Virtual Reality Game for Social Cue Detection Training

by

Ari Þórðarson

Thesis of 30 ECTS credits submitted to the School of Computer Science
at Reykjavík University in partial fulfillment
of the requirements for the degree of
Master of Science (M.Sc.) in Computer Science

June 2018

Examining Committee:

Hannes Högni Vilhjálmsson, Supervisor
Associate Professor, Reykjavík University, Iceland

Kristinn R. Thórisson, Examiner
Professor, Reykjavik University, Iceland

Berglind Sveinbjörnsdóttir, Examiner
Assistant Professor, Reykjavik University, Iceland

Virtual Reality Game for Social Cue Detection Training

Ari Þórðarson

June 2018

Abstract

The ability to recognize social cues and knowing how to approach people can be important in many social situations. Not everyone develops this ability to the level required by certain jobs or even for carrying out daily societal functions. For instance, some people with autism spectrum disorder have difficulty recognizing facial expressions. Training for this typically requires actual situations, which may put a limit on the amount and kind of training people may get. A simulated environment specifically created for this purpose could mitigate some of these limitations. This thesis presents a prototype virtual reality game intended for training people in detecting social cues and how to initiate contact with strangers. The goal is to provide training professionals with a proof of concept tool that brings together a number of advanced technologies into a single system for further development. The goal of a participant in the game is to sell something to virtual humans walking around in a public place. Only some of the virtual humans are interested in buying and that can be seen in their nonverbal behavior. The behavior of the characters is based on research on nonverbal communication. In particular, behavior correlated with rapport such as smiling, nodding and direct body orientation as well as behavior shown when trying to avoid interaction. The result is a fully functional prototype where the user can interact with virtual humans and complete a sales task using both voice and head gaze as input. The system demonstrates basic functionality and concept of a final system, opening up potential for further development of character behavior, virtual human appearance, and interaction naturalness.

Sýndarveruleikaleikur fyrir félagsþjálfun

Ari Þórðarson

júní 2018

Útdráttur

Hæfileikinn að bera kennsl á líkamstjáningu og hvernig maður hefur samskipti getur verið mikilvægur í margs konar félagslegum aðstæðum. Ekki allir komast á það stig í félagsfærni sem þarf fyrir ákveðin störf og jafnvel ekki fyrir dagleg mannleg samskipti. Til dæmis, á sumt fólk með einhverfu í erfiðleikum með að þekkja svipbrigði. Fyrir þjálfun í þessu, er oft þörf á alvöru aðstæðum, sem setja takmarkanir á magn og tegund af þjálfun sem fólk getur fengið. Sýndarumhverfi, sérstaklega gert fyrir þennan tilgang gæti minnkað þessar takmarkanir. Þessi ritgerð lýsir frumgerð fyrir sýndarveruleikaleik til þess að þjálfa fólk í að bera kennsl á líkamstjáningu og hvernig maður hefur samskipti við fólk. Markmiðið er að gefa sérfræðingum tæki á frumhugmyndastigi þar sem nokkur háþróuð tæknifyrirbæri eru tengd saman í eitt kerfi sem hægt er að þróa lengra. Markmið þáttakanda í leiknum er að selja eitthvað til sýndarmanneskja sem eru á gangi á almenningssvæði. Það eru bara sumir sem hafa áhuga á að kaupa og það sést á líkamstjáningu þeirra. Hegðun fólksins byggist á rannsóknum á líkamstjáningu. Einkum hegðun sem tengist því hvort einhverjum líkar vel við einhvern svo sem bros, að kinka kolli og snúningur á líkama og líka hegðun tengd því að komast hjá því að eiga samskipti. Útkoman er fullvirk frumgerð þar sem notandinn getur átt í samskiptum við sýndarmanneskjur og lokið söluverkefni með því að nota rödd og snúning á höfði sem inntak. Kerfið sýnir grunnvirkni og hugmynd að lokakerfi, með möguleikum fyrir meiri þróun á persónuhegðun, útliti á sýndarmanneskjum, og náttúrulegum samskiptun.

Virtual Reality Game for Social Cue Detection Training

Ari Þórðarson

Thesis of 30 ECTS credits submitted to the School of Computer Science
at Reykjavík University in partial fulfillment of
the requirements for the degree of
Master of Science (M.Sc.) in Computer Science

June 2018

Student:

....

Ari Þórðarson

Examining Committee:

....

Hannes Högni Vilhjálmsson

....

Kristinn R. Thórisson

....

Berglind Sveinbjörnsdóttir

....
date

....
Ari Þórðarson
Master of Science

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Imagine a scenario where you are required to interact with people in a way that is out of your comfort zone. The thought of it makes you nervous and you are not sure if you will be able to cope smoothly with the situation. What if you could practice this interaction in a safe way. What if you could practice interacting with people without interacting with people.

Lack of social skills can be detrimental. It can cause an individual to feel embarrassed in social situations leading to a fear of interacting with people. The prevalence of social phobia has been estimated to be 15.6% in the general population in Sweden [1]. A study in Germany estimated the prevalence in young adults to be 4.9% in males and 9.5% in females [2]. People with social phobia may avoid social situations leading to further stagnation of social skills. Another thing to note is that some peo-

Figure 1.1: Nonverbal communication in various settings. From pexels.com.

ple may have a poor natural ability in recognizing nonverbal signals. An example is poor performance of people with autism spectrum disorder (ASD) in recognizing facial expressions [3].

Technological advances in the past 30 years have made it possible to display interactive virtual characters on a screen. This creates the opportunity of social training applications where human users practice interaction with virtual humans. Individuals fearing social encounters might view meeting virtual humans less threatening than real encounters, allowing them to advance their social skills through a computer application.

Successful communication with people, depends not only on the words exchanged, but also on tone of voice, facial expressions, gestures, proximity and other nonverbal signals. An important part of increasing social skills is learning to communicate and read nonverbal signals.

In this project, a prototype of a virtual reality game for learning how to detect social cues and knowing how to initiate contact with strangers, was created. The goal is to provide training professionals with a proof of concept tool, that brings together a number of advanced techologies into a single comprehensive package, that can be further developed for different social skill training scenarios. The technologies that are brought together include a virtual reality headset with a microphone, animated characters, navigation, sensing and decision logic (a state machine and behavior trees), a virtual reality user interface and a 3D game environment. Creating a special interface, were professionals can configuring the virtual reality experience, is out of the scope of this project. However, the system is built with an eye towards further development. The goal of player in the game is to sell something to virtual characters. The virtual characters do not speak but they do display various nonverbal signals. In order to increase sales, the player has to spot which characters are actually interested or else time will be wasted trying to sell a product to characters that won't buy. The only way to know which characters are interested is by reading their nonverbal behavior. The player also has to know how to initiate contact with a character, i.e. knowing when and how to approach a character.

The focus was primarily technical. The research on the effectiveness of this method for training social skills is outside the scope of this thesis. It will however be addressed in a follow-up project.

The rest of this thesis is divided into the following chapters: In chapter 2 background work is discussed, i.e similar applications, reasearch on nonverbal communication and techology for creating virtual humans. Chapter 3 presents the work that was done in the project, i.e. the behavior model of the characters, the software itself and how it was implemented. Then chapter 4 presents and discusses some test results. Finally, the project is summarized and future work is discussed in chapter 5.

Chapter 2

Background

In this section, various training applications with virtual humans will be discussed as well as research on nonverbal communication. Finally, technologies that have been used for creating virtual characters such as animation, and artificial intelligence techniques for creating character behavior will be discussed.

2.1 Similar Applications

A number of virtual reality applications exist for teaching social skills. Among these are applications with the following uses:

- Improving social skills of people with autism spectrum disorder (ASD).
- Treatment of social phobia.
- Teaching interpersonal skills to professionals in fields such as health care, military and law enforcement.
- Teaching manners and communications in a foreign culture.

Several studies exist on the use of some of these applications.

The feasibility of using virtual environments for improving social skills in people with high functioning autism has been studied [4]. Eight young adults with high functioning autism went through a 10 session training program lasting for 5 weeks. The application used was based on the *Second Life* virtual environment software. Special tests were conducted before and after the training. The tests measured social skills in three different areas which were emotion recognition, the ability to infer thoughts of others (theory of mind) and conversational skills. A significant performance increase was found in emotion recognition tests and also in a test for the theory of mind. Some very positive comments were also made by the subjects about the program and among those comments were that anyone could benefit from this type of training, not just those with Asperger's.

Using virtual characters for job interview training in adults with ASD has also been studied [5]. The software used in the study has a graphical user interface where users engage in a conversation with a virtual character. The character is a human resource representative called Molly. Molly can ask many different questions. The answers provided by the users, can increase or damage their rapport with Molly. Study subjects, trained with the software found it easy to use, enjoyable and felt more prepared for real job interviews.

The same software was used in a study, where health care professionals where trained in alcohol screening and intervention [6]. The application had a virtual patient that users could engage in conversations with. The system was built in a way that allowed the virtual patient to respond in multiple different ways, making each practice session unique. According to 14 members of an expert panel, using the system felt like entering an actual room where they were interviewing a real patient. Results from the study showed a significant skill increase in those using the application.

The use of virtual reality to treat social phobia has also been studied [7]. Thirty six participants were either assigned to traditional cognitive behavior therapy or virtual reality therapy. The virtual environments used, focused on four different social anxiety situations: Performance (the fear of public speaking), intimacy (the fear of connecting with others), scrutiny (the fear of acting while being observed), assertiveness (the fear of protecting own interests). The results showed that the difference in improvements using virtual reality therapy versus cognitive behavior therapy where not much.

In anothter study, the feasibility of using virtual reality for treating the fear of public speaking [8], was investigated. Students in a speaking course went through a virtual reality therapy program. The program was divided into four sessions and each session was different from the others. The program made use of a head-mounted-display with head-tracking. Results from the study indicate that virtual reality can be used for treating the fear of public speaking.

