



e-ISSN: 2319-8753 | p-ISSN: 2320-6710

IJIRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 10, Issue 6, June 2021

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 7.512



9940 572 462



6381 907 438



ijirset@gmail.com



www.ijirset.com

Recognition of Handwritten Digit using CNN in Python with Tensorflow and Comparison of Performance for Various Hidden Layers

Sumita Kulkarni¹, Yugesh Raut², Shubham Shrivastava³, S.V.Wankhede⁴

U.G. Student, Department of Computer, Sinhgad Academy of Engineering, Pune, Maharashtra, India¹

U.G. Student, Department of Computer, Sinhgad Academy of Engineering, Pune, Maharashtra, India²

U.G. Student, Department of Computer, Sinhgad Academy of Engineering, Pune, Maharashtra, India³

Professor, Department of Computer, Sinhgad Academy of Engineering, Pune, Maharashtra, India⁴

ABSTRACT: In recent times, with the increase of Artificial Neural Network (ANN), deep learning has brought a dramatic twist in the field of machine learning by making it more artificially intelligent. Deep learning is remarkably used in vast ranges of fields because of its diverse range of applications such as surveillance, health, medicine, sports, robotics, drones, etc. In deep learning, Convolutional Neural Network (CNN) is at the center of spectacular advances that mixes Artificial Neural Network (ANN) and up to date deep learning strategies. It has been used broadly in pattern recognition, sentence classification, speech recognition, face recognition, text categorization, document analysis, scene, and handwritten digit recognition. The goal of this project is to observe the variation of accuracies of CNN to classify handwritten digits using various numbers of hidden layers and epochs and to make the comparison between the accuracies. For this performance evaluation of CNN, we performed our experiment using Modified National Institute of Standards and Technology (MNIST) dataset. Further, the network is trained using stochastic gradient descent and the backpropagation algorithm.

KEYWORD: Handwritten digit recognition, Convolutional Neural Network (CNN), Deep learning, MNIST dataset, Epochs, Hidden Layers, Stochastic Gradient Descent, Backpropagation

I. INTRODUCTION

With time the numbers of fields are increasing in which deep learning can be applied. In deep learning, Convolutional Neural Networking (CNN) is being used for visual imagery analyzing. Object detection, face recognition, robotics, video analysis, segmentation, pattern recognition, natural language processing, spam detection, topic categorization, regression analysis, speech recognition, image classification are some of the examples that can be done using Convolutional Neural Networking.

In handwritten recognition digits, characters are given as input. The model can be recognized by the system. A simple artificial neural network (ANN) has an input layer, an output layer and some hidden layers between the input and output layer. CNN has a very similar architecture as ANN. There are several neurons in each layer in ANN. The weighted sum of all the neurons of a layer becomes the input of a neuron of the next layer adding a biased value. In CNN the layer has three dimensions. Here all the neurons are not fully connected. Instead, every neuron in the layer is connected to the local receptive field. A cost function generates in order to train the network.

It compares the output of the network with the desired output. The signal propagates back to the system, again and again, to update the shared weights and biases in all the receptive fields to minimize the value of cost function which increases the network's performance.

II. 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. 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.

The accuracies in these fields including handwritten digits recognition using Deep Convolutional Neural Networks (CNNs) have reached human level perfection. Mammalian visual systems' biological model is the one by which the architecture of the CNN is inspired. Cells in the cat's visual cortex are sensitized to a tiny area of the visual field identified which is recognized as the receptive field [3]. It was found by D. H. Hubel et al. in 1962. The neocognitron [4], the pattern recognition model inspired by the work of D. H. Hubel et al. [5, 6] was the first computer vision. It was introduced by Fukushima in 1980. In 1998, the framework of CNNs is designed by LeCun et al. [7] which had seven layers of convolutional neural networks. It was adept in handwritten digits classification direct from pixel values of images [8]. Gradient descent and back propagation algorithm [9] is used for training the model.

III. ARCHITECTURE

The purpose of this document is to investigate the proposed system's design possibilities, such as architecture design, block diagram, sequence diagram, and data flow diagram.

