International Conference on Artificial Reality and Telexistence
Eurographics Symposium on Virtual Environments (2024)
S. Hasegawa, N. Sakata and V. Sundstedt (Editors)

Conversational Agent for Procedural Building Design in Virtual Reality

M. Bosco, P. Kán and H. Kaufmann

Institute of Visual Computing and Human-Centered Technology, TU Wien, Austria

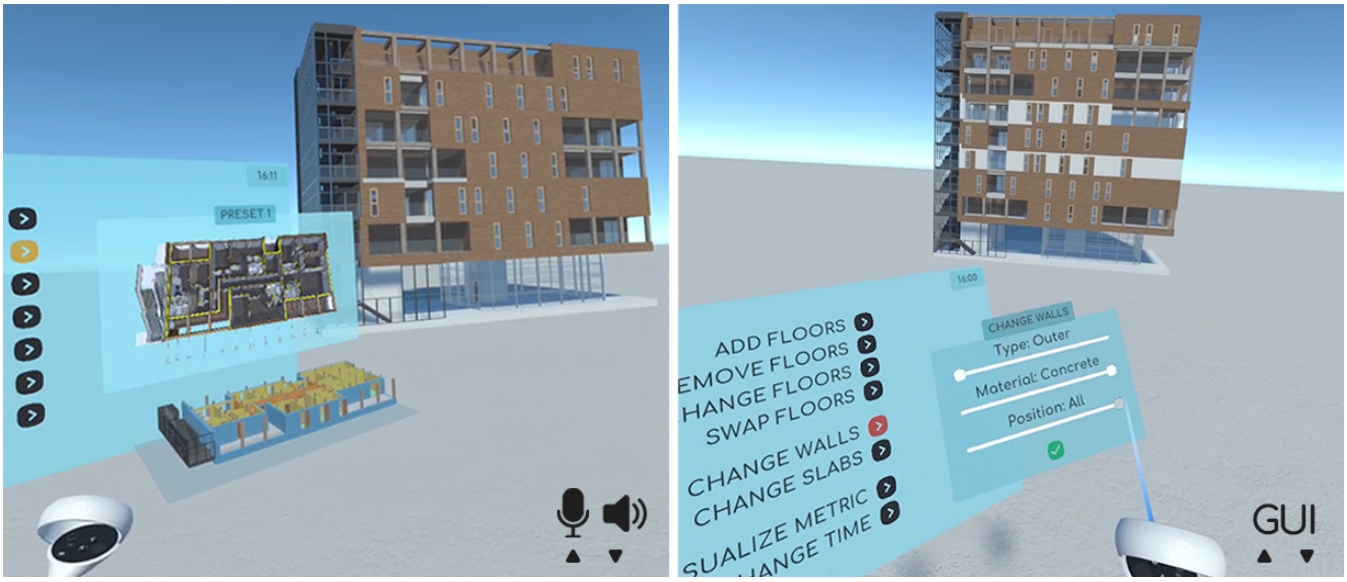

Figure 1: *User's perspective of the virtual environment for procedural building design: interaction via natural language with a non-embodied conversational agent (left) and interaction via graphical user interface (right).*

Abstract

With the emergence of large language models (LLMs), conversational agents have gained significant attention across various domains, including virtual reality (VR). This paper investigates the use of conversational agents as an interface for procedural building design in VR. We propose a voice interface that allows a user to control parameters of procedural generation and gain insights about the building construction metrics through natural conversation. The pipeline introduced for the conversational agent involves utilizing LLMs in two separate API calls for natural language understanding and natural language generation. This separation enables the invocation of various actions in procedural generation as well as meaningful agent responses to building-related questions. Furthermore, we conducted a user study to assess our proposed conversational interface in comparison to a traditional graphical user interface (GUI) in a VR architectural design task focused on circular economy. The study scrutinize the user-reported usability, presence, realism, errors, and effectiveness of both interfaces. Results suggest that while the non-embodied conversational agent enhances effectiveness due to its explanatory capabilities, it surprisingly decreases realism compared to the GUI. Overall, the preference between the conversational agent and the GUI varied greatly among participants, highlighting the need for further research into the evolving shift towards speech interaction in VR.

CCS Concepts

• ***Human-centered computing*** → *Virtual reality; Natural language interfaces; Graphical user interfaces; Interactive systems and tools;*

DOI: 10.2312/egve.20241378

1. Introduction

In the realm of digital interaction, written communication has long been the dominant paradigm. However, with the advent of technologies such as virtual reality (VR) and conversational agents (CAs), speech is emerging as a more prominent mode of communication. In VR, where immersiveness is often crucial, these agents can enhance presence and sense of immersion, while providing explainability and enabling users to interact simply by using their voice. While being a relatively new area, research on speech as an interaction paradigm in VR is limited. Our study aims at exploring this paradigm shift by conducting a comparative analysis between a non-embodied conversational agent and a graphical user interface in VR.

The context of the study is a VR application for procedural building generation that was designed to sensitize people, especially stakeholders, to use sustainable materials in construction, thereby promoting the circular economy. By 2030, the demand for natural resources will exceed the capacity of Earth by more than double. This underlines the urgent need for a transition to circular systems, especially in the construction industry. However, GUIs often struggle to effectively control procedural modeling due to the sheer number of parameters, leading to interface overload. Therefore, CAs have the potential to significantly enhance efficiency in this regard by breaking down complex modeling tasks into simpler conversational interactions. Additionally, the paper presents a two-API-call pipeline for a CA, separating natural language understanding (NLU) and natural language generation (NLG). This setup enables prompt inference of relevant information, rather than inferring all data each time. This method is essential for prompt optimization, especially due to the limited tokens available for LLMs' context window. Furthermore, the paper introduces a novel UDP-based framework for running computationally heavy workloads on dedicated hardware while maintaining standalone visualization on a head-mounted display (HMD).

While previous research has compared CAs with GUIs within a VR futuristic workplace scenario [BWP*22, BWN*22a, BWN*22b, BWN*22c], our study focuses on usability, presence, realism, errors detection and effectiveness in a real use-case of architectural design. These metrics were used to investigate how a conversational agent might impact the VR experience and to assess its effectiveness compared to a graphical user interface. The user study involved procedural building design tasks in VR using both the non-embodied conversational agent and the GUI, with quantitative and qualitative data collected for evaluation. We hypothesized that the conversational agent would increase presence, realism, and effectiveness but also introduce more errors. This hypothesis stems from the assumption that natural speech feels more intuitive for users, particularly those unfamiliar with VR environments, but it tends to be less reliable due to the greater challenges involved in processing speech accurately, especially in dynamic or unstructured contexts. The results suggest that while the conversational agent condition achieves higher presence and expected effectiveness, it surprisingly lowers realism. Qualitative data revealed that some users found the disembodied voice of the agent slightly alienating, as they felt disconnected by the absence of a speaking avatar, especially considering the lifelike voice. This finding is definitely intriguing and suggests avenues for future research. In general, while some users favored the conversational agent due to its explanatory capability, others preferred the GUI for its practicalities. These preferences were highly personal, influenced by users' backgrounds and speech capabilities, thus varying across individuals. This highlights the necessity for further research on the growing paradigm shift towards speech interaction.

