**CNN Based Deep Learning Model for Handwritten Digit Recognition (HDR)**

**Prakanshu, Prakhar**

**Abstract** – Human can see and visually sense the world around them by using their eyes and brain computer vision work an enabling computer to see and process image in the same way that human vision does. Handwritten digit Recognition is one of the important applications of computer vision field to recognise image(digit). Our aim to build a model to identify and determine the handwritten digit from its image with better accuracy. Convolution neural network (CNNs) are very effective in perceiving the structure of handwritten digit way that help in automatic extraction of distinct features and make CNN the most suitable approach for solving handwritten digit recognition problems.

**Keywords-** Convolution neural network, Support Vector Machine, Deep Learning, Classifier

**1.Introduction**

Whenever we heard a Word Handwritten digit recognition. It plays an important role in information processing. a lot of information is available on paper. The aim of a Handwriting recognition system is to convert handwritten digits or character into machine readable formats. The main application is vehicle licence -plate recognition, cheque truncation system (CTS)scanning, old documents automation in libraries and banks etc. All these deal with large databases and hence demands high recognition accuracy, lesser computational complexity, and consistent performance of the recognition system [5].

In Handwritten recognition digits character are given as input, the model can be recognized by the system. A simple artificial neural network (ANN) as an input layer and some hidden layer between the input and output layer. CNN has very similar architecture as ANN. In CNN, the layer has three dimensions here all the neurons are fully connected to the local receptive field.[4] we attempt to reduce the overall classification time by reducing the feature space used to train the model to get an optimal model to classify handwritten digits. The feature map reduction has been done by selecting the filter maps of a convolutional layer in the CNN randomly. The experimented results provide conclusive evidence for the usefulness of CNNs with reduced feature space to deal with less complex problems. The results signify that the projected CNN model leads to an improvement in the recognition rate compared with other CNN-based algorithms with greater accuracy. This work will open a new way toward digitalization. Furthermore, this work could be extended to letters reducing humanistic efforts, as the digit recognition performance of our proposed framework was beyond what can be achieved by a skilled human.

Support vector machine (SVM) is a learning method based on statistical learning theories, which proposed by Vapnik. Basing on the principle of structural risk minimization, SVM can improve the generalization ability of the learning machine as much as possible. Even the decision rules obtained from limited training samples can still get small errors for independent test datasets. In recent years, SVM has been widely used in pattern recognition, regression analysis and feature extraction. Vapnik found that different kernel functions had little effect on SVM performance. The key factors affecting SVM performance are kernel function parameters and penalty coefficient. Therefore, the study of kernel function parameters and penalty coefficients is an important field to improve the performance of machine learning [7].

**2.Related work**

Handwriting digit recognition has an active community of academics studying it. A lot of important work on convolutional neural networks happened for handwritten digit recognition [1,6,8,10]. There are many active areas of research such as Online Recognition, Offline recognition, Real-Time Handwriting Recognition, Signature Verification, Postal-Address Interpretation, Bank-Check Processing, Writer Recognition [3]. CNN is playing an important role in many sectors like image processing. It has a powerful impact on many fields. Even, in nanotechnologies like manufacturing semiconductors, CNN is used for fault detection and classification. Handwritten digit recognition has become an issue of interest among researchers. There are many papers and articles are being published these days about this topic. In research, it is shown that Deep Learning algorithm like multilayer CNN using Kera’s with Theano and TensorFlow gives the highest accuracy in comparison with the most widely used machine learning algorithms like SVM, KNN & RFC. Because of its highest accuracy, Convolutional Neural Network (CNN) is being used on a large scale in image classification, video analysis, etc. Many researchers are trying to make sentiment recognition in a sentence. CNN is being used in natural language processing and sentiment recognition by varying different parameters [4]. It is challenging to get a good performance as more parameters are needed for the large-scale neural network. Many researchers are trying to increase the accuracy with less error in CNN. The performance of CNNs depends mainly on the choice of hyper-parameters, which are usually decided on a trial-and-error basis. Some of the hyper-parameters are, namely, activation function, number of epochs, kernel size, learning rate, hidden units, hidden layers, etc. These parameters are very important as they control the way an algorithm learns from data [5]. Hyper-parameters differ from model parameters and must be decided before the training begins.

In recent years, the convolutional neural networks have been effectively used for handwritten digit recognition and primarily for benchmark MNIST handwritten digit dataset. Most of the experiments achieved high recognition accuracy more than 98% or 99% [32]. The high recognition accuracy of 99.73% on MNIST dataset is achieved while experimenting with the famous committee technique of combining multiple CNN’s in an ensemble network. The work was further extended into 35-net committee from the earlier 7-net committee and reported very high accuracy of 99.77% [34]. The Deep Belief Networks (DBN) with three layers along with a greedy algorithm were investigated for MNIST dataset and reported the accuracy of 98.75% [8]. The bend directional feature maps were investigated using CNN for in–air handwritten Chinese character recognition. Lauer et al. work on a LeNet5 convolutional neural network architecture-based feature extractor for the MNIST database [5]. The work reported excellent recognition accuracy. The impressive performance of the research work clearly shows the effectiveness of CNN feature extraction step. The present work is also motivated by the performance of deep learning methods in handwriting recognition domain. The structural risk minimization ability of SVM and deep feature extraction ability of CNN when combined has proved to be extremely useful in many domains [4]. The fusion of CNN-SVM can be highly useful in handwriting recognition and hence the aim of the present work. Niu and Suen integrates the CNN and SVM for MNIST digit database and reported a recognition rate of 99.81% [6]. The authors used rejection rules to achieve high reliabilities. Guo et al. investigated hybrid CNN-Hidden Markov Model (HMM) for house numbers recognition from the street view images. The experimental results demonstrated that the hybrid CNN-SVM model effectively improved the recognition accuracy for handwritten digit.

