

# **PROJECT REPORT**

## ***A NOVEL METHOD FOR HANDWRITTEN DIGIT RECOGNITION SYSTEM***

### **TEAM MEMBERS**

- SHEIKH AMEENUL HAJI**
- RITHISH A**
- SRIKANTH**
- ARAVIND J**

# Project Report Format

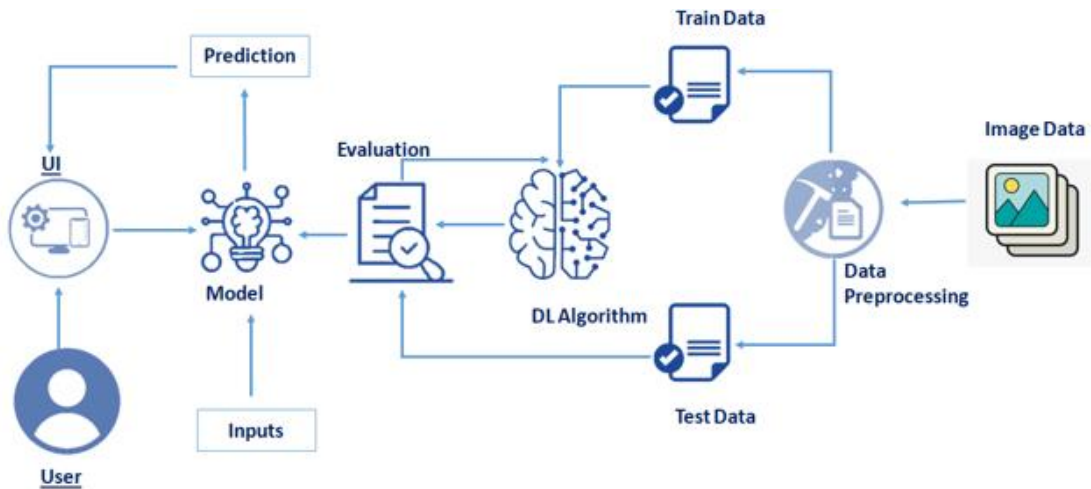
## 1. INTRODUCTION

### 1.1 Project Overview

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

Domain- Artificial Intelligence

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. 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.



### 1.2 Purpose

- The purpose of this project is to:
- The user interacts with the UI (User Interface) to upload the image as input
- The uploaded image is analyzed by the model which is integrated
- Once the model analyses the uploaded image, the prediction is showcased on the UI

## 2. LITERATURE SURVEY

### 2.1 Existing problem

As we know that today time is of great importance so we have try to manage time correctly. We can see that bank employee cannot manually enter the details in computer of a challan ,cheque,

or withdrawal form if there is a long queue of people are waiting for getting their work done. Sorting letters according to their pincodes is also difficult task which takes a long time. Hence to resolve all this problem that we face in work life, we can use software for handwritten recognition software which will greater the reduce the effect and time.

## 2.2 References

- [1] J. Li, G. Sun, L. Yi, Q. Cao, F. Liang and Y. Sun, "Handwritten Digit Recognition System Based on Convolutional Neural Network," 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications( AEECA), 2020, pp. 739-742, doi: 10.1109/AEECA49918.2020.9213619.
- [2] A. Chakraborty, R. De, S. Malakar, F. Schwenker and R. Sarkar, "Handwritten Digit String Recognition using Deep Autoencoder based Segmentation and ResNet based Recognition Approach," 2020 25th International Conference on Pattern Recognition (ICPR), 2021, pp. 7737-7742, doi: 10.1109/ICPR48806.2021.9412198.
- [3] S. Aly and S. Almotairi, "Deep Convolutional Self-Organizing Map Network for Robust Handwritten Digit Recognition," in IEEE Access, vol. 8, pp. 107035-107045, 2020, doi: 10.1109/ACCESS.2020.3000829.
- [4] S. Aly and A. Mohamed, "Unknown-Length Handwritten Numeral String Recognition Using Cascade of PCA-SVMNet Classifiers," in IEEE Access, vol. 7, pp. 52024-52034, 2019, doi: 10.1109/ACCESS.2019.2911851.
- [5] A. K. Agrawal, A. K. Shrivastava and V. K. Awasthi, "A Robust Model for Handwritten Digit Recognition using Machine and Deep Learning Technique," 2021 2nd International Conference for Emerging Technology (INCET), 2021, pp. 1-4, doi: 10.1109/INCET51464.2021.9456118.
- [6] W. Liu, J. Wei and Q. Meng, "Comparisons on KNN, SVM, BP and the CNN for Handwritten Digit Recognition," 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications( AEECA), 2020, pp. 587-590, doi: 10.1109/AEECA49918.2020.9213482.
- [7] T. Pala, U. Güvenç, H. T. Kahraman, İ. Yücedağ and Y. Sönmez, "Comparison of Pooling Methods for Handwritten Digit Recognition Problem," 2018 International Conference on Artificial Intelligence and Data Processing (IDAP), 2018, pp. 1-5, doi: 10.1109/IDAP.2018.8620848.
- [8] S. Ahlawat and R. Rishi, "Handwritten Digit Recognition using Adaptive Neuro-Fuzzy System and Ranked Features," 2018 International Conference on Computing, Power and Communication Technologies (GUCON), 2018, pp. 1128-1132, doi: 10.1109/GUCON.2018.8675013.

## 2.3 Problem Statement Definition

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. Recognition of Handwritten Digits using Convolutional neural networks.

## 3. IDEATION & PROPOSED SOLUTION

### 3.1 Empathy Map Canvas

- An empathy map is a straightforward, simple-to-understand picture that summarizes information about a user's actions and views.
- Teams can utilize an empathy map as a collaborative tool to obtain a deeper understanding of their customer.
- An empathy map is a popular visualization tool in the UX and HCI fields of practice.
- An empathy map's main objective in empathetic design is to bridge the understanding of the end user.

- A rectangle divided into four quadrants, with the user or client in the center, is an empathy map. A category is included in each of the four quadrants to assist us better understand the user's perspective.
- The four empathy map quadrants examine the user's actions, thoughts, and feelings.

