HANDWRITTEN DIGIT RECOGNITION

# A PROJECT REPORT

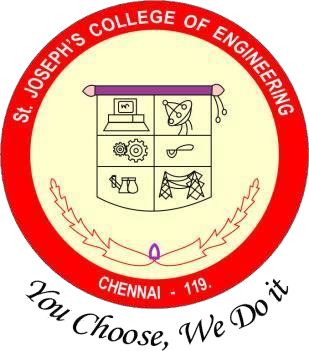
**Submitted By**

**RAMPRASAD D 312318205122**

**PREM KISHORE V 312318205109**

**in partial fulfillment for the award of the degree of BACHELOR OF TECHNOLOGY**

**in INFORMATION TECHNOLOGY**



**St.JOSEPH’S COLLEGE OF ENGINEERING**

# (An Autonomous Institution)

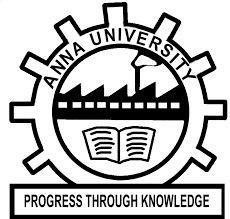
**St. Joseph’s Groups of Institution Jeppiaar Educational Trust**

**OMR, Chennai600119**

**ANNA UNIVERSITY: CHENNAI**

**May-2022**

# ANNA UNIVERSITY, CHENNAI 600 025



**BONAFIDE CERTIFICATE**

Certified that this project report “**HANDWRITTEN DIGIT RECOGNITION**” is the bonafide work of **RAMPRASAD D (312318205122)** and **PREM KISHORE V (312318205109)** who carried out the project work under my supervision, for the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Information Technology.

# SIGNATURE SIGNATURE

Dr. V. Muthulakshmi M.E., Ph.D., Associate Professor

# HEAD OF THE DEPARTMENT - LAB AFFAIRS

Department of Information Technology,

St. Joseph’s College of Engineering, Old Mamallapuram Road,

Chennai 600 119

Dr. G.Manikandan M.E., Ph.D.,Assistant Professor

# SUPERVISOR

Department of Information Technology,

St. Joseph’s College of Engineering, Old Mamallapuram Road,

Chennai 600 119

Submitted for the Viva-Voce held on

# (EXTERNAL EXAMINER) (INTERNAL EXAMINER)

**CERTIFICATE OF EVALUATION**

**College name :** St. Joseph’s College of Engineering

**Branch & Semester :** Information Technology (VIII)

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **NAME OF**  **THE STUDENTS** | **TITLE OF THE PROJECT** | **NAME OF THE SUPERVISOR** |
| 1.  2. | RAMPRASAD D  (312318205122)  PREM KISHORE V  (312318205109) | HANDWRITTEN  DIGIT  RECOGNITION | Dr. MANIKANDAN  M.E., Ph.D. |

The report of the project work submitted by the above students in partial fulfillment for the award of Bachelor of Technology degree in Information Technology of Anna University were evaluated and confirmed to be reports of the work done by the above students.

# (EXTERNAL EXAMINER) (INTERNAL EXAMINER)

**ACKNOWLEDGEMENT**

The contentment and elation that accompany the successful completion of any work be incomplete without mentioning the people who made it possible.

We express our gratitude in thanking our Chairman **Dr. B. Babu Manoharan M.A., M.B.A., Ph.D.,** our Managing Director **Mrs. B. Jessie Priya M.Com.,** our Director **Mr. B. Shashi Sekar M.Sc.,** our Principal **Dr. Vaddi Seshagiri Rao M.E., M.B.A., Ph.D** for having encouraged us to do our under graduation in Information Technology in this esteemed college.

We express our sincere thanks and most heartfelt sense of gratitude to our eminent Head of the Department-Lab Affairs **Dr. V. Muthulakshmi M.E., Ph.D.,** for having extended her helping hand at all times.

It is with a deep sense of gratitude that we acknowledge our indebtedness to our supervisor **Dr. G.Manikandan M.E., Ph.D.,** a perfectionist for his expert guidance and connoisseur suggestion.

Last but not the least, we thank our family members and friends who have been the greatest source of support to us.

# ABSTRACT

The reliance of humans over machines has never been so high such that from object classification in photographs to adding sound to silent movies everything can be performed with the help of deep learning and machine learning algorithms. Likewise, Handwritten text recognition is one of the significant areas of research and development with a streaming number of possibilities that could be attained. Handwriting recognition (HWR), also known as Handwritten Text Recognition (HTR), is the ability of a computer to receive and interpret intelligible handwritten input from sources such as paper documents, photographs, touch-screens and other devices . Apparently, in this paper, we have performed handwritten digit recognition with the help of MNIST datasets using Support Vector Machines (SVM), Multi-Layer Perceptron (MLP) and Convolution Neural Network (CNN) models. Our main objective is to compare the accuracy of the models stated above along with their execution time to get the best possible model for digit recognition.

**TABLE OF CONTENTS**

**CHAPTER TITLE PAGE NO**

ABSTRACT v

LIST OF FIGURES viii

1. **INTRODUCTION**
   1. DIGITS RECOGNITION 1
   2. SYSTEM OVERVIEW 2
   3. SCOPE OF THE PROJECT 3
2. **LITERATURE SURVEY** 4
3. **SYSTEM ANALYSIS**
   1. DRAWBACKS OF EXISTING 12

SYSTEM

* 1. ADVANTAGES OF PROPOSED 12

SYSTEM

* 1. REQUIREMENT SPECIFICATION
     1. Hardware Requirements 13
     2. Software Requirements 14

1. **SYSTEM DESIGN**
   1. SYSTEM ARCHITECTURE 16
   2. USE CASE DIAGRAM 20
   3. SEQUENCE DIAGRAM 21
   4. DATAFLOW DIAGRAM 23
   5. DEPLOYMENT DIAGRAM 24
   6. CLASS DIAGRAM 24
   7. COMPONENT DIAGRAM 25
2. **MODULE DESCRIPTION**
   1. MODULES
      1. DATA PREPROCESSING 26
      2. FEATURE EXTRACTION 26
      3. TRAINING MODEL 27
      4. GUI CREATION 27
3. **CONCLUSION** **AND FUTURE ENHANCEMENT**
   1. CONCLUSION 28
   2. FUTURE ENHANCEMENT 28

**APPENDIX 1** 29

**APPENDIX 2** 39

**REFERENCES** 43

**LIST OF FIGURES**

**FIGURE NO TITLE PAGE NO**

4.1SYSTEM ARCHITECTURE 16

4.2 MNIST DATASET 18

4.3 USE CASE DIAGRAM 20

4.4 SEQUENCE DIAGRAM 21

4.5 DATAFLOW DIAGRAM 23

4.6 DEPLOYMENT DIAGRAM 24

4.7 CLASS DIAGRAM 24

4.8 COMPONENT DIAGRAM 25

# CHAPTER 1 INTRODUCTION

**1.1. DIGIT RECOGNITION**

Handwritten digit recognition is the ability of a computer to recognize the human handwritten digits from different sources like images, papers, touch screens, etc, and classify them into 10 predefined classes (0-9). This has been a topic of boundless-research in the field of deep learning. Digit recognition has many applications like number plate recognition, postal mail sorting, bank check processing, etc [2]. In Handwritten digit recognition, we face many challenges because of different styles of writing of different peoples as it is not an Optical character recognition. This research provides a comprehensive comparison between different machine learning and deep learning algorithms for the purpose of handwritten digit recognition. For this, we have used Support Vector Machine, Multilayer Perceptron, and Convolutional Neural Network. The comparison between these algorithms is carried out on the basis of their accuracy, errors, and testing-training time corroborated by plots and charts that have been constructed using matplotlib for visualization.

The accuracy of any model is paramount as more accurate models make better decisions. The models with low accuracy are not suitable for real-world applications. Ex- For an automated bank cheque processing system where the system recognizes the amount and date on the check, high accuracy is very critical. If the system incorrectly recognizes a digit, it can lead to major damage which is not desirable. That’s why an algorithm with high accuracy is required in these realworld applications. Hence, we are providing a comparison of different algorithms based on their accuracy so that the most accurate algorithm with the least chances of errors can be employed in various applications of handwritten digit recognition.

