Optical Character Recognition using Deep Neural Network

Raajkumar G.
PG Scholar
Department of Computer Science and Engineering
PSG College of Technology
Coimbatore, India

Indumathi D., PhD
Assistant Professor (Sl.Gr)
Department of Computer Science and Engineering
PSG College of Technology
Coimbatore, India

ABSTRACT

Optical Character Recognition is a method of extracting text from image. Its main purpose is to make editable document from existing paper or image files. Optical character recognition task involves identifying simple edge detection technique and matching them with predefined patterns. It is a compartment of image recognition and is extensively used as a form of data entry with the input being some sort of printed document or data record like statements from bank, invoices, resume, business card and passport. An existing neural network (LSTM) based OCR model is able to identify text in an image but it could not able to identify the text in a tilted / rotated image. This paper aims at analyzing various text images like blurred image, tilted image and it identifies the text from these images using deep learning models.

General Terms

Image processing, OCR model, long short term memory

Keywords

Deep neural network, LSTM, CNN

1. INTRODUCTION

Nowadays obtaining information and altering the content in pictures which are present within background images are time consumable. Optical Character Recognition (OCR) idea emerges to take care of this issue. OCR works primarily utilizing the AI based computation and it is significant in developing and exploring in man-made brainpower. Optical character acknowledgment permits to change over the characters in printed archives, computerized picture and examined records with the word group. The conventional method for entering the information of the printed reports, checked records and picture into the PC is through console, which is inefficient when there is a huge volume of information. OCR is utilized to enter the information from those records electronically, without the intercession of people. Types of OCR are shown in the Figure 1.

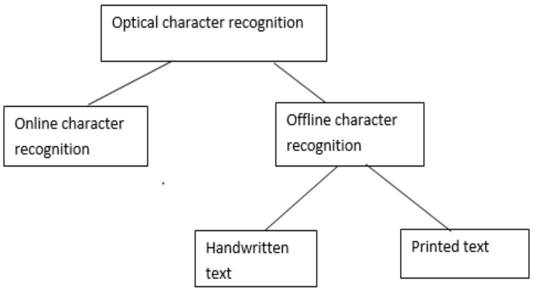
Optical character recognition
Online character recognition
Offline character recognition
Handwritten text
Printed text

Figure 1: OCR Types

2. RELATED WORK

2.1 Text Segmentation

Singh, Raghuraj et al proposed a method to recognize printed script using Optical Character Recognition (OCR). When a document is placed in an arbitrary angle, it would appear on the computer monitor at the same angle, when the document is scanned. So there is a need to correct this skew angle. Since incorrectly segmented characters results in poor recognition, recognition rate also suffers. This paper deals with an algorithm to correct this skew [1]. The algorithm segments the text in the image document into character, words and lines. By converting the image from spatial domain to frequency domain and looking at the frequency distribution, this strategy is accomplished. This method finds the total no. of lines, total no. of words from the input text document. It also determines the no. of words in a specific line. This method uses only good quality of printed/handwritten document without any overlapping or broken characters.

2.2 Image Segmentation

C.N.Anagnostopoulos et al proposed a Computer vision and character recognition algorithm based on image segmentation technique for vehicle license plate identification [2]. This is primarily used as an Intelligent Infrastructure system for electronic toll and parking fee payment, access control system for monitoring unauthorized vehicle entering private areas. The segmentation technique called Sliding Concentric Windows is used to segment the Target Area and also to detect plaques of various sizes and positions [3]. The segmentation module could handle more than 1 license plate at the same time with low computational needs and size of the license plate that can be recognized by the license plate recognition algorithm is said to 16*46 pixels experimentally. License plates that are less than this dimension will be rejected by the algorithm. Due to varying light levels over a 24 hour period and the subsequent physical appearance of the plates, the discrimination analysis of the algorithm suffered.

