A NOVEL METHOD FOR HANDWRITTEN DIGIT RECOGNITION

A PROJECT REPORT

Submitted By

JELCY EVANGELIN.A	210819205018
JEYASRI POOJA.S	210819205021
JENIFER.R	210819205019
AARTHI S.A	210819205002

In partial fulfillment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

In

INFORMATION TECHNOLOGY

KINGS ENGINEERING COLLEGE, IRUNGATTUKOTAI

ANNA UNIVERSITY:CHENNAI 600025

November 2022

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that project based learning report **“A NOVEL METHOD FOR HANDWRITTEN DIGIT RECOGNITION”** is bonafide work of **“JELCY EVANGELIN, JEYASRI POOJA, JENIFER, AARTHI”** who carried out this project based learning work under my supervision.

SIGNATURE	**SIGNATURE**
Dr.G.MANIKANDAN	MR.J.BRISO BECKY BELL
HEAD OF THE DEPARTMENT	**MENTOR**
Professor	**Assistant Professor**
Dept of Information Technology,	Dept of Information Technology,
Kings Engineering College,	Kings Engineering College,
Irungattukottai,	Irungattukottai,
Chennai-602 117	Chennai-602 117.

ACKNOWLEDGEMENT

We thank God for his blessings and also for giving as good knowledgeand strength in enabling us to finish our project. Our deep gratitude goes to our founder late **Dr. D. SELVARAJ, M.A., M.Phil.,** for his patronage in the completion of our project. We like to take this opportunity to thank our honourable chairperson **Dr.S. NALINI SELVARAJ, M.COM., MPhil., Ph.D.** and honourable director, **MR.S. AMIRTHARAJ, M.Tech., M.B.A** for their support given to us to finish our project successfully. We wish to express our sincere thanks to our beloved principal. **Dr.T. JOHN ORAL BASKAR., M.E., Ph.D** for his kind encouragement and his interest towards us.

We are extremely grateful and thanks to our professor **Dr.G.MANIKANDAN,** head of the Information Technology department, Kings Engineering College, for his valuable suggestion, guidance and encouragement. We wish to express our sense of gratitude to our project Mentor **Mr.J.BRISO BECKY BELL,** Assistant Professor of Information Technology Department, Kings Engineering College and our project Evaluator **Dr.M.ROBINSON JOEL ,** Associate Professor of Information Technology Department, Kings Engineering College, with their guidance and direction made our project a grand success. Also we thank our industrial Mentor **Ms.PRADEEPTHI,** for tuning this project as remarkable one. We Also Thank our College Experimental Project Based Learning SPOC **Dr. Mrs. M. PARAMESHWARI** for managing this initiative in correct direction. We express our sincere thanks to our parents, friends and staff members, who have helpedand encouraged us during the entire course of completing this project work successfully.

PROBLEM STATEMENT:

Handwritten digit recognition has recently been of very interest among the researchers because of the evolution of various Machine Learning, Deep Learning and Computer Vision algorithms. In this report, We compare the results of some of the most widely used Machine Learning Algorithms convolution neural networks and with Deep Learning algorithm like multilayer CNN using Keras with Theano and Tensorflow. MNIST is a dataset which is widely used for handwritten digit recognition. The dataset consist of 60,000 training images and 10,000 test images.The artificial neural neworks can all most mimic the human brain and are a key ingredient in image processing field.For example Convolution Neural networks with back propagation for image processing.The applications where these handwritten digit recognition can be used are 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.

. .

„

TABLE OF CONTENTS

,,

,,

CHAPTER 1

INTRODUCTION

1.1 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 recognize 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.

1.2 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 and so on.

CHAPTER 2

LITERATURE SURVEY

2.1 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 individuals handwriting influence the structure and appearance of the digits.

2.2 REFERENCES

An artificial neural network has one hidden layer between the input and output layers, whereas a deep neural network has numerous hidden layers with input and output layers. Deep neural networks use several hidden layers to increase model performance and achieve higher accuracy compared to accuracy of machine learning models.

Most researchers do their research in the area of pa1ern recognition. In the field of pa1ern recognition,there are many pa1erns that can be used, including handwri1en numbers, characters, pictures, faces,sounds, and speech. This study focuses on the classification and recognition of handwri1en digits.1000 were utilized as test samples and 1000 were training samples.10000 picture samples make up the USPS dataset, of which 7291 serve as training samples and 2007 serve as testing samples. We've used the proposed deep neural network technique in this paper to classify and identify data from the ARDIS and USPS datasets. The

,,

suggested model consists of six layers with so𝑓max and relu activation functions. A𝑓er model implementation, accuracy for ARDIS samples reached 98.70% testing and 99.76% training, which is greater than accuracy from prior research. Additionally, using the USPS10 samples dataset, 98.22% training accuracy and 93.01% testing accuracy were a1ained. When compared to earlier methodologies, the data show that deep neural networks perform incredibly well.

Recognition of isolated and simply connected handwritten numerals, Pattern Recognition. (1986)

M. Shridhar and A. Badreldin

In this paper the authors describe the results of their investigation into the development of a recognition algorithm for identifying numerals that may be isolated or connected, broken or continuous. Using a structural classification scheme, the recognition algorithm is derived as a tree classifier. In an extensive test experiment, an accuracy of 99% was realized with isolated numerals. When connected numerals were also included a recognition accuracy of 93% was obtained.

Handwritten Character Recognition using Neural Network and TensorFlow (2019)

Megha Agarwal, Shalika, Vinam Tomar, Priyanka Gupta

The offline handwri1en character recognition in this study will be carried out using Tensorflow and a convolutional neural network. a process known as using So𝑓Max Regression, one may assign probabilities to one of the many characters in the handwri1en text that offers the range of values from 0 to 1, summed to

,,

1. The objective is to create so*f*ware that is extremely accurate and that has a minimum level of spatial and temporal complexity. It was determined that strategies for feature extraction like diagonal and direction are significantly be1er at producing high accuracy. Outcomes in comparison to other conventional vertical and horizontal techniques moreover use the best Neural network tried layers provides the benefit of a higher accurate outcome by having a high noise tolerance. The feed forward model in neural networks is the back-propagation algorithm that was primarily used to classify the characters, recognise them, and receive training continually more. In addition to these, normalizing along with feature extraction, the results were be1er and more effective. Character recognition is the outcome of accuracy. The paper will describe the best approach to get more than 90% accuracy in the field of Handwri1en Character Recognition (HCR).

