STATE OF THE ART CHARECTER RECOGNITION USING OCR ENGINE

[1]Jagan Mohan G, [2]Vivek Sunderraj, [3]Naveen Kumar M,
[4]Susang Ramesh, [5]P.Julian

Department of Electronics and Communication Engineering,
Velammal Engineering College,
1,2,3,4 Students, 5 Assistant Professor,
julian@velammal.edu.in

Abstract— In the running world, there is growing demand for the software systems to recognize characters in computer system when information is scanned through paper documents. One simple way to store information in these paper documents in to computer system is to first scan the documents and then store them as IMAGES. But to reuse this information it is very difficult to read the individual contents and searching the contents form these documents line-by-line and word-by-word. The aim of this project is to develop an OCR software for character recognition. OCR is an Optical character recognition and is the mechanical or electronic translation of images of typewritten text (usually captured by a scanner/camera) into machine-editable text. OCR is a field of research in pattern recognition, artificial intelligence and machine vision

Keywords—Tesseract engine, fine-grained classifcation, Text-retrieval, text detection, text saliency, Gaussian Filtering, Grey scale conversion

I. Introduction

Nowadays, a lot of paper documents are transformed to electronic form, which makes information processing easier, like searching, analysis and conversion. Many companies and other institutions decide to digitalize their documents. Working with files is cheaper than processing traditional documents, because there is no space required for document storage. There are three main steps of document digitalization: scanning, indexation (data entry) and presentation of digitalized documents. Researchers proved that the recognition of both barcodes and printed text through Optical Character Recognition or OCR is reliable and significantly accelerates data processing.

On the contrary, it is quite difficult to recognize and acquire images that are quite noisy.
The method that we adopted for this paper is that we have introduced optical character recognition. The first method is to capture the image using a camera and then following it up and giving this image as an input to the software. The software is developed with python programming language as base.

The versatility and the user friendliness of the python language in addition to the number of library functions that it supports makes it an ideal language to design and develop an efficient and robust algorithm which we have made effective use of

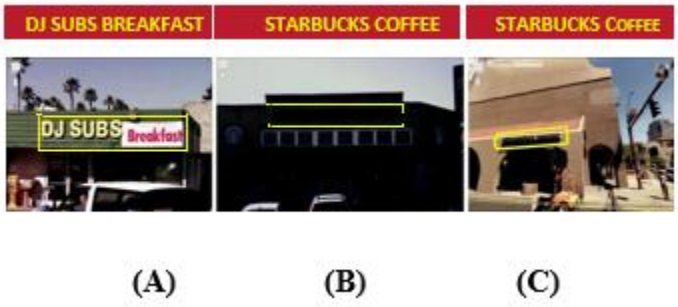


Fig. 1. An example of fine-grained *Building* classification [2]. Visual cues would group (a)-(b) whereas scene text reveals the semantics and clusters (b)-(c).

The 2 major components which we have imbibed into the software that form an essential part of the program are Python Open CV and OCR Engine. The Open CV, an Open source library, which is used to perform image related operations considering the number of images that must be compared to get the desired output is used so as to perform grey scale conversion. OCR stands for Optical Character Recognition. It is one such system that allows us to scan printed, typewritten or handwritten text (numerals, letters or symbols) and/or convert scanned image in to a computer process able format, either in the form of a plain text or a word document.

OCR is used when recreating a similar document in paper as a document in electronic form takes more time. The converted text files take less space than the original image file and can be indexed. Hence the use of OCR adds an advantage to the user who had to deal with conversion of great amount of paper works in to electronic form. In the running world there is a growing demand for the users to convert the printed documents in to electronic documents for maintaining the security of their data. Hence the basic OCR system was invented to convert the data available on papers in to computer process able documents, So that the documents can be editable and reusable.Our proposed system is OCR ON A GRID INFRASTRUCTURE which is a character recognition system that supports recognition of the characters of multiple languages.Once a printed page is in this machine readable text form, one can do all kinds of thing that couldn't do before.
Machine-readable text can also be decoded by screen readers, tools that use speech synthesizers to read out the words on a screen so blind and visually impaired people can understand them. In the 1970s, one of the first major uses of OCR was in a photocopier-like device called the Kurzweil Reading Machine, which could read printed books out loud to blind people.

