Handwritten Digit Recognition Using Neural Networks

Navneet Kaur
Computer Science and Engineering Department
Thapar Institute of Engineering and Technology
Patiala, Punjab, India
nkaur_be16@thapar.edu

Jodh Singh
Computer Science and Engineering Department
Thapar Institute of Engineering and Technology
Patiala, Punjab, India
jsingh6_be16@thapar.edu

ABSTRACT
Recognition of hand written digits is one of the most difficult tasks in field of machine learning attributed to varied styles of handwriting of an individual. Hand writing recognition has been gaining significant importance over the past few years on account of plethora of applications that it can be used in. It has a large potential of being used in automatic data entry, automated answer sheet evaluation, check processing in banks, postal mail checking, etc. As the amount of data recorded daily keeps on increasing exponentially there ought to be techniques to extract knowledge from that data and use it for making decisions as well as performing appropriate action. Similar is the case with recognition of hand written digits. In this paper we aim to study the techniques used in recognizing hand written digits, train a machine learning algorithm in python language that will recognize hand written digits from 0 to 9 using the dataset from MNIST and finally test the model on unseen images from the internet. Accuracy of the model will be analyzed on classification parameters.

Categories and Subject Descriptors
Machine Learning and Deep Learning

General Terms
Recognition, Training

Keywords
Handwritten, Neural Networks

.

1. INTRODUCTION
Recognition of objects using the techniques of machine learning and deep learning on image data is one of the major tasks that can be performed under the domain of these subjects. The MNIST (Modified National Institute of Standards and Technology database) dataset for the handwritten digits forms the basis for handwritten digit recognition using models studied under these subjects. The features represented by the handwritten digits are often difficult to recognize, model and extract but with the help of CNN (Convolutional Neural Networks) the task gets simplified up to a great extent. The CNN model uses some constraints based on feature selection, feature extraction and modification and subsampling. Also the neural network model proposed uses an input layer, some hidden layers (described in later section) and an output layer to label the images into one out of ten possible classes.

1.1 Motivation
The motivation for writing this research paper stems from the fact that we opted for Machine Learning as our elective subject in the 3rd year of our Bachelor of Engineering degree. During this five months long course we realized the true power of machine learning and how it can be used in various applications to provide solutions for numerous real world challenges. One such application we came to know about was handwritten digit recognition and its vast application areas. While doing the research for our research paper we came across few articles which referred to this application as the "hello world" of object recognition tasks [1]. Therefore we decided to opt for this project so that we can get to learn about this interesting field and clearly understand the basic concepts of object recognition tasks so that we develop a strong base for the more advanced applications of machine learning. And at the same time try to develop a model close to the state of the art accuracy levels.

1.2 Need of the system
Human handwriting varies across individuals and therefore there arise a need to develop handwriting recognition systems which can identify the characters based on their handwritten formation. The files, photographs and documents which contain handwritten text can be converted to digital text with the help of this system. Such a system will be helpful in removing the incongruities in human handwriting and can also evaluate the mathematical handwritten text to give appropriate results easing the work of bookkeepers.

1.3 Applications of the proposed system
This handwritten digit recognition system has numerous applications dedicated in various fields. The scanned text can be directly used in Google search instead of typing the required query. Images that have been

captioned with handwritten text can be scanned for converting the text into different languages. The mathematical handwritten text can be scanned and corresponding calculations can be performed to obtain desired results. Answer sheets can be evaluated by scanning the handwritten text for the answers. The system will simplify data entry processes as data can be added directly to database by scanning the related documents.

