

Quan Tran

gmt1@rice.edu • <https://github.com/QuanTran255> • quan-tran.com

EDUCATION

Rice University - B.S. and Master of Computer Science (5th-Year Program)

Houston, TX

August 2022 - Expected May 2027 (GPA: 3.82/4.00)

Relevant Courses: Computer Vision, Quantum Computing, Software Methodology, Practical Machine Learning, Concurrent Program Design, Compiler Construction, Computer Systems, Tools and Models for Data Science, Probability and Statistics, Linear Algebra

SKILLS

Programming Languages: Python, Go, Java, C++, C, C#, TypeScript, JavaScript, Bash, R, Assembly, HTML/CSS, React

Cloud & DevOps: AWS, Azure, Docker, Jenkins, Git, CI/CD, Postman

Databases & Storage: NoSQL (MongoDB, Redis, CosmosDB), Relational (MySQL), GraphQL

Software Architecture & Systems: Object-Oriented Programming, RESTful APIs, Model-View-Controller (MVC), Client-Server Model

Data Science & Machine Learning: CUDA, TensorFlow, PyTorch, Pandas, Numpy, scikit-learn, Regression Models, NLP, RNN, CNN

WORK EXPERIENCES

Rice University Computer Vision Lab - Research Intern - May 2025 - November 2025

- Developed an attention weight bias mechanism with semantic segmentation that improves point tracking for rigid object by 5%
- Built a full-stack annotation tool to evaluate rigid object tracking performance, reducing manual annotation time by 30%
- Enabled dynamic point and line selection, real-time track visualization, and seamless integration with CoTracker3 and PIP++
- Designed backend architecture for loading video data, managing tracker configuration, and multi-model support

HP/Poly - Software Engineer Intern - May 2024 - August 2024

- Developed an end-to-end Python framework using Zoom Rest API for automated testing, enabling Zoom Rooms configurations before and after testing to validate and certify software for Zoom
- Integrated the framework with existing Jenkins pipeline to streamline setup and creation of new test cases with minimal configuration
- Collected and benchmarked Quality of Service data to evaluate live video performance in real-time test environments
- Refactored legacy codebase to double data retrieval speed and improve API efficiency

VNG Cloud - Software Engineer Intern - June 2023 - August 2023

- Built an ML anomaly detection system monitoring server traffic for 1M daily requests, improving incident response time by 30%
- Developed an online-learning time-series model that improved detection performance by 10% over baseline on seasonal traffic data
- Integrated a relational database to log and trigger user alerts on server traffic anomalies, with average delivery latency <3 seconds
- Collaborated with notification team to deploy the alert system within the existing monitoring infrastructure

PROJECTS

Crater Detection For Moon Navigation - August 2025 - Present

- Developed an instance segmentation model for lunar crater detection, achieving 50% accuracy improvement over baseline
- Optimized model memory usage by 65% vs. baseline, enabling efficient deployment on space-constrained hardware
- Improved inference speed by 30% vs. baseline, supporting near real-time moon navigation missions
- Built an automated labeling pipeline for 100,000+ craters across 3,700 images, eliminating 1,000 hours of manual annotation

Text-Based Game Editor and Engine - January 2025 - May 2025

- Led the UI/UX development team and coordinated design discussions to ensure seamless integration with backend systems
- Managed the database team by using CosmosDB and Redis to support scalable, real-time data storage
- Simulated 500 concurrent users during load testing to identify bottlenecks in CosmosDB and Redis-backed real-time operations
- Collaborated cross-team to integrate backend infrastructure with frontend editor features and conducted end-to-end testing

OwlDB - Custom NoSQL Database - August 2024 - October 2024

- Developed an extensible NoSQL database with RESTful APIs, supporting 100 concurrent connections
- Designed and implemented 10 robust structured APIs endpoints in Go to support CRUD operations
- Leveraged Go's concurrency features to handle high-volume requests efficiently
- Achieved 92% code coverage through extensive unit and integration testing, improving reliability

H.O.L.L.Y. - December 2023 - August 2024

- Engineered a productivity device transforming flat surfaces into interactive screens using Python-based computer vision
- Optimized computer vision pipeline to reduce gesture detection latency by 40%, enabling real-time interactions
- Implemented lens correction algorithms for accurate hand gesture tracking, improving tracking accuracy by 12% over baseline
- Integrated LLM (via HuggingFace), speech recognition, and text-to-speech technologies for voice-activated assistant