# VR BASED "9-SQUARE MATRIX" AEROBIC EXERCISE FOR PREVENTION OF PHYSICAL AND COGNITIVE DECLINE IN OLDER ADULTS

**BY**

**SWETA DWIVEDI**

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF ENGINEERING (INFORMATION AND COMMUNICATION TECHNOLOGY FOR EMBEDDED SYSTEMS)**

**SIRINDHORN INTERNATIONAL INSTITUTE OF TECHNOLOGY**
**THAMMASAT UNIVERSITY**

**ACADEMIC YEAR 2017**

# VR BASED "9-SQUARE MATRIX" AEROBIC EXERCISE FOR PREVENTION OF PHYSICAL AND COGNITIVE DECLINE IN OLDER ADULTS

BY

SWETA DWIVEDI

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF ENGINEERING (INFORMATION AND COMMUNICATION TECHNOLOGY FOR EMBEDDED SYSTEMS)
SIRINDHORN INTERNATIONAL INSTITUTE OF TECHNOLOGY
THAMMASAT UNIVERSITY
ACADEMIC YEAR 2017

VR BASED "9-SQUARE MATRIX" AEROBIC EXERCISE FOR PREVENTION OF PHYSICAL AND COGNITIVE DECLINE IN OLDER ADULTS

A Thesis Presented

By

SWETA DWIVEDI

Submitted to

Sirindhorn International Institute of Technology

Thammasat University

In partial fulfillment of the requirements for the degree of

MASTER OF ENGINEERING (INFORMATION AND COMMUNICATION TECHNOLOGY FOR EMBEDDED SYSTEMS)

Approved as to style and content by

Advisor and Chairperson of Thesis Committee ______________________

(Assoc. Prof. Dr. Ekawit Nantajeewarawat)

Co-Advisor ______________________

Surapa T.

(Asst. Prof. Dr. Surapa Thiemjarus)

Committee Member and

Chairperson of Examination Committee ______________________

小池康晴

(Prof. Dr. Yasuharu Koike)

Committee Member ______________________

T.

(Prof. Dr. Thanaruk Theeramunkong)

JUNE 2018

# Abstract

VR Based "9-Square Matrix" Aerobic Exercise for Prevention of Physical and Cognitive Decline in Older Adults

By

SWETA DWIVEDI

Bachelor of Science, Sirinhorn International Institute of Technology, 2012

Master of Engineering, Sirinhorn International Institute of Technology, 2018

Medical and technological advancements enable a better quality of life than was previously possible. Currently, there exists a wide variety of devices and sensors which enable the convergence of these two fields. Serious games and exergames show promising results for using technology as a tool for rehabilitation and improving the health and self-efficacy of the elderly community. This research focuses on the implementation of the traditional “9-Square Matrix” aerobic exercise in the form of an exergame. The implementation is done using a normal webcam and the widely popular Microsoft Kinect sensor. The outcome of the experiments shows that the webcam and the Kinect sensor provides an acceptable rate of success when tested with subjects.


**Keywords**: Exergame, Image Processing, Virtual Reality

## Acknowledgments

First and Foremost, I would like to extend my sincere gratitude to my advisor Assoc. Prof. Dr. Ekawit Nantajeewarawat, whose constant guidance, support and motivation ensured I would keep pushing myself to achieve all the milestones. Furthermore, I would like to thank my co-advisor, Asst. Prof. Dr. Surapa Thiemjarus, who encouraged me to begin this journey. For her guidance, patience and constructive feedback on each stage of this research. I could not have completed this journey without their support and encouragement.

I would like to thank all the committee members, Prof. Dr. Yasuharu Koike and Prof. Dr. Thanaruk Theeramunkong, for their valuable advice and comments during the progress presentation. The TAIST-Tokyo tech scholarship has provided me with an opportunity to learn and grow, opening new doors towards the future.

Lastly, this research would not have been completed without the constant love and support of my beloved friends and family members, who have been there for me throughout my life.

Table of Contents


| Chapter | Title | Page |
|---|---|---|
| | Sign page | i |
| | Abstract | ii |
| | Acknowledgments | iii |
| | Table of Contents | iv |
| | List of Tables | vii |
| | List of Figures | ix |
| 1 | Introduction | 1 |
| | 1.1 Motivation and Objectives | 2 |
| | 1.2 Thesis Structure | 4 |
| 2 | Related Works | 5 |
| | 2.1 The Design of Exergames for Older Population | 5 |
| | 2.1.1 A Framework to Analyze the Effects of Exergames on Individuals | 5 |
| | 2.1.2 Design Guidelines for Exergames targeting the Elderly Population based on Microsoft Kinect Sensor | 6 |
| | 2.2 Application of Exergames for Older Adults | 8 |
| | 2.2.1 Endurance Training | 8 |
| | 2.2.2 Resistance and Strength Training | 9 |
| | 2.2.3 Sensory-motor Training | 9 |
| | 2.3 Comparison of Different Wireless Motion Sensors for Implementing Exergames | 10 |
| | 2.4 Existing VR-Based Exergames for Elderly Population and their Outcomes | 13 |
| | 2.4.1 Studies based on Exergames using the Nintendo Wii | 13 |
| | 2.4.2 Studies based on Exergames using the Dance Step-pads | 14 |
| 3 | “9-Square Matrix” Aerobic Exercise Recognition using Image Processing | 16 |
| | 3.1 System Design | 16 |
| | 3.1.1 Framework and Experimental Scheme | 16 |

3.1.2 Camera Calibration and Computing Re-Projection Error 17
3.1.3 User Tracking based on Color Thresholding and Image Moments 19
3.2 Experimentation and Results 20
3.2.1 Detecting Optimal Camera Placement 20
3.2.2 User Tracking and Game Mechanics 22
3.3 Evaluation 24

4 VR-Based “9-Square Matrix” Aerobic Exercise for Preventing Physical and Cognitive Decline in Older Adults 25
4.1 Design for the VR-Based "9-Square Matrix" using Microsoft Kinect Sensor 25
4.1.1 Determining the Setup Variables and Criteria for Sensor Placement 26
4.1.2 Creating a Model to Determine the Sensor Pose (Tilt and Elevation) based on User Height and Matrix Distance 27
4.1.3 Detecting the Pattern and User Tracking 29
4.2 Experimentation and Result 30
4.2.1 Optimal Sensor Placement and Depth Accuracy Estimation 30
4.2.2 User Tracking and Game Mechanics 31
4.2.3 Comparison of Implementation Techniques for "9-Square Matrix" Aerobic Exercise Recognition using Webcam and Microsoft Kinect Sensor 32
4.3 Evaluation 33

5 “9-Square Matrix” Exergame 35
5.1 Game Design 35
5.2 System Architecture 36
5.2.1 User Interface and Design Flow 36
5.2.2 Game Mechanics 37
5.2.3 The Results of Implementing "9-Square Matrix" Aerobic Exergame using Kinect 39

6 Conclusion and Future Work 41

References 42

Appendices 45

Appendix A 46
Appendix B 49
Appendix C 55

# List of Tables

**Tables** **Page**


1.1 Step Sequence for "9-Square Matrix" Aerobic Exercise 3

2.1 UCD Protocol of Exergame Design Phase for Older Adults 6

2.2 Existing Studies on Exergames based on Resistance and Strength Training 10

3.1 Re-Projection Error (mm) for Camera Placement 21

3.2 Probability Error for Detection of "9-Square Matrix" 23

4.1 Depth Approximation Error for Localized Corner Points 31

4.2 Probability of Error Detection 31

5.1 Probability of Insertion, Deletion and Substitution Error(s) for Kinect with 10 test subjects 39

A.1 Assessment Tests to Measure the Physical Functioning in Older Adults 46

B.1 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 1 49

B.2 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 2 49

B.3 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 3 49

B.4 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 4 50

B.5 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 5 50

B.6 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 6 50

B.7 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 7 50

B.8 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 8 51

B.9 Depth Data for Tilt Angle: -26°, Elevation: 189cm and Distance: 170cm 51

B.10 Depth Data for Tilt Angle: -23°, Elevation: 180cm and Distance: 180cm 51

B.11 Depth Data for Tilt Angle: -20°, Elevation: 170cm and Distance: 190cm 51

B.12 Depth Data for Tilt Angle: -16°, Elevation: 156cm and Distance: 200cm 51

B.13 "9-Square Matrix" Exergame Error Sequence for Subject 1 52

B.14 “9-Square Matrix” Exergame Error Sequence for Subject 2 52
B.15 “9-Square Matrix” Exergame Error Sequence for Subject 3 52
B.16 “9-Square Matrix” Exergame Error Sequence for Subject 4 53
B.17 “9-Square Matrix” Exergame Error Sequence for Subject 5 53
B.18 “9-Square Matrix” Exergame Error Sequence for Subject 6 53
B.19 “9-Square Matrix” Exergame Error Sequence for Subject 7 53
B.20 “9-Square Matrix” Exergame Error Sequence for Subject 8 54
B.21 “9-Square Matrix” Exergame Error Sequence for Subject 9 54
B.22 “9-Square Matrix” Exergame Error Sequence for Subject 10 54

# List of Figures

**Figures** **Page**


| | |
|---|---|
| 1.1 “9-Square Matrix” Step Patterns | 3 |
| 2.1 Nintendo Wii Console and Accessories | 11 |
| 2.2 Microsoft Kinect Sensor | 12 |
| 3.1 Camera Calibration and Distortion Removal | 18 |
| 3.2 HSV Color Thresholding | 19 |
| 3.3 “9-Square Matrix” Setup using Webcam | 22 |
| 4.1 VR-Based “9-Square Matrix” Process Flow | 27 |
| 4.2 Sensor Setup Parameters for “9-Square Matrix” | 28 |
| 4.3 “9-Square Matrix” Exergame Setup | 34 |
| 5.1 The User Interface for “9-Square Matrix” Exergame: (a) Main Menu, (b) Instruction Screen, (c) Settings Menu, (d) Scores Page | 37 |
| 5.2 Matrix Detection and Capture for “9-Square Matrix” Exergame | 37 |
| 5.3 Game Play User Interface | 38 |
| 5.4 “9-Square Matrix” Exergame Implementation Experiment | 40 |
| A.1 Find Chessboard Algorithm | 48 |

# Chapter 1
# Introduction

The world population is slowly seeing a rise in the aging demographic due to falling fertility rates and longer life expectancies. Hence, it has become crucial to understand the implications of this trend on social, economic and health-care sectors. According to the world population aging report 2017 [1], it has been indicated that by 2030, senior citizens (age > 60) will outnumber children below the age of 10.

Aging brings about numerous physiological and psychological changes which affect the overall health and well-being of individuals. The process of aging also makes individuals over a certain age more susceptible to diseases such as heart disease, stroke, and cancer. The risk factors for such diseases are also influenced by lifestyle choices and environmental factors. However, one of the most decapacitating age-related illness includes dementia and advanced form of dementia known as Alzheimer’s disease. The chances of an individual developing some form of dementia increase drastically with advanced age. The result of which is the inability to perform day to day activities, and dependence on a fulltime caretaker.

Numerous studies and research have concluded that physical inactivity in midlife and advanced age increase the risk of an individual developing cardiovascular diseases and mental conditions such as dementia [2]. Moderate to intense exercise and physical activity on a daily basis maintains cardiorespiratory functions, motor abilities, prevents muscle loss and, the onset of cognitive decline [3]. To sustain the health benefits of exercise in older adults, it is important to engage in regular exercise for 30 minutes per day and 5 days per week.

The development of innovative new devices and sensors in the gaming sector of the tech industry has enabled immersive experience for the users in the form of Virtual Reality (VR). The popular VR devices include the Microsoft Kinect sensor, SonyEyetoy Kinetics, and Nintendo Wii. The emergence of these devices has propelled a new segment of games known as serious games and exergames.

Serious games and exergames are a category of games that are designed to obtain a particular objective such as rehabilitation, exercise, education, and training. Many of these games are targeted towards the older population for rehabilitation and achieving a daily fitness target. Exergames are specifically designed for the purpose of physical exertion and mostly involve the gamification of traditional exercise routines.

### 1.1 Motivation and Objectives

The motivation for this research lies in pursuing older adults to engage in regular aerobic exercise routine by creating a VR-based exergame on the “9-Square Matrix” aerobic exercise.

The “9-Square Matrix” or “Tarang-9-Chong” is a traditional aerobic exercise developed by Dr. Krabuanrat in 1995 [4]. This exercise routine is very popular among autistic children and the elderly community in Thailand. A study conducted with autistic children performing the exercise on a regular basis proved that there are numerous benefits such as overall improvement in their moods, emotions, language skills and motor abilities [5]. The exercise routine involves stepping on a matrix of 9 squares in a sequence followed by one of the 8 dance step patterns (shown in Table Table 1.1) along the rhythm of the music. The dance step sequence follows a shape pattern that can be easily remembered (Figure 1.1).

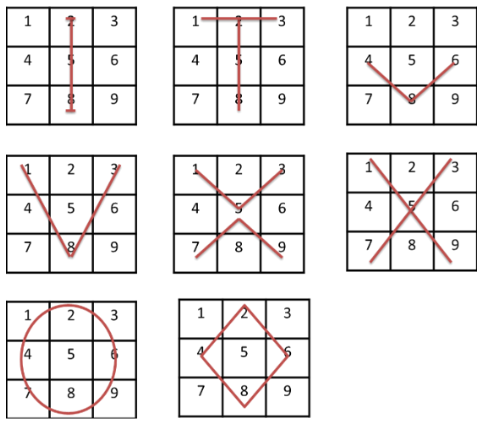


Figure 1.1 “9-Square Matrix” Step Patterns

In this study, implementation of the “9-Square Matrix” exergame is done using both a normal webcam and the Microsoft Kinect sensor. The webcam implementation solely relies on image processing techniques for matrix detection and user tracking, whereas the Microsoft Kinect sensor has an additional depth camera and relies on sophisticated machine learning algorithms for user tracking. In order to minimize errors and increase the reliability of the detection for each implementation, it is crucial to determine the optimal placement or setup for the devices such that the observability of both the matrix and the users are maximized. Once the optimal setup has been determined in each case, experiments with 8 and 10 test subjects will be performed to determine the accuracy of the implementation by recording the data and checking for insertion, deletion and substitution errors in tracking.

