular to his forearm and always located in the imaginary plane that the hand, elbow and shoulder joints constitute. The ingame sword is controlled with arm movements of the player. To rotate the sword in its local z axis, the player needs to move his elbow to the left and right sides. To rotate the sword in its local y axis, the player needs to move his wrist downwards and upwards. Instructions screen of the control game which explains the player how to play the game and informs him about the winning condition is shown in Figure 4.2. To avoid giving users any clue about which of the interaction style is the developed one, in the instructions screen names are used such as Imaginary Sword and Real Sword instead of Version 1 and Version 2, which generally represent improvement in software engineering with the increased number.

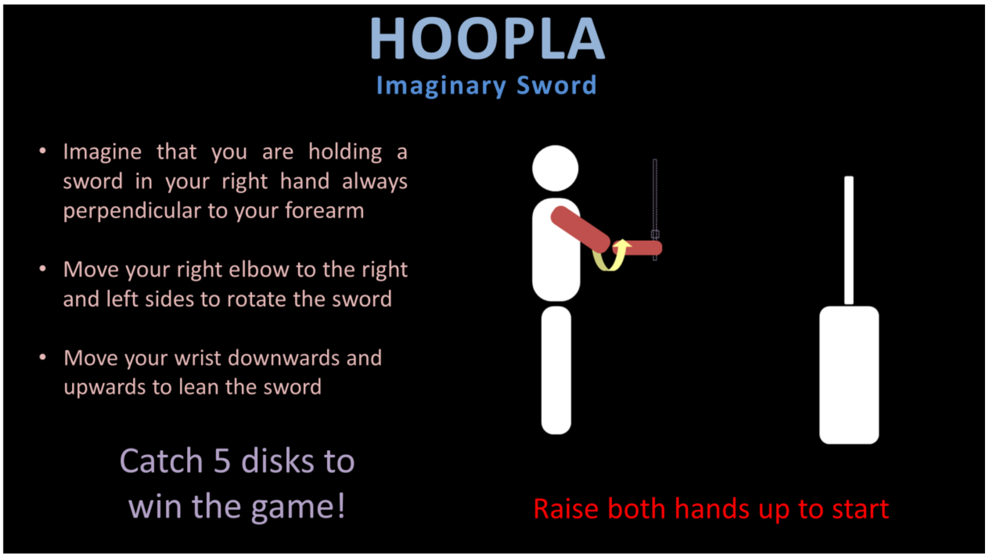


Figure 4.2: Instructions Screen of the Control Game which Presents the Player Required Actions to Play the Game and the Winning Condition

Two photographs that are taken while the player was playing the control game are presented in Figure 4.3 to provide a clear understanding of the interaction. In the left side of the figure, the player opens his elbow to his right side to lean the sword left. The player also keeps his wrist below to lean the sword front. In the right side of the figure, the player keeps his elbow close to his body to lean the sword right. He keeps his wrist below again to lean the sword front.

Figure 4.3: Player Playing the Control Game
Left: Elbow is Opened to Right to Lean the Sword Left and the Wrist Kept Below to Lean the Sword Front Right: Elbow is Closed to Left to Lean the Sword Right and the Wrist Kept Below to Lean the Sword Front

4.1.2. Experiment Version of the Experimental Game

Experiment game is the second version of the game that is controlled by using hand-held tangible object. Since movements of the hand-held sword are projected into gameplay one to one, no controlling gestures are needed to be assigned. To rotate the ingame sword in its local z axis, the player just needs to rotate the tangible sword in its local z axis and similarly to rotate the sword in its local y axis, the player needs to rotate the tangible sword in its local y axis since hand-held sword orientation is projected directly into gameplay. Instructions screen of the experiment game which explains the player how to play the game and informs him about the winning condition is shown in Figure 4.4.

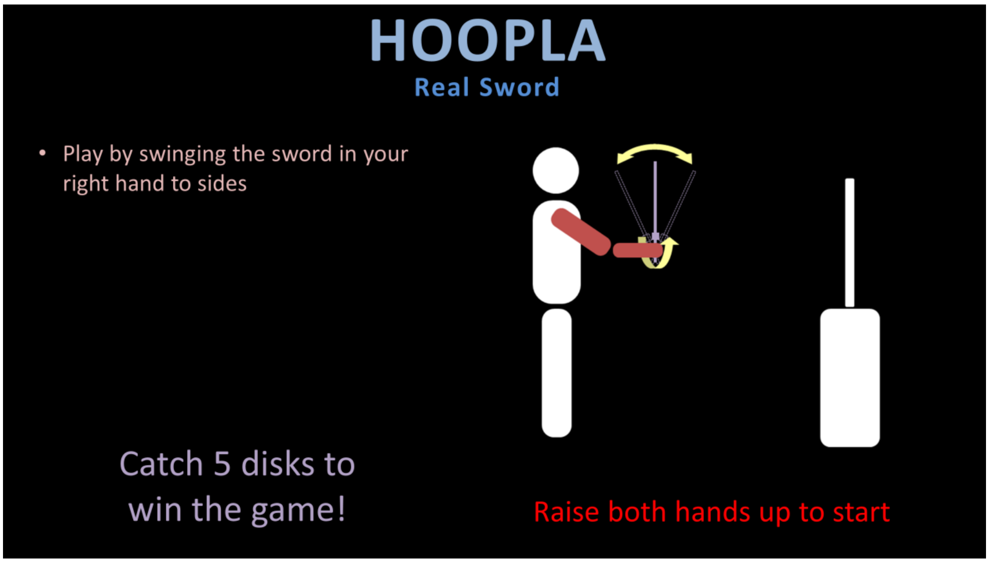


Figure 4.4: Instructions Screen of the Experiment Game which Presents the Player Required Actions to Play the Game and the Winning Condition

Two photographs that are taken while the player was playing the experiment game are presented in Figure 4.5 to provide a clear understanding of the interaction. In the left side of the figure, the player leans the sword he is holding to his right forward which is projected directly to the virtual ingame sword that can be seen in the figure. In the right side of the figure, the player leans the tangible sword to his left which is projected directly to the virtual ingame sword.

Figure 4.5: Player Playing the Experiment Game

Left: Hand-Held Sword is Leaned to Right to Lean the Ingame Sword Right

Right: Hand-Held Sword is Leaned to Left to Lean the Ingame Sword Left

4.2. Participant Information

A user study is performed to evaluate effects of the proposed system on user experience. The study involves participant's playing the same game that is controlled bare hand using only body movements and using a tangible toy sword grabbed, and then evaluating versions that are played. 16 subjects participated to the study with ages ranging from 18 to 29 having a mean of 23 and standard deviation of 3.27. All of the participants were students with a specialization distribution as follows: 6 computer engineering undergraduate students (%37.5), 5 advertising undergraduate students (31.25%), 2 mechanical engineering graduate students (12.5%), 2 industrial engineering graduate students (%12.5) and one high school student (6.25%). Participants were recruited by e-mails and word of mouth. All participants are chosen to be right handed since the game interaction is designed to be played with right hand as the dominant one, considering that most of the people are right handed, according to the study of Raymond et al. with a percentage of 87-90% [48]. To overcome any possible gender bias, participants are selected to be equal number of 8 males and 8 females.

Considering how often do they play video games, 4 participants stated that they play video games less than once a week (25%) and 12 participants stated that they play video games more than once a week (75%). Participants' video game playing hours in a week range from 1 hour to 50 hours having a mean of 20.06 and standard deviation of 15.08. Since data is scattered having a wide range and large standard deviation, to provide a clear and more meaningful idea on the game playing hours of the participants in a week, data are presented below in Table 4.1, in categories expressing the frequency of game playing. According to data, 4 participants spend less than 5 hours playing video games in a week, 3 participants spend more time between 6 - 15 hours, 6 participants spend 16 - 30 hours playing video games in a week and 3 participants play video games frequently spending between 31 - 50 hours in a week. The data shows a scattered distribution of playing hours, which is desirable for the scope of this user study since it eliminates any possible bias that may be caused from prior gameplay frequency of the participants, yielding to a tendency in some particular controlling styles.

Table 4.1 - Weekly Game Playing Hours of the Participants

Playing Hours a Week	Infrequent (1 - 5)	Moderate (6 - 15)	Often (16 - 30)	Frequent (31 - 50)
Number of Participants	4	3	6	3

As their prior Kinect experience is questioned, 7 participants with a percentage of 43.75% stated that they have prior game play experience with Kinect. 9 participants with a percentage of 56.25% stated that they have no prior Kinect experience and will be interacting with it first time during the study. This provides an approximately even ratio of players who have prior Kinect experience and not, which is desirable for the scope of this user study since it eliminates any possible bias that may be caused from prior Kinect experience of the players, yielding to a tendency in some particular controlling habits.

4.3. Experiment Procedure

At the beginning of the study, users are informed about the process of the experiment, requested to fill out a form consisting of demographic information and their prior gaming experience, which is presented in Appendix A. Not to affect their evaluation, the participants are not informed about which interaction style is the developed one in this thesis, they only told that they will be playing the same game with two different interaction styles, and will evaluate the interactions they experienced individually.

