IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

HANDWRITTEN CHARACTER RECOGNITION

[1]Deepika Mangnani, [2]Jasika Sukheja, [3]Hanisha Mohinani, [4]Prashant Kanade

[1]Student, [2]Student, [3]Student, [4]Professor,

[1]Department of Computer Engineering,

[1]Vivekanand Education Society's Institute of Technology, Chembur, Mumbai.

Abstract: The method of identifying the handwritten text using machine interface is known as handwritten character recognition. Handwritten characters present challenges because they differ from one writer to the other; even though the same author writes the same character, the form, size, and location of the character vary. Handwritten Character Recognition has been one of the challenging areas in the field of image processing. It has numerous applications which include, bank cheques and conversion of any handwritten document into digital text form. There is a prominent need for storing information to desktop storage from the information available in handwritten images to later reprocess this information utilizing computers. But to re-process this information, it would be very difficult to read the other information from these image files. Therefore, a technique to automatically resolve and store information, in particular text, from image files is needed. This paper aims to classify an individual handwritten word so that handwritten text for English alphabets can be translated to a typed or editable form. We used the Convolutional Neural Networks to accomplish this task. We used this architecture to train a model that can accurately predict words. Trends suggest that the market value for such applications is going to increase exponentially in the future.

Index Terms **- Convolutional Network Layer, Architecture of CNN, English Words, image files.**

I. INTRODUCTION

Many of us prefer to take our notes traditionally: with a pen. However, it's troublesome to store and consistently retrieve documents. Thus, heaps of necessary information gets lost or doesn't get reprocessed as a result of documents not getting transferred to editable text format. Thus, the main target of this project is to explore the task of classifying written text and changing written text into a digital format.

Handwritten Recognition refers to the ability to interpret the handwritten text or words and convert them to digital format. Offline handwriting recognition systems generally consist of four processes: collection, segmentation, recognition, and prediction. First, the handwritten text or words in image format or any noneditable format is collected. Second, the text or word is segmented into characters. Third, each character is recognized using Handwritten recognition techniques. Finally characters are merged together to reconstruct an entire word.

In recent years, handwriting recognition has emerged as a significant attention-grabbing and sophisticated research area in the fields of image processing and pattern recognition. It makes a significant contribution to the sequence of a mechanisation technique and improves the interface between humans and machines. Various research efforts are focusing on novel methods and schemes that can shorten the process time while ensuring the highest recognition accuracy.

The next section discusses the related work done by various authors followed by details of the planned system.

II. RELATED WORK

Word Recognition has been an effective area of research in the past and due to its diverse applications, there has been significant research when it comes to Optical Characters Recognition. Several researchers have implemented various algorithms for handwritten character recognition. *Mehmet Kaya [2], Deepak Kadam [1] and* R. *Francis [3]* have done a significant research on how the noise in the image can be reduced and implemented the different algorithms

The discussion is further continued in the following sub-points.

A. Handwritten *Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR)*

The authors of this paper Jamshed Memon, Maira Sami, Rizwan Ahmed Khan and Mueen Uddin have explained various classification methods for digitizing Handwritten images. A comparative study is also shown for various datasets to be used in

recognizing the handwritten text in the documents. The authors have analysed the working of algorithms for six widely spoken languages including Urdu, Chinese, Arabic and Persian.[1]

B. *Identification of Optical Character Recognition (OCR) Engine for Proposed System*

Aditya Lakhadive and Snehwardhan Marathe have proposed a system of Handwritten Character Recognition using OCR. Steps used for the proposed system are Digitization, Pre-processing, Segmentation, Feature Extraction and Post-processing. Tesseract, an inbuild library, is used for OCR. The accuracy of tesseract is increased by processing the input image character by character and also border removal of the image is done.

C. *Literature Survey on Recognition and Evaluation of Optical Character Recognition (OCR)*

The authors of this paper Deepak Kadam, Prathamesh Chavan and Prashant Pandhara [3] have used classifiers SVM, KNN and NN for recognition for handwritten characters. The images given to the model are in JPEG, PNG or BMP formats and convert the RGB image to equivalent HSV colour space image. Further, extraction of the skeleton of character and dilation is performed to remove the extraneous noise and at last segmentation is done. Limitations are specially designed fonts are not considered and recognition of single hand drawn alphabets which are not connected is considered.

