PHASE – 1

RECOGNIZING HANDWRITTEN DIGITS WITH DEEP LEARNING FOR SMARTER AI APPLICATIONS

STUDENT NAME : S. JOTHIGA

REGISTER NUMBER : 422623104018

INSTITUTION : UNIVERSITY COLLEG OF ENGINEERING,PANRUTI

DEPARTMENT : COMPUTER SCIENCE AND ENGINEERING

DATE OF SUBMISSION : 29.04.2025

Problem Statement :

In an increasingly digital world, enabling machines to understand handwritten input is vital for applications such as digital forms, automated postal systems, and note-taking applications. Despite the widespread use of printed text, handwritten digits are still common in forms, invoices, and classroom work. Manually processing such data is time-consuming and error-prone. This project aims to solve the problem of recognizing handwritten digits using deep learning techniques, which can

significantly improve the efficiency and accuracy of data entry and verification processes.

🎯 Objectives of the Project :

- Develop a deep learning model capable of accurately recognizing handwritten digits.
- Train the model on a publicly available dataset and evaluate its performance.
- Optimize the model for accuracy and performance.
- Visualize the results and decision-making process of the model.
- Optionally, deploy the model as an interactive web app or notebook-based tool for user testing.

🧭 Scope of the Project :

Features to be built/analyzed:

- Image preprocessing pipeline for handwritten digit images.
- Convolutional Neural Network (CNN) architecture for digit recognition.
- Training, validation, and testing workflow with performance metrics.
- Visualization of misclassified digits and model insights.

Limitations and Constraints:

- Project will use the MNIST dataset only.

- Model will be trained and tested within a notebook environment (e.g., Google Colab).
- Deployment (if implemented) will be via basic tools like Streamlit or Gradio.
- Focused only on digits (0–9), not alphabets or symbols.

Data Sources :

Dataset: MNIST Handwritten Digits Dataset
Source : Public dataset from Kaggle or from tensorflow.keras.datasets.
Type : Public and static dataset.
Format : 28x28 grayscale images, labeled 0–9.

High-Level Methodology :

- **Data Collection:** Dataset will be downloaded from Keras datasets or Kaggle.
- **Data Cleaning:** Normalize pixel values, convert labels to categorical form, check for shape consistency.
- **Exploratory Data Analysis (EDA):** Visualize sample digits, class distribution, and pixel intensity histograms.

- **Feature Engineering:** Normalize images, reshape for CNN input, one-hot encoding of labels.
- **Model Building:** Use CNN (e.g., Conv2D, MaxPooling2D, Dense layers); test different architectures.
- **Model Evaluation:** Use accuracy, confusion matrix, and validation loss/accuracy plots.
- **Visualization & Interpretation:** Visualize filters, activation maps, confusion matrix, misclassifications.
- **Deployment (Optional):** Build a web interface using Streamlit or Gradio for real-time predictions.

Tools and Technologies :

- **Programming Language:** Python
- **Notebook/IDE:** Google Colab or Jupyter Notebook
- **Libraries:**
    - Data Processing: numpy, pandas
    - Visualization: matplotlib, seaborn
    - Modeling: tensorflow, keras, scikit-learn
- **Optional Tools for Deployment:** Streamlit, Gradio.

Team Members and Roles :

Name	Role and Responsibilities
S.SANGDEENA	**Project Lead** – Oversees the entire project, coordinates tasks, and works on model training and optimization.
P.VAISHNAVI	**Data Engineer** – Responsible for data loading, preprocessing, and handling data cleaning and augmentation.
S.JOTHIGA	**Exploratory Data Analyst** – Performs EDA, visualizes patterns, class distribution, and contributes to insights.

Name	Role and Responsibilities
R.PRADHISH	**Model Architect** – Designs the CNN model architecture, experiments with hyperparameters, and evaluates models.
T.AGILESH	**Deployment & Documentation Specialist** – Builds the deployment interface (Streamlit/Gradio) and prepares the final report and presentations.