Following, they are forwarded to the gaming area which is a plain room that is not exposed to direct sunlight, hosting a television and Kinect inside. The order of game versions in which the subjects were presented with was randomized to eliminate any related bias. The subjects were then presented an instruction screen as they were shown in Figure 4.2 and Figure 4.4, involving the instructions on how to play the game and the win condition. Win condition of the game is determined to be catching 5 disks successfully for both conditions, which is optimized by self playing tests to give the users enough time to experience the interaction. Since the game involves 3D spatial variables, scoring requires coordination and is not so easy hence 5 scores give the users enough time to evaluate the interaction on the other hand not making them tired which may affect the following interaction evaluation negatively.

After finishing playing the game by meeting the win condition, participants are requested to fill out a questionnaire according to their first assigned version which consists of the questions that are presented in Appendix B for version 1 (control game) and consists of the questions that are presented in Appendix C for version 2 (experiment game). After completing the questionnaire, they are requested to fill out the Subjective Mental Effort Questionnaire (SMEQ) chart of their assigned version which is presented in Appendix D for control version and presented in Appendix E for experiment version. Following, the participants are requested to state their most positive and negative experience for the assigned controlling style in their own words in a sentence or so using the forms that are presented in Appendix F and Appendix G. The participant is then assigned with the other version of the game and after finishing the game he/she repeats the process that is explained above for this second version. At the end, participants are requested to indicate their preferred controlling style between the two they experienced filling out the form that is presented in Appendix H.

4.4. Results

Results of the user study are presented below in relevant subsections with their statistical analyses. First, questionnaire results collected from the participants are presented that are followed by the discussion of Subjective Mental Effort Questionnaire results. Some quantitative data have been collected in the background while the participants were playing the games and they are presented next. User preference and user reviews are presented in the final part of the subsection.

4.4.1. Questionnaire Results

Subjects were requested to fill out a questionnaire of their assigned version consisting of the questions that are presented in Appendix B for the control version and in Appendix C for version the experiment version. The questions in the questionnaire are designed with the aim of measuring user experience on different aspects of interaction following the guidelines that are presented in the work of Brown et al. on evaluating game controllers [49]. The questions assess the user experience in different categories of enjoyment, frustration, gameplay, control, learning burden, naturalness, physical sense, immersion and performance. These categories are constructed utilizing information on the assessment of core elements of the player experience on gaming in the work of Gámez et al [50]. Questions that are related to the scope of the thesis are adopted from Core Elements of the Gaming Experience Questionnaire (CEGEQ) and many additional questions are designed on interaction experience since the study focuses on comparing two different interaction styles. Participants are expected to answer the questions using a five point Likert scale where 1 corresponds to strongly disagree and 5 corresponds to strongly agree. Results of the questionnaire are presented below in their relevant categories with their statistical analyses. In the discussions below, version 1 refers to the control version that is played by using only body movements and version 2 refers to the developed experiment version that is played by using hand-held tangible object. All t-tests are performed with alpha of 0.05.

Enjoyment: Three questions are asked to participants to assess effect of the developed interaction method on player enjoyment. Answers to these questions are analyzed statistically as a group by performing t-test which resulted in a mean score of 2.833 for version 1 with standard deviation 0.781, and a mean score of 3.521 for version 2 with standard deviation 0.744, turning out to be statistically significant with a t value of -5.575 and p value of 0.0000 which can be interpreted as the players enjoyed the game more while they played it with the interaction style of version 2 in which tangible object is used.

To investigate different aspects of the enjoyment assessment closer, individual t-tests are also performed for each question in the group. T-test for the question “I enjoyed playing the game.” resulted in a mean score of 2.875 for version 1 with standard deviation 0.619, and a mean score of 3.563 for version 2 with standard deviation 0.727, turning out to be statistically significant with a t value of -3.905 and p value of 0.0007 which can be interpreted as the players enjoyed playing the game more with the interaction style of version 2.

T-test for the question “I liked the game.” resulted in a mean score of 2.875 for version 1 with standard deviation 0.650, and a mean score of 3.375 for version 2 with standard deviation 0.517, turning out to be statistically significant with a t value of -2.236 and p value of 0.0205 which can be interpreted as players liked the game more with the interaction of version 2.

T-test for the question “I would play this game again.” resulted in a mean score of 2.750 for version 1 with standard deviation 0.931, and a mean score of 3.625 for version 2 with standard deviation 0.806, turning out to be statistically significant with a t value of -3.656 and p value of 0.0012 which can be interpreted as players felt as more probable to play the game again when they played it with the interaction style of version 2.

These results indicate that interaction style affects the player’s perception on game considering that the participants stated that they found the game more enjoying as they played with the interaction version 2. Of course, this does not mean that touchless interaction does not provide the players enjoyment. Game design has a direct effect on the results. The players may have found the game more enjoyable while playing with the tangible object due to the game's nature of being played with physical objects. Other game designs, especially the ones having abstract themes, may yield to different results. Enjoyment of the game can be adjusted by adding different components and challenges that keep the player in flow state. But considering this experimental game design in which interaction and challenges kept simple to incline the player focus on the interaction more, players state that they enjoyed playing the game more with a tangible object.

Frustration: One question is asked to the participants to assess the effect of the developed interaction style on frustration. The question is analyzed statistically by performing t-test which resulted in a mean score of 2.875 for version 1 with standard deviation 0.957, and a mean score of 2.125 for version 2 with standard deviation 0.885, turning out to be statistically significant with a t value of 3.873 and p value of 0.0008 which can be interpreted as players got frustrated more while they were playing the game with the interaction style of version 1. Frustration of the players while playing with the interaction style of version 1 may have been caused from the difficulty they might have encountered in mental projections of the actions they perform into the gameplay. This may have been resulted in failures which make then frustrated. Lower mean score for the interaction of version 2 was expected since physical objects help the players in understanding projections of the motions easily due to their naturally being physical entities rather than abstract ones.

Gameplay: Three questions are asked to participants to assess the effect of the developed interaction on gameplay. These questions are analyzed statistically as a group by performing t-test which resulted in a mean score of 2.229 for version 1 with standard deviation 1.016, and a mean score of 3.479 for version 2 with standard deviation 0.989, turning out to be statistically significant with a t value of -7.364 and p value of 0.0000 which can be interpreted as players enjoyed gameplay as they played the game with the interaction style of version 2 more.

To investigate different aspects of gameplay assessment closer, individual t-tests are also performed for each question in the group. T-test for the question "It was easy to score." resulted in a mean score of 1.500 for version 1 with standard deviation 0.817, and a mean score of 2.938 for version 2 with standard deviation 0.772, turning out to be statistically significant with a t value of -5.965 and p value of 0.0000 which can be interpreted as players scored easier with the interaction of version 2 than they did with the interaction style of version 1. This difference in mean scores again may have been arisen from the physical object's assistance in mental projection of the actions performed, yielding to easier scoring of the player.

T-test for the question “I understood how to play the game easily.” resulted in a mean score of 3.125 for version 1 with standard deviation 0.619, and a mean score of 3.938 for version 2 with standard deviation 0.854, turning out to be statistically significant with a t value of -3.569 and p value of 0.0014 which can be interpreted as the players understood how to play the game easier with the interaction style of version 2. The result is expected since in version 1, there are gestures that the players need to learn and perform in the game which incurs a mental load on them while in version 2 there are no learned gestures or movements, the players just use the sword in their hands, hence understanding how to play the game easier as compared to version 1.

T-test for the question “The game was easy.” resulted in a mean score of 2.063 for version 1 with standard deviation 0.854, and a mean score of 3.563 for version 2 with standard deviation 1.094, turning out to be statistically significant with a t value of -3.985 and p value of 0.0006 which can be interpreted as the players perceived the game to be easier as they played it with the interaction style of version 2. The game’s being perceived easier is not a positive thing from the point of a good game design. Since they interact with the game and score easier, the players perceive the game as to be easier with interaction version 2. There should be added additional challenges to keep the player in the flow state with this interaction style since an easy game would cause boredom and result in player losses.

Control: Three questions are asked to participants to assess the effect of the developed interaction on control. These questions are analyzed statistically as a group by performing t-test which resulted in a mean score of 2.625 for version 1 with standard deviation 0.841, and a mean score of 3.688 for version 2 with standard deviation 0.948, turning out to be statistically significant with a t value of -7.223 and p value of 0.0000 which can be interpreted as players liked the control that version 2 provided more.

To investigate different aspects of control assessment closer, individual t-tests are also performed for each question in the group. T-test for the question "I interacted with the game easily." resulted in a mean score of 2.938 for version 1 with standard deviation 0.772, and a mean score of 3.688 for version 2 with standard deviation 0.873, turning out to be statistically significant with a t value of -3.223 and p value of 0.0028 which can be interpreted as the players interacted with the game easier with the interaction style of version 2.

T-test for the question "The game was easy to control." resulted in a mean score of 2.438 for version 1 with standard deviation 0.814, and a mean score of 3.688 for version 2 with standard deviation 1.138, turning out to be statistically significant with a t value of -4.443 and p value of 0.0002 which can be interpreted as players controlled the game easier with the interaction style of version 2 than they did with the interaction style of version 1.

T-test for the question “I felt that the control was on me during the game.” resulted in a mean score of 2.500 for version 1 with standard deviation 0.894, and a mean score of 3.688 for version 2 with standard deviation 0.873, turning out to be statistically significant with a t value of -4.842 and p value of 0.0001 which can be interpreted as players felt the control on them more with the interaction style of version 2.

The player preference of version 2 on controlling aspects may have been due to the tangible object’s assistance in interaction. The players stated that they felt control on them more with the interaction of version 2, most probably because of the tactile sense they experienced during gameplay.

