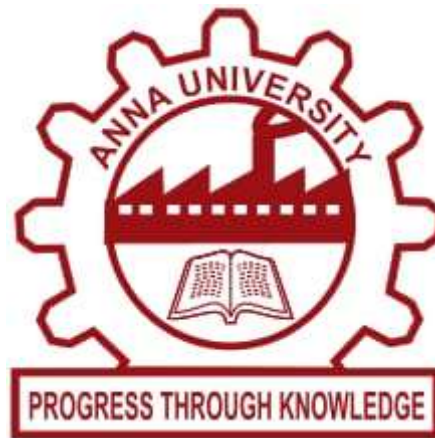


**UNIVERSITY COLLEGE OF ENGINEERING
VILLUPURAM**



**A Novel Method for Handwritten Digit
Recognition System**

SUBMITTED BY

ILAKIYA.M - 422519106015

KALAISELVI.G - 422519106017

PRABAVATHI.S - 422519106029

DHARANI.E - 422519106007

TEAM ID - PNT2022TMID29320

INTRODUCTION 1

1 PROJECT OVERVIEW

1 PURPOSE

1. LITERATURE SURVEY 2

2 EXISTING PROBLEM

2 REFERENCES

5 PROBLEM STATEMENT DEFINITION

2. IDEATION AND PROPOSED SOLUTION 6

6 EMPATHY MAP CANVAS

7 IDEATION & BRAINSTORMING

8 PROPOSED SOLUTION

9 PROBLEM SOLUTION FIT

3. REQUIREMENT ANALYSIS 10

10 FUNCTIONAL REQUIREMENTS

11	NON FUNCTIONAL REQUIREMENTS	
4.	PROJECT DESIGN	12
12	DATA FLOW DIAGRAM	
13	SOLUTION & TECHNICAL ARCHITECTURE	
15	USER STORIES	
5.	PROJECT PLANNING AND SCHEDULING	16
16	SPRINT PLANNING AND ESTIMATION	
17	SPRINT DELIVERY SCHEDULE	
6.	CODING & SOLUTIONING	18
7.	TESTING	20
20	TEST CASES	
22	USER ACCEPTANCE TESTING	
22	DEFECT ANALYSIS	
22	TEST CASE ANALYSIS	
8.	RESULTS	23
23	PERFORMANCE METRICS	
9.	ADVANTAGES & DISADVANTAGES	25

	ADVANTAGES	
25		
	DISADVANTAGES	
25		
	10. CONCLUSION	26
	11. FUTURE SCOPE	27
	APPENDIX	
28		
	SOURCE CODE	
28		
	GITHUB	
37		
	PROJECT DEMO	
37		

CHAPTER 1

INTRODUCTION

PROJECT OVERVIEW

Machine learning and deep learning play an important role in computer technology and artificial intelligence. With the use of deep learning and machine learning, human effort can be reduced in recognizing, learning, predictions and in many more areas.

Handwritten Digit Recognition is the ability of computer systems to recognise handwritten digits from various sources, such as images, documents, and so on. This project aims to let users take advantage of machine learning to reduce manual tasks in recognizing digits.

PURPOSE

Digit recognition systems are capable of recognizing the digits from different sources like emails, bank cheque, papers, images, etc. and in different real-world scenarios for online handwriting recognition on computer tablets or system, recognize number plates of vehicles, processing bank cheque amounts, numeric entries in forms filled up by hand (tax forms) and so on.

CHAPTER 2

LITERATURE SURVEY

EXISTING PROBLEM

The fundamental problem with handwritten digit recognition is that handwritten digits do not always have the same size, width, orientation, and margins since they vary from person to person. Additionally, there would be issues with identifying the numbers because of similarities between numerals like 1 and 7, 5 and 6, 3 and 8, 2 and 5, 2 and 7, etc. Finally, the individuality and variation of each individual's handwriting influence the structure and appearance of the

digits.

REFERENCE

Aparna et al, proposed a method to construct a handwritten Tamil character by executing a sequence of strokes. A structure or shape-based representation of a stroke was used in which a stroke was represented as a string of shape features. Using this string representation, an unknown stroke was identified by comparing it with a database of strokes using a flexible string matching procedure.

Renata F. P. Neves has proposed SVM based offline handwritten digit recognition. Authors claim that SVM outperforms the Multilayer perceptron classifier. Experiment is carried out on NIST SD19 standard dataset. Advantage of MLP is that it is able to segment non-linearly separable classes. However, MLP can easily fall into a region of local minimum, where the training will stop assuming it has achieved an optimal point in the error surface. Another hindrance is defining the best network architecture to solve the problem, considering the number of layers and the number of perceptrons in each hidden layer. Because of these disadvantages, a digit recognizer using the MLP structure may not produce the desired low error rate.

Improved Handwritten Digit Recognition Using Quantum K-Nearest Neighbor Algorithm (2019)

Wang, Yuxiang and Wang, Ruijin and Li, Dongfen and Adu-Gyamfi, Daniel and Tian, Kaibin and Zhu, Yixin

