Evaluating UI Placement Techniques in Virtual Reality: A Comparison of Manual Multimodal, Automatic, and Adaptive Machine Learning Based Approaches

MATTHEW ALLPASS, Colorado State University

Implementing adapted 2D menus into 3D virtual reality (VR) space is a common mechanism for information output in VR environments. These adapted 2D menus introduce distinct interaction design challenges as they appear as floating UI panels within the virtual environment which can block the user from engaging with the environment and may need to be repositioned frequently to remain useful. This project evaluates three control methods for managing an adapted 2D menu in the form of a floating UI panel within a custom VR puzzle room. The three control methods include: manual multimodal input (gesture and voice), automated scripting behavior, and adaptive control using a reinforcement learning (RL) model. A within-subject study with ten participants was conducted and task completion time, interaction frequency, and user preference was measured. Manual control, using gesture and voice, significantly outperformed other methods in speed and user preference, highlighting the value of multimodal interaction for user agency. Scripted control was reliable but less preferred due to reduced user agency, while the RL model failed in 50 percent of trials due to training instabilities. The findings highlight the effectiveness of multimodal input for VR UI control and reveal practical challenges in applying RL to dynamic spatial positioning.

Key Words and Phrases: Multimodal interaction, gesture control, reinforcement learning, machine learning, virtual reality, voice control, gaze tracking

ACM Reference Format:
Matthew Allpass. 2025. Evaluating UI Placement Techniques in Virtual Reality: A Comparison of Manual Multimodal, Automatic, and Adaptive Machine Learning Based Approaches. In *Multimodal Interaction for 3DUI Capstone, May 2025, Fort Collins, CO, USA.* ACM, Fort Collins, CO, USA, 9 pages. https://doi.org/XX.XXXX/XXXXXXX.XXXXXXX

1 Introduction

Floating UI panels provide a mechanism for information input and output in VR however they require effective control methods to ensure usability in 3D spaces. Compared to 2D interfaces in traditional computer systems, the complexity of how these panels may need to be repositioned to avoid obstruction while maintaining their functionality in information delivery increases in a 3D space. Prior work has shown the efficacy of multimodal input combining gesture and voice for intuitive VR interaction [1, 4], while reinforcement learning using a proximal policy optimization (PPO) algorithm shows functionality with RL model based control of UI panels within a mixed reality space [3]. This project extends the prior work by (1) applying a reinforcement learning model using PPO to a fully VR space and (2) conducting a user study directly comparing three distinct methods of control for a floating UI panel. The three methods of control include: manual multimodal control using gesture and voice commands, adaptive control using an RL model trained to optimize panel placement, and scripted

Author's Contact Information: Matthew Allpass, Colorado State University.

CS 465, Fort Collins, CO, USA

ACM ISBN XXXX-X/25/05
https://doi.org/XX.XXXX/XXXXXXX.XXXXXXX

automation - which is similar to adaptive control's lack of user agency but separates itself due to its much higher predictability.

The study investigates three questions:

- Which method was preferred by the players?
- Which method led to the fastest task performance?
- What limitations emerge when applying reinforcement learning to unconstrained interface positioning in VR?

To answer these questions a within-subject study was conducted with ten participants. Each participant used all three control methods to complete a structured task sequence in a VR puzzle room requiring frequent UI repositioning. It was hypothesized that manual control would be the most preferred method of control due to its familiarity and user agency.

Manual control outperformed the other methods in all measurements demonstrating the utility of multimodal input, while scripted automation lagged just behind manual controls in preference and performance due to a lack of user agency. The RL model was unstable and resulted in a failed trial for half of the participants. The project contributes evidence for prioritizing manual multimodal controls in VR and acts as a failed case study that highlights the practical limitations and hurdles in training RL models in unconstrained spacial tasks.

