# A Novel Method for Handwritten Digit RecognitionSystem

**A PROJECT REPORT**

*Submitted by*


| | |
|---|---|
| 19101066 | R.MUTHUKOWSALYA |
| 19101076 | P.PAVITHRAYALINI |
| 19101083 | P.PRIYADHARSHINI |
| 19101121 | A.YAKSHITHA |


**TEAM ID : PNT2022TMID23749**

*In partial fulfillment for the award of the certification of*

**IBM PROJECT**

*in*

**Artificial Intelligence Domain**



**VIVEKANANDHA COLLEGE OF ENGINEERING FOR WOMEN**

**(AUTONOMOUS)**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

# ACKNOWLEDGMENT

An endeavour over a long period can be successful with the advice and support of many well wishers.We take this opportunity to express our gratitude and appreciation to all of them. We owe our tributes to **Mr.E.Thangadurai**, Assistant Professor/Information Technology, Vivekanandha College of Engineering for women, Tiruchengode, for his valuable support and guidance during the period of project implementation. We wish to express our sincere thanks and gratitude to our project guide **Mrs.Dr.R.NITHYA** AP/CSE , Vivekanandha College of Engineering for women,Tiruchengode, for the simulating discussions, in analysing problems associated with our project work and for guiding us through out the project. Project meetings were highly informative. We express our warm and sincere thanks for the encouragement, untiring guidance and the confidence they had shown in us. We are immensely indebted for their valuable guidance throughout our project. We also thank all the staff members of CSE department for their valuable advices. We also thank supporting staff for providing resources as and when required.

# 1. INTRODUCTION

## 1.1 Project Overview

Machine learning and deep learning plays an important role in Information technology and artificial intelligence. With the use of deep learning and machine learning, human effort can be reduced in recognizing, learning, predictions and many more areas. This article presents recognizing the handwritten digits (0 to 9) from the famous MNIST dataset, comparing classifiers like KNN, PSVM, NN and convolution neural network on basis of performance, accuracy, time, sensitivity, positive productivity, and specificity with using different parameters with the classifiers. To make machines more intelligent, the developers are diving into machine learning and deep learning techniques. A human learns to perform a task by practicing and repeating it again and again so that it memorizes how to perform the tasks. Then the neurons in his brain automatically trigger and they can quickly perform the task they have learned. Deep learning is also very similar to this. It uses different types of neural network architectures for different types of problems.

**For example** – object recognition, image and sound classification, object detection, image segmentation, etc. The handwritten digit recognition is the ability of computers to recognize human handwritten digits. It is a hard task for the machine because handwritten digits are not perfect and can be made with many different flavors. The handwritten digit recognition is the solution to this problem which uses the image of a digit and recognizes the digit present in the image.

## 1.2 Purpose of Digit Recognition System:

Digit recognition system is the working of a machine to train itself or 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 , numeric entries in forms filled up by hand and so on.

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.

# 2. LITERATURE SURVEY

## 2.1 Existing Problem

An early notable attempt in the area of character recognition research is by Grimsdale in 1959. The origin of a great deal of research work in the early sixties was based on an approach known as analysis by-synthesis method suggested by Eden in 1968. The great importance of Eden's work was that he formally proved that all handwritten characters are formed by a finite number of schematic features, a point that was implicitly included in previous works.

## 2.2 Reference

**1. Ahlawat Savita, Amit Choudhary,** The aim of this paper is to develop a hybrid model of a powerful Convolutional Neural Networks (CNN) and Support Vector Machine (SVM) for recognition of handwritten digit from MNIST dataset. The proposed hybrid model combines the key properties of both the classifiers. In the proposed hybrid model, CNN works as an automatic feature extractor and SVM works as a binary classifier. The MNIST dataset of handwritten digits is used for training and testing the algorithm adopted in the proposed model. The MNIST dataset consists of handwritten digits images which are diverse and highly distorted. The receptive field of CNN helps in automatically extracting the most distinguishable features from these handwritten digits. The experimental results demonstrate the effectiveness of the proposed framework by achieving a recognition accuracy of 99.28% over MNIST handwritten digits dataset.

**2. Ali Abdullah Yahya, Min Hu,** An enormous number of CNN classification algorithms have beenproposed in the literature. Nevertheless, in these algorithms, appropriate filter size selection, data preparation, limitations in datasets, and noise have not been taken into consideration. As a consequence, most of the algorithms have failed to make a noticeable improvement in classification accuracy. To address the shortcomings of these algorithms, our paper presents the following contributions: Firstly, after taking the domain knowledge into consideration, the size of the effective receptive field (ERF) is calculated. Calculating the size of the ERF helps us to select atypical filter size which leads to enhancing the classification accuracy of our CNN. our CNN algorithm achieves state-of-the-art results in handwritten digit recognition, with a recognition accuracy of 99.98%, and 99.40% with 50% noise.

