Research Article

A Survey On Telugu Optical Character Recognition From Digital Images

[1]Srinivasa Rao Dhanikonda, [2]Subhash Chandra N,
[1]Research Scholar, JNTUH Hyderabad and Assistant Professor, GITAM Deemed To Be University, srinivasarao.dhanikonda@gmail.com
[2]Professor, CVRCOE, subhashchandra.n.cse@gmail.com

Article History: Received: 11 January 2021; Accepted: 27 February 2021; Published online: 5 April 2021

Abstract: Images play an essential function in the electronic media to share information. Nowadays, each event is going to be recorded in the arrangement of digital images. Text from the image file won't be in a format on the computer. OCR (Optical Character Recognition) for English vocabulary is well constructed. Currently, there's a requirement of OCR for Indian languages to maintain historical documents composed mainly in Indian languages to arrange publications in the library and for program form processing. OCR for the Telugu language is challenging as consonants and vowels plays a vital role in forming words along with vattus and gunithas. It may be a mixture of vowels and consonants that may form a compound character. This paper presents research on methods utilized in the OCR method for the Telugu Language until today.

Keywords: Text segmentation, Text Extraction, image-based, Document processing, OCR

I. Introduction

There was limited research in the maturation of a complete OCR program for Telugu script. While the access to a massive internet corpus of scanned files warrants the requirement to get the OCR system, the more complex script and agglutinative grammar create the issue hard. Constructing a system that works nicely on real-world files comprising sound and erasure is more complicated. The endeavor of OCR is principally divided into segmentation and recognition. That of another directs the plan of each. The stronger (to sound, erasure, skew, etc.) that the segmentation will be, the simpler the job of this recognizer becomes and vice-versa. The techniques utilized in segmentation are similar through areas. That is because, generally, one connected component (a neighboring area of ink) could be expressed as one unit of text or character. Although this principle applies to the Roman broadcasts with few exceptions, it doesn't hold complicated scripts such as Devanagari and Arabic. Phrases aren't letters; they have been written in a single contiguous slice of ink. The Telugu script consists of intermediate complexity, in which consonant-vowel pairs have been composed as a single unit. The recognition task is split to feature extraction and classification. The former was hand-engineered for a lengthy moment. They train multiple neural networks, and pre-classify an input image based on its aspect ratio and feed it to the corresponding network. It reduces the number of classes that each sub-network needs to learn. But this is likely to increase the error rate, as a failure in pre-classification is not recoverable. The neural network employed is a Hopfield net on a down-sampled vectorized image. Later work on Telugu OCR primarily followed the featurization-classification paradigm. Combinations like ink-based features with the nearest class centroid (Negi, Bhagvati and Krishna, 2001); ink-gradients with nearest neighbours (Lakshmi and Patvardhan, 2002); principal components with support vector machines (Jawahar, Kumar and Kiran, 2003); wavelet features with Hopfield nets (Pujari et al., 2004) were used. More recent work in this field (Kumar et al., 2011) focuses on improving the supporting modules like segmentation, skew-correction and language modeling. While our work was under review, Google Drive added an OCR functionality that works for Telugu and many other world languages. Although its details are not public, it seems to be based on their Tesseract multilingual OCR system (Smith, 2007) augmented with neural networks.

II. Related review

Optical character recognition (OCR) has been among the most studied issues in pattern recognition. However, the achievement of CNN's motivated us to utilize them for Telugu character recognition. The first recorded work on OCR to get Telugu could be dated back as early as 1977 from Rajasekharan; also, Deekshatulu utilized features that synthesize the curves that follow a letter also compare that this encoding using a group of predefined templates [12]. It managed to spot 50 primitive features also suggests that a two-stage syntax-aided character recognition program. The first effort to use neural networks first created with M.B. Sukhaswami et al., which compels several neural systems and pre-classifies a picture based on its characteristic ratio. It then feeds it into the corresponding system [17]. It revealed that the robustness of a Hopfield system to understand noisy Telugu characters. Afterwards, work on Telugu OCR mostly adopted closely by the feature classification paradigm. Jawahar et al. [14] describe a bilingual Hindi-Telugu OCR for documents containing Hindi and Telugu text. It is based on Principal Component

analysis followed by support vector regression. They report an overall accuracy of 96.7% over an independent test set. They perform character level segmentation offline by their data collecting tools. However, they have only considered 330 distinct classes. The work by Rakesh and Trevor [5] on Telugu OCR using convolutional neural networks is also fascinating. They used 50 fonts in four styles for training data, each image of size 48x48. However, they not consider all possible outputs (only 457 classes) of CNN. Kunte and Samuel work on Kannada OCR employs a two stage classification system similar to our approach. They have first used wavelets for feature extraction and then two-stage multi-layer perceptrons for the task of classification. They have divided the characters into separate subclasses but have not considered all possible combinations.

