IBM Innovation Center for Education

PHEMESOFT Software That Matters
TRUSTED DELIVERY PARTNER for IBM ICE

MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT & ENGINEERING

Final Project Report

on

Handwriting Digit Recognition

Team members:
Sonakshi Darak (N006)
Madhumita Joshi (N015)
Samiksha Shetty (N038)
Saakshi Yadav (N057)

TABLE OF CONTENTS

Executive Summary

Handwriting recognition (HWR) is the ability of a computer to receive and interpret intelligible handwritten input from sources such as paper documents, photographs, touch-screens and other devices. The image of the written text may be sensed "off line" from a piece of paper by optical scanning (optical character recognition) or intelligent word recognition. Alternatively, the movements of the pen tip may be sensed "on line", for example by a pen-based computer screen surface, a generally easier task as there are more clues available. A handwriting recognition system handles formatting, performs correct segmentation into characters, and finds the most plausible words.

Handwritten character recognition is one of the practically important issues in pattern recognition applications. The applications of digit recognition include postal mail sorting, bank check processing, form data entry, digital pens etc. The heart of the problem lies within the ability to develop an efficient algorithm that can recognize handwritten digits and which is submitted by users by way of a scanner, tablet, and other digital devices.

We have taken this a step further where our handwritten digit recognition system not only detects scanned images of handwritten digits but also allows writing digits on the screen with the help of an integrated GUI for recognition.

1. Background

1.1.

1.2. Aim

The aim is to create a handwriting digit recognition system that uses pattern matching to convert handwritten characters/digits into corresponding computer text or commands in real time.It is a hard task for the machine because handwritten digits are not uniform. Handwritten digit recognition is the solution to this problem which uses the image of a digit and recognizes the digit present in the image.

1.3. Technologies

a) Image Processing : Used for acquiring and preprocessing images of human handwriting.

b) OCR : OCR stands for "Optical Character Recognition." It is a technology that recognizes text within a digital image. It is commonly used to recognize text in scanned. .documents and images.

c) CNN: A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.

1.4. Hardware Architecture

Device required	• Laptop/Desktop. • Mobile phone. • Camera with moderate resolution.
Operating system	• Laptop/Desktop: Ubuntu,Windows • Mobile phone:ios,Android
Memory (RAM)	• Minimum 4GB
Processor	• Generation i5 onwards
Internet	• 4G LTE or WI-fi connectivity is encouraged
GPS connectivity	• Required
Image requirements	• Moderate resolution • Colour images only

1.5. Software Architecture

Language:

a) Python

Libraries:

a) Keras - Open source Python library for developing and evaluating deep learning models.
b) Tensorflow - TensorFlow is one of the most in-demand and popular open-source deep learning frameworks available today.
c) Sklearn - It is a machine learning library that provides several useful tools
d) Tkinter - The tkinter package is the standard Python interface to the Tcl/Tk GUI toolkit.
e) Flask - Flask is a web application framework written in Python.
f) Numpy - Python library that is used to perform operations on arrays.
g) Google Colab

2. System

2.1. Requirements

2.1.1. Functional requirements

a) The system should process the input given by the user by drawing the digit to be

detected on GUI.

b) System shall wait for the user to draw the digit to be predicted.

c) System should detect numbers drawn on the user interface after the user clicks on the Recognise tab.

d) System should display how accurately it has predicted the digit.

2.1.2. User requirements

a) Users should be able to understand the working of the system and the user interface.

b) Users should be able to draw digits on the screen as per their convenience for the system to recognise.

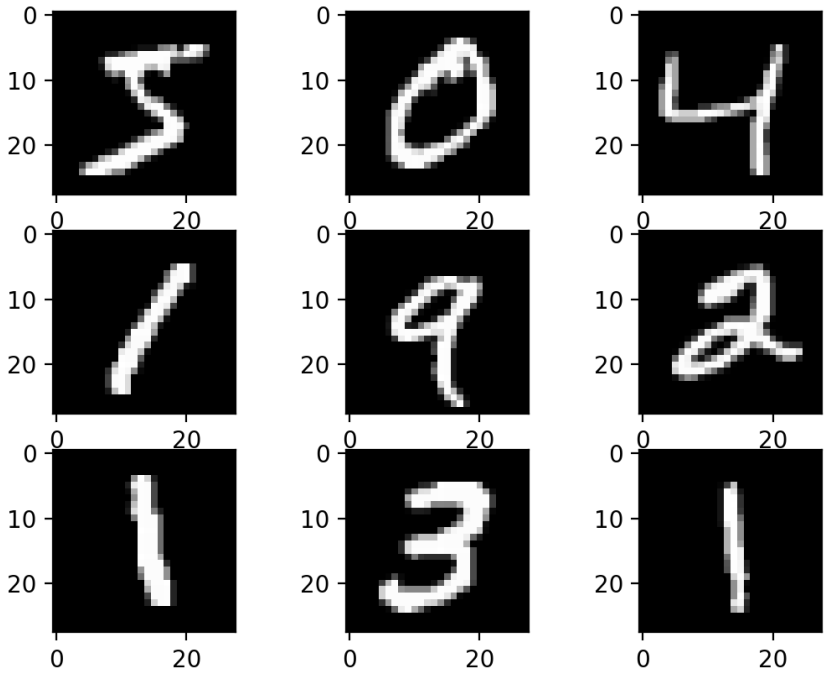
c) Users should also be able to see how well the system has predicted the digit they have drawn.