2. Related Work

2.1. Conversational Agents

Since the inception of the first conversational agent ELIZA [Wei66] to the present day, natural language processing (NLP) has undergone significant transformations. Conversational agents have seen a remarkable improvement with the emergence of LLMs, particularly with the development of transformer-based architectures and the attention mechanism [VSP*23]. These advancements have made conversational agents increasingly realistic, paving the way for revolutionary changes in human-machine interactions.

Until today, numerous conversational agents have been developed for various purposes, ranging from research to industry applications. Examples include the use of conversational agents in online sales [Jus18], in educational settings [SOR17, ZVB21], or as laboratory assistants [CLTK19] and personal assistants [YGDSN*23]. However, most CAs are restricted to simple tasks, typically supported by standalone commands. To handle more complex tasks, agents need to be capable of generalizing and combining the commands they understand. This is the reason why Fast et al. introduced Iris [FCM*18], a textual conversational agent that allows users to combine commands through nested conversations to accomplish open-ended data science tasks.

2.2. Conversational Agents in Mixed and Virtual Reality

Virtual reality stands out as the premier medium for immersive experiences, making it an ideal environment for leveraging CAs to enhance immersion. One of the earliest examples of a CA in VR is Max [KJPLW03], an anthropomorphic agent designed for cooperative construction tasks. Max's innovation lies in its use of synthetic speech, gaze, facial expressions, and gestures. Since then, numerous applications for CAs in VR have emerged, such as virtual agents in online shopping scenarios to improve consumer experiences [SZD23].

Nowadays, most widespread applications of CAs in VR revolve around training scenarios. For instance, a VR game system has been developed for training customer service employees, incorporating a multimodal conversational agent with speech and gesture dialogues [FOF*20]. Other examples, fostering inclusivity, include a conversational AI-based VR system to enhance construction safety training [HSL*24], and the use of LLMs to aid interview preparation for underrepresented professionals in computer science [ANA*23]. Additionally, a recent paper investigates the impact of embodied conversational agents (ECAs) on users in a first responder VR training scenario [KRK23], focusing on aspects such as realism, presence, and task performance.

Another crucial sector where CAs prove very useful is in teaching. CARLA [MHSBI20] is an assistant providing interaction

and support within an e-learning platform through spoken dialogue. Furthermore, Callaghan et al. developed a voice-driven virtual assistant tutor to support students in a remote VR laboratory [CBF*19]. A proper design of virtual learning environments (VLEs) is in fact vital for utilizing VR technology in educational settings, and research is increasingly focused on improving the effectiveness of such environments [NNQ17, CSMH*24]. Moreover, mixed reality (MR) is also explored for educational purposes, as seen in a research from the University of Basilicata investigating distributed pair programming (DPP) education [MEG23], enabling developers to collaborate regardless of location through the use of a CA.

The strength of virtual reality is also particularly relevant when considering individuals with intellectual disabilities [GPCV*23] or autism [AGS23], who can greatly benefit from conversational agents in VR, especially for competence training and well-being management. In fact, the immersive nature of VR can provide a safe and controlled environment for tailored interventions and support.

2.3. Comparison of Conversational Agents and GUIs

CA-supported voice interaction represents a relatively new frontier, particularly with the advancing accuracy of automatic speech recognition (ASR) technology. This is why the literature is rich in guidelines and heuristics for GUIs but lacks similar resources for voice user interfaces (VUIs). GUIs have been in use for decades and have evolved alongside technological advancements, such as the mobile revolution which significantly impacted their design. However, finding comprehensive guidelines for VUIs is more challenging, highlighting the need for further research. Murad et al. [MMCC19] conducted an extensive literature review on this topic and identified the most relevant heuristics in two sets of guidelines: 10 from the studies of Suhm [Suh03] and 17 from the studies of Wei and Landay [WL18]. These heuristics serve as a foundation towards the development of general VUI guidelines, similar to those existing for GUIs. In our study, we focused on applying the most useful heuristics to our CA to maximize its usability.

While general guidelines for VUIs are under-researched, the exploration of effective principles in the realm of VR is almost completely uncharted. To the best of our knowledge, there is only one project in literature that examines and compares a GUI and a VUI within an interactive virtual environment (IVE): the NUX project [BWP*22, BWN*22a, BWN*22b, BWN*22c]. The research aims to investigate user experience and usability using qualitative and quantitative methodologies. Specifically, Buchta et al. investigated user attitudes towards microtransactions in a VR environment by comparing a GUI with a robotic CA in a futuristic IVE [BWN*22a]. The research group also investigated the assistant's behavior and collected user opinions on the features for both GUI and VUI [BWN*22b]. The key difference with our study lies in the embodiment of the CA, as we employed a non-embodied agent, while a robot-like model was used for the NUX project. This difference is crucial as the user interaction between embodied and non-embodied CAs differs greatly. As a matter of fact, the substantial disparity in results from our study and theirs regarding realism serves as confirmation. Furthermore, the NUX project focused on interaction and communication with the interfaces, while our investigation delved into metrics such as user presence, realism of experience, encountered errors, and effectiveness for a real use case in architectural settings. In addition, we employ the System Usability Scale (SUS) [Bro95] to quantify usability across both GUI and VUI systems.

2.4. Procedural Design in Virtual Reality

As computational power advances and devices become increasingly capable of handling larger volumes of data, the demand for more detailed and sophisticated environments continues to rise. This is especially true for VR, where the sense of immersion and attention to detail are crucial. Procedural generation of environments has become an important tool to help speed up the design process and optimize performance [GMD23]. Recognizing this, Joklova et al. [JB19] conducted research on tools that support digital creation and representation, focusing on areas such as architecture, landscape planning and civil engineering. Their aim was to embrace the dynamic nature of this field and provide valuable educational resources for students at the Faculty of Architecture at the Slovak University of Technology in Bratislava.

