AIT 736 - Applied Machine Learning (Spring 2025)

Title: Handwritten Digit Recognition using Convolutional Neural Networks (CNN)

Under guidance of Dr. Lei Yang

Project Presentation

Presenters: Group 02

- Saivarun T.R. (G01475545)
- Mohammed Tareq (G01478697)
- Suraj Poldas (G01483726)
- Mano Harsha Sappa (G01459796)

Affiliation: George Mason University

MASON
George Mason University

Date: April 28, 2025

Introduction

- Handwritten digit recognition is a fundamental task in the field of computer vision and deep learning.
- The MNIST dataset serves as a standard benchmark for testing digit recognition algorithms.
- Convolutional Neural Networks (CNNs) have revolutionized pattern recognition by automatically learning features from images.
- Our project aims to build a deep learning model that can efficiently recognize handwritten digits, achieving high accuracy with real-world robustness.
- We also focus on analyzing model performance through confusion matrices, training-validation trends, and hyperparameter tuning.

Problem Statement

Recognizing handwritten digits accurately is challenging due to:

- Variations in handwriting style
- Differences in size, thickness, orientation, and noise

Traditional machine learning models struggle to capture complex patterns in raw pixel data.

A deep learning approach is needed to automatically extract relevant features and ensure high generalization across diverse handwriting samples.

The goal is to design a robust CNN model that achieves high accuracy and can generalize well on unseen handwritten digits.

Dataset Description

Dataset Name: **MNIST Handwritten Digit Database**

Number of Samples:

60,000 training images
10,000 testing images

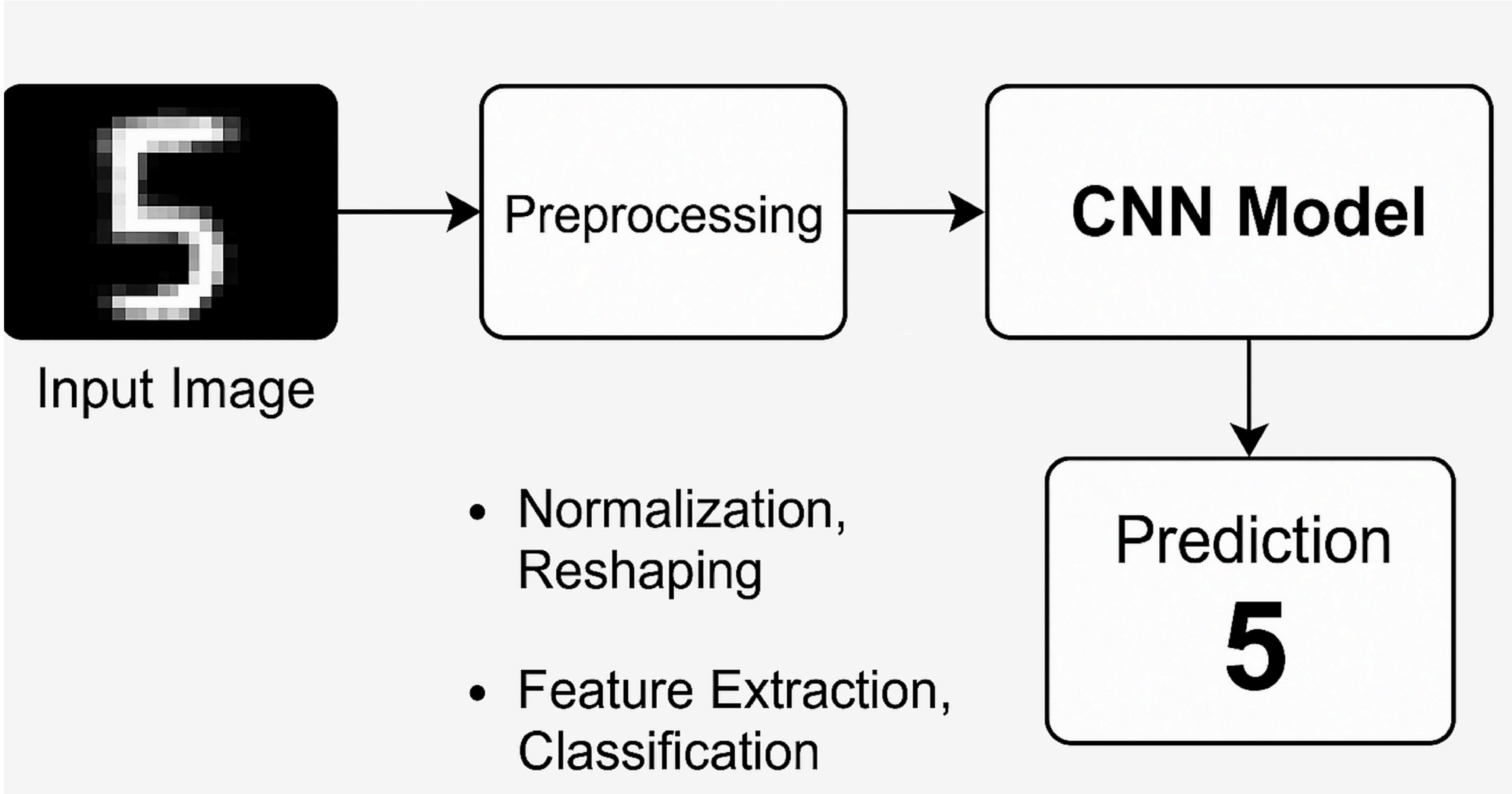
Classes: 10 classes (**digits 0 through 9**)

