ACKNOWLEDGEMENT

We express our sincere thanks to Karpagam educational and charitable trust for providing necessary facilities to bring out the project successfully. We felt great to record our thanks to the chairman **Dr.R.VASANTHAKUMAR, B.E.,(Hons),D.Sc.**for all his support and ray of strengthening hope extended.

It is a moment of immense pride for us to reveal our profound thanks to our respected principal, **Dr.P.VIJAYAKUMAR, M.E., Ph.D.** who happens to be a striving force in all our endeavors.

We express our sincere thanks to our **Mr. Manikandan Sundaram M. Tech, (Ph. D)** Head of the Department of Artificial Intelligence and Data Science for providing an opportunity to work on this project. His valuable suggestions helped us a lot to do this project.

A word of thanks would not be sufficient for the work of our project guide **Mr. Surenther. I M.E (Ph. D)** Department of Artificial Intelligence and Data Science whose efforts and inspiration lead us through every trying circumstance.

We would also like to recollect the courage and enthusiasm that was inculcated in us by our project coordinator, **Mr. Surenther. I M.E (Ph. D)** Department of Artificial Intelligence and Data Science for valuable guidance and support through the tenure of our project.

We deeply express our gratitude to all the members of the faculty of the department of Artificial Intelligence and Data Science for the encouragement, which we received throughout the semester.

ABSTRACT

Humans' reliance on technology has never been higher, to the point that deep learning and machine learning algorithms can conduct everything from object detection in images to adding sound to silent movies.Similarly, handwritten text recognition is a major area of research and development with a plethora of possibilities.The capacity to design an efficient algorithm that can recognise handwritten digits and which is submitted by users via a scanner, tablet, and other digital devices is at the heart of the challenge.

A method for recognizing handwritten digits off-line using various machine learning techniques The major goal of this study is to ensure that methods for recognizing handwritten digits are both effective and trustworthy.Handwriting recognition (HTR) is the ability of a computer to accept and understand comprehensible handwriting input from sources such as paper documents, pictures, touch displays, and other devices.Apparently, we used Support Vector Machines (SVM) models to achieve handwritten digit recognition using MNIST datasets in this research.

Keywords: pattern recognition, handwritten recognition, digit recognition, machine learning, WEKA, off-line handwritten recognition, machine learning algorithm, neural network, classification algorithm.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

SVM	Support Vector Machine
HTR	Handwriting recognition
CNN	Convolutional Neural Network
MNIST	Modified National Institute of Standards and Technology database

I . INTRODUCTION

Handwritten digit recognition refers to a computer's capacity to recognise human handwritten digits from a variety of sources, such as photographs, papers, and touch displays, and classify them into ten specified categories (0-9).In the realm of deep learning, this has been a topic of inexhaustible investigation.Numerous applications exist for digit recognition, including number plate identification, postal mail sorting, and bank check processing.

Because handwritten digit recognition is not an optical character recognition, we encounter numerous obstacles due to the various styles of writing used by different people.This study compares and contrasts various machine learning and deep learning methods for handwritten digit recognition.We employed Support Vector Machines, Multilayer Perceptrons, and Convolutional Neural Networks to achieve this.

The accuracy of the 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 real-world 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.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.

Hence, robust feature extraction is very important to improve the performance of a handwritten character recognition system. Nowadays handwritten digit recognition has obtained a lot of concentration in the area of pattern recognition systems, showing its application in diverse fields. In the next few days, the character recognition system might serve as a cornerstone to initiate paperless surroundings by digitizing and processing existing paper documents.

Handwritten digit dataset are vague in nature because there may not always be sharp and perfectly straight lines. The main goal in digit recognition is feature extraction is to remove the redundancy from the data and gain a more effective embodiment of the word image through a set of numerical attributes. It deals with extracting most of the essential information from raw data.

In addition the curves are not necessarily smooth like the printed characters. Furthermore,characters dataset can be drawn in different sizes and the orientation which are always supposed to be written on a guideline in an upright or downright point. Accordingly, an efficient handwritten recognition system can be developed by considering these limitations. It is quite exhausting that sometimes to identify handwritten characters as it can be seen that most of the human beings can’t even recognize their own written scripts. Hence, there exists constraint for a writer to write apparently for recognition of handwritten documents

Before revealing the method used in conducting this research, a software engineering module is first presented. Pattern recognition along with Image processing plays a compelling role in the area of handwritten character recognition. The study describes numerous types of classification of feature extraction techniques like structural feature based methods, statistical feature based methods and global transformation techniques. Statistical approaches are established on planning of how data are selected. It utilizes the information of the statistical distribution of pixels in the image.

With multiple state-of-the-art techniques, feature representations, and datasets, the study covers a comprehensive criterion of handwritten digit recognition.However, the relationship between training set size and accuracy/error, as well as the trained models' dataset independence, is investigated.The paper presents convolution neural networks into the handwritten digit recognition research and describes a system which can still be considered state of the art.

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 SVM 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.

