



Handwritten Digit Recognition System

¹Radhika Ambavkar, ²Niralee Kolge, ³Poorva Dalvi, ⁴Swati Powar

¹Student, ²Student, ³Student, ⁴ Assistant Professor

¹Department of Information Technology,

¹Finolex Academy of Management and Technology , Ratnagiri, India

Abstract: Handwritten Digit Recognition System offers a platform to recognize digits manually which is handwritten. There is no computer which can compete the level of efficiency of human brain. These inefficiencies of the computer have led to evolution of "Neural Network". Digit recognition system is a way for a machine to identify digits from numerous sources such as photographs, documents, e-mails, checks, cellular phones, and so on. It brings ease in digit recognition and thus improves the customer satisfaction. This study presents the structure of our system in depth, as our suggested method resolves the drawbacks of previous systems.

Keywords – Handwritten Digit Recognition, MNIST Dataset, Convolutional neural network.

I. INTRODUCTION

Machine learning is a method which automates analytical model building. There are several methods in machine learning. From that human efforts can be reduced to recognizing hand written digits. There are many algorithms of Machine learning are used for building a required application or a system. Linear Regression Logistic Regression, CNN, SVM, Naive Bayes, KNN, Random Forest, etc. CNN has become popular the recent times. CNN is part of deep learning. As a result, we chose CNN for the numeric identification task in authentic situations such as online handwritten recognition on a desktop or smartphone to detect vehicle licence plates, analyze check payments, and digits in an application data written by hand, among other things.

The term "recognition" refers to the act of distinguishing or locating something, as well as the application of prior information or understanding. CNN is consists of deep artificial neural network that could really execute a multitude of activities faster and more accurately than those of other algorithms in image and video recognition applications.

II. LITERATURE SURVEY

In the paper publish by Ankit Sharma, Yogiraj Barole, Kaustubh Kerhalkar, Dr. Prabhu K. R , they trained the Neural Network to recognize the off-line handwritten character (0-9). In this paper they used the linear regression algorithm to classify the characters and also to train the given dataset. A network performance curve was developed after training the network. From that they were able to get approximately 90% accuracy in results. [1]

"Handwritten Digit Recognition using Deep Learning" by Anuj Dutt and Aashi Dutt. In this paper they used 4 different classifiers. Machine Learning algorithm are RFC, SVM, KNN and with Deep Learning Algorithm like CNN (with some libraries that are Keras, Theano and Tensorflow). They compared the results of these classifiers. From this, they were able to get accuracy of 98.70%-CNN as compared to 97.91%-SVM , 96.67%-KNN , 96.89%-RFC [2].

III. DATASET

We have the dataset in order to train and generalize the solution. In this case we are using MNIST dataset which contains 70000 images of 28 x 28 pixels written by different people in different styles. The dataset is partitioned into two sets. The first set contains 60,000 examples used for training, and the second set of 10,000 examples used for testing part. MNIST database is very helpful for the all who want to learn techniques and pattern recognition methods on real-time data with minimum efforts on preprocessing and formatting.

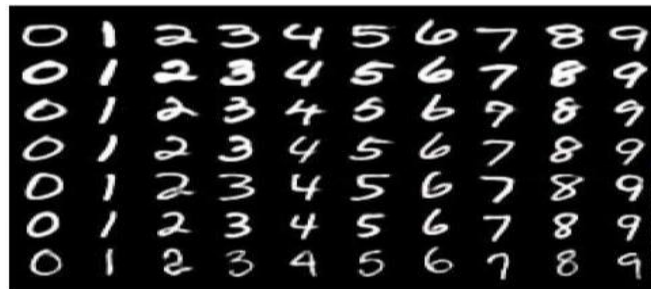


Fig1. MNIST dataset

IV. MODELLING OF CNN TO CLASSIFY HANDWRITTEN DIGITS

We need to classify image which category it belongs to. These days we use deep learning architecture which automatically detect everything by itself. Convolutional Neural Network used to recognize, image classification, object detection etc. Our goal to design the concept of Convolutional Neural Network for a handwritten digit recognition. CNN is good for image classification. CNN is a special type of neural network that used in image recognition and processing that is specifically designed to process pixel data. We use CNN to train and test our handwritten digits.

