Shreyas Kanjalkar

https://skanjalkar.github.io/ Mobile : +1-774-701-8250
Github: https://github.com/skanjalkar Email : skanjalkar3@gatech.edu

Education

- **Georgia Institute of Technology** Atlanta, GA
  Master of Science, Computer Science *Aug 23 - May 25*
  - Specialization: Distributed Systems.
  - Courses: Software Engineering, Distributed Computing, Algorithms, Computer Networks
- **Worcester Polytechnic Institute** Worcester, MA
  Master of Science, Robotics; GPA: 4.00/4.00 *Aug 21 - May 23*
  - Research Assistant at PeAR (Vision) Lab, WPI, supervised by Prof. Nitin Sanket
  - Courses: Computer Vision, Robot Dynamics and Control, DBMS, Data Structures and Algorithms
- **Manipal Institute of Technology** Manipal, India
  Bachelor of Technology, Mechanical Engineer; GPA: 8.26/10.00 *Aug 16 - May 20*
  - Undergraduate Thesis on Blood Flow in CFD, published in **JESTEC** 21.

Technical Skills

- **Languages**: C++, C, Python, GO, SQL, JavaScript, nodejs, Flask, React, MATLAB
- **Tools/Libraries**: AWS, Linux, Git, GitHub Actions CI, Docker, Pytorch, numpy, opencv, CAD

Industry Experience

- **Software Firmware Intern — RoboMatter Inc**: *Jan 23-May 23*
  - Prototyped object color detection algorithms in **Python 3** for rapid iteration and testing. Adapted and implemented the algorithm in **embedded C** to run on a memory-constrained device - ESP32S3
  - Designed and simulated a friction-inclusive, power-loss model of a 3-wheel AIM robot using **MATLAB**
- **Robot and Automation Design Engineer Intern — Wipro PARI**: *June 22-Aug 22*
  - Collaborated with design, manufacturing and electronics teams with over **50** people to design Pallet used in Tower Parking System

Academic Projects

- **Failure Detection in Distributed Systems - SWIM Protocol Implementation**: *Summer '23*
  - Implemented SWIM in **C++** protocol for scalable membership management in distributed systems using gossip-based communication and periodic updates
  - Utilized combination of direct and indirect pings for efficient failure detection, ensuring reliable and up-to-date view of active members while handling network delays and false positives github
- **Full Stack - Face Swap Technology Development in Digital Media**: *Fall '22*
  - Implemented Face Swap with Delaunay Triangulation in Python 3 using openCV
  - Deployed Face Swap service on AWS EC2 machines using a two-machine microservice architecture. One machine hosted the web service, and the other handled face swap logic
  - Accelerated software development and delivery through automated **unit testing** with pytest, GitHub Actions **CI**, and git version control pdf
- **Aerial Dance of Quadrotor Swarm - Dynamic Choreography**: *Spring '23*
  - Implemented low-latency UDP based **socket programming** to synchronize **4** drones with human dancing motion
  - Implemented collision-free drone movement algorithm in **Python 3** using conflict-based search methods
  - Utilized multi-threading and computer vision on a surface computer to configure drones within **10ms** latency
- **Data analysis on Lichess database**: *Fall '22*
  - Created a Relational Database application using **sqlite3** and SQLalchemy with **Flask**
  - Contributed to Berserk, an open-source library used to interact with Lichess API.
  - Created a front-end UI using **React** to allow users to interact with the application by using **REST** API github