Industrial Project Report

Submitted in partial fulfilment of the degree of

B-tech in Computer Science & Engineering

By

Sourav Chowdhury [11900121002]
Debangan Banerjee [11900121003]
Shnehill Adhikary [11900121038]
Priyadeep Sen [11900121041]
Ritindra Debnath [11900121044]
Devneel Roy [11900121048]

Second-year student of

SILIGURI INSTITUTE OF TECHNOLOGY

THIS IS SUBMITTED IN FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

AFFILIATED TO

Maulana Abul Kalam Azad University of Technology

Under the supervision of:- Mr. Ripam Kundu

Sikharthy Infotech Pvt. Ltd.

PROJECT ON: HANDWRITTEN DIGIT RECOGNITION USING TENSORFLOW AND OPENCV

By

Sourav Chowdhury [11900121002]
Debangan Banerjee [11900121003]
Shnehill Adhikary [11900121038]
Priyadeep Sen [11900121041]
Ritindra Debnath [11900121044]
Devneel Roy [11900121048]

UNDER THE GUIDANCE OF

Mr. Ripam Kundu

Project Guide

Sikharthy Infotech Pvt. Ltd.

THIS IS SUBMITTED IN FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

B.Tech

IN

Computer Science and Engineering

SILIGURI INSTITUTE OF TECHNOLOGY

AFFILIATED TO

Siliguri Institute of Technology Hill Cart Road, Salbari, Sukna, West Bengal 734009

Maulana Abul Kalam Azad University of Technology

Department of Computer Science and Engineering

I hereby forward the documentation prepared under my supervision by **Mr. Ripam Kundu** entitled **Siliguri Institute Of Technology** to be accepted as fulfilment of the requirement for the Degree of Bachelor of Technology in Computer Science and Engineering, **Siliguri Institute Of Technology** affiliated to **Maulana Abul Kalam Azad University of Technology** (**MAKAUT**).

Mr.Ripam Kundu
(Software Developer)
Project Guide
Sikharthy Infotech Pvt. Ltd.

HOD

Department Of Computer Science & Engineering,
SIT

Shilpi Ghosal
(Director)
Sikharthy Infotech Pvt. Ltd.

TPO

Certificate of Approval

The foregoing project is hereby approved as a creditable study for the B.Tech in Computer Science & Engineering presented in a manner of satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approved any statement made, opinion expressed or conclusion therein but approve this project only for the purpose for which it is submitted.

Final Examination for
Evaluation of the Project

Signatures of Examiners

ABSTRACT

*The paper explains the working of a Handwritten Digit recognition using Tensorflow and Opencv. The code uses a dataset **mnist** which is a part of the **Keras library** which is a part of **Tensorflow** developed by Google developers. This code is meant to open a camera that scans handwritten digits and returns the outputs i.e identify the digits.*

Handwritten digit recognition has many practical applications, such as in optical character recognition (OCR), where it can be used to read and interpret handwriting on documents, or in check processing systems, where it can be used to automatically read handwritten numbers on checks.

ACKNOWLEDGEMENT

It is a great pleasure for me to acknowledge the assistance and participation of a large number of individuals in this attempt. Our project report has been structured under the valued suggestion, support, and guidance of **Mr. Ripam Kundu**. Under his guidance, we have accomplished the challenging task in a very short time.

Finally, we express our sincere thankfulness to our family members for inspiring me all throughout and always encouraging us.

Group Member Signature

_____

_____

_____

_____

_____

_____

TABLE OF CONTENTS

INTRODUCTION

We have used Python and its different libraries to complete the uber data analysis

Python is a high-level, general-purpose and a very popular programming language. Python programming language (latest Python 3) is being used in web development, Machine Learning applications, along with all cutting-edge technology in Software Industry. Python Programming Language is very well suited for Beginners, also for experienced programmers with other programming languages like C++ and Java.

Below are some facts about Python Programming Language:

- Python is currently the most widely used multi-purpose, high-level programming language.
- Python allows programming in Object-Oriented and Procedural paradigms.
- Python programs generally are smaller than other programming languages like Java. Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time.
- Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc.

