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

1.1 Problem Setting & Motivation

本项目拟构建一个面向玩家的伴随式（*companion-style*）实时游戏助手：围绕事件提示、策略建议与语音交互闭环提供低时延（low-latency）体验与可复现评测（reproducible evaluation）。当前生态从*AI*游戏主播/虚拟角色到*LLM* 驱动的*NPC*/插件快速涌现，但大量实践依赖场景定制，统一的动作接口与可复现实验协议仍不充分；研究侧正通过统一评测/模块消融与污染控制/协议一致等方法学收敛评测与对比[1], [2]。本提案据此明确项目目标、范围与评测要素。

1.2 Scope & Working Definitions

本项目以**GUI**（**GCC**）通道为主要执行路径（*screen-in, keyboard/mouse-out* 的人类同态接口human-homomorphic interface），强调跨应用/游戏的通用性与可迁移性（portability）。代表性工作*CRADLE* 报告了在不依赖应用API 的前提下，通过规划/技能整理/反思/记忆的管线完成长链路任务的可行性[3]。 **MCP-style** 编排在本项目中被用作内部模块/技能的注册与路由（registration/orchestration）的通用思路；不涵盖针对具体应用/游戏的专用API 适配工作。 关于“统一评测/消融、plug-and-play”的组织可参考*ORAK*[1]；基于*procedural generation* 的OOD 方法学可参考*Benchmarking-VLA-VLM*[4]；把真实游戏“转化为可靠评测”的协议化实践可参考*lmgame*[2]。 在GUI 场景中，*UI-Venus* 展示了截图输入（*screenshot*）+ 结构化动作（*structured output*）的端到端导航路径[5]；*V-MAGE* 聚焦*visual-only/continuous-space* 的视觉中心评测[6]。相关工作将在第二部分系统梳理。

Screenshot-first GUI feasibility. 在真实平台上，“截图输入+ 结构化动作输出”路线已有具体成绩：UI-Venus 在AndroidWorld 报告**65.9%** *pass@1*，在ScreenSpot-V2/Pro 分别为**95.3%/61.9%** [5]，表明在不依赖A11y/DOM 的条件下亦可实现可比导航能力。

MLLM architecture at a glance. 图 1 给出多模态大模型（MLLM）的通用结构：文本经*tokenizer* 输入LLM，非文本模态由*Multimodal Encoder/Projector* 对齐后与文本融合[7]。

Survey cues. 更广义的“Agent AI”综述从“下一步具身动作预测（next-embodied action prediction）”出发，讨论外部知识、人类反馈与多传感输入在*grounded* 场景中的作用[8]；面向GUI 自动化的综述系统梳理了以LLM 为中枢的框架、*LAM*（large action models）、基准与指标[9]；另有针对(M)LLM-based GUI agents 的综述从感知—探索/知识—规划—交互四组成对齐术语与挑战[10]；在OS 视角，*OS Agents* 综述提出“环境/观测/动作—理解/规划/落地”的要素图谱[11]；端侧推理综述为“低时延”语境提供工程参考**ondevice-llm**；“LLM×游戏智能体”的方法与基准的更广泛总览可参见近期arXiv 工作**game-agents-large-models**。

Industry/Community signals. 作为动机与生态线索（非技术有效性论证），*Neuro-*

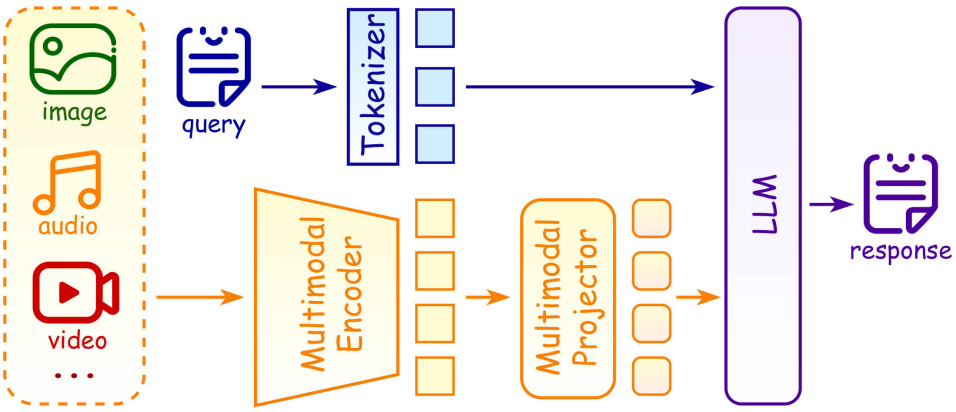

Figure 1: The overall architecture of MLLMs [7].

