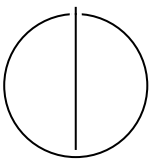
SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY — INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Evaluating the Efficiency of Human-AI Interaction in Virtual Reality Using Foundation Models

Md Razaul Haque Usmani



SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY — INFORMATICS

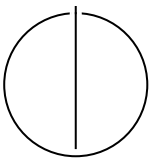
TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Evaluating the Efficiency of Human-AI Interaction in Virtual Reality Using Foundation Models

Bewertung der Effizienz der Mensch-KI-Interaktion in der Virtuellen Realität unter Verwendung von Foundation-Modellen

Author:	Md Razaul Haque Usmani
Supervisor:	Prof. David A. Plecher
Advisors:	Christian Eichhorn and Yannick Weiß
Submission Date:	1st August, 2025



I confirm that this master's thesis is my own work and I have documented all sources and material used.

Munich, 1st August, 2025 Md Razaul Haque Usmani

Abstract

Large Language Models (LLMs) and AI-based tools have seen rapid adoption across various industries, and Virtual Reality (VR) applications are no exception. Many such systems incorporate LLMs through text - or speech -based input methods and hence, the image understanding capabilities of the newer models remain under-researched. To mitigate this, we propose an application focused on minimalism and accessibility, using the foundation model *GPT-4o* with zero-shot training, integrated with Unity and Quest 2, to explore whether LLMs can process a different form of input — hand gestures — via image recognition. The results are promising, achieving an accuracy of 73.6% despite a slower response time (6.8 seconds). A pilot study further demonstrates that gesture recognition performs competitively compared to conventional speech-to-text approaches and is a similarly desirable approach of interaction in VR.

Große Sprachmodelle (Large Language Models, LLMs) und KI-basierte Werkzeuge haben in verschiedenen Industriezweigen rasch an Verbreitung gewonnen, und Anwendungen im Bereich der Virtuellen Realität (VR) bilden hierbei keine Ausnahme. Viele dieser Systeme integrieren LLMs über text- oder sprachbasierte Eingabemethoden. Die Fähigkeit neuerer Modelle zur Bildverarbeitung bleibt jedoch bislang weitgehend unerforscht. Um diesem Umstand entgegenzuwirken, schlagen wir eine Anwendung vor, die auf Minimalismus und Barrierefreiheit ausgerichtet ist. Dabei nutzen wir das Foundation Model *GPT-4o* im Zero-Shot-Modus, integriert in Unity und Oculus Quest 2, um zu untersuchen, ob LLMs eine alternative Form der Eingabe – nämlich Handgesten – mittels Bilderkennung verarbeiten können. Die Ergebnisse sind vielversprechend: Es wurde eine Genauigkeit von 73,6 % erreicht, trotz einer vergleichsweise langen Reaktionszeit von 6,8 Sekunden. Eine Pilotstudie zeigt darüber hinaus, dass die Gestenerkennung im Vergleich zu konventionellen Speech-to-Text-Ansätzen eine konkurrenzfähige Leistung erzielt und als ebenso attraktive Interaktionsmöglichkeit in VR wahrgenommen wird.

Contents

Abstract **iv**

1. Introduction **1**
1.1. Motivation . . . 1
1.2. Document Structure . . . 2

2. Theoretical Background **3**
2.1. Artificial Intelligence and Language Models . . . 3
2.1.1. Generative and Conversational AI . . . 3
2.1.2. LLMs and GPTs . . . 4
2.2. Human Computer Interaction with Conversational AI . . . 6
2.3. Virtual Reality Immersion and Tracking . . . 6
2.3.1. Virtual Reality . . . 6
2.3.2. VR Immersion and Presence . . . 6
2.3.3. VR Tracking . . . 7
2.4. Gesture and Intent Recognition . . . 9

3. Related Works **10**
3.1. Human-AI Interaction . . . 10
3.2. Using LLMs in VR . . . 10
3.3. LLM Scene Understanding in VR . . . 11
3.4. LLM Gesture Understanding . . . 13
3.5. LLM-Based NPCs in VR . . . 14

4. Methodology **19**
4.1. Overview . . . 19
4.2. Implementation . . . 20
4.2.1. Level Design . . . 20
4.2.2. User Interface Design . . . 23
4.2.3. Quest Design . . . 25
4.3. System Development and Implementation . . . 26
4.3.1. Early Gesture Recognition Approaches . . . 26
4.3.2. Speech To Text Implementation as a Baseline . . . 27

4.3.3. Implementation of Modules: Core Functions 28
4.3.4. Implementation of Modules: Classes 31
4.3.5. Prompting 34
4.4. Experimental Setup 37
4.4.1. Apparatus 37
4.4.2. Experiment Design 38
4.4.3. Questionnaire Selection 42

5. Results **48**
5.1. Quantitative Results 48
5.1.1. Data Collection From Logs 48
5.1.2. Data from Questionnaires 53
5.2. Qualitative Results 58
5.2.1. Findings from the Interview 58
5.2.2. Experiment Observations 61

6. Discussion **65**
6.1. Learning Outcomes 65
6.2. Limitations 68
6.3. Future Work 70
6.4. Conclusion 71

A. Appendix A: Questionnaire Response Data **73**
A.0.1. About the Experience: Gesture Recognition Task 73
A.0.2. About the Experience: Speech To Text Task 75
A.0.3. ASAQ: Gesture Recognition Task 77
A.0.4. ASAQ: Speech To Text Task 79
A.0.5. SoAS: Gesture Recognition Task 81
A.0.6. SoAS: Speech To Text Task 82
A.0.7. MEC-Spatial Presence Questionnaire 83

B. Appendix B: Movement Heat-map Data **84**
B.1. Gesture Recognition Task 84
B.2. Speech To Text Task 85

C. Appendix C: Images from the VR Experience **86**
C.1. Speech To Text Task 86
C.2. Gesture Recognition Task 87

D. Appendix D: Additional Data **88**
D.1. NPC Animation Graphs 88
D.2. NPC Unique Questions 89
D.3. Shop Item Labels 90
D.4. VR Invitation Poster 91

Abbreviations **92**

List of Figures **94**

List of Tables **96**

Bibliography **97**

1. Introduction

1.1. Motivation

In recent years, the world of technology has witnessed a sudden and substantial shift in focus on how we interact with it - from the individual to the industry level. Creating a functioning Artificial General Intelligence (AGI) has demanded the attention of the industry and the current rush towards it - while only at its preliminary stage - is already reaching its peak. With the prevalence of Generative Artificial Intelligence (AI) and Large Language Models (LLM) (Vaswani et al. 2017), creating any form of digital content has become trivial. Consequently, a significant interest has naturally developed in incorporating such technologies to the realms of virtual and extended realities.

Despite this, there is a large unexplored field when it comes to understanding the human-AI interactions using such LLMs in virtual environments - specifically for human gesture and image recognition in general.Our study aims mitigate this gap by integrating such LLMs such as OpenAI's (GPT-4)(OpenAI 2023), in Meta's Quest Virtual Reality (VR)(Meta Platforms, Inc. 2020), world and creating an experience that would test the efficacy of the conversational model when interacting with users using gestures. Despite the widening potential and scope of such AI in virtual reality, the interactions are currently limited to text or voice input, and hence there is limited work done where motion and gesture are also incorporated to enhance the interactive experience. The paper goes into further discussion, where prior research includes features that allow one to record different gestures in VR beforehand[5], however, very few attempt an on-the-fly recognition in the VR space. Similarly, some existing studies are based on capturing clear images which are then used for recognition on-the-fly or accumulated as a dataset for further training and creating custom Generative Pre-trained Transformers (GPT) such as *GestureGPT* (Zeng et al. 2024) or *IntentGPT*(Rodriguez et al. 2024). However, very few have VR based, zero shot, real-time gesture recognition systems that utilizes or tests the stable, commercially available models. Therefore, the paper highlights this attempt at a novel approach in incorporating language models with virtual reality through tackling the following questions:

1. ***How accurately can the model identify the user's gesture?***

2. ***How quickly can the model read the user's gesture? Can the interactions occur in real time?***
3. ***How does the model impact the user experience in a given scenario?***

1.2. Document Structure

This paper is structured as follows:

- Chapter 2 Describes the theoretical key concepts concerning AI, LLM, VR, among others.
- Chapter 3 Reviews existing research on gesture and intention interpretation and content generation capabilities of LLMs, performance of virtual reality tracking and interfacing with humans.
- Chapter 4 Elaborates in-depth the methodology of the research - the development of the VR experience, the selection and implementation of the questionnaires and the experiment design.
- Chapter 5 Provides key insights and results obtained from the procedure described in Chapter 4
- Chapter 6 Concludes the research by discussing what was achieved, the shortcomings and its future trajectory.

2. Theoretical Background

2.1. Artificial Intelligence and Language Models

At the core of Large Language Models are Transformers. A Transformer is an architecture for handling sequential data, introduced by Vaswani et al. (2017), was initially used for NLP applications such as BERT(Devlin et al. 2019) and GPT(Radford, Narasimhan, et al. 2018) and later incorporated in images such as OpenAI's DALL-E(Ramesh et al. 2021). In comparison, sequential solvers such as Recursive Neural Networks (RNN) (Elman 1990) and Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber 1997) have a limited reference window for words, whereas the Attention mechanism used in Transformers allows for an infinite reference window given enough computational strength. The key parts are the *Positional Encoding* and *Multi Headed Attention* blocks in the transformer architecture. Transformers follow the encoder-decoder structure where the decoder takes the input from the encoder and provides a single output, while taking previous outputs as the inputs. This occurs recursively until an end of sequence is indicated. Moreover, unlike RNNs, transformers utilize parallelism through the compute power of GPUs and TPUs, allowing them to compute weights of all the tokens simultaneously, greatly accelerating the training process of such LLMs.

2.1.1. Generative and Conversational AI

Generative AI

Generative AI is a subset of artificial intelligence that utilises unsupervised learning to train on enormous quantities of data to identify patterns, styles and semantics and use this knowledge to create completely new data - including text, image, audio or video - with an emphasis on providing a human-like feeling to the content. As aforementioned, the models that utilise a transformer based architecture leverages the attention mechanism to learn the content and the context more efficiently. Natural language based models such as GPT and BERT(Devlin et al. 2019), or image processing and generating tools such as DALL-E (Ramesh et al. 2021) exploit the vast datasets and computing resources to generate content seemingly effortlessly. The work by Brynjolfsson et al. (2023) reports the (now older) GPT-3 model of working with 175

billion parameters and costing $5 million dollars in computing costs alone. While the more recent GPT-4 models are estimated to include trillions of parameters and costing significantly higher. The field of generative AI ultimately aims to achieve higher accuracy, realism and customisability while closing the gap between man and machine made content.

Conversational AI

Natural language based generative AI has expanded towards creating Conversational AI models that effectively understands human-like conversations and provide similarly naturally sounding output. Models such as ChatGPT are therefore trained and fine-tuned to provide coherent and relevant answers that retains conversation memory to provide maximum user satisfaction. Further use cases encompass voice-based virtual assistants such as Apple's Siri, customer service voice or chat assistant applications among others.

2.1.2. LLMs and GPTs

Multimodal Learning

Advancing from training models on a singular medium, multimodal learning takes a step further into mimicking human perception, concurrently learning from diverse input formats, which allows for handling more complex yet natural tasks such as describing the contents of an image in words.

Liang et al. (2022) explains the fundamentals and challenges of how effectively multimodal learning functions. It first gathers data of different types known as a **modality**, which may include text (natural language, symbols or otherwise), visual, aural, haptics, etc. Modalities are considered to be heterogeneous where a measure of the differences in each element in terms of its distribution, structure, task relevance among others are calculated. These are then used to create interconnections between the modalities to improve the learned context in different *fusion* stages. The *Alignment* process involves mapping the data to identify interactions between two or more modalities, for instance, matching text from a transcript with text extracted from an audio file. The model is then trained to create inference and reasoning behind the connections based on the specified task, create completely new, similarly relevant modalities and even transfer knowledge between each other to improve a target modality which may lack sufficient resources.

This is achieved using reliable transformer architectures, leveraging its attention mechanism to focus on the relevant parts of the modalities, using cross-attention layers to allow and observe interactions between them and autoencoders to improve

multimodal context and facilitate representation learning. Consequently, the utilization of such models are vast - from captioning images and speech translations, to self-driving vehicle navigation and multimedia healthcare monitoring systems. Multimodal learning hence lays a foundation to how large language models are now capable of learning from and providing output in different forms.

Large Language Models

LLMs are another form of transformer based models that train on large quantities of data as the name implies. This data, although initially natural language based derived from innumerable books, articles and websites, have now expanded to other forms with the aid of transfer and multimodal learning techniques. This paper (Naveed et al. 2023) elaborates in great detail the components of how LLMs function.

Tokenization refers to the process where text input is first broken down into small tokens in the word or grammatical element levels or in byte-pair encodings (Sennrich et al. 2015) where the frequently occurring pairs of characters are encoded to compress the string. Moreover, it utilises **positional encodings** in addition to the token embeddings to improve content structure accuracy. The self-attention mechanism assigns weights to the tokens and these are then used for pre-training. These weights or **parameters** are what models such as OpenAI's GPT-3 are trained on in the billions.

Pre-training involves training the model on massive datasets with a set **objective** (T. Wang et al. 2022) such as casual or masked language modelling to predict the next token in the sequence or a hidden token in the sentence respectively or train using multimodal data such as images and audio.

Generative Pre-trained Transformers

GPTs are examples of such large language models based on the transformer architecture. This paper on OpenAI's introductory GPT model from Radford, Narasimhan, et al. (2018) details the two step process of how the model functions: it first undergoes an unsupervised pre-training phase, learning in an autoregressive manner to understand the structure (and grammar in this case) of the data. It is then fine tuned based on a supervised set of data to handle task specific prompts and responses. As previously mentioned, subsequent iteration of the GPT models have significantly evolved with larger model size and training data, while the issues of data completion and biases, along with ethical concerns of misuse remain a challenge.

2.2. Human Computer Interaction with Conversational AI

As the name suggests, Human Computer Interaction or HCI is the study of designing, observing and interpreting the human behaviour and relation with computers and robots. Depending on the application, the user interaction may include prompting the machine for simple conversations in the forms of chatbots, seeking aid in language translation tasks or providing captured images for image and pattern recognition. With the emergence of generative and conversational AI, several studies have reported on the paradigm shift in such human-robot interactions (Nicolescu and Tudorache 2022) and (Liu 2024).

2.3. Virtual Reality Immersion and Tracking

2.3.1. Virtual Reality

Virtual Reality refers to the simulated immersive experience of a virtually created environment. It utilises wearable 3D near-eye displays that supports pose tracking, allowing users to move and look naturally around the artificial world. Further development has allowed users to interact with virtual objects within the scene with near real-time hand position and gesture tracking.

As the virtual reality technologies have expanded, new and immersive interaction options with the virtual world have been introduced in both the software and hardware level. VR headset providers such as Meta's Oculus have expanded the virtual reality market to a much wider consumer base whereas traditionally these technologies were only available to a niche market for research purposes. Each product is aimed towards their achieving a particular goal. For instance, Meta's Quest line of VR HMDs (or Head Mounted Displays) are targeted towards average users for ideally daily usage and so it strives to be more accessible via price, and ease of setup and use while compromising minute accuracy on tracking. In contrast, HTC's Vive is aimed towards optimal performance and tracking accuracy which requires a more complex setup with sufficient empty space, additional camera setup and wired connections.

2.3.2. VR Immersion and Presence

The vision of virtual reality is to create a perfect simulation where the user is fully immersed and presence which leads to the rise of study in these fields. While immersion and presence may be used interchangeably, it achieves niche features that influences the user's perception in and of the virtual world. Several studies from Slater provide in-depth breakdowns of immersion and presence: classification of the levels of immersion

(Slater 2018), expansion of the concept of presence (Slater 2009). **Immersion** is described as providing more senses for the user to work with in the virtual world. Being able to openly perceive in the visual, aural, haptic and locomotive or proprioception senses creates a stronger sense of immersion. Greater accessibility to such features combined with higher accuracy would result in a higher the level of immersion as described in Slater (2018).

While the term 'presence' has been deemed ambiguous for their research purpose, Slater (2009) describes it via the *Place Illusion* - the illusion that the user are actually present in the environment while they believe that they are not, in reality. The sense of agency the user feels - believing that they can manipulate and influence the virtual environment by their actions. This is achieved through (in addition to the accessibility to the basic senses) the sensory fidelity of the environment and ability to interact realistically with it. Furthermore, the belief that the world in turn has an impact on the users themselves, where one is convinced that the events are definitely occurring is described by the *Plausibility Illusion*. Both these illusions along with higher levels of immersion determines whether the user would 'Response As If Real' or RAIR, as coined by Slater.

Performance also plays a vital role in keeping the user immersion and providing the illusions of presence. Several technical factors such as the field-of-view or FOV, the frame rate or refresh rate, view resolution, etc. are key factors to ensure consistency, as mentioned in Sánchez-Vives and Slater (2005).

2.3.3. VR Tracking

Whether wired or wireless, these devices achieve motion tracking through optical and non-optical tracking technologies. Optical tracking depends on cameras either integrated to the HMDs or additionally installed within the recommended working area and hence requires well lit environments to track features like head and hands effectively. Non-optical tracking as described in Vox et al. (2021) uses markers placed on the body and organised hierarchically based on the parent and child joint rigs to record the kinematic data. The head mounted display primarily allows viewing the virtual world in 360 degrees in a 3-dimensional space using *inside-out* or *outside-in tracking*(Alanko 2023).

Inside-out tracking is where the cameras are placed within the HMDs, facing towards the environment to track the user's relative position. Computer vision and SLAM (Simultaneous Localization and Mapping) algorithms are used to constantly create dynamic maps and track visual markers such as object corners. Additional sensors such as gyroscopes and accelerometers provide measurements in inertial measurement units or IMUs that is used to further enhance the tracking accuracy. Having all the hardware

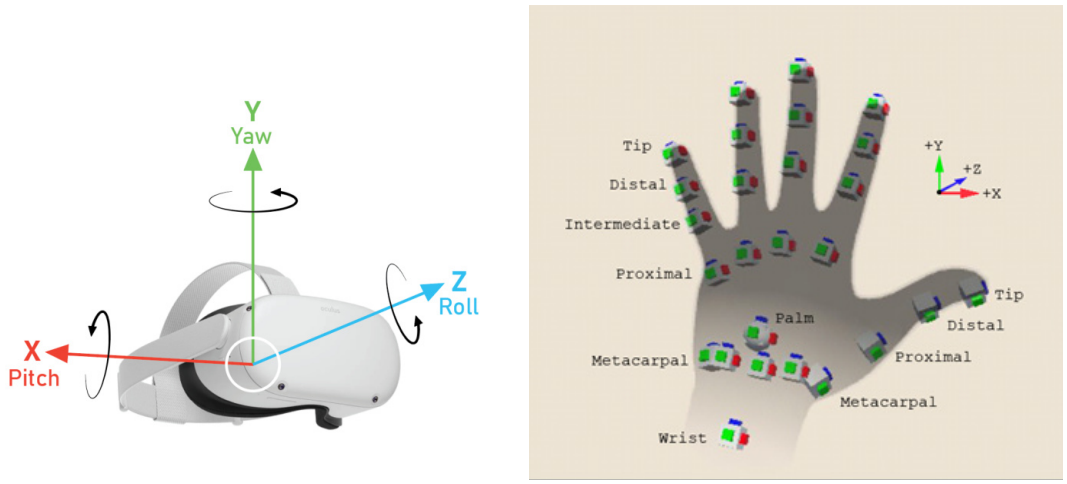


Figure 2.1.: **Left:** 6 Degrees of Freedom of movement tracked by the Quest 2 HMD - tracking position and rotation. **Right:** Unity OpenXR hand tracking and joint orientation

in a singular device allows for portability and ease of use. However, it is heavily reliant on the efficacy of the vision algorithms which poses their own challenges of working with poorly lit or featureless environments such as a solid coloured wall.

Outside-in tracking requires cameras and sensors to be set up externally in the work area that track the orientation of the device. These setups allow high-precision tracking and are independent from environmental and lighting conditions. However, since these require a heavily controlled work environment with correct sensor placements and complex connectivity in addition to very high prices, devices with outside-in tracking are expected to be utilised in professional or enterprise scenarios.

Head and Eye Tracking

Head tracking involves constantly monitoring the orientation of the user's head via the HMD using the concept of Degrees of Freedom (DoF). In a 3-DoF system, the users head rotation pivoted around the neck in 3 axes are calculated, namely the pitch, yaw and roll as show in Figure 2.1. This allows the user turn and tilt their heads in the x,y and z axes.

This immersive experience is enhanced when movement features are added in the 6-DoF system, allowing users to move up or down, left or right and forward or backward. 6-DoF systems create a much wider range of motions a user can experience, increasing the interaction possibilities with the virtual environment.

Furthermore, research has also been done to refine the tracking of eyes (Adhanom et al. 2023; Clay et al. 2019) adding another layer of natural movement and interaction via gaze. This can be utilized for improving user experience especially for the physically impaired, allowing them to highlight and select interface options with their eyes. Or aid in optimising the scene rendering through *foveated rendering* (Meng et al. 2020) which involves dynamically increasing visual fidelity where the eye is focusing and reducing the quality in the peripheral areas, hence reducing workload.

Hand Tracking

Hand tracking (Buckingham 2021) is yet another extension to enhancing immersion and natural movement capabilities by preventing the requirements of hand-held input devices such as keyboard and mouse or controllers. It works similarly to head tracking, allowing hands with multiple joints to rotate with 6-DoF with natural physical constraints. Hand tracking enables further means of interaction with the environment by tapping, grabbing or throwing virtual objects in the space and working with hand poses and gesture recognition. Figure 2.1 shows how Unity OpenXR achieves motion tracking of hands (Technologies 2023). Each hand is rigged with 26 joints from wrist to finger tips that moves with 6 degrees of freedom. Currently, it effectively recognises tap and pinch motions using the thumb and index fingers.

2.4. Gesture and Intent Recognition

Gestures in general are non-verbal forms of communication - through written symbols, facial expressions, hand and body poses and so on. Several human factors are involved such as the orientation, speed and holding times of said gesture, sequence of gestures performed, etc. Gesture recognition refers to the study of such aspects of gestures from the view of computer vision and human-computer interaction and training machines to accurately recognise them. Machine learning techniques and neural networks have enabled such models to be created while multimodal learning in large language models expands the capability to provide context such as pose description and intention prediction from the given gestures.

3. Related Works

This chapter highlights the works and research conducted that are relevant and related to our implementation and methodology. The topics include human-AI interactions, using LLMs in VR in general, LLM based NPCs in VR and their interactions with users and vision and image recognition capabilities of LLMs.

3.1. Human-AI Interaction

In order to test how LLMs interpret human gestures, Wicke (2024) explores their capabilities to decipher gesture cues based on different contextual and cultural backgrounds of people, embedded in text descriptions. A Turning Experiment (a specialised test of LLMs on human behaviour, developed by Aher et al. (2023)) is conducted on the model using a vast dataset of gesture descriptions called Verbal Message List by Matsumoto and Hwang (2013) .

Subsequently, from gesture understanding to assessing intent prediction using LLMs, Ali et al. (2024) proposed a multi-modal approach to improve human-robot collaboration via an object categorisation task where speech, hand gestures and body languages are used. The study incorporate physical implements of interaction using a NICOL semi humanoid robot (Kerzel et al. 2023) with face and arms. For vision and synthesis, Tasks API of MediaPipe (Lugaresi et al. 2019) and its complementary tools are utilitised to identify overall posture, hand gesture and facial expression of the user. This data along with a system prompt, is transformed by the LLM to provide contextual and meaningful responses. The object categorisation task involves the user to isolate few objects to either side of a table and ask the model to categorise the rest accordingly. Upon evaluating five LLMs, GPT-4 performed the best overall with category definition, reasoning and item explanation.

3.2. Using LLMs in VR

This paper by Roberts et al. (2022) highlights the potential of few-shot LLMs in simultaneously contributing to game play and design via content generation during runtime. The experience describes a mutliplayer ping pong game with changing ball and paddle

based on voice commands from the user, powered by Azure Cognitive Speech to Text Microsoft Corporation (n.d.). First, the game objective description and the resulting object context was provided to GPT-3-text-davinci-002 model based on the current ball and paddle objects with a temperature T=0.5. Subsequently, code-davinci-002 model's code generation was used to initialise model download via the Sketchfab API with proper orientation and scaling. Based on search term relevance, the models were selected randomly from the most liked ones with the lowest vertex count.

Song et al. (2024) introduced the LearningverseVR - a game-based learning platform in virtual reality using generative AI. It operates on two core principles of providing learning with generative AI and ensuring immersion in VR. The first describes harnessing the capabilities of generative models to create and animate interactive NPCs and providing a "scriptless interactive design" which allows direct communication between the user and the model without any predefined scripted dialogues. It moreover utilised text to speech and image generation to incorpate in the experience. Primarily using GPT-3.5 combined with a locally run LLM, a long term interaction memory is ensured, enhancing learning experience.

Expanding from virtual reality, Lee et al. (2024) implemented GazePointAR - an Augemented Reality(AR) Voice Assistant (VA) program that utilises LLMs to identify objects and people pointed out by the user. Using Microsoft's HoloLens 2 and Unity's mixed-reality plugin, realtime gaze and hand tracking was achieved. When the user looked and pointed to an object or character, the image was classified using multiple computer vision algorithms such as Google Cloud Vision and Amazon's Celebrity recognition model from which the labels were extracted and categorised from broad to specific item details. This was then used as the context prompt for the LLM (GPT-3) to add to the user's question and provide appropriate responses. A 3 part study was conducted on 12 participants to test on multiple VA systems, surrounding various regular tasks where GazePointAR answered 20 out of 48 queries correctly. The total time for the procedure - from user's question to receiving a response took 7.51 seconds.

3.3. LLM Scene Understanding in VR

In order to test whether LLMs are capable of scene comprehension and spatial understanding from variable Field of Views(FOVs) in VR, the research conducted by Qi et al. (2025) investigates five important aspects: detection accuracy, entity feature description (colour, shape, texture, etc.), spatial understanding, entity cross labelling and consistency over multiple FOVs. For VR Quest 2 was used and a database comprised of 135 compressed images from a VR game with variable FOVs, perspectives and lighting conditions, categorised by visual perception difficulty. The images were provided along

with the context from each aspect as prompts to receive results from the model, which proved with great success to be able to recognise the scene correctly and understand spatial relationships and identify the entity features with great accuracy. This paper proves that with better prompt design, significant improvements can be made - 41.67% to 71.3% correct detection rate from using a basic prompt to a more well defined one. However, when asked to label objects from training sets to locally in VR, it performed poorly (all incorrect labellings), therefore, being sufficient at identifying the features but bad at labelling over the same image.