Learning Burden: Two questions are asked to participants to assess the effect of the developed interaction on learning burden. These questions are analyzed statistically as a group by performing t-test which resulted in a mean score of 2.469 for version 1 with standard deviation 1.016, and a mean score of 2.031 for version 2 with standard deviation 0.740, turning out to be statistically significant with a t value of 2.301 and p value of 0.0142 which can be interpreted as control of version 2 imposed less learning burden on players.

To investigate different aspects of learning burden assessment closer, individual t-tests are also performed for each question in the group. T-test for the question “It was difficult to learn the gestures that are required to control the game." resulted in a mean score of 2.500 for version 1 with standard deviation 1.033, and a mean score of 2.375 for version 2 with standard deviation 0.719, turning out to be statistically insignificant with a t value of 0.436 and p value of 0.3346 which can be interpreted as players did not learn the gestures that were required to control the game more difficultly with the interaction of version 1. Although it was expected that version 1 provided more difficulty in learning, the difference did not prove to be significant between two versions. The reason for this closeness between mean scores may have been the gestures being kept simple and easy to understand during game design. The players may not have been found the gestures of version 1 difficult to learn at all but if more complex interaction mechanics were involved in the game, requiring more gestures to be learnt, the difference may have been more significant.

T-test for the question "It was difficult to remember which movement to perform to do the required actions during the game." resulted in a mean score of 2.438 for version 1 with standard deviation 1.031, and a mean score of 1.688 for version 2 with standard deviation 0.602, turning out to be statistically significant with a t value of 3.223 and p value of 0.0028 which can be interpreted as players found it more difficult to remember the gestures that were required to control the game with the interaction of version 1. Although the difference is statistically significant, mean scores are close again, which may have been caused by the simple and easy to understand gesture design of version 1. It was expected that the players would have more difficulty in remembering the required actions to be performed during gameplay with version 1 since it involves learned gestures while version 2 provides a completely natural one to one projected interaction in which the player does not need to learn any gestures.

Naturalness: Four questions are asked to the participants to assess the effect of the developed interaction style on naturalness. These questions are analyzed statistically as a group by performing t-test which resulted in a mean score of 2.766 for version 1 with standard deviation 0.792, and a mean score of 3.688 for version 2 with standard deviation 0.958, turning out to be statistically significant with a t value of -5.956 and p value of 0.0000 which can be interpreted as version 2 provided the players a more natural way of interaction.

To investigate different aspects of naturalness assessment closer, individual t-tests are also performed for each question in the group. T-test for the question "I think the interaction was natural." resulted in a mean score of 2.875 for version 1 with standard deviation 0.619, and a mean score of 3.938 for version 2 with standard deviation 1.063, turning out to be statistically significant with a t value of -3.171 and p value of 0.0032 which can be interpreted as players found the interaction style of version 2 more natural.

T-test for the question "Controls were intuitive." resulted in a mean score of 2.438 for version 1 with standard deviation 0.814, and a mean score of 3.625 for version 2 with standard deviation 0.806, turning out to be statistically significant with a t value of -3.721 and p value of 0.0010 which can be interpreted as players found the controlling inputs of the interaction style version 2 more intuitive.

T-test for the question “It was familiar for me to control the game.” resulted in a mean score of 3.188 for version 1 with standard deviation 0.750, and a mean score of 3.938 for version 2 with standard deviation 0.854, turning out to be statistically significant with a t value of -2.818 and p value of 0.0065 which can be interpreted as the players found the interaction style of version 2 more familiar.

T-test for the question “I felt that I already knew how to play the game.” resulted in a mean score of 2.563 for version 1 with standard deviation 0.814, and a mean score of 3.250 for version 2 with standard deviation 1.000, turning out to be statistically significant with a t value of -2.112 and p value of 0.0259 which can be interpreted as players felt like as if they knew the interaction already while playing with version 2.

The significant score difference between two versions was expected since tangible object provides a more natural interaction for the developed game. Player found it more natural, intuitive and familiar to interact using the tangible object, performing actions and seeing them mirrored into gameplay one to one. This may not be the case for all game designs though, since not all physical world activities are performed with a tangible object, tangible interaction may not fit into every game design, and may provide a worse experience, but it is expected to provide a more natural way of interaction especially for video games of physically controlled real world activities. The users also stated that they felt like they already knew how to play the game more with the interaction style of version 2 which is expected since version 2 does not require the player to perform any imaginary or meaningless actions, the player just needs to perform naturally as if he is in a physical activity of real life.

Physical Sense: To assess the effect of the developed interaction on physical sense, a question is asked to participants for each version. In version 1, players answered the question "I disliked the feeling of the lacking that is caused by holding nothing physical in my hand during gameplay." with a mean score of 3.938 having a standard deviation 0.772, which can be interpreted as players mostly agreed having been disliked the lacking of any physical object during interaction. In version 2, players answered the question "I enjoyed feeling the sword in my hand physically during gameplay." with a mean score of 4.438 having a standard deviation 0.629, which can be interpreted as players mostly agreed having been enjoyed having a physical object during interaction. Since a version involves physical object and a version does not, the questions could not be united, hence are examined separately. The results indicate that players enjoyed the presence of a physical object in their hands. This may have many aspects; the users might have felt social confidence while interacting with the sword, they might have enjoyed the tactile feedback provided by the sword or felt more confident by the ease of movement projection the sword provided.

Immersion: One question is asked to participants to assess the effect of the developed interaction on immersion. The question "I forgot everything around me during gameplay." is analyzed statistically by performing t-test which resulted in a mean score of 2.375 for version 1 with standard deviation 0.719, and a mean score of 2.625 for version 2 with standard deviation 0.718, turning out to be statistically insignificant with a t value of -1.291 and p value of 0.1081 which can be interpreted as players did not get significantly more immersed with the interaction style of version 2, although mean score is slightly higher than it is for version 1. The presence of tangible object is expected to help the user enter into game world easier, providing a physical means of movement projection but the mean difference is found to be statistically insignificant although version 2 provided a slightly higher mean score of user immersion. This may have been caused by the game's involving few variables to draw the player in.

Performance: Four questions are asked to participants to assess the effect of the developed interaction on performance. These questions are analyzed statistically as a group by performing t-test which resulted in a mean score of 2.078 for version 1 with standard deviation 0.803, and a mean score of 3.438 for version 2 with standard deviation 1.082, turning out to be statistically significant with a t value of -9.728 and p value of 0.0000 which can be interpreted as version 2 provided the players better performance opportunity.

To investigate different aspects of performance assessment closer, individual t-tests are also performed for each question in the group. T-test for the question "It was easy to project the actions I desired to perform into the game." resulted in a mean score of 2.063 for version 1 with standard deviation 0.680, and a mean score of 4.063 for version 2 with standard deviation 0.574, turning out to be statistically significant with a t value of -12.649 and p value of 0.0000 which can be interpreted as players projected the actions they desire to perform significantly easier with the interaction style of version 2 with the presence of a tangible object. Significant difference between mean scores may have been due to the ease tangible object provides in motion projection and mental modeling, providing a physical medium to the player to interact with.

T-test for the question "I felt what was happening in the game was my own doing." resulted in a mean score of 2.188 for version 1 with standard deviation 0.750, and a mean score of 3.250 for version 2 with standard deviation 1.065, turning out to be statistically significant with a t value of -4.000 and p value of 0.0006 which can be interpreted as players felt that the actions performed in the game were their own doings more with the interaction style of version 2. The significant mean difference between two versions was expected since version 2 provides the player a direct interaction opportunity in which actions of the player are projected directly into gameplay, yielding to the feeling of effecting gameplay with actions performed more.

T-test for the question "I think I performed well on the game." resulted in a mean score of 1.750 for version 1 with standard deviation 0.683, and a mean score of 2.625 for version 2 with standard deviation 1.088, turning out to be statistically significant with a t value of -3.050 and p value of 0.0040 which can be interpreted as players felt that they performed better on the game with the interaction style offered by version 2. The perception of better self performance may have been caused by the easy interaction that tangible object provides to the player.

T-test for the question "I felt like I really performed the actions in the game." resulted in a mean score of 2.313 for version 1 with standard deviation 1.015, and a mean score of 3.813 for version 2 with standard deviation 0.981, turning out to be statistically significant with a t value of -4.743 and p value of 0.0001 which can be interpreted as players felt like they really performed the actions in the game more with the interaction of version 2. The feeling of really performing the actions is expected with interaction version 2 since the player performs the actions in the game one to one as they are mirrored into gameplay.

To find out if there exists any bias caused by a difference between motion recognition sensitivity of the two versions that may have been effected the given answers and performed t-tests, a question of "Motion recognition was sensitive." is asked to the participants for both versions of the game and the t-test is resulted in a mean score of 3.563 for version 1 with standard deviation 0.814, and a mean score of 3.688 for version 2 with standard deviation 0.873, turning out to be statistically insignificant with a t value of -0.565 and p value of 0.2902 which can be interpreted as motion recognition sensitivity did not vary significantly in two versions.