**3.Experimental setup**

Deep Learning has emerged as a central tool for self-perception problems like understanding images, a voice from humans, robots exploring the world. We aim to implement the concept of Convolutional Neural Network for digit recognition. Understanding CNN and applying it to the handwritten digit recognition system is the target of the proposed model. Convolutional Neural Network extracts the features maps from the 2D images.[3]

**3.1 Dataset**

In our dataset make firstly, split the images of all digits (0-9), secondly make separate folder (0-1-2-3-4-5-6-7-8-9) all folder in captured\_images folder. In ‘0’ folder store zero-digit images and ‘1’ digit folder store one-digit images and store as like up to 9. Every single folder contains 200 images. We make our dataset of 2000 images.

The dataset contain 2000 images in which 1600 images used for training as well as few of them can be used for cross validation purposed & 400 images used for testing .All the digit are Grayscale and positioned in a fixed size where the intensity lies at the centre of the image with 28\*28 pixels ,Since all the images are 28\*28 pixels ,it form an array which can be flattened into 28\*28 = 784 dimensional vector .Each component of the vector is a binary value which describe the intensity of the pixels.[4]

**3.2. One hot Encoding vector**

A one hot encoding is a representation of categorical variables as binary vectors. This first requires that the categorical values be mapped to integer values. Then, each integer value is represented as a binary vector that is all zero values except the index of the integer, which is marked with a 1.

In our model having 10 categories (0,1,2,3,4,5,6,7,8,9) make 10 labels, create one hot encoded vector from image name.

