

Tejas Naik

Address: 700 Health Sciences Drive, Stony Brook, New York 11790

+1-631-8802768 tjnaik07@gmail.com <https://www.linkedin.com/in/tejasnaik94> <https://www.github.com/TejasNaikk>

OBJECTIVE

Seeking summer 2018 internship opportunities at Google as a Software Engineer to apply my strong problem-solving and analytical skills in real-life challenges.

EDUCATION

Stony Brook University, New York **Aug'17-Dec'18 (Expected)**
Master of Science in Computer Science
Member of the Human Interaction Lab: under the guidance of **Professor Roy Shilkrot**.
Course Highlights: Analysis of Algorithms, Computer Vision, Artificial Intelligence, Machine Learning, Probability and Statistics for Data Scientists, Database Systems.

Pune Institute of Computer Technology (PICT), University of Pune, India **May'13-May'17**
Bachelor of Engineering (Information Technology) **GPA: 3.67/4.0**
Course Highlights: Machine Learning, Operating Systems, Distributed Systems, Data Structures, Design & Analysis of Algorithms, Information Storage & Retrieval, Cloud Computing.

SKILL SET

Programming Languages : C++, Python, C,R.
Tools and Technologies : OpenCV, MATLAB, Tensorflow (Open Source), Keras, Flask, SQL, Hadoop, MongoDB, Cassandra, PHP, HTML, CSS, Ajax, JavaScript, Linux.

WORK EXPERIENCE

BMC Software India Pvt. Ltd. (Software Developer Intern) **Aug'16-May'17**
Auto-Scaling of Application Servers

- Developed a project in a team of four to **Auto-Scale** application servers for BMC's product '**BladeLogic Server Automation(BSA)**'.
- Collaborated micro-services to fetch real time data (CPU Load etc.) from **Nagios Monitoring Tool** and to notify the administrator via email or text message.
- Contributed towards writing the **REST API's** for fetching and storing the data received from the monitoring tool and in generating a machine learning model for analyzing data.
- Integrated this model with the load balancer of '**BladeLogic Server Automation**' to put into the **production environment of BMC Software**.
- Technologies used:** *Amazon Web Services, Machine Learning, Python 2.7, Nagios Monitoring Tool, Flask Interface, Micro-service Architecture, MongoDB.*
- Project Demo Link:** [Auto scaling of Application Servers](#)

PERSONAL PROJECTS

Drowsy Vehicle-Driver Detection. **Nov'17 – Dec'17**

- Engineered a model for detecting the drowsiness level of a driver by continuous eye tracking in a real-time video using **Haar Cascades** and **CamShift** algorithm.
- Retrained the final layer of **Inception-v3** model to the eyes dataset by using **Transfer Learning**. The final **pooling layer** of Convolutional Neural Network for each video frame (2048-D) was used to make a sequence of frames, which was fed as an input to the **Long Short-Term Memory (LSTM) Recurrent Neural Networks (RNN)**.
- Utilized the output of the **RNN** to predict the drowsiness/alertness of the driver and sound an alarm in case of the former with an **accuracy of 87.5%**.
- Technologies used:** *Python 2.7, OpenCV 3.3.0, Tensorflow, CNN, RNN, LSTM.*
- Project Demo Link:** [Drowsy Driver Detection - YouTube](#)

Hashtag-based Instagram image retrieval & recognition **Mar'17-Apr'17**

- Modeled a software to retrieve and store images based on **Hashtag Search** from a public **Instagram account**. These images are then fed to Convolutional Neural Networks for **image recognition** which returns the prediction accuracies for the top 5 image categories. Performed smart analysis of comments, likes, tags and stored in **Cassandra**.
- Technologies used:** *Python 2.7, Cassandra, Tensorflow, CNN.*

ACADEMIC PROJECTS

3-D Scanner **Sep'17- Dec'17**

- Constructed a **3-Dimensional** scene from multiple **Structured-light** scanning of it by calibrating the projector and obtaining 2D-3D pixel correspondences. Performed stereo calibration and stereo triangulation to get depth maps.
- Technologies used:** *Python 2.7, OpenCV 3.3.0*

Face Detection/Tracking **Sep'17-Dec'17**

- Developed a **Face-Detector** to detect a face in video frame using pre-trained **Viola-Jones** detector, and then to track the face throughout the video frames using **CamShift**, **MeanShift**, **Particle Filter**, **Kalman Filter**, **Optical Flow** trackers.
- Observed that the **Best Face Tracking** results were obtained from the **Optical Flow** algorithm with an **accuracy of 90%**.
- Technologies used:** *Python 2.7, OpenCV 3.3.0*

Binary Segmentation in images **Sep'17-Dec'17**

- Designed an interactive semi-automatic binary segmentation model to segment the foreground objects from its background. After the user provides sparse background-foreground markings, the model computes **Simple Linear Iterative Clustering (SLIC) super-pixels**, and runs a graph-cut algorithm.
- Technologies used:** *Python 2.7, OpenCV 3.3.0.*

Email Spam Filtering **Sep'17-Dec'17**

- Modeled an Email Spam filter that classifies every email as spam or ham by following the **multinomial approach of the Naïve Bayes Algorithm**.
- Applied the **Laplacian smoothing** and tuned the parameters to improve the **testing accuracy from 88.3% to 91.8%**.
- Technologies used:** *Python 2.7, Artificial Intelligence.*

Intelligent Pacman Project **Oct'17-Nov'17**

- Programmed a **Pacman game** by using variety of artificial intelligence techniques such as **Reinforcement Learning**, **Probabilistic Inference** and generic graph search algorithms such as Depth First Search, **A-star with heuristics**. This enabled the pacman to find the **most optimum route** to the food in different game versions.
- Implemented **Mini-max**, **Alpha-beta pruning**, **Probabilistic Inference** and **Expectimax** algorithms to enable multi-agent pacman adversarial search.
- Technologies used:** *Python 2.7, Artificial Intelligence*

ACHIEVEMENTS & EXTRA-CURRICULARS

- Runners up for the Overall Best Project (Auto-scaling of Application Servers)** sponsored by Calsoft and **domain winners for the Best Project** at 'Impetus and Concepts (INC)', a renowned technical festival.
- Received **State Level Excellence Award** in the game of Tennis.