e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

HAND DRAWN CHARACTER RECOGNITION USING CONVOLUTIONAL NEURAL NETWORK

Prof. Prachi V. Kale*1, Abhijit D. Shahane*2, Vidhesh S. Sawarkar*3, Gauri S. Tayde*4, Vaishnavi V. Ingole*5, Nikita S. Khandelwal*6

*1,2,3,4,5,6Computer Science And Department Engineering, PRPCEM, Amravati.

ABSTRACT

The living beings with vision and intelligence can visualize and sense different surrounding things as well as text patterns and can make decisions according to the same. But for making computer systems that much capable of performing visualization and decision making, some special ways are being used that are known as algorithms. Convolutional Neural Network is a special algorithm which makes computer systems capable of performing various intelligent tasks. The goal of this paper is to build a CNN model that will be able to extract and recognize the character from the image provided to the model with less complexity and better accuracy. We intent to complete this by using CNN along with MNIST Dataset and other python libraries which are needful. Though the aim is to build a model which can recognize a single character based result, we also aim to enhance its functionality for a sentence recognition and then towards person's handwriting. Throughout this work, we aim to learn CNN and other things and apply it as a logic on our model.

Keywords: Convolutional Neural Network, MNIST, RELU.

I. INTRODUCTION

The problem of handwritten characters acknowledgement has been broadly considered as every person may have different hand writings. From which most of the handwritings can be read by the reader or most of the handwritings are even not properly visible to the reader and the reader may need to make assumptions on that written lines which most probably go wrong. For making those assumptions, reader uses his/her own intelligence to predict the character. The primary issue of manual prediction of handwritten characters is sudden changes in size, stroke thickness, interpretation and sudden twisting of characters. To convert this manual prediction process into automatic or computer system based evaluation, one should provide some sort of intelligence to the system. This can be implemented using machine learning techniques. Convolutional Neural Network is a special algorithm which makes computer systems capable of performing various intelligent tasks. CNN along with MNIST dataset and some python programming libraries can be used together to build a model which can predict hand drawn character on canvas.

Pham et al. author reported improvement in RNN performance with significant reduction in the character error rate (CER) and word error rate (WER).

a) Convolutional Neural Networks:

CNN is a specific type of deep neural network having wide applications in image classification, object recognition, recommendation systems, signal processing, natural language processing, computer vision, and face recognition. The reason CNN is preferred is it can detect features of an object without any human supervision. Hence Hierarchical feature learning is more efficient which makes CNN highly efficient.

b) MNIST dataset:

Each row represents a single image of shape -> (1,784) There are a whole 784 columns. The first column contains the labels for each image and 10 Labels in map_images represent digits(0->9). The images and their labels will be retrieved by us. Then resize the images to (28,28) as all images should have same size for recognition. Then transform the images into numpy array. This papers work shows the exact proceedings of obtaining system generated results with help of MNIST and CNN for the hand drawn object image passed as an input with better accuracy.

Proposed System:

- Handwritten digit recognition plays an important role in today's digital world in information processing. plenty of information is obtainable on paper and it is more inexpensive to process digital files than traditional paper files. Converting handwritten characters into machine readable form is the main purpose

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

of handwritten digit recognition. Applications of handwritten digit recognition includes postal letter sorting, license plate reader for parking structures, automation of old documents in banks and libraries, etc All these field mentioned above functions with large database and that is why they need high recognition accuracy , minimum computational complexity and accordant performance of the system. It has been proposed that shallow neural architectures are not as beneficial as deep neural architectures.

- Now we present a CNN, which is distinct type of deep neural network which has many uses in object recognition, image classification, signal processing, face recognition and computer vision. which is a specific type of deep neural network having wide applications in image classification, object recognition, recommendation systems, signal processing,
- Natural language processing, computer vision, and face recognition. For Computer vision tasks CNN is a golden Standard since it does the feature extraction for us. In initial days of computer vision, people used to manually, extract features and implement to do image classification task.

