Expertise
and insight
for the future

Kun Zhu

Augmented Reality for Exercises for Elderly People

Metropolia University of Applied Sciences

Bachelor of Engineering

Electronics

Bachelor's Thesis

11 November 2020

Author Title Number of Pages Date	Kun Zhu Augmented Reality for Exercises for Elderly People 48 pages + 12 appendices 11 November 2020
Degree	Bachelor of Engineering
Degree Programme	Electronics
Professional Major	Electronics
Instructors	Anssi Ikonen, Head of Degree Programme

Augmented Reality (AR) physical training may have advantages over regular exercise training among elderly people. So exercise with AR tools can help to decrease the chance of falling and increase personal health. However, not a lot of actual products have been developed with this technology, they are mostly research-based. Therefore, this project aims to motivate elderly people to exercise more regularly by using AR technology and monitoring exercise quality. The project consisted of 3 members, thus please refer to Appendix 6 for workload distribution.

Using a Kinect V2, videos of participants performing physical exercises can be recorded. This data is then used to create algorithms to detect the same correct movements. These algorithms can be implemented in a game with Unity 3D.

The game has been designed and developed intuitively. The game is designed to use the human body as a controller, no handheld or physical buttons needed. Big images and texts are used, with a special focus on the picking of colours with high contrast. Furthermore, to enhance the motivational part, the game centres around travelling the world and showing the user different popular places.

The results show the importance of using AR as a technology for stimulating elderly people to exercise regularly. Besides, this project shows the possibility to monitor elderly people while exercising without the need of a physiotherapist attending. Lastly, good design and intuitiveness are key factors when developing a game for this target group.

Keywords	AR, Gaming, Sensor, Kinect, C#, Unity

Contents

List of Abbreviations

AR	Augmented reality (AR) is using certain devices to add virtual elements into real-world environments.
VR	Virtual reality (VR) is to simulate similar or different experiences from the real-world. VR applications can be used for entertainment (i.e. gaming) and educational purposes (i.e. medical or military training).
AI	Artificial intelligence (AI) enables systems to automatically learn and improve from experience. This process is programmed to simulate human intelligence in machines.
ML	Machine learning (ML) is an application of AI. The machines are programmed to automatically learn and improve from experience.
SDK	Software Development Kit (SDK), as the name suggests, is a collection of software development tools in one installation package [1].

1 Introduction

Elderly people tend to be more prone to falling than young people [2]. Besides, falling typically causes physical damages to elderly people. Lack of physical exercise is one of the common reasons that lead to falling. As shown in ”To prevent falls in older adults” [3], performing exercises regularly can help to reduce the chance of falling. This thesis is based on a study project, which reports about using AR technology to motivate elderly people to exercise. Three students were working for this study project. The workload and task distribution are documented as in Appendix 6.

As a side-effect, after falling for the first time, the elderly fear of falling again [2]. Rebuilding confidence in this area is one aspect of this study project as well. The study “Fear of falling: efficacy of virtual reality associated with serious games in elderly people” [4] showed the potential of gamified exercises. The exercises can improve the balance of the elderly, strengthen muscles and bring more confidence to the elderly when they are moving. Besides that, using AR or VR technology is possible to monitor elderly people, which reduces the need for assisting physical therapist.

There are studies about how to use innovative methods, such as AR and VR to encourage elderly people to move more. The study “Virtual reality balance training for elderly: Similar skiing games elicit” [5] talks about a ski game designed using VR equipment. Another research “Fear of falling: efficacy of virtual reality associated with serious games in elderly people” [4] also used the VR game approach. After reading through the study called “Virtual reality system based on Kinect [6] for the elderly in fall prevention”, the team decided to use Kinect. The team also tried to find more solutions for motivating the elderly to exercise more regularly.

This study project (thesis) focuses more on software development and game development. There is also a tutorial for building the involved hardware. This thesis investigates existing research about falling and fall prevention among elderly people. This thesis introduces methods and tools for developing an AR game. This thesis also presents the ideas and project evolvement to the prototype. The result section lists the game story, gameplay mechanism and feedback system.

2 Existing AR, VR applications and exercise program

2.1 Existing research about VR or AR projects and games

Researches have shown that VR can be used on balance training for elderly people. This has been described in the article “Fear of falling: efficacy of virtual reality associated with serious games in elderly people” [5], which uses skiing games to elicit different challenges in balance training. Similar games impose different challenges. There are researches on similar games, which aimed at decreasing unnecessary difficulties and increasing the effectiveness of the game as a balance-training tool.

Furthermore, with another research using VR in fall prevention [6], the authors developed an algorithm capturing the movements from players, then converted the motions into the game linked avatar. This avatar is the virtual representation of the player in the “digital world”. Different objects will appear from all sides on the screen and the player can try to “catch” or “touch” those virtual objects by making certain movements. Thus, the players can perform some designed exercises while playing a game.

Virtual reality exposure therapy can be associated with games that are designed for the treatment of fear of falling. This is presented in another study “Virtual reality balance training for elderly: Similar skiing games elicit” [4].

2.2 Choosing the exercise program

This chapter discusses the articles and topics related to this study project (thesis). The information from the articles helped the team to form a good and deep understanding of the topic. The chapter also includes introductions to the existing exercise programme, AR-related technologies and existing cases.

2.2.1 Otago exercise programme

Otago exercise program [7] is a result of many years of research [2], this exercise program was designed specifically to prevent people from falling. The exercise program consists of a plan for leg muscle strengthening, balance retraining and walking sets. Physiotherapists will prescribe the exercises individually depending on the patients' health conditions.

Otago includes movements at different complexities, and there are existing research articles and backup data for the performance of Otago. Thus, the Otago was selected for the gamification.

2.2.2 Picking the trunk movement for gamification

After a few test run with Kinect, several limitations were discovered. Kinect only has the function to capture the body frame from the front side or backside views of a player correctly. This means Kinect can not process a player's side view properly. Meanwhile, view areas are limited by the distance to the sensor.

With considering the limitations, the trunk movement as in Appendix 5 was selected for making the game. The entire game was developed based on this decision.

2.3 Augmented Reality

Different devices are used for different AR applications. For example, AR glasses called "HoloLens" from Microsoft uses AR technology to display information. Also, smartphones can use AR, like the famous game "Pokémon Go". In the game, users can catch creatures (Pokémon) in the street. In contrast to AR, Virtual Reality is a technology that immerses users in a virtual world thanks to VR headset.

In this game project, AR was used to motivate users to exercise more by gamifying exercises. Users will see virtual avatars showing them how to perform the exercises. Additionally, virtual elements will appear to give feedback about the performance.

2.3.1 Existing projects and games

2.3.1.1 Viarama [8]

Viarama uses VR within communities to improve the life quality of schoolchildren, senior citizens, and youths. Since seniors are included in the target group, Viarama is seen as an interesting reference for this game project.

Figure 1. A user playing VR game [8].

2.3.1.2 Pokémon GO [9]

Pokémons will be displayed on the mobile screen when the players point the screen to a correct location in the real world. The players (trainers) can “catch” those Pokémons.

Figure 2. Pokémon GO [9].

With collected Pokémons, the trainers can also combat with other trainers, battles will produce some rewards or losses for the trainers depending on performances.

This game also includes social networking. Trainers from different parts of the real world can communicate, combat and exchange gifts and trade Pokémons as they wish. This game was a great success for AR and mixed reality.

2.3.1.3 EchoPixel [10]

EchoPixel develops medical imaging devices to produce standard DICOM CT, MR, echocardiography and C-Arm fluoroscopy. True 3D software then translates these to real-world 3D models and holographic-like images. In this way physicians can rotate, re-seize, dissect and create virtual patient-specific surgical views.

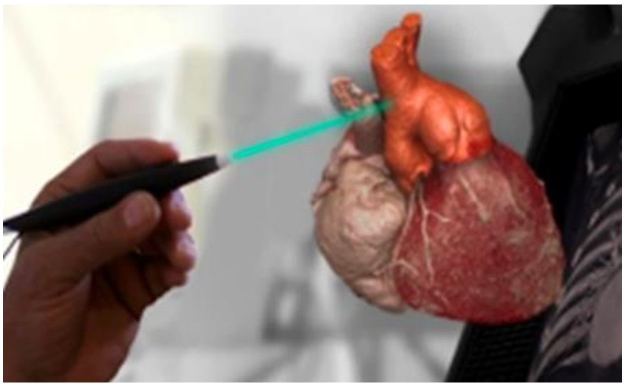


Figure 3. 3D renders from CT images [10].

3 Project management

This chapter discusses teamwork in a multidisciplinary team and the process to choose suitable development tools.

3.1 Teamwork

All project members had experience with the agile framework. The nature of the project includes the need for adapting to changes in requirements and skills quickly. So the agile framework was used for development.

Meanwhile, Scrum was used to include more definitions and specifications on top of the basic agile framework. Again, this choice was made based on the experience of the team members. The sprints period was one week, with a daily stand-up between 9 and 10 am. The sprint planning was on every Monday.

3.2 Progress tracking

3.2.1 Trello as the task tracking tool

Trello is a collaborative space, which enables teams to organize and prioritize a task flexibly. All team members had experience using Trello, this eliminated the time to study a new tool. Thus, Trello was selected for organizing the tasks and workload. All the “to do tasks” for the project were listed. There was a board with 3 categories as in Figure 4:

- To do.
- In progress.
- Done.

The team reviewed what was done on every Monday. Different tasks for the new week were distributed. The board was updated daily.

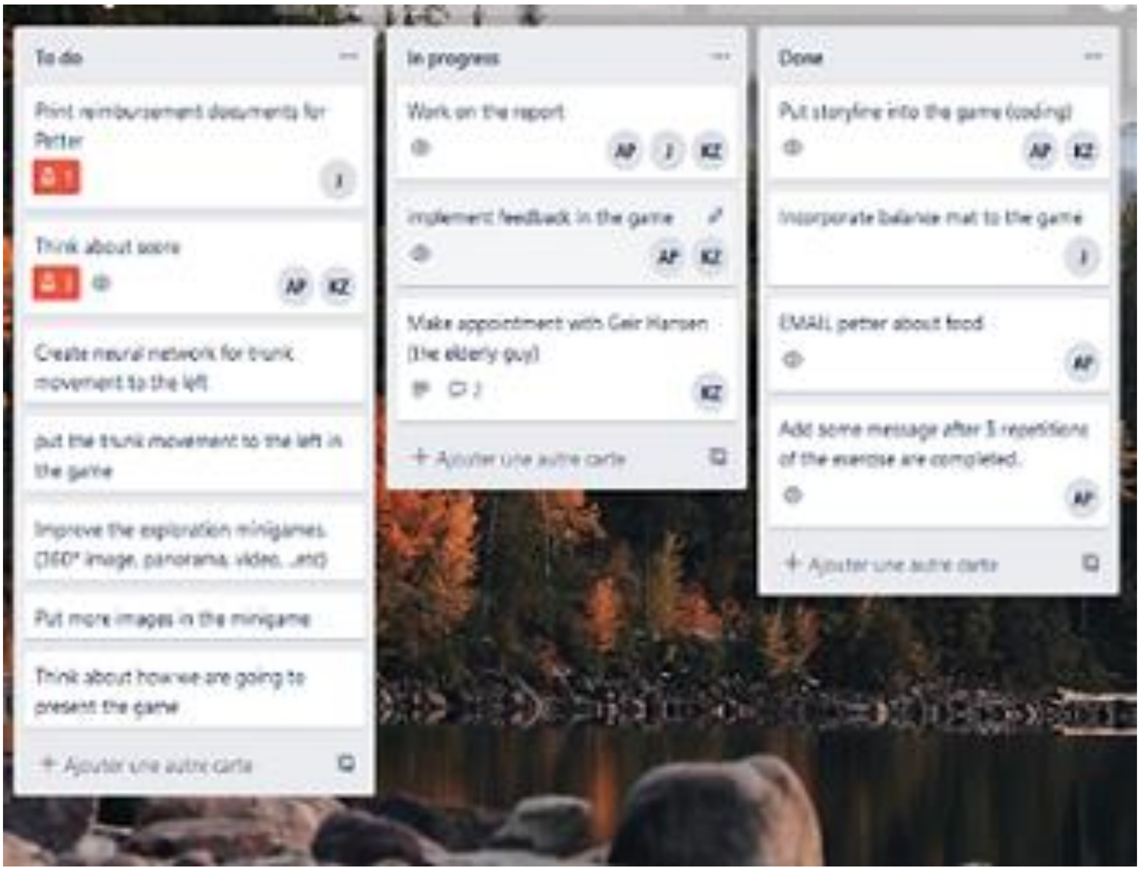


Figure 4. Trello task board

3.2.2 Whiteboard (Kanban)

In the workplace, there was a Whiteboard used for brainstorming and making schedules. Every Monday a calendar of the week were drawn, which included events like meeting or lectures. Since words do not always explain the ideas, the board was a good tool to remember different information and brainstorm sessions.

3.2.3 Gantt chart & timeline

The Gantt chart was used to keep track of the long-term tasks. Figure 5 shows how tasks were organized in the Gantt chart.

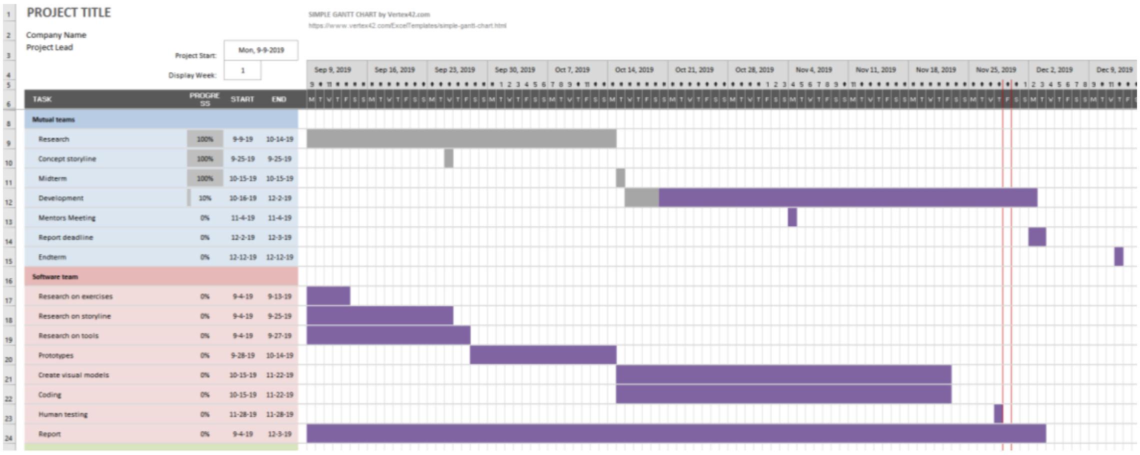


Figure 5. Gantt chart timeline view

In Figure 5, the left part was the task name, the horizontal bars represented the time set for completing one task. The greyed bars meant the completion of the tasks, and the purple means ongoing tasks.