Many organizations such as military and law enforcement entities require expensive training for their personnel. One of the reasons for the development of virtual reality training applications is the cost of role playing exercises with real actors and real sets. The National Training Center (NTC) for the United States Military has used props, actors, make up and set designers to create realistic military exercises. In one instance more than 1700 role players were used [9]. Other than the cost, virtual reality may have some other advantages over traditional role playing training methods:

- Lower cost of travelling to training location.
- Easier to standardize the training.
- Scheduling of training sessions might be more flexible.
- In many situations virtual reality may be preferred because of safety reasons.
- Quick customization of characters, environment etc.

JUST-TALK is an application that has been used for training law enforcement personnel [10]. It uses virtual reality and natural language technologies. A majority of the students using the application found it easy to use, said that it enhanced their learning and rated it better than or comparable to discussion, lectures and role-play.

The Tactical Language and Culture Training System (TLCTS) [11] is a video game like application that uses virtual humans to teach people communication skills in a foreign culture. It uses artificial intelligence for various things such as speech recognition and for controlling the behavior of characters. It has been used by the US Military to teach the Arabic language and communication skills needed in Iraq. Participants in a study said that the system was better than classroom language instruction [12]. An evaluation study showed that 78% of participants who had previously been to Iraq felt they had reached a functional ability in Arabic after 50 hours of training. However, only 22% of those who had not been to Iraq felt so [11]. The participants that had

Behavior	Description	Possible function
Gaze	Movement of eyes, head and body indicating where a person is looking.	To put emphasis on a word in a conversation [15].
Head movement	Head movements not related to gaze such as a headtoss, nodding and changing orientation of head.	Greeting someone with a headtoss [16].
Facial expressions	Things such as eye blinking, mouth movements, smiling and other expressions related to intent and emotional state.	Using facial muscles to express an emotion [17].
Hand gestures	Hand movements such as pointing at something, moving hands to the beat of speech or positioning hand so that it represents something in speech.	Moving hand upwards while talking about something going up [18].

Table 2.1: A few categories of nonverbal communication.

previously been to Iraq also gave the software a higher rating. [11] suggest the difference between those who had already been to Iraq and the others, might be because those who had been there understood the importance of this training.

To summarize, virtual characters can be used in a great variety of training and therapy applications. Training applications include social training applications for people with ASD, language and culture training and training for various professionals. Among therapy applications, are applications for treating social phobia.

2.2 Nonverbal Communication

The behavior of the virtual humans in this project is based upon research on nonverbal communication [13] [14]. An example on how important this type of communication is for successful communication is that various non verbal signals such as glancing happen at certain times in a conversation between people [15], e.g. when a speaker wants to put emphasis on something. Some common categories of nonverbal communication can be seen in table 2.1.

The type of human behavior that is especially important for this project is behavior that is associated with rapport and behavior indicating whether a person is interested in further interaction with another person.

2.2.1 Rapport

An important concept for those who want to establish a relationship with someone is rapport. Rapport has been described as not being a personality trait but something that people establish with one another [13]. They suggest a model of rapport, composed of three parts: Mutual attentiveness, positivity and coordination [13]. Two people being attentive to each other is defined as their attention being focused on each other. Positivity is defined as the interaction between the people being friendly. Coordination is defined as the individuals behaving in syncrony with each other as they were one person. They also suggest that positivity is more important in the beginning of a relationship, e.g. among people that just met each other while coordination is more important in the long term. The study suggests the following nonverbal behaviors as indicators of positivity.

- Smiling.
- Head nodding.
- Forward lean of torso.
- Direct body orientation.

These behaviors can be shown by a virtual character to show high enthusiasm, approval or interest in buying something in a situation where they are being approached by a stranger in a public place.

2.2.2 Avoiding Interactions with Strangers

It is important to note some behaviors that people show when they do not want to be approached by someone. When walking out in public, people show certain signals when they get close to a stranger and do not want to engage in a further social interactions with that stranger. Some signals that people show to avoid interaction with a stranger in public have been observed [14]. Some of these include:

- Lip compression (lips pressed tightly together and rolled inward).
- Lip bite (upper teeth contacting lower lip).
- Tongue show (tongue slightly visible between the lips).
- Tongue in cheek (tongue pressing against the inside of the cheek so that the cheek is bulging out).
- Gaze avoidance.

Implications of this work is that virtual humans can be designed to display these behaviors when they want to avoid interacting with someone.

2.2.3 Nonverbal Communication in Virtual Characters

Significant effort has been put into making virtual characters display realistic nonverbal behavior. There are also some computer applications that make use of the ability of humans to read and display nonverbal signals.

An *embodied conversational agent* (ECA) is a virtual character that can communicate verbally as well as nonverbaly with a human user. An early example of this was the virtual character Gandalf [19]. Gandalf used a microphone for listening to the user. An eye tracker and a body suit was used for sensing the eye and body movements of the user. Gandalf could move eyebrows in relation to what was happening in a conversation, look at an area of interest, show appropriate facial expressions and was capable of conversational turn taking behavior. An important thing to note, is that the main goal here was not realism but rather to allow a human user to interact with a computer in a multimodal way. That is, taking advantage of the natural communication abilites of humans.

Another similar example was the ECA Rea [20]. Rea acted as a real estate agent that could engage in a conversation with a human user. Rea spoke and displayed nonverbal messages and she could also understand the user's speech and understand the nonverbal behavior of the user. Rea used verbal and nonverbal messages to initiate and terminate conversations and for conversational turn taking (such as looking away while planning her response to the user). Rea was also capaple of displaying gestures such as pointing to objects of interest and could provide feedback such as head nodding in response to the users input.

Idle gaze has been studied by Cafaro et al. (2009)[21] and data from the studies used for animating virtual characters. Two studied were performed for the project where people where filmed in public places. Some key observations were:

- When waiting at a bus terminal, certain subjects made short glances throughout the study, while others gazed at specific targets for longer time periods throughout the study.
- While walking down a street, subjects often looked to the ground and that is the first choice in gaze aversion behavior.
- Subjects often blinked their eyes, right before changing gaze direction.

An approach for generating social behavior for virtual humans based on theories on human territoriality has also been presented by Pedica et al. (2009) [22]. Social norms were used to affect the behavior of characters when they are in some kind of social gathering such as a circular group conversation. The social norms were based on constraints that influenced the behavior of the characters in some way such as making them keep personal distance or keep conversation equality.

In another study involving ECAs [23], the application Virtual Reykjavik was implemented. The purpose of the application was to help it's users learn Icelandic as a foreign language. Speech recognition enabled users to start conversations with the virtual characters. It was studied how people initiate conversations with strangers. The study involved non-native speakers of Icelandic walking up to strangers and asking for directions. The behavior was recorded with video cameras, then further analysed and used for designing a behavior model for the virtual characters in the application.

As can be seen in, a number of applications exist, with virtual characters whose behavior is based on research on nonverbal communication and in some cases, studies on nonverbal communication were conducted specially for the application. Information from these studies was not directly used in this project but that is something that might be useful to do in the future. E.g. for creating better idle gaze behavior.

2.3 Technology

Advances in computer hardware and software over the last 30 years make it possible to create animated characters on a computer screen that can respond to a human user in some way. Creating intelligent behavior for 3D graphics characters displayed on a screen is in some ways similar to creating intelligent behavior for robots. However, it is different in the way that the behavior of the screen characters often only needs to look good. An example of this difference would be that in order for a virtual character to walk, it is often enough to move its body while playing a walking animation. This is different from a robot with limbs that actually has to move its limbs in order to walk.

2.3.1 Animation

The limbs of a 3D graphics character can be moved using a skeleton like object [24], and this method is called *skeletal animation*. 3D characters are like other 3D graphics models made out of meshes, made out of polygons, made out of vertices that are the corner points of the polygons. A vertex in the mesh forming the skin of the character can be connected to a bone in a skeleton. When that bone moves the vertices connected to it will move. Sometimes a vertex is controlled by multiple bones and then there is a weight associated with each bone for that vertex. The process of creating the skeleton for a character is called *rigging*. The process of connecting the bones to the mesh is called *skinning*.

Once a character has been rigged and skinned it can be animated in a game engine like *Unity* by using special animation clips that define how the bones move over time. Another method is to animate characters procedurally, i.e. creating a special program or script that moves their bones.

An alternative to *using skeletal* animation is using *blend shapes* [25], also known as *blend keys*. *Blend shapes* can also be used alongside with *skeletal animation*. This techique has also been called *morph target animation* and *per-vertex animation*. *Blend shapes* are created by changing the shape of a mesh by moving some vertices and then giving that new shape an id or a key. When multiple *blend shapes* have been created for a mesh, that mesh can be morphed to those shapes by linearly interpolating between them. *Blend shapes* are useful for facial animation. Another way to do facial animation is to use a physically based model that simulates the skin, flesh and bones in a face [26].

2.3.2 Character Behavior

Behavior of virtual characters can be controlled via code written in some programming language, such as C++ or C#. However it is often usefull to use some well defined control architecture or behavior selection algorithm. For simple behavior models a *finite state machine* can be used. However, when the number of states grows it becomes difficult and error prone to create and modify a finite state machine. *Hierarchical finite state machine*, also known as *statecharts* [27] addresses the limitations of *finite state machines* by allowing a state to contain a state machine. This can reduce the number of transitions needed.

Another control architecture are *behavior trees*. They differ from state machines in a way that they are based on tasks instead of states. A *behavior tree* has a root task, which is typically composed of other tasks and possibly other *behavior trees*. Because each task is modular, the designer can focus on one task at a time. Therefore, very large *behavior trees* can easily be managed by the designer. In recent years, *behavior trees* have been used in the game industry [28]. They have also been used in robotics [29].

As the name suggests, a *behavior tree* has a tree like structure. An important feature of *behavior trees* is that a node in the tree may take several "ticks" to complete running.

This is because many nodes are not designed to complete running immediately but instead are designed to run over multiple frames in a game or application. When a node or branch in a tree is ticked it can return three different values: *Success*, *failure* or *running*. *Success* means that the task completed successfully. *Failure* means that the task failed. *Running* means that the task has not finished yet. There are three main categories of nodes: *Composite*, *decorator* and *leaf*. *Leaf* nodes do not contain any child nodes and their purpose can either be to perform some action or to check if some condition is true or false. *Composite* nodes have child nodes and they define how the child nodes are executed, e.g. in what order they are executed. Examples of *composite* nodes are *sequencer* nodes, which execute child nodes one after another or *parallel* nodes which execute child nodes in parallel. *Decorator* nodes modify the behavior of child nodes in some way, such as repeating them a certain amount of times or inverting the return value of a child node.