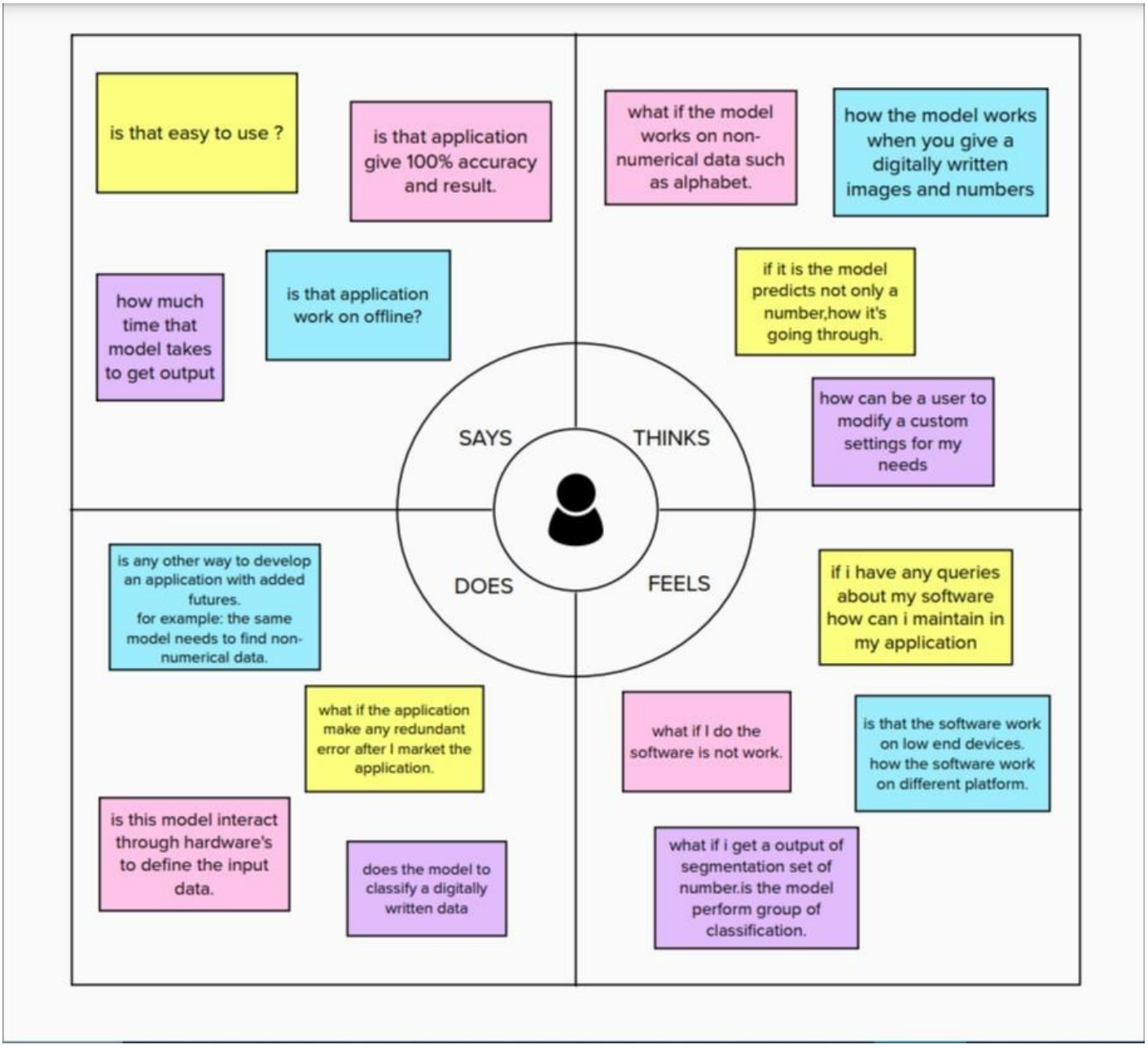
## 2.3 Problem Statement Definition

The goal of this project is to create a model that will be able to recognize and determine the handwritten digits from its image by using the concepts of Convolution Neural Network. Though the goal is to create a model which can recognize the digits, it can be extended to letters and an individual's handwriting. The major goal of the proposed system is understanding Convolutional Neural Network, and applying it to the handwritten recognition system.

# 3. IDEATION & PROPOSED SOLUTION

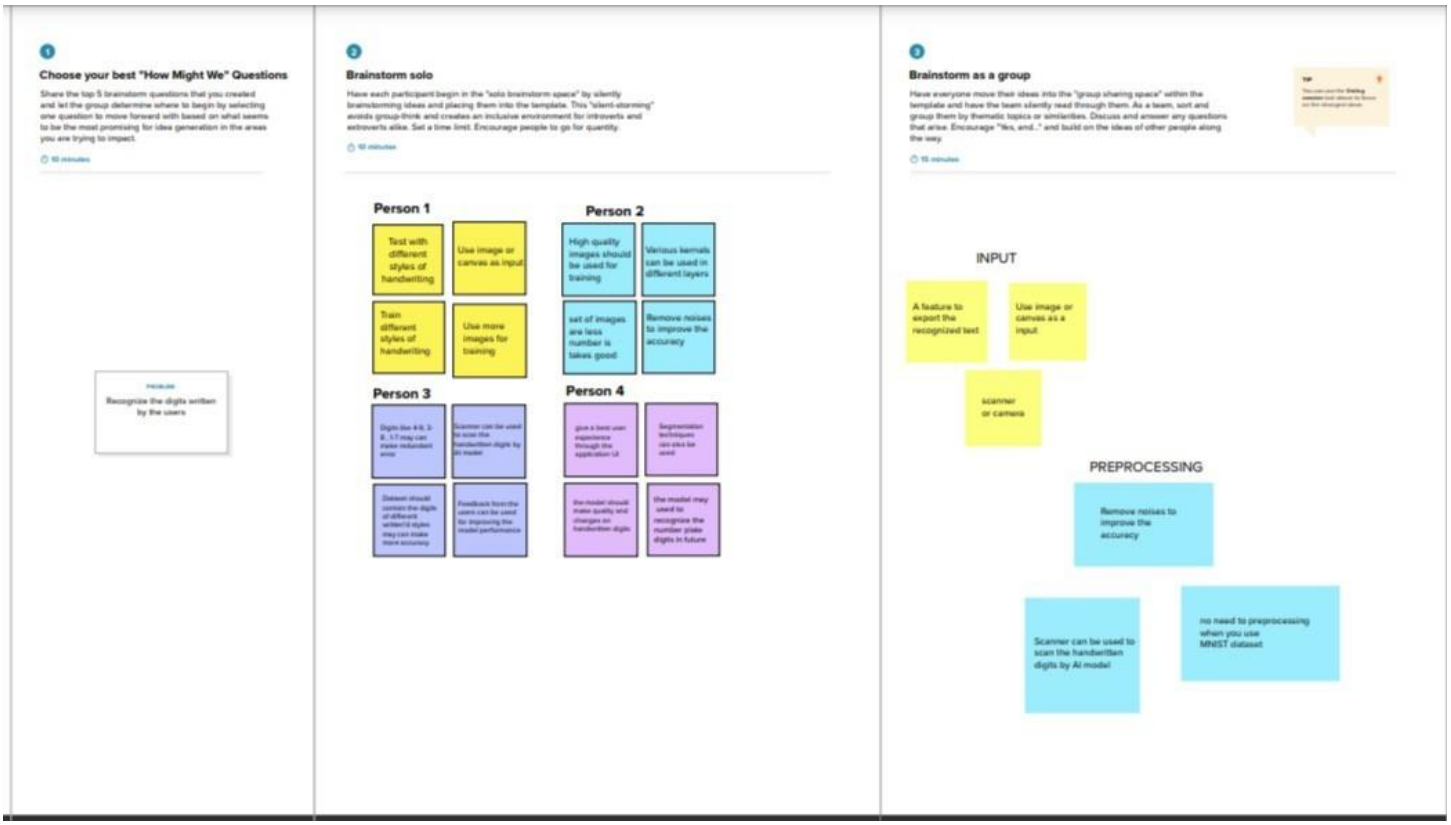
## 3.1 Empathy Map Canvas

An empathy map is **a collaborative tool teams can use to gain a deeper insight into their customers**. Much like a user persona, an empathy map can represent a group of users, such as a customer segment. The empathy map was originally created by Dave Gray and has gained much popularity within the agile community.



