e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-11, Issue-2 (April 2021)
https://doi.org/10.31033/ijemr.11.2.11

OCR Text Extraction

Alan Jiju[1], Shaun Tuscano[2] and Chetana Badgujar[3]
[1]Student, Department of IT, Fr. Conceicao Rodrigues Institute of Technology, Vashi, Navi Mumbai, INDIA
[2]Student, Department of IT, Fr. Conceicao Rodrigues Institute of Technology, Vashi, Navi Mumbai, INDIA
[3]Assistant Professor, Department of IT, Fr. Conceicao Rodrigues Institute of Technology, Vashi, Navi Mumbai, INDIA

[1]Corresponding Author: alanvilayil007@gmail.com

ABSTRACT

This research tries to find out a methodology through which any data from the daily-use printed bills and invoices can be extracted. The data from these bills or invoices can be used extensively later on – such as machine learning or statistical analysis. This research focuses on extraction of final bill-amount, itinerary, date and similar data from bills and invoices as they encapsulate an ample amount of information about the users purchases, likes or dislikes etc. Optical Character Recognition (OCR) technology is a system that provides a full alphanumeric recognition of printed or handwritten characters from images. Initially, OpenCV has been used to detect the bill or invoice from the image and filter out the unnecessary noise from the image. Then intermediate image is passed for further processing using Tesseract OCR engine, which is an optical character recognition engine. Tesseract intends to apply Text Segmentation in order to extract written text in various fonts and languages. Our methodology proves to be highly accurate while tested on a variety of input images of bills and invoices.

Keywords— OpenCV; Optical Character Reader (OCR); Tesseract, Document Detection

I. INTRODUCTION

This research was considered and taken into mainstream because the idea of processing and extracting text from an image is enticing enough. This project also clears the hidden opportunities of further research with ample amount of data, which is the next big thing in this fast-moving technological era. The research aims to extract data from the bills present in the image, here focusing on bills and invoices only, which would later help us to find what people prefer these days in the market and how their choices and likings vary over geography, time etc. The process starts by taking an image and trying various techniques to extract the text from it, and we end up utilizing Open CV and Tesseract OCR which are very powerful tools in their respective fields. To verify our results of our research we implemented an Android app to see around the results and accuracy of methods and way of implementation of these tools. The Android App aimed to take images from the users, images as Bills or Invoices and on the basis of the total amount present in the bill, we reward user with some cash back points to them, which they can utilize in buying some discount vouchers.

II. BACKGROUND STUDY

Before starting this research, many available technologies and research were studied and taken into consideration which gave us fair knowledge and information that made clear about the methodology which has been done to get the best possible result. OCR (Optical Character Recognition) is an engine that helps to recognize and retrieve a full alphanumeric recognition of printed or handwritten character units from text-based images or from scanned documents. Tesseract is an open source and a popular optical character recognition engine till date. The basic processing by the Tesseract is a step-by-step pipeline methodology in which a connected component analysis is performed. Outlines are analyzed, filtered and then gathered together as Blobs. Blobs are processed and organized into text lines. Then the process continues by breaking down text-lines words. First pass of recognition process attempts to recognize each word in turn. Words satisfying the constraints are further passed to adaptive trainer. Adaptive trainer learns lesson from the first pass and employs it in second pass, here reattempts to recognize the words are made that were not satisfying all the constraints of the first pass. Fuzzy spaces resolved and text checked for small caps. Digital texts are then outputted. Text image segmentation is typically used to locate objects and boundaries in images. Text Segmentation as the word suggests is a process of segmenting scanned text into useful elements such as words, sentences or topics. Text segmentation was a requirement for information retrieval. Tesseract OCR engine have poor quality output if the input images are too noisy or may contain unnecessary objects. So, images have to be pre-processed before passing it to tesseract OCR, so to have a better quality and more accurate output. Hence, for aim to remove the unnecessary noise and objects from the background, OpenCV tool has been taken in to consideration for this purpose. OpenCV (Open Source Computer Vision Library) is a library of programming functions mainly aimed at real time computer vision. First the edge detection algorithm is applied to the image and then contours tracking are performed. All the contour present in the image, four-point contour is searched; the largest four-point contour is our required object from the image that is Bill or Invoice from the image. Then the detected object is cropped out from the original image and some filters are applied to it.