**4.Methodology**

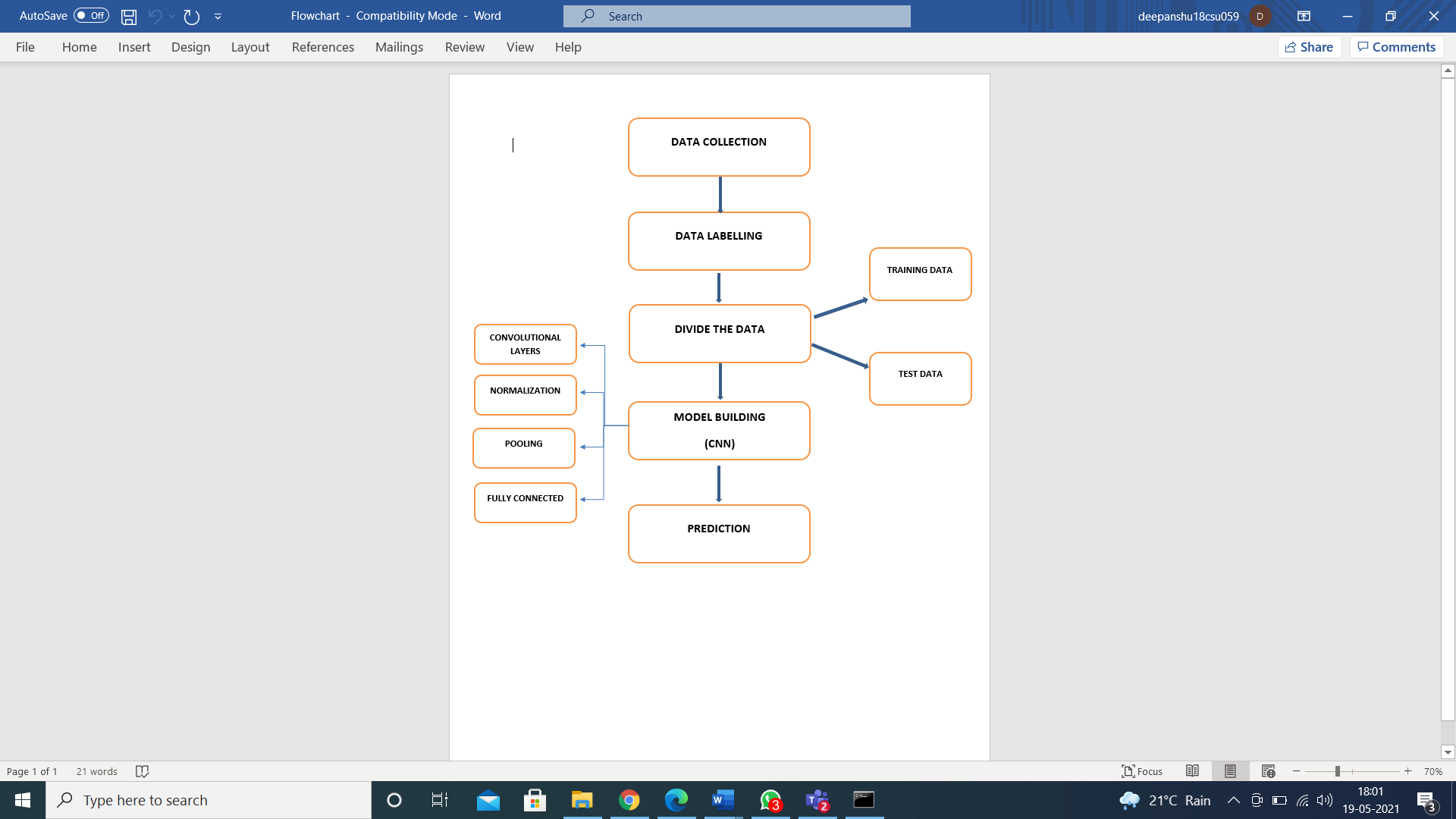
****

Fig 1: -Flow chart of methodology

**4.1Data collection: -** Data collection is defined as the procedure of collecting, measuring and analyzing accurate insights for research using standard validated techniques. A researcher can evaluate their hypothesis based on collected data. In most cases, data collection is the primary and most important step for research, irrespective of the field of research. The approach of data collection is different for different fields of study, depending on the required information.

In this cell we are Using train.csv file it having numerical information of all 10digits combine information (0 to 9), Firstly we fetch the information separate digit wise and make separate folder 0-9 In separate folder like o folder having only 0 images as like make up to 10 folders.

**4.2Create the data with label: -** this part of code using one-hot Encoding technique for making label of each category (0,1,2—9) having total 10 category for complete dataset, So Make 10 hot Encoding vector for separate folder representation. A one hot encoding is a representation of categorical variables as binary vectors. This first requires that the categorical values be mapped to integer values. Then, each integer value is represented as a binary vector that is all zero values except the index of the integer, which is marked with a 1.

**4.3Dividing data into training and testing part: -** In which divide the data into testing and training part. Our dataset contains total 2000 images in which 1600 images used for training as well as few of them can be used for cross validation purposed & 400 images used for testing. Our dataset divides 80 ratio 20 ,80% of training part and 20% of testing of our model

**4.4 Model Building**

**Convolutional Neural Network Architecture**

A basic convolutional neural network comprises three components, namely, the convolutional layer, the pooling layer, and the output layer. The pooling layer is optional sometimes. The typical convolutional neural network architecture with three convolutional layers is well adapted for the classification of handwritten images. Hidden layer constitutes network of repetitive convolutional and pooling layers thus finally ends at one or more fully connected layer.

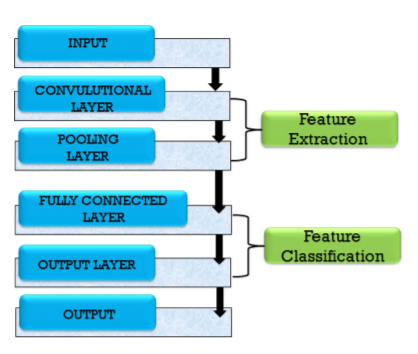


Fig2: - Architecture of Convolutional Neural Network Model

**Convolutional layer**

The convolutional layer is the first layer which can extract features from the images. Because pixels are only related to the adjacent and close pixels, convolution allows us to preserve the relationship between different parts of an image. Convolution is filtering the image with a smaller pixel filter to decrease the size of the image without losing the relationship between pixels. When we apply convolution to the 5x5 image by using a 3x3 filter with 1x1 stride.[3].In our Model apply

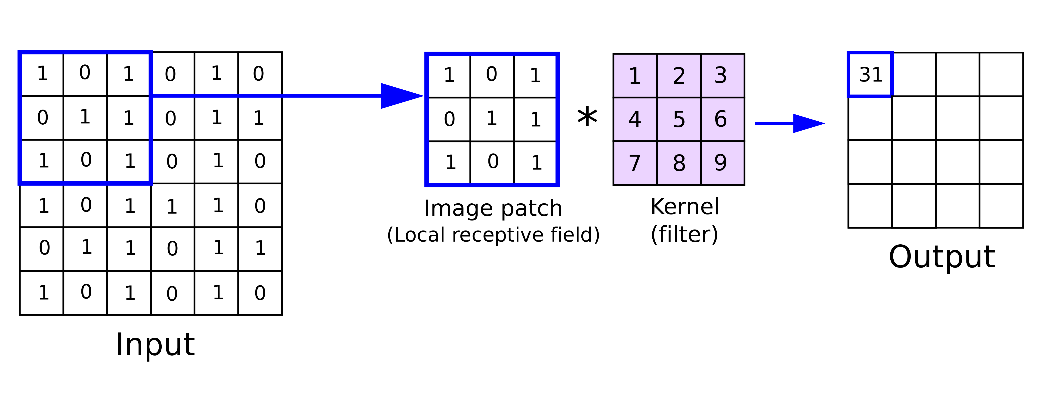


Fig3: - Convolution Layer

**Pooling layer**

A pooling layer is added between two convolutional layers to reduce the input dimensionally and hence to reduce the computational complexity. Pooling allows the selected values to be passed to the next layer while leaving the unnecessary values behind. The pooling layer also helps in feature selection and in controlling overfitting. The pooling operation is done independently. It works by extracting only one output value from the tiled non-overlapping sub-regions of the input images. The common types of pooling operations are max-pooling and avg-pooling (where max and avg represent maxima and average, respectively). The max-pooling operation is generally favourable in modern applications because it takes the maximum values from each sub-region, keeping maximum information. This leads to faster convergence and better generalization [2]. The max-pooling operation for converting a 4 \* 4 convolved output into a 2 \* 2 output with stride size 2. The maximum number is taken from each convolved output (of size 2 \* 2) resulting is for reducing the overall size 2\*2.[5]