Advantages Of Proposed System:

- Accuracy level is very high when compared to the existing system
- It is very time saving
- It is user friendly
- Besides the word and character error rates, another measure to consider when evaluating a recognition engine is the decoding time, since a high value might diminish the usability of the recognition system in practical applications. Our proposed system results are very fast and accurate.

II. DATA SOURCE MODULE

Neural organization and regular neural organization are the best example of some AI issue. It has been utilized by analysts for speculations of AI calculations as trials for a long time. Mixture of CNN-SVM model for manually written digit acknowledgement is planned. This epic mixture creates the expectations and the crude model pictures are highlights and extricates consequently. The utilization of non-immersing neurons and the convolution activity for an extremely productive GPU execution to decrease overfitting in associated layers.[3]The creator represents strategy from a novel that gives knowledge into the capacity of layers and the methodology of the classifier have watched convolutional net design that can be utilized in any event, when the measure of learning information is restricted.[4]The creator utilized new structure, called Spatial Pyramid Pooling SPP-net, and it can create a fixed-length portrayal paying little mind to picture. MNIST digits are utlized by Multi- segment DNN (MCDNN).The outcome has a low 0.23%blunder rate. Hayder M. Albeahdili et al. have played out another CNN engineering which accomplishes cutting edge arrangement results on the distinctive test benchmarks. The blunder rate for this methodology is 0.39 % for MNIST dataset.

Digits 0-9 are introduced here and contains dataset about 70,000 images. These are further divided into two parts:-

i. e:-

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

- Training set data
- Testing set data.

The training dataset contains about 60,000 handwritten digits and the test dataset contains about 10,000 handwritten digits. Pre-processing the dataset. To ease our processing we will change some parameters such as colour of the images, size of the images. Next step will be to build the model that will help us in prediction. And here we introduced the part of CNN. The model is based on CNN and all the layers of CNN. After building the model it's time for training and testing the model and also check the accuracy that it's providing in the validation set. At the end, it's time for evaluation and prediction. We will evaluate the model and then start with our prediction.

III. DIGITAL IDENTIFICATION

MODULE

a) Pre-Processing

Whenever a predictive model is built, we have to consider the manipulation of data like importing images, changing the size of images , visualising data of images ,changing colour of images etc, before we start modelling it .And combing all these steps under one thing is called Exploratory Data Analysis. The reason to perform these steps is to reduce complexity of the model and ease computing speed.

b) Principle of Convolutional neural network

After the pre-processing is done, we move to the creation of CNN model. CNN i.e. Convolutional Neural Network has 4 hidden layers which extracts the feature from the images and helps to predict the result. The CNN layers Consist of Convolutional layer, ReLu Layer, Pooling Layer, Fully Connected Layer. The reason behind using CNN is it recognizes the highlights of a image without any human supervision.

Convolutional Layer

Convolutional layer is the first layer of CNN which extracts various inputs from the input image. Here the operation of convolution is performed between the input image and a filter of a particular size. The output we get is in the form of feature map which gives information about image like corners and edges. Later, this feature map is fed to other layers to learn several other features of the input image.

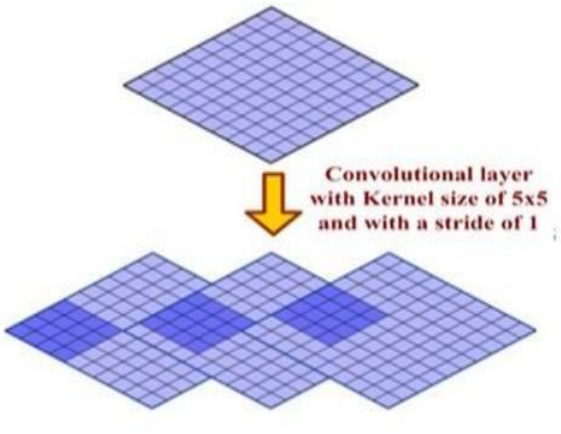


Fig. 3 Representation of Kernel and Stride in a Convolutional Layer

Relu Layer

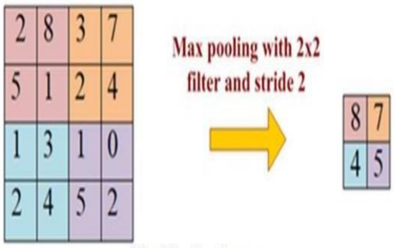
So Relu does nothing but simply removes all the negative pixel vales from the image and replace them with zero. We do this to avoid summing up of pixel values to zero.