As virtual characters get more complex it is often useful to define some overall framework for controlling their behavior.

SAIBA (Situation, Agent, Intention, Behavior, Animation) is a framework for behavior generation composed of three stages [30]. The states are *intent planning*, *behavior planning* and *behavior realization*. Special markup languages are used for sending data from one stage to another. The *Functional Markup Language* (FML) is for sending data from the first stage to the second stage. With it, it is possible to specify intent without specifying physical behavior. The *Behavioral Markup Language* (BML) is an xml markup language for sending data from the second stage to the third stage. With it, it is possible to specify the physical behavior of a virtual human. BML is mainly focused on communicative behaviors such as speech and nonverbal communication.

ADAPT is an open source library for the *Unity* game engine for authoring the behavior of virtual humans [31]. It has a navigation component and animation system providing locomotion, gaze, gesture, reaching and sitting behavior as well as reaction to physical forces. *Behavior trees* are used for directing the behavior of the characters.

Of these technologies, the most important ones for this project is technology for animating 3D graphics characters as well as behavior selection techniques such as *behavior trees*.

Chapter 3

Approach and Implementation

In this section a behavior model for the virtual humans in the application will be presented. Then the implementation of the application will be described.

The application that was implemented, is a virtual reality game, with the purpose of training people to detect social cues. The application makes use of a virtual reality headset. The virtual environment is a mall like scene with virtual humans walking around. In order to sell something, the player has to speak to the characters. The sound uttered by the player is picked up by the microphone in the headset. The characters do not understand human language but they sense if the player makes a sound.

The player has to know the right time to approach a character. There is a limited window of opportunity when the characters can be approached.

Each character is either a buyer or a non-buyer. The player has no chance of selling anything to a character who is a non-buyer. If a character is a buyer then the player has a chance to sell.

When each character is generated there is a 50% chance of him either being a buyer or a non-buyer.

3.1 Model

This section is about the behavior model of the virtual humans in the application.

There are a few nonverbal signals shown when looking at the player, that are different between buyers and non-buyers. These differences are based on the following research on nonverbal communication. Smiling, head nodding and direct body orientation are suggested to be indicators of positivity [13]. Positivity is defined as friendliness and in the same study, suggested to be correlated with rapport. Another finding used here, is that when strangers want to avoid interaction, one expression that they may show is a compressed lips expression (lips pressed tightly together and rolled inwards) [14].

These nonverbal signals were used by the characters in the following way and it differs between buyers and non-buyers:

- Buyers smile while non-buyers show a compressed lips expression.
- While non-buyers will look at the player with their eyes, buyers will also rotate their body more towards the player when they are looking at him or her.

	Buyers	Non-buyers
Facial expressions	Smile	Compressed lips
Body orientation	Oriented more towards the player.	Orented less towards the player.
Nodding	Nodding occurs while standing in front of player.	Nodding never occurs.

Table 3.1: Difference in nonverbal behavior shown by buyers and non-buyers.

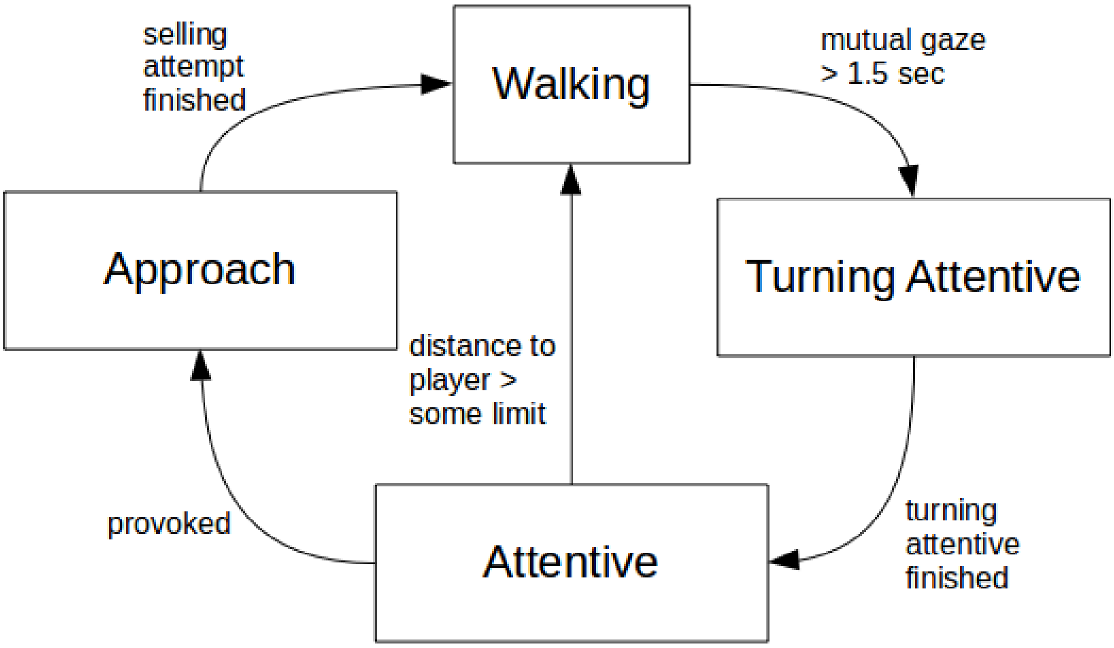

Figure 3.1: A high level state machine representing the behavior of the characters. Note that these are high level states. Each state can last for some time and certain behavior happens in each state. The *provoked* event happens when the character has been provoked by the player which happens if the player speaks while looking at the character.

- While the buyers are standing in front of the player they nod their head.

Table 3.1 summarizes the nonverbal behavior shown by buyers versus non-buyers.

The state machine in figure 3.1 has the high level states that the virtual humans can be in. They are described as being high level because many actions are done in each state. What happens inside such a high level state can result in the state finishing.

Description of the states:

- *Walking*: Character wanders around, gazing at random people or objects.
- *Turning Attentive*: The length of this state is always 0.5 seconds. Character gazes at player. If character is interested in buying he/she will rotate head and body towards player. If character is not interested in buying he/she will look at player with eyes but head and body will not be rotated towards player. If

Figure 3.2: A screenshot from the application. The player is in the process of selling to this woman. A white panel is shown, with 3 bits of information. In the lower left corner, the volume of the sound recorded by the microphone is shown. The lower right corner shows the earned money. When selling to a character, a bar showing the selling progress, appears in the top of the panel.

character is interested in buying he/she will smile. If not he/she will show a compressed lips expression.

- *Attentive*: Character will continue showing the right nonverbal signals (depending on whether he/she is a buyer or non-buyer) until distance to player exceeds a certain limit.

- *Approach*: Character walks towards player. Character continues to show the right nonverbal signals (depending on whether he/she is a buyer or a non-buyer). Additionally, if the character is a buyer he/she will nod head.

A character can only be approached by the player when he/she is in the *Attentive* state. The *provoked* transition is used when the player speaks to the character. Speaking requests from the player will be ignored in the *Walking* and *Turning Attentive* states.

3.2 User Interface

Figure 3.2 shows a screenshot from the application that was implemented. The application is a virtual reality game for social training. The game makes use of the HTC

Vive virtual reality headset, which has a built in microphone. The microphone is used to sense if the player is making an utterance. If the volume of the recorded sound goes over a certain threshold, the player is assumed to be speaking. In the game, the player is situated in a mall with people walking around.

There are in total three ways the player can affect the system.

- By rotating his or her head. The direction of the player's head affects the characters and the user interface.
- By changing the position of his or her head. This affects the assumed location of the player in the world and that has an affect on the characters and the user interface.
- By making an utterance. The sound recorded by a microphone in the headset is used as input. Only the volume matters. The volume directly affects the user interface and if it passes a certain threshold a special event is triggered that can affect the characters.

3.2.1 Information Panel

A white panel is displayed in front of the player, with 3 bits of information.

In the lower left corner of the panel, the volume of the sound recorded by the microphone is shown with a bar. If the volume exceeds the speaking threshold, the bar becomes more opaque. The purpose of the volume bar is to display the recorded volume, show if the speaking threshold is being reached and most importantly to indicate to the player that making an utterance actually influences the game.

The lower right corner of the panel shows the amount of money the player has earned. This tells the player how well he or she is doing in the game.

When the player is in the process of selling to a character, a bar showing the selling progress, appears in the top of the panel. The way the player tries to sell to a character is by looking at them while talking. If the character is in the *Attentive* state described in section 3.1 he or she will walk over to the player and stand in front of the player until the selling process is completed. The selling process takes a few seconds and the player does not need to do anything during it. Implementing an actual selling interaction model is outside of the scope of this thesis. The only purpose of the selling process is to waste the player's time. Because the selling process takes time, it is best for the player to avoid approaching characters that aren't interested in buying anything.

3.2.2 Character Arrow

An arrow appears over a character, the player is focusing on. This can be seen in figure 3.3. The appearance is based on the character's state. Because of this, it is possible for the player to see the state of the character by looking at the arrow. This is usefull for debugging purposes and perhaps for inexperienced players. However it might be detrimental for the training if the player can see the state of characters by looking at the arrow instead of the their gazing behavior and facial expressions. Because of this, the ability to disable the arrow might be useful. That is a feature that might be added in the future. Figure 3.4 shows the different types of arrows.

Figure 3.3: An arrow above a character. The arrow appears above the character that the player is focusing on. The character that the player is focusing on, is chosen based on where the player's head is turned. The player can only focus on characters that are within a certain distance. If more than one characters are in front of the player, the nearest character is considered to be the character that the player is focusing on. The style of the arrow in the picture shows that the character is in the *Approach* state. This character has a compressed lips expression and from that, it can be known that he will not buy anything.