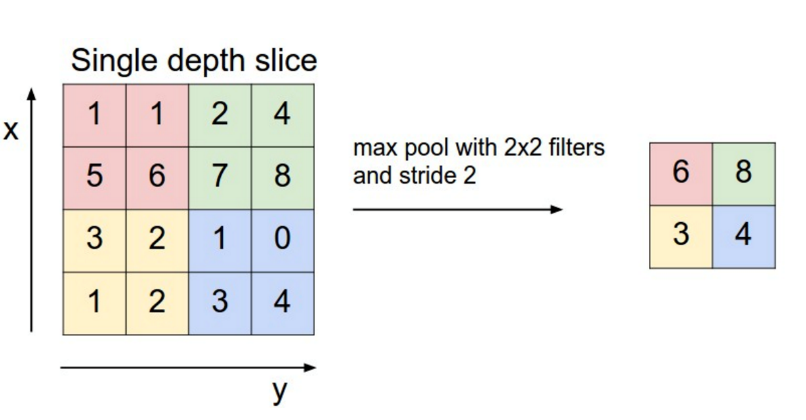
:

Fig4: -Pooling Layer

**Fully connected layer**

Lastly, there is fully connected layer after convolution and pooling layer in the standard neural network (separate neuron for each pixel) which is comprised of n numbers of neurons, where n is the predicted class number. For example, there are ten neurons for ten classes (0–9) in digit character classification problem. However, there should be 26 neurons for 26 classes (a–z) for English character classification problem.[1]

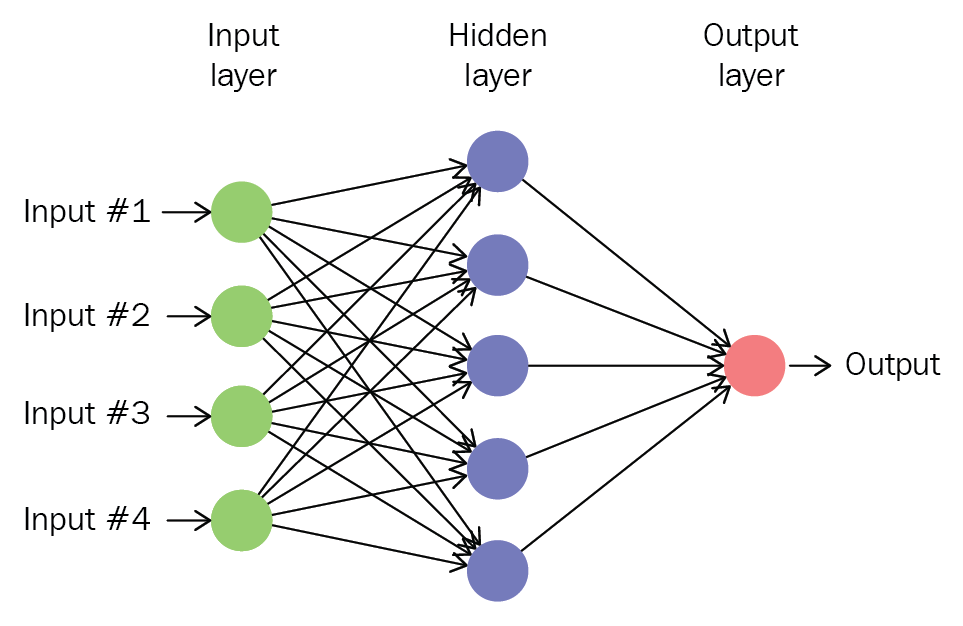


Fig5:-Fully Connected layer

**Normalisation-**

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

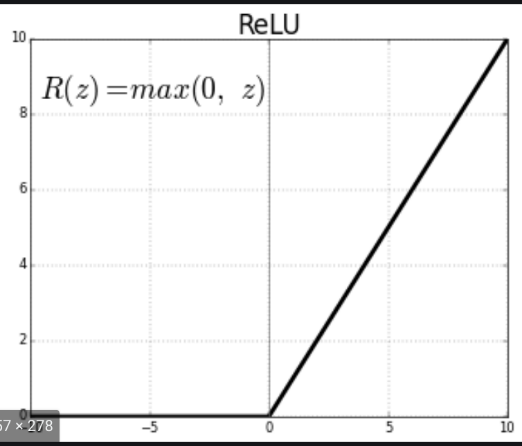


Fig6: - Relu-Activation Function

**Support Vector Classifier process**

The main steps of handwritten digit recognition are as follows:

Firstly, handwritten digits are analysed, and a series of pre-processing is carried out to extract a set of feature vectors. Then, those training feature vectors are sent to the training I/O of SVM to train the parameters and support vectors. After that, the feature vectors of the handwritten digits, which needs to be recognized, are extracted, and sent to the prediction I/O of SVM. At last, the prediction results are outputted, and the recognition rate is calculated.