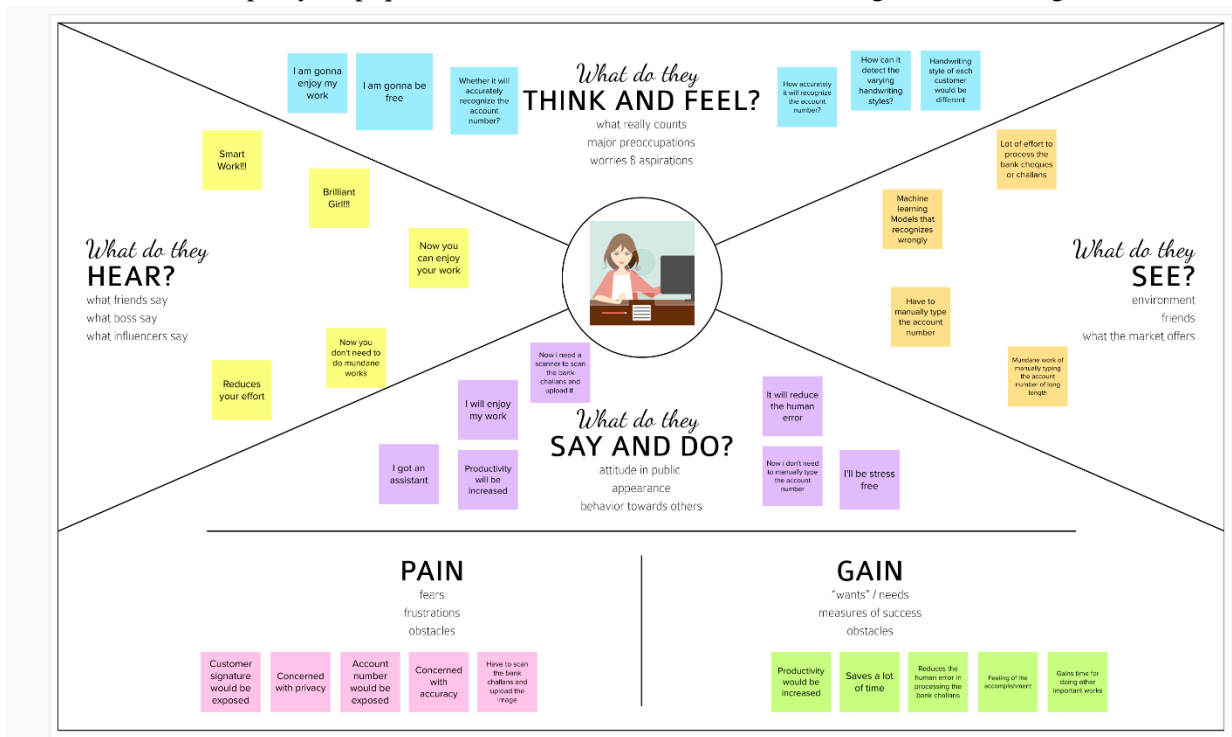


Figure 3.1 Empathy Map

### 3.2 Ideation & Brainstorming

1

## Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

🕒 5 minutes

---

### PROBLEM

Recognize the digits written  
by the users

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!



### Sheikh

A feature to export the recognized text

Remove noises to improve the accuracy

Various algorithms can be used

Scan using mobile phone

### Rithish

Use image or canva as input

Digits like 3-8, 1-7 should be trained more

Scanner can be used to scan the handwritten documents

Dataset should contain the digits of varying styles

### Srikanth

CNN can be used as a model for recognition

High quality images should be used for training

Use more images for training

Train different styles of handwriting

### Aravind

Segmentation techniques can also be used for recognizing the string of digits

Feedback from the users can be used for improving the model performance

Alert user if any digit cannot be scanned

Test with different styles of handwriting

3

### Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, 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

#### TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

## Train Datasets

Dataset should contain the digits of varying styles

Train different Styles of handwriting

Digits like 3-8 , 1-7 should be trained more

## Input

Use image or canva as a input

Scan using mobile phone

Scanner can be used to scan the handwritten documents

## Preprocessing

Remove noises to improve the accuracy

Segmentation techniques can also be used for recognizing the string of digits

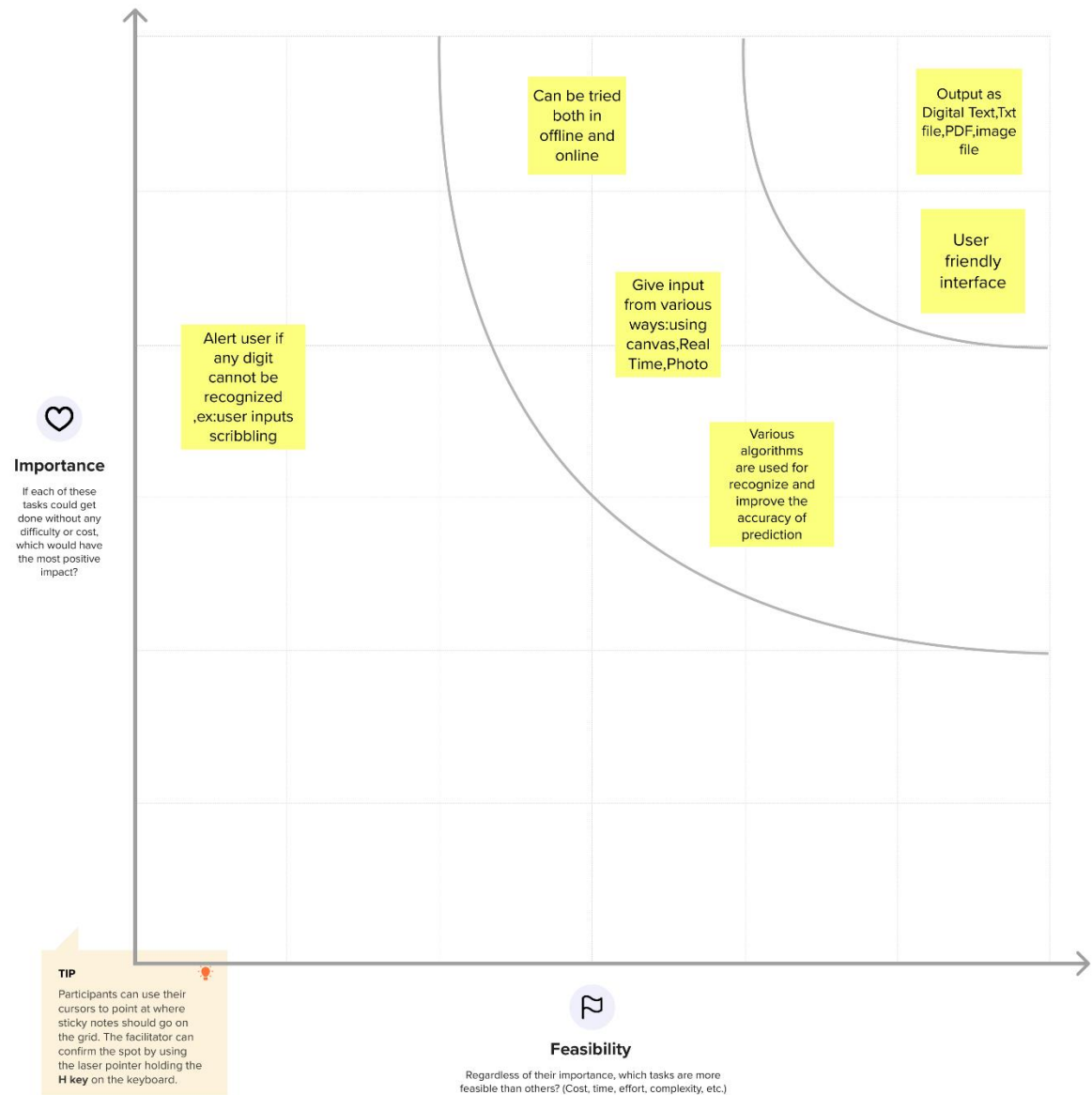
Scanner can be used to scan the handwritten documents