## 3.2 Ideation & Brainstorming

**Ideation** refers to the whole creative process of coming up with and communicating new ideas.Brainstorming is **a group problem-solving method that involves the spontaneous**
**contribution of creative ideas and solutions**. This technique requires intensive, freewheeling discussion in which every member of the group is encouraged to think aloud and suggest as many ideas as possible based on their diverse knowledge.




## 3.3 Proposed Solution

The handwritten digit recognition is the capability of computer applications to recognize the human handwritten digits. It is a hard task for the machine because handwritten digits are not perfect and can be made with many different shapes and sizes. The handwritten digit recognition system is a way to tackle this problem which uses the image of a digit and recognizes the digit present in the image. Convolutional Neural Network model created using TensorFlow.Keras library over the MNIST dataset to recognize handwritten digits.

**IDEA/SOLUTION DESCRIPTION:**

MNIST database contains 60,000 training images of handwritten digits from zero to nine and 10,000 images for testing. We will create our CNN model. It works better for data that are represented as grid structures; this is the reason why CNN works well for image classification problems. we will load our model. Then we will use ImagedataGenerator to get to the test folder and call the image files collectively. And will call the predict function and set it with the image generator function. Now, we will plot our results along with the images.

**NOVELTY/UNIQUENESS:**

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 have taken this a step further where are handwritten digit recognition system not only detects the scanned images of handwritten digits but also allows writing digits on the screen with the help of an Integrated GUI for recognition.

**SOCIAL IMPACT/CUSTOMER SATISFACTION:**

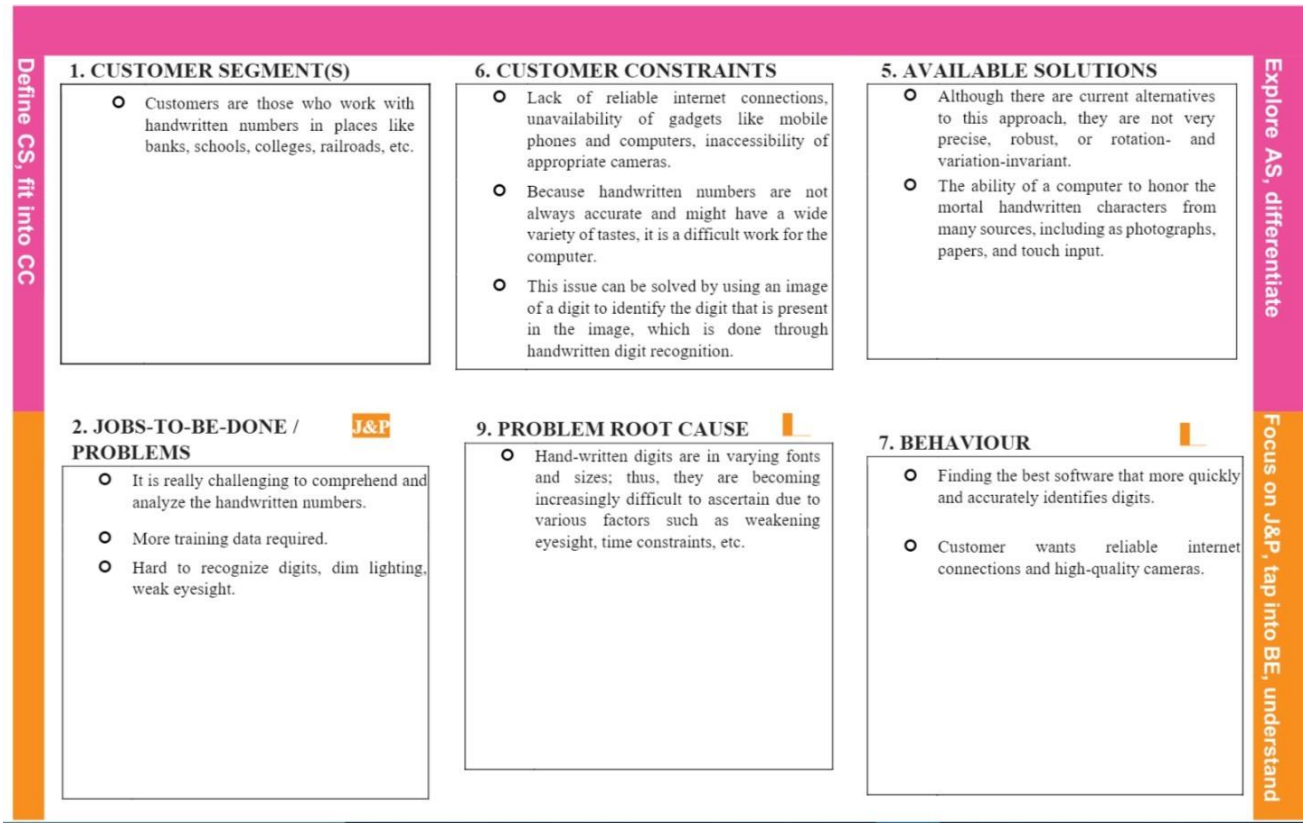
1. Digital Recognition is nothing other than recognizing or identifying digits in any document. The framework of digital recognition is simply the operation of the machine to prepare or interpret digits.
2. Handwritten Digit Recognition is the power of computers to translate handwritten digits from a variety of sources such as text messages, bank checks, papers, photos, etc. method With the use of in-depth learning methods, human efforts can be reduced in perception, learning, perception and in too many regions.
3. Using in-depth learning, the computer learns to perform distinctive functions in images or content anywhere accuracy, in addition to the performance of the human level. The digital recognition model uses large data sets to detect digits from different sources.

**BUSINESS MODEL:**

> Handwritten digit recognition refers to a model's (machine's) capacity to detect any handwritten digits from various sources, such as photographs, papers, and touch displays, and classify them into ten specified categories 0-9.

> We used the CNN (Convolutional Neural network) algorithm to recognize handwritten digits in this project.