Z. Wang et al. (2023) introduced Chat-3D, a dialogue-based interaction method focusing on 3D scene understanding via a pre-trained LLM. The scene images are captured, segmented to objects using a separate segmentation model and embedded with object properties and correlations which are used to train the LLM in three stages: 1. Object alignment of 3d mesh with word embeddings of pre-trained LLM. Computing the similarity between both allows better learning of projectors (object property learning and word embedding). 2. Scene alignment of the entire scene with the LLM. Can also utilise 2D scenes to learn spatial relationships between segmented objects and 3. Instruction tuning based on detailed relevant data. From the user end, one can select an object from the scene and ask questions regarding its visual and spatial properties in the scene which the LLM (GPT-4 in this case) takes as context and answers accordingly to the training. Consequently, comparing with similar works done using two stage training schemes, Chat-3D performed significantly better (75.6 overall evaluation score with an increase of 8.6).

Another research regarding scene understanding in VR is the VR-GPT framework by Konenkov et al. (2024). It is a custom made Visual Language Model (VLM) based on Qwen-VL, with two scenes - a kitchen and a laboratory - created in Unity and Quest 2 with progression event trigger systems for simple 'pick-and-place' tasks. The users are also able to conversate with the LLM via ESPNet TTS and Whisper ASR. The model was one-shot trained using a dataset containing images from the scenes and task descriptions. The user study consisted of the two scenes with 6 pick-and-place tasks in randomised order and a 9 question survey on a 5-point Likert scale and compared to baseline, no visual nstructions were provided and rather the model guided the users to the correct tasks. The outcome showed that compared to the baseline, VR-GPT had lower mean task execution time and performed significantly better (3 times) in both scenarios and for both trained and untrained tasks.

3.4. LLM Gesture Understanding

Pang et al. (2024) implemented the LLMGesticulator, which aims to enhance the natural human interaction and immersion with NPCs using co-speech recognition methods from user generated audio clips and text prompts to synchronise with the contextual natural motions of the character models. The QWen 1.5((Qwen Team 2024)) LLM was used with 7B parameters with (and without) training with transcriptions from a motion video dataset describing body and hand gestures. The animation of the characters models were supported by a motion capture dataset. A test on 30 pairs of videos (from the authors and other comparative works) was conducted that resulted in various scores that outperformed the comparison while aligning closer to the ground truth. Additionally, a study was conducted on 13 participants which included watching two 10-second videos to be scored based on human likeness and audio-animation comprehension. As a result, it scored between 60-70% in those metrics compared to others (30-40%).

Moreover, the GestureGPT framework presented by Zeng et al. (2024) evaluates and provides the first zero-shot gesture understanding solution using GPT-4. A 3 step procedure is implemented on two scenarios - smart home control and video streaming. Firstly, the Gesture description agent breaks down video frames to a matrix of actions and describes them in natural language. The Gesture inference agent then gathers the descriptions and with a certain confidence level either applies a function or retrieves relevant context from the Context Management Agent - a library of context relating to the current environment. For each scenario an experiment was conducted on 16 participants with 8 gesture related task and through two expert questionnaires scored - on a 5-point Likert scale - scored positively with 3.28 and 3.74 mean (1.41 and 0.73 SD respectively).

Further research into LLM gesture understanding was done by Kobzarev et al. (2025) with GestLLM, a model based on GPT-4 and using MediaPipe for hand gesture feature extraction. In a three step process , the features are first extracted, then further description is provided to each feature such as curvature and distance between fingers, and finally the context is translated to machine-readable instructions for the collaborative robot arm. Two tests are run to evaluate the performance: Firstly, a dataset is created with hundreds of images of four uncommonly recognised hand gestures with varying lighting conditions and distances to the camera. These images are then zero-shot tested on both GestLLM and GPT-4o and GestLLM outperforms in both close (0.7m) and long distances, upto 4 meters. The other test involved controlling the robot arm using basic movement commands using gestures or gamepad controls and through a NASA-TLX study (Hart and Staveland 1988) discovered that, while using gestures required more physical and mental effort, it performed better with lower frustration

and effort needed.

The potential of LLM's gesture understanding was also expanded to the military domain by Naidu et al. (2025). Using GPT-4o and its vision capabilities to accept image as input, and Microsoft Azure Kinect (Microsoft 2019) for recording video gestures, the gesture recognition task was performed on the US military hand signals. As it is a particular set of gesture data, the domain-specific knowledge was established using Retrieval Augmented Generation (RAG) by providing the military gesture documentation to the model. Additionally, Chain-of-Thought (CoT) prompting was used to reinforce contextual knowledge regarding how and when these gestures were used. Using the recordings from the Kinect, every 15th frame was provided to the model for each of the 7 selected gestures which was then labeled and encoded to machine readable commands that the collaborative robot Stretch 3 mobile robot (Inc. 2024) performed accordingly. From a study on 4 participants performing all 7 gestures, the model scored an overall 80% gesture detection accuracy with an 89.9% F1 score. However it reported high inference delays from 7.88 to 18.3 seconds on average depending on the gesture frames.

3.5. LLM-Based NPCs in VR

This research by Maslych et al. (2025) conducts a pilot study in VR using LLM powered human avatars with various feedback types: state lights objects in scene to show the current action state of the LLM, thinking state using a loading bar to show the LLM's progress in processing information (similar to our study where the thinking time of gesture recognition is masked by a specific dialogue and animation), a chat box UI which shows the user question and LLM's elaborate response. Additionally an avatar type with no feedback was also observed. The process involves using Unity's Whisper(Radford, Gao, et al. 2022) Automatic Speech Recognition (ASR) which converts user's speech to text and adds to the LLM message history pool and examines the current event sequence of the VR experience using their state machine-based behaviour transition check. Based on how the query is set in the prior process, the LLM provides a response which is then integrated with a Text-To-Speech (TTS) converter that the avatars use to speak using Unity's AudioPlayer, further enahnced with OVR Lip Sync. Consequently, the study reported a low avatar realism score (3.12 out of 7), one factor being lack of animated responses, in addition to an avatar responsiveness score of 4.38 out of 7 due to the high response time of 3.2 seconds. Aside from the TTS and types of avatar used, the process is similar to our approach. However, leveraging animated avatars in our study resulted a significantly higher realism score from various measures in our survey. Moreover, while a TTS feature is highly desirable for added immersion,

the lack of it in contributed to a lower response time in the Speech-To-Text (STT) segment of our study (1.34 seconds), but with a significantly higher and therefore undesirable average response time for image recognition (6.8 seconds).

Li et al. (2025) explore the efficacy of LLM based NPCs in VR in multiple contexts and scenarios with a strong focus on the model (GPT-4-Turbo) being able to learn from the environment and create dynamic interactions feasible with the user input. A scene schema is created to describe the scenario containing the following parameters: A context to describe the structure of the schema, the scenario rules and a list of accessible functions, Objects and Characters schema contains metadata of objects and NPCs in the scene - postions, item names, static or grabbable, etc. In addition, a Spots schema to describe for interaction zones and finally Communications, keep track of player and model interactions which included pointing, grabbing objects, controller inputs and voice interactions using OpenAI. Five scenarios were created with their unique schema and tested on 14 participants in VR using Meta Rift S and the model performed impressively with 80 to 100% context awareness success rate for different interactions such as touching and pointing to objects. However, a common limitation with high response time was also observed here as the mean response time ranged from 3.51 seconds (scenario 1) to 4.77 seconds (scenario 3).

Broadening the scope for LLM based learning, Pan et al. (2024) created an NPC agent, ELLMA-T to support learning the English language in the interactive VR platform, VRChat (VRChat Inc. n.d.). Based on GPT-4, ELLMA-T is instructed to provide contextual responses based on user's backgrounds and estimate their language level based on speech recognition powered by Whisper. Furthermore in the tutorial phase, it carries out multiple tasks to improve the learning experience such as role playing user- or pre-defined scenarios from the current VRChat virtual world, grammar correction, providing examples, etc. using Whisper ASR, it detects user's emotions via keyword detection and mirrors it with its own body language and facial expressions which are powered by the VRChatOSC tool(VRChatOSC Contributors n.d.) and for the voice output uses OpenAI TTS. This (15 minute) experience was tested on 12 participants using Meta Quest 3, resulting in positive responses from the 30 minutes semi structured interview. The users responded that the NPC was 'human-like' and 'believable', successfully proving LLM's to capability to mimic anthropomorphous conversations. The limitations of slow response times were present here too as users mentioned breaking of conversation flow when they noticed delays in response from the model or when they could not interject.

Similarly, Wan et al. (2024) utilise VRChat and GPT-4 to create diverse NPCs interaction options for VR which is constructed using four key modules. Firstly a database of various NPCs is created by assigning base system prompts containing the personality description of each NPC. These NPCs also accumulate memory objects from the context

and user interactions for persistent recall for future conversations as observations. Based on the top N observations, on further interaction with the user (using text or voice), metrics such as context relevance and recency (observations gathered in the most recent interaction) are tested. In total, 3 tests are conducted with 7 conditions alternating between two potential response sets. These tests are then judged by both Human and LLM judges using Mistral-7b, Llama-2-13b and GPT-4 and observed that GPT-4 scored the highest average precision in the NPC responses. This study is yet to be evaluated from a player's perspective and observe overall performance and yet again is limited by the high response time due to the STT API.

The use of LLM driven NPCs in educational scenarios was also implemented by Triyono et al. (2024) which presented a familiar classroom environment with multiple NPCs acting as instructors or fellow classmates. The aim of the study (similar to ours) was to - via experiments - evaluate the model's response time, accuracy and consistency. Further surveys and interviews were conducted to assess experience realism, responsiveness and overall user satisfaction. The task involved simple interactions with the NPCs using voice input where the interactions could be triggered by either party. The experiment, conducted on 30 participants resulted in 200 distinct interactions with high 87% accuracy , an average response time of 1.2 seconds (SD=0.3 - similar to ours of 1.34 seconds) and a 4.2 out of 5 overall user satisfaction score.

To further study human behaviour when interacting with LLM based NPCs in VR, Lim et al. (2025) presented a scenario where such NPCs (powered by GPT-4) were used to assess certain behavioral impacts of interacting with gender matched users. Users' gaze interaction and proclivity to discuss health related topics and activities were evaluated when they interacted with same gendered versus opposite-gendered NPCs. Using Meta Quest Pro VR headset and Azure STT, the users engage with the NPCs via voice chat and receive voice responses. Several studies on agent's likability, immersion and social co-presence were conducted on 60 participants where the overall scores significantly favoured the use of VR compared to text based agents. One consistent limitation is no exception in this case as the authors reported a waiting time between 5 to 7 seconds for LLM responses for the entire process which is undesirable.

Furthermore, some tools were also created in order to streamline the human-NPC interaction process in virtual environments. Firstly, CUIfy by Buldu et al. (2025) - a generic Unity plugin that uses LLM to power voice conversations with multiple NPCs in an XR environment. It supports multiple LLM, STT and TTS models which can be instanced inside Unity to each game object (NPC in this case) with a custom system prompt that will assume the role accordingly. The plugin is modular and runs locally which ensures lower latency and flexibility of use at the cost of computational load and overall response quality.

Similarly, Shoa and Friedman (2025) presents an open source system to allow virtual

NPC interactions, called Milo. It is also a voice conversation based system, supporting multiple LLMs and TTS models (uses GPT-3 as the base) and Google API for voice activity detection. Moreover it includes a control interface to inspect model responses and user speech transcriptions and allows overrides in case of errors. The NPC reacts using simple animations like talking, listening and idle animations and changes gaze to look at manually assigned gaze points for added immersion. It performs in two modes, a dialogue chat mode where it listens and responds when it detects silence and a multi-party conversation assist mode where it responds only upon user prompt. However, it only supports one NPC per scene and similarly suffers from response latency and anomalous model response quality.

Table 3.1.: A Literature overview highlighting comparative metrics (STT - Speech To Text, IR - Image Recognition)

Paper(Year)	Model(s)	Input Type(s)	Accuracy	Response Time(s)	Study Participants
LLM in VR					
Roberts et al. (2022)	GPT-3-text-davinci-002	STT	-	-	-
Song et al. (2024)	GPT-3.5	STT,Text	-	-	-
Lee et al. (2024)	GPT-3	STT,IR,Gaze	42%	7.51	12
LLM Scene Understanding					
Qi et al. (2025)	GPT-4o	IR	71.3%	-	-
Z. Wang et al. (2023)	GPT-4	Text	75.6 Eval score	-	-
Konenkov et al. (2024)	Custom from Qwen-VL	STT	Variable (48 to 78%)	-	-
LLM Gesture Understanding					
Pang et al. (2024)	QWen-1.5 7B	Audio, Text	Variable (60-70%)	-	13
Zeng et al. (2024)	GPT-4	IR	-	-	16
Kobzarev et al. (2025)	Custom from GPT-4	IR	-	-	-
Naidu et al. (2025)	GPT-4o	IR	80%	Variable 7.88 - 18.3	4
LLM NPC in VR					
Maslych et al. (2025)	GPT-4	STT	-	3.2	-
Li et al. (2025)	GPT-4-Turbo	STT	Variable (80-100%)	Variable (3.51 to 4.77)	14
Pan et al. (2024)	GPT-4	STT	-	Not Specified	12
Wan et al. (2024)	GPT-4	STT	-	-	Not Conducted
Triyono et al. (2024)	Not Specified	STT	87%	1.2	30
Lim et al. (2025)	GPT-4	STT	Variable	5-7	60
Shoa and Friedman (2025)	GPT-3	STT	-	Not Specified	Variable
Ours	GPT-4-Turbo(4o)	IR	73.6%	6.8	12

From the table and mentioned literature above, it can be inferred that the focus on image and gesture recognition using foundation LLM models in VR are under-researched or rather an up-and-coming topic. Especially related to direct user interactions with the model disguised as NPCs that perform image recognition with no pre-processing, training or computer vision systems involved. Also, as mentioned by Brito et al. (2025) in a recent study, despite the potential of such models, the interfaces are "confined to chat-based interfaces" and are yet to be fully utilised by being incorporated into VR environments via human-like interactions with NPCs (or "Digital Humans"). The paper

also mentions current challenges such as computational load and more prominently the response latency that has been the common limitation for many comparative works.

To address this gap, we implemented our VR program that uses zero-shot trained GPT-4 model to assess whether the model can provide natural responses (with accuracy and time) to the user's gesture using its intrinsic image recognition capabilities . A study was conducted on 12 participants, each required to play through a gesture recognition based VR session and another control STT-based session with alternating sessions for each participant to counter balance any sequence bias. Their experiences were scored based on various questionnaires regarding spatial awareness, perception of the AI and sense of agency.

4. Methodology

4.1. Overview

The experiment tests the efficacy of using ubiquitous and readily available LLM and their various communication methods in VR based applications and whether they can replace otherwise hard-coded traditional means of providing communication features in such scenarios. The experiment focuses on the image recognition capabilities of LLMs to evaluate interactions with the players and compares it to a more accessible and common means of communication via speech to text. Henceforth the participant or the human user shall be identified as the ***player*** and the AI or LLM-based ***Non-Player Character (NPC)***. As mentioned above, the aim is to investigate the following RQs or research questions:

RQ1 *How accurately can the model identify the user's gesture?* The primary metric of evaluating such models is their accuracy to perform the tasks give the capability. To enhance believability of interactions in the scene, the model must respond to the user's input - hand gestures - accurately and react accordingly. It is hypothesised that the GPT-4o model will prove to be proficient in handling such image recognition tasks, even in non-trivial, crowded scenes.

RQ2 *How quickly can the model read the user's gesture? Can the interactions occur in real time?* In order to determine whether such ubiquitous LLMs can be used in more VR based applications and moreover, whether unconventional means such as hand gestures can be effectively used for communication, the time taken for the NPC to detect, read and respond to the image is a crucial factor. The current hypothesis is that the models are fast enough to respond in a convincing period of time where the player neither has to wait too long, nor does it feel unnatural to them during the interaction.

RQ3 *How does the model impact the **user experience** in a given scenario?* Finally, the research overall attempts to determine, positively - via the scenarios presented and the information gained - whether the use of AI embodied as NPCs provide a rich and diverse user experience in any given scenario and to find potential use cases where such LLM agents could be incorporated.

This section goes into detail regarding the two evaluation methods used for the experiment:

Gesture Recognition: The players interact with the NPCs using hand gestures. This is limited to the VR based hands, therefore the wrist and finger joints. The NPCs are required to ask certain questions that the players must answer with the correct one or two handed gestures in order to complete the objective and proceed to the next one.

Speech To Text: The scenario remains identical with a different input method where the player is allowed to conduct spoken conversations with the NPCs. Depending on the defined control parameters - number of interactions or elapsed time, etc. - the narrative progresses and the player attempts to complete the next objective.

This section will further elaborate how these scenarios have been created separately in order for the player to experience each and then complete a set of questionnaires for a pilot study, related to the experience in general, experience with the NPCs and regarding immersion and spatial presence. In addition, the following section also highlights the use of all the required tools and technologies in order to successfully conduct the experiment. Briefly listed are the following:

1. **Unity** as the base game engine.
2. Open AI's **GPT-4-turbo** as the base model for image recognition and chat completion.
3. Meta Oculus **Quest 2** as the VR headset and,
4. **Whisper**, a Unity Plug-in for locally running automatic speech recognition and speech to text conversion in runtime.

4.2. Implementation

4.2.1. Level Design

While the goal is to learn the behaviour of interactivity between and NPCs and the players, in order to immerse them in a relatable surrounding and create a sense of spatial presence, a typical scenario was modelled. This scenario depicts both the outside and interior of a simple supermarket where the player meets multiple NPCs to interact with while being able to move freely and pick up items from the shop and add them to a cart.

Figure 4.1.: Overview of the Inside and Outside Scenes

Environment Setup

The mesh assets (Figures 4.2 and 4.1)were first modelled in Blender (Blender Foundation 2025), a free and open source application supporting many 3D and 2D design features. This included the exterior road and side walk, a building, the shop interior and all the meshes within. Additionally, for expediency a tree model was obtained from Free3D (2025) but was heavily modified. All the created meshes were then UV mapped in Blender to prepare for texturing. For the supermarket item labels and signage, Illustrator (Adobe Inc. 2025a) was used and Substance 3D Painter (Adobe Inc. 2025b) was used to texture the base materials with the labels. A total 72 different models of which 41 were supermarket item meshes with 4 different texture sets applied to them, and incorporating 65 uniquely designed labels in order to create a larger variety and added realism to the shopping scene. These additions to the scene are not only for visual fidelity, but rather play an important role setting up the quest design and adding variety to the gameplay (See Quest Design). To set up the scene, Unity's URP or Universal Render Pipeline was used to create a light-weight, performance focused VR application. To set up the lighting, Unity's default lights and reflection probes were used along with one of the skybox or HDRI textures from Unity assets store's Skybox Series Free and were baked into a light map to further improve performance. Finally, Unity's particle system was used for the falling leaves in the outdoors scene and to indicate one NPC's audio playback.

Characters

During the initial phase of the quest design, a simpler floating, robotic avatar was created with the aim of interacting with the player and guiding them through the

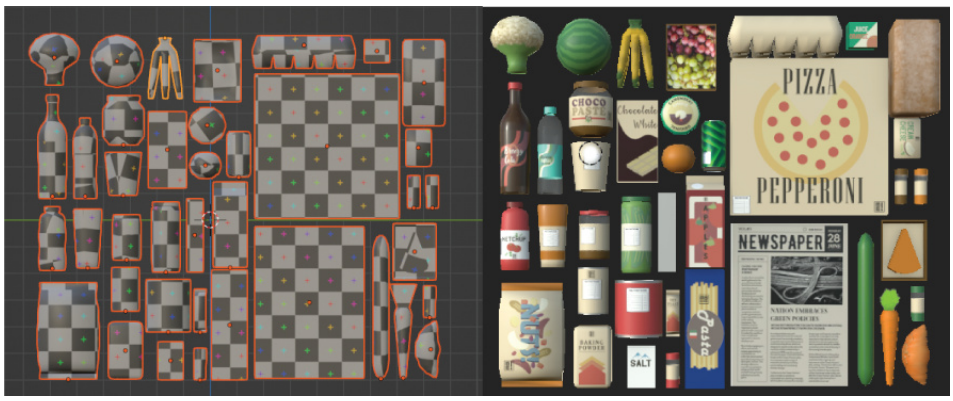


Figure 4.2.: Shop Items Models and Textures

objectives in the experience. However it was determined that choosing the right representation of characters in a realistic environment was vital to enhance immersion and for the study of perceived realism via AI based NPCs - not only in visuals but also how they sounded and behaved. Unlike static meshes created for the environment, dynamic character assets are significantly challenging to produce as they require cleaner mesh topology, one that must be perfected for rigging the mesh with skeleton joints. Furthermore these joints must be go through a process called *skinning*, where parts of the mesh gets assigned weights in order to be sufficiently flexible while animating them which is yet another obstacle.

To solve characters, Mixamo (Adobe Inc. 2025c) was used - a free online tool that provides ready-to-use, textured and rigged 3D characters with many pre-made animations to choose from. Improving from the initial iterations, 4 characters were planned to be implemented rather than 1. This would allow the players more opportunities for interacting with the NPCs, enriching the study on their behaviour in addition to providing a more natural feeling environment. The characters taken from Mixamo were of 'Megan', 'Kate', 'Remy' and 'Leonard', who will be referred to further in the Quest Design section.

Moreover, these NPC were partially voiced using ElevenLabs ((ElevenLabs Inc. 2025)) - an online tool specialising in generating audio via speech to text, conversational AI and sound effects such as voice changers, among others. These were used in places where some basic response were hard-coded. The four mentioned NPCs utilised four separate voices from ElevenLabs: 'Jessica' for Megan, 'Cassidy' for Kate, 'Arnold' for Remy and 'Antoni' for Leonard. Each voice agent can be controlled via 4 parameters: speed, stability, similarity and exaggeration. For our purposes, these parameters were left default (medium similarity, no exaggeration, slightly variable speed and medium

stability), however each voice line required three to four generations on average to achieve the ideal cadence and intonation. A total of 38 voice line audio files were generated for the four NPCs.

4.2.2. User Interface Design

To preserve a natural experience and as to not bloat the limited field of view in VR, a minimalist approach was considered when designing the UI and controls.

Interface

The user interface can be categorised into three parts: the Heads-Up Display (HUD), tutorial and objects and events interfaces. Unity refers to these interfaces as canvas and the terms will be used interchangeably in this section. All these canvases are bound to the Head-Mounted Display (HMD) camera which allows displaying independent of player's movement or view. The first is the main canvas or the HUD interface, which includes all the interactivity related elements with the NPCs: NPC name and dialogue panels, current objective and button controls panels. The tutorial panels consists of short video demonstrations with descriptions on how to carry out tasks such as picking up objects and using hint events (For hint events, see Quest Design). Finally, the objects and events interfaces include displaying the shop entrance graphics, the grocery list as an objective and handling the end game event. Furthermore, included in the HUD UI are elements specific to the hand gesture based experience: a gesture recognition zone where the user can place one or two hands (where prompted) to do gestures as well as a time indicator to hold to generate the image. The voice based system however, consists of a audio recording indicator where the player can hold a button and let go to send a voice message (see Controls). The icons for these indicators were created in Adobe Illustrator 2020.

All these UI elements are displayed dynamically based on events in order to keep the view as clean as possible and retain maximum focus from the player: The main canvas is dynamically visible based on the proximity to an NPC. The tutorial interface and the main UI's grab instructions appears once the player enters the shop and each tutorial video loops once before allowing the player to proceed in order to ensure acknowledgment. Similarly the objects and events interfaces appear according to their corresponding time and relevance. The Quest Design section highlights the appearance of these UI elements.

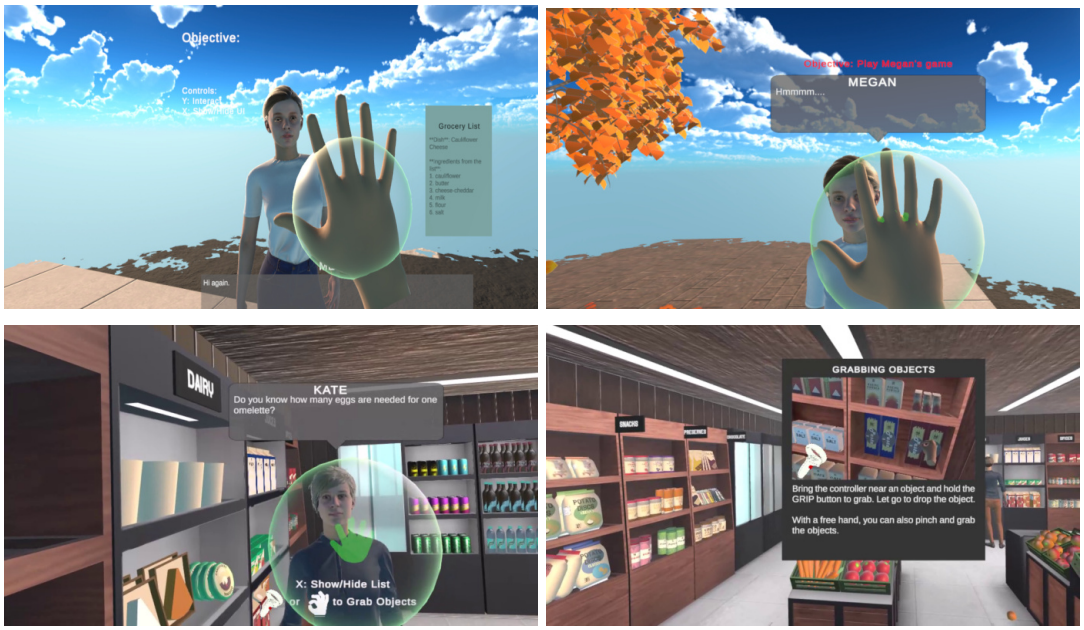


Figure 4.3.: UI Design Evolution: Top row: **Left:** Old UI, **Right and Bottom row Left:** New UI ; Bottom row Right: Tutorial Tooltip

UI Design Evolution

With more user playtests and feedback, the UI improved drastically in terms of consuming less active screenspace and overall visuals (Figure 4.3). Now, users are allowed to hide or unhide certain UI elements such as the grocery list page and objective text, the speech bubble was repositioned from the bottom of the screen to over the NPC's heads and is automatically disabled when the user is outside the proximity of any NPC. This not only allows the elements to blend naturally and intuitively but also prevent visual strain for the user when looking around. Other elements such as the controls and tutorial help texts are displayed during relevant triggering of events and are embedded to the interaction prompt rather than a separate list on the side of the view. The vital factor to the design change was ensuring the user's focus remain at the centre of the view which inturn reduced visual clutter and distractions and gradual strain. This also allowed overcoming the limitations the Quest 2 sets by locking the field of view in Unity. Finally, statistical information such as even timings and image processing progress were removed from the view and added as logs.

