WillmGlasses: An Open-Source LLM-Powered Smart Glasses Platform over Wireless Networks

1st Boyi Liu
HKUST & Pencheng Laboratory
Hong Kong & Shenzhen, China
bliubd@connect.ust.hk

co-1st Yongguang Lu
SYSU & Pengcheng Laboratory
Shenzhen, China
luyg5@mail2.sysu.edu.cn

2nd Wen Wu
Pengcheng Laboratory
Shenzhen, China
wuw02@pcl.ac.cn

3rd Jun Zhang
HKUST
Hong Kong, China
eejzhang@ust.hk

***Abstract*—Large language models (LLMs) are revolutionizing wearable augmented reality experiences by enabling AI-driven visual assistance. However, progress is significantly hindered by the absence of open testbeds for systematically studying network resource management and system optimization. We introduce WillmGlasses, an innovative open source platform that integrates wireless networks into llm-powered smart glasses, providing a real-world environment for interactive scenarios. Built on OpenAirInterface (OAI) and Open5GS, and orchestrated by an intelligent controller and FlexRIC, WillmGlasses offloads tasks to edge servers, allowing for unprecedented flexibility in performance under diverse bandwidth and latency constraints. By continuously monitoring network conditions, the platform not only optimizes physical resource block (PRB) allocation but also intelligently manages LLM outputs to ensure a consistent quality of experience. Ultimately, this comprehensive prototype enables advanced exploration of resource orchestration, multi-modal data processing, and user-level AI interaction. This effectively bridges the divide between theoretical design and the practical deployment of LLM-driven smart glasses in wireless networks.**

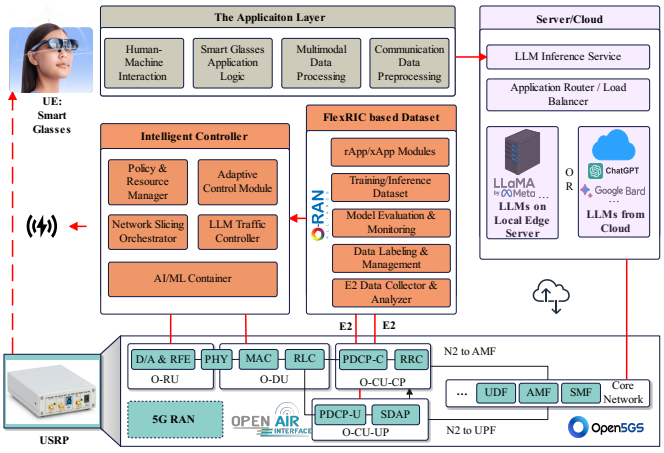

Fig. 1. WillmGlasses system architecture.

I. Introduction

Recent breakthroughs in LLMs have opened new avenues for wearable Augmented Reality (AR) devices, particularly smart glasses, enabling immersive user experiences. By overlaying intelligent assistance in real time, such as natural language interaction or visual object detection, LLMs p romise to transform the way users interact with both physical and digital environments. Yet, the lack of open and end-to-end testbeds has impeded systematic investigation of network resource management, Artificial Intelligence-driven adaptation, and system-level optimization under realistic deployment conditions.

This challenge is especially pronounced in wireless networks, where high throughput, low latency resource slicing and flexible application-layer adjustment are essential for advanced AR. Although Open RAN solutions exemplified by OpenAirInterface and Open5GS offer modular software-defined infrastructures, the integration of LLM-driven smart glass applications remains underexplored. Researchers often struggle to prototype comprehensive systems that traverse the entire pipeline, from wearable devices through radio access networks and core networks to the edge/cloud LLM inference, making it difficult to isolate bottlenecks and cross-layer collaborative design.

To address these challenges, we introduce WillmGlasses, the first open-source platform that seamlessly integrates smart glasses and LLMs over wireless networks. WillmGlasses leverages customized OAI [1] for the radio access network, Open5GS for the core network, and FlexRIC [2] for fine-grained performance monitoring. What sets WillmGlasses apart is its innovative system-level end-to-end intelligent control, which dynamically manages resource slicing, Physical Resource Block (PRB) allocation, and LLM output adjustments based on real-time network conditions. It not only offloads computation intensive tasks to edge servers but also enables adaptive control over various parameters, such as response length and processing priorities, ensuring optimal performance under diverse bandwidth and latency constraints.

WillmGlasses enables researchers to explore innovative network slicing and multi-modal data processing for enhanced AR interaction. Its unique AI-driven control bridges theory and practice, optimizing user experience via intelligent orchestration. The demo showcases LLM-powered smart glasses interacting with real-time network intelligence.

This work was supported by the Hong Kong Research Grants Council under the Areas of Excellence scheme grant AoE/E-601/22-R and NSFC/RGC Collaborative Research Scheme grant CRS_HKUST603/22.