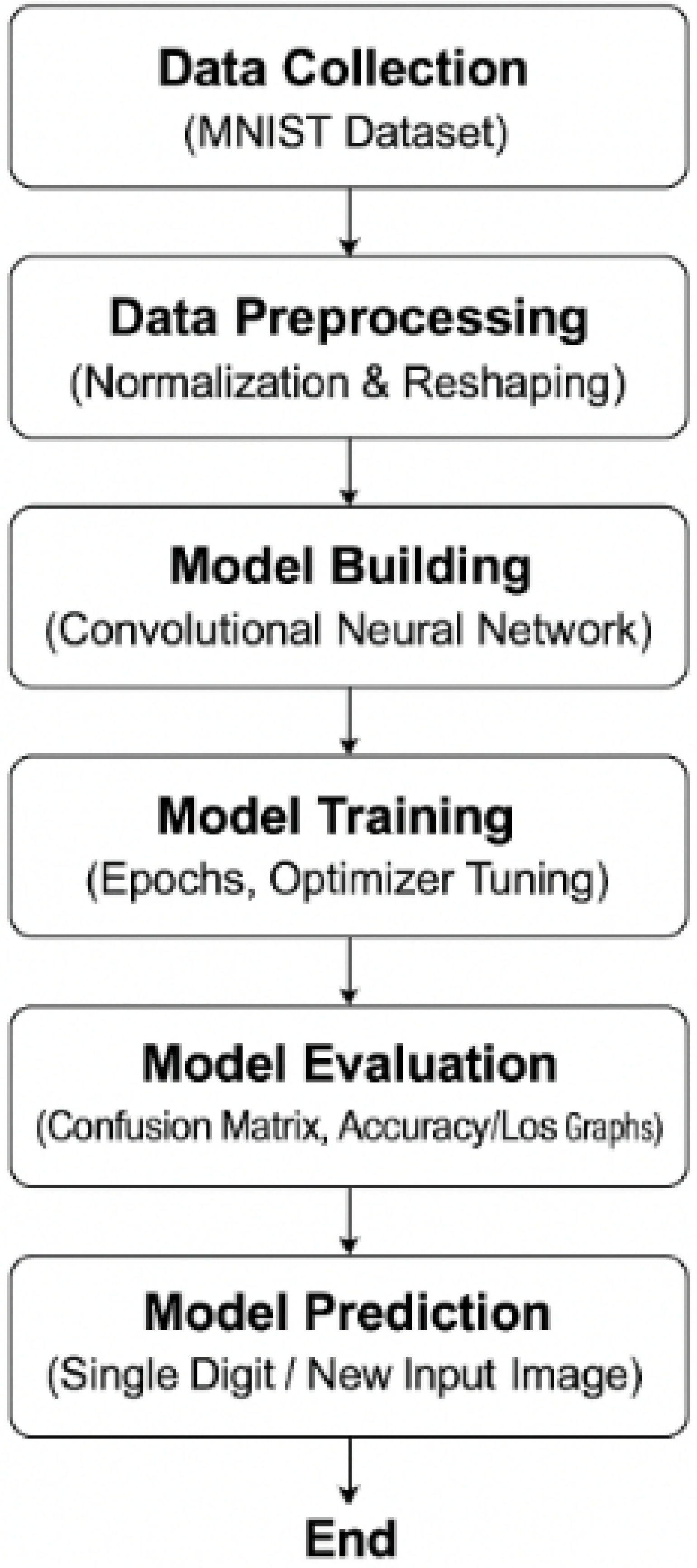
Image Size: 28 × 28 pixels, grayscale
Data Format: Each image is a 28x28 array; pixel values range from 0 to 255.

Challenges Addressed:

- Variations in handwriting style
- Class imbalance (minor differences)
- Real-world noise and distortions

