A CLOUD-BASED EXTENSIBLE AVATAR FOR HUMAN ROBOT INTERACTION

ENAS KHALED ALTARAWNEH

A THESIS SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

MASTER OF SCIENCE

GRADUATE PROGRAM IN ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO
FEBRUARY, 2019

Abstract

Adding an interactive avatar to a human-robot or human-machine interface requires the development of tools that animate the avatar so as to simulate an intelligent partner in the conversation. Although there has been considerable advancement in the development of cloud-based speech-to-text, text-to-speech and natural language understanding and generation, there exist few tools to support interactive avatar animation and modeling. This thesis addresses this issue.

The human-robot interaction avatar developed here utilizes standard speech-to-text cloud-based software to perform generic speech-to-text mapping. This mapping provides support for continuous and active listening that detects sound and reduces the surrounding noise participants. The speech to text module can be tuned to expected queries/commands from human operators thus enhancing the expected accuracy of the process and ensuring that the resulting text maps to pre-determined commands for the robot itself. The avatar's text-to-speech module combines a standard cloud-based or local text-to-speech generation system with a 3D avatar (puppet) whose animation is tied to the utterance being generated. Text messages presented to the text-to-speech module are embedded within an XML structure that allows the user to tune the nature of the puppet animation so that different emotional states of the puppet can be simulated. The

combination of these two modules enables the avatar representing the robot to appear as if it listens and recognizes speech. Utilizing this approach avatars can answer and respond to questions given to them and can be programmed to answer customized questions. An expression package controls the animated character's mood and facial expressions. An idle loop process animates the avatar puppet between utterances so that the character being rendered is never still but rather appears to interact with external users even when not being spoken to directly. This also helps to obscure latencies in the speech understanding - rendering loop. The efficiency of the approach is validated through a formal user study.

Acknowledgements

First and foremost I would like to thank God for all the blessings and strength he gives me to overcome all challenges. I would like to acknowledge the guidance and support of my supervisor Prof. Michael Jenkin, one of the most inspirational people I know. A thank you to my lab mates, namely Robert, Bikram, Masoud and Arhum for their support. I would also like to thank my parents Khaled and Suhair for their continuous sacrifices and selfless devotion. A special thanks to my sisters Shatha, Maess, Aseel and Anoud, and my brother Feras for their love and support throughout my life. Last but certainly not least, I would like to thank my husband Ahmad and my children Bara, Adam and Jad for being a positive force pushing me towards success.

Table of Contents

List of Abbreviations

EA Extensible Avatar

ROS Robot Operating System

AUML Avatar Utterance Markup Language

XML eXtensible Markup Language

HRI HumanRobot Interaction

HCI HumanComputer Interaction

VAD Voice Activity Detection

AAM Active Appearance Model

GPU Graphics Processing Unit

HMM Hidden Markov Model

CAT cluster adaptive training

GUI Graphical User Interface

RA Realistic Avatar

CA Cartoon Avatar

DDR Double Data Rate

DIMM Dual In-line Memory Module

SATA Serial Advanced Technology Attachment

QUIS Questionnaire for User Interface Satisfaction

PUEU Perceived Usefulness and Ease of Use Questionnaire

CSUQ Computer System Usability Questionnaire

ASQ After-Scenario Questionnaire

ANOVA Analysis of Variance

AI Artificial Intelligence

IA Intelligent Agent

EaaS Everything as a Service

IaaS Infrastructure as a Service

PaaS Platform as a Service

SaaS Software as a Service

DBaaS Database as a Service

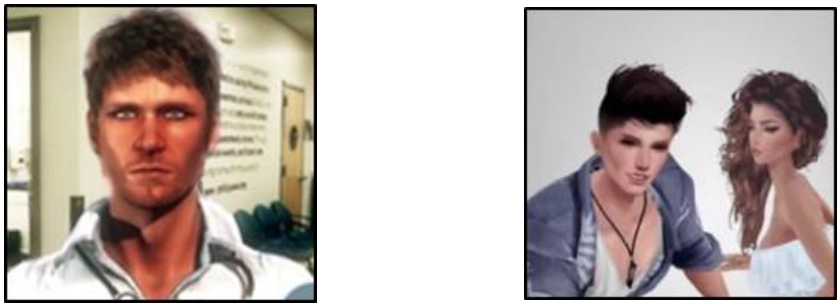
FaaS Funtion as a Service

Chapter 1

Introduction

A range of different technologies have emerged to enhance the effectiveness and quality of the computer-based services we receive. These technologies include online and self-service applications, virtual agents and robotic systems; all providing information and assistance. A common link between these digital implementations is that they require human computer interaction through an interface. Central to the art of creating such interfaces is that in addition to providing functionality the interface must be interactive, user friendly and inviting, so that the interface can encourage user involvement and cooperation with the application. The challenge is to design an interface so that it enables smooth interactions between the users and the application. One way of providing a natural and effective user interface is through the simulation of some active agent or avatar and having that simulation interact with the user. Examples of such avatars can be found in fictional, prior, and existing user interfaces. Examples of each of these avatars are shown in Figure 1.1. There are a number of reasons why avatars are found in user interfaces. For one, they can be used to put a "friendly face" on the interface. To take an early example consider Clippit shown in Figure 1.1(a). Clippit, or as more commonly known "Clippy", was one of the first widely used interactive animated characters. Clippit assisted

(a) Clippit (b) Max headroom (c) Nadia



(d) Fredrick (e) IMVU chat

Figure 1.1: **Examples of previous and existing interactive avatars.**
(a) Clippit a discontinued animated character that interfaced Microsoft Office help. Image reprinted from [5]. (b) Max Headroom was a fictional avatar from a TV show of the same name. Image reprinted from [1]. (c) Nadia, a current interactive avatar used for a company's online support. Image reprinted from [2] d) Fredrick, an animated medical assistant. Image reprinted from [3]. (e) Examples of avatars created for IMVU virtual chatrooms. Image reprinted from [4].

Microsoft Office users and was an interface for Microsoft Office help content. Clippit and the other animated Microsoft Office Assistants were discontinued after criticism from users and employees. Max Headroom shown in Figure 1.1(b) is a fictional artificial intelligence (AI) avatar portrayed as a computer-generated TV character [1]. Max headroom was known for his wit and his constant criticism of his 'user'. Interestingly Max 'stuttered' intentionally when rendered to highlight his 'computer generated' nature. Nadia, shown in Figure 1.1(c), is an emotionally intelligent, lifelike avatar used for interaction with the users of Soul Machine's company website [2]. Figure 1.1(d) shows an avatar named Fredrick that provides an interface to a virtual medical assistant [3]. Figure 1.1(e) shows characters from the Instant Messaging Virtual Universe (IMVU) chat room [4].

Avatars are inherently multimodal in nature and allow for a more intimate form of interaction than simple visual- or audio-only interfaces. They literally give the machine a face. The avatar as a multimodal interface offers the ability to transform a human-robot communication or human-machine communication into an interaction that integrates gestures and visual clues with audio that elaborates the intended message.

Although human-machine interaction examples occur over a vast range of different applications, one area of particular interest occurs when humans interact with robots. Robotic technologies do not need to be fully autonomous to be compelling and persuasive. Lifelike robots with avatars that mimic our facial expressions or possibly create their own add a new layer of compliance and persuasiveness to fully- or semi-autonomous robotic systems. There are many anthropomorphic robots that use, or at least try to use, human facial gestures as part of their human-machine interaction infrastructure. A number of such devices use physical face structures (e.g., [6, 7, 8]) while others use a computer display that mimics a human head (e.g., [2, 3]). A recent example here is the floating head robot that has been developed on the international space shuttle [9]. An avatar's human-machine construct provides a combination of different interaction modalities and are the next rational step in the development of human-communication. A robot can 'see' using a vision system, 'listen' through an audio recognition system, and 'touch' through pressure sensors. Adding some animated physical embodiment of a real human creates the lifelike illusion of the robot or AI which can be greatly compelling to the person interacting with that robot device[10].

One practical problem in the development of human-machine and human-robot interaction systems is the lack of common tools and tool chains to support critical aspects of the interaction between a robot and a human. This work described in thesis helps to address this issue. In particular, this work describes the development and evaluation of an animated and programmable avatar with an associated toolkit that can be easily integrated into any robot or AI that utilizes the ROS (Robot Operating System) middleware [11]. The Cloud-based Extensible Avatar for Human Robot Interaction toolkit (or the EA toolkit for short) is a rendered 3D visual puppet (avatar) through which humans can interact with a robot or other software systems. The toolkit utilizes cloud-based rendering and speech understanding as its core computational aspects.

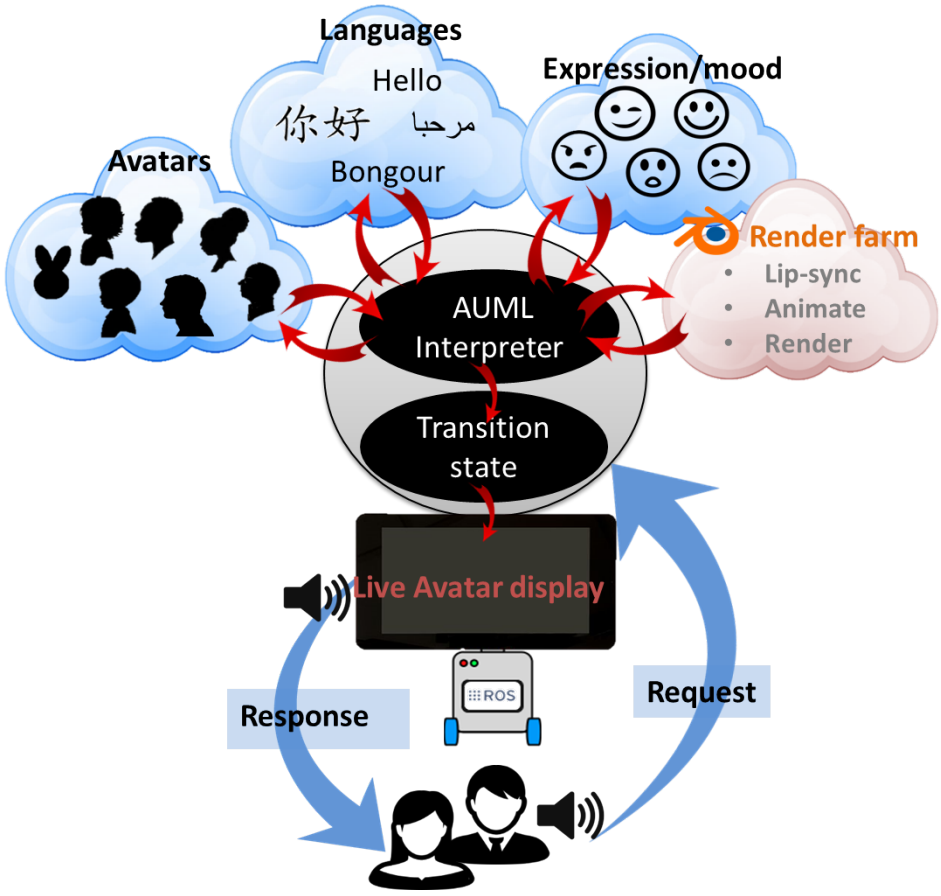


Figure 1.2: **Extensible Avatar (EA) toolkit.**
Users interact with a robot or other software agent through spoken word and responses are made through utterances and a synchronized 3D rendered visual avatar. The avatar can be customized for language, appearance and tenor of the conversation.

The toolkit relies on two critical software components: a generic speech-to-text module that converts utterances captured by a microphone in proximity to the robot into a standard ROS text message, and a generic text-to-utterance module that is capable of emitting natural language speech. Augmenting these utterances with a visual avatar requires rendering complex and detailed animations in real time and synchronizing these animations to the utterance. Such rendering typically requires specific and/or resource intensive hardware which may not be available on an autonomous robot. In order to overcome this limitation, this work explores utterance recognition and the rendering of the avatar using cloud-based computational resources. A high level view of this process is shown in Figure 1.2.

The software infrastructure described here is based on the development of two basic modules: an audio-understanding input module and a rendering output avatar module. These two modules are ROS-based and are designed to be easily integrated into a range of different robot systems. One goal of this work is to provide a plug and play package to support HRI that is platform independent while providing an effective interface for human-robot interaction.

This work leverages substantive earlier work in speech understanding, text-to-speech processing, cloud-based natural language understanding, and cloud-based computation generally. For example, a generalized speech understanding module was developed that leverages large-scale efforts in this area so that any speech-to-text engine can be utilized within this toolkit. Similar generic wrappers were developed to develop abstract representations for text-to-speech generation and natural language understanding. With that said, this work concentrates on the use of specific cloud-based resources that address these tasks. For example, Google's cloud based speech-to-text engine [12] is the primary library used for this task. This engine simulates human listening by detecting speech in the environment, reducing the surrounding noise, and utilizing a cloud-based AI engine to obtain the spoken words as text. Although this module is a core component of the this thesis, for the most part this thesis utilizes the speech recognition toolkit with few, if any, customizations beyond those already present in the underlying systems.

Similarly, in terms of utterance generation this work relies on an abstract definition of this functionality. That being said, here again the work is targeted towards Wolfram Alpha and Google's efforts in this area. Systems such as Wolfram Alpha [13] provide responses to requests and systems such as Google's Text to Utterance module[14] provides spoken utterances from stored ASCII text. This thesis integrates this information with a synchronized 3D rendered puppet avatar to enhance human-robot interaction. The text-to-utterance module combines the output of text-to-speech with 3D avatar (puppet) facial animation to generate the requested utterance. In order to allow for customization and personalization of the 3D avatar, rather than taking raw text and the corresponding audio file as input, the textual input is augmented with information about the manner in which the avatar should behave during different portions of the rendering

(a) Winking (b) Disgusted (c) Looking up

(d) Surprised (e) Hopefull (f) Sad

Figure 1.3: **Some simulated expressions of an avatar developed in this work**
(a) shows the avatar winking (b) shows the avatar disgusted, (c) shows the avatar looking up (d) Shows the avatar surprised, (e) Shows the avatar hopefull, (f) shows the avatar sad.

process. Specifically, the text messages passed to the text-to-speech module is embedded within an XML structure known as the Avatar Utterance Markup Language (AUML) to define this personalization. The Avatar Utterance Markup Language enables the user to integrate expressions and emotions within the spoken words. Figure 1.3 shows some simulated expressions. The combination of these two modules allows the avatar representing the robot to appear as if its listens and recognizes vocal commands given to it. The avatar can answer and respond to questions given to it and can be programmed to answer customized questions with actions.

The system developed in this thesis leverages a number of state of the art cloud-based software components. In particular it relies on a speech-to-text recognition module, a knowledge engine, a text-to-speech

engine, a 3D character designing program, a 3D animation program, and a lip-syncing plugin for the animation program that extracts the sounds in words, maps them to mouth shapes and plots them according to duration and occurrence in the text in real time. An expression package controls the animated characters mood and facial expressions. Rather than seeking to advance our understanding in terms of these aspects, this thesis considers how to integrate these modules to provide an animated cloud-based avatar. Recognizing that the use of cloud-based resources will introduce unwanted delays in the recognition and rendering process, key technical contributions in my thesis include (i) the development of an adaptive parallelization strategy to leverage cloud-based rendering resources to minimize the latency itself, (ii) the development of an "idle loop" process to obscure any resulting latency in the recognition, response and rendering process, and (iii) the use of last-minute latency stuttering obscuring through forward and backward rolling of the video at the point of display. The idle loop process animates the avatar puppet between utterances so that the character being rendered is not still but rather appears to interact with external users even when not being spoken to directly. This process further obscures initial rendering latency. Finally, my thesis includes a user study to quantify the value of enhancing human-robot interaction through the use of a visual avatar. Figure 1.4 shows a strip of the animation of one character, "Manjot", developed in this work.

Beyond the direct application to autonomous systems, the toolkit developed within my thesis has many other applications. As the ROS nodes are general and take simple text as input and present them as output, the software can be used to provide quick and simple avatar-based interfaces to a wide range of applications. As such, the infrastructure can be used in locations where it is difficult to find interested employees to act as greeters or receptionists. For example, the technology can be used in museums, banks, schools, universities, real-estate offices, etc to drive general human-machine interaction. The ability to customize the character and add expression provides additional value and sophistication to these products. Moving the rendering part of this application to the cloud is essential to provide a cost-effective user-friendly device or robot. Rendering near-real-time animations requires specialized hardware that is generally costly, heavy and requires considerable power to operate. Moving rendering on to the cloud to elevate the human-robot

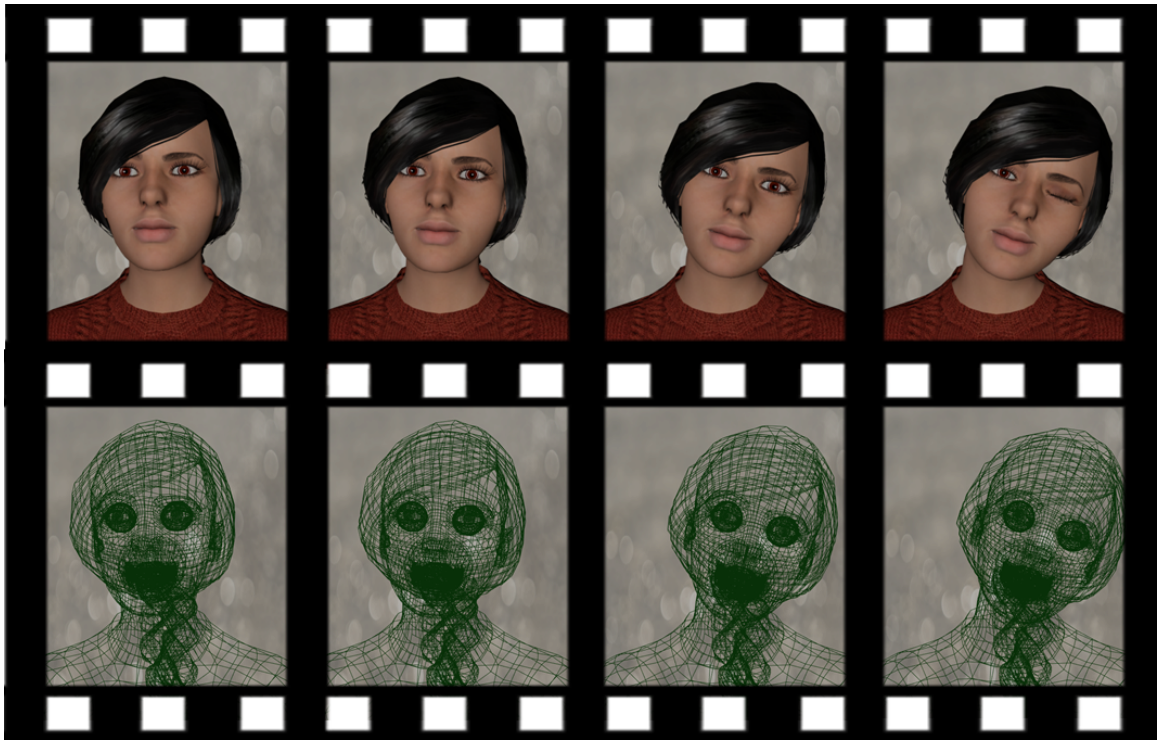


Figure 1.4: **A film strip of an animated character**
The avatar is controlled by control points. These controls are of two types, facial controls and the body rig. Examples of a facial controls include mouth-open, left-brow-right-up, left-brow-mid-down and mouth-right-corner-up. Examples of body rig include neck, left-eye-lid, jaw and spine.

interaction is a realistic step to take based on the trends in cloud computing.

Avatars as a form of knowledge-based communication raise a number of questions. How will users react to avatars as a form of knowledge-based communication? Would the idle loop process create a good enough distraction from rendering latency? Why not other forms of knowledge-based communication such as text or audio? Are avatars appropriate for every knowledge-based application? A user study conducted in this work addresses these overall questions and evaluates the usefulness of a full 3D realistic avatar for human-robot interaction.

1.1 Structure of this work

This thesis is organized into seven chapters. This section is the culmination of the chapter entitled "Introduction". This chapter introduced the key aspects of the problems associated with this research as well as the motivations behind conducting this research. The second chapter entitled "Background" explores aspects of related fields of research with a primary focus on human-robot interaction using avatars. The third chapter entitled "Speech-to-text processing with ROS" details the process of taking human utterances and understanding them. The fourth chapter entitled "Text-to-speech processing with ROS" details the process of creating an animated speech of the response for the human-robot interaction. The fifth chapter entitled "Cloud-based rendering and real-time display" details the creation of the cloud-rendering farm and the multi-process optimization of the cloud-based rendering used to produce real-time results. The sixth chapter entitled "Human-robot interaction user study" is a description of human factors experiments to evaluate the usefulness of a full 3D realistic avatar for human-robot interaction and the results of that user study. The final chapter entitled "Summary and future work" contains concluding remarks about the research and suggests possible avenues for further enhancements.

Chapter 2

Background

The human-robot interaction (HRI) field is multidisciplinary, drawing on contributions from a range of fields including artificial intelligence, robotics, human-computer interaction (HCI), speech, and the social sciences. Due to the increasing availability of complex robots and non-experts' exposure to them, this relatively young field has attracted considerable attention over the past few years. Robots are now being developed for a range of different applications including rehabilitation, eldercare, education and assisted therapy. The successful application of robots in such domains requires addressing issues with both the engineering of the device and also in terms of the design of the interaction of the robot with human users [15]. As autonomous robots move out of the research lab they are expected to interact in a socially acceptable manner with groups of multiple people and robot naïve people. Applicable research that develops methods and technologies to support natural and easy communication between such users and the robot through speech, gestures, and facial expression is a key goal of research in the human-robot interaction space (see, [16, 17, 18]).

In our daily lives, we often think of robots as either toys or industrial machines but personal robots now exist that interact with humans one-on-one on a daily basis. Take the Milo robot as an example [19].

(a) Milo (b) PRIveT (c) Hygeiorobot

Figure 2.1: **Avatar Examples**
(a) Milo, a robot to stimulate interaction with autistic children.Image reprinted from [19]. (b) PRIveT, a robotic system that engages and adapts to its users. Image reprinted from [15]. (c)Hygeiorobot a mobile assistant for hospitals. Image reprinted from [20].

Milo was developed by Robokind to interact with people using vocal and facial expressions. Milo helps children with autism practice and can be used to develop and enhance important social skills. Milo must interact with children in a natural manner or it will have no clinical impact [19]. As another example, Lopes and Teixeira [20] describe work on human-robot spoken dialogue interaction and the Hygeiorobot a mobile robotic assistant for hospitals. As yet another example, Sandygulova et al. [15] describe PRIveT testbed which includes an ubiquitous robotic system that is able to autonomously engage and adapt to its users. These examples are shown in Figure 2.1.

In order to create more sophisticated robots that can integrate with human society, understanding how robots and humans interact is essential to successfully accomplish particular tasks. In order to be effective social robots should be able to learn the preferences and capabilities of the humans with whom they interact and then adapt their behaviors to achieve more efficient and user-friendly interactions. Brief reviews of the techniques used in human-robot interaction, a review of the use of avatars as a form of human-robot interaction, and the computational tasks associated with using avatars and cloud-based rendering are detailed in the following sections of this chapter.

2.1 Techniques for human-robot interaction

It is already possible to interact with robots via natural communication means including speech (e.g., [20, 21, 22, 23, 24, 25]), and gesture (e.g., [26, 27, 28, 29]), as well as through the use of traditional joysticks (e.g., [30, 31, 32]) and computer keyboards and monitors (e.g., [30, 33, 34, 35]). Beyond the modality of the interaction, there remains the problem of how to use the modality (or set of modalities) to interact. There are many possible approaches. For example, "real world point and click" is a technique for human-robot interaction that enables a human to unambiguously select a 3D location in the world and communicate it to a mobile robot [36]. Here the human points at a location of interest and then ("clicks it") with a pointer device such as a laser pointer. The robot detects the resulting laser spot with an omnidirectional camera using a narrow-band green filter. After detection, the robot moves and estimates the location's 3D position with respect to the robot's frame of reference and can then move to that location. Pointing and clicking can also be used to reduce the dimensionality of complex robot tasks. Georgia Tech's Robot Autonomy and Interactive Learning (RAIL) group implemented a "constrained positioning" method for point and click for grasping tasks, which intelligently limits the degrees of freedom that a user needs to specify in order to position something [37]. With this system the user needs to select only a grasp point, approach angle, and grasp depth to control the robot. Putting these approaches together allows for a range of options for teleoperated grasping, from full 6-DOF manual control to 3-DOF constrained positioning grasping to single-click automated grasping.

It is also possible to interact with a robot using gestures that do not rely on a specific pointing device. One example of an application that uses gestures as a human-machine interaction is the TENZR wristband [38]. The TENZR system relies on a custom sensing technology that recognizes users gestures, permitting them to control mixed reality environments, complex robotic systems, home appliances, and even medical equipment in operating rooms. Gestures are complementary to the use of language (voice) and when communicating allows humans to interact with technology more intuitively and naturally. To take one very common example,

Google Gesture Search [39] uses gestures for human-computer interaction. Google Gesture Search is designed for the Android Eclair operating system (Android 2.0) and above, and enables users to search their phones contacts, bookmarks, applications and music simply by scribbling out letters with their finger. But it is also possible to use gestures without requiring a special sensor in contact with the user to provide gesture informatoin for human-machine interaction. For example, the Simon Frasier University's Autonomy Lab have developed drones that react to human faces along with contextual voice and gesture commands. The drones take commands individually as well as part of a group [40]. Researchers in the this lab have also developed drones that can be piloted through the user's facial expressions [41].

Applications that use natural language as an interface engage in conversations as humans naturally do. There are many examples of this type of interaction including modern systems such as Siri [42], Alexa [43] and Cortana [44]. Siri is an intelligent personal assistant, part of Apple Inc.'s iOS, watchOS, macOS, and tvOS operating systems. The assistant uses voice queries and a natural language-based user interface to answer questions, make recommendations, and perform actions by delegating requests to a set of Internet services. The software adapts to users' individual language usages, searches, and preferences, with continuing use. Returned results are individualized. Alexa is Amazon's cloud-based voice service available on tens of millions of devices from Amazon and third-party manufacturers. With Alexa, a developer can build natural voice experiences that offer customers an intuitive way to interact with the technology they use every day. Alexa's Cortana is a virtual assistant created by Microsoft for Windows systems and other Microsoft products. Cortana can set reminders, recognize natural voice without the requirement for keyboard input, and answer questions using information from the Bing search engine. But what are the advantages and disadvantages of the various interaction approaches? For example, Medicherla and Sekmen [23] report results of a user study that indicates that voice-control and the ability of spatial reasoning were reliable indicators of efficiency of robot teleoperation. In this study 75% of the subjects who demonstrated a high ability of spatial reasoning favored using voice-control over manual control. But are people more comfortable in interacting with realistic human avatars? Is there an uncanny valley here at which point avatars become less effective?

2.2 A brief history of avatars

The word "avatar" originates from the Sanskrit word "avatara", meaning "descent". It is used to describe an incarnation or a bodily manifestation of an immortal being in Hinduism. In the era of information and technology, any form of representation that is used to denote a user's entity can be considered an avatar. In the realm of science fiction the term avatar can be traced back to at least Isaac Asimov [45] in his early robot stories from the 1950's. Avatar's have appeared in a number of books, television shows, and movies since then. For example, in Norman Spinrad's novel *Songs from the Stars* (1980) [46], the term avatar is used in a description of a computer generated virtual experience. In 1982, *Tron* [47] an American science fiction action-adventure film presents a computer programmer who is transported as an avatar inside the software world of a mainframe computer where he interacts with programs in an attempt to escape. *Max Headroom* [1] is a fictional artificial intelligence (AI) avatar introduced in early 1984. The character was portrayed as "The World's first computer-generated TV host". Max Headroom's computer-generated appearance was achieved with a real actor, prosthetic makeup and hand-drawn backgrounds. In 1992, Neal Stephenson's [48] science fiction novel, *Snow Crash*, presented users of a computer-system entering a virtual world and interacting with virtual versions of themselves, naming them "avatars". *The Matrix* [49] is a 1999 science fiction action film that depicts a future in which reality is actually a simulated reality *The Matrix*, created by machines to subdue the human population, while their bodies' heat is used as a source of energy. More recently in 2007 a vocaloid voicebank called *Hatsune Miku* [50] was developed by Crypton Future Media. Her voice is modeled from a Japanese voice actress. *Hatsune Miku* is marketed as a virtual idol, and has performed as an animated projection onstage at concerts. The *Avatar* movie [51] is a 2009 American science fiction film. It is set in the 22nd century, when humans colonize a moon in a star system. In order to mine the mineral unobtanium which is a room-temperature superconductor, humans use avatars to represent themselves on the planet's surface. Even more recently *Ready Player One* [52] is a 2018 American science fiction story set in the year 2045. In this film humans try to engage as avatars in work and play using the virtual reality

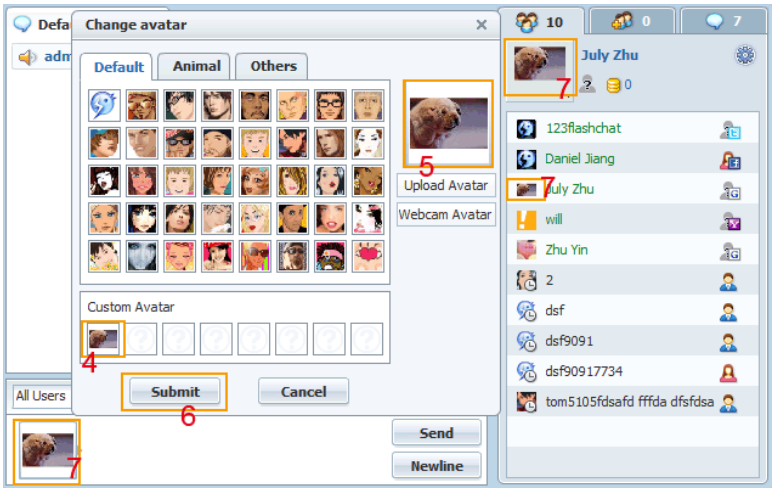


Figure 2.2: **Avatars used as a visual marker in a chat room. Image reprinted from [55].**

software OASIS, escaping the desolation of the real world.