2 Related Works

2.1 Study of Gesture and Voice Driven Input Modalities in Virtual Reality

Multimodal input combining gesture and voice has been studied in VR domains such as molecular data analysis and collaborative task completion. One study evaluated gesture and voice controls in an asymmetric mentor-mentee task and found users preferred access to all input types at once, despite similar task performance when limited to unimodal interaction methods [1]. Another study implemented voice and gesture controls for exploring molecules in VR and reported that the combination of voice and gesture control options enabled a more intuitive exploration of the molecule [4]. These findings informed the manual control design in this project, combining gestures and voice control due to their proven effectiveness, while conducting the user study in a more domain neutral environment. The puzzle room required simple object interaction and navigation, minimizing confounding factors like molecular knowledge or the need to collaborate with another in the VR space.

2.2 Study of Automated and Machine Learning Models in Virtual Reality

A specialized area of HCI and XR research explores the use of machine learning for adaptive interface behavior. Both automated controls and adaptive UI systems aim to reduce cognitive load and improve comfort by responding to user behavior in real time [5]. In mixed reality environments, reinforcement learning has been used to dynamically position 3D UI elements based on the user's pose and environment, leveraging real world anchors with the aim of staying in the user's field of view [3]. Other work has shown that machine learning agents trained in complex 3D virtual environments can benefit from giving the agent access to multiple spatial perspectives (egocentric and allocentric views) during training [2]. These works show the potential for adaptive systems to improve interactions - but highlight the complexity of learning in environments with many input vectors in a 3D space. This project builds on the prior reinforcement learning research by removing the ability to use the real world anchors present in mixed reality through training the model in a fully virtual environment, though the model will be given both egocentric and allocentric information throughout training. Additionally, unlike previous work which evaluates the models in isolation, this project directly compares a reinforcement learning based system against both manual and automated control baselines.

2.3 Project in Context

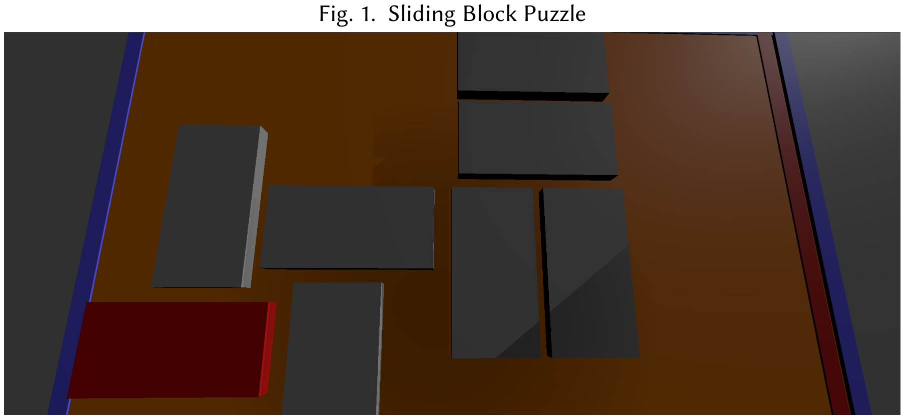
While prior research has investigated gesture and voice controls and other research has looked at adaptive controls in a vacuum, this study will directly compare these methods within the same environment/task context. Some existing multimodal research focuses on specialized domains such as molecular data [4] where meaningful interaction requires domain knowledge. Existing research into adaptive control demonstrates the complexities of reinforcement learning development in a VR space but this research focuses on the development itself rather than user feedback. This project addresses these gaps by comparing control methods side by side in a single player user study.

3 Methodology

3.1 Puzzle Room Design

The VR puzzle room, built in Unity for the Meta Quest 3, places the player in a closed cubic puzzle room containing a central column and multiple interactive zones. One wall contains a block sliding puzzle which occupies the player's entire field of view and requires it to be unobstructed to solve. Solving the block sliding puzzle opens a compartment in the central column revealing a keycard. The keycard must be inserted into the floating UI panel to display a code, which is then entered into a keypad on the wall opposite the sliding block puzzle. When the player enters digits they appear on the UI panel. Completing the tasks requires the UI panel to be repositioned several times while exploring the room, obtaining visibility of the puzzle wall, interaction with the keycard, and reading the code while entering it in the keypad.