4.4.2. Subjective Mental Effort Questionnaire

Participants are asked to fill out the SMEQ chart following the questionnaire after having finished playing each version, that are presented in Appendix D and Appendix E. Aim of the chart is to measure the mental effort a user spends while performing an activity. The chart is presented in Figure 4.6. On the chart there are different mental effort levels that are scaled from 0 to 150 with verbal explanations of the mental effort related to scales at some representative levels. The participant is asked to mark the level that fits the mental effort they spent during the activity most. Participants reflected the mental effort they spent for version 1 with a mean score of 81.06 having standard deviation 9.719, and mental effort they spent for version 2 with a mean score of 57.88 having standard deviation 18.822, indicating that players spent more mental effort while playing with version 1, which is expected since there are more controlling requirements that need to be considered in this version as opposed to version 2, having a completely natural form of interaction which does not need many thinking since actions performed with the physical sword are mirrored into the virtual sword in the game one to one. SMEQ histograms for version 1 and version 2 which shows the mean SMEQ scores with their standard deviations are presented in Figure 4.7. SMEQ distributions for version 1 and version 2 which shows the SMEQ score of each participant are presented in Figure 4.8.

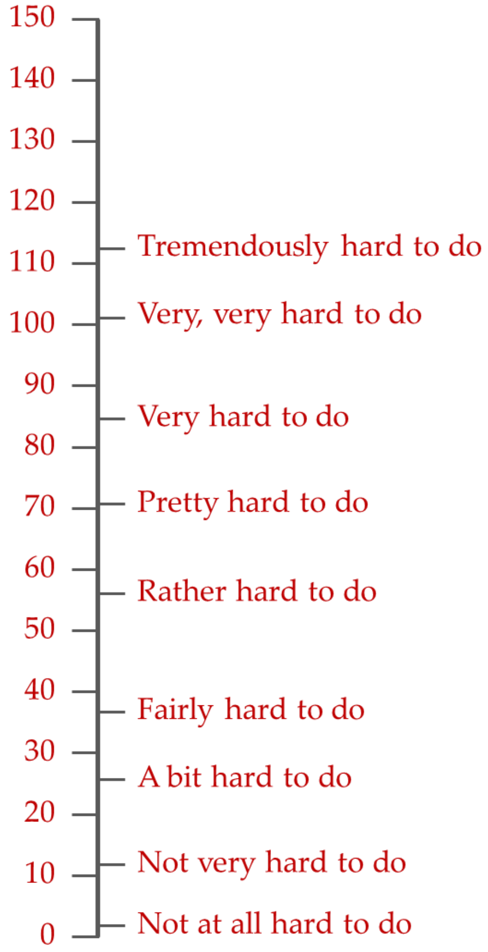


Figure 4.6: SMEQ Chart

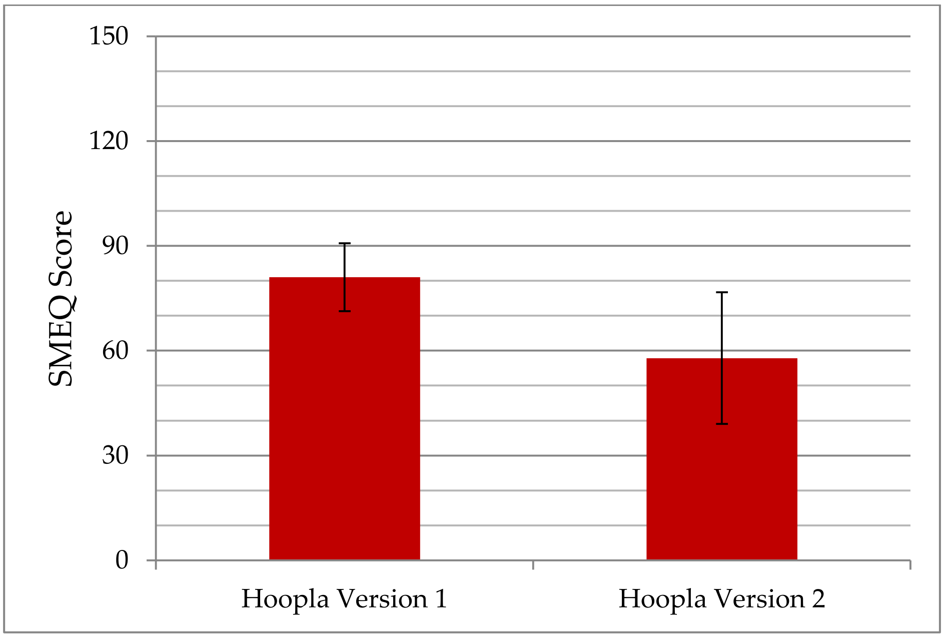


Figure 4.7: SMEQ Histograms for Version 1 and Version 2

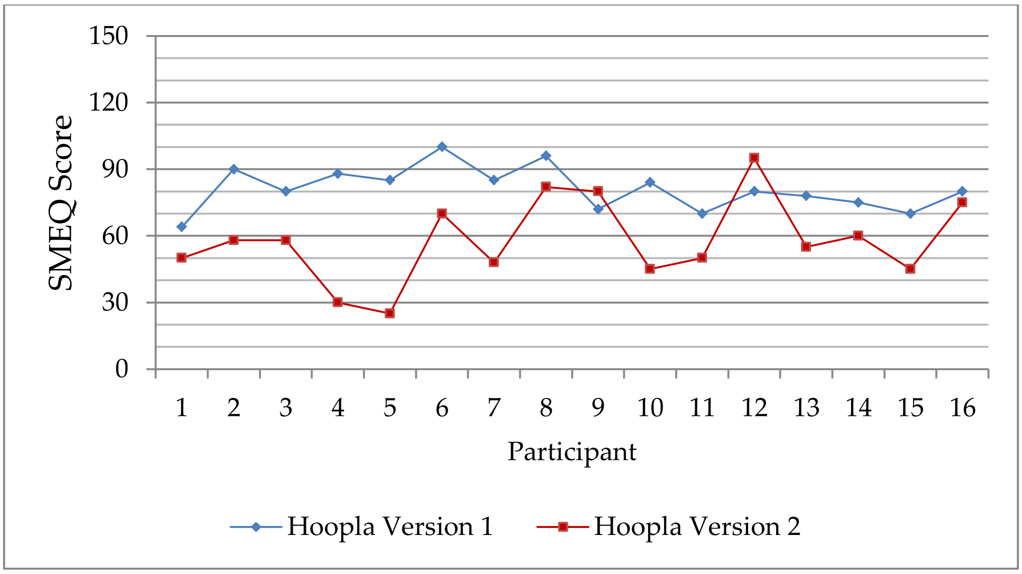


Figure 4.8: SMEQ Distributions for Version 1 and Version 2

4.4.3. Quantitative Data

Besides the user completed questionnaires, quantitative data are also collected during gameplay at the background which are helpful indicators of user experience assessment. As the quantitative measures, number of fails, game completion time and average success times are collected. These quantitative data are presented and discussed below in their relevant categories with illustrative graphs and statistical analysis.

Number of Fails: The quantitative measure number of fails indicates the times number of fails the participant makes until achieving the goal of five successes that is required to finish the game. If a disk is not caught by the player successfully, it is considered as a fail. Number of fails that the participants made has a mean of 77.00 for version 1 with a standard deviation of 46.256 and a mean of 47.94 for version 2 with a standard deviation of 23.101. Histograms of the average number of fails for both version 1 and version 2 are presented in Figure 4.9. There is a considerable difference in number of fails of two interaction versions. To validate the significance of this difference, t-test is applied to the data which turned out to be statistically significant with a t value of 3.672 and p value of 0.0023 which can be interpreted as number of fails varied significantly in two versions of interaction. Lower number of fails observed with interaction version 2 may have been due to the ease of spatial motion projection the tangible object provides and the player's easily mapping desired actions into gameplay with the help of tangible object.

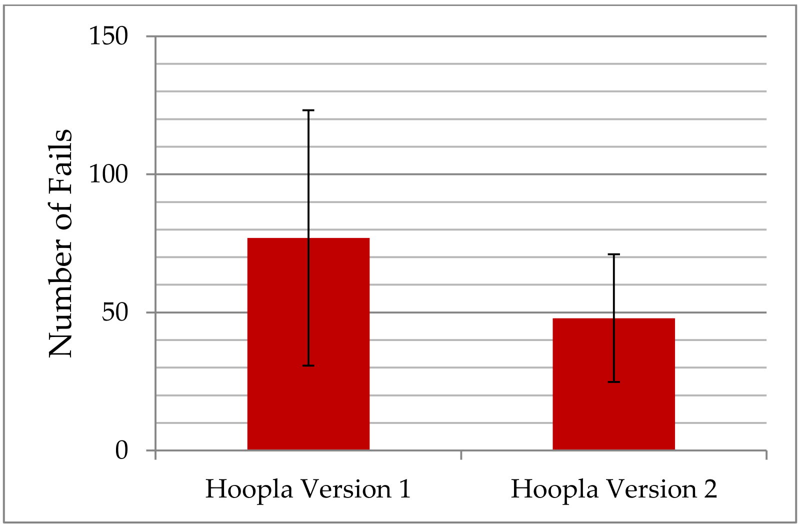


Figure 4.9: Number of Fails Histograms for Version 1 and Version 2

Completion Time: The quantitative measure completion time indicates the time it takes for the participant to finish the game, according to the win condition catch 5 disks successfully. Completion time has a mean of 251.94 seconds for version 1 with a standard deviation of 138.748 and a mean of 163.88 seconds for version 2 with a standard deviation of 69.330. Histograms of the completion times for both version 1 and version 2 which shows the mean completion time of the participants with their standard deviations are presented in Figure 4.10. Distributions of the completion times for both version 1 and version 2 which shows the completion time of each participant are presented in Figure 4.11. There is a considerable difference in completion times for the two interaction versions as it can be observed from the histograms. To validate the significance of this difference, t-test is applied to data which turned out to be statistically significant with a t value of 3.717 and p value of

0.0021 which can be interpreted as completion times varied significantly in two versions of interaction. Shorter game completion times observed with interaction style of version 2 may have been due to the ease of gameplay the tangible object provided, enabling the player reach winning condition faster.

