

# AMAN SACHAN

amansachan.com

github.com/Aman-Sachan-asach

asach@seas.upenn.edu

(267)-361-8276

## SKILLS

### PROGRAMMING

- C/C++
- GLSL
- MEL
- Javascript
- C#
- HTML/CSS
- Java

### GRAPHICS

- CUDA
- OpenGL, WebGL
- Maya API
- Threejs

### SOFTWARE

- Git
- Unity
- Maya
- Qt
- Visual Studio

## COURSEWORK

- GPU Programming (Fall '17)
- Advanced Computer Graphics
- Procedural Graphics
- Game Design (Fall '17)
- Computer Animation
- Data Structures and Algorithms

## LEADERSHIP & AWARDS

### HELIOS - 2016

- ◆ Project Lead; Received Rs. 1,20,000/- in funding
- ◆ Finalist of KPIT Sparkle & Engineer Infinite

### EARTHIAN - 2014

- ◆ Team Lead; Awarded Rs. 1,50,000/-

### VIDYUT 2k14

- ◆ Prime Coordinator; Head of Sponsorship; Public Speaking

## EDUCATION

- UNIVERSITY OF PENNSYLVANIA**, Pennsylvania, USA May, 2018  
M.S.E. COMPUTER GRAPHICS AND GAME TECHNOLOGY  
GPA: 3.45/4.0
- VISVESVARAYA TECHNOLOGICAL UNIVERSITY**, Bangalore, India July, 2016  
B.E. ELECTRICAL AND ELECTRONICS ENGINEERING

## EXPERIENCE

- SIG CENTER FOR COMPUTER GRAPHICS** May - Aug, 2017
- RESEARCH ASSISTANT ◆ C#, Unity ◆ Oculus DK2, SMI Eye Tracker
- SUBLIMINALLY DIRECTING GAZE IN VR under DR. STEPHEN LANE
- Implemented a real time **CMA-ES algorithm** (a machine learning algorithm)
  - Developed a **game in Virtual Reality** that used visual stimuli to subliminally ( without conscious perception ) direct user attention
  - Supervised and taught an undergraduate intern intern working on the project

## PROJECTS (See more projects at amansachan.com)

- GPU BOID FLOCKING** ◆ C++, CUDA Sept, 2017
- Implemented Craig Reynold's crowd simulation algorithm to model flocking behavior
  - Visualize **1.6 million particles** running at **60 FPS** on a notebook GTX 1070
- CPU MONTE CARLO PATH TRACER** ◆ C++, OpenGL Feb - May, 2017
- Supports **Volumetric Rendering, Multiple Importance Sampling, BVH Acceleration (9800% speed up), Multi-Threading**
  - Handled materials with **Micro-facet** surfaces and **Fresnel reflectance** models;
  - **Realistic** modeling of **light** sources and Thin Lens **camera models**;
- INTERESTING LEVEL GENERATOR** ◆ Javascript, WebGL, GLSL, Threejs April, 2017
- A procedural **multi-layer dungeon generator** that generates levels based on a voronoi-like graph after it has been heavily modified by various filters to create interesting level layouts
  - Implemented: a **Realistic Fog** shader; Biome and Elevation dependent **Terrain** on the GPU
  - Implemented a controllable **Crumbling Pathway** aesthetic
- GRAVITY WELLS** ◆ C#, Unity Sept, 2017
- A puzzle game that involves slingshotting your ship using black holes
  - Designed and Implemented the gameplay and physics
- IMPLICIT SURFACES** ◆ Javascript, WebGL, GLSL, Threejs Feb, 2017
- Generated **metaballs** in **real time** using the **marching cubes** algorithm
  - **~1700 triangles** dynamic triangles running at **60 FPS** on a GTX 1070
- ART OF COLLISIONS** ◆ Group Project ◆ C++, MEL, Maya API March - April, 2017
- Implemented a particle based rigid-body simulator based on the paper, "**Unified particle physics for real-time applications**", by Macklin, Muller, Chentanez, and Kim
  - Jointly implemented **Shape Matching Constraints** and **Position Based Dynamics**
  - Implemented the conversion of arbitrary meshes into particle groups
- MESH EDITOR** ◆ C++, OpenGL Nov, 2016
- Implemented an interactive **Half-Edge Mesh data structure, Catmull-Clark subdivision, Interactive Skeleton Structure, Skinning**, and Shader Based **Skin Deformation**