

PROJECT REPORT

On

HANDWRITTEN DIGIT DETECTION

(using python)

(CSE III Semester Mini project)

2023-2024



Submitted to:

Ms. Jyoti Ramola

(CC-CSE-III SEM)

Submitted by:

TUSHAR SONI

Roll no.: 2220037

CSE-III SEM

Session: 2023-24

DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

GRAPHIC ERA HILL UNIVERSITY, DEHRADUN

CERTIFICATE

Certified that Mr. Tushar Soni has developed mini project on “Digit Detection using Python” for the CS III Semester Mini Project Lab in Graphic Era Hill University, Dehradun. The project carried out by Students is their own work as best of my knowledge.

DATE:

Ms. Jyoti Ramola

Class Co-ordinator

CSE-CC-S-III SEM

(CSE Department)

GEHU, Dehradun

ACKNOWLEDGEMENT

We would like to express our gratitude to The Almighty Shiva Baba, the most Beneficent and the most Merciful, for completion of project.

We wish to thank our parents for their continuing support and encouragement. We also wish to thank them for providing us with the opportunity to reach this far in our studies.

We would like to thank particularly our project Co-ordinator Ms Jyoti Ramola for her patience, support and encouragement throughout the completion of this project and having faith in us.

At last but not the least We greatly indebted to all other persons who directly or indirectly helped us during this work.

Mr. Tushar Soni

Roll no.: 2220037

CSE-III SEM

Session: 2023-24

GEHU, Dehradun

TABLE OF CONTENTS

<u>CONTENT</u>	<u>PAGE NO.</u>
Introduction	1-2
IMPORTING IMPORTANT LIBRARIES	3-4
APPLICATIONS FOR THIS SYSTEM	5
BREAKDOWN OF WORKING OF THE SYSTEM	6-7
Abstract Summary	8
References	9

INTRODUCTION

What is python?

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

- web development (server-side),
- software development,
- mathematics,
- system scripting.

What can Python do?

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

Why Python?

- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
- Python can be treated in a procedural way, an object-oriented way or a functional way.

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

IMPORTING IMPORTANT LIBRARIES

- Cv2:
 - The cv2 module, often referred to as the OpenCV module in Python, is a wrapper for the OpenCV library and provides a Python interface for many of its functions.
 - The cv2 module provides functions for reading, writing, and processing images and videos. It allows developers to perform operations such as resizing, cropping, color space conversion, and more.
 - OpenCV in Python includes tools for creating graphical user interfaces (GUIs) to interact with images and videos.
 - The cv2 module seamlessly integrates with the NumPy library, making it convenient to manipulate and process image data using NumPy arrays. This integration enhances the efficiency and flexibility of image processing operations.
- Numpy:
 - NumPy introduces the `numpy.array` class, which represents multi-dimensional arrays. These arrays can be one-dimensional (vectors), two-dimensional (matrices), or even higher-dimensional.
 - NumPy supports broadcasting, a powerful feature that allows operations between arrays of different shapes and sizes.
 - NumPy's array representation is memory-efficient, allowing for the storage of large datasets without the overhead associated with Python lists. NumPy arrays also enable efficient memory operations and data manipulation.
- matplotlib:
 - Matplotlib is a popular Python library for creating static, interactive, and animated visualizations in a wide range of formats.
 - Matplotlib provides a variety of functions for creating different types of plots, including line plots, scatter plots, bar plots, histograms, pie charts, and more. These functions are part of the `pyplot` module within Matplotlib.
 -

- Matplotlib includes functionality for creating 3D plots and visualizations. This is useful for visualizing three-dimensional data and surfaces.
- tensorflow:
 - TensorFlow is an open-source machine learning library developed by the Google Brain team. It is widely used for building and training machine learning models, especially in the field of deep learning.
 - TensorFlow is highly flexible and can be used for a wide range of machine learning tasks, including but not limited to deep learning, neural networks, natural language processing, image recognition, and more.
- os:
 - Used for interacting with the operating system, such as changing directories for saving files.

APPLICATIONS FOR THIS SYSTEM

Digit Recognition

Automates the classification of handwritten digits (0-9) in images, with applications in:

- Postal address sorting
- Bank check processing
- Educational assessment
- Forms processing
- Historical document analysis

Education and Learning:

- Demonstrates fundamental concepts of machine learning and neural networks:
 - Image processing
 - Model training and evaluation
 - Prediction using trained model

Integration into Larger Systems:

- Serves as a building block for more complex tasks:
 - Extracting numerical information from images or documents
 - Building handwriting recognition systems for diverse languages and scripts

BREAKDOWN OF WORKING OF THE SYSTEM

Data Acquisition and Preprocessing:

- **MNIST Dataset:** The project utilizes the MNIST dataset, consisting of thousands of labeled images of handwritten digits (0-9).
- **Data Loading and Normalization:** The script imports the MNIST data and normalizes the pixel values of each image between 0 and 1 for better training.

Model Building and Training:

- **Sequential Model:** A sequential model with three layers is created:
 - **Flatten Layer:** Reshapes the 28x28 pixel images into 784-dimensional vectors.
 - **Two Dense Layers:** Each with 128 neurons and ReLU activation, allowing the model to learn complex features from the data.
 - **Output Layer:** Contains 10 neurons (one for each digit) and uses the Softmax activation to predict the probability distribution of each digit class for an input image.
- **Model Compilation:** Configures the training process by specifying the optimizer (Adam), loss function (Sparse Categorical Cross-Entropy), and evaluation metric (Accuracy).
- **Model Training:** The model is trained for 3 epochs using the training data, adjusting its internal parameters to minimize the loss and improve accuracy.

Model Evaluation and Deployment:

- **Model Evaluation:** The trained model is evaluated on the unseen test data, reporting the final loss and accuracy metrics.
- **Model Saving:** The trained model is saved as "handwritten.model" for future use on new images.

- Model Deployment: The script opens images from the "Samples" directory, predicts the digit class with the highest probability, and displays the image and prediction.

Overall Workflow:

- Load and preprocess MNIST data.
- Build and train a deep learning model.
- Evaluate the model's performance.
- Save the trained model for later use.
- Apply the model to classify handwritten digits in new images.

Key Technologies:

- TensorFlow: Deep learning framework for model building and training.
- Keras: High-level API for TensorFlow, simplifying model creation and development.
- MNIST Dataset: Widely used benchmark dataset for handwritten digit recognition.

Additional Notes:

The script includes exception handling to gracefully handle errors during image loading.

The provided abstract can be used as a starting point for your report.

Abstract Summary

This report explores the development and implementation of a deep learning model for handwritten digit recognition. The model utilizes the MNIST dataset, TensorFlow, and Keras to achieve high accuracy in classifying digits from images. The code is structured to train the model, evaluate its performance, save it for future use, and apply it to new images. Potential applications of the model span various domains, including postal automation, banking, education, and document analysis. The project also serves as a valuable educational tool for understanding machine learning concepts.

REFERENCES

- Github.com
- Bard AI
- Wikipedia
- YouTube