Fictional avatars can be very effective and of course have no technical limitations outside of those required for plot development. Real world avatars on the other hand must deal with the realities of the underlying technology. Early real-world examples of avatars are very simplistic, even by today's standards. According to [53] a name, a voice or a photo can serve as a user's avatar even if they may not look or behave like the real user. In the past, avatars were icons with limited motion serving merely as a visual markers for users, while most interaction was textual [54]. Figure 2.2 shows an example of avatars used as visual markers in a textual chat room [55]. Although these avatars resembled humans they are primitive in form, rendered with coarse graphics and limited in terms of customization. More recent avatars are much more customizable. In such systems users are able to customize the avatar in more ways. For example customizing eye color, race, age, hair style, height, body shape, clothing, and even facial expressions [56]. Using these features, users can create virtual humans with distinctive personalities,and with a unique appearance. Figure 2.3 shows an example made with Make Human, a software tool that allows for the creation of such advanced avatars.

Avatars can be used to provide a presence for a user within some software construct. They can also be used to provide a similar presence for a software agent. The term "virtual agent" and "avatar" are often

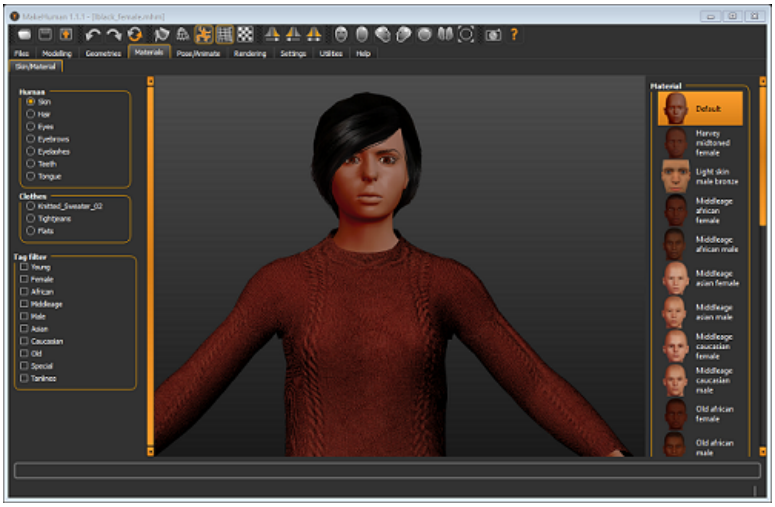


Figure 2.3: **A screen shot of Make Human, a software for building 3D highly customizable avatars.**

used inter-changeably when referring to an avatar that relies on Artificial Intelligence (AI). An artificially intelligent agent (IA) is an autonomous entity that observes the environment through sensors and acts upon it using actuators, directing its activity towards achieving a specific set of goals [57]. A virtual agent or intelligent avatar has applications in almost every field. For example, avatars and virtual agents have been used as an interface for home care monitoring and companionship (see[3]). They are also inherently multi-modal in nature and allow an intimate relation between the users and the avatar. Figure 2.4 shows an avatar that provides an interface to a virtual medical assistant [3].

An example of an advanced avatar associated with an intelligent agent is the Soul Machine's emotionally intelligent lifelike avatar "Baby X". Nadia is an example of a "Baby X" avatar (see Figure 2.5). "Baby X" was developed for the NDIS (National Disability Insurance Scheme) in Australia using IBM Watson's artificial intelligence technology as a cognitive back-end by FaceMe, an Auckland-based real-time video communication company. Nadia interacts directly with NDIS customers on the company's website. The webcam on the user's computer acts as Nadia's eyes, while the microphone acts as her ears, resulting in a human-like conversation with an on-line avatar [2].

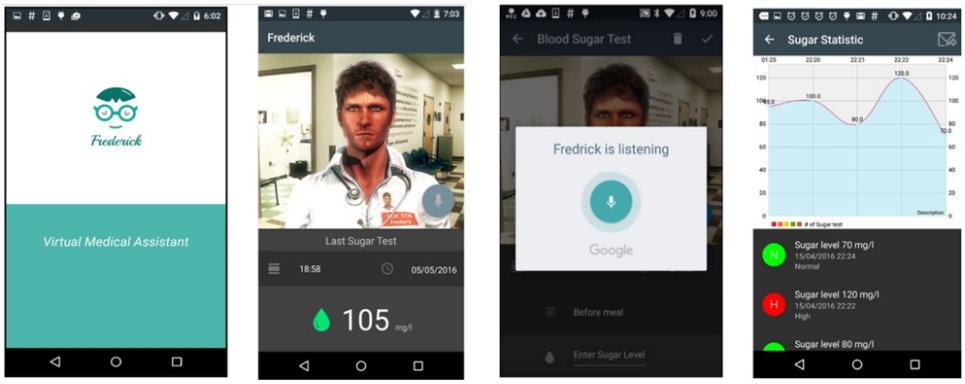


Figure 2.4: **An virtual medical assistant avatar. Image reprinted from [3]**

2.3 Computational tasks associated with an avatar

Using an avatar as a human-robot or human-computer interface presents several complex computational tasks. One natural form of communication between a human and an avatar, given an avatar as an embodiment of a human-like form, is to receive input through direct speech. This speech needs to be recognized by the avatar and then processed to generate a logical and reasonable response. Naturally a human user would expect that the avatar's response to be in the form of a lifelike response of spoken words with appropriate meaningful gestures and visual cues from the avatar such as the one shown in Figure 2.5. If the avatar is human in appearance, one would expect the avatar's mouth to move and follow the motions associated with the utterances being spoken, the eyes to move appropriately, as should other facial features. But is this really necessary of an avatar to be accepted by users?

One of the challenging computational tasks associated with animating an avatar is expediting the avatar's response. The computational complexity and hence the latency associated with rendering and displaying a talking head on a client device can be reduced by caching pre-rendered portions of the avatar's display locally (see [58] as an example). In this approach the client device has a local cache that stores audio and visual data associated with rendering the avatar. These cached sentences and sentence templates are used to generate full and partial utterances. The system uses the appropriate stored sentence or template from

Figure 2.5: **Nadia a lifelike Avatar for NDIS. Image reprinted from [2].**

the client cache to render the talking head response and renders on demand only that portion of the talking head response that is not stored in the cache. A similar approach for audio is found in a study by Chen et al. [59] which reduces latency in web-browsing text-to-speech (TTS) systems without the use of a browser plug-in. The system allows the browser to send prosodically meaningful sections of text to a web server. The system uses a TTS server to convert intonational phrases of text into audio and then sends the audio file as a response to the browser. The system uses a cache to save audio files within the browser in case similar text requests occur later. Reusing animation components [60] in a tool that prepares animated characters can reduce latency in those specific tools. Systems that automatically choreograph and synchronize animations can make use of reusable components of dialog streams and gestures.

2.3.1 Speech to text

The term "speech-to-text" refers to the use of methodologies and technologies that enable the recognition of spoken language and its translation into text by computers [61]. There is a long history of the development

of such technologies and a survey of existing approaches and descriptions of current state of the art systems can be found in a number of studies including [62], [63] and [64]. Speech-to-text plays an important role in human-robot interaction (see [16, 17, 18] for examples). All speech recognition software requires a mechanism to capture audio from the environment, typically a sound card and microphone. The microphone captures the user's speech and an associated sound card converts the speech into a digital form that the software can interpret. This conversion is a preprocessing step that involves sampling, windowing and de-noising of the raw audio signal [65, 66]. Following this preprocessing, compressed and filtered speech frames are forwarded to a feature extraction stage. The feature extraction stage derives descriptive features from the windowed and enhanced speech signal to allow for a classification of sounds [67, 68]. The feature extraction is typically required because the raw speech signal contains information besides the linguistic and classifying the signal based on the raw data can result in a high word recognition error rate. Next, an acoustic model is used to translate the sequence of features into phonemes or higher-level structures. There are a number of approaches used in acoustic modeling including Artificial Neural Networks [69, 70], Hidden Markov Models [71, 72], Gaussian Mixture Models [73] and Dynamic time warping [74]. Finally at the language model stage, sentences are extracted from the corresponding phoneme sequence. For this, the language model stage must determine the most probable words given the phoneme sequence [75]. Current speech recognition systems typically exploit neural networks (see [76, 77, 78, 79, 80]). Earlier approaches (e.g., [81, 82]) typically relied on generative modeling approaches, however with recent advances in big training data and computing power the use of these generative approaches has declined. The basic speech-to-text process is summarized in Figure 2.6.

2.3.2 Text to speech

"Text-to-speech" or "text-to-utterance" is the term used for the artificial production of human speech by converting natural language text into an audio signal that represents the generated speech. The generated speech can be created by concatenating stored pieces of recorded speech corresponding to phrases, words or

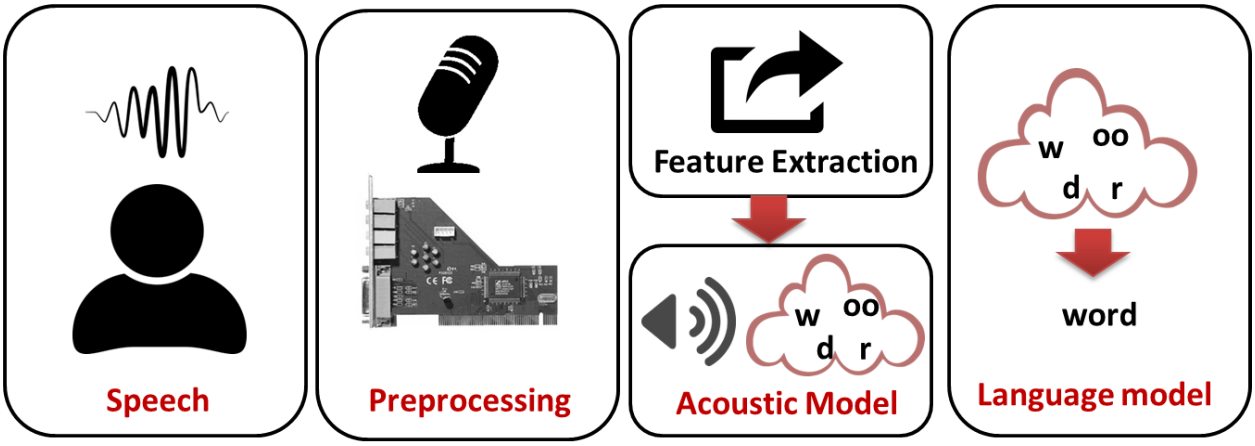


Figure 2.6: **An illustration of the speech to text process. See text for details.**

phonemes identified in the natural language text. The size of the stored speech units can differ. Systems that store phonemes provide the largest output range, but may lack clarity in the output [83]. Some systems store a dictionary of entire words or sentences to allow for higher-quality output [84]. In order to better mimic human sounds some systems include a model of the vocal tract and human voice characteristics to create realistic "synthetic" voice output [83]. Regardless of the underlying model used to represent the utterance as an audio signal typically these speech engines typically consist of two stages [85]. The text normalization, pre-processing or tokenization step is the first stage. This stage converts the raw text containing symbols like numbers and abbreviations into the equivalent written-out words. Phonetic transcripts of each word are then generated. The engine extracts information about the patterns of stress and intonation in the language (its prosody) by splitting the text into phrases, clauses, and sentences and marking them. Phonetic transcriptions and prosody information are the building blocks of a symbolic linguistic representation. At the phonetic level, prosody is characterized vocal pitch (fundamental frequency), loudness (acoustic intensity) and rhythm (phoneme and syllable duration)[86]. The next stage in text-to-speech engines involves converting this symbolic linguistic representation into sound. At this stage some systems include the computation of the target prosody which includes pitch contour and phoneme duration to impose it on the output speech [87]. The basic process is illustrated in Figure 2.7.

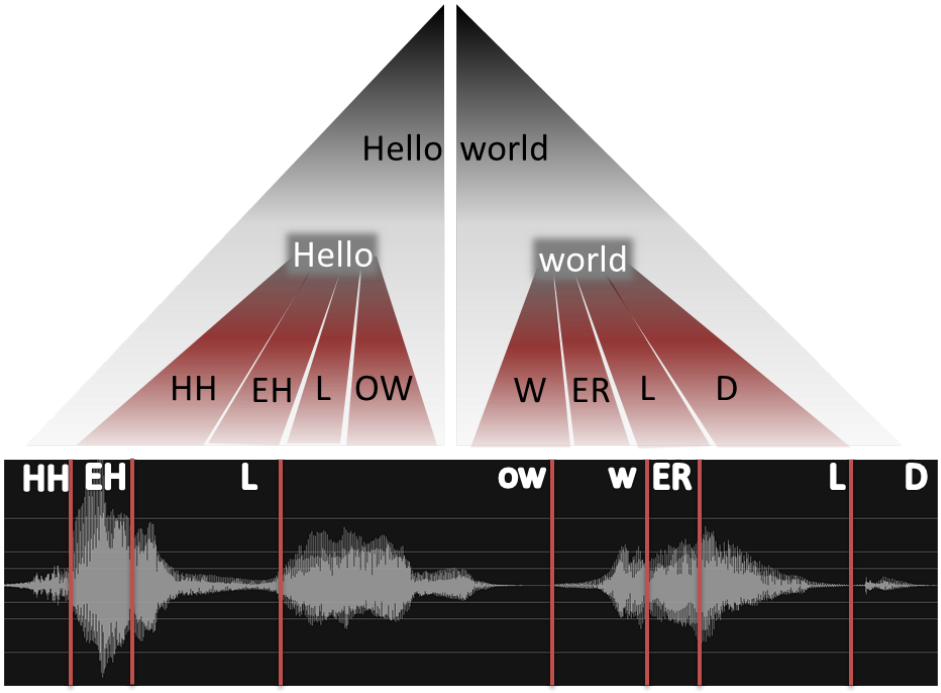


Figure 2.7: **An illustration of the text to speech process.**
A text structure is converted into words. Each word into phonemes. And for these to an actual audio signal.

2.3.3 Multi-modal text to speech

Although text-to-speech approaches can produce realistic audio, human perception is multi-sensory. Humans often use multiple senses when interacting with their environment, but of these, two, seeing and hearing are perhaps the most important [88]. Humans combine audio information and the movements of the lips, tongue and other facial muscles generated by a speaker in order to recognize emotion and behavior [88]. HRI systems that combine audio and a visual talking realistic head rendering of an utterance are likely to improve a human's perception of the interaction over interaction devices that lack these features. That being said, it is necessary to avoid the "uncanny valley" [89] in which systems can become too lifelike and uncomfortable to interact with.

There have been a number of previous attempts to create realistic 3D expressive talking heads and some have shown encouraging results (e.g.,[16, 90, 91, 92]); however existing systems have not yet achieved

the level of realism of their 2D counterparts [91]. 2D talking heads presently look more realistic than their 3D counterparts, but they are limited in the variety of poses and in the lighting conditions that can be simulated [91]. Enabling a talking head to express emotion along with a synchronized utterance is a challenging problem. Model-based approaches have shown some potential in solving this problem. For example, the avatar described by Andesron et al. [91] is driven by text and emotion inputs and generates expressive speech with corresponding facial movements. It used a Hidden Markov Model (HMM)-based text-to-speech synthesis system [93] with an active appearance model (AAM)-based facial animation system [94]. The system used a cluster adaptive training (CAT) framework to train both the speech and facial parameters which allows for the creation of expressions of different intensity and the combining of different expressions together to create new ones. Results on an emotion-recognition task show that recognition rates given the synthetic output are comparable to those given the original videos of the speaker. Andesron et al. [92] presents a similar study that produced a talking head given an input text and a set of continuous expression weights. The face is modeled using an active appearance model (AAM), and several extensions that enhance the face. The model allows for normalization with respect to both pose and blink state which significantly reduces artifacts in the resulting synthesized sequences [92].

2.4 Cloud-based rendering

Existing avatar user interface systems utilize sophisticated local computational and rendering capabilities to display the avatar and its utterance. Given that prevalence of cloud-based approaches to natural langauge understanding system and audio synthesis systems it seems appropriate to consdier the potential for cloud-based rendering of the avatar as well. Cloud-Based rendering (Figure 2.9) is a form of Cloud computing [95] in which internet (cloud)-based computing resources are used to render animations or other visual scenes[96, 97]. Cloud computing reduces the costs at the end-user[98], but introduces issues related to latency/lag due to the transmission costs from the cloud to the user and the limited bandwidth of such communication.

2.4.1 Cloud-based solutions

Cloud solutions provide a service-oriented architecture that presents everything as a service (EaaS). Cloud-computing provides three standard models of service which are Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS)[99]. Microsoft Azure[100], Google Cloud platform[14] and Amazon Elastic Compute Cloud[101] are examples of currently available cloud-based solutions. Similar to all cloud computing services Microsoft Azure, formerly known as Windows Azure, is used for building, testing, deploying, and managing applications through a global network of data centers. This infrastructure provides the three standard types of cloud services and supports different programming languages, tools and frameworks. Currently available Microsoft Azure services include compute, mobile, storage, media, developer and machine learning services [14]. A competing solution is the Google Cloud Platform which includes a set of management tools, provides a series of modular cloud services including computing, data storage, data analytics and machine learning. The Google cloud platform also provides Database as a Service (DBaaS). This is a document-oriented database and operates as a Function as a Service (FaaS) by providing serverless functions to be triggered by cloud events. Similar to all cloud solutions, Amazon's Elastic Compute Cloud (EC2) or Amazon Web Services (AWS) allow users to rent virtual computers where they can run their own computer applications. Cloud solutions encourage scalable deployment of applications by providing virtual machine images to configure virtual machines called "instances". These instances can host the users desired software. Users can create, start, and stop these instances as required. Users can also select the geographical location of instances, allowing for a reduction in communication latency.

2.4.2 3D computer graphics software

3D computer graphics use a three-dimensional representation of geometric data stored in a computer for the purposes of performing calculations and rendering images. The 3D model provides a mathematical representation of the object. A model can be displayed visually as a two-dimensional image after 3D rendering

used for non-graphical computer simulations and calculations. 3D models can also be rendered into a 3D physical representation of the model in various ways, for example by using 3D printing.

The creation of a 3D computer graphics display starts with a 3D modeling step which is the process of forming a computer model of an object's shape. Next, Layout and animation are used the place and move the 3D objects within a scene. Finally, the 3D rendering process involves generating an image or series of images based on a model of light placement, surface type, and other intended image qualities.

There exist a large number of rendering systems and engines. One example of a 3D design and animation application is Blender[102]. Blender is a professional, free and open-source 3D computer graphics modeling and rendering toolkit. Blender can be used for creating animated videos, 3D printed models, interactive 3D software and video games. Blender's features include 3D modeling, UV unwrapping, texturing, rigging, animating, camera tracking, rendering, motion graphics and video editing. Autodesk Maya [103] is another example of a 3D computer graphics application that can be used to create interactive 3D applications, including video games, animated film, TV series, or visual effects. In these and similar applications users define a scene to implement and edit media associated with the project. Unity[104] is a game engine used to develop both three-dimensional and two-dimensional video games and simulations for computers, consoles, and mobile devices. These and similar commodity modeling and game engine tools can be used to build sophisticated 3D models of complex objects such as avatars. Unfortunately such tools typically require substantive computational resources.

2.4.3 Lip syncing speech using graphic software

Character speech animation with lip synchronization (lip-sync) is considered an important but tedious task. There exists software systems and human aided approaches that can be used to partially or completely automate the creation of facial and speech animation. One example of such work is the framework for synthesizing a 3D lip-sync speech animation to a given speech sequence and its corresponding texts described

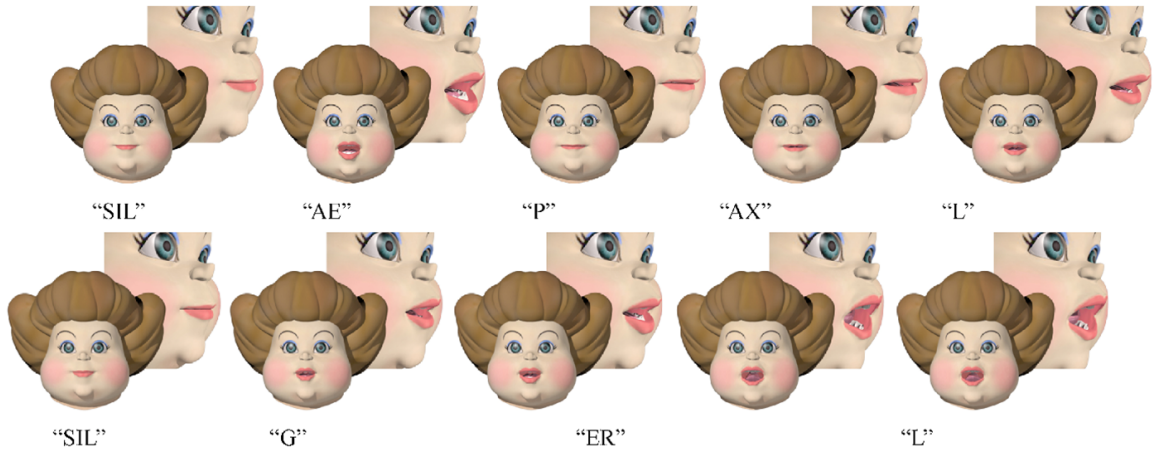


Figure 2.8: **An example of a lip-sync using DAMs. Image reprinted from [106].**

by Chen et al. [105]. The first step in this process identifies the key-lip-shapes from a training video that guides the creation of corresponding 3D key-faces. These 3D key-faces are used to construct Dominated Animeme Models (DAM) for each kind of phoneme. Considering the coarticulation effects, which is the articulation of two or more speech sounds together so that one influences or dominates over the other, the DAM computes the polynomial-fitted animeme shape for each key-face and its corresponding dominance weight [105, 106]. Figure 2.8 shows an example of the process described in [105, 106]. other approaches exist to create this blending. For example, Wang et al. [107] describes a statistical, multi-streamed Hidden Markov Model (HMM) is trained using super feature vectors consisting of 3D geometry, texture and speech. The HMM is then used to synthesize both the trajectories of head motion animation and the corresponding dynamics of texture. The resulting 3D talking head animation is controlled by the model predicted geometric trajectory and the articulator movements, e.g., lips, are rendered with dynamic 2D texture image sequences. In yet another example, the speech signal which is represented by Mel-Frequency Cepstral Coefficients MFCC vectors is classified into classes of visemes using neural networks [108]. Using genetic algorithms the topology of neural networks is automatically configured using genetic algorithms. This eliminates the need for manual neural network design and considerably improves the viseme classification results.

When manually lip-syncing, graphic designers and animators follow a number of guidelines to improve

the result of the lip-sync appearance [109]. The first is to not lip-sync every sound. Lip-syncing every sound can cause the avatar's jaw to open and close more frequently than it would naturally. Another tip is to offset the lip sync for readability. For the user to be able to read the lip sync the jaw opening should offset one to two video frames before the audio otherwise the user may feel like the lip-sync is unsynchronized with the audio. The last guideline is to dominately animate the closed mouth shapes. The mouth is closed with the B, M, and P sound. These sounds are dominant sounds and should not be blended because they need to be read clearly. For example, the key-shapes for these sounds can be held for a couple frames for more emphasis[109].

There exists a range of software tools, plugin and add-on solutions that aid animators with lip-syncing and facial animations. One example is CrazyTalk [110]. Crazy Talk is a 2D real-time facial animation software that uses voice and text to animate facial images. It allows the animators to use their own voice to create their animations in real-time using an automatic motion engine. Another example is Faceshift, which is a software solution that can capture the user's facial expressions in real time and generates an avatar that mimics the user[111]. Faceshift technology uses off-the-shelf RGBD cameras. Faceshift is compatible with most available 3D software packages via plugins and data export. Facerig [112] is another real-time facial animation software solution. Facerig uses the user's webcam to digitally embody a character. Facerig is an open creation platform that enables users to make their own characters, backgrounds or props. The facial animation or mimicking provided in both Facerig and Faceshift include tools to automatically lip-sync the user's speech. A Blender plugin called Quicktalk [113] can semi-automatically lip-sync any selected audio using the MakeHuman MxH2 character [56]. To use this plugin the user manually exports a Makehumans MXH2 character, adds the sound track, adds the word-sound dictionary, adds the text script to be lip-synced, and then uses the plugin to automatically plot the lip-sync. The word plot markers then need to be manually adjusted to match the audio of each word so the lip-sync is not out of place.

2.4.4 Rendering lag and latency

Real-time rendering typically uses a local graphics processing unit (GPU) to perform the necessary complex calculations. Real-time rendering systems find perhaps their largest mark in terms of video games as they typically must render complex animations in real-time [114]. Minimizing the amount of delay between a user input action and the corresponding change of the system's output (system lag) is crucial in maintaining the illusion that the user is actually part of the simulated world. Beyond the rendering pipeline, there are other factors that contribute to the overall latency in rendering animations in real time. These include the input devices used, the nature of the display, the rendering software and even the graphics card driver settings [115]. As the computation is moved away from a device connected directly to the display to some remote computational resource such as the cloud, the rendering pipeline is impacted by the delay associate with transmitting the resulting imagery to the display, and in terms of a cloud-based rendering process the network latency associated with the cloud itself.

Computer graphics processing is dominated by data parallel operations. This type of parallelism distributes data sets across the multiple computational units. Operations can be performed by different processors on individual data sets and the outputs combined. Modern display processors, for example, typically perform different tasks at each pixel in the rendered image. In terms of the cloud-based rendering of an avatar, a key observation is that different portions of any utterance may be rendered in parallel by the massively parallel computational resources available in the cloud. Such parallel renderings can then be stitched together in order to produce the full or partial animation of the avatar. This basic concept is illustrated in Figure 2.9. By properly structuring such parallelization it is possible to both meet latency requirements as well as producing a complete visual rendering of the avatar using off-site computational resources.

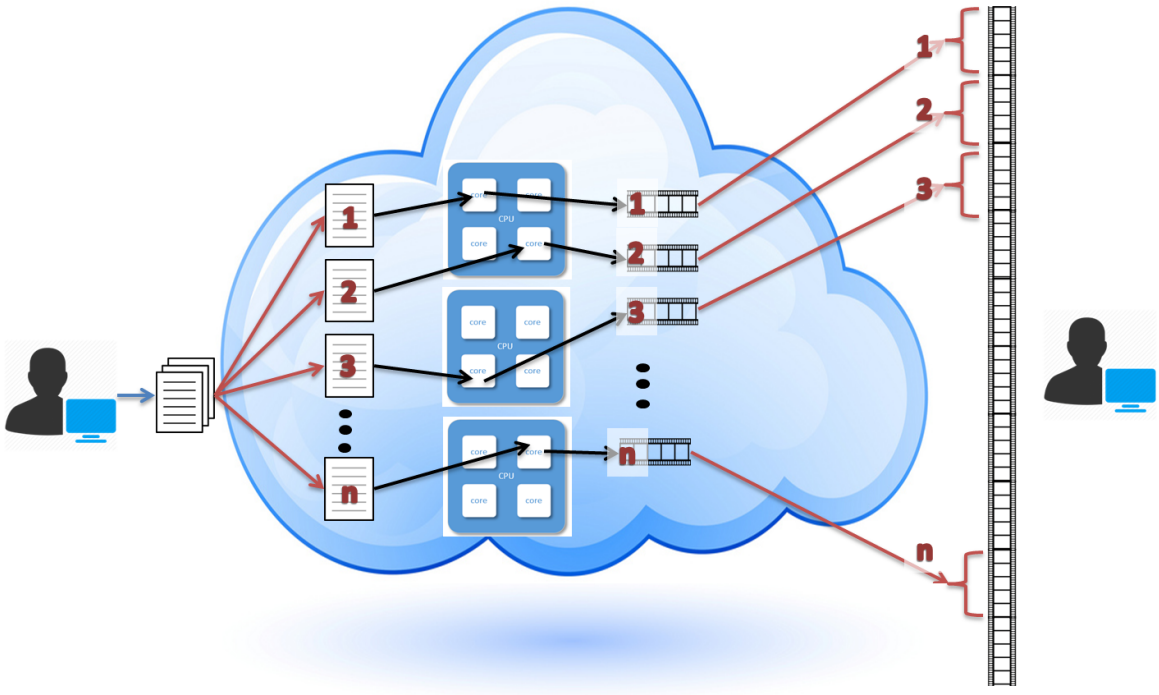


Figure 2.9: **A illustration of cloud based rendering.**
A rendered sequence can be parallelized and then re-integrated into a coherent video sequence.

2.5 Social aspects of human-robot interaction

When humans interact and collaborate with each other they use verbal and nonverbal signals to coordinate their turn-taking actions [116]. These signals are expressed in many ways including through expressions of the face and voice cues. Social robots of the future are expected to interact with "naïve" humans, thus, it is critical that these social robots can produce and recognize these actions. Skantze [116] provides an overview of a number of studies showing that humans in their interaction with a human-like robot make use of coordination cues found in human-human interaction. This study also shows that it is possible for a robot to detect these cues and use them to facilitate real-time coordination. Alonso-Martin et al. [117] present a multi-modal human robot interaction system that focuses on emotion cues. The results of this study show a high success rate in automatic user emotion recognition through the use of two information channels (audio and visual) relative to approaches based on a single channel alone. Other research, such as that presented

by Dragone et al. [118] seek to address the need to develop clear social interactions between socially capable robots and between robots and humans. Dragone et al. [118] describe a framework which supports rational social interaction between real and artificial systems.

The perception of social robots varies from individual to individual. The representation, behavior and visual characteristics of a robot have an effect on the user's perception of the agent, it's intelligence and safety during the interaction. Based on a study of elderly users, the perception of a robot avatar changes with the simulated age of the avatar [119]. The study used a young and an old avatar that provides the elderly assistance with errand services, communication and entertainment tools. The participants perceived the older avatar as being more competent. For perceived safety, the users considered their own state as more quiescent interacting with the older avatar. In a study that compared the users' trust in three media representations of an advisor: video, a computer generated avatar, or a robot, the users were presented with two of the advisors one of which was an expert, and the other was a non-expert. Users preferred seeking advice from the expert and had a tendency to seek advice from the robot or video and rarely from the avatar [120].