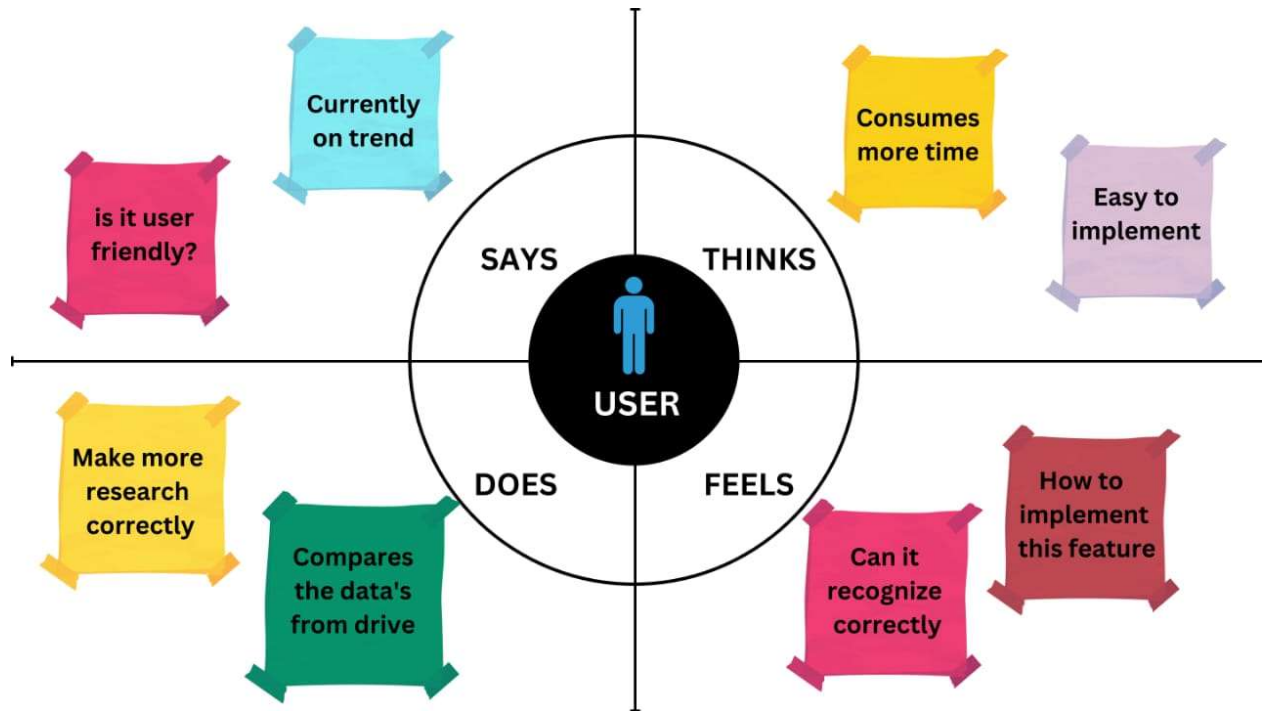
The KNN classical machine learning technique is used in this research to enable quantum parallel computing and superposition. They used the KNN algorithm with quantum acceleration to enhance handwritten digit recognition. When dealing with more complicated and sizable handwritten digital data sets, their suggested method considerably lowered the computational time complexity of the traditional KNN algorithm. The paper offered a theoretical investigation of how quantum concepts can be applied to machine learning. Finally, they established a fundamental operational concept and procedure for machine learning with quantum acceleration.

PROBLEM STATEMENT DEFINITION

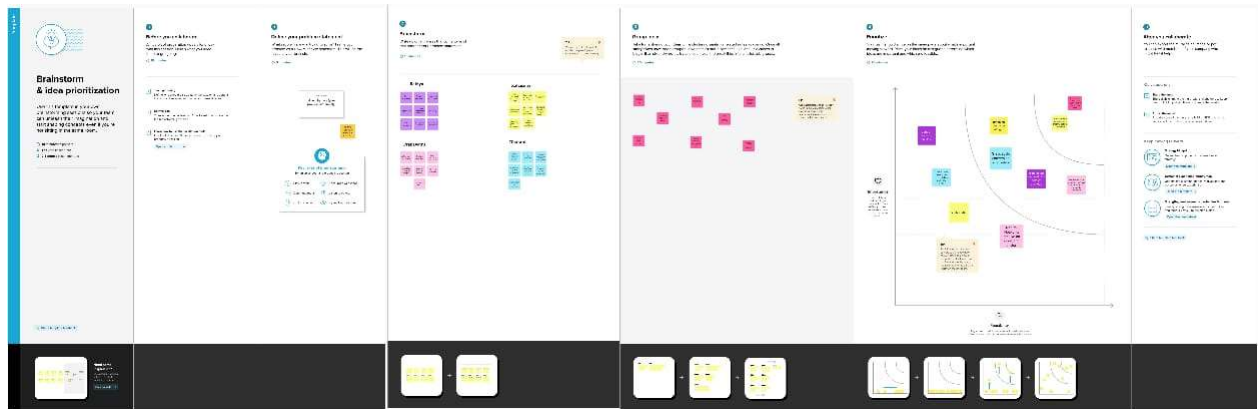
For years, the traffic department has been combating traffic law violators. These offenders endanger not only their own lives, but also the lives of other individuals. Punishing these offenders is critical to ensuring that others do not become like them. Identification of these offenders is next to impossible because it is impossible for the average individual to write down the license plate of a reckless driver. Therefore, the goal of this project is to help the traffic department identify these offenders and reduce traffic violations as a result.

CHAPTER 3

IDEATION AND PROPOSED SOLUTION



IDEATION & BRAINSTORMING



PROPOSED SOLUTION

S.NO	PARAMETER	DESCRIPTION
------	-----------	-------------

1	Problem Statement	To create an application that recognizes handwritten digits
2	Idea / Solution Description	The application takes an image as the input and accurately detects the digits in it.
3	Novelty / Uniqueness	Instead of recognizing every text, the application accurately recognizes only the digits
4	Social Impact / Customer Satisfaction	This application reduces the manual tasks that need to be performed. This improves productivity in the workplace.
5	Business Model	<p>The application can be integrated with traffic surveillance cameras to recognize vehicle number plates</p> <p>The application can be integrated with Postal systems to recognize the pin codes effectively</p>
6	Scalability of the Solution	The application can easily be scaled to accept multiple inputs and process them parallelly to further increase efficiency