Table 1.1 Step Sequence for “9-Square Matrix” Aerobic Exercise

| **Step** | **Step Shape** | **Step Sequence** | |
|---|---|---|---|
| | | **Left** | **Right** |
| 1 | I | 8,5,2 | 8,5,2 |
| 2 | T | 8,5,2,1,2,5,8 | 8,5,2,3,2,5,8 |
| 3 | v | 8,4,8 | 8,6,8 |
| 4 | V | 8,1,8 | 8,3,8 |
| 5 | x | 8,1,5,7,5,8 | 8,3,5,9,5,8 |
| 6 | X | 8,1,5,7,5,8 | 8,9,5,3,5,8 |
| 7 | O | 8,3,7,8 | 8,1,9,8 |
| 8 | ◊ | 8,4,2,4,8 | 8,6,2,6,8 |

## 1.2 Thesis Structure

**Chapter 2** presents previous work done in the field of serious games and exergames for the elderly community using commercially available consoles. The design criteria for implementation and application of such games, and the results from the clinical trials measuring the health benefits of the intervention.

**Chapter 3** discusses the implementation of the “9-Square Matrix” using a normal webcam. The algorithms applied for detection of the matrix and user tracking. The study presents the results from the optimal camera placement experiment and actual implementation of the exergame using an error classification model with 8 test subjects.

**Chapter 4** discusses the implementation of the “9-Square Matrix” using a Microsoft Kinect sensor. The algorithms applied for detection of the “9-Square Matrix” and user tracking. The study also presents the results from the experiment for optimal sensor setup by measuring the depth accuracy of matrix detection and implementation of the exergame using the error classification model described in chapter 3 with 10 test subjects.

**Chapter 5** presents the design of the “9-Square Matrix” Exergame user interface and game mechanics along with the discussion on the experiment performed with the participants.

**Chapter 6** presents the conclusion for the techniques used in this study to achieve the implementation of the “9-Square Matrix” exergame. It also analyzes the cause of various errors introduced during gameplay with both the devices and a proposed solution to reduce the overall errors and improve the accuracy.

# Chapter 2
# Related Works

## 2.1 The Design of Exergames for Older Population

### 2.1.1 A Framework to Analyze the Effects of Exergames on Individuals

Mueller et al. [6] presents a study about the effects of exergaming on individuals by analyzing different exergames and their outcomes on the participants on every level using the 4 Lens model. The model describes an individual's experience of exergames on 4 levels:

1. **The Responding Body** describes the effects of exertion on the physiological level, such as improvement in the cardiovascular and cardiorespiratory systems and brain plasticity. The effects and benefits of exergames on the responding body is sustained even after the physical activity or exertion has ended, for example, weight loss, increased muscle mass and strengthened bone structure.

2. **The Moving Body** describes the effect on the sensory-motor skills of an individual depending on the type of exertion activity being performed. These skills include improved balance ability and reaction times, which can be enhanced and transferred onto day-to-day activities.

3. **The Sensing Body** describes the way in which the body experiences objects and context present both in real and virtual environments. Many exergames augment both physical and virtual objects to create a hybrid space to enhance the overall experience of the players.

4. **The Relating Body** describes the social interaction and communication with regards to technology. Augmenting social interaction in the form of coplayers and opponents increase the motivation for people to engage in exergames daily.

In order to understand the effectiveness of serious games and exergames, it is important to analyze the outcomes of the game using the model which describes the influence on the physiological, psychological, sensory-motor and social interaction levels.

### 2.1.2 Design Guidelines for Exergames targeting the Elderly Population based on Microsoft Kinect Sensor

Bronx et al. [7] presented a case study of user-centered design (UCD) for serious games and exergames for older adults. This study utilizes a series of mini-games using Microsoft Kinect sensor called the GameUp project. The mini-games included 3 games which focused on balance and additional 4 games targeting strength and flexibility in elderly with varying levels of difficulty. The trial involved 10 participants (2 men, 8 women) with an average age of 81.7 years and trial period lasting 3 years. The study follows a structure for user-centered design protocol and results of implementing the protocol which is divided into 4 phases (as shown in Table 2.1)

Table 2.1 UCD Protocol of Exergame Design Phase for Older Adults

| Phase | Design | Results |
|---|---|---|
| Requirement Gathering and Analysis | • Literature review<br>• Questionnaire with background information<br>• User observation while playing commercial games<br>• Group discussions regarding gameplay<br>• Existing knowledge of gameplay, taking inputs from physiotherapist | • Physiotherapist provided input for appropriate exercises<br>• Game developers assessed the suitable technology for implementation of exergames. (Wii, Kinect, etc.) |
| Design | • Observations<br>• Semi-structured interviews<br>• Group discussions | • Designing suitable game elements such as themes, graphics, movements, GUI, sounds, playability etc.<br>• Determined suitable technology to implement exercise routine defined by physiotherapist. i.e. Microsoft Kinect sensor. |

| | | |
|---|---|---|
| Implementation | • Observations<br>• Semi-structured interviews<br>• Group discussions | • Testing early prototypes and user testing<br>• Testing iteratively with users according to inputs from earlier prototypes<br>• Testing the user interface elements<br>• Testing focused on gameplay and theme |
| Evaluation | • Structured and semi-structured interviews<br>• Questionnaires<br>• Prototype trials with new participants | • Testing the final prototype |

Apart from the UCD design guideline, the study also suggests key functionality elements for exergames as follows:

1. **Speed:** The users need time to adjust and react to the exergame environment appropriately, therefore the speed of the game should be adjusted for different actions.
2. **Movements:** Age factor has an impact on the user's flexibility and motor abilities, hence the movements should be designed to not stretch the abilities to extreme leading to loss in balance or injury.
3. **Information:** Elderly users direct most of their focus on the mechanics of gameplay, hence it is better to present all useful statistics at the end of the game session.
4. **Colors and Contrasts:** most of the elderly population have diminished eyesight, making it important for graphics to be bright and have a good amount of contrast. The variation of colors used can be reduced.
5. **Small Details:** the user's focus on the gameplay mechanics can detract them from noticing small details leading to confusion about the object(s).
6. **Text/Font:** The text font used should be large, bold and clear with minimal text during gameplay. It is also crucial to provide oral feedback which is easier to follow for older adults.
7. **Menu:** The interactive buttons should be large and have sufficient space between them to avoid accidental activation.
8. **Sound:** The volume throughout the gameplay should be consistent.

## 2.2 Application of Exergames for Older Adults

Weimeyer and Kliem [8] presented a literature review on existing serious games and exergames while analyzing their success in terms of achieving specific targets and analyzing their outcomes on the individuals using the 4 Lens model. Exergames can be divided into the following categories based on their objectives:

### 2.2.1 Endurance Training

For endurance training, the goal of the exergames is to raise the energy expenditure below a minimum threshold of 600-800 kCal per week. Most of the exergames that focus on energy expenditure utilize commercially available consoles such as Nintendo Wii Sports and Wii Fit [9-15], Sony EyeToy Kinetics [16], Microsoft Kinect sensor [17, 18] and Konami Dance Dance Revolution (DDR) [19]. However, the studies concluded that highest energy expenditure observed was around 400 kCal per hour. In order for the exergames to meet the minimum required dose of PA, the overall duration of the exergame sessions need to be 2 hours per week and to achieve the optimal threshold around 7.5 hours per week.

The study presented by Wollersheim et al. [20] investigating the physical and psychosocial impact of exergames on community-dwelling older women concluded that there were no substantial physical gains by engaging in a 6-week intervention playing the Nintendo Wii Sports. However, the participants enjoyed the experience and were motivated to continue engaging in such activities.

To conclude about the studies involving exergames on energy expenditure using the 4 Lens model, the exergames had lesser impact on the physiological functions (responding body) of the body but more so on the psychological level (moving body). The exergames failed to reach the optimum level of PA but had positive effects such as increased motivation, positive mood and overall social well-being.

### 2.2.2 Resistance and Strength Training

For exergames focused on resistance and strength training, the aim is to improve the strength of the upper and lower extremities. A study by King [21] showed that embedding game-context into traditional strengthening exercises had increased the number of repetitions of the exercise. This intervention included 146 patients between the age range of 16-78 years.

Another randomized control trial by Sonhsmeyer et al. [22] involving 40 subjects was conducted using the Nintendo Wii Bowling game. The exercise group was made to play Nintendo Wii Bowling for a duration of 6 weeks, whereas the control group did not engage in any exercise program. At the end of the intervention program, the exercise group had shown significant improvements in the strength of their left and right quadriceps compared to the control.

The quality of study presented in this category is low because of the lack of detailed quantitative data, but the benefit of exergames can be seen on the sensing body and the relating body.

### 2.2.3 Sensory-motor Training

The exergames focused on sensory-motor training aim to improve the reaction times and balance in elderly people. There have been a lot of studies and trials based on the intervention of using exergames to improve sensory-motor abilities using commercially available consoles such as Nintendo Wii (Table 2.2).

According to the 4 Lens model, the studies showed an improvement on the psychological level and the sensory-motor level (sensing body). The exergames in the category of sensory-motor training had the intended effect of improving the balance ability and reaction times in the elderly population.

Table 2.2 Existing Studies on Exergames based on Resistance and Strength Training

| Author(s) | Study Type | Summary |
|---|---|---|
| Kliem and Weimeyer [23] | • Sample: 22<br>• Mean age: 47.36<br>• SD: 13.14<br>• Duration: 3 weeks,3 sessions per week (10-12min)<br>• EG: Digital game<br>• CG: Traditional<br>• Exercise | • Both the EG and CG showed improvements in 4 out of 5 balance tests.<br>• The effectiveness of the traditional exercise program was deemed better than game-based program |
| Williams et al. [24] | • Sample: 15<br>• Mean age: 70<br>• Duration: 12 weeks,<br>• 2 sessions/week.<br>• EG: Wii Fit<br>• CG: Traditional Exercise | • The participants involved older adults who had a history of falls<br>• The EG had shown improvement in their balance ability by 4th week of intervention<br>• The CG had not improved their balance skills |
| Harley et al. [12] | • Sample: 30<br>• Age: 60-94<br>• Duration: 1 year<br>• EG: Wii Bowling<br>• CG: N/A | • Qualitative study based on the experience of older adults playing Wii Bowling<br>• The study concluded that it was not too hard for older adults to adapt to new technologies and had a fun experience<br>• The game sessions had helped in enhancing the social connections between peers |
| Young et al. [15] | • Sample: 6<br>• Mean age: 84.1 ± 5.1<br>• Duration: 4 weeks,<br>• 10 sessions (20 min)<br>• EG: Wii Balance Board<br>• CG: N/A | • The participants showed an improvement in overall balance and self-efficacy |

## 2.3 Comparison of Different Wireless Motion Sensors for Implementing Exergames

Fraile et al. [25] presented a suitability analysis for wireless motion sensors available in the market for the purpose of implementing exergames. Currently, the two most popular choices of commercially available wireless motion sensors for exergaming include the Nintendo Wii and the Microsoft Kinect sensor.


Figure 2.1 Nintendo Wii Console and Accessories

Most of the early research and studies based on exergames for the elderly community is based on commercially available and the highly popular console known as the Nintendo Wii. The Nintendo Wii console includes a remote called the Wii-mote, which uses Bluetooth to relay the data from its in-built triaxial accelerometer [26]. The optical sensor in the Wii-mote can also send data related to its positioning by detecting the 5 LED lights on the sensor bar and their projection on its own infrared panel. It also contains several buttons which can be used to provide direct feedback to the game. The Nintendo Wii console provides another motion sensor called the Nunchuk that works in tandem with the Wii-mote. In addition to the Wii mote and Nunchuk, Nintendo had launched a balance board called the Wii Balance Board. The balance board contains 4 pressure sensors and is used to measure the user's center of balance. The Wii balance board has been used extensively in exergames targeted to improve the balance skills of the elderly prone to falls.

The other most widely popular choice of wireless sensor for exergames involving the elderly is the Microsoft Kinect sensor (Figure 2.2). The sensor includes an RGB camera coupled with a depth camera and an infrared projector. The infrared projects a speckle pattern which can be seen by the depth camera. The sensor calculates depth by using the triangulation method. The Kinect sensor does not require any additional accessories and detects the user skeleton by applying sophisticated machine learning algorithms. The sensor can return position data of up to 20 estimated joint values.

Fraile et al. compared the specifications of both the consoles and presented the outcome of the usage of both the technologies by creating a tool called SANDRA. It is a multiplatform browser-based game, that uses OpenNI library for handling Kinect data, wiigee for Wii controllers and, Wiiboard-simple for the Wii balance board interfaces. The gameplay consisted a team of 4 players, who played the games together and once the game ended were led to the scoring page to see their performance outcomes. The exercise included stretching out to reach different objects in the scenes placed randomly above the participants.

The challenges that occurred during the implementation of SANDRA included the crashing of the OpenNI library for Kinect. The developers had to use 1:1 movements that Kinect provides rather than substitute avatar animations in certain scenarios. The authors concluded that although the Nintendo Wii controllers were a suitable option, they needed to be adapted for the elderly users and could not be used off the shelves. The other constraints of using the Wii controllers were limited amount of data obtained by the sensor. The data returned by Wii controllers could not be used to animate the full body avatars of the players.

