

# AI-Powered Image Classification using Deep Learning

Submitted by: Uday Kiran Manthana

Assignment: Artificial Intelligence Image Classification System

## 1. Objective

The objective of this project is to design, train, and evaluate a Convolutional Neural Network (CNN) model capable of classifying images into multiple categories using deep learning techniques. The model aims to recognize handwritten digits (0-9) from the MNIST dataset and achieve high accuracy through proper data preprocessing, model optimization, and performance evaluation.

## 2. Dataset Description

Dataset Used: MNIST (Modified National Institute of Standards and Technology)

Dataset Type: Grayscale handwritten digits (0-9)

Number of Images: 70,000 (60,000 for training, 10,000 for testing)

Image Dimensions:  $28 \times 28$  pixels

Classes: 10 categories (digits 0-9)

Train/Validation/Test Split: 70% / 15% / 15%

## 3. Data Preprocessing

- Converted all images to grayscale.
- Normalized pixel values between 0-1.
- Applied inversion so that digits appear white on black background.
- Used centering algorithm to align digits.
- Implemented data augmentation (rotation, shifting, zooming).

## 4. Model Architecture

Framework: TensorFlow/Keras

Layers: 3×Conv2D, BatchNormalization, MaxPooling, Dropout, Dense

Activation: ReLU, Softmax (output)

Optimizer: Adam (learning rate =  $1e-3$ )

Loss Function: Categorical Crossentropy

Callbacks: EarlyStopping, ReduceLROnPlateau, ModelCheckpoint

## 5. Model Training

Epochs: 20

Batch Size: 64

Data Generator: ImageDataGenerator with small rotation and zoom

Early Stopping and Learning Rate Scheduler implemented to prevent overfitting.

## 6. Evaluation Metrics

Training Accuracy: 99.2%

Validation Accuracy: 99.3%

Test Accuracy: 99.4%

Precision / Recall / F1-Score: ~0.99 average

Loss (Test): 0.02

## 7. Visualization

Generated training vs validation accuracy/loss graph and confusion matrix. Both are stored in the outputs/ folder as training\_history.png and confusion\_matrix.png.

## 8. Streamlit Web App

A Streamlit-based web app was developed for real-time prediction. Users can upload images, which are preprocessed and classified instantly. Uploaded images and predictions are stored in the data/ folder.

## 9. Key Challenges & Solutions

1. Incorrect predictions for real-world images → Fixed by inversion and centering preprocessing.
2. Blank image crashes → Implemented safety checks.
3. Model misclassification → Improved data augmentation and added Dropout layers.

## 10. Results Summary

The CNN achieved 99.4% test accuracy with strong generalisation across all digit classes. The model performs well even on handwritten inputs through the Streamlit interface.

## 11. Conclusion

The project successfully implemented a deep learning-based image classification system capable of recognising handwritten digits. With 99%+ accuracy and an interactive web app, this project demonstrates a complete AI development lifecycle — from data preprocessing to deployment.

## 12. Future Scope

- Extend the model to Fashion-MNIST or CIFAR-10 for general image classification.
- Deploy the Streamlit app on Hugging Face Spaces or Render.
- Experiment with Transfer Learning (MobileNetV2 / ResNet50).

## 13. References

1. TensorFlow / Keras Documentation
2. MNIST Dataset (LeCun et al.)
3. Streamlit Documentation
4. Scikit-learn Metrics Library