Fig. 1. Sliding Block Puzzle

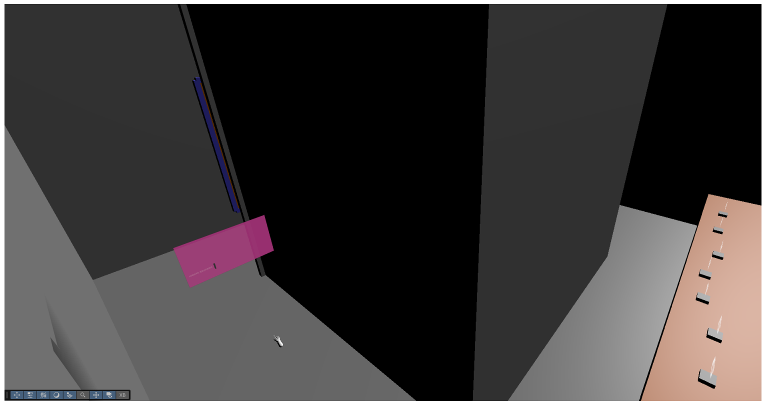


Manual controls include direct grabbing via the motion controllers and a voice command ("move") teleports the panel to a fixed position in the player's field of view. Scripted automation uses gaze tracking to reposition the panel relative to head orientation. The reinforcement learning model attempts to adapt panel position based on the player's gaze, hands, and inferred gameplay context.

3.2 Participants

A total of ten participants were recruited for the comparison study. All were between the ages of 21 and 27, 6 identified as male and 4 identified as female. All ten participants reported no prior experience with VR. All reported regular (weekly-daily) engagement with traditional video games. Participants were recruited informally

Fig. 2. Puzzle Room from above



from the researcher's personal network and no compensation was provided. Biases and limited demographic variance are acknowledged.

3.3 User Study Procedure

Participants first completed an introductory VR acclimation session lasting up to 10 minutes. They were shown how to move and manipulate objects in the Meta Quest 3 and how to adjust the headset.

Each participant then completed four trials in a fixed order in the VR puzzle room:

- Trial 1: Manual control. 10 minute time limit. No performance pressure. Completion time, interaction type and frequency were recorded.

The first trial functioned as a way to reduce trial order performance bias. After completing the first trial at their own pace while learning how to complete the tasks the participants were then told to attempt to complete the subsequent trials as fast as they could.

- Trial 2: Manual control repeat. 5 minute limit. Pressure to complete the trial as fast as they could. Completion time, interaction type and frequency were recorded.
- Trial 3: Scripted automation. No manual input allowed. 5 minute limit. Pressure to complete the trial as fast as they could. Completion time was recorded.
- Trial 4: Reinforcement learning model. No manual input allowed. 5 minute time limit. Pressure to complete the trial as fast as they could. Completion time was recorded.

After the trials, participants completed a preference ranking and provided written comments.

3.4 Training the Model

The reinforcement learning system was developed using Unity ML-Agents with the PPO algorithm. Training consisted of three stages:

(1) Stage 1 Bounded Constraint: the model was rewarded for keeping the panel within the room's limits and outside of the central column. Penalties were given for exiting the bounds or entering the central column. The model was given access to boundary information. Success in this stage was defined through behavioral analysis of the model: keeping the panel within the bounds of the room for ten minutes (the maximum allowable time per trial in the user study).