3.2.4 Tool for version control

Version control plays a key role in software development projects. In the combination of web search results and experience, Gitlab and GitHub as the 2 biggest platforms were considered for version control. After looking into both platforms, Gitlab was chosen. The main reason was that the number of collaborators for a private repository is unlimited with Gitlab.

3.3 Method and Development tools

3.3.1 Video game engine

Professional recommendations and online sources provided knowledge about video game engines. The professionals were either studying or working in the gaming industry.

To choose the most suitable game engine, the following factors were looked through:

- Required skills for the programming language.

- Self-learning and other relevant skills required for using the tools.
- Available support communities.
- Support from the industry. As recommended, there is a design company called EggsDesign was developing a similar project, the smart rollator [11]. The team then paid a visited the person in charge at EggsDesign. The project itself integrated AI into a rollator. Thus, allowing elderly users to exercise around the rollator while the AR app helps to monitor their conditions.
- Friends to provide help and assistance

After comparing all the above-listed factors. Unity and Unreal, 2 of the most popular and easy reachable game engines were put on the shortlist.

3.3.1.1 Available support

Due to lack of experience with video game engines in the team, the team needed some supports while creating the prototype. Below were the three types of supports the team looked for:

- Community support from online sources and blogs.
- Company support from professionals who are working on similar products.
- Support from within social circles from friends or schoolmates.

The above points of the 2 game engines are compared in table 1.

Table 1. Game engine comparison.

	Unity	**Unreal Engine**
Community support	Yes	Yes
Company support	Yes	Yes
Social circle support	No	Yes

3.3.1.2 Required skills for the game engines

Programming language requirements are fundamental to develop a game. Unity uses C#, while Unreal uses C++ as coding languages. In the team, all members have some knowledge of Java, which makes the learning and developing easier under the Unity environment. Besides, C# as a programming language is similar to Java. Thus, for the team, C# was a better option than C++.

Both Unity and Unreal support 2D and 3D modelling.

In case of any future development and reference to operating platforms. Table 2 shows the supported platforms.

Table 2. The supported platforms.

Unity	Unreal Engine
MacOS X, Windows, Linux, Android	Windows, Linux, Android

3.3.1.3 Self-study for the development tools

Suggested by experienced developers, the team also looked into the factors about the 2 game engines as in Table 3.

Table 3. Usability and ease of using.

	Unity	Unreal
Programming language	C# and Javascript	C++
Interface	Simply to use	Complex but powerful
Toolbox	Commons are included	Complicate, can achieve more

As a comparison to the mentioned factors about the 2 game engines, the team decided to use Unity.

3.3.2 The working mechanisms and functionalities of Kinect

3.3.2.1 Choosing the movement sensing device, Kinect V2

As searched from the internet and investigations to similar projects, there were several options for movement capture devices, such as Azura Kinect, Oculus, Kinect V2. In consideration of the information, time and materials that can be gathered, Kinect V2 was chosen based on the following criterion:

- There was an available Kinect V2, using this one would save time waiting for buying other devices.
- The device was released a few years ago, therefore there were many existing repositories and tutorials to learn from.
- For a proof of concept, the device was good enough to achieve the goal.

3.3.2.2 Kinect and its functionalities

Kinect is embedded with infrared projector, infrared camera and 3D scanner with the Light Coding. The Light Coding employs a variant of image-based 3D reconstruction. There are also a depth sensor, an RGB camera, and microphone arrays integrated into the device. The mentioned sensor units can provide facial recognition, full-body 3D motion capture and voice recognition.

With the trained data and comparing the user's 3D postures to the trained data, Kinect SDK could determine whether the user was performing the exercise as instructed.

3.4 A learning project on how to use Unity and Kinect

3.4.1 The reason for a learning project

Since it was the first time for all the team members to use the API Kinect and Unity. Before the actual game development, the team set up a quick and short learning project on how to use Unity and Kinect. This learning project was to also test whether the chosen devices and game engine could achieve the desired game functionalities.

3.4.2 The goal for this learning project

For this learning project, the movement was simple, as lifting the right arm to shoulder height. This movement was simply used for trying out the API with Kinect. The point for this learning project was to learn the basics to combine Kinect and Unity to develop a simple game.

3.4.3 Game concept

The theme and game concept was based on a survey (as Appendix 2), which interviewed 10 elderly people at their retirement age (60 – 80 years old). The elderly tend to move less due to physical conditions, but they want to exercise more and keep fit. Gardening involves a decent amount of physical exercises. Thus, the chosen thematic was Gardening. The goal of this game was to grow the garden by making a move.

3.4.4 Initial design and test scenes

The design was a simple representation of a Garden. There were 3D models of flowers, trees, fountain and so on. These 3D elements were free on the assets store in Unity.

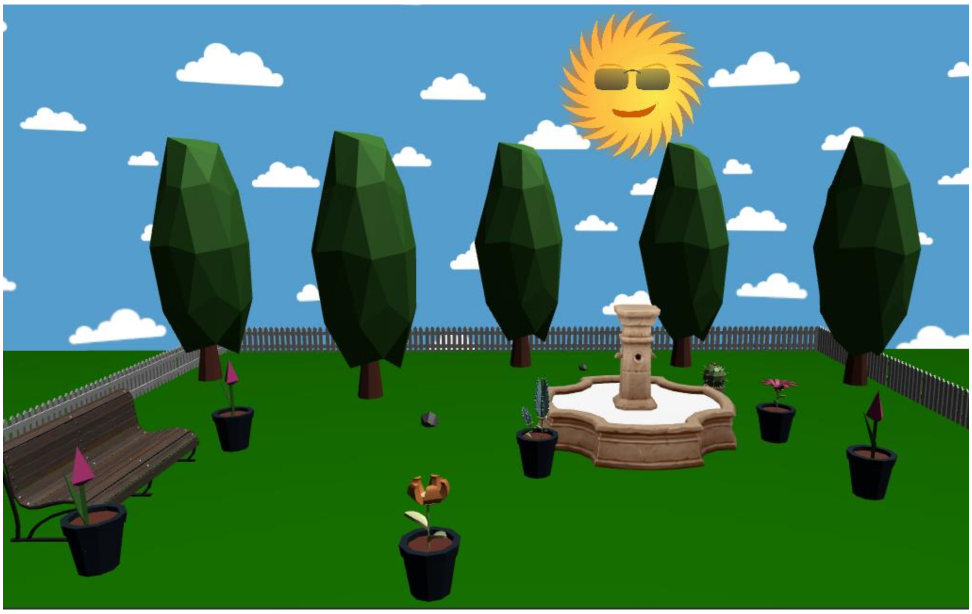


Figure 6. The correct movement followed by the sun “smiled” and flowers grew.

When the user does the movement correctly, the sun changes into a happy Sun and flowers appears as in Figure 6.

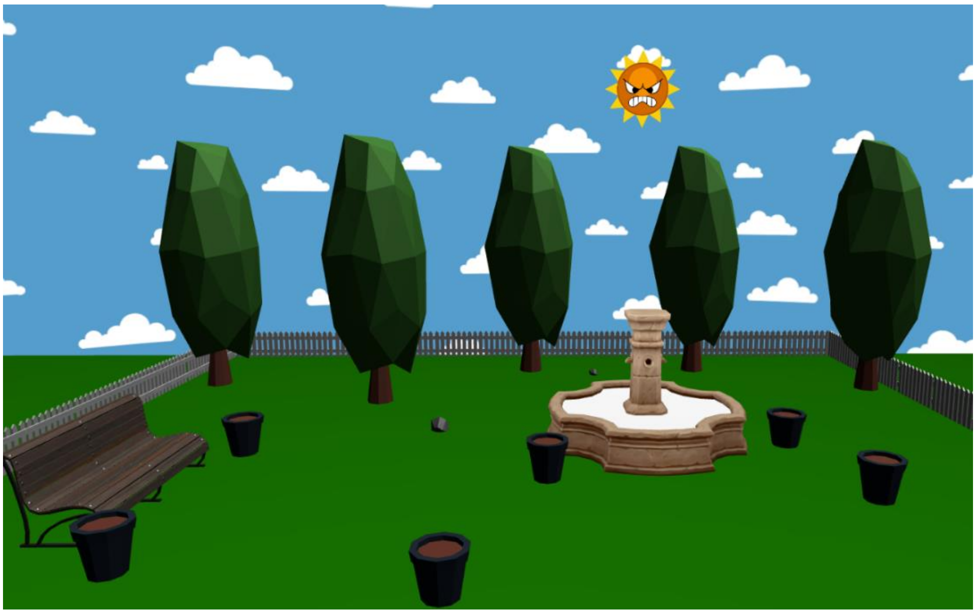


Figure 7. The incorrect movement followed by the sun “got angry” and flowers disappeared.

In comparison to Figure 6, when the user did the movement incorrectly, an angry sun replaced the happy sun and all flowers disappeared in Figure 7.

Even if the designers have solid experience with CAD (Computer-Aided Design) tools, making 3D elements and models are time-consuming. So, using the assets store can save time. The team followed this strategy. As a result of this learning project, the team learnt how to use the Kinect API and how to implement exercise movements into a game.

4 Design and development

This chapter goes over the topics on how to achieve the desired game mechanisms. For exercise recognition, the data about movement is collected by Kinect. After collecting, the data will be labelled, then fed into the SDK neural network. Thus, the SDK is capable to recognize movements. The output of the neural network is a value of percentage confidence. This value is an indicator of whether the neural network thinks the movement is in a target set. At the end of this chapter, the method of judging movements is modelled mathematically.

4.1 Movement recognition

To verify if the player is performing the correct exercise well, the correct movements has to be set in the game. For this purpose, a piece of software capable of recognizing movements was needed. Since there was no specific software like this available online, the team used machine learning for developing the neural network. The team created a neural network of the exercise that has to be recognized. Figure 8 shows the steps used to create these neural networks, which was used for movement recognition.

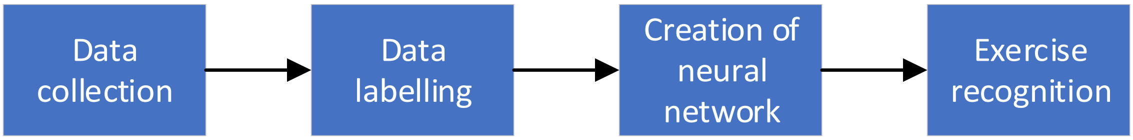


Figure 8. Visualized steps of creating a neural network for gesture recognition.

4.1.1 Data collection

The first step of creating a neural network was to collect the training data. Kinect in combination with Kinect Studio 2.0 [12] can record the required videos clips. One single neural network needed 15 to 20 recorded video clips of each exercise movements. These clips captured a person performing the exercise with the correct postures and movements. The more data references the better output predictions the neural network make. Thus, all project members participated in recording the clips. In Figure 9, the images are the progression of one trunk movement exercise as screenshots of the recorded video clips. The video clips were captured with the 3D camera of the Kinect. These images showed that the Kinect was capable of creating a skeleton view of the user. Kinect used the joint points as 3D special positioning anchors.

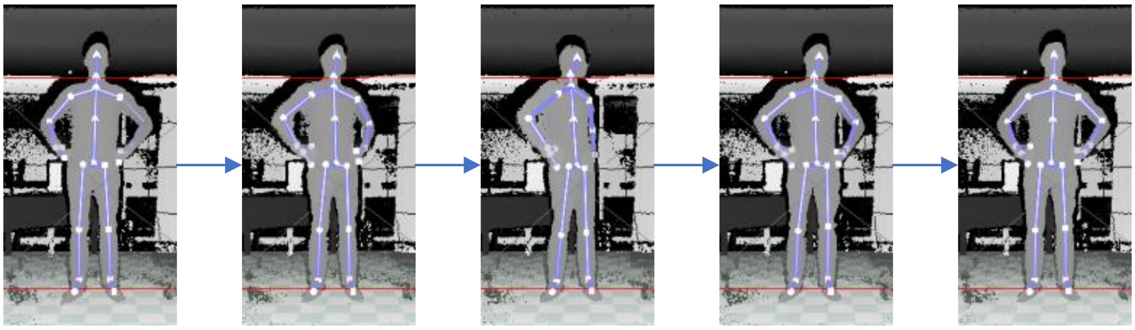


Figure 9. Progression of a test person performing the trunk movement to his left.

4.1.2 Data labelling

Once the data collection was completed, the next step was to use Visual Gesture Builder [12] to label this data. Then all frames in the video were to be labelled with a specific tag.

As in Figure 10, the horizontal blue lines indicates the labelled data. The light grey bar indicates all the frames of the video in chronological order. The vertical dark grey line indicates the currently selected frame. The full representation of the currently selected frame is shown on the top of the light grey bar, this frame shows a person in the middle of a trunk movement. The blue lines show how each frame was labelled. The definitions of the blue lines are as the following:

- **The upper blue line**: performing a trunk movement.
- **The lower Blue line**: not performing a trunk movement.