sama 的*AI streamer* 现象与相关开源生态（如*Open LLM VTuber*、*Airi Project*、*Kimjammer-Neuro*）反映出该方向的关注与活跃[12]–[15]；平台侧量化数据亦显示较大受众体量：Neuro-sama 于2025/01 创下**Hype Train Level 111**（约**85K** 付费订阅、约**1.2M** bits），Twitch 2024 年总观看时长约**18.5–20.8B** 小时、2024Q4 全行业约**21B** 小时[16]–[21]。

1.3 Design Principles & System Preview

Design principles：结构化输出（structured output）、可审计（auditability）、可复现（reproducibility）、低耦合（MCP-style 编排）。**System preview**（一句话流程）：*screen/audio* → 轻量*VLM* → *LLM/agentic*（planning/memory/reflection）→ *MCP-style* 技能注册/路由→ *GUI* 执行（kb/mouse）→ *safety*（permissions, rollback, kill-switch）。默认工程姿态（不作实现承诺）：single-flight（不并发大模型请求）、event-triggered（事件触发）、frame-window（3–5 帧堆叠）、text-first（可解析为结构化文本则优先文本路径），以降低延迟与方差。

1.4 Key Challenges

（i）长链路稳定性（**long-horizon stability**）：在GUI（GCC）通道下，错误累积与状态漂移更易放大；文献中的*skills*/反思/记忆管线可缓解，但挑战仍存[3]；（ii）视觉中心定位与记忆（**vision-centric grounding & memory**）：*visual-only*/*continuous-space* 对定位、时机、视觉记忆与高层推理提出更高要求[6]；（iii）**OOD** 与协议一致（**OOD & protocol consistency**）：需要在过程生成与变量可控条件下比较架构/数据/后处理，减少不可比性[4]；（iv）提示方差与污染（**prompt variance & contamination**）：把“游戏→评测”落到可复现协议需稳定交互回路并记录后处理[2]；（v）无效动作与幻觉（**invalid actions & hallucination**）：结构化输出/约束解码可降低invalid action，但*think–action mismatch* 等现象仍被报告[5]；（vi）时延与交互体验（**latency & UX**）：实时伴随式场景强调*voice RTT* 与帧到提示响应时间，需要与稳定性指标共同考量[1], [2]。

1.5 Project Objectives & Expected Deliverables

Objectives.

- 实现一个以*GUI*（*GCC*）为主的伴随式实时游戏助手原型，覆盖事件提示、策略建议与语音回路，面向低时延与可复现评测。
- 采用*MCP-style* 的内部编排思想组织*skills/macros*、*planning*、*memory*、*reflection*，在不依赖应用专用API 的前提下实现可插拔与可审计（auditability）。
- 明确一套小而可复现的评测要素（任务脚本与指标族），关注*advice adoption*、*voice RTT*、*macro success* 等体验相关量。

Expected Deliverables.

- 系统原型：屏幕采集与轻量感知、agentic 模块、MCP-style 技能总线、GUI 执行器、基础安全护栏（权限/回滚/急停）。
- 评测脚本与配置：可复现实验的任务脚本、指标计算与日志审计工具（含模块开关用于对照）。
- 使用文档与演示：安装/运行说明、配置模板与演示视频；范围外：针对具体应用/游戏的专用API 适配与大规模模型微调。

1.6 Assumptions & Out-of-Scope

Working assumptions. 默认采用"*single-flight* + *event-triggered* + *frame-window*（*3–5* 帧）+ *text-first*"的工程姿态以降低时延与方差；评测记录*post-processing* 与交互版本以保持协议一致。

Out-of-scope. 不开展每个游戏/应用的专用API 适配；不在本报告中承诺大规模端到端训练与数据采集；不涉及平台级增强权限（如A11y/私有DOM 钩子）的依赖。

