

A NOVEL METHOD FOR HANDWRITTEN DIGIT RECOGNITION SYSTEM

TEAM ID: PNT2022TMID38341

TEAM LEADER : KARTHICK B

TEAM MEMBER : NITHIYANANDAM K

TEAM MEMBER : PRAVEEN KUMAR S

TEAM MEMBER : AMARNATH D

TAGORE ENGINEERING COLLEGE

TABLE OF CONTENT

CHAPTER NUMBER	TITLE	PAGE NUMBER
1	INTRODUCTION	5
	1.1 Project Overview	5
	1.2 Purpose	7
2	LITERATURE SURVEY	7
	2.1 Existing problem	8
	2.2 References	8
	2.3 Problem Statement Definition	10
3	IDEATION & PROPOSED SOLUTION	12
	3.1 Empathy Map Canvas	12
	3.2 Ideation & Brainstorming	13
	3.3 Proposed Solution	15
	3.4 Problem Solution fit	17
4	REQUIREMENT ANALYSIS	19
	4.1 Functional requirement	19

	4.2 Non-Functional requirements	20
5	PROJECT DESIGN	21
	5.1 Data Flow Diagrams	21
	5.2 Solution & Technical Architecture	21
	5.3 User Stories	24
6	PROJECT PLANNING & SCHEDULING	25
	6.1 Sprint Planning & Estimation	25
	6.2 Sprint Delivery Schedule	28
	6.3 Reports from JIRA	31
7	CODING & SOLUTIONING	32
	7.1 Feature 1	32
	7.2 Feature 2	35
8	TESTING	36
	8.1 Test Cases	36
	8.2 User Acceptance Testing	37
9	RESULTS	38
	9.1 Performance Metrics	38

10	ADVANTAGES & DISADVANTAGES	39
11	CONCLUSION	39
12	FUTURE SCOPE	40
13	APPENDIX	40
	Source Code	52
	GitHub & Project Demo Link	52

1. INTRODUCTION

The hand written digit recognition is the ability of computers to recognize human handwritten digits. It is the hard task for machine to identify the handwritten text. 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 repeatedly 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 alike to this. It uses different types of neural network architectures for different types of problems.

1.1 PROJECT OVERVIEW

Character recognition is a fundamental, but most challenging in the field of pattern recognition with large number of useful applications. It has been an intense field of research since the early days of computer science due to it being a natural way of interactions between computers and humans. More precisely Character recognition is the process of detecting and recognizing characters from the input image and converts it into ASCII or other equivalent machine editable form. The technique by which a computer system can recognize characters and other symbols written by hand in natural handwriting is called handwriting recognition system. Handwriting recognition is classified into offline handwriting recognition and online handwriting recognition . If handwriting is scanned and then understood by the computer, it is called offline handwriting recognition. In this case, the handwriting is recognized while writing through touch pad using stylus pen, it's called online handwriting recognition. From the classifier perspective, character recognition systems are classified into two main categories i.e. segmentation free and segmentation based.

The segmentation free also known as the holistic approach to recognize the character without segmenting it into subunits or characters. Each word is represented as a set of global features. Handwritten character processing systems are domain and

application specific, like it is not possible to design a generic system which can process all kinds of handwritten scripts and language.

The issue of transcribed digit acknowledgment has for some time been an open issue in example order. A few examined have demonstrated that neural network has an incredible execution in information arrangement. The fundamental target of this paper is to give effective and solid procedures to acknowledgment of transcribed numerical by looking at different existing arrangement models. This paper thinks about the exhibition of convolutional neural network. Results demonstrate that classifier beat over neural network with critical improved computational effectiveness without relinquishing execution. Handwritten digit recognition can be performed using the convolutional neural network from machine learning.

The issue of manually written numerals acknowledgment has been broadly concentrated lately and the huge amount of pre-processing strategies and arrangement calculations have been created. Notwithstanding, transcribed numerals acknowledgment is as yet a test for us. The primary trouble of transcribed numerals acknowledgment is the genuine change in size, interpretation, etc., of the numeral picture as a result of written by hand digits are composed by various clients and their composing style. And is not quite the same as one client to another. A few considered have utilized various approaches to manually written digit with various AI procedures Khotanzad et al (1998) who have applied the ideas of Machine Learning and Neural Networks to perceive and decide the transcribed digits from its picture. This investigation has indicated that digit acknowledgment is an amazing model issue for finding out about neural organizations and it gives an extraordinary method to grow further developed strategies like profound learning. Transcribed acknowledgment (HWR) is the capacity of a PC to get and comprehend understandable manually written contribution from sources, for example, paper archives, client input contact screens and different gadgets.[1] The picture of the composed content might be detected from a bit

of paper by optical filtering (optical character acknowledgment) or canny word acknowledgment or by client input. Then again, the developments of the pen tip might be detected "on line", for instance by a pen-based PC screen surface, a for the most part simpler undertaking as there are more hints accessible This paper presents perceiving the manually written digits (0 to 9) from the renowned MNIST dataset utilizing Tensor Flow framework (library) and python as language and its libraries as client enters the particular digit the machine would perceive and show the outcomes with exactness rate.

1.2 PURPOSE

This project aims to meet the following objectives:

- To develop handwritten digit recognizing system that enables users to automate the process of digit recognition using this deep learning model.
- To test the accuracy of the model
- Efficient model which is less computation intensive

2. LITERATURE SURVEY

A literature review is a comprehensive summary of previous research on a topic. The literature review surveys scholarly articles, books, and other sources relevant to a particular area of research. The review should enumerate, describe, summarize, objectively evaluate and clarify this previous research. It should give a theoretical base for the research and help you (the author) determine the nature of your research. The literature review acknowledges the work of previous researchers, and in so doing, assures the reader that your work has been well conceived. It is assumed that by mentioning a previous work in study, that the author has read, evaluated, and assimilated that work into the work at hand.

A literature review creates a "landscape" for the reader, giving her or him a full understanding of the developments in the field. This landscape informs the reader that the author has indeed assimilated all (or the vast majority of) previous, significant works in the field into her or his research.

2.1 EXISTING PROBLEM

Handwritten digit recognition finds its application in various fields such as post mail sorting system where scanned images of mail envelopes are made into queue and extract the section describing postcode to be delivered. With the help of digit recognizer, sorting of mails can be done based on these postcodes according to their region. Another application that utilizes this technique is form processing, digits are extracted from certain columns of a form and users put certain filters to get the desired results they want. But there is no interface for a user to get their images scanned and recognized which makes the task complicated to use for a normal user.

2.2 REFERENCES

- [1] Kai Ding, Zhibin Liu, Lianwen Jin, Xinghua Zhu, "A Comparative study of GABOR feature and gradient feature for handwritten chinese character recognition", International Conference on Wavelet Analysis and Pattern Recognition, pp. 1182-1186, Beijing, China, 2-4 Nov. 2007.
- [2] Pranob K Charles, V.Harish, M.Swathi, CH. Deepthi, "A Review on the Various Techniques used for Optical Character Recognition", International Journal of Engineering Research and Applications, Vol. 2, Issue 1, pp. 659-662, Jan-Feb 2012.
- [3] Bhatia Neetu, "Optical Character Recognition Techniques", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 5, May 2014.
- [4] Liana M. Lorigo and Venu Govindaraju, "Offline Arabic Handwriting Recognition: A Survey", IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 28 Issue 5, May 2006.
- [5] K. Gaurav and Bhatia P. K., "Analytical Review of Preprocessing Techniques for Offline

Handwritten Character Recognition”, 2nd International Conference on Emerging Trends in Engineering & Management, ICETEM, 2013.

[6] Salvador España-Boquera, Maria J. C. B., Jorge G. M. and Francisco Z. M., “Improving Offline Handwritten Text Recognition with Hybrid HMM/ANN Models”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 33, No. 4, April 2011.

[7] U. Pal, T. Wakabayashi and F. Kimura, “Handwritten numeral recognition of six popular scripts,” Ninth International conference on Document Analysis and Recognition ICDAR 07, Vol.2, pp.749-753, 2007.

[8] Anita Pal & Dayashankar Singh, “Handwritten English Character Recognition Using Neural,” Network International Journal of Computer Science & Communication. Vol. 1, No. 2, July-December 2010, pp. 141-144.

[9] J. Pradeep, E. Srinivasan and S. Himavathi, “Diagonal Based Feature Extraction For Handwritten Alphabets Recognition System Using Neural Network”, International Journal of Computer Science & Information Technology (IJCSIT), Vol 3, No 1, Feb 2011.