II. RELATED WORK

Handwriting recognition of digits started around the 1980s. The task of handwritten digit recognition, using a classifier,has great significance and it has many applications. Handwritten digits are not similar as they differ in size, thickness, position relative to the margin. These are some of the difficulties we faced while trying to solve this problem. Y. LeCun et al. presented an application of back-propagation networks to handwritten digit recognition

Existing System

X. Han et al summarizes the latest development of CNN and expounds the relative research of image recognition technology and elaborates on the application of CNN in handwritten numeral recognition. However, every neural network has some error rate due to parallel in digit shape. R. Sudhakar et al developed a hybrid model by integrating a non-linear regression model and optimization-driven deep learner for video super resolution. Initially, the low-resolution frames are subjected to framing, and each frame is provided to both Fractional-Group Search Optimizer-based Deep Belief Network (FrGSO-DBN) classifier and the non-linear regression model.

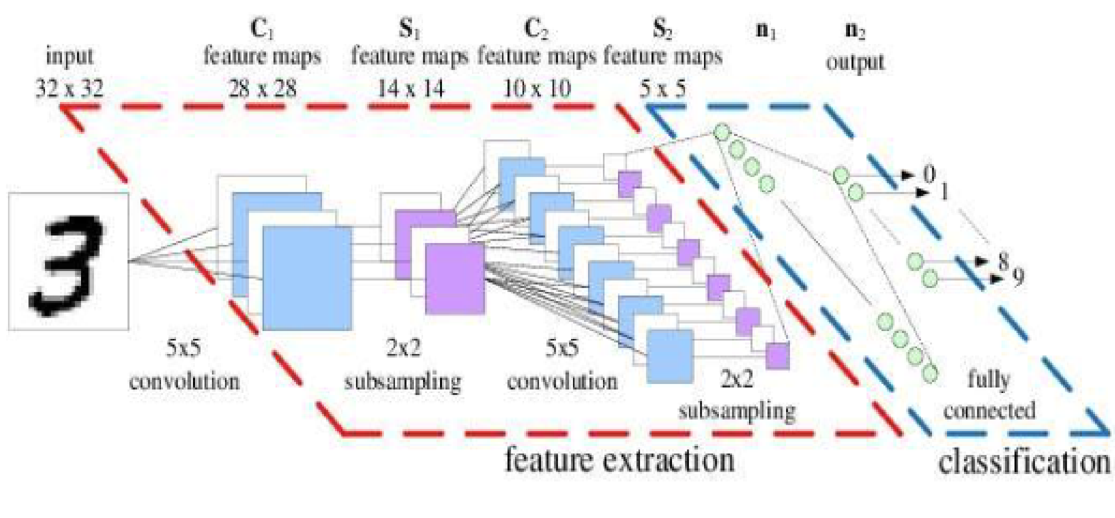

Fig 2.1:Existing system of HDR

ARCHITECTURE:

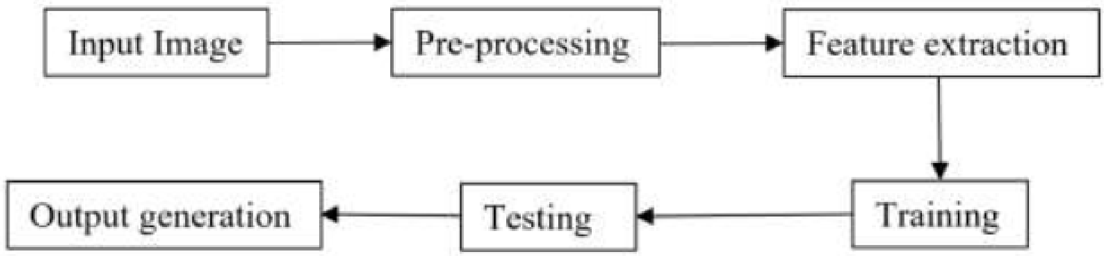

Fig 2.2: Architecture

Feature Extraction:

Different types of algorithms used for feature extraction have different types of error rate. The errors made by each separate algorithm does not overlap, so combining all these methods lead to a perfect recognition rate and also helps to reject the ambiguous digits recognition and improve the recognition rate of misclassified digits that can be recognized by humans

With the humanization of machines, a significant amount of research and development activity has sparked interest in deep learning, machine learning, and artificial intelligence. Machines are becoming increasingly smart throughout time, and they have made our lives more secure and manageable, from fundamental math to retina recognition.

Similarly, handwritten text recognition is an important application of deep learning and machine learning that aids in the detection of forgeries, and a wide range of research has already been done that includes a comprehensive study and implementation of various popular algorithms, such as works done by S M Shamim, Anuj Dutt, Norhidayu binti, and Hongkai Wang to compare the different models of CNN with the fundamental machine learning algorithms on various grounds.

Classification and Recognition:

Concluded that the Multilayer Perceptron classifier gave the most accurate results with minimum error rate followed by Support Vector Machine, Random Forest Algorithm, Bayes Net, Naıve Bayes, j48, and Random Tree respectively while presented a comparison between SVM, CNN, KNN, RFC and were able to achieve the highest accuracy of 98.72% using CNN (which took maximum execution time) and lowest accuracy using RFC. Did the detailed study-comparison on SVM, KNN and MLP models to classify the handwritten text and concluded that KNN and SVM predict all the classes of dataset correctly with 99.26% accuracy but the thing process goes little complicated with MLP when it was having trouble classifying number 9, for which the authors suggested to use CNN with Keras to improve the classification.