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)	Recognize handwritten expressions consisting of digits and mathematical symbols and evaluate the expression.
2.	Idea / Solution description	Image dataset of digits and symbols are used to train the CNN model and it is used to find the digits or expression in digital form and it is processed
3.	Novelty / Uniqueness	Rather than only recognizing the (0-9) digits our project can recognize the digits along with mathematical symbols (+,-,*,/,()) and calculate the value of the expression ("2 +3-6", "8*6-4")
4.	Social Impact / Customer Satisfaction	The project will be useful for customers like bank staffs, students, etc to recognize digits and also solve expression
5.	Business Model (Revenue Model)	Revenue can be generated by making it as an web app by which anyone can access it.
6.	Scalability of the Solution	In the future the model can be used to recognize complex mathematical expression which has some different symbols like root functions, sin ,cos, differential, integration, etc. It will be very helpful to recognize some notes which are tough to understand.

### 3.4 Problem Solution fit

Project Title:

Project Design Phase-I - Solution Fit Template

Team ID: PNT2022TMDxxxxx

Define CS, fit into CC	<b>1. CUSTOMER SEGMENT(S)</b> Who is your customer? i.e. working parents of 0-5 y.o. kids  <b>Children of class (1-3) who wants to find the answers for the arithmetic problems.</b>	<b>6. CUSTOMER CONSTRAINTS</b> What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.  1.Children of class (1-3) would not be given cell phones. 2.Their parents would be busy so children can't ask their parents	<b>5. AVAILABLE SOLUTIONS</b> Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking.  1.Ask their friends to cross check with the answers Cons: They also would not be sure of the answers. Pros: Answers might be correct but not sure.  2.Ask their teachers Cons: They would be busy Pros: Answers would be accurate	Explore AS, differentiate
	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one, explore different sides.  1.To recognize the symbols(0-9,+,-,(,),!,*,/) written by the children. 2.To solve the handwritten arithmetic problems.	<b>9. PROBLEM ROOT CAUSE</b> What is the real reason that this problem exists? What is the back story behind the need to do this job? i.e. customers have to do it because of the change in regulations.  Inability to do the arithmetic problems on their own. They have to use our project to cross check their answers with our results	<b>7. BEHAVIOUR</b> What does your customer do to address the problem and get the job done? i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace).  They'll ask their friends and teachers and sometimes from the books.  Search in the google how to solve the similar problem and do by that method.	
Focus on J&P, tap into BE, understand RC	<b>3. TRIGGERS</b> What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.  1.To cross check the answers. 2.To solve the questions in a faster way	<b>10. YOUR SOLUTION</b> If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.  Build the machine learning model that recognizes the digits and symbols and solve the expressions.	<b>8. CHANNELS of BEHAVIOUR</b> <b>8.1 ONLINE</b> What kind of actions do customers take online? Extract online channels from #7  <b>8.2 OFFLINE</b> What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.  Offline -> Friends, teachers, family Online->Search in the google for the similar problem	Identify strong TR & EM
	<b>4. EMOTIONS: BEFORE / AFTER</b> How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure > confident, in control - use it in your communication strategy & design.  Before: They are not confident whether they have solved the question correctly or not. After: They can identify whether they have solved correctly.		<b>Identify strong TR &amp; EM</b>	

## 4. REQUIREMENT ANALYSIS

### 4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Website	UI for the website
FR-2	Upload option	The upload option used by the user to upload the iage to be recognized.
FR-3	Result	The result will be displayed in the website in digital form.

### 4.2 Non-Functional requirements

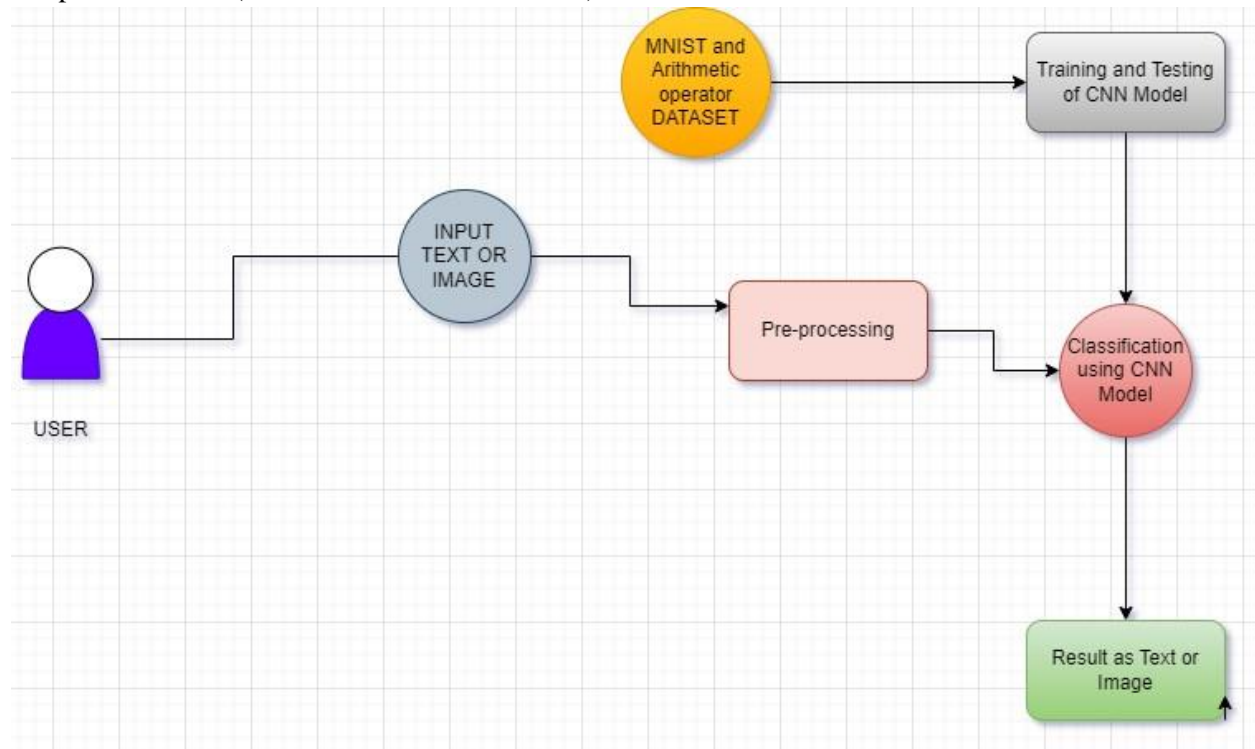
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The website is easy to use.
NFR-2	Reliability	Since more amount and variety of dataset is used, the output is reliable to an greater extent.
NFR-3	Performance	Since CNN is used, the performance of the model will be fast.

