Gunjoong Kim

Education

Yonsei University, M.S. in Computer Science and Engineering — Mar. 2024 – Feb. 2026
- GPA: 4.27/4.30

University of Seoul, B.S. in Computer Science — Mar. 2017 – Feb. 2024
- GPA: 3.85/4.50, Major GPA: 4.10/4.50

Experience

Research Assistant — Jan 2024 – Present
Mobile Embedded Systems Lab., Yonsei University, Seoul, Republic of Korea
- **Efficient On-device AI** Worked on advancing efficient on-device AI, exploring both model-level and system-level approaches to bridge the gap between heavy AI workloads and the limited resources of mobile devices. Focused on co-optimizing algorithms and runtime systems to enable responsive and reliable AI experiences in real-world settings.
- **Immersive Computing** Investigated system designs that support next-generation immersive applications including augmented reality and volumetric media, where high-quality perception and interaction are critical.
- Awarded Best Paper at MobiSys 2025 for the ARIA.

Publications(*Co-first authors)

Vega: Fully Immersive Volumetric Video Streaming with 3D Gaussian Splatting — *MobiCom'25*
Gunjoong Kim*, *Seonghoon Park**, *Jeho Lee*, *Chanyoung Jung*, *Hyungchul Jun*, *Hojung Cha*
- ACM Annual International Conference on Mobile Computing and Networking

ARIA: Optimizing Vision Foundation Model Inference on Heterogeneous Mobile Processors for Augmented Reality — *MobiSys'25*
*Chanyoung Jung**, *Jeho Lee**, ***Gunjoong Kim***, *Jiwon Kim*, *Seonghoon Park*, *Hojung Cha*
- ACM Annual International Conference on Mobile Systems, Applications, and Services
- **Best Paper Award**

viNPU: Optimizing Vision Transformer Inference on Mobile NPUs — *EuroSys'26*
Jeho Lee, ***Gunjoong Kim***, *Chanyoung Jung*, *Jaehee Kim*, *Seonghoo Park*, *Hojung Cha*
- European Conference on Computer Systems

Projects

viNPU: On-device Inference Optimization Framework — 2025
- Built a framework to run Vision Transformers efficiently on mobile NPUs by combining sensitivity-guided mixed precision with graph rewrites to keep data on-chip. Across devices, achieved average 11.5× faster inference, up to 44.9× lower energy, with near-no accuracy loss vs FP16.
- Led activation-pattern analysis to design a sensitivity-guided mixed-precision policy and applied it to the ViT model, deploying on real devices (Hexagon NPU).
- Tools Used: Qualcomm Neural Processing SDK (QNN), Aimet, Pytorch, ONNX, Android

Vega: Mobile Volumetric Video Streaming System with 3D Gaussian Splatting — 2024 - 2025
- Designed Vega, the first system to achieve 30 FPS full-scene volumetric video streaming on mobile devices with 3D Gaussian Splatting, leveraging object-level selective computation to reduce data size by 99% while maintaining high visual quality.
- Developed a mobile-friendly 3DGS video encoding scheme on the server and a mobile-optimized Gaussian Splatting renderer on the client, with a task scheduler distributing workloads across CPU/GPU/NPU to ensure smooth real-time playback on smartphones.

- Tools Used: PyTorch, OpenGL ES, TensorFlow Lite, Qualcomm Neural Processing SDK, Android

ARIA: Heterogeneous Processor Acceleration for VFM Inference 2024

- Developed ARIA, a system for real-time VFM inference on mobile devices by combining GPU full-frame predictions with NPU dynamic-region updates, achieving 99.9% deadline success and up to 50% accuracy improvement.
- Designed and implemented comprehensive baselines and conducted evaluations on a self-collected dataset tailored for real-world AR scenarios.
- Tools Used: Qualcomm AI Engine Direct SDK(QNN), LiteRT, ONNX, Android

Cosmos: Mobile 360° Video Streaming with Content-Aware Super-Resolution 2024

- Developed Cosmos, a content-aware SR system for real-time 360° video streaming on mobile, introducing CosmosNet with selective training and dynamic inference; sustained 30 FPS and improved tile quality by up to 21.8%p and PSNR by 3.0 dB over prior work.
- Designed and built an Android player with OpenGL ES, integrating viewport rendering and SR pipelines; optimized GPU/CPU workload (interpolation on CPU, SR on GPU) to sustain real-time performance under mobile constraints.
- Tools Used: PyTorch, TensorFlow Lite, OpenCV, OpenGL ES, Android

Skills

Languages: C++, C, Java, Python

ML Frameworks: Pytorch, ONNX, LiteRT(TensorFlow Lite)

GPU/NPU Frameworks: CUDA, OpenCL, OpenGL, Qualcomm AI Engine DIrect(QNN) SDK

Platforms: Android, Linux

Language: Korean(Native), English(Intermediate)