The system's flow diagram and user interface design are to define stages like pre-processing, feature extraction, and so on. Digit segmentation, classification, and recognition

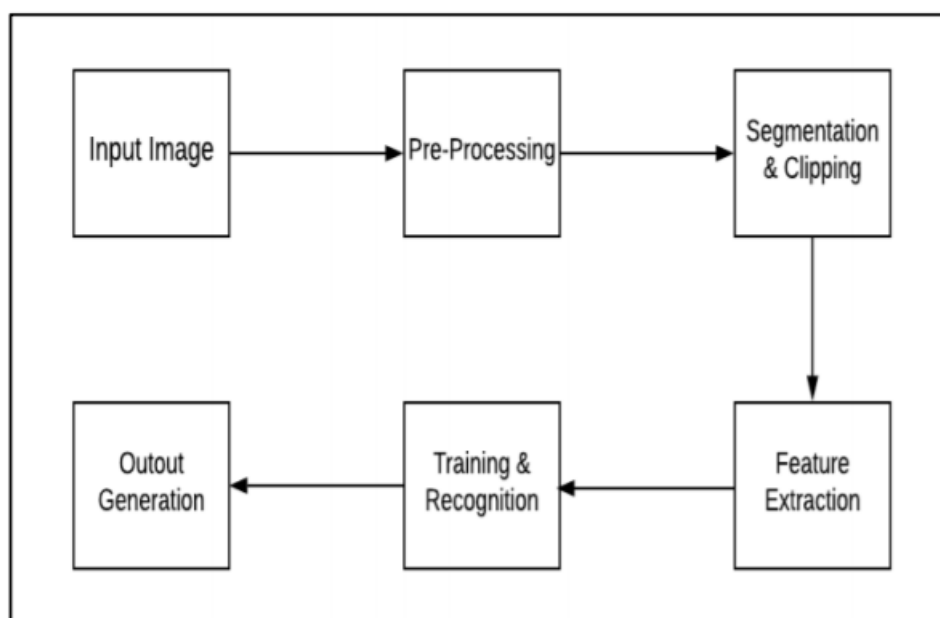


Fig 1:- Architecture of the Proposed System

The proposed system's architecture is depicted in the diagram above. In order to classify and detect the digits, the suggested model has four stages:

- A. Pre-processing
- B. Segmentation
- C. Feature Extraction
- D. Classification and Recognition

A. Pre-processing

The role of the pre-processing step is to perform 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 at this stage. The pre-processing additionally characterizes a smaller portrayal of the example. Binarization changes over a grayscale image into a binary image.



Fig 2:- Sample images taken from MNIST database

B. Segmentation

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

C. Feature Extraction

After the completion of the pre-processing stage and segmentation stage, the pre-processed images are represented in the form of a matrix that contains pixels of the images that are of very large size. 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 classifiers.

IV. METHODOLOGY

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.

Then it can classify the images using the features maps. The convolutional neural network considers the mapping of image pixels with the neighborhood space rather than having a fully connected layer of neurons. The convolutional neural network is a powerful tool in signal and image processing. Even in the fields of computer vision such as handwriting recognition, natural object classification, and segmentation, CNN has been a much better tool compared to all other previously implemented tools. The broader aim may be to develop a machine learning model that could recognize people's handwriting.

The MNIST handwritten digits database is used for the experiment. Out of 70,000 scanned images of handwritten digits from the MNIST database, 60,000 scanned images of digits are used for training the network and 10,000 scanned images of digits are used to test the network. The images that are used for training and testing the network all are the grayscale image with a size of 28×28 pixels. Character x is used to represent a training input where x is a 784-dimensional vector as the input of x is regarded as 28×28 pixels. The equivalent desired output is expressed by $y(x)$, where y is a 10-dimensional vector. The network aims is to find the convenient weights and biases so that the output of the network approximates $y(x)$ for all training inputs x as it completely depends on weight values and bias values. To compute the network performances, a cost function is defined, expressed by equation below.

$$C(w, b) = \frac{1}{2n} \sum_x [y(x) - a]^2$$

To reduce the cost $C(w, b)$ to a smaller degree as a function of weight and biases, the training algorithm has to find a set of weight and biases which cause the cost to become as small as possible. This is done using an algorithm known as gradient descent. Gradient descent is an optimization algorithm that twists its parameters iteratively to minimize a cost function to its local minimum. However, the gradient descent algorithm may be unusable when the training data size is very large. Therefore, to enhance the performance of the network, a stochastic version of the algorithm is used.