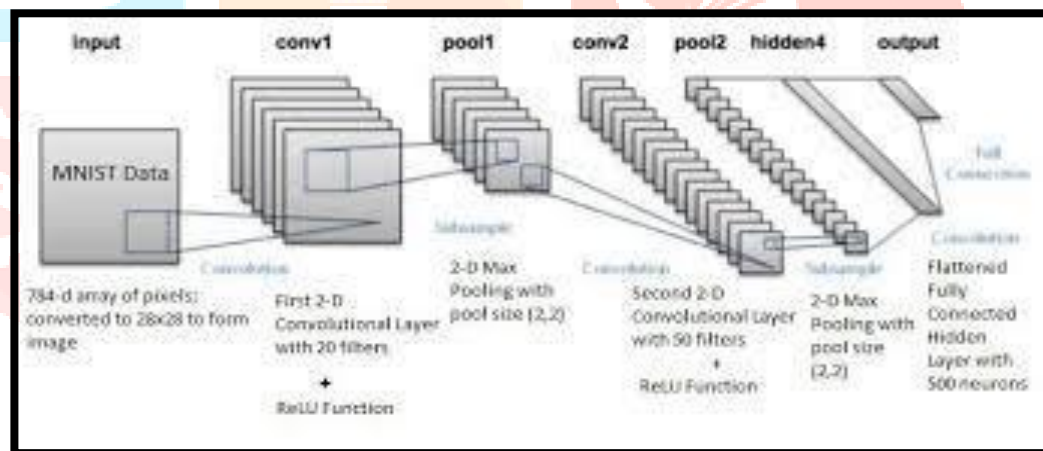


Fig2. CNN Model

A. THE ARCHITECTURE OF THE MODEL

The model takes an image as an input. Then process the image and classifies it under a specific category. In the CNN model the training and testing of the input image is performed by passing the image through a series of convolutional layers with filters. The filters are called as kernels. For all the operations of Convolutional Neural Network the convolutional layers, pooling layers and fully connected layers are the most important. The first layer is the convolutional layer. This layer is used to extract the various features from input image. This layer ensures the spatial relationship between pixels by learning image features using small squares of input data. It's simply a mathematical operation that takes two inputs such as image matrix and filter.

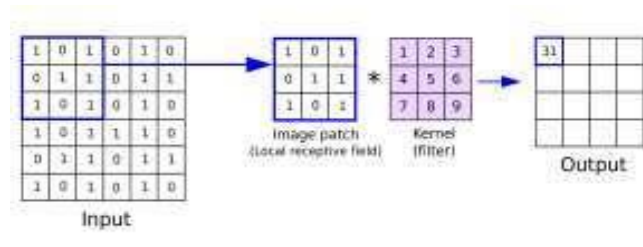


Fig3. Convolutional Layer

Example, When we apply convolutional to the 5 x 5 image by using a 3 x 3 filter with stride value 1 (When stride value is 1, the filter is moved one pixel at a time) finally producing an output features map of 3 x 3 x 1 matrix. The second layer is pooling layer. There are different types of pooling. Max pooling, min pooling, and average pooling. To reduce the number of parameters Pooling is used, when the images are large. Max pooling – In these largest elements are extracted as subsampling.

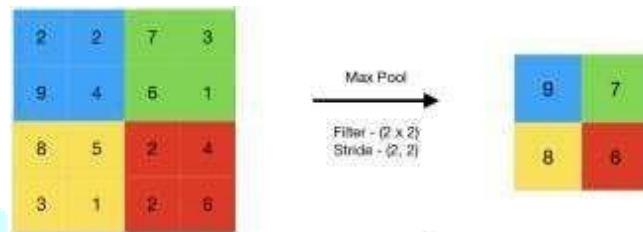


Fig4. Max Pooling

The third layer is fully connected layer. Each CNN architecture is provided with a fully connected layer. It is typical neural network which is connected to all neuron. It is used for final classification result

