International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 4, Issue 4, pp 50-53, April-2020

Recognition of Handwritten Character using Artificial Intelligence

Naresh Katkuri

Associate Professor, Department of Computer Science and Engineering, Malla Reddy College of Engineering for Women, Hyderabad -500100, Telangana, India

Abstract - There are many things that humans have in common, yet there are other things that are very unique to every individual and one of them is handwriting. Handwriting is a skill that is personal to individuals. It has continued as a means of communication and recording information in day-to-day life. Because each person's handwriting is unique, it is sometimes hard to interpret the information they try to convey. As computerization is becoming more prominent these days, Handwriting Recognition is gaining importance in various fields. The major focus is to understand the handwriting and convert it into readable text. Deep learning, an ability of Artificial Intelligence (AI), is used for the system to learn the input automatically and convert the handwritten text to printed text.

Keywords: Handwriting Recognition, Deep learning.

1. INTRODUCTION

Handwriting has continued as a means of communication in our day-to-day life. As each person's handwriting is unique, it is sometimes hard to interpret the information they try to convey. Handwriting Recognition is an ability of a computer to receive and interpret intelligible handwritten input from sources such as paper documents, photographs, touch-screens and other devices. Though it is a difficult problem due to the great variations of writing styles, different size and orientation angle of the characters, it is still found useful for the applications in some way. The main objective is to understand the handwriting and convert it into readable text, which includes characters, words, lines, paragraphs etc. In this project, the challenge is classifying the image of any handwritten word, which might be of the form of cursive or block writing. Along with this, Text-to-Speech is used to help people who have trouble reading on-screen text.

2. RELATED WORKS

Hao Zeng et al. proposes a method that focuses on using a simpler neural network instead of complicated ones that require high quality computer configuration to recognize handwritten digits with relatively promising accuracy[1]. MNIST dataset is used to train the neural network. An Efficient Algorithm for Real-Time Handwritten Character Recognition in Mobile Devices[2] that is used to measure the algorithm efficiency, tests were applied to 8 persons, who are related to the computing scene and outside it, with different writing styles. Mobile Client-Server Approach for handwritten digit recognition [3] is another one, in which CNN was also used to improve the performance of neural networks. The digit recognition consists of some modules for the processing.

Rohan Vaidya et al. designed an image segmentation based handwritten character recognition system using Deep-Learning [4]. OpenCV was used for performing Image processing and Tensor flow was used to train a neural network. Haishi Du et al. proposed a system that identifies words using acoustic signals generated by pens and paper using Deep-Learning [5]. The framework is created with three major components: segmentation, classification, and word suggestion. Handwritten Document into Digitized Text Using Segmentation Algorithm [6]. The main aim is to help in preserving history by making information searchable, easily, and reportable without the need for human labor.

Edgard Chammas et al. proposed a mobile application that is built on a distributed architecture that allows tourists to obtain additional information about location and menu entries in the Arabic language [7]. The recognition of printed texts is done using optical character recognition. SolveIt- an Application for Automated Recognition and Processing of Handwritten Mathematical Equations[8]. Here a convolutional neural network (CNN) is used to classify symbols. The recognized symbols are strung together to form an equation that can be parsed by the math engine (SymPy2). A framework that takes the image of multiple printed-papers using a mobile device's camera is used in Optical Character Recognition (OCR) Performance in Server-based Mobile Environment. After the first image is captured, the image is then directly sent to a server. Server processes the image using the OCR application directly and sends the text file back to the mobile device [9]. Handwritten Character Recognition to obtain Editable Text [10] is a system proposed by Ms. Jyoti A. Katkar. No internet connectivity is required for character recognition in the system. And the system offers 90% accuracy.

