DTS311TC FINAL YEAR PROJECT

Player-Aware Intelligent Monitoring and Operations Navigator

Proposal Report

In Partial Fulfillment
of the Requirements for the Degree of
Bachelor of Engineering

Student Name	:	Taimingwang Liu
Student ID	:	2037690
Supervisor	:	Xihan Bian

School of AI and Advanced Computing
Xi'an Jiaotong-Liverpool University

November 2025

Abstract

Apply the font of Times New Roman to the paragraphs of the abstract using font size of 12. An abstract is usually one to three paragraphs long with a length of 150 to 350 words.

Contents

1 Introduction **1**

2 Literature Review **3**

2.1 Feasibility of Human-Homomorphic Interfaces 3

2.2 Evaluation Protocols & Performance Metrics 4

2.3 Agentic Modules and Stability in Long-Horizon Tasks 4

2.4 Learning Paradigms for Robust Agent Training 5

2.5 Challenges and Future Directions 6

2.6 Synthesis and Gaps in the Literature 6

References **8**

Appendix A. Title of Appendix A **I**

A.1 Appendix Heading 1 I

A.2 Appendix Heading 2 I

A.3 Appendix Table and Figure Captions I

Appendix B. Title of Appendix B **I**

1 Introduction

The demand for **interactive and companion experiences** in real-time entertainment is on the rise. Players are no longer satisfied with simple automation or static overlays; they seek dynamic, engaging partners that offer meaningful interaction. As gaming experiences evolve, the focus is shifting towards assistants that not only provide practical help but also enrich the player's journey through companionship and engagement.

Games like AI2U: "With You 'Til The End" are already capitalizing on the demand for interactive experiences, offering players the novelty of engaging with generative AI characters [1]. At the same time, major companies such as NVIDIA, with its Avatar Cloud Engine (ACE), and Ubisoft, with its "NEO NPCs", are advancing foundational technologies to create autonomous agents that enhance gameplay by offering more than just conversation [2], [3]. These developments demonstrate the growing commercial viability of AI-driven experiences, supported by increased player engagement and media attention [4].

The rise of AI-driven virtual streamers, especially the Neuro-sama phenomenon, highlights a significant shift in both technology and community-driven commercialization. Neuro-sama, an AI-powered VTuber, engages in real-time conversations and dynamic gameplay, capturing the attention of a wide audience [5]–[7]. Although Neuro-sama remains closed-source, its success has sparked a vibrant open-source ecosystem, with developers working to replicate or expand upon its capabilities [8]–[10]. This technological shift has been met with strong commercial traction on platforms like Twitch, underscoring the growing market demand for AI that combines both utility and companionship [11].

The success of these AI-driven personalities validates the market's desire for this combination of utility and companionship; this project builds upon this principle, seeking to translate this proven engagement from a broadcast-entertainment model into a player-centric, companion-style assistant. This assistant is designed to augment, not automate, the player's agency. Its role is distinct from broadcast-focused AI-Vtubers and player-replacing automation bots; it is scoped as a persistent, in-game partner that uses a **full voice-loop** to provide support that is both relational and functional. The system's core capabilities are threefold: offering **proactive, contextual companionship** by using in-character awareness to identify opportunities (e.g., spotting missed items); engaging in **collaborative problem-solving** to understand player goals and access external knowledge (e.g., researching online strategies); and supporting **on-demand task delegation**, allowing the player to hand over control for specific, well-defined tasks (e.g., 'explore this area while I'm away').

The technical scope to achieve these capabilities involves three key techniques. First, the system will use **a Unified Action Interface via GUI (GCC)**, a human-homomorphic interface with a screen-in, keyboard/mouse-out paradigm. This approach eliminates the need for game-specific APIs and ensures cross-platform adaptability [12], [13]. Second, actions will be managed through **Constrained Action Generation with Structured Output**. This method, which

formats commands from a predefined set of valid actions (e.g., 'move forward') into structured output (e.g., JSON), reduces errors like hallucinations and ensures that delegated tasks are legal and reproducible [14]. Third, the system will employ **a Low-Coupling Orchestration** (i.e., modularization by MCP-style). This plug-and-play approach is crucial, ensuring both scalability for integrating new skills like problem-solving and task-delegation, and testability which enables systematic ablation studies to evaluate each component's contribution [15].

With a clear understanding of market demand and technological feasibility, this project aims to develop a companion-style assistant that seamlessly integrates into gameplay. By demonstrating a system capable of this deeper, functional partnership—blending both companionship and shared agency—this work will enhance the overall player experience through dynamic, interactive assistance.