One of the most straightforward applications of procedural design is in city planning, due to the repetitive nature of urban structures and the ease of assigning parameters to control various building features. Numerous studies in the literature explore techniques for procedurally generating cities. Some focus on approaches that integrate real-world street data [BWAB20, WBE22] or aim to enhance it by automatically detecting and filling the empty spaces between buildings [ELNN*19], while others emphasize the creation of entirely new cities based on specific architectural styles and universally applicable building criteria [SdCH*22, LZK12]. A unique approach by Rodrigues et al. [RBCNV15] involves the use of genetic algorithms to optimize grammars that generate models with the desired properties. Furthermore, Cogo et al. [CPH*19] provide a comprehensive overview of the integration of procedural modeling techniques required to create a complete virtual city, including streets, roads, outdoor areas and fully furnished indoor spaces.

Procedural techniques also have significant applications in interior design. Du et al. [DZH*23] developed MyRoom, a Unity plugin that imports layout datasets for indoor synthesis and allows users to procedurally generate and interactively design interior scenes. This plugin simplifies the creation of high-quality indoor environments in games, helping users achieve their design goals efficiently. Similarly, Cheng et al. [COHW19] introduced VRoamer, a system that enables users to explore real-world spaces virtually, with scenes generated dynamically by extracting walkable areas and detecting physical obstacles in real time. Another notable application of procedural generation is the creation of terrains. Meng et al. [MCSL09] present an efficient algorithm based on midpoint displacement for generating terrain in games and simulations, including procedural texture generation that matches the heightmap, thereby enhancing the realism of the generated landscapes.

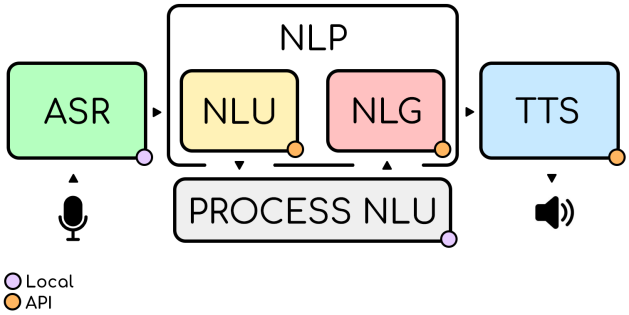

Figure 2: *Architecture of the conversational agent. Local services are running on a dedicated workstation and online services are accessed using the OpenAI API.*

3. Conversational Agent for Building Design

Our research focuses on enabling procedural building design in virtual reality through a CA interface, which offers significant advantages in explainability compared to conventional GUIs. Furthermore, the use of a CA creates a user-friendly interface that does not require extensive learning from users, thus improving accessibility for those who are not technologically proficient or have limited experience with complex GUIs. Through the use of voice, users can manipulate the virtual environment and engage in a dialogue with the agent to gain a deeper understanding of the end-of-life (EoL) considerations of different materials and their impact on the building being designed.

3.1. Conversational Agent Architecture

Figure 2 shows a schematic representation of the CA pipeline. The first module consists of automatic speech recognition (ASR), which converts the user's speech into text. This module is implemented using a local Nvidia Riva server running in a Docker container on the workstation. The ASR is followed by the natural language processing (NLP) phase, which consists of the natural language understanding (NLU) module and the natural language generation (NLG) module. Both modules use the OpenAI GPT-4 Turbo model, which is known for its significantly improved processing speed and performance compared to other LLMs.

The NLU module is responsible for recognizing the user intents, extracting functions and parameters, and translating them into programmatic commands. To achieve this, a prompt has been optimized to precisely define possible functions and parameters that the model should output. Additionally, the module can identify when the user seeks general conversation or requests information regarding the building, materials or metrics. This versatility enables the agent to handle various conversational scenarios, efficiently and effectively. Moreover, the agent is able to process multiple requests simultaneously, allowing it to handle multiple requests from a single sentence. To detail, the first part of the NLU prompt provides a general description of the task and guidelines for handling various types of user queries, including how to classify invalid or conversational requests. The second part lists all possible parameters categorized into various groups, while the third part outlines the specific commands and their corresponding outputs based on the given parameters. Additionally, building information such as the number of floors and materials is inferred and updated with each request to increase accuracy.

After the NLU module, the output command is processed accordingly to the detected intent. If the intent involves a function, the programmatic command is sent to the VR application, running on HMD, to be executed and to update the visualization. Subsequently, the VR application sends a status message to the agent indicating whether the execution of the function was successful or whether errors occurred. In addition, the VR application sends an encoded string representing the current state of the building the user is designing. This data allows the agent to accurately address questions related to the building. Alternatively, if the detected intent is conversational or informational, a subset is extracted from a database containing the agent's knowledge in text format. This subset, along with the decoded building string and status response, is then inferred to the NLG API request prompt, allowing the NLG module to generate a natural response based on the specific context and information requested.

We have included a memory feature for both the NLU and NLG modules that stores the last 10 interactions between the user and the CA. This allows the models to retain earlier parts of the conversation, which greatly improves contextual understanding. As a result, users do not have to repeat topics and formulate complete sentences each time, which promotes a more natural flow of conversation.

The last module is the text-to-speech (TTS) module, which was implemented using the OpenAI TTS HD model. The text generated by the NLG module is transmitted to the HMD, where an online TTS API request is sent directly from the VR application. This approach enables audio playback directly from the headset's speakers, eliminating the need for external headphones. We chose Shimmers' voice from the OpenAI library for the agent because we found it provided the perfect balance of warmth, clarity and friendliness, allowing for a natural and welcoming interaction with users.

3.2. Workstation-HMD Framework

Within VR, having a standalone HMD is particularly convenient as it allows users to move freely without being restricted by cables. However, the downside of a standalone headset is its limited computational power, often insufficient for running hardware-intensive computations. This limitation motivated our decision to design a framework based on User Datagram Protocol (UDP) to separate the visualization and computational components. By implementing this solution, we enable the system to be highly scalable and upgradable in the future without being constrained by the hardware limitations of the HMD. UDP is chosen over Transmission Control Protocol (TCP) or other protocols due to its fast data transmission, as it does not entail any processing for data validation. In VUIs, swift response times are crucial to create a natural conversational flow with agents. Additionally, our future plans include integration with architectural software that support UDP connectivity, such as Grasshopper and Archicad, in order to enable real-time procedural geometry generation.