Applications of offline handwriting recognition are numerous: reading postal addresses, bank check amounts, and forms. Furthermore, OCR plays an important role for digital libraries, allowing the entry of image textual information into computers by digitization, image restoration, and recognition methods.

**1.2. SYSTEM OVERVIEW**

**1.2.1 Support Vector Machine**

Support Vector Machine (SVM) is a supervised machine learning algorithm. In this, we generally plot data items in n-dimensional space where n is the number of features, a particular coordinate represents the value of a feature, we perform the classification by finding the hyperplane that distinguishes the two classes. It will choose the hyperplane that separates the classes correctly. SVM chooses the extreme vectors that help in creating the hyperplane. These extreme cases are called support vectors, and hence the algorithm is termed as Support Vector Machine

**1.2.2. Convolutional Neural Networks**

CNN is a deep learning algorithm that is widely used for image recognition and classification. It is a class of deep neural networks that require minimum pre-processing. It inputs the image in the form of small chunks rather than inputting a single pixel at a time, so the network can detect uncertain patterns (edges) in the image more efficiently

**1.2.3. Mnist Dataset**

We have used the MNIST dataset (i.e. handwritten digit dataset) to compare different level algorithm of deep and machine learning (i.e. SVM, ANN-MLP, CNN) on the basis of execution time, complexity, accuracy rate, number of epochs and number of hidden layers (in the case of deep learning algorithms).

**1.3 SCOPE OF THE PROJECT**

Recently handwritten digit recognition becomes vital scope and it is appealing many researchers because of its using in variety of machine learning and computer vision applications. As a challenging dataset is used for evaluation, a robust deep convolutional neural network is used for classification and superior results are achieved**.**

The aim of this project is to implement a classification algorithm to recognize the handwritten digits. The after effects of probably the most broadly utilized Machine Learning Algorithms like SVM, KNN and RFC and with Deep Learning calculation like multilayer CNN utilizing Keras with Tensorflow. Utilizing these, the accuracy of 90.70% utilizing CNN (Keras + Tensor) .

# CHAPTER 2 LITERATURE SURVEY

1. **Ritik Dixit, Rishika Kushwah, Samay Pashine , “Handwritten Digit Recognition using Machine and Deep Learning Algorithms” , International Journal of Computer Applications (ARXIV) , 2021.**

This paper provides a reasonable understanding of machine learning and deep learning algorithms like SVM, CNN, and MLP for handwritten digit recognition. It furthermore gives you the information about which algorithm is efficient in performing the task of digit recognition. In further sections of this paper, we will be discussing the related work that has been done in this field followed by the methodology and implementation of all the three algorithms for the fairer understanding of them. Next, it presents the conclusion and result bolstered by the work we have done in this paper. Moreover, it will also give you some potential future enhancements that can be done in this field. The last section of this paper contains citations and references used

# S M Shamim, Mohammad Badrul Alam Miah, Angona Sarker, Masud Rana & Abdullah Al Jobair , “Handwritten Digit Recognition using Machine Learning Algorithms” , Global Journal of Computer Science and Technology( GJCST ) , 2018.

# Handwritten character recognition is one of the practically important issues in pattern recognition applications.The applications of digit recognition includes in postal mail sorting, bank check processing, form data entry, etc. The heart of the problem lies within the ability to develop an efficient algorithm that can recognize hand written digits and which is submitted by users by the way of a scanner, tablet, and other digital devices. This paper presents an approach to off-line handwritten digit recognition based on different machine learning technique. The main objective of this paper is to ensure effective and reliable approaches for recognition of handwritten digits. Several machines learning algorithm namely, Multilayer Perceptron, Support Vector Machine, Naïve Bayes, Bayes Net, Random Forest, J48 and Random Tree has been used for the recognition of digits using WEKA. The result of this paper shows that highest 90.37% accuracy has been obtained for Multilayer Perceptron.

# [Mahmoud M. Abu Ghosh](https://ieeexplore.ieee.org/author/37085882676), [Ashraf Y. Maghari](https://ieeexplore.ieee.org/author/37085875507) , “A Comparative Study on Handwriting Digit Recognition Using Neural Networks” , International Conference on Promising Electronic Technologies (ICPET) , 2017.

The handwritten digit recognition problem becomes one of the most famous problems in machine learning and computer vision applications. Many machine learning techniques have been employed to solve the handwritten digit recognition problem. This paper focuses on Neural Network (NN) approaches. The most three famous NN approaches are deep neural network (DNN), deep belief network (DBN) and convolutional neural network (CNN). In this paper, the three NN approaches are compared and evaluated in terms of many factors such as accuracy and performance. Recognition accuracy rate and performance, however, is not the only criterion in the evaluation process, but there are interesting criteria such as execution time. Random and standard dataset of handwritten digit have been used for conducting the experiments. The results show that among the three NN approaches, DNN is the most accurate algorithm; it has 98.08% accuracy rate. However, the execution time of DNN is comparable with the other two algorithms. On the other hand, each algorithm has an error rate of 1-2% because of the similarity in digit shapes, specially, with the digits (1,7), (3,5), (3,8), (8,5) and (6,9)

# [Jinze Li](https://ieeexplore.ieee.org/author/37088482353),[Gongbo Sun](https://ieeexplore.ieee.org/author/37088522488), [Leiye Yi](https://ieeexplore.ieee.org/author/37088524958), [Qian Cao](https://ieeexplore.ieee.org/author/37088524590), [Fusen Liang](https://ieeexplore.ieee.org/author/37088520641), [Yu Sun](https://ieeexplore.ieee.org/author/37088520682) , “Handwritten Digit Recognition System Based on Convolutional Neural Network” , [IEEE International Conference on Advances in Electrical Engineering and Computer Applications ( AEECA)](https://ieeexplore.ieee.org/xpl/conhome/9210148/proceeding) , 2020.

Image recognition is widely used in the field of computer vision today. As a kind of image recognition, digit recognition is widely used. Today, the online recognition technology in digit recognition is relatively mature while the offline recognition technology is not. This paper mainly introduces an offline recognition system for handwritten digits based on convolutional neural networks. The system uses the MINST dataset as a training sample and pre-processes the picture with the Opencv toolkit. Then it uses LeNet-5 in the convolutional neural network to extract the handwritten digit image features, repeatedly convolution pooling, and pull the result into a one-dimensional vector. And finally find the highest probability point to determine the result to achieve handwritten digit recognition with the Softmax regression model. The application of this system can greatly reduce labor costs and improve work efficiency, which is of great significance in many fields.

With the rapid development of electronic information, computer input has become more and more common, but handwriting is still an irreplaceable way for people to transfer information. As a link combining handwritten characters and computer input, handwriting recognition has received more and more attention for its practicability. Handwriting recognition technology is the basis of handwriting interpolation and handwriting identification. In the past decade, machine learning and pattern recognition have extended many highly intelligent handwriting recognition classifications, including artificial neural networks (ANN) [1], support vector machine (SVM) [2], modified quadratic discriminant function (MQDF) [3] and hidden Markov model [4], etc.

1. **S. Knerr, L. Personnaz, G. Dreyfus , “Handwritten digit recognition by neural networks with single-layer training” , IEEE Transactions on Neural Networks (IEEE), 1992.**

# 

In Handwritten character recognition is one of the

practically important issues in pattern recognition applications.

The applications of digit recognition includes in postal mail

sorting, bank check processing, form data entry, etc. The heart

of the problem lies within the ability to develop an efficient

algorithm that can recognize hand written digits and which is

submitted by users by the way of a scanner, tablet, and other

digital devices. This paper presents an approach to off-line

handwritten digit recognition based on different machine

learning technique. The main objective of this paper is to

ensure effective and reliable approaches for recognition of

handwritten digits. Several machines learning algorithm

namely, Multilayer Perceptron, Support Vector Machine, Naïve

Bayes, Bayes Net, Random Forest, J48 and Random Tree has

been used for the recognition of digits using WEKA. The result

of this paper shows that highest 90.37% accuracy has been

obtained for Multilayer Perceptron

creasi Handwritten character recognition is one of the

practically important issues in pattern recognition applications.

