



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

IJIRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 11, Issue 5, May 2022

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.118

Identifying Handwritten Numericals and Text Using DL Methods

Chandana A S¹, Chethana A S², Shubha Shree N³, Thanushree D K⁴, Rashmi Mothkur⁵

U.G. Student, Department of Computer Engineering, Dayananda Sagar University, Kudlu Gate, Bangalore, India¹

U.G. Student, Department of Computer Engineering, Dayananda Sagar University, Kudlu Gate, Bangalore, India²

U.G. Student, Department of Computer Engineering, Dayananda Sagar University, Kudlu Gate, Bangalore, India³

U.G. Student, Department of Computer Engineering, Dayananda Sagar University, Kudlu Gate, Bangalore, India⁴

Assistant Professor, Department of Computer Engineering, Dayananda Sagar University, Kudlu Gate, Bangalore, India⁵

ABSTRACT: Handwritten recognition is application based on the concept of interconnected system. It is an ability of a system to recognise as well as understand intelligible handwritten input from sources, likely paper documents. The motive of identifying handwritten numerical is to create an interconnected net which can appreciate and predict the manuscript digits and texts of copious languages by the depiction. By training this network we can help users provide any image in any shape or form in which this will be able to recognise the given image. This is a basic step towards analysing networks and cv. The aim of a handwritten prediction system is a conversion of manuscript digits and text into a system readable format. The main intent of this work is to ensure effective and reliable approaches for identification of the manuscript digits, text and make banking operations easier and error free.

KEYWORDS: CNN, ANN, MNIST, RE-LU, EasyOCR, TENSOE-FLOW, KERAS, PYTHON.

I. INTRODUCTION

This project helps texts, numerical which are in the manuscript form transform into computer readable form. The intention of this project is to find an effective and reliable way for recognising the manuscript numerical, text and make bank performance easy and also make it error-free.

Digit recognition is an application of imagerecognition, though considered as the first step to pattern identification and image identification it's a major leap towards utilising Artificial Neural Networks toward acknowledging the image. The Present project uses CNN for classifying given image of digits and text. Though considered as a solution in this field, utilising CNN to predict is much more efficient and accurate to predict the given text and digits and first steps towards accurate image identification. In this project, there are various layers, which can also be called as Regular Nets which will be trained with the information i.e., images.

An Artificial Neural Network (ANN) is data distillation ideal whatever stimulate the procedure of biotic nervous structure, like the cerebrum, action data. Essential technique for novel structure of details distillation system. Assured a lot of most associated clarified parts running in accord to investigate. ANNs, Individuals acquire knowledge at samples. An ANN is constructed for particular appeal, including pattern identification or data categorization, along erudition. erudition in biological systems presumes accustoming to synaptic linkage a certain be betwixt the axon.

CNN is mostly engaged in analysing visual-representation. CNN uses various layers for perceptions designed to enable minimal pre-processing. CNN was motivated by their design which was a connection of neurons that relate the organisation of the mammal striated layer. Single cortical neurons respond to stimuli only in an obstructed place of the peripheral vision which is called the perceptive vision. These fields slightly overlap in such a way that they cover the complete peripheral vision. CNN uses proportionally few initialisation images as contrast to analysing algorithms. This tells us that the network finds out the filters that were used in the customary algorithm. They have applications of pictures and tape spotting, recommendation technique, picture classification, medical picture analysis, and NLP.

EasyOCR is a raft in python that permits the conversion of image to text. It is by far the easiest way to execute OCR and it has access to many languages including English, Hindi, kannada and many more languages. EasyOCR doesn't have many software dependencies, it can directly be used with its API. It can operate numerous languages synchronously and it is agreeable with others as well.