Text detection methods aim at detecting and generating bounding boxes of words in natural scene images. Text detection methods can be categorized into two classes based on how they search character regions: a con-nected component and a sliding window approach Connected component approaches aim at segmenting characters using pixel similarities, e.g. contrast stroke width and intensity whereas sliding window based approaches search the image over different scales and window sizes to locate character regions. For both methods, word candidates are

detected by further verifying and combining the generated character candidates. To verify and combine character candidates, geometric, structural and appearance properties of text are derived from hand-crafted rules or obtained by learning

In the running world there is a growing demand for the users to convert the printed documents in to electronic documents for maintaining the security of their data. Hence the basic OCR system was invented to convert the data available on papers in to computer process able documents, So that the documents can be editable and reusable. The existing system/the previous system of OCR on a grid infrastructure is just OCR without grid functionality. That is the existing system deals with the homogeneous character recognition or character recognition of single languages.

These various methods that are proposed generally extract structural , appearance and geometric features from candidate regions so as to verify, if a region contains text or not. Generally to have one global pattern it is quite difficult and time consuming setting which would accommodate for all possible text variations in natural images Therefore, it is necessary to tune these parameters for every new alphabet, text style and size. But , in contrast to the existing system, our approach focuses more on background connectivity rather than text regions.

The proposed system is what we call grid infrastructure which eliminates the problem of heterogeneous character recognition. In this context, Grid infrastructure means the infrastructure that supports group of specific set of languages. Thus OCR on a grid infrastructure is multilingual

The existing method does not extract text specific features. It extracts all the regions of the image hence making it less efficient.

Therefore, it does not require any tuning for varying text size, style and orientation. In background information is used to avoid parameter tuning for text binarization in document images. Additionally, state-of-the-art methods combine characters into words by a learning or a rule based approach. However, the information loss at these steps are irreversible. In contrast, we use characters instead of words to represent textual information in the images. Recently, use similar ideas as in object proposals but this time to generate a small set of word candidates. A reduced number of word candidates makes it possible to use more complex classifiers for word recognition. Such work can highly benefit from the proposed text detection method to reduce word box proposals even further.

Visual saliency. The aim of visual saliency detection is to separate attention-driven regions and other regions (e.g. background). In this way, the vast amount of incoming visual data (background) is eliminated. This helps to extract more reliable information because the background is eliminated. Therefore, it is widely used in image processing, for scene classification object recognition and visual search. Saliency for text detection has only recently received some attention. Text in natural scenes is typically designed to attract attention.Rule-based algorithms apply pixel-level image processing to extract text information from predefined text layouts such as character size, aspect ratio, edge density, character structure, color uniformity of text string, etc. Phan et al. [19] analyzed edge pixel density with the Laplacian operator and employed maximum gradient differences to identify text regions. Shivakumara et al. used gradient difference maps and performed global binarization to obtain text regions. Epshtein et al. designed stroke width transforms to

localize text characters. Nikolaou and Papamarkos applied colour reduction to extract text in uniform colours.

In colour-based text segmentation is performed through a Gaussian mixture model for calculating a confidence value for text regions. This type of algorithm tries to define a universal feature descriptor of text. Learning-based algorithms, on the other hand, model text structure and extract representative text features to build text classifiers. Chen and Yuille presented five types of Haarbased block patterns to train text classifiers in an Adaboost learning model. Kim et al. considered text as a specific texture and analyzed the textural features of characters by a support vector machine (SVM) model. Kumar et al. used globally matched wavelet filter responses of text structure as features.

Metal. performed classification of text edges by using histograms of oriented gradients and local binary patterns as local features on the SVM model. Shi eh employed gradient and curvature features to model the grayscale curve for handwritten numeral recognition under a Bayesian discriminant function. In our research group, we have previously developed rule-based algorithms to extract text from scene images. A survey paper about computer- vision-based assistive technologies to help people with visual impairments can be found in .

