Muhammad Aliyan

PortFolio: https://aliyan16.github.io/

LinkedIn: linkedin.com/in/muhammad-aliyan-b0525b26b/

Email: aliyanm12376@gmail.com

Phone: +923359354084

EDUCATION

National University of Science and Technology (NUST- CEME) Rawalpindi

Bachelor of Engineering in Computer Engineering

EXPERIENCE

Freelance (Fiverr) June 2023 – Present

- Designed and implemented machine learning and deep learning models for diverse applications, including computer vision, image classification, and predictive analytics.
- Developed various projects for clients, utilizing technologies such as TensorFlow, Keras, and OpenCV to address specific project requirements.
- Completed multiple web development projects for clients, leveraging ReactJS, WordPress, and Shopify to build and enhance websites and online platforms.
- Customized and optimized solutions based on client needs, providing end-to-end support from initial concept through to deployment and maintenance.
- Applied advanced techniques in neural networks, including convolutional neural networks (CNNs), for tasks such as object detection and image segmentation.

PROJECTS

Autonomous Vehicle System using Raspberry Pi and Classical Image Processing | Python, OpenCV, Raspberry Pi Camera, GPIO

- Designed and developed a self-driving car prototype using Raspberry Pi equipped with a camera module to simulate autonomous navigation in real-world scenarios.
- Implemented classical image processing algorithms (edge detection, contour analysis, color segmentation) using OpenCV to recognize obstacles, road boundaries, and directional cues in real-time.
- Developed decision-making logic enabling the vehicle to adjust its path dynamically based on detected objects and environmental observations.
- Utilized Raspberry Pi GPIO pins to control vehicle motion (e.g., motor speed and direction) based on visual input and obstacle detection.
- Created a lightweight and efficient pipeline to process camera input frames, extract meaningful features, and trigger navigation responses with minimal latency.
- Ensured robustness of the system by testing in various lighting and surface conditions and optimizing the image processing pipeline for performance and reliability.
- Achieved obstacle avoidance and basic lane-following capabilities without the use of deep learning, demonstrating the power of classical CV techniques on low-power hardware.

Mission Logging and Tracking System for Pakistan Army Pilots | Flutter, Dart, Django, Python, SQL, HTTP

- Developed a cross-platform mobile application using Flutter and Dart to streamline mission logging and tracking for Pakistan Army pilots and their units.

- Implemented secure user authentication for pilots and unit personnel, allowing pilots to log new missions and view their mission history.
- Designed a Django backend with RESTful APIs (using Django REST Framework) to handle mission data, user authentication, and real-time updates.
- Integrated SQL database for efficient storage and retrieval of mission details, including pilot information, mission locations, timestamps, and status.
- Enabled real-time mission tracking for units, allowing them to view recently added missions, apply filters (by pilot, location, or date), and assign follow-ups.
- Utilized HTTP requests for seamless communication between the Flutter frontend and Django backend, ensuring data consistency and reliability.
- Enhanced user experience with responsive UI/UX design, including mission summaries, search functionality, and notification alerts for new missions.
- Ensured data security with encrypted transmissions and role-based access control (RBAC) to restrict unauthorized access.

Anti-Sleep Detection System for Vehicle Drivers | Python, C++, Arduino, ESP32

- Designed and implemented an anti-sleep detection system for vehicle drivers using Arduino, ESP32, and various sensors.
- Developed a system to continuously monitor the driver's eyes using cameras and sensors, with an algorithm to detect drowsiness based on eye closure duration (3-5 seconds).
- Integrated an alarm mechanism to alert the driver if drowsiness was detected and programmed the system to automatically stop the vehicle if the driver failed to respond.
- Implemented signaling features to notify other drivers of potential hazards when the vehicle is stopped due to drowsiness.
- Created a mobile app for real-time monitoring of driver and vehicle status, and for remote control of vehicle functions.
- Utilized motors and other hardware components to control vehicle safety features and interface with the vehicle's existing control systems.

Audio classification using CNN | Python, TensorFlow, keras, Librosa, Numpy, Matplotlib

- Developed an audio classification model using deep learning to categorize sound signals into speech, street_music, and noise using the UrbanSound8K dataset.
- Extracted MFCC features (Mel-Frequency Cepstral Coefficients) from audio signals for feature representation and classification.
- Built and trained a deep neural network (DNN) using TensorFlow/Keras, achieving high classification accuracy.
- Visualized spectrograms and waveforms to analyze different sound categories using Librosa and Matplotlib.
- Processed and augmented audio data to improve model robustness and generalization.
- Evaluated model performance using accuracy metrics, confusion matrix analysis, and loss plots.
- Implemented data preprocessing techniques such as feature scaling, label encoding, and train-test splitting for optimized model training.
- Designed and trained the model using Adam optimizer and categorical cross-entropy loss function to handle multi-class classification effectively.

Real-Time Hand and Face Detection Using OpenCV and MediaPipe | Python, OpenCV, MediaPipe, NumPy

- Developed a real-time hand and face detection system using OpenCV and MediaPipe to detect and track hands and faces in live video streams.
- Implemented hand landmark detection to count the number of fingers raised, enabling gesture-based interaction.
- Utilized MediaPipe's Face Detection module to detect faces and draw bounding boxes around them in real-time.
- Processed and visualized hand landmarks and face bounding boxes using OpenCV for real-time video analysis.
- Enhanced the system by adding visual feedback, such as bounding boxes and finger count displays, for better user interaction.
- Leveraged NumPy for efficient array manipulations and calculations during image processing.