D. *A Detailed Analysis of Optical Character Recognition Technology*

Authors of the paper Karez Abdulwahhab Hamad and Mehmet Kaya have mentioned about the major challenges of OCR such as Scene Complexity, Conditions of Uneven Lightning, Skewness, Blurring and Degradation, Aspect Ratios, Tilting, Fonts, Multilingual Environments and Warping. Also, the paper describes various phases and architecture of OCR which includes Pre-Processing Phase, Segmentation Phase, Normalization Phase, Feature Extraction Phase, Classification Phase and Post Processing Phase. [4]

E. *A Detailed Analysis of Optical Character Recognition Technology*

K. Karthick, K. B. Ravindrakumar, R. Francis and S. Ilankannan, [5] the authors have described various stages in text recognition, handwritten OCR systems classification according to the type of text and implemented the study on Chinese and Arabic Text. Images are given as input which are converted to gray scale and MSER regions are detected. Further, kWh Area is cropped, dilation is performed and text recognition is done using OCR and results are stored in the notepad. Their study has obtained optimal results with less computation time for multilingual character segmentation having better accuracy.

F. *A Survey on Optical Character Recognition System, Journal of Information & Communication*

Noman Islam, Zeeshan Islam and Nazia Noor have presented an overview of various techniques of OCR. Various phases of OCR such as acquisition, pre- processing, segmentation, feature extraction, classification and post-processing have been described in detail. Along with this, they have mentioned various applications of OCR in number plate recognition, bank cheque reading, passport validation, utility bills and smart libraries. [6]

III. METHODOLOGY

In the proposed system the Convolutional Neural Network is used for character recognition after the given input is preprocessed. The Neural Network along with character segmentation is explained in detail further.

First and foremost step is data collection. In order to train the proposed model, we have used the EMNIST dataset. The EMNIST Handwriting Dataset contains various handwritten capital and small alphabets and numbers which can be used to train and test handwritten text recognizers and to perform writer identification and verification experiments. The EMNIST dataset is a set of handwritten character digits derived from the NIST Special Database 19 and converted to a 28x28 pixel image format and dataset structure.

After the data collection, the next step is preprocessing the data collected. Thresholding, resizing, reshaping, Blurring are the techniques in which the input images are preprocessed.For preprocessing user's input images in Python, OPENCV that is cv2 library is used. Thresholding is a form of image segmentation in which the pixels of an image are changed to make it easier to interpret. Thresholding is the process of converting a colour or grayscale image into a binary image, which is simply black and white image.In this model, thresholding is used as a way to select areas of interest of an image.A Gaussian blur (also known as Gaussian smoothing) is the effect of blurring an image with a Gaussian function in image processing. It's a common effect in graphics software that's used to minimise image noise and detail.Since user's input image can be of any size, the image for the proposed model is resized to 28x28 px Binarization is the process of transforming a grayscale image to black-and-white, effectively reducing the image's information from 256 shades of grey to two: black and white, yielding a binary image.

After pre-processing of the image is done, the characters in the given text or word are recognized or segmented if necessary using the cv2 library. After the images containing text has been segmented or processed, the next step is designing and training the model. The proposed model consists of 3 convolutional neural network (CNN) layers.

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 26, 26, 32)	320
max_pooling2d (MaxPooling2D)	(None, 13, 13, 32)	0
conv2d_1 (Conv2D)	(None, 13, 13, 64)	18496
max_pooling2d_1 (MaxPooling2	(None, 6, 6, 64)	0
conv2d_2 (Conv2D)	(None, 4, 4, 128)	73856
max_pooling2d_2 (MaxPooling2	(None, 2, 2, 128)	0
flatten (Flatten)	(None, 512)	0
dense (Dense)	(None, 64)	32832
dropout (Dropout)	(None, 64)	0
batch_normalization (BatchNo	(None, 64)	256
dense_1 (Dense)	(None, 128)	8320
dropout_1 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 62)	7998