The authors concluded that the Microsoft Kinect sensor is a good choice as a standalone device for developing a wide variety of exergames due to its skeleton tracking abilities which can perfectly mimic the natural state of the body without any additional accessories and its ease of use.


Figure 2.2 Microsoft Kinect Sensor

## 2.4 Existing VR-Based Exergames for Elderly Population and their Outcomes

There exists a lot of studies on the usage of exergames with the elderly population for rehabilitation and increase in physical activity. Majority of these trials with the elderly community are based on the Nintendo Wii console [10, 11, 13, 27-30] in tandem with the Wii Fit game suite. The suite consists of a series of games that are mainly designed to promote physical fitness by augmenting exercises that incorporate strength training, aerobic exercise, and balance training. The other existing VR exergames include dance-based step pads [31].

To analyze the effect of exergames on the elderly community, it is important to understand the design methodologies employed by the study along with the different assessments made before and after the intervention. Most popular assessment tools employed in these research studies measure changes in balance and sensory-motor skills using TUG, BBS, FES / FES-I and Tinetti-POMA assessment tasks and questionnaires (Table A.1).

### 2.4.1 Studies based on Exergames using the Nintendo Wii

The randomized controlled trials based on Nintendo Wii mostly evaluated the gains in mobility using Timed-Up and Go (TUG) tests. The trials using TUG assessment [10, 13, 27, 29] showed an improvement for the intervention group [13, 27] compared to the control group. However, the studies [10, 29] did not show any improvement in either groups.

The following studies [10, 11, 13] used the Berg Balance Scale (BBS) to assess the improvement in the balance ability of the participants. The outcome of BBS for the studies [10, 13] showed a significant improvement for the intervention group compared to the control. Although there was an improvement of the BBS scores compared to the pre-intervention period for study [11], there was no significant difference between the intervention groups within the study.

Tinetti-POMA is used to analyze the mobility and gait in the elderly community prone to falls [11, 14, 29]. For the study [11], there was an improvement in the Tinetti-POMA scores. However, the studies [14, 29] showed improvements over the control group but not between the intervention groups within the study. The other forms of popular assessment made included FES/FES-I [27, 29] and FRT [10, 30] which shows an improvement in the study [27] and no improvement in the rest.

### 2.4.2 Studies based on Exergames using the Dance Step-pads

Schoene et al. [31] presented a randomized controlled trial with 37 participants around the mean age of 78 ± 5 years. The study included a computer game that required the participants in the exercise group to step on a step-pad following the instructions from the game. The control group did not receive any intervention. The intervention group was suggested to take part in the game sessions at least 2-3 times per week for 10-15 minutes. After 8 weeks, the follow up assessments show an improvement in the Choice Stepping Reaction Time (CSRT), Physiological Profile Assessment (PPA) composite scores, as well as the postural sway and contrast sensitivity PPA sub-component scores. Another significant improvement can be seen in the TUG dual-task.

# Chapter 3
# "9-Square Matrix" Aerobic Exercise Recognition using Image Processing

This chapter explains the design and methodology for the "9-Square Matrix" aerobic exercise implementation using image processing techniques. The study describes the algorithms used to detect the "9-Square Matrix" and perform user tracking based on a single low-cost webcam. The performance of the "9-Square Matrix" aerobic exercise recognition is further tested with 8 subjects by classifying the different types of errors detected.

## 3.1 System Design

### 3.1.1 Framework and Experimental Scheme

For the implementation of the "9-Square Matrix" using a normal Logitech webcam, it is important to achieve two objectives; (1) Detecting the "9-Square Matrix" and (2) Tracking the region of interest (ROI) successfully. Both these tasks rely solely on image processing techniques.

To ensure accurate detection of the "9-Square Matrix", it is important to understand the various factors that may affect the overall image quality. Most vision-based algorithms suffer from excessive noise introduced by poor lighting conditions and low image resolution. Other factors include radial and tangential distortion caused by cheap cameras.

The "9-Square Matrix" by design resembles a standard chessboard pattern. This allows the detection and extraction of its corner points using the find chessboard algorithm (as explained in Figure A.1) by the image processing library called OpenCV. To remove the distortion and ensure maximum accuracy in detection, there are two parameters that need to be estimated before-hand. The two parameters include the camera matrix (focal length and principal centers) and the pose of the camera.

Once both the parameters have been obtained, any corresponding point from the 3D world can be re-projected onto the screen. Hence, to determine the accuracy of the matrix, we eliminate the distortion coefficients and compare the difference between the detected corner points and the re-projected corner points using the re-projection error.

The first part of the experiment is to determine the optimal camera placement. The camera height and distance of the matrix will be altered for each setup and re-projection error will be calculated. The re-projection error will be used to decide the best setup for the "9-Square Matrix" implementation.

The second most crucial task is to detect the user's foot (ROI) and track it in real-time. In order to detect the ROI, all other objects in the scene needs to be filtered out. Here the filter is based on color thresholding. After the filter is applied, the binary image obtained is tracked by calculating the image moments. The image moments function calculates the centroid of the binary image or blob, which is the real-time coordinates of the object being tracked.

### 3.1.2 Camera Calibration and Computing Re-Projection Error

The camera's intrinsic parameters include its focal length ($f_x, f_y$) and the principal point ($c_x, c_y$), which describes the image center. The extrinsic parameters or pose is the rotation and translation of the camera in relation to an object. The intrinsic and extrinsic parameters together form the projection matrix (Equation 3.1) which can be used to project any 3D point to its corresponding 2D projection on the screen.

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3.1}$$

The camera's intrinsic and extrinsic parameters can be estimated by the process of camera calibration. The calibration process requires multiple snapshots of a calibration rig (e.g. chessboard) in various poses.

The extrinsic parameters for each pose is calculated independently using the solve PNP Ransac function and for each pose the re-projected corner points are calculated. The difference between the estimated values obtained from Equation 3.1 and the captured corner points on the screen is calculated using the root mean square error (RMSE). The RMSE (Equation 3.2) is also known as re-projection error. The average re-projection error for different chessboard views captured indicates the accuracy of the calibration process.

$$RMSError = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}} \tag{3.2}$$

For calibrating the Logitech webcam and removing distortions (radial and tangential), a 10x7 chessboard with 8cm squares was used. The pattern was captured in varying poses and a sample of 50 views were collected (Figure 3.1). The resulting average re-projection error value of 0.301135mm was obtained.

Figure 3.1 Camera Calibration and Distortion Removal

To compute the re-projection error of the "9-Square Matrix", the pose of the camera with respect to the matrix needs to be estimated before-hand. For the pose estimation problem, the camera parameters that needs to be estimated include 6 unknowns, in which the first 3 unknowns define the rotation of the camera (yaw, pitch, roll) using Euler angles and the rest 3 unknowns define the 3D translation of the camera with respect to the matrix in the x-, y- and z-directions.

Given the 3D world coordinates of the detected corner points and their corresponding 2D screen projections, the pose estimation problem can be solved using direct linear transformation (DLT) algorithm. Furthermore, the RANSAC method implements the Levenberg-Marquardt optimization to remove the outliers and minimize the re-projection error.

Once the pose for each setup has been computed, the re-projection error for all the views can be calculated individually. The view resulting in the minimum re-projection error value will be selected for the setup.

### 3.1.3 User Tracking based on Color Thresholding and Image Moments

To track an object, the region of interest must be defined first. The region of interest in this case is the user's foot. The players must wear colored slippers which can be easily detected and segmented out from the background (Figure 3.2) by performing color-based thresholding. Initially, the image frames are converted from the BGR color space to HSV color space. The HSV color space defines a color by its hue, saturation and value. It is more robust to noise introduced by various lighting conditions compared to other color spaces.

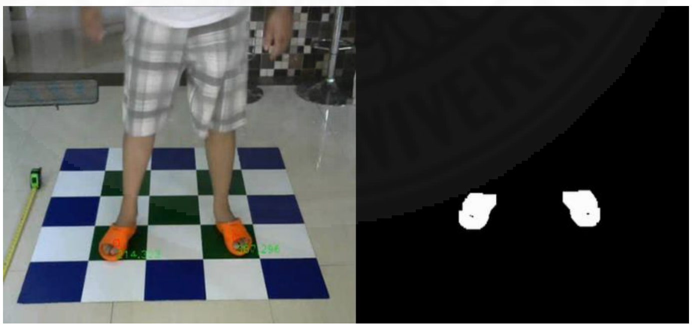

Figure 3.2 HSV Color Thresholding

The segmentation of the ROI from the background is done by performing binary thresholding on the input image for a specific color value (i.e. the slippers color). Once the binary image with only the ROI is obtained, morphological operations such as dilution and erosion are performed to eliminate any remaining noise.

The binary image which contains only the foot region of the player is used to calculate its moments. Image moments are functions that represent a statistically significant value. For tracking purposes, the 0th order moment (Equation 3.3) of the image is calculated for determining the area of the ROI.

$$\mu_{0,0} = \sum_{x=0}^{w}\sum_{y=0}^{h} f(x,y) \tag{3.3}$$

The moments function in this case is a weighted sum of all the x-y pixel values of the binary image. The centroid ($\mu_{1,0}, \mu_{0,1}$) of the object to be tracked can then be calculated by dividing the weighted average of the pixels in the x- and y- directions (Equation 3.4) individually over the area of ROI (Equation 3.5).

$$sum_x = \sum\sum x * f(x,y) \text{ and } sum_y = \sum\sum y * f(x,y) \tag{3.4}$$

$$\mu_{1,0} = \frac{sum_x}{\mu_{0,0}} \text{ and } \mu_{0,1} = \frac{sum_y}{\mu_{0,0}} \tag{3.5}$$

## 3.2 Experimentation and Results

### 3.2.1 Detecting Optimal Camera Placement

To achieve the objective of detecting the "9-Square Matrix" with maximum accuracy, it is important to determine the optimal camera placement. The goal of this experiment is to determine the setup that will maximize the observability and detection of the matrix. For each setup, a snapshot of the "9-Square Matrix" will be taken and the re-projection error will be computed.

In this experiment, the camera was mounted on a tripod with a fixed tilt angle of -27° towards the floor plane (Figure 3.3). The height of the tripod was varied from 100-160cm in 20cm intervals, whilst the matrix on the floor was shifted by 24cm each time. The re-projection error (mm) was computed for each view (Table 3.1).

Table 3.1 Re-Projection Error (mm) for Camera Placement

| Height (cm) | Range (cm) | Avg. RMSE (mm) | Min. RMSE (mm) | Distance (mm) |
|---|---|---|---|---|
| 100 | 80-152 | 32.78 | 22.04 | 152 |
| 120 | 110-182 | 27.97 | 22.85 | 182 |
| 140 | 120-192 | 25.92 | 18.74 | 192 |
| 160 | 180-252 | 23.92 | 14.38 | 252 |

The results of this experiment indicate that the re-projection error values decrease with increasing distances. According to the data, the best setup for the implementation of the “9-Square Matrix” is at the camera height of 160cm and the matrix distance of 252cm resulting in a minimum RMSE value of 14.38mm. Although the RMSE is a good estimator for determining the accuracy of the camera calibration process, it is not an ideal indicator for the best setup.

This can be attributed to the fact the re-projection error is calculated by finding the Euclidean distance between the captured points (θ) and the estimated points (θ’). However, this study [32] on camera placement shows that the traditional pin-hole camera model describes an inverse relationship between the distance of the object and its observability with respect to the camera. The reduced observability of the matrix contributed to the minimum re-projection error over large distances. The other factor that is reducing the observability of the matrix is that it’s not lying parallel to the camera increasing the foreshortening effect, reducing the accuracy of detection even further.

The better setup parameter would be at a mid-height camera elevation of 140cm and the distance of 192cm. The matrix lies on the center of the camera’s FOV, reducing the barrel distortion while not necessarily reducing the matrix observability.


Figure 3.3 “9-Square Matrix” Setup using Webcam

### 3.2.2 User Tracking and Game Mechanics

The second experiment is to perform user tracking for evaluating the accuracy rate for the detection of ‘9-Square Matrix” Aerobic Exercise. For this experiment, 8 subjects were asked to perform the 8 step sequences from the “9-Square Matrix” exercise. The steps have a number sequence that needs to be followed in exact order. The players are given the step to perform and the output is recorded as list of steps that was detected by the program.

To measure the accuracy of the detection from the program, there needs to be a classification of different types of errors that can occur while performing the “9-Square Matrix” exercise. The 3 types of errors recorded are categorized as follows:

1. Insertion Error(s): This type of error occurs when the user is midst of performing or progressing towards the next step in the sequence and the program does a false detection. (False Positive)
2. Deletion Error(s): This type of error occurs when the user steps on the matrix cell, but it does not get registered by the program. (False Negative)
3. Substitution Error(s): This type of error occurs when a different cell gets substituted instead of the matrix cell the user is stepping on.

The probability of each type of error for each individual step sequence is given by Equation 3.6:

$$P(Error) = \frac{No.of\ Errenous\ Steps\ Detected}{Total\ No.of\ Steps} \quad (3.6)$$

The results from the program output (Table 3.2) shows that the probability of the highest number of error recorded was the insertion error. This can be attributed to the fact that the implementation was using a normal webcam which lacked any depth sensing capabilities, thus detecting a false positive when the user's foot was not touching the floor plane.