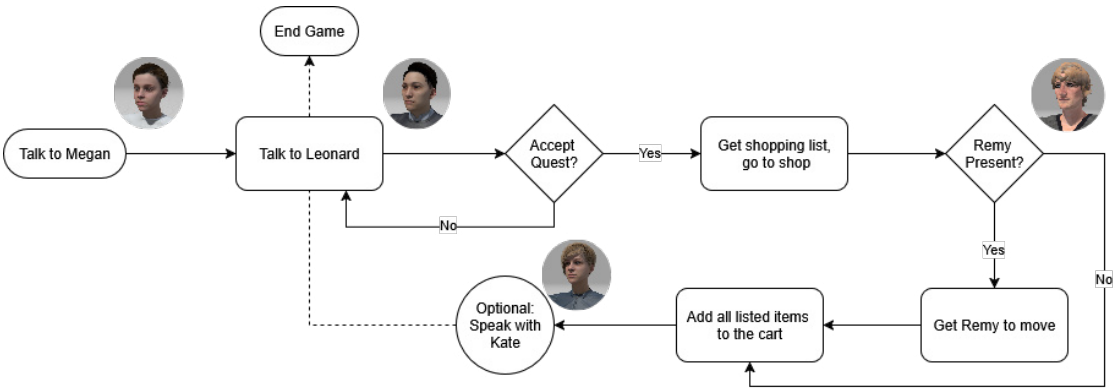


Figure 4.4.: Simplified Quest Flow of the VR Experience

Controls

Depending on the input type of the experience (hand gestures or voice based), the player uses one or both the left and right controllers of the Quest 2. However, the experience allows them to simultaneously use both hands and controllers. Some controls common to both versions include the player being able to initiate important actions with the Y button on the left controller such as speaking with the NPC's, entering or leaving the supermarket and using the X button to show or hide elements of the UI. Moreover, they can move around freely in the environment using the left joystick. Specific to the voice based interaction is holding the A button on the right controller to speak to the Quest 2's microphone while the left trigger can be particularly used when hint events are triggered . Finally, with either pinching their hands or using the grip buttons on either controllers, the player can grab objects such as items and carts in the shop.

4.2.3. Quest Design

The quest is therefore designed accordingly in order to keep the players engaged with both activities of NPC interaction and shopping (Figure 4.4). Further measures were taken to create a sense of continuity in the two separate experiences each player would participate in.

```
<HandPoses>
  <Pose>
    <Timestamp>2024-09-26T16:59:18.6200624Z</Timestamp>
    <JointData>
      <Joint ID="Wrist">
        <Position X="0.200" Y="1.118" Z="0.500" />
        <Rotation X="-0.500" Y="0.000" Z="0.000" W="0.866" />
      </Joint>
      <Joint ID="IndexMetacarpal">
        <Position X="0.180" Y="1.144" Z="0.526" />
        <Rotation X="-0.500" Y="0.000" Z="0.000" W="0.866" />
      </Joint>
      <Joint ID="IndexProximal">
      <Joint ID="IndexIntermediate">
```

Figure 4.5.: Output Segment From the Joint Coordinate Recording Approach

4.3. System Development and Implementation

4.3.1. Early Gesture Recognition Approaches

in the early development phase of the gesture recognition module, a more rudimentary approach was taken. Unity's XR hand tracking plugin combined with the Quest 2 allows for the detection of 26 distinct joints (Figure 2.1, from the wrist to each of the fingertips. Each joint stores the coordinates that represents the position and rotation of said joints (Figure 4.5) . Multiple input methods were considered, for instance, upon setting the right gesture, the user would - using a held timer or a button press - capture these joint positions. An initial set of prompts would firstly, define the default 'forward palm with fingers stretched' pose using the corresponding coordinate set, followed by the instructions to identify changes in coordinates to indicate joint movement and rotation and hence the gesture.

However, this led to drastic increase in prompt size, exponentially expandeding the chat history for the API, leading to 'InvalidRequestErrors' from exceeding request token limits. Furthermore, using joint coordinates as prompts exposes the nature of how such large language models are trained - focusing on natural language reflects their inaccuracies in detecting hand gestures using lengthy lists of raw joint coordinates. This is especially augmented without any set origin of the hands as they must be free to control the users, forcing variability. The following tables (4.6) highlight the results of a simple pinch pose recognition followed by a variation of different poses tested on GPT-3.5, GPT-4, GPT-4o, GPT-4o-mini and Google's Gemini using the joint coordinate system. An alternative solution determined to be more effective - feeding converted and optimised images as prompts which is explained in subsection **B.Gesture Recognition** - *TakeScreenshot and SendMessage*.

Pinch Test 1 -3.5		Pinch Test 2 – 3.5		Pinch Test 3 – 3.5	
TRUE	PRED	TRUE	PRED	TRUE	PRED
No	No	Yes	Yes	No	No
No	No	No	No	Yes	No
Yes	Yes	No	Yes	No	No
No	Yes	Yes	Yes	No	Yes
Yes	Yes	Yes	Yes	No	Yes

Pinch Test 3 – 4o		Pinch Test 3 – 4o mini		Pinch Test 3 - Gemini	
TRUE	PRED	TRUE	PRED	TRUE	PRED
No	Yes	No	Yes	No	Yes
Yes	No	Yes	No	Yes	No
No	No	No	Yes	No	Yes
No	No	No	Yes	No	Yes
No	No	No	Yes	No	No

	Pose Test 1 – 3.5	Pose Test – 4o mini	Pose Test - Gemini
TRUE	PRED	PRED	PRED
Resting	Fingers curled	Thumbs up	Resting
Fist	Pointing	Fist	Fist
Thumbs up	Thumbs up	Resting/open hand	Fist
Victory	Ok	Pinching	Resting/Open hand
Rock	victory	Open hand	Open hand
OK	pointing	Open hand	victory

Figure 4.6.: Early Approach Image Recognition Results from querying joint coordinates to multiple LLMs

4.3.2. Speech To Text Implementation as a Baseline

To evaluate the efficacy of the image recognition input method, a legacy method was also implemented using the same scene setup. Many programs, VR or otherwise, utilise either direct text input or use speech to text technology when incorporating LLMs as a means of interacting with the users. For our comparison, speech to text was preferred which is effective as a natural medium of interaction and prevents the tedium and lack of immersion of typing on a virtual keyboard. To achieve this, OpenAI's Whisper (Radford, Gao, et al. 2022) was used which is an Automatic Speech Recognition (ASR) application that leverages enormous lengths of training data capacity - 680,000 hours of labelled multitask, multilingual audio data - which allows for better quality text conversions without any pre-training or fine-tuning. Additionally, it the dataset covers 117,000 hours of multilingual audio spanning 96 languages which can then be translated to English via training on 125,000 hours of translation audio data. To incorporate Whisper to Unity, an open source, instance binding application was used via this repository (OpenAI 2022) that allows Whisper to run locally, improving conversion rates. Utilising its features, a speech to text script was written that sets the

Figure 4.7.: Interction with NPC Megan in STT Session

language and translation settings, listen for button presses and record and transcribe the audio to a text log output. Other settings such as *max length seconds* and audio *frequency* were set to 5 seconds and 16,000Hz respectively to avoid recording long input texts to feed to the LLM and keep the conversations natural.

4.3.3. Implementation of Modules: Core Functions

A. General

1. *Interact with NPC:* This function keeps track via Unity's collision system, which particular NPC the player is currently in range of. This is crucial information as it allows dynamic dialogue, audio and animations to play along with progressing in the event sequence. Each NPC is identified using their respective game object name, in addition to a specific character parameter for the different actions each NPC can carry out based on the API's response (see Evaluate Response). In the speech to text version this additionally includes system messages to define the character in proximity to provide interesting dialogue options.

2. *Evaluate Response:* After the message is sent to the API, its response is parsed in the evaluate response function. When querying the API for certain actions, it is asked to respond with distinct characters which are used to trigger events. Therefore the response string must be processed. First it is trimmed of all spaces and new lines and according to the queries, the first character or word is evaluated with an array of expected outcomes. In case of a match, the *InteractWithNPC* method is called with the currently colliding NPC and the matched character or word. Moreover, processing of correct responses are also carried out by this function. In the first segment of answering gesture related questions, for each correct or incorrect answer, the count is kept track of with additional elements including playing voice lines, queries to the API with further dialogue options or

Figure 4.8.: Interction with NPC Megan in Gesture Recognition Session

invoking quest events.

A simpler and more flexible handling of responses was implemented for the case of speech to text. Each response is trimmed to get the leading character as mentioned in the instructions to the LLM and *InteractWithNPC* function is called on it. One particular NPC case is handled here with an interaction limit to avoid endless conversations.

B. Gesture recognition

Take Screenshot and Send Message: The *TakeScreenshot()* and *SendMessage()* functions work in conjunction to create images with timestamps, get past chat history, and prepare request packages based on string or image-based input. If an image is passed, indicating that the user has requested for image recognition via holding a pose, it is first broken down into an array of bytes which is then converted to a base 64 string. Naturally, these strings are enormous and in order to prevent large requests packets, these are not added to the chat history and rather only the responses from the LLM to the images are preserved. Additionally, a helper query is passed with the image to provide more context and receive curated responses. This is determined by tree of conditions based on the current state of the game and adjusts the query sent to the model accordingly (See Prompting).

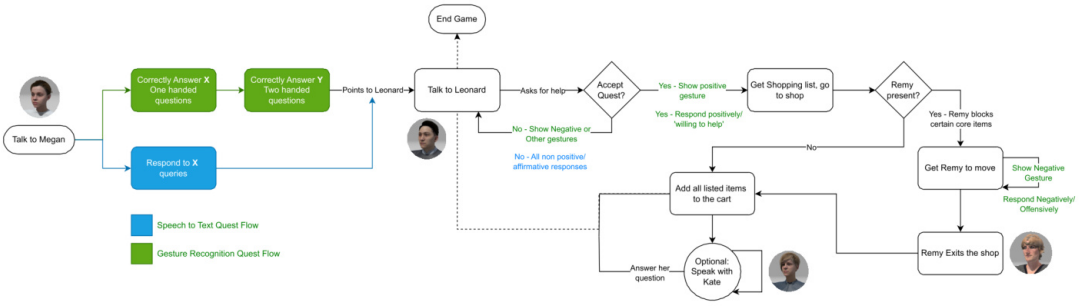


Figure 4.9.: A detailed quest flow that highlights all the actions a user must take in order to complete the experience

The base-64 string (when images are sent) and the condition-dependent query are incorporated to a standardised message format for the API and the data is packaged into a serialised JSON request with the GPT model name, number of maximum response tokens (words broken down as tokens by the API) and message temperature (allowed level of creativity in responses) and sent to the API. In the frontend, the screenshot is taken when the player holds a pose within a defined boundary for a specified period of time (see User Interface Design: Interface). To reduce overhead, the screenshots are taken in a 1280x720 HD resolution with reported 'Medium' visual complexity, which refers to and typically includes a VR environment, a human avatar, a hand gesture, text and UI elements, and diverse foreground and background objects. Lastly, the response generated by the API is then handled by the *EvaluateResponse()* function.

C. Speech to text: *Send Voice Message*

For the speech to text segment the messages are less restricted in terms of content. Users are completely free to chat with the NPCs regarding any topic. However to drive the narrative and direct the users through the experience, certain conditions are placed (Figure 4.9). Each main NPC takes a certain number of user questions then forces the player to work on the next objective in order to complete the experience in a desirable time. The non essential NPCs can be talked to openly without time limits before the final objective is completed. However, to ensure immersion, simple context messages provides the NPCs more personality (Figure 4.10) via speaking around the theme of the exercise environment and to prevent the users to dive into completely irrelevant topics and make the characters seem like mere answering machines.

NPC	systemPrompt =
	"You are Megan, a 26 YO female who is polite and friendly. You met the user outside a shop and are having a casual conversation. Make up the rest of the personal details yourself."
	"You are Leonard, a 28 YO male who is kind and energetic but is now exhausted from carrying a lot of shopping. Make up the rest of the personal details yourself."
	"You are Kate, a 30 YO female who is extroverted and a bit sassy. She is currently in the grocery store looking for stuff. Make up the rest of the personal details yourself"
	systemPrompt += "**NEVER REVEAL THAT YOU ARE AN AI, ALWAYS STAY IN CHARACTER**"

Figure 4.10.: System prompts created for each NPC

4.3.4. Implementation of Modules: Classes

1. **Chat Memory and API Manager:** A simple script that stores the conversation history in a list and after a certain max value discards it. It also discards the long image conversion strings to avoid large request being sent each time. Allows for fetching and clearing the chat history which is utilised to clear history from singular response cases like the grocery list request. The API Manager links to the OpenAI API endpoint with chat completions which supports images as chat.

2. **Item Collider Checker and Disabler:** The checker contains a list of object names which it derives when said object collides with the primary object - the cart in this case. It checks for duplicates and accepts only unique object names and hence the naming is vital for how it functions. This is explicitly handled both in the Unity editor and as a strict instruction to the LLM to list down the exact name provided. Each frame it first checks whether the grocery list has been updated with a response from the LLM and then evaluate that string with the list of items. If an item name matches the word in the list then it's stricken off (Figure 4.11). For the image recognition experience, additionally an item disabler is prepared as a cautionary step to prevent the user from interacting with some vital objects until a certain event has occurred. This ensures coherence in the intended experience. This is simply achieved via identifying the target objects' parents, and disabling the colliders and physics interactions of their corresponding child objects.

Figure 4.11.: The Grocery List - an important UI element to help players complete the shopping objective

3. **NPC Animations and Voice Manager:** The NPC animations manager is of particular importance as it returns the currently targeted NPC's properties in the proximity of the user. Moreover both the Animations and Voice manager scripts hold an array of predefined animations and voice audio files and are played upon calling by the respective animation function or audio clip index. To create dynamic animations based on time, a function is created that allows for pausing and resuming after a specified number of frames of the target animation sequence. Furthermore, certain animations like 'standUp' is handled separately so that the NPCs follow or look at the player's direction after the one animation completes.

4. **Player Movement and tracking:** From the different movement options applicable in a VR scene (nav mesh or raycast teleportation, free walking), the traditional way of using controller joysticks to move around seemed to be the more intuitive, natural and familiar choice. Moreover, although the quest tracks natural head movements in the HMD, the system is designed for maximising user comfort where one can optionally sit and participate in the experience. Therefore the rotation is also bound to the same joystick with a variable snap rotation angle. This is to prevent the motion sickness induced with frame or time-based rotation. In addition, the player position is tracked using a python script to plot a heat map based on the recorded player's X and Z (the height data from Y is irrelevant) coordinates. This is initialised when the player enters the shop in order to further understand the behaviour and actions of different users. (Figure 4.12). The shop environment overlays the map for a clearer read on the positions.

5. **Optimisation, Data Logging and Exception Handling:** Firstly, to keep better track of user interactions with respect to time, the *RuntimeLogger* class was implemented. This accumulates all the Debugger logs that are called during runtime with their corresponding times- further processed using word filters to discard redundant

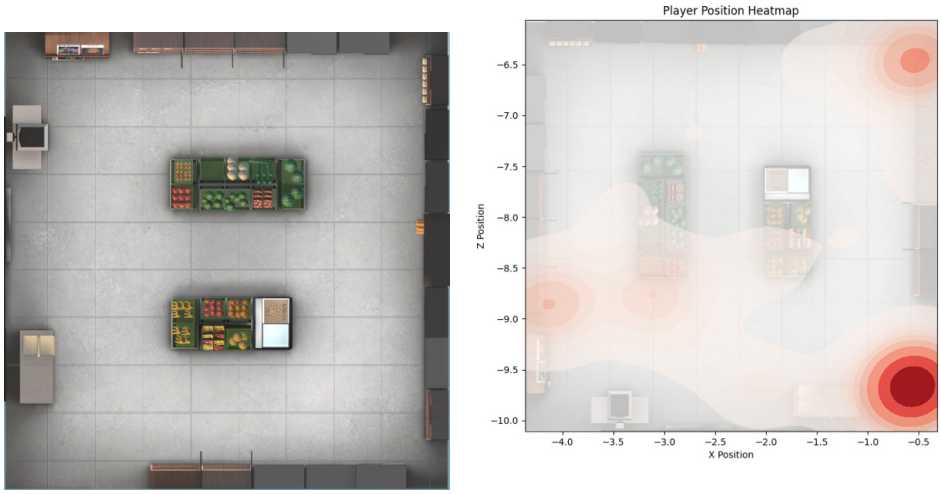


Figure 4.12.: **Left:** Top-down view of the shop interior; **Right:** Example of a heat map generated based on player position in shop

entries - and creates a simple text file. Moreover, contingency fail safe functions were created to handle exceptions during the tests. The *RunTestCases()* is a simple function that triggers different events throughout the program bound to distinct keys which allows flexibility and accessibility when users face unpredictable outcomes. This may include physics based glitches, or unresponsive controls or characters. This table (4.1) shows the different runtime test cases available.

Since the program consists of two scenes, indoors and outdoors, with the indoor scene containing more time-consuming activities, further care was taken to handle any bugs occurring during this segment. The *MarketSceneReset* class allows resetting the indoor segment of the program from any point during the playthrough. This not only resets the player's position to a default location, but also caches the transforms of all the physics based objects inside which is vital when there are dozens of objects that could be manipulated by the engine's physics systems such as gravity or velocity or by player intervention through grabbing and moving them. This results in a simple fallback mechanic to deal with collision and physics related glitches.

Furthermore, optimisation was done as much as possible on an engine and hardware level to improve performance, particularly for the Quest 2. The render resolution was reduced, the target frames per second (FPS) was capped at a lower count, with forced disabled V-Sync in order to lower the load on the HMD

Table 4.1.: Exception handling during runtime with keyboard shortcuts

Key	Action
0	Reset position
1	Start two-handed questions
2	Start Act 2 - Point to Leonard
3	Leonard hands the list
4	Move Leonard to shop
5	Enable blocked item collider
6	Audio test
7	Replay tutorial
R	Kick Remy from shop
N	End game

without compromising visual fidelity.

4.3.5. Prompting

Prompts are input which can be in various forms such as text, voice or input that are provided to the model to receive a corresponding output. Depending on the model and the output types it can provide, the prompts can be heavily detailed, leading to a field of study of its own as mentioned in this paper (Schulhoff et al. 2024). It provides further benchmark for the GPT-4o (or GPT-4-turbo) which the project utilises that boasts multi-step reasoning and cross-medium analysis capabilities and handling of large prompts (25000+ words). However, keeping consistency with the development goals of minimalism for testing the default capabilities of the models, the prompts were used sparingly and only during vital moments in the experience such as event changes or evaluating user input to provide specific responses. Firstly, in order to mask the AI by giving it personalities based on the NPCs, some system prompts were sent, as shown in this figure. System prompts, in this scenario, are simply instructions or guidelines provided to the LLM before initially responding to the user, which can change the tone or theme of the future responses. Additionally, the following figure explains how the prompts are used in conjunction with the quest flow.

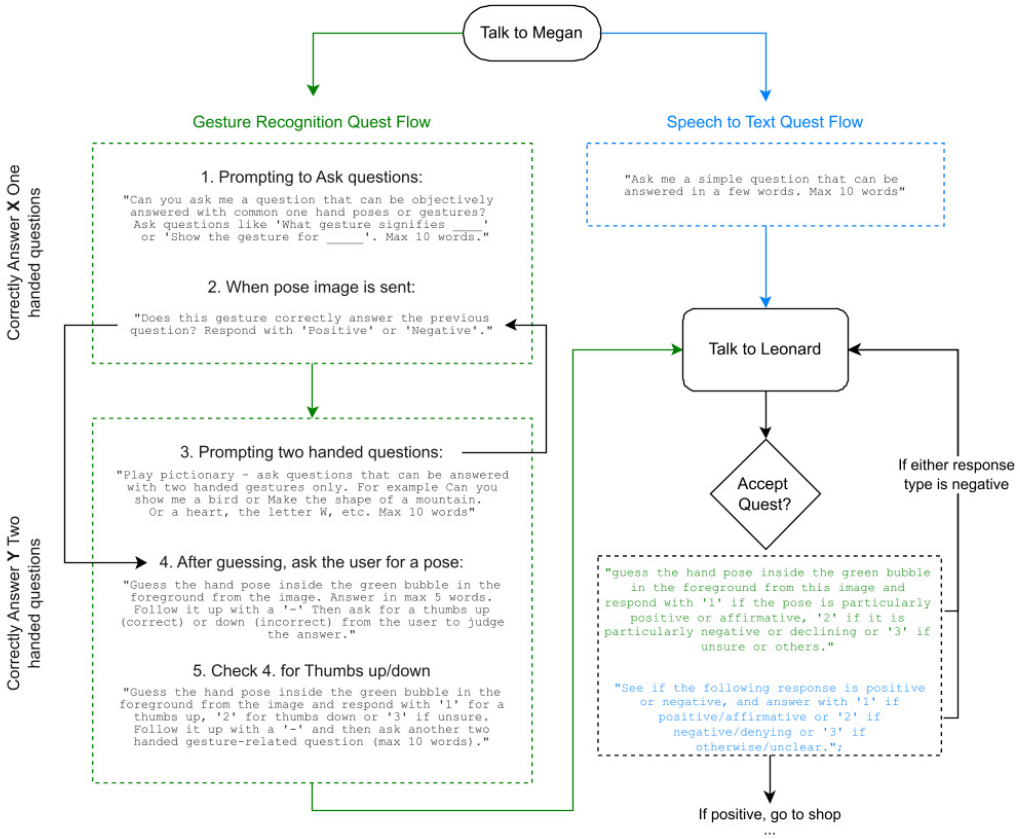


Figure 4.13.: The LLM prompts being queried based on the current objective and NPC in proximity

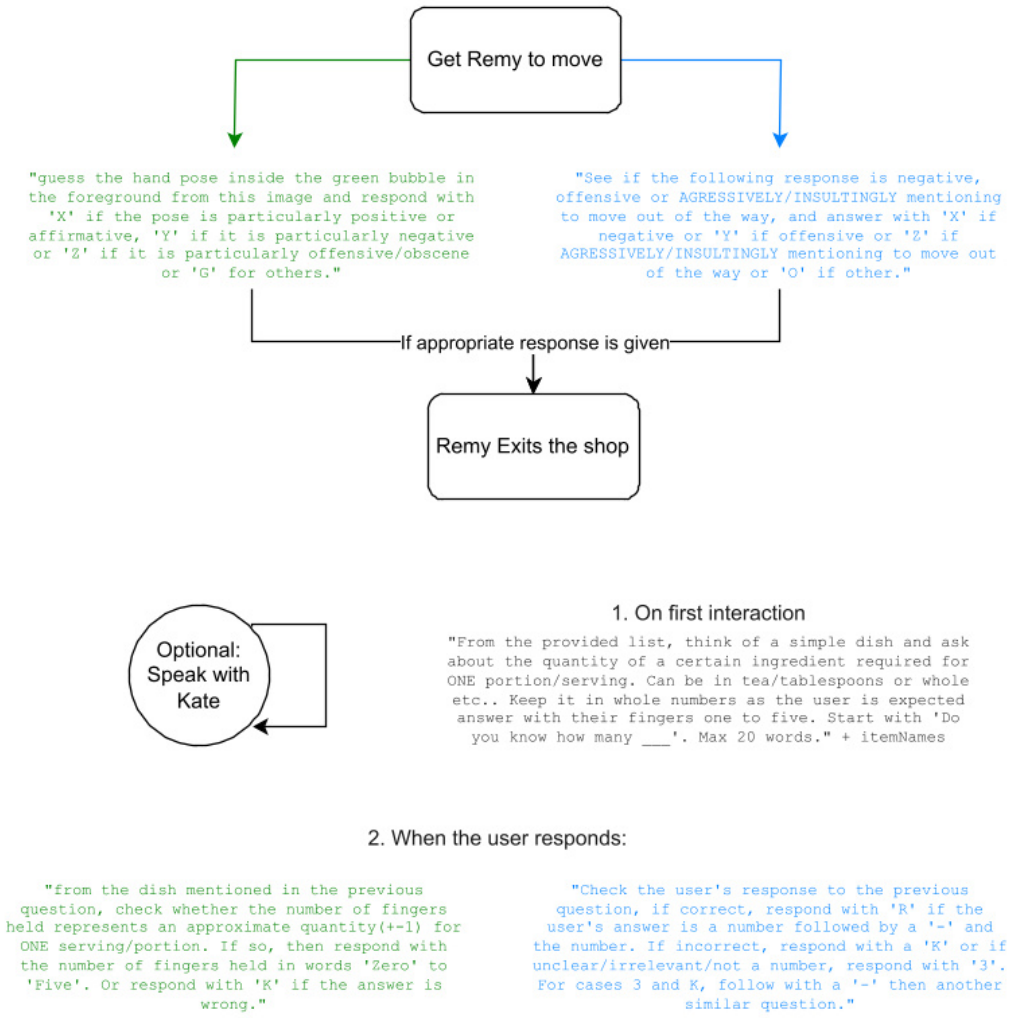


Figure 4.14.: The LLM prompts being queried based on the current objective and NPC in proximity

4.4. Experimental Setup

4.4.1. Apparatus

Minimal components allowed the hardware setup and the user studies to be simple, portable and time effective.

Computer

A laptop with the following specifications was used for development:

- **Model:** Lenovo Legion 5 15ACH6H
- **Processor:** AMD Ryzen 5 5600H, 6 Cores @3.3GHz
- **Memory:** 16GB DDR4
- **Graphics:** Nvidia RTX 3070 8GB GDDR6
- **Operating System:** Windows 10 Home 64-bit

VR Setup

For the gesture recognition and general gameplay, the Quest 2's optical hand tracking was used and its built-in microphone was used for the speech to text segment. For the experiment, it was sufficient for users to work on it while sitting and so the default stationary boundary of 1m x 1m (or 3 feet by 3 feet) set but Quest 2 was used (Figure 4.15).

Software and Plugins

For the development of the experiment, Unity editor version 2022.3.11f1 was used along with the following plugins and supporting software:

- **VR Tools:** Meta Quest Link App to connect the HMD to the computer and run the program. Meta All-in-One SDK version 72.0.0 and Oculus XR Plugin 4.1.2 to enable support for Oculus devices in Unity.
- **Programming and IDEs:** Visual Studio Code was used to develop the Unity project in C#. Python and PyCharm was utilised with updated versions of *numpy*, *pandas*, *seaborn* and *matplotlib* to handle calculations, read CSV files, generate distribution charts and plot information respectively. This was used to generate player position heat maps.