NFR-4	<b>Availability</b>	The project is hosted on a platform where it is always available.
-------	---------------------	---

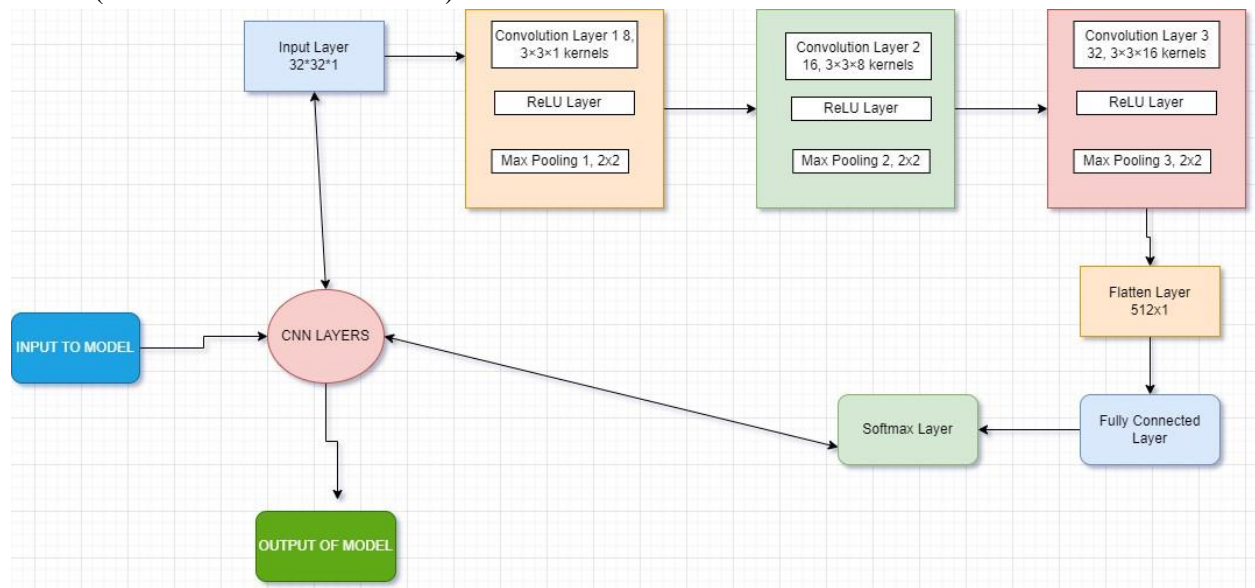
## 5. PROJECT DESIGN

### 5.1 Data Flow Diagrams

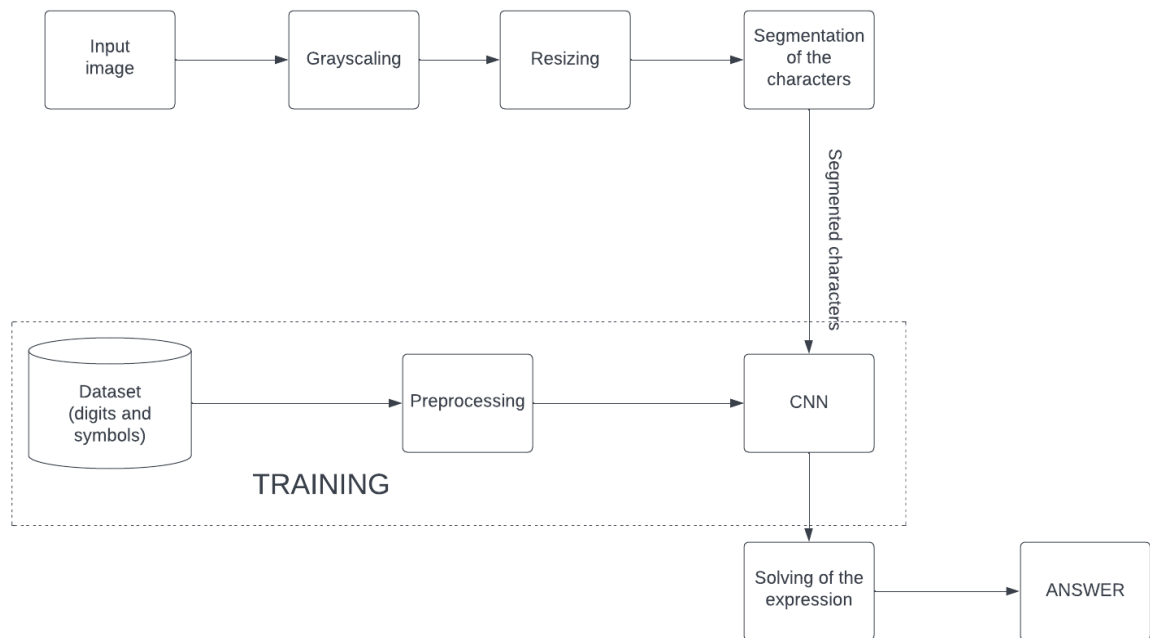
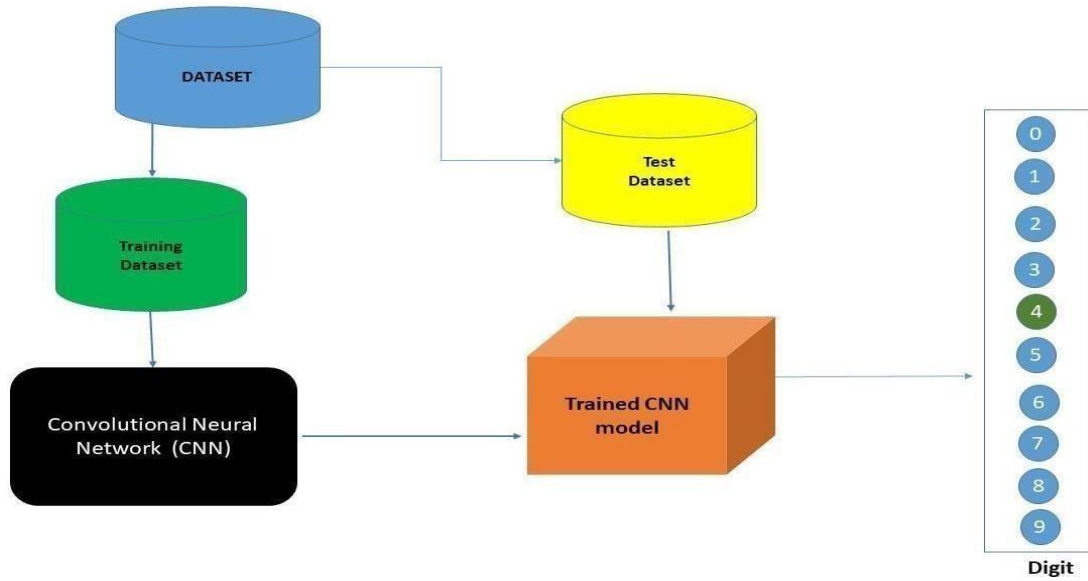
Simplified: DFD-0(DATA FLOW DIAGRAM-0)