Crowelly et al. [25] investigates differences in the way that males and females view robots. An on-line magazine article [89] argues that the user's mental state influences the nature of the interaction and that stressed, overworked and exhausted people would rather leave a voice mail or send an email than talk face-to-face. That is, they believe that people would say, "I'd rather talk to the robot", rather than to talk to the person under certain conditions. The study suggests that under certain circumstances that friends can be exhausting and that as a robot will always be there until it's not required, sometimes robots are more desirable than friends! There is some evidence that people prefer to talk to robots than other people under other circumstances as well. For example, Niemelä et al. [17] showed that people tend to respond positively to social service robots in field trials in public places. The results of the survey conducted by Niemelä et al. [17] indicate a high social acceptance among humans engaging with service robots in a shopping mall. However, it is unknown how their opinions and attitudes might evolve if the same robot continues to be

presented in the same service after the novelty effect of it wears off.

Robots can affect our perception of each other. There exist tele-operated robot avatars in which an operator's behaviors are mimicked to improve distance interaction [16]. This study investigates how robot mediation affects the way the personality of the operator is perceived. The goal of this study was to investigate if judgment of the personality of the operator can be consistent in assessing personality traits, if users can agree with one another, if users agree with the operators' self-assessed personality, and if users shift their perceptions to incorporate characteristics associated with the robot's appearance. The study showed that:

- Participants utilize robot appearance cues along with operator vocal cues to make their judgments of simulated personality traits.
- Operators' arm gestures reproduced on the robot aid personality judgments by the participants.
- How personality cues are perceived and evaluated through speech, gesture and robot appearance is highly operator-dependent.

2.6 Summary

The human-robot interaction field draws on contributions from artificial intelligence, robotics, human-computer interaction, speech, and the social sciences. In order to create more sophisticated robots that can integrate with human society, understanding how robots and humans can interact to successfully accomplish particular tasks is essential. The advancement of interaction via natural communication means such as speech in human-computer interaction have greatly benefited human-robot interaction. Applications that use natural language as an interface use knowledge engines and engage in conversations as humans do naturally. Adding some animated physical embodiment of a real human, e.g, an avatar, to the speech interaction can help create the lifelike illusion of the robot or AI which can be compelling to the person interacting

with that robot device. Avatars have become complex 3D entities. They can now be rendered as three dimensional forms with animated movements that can aid in the expression of the avatar's personality and complement various social interactions. This complexity and sophistication comes with a set of associated computational tasks to enabling this type of animated agent-driven avatar. Speech-to-text, text-to-speech and avatar animation modules need to coordinate to achieve the desired response from the avatar. Having such an interaction readily available at all times to users with minimal computational power on their devices is possible due to the existence of cloud-based rendering. Cloud computing reduces the costs at the end-user, but introduces issues related to latency/lag due to the transmission costs from the cloud to the user and the limited bandwidth of such communication.

Chapter 3

Speech to text processing with ROS

This chapter considers the problem of speech-to-text processing within ROS. The main goal of the speech-to-text moudle is to provide spoken input control of the robot or device. The module utilizes cloud-based infrastructure and software API to perform generic speech-to-text mapping. This provides for continuous and active listening that detects speech in the environment, reduces the surrounding noise, and obtains the spoken words as text, simulating human listening. In addition to performing generic speech-to-text translation, the speech-to-text module can be tuned to expected queries/commands from human operators thus enhancing the accuracy of the process and ensuring that the resulting text maps to pre-determined commands for the robot itself. The basic structure of a speech-to-text module process developed in this work is given in Figure 3.1.

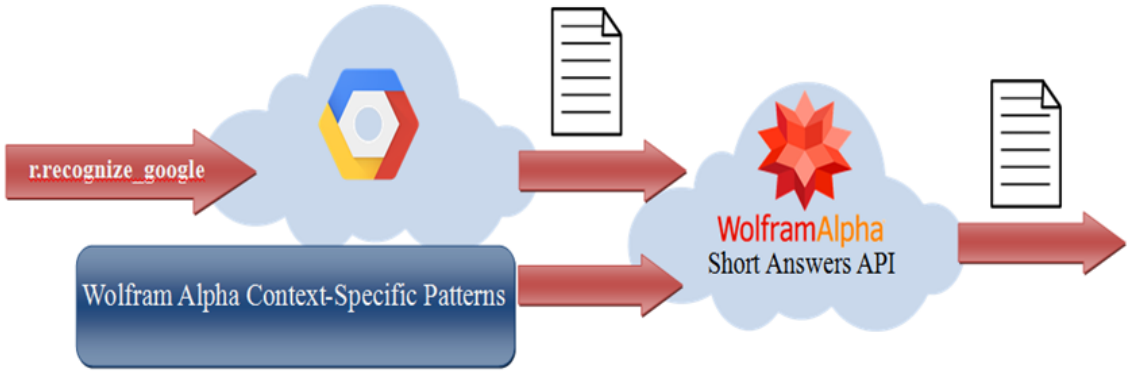


Figure 3.1: **An example of the speech to text module process.**

3.1 Abstract ROS implementation

The Robot Operating System (ROS)[11] has emerged as the de facto standard middleware for research robot software. ROS is a message passing framework with in which computational nodes transduce messages and pass them on to other nodes. Unfortunately, of present, ROS lacks an independent speech-to-text framework. As there exist a number of different suppliers of cloud-based speech recognition system, here we describe an abstract model of a speech-to-text ROS node. This 'abstract node' has been implemented utilizing a number of different cloud-based and local hardware speech-to-text engines, although the work has concentrated on the Google Engine[12] primarily. Figure 3.2 illustrates the abstract model for the speech-to-text recognizer.

The development of this abstract toolkit builds upon substantive previous efforts in this domain. Speech-to-text recognition systems have been implemented in a variety of different robot systems (e.g.,[21] [22][23][20]). A standard toolkit for local recognition that can be found online [12] and can be easily integrated into any ROS robot system. Google and others provide toolkits to integrate their recognizer with 3rd party software (e.g.,[12][121]). The output of this process is a natural language expression typically represented as a sequence of words in the recognition language.

This work integrates the previous work [12] in ROS by providing a structure of the recognizer that is

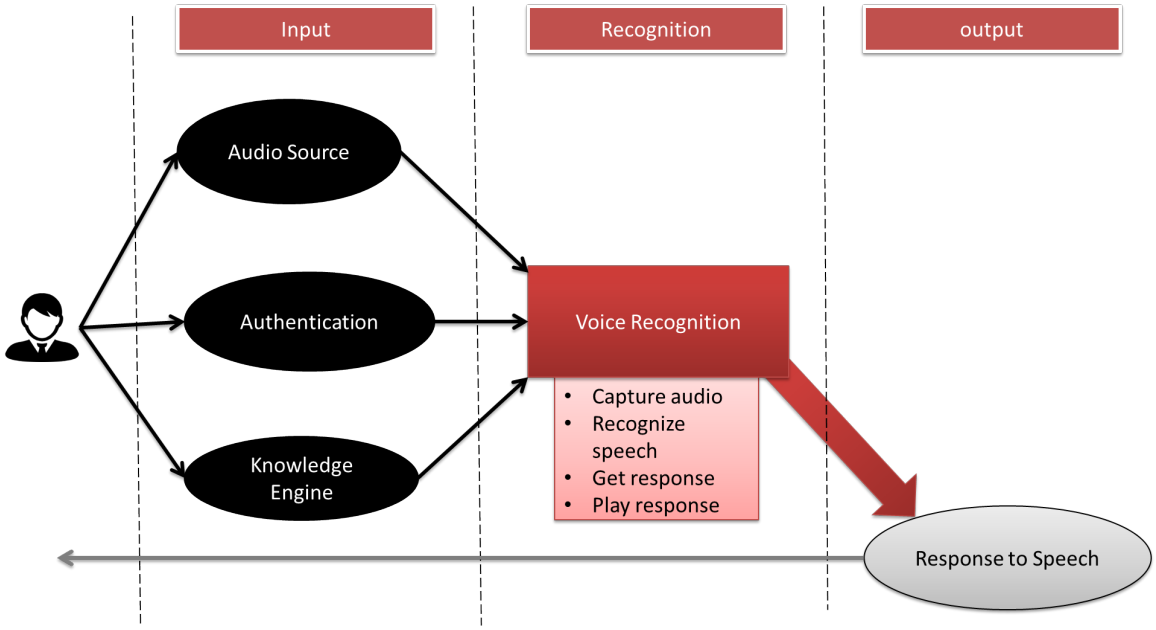


Figure 3.2: **Illustration of the abstract model of the voice recognizer.**

tailored for robots through the use of the ROS messaging system. This work also provides appropriate human-robot interaction feedback that mimics natural conversation for an enhanced user experience, such as using phrases like "pardon me" or "can you repeat that" instead of using technical feedback to the user. This work also redefines the output to be used with the knowledge engine WolfarmAlpha.

The speech-to-text module allows for application-specific robot actions as specified by context-specific patterns. Predefined sentences that match the commanded action are sent for processing as utterances. For example, if the commended action was for the robot to move in a specified direction, the selected sentence to be uttered would match that command. The ROS tools developed here provide a mechanism for receiving input from a user using speech and then generating text from that utterance [1]. A detailed illustration of the structure of this audio input system is shown in Figure 3.3.

As shown in Figure 3.3 the ROS model has an input layer that extracts the audio data and is split into two levels. The first level extracts audio data from a source to construct the audio object required for

[1]code: https://github.com/enastarawneh/avatar1.0-TTS.git.

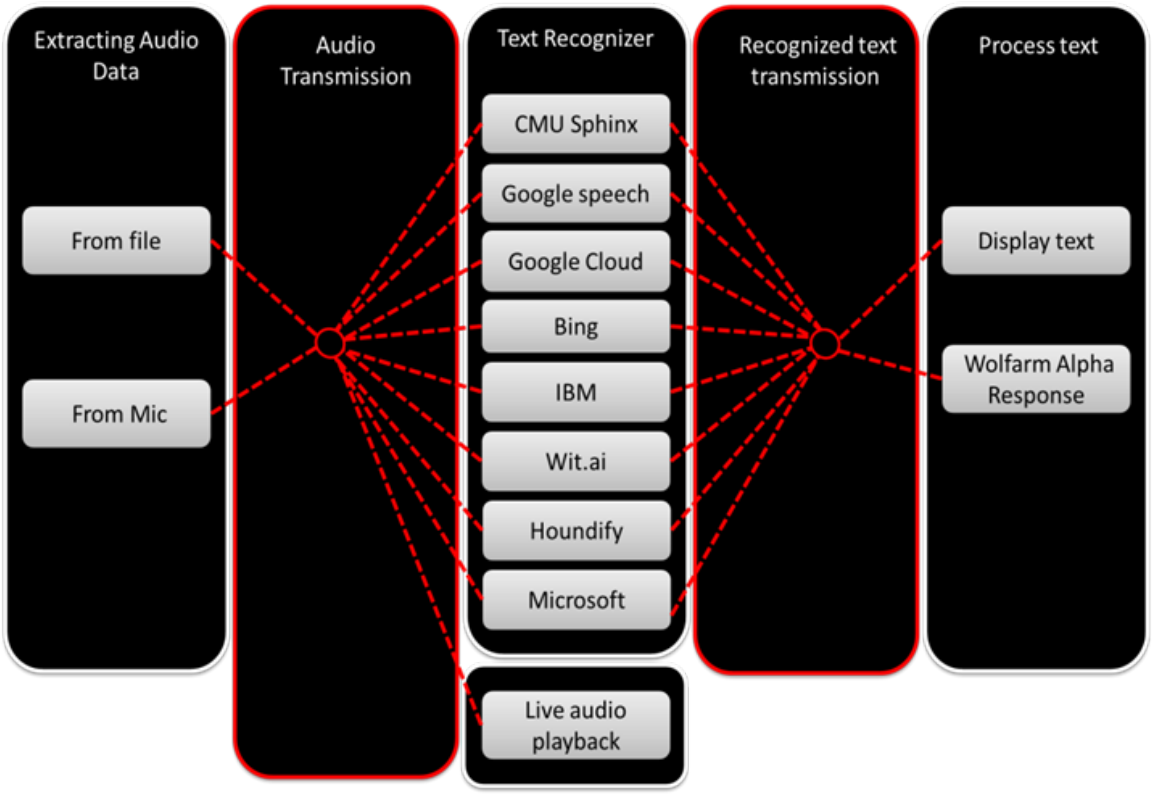


Figure 3.3: **Illustration of the nodes and structure of this input system.**

recognition. The second level reduces the noise in order to construct the required audio object. This audio data is passed from the input layer to the recognition layer using the ROS messaging system. After this message has been passed to the text recognizer, the raw audio data is used to re-create the audio object in the form required by the recognizer. The currently supported text recognizers are shown in Figure 3.3. This text recognizers implemented include Google, Bing, Microsoft and Houndify.

3.2 Google implementation

Although a generic toolkit has been developed, the most completely implemented module utilizes Google's cloud-based text-to-speech engine[12] that continuously listens for input. This engine reduces noise, supports many languages and supports context aware recognition. If the engine recognizes what is being said it

generates a sentence that is passed on to the system for further processing, If it does not, the system selects one of a number of a predefined sentences to alert the user that the system could not identify what has been said. Examples of predefined sentences include “Could you repeat that” and “Sorry I didn’t get that”. These predefined sentences are sent as utterances and processed as a response. A similar strategy is used to select sentences to be uttered in the case of lost Internet connectivity. An example response to this exception is “Hold on while I try to get connected to my brain”.

3.3 Wolfram Short API

In the case where the recognized input is a question requiring an answer that cannot be provided directly, a request is sent to a cloud knowledge engine. Although the software infrastructure is intended to support a wide number of different cloud-based knowledg engines the currently utilized knowledge engine is WolframAlpha[13]. WolframAlph’s short answer API provides the response to these requests, which in turn are input for generating the avatar utterance. The short answer API returns a single plain text result directly from the Wolfram Alpha engine. This API type is designed to deliver brief answers in the most basic format possible. It is implemented in a standard REST protocol using HTTP GET requests.

3.4 Summary

Speech provides natural communication in a human-robot interaction. The robot would be required to understand and process speech in a meaningful way to be able to provide such a method of communication. Establishing this form of communication between humans and robots using the standard robot operating system ROS allows for the creation of a more robust and portable toolkit that is found on most robots.

Chapter 4

Text to speech processing with ROS

A central technical goal of this work is to put an interactive (human) face on an interactive robot. In order to accomplish this a standard text-to-speech generation system is combined with a 3D avatar (puppet) whose facial animation is tied to the utterance being generated. In order to embed emotional state and other out of band information, messages presented to the text-to-speech module are embedded within an XML structure known as the Avatar Utterance Markup language (AUML) that allows the user to tune the nature of the puppet animation so that different emotional states of the puppet can be simulated. Operationally, the text to animated avatar process operates as a ROS node that accepts text to speak (an utterance) and animates an avatar based on this text[2]. An overview of the process is shown in Figure 4.1(a).

[2]code: https://github.com/enastarawneh/avatar1.0-STU.git.

4.1 Components

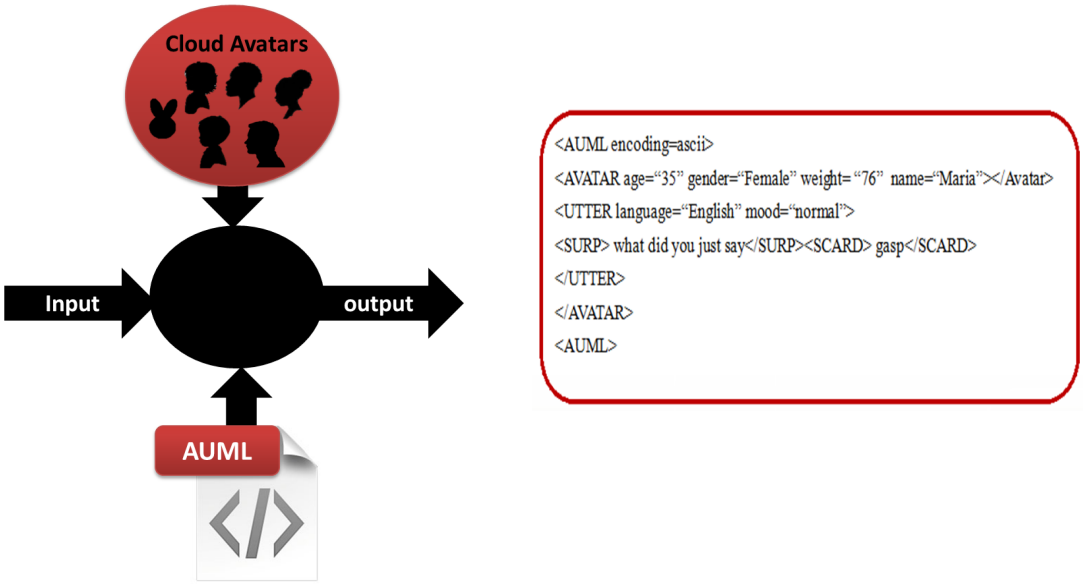
4.1.1 The Avatar Utterance Markup Language (AUML)

Rather than transmitting straight English text, the text to be rendered by the Avatar is placed within a structured framework that provides rendering hints. The Avatar Utterance Markup Language (AUML) developed in this work is a formal language for avatar utterances. This language is a XML representation; it defines a set of rules for encoding a desired output using a textual data format. Every utterance includes an avatar's detailed description, language, spoken words, expression associated with sub phrases and general mood. An example AUML script is shown in Figure 4.1(b). The goal of this language is to standardize and facilitate the use of available avatars, languages and expressions. It also allows for extensibility by a simple inclusion of new tags or values. This provides a simple interface with the system. AUML is defined through the DTD in Figure 4.2.

4.1.2 The Avatars

Avatars are 3D puppets properly rigged for animation. Work completed to date utilizes an open source realistic 3D human character design software called MakeHuman[56]. MakeHuman provides the ability to manipulate age, weight, length, gender, and race of the avatar. The software also allows for changes in facial details, hair, eyes, skin and clothes. Users can select from a variety of 3D meshes and bone structures for each character. Characters are exported using the Mhx2 rig[122] which enables MakeHuman structures to be imported into the Blender renderer[102]. Currently implemented avatars are shown in Figure 4.3.

The Avatar MHX2 model contain a full body rig and full facial controls including mouth, tongue and lip control. The avatar appearance can be adjusted by maintaining pre-defined values for the body and facial controls that can be immediately assigned to the avatar.



(a) *AUML to avatar abstract overview* (b) *An example AUML script*

Figure 4.1: **AUML overview and example script**
(a) shows the abstract overview of AUML to avatar, (b) shows an example of an AUML script

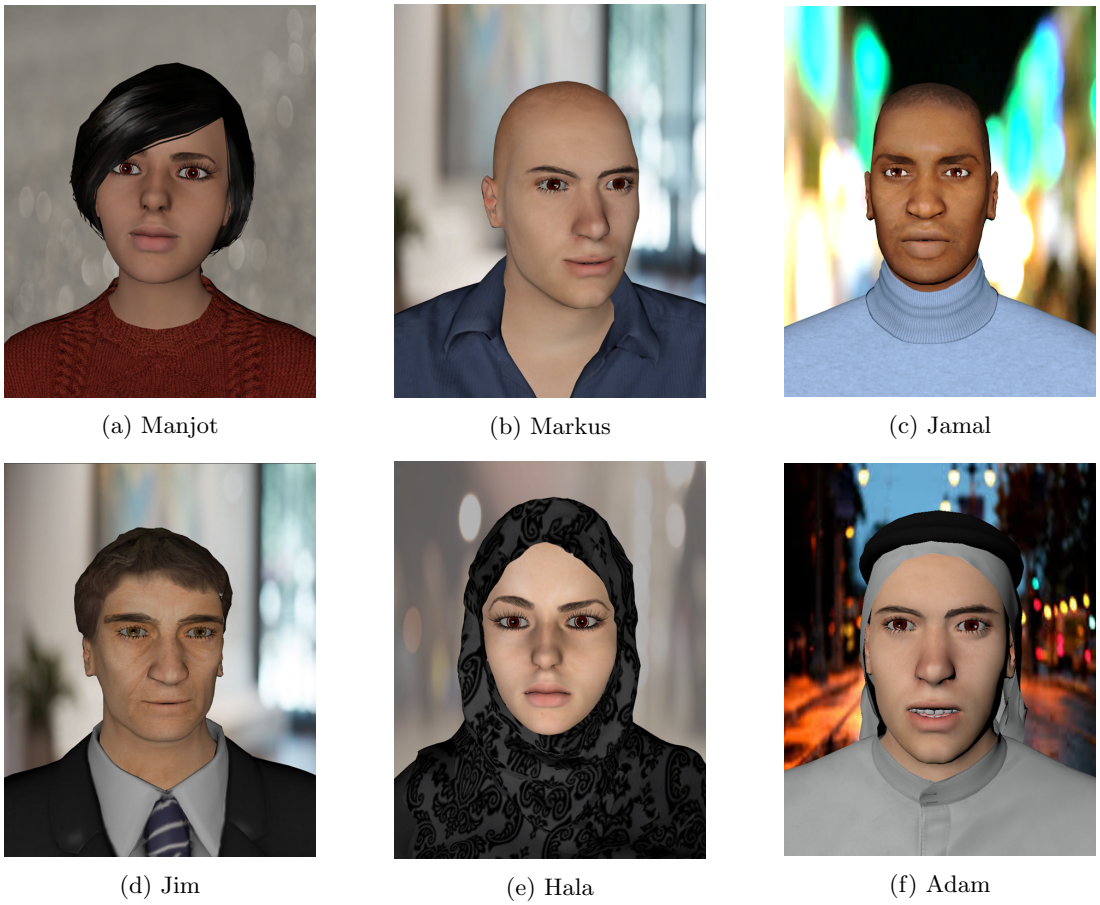
4.1.3 Creating a utterance

As shown in Figure 4.2, The AUML markup language defines an avatar using the tag <AVATAR>. The <AVATAR> tag has attributes that allow for a closest selection from the set of available avatars. The selection of avatar will determine which avatar will be uttering a response to the user. An avatar generates responses using the <UTTER>. While uttering responses the avatar can be requested to include expressions such as a wink using the <WINK> tag. As shown in Figure 4.2, there are many expressions to choose from such as surprised, smile, sad, nod,...etc. These expressions have a variety of styles. Each style contains a set of pre-defined values for the facial and bone controls in the avatar. The duration of the expression can be defined as well. The expressions can be used while speaking or during idle times. If the expression tag is empty then there is no speech and the expression is plotted independently according to the selected style and duration adding the overall duration of the utterance. If the expression tag includes text then it is included

```
<?xml version="1.0"?>
<!DOCTYPE AUML [
<!ELEMENT AUML(AVATAR*)>
<!ELEMENT AVATAR (UTTER*)>
<!ELEMENT UTTER (WINK*, SMILE*, SURP*, SAD*, SCARD*, HOPE*, DISCUST*, SHAKE*, NOD*)>
<!ELEMENT WINK (#PCDATA)>
<!ELEMENT SMILE (#PCDATA)>
<!ELEMENT SURP (#PCDATA)>
<!ELEMENT SAD (#PCDATA)>
<!ELEMENT SCARD (#PCDATA)>
<!ELEMENT HOPE (#PCDATA)>
<!ELEMENT DISCUST (#PCDATA)>
<!ELEMENT SHAKE (#PCDATA)>
<!ELEMENT NOD (#PCDATA)>

<!ATTLIST AVATAR name CDATA "Manjot">
<!ATTLIST AVATAR age CDATA  "35">
<!ATTLIST AVATAR gender CDATA "female">
<!ATTLIST AVATAR weight CDATA "69">

<!ATTLIST UTTER language CDATA "English">
<!ATTLIST UTTER mood CDATA "normal">

<!ATTLIST WINK style CDATA "1">
<!ATTLIST WINK duration CDATA "1">

<!ATTLIST SMILE style CDATA "1">
<!ATTLIST SMILE duration CDATA "1">

<!ATTLIST SURP style CDATA "1">
<!ATTLIST SURP duration CDATA "1">

<!ATTLIST SAD style CDATA "1">
<!ATTLIST SAD duration CDATA "1">

<!ATTLIST SCARD style CDATA "1">
<!ATTLIST SCARD duration CDATA "1">

<!ATTLIST HOPE style CDATA "1">
<!ATTLIST HOPE duration CDATA "1">

<!ATTLIST DISCUST style CDATA "1">
<!ATTLIST DISCUST duration CDATA "1">

<!ATTLIST SHAKE style CDATA "1">
<!ATTLIST SHAKE duration CDATA "1">

<!ATTLIST NOD style CDATA "1">
<!ATTLIST NOD duration CDATA "1">
]>
```

Figure 4.2: **AUML DTD.**

within a speech and it's selected duration will be part of the duration of the speech. This does not add to the duration of the utterance. The expressions and spoken words are plotted and animated in the order as they appear in the AUML using the hints provided. The time required for the spoken text is extracted from



(a) Manjot (b) Markus (c) Jamal

(d) Jim (e) Hala (f) Adam

Figure 4.3: **Examples of developed avatars**.
(a) Manjot is an avatar inspired from females in India, (b) Markus is an avatar inspired from middle aged Caucasian males, (c) Jamal is an avatar inspired from middle aged African males, (d) Jim is an avatar inspired from old Caucasian males, (e) Hala is an avatar inspired from young Arabic Muslim females, (f) Adam is an avatar inspired from middle aged Arabic males.

the retrieved audio of the spoken words.

4.1.3.1 The Audio of the spoken words

As sentences in the AUML are extracted they are submitted to a cloud-based text-to-speech tool, which in turn returns the audio of the spoken words. The audio is used to guide avatar lip-syncing. Typically these text-to-speech tools can be customized. For example, the Google text-to-speech project allows for requesting specific voices, genders, accent and speed of the audio. The different voices available through the original

Google text-to-speech are assigned in the system to the available avatars.

4.1.3.2 Lip-syncing spoken words

Spoken words in the utterance are lip-synced with the audio to provide a realistic utterance. A key requirement here is understanding the time indexing of individual events in the utterance. As we Know the text used to generate the audio we can use the text to help sync the lip animation. We utilize a dictionary of the sounds in words and use this to compute the timing of events in the utterance. Having prior knowledge of the duration of every possible word (or at least most common words) helps to automate realistic lip-syncing and more generally allows us to predict how long the resulting audio and video sequences should be. In order to obtain the expected duration of utterances we trained our system on the duration of every word in a dictionary using the text-to-speech engine. For the text-to-speech engines evaluated to date, the duration $t(x)$ of the spoken word x is independent of its context within which x was used. This simplifies the process of estimating the duration of the spoken phrases. The duration of the word $t(x)$ is saved in the same dictionary used to retrieve the sounds of the words x to optimize data retrieval. An audio strip generated by a text to audio engine are typically embedded in a quiet clip. The result audio duration usually includes empty audio at the beginning and the end of the audio strip. An audio clip consists of a constant number of frames (f) per second (typically 24) and the pre and post clip residue have proven to be of constant duration. The sum of the durations associated with the individual words previously trained on word is usually larger than the duration of the combined sentence of the words. To accommodate this, the duration of each word is used as a weight for the actual plot time of the word in the lip-sync animation of the sentence. The time marker of each word is calculated using equation $w(x) = t(x)/\sum_{i=1}^{n}(t(x_i))$. The duration of the word x in the actual sentence $T_s(x)$ is approximated by the weight of the word multiplied by the actual duration of the sentence $t_s(x) = w(x) * t(x)$. The marker of each word in the actual sentence $m(xs)$ is the marker for the first frame (f_0) plus the number of frames $(NF(d))$ in the duration space (d) of every word that comes before it, typically the number of frames per second is 24, and thus $NF(d) = 24 * d$. The frame marker for

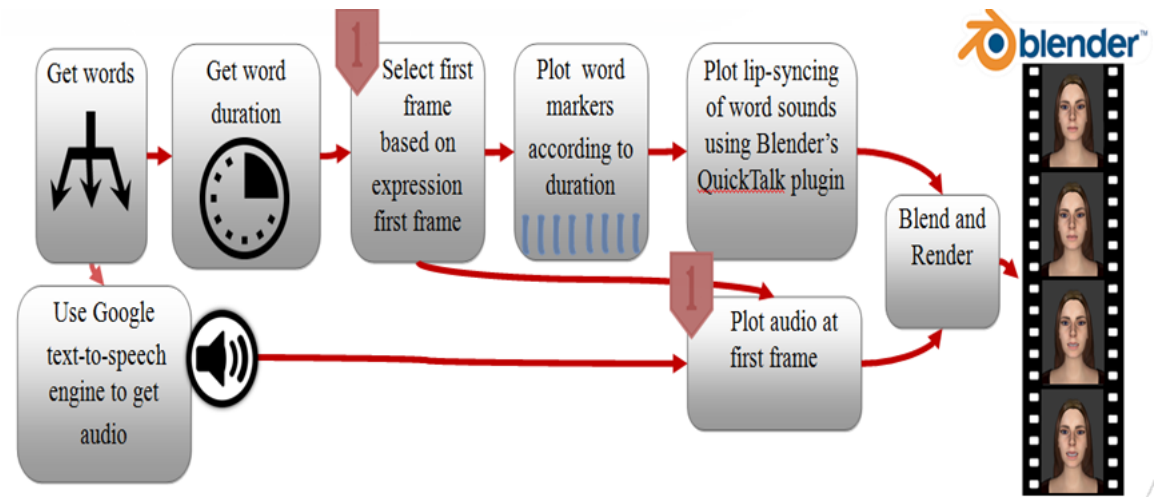


Figure 4.4: **The process of lip-syncing the spoken words.**

each word is calculated using equation 4.1

$$m(x) = f_0 + NF(\sum_{j=1}^{J<i}(w(x_j) * t(x_j)) \tag{4.1}$$

The sounds "vismes" in every word are mapped to mouth and lip key-frame shapes. These key-frame shapes are used to plot the vismes associated with each word. We utilize key-frame shapes that are part of the MHx2 facial rig exported from MakeHuman[56] and imported into Blender[102]. This automated lip-sync process is based on a manual process that uses a blender plugin called QuickTalk [113]. We automated this process and optimized the word markers based on the actual duration of the word instead of using equally divided markers. The QuickTalk plugin creates an indexed dictionary of all the words in the vismes dictionary for every lip-synced phrase. This work optimizes the vismes retrieval mechanism by using one pass for the words. The sounds plotted using key-frame shapes were based on the originally hard coded values for each visme found in MHX2. The original values created exaggerated mouth movements for each visme which did not seem to produce a realistic outcome. These default values were adjusted to create a more desired effect by changing the values using controls and observing the change in the shape of the mouth

and lips of the avatar. The lip-syncing process is described in Figure 4.4.