Table 3.2 Probability Error for Detection of "9-Square Matrix"

| **Step** | **Probability of Error** | | | **Total Error (%)** |
|---|---|---|---|---|
| | **$P_I(E)$** | **$P_D(E)$** | **$P_S(E)$** | |
| 1 | 0.025 | 0.013 | 0.000 | 3.75 |
| 2 | 0.054 | 0.036 | 0.000 | 8.93 |
| 3 | 0.125 | 0.021 | 0.021 | 16.67 |
| 4 | 0.188 | 0.021 | 0.000 | 20.83 |
| 5 | 0.146 | 0.042 | 0.000 | 18.75 |
| 6 | 0.094 | 0.042 | 0.010 | 14.58 |
| 7 | 0.141 | 0.000 | 0.000 | 14.06 |
| 8 | 0.150 | 0.000 | 0.000 | 15.00 |

The occurrence of deletion error was overall low but had been observed when the user was performing a step too fast for the program to correctly register resulting in a false negative detection. The substitution error was very minimal and occurred due to the user's foot overlapping the neighboring cells rather than incorrect detection by the program. However, it is important to note that for most steps the probability of overall error remains below 20%.

### 3.3 Evaluation

The first experiment regarding the optimal camera placement shows that the re-projection error or RMSE used in the camera-calibration process is not a good indicator for the best setup value. This technique works best for calibration patterns that are mostly placed parallel to the camera, which does not necessarily diminish the observability of the calibration pattern. In addition, a good estimation should include a at least 10 snapshots of the pattern. Hence, the camera calibration performed with 50 views of the chessboard covering the entire field of view (FOV) of the camera in various poses returned a good re-projection error rate of 0.301135mm.

For the second experiment following the implementation of the “9-Square Matrix” aerobic exercise. It can be noted that the color-based segmentation for defining the ROI and tracking the ROI using image moments performed well due to the lower amount of error rate for majority of the steps. The highest error probability was for insertion errors since the camera used was a normal camera without any additional information on depth.

# Chapter 4

# VR-Based “9-Square Matrix” Aerobic Exercise for Preventing Physical and Cognitive Decline in Older Adults

This chapter focuses on the implementation of the “9-Square Matrix” Aerobic Exercise as an exergame using the Microsoft Kinect sensor. The first phase of the implementation presents a model for determining the optimal sensor setup based on the user’s height and matrix distance. Once the setup variables are determined, the optimal setup is obtained by performing an experiment to determine the depth accuracy on each of the setup variables. The second phase of implementation includes testing the exergame with 10 test subjects and classifying the results based on the error detection model described in Chapter 3.

## 4.1 Design for the VR-Based "9-Square Matrix" using Microsoft Kinect Sensor

For the implementation of the “9-Square Matrix” exercise into a VR-based exergame, it is crucial understand the process flow of the program (Figure 4.1). The program design relies heavily on the basic tasks of matrix detection and tracking the user’s foot location accurately on each of the matrix cells.

The matrix detection is performed by the find chessboard algorithm (described in Figure A.1) using the image processing library called EmguCV. Whereas the user tracking is done using the Kinect SDK, that relies on machine learning algorithms to detect the user’s skeleton. The first part of the implementation of the exergame involves determining the optimal sensor setup by testing for the depth accuracy at each given setup. The setup variables such as the camera height, angle and matrix distance are determined by a mathematical model with the minimum user height acting as a constraint. Once the setup variables have been calculated, the experiment for depth accuracy will be conducted on those parameters.

The second part of the implementation is based on user tracking. The 10 subjects are asked to perform each of the 8 step sequences and the program output is recorded. The errors are then classified into insertion, deletion and substitution error as defined in Section 3.2.2.

### 4.1.1 Determining the Setup Variables and Criteria for Sensor Placement

The Microsoft Kinect sensor provides depth sensing capabilities by including an IR projector and a depth camera along with a standard RGB camera. The Kinect sensor creates a depth map of the surrounding environment by projecting a laser speckle pattern which is visible to the depth camera. Depth is calculated by triangulation between the source (IR projector) and the depth camera. The skeleton data of the users is also generated from the depth map using machine learning algorithms.

The Microsoft human computer interface guidelines for Kinect [33], recommends the distance for the player and objection detection using depth mode to be between 1.2 meter to 3.5 meters. The depth resolution reduces drastically beyond this range. The maximum supported resolution of the depth camera is 640x480 pixels at 30 frames per second. The angular field of view (AFOV) for both the RGB and depth camera are the same, horizontal FOV is 57.5° and the vertical FOV is 43.5°

The Kinect sensor also contains a motor at the bottom to adjust the tilt angles (±27°) of the sensor manually. To design the experiment for the optimal sensor placement, there are 3 setup variables that need to be determined: (1) the elevation of the sensor, (2) the tilt angle of the sensor and (3) the ideal distance of the matrix to the sensor.

Additionally, the optimal setup for the sensor needs to accommodate both the view of the “9-Square Matrix” as well as the user. In this case, the constraints arise from the varying heights of different players. All users must lie within the sensors field of view for every setup that include various sensor poses (elevation and tilt) and matrix distance. If any part of the user is occluded, the skeleton tracking fails to detect the user.

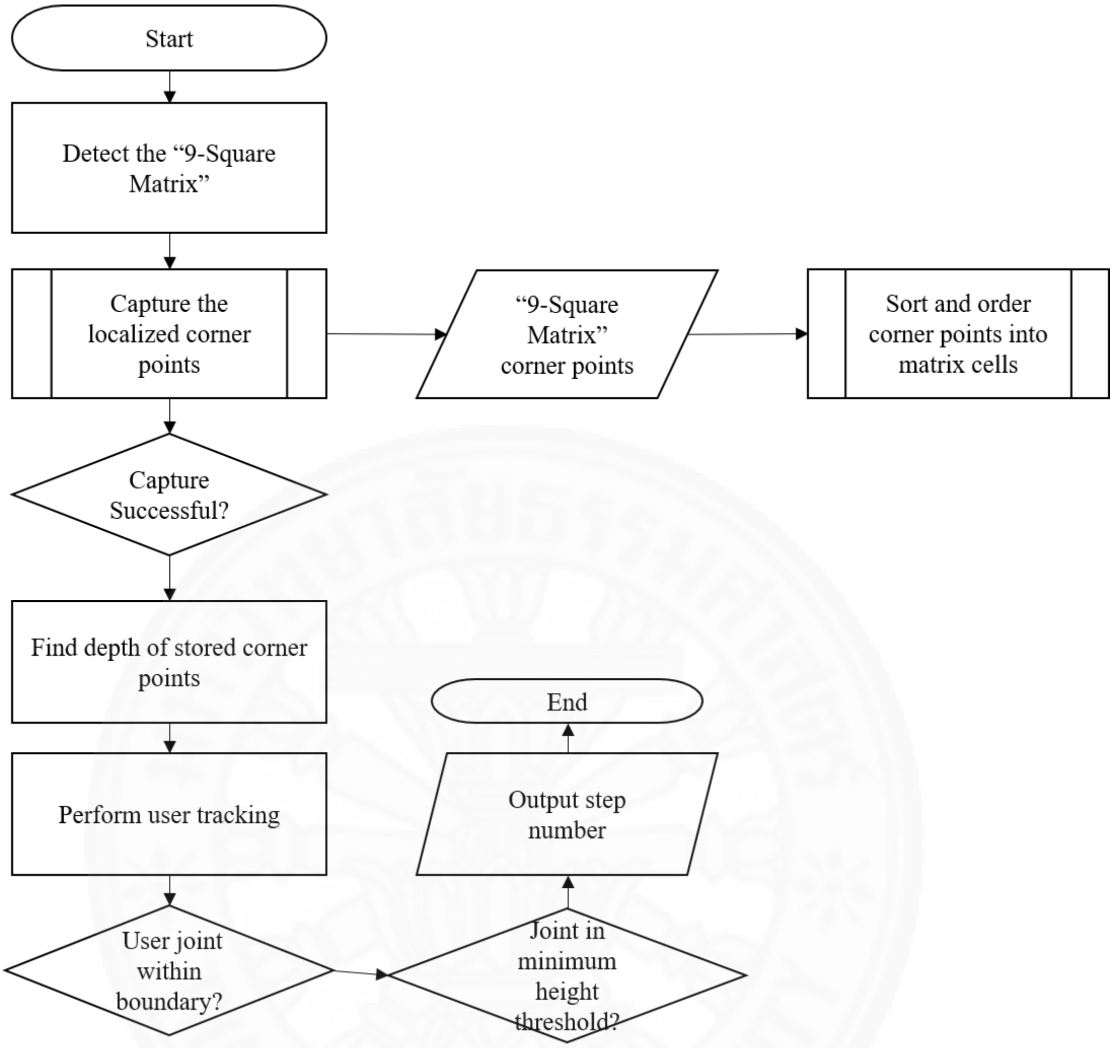


Figure 4.1 VR-Based "9-Square Matrix" Process Flow

### 4.1.2 Creating a Model to Determine the Sensor Pose (Tilt and Elevation) based on User Height and Matrix Distance

The objective is to find the optimal sensor elevation and tilt angle that can accommodate users within a minimum defined height. The tilt angle and distance affect the view pyramid; therefore, the user's height needs to be the constraint in formulating the problem. The sensors field of view is fixed but the tilt of the sensor towards the matrix lying on the floor plane can cause occlusion for users over a certain height. The setup scenario of the exergame is depicted in Figure 4.2.

The adjustment of the sensor's tilt allows varying observability of the matrix by increasing or decreasing the tilt angle (θ) towards the floor plane. The following equations can be formulated from Figure 4.2:

$$\tan(21.75 - \theta) = \frac{h_{user} - h_{camera}}{d} \tag{4.1}$$

$$\tan(21.75 + \theta) = \frac{h_{user} * (1 - 0.1591 * \tan^2(\theta))}{d} \tag{4.2}$$

Combining Equations 4.1 and 4.2, results in Equation 4.3:

$$0.7978 + 0.7978 * \tan^2(\theta) = \frac{h_{user} * (1 - 0.1591 * \tan^2(\theta))}{d} \tag{4.3}$$

By specifying the minimum user height ($h_{user}$) and distance ($d$) to the sensor in Equation 4.3, the tilt angle ($\theta$) is obtained. The tilt angle can then be substituted in Equation 4.1 to derive the sensor height ($h_{camera}$) for that tilt angle. This model will provide us the sensor elevation and tilt angle which will cover both the "9-Square Matrix" and user at the same time.

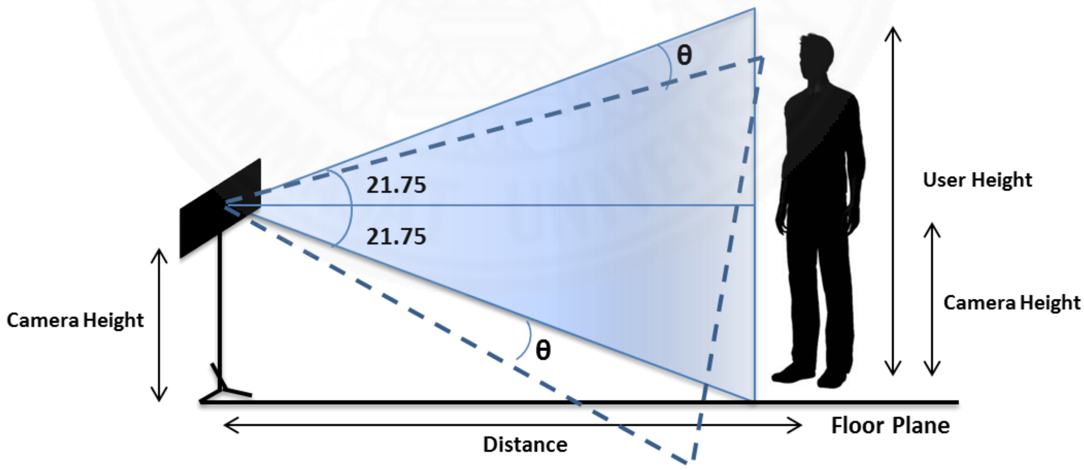

Figure 4.2 Sensor Setup Parameters for "9-Square Matrix"

### 4.1.3 Detecting the Pattern and User Tracking

The "9-Square Matrix" used in this exergame is identical to the chessboard calibration pattern used in the image processing library such as OpenCV. Similarly, for this implementation the matrix detection will be handled by using the find chessboard corners algorithm. After the extraction of all the individual corner points, the neighboring corners will be arranged into individual matrix cells.

For each of the detected corner points, the depth of the corner points will be obtained by mapping the pixel location from the RGB camera to the depth camera. This is only possible when the resolution of the RGB and depth camera is set to be the same resolution (i.e. 640 x 480 pixels) because the Kinect SDK does not allow mapping from the color space to the depth space.

Apart from providing depth values, the Kinect sensor also supports skeleton tracking. It relies on depth data to segment out the player pixels from the static objects in the scene. The skeleton tracking is enabled by sophisticated machine learning algorithms used by the Kinect sensor. The skeletal tracking algorithm can return data for up to 20 joint values. To track if the user is stepping on the matrix cell, the skeleton joint needs to be checked, if its overlapping with the matrix.

In addition, to the checking the if the user's foot lies within the boundaries of the matrix in relation to distance from the sensor, it is also crucial to detect the user's current posture. The Kinect sensor provides an estimation of the floor clip plane by 4 floating point values x-y-z and w, where w is the height of the sensor and the rest defines the orientation of the floor plane. The distance between the user's foot from the floor can be estimated using the point-plane distance formula as follows:

$$D = \frac{a_{x0} + b_{y0} + c_{z0} + d}{\sqrt{a^2 + b^2 + c^2}} \tag{4.5}$$

## 4.2 Experimentation and Results

### 4.2.1 Optimal Sensor Placement and Depth Accuracy Estimation

The goal is to determine the optimal sensor placement for maximum depth accuracy of the matrix and the user's joint data. The VR-based exergame implementation of the "9-Square Matrix" relies heavily on depth data for determining the distance of the matrix and skeleton tracking, therefore it is crucial to pick the most suitable sensor setup which results in minimal depth approximation error ($\eta$).

