Xinhuiyu (Huiyu) Liu

San Jose, CA | xinhuiyuliu@gmail.com | (434) 242-9031 | linkedin.com/in/xinhuiyu-huiyu-liu | github.com/liuxhy

Education

Georgia Institute of Technology, M.S. in Computer Science — Expected Dec 2026
- **Coursework:** Graduate Algorithms, Operating Systems, Software Development Process, Machine Learning

University of Virginia, Ph.D. in Environmental Sciences — Aug 2019 – May 2025

Experience

Dyneti Technologies (seed-stage fintech startup) — Jun 2026 – Jul 2026
Software Engineer — *San Mateo, CA*
- Designed, tested, and deployed an asynchronous fraud-detection workflow for a credit-card scanning **mobile SDK**, cutting scan-to-response latency by over **75%** by decoupling server-side ML inference from the synchronous scan path
- Built a **server-to-server Fraud Decision API** allowing merchant backends to retrieve fraud results precomputed at scan time, eliminating a redundant second scan and query-time inference delay
- Defined the async lifecycle across a **Java/Kotlin** Android SDK and **Node.js** backend, covering API contracts, RDS-backed schemas, inference state transitions, and pending-result handling, while preserving backward compatibility for live partner integrations
- Owned releases end to end via **GitHub Actions CI/CD**, pairing automated unit/integration tests with multi-device **instrumented tests** covering async workflows, failure handling, and backward compatibility

IBSS Corp — Sep 2025 – May 2026
Software Engineer Intern — *Remote, US*
- Built TerraPrecise, a **Python/FastAPI** data platform ingesting forecast, station, and terrain data and serving location-specific weather risk products to React/Leaflet frontend clients and alerting workflows
- Designed the data layer around access patterns: **PostgreSQL/PostGIS** for subscriber and spatial metadata, **TimescaleDB** for forecast and observation time series, and **S3** for large gridded outputs
- Deployed a **Dockerized microservice architecture** on AWS isolating GPU-based downscaling inference from lightweight API and alerting services, with **Lambda** driving event-driven ingestion
- Designed a cloud-native geospatial serving layer (**eoAPI, STAC, COG, TiTiler**) making gridded datasets discoverable, queryable, and servable as scalable map layers
- Shipped an internal **Vertex AI** chatbot streamlining HR support workflows, automating its knowledge-base updates with scheduled **Cloud Run** jobs that sync documents from Google Drive to Cloud Storage with safeguards against accidental deletion

TGS (leading energy data analytics company) — May 2024 – Aug 2024
Software Engineer Intern — *Houston, TX*
- Built **parallel** data ingestion and transformation pipelines (Python/Xarray/Dask) on GCP processing **20 TB** of simulation and offshore sensor data for a wind-energy analytics platform
- Cut data processing time **70%** by redesigning **Zarr and Parquet storage layouts** and chunking strategies, letting analytics jobs handle larger datasets within fixed memory limits

University of Virginia — Aug 2019 – May 2025
PhD Researcher (Funded by NASA) — *Charlottesville, VA*
- Wrote parallel Python pipelines (Xarray/Dask) for ingestion, quality control, and feature extraction across **100+ TB** of satellite and model-derived data in **Unix/Linux** environments

Projects

Synchronized Distributed File System | `C++, gRPC, Protobuf, inotify, pthreads`
- Built a **gRPC-based distributed file system** with whole-file caching, CRC32 diffing, and timestamp-based conflict resolution under single-writer semantics
- Implemented server-side writer locks and deletion propagation via async callbacks, using **inotify** for local file monitoring and persistent gRPC streams for remote updates, with **mutexes** eliminating race conditions across concurrent client sessions

Multi-Agent Travel Planner | `Python, Streamlit, Google ADK, Gemini 2.5`
- Composed **8 Google ADK agents** into a sequential pipeline with hierarchical delegation, exposing flight, hotel, and activity agents as callable tools to a lead planner grounded on live weather and search APIs
- Built a bounded **critic-refiner loop** that validates itineraries against drive-time realism, transport consistency, and daily pacing, repairing violations before exporting a structured day-by-day plan to Excel

Skills

Languages: Python, C++, Java, Kotlin, JavaScript/TypeScript, SQL
Distributed & Systems: gRPC, Protobuf, concurrency (pthreads, mutexes), Dask, Docker, Unix/Linux
Backend & Web/Mobile: Node.js, Express, FastAPI, React, Android SDK, REST APIs
Cloud & Tooling: AWS (EC2, S3, Lambda, RDS), GCP (Vertex AI, Cloud Run, GCS), GitHub Actions CI/CD, Git
Data & ML: PostgreSQL, PostGIS, TimescaleDB, Xarray, Parquet, Zarr, PyTorch, scikit-learn, production ML inference