[10]A. Brakensiek, J. Rottland, A. Kosmala and J. Rigoll, “Offline Handwriting Recognition using various Hybrid Modeling Techniques & Character N-Grams”, Available at <http://irs.ub.rug.nl/dbi/4357a84695495>.

[11]Reena Bajaj, Lipika Dey, and S. Chaudhury, “Devnagari numeral recognition by combining decision of multiple connections classifiers”, Sadhana, Vol.27, part. 1, pp.-59-72, 2002.

[12]Sandhya Arora, “Combining Multiple Feature Extraction Techniques for Handwritten

Devnagari Character Recognition”, IEEE Region 10 Colloquium and the Third ICIIIS, Kharagpur, INDIA, December 2008.

[13]Mohammed Z. Khedher, Gheith A. Abandah, and Ahmed M. Al- Khawaldeh, “Optimizing Feature Selection for Recognizing Handwritten Arabic Characters”, proceedings of World Academy of Science Engineering and Technology, vol. 4, February 2005 ISSN 1307-6884.

[14]Sushree Sangita Patnaik and Anup Kumar Panda, “Particle Swarm Optimization and Bacterial Foraging Optimization Techniques for Optimal Current Harmonic Mitigation by Employing Active Power Filter Applied Computational Intelligence and Soft Computing”, Volume 2012, Article ID 897127.

[15]T.Som, Sumit Saha, "Handwritten Character Recognition Using Fuzzy Membership Function", International Journal of Emerging Technologies in Sciences and Engineering, Vol.5, No.2, pp. 11-15, Dec 2011.

[16]Renata F. P. Neves, Alberto N. G. Lopes Filho, Carlos A.B.Mello, CleberZanchettin, “A SVM Based Off-Line Handwritten Digit Recognizer”, International conference on Systems, Man and Cybernetics, IEEE Xplore, pp. 510-515, 9-12 Oct, 2011, Brazil

2.3 PROBLEM STATEMENT DEFINITION

Deep learning has been widely used to recognize handwriting. In offline handwriting recognition, text is analyzed after being written. The only information that can be analyzed is the binary output of a character against a background. Although shifts towards digital stylus for writing gives more information, such as pen stroke, pressure and speed of writing, there is still a necessity for offline methods, when online is inaccessible. It is particularly necessary for historical documents, archives, or mass

digitization of hand-filled forms. Extensive research into this field has resulted in significant progress from classical methods, right up to human-competitive performance. This article serves as an overview of that journey, and the potential future of the field. Handwritten text classifiers were first required for classification of postal mail. Using scanning equipment, hardwired logic recognized mono-spaced fonts. The first Optical Character Recognition (OCR) software developed in 1974 by Ray Kurzweil. By reducing the problem domain, the process was more accurate. This allowed for recognition in handwritten forms. Foremost, it lacked efficiency and knowledge of unexpected characters. These classical techniques carried heavy limitations in two key areas:

Character extraction — Individual characters are recognized by ease with OCR. Cursive handwriting, which is connected, poses more issues with evaluation. It is difficult to interpret handwriting with no distinct separation between characters.

Feature extraction — Individual properties of symbols were hard-coded, and matched to input symbols. Properties include aspect ratio, pixel distribution, number of strokes, distance from the image center, and reflection. This requires development time, as these properties are added manually.

Following are the constraints faced when computers approach to recognize handwritten digits:

1. The Handwritten digits are not always of the same size, width, orientation and justified to margins as they differ from writing of person to person.
2. The similarity between digits such as 1 and 7, 5 and 6, 3 and 8, 2 and 7 etc. So, classifying between these numbers is also a major problem for computers.
3. The uniqueness and variety in the handwriting of different individuals also influence the formation and appearance of the digits.

THE SYSTEM CAN BE USED TO OVERCOME THE ONE THE ISSUE LISTED BELOW

HANDWRITTEN DIGIT RECOGNITION USED IN BANKING SECTORS

- A bank employee need to
- Identify the user account number which written in challan
- It is difficult for the officer to identify what is written in the challan
- Not aware of handwritten styles
- Uncomfortable to do their transaction
- This software solves the bank employee issues.

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

WHAT DO THEY THINK AND FEEL (What really counts major preoccupations worries and aspirations.)

- Curious about the milestone
- Logic should work
- Design and feel

WHAT DO THEY SEE (Environment, Friends, What market offers)

- Calculations
- Colorful display
- Digits and Letters

WHAT DO THEY HEAR (What friends say, What boss say, What influencers say)

- Run a Query
- Accuracy
- Team complaints

WHAT DO THEY SAY AND DO (Attitude in public, Apperance, Behavior towards others)

- Avoid excessive meetings
- Suggestions

PAIN (Fear, Frustration, Obstacles)	GAIN (Need, Measure of success, Obstacles)
No Human Improvement	Easy
High Cost	Accuracy
Uncertainty	User-friendly

3.2 IDEATION & BRAINSTORMING

Handwritten digit recognition, is the ability of a computer to recognize the human handwritten digits from different sources. In handwritten recognition digits, characters are given as input. The model can be recognized by the system. A simple Convolutional Neural Network (CNN) has an input layer, an output layer and some hidden layers between the input and output layer.

BRAINSTORMING

Brainstorming is a method of generating ideas and sharing knowledge to solve a particular commercial or technical problem, in which participants are encouraged to think without interruption. Brainstorming is a group activity where each participant shares their ideas as soon as they come to mind.

- **POSTAL MAIL SORTING (KARTHICK B [TL])**

- ◇ Helps to reduce the mail sorting time
- ◇ It improves the speed of reading digits
- ◇ Helps to eliminate the human errors
- ◇ It can be used to sort both Incoming and outgoing mail
- ◇ It can sort the mails using the pin code
- ◇ Requires minimum man power
- ◇ Used to recognize hand printed digits
- ◇ Helps to recognize the postal codes

- **AUTOMATIC LICENSE PLATE RECOGNITION (NITHIYANANDAM K)**

- ◇ Helps to recognize the data from images

- ◇ It allows automated alerts while making the mistakes
 - ◇ Accuracy of real time analytics
 - ◇ Enhances the parking management
 - ◇ Helps to reduce the errors which are made by man
 - ◇ Enhanced security and safety
 - ◇ Helps to stop criminal behavior
 - ◇ Easier management of resources
- **BANK CHECK PROCESSING (PRAVEEN KUMAR S)**
 - ◇ Ensure effective and reliable approaches for recognition
 - ◇ Evaluated on self generated data set of bank cheque
 - ◇ Make banking operation easier and error free
 - ◇ A cheque processing system becomes commercially efficient when error rate is low
 - ◇ Complexity and effort will be less while processing cheque
 - ◇ Provides high fault tolerance and parallel architecture
 - ◇ Largely automate the system by reducing workload, time and cost per transaction
 - ◇ Processed with minimal human intervention
- **FORM DATA ENTRY (AMARNATH D)**
 - ◇ It can be used to store the data in efficient way
 - ◇ It tries to provide a error free solutions
 - ◇ It handles form processing in a large scale for fast
 - ◇ Recognize the data from the images
 - ◇ It helps to decrease the man power
 - ◇ Keeps the saved form open for further editing
 - ◇ Identifies information being incorrectly put into the system
 - ◇ It saves the form information

3.3 PROPOSED SOLUTION

Deep Learning has emerged as a central tool for self-perception problems like understanding images, voice from humans, robots exploring the world. The project aims to implement the concept of Convolutional Network which is one of the important architecture of deep learning. Understanding CNN and applying it to the handwritten recognition system, is the major target of the proposed system. This project is divided into 3 sections:

► Image Feature Extraction

During our method, we use CNN LeNet-5 [5] to obtain more diverse features from each handwritten digit image. The LeNet architecture is considered as the first architecture for convolutional neural networks. We can easily see from the LeNet-5 that many feature maps are generated in each layer. So we can obtain more diverse features than using other 14 common methods. The LeNet-5 is an excellent architecture for handwritten digit recognition. The LeNet-5 has two parts, one is feature extraction, whereas the other one is classification which is used to classify objects. Given an image of $32 \times 32 \times 1$, firstly, a convolution layer with six 5×5 filters with the stride of 1 is used and an output matrix of $28 \times 28 \times 6$ is generated. With the stride of 1 and no padding, the feature map is reduced from 32×32 to 28×28 . Then average pooling with the filter width of 2 and the stride of 2 is taken and the dimension is reduced by the factor of 2 and ends up with $14 \times 14 \times 6$. Furthermore, another convolution layer with sixteen 5×5 filters is used leading to an output matrix of $10 \times 10 \times 16$. Then another pooling layer is involved and ends up with an output matrix of $5 \times 5 \times 16$. Therefore, we extract sixteen 5×5 feature maps from each image, and each feature map (5×5) is treated as a column vector (25×1). Overall, there are two convolution layers, two sub sampling layers, and two fully connected layers in the LeNet-5.

