

# Design of Machine Learning Technique for Handwritten Text Recognition

<sup>2</sup>Sachin Kumar Gupta, <sup>2</sup>Dr. Ramesh Solanki

<sup>1</sup>Students, <sup>2</sup>Assistant Professor,  
Department of MCA,  
Vivekanand Education Society's Institute of Technology,  
University of Mumbai, India

**Abstract:** Handwriting Detection is a technique or ability of a Computer to receive and interpret intelligible handwritten input from source such as paper documents, touch screen, photo graphs etc. Handwritten Text recognition is one of area pattern recognition. The purpose of pattern recognition is to categorizing or classification data or object of one of the classes or categories. Handwriting recognition is defined as the task of transforming a language represented in its spatial form of graphical marks into its symbolic representation. Each script has a set of icons, which are known as characters or letters, which have certain basic shapes. The goal of handwriting is to identify input characters or image correctly then analysed to many automated process systems. This system will be applied to detect the writings of different format. The development of handwriting is more sophisticated, which is found various kinds of handwritten character such as digit, numeral, cursive script, symbols, and scripts including English and other languages. The automatic recognition of handwritten text can be extremely useful in many applications where it is necessary to process large volumes of handwritten data, such as recognition of addresses and postcodes on envelopes, interpretation of amounts on bank checks, document analysis, and verification of signatures. Therefore, computer is needed to be able to read document or data for ease of document processing.

**Keywords:** NLP- Natural Language Processing, CNN- Convolution Neural Network, OCR- Optical Character Recognition.

## 1. INTRODUCTION

The Handwritten text system is commonly used system in various applications, and it is a technology that is a mandatory need in this world as of now. Before the correct implementation of this technology we have dependent on writing texts with our own hands that result in errors. It's difficult to store, access physical data and process the data in efficient manner. Manually it is needed to update, and labor is required in order to maintain proper organization of the data. Since for long time we have encountered a severe loss of data because of the traditional method of storing data.

## 2. PROGRAMMING INTERFACE

In the development of this project we have used PYTHON version 3 as language to develop the software in order to meet the project requirements. For the initial development and exploration, we have used SPYDER as an ide and then later we have shifted to PYCHARM another ide which supports Python and makes the task completion ease. Various other libraries and pre-trained models we have used include Pytesseract, OS, Gtts (Google Text to Speech) etc.

These Programming algorithms have well defined models that are trained with some samples. We use this algorithms and models to increase the accuracy of the result. The result will be dependent on the amount of training given to the model to get better output.

## 3. PROPOSED SYSTEM

### 3.1 BLOCK DIAGRAM

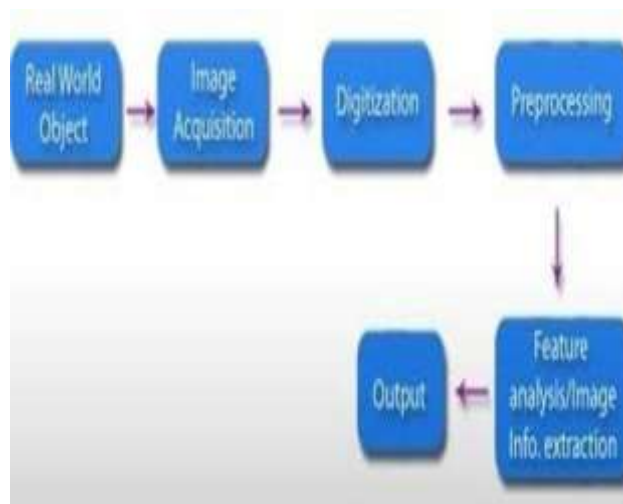


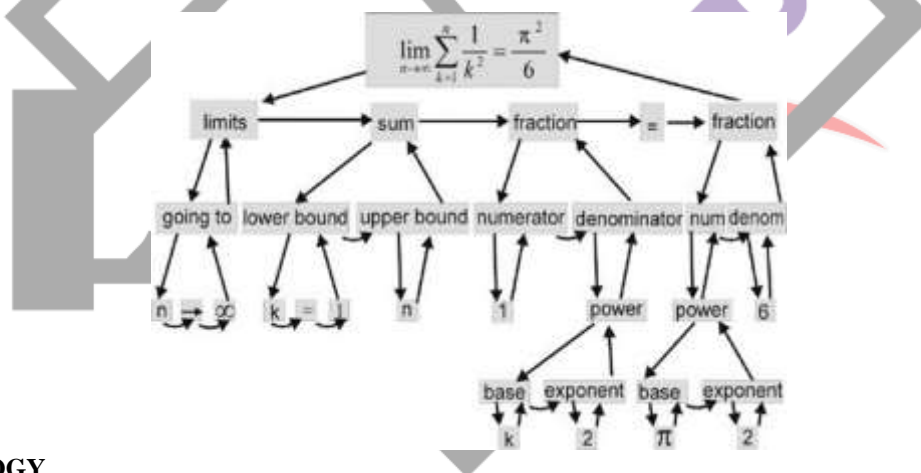
Fig.1 Block diagram

### 3.2 ABBREVIATIONS AND ACRONYMS

Some of the abbreviations and acronyms used in this work are:

1. .py: Extension used for python scripts.
2. GTTS: Google Text To Speech.
3. NLP: Natural Language Processing.
4. OCR: Optical Character Recognition.
5. PIL: Python Imaging Library

### 3.3 MATH BEHIND RECOGNITION



## 4. METHODOLOGY

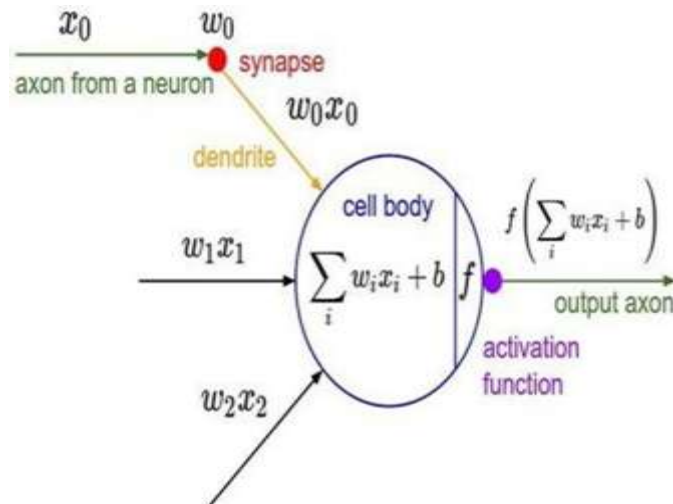
- A. Recognition strategies heavily depends on the nature of the data to be recognized.
- B. In the cursive case, the problem is made complex by the fact that the writing is fundamentally ambiguous
- C. As the letters in the word are generally linked together, poorly written and may even be missing.
- D. On the contrary, hand printed word recognition is more related to printed word recognition, the individual letters composing the word being usually much easier to isolate and to identify.

The Algorithm that we have used in mainly based on the Pretrained Models, these models helps us to get the results required within less time.

These pretrained model of Tesseract and GTTS are used in this research. This research mainly concentrates on the working procedure of pretrained models and their application to solve problems. These Model intern uses Neural Network, Decision Trees to solve and segregate the Characters and words.

#### 4.1 Decision Tree

These trees flow the approach of trial and error method, solves every possible way and conclude the decision. These models improve their accuracy for every time they used to detect the outputs.



Equations:

Gini Index :

$$I_G = 1 - \sum_{j=1}^c p_j^2$$

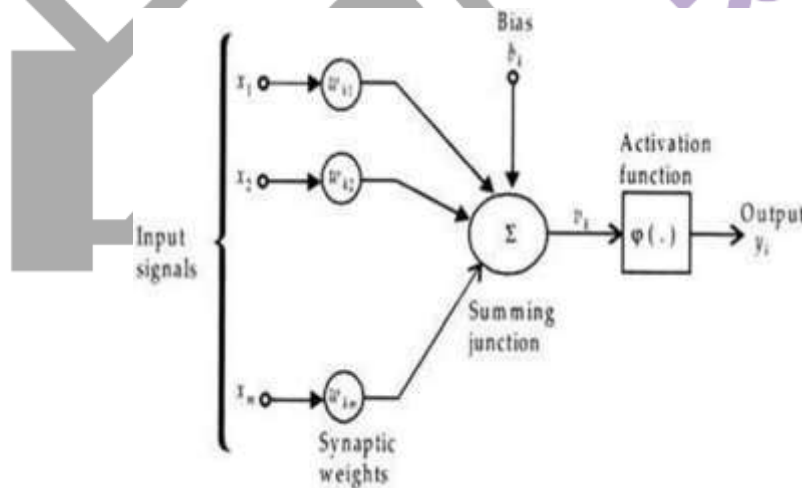
Entropy :

$$I_H = - \sum_{j=1}^c p_j \log_2(p_j)$$

Where p is the proportional of samples belongs to a class c, for a particular node. Based on these Character selection is done.