2 Literature Review

This section reviews the foundational research on AI agents for complex, interactive tasks, which underpins the development of the proposed game companion. It first establishes the technical feasibility of using human-homomorphic interfaces for game-agnostic control. Following this, it examines the evaluation protocols and performance metrics critical for benchmarking such systems. The review then explores the core **agentic chain**—a synthesis from recent literature comprising **planning, skills, reflection, and memory**—necessary for maintaining stability in long-horizon tasks. Subsequently, it details the modern learning paradigms required for robust agent training, before concluding with a discussion of persistent challenges, such as latency and safety, and a synthesis of the key literature gaps this project aims to address.

2.1 Feasibility of Human-Homomorphic Interfaces

Achieving the proposed unified action interface requires grounding the agent in **human-homomorphic interfaces**, a paradigm often referred to as General Computer Control (GCC). The feasibility of this approach is strongly demonstrated by Tan et al. in the **CRADLE** framework, which introduces the "Screen as input, keyboard/mouse as output" paradigm. This proves that a unified agent can perform complex, long-horizon tasks across different software, including games, without relying on specialized APIs [12]. This pipeline, proceeding from **screenshot to structured action**, is further validated by Gu et al., who use Reinforcement Fine-Tuning (RFT) to successfully train agents on GUI tasks, reinforcing that GUI-based control is a viable and powerful paradigm [13].

While feasible, this approach requires mechanisms to ensure stability. The **ORAK** benchmark, introduced by Park et al., highlights the importance of using a restricted valid-action set, or **Legal Move Constraints**. This feature, which greatly constrains the model's output space, is a critical optimization that can reduce action errors by 15-20% [15]. The authors also emphasize that standardized **Move Representation Formats** and **State Representation Formats** are non-trivial and essential for cross-game generalization. This includes standardizing how actions are defined and what contextual state (e.g., agent position, task progress) is provided to the agent [15].

On the input side, the ORAK benchmark is explicitly designed to evaluate agents using three different **input modalities**: Text-only, Image-only, and Text + Image, allowing for direct comparison [15]. This is a critical area of research, as the **V-MAGE** benchmark from Zheng et al. demonstrates that **vision-only inputs** pose significant generalization challenges for game-playing agents, particularly in handling novel scenes or tracking objects [16].

2.2 Evaluation Protocols & Performance Metrics

To rigorously measure the performance of these agents, a new generation of evaluation frameworks has been established. Proper benchmarking is essential, with systems like ORAK and LMGame-Bench providing comprehensive frameworks for ensuring reliable results across diverse game types [15], [17]. A primary outcome from these evaluations is the poor performance of current models on out-of-distribution (OOD) tasks. The study by Guruprasad et al. notes this as poor OOD contextual generalization [14], and the V-MAGE assessment concludes that current MLLMs exhibit significant limitations, highlighting this as a major challenge [16].

Within these evaluations, researchers rely on a suite of metrics. General metrics such as `pass@k` are commonly used to measure success over repeated trials. More specific to GUI agents, metrics like `Invalid%` track the proportion of invalid actions, a critical diagnostic for models struggling to produce valid outputs [14]. To properly evaluate agents on long-horizon tasks, benchmarks like ORAK also employ a mix of reward types, including **Dense Rewards** (for continuous feedback) and **Auxiliary Rewards** (e.g., for "correct format output") to guide and assess complex behaviors [15].

Beyond simple success rates, recent work has introduced more sophisticated systems for comparative and granular assessment. To address the difficulty of comparing models across different games, Zheng et al. introduce a **Dynamic ELO system** to standardize agent performance [16]. The same framework also moves beyond a single score by using "Unit Tests for Core Visual Abilities." This method provides a granular assessment of MLLM failures, identifying specific failure modes such as errors in **positioning**, **direction**, **identification** (e.g., seeing paths as obstacles), and a **conflict** between correct reasoning and incorrect execution [16].

2.3 Agentic Modules and Stability in Long-Horizon Tasks

For an assistant to be a reliable, long-term partner, it must employ a set of **agentic modules** to manage complex tasks and mitigate error accumulation. While single-step actions are relatively simple, maintaining stability over long-horizon tasks requires a robust architecture. Recent literature, including surveys by Hu et al. and Xu et al., has synthesized this into an "agentic chain" of planning, skills, reflection, and memory, which are essential for handling complex, sequential decision-making [18], [19].