4.2 Building a realistic utterance state transition

Between utterances we do not want the avatar to be still. Rather we wish the avatar to engage in apparently normal motion when not engaged in conversation with a user. Furthermore, we wish the avatar to transit from this delay behavior to utterance behavior seamlessly. We accomplish this by pre-rendering and pre-loading to the local display a collection of renderings that can be played when the avatar is idle and which are designed to be combined together to make arbitrarily long sequences of idle behavior. The Avatar Delay Graph (ADG) provides a formal structure within which to encode short locally cached video sequences that can be played so as to provid an animation of the avatar between utterances. This structure also provides a mechanism within which to obscure rendering and transmission latencies which are unavoidable given the cloud-based rendering of the avatar.

We model the ADG as a labeled directed graph $G = (V, E)$, where $V = \{x_1, x_2, ..., x_n\}$ and $E = \{e_1, e_2, ..., e_n\}$. Nodes correspond to points at which specific video sequences can be stitched together smoothly and edges model individual video sequences. Each edge $e = (x_a, x_b)$ is labeled with $\tau(e)$, how long it takes the play the sequence corresponding to e. When the avatar plays the video sequence corresponding to edge e the avatar's representation within the ADG transits from x_a to x_b. Also associated with edge e is an "expressive state" $es = (s_1, s_2, .., s_p)$ an encoding of the nature of the avatar as it is perceived by a user. The dimensionality of es is avatar dependent.

Initially the avatar is in some node x_0 and has some avatar state S. When the avatar is not uttering an expression it walks the ADG in a stochastic manner as described below. When in node x it chooses from the edges departing from x. For each candidate edge e_i the avatar delay engine computes the difference from S to $es(e_i)$, $d_i = |S - es(e_i)|$. The avatar then chooses randomly from each of the incident edges with a probability inversely proportional to this distance. Specifically, with a probability proportional to $1/(d_i + \epsilon)$

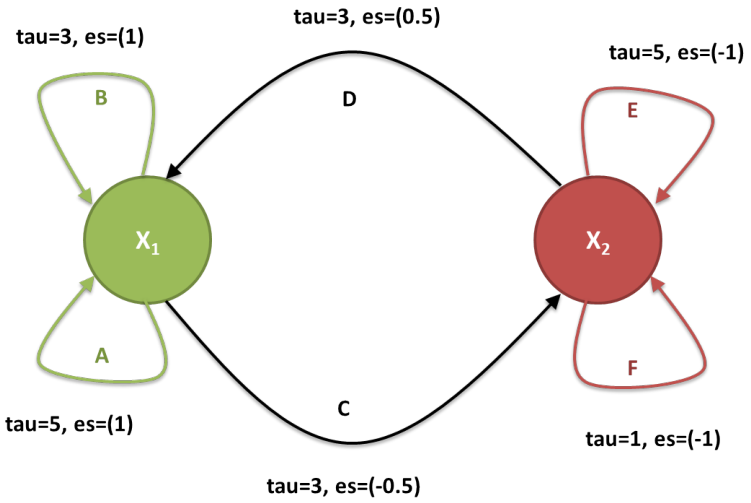


Figure 4.5: **Sample ADG and its operation.**
Here $V = \{x_1, x_2\}$ with multiple edges connecting x_1 and x_2 to themselves and transitions between x_1 and x_2. Suppose the avatar is at x_1 with $S = 1$. The next video squence to be payed will be one of A, B, and D. $d_A = 0$, $d_B = 0$ and $d_D = 0.5$. The avatar then chooses stochastically which of A, B or D based on relative probabilities $P_A = 1/\epsilon$, $P_B = 1/\epsilon$ and $P_D = 1/(0.5 + \epsilon)$. Suppose that B is chosen. Then the video sequence B is presented (duration 3 seconds), S is updated as $S' = \lambda S = (1 - \lambda)es(B)$ and the process continues.

where ϵ is a small positive constant to avoid overflow. Once a best edge e_{best} is chosen the avatar's state S is updated using $S' = \lambda S + (1 - \lambda)es(e_{best})$. Figure 4.5 provides a simple example ADG and its operation.

Vertices in the ADG are optionally labeled as being a starting or terminating node to aid in terms of merging ADG transitions and renderings with renderings associated with utterances. A node can be both a starting and terminating node. When an utterance is to be generated an appropriate terminating node in the ADG is determined based on the duration of the path $\sum r(e)$ and similarity of the chosen transitioning node to the current avatar state as described below.

When the avatar is to render some utterance with state S, a new temporary edge $E = (x_{start}, x_{end})$ is constructed. Here the x_{start} and x_{end} nodes are chosen from the set of starting and terminating nodes in the ADG. The utterance will be rendered between node x_{start} and x_{end} of the ADG. To accomplish this, we must first identify x_{start} and x_{end} in the ADG. The x_{end} node is chosen such that

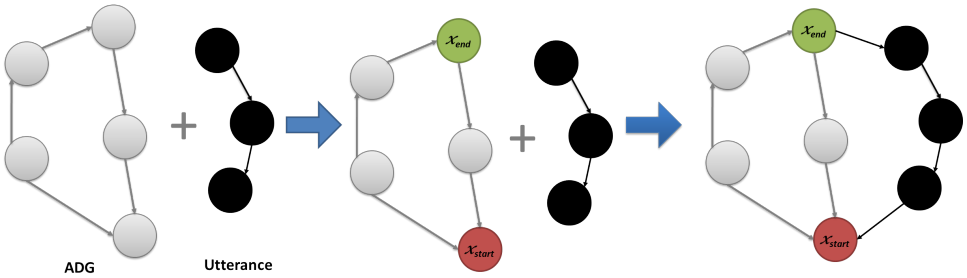


Figure 4.6: **Generating a path in ADG.**

- It is labeled e.
- It has a terminating node.
- The mean of $es((x_{end}, x_k)) - S$ is minimized

That is, when the utterance is generated it can terminate in a state where there is a good exiting edge in the ADG form x_{end}.

The choice of start node is similar, but it is also necessary to identify a node that can be accessed quickly in terms of transitions in the ADT in order to avoid the introduction of abrupt changes in the avatar's appearance. See Figure 4.6. The x_{start} node is chosen such that

- x_{start} has a starting label.
- The cost of $\sum \alpha\tau(e) + (1-\alpha)|es(e) - S|$ is minimized. Where here the sum is over the path in the ADG from the avatar's current state to the x_{start} node.

This chooses a nearby start node such that the es values are similar to the current state of the avatar S. Note that the process of selecting the x_{start} node also enables the computation of the expected delay before it is necessary to start rendering the utterance.

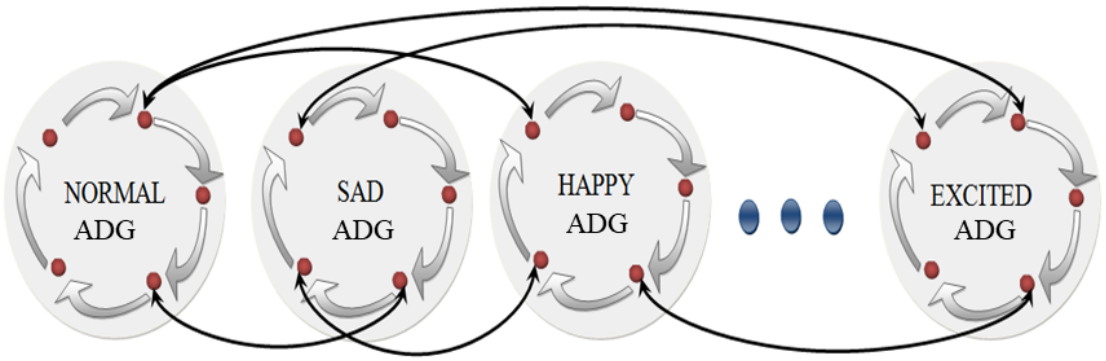


Figure 4.7: **The realistic utterance state transition.**

Once the x_{start} and x_{end} nodes have been identified the avatar begins to move deterministically through the ADG to x_{start} node following the sequence identified in the process of identifying this node. When it reaches the start node it then executes the rendered utterance and re-enters the ADG at the end node. The value of S is unchanged by this process although clearly it would be possible to associate a change in S with each utterance. Once at the end node the stochastic walk through the ADG continues until the next utterance is available and the process continues.

When not generating utterances the avatar continues to animate through one of a number of waiting states to simulate a non-engaged but nevertheless animated speaker. We can structure such waiting states to simulate emotion or mood (see Figure 4.7). Figure 4.7 also shows how common connectors allow for realistic transition. Figure 4.8 illustrates how these idle loops are combined stochastically in order to generate smooth idle sequences. The idle loops cross path to form a graph of nodes and edges.

To illustrate this point more clearly consider Figure 4.9. Here the Bored and Engaged state is structured as two idle loops. Suppose that the starting and ending nodes exist only in the "engaged" portion of the ADG. Then it would need a bridge for an utterance while it is in the "bored" portion of the ADG. Figure 4.10 presents an example illustration of the transition from bored to engaged.

In addition to a general mood in the waiting state the avatar supports automated facial expressions

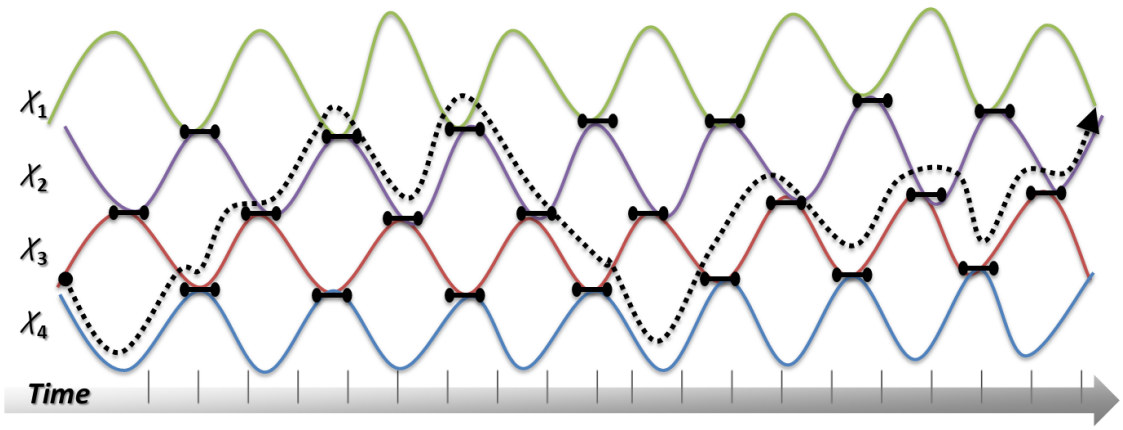


Figure 4.8: **Multiple edge state transition**
Four idle loops x_1 through x_4 are illustrated with transitions between x_1 and x_2, x_2 and x_3, and x_3 and x_4. Potential idle loops in each of x_1 through x_4 shown in different colors an a possible stochastic path is shown as a dotted line.

manipulation. We can automate expression based on predefined rotation and translation values for our facial rig or a given set of values at any time. The importance of this automation is that expression manipulation and animation can be done on demand without the need for 3D GUI manipulation which can be extremely time consuming.

4.2.1 Preparation of pre-animated video sequences

For a given avatar the ADG structure of the corresponding animation sequences are manually constructed and pre-animated. The edge transitions are pre-rendered, stored locally and presented based on the defined graphs and smooth transitions.

4.3 Summary

This work puts an interactive face on a robot. In order to accomplish this a standard text-to-speech generation system is combined with a 3D avatar (puppet) whose facial animation is tied to the utterance being

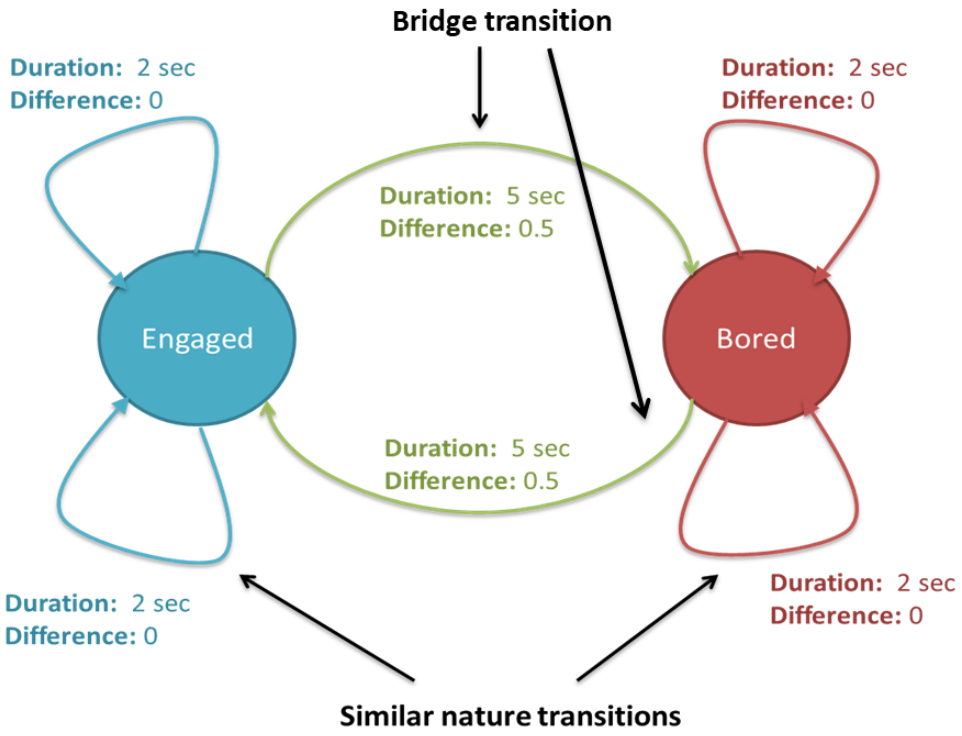


Figure 4.9: **An example ADG, representing two different emotional states.**

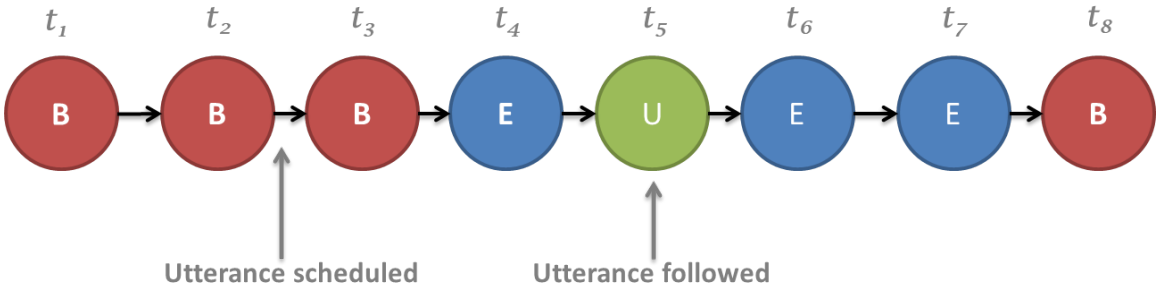


Figure 4.10: **An example transition sequence.**
Initially the avatar is bored in the bored state. At t_2 an utterance is scheduled for after t_4. The only start node is in the engaged state, so the avatar transits to the engaged node and executes the utterance at t_5. After the utterance the avatar returns to the engaged state where it continues to walk the graph.

generated. In order to embed emotional state and other out of band information, messages presented to the text-to-speech module are embedded within an XML structure known as the Avatar Utterance Markup language (AUML) that allows the user to tune the nature of the avatar animation so that different emotional states can be simulated. The expressions and spoken words are plotted and animated in the sequential order as they appear in the AUML. Between utterances the avatar is not still. Rather the avatar appears in apparently normal motion when not engaged with a user. Furthermore, the avatar transits from this delay behavior to utterance behavior seamlessly and back again after the utterance. We accomplish this by pre-rendering and pre-loading to the local display a collection of renderings that can be played when the avatar is idle and which are designed to be combined together to make arbitrarily long sequences of idle behavior.The Avatar Delay Graph (ADG) provides a formal structure within which to encode short locally cached video sequences that can be played so as to provide an animation of the avatar between utterances. This structure also provides a mechanism within which to obscure rendering and transmission latencies which are unavoidable given the cloud-based rendering of the avatar.

Chapter 5

Cloud-based rendering and real-time display

In order to provide real time display many of the expected utterances can be pre-rendered and played in real-time with no latency for spoken words or transition from one expression state to another. When a new utterance is required or requested then rendering of the animation will need to be done at the time of request. Cloud-based utterance recognition, responding to the utterance, and rendering and transmitting the resulting video sequence introduces latency. The following sections describe how we seek to reduce the impact of this latency.

5.1 Optimizing Blender files for real-time rendering

There are a number of setting that are turned on by default in Blender files that add unneeded latency on rendering the animations [123]. These setting can increase the quality of the rendered images or videos, but this feature is not critical here. Thus the following settings were disabled or reduced:

- Ray Tracing was disabled: It's not uncommon for ray tracing to multiply render times by a factor of ten.
- SubSurf levels were reduced.
- Soft shadows were disabled: soft shadows can consume hours of rendering, so the soft size and the samples for all spot lamps is set to 1 to reduce render time.
- Ambient Occlusion was disabled: Ambient occlusion can add realism to a scene by simulating indirect shadows. However it's computationally expensive.
- Simplification was enabled: The Simplify option in Blender allows the user to set global limits on subdivision, shadow samples and Ambient Occlusion(AO) and Soft Shadows (SS).
- Blurry reflections was disabled: By default blurry reflections are turned off, as they are computationally expensive.
- Subsurface Scattering was disabled: Subsurface Scattering multiples rendertimes. Turning this off could reduce render times by a factor of 6.
- Shadows were disabled.
- Anti Aliasing was disabled: Enabled by default, this option ensures that all the edges in your scene are smooth and unjagged. Render time is reduced by approximately half by turning this feature off.
- Tiles setting was increased: Increasing tiles will ensure that all cores work on the render until it's finished without one core finishing before another.
- Baking textures was used: As the avatar have few shadows this data cab be baked into textures so that Blender only needs to calculate those values once.
- Materials were made non-traceable: As the avatar does not need shadows or reflections this can be disabled.

- Image dimensions was reduced: Setting the resolution percentage to 50% will render the scene up to four times faster.

5.2 From compute engine to render farm

Google cloud platform provides a compute engine that allows for the creation of virtual machines with various levels of computation power (number of virtual CPUs, the inclusion of a GPU, size of RAM and disk space) and different choics of operating systems. The virtual machines can be customized based on the user needs. This work requires the creation of K identical virtual machines for multiprocessing and transforming these headless virtual machines into rendering engines.

In order to create a blender rendering farm from a compute engine on the Google cloud platform, a number of steps need to be taken. First, a Google cloud platform account and a project associated with this account within the platform is created using compute engines that can be transformed into rendering farms. The user associates instances of virtual machines to the created project. The machine type selected for this project is the Ubuntu 16.04.4 LTS (Xenial Xerus). Every instance is located in a physical zone of choice. The zone should be selected so as to minimize the geographical distance between the location of the instance and the local machine communicating with the instance. Individual machines can vary in the number of virtual CPUs used. While more virtual CPUs mean more computational power, it also means that more cores need to be available for allocation at a single point of time. This could possibly lead to delays in availability of an instance as the system must wait until an appropriate number of cloud cores are available simultaneously. A given Google project could have K instances of 1vCPU, one instance of KvCPU or many instances of KvCPU, For example, since the created instance will be used as a pool of instances performing the requests of a multiprocessing ROS node located on the local machine, the independence of the multi-processed tasks suggests having K instances of a specific virtual machine for this application. Thus, this work utilizes n machine instances with KvCPUs and 1 GPU in each instance.

5.2.1 Headless instance rendering

By default the instances created in a project on the Google cloud platform do not include a GUI, graphics display device or an audio playback device. Individual instances are headless servers with computational power and are described as compute engines. This work requires rendering engines. Rendering animations with Blender requires an X server, display screen and audio sink. To create a rendering engine from a compute engine a dummy audio sink needs to be created and activated and an X server needs to be started using a virtual display screen and assigning it values for resolution and color. Unfortunately off screen rendering can not make use of OpenGL which allows for the use of hardware acceleration. In order to enable rendering each instance is provided with VirtualGL [124]. VirtualGL is a software that can forward off-screen rending requests to the GPU for hardware acceleration. VirtualGL requires two displays (a 3D display to render from and a 2D display to render to) and a real X server. So, in addition to the previously mentioned virtual display and audio dummy sink, each instance requires a real X server running and a virtual 3D display linked to the GPU driver.

5.3 Distributed rendering in the cloud

First, we observe that we can parallelize the rendering process of the avatar. We can break the entire sequence up into smaller pieces, render those pieces in parallel in the cloud, and then present the rendered clips in sequence to the user. If we approximate the relationship between playing time T_p and rendering and network latency time T_r as a multiplicative factor (k), then $T_p = kT_r$ and if we have a pre-defined acceptable rendering latency (T), then we he have T seconds to render the first clip. This latency will result in kT seconds of played video. The second rendering stream also starts at time 0, and has the initial latency plus the time of the first clip's playing time within to render, resulting in $T + kT$ seconds of rendering time and $k(T + kT)$ seconds of rendered video from the second processor. Or more generally, $T_r^n = T + \sum_i^{n-1} T_p^i$ and

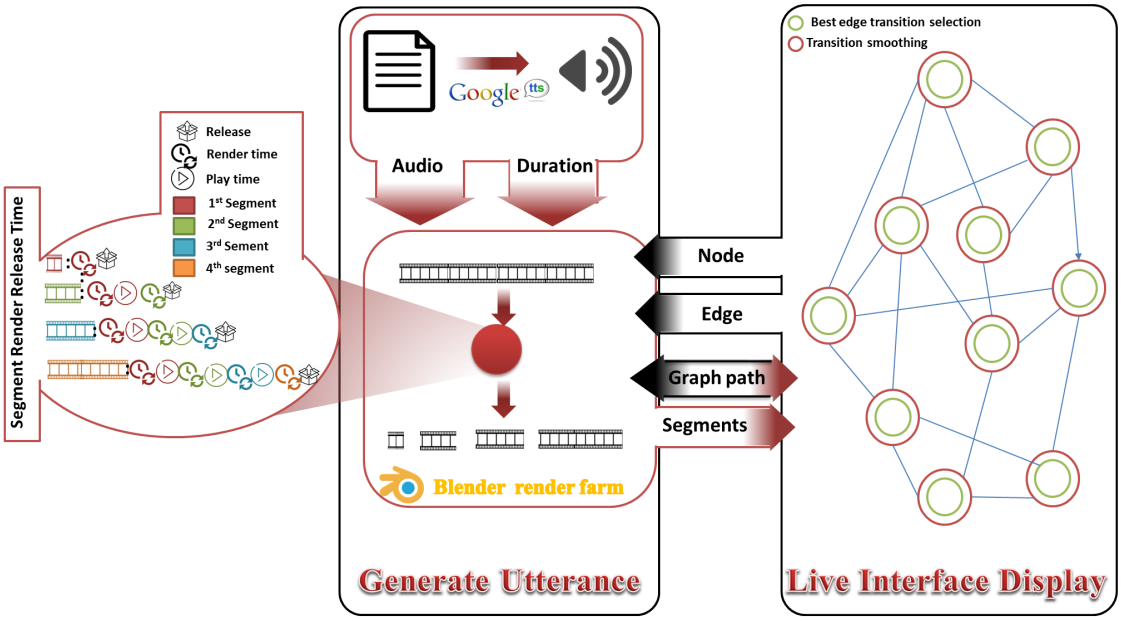


Figure 5.1: **A parallel multiprocess generation of the utterance to reduce latency in display.**

under the assumption that $T_p = kT_r$ then $T_r^n = T + k\sum_i^{n-1} T_r^i$.

Second, we observe that delays in the cloud are estimable, but are stochastic. So there is some small probability that the next clip to play may not be available when it is needed. Furthermore, we wish to simplify the problem of stitching the clips together when playing them so arbitrary clip points are to be avoided. So instead of using the break points as identified above we treat the theoretical break points identified above as maximum values and seek the next earliest point in the utterance that corresponds to a word break or punctuation. This gives us more natural break points in the rendering.

More rigorously, suppose we have a break point T_p identified through the process described above. Then rather than breaking splitting the input at this point we scan backwards looking for the first break in the input, either the first punctuation or space between words. Call the time moving backwards in the clip until the first word break T_B and the time unilt the first punctuation T_P. We weight each by k_B and k_P respectively and choose the minimum of $T_B k_B$, $T_P k_p$ and V_{max} as the break point. Here V_{max} is a maximum weighted distance to backup. Note that – especially for very short duration clips – one or more of T_B and

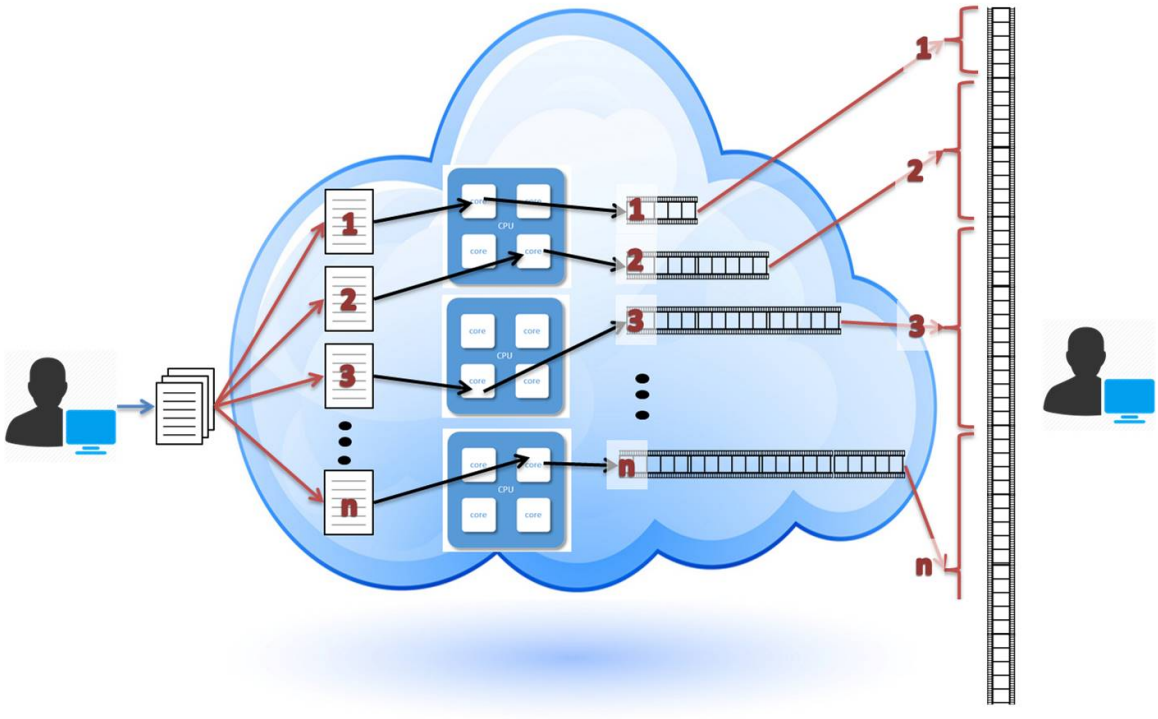


Figure 5.2: **The multi-process rendering system.**
A clip to be rendered is split into non-uniform length pieces and distributed to the rendering farm. The rendered sections are pieced together in the correct display sequence.

T_P may not exist.

Third, we observe that we can 'stall' the video being generated should it be necessary by rocking the video to be played back and forward a small amount to avoid the avatar becoming 'stuck' or 'stuttering'. Such rolling the video backwards and forwards will always be consistent with the video being played and can be used to hide unexpected latency.

5.4 Summary

When a new utterance is required then rendering the animation will need to be done at the time of request. Cloud-based utterance recognition, responding to the utterance, and rendering and transmitting the resulting video sequence introduces latency. There are two things that can be done to reduce this latency. The

first is turn off some settings on the Blender engine that add unneeded latency on rendering the avatar animations. The second is to multiprocess the rendering on the cloud. To do that, we generate a cloud-based Blender rendering farm where each instance will be used as a pool of instances performing the requests of a multiprocessing ROS node located on the local machine. By default, instances created on a cloud platform do not have a GUI, graphics display device and audio playback device. This work requires rendering engines. Rendering animations with blender requires an X server, display screen and audio sink. To create a rendering engine from a compute engine a dummy audio sink needs to be created and activated and an x server needs to be started using a virtual display screen that supports graphical rendering. Each instance is used to render different segments of the animation in parallel, reducing the time taken to render the avatar. These segments are then played in the correct sequence. When applying the multiprocessing to the rendering of our animations we divide the required rendering into components that can be rendered separately and begin displaying the rendered utterance as soon as we possibly can. A key question becomes how to distribute the rendering over the available processors. This work displays the first segment based on an acceptable latency and allows each subsequent segment to continue to render during rendering and display of all of the segments previous to it. This allows for a reduced latency and more efficient use of cloud resources.

