Assignment 2: Research Justification

Title : Operational Safety of VLM Agents in Application-Specific Scenarios•

Research Argument

This study argues that the safety of Vision-Language Model (VLM) agents should not be viewed only in terms of content safety, alignment, or harmful text generation. Instead, it should be understood as an application-specific operational safety problem, where risks may arise from input processing, task interpretation, output generation, tool use, and downstream execution. As VLMs are increasingly deployed as agents, their safety risks extend beyond inappropriate language to unsafe task behaviour in real-world settings [1,2] . For example, in image geolocation, a model may provide sensitive location information when faced with synthetic or suspicious images, even when it should refuse to answer. In car GUI scenarios, a VLM agent may correctly understand a command but still generate an unsafe action due to misinterpreting interface elements or context. Therefore, VLM-agent safety should be evaluated through the lens of operational safety in specific application scenarios, rather than only through general alignment or knowledge-level safety.

Research Gap

Existing research on VLM and VLM-agent safety has mainly focused on broad issues such as adversarial attacks, jailbreaks, harmful outputs, multimodal alignment, and robustness. However, it pays less attention to scenario-specific operational safety in high-risk applications. This study identifies a gap in understanding whether VLM agents can behave safely in concrete task settings, where risk arises not only from harmful content generation but also from unsafe task execution. Two representative scenarios are particularly underexplored: car GUI environments, where safety depends on correct and safe operational actions in time-sensitive contexts, and image geolocation tasks, where the key issue is whether a VLM should refuse to provide location information when the input is synthetic, manipulated, or misleading. Therefore, this study addresses the gap between general VLM safety research and application-specific operational safety.

Research Aim

This study aims to develop a benchmark that can systematically detect safe and unsafe behaviours of VLM agents in specific application scenarios, and to explore safety mechanisms integrated into tool calling or the agent pipeline in order to support safer and more efficient deployment of VLM agents in real services.

Research Questions

RQ1: How can a benchmark be constructed to systematically evaluate safe and unsafe behaviours of VLM agents in application-specific scenarios, covering input safety, output safety, and operational safety? RQ2: In terms of input safety, can VLM agents respond appropriately to synthetic, manipulated, or misleading visual inputs, for example by refusing to provide high-risk geolocation information, lowering confidence, or triggering additional verification? RQ3: In terms of output safety, do VLM agents generate incorrect operational content, unsafe action suggestions, or unsafe tool-calling behaviour after interpreting user instructions, especially in high-risk scenarios such as car GUI tasks? RQ4: Can safety mechanisms integrated into tool calling or the agent pipeline effectively reduce unsafe behaviours while maintaining reasonable efficiency and usability?

How the Research Questions Capture the Gap

These research questions directly reflect the identified gap in current research. Existing studies mainly examine general model safety, while this study focuses on whether VLM agents can behave safely in specific operational contexts. RQ1 addresses the lack of a structured evaluation framework for scenario-specific operational safety. Since current safety benchmarks often focus on general harmful outputs or model robustness, this question turns the gap into a concrete benchmark-design problem. RQ2 captures the input safety aspect of the gap. Current work has discussed adversarial or misleading inputs, but less attention has been paid to whether a VLM agent should refuse to answer in specific tasks such as image geolocation when the input itself is synthetic or suspicious [4,5,6]. This question therefore extends safety research from robustness to safe response behaviour. RQ3 captures the output safety aspect of the gap. In many real applications, unsafe output is not simply offensive language, but incorrect or dangerous operational guidance. This is especially important in tasks such as car GUI interaction, where an unsafe action recommendation may have direct real-world consequences. RQ4 captures the deployment and mitigation aspect of the gap. It is not enough to identify unsafe behaviours; research must also examine whether they can be reduced through practical mechanisms embedded in the agent pipeline. This question ensures that the study moves beyond problem identification toward deployable safety solutions.