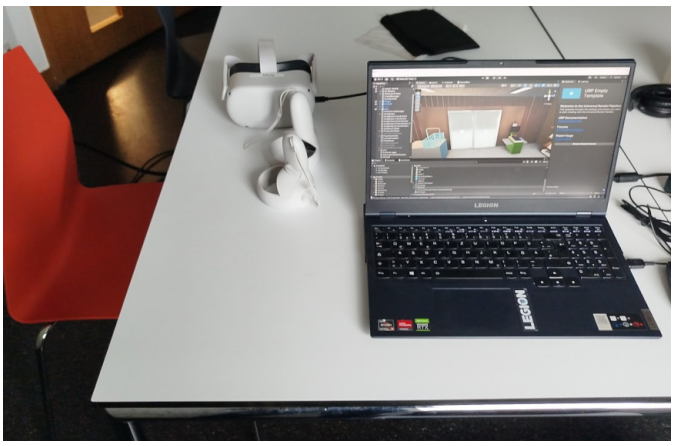


Figure 4.15.: The VR Test Setup

- **AI and LLM Tools:** OpenAI's GPT-4-turbo, also known as GPT-4o API's Chat Completions endpoint was used for the image recognition and text chat. The text chat was converted from speech using OpenAI's Whisper, with native run and Unity support via a bindings provided by user Macoron on GitHub (OpenAI 2022). For the NPC voice lines, an online voice generation tool ElevenLabs (ElevenLabs Inc. 2025) was used.
- **Project Assets Creation:** To create the environment, Blender was used for models and UVs, Adobe Illustrator for labels and decals and Adobe Substance 3D Painter for textures. All the secondary audio such as ambience and SFX were taken from Pixabay (Pixabay n.d.), a total of 6 audio files. Unity's TextMeshPro was used for all the in game text. For the NPCs, a free resource available online called Mixamo (Adobe Inc. 2025c) was used to obtain the character meshes and relevant animations.
- **Data Collection and Logging:** For generating logs from Unity, simple text editors such as Notepad++ was used, CSV Excel sheets were used as output for player position tracking, and Google Forms were used to collect user responses to questionnaires for further analysis.

4.4.2. Experiment Design

This subsection describes the experiment - the activities it involves, how it is set up, the selection of the participants and the data collected from it.

Procedure

To keep consistency with time and cohesion of content with the participants, an experiment protocol was established, elaborating all the required steps - from setting up the hardware to the completion of the final user discussion. The procedure is categorised as follows:

1. **Before the study:** Before the participants arrive for the experiment, the required hardware and play area must be set up and all the components must be tested. This consists of testing the HMD, controllers and hand tracking, both the VR experiences, gesture recognition and speech to text segments, the position and interaction logging data and the screen capture. Moreover it must be ensured that there are no delays or lags whilst running the program. In addition, the necessary documents required to record the participant's responses, reactions and answers to the discussion questions must be present.

2. **Consent and Onboarding:** Before beginning with the experiments, a brief description will be given, after which, each participant will be asked to provide their consent for data protection and confidentiality under the General Data Protection Regulation (GDPR) of the European Union (EU), acknowledge and allow the collection of the corresponding data relevant to the study. Consequently, the participants will be given a unique identifier or a participant number that can be used to identify the filled questionnaires. They would then be asked to answer the general information segment of the questionnaire which would record their participant number, age, gender, occupation and past experience with VR technology. Finally, after familiarising themselves with the Quest 2 controllers and comfortably fitting the HMD, the first part of the experiment would begin.

3. **During the Experiment:** For the gesture recognition and speech-to-text segments there will be one VR session followed by answering a set of questionnaires each. Each VR session is projected through initial independent tests to take 12 minutes and less for the second experiment, similar to answering each set of questionnaires. During the experiment, the participant's actions and reactions will be thoroughly observed and noted, guiding them where necessary and aiding them in bypassing any exceptions faced. (See 4.3.4 - 5. Optimisation, Data Logging and Exception Handling). Additionally, to ensure stability, the experiment type is altered with each participant. Therefore, odd numbered participant starts with the gesture recognition whereas even ones start with the speech to text experiment.

4. **Final Discussion:** After the experiments and the questionnaires, the participant were invited to a short discussion regarding the overall experience (See Post Study

Interview).

Participant Selection

The participants were openly selected without any bias to demographic or background. Different approaches of recruitment were taken, including convenience sampling through personal contacts and word-of-mouth, university mailing lists, social media and online forum postings and classroom announcements . Due to time, accessibility and the availability, many of the participants were notably university students, causing a natural demographic tilt. However, no criteria were placed on the selection except their own availability and volition to volunteer. The final number of participants were 12 of which 9 were male, 3 were female, ranging from 22 to 30 years in age (Mean Age: 27). As mentioned, most participants were students (11), besides one being a research assistant. 8 out of 12 participants had no prior experience with VR, however they all had experience with LLM based technology, most prominently in the form of chat assistants.

Data Collection

Through the logging of event and position data, along with the questionnaires and post study discussions, the following are recorded for analysis:

- **Mean duration of each VR session:** For each session type - gesture recognition and speech to text, how long did each participant take, on average, to complete the sessions.
- **LLM Performance Metrics:** How many times did the LLM respond correctly to the image gestures provided by the participant? This refers to the confusion matrix quadrants:
  - True Positive (TP): The user provides the right pose and the LLM outputs correct.
  - True Negative (TN): The user provides a wrong pose and the LLM answers incorrect.
  - False Positive (FP): The user provides a wrong pose and the LLM answers correct.
  - False Negative (FN): The user provides the right pose and the LLM answers incorrect.

Using the generated logs, these values are counted to compute the overall accuracy, precision and recall (how many of the actually correct poses can the model identify accurately), along with the harmonic F1 mean.

$$\text{Accuracy} = \frac{\text{TP + TN}}{\text{Total Questions}} \times 100\%$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP + FP}} \times 100\%$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP + FN}} \times 100\%$$

$$\text{F1} = 2 \times \frac{\text{Precision } \times \text{Recall}}{\text{Precision + Recall}}$$

- **Mean Time taken for each response:** How long did it take, on average, for the LLM to respond to the user's interactions (voice or image gestures). A longer response time would hamper immersion and make the interaction feel unnatural, breaking the 'illusion' of interacting with intelligent agents.

  For each participant:

  $$\bar{t}_i = \frac{T_i}{n_i}$$

  Where T_i is the Total response time and n_i is the number of responses in the session.

  For X participants:

  $$\bar{t}_{\text{overall}} = \frac{1}{X} \sum_{i=1}^{X} \left(\frac{T_i}{n_i} \right)$$

- **Uniqueness of responses:** - breakdown: unique questions asked (1H, 2H, Kate's questions) / total number of questions x 100%. Given the comparatively higher temperature and therefore creativity of responses from the LLM, how many distinct and unique responses did it provide over multiple runs and sessions with all the participants?

  $$\text{Uniqueness} = \frac{\text{Unique Responses}}{\text{All Responses}} \times 100\%$$

- **Player position data:** By observing the heatmaps generated from the player position (X and Z coordinates w.r.t. time), certain conclusions can be drawn. For instance, where the participants spent the most time and the focus of their activities, whether more time was given to the shopping objective or to interacting with the NPCs indoors.
- **Mean time taken to gather the items:** Besides tracking player position, a data point of interest is also the time taken for the participants to complete the shopping activity. This is tracked from the first time one enters the shop, after acknowledging the tutorial to adding the last listed item in the cart. The time taken to interact with the NPCs in the shop will be deducted.
- **Mean time taken between events:** Furthermore, it is important to track the average time taken by the participants to complete the each of objectives or events during the experience - from first meeting Megan, to speaking and helping Leonard, interacting with Remy when present (and optionally Kate) and completing the shopping quest. (See Figure 4.9).
- **Data from Questionnaires and Interviews:** Finally, in order to gain user insights on the experience with VR and the AI and the experiment in general, the qualitative discussion notes and quantitative questionnaire results will be extracted. The selected questionnaires for this research are described in the next subsection.

4.4.3. Questionnaire Selection

For an effective overall study, it must be accompanied by the right questions and hence a considerable effort was put to filter and select the most suitable set of questionnaires that is effective and complies with the research objectives while being time efficient. How a user perceived the AI through interacting with the NPCs while feeling spatially present and immersed was prioritised, along with their sense of agency and control during the experience. Initially considered questionnaires included the well-established Igroup Presence Questionnaire (IPQ) (Schubert et al. 2001), parts of the Intrinsic Motivation Inventory (IMI) (Ryan and Deci 2022) and the Trust Perception Human-Robot Interaction (HRI) (Schaefer 2016) among others. Table 4.2 shows the structure of the final questionnaires for the two VR experiment sessions.

Briefly, the IPQ is a time-tested and verified questionnaire to study the sense of presence felt in a Virtual Environment (VE). It consists of 14 questions considering multiple factors or 'subscales', namely, spatial presence - the sense of physical presence is the VE, involvement or level of engagement with the VE and the subjective experienced or perceived realism in the VE. The questions are set of a 7-point Likert scale from -3

to 3. Furthermore, The IMI questionnaire is similar in regards to consisting multiple independently valued questionnaire sets (on a 7-point Likert scale) related to a user's subjective experience with a target activity. This ranges from how the activity was perceived to the user's volition and motivation in taking part in the activity. Finally the Trust Perception HRI scale involves 40 questions in an 11 point scale from 0 to 100% asking users different questions about how much do they trust a robot or an agent to behave or perform in a certain manner.

However, despite the robust choices of questionnaires to consider, it proved to be a challenge to balance the duration, comprehension and cohesion with the current research objectives. Some deemed to be either simplistic or complex in its content or the number of questions were not suitable to extract an acceptable quantity or depth of information from the users. As a result, the questionnaires introduced in the following subsections were included in the final selection.

Artificial Social Agent Questionnaire (ASAQ)

The Artificial Social Agent Questionnaire (ASAQ) (Fitrianie et al. 2025) studies the human perception of the non human or artificial agents in a given task. The questionnaire is based on 19 uniquely labelled constructs or dimensions, ranging from the believability of the agent's appearance and behaviour to its personality and trustworthiness. It consists of a long and a short version with 90 and 24 items respectively, each scaled from -3 (disagree) to +3 (agree). The short version was selected for the final set of questionnaires which simply requires adding the construct scores for each participant. Since the scores are generated for each *agent* or the artificial social agent, firstly the data from both questionnaires were sorted by the VR session types (gesture recognition and speech to text) as the sessions were altered with each participant. Second, the scaled responses were added for each participant and a final mean was measured to provide the total ASAQ score of the agent, computed for the GR and STT agent/session type.

Sense of Agency Scale (SoAS)

Another desirable and important aspect of observing the users through the experiment was how much in control of their actions they felt and the free will they could exercise during the VR experience. This study (Tapal et al. 2017) elaborates on the direct and indirect measures of sense of agency. The direct measure refers to directly asking one regarding their sense of agency while the perceived sensory feedback one receives of agency and control is the indirect measure of the sense of agency. This study introduces the Sense of Agency Scale (SoAS) - an 11-item questionnaire - from an initial 36-item set - that studies the sense of positive and negative agency, where each question or

statement is scored in a 7-point Likert scale, from 'strongly disagree' to 'strongly agree'. It is split into two parts - Sense of Negative Agency (SoNA), which describes lack of self control and feeling of being controlled externally (these questions must be reverse scored) and vice versa with Sense of Positive Agency (SoPA), an increase sense of control and authorship over a user's actions. For each SoAS questionnaire, the responses were accumulated in a CSV file, and using several python scripts, the responses were divided by the SoNA and SoPA questions, reverse scored the SoNA responses accordingly and the mean values were computed.

Spatial Presence Questionnaire: MEC-SPQ

The Measurement, Effects, Conditions - Spatial Presence Questionnaire (MEC-SPQ) (Vorderer et al. 2004) consists of several sets of questions in a 5-point Likert scale, from 1 to 5, handling various topics around spatial presence. The sets are modular i.e. they can be broken down into four, six or eight question segments as required - each with its own scaled values. Three modules of four questions each, related to spatial presence were selected, namely, Spatial Situation Model (SSM), Self Location (SPSL), Possible Actions (SPPA), in addition to the Higher Cognitive Involvement and Suspension of Disbelief (SoD) questions. The computation of scores simply include general statistical metrics such as mean and standard deviation. A python script was utilised to compare the resulting mean values with the Confidence Interval (CI) bounds to the reference values from the paper (Vorderer et al. 2004).

General Questionnaire

A custom questionnaire with 18 questions in a 7-point Likert scale ('Strongly disagree' to 'Strongly Agree') was created to understand the perception the users had, specific to our VR experience. Despite not appearing in the final selection, some questions from the aforementioned questionnaires were adapted to complete this general set. In order to calculate relevant statistics, firstly, the negatively-worded statements or questions were reverse-scored (marked with '[R]', see Table 4.3) to ensure stability and readability in its interpretation(this means higher scores would always correlate to a positive response). The values from the answers are used to calculate the mean, median and standard deviation to understand user sentiment. Furthermore, the questions are categorised into 5 parts: *usability, interactivity, immersion, agency* and *psychological*. This is done in order to calculate the Cronbach alpha scores for each category, to determine the validity of the questions within each group.

Table 4.2.: Structure of questionnaire sets provided after each VR experiment session. Set 1 is asked to complete after the first experiment and 2 after the second, with the Estimated Time of Completion(ETC) for each segment of the questionnaire

Set	Questionnaire	Questions	ETC
1	About the experience	18	3 min
	ASAQ	24	4 min
	SoAS	11	3 min
2	About the experience	18	3 min
	ASAQ	24	4 min
	SoAS	11	3 min
	MEC-SPQ (in 5 parts)	20	3 min
	Total		23 min

Post Study Interview

Finally, the experiment session concludes with a short discussion using the following set of questions shown in Table 4.4. For this experiment, the questions and their corresponding classifications from this research (Rasch et al. 2025) deemed to be suitable and therefore, have been partially adapted.

Table 4.3.: Categorised list of general questions and statements about the experience, set on a 7-point Likert Scale from 'Strongly Disagree' to 'Strongly Agree'

Category	**No.**	**Questions**
Usability	2	The instructions were easy to follow
	6	The flow of the experience was easy to comprehend
Interactivity	3	I like the way I could interact with the NPCs
	4	The NPCs responded to my input correctly most of the time
	10	The activities in the experience were engaging
	11	I felt like I could relate to the NPCs in the experience
Immersion	5	The activities in the experience resembles the real world
	7	[R] I didn't feel immersed while doing the activities
	14	The interactions with the NPCs resembled reality
	15	The environment in the experience looked similar to the real world
	16	[R] I felt detached from the NPCs in the experience
Agency	8	The experience enabled me to think openly and creatively
	9	[R] I felt like I had no control or influence over the experience
Psychological	1	How much Motion Sickness did you feel during the experience?
	12	[R] The experience was too long
	13	I felt accomplished while doing the tasks in the experience
	17	[R] I felt stressed during the experience
	18	[R] The experience was boring

Table 4.4.: Classified List of Final User Discussion Questions

Category	**Questions**
General Experience	How was your experience carrying out the given task? Did the experience feel complete?
Immersion & Cohesion	How easy or difficult was it to follow instructions? How easy was it to get immersed in the environment?
User Preferences	Which control scheme did you like? Which medium did you like best - voice or hand gestures? Any outstanding NPC (good or bad)?
Perception of AI	How would you describe the relationship between you and the AI in the context of these tasks? Would you have preferred the AI to do more/less /other things? How else would you want to see such LLMs incorporated?
Suggestions	What features were/are you expecting in the future? Do you have any other comments or suggestions?

5. Results

This chapter presents the results obtained from the user experiments, surveys and interviews conducted, in addition to data collected during each user's play-through of both gesture and speech based VR experiences.

5.1. Quantitative Results

5.1.1. Data Collection From Logs

This subsection analyses the data collected from the generated logs when the users played through the different VR sessions. The experiment contains two such sessions, one for gesture recognition (GR) and the speech to text (STT) segment for control. With each participant, the VR session types alternate to counterbalance the order and reduce any biases occurring from the order of the sessions and allowing only the user's experience to affect the conditions. The results reflect the experience of 12 participants, identified by their unique participant numbers P1 to P12.

Mean duration of each VR session:

The mean time for all 12 participants to complete the gesture recognition and speech to text sessions were 10.2 minutes and 8.62 respectively. The minimum GR and STT times were 6.32 (P2) and 3.73 minutes (P7) respectively, where as the longest time taken were 14.1 (P5) and 12.35 (P10) minutes respectively. Both the times reflect increasing user familiarity with the Quest 2 controls and VR experience in general as the minimum times for both session types were achieved during the second session and the maximum times on the first (Table 5.1).

LLM Performance Metrics

This analysis is done on the gesture recognition session to answer textbf*RQ1* regarding the efficacy of the model's image recognition capabilities. This table (5.2) highlights the confusion matrix parameters to determine the overall accuracy and recall of the LLM, GPT-4-turbo or GPT-4o in particular. Firstly, the total number of relevant questions are recorded. Inside the experience, this refers to the distinct questions asked by the NPC

Table 5.1.: Task Completion Times by Participant (Total, GR, STT, and Shop Quest in Minutes)

P-No.	Total Time	GR Time	STT Time	GR Shop Quest	STT Shop Quest
P1	19.98	12.40	7.58	5.73	4.08
P2	11.52	6.32	5.20	5.17	2.76
P3	19.00	12.30	6.70	8.22	4.60
P4	18.96	10.50	8.46	3.13	1.60
P5	22.30	14.10	8.20	5.75	6.55
P6	19.18	7.58	11.60	7.83	3.60
P7	10.44	6.71	3.73	2.15	3.10
P8	19.51	7.76	11.75	3.82	5.80
P9	22.55	12.60	9.95	2.78	8.13
P10	21.73	9.38	12.35	2.95	6.80
P11	20.20	13.40	6.80	4.34	9.64
P12	20.91	9.81	11.10	3.98	6.97
Mean time	**18.90**	**10.20**	**8.62**	**4.65**	**5.30**

Megan to calibrate the one hand and two hand gestures posed by the participant. As per the event sequence, Megan keeps asking the user for one handed gestures until the user provides 3 correct poses and the model detects them correctly, therefore the true positive (TP) value in each GR session is 3. Moreover, after 3 correctly identified one hand poses, Megan asks the user to show two hand gestures, of which 2 must be answered correctly in order to proceed (TP = 2 for each session). As a result, out of 106 total questions asked to all participants in the GR sessions, total accuracy of the model was 73.6% with a 68.1% recall. Moreover, with no false positives, it scored 100% in recall with a 0.81 F1 score.

Table 5.2.: Confusion Matrix Summary for GR Task

Metric	Value
Total Questions	106
True Positives	60
False Positives	0
True Negatives (1H + 2H)	18
False Negatives	28
Accuracy	**73.6%**
Precision	**100.0%**
Recall	**68.1%**

Additionally, between each two hand gesture question, Megan is also prompted to ask the user to provide any gesture of their choice which the model will then guess. Here, out of a total 25 such questions, 11 were accurately identified (44%). As seen in Table 5.3, the number of questions asked fluctuates based on how many the NPC has to ask the participant before reaching the acceptance criteria (3 correct one handed and 2 two handed gestures).

Table 5.3.: GR Question Response Breakdown by Participant

P-No.	GR No. of Qs	GR 1H True Neg	GR 1H False Neg	GR 2H True Neg	GR 2H False Neg
P1	9	0	2	0	2
P2	6	0	0	0	1
P3	10	3	1	0	1
P4	10	0	2	1	2
P5	9	0	2	1	0
P6	9	4	2	-	-
P7	8	0	2	0	1
P8	7	0	1	0	1
P9	8	1	1	0	1
P10	10	2	3	0	0
P11	10	3	0	0	1
P12	10	2	1	1	1
Total	**106**	**15**	**17**	**3**	**11**

Furthermore, it was noted that, in the speech to text sessions, the responses were only

dependent on how accurately the ASR tool, Whisper, was able to convert the user's voice into text to send to the model. The prompt in the STT sessions were to ask users general questions and with the right STT conversion, the model managed to follow up with the correct responses flawlessly. In addition, in case of conversion errors, the model could be prompted to repeat previous messages and it was capable 100% of the time. Therefore, the GPT-4o model was well capable of following up with correct responses in all cases and the performance was only deterred by the accuracy of the speech to text conversion. In that regard, Whisper correctly performed in 36 of 51 such voice conversion instances, a 70.6% success rate. These voice instances are similar to the GR sessions where they refer to participants answering Megan's questions. However, there are some user related factors to be considered that could influence the outcome such as speaking unclearly or in a softer tone and reluctance or shyness to speak. This resulted in Whisper to falsely transcribe the words to generic sounds (transcriptions such as "[*bell sound*]", "[*music*]" or "[*BLANK_AUDIO*]"). Voice prompts influenced by errors such as answering in multiple sentences and prematurely releasing the voice record controls were not included. Overall, the conversion model performed adequately for the task.

Mean time taken for each response

In order to answer the ***RQ2*** - whether the model allows interactions with real time responses - the response time was recorded for both GR and STT sessions. As shown in Table 5.4, the response time is the difference between the image query sent to the LLM API and its response to the user. Similarly with the voice messages sent in the STT sessions. For GR, the average of the first 5 instances of response time were calculated and the first 3 for the STT. The resulting mean response time were 6.83 and 1.34 seconds for GR and STT sessions respectively. This exposes a significant drawback of such foundation models - the image recognition process and feedback is time-consuming (also noted in the paper by Maslych et al. (2025)), which results in unnatural and immersion-breaking interactions with humanoid NPCs. While this has been slightly mitigated using better visuals such as changing dialogues or animations indicating NPC thinking (reflective in the questionnaire results), according to the studies by Shi and Deng (2024) and Gnewuch et al. (2022), 6.83 seconds on average is an undesirable result for such interactions.

Uniqueness of responses

As implied in ***RQ3***, another aim of this research is to evaluate the diverse experience the use of such LLMs can provide and to inspect other ways to incorporate such

Table 5.4.: Average Response Time (in Seconds) for GR and STT Tasks by Participant

Metric	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	Mean
GR	7.01	7.00	7.40	6.54	6.40	6.40	6.60	6.60	7.20	6.40	7.20	7.62	**6.83**
STT	1.80	1.50	1.42	1.46	1.33	1.46	1.02	1.48	1.13	1.41	1.14	0.95	**1.34**

models. In the case of our VR scenes, the diversity in experience came from the distinct questions the NPCs (especially Megan for the calibration) asked the user. For the gesture recognition, out of the 106 total questions asked, only 16 were unique (15.1%), which were repeated in the other sessions. For the STT, the NPC Megan asked 3 questions each before the event transition, therefore a total of 36 questions of which 12 were unique (33.3%). Tables D.1 and D.2 list the unique questions asked in both session types.

The response *temperature* - which defines (from 0 to 2) the level of creativity allowed for the responses by the LLM (particularly OpenAI's GPT models) - was set to 0.5 for the images and 0.8 for the string message inputs. Despite this, while the unique questions asked per session was high, the overall number of repeated questions were significant. Moreover this is neither due to there being a limited number of one or two handed gestures one can do, nor is it limited by a restrained prompt to the API. There are many generic gestures such as the "rock pose" or showing numbers or shapes with one or both hands, that were never asked to any participant unless explicitly specified, for instance, asking for a recipe ingredient quantity in the case of the NPC Kate. Therefore, with minimal prompting or training, the overall diversity of responses are low.

Mean duration of Objectives

Additionally, the participants were observed on the duration of completing the prominent activity of shopping and collecting the items in the list. Each VR session (GR or STT) is broken down into three objectives which is implemented and represented in the logs as an integer called *Interaction Count*. Based on the acceptance criterion, the count is incremented. This table(5.5) defines the different interaction count values. It is of interest to observe the behaviour of the participants via how long they spend completing each objective as they are flexible depending on their interactions with the NPCs. For the GR sessions the mean time to reach objectives 1,2 and 3 are 3.21, 1 and 5 minutes respectively. Whereas for the STT, it is 1.26, 1 and 5.67 minutes respectively. Some noteworthy cases where individual timings varied greatly were for instance, completing objective 3 in the STT session took P7 1.72 minutes, and the same for P10

took 10.15 minutes. Table 5.6 provides the individual and mean times for objective completion.

Table 5.5.: Descriptions of interactions for GR and STT

Interaction Count	In GR	In STT
1	Answer 3 one handed and 2 two handed questions correctly	Talk to Megan for 3 turns
2	Show positive gesture to Leonard	Respond positively to Leonard
3	Collect all items on the list	Collect all items on the list

5.1.2. Data from Questionnaires

This subsection presents the results from user responses to the questionnaires answered after each VR session.

General Questionnaire - About the Experience

The general questionnaire consists of 18 questions regarding the overall user experience ranging from VR usability, activity engagement, duration, environment immersion and NPC interaction. After grouping and reverse scoring the negatively worded questions, the mean, median and standard deviation scores for each question as well as the upper and lower 95% confidence interval scores. The overall mean scores out of 7 resulted in 5.85 with a low SD of 1.2 for the first session type (GR) and 5.79 and 1.17 respectively for the second (STT) (Table 5.10). Moreover, the following Figures 5.1 show the mean scores for each question with their corresponding CI scores, a density plot for the domain classification.

Table 5.6.: Time taken by each participant to complete each objective by the Interaction Count (IC)

Sequence	P-No.	GR 1	GR 2	GR 3	STT 1	STT 2	STT 3
GR First	P1	3.25	1.65	6.76	1.56	0.87	4.75
	P3	4.13	1.15	5.97	1.40	0.85	2.91
	P5	2.90	1.55	9.27	2.13	0.85	4.90
	P7	2.08	0.67	3.75	0.90	0.78	1.72
	P9	4.25	1.35	6.68	1.38	0.71	6.88
	P11	3.52	0.92	8.60	0.90	1.39	4.19
Local Mean		3.36	1.22	6.84	1.38	0.91	4.22
STT First	P2	2.47	0.70	2.78	1.06	0.60	3.28
	P4	4.42	0.78	4.77	0.75	1.40	5.95
	P6	3.00	0.76	3.52	1.05	1.30	8.33
	P8	3.01	0.85	3.60	1.63	1.12	7.28
	P10	3.15	0.77	4.22	0.80	0.80	10.15
	P12	2.70	0.68	3.66	1.58	1.40	7.70
Local Mean		3.24	0.985	5.30	1.15	1.10	7.11
Mean Time		**3.21**	**1**	**5**	**1.26**	**1**	**5.67**

Table 5.7.: Cronbach's α Scores for GR and STT from the general questionnaire

Domain	GR	STT
Agency	-0.1422	0.5255
Immersion	0.1977	0.4397
Interactivity	0.3169	0.6088
Psychological	0.2797	0.2733
Usability	0.6344	0.51

Table 5.8.: Cronbach's α Scores for MEC-SPQ

Domain	Alpha
CogInv	0.6227
SPPA	0.2183
SPSL	0.8678
SSM	0.709
SoD	0.2843

Furthermore, As shown in Table 4.3, the 18 questions were divided into 5 categories in order to compute the Cronbach alpha scores which are detailed in Table 5.7. A few notable remarks from the observations are that the usability alpha score for GR is higher (α = 0.634 vs STT's 0.51), STT outperformed GR in terms of immersion, interactivity

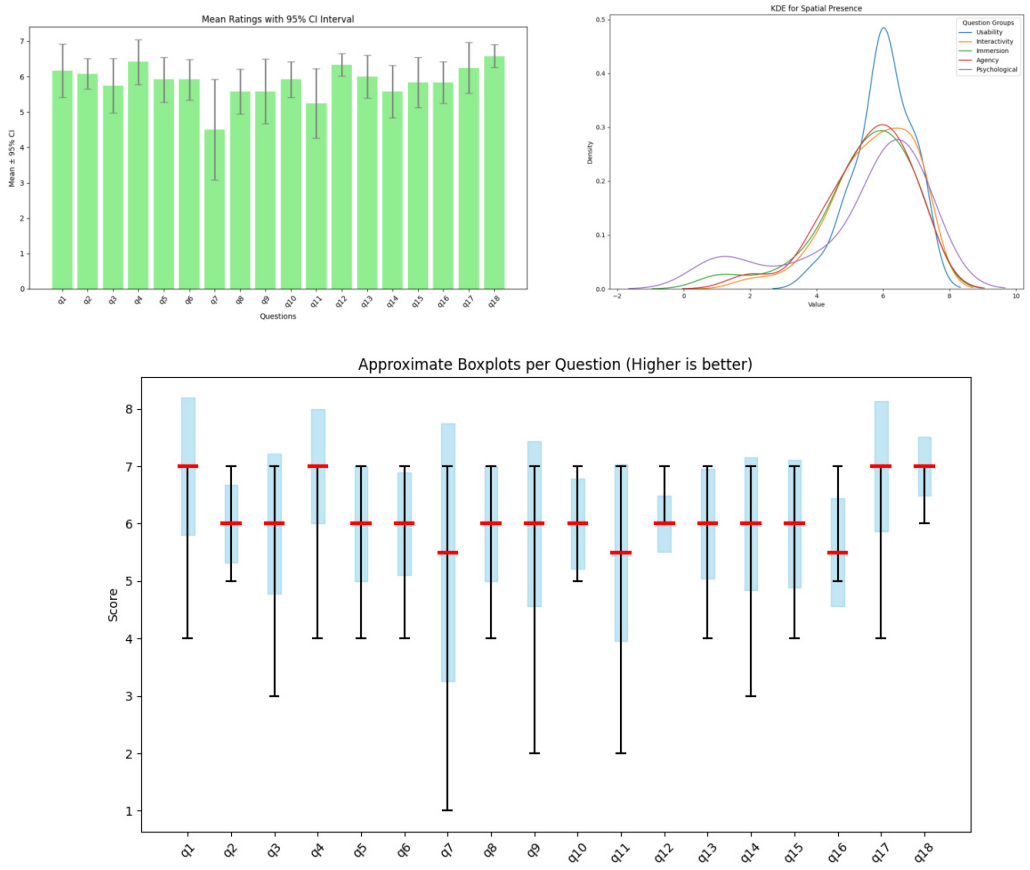


Figure 5.1.: **Top:** *Left:* Mean with 95% CI, *Right:* KDE Plot for questionnaire domain; **Bottom:** Approximate box plots for each question

and sense of agency . To conclude, with the slightly higher overall mean scores with similarly low standard deviation, it can be assumed that the gesture recognition is a similarly effective input and interaction method to the speech to text.