Figure 3 displays a graph of the framework. All system compo-

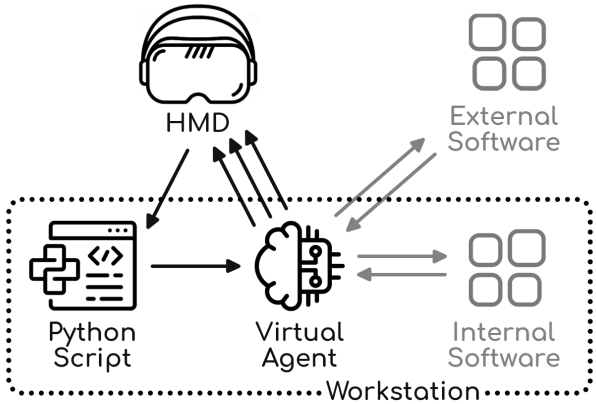

Figure 3: *Workstation-HMD framework. Internal and external software illustrate the hardware scalability of the agent.*

nents share the same private IP address range, placing both HMD and workstation in the same local area network (LAN). Each arrow represents a UDP connection through a specific port. The HMD used in this study is the Meta Quest 2, while the workstation is an ASUS ROG Strix equipped with an Intel Core i9-13980HX CPU and an Nvidia GeForce RTX 4090 GPU. Both the VR application for visualization on the HMD and the agent have been developed in Unity 2022.3.10f1. No external or internal software were used for this study, as the implementation has already been completed but will be utilized in future work.

The agent sends data to the HMD via three UDP connections on different ports. The first channel is used to send the NLU API response, enabling the HMD to process it graphically. The second channel is for sending the NLG API response, so that the TTS API request can be sent directly from the HMD. Finally, the third channel is reserved for data transmission, such as the computation of metrics that occur in the workstation but need to be visualized in the VR application.

All data transmitted from the HMD to the workstation passes through a Python script, primarily designed to filter the data and control the workstation's microphone. The user is able to control the communication with the agent by pressing the trigger button on the left Quest 2 controller. This action is mapped to operate the microphone on the workstation on which the Nvidia Riva server for the ASR module is running locally. Subsequently, all messages that are not related to microphone control, such as the building encoded string and the status message, are forwarded to the agent running on the workstation.

3.3. Procedural Building Design

The current setup comprehend seven different floor presets, including one ground floor, one roof floor and five middle floors. Each preset varies in composition, with some having a larger portion dedicated to apartments while others allocate more space to offices, and in total gross floor area (GFA), a parameter used to compute metrics. The selection of materials provides two choices for each category of walls, including exterior walls, non-load bearing interior walls, and load bearing interior walls, and three alternatives for flooring. This brings the total number of distinct material options to nine. For environmental metrics, the system can compute and display global warming potential (GWP), acidification potential (AP), primary energy intensity (PEI) and oekoindex (OI3). When visualized, the metrics are displayed for each floor with a color gradient from red to green, with red indicating a poor value and green indicating a good value. The functions that both CA and GUI can execute are reported in Table 1, which lists the respective functions along with their required and optional parameters.

Functions	Parameters
add_floors	number, floor preset, (position)
remove_floor	position
change_floor	floor preset, position
swap_floors	position1, position2
change_walls	wall type, wall material, (position)
change_slabs	slab material, (position)
visualize_metric	metric / off
change_time	time / daily event

Table 1: *Available functions and relative parameters that both CA and GUI can execute for the procedural generation of the building. Optional parameters are in parentheses.*

The VR application operates in two distinct modes: design mode and immersive mode. In design mode, users can manipulate the position, orientation, and scale of the building, as well as change floors, materials, and view metrics to determine the environmental impact of their designs. In immersive mode, the building is scaled to its original size and the user is teleported inside, allowing them to navigate the space as though they were physically there. Teleportation was used as a locomotion metaphor during building exploration. In this mode, the user can explore the designed building and easily change the interior materials, benefiting from immediate visual feedback on the changes. Inside the building, functions are resticted as it is not possible to make structural changes such as adding or removing floors.

Currently, the procedural generation of the building is entirely done with Unity, which allows to easily manipulate geometry within a virtual environment. In future work, we plan to integrate Grasshopper into our framework to improve the procedural generation capabilities and allow for more flexibility and more sophisticated geometry generation.

4. User Study

We conducted a study to investigate the effects of integrating a CA into a VR application as user interface for procedural building design. Participants engaged in building design tasks using both the CA and a traditional GUI interface conditions in VR.

Conversational Agent Condition
In the CA condition, participants completed the building design tasks by giving voice commands to the CA, which executed the requested functions listed in Table 1. Participants were also able to

consult the GUI, displayed as a floating panel anchored to their left hand, to view information such as available floors, materials and metrics. The GUI could be interacted with using rays and trigger buttons. Although the GUI served as a visual aid, all actual design interactions and modifications were carried out exclusively through the CA, with participants relying solely on voice commands for function execution.

Graphical User Interface Condition
In the GUI condition, the CA was disabled and participants relied solely on the GUI to perform the functions listed in Table 1. While the GUI sections displaying available floors, materials and metrics remained accessible for consultation, an additional GUI section was enabled, allowing participants to directly select parameters and execute functions through the GUI. In this condition, all tasks were completed exclusively through the GUI, without any reliance on the CA.

The study followed within-groups design. We collected both quantitative and qualitative data for evaluation, focusing on user-reported usability, presence, realism, perception of errors, and effectiveness of both interfaces. To measure these metrics, we used a post-experiment questionnaire and conducted short interviews with participants. Presence was assessed using questions from previous research [SUS94, WS98], while usability was assessed using the System Usability Scale [Bro95]. We introduced new items to assess realism, perception of errors and effectiveness. In addition, four open-ended questions about the agent were included to explore potential future improvements, along with a final open-ended question to summarize the user experience. Our hypotheses were that the integration of the CA would improve presence, realism and effectiveness in comparison to the GUI by creating a more engaging and authentic experience. However, we also anticipated that this integration could result in more errors due to the increased complexity of interactions.

4.1. Task

During the user study, participants were asked to design a building while exploring how different materials impact the circular economy. At the beginning, participants spent about 5 minutes familiarizing with the examined interface, interacting with it to learn its functions. During this time, they explored actions such as adding or removing floors and changing wall and slab materials. Participants were also able to review the available floor presets and materials, as well as the metrics they could visualize and discuss with the CA. For this purpose, the GUI was consultable throughout the entire user study for both conditions. However, during the CA condition, performing the functions listed in Table 1 via the GUI was restricted, allowing only the review of information to facilitate the interaction with the CA.

Subsequently, participants were then given the first actual assignment, which involved the design phase of the task. This required using the available floor presets to instantiate a building composed of a ground floor, five middle floors, and a roof floor. During this phase, participants were able to view the building from the outside and perform manipulations such as rotation, movement and scaling to better visualize it during the design process.

After the first phase, the material phase began. In this phase, the participants were asked to change some materials of the designed building. The first materials to be changed were the exterior walls of some floors from timber, which is the default parameter, to concrete. After that, participants were asked to activate the immersive mode and explore the building from the inside, while changing the materials of the load-bearing and non-load-bearing interior walls, as well as the flooring.