2. LITERATURE SURVEY

In order to recognize handwritten digits efficiently various machine learning algorithms have been proposed all of which have different pre-processing steps, accuracy results and datasets involved. The literature survey regarding this topic brought forward all the possible methods that can possibly be adopted to recognize handwritten digits. Paper [2] presented a comparative analysis of four major methods that could be used to recognize handwritten digits and the winner came out to be the method that involved extensive pre-processing on the dataset. In [3] the simplicity of usage of machine learning algorithms such as K-Nearest-Neighbors (KNN) and Support Vector Machine (SVM) has been highlighted along with listing the limitations of each of them. In [4] the MNIST dataset of handwritten images has been deformed by changing the pixel values in the images randomly and neural network model has been trained on this deformed set of images. In spite of this an accuracy quiet near to the actual model has been observed. In [5] LeNet5 convolutional neural network has been used which overcomes the problem of feature extraction from image dataset and increases the accuracy results for overall classification. Paper [6] highlights the procedure for dataset collection where subjects have been asked to fill forms for digits from zero to nine. Further steps involved scanning of digit images from this dataset, resizing the images and experimentation regarding selecting appropriate parameters for neural network model. Paper [7] shows that steps involving feature extraction and application of ensemble model boost the accuracy of the recognizer. In [8] the model was trained using Pareto principle and testing of the model was done on a different dataset than the one it had been trained on. In [9] the dataset of images was first converted into binary set of images before applying it to the classical neural network model using back propagation technique. Research work in [10] employed a method known as Limited Receptive Area which has lesser training time and improved accuracy. Paper [11] has listed substantial work with Arabic digits rather than the digits from zero to nine. Out of this literature survey, the best suited model came out be the CNN classifier the experimentation and parameters related to which have been described in detail in section 4. The table below lists the inferences that could be drawn from various research papers solely devoted to the area of digit classification.

Table 1: Research Findings from Related Literature

Paper Title	Author Name	Conclusions/Findings
Handwritten digit recognition using neural networks. [2]	Amanda Biscoff, Patrick S.P. Wang	The paper highlights four major methods of recognizing handwritten digits efficiently. A comparative analysis of these four methods has been done and extensively presented in the paper. The finest of them all comes out to be a system that uses time series information along with rigorous pre-processing on the handwritten data.
Handwritten Digit Recognition Using K Nearest-Neighbor, Radial-Basis Function, and Backpropogati-on Neural Netwroks. [3]	Yuchun Lee	The findings in this paper show that the machine learning models using K Nearest Neighbor (KNN) and Support Vector Machine (SVM) model using the radial basis function classify the hand written digits with a very little error rate when the handwritten image dataset is quiet extensive. The KNN method is a simple model but requires large memory and training time. On the other hand the accuracy for radial basis classifier is high because it is capable of rejecting ambiguous data.
Deep, Big, Simple Neural Nets for Handwritten Digit Recognition. [4]	Dan Claudiu Ciresan, Ueli Meier, Luca Maria, J.Schmidhuber	The neural networks have been trained and tested on deformed images from the MNIST dataset and an improved accuracy has been observed in this paper. The learning has been sped up adjusting the number of hidden layers, number of neurons in each layer.
A trainable feature extractor for handwritten digit recognition. [5]	Fabien Lauer, Ching Y.Suen, Gerard Bloch	In this paper problems faced while extracting features from data before feeding it to the classification algorithm have been presented. The problem has been solved with the help of LeNet5 which is a convolutional neural network performing feature extraction prior to training. The results of training this model provide an accuracy which is very close to the highest obtainable accuracy till now on MNIST dataset.