Artificial Social Agent Questionnaire

For each set of questionnaire presented to the participants, the data was processed in Google Forms, converted to CSV and scaled from a 1-7 Likert score to a -3 to +3 scale. Subsequently, the alternating participant scores were adjusted in order to split the files based on the VR session types, hence into a GR and STT based agent data. As shown in Table 5.11, summing the mean scores for the GR resulted in an ASA score of 38

Table 5.9.: Summary of control-related questions with means and standard deviations

Question	GR Mean	GR SD	STT Mean	STT SD
I feel in full control of my actions	6.17	0.94	6.42	0.79
The things I do are subject only to my free will	6.17	0.83	6.50	0.80
The decision whether and when to act is within my hands	6.00	1.13	6.42	0.67
My behavior is planned by me from the very beginning to the very end	4.58	1.83	5.17	1.70
I am completely responsible for everything that results from my actions	5.75	1.54	5.92	1.24
I feel like I am just an instrument in the hands of somebody or something else	5.75	0.87	6.00	1.04
My actions just happen without my intention	5.92	1.08	6.58	0.51
"My movements are automatic; my body simply makes them"	4.83	4.00	4.92	2.15
"The outcomes of my actions generally surprise me"	5.33	6.00	5.25	1.66
Nothing I do is actually voluntary	5.92	6.50	6.17	1.11
"While I am in action, I feel like I am a remote controlled robot"	6.33	7.00	6.17	1.11
SoPA	**5.734**		**6.09**	
SoNA (R-SoNA)	**2.32 (5.68)**		**2.15 (5.85)**	

and 38.66 for STT, a score higher than the 95th percentile benchmark reported in the paper (Fitrianie et al. 2025). While a direct comparison cannot be made due to the large number of participants and agents the benchmark is based on (*n=29*), it suggests that the two agents presented in our study aligns strongly with the original with a positive perception of the agents. Figures in 5.2 show the comparison between our GR and STT agent's score compared to the percentile ranking from the original paper.

Sense of Agency Scale

The questions from the SoAS are divided into positive(SoPA) and negative(SoNA) scales, determined by the average of all the corresponding questions in the subsets. In a 1 to 7 scale, the higher scores of SoPA and reversed-SoNA represent a stronger sense of perceived agency and control and a lower SoNA score defines a decreased sense of

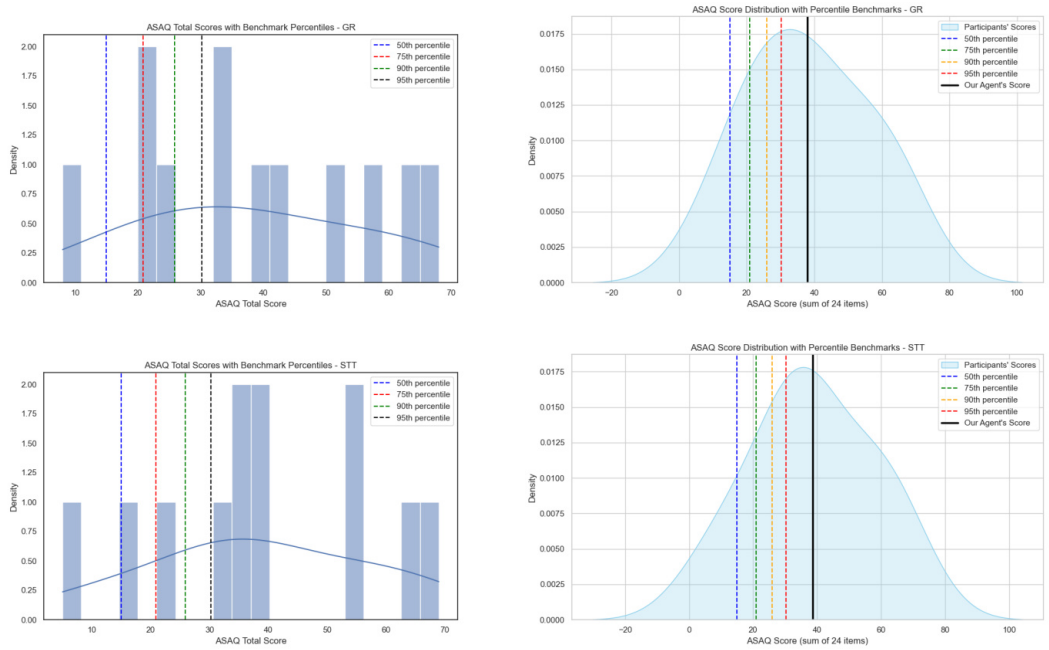


Figure 5.2.: Distributions of our agent's ASAQ score compared to the top percentile scores from the reference paper. **Top row:** Gesture recognition, **Bottom row:** Speech To Text

negative agency - a lack of control or feeling of external control. From our two sets of SoAS tests, the SoPA and SoNA scores were 5.73, 2.32, 6.09 and 2.15 respectively, as shown in Table 5.9. This indicates strong sense of agency over the participant's actions when interacting with the VR sessions. Figures in 5.3 show the density representations of the SoSA responses for both sessions.

MEC-Spatial Presence Questionnaire

Lastly, the MEC-SPQ requires simple calculations of general statistical metrics, the most prominent being the mean and SD values. From a scale of 1 to 5, the overall mean and SD from the survey (after reverse scoring) resulted in 4 and 0.69 respectively, indicating a strong and positive sense of presence and immersion in the VR environment, promoting the feeling of being physically present in a virtual environment. With the 20 questions being split into 5 domains, the Cronbach alpha scores were tested with acceptable SSM and SPSL values (Table 5.8). Furthermore, the reference mean values from the paper (Vorderer et al. 2004) were compared to the ones generated from our study with a 95% confidence interval bound and we found that even with 12 participants, responses

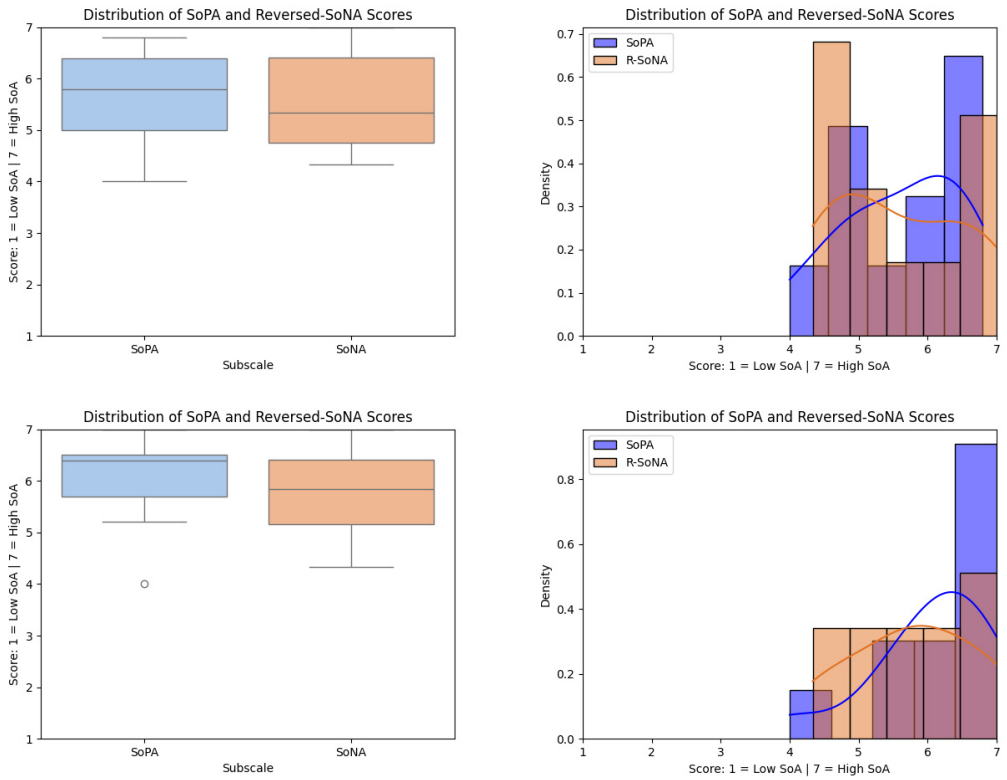


Figure 5.3.: Sense of Positive and Negative Agency Scale results. The higher the SoPA and Reversed-SoNA, the more the user feels positive control. **Top row:** Distribution for the GR Task, **bottom row:** Distribution for the STT Task

from 3 questions fell within the interval (Figure 5.4). Figure 5.5 depicts the distribution density of the responses based on the questionnaire domains.

5.2. Qualitative Results

5.2.1. Findings from the Interview

After both the VR sessions and their corresponding questionnaire sets, participants were invited to a voluntary interview session. 9 of 12 participants opted for the discussion. The following are some of the findings based on their responses structured according to categories Table 4.4.

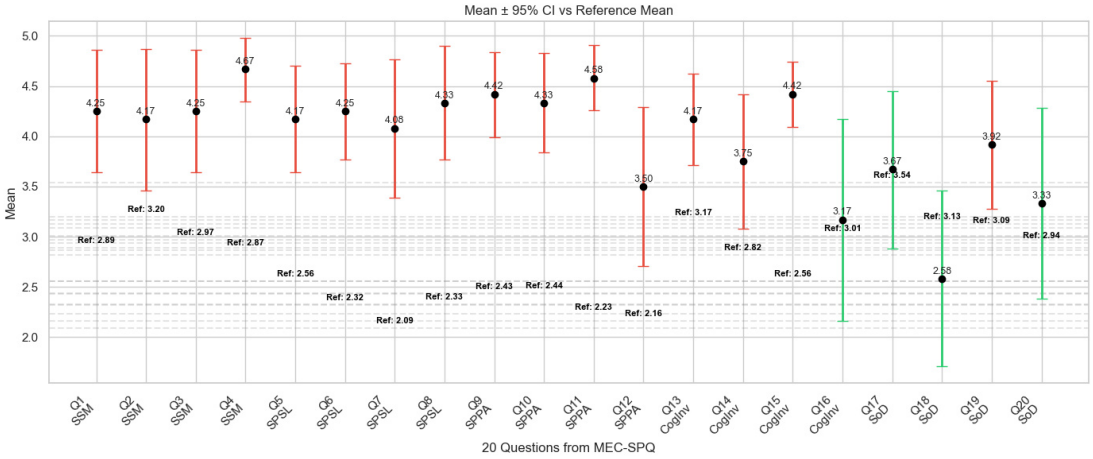


Figure 5.4.: Mean score with 95% CI compared with reference paper values

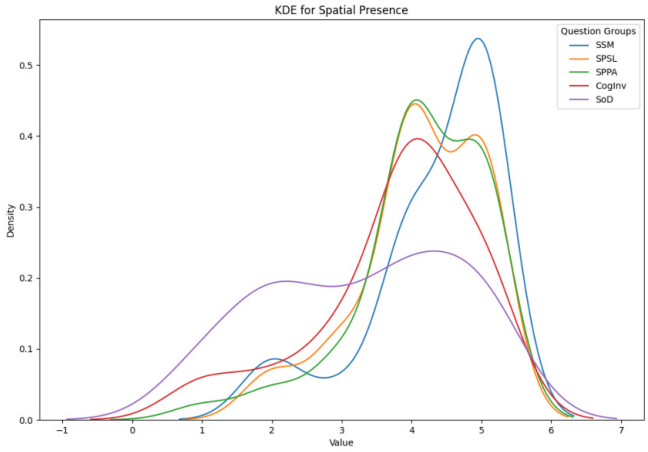


Figure 5.5.: KDE Plot for Spatial Presence Questionnaire

General Experience

Regarding whether the experience felt complete and made sense, especially the interactions with the NPCs, while the majority answered positively, P2, P4 and P8 pointed out that when they started the session, the initial interaction with the NPC Megan felt "disjointed" with the rest of the experience. As P4 remarked: *"It feels like you just jump in the game and she (Megan) just randomly starts asking you questions from nowhere. But after her part, it feels natural"*. As users are unaware of their calibration task with Megan for blind impressions, this is an important insight to improve game design and incorporate NPC's roles in the experience more comprehensively.

Majority of the users agreed that the experience - the scenario and activities - felt comprehensive and relatable, despite lacking the amount of activities in the scene. When asked about their perceived relationship with the AI in the context of the tasks, some participants commented:

P1:*"It like a basic human interaction."*

P2:*"There aren't many interactions but it felt real - like helping someone."*

P6:*"It felt real, like a stranger asking for help in the scenario."*

This shows promise and potential for incorporating such LLM or AI based agents in more virtual settings.

Immersion and Cohesion

Majority of the users responded that they felt immersed in the environment; *("Fully immersed", "Engaging", "felt like a real shop", etc.)*, especially inside the shop. P3 and P8 particularly remarked:

"The outside area looks cut off, it feels unrealistic but it's great once I am in the closed space inside the shop".

"The exterior made me feel like I'm floating on some island with the people but inside the shop it looks just like a regular Bio-markt or Penny (supermarkets in Germany)".

Users unanimously agreed that the instructions were intuitive, hence, easy to follow. *"Yes, it was easy to understand because there weren't many controls"* and P10: *"I never used VR so it was difficult at first but I could use it very easily on the second try"*.

User Preferences

When asked for control type preference, majority of the users (6) opted for hand tracking as it allowed them more agency and natural control while 3 participants preferred using both. *"It felt very natural and intuitive to grab things with my hands in the real world and do the motion and see it happening in the VR"*. Additional positive remarks

were made when the users could realise that they could use both hands to grab multiple items.

In terms of session type preference, 5 participants mentioned they preferred the STT, 3 mentioned GR and one mentioned both.

Perception of AI

An interesting psychological phenomenon was observed when asking about the personality of the NPCs. Of the four, a majority of the participants mentioned that they disliked Remy because as an event progression criteria they were supposed to either give a negative or offensive gesture or say something similar for the STT. Some participants remarked that they "felt reluctant" to behave accordingly because it seemed realistic. This hints to the effective immersion and natural appearance of the NPCs.

Finally, when asked about how else would the users like to see such agents incorporated, or any features they expected before or after their experience, they commonly responded with usability in conversation and interaction focused games and educational assistance:

P2: *"It (the STT version) can be fun in games. Cosy games like Animal Crossing can use this because at one point they run out of dialogues and I want to talk to them more. If they can have some small AI for each NPC then there can be more conversations."*

P11: *"It will be fun to see this incorporated in some kind of online grocery where you can actually buy things from the supermarkets and get them delivered..."*

Suggestions

In terms of feedback on current features, there was a resounding response for simply having more content to interact with - more NPCs, interactions, longer conversations, dynamic animations and reactions and independent actions powered by the NPCs. P6 for instance mentioned that there was a *"cash point in the shop"* and desired to *"buy and pay for stuff myself"* as also remarked by P1.

About the UI, P6 and P10 remarked issues with focusing on certain items on the foreground like the grocery list. P5 noted the typical physics glitches with multiple object collisions and UI layering.

5.2.2. Experiment Observations

Throughout the experience, the participants were observed and guided where required, along with the elements in the VR sessions and some noteworthy observations were made:

- The model's image recognition performance dropped significantly while detecting two handed gestures and in case of higher occlusions. For instance, when Megan prompts the user to provide a pose to guess, the user's two handed gestures are incorrectly recognised as being part of Megan (User showing a fist gesture received a response "putting hand over face"). Moreover, inside the shop, when showing a gesture to Remy - typically a negative, thumbs down gesture it was detected incorrectly as "hand on back" on many instances. This shows the necessity of more effective prompt engineering for such descriptive cases.
- Likely due to more data available for training the LLMs that cover common poses such as thumbs-up or the open palm gesture, these were recognised correctly more frequently than others.
- Interestingly, most participants saw and acknowledged the tutorials but didn't implement them, except P6 and P10. For instance, some read the instructions on how to pick up objects with their hands but were surprised when they were later told that it was indeed possible. However, this may be the result of lack of VR experience.
- Some users were observed expecting more responsive features from the GR or STT methods or the VR hand tracking capabilities in general. For instance, in case of Remy some tried to tap his shoulder to get his attention, while others tried to speak to Kate, asking and expecting her to move. This proves that the users anticipated more dynamic dialogue, physics and animation options from the otherwise predefined character features, with their actions tied to the experience's event logic.
- Lastly, the duration spent on locations inside the shop depended on whether Remy was present in the shop and what items the users had to pick for the objective which is reflected by their generated heatmaps (Appendix B).

Table 5.10.: Summary statistics for the General Questionnaire

Question	Mean	Median	SD	95% CI ↓	95% CI ↑
How much Motion Sickness did you feel during the experience?	6.42	7	1	5.78	7.05
The instructions were easy to follow	6.08	6	0.67	5.66	6.51
I like the way I could interact with the NPCs	5.92	6	1.24	5.13	6.7
The NPCs responded to my input correctly most of the time	6.25	7	1.06	5.58	6.92
The activities in the experience resembles the real world	5.67	5.5	0.98	5.04	6.29
The flow of the experience was easy to comprehend	6	6	0.85	5.46	6.54
I didn't feel immersed while doing the activities	5.17	6	2.04	3.87	6.46
The experience enabled me to think openly and creatively	5.75	6	1.14	5.03	6.47
I felt like I had no control or influence over the experience	5	6	2.04	3.7	6.3
The activities in the experience were engaging	6.08	6	0.79	5.58	6.59
I felt like I could relate to the NPCs in the experience	5.25	5.5	1.6	4.23	6.27
I felt accomplished while doing the tasks in the experience	5.92	6	1.62	4.89	6.95
The experience was too long	5.58	6	1.68	4.52	6.65
The interactions with the NPCs resembled reality	5.75	6	1.22	4.98	6.52
I felt detached from the NPCs in the experience	5.92	6	1	5.28	6.55
The environment in the experience looked similar to the real world	5.83	5.5	0.94	5.24	6.43
I felt stressed during the experience	6.08	6.5	1.31	5.25	6.92
The experience was boring	6.67	7	0.49	6.35	6.98
RESULTS	**5.85**	**6.11**	**1.20**		

Table 5.11.: Summary statistics for The Artificial Social Agent Questionnaire

Questions	**GR Mean**	**GR SD**	**STT Mean**	**STT SD**
The NPCs have the appearance of a human	2.17	0.94	2.08	1.24
The NPCs have a human-like manner	2.08	0.79	1.83	1.34
The NPCs seem natural from their outward appearance	2.00	0.85	2.08	0.79
The NPCs react like a living organism	1.33	1.61	1.50	1.17
The NPCs' appearance is appropriate	2.17	0.72	2.42	0.67
The NPC agents are easy to use	1.58	1.24	2.08	0.79
Each NPC does its task well	2.08	1.00	2.00	0.85
I like the NPCs	1.67	1.23	1.83	1.19
The NPCs can easily mix socially	1.33	1.50	1.33	1.23
The NPCs have distinctive character	1.17	1.47	1.25	1.48
I will use the NPC agent again in the future	1.92	1.00	1.92	1.08
The NPCs are boring	2.00	1.41	2.00	1.04
The interactions captured my attention	2.42	0.51	1.83	0.94
I can rely on the NPCs	0.67	1.72	0.67	1.50
The NPCs and I have a strategic alliance	0.75	1.22	0.67	0.89
The NPCs are attentive	1.75	1.06	1.67	1.37
The NPC's behaviour does not make sense	2.25	0.62	2.00	1.54
The NPCs have no clue of what they are doing	1.92	1.16	2.00	1.13
I see the interaction with the NPCs as something positive	1.83	0.83	1.92	0.90
The NPCs are social entities	1.42	1.16	1.83	0.72
Others would encourage me to use the NPC agent	0.67	1.87	0.83	1.75
The NPCs are emotionless	1.08	1.83	0.67	2.15
The emotions I feel during the interaction are caused by the NPCs	1.00	1.65	1.08	1.56
The NPC's and my emotions change to what we do to each other	0.75	1.66	1.17	1.11
Short ASAQ (Sum of Mean)	**38.01**		**38.66**	

6. Discussion

This chapter completes the presented research with a discussion on the overall outcome and what was learned from it, what limitations were present and what are the steps that could be considered in the future to overcome them.

6.1. Learning Outcomes

From the experiments, surveys and interviews conducted on the participants and from the comparative literature relating to use of LLMs in VR to create interactive NPCs, the following were observed:

Firstly, in order to create lifelike interactions between humans and LLMs, it is imperative to ensure the models respond within natural time and therefore the latency is a major factor. This has been emphasised by authors their respective studies as well. There is however one interim solution to superficially mitigate the feeling of delay, that is through proper implementation of user experience design. For instance, even with a response delay for gestures from the model being 6.8 seconds on average (where human responses are under a second), many users tend to agree that the NPCs reacted realistically and none actually pointed out the delay. Figures in 6.1 show two such statistics of user responses, in both the general and artificial social agent questionnaires that the majority agree that the NPC interactions and behaviour represented reality. This may be due to effective UX design choices - using a thinking animation and dialogue for instance (Figure 4.8). Strategies mentioned in several research (Buldu et al. (2025), Shoa and Friedman (2025), etc.) such as running models on local servers and parallel transcription and response, etc. should be adopted to achieve better results.

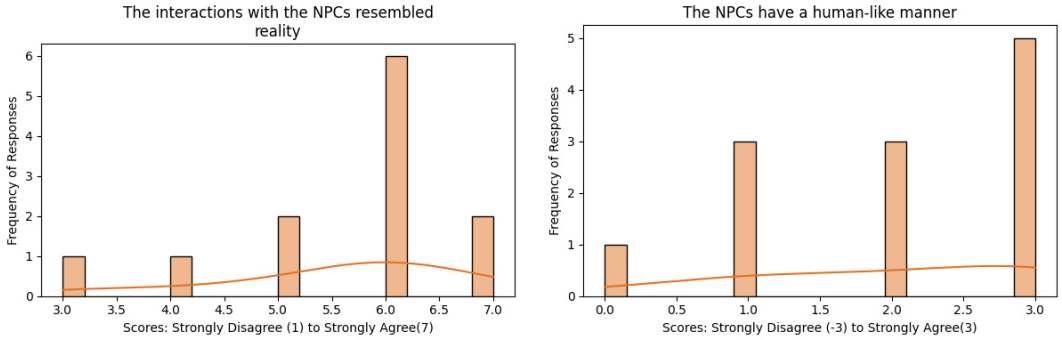


Figure 6.1.: User responses for NPCs Human Likeness.

In addition to quickness of response, the model must also be able to provide correct responses and minimise context bias or hallucinations. This is also something most papers (and ours) have aimed to achieve (Table 3.1) - a better overall accuracy of detection or recognition of input. The current GPT-4o model shows promise in providing without any pre-processing or computer vision aid based on our study (73.6%) which is also reflected by the user responses over multiple questionnaires as shown in Figure6.2. This can be considered as a foundational step as techniques displayed by Naidu et al. (2025), Qi et al. (2025), etc. can only lead to further improvement. Techniques including writing more precise and descriptive prompts with chain-of-thought reasoning or adding more knowledge via RAG can significantly improve gesture recognition accuracy.

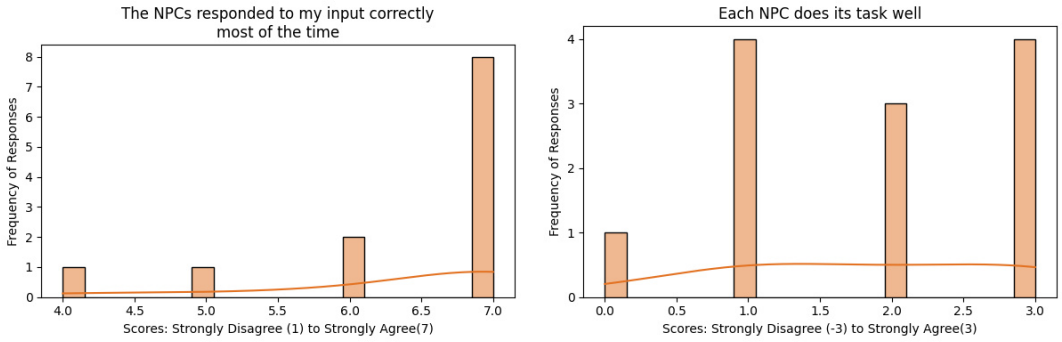


Figure 6.2.: User responses for NPCs Response Accuracy

Lastly, another attribute to ensuring a likable and realistic interaction scenario with LLM based avatars is the response consistency (also mentioned by Triyono et al. (2024)). This refers to the NPCs being capable of answering the user with the right context at the correct sequence and prevent any repetitions unless it is able to recall the information in

a realistic way. This can be achieved using better chat memory management techniques and overall with more optimised models in the future. Our study attempted to measure this by learning the uniqueness of responses, resulting 15.1% and 33.3% uniqueness scores for gesture and speech based sessions respectively - a particularly lower score for the speech to text segment given the open ended nature of the question prompted to the model. Hence, in order to further successfully develop such AI or LLM based interactions in VR, these factors must be strictly considered.