Following tools were used during the current research work:

a. OpenCV

OpenCV is a Computer Vision library with APIs that let you setup a pipeline for your Computer Vision project. I/O. Loading data from image files, videos, capturing devices. OpenCV contains a long list of existing algorithms so you don't have to implement them yourself.

b. Android Studio

Android Studio is Android's official IDE. It is purpose built for Android to accelerate your development and help you build the highest-quality apps for every Android device. It offers tools custom-tailored for Android developers, including rich code editing, debugging, testing, and profiling tools.

c. Tesseract OCR

Tesseract is finding templates in pixels, letters, words and sentences. It uses two-step approach that calls adaptive recognition. It requires one data stage for character recognition[1], then the second stage to fulfil any letters, it wasn't insured in, by letters that can match the word or sentence context.

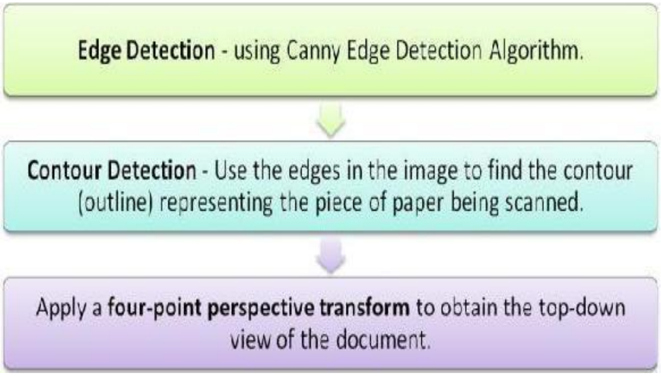

Edge Detection - using Canny Edge Detection Algorithm.
Contour Detection - Use the edges in the image to find the contour (outline) representing the piece of paper being scanned.
Apply a four-point perspective transform to obtain the top-down view of the document.

Figure 1: The methodology used in our research work

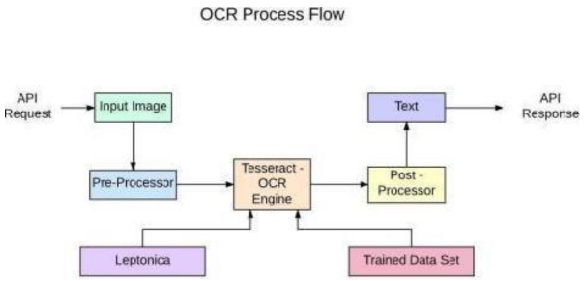


Figure 2: OCR process flow

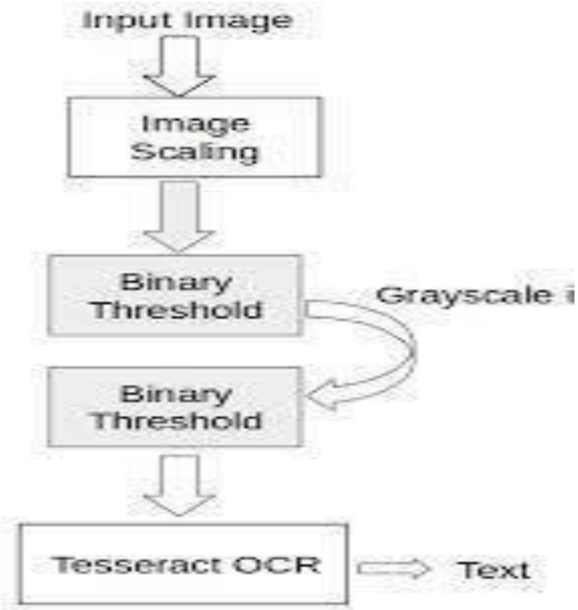


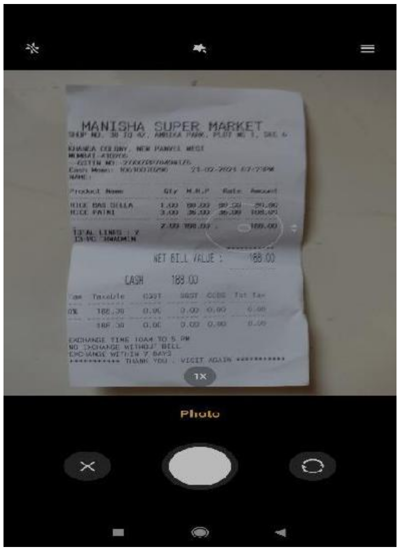
Figure 3: Block diagram

d. Google Firebase

Google Firebase is a Google-backed application development software that enables developers to develop iOS, Android and Web apps. Firebase provides tools for tracking analytics, reporting and fixing app crashes, creating marketing and product experiment.