Handwritten Digit Recognition of MNIST dataset using Deep learning state-of-the-art Artificial Neural Network (ANN) and Convolutional Neural Network (CNN) (2021)

Drishti Beohar, A. Rasool

Handwritten digit recognition is an intricate assignment that is vital for developing applications, in computer vision digit recognition is one of the major applications. There has been a copious exploration done in the Handwri1en Character Recognition utilizing different deep learning models. Deep learning is rapidly increasing in demand due to its resemblance to the human brain. The two major Deep learning algorithms Artificial Neural Network and Convolutional Neural Network which have been compared in this paper considering their feature extraction and classification stages of recognition. The models were trained using categorical cross-entropy loss and ADAM optimizer on the MNIST dataset. Backpropagation along with Gradient Descent is being used to train the networks along with reLU activations in the network which do automatic feature extraction. In neural networks, Convolution Neural Network (ConvNets or Convolutional neural networks) is one of the primary classifiers to do image recognition, image classification tasks in

„

Computer Vision.

2.3 PROBLEM STATEMENT DEFINITION

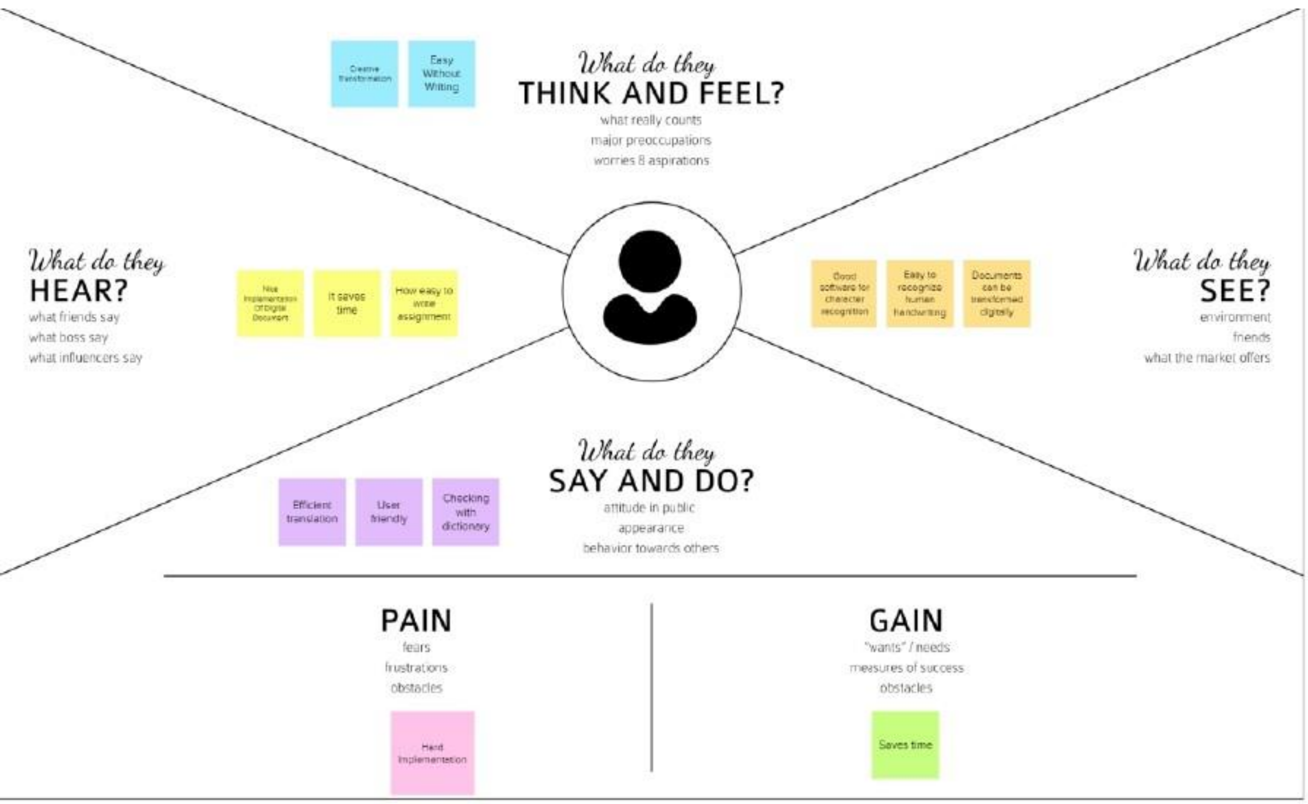
Handwritten digit recognition is the ability of computers to recognize human handwritten digits. It is a hard task for the machine because handwritten digits are not perfect and can be made with many different flavors. The handwritten digit recognition is the solution to this problem which uses the image of a digit and recognizes the digit present in the image. 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 and so on.

CHAPTER 3

IDEATION AND PROPOSED SOLUTION

„

3.1 EMPATHY MAP CANVAS



3.2 IDEATION AND BRAINSTORMING

,,

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!

JELCY EVANGELIN

- Training data set
- Data set should be gathered
- For input extract factor

JENIFER.R

- Test data set
- Handwriting image input
- Features compared with neural network model

JEYASRI POOJA.S

- Machine learning for training data set
- Convolution neural network for learning the digit in data set
- OCR test data with training data

AARTHI.S.A

- Image classification
- OCR test data with training data
- OCR recognition

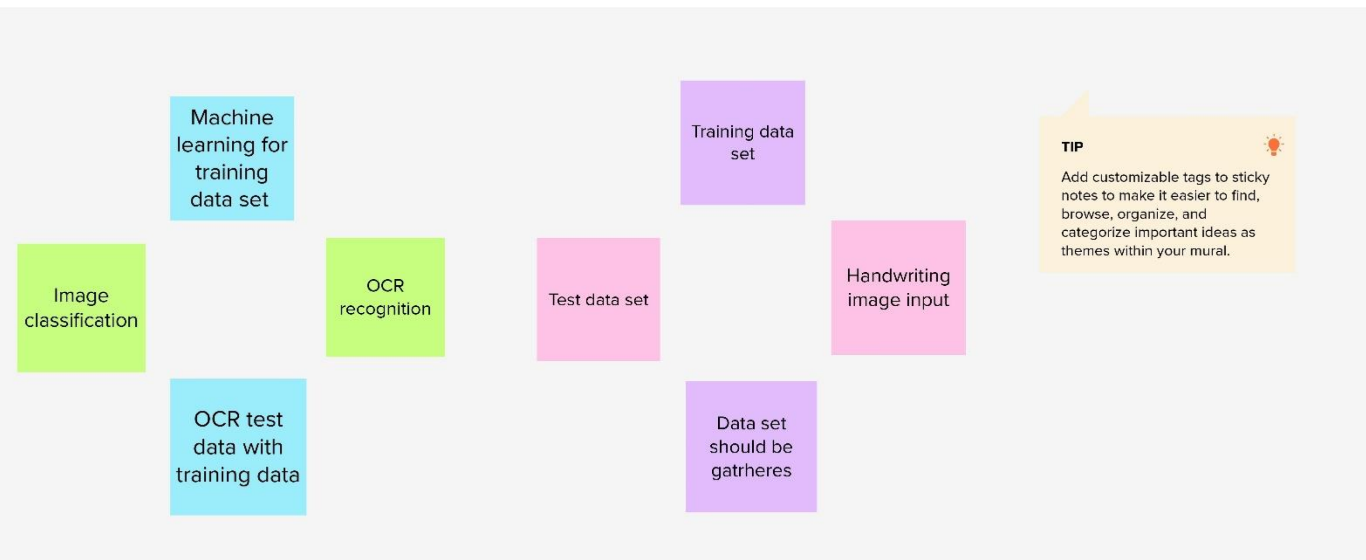
,,

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 and break it up into smaller sub-groups.