➤ **Image Classification**

- Once the feature extraction has been done, Pooled Feature Map is flattened to get fully connected layer. This fully connected layer has 120 feature maps each of size 1x1. Each of the 120 units is connected to all the 400 nodes (5x5x16) in the fully connected layer.
- Sixth layer is a fully connected layer with 84 units in order to reduce number of trainable parameters from 48120 (with 120 units layer 5) to 10164.
- Finally, there is a fully connected softmax output layer y^{\wedge} with possible values corresponding to the digits from 0 to 9.

➤ **GUI development for digits prediction**

- After we get the desired testing input, an interface is developed for the purpose of enabling users with a choice to detect the digits depicted or written in images or drawer respectively. When users opens up the interface, they will be provided with an option to choose whether they wants to draw the digits all by themselves or insert image files from their local directory containing digits.
- If users chooses first option, he will be guided to a drawer interface where he can draw digits by themselves and get their digits recognized along with their accuracy.
- If users chooses second option, he will be asked to insert image file from their local directory and can get their digits written in image files predicted with optimum accuracy and along with percentage.
- So by giving recognized digits as a results against inputs made by users to its users, this project fulfils all its objective.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	A Novel Method for Handwritten Digit Recognition System
2.	Idea / Solution description	The proposed solution is to classify the digits which is in handwritten format by using CNN based model and this model can be trained by using MNIST database which contains 60,000 training samples and 10,000 test samples.
3.	Novelty / Uniqueness	To classify the image datasets by using CNN, which provides efficient solution compare to other methods. Here ANN algorithm is used for voice recognition which helps blind people.
4.	Social Impact / Customer Satisfaction	Users no need to use external dependencies or devices to recognize the digits, this process can be done through our mobile phones.
5.	Business Model (Revenue Model)	Input module Image processing module Segmentation module Feature extraction module Data set training module
6.	Scalability of the Solution	The accuracy of the result for the training data set is 99.98%, and 99.40% with 50% noise by using MNIST. Even we can improve this model to achieve the better results by training different types of datasets

3.4 PROBLEM SOLUTION FIT

1. CUSTOMER SEGMENT(S)

The Bank Employee who makes the transactions through the cheque.

2. JOBS-TO-BE-DONE / PROBLEMS

Every single has their own style of writing which could not recognize by the computer

3. **TRIGGERS**

Feel free to make transactions without any fear about their style of writing

4. **EMOTIONS: BEFORE / AFTER**

If the person faces a problem regarding the transactions they could confidently handle the situation by using handwritten digit recognition system

5. **AVAILABLE SOLUTIONS**

- a. Automatic digit recognition
- b. In past, people identify the digits to their analysis sometimes it causes wrong transactions.
- c. By using this application, they could easily identify the digits

6. **CUSTOMER CONSTRAINTS**

External dependencies are quite expensive and it's not offered by the people, So this process overcome the problem through their installation in mobile.

7. **BEHAVIOUR**

To classify the digits in correct way, they could make the transactions easier without any doubtfulness

8. **CHANNELS OF BEHAVIOUR**

a. ONLINE:

Promoting this application through the mobiles, the transaction could be done at any place without the presence in bank

b. OFFLINE:

The identification of the digits which is in the handwritten form directly captured by using mobile application and that could be used to convert the those digits into machine readable forms.

9. PROBLEM ROOT CAUSE

Every single has their own style of writing which could not recognize by the computer.

10. SOLUTION

- a. CNN model could be used to provide very High accuracy in image recognition problems and also reduces the high dimensional of the images, without losing its information.
- b. It can be used to convert the handwritten digits to machine readable format

4. REQUIREMENT ANALYSIS

Requirements analysis is very critical process that enables the success of a system or software project to be assessed. Requirements are generally split into two types:

- Functional requirements
- Non-functional requirements.

4.1 FUNCTIONAL REQUIREMENTS

These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	The product essentially converts handwritten digits to digital form.	The user is first asked to draw a number on the canvas, and the model that is built is then utilised to compare the data and provide an output in

		digitalize d form.
FR-2	Recognizing the handwritten digit and displaying.	Recognizing the handwritten digit and displaying.
FR-3	Import dataset file directly to the program from a command that will download the dataset from its website. Save the dataset file in the same directory as the program	Installing packages and applications.
FR-4	Build a Neural Network with many nodes in the input layer equal to the number of pixels in the arrays	Nil
FR-5	Activating the Neural Network	Packages – tensorflow

4.2 NON-FUNCTIONAL REQUIREMENTS

These are the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.

NFR No.	Non-Functional Requirement	Description
NFR-1	Usability	System design should be easily understood and user friendly to users. Furthermore, users of all skill levels of users should be able to navigate it without problems.
NFR-2	Security	The system should automatically be able to authenticate all users with their unique username and password
NFR-3	Performance	Should reduce the delay in information when hundreds of requests are given.
NFR-4	Availability	Information is restricted to each users limited access
NFR-5	Scalability	The system should be able to handle 10000 users accessing the site simultaneously

5. PROJECT DESIGN

Project design is an early phase of the project lifecycle where ideas, processes, resources, and deliverables are planned out. A project design comes before a project plan as it has a broad overview whereas a project plan includes more detailed information.

5.1 DATAFLOW DIAGRAM

A data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various sub processes the data moves through. DFDs are built using standardized symbols and notation to describe various entities and their relationships.

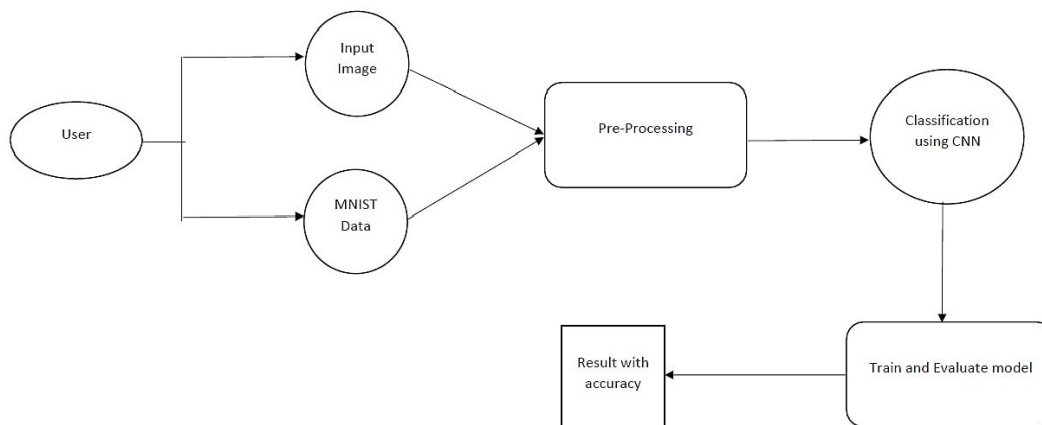


Fig. 5.1.1 Data-flow Diagram (DFD)

5.2 SOLUTION & TECHNICAL ARCHITECTURE

SOLUTION

Solution Architects are most similar to project managers, ensuring that all parties, including stakeholders, are and moving in the right direction at all stages.