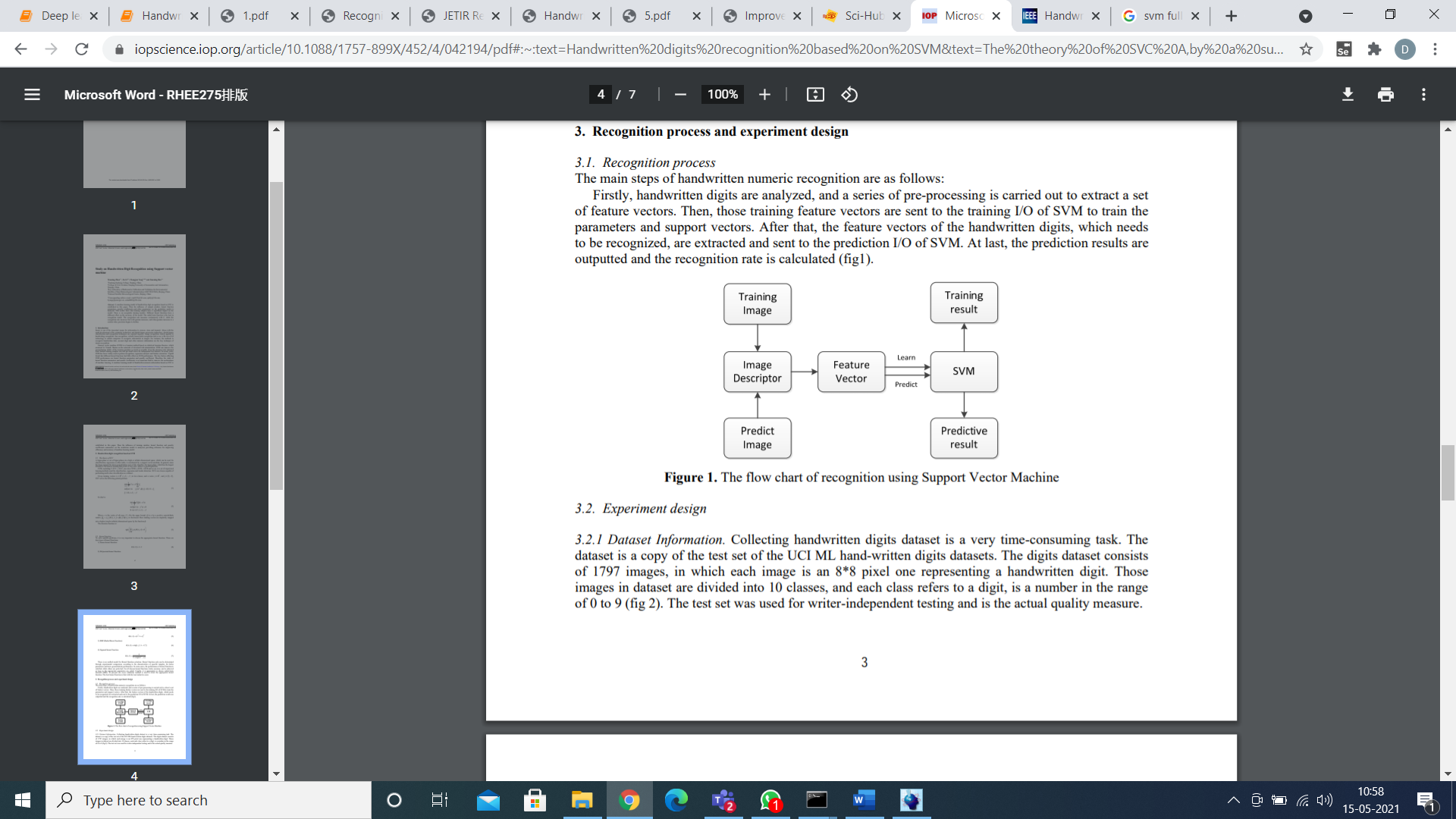


Fig7: - The flow chart of recognition using Support Vector Machine

**4.5 Prediction:** - In prediction part of our model make ‘new\_images’ folder it is having 10 images of all digits 0,1,2,3,4,5,6,7,8,9 this folder pass for testing part of our model it. The proposed CNN achieved relatively higher prediction accuracy of 93.75%, while the SVM algorithms obtained prediction accuracies of 84%.

**5.Results and Experiment**

In this paper apply two different models for finding better accuracy and best prediction outcomes first one is Convolutional Neural Network Model (CNN), and second model is Support Vector Classifier (SVC)Model

**CNN Model**

There are some digits which are in good handwriting. our model will be able classify them correctly, for example in Fig8: - shown in testing part given 10 images our model and model classify 10-digit images correctly.

