

**Serious Games for Virtual Rehabilitation in a Large Scale Virtual Reality  
Environment**

BY

ANDREA ROTTIGNI  
B.S, Politecnico di Milano, Milano, Italy, 2014

THESIS

Submitted as partial fulfillment of the requirements  
for the degree of Master of Science in Computer Science  
in the Graduate College of the  
University of Illinois at Chicago, 2016

Chicago, Illinois

Defense Committee:

Andrew Johnson, Chair and Advisor  
G. Elisabeta Marai  
Pier Luca Lanzi, Politecnico di Torino

## ACKNOWLEDGMENTS

I want to thank Veronica for her patience and because she has always supported me since the first day.

I want to thank my family for all the sacrifices that made possible being here.

I want to thank Filippo and Marco who share with me the most of the time here in Chicago

I want to thank Andrea and Davide, who I share the office with, in the second semester.

I want to thank all the other Italians who shared with me this incredible experience.

I want to thank all the EVLers, in particular Luc, Lance and Arthur

I want to thank my advisors Prof. Johnson and Prof. Marai and my Italian advisor Prof. Lanzi.

AR

## TABLE OF CONTENTS

<b><u>CHAPTER</u></b>	<b><u>PAGE</u></b>
<b>1 INTRODUCTION AND MOTIVATIONS . . . . .</b>	<b>1</b>
1.1 Background: Virtual Rehabilitation . . . . .	1
1.2 Serious Games . . . . .	3
1.3 Structure . . . . .	5
1.4 Innovative Aspects . . . . .	6
<b>2 STATE OF THE ART . . . . .</b>	<b>8</b>
2.1 Virtual Rehabilitation . . . . .	8
2.2 Serious Games for Virtual Rehabilitation . . . . .	10
2.3 Limits of the current approaches . . . . .	22
<b>3 CAVE2 ENVIRONMENT . . . . .</b>	<b>24</b>
3.1 CAVE2 Architecture . . . . .	25
3.2 Tracking System . . . . .	25
3.3 Audio System . . . . .	26
3.4 Developing Application for CAVE2 . . . . .	26
<b>4 METHODS . . . . .</b>	<b>27</b>
4.1 System Overview . . . . .	27
4.2 Application Design . . . . .	29
4.3 Training modes . . . . .	31
4.3.1 Tutorial Mode . . . . .	31
4.3.2 Random Objects Mode . . . . .	33
4.3.3 Progressive Mode . . . . .	33
4.3.4 Custom Mode . . . . .	33
4.4 Trajectory Mode and Distorted Mode . . . . .	34
4.4.1 Distorted Reality . . . . .	34
4.5 Patient and Therapist Interaction . . . . .	36
4.6 User Interaction . . . . .	36
4.7 Implementation Details . . . . .	37
4.7.1 Scene Description . . . . .	37
4.7.2 Joint Position and Orientation . . . . .	40
4.7.3 Patient Identification . . . . .	41
4.7.4 User Interface and Interaction . . . . .	41
4.7.5 Perspective . . . . .	42
4.8 Data Gathering . . . . .	44

## TABLE OF CONTENTS (continued)

<u>CHAPTER</u>		<u>PAGE</u>
<b>5</b>	<b>CAVECHEF . . . . .</b>	<b>45</b>
5.1	System Overview . . . . .	45
5.2	Application Design . . . . .	47
5.3	Tracking . . . . .	47
5.4	Game Description . . . . .	49
5.4.1	Game Topic . . . . .	49
5.4.2	Tasks and Obstacles . . . . .	49
5.4.2.1	Level One Description . . . . .	52
5.4.3	Scoring System . . . . .	52
5.5	Implementation Details . . . . .	54
5.5.1	Joint Position, Orientation and Scale . . . . .	54
5.5.2	Scene Description . . . . .	55
5.6	User Interface and User Interaction . . . . .	56
5.7	Highly Motivational Environment . . . . .	58
<b>6</b>	<b>USER STUDY AND EVALUATION . . . . .</b>	<b>59</b>
6.1	User Study Description and Motivation . . . . .	59
6.2	User Study Settings . . . . .	62
6.2.1	Demographic Data . . . . .	62
6.2.2	Technical settings . . . . .	62
6.2.3	Assumptions . . . . .	63
6.3	Results . . . . .	63
6.3.1	Objects Caught . . . . .	63
6.3.2	Completion Time . . . . .	65
6.3.3	Adaptation to Distorted Reality . . . . .	68
6.3.4	Objects Position Detection Errors . . . . .	69
6.3.5	Trajectory Mode . . . . .	70
6.3.6	User Interface Interaction . . . . .	71
6.4	Questionnaire . . . . .	73
6.4.1	Third Person vs First Person . . . . .	75
6.4.2	Humanoid Avatar vs Cartoon Avatar . . . . .	76
6.4.3	User Interface . . . . .	76
6.4.4	General Comments . . . . .	76
<b>7</b>	<b>CONCLUSION . . . . .</b>	<b>78</b>
7.1	Future Works . . . . .	79
	<b>CITED LITERATURE . . . . .</b>	<b>80</b>
	<b>VITA . . . . .</b>	<b>83</b>



## LIST OF FIGURES

<b><u>FIGURE</u></b>		<b><u>PAGE</u></b>
1	CAVE2. . . . .	24
2	RehabJim. . . . .	28
3	Tutorial: on the left the avatars are introduced to the user. On the right the user reaches the training position. . . . .	32
4	Tutorial: on the left the user learns how to reach objects, on the right the user learn how to interact with the user interface. . . . .	32
5	Trajectory: the image on the top shows the best path to reach the object is shown to the user. Distorted Mode: in the image on the bottom left and right movements are swapped. . . . .	35
6	Patient-therapist interaction: the top image shows the patient and two therapists. Help menu: the image on the bottom shows the user looking at the help functionality. . . . .	38
7	Visual reward when the user reaches the object. . . . .	43
8	CAVEChef, overview of the game. . . . .	46
9	Plastic plate: top left figure shows a plastic plate with three markers to track users' joints. In particular that one is for the left knee. Wand controller: top right image shows the wand controller with for markers to track users' hand. User wearing all the markers: bottom figure shows an user wearing all the necessary markers. . . . .	48
10	Top image: A simple kitchen with light and shadows to improve the depth perception. Bottom image: The avatar inside the kitchen mimic users' movements. . . . .	50
11	Level One: in the top image the user has to bring the tomato to the pan to complete the subtask. Level one: in the bottom image the user has to move the pan to the stove to prepare the tomato sauce. . . . .	53
12	Comparison of number of objects caught when using a first person perspective and a third person perspective. It is represented the average number of objects caught by each subject in the three sessions. . . . .	64
13	Comparison of number of objects caught when using a first person perspective and a third person perspective. It is represented the average number of objects caught by each subject in the three sessions. This picture highlights the small difference between the two perspectives. . . . .	64
14	Time for completing the random objects training session. . . . .	66
15	Time for completing the random objects training session in distorted reality. . . . .	66
16	Time for completing the random objects training session with trajectory enabled. . . . .	66

## LIST OF FIGURES (continued)

<u>FIGURE</u>		<u>PAGE</u>
17	Average completion time of the three different training sessions for each perspective. . . . .	68
18	Average number of object position detection mistakes when reaching 30 objects in each different training session and for each perspective.	71
19	Comparison of the average time spent to complete a task when using wand controller and speech recognition. Only tasks that were successfully completed with both interaction techniques are considered to compute this statistic. . . . .	72
20	Left: Percentage of users that used the help functionality when using the speech recognition. Right: Percentage of users that used the help functionality when using the Wand Controller. . . . .	74
21	Box-plot of questionnaire answers. User were asked to evaluate the ease of use of several features, where 0 means difficult and 10 means easy. Users were also asked to evaluate the usefulness of the tutorial, where 0 means useless and 10 means useful. . . . .	74
22	Distribution of questionnaire answers about which perspective was easier to use. . . . .	75

## LIST OF ABBREVIATIONS

HMD	Head Mounted Display
GUI	Graphical User Interface
ADL	Activity of Daily Living
CAVE	CAVE Automatic Virtual Environment
TBI	Traumatic brain injury
AF	Augmented Feedback
UIC	University of Illinois at Chicago

## SUMMARY

Virtual Rehabilitation is becoming popular in clinical research thanks to the development and availability of new immersive technologies. For example, the rehabilitation of post-stroke patients frequently uses virtual reality, and sometimes employs a first-person Serious Games approach to make therapy more enjoyable for patients (see Section 2). In such approaches, the patient experiences the virtual environment in a first-person view. Unlike the existing body of work, we present two virtual rehabilitation applications and we evaluate one of them, which uses both a first person perspective, where the patient controls only the arms of a 3D avatar, and a third person perspective, where the patient controls with his/her body a 3D avatar. The application was commissioned by biomechanics researchers at a major center for rehabilitation research, and could be used to compare the learning-curve and time needed for rehabilitation across different therapies and environments. One of the main challenges in this work is the design of simple, non-distracting avatars and virtual world, which allows the patient to focus on correctly performing the rehabilitation exercises while providing an enjoyable and entertaining experience.

Our first solution is RehabJim, a third-person and first-person serious game for the rehabilitation of upper limbs in post-stroke patients, with particular emphasis on reaching movements. Our second solution is CAVEChef, a third-person serious game for walking rehabilitation in post-stroke patients. Our solutions use a state-of-the-art virtual reality environment and Unity

## SUMMARY (continued)

3D. The first one uses a Kinect controller, while the second one ten Vicon Bonita infrared cameras as tracking devices.

## CHAPTER 1

### INTRODUCTION AND MOTIVATIONS

In this chapter we introduce our work. In the first two sections we provide a short background about virtual rehabilitation and serious games. In section 1.1 we analyze virtual rehabilitation, explaining its main advantages and disadvantages, and which technologies are usually employed in such applications. In section 1.2 we present Serious Games, and explain why they are considered powerful learning tools. Section 1.3 describes the structure of our work, giving a brief description of the content of each. Section 1.4 is dedicated to the innovative aspects of our applications and we explain why they can be helpful to the rehabilitation community.

#### **1.1 Background: Virtual Rehabilitation**

There are three crucial aspects that characterize a rehabilitation therapy: repetition, feedback and motivation. Repetition is necessary in any learning process but unfortunately at the same time is one of the causes that often leads to unsuccessful therapies. Feedback is used to give patients information about their performance. Motivation can be achieved, for example, by rewarding patients when performing well or by adding some sort of competition. These three aspects can be effectively addressed by virtual reality applications, especially as virtual reality offers the possibility of real time feedback of task performances. [1].

Of course virtual rehabilitation applications do not have only advantages. Rizzo and Kim [2] describe in detail their strengths, weaknesses, opportunities and threats (SWOT). We do

not provide here the complete SWOT analysis, but we only describe the most important aspects. Among the strengths, we consider the following to be particularly interesting: "real time performance feedback" and "gaming factor to enhance performance". The former gives the possibility to enhance rehabilitation results by providing feedback which are not possible in the real world. Thanks to real time feedback patients can understand in a short time how well they are performing and, as a consequence, improve their performance. The gaming factor is instead an effective way to improve motivation and reduce the burden of repetitiveness (See section 2.2). Unfortunately virtual rehabilitation has some caveats. Among these, designing an interface for interacting with the application is probably one of the most challenging aspects. Also, tracking devices can be a severe limitation: they can be too invasive and make patients uncomfortable, not be enough accurate or track patients only in a very limited range. For these reasons we use two different tracking devices in the two applications that we describe, in order to achieve the best performances as possible. The growth of the gaming industry will probably lead to a growth in the virtual rehabilitation field. Gaming platforms are currently present in almost every home and people are becoming more and more familiar with video games. Therefore if someone might need some sort of rehabilitation for a certain period of his/her life, he/she is already familiar with video games, and so naturally inclined to embrace virtual rehabilitation. The perception that rehabilitation can be performed without a therapist is probably one of the biggest threats. However domain scientists all agree that therapist guidance is important to achieve successful results.

Virtual rehabilitation can be performed using several hardware and software devices, and depending on the device used, we can create an environment with different degrees of immersion. Among such devices the most popular are a standard 2D graphics display, 3D display and HMD (head mounted display). Tradeoffs must be considered when choosing the underline technology. For example choosing simple 2D displays are clearly cheaper, but providing an highly motivating environment might be difficult [1].

## **1.2 Serious Games**

This section provides some information about Serious Games, and discusses why they are considered a powerful tool to educate players. Serious Games are used for several different contexts such as Military, Health Care and education.

First of all, since Serious Game is a broad term, it is better to provide a rigorous definition. According to Houda Mouaheb et al. [3] there are two different definitions that describe accurately the concept of Serious Games. In the first one, Serious Games are defined as "utilities video games" which try to transform players by improving their skills or their knowledge of a particular phenomenon or their adaptation to the environment. The second defines Serious Games as computer applications which should not only contain entertaining features, but also aspects of teaching and learning so that they can go beyond the mere entertainment. Houda Mouaheb et al. also provide a list of features that identify a Serious Game:

- Learning Process: the journey through the levels of the game should be associated with a learning process.



- Fun Component: a serious game should also include a fun component because this is the key aspect of a good game and also can be an effective way to maximize the learning process by increasing the learner's motivations.
- Learning Objectives: a serious game should address one or multiple learning objectives such as physical training or teaching.
- Audience: all age groups can be addressed with a serious game.

Andrew J. Stapleton [4] provides an interesting comparison between Serious Games and school. The main difference between these two educational strategies is who is in control of the learning experience: in games, players are in control of it, while in schools, teachers guide learners. This difference is one possible explanation of why Serious Games are considered so powerful, they use a learner-centered approach. With this approach players are in charge of the learning journey which is likely to result in an enjoyable experience.

Dennis Charsky [5] provides a meaningful categorization of the features that each game should possess highlighting how these features are treated in Serious Games. But before going through the list it is important to define a term which is particularly correlated to Serious Games: "Edutainment". Edutainment was first introduced by Robert Heyman in 1973 to identify something which combines education and entertainment. Serious Games clearly belong to this genre. Here the list of features by Dennis Charsky:

- Competition and Goals: in Serious Games goals usually correspond to the learning objectives and adding some form of competition is an effective way to motivate users and make their experience more enjoyable.