2.3 Convolution Neural Network for Recognition of Characters

Shahina, Ajward et al proposed a method to deal with an OCR algorithm for digitizing the Sinhala language characters spoken in Sri Lanka. The project is divided into two phases, namely: identification of characters using OCR and extraction of document layout information. The first step identifies the connected details in the picture, extracts the features of these connected details and not identifies predetermined characters in the neural network. The second phase, extracts features like layout or font of characters. The identified characters from phase 1 and its features in phase 2 are combined to reproduce the original document in RTF format [4]. Character recognition preserving the layout of the original document was tested successfully in the Sinhala language. An editable file was obtained from the scanned image with preservation of original contents using the various software tool. Character recognition due to Merging of characters is avoided. Font attributes, intensity values in the document are not considered. R.Anil et al focuses on classification of Malayalam character images using a Convolutional Neural Network (CNN), with algorithm for gradient-based learning and back propagation. Characters in Malayalam are mainly uncase with structural symmetry and recognizing this is a challenge. This paper makes use of a multistage CNN architecture. The first stage does the recognition and is called the outer classification while in the second stage, the misclassified characters from the first are classified using a multi class Support Vector Machine, called the inner classification. Most of the misclassified characters are learned using this network. The accuracies for the outer classification is found to be 92% while it is in the range of 99-100% for inner classification using SVM.

Prashanth Vijayaraghavan et al experimented convolutional neural network architecture with different regularization techniques for character classification in Tamil language. ConvNets learn a unique set of features automatically in a hierarchical fashion [5]. Using stochastic pooling, probabilistic weighted pooling, and local contrast normalization, ConvNetJS library for learning features is used to create a new state of the art 94.4% accuracy on the IWFHR-10 dataset.

R. Deepa et al proposed an image classification using CNN. The text is then extracted from the secret image using Tesseract, which is implemented using a reconnaissance engine based on Long Short-Term Memory (LSTM) [6]. The LSTM networks are Recurrent Neural Network modules. On very large data sets the CNN performs better by solving the issue of over fitting. Alternatively, extraction of single line text is replaced by extraction of multiple line text. Therefore, by integrating a large dataset and increasing the number of epochs, the precision is increased. Furthermore, a technique for trial-and-error is used to evaluate the number of convolution and pooling layers with the number of nodes in each row. Finally, CNNs use very little preprocessing compared to other algorithms for the image classification.

There are a lot of algorithms that are being used for image classification. Features are created from the image and those features are fed into a classification algorithm like SVM. Some algorithms used the pixel value of the image as a feature vector too. To give an example, SVM could be trained with 784 features where each feature is the pixel value for a 28*28 image.CNN can be thought of automatic feature extractor from the image Other classification algorithm uses pixel vector where the spatial interaction between pixel is lost. CNN effectively uses adjacent pixel information to downsample the image first and then use the prediction layer at the end.

3. PROPOSED SYSTEM

This method focuses on getting image input and uses Convolutional Neural Networks (CNN) to find text from the images.

Deep learning is the field of computer science where the machines be trained by themselves to imitate the human brain. A deep learning approach consists of two steps – training and testing. During training huge amounts of data will be labelled and matching characteristics will be recognized. During the testing phase, the model makes prediction and labels unknown data using knowledge it gained during training.

Convolution Neural Network is one of the main technologies for recognizing images, classifying images, detecting objects, recognizing faces, etc. CNN uses comparatively less pre-processing when compared with other image processing algorithms. In CNN, the input is an image and convolutions with various kernels take place in each layer to produce the convolved image as the input into the next layers. There are often other types of layers in better convolutional layers such as pooling layers that mine the significant information/features from the previous layer’s convolutions.

Deep neural networks can be used for solving OCR problems be combining tasks of localizing text in an image along with recognizing and understanding the text. Functional block diagram are show in figure 2.