Handwritten Digit Recognition Using Image Processing and Neural Networks. [6]	Faisal Tehseen Shah, KamranYousaf	The research work in this paper has been presented in two phases. In the first phase data has been collected from different subjects by making them fill the forms and then scanning of those forms has been done. In the second phase images of different handwritten digits have been extracted by collecting 100 images from one form of 16*16 pixel size. This dataset has then been processed by CNN containing three layers.
Handwritten Digit Recognition Using Multiple Feature Extraction Techniques and Classifier Ensemble. [7]	Rafael M. O. Cruz, George D. C. Cavalcanti, Tsang Ing Ren	In this paper the authors observed that different feature extraction algorithms yield better results for different types of digits. The designed model employed feature extraction methods and ensemble classifier, and six feature sets were extracted in which two were proposed by the authors and the remaining four were published in the previous research. The accuracy of the model is 99.68% for recognition of digits.
Feature selection using multi-objective genetic algorithms for handwritten digit recognition. [8]	S. Oliveira, R. Sabourin, F. Bortolozzi	In this research Pareto based method was employed in which fitness were evaluated using neural networks and the sensitivity analysis. Later on this method was validated on a different dataset to provide better generalization. And at last the proposed model was applied on the recognition system after training it.
Offline Handwritten Digit Recognition Using Neural Network. [9]	Sumedha B. Hallale, Geeta D. Salunke	In this paper the classical approach of neural network classification with back propagation is discussed by the authors. The dataset of binary images were preprocessed by normalizing and skeletonizing them before employing on the model.
Improved method of handwritten digit recognition tested on MNIST database. [10]	Ernst Kussul, Tatiana Baidyk	In this paper the LIRA (Limited Receptive Area) classifier was employed. The main highlight of the LIRA classifier is that the training process is comparatively very rapid. The developed model was initially trained and tested on the MNIST database. After that the model was tested on another database containing assembly microdevice images. The designed model had one of the best recognition accuracy of 99.41% on MNIST database.
Handwritten Digit Recognition Using Convolutiona-l Neural Networks. [11]	Haider A. Alwzwazy, Hayder M. Albehadili, Younes S. Alwan, Naz E. Islam	In this research paper the traditional CNN model is employed. The aim of the authors was to train and test the model on handwritten digits written in Arabic because not much work was done when this paper was published as Arabic digits possess a higher challenge to than digits written in Roman scripts. The dataset for the research was designed from digit-forms of various educational institutions of Arab countries.

3. RESEARCH OBJECTIVES

The objectives of conducting our research work on handwritten digit classification using convolutional neural networks was to study the science behind digit classification which is a major field of machine leaning. The aim was to scrutinize various machine learning and deep learning models that have been developed till now and are still being used extensively in digit classification, perform a comparative analysis of all such models and decide the best suited model which could be used for classifying handwritten digits. Other major objectives of our research are as presented in the sub-sections underneath.

3.1 Training a machine learning model

One of the research objectives was to train a machine learning model on the handwritten dataset of images by feeding the dataset into a convolutional neural network. Also the parameters of CNN were studied which involved selecting appropriate number of hidden layers and further appropriate number of neurons in each layer.

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 24, 24, 30)	780
max_pooling2d_3 (MaxPooling2	(None, 12, 12, 30)	0
conv2d_4 (Conv2D)	(None, 10, 10, 15)	4065
max_pooling2d_4 (MaxPooling2	(None, 5, 5, 15)	0
flatten_2 (Flatten)	(None, 375)	0
dense_3 (Dense)	(None, 500)	188000
dropout_2 (Dropout)	(None, 500)	0
dense_4 (Dense)	(None, 10)	5010

Figure 1: Hidden layers parameters

The pre-processing steps that were required to be conducted on the dataset were also catered to in the research work.

3.2 Testing the model

Another objective was to test the built model on unseen images from the internet and find out the rate of correct and incorrect classification i.e. rate of miss-classification.

3.3 Analyzing model using various evaluation parameters

After testing the model, further intent was to check its accuracy by using various parameters for evaluating a classification model. The aim was to calculate mean square error, root mean square error, false positive classification rate, true positive classification rate and error rate because of miss-classification. These parameters would help to judge the correctness of the model and how well it is able to label the unseen images into different classes as prescribed in the model.

4. METHODOLOGY

In order to construct a handwritten digit classifier extensive study has been done to perform a comparative analysis of various machine learning models that best perform digit classification on handwritten dataset of images. Methodology timeline starts with finalizing the dataset to be used, classification model to be used, training the model on finalized dataset and testing it on unseen images. Various evaluation parameters have been calculated based on the classification rate of the model and a close to the state of art model has been built using the above defined timeline of our methodology.

4.1 Dataset

The NIST (National Institute of Standards and Technology) has built the MNIST dataset which consists of images of handwritten digits consisting of numbers from zero to nine. The images have been collected by scanning various files, documents and handwritten captions of images which were then normalized by pixel value. The dataset is ideal for used by developers as the images it contains have been pre-processed and thus very little time and effort has to be spent on correcting the incongruities of the scanned images. The size of each image is 28*28 which means 784 square pixels in total. The dataset has been divided into 60,000 images for training the model and another 10,000 for testing it. Final testing is performed on unseen images from the internet.