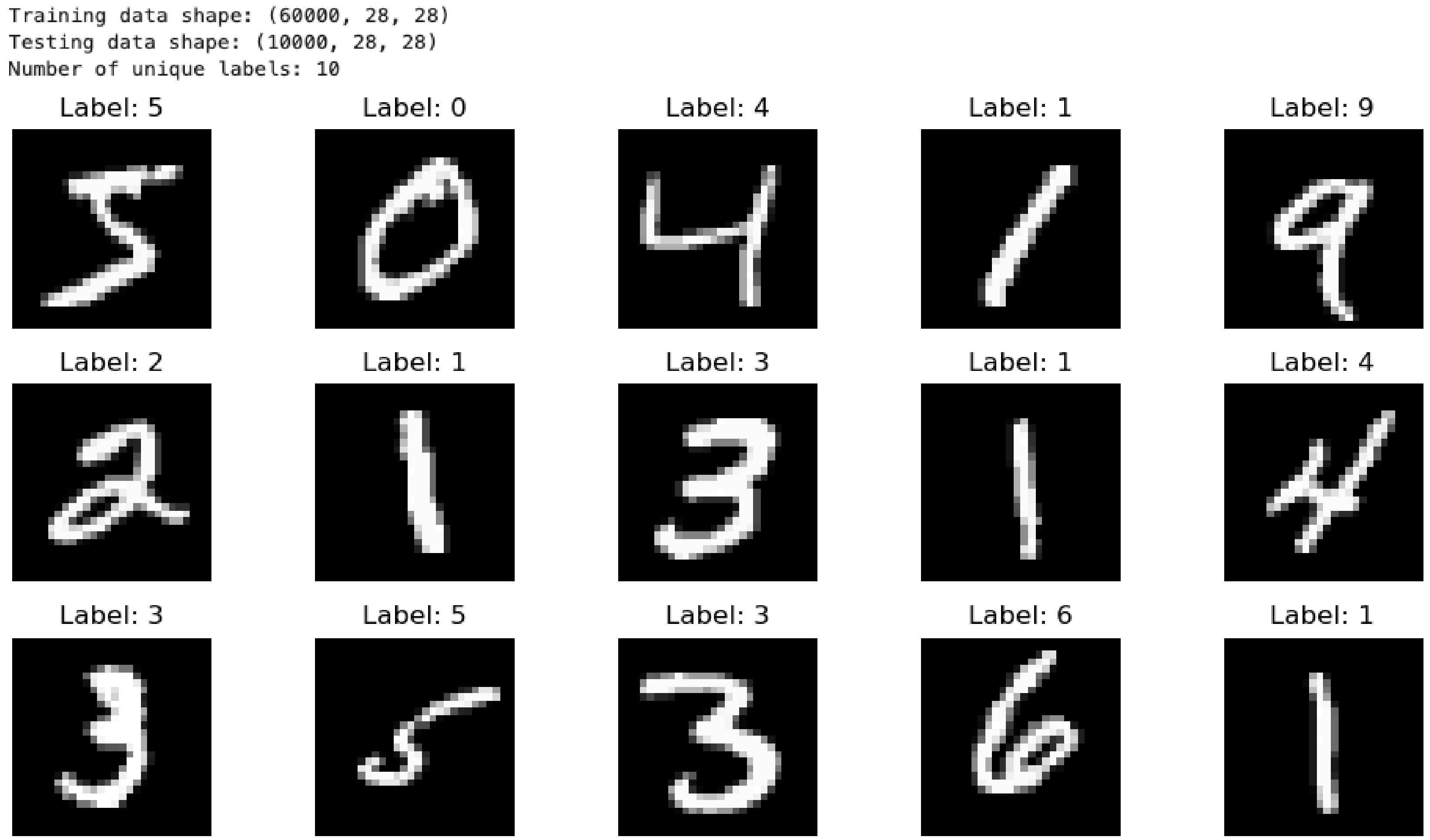
System Overview Diagram



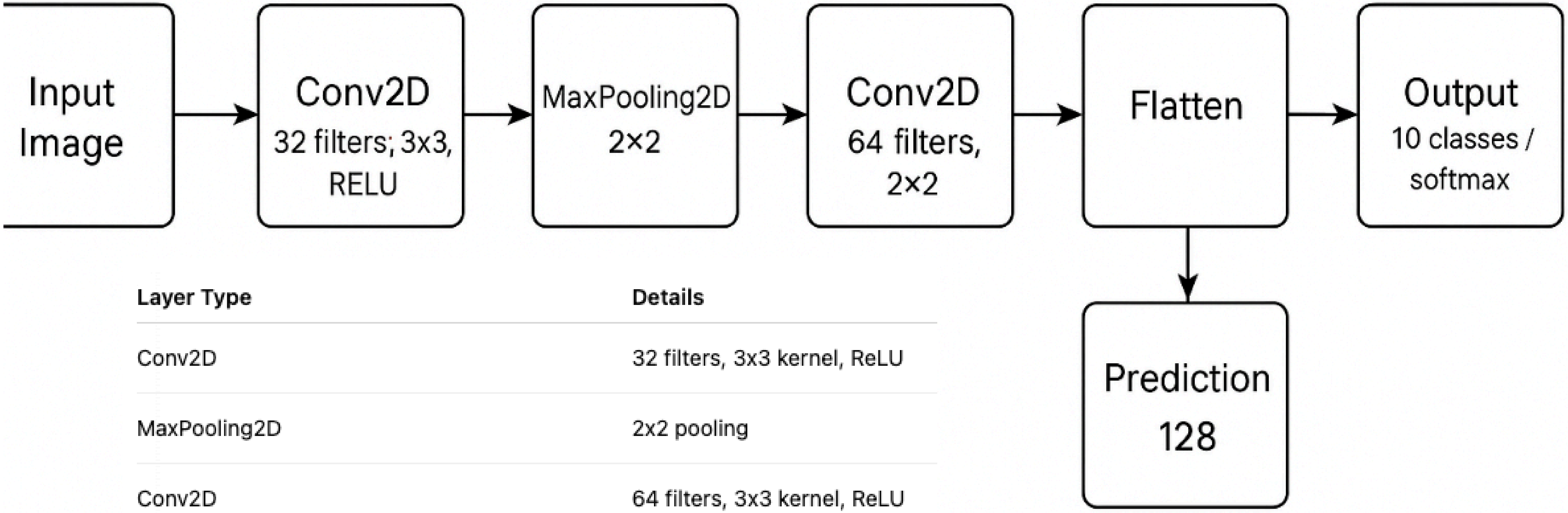


Exploratory Data Analysis (EDA)

- The dataset contains 60,000 training images and 10,000 testing images.
- Each image is 28 × 28 pixels, grayscale.
- Pixel intensity values range from 0 (black) to 255 (white).
- The dataset is balanced across 10 digit classes (0–9).
- Significant variation in writing styles, digit thickness, and orientation.