CO2 Emission Prediction for City Cars | Python, Ridge, Lasso, ElasticNet

- Predicted CO2 emissions of cars based on features like engine size, cylinders, and model, using city car data.
- Replaced non-numeric variables with dummy variables and scaled numerical features.
- Implemented Ridge, Lasso, and ElasticNet regression models to improve prediction accuracy.
- Tuned models on different train-test splits and evaluated their performance.
- The model outputs CO2 emissions based on user-provided features, helping to identify emissions trends in urban vehicles.

Human Activity Recognition with Smartphone Data | Logistic Regression, L1 & L2 Regularization

- Analyzed smartphone sensor data to classify various human activities.
- Processed data by examining types, encoding activity labels, scaling floating-point features, and calculating correlations between variables.
- Used StratifiedShuffleSplit to maintain class ratios in train-test splits and analyzed the class distribution in both sets.
- Fitted logistic regression models (without regularization and with L1 & L2 regularization) and compared their coefficient magnitudes.
- Evaluated model performance with cross-validation and compared accuracy, precision, recall, F-score, and confusion matrix metrics.
- Visualized confusion matrices for each model, providing insights into prediction accuracy across activities.

Breast Cancer Classification with Neural Network | Python, TensorFlow, Keras

- Developed a neural network model to classify breast cancer as malignant or benign using the breast cancer dataset from sklearn.
- Preprocessed the dataset by scaling features using StandardScaler and splitting data into training and test sets.
- Designed a multi-layer neural network using Keras with a Dense hidden layer and sigmoid activation for binary classification.
- Trained the neural network model using the Adam optimizer and sparse categorical cross-entropy loss function.
- Visualized model performance by plotting accuracy and loss for both training and validation datasets.
- Achieved significant accuracy on test data, correctly identifying breast cancer as malignant or benign.
- Deployed the model for real-time prediction, with an interface that accepts input data and outputs a classification label for tumor type (malignant or benign).

Plant Disease Prediction with Convolutional Neural Networks (CNN) | Python, TensorFlow

- Developed a Convolutional Neural Network (CNN) to classify plant diseases using a dataset from Kaggle containing over 50,000 images of 38 different plant diseases.
- Preprocessed data using image augmentation and normalization techniques, and utilized TensorFlow's ImageDataGenerator for efficient data handling and augmentation.
- Constructed and trained a CNN model with two convolutional layers, max pooling, and dense layers, achieving a validation accuracy of 88.28% over 5 epochs.
- Implemented model evaluation and visualization, including accuracy and loss plots, to monitor training performance.
- Created functions to load and preprocess images for prediction, and to map class indices to class names for easy interpretation of results.
- Saved the trained model in both HDF5 and native Keras formats for deployment and future use.
- Tools: Python, TensorFlow, Keras, Matplotlib, Kaggle API

Skills

Technical Skills

Machine Learning & Deep Learning:

- Proficient in building and deploying machine learning models using Scikit-Learn, TensorFlow, and Keras.
- Experienced in data preprocessing, feature engineering, and model evaluation techniques.
- Skilled in implementing supervised and unsupervised learning algorithms for classification, regression, and clustering tasks.
- Hands-on experience with neural networks, including CNNs for image and audio data, and transfer learning.

Data Analysis & Visualization:

- Expertise in data manipulation and analysis using NumPy and Pandas.
- Proficient in creating insightful visualizations using Matplotlib, Seaborn, and Plotly.

Web Development:

- Skilled in building dynamic and responsive user interfaces using ReactJS.
- Familiar with HTML, CSS, and JavaScript for frontend development.
- Currently learning Flask for backend development to build RESTful APIs and integrate machine learning models into web applications.

Mobile & Full-Stack Development:

- Developed cross-platform mobile apps using Flutter & Dart (e.g., mission tracking system for Pakistan Army).
- Built RESTful APIs with Django (Python) and integrated them with Flutter frontends.
- Skilled in ReactJS for dynamic web interfaces and familiar with HTML, CSS, JavaScript.
- Currently learning Flask for lightweight backend development and model deployment.

Databases & Backend:

- Worked with SQL databases (MySQL, PostgreSQL) for structured data storage and retrieval.
- Used HTTP/API integration (Django REST Framework, Fetch, Axios) for seamless frontend-backend communication.

Version Control & Collaboration:

- Experienced in version control using Git and GitHub for collaborative project development.
- Familiar with DVC (Data Version Control) for managing machine learning datasets and pipelines.

MLOps & Model Deployment:

- Knowledge of MLflow for experiment tracking, model packaging, and deployment.
- Currently learning Docker for containerization and deployment of machine learning models.
- Exploring additional MLOps tools to streamline model lifecycle management, including CI/CD pipelines and monitoring.

Programming & Tools:

- Strong programming skills in Python for machine learning, data analysis, and automation.
- Familiar with OS module for file handling and system operations.

Continuous Learning:

- Actively expanding knowledge in MLOps tools and practices to enhance model deployment and scalability.
- Enrolled in advanced courses and certifications to stay updated with the latest trends in AI, machine learning, and full-stack development.

CERTIFICATIONS

Director Web & IT wing Comppec

Deputy Director Admin & HR Comppec

Deputy Director Marketing team Comppec

Machine Learning With Python By Coursera

Supervised Machine Learning: Regression Classification By Coursera

Unsupervised Machine Learning: Recommenders, Reinforcement Learning By Coursera

Advance Learning Algorithms By Coursera

Machine Learning Specialization By Coursera

Deep Learning Specialization By Coursera

Computer Vision and Image Processing By Coursera

Advance Computer Vision With TensorFlow By Coursera

OpenCV course By FreeCodeCamp