20 minutes



,,

Training data set
OCR recognition
Image classification
Machine learning for training data set
For input extract factor
TIP
Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.
Test data set
Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

,,

3.3 PROPOSED SOLUTION

S.No	Parameter	Description
1.	Problem statement (problem to be solved)	The total world is working with the various problems of the machine learning.The goal of the machine learning is to factorize and to manipulate the real life data and the real life part of the human interaction or complex ideas or the problems in the real life.The most curious of those is Handwritten Character Recognition because it is the building block of the human certified and the classification interaction between other humans. So, the goal was to create an appropriate algorithm that can give the output of the handwritten character by taking just a picture of that character. If one asks about Image processing then this problem can't be solved because there can be a lot of noises in that taken image which can't be controlled by human.
2.	Idea/solution description	Handwriting recognition, also known as handwriting OCR or cursive OCR, is a subfield of OCR technology that translates handwritten letters to corresponding digital text or commands in real-time. To perform this task, these systems benefit from pattern matching to identify various styles of handwritten letters

,,

3.	Novelty/uniqueness	The HWR domain, as defined for this paper's proposed benchmark, consistsof two high-level tasks: (1) text transcription and (2) style recognition. Thetranscription task involves an agent taking a digital image of a handwrittendocument as input and processing it to recognize the individual characters toproduce a plaintext output. The style recognition task involves the agent identifying known and unknown aspects of visual appearance for both the text (e.g.,how are individual characters stylized?) and page (i.e., what does the page looklike holistically?). Two subtasks for style recognition are considered: (2a) writeridentification and (2b) overall document appearance identification (ODAI). Theformer involves multi-class classification to distinguish between individual knownwriters and new writers unseen at training time, while the latter involves multi-class classification to distinguish between known global appearances of handwrit-ten documents and appearances unseen at training time.
4.	Social impact/customer satisfication	Today, there is an increasing demand of efficient archival and retrieval methods for online handwritten data. For such tasks, text categorization is of particular interest. The textual data available in online documents can be extracted through online handwriting recognition; however, this process produces errors in the resulting text. This work reports experiments on the categorization of online handwritten documents based on their textual contents.

„

5.	Business Model(Revenue Model)	Work related to HWR with novelty can be found in thefields of machine learning and computer vision. There is a strong foundationin deep learning-based approaches to HWR, which have yielded good perfor-mance in closed world data set evaluations. State-of-the-art approaches for di-verse document sets [27,20] are based on the Convolutional Recurrent NeuralNetwork (CRNN) [23] in combination with a Connectionist Temporal Classifi-cation (CTC) loss [9].Beyond anomaly detection [18], machine learning work on classifiers hasstarted to look at other ways in which novelty can be handled. Promising workin this direction relies on statistical modeling using extreme value theory, whichmore accurately accounts for the samples in the tails of distributions, which isconsequential for decision .
6.	Scalability of the solution	This paper introduced an agent-centric approach to handling novelty in theHWR domain. This domain is attractive for the study of novelty, as it consistsof a key challenge problem within AI: reading in a more human-like way. TheHWR domain with novelty was formalized, an evaluation protocol with bench-mark data was introduced, and comprehensive results from a baseline agent 14 D. Prijatelj et al.were presented to provide the research community with a starting point to buildupon. Beyond incremental improvements in transcription performance and stylerecognition in the presence of novelty, we suggest that adaptation via incremen-tal learning is the next step. Agents that can properly react to and managenovelty, as opposed to merely detecting novelty, will perform better on the taskover time. With additions to the evaluation protocol supporting this, we expect anew class of

,,

		agents to appear for a number of document processing application

3.4 PROBLEM SOLUTION FIT

,,

2. JOBS-TO-BE-DONE / PROBLEMS

J&P

Which jobs-to-be-done (or problems) do you address for your customers?

* Misidentification of medicine names.

* Correct labelling of Diseases.

9. PROBLEM ROOT CAUSE

RC

What is the real reason that this problem exists?
What is the back story behind the need to do this job?

The reason is using the same algorithm.Use another algorithm for digit recognition and compare both output with use of medical dictionary

7. BEHAVIOUR

BE

What does your customer do to address the problem and get the job done?

They should check the content for many times and use another algorithm for digit recognition and then compare the output.

Identify strong TR & EM

3. TRIGGERS

TR

What triggers customers to act?

It gives the clear content and it takes less time to transcript ,so the customer likes to use this handwritten digit recognition,this will trigger the customer.

4. EMOTIONS: BEFORE / AFTER

EM

How do customers feel when they face a problem or a job and afterwards?

The customer feel insecure because it is the confidential thing so they have to feel ,they are protected..After solve the problem also they have insecurity feeling.

10. YOUR SOLUTION

SL

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.

The intention is to make it work on real life data apart from the test data set . wecan ask the user to draw the digits with gestures and then de

8.CHANNELS of BEHAVIOUR

CH

8.1 ONLINE

What kind of actions do customers take online?

will recieve the image through online and recognize it .

8.2 OFFLINE

What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

the hard copy will be recieved and recognized.

Identify strong TR & EM

,,

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

FR	Sub Requirement (Story / Sub-Task)

,,

No.	
FR-1	Image Data: Handwritten digit recognition refers to a computer's capacity to identify human handwritten digits from a variety of sources, such as photographs, documents, touch screens, etc., and categorize them into ten established classifications (0-9). In the realm of deep learning, this has been the subject of countless studies.
FR-2	Website: Web hosting makes the code, graphics, and other items that make up a website accessible online. A server hosts every website you've ever visited. The hosting determines how much space is allotted to a website on a server. The four primary varieties are shared dedicated, VPS, and reseller hosting.
FR-3	Digit Classifier Model: To train a convolutional network to predict the digit from an image, use the MNIST database of handwritten digits. Get the training and validation data first.
FR-4	Cloud: The cloud offers a range of IT services, including virtual storage, networking, servers, databases, and applications. In plain English, cloud computing is described as a virtual platform that enables unlimited storage and access to your data over the internet.