The applications of digit recognition includes in postal mail

sorting, bank check processing, form data entry, etc. The heart

of the problem lies within the ability to develop an efficient

algorithm that can recognize hand written digits and which is

submitted by users by the way of a scanner, tablet, and other

digital devices. This paper presents an approach to off-line

handwritten digit recognition based on different machine

learning technique. The main objective of this paper is to

ensure effective and reliable approaches for recognition of

handwritten digits. Several machines learning algorithm

namely, Multilayer Perceptron, Support Vector Machine, Naïve

Bayes, Bayes Net, Random Forest, J48 and Random Tree has

been used for the recognition of digits using WEKA. The result

of this paper shows that highest 90.37% accuracy has been

obtained for Multilayer Perceptron

ng Handwritten character recognition is one of the

practically important issues in pattern recognition applications.

The applications of digit recognition includes in postal mail

sorting, bank check processing, form data entry, etc. The heart

of the problem lies within the ability to develop an efficient

algorithm that can recognize hand written digits and which is

submitted by users by the way of a scanner, tablet, and other

digital devices. This paper presents an approach to off-line

handwritten digit recognition based on different machine

learning technique. The main objective of this paper is to

ensure effective and reliable approaches for recognition of

handwritten digits. Several machines learning algorithm

namely, Multilayer Perceptron, Support Vector Machine, Naïve

Bayes, Bayes Net, Random Forest, J48 and Random Tree has

been used for the recognition of digits using WEKA. The result

of this paper shows that highest 90.37% accuracy has been

obtained for Multilayer Perceptro

mobil Handwritten character recognition is one of the

practically important issues in pattern recognition applications.

The applications of digit recognition includes in postal mail

sorting, bank check processing, form data entry, etc. The heart

of the problem lies within the ability to develop an efficient

algorithm that can recognize hand written digits and which is

submitted by users by the way of a scanner, tablet, and other

digital devices. This paper presents an approach to off-line

handwritten digit recognition based on different machine

learning technique. The main objective of this paper is to

ensure effective and reliable approaches for recognition of

handwritten digits. Several machines learning algorithm

namely, Multilayer Perceptron, Support Vector Machine, Naïve

Bayes, Bayes Net, Random Forest, J48 and Random Tree has

been used for the recognition of digits using WEKA. The result

of this paper shows that highest 90.37% accuracy has been

obtained for Multilayer Perceptro

it ligent image analysis is an appealing research

area in Artificial Intelligence and also crucial for a

variety of present open research difficulties.

Handwritten digits recognition is a well-researched

subarea within the field that is concerned with learning

models to distinguish pre-segmented handwritten digits.

It is one of the most important issues in data mining,

machine learning, pattern recognition along with many

other disciplines of artificial intelligence [1].The main

application of machine learning methods over the last

decade has determined efficacious

in conforming

decisive systems which are competing to human

performance and which accomplish far improved than

manually written classical

artificial intelligence systems

used in the beginnings of optical character recognition

technology [2]. However, not all features of those

specific models have been previously inspected.

A great attempt of research worker in machine

learning and data mining has been contrived to achieve

efficient approaches for approximation of recognition

from data [3]. In twenty first Century handwritten digit

communication has its own standard and most of the

times in daily life are being used as means of

conversation and recording the information to be shared

with individuals. One of the challenges in handwritten

characters recognition wholly lies in the variation and

distortion of handwritten character set because distinct

community may use diverse style of handwriting, and

control to draw the similar pattern of the characters of

their recognized script.

Identification of digit from where best

discriminating features can be extracted is one of the

major tasks in the area of digit recognition system. To

locate such regions different kind of region sampling

techniques are used in pattern recognition [4].The

challenge in handwritten character recognition is mainly

caused by the large variation of individual writing styles

[5]. Hence, robust feature extraction is very important to

improve the performance of a handwritten character

recognition system. Nowadays handwritten digit

recognition has obtained lot of concentration in the area

of pattern recognition system sowing to its application in

diverse fields. In next days, character recognition

system might serve as a cornerstone to initiate

paperless surroundings by digitizing and processing

existing paper documents.

y demands raise the pressure on existing transport networks. As the most used mode of transport, private cars have a particularly strong environmental impact and produce congestion. Ridesharing or carpooling, where a driver and several riders form a carpool, can help to address these issues by increasing the number of per ligent image analysis is an appealing research

area in Artificial Intelligence and also crucial for a

variety of present open research difficulties.

Handwritten digits recognition is a well-researched

subarea within the field that is concerned with learning

models to distinguish pre-segmented handwritten digits.

It is one of the most important issues in data mining,

machine learning, pattern recognition along with many

other disciplines of artificial intelligence [1].The main

application of machine learning methods over the last

decade has determined efficacious

in conforming

decisive systems which are competing to human

performance and which accomplish far improved than

manually written classical

artificial intelligence systems

used in the beginnings of optical character recognition

technology [2]. However, not all features of those

specific models have been previously inspected.

A great attempt of research worker in machine

learning and data mining has been contrived to achieve

efficient approaches for approximation of recognition

from data [3]. In twenty first Century handwritten digit

communication has its own standard and most of the

times in daily life are being used as means of

conversation and recording the information to be shared

with individuals. One of the challenges in handwritten

characters recognition wholly lies in the variation and

distortion of handwritten character set because distinct

community may use diverse style of handwriting, and

control to draw the similar pattern of the characters of

their recognized script.

Identification of digit from where best

discriminating features can be extracted is one of the

major tasks in the area of digit recognition system. To

locate such regions different kind of region sampling

techniques are used in pattern recognition [4].The

challenge in handwritten character recognition is mainly

caused by the large variation of individual writing styles

[5]. Hence, robust feature extraction is very important to

improve the performance of a handwritten character

recognition system. Nowadays handwritten digit

recognition has obtained lot of concentration in the area

of pattern recognition system sowing to its application in

diverse fields. In next days, character recognition

system might serve as a cornerstone to initiate

paperless surroundings by digitizing and processing

existing paper documents.

It is shown that neural network classifiers with single-layer training can be applied efficiently to complex real-world classification problems such as the recognition of handwritten digits. The STEPNET procedure, which decomposes the problem into simpler subproblems which can be solved by linear separators, is introduced. Provided appropriate data representations and learning rules are used, performance comparable to that obtained by more complex networks can be achieved. Results from two different databases are presented: an European database comprising 8700 isolated digits and a zip code database from the US Postal Service comprising 9000 segmented digits. A hardware implementation of the classifier is briefly described.

# Md. Anwar Hossain & Md. Mohon Ali , “Recognition of Handwritten Digit using Convolutional Neural Network (CNN)” , Global Journal of Computer Science and Technology (GJCST) , 2019.

Humans can see and visually sense the world around them by using their eyes and brains. Computer vision works on enabling computers to see and process images in the same way that human vision does. Several algorithms developed in the area of computer vision to recognize images. The goal of our work will be to create a model that will be able to identify and determine the handwritten digit from its image with better accuracy. We aim to complete this by using the concepts of Convolutional Neural Network and MNIST dataset. We will also show how MatConvNet can be used to implement our model with CPU training as well as less training time. Though the goal is to create a model which can recognize the digits, we can extend it for letters and then a person’s handwriting. Through this work, we aim to learn and practically apply the concepts of Convolutional Neural Networks.

# Vijayalaxmi R Rudraswamimath, Bhavanishankar K , “Handwritten Digit Recognition using CNN” , International Journal of Innovative Science and Research Technology (IJISRT) , 2019.

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. The after effects of probably the most broadly utilized Machine Learning Algorithms like SVM, KNN and RFC and with Deep Learning calculation like multilayer CNN utilizing Keras with Theano and Tensorflow. Utilizing these, the accuracy of 98.70% utilizing CNN (Keras + Theano) when contrasted with 97.91% utilizing SVM, 96.67% utilizing KNN, 96.89% utilizing RFC was obtained.