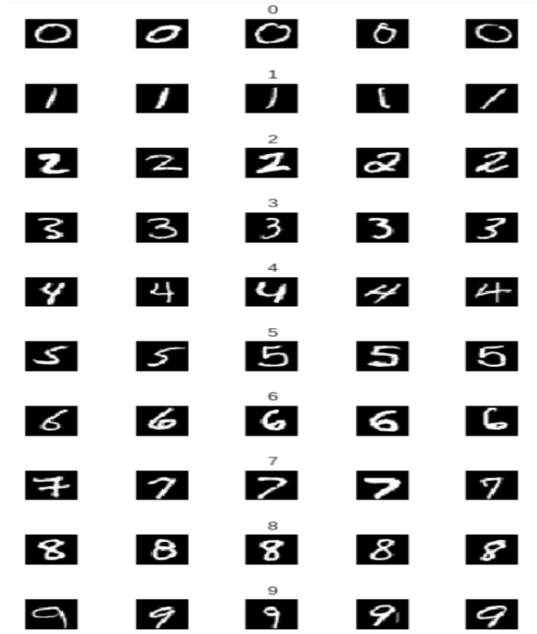


Figure 2: Sample images from MNIST dataset [12]

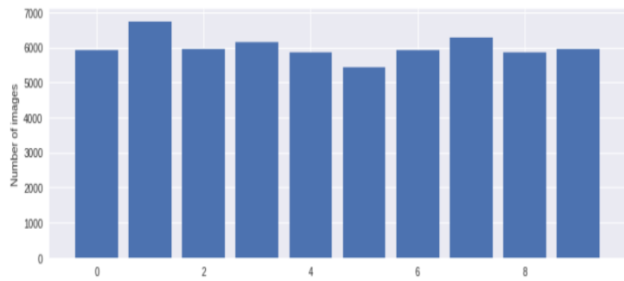


Figure 3: Distribution of training dataset

4.2 Tools and Technologies Used

The computation power required by the convolutional neural networks is quiet larger and cannot be provided by an ordinary desktop. The local software such as Anaconda even with the latest version take a significant amount of time to train the CNN model hence are not generally suited if fast and high end performance is desired from the classifier embedded in various applications. Thus an online platform famous amongst machine learning developers is GOOGLE COLAB which provides an environment similar to the one provided by a local host but with faster and more reliable procedure involved in training the model thereby overcoming the limitations of the local host applications. The above tool has been used while performing the research and training the machine learning model.

4.3 Machine Learning Model

1. **Convolutional Layer**: Convolutional Layer is a hidden layer containing a specific number of feature maps. All the feature maps in one particular layer have the same size.

2. **Pooling Layer:** This layer takes a max which is also known as Max Pooling 2D. The size of the pooling layer in the model designed is 2*2.

3. **Dropout:** Dropout layer is a regularization layer. In the designed model it is set to exclude 20% neurons so that the over-fitting is decreased.

4. **Flatten:** This layer converts data in a 2D matrix form to a data which can be processed by the next layers. The converted data is called as Flatten.

5. **Fully Connected Layer:** Data from the previous layer(Flatten) is fed to this layer which contains a specific number of neurons and Rectifier activation function is used.

6. **Output Layer**: Output layer classifies 1 class out of the 10 classes. It contains 10 neurons for the 10 classes and uses softmax activation function.

Table 2: Description of CNN layers used in training of the model

Layer	Description
Layer 1: Convolutional Layer	Number of feature maps : 30 Size of each feature map : 5 x 5
Layer 2: Pooling Layer	Size of patches : 2 x 2
Layer 3: Convolutional Layer	Number of feature maps : 15 Size of each feature map : 3 x 3
Layer 4: Pooling Layer	Size of patches : 2 x 2
Layer 5: Dropout	Probability : 20%
Layer 6: Flatten	Helps in processing output.
Layer 7: Fully Connected Layer	Number of neurons : 128 Activation Function : Rectifier
Layer 8: Fully Connected Layer	Number of neurons : 50 Activation Function : Rectifier
Layer 9 : Output Layer	Number of neurons : 10 Activation Function : Softmax

5. RESULTS AND DISCUSSIONS

As per the designed model, the accuracy peaks to 99.04% which is not as high as the state-of-the-art results. The reason the designed model was not able to achieve state-of-the-art accuracy level because for several tasks the model gets over trained. There is a way to get around if the model gets over-trained. That is, by increasing the training times and subsequently the training error will be reduced. The accuracy of the model is also affected by the size of the training set. The accuracy can be incremented by incorporating higher amount of data in the training set. This is because the impact of the values which cause the model to get over-trained gets reduced by increasing the amount of data on which the model is trained.