PROBLEM SOLUTION FIT

Team ID:PNT2022TMID29320

<p>define CS, fit into CC</p> <p>1. CUSTOMER SEGMENT(S) Who is your customer? Is it a wider group of 1's or 10's</p> <p>The customer segment consists of users of bank post office, colleges, schools etc.</p>	<p>3. CUSTOMER CONSTRAINTS What constraints prevent your customers from taking action or limit their choice of solution or existing issues, budget, no ready, network connectivity, available already.</p> <p>Clarity of the input</p>	<p>5. AVAILABLE SOLUTIONS Which solutions are available to the customers when they face the problem Is there fit to get the job done? What have they tried in the past? What price is right for the solution? What's a past prototype is an alternative to digital scanning.</p> <p>Previously people needed to recognize the written digits and organize them. Now the hand written digit recognition tool will help to recognize and organize hand written digits efficiently.</p>
<p>Focus on JAR, map into BE, understand PC</p> <p>2. JOBS TO BE DONE / PROBLEMS Which jobs to be done by (problems) do you address for your customers? This could be more than only capture different jobs.</p> <p>The algorithm must be in such a way that the user finds the tool easy to use. The inefficient techniques used in past must be avoided.</p>	<p>9. PROBLEM ROOT CAUSE What is the root reason that this problem occurred? What is the task they had to do the need to do this job? Is it dependent down to do it because of the change in requirement</p> <p>Recognizing handwritten digits manually is too much exhaustive, time consuming and not efficient.</p>	<p>7. BEHAVIOUR What does your customer do to address the problem and get the job done? Is it directly related to the right order placed, available, customer usage and workflow, and the concerned customers cannot have time in understanding with it (consequence)</p> <p>Technical issues are handled by the technical team. The hand written digit recognition tool is easy to handle and accurate.</p>
<p>4. TRIGGERS What triggers customers to act? i.e. seeing their neighbours using digital scanner, hearing about a more efficient solution in the store</p> <p>The success of this hand written digit recognition tool will lead many banks, post offices, schools and colleges adopting this method.</p>	<p>10. YOUR SOLUTION If you are working on an existing business, try to derive your current solution from 2,3,4,5,6,7,8,9, and should focus much on 4,5,6,7,8,9 If you are working on a new business proposition, then keep it blank until you fill with purpose and come up with a solution that solves customer limitations, solves a problem, and satisfies customer behaviour.</p>	<p>8. CHANNELS OF BEHAVIOUR ONLINE What kind of solution do customers take online? (Stored online channels from IT) To upload the images for recognition</p> <p>OFFLINE What kind of solution do customers take offline? (Stored offline channels from physical store for customer convenience) Stored digitalized output images can be viewed</p>
<p>4. EMOTIONS: BEFORE / AFTER How do customers feel about their own problem or a job and afterwards? i.e. both, business + emotional, financial + any of the more communication strategy & change.</p> <p>Digitalization of hand written digits is done efficiently and easily without human interaction</p>	<p>Webpages can be used instead of applications for more interaction. The scanning should be able to done in such a way that the hand written digits can be recognized easily.</p>	

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Input correlation	Image Correlation is a technique used to recognize characters from images.
FR-2	Data Preparation	Collecting data and prepare it for training
FR-3	Feature extraction	Feature extraction is analysing the images and derive some characteristics from these images that identify each specific element
FR-4	Character classification	During the classification phase, the attributes of the data in the picture are compared to the classes in the database to determine which class the picture belongs to.

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

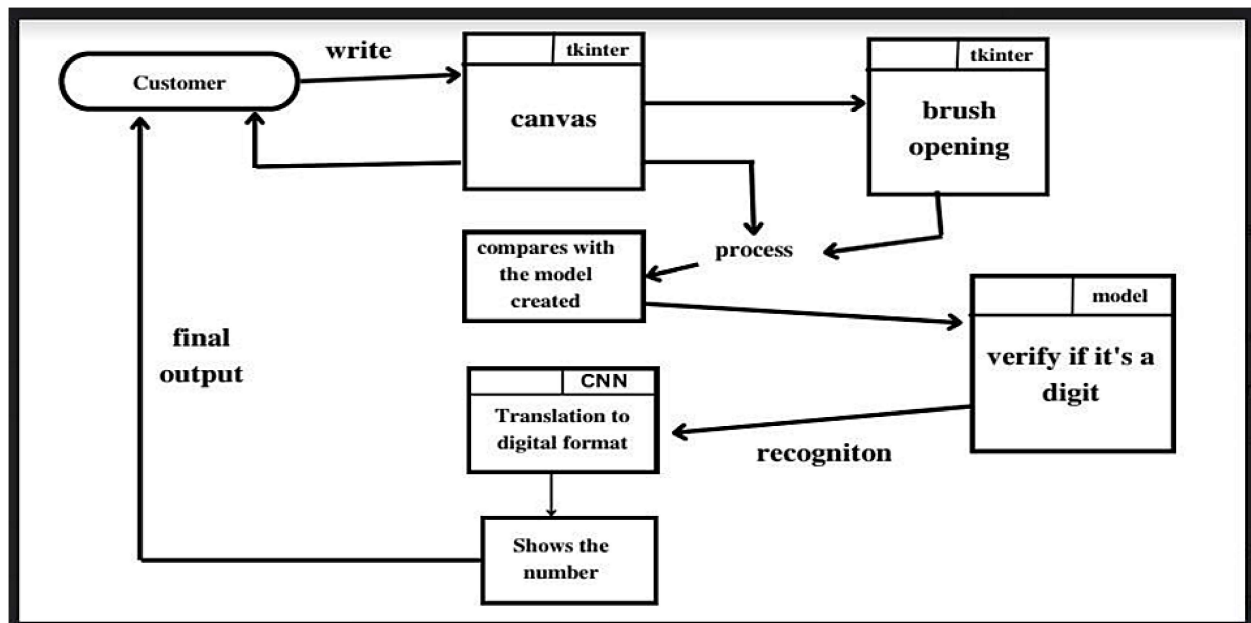
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The software is very easy to use and reduces the learning work.To recognize the digits from bank cheque,papers,numeric entry in forms etc.
NFR-2	Security	The handwritten digit recognition can be used by banking sector where it can be used to maintain the security pin numbers, it can be also used for blind peoples by using sound output.
NFR-3	Reliability	This software will work reliably for low resolution images and not for graphical images.
NFR-4	Performance	Handwritten characters in the input image will be recognized with an accuracy of about 90% and more.
NFR-5	Availability	This system will retrieve the handwritten text regions only if the image contains written text in it.
NFR-6	Scalability	It contains thousands of handwritten digits that have been used in the development of programs .