a. Stochastic Gradient Descent:

In Stochastic Gradient Descent (SDG) a small number of iteration will find effective solutions for the optimization problems. Moreover, in SDG, a small number of iteration will lead to a suitable solution. The Stochastic Gradient Descent algorithm utilizes the following equations :

$$w^{new} = w^{old} - \frac{\eta}{m} \frac{\partial C_{xj}}{\partial w^{old}}$$

$$b^{new} = b^{old} - \frac{\eta}{m} \frac{\partial C_{xj}}{\partial w^{old}}$$

The output of the network can be expressed by:

$$a = f(z) = f(wa + b)$$

To find the amount of weight that contributes to the total error of the network Backpropagation method is used.

b. Backpropagation algorithm :

In fitting a neural network, backpropagation computes the gradient of the loss function with respect to the weights of the network for a single input–output example, and does so efficiently, unlike a naive direct computation of the gradient with respect to each weight individually.

$$\begin{aligned}\delta^L &= \frac{\partial C}{\partial a^{(L)}} \frac{\partial a^{(L)}}{\partial z^{(L)}} = \frac{1}{n} (a^{(L)} - 1) f'(z^{(L)}) \\ \delta^l &= \frac{\partial C}{\partial z^{(l)}} = \frac{\partial C}{\partial z^{(l)}} \frac{\partial z^{(l+1)}}{\partial z^{(l)}} = \frac{\partial z^{(l+1)}}{\partial z^{(l)}} \delta^{l+1} = w^{l+1} \delta^{l+1} f'(z^l) \\ \frac{\partial C}{\partial b^{(l)}} &= \delta^l \\ \frac{\partial C}{\partial w^{(l)}} &= a^{l-1} \delta^l\end{aligned}$$

V. MODELING OF CNN TO CLASSIFY HANDWRITTEN DIGITS

To recognize the handwritten digits, a seven-layered convolutional neural network with one input layer followed by five hidden layers and one output layer is designed.

The input layer consists of 28 by 28 pixel images which mean that the network contains 784 neurons as input data. The input pixels are grayscale with a value 0 for a white pixel and 1 for a black pixel. Here, this model of CNN has five hidden layers. We defined a baseline convolutional neural network model for the problem. The model has two main aspects: the feature extraction front end comprised of convolutional and pooling layers, and the classifier backend that will make a prediction. For the convolutional front-end, we can start with a single convolutional layer with a small filter size (3,3) and a modest number of filters followed by a max pooling layer. The filter maps can then be flattened to provide features to the classifier. Assuming that the problem is a multi-class classification task, we know that we will require an output layer with 10 nodes in order to predict the probability distribution of an image belonging to each of the 10 classes. This will also require the use of a softmax activation function. Between the feature extractor and the output layer, we have added a dense layer to interpret the features, in this with 100 nodes. All layers will use the ReLU activation function and the He weight initialization scheme, both best practices. We have used a conservative configuration for the stochastic gradient descent optimizer with a learning rate of 0.01 and a momentum of 0.9. The categorical cross-entropy loss function will be optimized, suitable for multi-class classification, and we will monitor the classification accuracy metric, which is appropriate given we have the same number of examples in each of the 10 classes.

The model is evaluated using five-fold cross-validation. The value of $k=5$ was chosen to provide a baseline for both repeated evaluation and to not be so large as to require a long running time. Each test set will be 20% of the training dataset, or about 12,000 examples, close to the size of the actual test set for this problem. The training dataset is shuffled prior to being split, and the sample shuffling is performed each time, so that any model we evaluate will have the same train and test datasets in each fold, providing an apples-to-apples comparison between models. We have trained the baseline model for a modest 10 training epochs with a default batch size of 32 examples. The test set for each fold is used to evaluate the model both during each epoch of the training run so that we can later create learning curves and at the end of the run so that we can estimate the performance of the model. As such, we will keep track of the resulting history from each run, as well as the classification accuracy of the fold.

The model assumes that new images are grayscale, that they have been aligned so that one image contains one centered handwritten digit, and that the size of the image is square with the size 28×28 pixels.