$$R(z) = \begin{Bmatrix} z & z > 0 \\ 0 & z <= 0 \end{Bmatrix} \quad (1)$$

Pooling Layer

Pooling layer is used to reduce the dimension of the image so that the computational cost is decreased and computational speed is improved.

So this layer takes a 2 x 2 matrix and stride of one and move window all over the image. Highest value is taken in each window and the same process is carried out on each part of the image. For example if we had a matrix of 4 x 4 before pooling then the image matrix will reduce to 2 x 2 after pooling layer.



Fir. 4 Pooling Layer

Fully connected Layer

In this layer the actual classification happens. Here the matrix obtained form the pooling layer are stacked up and the put into a single list. The higher values are the points of prediction for given image.

Building Model

```
In [13]: from keras.models import Sequential
         from keras.layers import Dense, Dropout, Flatten
         from keras.layers import Conv2D, MaxPooling2D

In [14]: batch_size = 200
         num_classes = 10
         epochs = 5

         model = Sequential()
         model.add(Conv2D(32, kernel_size=(3, 3),activation='relu',input_shape=(28, 28, 1)))
         model.add(Conv2D(64, (3,3), activation='relu'))
         model.add(MaxPooling2D(pool_size=(2, 2)))
         model.add(Dropout(0.25))
         model.add(Flatten())
         model.add(Dense(256, activation='relu'))
         model.add(Dropout(0.5))
         model.add(Dense(num_classes, activation='softmax'))

         model.compile(loss=keras.losses.categorical_crossentropy,optimizer=keras.optimizers.Adadelta(),metrics=
```

Fig. 5 CNN Model

Classification of Recognition results

After the whole model is built the next step is evaluating the model. Here we check if the model works as we expected it to work or not .But the model is trained first on the evaluation dataset.

```
In [15]: final=model.fit(x_train,y_train,batch_size=batch_size,epochs=epochs,verbose=1,validation_data=(x_test,y
         print("Model is successfull.")
         model.save("MNIST digit recognizer.")
         print("Model is saved.")

         Train on 60000 samples, validate on 10000 samples
         Epoch 1/5
         60000/60000 [==============================] - 212s 4ms/step - loss: 0.2872 - accuracy: 0.9111 - val_
         loss: 0.0586 - val_accuracy: 0.9818
         Epoch 2/5
         60000/60000 [==============================] - 191s 3ms/step - loss: 0.0808 - accuracy: 0.9749 - val_
         loss: 0.0425 - val_accuracy: 0.9859
         Epoch 3/5
         60000/60000 [==============================] - 184s 3ms/step - loss: 0.0570 - accuracy: 0.9829 - val_
         loss: 0.0323 - val_accuracy: 0.9898
         Epoch 4/5
         60000/60000 [==============================] - 193s 3ms/step - loss: 0.0457 - accuracy: 0.9862 - val_
         loss: 0.0304 - val_accuracy: 0.9889
         Epoch 5/5
         60000/60000 [==============================] - 192s 3ms/step - loss: 0.0382 - accuracy: 0.9880 - val_
         loss: 0.0271 - val_accuracy: 0.9917
         Model is successfull.
         Model is saved.
```