- Rules: rules are useful to determine which actions player can or cannot do. In Serious Games they are usually fixed and particularly important in educational game such as games that teaches physics, where rules might correspond to laws of physics.
- Challenges: tasks and activities represent the challenges of a serious game and they are part of the learning process. By completing them users can practice and improve their skills in order to gradually reach the learning objectives of the game.
- Fantasy: fantasy elements are used to create a highly motivating environment and improve the game play. Fantasy can be used in Serious Games as rewards to reinforce correct behaviors, but also to improve the gamer knowledge.

### **1.3 Structure**

After reviewing the literature about virtual rehabilitation and serious games for virtual rehabilitation we present, in this work, our two applications: RehabJim and CAVEChef.

RehabJim is a Serious Game for upper-limbs rehabilitation in post-stroke patients. Patients can perform a rehabilitation session by either using a first person perspective or a third person perspective. In the first person perspective the camera is oriented in such a way that only the avatar's arms are visible in the virtual world and they are a natural extension of patient's real arms. In a third person perspective the entire avatar body is visible in the virtual world. The patient controls the avatar with his/her body.

CAVEChef is a Serious Game for motor rehabilitation set in a commonly used environment: a kitchen. We believe that this setting can be helpful to involve patients in the rehabilitation experience.

Then we report the result of a preliminary evaluation of RehabJim. Ten users were asked to perform several rehabilitation training sessions using both perspectives. Even though the people who try our application weren't real patients we believe that by combining their results together with their feedback we can obtain useful insights. We are especially interested in understanding if a third person perspective might be an effective alternative to the more common first person perspective or even improve the overall experience.

In the conclusion of this work we describe what can be the future of our two applications and how they might be improved.

#### **1.4 Innovative Aspects**

Our two applications contain some innovative features, which might help to create a better rehabilitation experience for patients. First of all we can find very few examples of Serious Games for virtual rehabilitation that allow switching from a first person perspective to a third person perspective. The third person approach has been rarely explored in the past and our applications together with the evaluation of RehabJim can be useful for domain scientists to understand which of the two strategies can help patients to adhere better to their rehabilitation routine.

Walking rehabilitation applications usually employ a treadmill or other ad-hoc devices (See section 2.2). On the other hand, for CAVEChef, we decided to leverage the space offered by CAVE2 and the CAVE2 tracking system to design the game in order to exploit this space as much as possible and to avoid external devices. Furthermore walking inside the game with your own body is something that we do not often see in video games developed for the CAVE. Usually

they employ a controller (e.g. a Playstation Wand Controller [6]) to navigate in the game. Therefore CAVEChef might open a new way to interact with games in CAVE-like environments.

## CHAPTER 2

### STATE OF THE ART

This chapter presents how other work addresses the challenge of using serious games to perform Virtual Rehabilitation. Although the distinction between serious games for virtual rehabilitation and virtual rehabilitation is not always very clear, we prefer to analyze these two categories in two separate sections. In section 2.1 we analyze some different approaches and technologies used in virtual rehabilitation which use both virtual and augmented reality. In section 2.2 we describe the state of the art in the field of serious games for rehabilitation. In the concluding section we briefly describe some of the possible limitations of the previous work and we introduce our two applications.

#### **2.1 Virtual Rehabilitation**

One application that uses virtual rehabilitation outside the game field, is Hoerman et al. [7], where augmented reality is used to perform mirror therapy following the "Berlin Protocol for Mirror Therapy". In this approach they use a 22 inches screen located in front of the user who puts his/her hand behind the screen; in this way the hand is hidden by the screen. A Webcam is employed to capture the hand movements, which are then reproduced and mirrored on the screen.

Even though there is a scoring system, another application which employs virtual rehabilitation without an explicit use of game techniques is the work of Patel et al. [8]. They use three different mirroring tasks and we can find virtual rehabilitation especially in two of them:

- Flexion task without exoskeleton: hands are tracked using a glove that can detect the angles between the fingers. Both hands are hidden by a screen and they are reproduced on the display. Users only move their unaffected hand to align it with a target. However the visual feedback given by the application makes the users believe that they are also moving their affected hand.
- Extension task with exoskeleton: this task is a variation of the previous one. The affected hand is surrounded by an exoskeleton which is controlled by the unaffected hand. In this way, when the user tries to hit the target using his/her unaffected hand, he/she has not only the sensation of moving his/her affected hand thanks to the visual feedback, but also moving the the affected hand by means of the exoskeleton.

Klein et al. [9] developed an augmented reality application for virtual rehabilitation. The application uses a webcam to capture the front view of a patient, a glove to easily detect patients' hands movements, and an electromyography device to capture signals from patient's affected arm. The images that are captured by the webcam are projected onto a screen. In the rehabilitation training session a computerized arm is added to the video sequence captured using the webcam, which is used to perform three different exercises for shoulders:

- Shoulder abduction: this consists of moving the virtual arm out the side of the body with a range from 0 to  $-\pi$  radians.
- Shoulder flexion: this consists of moving the virtual arm up and down in front of the user's body with a range from 0 to  $\pi$  radians degrees.
- Shoulder horizontal flexion: this exercise is similar to the previous one, but now the arm is moved horizontally in front of the user's body with an angle that ranges form 0 to 40 degrees.

## **2.2 Serious Games for Virtual Rehabilitation**

In this section we discuss about several examples of Serious Games for rehabilitation.

The first one is the work of Ma et al. [10], which consists of three different games with strenghtening execrcise for fingers, hands and limbs in virtual immersive environments. To create the virtual immersive environment they use a high resolution HMD, Virtual Research 1280 stereo device.

- Catch-the-orange: patients hold virtual baskets to catch oranges that are falling in random positions from the top of the screen. The aim of the game is catching as many oranges as possible and keeping the basket in a horizontal orientation, otherwise oranges may fall out from the basket.
- Fishing game: in this game patients are under water and they are using their bare hands to catch fish that are swimming. When a fish is captured it disappears. In order to be captured a fish need to be touched by both hands simultaneously.

- Whack-a-mouse: in this game the patient is holding a hammer and he/she tries to hit mice that appear on the table. However each mouse stays in a specific position only for a limited amount of time. Distractors are added to the game: in addition to mice, there are dogs in the scene which should not be hit with the hammer.

One interesting aspect of these games is that the gameplay can be adapted for each patient. For example the game can be made easier for a patient with severe motion disabilities and more difficult for a patient with few motion problems.

We can find three other serious games for rehabilitation of upper-limbs in post-stroke patients in the work of Burke et al. [11]. This system leverages webcams that allow the user to interact with the game. To track player's hands they use a special kind of gloves or a particular monochromatic object. One clear advantage of this approach is that it does not require expensive technology and the configuration seems very simple. However these advantages come at a price: it seems very difficult to create a sensation of immersion in the virtual world by simply using webcams and personal computers.

- Rabbit Chase: in this game several holes are displayed on the screen. A rabbit is generated randomly in one hole at a time and users need to catch the rabbit before the time expires and the rabbit changes hole.
- Bubble Trouble: this game is similar to the previous one, but in this one bubbles appear in random position on the screen and the player has to reach them before time expires. There are two variations of this game: in the first one only the movements of one hand are tracked while in the second one both hands can be used to capture the bubbles.



- Arrow Attack: here four boxes and two arrows are displayed on the screen. One arrow is pointing to the right while the other one is pointing to the left and each arrow is moving across the screen. To complete the level the player has to touch the two arrows simultaneously as soon as they enter in the boxes.

Another example of a serious game is the one developed by Mumford et al. [12] [13] called Element System. One of the main differences with respect to the previous games is that this one was developed for patients affected by TBI (traumatic brain injury). The game runs on a commercial computer and the virtual environment is displayed onto an horizontal LCD panel. Users move a physical cylinder over the display which is tracked using stereo cameras. One important feature introduced by this work is the so called Augmented Feedback (AF) for two reasons: AF is an effective way to give users additional information about their movements, and to pay attention to the effects produced by their movements. The Element System consists of several tasks. These tasks can be subdivided in two categories:

- Goal Oriented:
  - Bases: where we can find a home base and three different targets which are highlighted in a predefined order.
  - Random Bases: this task is exactly the same of the previous task, but now targets are highlighted randomly.

- Chase Task: in this task targets can appear only in nine locations around the home base. The nine locations are not displayed on the screen, which is initially blank, and only the current target is shown.
- Go-no-go: this task is a variation of the previous one. The configuration is basically the same, but visual distractors are added to the setting.
- Exploratory:
  - Mixer: the scene is composed by nine circles arranged on the screen. When users move the cylinder over one of the circles they trigger animation and sound effects. Users can choose the order in which they visit the circles in order to create different combinations of sounds.
  - Squiggles: in this task users can combine four different objects, each one with a different trail animation sound, to create visual and sound effects on the screen.
  - Swarm: here users need to use both hands to interact with the virtual environment. The configuration is similar to the previous one: users can move around four different objects with different shapes and effects. But now there is a so called "audiovisual relationship" among objects: objects have different audiovisual effects based on how they are combined.

A commercial computer combined with a 3D motion tracking system and an LCD projector are the tools used by Turolla et al. [14] to track users' movements and reproduce the virtual reality environment. They use different scenarios but in everyone the patients hold a physical

object and they need to complete some simple tasks. For example, one of these tasks consists of putting a glass on a shelf. The glass and the shelf are recreated in the virtual world, which users can see projected on a large screen. One innovative aspect of this game with respect to the previous ones is the interaction among patients and therapists. The therapist holds a physical glass, which is tracked using a marker, and puts it on the virtual shelf. After that the patient tries to reproduce the therapist's movement. In order to facilitate the task the path described by the therapist while putting the glass on the shelf is displayed in the background. The therapist can create different paths and personalize them according to the patients' needs.

The Rehabilitation Gaming System (RGS) of Cameraio et al. [15] uses a standard computer, an LCD monitor, a video camera and a pair of gloves to track users' hand movements while arm movements are captured using a vision based tracking system. The game consists of two calibration phases and the training phase. The training phase consists of three tasks in an increasing order of complexity. All three tasks are performed in the same minimalistic virtual environment where users control the arms of an avatar using a first person perspective. These tasks are:

- Hitting: balls are thrown towards the user who has to hit the balls using his/her virtual arms. The complexity of this task can be tuned using three different parameters: speed of the spheres, time interval between spheres and position of the sphere. The game dynamically adapts these three parameters according to user performance
- Grasping: this task is very similar to the previous one, but the user has not only to hit the ball but also to grasp the sphere to get points.

- Placing: in this last task the environment is enriched with some baskets at the bottom of the scene, each one with a different color. Colored spheres are thrown towards the user who has to grasp and put them in the basket corresponding to their color.

A game that tries to go beyond performing simple tasks is VOTA developed by Adams et al. [16]. This game is developed following the same principle that we use in our second application: patients are inclined to use an application if the game is related to real world actions [16]. For this reason the environment of this game is a kitchen and the exercises in the game are activities of daily living (ADL). The system employs a standard computer to display the scene and a Microsoft Kinect [17] to track user's movements. Users sit in front of the display and they can be in two different interaction modes according to the location where they are. They can move freely around the virtual world or they can be in an activity area where the kinect is employed in order to map the user's arm movements to the avatar's arm movements, which is displayed using a first person perspective. Each activity can be decomposed into several tasks and subtasks which must be completed sequentially in order to successfully complete the activity. For example, one of the activities is "Meal Preparation". The activity can be decomposed into four tasks:

- Get a Pan: this task is divided in four simpler subtasks such as open cabinet, reach a pan, bring the pan to the stove, close the cabinet.
- Pour oil in the pan: this task can be subdivided basically in the same subtasks of the previous one.
- Turn on stove: User only needs to reach the stove control to complete this task.

- Cook Eggs: this task is subdivided in four subtasks: reach for the first egg, bring the first egg to the pan and then do the same two operations for the second egg.

We consider the subdivision in activities, tasks and subtasks a good strategy to design games for rehabilitation which can be more challenging.

Another game in which tasks are inspired by daily activities is the work of Sucar et al. [18]. The goal of this application is hand rehabilitation. The system is composed by an off the shelf computer, a webcam to record user's movements and a handgrip which plays the role of tracking device. This game is composed by three mini-games:

- Steak: the environment of the game consists of a steak and a grill. The goal is to avoid burning the steak. In order to do that the user needs to touch the steak before it is too late. When he/she touches the steak it disappears and another one appears in a different position. The distance between two consecutive steaks is computed automatically, adapting the difficulty of the game.
- Clean Window: in this second game the scene is composed of a window and an hand holding a wiper. Stains on the window are generated by the game and the user needs to clean them. As soon as the user cleans one stain another one appears in another position. The main difference with the steak game is that here the movement of the hand is bidirectional.
- Fly Killer: the goal of this game is to kill a mosquito before it reaches the patient's hand. The mosquito moves over a window and the virtual hand of the patient holds an

insecticide sprayer. The mosquitos speed and the pressure needed to spray the insecticide are tunable parameters.

Daily activities are also considered by Jacobs et al. [19] to address upper-limbs rehabilitation in post-stroke patients, but in a quite different way with respect to the previous two games described: players hold physical everyday objects. The game runs on a conventional computer connected to a 27-inch multi-touch display which is able to detect objects. Before starting the game the user needs to put the object which he/she will use during the game on the display, so that the system can detect the object fingerprint. The game consists of several vertical bars (obstacles) that enter from one side of the screen. The user should navigate around the obstacles and sometimes skip over them by lifting the object. To motivate the user to place the object again on the screen some collectible items are generated in the game which can be captured using the object. The game difficulty dynamically adapts to patient's performance: vertical bars move faster if the patient is performing well.

The Sorcerer's Apprentice [20] is a serious game that mixes a highly motivational environment with rehabilitation of shoulders. The target of this game are patients with Shoulder-Impingement-Syndrome. The Microsoft Kinect is used as a tracking device for patients' arms and the game is displayed on a standard LCD display. This game can be classified as a fantasy game: the patient's character in the game is empowered with magical skills. The gameplay can be divided in two phases:

- Exploration: the player goes around the magical world to look for the so called "primary artifacts" which are items that represent the exercises. These items are generated in

random positions and the patient should collect them in order to enable the exercise mode.