PROJECT DESIGN

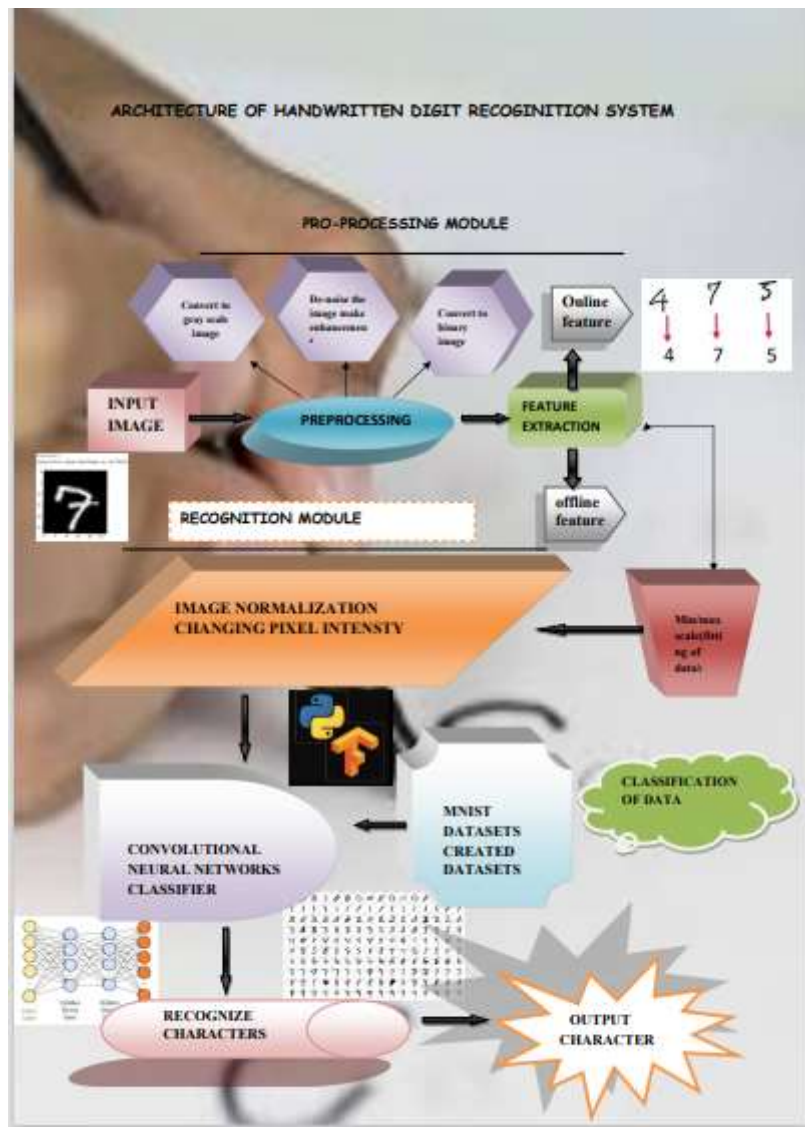
DATAFLOW DIAGRAMS:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

Example: DFD Level 0 (Industry Standard)



SOLUTION AND TECHNICAL ARCHITECTURE



USER STORIES:

User Type	Functional Requirements	User Story Number	User Story / Task	Acceptance Criteria	Priority	Release
Customer	Accessing the Application	USN-1	As a user, I should be able to access the application from anywhere and use on any devices	User can access the application using the browser on any device	High	Sprint-4
	Uploading Image	USN-2	As a user, I should be able to upload images to predict the digits	User can upload images	High	Sprint-3
	Viewing the Results	USN-3	As a user, I should be able to view the results	The result of the prediction is displayed	High	Sprint-3

	Viewing Other Prediction	USN-4	As a user, I should be able to see other close predictions	The accuracy of other values must be displayed	Medium	Sprint-4
	Usage Instruction	USN-5	As a user, I should have a usage instruction to know how to use the application	The usage instruction is displayed on the home page	Medium	Sprint-4

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

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	Low	Ilakiya Kalaiselvi
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	Prabavathi Dharani
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	Ilakiya Kalaiselvi Prabavathi Dharani
Sprint-2	Add CNN layers	USN-4	Creating the model and adding the input, hidden, and output layers to it.	5	High	Ilakiya Kalaiselvi Prabavathi Dharani

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	Kalaiselvi Prabavathi
Sprint-2	Train & test the model	USN-6	As a user, let us train our model with our image dataset.	6	Medium	Dharani Ilakiya
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	Ilakiya
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	Kalaiselvi
Sprint-3		USN-9	As a user, I can know the details of the fundamental usage of the application.	5	Low	Dharani
Sprint-3		USN-10	As a user, I can see the predicted / recognized digits in the application.	5	Medium	Prabavathi
Sprint-4	Train the model on IBM	USN-11	As a user, I train the model on IBM and integrate flask/Django with scoring end point.	10	High	Ilakiya Kalaiselvi Prabavathi Dharani
Sprint-4	Cloud Deployment	USN-12	As a user, I can access the web application and make the use of the product from anywhere.	10	High	Kalaiselvi Ilakiya

SPRINT DELIVERY SCHEDULE:

Sprint and Duration Chart

Sprint	Functional Requirement	Task
Sprint-1	Image Data	As a User need to collect the Image Data of Handly Written Images to train the model.

Sprint-2	Dash Board or Website	We using Python Flask Framework to create a dynamic Webpage to host our model (UI).
Sprint-3	Classifier Model	Using CNN Model for Image Classification.
Sprint-4	Cloud	Hosting the Organized appication in Cloud platform.