# Yevhen Chychkarova , Anastasiia Serhiienkob , Iryna Syrmamiikha , AnatoliiKarginc,“Handwritten Digits Recognition Using SVM, KNN, RF and Deep Learning Neural Networks”  , (CEUR-WS) , 2021.

This article discusses several classification algorithms of recognizing numbers from photographic images or with manual input, namely: support vector machine (SVM), Knearest neighbors (KNN), random forest (RF) and several variants of neural networks. The success rates of the algorithms in the field of handwriting recognition were compared. Six variants of recognition technology were analyzed and tested: using classifier from Scikitlearn package and using deep learning neural networks. Recognition images were scaled to a size of 28x28 (784 cells in one dimensional representation). Preliminary processing of images (filtering, scaling, etc.) was carried out using the OpenCV library. For recognition, each image of a digit was converted to a 28x28 size and fed to the input of a pre-trained neural network. A technique to select the area of interest in photographs containing hand-written digits for further recognition has been devised. For handwritten digit recognition, the best recognition accuracy is provided by a convolutional neural network, as 97.6% of car ladle digits were recognized correctly with it. After building recognition models using all the algorithms mentioned above, the recognition accuracy of all handwritten digits on the test program turned out to be within 98-100%. For industrial images regardless of the used neural network version, the recognition accuracy was 96-98%.

# [Farhana Sultana](https://ieeexplore.ieee.org/author/37086840827), [Abu Sufian](https://ieeexplore.ieee.org/author/37085659526), [Paramartha Dutta](https://ieeexplore.ieee.org/author/37429446600) , “Advancements in Image Classification using Convolutional Neural Network” , (IEEE) 2019.

# Convolutional Neural Network (CNN) is the state-of-the-art for image classification task. Here we have briefly discussed different components of CNN. In this paper, We have explained different CNN architectures for image classification. Through this paper, we have shown advancements in CNN from LeNet-5 to latest SENet model. We have discussed the model description and training details of each model. We have also drawn a comparison among those models.

# Pranit Patil, Bhupinder Kaur , “Handwritten Digit Recognition Using Various Machine Learning Algorithms and Models” , International Journal of Innovative Research in Computer Science & Technology (IJIRCST) , 2020.

# 

# Handwritten digit recognition is a technique or technology for automatically recognizing and detecting handwritten digital data through different Machine Learning models. In this paper we use various Machine Learning algorithms to enhance the productiveness of technique and reduce the complexity using various models. Machine Learning is an application of Artificial Intelligence that learns from previous experience and improves automatically through experience. We illustrate various Machine learning algorithms such as Support Vector Machine, Convolutional Neural Network, Quantum Computing, K-Nearest Neighbor Algorithm, Deep Learning used in Recognition technique.

# Toshiba Kamruzzaman1, Mashrief Bin Zulfiquer , “Handwritten Bangla Digit Recognition using Capsule Network” , International Research Journal of Engineering and Technology (IRJET) , 2020.

# 

# A capsule is a set of neurons. It stores instantiating parameters of an object such as position, scale, angle of view, rotation deformation, velocity, albedo, hue, texture on in a high dimensional vector space. In this paper, we exploit CapsuleNet for handwritten Bangla numeral recognition. From experiments, we have achieved 99.91% recognition rate on the handwritten Bangla numerical data set NumtaDB which comprise more than 85,000 images of hand-written Bengali digit.

# In this paper, we have presented a CapsuleNet for recognition of handwritten Bangla digits. The Capsulenet gives excellent results on segmentation tasks and outperforms other models and was very lightweight. For the Bangla digits recognition, we have received 99.91% accuracy which is better than all the other CNN models. Variation was observed in the overall classification accuracy by altering the number of hidden layers and batch size. In future, compound digits will be evaluated and handwritten mathematical signs and equations will be analyzed.

# Mayank Singh1, Rahul2 , “Handwritten Digit Recognition using Machine Learning” , International Research Journal of Engineering and Technology (IRJET) , 2020

# People write in as many different ways as there are stars in a galaxy. This leads to development of different patterns in writing. Costly manual labour is required to do a mundane and tedious job of converting the physical written data and information into digital form for storing it in a digital form. This project discusses the solution to a part of problem as we have limited the scope to only the hand written digits (0-9). We have trained a model using deep neural networks for digit recognition using Google’s Machine Learning tool TensorFlow and Python Programming language. We have used the ‘MNIST DATABASE’ which consist of training and test set for hand written digits (0-9) of size (28x28) pixels i.e. 784 pixels. The data set consist of 60,000 training data and 10,000 test data. The limitation of this model will be if digits other than (0-9) are given then the model will not be able to recognize and classify it and the model will be able to predict numbers only in black and white images.

**CHAPTER 3**

**SYSTEM ANALYSIS**

**3.1.DRWABACKS OF EXISTING SYSTEM**

* These days, an ever-increasing number of individuals use pictures to transmit data. It is additionally main stream to separate critical data from pictures.
* Image Recognition is an imperative research area for its generally used applications.
* In general, the field of pattern recognition, one of the difficult undertakings is the precise computerized recognition of human handwriting.
* Without a doubt, this is a very difficult issue because there is an extensive diversity in handwriting from an individual to another individual.

**3.2.ADVANTAGES OF PROPOSED SYSTEM**

* We proposed new fresh approach to existing work for improvements of learning algorithm and novel feature extraction method for Handwritten Digit Recognition.
* Our study forces us to improve following parameters to be considered to improve the system.
* Decrease training time without affecting accuracy
* Implementing method and architecture uses GPU for parallel processing to speed up learning process

**3. 3. REQUIREMENTS SPECIFICATION**

**3. 3. 1. Hardware Requirements**

● **NVIDIA GeForce Graphics Card:**

NVIDIA GeForce Graphic Card Nvidia Corporation more regularly specified to as Nvidia (adapted as NVIDIA), is an American innovation organization. It structures graphical processing units (GPUs) for the gaming and expert markets, just as a framework, system on a chip units (SoCs) for the versatile figuring and car showcase. It’s essential GPU product is named as "GeForce". With outstanding GPU fabricating, Nvidia gives parallel processing capacities to analysts and researchers that enable them to effectively run superior applications.

● **Intel core i5 processor:**

Intel Core is a line of streamlined midrange consumer, workstation and enthusiast computer central processing units (CPUs) marketed by Intel Corporation. These processors displaced the existing mid- to high-end Pentium processors at the time of their introduction, moving the Pentium to the entry level. Identical or more capable versions of Core processors are also sold as Xeon processors for the server and workstation markets.

● Minimum 4MB RAM

● Minimum 4GB disk space 25

**3. 3. 2. Software Requirements**

● **Tensorflow:**

TensorFlow is an amazing information stream in machine learning library made by the Brain Team of Google and made open source in 2015.It is intended to ease the use and broadly relevant to both numeric and neural system issues just as different spaces. Fundamentally, TensorFlow is a lowlevel tool for doing entangled math and it targets specialists who recognize what they're doing to construct exploratory learning structures, to play around with them and to transform them into running programs. For the most, it can be considered as a programming framework in which one can entitle to calculations as graphs. Nodes in the graph speak the math activities, and the edges contain the multi-dimensional information clusters (tensors) related between them.

* **Python 3.7**

Python is broadly utilized universally and is a high-level programming language. It was primarily introduced for prominence on code, and its language structure enables software engineers to express ideas in fewer lines of code. Python is a programming language that gives you a chance to work rapidly and coordinate frameworks more effectively.

● **Microsoft Visual Studio:**

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a code profiler, designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that expand the functionality at almost every level—including adding support for source control systems (like Subversion and Git) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Azure DevOps client: Team Explorer).