- Exercises: the rehabilitation session takes place in this phase of the game. Patients need to complete all the exercises, corresponding to the items picked up in the previous phase, in order to unlock new levels. The player is provided with visual feedback during the exercises to make clear whether the movements that he/she is performing are correct or not. Also textual instructions are added to the screen when the patient's movements are out of bounds.

Not all the games that we can find in literature are for rehabilitation of upper limbs. They are certainly the most popular, but we can find a few examples of serious games for rehabilitation of lower limbs. Among these there is the work of Boian et al. [21] who developed a haptic interface to interact with the virtual world for ankle rehabilitation. The interface is called "Rutgers Ankle Haptic Interface" and it is a rather complex mechanical setup consisting of a platform where users can put one foot. To interact with the application two platforms are necessary, one for each foot. The user sits on a medical chair; a monitor displaying the virtual world is located in front of him/her. The rehabilitation session consists of two rather similar exercises while the virtual environment is rather different.

- Airplane Exercise: in this exercise the patient is flying an airplane by moving his ankle on the platform. The goal of the exercise is to fly the plane through several hoops that are displayed on the screen. Hoops appear in the positions specified by the therapist in order to stimulate a specific ankle part.

- Boat Exercise: the concept of the second game is similar to the previous, the patient is driving a boat using his/her ankle and the movements are tracked by using the platform underneath the foot. Hoops are substituted by buoys. However this exercise is more complex because the boat needs to stay above water while in the previous exercise the plane could take any path to reach the next hoop.

The complexity of these two exercises can be tuned by adding some visual effects such as changing weather conditions or simulating turbulence and so on. Boian et al. [22] developed also a variation of the platform, which allows user to stand. Thanks to this variation we can find other two different virtual reality exercises for lower limbs:

- Street Crossing: as the name suggests the goal for the user is to cross a road: he/she is in front of a traffic light and needs to reach the other side of the road while the light is still green.
- Park Walk: the user is walking in a park and needs to follow the path represented by the road in the park. The complexity of this exercise can be tuned by changing for example the number of obstacles, the season or by adding noises as distractors.

“Kinect-o-Therapy: Making Rehabilitation fun, engaging and rewarding” is a video game [23] composed by four serious games for upper-limbs, walk and attention deficiency disorder rehabilitation. As the name suggests the main purpose of this application is to add a fun component to the rehabilitation environment. The software is developed for standard computers and users’



movements are tracked using the Microsoft Kinect. The application can be customized according to users needs. We analyze only the games for upper-limbs and walk rehabilitation:

- **Shoulder Abduction:** the patient sees an avatar on the screen which mimics his/her movements, it is basically the same as being in front of a mirror. The user needs to move in a straight line on the screen with his/her shoulder in order to complete the level. If he/she doesn't follow a straight line an unsuccessful attempt is registered by the system. Auditory feedback is used to indicate to the user whether he/she is performing the right movement or not.
- **Path Follower:** this mini-game is designed specifically for walking and balance rehabilitation. A path seen from the top is displayed on the screen and the patient is represented with a white marker. The patient needs to complete going from the starting point to the exit point without leaving the path. When the user leaves the path it changes color. Paths are ordered in increasing order of difficulty.
- **Play Along:** the perspective is the same of the first game. However this time more than one person is tracked and displayed on the screen. This feature can be exploited to perform rehabilitation in this way: one of the two avatars tracks movements performed by a therapist, while the patient tries, using his/her avatar, to reproduce the movements of the therapist's avatar. This is one of the few examples which allows interaction between therapists and patients in the virtual world.

Most of the games that we have analyzed so far use standard computers and LCD displays, but since mobile devices are becoming pervasive there is an increasing interest in developing serious games for rehabilitation for them.

For example PRehab [24] is a serious game for arm rehabilitation developed for a graphics tablet. Using the mouse the patient has to reach some virtual targets to increase his/her range of motion. The final objective of the game is to protect Nature by collecting some items called Gemstones. PRehab is composed by several mini-games with basically the same structure. Each mini-game is subdivided into levels. The player controls his/her character and guides him towards the end of each level by eliminating enemies in order to collect the Gemstones. To increase the score the player can collect coins. As mentioned above, the rehabilitation task in this game is to reach virtual targets, the enemies. The position of the enemies is adjusted according to the patient's performances.

Another device, which is becoming popular for post-stroke rehabilitation, is the Nintendo Wii Fit Balance Board. RehaLabyrinth [25] is a game that employs this device to perform balance rehabilitation in post-stroke patient. In addition to the balance board the game needs a conventional computer with an LCD display. The player controls a white ball located inside a maze through the balance board. The goal is to guide the ball to the exit of the maze and, at the same time, collect as many coins as possible to increment the final score. The game provides a set of predefined levels, but further levels can be added to personalize them for each patient. Also the Wii Fit sensitivity can be tuned according to patients' needs.

### 2.3 Limits of the current approaches

Most of approaches used so far fail in providing a sensation of immersion. Although using a standard computer has certainly some advantages such as price and portability of the application, it might not be enough to engage patients. Head Mounted Display are a good way to make patients feel like they are in the virtual world, however they have several limitations when the rehabilitation task requires patients to move in the real world.

There are few research papers that deal with leg rehabilitation in virtual reality specifically without using treadmills or other specific kinds of mechanical devices such as name probably because in normal environments there is not enough space to address walking. Thanks to CAVE2 we can develop a game which addresses lower-limbs rehabilitation in virtual reality without using invasive devices.

Introduce an interest concept: using daily activities as rehabilitation tasks to make users more comfortable. This idea inspired also our second application, but we track the entire body of the patient which is free to move around, instead of sitting on a chair, making the game more similar to the real cooking activity.

Not many games allow interaction among patients and therapists in the virtual world. We think that this interaction can create interesting rehabilitation dynamics and for this reason in our first game both the patient and the therapist have their own avatars in the virtual world.

One major difference between our two applications and most the other approaches is the possibility to switch between third and first person perspectives. This feature will allow domain

scientists to discover whether a third person perspective can make the rehabilitation process faster, while we compare the user experience using one perspective rather than the other one.

## CHAPTER 3

### CAVE2 ENVIRONMENT

In the chapter we will briefly describe the CAVE2 environment which is the enabling technology of our application. CAVE2 is a hybrid immersive reality system. The advantages of using scalable-resolution display walls and and virtual-reality environment are combined together in CAVE2. In section 3.1 we analyze the architectural structure of CAVE2, in section 3.2 we describe how the tracking system works and in section 3.3 we illustrate the audio system. In the concluding section we briefly explain how to develop an application for CAVE2.



Figure 1: CAVE2.

### **3.1 CAVE2 Architecture**

Users in CAVE2 can enjoy a 320-degree panoramic environment. 72 displays are arranged in 18 columns consisting of 4 displays. Each pair of displays is connected to a different computer for a total of 36 computers plus two master nodes. Each display is a near-seamless passive stereo LCD panel and has a resolution of 1366x768 pixels, providing an overall resolution of 36 megapixel per eye.

### **3.2 Tracking System**

Thanks to 10 infrared cameras, arranged in a circular configuration above the displays, CAVE2 allows the optical motion tracking system to know in real-time the position in space of a user who wears a particular pair of 3D glasses equipped with retroreflective markers, and also the orientation of his/her head, guaranteeing a three-dimensional perspective that varies according to the movements of the user inside the space. At the same time, the system can track the 6DOF of multiple controllers equipped with retroreflective markers in different configurations, which can be used in order to interact with specific applications. Cameras are arranged in order to track as much area as possible.

Cameras are not the only choice to track people inside CAVE2. A Microsoft Kinect V2, placed at a height of approximately 8 inches from the ground (20 cm), with an inclination of 4 degrees up, can be used to track the position over time of up to 29 human joints. The maximum number of trackable users is currently 6, and the allowed user-distance from the device is approximately 12 feet. Kinect V2 offers the possibility to use its speech recognition feature to interact with applications.

### **3.3 Audio System**

Twenty speakers are arranged in circular configuration above each column of displays. In addition there are two subwoofers located just outside CAVE2. This combination of speakers and subwoofers provides a 3D virtual sound system which can be useful to increase the feeling of immersion in the virtual world.

### **3.4 Developing Application for CAVE2**

The computer cluster driving CAVE2 can run both Linux and Windows.

- CAVE2 and Linux: OmegaLib offers the possibility to develop native OmegaLib application using the C++ API or by writing Python scripts, which require less technical knowledge and are interpreted by the Python interpreter.
- CAVE2 and Windows: Windows must be used to develop application using Unity3D for CAVE2. Each computer in the cluster runs its own copy of Unity, but thanks to the GetReal3D plugin, developed by MechDyne Corporation, all the 36 computer nodes are synchronized to show the correct portion of the 3D scene. Developing Unity application for CAVE2 is similar to developing Unity desktop application. There are some differences due to the fact that each computer is running a different version of Unity and sometimes they require explicit operations for synchronization.

## CHAPTER 4

### METHODS

In this chapter we present our first application: RehabJim. RehabJim is a Serious Game for upper-limbs rehabilitation in post stroke patients which allows switching between a first person perspective and a third person perspective. In section 4.1 we describe in detail the devices used to realize the application and how the application is integrated with the CAVE2 environment. In section 4.2 we discuss the design principles followed to design this game to create an enjoyable experience for patients. In section 4.3 we list the different training modes implemented, while in section 4.4 we provide some implementation details about the most relevant features. Then, in section 4.5, we describe how data are sampled and stored to be later analyzed. In the concluding section we talk about the user interface of our application and how users can interact with the application.

#### 4.1 System Overview

Our application was developed in Unity 3D, and follows the constraints and the possibilities offered by the CAVE2 environment. A third party wrapper allows us to define the position and orientation in space of each of the 64 screens, in order to render on each of them the correct portion of the scene oriented in a proper way. Each of the 32 computers then runs its own instance of the application and is controlled by a master node, which takes care of uploading content to the whole cluster and which makes synchronous requests to the other nodes.





Figure 2: RehabJim.

For example, generating an object in a random position would require the master node to compute the position and then transmit it to all the nodes, otherwise each computer would produce a different result. Due to wire extension issues, the Kinect v2 device is connected to an external server which streams the position of the joints of the users to each node via a local network.

The application is typically initiated by a therapist from the master node of CAVE2, located just outside the environment, where simple data about the patient can be inserted for logging purposes. Then the patient can start his/her training session inside CAVE2 with or without the help of the therapist. Both patient and therapist are represented by an avatar in the virtual environment and their movements are tracked with the aid of the Kinect. Through the 3D user interface of the voice commands, each user can decide which training mode to start and which features to enable during the session. All data involving the training is then stored onto the master node inside a log file, which can be later used for post-training analysis.

## **4.2 Application Design**

The application is meant to be engaging and enjoyable for patients, adopting the possibilities offered by virtual reality and leveraging the principles of gamification. Specifically our aim is to create an immersive experience for patients. Following several low-fidelity and high-fidelity prototypes and based on initial feedback from the rehabilitation researchers, we adopted a minimalistic environment characterized by a few distinct natural features, a neutral background and an easily understandable color palette. Since an excessive number of elements would have

provided a relevant source of distraction for some patients, we decided to keep the virtual scene as simple as possible.

The movements of the patient are tracked and reproduced by the virtual avatar moving in front of him/her. When using a third person perspective the entire body of the avatar is visible in the scene. When the first person perspective is selected the camera is positioned to show only the avatar's arms.

The domain experts considered photorealism to not be a crucial aspect for rehabilitation. We thus decided to adopt a stylized cartoon shape for our avatar, further maintaining coherency with the rest of the scene. The virtual objects that the patient has to reach are rendered as spheres, in order to have the distance from the user independent of the angle of approach.

A first issue regarding positioning virtual objects involved excluding positions in which they could not be seen by the patient, considering that the avatar occupies a relevant portion of the screen. Since the avatar, having roughly human size, is virtually situated in front of the user and is seen from the back, objects need to appear even further than the avatar in order to be caught: for this reason, our custom seeding algorithm generates objects only in positions reachable in relation to the current position of the patient and never directly in front of him/her.

A second issue involves the perception of depth in the 3D virtual world: in virtual environments, sometimes, it is not easy to distinguish how far an object is and how much we should move in order to reach it. By experimenting with different alternatives, we decided to use the camera angle to place realistic shadows on the scene. These shadows are projected by the avatar

and the objects on the ground. An effective size of the objects was empirically determined to be  $2/3$  of the size of the avatar's hands.

### **4.3 Training modes**

The following training modes were designed on the basis of repeated interviews with the domain experts, in which they discussed the state of the art of therapeutic approaches. In order to allow new users to learn how to use the system, we introduced a two-minute long interactive tutorial to make them feel more comfortable and learn the basics. Three different training modes are provided, one of which is fully customizable by the therapist. At the end of each mode, simple statistics are provided to the user and therapist. All the training modes can be completed using a first person perspective or third person perspective. We describe briefly each mode below.

#### **4.3.1 Tutorial Mode**

The Tutorial Mode ( Figure 3 and Figure 4) is accompanied both by visual text appearing on screen and by a voiceover giving instructions to the patient. The two main types of avatar, the patient's and the therapist's, are presented to the user with graphical animations, then the user is asked to walk to the center of CAVE2 inside a virtual red circle, to the position he/she will have to maintain during the training. The position of the user in this case is computed again by using the Kinect v2. Simple tasks of reaching objects follow, with visual and audio feedback in order to involve the user and get him/her accustomed to the system. The final part of the tutorial involves using the Wand controller in order to enable and use the graphical user interface, as well as the speech recognition feature to activate specific options.



Figure 3: Tutorial: on the left the avatars are introduced to the user. On the right the user reaches the training position.



Figure 4: Tutorial: on the left the user learns how to reach objects, on the right the user learn how to interact with the user interface.

### 4.3.2 Random Objects Mode

The Random Objects mode is meant to be a quick way to run the application every time in a slightly different way. This mode generates, one at a time, a given number of virtual objects in random positions, with a new random seed at each execution. Objects, as mentioned before, are placed only in allowable positions in order not to be occluded. Our algorithm provides a dynamic timeout for each object that also considers the current behavior of the user: if the first time the patient doesn't reach the object in a predefined amount of time, the object vanishes and the next one appears; the system will now allow more time to the user for reaching successive objects. The main purpose of the timeout is to avoid particularly difficult positions for a specific user and to limit the impact of possibly unfortunate randomly-generated positions.

