## HAND WRITTEN DIGIT RECOGNISOR

PFSD Project Report

Submitted in the partial fulfillment of the requirements for the award of the degree of

# Bachelor of Technology in

Department of Computer Science and Engineering

By

SJ Sumanth (2010030377)

K. Sreevarun (2010030451)

E. Pavan Sai (2010030538)

E. Shiva Goud (2010030542)

under the supervision of

# **Mr. ABHISHEK TRIPATHI**

# Assistant professor



Department of Computer Science and Engineering

K L University Hyderabad,

Aziz Nagar, Moinabad Road, Hyderabad – 500 075, Telangana, India.

March, 202

**Declaration**

The PFSD Report entitled “**HAND WRITTEN-DIGIT RECOGNISOR**” is a record of bonafide work of S.J Sumanth (2010030377), K. Sreevarun (20100030451),

E. Pavan Sai (2010030538), E. Shiva Goud (2010030542), submitted in partial fulfillment for the award of B.Tech in the Department of Computer Science and Engineering to the K L University, Hyderabad. The results embodied in this report have not been copied from any other Departments/ University/ Institute.

**Certificate**

This is to certify that the Social Internship Report entitled “HAND WRITTEN-DIGIT RECOGNISOR” is being submitted S. J Sumanth (2010030377), K. SreeVarun (20100030451), E. Pavan Sai (2010030538), E. Shiva Goud (2010030542) submitted in partial fulfillment for the award of B.Tech in CSE to the K L University, Hyderabad is a record of bonafide work carried out under our guidance and supervision.

The results embodied in this report have not been copied from any other departments/ University/Institute.

## Signature of the Supervisor

## MR. ABHISHEK TRIPATHI

## Assistant professor

# 

## Signature of the HOD Signature of the External Examin

**ACKNOWLEDGEMENT**

First and foremost, we thank the lord almighty for all his grace & mercy showered upon us, for completing this project successfully.

We take grateful opportunity to thank our beloved Founder and Chairman who has given constant encouragement during our course and motivated us to do this project. We are grateful to our Principal **Dr. L. Koteswara Rao** who has been constantly bearing the torch for all the curricular activities undertaken by us.

We pay our grateful acknowledgement & sincere thanks to our Head of the Department **Dr. Chiranjeevi Manike** for his exemplary guidance, monitoring, and constant encouragement throughout the project. We thank Mr. Abhishek Tripathi of our department who has supported throughout this Project holding a position of supervisor.

We whole heartedly thank all the teaching and non-teaching staff of our department without whom we won’t have made this project a reality. We would like to extend our sincere thanks especially to our parent, our family members and friends who have supported us to make this project a grand success.

**ABSTRACT**

Digit Recognition is a noteworthy and important issue. As the manually written digits are not of a similar size, thickness, position and direction, in this manner, various difficulties must be considered to determine the issue of handwritten digit recognition. The uniqueness and assortment in the composition styles of various individuals additionally influence the example and presence of the digits. It is the strategy for perceiving and arranging transcribed digits. It has a wide range of applications, for example, programmed bank checks, postal locations and tax documents and so on. The aim of this project is to implement a classification algorithm to recognize the handwritten digits. Optical Character Recognition (OCR) is a subfield of Image Processing which is concerned with extracting text from images or scanned documents.

An OCR system depends mainly on feature extraction, recognition and classification into appropriate labels. In this project, we have chosen to focus on recognizing handwritten digits with at most accuracy. The challenge in this project is to use CNN algorithm, in order to maximize the accuracy of predicting the handwritten digits.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| S.no | Title | Page number |
| 01 | Introduction | 09 |
| 02 | Literature Survey | 10 |
| 03 | Hardware & software requirement | 11 |
| 04 | Functional and Non-Functional requirement | 12 |
| 05 | Convolutional Neural Network Architecture | 13 |
| 06 | Proposed system | 18 |
| 07 | Implementation | 18 |
| 08 | Results Discussion | 20 |
| 09 | Conclusion and Future Work | 21 |
| 10 | Reference | 22 |