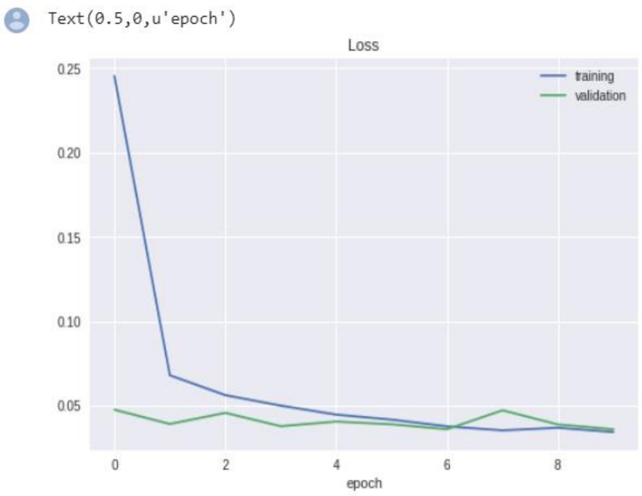


Figure 4: Training and Validation loss after 10 epochs.

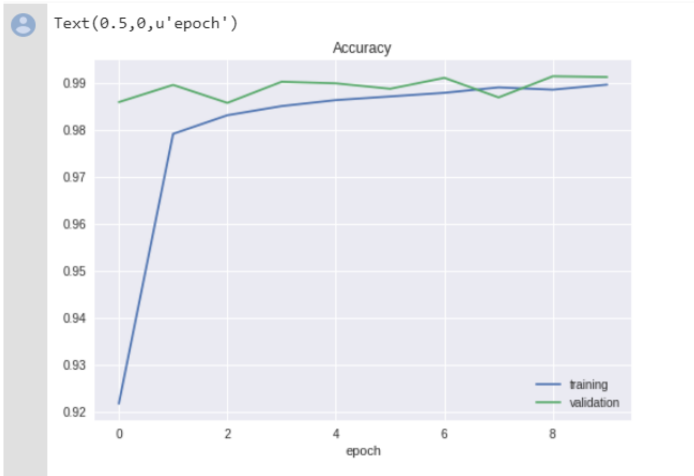


Figure 5: Accuracy after 10 epochs.

After following this approach, the accuracy of the model witnessed a sharp increase. But even after doing all the possible modifications there is no way in which the accuracy of the model can reach the magical mark of 100%. This is because some digits written by hand are too arduous to understand. Even the humans will fail to identify that digit correctly, for example, digit 6 and digit 0. In the MNIST dataset, there were quite a few inputs in which the people who intended to write digit 6 wrote it in such a way that the trailing circle of the digit 6 was very close to the upper cap of digit 6.

```
print('Test score:', score[0])
print('Test accuracy:', score[1])
```

```
('Test score:', 0.033884856743932414)
('Test accuracy:', 0.9904)
```

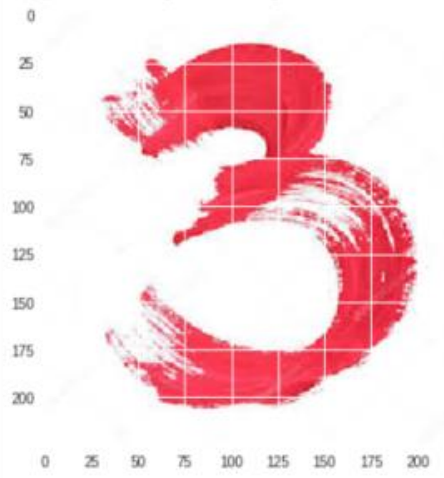
Figure 6: Results for accuracy.