DFD-1(DATA FLOW DIAGRAM 1)



## 5.2 Solution & Technical Architecture



### 5.3 User Stories

User Type	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	USN-1	I can use this Web App to do calculation for basic math like addition, subtraction etc.	I am getting the result	Medium	Sprint-1
	USN-2	I am a postman, I want to recognize the numbers in letters for delivery.	I can get the digital text to store it in computer memory	Medium	Sprint-1
	USN-3	I am bank employee,I want to recognize digits of cheque or challan and enter in computer	I can get the numbers from the cheque	Medium	Sprint-2
	USN-4	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-2
	USN-5	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-6	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-3
Customer (Web user)	USN-7	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

## 6. PROJECT PLANNING & SCHEDULING

### 6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data Collection	USN-1	Collects the different handwritten digits along with the handwritten mathematical symbols(+, -, *, /)	2	High	Sheik Ameenul Haji,Aravind
Sprint-1	Data Preprocessing	USN-2	Preprocess the Collected data like scaling ,resizing the images etc	1	High	Rithish,Srikanth
Sprint-2	Model Building	USN-3	Build the CNN Model	1	Medium	Sheik,Aravind

<b>Sprint</b>	<b>Functional Requirement (Epic)</b>	<b>User Story Number</b>	<b>User Story / Task</b>	<b>Story Points</b>	<b>Priority</b>	<b>Team Members</b>
Sprint-2	Compiling Model	USN-4	Compile the built model	1	Medium	Srikanth,Sheik
Sprint-2	Train the model	USN-5	Train the built model using the datasets	1	High	Rithish,Sheik
Sprint -2	Save the model	USN-6	Save the trained model weights	-	Medium	Aravind,Srikanth
Sprint-3	Evaluate the user given arithmetic expression	USN-7	User should give the handwritten arithmetic expression, the model should give the correct result	1	High	Rithish,Aravind
Sprint -4	Build the web app	USN-8	User should upload the image using web app and the answer should be displayed in the web app	1	Medium	Sheik,Srikanth

## 6.2 Sprint Delivery Schedule

<b>Sprint</b>	<b>Total Story Points</b>	<b>Duration</b>	<b>Sprint Start Date</b>	<b>Sprint End Date (Planned)</b>	<b>Story Points Completed (as on Planned End Date)</b>	<b>Sprint Release Date (Actual)</b>
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	1,2	30 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	2,3	04 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	1,2	11 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	1,2	19 Nov 2022

## 7. CODING & SOLUTIONING (Explain the features added in the project along with code)

The user can upload the image for prediction choose from the from his local directory and the image is predicted and displayed in the front end. The trained model is saved in same directory and used for prediction as given.

### 7.1.Feature1

```
app.py > upload
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 keras.models import load_model
7  from keras.preprocessing import image
8  from flask import send_from_directory
9  from keras.utils import img_to_array
10 import cv2
11
12 UPLOAD_FOLDER = 'C:/College/Semesters/7th sem/IBM Project Works/Data'
13 app=Flask(__name__)
14 app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
15
16 model=load_model(".venv/assets/mnistCNN.h5")
17
18 @app.route("/")
19 def homepage():
20     return render_template("index.html")
21
22
23 @app.route(['/predict', methods=['GET', 'POST']])
24 def upload():
25     if request.method == "POST":
26         f = request.files["image"]
27         filepath = secure_filename(f.filename)
28         f.save(os.path.join(app.config['UPLOAD_FOLDER'], filepath))
29
30         upload_img = os.path.join(UPLOAD_FOLDER, filepath)
31         img = Image.open(upload_img).convert("L") # convert image to monochrome
32         img=np.asarray(img)
33         img=cv2.resize(img,(28,28),interpolation=cv2.INTER_LINEAR)
34         thresh,bw_image=cv2.threshold(img,127,255,cv2.THRESH_BINARY) #converting grayscale to binary image
35         bw_image=255-bw_image
36         bw_image=img_to_array(bw_image)
37         bw_image=np.asarray(bw_image)
38         bw_image=np.expand_dims(bw_image,0)
39         pred = model.predict(bw_image)
40
41         num = np.argmax(pred, axis=1) # printing our Labels
42
43         return render_template('predict.html', num=str(num[0]))
44
45 if __name__=="__main__":
46     app.run()
```