* **Anaconda3 5.3.1**

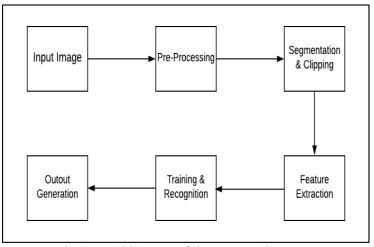
Anaconda is a free and open-source appropriation of the Python and R programming for logical figuring like information science, AI applications, large-scale information preparing, prescient investigation, and so forth. Anaconda accompanies in excess of 1,400 packages just as the Conda package and virtual environment director, called Anaconda Navigator, so it takes out the need to figure out how to introduce every library freely. Anaconda Navigator is a graphical UI (GUI) incorporated into Anaconda appropriation that enables clients to dispatch applications and oversee conda packages, conditions and channels without utilizing command line directions.

**CHAPTER 4**

**SYSTEM DESIGN**

**4.1 SYSTEM ARCHITECTURE**

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.

****

**FIGURE 4.1 ARCHITECTURE DIAGRAM**

The proposed model contains the four stages in order to classify and detect the digits:

A. Pre-processing

B. Segmentation

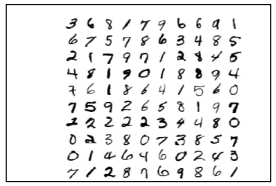
C. Feature Extraction

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 4.2 shows a sample of images taken from the MNIST database.



**FIGURE 4.2 SAMPLES FROM THE MNIST DATASET**

**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 following three classifiers:

• K-Nearest Neighbor

• Random Forest Classifier

• Support Vector Machine

* **K-Nearest Neighbor**

KNN is an instance based learning algorithm. There are two main benefits of using KNN algorithm, that is, it is robust to noisy training data and it is very efficient if the data is very large in size. To perform admirably, this algorithm requires a set of training datasets which includes perfectly labeled data points. KNN is also a non-parametric classifier. The algorithm considers new data point as its input and performs classification by calculating distance between new and labeled data points using the Euclidean or Hamming distance formulas. The Euclidean distance is calculated using the following formula:

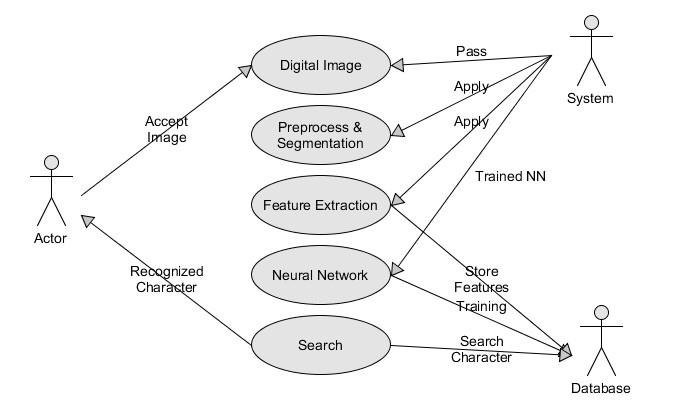
* **Random Forest Classifier**

RFC is a supervised learning method. It infers that there is an immediate connection between the total number of trees and the result it gets: the bigger the number of trees, the more precise the outcome will be. This classifier can be used for regression as well as classification. For RFC algorithm if there are sufficient trees then the classifier will not over fit the model, instead it avoids the over fitting issues. This classifier can deal with the missing quantities. Once the training is done, predictions are taken from each individual tree and the average is calculated using the following formula:

* **Support Vector Machine**

SVM is also a supervised learning method. It is also used for both classification and regression tasks. In this type of algorithm, there are data items which are considered as points in an n-dimensional space. This classifier finds the hyper plane by performing classifications between the two classes. One of the main advantages of this algorithm is that it provides a regularization parameter which avoids the over fitting problems. The block diagram shown below in the Figure 3 describes all these above steps.

**4.2 USE CASE DIAGRAM**

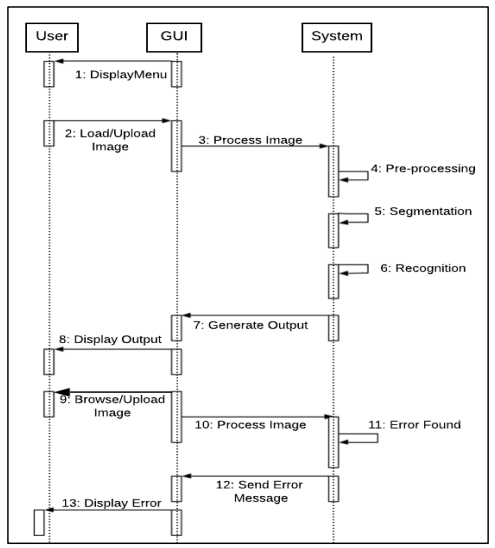
****

**FIGURE 4.3 USE CASE DIAGRAM**

A use case diagram is a representation of the user’s interaction with the system and depicting the specifications of the use case. It portrays the different types of users of a system and the various ways they interact with the system.

In Figure 4.3 the first process is accepting the digital image , then it is the preprocessing , segmentation , feature extraction and application of neural networks by the system. Finally the digit can be recognized.

**4.3 SEQUENCE DIAGRAM**

****

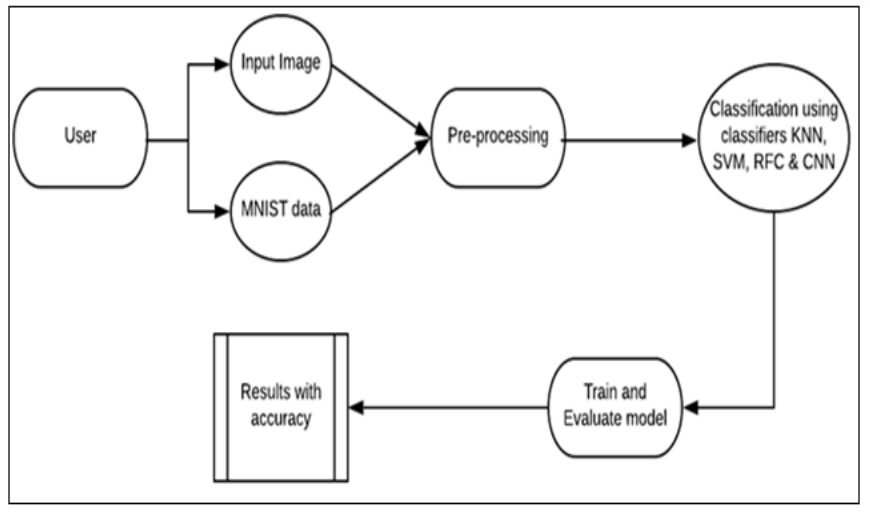
**FIGURE 4.4 SEQUENCE DIAGRAM**

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

The Figure 4.4 illustrates the sequence diagram of the system model. The figure describes the sequence of steps to be taken while performing execution. The CNN model works in the following sequence. User uploads a particular image of any digit which he wants to recognize. The image will be processed by the system. On running the system code the output is generated that shows which is the digit uploaded by the user and also displays the accuracy rate predicted by the model. On uploading image with different resolutions other than the one mentioned in the code, the output generated shows error, and displays an error message to the user.

**4.4 DATAFLOW DIAGRAM**

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically**.** The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow — there are no decision rules and no loops. The data-flow diagram is a tool that is part of structured analysis and data modeling. When using UML, the activity diagram typically takes over the role of the data-flow diagram. A special form of data-flow plan is a site-oriented data-flow plan.

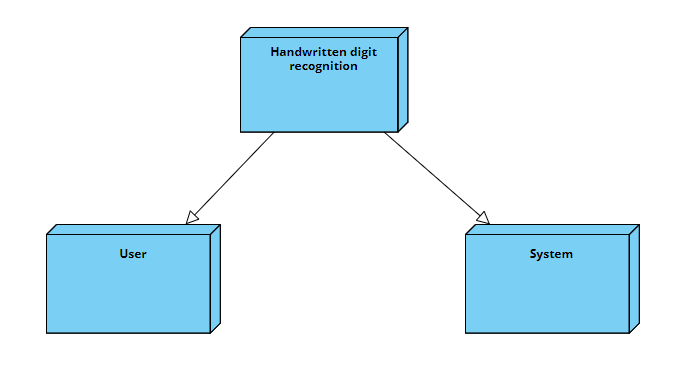
****

**FIGURE 4.5 DATAFLOW DIAGRAM**

The following Figure 4.5 describes the Data flow diagram of the proposed system model. There are two ways to provide input to the system. The user can either upload the image of the digit he wants to detect or the data from the MNIST dataset. The input images are pre-processed. Using the different classifiers the recognized digits’ accuracy is compared and the result is obtained. The results obtained are displayed along with the accuracy.