**1.Introduction**

Recognition is identifying or distinguishing a thing or an individual from the past experiences or learning. Similarly, Digit Recognition is nothing but recognizing or identifying the digits in any document. Digit recognition framework is simply the working of a machine to prepare itself or interpret the digits. Handwritten Digit Recognition is the capacity of a computer to interpret the manually written digits from various sources like messages, bank cheques, papers, pictures, and so forth and in various situations for web-based handwriting recognition on PC tablets, identifying number plates of vehicles, handling bank cheques, digits entered in any forms etc. Machine Learning provides various methods through which human efforts can be reduced in recognizing the manually written digits. Deep Learning is a machine learning method that trains computers to do what easily falls into place for people: learning through examples. With the utilization of deep learning methods, human attempts can be diminished in perceiving, learning, recognizing and in a lot more regions. Using deep learning, the computer learns to carry out classification works from pictures or contents from any document. Deep Learning models can accomplish state-of-art accuracy, beyond the human level performance. The digit recognition model uses large datasets in order to recognize digits from distinctive sources. Handwriting recognition of characters has been around since the 1980s. The task of handwritten digit recognition, using a classifier, has extraordinary significance and use such as – online digit recognition on PC tablets, recognize zip codes on mail, processing bank check amounts, numeric sections in structures filled up by hand (for example ‐ tax forms) and so on. There are diverse challenges faced while attempting to solve this problem. The handwritten digits are not always of the same size, thickness, or orientation and position relative to the margins. The main objective was to actualize a pattern characterization method to perceive the handwritten digits provided in the MINIST data set of images of handwritten digits (0‐9).

**2.Literature Survey**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.no** | **Title** | **Author** | **Journal and year** | **scope** |
| 1 | Handwritten Digit Recognition:A Neural Network Demo | erend Jan van der Zwaag | Research Gate  --2001 | Improved detection in digits |
| 2 | Handwritten Digit Recognition using Machine and Deep Learning Algorithms | Ritik Dixit,  Rishika Kushwah,  Samay Pashine | opensource  -2021 | Improvement using deep learning algo |
| 3 | Recognition of Handwritten Digit using Convolutional Neural Network in Python with Tensorflow and Comparison of Performance for Various Hidden Layers | Fathma Siddique, Shadman Sakib , Md. Abu Bakr Siddique | Opensource  -2018 | Recognition using cnn neural networks |
| 4 | Offline Handwritten Digits Recognition Using Machine learning | Shengfeng Chen,  Rabia,  Almamlook,  Dr. Lee wells | ICIE  -2018 | Without intenet connection working |

Anuj Dutt in his paper demonstrated that utilizing Deep Learning systems, he had the capacity to get an extremely high measure of accuracy. By utilizing the convolutional Neural Network with Keras and Theano as backend, he was getting a accuracy of 98.72%. In addition, execution of CNN utilizing TensorFlow gives a stunningly better consequence of 99.70%. Despite the fact that the complication of the procedure and codes appears to be more when contrasted with typical Machine Learning algorithms yet the accuracy he got is increasingly obvious.

In a paper published by Saeed AL-Mansoori, Multilayer Perceptron (MLP) Neural Network was implemented to recognize and predict handwritten digits from 0 to 9. The proposed neural system was trained and tested on a dataset achieved from MNIST.

**3. Hardware and Software Requirements**

## Recommended Operating Systems

* **Windows:**7 or newer
* **MAC:** OS X v10.7 or higher
* **Linux:** Ubuntu

## Hardware Requirements

We strongly recommend a computer fewer than 5 years old.

* Processor: Minimum 1 GHz; Recommended 2GHz or more
* Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
* Hard Drive: Minimum 32 GB; Recommended 64 GB or more
* Memory (RAM): Minimum 1 GB; Recommended 4 GB or above

## Software Requirements:

Python (version: 3.7 or higher) with Flask, Django, NumPy, Keras, Pillow,

Matplotlib, Torch Packages.

Pycharm.

**4.Functional and Non-Functional requirement**

**4.1 Functional Requirements**

Import the Modified National Institute of Science and Technology Dataset:

* 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

Create the model:

* Take the value of the color is the pixels
* Put the value of all the pixels in a one-dimensional array
* Build a Neural Network with a number of nodes in the input layer equal to the number of pixels in the arrays
* Activate the Neural Network
* Test the precision of the model in a testing set

Recognize handwritten number input:

* Allow user to input a number.
* Predict in real-time the value of the number written

**4.2 Non-Functional Requirements**

* Using Python
* Using TensorFlow

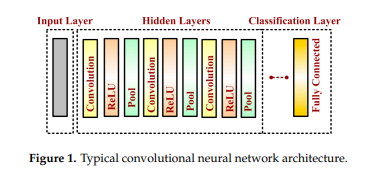
**5.Convolutional Neural Network Architecture**

A basic convolutional neural network comprises three components, namely, the convolutional layer, the pooling layer and the output layer. The pooling layer is optional sometimes. The typical convolutional neural network architecture with three convolutional layers is well adapted for the classification of handwritten images. It consists of the input layer, multiple hidden layers (repetitions of convolutional, normalization, pooling) and a fully connected and an output layer.