Fig. 6 Training and Testing of CNN Model

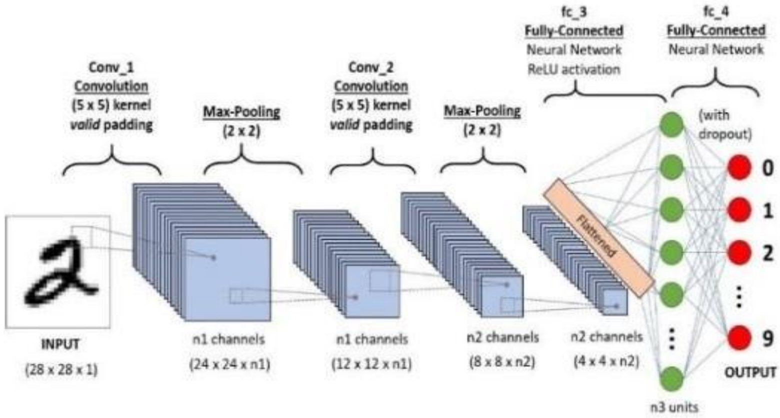


Fig: Hand Drawn Character Recognition system using CNN

Layer-1 consists of a convolutional layer with ReLu (Rectified Linear Unit) activation function. It is the first convolutional layer of the CNN architecture. This layer gets the pre-processed image as the input of size n*n=28*28.

Layer-2 is the max pooling layer. This layer gets the input of size 32@24*24 from the previous layer. The pooling size is 2*2; padding is 0 and stride is 2.

Layer-3 is the second convolutional layer with ReLu activation function. This layer gets the input of size 32@12*12 from the previous layer. The filter size is 5*5; padding is 0, the stride is 1 and the number of filters is 32.

Layer-4 is the second max pooling layer. This layer gets the input of size 32@8*8 from the previous layer. The pooling size is 2*2; padding is 0 and stride is After this max pooling operation, we get a feature map of size 32@4*4.

Layer-5 is the third convolutional layer without ReLu activation function. This layer gets the input of size 32@4*4 from the previous layer. The filter size is 4*4; padding is 0, the stride is 1 and the number of filters is 64.

Layer-6 is the fully connected layer. This layer takes one-dimensional vector of size 64 as input and provides a one-dimensional vector of size 256 as an output. It has ReLu activation function.

Layer-7 is the last layer of the network. It is also fully connected layer. This layer will compute the result, such as among the ten categories of MNIST dataset. It has some activation function named softmax activation function for final outputs.

In particular, the convolution / fully connected layers perform transformations that are a function of not only the activations in the input volume but also of the parameters (the weights and biases of the neurons). On the other hand, a fixed function will be implemented by ReLu / pooling layers.

PREDICTION

In this module of the project, we are going to test whether our model predicts correct characters. We give our data in the form of image to the model and by using CNN's image processing it will predict which number or alphabets are drawn with the help of cursor.

As we move cursor in the black part of the UI it will be convert that black part into white one. Now this input is taken as form and converting it into string array format. Then with the help of Numpy module first we will be converting it into binary and then into unsigned long. Now with the help of open CV we will be converting it into color image format. But the color image is cannot be process by Keras, so that's why we will be converting it into grey image. To get more accurate result we will be converting it into 28x28 pixels because CNN will be more accurate and efficient with the 28x28 pixel size.

IRJMETS
e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

Fig: Input

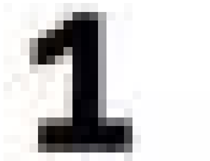


Fig: Output

What I have done here is created some digit images on MS PAINT. I have created images of digits 0-9 on MS PAINT. Keeping the background black and digit with white to keep the model working. Surprisingly, I was shocked with the results. The accuracy was up to the mark. The model predicted all the images correctly with an accuracy of 100%.

IV. RESULTS

The test after effects of the MNIST transcribed digit dataset utilizing various boundaries of CNN models are recorded and broke down the discoveries of approval affirm the part of various engineering boundaries on the presentation of our acknowledgment framework. The preparation boundary utilized here with a learning pace of 0.02 and epochs of 5. The most elevated acknowledgment exactness accomplished is with CNN design consisting of four layers, is 99.16% for the component. The goal of the proposed work is to completely explore all the boundaries of CNN design that convey best acknowledgment exactness for a MNIST dataset. Generally speaking, it has been watched that the proposed model of CNN design with three layers conveyed better acknowledgment exactness of 99.16% with the streamlining agent.