(2) Stage 2 FOV Alignment: the model was rewarded for placing the panel within the player's field of view and penalized for being out of view or rotated so it was not facing the player when it was in the field of view. The model was given access to the player's gaze/camera information. Success in this stage was defined through behavioral analysis of the model: remaining within the player's FOV for ten minutes.
(3) Stage 3 Context Based Movement: the model was rewarded for optimal panel placement while the player was stationary, moving, facing the block sliding puzzle while it was unsolved, holding the key card, or entering the code. It was penalized otherwise. The model was given access to the player's gaze and position as well as game state information such as if block sliding puzzle had been solved or not.

Training was conducted using a mix of single environment training and parallel environment training. A total of 350+ sessions were executed, each lasting 30-120 minutes of in game time. Approximate cumulative training time exceeded 400 in game hours. Despite progress in the first two stages with the model learning to stay in bounds and within the player's field of view, behavioral regression was observed in stage 3 with the model unlearning the bounding constraints and FOV alignment policies.

4 Results

4.1 Task Completion Time

Table 1 shows the user study timing results. The introductory session column displays how long each player took for their introductory session - rounded. The Trial 1 column shows the time remaining from the 10 minutes given to complete the trial. The other Trial columns (2-4) show the time remaining from the 5 minutes given to complete the trial - the higher the number the faster the completion. Manual control (Trial 2) saw the highest average remaining time (3:31, SD=65.54), followed by scripted control (Trial 3) (3:25, SD=69.4). The RL model (Trial 4) averaged 2:11 remaining (SD=51.72) with five DNF cases due to the panel positioning out of reach.

Table 1. Time Remaining (Minutes:Seconds) per Trial

Participant	Introductory Session	Trial 1 Remaining	Trial 2 Remaining	Trial 3 Remaining	Trial 4 Remaining
P1	1:00	8:12	4:21	4:03	1:37
P2	2:00	7:43	3:33	3:21	0:00
P3	5:00	8:22	4:32	4:13	2:01
P4	2:00	6:40	1:45	1:32	0:00
P5	3:30	6:55	1:14	1:11	0:00
P6	5:00	3:48	3:12	3:58	1:45
P7	2:15	5:53	4:07	4:17	1:42
P8	1:00	7:11	4:21	4:33	3:54
P9	7:30	8:23	4:18	4:22	0:00
P10	2:30	7:00	3:50	2:40	0:00
Mean	-	-	3:31	3:25	2:11
Std Dev	-	-	65.54	69.4	51.72
DNF	-	-	0	0	5

4.2 Panel Interaction Information for Trials 1 and 2

Table 2 summarizes the manual control interactions for each participant during trials 1 and 2. Voice is the number of times the "move" command was used, Grab (Large) was the number of times the player moved the panel from one place in their field of view to a spot that was previously outside their field of view (generalizing large movements of the panel), and Grab (Small) was the number of times the player moved the panel from one place in their field of view to another (generalizing fine adjustments of panel placement). This data reflects these participant's biases more than control method preference, with several participants using voice input inconsistently due to its novelty and the social comfort afforded during the trial. Other outliers such as P4's high amount of small grabs reflected the participants' curiosity with VR interactions due it being their first experience with VR - which similarly skewed the usability of the data due to participant biases.

Table 2. Number of UI Panel Movements by Type (Manual Trials)

Participant	Trial	Voice	Grab (Large)	Grab (Small)
P1	Trial 1	13	2	4
P1	Trial 2	2	1	2
P2	Trial 1	4	2	3
P2	Trial 2	4	0	0
P3	Trial 1	8	1	0
P3	Trial 2	2	1	0
P4	Trial 1	2	4	17
P4	Trial 2	0	4	25
P5	Trial 1	1	6	5
P5	Trial 2	0	4	3
P6	Trial 1	1	3	7
P6	Trial 2	2	5	2
P7	Trial 1	4	0	3
P7	Trial 2	6	0	0
P8	Trial 1	8	2	4
P8	Trial 2	8	1	1
P9	Trial 1	1	8	3
P9	Trial 2	0	4	4
P10	Trial 1	0	2	3
P10	Trial 2	0	4	6

4.3 Participant Preference Rankings

Following the study participants ranked the three control methods and their preferences were analyzed using the Friedman test. Rankings were 1 (most preferred), 2, and 3 (least preferred).