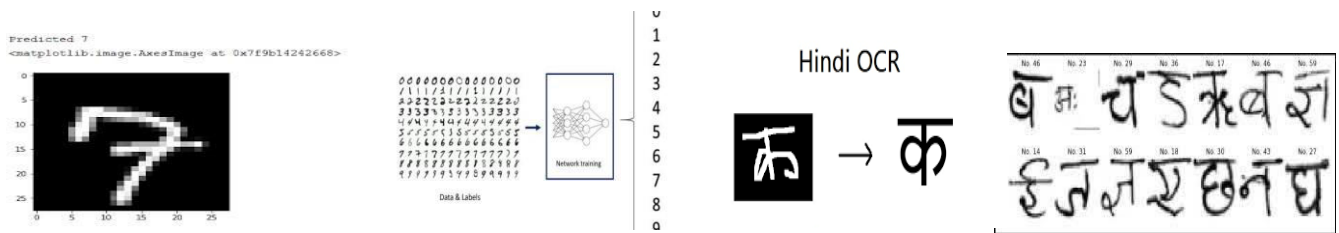


Fig. a. sample data

II. RELATED WORK

Immense research is going on in the field of handwritten recognition. Many people have developed systems for handwritten digit recognition and handwritten character recognition. We have studied some of the systems:[1]. Recognition of Handwritten digit recognition using convolutional neural networks. It shows that neural network classifiers with single-layer training can be applied efficiently to complex real-world, classification such as the recognition of handwritten digits.[2]. Handwritten digit recognition using CNN. It trains and tests a set of classifiers for handwritten digit recognition using MNIST database.[3]. Akeras based implementation for efficient handwritten digit Recognition using convolutional neural network. This paper defines process of handwritten digit recognition.[4] Handwritten Hindi Digits Recognition Using Convolutional Neural Network with RMSprop Optimization. It is an efficient handwritten Hindi numeral digit recognition structure based on Convolutional Neural Network (CNN) with RMSprop optimization technique.[5]. MNIST Kannada Digit Recognition. This paper presents an efficient handwritten Kannada digit recognition approach based on comparing the prediction accuracy of the CNN model and various machine learning algorithms.[6]. Handwritten Hindi Digits Recognition Using Convolutional Neural Network with RMSprop Optimization. It is an efficient handwritten Hindi numeral digit recognition structure based on Convolutional Neural Network (CNN) with RMSprop optimization technique.

III. METHODOLOGY

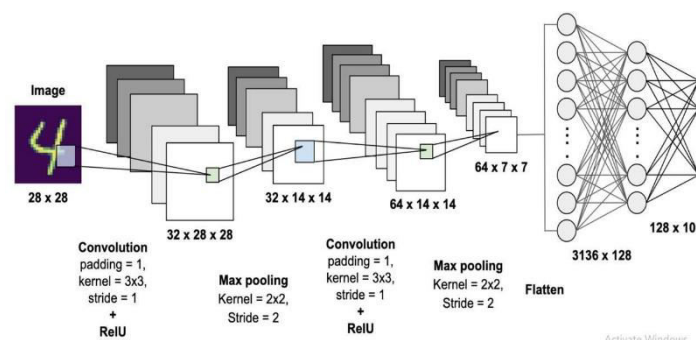


Fig. b. Proposed model architecture

The proposed model in the fig. b. consists of different layers; Sequential layer—This constructs the model for training, The most important class acclimated to configure the plan. Beginning is to construct ANN. This category gets back an item which is used to build the design layer after layer.3 Conv2D—This layer generates a complexity core that is convoluted with the layer put in above an isolated spatial extent to construct a tensor of turnout, this is the very first layer used in a model. Here attributes are being drowned out from our datasets. MaxPooling2D—A layer in CNN which helps in load identification, which reduces the measurements and allows for expectation about characteristics present in the sub-regions binned this operation is done on temporal details. Flatten—Flattening a tensor means to remove all of the

dimensions except for one. Dense-This is used to make well-regulated dimly-connected NN layers. This is purely a frequent film of axone in a NN. Individual axon is given for each axone in the preceding fill, thus closely linked. The activation function used in CNN; ReLU-This refers to an estimated strenuous role that accomplishes on multilayers of neural networks. In this layer we detach each of the pessimistic merits from the absolute image and supplant it with zero, ReLU helps to avert the steady growth to utilize the neural network.

Anaconda is applied as the python distribution and helps to set-up the libraries and apply them in all python scripts. The libraries used are Keras, Tensor-Flow, EasyOCR python package. The dataset used is MNIST manuscript data-set. The fig. c. flow chart shows how the output is predicted undergoing various process.