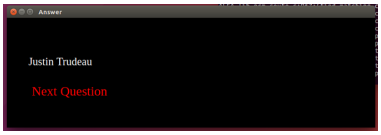
Chapter 6

Human-robot interaction user study

A key issue in the development of an avatar for human-robot interaction (HRI) is the process of evaluating its efficacy. Although demonstrations of the system in operation are desirable, determining if the approach is useful requires some set of quantitative testing/evaluation. The question of evaluating some novel interaction technology for human-machine interaction arises commonly in the field of human-computer interaction (HCI) and can be applied to HRI as well. In HCI standard approaches (e.g., [125][126][127][128][23][116][25][17]) now exist in terms of providing quantitative evaluation of such technologies. This thesis utilizes an approach similar to that used in a study by Wen-Yau et al. [127] that evaluated the performance of an intelligent agent by collecting empirical data and participant feedback through questionnaires. Here a similar methodology is used to evaluate the realistic avatar approach relative to other related approaches. Specifically, four approaches are compared, in which voice-based participant queries are responded to using a text-only response (T), an audio-only response (A), an avatar response that relies on a cartoon 3D avatar (CA), and a realistic 3D rendered avatar (RA). Figure 6.1 shows the interfaces for these tools. The interfaces use a common underlying speech recognition and knowledge engine to obtain text responses for participant queries but differ

in how responses are presented to the participant. The text interface displays the response as displayed text on a laptop screen and then displays another text message that indicates that the interface is ready for the next question. The audio interface uses the text response to generate an audio response; it plays the audio response and then displays a text message on the laptop screen indicating that the interface is ready for the next question. The cartoon avatar interface used in the participant study differs from the cartoon avatar described earlier in this thesis which animates a simple avatar on the screen. For the cartoon avatar response animations are pre-rendered and are available on the local machine. The cartoon avatar presents in two states; mouth closed and mouth open, and this provides a simple and computationally inexpensive lip-sync to the responses. The cartoon avatar uses the generated audio response and plays this synchronized with the animated character. The realistic avatar follows the approach described earlier in this work where lip-synchronization is animated based on the utterance and an ADG is used to animate the avatar between utterances. The realistic avatar uses the text and audio generated for lip synchronization animation. For the realistic avatar, a rendering farm on the cloud is used to provide an actual lip synchronization of all the sounds in the response in real-time. The realistic avatar then transitions to a utterance starting node to play the audio and rendered animation. The starting and ending node are determined prior to the cloud rendering.

A pre-study questionnaire was used to obtain a baseline of each participant's previous experience, perceptions and expectations in terms of HRI. A second questionnaire was completed by the participants after interaction with all of the interfaces to evaluate the participant's perception of them. The participant study extracted information related to the application itself and also concerning the participants' perception of the application. Information related to the application itself includes the accuracy of the system responses and the timing. Information related to the participants' perception of each interface investigates the participants' preference among the different interfaces and surveys the participants on the acceptable venues in which these interfaces might be used. The empirical evaluation and analysis follows methods detailed by MacKenzie [129]. Ethics approval for this study was granted from the office of Research Ethics of York University. A copy of



(a) Text interface (T).

(b) Audio interface (A).

(c) Cartoon avatar interface (CA).

(d) Realistic avatar interface (RA).

Figure 6.1: **The participant study interfaces.**
For each interface the participant asked questions using spoken words and received a response. For (a) the text interface (T), responses were printed text. For (b) the audio interface (A), responses were by audio only. For (c) the cartoon avatar interface (CA), responses were by a cartoon avatar. For (d) the realistic avatar interface (RA), responses were by a realistic avatar.

the informed consent form and the questionnaires used can be found in Appendix A. Quantitative timing and accuracy data was also collected during the participant's interaction with the interfaces.

6.1 Method

6.1.1 Participants

Twenty four English-speaking speakers participated in this study. The participants were divided randomly into four groups to counterbalance the order of testing and to offset learning effects. Each participant experienced all four interfaces. Each group contained 6 participants. There were 3 males and 3 females in each group. Participants were between the ages of 18 to 34 years ($\bar{x} = 22.1$). Participants' education

level ranged from a four year bachelor degree to a PhD degree. In general the participants expressed that they were highly experienced with textual knowledge engines such as the Google search engine ($\bar{x} = 1.66$, on a scale from 1 = highly experienced to 7 = not experienced), moderately experienced with respect to audio-based knowledge engines such as Siri and Alexa ($\bar{x} = 4.6$, on a scale from 1 = highly experienced to 7 = not experienced) but had little to no experience with animated intelligent agents ($\bar{x} = 6.1$, on a scale from 1 = highly experienced to 7 = not experienced). Participants received a gift card for ten dollars as an incentive for participation.

6.1.2 Apparatus

The evaluation used two questionnaires to gather information about and from the participants. The first was provided prior to the experiment itself and was used to gather general details about the participants including age, gender and previous interactions with similar AI. The second questionnaire was provided after the experiment and gathered information about the participants' experience in the experiment. Both questioners were provided using a computer and were created on-line using Google forms. Copies of the questionnaires can be found in Appendix A. Testing was performed using a laptop (a Hewlett Packard with Intel Core i7-8550U Processor at 1.8 GHz processor, 16 GB DDR4 (2-DIMM) RAM, 1 TB 7200 RPM SATA Hard Drive and a 15 inch screen). For audio input and audio output, the laptop's microphone and speaker were used respectively. The laptop used software developed under Ubuntu 16.04.3 LTS (Xenial Xerus) that contains a tool to animate the 3D avatar response, a tool to animate the 3D cartoon avatar response, a tool to obtain audio responses and a tool to obtain text responses. Figure 6.2 shows a subject using the realistic avatar interface.

Figure 6.2: **A participant interacting using the realistic avatar interface.**

6.1.3 Procedure

Each session was performed separately in a single sitting. In each session the participant was invited to take a seat at a desk with a laptop. Each participant was briefed on the purpose of the experiment and read and signed an informed consent form (see Appendix A). Using the computer, the first on-line questionnaire was presented and the participant completed it. Upon completing the first questionnaire, the participant was shown their first interface. Each participant was asked to use the interface to ask a list of questions of the interface. Questions asked by the participants were presented to the participant as printed text on a sheet of paper. Each interaction with a given interface consisted of the participant asking ten questions to the interface. As the interactions with one interface was completed, the participants moved on to the next interface. The interfaces were presented following the order given Table 6.1. Following the forth interface, the participants completed the on-line exit questionnaire. All participants asked the same set of 40 questions, broken down into 10 groups. Each group contains four questions of a similar nature. The full set of questions

	First Interface	Second Interface	Third Interface	Forth interface
Group1	T	A	CA	RA
Group2	RA	T	A	CA
Group3	CA	RA	T	A
Group4	A	CA	RA	T

Table 6.1: **Counterbalancing the participant interaction groups.**
Here T is the text interface, A is the audio interface, CA is the cartoon avatar interface, and RA is the realistic avatar interface.

are given in Figure 6.3. When presenting a specific question to an interface, participants were allowed up to three attempts per question. If the participant was unable to ask a specific question after three attempts, this failure was recorded and the participant moved on to the next question. No practice period was given. GoStats[130] was used to analyze the collected data.

6.1.4 Design

As the individual question categories are uninteresting we average quantitative measures related to timing over the question categories. This results in a within-subjects design with one factor(interfaces) having four levels. The four levels are the Text (T), Audio (A), Realistic Avatar (RA) and Cartoon Avatar (CA) interfaces. The dependent variables are the participant input time, response generation time, query failure rate and participant attentiveness were scored. Data was collected using the experimental tool, except for the participant attentiveness which was scored by an experimenter. The total number of interactions is 4 interfaces $\times$ 10 question groups $\times$ 24 participants = 960 questions (interactions).

Figure 6.4 summarizes the definition of, and relationship between, the different events in a given interaction. The *interaction time* is the time required to complete an interaction. It starts when the participant begins to ask the question and ends after the system completes its response. The interaction time includes

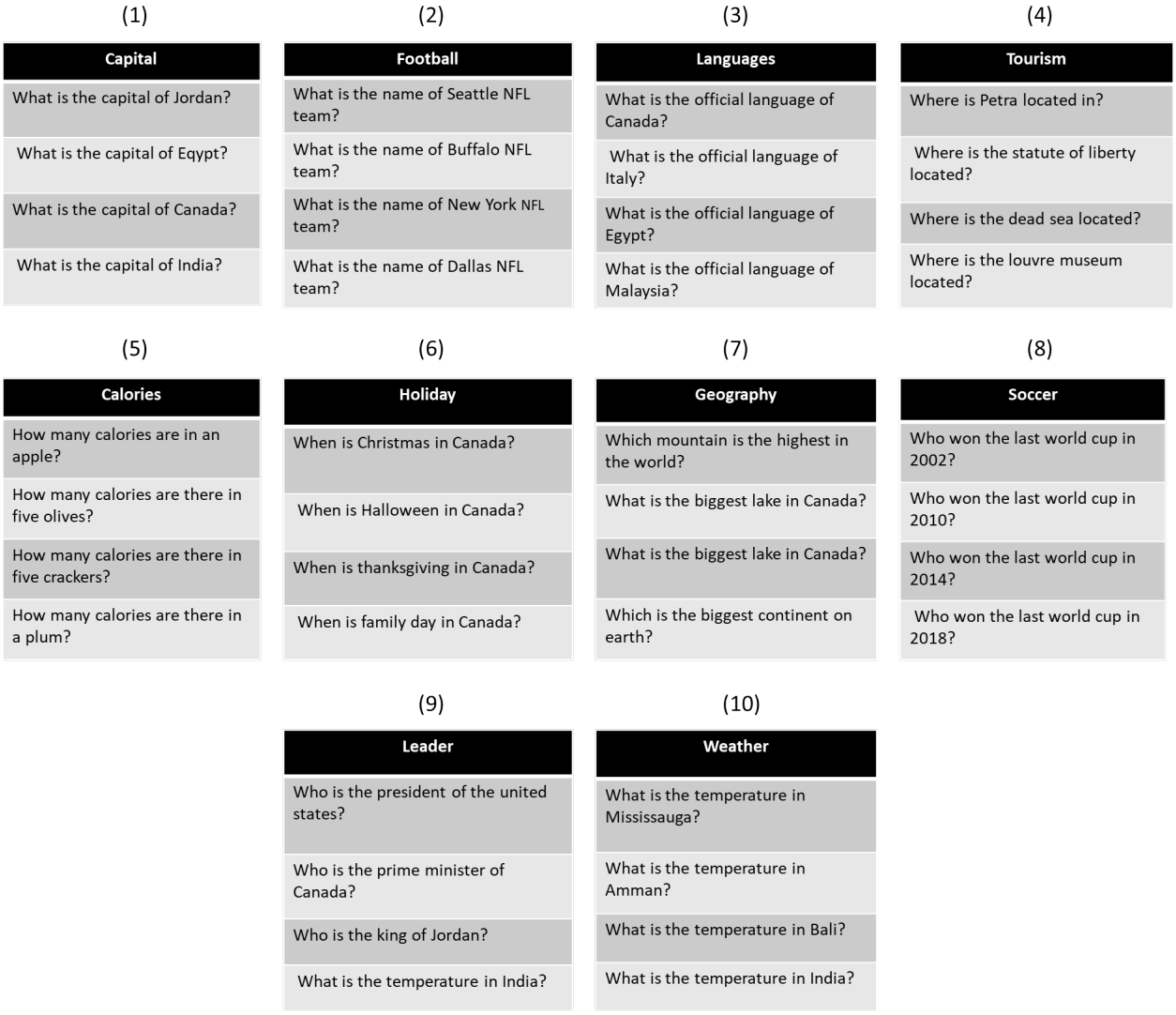


Figure 6.3: **The list of questions in each category.**
For a given interface, the participant asked one question from each list.

the participant query time, the speech recognition time, the text response time, the utterance generation time and the utterance response time. These time durations were automatically recorded by the system and are defined as follows:

- *Participant query time (input time)*: The time it takes the participant to speak a question.
- *Speech recognition time*: The time the system requires to recognize the question as text.

- *Text response time*: The time required to get the response back as text from the knowledge engine.

- *Utterance generation delay time*: The time required to begin presenting the answer to the participant. Utterance generation delay starts after a text response has been received and ends when the system loads the utterance for presentation. Utterance generation delay time differs from one interface to another. In the case of the text interface, it is the time required to load the text response for presentation. In the case of the audio interface it is the time required to generate the audio response and begin to present it. In the case of the cartoon avatar interface it is the time required to generate an audio response and begin to play it in sync with the cartoon avatar animation. In the case of the realistic avatar interface it is the time required to generate an audio response, render an associated animation so that it can begin to play synchronized with the realistic avatar animation.

- *Utterance production*: the time required to provide the response to the participant. Utterance response starts after an utterance has been generated and ends after the system has completed presenting it. As with the utterance generation delay this value is very dependent on the type of interface.

New trials initiate a new interaction and the clock is reset. This study considers two of the timings shown in Figure 6.4. These timings are the participant query time (input time) and the response generation time (the combination of speech recognition time, text response time and utterance generation delay). This study also considers quantitative measures of participant performance and engagement. These are described below.

1. *Input time*: Participant input time is the time in seconds required by the participant to ask a question. The entry of questions was by speech.

2. *Response generation time*: The response generation time begins after the participant finishes asking the question and ends after the system starts to utters the response. The response generation time is measured in seconds.

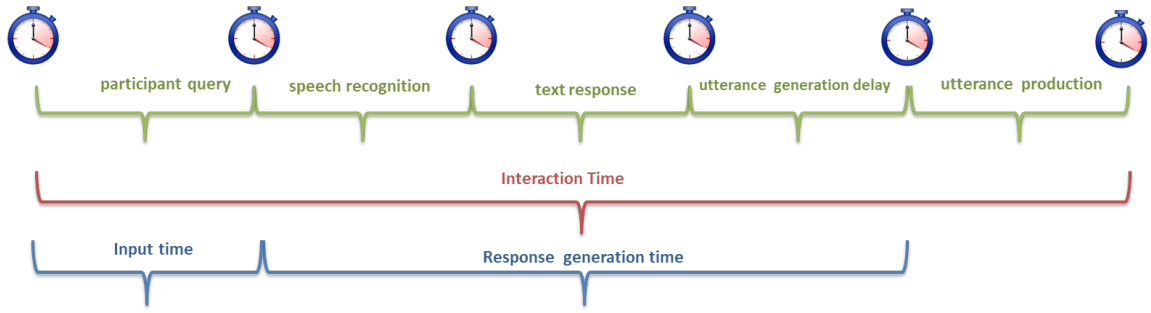


Figure 6.4: **Illustration of the interaction time line.**

3. *Query failure rate*: The participant is allowed three trials for every question. The query failure rate of a given question is 1 if all trials fail, 0.6666 if 2 trials fail, 0.33333 if 1 trial fails and 0 if the question succeeds on the first trial.

4. *Participant attentiveness*: Participant attentiveness is the percentage of time within an entire interaction that a participant was looking in the direction of the display. During an interaction there may exist intervals of time within which a participant is not completely facing the display, and there was interval of time where the participant is facing the display. The summation of the intervals of time within which the participant was completely facing the screen is represented as a fraction of to the overall interaction time. This measure was estimated manually by an observer of who clicked on a tool that recorded the time of transitions from attentive to inattentive and vice versa.

6.1.4.1 Questionnaire design

Participants' reactions can have a strong impact on the design and development of an interface[131][132][133]. Questionnaires provide information regarding participants' preferences and ideas about the design in many stages of the interface development. The questionnaire completed prior to the experiment focused on collecting information about the participant. Tips provided in an online website by Zhao[134] were used to help design the questionnaires used in this study. This study identifies the participants age, gender, level

Category	Quality metric
Functionality	Executes requested tasks
	Accuracy of output
	General ease of use
Humanity	Convincing
	Satisfying
	Natural interaction
Affect	Makes tasks more interesting and fun
Ethics and behavior	Trustworthiness

Table 6.2: **Quality categories and metrics used in the study.**

of education and level of previous experience with similar interfaces, so that the results of the study can be associated with the obtained information.

After interactions with the four interfaces were completed the post-experiment questionnaire focused on a number of the quality metrics described in [131] and shown in Table 6.2 and illustrated in Figure 6.5. The questions in the post evaluation questionnaire are taken from or inspired by the work of Jaferian[135] and other post-evaluation human-computer interaction including [136][137][138][131] with a focus on our selected quality metrics. Copies of both questionnaires can be found in Appendix A. In terms of the design of the post-study questionnaire our study used a number of questions from the categories overall reaction to the software, terminology and system information, learning and system capabilities, from the Questionnaire for User Interface Satisfaction (QUIS)[136] that are applicable to our study. The study also makes use of some questions in the usefulness and ease of use categories from the Perceived Usefulness and Ease of Use Questionnaire (PUEU)[138] that are applicable to our study. The study also makes use of questions from the Computer System Usability Questionnaire (CSUQ) and the After-Scenario Questionnaire (ASQ) [137]. These questions cover the functionality, humanity, ethics and behavior metrics presented in Figure 6.5.

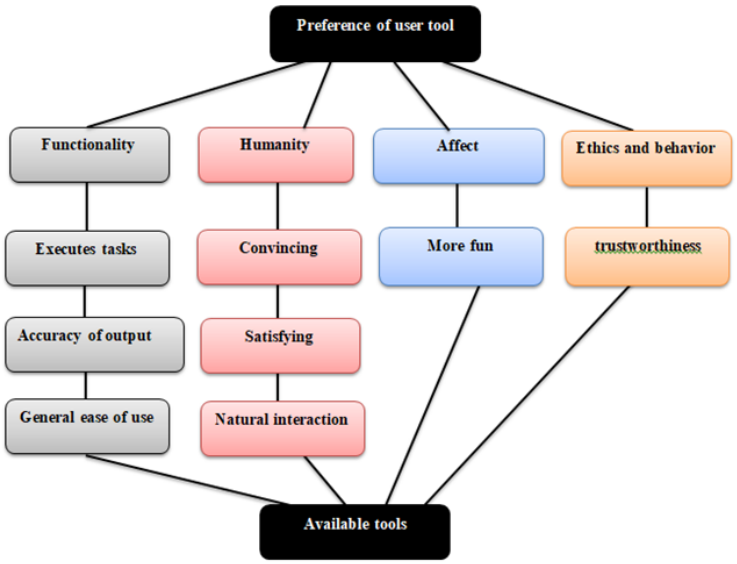


Figure 6.5: **Quality categories and metrics used in the study.**

6.2 Results

For purely quantitative data, a repeated measures ANOVA was performed. For other datasets a Friedman non-parametric test was used. An application by MacKenzie called GoStats [130] was used to analyze the collected data using the required method of analysis. This tool was used instead of more general statistic tools such as SPSS and R because it is customized for human-computer interaction.

6.2.1 Input time

As shown in Figure 6.4, input time is the time the participant needs to ask a given question. In the study, input time is equal to the duration of the recognized audio uttered by the participant. The means for input time by interface were, Text (T): 2.83 (s), Audio (A): 2.12 (s), Cartoon Avatar (CA): 2.72 (s), and Realistic Avatar (RA): 2.68 (s) as shown in Figure 6.6. The main effect of interface type on input time was

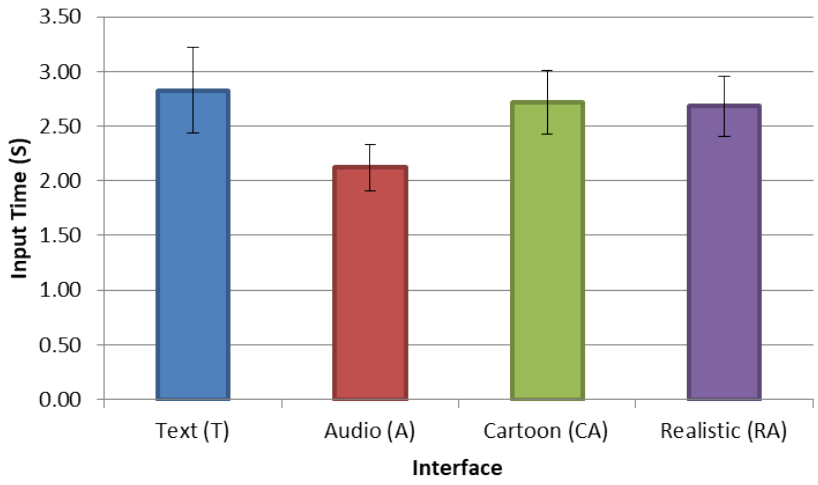


Figure 6.6: **Mean input time (s) by interface. Error bars show ±1 SD.** *Audio (A) has a significantly lower input time than the other interfaces.*

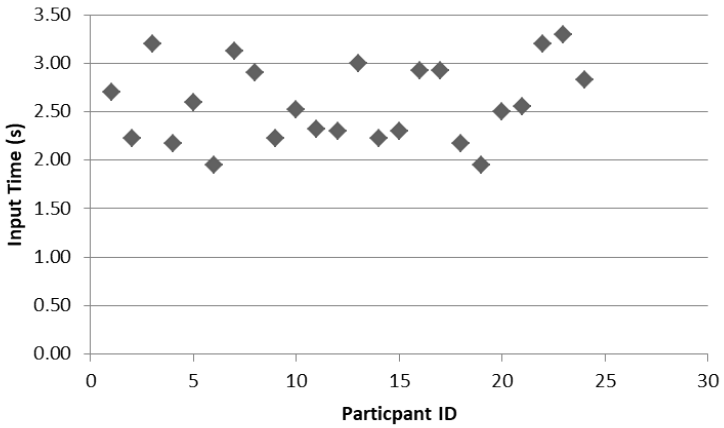


Figure 6.7: **Mean input time (s) by participant.**

statistically significant ($F_{3,\,60} = 25.9$, $p < .0001$). A Bonferroni-Dunn [129] post hoc test revealed that all pairwise comparisons with the audio interface were statistically significant. Input time in the audio interface was significantly lower than the other interfaces, indicating that participants spoke faster when asking the questions when using the audio interface. Post hoc test values can be found in Appendix B.

Figure 6.7 shows the mean input time by participant. There was variability in input time by participant. Different individuals speak at a range of different speeds.

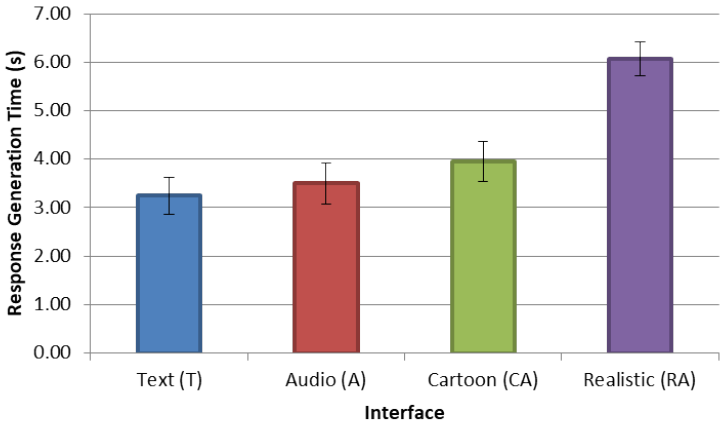


Figure 6.8: **Mean response generation time (s) by interface. Error bars show ±1 SD.** *Post hoc testing reflected that all pairwise comparisons are statistically different except text(T) with Audio(A).*

6.2.2 Response generation time

The response generation time starts after the participant asks a question and ends once a response is ready to be presented. The means for response generation time by interface were Text (T): 3.25 (s), Audio (A): 3.50 (s), Cartoon Avatar (CA): 3.96 (s), and Realistic Avatar (RA): 6.07 (s) as shown in Figure 6.8. The main effect of interface type on response generation time was statistically significant ($F_{3,\,60} = 188.2$, $p < .0001$). A Bonferroni-Dunn post hoc test revealed that all pairwise comparisons were statistically significant except for the text and audio interface pair. Post hoc test values can be found in Appendix B.

Figure 6.9 shows the response generation time by participant. The response generation time varies based on cloud availability and network speed.

6.2.3 Participant attentiveness

Participant attentiveness is the percentage of the interaction time (see Figure 6.4) that the subject was facing the laptop display during an interaction. The means for percentage of participant attentiveness by

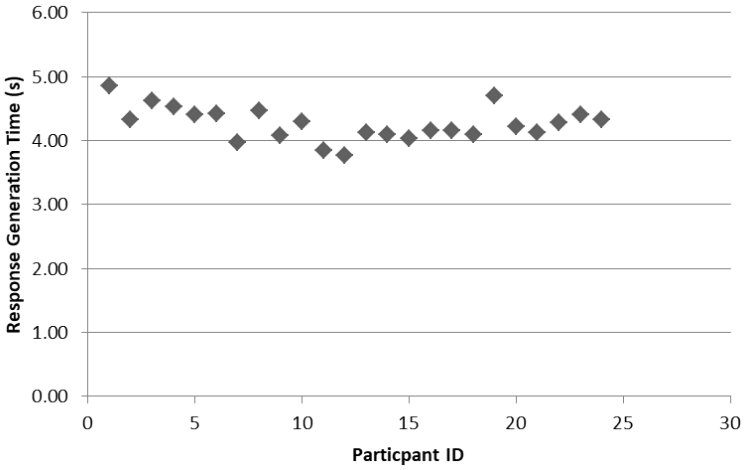


Figure 6.9: **Mean response generation time (s) by participant.**

interface were Text (T): 55%, Audio (A): 32%, Cartoon Avatar (CA): 43%, and Realistic Avatar (RA): 48% as shown in Figure 6.10. The main effect of interface on attentiveness was statistically significant ($F_{3,\,60} = 2345.82$, $p < .0001$). A Bonferroni-Dunn post hoc test revealed that all pairwise comparisons were statistically significant. Post hoc test values can be found in Appendix B. Figure 6.10 shows higher attentiveness with the text interface, which is expected because answers were read through the text presented to the participants on the interface. The participants were least attentive with the audio interface, which is also expected due to the interface having little visual cues to follow. Participants were presented with the question to ask on paper and thus were not attending to the display when reading the question to ask. This explains the relatively low attentiveness range.

The participants asked ten questions, one from each category, with each interface. Participants became less attentive the longer they interacted with the interface and this decline was well modeled using linear regression (Figure 6.11). R^2 values were Text (T): 0.9815, Audio (A): 0.9331, Cartoon Avatar (CA): 0.9349, and Realistic Avatar (RA): 0.9874 with regression lines Text (T): $y = -0.0179x + 0.677$, Audio (A): $y = -0.0102x + 0.3761$, Cartoon Avatar (CA): $y = -0.0108x + 0.4935$, and Realistic Avatar (RA): $y = -0.0157x + 0.5676$.

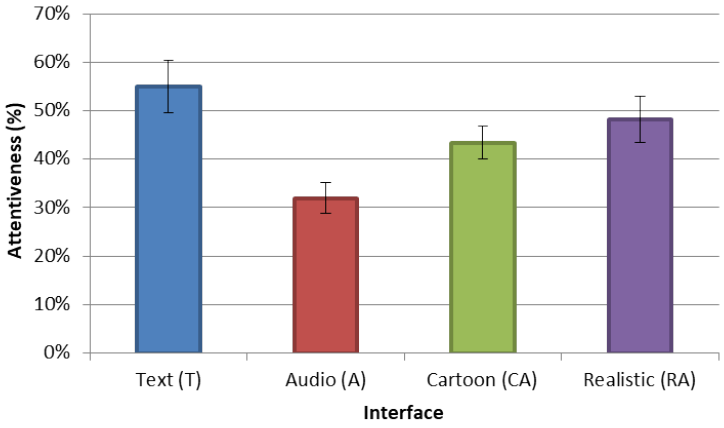


Figure 6.10: **Mean participant attentiveness (%) by interface. Error bars show ±1 SD** *All pairwise comparisons are statistically different.*

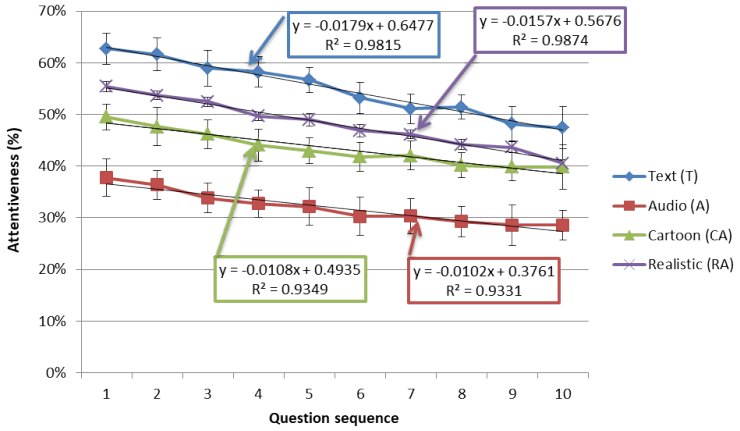


Figure 6.11:
Mean participant attentiveness (%) by request and interface. Error bars show ±1 SD. *Attentiveness decline with the number of questions the participant asked of the interface.*

6.2.4 Query failure rate

Query failure rate is the percentage failure (not getting a successful response) on trials for every question being asked. The means for Query failure rate by interface were Text (T): 3% , Audio (A): 12%, Cartoon Avatar (CA): 3%, and Realistic Avatar(RA): 4% as shown in Figure 6.12 . The main effect of interface on query failure rate was statistically significant ($F_{3,\,60} = 6.228$, $p < .001$). A Bonferroni-Dunn post hoc test revealed that all pairwise comparisons with audio were statistically significant. Post hoc test values can be

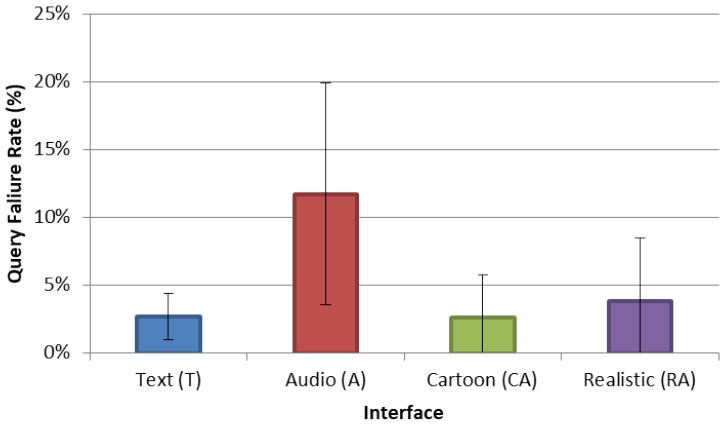


Figure 6.12: **Mean query failure rate (%) by interface. Error bars show ±1 SD.** *Audio (A) has a significantly higher query failure rate than the other interfaces.*

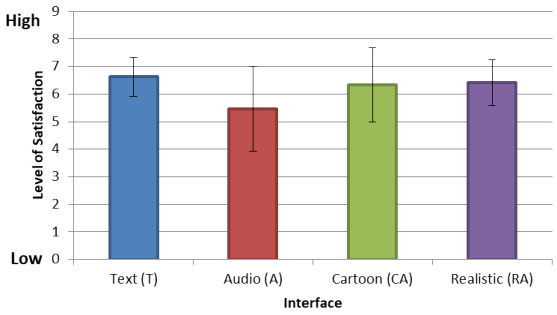
found in Appendix B. The audio interface had a higher query failure rate than the other interfaces.