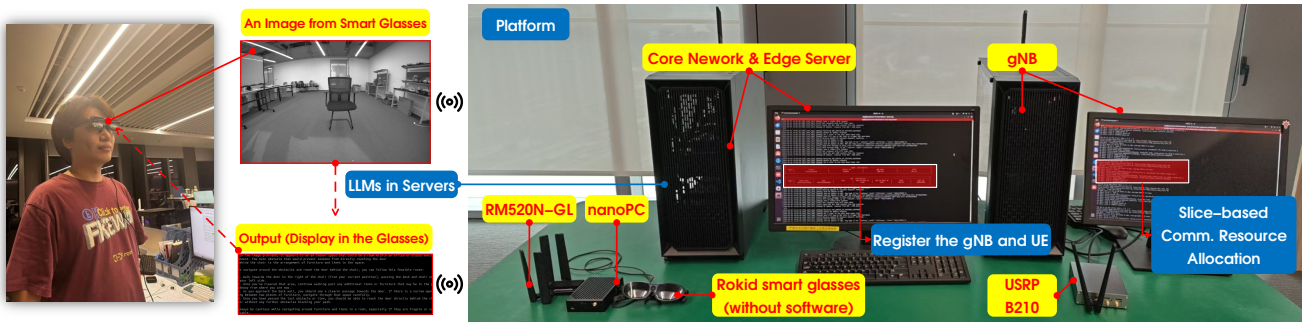

Fig. 2. The implementation of WillmGlasses.

II. System Implementation

Figure 1 illustrates the **WillmGlasses** framework, which connects *smart glasses* (i.e., UE) to a *5G RAN* (via **OAI**) and *core network* (via **Open5GS**), and then offloads **LLM inference** tasks to edge servers. A software-defined radio device **USRP** provides over-the-air transmissions, forming an end-to-end testbed for dynamic resource slicing and intelligent orchestration. In the following, the key components of the platform are detailed.

Application Layer: Running on the smart glasses, this layer (i) captures user inputs (e.g., video) through *human-machine interaction* and *multimodal processing* modules, (ii) performs *preprocessing* (e.g., compression), and (iii) offloads computation-intensive tasks to the LLM. This design keeps the UE lightweight and energy-efficient.

Intelligent Controller: A central *intelligent controller* coordinates end-to-end resource usage, enabling real-time **dynamic slicing** and **adaptive control**. Its key submodules include:

- **Policy & Resource Manager**: Allocates radio resources and enforces service-level constraints.
- **Adaptive Control Module**: Adjusts application parameters (e.g., video resolution, LLM response length) to maintain user QoS under congestion.
- **Network Slicing Orchestrator**: Customizes slices for LLM-driven traffic, connecting with OAI and Open5GS.
- **LLM Traffic Controller**: Steers queries between local edge servers or cloud based on real-time conditions.

FlexRIC-Based Monitoring: Following O-RAN principles, **FlexRIC** collects fine-grained performance metrics through *rApp/xApp* modules, such as *throughput*, *latency*, and *packet loss*. The data are fed back into the *Intelligent Controller*, enabling data-driven policy updates and adaptive optimizations.

Server/Cloud Infrastructure: We host the **LLM Inference Service** on *local edge servers*. An **Application Router/Load Balancer** distributes requests to appropriate LLM endpoints, aligning service-level agreements with network conditions.

5G RAN and Core Network: Underpinning these layers is the **OAI** stack (including O-RU, O-DU, O-CU) and the **Open5GS** core (including AMF, SMF, UPF), which are configured to isolate LLM traffic in a dedicated slice for precise performance evaluation.

III. Experimental Results

Figure 2 shows the hardware implementation of WillmGlasses. The smart glasses are connected to a local computing server and access the mobile network via a 5G module. It generates gesture-based frames (e.g., triggered by hand movements) at different time intervals and occasionally dispatches LLM queries to the LLaVA [3] model deployed on the edge server. A short time interval can incur high communication overhead and computation workload. We compare two modes: (1) a baseline allocation scheme, i.e., static PRB assignment, without dynamic LLM output adjustment and (2) an adaptive control scheme, i.e., real-time resource orchestration and on-the-fly LLM response tuning.

Table I shows that as gesture frequency increases, communication quality degrades, raising latency in the baseline scheme's static PRB allocation due to its inability to adapt. In contrast, our adaptive control adjusts PRBs and LLM output lengths to reduce overall latency. These results highlight the need for AI-adaptive cross-layer control for robust smart glasses interactions. WillmGlasses provides this capability for reliable AR applications over dynamic networks. More scenarios and data will be presented in future work.

TABLE I

Scheme	Inter.(frames/s)	Comm. (ms)	LLM (ms)	Total (ms)
Baseline	2	40	301	350
Adaptive	2	30	301	300
Baseline	4	58	345	403
Adaptive	4	50	300	350
Baseline	6	75	380	455
Adaptive	6	68	336	404
Baseline	8	90	420	510
Adaptive	8	82	368	450

* LLM denotes the latency in server. Comm. is the communication latency.

References

[1] N. Nikaein, M. K. Marina, S. Manickam, A. Dawson, R. Knopp, and C. Bonnet, "Openairinterface: A flexible platform for 5G research," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 33–38, 2014.

[2] M. I. Robert Schmidt and N. Nikaein, "FlexRIC: an SDK for next-generation SD-RANs," in *Proc. CoNEXT*, 2021, pp. 411–425.

[3] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," 2023.