Once the material phase was completed, the final phase, focused on exploring the building metrics, concluded the task. During this phase, participants were asked to leave the immersive mode and visualize some environmental metrics. Through this process, participants were able to observe how different materials influenced the metrics, highlighting their respective impacts on circularity. This not only helped them understand the environmental implications of their material choices, but also empowered them to make informed decisions for sustainable building design by replacing less sustainable materials with better alternatives.

4.2. Procedure

Participants began by reading and signing a consent form outlining the study's purpose, description, and potential discomforts such as motion sickness. Following this, each participant completed a brief demographic questionnaire and received instructions on how to use the HMD. Once familiar with the VR application, they proceeded to perform the task described in Section 4.1 with the first interface. To minimize the learning effect, half of the participants started with the GUI while the other half started with the CA. After completing the task with the first interface, participants filled out a questionnaire assessing usability, presence, realism, encountered errors, and effectiveness of the interface in real-world scenarios. Following that, they repeated the task using the second interface and filled out the questionnaire again. After assessing both interfaces, a short interview was conducted to orally gather feedback from participants. Finally, they were asked to formulate a written answer to an open-ended question regarding their overall experience. Participants were given 10 minutes on each task, with an additional 5 minutes at the beginning of the study for learning how to use the VR application, resulting in a total time of around 25 minutes spent in VR by the end of the study.

4.3. Participants

20 participants were involved in the user study, consisting of 7 females and 13 males. The average age of participants was 27.5 years (SD = 4.8 years). To gather feedback from both individuals in the architecture field and those outside of it, we selected 12 participants with backgrounds in or currently studying architecture-related fields, and 8 participants from other disciplines. Most participants were from the academic environment, including students and researchers. Their average knowledge level about VR was 3.25 (SD = 1.8) on a scale from 1 to 7, where 1 indicated no experience and 7 indicated expertise.

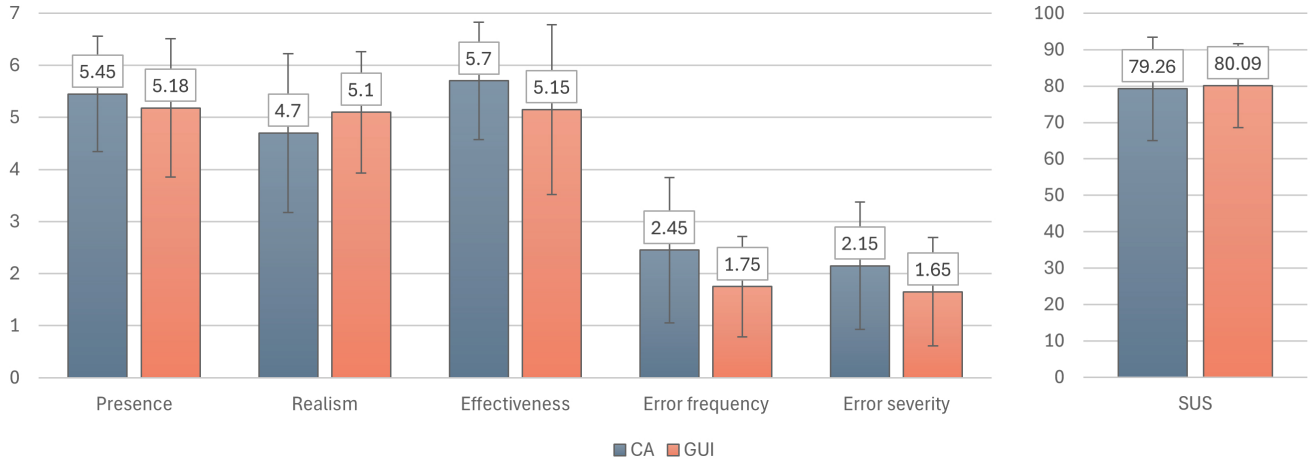

Figure 4: *Results of the questionnaire metrics investigated in the user study.*

Metrics	Presence	Realism	Effectiveness	Error frequency	Error severity	SUS
Shapiro-Wilk Test	.353 .291	.007 .162	.018 .021	.005 .001	.001 .001	.075 .570
t(19)	1.615					.272
Z-value		-1.999	-2.072	-2.038	-1.715	
p-value	.123	**.046**	**.038**	**.042**	.086	.788

Table 2: *Significance of differences between the compared interfaces. Shapiro-Wilk Test indicates if the data was normally distributed for CA and GUI conditions and subsequently significance of difference (p-value) was calculated by either paired sample t-test or by the Wilcoxon signed-rank test. Additionally, t-values are reported for normal distributions and Z-values for non-normal distributions.*

4.4. Results

The results from the questionnaires comparing the CA and GUI can be observed in Figure 4. The y-axis represents values on the Likert scale, with each bar representing the average response of participants for a specific condition, while error bars indicate the standard deviations in responses.

To analyze the statistical significance of differences between the two interfaces, we first used the Shapiro-Wilk test to assess the normality of the distributions. A value below 0.05 suggests significant departure from normal distribution. We then performed the paired sample t-test for normal distributions and the Wilcoxon test for non-normal distributions. Table 2 illustrates the results of normality tests conducted using the Shapiro-Wilk test, where the first value in each pair corresponds to the CA distributions and the second value to the GUI distributions. We also report the t-values for normal distributions and the Z-values for non-normal distributions. The table finally displays the output *p*-values from the corresponding statistical tests, with a significance threshold of .05 used for interpretation. Analysis of these results demonstrates that the differences in realism, effectiveness, and error frequency metrics between the two interfaces are statistically significant.

As predicted, the CA enhanced presence and effectiveness compared to the GUI by providing explanatory capabilities, allowing for clearer communication of information. While the difference between conditions in subjectively-rated effectiveness was significant, the difference in presence was not significant. Thus, only our hypothesis about the effectiveness was supported by the results. However, the CA interface also introduced significantly more errors than GUI, since natural conversations are susceptible to misunderstandings. Interestingly, participants perceived that the GUI significantly enhanced realism in comparison to the CA, contradicting our initial hypothesis. From the interviews and responses to the final open-ended question, two main reasons emerged. Firstly, some participants found the lack of embodiment in the agent slightly disconcerting. Being in a virtual environment with a disembodied voice felt to some extent alienating, reducing the realism of their experience. Secondly, users sometimes felt that the response time of the CA was too long, with occasional waits exceeding 10 seconds. This was primarily due to the internet speed, as the agent relies on the OpenAI API for its functionality. In terms of usability, the two interfaces did not differ significantly, both scoring a very high grade of A- in the System Usability Scale, indicating nearly excellent usability.