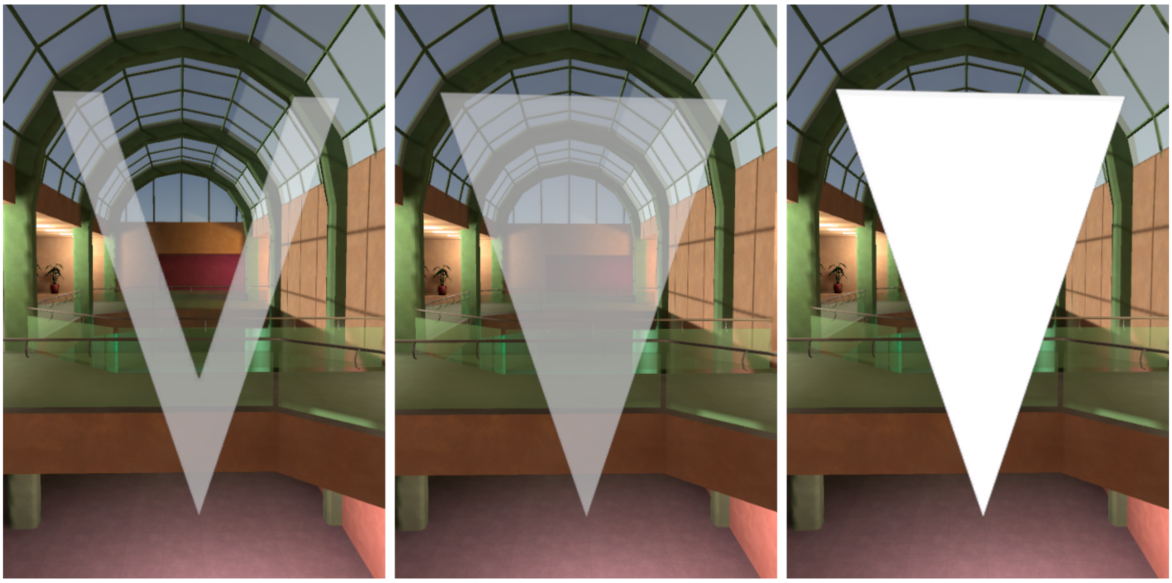

Figure 3.4: Different styles of arrows. Three types of arrows that are displayed above a character. The leftmost one is displayed for the *Walking* and *Turning Attentive* states. The one in the middle is displayed for the *Attentive* state and the one to the right is for the *Approach* state.

Note that the character in figure 3.3 is in the Approach state. The character that the player is focusing on, can be in any of the four states. What character is considered to be focused on by the player, depends on where the player's head is turned and the distance from the player to the character.

3.2.3 Facial Expressions

As has been described in section 3, each character is either a buyer or a non-buyer. The different expressions that buyers and non-buyers show when they see the player can be seen in figure 3.5

3.3 Implementation

The game was created using the game engine *Unity* [1]. The characters were created using the tool *MakeHuman* [2], which is a tool for making fully rigged 3D models of humans. A male and a female model were created with *MakeHuman* and modified in the 3D modeling creation suite *Blender* [3]. Among the things that were added in *Blender* were *shape keys* for facial expressions and changing the look of the characters.

Unity is a modern game engine that can be used to create a variety of both 2D and 3D games and other applications. It includes a scene editor where various objects called *GameObjects* can be put into a scene. *GameObjects* can have *components* attached to them that affect their behavior and appearance.

Unity has a powerful system for working with animated characters. Some key features are:

[1] https://unity3d.com
[2] http://www.makehumancommunity.org
[3] https://www.blender.org

Figure 3.5: Facial expressions of the characters. The two pictures in the upper part show smiling characters and the pictures in the lower part show characters with a compressed lips expression.

- Characters and animation clips can be imported and it is possible to create special state machines that play certain animations in certain states. Several layers of state machines can be created making it possible to add or blend animations together. A special feature called a *blend tree* can be used to blend two or more animations based on some value. This can be useful for, e.g. changing from a walking animation to a running animation based on a characters speed.

- Working with human characters is especially easy because there is a tool within *Unity* that is designed for configuring human character rigs and it is possible to get animation clips for humans from various sources.

Unity also has a powerful physics system. Some key features include:

- Collision detection. *Colliders* of various shapes can be created.

- Rigid body physics. In *Unity* a *Rigidbody component* can be attached to a game object. Typically, this will result in the *GameObject* falling because of gravity and colliding with *colliders*.

- Raycasting. It is possible to make an invisible straight line that starts in some point and ends somewhere else and check if the line collides with a collider. If that happens, the point of collision is known. This is useful for various things such as checking if an object is visible from a character.

Unity also has a navigation system that is useful for making characters move around a scene. With it, the shortest path to some location can be found so a character can walk over there along the path that was found. An object called *NavMesh* needs to be created for an environment, for this to work.

In order for the characters in the game to be able to navigate the environment the built in navigation system in *Unity* was first used. However in order to make the movement of the characters look more natural a special class called *AutoNavigator* was created. The class allows a game object to sense its closest environment using raycasting. Information from the raycasts is used to calculate the angular speed and velocity of the object. Figure 3.6 shows an example of rays, when this feature is being used.

The main idea behind this method, is that objects using it will move in the direction with the least resistance, or in other words more open space. To prevent character using this feature from bumping into each other a *collider* must be around the characters. A useful trick was discovered that can be used to prevent characters from rudely walking into other characters immediate walking zone. The trick was to put an elongated *collider* on characters like the one in figure 3.7.

A special system was created in order to animate the characters. A special class called *Choreographer* was created. The purpose of this class is to provide an interface for classes that handle parts of the characters animation such as gaze or locomotion. A special class called *ChoreographerManager* manages the choreographers. It provides the possibility to control in what order procedural animations are applied, which is important for *skeletal animation*. There are currently four classes being used, extending *Choreographer*. These are the choreographers in the order they are applied.

- *HumanLocomotion*: Calculates the speed of the character. The speed is used in a *blend tree* to slide between an idle and walking animation.

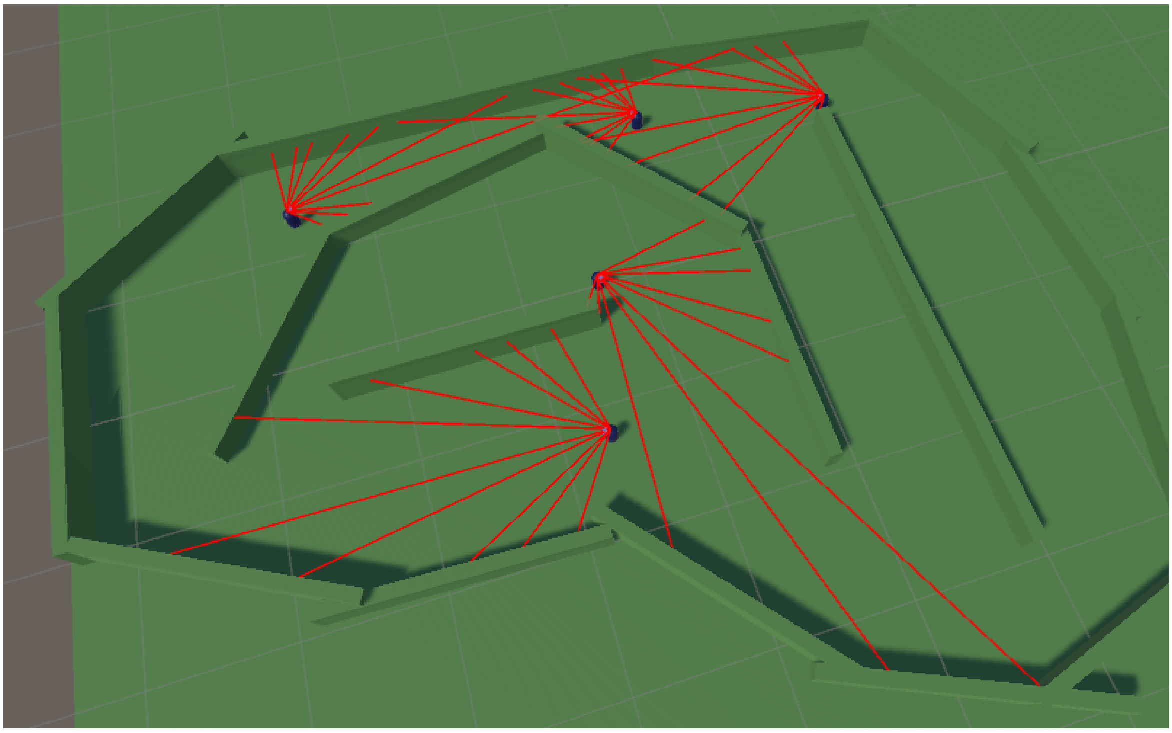

Figure 3.6: The shape of an environment sensed with raycasting. The blue capsule shaped agents in this picture cast rays (red lines) in a few directions in front of them. In this way the character can get a crude sense of the space in front of them.

