Manas Kale

B-1701, Dream Heights, Sector 19, Plot 28, Kharghar - 410210, Navi Mumbai, Maharashtra, India

☐ (+91) 7738676567 | ✉ manaskale@hotmail.com | ⌨ manas96 | in ManasKale | manas96.github.io

Education

Bachelor of Engineering - Maharashtra Institute of Technology (Pune University) *Pune, India*
COMPUTER SCIENCE AND ENGINEERING *2014 - May. 2018*
- First class with distinction: 74.47% (final year)

12th - Ryan International School, Kharghar *Navi Mumbai, India*
ISC BOARD *March 2014 exam*
-Aggregate: 75% (Computer Science: 96/100)

10th - Ryan International School, Kharghar *Navi Mumbai, India*
ICSC BOARD *March 2012 exam*
-Aggregate: 83.71% (Computer Applications: 98/100)

Work experience & Internship

Data Scientist - Infinite Uptime *Aundh, Pune*
PYTHON (KAFKA STREAMS, APACHE BEAM, SCIKIT-LEARN, HOLOVIEWS), C++ (EMBEDDED FFT LIBRARIES), FOURIER ANALYSIS, STREAMING *February 2019 - Present*
ANALYTICS, BASH, INTERNET OF THINGS (MQTT PROTOCOL), GIT

- Working on whole platform - from embedded edge computation device (**C++**) to cloud platform (**Python**).
- To handle continuous real time vibration data generated by edge device, I am currently working on setting up an infrastructure agnostic streaming analytics pipeline (using **Kafka/Apache beam**) to replace older batch processing pipeline.
- Created a framework to visualize data flowing between transformations in a stream processing pipeline (**HoloViews, Apache Beam**).
- Helped implement dynamic thresholding functionality by analysing historic data and setting thresholds appropriately (**Python**). This reduced manual intervention required by support personnel.
- Helped develop a prototype machine fault classifier. Based on edge device's vibration data, our **K Nearest Neighbours** classifier (**scikit-learn**) was able to detect different types of faults in an industrial motor.
- Fixed critical stability issues for embedded device's WiFi (**ESP8266 chip, C++, MQTT protocol**) firmware.
- Involved in effort to thoroughly understand previously undocumented C++ codebase. Wrote **extensive documentation** for further reference.
- Helped increase edge device's FFT (**Fast Fourier Transform**) block size from **512 to 4096**, improving frequency resolution **from 6Hz to 0.2Hz** while working under tight memory constraints.
- Implemented custom **binary data format** for transmitting edge device's FFT data(**C++**), reducing size of typical data packet by **3x**. Wrote server side decoding logic in Python.
- Contributed to writing an automated firmware flashing script (**bash**), improving manufacturing workflow.
- Imporved embedded dev workflow by introducing new tools. Customized VSCode IDE for our C++ dev environment, formalized version control (**git**) and software release strategy.

Associate Software Engineer - NICE Interactive Solutions *Hinjewadi, Pune*
JAVA (SPRING FRAMEWORK), JAVASCRIPT (ANGULARJS), AMAZON AWS, GIT, AGILE *July 2018 - December 2018*

- Part of team responsible for **Tenant Management** microservice - a service which handles creation, maintenance, billing and license/subscription tracking of tenants (third party vendors) on NICE's CXOne cloud platform.
- My tasks included developing new features and implementing business logic using both **Spring** framework and **AngularJS** in an agile project management environment.

Intern - Tata Consultancy Services *Hinjewadi, Pune*
IBM MAXIMO, PYTHON, WEBSOCKETS, REST API, JAVASCRIPT, HTML BOOTSTRAP *2 June - 28 July 2017*

- Made a full stack webapp to monitor asset data in real time, detect anomalous data and issue warnings.
- The web server (**Python2**) pulled data from IBM Maximo's REST API which was broadcast through **websockets**.
- The webpage UI (**HTML bootstrap**) tracked each asset on a map and provided real time graphs for each sensor, issuing alerts in case of anomalous data.
- **Project report :** manas96.github.io/internship_report/

Projects

Raytracer
C++ (GLM) *May. 2019 - Present*

- **Source code & screenshot(s) :** github.com/manas96/Raytracer
- A backwards raytracer written for learning purposes using **Peter Shirley's Raytracing** series as reference.
- Implemented **diffuse, dielectric** and **metallic** materials.
- Using GLM (**OpenGL Mathematics**) library for vector arithmetic.
- Currently working on optimization using parallel processing and acceleration structures.

Satellite tracking ground station for SatNOGS network
RASPBERRYPI, SOFTWARE-DEFINED RADIO *March. 2019*

- SatNOGS is a crowd-sourced satellite data collection network with stations all over the globe. To track and collect data from **MOVE-II cubesat** (move2space.de/MOVE-II/), I helped build (assembled radio antenna, installed SatNOGS software on Raspberry) station **mumbai-gs** (network.satnogs.org/stations/440/), the only one in India. I am currently the sole maintainer of this station.