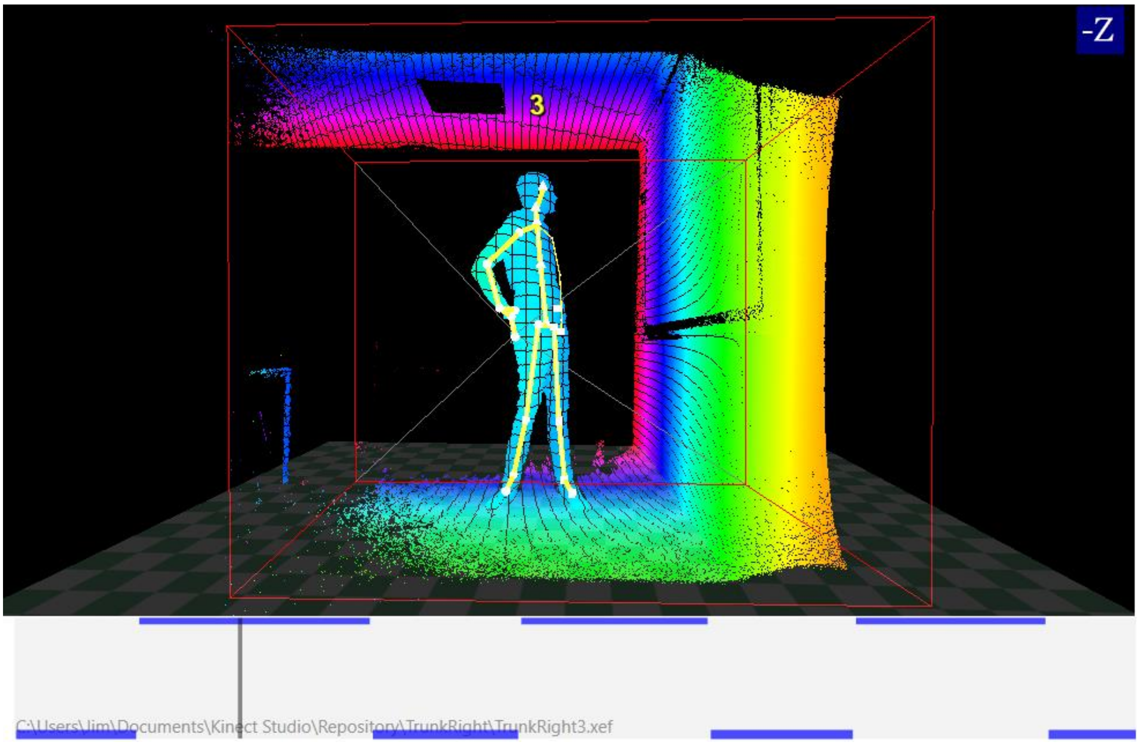


Figure 10. Video screenshot of a user performing the 3rd repetition of the trunk movement.

4.1.3 Creation of the neural network

Once all the data labelling was completed, one neural network creation cycle was also completed. The programs took all the data from the video clips and created one model out of those, this model is called a neural network. One neural network can only detect one specific exercise movement as designed. Thus, for the scope of this game project, multiple of these had to be created using this method.

4.1.4 Movement recognition

To test the output of the neural network, the "live preview" feature of Visual Gesture Builder [12] was used. Figure 11 shows the screenshot of live video input on the left and an output graph on the right, the person in the video was performing a trunk movement exercise. The graph shows the time horizontally and the confidence of the network vertically.

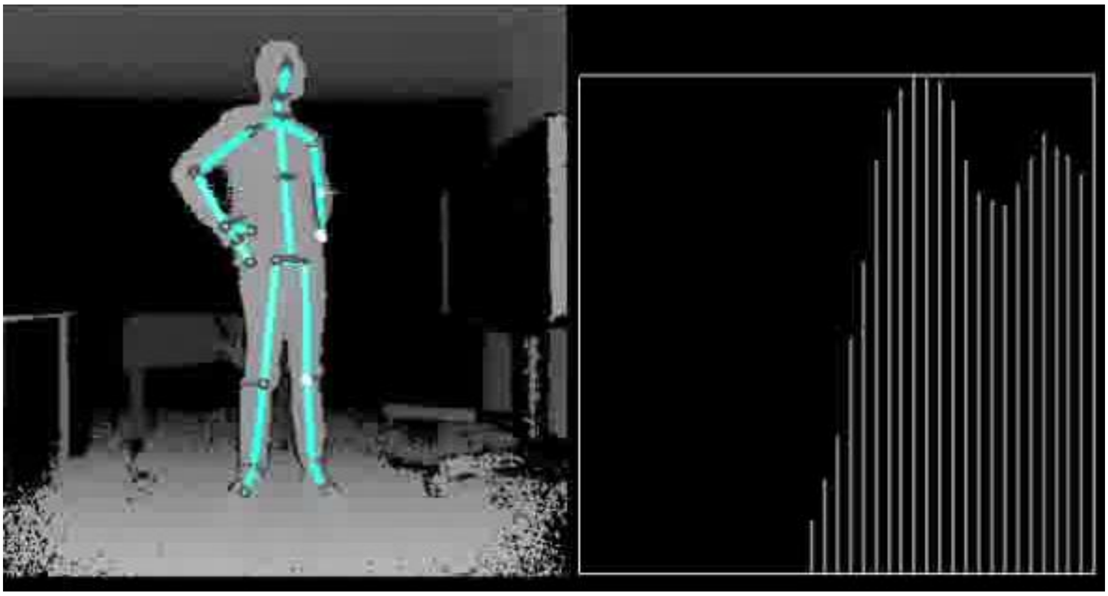


Figure 11. Neural network visualizing results based on live video input.

4.2 Refining the raw data

4.2.1 Distinguishing the correct and incorrect movements

An algorithm was created to make distinctions between correct and incorrect movements. The algorithm counts the numbers of movements performed chronically. The neural network outputs a prediction at any given moment, and the confidence value is only an estimation. Thus, the confidence values do not represent a fact that the person is performing targeted movements. A better approach must be set for the computer to decide whether an exercise has been performed correctly. The following is the mathematical way that explains how the computer decides the movements.

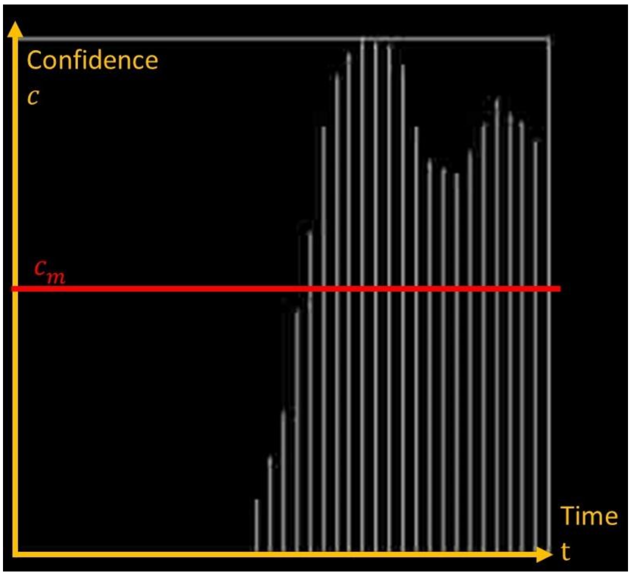


Figure 12. The output graph from Figure 11.

Assume the confidence value as a function of time $c(t)$. At arbitrary and continuous-time points t_1, t_2, t_3,..., the corresponding confidence values are c_1, c_2, c_3,.... As previously mentioned, the confidence values are predictions, so the bigger the c values the better. Thus a threshold c_m as the minimum confidence is set.

In a perfect situation, the c values will increase gradually, then reach the peaks for a certain amount of time. After the peaks, the c values drop gradually. The whole graph should look similar to a sinusoid function.

However, due to environmental noise and other affecting effects, there could be sudden spikes or drops to the confidence values. Thus, there has to be a good amount of c values in a fixed period to determining the correct movement. Assume a set arbitrary time frame as T to act as the correct movement recognition period.

In any given period, the following equation should be fulfilled for correct movements:

$$\int_{t_n}^{t_{n+1}} \mathrm{c}\, dt > T * c_m \qquad (1)$$

Then $Boolean(trunk) = TRUE$, otherwise $Boolean(trunk) = FALSE$.

With some combination of experiments and personal intuitions, values for $c_m = 50\%$, $T = 100\ ms$.

4.2.2 Determining a well-performed exercise

The trained data can determine whether a user is performing the exercise as instructed. The data checks the user's 3D postures and compares the postures to the trained data. This only tells whether the movements are correct or not. However, to tell users if they are performing well or not is also important for game interactions. A feedback system can act as an indicator to the users. So animated feedback system was implemented in the game to notify users. Meanwhile, the feedback also instructs the users for the correct movements. For this gaming project, the exercise speed and feet distance feedbacks are implemented.

4.3 The development of the balance mat

As explained in paragraph Otago exercise programme, balance is a key factor in performing exercises correctly. A balance mat was introduced to capture the balance and integrate these measurements into the game. The user steps on the mat, then the mat sends data about the pressure the computer. Figure 13 is a picture of the balance mat. Figure 14 shows the steps taken from a user stepping on the mat to the visualization within the game.

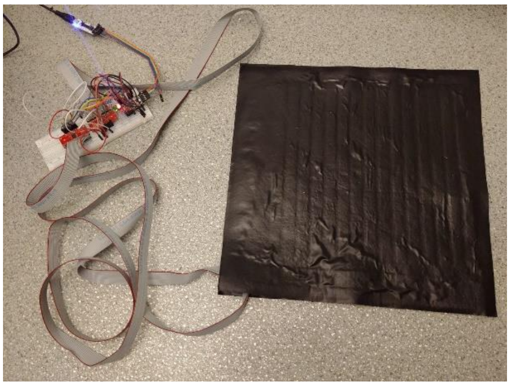


Figure 13. The balance mat.

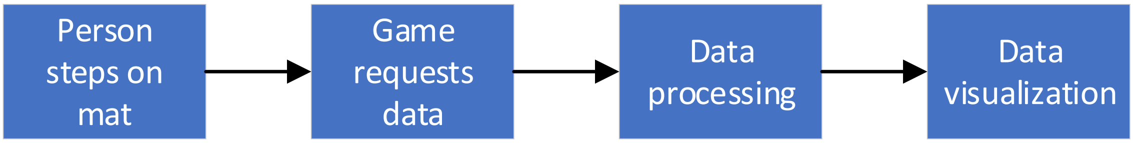


Figure 14. Visualized steps of putting the balance of the user in the game.

4.3.1 The working mechanism of the mat and the building tutorial

The mat was composed of rows and columns of copper tape (as in Figure 15), the row or column was stacked on top of the other. In the middle, space was filled with the Heavy Weight and Moisture Resistant Shielding Plastic called “velostat”.

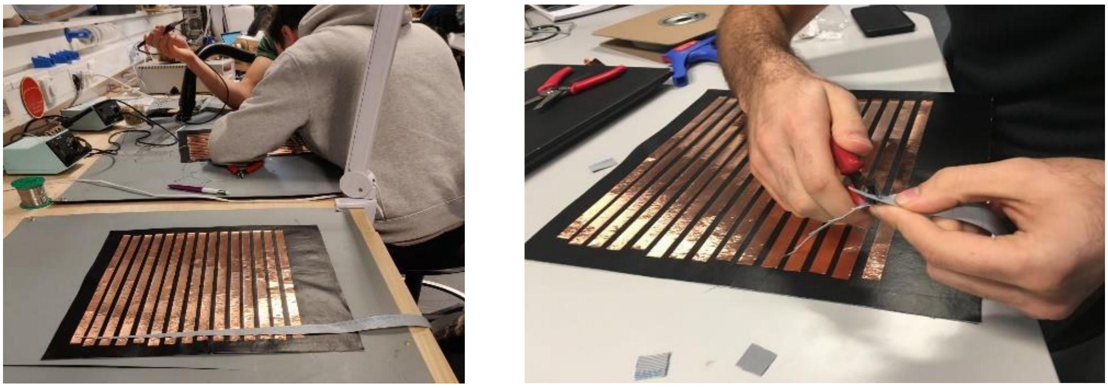


Figure 15. The balance mat "row" or "column".

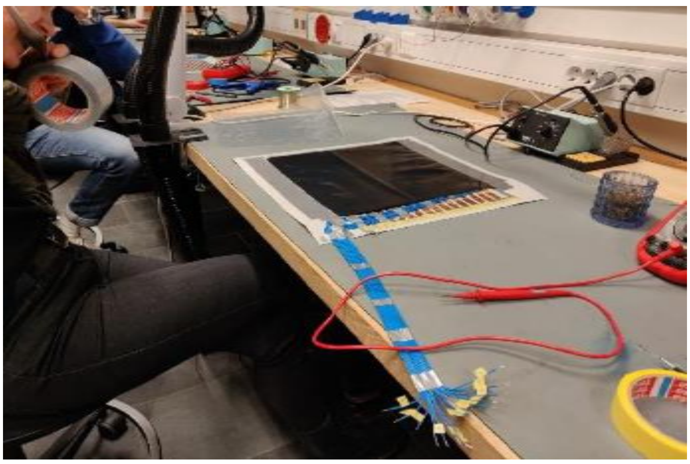


Figure 16. The velostat “filling”.

As in Figure 16, the velostat was filled in between the columns and rows. The resistance of the velostat changes by applied pressure, thus to create a 15 by 15 sensor matrix.

4.3.2 Reading the analog signal with Arduino

Arduino system was connected with balance mat using two 16 channel multiplexers as shown in figure 17 and 18. The bill of materials is listed in Appendix 3.

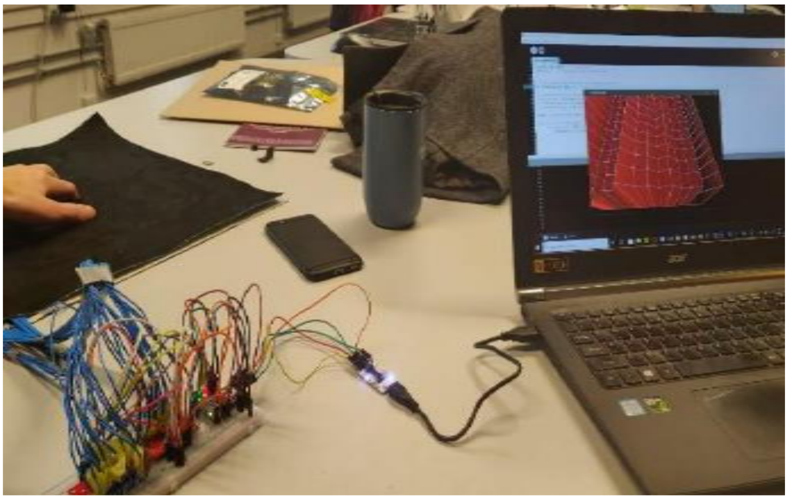


Figure 17. Arduino connection on a breadboard.

