Poster Abstract: LLM-Piloted Visual Privacy Agent on Mobile Systems

Yihong Hang, Hao Li, Fengxu Yang, Zhice Yang
School of Information Science and Technology,
ShanghaiTech University
Shanghai, China

Huangxun Chen
IoT Thrust, Information Hub, Hong Kong University of
Science and Technology (Guangzhou)
Guangzhou, China

ABSTRACT

The increasing use of camera streams on mobile systems has raised significant privacy concerns due to unauthorized visual data access by applications. Existing solutions either burden users with excessive interaction or lack semantic understanding of contextual privacy norms. This paper introduces *PrivacyAgent*, a novel visual privacy protection framework leveraging multimodal large language models (LLMs) to enable context-aware and fine-grained privacy control on mobile systems. PrivacyAgent intercepts camera streams via a virtualized I/O layer and restricts untrusted apps to privacy-compliant content with minimal user overhead.

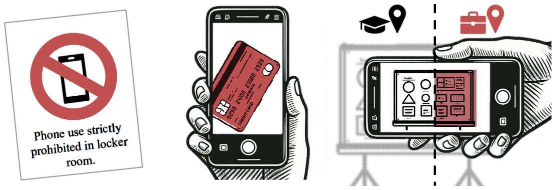
ACM Reference Format:
Yihong Hang, Hao Li, Fengxu Yang, Zhice Yang and Huangxun Chen. 2025. Poster Abstract: LLM-Piloted Visual Privacy Agent on Mobile Systems. In *The 23rd ACM Conference on Embedded Networked Sensor Systems (SenSys '25), May 6–9, 2025, Irvine, CA, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3715014.3724025

1 INTRODUCTION

The proliferation of camera-equipped mobile systems has enabled diverse visual applications like QR payments and object recognition. However, unauthorized camera access by apps (*e.g.*, TikTok's privacy scandals [4]) raises critical concerns about privacy violations. To address these challenges, a foundational step lies in establishing a clear understanding of "privacy-sensitive" visual content. Nissenbaum's theory of contextual integrity [5] defines privacy as adherence to situational norms – meaning effective visual privacy control mechanisms must first perceive contextual information (*e.g.*, environmental settings, user activities, data purpose) before evaluating compliance with context-specific privacy expectations (Figure 1).

Existing solutions face trade-offs: user-in-charged approaches [1] cause additional interactions, cognitive overhead, and even decision fatigue, while automatic models [6] lack semantic understanding of visual and non-visual contexts.

We propose a *PrivacyAgent* based on **LLM-Piloted Decision-Making**, which leverages multimodal large language models to reinterpret visual privacy control. LLMs analyze visual semantics alongside contextual information to assess privacy violations. Their capability stems from training on vast amounts of data, encompassing facts, common sense, and information from various domains.



(a) Out-of-Frame Context (b) In-Frame Context (c) Holistic Context

Figure 1: Visual Privacy is Context-Dependent. Whether an application can access certain visual content is determined by the context at the time the access request is made. This context is a set of information that includes both the visual data itself and relevant information outside of it.

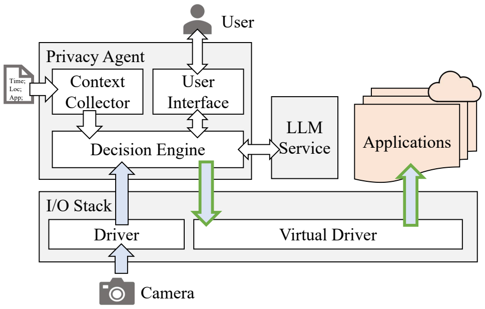


Figure 2: System Architecture.

The *PrivacyAgent* framework is structured to enable context-aware and fine-grained visual privacy control, which is shown in Figure 2. The framework leverages a virtual driver layer to intercept raw camera streams, isolating untrusted applications from direct access to sensitive visual data. Instead, the *PrivacyAgent* retrieves the visual frames directly from the camera driver and write the sanitized frames to the virtual driver based on the decision of *PrivacyAgent*. For the untrusted applications, only sanitized copies from the virtual driver is avaliable, which enforces visual privacy control for all applications.

In *PrivacyAgent*, there are three key components: a context collector aggregates contextual information for privacy element decision, a decision engine powered by a multimodal LLM to generate sanitized frames, and a user interface provides minimal interaction controls for users. In addition, the *PrivacyAgent* is implemented via a virtualized I/O framework that intercepts camera streams in user-space, enabling develop-able and upgradable real-time video processing without operation system modifications. The key contribution of this work is a mobile-deployable multimodal LLM-based *PrivacyAgent* which bridges AI reasoning with system security, offering scalable, context-aware privacy protection.

SenSys '25, May 6–9, 2025, Irvine, CA, USA

ACM ISBN 979-8-4007-1479-5/25/05
https://doi.org/10.1145/3715014.3724025

2 DESIGN

2.1 Context Collector

Since the visual frame information alone is not enough to help determine the context (Figure 1), a context collector aggregating necessary out-frame contextual data to help privacy assessments: **Location information** is derived from GPS coordinates and network-based localization, translated into semantic descriptors via digital maps. **Temporal context** includes absolute time from the system clock and calendar-derived contextual cues. **Application metadata** identifies the foreground app via Android's AccessibilityService and enriches it with functionality descriptions from Google Play. In addition to these, the device can provide many other types of contextual information for further enhance the context.