**4.5 DEPLOYMENT DIAGRAM**

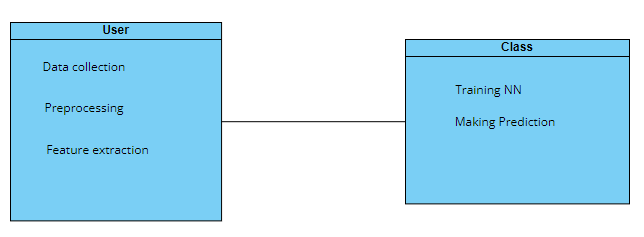
Deployment diagrams are used to visualize the topology of the physical components of a system where the software components are deployed and describe the static deployment view of a system.



**FIGURE 4.6 DEPLOYMENT DIAGRAM**

In Figure 4.6, the required data is gathered from Mnist dataset and all the pre-processing functions are performed to process all input images. Features are extracted from the data gathered from various news sources and only the required features are selected.

**4.6 CLASS DIAGRAM**

****

**FIGURE 4.7 CLASS DIAGRAM**

# The class diagram is a static diagram. It represents the static view of an

# application. Class diagram is not only used for visualizing, describing and

# documenting different aspects of a system but also for constructing executable code of the software application. In Figure 4.8, To process all input images , the data is first obtained, then all pre-processing functions are performed. In order to pick features ,we used feature extraction and selection methods.

# 4.7 COMPONENT DIAGRAM

# 

# FIGURE 4.8 COMPONENT DIAGRAM

# Component Diagrams does not describe the functionality of the system but it

# describes the components used to make those functionalities. Visualize the

# components of a system. In Figure 4.8, the first component is image collection here the required dataset is collected ,the next component is training the digit recognizer and there is finally the prediction of digits

# CHAPTER 5

## MODULE DESCRIPTION

## 5.1 MODULES

## The system consists of the following modules

## Data preprocessing

## Feature extraction

## Training model

## GUI creation

## 5.1.1 DATA PREPROCESSING

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So for this, we use data preprocessing task.The MNIST dataset consists training examples in the test set. It’s a good dataset for those who want to learn techniques and pattern recognition methods on real-world data without much effort in data-preprocessing.

**5.1.2 FEATURE EXTRACTION**

Feature extraction is a type of dimensionality reduction where a large number of pixels of the image are efficiently represented in such a way that interesting parts of the image are captured effectively. Feature extraction starts from an initial set of measured data and builds derived values (intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps, and in some cases leading to better human interpretations. Determining a subset of the initial features is called features.The selected features are expected to contain the relevant information from the input data, so that the desired task can be performed by using this reduced representation instead of the complete initial data.

**5.1.3 TRAINING MODEL**

A training model is a dataset that is used to train an ML algorithm. It consists of the sample output data and the corresponding sets of input data that have an influence on the output. The training model is used to run the input data through the algorithm to correlate the processed output against the sample output. The result from this correlation is used to modify the model.

## 5.1.4 GUI CREATION

To build an interactive window we have created the GUI. A GUI is nothing but a Guided User Interface . In this , you can draw digits on canvas, you can add image or reset an image depending on the digit drawn and have various buttons for different operations, you can identify the digit once you click add image. The Tkinter library is the part of Python standard library. Our method takes the picture as input and then activates the trained model to predict the digit.

After that to build the GUI for our app we have created the App class. In GUI canvas you will have a small window where you can draw a digit by capturing the mouse event and with a button click it shows the results. The accuracy may differ upon the different flavors in drawing.

**CHAPTER 6**

**CONCLUSION AND FUTURE ENHANCEMENTS**

**CONCLUSION**

The main objective of this paper is to find a representation of isolated handwritten digits that allow their effective recognition. In this paper used different algorithms for recognition of handwritten numerals. In any recognition process, the important problem is to address the feature extraction and correct classification approaches. The proposed algorithm tries to address both the factors and well in terms of accuracy and time complexity. The overall highest accuracy 90.37% is achieved in the recognition process by Multilayer Perception. This work is carried out as an initial attempt, and the aim of the paper is to facilitate for recognition of handwritten numeral without using any standard classification techniques.

**FUTURE ENHANCEMENTS**

Some of the future works possible to implement by CNN’s are compressing or obtaining same results from smaller networks by optimization tricks , more invariant feature learning such that the input images dosen’t gets distorted. The major 3D vision networks is a scope for researches to develop using LeNet architecture and more biologically concordant methods , a hope for future is that Unsupervised CNN’s .

**APPENDIX I**

**SAMPLE CODE**

**#**Data preparation and preprocessing

from keras.datasets import mnist

import keras.backend as K

from keras.models import Sequential

from keras.layers import Dense

from keras.layers import Conv2D, MaxPool2D, Flatten, Activation

from keras.utils.np\_utils import to\_categorical

from keras.optimizers import Adam

import pandas as pd

import numpy as np

from matplotlib import pyplot as plt

%matplotlib inline

%config InlineBackend.figure\_format='retina'

#training model

(X\_train, y\_train), (X\_test, y\_test) = mnist.load\_data()

X\_train = X\_train.reshape(-1, 28, 28, 1)

X\_test = X\_test.reshape(-1, 28, 28, 1)

y\_train\_cat = to\_categorical(y\_train)

y\_test\_cat = to\_categorical(y\_test)

X\_train = X\_train.astype('float32')

X\_test = X\_test.astype('float32')

X\_train /= 255.0

X\_test /= 255.0

model = Sequential()

model.add(Conv2D(32, (3, 3), input\_shape=(28, 28, 1)))

model.add(MaxPool2D(pool\_size=(2, 2)))

model.add(Activation('relu'))

model.add(Flatten())

model.add(Dense(128, activation='relu'))

model.add(Dense(10, activation='softmax'))

model.compile(loss='categorical\_crossentropy', optimizer=Adam(), metrics=['accuracy'])

model.fit(X\_train, y\_train\_cat, batch\_size=32, epochs=2, verbose=1, validation\_split=0.3)

model.evaluate(X\_test, y\_test\_cat)

from numpy import argmax

from tkinter import \*

import tkinter as tk

import math

from PIL import Image, ImageDraw

#GUI creation

white = (255, 255, 255)

black = (0, 0, 0)

window = Tk()

window.title("Handwriting Calculator")

window.geometry('500x500')

lbl = Label(window, text="Write digits with your mouse in the gray square",font=('Arial Bold',15))

lbl.grid(column=3, row=0)

canvas\_width = 120

canvas\_height = 120

image1 = Image.new("RGB", (canvas\_width, canvas\_height),white)

draw = ImageDraw.Draw(image1)

counter=0

xpoints=[]

ypoints=[]

x2points=[]

y2points=[]

global predictions

predictions = []

number1 = []

digits=0

def paint( event ):

x1, y1 = ( event.x - 4 ), ( event.y - 4 )

x2, y2 = ( event.x + 4 ), ( event.y + 4 )

w.create\_oval( x1, y1, x2, y2, fill = 'black' )

xpoints.append(x1)

ypoints.append(y1)

x2points.append(x2)

y2points.append(y2)

def imagen ():

global counter

global xpoints

global ypoints

global x2points

global y2points

counter=counter+1

image1 = Image.new("RGB", (canvas\_width, canvas\_height),black)

draw = ImageDraw.Draw(image1)

elementos=len(xpoints)

for p in range (elementos):

x=xpoints[p]

y=ypoints[p]

x2=x2points[p]

y2=y2points[p]

draw.ellipse((x,y,x2,y2),'white')

w.create\_oval( x-4, y-4, x2+4, y2+4,outline='gray85', fill = 'gray85' )

size=(28,28)

