International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 05 | May - 2021 ISSN: 2582-3930

Image to Text Conversion using Open CV Technology

Aditya Adak[1], Shubham Mahajan[2], Bhushan Toke[3] ,Dr.Nita Patil[4]

[1-3]Student,Computer Department , Datta Meghe College Of Engineering , Navi Mumbai ,Maharashtra ,India
[4]Faculty,Computer Department , Datta Meghe College Of Engineering , Navi Mumbai ,Maharashtra ,India

---

Abstract -*In world the smartphones are everywhere and everyone uses smartphones for their day to day life. The smartphones have been developed for various purposes like capturing images, record videos, surf the internet and etc. With advancement of technology, it is possible to apply some techniques to perform text detection and translation. In this study, we have tried to integrate the Tesseract OCR engine and the Google Vision library and develop an application on android platform that allows user to capture the images using camera and extract the text from it. This application recognizes the text that is captured by a mobile phone camera and displays back the recognized text on to the screen. To develop this application we have used the Optical Character Recognition, OCR engine, and we develop our own open source Android application. This application has come up with the solutions for the problems of retyping any hard copy of any documents, scanning the documents in the scanner then use the computer oriented software to recognize the text in the file, or guessing the text and typing it. It aims to develop a piece of software that runs on smartphones and can be used to identify and recognize text of any text based image using the phone's camera. In case of a finding, In order to achieve fast processing times, the project has to deal with the low computing power of smartphones. For this purpose, different approaches to object detection, such as concepts from the machine learning domain, and their implementation on mobile platforms will be analysed.*

Key Words:Tesseract , Optical Character Recognition , Scanner .

1.INTRODUCTION

Image to text recognition project is an active research area. Which attempts to develop a computer application with the ability to automatically read the text from images. It uses OCR algorithm to convert the physical image in the surrounding to digitally editable format.
OCR comes in the OpenCV application contributes immensely the advancement of an automation process and can improve the interface between man and machine in numerous applications.

1.1 Aims and Objectives

The objective of OCR software is to recognize the text and then convert it to editable form. Thus, developing computer algorithms to identify the character in the text is the principal task of OCR. A document is first scanned by an optical scanner, which produces an image form of it that is not editable. Then Optical character recognition involves.

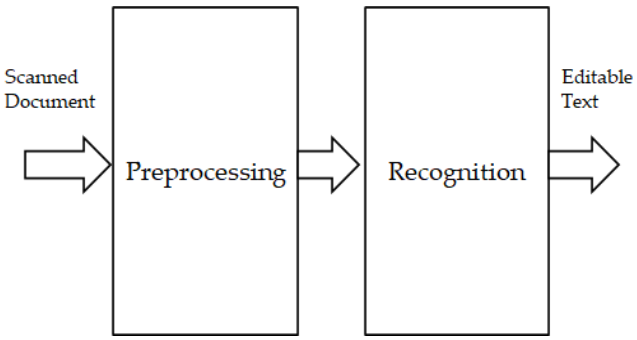


Fig. 1.1.1OCR Engine

And it coverts that text image into the electronic data which is in editable format.

1.2 Applications

Text recognition technology is also applied throughout the complete spectrum of industries, revolutionizing the document management method. This technology modify scan documents to become quite simply image files, turning into totally searchable documents with text content that's recognized by computers. With the assistance of this technology, folks now not have to be compelled to manually retype necessary documents once coming into them into electronic databases. Instead, Text recognition system extracts relevant data and enters it mechanically. The result's correct, economical information science in less time.

1.1 Text in images

A variety of approaches to text information extraction (TIE) from images and video have been proposed for specific applications including page segmentation, address block location, license plate location, and content based image/video indexing. In spite of such extensive studies, it is still not easy to design a general-purpose TIE system. This is because there are so many possible sources of variation when extracting text from a shaded or textured background, from low-contrast or complex images, or from images having variations in font size, style, color, orientation, and alignment. These variations make the problem of automatic TIE extremely difficult.

Fig. 1-3 show some examples of text in images. Page layout analysis usually deals with document images (Fig.1). Readers may refer to papers on document segmentation/analysis for more examples of document images. Although images acquired by scanning book covers, CD covers, or other multi-colored documents have similar characteristics as the document images (Fig.2), which is artificially overlaid on the image, or scene text (Fig.3), which exists naturally in the image. Some researchers like to use the term 'graphics text' for scene text, and 'superimposed text' or 'artificial text' for caption text. It is well known that scene text is more difficult to detect and very little work has been done in this area. In contrast to caption text, scene text can have any orientation and may be distorted by the perspective projection. Moreover, it is often affected by variations in scene and camera parameters such as illumination, focus, motion, etc.