Model Architecture (CNN)



Layer Type	Details
Conv2D	32 filters, 3x3 kernel, ReLU
MaxPooling2D	2x2 pooling
Conv2D	64 filters, 3x3 kernel, ReLU
MaxPooling2D	2x2 pooling
Flatten	-
Dense	128 units, ReLU
Dense (Output)	10 units, Softmax

Model Training and Hyperparameter Tuning

Training Strategy:

- Dataset split into training and testing sets.
- Loss function: Categorical Crossentropy
- Optimizer: Adam (Adaptive Moment Estimation)
- Evaluation Metric: Accuracy

Hyperparameter Tuning:

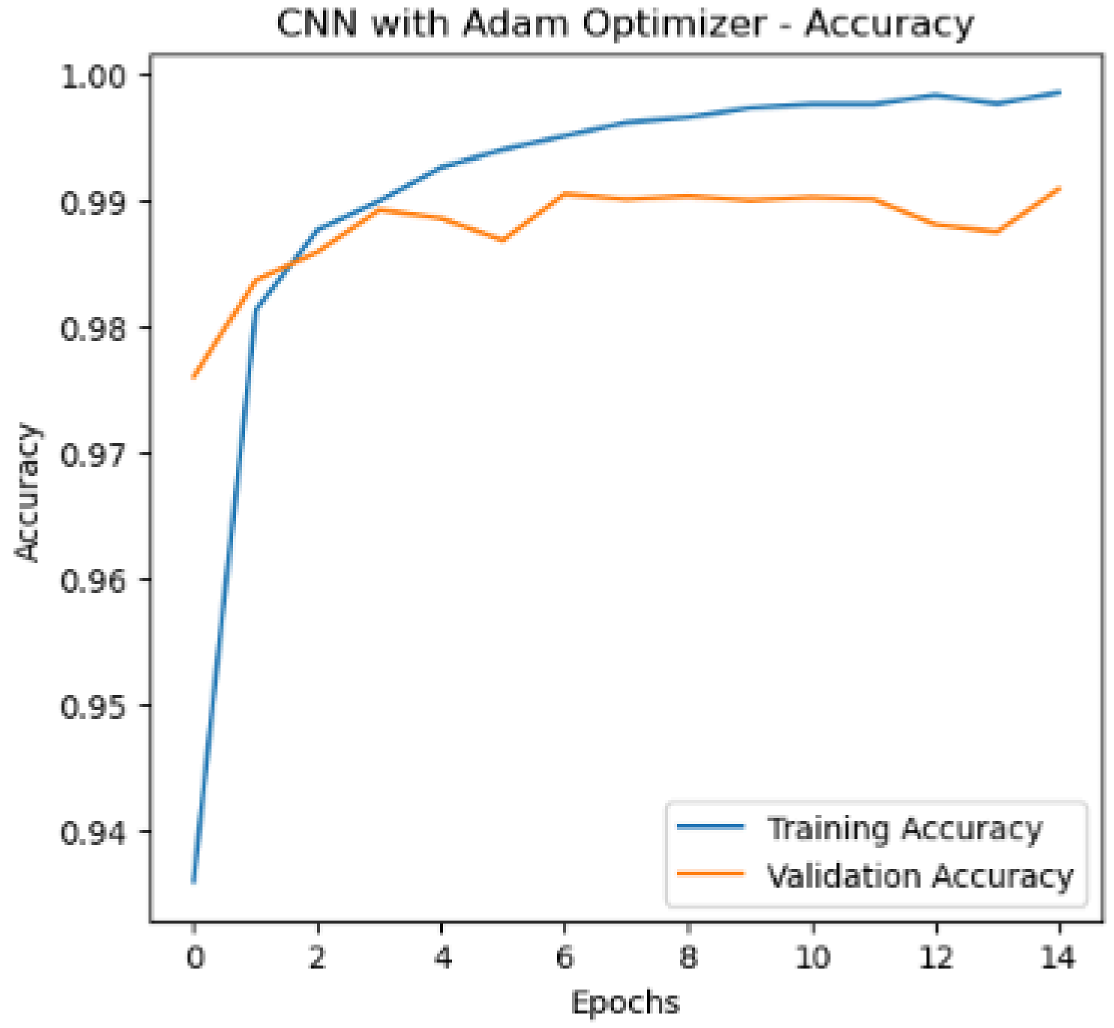
- Batch Size: 32 → balanced speed and performance.
- Epochs: 15 → enough to achieve convergence without overfitting.
- Learning Rate: Default of Adam (0.001).