#### 4.2. Neural Network Model

Neural Networks are artificial Networks that are working same the Human Neurons, but this carry information in form of weights. Different equations hold for the different situations or types to use the Threshold Equation Architecture is:



These are the Weighted sum of the Neurons passing to threshold gives the output equations:



By using these Types of Formulae and we find the Threshold value and feature extraction matching. This is useful to conclude the handwritten text into characters of different types by taking the features of text and then the constructing the real time system.

The Characters recognise and word recognise were there and constructed them with help of neural network algorithm but the data is segregated or accepted by combination algorithms in pretrained models. These Time saving techniques are helpful for industrial in order to solve the problem in very efficient way.

Layer wise Dividing and comparing to database original characters and determine the identity of character and these a combination of the words and then convert file to text format and mp3 format.

## 5. RECOGNITION STRATEGIES

Recognition heavily depends upon Python Text to Speech in python, there a score of API's available to convert text to speech such as gTTS (Google Text to Speech API), eSpeakNG, pyttsx3 et. The gTTS convert the input text in to audio and saved as mp3 file for speaking purpose. It works only when the internet is connected and it has a very natural sound. The cross platform and offline TTS library is pyttsx3 which works for both python versions. It supports different speech engines like espeak, sapi5, nsss. The eSpeakNG is an open source, small size and multiple language speech synthesizer supporting python.

Output-1

Hope you have done it

Process finished with exit code 0

This is the Text Format Output

Output-2:

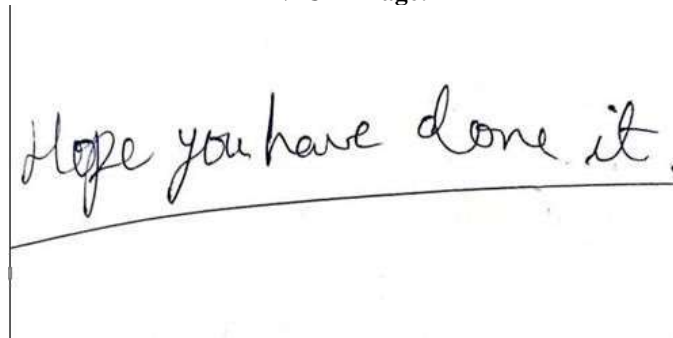


This is second output as a speech signal form

## 7. RESULT

For the Given inputs in various forms we have developed the model for the research and the input output is noted.

**INPUT Image:**



These input images which we have produced has the 2 types of the outputs out of which the first one being output as computerized text based output and the other one being speech signal of the recognized text using the speakers of the computer or the attached ones to the system.

## 6. FUNCTIONS

Some of the important functions used in this project are:

- **get\_tesseract\_version** Returns the Tesseract version installed in the system.
- **image\_to\_string** Returns the result of a Tesseract OCR run on the image to string
- **image\_to\_boxes** Returns result containing recognized characters and their box boundaries
- **image\_to\_data** Returns result containing box boundaries, confidences, and other information. Requires Tesseract 3.05+.

**image\_to\_osd** Returns result containing information about orientation and script detection

- **run\_and\_get\_output** Returns the raw output from Tesseract OCR. Gives a bit more control over the parameters that are sent to tesseract.

## 8. ANALYSIS

In the input image out of 17 characters, our model has recognized and processed 17 characters out of which 16 were correctly recognized and 1 was faulty. Hence with an accuracy of 94% for this input and an overall accuracy of 92.7%. These Results are accurate and hypothetical results that have good accuracy and clear voice file which helps us to hear after the Subject.

## 9. CONCLUSION

Handwritten character recognition is a complex problem, which is not easily solvable. The necessity is around dataset and database. This model is built to analyze the text we have written and convert it in Computer Text and Voice formats. This application is applicable in many sectors of healthcare and Consumer sector. This type of model used in health application can save understanding perspectives of people and store each and every record digitally.

## REFERENCES

- [1] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, "Emnist: an extension of mnist to handwritten letters," arXiv preprint arXiv:1702.05373, 2017.
- [2] L. Eikvil, "Optical character recognition," citeseer. ist.psu.edu/142042.html, 1993.
- [3] Singh, Sameer, Mark Hewitt, "Cursive Digit And Character Recognition on Cedar Database", Pattern Recognition, 2000.Proceedings. 15th international conference on. Vol. 2. IEEE 2000.
- [4] Anita Pal and Davashankar Singh, "Handwritten English Character Recognition Using Neural Network",
- [5] International Journal of Computer Science and Communication, pp: 141- 144, 2011.
- [6] Manoj Sonkusare and Narendra Sahu "A SURVEY ON HANDWRITTEN CHARACTER RECOGNITION
- [7] (HCR) TECHNIQUES FOR ENGLISH ALPHABETS" Advances in Vision Computing: An International Journal (AVC) Vol.3, No.1, March 2016