From the VR perspective, we also learned that such interactions may only seem realistic if the users themselves feel at ease to engage with such conversation agents. Therefore, the development process should also take the virtual environment into consideration. The user experience design must deliver more than simply the aesthetics, it should provide the ease of use and control for the users. For instance, ensuring minimum sense of motion sickness when using the VR is a vital first step of usability. For our study user comfort and accessibility was a cornerstone for the overall design and therefore choices such as seating users for the experience or enforcing snap rotation with the movement controls were implemented. As a result, while all the participants were relatively new to VR and a couple (P2, P4) mentioned prior concerns with motion sickness, the feedback on it was positive (6.3A). Moreover, users reported feeling of positive control over their actions (6.3B) and no particular discomforts when interacting with the NPCs from a VR standpoint, which also reflected positively in spatial presence scores (Appendix A.0.7).

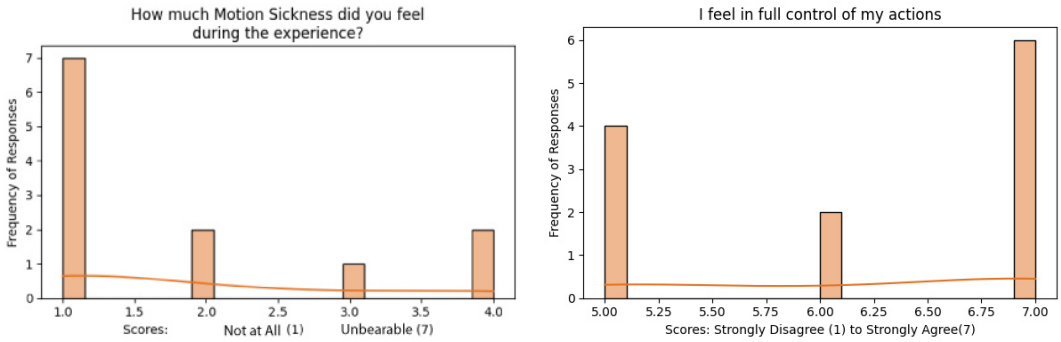


Figure 6.3.: User responses for Motion sickness and control during the experience - General and Sense of Agency Questionnaires

Overall, in it's current state, the foundation GPT-4o model is considerably reliable as a conversation agent for multimodal input. According to the users, the NPCs in general were believable, reliable, provided distinct characteristics and showed personality using simple system prompts and had emotional influence over them (Figure 6.4). For

detailed user responses see Section 5.2. However, this must also be complemented with the appropriate visuals and sounds for enhances realism and immersion such as using human avatars as described by Pan et al. (2024), Lim et al. (2025), etc.

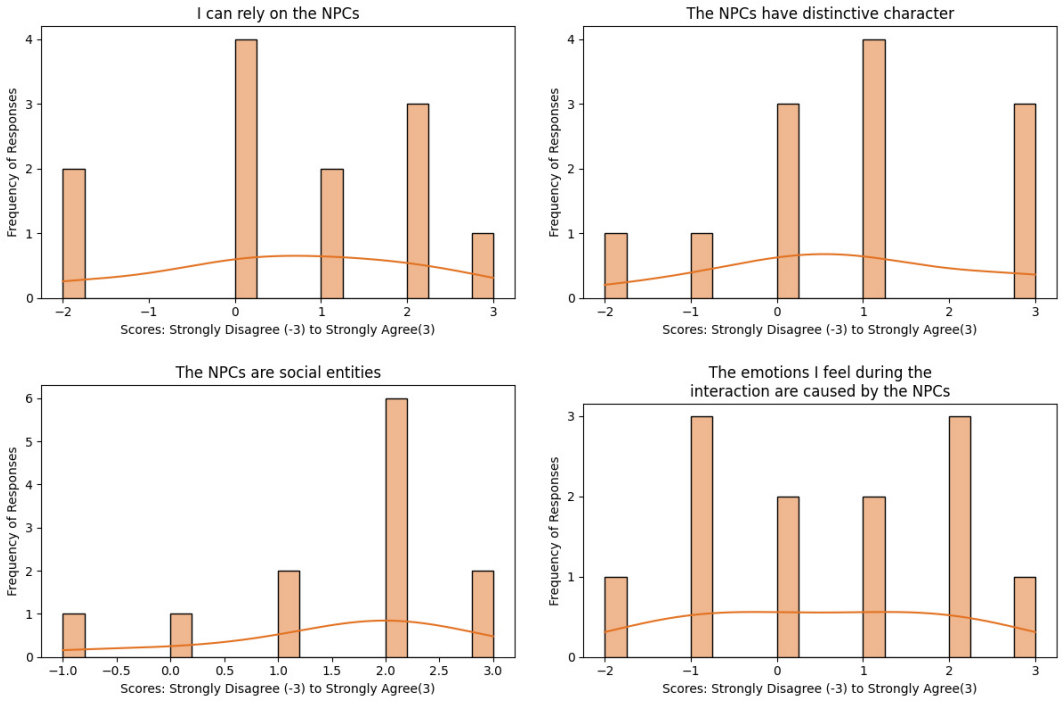


Figure 6.4.: User responses for NPC being believable and reliable social entities - from the Artificial Social Agent Questionnaire

6.2. Limitations

Lack of Participants

The primary shortcoming was the inadequate number of users to participate in the study. The nature of this voluntary study contributed to - among other external factors - the attendance of the participants which in turn greatly influenced the scope of the generated results. Firstly, the announced approximate completion time of the entire study was 50 minutes which was advertised primarily to university student social media and communication groups, mail listings and in-class announcements. Without a strong motivation to volunteer, the study duration dissuaded and hampered attendance. Moreover, this also influenced the demographic of the participants as 11 of 12 were

students with a small margin in age (which, as a pilot study, is rather favourable).

Results Comparison

As aforementioned, the scarcity of participants greatly determined the final metrics. Firstly, due to time constraints, user participation, engagement and general expectation of user demographic, the selection of questionnaires were limited. In addition, many of the available and well established questionnaires have not yet been standardised or adapted for AI based agent interactions. The final selection were confirmed as they conformed to the nature of the research, namely user's sense of agency, spatial presence and believability of the AI as a human.

The corresponding studies (ASAQ, SoAS and MEC-SPQ) have produced results based on hundreds of participants (and compares significantly large number of AI agents in the case of ASAQ) and therefore many scores were not directly comparable. These led to unstable and unexpected scores such as poor alpha scores for the MEC-SPQ and general questionnaires (this is perhaps due to number, arrangement, type or grouping of the questions). In addition, in case of the gesture recognition, it was noted that the model produced zero false positives, leading to a 100% score in precision, which proves that it is exceptionally conservative and strict in its responses and therefore, a strong yet inconclusive F1 score (0.81).

However, as an exploratory study, these questionnaires provided a good insight to understanding initial impressions. Moreover, some metrics indicated desirable results even with much fewer participants, for instance the confidence interval scores for some questions in the MEC-SPQ.

Sub-Optimal Model Performance

Despite the capabilities of the GPT-4o model, similar response were observed even with higher temperature scores. For instance, when asking for a recipe, it provided the same recipe 90% of the time even with temperature as high as 1.5, and anything beyond it completely broke the response structure. To overcome this, a list of different recipes were requested and from there a random one was selected to generate for a run. Similarly with the GR session questions from Megan (15.1% unique questions) or recipe related questions from Kate, in most runs, the same questions were asked with a message temperature of 0.8, which may be the result of inadequate prompting, using the minimal, zero shot approach on foundation models. Moreover, in an attempt to decrease the response time for image recognitions, the model also performed poorly, when the images were reasonably scaled down in resolution or quality.

Technical Limitations

Lastly, some unexpected hardware and software limitations were observed. First of all, due to the Quest 2 fixing its Field of View (FoV) within Unity, the perspective remained fixed which prevented a wider and more natural FoV to be set, creating UI/UX challenges such as interface element placement (most prominently the gesture detection sphere). Moreover, despite an attempt to force the Unity's UI Canvas with the highest layer order using a script, it did not render on top, leading elements such as dialogue boxes and the grocery list to move behind other objects in the scene.

6.3. Future Work

Improve Test Conditions

The first direction we aim to improve on is our primary limitation with the user tests. This will be achieved by investigating options and constraints by conducting a preliminary feasibility analysis and scaling the study accordingly. The scaling may include reconsidering and redesigning the study with a larger time frame and (if required) a reevaluation of the questionnaires present. The tests will also be promoted to a wider and diverse demographic for more reliable general scores.

Model Performance Improvements

In the future, we aim to vastly improve the model's performances, particularly the gesture recognition and its response times with effective image compression techniques, more comprehensive yet concise prompts or parallel processing/pre-processing methods (if applicable) and investigating locally run LLMs to eliminate any network latencies. Currently with a mean time of 6.8 seconds, it is highly unnatural and undesirable and must be masked by visual or auditory elements which only slightly mitigates the issue.

Better Interactions

- Accordingly to participant feedback, 3D character animation services such as Mixamo can be (if available) integrated via an API into Unity in order to add more dynamic animations and interactions and to let the LLM decide which animations to call.
- Moreover, we will inspect and aim to introduce dynamic responses from the user end such as video capture gestures for increased interaction options.

- Furthermore, we also aim to test this model further with more body tracking capabilities via different VR hardware (such as HTC Vive (HTC Corporation 2016)) which will significantly raise the gesture options and allow testing for body language recognition as well.

Leverage Better Models

An enhanced model by OpenAI, specifically targeted for vision based tasked called GPT-4 Vision (OpenAI 2024) is aimed to be used upon commercial release of the API after exhaustive improvements to the current GPT-4o. This will provide certain benefits:

- Enhanced aptitude in handling vision-based tasks could also mean reduced complexity of prompts to the LLM dedicated to understanding the environment. Therefore, one target would be to replace system prompts with videos or images of the environment for the LLM to decide its behaviour and tone.
- Some participants wished for the NPCs in the shopping scenario to assume an assistant role and by using the scene understanding capabilities, the model could operate as guidance systems - knowing product location and proximity to the NPC to assist the player looking for them.

Robust Scenarios

To further test whether gesture recognition is a viable option over or with others such as voice, we would (also desired by participants) like to introduce more diverse scenarios and opportunities for interaction. This can also be aided by utilising generative AI like LLMR (Torre et al. 2024) for simple object generation at runtime, contributing to further increasing user playability and interaction options.

6.4. Conclusion

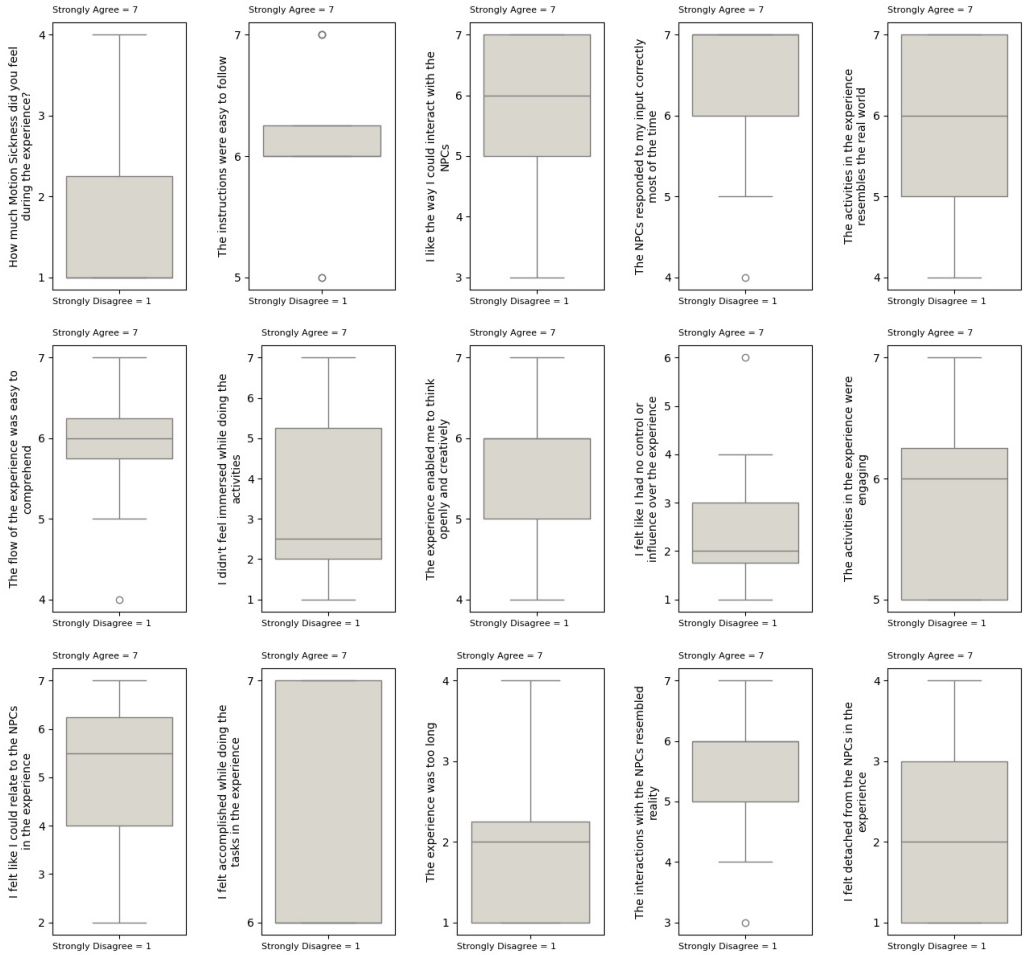
This paper presents a work on creating and testing a VR scene with in-game character AI powered by the LLM GPT-4o to handle gesture recognition and speech to text provided by the player. A study is conducted on several participants to understand - via the VR experience and appropriate questionnaires - their interactions with the AI and how they perceive the environment, whether they the AI comprehensible and believable, while feeling in control over their actions. The aim of the study is to see whether gesture recognition via hand tracking is viable as a medium of interaction in such virtual reality scenarios and whether the supporting AI can provide a realistic response to such gestures. The development of the experience was approached with minimalism

(zero-shot foundation models) and accessibility (readily available commercial hardware in Quest 2, using Unity engine for performance efficiency and extremely cost effective LLM API) in mind. Consequently, the model achieved an adequate detection accuracy of 73.6% with low errors (0 identified false positives) while having poor response times (6.8 seconds). Despite this and the low participant numbers, as an experience, it delivers overall positive response scores from the different questionnaires and lays the groundwork for a larger scope of improvement in the future.

A. Appendix A: Questionnaire Response Data

A.0.1. About the Experience: Gesture Recognition Task

Figure A.1.: Participant responses for the general questionnaire (Gesture Recognition)



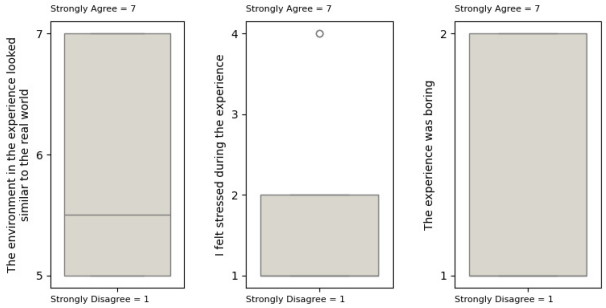


Figure A.2.: General Questionnaire Correlation Heatmap - GR

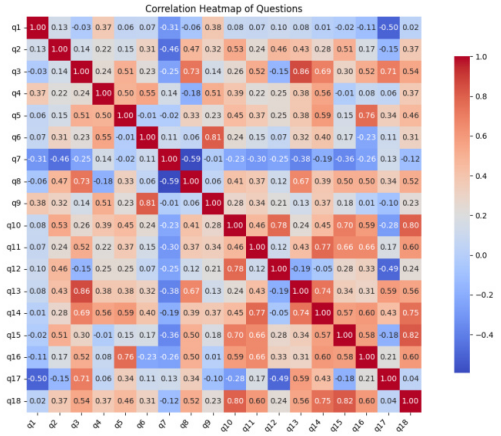
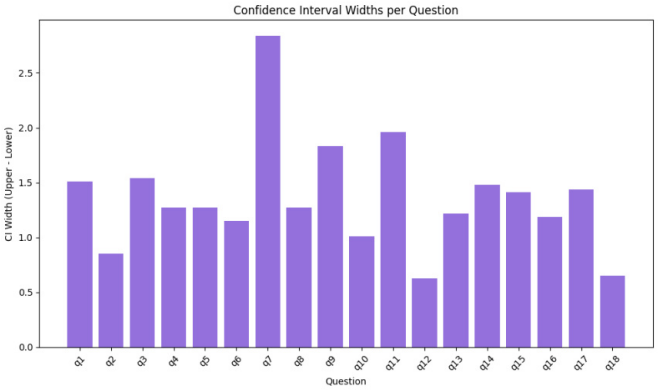


Figure A.3.: GR Task CI Interval Chart



A.0.2. About the Experience: Speech To Text Task

Figure A.4.: Participant responses for the general questionnaire (Speech To Text)

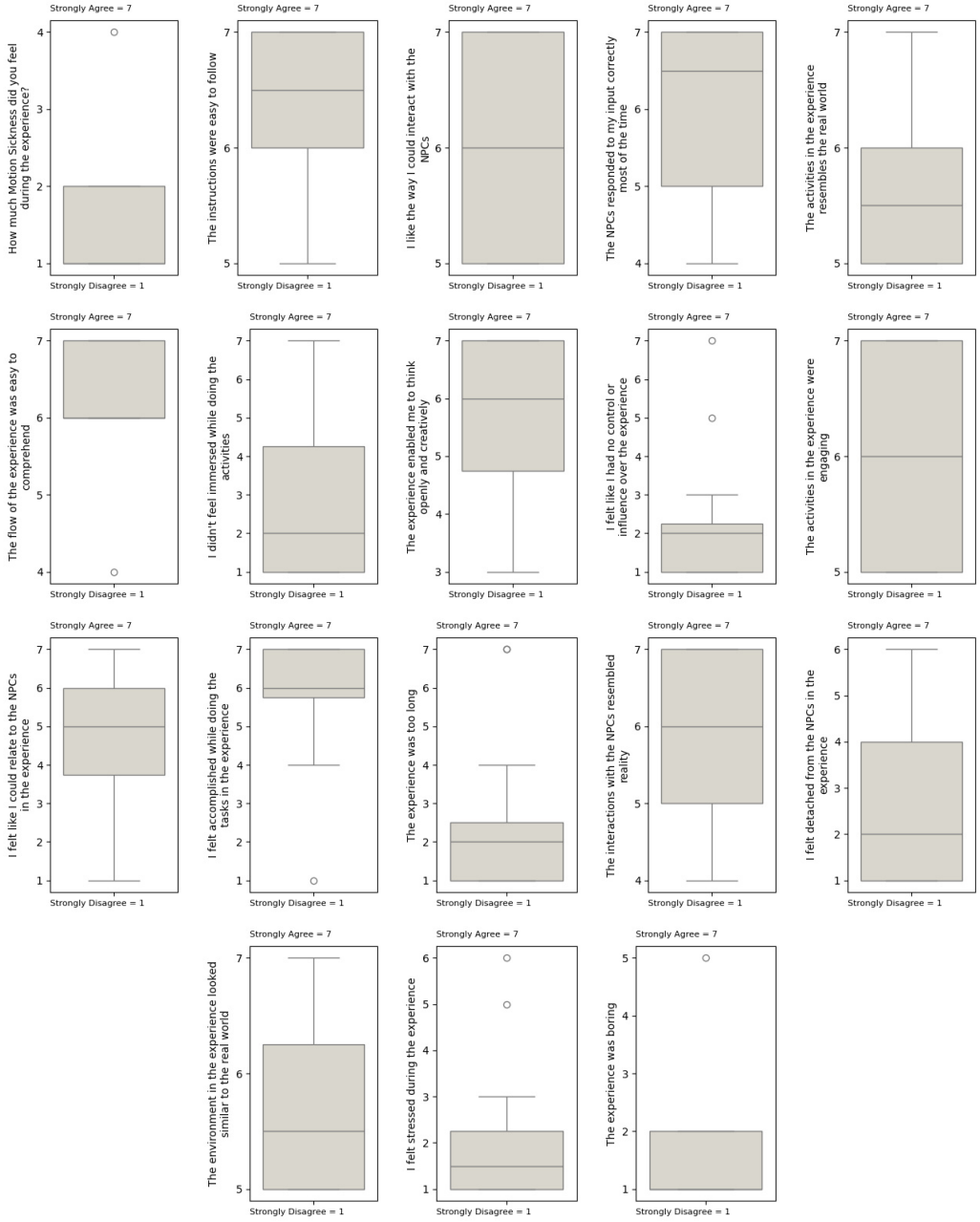


Figure A.5.: General Questionnaire Mean, CI and KDE Charts - STT

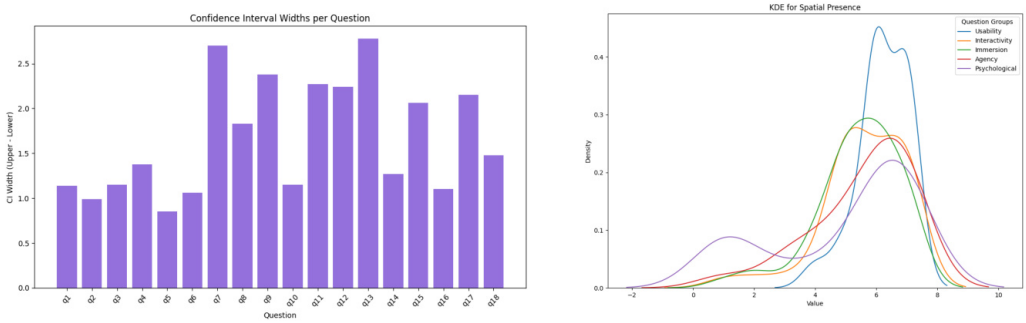
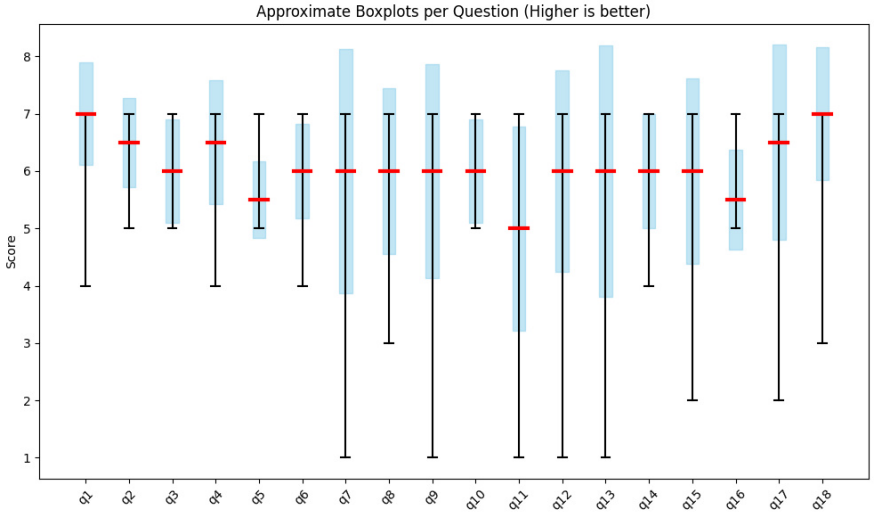
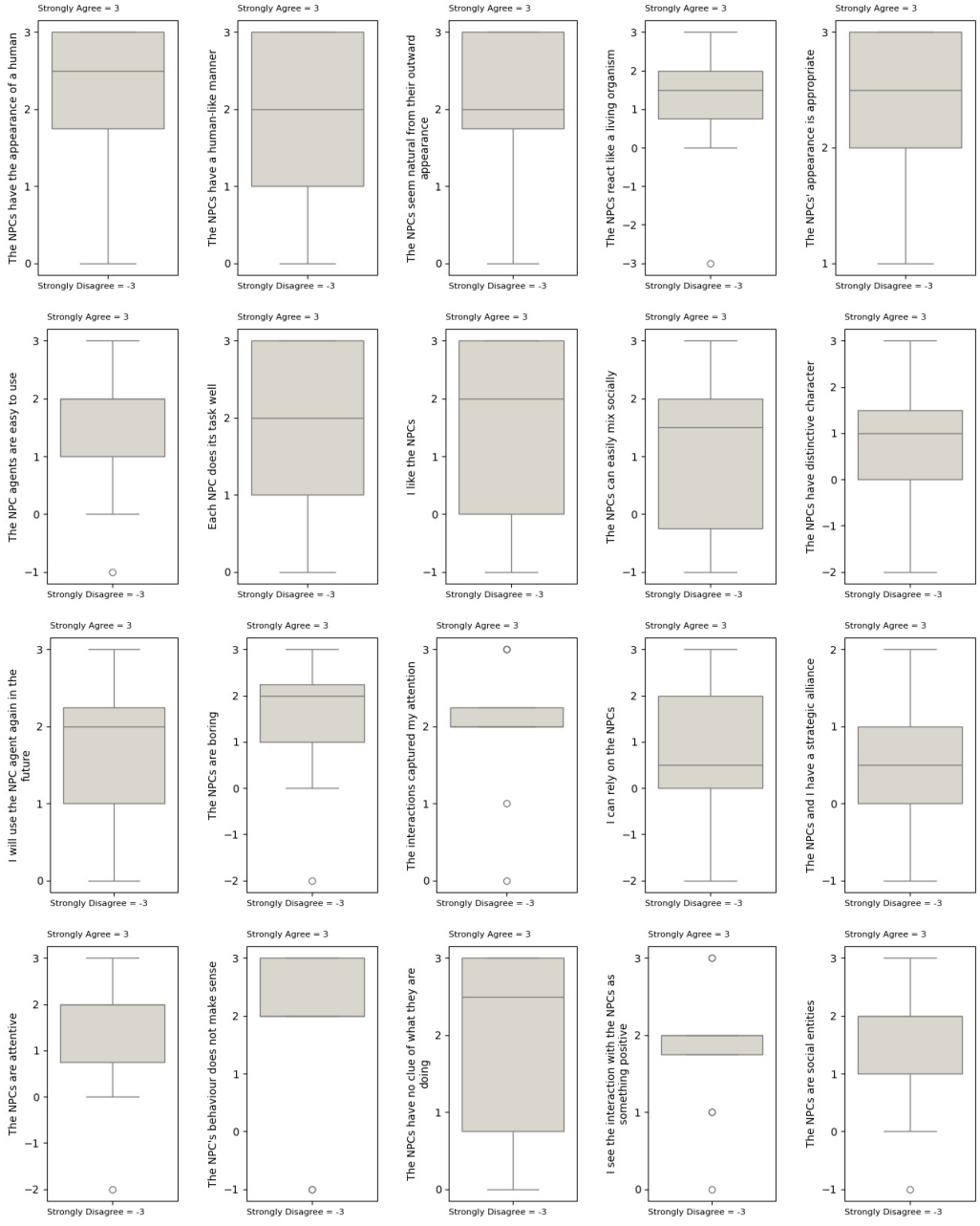