### 4.3.3 Progressive Mode

The Progressive Mode is very similar to the previous one, but focuses on generating progressively distant objects in a predefined direction. The aim of this mode is to start with simple movements and make them more difficult little by little, in order to acclimatize the patient or to allow him/her to better see improvements over time.

### 4.3.4 Custom Mode

The Custom Mode is completely configurable by the therapist through a file present on the master node which operates the CAVE2 environment. The position of each object in space and its related timeout can be simply set by inserting the respective value in the system. This approach allows unlimited types of training, which can be made more specific for the particular situation of each patient.

#### 4.4 Trajectory Mode and Distorted Mode

Our application offers two additional ways to customize the user experience: a trajectory visualization mode and a distorted mode, which can be both activated at any moment with the Wand controller or using voice control.

This feature is intended to improve the movements of the user. The feature employs visible trails which emanate from the hands of the avatar and decay after a predefined amount of time ( Figure 5). This allows both the patient to be more mindful of his/her own movements and the therapist to notice significant details during the training. The trails are accompanied by a visualization of the optimal trajectory to reach a target: when a new virtual object appears, a ray is cast from the nearest hand of the avatar to that object, indicating the shortest path which could be followed to reach it. By visually comparing the white trails and the red trajectory, it is very easy to identify qualitatively the movement error.

##### 4.4.1 Distorted Reality

The current implementation of the *Distorted Reality* mode inverts the movements of the left and right parts of the body, so that if the user moves him/her left arm, the right arm of the avatar will move instead. The main purpose of adding this feature was to study how patients respond when they are given an unusual situation or a new challenge that requires more concentration with respect to the normal task. The movement, in this case, requires a completely new approach by the patient, who typically finds it initially totally counter intuitive. A possible interesting application of this feature would involve people who have limb amputations: for example, even if a patient does not have his/her left arm anymore, sometimes he/she may



Figure 5: Trajectory: the image on the top shows the best path to reach the object is shown to the user. Distorted Mode: in the image on the bottom left and right movements are swapped.



experience phantom pain – feeling pain in that limb as though it was still there. Moving the right limb and seeing the left one moving instead in virtual reality could help the patient relieve that pain.

#### **4.5 Patient and Therapist Interaction**

Having up to five therapists and the patient inside tracked at the same time offers an additional training mode. The therapist can perform some particular movements using his/her avatar that the patient should try to reproduce using his/her avatar. This feature is particularly meaningful when the third person perspective is selected since all the user's avatar bodies are visible.

There is also another advantage. In some circumstances the therapist might want to interact with a patient training session, for example to help him/her performing a specific task or to show which might be the correct movement and so on.

#### **4.6 User Interaction**

User interaction in the case of patients with upper-limbs disabilities is a fundamental concern when designing the application.

Originally, the therapist was meant to activate the system and assist the patient during the whole session. The therapist is represented by his/her own avatar in the virtual scene, different from the one of the patient. While the application is started from the master node of CAVE2, the Wand controller can be used for opening a menu panel. The panel is implemented as a virtual 3D user interface. This menu appears in the scene to the left of the avatar. The menu allows the user and the therapist to select which training mode to start (or restart), or to exit

the application. Selections can be made by using specific buttons on the controller or by moving the avatar so that it touches with its hand the virtual interface.

However, we found this approach inefficient, as the CAVE2 environment renders the 3D perspective of the patient, and not the one of the therapist. Nevertheless, if we want to allow the patient to modify the training at run time, we have to consider that it is not always feasible for his/her to use a manual controller. To alleviate this problem, we added speech recognition with predefined words, that allow users to perform the functions that could have already been enabled with the Wand.

For example, a patient can use his/her voice to change the training mode, restart it, close a menu, show the help panel, visualize the optimal trajectory or activate the distorted mode. Overall, since the different types of interaction do not interfere with each other, we let the user decide which interaction type he/she prefers. Animations, sound and visual effects are used in order to involve the patient and provide feedback.

## **4.7 Implementation Details**

### **4.7.1 Scene Description**

We now review the most important gameobjects and scripts contained in the main scene of the game.

- CAVE2Manager: this gameobject is essential to develop Unity applications for CAVE2 and it is publicly available. It handles the communication between getReal3D and Unity and between the CAVE2 tracking system and Unity.
- OmicronPlayerController: this gameobject has several children. The most important are:

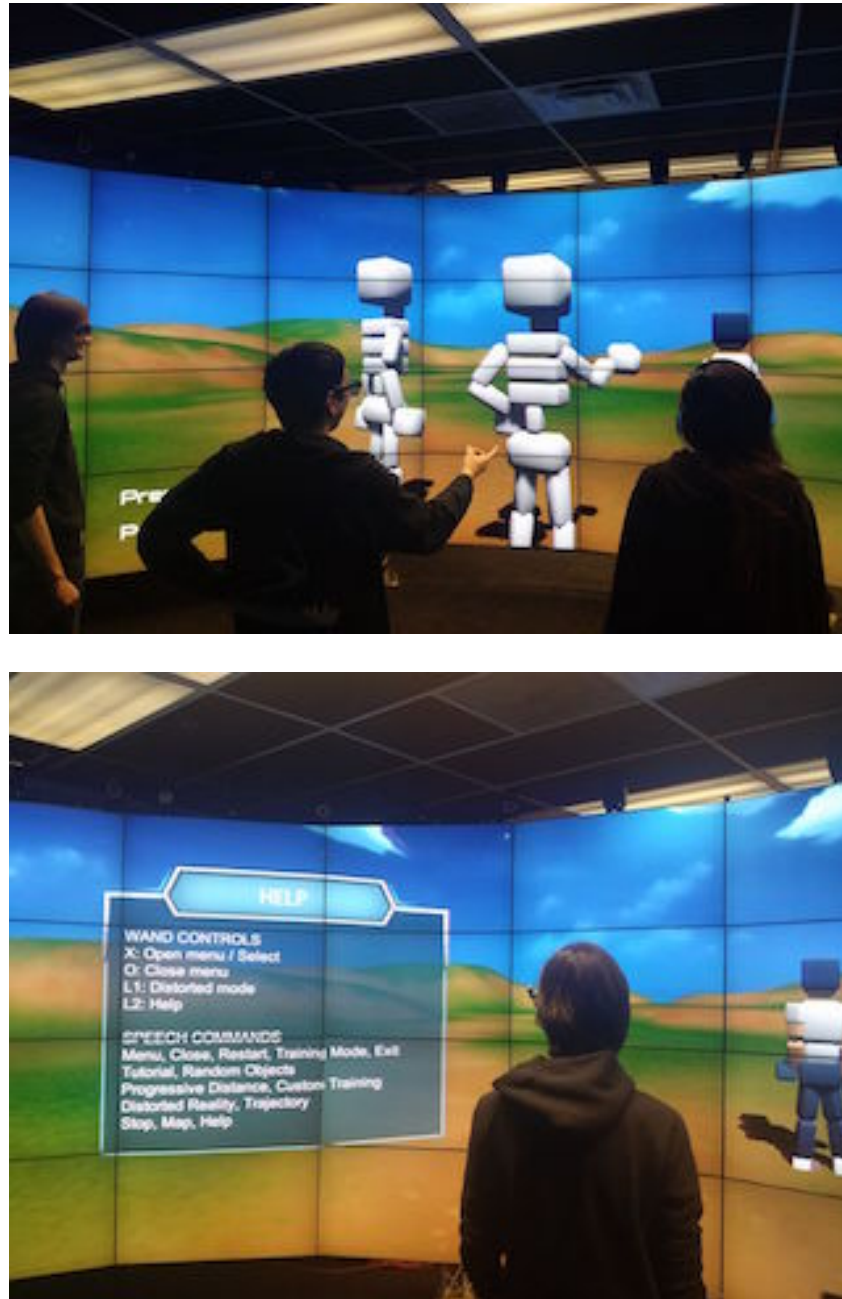


Figure 6: Patient-therapist interaction: the top image shows the patient and two therapists. Help menu: the image on the bottom shows the user looking at the help functionality.

- CameraController: the camera controller scripts are in charge of updating the camera according to the position of the tracking glasses. It also computes the correct orientation of the camera based on the cluster node running this copy of the game, because each computer in the cluster renders a different portion of the screen.
- KinectManager: this gameobject contains the script that is in charge of handling updates received from the Kinect through the Omicron Server. These updates consist of objects containing several attributes, such as the type of the event (e.g. position, voice), the spatial coordinates of the event or the recognized word. After an update is received the scripts decides whether to generate a new body, destroy one particular body or to send an update to an existing one.
- AvatarController: this is the script associated with each body. It is in charge of applying to the corresponding the body the update received through the KinectManager. It applies to each joint the position received measured by the Kinect plus, in case the game is using the first person perspective, an offset on the z-axis.
- SessionManager: this gameobject contains one of the most important scripts of the game. The SessionManager is in charged of handling the various phases of the game and it keeps track of the state of the game. Reasoning in terms of the MVC design pattern the SessionManager can be considered the controller.
- VoiceManager: gameobject that contains scripts to handle voice messages and audio effects. When the SessionManager updates the state of the game it sends an event to

the VoiceManager, specifying which updates has triggered the event. The VoiceManager knows how to interpret the event and it reproduces the appropriate sound track.

- GUI: as the name suggests this gameobject groups all the elements and scripts used to display the user interface in the virtual world as well as messages for the user of the application (e.g. statistics about his/her performances). All the methods necessary to update the GUI are called by the SessionManager each time that the state of the game is modified.

#### **4.7.2 Joint Position and Orientation**

Data received from the Kinect, actually from a Server directly connected to the Kinect, consists of an object containing the ID of the body tracked, the ID of the joint tracked and the position in space of the joint. Therefore each avatar in the scene is a different object with a different ID corresponding to the body ID sent by the Kinect and each time an update is received the correct avatar object is updated.

Unfortunately we do not have access to the joint orientation because the software handling the kinect on the server side was originally developed for the Kinect v1 which doesn't provide data about joints rotations, and then reused for the Kinect v2 which instead provides rotational data. However to make the avatar move in a realistic way we cannot ignore the orientation of the joints. Since we know the original relative rotation that exists among each joint in the original avatar model we use this rotation to compute the correct orientation after each movement using a lookAt function provided in the Unity environment.

### 4.7.3 Patient Identification

As we explained in the previous section, an ID is associated with each body tracked by the Kinect. When a new ID is received a new body object is instantiated and stored in an array containing all the active bodies and an avatar is added to the scene. The first body tracked is assumed to be the patient and all the subsequent ones are considered to be therapists. If no updates about a certain body ID are received for a certain amount of time the corresponding avatar is deleted from the scene and the object associated is removed from the array containing the current tracked bodies. If the avatar destroyed is the one representing the patient, one of the bodies currently tracked is associated with the patient's avatar. To do that we decided to identify the patient as the body which is nearest to the Kinect at that moment.

### 4.7.4 User Interface and Interaction

All the user interface components are realized using the Unity UI Elements. However they cannot be used in screen coordinates, but they must be placed in the Unity world coordinate system because otherwise they would have been reproduced on every screen.

As we have already discussed there are two possible ways to interact with the application user interface: the Wand controller and the Kinect Speech Recognition. We now discuss how these two different approaches are handled inside the application.

- Wand Controller: when a wand button is pressed this information is communicated through the getReal3D plugin to the CaveManager, a gameobject provided in the standard Unity set up for CAVE2. Unfortunately this does not generate a specific event, and the only solution to check whether a button has been pressed is to iterate every frame

over the possible buttons and check whether they are currently pressed or not. For this reason we implemented a menu manager which is in charge of handling wand actions. Using the observer pattern, all the menu manager listeners are informed when a wand button has been pressed in order to update correctly the state of the game.

- **Speech Recognition:** an XML file containing all the recognized words must be provided to the Kinect Software. Basically this file is a list of rules consisting in a set of key value pairs. The value is a list of words which trigger this specific rule, while the key is a string that is sent when the rule is triggered. Speech data is sent to the unity game using the Omicron Server in the same way as tracking data.

Speech data is received in Unity through the omicron server as an object containing the recognized string and the confidence of the recognition. If the confidence is above a minimum threshold the string is parsed and the game state is updated according to the command just executed.

#### **4.7.5 Perspective**

The virtual camera rendering the scene is located on the head of the patient, who is wearing a special pair of 3D glasses equipped with markers that allow the CAVE2 tracking system to locate their position and rotation in space. This position is sent to the Unity Game and it is accessible through the CaveManager gameobject, which we described in the previous section.

In order to achieve the third person perspective an offset on the z-axis is added to the position measured by the Kinect so that the full avatar body is visible. On the other hand for the first person perspective the original positions are used.



Figure 7: Visual reward when the user reaches the object.



## 4.8 Data Gathering

During each training session, the system records the position and rotation in space of each joint with a frequency of 30Hz. Moreover, additional information about the objects and the time in which they were reached are added to a log file, stored on the master node of the CAVE2. This information is then automatically sent to the laboratory, where data relative to a specific patient can be further analyzed. A different log for each mode is stored, even if incomplete, with basic metadata characteristics for the patient. The time spent by the patient in reaching a single target is saved and used as a performance measure, but other types of error measurements can be inferred and used from the recorded data (e.g., the relative angle to the optimal trajectory).

## CHAPTER 5

### CAVECHEF

In this chapter we present our second application: CAVEChef. Section 5.1 introduces the game highlighting some of the main differences with RehabJim. Then in section 5.2 we explain the principles that guided us when designing this application. Section 5.3 describes how we track users' bodies and why we decided to change the tracking device. Section 5.4 presents the game, explaining its structure and describing the underlying story. Section 5.5 points out some implementation details, concentrating on how we mapped the user's body to the avatar's body. In section 5.6 we list the possible interaction methods offered to users to interact with the user interface of the application and also how users can interact with objects inside the scene. Section 5.7 explains how we tried to create an highly motivational environment which is one of the key aspect to create a successful Serious Game for Virtual Rehabilitation.