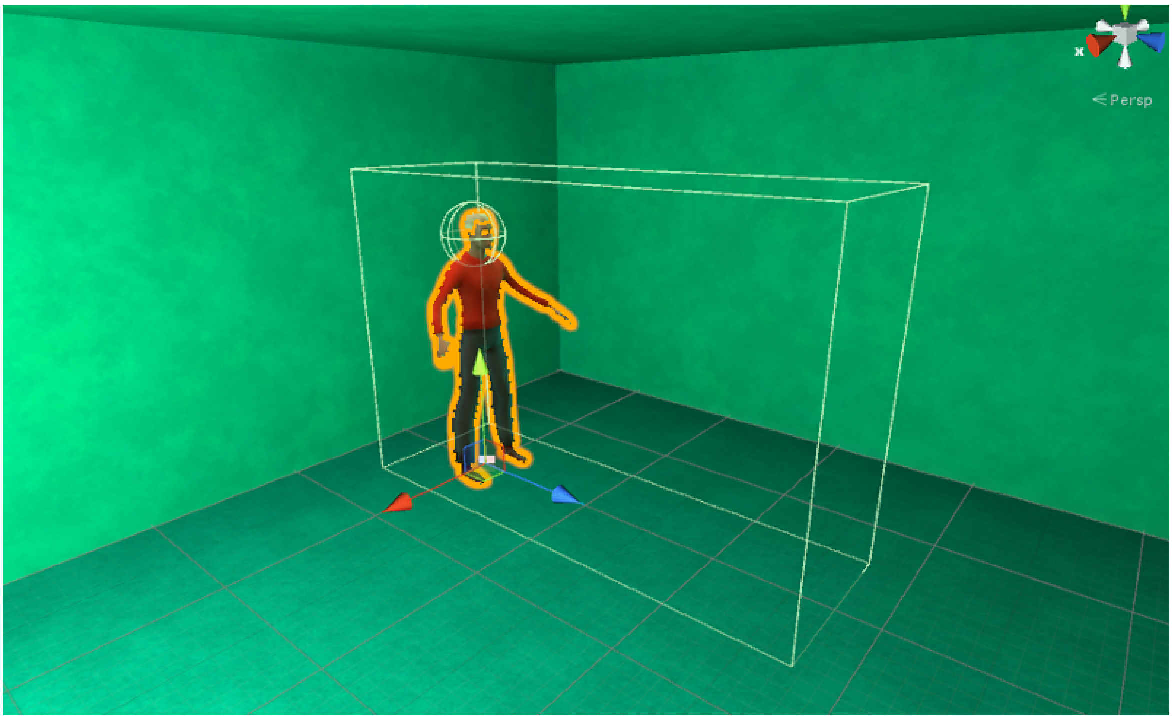

Figure 3.7: An elongated *box collider* around a character. Other character using the *AutoNavigator Component* will try to stay out of the way of this character.

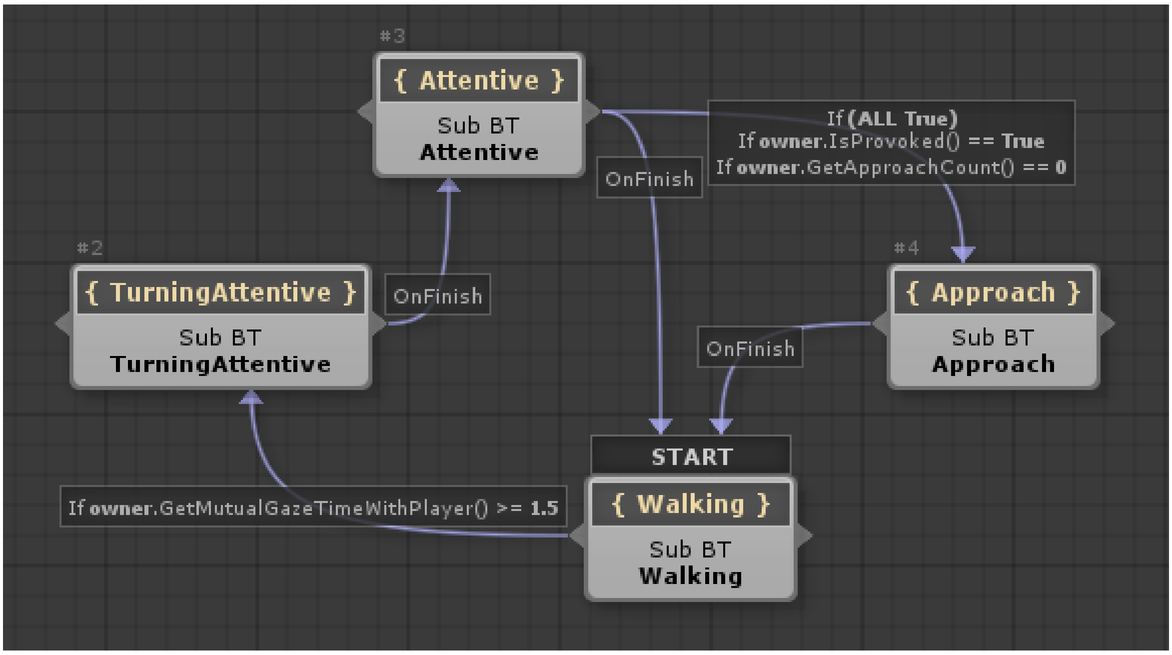

Figure 3.8: The *state machine* for the character behavior. The picture is from the *NodeCanvas* graphical node editor. In each state a special *behavior tree* is called. *Behavior trees* can also be created in the *NodeCanvas* graphical node editor.

- *HumanGaze*: Controls where the character appears to be looking. Rotates the head, neck, torso and eyes. It is possible to control how much each body part rotates and there can be a lag in the body rotation. Because of the lag in the body and head rotation, it is possible to make small eye movements without moving the head or body.

- *HumanHead*: The purpose of this, is to provide animation for head that is not related to gaze. Currently, the only function provided here is the ability for the character to nod the head.

- *EyeBlinker*: Makes the character blink eyes at random times.

There is also a special class called *BlendShapeManager* that is used to set weights for different *blend shapes*. It is used when controlling facial expressions such as smiling. It is also used for customizing the appearance of the characters. The characters have many *blend shapes* on them. Some are used for animation and other for changing the appearance of characters. By setting blend shape weights and changing the color of the clothes, it is possible to randomly create many different looking characters.

In order to author the high level behavior of the characters the tool *NodeCanvas* [4] from *Paradox Notion* was used. *NodeCanvas* is a behavior authoring framework for *Unity*. With it, *hierarchical state machines* and *behavior trees* can be created in a graphical node editor.

Figure 3.8 shows a screenshot from the *NodeCanvas* node editor. The *state machine* in the figure is for the behavior of the characters. In each state a special *behavior tree* is called.

Figure 3.9 shows the *behavior tree* used in the state *TurningAttentive*. The node with the *START* label is a *sequencer* node, meaning that the three children of that

[4]http://nodecanvas.paradoxnotion.com

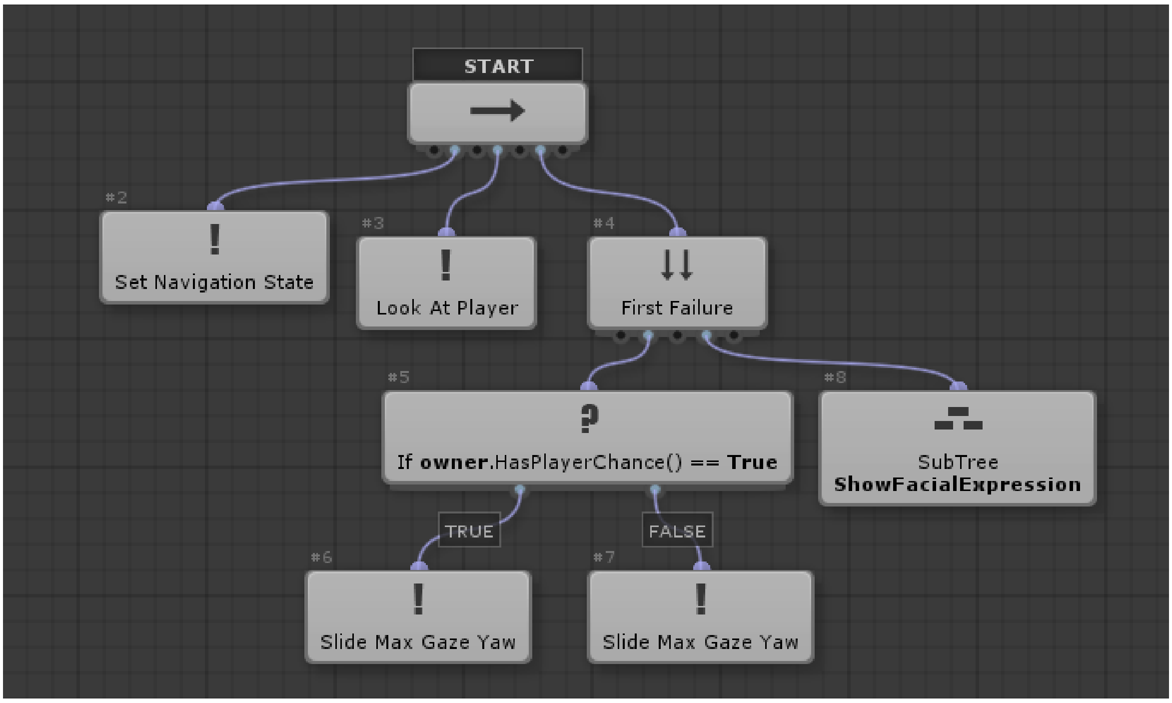

Figure 3.9: The *behavior tree* used in the *TurningAttentive* state. The root node is a *sequencer* node, which means that it's three child nodes are executed one after another.

node are executed one after another. The node with the two arrows with the text *First Failure*, is a *parallel* node, meaning that the two children of that node are executed at the same time. The node with the question mark works like an if-else statement, meaning that if the condition specified in the node is true, the left child is executed, else the right child is executed. The node with the text *ShowFacialExpression* calls another *behavior tree*. The other leaf nodes are all *action task* nodes that call special tasks implemented in the C# programming language.

Four other behavior trees are used and they are shown in figures 3.10, 3.11, 3.12, 3.13.

3.3.1 Extending the Prototype

The implemented application is made out of modular parts that can be replaced without doing to much modifications to other parts. This is important if the system will be used more in the future and modifications are necessary. However, in order to do this, one has to have some knowledge of the system.

It is possible to replace the character models with new ones as long as those are humanoid models. However, if the new models do not have the same blend shapes as the current ones, that needs to be considered because the application uses blend shapes for facial expressions and controlling the appearance of the characters. When switching to new character models there is also some configuring that needs to be done. In particular one needs to specify certain bones of the character used by *choreographer* classes.

If the environment (the mall scene) is to be changed, then the easiest way to do that might simply be to copy the whole scene, delete the mall from the scene and add some new 3D models to the scene.

