PORTFOLIO

1. Trash Bag Segmentation, Detection, Classification, Tracking, and Counting On Real Time

We designed an advanced AI project aimed at optimising trash bag collection processes for NetVision Company . Leveraging the power of YOLOv8, our team created a sophisticated system **capable of detecting, classifying, segmenting, tracking, and estimating the size of trash bags within real-time video streams**. My part included reading research papers and training models on YOLOv 8. With this project, we participated in the capstone competition 2023 at Woosong University. We defeated **more than 200 teams** from 29 departments and got the Woosong University President Award. Here is the video of the awarding our team : Video Link . For security purposes, I will not give access to the project, but tell briefly in the following paragraphs:

Technical Strategy:
YOLOv8 Integration: Employed YOLOv8, a state-of-the-art object detection model, to effectively identify and categorise trash bags within live video streams.
Real-time Processing: Engineered algorithms for immediate analysis of video data to swiftly and accurately identify and track trash bags.
Precise Segmentation: Utilised advanced segmentation techniques to precisely isolate trash bags in video frames for detailed analysis.

Project Deliverables:
Detection & Classification: Accurate identification and classification of various types of trash bags (e.g., recyclable, non-recyclable) within video feeds.
Segmentation & Tracking: Precise isolation and continuous tracking of trash bags throughout the video streams for efficient monitoring.
Size Estimation: Providing estimations of trash bag sizes for optimal collection planning and resource allocation.

Tools Used: PyTorch, OpenCV, YOLOv8, Ultralytics
ML Algorithms: YOLOv8 object detection

2.Real Time Face Mask Detection with PyTorch and OpenCV

This project utilises PyTorch, OpenCV, and the MTCNN for face detection and mask presence identification. The model is trained on labelled data to accurately detect the presence of face masks in images or real-time video feeds.
Tools Used: PyTorch, OpenCV, MTCNN model
ML Algorithm: Resnet 18, Resnet 50 via Transfer Learning

Project link | Presentation link | Demo Link

3. Multiclass Weather Classification using PyTorch and CNNs

Employing PyTorch and Convolutional Neural Networks, this project categorises images into various weather conditions. By utilising a custom dataset of weather from kaggle datasets and employing techniques like data augmentation and transfer learning, the model's accuracy and performance are enhanced.
Tools Used: PyTorch, Python
ML Algorithm: CNN, Resnet 9

Project link | Presentation Link

4. StatFlow EDA Web App using Streamlit and

StatFlow EDA App is a web-based application facilitating Exploratory Data Analysis (EDA) on user-uploaded datasets. Created using Streamlit and data science libraries, the app is hosted on Streamlit cloud. It offers features like descriptive statistics, data cleaning, visualisation, and correlation analysis.
Tools Used: Streamlit, Pandas, Numpy, Matplotlib, Seaborn, Pandas Profiling
ML Algorithm: CNN, Resnet 9

Project link | Demo link

5. Indoor Air Quality Monitoring System using IoT and ThingSpeak

This project is dedicated to an innovative IoT-based indoor air quality monitoring system. Leveraging the ESP Wi-Fi module and ThingSpeak cloud platform, the system integrates various sensors to track and display real-time air quality parameters. Additionally, it incorporates data from the OpenWeather API to compare indoor and outdoor air quality.

The system's key features include:

- IoT Integration: Utilising ESP Wi-Fi modules and various sensors to collect indoor air quality data.
- ThingSpeak Cloud: Storing and processing real-time air quality parameters received from different sensors.
- Real-Time Dashboard: Displaying air quality metrics in an interactive and user-friendly dashboard.
- OpenWeather API Integration: Fetching outdoor air quality data to facilitate a comparison with indoor air quality metrics.

Tools Used: Arduino, C language, ThingSpeak Cloud, ESP Wi-Fi, Sensors, Streamlit

Project Link | Presentation link | Demo Link

6. Auto-Detection of Kidney Conditions with Deep Learning Methods

Using architectures like ResNet50, VGG16, VGG19, and U-Net, this project automates the detection of kidney disease within medical images. Although U-net is used for segmentation, I used it for classification tasks by changing the last layers of the neural network. I achieved higher accuracy and precision using U-net autoencoder architecture. The models leverage deep learning and transfer learning techniques for accurate kidney illness identification.
Tools Used:PyTorch, Python
ML Algorithm: CNN, ResNet 50, VGG16, VGG19, U-Net via Transfer Learning

Project Link | Presentation Link | Report Link

7. Bank term marketing classification and subscription prediction

This project aims to classify potential customers for a bank's term deposit marketing campaign. Using machine learning classification techniques, the model predicts whether a customer is likely to subscribe to a bank's term deposit based on various characteristics and historical data. This predictive model helps to target potential subscribers more effectively, optimise marketing strategies and increase subscription rates. Overall, I used 5 different machine learning algorithms and found the logistic regression accurate.
Tools Used: Numpy, Pandas, Matplotlib, Scikit-Learn
ML Algorithm: KNN, Decision Tree, Random Forest, Logistic Regression, SVM

Project Link | Presentation Link

8. Insurance Price Prediction using PyTorch
This project focuses on predicting insurance prices by employing PyTorch and machine learning algorithms. Trained on relevant datasets, the model predicts insurance premiums based on various factors.
Tools Used: Numpy, PyTorch
ML Algorithm: Neural Network, Linear Regression

Project Link

9. House Price Prediction using PyTorch

Utilising PyTorch and machine learning, this project predicts house prices based on relevant datasets from kaggle. The model is trained to predict house prices using various features.
Tools Used: Numpy, PyTorch
ML Algorithm: Neural Network, Linear Regression

[Project Link](#)