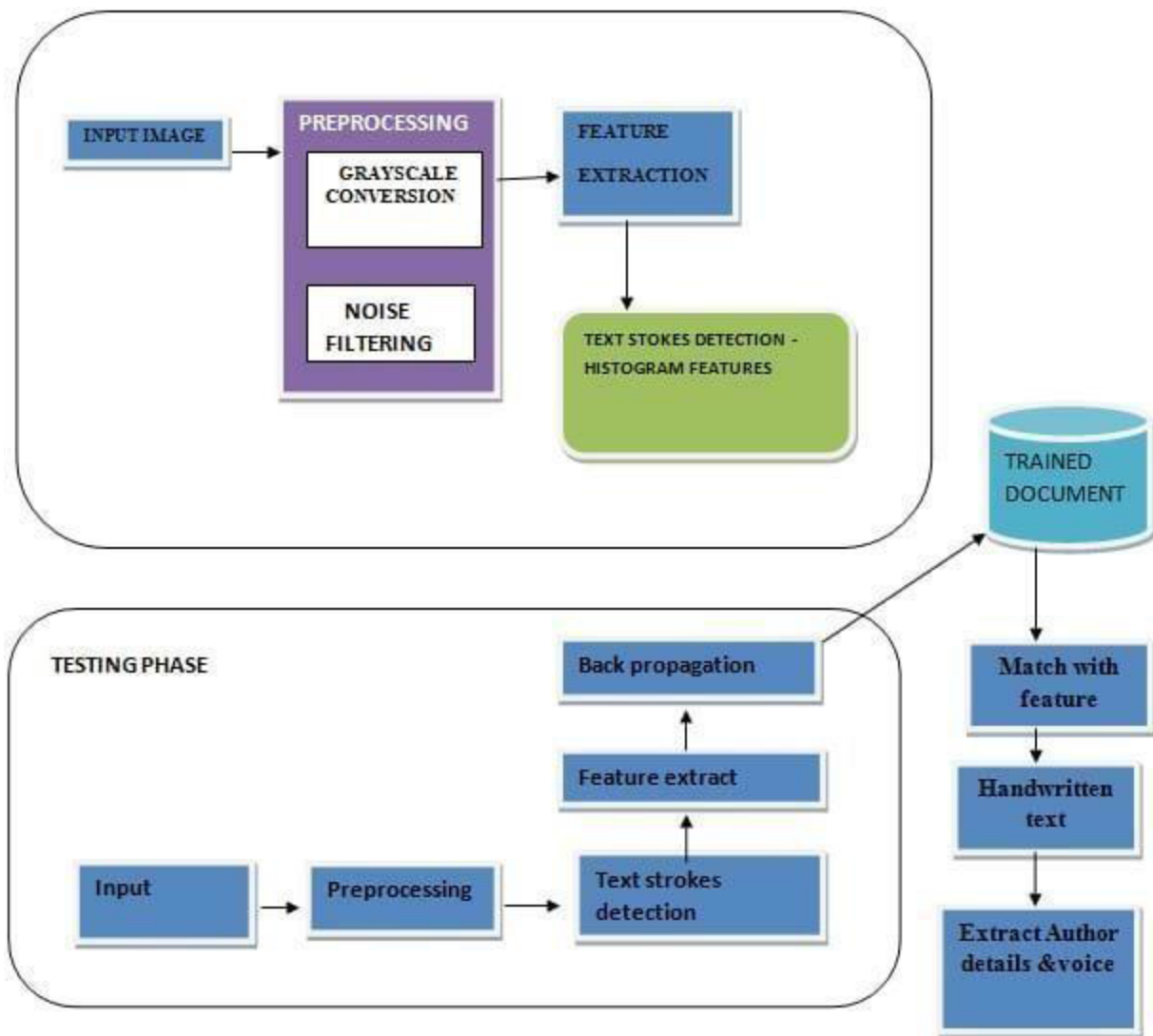


Fig. 5.2.1 Block Diagram

TECHNICAL ARCHITECTURE

Technical Architecture (TA) is a form of IT architecture that is used to design computer systems. It involves the development of a technical blueprint about the arrangement, interaction, and interdependence of all elements so that system-relevant requirements are met.

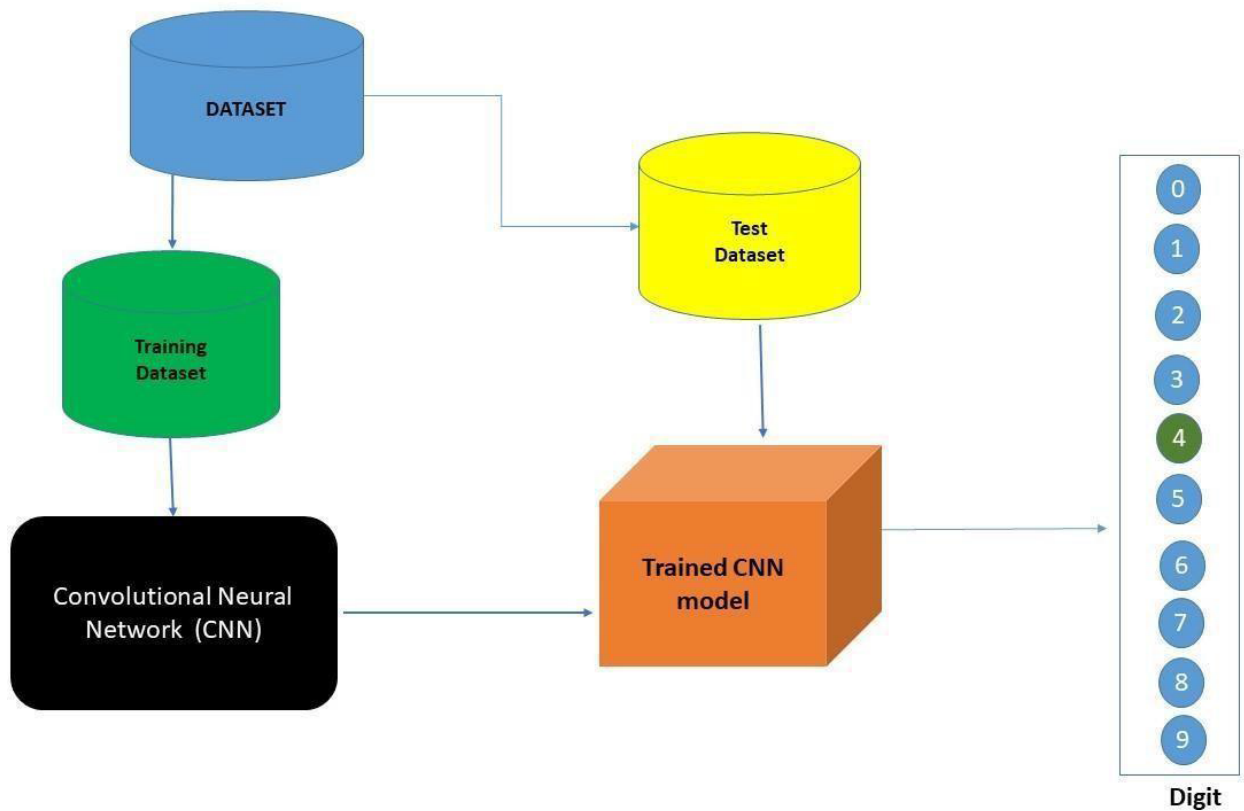


Fig. 5.2.2 Technical Architecture

COMPONENTS & TECHNOLOGIES

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g., Mobile Application	HTML, CSS, JavaScript / Angular JS / Node Red
2.	Application Logic-1	Logic for a process in the application	Java / Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on AI in cloud	IBM DB2
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage

			Service or Local File system
8.	External API-1	Purpose of External API used in the application	IBM Weather API, etc.
9.	Internet of Things Model	Purpose of AI Model is for integrating the sensors with a user interface.	IBM AI Platform
10.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model
11.	Infrastructure (Server / AI)	Application Deployment on Local System / AI Local Server Configuration AI Server Configuration	Local, Kubernetes, etc.

APPLICATION CHARACTERISTICS

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Deep learning frameworks can help you upload data and train a deep learning model that would lead to accurate and intuitive predictive analysis.	Tensor-flow, PyTorch
2.	Security Implementations	The system should automatically be able to authenticate all users with their unique username and password	NA
3.	Scalable Architecture	The system should be able to handle 10000 users accessing the site simultaneously	NA
4.	Availability	Availability	NA
5.	Performance	Should reduce the delay in information when hundreds of requests are given	Google Co-Lab Pro/ Require high end system.

5.3 USER STORIES

A user story is an informal, general explanation of a software feature written from the perspective of the end user or customer. The purpose of a user story is to articulate

how a piece of work will deliver a particular value back to the customer.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Application	USN-1	As a user, I can application by opening it easily.	I can download the application	High	Sprint-1
		USN-2	As a user, I will be given access to the canvas board to draw or write the number	I can access the canvas	High	Sprint-1
		USN-3	As a user, I can change the color of the pen ink.	I can use the canvas pen	Medium	Sprint-2

6. PROJECT PLANNING & SCHEDULING

Planning and scheduling are two important steps in project development that allow companies to achieve a project's goals and deliver quality results. Each serves an important, unique purpose in the process, despite sharing similarities. Understanding these concepts can help you better use each of them for more efficient and productive projects.