The most critical of these modules are **planning and reflection**, which work in a continuous loop. The CRADLE framework provides a concrete example of this, defining a modular process for **"Info Gathering"** that informs **"Reflection,"** which is followed by **"Skill Curation," "Action Planning,"** and **"Task Inference"** [12]. This aligns with the OS-Agents survey, which also identifies adaptive planning based on **environmental feedback** as a cornerstone of modern agent design [20]. This feedback loop is heavily reliant on **memory**. Both short-term and long-term memory are critical for stability, whether implemented as memory scaffolding in LMGame-Bench or as a core component of task planning in the ORAK benchmark [15],

[17]. In practice, this includes recording actions (e.g., logs, navigation, form data) to provide a persistent context for the agent [20].

To make long-term planning tractable, agents also rely on a curated set of **atomic skills**. This concept is widely adopted, with the survey by Durante et al. identifying "Macros (or skills)" as a common feature [21]. The CRADLE framework implements this directly with its **"Skill Curation"** module, which dynamically generates and updates re-usable skills to handle complex, long-term tasks [12]. A key new direction for agents is **Self-Evolution**, or the ability to automatically assemble new skill-sets for novel tasks [18]. The UI-Venus framework provides a practical implementation of this with a **"Self-Evolving Trajectory History Alignment"** framework. This method acts as automated reflection, allowing the agent to refine its own reasoning and "thought-action pairs," which leads to "more coherent planning" [13].

2.4 Learning Paradigms for Robust Agent Training

A stable agent architecture must be supported by a robust training paradigm that bridges the gap between single-step imitation and sustained, interactive assistance. While many foundational models are trained via supervised imitation, this paradigm struggles with the long-horizon, multi-step nature of interactive tasks. Consequently, modern agent systems are increasingly applying **Reinforcement Fine-Tuning (RFT)** on top of pre-trained models. This approach, as demonstrated in studies by Gu et al., Xi et al., and Peng et al., has proven essential for adapting agents to complex, sequential decision-making [13], [22], [23]. As an alternative, offline learning paradigms like the **Decision Transformer (DT)** are also explored. The R2-Play framework from Jin et al., for example, uses this approach, framing the task as sequence modeling rather than reinforcement learning, which can be effective for learning from static datasets of expert behavior [24].

This shift to more advanced training is enabled by sophisticated reward shaping and iterative refinement. To overcome sparse rewards, systems employ a combination of **dense rewards** (feedback at each timestep) and **auxiliary rewards** (feedback for sub-goals or correct formatting). This reward structure is shown to accelerate learning and improve task performance, especially in complex, long-horizon scenarios [22], [23]. This is then combined with iterative **feedback loops** that allow the agent to refine its strategy over time, such as the "progressive interaction scaling" proposed in AgentGym-RL or the "self-evolving trajectories" of the UI-Venus framework [13], [22].

Finally, a critical component for reliability is the use of **structured action generation** and **grounded task execution**. To reduce errors, agents are trained to generate actions in a highly structured form (e.g., JSON) rather than as free-form text [13]. This is often combined with "Grounded" task execution, which, as noted by Durante et al., explicitly ties an agent's actions to the UI or game state, a method that helps to reduce ambiguity and prevent the model from hallucinating invalid actions [21].

2.5 Challenges and Future Directions

Despite the rapid progress in agent capabilities, several persistent challenges must be addressed to develop truly robust, real-time AI companions. The most immediate of these is **latency**, which is critical for a seamless user experience in interactive entertainment. Research into on-device processing, such as the UI-Venus framework, aims to solve the "unacceptably long inference latency" of large models [13]. This is a non-trivial requirement; studies in real-time human-AI coordination by Liu et al. have empirically demonstrated that latency beyond approximately 100ms is perceived as lag and significantly degrades the sense of fluid cooperation [25].

Beyond performance, significant questions of **safety and robustness** remain. For an agent to be trusted with any level of control, it must be able to handle dangerous actions and critical errors. The OS-Agents survey identifies "Error Recovery Mechanisms" (such as rollbacks) and "Human-in-the-Loop Control" (such as confirmation mechanisms) as major unsolved challenges for the field [20]. This need for alignment is a key research area, with work by Zubia et al. focusing on robustifying agents to ensure safe "transfer" to new situations and prevent unintended consequences [26].

