

A Novel Method for Handwritten Digit Recognition System -AI

A Project report submitted in partial fulfillment of 7th semester in degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

Team ID: PNT2022TMID42292

TEAM MEMBERS

AGILAN T 710119104002

BENISHA M 710119104004

SIVA MOORTHY K 710119104017

SOWMIYAN P 710119104018



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ADITHYA INSTITUTE OF TECHNOLOGY, COIMBATORE

ANNA UNIVERSITY: CHENNAI 600025

NOV-2022

ADITHYA INSTITUTE OF TECHNOLOGY,COIMBATORE
(A Constituent College of Anna University, Chennai)



BONAFIDE CERTIFICATE

Certified that this project report ” A Novel Method for Handwritten Digit Recognition System” is the bonafide record work done by Ms BENISHA.M(710119104004),Mr AGILAN.T(710119104004),Mr SIVA MOORTHY.K(710119104017) and Mr SOWMIYAN.P(710119104018) for IBM-NALAIYATHIRAN in VII semester of B.E., degree course in Computer Science and Engineering branch during the academic year of 2022 - 2023.

Staff-In charge
Dr.k.Pazhanisamy

Evaluator
Dr.R.Arunprakash

Head of the Department
Dr.S.Jayanthi

ACKNOWLEDGEMENT

We express our breathless thanks to our **Dr.P.Senthamil Selvan M.Pharm., Ph.D.**, the Dean(i/c), University College of Engineering, Ariyalur, for giving constant motivation in succeeding in our goal.

We acknowledge our sincere thanks to Head of the Department (i/c) **Dr.S.Jayanthi M.Tech., Ph.D.**, for giving us valuable suggestion and help towards us throughout this Project.

We are highly grateful to thank our Project coordinator **Dr.K.Pazhanisamy M.Tech., Ph.D.**, and our Project Evaluator **Dr.R.Arunprakash M.E.,(Ph.D)**., Department of Computer Science and Engineering, University College of Engineering , Ariyalur, for the coordinating us throughout this Project.

We are very much indebted to thank all the faculty members of Department of Computer science and Engineering in our Institute, for their excellent moral support and suggestions to complete our Project work successfully.

Finally our acknowledgment does our parents, sisters and friends those who had extended their excellent support and ideas to make our Project a pledge one.

Benisha M
Agilan T
Sivamoorthi K
Sowmiyan P

ABSTRACT

An enormous number of CNN classification algorithms have been proposed in the literature. Nevertheless, in these algorithms, appropriate filter size selection, data preparation, limitations in datasets, and noise have not been taken into consideration. In recent decades, the image classification problem has been widely addressed in the literature and it is still an active research field today.

We have used MNIST handwritten digits as a dataset. As a result, our CNN algorithm achieves state-of-the-art results in handwritten digit recognition, with a recognition accuracy of 99.98%, and 99.40% with noise. To guarantee that the dataset is free from any unnecessary details and is suitable to apply in our CNN model, data preparation is considered as essential first step in our proposed model.

1. INTRODUCTION

1.1 Project Overview

1.2 Purpose

2. LITERATURE SURVEY

2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

4.2 Non-Functional requirements

5. PROJECT DESIGN

5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

6.3 Reports from JIRA

7. CODING & SOLUTIONING

7.1 Feature 1

7.2 Feature 2

7.3 Database Schema (if Applicable)

8. TESTING

8.1 Test Cases

8.2 User Acceptance Testing

9. RESULTS

9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX

Source Code GitHub & Project Demo Link

1.INTRODUCTION

1.1 Project Overview:

In the world of artificial intelligence (AI), the recognition of handwritten digits proves that you got your neurons right and in working condition. This application of AI is already quite old – its breakthrough came in 1989 when a reliable machine-enabled parsing of ZIP codes for postal services was achieved. Soon after, it was proven that multi-layer feed-forward networks can implement any function. Shortly, financial institutions adopted the technique for the automatic parsing of account numbers on remittance slips for wire transfers or bank checks. Today the recognition of handwritten digits using different AI-techniques is also famous in academia for teaching and learning purposes.

MNIST dataset contains 60,000 training images of handwritten digits from zero to nine and 10,000 images for testing. So, the MNIST dataset has 10 different classes. The handwritten digits images are represented as a 28×28 matrix where each cell contains grayscale pixel value.

The following steps will be followed in handwritten digit recognition,

1. Import the libraries and load the dataset
2. Preprocess the data
3. Create the model
4. Train the model
5. Evaluate the model
6. Create GUI to predict digits

1.2 Purpose:

Applications of offline handwritten digit recognition are numerous: **reading postal addresses, bank check amounts, and forms**. Furthermore, OCR plays an important role for digital libraries, allowing the entry of image textual information into computers by digitization, image restoration, and recognition methods.

Handwritten digit recognition is the process to **provide the ability to machines to recognize human handwritten digits**. It is not an easy task for the machine because handwritten digits are not perfect, vary from person-to-person, and can be made with many different flavors.

They can be divided into three main steps: **preprocessing step, feature extraction and selection step, and classification and verification step**. Handwritten digit recognition problem can be seen as a subtask of the optical character recognition (OCR) problem.

Handwritten digit recognition using MNIST dataset is a major project made with the help of Neural Network. It basically **detects the scanned images of handwritten digits**. Handwritten digit recognition is **classes (0-9). the ability of a computer to recognize the human handwritten digits from different sources like images, papers, touch screens, etc, and classify them into 10 predefined** .

2.Literature Survey:

2.1 Existing Problem:

Create a problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love.

A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face. Throughout the process, you'll also be able to empathize with your customers, which helps you better understand how they perceive your product or service.

2.2 References:

Paper 1: Novel Deep Neural Network Model for Handwritten Digit Classification and Recognition

Year: 2021

Authors: Ayush Kumar Agrawal and Vineet Kumar Awasthi

Paper 2: A Novel Handwritten Digit Classification System.Based on Convolutional Neural Network Approach

Year: 2021

Authors: Ali Abdullah Yahya, Jieqing Tan, Min Hu

Paper 3: Handwritten Character Recognition using Neural Network and TensorFlow

Year : 2019

Authors : Megha Agarwal, Shalika, Vinam Tomar, Priyanka Gupta

Paper 4: Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN)

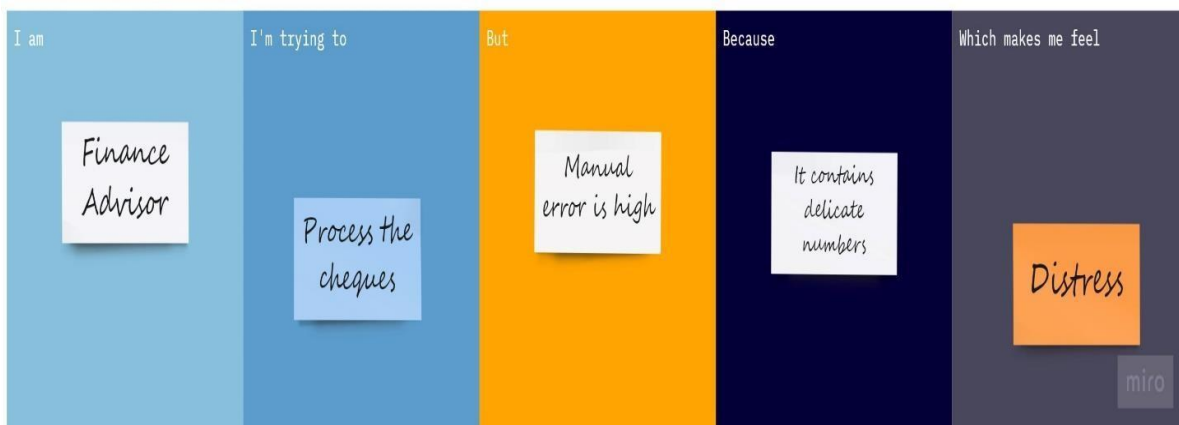
Year : 2020

Authors: Savita Ahlawat , Amit Choudhary , Anand Nayyar , Saurabh Singh and Byungun Yoon

2.3 PROBLEM STATEMENT:

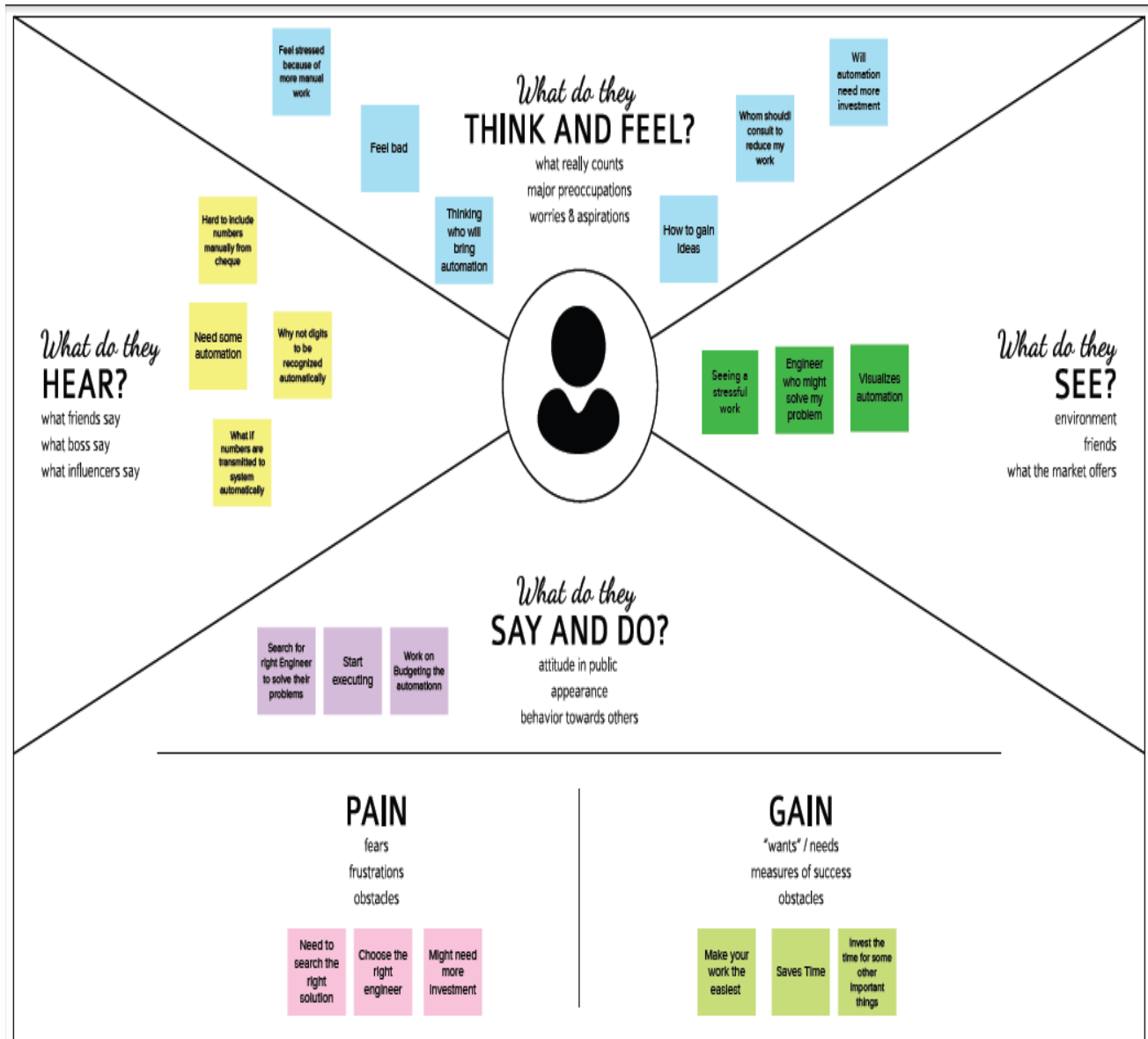
Given a handwritten character, the system needs to predict the type of the character. In other words if we can write the character “6” the system predict the character that it is truly “6” or the input character is nearer to “6” or something else. The purpose of this project is to take the hand written characters as an input process the character, train the neural network effectively by using the algorithm to recognize the pattern.

