Manas Kale

(+1) 438-680-5129 | manaskale@hotmail.com | manas96 | ManasKale | manas96.github.io

Education

Master of Science - McGill University — *Montreal, Canada*

Computer Science: COMP 558 (Computer Vision), COMP 550 (Natural Language Processing) — *Sept. 2021 - May. 2023 (expected)*

Bachelor of Engineering - Pune University — *Pune, India*

Computer Science and Engineering, First class with distinction — *2014 - May. 2018*

Work experience

TA - COMP 202, Foundations of Programming — *Department of Computer Science, McGill University*

Python — *Fall 2021 (ongoing)*

- Responsibilities include preparing assignments, writing tests, grading, conducting live coding/presentation sessions and holding office hours.
- 512 registered students with 13 TA's.

Data Scientist - Infinite Uptime — *Pune, India*

Python (Apache Kafka, SciPy, scikit-learn, Flask), C++ (Embedded FFT libraries), Java (Apache Flink), Fourier analysis, Streaming analytics, Docker, Git — *February 2019 - February 2021*

- Customer-facing contributions to embedded IoT device (***C++***), analytics algorithms (***Python***) and data processing backend (***Java***).
- Designed, implemented and deployed stream processing pipeline for 5k + IoT devices (***Apache Flink - Java***) to detect alarms, track machine state (using session windows) and reduce infrastructure cost.
- Proactively created framework to collate and explore organization wide sensor data(***Python - HoloViews***), improving data analytics workflow.
- Modeled automated thresholding functionality as a Constraint Satisfaction Problem, implemented and deployed solution ***(Python - SciPy, Flask, Docker)*** saving tedious man-hours per customer.
- Modeled time to alarm prediction as a Linear Regression problem, implemented and deployed solution ***(Python - scikit-learn, Flask, Docker)***, providing customers with an approximate time to machine failure.
- Improved peak-detection algorithm using data-driven heuristics ***(Python - SciPy, Flask, Docker)***, allowing peak detection in noisy signals.
- Implemented extensible ***serialization data format*** for transmitting edge device's FFT data (***C++***), reducing size of typical data packet by ***3x***. Wrote server side deserialization library in ***Python***.
- Improved edge device's FFT (***Fast Fourier Transform***) sampling block size from ***512 to 4096***, improving frequency resolution ***from 6Hz to 0.2Hz*** while working under tight memory constraints. Wrote ***extensive documentation (C++)*** for the previously undocumented codebase.
- **[Patent published : 02/10/2020]** "SYSTEM AND METHOD FOR SEGMENTING TRANSMISSION OF DATA", Application No.202021020386 A

Associate Software Engineer - NICE Interactive Solutions — *Pune, India*

Java (Spring framework), JavaScript (AngularJS), Amazon AWS, Git, Agile — *July 2018 - January 2019*

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

Selected projects

Raytracer

C++ (OpenGL Mathematics, OpenMP, Monte Carlo simulation) — *May. 2019*

- **Source code & screenshots :** github.com/manas96/Raytracer
- ***[Open Source Contribution]*** Contributed to improving all three of ***Peter Shirley's Ray Tracing in One Weekend*** book series: https://raytracing.github.io/books/RayTracingTheRestOfYourLife.html#acknowledgments

Multimodal emotion recognition (Bachelor's Thesis Project)

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

- A novel approach to determine user's emotion using a ***weighted sum*** of inputs: ***facial features***, ***spoken text*** and ***voice characteristics(tone)***.
- Contributed to the facial module, dynamic weight adjustment algorithm, webpage UI and web server.
- **Detailed project report :** manas96.github.io/project_thesis.pdf

Driver profiling using realistic racing games

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

- **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**

Skills

Languages: Python (6/10), Java (6/10), C++ (4/10), GLSL (4/10)
Visualization: HoloViews, Bokeh, Matplotlib
Big data: Apache Flink, Apache Kafka, Redis
Web frameworks: Flask

Graphics: OpenGL, OpenGL Mathematics (GLM)
Data science: Numpy, Scikit-learn, Pandas, SciPy
Tools: Git, LaTeX, Doxygen, Jupyter