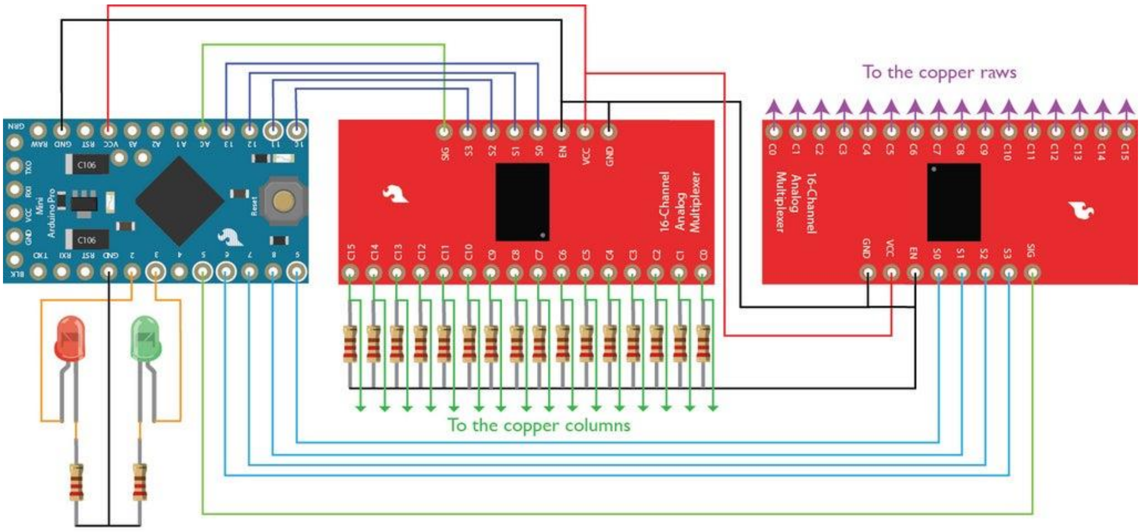


Figure 18. The Arduino connection diagram [13].

As shown in Figure 18, there is 1 Arduino mini pro, 2 multiplexers, 2 LEDs and several resistors. In this setup, the green LED is corresponding to the status of the “columns” and “rows” connection. The red LED is indicating whether the values are changing, meaning if there is pressure applied to the mat. The 2 resistors connected to the LEDs are current limiting resistors, which has the same purpose as the resistors connected to the “columns”. When the connections are established, the next step is to upload the code (see Appendix 4) to the Arduino mini pro.

4.3.3 Communication between Arduino and Unity

The communication between Arduino and Unity was based on serial data via serial ports. Note that the baud rate in Unity should be kept the same as in the Arduino setup as in Listing 1.

```
stream = new SerialPort("COM4", 115200);
```

Listing 1. The code line that sets up the serial port and baud rate from the ArduinoController.cs. The complete code pack can be found in Appendix 4.

After the communication between Arduino and Unity was established, the corresponding pressure map can be read from the balance mat as in Figure 35 to Figure 40. Since the program calibrates each time when the pressure is removed, the values shown in Unity were relative. Thus calculating the absolute pressure values was not necessary. Moreover, weight measurement can also be implemented if wanted.

4.3.4 Data requisition of the game

The first step was to apply pressure on the balance mat. A communication channel using the serial port was set up for Unity to request data from Arduino. Since Arduino does not have functionality for communication via built-in serial ports, an open-source library called “ArduinoSerialCommand” [14] was used to compensate for this.

The game, within a set time interval, requested every point pressure measured. The data came in a serial sequence. The sequence was then organized as shown in Figure 19, every number represented one pressure point on the mat.

```
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,2,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
```

Figure 19. Data requested from the balance mat.

4.3.5 Data processing and visualization

The data needed to be validated for it to be processed. After the validation, the data was added to a list and sent to the visualization part of the program.

The last step was to visualize this data within the game. The idea was to display the pressure with 2 parameters, the colour and the depth. The below image Figure 20 was an inspiration for the pressure map as in Figure 35 to Figure 40. This code pack is under ArduinoController.cs, included in Appendix 1.

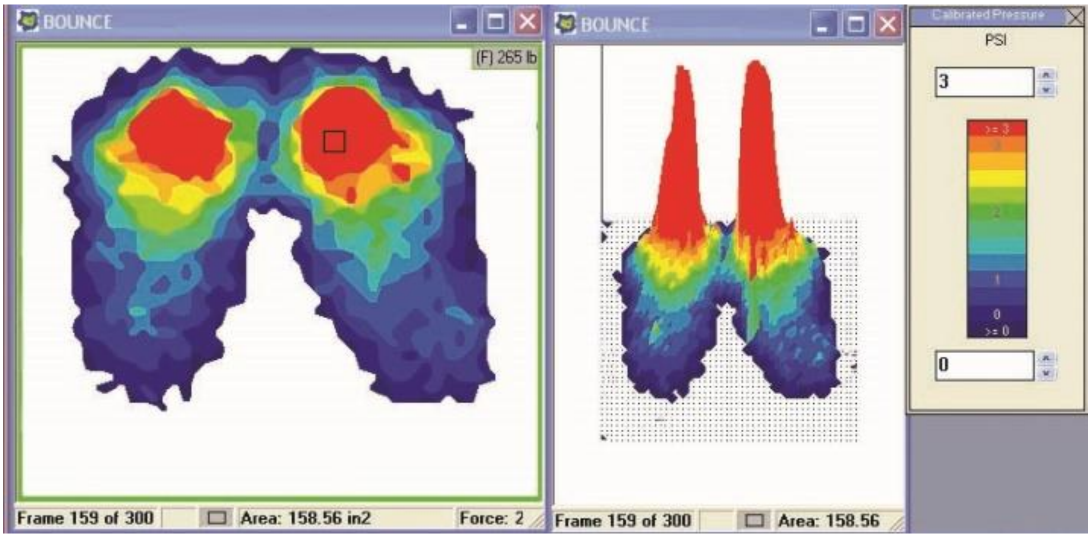


Figure 20. Inspiration visualization of balance mat [15].

4.4 Interface design

4.4.1 Trunk Movement Explanation

To introduce the exercise “Trunk Movement ” (see Appendix 5), a scene was created to explain clearly how to do the exercise. This scene consists of 4 instructions, advice and the number of repetitions. A humanoid avatar shows the movement from the top, the front and one side. The user is free to go to the next scene by forming a T-pose.

The starting scene:

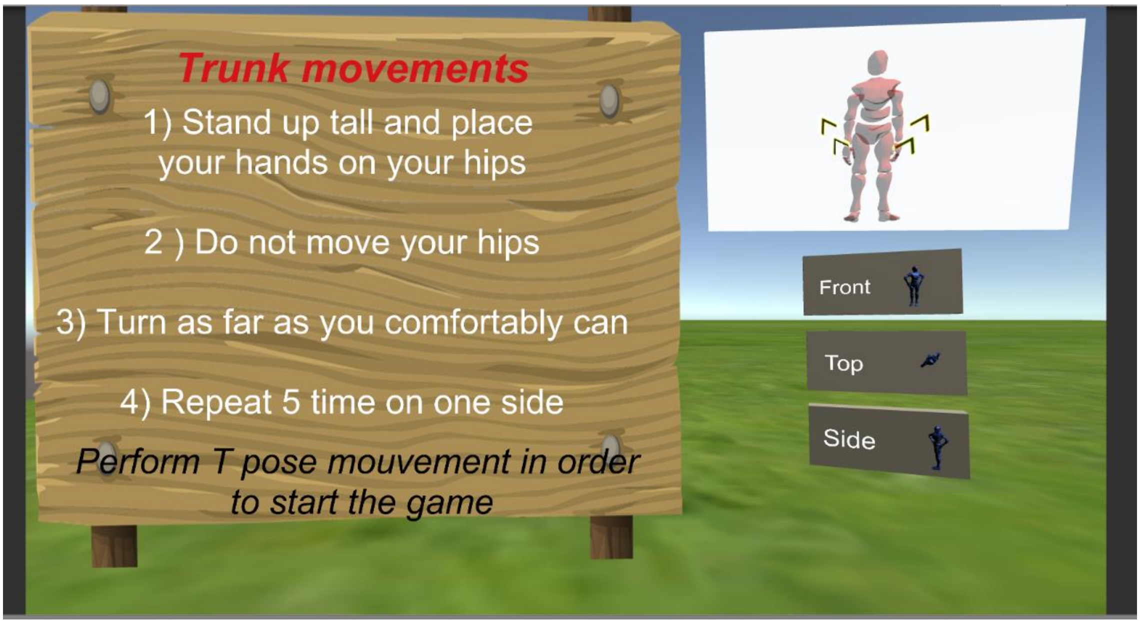


Figure 21. The initial design of the trunk movement scene.

The first step was about making the scene with all the elements. As we can see in Figure 21, the graphics need improvements. Designing a “beautiful” scene is difficult.

The team asked for advice from students with designing background, “too many different colours“ as they commented. Based on the feedback, sets of colours were minimized. The scene was modified visually as in Figure 22.

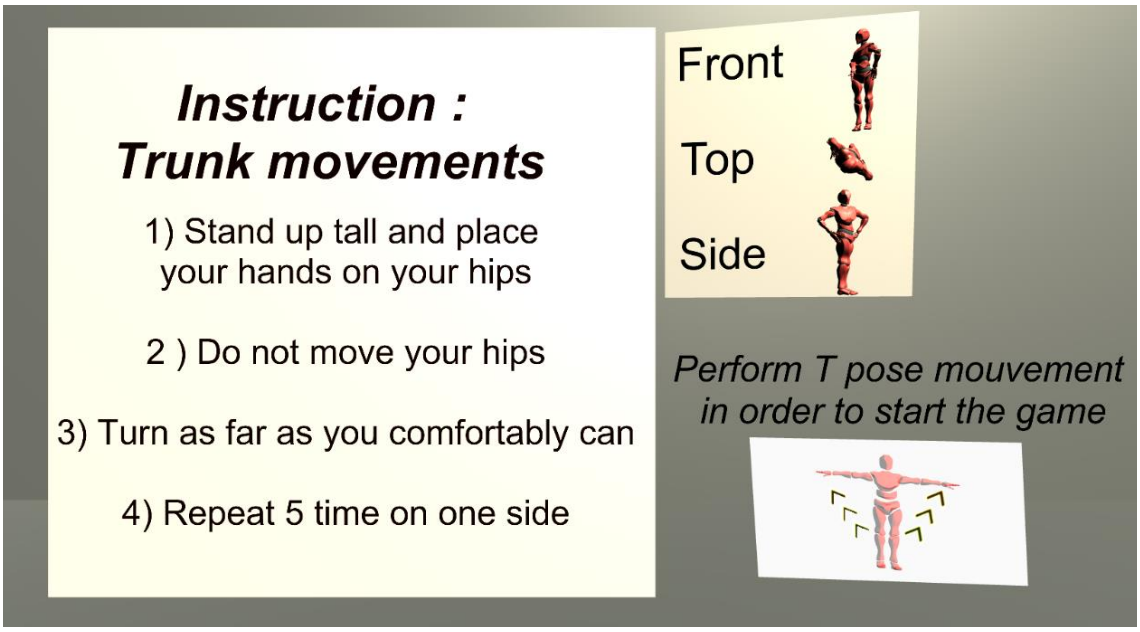


Figure 22. The revised design scene of the trunk movement scene.

The final design of this scene is simpler and lighter for the eyes. Different areas are easy to identify for users, especially for elderly people.

4.5 Feedback in the game

Initially, 2 types of feedback to the users were considered. One was exercise speed, the other one was feet distance. However, the team did not figure out a good algorithm for giving exercise speed feedback. So only feet distance feedback was implemented in the game.

4.5.1 Feet distance feedback

The idea for this was to give users feedback about the distance between their feet while standing and performing the trunk movement exercise. The most intuitive approach as discussed in the team was using animated avatars that demonstrate the feedback. However, in this way, a much more complicated system must be developed. Eventually, feet images on the screen showing the directions for moving was considered simple and clear enough. The mechanism is described as the following:

- When feet distance is incorrect, the symbols of feet appear and notify the users as in Figure 23 or Figure 24;
- The screen is clear with no feet symbols when everything is correct;
- The directional arrow can show to the users the moving directions;

To decide the distance value between 2 feet, as to Otago instructions [2], the exercises were designed based on the comfortableness while performing the exercises. So the team experimented with feet distance values starting at 0.5 to 3 times of the shoulder-width, the step values were 0.3, 0.5 and 0.8. As a result, the value range close to the shoulder width was comfortable enough and implemented. However, dynamic values which fit different users would be optimal.

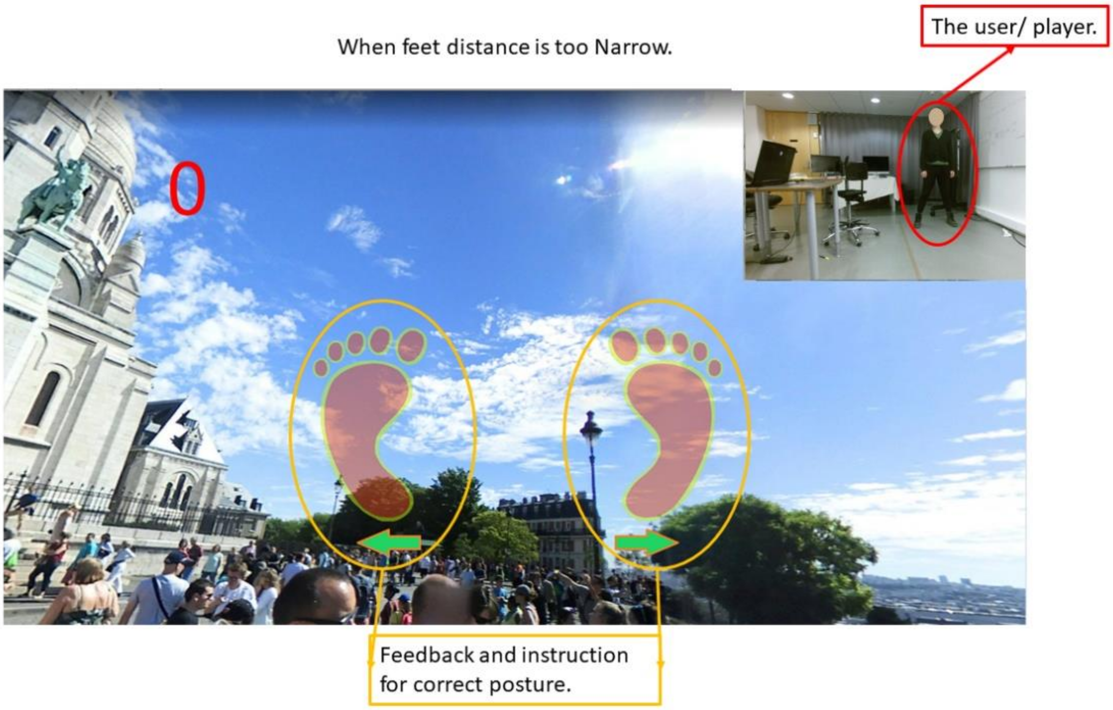


Figure 23. Feedback for narrow feet distance.