Customer Problem Statement Template:



3.IDEATION AND PROPOSED SOLUTION

3.1 Empathy Map Canvas:



3.2 Ideation & Brainstorming:

[illegible]

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

 10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Person 1

Reduces Human efforts	A feature to export the recognized text	Can implement Fraud Detection
Scan using mobile phone	Reduces Bugs	Alert user if any digit cannot be scanned
To implement Agile Methodology	Use image or camera as input	Preprocessing and cleaning has to be done

Person 2

Maintains data privacy	Recognize through app	Helps in form data entry
Make application real time	an access anytime at anywhere	Recognize through scanner
Test with different styles of handwriting	Validate and test with custom input	Rich description of digits

Person 3

User Friendly	Provides high quality images of digits	Banks for reading cheques
Identifies spacing differences	Helps in form data entry	Classified into predefined classes
This method involves small numbers of parameters	Train different styles of handwriting	Use for pattern recognition applications

Person 4

High level accuracy	Remove noises to improve the accuracy	Generative models can perform recognition
Reduces time consumption	Generative models can perform recognition	Provides minimal space complexity
Dataset should contain the digits of varying styles	Detection of vehicle number	Use more images for training

3

Group Ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

🕒 20 minutes

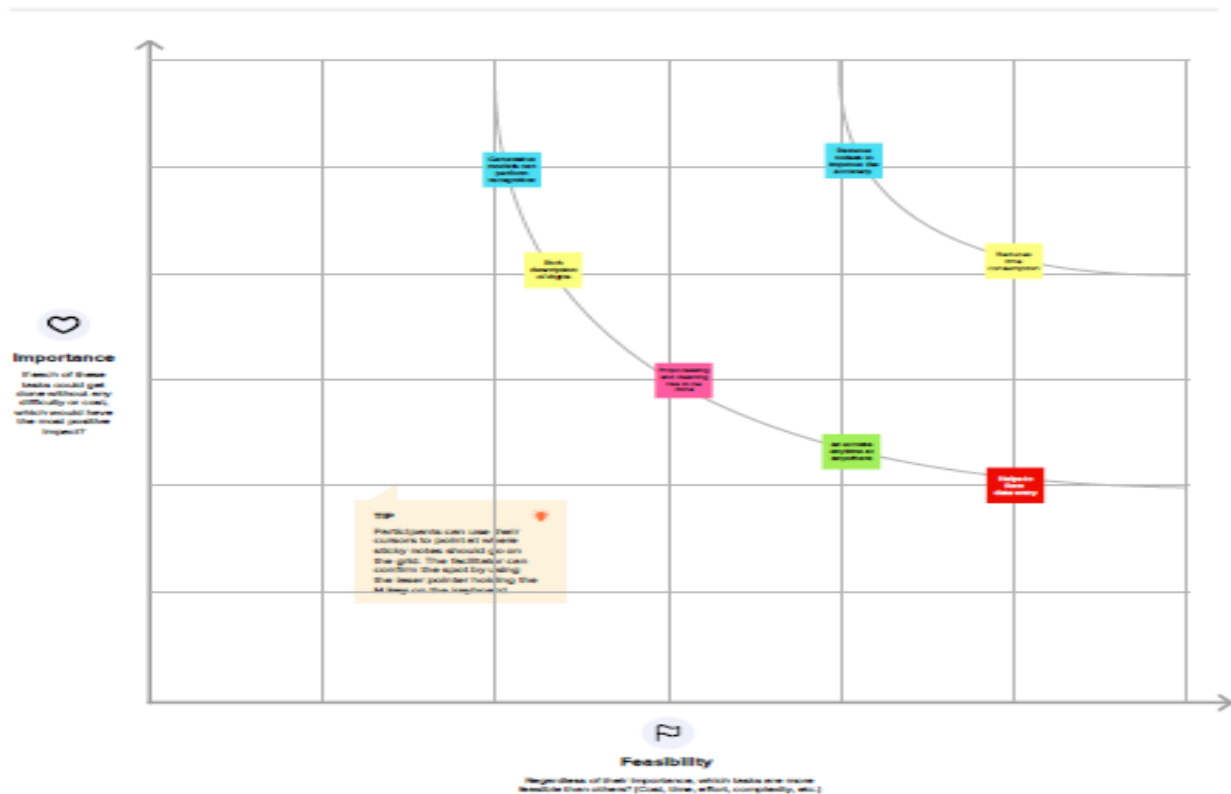


4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

🕒 20 minutes



3.3 Proposed Solution:

S.No	Parameter	Description
1.	Problem Statement (Problem to be solved)	<p>In the modern world, digit recognition is crucial. It is capable of solving increasingly difficult problems and making humans' jobs easier. Handwritten digit recognition is one example. This is a worldwide system for recognizing zip codes or postal codes for mail sorting. Handwritten digit recognition can be accomplished using a variety of approaches. The machine has a difficult duty because handwritten digits are not flawless and can be generated with a variety of flavors. The solution to this issue is handwritten digit recognition, which uses an image of a digit and identifies the digit represented in the image.</p>
2.	Idea / Solution description	<p>Handwritten digit recognition is performed using the MNIST dataset which contains 60,000 training images of handwritten digits from zero to nine and 10,000 images for testing. So, the MNIST dataset has 10 different classes. In this project, we are going to implement a handwritten digit recognition application trained using the Convolutional Neural Networks model. In the end, a GUI is built where the user gives the handwritten digit as input where it is recognized and the result is displayed immediately.</p>
3.	Novelty / Uniqueness	<p>This project introduces an operative strategy for dealing with novelty</p>

		<p>in the handwritten visual recognition domain. A perfect transcription agent would be able to distinguish known and unknown characters in a picture, as well as determine any aesthetic variations that may occur inside or between texts. The existence of novelty has shown to be a major stumbling block for even the most robust machine learning-based algorithms for these activities. Novelty in handwritten papers might include, among other things, a change in the writer, character properties, writing attributes, or overall document appearance. Instead of examining each element separately, we believe that an integrated agent capable of processing known characters and novelties concurrently is a superior technique. The handwritten digit recognition problem can be seen as a subtask of the optical character recognition (OCR) problem</p>
4.	Social Impact / Customer Satisfaction	<p>There are many benefits associated with the handwriting recognition system. In addition to reading postal addresses and bank check amounts, it is also useful for reading forms. Furthermore, it's used in fraud detection because it makes it easy to compare two texts and determine which one is a copy. As a result, this system fulfills customers' expectations, as it is a novel method for recognizing handwritten digits, ensuring high accuracy for the model and meeting all customer expectations. Users will save a lot of time and effort if the system provides various synonyms for the words recognized. Due to the fact that the users in rural areas will be using their own regional language, this proposed system should be able to detect</p>

		those digits as well. As the system is being used in socially crowded places such as banks to check amounts, it should be fast and reliable. As it is designed to solve real-world problems, it should be highly reliable and trustworthy in every way, and users throughout the world should be able to use it effectively.
5.	Business Model (Revenue Model)	A revenue model means understanding how a startup can make money. Our major revenue sources consist of sales, government funds, and public donations. The introduction of novel ideas increases revenue streams, such as introducing gesture or touch features , voice read out of recognised digits, etc..
6.	Scalability of the Solution	One of the approaches to make the handwritten digit recognition system scalable is to make use of cloud-native methods. For example, one of the cloud solutions for making AI scalable is IBM Cloud. IBM Cloud Build helps run and manage AI models, optimize decisions at scale across any cloud. The advantage of using cloud to make solutions scalable is that we can deploy our AI application on the specific cloud environment that best supports our business needs. We can take advantage of built- in security capabilities and AI model monitoring. We can Automate AI lifecycles with ModelOps pipelines, deploy and run models through one-click integration and also prepare and build models visually and programmatically. Looking at these advantages, we can drive better business

		outcomes by optimizing our decisions and also make our solution scalable using cloud.
--	--	---

3.4 Problem Solution Fit:

1.CUSTOMER SEGMENT(S): The Customers who deal with handwritten digits like Banking sectors , schools , colleges , railways , firms , etc.	5. AVAILABLE SOLUTIONS There are no widely used software's to detect handwriting; instead, they check with other people to affirm what number it is.	8. CHANNELS OF BEHAVIOUR Using software that is available on the internet. Obtaining assistance from those nearby in order to recognise the digits written by their customers.
2. JOBS-TO-BE-DONE/PROBLEMS: Handwritten digits can be difficult to understand and interpret at times. It may cause errors when dealing with rough handwriting.	6.CUSTOMER CONSTRAINT(S): They believe that the alternatives will result in errors and faults and will be inconvenient.	9. PROBLEM ROOT CAUSE We face numerous challenges in handwritten number recognition. because of different people's jotting styles and the lack of Optic character recognition This investigation offers an in-depth comparison of various machine literacy and deep literacy
3. TRIGGERS To obtain the numbers accurately and quickly.	7. BEHAVIOUR Finding the best software for detecting accurate digits in a more efficient manner	10. YOUR SOLUTION A solution to this problem is the Handwritten digit recognition system, which uses a picture of a digit and recognises the digit present in the image. Convolutional Neural Network model built with PyTorch and applied to the MNIST dataset to recognise handwritten digits.
4. EMOTIONS :BEFORE/AFTER Feels frustrated and sad when numbers are not entered.		

4. REQUIREMENT AND ANALYSIS:

4.1 Functional Requirement:

FR No.	Sub Requirement (Story / Sub-Task)
FR-1	<p>Image Data: Handwritten digit recognition refers to a computer's capacity to identify human handwritten digits from a variety of sources, such as photographs, documents, touch screens, etc., and categorise them into ten established classifications (0-9).</p> <p>In the realm of deep learning, this has been the subject of countless studies.</p>
FR-2	<p>Website: Web hosting makes the code, graphics, and other items that make up a website accessible online. A server hosts every website you've ever visited. The type of hosting determines how much space is allotted to a website on a server. Shared, dedicated, VPS, and reseller hosting are the four basic varieties.</p>
FR-3	<p>Digit Classifier Model: To train a convolutional network to predict the digit from an image, use the MNIST database of handwritten digits. get the training and validation data first.</p>
FR-4	<p>Cloud: The cloud offers a range of IT services, including virtual storage, networking, servers, databases, and applications. In plain English, cloud computing is described as a virtual platform that enables unlimited storage and access to your data over the internet.</p>
FR-5	<p>Modified National Institute of Standards and Technology dataset: The abbreviation MNIST stands for the MNIST dataset. It is a collection of 60,000 tiny square grayscale photographs, each measuring 28 by 28, comprising handwritten single digits between 0 and 9.</p>

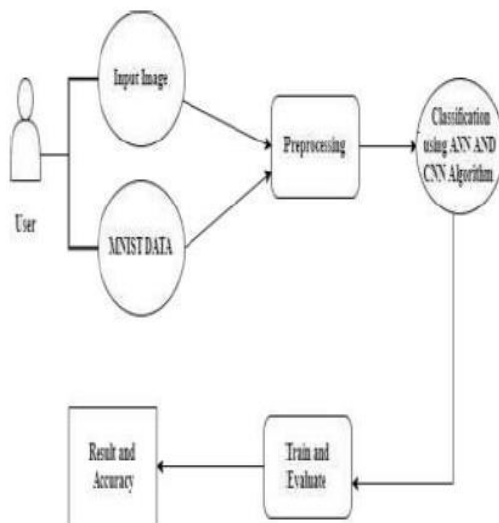
4.2 Non-functional Requirements:

Fr.No.	Non-Functional Requirement	Description
NFR-1	Usability	One of the very significant problems in pattern recognition applications is the recognition of handwritten characters. Applications for digit recognition include filling out forms, processing bank checks, and sorting mail.
NFR-2	Security	<ol style="list-style-type: none">1) The system generates a thorough description of the instantiation parameters, which might reveal information like the writing style, in addition to a categorization of the digit.2) The generative models are capable of segmentation driven by recognition.3) The procedure uses a relatively.
NFR-3	Reliability	The samples are used by the neural network to automatically deduce rules for reading handwritten digits. Furthermore, the network may learn more about handwriting and hence enhance its accuracy by increasing the quantity of training instances. Numerous techniques and algorithms, such as Deep Learning/CNN, SVM, Gaussian Naive Bayes, KNN, Decision Trees, Random Forests, etc., can be used to recognise handwritten numbers.
NFR-4	Accuracy	With typed text in high-quality photos, optical character recognition (OCR) technology offers accuracy rates of

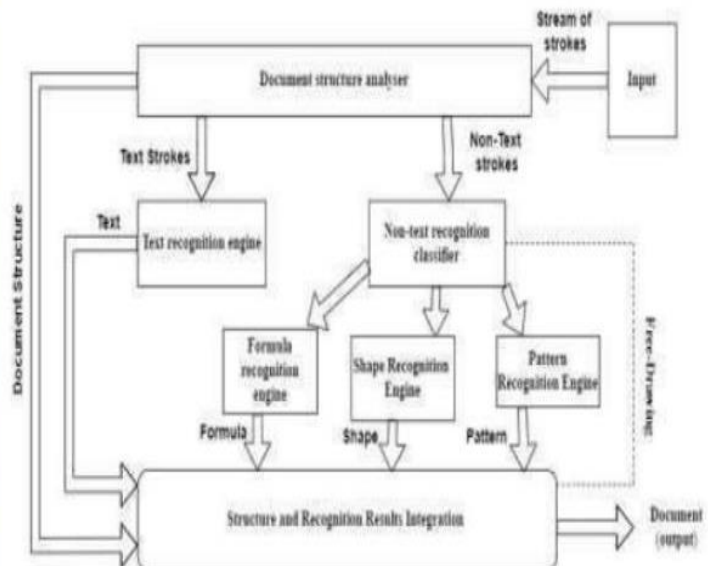
		greater than 99%. However, variances in spacing, abnormalities in handwriting, and the variety of human writing styles result in less precise character identification.
NFR-5	Availability	Work on the Python deep learning project to build a handwritten digit recognition app using MNIST dataset, convolutional neural network and a GUI.