6.1 SPRINT PLANNING & ESTIMATION

Activity Number	Activity Name	Detailed Activity Description	Assigned To	Status/ Comments
1	Preparation Phase	Access the resources (courses) in project dashboard Access the	Karthick B, Nithiyanandam K, Praveen Kumar S,	It refers to done the listed activities in the

		guided project workspace Create GitHub account & collaborate with Project Repository in project workspace Set-up the Laptop / Computers based on the prerequisites for each technology track	Amarnath D	preparation phase and done Prerequisites, Registration, Environment setup
2	Ideation Phase	Literature survey on the selected project & Information Gathering, Preparation of Empathy Map Canvas to capture the user Pains & Gains, prepare list of problem statements List the ideas by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	The activities in ideation phase refers to when gathering the idea for project information and picturize in Empathy map, referring the literature survey& brain storming the ideas for this project.
3	Project Design Phase- I			
3.1	Proposed Solution	Preparation of proposed solution document, which includes the novelty, feasibility of idea, business model, socialimpact, scalability of solution	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	The solution for the project is prepared as a standard document structure from Team members
3.2	Problem Solution fit	Preparation of problem solution fit	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	Prepared analyzed and make effective solutions for the problem
3.3	Solution	Prepare an architecture	Karthick B,	Suitable block

	Architecture	for solution	Nithiyanandam K, Praveen Kumar S, Amarnath D	Diagram template used to prepare Solution architecture
4	Project Design Phase -II			
4.1	Requirement Analysis	Prepare the Functional Requirement and the Non-Functional Requirement	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	Listing of functional and Non-Functional Requirements of the project
4.2	Customer Journey	Preparation of customer journey maps to Journey understands the user interactions & experiences with the application (entry to exit)	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	Customer Journey map prepared by suitable template by team members
4.3	Data Flow Diagrams	Prepare a Data Flow Diagram for Project	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	Use suitable data flow diagram rules and standards to prepare DFD
4.4	Technology Architecture	Prepare Technology Architecture of the solution	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	We created architecture diagram and technologies used for this project
5	Project planning phase			
5.1	Milestones & Tasks	Prepare Milestone & Activity List	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	When project begins then it is expected that project related activities must be initiated. In project planning, series of

				milestones must be established.
5.2	Sprint Schedules	Prepare Sprint Delivery Plan	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	In this, Product Backlog, Sprint Schedule for the Project are estimated.
6	Project Development Phase		Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	In this, we are going to develop & submit the developed code by testing it.
6.1	Coding & Solutioning	Sprint-1 Delivery: Develop the Code, Test and push it to GitHub.	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	
6.2	Acceptance Testing	Sprint-2 Delivery: Develop the Code, Test and push it to GitHub. Sprint-3 Delivery: Develop the Code, Test and push it to GitHub.	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	
6.3	Performance Testing	Sprint-4 Delivery: Develop the Code, Test and push it to GitHub.	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D	

6.3 SPRINT DELIVERY SCHEDULE

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Home	USN-1	As a user, I	1	Medium	Karthick B,

			can view the guide and awareness to use this application.			Nithiyanandam K, Praveen Kumar S, Amarnath D
Sprint-1		USN-2	As a user, I'm allowed to view the guided video to use the interface of this application.	3	High	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D
Sprint-1		USN-3	As a user, I can read the instructions to use this application.	2	Low	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D
Sprint-2	Recognize	USN-4	As a user, In this recognition page I get to choose the image.	4	High	Karthick B, Nithiyanandam K, Praveen Kumar S, Amarnath D

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-3	Predict	USN-5	As a user, I'm Allowed to upload	3	Low	Karthick B, Nithiyanandam K,

			and choose the image to be uploaded			Praveen Kumar S, Amarnath D
Sprint-3		USN-6	As a user, I will train and test the input to get the maximum accuracy of output.	4	High	Karthick B, Nithiyanan dam K, Praveen Kumar S, Amarnath D
Sprint-3		USN-7	As a user, I can access the MNIST data set	2	Medium	Karthick B, Nithiyanan dam K, Praveen Kumar S, Amarnath D

Project Tracker, Velocity & Burndown Chart:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	31Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-2	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-3	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022
Sprint-4	20	6 Days	21Nov 2022	26 Nov 2022	20	26 Nov 2022

6.3 REPORTS FROM JIRA

BURNDOWN CHART

A burndown chart shows the amount of work that has been completed in an epic or sprint, and the total work remaining. Burndown charts are used to predict your team's likelihood of completing their work in the time available. They're also great for keeping the team aware of any scope creep that occurs.

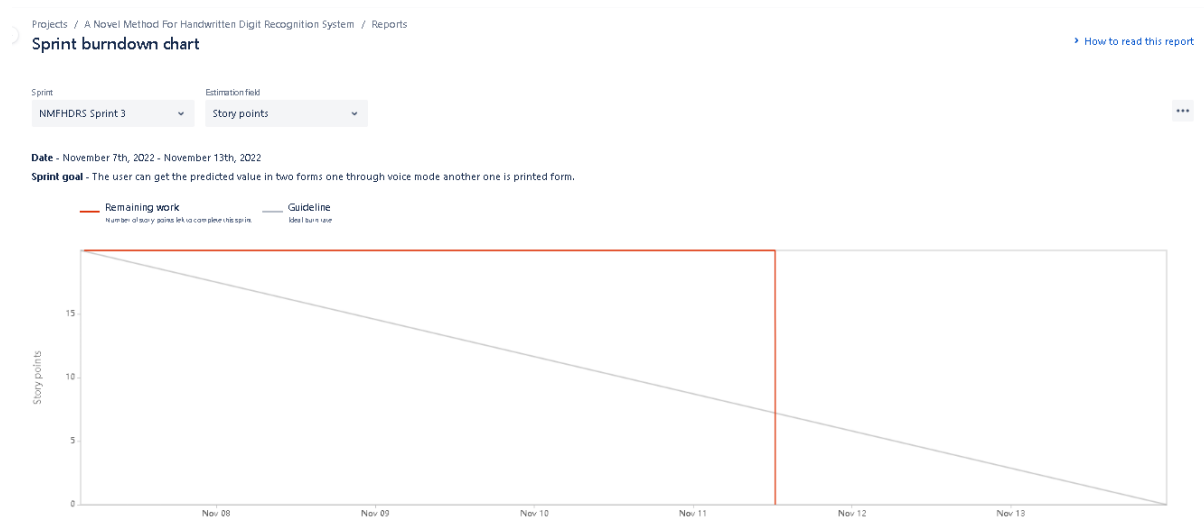


Fig. 6.3.1 Burndown Chart

VELOCITY REPORT

The Velocity report shows how much work was delivered in each sprint. This helps you predict the amount of work your team can do in future sprints. It's useful during your sprint planning meetings, to help you decide how much work you can feasibly commit to.



Fig. 6.3.2 Velocity Report

7. CODING & SOLUTIONING

CodingSolutions is a highly competitive job accelerator and talent refinement program that recruits and transitions college graduates with past programming experience or technical degrees into professional careers with Alabama companies and organizations at no cost to the graduates.

7.1 FEATURE 1(FRONT END)

HTML, CSS, JAVASCRIPT

main.html

```

<html>
<head>
  <title>Handwritten </title>
  <link rel="stylesheet" href="main.css">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <script src="https://unpkg.com/feather-icons"></script>
</head>
<body>
  <div class="container">
    <div class="header">
      <h1>Handwritten Digit Recognizer</h1>
      <h2>Easy analyze and detect handwritten digits</h2>
    </div>
    <div class="sub_container">
      <div class="sub_container1">
        <div class="sub_container2">
          <form class="upload" action="/predict" method="post" enctype="multipart/form-data">
            <!-- select -->
            <label id="label" for="upload-image"><i data-feather="file-plus"></i>Select File</label>
            <input type="file" name="photo" id="upload-image" hidden>
            <!-- upload -->
          </form>
          
        </div>
      </div>
    </div>
  </div>
</body>
</html>

```



```

main.html - Notepad
File Edit View

<!-- select -->
<label id="label" form="upload-image"><i data-feather="file-plus"></i>Select File</label>

<input type="file" name="photo" id="upload-image" hidden>
<!--
upload -->

</form>


</div>
</div>
</div>
<p> <style = "text-align:center">Team ID:PWT2022TMD138341<br>Team: Karthick(TL), Nithiyanandas, Praveen Kumar, Amarnath</p>
<script rel="text/javascript">
    feather.replace();

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

    console.log("working...")
    select.addEventListener('change',(e)=>{
        e.preventDefault();

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

    })

</script>
<p> <style = "text-align:center">Team ID:PWT2022TMD138341</p>
</body>
</html>
Ln 18, Col 18 100% Windows (CRLF) UTF-8