Sprint and Duration Chart

Sprint	Duration	Sprint Start Date	Sprint End Date
Sprint-1	6 Days	25 Oct 2022	29 Oct 2022
Sprint-2	6 Days	31 Oct 2022	05 Nov 2022
Sprint-3	6 Days	07 Nov 2022	12 Nov 2022

CHAPTER 7

CODING & SOLUTIONING

```
1 import numpy as np
2 import os
3 from PIL import Image
4 from flask import Flask, request, render_template, url_for
5 from werkzeug.utils import secure_filename, redirect
6 from gevent.pywsgi import WSGIServer
7 from keras.models import load_model
8 from keras.preprocessing import image
9 from flask import send_from_directory
10
```

```

11 UPLOAD_FOLDER = 'C:/Users/Dell/PycharmProjects/A-novel-method-for-digit-recognition-system/flask_app/uploads'
12
13
14 app = Flask(__name__)
15 app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
16
17 model = load_model("mnistCNN.h5")
18
19
20 @app.route('/')
21 def index():
22     return render_template('index.html')
23
24

```

```

20 @app.route('/')
21 def index():
22     return render_template('index.html')
23
24
25 @app.route('/predict', methods=['GET', 'POST'])
26 def upload():
27     if request.method == "POST":
28         f = request.files["image"]
29         filepath = secure_filename(f.filename)
30         f.save(os.path.join(app.config['UPLOAD_FOLDER'], filepath))
31
32         upload_img = os.path.join(UPLOAD_FOLDER, filepath)
33         img = Image.open(upload_img).convert("L") # convert image to monochrome
34         img = img.resize((28, 28)) # resizing of input image
35
36         im2arr = np.array(img) # converting to image
37         im2arr = im2arr.reshape(1, 28, 28, 1) # reshaping according to our requirement
38
39         pred = model.predict(im2arr)
40
41         num = np.argmax(pred, axis=1) # printing our labels
42
43         return render_template('predict.html', num=str(num[0]))
44
45
46 if __name__ == '__main__':
47     app.run(debug=True, threaded=False)
48

```

CHAPTER 8

TESTING

TEST CASES

Test case ID	Feature Type	Test Scenario	Expected Result	Actual Result	Status
HP_TC_001	UI	Verify UI elements in the Home Page	The Home page must be displayed properly	Working as expected	FAIL
HP_TC_002	UI	Check if the UI elements are displayed properly in different screen sizes	The Home page must be displayed properly in all sizes	The UI is not displayed properly in screen size 2560 x 1801 and 768 x 630	FAIL
HP_TC_003	Functional	Check if user can upload their file	The input image should be uploaded to the application successfully	Working as expected	PASS
HP_TC_004	Functional	Check if user cannot upload unsupported files	The application should not allow user to select a non image file	User is able to upload any file	FAIL

HP_TC_005	Functional	Check if the page redirects to the result page once the input is given	The page should redirect to the results page	Working as expected	PASS
-----------	------------	--	--	---------------------	------

BE_TC_001	Functional	Backend	Check if all the routes are working properly	All the routes should properly work	Working as expected	PASS
M_TC_001	Functional	Model	Check if the model can handle various image sizes	The model should rescale the image and predict the results	Working as expected	PASS
M_TC_002	Functional	Model	Check if the model predicts the digit	The model should predict the number	Working as expected	PASS
M_TC_003	Functional	Model	Check if the model can handle complex input image	The model should predict the number in the complex image	The model fails to identify the digit since the model is not built to handle such data	FAIL

RP_TC_001	UI	Result Page	Verify UI elements in the Result Page	The Result page must be displayed properly	Working as expected	PASS
RP_TC_002	UI	Result Page	Check if the input image is displayed properly	The input image should be displayed properly	The size of the input image exceeds the display container	FAIL
RP_TC_003	UI	Result Page	Check if the result is displayed properly	The result should be displayed properly	Working as expected	PASS
RP_TC_004	UI	Result Page	Check if the other predictions are displayed properly	The other predictions should be displayed properly	Working as expected	PASS

USER ACCEPTANCE TESTING **DEFECT ANALYSIS**

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Total
By Design	1	0	1	0	2
Duplicate	0	0	0	0	0
External	0	0	2	0	2
Fixed	4	1	0	1	6
Not Reproduced	0	0	0	1	1
Skipped	0	0	0	1	1
Won't Fix	1	0	1	0	2
Total	6	1	4	3	14

TEST CASE ANALYSIS

Section	Total Cases	Not Tested	Fail	Pass
Client Application	10	0	3	7
Security	2	0	1	1
Performance	3	0	1	2

Exception Reporting	2	0	0	2
---------------------	---	---	---	---

CHAPTER 9

RESULTS

PERFORMANCE METRICS

Locust Test Report

During: 11/15/2022, 9:50:40 AM - 11/15/2022, 10:01:59 AM

Target Host: http://127.0.0.1:5000/

Script: locust.py

Request Statistics

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	//	1043	0	13	4	290	1079	1.9	0.0
GET	//predict	1005	0	39648	385	59814	2670	1.8	0.0
Aggregated		2048	0	19462	4	59814	1859	3.7	0.0

Response Time Statistics

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	//	10	11	13	15	19	22	62	290
GET	//predict	44000	46000	47000	48000	50000	52000	55000	60000
Aggregated		36	36000	43000	45000	48000	50000	54000	60000