## 3.4 Problem Solution Fit



# 4. Requirements Analysis

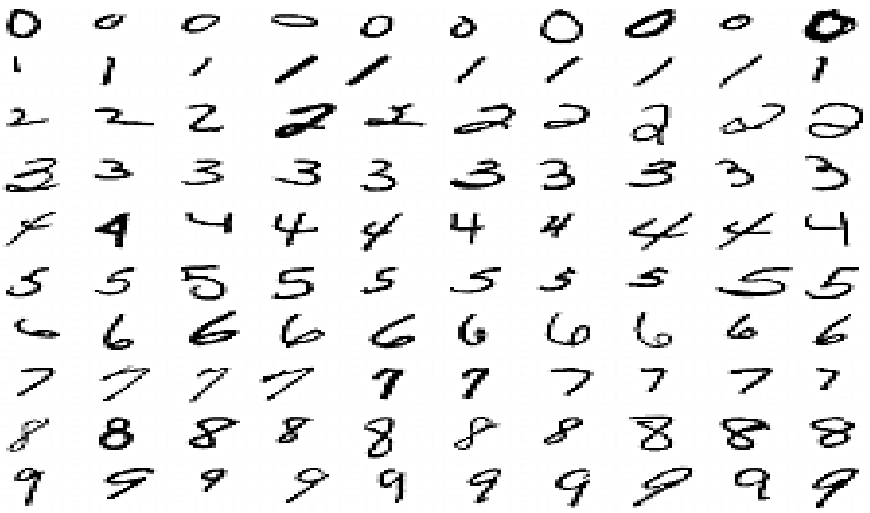
## 4.1 Functional Requirements:

1. **Image Data:** The handwritten digit recognition is **the ability of computers to recognize human handwritten digits**. It is a hard task for the machine because handwritten digits are not perfect andcan be made with many different flavors.

2. **Hosting** (also known as Web site hosting, Web hosting, and Webhosting) is **the business ofhousing, serving, and maintaining files for one or more Web sites**. More important than the computer space that is provided for Web site files is the fast connection to the Internet.

3. **Use the MNIST database of handwritten digits to train a convolutional network to predict thedigit given an image**. First obtain the training and validation data.

4. **MNIST Dataset:** The MNIST dataset is an acronym that stands for the **Modified National Institute of Standards and Technology dataset**. It is a dataset of 60,000 small square 28×28 pixelgrayscale images of handwritten single digits between 0 and 9.



5. **Cloud**: The cloud provides a number of IT services such as servers,databases, software, virtual storage, and networking, among others. In layman's terms, Cloud Computing is defined as a virtual platform that allows you to store and access your data over the internet without any limitations.

## 4.2 Non-Functional Requirements

**Usability:**

Handwritten character recognition is one of the practically important issues in pattern recognition applications. The applications of digit recognition include in postal mail sorting, bank check processing, form data entry, etc.

**Reliability:**

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.

**Performance**:

the neural network uses the examples to automatically infer rules for recognizing handwritten digits. Furthermore, by increasing the number of training examples, the network can learn more about handwriting, and so improve its accuracy. There are a number of ways and algorithms to recognize handwritten digits, including Deep Learning/CNN, SVM, Gaussian Naive Bayes, KNN, Decision Trees, Random Forests, etc.

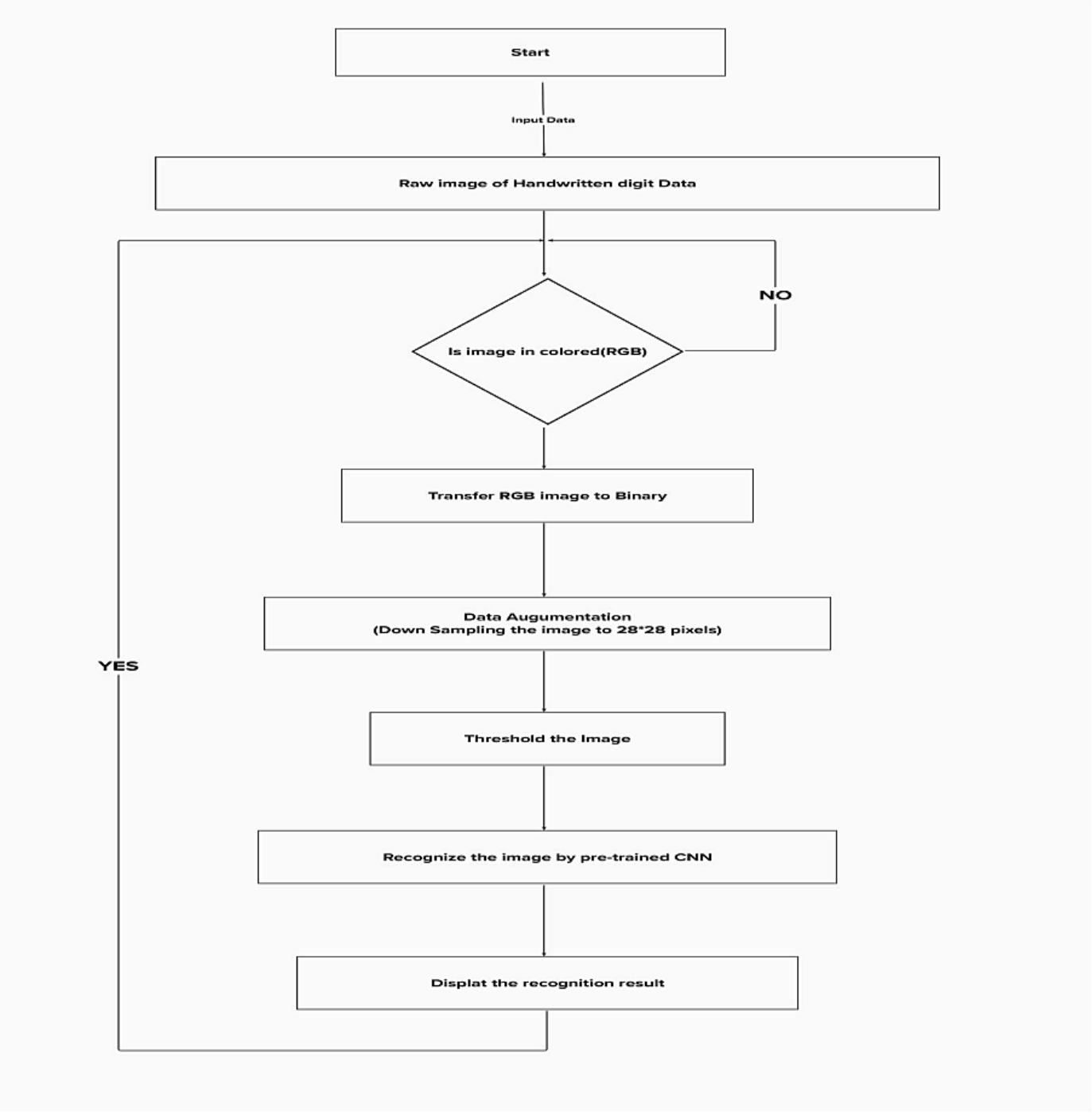
**Accuracy:**

MNIST - Classification of digits (accuracy=98%).

