Manas Jyoti Buragohain

Email: manasjb@umich.edu
Website: manasburagohain.github.io
Mobile: 734-881-5295
Github: github.com/manasburagohain
LinkedIn: linkedin.com/in/manas-jyoti

Education

- **University of Michigan** Ann Arbor, MI, USA
  Master of Science in Robotics *Sept 2019 - April 2021 (expected)*
  GPA: 3.90/4.0
  Courses: Deep Learning for Computer Vision, Foundations of Computer Vision, Advanced Topics in Computer Vision, Machine Learning, Applied GPU Programming
- **Delhi Technological University** Delhi, India
  Bachelor of Engineering in Electronics and Communication Engineering *Aug 2013 - May 2017*
  Courses: Digital Image Processing, Computer Vision, Pattern Recognition, Robotics & Object Tracking
  Leadership Experience: Team Lead for DTU - AUV Student Research Team (Aug 2014 – May 2017)

Skills Summary

- **Concentration Areas**: Computer Vision, Deep Learning, 3D Reconstruction
- **Progamming Languages**: Python, C, C++, JavaScript, HTML, Bash
- **Tools**: PyTorch, PyTorch3D, OpenCV, CUDA, Caffe, NumPy, Matplotlib

Research Experience

- **Johnson AI Lab, University of Michigan** Ann Arbor, MI, USA
  Graduate Research Assistant | Advisor: Justin Johnson *Jan 2020 - Present*
  - Designed a grid based point cloud prediction network using ResNet-50 backbone achieving state of the art performance.
  - Developed a novel approach for point cloud refinement using local context and attention-based supervision through an augmented Transformer Architecture.
  - Implemented differentiable Top-K selection through Reparameterizable Subset Sampling for face selection using customized CUDA Kernels.

Work Experience

- **NXP Semiconductors** Noida, India
  ADAS Engineer, Functional Validation *Aug 2017 - feb 2019*
  - Coded C++ programs for Advanced Driver Assistance System (ADAS) system to perform Lane and Pedestrian Detection using SSD architecture optimized for embedded systems.
  - Executed continuous testing and integration of Low Light Noise Reduction and Histogram of Gradients Generation modules for accelerating hardware computation on ADAS system.
  - Formulated and streamlined C++ unit tests of FlexCAN and LINFlex protocol modules for intra vehicular communication.

Selected Projects

- **Sparse Neural Generative Inference Based Pose Estimation**: Attempted to build a particle filter based pose estimator where each particle learns latent embedding to infer pose, object likelihood, and re-sampling objective iteratively. Key: Particle Filter, PyTorch, Pose Estimation, RGB-D
- **Single Image 3D Reconstruction based on Conditional Generative Adverserial Networks**: An end-to-end conditional GAN framework for generating 3D objects from single RGB image. We achieve improved qualitative 3D reconstructions compared to the Pixel2Mesh baseline. Key: 3D Reconstruction, PyTorch, Deep Learning, GAN
- **Probabilistic Data Association for Semantic SLAM with Loop Closure Detection**: Replicate and improve upon the work of Bowman et al with augmentations to object detection framework along with incorporation of loop closure for better offline map generation. Key: SLAM, Objection Detection, Stereo Vision, Depth Estimation, PyTorch, Python, MATLAB
- **Robot Middle-ware Development**: Developed a web-based dynamic simulator and set-point controller for mobile manipulators like Fetch, Baxter and Sawyer. Implemented motion planners like A*, Bi-directional RRT-connect and RRT* in the simulator for any obstacle environment. Key: JavaScript, Path Planning, Manipulator Modelling, A-Star, RRT
- **Slam and Path Planning for MBot**: Explored and implemented various mapping, path planning and motion control algorithms on a simulation model for a differential drive robot. Key: C++, LIDAR, SLAM, A-Star, Path Planning
- **6-DOF Serial Link Robotic Manipulator**: Produced a codebase in Python to drive serially connected motors autonomously, employing object detection using a kinect camera suite for pick-n-place operation. Key: Python, Manipulator Modelling, Objection Detection, OpenCV, Path Planning - Smoothing, State Machines
- **Mobile Inverted Pendulum System**: Designed a cascaded control architecture to balance a two-wheeled robot and to autonomously drive in pre-defined trajectories. Key: C, inverted Pendulum, Trajectory Following, IMU, PID, Beaglebone, Robot Control Library

Publications

- **Fish species classification using graph embedding discriminant analysis**: Conference on Machine Vision and Information Technology, 2017. Authors: **Manas Jyoti Buragohain***, Snigdhaa Hasija*, and S. Indu