,,

FR-5	Modified National Institute of Standards and Technology dataset: The abbreviation MNIST stands for the MNIST dataset. It is a collection of 60,000 tiny square grayscale photographs, each measuring 28 by 28, comprising handwritten single digits between 0 and 9.

4.2 NON FUNCTIONAL REQUIREMENTS

,,

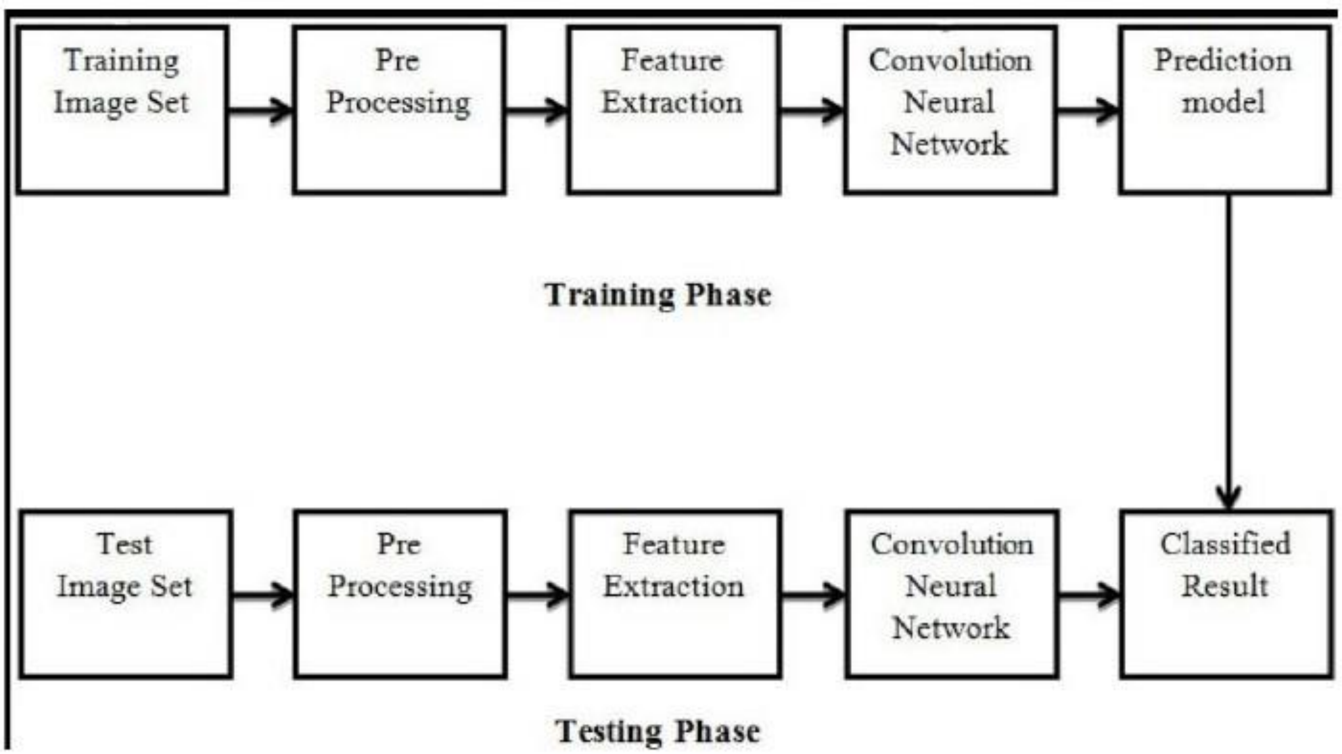
NFR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	One of the important significant problems in pattern recognition applications is the recognition of handwritten characters. Applications for digit recognition include filling out forms, processing bank checks, and sorting mail.
NFR-2	**Security**	1) The system generates through the description of the instantiation parameters, which might
NFR-3	**Reliability**	The samples are used by the neural network to automatically deduce rules for reading handwritten digits. Furthermore, the network may learn more about handwriting and hence enhance its accuracy by increasing the quantity of training instances. Numerous techniques and algorithms, such as Deep Learning/CNN, SVM, Gaussian Naive Bayes, KNN, Decision Trees, Random Forests, etc., can be used to recognise handwritten numbers.
NFR-4	**Accuracy**	With typed text in high-quality photos, optical character recognition (OCR) technology offers accuracy rates of greater than 99%. However, variances in spacing, abnormalities in handwriting, and the variety of human writing styles result in less precise character identification.
NFR-5	**Availability**	1)Reveal information like the writing style, in addition to a categorization of the digit. 2) The generative models are capable of segmentation driven by recognition. 3) The procedure uses a relatively.