V. CONCLUSION

In above project with the help of machine learning, convolutional neural network and a data set with the thousands of data which is helpful for training and testing our CNN model we achieve 87% of accuracy which will help us to get accurate results and increase the efficiency of the project. As CNN is the part of deep learning, like other deep learning module if we give CNN a clean and consistent data then, it has an ability to match up with the human ability to for recognizing any kind of characters.

In the market there are other types of neural networks are available like Artificial Neural Network (ANN) and Recurrent Neural Network (RNN) but CNN is considered to be more powerful and efficient than ANN or RNN due to some advantages over other like facial recognition, text digitization and natural language processing etc. Thus CNN can be considered as a best technique character recognition and image processing using machine learning and deep learning. The impact of expanding the quantity of convolutional layers in CNN design on the presentation of transcribed digit acknowledgment is unmistakably introduced through the tests. The oddity of the current work is that it altogether explores all the boundaries of CNN engineering that convey best acknowledgment precision for a MNIST dataset. Companion scientists couldn't coordinate this precision utilizing an unadulterated CNN model. A few analysts utilized gathering CNN network models for the equivalent dataset to improve their acknowledgment precision at the expense of expanded computational expense and high testing multifaceted nature yet with practically identical exactness as accomplished in the present work.

In future, various designs of CNN, in particular, cross breed CNN, viz., CNN-RNN and CNN-HMM models, and space explicit acknowledgment frameworks, can be researched. Developmental calculations can be investigated for streamlining CNN learning boundaries, to be specific, the quantity of layers, learning rate and portion sizes of convolutional channels.

VI. REFERENCES

[1] Niu, X.X.; Suen, C.Y. A novel hybrid CNN– SVM classifier for recognizing handwritten digits. Pattern Recognit. 2012, 45, 1318– 1325.

[2] Long, M.; Yan, Z. Detecting iris liveness with batch normalized convolutional neural network. Compute, Mater. Contin. 2019, 58, 493–504.

[3] Y. LeCun et al., "Backpropagation applied to handwritten zip code recognition," Neural computation, vol. 1, no. 4, pp. 541-551, 1989.

[4] Sueiras, J.; Ruiz, V.; Sanchez, A.; Velez, J.F. Offline continuous handwriting recognition using sequence to sequence neural networks. Neurocomputing. 2018, 289, 119–128.

[5] Wells, Lee & Chen, Shengfeng & Almamlook, Rabia & Gu,Yuwen.(2018). Offline Handwritten Digits Recognition Using Machine learning.

[6] Burel, G., Pottier, I., &Catros, J. Y. (1992, June). Recognition of handwritten digits by image processing and neural network. In Neural Networks, 1992. IJCNN, International Joint Conference on (Vol. 3, pp. 666-671) IEEE.

[7] Salvador Boquera, Maria J. C. B., Jorge G. M. and Francisco Z. M., “Improving the Offline Handwritten Text Recognition with Hybrid HMM/ANN Models”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 33, No. 4, April 2014.

[8] Ahmed, M., Rasool, A. G., Afzal, H., & Siddiqi, (2017). Improving handwriting-based gender classification using ensemble classifiers. Expert Systems with Applications, 85, 158-168.

[9] Sadri, J., Suen, C. Y., & Bui, T. D. (2007). A genetic framework using contextual knowledge for segmentation and recognition of handwritten numeral strings. Pattern Recognition, 40(3), 898-919.

[10] Sarkhel, R., Das, N., Das, A., Kundu, M., & Nasipuri, M. (2017). A multi-scale deep quad tree based feature extraction method for the recognition of isolated handwritten characters of popular Indic scripts. Pattern Recognition, 71, 78-93.