Dataset Digit Recognization Using CNN For Enhanced Accuracy On MNIST

Sarthak Sahu, Prof.Vikram Rajput
Madhav Institute of Technology and Science, Gwalior (M.P.)

***Abstract*—This research delves into the implementation of Convolutional Neural Networks (CNNs) for the task of handwritten digit recognition using the MNIST dataset. The objective was to design and optimize a CNN-based model to surpass the performance of traditional machine learning methods and earlier CNN architectures. The final deep learning model, constructed using TensorFlow and Keras, achieved an accuracy exceeding 99% on the test dataset. The study demonstrates the importance of architectural design choices, including convolutional layers, pooling layers, batch normalization, dropout, and ReLU activation functions in enhancing model generalization and reducing overfitting. The results validate the efficacy of the enhanced CNN and suggest its applicability in real-world digit recognition systems.**

***Keywords*—Convolutional Neural Network, MNIST, Handwritten Digit Recognition, Deep Learning, Image Classification**

I. INTRODUCTION

Handwritten digit recognition is an essential task in image processing and computer vision. Applications such as postal code recognition, bank cheque processing, and automated form reading rely heavily on precise classification of numeric digits. In this context, machine learning and deep learning models have emerged as powerful tools for building intelligent systems that can emulate human-level accuracy in visual tasks.

The MNIST dataset, comprising 70,000 images of handwritten digits, serves as a fundamental benchmark for evaluating classification models. While classical machine learning techniques including Support Vector Machines (SVM), k-Nearest Neighbors (k-NN), and Multilayer Perceptrons (MLP) have performed reasonably well on this dataset, they often fall short of capturing intricate spatial hierarchies within images.

CNNs overcome these limitations by learning local patterns and hierarchical features automatically. This research aims to build an enhanced CNN model by experimenting with architectural configurations that lead to improved accuracy and generalization. By integrating techniques such as batch normalization and dropout, the final model demonstrated superior performance compared to its predecessors.

II. LITERATURE SURVEY

The evolution of digit recognition has undergone significant transformations. Early approaches relied on hand-crafted features and statistical classifiers. LeNet-5, a pioneering CNN developed by Yann LeCun in the late 1990s, showcased the power of local receptive fields and parameter sharing in image classification.

Subsequent advancements introduced deeper networks such as AlexNet, VGGNet, and ResNet, which improved feature extraction through stacked convolutional layers. Despite their success, these models are often too complex for relatively simple datasets like MNIST. Hence, lightweight yet effective architectures are more suitable.
Batch normalization, introduced by Ioffe and Szegedy, was a milestone that enabled faster and more stable training. Similarly, dropout, proposed by Hinton et al., offered a regularization technique that prevents overfitting. Research also emphasized the benefits of small filters, deeper stacks, and efficient activation functions in enhancing CNN capabilities.

Recent studies have applied ensemble methods, data augmentation, and hybrid CNN-RNN models to further boost performance. However, the trade-off between
complexity and accuracy remains a critical consideration, especially for real-time and embedded applications.

III. METHODOLOGY

3.1 Dataset Overview The MNIST dataset includes 60,000 training images and 10,000 testing images. Each image is a 28x28 grayscale representation of a digit from 0 to 9. The simplicity of the dataset allows for rapid prototyping, while still offering enough complexity for meaningful evaluation.

3.2 Data Preprocessing Preprocessing is a crucial step in any machine learning pipeline. For this project, the images were:

- Reshaped to (28, 28, 1) to include a single color channel
- Normalized to scale pixel values from [0–255] to [0–1]
- Randomly split into training and validation sets to monitor performance

Baseline CNN Architecture The baseline model included the following layers:

- Conv2D with 32 filters, kernel size (3x3), activation ReLU
- MaxPooling2D with pool size (2x2)
- Conv2D with 64 filters, kernel size (3x3), activation ReLU
- MaxPooling2D
- Conv2D with 64 filters, kernel size (3x3), activation
- Flatten layer to convert feature maps into a vector
- Dense layer with 64 units, ReLU
- Output Dense layer with 10 units and softmax activation This model achieved 98.9% test accuracy.