While has focused on comparing deep learning methods with machine learning methods and comparing their characteristics to know which is better for classifying mediastinal lymph node metastasis of non-small cell lung cancer from 18 F-FDG PET/CT images and also to compare the discriminative power of the recently popular PET/CT texture features with the widely used diagnostic features.

It concluded that the performance of CNN is not significantly different from the best classical methods and human doctors for classifying mediastinal lymph node metastasis of NSCLC from PET/CT images. However, CNN does not make use of the import diagnostic features, which have been proved more discriminative than the texture features for classifying small sized lymph nodes. Therefore, incorporating the diagnostic features into CNN is a promising direction for future research.

All we need is lots of data and information and we will be able to train a big neural net to do what we want, so a convolution can be understood as ”looking at functions surrounding to make a precise prognosis of its outcome., has used a convolution neural network for handwritten digit recognition using MNIST datasets has used 7 layered CNN model with 5 hidden layers along with gradient descent and back prorogation model to find and compare the accuracy on different epochs, thereby getting maximum accuracy of 99.2% , they have briefly discussed different components of CNN, its advancement from LeNet-5 to SENet and comparisons between different model like AlexNet, DenseNet and ResNet.

The research outputs the LeNet-5 and LeNet-5 (with distortion) achieved test error rate of 0.95% and 0.8% respectively on MNIST data set, the architecture and accuracy rate of AlexNet is same as LeNet-5 but much bigger with around 4096000 parameters and ”Squeeze-and-Excitation network” (SENet) have become the winner of ILSVRC-2017 since they have reduced the top-5 error rate to 2.25% and by far the most sophisticated model of CNN in existence.

Using the fit() method, a model can be trained. In order to see the skill of the trained model, test data is used as a validation dataset. Finally, to evaluate a model, the test dataset is used. Training is less complex because each module is designed to handle a specific subproblem. It is expected that each module can tackle the specific problem more efficiently and accurately because each module is trained independently which is easy to add and delete modules.

III. METHODOLOGY

DATASET

Handwritten character recognition is a broad study subject with numerous implementation options, including large learning datasets, popular algorithms, feature scaling, and feature extraction approaches.The MNIST dataset (Modified National Institute of Standards and Technology database) is a subset of the NIST dataset, which is made up of two different NIST databases: Special Database 1 and Special Database 3.The digits in Special Database 1 and Special Database 3 were written by high school students and US Census Bureau personnel, respectively.MNIST contains a total of 70,000 handwritten digit images (60,000 - training set and 10,000 - test set) in 28x28 pixel bounding box and anti-aliased. All these images have corresponding Y values which apprises what the digit is.

The handwritten digit recognition is an extensive research topic which gives a comprehensive survey of the area including major feature sets, learning datasets, and algorithms . Contrary to optical character recognition which focuses on recognition of machine- printed output, where special fonts can be used and the variability between characters along with the same size,font, and font attributes are fairly small.

In the performance of an offline character recognition system, the feature extraction and classification technique are critical.For character recognition systems, various feature extraction algorithms have been proposed.Handwritten numeral recognition difficulties have been investigated using approaches such as Dynamic programming, neural networks, knowledge systems, and combinations of these techniques.In many languages, such as English, Chinese, Japanese, and Arabic, more extensive work has been done on digit recognition.

We used a digit dataset provided by the Austrian Research Institute for Artificial Intelligence in our investigation.According to this data set, unconstrained scaling and a blur value of 2.5 for the Mitchell down-sampling filter should work well when down-sampling to 16x16 pixels.

SAMPLE OF DATASETS:

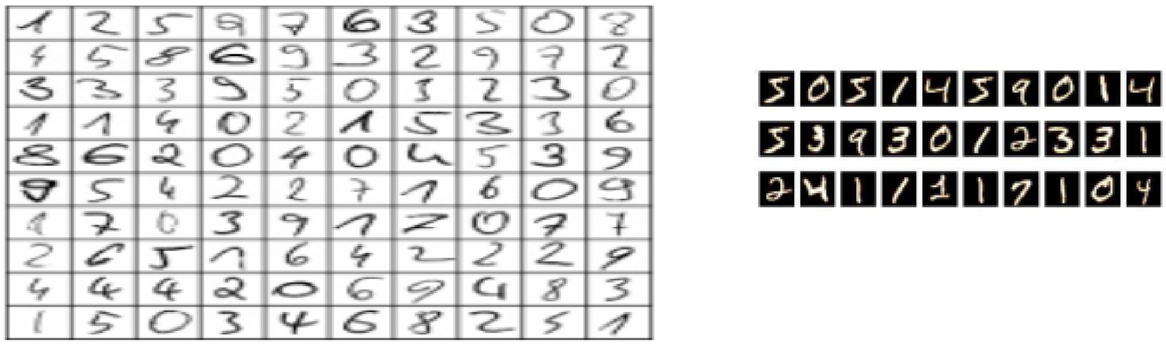

Fig 3.1

SUPPORT VECTOR MACHINE

The Support Vector Machine, or SVM, is a popular Supervised Learning technique that may be used to solve both classification and regression issues.However, it is mostly utilized in Machine Learning for Classification difficulties.