In each and every system that were proposed earlier, the number of steps needed to tune down the image and get the desired grey scale image were very long. So this increased the system requirements that were needed and also the energy used was more. Although these proposed systems proved to provide an efficiency of 93%, the key essential factor that one is thriving for is the reduction in energy and also increased efficiency There by extracting maximum useable efficiency from minimal present energy.

II. PROPOSED SYSTEM

The system that we propose is a increased system that identifies key areas where the existing system has failed and tries to overcome them Our proposed system is OCR on a grid infrastructure which is a character recognition system that supports recognition of the characters of multiple languages. This feature is what we call grid infrastructure which eliminates the problem of heterogeneous character recognition and supports multiple functionalities to be performed on the document. The multiple functionalities include editing and searching too where as the existing system supports only editing of the document. In this context, Grid infrastructure means the Infrastructure that supports group of specific set of languages. Thus OCR on a grid infrastructure is multi- lingual.

However this assumption might not be met in practical scenarios since such a priori information is not available in the receiving end of the proposed model.

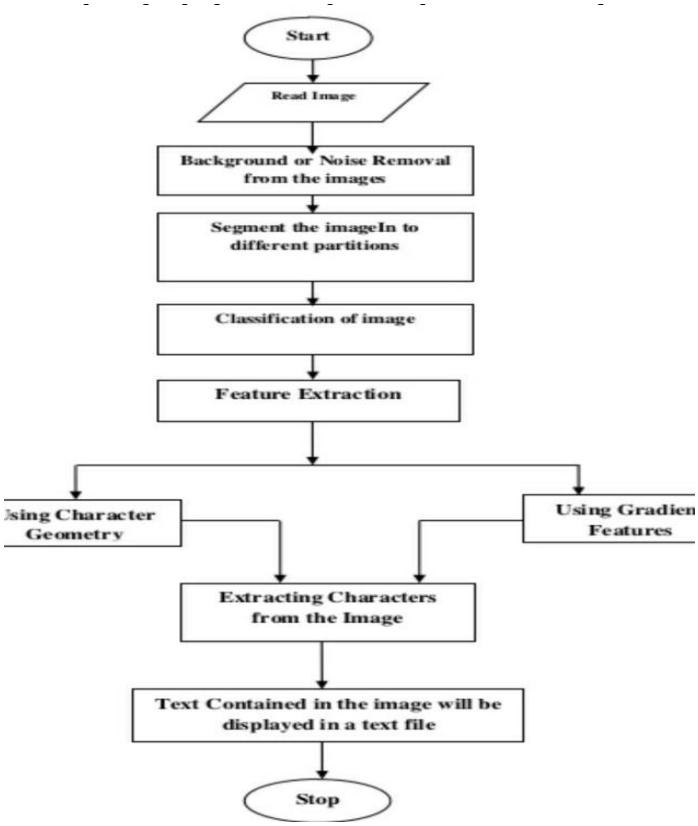
So In order to facilitate such a model, the number of languages that need to be detected must be increased . So we have imbibed an essential component in the code called as Tesseract OCR Engine.

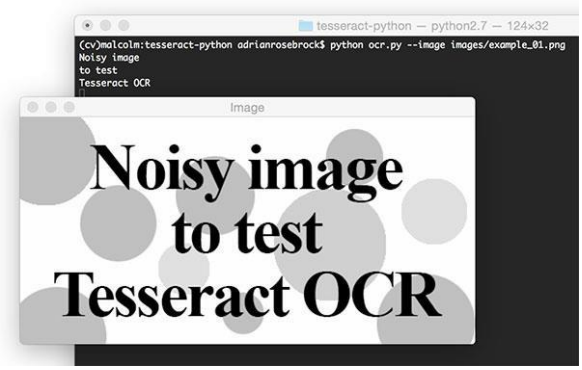
Hence, text boundaries usually correspond to strong intensity changes. Therefore, we propose to select initial background seeds fromnon-salientpixels and grow these seeds using connectivity of background. These seeds will grow until strong intensity changes are reached e.g., text/background transitions. Background seeds form connectivity between all

pixels except those that belong to the text regions. An illustration of seed growing is shown in Fig. 3a. Blue dots represent initial background seeds whereas red lines represent the connectivity path formed by these initial seeds (blue dots). To form connectivity between background pixels, we use conditional dilation (δ). Conditional dilation is a morphological operation where the dilation of a marker image is conditioned by a mask image (I). In this work, we use the image consisting of only background seeds as marker image γ and gray-level image as mask image. The conditioning is performed by intersecting dilated marker image with mask image, described as:

Text can appear on unknown background with unknown text size, style and orientation in natural scene images. It is difficult to have one global parameter setting which would accommodate for all these variations in text [62]. Therefore, we tackle the problem of detecting text from a different point of view. Rather than asking "what is the property of scene text", we ask the question "what is the property of scene background". Keeping in mind that scene text is highly contrasted with background, answering this question would also reveal the location of scene text. As a result, the proposed method would not require any tuning for varying text size, style and orientation. Moreover, eliminating background to infer text location has additional benefits ,the search spaces reduced allowing the extraction of computationally intensive features for character recognition, background clutter is removed reducing false text detections/recognitions.

BLOCK DIAGRAM





The block diagram consists of first extracting the image file and converting into grey scale This is done through image acquisition through gaussian filters. The OCR tesseract engine plays a key role in this crucial junction the conversion of image to grey scale needs to be performed so as to reduce the noise which is the main form of hindrance that affects the output of the system .

Advantage:

- The primary advantage is to speed up the process of character recognition in document processing. As a result the system can process huge number of documents with-in less time and hence saves the time.
- Since our character recognition is based on a grid infrastructure, it aims to recognize multiple heterogeneous characters that belong to different universal languages with different font properties and alignments.

III. WORKING OF TESSERACT

An image with the text is given as input to the Tesseract engine that is command based tool. Then it is processed by Tesseract command as shown in fig. Tesseract command takes two arguments: First argument is image file name that contains text and second argument is output text file in which, extracted text is stored. The output file extension is given as .txt by Tesseract, so no need to specify the file extension while specifying the output file name as a second argument in Tesseract command.

As Tesseract supports various languages, the language training data file must be kept in the tessdata folder. In this research, the purpose is to extract English text from the images so we have kept only English language file in the tessdata folder. After processing is completed, the content of the output file shown in fig . In simple images with or without colour (gray scale), Tesseract provides results with 100% accuracy. But in the case of some complex images Tesseract provides better accuracy results if the images are in the gray scale mode as compared to colour images. To prove this hypothesis, OCR of same colour images and gray scale images is performed and in both cases different result are achieved.

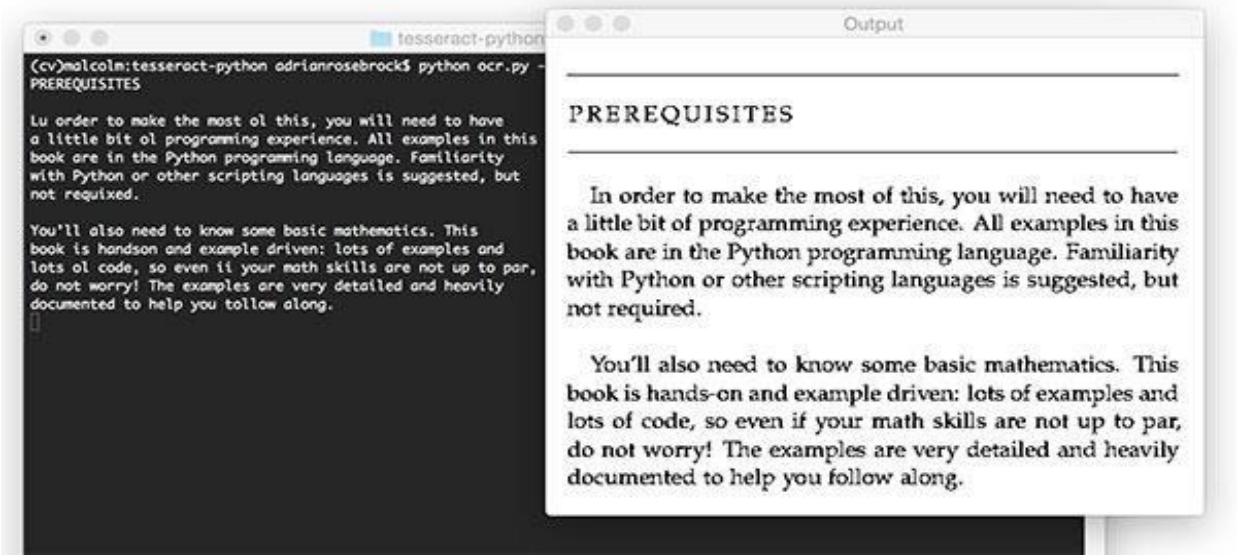
A complex colour image shown in the figure is performed by Tesseract and after OCR processing of image; the text extracted in the image is not as accurate as it is expected. The extracted text is not exactly same as it is visible in the image . That means Tesseract is not able to extract the text with 100% accuracy. So to get more accurate result the same image is converted to the gray scale image and OCR is performed on this image.