III. BILL DETECTION METHODOLOGY

We have used OpenCV technology in our project to get the document i.e. bill from the image of bill from the user that may have noise in the background and provide a transformed image of the bill or invoice only so that accuracy of the text extraction could be increased. The printed bill or invoice is a rectangular piece of paper; hence we need to find the largest piece of paper i.e. the largest rectangular object in the image that will be our bill in the image Consider an example image as shown in Fig. 1. The Canny edge detector is an optimal edge detection algorithm developed by John F. Canny in 1986[3]. As the name suggests it is used to detect edges in an image. It uses a multistage algorithm to detect a wide range of edges in images. For its accuracy and ease to basic algorithm to perform any line, edge or contour detection. Also known to many as the optimal detector. Main aim of the algorithm is to satisfy three main criteria: Low error rate: Meaning a good detection of only existent edges. Good localization: The distance between edge pixels detected and real edge pixels have to be minimized. Minimal response: Only one detector response per edge. The output of the example image after canny edge detection. Contour detection is meaning finding the outline of the image. To perform contours detection OpenCV provide a pre-build function named as Find Contours () whose job is to help one detect contours in the provided image.



MANISHA SUPER MARKET

NET BILL VALUE : 188.00

CASH 188.00

EXCHANGE TIME 10AM TO 5 PM
NO EXCHANGE WITHOUT BILL
EXCHANGE WITHIN 7 DAYS
*********** THANK YOU . VISIT AGAIN ***********

Figure 4: Image of invoice

I order to get good quality output some treatment should be applied to the image. As since our purpose is to detect a bill from the image, we assume the bill to be a document[4] which can also be a rectangular object, so one can easily assume that the largest contour in the image with exactly four points is our piece of paper or bill to be scanned.

- Firstly, we convert input image to gray, since we need to work on grayscale picture.
- A simple threshold is applied to roughly get contours.
- The OPEN and CLOSE trick is used to get more smooth contours
- A list of contours can retrieve by using FindCountour () function.
- Sort the retrieved contour list in non-increasing order.
- Loop over the sorted contour list and check which contour has 4 points.
- Once found the 4-point contour, finally drawn on the color as if we were working on it

The last step for a bill's image detection is to crop the image along the four borders of the bill in order to get a top-down view of the image. In addition to step 3 we can use adaptive thresh holding to give our image in a clean black and white image, giving document a paper effect.

IV. TEXT RECOGNITION

Text Extraction and Optical Character Recognition (OCR) is done via help of an open source optical character recognition engine Tesseract [2] and OCR [5]. The processed image is passed to the Tesseract OCR engine and the engine helps to recognize text or extract the text from the image. The engine The steps of the process of engine is demonstrated in the flow diagram as in Figure 1.The initial step Is to analyze the connected elements for that the outline of the elements are saved. Though it was a costly step but it resulted in huge gain, the analysis of the outlines of the encapsulated elements, the text was identified in a much simpler way as dark element over light one., and segmentation process is applied here. Text Segmentation [6] is the process of partitioning a digital image into multiple segments (sets of pixels). Segmentation is used for text- based images, aimed in retrieval of specific information from the entire image.

Levels of segmentation process are as:

Line Segmentation – The initial step for image segmentation is line segmentation, in which the horizontal scan of the image is done by the engine.

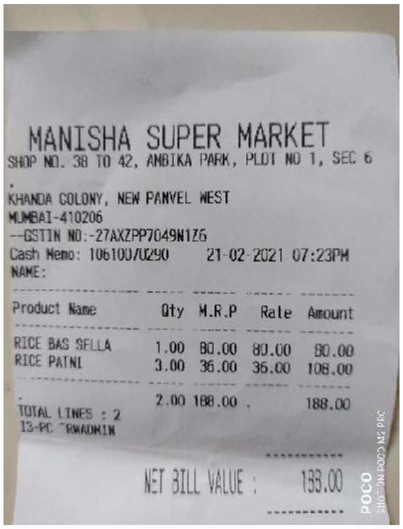
A. Word Segmentation– The second step after the horizontal scan is the vertical scan, which is called as word segmentation.

B. Character Segmentation- Character segmentation is the final level for text-based image segmentation. It includes vertical scanning of the image.

Segmentation here is basically a process to decompose a piece of text into a set of words and further into characters and further to individual segments. The need to do so is to recognize the final characters and make a meaningful analysis. Every word that is found to be sensible is then passed to the adaptive classifier as training data. The basic functionality of the adaptive classifier is to recognize the lower page more precisely so as to get a much better result. It may be possible that the adaptive classifier finds out something unique which was not earlier noticed on the upper page, so a second check is given to the complete page for the similar segmentation. Once this is done, a final check is given wherein vague spaces and small caps are rechecked