```

predict.html

```

predict.html - Notepad
File Edit View

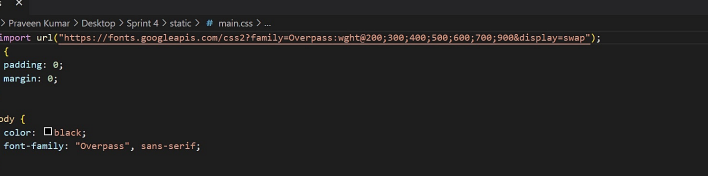
<html>
<head>
<title>result page</title>
<link rel="stylesheet" href="{{url_for('static',filename='css/result.css')}}"/>
</head>
<meta name="viewport" content="width=device-width, initial-scale=1.0"/>
<body>
<div class="container">
<h1>Prediction</h1>
<div class="sub_container1">

<div class="input_box box">

</div>
<div class="result_box box">
<div class="value">{{best.0}}</div>
<div class="v_accuracy">{{best.1}}%</div>
</div>
<h1>Other Prediction</h1>
<div class="other_predictions">
{% for x in others %}
<div class="p_box">
<h2>{{x.0}}</h2>
<div class="0_accu">{{x.1}}%</div>
</div>
{% endfor %}
</div>
</div>
</body>
</html>
Ln 1, Col 1 100% Windows (CRLF) UTF-8

```

main.css



```
1 @import url("https://fonts.googleapis.com/css2?family=Overpass:wght@200;300;400;500;600;700;900&display=swap");
2 * {
3   padding: 0;
4   margin: 0;
5 }
6
7 body {
8   color: black;
9   font-family: "Overpass", sans-serif;
10 }
11
12 .container {
13   width: 100%;
14   height: 100%;
15   display: flex;
16   flex-direction: column;
17   justify-content: center;
18   align-items: center;
19   background-color: white;
20 }
21
22 .container .heading {
23   margin-top: 2rem;
24   padding-bottom: 2rem;
25   width: fit-content;
26   text-align: center;
27 }
28
29 .container .heading h1 {
30   font-size: 3rem;
31   font-weight: 550;
32 }
33
34 .container .heading h2 {
35   font-size: 2rem;
36   font-weight: 550;
37 }
```

predict.css

File Edit Selection View Go Run Terminal Help predict.css - Visual Studio Code

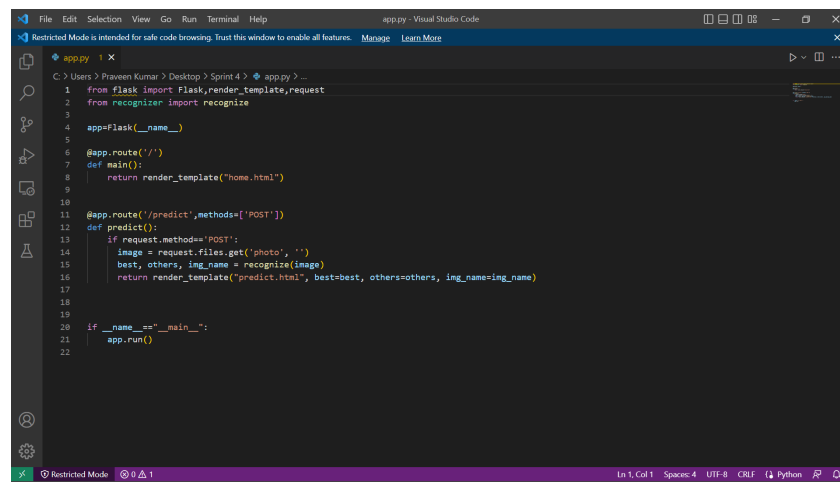
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

```
# main.css # predict.css
C:\Users> Praveen Kumar > Desktop > Sprint 4 > static > # predict.css > ...
1 @import url('https://fonts.googleapis.com/css2?family=Overpass:wght@200;300;400;500;600;700;900&display=swap');
2 body {
3   color: black;
4   font-family: "Overpass", sans-serif;
5 }
6
7 .container {
8   display: flex;
9   justify-content: center;
10  align-items: center;
11  flex-direction: column;
12 }
13 .container h1 {
14   padding-top: 1.2rem;
15 }
16
17 .container .sub_container1 {
18   width: -webkit-fit-content;
19   width: -moz-fit-content;
20   width: fit-content;
21   height: -webkit-fit-content;
22   height: -moz-fit-content;
23   height: fit-content;
24   box-shadow: 0 0 10px rgb(126, 125, 125);
25   padding: 1.5rem;
26   display: flex;
27   justify-content: center;
28   align-items: center;
29   -moz-column-gap: 1rem;
30   column-gap: 1rem;
31 }
```

VS Code status bar: Restricted Mode In-1, Col-1 Spaces: 2 UTF-8 CRLF CSS

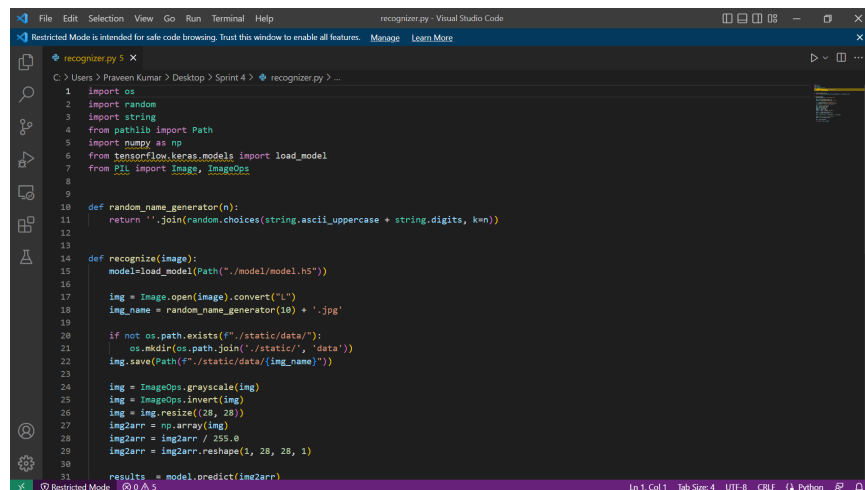
FEATURE 2 (BACK END)

app.py

A screenshot of a Visual Studio Code editor window showing the code for app.py. The file explorer on the left shows the file is named 'app.py'. The code defines a Flask application with two routes: a home page and a prediction endpoint. The prediction endpoint uses the recognizer module to process an image and return a template with the prediction results.

```
1 from flask import Flask, render_template, request
2 from recognizer import recognize
3
4 app = Flask(__name__)
5
6 @app.route('/')
7 def main():
8     return render_template("home.html")
9
10
11 @app.route('/predict', methods=['POST'])
12 def predict():
13     if request.method == 'POST':
14         image = request.files.get('photo', '')
15         best, others, img_name = recognize(image)
16         return render_template("predict.html", best=best, others=others, img_name=img_name)
17
18
19 if __name__ == '__main__':
20     app.run()
```

recognizer.py

A screenshot of a Visual Studio Code editor window showing the code for recognizer.py. The file explorer on the left shows the file is named 'recognizer.py'. The code imports various modules including os, random, string, pathlib, numpy, tensorflow.keras.models, PIL, and ImageOps. It defines a function to generate random names and a function to recognize an image by loading a model, processing the image, and saving it to a static directory before making a prediction.

```
1 import os
2 import random
3 import string
4 from pathlib import Path
5 import numpy as np
6 from tensorflow.keras.models import load_model
7 from PIL import Image, ImageOps
8
9
10 def random_name_generator(n):
11     return ''.join(random.choices(string.ascii_uppercase + string.digits, k=n))
12
13
14 def recognize(image):
15     model = load_model(Path("../model/model.h5"))
16
17     img = Image.open(image).convert("L")
18     img_name = random_name_generator(10) + ".jpg"
19
20     if not os.path.exists("../static/data/"):
21         os.mkdir(os.path.join("../static/", "data"))
22     img.save(Path("../static/data/{img_name}"))
23
24     img = ImageOps.grayscale(img)
25     img = ImageOps.invert(img)
26     img = img.resize((28, 28))
27     img2arr = np.array(img)
28     img2arr = img2arr / 255.0
29     img2arr = img2arr.reshape(1, 28, 28, 1)
30
31     results = model.predict(img2arr)
```

```

17 img = Image.open(image).convert("L")
18 img_name = random_name_generator(10) + '.jpg'
19
20 if not os.path.exists(f"./static/data/"):
21     os.mkdir(os.path.join("./static/", 'data'))
22     img.save(Path(f"./static/data/{img_name}"))
23
24 img = ImageOps.grayscale(img)
25 img = ImageOps.invert(img)
26 img = img.resize((28, 28))
27 img2arr = np.array(img)
28 img2arr = img2arr / 255.0
29 img2arr = img2arr.reshape(1, 28, 28, 1)
30
31 results = model.predict(img2arr)
32 best = np.argmax(results,axis = 1)[0]
33
34 pred = list(map(lambda x: round(x*100, 2), results[0]))
35
36 values = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
37 others = list(zip(values, pred))
38 best = others.pop(best)
39
40 return best, others, img_name
41
42