The biggest strength of Python is huge collection of standard libraries which can be used for the following:

1. Machine Learning
2. GUI Applications (like Kivy, Tkinter, PyQt etc.)
3. Web frameworks like Django (used by YouTube, Instagram, Dropbox)
4. Image processing (like OpenCV, Pillow)
5. Web scraping (like Scrapy, BeautifulSoup, Selenium)
6. Test frameworks
7. Multimedia
8. Scientific computing

Disadvantages of python:

Python is a widely used general-purpose, high-level programming language. It is widely used by developers in various domains, from web-development to Machine Learning. Though, Python has its own set of advantages and disadvantages. Let's see some of the disadvantages of Python.

Speed: Python is an interpreted language and is slow as compared to C/C++ or Java. Unlike C or C++ it's not closer to hardware because Python is a high-level language. As we all know that compilation and execution help to work normally, but in this case, execution of Python takes place with the help of an interpreter instead of the compiler as we have seen that Python code is executed line by line, which causes it to slow down. Speed is a focal point for the project required by any programmer. On the other hand, it can be seen that it is fast for many web applications too.

Mobile Development: However Python is strong in desktop and server platforms, that is it is an excellent server-side language but for mobile development, Python is not a very good language which means it is a weak language for mobile development. It is very rarely used for mobile development. This is the reason very few mobile applications are built in it like Carbonnelle, which is built-in python.

Memory Consumption: For any memory intensive tasks Python is not a good choice. That is why it is not used for that purpose. Python's memory consumption is also high, due to the flexibility of the data types.

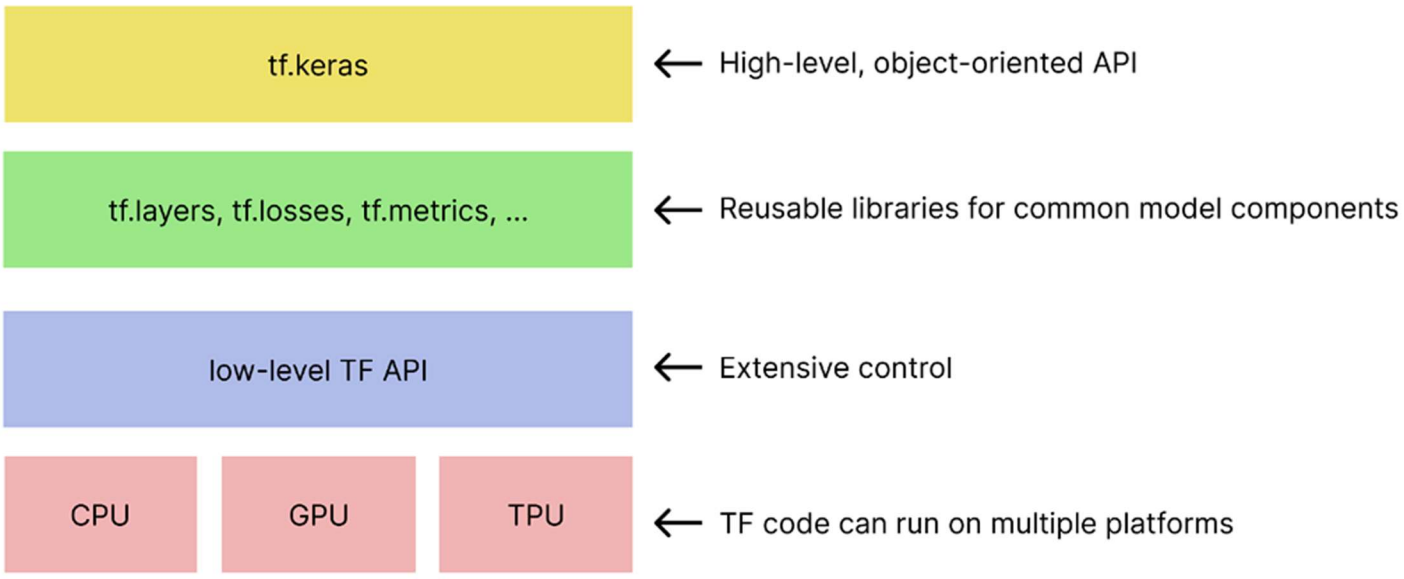
LIBRARIES USED

TensorFlow:

TensorFlow is an end-to-end open-source platform for machine learning. TensorFlow is a rich system for managing all aspects of a machine learning system; however, this class focuses on using a particular TensorFlow API to develop and train machine learning models. See the TensorFlow documentation for complete details on the broader TensorFlow system.