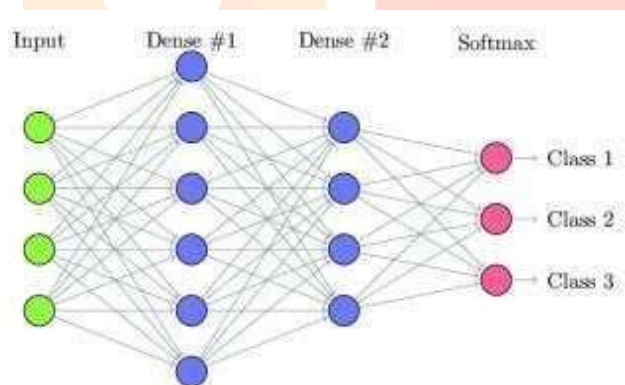


Fig5. Fully Connected Layer

B.IMPLEMENTATION OF MODEL

A simple convolutional neural network is nothing but a sequence of layer. The type of keras model which we have used to build the model is 'sequential'. It helps to build the model consecutively. First of all, we need some pre-processing the images like resizing images, normalization etc. to make it suitable to apply convolutional operation. Layer 1 consist of convolutional layer the number of kernel or filter are 64 and filter has 3x3 size of Relu activation function. This function returns 0 if the input is negative. Relu is used to enhance the performance of the model. The input shape is (28.28.1) which signifies that each image has pixel dimension of 28 X28 and the '1' defines grayscale image.

The next layer is max pooling layer which is of size 2 X 2. This layer help to extract the main features from the previous convolution layer. We have total 3 convolutional layers. In the 2nd & 3rd convolutional we are using 64 different kernels and each will have 3x3 different matrix size followed by another pooling layer. Next layer is fully connected layer 1, in that we use flatten layer, which convert the data from 2D to 1D vector. It has Relu activation function. Next is dense layer which neural network because all neurons are connected. The next layer is fully connected.

Layer 2, this layer consists of dense & Relu activation function to reduce size to reach 10 classes. The last Layer fully connected layer consist dense layer. This last dense layer must be equal to 10. Activation Function changes to softmax. As we have multiple classes like 10 in this. Softmax is used for getting the outputs in the form of probabilities. Then compile the model parameter 'optimizer' set to " adam " as it is very good optimizer. Loss function is set to "sparse_categorical_crossentropy". For improved accuracy target matrix is set to be "accuracy". The next step is training the model, model.fit() . Data is x_train & label is y_train used 5 epochs for training. Using the testing data we evaluate our model.

V. RESULTS

We take single handwritten digit to check performance of our model. The images were hand drawn on Microsoft Paint and saved in the format of PNG. The image is of "seven" number.



VI. CONCLUSION

In this paper, a system is proposed for creation of handwritten digit recognition system and recognize manually written digit by using this system. The purpose of our project is to introduce neural networks through a relatively easy-to-understand application to the general public. The main goal of our project was to create a model that can recognize and determine the handwritten digits from the image by using the concepts of Convolution Neural Network.

REFERENCES

- [1] Neural Network Based Handwritten Digit Recognition for Managing Examination Score in Paper Based Test, By Ankit Sharma, YogirajBarole, Kaustubh Kerhalkar, Dr. Prabhu K.R, vol. 5, Issue 3, March 2016
- [2] Dutta and A. Dutta, Handwritten digit recognition using deep learning, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), vol. 6, no. 7, July 2017.
- [3] <https://www.geeksforgeeks.org/introduction-convolutionneural-network/>
- [4] Handwritten Digit Recognition using CNN, By Vijayalaxmi R Rudraswamimath, Bhavanishankar K, Volume 4, Issue 6, June – 2019.
- [5] https://en.wikipedia.org/wiki/MNIST_database.
- [6] <https://www.upgrad.com/blog/basic-cnn-architecture/>
- [7] <https://towardsdatascience.com/types-of-convolution-kernels-simplified-f040cb307c37>
- [8] <https://www.kaggle.com/dansbecker/rectified-linear-units-relu-in-deep-learning>
- [9] <https://www.analyticsvidhya.com/blog/2020/01/fundamentals-deep-learning-activation-Functions-when-to-use-them/>

[10] <https://machinelearningmastery.com/softmaxactivationfunctionwithpython/#:~:text=The%20softmax%20function%20is%20used%20as%20the%20activation%20function%20in,more%20than%20two%20class%20labels.>

[11] <https://www.quora.com/What-is-the-meaning-of-flattening-step-in-a-convolutional-neural-network>