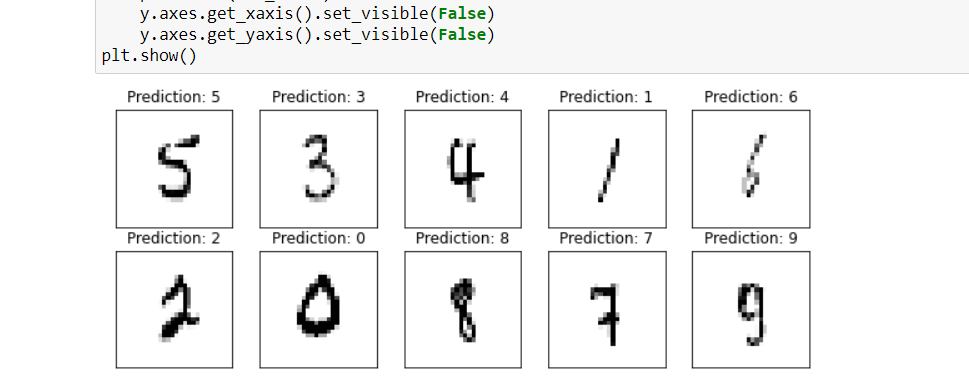


Fig8**: -** Examining result

Table1**: -** Summary of the experiment

|  |  |  |
| --- | --- | --- |
| **Drop out (Overfitting)** | **No. of epochs** | **Accuracy** |
| 0.8 | 10 | 83 % |
| 0.8 | 12 | 87.89 % |
| 0.6 | 16 | 93.75 % |

Testing accuracy 93.75% implies that the model is trained well for prediction. Training size effect the accuracy and accuracy increase as the number of data increases. The more data in the training set, the smaller the impact of training error and test error and ultimately the accuracy can be improved.

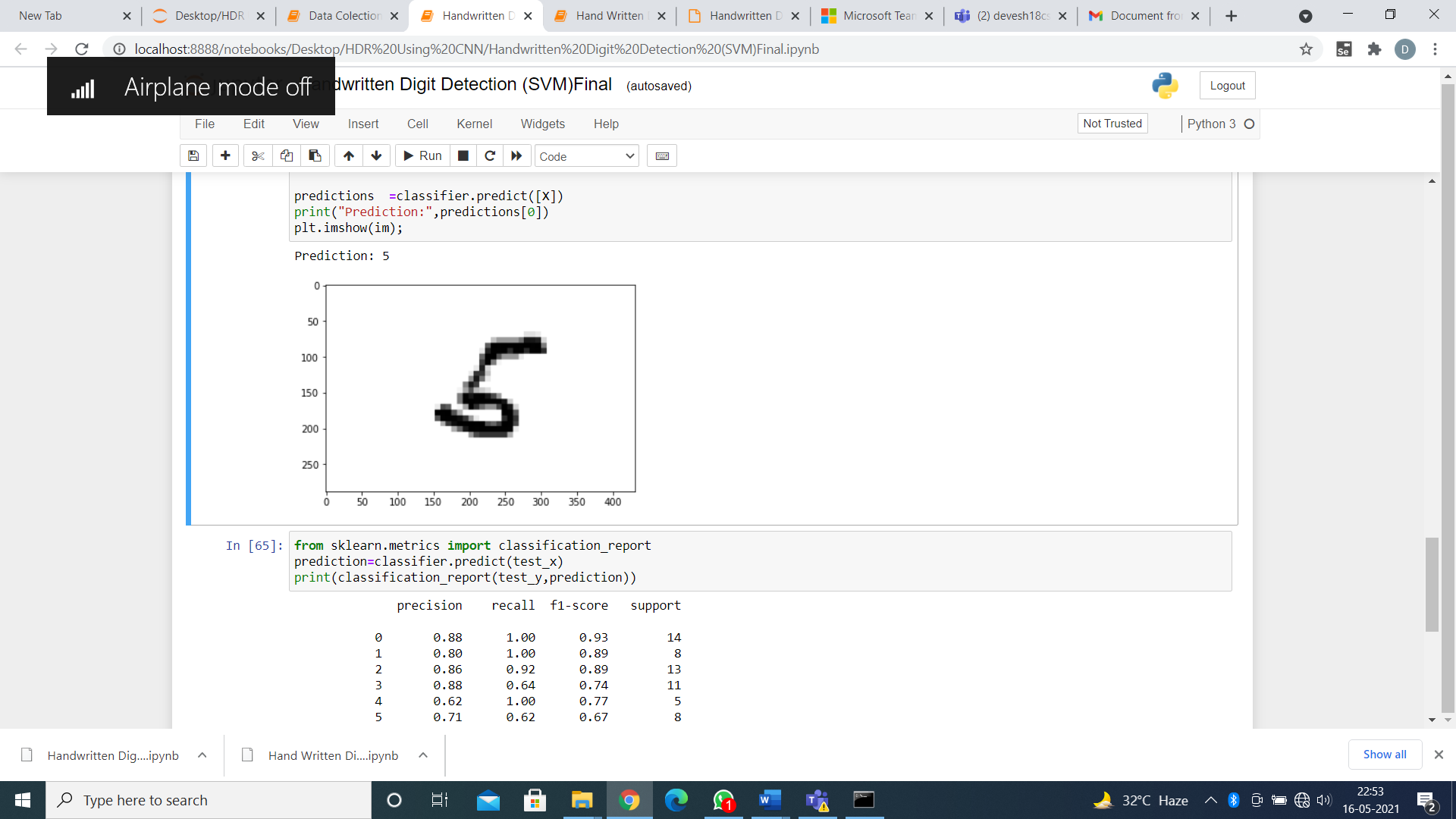
**SVC Model**

Two class SVM or binary classifier is applied to solve the problem of Handwritten digit recognition by using one verses all method. We have used C-SVM as the classifier with polynomial function as the kernel type and we have set the values of other parameters to their default values. The SVM has been trained by using the training samples from MNIST dataset. The classifier works in two modes i.e., training and testing. Training is done by taking the feature vectors which are stored in matrices form after the completion of pre-processing and feature extraction. The testing of the characters has been done by using the result of training. The accuracy of the SVM classifier for recognition of handwritten digit is found to be 84%. Accuracy of digits has been shown in Table2: - Evaluation on test data.