Fig. 1.3.1 Grayscale document images: (a) single-column text from a book, (b) two-column page, and (c) an electrical drawing

Fig. 1.3.2 Multi-color document images: each text line may or may not be of the same color.

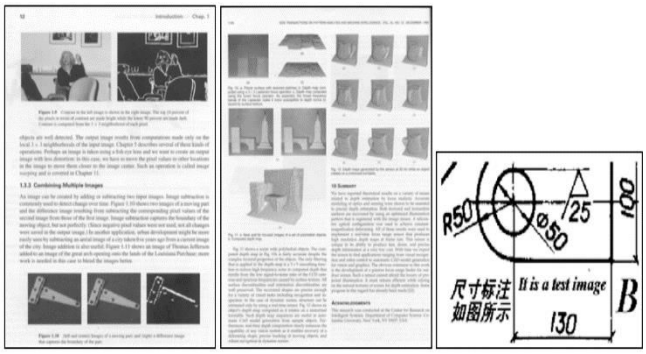


Fig. 1.3.3 Scene text images: Images with variations in skew, perspective, blur, illumination, and alignment.

Before we attempt to classify the various techniques used in TIE, it is important to define the commonly used terms and summarize the characteristics of text that can be used for TIE algorithms. Table 1 shows a list of properties that have been utilized in recently published algorithms. Text in images can exhibit many variations with respect to the following properties:

1.1 Geometry:

- Size: Although the text size can vary a lot, assumptions can be made depending on the application domain.
- Alignment: The characters in the caption text appear in clusters and usually lie horizontally, although sometimes they can appear as non-planar texts as a result of special effects. This does not apply to scene text, which can have various perspective distortions. Scene text can be aligned in any direction and can have geometric distortions (Fig. 4).
- Inter-character distance: characters in a text line have a uniform distance between them.

2.1 Color:

The characters in a text line tend to have the same or similar colors. This property makes it possible to use a connected component-based approach for text detection. Most of the research reported till date has concentrated on finding 'text strings of a single color (monochrome)'. However, video images and other complex color documents can contain 'text strings with more than two colors (polychrome)' for effective visualization, i.e., different colors within one word.

3.1 Motion:

The same characters usually exist in consecutive frames in a video with or without movement. All these properties may not be important to every application.

4.1 Edge:

Most caption and scene text are designed to be easily read, thereby resulting in strong edges at the boundaries of text and background.

5.1 Compression:

Many digital images are recorded, transferred, and processed in a compressed format. Thus, a faster TIE system can be achieved if one can extract text without decompression.

2. Literature of Review

2.1 First generation OCR systems

The first commercialized OCR of this generation was developed by IBM, which was designed to read a special IBM font407. The recognition method was template matching, which compares the character image with a library of prototype images for each character of each font.

2.2 Second generation OCR systems

Next generation machines were able to recognize regular machine-printed and hand printed characters. The character set was limited to numerals and a few letters and symbols.

2.3 Third generation OCR systems

For the third generation of OCR systems, the challenges were documents of poor quality and large printed and hand-written character sets. Low cost and high performance were also important objectives.

2.4 Fourth generation OCR systems

The fourth generation can be characterized by the OCR of complex documents intermixing with text, graphics, tables and mathematical symbols, unconstrained handwritten characters, color documents, low-quality noisy documents, etc. Among the commercial products, postal address readers, and reading aids for the blind are available in the market.

2.5 Latest OCR systems

Todays, OCR systems gives more accuracy as compared to old ones but todays OCR system applications failed to recognize tables and tabular data. And the applications take more time to convert the images into editable format, Also the converted video into images are hard to convert into text because many times images are framed with motion and many images are blurred so it is hard to convert it into the editable format.

SummarisedFindings :-

Sr. No.	Paper Title	Authors	Year	Methods Used	Advantages
1.	Text recognition	K. Karthick R. Francis	2019	Point processing techniques, Mask processing techniques, Thresholding using Otsu's Method	Detect text from same text colour background.
2.	Text recognition using image processiong	Suparna Karmarkar, Tridev Chakraborty	2017	Text detection on Edge based Method	Recognise text from video also
3.	Extracting text from document	K. Natei, S. Sasikumar	2018	Connected component labelling, Edge based method	Extract text with hight accuracy.
4.	Character recognition	Jaswinder Kaur, Rupinder Kaur	2018	Mistreatment technique, Otsu's algorithm, Skeletonization for image processing	Multilingual character detection.
5.	Text extraction from image using android	Amar Waghade, Suraj Shelke	2018	Connected component analysis, Google API's, Text feature extraction (filtration, fusion, mapping)	It uses google API's for translate the text into speech.