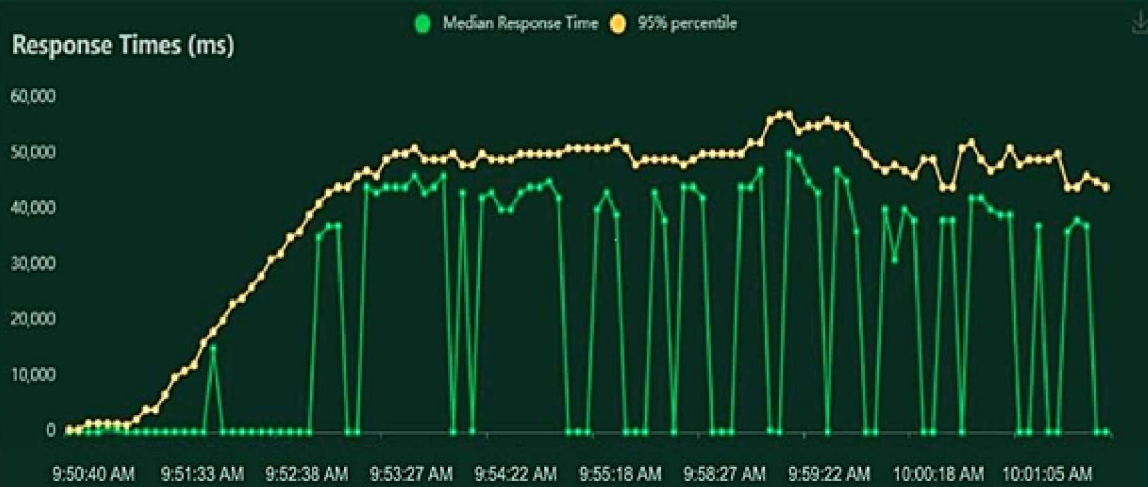
-

Charts

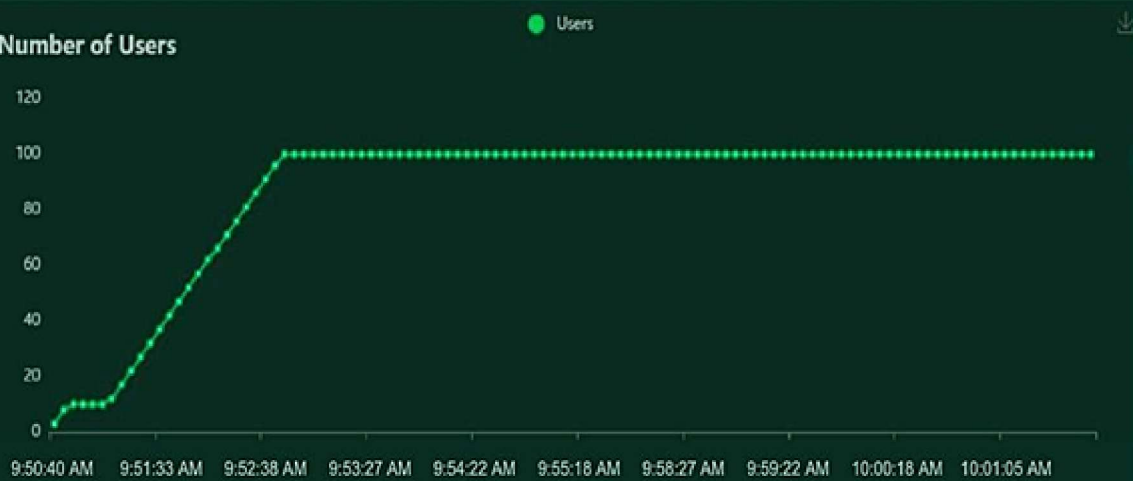
Total Requests per Second



Response Times (ms)



Number of Users



CHAPTER 10

ADVANTAGE & DISADVANTAGE

ADVANTAGE

- **Reduce manual work**
- **More accurate than human**
- **Capable of handling a lot of data**
- **Can be use any where fom any device**

DISADVANTAGES

- **cannot handle complex data**
- **All the data must be in digital format**
- **Requires a high performance server for faster prediction**
- **Prone to occasional errors**

CHAPTER 11

CONCLUSION

This project demonstrated a web application that uses machine learning to recognise handwritten numbers. Flask, HTML, CSS, JavaScript, and a few other technologies were used to create this project. The model predicts the handwritten digit using a CNN network. During testing, the model achieved a 99.61% recognition rate. The proposed project is scalable and can easily handle a huge number of users. Since it is a web application, it is compatible with any device that can run a browser. This project is extremely useful in real-world scenarios such as recognizing number plates of vehicles, processing bank cheque amounts, numeric entries in forms filled up by hand (tax forms) and so on. There is so much room for improvement, which can be implemented in subsequent versions

CHAPTER 12

FUTURE SCOPE

This project is far from complete and there is a lot of room for improvement. Some of the improvements that can be made to this project are as follows:

1. Add support to detect from digits multiple images and save the results
2. Add support to detect multiple digits
3. Improve model to detect digits from complex images
4. Add support to different languages to help users from all over the world

This project has endless potential and can always be enhanced to become better. Implementing this concept in the real world will benefit several industries and reduce the workload on many workers, enhancing overall work efficiency.

APPENDIX

SOURCE CODE

MODEL CREATION

```
import numpy as np
import tensorflow #open source used for both ML and DL for computation
from tensorflow.keras.datasets import mnist #mnist dataset
from tensorflow.keras.models import Sequential #it is a plain stack of layers
from tensorflow.keras import layers #A layer consists of a tensor- in tensor-out computation function
from tensorflow.keras.layers import Dense, Flatten #Dense-Dense Layer is the regular deeply connected n
#flatten -used for flattening the input or change the dimension
from tensorflow.keras.layers import Conv2D #convolutional layer
from tensorflow.keras.optimizers import Adam #optimizer
from keras.utils import np_utils #used for one-hot encoding
import matplotlib.pyplot as plt #used for data visualization

(x_train, y_train), (x_test, y_test)=mnist.load_data ()
x_train=x_train.reshape (60000, 28, 28, 1).astype('float32')
x_test=x_test.reshape (10000, 28, 28, 1).astype ('float32')
number_of_classes = 10 #storing the no of classes in a variable
y_train = np_utils.to_categorical (y_train, number_of_classes) #converts the output in binary format
y_test = np_utils.to_categorical (y_test, number_of_classes)
```

```
#adding model Layer
model.add(Conv2D(64, (3, 3), input_shape=(28, 28, 1), activation='relu'))
model.add(Conv2D(32, (3, 3), activation = 'relu'))
```

```
#flatten the dimension of the image
model.add(Flatten())
```

```
#output layer with 10 neurons
model.add(Dense(number_of_classes,activation = 'softmax'))
```

Compiling the model

```
#Compile model
model.compile(loss= 'categorical_crossentropy', optimizer="Adam", metrics=['accuracy'])
```

```
x_train = np.asarray(x_train)
y_train = np.asarray(y_train)
```

```
#Compile model
model.compile(loss= 'categorical_crossentropy', optimizer="Adam", metrics=['accuracy'])
```

```
x_train = np.asarray(x_train)
y_train = np.asarray(y_train)
```

Train the model

[+ Code](#) [+ Markdown](#)

```
#fit the model
model.fit(x_train, y_train, validation_data=(x_test, y_test), epochs=5, batch_size=32)
```