2 Literature Review

2.1 接口可行性（Interface Feasibility: GUI/GCC）

要把“伴随式实时助手”落到真实游戏，首先要回答：不接专用**API**、仅用**GUI**（**GCC, screen-in & keyboard/mouse-out**）是否可行？代表性工作证明，在不依赖应用专用接口的条件下，通过“信息采集—规划—技能整理（macro/skill）—自反思—记忆”的管线，可以在桌面/游戏场景跑通长链路任务，从而确立了人类同态接口（*human-homomorphic interface*）的可迁移性与可复现性[3]。

进一步地，端到端实践展示了“截图 → 结构化动作（*structured output*）”的可落地路径：以受约束的动作模式替代自由文本，使执行更稳且便于审计（auditability）；同时，评测框架通过感知/记忆脚手架（scaffolds）稳定交互回路，把“游戏→评测”流程工程化[2], [5]。对比研究还提出合法动作映射与约束解码（**constrained decoding**）以显著降低无效动作（invalid actions），为后文的评测协议埋下方法学钩子[4]。既然“能做”已被验证，接下来就需要统一“怎么比”。

2.2 评测协议与实用要素（Protocols & Practicalities）

在统一协议方面，文献以*MCP-style* 的模块编排（orchestration）解耦代理与多游戏环境，统一配置与日志，在同一协议下做可复现的消融以比较*planning / reflection / memory / skills* 等模块；同时配套*leaderboard* 与*battle arena*，维持跨任务可比性[1]。为减少提示方差与避免训练—测试污染，评测侧引入Gym-style 接口与感知/记忆脚手架（harness），记录*post-processing* 并固定环境版本，报告在多模型下具有良好分离度的结果[2]。

在“无效动作治理”上，动作显式化是共同结论：将模型输出映射到合法离散动作空间，并采用结构化输出/约束解码联合抑制*Invalid%* 与坐标偏差；评估层面同时使用*Micro/Macro Precision/Recall/F1* 与*Brier/MAE* 兼顾分类与校准。端到端*GUI* 导航还在RL-finetune 中把奖励拆分为格式/动作类型/坐标/内容四粒度，将“合规性与细粒度正确性”纳入可学习信号，形成工程可落地的闭环[4], [5]。

为了与“伴随式实时体验”对齐，本文在客观指标（*success/pass@k*、*time-to-completion, TTC*、*misclick/rollback*、*latency*——含*voice RTT*）与*macro/micro* 并报之外，引入三项机会导向工作定义：**OAS**（*opportunity-normalized success*，成功次数÷可操作机会数）、**RT**（*reaction time per opportunity*，机会出现到首个有效动作/提示的时间）与**APO**（*attempts per opportunity*，每次机会的平均尝试/回滚次数）[1], [2], [4], [5]。在这套协议下，“谁在起作用”不再“口说无凭”，而是可以用可复现的模块消融实验来回答。

2.3 Agentic 模块（Planning / Memory / Reflection / Skills）

围绕长链路稳定性，研究把代理拆解为可组合的四要素：*planning*（任务分解

与策略选择）、*memory*（短长时与用户偏好）、*self-reflection*（纠错与风格一致）与*skills/macros*（原子到复合技能）。实践表明，该组合能缓解错误累积与状态漂移，支撑*GCC* 通道上的可迁移闭环[3]；而在统一协议下，可复现的模块消融给出了边际效应与搭配选择的证据，使“用/不用、强/弱”不再停留于经验判断[1]。

训练与数据侧的配套通常与结构化动作相匹配：通过*RL-finetune* 与反馈式调优（如*GRPO/RFT*），并以“格式/类型/坐标/内容”四粒度奖励，收紧输出空间同时提升细粒度正确性，从而在“反思—记忆—技能”的外圈外，再加一层可学习的约束[5]。此外，接口位形也影响模块产物：*VLM* 倾向输出文本/JSON，经由映射进入动作空间；*VLA* 则直接产出动作向量/分布，二者在无效率、校准与OOD 行为上的取舍需要在同一协议中对照评测[4]。要让这些模块更稳/更强，下一步问题就是：如何学习与调优。