Neurons in one layer connect with some of the neurons present in the next layer, making the scaling easier for the higher resolution images. The operation of pooling or sub-sampling can be used to reduce the dimensions of the input. In a CNN model, the input image is considered as a collection of small sub-regions called the “receptive fields”. A mathematical operation of the convolution is applied on the input layer, which emulates the response to the next layer. The response is basically a visual stimulus. The detailed description is as follows:

**5.1 Input Layer**

The input data is loaded and stored in the input layer. This layer describes the height, width and number of channels (RGB information) of the input image.



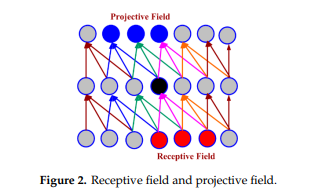
**5.2 Hidden Layer**

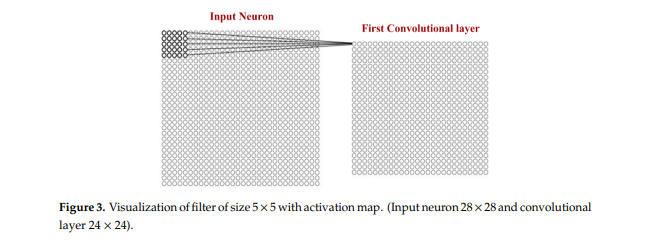
The hidden layers are the backbone of CNN architecture. They perform a feature extraction process where a series of convolution, pooling and activation functions are used. The distinguishable features of handwritten digits are detected at this stage.

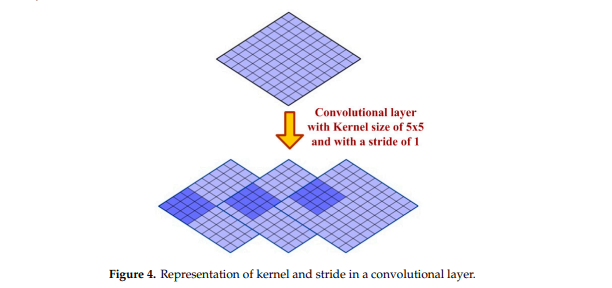
**5.3 Convolutional Layer**

The convolutional layer is the first layer placed above the input image. It is used for extracting the features of an image. The n × n input neurons of the input layer are convoluted with an m × m filter and in return deliver (n − m + 1) × (n − m + 1) as output. It introduces non-linearity through a neural activation function. The main contributors of the convolutional layer are receptive field, stride, dilation and padding, as described in the following paragraph. CNN computation is inspired by the visual cortex in animals. The visual cortex is a part of the brain that processes the information forwarded from the retina.

It processes visual information and is subtle to small sub-regions of the input. Similarly, a receptive field is calculated in a CNN, which is a small region of an input image that can affect a specific region of the network. It is also one of the important design parameters of the CNN architecture and helps in setting other CNN parameters. It has the same size as the kernel and works in a similar fashion as the foveal vision of the human eye works for producing sharp central vision. The receptive field is influenced by striding, pooling, kernel size and depth of the CNN. Receptive field (r), effective receptive field (ERF) and projective field (PF) are terminology used in calculating effective sub-regions in a network. The area of the original image influencing the activation of a neuron is described using the ERF, whereas the PF is a count of neurons to which neurons project their outputs. Stride is another parameter used in CNN architecture. It is defined as the step size by which the filter moves every time. A stride value of 1 indicates the filter sliding movement pixel by pixel. A larger stride size shows less overlapping between the cells.







The concept of padding is introduced in CNN architecture to get more accuracy. Padding is introduced to control the shrinking of the output of the convolutional layer. The output from the convolutional layer is a feature map, which is smaller than the input image. The output feature map contains more information on middle pixels and hence loses lots of information present on corners. The rows and the columns of zeros are added to the border of an image to prevent shrinking of the feature map. Equations (1) and (2) describe the relationship between the size of the feature map, the kernel size and stride while calculating the size of the output feature map.