```

8. TESTING

8.1 TEST CASES

TEST SCENERIO	EXPECTED RESULT	ACTUAL RESULT	STATUS	EXECUTED BY
Verify UI elements in the home page	The home page must be displayed properly	Work as per expected	Pass	Karthick, Amarnath
Check if the UI element are displayed properly in different screen size	The home page must be properly submitted	The UI not displayed properly	Fail	Praveen Kumar, Amarnath
Check if user can upload their file	The input image should be uploaded to the application successfully	Working as expected	PASS	Nithiyanandam, Praveen Kumar
Check if user cannot upload unsupported file	The application should not allow user to select a non image file	User is able to upload any file	FAIL	Nithiyanandam, Amarnath
Check if the page redirects to the result page once the input is given	All the routes should properly work	Working as expected	Pass	Nithiyanandam, Amarnath

Check if the result is displayed properly	The result should be displayed properly	Working as expected	Pass	Karthick, Praveen Kumar
Check if the other predictions are displayed properly	The other predictions should be displayed properly	Working as expected	Pass	Amarnath, Nithiyanandam
Check if the model predicts the digit	The model should predict the number	Working as expected	Pass	Praveen Kumar, Nithiyanandam
Check if the model can handle complex input image	The model should predict the number in the complex image	The model fails to identify the digit since the model is not built to handle such data	Fail	Nithiyanandam, Amarnath

8.2 UAT REPORT

DEFECT ANALYSIS

Resolution	Severity1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	5	8	3	3	19
Duplicate	2	1	5	0	8
External	5	4	9	0	18
Fixed	13	5	9	18	45
Not produced	0	0	1	0	1
Skipped	0	0	1	0	1
Won't Fix	0	4	1	0	5
Total	25	22	29	21	97

TEST CASE ANALYSIS

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2

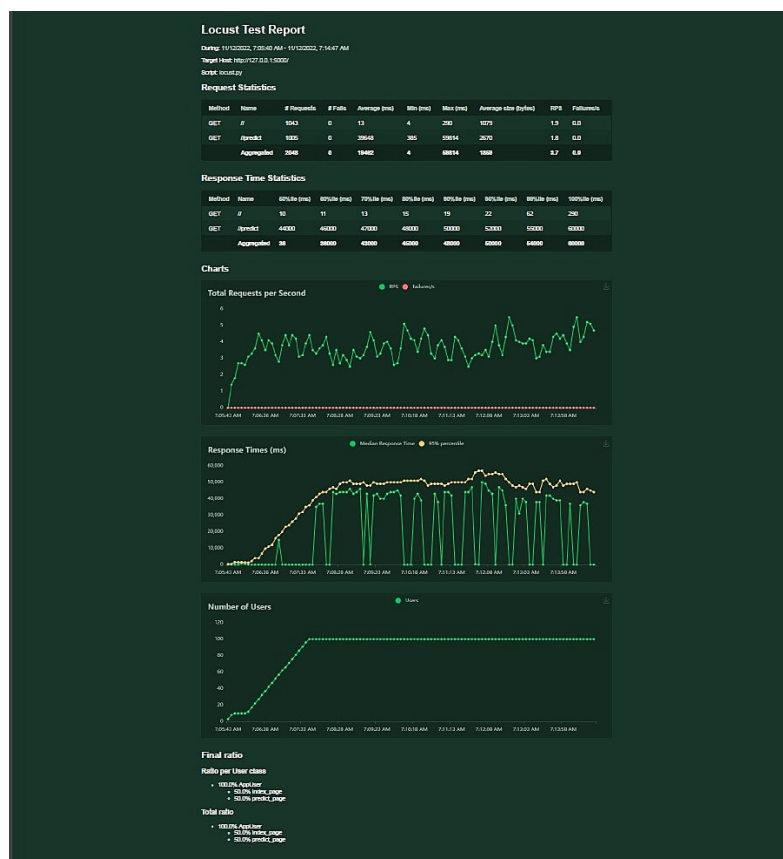
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Vision Control	2	0	0	2

9. RESULTS

The results section of the research paper is where you report the findings of your study based upon the information gathered as a result of the methodology [or methodologies] you applied. The results section should simply state the findings, without bias or interpretation, and arranged in a logical sequence.

9.1 PERFORMANCE METRICS

Performance metrics are defined as figures and data representative of an organization's actions, abilities, and overall quality.



10. ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- 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
- The generative models can perform recognition driven segmentation
- The method involves a relatively small number of parameters and hence training is relatively easy and fast
- Unlike many other recognition schemes, it does not rely on some form of pre-normalization of input images, but can handle arbitrary scalings, translations and a limited degree of image rotation.

DISADVANTAGES

- Good Handwriting is not all it's cracked up to be either
- It is not just poor handwriting that can cause an issue, but also joined / cursive handwritten text is known to cause similar issues
- The Perils of Poor Handwriting.

11. CONCLUSION

The objectives with which this project was initiated such as to develop handwritten digit recognizing system that enables users to recognize their handwritten digits using this deep learning model, less computation intensive efficient model has been achieved. The model which I built got an average accuracy of 98.23%.

Also the underlying problems of not having the same size ,width, orientation, and margin always has been taken care of with the help of computer vision's opencv library's functionalities. The problem of difficulty in distinguishing the difference between digits such as 1 and 7, 5 and 6, 3 and 8 etc has been resolved to a great extent with the opencv's edge detection and contour features. Also problems of dim lighting and blurry or unclear edges in images are corrected with the help of Gaussian blur technique. Now users can find their handwritten digits in one go without much complications .

This project is based on a deep neural network where users are going to get an interface for recognition of their digit images.

On the top of this model, this project can be extended to append various functionalities which can be used to filter the desired results based on digits recognized by this model.

For instance, if any academic institute wants to disburse scholarship to their talented students who lacks money can use this model to process forms submitted by students and filters the needy students. Many regional languages throughout world have different writing styles which can be recognized with HCR systems using proper algorithm and strategies. We have learning for recognition of English characters. It has been found that recognition of handwritten character becomes difficult due to presence of odd characters or similarity in shapes for multiple characters. Scanned image is pre-processed to get a cleaned image and the characters are isolated into individual characters. Preprocessing work is done in which normalization, filtration is performed using processing steps which produce noise free and clean output.

Managing our evolution algorithm with proper training, evaluation other step wise process will lead to successful output of system with better efficiency. Use of some statistical features and geometric features through neural network will provided better recognition result of English characters. This work will be helpful to the researchers for the work towards other script

12. FUTURE SCOPE

This work further extended to the character recognition for other languages. It can be used to convert the fax and news papers into text format. In order to recognize words, sentences or paragraphs we can use multiple ANN for classification. It can be used in post office for reading postal address.