Total params: 142,078
Trainable params: 141,950
Non-trainable params: 128

Figure 1: Layers for CNN Model

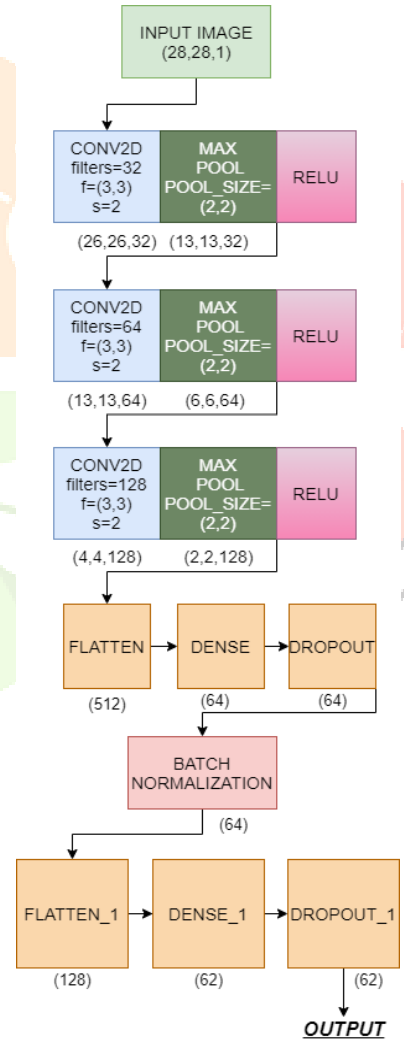


Figure 2: Architecture of CNN Model

CNN stands for Convolutional Neural Networks that are used to extract the features of the images using several layers using filters. The convolution layers are generally followed by maxpool layers that are used to concentrate the number of features extracted and ultimately the output of the maxpool layers and convolution layers are flattened into a vector of single dimension and are given as an input to the Dense layer (The fully connected network).

In the present model input image is a gray-value image and has a size of 28X28.Also here, we have used 3 CNN layers. Convolutional layer to connect the first layer as a convolutional layer with 32 filters of scale (3,3). The normalization of the negative values, achieved by ReLu Activation functions. To pick the maximum element of the features section, a pooling layer of size (2,2) with stride 2 is used. The output from the pooling layer is forwarded to second Convolutional layers with 64 filters of scale (3,3) and to the pooling layer. The output from the second pooling layer is forwarded to third Convolutional layers with 128 filters of

scale (3,3) and again to the pooling layer. The output of the final maxpool layer is flattened and passed as input to a fully connected dense layer achieved by ReLu Activation functions. The dropout function is set to 0.2 which avoids overfitting. After that we have applied batch normalization because BN reduces training time and accelerates training, and make very deep network trainable.

For taking the input and displaying the predicted output Flask Framework is used as the front-end.