5.PROJECT DESIGN:

5.1 Data Flow Diagrams:



Example: DFD Level 0 (Industry Standard)



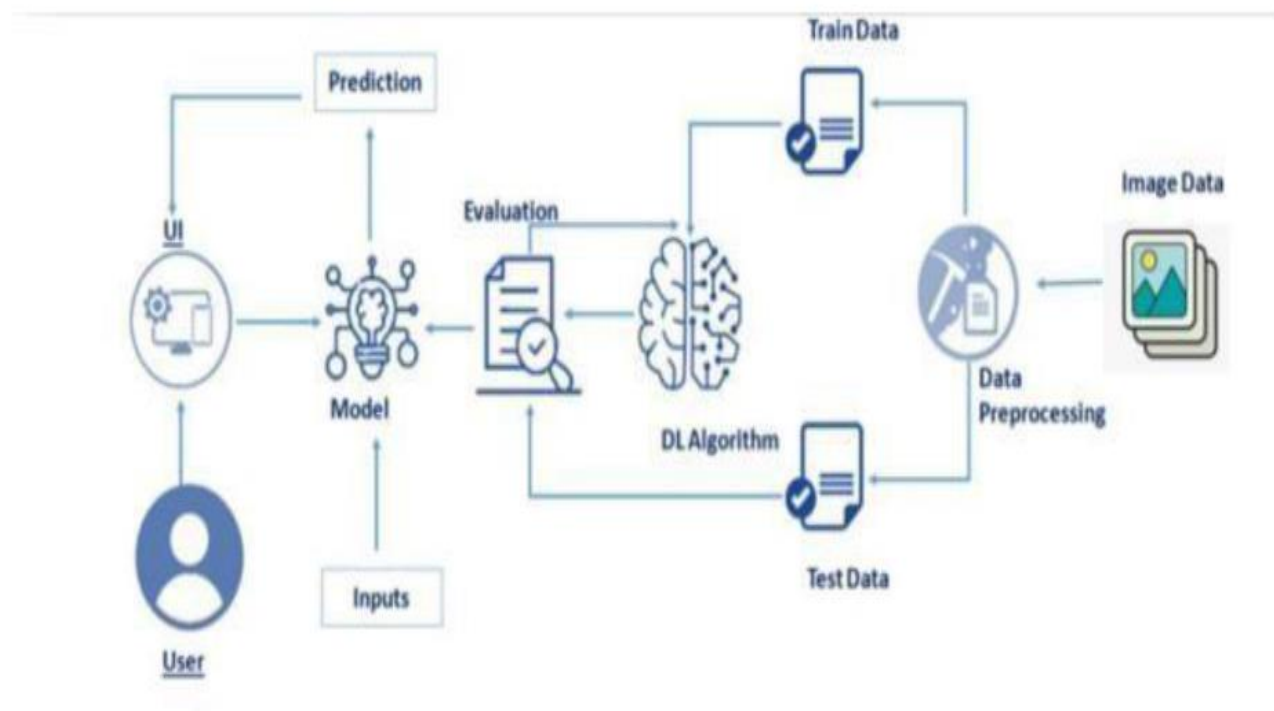
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can register the application with Gmail	Medium	Sprint-2
	Login	USN-5	As a user, I can log into the application by entering email & password	I can login to the application	High	Sprint-1
	Home	USN-6	As a user, I can view the application's home page where I can read the instructions to use this application	I can read instructions also and the home page is user-friendly.	Low	Sprint-1
	Upload Image	USN-7	As a user, I can able to input the images of digital documents to the application	As a user, I can able to input the images of digital documents to the application	High	Sprint-3
	Predict	USN-8	As a user I can able to get the recognised digit as output from the images of digital documents or images	I can access the recognized digits from digital document or images	High	Sprint-3
		USN-9	As a user, I will train and test the input to get the maximum accuracy of output.	I can able to train and test the application until it gets maximum accuracy of the result.	Medium	Sprint-4
Customer (Web user)	Accessibility	USN-10	As a user, I can use the web application virtually anywhere.	I can use the application in any device with a browser	Medium	Sprint-4

5.2 Solution & Technical Architecture:

PROJECT DESCRIPTION:

Given that everyone in the earth has their own writing polished and tasteful form, hand- writing discovery is one of the most making secret designs operation of making observations projects now underway. It is the knowledge processing machine's amount of room to automatically (be conscious) have seen before and get through knowledge of hand-written forms or letters. Because of moves-forward in science and technology, everything is being digitalized in order to get changed to other form to do with man hard work. As an outcome, hand-written any number 0 to 9 seeing who a person is needed in many now applications 1. The MNIST facts group, which has in it 70000 hand-written any numbers 0 to 9, is commonly given work in this wide approval of one's work process. To train these pictures by camera and make come into existence a deep learning design to be copied, we use artificial 2 neural 3 networks 4. A net application 5 is have undergone growth that lets the user to upload an image of a hand- written any number 0 to 9

TECHNICAL ARCHITECTURE:



5.3 User Stories:

Customer experience journey map

Use this framework to better understand customer needs, motivations, and obstacles by illustrating a key scenario or process from start to finish. When possible, use this map to document and summarize interviews and observations with real people rather than relying on your hunches or assumptions.

Created by [personality labs](#)

Product School

100+ templates feedback

Document an existing experience

Narrow your focus to a specific scenario or process within an existing product or service. In the **Steps** row, document the step-by-step process someone typically experiences, then add detail to each of the other rows.

Tip: As you add steps to the experience, note what flows "Next" to add a right dependency or the scenario you are documenting.

	Entice	Enter	Engage	Exit	Extend
Scenario Breaking, learning, attending, and selling a locality tour					
Steps What does the person do (greatly experienced)?	<div>Step 1: Entice</div> <div>Step 2: Enter</div> <div>Step 3: Engage</div>	<div>Step 1: Enter</div> <div>Step 2: Engage</div> <div>Step 3: Exit</div>	<div>Step 1: Engage</div> <div>Step 2: Exit</div> <div>Step 3: Extend</div>	<div>Step 1: Exit</div> <div>Step 2: Extend</div>	<div>Step 1: Extend</div>
Interactions What interactions do they have at each step along the way?	<div>Interaction 1: Entice</div> <div>Interaction 2: Enter</div> <div>Interaction 3: Engage</div>	<div>Interaction 1: Enter</div> <div>Interaction 2: Engage</div>	<div>Interaction 1: Engage</div> <div>Interaction 2: Exit</div>	<div>Interaction 1: Exit</div>	<div>Interaction 1: Extend</div>
Goals & motivations At each step, what is a person's primary goal or motivation? ("Help me..." or "Help me avoid...")	<div>Goal 1: Entice</div> <div>Goal 2: Enter</div> <div>Goal 3: Engage</div>	<div>Goal 1: Enter</div> <div>Goal 2: Engage</div>	<div>Goal 1: Engage</div> <div>Goal 2: Exit</div>	<div>Goal 1: Exit</div>	<div>Goal 1: Extend</div>
Positive moments What steps does a typical person find enjoyable, productive, fun, motivating, delightful, or exciting?	<div>Positive Moment 1: Entice</div> <div>Positive Moment 2: Enter</div> <div>Positive Moment 3: Engage</div>	<div>Positive Moment 1: Enter</div> <div>Positive Moment 2: Engage</div>	<div>Positive Moment 1: Engage</div> <div>Positive Moment 2: Exit</div>	<div>Positive Moment 1: Exit</div>	<div>Positive Moment 1: Extend</div>
Negative moments What steps does a typical person find frustrating, confusing, boring, costly, or time-consuming?	<div>Negative Moment 1: Entice</div> <div>Negative Moment 2: Enter</div> <div>Negative Moment 3: Engage</div>	<div>Negative Moment 1: Enter</div> <div>Negative Moment 2: Engage</div>	<div>Negative Moment 1: Engage</div> <div>Negative Moment 2: Exit</div>	<div>Negative Moment 1: Exit</div>	<div>Negative Moment 1: Extend</div>
Areas of opportunity How might we make each step better? What ideas do we have? What have others suggested?	<div>Area of Opportunity 1: Entice</div> <div>Area of Opportunity 2: Enter</div> <div>Area of Opportunity 3: Engage</div>	<div>Area of Opportunity 1: Enter</div> <div>Area of Opportunity 2: Engage</div>	<div>Area of Opportunity 1: Engage</div> <div>Area of Opportunity 2: Exit</div>	<div>Area of Opportunity 1: Exit</div>	<div>Area of Opportunity 1: Extend</div>

6.PROJECT PLANNING AND SCHEDULING

6.1 Sprint Planning and Estimation:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data Collection	USN-1	As a user, I can collect the dataset from various resources with different handwritings.	10	Medium	Benisha
Sprint-1	Data Preprocessing	USN-2	As a user, I can load the dataset, handling the missing data, scaling and split data into train and test.	10	Medium	<u>Sivamoorthi</u>
Sprint-2	Model Building	USN-3	As a user, I will get an application with ML model which provides high accuracy of recognized handwritten digit.	5	High	<u>Agilan</u>
Sprint-2	Add CNN layers	USN-4	Creating the model and adding the input, hidden, and output layers to it.	5	High	<u>Sowmivan</u>
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2	Compiling the model	USN-5	With both the training data defined and model defined, it's time to configure the learning process.	2	Medium	<u>Sowmivan</u>
Sprint-2	Train & test the model	USN-6	As a user, let us train our model with our image dataset.	6	Medium	<u>Agilan</u>
Sprint-2	Save the model	USN-7	As a user, the model is saved & integrated with an android application or web application in order to predict something.	2	Low	Benisha
Sprint-3	Building UI Application	USN-8	As a user, I will upload the handwritten digit image to the application by clicking a upload button.	5	High	<u>Sivamoorthi</u>
Sprint-3		USN-9	As a user, I can know the details of the fundamental usage of the application.	5	Low	<u>Agilan</u>
Sprint-3		USN-10	As a user, I can see the predicted / recognized digits in the application.	5	Medium	<u>Sivamoorthi</u>
Sprint-4	Train the model on IBM	USN-11	As a user, I train the model on IBM and integrate flask/ <u>Django</u> with scoring end point.	10	High	Benisha
Sprint-4	Cloud Deployment	USN-12	As a user. I can access the web application	10	High	<u>Agilan</u>

Project Tracker, Velocity & Burndown Chart:

JIRA SOFTWARE DASHBOARD OF THE PROJECT:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day).

$$AV = \frac{\text{sprint duration}}{\text{velocity}} = \frac{20}{10} = 2$$

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

<https://www.visual-paradigm.com/scrum/scrum-burndown-chart/> <https://www.atlassian.com/agile/tutorials/burndown-charts>

Reference:

<https://www.atlassian.com/agile/project-management>

<https://www.atlassian.com/agile/tutorials/how-to-do-scrum-with-jira-software> <https://www.atlassian.com/agile/tutorials/epics>

<https://www.atlassian.com/agile/tutorials/sprints>

<https://www.atlassian.com/agile/project-management/estimation>

<https://www.atlassian.com/agile/tutorials/burndown-charts>

<https://projectworlds.in/artificial-intelligence-project-handwritten-digits-recognition>

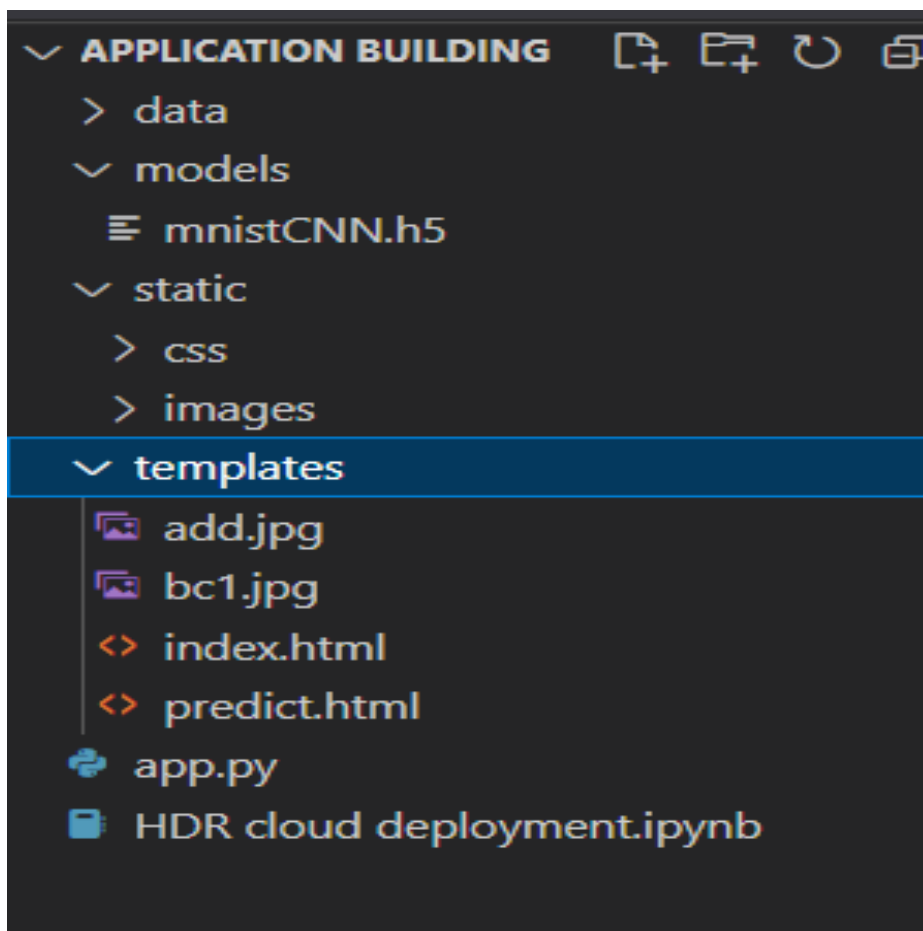
7.CODING AND SOLUTIONING:

This Folder consists of **app.py** file which will apply **CNN** algorithms and **MNIST** dataset. We use **index.html** and **predict.html** where the web application will run and predict the result.