CHAPTER 5

PROJECT DESIGN

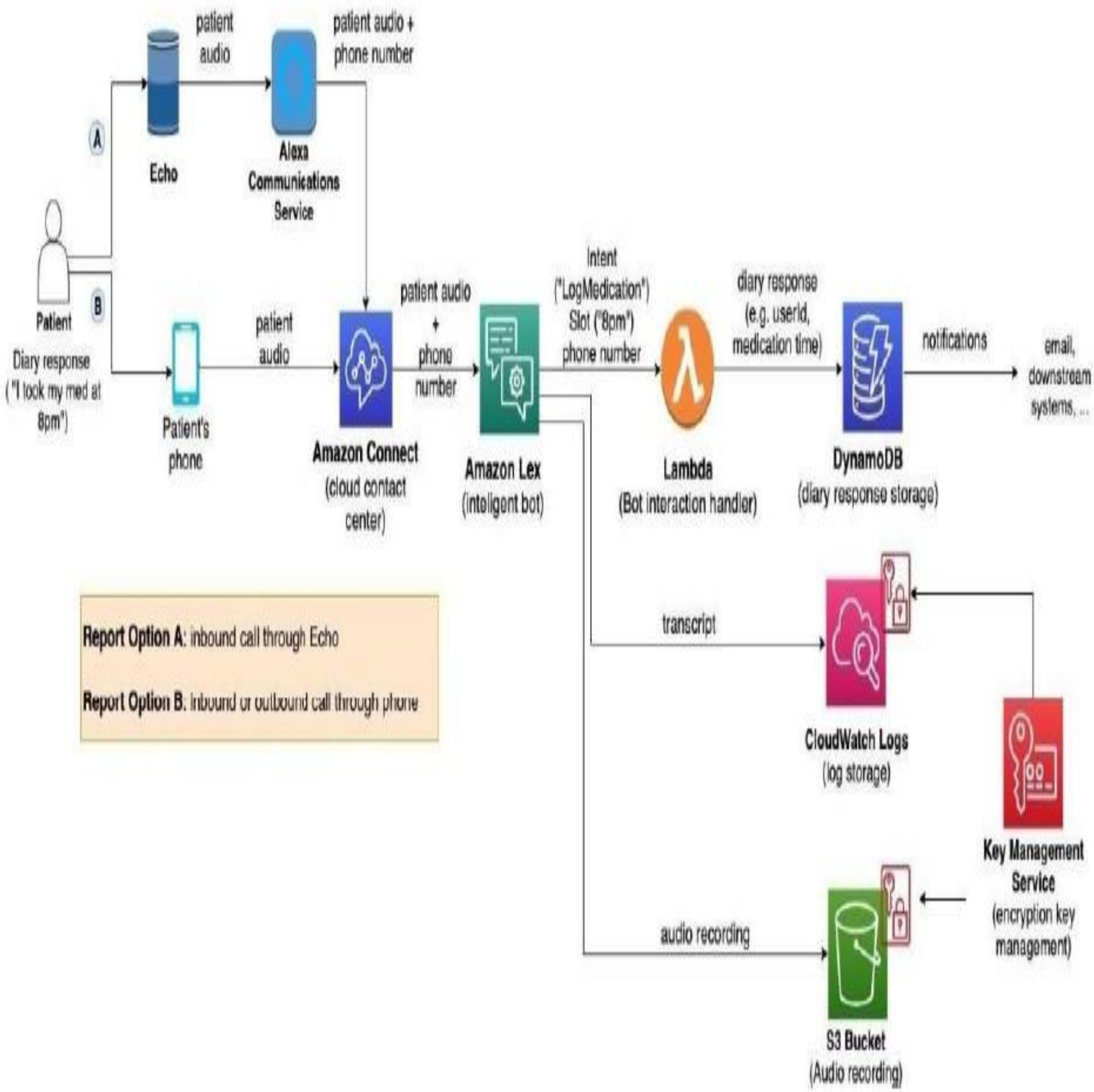
,,

5.1 DATA FLOW DIAGRAM



5.2 SOLUTION AND TECHNICAL ARCHITECTURE

,,



5.3 USER STORIES

,,

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Transcription analyst	Pre Processing	USN-1	Noise in the digital handwritten image can be reduced.	It uses noise filters.	High	Sprint-1
		USN-2	Blurred image can be modified.	Sobel filter can be used to sharpen the image.	High	Sprint-3
	Feature Extraction	USN-3	How the features can be identified.	By extracting the foreground image from background image.	Low	Sprint-2
		USN-4	How shape edges can be detected.	Curves of the letters can be found.	Medium	Sprint-1
		USN-5	How words are recognized based on sizes.	By identifying the size of the word.	High	Sprint-3
	Prediction	USN-6	How letters are predicted.	By comparing the features of each letter with the features of actual letters.	High	Sprint-4
		USN-7	How capital and small letters identified.	By separating bigger font images with smaller font images.	Low	Sprint-2
	Classified result	USN-8	How the prediction seperates the identification of the digital letter.	It seperates as correctly predicted image and wrongly predicted image.	Medium	Sprint-4
		USN-9	How the words are predicted.	Once the letters are correctly predicted it uses a dictionary to identify the words.	High	Sprint-1

,,

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	Team Members
Sprint-1	Pre Processing	USN-1	Noise in the digital handwritten image can be reduced.	High	JENIFER R
		USN-2	Blurred image can be modified.	Low	
Sprint-2	Feature Extraction	USN-3	How the features can be identified.	High	JELCY EVANGELIN A
		USN-4	How shape edges can be detected.	Low	
		USN-5	How words are recognized based on sizes.	High	
Sprint-3	Prediction	USN-6	How letters are predicted.	High	JEYASRI POOJA S
		USN-7	How capital and small letters identified.	Low	
Sprint-4	Classified result	USN-8	How the prediction seperates the identification of the digital letter.	Medium	AARTHI S A
		USN-9	How the words are predicted.	High	

,,

6.2 SPRINT DELIVERY SCHEDULE

print	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
print-1	1	3 Days	24 Oct 2022	26 Oct 2022	1	26 Oct 2022
print-2	1	3 Days	31 Oct 2022	02 Nov 2022	1	02 Nov 2022
print-3	1	3 Days	07 Nov 2022	09 Nov 2022	1	09 Nov 2022
print-4	1	3 Days	14 Nov 2022	16 Nov 2022	1	16 Nov 2022

CHAPTER 7

CODING AND SOLUTIONING

Import Required Librauries

In [52]:
```
import numpy as np
import tensorflow as tf
import matplotlib.pyplot as plt
from keras.utils import np_utils
import tensorflow as tf
from tensorflow.keras.layers import Conv2D, Dense, Flatten
```

In [2]:
```
print(tf.__version__)
```

```
2.9.2
```

In [3]:
```
mnist_ds = tf.keras.datasets.mnist.load_data()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz
11490434/11490434 [==============================] - 0s 0us/step
```

In [4]:
```
mnist_ds
```


,,

Building The Model

In [53]:
```
model = tf.keras.models.Sequential([tf.keras.layers.Conv2D(64, (3, 3), input_shape=(28, 28, 1), activation="relu"),
                                    tf.keras.layers.Conv2D(32, (3, 3), activation="relu"),
                                    tf.keras.layers.Flatten(),
                                    tf.keras.layers.Dense(10, activation=tf.nn.softmax)])
```

In [54]:
```
model.compile(loss='categorical_crossentropy', optimizer="Adam", metrics=["accuracy"])
```

In [55]:
```
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 26, 26, 64)        640

 conv2d_1 (Conv2D)           (None, 24, 24, 32)        18464

 flatten (Flatten)           (None, 18432)             0

 dense (Dense)               (None, 10)                184330

=================================================================
Total params: 203,434
Trainable params: 203,434
Non-trainable params: 0
_________________________________________________________________
```