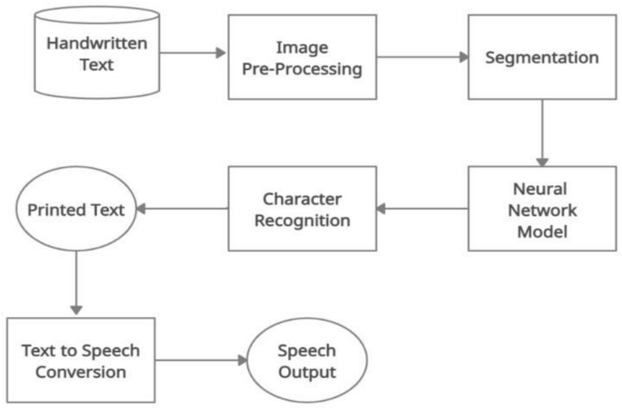

Fig - 1: An overview of the system

3. VARIOUS MODULES IN THE PROPOSED SYSTEM

The design of the proposed Handwriting recognition system can be broadly categorized into two significant parts – front-end and back-end, which is divided into five modules – Registration and authentication, Image processing, Neural network modeling and training, Character recognition and Text-to-speech conversion. For the front-end, Flutter is used. Flutter is an open-source UI software development kit created by Google. It is used to develop applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase. And for back-end keras, opencv, tensorflow lite, google text-to-speech, and firebase are used.

Registration and Authentication

The registration module consists of the user's sign-up and login procedures. The user can create an account by adding an email-id and password and can login from anywhere at any time using that email-id and password.

Image processing

The image processing module is divided into two phases:

a) Image Uploading, b) Image Preprocessing

In Image Uploading the image can be given as and input either by capturing the image through a mobile camera or by selecting an image from the mobile storage. Only one image can be given as the input. In Image Preprocessing, this phase is divided into two parts: Image Preprocessing and Segmentation. Image Preprocessing involves different stages of operations in order to enhance the image for further processing. Preprocessing involves resizing of image, gray-scaling and binarization. The input images can be of varying size, so the images are resized to a standard size before feeding the input to the model. Gray-scaling is the process of converting an image from other color shades to shades of gray. Gray-scaled images are typically composed of shades of gray, varying from black, at the weakest intensity to white, at the strongest. The values of intensity range from 0 to 255. Binarization is the process of transforming character image into the binary (0 and 1) form. It is an important stage to be performed on gray scale images. Segmentation and character recognition would be much easier once the process of binarization is carried out in a proper manner.

Image segmentation is the process of partitioning an image into multiple segments. Image classification is the process of predicting a specific class, or label, for something that is defined by a set of data points. A comparison takes place between the input and the stored values (patterns) to find the appropriate match class for the input images. The training and test dataset are reshaped so that the refined dataset can be given to the model. Keras [12] is an open-source software library that provides a Python interface for artificial neural networks. It acts as an interface for the TensorFlow library. Keras has support for convolutional and recurrent neural networks and other common utility layers. TensorFlow Lite [13] provides a framework for a trained TensorFlow model to be compressed and deployed to a mobile or embedded application. The computationally expensive process of training can still be performed by Tensor Flow in the environment that best suits it.

Neural Network Modeling and Training

A Convolutional Neural Network (CNN) [14] is used to perform recognition tasks and classification. The following steps are performed in the CNN model: 1. Input image isfed into the CNN layers, 2. Image processing, 3. Comparison and prediction takes place 4. Output is the printed text. A CNN model for recognizing the handwritten characters was created using Keras and Tensorflow backend. Keras is an open-source software library that provides a Python interface for artificial neural networks and also acts as an interface for Tensorflow library. 4.4 Character Recognition

The handwritten character image is converted into printed text by sending the image to the model. In the model, each character is recognized after preprocessing and they are converted into readable text.

Text to Speech conversion

The printed text is converted into audio by sending the printed text to the text-to-speech converter. Google Text-to-Speech is used for this. It is a screen reader application developed by Google for its Android operating system. It powers applications to read aloud (speak) the text on the screen with support for many languages. This audio feature can be used by the user only if they want to hear it. However the audio is generated after character recognition.

4. EXPERIMENTAL RESULTS

Training loss is the error on the training set of data. Validation loss is the error after running the validation set of data through the trained network. The loss obtained by the model is shown as a graph in figure 3, i.e.; how well or poorly the model behaves after each epoch. The red line represents the training loss and the blue line represents the validation loss. Our aim is to make the validation loss as low as possible.

