

SURAJ MANIYAR

3141 Kings Court, Apt-A, Raleigh 27606, North Carolina | smaniya@ncsu.edu | +1-919-946-7359
[linkedin.com/in/suraj-maniyar](https://www.linkedin.com/in/suraj-maniyar) | github.com/suraj-maniyar

EDUCATION

- North Carolina State University, Raleigh, North Carolina** Expected May 2019
Master of Science in Electrical Engineering, GPA: 3.66/4.0
Courses :- Data Science | Digital Imaging Systems | Probabilistic Graphical Models for Signal Processing and Computer Vision | Spatial and Temporal Data Mining | Design of a Robotic Computer Vision System for Autonomous Navigation | Computer Vision
- Veermata Jijabai Technological Institute (VJTI), Mumbai, India** Jun 2013 – Jul 2017
Bachelor of Technology in Electronics Engineering, GPA 7.72/10.0
Courses :- Signal Processing | Robotics | Image Processing | Computer Programming | Embedded Systems | Control Systems

EXPERIENCE

Project Intern, National Instruments, India Aug 2015 - Nov 2015
Task Learning Robot

- Implemented a Computer-Vision based approach for 'Robot Learning from Demonstration' using industrial robotic arm Scorbot ER-VII
- Shortlisted in the **top 20 teams** for the **National Level Contest, NIYANTRA**, organized by National Instruments, India

Research Intern, Complex and Non-linear Systems, V.J.T.I., Mumbai Nov 2015-Dec 2015
Model Predictive Control Technique for an Inverted Pendulum

- Optimized a MATLAB Simulink code for an inverted pendulum using Model Predictive Control technique
- Implemented the optimized code on TMS 320F28335 Digital Signal Processor

TECHNICAL SKILLS

Programming Languages : Python, C++, Java, Linux scripting
Frameworks & Libraries : PyTorch, Tensorflow, Keras, OpenCV
Softwares & OS : Robot Operating System (ROS), MATLAB, LabVIEW, Linux (Ubuntu), Windows
Hardware : Raspberry Pi, Beaglebone Black, NI-myRio development board, AVR series microcontrollers

PROJECTS

Design of a Robotic Computer Vision System for Autonomous Navigation (ROS, C++, Python, OpenCV) Jan 2018 – May 2018

- Implemented VINS-Mono and ORB SLAM2 algorithms separately, for an aerial robot Blimp for mapping a construction site
- Used NVIDIA Jetson TX1 development board and monocular camera with an IMU for localizing and mapping the surrounding which used Point Cloud Library (PCL) and Robot Operating System (ROS)
- Co-ordinated with Hardware and Context Awareness teams to integrate their work to develop a fully functional prototype

Deep Visual Attention Prediction (Python, Keras, Tensorflow, OpenCV) Apr 2018

- Replicated the results from the paper titled 'Deep Visual Attention Prediction' which predicts human eye fixation on view-free scenes
- Obtained an accuracy of **64%** by incorporating multi-level saliency predictions

Respiratory Rate Estimation (Python, Keras) Dec 2017

- Estimated the respiratory rate of a human based on accelerometer data, heart rate and body temperature using Ridge Regression and Neural Networks separately with a **RMSE of 4.58**
- Reduced the RMSE to **3.68** by incorporating temporal dynamics using Hidden Markov Model (HMM)

Foraminifera Image Segmentation using Markov Random Field (MRF) (Python, OpenCV) Dec 2017

- Used MRF based approach called Graph-Cut to segment the chambers of a foraminifera (marine species) from its edge probability map
- Obtained an accuracy of **71.40%** using morphological refining and watershed transformation

Stock Trading using Machine Learning (Python, Keras) Sept 2016 - May 2017

- Implemented a Machine Learning based approach to maximally increase the profits involved in stock trading
- Optimized the user's portfolio and implemented technical analysis using Neural Networks and Reinforcement Learning separately to suggest best actions (buy, sell or hold) to the user

CO-CURRICULAR ACTIVITIES

- Senate member of **Society of Robotics and Automation (S.R.A.)**, V.J.T.I. which deals with robotics, machine vision and automation
- Managed and conducted workshops with a team of 10, to teach students about line-following robots, embedded systems, Bluetooth technology and Internet of Things (IoT)