The SVM algorithm's purpose is to find the optimum line or decision boundary for categorizing n-dimensional space into classes so that additional data points can be readily placed in the correct category in the future.A hyperplane is the name for the optimal choice boundary.

The extreme points/vectors that assist create the hyperplane are chosen via SVM.Support vectors represent the extreme examples, which is why the technique is called Support Vector Machine.Consider the diagram below, which shows how a decision boundary or hyperplane is used to classify two different categories:

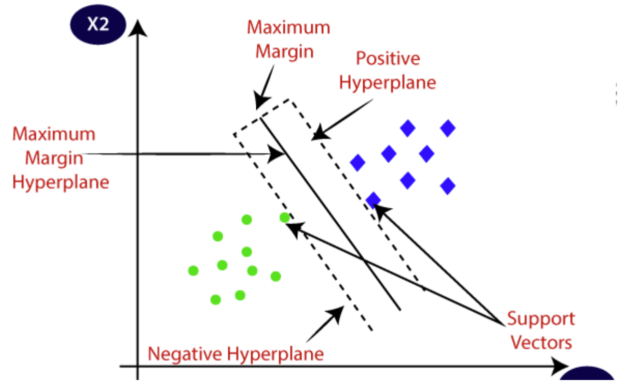

Fig 3.2

The example we used in the KNN classifier can help you understand SVM.If we observe an unusual cat that also has some dog-like characteristics, we can use the SVM method to develop a model that can properly identify whether it is a cat or a dog.We'll first train our model with a large number of photographs of cats and dogs so that it can learn about their many characteristics, and then we'll put it to the test with this weird creature.As a result, the extreme case of cat and dog will be shown because the support vector forms a decision boundary between these two data (cat and dog) and chooses extreme cases (support vectors).

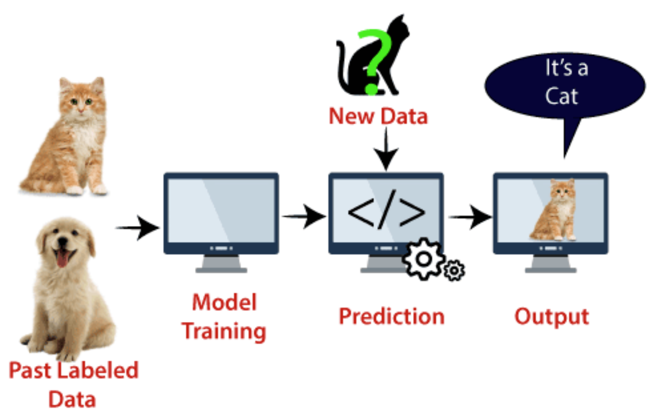

Fig 3.3

SVM algorithms can be used for **Face detection, image classification, text categorization,** etc.

Types of Support Vector Machine:

Linear SVM:

Linear SVM is a classifier that is used for linearly separable data, which implies that if a dataset can be classified into two classes using a single straight line, it is called linearly separable data, and the classifier is called Linear SVM classifier.

Non-linear SVM:

Non-Linear SVM is used to classify non-linearly separated data, which implies that if a dataset cannot be classified using a straight line, it is classified as non-linear data, and the classifier employed is the Non-Linear SVM classifier.

Hyperplane and Support Vectors in the SVM algorithm:

There can be multiple lines/decision boundaries to segregate the classes in n-dimensional space, but we need to find out the best decision boundary that helps to classify the data points. This best boundary is known as the hyperplane of SVM.

The dimensions of the hyperplane depend on the features present in the dataset, which means if there are 2 features (as shown in image), then the hyperplane will be a straight line. And if there are 3 features, then the hyperplane will be a 2-dimension plane.We always create a hyperplane that has a maximum margin, which means the maximum distance between the data points.

How does the Support Vector machine work?

Linear SVM:

The working of the SVM algorithm can be understood by using an example. Suppose we have a dataset that has two tags (green and blue), and the dataset has two features x1 and x2. We want a classifier that can classify the pair(x1, x2) of coordinates in either green or blue. Consider the below image.