d) Users should be able to clear the output and draw a new digit to be recognised.

2.2. Implementation

Dataset - MNIST

The MNIST database of handwritten digits, available from this page, has a training set of 60,000 examples, and a test set of 10,000 examples. It is a subset of a larger set available from NIST. The digits have been size-normalized and centered in a fixed-size image. It is a good database for people who want to try learning techniques and pattern recognition methods on real-world data while spending minimal efforts on preprocessing and formatting.



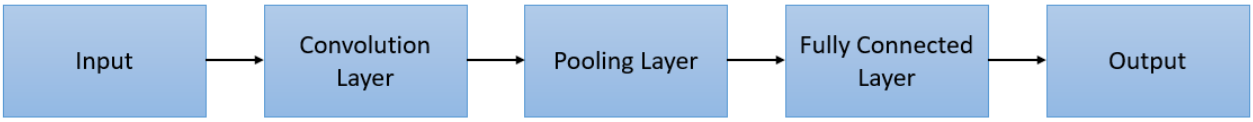
PREPROCESSING

Pre-processing is a stage in machine and deep learning that focuses on removing undesired contaminants and redundancy from the input data. We reshaped all of the photos in the dataset into 2-dimensional images

to simplify and break down the input data (28,28,1). The photos' pixel values range from 0 to 255, therefore we normalized them by converting the dataset to 'float32' and then dividing by 255.0, resulting in input features that range from 0.0 to 1.0. Following that, we used one-hot encoding to convert the y values into zeros and ones, converting each number into a categorical value, such as an output value of 4 will be converted to zeros and ones i.e [0,0,0,0,1,0,0,0,0]

CONVOLUTIONAL NEURAL NETWORK

CNN is a deep learning algorithm that is widely used for image recognition and classification .A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. It is a class of deep neural networks that require minimum pre-processing.They have three main types of layers, which are Convolutional layer, Pooling layer, Fully-connected (FC) layer. The convolutional layer is the first layer of a convolutional network. While convolutional layers can be followed by additional convolutional layers or pooling layers, the fully-connected layer is the final layer.



Keras is used to implement handwritten digit recognition using a Convolutional Neural Network. It is a free and open-source neural network library for creating and implementing deep learning models. We used a Sequential class from Keras, which allowed us to generate a layer-by-layer model.

The input image's dimensions are 28 (height), 28 (width), and 1 (height) (Number of channels). Next, we