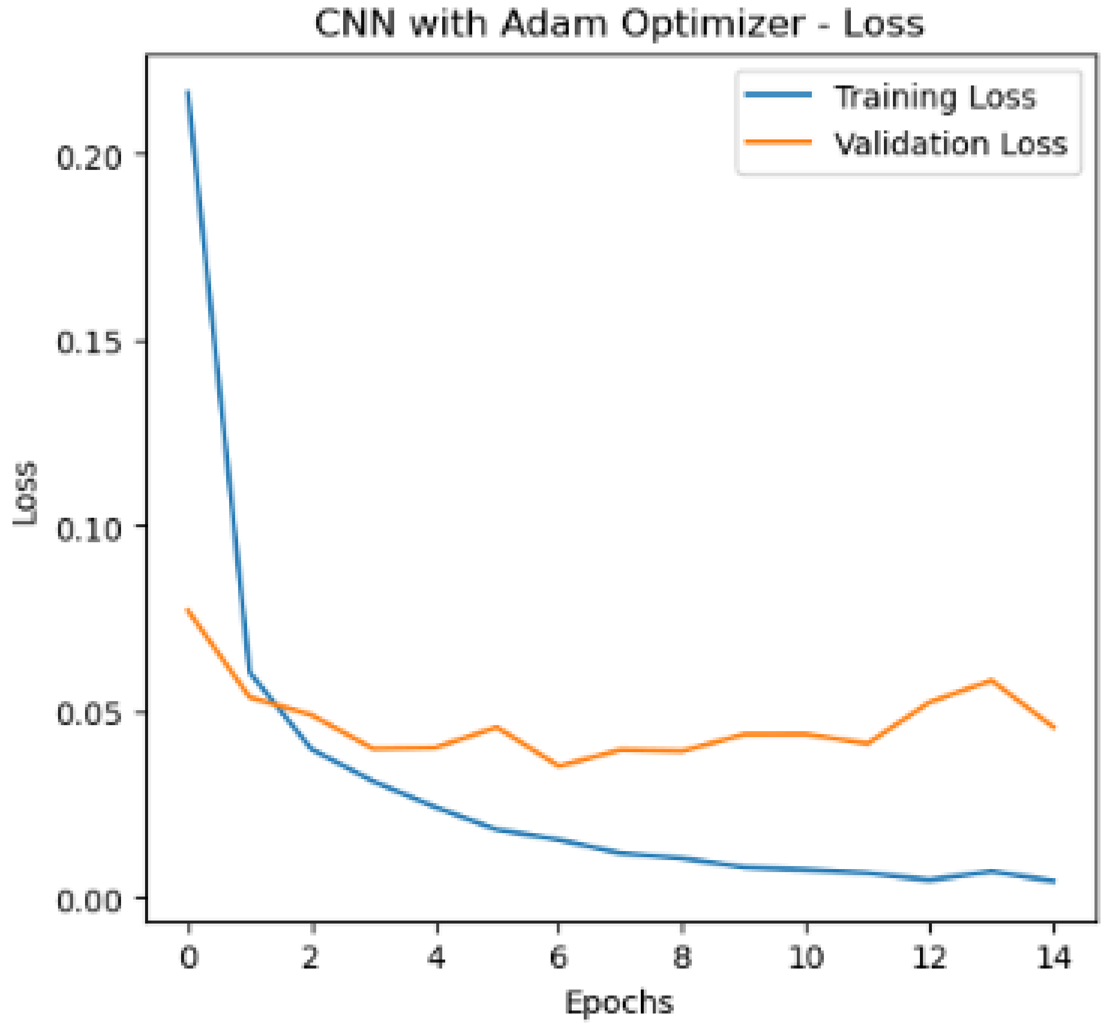
Early Observations:

- Rapid convergence within the first 5 epochs.
- Validation accuracy remained consistently high (~99%).