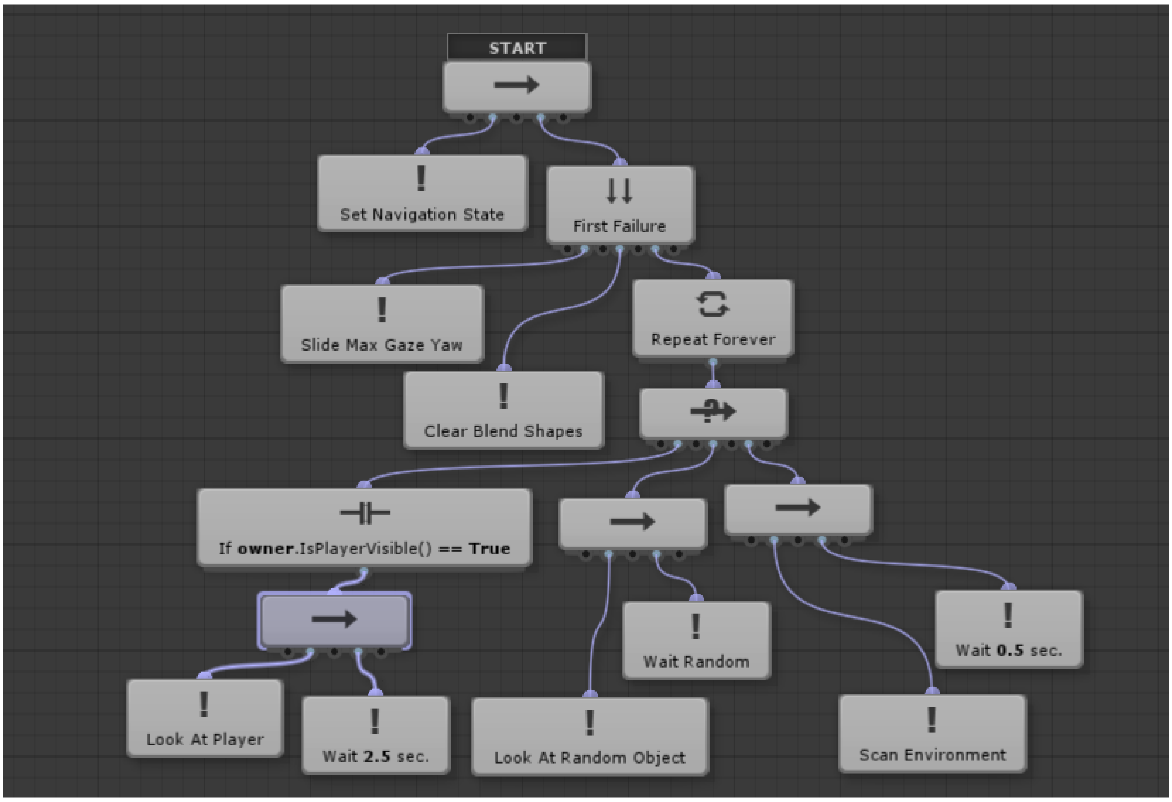

Figure 3.10: The *behavior tree* used in the *Walking* state. When this *behavior tree* is executed, the character will walk, look around and if possible, look at the player.

A class that is central to the character behavior is called *HumanAgent*. For each character there is an instance of this class attached to that character. All of the behavior trees used for the character behavior have access to this object. Through this *HumanAgent* object it is possible to get access to other objects used for controlling the character and those are objects used for animation such as the *choreographers* and a *BlendShapeManager* object as well as an *AutoNavigator* object used for navigation.

When using the *NodeCanvas* framework it is possible to implement special tasks by extending a class from the *NodeCanvas* framework called *ActionTask*. These tasks can then be used in a behavior tree. Multiple such tasks were implemented for this application.

It is fairly simple to add animation functionality to the *HumanAgent* class by adding new *choreographer* classes. However, it is more complicated to remove *choreographers* because some *NodeCanvas* tasks may assume those objects exits. There are also some *NodeCanvas* tasks that make use of the *BlendShapeManager* object so that needs to be considered if the *BlendShapeManager* object is removed from the *HumanAgent* class.

The characters use the *AutoNavigator* class for navigation. Not much code changes are required in order to skip using this method. Using this method for navigation can also be turned off at runtime if a character is to change his or her navigation techique.

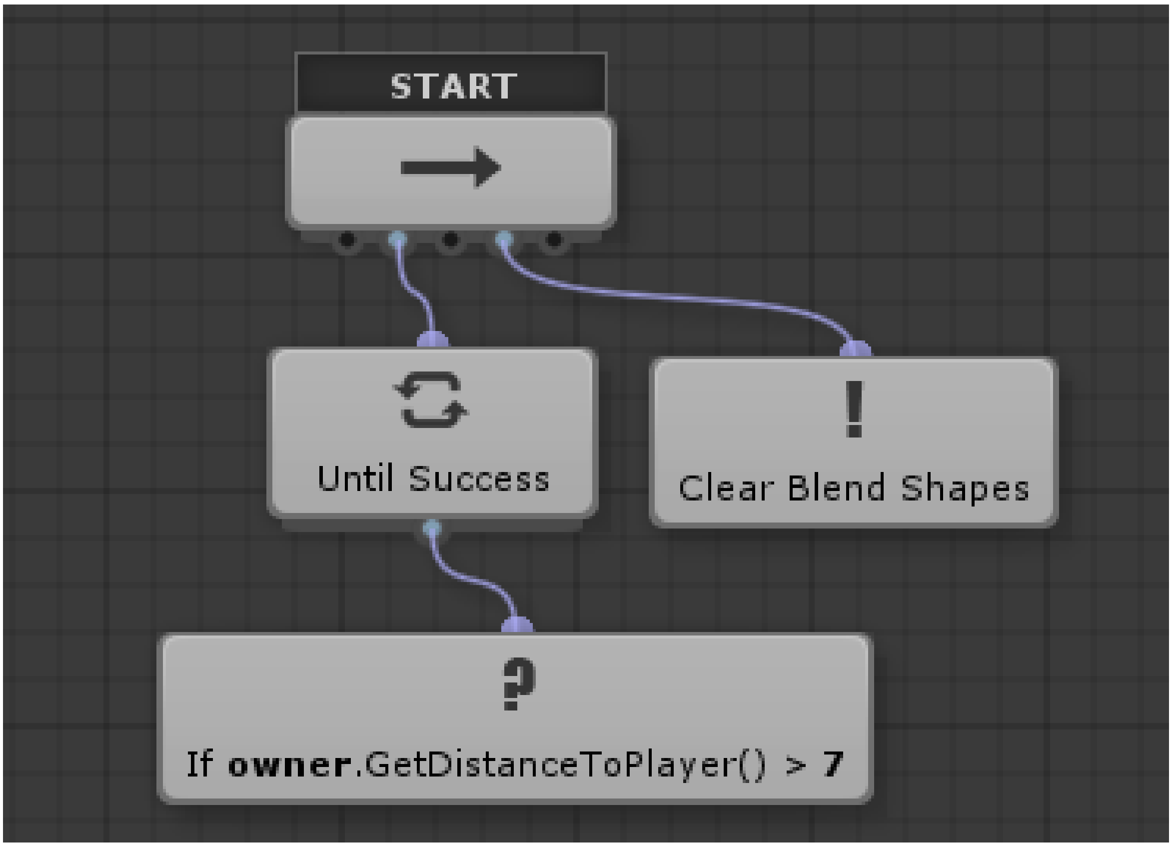

Figure 3.11: The *behavior tree* used in the *Attentive* state. As can be seen, this *behavior tree* does not do much. Most of the things that need to be set up, have been done in the *Turning Attentive* state that happens right before this *behavior tree* is called.

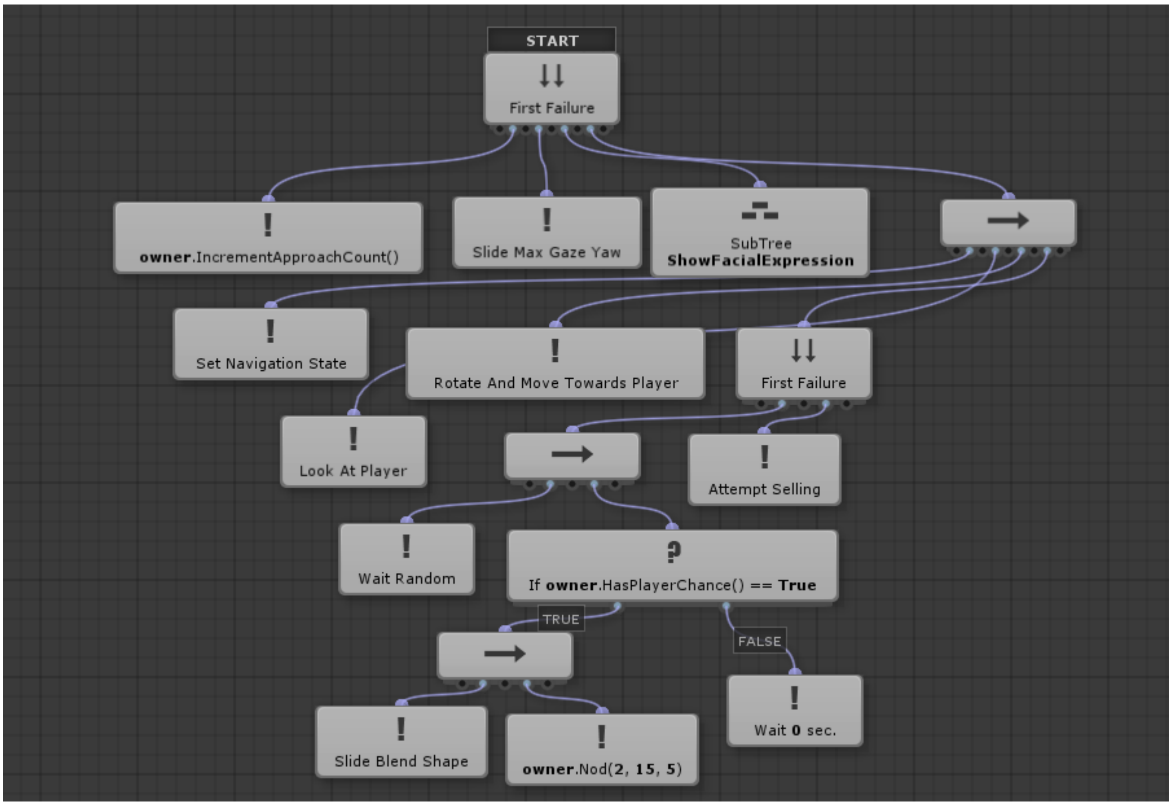

Figure 3.12: The *behavior tree* used in the *Approach* state. When this *behavior tree* is executed, the character will move towards the player and nod head while the selling process occurs.