3.3 Enhanced CNN Architecture To improve upon the baseline, the following enhancements were made:

- Added a Conv2D layer with 128 filters for deeper feature extraction
- BatchNormalization layers added after Conv2D layers to stabilize learning
- Dropout layers introduced to mitigate overfitting
- Increased Dense layer units from 64 to 128 for better representation Architecture Details:
- Conv2D (32) → BatchNormalization → ReLU
- MaxPooling2D
- Conv2D (64) → BatchNormalization → ReLU
- MaxPooling2D
- Conv2D (64) → ReLU
- Conv2D (128) → ReLU
- MaxPooling2D
- Flatten
- Dense (128) → ReLU → Dropout (0.5)
- Dense (10) → Softmax

3.4 Training Configuration The model was compiled using:

- Loss: Sparse Categorical Crossentropy
- Optimizer: Adam (learning rate: 0.001)
- Batch Size: 128
- Epochs: 10
- EarlyStopping and ModelCheckpoint callbacks

IV.RESULTS AND DISCUSSION

4.1 Performance Metrics The enhanced model yielded the following:

- Training Accuracy: 99.80%
- Validation Accuracy: 99.45%
- Test Accuracy: 99.42%

This shows consistent performance across datasets, indicating strong generalization. In Figure 1.

Figure 1

4.2 Confusion Matrix Analysis A confusion matrix revealed that most digits were classified accurately. The misclassifications were minimal and typically occurred between visually similar digits such as 4 and 9 or 3 and 5.

4.3 Learning Curves The training and validation loss curves showed smooth convergence, with no significant signs of overfitting. The accuracy curves reflected a steady increase in performance, validating the impact of model enhancements.

4.4 Comparative Analysis

Model	Test Accuracy
k-Nearest Neighbors	97.30%
SupportVector Machine	98.60%
Multilayer Perceptron	98.00%
Initial CNN	98.90%
Enhanced CNN	99.42%

The enhanced CNN model demonstrates clear superiority over classical approaches. Accuracy shown in figure no.2