The above are the accuracy results obtained. Subsequent figure shows the result of testing the model on an unseen image from the internet.

```
url = 'https://encrypted-tbn0.gstatic.com/images?q=

r = requests.get(url, stream=True)
img = Image.open(r.raw)
plt.imshow(img, cmap=plt.get_cmap('gray'))
```

```
<matplotlib.image.AxesImage at 0x7ff3566e83d0>
```




```
print(img.shape)
print("predicted digit: "+str(model.predict_classes(img)))
```

```
(1, 28, 28, 1)
predicted digit: [3]
```

Figure 7: Classification on unseen image from internet

The designed model classified such input as digit 0. To cap it all, the model achieved very satisfactory results. But considering the limitations that were set in the beginning there is a room for a lot of improvements to be made.

6. CONCLUSION AND FUTURE SCOPE

The research work performed consisted of constructing the famous handwritten digit recognizer which is also known as the prototype model of convolutional neural networks. This model is the classical example of dealing with pre-processing in image processing framework. Problems such as finding an appropriate training platform were encountered during the training phase of the work but they were overcome with the help of high end interfaces for building neural networks in python. Moreover, the handwritten digit images used in the research work are black and white images hence the model can be extended further to deal with colored images which will involve more pre-processing. The study was concentrated on the MNIST dataset to reduce the overhead involved in preparing the data for training the model as MNIST is undoubtedly an already processed form of data. In future the domain of the research can be extended to deal with raw images with different features and specifications. Handwritten digits have been the sole are of study and classification in this paper, if image classification of different animals or emotions recognition has to be done the image dataset needs to be considered from a three-dimensional point of view and advanced models of deep learning need to be applied to study the classification results thereby extending the research work in future.

7. REFERENCES

[1] Handwritten Digit Recognition using Convolutional Neural Networks in Python with Keras. https://machinelearningmastery.com/handwritten-digit-recognition-using-convolutional-neural-networks-python-keras/. Last Accessed: 2019-11-24.

[2] A. Biscoff, and P.S.P. Wang. Handwritten digit recognition using neural networks. *Intelligent Robots and Computer Vision X: Neural, Biological and 3-D Methods*, 1992.

[3] Y. Lee. Handwritten Digit Recognition using K Nearest-Neighbor, Radial-Basis Function and Back Propagation Neural Networks. *Neural Computation* 3(3):440–449, 2008.

[4] D.C. Ciresan, U. Meier, L.M. Gambardella, and J. Schmidhuber. Deep, Big, Simple Neural Nets for Handwritten Digit Recognition. *Neural Computation*, 22(12):3207–3220, 2010.

[5] F. Lauer, C.Y. Suen, and G. Bloch. A trainable feature extractor for handwritten digit recognition. *Pattern Recognition*, 40(6):1816–1824, 2007.

[6] F.T. Shah, and K. Yousaf. Handwritten Digit Recognition Using Image Processing and Neural Networks. *Proceedings of the World Congress on Engineering*, 1, 2007.

[7] R.M.O Cruz, T.I. Ren, and G.D.C Cavalcanti. Handwritten Digit Recognition Using Multiple Feature Extraction Techniques and Classifier Ensemble. *17th International Conference on Systems, Signals and Image Processing*, 2015.

[8] L.S. Oliveira, R. Sabourin, F. Bortolozzi, and C.Y. Suen. Feature Selection using multi-objective genetic algorithms for handwritten digit recognition. *Object recognition supported by user interaction for service robots*, 2002.

[9] S.B. Hallale, and G.D. Salunke. Offline Handwritten Digit Recognition Using Neural Networks. *International Journal of Advanced Research in Electrical, Electronics, and Instrumentation Engineering*, 2(9), 2013.

[10] E. Kussul, and T. Baidyk. Improved method of handwritten recognition tested on MNIST database. *Image and Visin Computing*, 22(12):971–981, 2004.

[11] H.A. Alwzwazy, H.M Albehadili, Y.S. Alwan, and N.E. Islam. Handwritten Digit Recognition Using Convolutional Neural Networks *International Journal of Advanced Research in Electrical, Electronics, and Instrumentation Engineering*, 4(2), 2016.

[12] The MNIST Database of handwritten digits. http://yann.lecun.com/exdb/mnist/. Last Accessed: 2019-11-24.