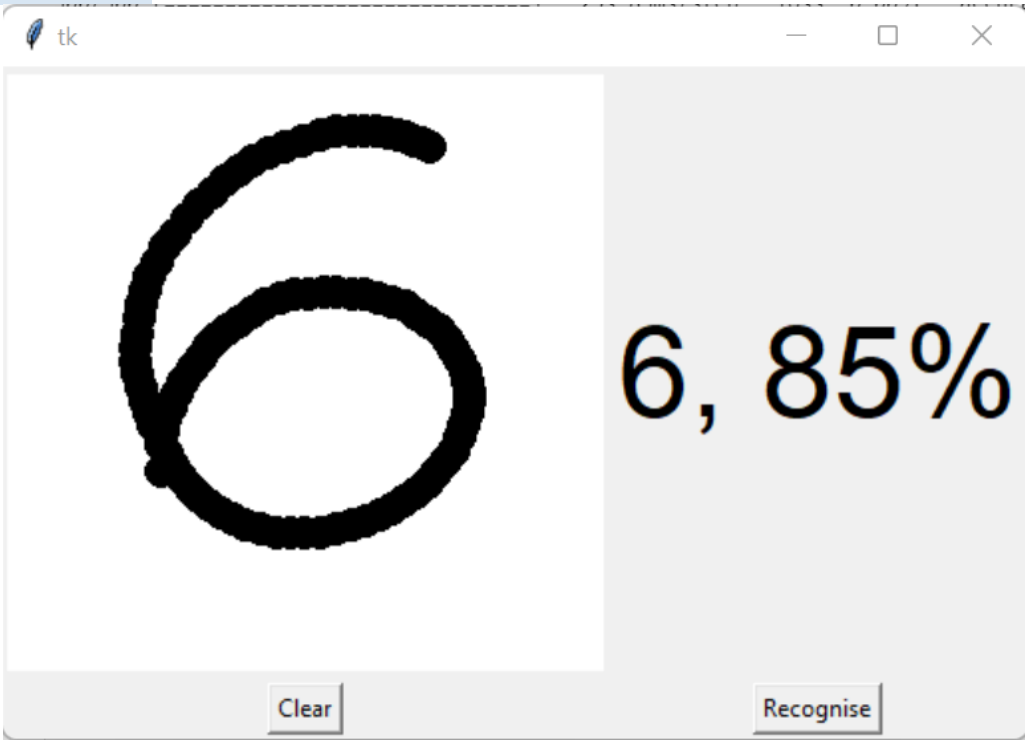
generated a model using a Conv layer as the first layer. This layer employs a matrix to extract features from the input data by convolving it over its height and width. A Filter or Kernel is the name given to this matrix. Weights are the values in the filter matrix. With a stride of 1, we utilized 25 filters in each of the dimensions (3,3).

The number of pixels that shift is determined by the stride. The dimension of activation maps is determined by the formula: ((N + 2P - F)/S) + 1, where N is the dimension of the input image, P is padding, F is filter dimension, and S is stride. We have used the ReLu activation function.

Then we utilized a MAX-pooling layer, which only maintains the highest value from a pool with a pool-size (2,2). Following that, a Flatten Layer is utilized, which involves flattening the 2-dimensional matrix, i.e. constructing a column matrix (vector) from it. The completely connected layer will receive this column vector. There are 100 neurons in this layer. The output of the Relu activation function is fed into the model's output layer, which is the last layer. This layer has ten neurons that represent classes (numbers ranging from 0 to 9) and uses the SoftMax algorithm to classify them. The probability distribution for all ten classes is returned by this function. The output is the class with the highest probability.

2.3. Graphical User Interface

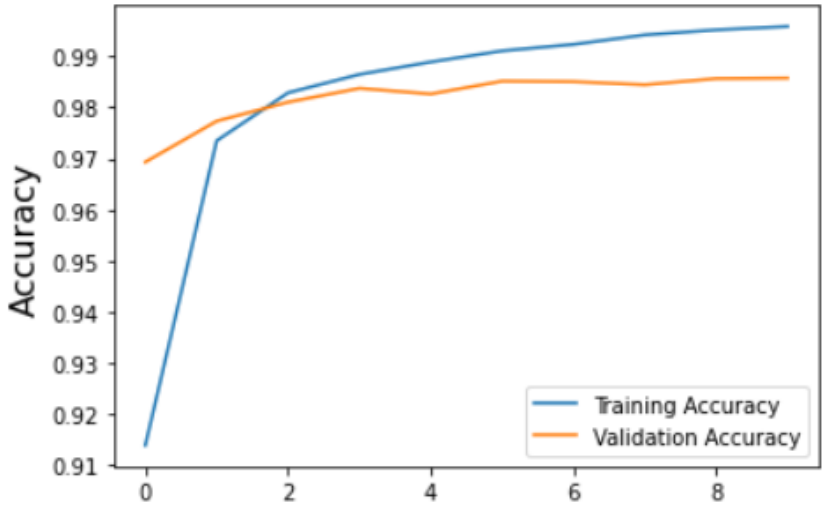
Initial Output


tk
6, 85%
Clear
Recognise


tk
3, 93%
Clear
Recognise

2.4. Evaluation

The model is trained with the batch size of 250 for 10 epochs. This provides us with a training accuracy of 99.53% and test accuracy of 98.46%.



3. Conclusion

In this project, we have successfully built a Python deep learning model on a handwritten digit

recognition app. We have built and trained the Convolutional neural network which is very effective for image classification purposes. Later on, we build the GUI where we draw a digit on the canvas then we classify the digit and show the results.

By comparing the execution times of the algorithms, we discovered that increasing the number of epochs without changing the configuration of the algorithm is pointless due to a model's limitation, and we discovered that after a certain number of epochs, the model starts overfitting the dataset and gives us biased predictions.