

# Tsingtao Zhang

[tsingzhang1@gmail.com](mailto:tsingzhang1@gmail.com), Rochester, NY (willing to relocate), 585-615-0628  
[LinkedIn](#), [Portfolio](#), [Github](#)

## Skills

**Game Engine and Graphics API:** Unreal 5, Unity, OpenGL, Direct 3D 11

**Programming Language:** C/C++, C#, HLSL, GLSL, Python

**Software:** RenderDoc, Blender, Houdini, Photoshop, Substance Painter, Git, Perforce, Figma

**Area of Focus:** Asset Pipeline, Tool Development, Asset Creation, Procedural Content Generation, VR, Multiplayer

## Work Experience

**Role: Contract Technical Artist**

July - Oct 2025

**The Forge Interactive Inc.** Using **Blender**, **Substance Painter**, **PBR workflow**, **HLSL**, **Python**

- Optimized San Miguel scene assets for mobile platform demo by refining **mesh topology**, **UV unwrapping**, and **textures** through **custom PBR pipeline**, using **Blender**, **Materialize**, **Photoshop**, and **Substance Painter** to create high-poly details.
- Collaborated with software engineers to design the **asset creation workflow** in a proprietary asset pipeline.
- Using **Python**, automated file management, texture processing, and validation in The Forge's asset pipeline.
- Collaborated with graphics programmers to provide technical requirements for particle systems, UI implementation, and save/load functionality, while delivering iterative feedback on feature development.
- Developed **particle and lighting effects** using The proprietary particle editor for mobile platform, while resolving shading issues in legacy codebase using **HLSL**.

**Role: Graduate Research Assistant**

June - Aug 2024

**Rochester Institute of Technology** Using **VR**, **Unity**, **URP**, **Meta Quest**, **Android**

- Enhanced a VR exercise game by switching the render pipeline to **URP** and optimizing gameplay performance.
- Optimized rendering and game logic, deployed a **standalone** app that can run directly in the **Meta Quest** headset, without the need of PC Link.
- Designed and developed an physics-based aiming and block-avoiding **AI shooter** with **character animations**, enabling other researchers to guide users' limb positioning by easily setting the shooting position.
- Parameterized in-game variables that are tuned during gameplay based on **research data collection** requirements.
- Configured a wireless environment for seamless **data and video streaming** of headset imagery using Socket, reducing data collecting time from 5 min to 10 seconds per user test.

## Projects

**Role: Technical Artist and Gameplay Programmer**

Sep 2024 - June 2025

**Duolatera:** A VR Multiplayer Puzzle Game, using **VR**, **Unreal 5**, **C/C++**, **Niagara**, **Python**, **Blender**, **Perforce**, **HLSL**

- Implemented **procedural content generation assets** using **Unreal PCG Graph**, and spline's **auto-snapping tool** using **C++** and **Blueprint**, efficiently utilized 3D assets, reduced level layout time by 90%.
- With 3D asset creating skills, established a **custom asset production pipeline** in an art bible, led an external art team of 5, **trained** 3 external 3D artists with custom pipeline and workflow.
- Using **Python** and **Unreal Editor Utilities Widget**, designed and implemented an **automated asset import tool**, reduced 90% of related manual work.
- Developed **Python** and **PyQt-based tools**, including a texture converter, asset renamer, and Blender validation plugin, automating asset workflows and eliminating key pipeline bottlenecks.
- Created procedural and stylized **material and VFX** using **Material Graph**, **HLSL**, and **Niagara system**.
- **Rigged character avatars**, and built IK retargeted/predicted **avatar animation** based on player's movement.
- Implemented the **online multiplayer gaming** feature, allowing 2 players to cooperate remotely through Steam.

**Role: Technical Artist**

July 2025 - now

**A Runtime Procedural Tower Defense Game Prototype**, using **Unity**, **HLSL**, **RenderDoc**, **Unreal 5**, **Houdini**

- Developed a **real-time procedural wall generator** in Unity using **compute shaders** and Unity's low level **graphics API**, achieving **10K+ instances per frame at 50+ FPS** on RTX 4060 (profiled using **RenderDoc**) by offloading spline-based instance transform calculations from CPU to GPU.
- Implemented **Centripetal Catmull-Rom spline** evaluation and a **distance lookup table in HLSL** for efficient instance positioning and randomized segment lengths.
- Created two similar procedural fences using **Houdini** and **Unreal PCG Graph** for comparison. Conducted cross-engine performance analysis, identifying limitations in runtime PCG updates through codebase investigation.

## Education

**Rochester Institute of Technology**, Rochester, NY.

Aug 2023 - Aug 2025

M.S., Game Design and Development

GPA: 3.89

**China Agricultural University**, Beijing, China.

Sep 2018 - June 2022

B.Eng., Agricultural Structure Environment Engineering

GPA: 3.44