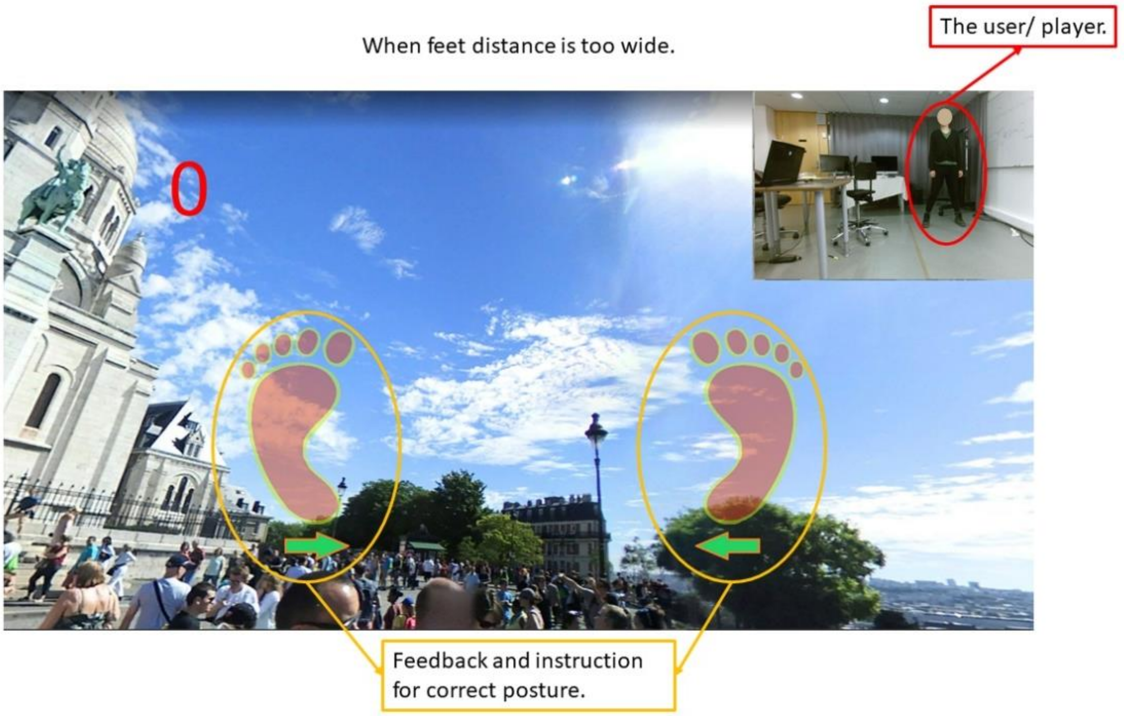


Figure 24. Feedback for wide feet distance.

The above Figure 23 and Figure 24 show the feedback for feet distance. In this game scene, the top right corner is the real-world view of the player. The background is a picture of the travel site. In normal cases, the middle is clear. There will be left and right foot symbols and arrows in case of a narrow or wide feet distance. When this happens, the feet symbols are animated to move either away or closer as the arrows indicate. The player is indicated to stand on a wider or narrower feet distance.

4.5.2 Exercise speed feedback

This feedback was created in a separated manner, meaning this functionality was not integrated into the final demo. The reason was that the team could not create a proper algorithm for the feedback. If this functionality can be properly implemented, the notifications will look as shown in the below pictures.

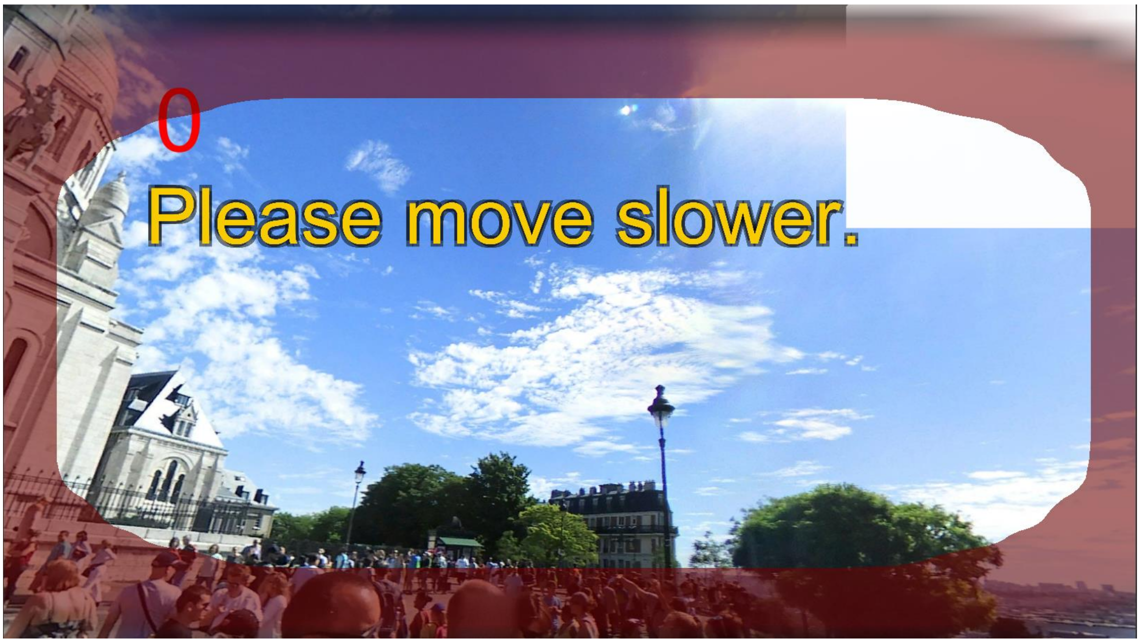


Figure 25. Speed feedback: too fast.

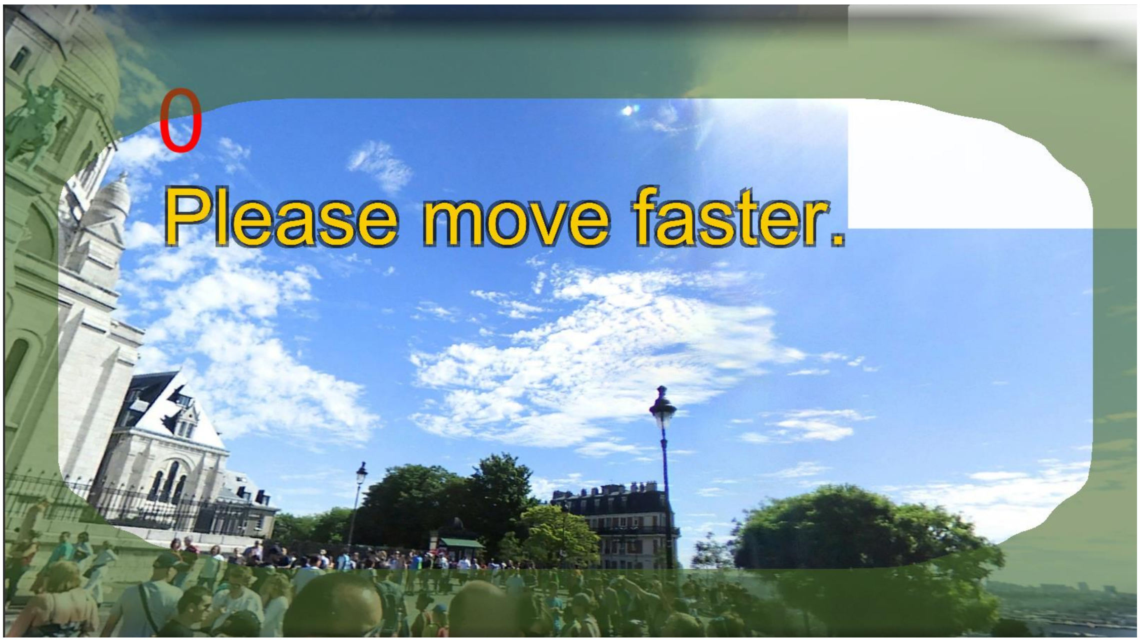


Figure 26. Speed feedback: too slow.

As in Figure 25 and Figure 26, the edge of the screen will be blinking, and the texts will appear if users perform the movements too slow or too fast. Otherwise, when the exercise is done correctly, the view is clear with the top right corner as the real-wold view and the background photo.

5 Gameplay

In order to play the game, the minimum requirements are a Kinect V2, a laptop that runs Windows 10 OS with Kinect SDK installed. For the best experience, use a big screen while playing.

5.1 Navigation in the game

In every game, a user has a way to navigate and interact with the computer. Thanks to the Kinect, in the game a fixed posture can act as navigation and interaction with the computer. The fixed posture is acting like using a mouse to a PC or touch to a phone.

5.1.1 T-pose

5.1.1.1 The T-pose

The player stands straight up, then raises the arms to the shoulder height. Thus to make a shape like a letter “T” as shown in Figure 27.

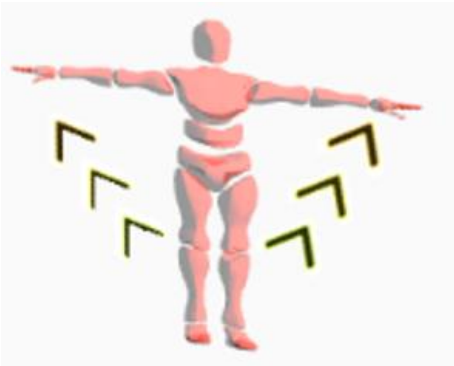


Figure 27. The T-pose.

5.1.1.2 Purpose of the T-pose

The T-pose is used to start the game for the following two reasons.

- The posture is simple.
- The posture is distinct to other movements for moving to the next scene.
- The posture adds a warm-up exercise for the user.

The user can make this T-pose to indicate for moving to the next step when ready. The user can read rules or instructions before moving to the next scene or starting a new round.

6 Results

The game is composed of 4 scenes, Storyline, Board-Scene, Trunk-Movement-Explanation, First-Minigame in order of appearing in the game flow.

Storyline: this appears in the beginning to tell the story of the game.

BoardScene: players will see 2 avatars, a granny and a grandson standing on a board and following certain routes to attraction sites.

TrunkMovementExplanation: there are both texts and animation instructions to players about how to perform the trunk movement.

FirstMinigame: this is where the players perform the exercises as instructed, they will be able to see the attraction sites in a panorama view while performing the exercises.

The prototype is capable of recognizing movements according to standards and feet distance as the set interval. Exercise speed detection was not successfully integrated. The last functionality is the pressure measurement board. The game also incorporates a balance mat, to check and compare the pressure of the feet and show the balance of the player.

For the final game demo, the team achieved most of the functionalities. The game can be used as a showcase of a combination of AR and games to motivate people for exercise, at the same time gather data for better exercise performance.

6.1 The storyline of the game

The idea is to make the game attractive, easy and fun to play for elderly people. So, a survey [16] was conducted, which took interviews from several elderly in their retirement. The results show board game is most popular among the interviewed. The results also indicate some other activities that a senior would love to do in spare time, such as travelling, gardening, playing with their grandkids and so on.

Considering the information collected, a board game styled idea came into being. Due to some reasons and health conditions, seniors are not always able to do what activities they want. The game combines some of their favourite activities, such as sight-seeing and travelling. Thus, a travelling mechanism was added upon to the board game.

6.1.1 The game storyline

When the game starts, some photos of the game trip destinations will be shown. At the same time, shorts stories and photos about the destinations appear. The below texts are the stories written for each slide for the trip destinations.

- It's a summer holiday for the grandson. Grandma is taking her grandson on a world trip. The journey starts from Paris in France. They will visit the Eiffel tower and eat croissant there.
- After Paris, both will head to Norway and check on the Vikings heritage. They will also enjoy the northern light during the stay in Norway.
- The fragrance of the tulips welcomes all travellers including the grandma and the grandson to the Netherlands. They ride bikes in the fields to visit windmills, the grandson went into a windmill for the first time and got to know how it works.
- Last stop in Europe will be the industrial house of Europe, Germany, both are enjoying the tasty sausages served with bread. Grandma takes also a bit German fine beer, but for grandson, it's too early now, maybe he will come back for the beer with his grandchildren.
- Flying in the sky from Europe to China, the Great Wall went into their eyes. Landed on the land, a group of different style of architectures immersed the two visitors from far.