Figure A.6.: General Questionnaire boxplots per question - STT



A.0.3. ASAQ: Gesture Recognition Task

Figure A.7.: Participant responses for the ASAQ (Gesture Recognition)



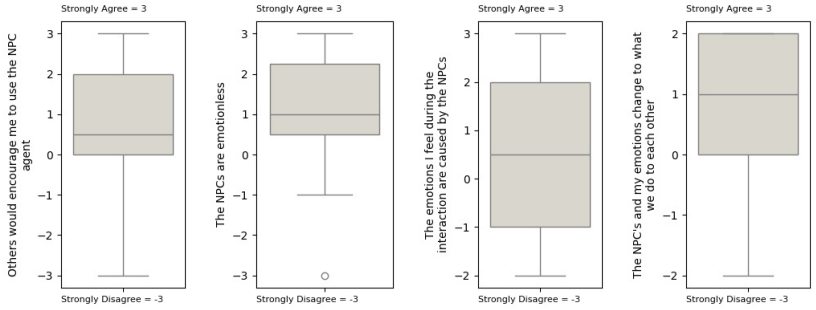
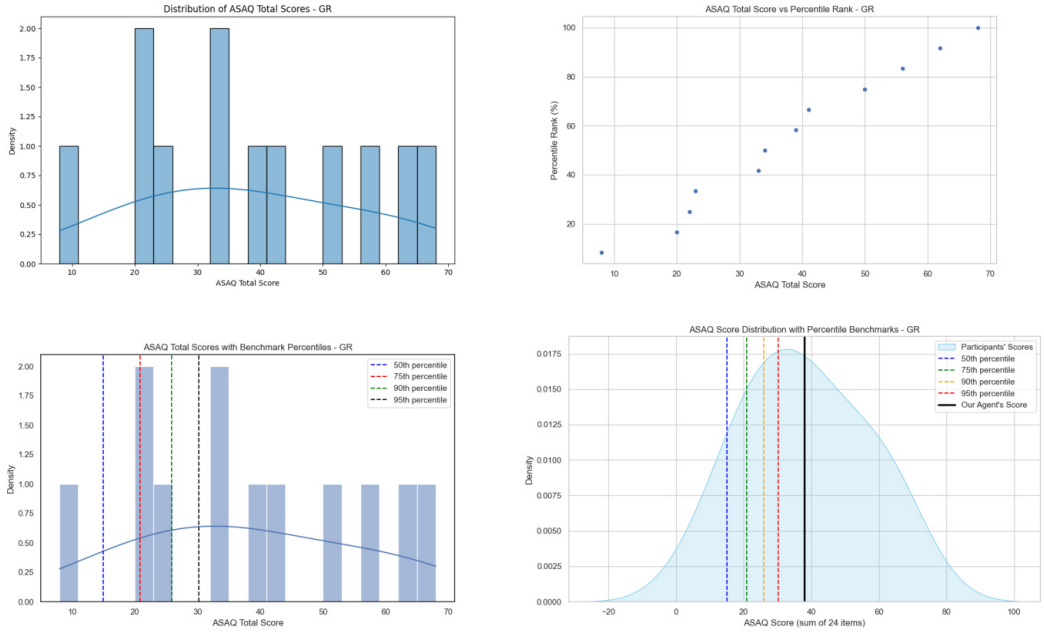
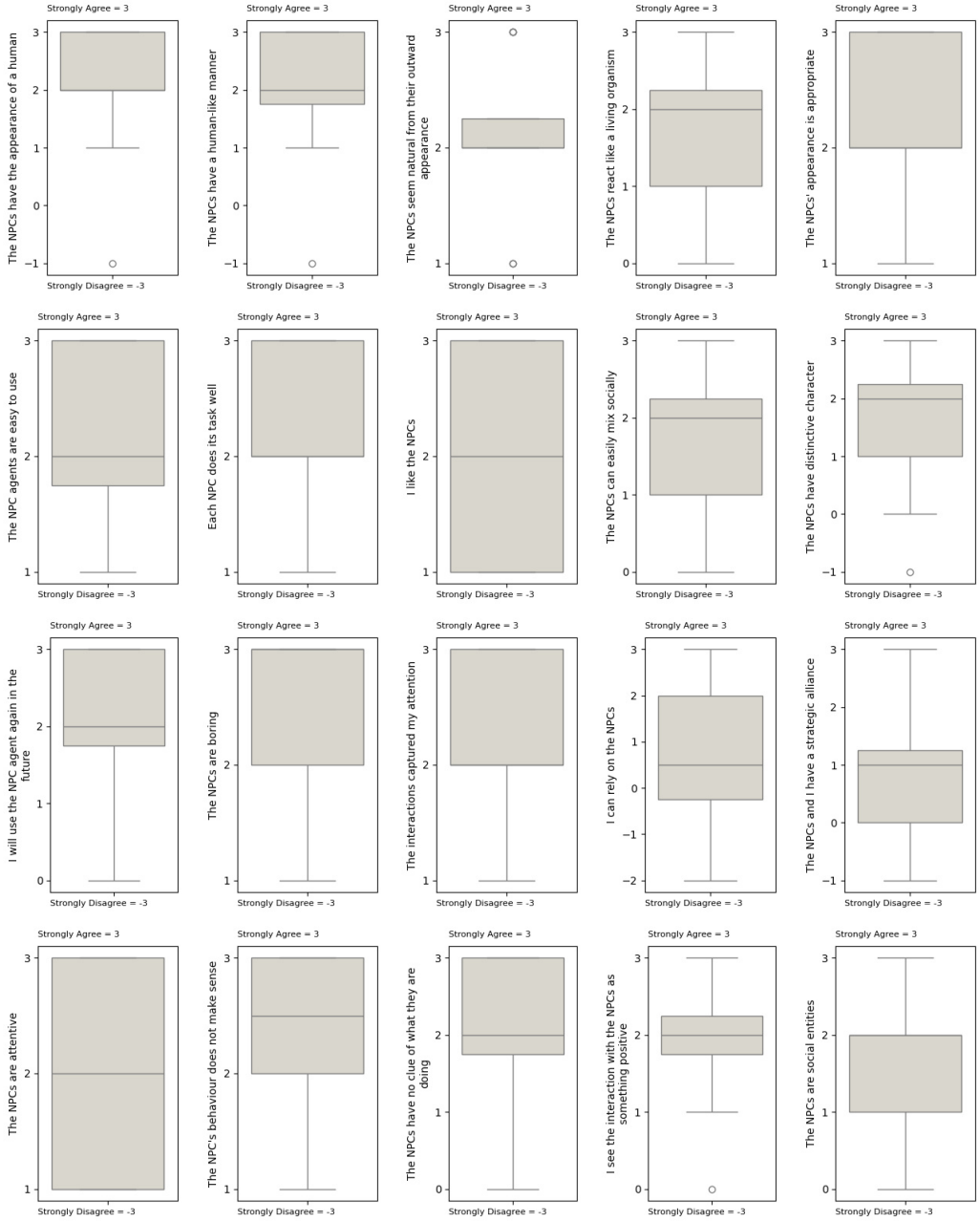


Figure A.8.: ASAQ Agent Score distribution compared to reference percentile scores - GR



A.0.4. ASAQ: Speech To Text Task

Figure A.9.: Participant responses for the ASAQ (Speech To Text)



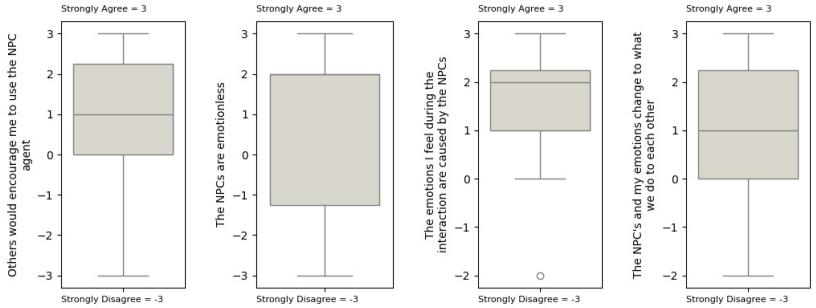
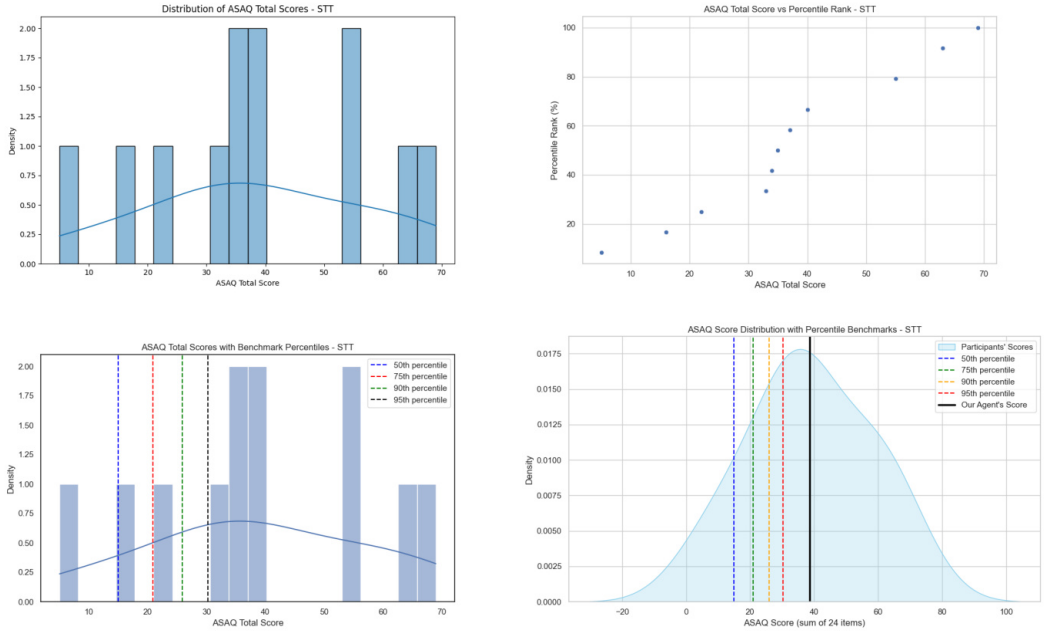


Figure A.10.: ASAQ Agent Score distribution compared to reference percentile scores - STT



A.0.5. SoAS: Gesture Recognition Task

Figure A.11.: Participant responses for the SoAS (Gesture Recognition)

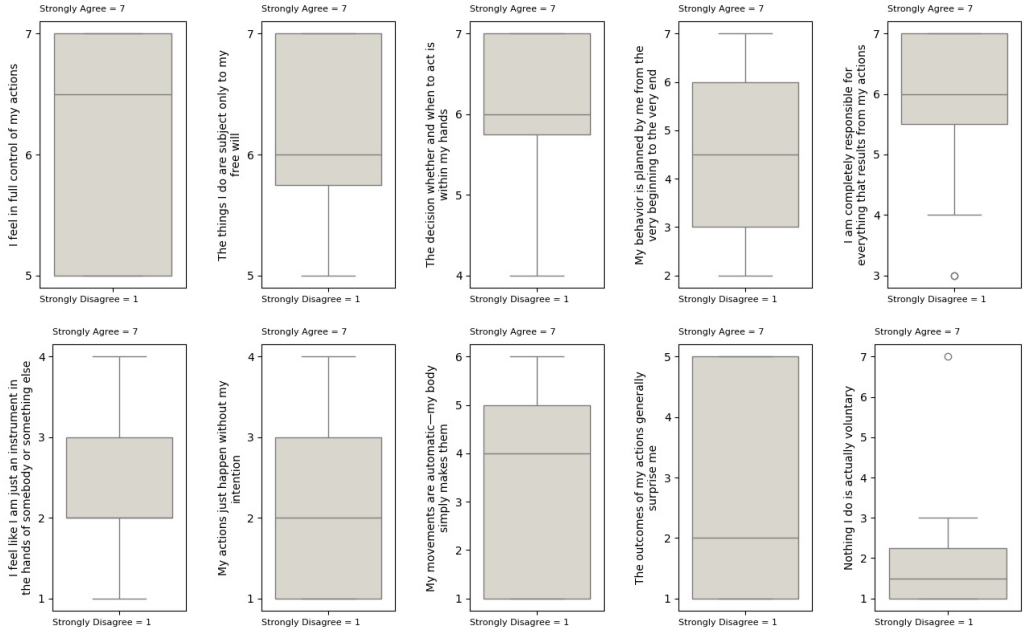
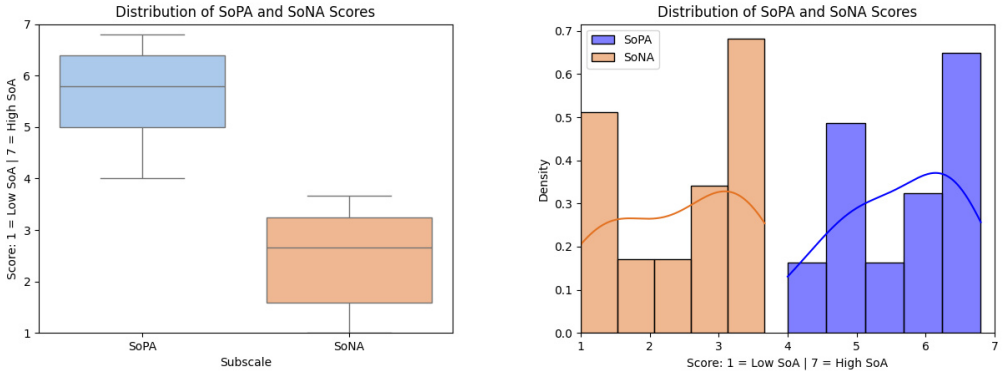


Figure A.12.: SoAS SoPA and SoNA Scores - GR



A.0.6. SoAS: Speech To Text Task

Figure A.13.: Participant responses for the ASAQ (Speech To Text)

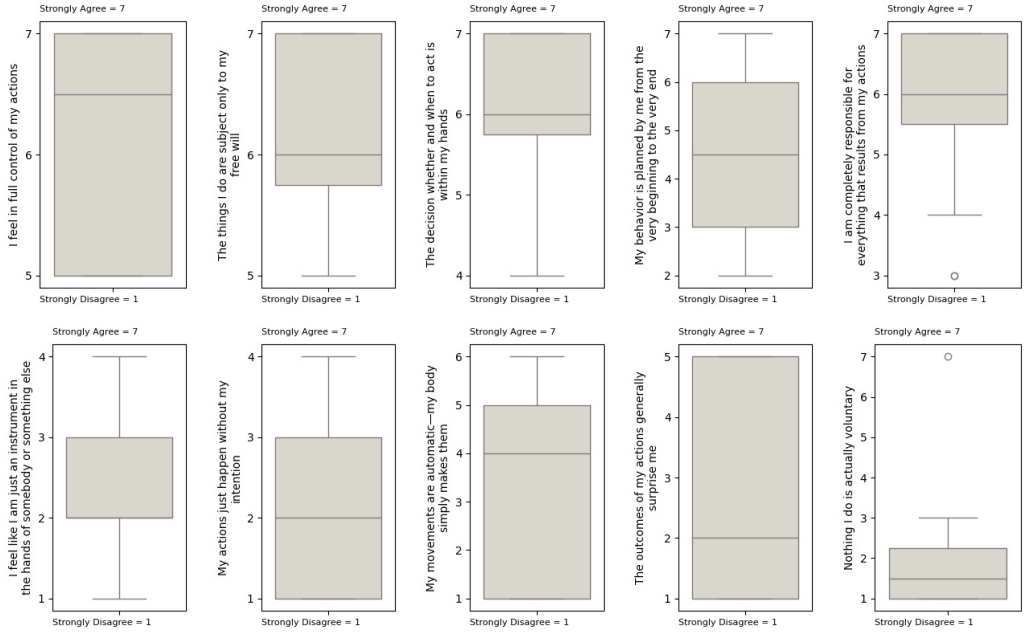
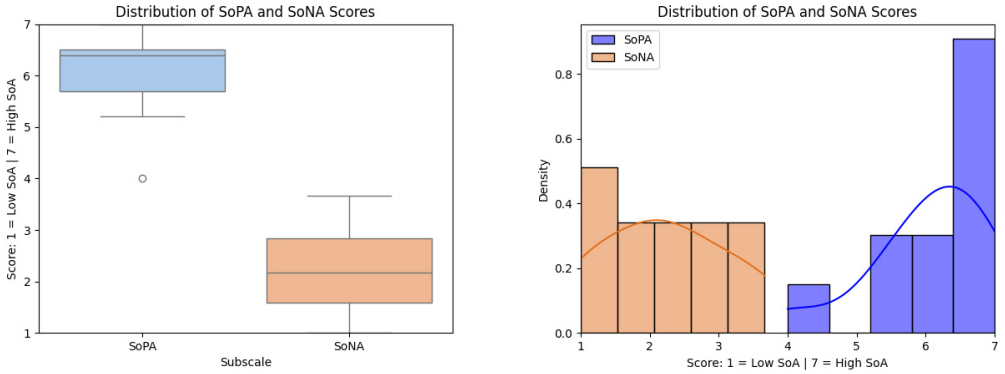


Figure A.14.: SoAS SoPA and SoNA Scores - STT



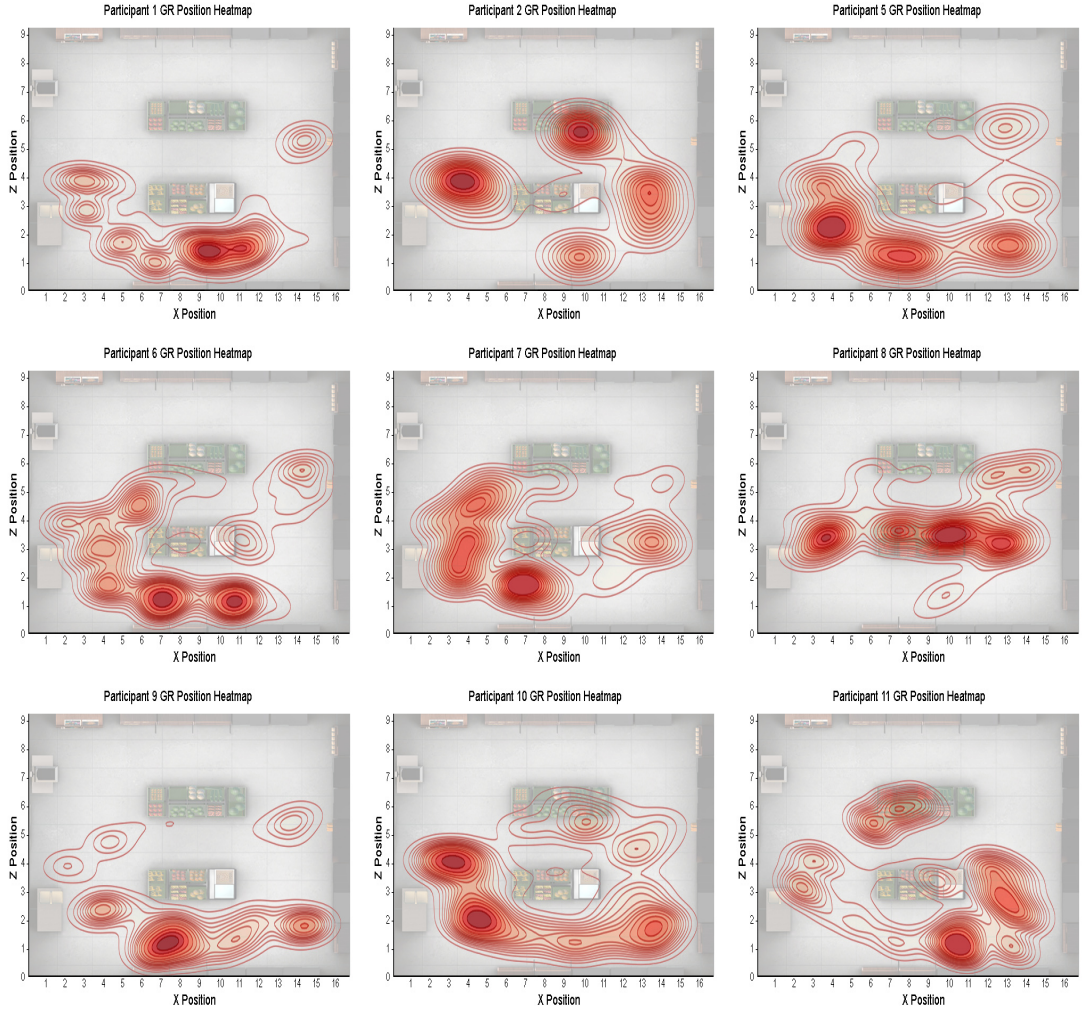
A.0.7. MEC-Spatial Presence Questionnaire

Figure A.15.: Participant responses for the MEC-SPQ

B. Appendix B: Movement Heat-map Data

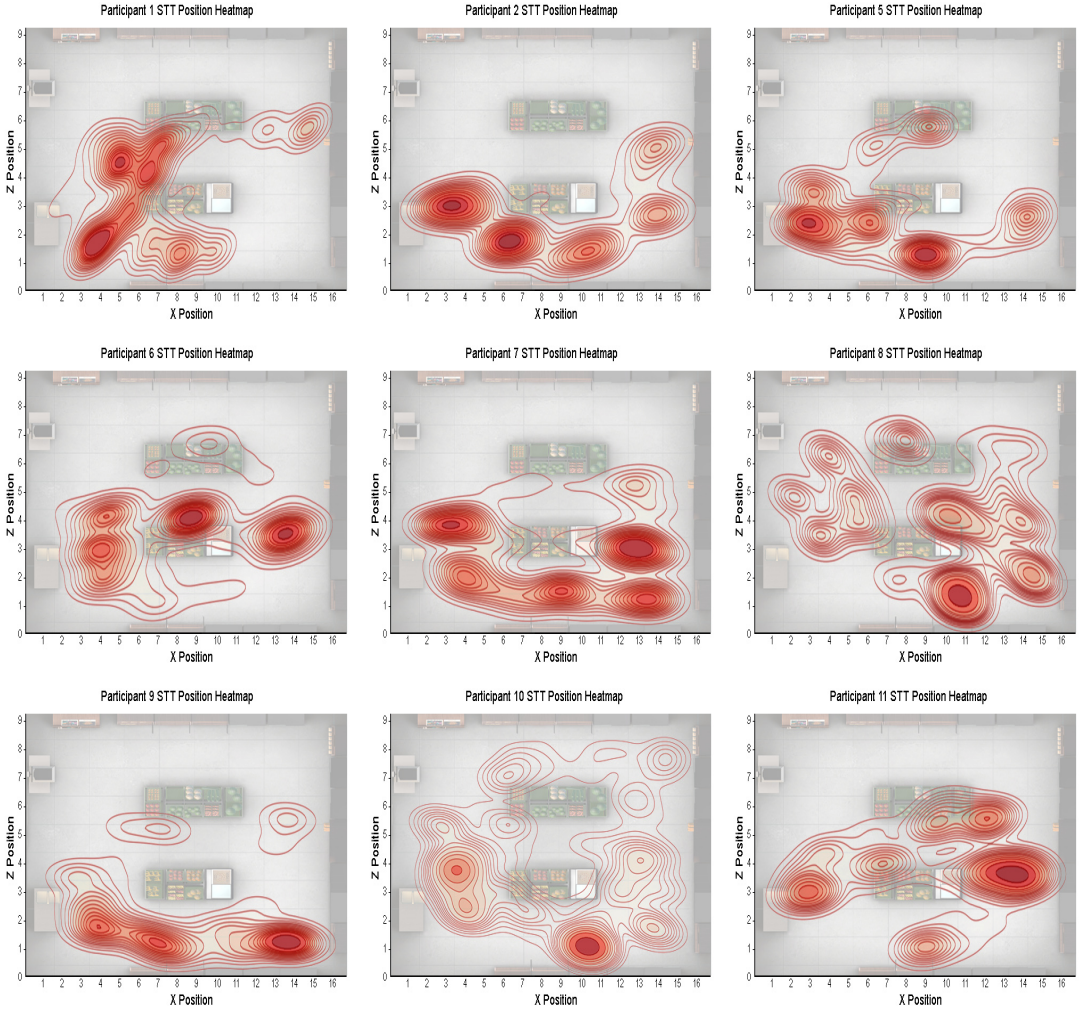
B.1. Gesture Recognition Task

Figure B.1.: Player position heatmaps for the gesture recognition based task



B.2. Speech To Text Task

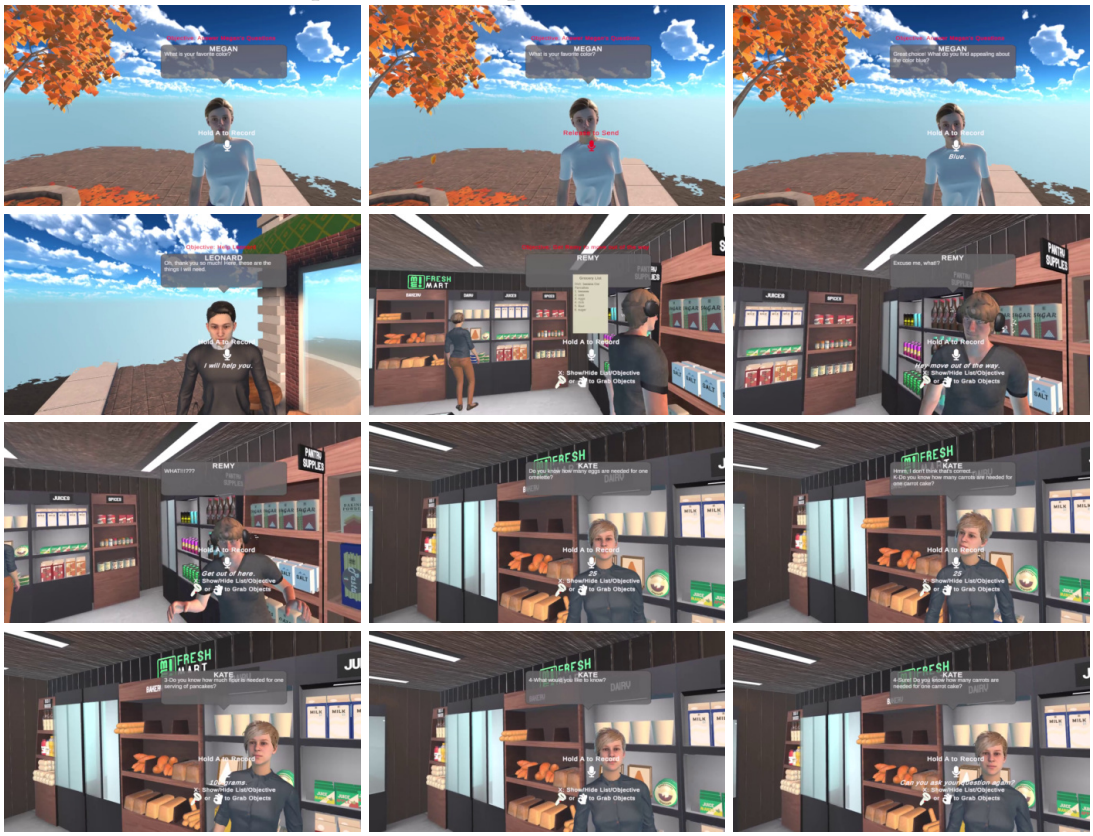
Figure B.2.: Player position heatmaps for the speech to text based task