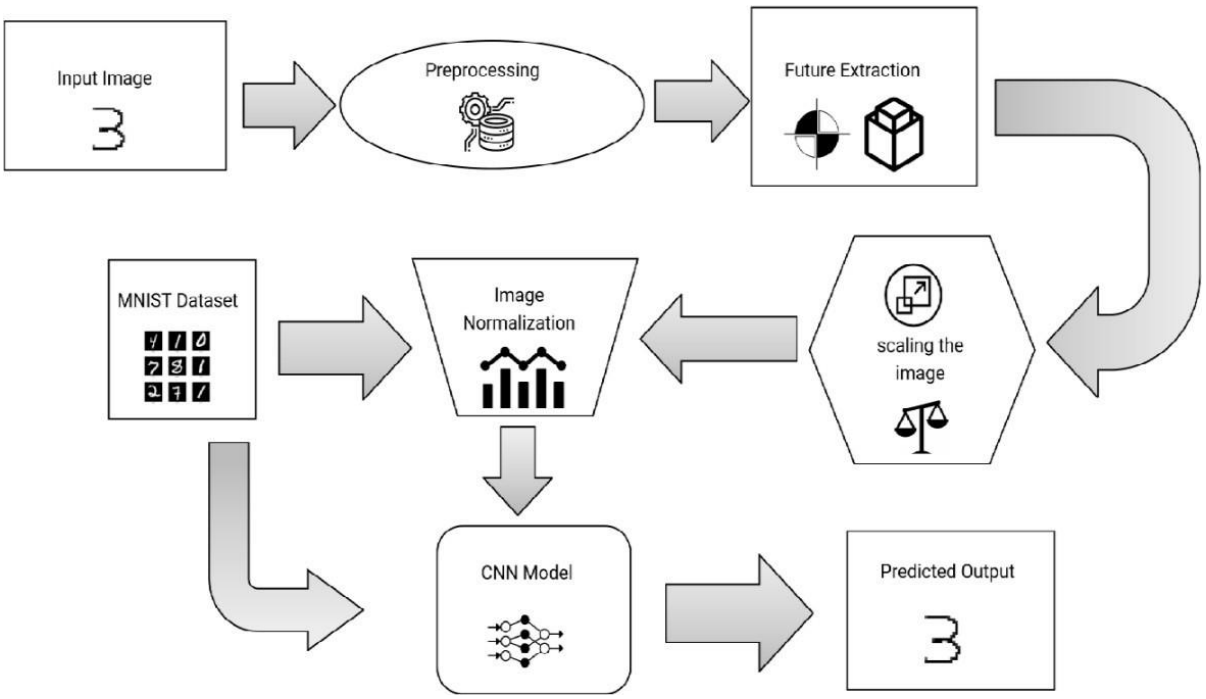
The MNIST dataset is an image dataset of handwritten digits. It has has 60,000 training images and 10,000 test images, each of which are grayscale 28 x 28 sized images.

# 5. PROJECT DESIGN

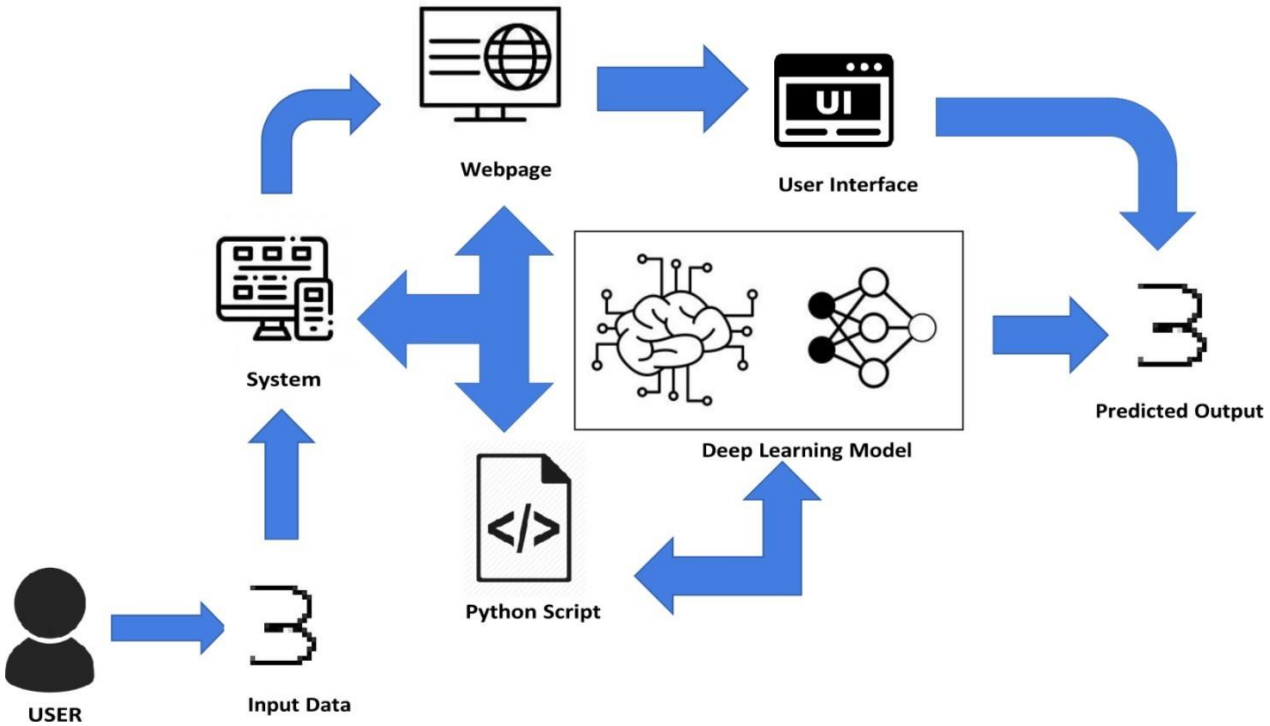
## 5.1 DataFlow Diagram



## 5.2 Solution & Technical Architecture




## ii.Interactive Technology Architecture



## 5.3 User Stories

### 1. Custome

#### ri.Home

User 1.As a user, I can view the guide and awareness to use this application.
Acceptence Criteria - I can view the awareness to use this application and its limitations.

User 2. As a user, I'm allowed to view the guided video to use the interface of this application.
Acceptance Criteria - I can gain knowledge to use this application by a practical method.

User 3.As a user, I can read the instructions to use this application.
Acceptance Criteria - can read instructions also to use it in a user-friendly method.