6.3 Questionnaire Results

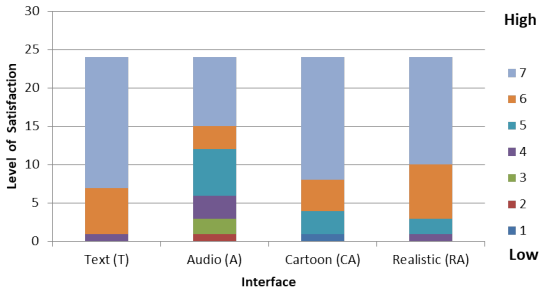
The questionnaire responses were analyzed using the non-parametric Friedman test. An application by MacKenzie called GoStats [130] was used to analyze the feedback data using the required method of analysis. This tool was used instead of more general statistic tools such as SPSS and R because it is customized for human-computer interaction.

6.3.1 Participant satisfaction with the interaction

The means of participant satisfaction level with the interaction by interface (1 = lowest level, 7 = highest level) were Text (T): 6.625, Audio (A): 5.458, Cartoon Avatar (CA): 6.333, and Realistic Avatar (RA): 6.416 are shown in Figure 6.13(a). All interfaces have a high level of participant satisfaction with the interaction. The audio interface had the least participant satisfaction level. There was a significant difference in the level of participant satisfaction with the interfaces ($\chi^2 = 11.826$, $p < .01$, $df = 3$). Figure 6.13(b) illustrates



(a) *Mean participant satisfaction with the interactions. Error bars show ±1 SD. The Audio (A) interface is significantly different from the other interfaces.*



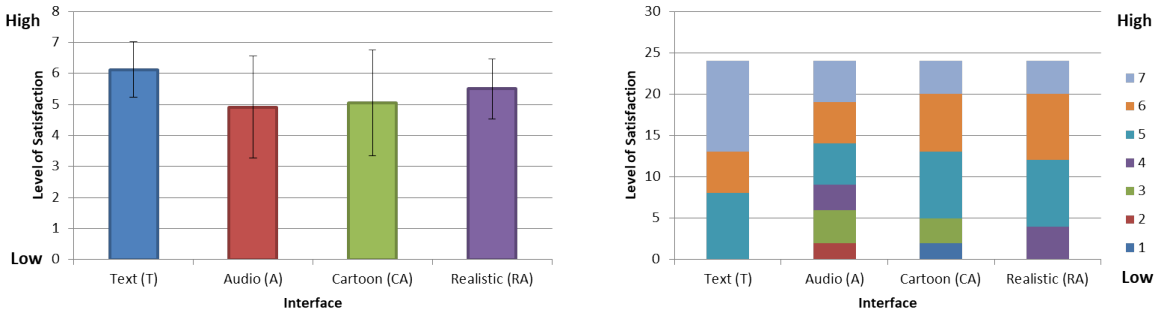
(b) *Participant satisfaction with the interactions. Bars show number of participants selecting each level of satisfaction in different colors.*

Figure 6.13: **Participant satisfaction with the interactions.**

levels of participant satisfaction for every interface. The figure plots the number of participants that selected each level of satisfaction per interface. A Conver's F post hoc test revealed that all pairwise comparisons with the audio interface were statistically significant. Post hoc test values can be found in Appendix B.

6.3.2 Participant satisfaction with the time to obtain a response from the interface

The means of participant satisfaction level with the time to obtain responses by interface (1 = lowest level, 7 = highest level) were Text (T): 6.125, Audio (A): 4.916, Cartoon Avatar (CA): 5.041, and Realistic Avatar (RA): 5.5 are shown in Figure 6.14(a). All interfaces have a high level of participant satisfaction. The text interface has the highest level of participant satisfaction with the time to obtain a response. There was a significant difference in the level of participant satisfaction with the amount of time to obtain responses by the interfaces ($\chi^2 = 10.243$, $p < .05$, $df = 3$). Figure 6.14(b) illustrates levels of participant satisfaction for every interface. The figure plots the number of participants that selected each level of satisfaction per interface. A Conver's F post hoc test revealed that only pairwise comparisons with the text interface were statistically significant. Post hoc test values can be found in Appendix B.



(a) *Mean participant satisfaction with time to get responses by interface. Error bars show ±1 SD. The Text (T) interface is significantly different from the other interfaces.*

(b) *Participant satisfaction with time to obtain a responses. Bars show number of participants selecting each level of satisfaction in different colors.*

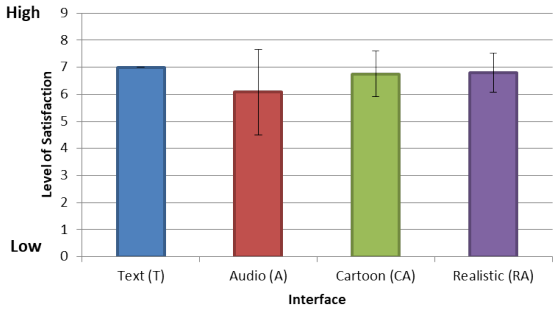
Figure 6.14: **Participant satisfaction with time to get responses.**

6.3.3 Participant perception on accuracy of the responses

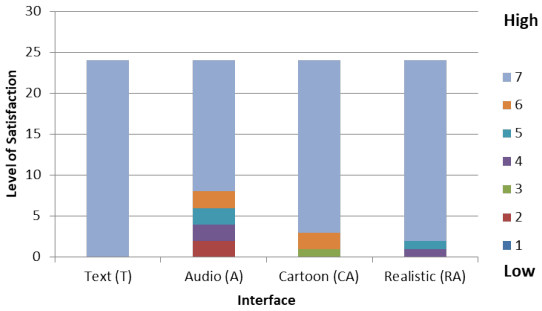
The means for participant perception level on accuracy of the responses by interface (1 = lowest level, 7 = highest level) were Text (T): 7, Audio (A): 6.083, Cartoon Avatar (CA): 6.75, and Realistic Avatar (RA): 6.79 are shown in Figure 6.15(a). All interfaces have a relatively high level of participant perception on accuracy of the responses. The audio interface has the lowest level of participant perception of accuracy of the responses. There was a significant difference in the participant perception on the accuracy of the responses given by the interfaces ($\chi^2 = 14.143$, $p < .01$, $df = 3$). Figure 6.15(b) illustrates levels of participant perception for every interface. The figure plots the number of participants that selected each level of perception per interface. A Conver's F post hoc test revealed that only pairwise comparisons with the audio interface were statistically significant. Post hoc test values can be found in Appendix B.

6.3.4 Participant perception of how fun each interface is to use

The means for participant perception level of how fun each interface is to use by interface (1 = lowest level, 7 = highest level) were Text (T): 4.875, Audio (A): 5.041, Cartoon Avatar (CA): 5.708, and Realistic Avatar (RA): 5.958 are shown in Figure 6.16(a). All interfaces have a relatively high level of participant perception

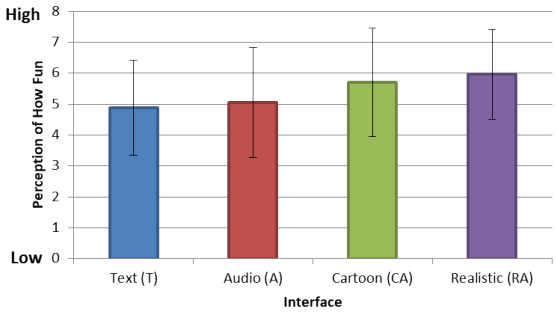


(a) *Mean participant perception on accuracy of the responses. Error bars show ±1 SD. The variance in the text interface data is zero. All participants selected level 7 for the text interface. The Audio (A) interface is significantly different from the other interfaces.*

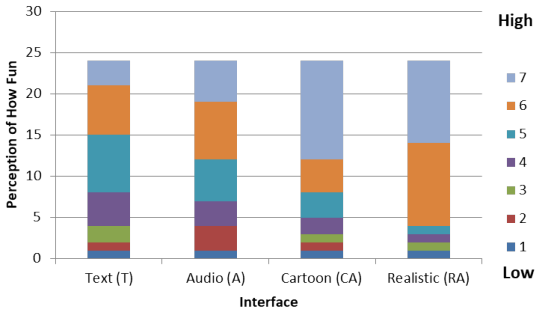


(b) *Participant perception on accuracy of the responses. Bars show number of participants selecting each level of satisfaction with accuracy in different colors.*

Figure 6.15: **Participant satisfaction with accuracy of responses.**



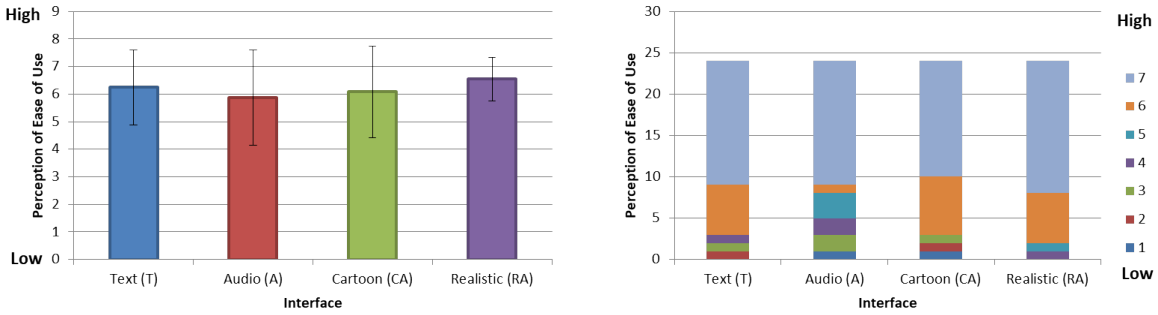
(a) *Mean participant perception of how fun each interface is to use. Error bars show ±1 SD. All pairwise comparisons with the avatar interfaces are significant.*



(b) *Participant perception of how fun each interface is to use. Bars show number of participants selecting each level (of how fun they found the interface) shown in different colors.*

Figure 6.16: **Participant perception of how fun each interface is to use.**

of how fun each interface is to use. There was a difference in the participant perception of how fun each interface is to use ($\chi^2 = 16.746, p < .001, df = 3$). Figure 6.16(b) illustrates levels of participant perception for every interface. The figure plots the number of participants that selected each level of perception per interface. A Conver's F post hoc test revealed that all pairwise comparisons with avatars were statistically significant. Post hoc test values can be found in Appendix B.



(a) *Mean participant perception of the ease of use of each interface. Error bars show ±1 SD. There was no significant difference between participant perception of the ease of use of each interface.*

(b) *Participant perception of ease of use of each interface. Bars show number of participants selecting each level (of perception of ease) of use shown in different colors.*

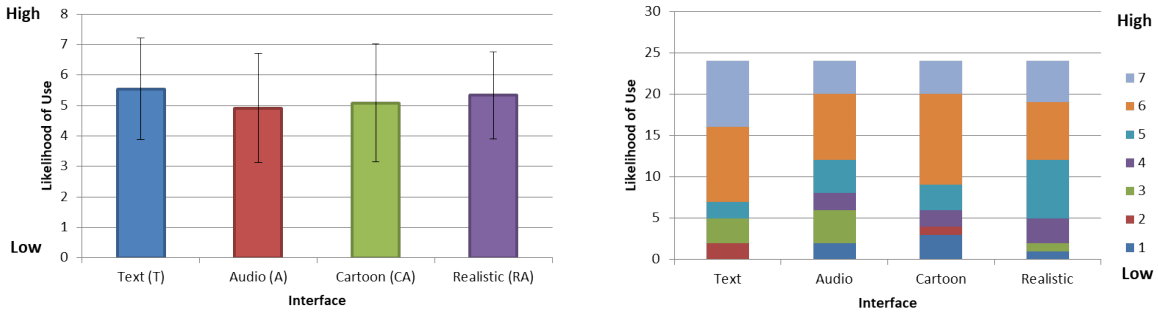
Figure 6.17: **Participant perception of the ease of use of each interface.**

6.3.5 Participant perception of the ease of use of each interface

The means for participant perception level of the ease of use of the interface by interface (1 = lowest level, 7 = highest level) were Text (T): 6.25, Audio (A): 5.875, Cartoon Avatar (CA): 6.083, and Realistic Avatar (RA): 6.541 are shown in Figure 6.17(a). All interfaces have a relatively high level of participant perception of the ease of use of the interface. There was no statistical difference in the participant perception of the ease of use of the interface ($\chi^2 = 1.650$, $p > .05$, $df = 3$). Figure 6.17(b) illustrates levels of participant perception for every interface.

6.3.6 Participant likelihood to use the interface in the future

The means for participant likelihood level to use each interface in the future (1 = lowest level, 7 = highest level) were Text (T): 5.541, Audio (A): 4.916, Cartoon Avatar (CA): 5.083, and Realistic Avatar (RA): 5.333 as shown in Figure 6.18(a). All interfaces have a relatively high level of participant likelihood to use the interface in the future. There was no statistical difference in the participant likelihood to use one interface over another ($\chi^2 = 1.125$, $p > .05$, $df = 3$). Figure 6.18(b) illustrates levels of participant likelihood to



(a) *Mean participant likelihood to use each interfaces. Error bars show ±1 SD. there was no significant difference between the interfaces.*

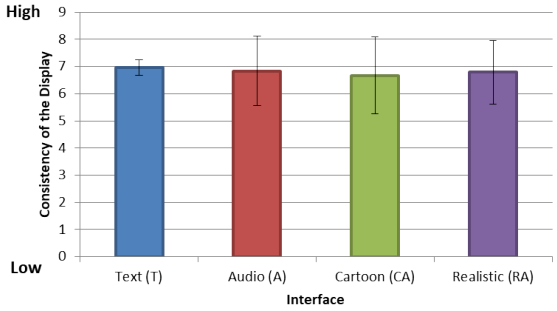
(b) *Participant likelihood to use each interfaces. Bars show number of participants selecting each level (of likelihood to use each interface) shown in different colors.*

Figure 6.18: **Participant likelihood to use each interfaces in the future.**

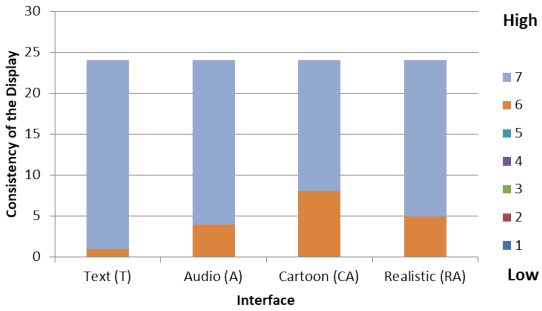
use the interfaces. The figure plots the number of participants that selected each level of likelihood of use per interface.

6.3.7 Participant perception of the consistency of the interface

The means for participant perception level of the consistency of the interface by interface (1 = lowest level, 7 = highest level) were Text (T): 6.958, Audio (A): 6.833, Cartoon Avatar (CA): 6.666, and Realistic Avatar (RA): 6.791 as shown in Figure 6.19(a). All interfaces have a high level of participant perception in terms of the consistency of the interface. There is a statistical difference in the participant's perception of the consistency of the display of the interface ($\chi^2 = 8.825$, $p < .05$, $df = 3$). Figure 6.19(b) illustrates levels of participant perception for every interface. A Conver's F post hoc test revealed that the difference between the text interface and the cartoon avatar interface was statistically significant. Post hoc test values can be found in Appendix B. Figure 6.19(b) plots the number of participants that selected each level of perception per interface. participants selected level 6 and 7 only indicating that participants found the interfaces to be highly consistent in displaying the answers.



(a) *Mean participant perception of the consistency of the display. Error bars show ±1 SD. There was no significant difference between the interfaces.*

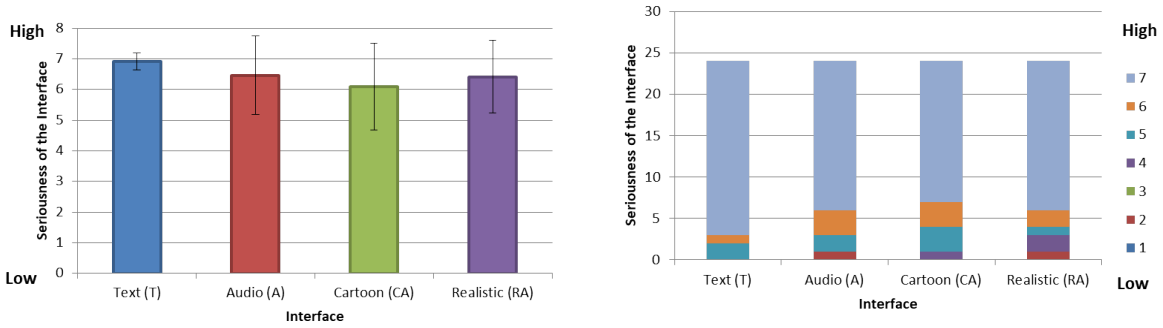


(b) *Participant perception of the consistency of the display. Bars show number of participants selecting each level (of perception of consistency of the display) shown in different colors.*

Figure 6.19: **Participant perception of the consistency of the display.**

6.3.8 Participant perception of the seriousness of the interface

Users were asked if the questions were being responded to in a serious manner and that they found the interfaces to be serious. The means for the perceived level of the seriousness of the interface (1 = lowest level, 7 = highest level) were Text (T): 6.916, Audio (A): 6.458, Cartoon Avatar (CA): 6.083, and Realistic Avatar (RA): 6.416 as shown in Figure 6.20(a). All interfaces have a high level of participant perception in terms of the seriousness of the interface. There is a statistical difference in the participant perception of the seriousness of the interface ($\chi^2 = 16.746$, $p < .001$, $df = 3$). Figure 6.20(b) illustrates levels of participant perception of the seriousness of every interface. The figure plots the number of participants that selected each level of perception per interface. Most participants selected level 6 and 7 indicating that participants found the interfaces to be highly serious. A Conver's F post hoc test revealed that the text interface pairwise comparisons with the cartoon avatar interface and the realistic avatar interface were statistically significant. Post hoc test values can be found in Appendix B.



(a) *Mean participant perception of the seriousness of the interface. Error bars show ±1 SD.*

(b) *Participant perception of the seriousness of the interface. Bars show number of participants selecting each level (of perception of seriousness of the interface) shown in different colors.*

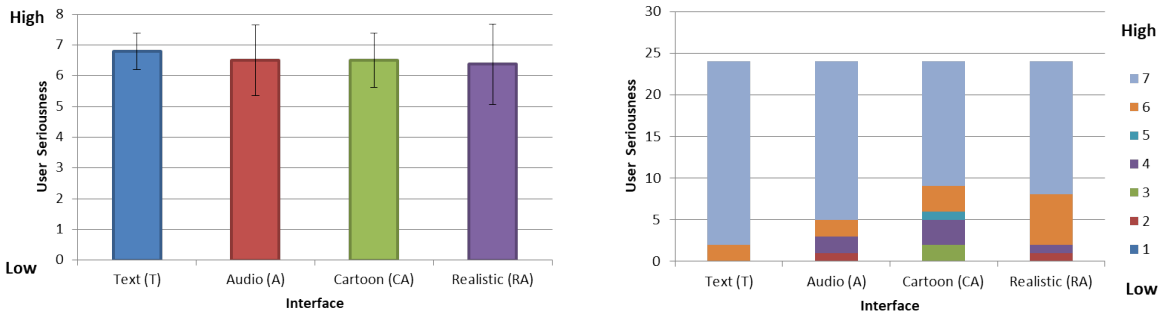
Figure 6.20: **Participant perception on the seriousness of the interface.**

6.3.9 How seriously the participants took the interface

Users were asked if they took the interfaces seriously and that they were asking the questions in a serious manner. The means of the level of how seriously the participants took the interface (1 = lowest level, 7 = highest level) were Text (T): 6.791, Audio (A): 6.5, Cartoon Avatar (CA): 6.5, and Realistic Avatar (RA): 6.375 as shown in Figure 6.21(a). All interfaces have a high level of how seriously the participants took the interface. There was no statistical difference on how seriously the participants took the interface ($\chi^2 = 3.857$, $p > .05$, $df = 3$). Figure 6.21(b) illustrates levels of how seriously the participants took the interface. The figure plots the number of participants that selected each level of perception per interface.

6.3.10 Participant preferences between the text-based and audio-based interfaces

Figure 6.22 illustrates the number of participants that selected the different levels of participant preference for these two interfaces. The figure shows that 11 out of 24 participants were highly confident with their preference for the audio-based interface over the text-based interface. In total there were 8 participants that preferred text and 14 participants that preferred audio. The remaining two participants did not have



(a) *Mean value for how seriously the participants took the interfacee. Error bars show ±1 SD.*

(b) *How seriously the participants took the interface. Bars show number of participants selecting each level (of how seriously they took the interface) shown in different colors.*

Figure 6.21: **How seriously the participants took the interface.**

a preference. Figure 6.22 also illustrates the number of female and male participants that selected the different levels of participant preference for these two interfaces. The figures shows that 6 out of the 11 participants that are highly confident with there preference of the audio-based interfaces over the text-based interfaces were female and the remaining 5 participants were male. 3 out of the 4 participants that are highly confident with there preference of text-based interfaces over audio-based interface were also female and the remaining participant was male. In total there were 6 out of 12 female participants that preferred the audio-based interfaces and 6 that preferred the text-based interfaces. For male participants, in total there were 8 out of 12 male participants that preferred the audio-based interfaces and 2 participants that preferred the audio-based interfaces. The remaining two participants did not have a preference.

6.3.11 Participant preferences between avatar-based and audio-based interfaces

Figure 6.23 illustrates number the of participants that selected the different levels of participant preference for these two interfaces. The figure shows that 9 out of 24 participants were highly confident with there preference of the avatar-based interfaces over the audio-based interface. In total there were about 10 participants that preferred the audio-based interface and 14 participants that preferred the avatar-based interfaces. Figure 6.23 also illustrates the number of female and male participants that selected the different levels of participant

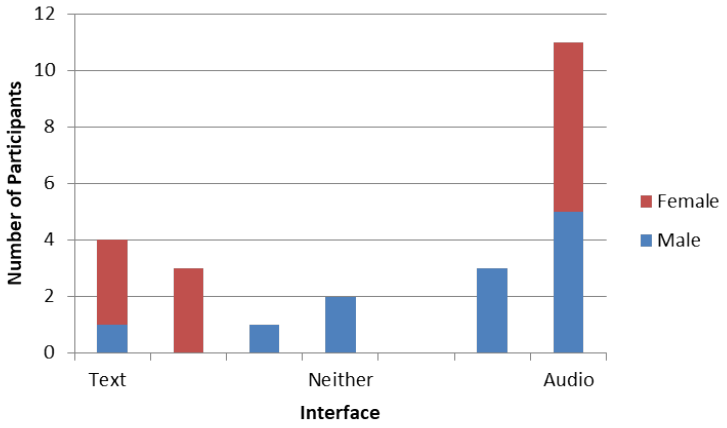


Figure 6.22: **Participant preferences between the text-based and audio-based interfaces.**

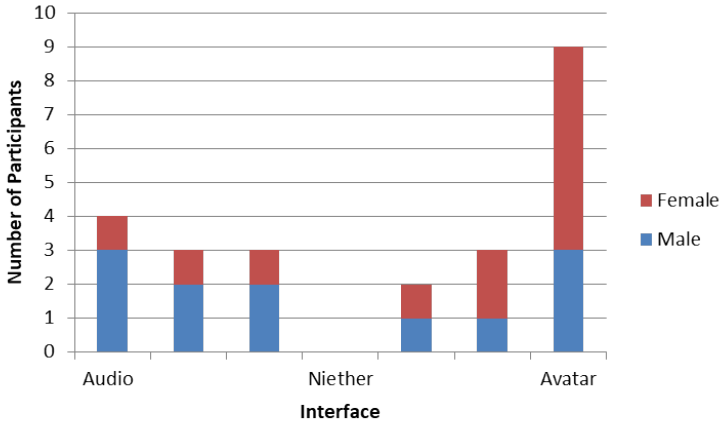


Figure 6.23: **Participant preferences between avatar-based and audio-based interfaces.**

preference for these two interfaces. From the 9 participants that were highly confident with there preference of the avatar-based interfaces over the audio-based interfaces 6 participants were female. From the 4 participants that were highly confident with there preference of audio-based interfaces over avatar-based interface 3 participants were female. In total there were 9 out of 12 female participants that preferred the avatar-based interfaces and 3 female participants that preferred the audio-based interfaces. For male participants, in total there were 5 out of 12 male participants that preferred the avatar-based interfaces and 7 male participants that preferred the audio-based interfaces.

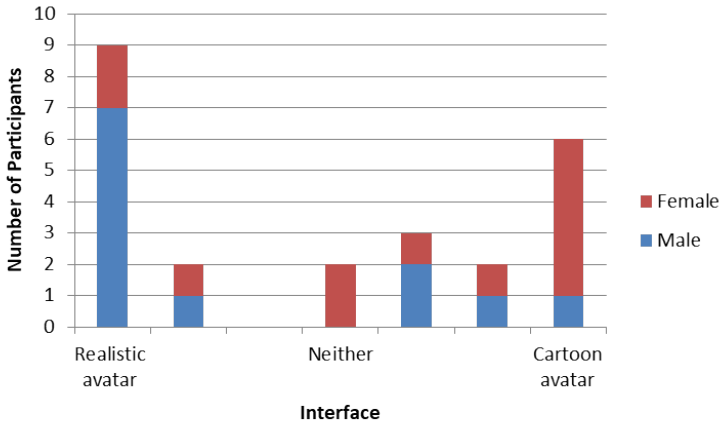


Figure 6.24: **Participant preferences between realistic avatar-based and cartoon avatar-based interfaces.**

6.3.12 Participant preferences between realistic avatar-based and cartoon avatar-based interfaces

Figure 6.24 illustrates number the of participants that selected the different levels of participant preference for these two interfaces. The figure shows that 9 out of 24 participants were highly confident with there preference of realistic avatar interface over cartoon avatar interface. In total there were 11 participants that preferred the cartoon avatar and 11 that preferred the realistic avatar. The remaining two participants did not have a preference. Figure 6.24 also illustrates the number of female and male participants that selected the different levels of participant preference for these two interfaces. From the 9 participants that were highly confident with there preference of realistic avatar interface over cartoon avatar interface 7 participants were male. From the 6 participants that were highly confident with there preference of cartoon avatar interface over realistic avatar interface 5 participants were female. In total there were 7 out of 12 female participants that preferred the cartoon avatar interface and 3 female participants that preferred the realistic avatar interface. The remaining two participants did not have a preference. For male participants, in total there were 4 out of 12 male participants that preferred the cartoon avatar interface and 8 male participants that preferred the realistic avatar interface.

6.4 Discussion

This work includes an empirical evaluation comparing participant responses through text (T), Audio (A), Realistic Avatar (RA) and Cartoon Avatar (CA) interfaces to an audio-only input. The study compared ratio scale data obtained during the participant's interaction The study requested participant's feedback on the interfaces' accuracy, speed, ease of use, likelihood of use, how fun the interface is to use, consistency of the interaction and their perception of how serious the interface was. Participants in general expressed a high level of satisfaction with the responses and the speed and accuracy of the responses for all interfaces. Participants in general found all interfaces to be consistent, easy and fun to use. Most participants expressed that using speech to ask questions was easy and felt natural. Participants showed that they are likely to use all the presented interfaces. This suggests that all interfaces could be used to develop knowledge-based applications. The participants expressed that in general they found the interfaces to be serious and that they took the interfaces seriously. The study also requested the participants feedback on the general preference among types of interfaces. In general more participants preferred the audio interface over the text interface, the avatar interfaces over the audio interface and the realistic avatar interface over the cartoon avatar interface.

6.4.1 Input time

The means of input time by interface ranged from 2.12 to 2.83 seconds. There was a significance difference in the audio interface input time compared to other interfaces. Input time in the audio interface was significantly lower than the other interfaces, indicating that participants spoke faster when asking the questions using this interface. One possible explanation for this result is that participants in the audio interface have no visual distractions and they were more focused on the questions to ask, resulting in faster input. The audio interface also had a higher query failure rate possibility as a result of the use of this strategy. A study [139]

confirms our findings by suggesting that faster speech results in more errors in speech recognition. The study also suggests that inserting even a 1ms pause between words improved the ability to identify errors, while a fast speech rate made the task more difficult.

6.4.2 Response generation time

The mean for response generation time by interface ranged from 3.25 to 6.07 seconds. The text interface had the lowest response generation time and the realistic avatar interface had the highest response generation time. There was a significant difference between the response generation time of the text interface and the other interfaces. This is to be expected as the text interface displays the result as text while the audio interface requires additional text-to-speech processing, the cartoon avatar interface requires text-to-speech processing and syncing the response to the animation, and the realistic avatar interface requires text-to-speech processing, video rendering and syncing the response to the animation.

6.4.3 Participant attentiveness specific results

The means for participant attentiveness ranged from 32% to 55% of the total interaction time. There was significant difference in participant attentiveness by interface. This result can be taken into consideration when developing knowledge applications. Different applications may require different levels of participant attentiveness levels.

There was an attentiveness decline in all interfaces, participants seemed to lose attention as rounds of questions went by. This suggests that applications may want to place important information at the beginning of the interaction. More interface specific results related to this ratio scale data is discussed below.

6.4.4 Query failure rate

The Query failure rate by interface ranged from 3% to 12%. There was a significance difference in the audio interface query failure rate compared to other interfaces. This may be a result of the lack of attentiveness of the participants to the display when using Audio(A) or the speed in which participants interact with the audio-only interface.