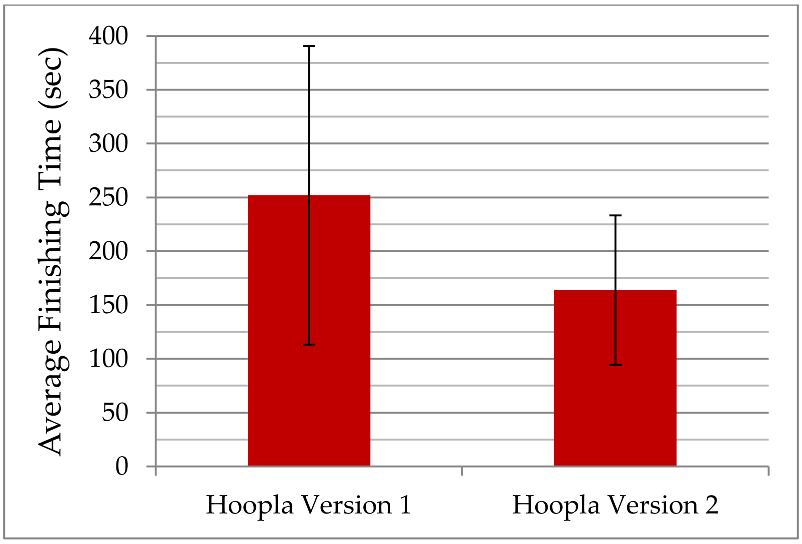


Figure 4.10: Completion Time Histograms for Version 1 and Version 2

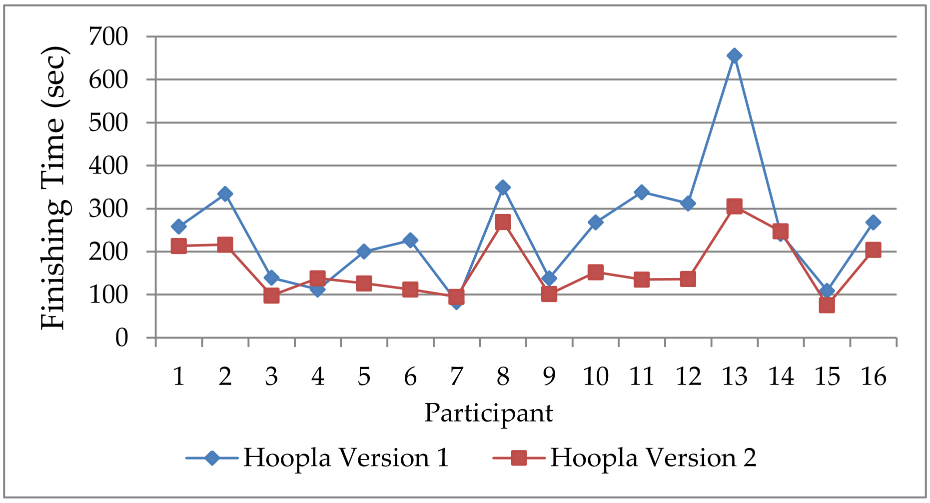


Figure 4.11: Completion Time Distributions for Version 1 and Version 2

Average Success Times: The final quantitative measure average success times indicate the time it takes for the participant to achieve two consecutive successes. Histograms of the average success times for both version 1 and version 2 are presented below in Figure 4.12. There can be observed from the figure a considerable difference in average success times for the two interaction versions. To overcome any accumulated bias that may arise from the continuous gameplay, time between consecutive successes are normalized and in order to validate the significance of this observed difference, t-test is applied to the normalized data which turned out to be statistically significant with a t value of 3.717 and p value of 0.0021 which can be interpreted as average success times varied significantly in two versions of interaction. Shorter average consecutive success times observed with interaction style of version 2 may again have been due to the ease of gameplay the tangible object provided, enabling the player scoring faster.

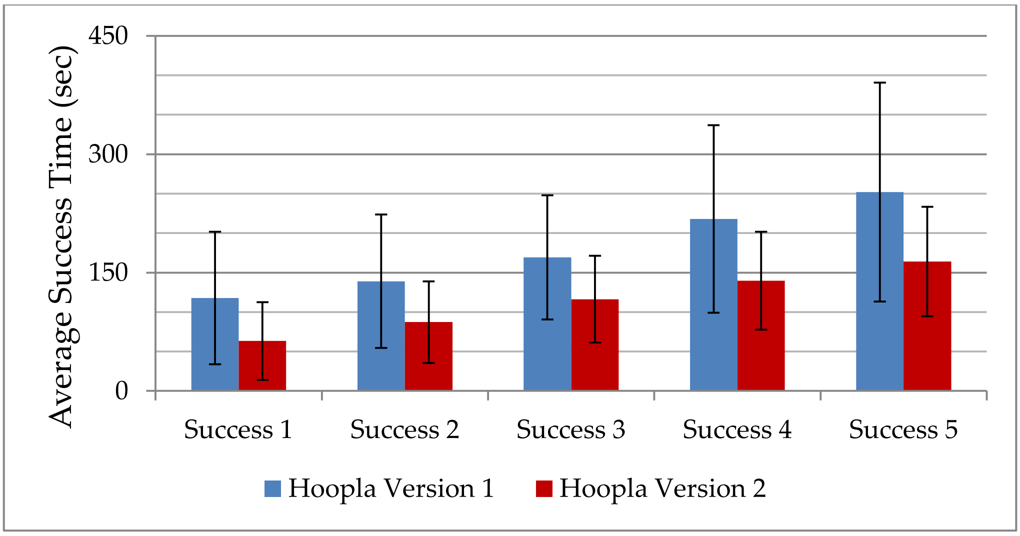


Figure 4.12: Average Success Times Histograms for Version 1 and Version 2

4.4.4. User Preference

At the end of the experiment, participants are asked to state their preference between experimented interaction versions using the form that is presented in Appendix H on a two way scale ranging from much preferring version 1 over version 2 to much preferring version 2 over version 1. The preferences are scored from 1 to 5, 1 indicating a strong preference of version 1 over version 2 and 5 indicating a strong preference of version 2 over version 1. Preference score distributions for both versions can be seen below in Figure 5.8. User preference has a mean of 4.13 with a standard deviation of 1.258 indicating that the users had a strong preference of version 2 over version 1. The strong user preference indicates that motion controlling with tangible objects may provide a favorable interaction style, especially in games that simulate real life physical activities.

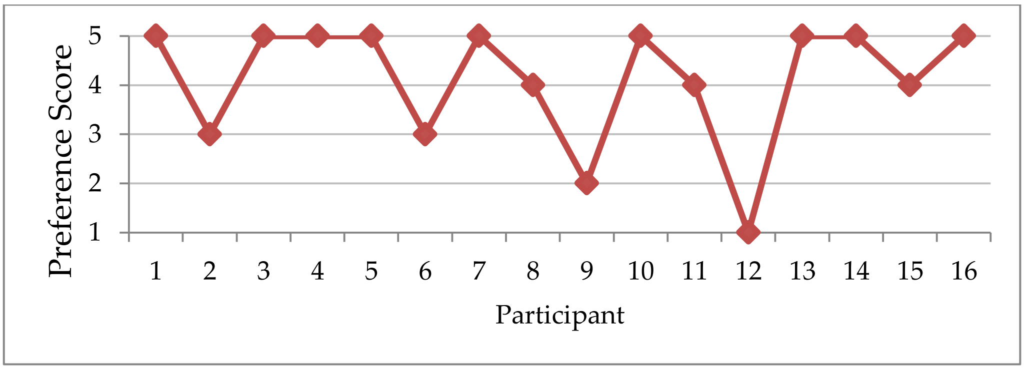


Figure 4.13: Preference Score Distributions for Version 1 and Version 2

4.4.5. User Reviews

The participants are asked to express their most positive and negative experience for both versions in their own words as a qualitative feedback measure that is designed by modifying Critical Incidents Technique (CIT) presented in the study of Brown et al. [49]. Since the measurement techniques that are developed so far on user experience evaluation may not be fully effective in measuring the difference between two novel and new interaction styles in the area of gaming, the user reviews are valuable feedbacks for the study that may reflect any overlooked and unpredicted aspects of the interactions. Significant ones selected among these user reviews are presented below, similar user reviews are avoided to be repeated.

Most of the user reviews were on the physicality and reality of the experience. A subject stated that "It was much better to feel the sword in my hand, it felt like 'really' playing the game after swinging my elbows to sides holding an imaginary sword." indicating that he felt uncomfortable with doing imaginary movements in version 1 which may have caused social distraction on him. Another subject stated that "I could not felt like that I was holding a sword in Hoopla Imaginary Sword." indicating that she had difficulty in imagining she is holding the virtual sword in her hand during the interaction of version 1. Another subject stated a close comment as "It is really hard to think that you are using an imaginary sword. Using a real sword was fun and it was also easier to catch the disks." both exhibiting his preference of using the physical object and mentioning that he caught the disks easier using the physical sword.