Table 2.6.1. Literature survey table

1) K Karthick, R. Francis : - Text Recognition (2019) used techniques such as Point Processing , Mask Processing ,Thresholding using Otsus Method . In Point processing there are methods like Adaptive Thresholding, Contrast Stretching, Histogram Equalization etc. The advantage of this methods were it detect text from same text colour background.

2) SuparnaKarmarkar, TridevChakraborty : - Text recognition using image processing (2017) used edge based method in which it detects and links edge pixels to form contours. A large group of methods – in gray level, color, texture, etc. Result from edge detection cannot be used directly. It recognize text from video also.

3) K Natei, S. Sasikumar :- Extracting text from document (2018) used Connected component labelling which extract text with hight accuracy . It is an algorithmic application of graph theory, where subsets of connected components are uniquely labeled based on a given heuristic.

4) Jaswinder Kaur, Rupinder Kaur :- Character Recognisition (2018) used Mistreatment technique and Skeletonization for image processing. It detected Multilangual character.

3. Text Information Extraction (TIE)

The problem of Text Information Extraction needs to be defined more precisely before proceeding further. A TIE system receives an input in the form of a still image or a sequence of images. The images can be in gray scale or color, compressed or un-compressed, and the text in the images may or may not move. The TIE problem can be divided into the following subproblems: (i) detection, (ii) localization, (iii) tracking, (iv) extraction and enhancement, and (v) recognition (OCR). to a binary image and enhanced before it is fed into an OCR engine.

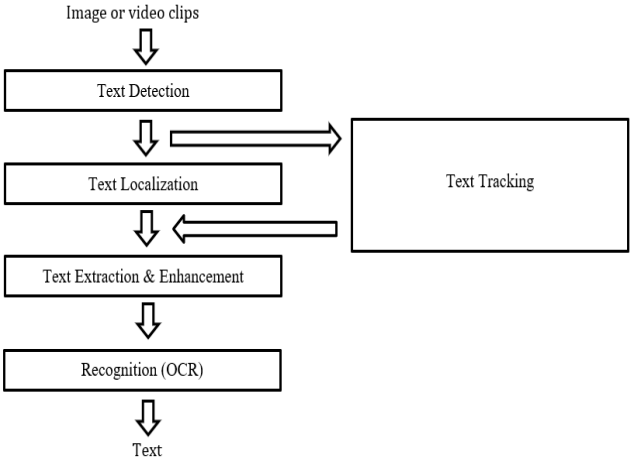


Fig. 3.1 Properties of text in image

Text detection, localization, and extraction are often used interchangeably in the literature. Text detection refers to the determination of the presence of text in a given frame (normally text detection is used for a sequence of images). Text localization is the process of determining the location of text in the image and generating bounding boxes around the text. Text tracking is performed to reduce the processing time for text localization and to maintain the integrity of position across adjacent frames. Although the precise location of text in an image can be indicated by bounding boxes, the text still needs to be segmented from the background to facilitate its recognition. This means that the extracted text image has to be converted.

Property		Variants or sub-classes
Geo-metry	Size	Regularity in size of text
	Alignment	Horizontal/vertical
		Straight line with skew (implies vertical direction)
		Curves
		3D perspective distortion
	Inter-character distance	Aggregation of characters with uniform distance
Color		Gray
		Color (monochrome, polychrome)
Motion		Static
		Linear movement
		2D rigid constrained movement
		3D rigid constrained movement
		Free movement
Edge		Strong edges (contrast) at text boundaries
Compression		Un-compressed image
		JPEG, MPEG-compressed image

Fig. 3.2 Architecture of TIE System

Text extraction is the stage where the text components are segmented from the background. Enhancement of the extracted text components is required because the text region usually has low-resolutionand is prone to noise. Thereafter, the extracted text images can be transformed into plain text using OCR technology.

4.Requirement Analysis

System Requirements :-

The minimum system requirements for running OCR:

Operating Systems: Microsoft Windows XP, Windows Vista, Windows 7, Windows 8, Windows 8.1, Windows 10

Hardware requirements:-

Intel or AMD Processors (more the cores and speed of processor faster the character recognition), minimum 100 MB of free hard-disk space, minimum 512 MB of RAM. If your ComputerHardware System is more upgraded the OCR Software run faster.