For this experiment, all the feasible sensor setup (elevation and tilt angle) values (Table 4.1) are calculated through the model described in Section 4.1.2. The sensor is mounted on an adjustable tripod and the tilt angles are modified programmatically. The program initially tries to detect the matrix lying on the floor, once detected the corner points are extracted and depth values are obtained by mapping the color space onto the depth space.

After the extraction of the depth values, the program calculates the distance between the adjacent corner points. The width of each matrix square ($\nu$) is exactly 24cm. By calculating the difference for each adjacent corner point, the average approximation error can be computed as follows:

$$\eta = \frac{\epsilon}{|\nu|} = \left|\frac{\nu - \nu_{approx}}{\nu}\right| = \left|1 - \frac{\nu_{approx}}{\nu}\right| \tag{4.6}$$

The average approximation error ($\eta$) is not the sole measure used for determining depth accuracy, the variance ($\sigma^2$) and standard deviation ($\sigma_x$) are also indicators of the error in depth data. The high variance and standard deviation values of the given population results in higher average approximation error.

Table 4.1 Depth Approximation Error for Localized Corner Points

| **Angle** | **Height (cm)** | **Distance** | **Variance ($\sigma^2$)** | **S.D ($\sigma_x$)** | **η (cm)** | **η (%)** |
|---|---|---|---|---|---|---|
| -26 | 189 | 170 | 3.804 | 1.950 | 1.408 | 5.868 |
| -23 | 180 | 180 | 5.669 | 2.381 | 2.058 | 8.576 |
| -20 | 170 | 190 | 4.866 | 2.206 | 1.775 | 7.396 |
| -16 | 156 | 200 | 5.986 | 2.439 | 2.400 | 10.000 |
| **Min** | | | 3.804 | 1.950 | 1.408 | 5.868 |
| **Max** | | | 5.986 | 2.439 | 2.400 | 10.000 |

The data from Table 4.1 illustrates that the best setup for a successful implementation of the “9-Square Matrix” exergame would be at a tilt angle of -26°, elevation height of 189cm and matrix distance of 170cm. The results indicate that the depth fidelity decreases with increasing distances, which is consistent with the Microsoft Kinect guidelines specification of depth range from 1.2 meters to 3.5 meters.

### 4.2.2 User Tracking and Game Mechanics

To test the success of the implementation of VR based “9-Square Matrix”, 10 subjects were asked to perform on all the 8 steps sequences from the exercise routine while the program output was recorded. The output was compared with the actual sequence and the error classification is performed using the same model defined in Chapter 3, Section 3.2.2. The result of the probability for each type of error is given in Table 4.2.

Table 4.2 Probability of Error Detection

| **Step** | **Probability of Detected Error(s)** | | | **Total Error (%)** |
|---|---|---|---|---|
| | **$P_I(E)$** | **$P_D(E)$** | **$P_S(E)$** | |
| 1 | 0 | 0.020 | 0 | 2 |
| 2 | 0.007 | 0.043 | 0 | 5 |
| 3 | 0.117 | 0.033 | 0 | 15 |
| 4 | 0.100 | 0.050 | 0 | 15 |
| 5 | 0.142 | 0.042 | 0 | 18 |
| 6 | 0.050 | 0.067 | 0.017 | 13 |
| 7 | 0.263 | 0.113 | 0.038 | 41 |
| 8 | 0.080 | 0.070 | 0 | 15 |

The occurrence of insertion error for the Kinect sensor occurs due to incorrect height threshold settings. The threshold is set for a minimum height of the joint from the floor plane. If the threshold is set too low then a false detection happens, whereas for higher thresholds the step might not get registered causing a false negative detection.

The deletion errors might also occur in the following cases: (1) Failure in skeleton tracking due to occlusion of the user's body parts and (2) Incorrect joint estimation by the sensor, which can be caused by the user's apparel or jitter in the data. The lowest probability of error detected was the substitution error. The substitution errors detected by the program is mostly caused by jitter in the data due to unreliable detection of the foot region.

It is important to note that for most of the steps the total error rate is well below 20% with an exception for step 7, which registers an error rate of 41%. The data indicates that while performing step 7, all types of errors were detected (insertion, deletion and substitution). This can be attributed to the fact that the step itself is highly complicated and creates an occlusion problem, which causes the skeleton tracking to fail temporarily.

### 4.2.3 Comparison of Implementation Techniques for "9-Square Matrix" Aerobic Exercise Recognition using Webcam and Microsoft Kinect Sensor

The "9-Square Matrix" Aerobic Exercise can be implemented using a simple webcam, relying solely on image processing tasks for detecting the matrix and perform user tracking. It is achieved by using the image processing algorithms provided by OpenCV library. The matrix detection is performed by the find chessboard corners algorithm, which is a robust algorithm that works even if the matrix is lying on the floor. It supports detection of the chessboard in various orientations and performs well even in poor lighting conditions.

Both the devices rely on the find chessboard corner algorithm for matrix detection using OpenCV library. The user tracking in case of the webcam is performed using color-based segmentation. This technique presents a constraint in terms of the accessory the user needs to wear (i.e. colored slippers) which might not be preferred by the users and the detection will fail in case there are objects of similar color placed in the webcams FOV. The color-based segmentation algorithm can also fail if there is too much noise introduced by different lighting conditions. The image moments function tracks the users foot with a good amount of precision, provided the detection of the slippers work correctly.

The user tracking for Microsoft Kinect sensor relies on the skeletal tracking engine to detect the various joints of the users. The skeleton tracking performs well for most of the joints estimated but can suffer while tracking the foot joint. The foot joint is the only joint estimation which suffers from jitter. Apart from the estimation problem, the occlusion of body parts can cause the skeleton tracking to stop working. This can be problematic due to the fact that the "9-Square Matrix" exercise routine consists of steps that can cause occlusion of body parts.

### 4.3 Evaluation

The first experiment is performed with the objective of finding the best sensor position and orientation for the detection of the "9-square matrix" and user tracking. According to the results of the experiment, the most optimal position and orientation of the sensor for this study is at the elevation level of 189cm and at an angle of -26° with the matrix at distance of 170 cm which results in **minimum approximation error** value of 5.86% given in Table 4.1.

The results in Table 4.2 demonstrate that 4 out of the 8 patterns have been performed with minimal insertion, deletion and substitution errors whereas the rest of the patterns have shown a larger amount of error. Therefore, the success of the implementation of VR based “9-square matrix” exergame would rely on more sophisticated machine learning algorithms that would be able to estimate the joint data in the case of occlusion or a combination of skeleton data and simple image processing algorithms such as color-based thresholding to achieve as high of an accuracy as possible in joint location estimation. An additional Microsoft Kinect sensor can also provide a better view of the user to eliminate the occlusion and improve the accuracy of detection drastically.

Figure 4.3 “9-Square Matrix” Exergame Setup

# Chapter 5
# "9-Square Matrix" Exergame

This chapter provides details on the design of the interface and game mechanics of the exergame for the "9-Square Matrix". The evaluation of the gameplay for the "9-Square Matrix" using Kinect sensor is given in Chapter 4.

## 5.1 Game Design

The "9-Square Matrix" exergame is designed for older adults to be able to perform aerobic exercise daily at their home without minimal external assistance. Hence the requirements of the system have been narrowed down to the following criteria's:

- A low cost and low maintenance system which can be used with minimal effort
- Developing the interface for elderly users
- The exercise routine for "9-Square Matrix" should be appropriate for elderly population
- Addition of new exercise routines can be easily done by the coach or physiotherapist
- In game assistance for the elderly by providing audio and visual feedback to the user

After considering the traditional "9-Square Matrix" exercise routine, it has been determined that the most suitable device for the implementation for the exergame would be Microsoft Kinect sensor due to its ability to perform full body tracking. The Kinect allows natural user interactions which are easier to learn compared to controller-based sensors and devices.

The system allows for selection of different levels of difficulty which can be adapted according to the ability of the users. The exergame can be further adapted and customized by the physiotherapist by simply adding new step sequence into a text file. The key design elements from previous studies [7, 34, 35] are incorporated for the design of the e" 9-Square Matrix" exergame.

## 5.2 System Architecture

### 5.2.1 User Interface and Design Flow

The initial prototype for the exergame was built using a normal Logitech webcam was implement in C++ programming language and OpenCV image processing library. The interface was simple and only provided output for the user's step on the matrix.

For the actual exergame implementation using the Kinect sensor, the system was designed to accommodate the design recommendations for older adults. The interface has minimal amount of text in large and clear fonts, there are buttons placed for navigations between each menu items. The gesture control for navigation is limited due to scaling issues which can be very hard and frustrating to use for the elderly. The only gesture used for navigation is to the end the game session using the wave gesture, since the user will be significantly away from the computer device.

There are audio and visual in-game assistance to guide the user, in case the users have forgotten a step. At the end of the game session, the user can see their performance in terms of the time of each session and score. For the different difficulty levels, a step will be either added or omitted.

The "9-Square Matrix" exergame user interface has been implemented using C# programming language and EmguCV image processing library. The user interface and overall system design is presented in Figure 5.1.

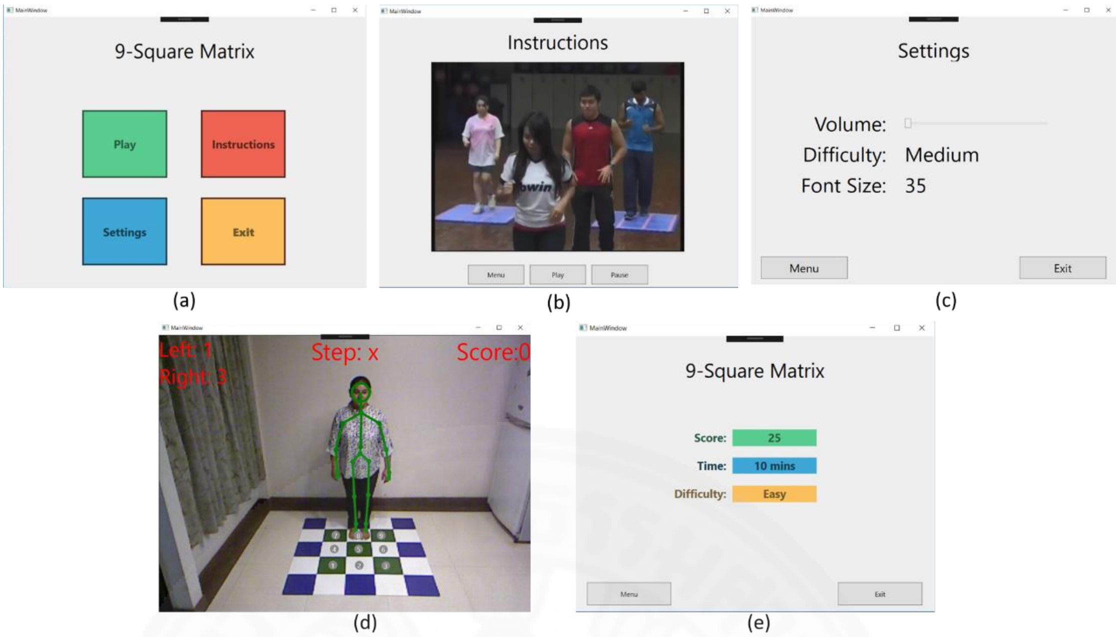


Figure 5.1 The User Interface for “9-Square Matrix” Exergame: (a) Main Menu, (b) Instruction Screen, (c) Settings Menu, (d) Scores Page

### 5.2.2 Game Mechanics

The game session starts with matrix detection and once the matrix has been captured successfully, the numbers on each respective matrix is displayed (Figure 5.2). The system presents a pattern to the user, along with the next steps for the left and right foot they need to perform, much like the gameplay for DDR (Figure 5.3). The step pattern is displayed visually, and the audio is presented to the user.

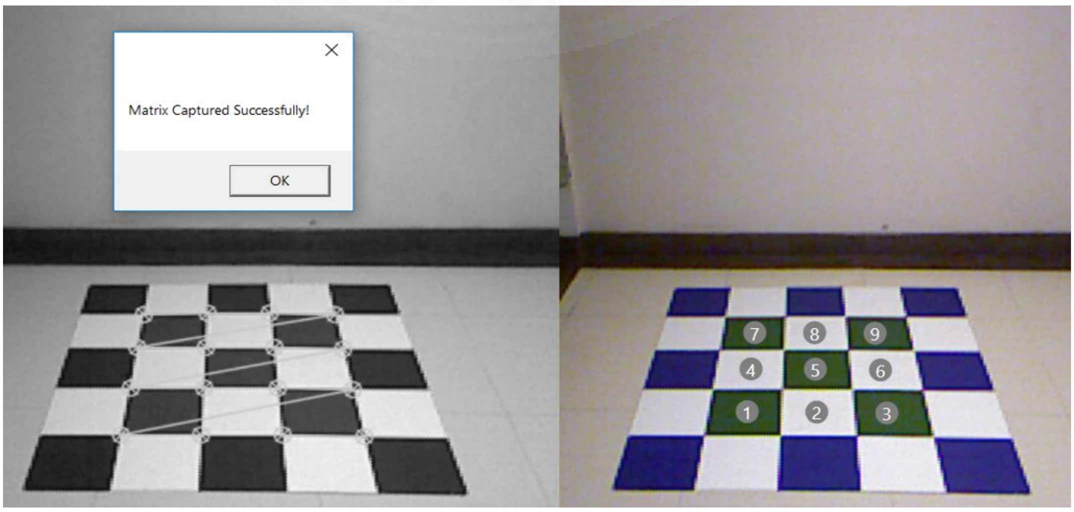


Figure 5.2 Matrix Detection and Capture for “9-Square Matrix” Exergame

In the background, there is a timer set to 35 seconds, during which the user needs to repeat the step until the times up. The user scores an additional point for each step performed correctly including the repetitions. The different levels of difficulty presented to the users include easy, medium and hard. The easy level only presents the users from step 1 to step 4, the medium level includes step 5 and 6 and the hard level includes step 7 and 8.  The steps are presented in a random order for the users to perform.