For Telugu text in printed form, Arun K Pujari et al. [15] in 2002 proposed an OCR system. Text is scanned in the form of a grayscale image. Horizontal and vertical projection techniques are used for line and word segmentation. The zero-padding technique is used to convert characters into a fixed size. Wavelet analysis is used for obtaining information of images at different scales like 32x32. Performed 2-dimensional filtering so that 32x32 image is converted into 4, 8x8 images, which gives the average image. Then by using thresholding, convert images to binary which gives 64 bits, and these are referred to as signature of the input symbol. For recognizing symbols, Dynamic Neural Network is used in which every node in the network is the Hopfield network. This method does not depend on font and shape. Some symbols dha, dhaa, na, and this technique does not correctly recognize sa. C. Vasantha Lakshmi et al. [16] proposed an OCR system in 2003 for printed text in Telugu. The scanned image is converted to a binary scale, and noise is removed through rectification. Skew is corrected, and then lines, words and symbols are extracted from text segmentation. Pre-Classification of each symbol by size property to compute real-valued direction features. Neural recognizers are used for classification, and finally, information associations of basic symbols for a word are outputted. Testing is performed on one lakh symbols, which resulted in 99% accuracy for DeskJet prints and laser prints using additional logic. OCR system for printed characters in Telugu was proposed by Negi et al. [13] in 2003. Nonlinear normalization is performed using a modified crossing count, which enhances the features of the input image. In different zones, pixel densities are used for searching the initial candidate of input glyph. If the candidates are found in-conclusive, they are passed through another stage where input image cavities are analyzed. Template matching is done based on Euclidean distance on normalized characters for nonlinear shapes which are controllable. This technique obtained correct results for 1463 glyphs out of 1500 glyphs which are collected from the magazine.

III. Problem statement

Telugu is a Dravidian language with over 80 million speakers, mainly in the southern India state of Andhra Pradesh. It has a robust consonant-vowel structure, i.e. most consonants are immediately followed by a vowel. A consonant and a vowel combine to give a syllable. For example, the syllable ka in the word sakarma is one such entity, and ra in the exact figure is another. Each such syllable is written as one contiguous ligature. This alpha syllabic form of writing is called an abugida as opposed to an alphabet. There are 16 vowels and 37 consonants that combine to give over 500 simple syllables. Of them, about 400 are commonly used. There are nearly sixty other symbols, including vowel-less consonants, punctuation and numbers. This brings the total number of symbols used in everyday writing to approximately 460. Figure 1 demonstrates the Telugu calligraphy via an example.

IV. Research study

Shobha Rani N. et al. (2015) the proposed algorithm for text line segmentation of Telugu document images consists of three significant steps. The first step generates a fringe map. In the second step, Peak fringe numbers (PFNs) are located in the fringe map. The PFNs between text lines are determined by performing a filtering operation. Identifying PFNs that belong to an adjacent line and generating a segmenting path is not easy because the filtering operation leaves gaps. Hence, a broad region is constructed to cover the consonant moodier of a line and vowel modifiers of the following line (the overlapping and touching components of adjacent text lines). These regions cluster the PFNs between adjacent lines. In the last step, a segmenting path between lines is generated by joining the region's PFNs.

Raashid MALIK et al. (2007) our initial objective is, therefore, to find and isolate the text in a scene. From a practical perspective, an extension of this work can lead to machine reading of highway signs such as exits, speed limits or cautions. It may also make barcoding superfluous since machines would able to read labels on merchandise directly. As graphical, textual information has been increasing; text extraction is a necessary procedure for recognition steps. Even though many approaches have been addressed now, the majority of them cannot solve text extraction problems. What surroundings make it complicated? Variants of font, style, size, special symbol, multilingual environment and performing on the binary images always hinder us from exploring the final gist. Thus

we propose the scheme to root out the above restrictions using an image, based on edge detection, histogram and width to height ratios, of which input is grey images, not binary. We have shown expected results by experiments, font style,size language independently, and text embedded into the background image.