4.1 Software Requirement

Operating Systems: Microsoft Windows XP, Windows Vista, Windows 7, Windows 8, Windows 8.1, Windows 10.

4.1.1 Software Platforms

CPU
MATLAB, OpenCV Java – JImage.

GPU
OpenGL
Cg scripting, HLSL (High Level Shader Language)

CPU + GPU's
OpenCL

Others platforms
JImage, OpenCV

4.2 Hardware Requirement

Intel or AMD Processors (more the cores and speed of processor faster the character recognition), minimum 100 MB of free hard- disk space, minimum 512 MB of RAM. If your Computer Hardware System is more upgraded the OCR Software run faster.

5.Project Design

Project design is an early phase of the project where a project's key features, structure, criteria for success, and major deliverables are all planned out. The point is to develop one or more designs which can be used to achieve the desired project goals.

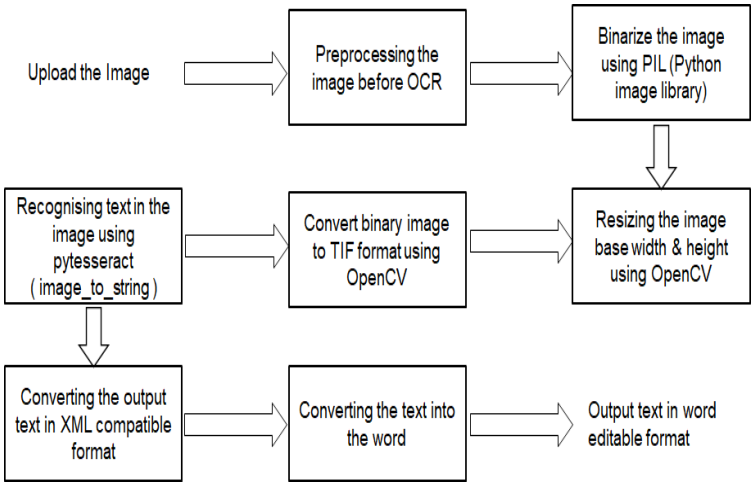


Fig. 5.1 Workflow Diagram

5.1. Design Implementation :-

For the Text Scanner, first we need to convert our image to binary format for preprocessing, for that we use OpenCV and we convert to binary image and then we set width and height of the image.

Now we need to convert binary image to tif format because pytesseract only accepts tif image. Using open cv we convert binary preprocessed image to tif image and this image is given to pytesseract while calling the function convert_image_to string which returns list of strings which are identified in the image.

After getting this list of strings we join all the strings. after getting the string to make it XML compatible i.e. to make it cleaned string we do string cleaning. After the strings get cleaned it returns the list of words and then we join the words and display it to the output screen

6. Methodology

In this section we briefly describe the overall architecture of text recognition system as shown in figure 5. A text recognition system receives an input in the form of image which contains some text information. The output of this system is in electronic format i.e. text information in image are stored in computer readable form. The text recognition system can be divided in following modules:

A. pre-processing
B. Tesseract Engine
C. post-processing

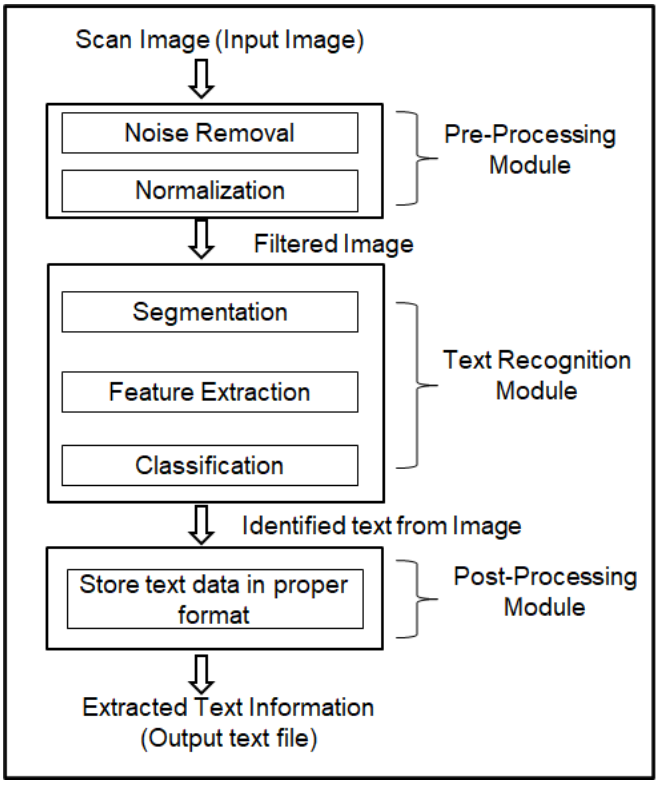


Fig. 6.1 Basic Architecture of text recognition system

A. Pre-processing Module:-

The paper document is generally scanned by the optical scanner and is converted in to the form of a picture. A picture is the combinations of picture elements which are also known as pixels. At this stage we have the data in the form of image and this image can be further analyzed so that's the important information can be retrieved. So to improve quality of the input image, few operation are performed for enhancement of image such as noise removal, normalization, binarization etc.