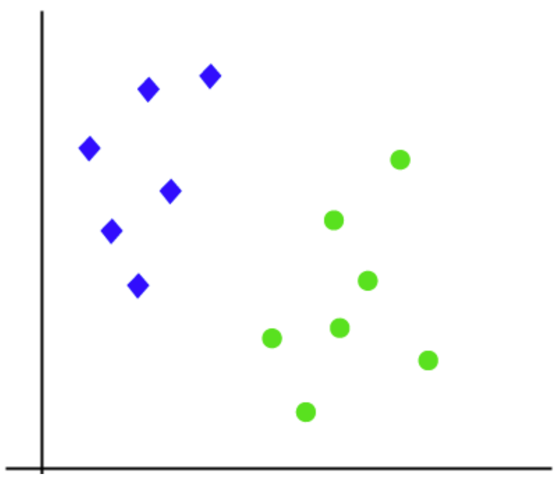

Fig 3.4

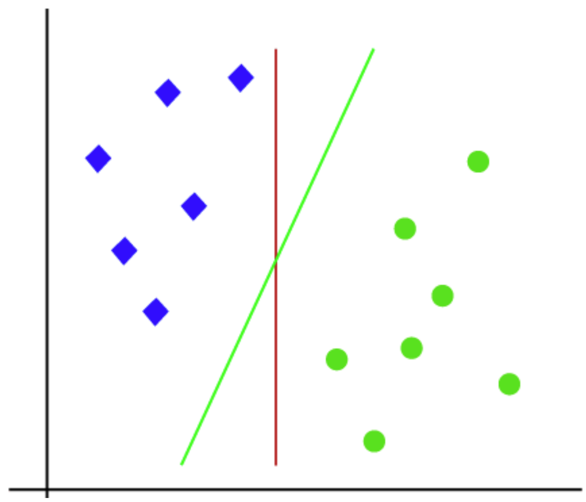

Fig 3.5

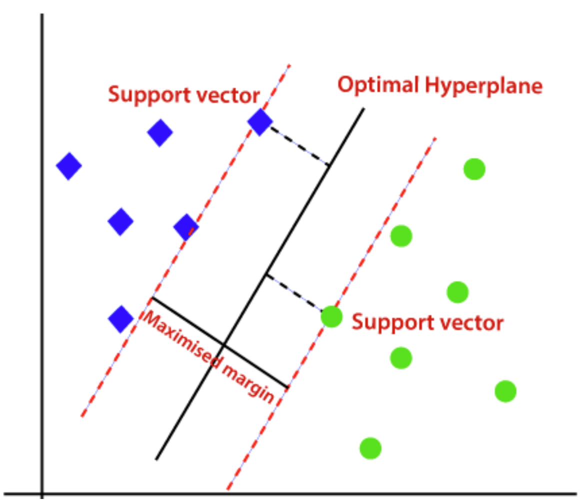

Fig 3.6

So as it is 2-d space so by just using a straight line, we can easily separate these two classes. But there can be multiple lines that can separate these classes. Consider the below image.

Hence, the SVM algorithm helps to find the best line or decision boundary; this best boundary or region is called a hyperplane. SVM algorithm finds the closest point of the lines from both the classes. These points are called support vectors. The distance between the vectors and the hyperplane is called the margin. And the goal of SVM is to maximize this margin. The hyperplane with maximum margin is called the optimal hyperplane.

Non-Linear SVM:

If data is linearly arranged, then we can separate it by using a straight line, but for non-linear data, we cannot draw a single straight line. Consider the below image.So to separate these data points, we need to add one more dimension. For linear data, we have used two dimensions x and y, so for non-linear data, we will add a third dimension z. It can be calculated as $z=x^2 +y^2$

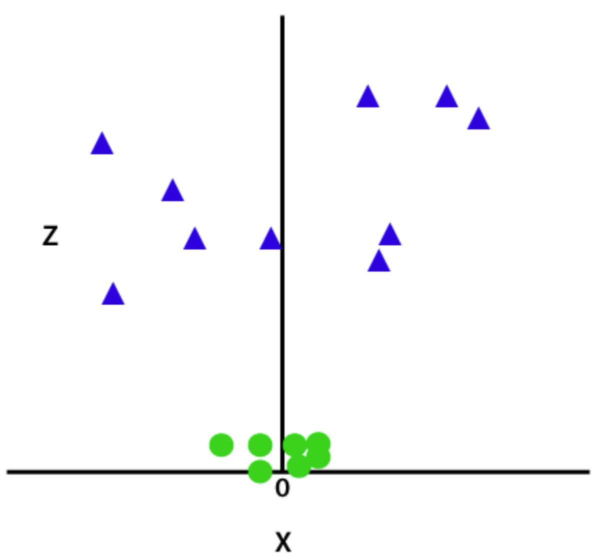

Fig 3.7

By adding the third dimension, the sample space will become as like this:

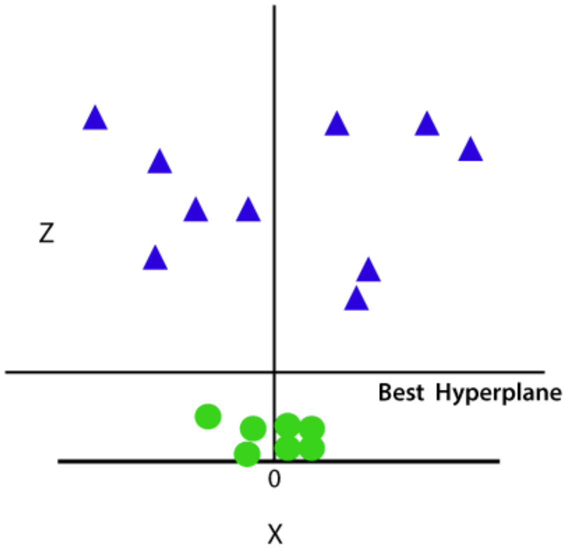

Fig 3.8
So now, SVM will divide the datasets into classes in the following way. Consider the be Image.