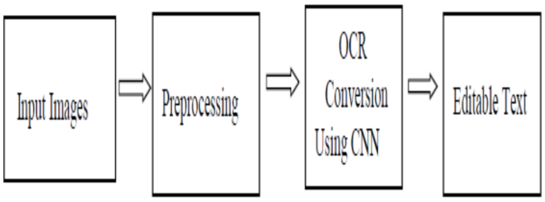

Figure 2: Functional Block Diagram

The input data to this application is the image captured through camera or mobile, pdf, document or scanned document which is to be recognized by the OCR.

3.1 Pre-processing

Following are the steps of pre-processing. These removes noise from the images and also enhance the quality of an image.

3.1.1 Filtering

Separating evacuates clamor and decrease misleading focuses brought about by uneven surface and low quality of the picture. The channel is of recurrence area channel and space area channel. In recurrence channel the high or low recurrence segments are sifted. In unique space channel the tasks are legitimately applied on the pixels of a picture. The Gaussian channel is utilized to decrease Gaussian commotion. The convolution idea is utilized. The predefined veil that is lattice is available and it is superimposed over the first picture network and changes the estimation of each by taking normal.

3.1.2 Morphological operations

Morphological tasks are dilation and disintegration. Dilation adds pixels to limits of a picture so the holes among the characters can be filled and disintegration evacuates pixels to smooth forms and prune wild focuses.

The thickening or diminishing off a double picture is constrained by the organizing component which is the network containing 0,s and 1's. The focal point of organizing component is starting point. On the off chance that the disintegration is trailed by enlargement it is opening of a picture the other way around is shutting of a picture. Disintegration and Dilation uses hit or miss change. Miss speaks to no match with organizing component and hit speaks to coordinating of at least one pixel and root is supplanted by 1.

3.1.3 Noise modeling

Noise modeling is done to evacuate salt and pepper noise. For picture the incentive for salt commotion is near 255 and the incentive for pepper noise is near 0. The commotion can be diminished by utilizing non direct and straight channels. The middle channel is utilized here to expel salt and pepper noise. The convolution bit is utilized and middle worth is determined in convolution process. The middle channel expels noise without influencing edges. Salt and pepper noise is caused due to random bit error in communication.

3.1.4 Normalization

To minimize differences and scale deformation, standardization is used. It is referred to as bringing the image to the normal state in general normalization. The image that has been being tilled somewhere or it may be skewed and the image may be slanted. The normalization of the skew is to detect the skew that rotates the image until it is horizontal. The closest cluster of neighbors is used to extract.

3.1.5 Segmentation

The division is done to put the words into individual characters. By parting the words into characters the acknowledgment is made simple for the recognizer. To portion the characters, utilize the maximally steady external area extractor accessible in OpenCV. MSER is a component locator. It recognizes the item in a picture. Henceforth the characters in a picture is recognized and portioned. MSER discovers the segments associated and estimated the removed locale and spare the areas as highlights.

3.2 OCR Conversion

The image input is segmented into characters and then transferred to recognize through CNN and the recognized character is arranged to replicate the text in the image. The following steps are image segmentation, segmented character data generation, CNN model to train character classification.

3.3 Convolution Neural Network

Convolution Neural Network (CNN) is one of the most important groups to do image recognition, classification of images.

CNN object recognition captures, processes and classifies an input image under certain categories. Computers view an image source as an array of pixels and that depends on the image resolution. It will see (h x w x d) (h= height, w= width, d= Image resolution-based dimension). So, e.g. An image of 6 x 6 x 3 RGB matrix array (3 refers to RGB values) and a grayscale matrix image of 4 x 4 x 1 array. Neural network with LSTM layer is used. Figure 3 describes the procedure for performing OCR using deep learning.