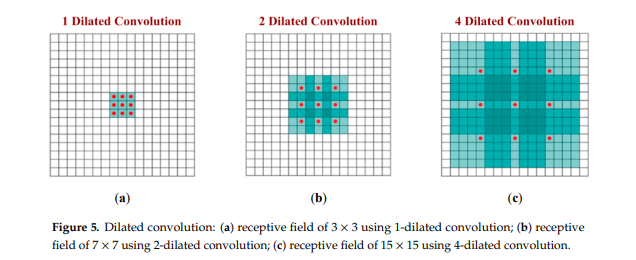
Wnx = Wn−1x − FnxSnx + 1------ (1)

Wny = Wn−1y − FnySny + 1------ (2)

where (Wnx, Wny) represent the size of the output feature map, (Snx, Sny) is stride size, and (Fx, Fy) is kernel size. Here ‘n’ is used to describe the index of layers. The dilation is another important parameter of CNN architecture that has a direct influence on the receptive field. The dilation can increase the field-of-view (FOV) of a CNN without modifying the feature map . Figure 5 clearly shows that dilation values can exponentially raise the receptive field of a CNN. Too large a dilation can increase the number of computations and hence can slow down the system by increasing the processing time. Therefore, it must be chosen wisely. The relationship between dilation, weight and input is shown in Equations (3) and (4) below.

0 − dialation = w[0] ∗ x[0] + w[1] ∗ x[1] + w[2] ∗ x[2]; -------(3)

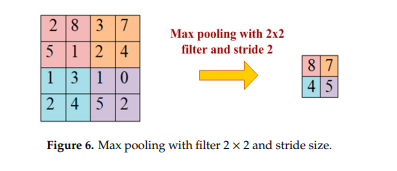
1 − dialation = w[0] ∗ x[0] + w[1] ∗ x[2] + w[2] ∗ x[4]; -------(4)



**5.4. Pooling Layer:**

A pooling layer is added between two convolutional layers to reduce the input dimensionality and hence to reduce the computational complexity. Pooling allows the selected values to be passed to the next layer while leaving the unnecessary values behind. The pooling layer also helps in feature selection and in controlling overfitting. The pooling operation is done independently. It works by extracting only one output value from the tiled non-overlapping sub-regions of the input images. The common types of pooling operations are max-pooling and avg-pooling (where max and avg represent maxima and average, respectively). The max-pooling operation is generally favorable in modern applications, because it takes the maximum values from each sub-region, keeping maximum information. This leads to faster convergence and better generalization. The max-pooling operation for converting a 4 × 4 convolved outputs into a 2 × 2 output with stride size 2 is described in Figure 6. The maximum number is taken from each convolved output

(Of size 2 × 2) resulting in reducing the overall size to 2 x 2.

****

**5.5. Activation Layer:**

Just like regular neural network architecture, CNN architecture also contains the activation function to introduce the non-linearity in the system. The sigmoid function, rectified linear unit (ReLu) and Softmax are some famous choices among various activation functions exploited extensively in deep learning models. It has been observed that the sigmoid activation function might weaken the CNN model because of the loss of information present in the input data. The activation function used in the present work is the non-linear rectified linear unit (ReLu) function, which has output 0 for input less than 0 and raw output otherwise. Some advantages of the ReLu activation function are its similarity with the human nerve system, simplicity in use and ability to perform faster training for larger networks.

**5.6. Classification Layer**

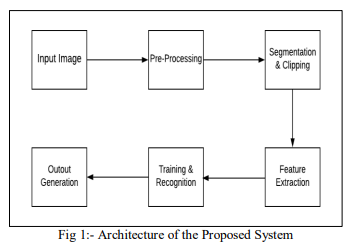
The classification layer is the last layer in CNN architecture. It is a fully connected feed forward network, mainly adopted as a classifier. The neurons in the fully connected layers are connected to all the neurons of the previous layer. This layer calculates predicted classes by identifying the input image, which is done by combining all the features learned by previous layers. The number of output classes depends on the number of classes present in the target dataset. In the present work, the classification layer uses the ‘softmax’ activation function for classifying the generated features of the input image received from the previous layer into various classes based on the training data.

**6.Proposed system**

The handwritten digit recognition is the ability of computers to recognize human handwritten digits. It is a hard task for the machine because handwritten digits are not perfect and can be made with many different flavors. The handwritten digit recognition is the solution to this problem which uses the image of a digit and recognizes the digit present in the image. We are using CNN algorithm, Python Full Stack Development to implement this project.

**7.Implementation**

The reason behind this document is to look into the design possibilities of the proposed system, such as architecture design, block diagram, sequence diagram, data flow diagram and user interface design of the system in order to define the steps such as pre-processing, feature extraction, segmentation, classification and recognition of digits.