Training The Model

In [56]:
```
model.fit(training_images, training_labels, batch_size=32, epochs=5, validation_data=(test_images,test_labels))
```

```
Epoch 1/5
1875/1875 [==============================] - 16s 4ms/step - loss: 0.1254 - accuracy: 0.9625 - val_loss: 0.0519 - val_accuracy: 0.9831
Epoch 2/5
1875/1875 [==============================] - 7s 4ms/step - loss: 0.0464 - accuracy: 0.9861 - val_loss: 0.0385 - val_accuracy: 0.9868
Epoch 3/5
1875/1875 [==============================] - 7s 4ms/step - loss: 0.0296 - accuracy: 0.9907 - val_loss: 0.0409 - val_accuracy: 0.9872
Epoch 4/5
1875/1875 [==============================] - 7s 4ms/step - loss: 0.0201 - accuracy: 0.9937 - val_loss: 0.0403 - val_accuracy: 0.9878
Epoch 5/5
1875/1875 [==============================] - 7s 4ms/step - loss: 0.0139 - accuracy: 0.9957 - val_loss: 0.0457 - val_accuracy: 0.9872
```

Out[56]:

,,

Test The Model

```
In [58]: metrics = model.evaluate(test_images, test_labels, verbose=0)

         print("Test Loss -> {}  \nTest Accuracy -> {}".format(metrics[0],metrics[1]))
```

```
Test Loss -> 0.04573516175150871
Test Accuracy -> 0.9872000217437744
```

```
In [67]: model.predict(test_images[2:8])
```

```
1/1 [==============================] - 0s 15ms/step
```

```
Out[67]: array([[2.32068427e-08, 9.99983430e-01, 8.10439190e-07, 1.28179977e-07,
                 9.64922492e-06, 1.83879649e-06, 1.62838049e-07, 1.56461965e-06,
                 2.34936374e-06, 3.22469944e-08],
                [9.99998927e-01, 1.04071238e-13, 7.69856399e-07, 1.84226245e-09,
                 3.37900985e-13, 4.71777106e-09, 8.84182239e-09, 2.02508791e-11,
                 3.22932721e-07, 9.56373647e-09],
                [8.73478698e-13, 4.26847549e-13, 1.15858136e-10, 3.97662771e-11,
                 9.99999881e-01, 2.68545906e-12, 8.56604648e-11, 7.41609482e-11,
                 4.87753553e-08, 1.05269102e-07],
                [2.15423035e-09, 9.99581635e-01, 2.15949945e-06, 1.08863398e-08,
                 2.10376020e-05, 2.63231090e-08, 3.26978977e-08, 3.81208694e-04,
                 1.27898356e-05, 1.17525337e-06],
                [2.41032138e-18, 9.36788008e-11, 3.97475330e-10, 3.54850779e-13,
                 9.99299288e-01, 6.94019908e-09, 6.61158953e-14, 4.28452246e-10,
                 7.00477336e-04, 2.29253416e-07],
                [1.89875802e-16, 2.21634187e-11, 1.76986703e-09, 2.65193867e-09,
                 2.56875592e-06, 2.88839956e-08, 1.60236594e-14, 2.67538752e-11,
                 5.46009005e-06, 9.99991894e-01]], dtype=float32)
```

```
In [74]: history=model.predict(np.array([test_images[7]]))
         history
```

```
1/1 [==============================] - 0s 17ms/step
```

```
Out[74]: array([[1.8987580e-16, 2.2163419e-11, 1.7698670e-09, 2.6519387e-09,
                 2.5687532e-06, 2.8883996e-08, 1.6023692e-14, 2.6753875e-11,
                 5.4600901e-06, 9.9999189e-01]], dtype=float32)
```

```
In [75]: np.argmax(history, axis=1)
```

```
Out[75]: array([9])
```

```
In [73]: #It predicted as 9
```

Let us see , It is correct or not?

```
In [78]: tl=test_labels[7]
         tl
```

```
Out[78]: array([0., 0., 0., 0., 0., 0., 0., 0., 0., 1.], dtype=float32)
```

```
In [81]: np.argmax(tl)
```

```
Out[81]: 9
```

It Predicted Correctly!!!

,,

CHAPTER 8

TESTING

8.1 TEST CASES

D	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Statu s	Comments	TC for Automation(Y/N)	BUG ID	Executed By
e_TC_OO	Functional	Home Page	Training data set.		1. Select the data set. 2. Training data set using CNN. 3. Model building.	127.0.0.1:5000	Home Page should be displayed.	Working as expected	Pass		N		JENIFER R
e_TC_OO	UI	Home Page	Validation testing is done.		1.Using the build model validation test set is done. 2.If validation score is high do below process.	1.png	Application should show below UI elements: A.choose file button b.predict button c.clear button	Working as expected	Pass		N		JELY EVANGELIN A
e_TC_OO	Functional	Home Page	If the validation score is above 50% the testing is processed.		1.Use the test data set in the CNN to predict the samples. 2.Performance measurement using confusion matrix. 3.Accuracy of classification.	1.png	Choose file popup screen must be displayed and user should be able to click on predict button	Working as expected	Pass		N		AARTHI S A
_TC_OO	Functional	Predict page	Performance testing is done.		1. Performance measurement using confusion matrix. 2. Training data set using CNN.	1.png	User must be navigated to the predict page and must view the predicted result	Working as expected	Pass		N		JEYASRI POOJA S
C_OO5	Functional	Home Page	The output predicted is displayed.		1.Use the test data set in the CNN to predict the samples. 2.Accuracy of classification.	1.png	The result is displayed.	Working as expected	Pass		N		JELCY EVANGELIN A

,,

8.2 USER ACCEPTANCE TESTING

8.2.1DEFECT ANALYSIS

resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	5	2	3	20
Duplicate	1	0	3	0	4
External	2	5	0	1	8
Fixed	10	3	2	18	33
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	23	18	11	24	76

,,

8.2.2 TESTCASE ANALYSIS

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	6	0	0	6
Client Application	62	0	0	58
Security	4	0	0	3
Outsource Shipping	2	0	0	2
Exception Reporting	7	0	0	7
Final Report Output	3	0	0	3
Version Control	6	0	0	5

