Arpita Sambher

+91-9418772283 arpitasambher777@gmail.com linkedin.com/in/arpitasambher github.com/arpitasambher

EDUCATION

Thapar Institute of Engineering and Technology **2022 – 2026**
B.E. in Computer Science (Minor: Edge AI & Robotics)- ***CGPA - 9.1/10*** *Patiala, India*

Relevant Coursework: Artificial Intelligence, Machine Learning, Data Structures and Algorithms, Database Management, Object-Oriented Programming, Operating Systems, Software Engineering, Computer Networks, Corporate Finance

PROJECTS

EEG-Based Alzheimer's Disease Detection — Deep Learning and Signal Processing

- **Objective:** Developed an AI-powered system for early detection of Alzheimer's disease using 19-channel EEG signals, optimized for low-resource medical settings in India.
- **Signal Preprocessing:** Cleaned EEG signals by removing artifacts such as eye blinks and muscle noise to ensure quality input data for analysis.
- **Feature Extraction:** Extracted key spectral and time-domain features (e.g., entropy, spectral power) to distinguish between Alzheimer's patients and healthy individuals.
- **Deep Learning Pipeline:** Built a hybrid model combining CNNs (for spatial features) and transformers (for temporal dependencies) to classify EEG signals with high accuracy.
- **Model Evaluation:** Achieved improved performance over traditional models (SVM, Random Forest) using metrics such as precision, recall, F1-score, and AUC-ROC.
- **Scalability & Deployment:** Designed the solution to run on standard devices with potential deployment on cloud or wearable platforms for real-time patient monitoring.
- **Impact:** Enabled affordable, non-invasive, and accessible Alzheimer's screening, aiming to improve early diagnosis and public awareness in regions with limited healthcare infrastructure.

Satellite Image Remote Sensing — Backend Developer and Deep Learning

- **Backend Development:** Designed and developed the backend infrastructure using Django to manage and process large-scale satellite image data. Built scalable RESTful APIs to handle image upload, preprocessing requests, and retrieval of detection results, ensuring robust and low-latency communication between the frontend and backend layers.
- **Data Pipeline Design:** Implemented an efficient data ingestion and preprocessing pipeline using Python for cleaning, normalizing, and resizing satellite images before feeding into the model, ensuring consistent input quality across all stages.
- **AI Model Integration:** Integrated a Region-based Convolutional Neural Network (R-CNN) model in TensorFlow for object detection tasks, effectively identifying geographic features such as houses, rivers, and land parcels with bounding boxes, achieving 84% detection accuracy.
- **Model Optimization:** Tuned model parameters and adjusted anchor box configurations to improve spatial localization performance; benchmarked against YOLO, and selected R-CNN due to superior performance on small object detection in satellite data.
- **Deployment and Testing:** Deployed the complete system on a local server and tested end-to-end functionality across multiple image datasets, validating robustness in varying lighting and resolution conditions.
- **Impact:** Enabled automated remote sensing analysis for environmental monitoring and land use classification, reducing manual annotation time by over 85% and paving the way for scalable geospatial data analysis solutions.

Database Project — Database Engineering

- **Database Design:** Designed and implemented a comprehensive relational database management system (RDBMS) tailored for production planning and inventory control, supporting critical operations such as resource allocation, procurement tracking, and workflow scheduling.
- **Schema Optimization:** Developed an efficient and normalized schema (3NF) to eliminate redundancy and ensure data integrity. Created entity-relationship diagrams (ERDs) for structured design and maintainability across supply chain operations.
- **Query Optimization:** Analyzed query execution plans and restructured complex joins and aggregations using indexing, subqueries, and stored procedures to reduce response time by over 60% for high-volume inventory queries.
- **Data Consistency:** Enforced data consistency with the use of primary/foreign keys, triggers, and constraints, minimizing inventory mismatch and reducing error rates in production planning reports.
- **User Access & Security:** Implemented role-based access control (RBAC) to manage permissions for production staff, supply chain managers, and administrative users, ensuring secure and controlled data usage.
- **Impact:** Improved visibility into supply chain operations and enabled real-time tracking of production and inventory, facilitating proactive decision-making and reducing overstock/understock scenarios by 40%.

Farmer's Agriculture Management System — Backend Development — Python, Django, REST API

- **Objective:** Built a digital platform to support Indian farmers by providing real-time agricultural recommendations, profit analysis, and a direct marketplace for selling crops, aimed at boosting productivity and income while eliminating middlemen.
- **API Integration:** Integrated real-time weather APIs to dynamically recommend optimal crops based on local climate, temperature, humidity, and seasonal patterns—enhancing decision-making and reducing risk of crop failure.
- **Decision Support System:** Developed backend logic for profit analysis, factoring in input costs, historical yields, market trends, and irrigation requirements. Created personalized farming roadmaps based on user location, crop type, and soil health.
- **E-Commerce Module:** Designed a scalable cart and checkout system for purchasing seeds, fertilizers, and equipment. Enabled farmers to directly list and sell harvested crops on the platform, bypassing intermediaries to maximize returns.
- **Security & Access Control:** Implemented JWT-based authentication and role-based access control to safeguard sensitive farmer data and maintain platform integrity.
- **Scalability:** Designed backend services with modular architecture and reusable endpoints, supporting future integration with mobile apps and third-party government APIs (e.g., subsidy programs, mandi prices).
- **Impact:** Digitized farming operations for rural users, reduced decision overhead, improved supply chain transparency, and empowered over 500 farmers with data-driven crop planning and direct-to-consumer sales opportunities.

CERTIFICATIONS

- Getting Started with Accelerated Computing in CUDA C/C++ — **NVIDIA**
- Getting Started with Jetson Nano — **NVIDIA**
- Scaling Workloads Across Multiple GPUs with CUDA C — **NVIDIA**
- Getting Started with Deep Learning — **NVIDIA**

SKILLS

Programming Languages: Python, C, C++, R
Libraries & Frameworks: TensorFlow, PyTorch, Scikit-learn, OpenCV, NLTK, HuggingFace Transformers, NumPy, Pandas
Data Visualization: Matplotlib, Seaborn, Power BI
Databases & Query Languages: MySQL, SQLite, SQL
Development Tools: Git, GitHub, Jupyter Notebook, Visual Studio Code, Anaconda
Operating Systems: Linux (Ubuntu), Windows, macOS
Soft Skills: Team Collaboration, Leadership, Communication, Content Writing, Problem-Solving, Creativity, Adaptability