By fusing multi-modal data, the LLM-driven decision engine holistically interprets scenarios for further privacy decisions with contextual integrity norms.

2.2 Decision Engine

The decision engine is the core component of *PrivacyAgent*. The decision engine includes two main steps, the context-aware privacy assessment step and the fine-grained privacy control decision step.

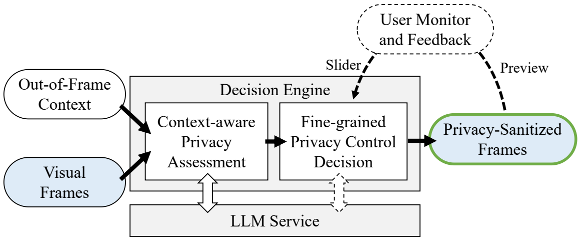


Figure 3: Decision Engine.

2.2.1 *Context-aware Privacy Assessment.*

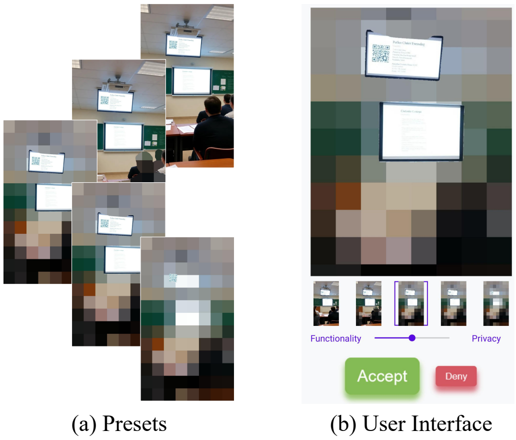
In this step, the engine takes the visual frames and the out-of-frame context collected by the context collector discussed in Section 2.1 as input for device's LLM service to assess each identifiable visual element in terms of their functionality relevance (an element's necessity for the requesting app's core operations) and privacy relevance (an element's sensitivity under privacy norms). Identified elements are spatially mapped to pixel regions via visual grounding and segmentation, enabling per-element granularity.

2.2.2 *Fine-Grained Privacy Control Decision.*

In this step, the engine first employs several preset strategies. The presets first accommodate different levels of functionality requirements based on the functionality relevance of each elements. Then the privacy-sensitive elements are restricted to provide intermediate options based on the privacy relevance. An example of the presets is displayed in Figure 4(a).

The presets with different privacy enforcing levels are heuristically designed for different intended purpose of visual content. For sharing purpose, the user may prefer revealing more content for the expected audience, while for non-sharing purpose, the user may want to reveal the content which strictly related to functionality, *e.g.*, QR scanning applications.

After that, the user can choose to refine the decisions via a bidirectional slider interface when the presets are sub-optimal due to incomplete contextual information, outlier personal preferences or noise from neural network models including hallucinations in LLMs or segmentation errors. When the functionality elements are not fully encompassed, we prioritize elements with higher functionality for expansion. When the privacy elements are not fully obscured, we prioritize elements with higher privacy for reduction. This heuristic-based interaction minimizes cognitive load while preserving fine-grained control. The user interface is shown in Figure 4(b).



(a) Presets

(b) User Interface

Figure 4: Example of Fine-Grained Privacy Control.

3 IMPLEMENTATION

PrivacyAgent is implemented across Windows, Linux, and Android platforms on different devices including Google Pixel 5 smartphone, Radxa Zero 3E single-board computer and desktops with Python and C++. On Windows, the `IMFVirtualCamera` API emulates a software-driven camera. On Linux and Android, the `V4L2Loopback` [7] kernel module is employed, while Android modifies the camera HAL to prioritize virtual devices over physical ones with root permission. The system relies on cloud-based ChatGPT-4o for privacy assessments and Grounding DINO [3] and Segment Anything [2] models for element visual grounding and segmentation.

REFERENCES

[1] Suman Jana, Arvind Narayanan, and Vitaly Shmatikov. 2013. A scanner darkly: Protecting user privacy from perceptual applications. In *2013 IEEE symposium on security and privacy*. IEEE, 349–363.

[2] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4015–4026.

[3] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. 2023. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499* (2023).

[4] Megan McCluskey. [n. d.]. TikTok Has Started Collecting Your 'Faceprints' and 'Voiceprints.' Here's What It Could Do With Them. https://time.com/6071773/tiktok-faceprints-voiceprints-privacy/.

[5] Helen Nissenbaum. 2004. Privacy as contextual integrity. *Wash. L. Rev.* 79 (2004), 119.

[6] Katarzyna Olejnik, Italo Dacosta, Joana Soares Machado, Kévin Huguenin, Mohammad Emtiyaz Khan, and Jean-Pierre Hubaux. 2017. Smarper: Context-aware and automatic runtime-permissions for mobile devices. In *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1058–1076.

[7] umlaeute. 2022. v4l2loopback. https://github.com/umlaeute/v4l2loopback.