The above Figure 1 illustrates the architecture diagram of the proposed system. The proposed model contains the four stages in order to classify and detect the digits:

A. Pre-processing

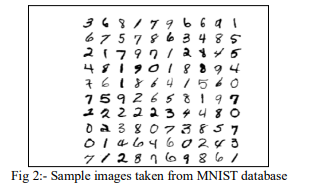
B. Segmentation

C. Feature Extraction

D. Classification and Recognition

**A. Pre-Processing:**

The role of the pre-processing step is it performs various tasks on the input image. It basically upgrades the image by making it reasonable for segmentation. The fundamental motivation behind pre-processing is to take off a fascinating example from the background. For the most part, noise filtering, smoothing and standardization are to be done in this stage. The pre-processing additionally characterizes a smaller portrayal of the example. Binarization changes over a gray scale image into a binary image. The initial approach to the training set images that are to be processed in order to reduce the data, by thresholding them into a binary image. The Figure 2 shows a sample of images taken from the MNIST database.



**B. Segmentation:**

Once the pre-processing of the input images is completed, sub-images of individual digits are formed from the sequence of images. Pre-processed digit images are segmented into a sub-image of individual digits, which are assigned a number to each digit. Each individual digit is resized into pixels. In this step an edge detection technique is being used for segmentation of dataset images.

**C. Feature Extraction:**

After the completion of pre-processing stage and segmentation stage, the pre-processed images are represented in the form of a matrix which contains pixels of the images that are of very large size. In this way it will be valuable to represent the digits in the images which contain the necessary information. This activity is called feature extraction. In the feature extraction stage redundancy from the data is removed.

**D. Classification and Recognition:**

In the classification and recognition step the extracted feature vectors are taken as an individual input to each of the following classifiers. In order to showcase the working system model extracted features are combined and defined using CNN algorithm.

**8.Results Discussion**

Graphical user interface, application

Description automatically generated

Application

Description automatically generated with low confidence

**9.Conclusion and Future Work**

Several OCR systems were studied and accurate methods, techniques were analysed.

OCR is a challenging area of research and introduces new challenges with its changing

dynamics. The most widely used Machine learning algorithm i.e., CNN is trained and tested on the same dataset CNN proves to be far better than other classifiers. The results can be made more accurate with more convolution layers and a greater number of hidden neurons. It can completely abolish the need for typing. Digit recognition is an excellent prototype problem for learning about neural networks and it gives a great way to develop more advanced techniques of deep learning. In future, we are planning to develop a real-time handwritten digit recognition system and also, we want to develop a system which can recognize alphabets.

**10.References**

1. DARPA: DARPA Neural Network Study. AFCEA International Press (1988) 60
2. Jain, L.C., Lazzerini, B. (eds.): Knowledge-based Intelligent Techniques in Character Recognition. CRC Press, Boca Raton FL (1999)
3. Kohonen, T.: Self-Organizing Maps. Springer-Verlag, Berlin (1995)
4. LeCun, Y., Jackel, L.D., Bottou, L., Brunot, A., Cortes, C., Denker, J.S., Drucker, H., Guyon, I., Muller, U.A., Sackinger, E., Simard, P., Vapnik, V.: Comparison of Learning Algorithms for Handwritten Digit Recognition. In: Fogelman, F., Gallinari, P. (eds.): Proc.

5. Int. Conf. on Artificial Neural Networks (1995) 53-60 5. Wentink, M.: Master-Slave

Kohonen Network Applied to Digit Recognition. Internal Report BSC-021N95,

University of Twente (1995).

6. Nimisha Jain, Kumar Rahul, Ipshita Khamaru. AnishKumar Jha, Anupam Ghosh

(2017). “Hand Written Digit Recognition using Convolutional Neural Network

(CNN)”, International Journal of Innovations & Advancement in Computer Science,

IJIACS, ISSN 2347 – 8616, Volume 6, Issue 5.

7. Dr. Kusum Gupta,"A Comprehensive Review on Handwritten Digit Recognition

Using Various Neural Network Approaches", International Journal of Enhanced

Research In Management & Computer Applications, Vol. 5, No. 5, Pp. 22-25, 2016.

8. Huimin Wu. CNN-Based Recognition of Handwritten Digits in MNIST Database.

9. Nurul Ilmi, Tjokorda Agung Budi W and Kurniawan Nur R, “Handwriting Digit

Recognation using Local Binary Pattern Varience and K-Nearest Neighbor,” 2016

Fourth International Conference on Information and Communication Technologies

(ICoICT).

10. Tobias Kutzner, Mario Dietze, Ingrid Bönninger, Carlos M. Travieso, Malay Kishore

Dutta, and Anushikha Singh, “Online Handwriting Verification with Safe Password

and Increasing Number of Features,”2016 3rd International Conference on Signal

Processing and Integrated Networks (SPIN).