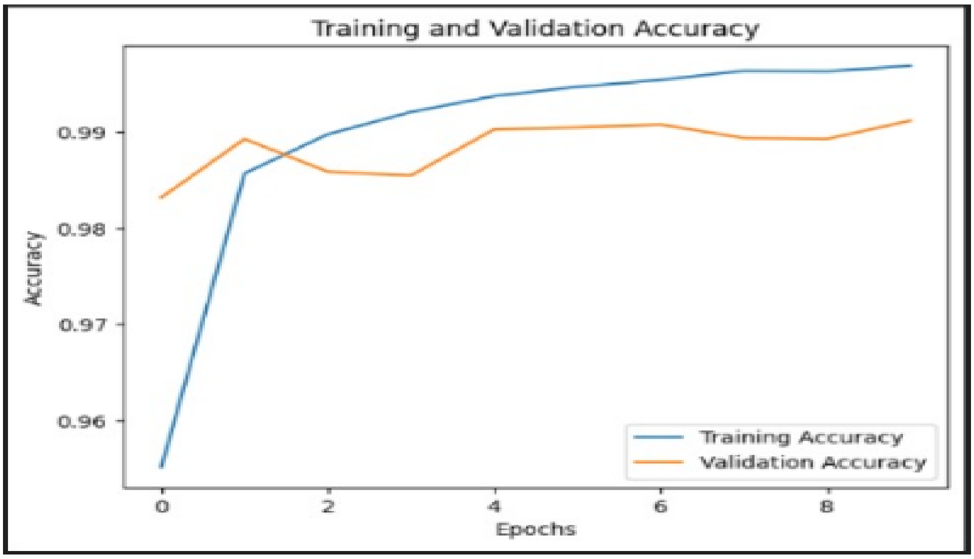
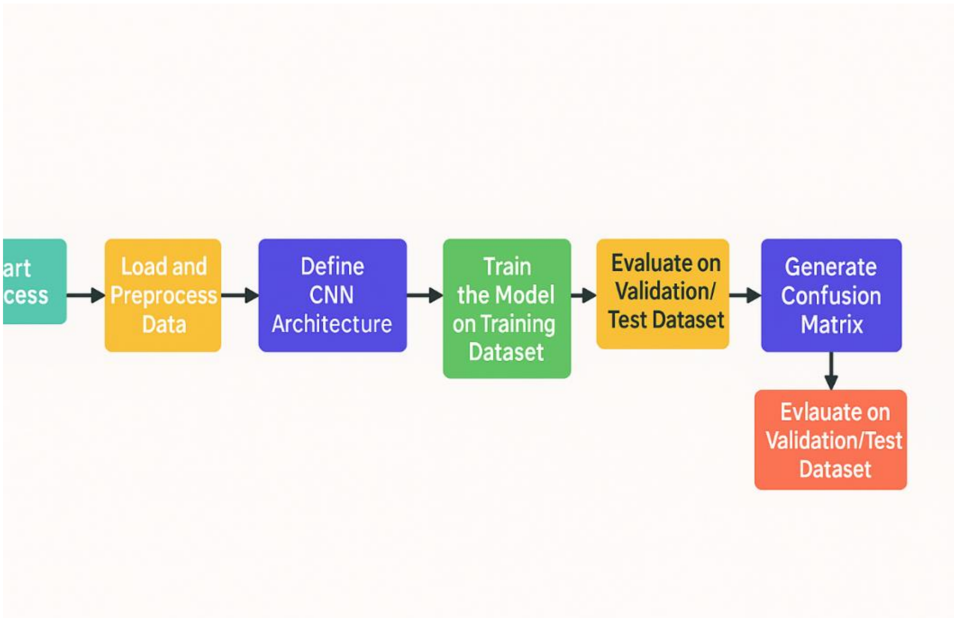
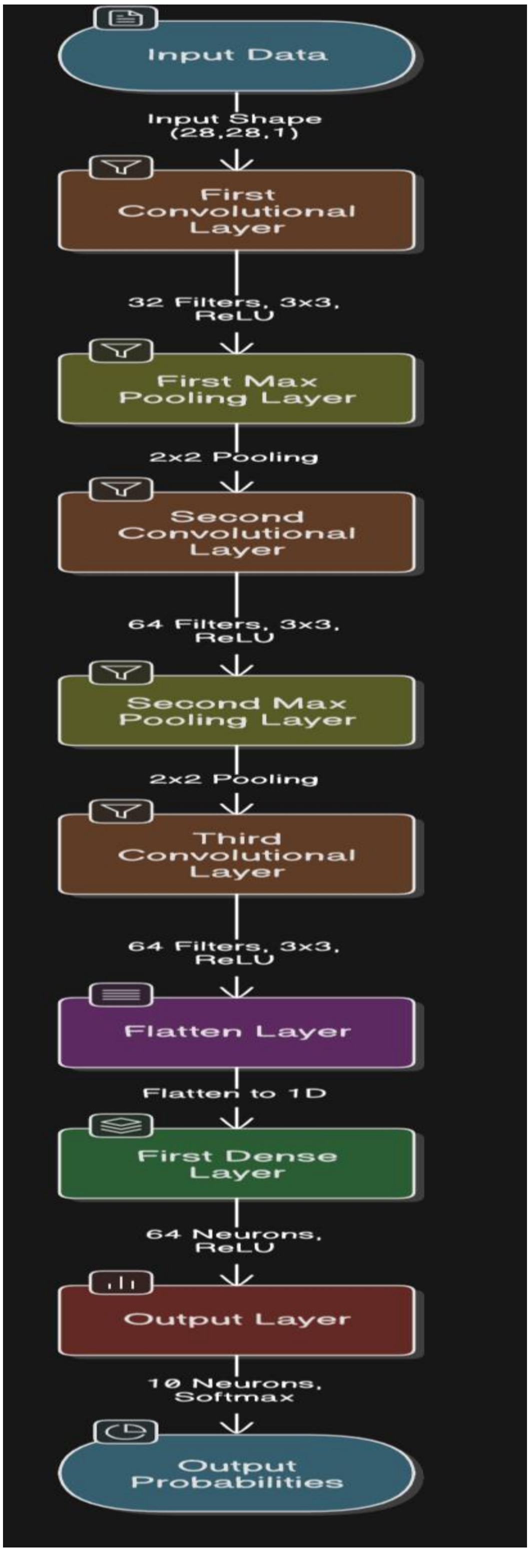


Figure 2

Flowchart of Proposed meth



Flowchart of CNN



V. CONCLUSION

This study effectively illustrates the strength of Convolutional Neural Networks in the domain of handwritten digit recognition. Starting from a basic CNN model, various architectural improvements were systematically integrated to enhance performance. The final model reached a remarkable 99.42% accuracy on the MNIST test dataset.

By leveraging advanced components like batch normalization, dropout layers, and additional convolutional blocks, the model not only achieved higher accuracy but also maintained robustness and generalization. These results were significantly better than those of traditional models and the baseline CNN architecture.

The project confirms the importance of architectural experimentation and parameter tuning in CNN development. The proposed model strikes a balance between complexity and accuracy, making it suitable for deployment in practical applications such as optical character recognition and automated form processing.

In the future, extending this approach to datasets with more variation (such as EMNIST or fashion MNIST) and incorporating data augmentation techniques can further improve performance. Real-time inference speed optimization and deployment on edge devices also present valuable research avenues.

REFERENCES

[1] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324.

[2] Simard, P. Y., Steinkraus, D., & Platt, J. C. (2003). Best practices for convolutional neural networks applied to visual document analysis. *ICDAR*.

[3] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *NIPS*, 25.

[4] Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ICML*.

[5] Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*