- Manual: Rank 1 by all participants (mean rank: 1.0)
- Automated: Rank 2 by 9 participants, Rank 3 by 1 (mean rank: 2.1)
- Adaptive: Rank 3 by 9 participants, Rank 2 by 1 (main rank: 2.9)

Test statistic = 18.2
p = 0.0001117
Effect size W = 0.91
This test confirms that a statistically significant difference in preference rankings across the control methods.

4.4 Participant Comment Results

Participant feedback reinforced the qualitative findings with the automated controls and adaptive controls receiving complaints in the written feedback. The scripted control was noted by four participants to be poor due to it obstructing their view. The adaptive control was consistently described as frustrating and unpredictable. The only participant who ranked adaptive control above automated cited the erratic nature of the model being "fun to chase."

4.5 RL Model Performance

The RL model was slow to learn during the first two stages due to the unconstrained 3D environment it had access to: it required over 70 training sessions before it saw success in stage 1 and over 100 additional sessions before it saw success in stage 2. Over 200 training sessions were conducted for stage 3 and with the new reward structure being put in place the model began to regress in behavior, losing its ability to stay within the player's FOV and behaving unpredictably resulting in the 50% DNF rate.

The observed behaviors of the model are also reflected in the qualitative analysis of the model. The model's entropy represents how random the model's decision making is at any given moment. In successful training this entropy value will trend downwards as the model creates policies and begins relying on them over random decision making. Policy loss is a representation of how those learned policies change over time, in successful training this value should trend downwards, though local fluctuations are expected as the model fine tunes its policies. Entropy and random decisions are what allow the model to learn the policies while policy loss shows the robustness of those learned policies.

The below figures are the average entropy and policy loss functions for models trained up to the model that concluded Stage 2 of training.

Fig. 3. Early Training Stages Entropy

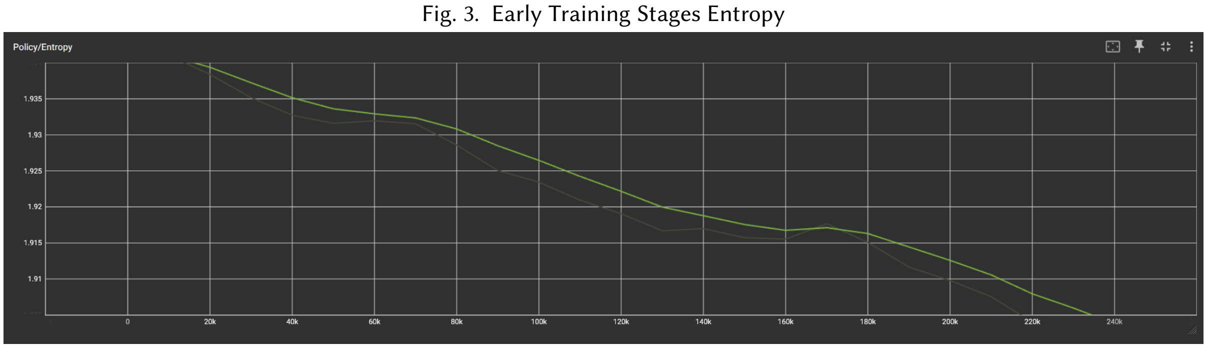
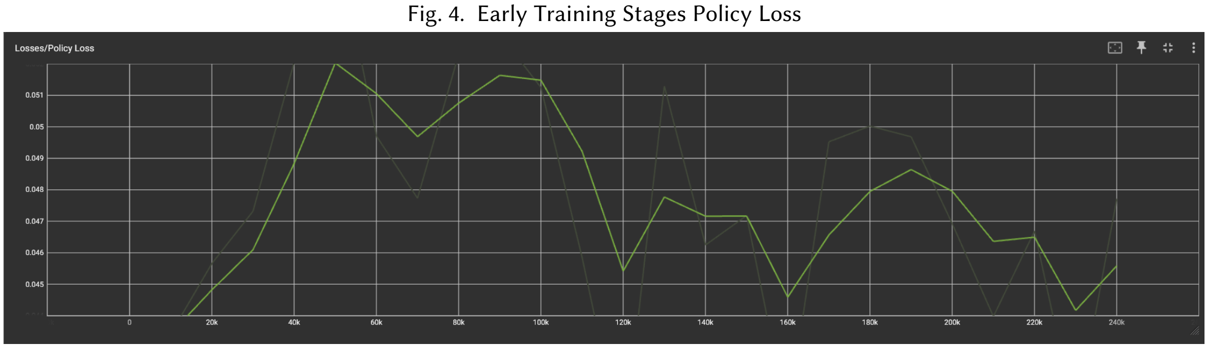