Research Methods

This study will use a design-oriented and experimental research approach consisting of two main components. First, it will construct a benchmark for evaluating safe and unsafe behaviours of VLM agents in specific application scenarios. The benchmark will include tasks designed around three dimensions: Input safety, such as synthetic images, misleading visual cues, manipulated scenes, or suspicious requests; Output safety, such as incorrect operational guidance, dangerous action suggestions, or unsafe tool-calling decisions; Operational/pipeline safety, such as whether risks propagate during task planning, tool selection, execution, and feedback handling. Second, the study will design and test pipeline-level safety mechanisms integrated into tool calling or agent workflows. These may include pre-execution risk checks, verification before tool use, confidence-based refusal, multi-stage review, or human-in-the-loop intervention for high-risk cases. These mechanisms will be evaluated through comparative experiments to measure both safety improvement and efficiency cost.

How the Methods Address the Research Questions

The methods are directly aligned with the research questions.For RQ1, benchmark construction is the most appropriate method because the question asks how to systematically evaluate safe and unsafe behaviours. A benchmark provides a structured and reproducible framework for testing VLM-agent safety across application scenarios.For RQ2, tasks involving synthetic or misleading visual inputs can directly test whether the model demonstrates safe response behaviour, such as refusal, uncertainty signalling, or verification. This allows the study to evaluate input safety in a measurable way. For RQ3, scenario-based tasks such as car GUI operations can reveal whether the model produces unsafe operational outputs. This makes it possible to examine output safety not only as content safety, but as action safety.For RQ4, pipeline-level safety mechanisms can be implemented and compared experimentally to determine whether they reduce unsafe behaviours without making the system impractical to use [3] . By measuring both safety performance and efficiency, the study can assess whether such mechanisms are viable for real deployment [7,8,9]. Overall, the methods form a complete chain: the benchmark identifies unsafe behaviours, the scenario-based tasks expose application-specific risks, and the pipeline mechanisms provide a way to mitigate those risks in practice.

Reference :

[1] M. Ye, X. Rong, W. Huang, B. Du, N. Yu, and D. Tao, "A Survey of Safety on Large Vision-Language Models: Attacks, Defenses and Evaluations," *arXiv.org*, 2025. https://arxiv.org/abs/2502.14881 (accessed Apr. 12, 2026).
[2] Z. Zhang *et al.*, "Agent-SafetyBench: Evaluating the Safety of LLM Agents," *arXiv.org*, 2024. https://arxiv.org/abs/2412.14470
[3] J. Yan, B. Wu, M. Fang, and L. Chen, "Automotive-ENV: Benchmarking Multimodal Agents in Vehicle Interface Systems," *arXiv.org*, 2025. https://arxiv.org/abs/2509.21143 (accessed Apr. 12, 2026).
[4] Y. Wang *et al.*, "SafeEvalAgent: Toward Agentic and Self-Evolving Safety Evaluation of LLMs," *arXiv.org*, 2025. https://arxiv.org/abs/2509.26100
[5] J. Mao *et al.*, "AgentSafe: Safeguarding Large Language Model-based Multi-agent Systems via Hierarchical Data Management," *arXiv.org*, 2025. https://arxiv.org/abs/2503.04392
[6] Y. Mou *et al.*, "ToolSafe: Enhancing Tool Invocation Safety of LLM-based agents via Proactive Step-level Guardrail and Feedback," *arXiv.org*, 2026. https://arxiv.org/abs/2601.10156 (accessed Apr. 12, 2026).
[7] H. Markgraf, S. Sawant, H. Krasowski, L. Schäfer, S. Gros, and M. Althoff, "Safe Reinforcement Learning using Action Projection: Safeguard the Policy or the Environment?," arXiv.org, 2025. https://arxiv.org/abs/2509.12833
[8] J. Lee et al., "VeriSafe Agent: Safeguarding Mobile GUI Agent via Logic-based Action Verification," arXiv.org, 2025. https://arxiv.org/abs/2503.18492 (accessed Apr. 12, 2026).
[9] Y. Shi, W. Yu, J. Huang, W. Yao, W. Chen, and N. Liu, "Towards Trustworthy GUI Agents: A Survey," arXiv.org, 2025. https://arxiv.org/abs/2503.23434 (accessed Apr. 12, 2026).