IV. A GRAYSCALE IMAGE OF TESSERACT

As it is visible in the fig , the color image is converted in the gray scale image. The image is converted to the gray scale by using the algorithm that is discussed in the following section. After processing the above image output is shown below. It is clearly visible that Tesseract has successfully retrieved the text , which is expected as output. This means that Tesseract has extracted the text with 100% accuracy

. The results discussed in section reveal that Tesseract provides more accuracy in processing gray scale images. The following section contains algorithm to convert any colour image to the gray scale image.

V. ALGORITHM TO CONVERT THE COLOURED IMAGE TO A GRAY SCALE IMAGE:



A digital image with M width (row) and N height (column) is represented as discrete function f(x, y) as: f(x,y) = (xi,yi), where i = 0, i < N,j = 0,j < M (1)

Here the pair (xi, yj) is known as pixel. The pair (0,0) is the first pixel and pair(M-1,N-1) is the last pixel in the image. Every pixel has its own RGB colour value. If the pixel has the same RGB value then it falls into gray colour family (black to white). So based on this observation the algorithm to convert colour image to gray scale is developed, which is shown under:

u(x,y) = Σ((x,y)r, (x,y)g, (x,y)b) 3 ∀(x,y) where −1 < r,g,b < 256 (2)

Here r, g and b are red colour, green colour, blue colour values of pixel (x, y) respectively. Range of r, g and b are mentioned in is the mean value of these pixels that is always less □(2) and is in between 0 to 255, which is□than 256. So value of assigned to red, green and blue pixels of pixel (x, y). This process is depicted in (3), (4) and (5).

(x ,y)r = u(x,y) ∀(x,y) where x N,YM (3)

(x ,y) = u(x,y)∀(x,y) where x N,YM (4)

(x, y)b = u(x,y)∀(x,y) where x N,YM (5)

So after applying above algorithm on colour image it becomes gray scale image.

VI.EXPERIMENTAL RESULTS OF TESSERACT OCR:

We have captured 20 different kinds of number plates' images from various types of vehicles and performed OCR of these number plates to extract vehicle number.This shows that Tesseract provides 71% accuracy with the colour images and 70% accuracy with gray scale images.So it indicates that Tesseract provides better accuracy in gray scale images as compared to colour images.

This experiment was carried out on computer with Intel Pentium (R) 4 2.4GHZ CPU and 1 GB RAM. The images of number plates are captured by 5-megapixel camera. We can observe that if colour images are converted to gray scale and given as input to Tesseract then accuracy of text extraction is increased. In some colour images where text extraction accuracy result is 100 % or near to 100 %, and if it is converted to gray scale then it produces same amount of extraction accuracy. In some colour images Tesseract is not able to provide more than 70 % of accuracy; we have converted these images into gray scale images by using the algorithm discussed in the previous section and then given these images as input to Tesseract. So after doing this process there is an increase in the average accuracy to extract the characters from the vehicle number plate. This accuracy of individual image processing varies from 65% to 100.

Also it is observed that the processing time of extracting characters from gray scale images is decreased. It is reduced by 10% to 50%. So we can say that Tesseract works fast and provides better text extraction accuracy in processing gray scale number plate images.