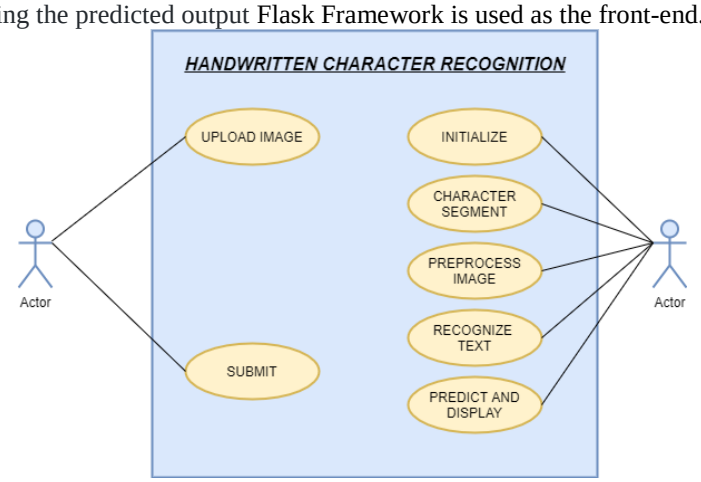


Figure 3: Use Case Diagram

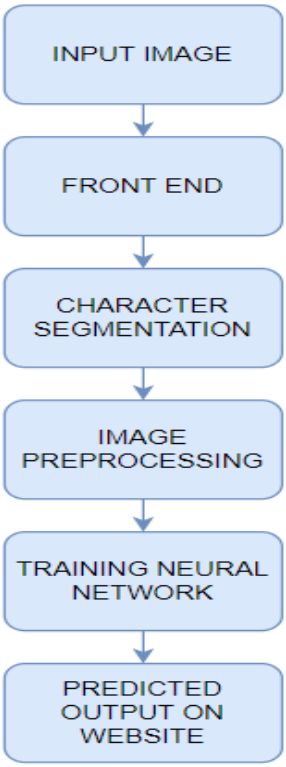


Figure 4: Block Diagram

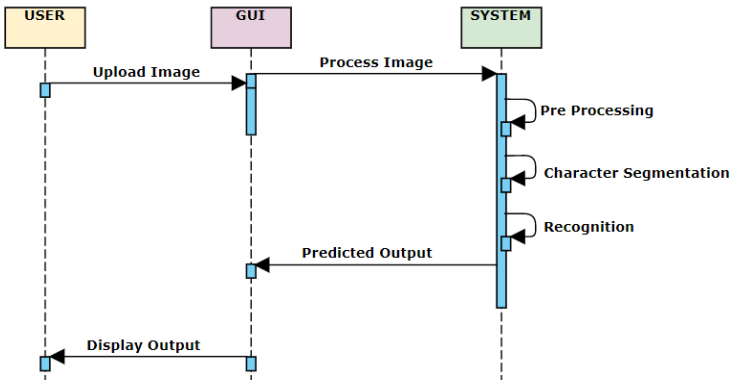


Figure 5: Sequence diagram

IV. OUTCOMES

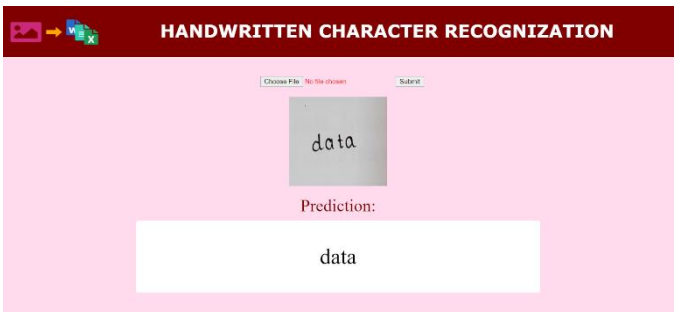
Our algorithm is implemented in Python using flask as a framework and tested on a PC with Intel Core i5-8590 CPU, 8GB RAM. The experimental results show how effective is the proposed deep learning-based system, for extracting the text from handwritten text image.

Training the NN model used for extraction in the chunk epochs we have generated an accuracy of 86.5% for the test dataset

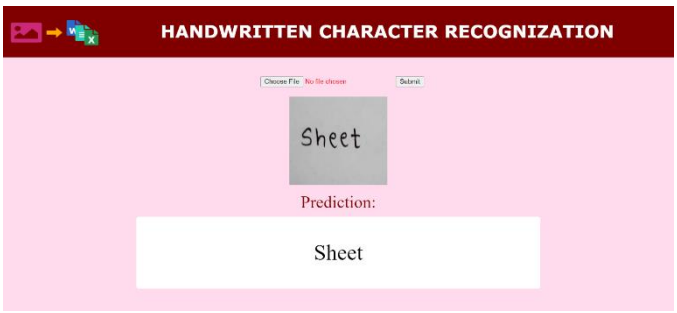
Following are the results that have been obtained.

```
Epoch 1/15
17449/17449 [==============================]
Epoch 2/15
17449/17449 [==============================]
Epoch 3/15
17449/17449 [==============================]
Epoch 4/15
17449/17449 [==============================]
Epoch 5/15
17449/17449 [==============================]
Epoch 6/15
17449/17449 [==============================]
Epoch 7/15
17449/17449 [==============================]
Epoch 8/15
17449/17449 [==============================]
Epoch 9/15
17449/17449 [==============================]
Epoch 10/15
17449/17449 [==============================]
Epoch 11/15
17449/17449 [==============================]
Epoch 12/15
17449/17449 [==============================]
Epoch 13/15
17449/17449 [==============================]
Epoch 14/15
17449/17449 [==============================]
Epoch 15/15
17449/17449 [==============================]
Model: "sequential"
```

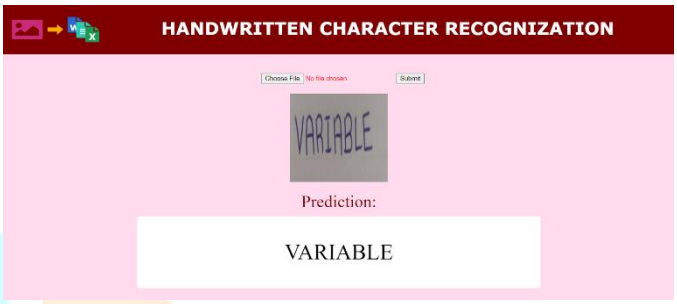
Output 1:



Output 2:



Output 3:



Output 4:

ACCURACY v/s EPOCH:

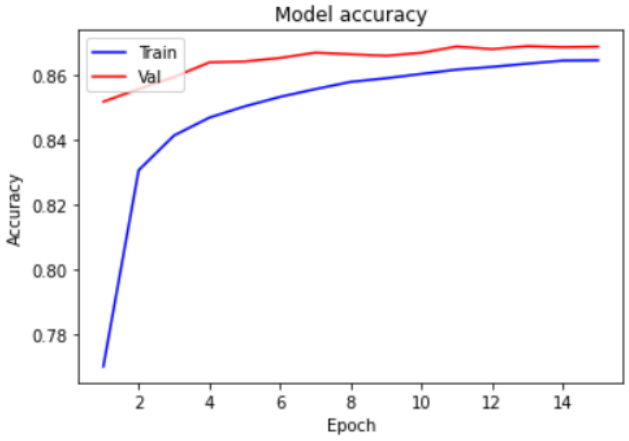


Figure 6: Accuracy vs Epoch graph

LOSS v/s EPOCH:

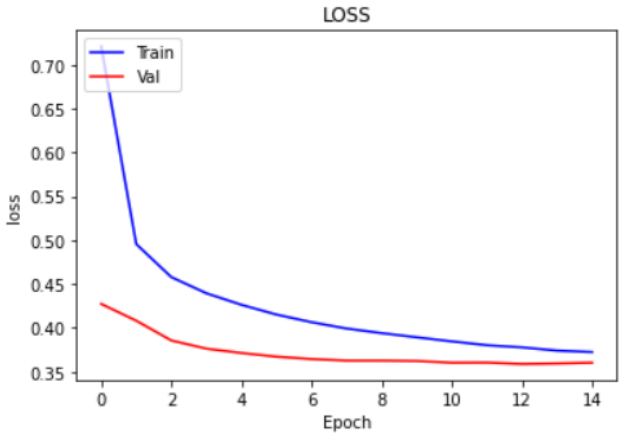


Figure 7: Loss vs Epoch graph

V. LIMITATIONS

- Noise in the input image should be as minimum as possible.
- Accuracy of the system is reduced while detecting 9 or g, 0 or o, i or j, r or 8.
- accuracy for ligatures or cursive handwriting is less.

VI. FUTURE SCOPE

- To build the same system for other languages such as Hindi, Urdu, Sindhi and South Indian Languages.
- This work can also serve as the initial phase for Translating English into different languages for the people who lag in understanding English.
- This work can be further expanded for recognition of long documents.

REFERENCES

[1] Jamshed Memon, Maira Sami, Rizwin Ahmed Khan and Mueen Uddin, Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR), IEEE XPLORE 28 July 2020.

[2] Mrunal Marne, Pravin Futane, Sakshi Kolekar,
Aditya Lakhadive and Snehwardhan Marathe, Identification of Optimal Optical Character Recognition(OCR) Engine for Proposed System, IEEE Xplore 25 April 2019.

[3] Deepak Kadam, Prathamesh Chavan and Prashant Pandhara, Literature Survey on Recognition and Evaluation of Optical Character Recognition (OCR), International Journal of Scientific & Engineering Research Volume 9, Issue 2, February-2018.

[4] Karez AbdulWahhab Hamad and Mehmed Kaya, A Detailed Analysis of Optical Character Recognition Technology, International Journal of Applied Mathematics Electronics and Computers December 2016.

[5] K. Karthick, K. B. Ravindrakumar, R. Francis and S. Ilankannan, Steps Involved in Text Recognition and Recent Research in OCR; A Study, International Journal of Recent Technology and Engineering (IJRTE), Volume-8, Issue-1, May 2019.

[6] Noman Islam, Zeeshan Islam and Nazia Noor, A Survey on Optical Character Recognition System, Journal of Information & Communication Technology-JICT Vol. 10 Issue. 2, December 2016.

[7] Vaibhav Mainkar, Jyoti Katkar, Ajinkya Upade and Poonam Pednekar, Handwritten Character Recognition to obtain Editable Text, IEEE Xplore 4th August 2020.

[8] Hitesh Mohapatra, Handwritten Character Recognition(HCR) using Neural Network, ResearchGate September 2009.

[9] G.Vamvakas, B.Gatos, N. Stamatopoulos, and S.J.Perantonis, A Complete Optical Character Recognition Methodology for Historical Documents, ResearchGate September 2008.

[10] Swati Tomar and Amit Kishore, A Review: Optical Character Recognition, International Journal of Engineering Sciences & Research Technology April 2018.

[11] Qing Chen and E.M. Petriu, Optical Character Recognition for model-based Recognition Applications, IEEE Xplore 10 November 2003.

[12] Vivank Sharma, B. Valaramathi, K. Santhi, Shobhit Srivastava and Sumit Jahagirdar, Performance Analysis of the Classifiers for Optical Character Recognition, IEEE Xplore 16 April 2020.

[13] Rishabh Mittal and Anchal Garg, Text Extraction using OCR: A Systematic Review, IEEE Xplore September 2020.

[14] Bhagyashree D. Upadhyay, Dr.Sachin Chaudhary, Monali Gulhane, A Review Paper on Handwritten Character Recognition Using Machine Learning, International Journal of Future Generation Communication and Networking Vol. 13, No. 2s, (2020), pp. 1256–1266.