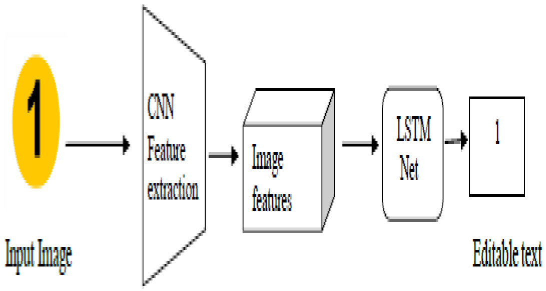

Figure 3: OCR using Deep Learning

3.3.1 CNN Architecture

A Convolution Neural Network consists of multiple layers built to identify visual patterns specifically. CNN architecture are explain in figure 4 as follows.

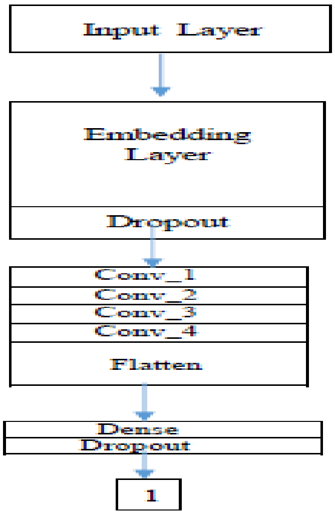

Figure 4: CNN Architecture

- Input Layer: The given input image is represented as matrix of pixels.
- Embedding layer: Convolution layer where features are extracted.
- Dropout: To avoid over fitting dropout layer drops out few features and make the model to learn with few features.
- Flattening Layer: Soft max fully connected layer.
- 1: represents the final output image.

3.3.2 Long Short Term Memory

Long Short Term Memory is an RNN version. It has the support to store the previous input for longer time so that the present input will depend on the previous output which is stored in the memory. LSTM architecture is explained in figure 5 as follows.

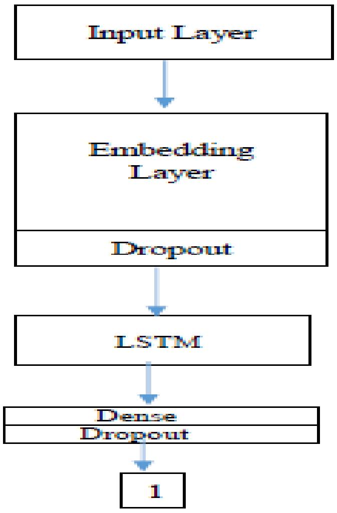

Figure 5: LSTM Architecture

- Input Layer: Feeds the input image
- Embedding Layer: Layer used to extract features
- Dropout: To drop few neurons to reduce over fitting
- LSTM: It consists two gates update and reset gate
- Dense Layer: Fully connected layer
- 1: The final output image

4. EXPERIMENTS

4.1 Implementation

Procedure for Pytesseract Implementation

Step 1: Pre-process image data like converting to grayscale.

Step 2: Detect lines, words and characters from the image.

Step 3: Generate a list of candidate rated characters based on a collection of trained data.

Step 4: After the characters have been recognized, they process it and pick the best characters based on trust from the previous step and the language data. Data on the language include dictionary, grammar rules, etc.

4.2 Performance analysis

To evaluate the performance of the system, four metrics have been used to assess device efficiency, namely precision, accuracy, recall and F-measure. The testing dataset consists of 1000 images among which 860 images where correctly converted to text while remaining 140 images are not properly identified due to the image quality and inclination of image. The confusion matrix table is given in table 1.

Table 1: Confusion Matrix

Confusion Matrix	Correctly Converted (Class1) Predicted	Not Correctly Converted (Class2) Predicted
Correctly Converted(Class1) Actual	TP	FN
Not Correctly Converted(Class2) Actual	FP	TN

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{F measure} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

4.3 Experimental result

From the result it is clear that pytesseract tool is giving better accuracy compared to the machine learning model SVM. The main reason for this betterment in accuracy is because pytesseract is built using the advanced deep learning models including CNN which outperforms the standard SVM model when used for images.