#### ii. Home

User 4. As a user, In this prediction page I get to choose the image.
Acceptance Criteria - I can choose the image from our local system and predict the output.

#### iii. Predict

User 5. As a user, I'm Allowed to upload and choose the image to be uploaded.
Acceptance Criteria - I can upload and choose the image from the system storage and also in any virtual storage.

User 6. As a user, I will train and test the input to get the maximum accuracy of output.
Acceptance Criteria - I can able to train and test the application until it gets maximum accuracy of the result.

User 7. As a user, I can access the MNIST data set.
Acceptance Criteria - I can access the MNIST data set to produce the accurate result.

User 8. As a user, I can view the guide to use the web application.
Acceptance Criteria - I can view the awareness of this application and its limitations.

# 6. Project Planning & Scheduling

## 6.1 Sprint Planning & Estimation

| TITLE | DESCRIPTION | DATE |
|---|---|---|
| Prepare Empathy Map | Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements | 24 SEPTEMBER 2022 |
| Literature Survey & Information Gathering | Literature survey on the selected project & gathering information by referring the, technical papers,research publications etc. | 28 SEPTEMBER 2022 |
| Ideation | List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance. | 25 SEPTEMBER 2022 |
| Proposed Solution | Creation of proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc. | 23 SEPTEMBER 2022 |
| Problem Solution Fit | Creation of problem solution fit document. | 30 SEPTEMBER 2022 |
| Solution Architecture | Creation of solution architecture document. | 28 SEPTEMBER 2022 |
| Customer Journey | Prepare the customer journey maps to understand the user interactions & experiences with the application. | 20 OCTOBER 2022 |
| Data Flow Diagrams | Draw the data flow diagrams and submit for review. | 9 OCTOBER 2022 |
| Technology Architecture | Prepare the technology architecture diagram. | 10 OCTOBER 2022 |

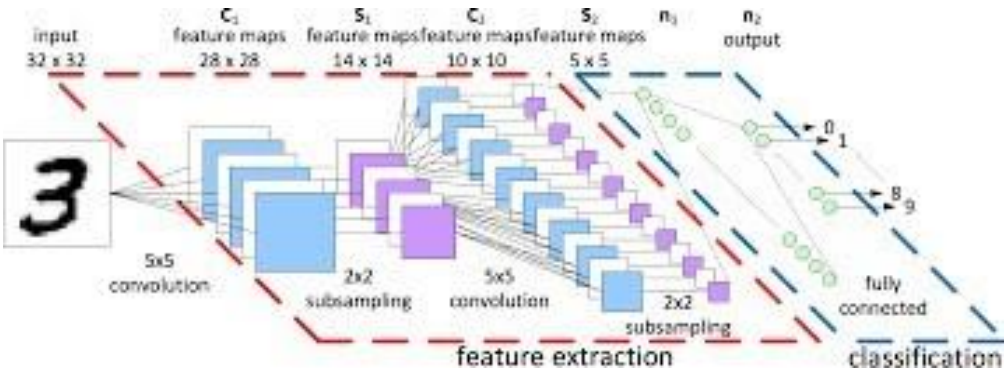
## 6.2 Sprint Delivery Schedule

| 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<br>or<br>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. |

## 7. CODING & SOLUTIONING

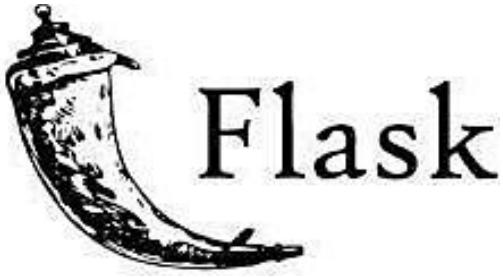
### 7.1 Feature [1]

**i).Using CNN Model in our Project** : **CNN** is basically a model known to be **Convolutional NeuralNetwork** and in recent times it has gained a lot of popularity because of its usefulness. CNN uses multilayer perceptrons to do computational works.
ii).CNN uses relatively little pre-processing compared to other image classification algorithms.This means the network learns through filters that in traditional algorithms were hand- engineered. So, for the image processing tasks CNNs are the best-suited option.




### 7.2 Feature [2]

**ii).Using Flask application in our Project** : Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.



# 8. TESTING

## 8.1 Test Cases

Testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is defect free. It involves the execution of a software component or system component to evaluate one or more properties of interest. Software testing also helps to identify errors, gaps, or missing requirements in contrary to the actual requirements.

**i).Unit Testing:**

When the testing happens for some individual group or some related units then that type of testing is called as Unit Testing.It is often done by programmer to test the part of the program he or she has implemented.

Unit Testing is successful means all the modules has been successfully tested and it can proceed further.

**ii).Functional Testing:**

This type of testing is tested because to check the functional components or the functionality required from the system is gained or not .It actually falls under the testing of the Black Box testing of Software Engineering.This part includes the feeding of the inputs in the system or the project and to check if that system or the project is getting the same value or not as expected if not then calculate the error as wanted and check for more.Functional Testing of this project mainly involves below things. All of these are tested successfully and errors are also calculated.

i)Verifyng the input image

ii)Verifying the work flow

iii)Correct recognition and calculate the error

**iii).Integration Testing:**

In a total project or the system, many groups of components are getting added or summed up in the purpose of the project query. Integration testing is about to check the interaction between various modules of the project or the system. This module also includes the hardware and the software requirements of the project.

All the individual modules are integrated and tested together.All the best and extreme cases that the modules are interacting or not are successfully checked and passed,errors are calculated for the deep learning platforms.

**iv).System Testing:**

This type of testing is actually meant for the system or the project and also the platform and the integrated softwares and tools,technologies are also tested.The idea or purpose behind the system testing is to check all the requirements that will be provided by the system. This application of the project along with the tools and technologies has been tested in both windows and linux.It passed successfully.

## 8.2 User Accceptance Testing

This is a type of system or software testing where a system has been tested for availability.The purpose of this test is to check the business requirements and assess whether it will be accepted for delivery.