Model Evaluation and Visualization

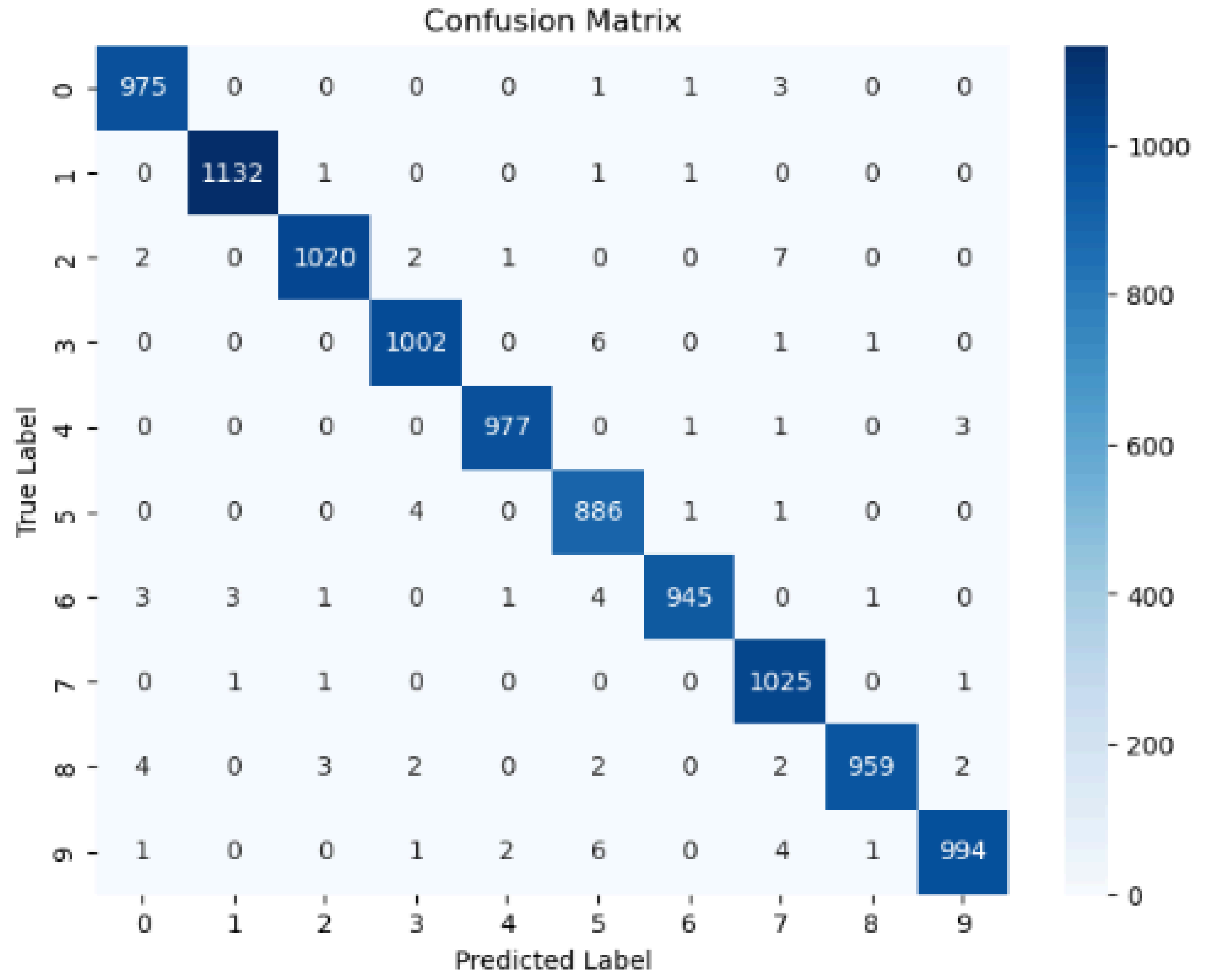
Performance Metrics – CNN Model (Adam Optimizer)





Test Set Accuracy: 99%

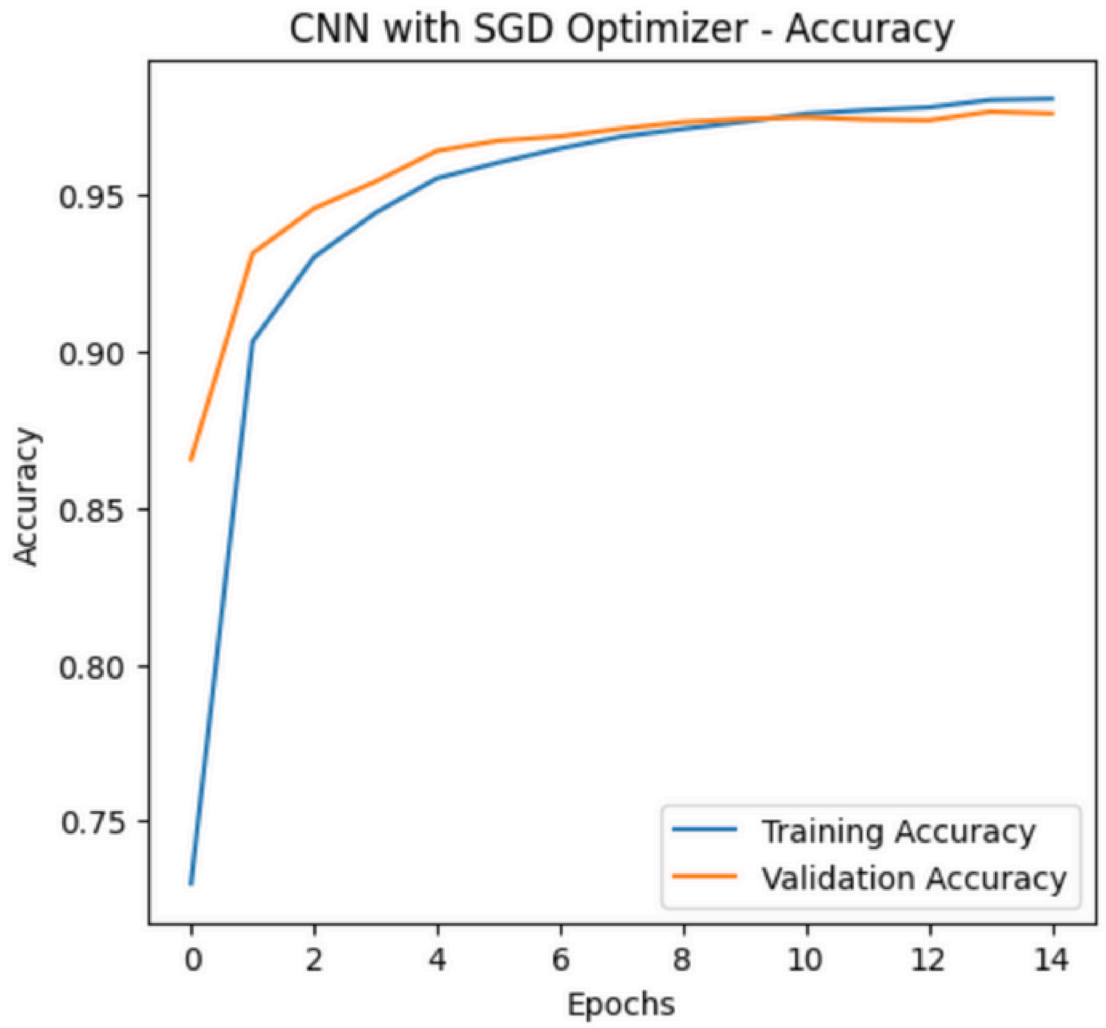
```
313/313 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step
```