6.1.2 The game flow

A normal scenario is described as the following shown in Figure 28:

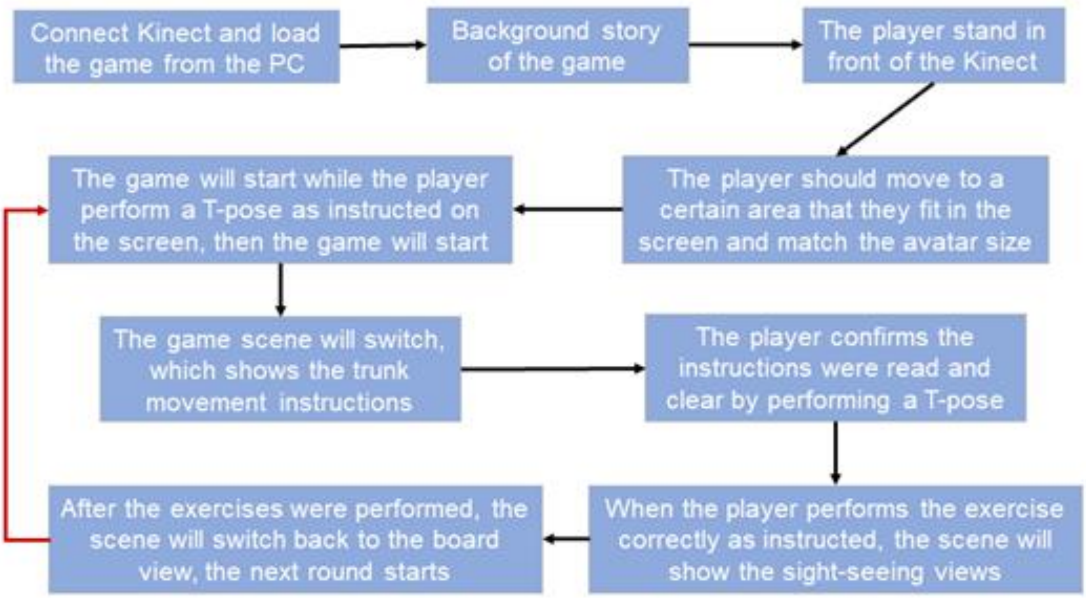


Figure 28. A typical game flow.

Figure 28 shows the typical game flow procedures. The game runs similar to a state machine indefinitely until the users quite the game.

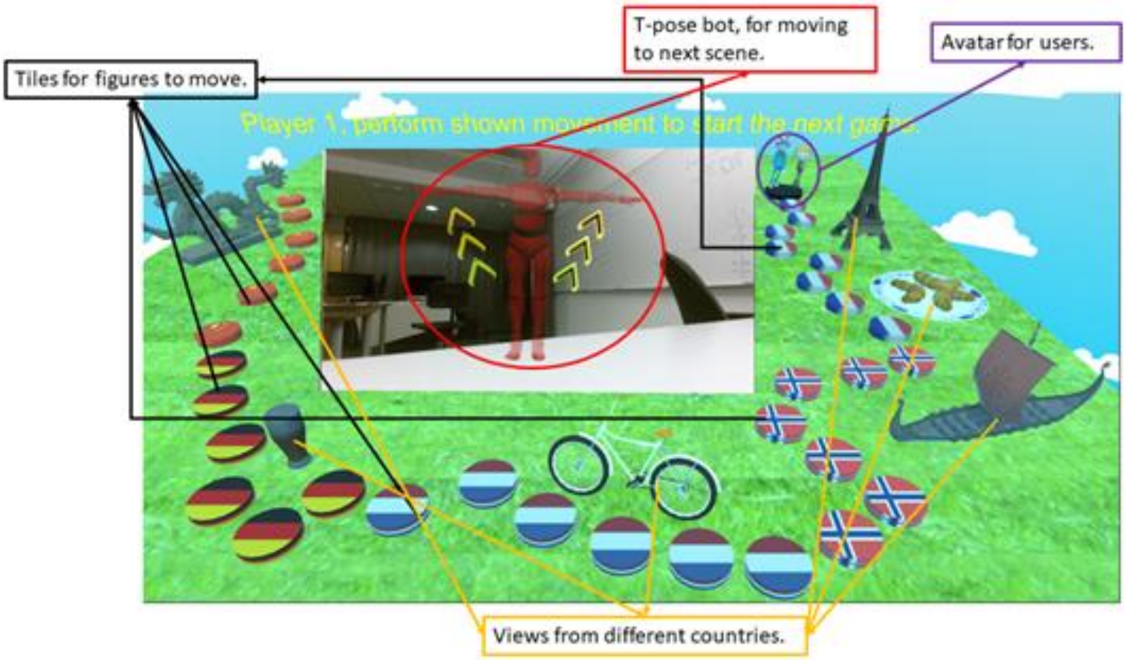


Figure 29. The mainboard view.

The mainboard view of the game is shown as in Figure 29. In this view, the steps stones are painted with country flags. Along the route, there are some well-known symbols. On the top right corner, there are 2 avatars, the granny and the grandson. In the top middle area, there are instruction texts for the users, the texts indicate which of the avatars is activated. So the users can decide what to do correspondingly. In the middle of the board view, there is a real-time camera view, this shows how to make a T-pose and move to the next scene when the user is ready.

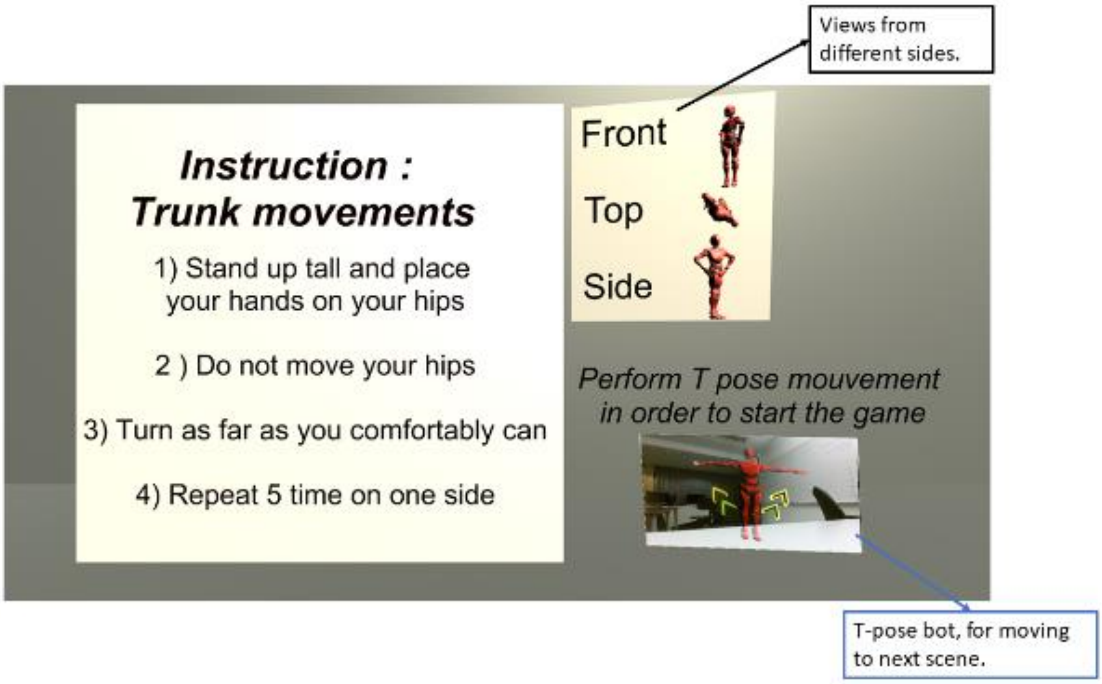


Figure 30. Exercise instructions view.

In Figure 30, there are 3 sections. The left, which gives instructions for the trunk movement in texts. The top right, which demonstrates the trunk movement in the “Front”, “Top” and “side” views. The bottom right, there is a real-time camera view for T-pose instructions.

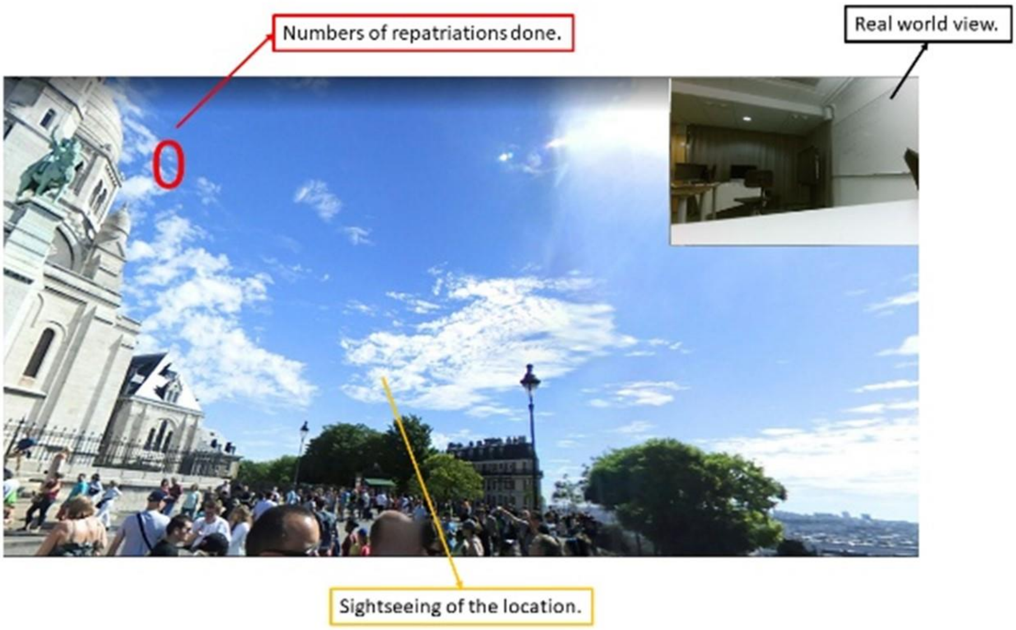


Figure 31. The minigame scene.

In Figure 31, there is a normal view of the first exercise scene. This view is a photo from a travel site.

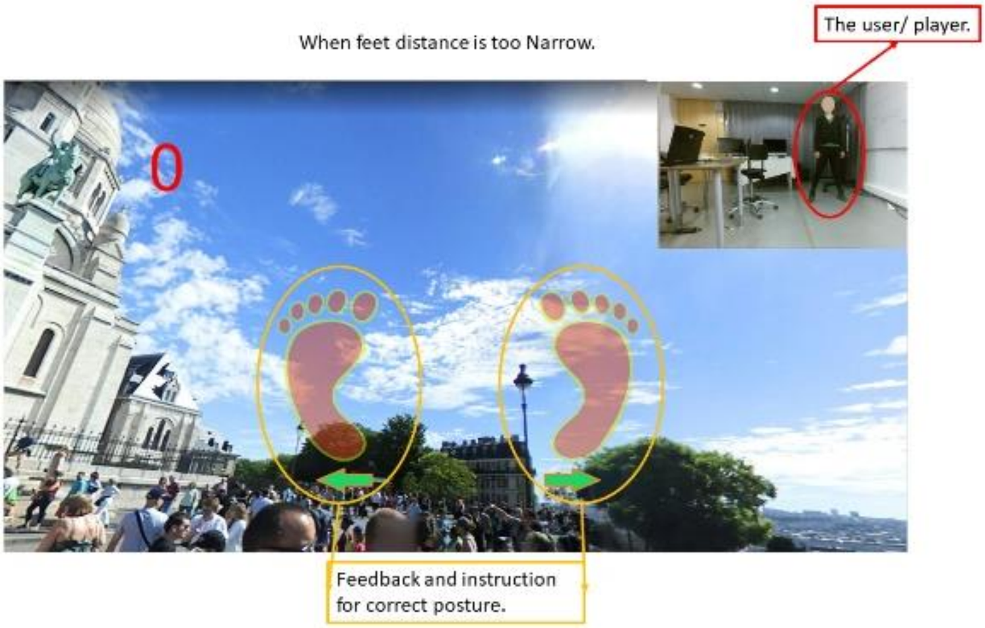


Figure 32. Feedback for wide feet distance.

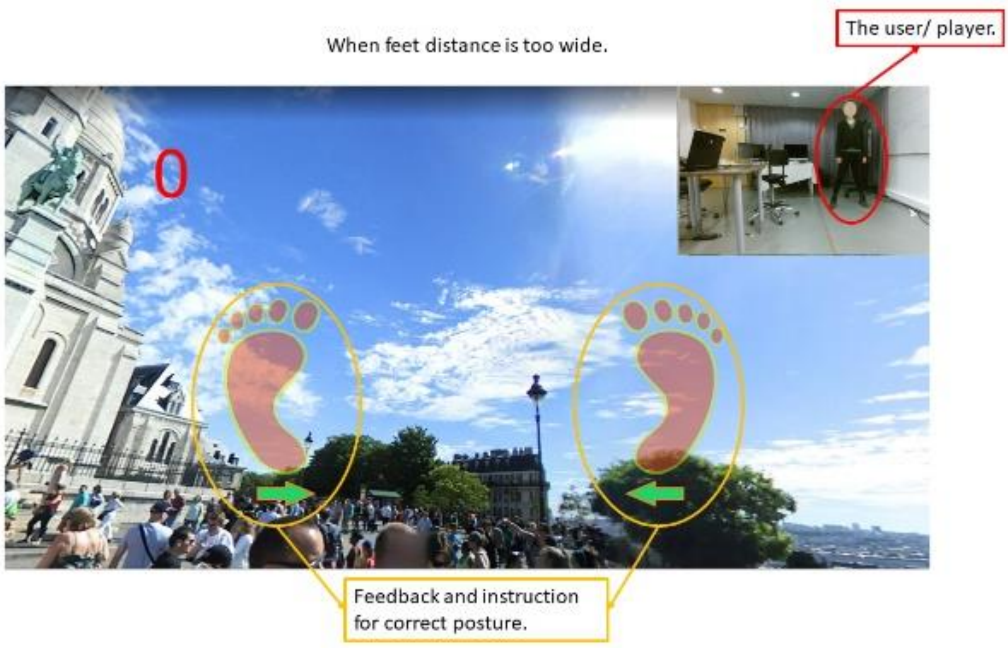


Figure 33. Feedback for narrow feet distance.

In Figure 31 and Figure 32, the feedback will appear only when the standing position is not good enough. The mechanism is explained as the previous Figure 23 and Figure 24.

Figure 34. Exercise complete scene.

As shown in Figure 34, the photo changes to a different angle. The views consist of several photos, similar to panorama in a regular smartphone camera. The view changes to the movement direction. When the user completes the required repetitions of the trunk movement, a message in the middle indicates the exercise completion.

6.2 Boardgame

For each loop as shown in the game flow, one of the users is represented in the board with an avatar. The avatar moves on the board with certain steps after completing one repetition of the exercise.

6.3 Minigame as the exercise

6.3.1 Exercise repetition

When one repetition of the exercise is completed correctly, the number on the top left of the screen will add 1 based on the previous value. When 5 times are completed, the user will be asked to perform the exercise to the other side.