3D Game Engine using OpenGL

OpenGL, Java (LWJGL), GLSL Shaders   *June. 2017 - September 2018*

- **Source code and screenshot(s):** github.com/manas96/3D-gameEngine-v2
- An *interactive* 3D rendering engine using *OpenGL* API (through LightWeight Java Game Library).
- Implemented features include: • *Lighting (ambient, point)* • *OBJ geometry file loader* • *Fog blending* • *Entity system architecture* • *Collision detection* • *Particle system* • *Skybox* • *Raycasting* • *Normal mapped textures* • *Fresnel reflection shader*

Human Emotion Detection using Multimodal input (BE Project)

Python (Flask, Scikit-learn, OpenCV), Machine Learning (SVM, multinomial NB, DNN), JavaScript (Bootstrap, HighCharts.js)   *June. 2017 - June 2018*

- We implemented a novel approach to determine user's emotion using a *weighted sum* of the following inputs: *facial features*, *spoken text* and *voice characteristics(tone)*.
- Used *IEMOCAP* dataset to train a *DNN* (Deep Neural Network) for tone module, *Cohn-Kanade* dataset for training a *SVM* (Support Vector Machine) for facial module and *IEMOCAP* to train a *multinomial NB* (Naive Bayes) classifier for spoken text module.
- Each module's *confidence score* along with it's *weight* was used to calculate final emotion. Weights were adjusted dynamically based on quality of input and confidence score.
- Our algorithm was able to perform better using dynamically adjusted weights when compared to individual modules.
- **Detailed project report :** manas96.github.io/project_thesis.pdf

Open source contributions to MovingBlocks organization

Java (libGDX), Gradle, Git   *March. 2018*

- Fixed bugs for Terasology(a minecraft inspired voxel engine) and DestinationSol(a 2D space shooter).
- **Fixed a UI issue for Terasology** : github.com/MovingBlocks/Terasology/pull/3275 • **Fixed a bug in DestinationSol where enemies spawned across game reloads** : github.com/MovingBlocks/DestinationSol/pull/252

Research

Driver profiling using realistic racing games

C++ (SDL,OGRE3D), Python (scikit-learn, matplotlib), Machine learning (SVM,KNN,NB)   *March 2017 - April 2018*

- Attempted to identified different (video game) drivers by logging keypress events and training various classifiers on this data.
- Edited source code of an open source 3D racing game, **StuntRally** (*C++*) to log keypresses. Cleaned, normalized and created a custom feature vector from this log data(*scikit-learn*). Trained *KNN*, *SVM* and *NB* classifiers on this data and found *SVM* performed best.
- M. Kale and M. V. Bedekar, "Driver Profiling Using Realistic Racing Games", 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, 2018, pp. 13-17. doi: 10.1109/ICICCT.2018.8473154
- **Pdf link :** manas96.github.io/driver_profiling.pdf

Achievements

Certificate of Appreciation from Infinite Uptime

For my work on fixing critical edge device issues & creating extensive documentation   *May 2019*

1st Runner Up in SmartIndia Hackathon 2017

C++, OpenCV, NVIDIA CUDA, QT framework   *April 2017*

- Led a team of 5 in a national level non-stop 36 hours hackathon with 351 participating teams. Problem statement: **create real-time video stabilization software for use on Unmanned Aerial Vehicles.**
- We created a working prototype in 36 hours using C++ **OpenCV** for video processing and optimized that with **NVIDIA CUDA** bindings for near real time video stabilization.

Reached final round of IEEE CodeJam 2016

Conducted by MIT IEEE (Institute of Electrical and Electronics Engineers) student chapter   *November 2016*

Reached final round of Battlecode in MITCOE's Tesla 2015

Conducted by MIT College of Engineering students for their annual fest, Tesla   *November 2015*

Extracurricular Activities

Led and organized teams in conducting DOTA2 tournaments

Aarohan festival(2016) - MITCOE & Texphyr festival (2018) - MIT Pune   *March 2018 & January 2016*

Advocated for and organized national level college e-sports tournaments for *DotA2*, a multiplayer online battle arena game I am passionate about.

Conducted hands-on practical session on data recovery, file carving for class students of MIT Pune

Kali Linux, Foremost file carving tool   *August 2016*

Explained file carving process along with demonstration.

Skills & hobbies

Languages: C, C++, Python, Java, GLSL, JavaScript, Bash
Libraries/API's, Protocols: OpenGL, WebGL, OpenCV, GLM, CUDA, Three.js • Matplotlib, HoloViews, Bokeh • Numpy, Scikit-learn • Flask, Redis • TCP, HTTP • **Operating System:** Linux
Software tools: Git, LaTeX, Unreal Engine 4 • Visual Studio, IntelliJ IDEA/Pycharm, Jupyter (Python)
Hobbies: Wildlife & astro photography (manas96.github.io/photography/), trekking, cycling, reading (science fiction/high fantasy), video games