VIGNESH SUNDARARAJAN

📞 +1-(765)-775-7397   ✉ sundar18@purdue.edu   🔗 linkedin.com/in/sundararajan-vignesh   🔗 github.com/vigneshsundararajan

Education

Purdue University | Master of Science - Aeronautics and Astronautics Engineering   **West Lafayette, USA**
Extra Curricular Work: Aerodynamics Team Member - Purdue Electric Racing | 🔗 **Blog Link**   2021 - Present

SRM University | Bachelor of Technology - Mechanical Engineering   **Chennai, IN**
Extra Curricular Work: *Design Lead and Team Vice Captain for collegiate FSAE Team*   2016-2020

Skills

Programming: C++, Python, MATLAB, SQL, R, Lua, HTML, CSS, Javascript, LaTeX
Frameworks/Libraries: Tensorflow/Keras, PyTorch, Scikit-Learn, Numpy, Eigen, openCV, Matplotlib, Pandas, Dask, Plotly, Folium, Flask
Developer Tools: AWS, GCP, Git, Vim, Linux, openMPI, CMake, Meson, ROS, Gazebo, Docker, Kubernetes, Confluence, Jira

Experience

Graduate Data Science Researcher | The Purdue Data Mine + John Deere   **West Lafayette, USA**
Tools Used: AWS, Python, SQL, Spark, ARCGIS, Machine Learning, Computer Vision, Geospatial Data Analysis   Aug 2022 - Present

- Developing an automated, large scale system to identify wildlife conservation opportunities for the Pheasant and Quail Foundation(PQF) by ingesting satellite data and producing profitability maps in ARCGIS for agricultural planning.

Data Scientist | Dauch Center for the Management of Manufacturing Enterprises + Wabash National   **West Lafayette, USA**
Tools Used: Python, Scikit-learn, Pandas, Dask, Parquet, Plotly, Folium, Geopy, Telematics   Jun 2022 - Aug 2022

- Performed a univariate exploratory data analysis to study sensor ping frequency for Wabash's nationwide fleet of trucks.
- **Achieved a 60×** speed improvement and an 85% **storage use reduction** by implementing a parallelized data ingestion and cleaning pipeline to handle multi-year raw Telematics data of the order of 100GB, using Python and Dask. Wrote documentation on the dataset.
- Built Python software to visualize high-traffic trucking hubs across North America using yearly fleet GPS data as input and employing the Mini-Batch K-Means clustering algorithm. Presented findings to the business end of both organizations.

Projects

Object tracking and following Robot in Gazebo | 🔗 Github Link   **West Lafayette, USA**
Tools Used: Python, ROS, OpenCV, Linux, Computer Vision   Apr 2022

- Computed distance to the object by performing image masking on a real time RGB video feed and calculating it's centroid.
- Developed a proportional velocity controller to check for the object in robot's FOV before following it.

Monocular Vision based Navigation of a 4-wheeled robot | 🔗 Github Link   **West Lafayette, USA**
Tools Used: Python, C++, CMake, Computer Vision, OpenCV, ROS, Raspberry Pi, L293D driver board   Mar 2022 - Apr 2022

- Calibrated a monocular camera interfaced with a Raspberry Pi acting as the vision system for a 4 wheeled DC Motor driven robot. Performed 3D localization of the robot in the environment using the Fiducials ROS package.
- Built a PID controller based motion algorithm to navigate the robot towards an Aruco marker placed in the environment.
- Developed a line following mechanism by interfacing IR sensors to the robot as a fallback in case the vision system fails.

Image Super-Resolution using a GAN with a Perceptual Loss Function | 🔗 Github Link   **West Lafayette, USA**
Tools Used: Python, Tensorflow/Keras, Matplotlib, Numpy, Pillow   Aug 2021 - Dec 2021

- Implemented SRGAN, based on the ResNet architecture(16 Residual Blocks with skip connections), with a perceptual loss function that uses high level feature maps of the VGG network
- Trained the neural network for 50,000 steps on the DIV2K data set and visualized the 4× upsampled images using matplotlib

CNN Autoencoder for Image Denoising | 🔗 Github Link   **West Lafayette, USA**
Tools Used: Python, Pytorch, Deep Learning, MNIST   Oct 2021

- Performed data augmentation on the MNIST dataset by adding a Gaussian Noise with $\mu = 0$ and $\sigma = 0.5$
- Built a Convolutional Neural Network based autoencoder model to remove noise from augmented dataset and **achieved an average loss of 17.33 after 10 epochs**

KNN Classifier to detect potential Credit Card fraud | 🔗 Blog Link   **West Lafayette, USA**
Tools Used: Python, Scikit-learn, Predictive Modelling   Sep 2021

- Built a predictive model to determine whether a credit card transaction is fraudulent or not and **achieved** 95.468% **training accuracy and** 90.625% **test accuracy**. Employed K-Nearest Neighbors Classification with cross validation to optimize the hyperparameters.

Miscellaneous Projects

- **Neovim Code Editor config using Lua** - *Integrated LSP and fuzzy finding capabilities with **4ms startup time*** | 🔗 **Github Link**
- **Portfolio website development using Hugo** - *Javascript, HTML, CSS, Markdown* | 🔗 **Github Link**