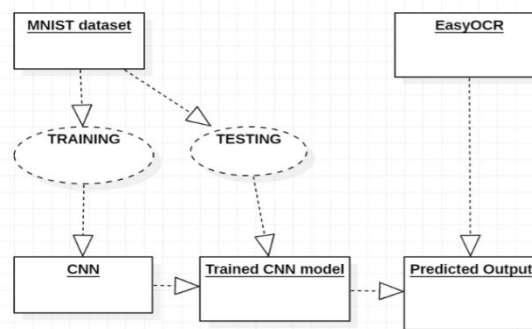


Fig. c. flow-chart

IV. EXPERIMENTAL RESULTS

Dataset-MNIST dataset is maintained by MNIST which holds more than 70,000 images. Various techniques are used to make the image into desirable form and shape. Then the images are given as the training set. After including the layers, we can test the output by giving the images to the model to predict. These neural networks are trained on more than 60,000 images. And tested on more than 10,000 images of digits and text in different languages.

```

Executing the pre-trained function

(16880, 28, 28)

2022-04-05 18:34:28.462990: I tensorflow/stream_executor/platform/default/dso_loader.cc:46] Could not load dynamic library 'cudart64_110.dll'; dlerror: cudart64_110.dll not found
2022-04-05 18:34:28.468952: W tensorflow/stream_executor/platform/default/dso_loader.cc:46] Could not load dynamic library 'cublas64_11.dll'; dlerror: cublas64_11.dll not found
2022-04-05 18:34:28.47022: W tensorflow/stream_executor/platform/default/dso_loader.cc:46] Could not load dynamic library 'cublas64_11.dll'; dlerror: cublas64_11.dll not found
2022-04-05 18:34:28.481462: W tensorflow/stream_executor/platform/default/dso_loader.cc:46] Could not load dynamic library 'cufft64_10.dll'; dlerror: cufft64_10.dll not found
2022-04-05 18:34:28.487707: W tensorflow/stream_executor/platform/default/dso_loader.cc:46] Could not load dynamic library 'cudnn64_10.dll'; dlerror: cudnn64_10.dll not found
2022-04-05 18:34:28.494338: W tensorflow/stream_executor/platform/default/dso_loader.cc:46] Could not load dynamic library 'cudnn64_10.dll'; dlerror: cudnn64_10.dll not found
2022-04-05 18:34:28.499990: W tensorflow/stream_executor/platform/default/dso_loader.cc:46] Could not load dynamic library 'cuspars64_11.dll'; dlerror: cuspars64_11.dll not found
2022-04-05 18:34:28.507162: W tensorflow/stream_executor/platform/default/dso_loader.cc:46] Could not load dynamic library 'cublas_8.dll'; dlerror: cublas_8.dll not found
2022-04-05 18:34:28.509022: W tensorflow/stream_executor/platform/default/dso_loader.cc:46] Could not load dynamic library 'cublas_8.dll'; dlerror: cublas_8.dll not found
Skipping registering GPU devices...
2022-04-05 18:34:28.522848: I tensorflow/stream_executor/platform/default/dso_loader.cc:46] This TensorFlow library is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the x86_64 FMA3 and AVX2 instructions, recompile TensorFlow with the appropriate compiler flags.
To enable these in other operations, rebuild TensorFlow with the appropriate compiler flags.
2022-04-05 18:34:28.596270: I tensorflow/compiler/mlc/trt_convert_utils.cc:185] None of the MLIR Optimization Passes are enabled (registered 2)
Epoch 1/2
1875/1875 [=====] - 3s 2ms/step - loss: 0.2598 - accuracy: 0.9344
Epoch 2/2
1875/1875 [=====] - 4s 2ms/step - loss: 0.1391 - accuracy: 0.9470
Epoch 3/2
1875/1875 [=====] - 3s 2ms/step - loss: 0.0707 - accuracy: 0.9702
2022-04-05 18:34:28.596270: I tensorflow/compiler/mlc/trt_convert_utils.cc:185] None of the MLIR Optimization Passes are enabled (registered 2)
0.09407768877943
0.97020022294309
  
```

(d)

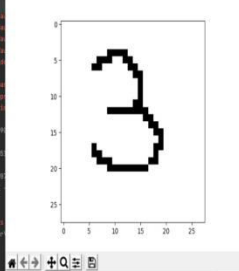
```

Python Project Handwritten-digit-recognition

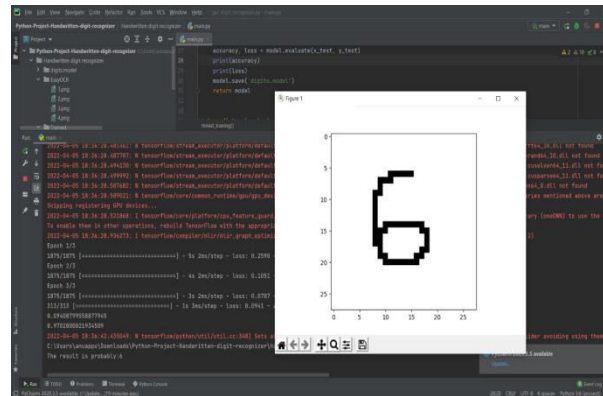
model.add(LSTM(64, input_shape=(1, 28, 28), return_sequences=True))
model.add(LSTM(64, input_shape=(1, 28, 28), return_sequences=True))
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
model.fit(train_images, train_labels, epochs=10)
accuracy = model.evaluate(test_images, test_labels)
print('Accuracy: %f' % accuracy)

Accuracy: 0.9470

The result is probably a
the result is probably 2
  