6.3.2 Game feedback system

6.3.2.1 Feet distance

When the user stands on a wider or narrower feet distance beyond the fixed interval, this feedback displays feet symbols on the screen. The symbols are animated, when the distance between feet is too wide, the symbols will move inwards with blinking arrows showing the directions. Vice versa when the foot distance is too narrow.

6.4 Balance of the user in balance maps

A display of pixels corresponding to each pressure point in the balance mat is shown. This technique can demonstrate the user’s balance, then they can respond and adjust the postures if needed. Some screenshots of the game while a person was standing on the mat are captured, the below photos describe the different observations.

- The top part of the image means the front side and the bottom means the backside of the user.
- The pressure intensity goes from low at the green colour, higher at the yellow colour and highest at the red colour.
- The bigger the pressure value, the taller the cube.

The below Figures can demonstrate how a pressure map look like in the game.

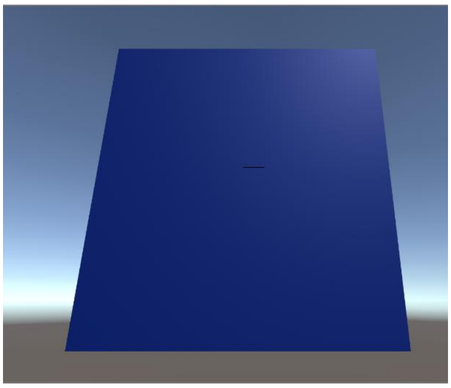


Figure 35. View without feet or pressure.

Figure 35 was the pressure map view that shows no foot is on the balance mat.

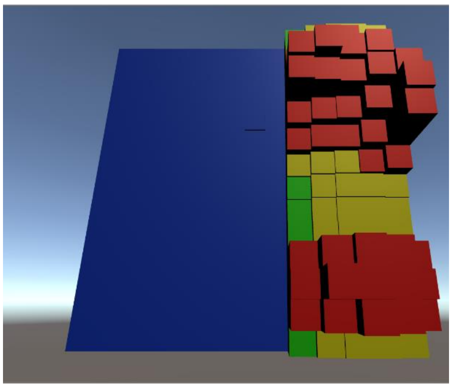


Figure 36. View of the right foot.

Figure 36 was the pressure map view of a right foot on the balance mat. This indicated the user was standing with only the right foot. The opposite can be a map view of the left foot on the balance mat.

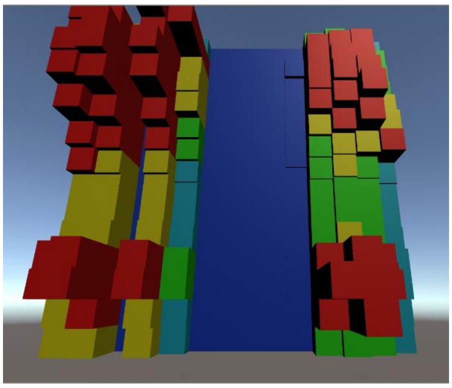


Figure 37. Leaning on the left foot.

In Figure 37, as the left cube heights were greater than the right, this pressure map view indicated leaning the left foot on the balance mat.

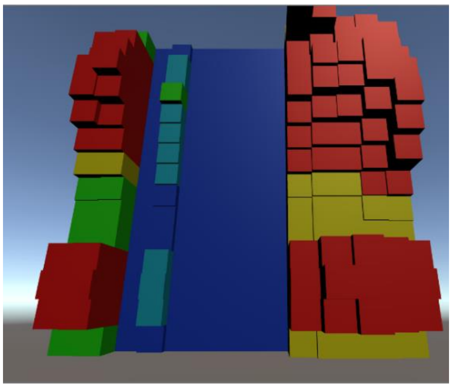


Figure 38. Leaning on the right foot.

In Figure 38, as the right cube heights were greater than the left, this pressure map view indicated leaning the right foot on the balance mat.

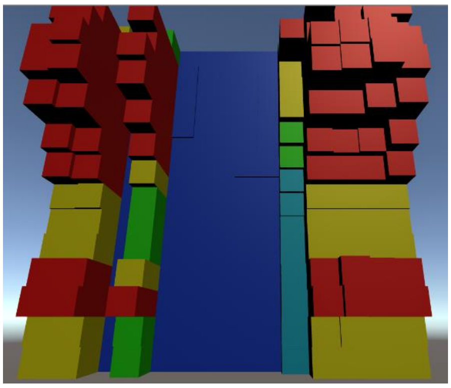


Figure 39. Leaning forwards.

In Figure 39, as the front cube heights and intensity were greater than the back on average, this pressure map view indicated leaning forwards on the balance mat.

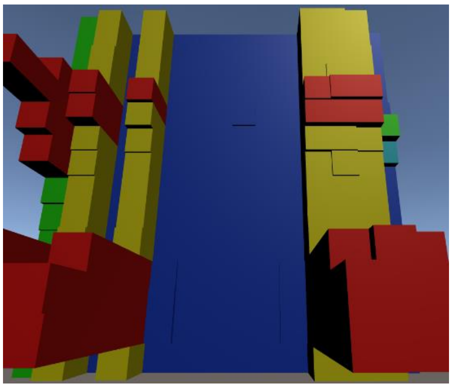


Figure 40. Leaning backwards.

In Figure 40, as the back cube heights and intensity were greater than the front on average, this pressure map view indicated leaning backwards on the balance mat.

6.5 User testing and feedback

On the 28 of November 2019, the team paid a visit to Attendo Paulus Sykehjem to test out the game with elderly people. Attendo is a nursing centre for the elderly in Oslo. There were two participants for testing and giving feedback. Below are the comments from the tested elderly and the nurses.

6.5.1 Questions about the game in general and the tested person

The following questions were asked after a general explanation of the game.

- Does the board design make sense, is it easy to understand?
- Do you have any comments on the colours, contrast or size of certain objects?
- Are you physically and cognitively able to play the game?

The answers to these questions are found in the next subparagraph.

6.5.2 User feedback

In total two elderly people participated in the game test, they will be referenced as the first and second tested. Both of the tested were male, between 75 and 90 years old and had a bad physical condition, meaning that they couldn't walk without assistance.

The first tested said the following about the design of the game:

> "It's difficult to see all the details of the game, it's blending".

He could see better when moved closer to the screen of the 52 inch TV. However, due to a closer distance to the screen and Kinect, he was no longer completely in the view of the Kinect (in line with the screen). After some adjustments and explanation about the game, the first tested understood the game. Unfortunately, he had to leave and relax more before being able to try the full game.

For the second tested, he had difficulties in understanding the concept of the game at first. He was placed in front of the Kinect and perform the T-pose. Since he was not able to stand up on his own without any form of support, Kinect was not able to recognize him fast and accurately. He had a chair, a physiotherapist behind him and a rollator in front of him. The initial T-pose recognition was successful with several trials. Then the tested went through to the introduction scene. He said the text was fine and understandable. The moving avatar animation, however, was not very clear to him. Especially the "top" view of the animation didn't make sense, "do I have to lay on the floor?" he responded.

The tested tried to move to the next scene by performing a T-pose movement. He got physically tired from standing up long while reading the exercise explanation. The tested had to sit down and rested for a short amount of time before moving on.

Lastly, the participant tried to play the first minigame. Again, he played incorrectly due to the cognitive and physical state. He could not perform one movement successfully. After playing, he commented that he liked the idea of the game of being able to travel to different places in the world. He was motived to play the game with the thematic. Even though he was not able to play the game fully, he said the AR game idea has potential.

One of the physiotherapists gave us her opinions too. She was quite enthusiastic and said the concept was good. But she missed some explanations within the game as they were a bit unclear visually at first. She also said the avatar animations were too small for her. She would like a real-world camera view against the avatars. In general, she liked this idea of gamifying the exercises with travel scenes. Geir Hansen, the Marketing and Communications Manager of Attendo, suggested to have different levels of difficulties in the game for different persons would be even better.

Figure 41. Testing the game.

Figure 41 are the photos of an elderly person playing the game in the elderly home.

7 Discussion

The discussion discusses the interpretation of results, the limitation of the project, a comparison with the previous project and to finish a future orientation of this project.

7.1 Interpretation of the result

For this game project, different goals were set. First, to develop this game using AR for elderly people. The game needs to be motivating, able to recognize if the user movements are correct and to give the balance feedback. Moreover, this game should host multiplayer.

As described in the result part of this project, the game contains 4 scenes for the main game. The concept originates from a board game, enhanced by travel theme.

By using the Kinect API, different user movements can be registered and determined as correct or not. The quality can be improved by using more models to feed the algorithm when given more resources and time.

Animated feedback was used. Unfortunately, one of the users was confused with the representation of this information about feet distance feedback during the user test. He said texts would have been better for him instead of animation.

The Kinect is not good at obtaining the correct skeleton view when there too many people in the view. So a multiplayer mode of playing together in the game is complicated and a higher level of development. Besides that, to develop a good algorithm to counter this problem is also time-consuming. This problem was tackled by creating the concept of making the users playing in turns. This removed the complexity of handling multiple players at the same time in front of the Kinect.

The balance mat was self-made as a proof of concept. The programme was to show that the possibility to integrate balance sensing into this game.

7.2 Limitations

The biggest limitations were time, coding skills, and knowledge about machine learning. To get a deep understanding of all aspects of developing a game was difficult. The team worked fairly for this game project with given resources.

There was a bug in the Microsoft SDK for Unity library. The problem arose when the code for the gesture recognition got to run a couple of times in a row. This bug was caused as the library not handling the thread-management well. The team tried multiple times and ways to fix this bug, but without success. The main reason was due to limitations of skills on this specific topic.

Another limitation came from the balance mat. Since the mat was just a proof of concept and was not completed in time, so the pressure map of the balance was not completely integrated in a usable way in the game demo. Meanwhile, the mat was too small and was not able to handle a person's weight.

7.3 Difficulties and problems faced

Version control:

This was one of the big problems that faced during the developing phase, the changes and updates of different versions in the software were prompted, which caused quite some problems for the team to work as a whole initially. This problem was later solved by using git under the combination of agile and scrum framework.

360° picture view:

One of the first ideas for the game was to use a 360° picture to immerse the user in a new place. Unfortunately, a proper source was not found. So, the team used simple images to create a panorama view.

The capacity for Kinect detection:

A big weakness of using the Kinect and the API related to the environment. The detection depended on the light or the background too much. When some elements were introduced like table, chair or like another human, then Kinect had some difficulties to recognize the wanted elements. Thus one player at a time during the game was set.

Using Avatars:

At the start of the project, to add an avatar to represent the user was considered. Finding a 3D model of an avatar for free is easy on the internet, but to link every joint anchor to 3D positioning is difficult. A special API like Vitruvius should have been used, but this would have consumed too much time. None of the team members had practical experience on a project using C# and Unity. So, the 2 avatars overlapped each other in some cases, and they were not linked to the user's movements. As a result, the avatar animation was just walking when a minigame is completed.

Feedback on the exercise speed:

A decision for exercise speed with the machine learning algorithm was not possible. So the speed feedback was only presented separately.

8 Conclusions

The testing with the elderly people shows AR games are likely to motivate the elderly to exercise more and provide “tailored” environments. By exercising more, elderly people will get stronger muscles and better balance. They can perform exercises by themselves without a physiotherapist attending. A game is attractive to the elderly when it is closely related to their daily life. Meanwhile, multiply difficulty levels should be designed for different users with different mobility levels. The age and physical conditions of the users must be considered at the same time to achieve better results. Customizable games according to prescriptions from physiotherapists would be optimal.

In addition, if more parameters such as muscle strength, heartbeats per minute, social activeness can be added, then to monitor and have an overall health condition of the users would be easier. Thus, one physiotherapist could take care of more elderly people who need help or assistance. In the meantime, families of the elderly and themselves can also know their health conditions. With knowing the basic health conditions, physiotherapists can design and implement better methods to contribute to better overall health conditions for elderly people.

References

[1] wikipedia, “Software development kit,” [Online]. Available: https://en.wikipedia.org/wiki/Software_development_kit.

[2] A. J. Campbell, “To prevent falls in older adults,” *OTAGO Exercise Programme,* pp. https://hfwcny.org/hfwcny-content/uploads/Otago_Exercise_Programme-Revised.pdf, March 2003.

[3] J. A. Campell, “To prevent falls in older adults,” *OTAGO Exercise Program,* p. 6, March 2003.

[4] A. W. d. Vries, G. Faber, I. Jonkers, J. H. Van Dieen and S. M. Verschueren, “Gait & Posture,” *Virtual reality balance training for elderly: Similar skiing games elicit,* p. 111, 2018.

[5] F. Levy, G. Rautureau, O. Komano, B. Millet, R. Jouvent and P. Leboucher, “Neuropsychiatric Disease and Treatment,” *Fear of falling: efficacy of virtual reality associated with serious games in elderly people,* 2016.

[6] W.-M. Hseih, C.-C. Chen, S.-C. Wang, S.-Y. Tan, Y.-S. Hwang, S.-C. Chen, J.-S. Lai and Y.-L. Chen, “Technology and Health Care,” *Virtual reality system based on Kinect for the,* 2014.

[7] O. “Otago Exercise Programme,” [Online]. Available: https://www.physio-pedia.com/Otago_Exercise_Programme.

[8] “Viarama home,” [Online]. Available: https://viarama.co.uk/.

[9] ®. N. © 2019 Pokémon. TM, “Pokémon GO - video games & apps,” 2019. [Online]. Available: https://www.pokemon.com/us/pokemon-video-games/pokemon-go/.

[10] Echopixel, “Echopixel home,” [Online]. Available: https://echopixeltech.com/index.html.

[11] E. d. “The smart rollator,” [Online]. Available: https://eggsdesign.com/work/case/the-smart-rollator.

[12] Microsoft, “Kinect for Windows SDK 2.0 [software],” 21 10 2014. [Online]. Available: https://www.microsoft.com/en-us/download/details.aspx?id=44561.

[13] O-Mat, “Arduino code,” [Online]. Available: https://content.instructables.com/ORIG/F0L/3PA5/IB4AVF15/F0L3PA5IB4AVF15.jpg?auto=webp&frame=1&width=1024&fit=bounds&md=9493f9a0f09c285d3096ee723d1ee9e9.