6.4.5 General discussion concerning the text (T) interface

The text interface had the lowest response generation time. The results show that the difference in mean response generation time between the text-based interface and the avatar-based interfaces was significant. Participants in general were more attentive to the text interface as they had to read the answer of the questions off the screen. There may be applications that require the participant to be fully aware of the given information for accuracy reasons. This result suggests that the text interface is the one in which participants are most attentive, thus the text interface is best for such applications. This is confirmed by [140, 141] that found that adding text, graphics, or visual cues increases the attention of users interacting with education tools. The text interface had the highest level of participant perception on the accuracy of responses, however it was not significantly higher that the other avatar interfaces. Participants found the text interface to be more consistent than the cartoon avatar in displaying the responses. Participants also found the text interface to be more serious than the avatar interfaces.

6.4.6 General discussion concerning the audio (A) interface

Even though participants showed a high satisfaction level with the responses and the accuracy of the audio interface, this satisfaction level was significantly lower than the other interfaces. This is possibly due to the

audio interface having a higher query failure rate. This may indicate that participants may not dissociate accuracy of the response with an error that promotes a retry. Several studies confirm [142, 143, 144] our finding that errors in interactions lower the general satisfaction with robots communicating with speech. This may suggest that engineering of AI applications that require a high level of perception of accuracy should reduce reporting of errors to only when necessary. Participants were the least attentive to the audio interface, which suggests that the audio is in-appropriate for application where the participant should be attentive. For example, consider an application that answers questions of a worker that needs to focus on other issues, such as safety regulation. A number of studies summarized in [145] suggest that audio-based communication should be used in applications that are "hand or eye busy". Even though participants had a lower satisfaction with the responses and accuracy of the responses given by the audio interface and the higher error rate of the interface, more participants selected the audio interface when asked to choose in general between the text interface and the audio interface.

6.4.7 General discussion concerning the cartoon avatar (CA) interface

Even though the response time for the cartoon avatar interface was significantly higher that the text interface, participants still expressed a high satisfaction level with the time to get responses. Participants were more attentive to the cartoon avatar interface than the audio interface. Participants had an intermediate level of attentiveness with the cartoon avatar interface which may fit the attentiveness requirement level of many applications. For example, participants can be as attentive as they please using an information desk in a mall. Cartoon avatar interfaces may be appropriate for such venues. Participants found the cartoon avatar interface to be significantly more fun to use than the other non-avatar interfaces. Using this result, one may conclude that applications that rely on a "fun factor" may want to use a cartoon avatar as a form of an informative mechanism.

6.4.8 General discussion concerning the realistic avatar (RA) interface

Even through response time for the realistic avatar was significantly higher than the other interfaces, participants still expressed a high satisfaction level with the time needed to obtain responses. There was no significant difference in participant satisfaction with the time to get responses from the realistic avatar interface and the audio and cartoon avatar interfaces. This is possibly due to the filler animations that give the illusion of a response starting before it actually does. participants were more attentive to the realistic avatar interface than the audio and the cartoon avatar interfaces. When developing application that require attention from the participant, but not necessarily fully attentive, this result suggests that a realistic avatar interface would provide more attention from the participants compared to audio or other avatar interfaces. Participants also found the realistic avatar interface to be significantly more fun to use than the other interfaces. Similar to the cartoon avatar, one may conclude that applications that rely on a "fun factor" may want to use the realistic avatar as a form of an informative mechanism. More participants selected the avatar interfaces when asked to generally select a preference among interfaces. Between avatars more participants selected the realistic avatar.

6.5 Summary

An empirical evaluation was conducted to compare interaction through text (T), Audio (A), Realistic Avatar (RA) and Cartoon Avatar (CA). Twenty four English speaking participants used these interfaces. The study compared ratio scale data obtained during the participants interacting with the system. The ratio scale data collected was input time, response time, query failure rate rate and participant attentiveness. The study requested participant's feedback on the interfaces accuracy, speed, ease of use, likelihood of use, how fun the interface is, consistency of display and seriousness. Input time was significantly lower for the audio interface indicating faster speech which possibly explains the significantly higher query failure rate for the

audio interface. The high query failure rate may explain the significantly lower satisfaction level shown by participants for the audio interface. The response time was significantly higher for the both avatar interfaces. This however, had no significant effect on the participant satisfaction with the responses given by these interfaces possibly due to the filler animations. Participants showed higher attentiveness towards the text interface followed by the avatar interfaces and participants were least attentive to the audio interface. There was a decline of attentiveness as participants asked more questions. In general participants expressed high level of satisfaction with the accuracy, speed, ease of use consistency of display and seriousness of the interfaces. They also expressed that all interfaces were fun and that they would likely use them. Participants expressed a higher perception of accuracy and speed for the text interface. Participants found the avatar interfaces to be the most fun among the interfaces. When asked about there general preference, more participants preferred the audio interface over the text interface, the avatar interfaces over the audio interface and the realistic avatar interface over the cartoon avatar interface.

Chapter 7

Summary and future work

7.1 Summary

There exist many different technologies that enhance the effectiveness and quality of computer-based services. These technologies include virtual agents and robotic systems; providing information and assistance. These digital implementations require human computer interaction. In addition to providing functionality, the interface for these human-computer interactions must be interactive, user friendly and inviting, so that it can encourage user involvement and acceptance of the application. The challenge is to design interfaces that fit the targeted users and that enables smooth interactions between the users and the application. One way of providing a natural and effective user interface is through the simulation of some active agent or avatar and having that simulation interact with the user. Using speech as an interactive method of communication in the human computer interaction application would provide a natural way of communication.

To use speech in the human-robot interaction application, a robot would be required to understand

and process speech in a meaningful way. Establishing this form of communication between humans and robots using the standard robot operating system ROS allows for the creation of a robust and portable toolkit. People interact through a microphone. Thus the basic implementation is to capture speech through a microphone and establish an audio object that can be sent through the ROS messaging system for further processing and manipulation. An additional features allows for the use of an audio file instead of speech in order to communicate commands to a robot. There are many speech recognition engines, Thus there is a need for a toolkit that provides a simple way of alternating between these engines for quality or functionality purposes. The main purpose of recognizing the speech is to generate an appropriate response or motion sequence. The toolkit utilizes Wolfram Alpha Short Answers API to provide such a response.

The next step is to communicate the response back to the human user. This work provides the response through an interactive face on a robot. In order to accomplish this a standard text-to-speech generation system is combined with a 3D avatar (puppet) whose facial animation is tied to the utterance being generated. In order to embed emotional state and other out of band information, messages presented to the text-to-speech module are embedded within an XML structure known as the Avatar Utterance Markup language (AUML) that allows the user to tune the nature of the puppet animation so that different emotional states of the puppet can be simulated. The expressions and spoken words are plotted and animated in the sequential order as they appear in the AUML. Between utterances the avatar is not still. Rather the avatar appears in apparently normal motion when not engaged with a user. Furthermore, the avatar transit from this delay behavior to utterance behavior seamlessly. We accomplish this by pre-rendering and pre-loading to the local display a collection of renderings that can be played when the avatar is idle and which are designed to be combined together to make arbitrarily long sequences of idle behavior. The Avatar Delay Graph (ADG) provides a formal structure within which to encode short locally cached video sequences that can be played so as to provid an animation of the avatar between utterances. This structure also provides a mechanism within which to obscure rendering and transmission latencies which are unavoidable given the cloud-based rendering of the avatar.

When a new utterance is required, rendering the animation takes place at the time of request. Cloud-based utterance recognition, responding to the utterance, and rendering and transmitting the resulting video sequence introduces latency. There are two things that can be done to reduce this latency. The first is to optimize the rendering engine so that unneeded effort in rendering can be avoided. The second is to parallalize the rendering in the cloud. To do that, we generate a cloud-based rendering farm where each instance is used as a pool of instances performing the requests of a multiprocessing ROS node located on the local machine. by default, instances created on a cloud platform do not have a GUI, graphics display device and audio playback device. This work requires rendering engines. Rendering animations with blender requires an X server, display screen and audio sink. In order to create a rendering engine from a compute engine a dummy audio sink needs to be created and activated and an x server needs to be started using a virtual display screen that supports graphical rendering. Each instance is used to render different segments of the animation in parallel, reducing the time taken to render the avatar. These segments are then played in the correct sequence. When applying multiprocessing to rendering of animations we divide the required rendering segment into components that can be rendered separately and begin displaying the rendered utterance as soon as we possibly can. this works displays the first segment as soon as possible and allows each subsequent segment to continue to render during rendering and display of all segments previous to it. This allows for faster display and more a more efficient use of cloud resources.

An empirical evaluation was conducted to compare interaction through text (T), Audio (A), Realistic Avatar (RA) and Cartoon Avatar (CA). 24 English speaking participants used these interfaces. The study compared ratio scale data obtained during the users interacting with the system. The ratio scale data collected was input time, response time, query failure rate rate and user attentiveness. The study requested user's feedback on the interfaces accuracy, speed, ease of use, likelihood of use, how fun the interface is, consistency of display and seriousness. Input time was significantly lower for the audio interface indicating faster speech which possibly explains the significantly higher query failure rate for the audio interface. The high query failure rate may explain the significantly lower satisfaction level shown by users for the

audio interface. The response time was significantly higher for both avatar interfaces. This however, had no significant effect on the user satisfaction with the responses given by these interfaces possibly due to the filler animations. Users showed higher attentiveness towards the text interface followed by the avatar interfaces and users were least attentive to the audio interface. There was a decline of attentiveness as users asked more questions. In general users expressed high level of satisfaction with the accuracy, speed, ease of use consistency of display and seriousness of the interfaces. They also expressed that all interfaces were fun and that they would likely use them. Users expressed a higher perception of accuracy and speed for the text interface. Users found the avatar interfaces to be the most fun among the interfaces. When asked about there general preference, more users preferred the audio interface over the text interface, the avatar interfaces over the audio interface and the realistic avatar interface over the cartoon avatar interface.

7.2 Future Work

For a more realistic human-robot engagement, the avatar would preferably be able to simulate natural human gaze towards a speaker in real-time. To integrate that into the avatar system a sound localizing procedure can be used to identify the location of the speaker allowing for an estimation of coordinates that the avatar can direct its gaze to. [146] describes the development of a trustworthy sound localization system that includes a VAD (Voice Activity Detection) component using three microphones and a face tracking system that uses a front-faced camera.

Bibliography

[1] (2018) Max headroom (character). [Online]. Available: https://en.wikipedia.org/wiki/Max_Headroom_(character)

[2] E. STRANG. (2017) Soul machines unveils its first emotionally intelligent, lifelike avatar. [Online]. Available: https://idealog.co.nz/tech/2017/02/soul-machines-unveils-its-first-emotionally-intelligent-lifelike-avatar

[3] N. A. Shaked, "Avatars and virtual agents relationship interfaces for the elderly," *Healthcare Technology Letters 4.3*, pp. 83–87, 2017.

[4] (2018) no. 1 avatar based social experience. [Online]. Available: https://secure.imvu.com/welcome/ftux/

[5] (2018) Clippit. [Online]. Available: http://mugen.wikia.com/wiki/Clippit

[6] (2018) Sophia. [Online]. Available: http://www.hansonrobotics.com/robot/sophia/

[7] (2010) Nexi - robot with facial expressions. [Online]. Available: http://allfamousnews.blogspot.ca/2010/11/nexi-robot-with-facial-expressions.html

[8] (2010) Affetto child robot with realistic facial expressions. [Online]. Available: http://www.technovelgy.com/ct/Science-Fiction-News.asp?NewsNum=3203

[9] K. Schwab. (2018) how IBM and Airbus designed a floating robot head for the ISS. . [Online]. Available: Available:https://amp.fastcompany.com/90180565/how-ibm-designed-a-floating-robot-head-to-help-out-around-the-iss

[10] J. K. Lee, R. L. Toscano, W. D. Stiehl, and C. Breazeal, "The design of a semi-autonomous robot avatar for family communication and education," in *RO-MAN 2008 - The 17th IEEE International Symposium on Robot and Human Interactive Communication*, Munich, Germany, Aug 2008, pp. 166–173.

[11] (2017) Ros.org — powering the world's robots. [Online]. Available: http://www.ros.org/

[12] (2017) Speechrecognition 3.8.1 : Python package index - pypis. Accessed 30-April-2017. [Online]. Available: https://pypi.python.org/pypi/SpeechRecognition/

[13] (2017) WolframAlpha APIs: Computational knowledge integration - products. Accessed 30-April-2017. [Online]. Available: https://products.wolframalpha.com/api/

[14] (2018) Google cloud computing, hosting services and APIs — Google cloud. [Online]. Available: https://cloud.google.com/

[15] A. Sandygulova, M. Dragone, and G. O'Hare, "Privet- a portable ubiquitous robotics testbed for adaptive human-robot interaction," *Journal of Ambient Intelligence and Smart Environments*, vol. 8, pp. 5–19, 01 2016.

[16] P. Bremner, O. Celiktutan, and H. Gunes, "Personality perception of robot avatar tele-operators," in *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Christchurch, New Zealand, March 2016, pp. 141–148.

[17] M. Niemelä, A. Arvola, and I. Aaltonen, "Monitoring the acceptance of a social service robot in a shopping mall: First results," in *Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. New York, NY: ACM, 2017, pp. 225–226. [Online]. Available: http://doi.acm.org/10.1145/3029798.3038333

[18] A. K. Pandey, L. de Silva, and R. Alami, "A novel concept of human-robot competition for evaluating a robot's reasoning capabilities in HRI," in *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Christchurch, New Zealand, March 2016, pp. 491–492.

[19] (2017) Meet milo, an intelligent robot that is really good at teaching children with autism social skills. Accessed 15-October-2017. [Online]. Available: http://www.acapela-group.com/meet-milo-an-intelligent-robot-that-is-really-good-at-teaching-children-with-autism-social-skills

[20] L. S. Lopes and A. Teixeira, "Human-robot interaction through spoken language dialogue," in *2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)*, vol. 1, Takamatsu, Japan, 2000, pp. 528–534 vol.1.

[21] Z. Henkel, V. Srinivasan, R. R. Murphy, V. Groom, and C. Nass, "A toolkit for exploring the role of voice in human-robot interaction," in *2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Lausanne, Switzerland, March 2011, pp. 255–256.

[22] (2017) A voice user interface for human-robot interaction on a service robot. Accessed 5-November-2017. [Online]. Available: http://www.ra.cs.uni-tuebingen.de/diplomarbeiten/BA_Mihael_Simonic.pdf

[23] H. Medicherla and A. Sekmen, "Humanrobot interaction via voice-controllable intelligent user interface," *Robotica*, vol. 25, pp. 521–527, 09 2007.

[24] J. Shi, H. Ma, J. Zhao, and Y. Liu, "Web-based human robot interaction via live video streaming and voice," in *Intelligent Robotics and Applications*, Y. Huang, H. Wu, H. Liu, and Z. Yin, Eds. Cham, Switzerland: Springer International Publishing, 2017, pp. 393–404.

[25] C. R. Crowelly, M. Villanoy, M. Scheutzz, and P. Schermerhornz, "Gendered voice and robot entities: Perceptions and reactions of male and female subjects," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, Oct 2009, pp. 3735–3741.

[26] M. V. den Bergh, D. Carton, R. D. Nijs, N. Mitsou, C. Landsiedel, K. Kuehnlenz, D. Wollherr, L. V. Gool, and M. Buss, "Real-time 3D hand gesture interaction with a robot for understanding directions from humans," in *2011 RO-MAN*, Atlanta, GA, July 2011, pp. 357–362.

[27] P. K. Pisharady and M. Saerbeck, "A robust gesture detection and recognition algorithm for domestic robot interactions," in *2014 13th International Conference on Control Automation Robotics Vision (ICARCV)*, Singapore, Singapore, Dec 2014, pp. 775–780.

[28] M. Y. Cho and Y. S. Jeong, "Human gesture recognition performance evaluation for service robots," in *2017 19th International Conference on Advanced Communication Technology (ICACT)*, Bongpyeong, South Korea, Feb 2017, pp. 847–851.

[29] V. Garg, A. Mukherjee, and B. Rajaram, "Classifying human-robot interaction using handshake data," in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Banff, Canada, Oct 2017, pp. 3153–3158.

[30] G. Randelli, M. Venanzi, and D. Nardi, "Tangible interfaces for robot teleoperation," in *2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Lausanne, Switzerland, March 2011, pp. 231–232.

[31] K. Yamashita, Y. Kato, K. Kurabe, M. Koike, K. Jinno, K. Kito, K. Tatsuno, and M. T. Sqalli, "Remote operation of a robot for maintaining electric power distribution system using a joystick and a master arm as a human robot interface medium," in *2016 International Symposium on Micro-NanoMechatronics and Human Science (MHS)*, Nagoya, Japan, Nov 2016, pp. 1–7.

[32] S. Kajikawa, K. Takahashi, and A. Mihara, "A joystick interface for tongue operation with adjustable reaction force feedback," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, Sept 2015, pp. 3753–3758.

[33] S. Faroque, B. Horan, and M. Joordens, "Keyboard control method for virtual reality micro-robotic cell injection training," in *2015 10th System of Systems Engineering Conference (SoSE)*, San Antonio, TX, May 2015, pp. 416–421.

[34] P. Rouanet, J. Bechu, and P. Y. Oudeyer, "A comparison of three interfaces using handheld devices to intuitively drive and show objects to a social robot: the impact of underlying metaphors," in *RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication*, Toyama, Japan, Sept 2009, pp. 1066–1072.

[35] S. T. Hayes, E. R. Hooten, and J. A. Adams, "Multi-touch interaction for tasking robots," in *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Osaka, Japan, March 2010, pp. 97–98.

[36] C. C. Kemp, C. D. Anderson, H. Nguyen, A. J. Trevor, and Z. Xu, "A point-and-click interface for the real world: Laser designation of objects for mobile manipulation," in *Proceedings of the 3rd ACM/IEEE International Conference on Human Robot Interaction (HRI)*. New York, NY: ACM, 2008, pp. 241–248.

[37] E. Ackerman. (2017) Point-and-click method makes robot grasping control less tedious. [Online]. Available: spectrum.ieee.org/automaton/robotics/robotics-software/point-and-click-method-robot-grasping-control.

[38] (2018) Gesture recognition device to fast-track with company's invite to techstars acceleratorr. [Online]. Available: https://www.sfu.ca/sfunews/stories/2018/02/gesture-recognition-device-invite-to-techstar-accelerator.html

[39] (2018) Search your android phone with written gestures. [Online]. Available: http://googlemobile.blogspot.ca/2010/03/search-your-android-phone-with-written.html

[40] (2017) Canadian face-detecting drones respond to voice, gesture commands. [Online]. Available: https://www.design-engineering.com/canadian-face-detecting-drones-respond-to-voice-gesture-commands-125448/ nov-13-sfu-drone-voice-360/

[41] M. Margaritoff. (2017) This drone is controlled by facial expressions. [Online]. Available: http://www.thedrive.com/aerial/18294/this-drone-is-controlled-by-facial-expressions

[42] (2018) Siri. [Online]. Available: https://en.wikipedia.org/wiki/Siri

[43] (2018) Alexa. [Online]. Available: https://developer.amazon.com/alexa

[44] (2018) Cortana. [Online]. Available: https://en.wikipedia.org/wiki/Cortana

[45] (2018) Isaac asimov. [Online]. Available: https://en.wikipedia.org/wiki/Isaac_Asimov

[46] (2018) Norman spinrad. [Online]. Available: https://en.wikipedia.org/wiki/Norman_Spinrad

[47] (2018) Tron. [Online]. Available: https://en.wikipedia.org/wiki/Tron

[48] (2018) Neal stephenson. [Online]. Available: https://en.wikipedia.org/wiki/Neal_Stephenson

[49] (2018) The matrix. [Online]. Available: https://en.wikipedia.org/wiki/The_Matrix

[50] (2018) Hatsune miku. [Online]. Available: https://en.wikipedia.org/wiki/Hatsune_Miku

[51] (2018) Avatar (2009 film). [Online]. Available: https://en.wikipedia.org/wiki/Avatar_(2009_film)

[52] E. Cline, *Ready Player One*, ser. Broadway Books. Broadway Paperbacks, 2011. [Online]. Available: https://books.google.ca/books?id=rWuODQAAQBAJ

[53] N. Bailenson, "Immersive virtual environment technology as a methodological tool for social psychology," *Psychological Inquiry*, vol. 13, pp. 103–124, 2002.

[54] T. L. Taylor, *Living Digitally: Embodiment in Virtual Worlds*. London: Springer London, 2002, pp. 40–62.

[55] (2018) Custom avatar. [Online]. Available: http://www.123flashchat.net/custom-avatar.html

[56] (2017) Makehuman — open source tool for making 3D characters.. [Online]. Available: http://www.makehuman.org/

[57] P. Russell, Stuart J.; Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. Saddle River, NJ, USA: Morgan Kaufmann Publishers Inc., 2003.

[58] E. Cosatto, H. P. Graf, and J. Ostermann, "System for low-latency animation of talking heads," Patent US 7 260 539, August, 2007.

[59] A. D. Conkie, M. C. Beutnagel, and T. Mishra, "System and method for low-latency web-based text-to-speech without plugins," Patent US 9 240 180, January, 2016.

[60] R. Merrick, M. Thenhaus, W. Bell, and M. Zartler, "System and method for automatic animation generation," Patent US 6 433 784, August, 2002.

[61] H. Beigi, *Fundamentals of Speaker Recognition*. USA: Springer Publishing Company, Incorporated, 2011.

[62] A. Schmitt, D. Zaykovskiy, and W. Minker, "Speech recognition for mobile devices," *International Journal of Speech Technology*, vol. 11, no. 2, pp. 63–72, Jun 2008.

[63] D. Zaykovskiy, "Survey of the speech recognition techniques for mobile devices," in *11th International Conference on Speech and Computer (SPECOM)*, St. Petersburg (Russia), Jun. 2006.

[64] M. M. Abbasi, A. M. Abbasi, and A. Q. Abbasi, "Speech recognition : A comprehensive study," *International Journal of Scientific and Engineering Research*, vol. 5, no. 1, pp. 192–195, Jan. 2014.

[65] C. Ssnderson and K. K. Paliwal, "Effect of different sampling rates and feature vector sizes on speech recognition performance," in *Proc. IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications., (TENCON)*, vol. 1, Dec 1997, pp. 161–164.

[66] N. A. Meseguer, "Speech Analysis for Automatic Speech Recognition," Master's thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2009.

[67] H. Bourlard, H. Hermansky, and N. Morgan, "Towards increasing speech recognition error rates," *Speech Commun.*, vol. 18, no. 3, pp. 205–231, May 1996.

[68] D. O'Shaughnessy, "Invited paper: Automatic speech recognition: History, methods and challenges," *Pattern Recogn.*, vol. 41, no. 10, pp. 2965–2979, Oct. 2008.

[69] E. Trentin and M. Gori, "Robust combination of neural networks and hidden markov models for speech recognition," *Trans. Neur. Netw.*, vol. 14, no. 6, pp. 1519–1531, Nov. 2003.

[70] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 328–339, Mar 1989.

[71] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, Feb 1989.

[72] S. Young, *HMMs and Related Speech Recognition Technologies.* Berlin, Heidelberg: Springer, 2008, pp. 539–558.

[73] D. Povey, L. Burget, M. Agarwal, P. Akyazi, K. Feng, A. Ghoshal, O. Glembek, N. K. Goel, M. Karafit, A. Rastrow, R. C. Rose, P. Schwarz, and S. Thomas, "Subspace Gaussian mixture models for speech recognition," in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, Dallas, TX, USA, March 2010, pp. 4330–4333.

[74] (2017) Dynamic time warping, wikipedia. [Online]. Available: http://en.wikipedia.org/wiki/Dynamic_time_warping

[75] B. I. Pawate and P. S. D. Robinson, "Implementation of an hmm-based, speaker-independent speech recognition system on the TMS320C2x and TMS320C5x," Texas Instruments Inc., Texas,USA, Tech. Rep., 1998.

[76] N. Morgan, H. Bourlard, S. Renals, M. Cohen, and H. Franco, "Hybrid neural network/hidden markov model systems for continuous speech recognition." *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 7, pp. 899–916, 08 1993.

[77] T. Robinson, "A real-time recurrent error propagation network word recognition system," in *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, San Francisco, CA, USA, Mar 1992, pp. 617–620 vol.1.

[78] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 328–339, Mar 1989.

[79] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, Nov 2012.

[80] L. Deng, G. Hinton, and B. Kingsbury, "New types of deep neural network learning for speech recognition and related applications: an overview," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada, May 2013, pp. 8599–8603.

[81] V. Goel and W. J. Byrne, "Minimum Bayes-risk automatic speech recognition," *Computer Speech and Language*, vol. 14, no. 2, pp. 115 – 135, 2000.

[82] J. M. Baker, L. Deng, J. Glass, S. Khudanpur, C. h. Lee, N. Morgan, and D. O'Shaughnessy, "Developments and directions in speech recognition and understanding, part 1 [dsp education]," *IEEE Signal Processing Magazine*, vol. 26, no. 3, pp. 75–80, May 2009.

[83] P. Rubin, T. Baer, and P. Mermelstein, "An articulatory synthesizer for perceptual research," *Journal of The Acoustical Society of America*, vol. 70, pp. 321–328, 1981.

[84] J. Allen, M. S. Hunnicutt, D. H. Klatt, R. C. Armstrong, and D. B. Pisoni, *From Text to Speech: The MITalk System*. New York: Cambridge University Press, 1987.

[85] J. P. H. van Santen, J. P. Olive, R. W. Sproat, and J. Hirschberg, Eds., *Progress in Speech Synthesis*. Berlin, Heidelberg: Springer-Verlag, 1997.

[86] R. Mannell. (2007) Phonetics and phonology prosody and intonation topics. [Online]. Available: http://clas.mq.edu.au/speech/phonetics/phonology/intonation/index.html

[87] J. Santen, "Assignment of segmental duration in text-to-speech synthesis," *Computer Speech & Language*, vol. 8, pp. 95–128, 04 1994.

[88] S. Pachoud A, S. Gong, and A. Cavallaro, "Audio and video reactive talking head avatar," 07 2008.

[89] (2017) The uncanny valley of human-robot interactions vunela. Accessed 5-November-2017. [Online]. Available: https://magazine.vunela.com/the-uncanny-valley-of-human-robot-interactions-9fe3c8b04718

[90] V. Wan, R. Anderson, A. Blokland, N. Braunschweiler, L. Chen, B. Kolluru, J. Latorre, R. Maia, B. Stenger, K. Yanagisawa, Y. Stylianou, M. Akamine, M. Gales, and R. Cipolla, "Photo-realistic expressive text to talking head synthesis," in *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, Lyon, France, 2013.

[91] R. Anderson, B. Stenger, V. Wan, and R. Cipolla, "An expressive text-driven 3d talking head," in *ACM SIGGRAPH 2013 Posters*. New York, NY: ACM, 2013, pp. 80:1–80:1.

[92] ——, "Expressive visual text-to-speech using active appearance models," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Washington, DC: IEEE Computer Society, 2013, pp. 3382–3389.

[93] H. Zen, T. Nose, J. Yamagishi, S. Sako, T. Masuko, A. W. Black, and K. Tokuda, "The HMM-based speech synthesis system (HTS) version 2.0," in *The Seventh ISCA Tutorial and Research Workshop on Speech Synthesis (SSW)*, Kyoto, Japan, 2007.

[94] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 681–685, Jun 2001.

[95] M. Haghighat, S. Zonouz, and M. Abdel-Mottaleb, "CloudID: Trustworthy cloud-based and cross-enterprise biometric identification," *Expert Systems with Applications*, vol. 42, no. 21, pp. 7905 – 7916, 2015.

[96] Q. F. Hassan, "Demystifying cloud computing," *CrossTalk: The Journal of Defense Software Engineering*, vol. 24, 2011.

[97] P. Mell and T. Grance, "The NIST definition of cloud computing," *Communications of the ACM*, vol. 53, 01 2011.

[98] (2017) What is cloud computing?. Amazon web services. [Online]. Available: https://aws.amazon.com/what-is-cloud-computing/.

[99] (2018) What is cloud computing? A beginner's guide. [Online]. Available: https://azure.microsoft.com/en-ca/overview/what-is-cloud-computing/

[100] (2018) Why choose Azure vs. AWS — Microsoft Azure. [Online]. Available: https://azure.microsoft.com/en-us/overview/azure-vs-aws/

[101] (2018) Amazon Web Services (AWS) - Cloud Computing Services. [Online]. Available: https://aws.amazon.com/

[102] (2017) Home of the Blender project - free and open 3D creation software. [Online]. Available: https://www.blender.org/

[103] (2018) Maya — computer animation and modeling software — autodesk. [Online]. Available: https://www.autodesk.ca/en/products/maya/overview

[104] (2018) Unity. [Online]. Available: https://unity3d.com/

[105] Y. M. Chen, F. C. Huang, S. H. Guan, and B. Y. Chen, "Animating lip-sync characters with dominated animeme models," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 9, pp. 1344–1353, Sept 2012.

[106] F.-C. Huang, Y.-M. Chen, T.-H. Wang, B.-Y. Chen, and S.-H. Guan, "Animating lip-sync speech faces by dominated animeme models," in *SIGGRAPH '09: Posters*. New York, NY: ACM, 2009, pp. 2:1–2:1.

[107] L. Wang, W. Han, and F. K. Soong, "High quality lip-sync animation for 3d photo-realistic talking head," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, Japan, March 2012, pp. 4529–4532.

[108] G. Zoric and I. S. Pandzic, "A real-time lip sync system using a genetic algorithm for automatic neural network configuration," in *2005 IEEE International Conference on Multimedia and Expo*, Amsterdam, Netherlands, July 2005, pp. 1366–1369.

[109] Pluralsight. (2013) Proven tips for animating believable lip sync. [Online]. Available: https://www.pluralsight.com/blog/film-games/proven-tips-animating-believable-lip-sync

[110] (2018) Create 3D talking heads with CrazyTalk. [Online]. Available: https://www.reallusion.com/crazytalk/

[111] (2018) Faceshift. [Online]. Available: http://openni.ru/solutions/faceshift/index.html

[112] (2018) Facerig. [Online]. Available: https://facerig.com/

[113] (2017) Quicktalk lip synch addon. [Online]. Available: Available:https://tentacles.org.uk/quicktalk

[114] G. S. Almasi and A. Gottlieb, *Highly Parallel Computing*. Redwood City, CA, USA: Benjamin-Cummings Publishing Co., Inc., 1989.