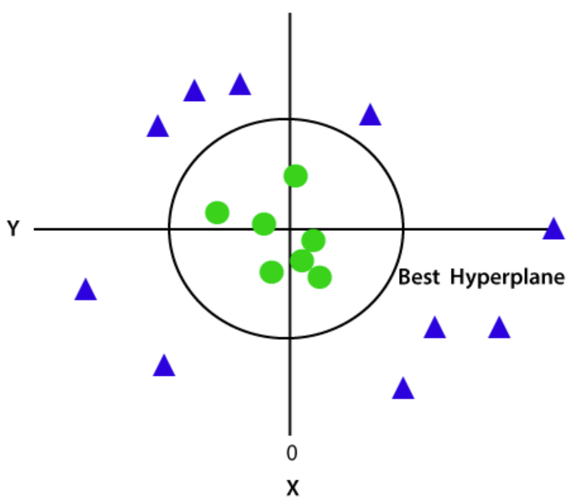

Fig 3.9
Since we are in 3-d Space, hence it is looking like a plane parallel to the x-axis. If we convert it in 2d space with z=1, then it will become like this.

NEED OF THE SOFTWARE:

The total project lies with a great computation speed and by an online server where run and compilation is done quickly. All the packages were imported that were needed for the software online. We need the tools to be imported also.

This project at first is in need of the software of python.The total code is written in python so it needs Python3. Python2 was not chosen because python3 has some additional upgrade over python2. The packages have been imported and the algorithm created which is done by installing the new packages from online in python3.

Apart from that the total project is online compiled or ran and done by the software provided by the Google Colab free version.Apart from choosing Anaconda Navigator, Spyder or Jupyter Notebook, Google Colab or Colabotory have been chosen because this provide more speed and accurate compilation as is known. Creation of the machine learning algorithms deals with data and bigger size programs.

This project deals with the End users with some functionalities.The major functionality can be the Recognition of digit.User can write a handwritten digit and this project will recognise it accurately. Edge detection can be set in the process of image processing.ML algorithm can differentiate the various digits from another by recognising it.

The better functionality where the building block can be this project is Mathematical Model solver. One can take any picture of a mathematical problem and by this project one can recognise the digit inside it and then computer can compute that problem on its own.If a wrong answer comes, it can be checked through a step by step process by the computer and if it recognized the answer wrongly, it must be trained again. One has to train the model in various extents to recognise the various digits not only 0 to 9 but also more and more figures, like derivative integration and others.

The better functionality of this project can be license plate verification.Car license plate can be checked and one can set the record rightfully that which car is passing the gate and when by the recognition of characters.

SYSTEM CONFIGURATION

Software requirements:

These are the software requirements for running this project.

- **Operating System**: Windows 10/8/7 (incl. 64-bit), Mac OS, Linux
- **Language**: Python 3
- **IDE**: JetBrains PyCharm Community Edition 2019.1.3 x64

Hardware requirements:

- **Processor**: 64 bit, quad-core, 2.5 GHz minimum per core
- **RAM**: 4 GB or more.
- **HDD:** 20 GB of available space or more.
- **Display**: Dual XGA (1024 x 768) or higher resolution monitors.
- **Camera:** A detachable webcam.
- **Keyboard:** A standard keyboard.

IV. IMPLEMENTATION

I. PRE-PROCESSING:

Pre-processing is an initial step in the machine and deep learning which focuses on improving the input data by reducing unwanted impurities and redundancy. To simplify and break down the input data we reshaped all the images present in the dataset in 2-dimensional images i.e (28,28,1). Each pixel value of the images lies between 0 to 255 so, we Normalized these pixel values by converting the dataset into 'float32' and then dividing by 255.0 so that the input features will range between 0.0 to 1.0.

we performed one-hot encoding to convert the y values into zeros and ones, making each number categorical, for example, an output value 4 will be converted into an array of zero and one i.e [0,0,0,0,1,0,0,0,0,0]

II. SUPPORT VECTOR MACHINE

The SVM in scikit-learn supports both dense (numpy.ndarray and convertible to that by numpy.asarray) and sparse (any scipy.sparse) sample vectors as input. In scikit-learn, SVC, NuSVC and LinearSVC are classes capable of performing multi-class classification on a dataset.

In this project we have used LinearSVC for classification of MNIST datasets that make use of a Linear kernel implemented with the help of LIBLINEAR. Various scikit-learn libraries like NumPy, matplotlib, pandas, Sklearn and seaborn have been used for the implementation purpose.

Firstly, we will download the MNIST datasets, followed by loading it and reading those CSV files using pandas. After this, plotting of some samples as well as converting into matrix followed by normalization and scaling of features have been done. Finally, we have created a linear SVM model and confusion matrix that is used to measure the accuracy of the model.