## 7.2.Feature 2

```
<> index.html <> predict.html X

<> predict.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <title>Prediction</title>
6  </head>
7
8  <style>
9    body{
10      /*background-image: url('static/images/Background_Image.jpg');*/
11      background-repeat: no-repeat;
12      background-size: cover;
13    }
14
15    #rectangle{
16      width:400px;
17      height:150px;
18      background-color: #000000;
19      border-radius: 15px;
20      position:absolute;
21      box-shadow: 0px 0px 10px 5px #white;
22      top:25%;
23      left:50%;
24      transform:translate(-50%,-50%);
25    }
26
27    #head{
28      text-align: center;
29      font-size: 30px;
30      margin: 0 auto;
31      padding: 3% 5%;
32      font-family: Arial, Helvetica, sans-serif;
33      color: #white;
34    }
35
36    #num{
37      font-size: 50px;
38    }
39
40  </style>
41
42  <body>
43
44    <div id="rectangle">
45      <h1 id="head">Predicted Number : <br><center id="num">{{num}}</center></h1>
46    </div>
47
48  </body>
49  </html>
```



```
<?xml version="1.0" encoding="UTF-8" ?>
<html>
<head>
<title>Handwritten Digit Recognition</title>
<meta name="viewport" content="width=device-width">
<link href="https://fonts.googleapis.com/css?family=Prompt:wght@800&display=swap" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Varela+Round&display=swap" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Source+Code+Pro:wght@900&display=swap" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Callistoga|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-ggOyR0iXCbMQVtE/W8+v30x3MPe/61/M9T0A8Pf4XG1J96I8Pn4k4Sb/Pk64" crossorigin="anonymous">
<link rel="stylesheet" type="text/css" href="{url_for('static',filename='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-q81/X+965D2004788b415D1QI3qVzpbz0s5mXk44fVvH+8abTTE1Pi6jzo" crossorigin="anonymous"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-Uo2eT8cJqoEhW3+dX9f9V0+o+S9dX/qna1IQY18o1J7rLLPE632/P1MzCk5tc" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/axios@1.0.0-beta.2/dist/axios.min.js" integrity="sha384-7J5WV5v3Y8p131K7yUo1xH3p8f4Y1Z531W73tD45ELG58bD60X31jEAA" crossorigin="anonymous"></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"></script>
</head>
<style>
body{
/*background-image: url("static/images/Background_Image.jpg");*/
background-repeat: no-repeat;
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 p-3 my-3 bg-dark text-white">
<p style = "text-align:center;font-family:Prompt" > A Novel Method for Handwritten Digit Recognition System</p>
<p style = "text-align:center;font-family:Prompt" > Team ID - PNT2022TMD35610</p>
</div>
<section id="content">
<div class="leftside">
<form action="/predict" 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>
<div id="frame" width="100px" height="100px">
<div class="buttons_div">
<button type="submit" class="btn btn-danger" id="predict_button">Predict</button>
</div>
</div>
</div>
</body>
```

8. TESTING

8.1 Test Cases

Test case ID	Feature Type	Component	Test Scenario	Steps To Execute	Test Data	Expected Result	Actual Result	Status	BUG ID	Executed By
HP_TC_001	UI	Home Page	Verify UI elements in the Home Page	1) Open the page 2) Check if all the UI elements are displayed 3) Click on My Account dropdown button 4) Verify	127.0.0.5000	The Home page must be displayed properly	Working as expected	Pass		Sheikh Ameenul Haji,Rithish A

				erify login/Sign up popup displayed or not						
HP_TC_002	UI	Home Page	Check if the UI elements are displayed properly in different screen sizes	1) Open the page in a specific device 2) Check if all the UI elements are displayed properly 3) Repeat the above steps with different device sizes	Screen sizes: 2560 x 1801, 1440 x 970, 1024 x 840, 768 x 630, 320 x 630	The Home page must be displayed properly in all sizes	Working as expected	Pass		Aravind J,Srikanth
HP_TC_003	Functional	Home page	Check if user can upload their file	1) Open the page 2) Click on select button 3) Select the input image	1.png	The input image should be uploaded to the application successfully	Working as expected	Pass		Sheikh Ameenul Haji,Srikanth
HP_TC_004	Functional	Home Page	Check if user cannot upload unsupported files	1) Open the page 2) Click on select button 3) Select a random input file	installer.exe	The application should not allow user to select a non image file	User is able to upload any file	Fail	BUG_ID_001	Aravind J,Rithish A
HP_TC_005	Functional	Home Page	Check if the page redirects to the result page once the input is given	1) Open the page 2) Click on select button 3) Select the input image 4) Check if the page redirects	1.png	The page should redirect to the results page	Working as expected	Pass		Sheikh Ameenul Haji,Rithish A
BE_TC_001	Functional	Backend	Check if all the routes are working properly	1) Go to Home Page 2) Upload the input image 3) Check the results page	1.png	All the routes should properly work	Working as expected	Pass		Aravind J,Srikanth
M_TC_001	Functional	Model	Check if the model can handle various image sizes	1) Open the page in a specific device 2) Upload the input image 3) Repeat the above steps with different input image	1.png, 2.png, 3.png	The model should rescale the image and predict the results	Working as expected	Pass		Sheikh Ameenul haji,Aravind J

M_TC_002	Functional	Model	Check if the model predicts the digit	1) Open the page 2) Click on select button 3) Select the input image 4) Check the results	1.png	The model should predict the number	Working as expected	Pass		Srikanth,Rithish A
M_TC_003	Functional	Model	Check if the model can handle complex input image	1) Open the page 2) Click on select button 3) Select the input image 4) Check the results	4.png	The model should predict the number in the complex image	Working as expected	Pass		Sheikh Ameenul Haji,Rithish A
RP_TC_001	UI	Result Page	Verify UI elements in the Result Page	1)Open the page 2)Click on select button 3)Select the input image 4)Check if all the UI elements are displayed properly	1.png	The Result page must be displayed properly	Working as expected	Pass		Aravind J,Srikanth
RP_TC_002	UI	Result Page	Check if the input image is displayed properly	1)Open the page 2)Click on select button 3)Select the input image 4) Check if the input image are displayed	1.png	The input image should be displayed properly	Working as expected	Pass		Sheikh Ameenul Haji,Srikanth
RP_TC_003	UI	Result Page	Check if the result is displayed properly	1) Open the page 2) Click on select button 3) Select the input image 4) Check if the result is displayed	1.png	The result should be displayed properly	Working as expected	Pass		Aravind J,Srikanth
RP_TC_004	UI	Result Page	Check if the other predictions are displayed properly	1)Open the page 2)Click on select button 3)Select the input image 4)Check if all the other predictions are displayed	1.png	The other predictions should be displayed properly	Working as expected	Pass		Sheikh Ameenul Haji,Rithish A

## 8.2 User Acceptance Testing

User Acceptance Testing (UAT) is a type of testing performed by the end user or the client to verify/accept the software system before moving the software application to the production environment. UAT is done in the final phase of testing after functional, integration and system testing is done. The main Purpose of UAT is to validate end to end business flow. It does not focus on cosmetic errors, spelling mistakes or system testing. User Acceptance Testing is carried out in a separate testing environment with production-like data setup. It is a kind of black box testing where two or more end-users will be involved. Need of User Acceptance Testing arises once software has undergone Unit, Integration and System testing because developers might have built software based on requirements document by their own understanding and further required changes during development may not be effectively communicated to them, so for testing whether the final product is accepted by client/end-user, user acceptance testing is needed.

### Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By design	1	0	1	1	3
Duplicate	1	0	0	0	1
External	0	0	2	0	2
Fixed	4	1	0	1	6
Not Reproduced	0	0	1	1	2
Skipped	0	0	1	1	2
Won't Fix	1	0	1	0	2
Total	7	1	6	4	18

### Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

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
---------------------	---	---	---	---

## 9. RESULTS

### 9.1 Performance Metrics

#### Model Summary

```
[12]: model.summary()
```

Model: "sequential\_2"

Layer (type)	Output Shape	Param #
conv2d_4 (Conv2D)	(None, 26, 26, 28)	280
max_pooling2d_4 (MaxPooling2D)	(None, 13, 13, 28)	0
conv2d_5 (Conv2D)	(None, 11, 11, 28)	7084
max_pooling2d_5 (MaxPooling2D)	(None, 5, 5, 28)	0
flatten_2 (Flatten)	(None, 700)	0
dense_6 (Dense)	(None, 512)	358912
dropout_4 (Dropout)	(None, 512)	0
dense_7 (Dense)	(None, 128)	65664
dropout_5 (Dropout)	(None, 128)	0
dense_8 (Dense)	(None, 10)	1290

Total params: 433,230  
Trainable params: 433,230  
Non-trainable params: 0

+ Code + Markdown

#### Accuracy

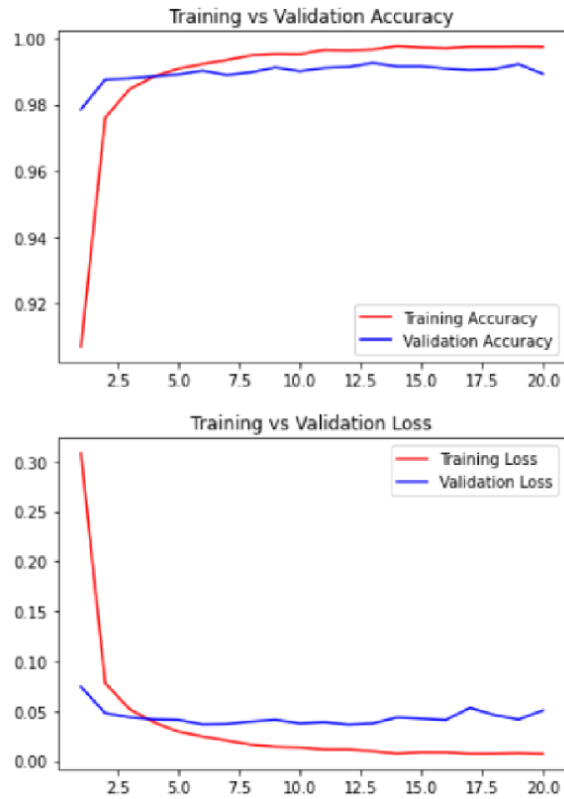
```
> def accuracy(x_train, y_train, model):  
    loss, acc = model.evaluate(train_samples, train_labels, verbose=0)  
    return acc  
  
acc = accuracy(train_samples, train_labels, model)  
print('Train accuracy is, ', acc*100, '%')
```

Train accuracy is, 99.816680339985 %

	Validation Accuracy	Training Accuracy	Validation Loss	Training Loss	Epoch
0	0.978500	0.906652	0.074422	0.306259	1
1	0.987500	0.975963	0.047910	0.078035	2
2	0.987833	0.984092	0.043792	0.051941	3
3	0.988500	0.988280	0.041744	0.038774	4
4	0.990000	0.990778	0.041208	0.029460	5
5	0.990167	0.992185	0.036655	0.024544	6
6	0.988833	0.993285	0.037043	0.020551	7
7	0.989667	0.994778	0.039521	0.016402	8
8	0.991167	0.995241	0.041283	0.014158	9
9	0.990000	0.995148	0.037630	0.013622	10
10	0.991000	0.996463	0.038866	0.011439	11
11	0.991333	0.996315	0.036490	0.011447	12
12	0.992500	0.996374	0.037620	0.009888	13
13	0.991500	0.997574	0.043672	0.007542	14
14	0.991500	0.997167	0.042455	0.008863	15
15	0.990833	0.996944	0.041079	0.008768	16
16	0.990111	0.997385	0.053278	0.007346	17
17	0.990667	0.997370	0.046048	0.007472	18
18	0.992167	0.997426	0.041623	0.007897	19
19	0.989167	0.997370	0.050672	0.007211	20

```
[48]: def accuracy(x_test, y_test, model):  
    loss, acc = model.evaluate(test_samples, test_labels, verbose=0)  
    return acc  
  
acc = accuracy(test_samples, test_labels, model)  
print('Test accuracy is, ', acc*100, '%')
```

Test accuracy is, 99.04000163078308 %



## Confusion matrix

```
[47]: def create_confusion_matrix(true_labels, predicted_labels):

    from sklearn.metrics import confusion_matrix
    cm = confusion_matrix(true_labels.argmax(axis=1), predicted_labels.argmax(axis=1))
    return cm

cm = create_confusion_matrix((train_labels), (predict(train_samples)))
print(cm)
```

```
[[5917  0  2  1  0  0  0  0  1  2]
 [  0 6738  0  1  0  1  2  0  0  0]
 [  0  1 5956  0  0  0  0  0  1  0]
 [  0  0  1 6128  0  1  0  0  1  0]
 [  0  3  0  0 5815  0  1  3  1 19]
 [  0  0  1  7  0 5406  5  0  2  0]
 [  0  0  0  0  1  0 5916  0  1  0]
 [  0  4 16  6  0  0  0 6236  0  3]
 [  0  0  3  8  0  0  0  0 5840  0]
 [  1  0  0  4  1  3  0  0  2 5938]]
```

+ Code

+ Markdown

The confusion matrix gives the performance of our model on a set of test data.

