Hanlin SUN

hanlinsu@seas.upenn.edu || +1 4849089054 ||**Target: Graphics Development**

EDUCATION

University of Pennsylvania **Sept. 2021 – Jul. 2023**

Master of Computer Graphics and Game Technology ,GPA 3.90/4.0

Main Course: Interactive Computer Graphics, GPU Programming, Advanced Rendering(**PBR**)

Shanghai University **Sept. 2016 – Jul. 2020**

Bachelor of Digital Media Technology, GPA:3.70/4.0 (Ranked 2/119)

Main Course: Data Structure, Operating Systems, DataBase(SQL), Computer Network

INTERNSHIPS

OPPO, U.S. Research Center **Jun.2022 -- Aug.2022**

AR Graphic EngineeringIntern

- Implemented **Real Time SSAO (Based on Java, libGDX and Android OpenGL ES 2.0**).
- Use Depth map to reconstruct view space position in SSAO and **improve it's performance by 50%.**
- Read several papers and implemented **Spherical Harmonic Light global illumination** instead of using **Monte Carlo** to sample diffuse map(based on OpenGL ES 2.0), **increase rendering speed by 66%**.
- Optimize **SSAO algorithm** to make it running under **60FPS** on **AR Camera.**
- Combine **Spherical Harmonic Light illumination** with **PBR** Shader to optimize it's rendering speed and lighting effect.
- Using **GPU Skinning Mesh** technique to improve joint support by **1000%**

SELECTED PROJECTS

OpenGL PBR Based Shading Model(C++,OpenGL 3.3) **Mar.2022--- Apr.2022**

- Use **Deferred Rendering Pipeline** to support multi light source and optimize rendering speed
- Developed **Cook-Torrance PBR shading model**
- Implemented **image based lighting(IBL)** technique(Diffuse map and Specular map).
- Use **Mipmap sampling** to do the anti-aliasing process

OpenGL GPU Path Tracing Engine(C++,OpenGL 3.3) **Jan.2022 --- Mar.2022**

- Implemented **Lambert BRDF, Specular BRDF, Specular BTDF** calculation
- Implemented **direct light sampling/ Fresnel** reflection calculate function
- Implemented **KD-Tree** accelerating structure to optimize rendering speed.
- Implemented **direct light illumination** and **indirect light illumination**.
- Implemented **Multiple Importance Sampling / full lighting** rendering Pipeline.

OpenGL Mini-MineCraft(C++, OpenGL 3.3) **Oct.2021--- Nov.2021**

- Implemented **Shadow map** technique.
- Use **Temperature-Moisture** model to generate noise and 4 types of biome
- Procedually generating grass and Trees use **Blue Noise**

SKILLS

Coding: C++ ,Java, C# ,GLSL,HLSL,Vulkan,OpenGL, CUDA
Software: Unity, Unreal 4, Qt, Maya, Houdini
Knowledge: Data Structure, Algorithm, Computer Graphic, GPU, 3D Modeling