VI. RESULT AND DISCUSSION

In this section, CNN has been applied on the MNIST dataset in order to observe the variation of accuracies for handwritten digits. The accuracies are obtained using Tensorflow in python. Training and validation accuracy for 10 different epochs were observed exchanging the hidden layers for various combinations of convolution and hidden layers. The first hidden layer is the convolutional layer which is used for the feature extraction. It consists of 32 filters with the kernel size of 3×3 pixels and the rectified linear units (ReLU) is used as an activation function to enhance the performance. The next hidden layer is a pooling layer is defined where max pooling is used with a pool size of 2×2 pixels to minimize the spatial size of the output of a convolution layer. A flatten layer is used after the pooling layer which converts the 2D filter matrix into 1D feature vector before entering into the fully connected layers. The next hidden layer used after flatten layer is the Dense layer consisting of 100 neurons and ReLU. Finally, the output layer which is used here as Dense layer contains 10 classes and determines the digits numbered from 0 to 9.

A softmax activation function is incorporated with the output layer to output digit from 0 through 9. The CNN is fit over 10 epochs. The overall validation accuracy in the performance is found at 98.68%.

VII. CONCLUSION

From the obtained simulated results, CNN has been applied on the MNIST dataset in order to observe the variation of accuracies for handwritten digits. The accuracies are obtained using Tensorflow in python. Training and validation accuracy for 3 different epochs were observed exchanging the hidden layers by taking the batch size 100 for all the cases.

Here we demonstrate a model which can recognize handwritten digit. Later it can be extended for character recognition and real-time person's handwriting. Handwritten digit recognition is the first step to the vast field of Artificial Intelligence and Computer Vision. 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 more number of hidden neurons. It can completely abolish the need for typing. Digit recognition is an excellent prototype problem for learning about neural networks and it gives a great way to develop more advanced techniques of deep learning. In future, we are planning to develop a real-time handwritten digit recognition system.

REFERENCES

- [1] Y. LeCun et al., "Backpropagation applied to handwritten zip code recognition," Neural computation, vol. 1, no. 4, pp. 541- 551, 1989.
- [2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, 2012, pp. 1097-1105.
- [3] D. Hubel and T. Wiesel, "Aberrant visual projections in the Siamese cat," The Journal of physiology, vol. 218, no. 1, pp. 33- 62, 1971.
- [4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," nature, vol. 521, no. 7553, p. 436, 2015.
- [5] D. Cireşan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," arXiv preprint arXiv:1202.2745, 2012.
- [6] K. Fukushima and S. Miyake, "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition," in Competition and cooperation in neural nets: Springer, 1982, pp. 267-285.
- [7] Y. LeCun et al., "Handwritten digit recognition with a backpropagation network," in Advances in neural information processing systems, 1990, pp. 396-404.
- [8] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradientbased learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, 1998.
- [9] R. Hecht-Nielsen, "Theory of the backpropagation neural network," in Neural networks for perception: Elsevier, 1992, pp. 65-93.
- [10] Y. LeCun, "LeNet-5, convolutional neural networks," URL: <http://yann.lecun.com/exdb/lenet>, vol. 20, 2015.
- [11] Ovi Paul, "Image Preprocessing on numtaDB for Bengali Handwritten Digit Recognition," in International Conference on Bangla Speech and Language Processing(ICBSLP), 21-22 September, 2018



- [12] W. Jiang, "MNIST-MAX: A multi-language handwritten digit recognition " in IOP SciNotes, vol 1, no. 2, arXiv:2004.03848v1 [cs.CV] 8 Apr 2020.
- [13] D. Sharma, "Isolated Handwritten Digit Recognition using adaptive unsupervised Incremental learning Technique," in International Journal of Computer Applications (0975 – 8887) Volume 7– No.4, September 2010. pp 0975 – 8887.
- [14] D. C. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification," in Twenty - Second International Joint Conference on Artificial Intelligence, 2011.
- [15] R. B. Arif, M. A. B. Siddique, M. M. R. Khan, and M. R. Oishe, "Study and Observation of the Variations of Accuracies for Handwritten Digits Recognition with Various Hidden Layers and Epochs using Convolutional Neural Network," in 2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT) , 2018, pp. 112 - 117:IEEE.



**Impact Factor:
7.512**



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com



www.ijirset.com

Scan to save the contact details