Vasudev T. et al. (2016) proposed a technique for feature extraction and classification of Telugu handwritten script based on customized template matching approach to support caching technique for better performance. The caching technique is implemented using the central database with a cache database, maintaining the frequently used character templates for a set of all character templates. The XML database is used for defining the classes for various character templates, and the class representations are provided using a novel class structure designed based on XML tags. The proposed system exhibits the recognition efficiency o Otsu's n our test dataset with an overall accuracy of 83.55% for handwritten characters. The feature extraction by shape matching in conjunction with correlation-based classification has provided satisfactory results. The inclusion of REQUIRED_SIMILARITY_MEASURE to find a suitable match between the test template and professional template significantly reduces the conventional template matching technique's worst-case time complexity. However, there are few cases of misrecognition, especially for some of the confusing character pairs [ఎ, ఏ] [ఐ, ఒ] [ఒ, ఓ] etc. The confusing characters possess minor differences in their structural orientation, but the use of REQUIRED_SIMILARITY_MEASURE improves the template matching algorithm's performance. However, the design of an efficient post-processing methodology can correct the recognition errors with prime regard to confusing character pairs. This suffices the reliability of the system, which is currently under investigation. In addition to this, the technique of caching improves the overall performance of the classification approach. The proposed system's experimental results still improved by replacing template matching with an efficient feature extraction technique to reach high recognition accuracy.
The significant steps to identify character regions are listed below:

Word region segmentation: Segmentation for documents is well developed in the case of English. Many papers in the literature use Maximally Stable Extremal Regions (MSER) to segmentation characters in English. This solution cannot be directly extended to Telugu because most of the dheergam and vattus get separated if MSER is directly applied. Due to these reasons, we made minor modifications to MSER to take into consideration dheergas and vattus.

Binarization: Having a robust binarization algorithm is essential for any OCR system. We have observed that images were taken by phone often have regions of shadow due to improper lighting conditions. Most of the previous works have used Otsu's algorithm for binarizing their text document. Otsu's thresholding has given us poor results because the shadow region, due to similarity in intensity level with that of characters, was binarized as a foreground region. We used a modified version of Otsu's thresholding for our binarization. The first step is to remove the noise present in the image. We have applied the morphological closing algorithm for that task. We have also tried to use a mode based thresholding algorithm. It was removing false predictions to some extent but was also decreasing the thickness of characters. After denoising, the logical OR is taken between the denoised image and the mode-based thresholding algorithm.

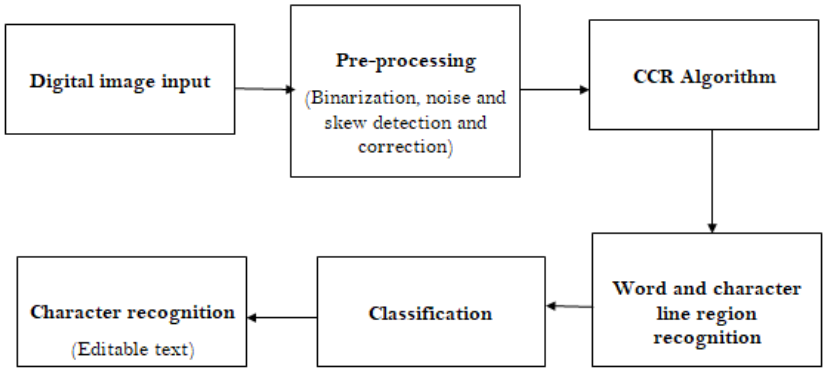

Figure 1: proposed OCR model with CCR algorithm

Classification: The performance of an OCR system depends hugely on the performance of its classifier. Previous works on Telugu OCR have done the character level segmentation based on histogram along X and Y direction. Assuming that the histogram method for segmentation would work perfectly, they have used SVM-based classifiers to classify characters. But we have observed that the histogram method fails to segment out the vattu and main character together properly in real scenarios. It also fails when the characters are rotated or share a common region projected on X-axis or X-axis. Inspired by the success of deep learning, we have explored CNNs to classify the characters and proposed a new architecture for the same. CNN is a type of feed-forward neural network or a sequence of multiple layers which is inspired by biological processes. It eliminates the dependency on hand-crafted features and directly learns useful features from the data itself. It combines both feature extractor and classifier and mainly consists of convolutional, pooling and fully connected layers.
Character level segmentation: Every character from the word is segmented using the Connected Components Algorithm in Image Processing. After binarization of the image, we apply the algorithm to separate all the letters and vattus. The letters and vattus are returned in the form of components(groups of binary pixels). In the process, little blobs are also removed from the components. In the Telugu language, some vattus are not connected with the base letter itself. To connect the base letter with its vattu, we measured the overlapping distance in the horizontal and vertical direction and grouped them.