Table 2: performance comparison with respect to training loss and accuracy

	Support Vector Machine	pytesseract
Train loss	23	14
Train accuracy	77	86

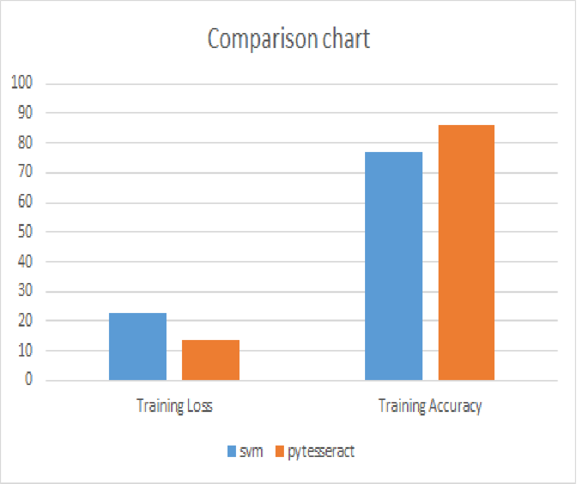

Figure 6: Comparison between SVM and Pytesseract

5. CONCLUSION

The work proposed utilizes the deep learning ideas to the OCR problem in an efficient way. The various pre-processing techniques are carried out to handle text image with noise. It is observed that the traditional existing machine learning technique was unable to identify the text when the image is tilted at particular angle. But pytesseract utilizes deep learning architecture namely CNN and LSTM to improve the accuracy to 86%.

6. REFERENCES

[1] Singh, Raghuraj, C. S. Yadav, Prabhat Verma, and Vibhash Yadav. "Optical character recognition (OCR) for printed devnagari script using artificial neural network." International Journal of Computer Science & Communication 1, no. 1 (2010): 91-95.

[2] C.N. Anagnostopoulos, V. Loumos, E. Kayafas, “A License Plate Recognition Algorithm for Intelligent Transportation System applications”, IEEE Transactions on Intelligent Transportation Systems, Volume: 7, Issue: 3 , September, 2006.

[3] Ajward, Shahina, Nalani Jayasundara, Sarasi Madushika, and Roshan Ragel. "Converting printed Sinhala

documents to formatted editable text." In 2010 Fifth International Conference on Information and Automation for Sustainability, pp. 138-143. IEEE, 2010.

[4] R. Anil, K. Manjusha, S. S. Kumar, and K. P. Soman ,"Convolutional Neural Networks for the Recognition of Malayalam Characters," in Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014, vol. 328, Springer International Publishing, 2015, pp. 493–500.

[5] Vijayaraghavan, Prashanth, and Misha Sra. "Handwritten tamil recognition using a convolutional neural network." In 2018 International Conference on Information, Communication, Engineering and Technology (ICICET), pp. 1-4. 2014.

[6] R. Deepa and K. N. Lalwani, "Image Classification and Text Extraction using Machine Learning," 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2019, pp. 680-684, doi: 10.1109/ICECA.2019.8821936.

[7] Sankaran, Naveen, Aman Neelappa, and C. V. Jawahar. "Devanagari text recognition: A transcription based formulation." In 2013 12th International Conference on Document Analysis and Recognition, pp. 678-682. IEEE, 2013.

[8] Wu, Victor, Raghavan Manmatha, and Edward M. Riseman. "Textfinder: An automatic system to detect and recognize text in images." IEEE Transactions on pattern analysis and machine intelligence 21, no. 11 (1999): 1224-1229.

[9] Kazmi, Wajahat, Ian Nabney, George Vogiatzis, Peter Rose, and Alex Codd. "An Efficient Industrial System for Vehicle Tyre (Tire) Detection and Text Recognition Using Deep Learning." IEEE Transactions on Intelligent Transportation Systems (2020).

[10] Kim, Michael D., and Jun Ueda. "Dynamics-based motion deblurring improves the performance of optical character recognition during fast scanning of a robotic eye." IEEE/ASME Transactions on Mechatronics 23, no. 1 (2018): 491-495.