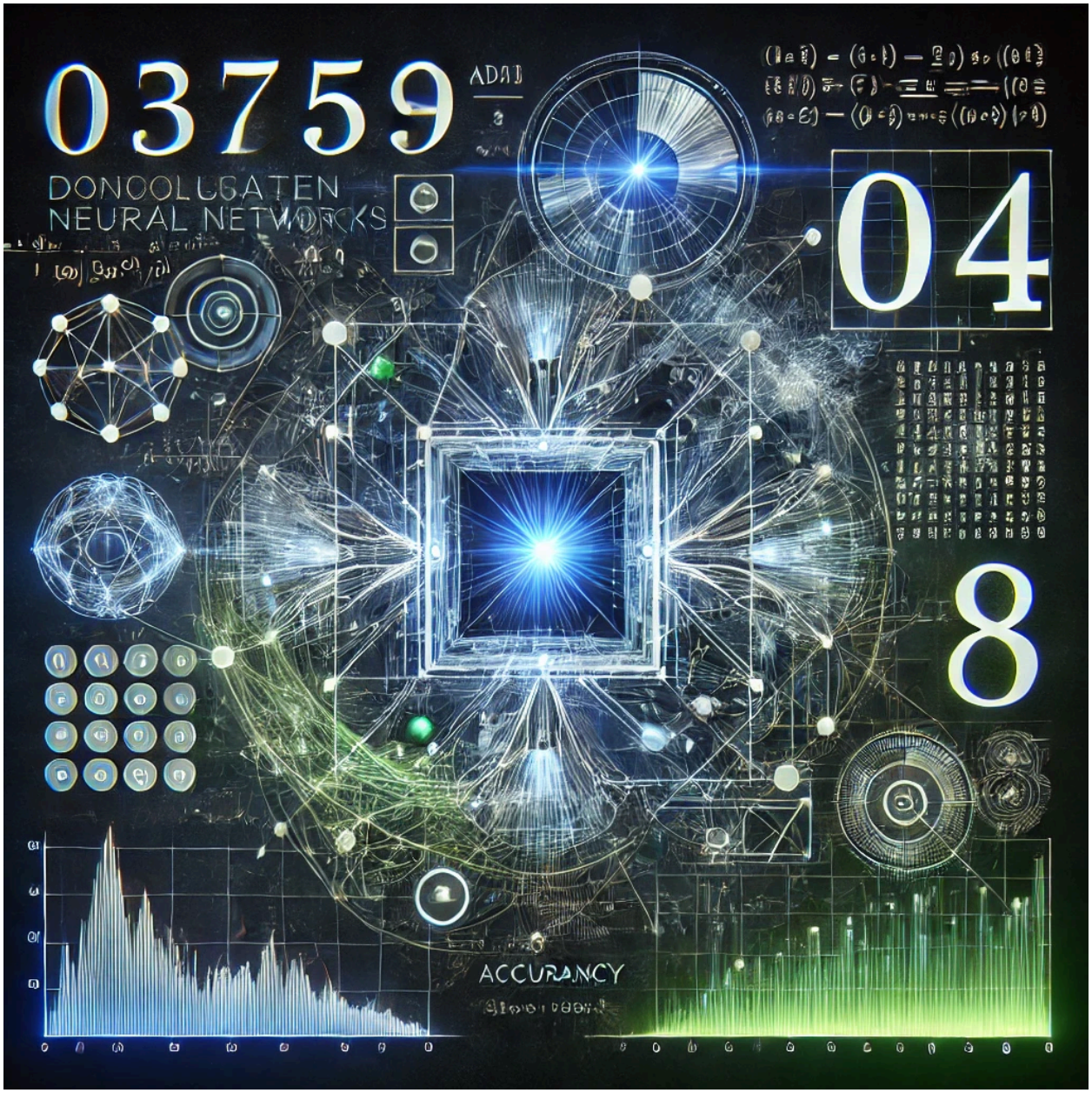
Handwritten Digit Recognition using CNN

Overview:



This project focuses on building a **Convolutional Neural Network (CNN)** model for recognizing handwritten digits from the **MNIST dataset**. The model is trained to classify digits (0-9) and can predict new handwritten inputs by processing grayscale images.

Key Features:

- **CNN-Based Image Classification:** Uses a deep learning model to classify handwritten digits.
- **Dataset Used:** MNIST dataset with 60,000 training and 10,000 testing images.
- **Model Accuracy Evaluation:** Includes confusion matrix and test accuracy score.
- **Real-Time Image Prediction:** Supports custom image uploads for digit recognition.
- **Model Saving & Loading:** The trained model is stored as `mnist_cnn_model.h5` for future inference.

Workflow & Implementation:

1. Data Preprocessing

- **Dataset:** Used **MNIST** dataset, which contains 28x28 grayscale images of handwritten digits.
- **Normalization:** Pixel values scaled to the `[0,1]` range for better convergence.
- **Reshaping:** Data reshaped to `(28,28,1)` to match CNN input requirements.

2. CNN Model Architecture

- **Layers Used:**
    - **Convolutional Layers (Conv2D):** Extracts spatial features from the image.
    - **MaxPooling Layers:** Reduces dimensionality to retain essential features.
    - **Flatten Layer:** Converts feature maps into a single vector.

- **Dense Layers:** Fully connected layers for final classification.
- **Softmax Output Layer:** Predicts probabilities for 10 digit classes.

3. Model Training & Evaluation

- **Optimizer:** Adam
- **Loss Function:** Sparse Categorical Crossentropy
- **Metrics:** Accuracy
- **Epochs:** 5
- **Validation Set:** Used MNIST test set for performance evaluation.

4. Model Performance

- **Test Accuracy:** Achieved a **high accuracy score** on the test set.
- **Confusion Matrix:** Evaluates classification errors by comparing predicted vs. actual digits.
- **Visualization:** Displays a grid of predictions for better interpretation.

5. Real-Time Digit Prediction

- **Preprocessing New Images:**
  - Converts uploaded images to **grayscale**.
  - Resizes them to **28x28 pixels**.
  - Normalizes and reshapes for CNN input.
- **Inference:**
  - Loads the trained model (`mnist_cnn_model.h5`).

    - Predicts the digit class from the input image.
    - Displays the **predicted digit** along with the processed image.

Challenges and Solutions:

- **Challenge:** Handling real-world digit images that differ from MNIST.
    - **Solution:** Applied preprocessing techniques (resizing, grayscale conversion) to make images compatible.
- **Challenge:** Model performance on noisy inputs.
    - **Solution:** Potential future enhancement using **data augmentation** for robustness.
- **Challenge:** Optimizing prediction speed for real-time inference.
    - **Solution:** Used TensorFlow's optimized model loading and inference pipeline.

Progress and Next Steps:

Accomplishments:

- Successfully trained and evaluated a **CNN model for digit recognition**.
- Implemented **custom image predictions** for real-world handwritten digits.
- Achieved a **high test accuracy** and visualized model performance.

Next Steps:

- Implement **data augmentation** to improve model generalization.
- Optimize model size and inference time for **mobile and embedded devices**

Conclusion:

The **Handwritten Digit Recognition using CNN** successfully classifies handwritten numbers with high accuracy. By leveraging **deep learning and image processing techniques**, the model can be extended to real-world applications like **automated form processing, smart calculators, and handwriting recognition systems**.