```

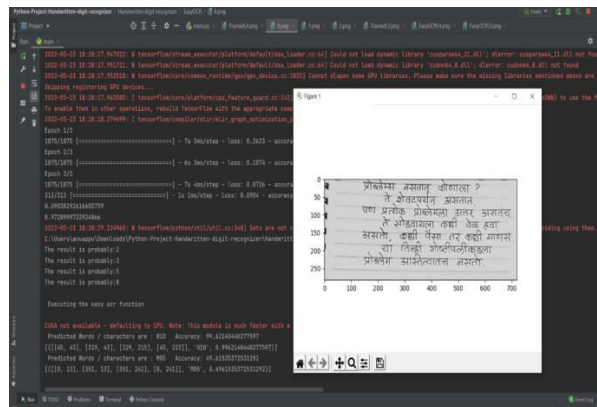


(e)

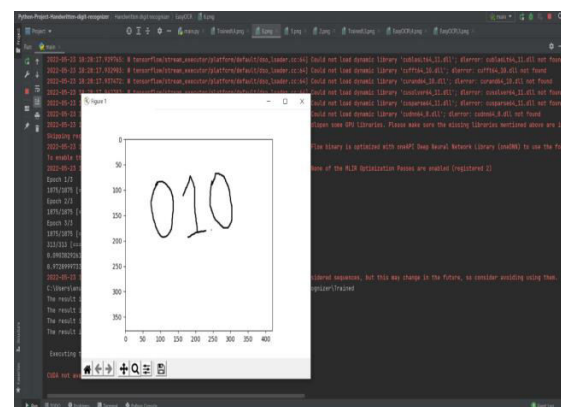


(f)

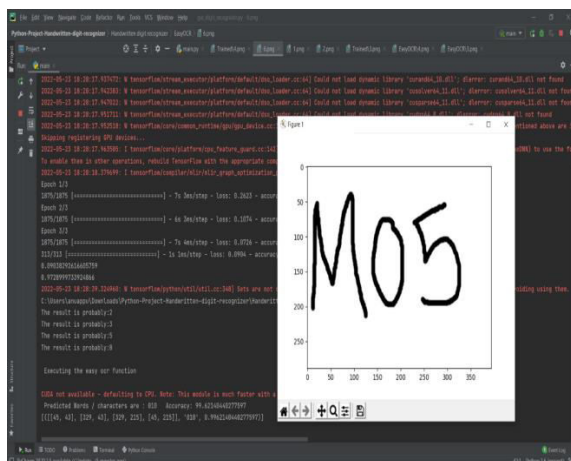
Fig.1 Trained function (d). shows executing the model and training the model. (e). shows 3 which is the predicted output for input 3. (f). predicting the handwritten digit 6.



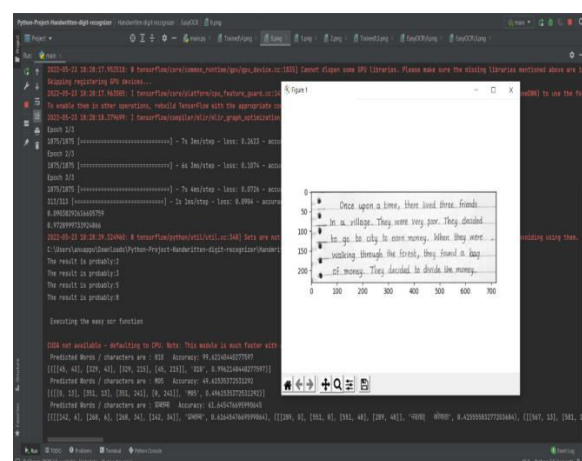
(g)