A primary cause of such errors is the agent's difficulty with **out-of-distribution (OOD) generalization**. Both the V-Mage benchmark and the Benchmarking-VLA-VLM study identify OOD generalization as a critical failure point [14], [16]. The V-Mage benchmark, for example, highlights that vision-only models struggle significantly to adapt to new visual scenarios [16]. This finding is not limited to vision; the Benchmarking-VLA-VLM study found that all evaluated models (VLAs and VLMs) had "significant limitations in zero-shot generalization to OOD tasks," noting that their performance was heavily influenced by factors like action representation and prompt engineering [14].

Recent surveys, such as those by Tang et al. and the OS-Agents survey, also identify that current agents, while proficient in simple tasks, still struggle with two key failures: **long-horizon tasks suffer from error accumulation**, and agents exhibit low fine-grained precision, making it difficult to perform precise actions [20], [27]. These challenges collectively define the **future directions** for the field. As outlined in the CRADLE framework, research must prioritize enhancing an agent's ability to handle complex, multi-step tasks while ensuring real-time reliability. This requires enhancing **multi-modal capabilities**, improving **accuracy** in fine-grained control, and solving the "prohibitive inference latency" of current systems, all while ensuring agents remain resilient in unseen environments [12].

2.6 Synthesis and Gaps in the Literature

This review reveals that while significant progress has been made in functional agent control, a clear and actionable gap exists in the literature regarding unified, companion-style assistants. Current research is largely bifurcated, prioritizing either high-performance task comple-

tion or the fundamentals of relational AI, with a significant lack of academic frameworks that effectively merge the two. For instance, while AI systems show growing competence in task execution, recent work by Gamage et al., such as the **Emotion AWARE** framework, highlights that integrating "multi-granular and explainable" **emotional intelligence** remains a complex, unsolved challenge—a feature largely absent in current agent architectures [28]. This gap is evident in state-of-the-art functional systems; frameworks like UI-Venus have made strides in performance via RFT and structured actions, but their architecture is exclusively focused on functional task completion and does not address this relational component [13].

This project is therefore positioned to fill this gap by creating an assistant that architecturally blends both companionship and agency. In doing so, it will also address two other persistent challenges in the field: **real-time adaptability** and **cross-game transferability**. The former remains a recognized challenge, with benchmarks such as LMGame-Bench being specifically designed to test the limitations of current models in spatiotemporal and long-context reasoning [17]. The latter, the difficulty of creating **game-agnostic agents**, has been pointed out by numerous studies. Both the ORAK benchmark, which highlights the difficulty of evaluating agents across diverse games, and the CRADLE framework, which was built to solve the reliance on proprietary game APIs, confirm that cross-game transferability is a key hurdle for the field [12], [15].

By focusing on a human-homomorphic interface and feedback-driven learning, this project will directly contribute to these unsolved areas, demonstrating a system capable of the deeper, functional partnership that current literature lacks.

References

[1] AlterStaff, *Ai2u: With you 'til the end*, https://store.steampowered.com/app/2880730/AI2U_With_You_Til_The_End/, Accessed: 2025-10-10, 2025.

[2] NVIDIA. "Nvidia ace for games - autonomous game characters." Accessed: 2025-11-05, NVIDIA Developer. [Online]. Available: https://developer.nvidia.com/ace-for-games.

[3] Ubisoft. "How ubisoft's new generative ai prototype 'neo npcs' changes the narrative." Accessed: 2025-11-05, Ubisoft News. [Online]. Available: https://news.ubisoft.com/en-gb/article/5qXdxhshJBXoanFZApdG3L/how-ubisofts-new-generative-ai-prototype-changes-the-narrative-for-npcs.

[4] J. Kim. "Bringing personality to pixels, inworld levels up game characters using generative ai," NVIDIA Blog. [Online]. Available: https://blogs.nvidia.com/blog/generative-ai-npcs/.

[5] Vedal and Neuro-sama, *Neuro-sama official youtube channel*, https://www.youtube.com/@Neurosama, Accessed: 2025-10-10, 2022.

[6] StreamElements. "State of the stream: Twitch 2024 year in review." Accessed: 2025-10-11. [Online]. Available: https://blog.streamelements.com/state-of-the-stream-twitch-2024-year-in-review-ef4d739e9be9.

[7] "Q4 2024 global live streaming landscape." Accessed: 2025-10-11, Streams Charts. [Online]. Available: https://streamscharts.com/news/q4-2024-global-livestreaming-landscape.

[8] O.-L.-V. contributors, *Open-llm-vtuber: An open-source ai vtuber framework*, https://github.com/Open-LLM-VTuber/Open-LLM-VTuber, Accessed: 2025-10-10, 2025.