Fig. 4. Early Training Stages Policy Loss



The below figures are the average entropy and policy loss functions for models trained from the model that concluded Stage 2 up to the model that was used in the user study.

Fig. 5. Late Training Stages Entropy

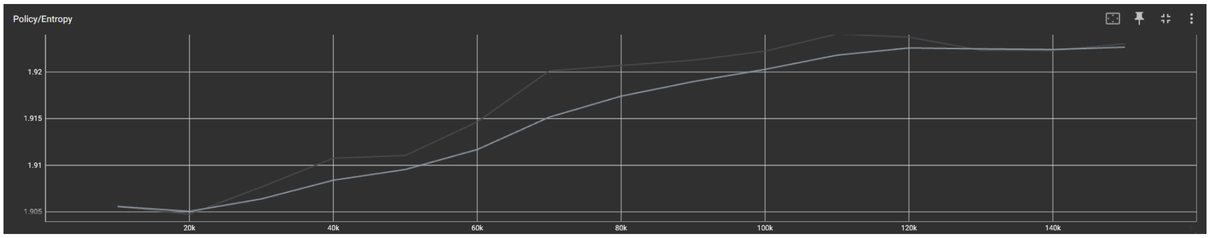
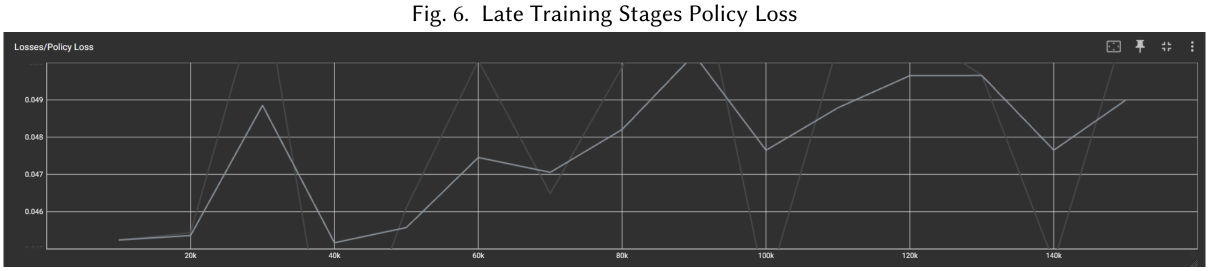


Fig. 6. Late Training Stages Policy Loss



5 Discussion

5.1 User Study Results

Manual control outperformed both scripted and adaptive methods in completion time and user preference. Users consistently ranked it the highest and it yielded the fastest average completion time. The manual controls offered direct manipulation and user agency. The lack of comments made about the manual controls may point to the participants' being inherently comfortable with them possibly due to their prior familiarity with video games.

Scripted control ranked second in both performance and preference. While it reduced the need for interaction with the panel, its fixed positioning introduced interference with some tasks and exploration. Four participants reported frustration when the panel obstructed their view of the puzzle wall and while exploring. Scripted control lacks user agency and introduces rigidity, reducing user satisfaction despite allowing participants to complete the tasks in a similar time frame as manual controls.