TensorFlow APIs are arranged hierarchically, with the high-level APIs built on the low-level APIs. Machine learning researchers use the low-level APIs to create and explore new machine learning algorithms. In this class, you will use a high-level API named tf.keras to define and train machine learning models and to make predictions. tf.keras is the TensorFlow variant of the open-source Keras API.

The following figure shows the hierarchy of TensorFlow toolkits:



OpenCV:

OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today's systems. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human. When it integrated with various libraries, such as NumPy, python is capable of processing the OpenCV array structure for analysis. To Identify image pattern and its various features we use vector space and perform mathematical operations on these features.

The first OpenCV version was 1.0. OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. When OpenCV was designed the main focus was real-time applications for computational efficiency. All things are written in optimized C/C++ to take advantage of multi-core processing.
Look at the following images.

Applications of OpenCV: There are lots of applications which are solved using OpenCV, some of them are listed below:

- face recognition
- Automated inspection and surveillance
- number of people – count (foot traffic in a mall, etc)
- Vehicle counting on highways along with their speeds
- Interactive art installations
- Anomaly (defect) detection in the manufacturing process (the odd defective products)
- Street view image stitching
- Video/image search and retrieval
- Robot and driver-less car navigation and control
- object recognition
- Medical image analysis
- Movies – 3D structure from motion
- TV Channels advertisement recognition

NumPy:

This article will help you get acquainted with the widely used array-processing library in Python, NumPy. What is NumPy? NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python. It is open-source software. It contains various features including these important ones:

A powerful N-dimensional array object
Sophisticated (broadcasting) functions
Tools for integrating C/C++ and Fortran code
Useful linear algebra, Fourier transform, and random number capabilities
Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using NumPy which allows NumPy to integrate with a wide variety of databases seamlessly and speedily.

With the revolution of data science, data analysis libraries like NumPy, SciPy, Pandas, etc. have seen a lot of growth. With a much easier syntax than other programming languages, python is the first-choice language for the data scientist.

NumPy provides a convenient and efficient way to handle the vast amount of data. NumPy is also very convenient with Matrix multiplication and data reshaping. NumPy is fast which makes it reasonable to work with a large set of data.

There are the following advantages of using NumPy for data analysis.

1. NumPy performs array-oriented computing.
2. It efficiently implements the multidimensional arrays.
3. It performs scientific computations.
4. It can perform Fourier Transform and reshaping the data stored in multidimensional arrays.
5. NumPy provides the in-built functions for linear algebra and random number generation.

Nowadays, NumPy in combination with SciPy and Mat-plotlib is used as the replacement to MATLAB as Python is more complete and easier programming language than MATLAB.

Matplotlib:

Matplotlib is a data visualization library for Python that allows you to create a wide range of static, animated, and interactive visualizations in Python. It is one of the most widely used visualization libraries in Python and is an essential tool for data scientists, researchers, and engineers who work with data.

Matplotlib provides a comprehensive set of tools for creating line plots, scatter plots, bar plots, histograms, and many other types of visualizations. It also provides a high degree of customization for each visualization, allowing you to fine-tune the appearance of your plots to meet your needs.

Matplotlib is designed to work seamlessly with other popular Python libraries for data analysis, such as NumPy, pandas, and SciPy. It is also compatible with Jupyter notebooks, making it easy to create interactive visualizations within your notebooks.

Matplotlib has a large and active user community that has contributed a vast number of examples, tutorials, and documentation to help users learn and use the library effectively. Additionally, Matplotlib provides extensive documentation and a variety of resources to help you get started and master its functionality.

There are several advantages of using Matplotlib for data visualization:

- Easy to use: Matplotlib is a user-friendly library that offers an intuitive and straightforward interface for creating visualizations.
- Wide range of visualisations: Matplotlib provides a broad range of visualizations, including line plots, scatter plots, bar charts, pie charts, and many more.
- Highly customizable: Matplotlib offers a high degree of customization for your visualizations.
- Integration with other libraries: Matplotlib can be easily integrated with other libraries such as NumPy, pandas, and SciPy.
- Interactive Visualisation: Matplotlib can be used to create interactive visualizations, allowing you to explore your data in real-time.