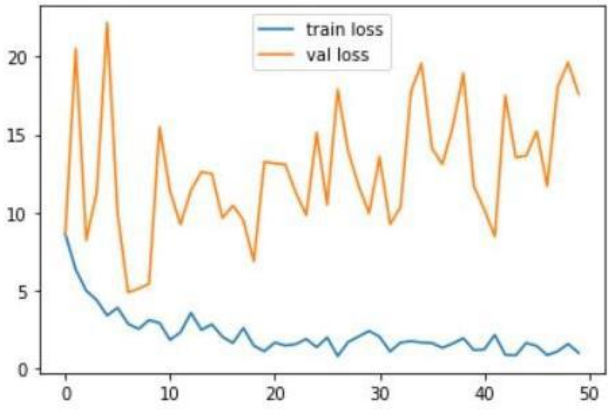

Fig - 2: Loss graph of the model

The accuracy of the model is represented in figure 4. The accuracy of a model on examples it was constructed on is called as the training accuracy and the validation accuracy is the accuracy you calculate on the data set you do not use for training, but you use (during the training process) for validating (or "testing") the generalization ability of your model or for "early stopping". The blue line indicates the training accuracy and the red line indicates the validation accuracy.

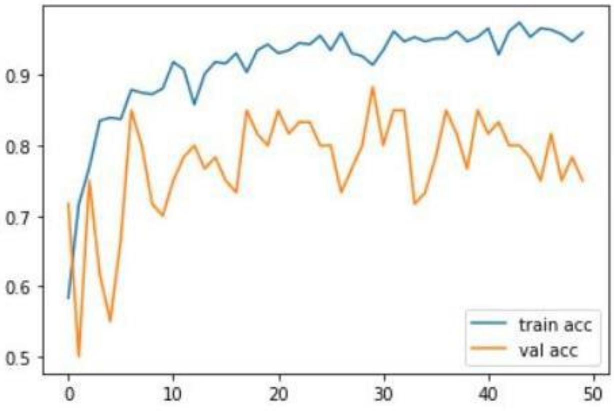

Fig -3: Accuracy graph of the model

The login page of the system is shown in Figure 3. The user can login to the app using their credentials. The image can be captured using the mobile camera or by uploading it from the device storage. This is shown in figure 4.

5. CONCLUSION

A mobile application is designed to provide a facility to convert written text into printed text. Along with this, Text-to-Speech is incorporated to help people who have trouble reading on-screen text. This application in the wider range will help many people in their day to day activities. The model created for handwritten character recognition has proved to be of fairly good efficiency.

6. FUTURE SCOPE

In future work, we train the model to read a whole handwritten or non-handwritten document and convert the printed text to different languages. We also incorporate a personalized scanned file storage set-up for each registered user. The model can be trained further for specific purposes.

REFERENCES

[1] R. Vaidya, D. Trivedi, S. Satra and P. M. Pimpale, "Handwritten Character Recognition Using Deep-Learning",2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018, pp. 772-775, doi: 10.1109/ICICCT.2018.8473291.

[2] H. Du, P. Li, H. Zhou, W. Gong, G. Luo and P. Yang, "WordRecorder: Accurate Acoustic-based Handwriting Recognition Using Deep Learning,"IEEE INFOCOM 2018 - IEEE Conference on Computer Communications, 2018, pp. 1448-1456, doi: 10.1109/INFOCOM.2018.8486285.

[3] Kavitha, D., & Shamini, P. (2016),"Handwritten Document into Digitized Text Using Segmentation Algorithm", An International Journal of Advanced Computer Technology. Retrieved from https://ijact.in/index.php/ijact/article/view/465

[4] H. Zeng,(2020)An Off-line Handwriting Recognition Employing Tensorflow. International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE).

[5] Beltrán and S. Mendoza, "Efficient algorithm for real-time handwritten character recognition in mobile devices ",2011, 8th International Conference on Electrical Engineering, Computing Science and Automatic Control, 2011, pp. 1-6, doi: 10.1109/ICEEE.2011.6106583.

[6] H. A. Shiddieqy, T. Adiono and I. Syafalni, "Mobile Client-Server Approach for Handwriting Digit

Recognition",2019 International Symposium on Electronics and Smart Devices (ISESD), 2019, pp. 1-4, doi: 10.1109/ISESD.2019.8909448.

****