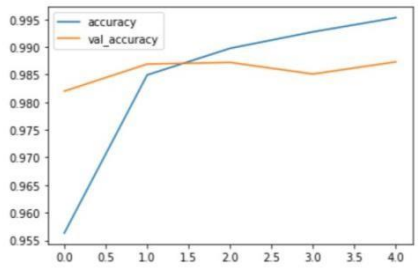
# 9. RESULTS

## 9.1 Performance Metrics

**i).Model Metrics :** Our model perform 98% of accuracy when train and testing session.

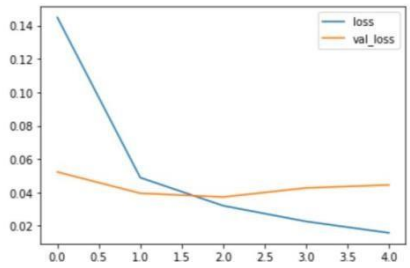
In [ ]:
```
losses[['accuracy','val_accuracy']].plot()
```

Out[ ]:




In [ ]:
```
losses[['loss','val_loss']].plot()
```

Out[ ]:




In [ ]:
```
print(model.metrics_names)
print(model.evaluate(x_test,y_cat_test,verbose=0))
```

```
['loss', 'accuracy']
[0.044522497802972794, 0.9872999787330627]
```

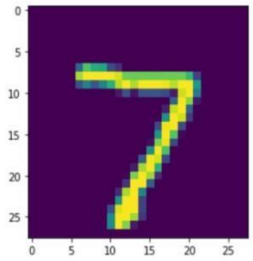
**ii).Overall Application Performance**

## Predicting a given image

In [ ]:
```
my_number = x_test[0]
```

In [ ]:
```
plt.imshow(my_number.reshape(28,28))
```

Out[ ]:

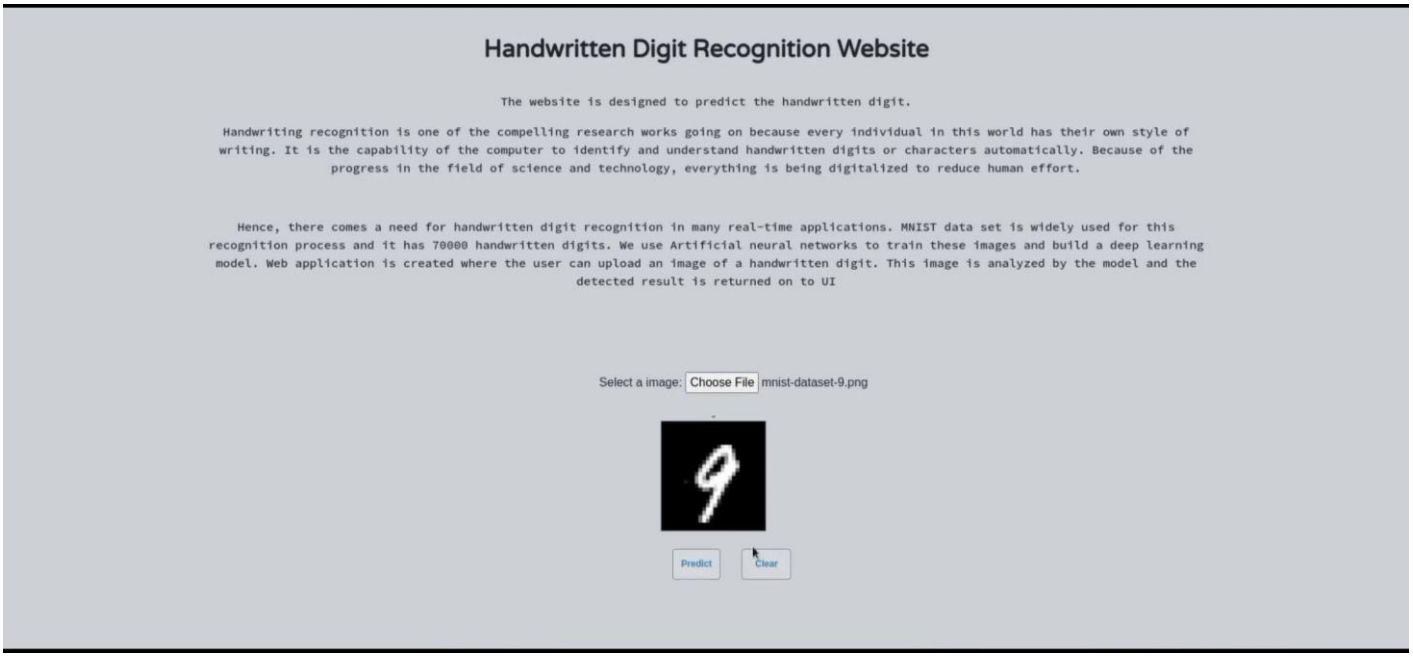



In [ ]:
```
# SHAPE --> (num_images,width,height,color_channels)
model.predict(my_number.reshape(1,28,28,1))
```

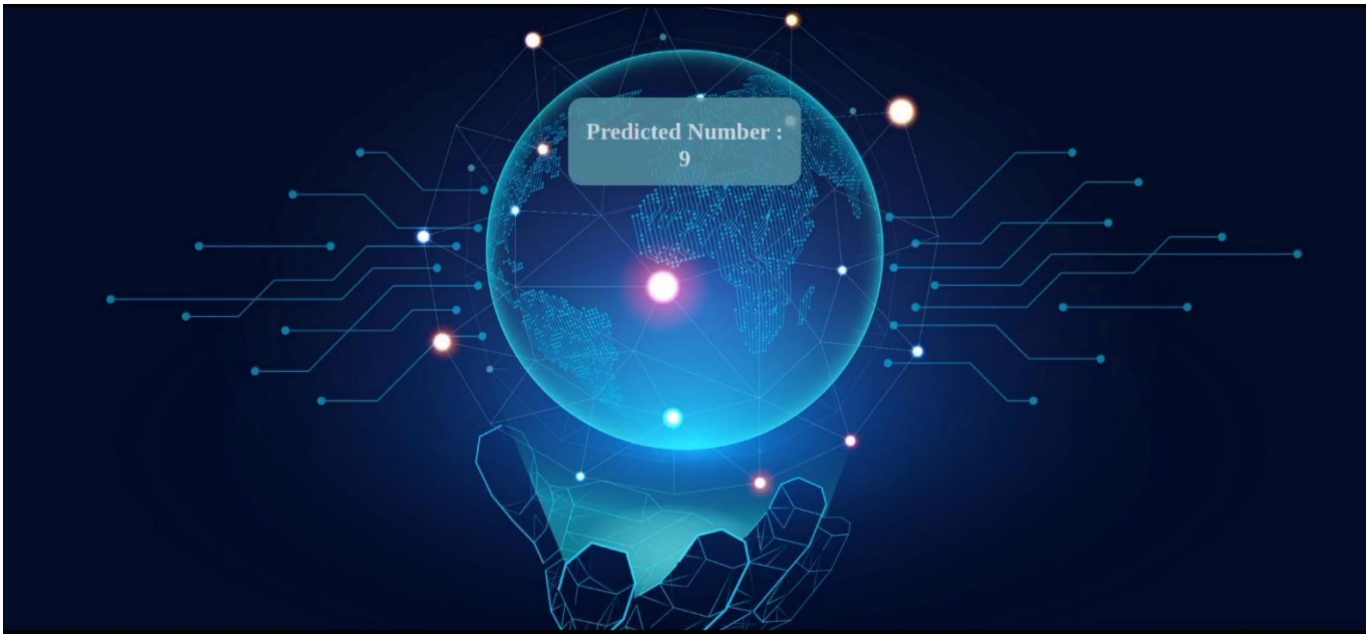
```
1/1 [==============================] - 0s 42ms/step
```