5. Discussion

The most intriguing finding of this study revolves around the decreased realism observed when using the CA as interaction interface in comparison to GUI. In VR, where users expect complete immersion due to its highly immersive features, it is often crucial to mimic real-world behavior. CAs are designed to emulate human speech and actions, essentially serving as metaphors for real human assistants. Users, hearing a human-like voice and interacting

within a realistic virtual environment, naturally expected a human avatar to converse with. Furthermore, achieving realism with CAs poses one of the most significant challenges, as their purpose is to simulate human behavior, requiring them to adapt to conversations, situations and unexpected events.

In our investigated scenario, which involved procedural building design to encourage stakeholders to use more sustainable materials in line with circular economy principles, the CA emerged as significantly more effective interface than GUI according to users' judgement. This was primarily due to its explanatory capabilities, allowing the agent to engage in direct conversations and explanations with users, addressing their questions effectively. This suggests that in applications where users need to comprehend complex instructions or learn new information, CAs serve as a valuable interface to achieve these objectives. While GUIs can efficiently display information and data, they often require users to interpret them independently. Additionally, with complex or large datasets, users may struggle to locate and access the specific information or functions they need. Conversely, CAs can provide access to this information through targeted questions, simplifying the process. However, in simpler systems, CAs may be slower than GUIs and more prone to errors, making them less suitable. Therefore, a combination of CAs and GUIs is often the optimal solution, allowing users to perform quick actions or access written information and data as needed, while still utilizing CAs for more complex tasks and inquiries.

A drawback of using CAs is related to the reliability of the interface, particularly in terms of errors. GUIs offer limited functions in a structured manner, clearly indicating the possible interactions users can have with them. However, CAs lack this feature, making it more challenging for users to understand the available interactions. To address this, written guidelines for using CAs are often helpful, providing users with a clear understanding of the potential interactions they can have. Equally crucial is the agent's ability to communicate its capabilities, indicating what actions it can or cannot perform. This allows the agent to provide feedback to users when certain functions are not possible, helping users understand the reason and preventing them from making the same requests again.

5.1. Limitations and Future Work

The CA has demonstrated considerable effectiveness for its intended task, gathering positive feedback from most participants of the study. However, we have identified some limitations and areas for improvement. One significant limitation arises from the agent's architecture, which relies on multiple API calls, thereby making the response speed of the CA highly dependent on internet speed and server load. This dependency often leads to variable response times, with the agent sometimes answering quickly and at other times taking several seconds to generate a response. Furthermore, another issue emerged during the user study concerning the ASR module. Trained primarily on American accents, it struggled to understand foreign speech patterns, leading to communication challenges, especially for participants with strong non-American accents. This occasionally caused misunderstandings, significantly impacting the experience of some of the participants.

In our upcoming work, we aim to improve the agent to provide a more robust and dependable interface. Initially, we plan to give the agent a body to explore whether this enhances the overall realism of the interaction. Additionally, we also intend to improve response times by employing a dedicated network for API requests and enhance speech recognition by adopting a context-based approach, which could prioritize certain words and potentially improve understanding for non-native speakers. Most significantly, we aim to expand procedural design capabilities by integrating external software such as Grasshopper and Archicad. This integration will allow users to generate geometry procedurally, without being confined to preset options. Furthermore, we will increase the agent's functionalities and refine its knowledge to offer more accurate and precise explanations of concepts and answers to questions. Lastly, additional research is required to explore the observed reduction in perceived realism associated with the non-embodied CA interface.

6. Conclusion

This paper investigated the use of a CA to perform procedural building design tasks in VR. A pipeline for the CA was introduced, which involved utilizing LLMs in two separate API calls, thereby separating natural language understanding and natural language generation. Additionally, a framework was presented to connect a HMD and a workstation to overcome the hardware limitations of standalone headsets, allowing the agent to utilize external software for hardware-intensive computations.

A user study was conducted to evaluate the non-embodied CA interface, comparing it with a traditional GUI in a VR architectural design task focused on circular economy principles. The study evaluated user-reported usability, presence, realism, errors, and effectiveness of both interfaces. The results indicate that while the CA was found to be more effective, it also introduced more errors. Particularly interesting was the decreased sensation of realism reported by participants using the non-embodied CA compared to the GUI. The exploration of VR procedural generation using voice interfaces represents an intriguing topic and this paradigm shift from written to spoken interaction may lead to increased efficiency of voice user interfaces in the future.

Acknowledgments

This research was conducted as part of the Circular Twin project, funded by the Austrian Ministry for Climate Action, Environment, Energy, Mobility, Innovation, and Technology, with support from the Austrian funding agency (FFG) under grant no. 899167.

We would like to express our gratitude to Robin Jakoubek for creating the building model used in the study, Valentinas Petrinas for generating the floor models, and Nooshin Shariattalab for conducting the life cycle assessment (LCA) of the elements and preparing the figures illustrating the different layers of the elements. Special thanks go to Stefan Schützenhofer for his administration and crucial assistance throughout the project. Lastly, we would like to thank our participants for taking part in our user study.

References

[AGS23] ALVARADO Y., GUERRERO R., SERÓN F.: Inclusive learning through immersive virtual reality and semantic embodied conversational agent: A case study in children with autism. *Journal of Computer Science and Technology 23*, 2 (Oct. 2023), e09. URL: https://journal.info.unlp.edu.ar/JCST/article/view/2727, doi:10.24215/16666038.23.e09. 3

[ANA*23] AJRI S. J., NGUYEN D., AGARWAL S., PADALA A. K. R., YILDIRIM C.: Virtual aivantage: Leveraging large language models for enhanced vr interview preparation among underrepresented professionals in computing. In *Proceedings of the 22nd International Conference on Mobile and Ubiquitous Multimedia* (New York, NY, USA, 2023), MUM '23, Association for Computing Machinery, p. 535–537. URL: https://doi.org/10.1145/3626705.3631799, doi:10.1145/3626705.3631799. 2

[Bro95] BROOKE J.: Sus: A quick and dirty usability scale. *Usability Eval. Ind. 189* (11 1995). 3, 6

[BWAB20] BURCH M., WALLNER G., ARENDS S. T., BERI P.: Procedural City Modeling for AR Applications. In *2020 24th International Conference Information Visualisation (IV)* (Sept. 2020), pp. 581–586. ISSN: 2375-0138. URL: https://ieeexplore.ieee.org/document/9373104, doi:10.1109/IV51561.2020.00098. 3

[BWN*22a] BUCHTA K., WÓJCIK P., NAKONIECZNY K., JANICKA J., GAŁUSZKA D., STERNA R., IGRAS-CYBULSKA M.: Microtransactions in vr. a qualitative comparison between voice user interface and graphical user interface. In *2022 15th International Conference on Human System Interaction (HSI)* (2022), pp. 1–5. doi:10.1109/HSI55341.2022.9869475. 2, 3