V. IMPLEMENTATION

Both the bill detection and text extraction methodology where implemented and tested on Python 3.5 platform, using the python bindings of the respective tools, which were discussed above. Latest versions of both the tools where used as being OpenCV 3.3.0.10 and the latest stable version of Tesseract-OCR 3.05.01, which was released on July 1st, 2017. We used an open source toll Flask, a micro framework for Python based on Werkzeug, Jinja 2 used to develop Web App in Python. We made an FlaskApp for our app, that can perform the above defined task and return the result True/False whenever the android app calls the FlaskApp. We hosted the FlaskApp on to a live webserver so that it can be used in Android App to perform a particular task. For testing the research, we developed an android app that asks its user to upload images of the bills or invoice using the mobile phone camera or user can upload it from the gallery and uploads it to a webserver where we hosted our FlaskApp which perform the above discussed task. Along with image the user also sends the total amount he paid for the invoice, we extracted the total amount from the image and matched it with the user entered data, if the data matched then it returned True as response and vice-versa



MANISHA SUPER MARKET
SHOP NO. 38 TO 42, ANBIKA PARK, PLOT NO 1, SEC 6
,
KHANDA COLONY, NEW PANVEL WEST
MUMBAI-410206
--GSTIN NO:-27AXZPP7049N1Z6
Cash Memo: 106100/0290   21-02-2021 07:23PM
NAME:

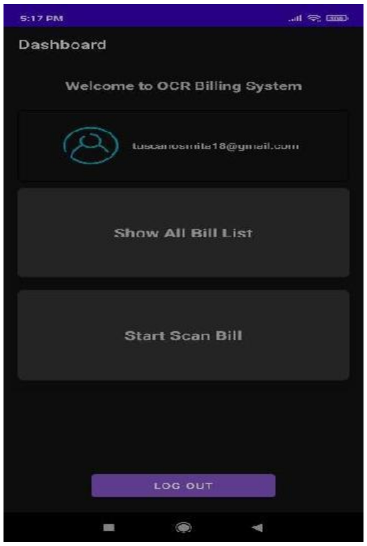
Product Name	Qty	M.R.P	Rate	Amount
RICE BAS SELLA	1.00	80.00	80.00	80.00
RICE PATNI	3.00	36.00	36.00	108.00
.	2.00	188.00	.	188.00

TOTAL LINES : 2
13-PC "RWADMIN

NET BILL VALUE :   188.00

POCO
SHOT ON POCO M2 PRO

Figure 5: cropped example image of the invoice



5:17 PM

Dashboard

Welcome to OCR Billing System

tuscanosmile18@gmail.com

Figure 6:.Main section

VI. DISCUSSION AND CONCLUSION

This research led the way to some amazing results but also highlighted the paths and scope of improvements. The technique that is applied has some limitation to it. The above research was able to extract the text front printed bills but it failed for handwritten bills and invoices, as tesseract uses text segmentation technique that recognizes first character in the line and then according to the position of the first character tries to read the whole line, hence in the case of handwritten text [6] the flow is not in a hence the characters aren't recognized well. Other drawback of the research is that if the image contains a document that is not a bill or any rectangular piece of paper or object, the research will detect that object and think it as the required bill or invoice whose text is needed to be extracted. Though further text processing can be applied and identify if the extracted information is of a bill or invoice. Many Computer vision methodology has been used for the detection of Bills or Invoices which is further processed to extract all the text from the bills that can be later used for Data Analysis. With the help of OpenCV, Tesseract-ocr and using image processing theory, we have achieved our research goal. Through the practice test, it has been came in notice that the methodology introduced in the paper has helped to improve the accuracy of the test output and the work efficiency.

REFERENCES

[1]https://opencv.org/.

[2] http://code.google.com/p/tesseract-ocr.

[3]Canny, J F. (1983). Finding edges and lines in images. *MIT Technical Report AI-TR-720*.

[4] Berg, A. C., Berg, T. L., & Malik, J. (2005). Shape matching and object recognition using low distortion correspondence. In: *Proc IEEE Conf on Computer Vision and Pattern Recognition, San Diego CA*, pp. 20-25.

[5] Teofilo E., de Campos, & Bodla Rakesh Babu. (2002). Optical Character Recognition (OCR) Technology. In: *IIIT Hyderabad Conference*.

[6] Rodolfo P. dos Santos, Gabriela S. Clemente, Tsang Ing Ren, & George D.C. Calvalcanti. (2009). Text line segmentation based on morphology and histogram projection. In: *10th International Conference on Document Analysis and Recognition*.

[7] L. Likforman-Sulem, A. Zahour, & B. Taconet. (2007). *Text line segmentation of historical documents: A survey*. Available at: https://link.springer.com/article/10.1007/s10032-006-0023-z.

[8] https://research.google.com/pubs/archive/33418.pdf.