

YINGTONG YU

✉ Email: yingtongyujobs@gmail.com

📁 Portfolio: JARVISHHH.github.io

🐙 Github: <https://github.com/JARVISHHH>

Education

Brown University

Sept. 2022 – May 2025 (Expected)

Master of Science in Computer Science - GPA: 4.0/4.0

Providence, RI

Courses taken: Advanced Computer Graphics, 3D/2D Game Engines

Nankai University

Sept. 2018 – May 2022

Bachelor of Engineering in Computer Science - GPA: 90.12/100.00

Tianjin, China

Work Experience

Tencent

Sept. 2023 – Present

Game Engine Development Engineer Intern | C++, Python

Shenzhen, China

- Using C++, Python and Blueprint to implement plugins for Unreal Engine.

eBay

Jun. 2023 – Aug. 2023

Full-stack Software Development Engineer Intern | Scala

San Jose, CA

- Worked as a full-stack software engineer and implemented a new feature called Multiple Vacations.
- For the back end, implemented several new APIs and Classes using **Scala**, including APIs that get data from the database and pack them up for the front end, APIs that validate newly scheduled vacations, etc.
- For the front end, modified code in **Node.js** and **Marko** to show the list of vacations and a newly added button.
- Implemented a new batch using **Spring Batch** to manage events-producing for partner services.

ByteDance(TikTok)

Apr. 2022 – Jul. 2022

Back-end Software Development Engineer Intern | Golang

Beijing, China

- Reduced the latency of the packing part of the Suggestion Middle Page **from 160ms to 10ms** by reducing the number of RPC calls and parallelizing different processes, and increased the speed by about **1500%**.
- Refactored an entire API service, making it more readable and extensible.
- Added metrics and AB test, built **Grafana** dashboards to visualize performance.
- Implemented Pinyin fuzzy search and supported the proximity-based filtering with **Elasticsearch**.
- Integrated the new version of the recommendation engine and provided more informative search bar options.

Projects – Portfolio: JARVISHHH.github.io

Ray Tracing and Path Tracing | C++

Feb. 2023 - Present

- Implemented traditional ray tracing and path tracing in C++, from reading scene data to outputting an image.
- Implemented basic features, like reflection, refraction, shadows and etc.
- Accelerated intersection calculation with bounding volume hierarchy(BVH) and k-dimensional tree(k-d tree).
- Implemented phong illumination model, implicit and explicit intersection, soft shadows under area lights, depth of field, texture mapping, super sampling and etc., for traditional ray tracing.
- Implemented four basic BRDFs (diffuse, glossy reflection, mirror reflection and refraction), Cook-Torrance microfacet model, importance sampling, stratified sampling and etc., for path tracing.

Escape - A Game Produced by Self-made 3D Game Engine | C++

Feb. 2023 - May 2023

- Designed and implemented a 3D game engine, and produced a dungeon escape rogue-like game using the engine.
- Implemented cylinder collision for dynamic game objects and Ellipsoid-Triangle collision for static game objects. Optimized collision checking with bounding volume hierarchy(BVH) and Hierarchical Grid.
- Implemented the A* algorithm based on the navmesh for pathfinding and behavior trees for decision-making.
- Implemented several graphics features based on OpenGL, like particle system, bump mapping, shadows, etc.
- Integrated a basic UI toolkit, including buttons, text and images.

Stylized Caustics: Progressive Rendering of Animated Caustics | C++

Apr. 2023 - May 2023

- Implemented the techniques introduced in the paper *Stylized Caustics: Progressive Rendering of Animated Caustics*.
- Designed and implemented the workflow framework, designed parameters for new input data and integrated new parameters into the existing scene file format for better readability and scalability.
- Projected generated photons from 3D space to 2D plane, and projected moved photons back to 3D space.
- Implemented greedy algorithm to assign photons to achieve minimal moving energy cost.

Technical Skills

Languages: C/C++, Java, Golang, Scala, Python, Shell, SQL

Developer Tools: VS Code, Visual Studio, Goland, IntelliJ, Anaconda, Virtual Box, Vim, Grafana, Qt Creator

Technologies/Frameworks: Git, Unreal Engine, OpenGL, Eigen, GLM, Linux, Elasticsearch, JavaFX, Flask, SQLite