## Classification report

```
[56]: from sklearn.metrics import classification_report, confusion_matrix

print(
    f"Classification report for classifier :\n"
    f"{classification_report(train_labels.argmax(axis=1), predict(train_samples).argmax(axis=1))}\n"
)
```

```
Classification report for classifier :
              precision    recall  f1 score   support
0               1.00      1.00      1.00     5923
1               1.00      1.00      1.00     6742
2               1.00      1.00      1.00     5958
3               1.00      1.00      1.00     6131
4               1.00      1.00      1.00     5842
5               1.00      1.00      1.00     5421
6               1.00      1.00      1.00     5918
7               1.00      1.00      1.00     6265
8               1.00      1.00      1.00     5851
9               1.00      1.00      1.00     5949

accuracy              1.00
macro avg              1.00
weighted avg           1.00
```

## 10. ADVANTAGES & DISADVANTAGES

- **Advantages** of handwritten digit recognition is that we can reduce our work load at certain places at works like a postman can use OCR to pincode in a letter and sort them accordingly, a teacher can use OCR to type students marks in a excel sheet, a bank employee can use OCR to store the challan number in digital storage.
- Extensive noise removal and smoothing is performed before feeding the image to the model, hence the application performs well even on noisy data.
- Since data augmentation is performed on the available dataset, the application performs well on a diverse range of input data making the model more robust.
- Along with the predicted digits, bounding boxes are displayed around the individual digits as an image to the user.
- **Disadvantages** of Handwritten digit recognition is that each person has different handwritten so it becomes difficult for the model classify images correctly hence it becomes necessary that we have to supply more data to the model.
- The application does not perform well on sequences containing partially/fully overlapping digits.
- Since the model has been trained on a standard dataset with images of a fixed shape, the application's performance might be affected when rescaling the image to the same fixed shape.
- It requires much more computation than more standard OCR techniques.
- It is not done in real time as a person writes and therefore not appropriate for immediate text input

## 11. CONCLUSION

There are numerous uses for handwritten digit recognition in the fields of medicine, banking, student administration, taxation, etc. To extract the digit from the handwritten image, a variety of classifiers including KNN, SVM, and CNN are employed. According to the evaluation, CNN performs better than the competition. This study discusses the stages of HDR using a CNN classifier. The MNIST dataset is a common dataset used to evaluate the performance of classifiers. It consists of handwritten numbers from 0 to 9. Three separate stages make up HDR. The first step is preprocessing, which involves converting the dataset into binary format and applying image processing on it. Segmentation, the second stage, involves dividing the image into several pieces. The third stage is feature extraction, during which image features are found. CNN is utilised in the classification stage, which comes last. The CNN classifier greatly enhances the outcomes of HDR, but it is still possible to further enhance the complexity, execution time, and accuracy of the results by combining classifiers or utilising other algorithms in addition to CNN.

## 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.

There is still much work to be done on this project, and it may use a lot of improvement. The following are a few ways this project could be improved:

- Add the ability to save the results of multiple image detection from digits.
- Adding capability to recognise multiple digits
- To detect numbers from complicated images, improve the model.
- Adding support for additional languages will benefit users worldwide. This undertaking has limitless potential and may constantly be improved. By putting this idea into practise in the real world, numerous sectors will gain, many workers' workloads will be reduced, and overall work efficiency will increase

## 13. APPENDIX

Source Code

**1.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 keras.models import load_model
from keras.preprocessing import image
from flask import send_from_directory
from keras.utils import img_to_array
import cv2

UPLOAD_FOLDER = 'C:/Users/DELL/Downloads/'
app=Flask(__name__)
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER

model=load_model("assets/mnistCNN.h5")

@app.route("/")
def homepage():
    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=np.asarray(img)
        img=cv2.resize(img,(28,28),interpolation=cv2.INTER_LINEAR)
        thresh,bw_image=cv2.threshold(img,127,255,cv2.THRESH_BINARY) #convertin
g grayscale to binary image
        bw_image=255-bw_image
        bw_image=img_to_array(bw_image)
        bw_image=np.asarray(bw_image)
        bw_image=np.expand_dims(bw_image,0)
        pred = model.predict(bw_image)

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

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

```

```
if __name__=="__main__":  
    app.run()
```

## 2.index.html

```
<html>  
  
<head>  
  <title>Handwritten Digit Recognition</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&display=s  
wap" rel="stylesheet">  
  <link  
href="https://fonts.googleapis.com/css?family=Calistoga|Josefin+Sans:400,700|Paci  
fico&display=swap" rel="stylesheet">  
  
  <link rel="stylesheet"  
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"  
integrity="sha384-  
gg0yR0iXCbMQV3Xipma34MD+dH/1fQ784/j6cY/iJTQU0hcWr7x9JVoRxT2MZw1T"  
crossorigin="anonymous">  
  <link rel="stylesheet" type= "text/css" href= "{{  
url_for('static',filename='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+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"  
crossorigin="anonymous"></script>  
  <script  
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"  
integrity="sha384-  
U02eT0CpHqdsSJQ6hJty5KVphtPhzWj9WO1c1HTMGA3JDZwrnQq4sF86dIHNDz0W1"  
crossorigin="anonymous"></script>
```

```

    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJGzIxFDs4x0xIM+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"></scri
pt>
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/js/bootstrap.bundle.min.js
"></script>

</head>
<style>
    body{
        /*background-image: url("static/images/Background_Image.jpg");*/
        background-repeat: no-repeat;
        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 p-3 my-3 bg-dark text-white">
        <p style = "text-align:center;font-family:Prompt" > A Novel Method
for Handwritten Digit Recognition System</p>

```

```

        <p style = "text-align:center;font-family:Prompt"> Team ID -
PNT2022TMID35610</P>
    </div>
    <section id="content">

        <div class="leftside">
            <form action="/predict" 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-danger"
id="Predict_button">Predict</button>
                    <button type="button" class="btn btn-danger"
id="clear_button">&nbsp; Clear &nbsp;</button>
                </div>
            </form>
        </div>
    </section>

</body>

</html>

```

### 3.predict.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Prediction</title>
</head>

<style>
    body{
        /*background-image: url('static/images/Background_Image.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;
}

</style>

<body>

    <div id="rectangle">
        <h1 id="head">Predicted Number : <br><center
id="num">{{num}}</center></h1>
    </div>

</body>
</html>

```

GitHub & Project Demo Link

GITHUB LINK→ <https://github.com/IBM-EPBL/IBM-Project-4372-1658730006>

YOUTUBE VIDEO LINK→ <https://www.youtube.com/watch?v=FpgdpiUeq1g>