A subject expressed his most negative experience of version 1 as "Hoopla imaginary sword game was way too virtual for me, Hoopla real sword was more real I think.". Another made a similar statement of "Real Sword game was more realistic and it was more comfortable to play.". Another subject stated her most positive experience of version 2 as "In Real Sword Hoopla it was good to feel that I actually was playing with a real sword." pointing out the physicality that the interaction style provides.

Other reviews were on the ease of play the physical sword provided. A subject stated his experience as "It was easier to catch the disks with the sword, I enjoyed the game more, felt like really catching those disks!" specifying that he caught the disks easier with the sword he held in version 2. Another subject stated her most negative experience of version 1 as "It was hard to understand and manage the movements of the sword without holding anything in my hand. It was better to control with the plastic sword, more realistic and easy to understand the movements. I liked it more." pointing out the difficulty she encountered in version 1 which is controlled only using body movements.

User reviews indicate in general that they liked playing the game more with the physical object, finding the interaction more realistic and familiar. The participants also had complaints on the difficulty of interpreting spatial actions while imagining that they are holding a virtual sword in their hands, and found the tangible sword helpful in spatial motion interpretation.

4.5. Discussion

Results of the user study indicate that introducing tangible objects into motion controlling may provide an improved gameplay experience with increased realism and more natural interaction form. Developed interaction system helps players to project their desired actions into virtual world easily with the presence of tangible object, seeing actual motions performed mirrored identically into virtual world. Tangible object also enables the player enter into virtual world from real world effectively and easily with the physical interaction provided serving as a connection between real and virtual worlds. Naturalness and intuitiveness of physical interaction provide the player easy adaptation into games as observed in the evaluation of the developed game in which players stated that they felt more like really playing the game while using the hand-held sword.

Tangible motion controlling method that is developed provides the player an easier way of interaction since users are expected to understand and interpret physicality better than abstract due to the acknowledged nature of human being inclined to physicality. Spatial cognition of the players is expected to be improved by the physicality the improved system provides. As another improvement, physical object that is introduced into motion controlling provides tactile feedback to the player which helps him associate interaction mechanics easily. Tangible object provides the player with physical models of the actions performed, making the interaction easier. Realistic physical interaction provide active discovery and more comfortable controlling with less mental load incurred, enabling the player enjoy more challenging games having fun rather than enforcing his brain.

Presence of the tangible object yields to fairer gameplay by eliminating rapid biased moves with its self weight and volume imposing motion constraints.

Besides the improvements mentioned so far, physical interaction also brings a fun factor into the gameplay. New possible actions are offered with the developed system that could not be performed with only using human skeleton but made possible with the recognized object serving as an extension to the body. As an example, new actions are made possible by the introduction of rolling axis with the help of the hand-held tangible object that is introduced into motion controlling with the developed algorithm. The tangible object provides alternative ways of gameplay.

Another advantage of the developed motion controlling system is that it requires less effort spent for learning abstract game controlling movements since it eliminates unnatural gestures that need to be learned by the player, providing a more natural and realistic form of interaction. The developed system may also help in improving children's skills since it enables them interaction with physicality and children are known to benefit from tactile learning. Moreover, incorporating their belonged objects into video games may provide richness in gameplay, nurturing their imagination. Children may enjoy incorporating their favorite toy into gameplay a lot. From another point of view, tangible interaction provides the user social confidence, enabling him perform realistic and meaningful moves rather than performing meaningless abstract actions, especially in the presence of other people.

In addition to the benefits of the developed algorithm that are mentioned so far, there may be disadvantages of the method as well. First, the developed motion controlling system may not be suitable for every gameplay style. The developed method is found to be suitable for video games of real life activities that are performed with tangible objects and may be suitable for other game designs which utilize tangible objects for controlling appropriately, such as using a wand in a spell casting game of a magical theme. As another disadvantage, the interaction may be tiring as compared to touchless interaction since the player is required to carry an object during gameplay. To overcome this problem of getting tired, light objects such as plastic toys may be used for interaction.

It should be considered that results of the study are strictly dependent on the experiment game. A different experiment game design may have been led to different results. Here, a representative motion controlled game that utilizes the hand-held object is chosen to be designed deliberatively. So, these results cannot be overgeneralized to be applicable for all kinds of video games, but are outcomes of the designed experiment game. Results of the study are considered to be valid for video games that are simulating real world physical activities that are performed with tangible objects.

Control game interaction design may also affect the results since the developed interaction style is compared with the control version. A complex gestural design involving many intricate elements in it may yield to a favor on the experiment version, which provides a natural interaction. In the study, gestural design of the control game is kept as simple and self expressive as possible but it still needed to be considered that results of the user study are open to interference from gestural design.

Another factor that may affect the results is the participants. Although the experiment participants are tried to be kept as representative as possible, consisting of experienced and non-experienced Kinect users; most of the participants were familiar with computers and had prior gaming experience, which may have been affected the results.

CHAPTER 5

CONCLUSION AND FUTURE WORK

This chapter presents conclusion of the study and future work implications that may inspire subsequent works.

5.1. Conclusion

In this work, an alternative tangible video game interaction method using Microsoft Kinect is proposed and implemented that enables the players control games with common found tangible objects such as plastic toys and bottles. There are many advantages of tangible interaction that add to the richness of the gameplay, throughout the work, these advantages are presented and discussed. A game is developed with two interaction versions, one achieved with the current motion tracking scope of Kinect as the control version and one achieved with the developed tangible object recognition algorithm as the experiment version. A user study is performed to assess the effects of the developed interaction style and results are analyzed statistically to validate outcomes of the study. Results of the user study revealed that the developed interaction system provided an improved gameplay experience for the experimental game. Improvements that are achieved with the developed interaction system are discussed. Limitations of the work and future work indications are presented.

5.2. Future Work

As future improvements, the system may support the user's carrying objects in both hands and changing them during gameplay. Necessary adjustment should be made into gameplay in this case, reflecting the hand change of the objects into gameplay with their dimensions. There may be constructed 3D models of the hand-held objects and incorporated into gameplay by recognizing objects that the user is holding and incorporating them into gameplay providing a similar virtual appearance to real hand-held objects, as another future endeavor. Also, additional user studies may be conducted to assess the effects of physical interaction on different game designs since game design has a direct effect on user experience with different interaction techniques.

REFERENCES

[1] “Wii Official Site at Nintendo” Internet: http://www.ninten do.com/wii [Aug. 08, 2012].

[2] “PlayStation® Move Motion Controller - PlayStation® 3 Move Info, Games & Updates” Internet: http://us.playstation.com/ps3/play sta-tion-move/ [Aug. 08, 2012].

[3] “Kinect - Xbox.com” Internet: http://www.xbox.com/en-US/kinect [Aug. 08, 2012].

[4] “Kinect Sales Surpass Ten Million - Xbox.com” Internet: http://ww w.xbox.com/en-US/Press/Archive/2011/0308-Ten-Million-Kinects [Nov. 13, 2011].

[5] E. Wiebe, J. Minogue, M. Jones, J. Cowley, and D. Krebs. 2009. "Haptic feedback and students' learning about levers: Unraveling the effect of simulated touch." Comput. Educ. 53, 3 (November 2009), 667-676. DOI=10.1016/j.compedu.2009.04.004 http://dx.do i.org/10.1016/j.compedu.2009.04.004.

[6] L. Yao, S. Dasgupta, N. Cheng, J. Spingarn-Koff, O. Rudakevych, and H. Ishii. 2011. "Multi-jump: jump roping over distances." In Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems(CHI EA '11). ACM, New York, NY, USA, 1729-1734. DOI=10.1145/1979742.1979836 http://doi.acm.org/10.1145/1979742.1979836.

[7] A. Cheok, M. Inakage, and N. Lee. 2011. "Letter from the editors." Comput. Entertain. 9, 2, Article 5 (July 2011), 4 pages. DOI=10.1145/1998376.1998382 http://doi.acm.org/10.1145/1998376.1998382.

[8] E. Hornecker. 2011. "The role of physicality in tangible and embodied interactions." Interactions 18, 2 (March 2011), 19-23. DOI=10.1145/1925820.1925826 http://doi.acm.org/10.1145/1925820.1925826.

[9] Lok, S. Naik, M. Whitton, and F. Brooks, Jr.. 2003. "Effects of handling real objects and self-avatar fidelity on cognitive task performance and sense of presence in virtual environments." Presence: Teleoper. Virtual Environ. 12, 6 (December 2003), 615-628. DOI=10.1162/105474603322955914 http://dx.doi.org/10.1162/105474603322955914.

[10] L. Spina-Caza. 2010. "Objects in play: virtual environments and tactile learning." InProceedings of the fourth international conference on Tangible, embedded, and embodied interaction (TEI '10). ACM, New York, NY, USA, 299-300. DOI=10.1145/1709886.1709952 http://doi.acm.org/10.1145/170988 6.1709952.

[11] J. Wachs, M. Kölsch, H. Stern, and Y. Edan. 2011. "Vision-based hand-gesture applications." Commun. ACM 54, 2 (February 2011), 60-71. DOI=10.1145/1897816.1897838 http://doi.acm.org/10.1145/1897816.1897838.