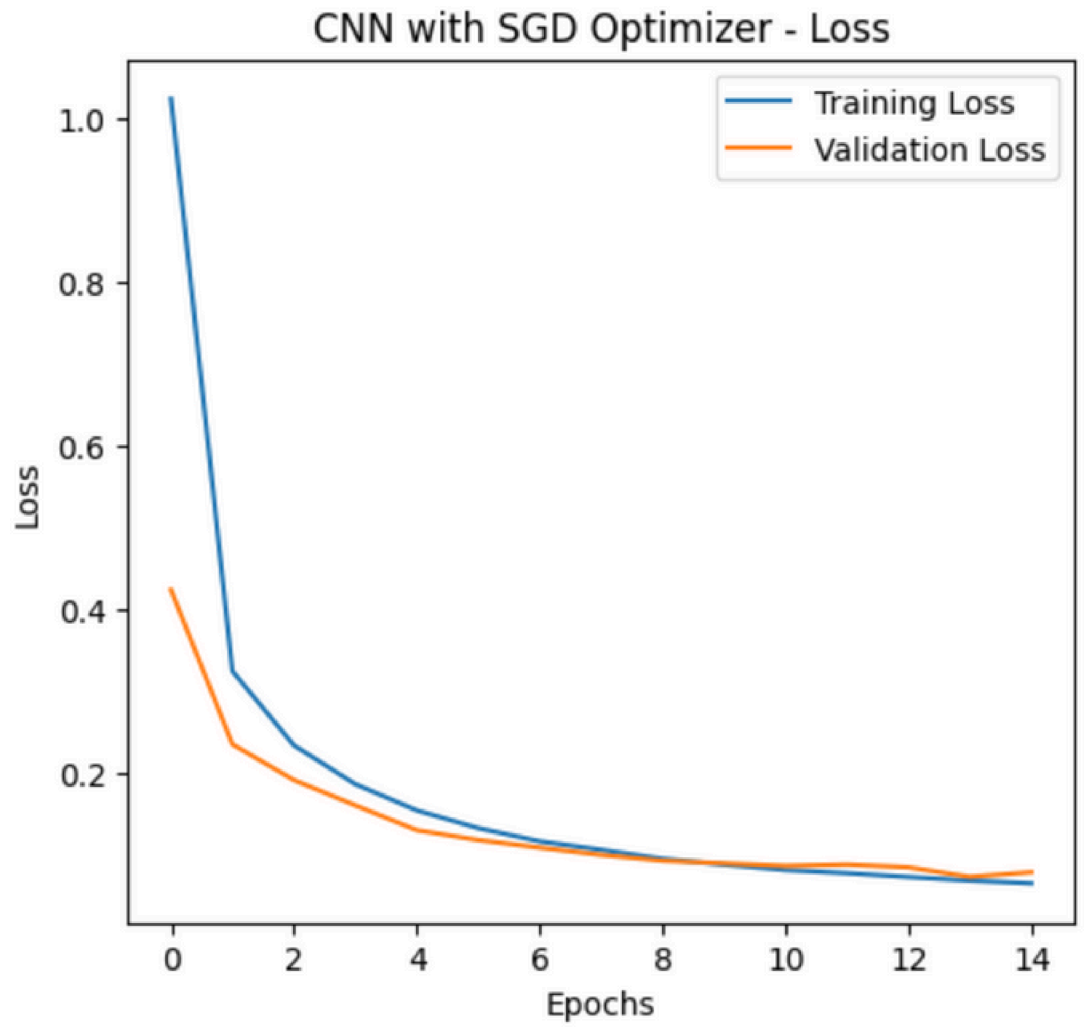



```
Classification Report:
              precision    recall  f1-score   support

           0       0.99      0.99      0.99       980
           1       1.00      1.00      1.00      1135
           2       0.99      0.99      0.99      1032
           3       0.99      0.99      0.99      1010
           4       1.00      0.99      1.00       982
           5       0.98      0.99      0.99       892
           6       1.00      0.99      0.99       958
           7       0.98      1.00      0.99      1028
           8       1.00      0.98      0.99       974
           9       0.99      0.99      0.99      1009

    accuracy                           0.99     10000
   macro avg       0.99      0.99      0.99     10000
weighted avg       0.99      0.99      0.99     10000
```