[9] moeru-ai, *Airi: Ai waifu / virtual character container inspired by neuro-sama*, https://github.com/moeru-ai/airi, Accessed: 2025-10-10, 2025.

[10] kimjammer, *Neuro: A local-model recreation of neuro-sama*, https://github.com/kimjammer/Neuro, Accessed: 2025-10-10, 2025.

[11] "Vedal's ai vtuber neuro-sama sets new twitch hype train world record." Accessed: 2025-10-11, Streams Charts. [Online]. Available: https://streamscharts.com/news/vedals-ai-vtuber-neuro-twitch-hype-train-record.

[12] W. Tan *et al.*, "Cradle: Empowering foundation agents towards general computer control," *arXiv preprint arXiv:2403.03186*, 2024.

[13] Z. Gu *et al.*, "Ui-venus technical report: Building high-performance ui agents with rft," *arXiv preprint arXiv:2508.10833*, 2025.

[14] P. Guruprasad, Y. Wang, S. Chowdhury, H. Sikka, and P. P. Liang, "Benchmarking vision, language, & action models in procedurally generated, open ended action environments," *arXiv preprint arXiv:2505.05540*, 2025.

[15] D. Park *et al.*, "Orak: A foundational benchmark for training and evaluating llm agents on diverse video games," 2025, arXiv:2506.03610. arXiv: 2506.03610.

[16] X. Zheng *et al.*, "V-mage: A game evaluation framework for assessing vision-centric capabilities in multimodal large language models," *arXiv preprint arXiv:2504.06148*, 2025.

[17] L. Hu *et al.*, "Lmgame-bench: How good are llms at playing games?, 2025a," *URL https://arxiv. org/abs/2505.15146*,

[18] S. Hu *et al.*, "A survey on large language model-based game agents," *arXiv preprint arXiv:2404.02039*, 2024.

[19] X. Xu *et al.*, "A survey on game playing agents and large models: Methods, applications, and challenges. arxiv pre-print," *arXiv preprint arXiv:2403.10249*, 2024.

[20] X. Hu *et al.*, *Os agents: A survey on mllm-based agents for computer, phone and browser use*, 2024.

[21] Z. Durante *et al.*, "Agent ai: Surveying the horizons of multimodal interaction," *arXiv preprint arXiv:2401.03568*, 2024.

[22] Z. Xi *et al.*, "Agentgym-rl: Training llm agents for long-horizon decision making through multi-turn reinforcement learning," *arXiv preprint arXiv:2509.08755*, 2025.

[23] Z. Peng *et al.*, "Improving agent behaviors with rl fine-tuning for autonomous driving," in *European Conference on Computer Vision*, Springer, 2024, pp. 165–181.

[24] Y. Jin *et al.*, "Read to play (r2-play): Decision transformer with multimodal game instruction," *arXiv preprint arXiv:2402.04154*, 2024.

[25] J. Liu *et al.*, "Llm-powered hierarchical language agent for real-time human-ai coordination," *arXiv preprint arXiv:2312.15224*, 2023.

[26] M. Zubia, T. D. Simão, and N. Jansen, "Robustifying RL agents for safe transfer through action disturbances," in *Proceedings of the BNL Conference on Artificial Intelligence and Machine Learning (BNAIC/BeNeLearn 2024)*, 2024. [Online]. Available: https://bnaic2024.sites.uu.nl/wp-content/uploads/sites/986/2024/10/Robustifying-RL-Agents-for-Safe-Transfer-through-Action-Disturbances.pdf.

[27] F. Tang *et al.*, "A survey on (m) llm-based gui agents," *arXiv preprint arXiv:2504.13865*, 2025.

[28] G. Gamage, D. De Silva, N. Mills, D. Alahakoon, and M. Manic, "Emotion aware: An artificial intelligence framework for adaptable, robust, explainable, and multi-granular emotion analysis," *Journal of Big Data*, vol. 11, no. 1, p. 93, 2024.

Appendix A. Title of Appendix A

A.1 Appendix Heading 1

Text of the appendix goes here

A.2 Appendix Heading 2

Text of the appendix goes here

A.3 Appendix Table and Figure Captions

In appendices, table and figure caption labels and numbers are typed in manually (e.g., Table A1, Table A2, etc.). These do not get generated into the lists that appear after the Table of Contents.

Appendix B. Title of Appendix B

Text of the appendix goes here if there is only a single heading.