#### **5.1 System Overview**

CAVEChef is a Serious Game for lower limb rehabilitation in post stroke patients using a third person perspective, developed for CAVE2. Most of features of this application are similar to RehabJim (Chapter 4). However there are some differences. In CAVEChef only the patient's avatar is represented in the virtual world and his/her movements are tracked using the motion capture system of the CAVE2 environment (See section 3.2).

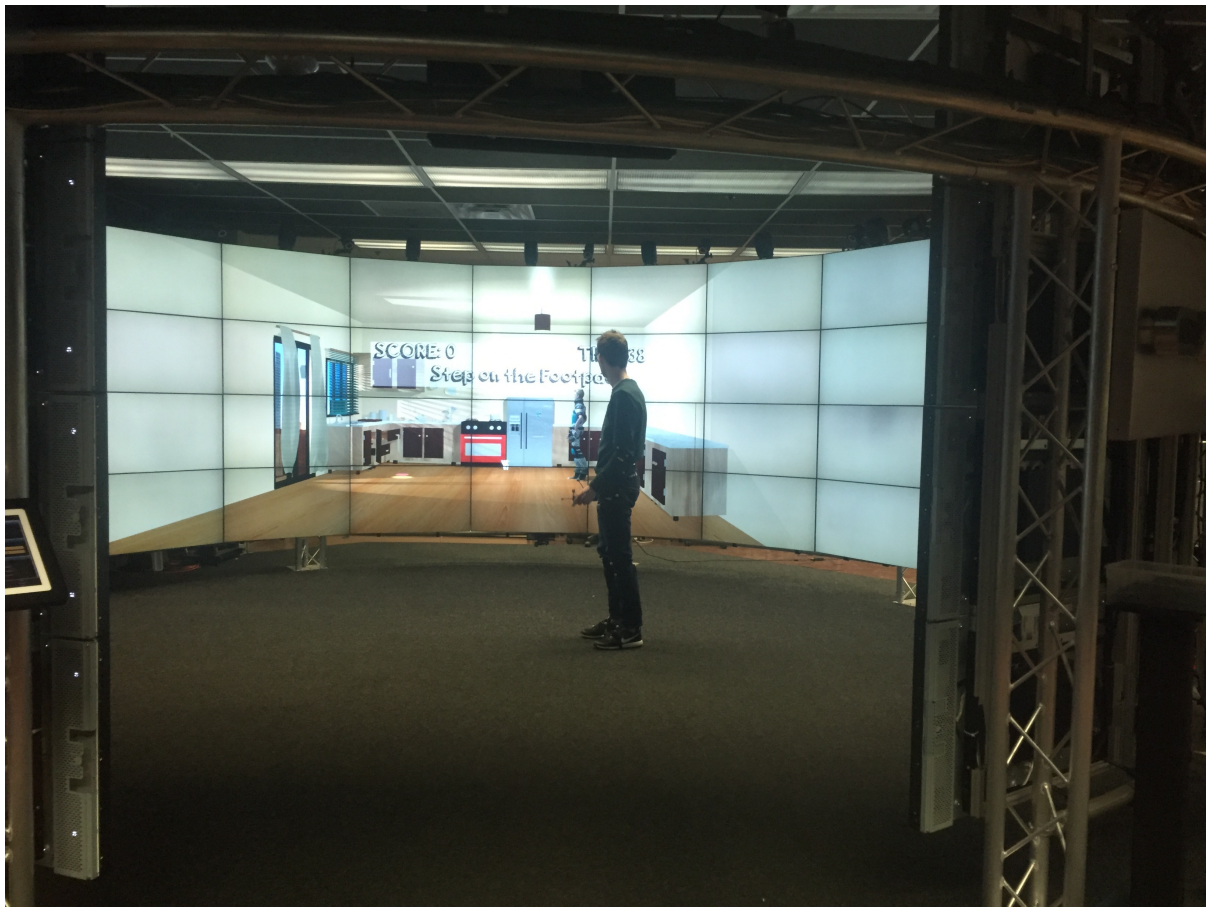


Figure 8: CAVEChef, overview of the game.

## 5.2 Application Design

A possible strategy to create appropriate Serious Games for rehabilitation purposes is using minimalistic environments. However this is not the only possible solution: researchers and literature suggest that reproducing environments that are used on a daily basis can be an effective way to involve patients in the rehabilitation process. For this reason our virtual environment is a standard kitchen and thanks to the possibilities offered by CAVE2 our users are really navigating in a 3D kitchen. Our kitchen is a simple model ( Figure 10 top) with a neutral color palette.

The design challenges of this game are similar to the ones faced in the previous game. First of all the model of the kitchen is adapted such that all its component are easily reachable by the user. Secondly, even if depth perception is far better in virtual reality than with 2D or non-tracked 3D displays, it is rather complex to provide an accurate depth perception for users. To do that we have positioned two windows and one lamp in strategic positions so that users can clearly understand the position of the objects and especially the relative position of the avatar respect to a specific object. Lights and shadows are also useful to enhance the perception of the shape of each object and they also create a much nicer scene.

## 5.3 Tracking

After an initial testing phase we decided to adopt the CAVE2 tracking system instead of the Microsoft Kinect V2, because the tracking range of the Kinect is not enough for the purpose of our application. We have explained in chapter 2 how the CAVE2 tracking system works. To track an entire body users need to wear 9 small plastic plates with 3 markers each ( Figure 9

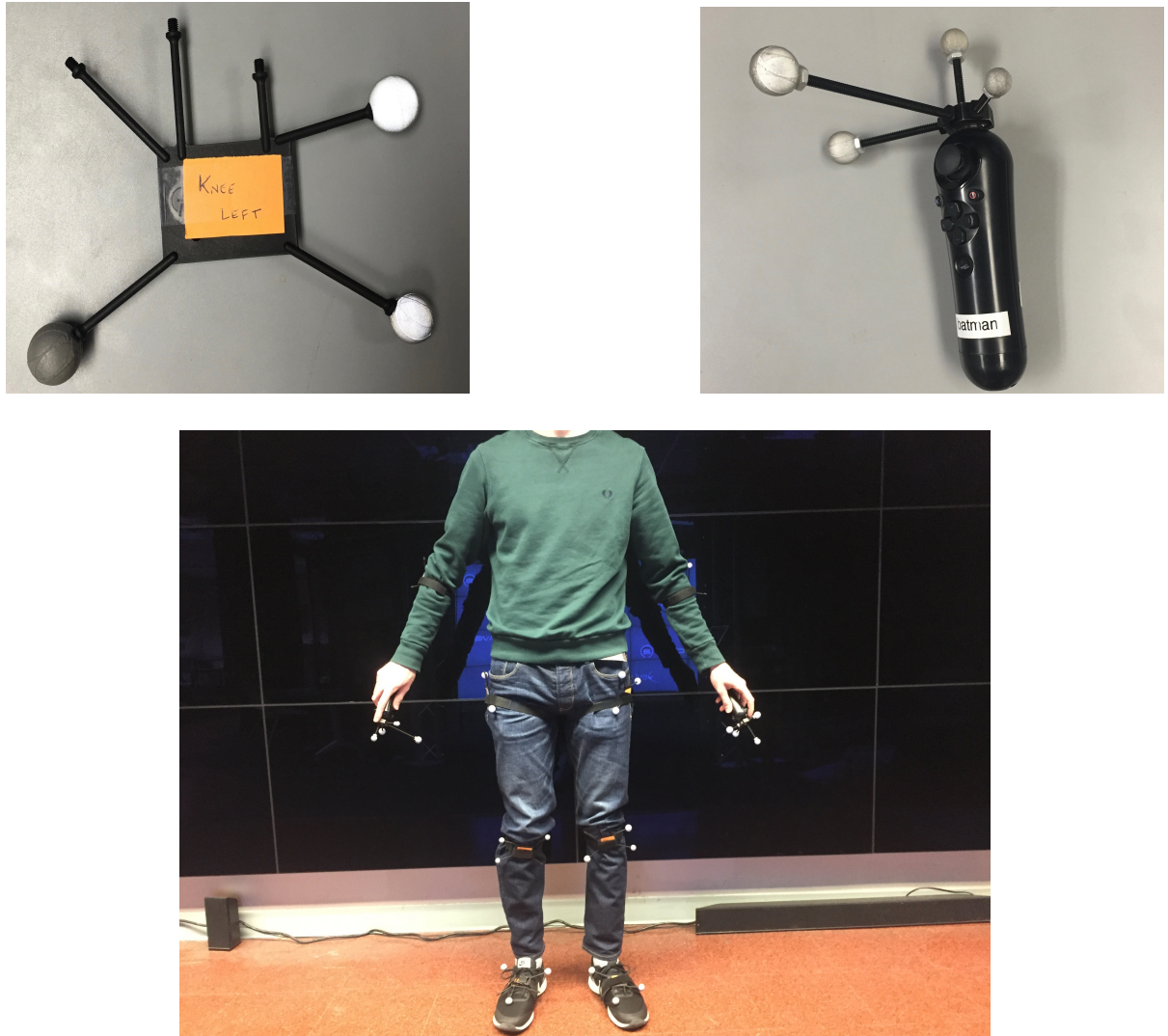


Figure 9: Plastic plate: top left figure shows a plastic plate with three markers to track users' joints. In particular that one is for the left knee. Wand controller: top right image shows the wand controller with four markers to track users' hand. User wearing all the markers: bottom figure shows an user wearing all the necessary markers.

top left) and two Playstation Wand Controllers with four markers each( Figure 9 top right) Plastic plates are placed one per foot, one per knee, one per hip and one per elbow. Each wand is used to track a different hand and to interact with the application. They are attached to the body using simple sport bands. The markers allow tracking the position and the rotation in space of the joints to which they are attached.

All the plates have the same shape, but the markers are positioned with different configurations so that the relative angles among them is different. This allows the tracking software to recognize the different plates and treat them as different objects in space. This software runs on a machine dedicated to handling the tracking system. This machine send to its clients the data measured. On the client machine getReal3D translates the data into objects that Unity3D can use: each position is a vector of 3 elements representing the x y and z coordinates in space and each rotation is a quaternion.

## **5.4    Game Description**

### **5.4.1    Game Topic**

A Chef coming from the future is trapped in a present day kitchen. The only way to get back to his time is to cook delicious meals before it is too late. However this kitchen seems to be cursed. Pans are falling from the roof, and suddenly obstacles appear on the ground to make our chef work even harder.

### **5.4.2    Tasks and Obstacles**

The game is composed of levels of increasing difficulty for the user. Each level consists of some tasks that need to be performed in a limited amount of time. The overall objective of the





Figure 10: Top image: A simple kitchen with light and shadows to improve the depth perception. Bottom image: The avatar inside the kitchen mimic users' movements.

tasks is usually preparing a meal, but they can be decomposed in simpler subtasks that must be performed in a specified sequence to achieve the desired final results. To limit the amount of cognitive load for users and allow them to concentrate on performing it we use some UI elements (e.g. text) specifying what the goal of the current subtask is. Since we are not using a treadmill, we need to design tasks in a smart way in order to force the user to move from one side of CAVE2 to another repetitively.

Obstacles play an important role in our game. The number of obstacles is dynamically tuned by the system according to the patient performances. There are two different types of obstacles which fulfill two different functions:

- Obstacles on the ground: they are useful to force the user to move on specific paths or perform some specific movements. The therapist, before the game starts, can tune some parameters to generate obstacles in specific positions (e.g. obstacles only on the side of the left leg).
- Obstacles falling from the roof: since one of the objective of this game is performing rehabilitation using a third person perspective is important that users look always at their virtual avatar and at the virtual world. However while patients are moving around might be inclined to look at their feet. To avoid this after several meeting with domain scientists we decided to introduce obstacles moving using some simple AI which try to hit the avatar of the patient making him/her losing points. We consider this an effective way to keep the patient's attention on the virtual world.



To make the experience more enjoyable each level is enriched with some mini-games, for example in one of these the patient uses his/her virtual avatar to kick some boxes on the ground and put them in a cabinet. Sometimes these mini-games does not contribute directly to the rehabilitation purpose, but they are useful to create a better experience which indirectly can influence the rehabilitation results.

We now briefly describe one of the levels of the game to highlight its structure and to explain how the different components of the game are mixed together. All the other levels follow the same underlying structure with some minor variations.

#### **5.4.2.1 Level One Description**

The "Cooking a Pasta" level can be decomposed in two tasks: cooking the sauce and cooking the pasta. The order of the execution of the subtasks is predetermined, meaning that in each phase of the level only one subtask can be executed. Users know which is the current sub-task thanks to the message board displayed on the top of the screen ( Figure 11). Some of the sub-tasks needed to complete the level are shown in ( Figure 11). Pans falling from the roof are used in this level as obstacles.