Table2 :- Evaluation on test data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-Score | Support |
| 0 | 0.77 | 0.83 | 0.80 | 12 |
| 1 | 1.00 | 1.00 | 1.00 | 11 |
| 2 | 0.93 | 0.82 | 0.87 | 17 |
| 3 | 0.92 | 0.85 | 0.88 | 13 |
| 4 | 0.83 | 0.00 | 0.91 | 10 |
| 5 | 0.70 | 0.70 | 0.70 | 10 |
| 6 | 0.70 | 0.88 | 0.78 | 8 |
| 7 | 0.89 | 0.89 | 0.79 | 9 |
| 8 | 0.75 | 0.60 | 0.67 | 5 |
| 9 | 0.75 | 0.60 | 0.67 | 5 |
| Accuracy |  |  | 0.84 | 100 |
| Macro avg | 0.82 | 0.82 | 0.82 | 100 |
| Weighted avg | 0.84 | 0.84 | 0.84 | 100 |

our model will be able classify them correctly, for example in Fig10: - shown in Examining Result given an image our model and model classify 5-digit images correctly.



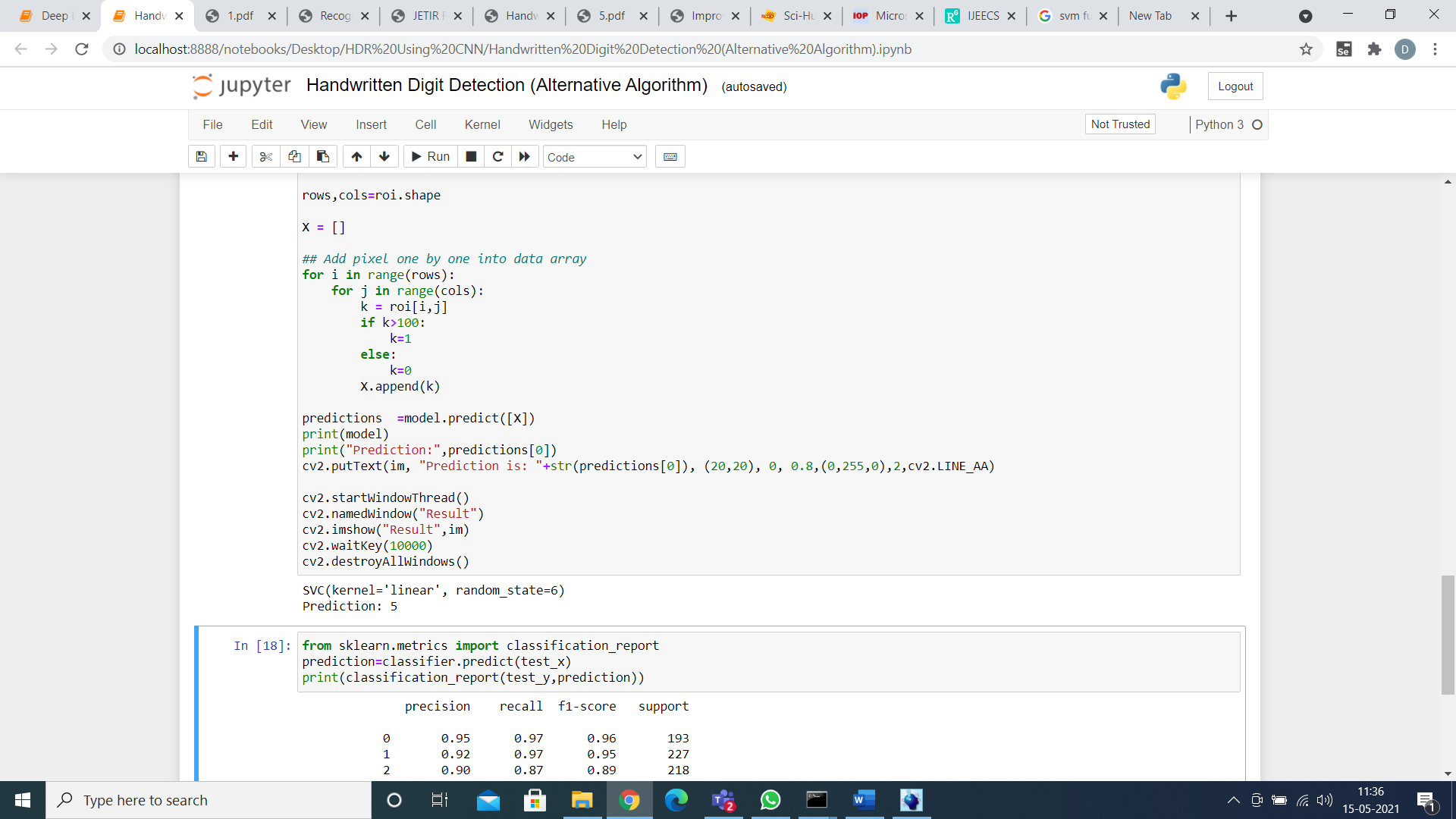


Fig9: - Examining result

**6.Conclusion**

In this work with the aim of improving the performance of handwritten digit recognition. The paper discusses in detail advances in the area of handwritten digit recognition. The most solution provided in this area directly or indirectly depends upon the quality of image as well as nature of material to be read. As seen from the results of the experiment, CNN proves to be far better than other classifiers. The results can be made more accurate with more convolution layers and a greater number of hidden neurons. It can completely abolish the need for typing. Digit recognition is an excellent prototype problem for learning about neural networks and it gives a great way to develop more advanced techniques of deep learning.