[12] Thomas. 2010. "The future of entertainment: how play and engaging experience can contribute to the society." Comput. Entertain. 8, 4, Article 22 (December 2010), 3 pages. DOI=10.1145/1921141.1921142 http://doi.acm.org/10.1145/1921141.1921142.

[13] Saffer. 2008. "Designing Gestural Interfaces: Touchscreens and Interactive Devices." O'Reilly Media, Inc..

[14] B. Zaman, V. Abeele, P. Markopoulos, and P. Marshall. 2009. "Tangibles for children,: the challenges." In Proceedings of the 27th international conference extended abstracts on Human factors in computing systems (CHI EA '09). ACM, New York, NY, USA, 4729-4732. DOI=10.1145/1520340.1520727 http://doi.acm.org/10.1145/1520340.1520727.

[15] J. LaViola and F. Keefe. 2011. "3D spatial interaction: applications for art, design, and science." In ACM SIGGRAPH 2011 Courses (SIGGRAPH '11). ACM, New York, NY, USA, , Article 1 , 75 pages. DOI=10.1145/2037636.2037637.

[16] A. Bowman, J. Wineman, L. F. Hodges, and D. Allison. 1998. "Designing Animal Habitats Within an Immersive VE." *IEEE Comput. Graph. Appl.* 18, 5 (September 1998), 9-13. DOI=10.1109/38.708555 http://dx.doi.org/10.1109/38.708555.

[17] "Kinect Sports: Season Two - Xbox.com" Internet: http://marketplace.xbox.com/en-US/Product/Kinect-Sports-Season-Two/66acd000-77fe-1000-9115-d8024d5309d6 [Aug. 08, 2012].

[18] M. E. Moore and J. Sward. 2007. "Introduction to the Game Industry", Prentice Hall, 2007. (ISBN: 0-13-168743-3), Chapters 8, pp.249-276.

[19] "E3 11: Kinect Star Wars presentation – YouTube" Internet: http://www.youtube.com/watch?v=OOmMHNsDtU8 [Aug. 16, 2012].

[20] A.Harfield, I. Jormanainen, and H. Shujau. 2009. "First steps in distributed tangible technologies: a virtual tug of war." In Proceedings of the 8th International Conference on Interaction Design and Children (IDC '09). ACM, New York, NY, USA, 178-181. DOI=10.1145/1551788.1551822 http://doi.acm.org/10.1145/1551788.1551822.

[21] H. Ishii, C. Wisneski, J. Orbanes, B. Chun, and J. Paradiso. 1999. "PingPongPlus: design of an athletic-tangible interface for computer-supported cooperative play." In Proceedings of the SIGCHI conference on Human factors in computing systems: the CHI is the limit (CHI '99). ACM, New York, NY, USA, 394-401. DOI=10.1145/302979.303115 http://doi.acm.org/10.1145/302979.303115.

[22] L. Yao, S. Dasgupta, N. Cheng, J. Spingarn-Koff, O. Rudakevych, and H. Ishii. 2011. "RopePlus: bridging distances with social and kinesthetic rope games." In Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems (CHI EA '11). ACM, New York, NY, USA, 223-232. DOI=10.1145/1979742.1979611 http://doi.acm.org/10.1145/1979742.1979611.

[23] K. Ryokai, S. Marti and H. Ishii. 2004 "I/O Brush: Drawing with Everyday Objects as Ink." In Proceedings of Conference on Human Factors in Computing Systems (CHI '04), (Vienna, Austria, April 24 - April 29, 2004).

[24] "Kinect Sports - Xbox.com" Internet: http://marketplace.xbox.com/en-US/Product/Kinect-Sports/66acd000-77fe-1000-9115-d8024d5308c9 [Aug. 08, 2012].

[25] "Harry Potter and the Deathly Hallows™ - Part 1 - Xbox.com" Internet: http://marketplace.xbox.com/en-US/Product/Harry-Potter-and-the-Deathly-Hallows-Part-1/66acd000-77fe-1000-9115-d802454108f9 [Aug. 08, 2012].

[26] "LucasArts.com | Kinect Star Wars" Internet: http://www.lucasarts.com/games/kinectstarwars/ [Aug. 12, 2012].

[27] "Fruit Ninja Kinect - Xbox.com" Internet: http://marketplace.xbox.com/en-US/Product/Fruit-Ninja-Kinect/66acd000-77fe-1000-9115-d80258410b79 [Aug. 12, 2012].

[28] “Welcome to Rovio.com! - Rovio Entertainment Ltd” Internet: http://www.rovio.com/ [Aug. 12, 2012].

[29] “Rovio’s Angry Birds Hits 1 Billion Cumulative Downloads” Inetrnet: http://thenextweb.com/mobile/2012/05/09/rovios-angry-birds-titles-hit-1-billion-cumulative-downloads/ [Aug. 12, 2012].

[30] “Super Angry Birds: a spring-loaded USB slingshot peripheral | News | TechRadar” Internet: http://www.techradar.com/news/gaming/super-angry-birds-a-spring-loaded-usb-slingshot-peripheral-1091925 [Aug. 12, 2012].

[31] “Cell Phones | 4G Phones | Android Smartphones | T-Mobile” Internet: http://www.t-mobile.com/ [Aug. 12, 2012].

[32] “T-Mobile: Angry Birds in real life | Creative Criminals” Internet: http://creativecriminals.com/outdoor/t-mobile-angry-birds-in-real-life/ [Aug. 12, 2012].

[33] “http://www.electricpig.co.uk/2011/01/07/play-angry-birds-for-real/" Internet: http://www.electricpig.co.uk/2011/01/07/play-angry-birds-for-real/ [Aug. 12, 2012].

[34] “WAVI Xtion :: Intuitive living room experience :: ASUS” Internet: http://event.asus.com/wavi/product/WAVI_Xtion.aspx [Aug. 12, 2012].

[35] "Experience the Future of Smart TV, Now | Samsung Smart TV" Internet: http://www.samsung.com/sg/smarttv/ [Aug. 12, 2012].

[36] "BBC News - Touchless smartphones and TVs could be on sale in 2012" Internet: http://www.bbc.com/news/technology-15970019 [Aug. 12, 2012].

[37] "At the Heart of the Wii, Micron-Size Machines - New York Times" Internet: view-source:http://www.nytimes.com/2006/12/21/technology/21howw.html [Aug. 08, 2012].

[38] T. Shiratori and J. K. Hodgins. 2008. "Accelerometer-based user interfaces for the control of a physically simulated character". *ACM Trans. Graph.* 27, 5, Article 123 (December 2008), 9 pages. DOI=10.1145/1409060.1409076 http://doi.acm.org/10.11 45/1409060.1409076.

[39] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. 2011. "Real-time human pose recognition in parts from single depth images." In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11). IEEE Computer Society, Washington, DC, USA, 1297-1304. DOI=10.1109/CVPR.2011.5995316 http://dx.doi.org/10.1109/CVPR.2011.5995316.

[40] "Amazon.com: Intec G8692 Xbox Kinect Sports Pack: Electronics" Internet: http://www.amazon.com/Intec-G8692-Xbox-Kinect-Sports/dp/B00526ICXO [Aug. 16, 2012].

[41] "[Accessories] Intec G8692 Xbox Kinect Sports Pack - Kinect Forum page" Internet: http://123kinect.com/kinect-forums/Thread-Accessories-Intec-G8692-Xbox-Kinect-Sports-Pack?highlight=sports+pack [Aug. 16, 2012].

[42] "Human Interface Guidelines, Kinect for Windows v1.5.0" Official Microsoft Kinect for Windows Webpage, Learning resources and documentation. Internet: http://www.microsoft.com/en-us/kinectforwindows/develop/learn.aspx [Jul. 26, 2012].

[43] 2011 Microsoft Corporation. "Readme for Kinect for Windows SDK" Internet: http://www.microsoft.com/en-us/kinectforwindows/develop/readme.htm [Jun. 5, 2012].

[44] "Unity - Game Engine" Internet: http://unity3d.com/ [Aug. 9, 2012].

[45] ".NET Downloads, Developer Resources & Case Studies | Microsoft .NET Framework" Internet: http://www.microsoft.co m/net/ [Aug. 6, 2012].

[46] ".NET Framework" Internet: http://msdn.microsoft.com/en-US/netframework/cc378097 [Aug. 6, 2012].

[47] "Unity3D" Internet: http://zigfu.com/en/zdk/unity3d/ [Aug. 8, 2012].

[48] M. Raymond, D. Pontier,A. Dufour and A. Moller. 1996. "Frequency-dependent maintenance of left handedness in humans." Proc. R. Soc. B 263, 1627–1633.

[49] M. Brown, A. Kehoe, J. Kirakowski and I. Pitt. 2010. “Beyond the Gamepad: HCI and Game Controller Design and Evaluation” Chapter 12 in Regina Bernhaupt (ed), "Evaluating User Experience in Games". Springer Verlag, London, UK, 197-219. DOI= 10.1007/978-1-84882-963-3_12 http://dx.doi.org/10.1007/978-1-84882-963-3_12.

[50] E. Calvillo-Gámez, P. Cairns and A. Cox. 2010. “Assessing the Core Elements of the Gaming Experience.” Chapter 4 in Regina Bernhaupt (ed), "Evaluating User Experience in Games". Springer Verlag, London, UK, 47-71. DOI= 10.1007/978-1-84882-963-3_4 http://dx.doi.org/10.1007/978-1-84882-963-3_4.

