3149 Casa De Campo, Apt 103
San Mateo CA 94403

HAO WU

(215) 588-7240
wuhao1117@gmail.com
www.wuhao.co

EMPLOYMENT

Software Engineer **Sony Computer Entertainment America** **June 2014 – March 2015**

4k gaming research (Summer 2014)

- Implemented SMAA T2X mode in Unity and integrated into 5 demos for nice and fast anti-aliasing
- Refactored camera system and designed multi-res adjustable split-screen mode for easy 2K/4K comparison

Virtual reality game prototyping with Morpheus (Fall 2014 – Spring 2015)

- Researched and implemented multiple navigation controls in Morpheus, effectively reducing motion nausea
- Developed two VR puzzle game prototypes in Unity; wrote shaders and developed gameplay features

EDUCATION

Philadelphia, PA **University of Pennsylvania** **Fall 2012 – May 2014**

- M.S.E. in Computer Graphics and Game Technology. GPA: 3.83
- Graduate Coursework: Computer Graphics; GPU Programming and Architecture; Computer Animation; Game Design and Development; Physically Based Animation; 3D Computer Modeling.

Beijing, China **Beijing Institute of Technology** **Fall 2008 – June 2012**

- B.S.E. in Electrical Engineering. In-major GPA: 3.86.
- Undergraduate Coursework: Data Structure; C++ Programming Language; Computer Architecture; Operating Systems; Advanced Algebra; Mathematical Analysis; Information Theory.

TECHNICAL EXPERIENCE

- **Real-Time Path Tracer on GPU (C++, CUDA, OpenGL)**
  - Supports diffuse/glossy/Fresnel-transparent surface, OBJ file loading, motion blur and depth of field
  - Separated direct illumination and indirect illumination for fast convergence and interactive camera
- **WebGL Ocean Simulation using Fast Fourier Transform (HTML, JavaScript, WebGL) | team of two**
  - Wrote GPU Fast Fourier Transform library in JavaScript for use in the WebGL simulation
  - Wrote simulation code using the FFT library and provided interface for rendering
  - Collaborated in infinite ocean tiling and physically based rendering of sun, sky and ocean
- **CUDA software rasterizer (C++, CUDA, OpenGL)**
  - Complete rendering pipeline with vertex shader, primitive assembly, rasterization, and fragment shader
  - Can achieve 120-140fps for 100k triangles. Implemented face culling, SSAA and early depth test
- **Deferred renderer (C++, OpenGL)**
  - Deferred rendering pipeline with G-buffer rendering
  - Implemented postprocessing effects including blooming, toon shading with edge detection and SSAO
- **SimBubble – Bubble simulator plugin integrated into Maya fluid solver (C++, MEL, OpenGL) | team of two**
  - Designed as standalone OpenGL executable; compiled into DLL after alpha version
  - Utilized Delaunay triangulation in CGAL to capture the topology of input bubble particles
  - Integrated the simulator with Maya Fluid and designed plugin interface
- **Tornadogeddon – Unity3D game: devastating levels by controlling a tornado (C#) | team of three**
  - Modeled tornado physics from scratch, animated tornado with Shuriken particle system
  - Implemented destructible environment, designed camera, avatar controller and other gameplay features

Languages and Technologies

- C++, C #, OpenGL, CUDA, JavaScript, HTML, Python, Maya Embedded Language
- Unity3D, Visual Studio, Maya SDK, Eclipse, MATLAB, Motion Builder, Houdini SDK, Photoshop, Premiere