Model Evaluation and Visualization

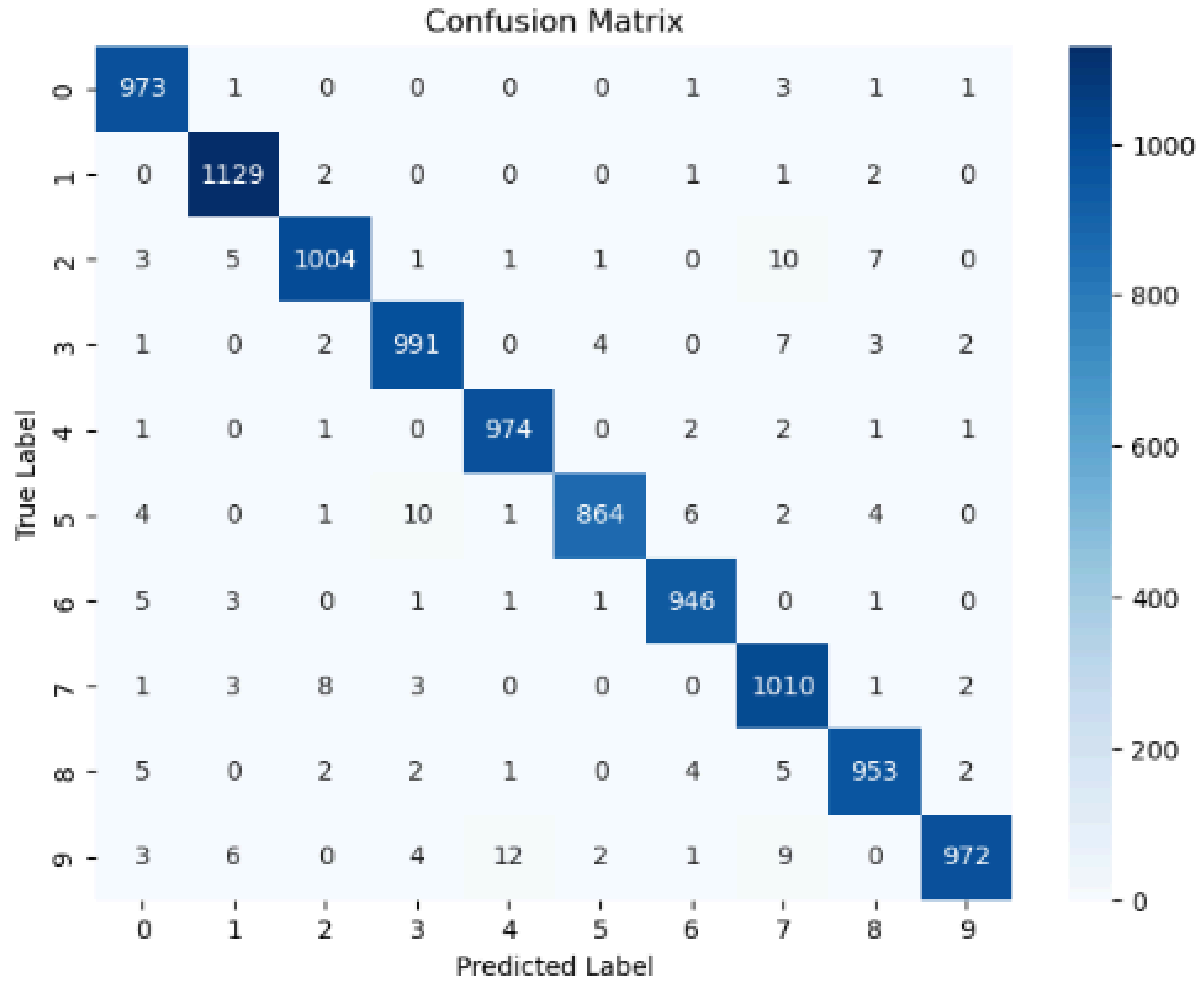
Performance Metrics – CNN Model (SGD Optimizer)





Test Set Accuracy: 97%

```
313/313 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step
```




```
Classification Report:
              precision    recall  f1-score   support

           0       0.98      0.99      0.98       980
           1       0.98      0.99      0.99      1135
           2       0.98      0.97      0.98      1032
           3       0.98      0.98      0.98      1010
           4       0.98      0.99      0.99       982
           5       0.99      0.97      0.98       892
           6       0.98      0.99      0.99       958
           7       0.96      0.98      0.97      1028
           8       0.98      0.98      0.98       974
           9       0.99      0.96      0.98      1009

    accuracy                           0.98     10000
   macro avg       0.98      0.98      0.98     10000
weighted avg       0.98      0.98      0.98     10000
```

Model Comparison: Adam vs SGD

Criteria	Adam Optimizer	SGD Optimizer
Test Accuracy	99%	97%
Validation Accuracy (Avg)	~99%	~96–97%
Training Stability	Very stable	Needs more epochs
Loss Reduction Speed	Fast	Slower
Overfitting	Minimal	Minimal
Final Recommendation	Better performance	Good but slower

Conclusion and Future Work

Conclusion:

- Successfully built and trained a CNN model to classify handwritten digits (MNIST dataset).
- Achieved **99% test accuracy** using the Adam optimizer.
- Visualization through accuracy, loss curves, and confusion matrix confirmed strong model performance.
- Demonstrated effective generalization with minimal overfitting.

Future Work:

- Experiment with **deeper architectures** (e.g., adding more Conv layers).
- Try **regularization techniques** like Dropout or L2 Regularization.
- Implement **data augmentation** to improve generalization.
- Extend model to **multi-digit recognition or custom handwritten datasets**

MASON KOREA
George Mason University®

Thank you all for your time and attention