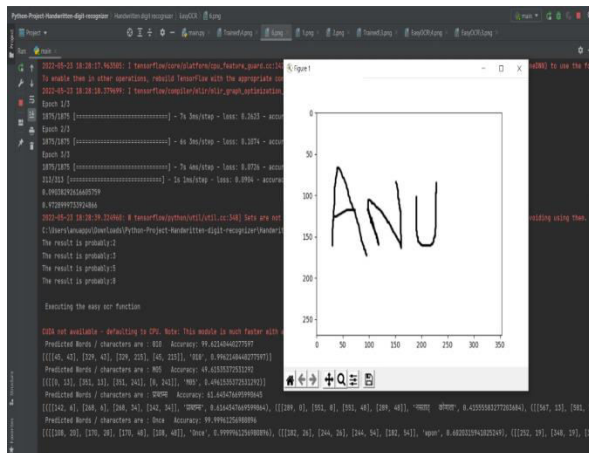
(h)



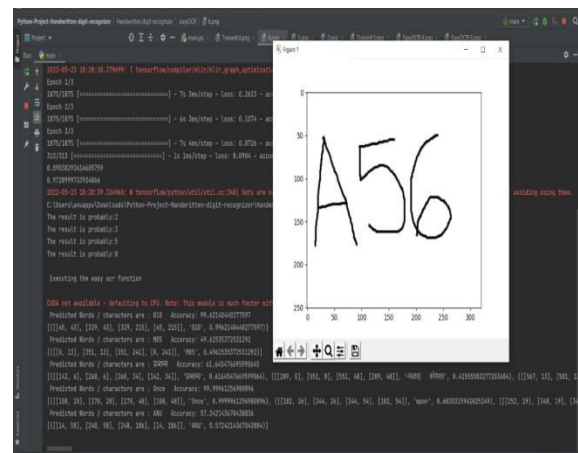
(i)



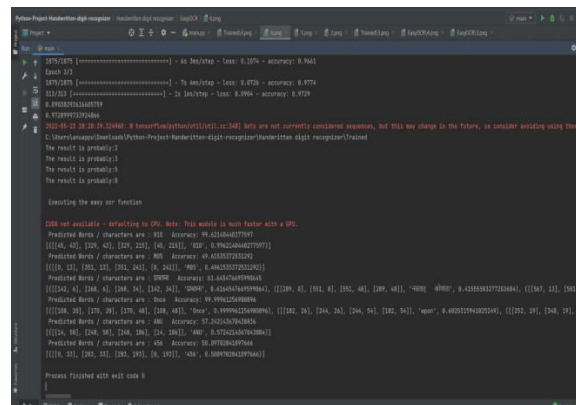
(j)



(k)



(l)



(m)

Fig.2 Easyocr function (g) predicting the handwritten text image of an Hindi language.(h). predicting handwritten triple digits.(i). predicting the handwritten alphanumeric.(j). predicting the handwritten text image of an English language.(k). predicting the handwritten text.(l). predicting the handwritten Alphanumeric. (m). Results of the handwritten images under easyocr.

V. CONCLUSION

We have implemented a handwritten digit and text recognition system. Our algorithm successfully detects the text and the digit up to 80% of the accurate results are observed. We have applied our algorithm on many images handwritten in various languages such as Hindi, English and found that we get accurate results.

REFERENCES

1. Md.Anwar Hossain, Md.MohonAli, "Recognition of Handwritten digit recognition using convolutional neural networks", Global journal of computer Science and Technology ,Volume 19 Issue 2 Version 1.0,2019
2. Vijayalaxmi R Rudraswamimath,Bhavanishankar, "Handwritten digit recognition using CNN"International Journal of Innovative Science and research Technology,2019
3. MuzamiliNawaz,SandeshGangwani,Isma Farah Siddiqui"Akeras based implementation for efficient handwritten digit Recognition using convolutional neural network ",Asia Pacific conference on information science and technology, 2020



4. FeiyangChen,NanChen,HanyangMao,HanlinHu“Assessing Four Neural Networks on Handwritten Digit Recognition Dataset (MNIST)”,CHUANG XINBAN JOURNAL OF COMPUTING,20 Jul 2019
5. S Ahlawat,AChoudhary,“Hybrid CNN-SVM classifier for handwritten digit recognition”,International conference on computational intelligence and data science, 2019
6. R. Vijaya Kumar Reddy,Dr. B. Srinivasa Rao,K. Prudvi Raju “Handwritten Hindi Digits Recognition Using Convolutional Neural Network with RMSprop Optimization” “Second International Conference on Intelligent Computing and Control Systems ,2018”.



INNO SPACE
SJIF Scientific Journal Impact Factor
Impact Factor: 8.118



ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



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