image1 = image1.resize(size)

image1 = image1.convert('L')

image1 = np.array(image1)

image1 = image1.reshape(-1, 28, 28, 1)

image1 = image1.astype('float32')

image1 /= 255.0

predictions.append(argmax(model.predict(image1)))

lbl2 = Label(window, text=predictions[counter-1],font=('Arial Bold',20))

lbl2.grid(column=3, row=10)

xpoints=[]

ypoints=[]

x2points=[]

y2points=[]

w = Canvas(window,

width=canvas\_width,

height=canvas\_height,bg='gray85')

w.grid(column=3,row=2)

def delete ():

global counter

counter = counter-1

del predictions[counter]

w1 = Canvas(window,

width=200,

height=20,bg='gray95')

w1.grid(column=3,row=10)

def add():

global operation

global counter

global digits

digits=counter

operation = 'add'

def subtract():

global operation

global counter

global digits

digits=counter

operation = 'subtract'

def multiply():

global operation

global counter

global digits

digits=counter

operation = 'multiply'

def divide():

global operation

global counter

global digits

digits=counter

operation = 'divide'

def equals():

digitone=''

digittwo=''

global digits

global predictions

global counter

digitstotal=len(predictions)

for x in range(digits):

digitone=digitone+str(predictions[x])

predictions[0]=int(digitone)

for x in range(digits,digitstotal):

digittwo=digittwo+str(predictions[x])

predictions[1]=int(digittwo)

if operation == 'add':

answer = predictions[0]+predictions[1]

if operation == 'subtract':

answer = predictions[0]-predictions[1]

if operation == 'multiply':

answer = predictions[0]\*predictions[1]

if operation == 'divide':

answer = predictions[0]/predictions[1]

lbl2 = Label(window, text=answer,font=('Arial Bold',20))

lbl2.grid(column=3, row=10)

predictions=[]

counter=0

def reset():

global predictions

global counter

predictions=[]

counter=0

w1 = Canvas(window,

width=200,

height=20,bg='gray95')

w1.grid(column=3,row=10)

w1 = Canvas(window, width=200, height=20,bg='gray95')

w1.grid(column=3,row=10)

w.bind( "<B1-Motion>", paint )

button = tk.Button(window, text='Save image', width=25, command=imagen)

button.grid(column=3,row=3)

button2 = tk.Button(window, text='Add', width=25, command=add)

button2.grid(column=3,row=5)

button3 = tk.Button(window, text='Subtract', width=25, command=subtract)

button3.grid(column=3,row=6)

button4 = tk.Button(window, text='Multiply', width=25, command=multiply)

button4.grid(column=3,row=7)

button5 = tk.Button(window, text='Divide', width=25, command=divide)

button5.grid(column=3,row=8)

button6 = tk.Button(window, text='=', width=25, command=equals)

button6.grid(column=3,row=9)

button6 = tk.Button(window, text='Click here if the number is not correct', width=35, command=delete)

button6.grid(column=3,row=12)

button7 = tk.Button(window, text='Reset', width=25, command=reset)

button7.grid(column=3,row=13)

window.mainloop()

The main objective of this investigation is to find

a representation of isolated handwritten digits that allow

their effective recognition. In this paper used different

machine learning algorithm for recognition of

handwritten numerals. In any recognition process, the

important problem is to address the feature extraction

and correct classification approaches. The proposed

algorithm tries to address both the factors and well in

terms of accuracy and time complexity. The overall

highest accuracy 90.37% is achieved in the recognition

process by Multilayer Perceptron. This work is carried

out as an initial attempt, and the aim of the paper is to

facilitate for recognition of handwritten numeral without

using any standard classification techniqu

The main objective of this investigation is to find

a representation of isolated handwritten digits that allow

their effective recognition. In this paper used different

machine learning algorithm for recognition of

handwritten numerals. In any recognition process, the

important problem is to address the feature extraction

and correct classification approaches. The proposed

algorithm tries to address both the factors and well in

terms of accuracy and time complexity. The overall

highest accuracy 90.37% is achieved in the recognition

process by Multilayer Perceptron. This work is carried

out as an initial attempt, and the aim of the paper is to

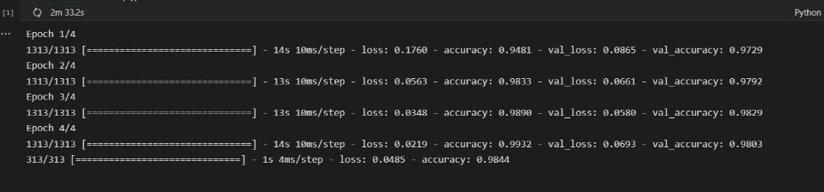
facilitate for recognition of handwritten numeral without