,,

CHAPTER 9

RESULTS

9.1 PERFORMANCE METRICS

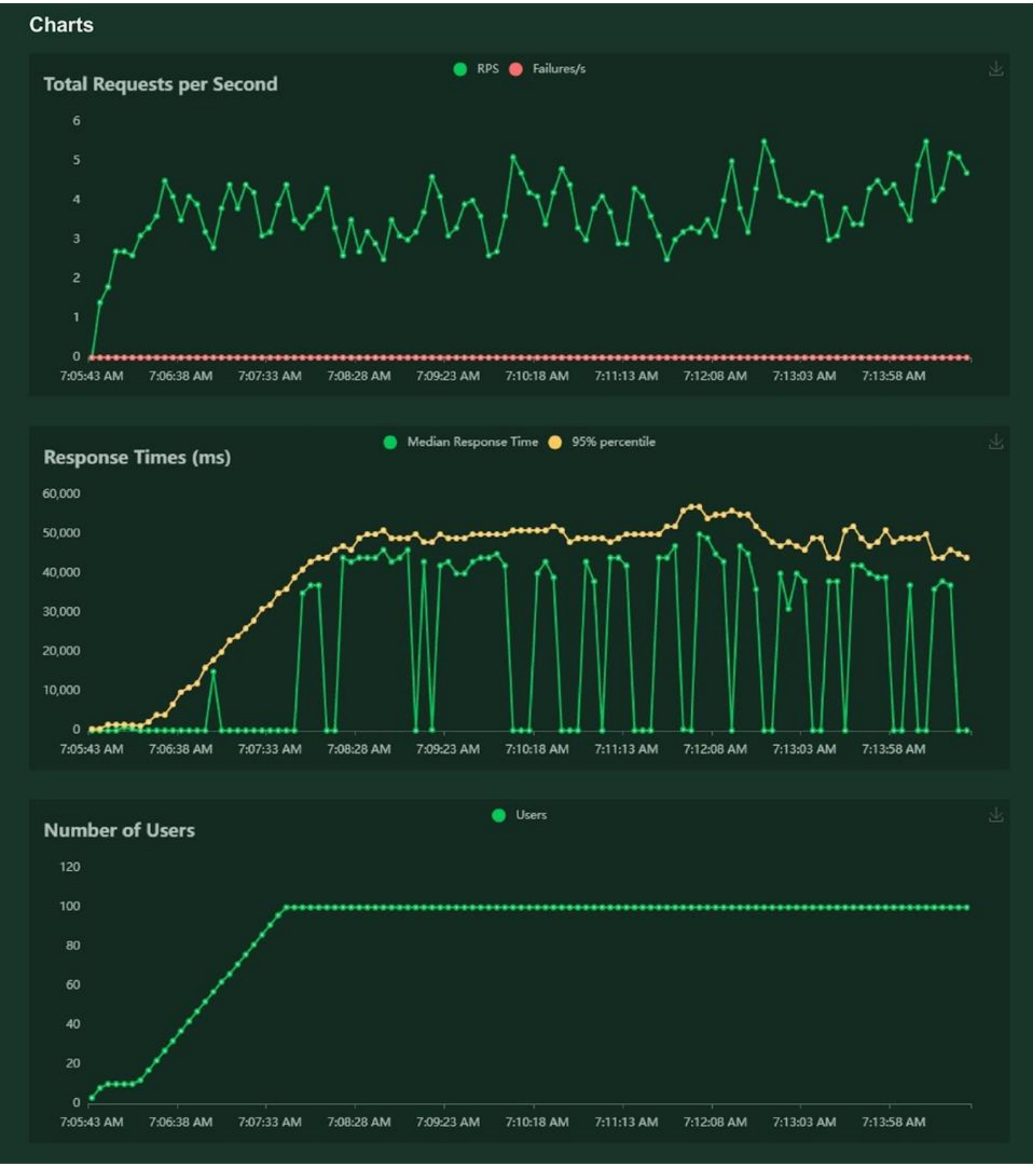
,,

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)
```

```
Metrics (Test loss &Test Accuracy) :
[0.08848220854997635, 0.9772999882698059]
```


,,


Charts
Total Requests per Second
RPS
Failures/s
6
5
4
3
2
1
0
7:05:43 AM
7:06:38 AM
7:07:33 AM
7:08:28 AM
7:09:23 AM
7:10:18 AM
7:11:13 AM
7:12:08 AM
7:13:03 AM
7:13:58 AM
Response Times (ms)
Median Response Time
95% percentile
60,000
50,000
40,000
30,000
20,000
10,000
0
Number of Users
Users
120
100
80
60
40
20
0

,,

CHAPTER 10

ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- Reduces manual work
- More accurate than average human
- Capable of handling a lot of data
- Can be used anywhere from any device

DISADVANTAGES

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

,,

CHAPTER 11
CONCLUSION

This project demonstrated a web application that uses machine learning to recognize 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:

- Add support to detect from digits multiple images and save the results
- Add support to detect multiple digits
- Improve model to detect digits from complex images
- 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

Building The Model

```
In [53]: model = tf.keras.models.Sequential([tf.keras.layers.Conv2D(64, (3, 3), input_shape=(28, 28, 1), activation="relu"),
                                             tf.keras.layers.Conv2D(32, (3, 3), activation="relu"),
                                             tf.keras.layers.Flatten(),
                                             tf.keras.layers.Dense(10, activation=tf.nn.softmax)])
```

```
In [54]: model.compile(loss='categorical_crossentropy', optimizer="Adam", metrics=["accuracy"])
```

```
In [55]: model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 26, 26, 64)        640

 conv2d_1 (Conv2D)           (None, 24, 24, 32)        18464

 flatten (Flatten)           (None, 18432)             0

 dense (Dense)               (None, 10)                184330

=================================================================
Total params: 203,434
Trainable params: 203,434
Non-trainable params: 0
_________________________________________________________________
```

Training The Model

```
In [56]: model.fit(training_images, training_labels, batch_size=32, epochs=5, validation_data=(test_images,test_labels))
```

```
Epoch 1/5
1875/1875 [==============================] - 16s 4ms/step - loss: 0.1254 - accuracy: 0.9625 - val_loss: 0.0519 - val_accuracy: 0.9831
Epoch 2/5
1875/1875 [==============================] - 7s 4ms/step - loss: 0.0464 - accuracy: 0.9861 - val_loss: 0.0385 - val_accuracy: 0.9868
Epoch 3/5
1875/1875 [==============================] - 7s 4ms/step - loss: 0.0296 - accuracy: 0.9907 - val_loss: 0.0409 - val_accuracy: 0.9872
Epoch 4/5
1875/1875 [==============================] - 7s 4ms/step - loss: 0.0201 - accuracy: 0.9937 - val_loss: 0.0403 - val_accuracy: 0.9878
Epoch 5/5
1875/1875 [==============================] - 7s 4ms/step - loss: 0.0139 - accuracy: 0.9957 - val_loss: 0.0457 - val_accuracy: 0.9872
Out[56]:
```

Test The Model

```
In [58]: metrics = model.evaluate(test_images, test_labels, verbose=0)

         print("Test Loss -> {}  \nTest Accuracy -> {}".format(metrics[0],metrics[1]))
```

```
Test Loss -> 0.04573516175150871
Test Accuracy -> 0.9872000217437744
```

FLASK APP

```python
from flask import Flask,render_template,request
from recognizer import recognize

app=Flask(__name__)

@app.route('/')
def main():
    return render_template("home.html")