After the end of every session, the user is presented with a summary of their performance. The summary page shows the score and time spent playing the game. The goal of this exergame is not for the users to maximize their scores, but to spend a minimum required time on each play session.

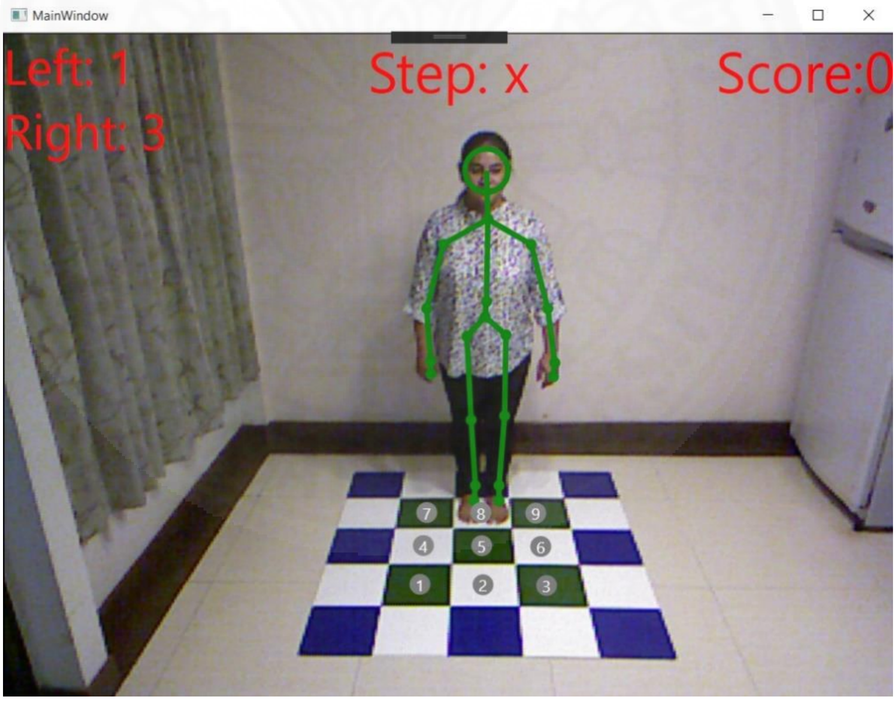

Figure 5.3 Game Play User Interface

### 5.2.3 The Results of Implementing "9-Square Matrix" Aerobic Exergame using Kinect

To understand the challenges of implementing the "9-Square Matrix" Aerobic exercise as an exergame, it is important to analyze the results from the implementation of Kinect (Table 5.1). The study participants based on the Kinect implementation of the exergame involved 10 healthy young adults (5 females, 5 males) with a mean age of 25 years ± 3 years.

Table 5.1 Probability of Insertion, Deletion and Substitution Error(s) for Kinect with 10 test subjects

| **Step** | **Error Sequence** | | | | | | **Total Error (%)** |
|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | |
| 1 | 0 | 0.000 | 2 | 0.020 | 0 | 0.000 | 2 |
| 2 | 1 | 0.007 | 6 | 0.043 | 0 | 0.000 | 5 |
| 3 | 7 | 0.117 | 2 | 0.033 | 0 | 0.000 | 15 |
| 4 | 6 | 0.100 | 3 | 0.050 | 0 | 0.000 | 15 |
| 5 | 17 | 0.142 | 5 | 0.042 | 0 | 0.000 | 18 |
| 6 | 6 | 0.050 | 8 | 0.067 | 2 | 0.017 | 13 |
| 7 | 21 | 0.263 | 9 | 0.113 | 3 | 0.038 | 41 |
| 8 | 8 | 0.080 | 7 | 0.070 | 0 | 0.000 | 15 |

The highest type of error recorded was the insertion error, followed by deletion and substitution error. The deletion error for Kinect sensor mostly resulted from occlusion and incorrect threshold settings. Lastly, substitution errors occurred from either the users stepping on the boundaries of matrix cells or the jitter in skeleton data for the foot joint.

As for the performance of the exercise routine itself, step 7 was deemed the most difficult step resulting in fall for 2 male participants. Apart from that, the subjects experienced fun while performing the dance routines (Figure 5.4).

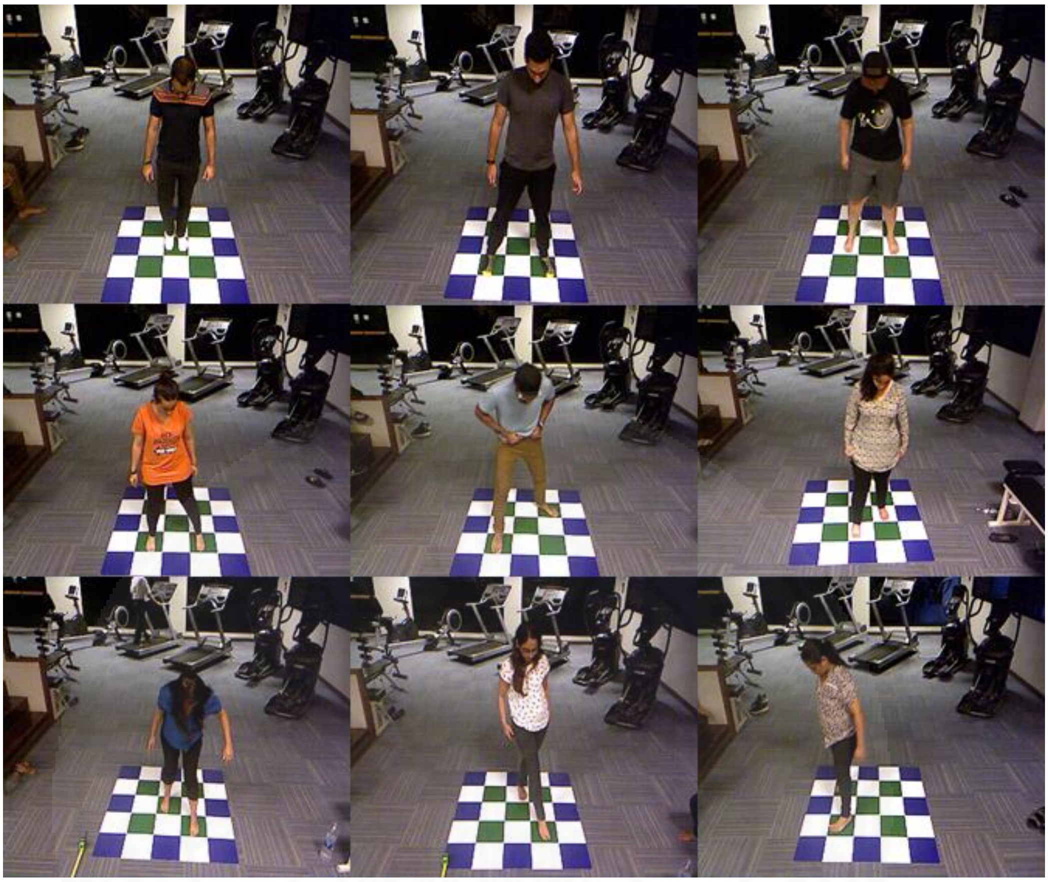

Figure 5.4 “9-Square Matrix” Exergame Implementation Experiment

# Chapter 6
# Conclusion and Future Work

This research focuses on the implementation of the "9-Square Matrix" Aerobic Exercise in the form of an exergame for older adults. The first phase of the implementation lies in recognizing the techniques for using image processing in the detection of the matrix. The matrix detection for both cases of implementation using webcam and Kinect sensor relies on image processing algorithm to successfully capture the matrix.

In both the cases, the optimal setup and camera placement has been determined. For the webcam placement problem, the re-projection error was a suggested as a measure of accuracy for the matrix detection. However, the re-projection error was not suitable for this purpose, due to incorrect results. Instead a setup was chosen to maximize the observability instead.

The optimal setup for Microsoft Kinect sensor using matrix detection required a model to determine the setup values, so that it would include the user's entire body in the FOV of the sensor to avoid failure in skeleton tracking engine. The accuracy was measured by calculating the length between all adjacent corner points using depth. The best-case scenario returned an average depth error rate of 5.9%.

The second phase of implementation for both the webcam and Kinect sensor included user tracking and recognition of the steps performed by the system. In this scenario, the Kinect sensor performed better than the webcam overall, but both the systems had shown an error rate below 20% for all the steps except one step.

Both the systems are implemented using completely different techniques and can be augmented together to further improve the accuracy of detection. The Kinect sensor can stop working in the case of occlusion for the body parts of the users, where the webcam does not suffer from this problem and the webcam is a cheap and effective solution to address the short-comings of the Kinect.

## References


1. Nations, U., *World Population Ageing Report*. 2017.

2. *Global Health and Aging*, U.S.D.o.H.a.H. Services, Editor. 2011, World Health Organization (WHO). p. 1-26.

3. Taylor, D., *Physical activity is medicine for older adults.* Postgraduate Medical Journal, 2014. 90(1059): p. 26-32.

4. Charoen, K., *Tarang 9 Chong and Developement of Cognitive Abilities in Older Adults.* Thai Journal of Nursing Council, 2013. 28(4): p. 68-80.

5. Charoen, K. and S. Salee, *Improving autistic students behavior through the matrix of nine squares.* The Journal of Physical Education and Sports, 2007. 15(1): p. 25-38.

6. Florian, M., et al. *Designing sports: a framework for exertion games*. in *the Special Interest Group Computer-Human Interface Conference on Human Factors in Computing Systems*. 2011. Vancouver, British Columbia.

7. Brox, E., S.T. Konstantinidis, and G. Evertsen, *User-Centered Design of Serious Games for Older Adults Following 3 Years of Experience With Exergames for Seniors: A Study Design.* JMIR Serious Games, 2017. 5(1): p. e2.

8. Wiemeyer, J. and A. Kliem, *Serious games in prevention and rehabilitation—a new panacea for elderly people?* European Review of Aging and Physical Activity, 2012. 9(1): p. 41-50.

9. Agmon, M., et al., *A pilot study of wii fit exergames to improve balance in older adults.* Journal of Geriatric Physical Therapy, 2011. 34(4): p. 161-167.

10. Bieryla, K.A. and N.M. Dold, *Feasibility of wii fit training to improve clinical measures of balance in older adults.* Clinical Interventions in Aging, 2013. 8: p. 775-781.

11. Franco, J.R., et al., *The effect of the Nintendo Wii Fit and exercise in improving balance and quality of life in community dwelling elders.* Technol. Health Care, 2012. 20(2): p. 95-115.

12. Harley, D., et al. *Making the Wii at Home: Game Play by Older People in Sheltered Housing*. 2010. Berlin, Heidelberg: Springer Berlin Heidelberg.

13. Laver, K., et al., *Use of an interactive video gaming program compared with conventional physiotherapy for hospitalised older adults: a feasibility trial.* Disability and Rehabilitation, 2012. 34(21): p. 1802-1808.

14. Toulotte, C., C. Toursel, and N. Olivier, *Wii Fit® training vs. Adapted Physical Activities: which one is the most appropriate to improve the balance of independent senior subjects? A randomized controlled study.* Clinical Rehabilitation, 2012. 26(9): p. 827-835.

15. Young, W., et al., *Assessing and training standing balance in older adults: A novel approach using the 'Nintendo Wii' Balance Board.* Gait & Posture, 2011. 33(2): p. 303-305.

16. Bohm, H., M. Hartmann, and B. Bohm, *Predictors of metabolic energy expenditure from body acceleration and mechanical energies in new generation active computer games.* Computer Science in Sport - Mission and Methods, 2008. 8372.

17. Gerling, K., et al., *Full-body motion-based game interaction for older adults*, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2012, ACM: Austin, Texas, USA. p. 1873-1882.

18. Hachaj, T. and M. Ogiela, *Rule-based approach to recognizing human body poses and gestures in real time.* Multimedia Systems, 2014. 20(1): p. 81-99.

19. Maloney, A.E., et al., *A Pilot of a Video Game (DDR) to Promote Physical Activity and Decrease Sedentary Screen Time.* Obesity, 2008. 16(9): p. 2074-2080.

20. Wollersheim, D., et al., *Physical and psychosocial effects of wii video game use among older women.* International Journal of Emerging Technologies and Society, 2010. 8(2): p. 85-98.

21. King, I.I.T.I., *Hand Strengthening With a Computer for Purposeful Activity.* American Journal of Occupational Therapy, 1993. 47(7): p. 635-637.

22. Sohnsmeyer, J., H. Gilbrich, and B. Weisser, *Effect of a six-week intervention with an activity-promoting video game on isometric muscle strength in elderly subjects.* International Journal of Computer Science in Sport, 2010. 9(2): p. 75-79.

23. Kliem, A. and J. Wiemeyer, *Comparison of a traditional and a video game based balance training program.* International Journal of Computer Science in Sport, 2010. 9(2): p. 78-89.

24. Williams, M.A., et al. *Exercising with computers in later life—pilot and feasibility study of the acceptability of the nintendo® wiifit in community-dwelling fallers*. 2010. 3.

25. Fraile, S.H., et al., *Suitability analysis of commercial open-source driven motion sensor devices applied to exergames for the elderly*, in *Ambient Assisted Living Forum Eindhoven*. 2012: Eindhoven.