V. Conclusion

OCR for the Telugu Language is the current area of research due to its enormous applications. OCR for printed text is developing now but needs improvisation in the processing stage and for handling broken characters and segmentation as none of the above techniques can achieve 99% accuracy. Segmentation and classification are significant challenges in recognition of Telugu characters. We studied different frameworks for segmentation and classifications. Still, there is a need to improve the segmentation algorithms to get more accuracy. A hybrid model must be proposed to solve the problems in the existing researches. The segmentation algorithm must be improved to the extent that every character is segmented together with its vattu and gunintham. Network accuracy can be further improved to make the classifier better.

VI. Future work

The same work optical character recognition will be done by proposing "A Dynamic Line Segmentation Technique for Extracting Telugu Characters from Document Images" to solve the problems in existing systems. This work follows the process of Binarization, Skewness correction, Projection Profiles (Horizontal and Vertical), Calculate the distance between pixels in projection profiles, Word and Character segmentations.

References

1. Babu, Arja Rajesh. “OCR for Printed Telugu Documents.” Diss. Indian Institute of Technology Bombay Mumbai, 2014.
2. Hu, Peifeng, et al. “Recognition of gray character using gabor filters.” Information Fusion, 2002. Proceedings of the Fifth International Conference on. Vol. 1. IEEE, 2002.
3. Ramanathan, R., et al. “Robust feature extraction technique for optical character recognition.” Advances in Computing, Control, & Telecommunication Technologies, 2009. ACT’09. International Conference on. IEEE, 2009.
4. Rao, P. V. S., and T. M. Ajitha. “Telugu script recognition-a feature based approach.” Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on. Vol. 1. IEEE, 1995.
5. Achanta, Rakesh, and Trevor Hastie. “Telugu OCR Framework using Deep Learning.” arXiv preprint arXiv:1509.05962 (2015).
6. LeCun, Yann, et al. “Gradient-based learning applied to document recognition.” Proceedings of the IEEE 86.11 (1998): 2278-2324.
7. Bastien, Frédéric, et al. “Theano: new features and speed improvements.” arXiv preprint arXiv:1211.5590 (2012).
8. Cire¸san, Dan, and Ueli Meier. “Multi-column deep neural networks for offline handwritten Chinese character classification.” Neural Networks (IJCNN), 2015 International Joint Conference on. IEEE, 2015.
9. Otsu, Nobuyuki. “A threshold selection method from gray-level histograms.” IEEE transactions on systems, man, and cybernetics 9.1 (1979): 62-66.
10. Wolf, Christian, J-M. Jolion, and Francoise Chassaing. “Text localization, enhancement and binarization in multimedia documents.” Pattern Recognition, 2002. Proceedings. 16th International Conference on. Vol. 2. IEEE, 2002.

11. Karatzas, Dimosthenis, et al. “ICDAR 2015 competition on robust reading.” Document Analysis and Recognition (ICDAR), 2015 13th International Conference on. IEEE, 2015.
12. Rajasekaran, S. N. S., and B. L. Deekshatulu. “Recognition of printed Telugu characters.” Computer graphics and image processing 6.4 (1977): 335-360.
13. Negi, Atul, Chakravarthy Bhagvati, and B. Krishna. “An OCR system for Telugu.” Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on. IEEE, 2001.
14. Jawahar, C. V., MNSSK Pavan Kumar, and SS Ravi Kiran. “A bilingual OCR for Hindi-Telugu documents and its applications.” Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on. IEEE, 2003.
15. Arun K Pujari, C Dhanunjaya Naidu, BC Jinaga, "An Adaptive Character Recognizer for Telugu Scripts using Multiresolution Analysis and Associative Memory “, ICVGIP, Ahmedabad,2002.
16. C. Vasantha Lakshmi, C. Patwardhan, "High accuracy OCR system for printed Telugu text”, TENCON 2003, Conference on Convergent Technologies for Asia-Pacific region, Volume 4,15-17 October.
17. M Swamy Das and Ram Mohan Rao, "Evaluation of Neural Based Feature Extraction Methods for Printed Telugu OCR System ", Advances in Computer Science and Information Technology, Volume 2, 11, 2015.
18. N. Shobha Rani, Vasudev T, “A performance efficient technique for recognition of Telugu script using template matching”, I. J. Image, Graphics and Signal Processing, Volume 8, 2016.