Following steps in jupyter notebook:

1. Understanding the data
2. Model Building
3. Application Building
4. Cloud Delpoyment

Coding Format(Folders):



7.1 Feature 1

index.html:

```
<html>
```

```
<head>
```

```
<title>HDR</title>
```

```
<meta name="viewport" content="width=device-width">
```

```
<link  
href="https://fonts.googleapis.com/css2?family=Prompt:wght@600&display=swap"  
rel="stylesheet">
```

```
<link  
href="https://fonts.googleapis.com/css2?family=Varela+Round&display=swap"  
rel="stylesheet">
```

```
<link  
href="https://fonts.googleapis.com/css2?family=Source+Code+Pro:wght@500&displa  
y=swap" rel="stylesheet">
```

```
<link  
href="https://fonts.googleapis.com/css?family=Calistoga|Josefin+Sans:400,700|Pacif  
ic&display=swap" rel="stylesheet">
```

```
<link rel="stylesheet"  
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"  
integrity="sha384-  
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1  
T" crossorigin="anonymous">
```

```
<link rel="stylesheet" type="text/css" href="{ {  
url_for('static',filename='C:\Users\MYCOMPUTER\Desktop\Application  
Building\static\css\style.css') } }">
```

```
<script src="https://kit.fontawesome.com/b3aed9cb07.js"  
crossorigin="anonymous"></script>
```

```
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-  
q8i/X+965DzOOrT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"  
crossorigin="anonymous"></script>
```

```
<script  
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"  
integrity="sha384-  
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0  
W1" crossorigin="anonymous"></script>
```

```
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"  
integrity="sha384-  
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"  
crossorigin="anonymous"></script>
```

```
<script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@latest"></script>
```

```
<link rel="stylesheet"  
href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/css/bootstrap.min.css">
```

```
<script  
src="https://cdn.jsdelivr.net/npm/jquery@3.6.0/dist/jquery.slim.min.js"></script>
```

```
<script  
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"></script>
```

```
<script  
src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/js/bootstrap.bundle.min.js"></s  
cript>
```

```
</head>
```

```
<style>
```

```
body{
```

```
background-image: url('bc1.jpg');
```

```
background-repeat: no-repeat;
```

```
background-attachment: fixed;
```

```
background-size: cover;
```

```
}
```

```
</style>
```

```
<script>
```

```
function preview() {
```

```
frame.src=URL.createObjectURL(event.target.files[0]);
```

```
}
```

```
$(document).ready(function() {
```

```
$('#clear_button').on('click', function() {
```

```
$('#image').val("");
```

```
$('#frame').attr('src', "");
```

```
});
```

```
});
```

```
</script>
```

```
<body>
```

```
<div class="container">
```

```
  <h1 style="color:red">HandWritten Digit Recognition System</h1>
```

```
</div>
```

```
  <div class="container p-3 my-3 bg-blue text-white">
```

```
    <p>Handwritten Digit Recognition is a technology that is much needed in this world as of Today.This Digit Recognition System is used to recognize the digits from different sources like email, posts, cheque etc. Before proper implementation of this technology we have relied on writing text with our own hands which can result in error.It's difficult to store and access physical data with efficiency.The project presents in representing the recognition of handwritten digits (0 - 9) from the famous MNIST dataset. Here we will be using AlexNet which is an architecture of Convolutional Neural Network.</p>
```

```
  <h4 style="color:red">Now it's time to predict .Choose a image and give a try!</h4>
```

```
  </div>
```

```
    <div class="container">
```

```
      <section id="content">
```

```
        <div class="center">
```

```
          <form action="predict.html" method="POST" enctype="multipart/form-data">
```

```
            <label style="color:white">Select a image:</label>
```

```
            <input id="image" type="file" name="image" accept="image/png, image/jpeg" onchage="preview()"><br><br>
```

```
            <img id="frame" width="150px" height="150px"/>
```

```
            <div class="buttons_div">
```

<button type="submit" class="btn btn-light">Predict</button>

<button type="button" class="btn btn-light"> Clear </button>

</div>

</form>

</div>

</section>

</div>

<!--

<h1 class="welcome">IBM PROJECT

<div id="team_id">TEAM ID : PNT2022TMID42292</div>

</h1>

<section id="title">

<h4 class="heading">Handwritten Digit Recognition Website</h4>

<p>

The website is designed to predict the handwritten digit.

</p>

<p>

The handwritten digit recognition is the ability of computers to recognize human handwritten digits.

Handwriting recognition is one of the compelling research works going on because every individual in this world

has their own style of writing. It is the capability of the computer to identify and understand

handwritten digits or characters automatically. Because of the progress in the field of science and technology,

everything is being digitalized to reduce human effort.</p>

<p> Hence, there comes a need for handwritten digit recognition in many real-time applications.

MNIST data set is widely used for this recognition process and it has 70000 handwritten digits.

We use Artificial neural networks to train these images and build a deep learning model.

Web application is created where the user can upload an image of a handwritten digit.

This image is analyzed by the model and the detected result is returned on to UI</p>

</section>

-->

<!--<section id="content">

<div class="center">


```
<form action="predict.html" method="POST" enctype="multipart/form-data">

<label>Select a image:</label>

<input id="image" type="file" name="image" accept="image/png, image/jpeg"
onchange="preview()"><br><br>

<img id="frame" width="100px" height="100px"/>

<div class="buttons_div">

  <button type="submit" class="btn btn-dark"
id="predict_button">Predict</button>

  <button type="button" class="btn btn-dark" id="clear_button">&nbsp; Clear
&nbsp;</button>

  <button type="submit" class="btn btn-light">Predict</button>

  <button type="button" class="btn btn-light">&nbsp; Clear &nbsp;</button>

</div>

</form>

</div>

</section>-->

</body>

</html>
```

predict.html:

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <title>Prediction</title>
```

```
</head>
```

```
<style>
```

```
  body{
```

```
    background-image: url('bc1.jpg');
```

```
    background-repeat: no-repeat;
```

```
    background-size: cover;
```

```
  }
```

```
  #rectangle{
```

```
    width:400px;
```

```
    height:150px;
```

```
    background-color: #000000;
```

```
    border-radius: 15px;
```

```
    position:absolute;
```

```
    box-shadow: 0px 0px 10px 5px white;
```

```
    top:25%;
```

```
    left:50%;
```

```
transform:translate(-50%,-50%);
```

```
}
```

```
#head{
```

```
text-align: center;
```

```
font-size: 30px;
```

```
margin: 0 auto;
```

```
padding: 3% 5%;
```

```
font-family: Arial, Helvetica, sans-serif;
```

```
color: white;
```

```
}
```

```
#num{
```

```
font-size: 50px;
```

```
}
```

```
marquee{
```

```
font-size:30px;
```

```
font-weight:800;
```

```
color:#8ebf42;
```

```
}
```

```
</style>
```

```
<body>
```

```
<div id="rectangle">
```

```
  <h1 id="head"> <center id="num">The Predicted number is 0</center></h1>
```

```
</div>
```

```
</body>
```

```
<body>
```

```
<a href="index.html">
```

```
<button>Predict Another Number</button>
```

```
</a>
```

```
</body>
```

```
<br>
```

```
<body>
```

```
</html>
```

style.css:

```
#clear_button{
```

```
  margin-left: 15px;
```

```
  font-weight: bold;
```

```
  color: rgb(0, 174, 255);
```

```
}
```

```
#confidence{
```

```
  font-family: 'Josefin Sans', sans-serif;
```

```
margin-top: 7.5%;  
}
```

```
#content{  
margin: 0 auto;  
padding: 2% 15%;  
padding-bottom: 0;  
}
```

```
.welcome{  
text-align: center;  
position: relative;  
color: rgb(0, 32, 112);  
background-color: skyblue;  
padding-top: 1%;  
padding-bottom: 1%;  
font-weight: bold;  
font-family: 'Bookman', 'URW Bookman L', serif;  
}
```

```
#team_id{  
text-align: right;  
font-size: 25px;  
padding-right: 3%;
```

```
}
```

```
#predict_button{  
    margin-right: 15px;  
    color: rgb(0, 255, 72);  
    font-weight: bold;  
}
```

```
#prediction_heading{  
    font-family: 'Josefin Sans', sans-serif;  
    margin-top: 7.5%;  
}
```

```
#result{  
    font-size: 5rem;  
}
```

```
#title{  
    padding: 1.5% 15%;  
    margin: 0 auto;  
    text-align: center;  
}
```

```
.btn {
```

```
font-size: 15px;

padding: 10px;

/* -webkit-appearance: none; */

background: #eee;

border: 1px solid #888;

margin-top: 20px;

margin-bottom: 20px;

}
```

```
.buttons_div{

margin-bottom: 30px;

margin-right: 80px;

}
```

```
.heading{

font-family:"American Typewriter", serif;

font-weight: 700;

font-size: 2rem;

display: inline;

}
```

```
.leftside{

text-align: center;

margin: 0 auto;
```

```
margin-top: 2%;
```

```
/* padding-left: 10%; */
```

```
}
```

```
#frame{
```

```
margin-right: 10%;
```

```
}
```

```
.predicted_answer{
```

```
text-align: center;
```

```
margin: 0 auto;
```

```
padding: 3% 5%;
```

```
padding-top: 0;
```

```
/* padding-left: 10%; */
```

```
}
```

```
h1{
```

```
text-align: center;
```

```
color: aliceblue;
```

```
padding: 100px 50px 65px 100px;
```

```
}
```

```
@media (min-width: 720px) {
```

```
.leftside{
```


$$\}$$

```
model = load_model("./models/mnistCNN.h5")
```

```
@app.route('/')
```

```
def index():
```

```
    return render_template('index.html')
```

```
@app.route('/predict', methods=['GET', 'POST'])
```

```
def upload():
```

```
    if request.method == "POST":
```

```
        f = request.files["image"]
```

```
        filepath = secure_filename(f.filename)
```

```
        f.save(os.path.join(app.config['UPLOAD_FOLDER'], filepath))
```

```
        upload_img = os.path.join(UPLOAD_FOLDER, filepath)
```

```
        img = Image.open(upload_img).convert("L") # convert image to monochrome
```

```
        img = img.resize((28, 28)) # resizing of input image
```

```
        im2arr = np.array(img) # converting to image
```

```
        im2arr = im2arr.reshape(1, 28, 28, 1) # reshaping according to our requirement
```

```
        pred = model.predict(im2arr)
```

```
        num = np.argmax(pred, axis=1) # printing our Labels
```

```
return render_template('predict.html', num=str(num[0]))
```

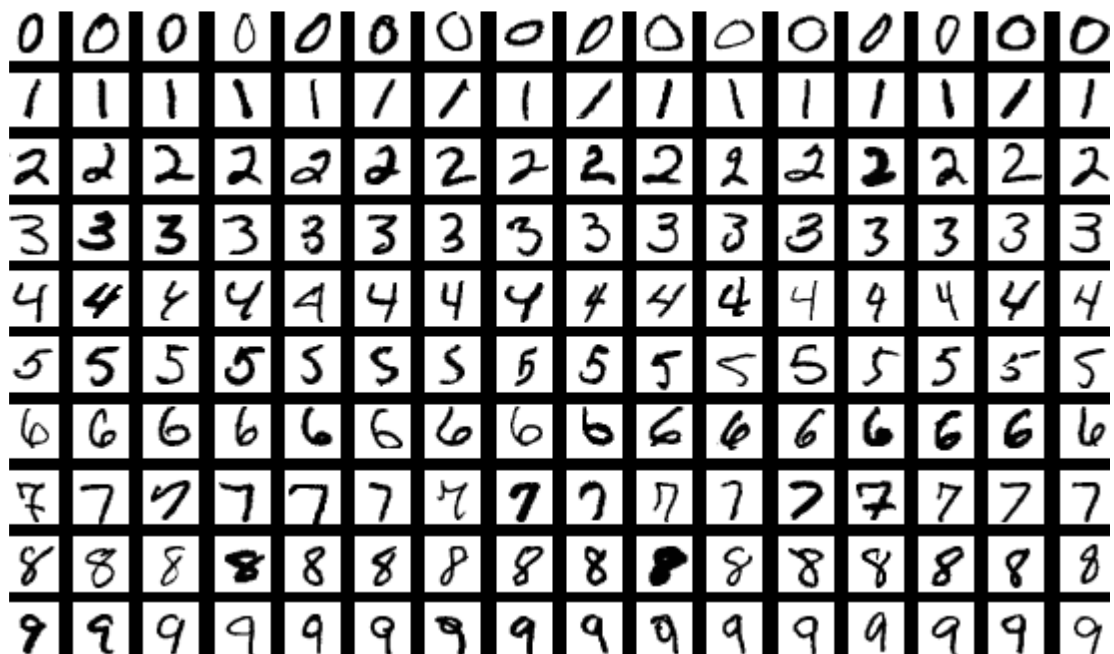
```
if __name__ == '__main__':
```