Observing the metrics

```
# Final evaluation of the model
metrics = model.evaluate(x_test, y_test, verbose=0)
print("Metrics (Test loss &Test Accuracy) : ")
print(metrics)
```

```
prediction=model.predict(x_test[6000:6001])
print(prediction)
```

```
plt.imshow(x_test[6000])
```

```
import numpy as np
print(np.argmax(prediction, axis=1)) #printing our labels from first 4 images
```

```
[9]
```

```
np.argmax(y_test[6000:6001]) #printing the actual labels
```

```
9
```

Save The model

```
# Save the model
model.save('mnistCNN.h5')
```

FLASK APP

```
20 @app.route('/')
21 def index():
22     return render_template('index.html')
23
24
25 @app.route('/predict', methods=['GET', 'POST'])
26 def upload():
27     if request.method == "POST":
28         f = request.files["image"]
29         filepath = secure_filename(f.filename)
30         f.save(os.path.join(app.config['UPLOAD_FOLDER'], filepath))
31
32         upload_img = os.path.join(UPLOAD_FOLDER, filepath)
33         img = Image.open(upload_img).convert("L") # convert image to monochrome
34         img = img.resize((28, 28)) # resizing of input image
35
36         im2arr = np.array(img) # converting to image
37         im2arr = im2arr.reshape(1, 28, 28, 1) # reshaping according to our requirement
38
39         pred = model.predict(im2arr)
40
41         num = np.argmax(pred, axis=1) # printing our labels
42
43         return render_template('predict.html', num=str(num[0]))
44
45
46 if __name__ == '__main__':
47     app.run(debug=True, threaded=False)
48
```

HOME PAGE HTML

```
1 <html>
2
3 <head>
4 <title>Digit Recognition WebApp</title>
5
6 <meta name="viewport" content="width=device-width">
7 <!-- GoogleFont -->
8 <link href="https://fonts.googleapis.com/css2?family=Prompt:wght@600&display=swap" rel="stylesheet">
9 <link href="https://fonts.googleapis.com/css2?family=Varela+Round&display=swap" rel="stylesheet">
10 <link href="https://fonts.googleapis.com/css2?family=Source+Code+Pro:wght@500&display=swap" rel="stylesheet">
11 <link href="https://fonts.googleapis.com/css?family=Calistoga|Josefin+Sans:400,700|Pacifico&display=swap" rel="stylesheet">
12 <!-- bootstrap -->
13 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" crossorigin="anonymous">
14 <link rel="stylesheet" type="text/css" href="{url_for('static',filename='css/style.css')}">
15 <!-- fontawesome -->
16 <script src="https://kit.fontawesome.com/b3aed9cb07.js" crossorigin="anonymous"></script>
17
18 <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" crossorigin="anonymous"></script>
19 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" crossorigin="anonymous"></script>
20 <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" crossorigin="anonymous"></script>
21 <script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@latest"></script>
22
23 </head>
24
25 <script>
26 function preview() {
27   frame.src=URL.createObjectURL(event.target.files[0]);
28 }
29
30 $(document).ready(function() {
31   $('#clear_button').on('click', function() {
32     $('#image').val('');
33     $('#frame').attr('src','');
34   });
35 });
36
37 </script>
```

```
39 <body>
40
41 <h1 class="welcome">IBM PROJECT
42 <div id="PNT2022TMD15689">TEAM ID : PNT2022TMD15689</div>
43 </h1>
44 <section id="title">
45 <h4 class="heading">Handwritten Digit Recognition Website</h4>
46 <br><br>
47 <p>
48 | The website is designed to predict the handwritten digit.
49 </p>
50 <p>
51 Handwriting recognition is one of the compelling research works going on because every individual in this world
52 has their own style of writing. It is the capability of the computer to identify and understand
53 handwritten digits or characters automatically. Because of the progress in the field of science and technology,
54 everything is being digitalized to reduce human effort.</p>
55 <br>
56 <p> Hence, there comes a need for handwritten digit recognition in many real-time applications.
57 MNIST data set is widely used for this recognition process and it has 70000 handwritten digits.
58 We use Artificial neural networks to train these images and build a deep learning model.
59 Web application is created where the user can upload an image of a handwritten digit.
60 This image is analyzed by the model and the detected result is returned on to UI.</p>
61
62 </section>
63
64 <section id="content">
65
66 <div class="leftside">
67 <form action="/predict" method="POST" enctype="multipart/form-data">
68 <label>Select a image:</label>
69 <input id="image" type="file" name="image" accept="image/png, image/jpeg" onchange="preview()"><br><br>
70 <img id="frame" src="" width="100px" height="100px"/>
71 <div class="buttons_div">
72 <button type="submit" class="btn btn-dark" id="predict_button">Predict</button>
73 <button type="button" class="btn btn-dark" id="clear_button">&nbsp;Clear &nbsp;&button>
74 </div>
75 </form>
76 </div>
77
78 </section>
```


HOME PAGE (CSS)

```
1  #clear_button{
2    margin-left: 15px;
3    font-weight: bold;
4    color: blue;
5  }
6
7  #confidence{
8    font-family: 'Josefin Sans', sans-serif;
9    margin-top: 7.5%;
10 }
11
12 #content{
13   margin: 0 auto;
14   padding: 2% 15%;
15   padding-bottom: 0;
16 }
17
18 .welcome{
19   text-align: center;
20   position: relative;
21   color: #honeydew;
22   background-color: #greenyellow;
23   padding-top: 1%;
24   padding-bottom: 1%;
25   font-weight: bold;
26   font-family: 'Prompt', sans-serif;
27 }
28
29 #PMT2022TMD15689{
30   text-align: right;
31   font-size: 25px;
32   padding-right: 3%;
33 }
34
35 #predict_button{
36   margin-right: 15px;
37   color: blue;
38   font-weight: bold;
39 }
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

```
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

```

84
85 #frame{
86   margin-right: 10%;
87 }
88
89 .predicted_answer{
90   text-align: center;
91   margin: 0 auto;
92   padding: 3% 5%;
93   padding-top: 0;
94   /* padding-left: 10%; */
95 }
96
97 p{
98   font-family: 'Source Code Pro', monospace,sans-serif;
99   margin-top: 1%;
100 }
101
102 @media (min-width: 720px) {
103   .leftside{
104     padding-left: 10%;
105   }
106 }
107