[BWN*22b] BUCHTA K., WÓJCIK P., NAKONIECZNY K., JANICKA J., GAŁUSZKA D., STERNA R., IGRAS-CYBULSKA M.: Modeling and optimizing the voice assistant behavior in virtual reality. In *2022 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)* (2022), pp. 397–402. doi:10.1109/ISMAR-Adjunct57072.2022.00086. 2, 3

[BWN*22c] BUCHTA K., WÓJCIK P., NAKONIECZNY K., JANICKA J., IGRAS-CYBULSKA M.: Nux characters - interaction with voice assistants in virtual reality. In *2022 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)* (2022), pp. 917–918. doi:10.1109/ISMAR-Adjunct57072.2022.00204. 2, 3

[BWP*22] BUCHTA K., WÓJCIK P., PELC M., GÓROWSKA A., MOTA D., BOICHENKO K., NAKONIECZNY K., WRONA K., SZYMCZYK M., CZUCHNOWSKI T., JANICKA J., GAŁUSZKA D., STERNA R., IGRAS-CYBULSKA M.: Nux ive - a research tool for comparing voice user interface and graphical user interface in vr. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)* (2022), pp. 982–983. doi:10.1109/VRW55335.2022.00342. 2, 3

[CBF*19] CALLAGHAN M. J., BENGLOAN G., FERRER J., CHEREL L., EL MOSTADI M. A., GOMEZ EGUÍLUZ A., MCSHANE N.: Voice driven virtual assistant tutor in virtual reality for electronic engineering remote laboratories. In *Smart Industry & Smart Education* (Cham, 2019), Auer M. E., Langmann R., (Eds.), Springer International Publishing, pp. 570–580. 3

[CLTK19] CAMBRE J., LIU Y., TAYLOR R. E., KULKARNI C.: Vitro: Designing a voice assistant for the scientific lab workplace. In *Proceedings of the 2019 on Designing Interactive Systems Conference* (New York, NY, USA, 2019), DIS '19, Association for Computing Machinery, p. 1531–1542. URL: https://doi.org/10.1145/3322276.3322298, doi:10.1145/3322276.3322298. 2

[COHW19] CHENG L.-P., OFEK E., HOLZ C., WILSON A. D.: VRoamer: Generating On-The-Fly VR Experiences While Walking inside Large, Unknown Real-World Building Environments. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (Mar. 2019), pp. 359–366. ISSN: 2642-5254. URL: https://ieeexplore.ieee.org/document/8798074, doi:10.1109/VR.2019.8798074. 3

[CPH*19] COGO E., PRAZINA I., HODZIC K., HASELJIC H., RIZVIC S.: Survey of integrability of procedural modeling techniques for generating a complete city. In *2019 XXVII International Conference on Information, Communication and Automation Technologies (ICAT)* (Oct. 2019), pp. 1–6. ISSN: 2643-1858. URL: https://ieeexplore.ieee.org/document/8939012, doi:10.1109/ICAT47117.2019.8939012. 3

[CSMH*24] CHHEANG V., SHARMIN S., MÁRQUEZ-HERNÁNDEZ R., PATEL M., RAJASEKARAN D., CAULFIELD G., KIAFAR B., LI J., KULLU P., BARMAKI R. L.: Towards anatomy education with generative ai-based virtual assistants in immersive virtual reality environments. In *2024 IEEE International Conference on Artificial Intelligence and eXtended and Virtual Reality (AIxVR)* (2024), pp. 21–30. doi:10.1109/AIxVR59861.2024.00011. 3

[DZH*23] DU H., ZHAO Y., HUANG S., BAI J., TIAN S., LIU J.: MyRoom: A Unity Plugin for Procedural and Interactive Indoor Scene Synthesis. In *2023 IEEE Conference on Games (CoG)* (Aug. 2023), pp. 1–2. ISSN: 2325-4289. URL: https://ieeexplore.ieee.org/document/10333189, doi:10.1109/CoG57401.2023.10333189. 3

[ELNN*19] EFREN C., LUO X., NAVARRO NEWBALL A. A., NAVARRO CADAVID A., LOZANO-GARZÓN C.: Procedural Placement of Existing Building Models in Virtual Cities. In *2019 International Conference on Virtual Reality and Visualization (ICVRV)* (Nov. 2019), pp. 238–242. ISSN: 2375-141X. URL: https://ieeexplore.ieee.org/document/9212987, doi:10.1109/ICVRV47840.2019.00056. 3

[FCM*18] FAST E., CHEN B., MENDELSOHN J., BASSEN J., BERNSTEIN M. S.: Iris: A conversational agent for complex tasks. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2018), CHI '18, Association for Computing Machinery, p. 1–12. URL: https://doi.org/10.1145/3173574.3174047, doi:10.1145/3173574.3174047. 2

[FOF*20] FURUNO T., OMI Y., FUJITA S., DONGHAO W., HOSHINO J.: Customer service training vr game system using a multimodal conversational agent. In *Entertainment Computing – ICEC 2020* (Cham, 2020), Nunes N. J., Ma L., Wang M., Correia N., Pan Z., (Eds.), Springer International Publishing, pp. 277–281. 2

[GMD23] GOVORI E., MURTURI I., DUSTDAR S.: A Comprehensive Performance Evaluation of Procedural Geometry Workloads on Resource-Constrained Devices. In *2023 IEEE International Conference on Edge Computing and Communications (EDGE)* (July 2023), pp. 271–279. ISSN: 2767-9918. URL: https://ieeexplore.ieee.org/document/10234246, doi:10.1109/EDGE60047.2023.00049. 3

[GPCV*23] GARCIA-PI B., CHAUDHURY R., VERSAW M., BACK J., KWON D., KICKLIGHTER C., TAELE P., SEO J. H.: Allychat: Developing a vr conversational ai agent using few-shot learning to support individuals with intellectual disabilities. In *Human-Computer Interaction – INTERACT 2023* (Cham, 2023), Abdelnour Nocera J., Kristín Lárusdóttir M., Petrie H., Piccinno A., Winckler M., (Eds.), Springer Nature Switzerland, pp. 402–407. 3

[HSL*24] HUSSAIN R., SABIR A., LEE D.-Y., ZAIDI S. F. A., PEDRO A., ABBAS M. S., PARK C.: Conversational ai-based vr system to improve construction safety training of migrant workers. *Automation in Construction 160* (2024), 105315. URL: https://www.sciencedirect.com/science/article/pii/S0926580524000517, doi:https://doi.org/10.1016/j.autcon.2024.105315. 2

