Yao Lin

Personal Web: matudinglin.github.io

Email : matudinglin@gmail.com

Mobile : +1-213-551-4786

Education

- **University of Southern California** — California, USA
  Master of Science in Computer Science; GPA: 3.92/4.0 — *Aug. 2022 – May. 2024(Expected)*
- **Dalian University of Technology** — Dalian, China
  Bachelor of Engineering in Computer Science and Technology; GPA: 86.6/100 — *Sept. 2018 – June. 2022*

Skills

- **Programming Languages:** C++, Python, C#, Java, SQL
- **Tools and Frameworks:** OpenGL, Qt, VTK, Unity Engine, OpenXR, XR Interaction Toolkit, Keras, Git, Linux, CUDA, HTML
- **Expertise Areas:** VR/AR, Software Development, Computer Graphics, Visualization, Game Development

Experience

- **Qualcomm** — California, USA
  Multimedia Engineering Intern at XR Research Group — *May. 2023 – Aug. 2023*
  - Deisgned a XR application that provides an **immersive visualization** of XR perception algorithms' internal workings.
  - Implemented **10+ features**, accelerating debugging and improvement of 6DoF tracking and 3D reconstruction algorithms.
  - Developed the software using **VTK**, **Qt**, OpenXR and Python. Supported loading and parsing 6+ different formats of dataset and enabled real-time customized data rendering in various view modes.
- **University of Waterloo** — Ontario, Canada
  Research Intern at CViSS Lab — *Jun. 2021 – Sept. 2021*
  - Designed a **VR point cloud viewer** using C# on Unity Engine and deployed it on **Oculus Quest 2**.
  - Designed and developed **8+** features and **5+** VR interaction tools for structure assessment, allowing users to load different scenes, explore and collaborate in 3D virtual environment.
- **Tsinghua University** — Beijing, China
  Research Assistant Intern at The Future Lab — *Sept. 2021 – Dec. 2021*
  - Built an real-time non-destructive fruit quality detection system by analyzing spectrum data collected from spectrometer, which has been adopted by industry with **92%+ accuracy** for passion fruit classification.
  - Wrote a system software for chemometrics and serial communication in **C++** and C#, allowing to show result in real-time.

Research & Development

- **VR Exploration Tool for Visually Impaired People** — Wisconsin, USA
  Research Assistant at University of Wisconsin-Madison — *Jun. 2022 – Sept. 2022*
  - Developed a **Mobile VR App** which enabled visually impaired people to explore and navigate in virtual environment.
  - Enhanced **user experience**, providing three view modes, gesture navigation, edge outlining and object interaction.
  - Implemented the application using C# on Unity Engine and deployed it on **IOS** and **Android** devices.
- **Automatic Generation of Indoor-scene Image Segmentation Datasets** — Dalian, China
  Research Assistant at Dalian University of Technology — *Dec. 2018 – Nov. 2019*
  - Wrote an indoor-scene image datasets generator for object segmentation model training in **C++** and **OpenGL**, including synthesized images, annotated images and reference images.
  - Enhanced the photorealism of training images by developing the generator on **Unreal Engine 4**.
  - Validated the datasets using **Python** and **Keras**, leading to a **13%** improvement in the accuracy of object segmentation model.

Projects

- **Computer Animation and Simulation Projects:** Developed three projects involving computer animation algorithms, including jelly cube simulation using mass-spring system, motion capture interpolation, and inverse kinematics with skinning. (Feb. 2023)
- **2D Platformer Game with Innovative Drawing Mechanics:** Led a 7-member team through the full-cycle design, development, and iteration process to create a fully-fledged game using Unity Engine from scratch. (Feb. 2023)
- **Computer Graphics Projects:** Developed projects related to computer graphics, covering shading pipeline, geometric transformation, ray tracing, path tracing, spline generation, texture mapping and physics simulation. (Aug. 2022)
- **C-like Language Compiler:** Designed and implemented a C-like language compiler using C++ with a Qt user interface, including lexical analysis, syntax analysis, error handling, semantic analysis, interpretation and execution features. (June. 2021)