```

HOME PAGE (JS)

```

feather.replace(); // Load feather icons

form = document.querySelector('.upload')
loading = document.querySelector("#Loading")
select = document.querySelector("#upload-image");

select.addEventListener("change", (e) => {
  e.preventDefault();

  form.submit()
  form.style.visibility = "hidden";
  loading.style.display = 'flex';
});

```

PREDICT PAGE HTML

```
<html>
  <head>
    <title>Prediction | Handwritten Digit Recognition</title>
    <link rel="stylesheet" href="{{url_for('static',filename='css/predict.css')}}" />
    <link rel="icon" type="image/svg" sizes="32x32" href="{{url_for('static',filename='images/icon.svg')}}"
  />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  </head>
  <body>
    <div class="container">
      <h1>Prediction</h1>
      <div class="result-wrapper">
        <div class="input-image-container">
          
        </div>
        <div class="result-container">
          <div class="value">{{best.0}}</div>
          <div class="accuracy">{{best.1}}%</div>
        </div>
      </div>
      <h1>Other Predictions</h1>
      <div class="other_predictions">
        {% for x in others %}
          <div class="value">
            <h2>{{x.0}}</h2>
            <div class="accuracy">{{x.1}}%</div>
          </div>
        {% endfor %}
      </div>
    </div>
  </body>
</html>
```

```

@import url("https://fonts.googleapis.com/css2?family=Overpass:wght@200;300;400;500;600;700;900&display=swap");

body {
  color: black;
  font-family: "Overpass", sans-serif;
}

h1 {
  padding-top: 2rem;
}

.container {
  display: flex;
  justify-content: center;
  align-items: center;
  flex-direction: column;
}

.result-wrapper {
  width: -webkit-fit-content;
  width: -moz-fit-content;
  width: fit-content;
  height: -webkit-fit-content;
  height: -moz-fit-content;
  height: fit-content;
  box-shadow: 0 0 10px rgb(125, 125, 125);
  padding: 1.5rem;
  display: flex;
  justify-content: center;
  align-items: center;
  -moz-column-gap: 1rem;
  column-gap: 1rem;
}

.result-wrapper .input-image-container,
.result-wrapper .result-container {
  width: 15rem;
  height: 15rem;
  border: 1px dashed black;
  justify-content: center;
  display: flex;
  align-items: center;
  flex-direction: column;
  background-color: rgb(200, 200, 200);
}

```

```

.result-wrapper .input-image-container img {
  width: 66%;
  height: 66%;
  background-color: aqua;
  background-size: contain;
}

.result-wrapper .result-container .value {
  font-size: 6rem;
}

.result-wrapper .result-container .accuracy {
  margin-top: -1rem;
}

.other_predictions {
  display: flex;
  justify-content: center;
  align-items: center;
  flex-wrap: wrap;
  column-gap: 1rem;
  row-gap: 1rem;
  font-weight: 700;
}

.other_predictions .value {
  display: flex;
  justify-content: center;
  align-items: center;
  flex-direction: column;
  width: 5rem;
  height: 5rem;
  box-shadow: 8 8 7px rgb(155, 157, 157);
}

.other_predictions .value div {
  margin-top: -1.2rem;
}

@media screen and (max-width: 768px) {
  h1 {
    font-size: 2.3rem;
  }

  .result-wrapper .input-image-container,
  .result-wrapper .result-container {
    width: 7rem;
    height: 7rem;
  }

  .result-wrapper .result-container .value {
    font-size: 4rem;
  }
}

```

GITHUB PROFILE

IBM-EPBL / IBM-Project-35718-1660287935Public

Watch1Fork0Star0

<> CodeIssuesPull requestsActionsProjectsWikiSecurityInsights

mainGo to fileAdd fileCode

ILAKIYA-M Update README.md1 minute ago136

Application Building	Add files via upload	3 days ago
Assignment	Add files via upload	last month
Final_Deliverables	Create readme.txt	last month
Model Building	Add files via upload	11 days ago
Project_Design_Phase_1	Add files via upload	last month
Project_Design_Phase_2	Add files via upload	last month
Project_Development_...	Add files via upload	3 days ago
Project_Planning_Phase	Add files via upload	28 days ago
Quiz	Add files via upload	29 days ago
Reference	Create README.md	2 months ago
Train the Model on IBM	Add files via upload	3 days ago
Understanding_the_Data	Add files via upload	26 days ago
ideation	Add files via upload	last month
IBM assignment 2.ipynb	Add files via upload	2 months ago
Assignment kalaiselvi ...	Add files via upload	2 months ago
Assignment.pdf	Add files via upload	2 months ago
Assignment1.ipynb	Add files via upload	2 months ago
Assignment_1 (1).ipynb	Add files via upload	2 months ago
Assignment_1 complet...	Add files via upload	2 months ago
Dharani_2_completed.i...	Add files via upload	2 months ago
Dharani_assignment_3...	Add files via upload	2 months ago
IBM assignment 1 (1).i...	Add files via upload	2 months ago
Ilakiya_assignment2.pdf	Add files via upload	2 months ago
Ilakiya_assignment3.pdf	Add files via upload	2 months ago
README.md	Update README.md	1 minute ago
kalaiselvi assignment2	Add files via upload	2 months ago

README.md

IBM PROJECT SUBMISSION

TEAM ID - PNT2022TMID29320

A Novel Method for Handwritten Digit Recognition System

About

A Novel Method for Handwritten Digit Recognition System

Readme0 stars1 watching0 forks

Releases

No releases publishedCreate a new release

Packages

No packages publishedPublish your first package

Contributors4

ILAKIYA-MPrabapmkalai-GDharani-E14

Languages

Jupyter Notebook100.0%

GITHUB LINK

<https://github.com/IBM-EPBL/IBM-Project-35718-1660287935.git>

GITHUB ID

IBM-PROJECT-35718-1660287935

<https://github.com/IBM-EPBL/IBM-Project-35718-1660287935>