MNIST Handwritten Digits Recognition

Harshith K
UG Scholar, Department of Electronics and Communication
Dayananda Sagar University
Bangalore, India
harshithkulkarni@gmail.com

Gopinath D K
UG Scholar, Department of Electronics and Communication
Dayananda Sagar University
Bangalore, India
dkgopinath240@gmail.com

Nikhil CG
UG Scholar, Department of Electronics and Communication
Dayananda Sagar University
Bangalore, India
nikhilcg81@gmail.com

Hemanth
UG Scholar, Department of Electronics and Communication
Dayananda Sagar University
Bangalore, India
hemantham28@gmail.com

Mrs Divyashree
Assistant Professor, Department of Electronics and Communication
Dayananda Sagar University
Bangalore, India
divyashree-ece@dsu.edu.in

Abstract: - In this paper, we present an innovative method for MNIST Handwritten digits Recognition employing a Convolutional neural network. Nowadays, it's become easier to coach a neural network due to the supply of the large amount of information and various Algorithmic innovations that are happening. Nowadays the quantity of computational power needed to coach a neural network has increased because of the provision of cloud-based services like google cloud platform which provides a resource to coach a Neural network on the cloud. In our system, we are going to implement a Convolutional neural network using Keras API. we used TensorFlow for training the neural network. we've developed this technique using Python programming language.

Keywords: - Convolutional neural network, TensorFlow, Python, MNIST dataset.

I. Introduction

Handwritten digits recognition mainly identifies 0-9 of 10 characters, and category of classification is much less than optical character recognition. Optical character recognition technology includes handwritten character recognition and printed character recognition. In recent years, along with the development of computer technology and pattern recognition technology, handwritten digital recognition has been widely used in postal code, financial value identification, tax form recognition, e-commerce digital processing, and even student achievement recognition.

With time the number of fields are increasing in which deep learning can be applied. CNN is exceptionally used in deep learning for visual imagery analysing, object detection, face recognition, robotics, video analysis, segmentation, pattern recognition, natural language processing, image classification, spam detection, speech recognition, topic categorization, regression analysis, etc. In these arenas, the accuracies in these fields which includes handwritten digits recognition using Deep Convolutional Neural Network (DCNN) have reached close to the humanoid level accomplishment. Mammalian visual system biological representation is the one by which the architecture of the CNN is inspired. Within the cat's visible cortical region cells are sensitized into a tiny region of the visual arena which is known as the receptive fields. In 1962 it was discovered by D.H Hubel et al. The recognition, was the primary computer vision which is motivated by the work of D.H Hubel et al.[5]

The framework of CNNs is designed by LeCun et al.[8] which had seven layers of convolutional neural networks. It was then practiced with the handwritten digit's classification direct after values of pixel images. Gradient decent also back propagation algorithm is used for training the model. In handwritten recognition digits, characters are given as input. The model can be recognized by the system. A simple artificial neural network consists of an input layer, output layer and several hidden layers among the input and output layers. The architecture of CNN is comparable to the architecture of ANN. In every layer of ANN, quite a lot of neurons are present. The sum of all the weighted neurons in the layer converted into the input of a neuron in the next layer by the accumulation of biased value. The layers of the CNN comprise of three dimensions where all the neurons are not fully connected to the local receptive field. In order to train the network, a cost function is generated which relates the output of the network with the desired output. To minimize the rate the of the cost function, the signal propagates back to the system, again and again, to appraise the shared weights as well as biases in all the receptive fields. The Key objective of this paper is to develop a web-based application software which can recognize the handwritten digit that is given as input image to our software. And also to develop supervised learning software using Convolution neural network to recognized hand written digits. To implement Convolutional neural network, we used Kera API.

II. Literature Survey

CNN plays an important role in many sectors like image processing. It has powerful impact on many fields. Even in nanotechnologies like manufacturing semiconductors. There are large number of papers and articles published on this topic. In research it is shown that CNN gives the highest accuracy in comparison with the most widely used machine learning algorithms like SVM, KNN and RFC. Because of it high accuracy, CNN is widely used for the purpose of classification of the image, video analysis etc. CNN is being used in natural language processing by varying different parameter.[10] It is very challenging to get a good performance as more parameters are needed for the large-scale neural network. Macy research are trying to increase the accuracy in CNN with less error. In some research they shown that deep nets perform better when they are trained by simple back-propagation. Their architecture results in the lowest error rate on MNIST[4]. Researchers are working on this issue to reduce the error rate as much as possible in handwritten recognition. Deep CNN Can be adjustable with the input image noise. Coherence recurrent convolutional network is a multi-model neutral architecture. It is being used in recovering sentence in an image. Some researchers are trying to come up with new techniques to avoid drawbacks of traditional convolutional layers. No combination of feature maps is a method which can be applied for better performance using MNIST datasets. Its accuracy is 99.81% and it can be applied for large scale date[1]. New applications of CNN are developing day by day with many kinds of research. Researchers are trying hard to minimize error rates. To clean blur images CNN is being used. For this purpose, a new model was proposed using MNIST dataset. This approach reaches an accuracy of 98% and loss range 0.1% to 8.5%.[10] In Germany, a traffic sign recognition model of CNN is suggested. Its proposed a faster performance with 99.65% accuracy.