13. APPENDIX

SOURCE CODE

main.html

```
<html>
  <head>
    <title>hand-written </title>
```



```

<link rel="stylesheet" href="main.css">

<meta name="viewport" content="width=device-width, initial-scale=1.0">
    <script src="https://unpkg.com/feather-icons"></script>

</head>
<body>

    <div class="container">
        <div class="heading">
            <h1>Handwritten Digit Recognizer</h1>
            <h2>Easy analyze and detect handwritten digits</h2>

        </div>
        <div class="sub_container1">
            <div class="sub_container2">

                <form class="upload" action="/predict" method="post"
enctype="multipart/form-data">
                    <!-- select -->
                    <label id="lable" for="upload-image"><i data-feather="file-plus"></i>Select
File</label>

                    <input type="file" name="photo" id="upload-image" hidden>
                    <!--      upload -->

                </form>
                

            </div>
        </div>
    </div>
    <p> <style = "text-align:center">Team ID:PNT2022TMID38341<br>Team: Karthick(TL),
Nithiyanandam, Praveen Kumar, Amarnath</p>
    <script rel="text/javascript">

```

```

        feather.replace();

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

        console.log("working...")
        select.addEventListener('change',(e)=>{
            e.preventDefault();

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

        })

    </script>
    <p> <style = "text-align:center">Team ID:PNT2022TMID38341</p>
    </body>
</html>

```

predict.html

```

<html>
<head>
    <title>result page</title>
    <link rel="stylesheet" href="{{url_for('static',filename='css/result.css')}}"/>

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
</head>
<body>
    <div class="container">
        <h1>Prediction</h1>
        <div class="sub_container1">

```

```

        <div class="input_box box">
            
        </div>
        <div class="result_box box">
            <div class="value">{{best.0}}</div>
            <div class="v_accuracy">{{best.1}}%</div>
        </div>
    </div>
    <h1>Other Prediction</h1>
    <div class="other_predictions">
        {% for x in others %}
            <div class="r_box">
                <h2>{{x.0}}</h2>
                <div class="0_accu">{{x.1}}%</div>
            </div>
        {% endfor %}
    </div>
</div>
</body>
</html>

```

main.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;
}

```

```
.container {
  width: 100%;
  height: 100%;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
  background-color: white;
}

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

.container .heading h1 {
  font-size: 3rem;
  font-weight: 550;
}

.container .heading h2 {
  font-size: 1rem;
  color: rgb(90, 88, 88);
}

.container .sub_container1 {
  box-shadow: 0 0 20px rgb(172, 170, 170);
  width: 40rem;
  height: 25rem;
  padding: 1.5rem;
}

.container .sub_container2 {
  background-color: rgba(190, 190, 190, 0.5);
  width: 100%;
  height: 100%;
  display: flex;
  border: 1px dashed black;
  justify-content: center;
  align-items: center;
}
```

```

}
.container .sub_container2 .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: white;
  background-color: rgb(114, 96, 182);
  box-shadow: 0 5px 10px rgb(146, 135, 247);
}

.container .sub_container2 #loading {
  display: none;
  justify-content: center;
  align-items: center;
  width: 10rem;
  height: auto;
  position: absolute;
}

.container .sub_container2 .upload label {
  font-size: 1rem;
  font-weight: 600;
  color: white;
  height: 100%;
  width: 100%;
  padding: 10px;
  display: block;
}

.container .sub_container2 .upload svg
{
  height: 15px;

```

```
width: auto;
padding-right: 8px;
margin-bottom: -2px;
}
```

```
@media screen and (max-width:700px) {
  .container .sub_container1 {
    height: 20rem;
    width: 18rem;
    margin-top: 3.5rem;
    margin-bottom: -8rem;

  }
  .container .heading h1 {
    margin-top: -6rem;
    font-size: 2rem;
    padding-bottom:1rem ;

  }
}
```

predict.css

```
@import
url("https://fonts.googleapis.com/css2?family=Overpass:wght@200;300;400;500;600;700;900&display=swap");
body {
  color: black;
  font-family: "Overpass", sans-serif;
}

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

```

.container h1 {

    padding-top: 1.2rem;
}
.container .sub_container1 {
    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(126, 125, 125);
    padding: 1.5rem;
    display: flex;
    justify-content: center;
    align-items: center;
    -moz-column-gap: 1rem;
    column-gap: 1rem;
}
.container .sub_container1 .box {
    width: 15rem;
    height: 15rem;
    border: 1px dashed black;
    justify-content: center;
    display: flex;
    align-items: center;
    flex-direction: column;
    background-color: rgb(209, 206, 206);
}

.container .sub_container1 .input_box img {
    width: 60%;
    height: 60%;
    background-color: aqua;
    background-size: contain;
}

```

```

.container .sub_container1 .result_box .value {
  font-size: 6rem;
}
.container .sub_container1 .result_box .v_accuracy{
  margin-top:-1rem ;
}
.container .other_predictions {
  display: flex;
  justify-content: center;
  align-items: center;
  flex-wrap: wrap;
  column-gap: 1rem;
  row-gap: 1rem;
  font-weight: 700;
}
.container .other_predictions .r_box {
  display: flex;
  justify-content: center;
  align-items: center;
  flex-direction: column;

  width: 5rem;
  height: 5rem;
  box-shadow: 0 0 7px rgb(158, 157, 157);
}
.container .other_predictions .r_box div {
  margin-top: -1.2rem;
}/*# sourceMappingURL=result.css.map */

@media screen and (max-width:700px){
  .container h1 {
    font-size: 2.3rem;
  }
  .container .sub_container1 .box {
    width: 7rem;
    height: 7rem;
  }

```



```
}  
.container .sub_container1 .result_box .value {  
  font-size: 4rem;  
}  
}
```

app.py

```
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()
```

recognizer.py

```
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

def random_name_generator(n):
    return ".join(random.choices(string.ascii_uppercase + string.digits, k=n))

def recognize(image):
    model=load_model(Path("./model/model.h5"))

    img = Image.open(image).convert("L")
    img_name = random_name_generator(10) + '.jpg'

    if not os.path.exists(f"./static/data/"):
        os.mkdir(os.path.join('./static/', 'data'))
    img.save(Path(f"./static/data/{img_name}"))

    img = ImageOps.grayscale(img)
    img = ImageOps.invert(img)
    img = img.resize((28, 28))
    img2arr = np.array(img)
    img2arr = img2arr / 255.0
    img2arr = img2arr.reshape(1, 28, 28, 1)

    results = model.predict(img2arr)
    best = np.argmax(results,axis = 1)[0]

```

```
pred = list(map(lambda x: round(x*100, 2), results[0]))
```

```
values = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
others = list(zip(values, pred))
```

```
best = others.pop(best)
```

```
return best, others, img_name
```

model.h5

```
import cv2
```

```
import numpy as np
```

```
from keras.datasets import mnist
```

```
from keras.layers import Dense, Flatten, MaxPooling2D, Dropout
```

```
from keras.layers.convolutional import Conv2D
```

```
from keras.models import Sequential
```

```
from keras.utils import to_categorical
```

```
import matplotlib.pyplot as plt
```

```
(X_train, y_train), (X_test, y_test) = mnist.load_data()
```

```
plt.imshow(X_train[0], cmap="gray")
```

```
plt.show()
```

```
print (y_train[0])
```

```
print ("Shape of X_train: {}".format(X_train.shape))
```

```
print ("Shape of y_train: {}".format(y_train.shape))
```

```
print ("Shape of X_test: {}".format(X_test.shape))
```

```
print ("Shape of y_test: {}".format(y_test.shape))
```

```
# Reshaping so as to convert images for our model
```

```
X_train = X_train.reshape(60000, 28, 28, 1)
```

```
X_test = X_test.reshape(10000, 28, 28, 1)
```

```
print ("Shape of X_train: {}".format(X_train.shape))
```

```
print ("Shape of y_train: {}".format(y_train.shape))
```

```
print ("Shape of X_test: {}".format(X_test.shape))
```

```
print ("Shape of y_test: {}".format(y_test.shape))
```

```
#one hot encoding
```

```
y_train = to_categorical(y_train)
```

```
y_test = to_categorical(y_test)
```

```
model = Sequential()
```

```
## Declare the layers
```

```
layer_1 = Conv2D(64, kernel_size=3, activation='relu', input_shape=(28, 28, 1))
```

```
layer_2 = MaxPooling2D(pool_size=2)
```

```
layer_3 = Conv2D(32, kernel_size=3, activation='relu')
```

```
layer_4 = MaxPooling2D(pool_size=2)
```

```
layer_5 = Dropout(0.5)
```

```
layer_6 = Flatten()
```

```
layer_7 = Dense(128, activation="relu")
```

```
layer_8 = Dropout(0.5)
```

```
layer_9 = Dense(10, activation='softmax')
```

```
## Add the layers to the model
```

```
model.add(layer_1)
```

```
model.add(layer_2)
```

```
model.add(layer_3)
```

```
model.add(layer_4)
```

```
model.add(layer_5)
```

```
model.add(layer_6)
```

```
model.add(layer_7)
```

```
model.add(layer_8)
```

```
model.add(layer_9)
```

```
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

(Refer more for model in Train on IBM Cloud)

Github Link: <https://github.com/IBM-EPBL/IBM-Project-41047-1660638949>

Project Demo Video: <https://www.youtube.com/watch?v=F930D-gGcGU>