Adaptive control failed on all metrics. Participant feedback described it as unpredictable and pain to work with due to the lack of agency and a lack of predictability. It caused 5 DNFs in the trial due to positioning itself in an unreachable spot. The only positive remark (being the model was fun to chase) highlights its novelty more than its utility.

The significant differences between all three methods, as confirmed by the Friedman test (p=0.0001, W=0.91), reinforce the conclusion that user agency and predictability are primary drivers of user satisfaction in VR interface manipulation.

5.2 Machine Learning Model Implementation Failures and Design Implications

The reinforcement learning model failed due to poor reward structures in later training stages which introduced contradictory behavior constraints, and insufficient training time to learn them. While the agent successfully learned basic behaviors such as remaining within bounds and maintaining field of view alignment, its performance collapsed when context sensitive and higher variable rules were introduced. For example, after being trained to keep the panel in view, the model was later penalized for obstructing the player's field of view during the sliding block puzzle - effectively contradicting its earlier objective with the added nuance.

The model required over 170 training sessions to learn basic spatial constraints using four parallel environments. In contrast, only 200 sessions were able to be completed when training the more complex behaviors, a comparatively insufficient amount of sessions given the added complexity and eventual onset of regressive behaviors.

These results suggest that reinforcement learning for VR UI control requires both careful reward design and extended training sessions to achieve stability. Without these conditions the learned behaviors may degrade, such as in this case, and the behavior can become unpredictable undermining usability.

6 Conclusion

This study evaluated three control methods for managing a floating UI panel in a VR puzzle room: manual multimodal input, scripted automation, and reinforcement learning-based adaptation. Manual input saw the highest task performance and user preference, showing how user agency and increased options for control schemes remains a reliable method for input design in VR. Scripted automation achieved comparable performance but was hindered due to its lack of context sensitivity and user agency. The reinforcement learning system failed to deliver functional behavior under complex interaction conditions, exposing the fragility of the RL approach taken in high variance, context dependent VR tasks without proper training.

References

[1] Ryan Khushan Ghamandi, Ravi Kiran Kattoju, Yahya Hmaiti, Mykola Maslych, Eugene Matthew Taranta, Ryan P. McMahan, and Joseph LaViola. 2024. Unlocking Understanding: An Investigation of Multimodal Communication in Virtual Reality Collaboration. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 602, 16 pages. doi:10.1145/3613904.3642491

[2] Xiaotian Liu, Victoria Armstrong, Sara Nabil, and Christian Muise. 2021. Exploring multi-view perspectives on deep reinforcement learning agents for embodied object navigation in virtual home environments. In *Proceedings of the 31st Annual International Conference on Computer Science and Software Engineering* (Toronto, Canada) *(CASCON '21)*. IBM Corp., USA, 190–195.

[3] Feiyu Lu, Mengyu Chen, Hsiang Hsu, Pranav Deshpande, Cheng Yao Wang, and Blair MacIntyre. 2024. Adaptive 3D UI Placement in Mixed Reality Using Deep Reinforcement Learning. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI EA '24)*. Association for Computing Machinery, New York, NY, USA, Article 32, 7 pages. doi:10.1145/3613905.3651059

[4] Ziyue Yuan, Yu Liu, and Lingyun Yu. 2022. MEinVR: Multimodal Interaction Paradigms in Immersive Exploration. In *2022 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. 85–90. doi:10.1109/ISMAR-Adjunct57072.2022.00026

[5] Shuwen Zhou, Wenxuan Zheng, Yang Xu, and Yingchia Liu. 2024. Enhancing User Experience in VR Environments through AI-Driven Adaptive UI Design. In *Journal of Artificial Intelligence General Science (JAIGS)*. USA, 59–82. doi:10.60087/jaigs.v6i1.230