III. **Modelling of Convolutional Neural Network 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 layers is designed. It is illustrated in fig 1

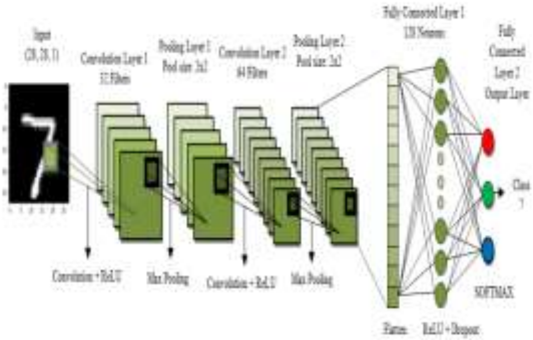
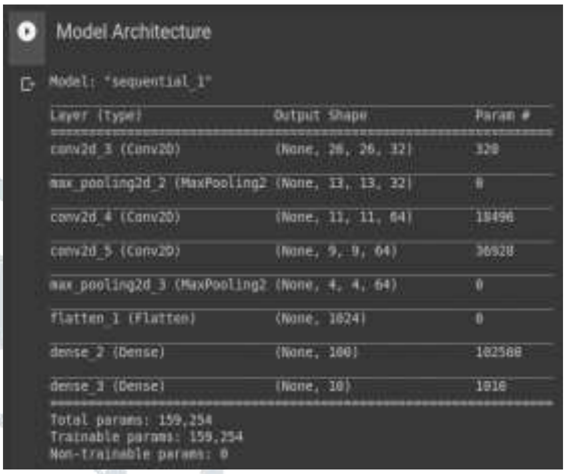


Fig 1



Model Architecture

Model: "sequential_1"

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 26, 26, 32)	320
max_pooling2d_2 (MaxPooling2	(None, 13, 13, 32)	0
conv2d_4 (Conv2D)	(None, 11, 11, 64)	18496
conv2d_5 (Conv2D)	(None, 9, 9, 64)	36928
max_pooling2d_3 (MaxPooling2	(None, 4, 4, 64)	0
flatten_1 (Flatten)	(None, 1024)	0
dense_2 (Dense)	(None, 100)	102500
dense_3 (Dense)	(None, 10)	1010

Total params: 159,254
Trainable params: 159,254
Non-trainable params: 0

Model Architecture

The input layer consists of 28 x 28-pixel images which mean that the network contains 784 neurons as input data.[3] The input pixel is grayscale with a value 0 for a white pixel and 1 for a black pixel. Here, this model of CNN has five hidden layers. The first hidden layers is the convolution layer 1 is responsible for feature extraction from an input. This layer performs convolution operation to small localized areas by convolving a filter with the previous layer. In addition, it consists of multiple feature maps with learnable kernels and rectified linear units (ReLU). The kernel size determines the locality of the filters. ReLU is used as an activation function at the end of each convolution layer as well as a fully connected layers to enhance the performance of the model[4]. The next hidden layers is the pooling layer 1. It reduces the output information from the convolutional layers and reduces the number of parameters and computational complexity of the model. The different types of pooling are max pooling, min pooling, average pooling, and L2 pooling. Here, max pooling is used to subsample the dimension of each feature map. Convolution layer 2 and pooling layer 2 which has the same function as convolution layer 1 and pooling layer 1 and operates in the same way except for their feature maps and kernel size varies[6]. A Flatten layer is used after the pooling layer which converts the 2D featured map matrix to a 1D feature vector and allows the output to get handled by the fully connected layers. A fully connected layer is another hidden layer also known as the dense layer. It is similar to the hidden layer of Artificial Neural Networks (ANNs) but here it is fully connected and connects every neuron from the previous layer to the next layer. In order to reduce overfitting, dropout regularization method is used at fully connected layer 1. It randomly switches off some neurons during training to improve the performance of the network by making it more robust. This causes the network to become capable of better generalization and less compelling to overfit the training data. The output layer of the network consists of ten neurons and determines the digits numbered from 0 to 9. Since the output layer uses an activation function such as softmax, which is used to enhance the performance of the model, classifies the output digit from 0 through 9 which has the highest activation value.