[14] S. Cogswell, “Github,” 21 08 2013. [Online]. Available: https://github.com/scogswell/ArduinoSerialCommand.

[15] Tekscan, “Body Pressure Measurement System (BPMS) - Research,” 2019. [Online]. Available: https://www.tekscan.com/products-solutions/systems/body-pressure-measurement-system-bpms-research. [Accessed 21 November 2019].

[16] Annika, Jeremy, Michael and Lars, “Survey,” October 2019. [Online]. Available: https://drive.google.com/open?id=15PszZUgoQrc7f9HWhAJ9M9rJZ5KcBPr6.

Gitlab repository for the game

Gitlab repository (code of the game): link.

All the code, Kinect related programming and Unity assets, visual effects can be found under the repository folder on Gitlab. One should be able to replicate the entire game by copy the repository to his/ her PC. Note that the correct hardware and its driver has to be installed before running the game.

Survey statistics for elderly activities

There are 10 surveys in total, all the notes can be found by clicking this link.

In this survey "/" symbol represent 1 vote from the surveyed.

What is your age?

60 - 64:	65 - 69://	70 - 74://	75 - 79:////	80 - 84://

What is your gender?

Male: ///	Female: ///////

What is your weight:

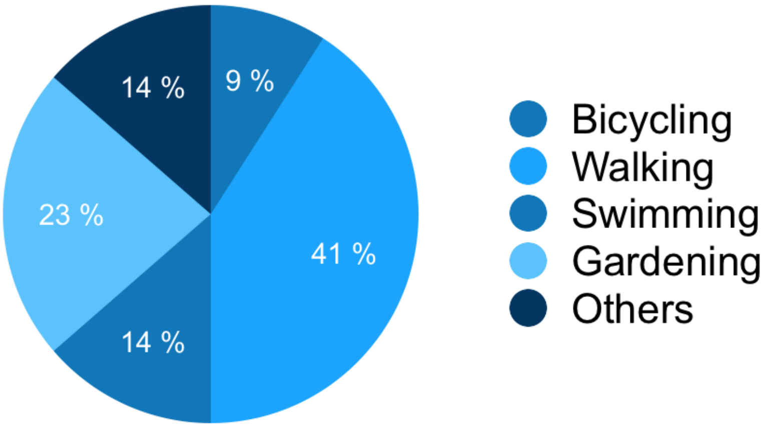
50 - 59 Kg:	60 - 69 Kg: ///	70 - 79 Kg: ///	80 - 89 Kg: ///	90 Kg --> : /

What is your height:

1,50 - 1,59 m:	1,60 - 169 m: ///////	1,70 - 1,79 m: //	1,80 - 1,89 m: /	1,90 m -->:

What exercises do you do in your daily life?

Bicycling: //	Walking: /////////	Running:	Swimming: ///	Gardening: /////	Fitness:	Other: Hiking Hunting Cleaning



Have you been to the physiotherapist last year and for what?

Yes: //// Injured shoulder Operation on the back Injured ankle	No: //////

The complaints you encounter in daily life activities?

- Not understanding technology.
- Getting slower in the physical movements.
- Knee issues, miss clicking of bigger fingers on Ipad.
- Not able to walk too long, hard time using the computer of bad eye-sight.
- Having problems with different kinds of technology, don't know how to get back to the previous page.
- Too little pressure in fingertips to push buttons
- Carry (medium) heavy objects
- Losing focus of balance after walking too long or chatting with other people.

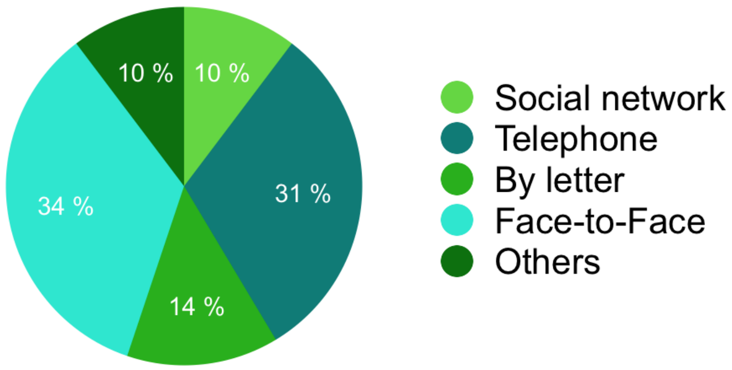
Do you have specific problems with using technology?

- I often need to zoom to see the screen clear.

- At first, it was hard to understand how some technology works.
- Large fingers that touch multiple buttons.
- Not downloading new updates because of changes that are not understandable so fast.
- I spend all of my life without these things so I will continue like that
- Not pressing the button enough because of low sensitivity in the fingertips.

How do you stay in contact with your social network?

Social network:	Telephone:	By letter:	Face - to - face:	Other:
///	/////////	////	//////////	Hunting meetings Tennis club Bowling ground area



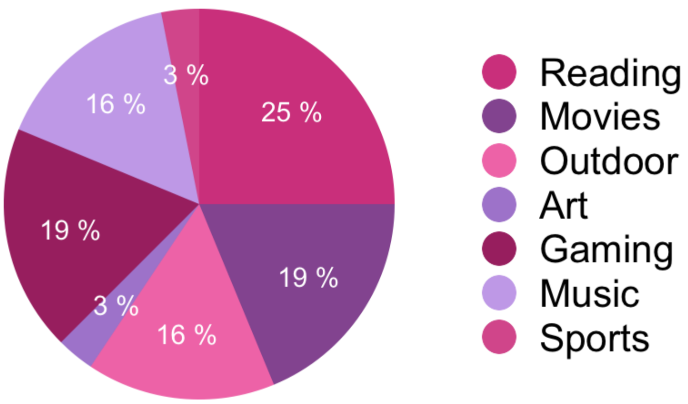
Where do you go to interact or socialise with other people?

- public events and activities
- churchly events
- occasional concerts and talks
- meeting with the neighbourhood
- When hunting
- Meet friends in the city
- Family dinners

- Lotto club
- The tennis club
- Still giving lectures in the university
- Cinema
- Diner
- Travel
- Church
- Choir
- In the apartment complex

Which of the following categories do you like the most?

Reading:	Movies:	Outdoor:	Gaming:	Music:	Art:	Sports:
////////	//////	/////	////	//////	/	/



Bill of materials for the balance mat

Velostat (datasheet)	
Copper tape (datasheet)	
Insulation fabric	
Ribbon wire (datasheet)	
Solder lead (datasheet)	
Breakaway headers (datasheet)	
Multiplexer (datasheet)	
Breadboard (datasheet)	
Arduino mini pro (datasheet)	
2 leds (datasheet)	
16 * 1k [ohm] Resistors 2* 220 [ohm] resistors	

Complete Arduino codes for the balance mat

```
//Mux control pins for analog signal (SIG_pin) default for arduino mini pro
const byte s0 = 13;
const byte s1 = 12;
const byte s2 = 11;
const byte s3 = 10;

//Mux control pins for Output signal (OUT_pin) default for arduino mini pro
const byte w0 = 9;
const byte w1 = 8;
const byte w2 = 7;
const byte w3 = 6;

//Mux in "SIG" pin default for arduino mini pro
const byte SIG_pin = 0;

//Mux out "SIG" pin default for arduino mini pro
const byte OUT_pin = 5;

//Row and Column pins default for arduino mini pro
const byte STATUS_pin = 3;
const byte COL_pin = 2;

const boolean muxChannel[16][4]={
    {0,0,0,0}, //channel 0
    {1,0,0,0}, //channel 1
    {0,1,0,0}, //channel 2
    {1,1,0,0}, //channel 3
    {0,0,1,0}, //channel 4
    {1,0,1,0}, //channel 5
    {0,1,1,0}, //channel 6
    {1,1,1,0}, //channel 7
    {0,0,0,1}, //channel 8
    {1,0,0,1}, //channel 9
    {0,1,0,1}, //channel 10
    {1,1,0,1}, //channel 11
    {0,0,1,1}, //channel 12
    {1,0,1,1}, //channel 13
    {0,1,1,1}, //channel 14
    {1,1,1,1}  //channel 15
  };


//incoming serial byte
int inByte = 0;

int valor = 0;               //variable for sending bytes to processing
int calibra[15][15];         //Calibration array for the min values of each od
the 225 sensors.
int minsensor=254;          //Variable for staring the min array
int multiplier = 254;
int pastmatrix[15][15];

void setup(){
```

```
pinMode(s0, OUTPUT);
pinMode(s1, OUTPUT);
pinMode(s2, OUTPUT);
pinMode(s3, OUTPUT);

pinMode(w0, OUTPUT);
pinMode(w1, OUTPUT);
pinMode(w2, OUTPUT);
pinMode(w3, OUTPUT);

pinMode(OUT_pin, OUTPUT);

pinMode(STATUS_pin, OUTPUT);
pinMode(COL_pin, OUTPUT);


digitalWrite(s0, LOW);
digitalWrite(s1, LOW);
digitalWrite(s2, LOW);
digitalWrite(s3, LOW);

digitalWrite(w0, LOW);
digitalWrite(w1, LOW);
digitalWrite(w2, LOW);
digitalWrite(w3, LOW);

digitalWrite(OUT_pin, HIGH);
digitalWrite(STATUS_pin, HIGH);
digitalWrite(COL_pin, HIGH);



Serial.begin(115200);

Serial.println("\n\Calibrating...\n");

// Full of 0's of initial matrix
for(byte j = 0; j < 15; j ++){
  writeMux(j);
  for(byte i = 0; i < 15; i ++)
    calibra[j][i] = 0;
}

// Calibration
for(byte k = 0; k < 50; k++){
  for(byte j = 0; j < 15; j ++){
    writeMux(j);
    for(byte i = 0; i < 15; i ++)
      calibra[j][i] = calibra[j][i] + readMux(i);
  }
}

//Print averages
for(byte j = 0; j < 15; j ++){
  writeMux(j);
  for(byte i = 0; i < 15; i ++){
    calibra[j][i] = calibra[j][i]/50;
    if(calibra[j][i] < minsensor)
```

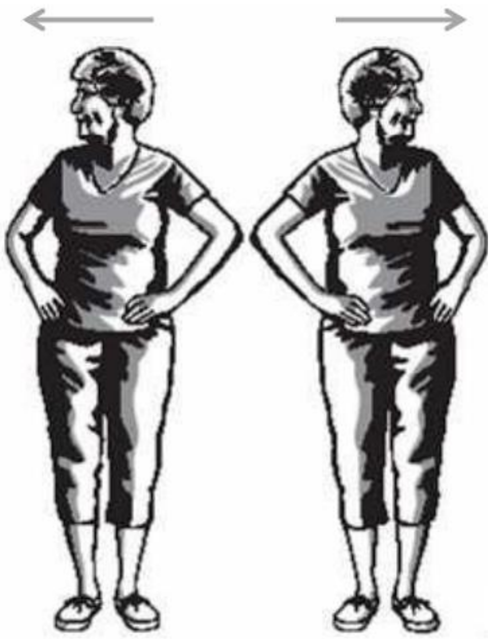
```
        minsensor = calibra[j][i];
      Serial.print(calibra[j][i]);
      Serial.print("\t");
    }
  Serial.println();
  }

  Serial.println();
  Serial.print("Minimum Value: ");
  Serial.println(minsensor);
  Serial.println();

  establishContact();

  digitalWrite(COL_pin, LOW);
}


void loop(){
  //Loop through and read all 16 values
  //Reports back Value at channel 6 is: 346
  if (Serial.available() > 0){
    inByte = Serial.read();

    if(inByte == 'A'){

      for(int j = 14; j >= 0; j--){
        writeMux(j);

        for(int i = 0; i < 15; i++){

          valor = readMux(i);

          //Saturation sensors
          int limsup = 450;
          if(valor > limsup)
            valor = limsup;

          if(valor < calibra[j][i])
            valor = calibra[j][i];

          valor = map(valor,minsensor, limsup,1,254);

          if(valor < 150)
            valor = 0;
          if(valor > 254)
            valor = 254;

          Serial.write(valor);
          digitalWrite(COL_pin,!digitalRead(COL_pin));
        }
      }
    }

  }
}


int readMux(byte channel){
```

```
  byte controlPin[] = {s0, s1, s2, s3};

  //loop through the 4 sig
  for(int i = 0; i < 4; i ++){
    digitalWrite(controlPin[i], muxChannel[channel][i]);
  }

  //read the value at the SIG pin
  int val = analogRead(SIG_pin);

  //return the value
  return val;
}

void writeMux(byte channel){
  byte controlPin[] = {w0, w1, w2, w3};

  //loop through the 4 sig
  for(byte i = 0; i < 4; i ++){
    digitalWrite(controlPin[i], muxChannel[channel][i]);
  }
}

void establishContact() {
  while (Serial.available() <= 0) {
    Serial.print('A');   // send a capital A
    delay(300);
  }
}
```

The trunk movement from the Otago exercise program

The trunk movement instructions:

- Stand up straight and place hands on hips.
- Keep hips still.
- Turn at a comfortably and as far as possible to the right, comfortably.
- Restore to the standing pose.
- Turn at a comfortably and as far as possible to the left, comfortably.
- Repeat five times to each side.



Video instruction for Otago: Otago Exercise Programme.

Work distribution

The entire project was teamwork, involving 3 students from different cultures and academic backgrounds. Members are Jim van Lienden (IT), Alexis Podikakis (Electrical), Kun Zhu (Electronics). Based on academic background, the work is roughly allocated to each member, the workload is scaled as from the least contribution at 1 to the most at 5. Details are listed as the below table:

Tasks	Jim van Lienden	Alexis Podikakis	Kun Zhu
Literature review	2	2	5
Project management	5	2	5
General coding	5	1	3
Pre-learning/ demo project	4	5	4
Kinect configuration	5	2	5
Machine learning video recording	5	1	3
Movement recognition	5	1	3
Game background and layout	3	5	2
Humanoid avatars	3	1	5
Feedback system	2	1	5
Version control	5	1	3
Game storyline	2	4	5
Balance pressure map view	5	1	1
Game scenes	3	5	5
User testing	4	3	5
Balance mat building and tutorial	0	0	5
Balance mat hardware configuration	0	0	5

The balance mat configuration and building tutorial section as in 4.3.1 and 4.3.2 are later added upon the original report, to meet requirements for this thesis.