

Yugank Mishra

findyugank@gmail.com | 682-666-8076 | [GitHub](#)

Education

University of Texas at Dallas, Bachelor of Science in Computer Science *August 2025 - Present*

- Member of **Computing Scholars** (Computer Science Honors Program)
- Coursework: Discrete Mathematics, C++ Programming, Mathematics (up to Linear Algebra)

Coppell High School *August 2021 - May 2025*
GPA: 5.46/6.0 *SAT: 1560 (760 R&W, 800 Math)*

Work Experience

UT Dallas Clark Summer Research Program *May 2025 - Aug 2025*

- Worked under Dr. Wei Yang's Lab for 10 weeks as part of a summer research program.
- Looked into efficiency adversarial attacks on Deep Automatic Speech Recognition Pipelines
- Worked over 200 hours as a researcher and presented at a undergraduate research event.
- **Won an award for work done over the summer.**

Honors and Awards

- Summer Platform for Undergraduate Research (SPUR) Poster Dept. of Computer Science Award Winner *August 2025*
- National Merit Scholarship Recipient *March 2025*
- AP Scholar with Distinction (AP Score Average: 4.92) *August 2022 - May 2025*
- CERN BL4S Outreach Award *April 2024*

Technologies

Languages: C++, C, Java, C#, SQL, Python

Technologies: Tensorflow, Keras, Pytorch, LWJGL, OpenGL, Unity Engine

Technical Certifications

- Microsoft Technology Associate (Introduction to Programming With Java) *May 2022*
- Information Technology Associate (Certiport) *May 2022*

Projects

Yope3D | [GitHub Repository](#) | Java, LWJGL *October 2023 - Present*

- Developed a **custom 3D game engine** from scratch, supporting 3D models, **physics**, audio, and a flexible scripting interface for game development.
- Enabled easy loading and manipulation of 3D models from software like Blender, with integrated support for texturing, transformation, and physics, all via scripting.
- Designed a **spatial audio system** with support for directional effects (pitch shift, attenuation, etc.)

and spatial audio positioning, enhancing game realism with minimal code.

- Built an advanced **UI system** allowing for **custom fonts**, **component design**, and easy integration with **2D/3D** game graphics for **overlays**, **menus**, and more.
- Implemented a **physics system** capable of simulating **soft-body** and **rigid-body** interactions, optimized with a clever reorganization of 3D space (**BSP Tree**) for efficient collision detection.
- Included both **rasterization** and **ray-tracing** capabilities for realistic graphics, with support for **shadows**, **recursive reflections**, and other advanced visual effects.
- Created extensive documentation on GitHub, supplemented by YouTube demos ([Youtube Channel](#)) and [Medium Articles](#) explaining key features.

Java AI | [GitHub Repository](#) | Java

February 2025 - May 2025

- A small Java codebase capable of **generating** and **training** a **neural network**, written from **scratch in Java** using no prebuilt libraries.
- Since no machine learning libraries were used, this project allowed me to gain a deep insight into the core mechanisms of machine learning, particularly how models learn under the hood.
- Includes features like a **custom tensor library**, **layer-wise processing**, **Batch Normalization**, and **Adam Optimizers**.
- The codebase is capable of creating and training a model that can accomplish the **MNIST digit identification** task with high (**96%**) accuracy.

Gen AI Digit Drawer | [GitHub Repository](#) | Python, Keras, Tensorflow, Numpy *November 2024*

- Developed a neural network model that generates realistic images of handwritten digits from scratch (**β -VAE**); a stepping stone towards a **diffusion model** for image generation.
- Used Tensorflow & Keras skills to create, train, and generate from this model.
- The model was trained & generated images on the Google Colab service, allowing me to learn how to build & operate models like these in new environments.

3D Gravity Simulation | [GitHub Repository](#) | Java, Java2D

November 2022

- Developed a **3D** gravity simulation using Java and Java2D to model gravitational systems based on Newton's laws of motion.
- The simulation can handle **complex gravitational interactions**, allowing for both **chaotic** and **stable** system behaviors to be observed.
- Built a custom math library to support the project, deepening my understanding of how gravitational equations work from a computational standpoint.
- Users can input various properties for celestial bodies (mass, velocity, etc.) and simulate the system, exploring both **chaotic** and **periodic** behaviors in real time.

Java Polynomial Solver | [GitHub Repository](#) | Java

February 2023

- Developed a simple Java program capable of solving user-defined equations using numerical methods.
- Implemented a computational version of **Newton's method** for finding the roots of polynomials, including a custom interpretation that allows the program to understand and **differentiate polynomial expressions**.
- This project was my first major foray into applied mathematics, where I learned how mathematical concepts like infinity & derivatives are approximated in the world of computing.