#### **5.4.3 Scoring System**

We consider the scoring system to be not only an effective way to evaluate the users' performances but also a powerful tool to motivate users to achieve better results. For this reason at the end of the game a ranking showing all of the past scores of the current user is shown. The overall score is determined by three components:

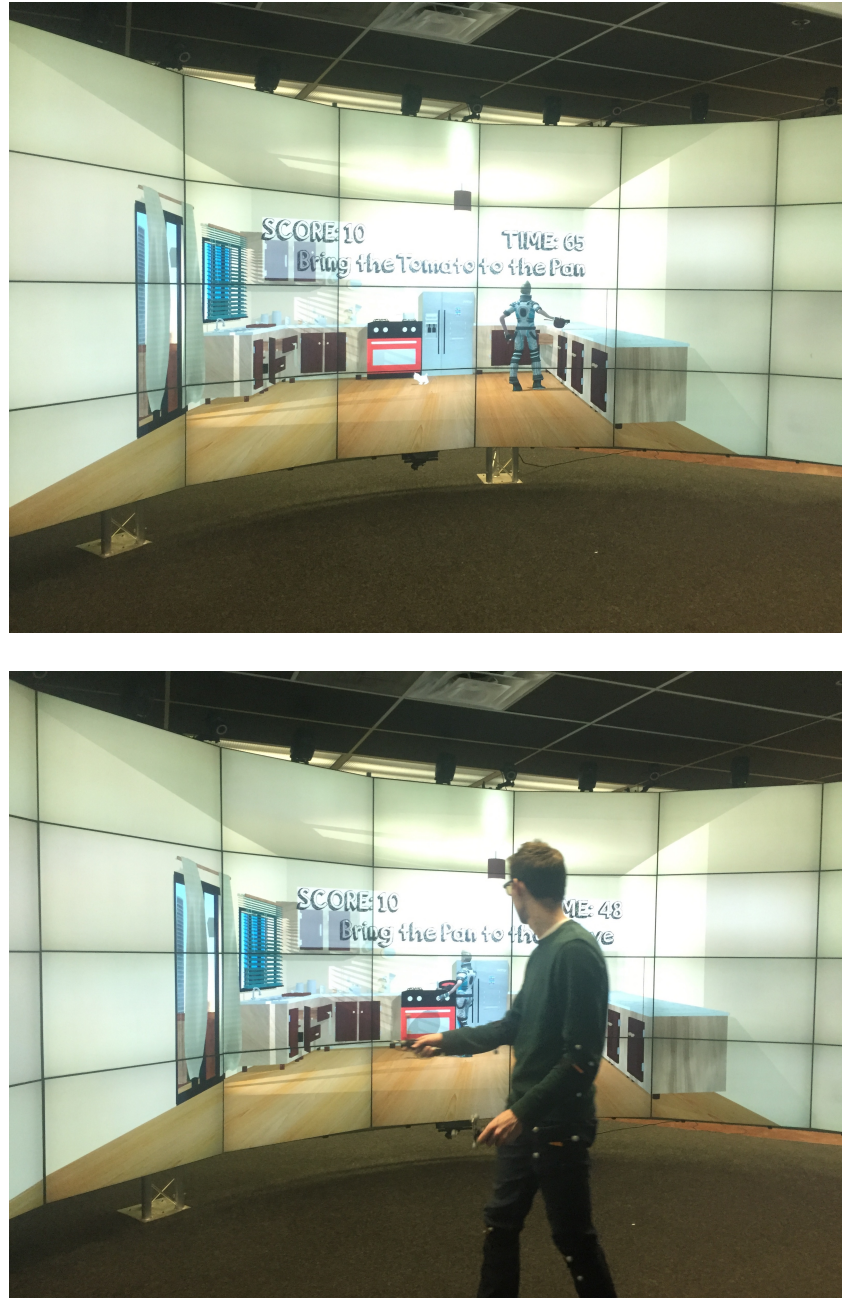


Figure 11: Level One: in the top image the user has to bring the tomato to the pan to complete the subtask. Level one: in the bottom image the user has to move the pan to the stove to prepare the tomato sauce.

- Time to complete the level: if the time required to complete a level is less than the maximum amount of time available for that specific level the score of the user is increased proportionally to the time left.
- Task and subtask success: each time that the user completes successfully a subtask or a task the score is increased according to the complexity of the task.
- Mini-games: mini games offer some bonus points. The amount of points achieved in a mini game is weighted in order to assign more points to mini games related to the rehabilitation objectives and much less points to the others.
- Obstacles hit: each time that an obstacle hits a patient the score is decreased. The penalty assigned is dynamically adjusted based on the user performance.

## **5.5 Implementation Details**

In this section we analyze in detail how the most important features of this game have been implemented. Some aspects are similar to the previous application and therefore we just touch upon them while concentrating on what is different.

### **5.5.1 Joint Position, Orientation and Scale**

On the tracking machine the Vicon Tracker program makes it easy to add new tracked objects. We used this software to add all the plastic plates used to track patients' joints. Then each new object was added to the server configuration file and to the client configuration file.

Some slight modifications to an xml configuration file were necessary to map each getReal3D sensor to the real sensor in order to access data measured by the infra-red cameras directly in

Unity. We can access information relative to each tracked plastic plate using a function that takes as argument the name of a specific plastic plate and returns an object with the position and rotation as properties. The position received can be directly applied to the avatar's joint corresponding to the tracked plate, while the rotation need some further modifications to take into account the orientation of the avatar's model.

Only the right shoulder is tracked. However its position is used only to scale the avatar and not to move any avatar joint. After an initial testing phase we decided to make the shoulders look at their corresponding elbows and move them according to the elbows. We decided to adopt this technique because the final results were better and we are not losing too much information, since shoulders can perform very few movements which are not reflected in elbows movements.

Height and width of patients do not usually match exactly the height and width of the avatar. To deal with this problem we use data received from the trackers to scale the avatar in an appropriate way and create a better visual effect. The distance between the two hips is used to scale the avatar's width while the distance between the right hip and the right shoulder is used to scale the upper-body of the avatar. Arms and legs are scaled using the distance between one joint and its neighbor (e.g. to scale the upper leg, we compute the distance between the right knee and the right hip).

### **5.5.2 Scene Description**

We now describe the main gameobjects and scripts that compose our game focusing specifically on the ones not used in the previous application.

- CAVE2-Manager and OmicronPlayerController: they carry out the same functions as the ones of the previous application (See section 4.7.1). The only difference is that the kinect manager here is only employed for speech recognition.
- BodyPart: this is the class that connects each body part to the sensor and moves it accordingly to the data measured by the infra-red cameras. It also computes the rotational offset that we mentioned above.
- BodyMover: this script is in charge of moving the avatar. It contains an array of body parts and every frame calls a function on each body part that updates the position and rotation of that avatar's joint.
- TriggerBody: This script maps the user body to the avatar body: it computes an offset rotation for each joint and scale each body part to match user's body. The mapping can be easily done by pressing a button on the wand controller or by using the alternative voice command.
- GameController: this gameobject handles the different phases and levels of the game through the script attached to it. The GameController class follows the singleton design pattern.

## **5.6 User Interface and User Interaction**

Users can open a virtual menu by either using a Wand Controller or their voice by leveraging the Microsoft Kinect V2 speech recognition feature. From this menu users can easily see their previous results and restart the game.

Several design decisions have been taken to establish how and when the user can interact with the menu. We decided to pause the game when the user is interacting with the menu in order to not influence his/her statistics. If the user resumes the game he/she needs to be approximately in the same position where he/she was before pausing the game. To make this process easier a red circle is shown in the virtual world indicating the position where the avatar was before pausing the game. As soon as the avatar enters the circle the game is resumed.

To interact with menus users use the Wand controller buttons. On the other hand to interact with objects in the scene users use directly their feet and the two Wand controllers that they keep in their hands. However in this case the Wand controllers are used as a simple tracking device, they track the movements of the two hands. There is no need to press any button to grab objects, as soon as the wand, i.e. the avatar's hand in the virtual world, is near enough to the virtual object, the object is attached to the avatar's hand. The object can be detached from the hand by putting it in its original position or in its target position. We decided to use wands to track hands and grab objects because, since we cannot track fingers, it was rather difficult to point precisely to objects with bare hands. With wands instead it is much easier to grab objects. This solution improves the overall user experience without sacrificing too much accuracy for hand movements.

Visual aids, such as arrows or scaling effects, are employed to highlight the objects users can interact with. For example if the current task involves grabbing a tomato, this object is re-sized at regular intervals to make clear that it is the target of the current task.

## 5.7 Highly Motivational Environment

To create a highly motivational environment we believe that sound effects play an important role especially because the CAVE2 environment offers an amazing audio system which increase that feeling of immersion inside the game that we consider one of the key aspects for an effective Serious Game. We have therefore added to the game a background music and other sound effects. However sounds can be easily disabled by patients who prefer playing in a quiet environment. Here is a list of the sound effects used:

- Feedback sound: when a user completes a sub-task a short audio track is reproduced to make clear the result of his/her actions. Two different tracks are used, so that it is clear when a sub-task has been completed successfully and when it has not.
- Winning sound: a joyful sound is reproduced when a user completes a level. This sound can be considered a sort of reward for completing the level and therefore acts as a motivational component.
- Foot steps sound: reproducing a sound each time that one of the avatar's feet touches the ground is an effective way to make the user feels inside the game. It is also a way of providing feedback that this action was completed.
- Time sound: users may be concentrated on the game and may not realize that the time is about to end. So there is a timer sound which starts a few seconds before the time to complete the level ends.

## CHAPTER 6

### USER STUDY AND EVALUATION

#### 6.1 User Study Description and Motivation

In this user study we tested our first application: RehabJim.

This study helped us understand what are the strengths of using Serious Games for Rehabilitation in a Large-Scale virtual reality environment and, at the same time, which techniques can help create a better immersive experience and whether it is easier to use a first person perspective or a third person perspective.

The purpose of the study is to acquire design insights to create a better experience for patients. However we will not test the application on real patients; we believe that we can gain useful insights by testing it on people without disabilities. The user study observes and compares the results of performing different tasks, observing one user at a time. Users will go through the task of completing a tutorial, using the interactive menu and completing different rehabilitation training modes, described in detail later, using both a first person perspective and a third person perspective.

A facilitator was in the room during the execution of the experiment to answer any questions regarding the application. Video and audio recording were used during the sessions to capture the subjects interaction with the tool and with the virtual environment.



The time to complete each session was automatically recorded by the system and was used as a metric to evaluate the effectiveness of the application. At the end of the session each user was asked to complete a questionnaire to gain more knowledge about their experience with the application. Another metric used to evaluate the application and compare the two perspectives is the number of mistakes committed when trying to reach an object.

Each session can be subdivided into two parts. In the first part users performed the tutorial provided by the application and went through some simple tasks in order to allow them to familiarize themselves with the environment and the application. The second part consisted of performing several rehabilitation sessions and interacting with the user interface of the application. Since the second part is the core of our user study we now describe in detail all the tasks that the users were asked to perform.

- The subject was asked to complete the first rehabilitation training session. The training mode selected was random objects and only the subject was inside the game. The total number of objects generated was 30. The subjects were asked to complete the training session using both perspectives. Only the subjects were inside CAVE2 while the facilitator was just outside, ready to help them in case they need advice.
- In the second part of the study the subjects were asked to interact with the user interface of the application using the Wand Controller.
- In the third part of the study subjects were asked to perform a rehabilitation training session using the random objects training mode with the distorted mode enabled, i.e. the movement of the left and right limbs of the subjects were exchanged in the avatar. The

maximum number of objects that users could catch was 30. Subjects completed this third task using both the first person perspective and the third person perspective.

- Then subjects were asked to interact with the user interface of the application using the speech recognition feature. The subjects were asked to complete the same tasks performed while using the Wand controller.
- In the last part of the study, subjects were asked to complete a rehabilitation training session where objects are randomly generated in the virtual world and visual aids representing the best possible path to reach the object are added to the scene. The total number of objects was 30. The task was completed using both a third person perspective and a first person perspective.

We decided to always use the random objects training mode in order to compare the user experience using the third and the first person perspective in the most objective way. If the objects were generated in specific positions the results of our study would have been influenced by the order in which the two perspectives were used.

To compare the two perspectives we combined quantitative measurements with the user comments about their experience. Quantitative measurements consisted of time spent to reach each object, number of errors committed while trying to reach an object, e.g. the user misses the object and needs to try again to catch it, and number of objects caught. The same metrics were combined to compare the two possible way to interact with the user interface. We measured the time spent trying to perform the task, the number of errors committed to perform that task, e.g. if the task consisted of starting a training mode an error is starting the wrong one.

## **6.2    User Study Settings**

### **6.2.1    Demographic Data**

Before discussing the results of our study it is important to give some demographic data about our subjects. As already said our ten users are not real patients. Nine subjects are male while only one subject is a female. Although there weren't any particular constraint about subjects' age, except being older than 18 years old, all of them range from 20 years old and 30 years old. All of them are student and use computer on a daily basis. 5 of them had never used CAVE2 before the study, 3 of them had only used CAVE2 very few times, and 2 of them had a good experience with CAVE2: they used it several times and developed some applications for it.

Nine subjects declared to have normal stereoscopic vision, while one declared to have corrected stereoscopic vision.

### **6.2.2    Technical settings**

Subjects stayed approximately 8 ft (250 cm) far from the screen and they were asked to maintain this position during the training.

Before starting the rehabilitation training session subjects had the possibility to try the application for about ten minutes. In this phase the principal investigator helped them familiarize with the application and in particular with the two perspectives. The two perspectives were shown to the subjects with the following criteria: if subject one was asked to try first person perspective first and third person perspective second, subject two was asked to try third person

perspective first and first person perspective second and so on. Similarly, the alternation of first and third person perspective in the training session followed the same criteria.

### **6.2.3 Assumptions**

There are two main conditions under which our experiment was performed that are worth mentioning:

- Unity3D is not designed for cluster based rendering. Therefore, even if Unity can be used in CAVE2, worlds generated with Unity for CAVE2 are not scientifically accurate because it has not been calibrated.
- We used a default eye separation for all subjects, therefore we did not set the stereo generated by the application based on the actual eye separation of each individual.

## **6.3 Results**

In this section we analyze the results of the user study. Firstly we compare the number of objects caught in first person and in third person, then the total completion time and the mistakes committed while trying to reach the objects. After that we discuss an interesting aspects discovered in the trajectory training mode and finally we compare the speech recognition with the Wand Controller.

### **6.3.1 Objects Caught**

The first metric used to compare first person and third person is the number of objects caught using one perspective or the other one. To compare the results we computed the arithmetic mean of the number of objects caught in the three different training modes. For example

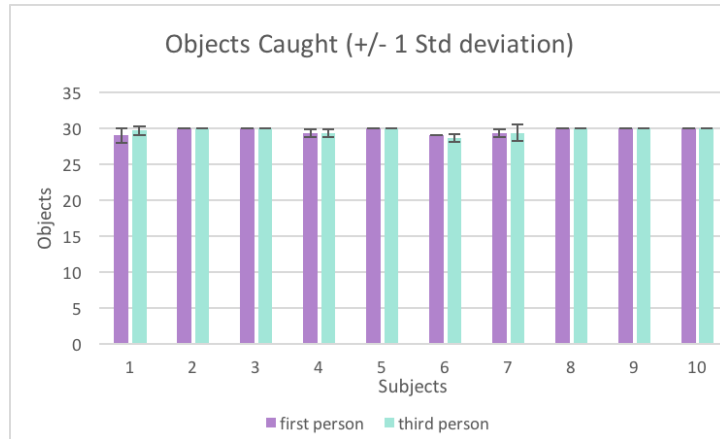


Figure 12: Comparison of number of objects caught when using a first person perspective and a third person perspective. It is represented the average number of objects caught by each subject in the three sessions.