[JB19] JOKLOVA V., BUDREYKO E.: Digital Technologies in Architectural Design, Verification and Representation. In *2019 International Conference on Engineering Technologies and Computer Science (EnT)* (Mar. 2019), pp. 102–106. URL: https://ieeexplore.ieee.org/document/8711907, doi:10.1109/EnT.2019.00028. 3

[Jus18] JUSOH S.: Intelligent conversational agent for online sales. In *2018 10th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)* (2018), pp. 1–4. doi:10.1109/ECAI.2018.8679045. 2

[KJPLW03] KOPP S., JUNG B., PFEIFFER-LESSMANN N., WACHSMUTH I.: Max - a multimodal assistant in virtual reality construction. *KI 17* (01 2003), 11–. 2

[KRK23] KÁN P., RUMPELNIK M., KAUFMANN H.: Embodied Conversational Agents with Situation Awareness for Training in Virtual Reality. In *ICAT-EGVE 2023 - International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments* (2023), Normand J.-M., Sugimoto M., Sundstedt V., (Eds.), The Eurographics Association. doi:10.2312/egve.20231310. 2

[LZK12] LI J., ZHANG Y., KONG D.: Rule-based procedural modeling of buildings. In *2012 IEEE International Conference on Computer Science and Automation Engineering (CSAE)* (May 2012), vol. 2, pp. 450–454. URL: https://ieeexplore.ieee.org/document/6272812, doi:10.1109/CSAE.2012.6272812. 3

[MCSL09] MENG D., CAI X., SU Z., LI J.: Photorealistic terrain generation method based on fractal geometry theory and procedural texture. In *2009 2nd IEEE International Conference on Computer Science and Information Technology* (Aug. 2009), pp. 341–344. URL: https://ieeexplore.ieee.org/document/5234644, doi:10.1109/ICCSIT.2009.5234644. 3

[MEG23] MANFREDI G., ERRA U., GILIO G.: A mixed reality approach for innovative pair programming education with a conversational ai virtual avatar. In *Proceedings of the 27th International Conference on Evaluation and Assessment in Software Engineering* (New York, NY, USA, 2023), EASE '23, Association for Computing Machinery, p. 450–454. URL: https://doi.org/10.1145/3593434.3593952, doi:10.1145/3593434.3593952. 3

[MHSBI20] MACIAS-HUERTA P., SANTAMARIA-BONFIL G., IBÁÑEZ M.: Carla: Conversational agent in virtual reality with analytics. 2

[MMCC19] MURAD C., MUNTEANU C., COWAN B. R., CLARK L.: Revolution or evolution? speech interaction and hci design guidelines. *IEEE Pervasive Computing 18*, 2 (2019), 33–45. doi:10.1109/MPRV.2019.2906991. 3

[NNQ17] NAKHAL B., NAKHAL B., QUERREC R.: Cognitive embodied conversational agents in virtual learning environment. In *2017 29th International Conference on Microelectronics (ICM)* (2017), pp. 1–4. doi:10.1109/ICM.2017.8268858. 3

[RBCNV15] RODRIGUES F. C. M., BENTO CAVALCANTE NETO J., VIDAL C. A.: Split Grammar Evolution for Procedural Modeling of Virtual Buildings. In *2015 XVII Symposium on Virtual and Augmented Reality* (May 2015), pp. 75–83. URL: https://ieeexplore.ieee.org/document/7300730, doi:10.1109/SVR.2015.18. 3

[SdCH*22] SALPISTI D., DE CLERK M., HINZ S., HENKIES F., KLINKER G.: A Procedural Building Generator Based on Real-World Data Enabling Designers to Create Context for XR Automotive Design Experiences. In *Virtual Reality and Mixed Reality* (Cham, 2022), Zachmann G., Alcañiz Raya M., Bourdot P., Marchal M., Stefanucci J., Yang X., (Eds.), Springer International Publishing, pp. 149–170. doi:10.1007/978-3-031-16234-3_9. 3

[SOR17] SONG D., OH E. Y., RICE M.: Interacting with a conversational agent system for educational purposes in online courses. In *2017 10th International Conference on Human System Interactions (HSI)* (2017), pp. 78–82. doi:10.1109/HSI.2017.8005002. 2

[Suh03] SUHM B.: Towards best practices for speech user interface design. doi:10.21437/Eurospeech.2003-621. 3

[SUS94] SLATER M., USOH M., STEED A.: Depth of presence in virtual environments. *Presence 3* (01 1994), 130–144. doi:10.1162/pres.1994.3.2.130. 6

[SZD23] SHANGSHANG ZHU WEI HU W. L., DONG Y.: Virtual agents in immersive virtual reality environments: Impact of humanoid avatars and output modalities on shopping experience. *International Journal of Human–Computer Interaction 0*, 0 (2023), 1–23. doi:10.1080/10447318.2023.2241293. 2

[VSP*23] VASWANI A., SHAZEER N., PARMAR N., USZKOREIT J., JONES L., GOMEZ A. N., KAISER L., POLOSUKHIN I.: Attention is all you need, 2023. arXiv:1706.03762. 2

[WBE22] WAGENER N., BECKMANN J., ECKSTEIN L.: Efficient Creation of 3D-Virtual Environments for Driving Simulators. In *2022 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)* (Nov. 2022), pp. 1–6. URL: https://ieeexplore.ieee.org/document/9988421, doi:10.1109/ICECCME55909.2022.9988421. 3

[Wei66] WEIZENBAUM J.: Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM 9*, 1 (1966), 36–45. 2

[WL18] WEI Z., LANDAY J. A.: Evaluating speech-based smart devices using new usability heuristics. *IEEE Pervasive Computing 17*, 2 (2018), 84–96. doi:10.1109/MPRV.2018.022511249. 3

[WS98] WITMER B. G., SINGER M. J.: Measuring Presence in Virtual Environments: A Presence Questionnaire. *Presence: Teleoperators and Virtual Environments 7*, 3 (06 1998), 225–240. URL: https://doi.org/10.1162/105474698565686, arXiv:https://direct.mit.edu/pvar/article-pdf/7/3/225/1836425/105474698565686.pdf, doi:10.1162/105474698565686. 6

[YGDSN*23] YADAV S. P., GUPTA A., DOS SANTOS NASCIMENTO C., HUGO C. DE ALBUQUERQUE V., NARUKA M. S., SINGH CHAUHAN S.: Voice-based virtual-controlled intelligent personal assistants. In *2023 International Conference on Computational Intelligence, Communication Technology and Networking (CICTN)* (2023), pp. 563–568. doi:10.1109/CICTN57981.2023.10141447. 2

[ZVB21] ZHU J., VAN BRUMMELEN J.: Teaching students about conversational ai using convo, a conversational programming agent. In *2021 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)* (2021), pp. 1–5. doi:10.1109/VL/HCC51201.2021.9576290. 2