To import all these libraries, we can use the below code:

```
import tensorflow as tf
import cv2
import numpy as np
import matplotlib.pyplot as plt
```

Load the MNIST dataset.

```
dataset = tf.keras.datasets.mnist

(X_train, y_train), (X_test, y_test) = dataset.load_data()
```

Divide Training and Testing data:

```
X_train= X_train/255.0
X_test= X_test/255.0
```

Reshape Train and Test data

```
X_train = X_train.reshape(X_train.shape[0], -1)
X_test = X_test.reshape(X_test.shape[0], -1)
```

Import Sequential, Dense, Dropout from keras models

```
from tensorflow.keras.models import load_model
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout
```

The Model

```
model = Sequential()

model.add(Dense(128, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(10, activation='softmax'))
```

Compile the Model

```
model.compile('adam', 'sparse_categorical_crossentropy', metrics=['acc'])
```

Epoch is the date and time relative to which a computer's clock and timestamp values are determined. Here, Epochs is 3

```
model.fit(X_train, y_train, epochs=3, batch_size=12, validation_split=0.1)
```

Upon entering the data, the output is:

```
Epoch 1/3
4500/4500 [==============================] - 21s 4ms/step - loss: 0.2889 - acc: 0.9118 - val_loss: 0.0982 - val_acc: 0.9718
Epoch 2/3
4500/4500 [==============================] - 17s 4ms/step - loss: 0.1548 - acc: 0.9543 - val_loss: 0.0838 - val_acc: 0.9758
Epoch 3/3
4500/4500 [==============================] - 16s 4ms/step - loss: 0.1240 - acc: 0.9623 - val_loss: 0.0928 - val_acc: 0.9695
<keras.callbacks.History at 0x7fa8cf187460>
```

Plotting an example value using matplotlib.pyplot and predicting the result:

```
plt.imshow(X_test[1550].reshape(28,28), cmap='gray')
plt.xlabel(y_test[1550])
plt.ylabel(np.argmax(model.predict(X_test)[1255]))
```

Save the model with '.h5' extension

```
model.save('digit_trained.h5')
```

Code for capturing the webcam using CV2 video capture.

```
cap = cv2.VideoCapture(0)  # '0' for Primary camera and '1' for Secondary camera
while True:
    ret, img = cap.read()
    #img = cv2.flip(img, 1)
    img = img[200:400, 200:400]
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    _, gray = cv2.threshold(gray, 127, 255, cv2.THRESH_BINARY_INV)
    cv2.imshow("gray_wind", gray)
    gray = cv2.resize(gray, (28, 28))
    #cv2.imshow('resized')
    gray = gray.reshape(1, 784)
    result = np.argmax(model.predict(gray))
    result = 'cnn : {}'.format(result)
    cv2.putText(img, org=(25,25), fontFace=cv2.FONT_HERSHEY_SIMPLEX, fontScale=1, text= result, color=(255,0,0), thickness=2)
    cv2.imshow("image", img)

    if cv2.waitKey(33) == ord('a'):
        break
cap.release()
cv2.destroyAllWindows()
```

Output:

FUNCTIONAL REQUIREMENTS OF THE SYSTEM

SOFTWARE:

- Operating System
- Windows OS 10 and later
- Python (3.9 and later)
- Visual Studio Code
- Jupyter Notebook

HARDWARE:

- Camera

PROGRAMMING LANGUAGE :

- Python

Conclusion:

In general, handwriting detection can be a useful technology for a variety of applications, such as digitizing handwritten documents, recognizing signatures, and even analysing handwriting for forensic purposes.

The project involves building a machine learning model to recognize handwriting, collecting and annotating a dataset of handwritten samples, and evaluating the model's performance using metrics such as accuracy and precision.

The conclusion of the project is that the model achieved high accuracy in recognizing handwriting, indicating that it has potential for practical use in relevant applications. Alternatively, the conclusion could be that there are limitations to the technology that still need to be addressed, such as difficulty recognizing handwriting with certain styles or variations.

Ultimately, the conclusion reflects the insights gained from the project and any potential implications for future research or practical implementation.

REFERENCE

www.geesksforgeesks.com

www.kaggle.com

www.w3school.com

www.github.com