

# Suriya Narayanan Lakshmanan

<https://suriyanitt.github.io> | <https://in.linkedin.com/in/suriya-narayanan> | Santa Clara | 412.626.8524 | [suriyanarayanan987@gmail.com](mailto:suriyanarayanan987@gmail.com)

## EDUCATION

### CMU, ROBOTICS INSTITUTE

#### MASTER'S IN COMPUTER VISION

Advised by Prof. Srinivasa Narasimhan

Dec 2018 | Pittsburgh, PA

Cum. GPA: 3.83/4.0

### NIT, TIRUCHIRAPPALLI

#### BACHELOR OF TECHNOLOGY IN ELECTRICAL AND ELECTRONICS ENGINEERING

May 2014 | Tiruchirappalli, India

Cum. GPA: 8.8 / 10.0

## COURSEWORK

Intro to Machine Learning  
Intro to Computer Vision  
Math fundamentals for Robotics  
Visual Learning and Recognition  
Geometry based maths in Vision  
Computational Photography

Algorithms and Data Structures  
Operating Systems  
Object Oriented Programming  
Digital Signal Processing

## SKILLS

### PROGRAMMING

C • C++ • Python • Matlab  
• OpenCL •  $\LaTeX$

### LIBRARIES

OpenCV • PyTorch (Python and C++)  
• ROS • PCL • Numpy • TensorFlow  
(Python and C++) • scikit-learn

### OPERATING SYSTEMS

Linux • Windows • Android

### OTHER SOFTWARE

Git • Microsoft Office • GIMP

## EXPERIENCE

### NVIDIA CORP. | COMPUTER VISION ENGINEER

Feb 2022 - Present | Bangalore, India

- Accelerated obstacle detection algorithm by porting CPU code to CUDA achieving 10x speedup

### NVIDIA CORP. | COMPUTER VISION ENGINEER

Jan 2021 - Jan 2022 | Santa Clara, CA, USA

- Prototyped and designed obstacle detection algorithm for small obstacle sizes
- Implemented the designed obstacle detection algorithm to run on CPU
- Designed porting the obstacle detection algorithm to Nvidia GPU using CUDA

### CYNGN INC (SELF-DRIVING VEHICLE COMPANY) | PERCEPTION ENGINEER

Feb 2019 - Oct 2020 | Menlo Park, CA, USA

- Designed and implemented traffic light detection system from scratch consisting of components such as traffic light detection, traffic light recognition and tracking
- 2d object detection using PyTorch
  - Built on top of open source model and finetuned by
    - \* Incorporating architectural advances such as weighted BIFPN from 'EfficientDet' paper
    - \* Using Swish activation over ReLU activation
    - \* Tuning hyperparameters such as learning rate, weight decay, focal loss parameters, loss weightage parameters using Bayesion Optimization package called 'Ax'
    - \* Creating hierarchy of classes to better differentiate between classes and easily adopt with different annotation formats
  - Ported 2d detection model to TorchScript and implemented 2D detection pipeline in C++
- Evaluation of lidar based SLAM algorithms for indoor applications
- Analyzed the depth error of stereo and came up with an optimal baseline width taking into account target working range, camera FOV and permissible blindspot
- Prototyped multilayer stixels world work on the in-house stereo camera setup which involved camera calibration, stereo rectification, tuning SGBM parameters and understanding stixels computation
- Created requirements for choosing camera format and lens focal length for required detection range and Field of View
- Designed and implemented software based Lidar-Camera synchronization system that is capable of operating at 10Hz under certain configurations

### SAMSUNG RESEARCH AMERICA | COMPUTER VISION RESEARCH INTERN Think Tank Team

May 2018 - August 2018 | Mountain View, CA, USA

- Developed human pose datasets for proprietary imaging sensors using unsupervised domain adaptation
- Developed a deep learning based human pose estimation network that can predict human poses on frames obtained from the proprietary imaging sensor
- Deployed the above developed network by creating a C++ application using TensorFlow APIs for building and executing the deep learning graph

## TEXAS INSTRUMENTS | COMPUTER VISION INTERN

May 2013 – July 2013 | Bangalore, India

- Improved an existing homography based Ground Plane Detection by **10%**. [*Ground plane detection, Patent 2017*]. [*Improved ground plane detection in real time systems using homography, ICCE 2014*]

## ACADEMIC PROJECTS

### UNSUPERVISED SEGMENTATION DATA GENERATION

October 2018 - December 2018 | CMU, Pittsburgh

Interpolated semantic segmentation labels to frames in between key frames in NYU depth dataset v2 videos using dense depth as supervision for intermediate frames improving mIoU by 2% over FCN baseline finetuned on NYU v2 dataset

### RGB SUPER SLOMO USING HIGH FPS DYNAMIC VISION SENSOR

October 2018 - December 2018 | CMU, Pittsburgh

Developed a deep learning network to produce high frame rate RGB video from a low frame rate RGB video using optical flow derived from high frame rate dynamic vision sensor video as supervision and achieving comparable performance as that of using high FPS RGB video supervision

### SMART RECONSTRUCTION

January 2018 - May 2018 | ILIM Lab, CMU, Pittsburgh

Reconstructed traffic from a single stationary camera using keypoint detection and ground plane assumption (similar to the work done by Beyond Pixels paper), tracking and geometric constraints while stabilizing the camera

### WEAKLY SUPERVISED OBJECT DETECTION

March 2018 - March 2018 | CMU, Pittsburgh

Implemented weakly supervised object detection algorithm: WSDDN as part of course project in TensorFlow

### SCENE CLASSIFICATION

September 2017 | CMU, Pittsburgh

Implemented scene classification using Spatial Pyramid Matching from scratch as part of course project in Matlab

### DIGITAL ART USING SFM

October 2017 - November 2017 | CMU, Pittsburgh

Developed an application to create portrait effect from single camera using SFM and 3D segmentation as part of course project using OpenCV C++

### INTELLIGENT INPAINTING

October 2017 - November 2017 | CMU, Pittsburgh

Developed an application that removes a person from an image from a single click using pedestrian detection, semantic segmentation and exemplar inpainting in C++

## PUBLICATIONS

- Improved ground plane detection in real-time using homography, ICCE [\*link to the paper\*](#)
- Understanding performance benefit of asynchronous data transfers in OpenCL programs executing on media processors, HiPC [\*link to the paper\*](#)