Out[ ]:
```
array([[1.1048161e-12, 2.5010433e-10, 8.5665883e-09, 6.9659154e-07,
        2.0304635e-12, 6.1936345e-11, 3.0275941e-15, 9.9999928e-01,
        2.5742997e-10, 1.9984461e-09]], dtype=float32)
```

**i).Index of the Application**




**ii).Prediction Page**



## 10. CONCLUSION

- ✓ the Handwritten Digit Recognition using Deep learning methods has been implemented. CNN have been trained and tested on the same data in order acquire the comparison between the classifiers. Utilising these deep learning techniques, a high amount of accuracy can be obtained.
- ✓ Compared to other research methods, this method focuses on which classifier works better by improving the accuracy of classification models by more than 99%.
- ✓ Using Keras as backend and Tensorflow as the software, a CNN model is able to give accuracy of about 98.72%.

## 11. Future Work

**The proposed system takes 28x28 pixel sized images as input. The same system with further modifications and improvements in the dataset and the model can be used to build Handwritten**
**Character Recognition System which recognizes human handwritten characters and predicts theoutput.**

# 12. APPENDIX

**Python:**

Python is an interpreted, high-level, general purpose programming language created by Guido Van Rossum and first released in 1991, Python's design philosophy emphasizes code Readability with its notable use of significant White space. Its language constructs and object oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed and garbage collected. It supports multiple programming paradigms, including procedural, objectoriented, and functional programming.

**Keras:**

Keras is a powerful and easy-to-use free open source Python library for developing and evaluating deep larning models. It wraps the efficient numerical computation libraries Theano and TensorFlow and allows you to define and train neural network models in just a few lines of code. It uses libraries such as Python, C#, C++ or standalone machine learning toolkits. Theano and TensorFlow are very powerful libraries but difficult to understand for creating neural networks. Keras is based on minimal structure that provides a clean and easy way to create deep learning models based on TensorFlow or Theano. Keras is designed to quickly define deep learning models. Well, Keras is an optimal choice for deep learning applications. Steps for creating a keras model:

1) First we must define a network model.
2) Compile it, which transforms the simple sequence of layers into a complex group of matrix operations.
3) Train or fit the network. To import: from keras.models import Sequential fromkeras.layers import Dense, Activation, Dropout

**TensorFlow:**

TensorFlow is a Python library for fast numerical computing created and released by Google. It is a foundation library that can be used to create Deep Learning models directly or by using wrapper libraries that simplify the process built on top of TensorFlow. TensorFlow tutorial is designed for both beginners and professionals. Our tutorial provides all the basic and advanced concept of machine learning and deep learning concept such as deep neural network, image processing and sentiment analysis.TensorFlow is one of the famous deep learning frameworks, developed by Google Team. It is a free and open source software library and designed in Python programming language, this tutorial is designed in such a way that we can easily implements deep learning project on TensorFlow in an easy and efficient way. Unlike other numerical libraries intended for use in Deep Learning like Theano, TensorFlow was designed for use both in research and development and in production systems. It can run on single CPU systems, GPUs as well as mobile devices and largescale distributed systems of hundreds of machines.

**Numpy:**
NumPy is a Python library used for working with arrays. It also has functions for working in domainof linear algebra, Fourier transform, and matrices. Numpy which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed. This tutorial explains the basics of NumPy such as its architecture and environment. It also discusses the various array functions, types of indexing, etc. It is an open source project and you can use it freely. NumPy stands for Numerical Python. NumPy aims to provide an array object that is up to 50x faster than traditional Python lists. The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy. Arrays are very frequently used in data science, where speed and resources are very important.

**Jupyter Notebook:**
JupyterLab is a web-based interactive development environment for Jupyter notebooks, code, and data. JupyterLab is flexible: configure and arrange the user interface to support a wide range of workflows in data science, scientific computing, and machine learning. JupyterLab is extensible and modular: write plugins that add new components and integrate with existing ones.

**Machine Learning:**
Machine learning is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention.

**Deep Learning:**
Deep learning is an artificial intelligence (AI) function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Deep learning is a subset of machine learning in artificial intelligence that has networks capable of learning unsupervised from data that is unstructured or unlabeled. Also known as deep neural learning or deep neural network.

**Neural Networks:** A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. In this sense, neural networks refer to systems of neurons, either organic or artificial in nature.

# SOURCE CODE

## 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  =  'C:/Users/ABISHEK/Desktop/selva  ibm/Project/uploads'


app = Flask(_name_)
app.config['UPLOAD_FOLDER'] =
UPLOAD_FOLDER

model  =  load_model("C:/Users/ABISHEK/Desktop/selva  ibm/Project/CNN-MNIST.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)
```

## index.html

```
<html>

<head>
  <title>Digit Recognition WebApp</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=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='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>

</head>
<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>

  <h1 class="welcome">IBM PROJECT
  <div id="team_id">TEAM ID : PNT2022TMID46095</div>
  </h1>
  <section id="title">
    <h4 class="heading">Handwritten Digit Recognition</h4>
    <br><br>
  </section>

  <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" src="" 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">  Clear  </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('static/images/image.jpg');
     background-repeat: no-repeat;
     background-size: cover;
    }

    #rectangle{
     width:250px
     height:100px;
     background-color: #20e4ff;
     border-radius: 25px;
     position:absolute;
     text-align:center;
     top:50%;
     left:50%;
     transform:translate(-50%,-50%);
    }

    #ans{
  text-align: center;
  font-size: 40px;
  margin: 0 auto;
  padding: 3% 5%;
  padding-top: 15%;
  color: white;
```

```
    }

</style>
<body>
    <div id="rectangle">
        <h1 id="ans">Predicted Number : {{num}}</h1>
    </div>
</body>
</html>
```

# GITHUB LINK

https://github.com/IBM-EPBL/IBM-Project-36389-1660294850