SVM or Support Vector Machine is a specific type of supervised ML method that intends to classify the data points by maximizing the margin among classes in a high-dimensional space . SVM is a representation of examples as points in space, mapped due to the fact that the separate classes are divided by a fair gap that is as extensive as possible. After that new examples are mapped into that same space and anticipated to reside in a category based on which side of the gap they fall on.

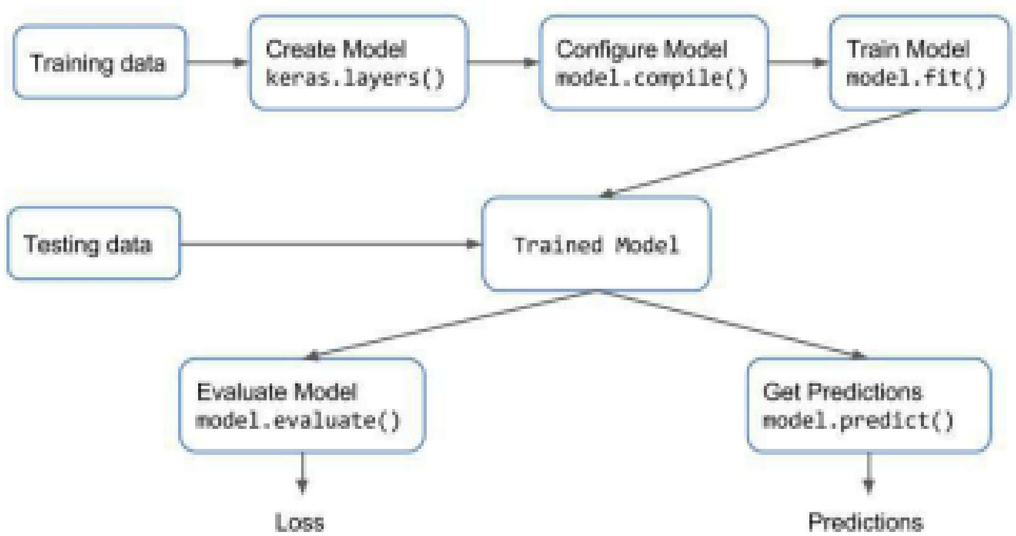

Fig 4.1

The optimum algorithm is developed through a “training” phase in which training data are adopted to develop an algorithm capable of discriminating between groups earlier defined by the operator, and the “testing” phase in which the algorithm is adopted to blind-predict the group to which a new perception belongs .

It also provides a very accurate classification performance over the training records and produces enough search space for the accurate classification of future data parameters. Hence it always ensures a series of parameter combinations no less than on a sensible subset of the data. In SVM it’s better to scale the data always, because it will extremely improve the results. Therefore be cautious with big datasets, as it may lead to an increase in training time.

Support Vector Machine (SVM) is a specific type of supervised ML method that intends to classify the data points by maximizing the margin among classes in a high-dimensional space (Pereira, 2009). SVM is a representation of examples as points in space,mapped due to the fact that the separate classes are divided by a fair gap that is as extensive as possible. After that new examples are mapped into that same space and anticipated to reside in a category based on which side of the gap they fall.

The optimum algorithm is developed through a “training” phase in which training data are adopted to develop an algorithm capable of discriminating between groups earlier defined by the operator , and the “testing” phase in which the algorithm is adopted to blind-predict the group to which a new perception . It also provides a very accurate classification performance over the training records and produces enough search space for the accurate classification of future data parameters. Hence, it always ensures a series of parameter combinations noless than on a sensible subset of the data.

APPLICATIONS

In recent years, many handwritten digit recognition systems have been proposed for practical applications which

- National ID number recognition system
- Postal office automation with code number recognition on Envelope
- Automatic license plate recognition

ADVANTAGE

- It has been proven successfully used for many handwriting and computer recognition.
- Reliable enough for writing recognition because of low accuracy.
- Accuracy results are quite high.
- Ease of use.
- Segmentation process simpler and the accuracy of the writing recognition.

DISADVANTAGE

- In order for high accuracy in the training process should use many Samples
- Long training time
- The accuracy of the system is not 100%.
- It can only recognise a single digit from the user.

OUTPUT

TRAIN IMAGE:

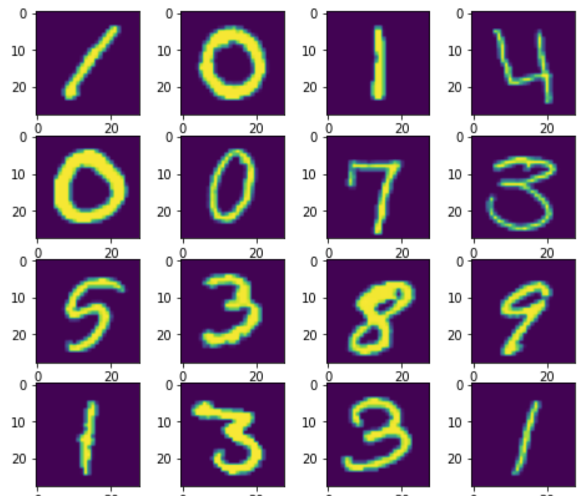

Fig 5.1

TEST IMAGE:

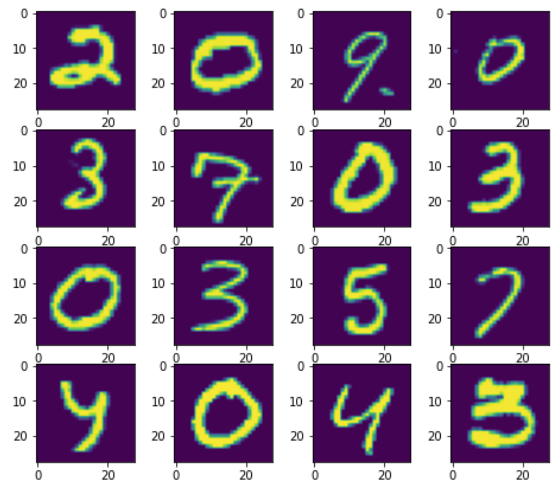

Fig 5.2

ACCURACY:

```
acc:  0.9899523809523809
Accuracy: 0.97 (+/- 0.00)
[2 2 2 ... 2 2 2]
```

RESULT:

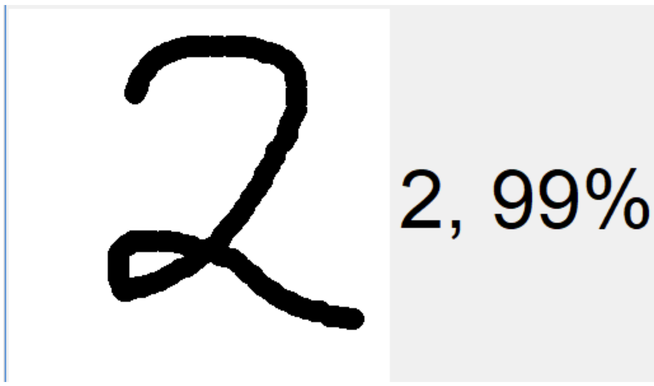

Fig 6.1

Fig 6.2

Fig 6.3

RESULT:

we have successfully built a Python deep learning project on handwritten digit recognition project.The intention was to make it work on real-life data, apart from the test dataset.We have built and trained the Support Vector Machine which is very effective for image classification purposes.we have built and trained the Convolutional neural network model which is very effective for image classification purposes. And we have correctly predicted the digit in a particular custom image also. Later on, we build the GUI where we draw a digit on the canvas then we classify the digit and show the results.

CONCLUSION:

The main objective of this investigation is to find a representation of isolated handwritten digits that allow their effective recognition. In this paper used a 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 well in terms of accuracy and time complexity. The overall highest accuracy 98% 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 recognition of handwritten numerals without using any standard classification techniques.

FUTURE ENHANCEMENT:

The future development of the applications based on algorithms of deep and machine learning is practically boundless. In the future, we can work on a denser or hybrid algorithm than the current set of algorithms with more manifold data to achieve the solutions to many problems.

In future, the application of these algorithms lies from the public to high-level authorities, as from the differentiation of the algorithms above and with future development we can attain high-level functioning applications which can be used in the classified or government agencies as well as for the common people, we can use these algorithms in hospitals application for detailed medical diagnosis, treatment and monitoring the patients, we can use it in surveillances system to keep tracks of the suspicious activity under the system, in fingerprint and retinal scanners, database filtering applications, Equipment checking for national forces and many more problems of both major and minor category.

The advancement in this field can help us create an environment of safety, awareness and comfort by using these algorithms in day to day application and high-level application (i.e. Corporate level or Government level). Application-based on artificial intelligence and deep learning is the future of the technological world because of their absolute accuracy and advantages over many major problems

REFERENCES:

[1]“Handwriting recognition”: https : //en.wikipedia.org/wiki/Handwritingrecognition

[2] “What can a digit recognizer be used for?”: https : //www.quora.com/W hat−can−a−digit−recognizer− be − used − for

[3] ”Handwritten Digit Recognition using Machine Learning Algorithms”, S M Shamim, Mohammad Badrul Alam Miah, Angona Sarker, Masud Rana & Abdullah Al Jobair.

[4] ”Handwritten Digit Recognition Using Deep Learning”, Anuj Dutt and Aashi Dutt.

[5] ”Handwritten recognition using SVM, KNN, and Neural networks”, Norhidayu binti Abdul Hamid, Nilam Nur Binti Amir Sharif.

[6] ”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.

[7] ”Advancements in Image Classification using Convolutional Neural Network” by Farhana Sultana, Abu Sufian & Paramartha Dutta.

[8] ”Comparison of machine learning methods for classifying mediastinal lymph node metastasis of non-small cell lung cancer from 18 FFDG PET/CT images” by Hongkai Wang, Zongwei Zhou, Yingci Li, Zhonghua Chen, Peiou Lu, Wenzhi Wang, Wanyu Liu, and Lijuan Yu.

[9] https : //github.com/rishikakushwah16/SV M digit recognition using MNIST dataset

[10] “Support Vector Machine Algorithm”: https : //www.javatpoint.com/machine−learning−support− vector − machine − algorithm