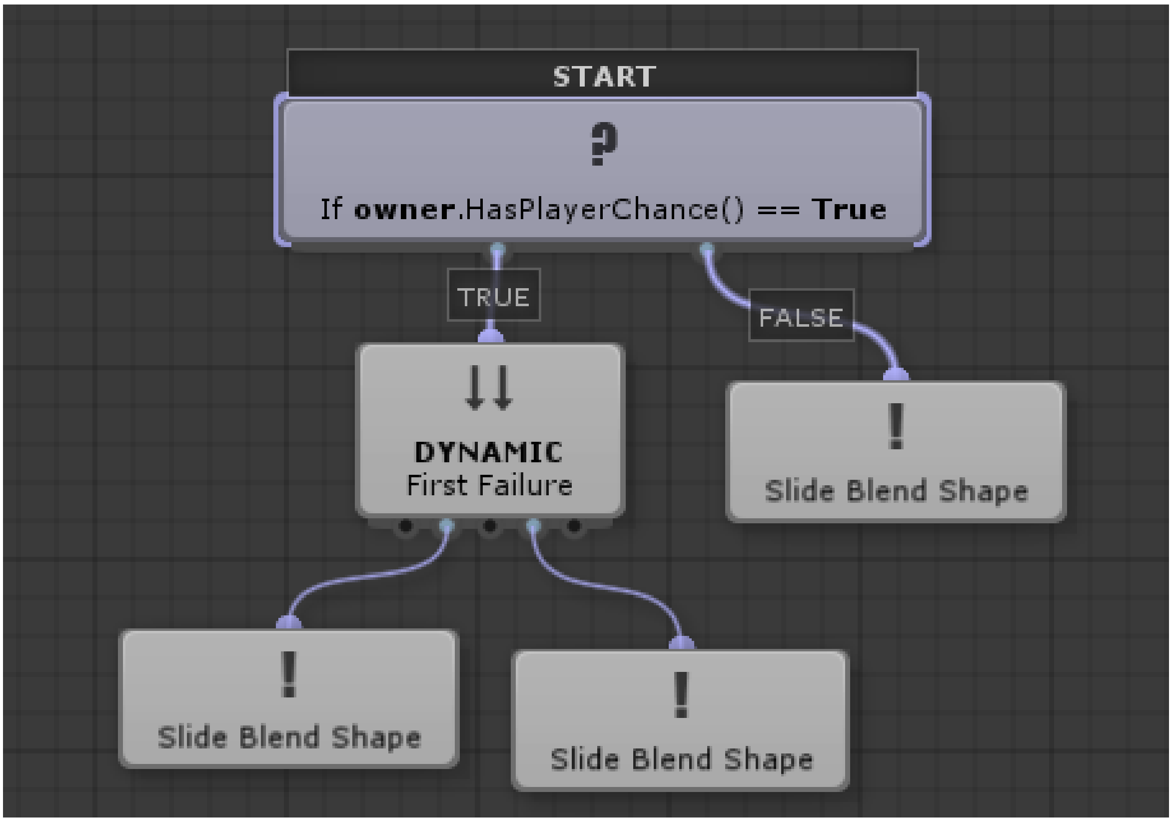

Figure 3.13: A *behavior tree* used for showing facial expressions. The method *HasPlayerChance* returns true if the character is a buyer. Different facial expressions are shown depending on whether the character is a buyer or a non-buyer. This behavior tree is called by the *Turning Attentive* and *Approach behavior trees*.

Chapter 4

Results and Evaluation

In this chapter, the resulting prototype game will be reviewed and feedback from test users will be presented. Finally, the chapter is concludes with further discussions.

4.1 Results

The result of the project is a virtual reality game with virtual characters walking around in a virtual environment and it is possible for the player fairly naturally initiate contact with those characters and sell to those who express interest. The application achieves the goal that was set forth at the start of the project, which was to create a prototype for a system that could possibly be used for social training. The application is fairly robust and the performance is good when using a modern computer with a modern graphics card. An intuitive user interface shows the player that it is possible to influence it by speaking, a bar is shown for the selling progress, an arrow is shown above characters that the player is focusing on and the earned money is displayed. The characters walk around without bumping into each other, show recognizable eye movements, head nodding, eye blinking and facial expressions when appropriate.

4.2 Usability Feedback

In order to get feedback on the usability of the application, a basic usability test was conducted. It has been shown that the majority of usability problems can be found with less than ten test participants [32]. In order to test the test procedure itself, one participant went through a pilot usability test. The data from that pilot test is not included in the results presented here. Data was also gathered during the usability test and some of it is included in appendix B.

Four people participated in usability tests. Each participant played the game for approximately 3 minutes. Some events in the game were logged to a file. Each participant was able to make at least three sales, which confirmed that they were able to finish the main task of the game.

Not all participants felt it was clear what actions could be taken by them in the game. Before each participant put on the virtual reality headset, they got oral instructions on what to do in the game. There were however no instructions or tips in the application itself. This is something that might be beneficial to add and it might even be useful to have some kind of an in game tutorial.

There were also some participants that felt that the behavior of the characters did not look natural. This might be because some elements of the behavior needed improvement, which will be discussed. However the majority of the participants agreed that it was clear from the behavior of the characters, which ones were ready to buy something or not. This might be because of the great difference in nonverbal behavior between buyers and non-buyers.

The following comments where also added by participants:

By anonymous participant: *Very realistic, however, the people floated a bit on the floor, that is, their feet did not stick to the floor. I was not trying to sell, but some reacted more strongly to me than others. Some seemed to ignore me.*

By another anonymous participant: *The sound that was played when a sale happened/failed is to loud, I was startled whenever it was played.*

By another anonymous participant: *Characters walk a bit close when they pass you.*

One participant also said that the arrow above the characters was to high up in the air. Another participant also said that the characters gazed to much.

No participants showed any signs of serious visible sickness during the usability test. However, the *Simulator Sickness Questionnaire* (SSQ) [33], was used to check if there were any changes in the simulator sickness condition of the participants after playing the game. The participants answered the questionnaire before and after playing the game. An increase in simulator sickness was found in two participants. This suggests that the application may cause simulator sickness in some people and that finding ways to prevent this would be beneficial, especially if longer sessions are expected. The simulator sickness scores for all participants before and after playing the game can be seen in appendix B.

4.3 Discussion

The goal of this project was to create a prototype of a social training application and this is what was achieved. The application implemented is a fully functional virtual reality game with virtual humans, were the player has to detect social cues in the characters and know how to approach them in order to succeed in the game. As the usability test showed, all participants were able to complete the main task of the game, which was to sell something to a character. This shows that the functionality brought together by all the underlying technologies is working. It is however, unknown wheather this method works for increasing social skills and testing that is out of the scope of this project.

As the comments from the usability tests suggest, a number of improvements could also be made to the application, including:

- Improve leg animation when walking. The movements of the feet are not perfectly in sync with the root motion of the characters. Currently, a *blend tree* is used to blend between idle and walking animation based on the speed of the characters. Possibly the animation only needs to be tweaked a bit. Another solution would be to enable root motion for the animations. If root motion is enabled, the root bone of the skeleton (the whole *GameObject*) will move in space when an animation is played. This will make the movement of the character perfectly in sync with the animation but this can however be inconvenient if the character is to be moved around with physics or with some script.

- The sounds that are played when a sale succedes or fails might be a bit unpleasant to the ear. In order to fix this they will need to be replaced by new sounds.
- The characters might walk a bit close to the player. In order to fix this it might be useful to implement some kind of proxemic system that is based on research on proxemics of humans.
- The position of the arrow above the characters might need tweaking.
- The characters might gaze a bit to much. Perhaps the gazing behavior needs to be tweaked so they don't look so much at the player and at each other. In order to make the gazing behavior more realistic, it might need to be based on data gathered on actual human gazing behavior while walking in public places or theories on human gazing behavior.

Chapter 5

Conclusions and Future Work

5.1 Summary of Contributions

A social training virtual reality game was created. The goal of the work was to create an application that could possibly be used for training social skills. In summary, these are the contributions:

- Design of an interaction model based on research on nonverbal communication. The virtual humans in the game are either buyers or non-buyers. The player can initiate contact with the characters by speaking while looking at them. The main difference in behavior between buyers and non-buyers is that buyers smile while non-buyers show a compressed lips expression. The buyer's body is also rotated more towards the player. The characters can be in a few different states and the player can only initiate contact with them if some mutual gaze has occurred.

- Technology for character behavior. A special navigation component was built for the movement of the characters, that uses *raycasting* for sensing the environment. Special classes handle animation related to specific behavior of the characters and they are gaze, locomotion, eye blinking and head movement not related to gaze. The gaze behavior allows moving the eyes independent of the head, allowing more realistic eye movements. *Skeletal animation* is used for animating the limbs and *blend shapes* are used for facial expressions. To control the higher level behavior of the characters, a *state machine* and *behavior trees* are used.

- Design of an intuitive user interface. A panel that follows the player is displayed with info on volume of recorded sound, a selling progress bar and earned money. An arrow appears above the character that the player is looking at. The usability of the application was evaluated with a usability test. Not all participants agreed that it was clear what actions could be taken. However, they all were able to complete the main task of the game.

5.2 Future Work

In an upcoming project at Reykjavik University, the effectiveness of using virtual reality for social training will be evaluated. Therefore, further development of the application, or some of its components, might be useful.

First, both the user interface and the character behavior could be improved, as discussed in section 4.3. Secondly, it might be beneficial to redesign how the characters navigate the environment. When they are in the *Walking* state they will wander around randomly. Because of this, there is some risk of a character walking again by the player, even if the player has already sold to that character. An interesting solution for the navigation of the characters would possibly be some kind of a vector field system. That is, a system based on vectors or arrows that are put in the environment, controlling the motion of the characters in different locations. This would probably need to be integrated with some kind of proxemics or territory system, stopping characters from getting to close to the player and each other and walking into each others walking territory.