@app.route('/predict',methods=['POST'])
def predict():
    if request.method=='POST':
      image = request.files.get('photo', '')
      best, others, img_name = recognize(image)
      return render_template("predict.html", best=best, others=others, img_name=img_name)


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

RECGONIZER

```python
import os
import random
import string
from pathlib import Path
import numpy as np
from tensorflow.keras.models import load_model
from PIL import Image, ImageOps

```

HOME PAGE(HTML)

,,

```html
<html>
    <head>
        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
        <title>Handwritten Digit Recognition</title>
        <link rel="icon" type="image/svg" sizes="32x32" href="{{url_for('static',filename='images/icon.svg')}}" />
        <link rel="stylesheet" href="{{url_for('static',filename='css/main.css')}}" />
        <script src="https://unpkg.com/feather-icons"></script>
        <script defer src="{{url_for('static',filename='js/script.js')}}"></script>
    </head>
    <body>
        <div class="container">
            <h1 class="heading__main">Handwritten Digit Recognizer</h1>
                <h2 class="heading__sub">Web Application to detect Handwritten digits</h2>

                <div class="form-wrapper">
                    <marquee width="100%" direction="left" height="60px">
                        <b><-----Provide an image for which you want to get the clear identity-------></b>
                        </marquee><br>
                    <form class="upload" action="/predict" method="post" enctype="multipart/form-data">
                        <label id="label" for="upload-image"><i data-feather="file-plus"></i>Select File</label>
                        <input type="file" name="photo" id="upload-image" hidden />
                        <button type="submit" id="up_btn"></button>
                    </form>
                    <img id="loading" src="{{url_for('static',filename='images/loading.gif')}}">

            </div>
        </div>
    </body>
</html>
```

,,

HOME PAGE CSS

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

* {
        padding: 0;
        margin: 0;
}

body {
        color: black;
        font-family: "Overpass", sans-serif;
        background-image: url('https://www.zastavki.com/pictures/1600x900/2015/Backgrounds_Orange_gradient_background_096901_25.jpg');

}

.container {
        width: 100%;
        height: 100%;
        display: flex;
        flex-direction: column;
        justify-content: center;
        align-items: center;
        background-color: rgba(102, 100, 100, 0.79);
}

.heading {
        margin-top: -2rem;
        padding-bottom: 2rem;
        width: fit-content;
        text-align: center;
}

.heading .heading__main {
        font-size: 3rem;
        font-weight: 550;
}

.heading .heading__sub {
        font-size: 1rem;
        color: rgb(90, 88, 88);
}

.upload-container {
        box-shadow: 0 0 20px rgb(172, 170, 170);
        width: 40rem;
        height: 25rem;
        padding: 1.5rem;
}

.form-wrapper {
        background-color: rgba(239, 12, 12, 0.5);
        width: 100%;
        height: 100%;
        display: flex;
        border: 1px dashed black;
        justify-content: center;
        align-items: center;
}

.form-wrapper #loading {
        display: none;
        position: absolute;
        width: 700px;
        height: 500px;
        border: 1px solid rgb(255, 98, 0);
}

.form-wrapper .upload {
        display: flex;
        justify-content: center;
        align-items: center;
        width: 8rem;
        height: -webkit-fit-content;
        height: -moz-fit-content;
        height: fit-content;
        border-radius: 6px;
        color: rgb(255, 255, 255);
        background-color: rgb(34, 14, 213);
        box-shadow: 0 5px 10px rgb(249, 127, 51);
}
```

,,

```
.form-wrapper .upload #up_btn {
        display: none;

}

.form-wrapper .upload label {
        font-size: 1rem;
        font-weight: 600;
        color: rgb(255, 255, 255);
        height: 100%;
        width: 100%;
        padding: 10px;
        display: block;
        background-color: blueviolet;
        text-align: left;

}

.form-wrapper .upload svg {
        height: 15px;
        width: auto;
        padding-right: 8px;
        margin-bottom: -2px;
}

@media screen and (max-width: 700px) {
        .upload-container {
                height: 20rem;
                width: 18rem;
                margin-top: 3.5rem;
                margin-bottom: -8rem;
        }

        .heading .heading__main {
                margin-top: -6rem;
                font-size: 2rem;
                padding-bottom: 1rem;
        }
}
```

,,

PREDICT PAGE (HTML)

```
<html>
        <head>
                <style>
                        .bt{
        background-color: gray;
        color: rgb(14, 13, 13);
        border: 1px solid #eee;
        border-radius: 20px;
        box-shadow: 5px 5px 5px #eee;
        text-shadow: none;
        text-align: center;
}
.button {
  border-radius: 4px;
  background-color: #043217;
  border: none;
  color: #FFFFFF;
  text-align: center;
  font-size: 22px;
  padding: 10px;
  width: 200px;
  transition: all 0.5s;
  cursor: pointer;
  margin: 5px;
}

```

```
.button span {
  cursor: pointer;
  display: inline-block;
  position: relative;
  transition: 0.5s;
}

.button span:after {
  content: ' \00AB';
  position: absolute;
  opacity: 0;
  top: 0;
  left: -20px;
  transition: 0.5s;
}

.button:hover span {
  padding-left: 25px;
}

.button:hover span:after {
  opacity: 1;
  left: 0;
}
                </style>
                <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')}}" />

        <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5ID
                <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">
                                        <img src="{{url_for('static',filename='data/')}}{{img_name}}" /><br>


                                </div>
                                <img src="https://media3.giphy.com/media/XwcRfl09HD0Sk6RaRM/giphy.gif" height="250">
                                <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>

                <form action="/" >
                        <div class="bt">
                                <button type="submit" class="button"><span>Choose another pic !  </span></button>
                </div>
                  </form>


         </body>
</html>
```

PREDICT PAGE(CSS)

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

body {
        color: rgb(255, 255, 255);
        font-family: "Overpass", sans-serif;
        background-image: url('https://wallpaperaccess.com/full/1092567.png');
}

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(124, 189, 245);
        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(129, 175, 231);
}

.result-wrapper .input-image-container img {
        width: 60%;
        height: 60%;
        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;
        border: 2px dotted black;
}

.other_predictions .value {
        display: flex;
        justify-content: center;
        align-items: center;
        flex-direction: column;
        width: 5rem;
        height: 5rem;
        box-shadow: 0 0 7px rgb(158, 157, 157);
        border: 2px dotted black;
}

.other_predictions .value div {
        margin-top: -1.2rem;
        border: 2px dotted black;

}

@media screen and (max-width: 700px) {
        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;
        }
}
```

,,

JAVA SCRIPT

```
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';
});
```

GITHUB

https://github.com/IBM-EPBL/IBM-Project-26498-1660028516

PROJECT DEMO

https://youtu.be/UNgu_s1GrBc