APPENDICES

APPENDIX A – PARTICIPANT INFORMATION FORM

Name:

Age:

Occupation:

I am Right Handed ☐ Left Handed ☐

How often do you play video games?:

Less than once a week ☐ Once a week ☐ More than Once a Week ☐

How many hours a week do you spend playing video games?:

Have you played any game with Kinect before?:

If so, in total how many hours/days did you spend playing with Kinect?:

APPENDIX B – CONTROL GAME QUESTIONNAIRE

Control Questions

I felt that the control was on me during the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I interacted with the game easily.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

The game was easy to control.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Enjoyment Questions

I enjoyed playing the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I liked the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I would play this game again.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Frustration Question

I was frustrated while playing the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Gameplay Questions

I understood how to play the game easily.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

It was easy to score.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

The game was easy.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Immersion Question

I forgot everything around me during gameplay.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Learning Burden Questions

It was difficult to learn the gestures that are required to control the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

It was difficult to remember which movement to perform to do the required actions during the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Motion Recognition Accuracy Question

Motion recognition was sensitive.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Naturalness Questions

Controls were intuitive.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I felt that I already knew how to play the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I think that the interaction was natural.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

It was familiar for me to control the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Performance Questions

I felt like I really performed the actions in the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I felt what was happening in the game was my own doing.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I think I performed well on the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

It was easy to project the actions I desired to perform into the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Physical Sense Question

I disliked the feeling of the lacking that is caused by holding nothing physical in my hand during gameplay.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

APPENDIX C – EXPERIMENT GAME QUESTIONNAIRE

Control Questions

I felt that the control was on me during the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I interacted with the game easily.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

The game was easy to control.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Enjoyment Questions

I enjoyed playing the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I liked the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I would play this game again.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Frustration Question

I was frustrated while playing the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Gameplay Questions

I understood how to play the game easily.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

It was easy to score.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

The game was easy.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Immersion Question

I forgot everything around me during gameplay.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Learning Burden Questions

It was difficult to learn the gestures that are required to control the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

It was difficult to remember which movement to perform to do the required actions during the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Motion Recognition Accuracy Question

Motion recognition was sensitive.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Naturalness Questions

Controls were intuitive.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I felt that I already knew how to play the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I think that the interaction was natural.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

It was familiar for me to control the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

Performance Questions

I felt like I really performed the actions in the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I felt what was happening in the game was my own doing.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

I think I performed well on the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

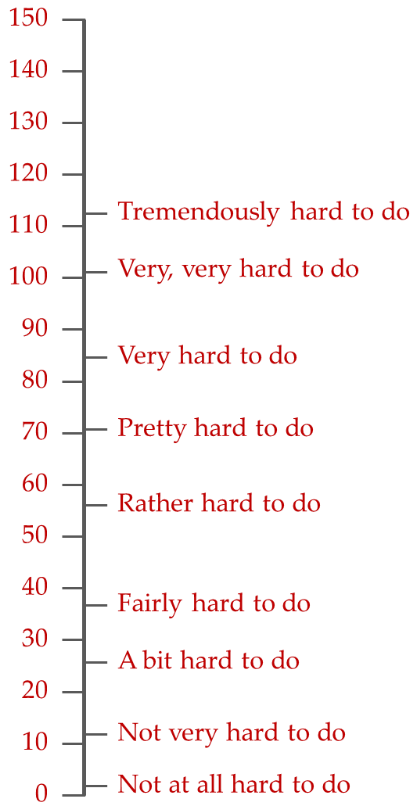
It was easy to project the actions I desired to perform into the game.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

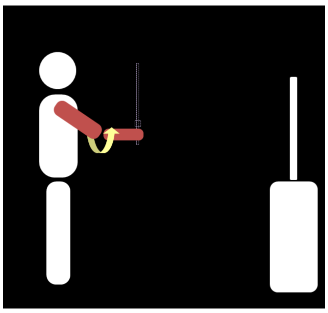
Physical Sense Question

I disliked the feeling of the lacking that is caused by holding nothing physical in my hand during gameplay.
Strongly Disagree □ Disagree □ Neutral □ Agree □ Strongly Agree □

APPENDIX D – CONTROL GAME SMEQ

Please indicate the mental effort you spent during playing the game, by drawing a straight horizontal line to the level that describes it most.



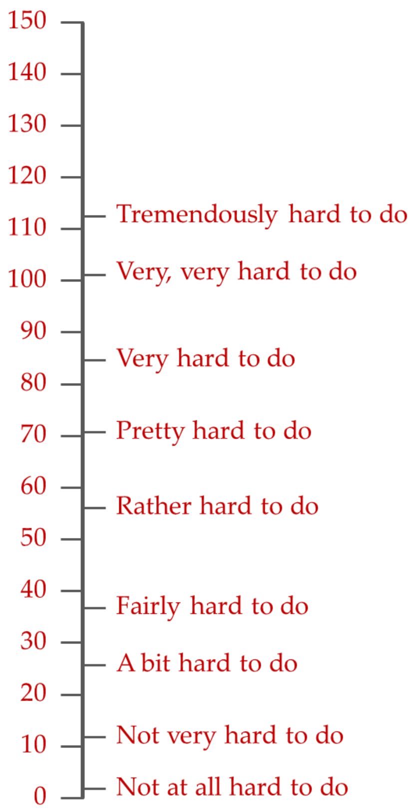


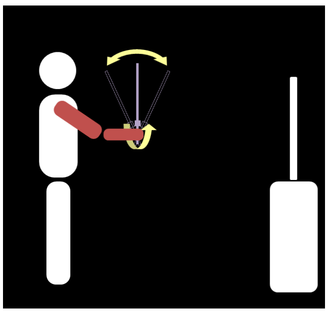
Hoopla Imaginary Sword

Disk catching game that is played by holding nothing physical

APPENDIX E – EXPERIMENT GAME SMEQ

Please indicate the mental effort you spent during playing the game, by drawing a straight horizontal line to the level that describes it most.





Hoopla Real Sword

Disk catching game that is played by holding a toy sword in hand

APPENDIX F – CONTROL GAME USER EXPERIENCE STATEMENT

Please indicate the most positive and negative experience for the controlling style you tested in your own words, in a sentence or so.

Hoopla Imaginary Sword
Disk catching game that is played by holding nothing physical

Positive Experience:

Negative Experience:

APPENDIX G – EXPERIMENT GAME USER EXPERIENCE STATEMENT

Please indicate the most positive and negative experience for the controlling style you tested in your own words, in a sentence or so.

Hoopla Imaginary Sword
Disk catching game that is played by holding nothing physical

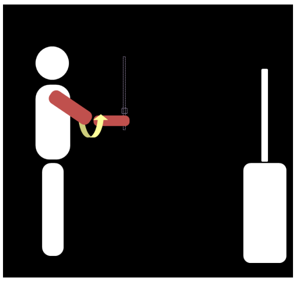
Positive Experience:

Negative Experience:

APPENDIX H – USER PREFERENCE FORM

Please indicate which game's controlling style would you prefer comparing two game pairs indicated below.

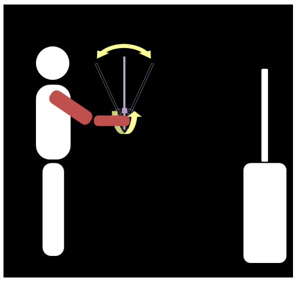
Game A



Hoopla Imaginary Sword

Disk catching game that is played by holding nothing physical

Game B



Hoopla Real Sword

Disk catching game that is played by holding a toy sword in hand

- ☐ **much prefer controlling style of game A**
- ☐ **prefer controlling style of game A**
- ☐ **neutral between controlling styles of two games**
- ☐ **prefer controlling style of game B**
- ☐ **much prefer controlling style of game B**

TEZ FOTOKOPİ İZİN FORMU

ENSTİTÜ

Fen Bilimleri Enstitüsü ☐

Sosyal Bilimler Enstitüsü ☐

Uygulamalı Matematik Enstitüsü ☐

Enformatik Enstitüsü ☒

Deniz Bilimleri Enstitüsü ☐

YAZARIN

Soyadı : BOZGEYİKLİ
Adı : LAL GAMZE
Bölümü : MODELLEME VE SİMÜLASYON/OYUN TEKNOLOJİLERİ

TEZİN ADI (İngilizce) : INTRODUCING TANGIBLE OBJECTS INTO MOTION CONTROLLED GAMEPLAY USING MICROSOFT KINECT

TEZİN TÜRÜ : Yüksek Lisans ☒ Doktora ☐

1. Tezimin tamamı dünya çapında erişime açılsın ve kaynak gösterilmek şartıyla tezimin bir kısmı veya tamamının fotokopisi alınsın. ☒
2. Tezimin tamamı yalnızca Orta Doğu Teknik Üniversitesi kullancılarının erişimine açılsın. (Bu seçenekle tezinizin fotokopisi ya da elektronik kopyası Kütüphane aracılığı ile ODTÜ dışına dağıtılmayacaktır.) ☐
3. Tezim bir (1) yıl süreyle erişime kapalı olsun. (Bu seçenekle tezinizin fotokopisi ya da elektronik kopyası Kütüphane aracılığı ile ODTÜ dışına dağıtılmayacaktır.) ☐

Yazarın imzası .... Tarih ....