26. Contributers, W. *Wii Remote*. 2018 [cited 2018 10 June].

27. Jorgensen, M.G., et al., *Efficacy of nintendo wii training on mechanical leg muscle function and postural balance in community-dwelling older adults: a randomized controlled trial.* The Journals of Gerontology Series A: Biological Sciences and Medical Sciences, 2013. 68(7): p. 845-852.

28. Maillot, P., A. Perrot, and A. Hartley, *Effects of interactive physical-activity video-game training on physical and cognitive function in older adults.* Psychology and Aging, 2012. 27(3): p. 589-600.

29. Pluchino, A., et al., *Pilot study comparing changes in postural control after training using a video game balance board program and 2 standard activity-based balance intervention programs.* Archives of Physical Medicine and Rehabilitation, 2012. 93(7): p. 1138-1146.

30. Rendon, A.A., et al., *The effect of virtual reality gaming on dynamic balance in older adults.* Age and Ageing, 2012. 41(4): p. 549-552.

31. Schoene, D., et al., *A Randomized Controlled Pilot Study of Home-Based Step Training in Older People Using Videogame Technology.* Public Library of Science ONE, 2013. 8(3).

32. Bodor, R., et al. *Mobile camera positioning to optimize the observability of human activity recognition tasks*. in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2005.

33. *Human Interface Guidelines v1.7*. 2013.

34. Kathrin, G., et al. *Full-body motion-based game interaction for older adults*. in *the Special Interest Group Computer-Human Interface Conference on Human Factors in Computing Systems*. 2012. Austin, Texas.

35. Ofli, F., et al., *Design and evaluation of an interactive exercise coaching system for older adults: lessons learned.* IEEE Journal of Biomedical and Health Informatics, 2015(99): p. 1-15.

# Appendices

# Appendix A
# "9-Square Matrix" Aerobic Exercise Design

The implementation of the "9-Square Matrix" relies on important image processing algorithms which are discussed in this section.

## A.1 Assessment Methodologies for Physical Functioning in Older Adults

There are a lot of existing methodologies for assessing the overall balance ability, motor skills and physical functioning in elderlies. Most of the trials and studies based on exergames use these assessments to test the improvement pre and post intervention. This section presents a summary of most widely used assessment such as Timed-Up and Go (TUG), Berg Balance Scale (BBS), Falls Efficacy Scale (FES) and Functional Reach Test (FRT).

Table A.1 Assessment Tests to Measure the Physical Functioning in Older Adults

| Methodology | Measurable Outcomes |
|---|---|
| **Berg Balance Scale (BBS)** | |
| • The BBS contains 14 sets of tasks to be performed by older adults to measure their dynamic and static balance ability.<br>• The examiner must rate each task on a scale from 0 – 4. (lowest to highest)<br>• To see a real difference in the balance ability of the elderly, a minimum 8-point difference needs to be achieved compared to the results before the intervention.<br>• The time required to perform the BBS test is 15-20 mins and requires a ruler, chairs (with and without arm rest), footstep, wrist watch and walking space. | • Balance ability and gait assessment in elderly subjects<br>• An 8-point score increment compared to previous results is required to measure an impact in the balance ability of the individual |
| **Timed Up and Go (TUG)** | |
| • The TUG test is designed to assess the mobility and balance for elderly people<br>• The TUG test consists of sitting on the chair with back leaning straight against the chair and arms on the arm rest, the patient is asked to walk to a marker placed 3 meters (9.8ft) away and then turn around and walk back and sit on the chair<br>• A stopwatch is used to time, how long it took the person to get up, walk and sit back in the chair starting from the instruction go. | • Tests the mobility of elderly people by measuring the time (seconds) to perform the test<br>• The higher the time to complete the test indicates higher risk of falls and balance ability<br>• Cut-off rate greater than 13.5s shows the ability to predict falls in community dwelling elderly, however there is no standardized cut-off score |

<table>
<tr><th colspan="2">Tinetti-POMA (Performance Oriented Mobility Assessment)</th></tr>
<tr><td>• Tinetti-POMA is a test used to evaluate the mobility and gait of elderly individuals<br>• The test is divided into 2 parts; Balance assessment and Gait assessment each consisting of different tasks which overall require 10-15 mints to complete<br>• The scores allotted vary from 0 – 2, 0 = most impaired and 2 = independent. The max score for balance assessment is 16, gait assessment is 12 making the overall max to 28 (balance + gait)<br>• The equipment needed to carry out the Tinetti-POMA tests include hard armless chair, stopwatch and 15ft walking space</td><td>• The evaluation is done based on the sum of the balance and gait score<br>• 25-18 = low fall risk, 19-24 = medium fall risk and < 19 = high fall risk<br>• The tasks consist of activities such as standing, sitting, walking etc., which helps predicts the risks of fall while performing day-to-day activities</td></tr>
<tr><th colspan="2">Morse Fall Scale (MFS)</th></tr>
<tr><td>• The Morse fall scale was created to assess the probability of a fall occurring<br>• Morse fall risk assessment is usually conducted when the patient is admitted to the hospital, changes wards, following a fall or monthly/weekly basis<br>• The Morse fall scale ratings are based on the following factors<br>  • History of previous falls<br>  • Whether the patient has any other medical conditions or secondary diagnosis<br>  • Ambulatory aid, meaning whether the patient requires nurse to aid them continuously (bed rest) or they need crutches and walker to move around or can walk by clutching on the furniture but doesn’t ask for help<br>  • Is an IV being administered to the patient or any other equipment is attached<br>  • Patient’s gait assessment is normal, weak or impaired<br>  • Patient’s mental status is normal or overestimate their abilities and are forgetful</td><td>• The MFS provides an overview over the probability of falls of a patient depending upon their scores in each section.<br>• The maximum score is 125 and the lowest is 0.<br>• While the lowest score indicates no risk for falls, anything below 25 is also considered a low risk patient whereas 24-45 present moderate risk and anything beyond 45 should be considered a high-risk patient.</td></tr>
<tr><th colspan="2">Falls Efficacy Scale – FES / FES-I (International)</th></tr>
<tr><td>• This is a questionnaire based on an individual’s self-assessment on risk of falling based on day-to-day activities (14 activities).<br>• The score is ranges from 1 (not at all concerned) – 4 (very concerned) and takes 5 mints to complete.<br>• The higher score reflects a lack of confidence in performing the task independently without falling.</td><td>• This test is basically a questionnaire for evaluating an individual’s efficacy in performing a task without falling or presenting a risk of falls</td></tr>
<tr><th colspan="2">Functional Reach Test (FRT)</th></tr>
<tr><td>• The functional reach test aims to measure the stability and balance in individuals<br>• A measuring tape is attached to a parallel wall and a subject is asked to reach as further as they can by keeping their body in the same position and keeping their arms at 90 degrees and stretching it forward without moving<br>• The distance is measured, and the test is performed 3 times overall to obtain an average value</td><td>• Balance and risk of fall detection based on reach values compared to the baseline values</td></tr>
</table>

## A.2 Matrix Detection using Find Chessboard Algorithm in OpenCV

The find chessboard algorithm is used for both the webcam and Kinect based implementation of the exergame. It uses many iterative approaches for the chessboard detection, which makes it the most robust and efficient algorithm for the matrix detection as well. The algorithm is described in Figure A.1.

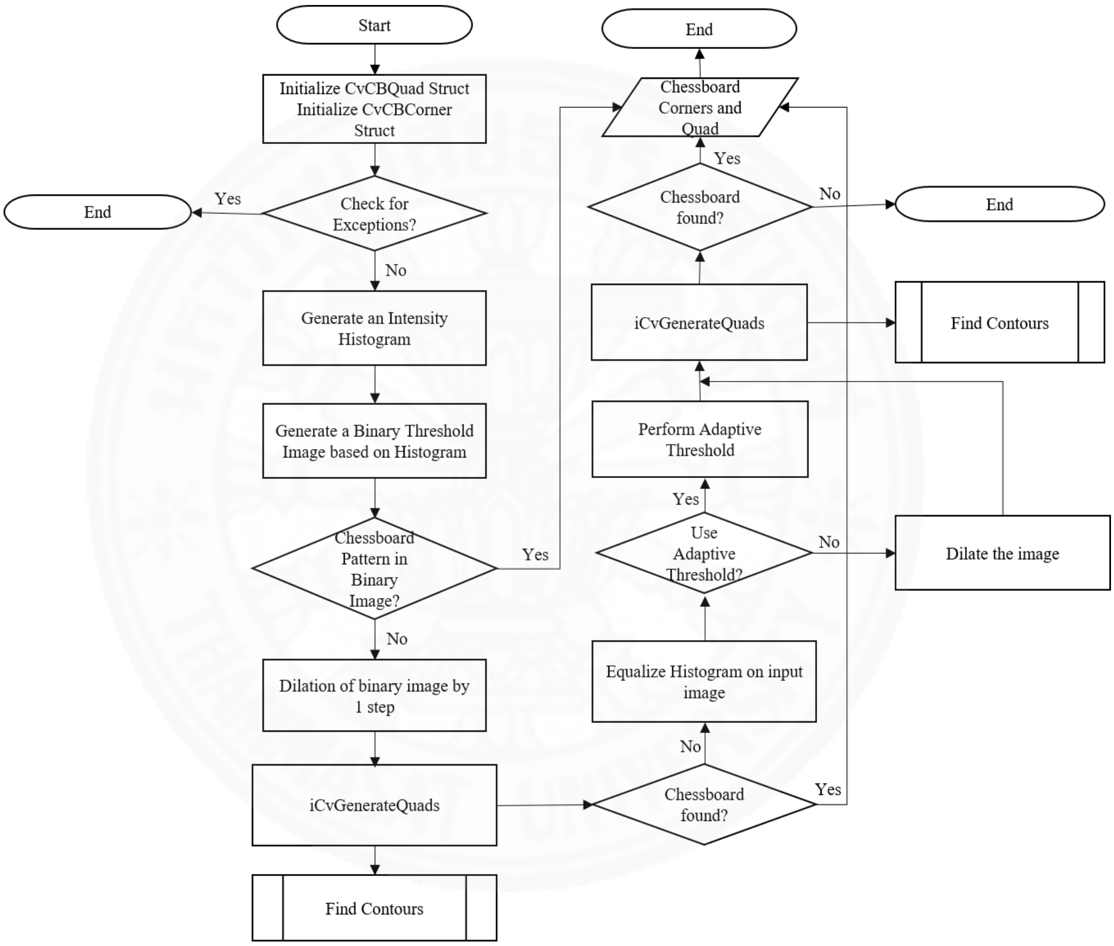


Figure A.1 Find Chessboard Algorithm

# Appendix B
# "9-Square Matrix" Exergame Experimental Results

This section presents the experimental results from Chapter 3 based on the optimal camera setup and Chapter 4 for the depth-based Kinect sensor setup along with user tracking results for both.

## B.1 User Tracking Data for "9-Square Matrix" using Webcam with 8 Test Subjects

Table B.1 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 1

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 1 | 0.100 | 0 | 0.000 | 0 | 0.000 | 1 | 0.100 |
| 2 | 0 | 0.000 | 1 | 0.071 | 0 | 0.000 | 1 | 0.071 |
| 3 | 0 | 0.000 | 1 | 0.167 | 1 | 0.167 | 2 | 0.333 |
| 4 | 1 | 0.167 | 1 | 0.167 | 0 | 0.000 | 2 | 0.333 |
| 5 | 1 | 0.083 | 2 | 0.167 | 0 | 0.000 | 3 | 0.250 |
| 6 | 0 | 0.000 | 3 | 0.250 | 1 | 0.083 | 4 | 0.333 |
| 7 | 1 | 0.125 | 0 | 0.000 | 0 | 0.000 | 1 | 0.125 |
| 8 | 2 | 0.200 | 0 | 0.000 | 0 | 0.000 | 2 | 0.200 |

Table B.2 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 2

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0.000 | 1 | 0.100 | 0 | 0 | 1 | 0.100 |
| 2 | 1 | 0.071 | 0 | 0.000 | 0 | 0 | 1 | 0.071 |
| 3 | 0 | 0.000 | 0 | 0.000 | 0 | 0 | 0 | 0.000 |
| 4 | 2 | 0.333 | 0 | 0.000 | 0 | 0 | 2 | 0.333 |
| 5 | 3 | 0.250 | 1 | 0.083 | 0 | 0 | 4 | 0.333 |
| 6 | 1 | 0.083 | 0 | 0.000 | 0 | 0 | 1 | 0.083 |
| 7 | 0 | 0.000 | 0 | 0.000 | 0 | 0 | 0 | 0.000 |
| 8 | 2 | 0.200 | 0 | 0.000 | 0 | 0 | 2 | 0.200 |

Table B.3 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 3

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 2 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 3 | 1 | 0.167 | 0 | 0 | 0 | 0 | 1 | 0.167 |
| 4 | 2 | 0.333 | 0 | 0 | 0 | 0 | 2 | 0.333 |
| 5 | 2 | 0.167 | 0 | 0 | 0 | 0 | 2 | 0.167 |
| 6 | 2 | 0.167 | 0 | 0 | 0 | 0 | 2 | 0.167 |
| 7 | 3 | 0.375 | 0 | 0 | 0 | 0 | 3 | 0.375 |
| 8 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |

Table B.4 “9-Square Matrix” Aerobic Exercise Recognition Data for Subject 4

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 2 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 3 | 1 | 0.167 | 0 | 0 | 0 | 0 | 1 | 0.167 |
| 4 | 2 | 0.333 | 0 | 0 | 0 | 0 | 2 | 0.333 |
| 5 | 2 | 0.167 | 0 | 0 | 0 | 0 | 2 | 0.167 |
| 6 | 2 | 0.167 | 0 | 0 | 0 | 0 | 2 | 0.167 |
| 7 | 3 | 0.375 | 0 | 0 | 0 | 0 | 3 | 0.375 |
| 8 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |

Table B.5 “9-Square Matrix” Aerobic Exercise Recognition Data for Subject 5

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 1 | 0.100 | 0 | 0 | 0 | 0 | 1 | 0.100 |
| 2 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 3 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 4 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 5 | 2 | 0.167 | 0 | 0 | 0 | 0 | 2 | 0.167 |
| 6 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 7 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 8 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |

Table B.6 “9-Square Matrix” Aerobic Exercise Recognition Data for Subject 6

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 2 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 3 | 1 | 0.167 | 0 | 0 | 0 | 0 | 1 | 0.167 |
| 4 | 2 | 0.333 | 0 | 0 | 0 | 0 | 2 | 0.333 |
| 5 | 2 | 0.167 | 0 | 0 | 0 | 0 | 2 | 0.167 |
| 6 | 3 | 0.250 | 0 | 0 | 0 | 0 | 3 | 0.250 |
| 7 | 1 | 0.125 | 0 | 0 | 0 | 0 | 1 | 0.125 |
| 8 | 1 | 0.100 | 0 | 0 | 0 | 0 | 1 | 0.100 |

Table B.7 “9-Square Matrix” Aerobic Exercise Recognition Data for Subject 7

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 2 | 0 | 0.000 | 0 | 0 | 0 | 0 | 0 | 0.000 |
| 3 | 1 | 0.167 | 0 | 0 | 0 | 0 | 1 | 0.167 |
| 4 | 1 | 0.167 | 0 | 0 | 0 | 0 | 1 | 0.167 |
| 5 | 1 | 0.083 | 0 | 0 | 0 | 0 | 1 | 0.083 |
| 6 | 1 | 0.083 | 0 | 0 | 0 | 0 | 1 | 0.083 |
| 7 | 1 | 0.125 | 0 | 0 | 0 | 0 | 1 | 0.125 |
| 8 | 2 | 0.200 | 0 | 0 | 0 | 0 | 2 | 0.200 |

Table B.8 "9-Square Matrix" Aerobic Exercise Recognition Data for Subject 8

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0.000 | 0 | 0.000 | 0 | 0 | 0 | 0.000 |
| 2 | 0 | 0.000 | 1 | 0.071 | 0 | 0 | 1 | 0.071 |
| 3 | 2 | 0.333 | 0 | 0.000 | 0 | 0 | 2 | 0.333 |
| 4 | 1 | 0.167 | 0 | 0.000 | 0 | 0 | 1 | 0.167 |
| 5 | 1 | 0.083 | 1 | 0.083 | 0 | 0 | 2 | 0.167 |
| 6 | 1 | 0.083 | 1 | 0.083 | 0 | 0 | 2 | 0.167 |
| 7 | 2 | 0.250 | 0 | 0.000 | 0 | 0 | 2 | 0.250 |
| 8 | 2 | 0.200 | 0 | 0.000 | 0 | 0 | 2 | 0.200 |

## B.2 Determining the Depth Accuracy for Each Setup Value

Table B.9 Depth Data for Tilt Angle: -26°, Elevation: 189cm and Distance: 170cm

| | Cell 1 | Cell 2 | Cell 3 | Cell 4 | Cell 5 | Cell 6 | Cell 7 | Cell 8 | Cell 9 |
|---|---|---|---|---|---|---|---|---|---|
| **Depth 1** | 23.8 | 24.2 | 24.2 | 28.6 | 26.2 | 23.4 | 24.7 | 21.4 | 24.2 |
| **Depth 2** | 24.2 | 24.2 | 26.6 | 26.2 | 23.4 | 23.8 | 21.4 | 24.2 | 21.4 |
| **Depth 3** | 2.4 | 0 | 2.4 | 0 | 2.8 | 2.8 | 3.3 | 0 | 0 |
| **Depth 4** | 2 | 0 | 0 | 2.4 | 0 | 2.4 | 0 | 2.8 | 2.8 |
| | **Mean** | 24.375 | | **Variance** | 3.804 | | **S.D.** | 1.950 | |

Table B.10 Depth Data for Tilt Angle: -23°, Elevation: 180cm and Distance: 180cm

| | Cell 1 | Cell 2 | Cell 3 | Cell 4 | Cell 5 | Cell 6 | Cell 7 | Cell 8 | Cell 9 |
|---|---|---|---|---|---|---|---|---|---|
| **Depth 1** | 21.8 | 22.5 | 26.6 | 25.8 | 23.8 | 21 | 24.2 | 28 | 27.5 |
| **Depth 2** | 22.5 | 26.6 | 22.5 | 23.8 | 21 | 23.8 | 28 | 27.5 | 28 |
| **Depth 3** | 4.8 | 0 | 0 | 2.8 | 2.8 | 2.8 | 6.6 | 3.3 | 3.3 |
| **Depth 4** | 4.1 | 4.1 | 4.1 | 4.8 | 0 | 0 | 2.8 | 2.8 | 2.8 |
| | **Mean** | 24.625 | | **Variance** | 5.669 | | **S.D.** | 2.381 | |

Table B.11 Depth Data for Tilt Angle: -20°, Elevation: 170cm and Distance: 190cm

| | Cell 1 | Cell 2 | Cell 3 | Cell 4 | Cell 5 | Cell 6 | Cell 7 | Cell 8 | Cell 9 |
|---|---|---|---|---|---|---|---|---|---|
| **Depth 1** | 22.5 | 25 | 20.4 | 23.8 | 24.2 | 26.7 | 28 | 25.1 | 28.6 |
| **Depth 2** | 25 | 20.4 | 22.9 | 24.2 | 26.7 | 24.2 | 25.1 | 28.6 | 25.1 |
| **Depth 3** | 0 | 2.1 | 0 | 2.5 | 2.5 | 2.5 | 2.9 | 0 | 0 |
| **Depth 4** | 2.5 | 2.5 | 2.5 | 2.9 | 0 | 0 | 0 | 3.5 | 3.5 |
| | **Mean** | 24.708 | | **Variance** | 4.866 | | **S.D.** | 2.206 | |

Table B.12 Depth Data for Tilt Angle: -16°, Elevation: 156cm and Distance: 200cm

| | Cell 1 | Cell 2 | Cell 3 | Cell 4 | Cell 5 | Cell 6 | Cell 7 | Cell 8 | Cell 9 |
|---|---|---|---|---|---|---|---|---|---|
| **Depth 1** | 26.6 | 26.6 | 29.1 | 21 | 23.8 | 21.3 | 27.5 | 24.7 | 28 |
| **Depth 2** | 26.6 | 29.1 | 27.1 | 23.8 | 21.3 | 24.2 | 24.7 | 28 | 25.1 |
| **Depth 3** | 0 | 0 | 2 | 0 | 2.5 | 0 | 2.8 | 0 | 2.9 |
| **Depth 4** | 0 | 2.5 | 0 | 2.8 | 0 | 2.9 | 0 | 3.3 | 0 |
| | **Mean** | 25.417 | | **Variance** | 5.948 | | **S.D.** | 2.439 | |

## B.3 User Tracking with "9-Square Matrix" Exergame using Microsoft Kinect Sensor for 10 Test Subjects

Table B.13 "9-Square Matrix" Exergame Error Sequence for Subject 1

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 1 | 0.083 | 0 | 0 | 0 | 0 | 1 | 0.083 |
| 7 | 3 | 0.375 | 0 | 0 | 0 | 0 | 3 | 0.375 |
| 8 | 2 | 0.2 | 0 | 0 | 0 | 0 | 2 | 0.2 |

Table B.14 "9-Square Matrix" Exergame Error Sequence for Subject 2

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 1 | 0.125 | 3 | 0.375 | 0 | 0 | 4 | 0.5 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Table B.15 "9-Square Matrix" Exergame Error Sequence for Subject 3

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0 | 1 | 0.1 | 0 | 0 | 1 | 0.1 |
| 2 | 0 | 0 | 1 | 0.071 | 0 | 0 | 1 | 0.071 |
| 3 | 1 | 0.167 | 0 | 0 | 0 | 0 | 1 | 0.167 |
| 4 | 1 | 0.167 | 0 | 0 | 0 | 0 | 1 | 0.167 |
| 5 | 0 | 0 | 2 | 0.167 | 0 | 0 | 2 | 0.167 |
| 6 | 0 | 0 | 1 | 0.083 | 0 | 0 | 1 | 0.083 |
| 7 | 0 | 0 | 0 | 0 | 1 | 0.125 | 1 | 0.125 |
| 8 | 0 | 0 | 2 | 0.2 | 0 | 0 | 2 | 0.2 |

Table B.16 “9-Square Matrix” Exergame Error Sequence for Subject 4

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 0.071 | 0 | 0 | 0 | 0 | 1 | 0.071 |
| 3 | 3 | 0.5 | 0 | 0 | 0 | 0 | 3 | 0.5 |
| 4 | 0 | 0 | 1 | 0.167 | 0 | 0 | 1 | 0.167 |
| 5 | 6 | 0.5 | 0 | 0 | 0 | 0 | 6 | 0.5 |
| 6 | 1 | 0.083 | 2 | 0.167 | 1 | 0.083 | 4 | 0.333 |
| 7 | 3 | 0.375 | 2 | 0.25 | 0 | 0 | 5 | 0.625 |
| 8 | 2 | 0.2 | 0 | 0 | 0 | 0 | 2 | 0.2 |

Table B.17 “9-Square Matrix” Exergame Error Sequence for Subject 5

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 2 | 0.143 | 0 | 0 | 2 | 0.143 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 1 | 0.083 | 0 | 0 | 1 | 0.083 |
| 6 | 0 | 0 | 1 | 0.083 | 0 | 0 | 1 | 0.083 |
| 7 | 1 | 0.125 | 3 | 0.375 | 0 | 0 | 4 | 0.5 |
| 8 | 2 | 0.2 | 0 | 0 | 0 | 0 | 2 | 0.2 |

Table B.18 “9-Square Matrix” Exergame Error Sequence for Subject 6

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 2 | 0.333 | 0 | 0 | 0 | 0 | 2 | 0.333 |
| 4 | 2 | 0.333 | 0 | 0 | 0 | 0 | 2 | 0.333 |
| 5 | 9 | 0.75 | 0 | 0 | 0 | 0 | 9 | 0.75 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 3 | 0.375 | 0 | 0 | 1 | 0.125 | 4 | 0.5 |
| 8 | 1 | 0.1 | 1 | 0 | 0 | 0 | 1 | 0.1 |

Table B.19 “9-Square Matrix” Exergame Error Sequence for Subject 7

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 2 | 0.333 | 0 | 0 | 2 | 0.333 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 2 | 0.167 | 1 | 0.083 | 0 | 0 | 3 | 0.25 |
| 6 | 2 | 0.167 | 0 | 0 | 1 | 0.083 | 3 | 0.25 |
| 7 | 0 | 0 | 1 | 0.125 | 1 | 0.125 | 2 | 0.25 |
| 8 | 1 | 0.1 | 2 | 0.2 | 0 | 0 | 3 | 0.3 |

Table B.20 “9-Square Matrix” Exergame Error Sequence for Subject 8

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0 | 1 | 0.1 | 0 | 0 | 1 | 0.1 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 1 | 0.167 | 0 | 0 | 0 | 0 | 1 | 0.167 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 3 | 0.375 | 0 | 0 | 0 | 0 | 3 | 0.375 |
| 8 | 0 | 0 | 1 | 0.1 | 0 | 0 | 1 | 0.1 |

Table B.21 “9-Square Matrix” Exergame Error Sequence for Subject 9

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 2 | 0.143 | 0 | 0 | 2 | 0.143 |
| 3 | 1 | 0.167 | 0 | 0 | 0 | 0 | 1 | 0.167 |
| 4 | 0 | 0 | 2 | 0.333 | 0 | 0 | 2 | 0.333 |
| 5 | 0 | 0 | 1 | 0.083 | 0 | 0 | 1 | 0.083 |
| 6 | 0 | 0 | 3 | 0.25 | 0 | 0 | 3 | 0.25 |
| 7 | 1 | 0.125 | 0 | 0 | 0 | 0 | 1 | 0.125 |
| 8 | 0 | 0 | 1 | 0.1 | 0 | 0 | 1 | 0.1 |

Table B.22 “9-Square Matrix” Exergame Error Sequence for Subject 10

| Step | Error Sequence | | | | | | Total Errors | $P_T(E)$ |
|---|---|---|---|---|---|---|---|---|
| | Insertion | $P_I(E)$ | Deletion | $P_D(E)$ | Substitution | $P_S(E)$ | | |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0.071 | 0 | 0 | 1 | 0.071 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 2 | 0.333 | 0 | 0 | 0 | 0 | 2 | 0.333 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 2 | 0.167 | 1 | 0.083 | 0 | 0 | 3 | 0.25 |
| 7 | 6 | 0.75 | 0 | 0 | 0 | 0 | 6 | 0.75 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# Appendix C
# List of Publications


1. Dwivedi, S., Thiemjarus, S., and Nantajeewarawat, E. (2018). "9-Square Matrix" Aerobic Exercise Recognition using Image Processing. *The 12th International Convention on Rehabilitation Engineering and Assistive Technology (iCREATe)* (pp. 200-204), Shianghai, 2018.

2. Dwivedi, S., Thiemjarus, S., and Nantajeewarawat, E. (2018). VR-based "9-Square Matrix" Aerobic Exercise for Preventing Physical Decline in Older Adults. *The International ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI-NCON)* (pp. 106-110), Chiang Rai, 2018.