**A Project for Data Science Lab**



**Handwritten Digit Recognition**

*A project submitted to the Department of Computer Science and Engineering in partial fulfillment of CSEL-4236 Data Science Lab course for the Degree of B.Sc. in Computer Science and Engineering.*

By

**Khandoker Habiba Toma**

**ID: B190305037**

**Tanmay Maitra Tanu**

**ID: B190305051**

**Md Emamul Haque Rafi**

**ID: B190305052**

Supervised By

**Dr. Md. Manowarul Islam**

**Associate Professor**

**Department of Computer Science and Engineering**

**Jagannath University , Dhaka**

Declaration

This is to certify that the work presented in this project is carried out by the candidates – **Khandoker Habiba Toma** (ID: B190305037) , **Tanmay Maitra Tanu** (ID: B190305051) and **Md Emamul Haque Rafi** (ID: B190305052) under the supervision of **Dr. Md. Manowarul Islam** in the Department of Computer Science and Engineering, Jagannath University, Dhaka - 1100, Bangladesh. It is also declared that neither of this project nor any part of this project has been submitted anywhere else for any degree of diploma. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

| Countersigned |  |  |
| --- | --- | --- |
| …………………….. |  |  |
| (Dr. Md. Manowarul Islam) |  |  |
| **Supervisor** |  |  |

ACKNOWLEDGEMENT

First of all we are thankful to Almighty Allah for his blessings for the successful completion of our project. The pleasure that accompanies the flourishing achievement of any task would be shortened without the mention of people whose continious cooperation made it possible, whose constant guidance and encouragement crown all efforts with success.

We extend our sincere thanks to Jagannath University, CSE Department which provided us with the opportunity to fulfill our wish and achieve our goal.

We would like to express deep debt to honorable teachers **Dr. Md. Manowarul Islam (**Project Supervisor)for his vital suggestions, meticulous guidance and constant motivation which went a long way in the successful completion of this project.

We cannot move on without thank beloved **Dr. Uzzal Kumar Acharjee**, Professor & Chairman of the Department for creating the required academic environment which made our task appreciable.

| Signature: ………………………    Date: ………………………  **Khandoker Habiba Toma**  Id: B190305037  Batch: 11th  Department of Computer Science &  Engineering  Jagannath University | Signature: ………………………    Date: ……………………………    **Tanmay Maitra Tanu**  Id: B190305051  Batch: 11th  Department of Computer Science &  Engineering  Jagannath University | Signature:  ………………………  Date:  ……………………………    **Md Emamul Haque Rafi**  Id: B190305052  Batch: 11th  Department of Computer Science &  Engineering  Jagannath University |
| --- | --- | --- |

**Table of Contents**

1. Title
2. Objective
3. Introduction
4. Tools and Technologies
5. Dataset
6. System Architecture

a. Data Preprocessing

b. Model Architecture

c. Model Compilation

d. Model Training

e. Model Saving

1. Result
2. Potential Enhancement
3. Conclusion

Lab Report: Handwritten Digit Recognition using CNN

# 1. Title

Handwritten Digit Recognition Using CNN with Keras and OpenCV.

# 2. Objective

To develop a convolutional neural network (CNN) model that can recognize handwritten digits (0–9) from the MNIST dataset. The objective is to preprocess the dataset, build and train the CNN, evaluate its performance, and deploy it using OpenCV or another interface.

# 3. Introduction

Handwritten digit recognition is one of the classic problems in the field of computer vision and deep learning. It is a multi-class classification problem where the input is an image of a digit and the output is a class label representing that digit. The MNIST dataset, which consists of thousands of labeled handwritten digits, provides a perfect starting point for training such a model. CNNs are highly effective in extracting spatial hierarchies from images and have been widely adopted for image classification tasks.

# 4. Tools and Technologies

- Python: Main programming language  
- TensorFlow/Keras: For building and training the CNN model  
- OpenCV: For preprocessing and interface development  
- NumPy: Numerical operations  
- MNIST Dataset: Dataset of 70,000 handwritten digit images  
- Matplotlib (Optional): For visualization of training performance

# 5. Dataset

**Name:** MNIST (Modified National Institute of Standards and Technology)

**Size:** 70,000 total images (60,000 training + 10,000 testing)

**Image size:** 28x28 grayscale

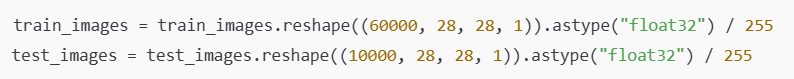
**Classes:** 10 (digits from 0 to 9)

# 6. System Architecture

The complete system consists of the following steps:

**A. Data Preprocessing**

- Reshape each image to (28,28,1) to add a channel dimension.  
- Normalize pixel values to a range of [0, 1] for better training convergence.



- Pad each image by 4 pixels on all sides to shift image size from 28x28 to 36x36 to help center the digits.



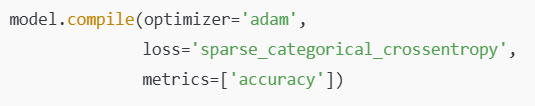
**B. Model Architecture**

- Conv2D layers: Extract features using 32 and 64 filters.  
- BatchNormalization: Normalize layer outputs to stabilize learning.  
- MaxPooling2D: Downsample the feature maps.  
- Dropout: Randomly drop neurons during training to avoid overfitting.  
- Flatten + Dense layers: For final classification.  
- Output: Dense layer with 10 units and softmax activation for 10 classes.



**C. Model Compilation**

- Optimizer: Adam  
- Loss Function: Sparse categorical crossentropy  
- Metrics: Accuracy



**D. Model Training**

Model is trained using fit() method with validation split and optionally early stopping. Evaluation should be done using test data for accuracy and loss.



**E. Model Saving**

After training, the model is saved in HDF5 format (.h5) using model.save().



# 7. Results

Though not shown in the code output, this CNN architecture generally yields over 98% accuracy on MNIST test data. The model is lightweight and effective for real-time applications. The .h5 file can be loaded later to make predictions on new digit inputs.

# 8. Potential Enhancements

- Add `model.evaluate(test\_images, test\_labels)` to get accuracy and loss.



- Add a real-time drawing interface using OpenCV or Tkinter.  
- Visualize training metrics (accuracy/loss) with matplotlib.  
- Use hyperparameter tuning (e.g., batch size, learning rate).  
- Deploy the model using Flask, Streamlit, or a desktop GUI.

# 9. Conclusion

This project demonstrates the successful development of a CNN model for handwritten digit recognition. The model is accurate, lightweight, and efficient. With some additional features like GUI or web interface, it can be transformed into a fully functional real-time digit recognition application.