VII.CONCLUSION

Nowadays, a lot of documents are produced in paper form but it is obvious, that automatic data recognition systems are very popular. Though researchers have suggested various sophisticated ideas and techniques, practical OCR systems suffer from a lack of various characteristics. It is because of the claims made by the researchers are not adequately justified by exposure of the systems into real working environments and the lack of practical feasibility of such advanced techniques with the available hardware from an economical viewpoint.

From these constraints and the lack of performances it can be concluded that the ability to read text by machines with the same fluency as the human remains an unachieved goal, though a great amount of effort has already been expended on the subject.

However, the frontiers of character recognition have now moved to the recognition of cursive script that is the recognition of characters which may be connected or written in calligraphy.

The future use of ocr tesseract can be:

- Page layout analysis
- More languages
- Improve accuracy
- Add a UI
- Support for connected scripts (like Arabian)

VIII. REFERENCES:

[1] Karaoglu, Sezer and van Gemert, Jan C and Gevers, Theo, Object Reading: Text Recognition for Object Recognition, in ECCV Workshops 2012

[2] Karaoglu, Sezer and van Gemert, Jan C and Gevers, Theo, Con-text: text detection using background connectivity for fine-grained object classification, in ACM-MM 2013

[3] Everingham, Mark and Eslami, SM Ali and Van Gool, Luc and Williams, Christopher KI and Winn, John and Zisserman, Andrew, The pascal visual object classes challenge: A retrospective. in IJCV 2014

[4] Mishra, Anand and Alahari, Karteek and Jawahar, CV, Image Retrieval using Textual Cues, in ICCV 2013

[5] Gavves, E. and Fernando, B. and Snoek, C. GM and WM Smeulders, A. and Tuytelaars, T., Local alignments for fine-grained categorization, in IJCV 2015

[6] Zhang, Ning and Donahue, Jeff and Girshick, Ross and Darrell, Trevor, Part-based R-CNNs for fine-grained category detection, in ECCV 2014

[7] Lu, Tong and Palaiahnakote, Shivakumara and Tan, ChewLim and Liu, Wenyin, Text Detection in Multimodal Video Analysis, in Video Text Detection (pp. 221-246), Springer London 2014

[8] Lu, Tong and Palaiahnakote, Shivakumara and Tan, Chew Lim and Liu, Wenyin, Video Text Detection, in Advances in Computer Vision and Pattern Recognition, Springer London 2014

[9] Gómez, Lluís, and Karatzas, Dimosthenis, Scene text recognition: No country for old men?, in ACCV Workshops 2014.

[10] Augereau, Olivier and Journet, Nicholas and Vialard, Anne and Domenger, Jean-Philippe, Improving classification of an industrial document image database by combining visual and textual features, in Document Analysis Systems (DAS) 2014

[11] Rusiñol, Marçal and Frinken, Volkmar and Karatzas, Dimosthenis and Bagdanov, Andrew D and Lladós, Josep, Multimodal page classification in administrative document image streams, in IJDAR 2014

[12] Szegedy, C. and Liu, W. and Jia, Y. and Sermanet, P. and Reed, S. and Anguelov, D. and Erhan, D. and Vanhoucke,V. and Rabinovich, A., Going Deeper with Convolutions, in CVPR 2015

[13] Uijlings, Jasper RR and van de Sande, Koen EA and Gevers, Theo and WM Smeulders, Arnold,Selective search for object recognition, in IJCV 2013

[14] Jiang, Ming and Huang, Zhao, Shengsheng and Duan, Juanyong and Qi, SALICON: Saliency in context." in CVPR 2015.

[15] Lin, Tsung-Yi and Maire, Michael and Belongie, Serge and Hays, James and Perona, Pietro and Ramanan, Deva and Dollár, Piotr and Zitnick, C. Lawrence, Microsoft coco: Common objects in context, in ECCV 2014.

[16] Veit, Andreas and Matera, Tomas and Neumann, Lukas and Matas, Jiri and Belongie, Serge, COCO-Text: Dataset and Benchmark for Text Detection and Recognition in Natural Images.