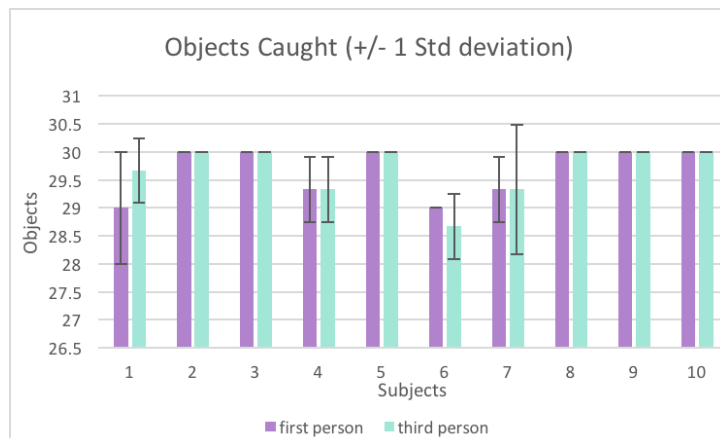


Figure 13: Comparison of number of objects caught when using a first person perspective and a third person perspective. It is represented the average number of objects caught by each subject in the three sessions. This picture highlights the small difference between the two perspectives.

if an user catches 30 objects in the first session, 29 in the second session, and 30 in the third session using a third person perspective the mean would be 29.6.

Figure 13 illustrates the results. As we can see there is no significant difference between the two perspectives in term of objects caught. Most of the users caught all the objects and there is a difference in average of at most one object between the two perspectives. Therefore according to this metric there is not a particular difference between first person and third person in term of task success.

### **6.3.2 Completion Time**

More significant results can be noticed when comparing the completion time of the training sessions. As we can notice from Figure 17, third person perspective is almost always faster than first person perspective. However there are two exceptions. First person perspective is faster for subject 7 in random object training mode ( Figure 14) and for subject 6 in random object training mode with trajectory enabled ( Figure 16). Thanks to the video recording we discovered that in those two particular training sessions, in third person perspective, at least one object was occluded by the avatar body. For this reason, the two subjects spent an additional amount of time to locate the objects that initially were not visible and they had to move their bodies in order to make them visible.

We believe that the main reason, which makes third person perspective faster, is related to human vision and in particular to human field of view. In third person users see themselves from behind, therefore they can easily see the entire range in which objects are generated, since it corresponds to the range of movements of their arms. Also in first person objects are



Figure 14: Time for completing the random objects training session.

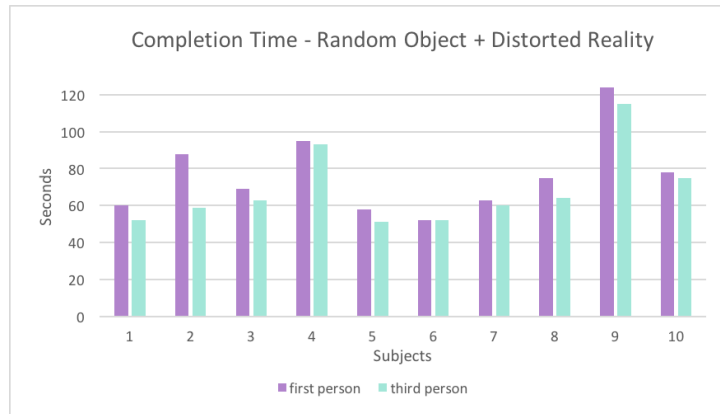


Figure 15: Time for completing the random objects training session in distorted reality.

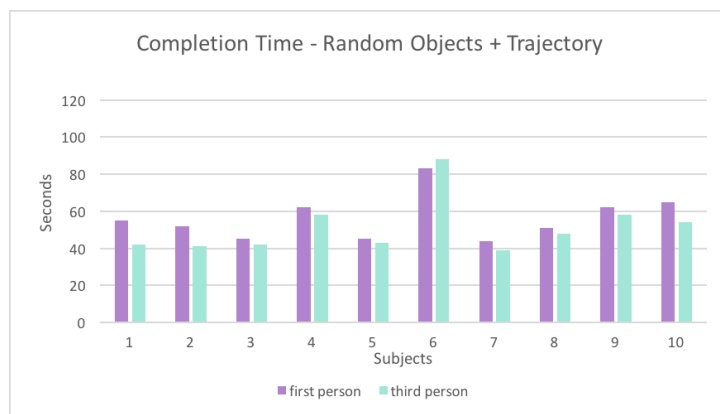


Figure 16: Time for completing the random objects training session with trajectory enabled.

generated in the range of movements of users' arms, however now users cannot see the entire range at once but they need to move their head. As in real life if we are looking right in front of us we can see what is at 90 degrees to the left or 90 degrees to the right. This is due to the fact that humans' binocular vision, which is in charge of depth perception, covers almost 110 degrees, while the remaining 70 degrees are seen by only one eye. Therefore to understand the position of the object in first person user needs to turn their head adding a certain overhead to each action. One user adopted a very efficient strategy to deal with this problem. Since he/she could see only about half of the scene, if the object was not present in the portion of the screen he/she was looking at, he/she started moving the other arm before turning his/her head. For example if he/she had just caught an object on the right side of the virtual world and after that no other object was visible in his/her field of view he/she started moving the left arm before turning his/her head and actually localize the new object.

The difference in time completion is particularly evident when users are performing the training session in distorted reality mode. By observing how users behave in distorted reality we believe that this is caused again by the fact that in first person can see only a portion of the virtual scene. While in third person, in distorted reality mode, users see both their virtual arm and their real arm, in first person they can only see one of them adding a difficulty coefficient to the entire task.

Another reason for this result is related to object position detection mistakes and we deal with it in the next section, where we compare errors caused by an erroneous position detection of the objects.



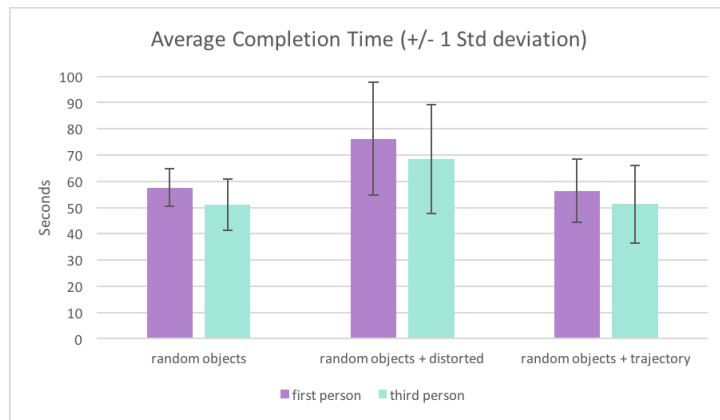


Figure 17: Average completion time of the three different training sessions for each perspective.

### 6.3.3 Adaptation to Distorted Reality

Another interesting aspect that is worth analyzing is how people adapt themselves to distorted reality, when left and right movements are swapped. We noticed two different categories. Both of different approaches followed a classical learning curve: at the beginning a lot of mistakes were committed (e.g. wrong arm moved), then each user found a strategy to deal with the new situation and after that almost no mistakes were committed. We briefly describe the three approaches:

- **Normal Movements:** this strategy was adopted by 70% of the user. After committing some mistakes at the beginning of the session, users adapted to the new situation. They moved only one arm at a time and after the adaptation phase it is usually the right one. They were slower than the second group, because they needed a certain overhead to think which was the correct arm to move.

- Double Movements: 20% of the users adopted this strategy. To improve their performances without increasing the cognitive load they decided to move both arms at the same time. In this way they bypassed the challenge of the task. They are the users with best completion times in distorted reality.

#### **6.3.4 Objects Position Detection Errors**

We analyze here the number of errors when detecting the position of each object. First of all it is important to understand how this statistic was computed. For each object an user can commit zero, one or two mistakes. If the user immediately understood the position and moves toward the object without making any major correction to his/her movement the task was considered successfully completed, i.e. zero mistakes committed. On the other hand if the user made one or more corrections in order to catch the object the task was considered partially successful: one mistake committed. If the user couldn't figure out the exact position of the object and therefore couldn't catch it in time the task was consider unsuccessful: two mistakes committed. To figure out whether a user performed a correction or not during his/her movement toward the object we looked at the video recording of each session and for each object we looked ad the user movement. It was almost always clear whether the trajectory followed by the users was natural or not.

In Figure 18 we computed the average number of mistakes committed by the users for each training session and each perspective. As we can see position detection errors are more numerous in first person perspective than in third person perspective. Even if, this might seem counter intuitive because humans are used to see the world with a first person perspective, given

the fact that worlds generated in Unity for CAVE2 are not currently scientifically accurate and that we used a default eye separation for each subject, we believe that this might be caused by an erroneous depth perception, which is lessened in third person thanks to shadows projected on the floor of the virtual world. This is not possible in first person because CAVE2 does not have imagery on the floor, therefore shadows are not visible. For this reason users usually made several mistakes before understanding the real position of the objects because they found it hard to understand the real distance between their arms and the objects. As we can see in Figure 18 the trajectory training mode helps reducing the number of position detection errors for both perspectives.

Therefore, although shadows are not necessary to understand the correct position of an object in the world, they seem helpful when detecting the position of an object in virtual worlds that might be not scientifically accurate.

### **6.3.5 Trajectory Mode**

An interesting observation arose from analyzing how people use the trajectory information. First of all trajectories are helpful to reduce the number of depth mistakes committed and therefore to reduce the total completion time. They are especially useful to understand the exact position of the object in space. However there are specific cases where users tend to ignore the trajectory and they follow the path that they consider more convenient. This usually happens when, for example, the nearest hand is the right one but the object is to the left of the user's or avatar's body: the majority of users ignore the suggestion and instead of using their right hand to reach the object they use their left hand even if it is much more distant. One

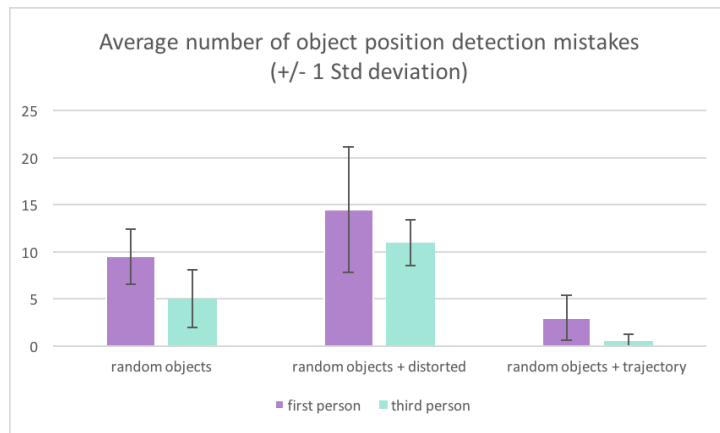


Figure 18: Average number of object position detection mistakes when reaching 30 objects in each different training session and for each perspective.

user followed the path suggested, when objects are on the opposite side of the suggested hand, with a frequency between 67% and 100%, two users followed the suggested trajectory with a frequency between 33% and 66% and seven users followed it with a frequency between 0% and 32%. Concerning this phenomenon there is not any difference between first person and third person. Users that were following the optimal path in first person with a certain frequency were doing it in third person with the same frequency.

### 6.3.6 User Interface Interaction

In this part of the user study users had to interact with the virtual menus of our interface. The actions that they had to perform were:

1. Start Progressive distance training mode
2. Activate distorted reality

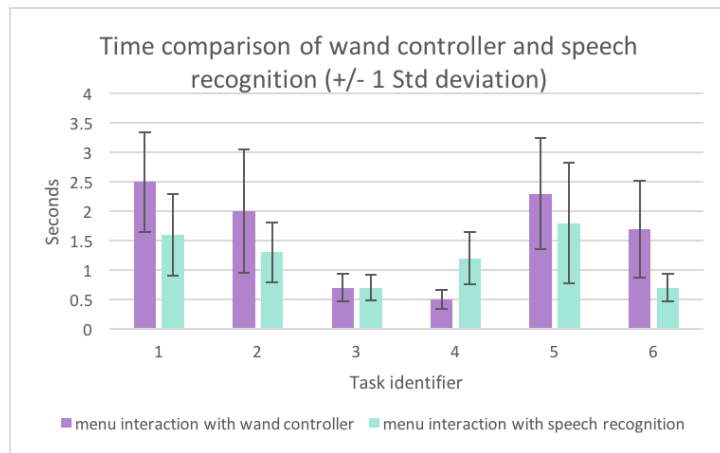


Figure 19: Comparison of the average time spent to complete a task when using wand controller and speech recognition. Only tasks that were successfully completed with both interaction techniques are considered to compute this statistic.

3. Restart the training session
4. Deactivate distorted reality
5. Activate Custom Training training mode
6. Exit from the game

Users were asked to complete these tasks using the Wand Controller and Speech recognition to understand which can be the most effective way for user interaction in environment like CAVE2.

Results show that Wand Controller has success rate of 100%, i.e. each action was performed successfully by users. Instead with speech recognition 30% of users failed in completing at least

one task. This problem is caused by users' accent which influence the speech recognition feature, therefore some non native English users' speakers had trouble in completing the tasks.

One advantage of speech recognition is that it is clear which is the correct command to perform: users just need to say the command with their voice and the system reacts. With Wand Controller users spent some time navigating menus to locate the place where the command was located. Furthermore there is a help functionality which lists all the voice commands and buttons command. The majority of users didn't use the help functionality when using the Wand Controller, while the majority of users used the help functionality when using the speech recognition ( Figure 20). Therefore users tend to figure out how to use the Wand controller by themselves navigating across menus and clicking random buttons, while prefer to use the guide when using speech recognition addressing directly the goal instead of trying random words.