C. Appendix C: Images from the VR Experience

C.1. Speech To Text Task

Figure C.1.: Screenshots from the VR experience ordered by sequence of events and interaction options for the speech to text based task



C.2. Gesture Recognition Task

Figure C.2.: Screenshots from the VR experience ordered by sequence of events and interaction options for the gesture recognition based task

D. Appendix D: Additional Data

D.1. NPC Animation Graphs

Figure D.1.: Animation Graphs for the NPC **Left:** Megan and **Right:** Leonard

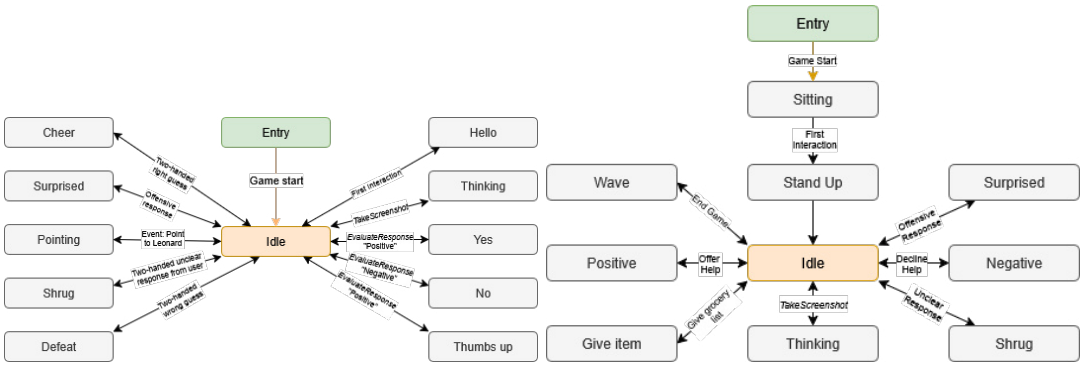
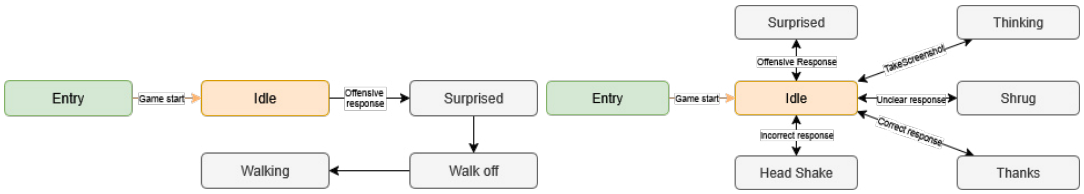


Figure D.2.: Animation Graphs for the NPC **Left:** Remy and **Right:** Kate



D.2. NPC Unique Questions

Table D.1.: Unique GR pose questions provided by the LLM when prompted: *"Can you ask me a question that can be objectively answered with common one hand poses or gestures? Ask questions like 'What gesture signifies X' or 'Show the gesture for X'. Max 10 words"*

No.	Asking for Pose
1	Peace
2	Stop
3	Good luck
4	Okay
5	Heart
6	W
7	M
8	Bird
9	Approval
10	Silence
11	Calling over
12	Moment to think
13	Butterfly
14	Thumbs up
15	I love you in sign language
15	Fist

Table D.2.: Unique STT General questions provided by the LLM when prompted: *"Ask me a simple question that can be answered in a few words. Max 10 words"*

No.	Topic / Question Prompt
1	Favourite Season
2	Favourite way to spend the weekend
3	Favourite movie
4	Favourite colour

D.3. Shop Item Labels

Figure D.3.: Item labels and Signage for the VR Shop scene

D.4. VR Invitation Poster

Figure D.4.: Poster designed to call for participants (Unused)

Abbreviations

NPC Non-Player Character

LLM Large Language Models

HMD Head-Mounted Display

AGI Artificial General Intelligence

AI Artificial Intelligence

GPT Generative Pre-trained Transformers

VR Virtual Reality

GDPR General Data Protection Regulation

EU European Union

ASAQ Artificial Social Agent Questionnaire

IPQ Igroup Presence Questionnaire

IMI Intrinsic Motivation Inventory

HRI Human-Robot Interaction

MEC-SPQ Measurement, Effects, Conditions - Spatial Presence Questionnaire

VE Virtual Environment

SoAS Sense of Agency Scale

SoPA Sense of Positive Agency

SoNA Sense of Negative Agency

ASR Automatic Speech Recognition

TTS Text-To-Speech

STT Speech-To-Text

CI Confidence Interval

FoV Field of View

RAG Retrieval Augmented Generation

CoT Chain-of-Thought

RNN Recursive Neural Networks

LSTM Long Short-Term Memory

List of Figures

2.1. VR Head and Hand Tracking 8

4.1. Inside and Outside Scenes 21
4.2. Shop Items 22
4.3. UI Design Evolution 24
4.4. Simplified Quest Flow 25
4.5. Early Approach of Using Joint Coordinates 26
4.6. Early Approach Image Recognition Results 27
4.7. Interaction with Megan in STT 28
4.8. Interaction with Megan in GR 29
4.9. Detailed Quest Flow 30
4.10. NPC System prompt 31
4.11. The Grocery List 32
4.12. Heatmap and Overlay 33
4.13. Query Flow 1 35
4.14. Query Flow 2 36
4.15. The VR Test Setup 38

5.1. General Questionnaire - Mean and KDE Charts 55
5.2. ASAQ - Percentile Comparisons 57
5.3. NPC System prompt 58
5.4. Spatial Presence - Mean Score Comparison 59
5.5. Spatial Presence - KDE Plot for Questionnaire 59

6.1. User responses for NPCs Human Likeness 66
6.2. User responses for NPCs Response Accuracy 66
6.3. User responses for Control and Comfort 67
6.4. User responses for NPC Reliability 68

A.1. General Questionnaire GR 73
A.2. GR Correlation Heatmap 74
A.3. GR CI Interval Chart 74
A.4. General Questionnaire STT 75

A.5. STT Mean and CI Chart 76
A.6. STT Boxplots 76
A.7. ASAQ GR 77
A.8. ASAQ Score - GR 78
A.9. ASAQ STT 79
A.10.ASAQ Score - STT 80
A.11.SoAS GR 81
A.12.SoAS GR SoPA and SoNA Scores 81
A.13.SoAS STT 82
A.14.SoAS STT SoPA and SoNA Scores 82
A.15.SPQ GR 83

B.1. Heatmap GR 84
B.2. Heatmap STT 85

C.1. Screenshots STT 86
C.2. Screenshots GR 87

D.1. Animation Graph Megan and Leonard 88
D.2. Animation Graph Remy and Kate 88
D.3. Shop Item Labels 90
D.4. VR Invitation Poster 91

List of Tables

3.1. Literature Overview 17

4.1. Test Cases 34
4.2. Questionnaire Structure 45
4.3. General Questionnaire 46
4.4. Interview Questions 47

5.1. Task Completion Times 49
5.2. Confusion Matrix Summary for GR Task 50
5.3. GR Response Breakdown 50
5.4. Average Response Times 52
5.5. Description of Interaction Counts 53
5.6. Objective Completion Time 54
5.7. General Questionnaire Alpha Scores 54
5.8. Spatial Presence Questionnaire Alpha Scores 54
5.9. SoAS - Summary Statistics 56
5.10. General Questionnaire - Summary Statistics 63
5.11. ASAQ - Summary Statistics 64

D.1. GR Unique Questions 89
D.2. STT Unique Questions 89

Bibliography

Adhanom, I. B., P. MacNeilage, and E. Folmer (2023). "Eye Tracking in Virtual Reality: a Broad Review of Applications and Challenges." In: *Virtual Reality* 27, pp. 1481–1505. DOI: 10.1007/s10055-022-00738-z.

Adobe Inc. (2025a). *Adobe Illustrator*. Version 2020.

– (2025b). *Adobe Substance 3D Painter*. Version 2019.

– (2025c). *Mixamo*. Web-based 3D character animation and rigging service.

Aher, G. V., R. I. Arriaga, and A. T. Kalai (2023). "Using large language models to simulate multiple humans and replicate human subject studies." In: *International Conference on Machine Learning*. PMLR, pp. 337–371.

Alanko, S. (2023). *Comparing Inside-out and Outside-in Tracking in Virtual Reality*. Bachelor's Thesis.

Ali, H., P. Allgeuer, and S. Wermter (Apr. 2024). *Comparing apples to oranges: LLM-powered multimodal intention prediction in an object categorization task*. arXiv. Preprint, Version 3. DOI: 10.48550/arXiv.2404.08424.

Blender Foundation (2025). *Blender*.

Brito, I. A., J. S. Dollis, F. B. Färber, P. S. F. B. Ribeiro, R. T. Sousa, and A. R. G. Filho (Feb. 2025). "Integrating Personality into Digital Humans: A Review of LLM-Driven Approaches for Virtual Reality." In: *arXiv preprint arXiv:2503.16457v1*.

Brynjolfsson, E., D. Li, and L. R. Raymond (2023). *Generative AI at Work: The Impact of a Generative AI-Based Conversational Assistant on Customer Support Agent Productivity*. Working Paper 31161. Revised November 2023. National Bureau of Economic Research. DOI: 10.3386/w31161.

Buckingham, G. (2021). "Hand Tracking for Immersive Virtual Reality: Opportunities and Challenges." In: *Frontiers in Virtual Reality* 2, p. 728461. DOI: 10.3389/frvir.2021.728461.

Buldu, K. B., S. Özdel, K. H. C. Lau, M. Wang, D. Saad, S. Schönborn, A. Boch, E. Kasneci, and E. Bozkir (2025). "CUIfy the XR: An Open-Source Package to Embed LLM-Powered Conversational Agents in XR." In: *arXiv preprint arXiv:2411.04671v2*. Version 2, submitted 27 Feb 2025.

Clay, V., P. König, and S. König (2019). "Eye Tracking in Virtual Reality." In: *Journal of Eye Movement Research* 12.1, p. 3. DOI: 10.16910/jemr.12.1.3.

Devlin, J., M.-W. Chang, K. Lee, and K. Toutanova (June 2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.

ElevenLabs Inc. (2025). *ElevenLabs: AI Voice Generation Platform*. AI-driven text-to-speech and voice cloning platform offering lifelike speech synthesis in over 70 languages.

Elman, J. L. (1990). "Finding structure in time." In: *Cognitive Science* 14.2, pp. 179–211. DOI: 10.1207/s15516709cog1402_1.

Fitrianie, S., M. Bruijnes, A. Abdulrahman, and W.-P. Brinkman (2025). "The Artificial Social Agent Questionnaire (ASAQ) — Development and evaluation of a validated instrument for capturing human interaction experiences with artificial social agents." In: *International Journal of Human-Computer Studies* 199, p. 103482. DOI: 10.1016/j.ijhcs.2025.103482.

Free3D (2025). *Free3D*. A platform offering free and premium 3D models in various formats including .obj, .fbx, .blend, .max, .ma, and more.

Gnewuch, U., S. Morana, M. T. P. Adam, and A. Maedche (2022). "Opposing Effects of Response Time in Human–Chatbot Interaction: The Moderating Role of Prior Experience." In: *Business & Information Systems Engineering* 64.6, pp. 773–791. DOI: 10.1007/s12599-022-00755-x.

Hart, S. G. and L. E. Staveland (1988). "Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research." In: *Advances in Psychology*. Vol. 52. North-Holland, pp. 139–183.

Hochreiter, S. and J. Schmidhuber (1997). "Long short-term memory." In: *Neural Computation* 9.8, pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

HTC Corporation (2016). *HTC Vive*. https://www.vive.com/. Virtual reality headset.

Inc., H. R. (2024). *Stretch 3 Mobile Manipulator*. Mobile Manipulator Robot. Lightweight, ROS-based mobile manipulator used in research and developer contexts, featuring wrist-mounted 3D camera, dexterous gripper, ROS 2, Python SDK for perception/-navigation and LLM agents :contentReference[oaicite:2]index=2.

Kerzel, M., P. Allgeuer, E. Strahl, N. Frick, J. G. Habekost, M. Eppe, and S. Wermter (2023). "NICOL: A neuro-inspired collaborative semi-humanoid robot that bridges social interaction and reliable manipulation." In: *IEEE Access* 11, pp. 123531–123542. DOI: 10.1109/ACCESS.2023.332937.

Kobzarev, O., A. Lykov, and D. Tsetserukou (2025). *GestLLM: Advanced Hand Gesture Interpretation via Large Language Models for Human-Robot Interaction*. Submitted on 13 January 2025 (v1); revised on 14 January 2025 (v2). arXiv: 2501.07295v2 [cs.RO].

Konenkov, M., A. Lykov, D. Trinitatova, and D. Tsetserukou (May 2024). *VR-GPT: Visual language model for intelligent virtual reality applications*. arXiv. Version 1. Preprint. DOI: 10.48550/arXiv.2405.11537.

Lee, J., J. Wang, E. Brown, L. Chu, S. S. Rodriguez, and J. E. Froehlich (2024). "GazePointAR: A Context-Aware Multimodal Voice Assistant for Pronoun Disambiguation in Wearable Augmented Reality." In: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. CHI '24. Honolulu, HI, USA: Association for Computing Machinery. ISBN: 9798400703300. DOI: 10.1145/3613904.3642230.

Li, Z., H. Zhang, C. Peng, and R. L. Peiris (2025). "Exploring Large Language Model–Driven Agents for Environment-Aware Spatial Interactions and Conversations in Virtual Reality Role-Play Scenarios." In: *Proceedings of the IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, pp. 1–11. DOI: 10.1109/VR59515.2025.00025.

Liang, P. P., A. Zadeh, and L.-P. Morency (2022). *Foundations and Trends in Multimodal Machine Learning: Principles, Challenges, and Open Questions*. arXiv preprint arXiv:2209.03430. Revised Feb 20, 2023 (v2).

Lim, S., R. Schmälzle, and G. Bente (June 2025). "Artificial social influence via human-embodied AI agent interaction in immersive virtual reality (VR): Effects of similarity-matching during health conversations." In: *Computers in Human Behavior* 165, p. 100172. DOI: 10.1016/j.chbah.2025.100172.

Liu, J. (2024). "ChatGPT: Perspectives from Human–Computer Interaction and Psychology." In: *Frontiers in Artificial Intelligence* 7, p. 1418869. DOI: 10.3389/frai.2024.1418869.

Lugaresi, C., J. Tang, H. Nash, C. McGuire, F. Lee, W. Chang, M. Yong, F. Tong, J. Etienne, and M. Grundmann (2019). "MediaPipe: A Framework for Building Perception Pipelines." In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, pp. 0–0.

Maslych, M., C. Pumarada, A. Ghasemaghaei, and J. J. J. LaViola (Jan. 2025). *Takeaways from Applying LLM Capabilities to Multiple Conversational Avatars in a VR Pilot Study*. arXiv preprint arXiv:2501.00168v2. Revised version (v2) submitted Jan 6, 2025. DOI: 10.48550/arXiv.2501.00168.

Matsumoto, D. and H. C. Hwang (2013). "Cultural similarities and differences in emblematic gestures." In: *Journal of Nonverbal Behavior* 37, pp. 1–27.

Meng, X., R. Du, and A. Varshney (2020). "Eye-dominance-guided Foveated Rendering." In: *IEEE Transactions on Visualization and Computer Graphics* 26.5, pp. 1972–1980. DOI: 10.1109/TVCG.2020.2973442.

Meta Platforms, Inc. (2020). *Oculus Quest 2*. https://www.meta.com/quest/products/quest-2/. Standalone VR headset.

Microsoft (2019). *Microsoft Azure Kinect DK*. Developer Kit. Spatialcomputing developer kit with AI-sensors including depth, RGB, IMU, and microphone array; used in

research for body tracking, mapping, telepresence, healthcare, and robotics :contentReference[oaicite:1]index=1.

Microsoft Corporation (n.d.). *Azure Speech to Text*. https://azure.microsoft.com/en-us/products/cognitive-services/speech-to-text. Accessed: 2025-07-19.

Naidu, S., E. Smith, C. Hagood, A. Rolly, S. Sarker, C. Hayes, and T. Iqbal (May 2025). "A Data Capture and Gesture Recognition System to Enable Human-Robot Collaboration." In: pp. 375–380. DOI: 10.1109/SIEDS65500.2025.11021158.

Naveed, H., A. U. Khan, S. Qiu, M. Saqib, S. Anwar, M. Usman, N. Akhtar, N. Barnes, and A. Mian (2023). "A Comprehensive Overview of Large Language Models." In: *arXiv preprint arXiv:2307.06435*.

Nicolescu, L. and M. T. Tudorache (2022). "Human–Computer Interaction in Customer Service: The Experience with AI Chatbots—A Systematic Literature Review." In: *Electronics* 11.10, p. 1579. DOI: 10.3390/electronics11101579.

OpenAI (2022). *Whisper: General-purpose speech recognition model*. https://github.com/openai/whisper. Accessed: 2025-07-19.

– (2023). *GPT-4*. Large language model.

– (2024). *Guide to OpenAI's Vision API*. https://platform.openai.com/docs/guides/vision. Accessed: 2025-07-19.

Pan, M., A. Kitson, H. Wan, and M. Prpa (Oct. 2024). *ELLMA-T: An embodied LLM-agent for supporting English language learning in social VR*. arXiv. Version 1. Preprint. DOI: 10.48550/arXiv.2410.02406.

Pang, H., T. Ding, L. He, and Q. Gan (2024). *LLM Gesticulator: Leveraging Large Language Models for Scalable and Controllable Co-Speech Gesture Synthesis*. Submitted on 6 October 2024 (version v1). arXiv: 2410.10851v1 [cs.GR].

Pixabay (n.d.). *Pixabay: Free Images, Videos, Music and Sound Effects*. Accessed: 2025-07-20.

Qi, Z., H. Li, H. Qin, K. Peng, S. He, and X. Qin (Jan. 2025). *Harnessing large language model for virtual reality exploration testing: A case study*. arXiv. Version 1. Preprint. DOI: 10.48550/arXiv.2501.05625.

Qwen Team (2024). *Introducing Qwen1.5*. Blog post and Hugging Face / ModelScope release. Alibaba Cloud – improved version of Qwen (sizes 0.5B to72B) released Feb/Apr 2024.

Radford, A., J. W. Gao, Y. Tao, W. Han, D. Nathani, R. Pinkney, P. Dhariwal, D. Amodei, J. Knight, and I. Sutskever (2022). *Robust Speech Recognition via Large-Scale Weak Supervision*. arXiv preprint arXiv:2212.04356. Version 2, December 2022.

Radford, A., K. Narasimhan, T. Salimans, and I. Sutskever (June 2018). "Improving Language Understanding by Generative Pre-Training." In: *OpenAI Technical Report*. Preprint introducing GPT-1.

Ramesh, A., M. Pavlov, G. Goh, S. Gray, C. Voss, A. Chen, and I. Sutskever (2021). "Zero-Shot Text-to-Image Generation." In: *arXiv preprint arXiv:2102.12092*. DOI: 10.48550/arXiv.2102.12092.

Rasch, J., J. Töws, T. Hirzle, F. Müller, and M. Schmitz (2025). "CreepyCoCreator? Investigating AI Representation Modes for 3D Object Co-Creation in Virtual Reality." In: *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. ACM, p. 14. DOI: 10.1145/3706598.3713720.

Roberts, J., A. Banburski, and J. Lanier (2022). "Surreal VR Pong: LLM approach to game design." In: *Proceedings of the Machine Learning for Creativity and Design Workshop (NeurIPS 2022)*.

Rodriguez, J. A., N. Botzer, D. Vázquez, C. Pal, M. Pedersoli, and I. Laradji (2024). "IntentGPT: Few-shot Intent Discovery with Large Language Models." In: *arXiv preprint arXiv:2411.10670*.

Ryan, R. M. and E. L. Deci (2022). *Intrinsic Motivation Inventory (IMI)*. https://selfdeterminationtheory.org/wp-content/uploads/2022/02/IMI_Complete.pdf. Accessed: 2025-07-19.

Sánchez-Vives, M. V. and M. Slater (2005). "From Presence to Consciousness through Virtual Reality." In: *Nature Reviews Neuroscience* 6.4, pp. 332–339. DOI: 10.1038/nrn1651.

Schaefer, K. E. (2016). "Measuring Trust in Human Robot Interactions: Development of the Trust Perception Scale-HRI." In: *Robust Intelligence and Trust in Autonomous Systems*. Springer, pp. 191–218. DOI: 10.1007/978-1-4899-7668-0_10.

Schubert, T., A. Friedmann, and H. J. Regenbrecht (2001). "The Igroup Presence Questionnaire (IPQ): A Measure to Assess Presence in Virtual Environments." In: *Proceedings of the 4th International Workshop on Presence*, pp. 39–42.

Schulhoff, S., M. Ilie, N. Balepur, K. Kahadze, A. Liu, C. Si, Y. Li, A. Gupta, H. Han, S. Schulhoff, P. S. Dulepet, S. Vidyadhara, D. Ki, S. Agrawal, C. Pham, G. Kroiz, F. Li, H. Tao, A. Srivastava, H. D. Costa, S. Gupta, M. L. Rogers, I. Goncearenco, G. Sarli, I. Galynker, D. Peskoff, M. Carpuat, J. White, S. Anadkat, A. Hoyle, and P. Resnik (2024). *The Prompt Report: A Systematic Survey of Prompting Techniques*. arXiv preprint arXiv:2406.06608. Accessed: 2025-07-19.

Sennrich, R., B. Haddow, and A. Birch (2015). "Neural Machine Translation of Rare Words with Subword Units." In: *CoRR* abs/1508.07909. DOI: 10.48550/arXiv.1508.07909.

Shi, Y. and B. Deng (2024). "Finding the Sweet Spot: Exploring the Optimal Communication Delay for AI Feedback Tools." In: *Information Processing & Management* 61.2, p. 103572. DOI: 10.1016/j.ipm.2023.103572.

Shoa, A. and D. Friedman (June 2025). "Milo: an LLM-based virtual human open-source platform for extended reality." In: *Frontiers in Virtual Reality* 6. Received: 03 January

2025; Accepted: 25 April 2025; Published: 05 June 2025. DOI: 10.3389/frvir.2025.1555173.

Slater, M. (2009). "Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments." In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 364.1535, pp. 3549–3557. DOI: 10.1098/rstb.2009.0138.

– (2018). "Immersion and the illusion of presence in virtual reality." In: *British Journal of Psychology* 109.3, pp. 431–433. DOI: 10.1111/bjop.12305.

Song, Y., K. Wu, and J. Ding (2024). "Developing an immersive game-based learning platform with generative artificial intelligence and virtual reality technologies: LearningverseVR." In: *Computers & Education: X Reality* 4, Article 100069. DOI: 10.1016/j.cexr.2024.100069.

Tapal, A., E. Oren, R. Dar, and B. Eitam (2017). "The Sense of Agency Scale: A Measure of Consciously Perceived Control over One's Mind, Body, and the Immediate Environment." In: *Frontiers in Psychology* 8, p. 1552. DOI: 10.3389/fpsyg.2017.01552.

Technologies, U. (2023). *Hand Data Model*. Accessed: 2025-07-20.

Torre, F. D. L., C. M. Fang, H. Huang, A. Banburski-Fahey, J. A. Fernandez, and J. Lanier (2024). "LLMR: Real-time Prompting of Interactive Worlds using Large Language Models." In: *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24)*. ACM, pp. 1–13. DOI: 10.1145/3613904.3642579.

Triyono, M. B., A. A. Rafiq, D. Hariyanto, D. Adinda, and M. Denami (2024). "In-World NPC: Analysing Artificial Intelligence Precision in Virtual Reality Settings." In: *International Journal of Online and Biomedical Engineering (iJOE)* 20.15, —. DOI: 10.3991/ijoe.v20i15.51437.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin (2017). "Attention Is All You Need." In: *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*. Curran Associates Inc., pp. 6000–6010.

Vorderer, P., W. Wirth, F. R. Gouveia, F. Biocca, T. Saari, L. Jäncke, S. Böcking, H. Schramm, A. Gysbers, T. Hartmann, C. Klimmt, J. Laarni, N. Ravaja, A. Sacau, T. Baumgartner, and P. Jäncke (2004). *MEC Spatial Presence Questionnaire (MEC-SPQ): Short Documentation and Instructions for Application*. Tech. rep. Project Presence: MEC (IST-2001-37661). DOI: 10.13140/RG.2.2.26232.42249.

Vox, J. P., A. Weber, K. I. Wolf, K. Izdebski, T. Schüler, P. König, F. Wallhoff, and D. Friemert (2021). "An Evaluation of Motion Trackers with Virtual Reality Sensor Technology in Comparison to a Marker-Based Motion Capture System Based on Joint Angles for Ergonomic Risk Assessment." In: *Sensors* 21.9, p. 3145. DOI: 10.3390/s21093145.

VRChat Inc. (n.d.). *VRChat*. https://vrchat.com/. Accessed: 2025-07-19.

VRChatOSC Contributors (n.d.). *VRChatOSC: Open Sound Control Bridge for VRChat*. https://github.com/Neos-Metaverse/VRChatOSC. Accessed: 2025-07-19.

Wan, H., J. Zhang, A. A. Suria, B. Yao, D. Wang, Y. Coady, and M. Prpa (2024). "Building LLM-based AI agents in social virtual reality." In: *Extended Abstracts of the 2024 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery. DOI: 10.1145/3613905.3651026.

Wang, T., A. Roberts, D. Hesslow, T. L. Scao, H. W. Chung, I. Beltagy, J. Launay, and C. Raffel (2022). "What Language Model Architecture and Pretraining Objective Work Best for Zero-Shot Generalization?" In: *arXiv preprint arXiv:2204.05832*. DOI: 10.48550/arXiv.2204.05832.

Wang, Z., H. Huang, Y. Zhao, Z. Zhang, and Z. Zhao (Aug. 2023). *Chat-3D: Data-efficiently tuning large language model for universal dialogue of 3D scenes*. arXiv. Version 1. Preprint. DOI: 10.48550/arXiv.2308.08769.

Wicke, P. (Jan. 2024). *Probing Language Models' Gesture Understanding for Enhanced Human–AI Interaction*. arXiv. arXiv:2401.17858v1 [Computer software].

Zeng, X., X. Wang, T. Zhang, C. Yu, S. Zhao, and Y. Chen (2024). "GestureGPT: Toward zero-shot free-form hand gesture understanding with large language model agents." In: *Proceedings of the ACM on Human-Computer Interaction* 8.ISS, Article 545. DOI: 10.1145/3698145.