

Anirudh Iyengar Kaniyar Narayana Iyengar

[in linkedin.com/in/anirudhiyengar-kn](https://www.linkedin.com/in/anirudhiyengar-kn) [GitHub anirudh6415](https://github.com/anirudh6415) [✉ akaniyar@asu.edu](mailto:akaniyar@asu.edu) [📞 480-919-4286](tel:480-919-4286) [Portfolio](#)

Education

MS in Robotics - AI - Master's Thesis in Computer Vision

Arizona State University

Expected December 2024

(GPA: 4.00) Tempe, AZ

B.Tech in Computer Science and Engineering

Dayananda Sagar University, India

August 2016 – May 2020

Bengaluru, KA

Technical Skills

Languages: Python, C, C++, SQL, HTML, MATLAB, Bash, R

Frameworks: PyTorch, Docker, Git, SciPy, PySpark, Scikit-Learn, OpenCV, mmdetection, NumPy, Huggingfaces, mmsegmentation, VScode, Jupiter, Detectron2, Pandas, Open AI API.

Tools: CVAT, Tableau, ClearML, Jira, Jenkins, Google Analytics, MySQL.

Concepts: Deep Learning, Computer Vision, Statistical Machine Learning, Data Science, AI Model Optimization.

Experience

JLiang Lab - Arizona State University

Research Aide - Computer Vision

July 2023 - December 2023

Tempe, AZ

- Collaborating with **Valley Wise Health** on deep learning models for lung disease classification, localization, and segmentation. Applying advanced techniques and computed, analyzed image data to optimize models based on lab research in **POPAR, Adam, Ark, and PEAC**.
- Facilitating the integration of models like **Dino, Intern image, UPerNet, and Mask Dino** with the **mmdetection, Detectron2** frameworks to advance lung disease diagnostics and through the utilization of transfer learning techniques.

Rajesh TM Lab - Dayananda Sagar University

Research Aide - Computer Vision

June 2020 – December 2020

Bengaluru, KA

- Performed preprocessing, analysis, and annotation **5,000** driving images of large datasets.
- Utilized the **CVAT** open data annotation tool for precise **data annotation** and analyzed data using **ML** algorithms.
- Demonstrated analysis and annotation of data, resulting in the submission of a comprehensive report outlining findings.

Centre for Artificial Intelligence and Robotics, DRDO Lab

Student Trainee - Computer Vision

January 2020 – May 2020

Bengaluru, KA

- Processed a total of 389 image pairs, including 194 training image pairs and 195 test image pairs of stereo image datasets for KITTI 2012, and KITTI 2015.
- Appraised GWC-Net using deep learning techniques and achieved **5%** higher performance in disparity estimation.
- Demonstrated the effectiveness of GWC-Net by improving the accuracy of disparity estimation on multiple datasets.

Projects

Sentiment Analysis Assisted Time Series Stock Prediction [*Python, PyTorch,SQL*]

January 2024 - April 2024

- Hypothesized and currently developing an advanced stock prediction model using natural language processing for precise analysis of financial news impact on stock movements.
- Currently developing and incorporating LSTM and large language models (LLMs) for enhancing stock price forecasting.

Detection For Autonomous Driving using Argoversehd [*Python, PyTorch, ClearML*]

May 2023 - June 2023

- Optimized object detection for autonomous driving scenarios leveraging the Argoversehd traffic dataset.
- Fine-tuned **YOLOv8** model for vehicles, persons, traffic signals, and road signs. Achieved a **1.5%** higher mean Average Precision (mAP) than the original Argoversehd implementation and visualized using **ClearML** tool.

Function Integration on Colonoscopy Polyp Dataset [*Python, PyTorch, OpenCV*]

January 2023 - April 2023

- Collaborated to design a **Swin Transformer-based UNET** architecture, integrating branches for classification, object localization, and Colon polyp image segmentation. Developed an all-in-one model surpassing individual model performance by **15%**, guided by Prof. Dr. Jianming Liang.

Anytime Stereo Image Depth Estimation using KITTI2012 [*Python and PyTorch*]

January 2023 - April 2023

- Engineered an innovative approach using a Unet model that incorporated a disparity network and residual map for each layer of the decoder, Improved the performance by **aggregating 2 decoders** in the model one after the other.
- Successfully predicted disparity images in real-time for each layer of the decoder, resulting in 5% performance better than the original implementation.