2.4 学习范式（Learning Paradigms）

指令化与条件化提供了“看懂—再行动”的可解释路径：*R2-Play* 将多模态游戏指令（*multimodal game instructions, MGI*）并入*Decision Transformer*，用“游戏描述—轨迹—操作引导（含关键元素位置）”三段式模板共享跨任务知识，在多任务与泛化上报告相对优势，可作为提示/指导结构的模板参考[22]。对结构化动作任务而言，*RL-finetune* 配合四粒度奖励把“合法性+定位/文本输入”压进可学习信号，直接对齐 2.2 节中的“无效动作治理”目标[5]。

为在资源与稳态间取舍，工具增强型*MLLM* 的综述总结了外部工具（OCR/检索/计算/专家模型等）在*tool use / MCP-style* 编排下对延迟与鲁棒性的影响与边界，提示通过检索与轻量工具链分担大模型负载；而接口对照研究将*VLM*+映射 与*VLA*直出动作并置，指出两类路径在无效率、置信校准与OOD 表现上的差异，需要结合具体协议与数据分布权衡[4], [7]。即便“能做—能比—能学”已形成闭环，模型仍有系统性短板，必须在评测中直面。

2.5 能力边界（Capability Gaps）

以视觉为中心的自由形式游戏显示，多模态模型在若干关键能力上仍与人类存在差距：定位/追踪/计数、历史依赖与锚定偏置、时机控制、视觉记忆、文本识别与空间理解以及高层时序推理等维度普遍薄弱。采用Elo 风格相对强度排名与“模型/策略”的管线分离，从评测组织上揭示了这些系统性短板与不稳定性[6]。更通用的LLM×游戏评测同样显示，交互稳定性与污染控制会显著影响结果分离度，提示伴随式场景需要机会归一化与反应时等细粒度度量以及脚手架约束[2]。

2.6 术语与范围对齐（Glossary & Scope Alignment）

GCC（General Computer Control）：*screen-in + keyboard/mouse-out* 的人类同态动作接口；本文默认执行通道[9]. **LAM（Large Action Models）**：以结构化动作为一等

产出的模型族；本文作为对照范式引用[9]. **VLM vs VLA**：文本/*JSON* 输出经映射进入动作空间vs 直接动作向量/分布；评测统一采用合法动作映射+ 约束解码口径[10]. **Scaffold vs Orchestration**（**MCP-style**）：前者为评测期的稳定交互"脚手架"，后者为模块/工具的注册与路由；二者互补[11]. 指标口径：*pass@k*、*TTC*、*Invalid%*、*macro/micro* 并报；机会导向的**OAS/RT/APO** 作为伴随式场景的核心补充[8].

综上，文献给出的"可行—可比—可稳—可学—有边界"五段证据限定了本文后续系统与评测的口径：以***GCC*** 为统一接口，以结构化输出**+** 合法动作映射 降无效；以可复现协议与脚手架 做对照；并用***OAS/RT/APO*** 捕捉伴随式实时体验。

3 Project Plan

3.1 Proposed Solution / Methodology

3.2 Experimental Design

3.3 Expected Results

3.4 Progress Analysis and Gantt Chart

3.4.1 Risk & Ethics

4 Conclusion

References

[1] 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.

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

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

[4] 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.

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

[6] 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.

[7] W. An, J. Nie, Y. Wu, F. Tian, S. Lu, and Q. Zheng, "Empowering multimodal llms with external tools: A comprehensive survey," *arXiv preprint arXiv:2508.10955*, 2025.

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

[9] C. Zhang *et al.*, "Large language model-brained gui agents: A survey," *arXiv preprint arXiv:2411.18279*, 2024.

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

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

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

[13] 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.

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

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

[16] "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.

[17] P. Staff. "With valorant's help, this ai vtuber just beat a massive twitch record." Accessed: 2025-10-11, PCGamesN. [Online]. Available: https://www.pcgamesn.com/valorant/neuro-sama-twitch-record.

[18] "Vedal987 — streamer overview & stats." Accessed: 2025-10-11, TwitchTracker. [Online]. Available: https://twitchtracker.com/vedal987.

[19] C. Xiang. "Meet neuro-sama, the ai twitch streamer who plays minecraft, sings karaoke, loves art." Accessed: 2025-10-11, Bloomberg. [Online]. Available: https://www.bloomberg.com/news/newsletters/2023-06-16/neuro-sama-an-ai-twitch-influencer-plays-minecraft-sings-karaoke-loves-art.

[20] 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.

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

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