

Anirudh Iyengar Kaniyar Narayana Iyengar

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

Education

MS in Robotics and Autonomous Systems - AI

January 2023 – December 2024

Arizona State University

Tempe, AZ

- **Relevant Coursework:** *Artificial Intelligence, Data Visualization, Intro Digital Image Processing, Image Analytics & Informatics, Applied Linear Algebra, Perception in Robotics, Modeling and Control of Robots.*

B.Tech in Computer Science and Engineering

August 2016 – May 2020

Dayananda Sagar University

Bengaluru, KA

Technical Skills

Languages: Python, C, C++, SQL, HTML, MATLAB, Bash, R, CSS, Javascript, D3.js.

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

Tools/Platforms: CVAT, Tableau, ClearML, Jira, Jenkins, Google Analytics, MySQL, AWS, SQLAlchemy

Experience

Synapse Labs

June 2024 - Present

Machine Learning Engineer

Scottsdale, AZ

- Automated an end-end pipeline for payment reconciliation using **OCR algorithms and LLMs** by achieving **80% reconciliation rate**.
- Leveraged **EC2, S3, Quicksight, PySpark, SQLAlchemy** for processing 1 million data processing and visualization.
- Contributed dashboards towards presenting complex technical insights to drive strategic business outcomes.

JLiang Lab - Arizona State University

July 2023 - July 2024

Research Aide and Teaching Assistant - Computer Vision

Tempe, AZ

- Collaborated with **Valleywise Health** on deep learning models for lung disease zero-shot learning, localization, segmentation, long-tail classification, and regression with the **mmdetection** and **Detectron2** frameworks.
- Applied transfer learning and contrastive learning to achieve **9th position** in the Long-Tail Classification challenge.
- Served as a *Teaching Assistant* for **CSE/BMI 591/BME 598: Image Analytics & Informatics (Spring 2024)** under Dr. Jianming Liang by advising and guiding students on advancements in medical image processing.

CAIR, DRDO Lab

January 2020 – May 2020

Student Trainee - Computer Vision

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 a generative adversarial network(GAN) for depth estimation.
- Demonstrated the effectiveness of GWC-Net by improving the **5%** accuracy of real-time depth estimation on multiple datasets.

Projects

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

Present

- Hypothesized and currently developing an advanced stock prediction model using natural language processing to analyze financial news impact on stock movements precisely.
- 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.