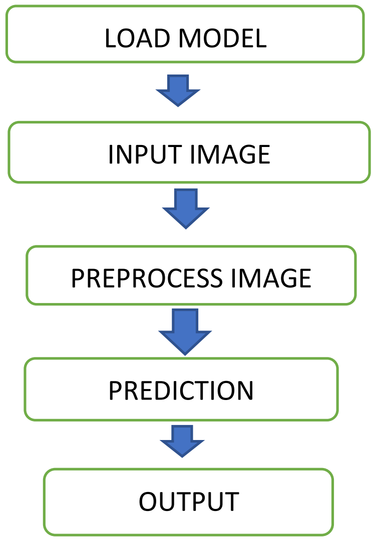
IV. MNIST Dataset

Modified National Institute of Standards and Technology (MNIST) is a large set of computer vision dataset which is extensively used for training and testing different systems. It was created from the two special datasets of National Institute of Standards and Technology (NIST) which holds binary images of handwritten digits. The training set contains handwritten digits from 250 people, among them 50% training dataset was employees from the Census Bureau and the rest of it was from high school students.[1] However, it is often attributed as the first datasets among other datasets to prove the effectiveness of the neural networks.

Fig. 2

The database contains 60,000 images used for training as well as few of them can be used for cross-validation purposes and 10,000 images used for testing [2]. All the digits 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 forms an array which can be flattened into 28*28=784-dimensional vector. Each component of the vector is a binary value which describes the intensity of the pixel.

V. Flow Chart



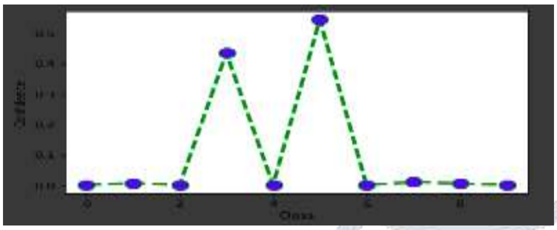
In This application first we impot all the required modules to train the model. Those modules include Tensor flow, Keras, Scikit-learn. Since we are already importing Keras, we don't require to import MINST dataset separately. After importing all the modules, we load the image into the model as input. Since Input image cannot be loaded into the model, so we need to perform certain operation to make it ready for our neural network. In this operation it actually upgrades the image by making segmentation. Next stage is Prediction, once pre-processing of input image is done. Those pre-processed image are divided into Pixels so it model can easily predict the Image and gives the accurate output.

VI. Conclusion

In this paper, the variation of accuracies for handwritten digit were observed for 15 epochs by varying the hidden layers. The accuracy curves were generated for the six cases for the different parameter using CNN MNIST digit dataset. The six cases perform differently because of the various combinations of hidden layers. The layers were taken randomly in a periodic sequence so that each case behaves differently during the experiment. The maximum and minimum accuracies were observed for different hidden layers variation with a batch size of 100. Among all the observation, the maximum accuracy in the performance was found 99.21% for 15 epochs in case 2 (Conv1, pool1, Conv2, pool2 with 2 dropouts). In digit recognition, this type of higher accuracy will cooperate to speed up the performance of the machine more adequately. However, the minimum accuracy among all observation in the performance was found 97.07% in case 6 (Conv1, pool1, Conv2, pool2 with 1 dropout) [8]. Moreover, among all the cases, the total highest test loss is approximately 0.049449 found in case 3 without dropout and the total lowest test loss is approximately 0.026303 found in case 2 with dropout. This low loss will provide CNN better performance to attain better image resolution and noise processing. In the future, we plan to observe the variation in the overall classification accuracy by varying the number of hidden layers and batch size.

VII. Result

We havre achieved a 96% accuracy in predicting the digit drawn on a HTML canvas



developed with the platform support of Google colaboratory.

In the above image, the number "5" is drawn using mouse on a HTML canvas

And the drawn number is predicted as "5" accurately.

VI. Reference

[1] Fathma Siddique, Shadman Sakib, Md Abu Bakr Siddique "Recognition of Handwritten Digits using Convolutional Neural Network in Python with TensorFlow and comparison of performance for various hidden layers."

[2] Shefali Arora, M.P.S Bhatia "Handwriting recognition using Deep Learning in Keras"

[3] Yawei Hos, Huailin Zhao " Handwritten Digit Recognition Based on Depth Neural Network"

[4] 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.

[5] D. Hubel and T. Wiesel, "Aberrant visual projections in the Siamese cat," *The Journal of physiology*, vol. 218, no. 1, pp. 33-62, 1971.

[6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[7] LeCun, "The MNIST database of handwritten digits," *http://yann. lecun. com/exdb/mnist/*, 1998.

[8] Y. LeCun *et al.*, "Handwritten digit recognition with a back-propagation network," in *Advances in neural information processing systems*, 1990, pp. 396-404.

[9] 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.

[10] K. G. Pasi and S. R. Naik, "Effect of parameter variations on accuracy of Convolutional Neural Network," in *2016 International Conference on Computing, Analytics and Security Trends (CAST)*, 2016, pp. 398-403: IEEE.