Bibliography

[1] T. Furmark, M. Tillfors, P.-O. Everz, I. Marteinsdottir, O. Gefvert, and M. Fredrikson, "Social phobia in the general population: Prevalence and sociodemographic profile", *Social Psychiatry and Psychiatric Epidemiology*, vol. 34, no. 8, pp. 416–424, 1999.

[2] H.-U. Wittchen, M. B. Stein, and R. C. Kessler, "Social fears and social phobia in a community sample of adolescents and young adults: Prevalence, risk factors and co-morbidity", *Psychological Medicine*, vol. 29, no. 2, pp. 309–323, 1999.

[3] G. Celani, M. W. Battacchi, and L. Arcidiacono, "The understanding of the emotional meaning of facial expressions in people with autism", *Journal of Autism and Developmental Disorders*, vol. 29, no. 1, pp. 57–66, 1999.

[4] M. R. Kandalaft, N. Didehbani, D. C. Krawczyk, T. T. Allen, and S. B. Chapman, "Virtual reality social cognition training for young adults with high-functioning autism", *Journal of Autism and Developmental Disorders*, vol. 43, no. 1, pp. 34–44, 2013.

[5] M. J. Smith, E. J. Ginger, K. Wright, M. A. Wright, J. L. Taylor, L. B. Humm, D. E. Olsen, M. D. Bell, and M. F. Fleming, "Virtual reality job interview training in adults with autism spectrum disorder", *Journal of Autism and Developmental Disorders*, vol. 44, no. 10, pp. 2450–2463, 2014.

[6] M. Fleming, D. Olsen, H. Stathes, L. Boteler, P. Grossberg, J. Pfeifer, S. Schiro, J. Banning, and S. Skochelak, "Virtual reality skills training for health care professionals in alcohol screening and brief intervention", *The Journal of the American Board of Family Medicine*, vol. 22, no. 4, pp. 387–398, 2009.

[7] E. Klinger, S. Bouchard, P. Légeron, S. Roy, F. Lauer, I. Chemin, and P. Nugues, "Virtual reality therapy versus cognitive behavior therapy for social phobia: A preliminary controlled study", *Cyberpsychology & Behavior*, vol. 8, no. 1, pp. 76–88, 2005.

[8] S. R. Harris, R. L. Kemmerling, and M. M. North, "Brief virtual reality therapy for public speaking anxiety", *Cyberpsychology & Behavior*, vol. 5, no. 6, pp. 543–550, 2002.

[9] D. Steele, "Ntc (national training center): Between hollywood and hell", *Army*, vol. 58, no. 7, pp. 26–28, 2008.

[10] G. Frank, C. Guinn, R. Hubal, P. Pope, M. Stanford, and D. Lamm-Weisel, "Just-talk: An application of responsive virtual human technology", in *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*, Citeseer, 2002, pp. 773–779.

[11] W. L. Johnson, "Serious use of a serious game for language learning", *Frontiers in Artificial Intelligence and Applications*, vol. 158, p. 67, 2007.

[12] W. L. Johnson, S. Marsella, N. Mote, H. Viljhálmsson, S. Narayanan, and S. Choi, "Tactical language training system: Supporting the rapid acquisition of foreign language and cultural skills", in *InSTIL/ICALL symposium 2004*, 2004.

[13] L. Tickle-Degnen and R. Rosenthal, "The nature of rapport and its nonverbal correlates", *Psychological Inquiry*, vol. 1, no. 4, pp. 285–293, 1990.

[14] D. Givens, "Greeting a stranger: Some commonly used nonverbal signals of aversiveness", *Semiotica*, vol. 22, no. 3-4, pp. 351–368, 1978.

[15] M. Argyle and M. Cook, *Gaze and Mutual Gaze.* Cambridge University Press, 1976.

[16] A. Kendon, *Conducting Interaction: Patterns of Behavior in Focused Encounters.* CUP Archive, 1990, vol. 7.

[17] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion.", *Journal of Personality and Social Psychology*, vol. 17, no. 2, p. 124, 1971.

[18] D. McNeill, *Hand and Mind: What Gestures Reveal About Thought.* University of Chicago Press, 1992.

[19] K. R. Thórisson, "Communicative humanoids: A computational model of psychosocial dialogue skills", PhD thesis, Massachusetts Institute of Technology, 1996.

[20] J. Cassell, T. Bickmore, H. Vilhjálmsson, and H. Yan, "More than just a pretty face: Affordances of embodiment", in *Proceedings of the 5th International Conference on Intelligent User Interfaces*, ACM, 2000, pp. 52–59.

[21] A. Cafaro, R. Gaito, and H. H. Vilhjálmsson, "Animating idle gaze in public places", in *International Workshop on Intelligent Virtual Agents*, Springer, 2009, pp. 250–256.

[22] C. Pedica and H. Vilhjálmsson, "Spontaneous avatar behavior for human territoriality", in *Intelligent Virtual Agents*, Springer, 2009, pp. 344–357.

[23] B. Bédi, H. E. E. Helgdóttir, B. Arnbjörnsdóttir, H. H. Vilhjálmsson, *et al.*, "Starting a conversation with strangers in virtual reykjavik: Explicit announcement of presence", in *Proceedings from the 3rd European Symposium on Multimodal Communication, Dublin, September 17-18, 2015*, Linköping University Electronic Press, 2016, pp. 62–68.

[24] N. Magnenat-Thalmann, R. Laperrire, and D. Thalmann, "Joint-dependent local deformations for hand animation and object grasping", in *In Proceedings on Graphics interface '88*, Citeseer, 1988.

[25] C. Liu, "An analysis of the current and future state of 3d facial animation techniques and systems", Master's thesis, School of Interactive Arts & Technology-Simon Fraser University, 2009.

[26] P. Joshi, W. C. Tien, M. Desbrun, and F. Pighin, "Learning controls for blend shape based realistic facial animation", in *ACM Siggraph 2006 Courses*, ACM, 2006, p. 17.

[27] D. Harel, "Statecharts: A visual formalism for complex systems", *Science of Computer Programming*, vol. 8, no. 3, pp. 231–274, 1987.

[28] D. Isla, "Building a better battle", in *Game Developers Conference, San Francisco*, 2008.

[29] D. Hu, Y. Gong, B. Hannaford, and E. J. Seibel, "Semi-autonomous simulated brain tumor ablation with ravenii surgical robot using behavior tree", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 3868–3875.

[30] S. Kopp, B. Krenn, S. Marsella, A. N. Marshall, C. Pelachaud, H. Pirker, K. R. Thórisson, and H. Vilhjálmsson, "Towards a common framework for multimodal generation: The behavior markup language", in *International Workshop on Intelligent Virtual Agents*, Springer, 2006, pp. 205–217.

[31] A. Shoulson, N. Marshak, M. Kapadia, and N. I. Badler, "Adapt: The agent development and prototyping testbed", *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 7, pp. 1035–1047, 2014.

[32] J. Nielsen and T. K. Landauer, "A mathematical model of the finding of usability problems", in *Proceedings of the INTERACT'93 and CHI'93 Conference on Human Factors in Computing Systems*, ACM, 1993, pp. 206–213.

[33] R. S. Kennedy, N. E. Lane, K. S. Berbaum, and M. G. Lilienthal, "Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness", *The International Journal of Aviation Psychology*, vol. 3, no. 3, pp. 203–220, 1993.

Appendix A

Credits

Resources used for the creation of the software in this project.

A.1 Software

- Standard Assets (by Unity Technologies)
  https://assetstore.unity.com/packages/essentials/asset-packs/standard-assets-32351
- NodeCanvas (by Paradox Notion)
  Official website:
  http://nodecanvas.paradoxnotion.com
  Unity asset store:
  https://assetstore.unity.com/packages/tools/visual-scripting/nodecanvas-14914

A.2 Models and textures

- Used assets from the program MakeHuman
  http://www.makehumancommunity.org/

A.3 Sounds

- Ambience-Centre of a busy mall.wav (by StudentTanita)
  https://freesound.org/people/StudentTanita/sounds/327146/
- Grove mall ambience (by EskimoNeil)
  https://freesound.org/people/EskimoNeil/sounds/404372/
- lloyd_center.flac (by tim.kahn)
  https://freesound.org/people/tim.kahn/sounds/34623/
- Shopping mall.wav (by Blahoslav)
  https://freesound.org/people/Blahoslav/sounds/316781/
- Shopping Mall Ambiance.wav (by CheneWessels_170017)
  https://freesound.org/people/CheneWessels_170017/sounds/407608/
- winkel centrum 1 AB.wav (by wrinex)
  https://freesound.org/people/wrinex/sounds/72612/

Appendix B

Data From Usability Test

This appendix presents some additional data that was collected in the usability test. Four people participated in a usability test following the pilot usability test. Two of them where male and two female. The participants were aged 26 - 60 years old and were all Icelandic. Participants were asked how much time they spent on average in a week using a computer. Two said they spent 6 - 40 hours and the other two said they spent more than 40 hours. They were also asked if they had experienced virtual reality using a virtual reality headset before and two of them said they had. These are the number of sales that each of the four participants made: 3, 6, 7, 7.

Participants were asked to rate 3 statements on a 5 level Likert scale. The ratings could be any of the following: Strongly disagree, disagree, neutral, agree, strongly agree.

These are the 3 statements:

- It was clear in the virtual environment what actions could be taken by me.
- The behavior of the characters felt natural.
- It was clear from the behavior of the characters which ones were ready to buy from me and which ones were not.

The results can be seen in table B.1. In the table, the statements are labeled as: Actions, Behavior, Buying.

The participants also answered the *Simulator Sickness Questionnaire* (SSQ) [33], both before and after playing the game. The total SSQ score can be seen in table B.2.

	Strongly disagree	Disagree	Neutral	Agree	Strongly agree
Actions	0	0	2	1	1
Behavior	0	1	1	2	0
Buying	0	0	1	3	0

Table B.1: Number of marks by rating for each statement.

Participant	SSQ score before	SSQ score after
1	0	0
2	0	18.7
3	29.92	33.66
4	22.44	22.44

Table B.2: SSQ total score for each participant, before and after playing the virtual reality game.