[115] (2017) Measuring input latency — renderingpipeline. [Online]. Available: http://renderingpipeline.com/2013/09/measuring-input-latency/

[116] G. Skantze, "Real-time coordination in human-robot interaction using face and voice," *Ai Magazine*, vol. 37, pp. 19–31, 12 2016.

[117] F. Alonso-Martn, M. Malfaz, J. Sequeira, J. F. Gorostiza, and M. A. Salichs, "A multimodal emotion detection system during humanrobot interaction," *Sensors*, vol. 13, no. 11, pp. 15 549–15 581, 2013.

[118] M. Dragone, B. R. Duffy, and G. M. P. O'Hare, "Social interaction between robots, avatars humans," in *ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication, 2005.*, Nashville, TN, Aug 2005, pp. 24–29.

[119] A. L. Marin, D. Jo, and S. Lee, "Designing robotic avatars are user's impression affected by avatar's age?" in *8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Tokyo, Japan, March 2013, pp. 195–196.

[120] Y. Pan and A. Steed, "A comparison of avatar-, video-, and robot-mediated interaction on users trust in expertise," *Frontiers in Robotics and AI*, vol. 3, p. 12, 2016.

[121] (2017) andre-luiz-dos-santos/speech-app - github.s. [Online]. Available: https://github.com/andre-luiz-dos-santos/speech-app

[122] (2017) Mhx2 documentation.. [Online]. Available: https://thomasmakehuman.wordpress.com/mhx2-documentation

[123] (2017) 13 ways to reduce your render times. [Online]. Available: https://www.blenderguru.com/articles/13-ways-to-reduce-render-times

[124] (2018) Virtualgl the virtualgl project. [Online]. Available: https://www.virtualgl.org/

[125] S. Schneider and F. Kummert, "Does the user's evaluation of a socially assistive robot change based on presence and companionship type?" in *Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. New York, NY: ACM, 2017, pp. 277–278.

[126] K. Hassani, A. Nahvi, and A. Ahmadi, "Design and implementation of an intelligent virtual environment for improving speaking and listening skills," *Interactive Learning Environments*, vol. 24, no. 1, pp. 252–271, 2016.

[127] W.-Y. Liang, C.-C. Huang, T.-L. B. Tseng, Y.-C. Lin, and J. Tseng, "The evaluation of intelligent agent performance an example of B2C e-commerce negotiation," *Computer Standards and Interfaces*, vol. 34, no. 5, pp. 439 – 446, 2012.

[128] M. Moore and T. Ahmed, "Implementation and evaluation of an virtual intelligent agent," in *AMIA Annual Symposium Proceedings*, 2003, p. 941.

[129] I. S. MacKenzie, *Human-Computer Interaction: An Empirical Research Perspective*, 1st ed. San Francisco, CA: Morgan Kaufmann Publishers Inc., 2013.

[130] (2017) Human-computer interaction: An empirical research perspective. [Online]. Available: http://www.yorku.ca/mack/HCIbook/

[131] N. M. Radziwill and M. C. Benton, "Evaluating quality of chatbots and intelligent conversational agents," *CoRR*, vol. abs/1704.04579, 2017.

[132] (2017) Questionnaire for user interaction satisfaction. [Online]. Available: https://en.wikipedia.org/wiki/Questionnaire_for_User_Interaction_Satisfaction

[133] (2017) survey design and implementation in hci. [Online]. Available: http://wiki.ggc.usg.edu/images/c/cf/QuestionnaireDesignInHCI_ozok2008.pdf

[134] (2017) How to design questionnaires for usability evaluatione. [Online]. Available: http://www.shengdongzhao.com/research_tips/how-to-design-a-questionnaire-for-usability-evaluation/

[135] (2017) Post-evaluation questionnaire. [Online]. Available: Availablehttp://ece.ubc.ca/~pooya/hestudy/pc1/postevalquest.html

[136] J. P. Chin, V. A. Diehl, and K. L. Norman, "Development of an instrument measuring user satisfaction of the human-computer interface," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*. New York, NY, USA: ACM, 1988, pp. 213–218.

[137] J. R. Lewis, "IBM computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use," *Int. J. Hum.-Comput. Interact.*, vol. 7, no. 1, pp. 57–78, Jan. 1995.

[138] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS Q.*, vol. 13, no. 3, pp. 319–340, Sep. 1989.

[139] J. Hong and L. Findlater, "Identifying speech input errors through audio-only interaction," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, ser. CHI '18. New York, NY, USA: ACM, 2018, pp. 567:1–567:12. [Online]. Available: http://doi.acm.org/10.1145/3173574.3174141

[140] P. S. Houts, C. C. Doak, L. G. Doak, and M. J. Loscalzo, "The role of pictures in improving health communication: A review of research on attention, comprehension, recall, and adherence," *Patient Education and Counseling*, vol. 61, no. 2, pp. 173 – 190, 2006.

[141] P. Madhavan, "Handbook of warnings; edited by michael s. wogalter; 2006, 841 pages, $260.00; mahwah, nj: Lawrence erlbaum associates; isbn 0805847243," *Ergonomics in Design*, vol. 15, no. 3, pp. 32–33, 2007.

[142] S. Honig and T. Oron-Gilad, "Understanding and resolving failures in human-robot interaction: Literature review and model development," *Frontiers in psychology*, vol. 9, Jun 2018.

[143] P. Y. Weinstock A., Oron-Gilad T., "The effect of system aesthetics on trust, cooperation, satisfaction and annoyance in an imperfect automated system," *Work 41 (Suppl. 1)*, pp. 258–265, 2012.

[144] M. Salem, G. Lakatos, F. Amirabdollahian, and K. Dautenhahn, "Would you trust a (faulty) robot?: Effects of error, task type and personality on human-robot cooperation and trust," in *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*. New York, NY, USA: ACM, 2015, pp. 141–148.

[145] D. B. Roe and J. G. Wilpon, Eds., *Voice Communication Between Humans and Machines*. Washington, DC, USA: National Academy Press, 1994.

[146] H.-D. Kim, J.-S. Choi, and M. Kim, "Human-robot interaction in real environments by audio-visual integration," *International Journal of Control, Automation and Systems*, vol. 5, 02 2007.

Chapter A

Appendix A

A.1 Informed Consent Form

Informed Consent Form

Study name:
A Cloud- based Extensible Avatar (EA) for Human Robot Interaction

Researcher:
Enas Khaled Altarawneh
Electrical engineering and computer science department
MSc of Computer Science
Email: enas@cse.yorku.ca

Purpose of the research: The purpose of the research is to evaluate the user's perception and acceptance of a realistic computer-generated avatar that can provide Intelligent responses to users questions during an interaction. During the interaction quantitative data will be captured. This data is related to the application itself and includes the latency of the response, whether a response was generated at all and the overall time required for the response. This data will help identify the validity and reliability of the system for real-time interactions.. This study compares the users perception and acceptance of different interface forms including Text only, Audio only, a Realistic Avatar and a Cartoon Avatar. The data related to the user's perception of the application is gathered through a questionnaire and will be used to understand the user's preference among the different interfaces and to survey the user on the acceptable venues in which these interfaces can be used. Collected information will be used in a MASc in Computer Engineering thesis including resulting documents, papers, data archives and presentations. You will be assigned a unique subject number when you begin the study and only you and the experimenter will know the linkage between your identity and your subject number. All data collected outside of this Informed Consent Form will only identify the data by this subject number.

What you will be asked to do in the research: As a participant you will be asking an application (The Avatar) questions through text and speech and waiting for a response. The

Avatar will be presented as a stationary autonomous robot using this application. You will ask each of four interfaces for a response to fifteen pre-set questions. You will attempt to ask each question up to three times, or until you obtain a satisfactory answer, whichever comes first. You will also participate in two computer-presented questionnaires. A pre-study questionnaire will be used to obtain non-identifying personal information and a baseline of your perceptions and expectations in terms of HRI. A post-study questionnaire will ask you about your experiences during the testing. During the testing process itself you will be observed for attentiveness (the interval of times in which you look at the display will be collected and summed) and monitored for performance (time it takes you to ask a question, number of tries it takes you to obtain a satisfactory response). The user study will extract information related to the application itself and your perception of the application. The estimated time to participate in this study is 2 hours. No incentives are offered.

Benefits of the research: There is no direct benefit to the participant, however the study may highlight whether avatars are a more acceptable form of communication when requesting online information. This would provide justification for further research and application engineering that could elevate the human-computer interaction by adding a visual component to audio or text applications.

Voluntary participation: Your participation in the research is completely voluntary and you may choose to stop participating at any time. Your decision not to continue participating will not influence your relationship or the nature of your relationship with the researchers or the staff of York University either now or in the future.

Withdrawal from the study: You may stop participating in the study at any time, for any reason, if you so decide. Your decision to stop participating, or to refuse to answer particular

questions, will not affect your relationship with the researchers, York University, or any other group associated with this project. In the event that you withdraw from the study, all associated data collected will be immediately destroyed wherever possible.

Confidentiality: The questionnaire will not request identifying information. The data will be stored indefinitely in a password protected laptop and document. Only the researcher and the supervisor will have access to the stored data. Your data will only ever be identified by a subject number and there will be no way to link your subject number to your identity.

Questions about the research? If you have questions about the research in general or their role in the study you should contact the researcher, the supervisor and/or the graduate program office.

Supervisor:
Micheal jenkin
E-mail:Jenkin @eecs.yorku.ca

Graduate program office :
Department of Electrical Engineering & Computer Science and Engineering
LAS 1012M
York University
Telephone: 416-736-5053
E-mail: enquiries@cse.yorku.ca

This research has been reviewed and approved by the Human Participants Review Sub-Committee, York University's Ethics Review Board and conforms to the standards of the Canadian Tri-Council Research Ethics guidelines. If you have any questions about this process or about your rights as a participant in the study, you may contact:

The Senior Manager and Policy Advisor for the Office of Research Ethics
5th Floor, York Research Tower
York University
Telephone: 416-736-5914
E-mail:ore@yorku.ca

A.2 Pre-experiment questionnaire

Avatar evaluation pre questionnaire

A CLOUD-BASED EXTENSIBLE AVATAR (EA) FOR HUMAN ROBOT INTERACTION
Master thesis
Supervisor: Michael Jenkin
Department of Electrical Engineering and Computer Science
York University

* Required

1. **Participant Number** *
Mark only one oval.

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24

2. **group Layout** *
Mark only one oval.

- T-A-RA-CA
- A-RA-CA-T
- RA-CA-T-A
- CA-T-A-RA

Questions

Please fill in the below with appropriate answers:

3. **Year of birth ?** *

4. **What is you gender?**
Mark only one oval.

- ◯ Female
- ◯ Male
- ◯ Other

5. **What is the highest level of education you have completed?** *
Mark only one oval.

- ◯ Grammar School
- ◯ High School or equivalent
- ◯ Vocational/Technical School (2 year)
- ◯ Some College
- ◯ College Graduate (4 year)
- ◯ Master's Degree (MS)
- ◯ Doctoral Degree (PhD)
- ◯ Professional Degree (MD,JD, etc.)
- ◯ Other

6. **On average, how often do you use knowledge engines in general?** *
Mark only one oval.

- ◯ More than 9 times/day
- ◯ 5 to 8 times/day
- ◯ 1 to 4 times/day
- ◯ A few times a week
- ◯ Once a week
- ◯ Once a month

7. **Rate your experience with knowledge engines to retrieve text based results (examples, google, wolfarm Alpha…etc.).** *
Mark only one oval.

	1	2	3	4	5	6	7	
Highly Experienced	◯	◯	◯	◯	◯	◯	◯	Not Experienced

8. **Rate your experience with knowledge engines to retrieve voice based results (examples, Siri, Alexa, Cortna …etc.).** *
Mark only one oval.

	1	2	3	4	5	6	7	
Highly Experienced	◯	◯	◯	◯	◯	◯	◯	Not Experienced

9. **Rate your experience with knowledge engines that use of animated character lip-syncing the audio result** *

Mark only one oval.

	1	2	3	4	5	6	7	
Highly Experienced	◯	◯	◯	◯	◯	◯	◯	Not Experienced

A.3 Post-experiment questionnaire

Avatar evaluation post questionnaire

A CLOUD-BASED EXTENSIBLE AVATAR (EA) FOR HUMAN ROBOT INTERACTION
Master thesis
Supervisor: Michael Jenkin
Department of Electrical Engineering and Computer Science
York University

* Required

1. **Participant Number** *
Mark only one oval.

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24

2. **group Layout** *
Mark only one oval.

- T-A-RA-CA
- A-RA-CA-T
- RA-CA-T-A
- CA-T-A-RA

Questions for 1st tool

Provide the most appropriate answer in your opinion

3. **Rate your satisfaction with the responses given to you by the interface.** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not satisfied	◯	◯	◯	◯	◯	◯	◯	Very satisfied

4. **Rate your satisfaction with the amount of time it took to get a responses** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not satisfied	◯	◯	◯	◯	◯	◯	◯	Very satisfied

5. **How accurate was the responses given to you ?** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not accurate	◯	◯	◯	◯	◯	◯	◯	Very accurate

6. **How likely would you use this interface to retrieve similar information?** *
Mark only one oval.

	1	2	3	4	5	6	7	
not likely	◯	◯	◯	◯	◯	◯	◯	very likely

7. **How fun was it to getting responses using the tool?** *
Mark only one oval.

	1	2	3	4	5	6	7	
not fun	◯	◯	◯	◯	◯	◯	◯	very fun

8. **In terms of simplicity/ease of use how did the interaction with the tool feel?** *
Mark only one oval.

	1	2	3	4	5	6	7	
forced	◯	◯	◯	◯	◯	◯	◯	intutive

9. **The appearance of the displayed responses on the screen is** *
Mark only one oval.

	1	2	3	4	5	6	7	
consistent	◯	◯	◯	◯	◯	◯	◯	inconsistent

10. **In terms of trust, do you agree that you were being taken seriously by the tool?** *
Mark only one oval.

	1	2	3	4	5	6	7	
strongly agree	◯	◯	◯	◯	◯	◯	◯	strongly disagree

11. **In terms of trust, do you agree that you were taking the tool seriously?** *
Mark only one oval.

	1	2	3	4	5	6	7	
strongly agree	◯	◯	◯	◯	◯	◯	◯	strongly disagree

Questions for 2nd tool

Provide the most appropriate answer in your opinion

12. **Rate your satisfaction with the responses given to you by the interface.** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not satisfied	◯	◯	◯	◯	◯	◯	◯	Very satisfied

13. **Rate your satisfaction with the amount of time it took to get a responses** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not satisfied	◯	◯	◯	◯	◯	◯	◯	Very satisfied

14. **How accurate was the responses given to you ?** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not accurate	◯	◯	◯	◯	◯	◯	◯	Very accurate

15. **How likely would you use this interface to retrieve similar information?** *
Mark only one oval.

	1	2	3	4	5	6	7	
not likely	◯	◯	◯	◯	◯	◯	◯	very likely

16. **How fun was it to getting responses using the tool?** *
Mark only one oval.

	1	2	3	4	5	6	7	
not fun	◯	◯	◯	◯	◯	◯	◯	very fun

17. **In terms of simplicity/ease of use how did the interaction with the tool feel?** *
Mark only one oval.

	1	2	3	4	5	6	7	
forced	◯	◯	◯	◯	◯	◯	◯	intutive

18. **The appearance of the displayed responses on the screen is** *
Mark only one oval.

	1	2	3	4	5	6	7	
consistent	◯	◯	◯	◯	◯	◯	◯	inconsistent

19. **In terms of trust, do you agree that you were being taken seriously by the tool?** *
Mark only one oval.

	1	2	3	4	5	6	7	
strongly agree	◯	◯	◯	◯	◯	◯	◯	strongly disagree

20. **In terms of trust, do you agree that you were taking the tool seriously?** *
Mark only one oval.

	1	2	3	4	5	6	7	
strongly agree	◯	◯	◯	◯	◯	◯	◯	strongly disagree

Questions for 3rd tool

Provide the most appropriate answer in your opinion

21. **Rate your satisfaction with the responses given to you by the interface.** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not satisfied	◯	◯	◯	◯	◯	◯	◯	Very satisfied

22. **Rate your satisfaction with the amount of time it took to get a responses** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not satisfied	◯	◯	◯	◯	◯	◯	◯	Very satisfied

23. **How accurate was the responses given to you ?** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not accurate	◯	◯	◯	◯	◯	◯	◯	Very accurate

24. **How likely would you use this interface to retrieve similar information?** *
Mark only one oval.

	1	2	3	4	5	6	7	
not likely	◯	◯	◯	◯	◯	◯	◯	very likely

25. **How fun was it to getting responses using the tool?** *
Mark only one oval.

	1	2	3	4	5	6	7	
not fun	◯	◯	◯	◯	◯	◯	◯	very fun

26. **In terms of simplicity/ease of use how did the interaction with the tool feel?** *
Mark only one oval.

	1	2	3	4	5	6	7	
forced	◯	◯	◯	◯	◯	◯	◯	intutive

27. **The appearance of the displayed responses on the screen is** *
Mark only one oval.

	1	2	3	4	5	6	7	
consistent	◯	◯	◯	◯	◯	◯	◯	inconsistent

28. **In terms of trust, do you agree that you were being taken seriously by the tool?**
Mark only one oval.

	1	2	3	4	5	6	7	
strongly agree	◯	◯	◯	◯	◯	◯	◯	strongly disagree

29. **In terms of trust, do you agree that you were taking the tool seriously?** *
Mark only one oval.

	1	2	3	4	5	6	7	
strongly agree	◯	◯	◯	◯	◯	◯	◯	strongly disagree

Questions for 4th tool

Provide the most appropriate answer in your opinion

30. **Rate your satisfaction with the responses given to you by the interface.** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not satisfied	◯	◯	◯	◯	◯	◯	◯	Very satisfied

31. **Rate your satisfaction with the amount of time it took to get a responses** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not satisfied	◯	◯	◯	◯	◯	◯	◯	Very satisfied

32. **How accurate was the responses given to you ?** *
Mark only one oval.

	1	2	3	4	5	6	7	
Not accurate	◯	◯	◯	◯	◯	◯	◯	Very accurate

33. **How likely would you use this interface to retrieve similar information?** *
Mark only one oval.

	1	2	3	4	5	6	7	
not likely	◯	◯	◯	◯	◯	◯	◯	very likely

34. **How fun was it to getting responses using the tool?** *
Mark only one oval.

	1	2	3	4	5	6	7	
not fun	◯	◯	◯	◯	◯	◯	◯	very fun

35. **In terms of simplicity/ease of use how did the interaction with the tool feel?** *
Mark only one oval.

	1	2	3	4	5	6	7	
forced	◯	◯	◯	◯	◯	◯	◯	intutive

36. **The appearance of the displayed responses on the screen is** *
Mark only one oval.

	1	2	3	4	5	6	7	
consistent	◯	◯	◯	◯	◯	◯	◯	inconsistent

37. **In terms of trust, do you agree that you were being taken seriously by the tool?**
Mark only one oval.

	1	2	3	4	5	6	7	
strongly agree	◯	◯	◯	◯	◯	◯	◯	strongly disagree

38. **In terms of trust, do you agree that you were taking the tool seriously?** *
Mark only one oval.

	1	2	3	4	5	6	7	
strongly agree	◯	◯	◯	◯	◯	◯	◯	strongly disagree

Overall preference between tools

Provide the most appropriate answer in your opinion

39. **Select your level of preference between the text and voice interfaces** *
Mark only one oval.

	1	2	3	4	5	6	7	
text interface	◯	◯	◯	◯	◯	◯	◯	voice interface

40. **Select your level of preference between the Voice and Avatar interfaces** *
Mark only one oval.

	1	2	3	4	5	6	7	
voice interface	◯	◯	◯	◯	◯	◯	◯	avatar interface

41. **Select your level of preference between the realistic avatar and the cartoonish Avatar interfaces** *

Mark only one oval.

	1	2	3	4	5	6	7	
realistic avatar interface	⬭	⬭	⬭	⬭	⬭	⬭	⬭	cartoonish Avatar interface

Chapter B

Appendix B

B.1 Post hoc results for input time

```
------------------------------------------------------------
---------- Pairwise Comparisons (Bonferroni-Dunn) ----------
------------------------------------------------------------
Pair  1:2  -->     0.71   >     0.36   ?     * (significant)
Pair  1:3  -->     0.11   >     0.36   ?     -
Pair  1:4  -->     0.13   >     0.36   ?     -
Pair  2:3  -->     0.60   >     0.36   ?     * (significant)
Pair  2:4  -->     0.57   >     0.36   ?     * (significant)
Pair  3:4  -->     0.02   >     0.36   ?     -
------------------------------------------------------------
```

B.2 Post hoc results for mean response generation time

```
-----------------------------------------------------------
---------- Pairwise Comparisons (Bonferroni-Dunn) ----------
-----------------------------------------------------------
Pair  1:2  -->     0.25   >     0.37   ?     -
Pair  1:3  -->     0.72   >     0.37   ?     * (significant)
Pair  1:4  -->     3.10   >     0.37   ?     * (significant)
Pair  2:3  -->     0.46   >     0.37   ?     * (significant)
Pair  2:4  -->     2.84   >     0.37   ?     * (significant)
Pair  3:4  -->     2.38   >     0.37   ?     * (significant)
-----------------------------------------------------------
Data_file: responsetime_posthoc.txt
```

B.3 Post hoc results for participant attentiveness

```
-----------------------------------------------------------
---------- Pairwise Comparisons (Bonferroni-Dunn) ----------
-----------------------------------------------------------
Pair  1:2  -->     0.23   >     0.01   ?     * (significant)
Pair  1:3  -->     0.12   >     0.01   ?     * (significant)
Pair  1:4  -->     0.07   >     0.01   ?     * (significant)
Pair  2:3  -->     0.11   >     0.01   ?     * (significant)
Pair  2:4  -->     0.16   >     0.01   ?     * (significant)
Pair  3:4  -->     0.05   >     0.01   ?     * (significant)
-----------------------------------------------------------
Data_file: attentive_posthoc.txt
```

B.4 Post hoc results for query failure rate

```
------------------------------------------------------------
---------- Pairwise Comparisons (Bonferroni-Dunn) ----------
------------------------------------------------------------
Pair  1:2  -->     0.09   >     0.06   ?     * (significant)
Pair  1:3  -->     0.00   >     0.06   ?     -
Pair  1:4  -->     0.01   >     0.06   ?     -
Pair  2:3  -->     0.09   >     0.06   ?     * (significant)
Pair  2:4  -->     0.08   >     0.06   ?     * (significant)
Pair  3:4  -->     0.01   >     0.06   ?     -
------------------------------------------------------------
Data_file: error_posthoc.txt

------------------------------------------------------------
```

B.5 Post hoc results for participant satisfaction with the interaction

```
H(3) = 7.637, p = 0.0541
H'(3) = 11.826, p' = 0.0080

------------------------------------------------------------------
------------ Pairwise Comparisons (using Conover's F) ------------
------------------------------------------------------------------
Pair  1:2  --> abs( 2.833 -  1.896)  >  0.558  ?     * (significant)
Pair  1:3  --> abs( 2.833 -  2.729)  >  0.558  ?     -
Pair  1:4  --> abs( 2.833 -  2.542)  >  0.558  ?     -
Pair  2:3  --> abs( 1.896 -  2.729)  >  0.558  ?     * (significant)
Pair  2:4  --> abs( 1.896 -  2.542)  >  0.558  ?     * (significant)
Pair  3:4  --> abs( 2.729 -  2.542)  >  0.558  ?     -
------------------------------------------------------------------
```

B.6 Post hoc results for participant satisfaction with the time to obtain a response from the interface

```
H(3) = 7.725, p = 0.0521
H'(3) = 10.243, p' = 0.0166

----------------------------------------------------------------
------------ Pairwise Comparisons (using Conover's F) ------------
----------------------------------------------------------------
Pair  1:2  --> abs( 3.104 -  2.188)  >  0.611  ?     * (significant)
Pair  1:3  --> abs( 3.104 -  2.229)  >  0.611  ?     * (significant)
Pair  1:4  --> abs( 3.104 -  2.479)  >  0.611  ?     * (significant)
Pair  2:3  --> abs( 2.188 -  2.229)  >  0.611  ?     -
Pair  2:4  --> abs( 2.188 -  2.479)  >  0.611  ?     -
Pair  3:4  --> abs( 2.229 -  2.479)  >  0.611  ?     -
----------------------------------------------------------------
```

B.7 Post hoc results for participant perception on accuracy of the responses

```
H(3) = 3.712, p = 0.2942
H'(3) = 14.143, p' = 0.0027

----------------------------------------------------------------
------------ Pairwise Comparisons (using Conover's F) ------------
----------------------------------------------------------------
Pair  1:2  --> abs( 2.771 -  2.083)  >  0.349  ?     * (significant)
Pair  1:3  --> abs( 2.771 -  2.563)  >  0.349  ?     -
Pair  1:4  --> abs( 2.771 -  2.583)  >  0.349  ?     -
Pair  2:3  --> abs( 2.083 -  2.563)  >  0.349  ?     * (significant)
Pair  2:4  --> abs( 2.083 -  2.583)  >  0.349  ?     * (significant)
Pair  3:4  --> abs( 2.563 -  2.583)  >  0.349  ?     -
----------------------------------------------------------------
```

B.8 Post hoc results for participant perception of how fun each interface is to use

```
H(3) = 13.188, p = 0.0042
H'(3) = 15.746, p' = 0.0013

----------------------------------------------------------------
------------ Pairwise Comparisons (using Conover's F) ------------
----------------------------------------------------------------
Pair  1:2  --> abs( 1.896 -  2.167)  >  0.614  ?     -
Pair  1:3  --> abs( 1.896 -  2.958)  >  0.614  ?     * (significant)
Pair  1:4  --> abs( 1.896 -  2.979)  >  0.614  ?     * (significant)
Pair  2:3  --> abs( 2.167 -  2.958)  >  0.614  ?     * (significant)
Pair  2:4  --> abs( 2.167 -  2.979)  >  0.614  ?     * (significant)
Pair  3:4  --> abs( 2.958 -  2.979)  >  0.614  ?     -
----------------------------------------------------------------
```

B.9 Post hoc results for participant perception of the ease of use of each interface

```
H(3) = 0.962, p = 0.8103
H'(3) = 1.650, p' = 0.6481

----------------------------------------------------------------
------------ Pairwise Comparisons (using Conover's F) ------------
----------------------------------------------------------------
Pair  1:2  --> abs( 2.458 -  2.354)  >  0.573  ?     -
Pair  1:3  --> abs( 2.458 -  2.479)  >  0.573  ?     -
Pair  1:4  --> abs( 2.458 -  2.708)  >  0.573  ?     -
Pair  2:3  --> abs( 2.354 -  2.479)  >  0.573  ?     -
Pair  2:4  --> abs( 2.354 -  2.708)  >  0.573  ?     -
Pair  3:4  --> abs( 2.479 -  2.708)  >  0.573  ?     -
----------------------------------------------------------------
```

B.10 Post hoc results for participant likelihood to use the interface in the future

```
H(3) = 0.938, p = 0.8164
H'(3) = 1.125, p' = 0.7710

------------------------------------------------------------------
------------ Pairwise Comparisons (using Conover's F) ------------
------------------------------------------------------------------
Pair  1:2  --> abs( 2.688 -  2.333)  >  0.688  ?     -
Pair  1:3  --> abs( 2.688 -  2.521)  >  0.688  ?     -
Pair  1:4  --> abs( 2.688 -  2.458)  >  0.688  ?     -
Pair  2:3  --> abs( 2.333 -  2.521)  >  0.688  ?     -
Pair  2:4  --> abs( 2.333 -  2.458)  >  0.688  ?     -
Pair  3:4  --> abs( 2.521 -  2.458)  >  0.688  ?     -
------------------------------------------------------------------
```

B.11 Post hoc results for participant perception of the consistency of the interface

```
H(3) = 2.500, p = 0.4753
H'(3) = 8.824, p' = 0.0317

------------------------------------------------------------------
------------ Pairwise Comparisons (using Conover's F) ------------
------------------------------------------------------------------
Pair  1:2  --> abs( 2.208 -  2.458)  >  0.378  ?     -
Pair  1:3  --> abs( 2.208 -  2.792)  >  0.378  ?     * (significant)
Pair  1:4  --> abs( 2.208 -  2.542)  >  0.378  ?     -
Pair  2:3  --> abs( 2.458 -  2.792)  >  0.378  ?     -
Pair  2:4  --> abs( 2.458 -  2.542)  >  0.378  ?     -
Pair  3:4  --> abs( 2.792 -  2.542)  >  0.378  ?     -
------------------------------------------------------------------
```

B.12 Post hoc results for participant perception of interface seriousness

```
H(3) = 3.212, p = 0.3600
H'(3) = 10.419, p' = 0.0153

----------------------------------------------------------------
------------ Pairwise Comparisons (using Conover's F) ------------
----------------------------------------------------------------
Pair  1:2  --> abs( 2.146 -  2.458)  >  0.390  ?     -
Pair  1:3  --> abs( 2.146 -  2.792)  >  0.390  ?     * (significant)
Pair  1:4  --> abs( 2.146 -  2.604)  >  0.390  ?     * (significant)
Pair  2:3  --> abs( 2.458 -  2.792)  >  0.390  ?     -
Pair  2:4  --> abs( 2.458 -  2.604)  >  0.390  ?     -
Pair  3:4  --> abs( 2.792 -  2.604)  >  0.390  ?     -
----------------------------------------------------------------
```

B.13 Post hoc results for how serious the participants were about the interface

```
H(3) = 1.012, p = 0.7982
H'(3) = 3.857, p' = 0.2773

----------------------------------------------------------------
------------ Pairwise Comparisons (using Conover's F) ------------
----------------------------------------------------------------
Pair  1:2  --> abs( 2.292 -  2.479)  >  0.378  ?     -
Pair  1:3  --> abs( 2.292 -  2.604)  >  0.378  ?     -
Pair  1:4  --> abs( 2.292 -  2.625)  >  0.378  ?     -
Pair  2:3  --> abs( 2.479 -  2.604)  >  0.378  ?     -
Pair  2:4  --> abs( 2.479 -  2.625)  >  0.378  ?     -
Pair  3:4  --> abs( 2.604 -  2.625)  >  0.378  ?     -
----------------------------------------------------------------
```