```
    app.run(debug=True, threaded=False)
```

7.3 Database Schema:

MNIST Dataset Description :

The MNIST hand-written any number 0 to 9 wide approval of one's work knowledge unit includes 60,000 training and 10,000 testing hand-written any number 0 to 9 images. Each image has a high level of 28 bit of picture and a distance from side to side of 28 bit of picture, for a total of 784 (2828) bit of picture. Each bit of picture is connected with a single bit of picture value. It points to how bright or dark that bit of picture is (more complex numbers points to darker bit of picture). This bit of picture value is an integer between 0 and 255.



PROCEDURE:

1. Install the latest TensorFlow library.
2. Prepare the dataset for the model.

3. Develop Single Layer Perceptron model for classifying the handwritten digits.
4. Plot the change in accuracy per epochs.
5. Evaluate the model on the testing data.
6. Analyse the model summary.
7. Add hidden layer to the model to make it Multi-Layer Perceptron.
8. Add Dropout to prevent overfitting and check its effect on accuracy.
9. Increasing the number of Hidden Layer neuron and check its effect on accuracy.
10. Use different optimizers and check its effect on accuracy.
11. Increase the hidden layers and check its effect on accuracy.
12. Manipulate the batch size and epochs and check its effect on accuracy.

MNIST is a dataset which is widely used for handwritten digit recognition. The dataset consists of 60,000 training images and 10,000 test images. The artificial neural networks can all most mimic the human brain and are a key ingredient in image processing field.

Handwritten digit recognition using MNIST dataset is a major project made with the help of Neural Network. It basically detects the scanned images of handwritten digits. We've taken it a step further, and our handwritten digit recognition technology not only recognises scanned images of handwritten numbers, but also allows you to write digits on the screen and have them recognised using an integrated GUI.

Approach:

We will approach this project by using a three-layered Neural Network.

- **The input layer:** It distributes the features of our examples to the next layer for calculation of activations of the next layer.
- **The hidden layer:** They are made of hidden units called activations providing nonlinearities for the network. A number of hidden layers can vary according to our requirements.
- **The output layer:** The nodes here are called output units. It provides us with the final prediction of the Neural Network on the basis of which final predictions can be made.

A neural network is a model based on how the brain functions. It is made up of several layers with numerous activations, which mirror neurons in our brain. A neural network attempts to learn a set of parameters from a set of data, which may aid in recognising underlying links. Because neural networks can adapt to changing input, they can produce the best possible results without having to rethink the output criteria.

METHODOLOGY:

We created a Neural Network with one hidden layer and 100 activation units (excluding bias units). Data is loaded from a .mat file, then features (X) and labels (Y) are extracted. Then, to avoid overflow during computation, features are divided by 255 and rescaled into a range of [0,1]. The data is divided into 60,000 training instances and 10,000 testing examples. Feedforward is used with the training set to calculate the hypothesis, followed by backpropagation to reduce the error between the layers. To solve the issue of overfitting, the regularisation parameter λ is adjusted to 0.1. The optimizer is run 70 times to get the best fit model.

ALGORITHM:

Forward Propagation Architecture:

It is a brief description of how the CNN module will extract features and categorize the image based on them. The network's input layer, hidden layers, and output layer are depicted in the design. The feature extraction phase of the network involves multiple layers, including convolution and resampling.

Explanation of given system:

- The first layer of the architecture is the User layer. User layer will comprise of the people who interacts with the app and for the required results.
- The next three layers is the frontend architecture of the application. The application will be developed using which is the open-source platform for HTML, CSS and JavaScript. The application is deployed in the localhost which is shown on the browser. Through the app, the user will be able to upload pictures of the handwritten digits and convert it into the digitalized form.
- The one in between the database and view layer is the business layer which is the logical calculations on the basis of the request from the client side. It also has the service interface.
- The backend layer consists of two datasets: Training Data and Test Data. The MNIST database has been used for that which is already divided into training set of 60,000 examples and test of 10,000 examples.
- The training algorithm used is Convolution Neural Network. This will prepare the trained model which will be used to classify the digits present in the test data. Thus, we can classify .

WORKING :

- Neural Networks receive an input and transform it through a series of hidden layers.
- Each hidden layer is made up of a set of neurons, where each neuron is fully connected to all neurons in the previous layer.
- Neurons in a single layer function completely independently.
- The last fully connected layer is called the "output layer".

Convolution Layer:

The Convolutional layer is the core building block of a CNN. The layer's parameters consist of a set of learnable filters (or kernels), which have a small receptive field, but extend through the full depth of the input volume. During the forward pass, each filter is convolved across the width and height of the input volume, computing the dot product between the entries of the filter and the input and producing a 2- dimensional activation map of that filter. As a result, the network learns filters that activate when they see some specific type of feature at some spatial position in the input.

Feature Extraction:

All neurons in a feature share the same weights .In this way all neurons detect the same feature at different positions in the input image. Reduce the number of free parameters.

Subsampling Layer:

Subsampling, or down sampling, refers to reducing the overall size of a signal .The subsampling layers reduce the spatial resolution of each feature map. Reduce the effect of noises and shift or distortion invariance is achieved.

Pooling layer:

It is common to periodically insert a Pooling layer in-between successive Conv layer in a Convent architecture. Its function is to progressively reduce the spatial size of the representation to reduce the number of parameters and computation in the network, and hence to also control overfitting. The Pooling Layer operates independently on every depth slice of the input and resizes it spatially, using the MAX operation.

TensorFlow:

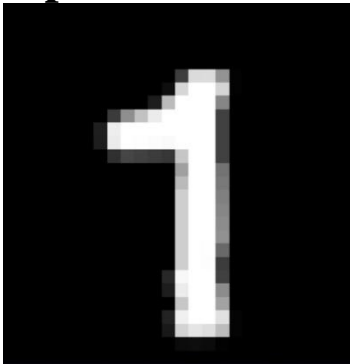
TensorFlow is an open-source machine learning library for research and production. TensorFlow offers APIs for beginners and experts to develop for desktop, mobile, web, and cloud. See the sections below to get started. By scanning the numerical digit and convert into png format using python3 command in terminal we can get text output and sound output.

8.TESTING:

8.1 Test cases:

Test case 1:

Input:

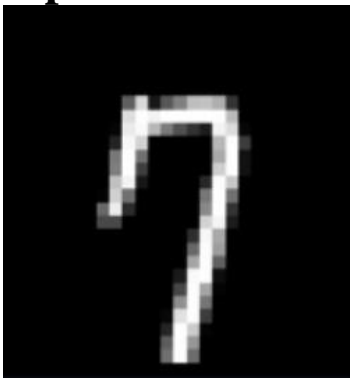


Output:

The predicted number is 1

Test case 2:

Input:

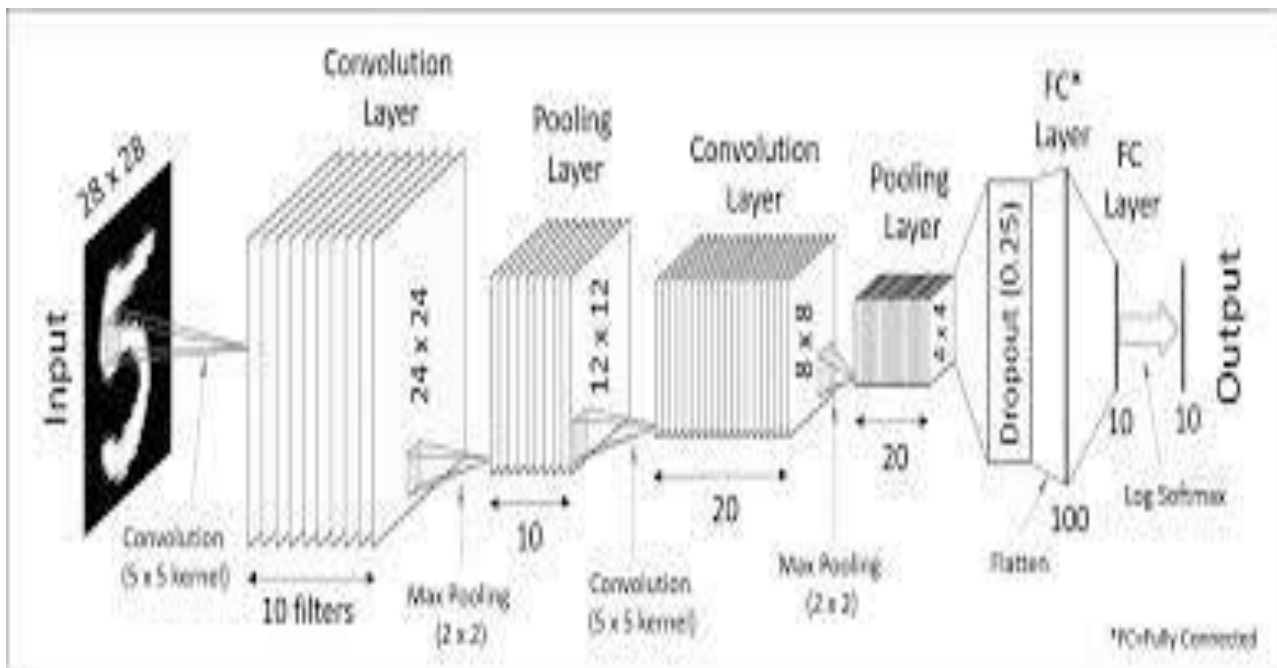


Output:

The predicted number is 7

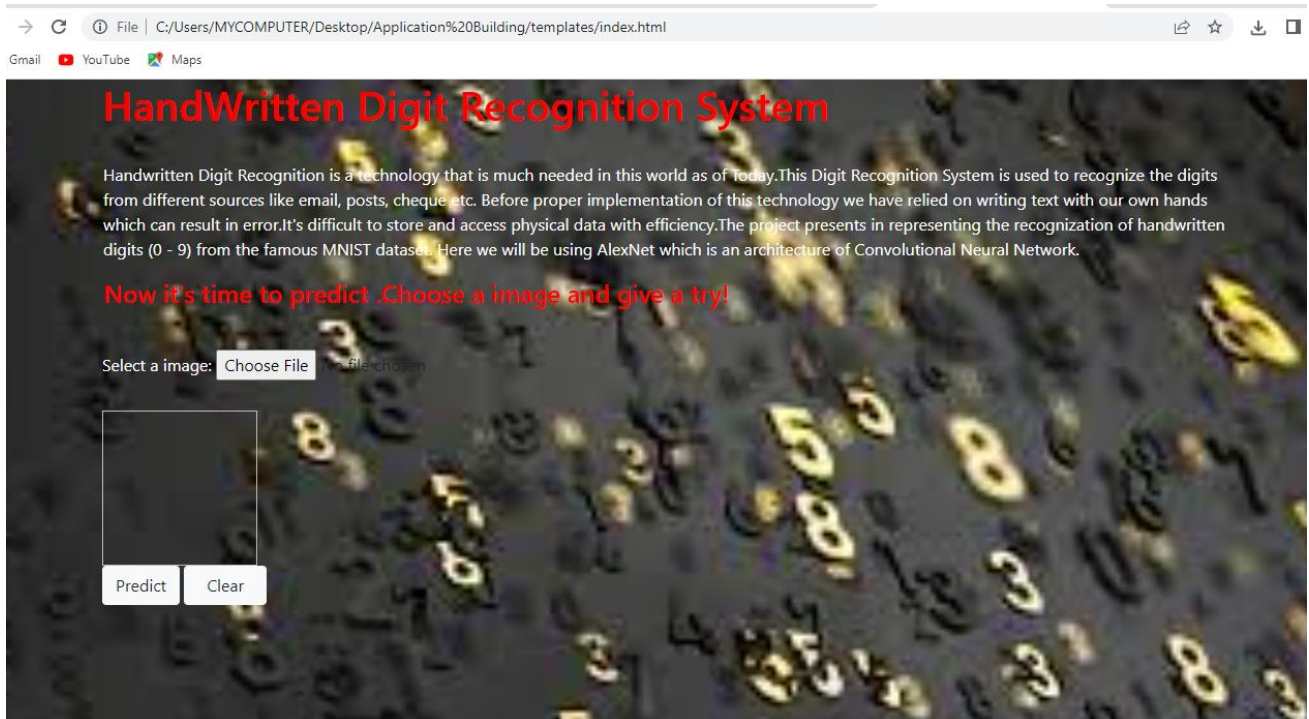
Similarly for all the digits from (0-9), the web application should predict the number given as image input.

8.2 Performance Metrics:

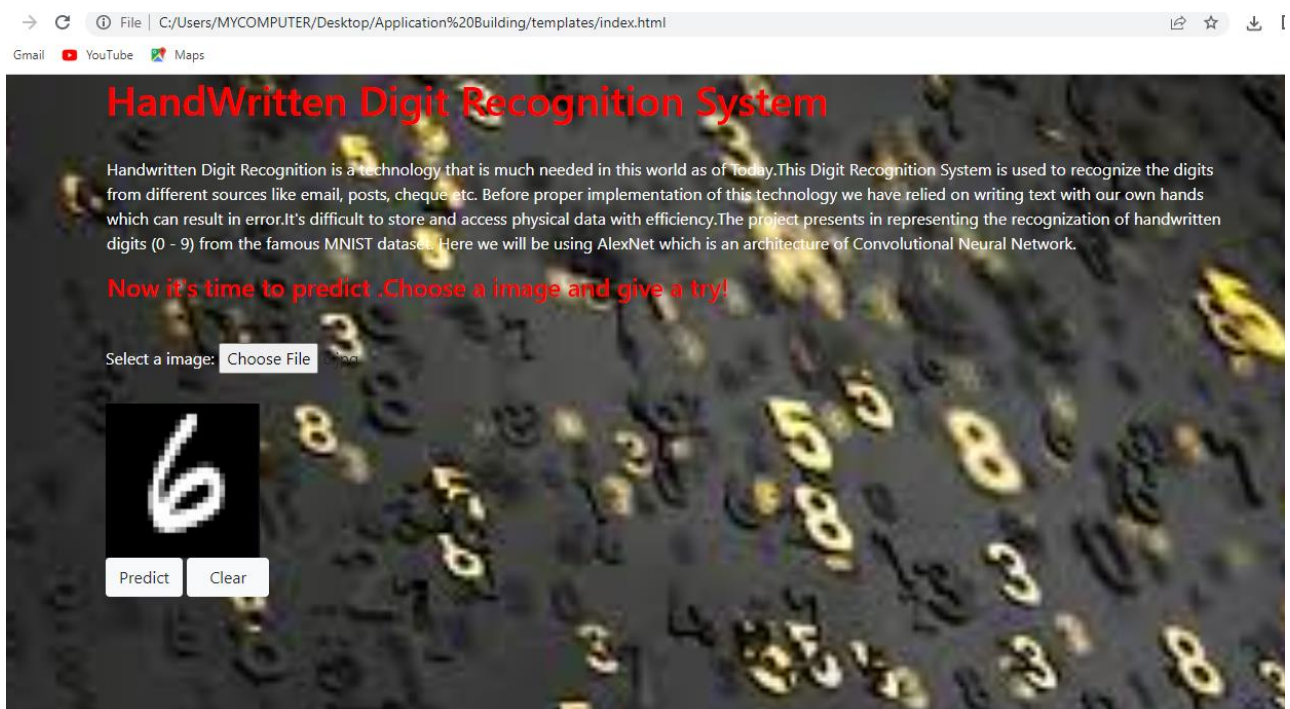


Steps:

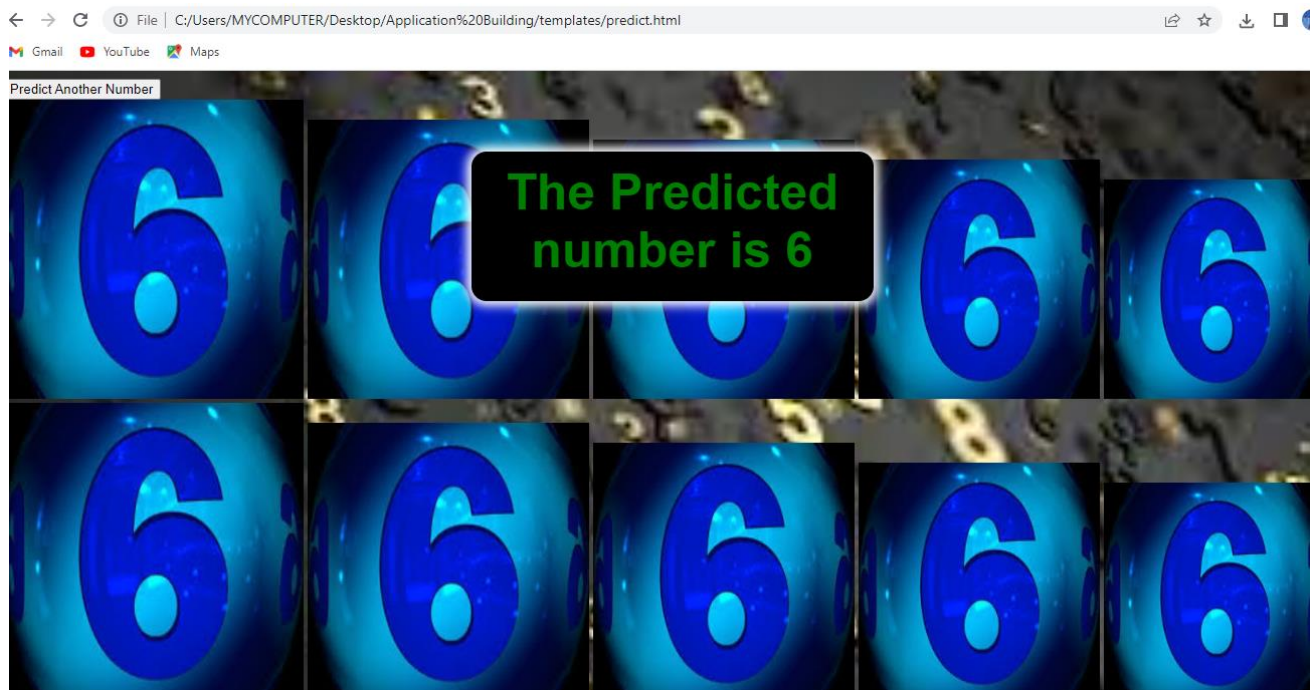
- 1.Import the libraries and load the dataset.
- 2.Preprocess the data.
- 3.Create the model.
- 4.Train the model.
- 5.Evaluate the model.
- 6.Create GUI to predict digits.



Created GUI to predict digit

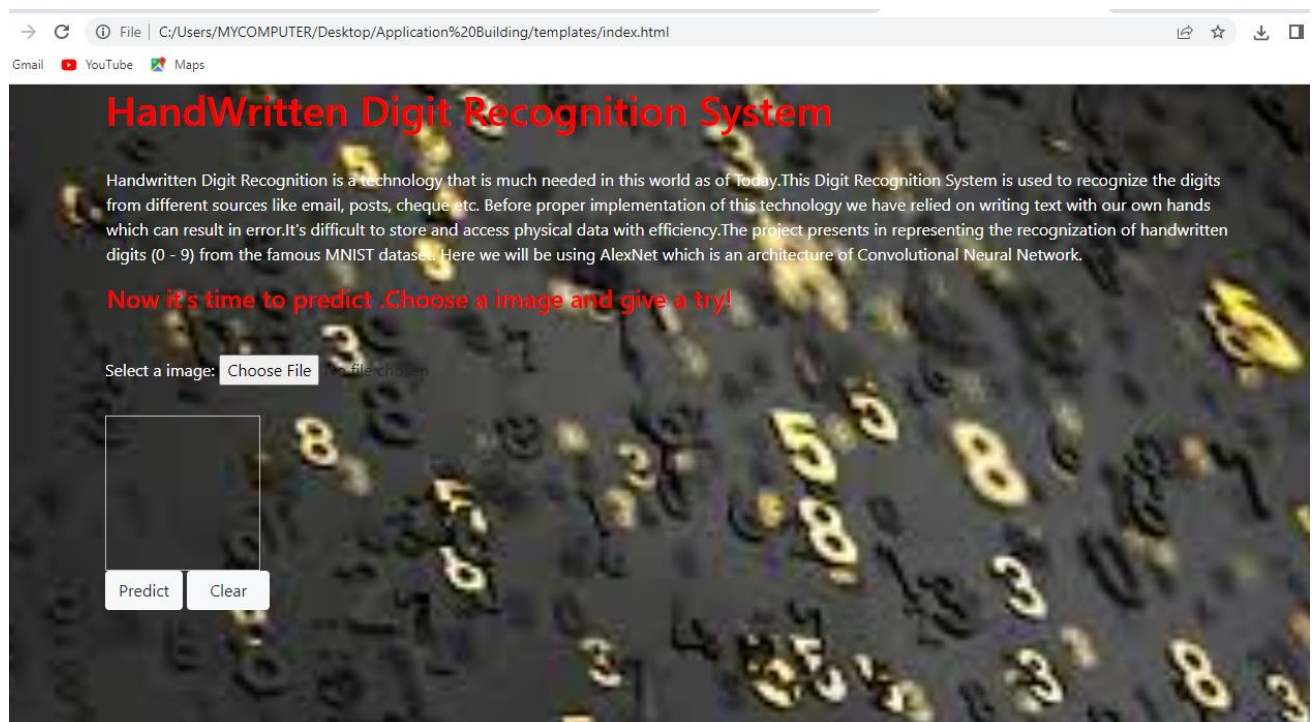


GUI after inserting image of digit 6

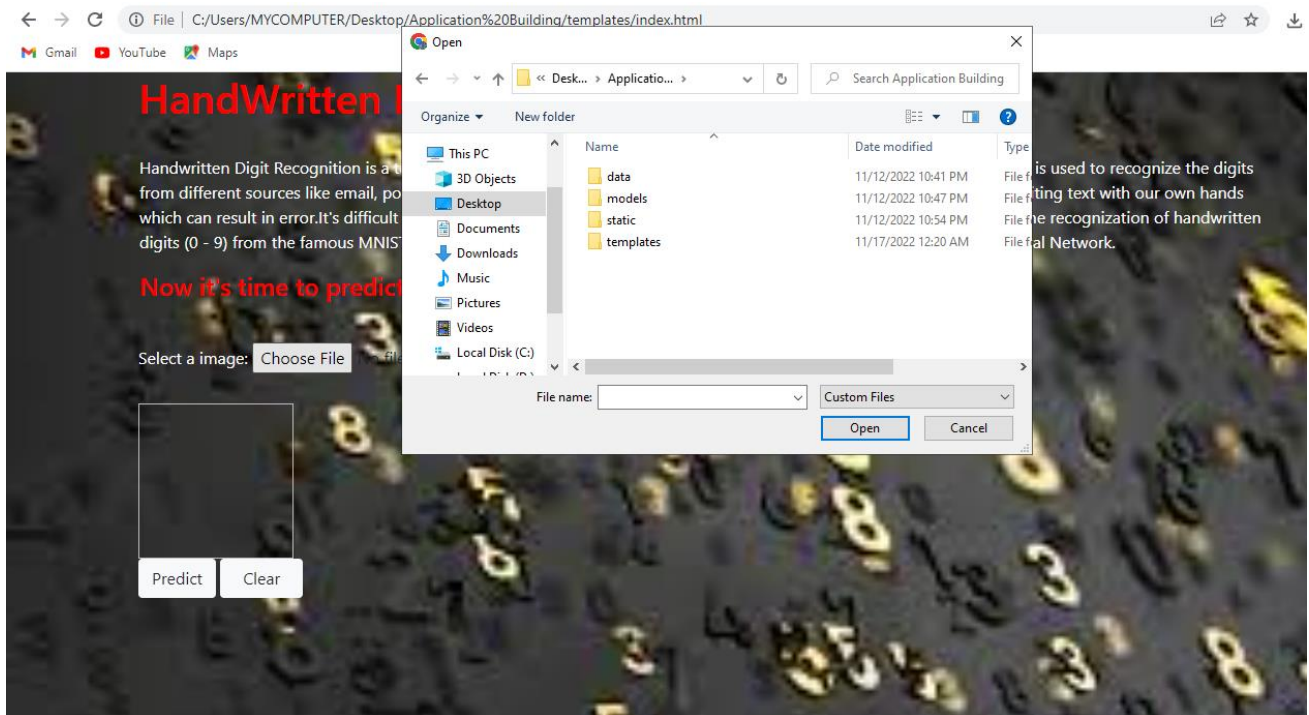


After Predicting the number as 6

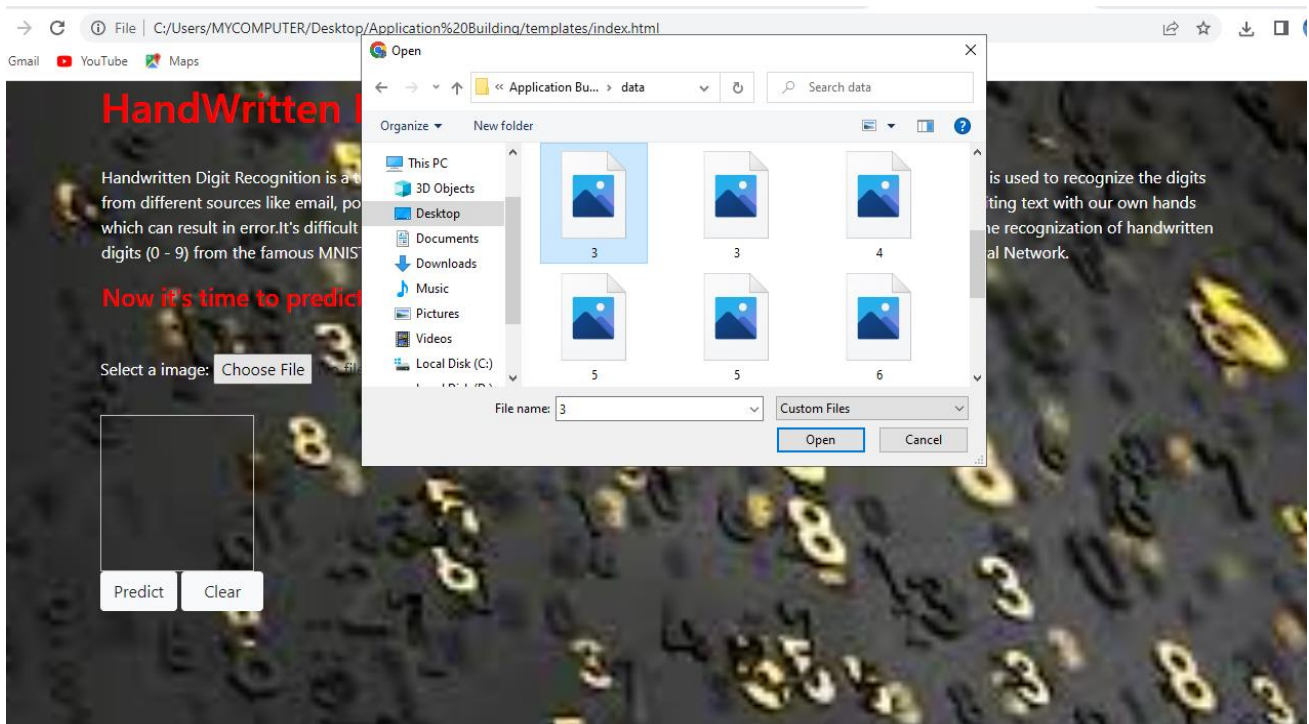
9.RESULTS:



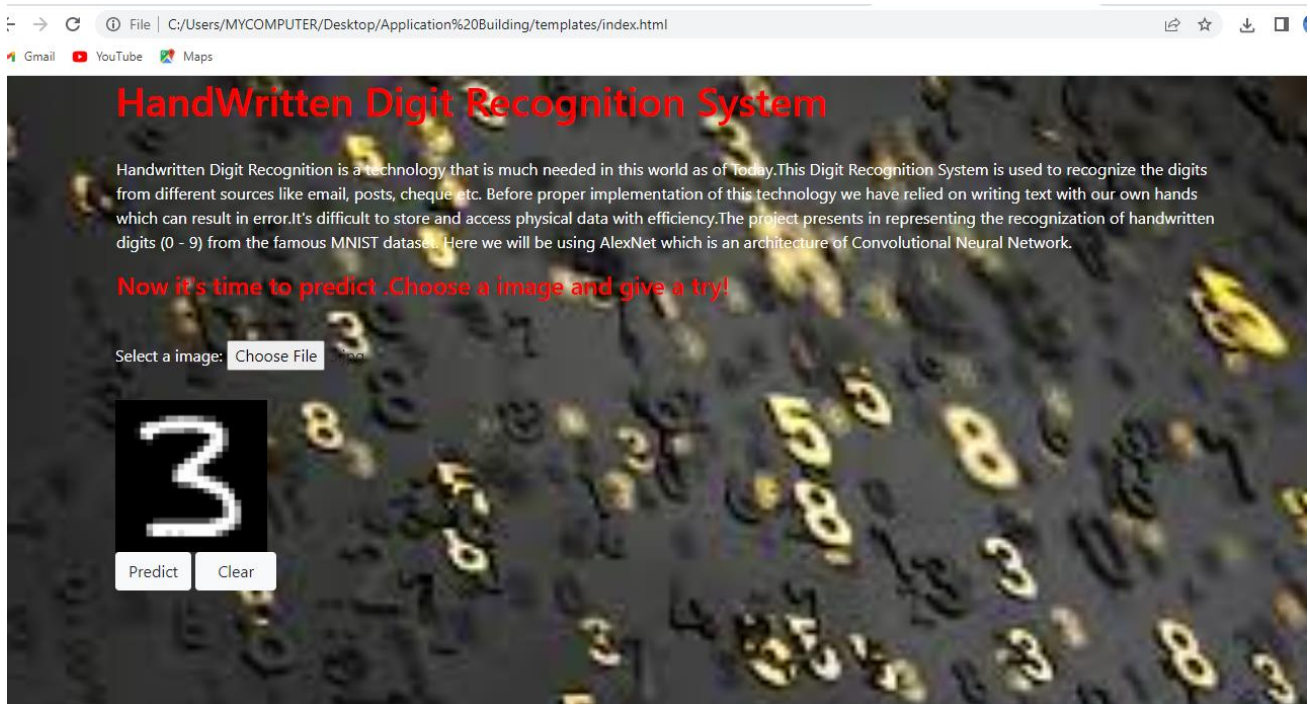
GUI for inserting image of digit:



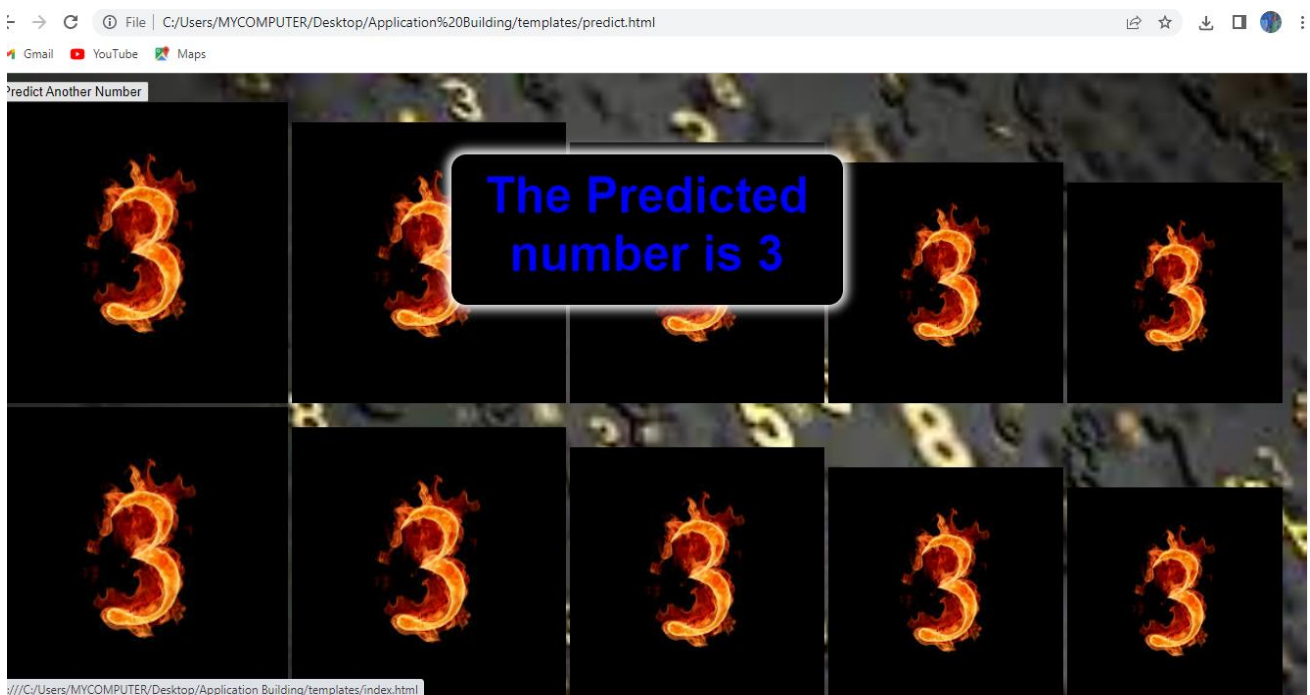
After clicking choose file button



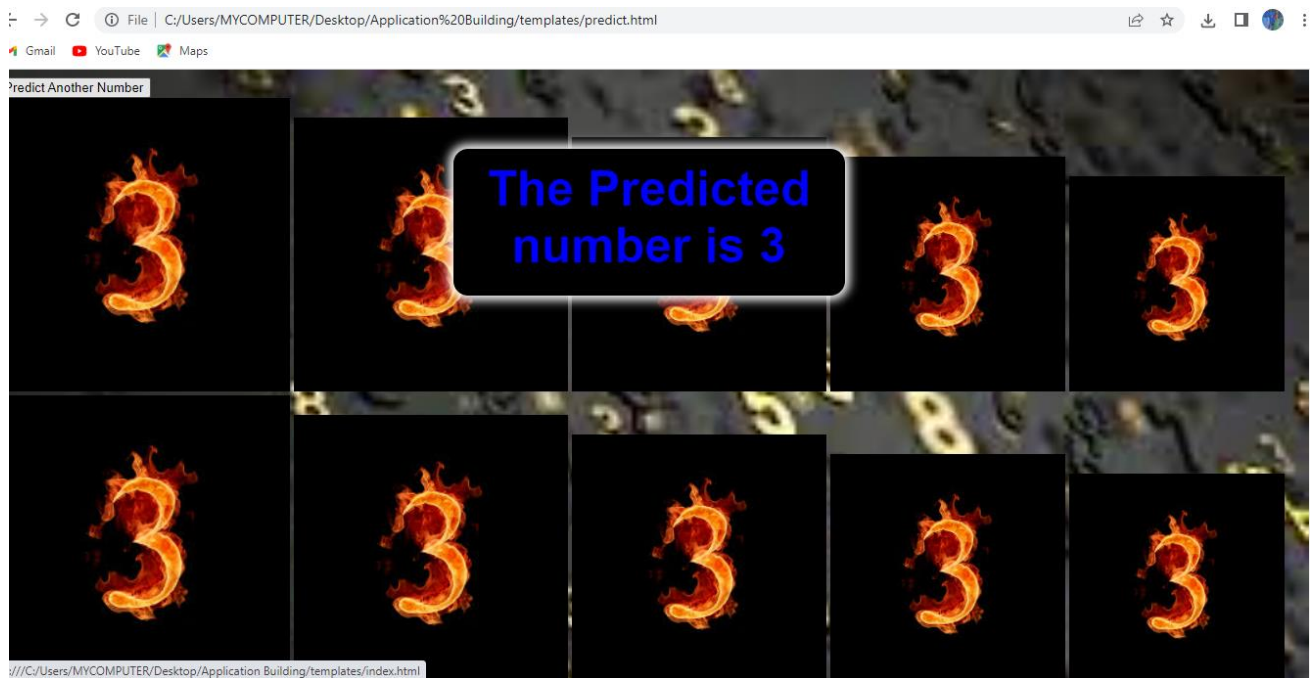
Choosing image of number 3



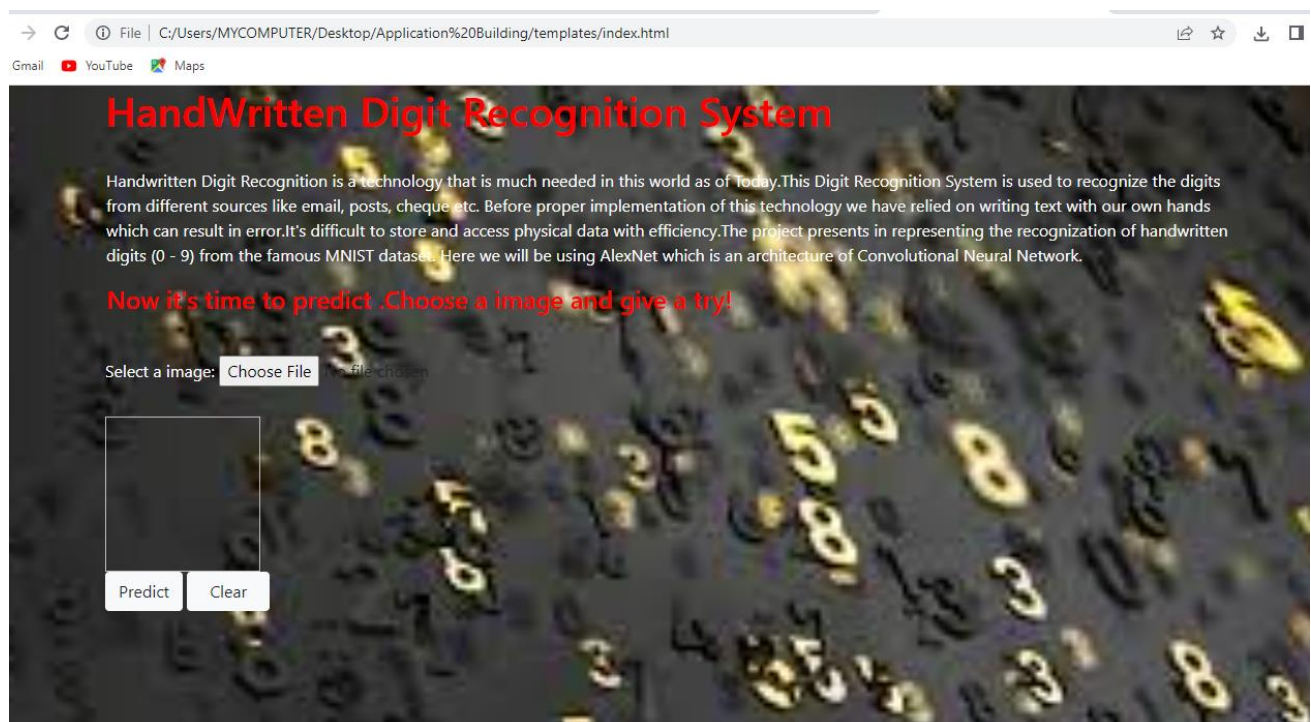
The choosen image is displayed in GUI



The image is predicted as 3 and displayed as 3 (i.e The algorithms written in source code will detect the patterns and determine the numbers)



Click on Predict Another Number Button



After clicking on Predict another number button it will move to a page where you could predict the digits you want

10.ADVANTAGES AND DISADVANTAGES:

Advantages:

- 1)The system not only produces a classification of the digit but also a rich description of the instantiation parameters which can yield information such as the writing style.
- 2) The generative models can perform recognition driven segmentation.
- 3) The applications of digit recognition include in postal mail sorting, bank check processing, form data entry, etc.
- 4) Automatically recognize patterns and regularities in data.

Disadvantages:

- 1) The disadvantage is that it is not done in real time as a person writes and therefore not appropriate for immediate text input.
- 2) Applications of offline handwriting recognition are numerous: reading postal addresses, bank check amounts, and forms.

11.CONCLUSION:

Using Neural Network system, back-propagation learning, to recognize handwritten digits was very successful. An image, which contained 100 samples of each number, was trained and tested. The accuracy rate of recognizing the number was 99%. This accuracy rate is very high. From the training and testing results, it was concluded that the system had more trouble identifying numeral —5“. This maybe caused by the fact that the digit is running together or maybe it is not fully connected. The system was not stable. It gave different training and testing results every day for each numeral. It will need to take a close look at the system and should look for improvements for the future. From the net-file, the system was able to produce an image-file. The image-file produced showed the recognized number. By looking at figure 5.2, it is concluded that the image-file produced does not show the numeral —5“ clear enough. This part will also need more improvements. Apart from the above problems and parts that need improvements, the overall recognition system was successful.

12.FUTURE SCOPE:

The task of handwritten digit recognition, using a classifier, has great importance and use such as – online handwriting recognition on computer tablets, recognize zip codes on mail for postal mail sorting, processing bank check amounts, numeric entries in forms filled up by hand (for example - tax forms) and so on.

13.APPENDIX: Source Code

index.html:

```
<html>
```

```
<head>
```

```
<title>HDR</title>
```

```
<meta name="viewport" content="width=device-width">
```

```
<link  
href="https://fonts.googleapis.com/css2?family=Prompt:wght@600&display=swap"  
rel="stylesheet">
```

```
<link  
href="https://fonts.googleapis.com/css2?family=Varela+Round&display=swap"  
rel="stylesheet">
```

```
<link  
href="https://fonts.googleapis.com/css2?family=Source+Code+Pro:wght@500&displa  
y=swap" rel="stylesheet">
```

```
<link
href="https://fonts.googleapis.com/css?family=Calistoga|Josefin+Sans:400,700|Pacifico&display=swap" rel="stylesheet">
```

```
<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">
```

```
<link rel="stylesheet" type="text/css" href="{ {
url_for('static',filename='C:\Users\MYCOMPUTER\Desktop\Application
Building\static\css\style.css') } }">
```

```
<script src="https://kit.fontawesome.com/b3aed9cb07.js"
crossorigin="anonymous"></script>
```

```
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
```

```
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>
```

```
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
```

```
<script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@latest"></script>
```

```
<link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/css/bootstrap.min.css">
```



```
<script  
src="https://cdn.jsdelivr.net/npm/jquery@3.6.0/dist/jquery.slim.min.js"></script>
```

```
<script  
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"></script>
```

```
<script  
src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/js/bootstrap.bundle.min.js"></s  
cript>
```

```
</head>
```

```
<style>
```

```
body{  
  
    background-image: url('bc1.jpg');  
  
    background-repeat: no-repeat;  
  
    background-attachment: fixed;  
  
    background-size: cover;  
  
}
```

```
</style>
```

```
<script>
```

```
function preview() {  
  
    frame.src=URL.createObjectURL(event.target.files[0]);  
  
}
```

```
$(document).ready(function() {  
  
    $('#clear_button').on('click', function() {
```

```
$('#image').val("");  
$('#frame').attr('src','');  
});  
});
```

```
</script>
```

```
<body>
```

```
<div class="container">
```

```
<h1 style="color:red">HandWritten Digit Recognition System</h1>
```

```
</div>
```

```
<div class="container p-3 my-3 bg-blue text-white">
```

```
<p>Handwritten Digit Recognition is a technology that is much needed in this world as of Today.This Digit Recognition System is used to recognize the digits from different sources like email, posts, cheque etc. Before proper implementation of this technology we have relied on writing text with our own hands which can result in error.It's difficult to store and access physical data with efficiency.The project presents in representing the recognition of handwritten digits (0 - 9) from the famous MNIST dataset. Here we will be using AlexNet which is an architecture of Convolutional Neural Network.</p>
```

```
<h4 style="color:red">Now it's time to predict .Choose a image and give a try!</h4>
```

```
</div>
```

```
<div class="container">
```

```
<section id="content">
```

```
<div class="center">
```

```
<form action="predict.html" method="POST" enctype="multipart/form-data">
  <label style="color:white">Select a image:</label>
  <input id="image" type="file" name="image" accept="image/png, image/jpeg"
onchange="preview()"><br><br>
  <img id="frame" width="150px" height="150px"/>
  <div class="buttons_div">
    <button type="submit" class="btn btn-light">Predict</button>
    <button type="button" class="btn btn-light">&nbsp; Clear &nbsp;</button>
  </div>
</form>
</div>
```

```
</section>
```

```
</div>
```

```
<!--
```

```
<h1 class="welcome">IBM PROJECT
```

```
<div id="team_id">TEAM ID : PNT2022TMID42292</div>
```

```
</h1>
```

```
<section id="title">
```

```
<h4 class="heading">Handwritten Digit Recognition Website</h4>
```

```
<br><br>
```

```
<p>
```

The website is designed to predict the handwritten digit.

```
</p>
```

<p>

The handwritten digit recognition is the ability of computers to recognize human handwritten digits.

Handwriting recognition is one of the compelling research works going on because every individual in this world

has their own style of writing. It is the capability of the computer to identify and understand

handwritten digits or characters automatically. Because of the progress in the field of science and technology,

everything is being digitalized to reduce human effort.</p>

<p> Hence, there comes a need for handwritten digit recognition in many real-time applications.

MNIST data set is widely used for this recognition process and it has 70000 handwritten digits.

We use Artificial neural networks to train these images and build a deep learning model.

Web application is created where the user can upload an image of a handwritten digit.

This image is analyzed by the model and the detected result is returned on to UI</p>

</section>

-->

```
<!--<section id="content">

    <div class="center">

        <form action="predict.html" method="POST" enctype="multipart/form-data">

            <label>Select a image:</label>

            <input id="image" type="file" name="image" accept="image/png, image/jpeg"
onchange="preview()"><br><br>

            <img id="frame" width="100px" height="100px"/>

            <div class="buttons_div">

                <button type="submit" class="btn btn-dark"
id="predict_button">Predict</button>

                <button type="button" class="btn btn-dark" id="clear_button">&nbsp; Clear
&nbsp;</button>

                <button type="submit" class="btn btn-light">Predict</button>

                <button type="button" class="btn btn-light">&nbsp; Clear &nbsp;</button>

            </div>

        </form>

    </div>

</section>-->

</body>

</html>
```

predict.html:

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <title>Prediction</title>
```

```
</head>
```

```
<style>
```

```
  body{
```

```
    background-image: url('bc1.jpg');
```

```
    background-repeat: no-repeat;
```

```
    background-size: cover;
```

```
  }
```

```
  #rectangle{
```

```
    width:400px;
```

```
    height:150px;
```

```
    background-color: #000000;
```

```
    border-radius: 15px;
```

```
position:absolute;
box-shadow: 0px 0px 10px 5px white;
top:25%;
left:50%;
transform:translate(-50%,-50%);
}
```

```
#head{
text-align: center;
font-size: 30px;
margin: 0 auto;
padding: 3% 5%;
font-family: Arial, Helvetica, sans-serif;
color: white;
}
```

```
#num{
    font-size: 50px;
}
marquee{
font-size:30px;
font-weight:800;
color:#8ebf42;
}
```

```
</style>
```

```
<body>
```

```
  <div id="rectangle">
```

```
    <h1 id="head"> <center id="num">The Predicted number is 0</center></h1>
```

```
  </div>
```

```
</body>
```

```
<body>
```

```
<a href="index.html">
```

```
<button>Predict Another Number</button>
```

```
</a>
```

```
</body>
```

```
<br>
```

```
<body>
```

```
</html>
```

style.css:

```
#clear_button{
```

```
  margin-left: 15px;
```

```
  font-weight: bold;
```

```
  color: rgb(0, 174, 255);
```



```
}
```

```
#confidence{
```

```
    font-family: 'Josefin Sans', sans-serif;
```

```
    margin-top: 7.5%;
```

```
}
```

```
#content{
```

```
    margin: 0 auto;
```

```
    padding: 2% 15%;
```

```
    padding-bottom: 0;
```

```
}
```

```
.welcome{
```

```
    text-align: center;
```

```
    position: relative;
```

```
    color: rgb(0, 32, 112);
```

```
    background-color: skyblue;
```

```
    padding-top: 1%;
```

```
    padding-bottom: 1%;
```

```
    font-weight: bold;
```

```
    font-family: 'Bookman', 'URW Bookman L', serif;
```

```
}
```

```
#team_id{  
    text-align: right;  
    font-size: 25px;  
    padding-right: 3%;  
}
```

```
#predict_button{  
    margin-right: 15px;  
    color: rgb(0, 255, 72);  
    font-weight: bold;  
}
```

```
#prediction_heading{  
    font-family: 'Josefin Sans', sans-serif;  
    margin-top: 7.5%;  
}
```

```
#result{  
    font-size: 5rem;  
}
```

```
#title{  
    padding: 1.5% 15%;  
    margin: 0 auto;
```

```
text-align: center;
}
```

```
.btn {
  font-size: 15px;
  padding: 10px;
  /* -webkit-appearance: none; */
  background: #eee;
  border: 1px solid #888;
  margin-top: 20px;
  margin-bottom: 20px;
}
```

```
.buttons_div{
  margin-bottom: 30px;
  margin-right: 80px;
}
```

```
.heading{
  font-family:"American Typewriter", serif;
  font-weight: 700;
  font-size: 2rem;
  display: inline;
}
```

```
.leftside{  
    text-align: center;  
    margin: 0 auto;  
    margin-top: 2%;  
    /* padding-left: 10%; */  
}
```

```
#frame{  
    margin-right: 10%;  
}
```

```
.predicted_answer{  
    text-align: center;  
    margin: 0 auto;  
    padding: 3% 5%;  
    padding-top: 0;  
    /* padding-left: 10%; */  
}
```

```
h1{  
    text-align: center;  
    color: aliceblue;  
    padding: 100px 50px 65px 100px;
```

```
}
```

```
@media (min-width: 720px) {
```

```
  .leftside{
```

```
    padding-left: 10%;
```

```
  }
```

```
}
```

app.py:

```
import numpy as np
```

```
import os
```

```
from PIL import Image
```

```
from flask import Flask, request, render_template, url_for
```

```
from werkzeug.utils import secure_filename, redirect
```

```
#from gevent.pywsgi import WSGIServer
```

```
from keras.models import load_model
```

```
from keras.preprocessing import image
```

```
from flask import send_from_directory
```

```
UPLOAD_FOLDER = 'D:/ibm/data'
```

```
app = Flask(__name__)
```

```
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
```

```
model = load_model("./models/mnistCNN.h5")
```

```
@app.route('/')
```

```
def index():
```

```
    return render_template('index.html')
```

```
@app.route('/predict', methods=['GET', 'POST'])
```

```
def upload():
```

```
    if request.method == "POST":
```

```
        f = request.files["image"]
```

```
        filepath = secure_filename(f.filename)
```

```
        f.save(os.path.join(app.config['UPLOAD_FOLDER'], filepath))
```

```
        upload_img = os.path.join(UPLOAD_FOLDER, filepath)
```

```
        img = Image.open(upload_img).convert("L") # convert image to monochrome
```

```
        img = img.resize((28, 28)) # resizing of input image
```

```
        im2arr = np.array(img) # converting to image
```

```
        im2arr = im2arr.reshape(1, 28, 28, 1) # reshaping according to our requirement
```

```
        pred = model.predict(im2arr)
```

```
num = np.argmax(pred, axis=1) # printing our Labels
```

```
return render_template('predict.html', num=str(num[0]))
```

```
if __name__ == '__main__':
```

```
    app.run(debug=True, threaded=False)
```

GitHub & Project Demo Link:

Github link:

<https://github.com/IBM-EPBL/IBM-Project-31973-1660207128>

Project Demo Link:

<https://drive.google.com/file/d/1WNJbbC1jgrG8yKcue5Z2uqK1dsVL1S9m/view?usp=drivesdk>