The CNN method performs the best with the accuracy of 93.75% and is the best among the methods that have been studied in this paper.

**7.References**

1. Ali, S., Shaukat, Z., Azeem, M., Sakhawat, Z., Mahmood, T., & Rehman, K. U. (2019). An efficient and improved scheme for handwritten digit recognition based on convolutional neural network. Beijing: Springer Nature.

2. Hossain, M., & Ali, M. (2019). Recognition of Handwritten Digit using Convolutional Neural Network (CNN). Pabna, Bangladesh: Global Journals.

4.Siddique, F., Sakib, S., & Sidiqque, A. B. (2019). Recognition of Handwritten Digit using Convolutional Neural Network in Python with Tensorflow and Comparison of Performance for Various Hidden Layers. Dhaka, Bangladesh: IEEE.

5.Ahlawat, S., Choudhary, A., Nayyar, A., Singh, S., & Yoon, B. (2020). Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN). India: MDPI.

6.Lan, L. S. (2006). M-SVC (mixed-norm SVC) - a novel form of support vector classifier. Greece: IEEE.

7.Tohme, M., & Lengelle, R. (2008). F-SVC: A SIMPLE AND FAST TRAINING ALGORITHM SOFT MARGIN SUPPORT VECTOR CLASSIFICATION. Rouffach, France: Sci Hub.

8.Katiyar, G., & Mehfuz, S. (2015). SVM Based Off-Line Handwritten Digit Recognition. New Delhi, India: Sci Hub.

9.Ghosh, M. M., & Maghari, A. Y. (2017). A Comparative Study on Handwriting Digit Recognition Using Neural Networks. Palestine: IEEE.

10.Shamim, S., Alam Miah, M., Sarker, A., Rana, M., & Jobair, A. A. (2018). Handwritten Digit Recognition using Machine Learning Algorithms. Bangladesh: Global Journals.

11. Sharma, A., Barole, Y., Kerhalkar, K., & K.R., D. (2016). *Neural Network Based Handwritten Digit Recognition for Managing Examination Score in Paper Based Test.* Tamil Nadu, India: IJAREEIE.

12.https://www.google.com/search?q=convolution+neural+network+fully+connected&tbm=isch&ved=2ahUKEwjxxLPP7KfwAhW2k0sFHWROAe4Q2-cCegQIABAA&oq=convolution+neural+network+fully+connected&gs\_lcp=CgNpbWcQAzoECCMQJzoGCAAQChAYOgQIABAYOgIIAFDd9QNYk7IEYKq0BGgBcAB4AIABiAGIAbsYkgEEMC4yN5gBAKABAaoBC2d3cy13aXotaW1nwAEB&sclient=img&ei=P\_SMYLGpGranrtoP5JyF8A4&bih=754&biw=1536&rlz=1C1CHBF\_enIN947IN948#imgrc=Gy1Q\_GaC-vLWoM&imgdii=\_9yQAi83CUFIqM

13.<https://www.google.com/search?q=convolution+neural+network+max+pooling&tbm=isch&ved=2ahUKEwiaoZiX7KfwAhWSPCsKHQcpDJ8Q2-cCegQIABAA&oq=convolution+neural+network+max+pooling&gs_lcp=CgNpbWcQAzoCCAA6BAgAEEM6BggAEAoQGDoECAAQGFCj3QZY44oHYPuNB2gBcAB4AIABiQGIAeoLkgEEMC4xM5gBAKABAaoBC2d3cy13aXotaW1nwAEB&sclient=img&ei=yfOMYNqYIZL5rAGH0rD4CQ&bih=754&biw=1536&rlz=1C1CHBF_enIN947IN948#imgrc=ZT7SI-NcXhcLVM&imgdii=9uswke-cEfs5HM>

14.<https://www.google.com/search?q=convolution+neural+network&rlz=1C1CHBF_enIN947IN948&sxsrf=ALeKk01RkANWuFjp8KSbydwuu9xh_-K_5g:1619850183302&source=lnms&tbm=isch&sa=X&ved=2ahUKEwjr7I6W7KfwAhX-zDgGHURKBrYQ_AUoAXoECAIQAw&biw=1536&bih=754&dpr=1.25#imgrc=KAUvm7W3C4G4qM&imgdii=LJE62p6XfyT2-M>

15.<https://www.google.com/url?sa=i&url=https%3A%2F%2Fmedium.com%2Fanalytics-vidhya%2Frelu-activation-increase-accuracy-by-being-greedy-6b93c7c40882&psig=AOvVaw1kUTaizEQ-U18et0ES2dxX&ust=1619941759492000&source=images&cd=vfe&ved=0CAIQjRxqFwoTCKDI1b__p_ACFQAAAAAdAAAAABAT>