using any standard classification techniq

**APPENDIX II**

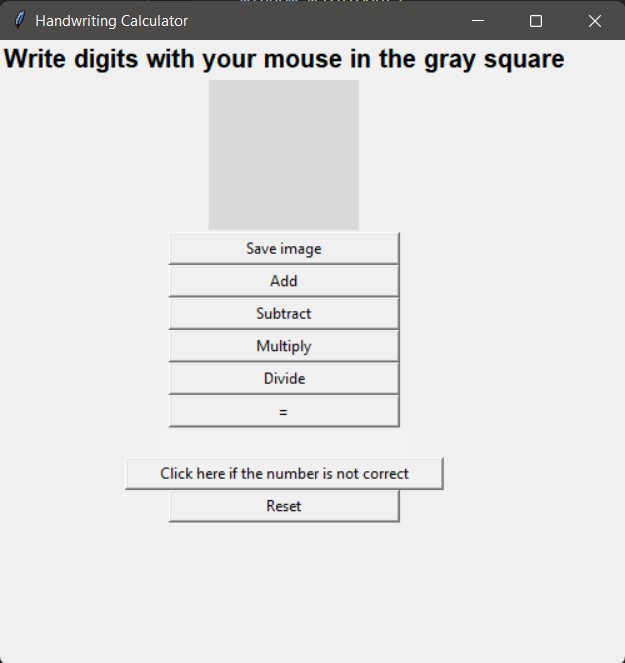
**SCREENSHOTS**

**Data Preprocessing**

****

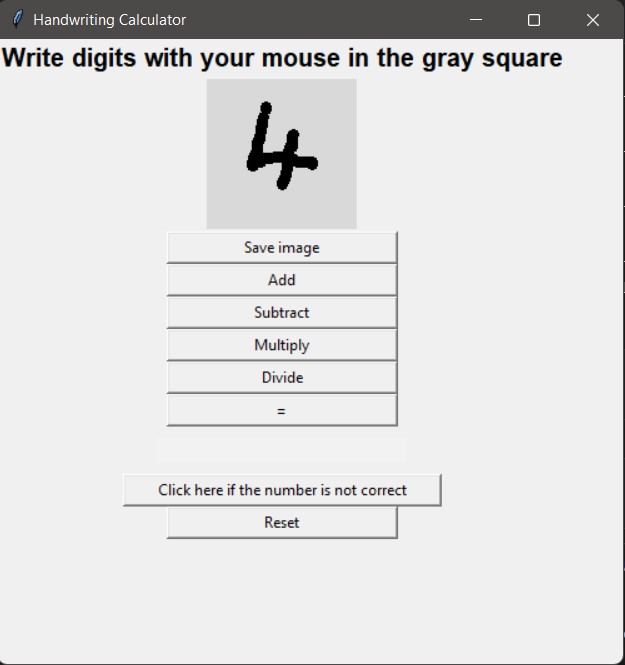
**Figure II.1**

**GUI**

****

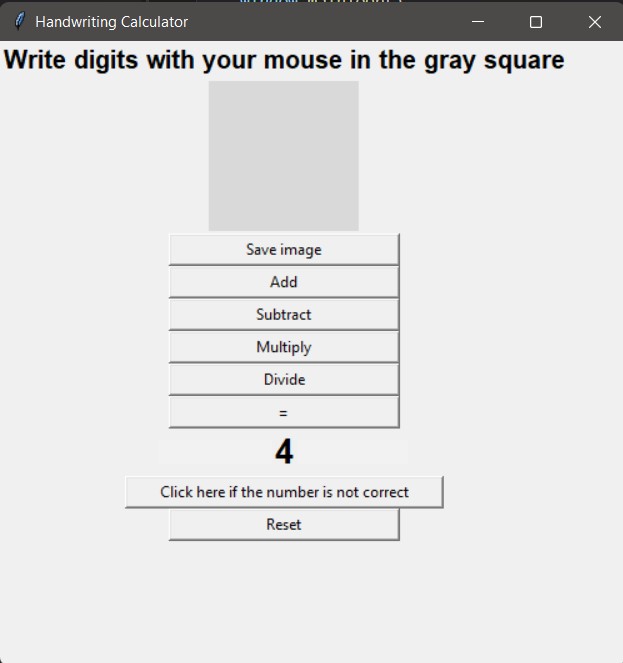
**Figure II.2**

**Drawing Number 4**



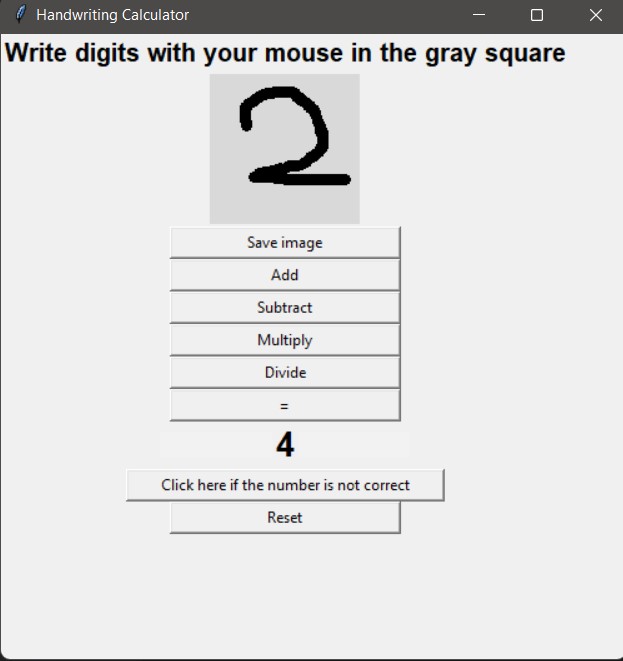
**Fig II.3**

**Saving The Image**

****

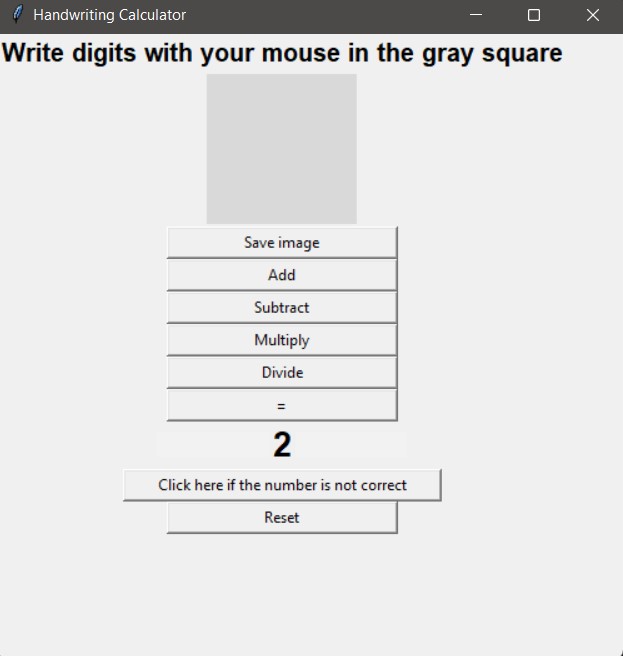
**Fig II.4**

**Drawing Number 2**

****

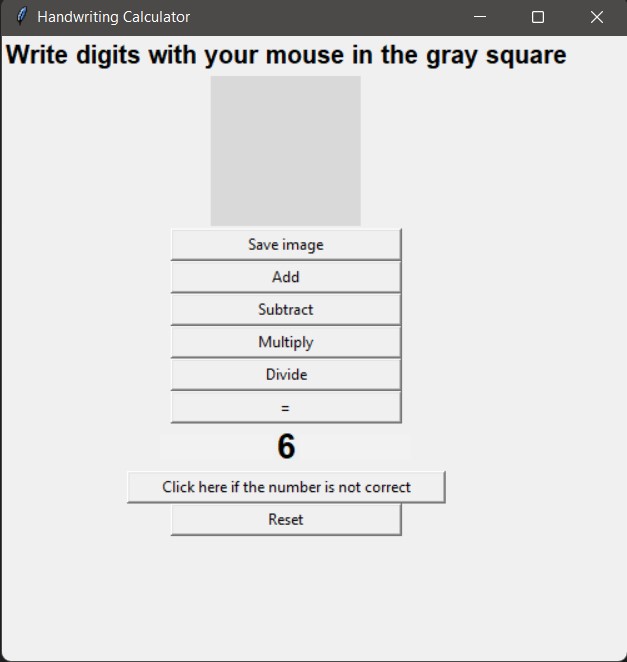
**Fig II.5**

**Saving The Image**

****

**Fig II.6**

**Result Of Equation**

****

**Fig II.7**

**REFERENCES**

1. Ritik Dixit, Rishika Kushwah, Samay Pashine , “Handwritten Digit Recognition using Machine and Deep Learning Algorithms” , International Journal of Computer Applications (ARXIV) , 2021.
2. S M Shamim, Mohammad Badrul Alam Miah, Angona Sarker, Masud Rana & Abdullah Al Jobair , “Handwritten Digit Recognition using Machine Learning Algorithms” , Global Journal of Computer Science and Technology( GJCST ) , 2018.
3. [Mahmoud M. Abu Ghosh](https://ieeexplore.ieee.org/author/37085882676), [Ashraf Y. Maghari](https://ieeexplore.ieee.org/author/37085875507) , “A Comparative Study on Handwriting Digit Recognition Using Neural Networks” , International Conference on Promising Electronic Technologies (ICPET) , 2017.
4. Jinze Li,Gongbo Sun, Leiye Yi, Qian Cao, Fusen Liang, Yu Sun , “Handwritten Digit Recognition System Based on Convolutional Neural Network” , IEEE International Conference on Advances in Electrical Engineering and Computer Applications ( AEECA) , 2020.
5. S. Knerr, L. Personnaz, G. Dreyfus , “Handwritten digit recognition by neural networks with single-layer training” , IEEE Transactions on Neural Networks (IEEE), 1992.
6. Md. Anwar Hossain & Md. Mohon Ali , “Recognition of Handwritten Digit using Convolutional Neural Network (CNN)” , Global Journal of Computer Science and Technology (GJCST) , 2019.
7. Vijayalaxmi R Rudraswamimath, Bhavanishankar K , “Handwritten Digit Recognition using CNN” , International Journal of Innovative Science and Research Technology (IJISRT) , 2019.
8. Yevhen Chychkarova , Anastasiia Serhiienkob , Iryna Syrmamiikha , AnatoliiKarginc,“Handwritten Digits Recognition Using SVM, KNN, RF and Deep Learning Neural Networks” , (CEUR-WS) , 2021.
9. Farhana Sultana, Abu Sufian, Paramartha Dutta , “Advancements in Image Classification using Convolutional Neural Network” , (IEEE) 2019.
10. Pranit Patil, Bhupinder Kaur , “Handwritten Digit Recognition Using Various Machine Learning Algorithms and Models” , International Journal of Innovative Research in Computer Science & Technology (IJIRCST) , 2020.
11. Toshiba Kamruzzaman , Mashrief Bin Zulfiquer , “Handwritten Bangla Digit Recognition using Capsule Network” , International Research Journal of Engineering and Technology (IRJET) , 2020.
12. Mayank Singh1, Rahul2 , “Handwritten Digit Recognition using Machine Learning” , International Research Journal of Engineering and Technology (IRJET) , 2020