We can conclude with speech recognition is more efficient than the Wand Controller once the users learn the commands, but having a backup controller is necessary because speech recognition is influenced by several factors such as ambient noises, users' accent and so on. This conclusion is validated also by the questionnaire where votes to Wand controller and speech recognition are almost equally distributed.

#### **6.4 Questionnaire**

At the end of the user study users were asked to complete a short questionnaire to give us more insights about their experience when using RehabJim.

Several questions consisted of a Likert scale from one to ten. The distribution of the answers are reported in Figure 21.



Figure 20: Left: Percentage of users that used the help functionality when using the speech recognition. Right: Percentage of users that used the help functionality when using the Wand Controller.

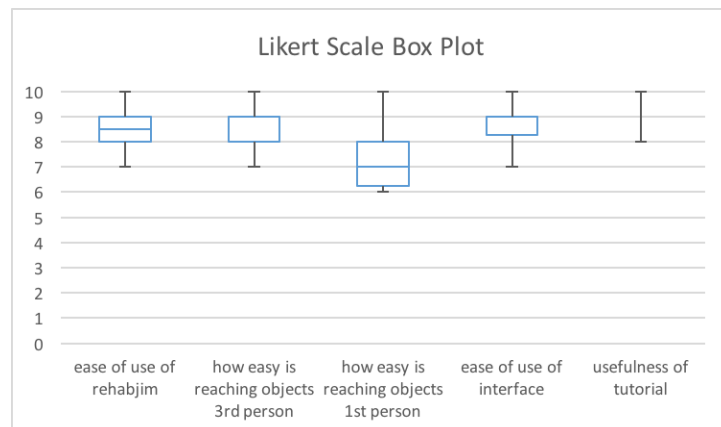


Figure 21: Box-plot of questionnaire answers. User were asked to evaluate the ease of use of several features, where 0 means difficult and 10 means easy. Users were also asked to evaluate the usefulness of the tutorial, where 0 means useless and 10 means useful.

#### 6.4.1 Third Person vs First Person

We can see how users found it easier to perform a rehabilitation training session when using a third person perspective than a first person perspective corroborating what we understood by looking at the data gathered (e.g. time to complete a task). Also a specific question about which perspective was easier to use was asked. As we can see in Figure 22 the majority of users indicated that third person was easier. When asked to explain why they chose third person perspective instead of first person they listed several reasons that can be summarized in the following two:

- Objects are easier to see in third person.
- Easier to understand distance.

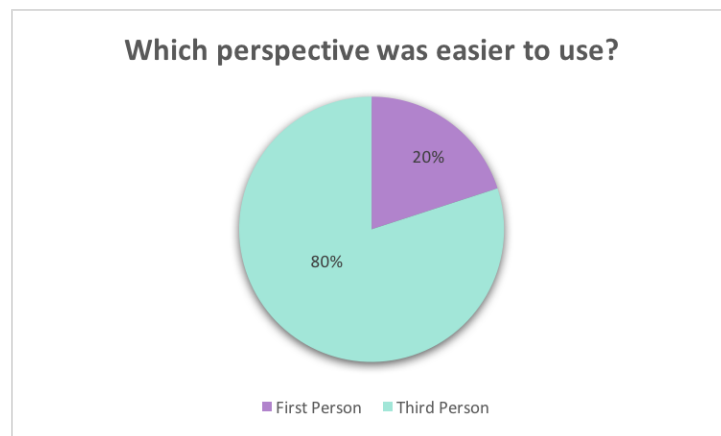


Figure 22: Distribution of questionnaire answers about which perspective was easier to use.



#### **6.4.2    Humanoid Avatar vs Cartoon Avatar**

Users were then asked if, according to them, a humanoid avatar could improve the overall rehabilitation experience. 70% of users indicated no as answer while the remaining 30% suggested that a humanoid avatar could help creating a more realistic and immersive game and also give a better perception of movements in the virtual world.

#### **6.4.3    User Interface**

As we can see from the box plot users found rather easy interact with the user interface and they appreciate its minimalistic features. When asked to choose which interaction mode they prefer, Wand Controller or Speech Recognition, the votes were almost equally distributed with a slightly preference for the Wand Controller. This is likely caused by the fact that Wand Controller always allowed them to successfully complete the tasks while speech recognition sometimes failed because of the reasons explained in Section 6.2.6.

#### **6.4.4    General Comments**

The last questions of the questionnaire asked about general suggestions to improve the overall experience. All users gave precious comments about several aspects of the game. For example one of the users suggested to add external goal and more rewards to increase the motivation of users to reach the object, e.g. adding a story to the game.

Another user suggested to reduce the visual feedback when using the first person perspective. Since objects are nearer to users, some of them complained that the animation, which confirmed that the object was caught, sometimes is too bright for the user's eyes.

All the users enjoyed their experience and give a positive feedback about the application. Some of them were really enthusiast about it and found very engaging interact with their body in virtual reality.

## CHAPTER 7

### CONCLUSION

We presented in this work two Serious Games for virtual rehabilitation RehabJim and CAVEChef. Most of the techniques employed to design the two applications follow the same guidelines but there are some differences. First of all the rehabilitation goal is different: RehabJim is developed to assist patients with arm impairments, while CAVEChef patients with walking impairments. The different rehabilitation forced us to choose a different tracking technology for each application. We used the Microsoft Kinect V2 for RehabJim and the CAVE2 tracking system for CAVEChef in order to increase the tracked area so that users can move freely in a bigger space. Although the two applications try to create a simple environment CAVEChef is set in a kitchen, an environment that is used every day by people. We explained in section 2.2 how researchers believe that introducing daily activities and commonly used environment might increase the adherence to the rehabilitation routine. Therefore we implemented this feature in CAVEChef, building the rehabilitation objective on top of some cooking tasks.

RehabJim was evaluated with a User Study. 10 users were observed while performing several rehabilitation training sessions and interacting with the user interface. The main objective of the user study was comparing the first person perspective with the third person perspective. Although the users tested were not real patients we understood several important aspects. By combining some quantitative measurements together with users' feedback gathered using a questionnaire, our study showed that users' easily understand the position of objects in space

when using a third person perspective while have some problem when using a first person perspective. We believe that this fact is mostly caused by the different portion of virtual space visible in the two perspectives. Another reason might be the presence of shadows on the floor, visible in third person, but not in first person. However, even if shadows surely help to detect the correct object position, they are not a necessary condition.

Other useful insights were obtained in the user study. For example we noticed that people tend to ignore visual aids showing the best path to reach objects, but they follow what they consider the best path. For what concerning the user interaction with RehabJim user interface, we learned that while Wand Controller can be used alone, speech recognition needed a backup controller because it is influenced by external factors such as users' accent or ambient noise.

## **7.1 Future Works**

At the moment users can use CAVEChef only using a third person perspective. Therefore a first extension of this work might be adding to the game also the first person perspective in order to compare the two perspectives also in a walking application and where tasks are not strongly influenced by the perspective.

Also RehabJim could be extended to increase the interaction between therapists and patients. For example by adding a training mode where the therapist can move objects at run time just with his/her hands.

## CITED LITERATURE

1. Holden, M. K.: Virtual environments for motor rehabilitation: review. Cyberpsychology & behavior, 8(3):187–211, 2005.
2. Rizzo, A. A. and Kim, G. J.: A swot analysis of the field of virtual reality rehabilitation and therapy. Presence, 14(2):119–146, 2005.
3. Mouaheb, H., Fahli, A., Moussetad, M., and Eljamali, S.: The serious game: what educational benefits? Procedia-Social and Behavioral Sciences, 46:5502–5508, 2012.
4. Stapleton, A. J.: Serious games: Serious opportunities. In Australian Game Developers Conference, Academic Summit, Melbourne, 2004.
5. Charsky, D.: From edutainment to serious games: A change in the use of game characteristics. Games and culture, 2010.
6. Playstation Navigation Controller description. <http://us.playstation.com/ps3/accessories/playstation-move-navigation-controller-ps3.html>. Accessed: 2016-04-18.
7. Hoermann, S., Santos, L., Morkisch, N., Jettkowski, K., Sillis, M., Cutfield, N., Schmidt, H., Hale, L., Kruger, J., Regenbrecht, H., et al.: Computerized mirror therapy with augmented reflection technology for stroke rehabilitation: A feasibility study in a rehabilitation center. In Virtual Rehabilitation Proceedings (ICVR), 2015 International Conference on, pages 199–206. IEEE, 2015.
8. Patel, J., Fluet, G., Merians, A., Qiu, Q., Yarossi, M., Adamovich, S., Tunik, E., and Massood, S.: Virtual reality-augmented rehabilitation in the acute phase post-stroke for individuals with flaccid upper extremities: A feasibility study. In Virtual Rehabilitation Proceedings (ICVR), 2015 International Conference on, pages 215–223. IEEE, 2015.
9. Klein, A. and De Assis, G. A.: A markerless augmented reality tracking for enhancing the user interaction during virtual rehabilitation. In Virtual and Augmented Reality (SVR), 2013 XV Symposium on, pages 117–124. IEEE, 2013.

### CITED LITERATURE (continued)

10. Ma, M. and Bechkoum, K.: Serious games for movement therapy after stroke. In Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on, pages 1872–1877. IEEE, 2008.
11. Burke, J. W., McNeill, M., Charles, D. K., Morrow, P. J., Crosbie, J. H., and McDonough, S. M.: Optimising engagement for stroke rehabilitation using serious games. The Visual Computer, 25(12):1085–1099, 2009.
12. Mumford, N., Duckworth, J., Thomas, P. R., Shum, D., Williams, G., and Wilson, P. H.: Upper limb virtual rehabilitation for traumatic brain injury: Initial evaluation of the elements system. Brain injury, 24(5):780–791, 2010.
13. Mumford, N., Duckworth, J., Thomas, P. R., Shum, D., Williams, G., and Wilson, P. H.: Upper-limb virtual rehabilitation for traumatic brain injury: A preliminary within-group evaluation of the elements system. Brain injury, 26(2):166–176, 2012.
14. Turolla, A., Dam, M., Ventura, L., Tonin, P., Agostini, M., Zucconi, C., Kiper, P., Cagnin, A., and Piron, L.: Virtual reality for the rehabilitation of the upper limb motor function after stroke: a prospective controlled trial. Journal of neuroengineering and rehabilitation, 10(1):1, 2013.
15. Cameirão, M. S., Bermúdez, I. B. S., Duarte Oller, E., and Verschure, P. F.: The rehabilitation gaming system: a review. Stud Health Technol Inform, 145(6), 2009.
16. Adams, R. J., Lichter, M. D., Krepkovich, E. T., Ellington, A., White, M., and Diamond, P. T.: Assessing upper extremity motor function in practice of virtual activities of daily living. Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 23(2):287–296, 2015.
17. Microsoft Kinect Sensor description. <https://msdn.microsoft.com/en-us/library/hh438998.aspx>. Accessed: 2016-04-18.
18. Sucar, L. E., Orihuela-Espina, F., Velazquez, R. L., Reinkensmeyer, D. J., Leder, R., and Hernandez-Franco, J.: Gesture therapy: An upper limb virtual reality-based motor rehabilitation platform. Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 22(3):634–643, 2014.
19. Jacobs, A., Timmermans, A., Michielsen, M., Vander Plaetse, M., and Markopoulos, P.: Contrast: gamification of arm-hand training for stroke survivors. In

## CITED LITERATURE (continued)

- CHI'13 Extended Abstracts on Human Factors in Computing Systems, pages 415–420. ACM, 2013.
20. Fikar, P., Schoenauer, C., and Kaufmann, H.: The sorcerer's apprentice a serious game aiding rehabilitation in the context of subacromial impingement syndrome. In Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2013 7th International Conference on, pages 327–330. IEEE, 2013.
  21. Boian, R. F., Deutsch, J. E., Lee, C. S., Burdea, G. C., and Lewis, J.: Haptic effects for virtual reality-based post-stroke rehabilitation. In Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2003. HAPTICS 2003. Proceedings. 11th Symposium on, pages 247–253. IEEE, 2003.
  22. Boian, R., Kourtev, H., Erickson, K., Deutsch, J., Lewis, J., and Burdea, G.: Dual stewart-platform gait rehabilitation system for individuals post-stroke. In Proceedings of the Second International Workshop on Virtual Rehab, volume 93, 2003.
  23. Roy, A. K., Soni, Y., and Dubey, S.: Enhancing effectiveness of motor rehabilitation using kinect motion sensing technology. In Global Humanitarian Technology Conference: South Asia Satellite (GHTC-SAS), 2013 IEEE, pages 298–304. IEEE, 2013.
  24. Hocine, N., Gouaich, A., Cerri, S. A., Mottet, D., Froger, J., and Laffont, I.: Adaptation in serious games for upper-limb rehabilitation: an approach to improve training outcomes. User Modeling and User-Adapted Interaction, 25(1):65–98, 2015.
  25. Baranyi, R., Willinger, R., Lederer, N., Grechenig, T., and Schramm, W.: Chances for serious games in rehabilitation of stroke patients on the example of utilizing the wii fit balance board. In 2013 IEEE 2nd International Conference on Serious Games and Applications for Health (SeGAH), pages 1–7. IEEE, 2013.

## VITA

NAME	Andrea Rottigni
EDUCATION	
	Master of Science in Computer Science, University of Illinois at Chicago, May 2016, USA
	Master of Science in Computer Science and Engineering, Politecnico di Milano, Sep 2016, Italy
	Bachelor's Degree in Computer Engineering, Politecnico di Milano, Sep 2014, Italy
LANGUAGE SKILLS	
Italian	Native speaker
English	Full working proficiency
	A.Y. 2015/16 One Year of study abroad in Chicago, Illinois
SCHOLARSHIPS	
Spring 2015	Research Assistantship (RA) position (20 hours/week) with full tuition waiver plus monthly stipend
TECHNICAL SKILLS	
Basic level	swift, php
Average level	Javascript, Python, D3.js, android, Smalltalk, html, css, mysql, SQL
Advanced level	Java, C++
WORK EXPERIENCE AND PROJECTS	
Dec 2015 - May 2016	Graduate Research Assistant at Electronic Visualization Lab, University of Illinois at Chicago, Chicago, USA
Mar 2015 - Jun 2015	Graduate Teaching Assistant, Politecnico di Milano, Milan, Italy