Zhongqian Duan

duanzqhenry@gmail.com | zlzq-duanzq.github.io | Github | Linkedin | 734-546-0202

Education

University of Michigan - Ann Arbor, MI Aug. 2022 – Apr. 2024
- *M.S.E. in **Computer Science and Engineering*** | **GPA** 4.0 / 4.0
- Coursework: Computer Networks, Web Systems, Parallel Computing, Natural Language Processing

University of Michigan - Ann Arbor, MI Aug. 2020 – May. 2022
- *B.S.E. in **Computer Science*** | **GPA** 3.9 / 4.0 | **Honors**: Dean's List, University Honors, James B. Angell Scholar
- Coursework: Operating Systems, Database Management Systems, Machine Learning, Computer Vision, Deep Learning for CV, Computer Game Development, Computer Security, Parallel Programming with GPUs

Shanghai Jiao Tong University - Shanghai, China Sep. 2018 – Aug. 2022
- *B.S.E. in Electrical and Computer Engineering* | **GPA** 3.7 / 4.0 | **Honors**: Undergrad Excellent Scholarship

Skills

Programming Languages: C/C++, C#, Python, Java, JavaScript, Matlab, HTML, CSS, React.js, Node.js, SQL
Tools and Frameworks: Git, LaTeX, CUDA, Pytorch, OpenCV, Scikit-learn, Mathematica, Unity, Linux, Docker, AWS

Internship Experience

Rec Room Jan. 2023 – Apr. 2023
Software Engineer Intern Seattle, WA
- Embedded on the **tool development** team, where works across the client, server, editor and website.

FantasyAR SJTU May. 2022 – Aug. 2022
Software Development Engineer Intern Shanghai, China
- Collaborated with a team of 4 developers to publish a full stack AR fighting game using **Unity** FantasyAR.
- Integrated the Natural Language Processing model **Recognissimo** to implement the voice-control and **GoMap** for location-based AR minimap.
- Developed a back-end server with **Node.js** and a database with **MySQL** to store and update in-game data.

NIO - Autonomous Driving Department May. 2021 – Aug. 2021
Machine Learning Engineer Intern Shanghai, China
- Optimized a **3D Object Tracking Network** for autonomous vehicles, improved the precision by 4%.
- Proposed a lightweight CNN in **PyTorch** to predict lens distortion parameters for removal.
- Utilized pre-trained vision models, such as **MaskRCNN**, with different backbones to detect vehicles and lane lines, and tested for autonomous driving systems on over 2000 on-screen videos.

Software Engineering Projects

Asylum 7: 3D Horror Game Asylum 7
- Led a team of 5 developers to build a horror, role-playing, escape game using **Unity** and **C#**.
- Planned and executed the project roadmap on **Jira**, and managed the development repo on **GitHub**.
- Implemented the core features with **C#** and **multi-thread**, including task management, enemy AI and navigation, detection of darkness, controls of trap and game story progression.
- Iterated three versions of game mechanics and design (alpha, beta, gold) based on 50 hours of playtests with over 200 players. Participated in UM+EMU Games Showcase (ranked 3rd).

Full Stack Website: Online Story Cards MERN project
- Built a full stack website with responsive home and search function with **MongoDB, Node.js, and React**
- Implemented login system to allow **CRUD** operations (post, delete, like) and deployed on **Heroku** and **Netlify**.

Parallel programming and optimaztion with CUDA CUDA project
- Implemented four different methods to accelerate the 4D convolutional layers in the LeNet-5 with **CUDA**.
- Tested different combinations of TILE_WIDTH and BLOCK_SIZE on each method, and the final result improved the time efficiency by approximately 2 times.

Operating System Simulation OS project
- Utilized **C++** to implement a thread library (thread, cv, mutex), and a network file server.
- Designed a **virtual memory manager** which managed various application address space.

Database Management System Simulation DB project
- Designed SQL database utilizing relational model to store information for fictional social media platform Fakebook, and generated external views to visualize data, laying foundation for centralizing data administration.
- Summarized 6 join algorithms and implemented Grace Hash Join during partition and probe phases in **C++**.