1) Noise Removal:
Noise removal is one of the most important process. Due to this quality of the image will increase and it will effect recognition process for better text recognition in images. And it results in generation of more accurate output at the end of text recognition processing. There are many methods for image noise removal such as mean filter, min-max filter, Gaussian filter etc.

2) Normalization:
Normalization is one of the important pre-processing operation for text recognition. The normalization is applied to obtain characters of uniform size, slant and rotation.

3) Binarization:
Binarization is one of the important pre-processing operation for text recognition. A printed document is first scanned and is converted into a gray scale image. Binarization is a technique by which the gray scale images are converted to binary images. This separation of text from background that is required for some operations such as segmentation. Figure 2 shows a colour image (a) gray image (b), and binary image (c) of a text image.

B. Tesseract Engine

Tesseract is an optical character recognition engine with open source code, this is the most popular & qualitative OCR library. OCR uses artificial intelligence for text search and its recognition on images.
Tesseract is finding templates in pixels, letters, words and sentences. It is use two step approach that calls adaptive recognition. It requires one data stage for character recognition, then the second stage to fulfil any letters, it wasn't insured in, by letters that can match the word or sentence context.

1) Segmentation:
In pytesseract engine, the segmentation is the most important process. Segmentation is done to make the separation between the individual characters of an image.

2) Feature Extraction:
Feature extraction is the process to retrieve the most important data from the raw data. The most important data means that's on the basis of that's the characters can be represented accurately.

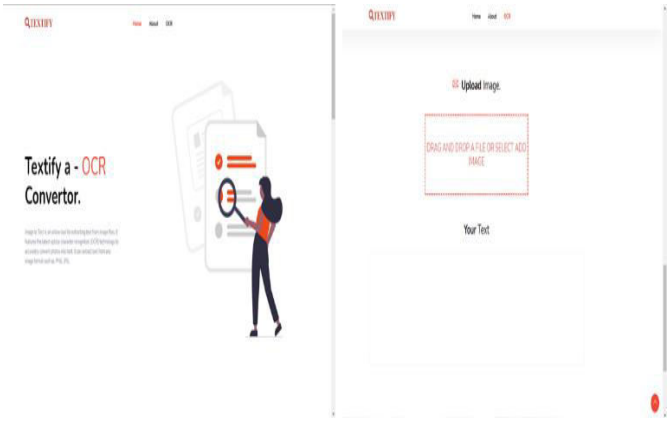
3) Classification:
The classification is the process of identifying each character and assigning to it the correct character class, so that texts in images are converted in to computer understandable form. This process used extracted feature of text image for classification i.e. input to this stage is output of the feature extraction process. Classifiers compare the input feature with stored pattern and find out best matching class for input.

C. Post-processing Module:

The output of Tesseract Engine is in the form text data which is converted in XML format. So there need to store it in to some proper format for farther use such as editing or searching in that data.

7. Web Functionalaties

Frontend consists of Web interface in which user has to upload the Image (content) on the TEXTIFY site and site shows the output in editable text.
User can edit the text in output box.
User have access to change and manipulate the data.



The Frontend consists of Web interface in which user has to upload an image on which he can get it by generating editable text in the image.

7.1. Description of Attributes

User:- The one that will interacts with the Optical Capture Recognition (OCR).

The Image:- Selected from the phone: must contain text.

The Text:- must have words.

Words:- English Alphabet.

7.2. Description of the relationship between the modules

1. User:- Web-Site: It is many to one relationship because many users interacts with the website at a time.
2. OCR Site:- Image: The website is able to process one image at a time.
3. User:- Image: The user can choose one Image from the phone directory.
4. Image:-Text: The image must contain text.
5. Text-Words:- The text can contain many words.
6. Words:- Characters:- Each word can have any number of English Alphabet's characters.
7. User:- Text: The user can copy, paste, or select the whole text or just a part of it.

8. Testing

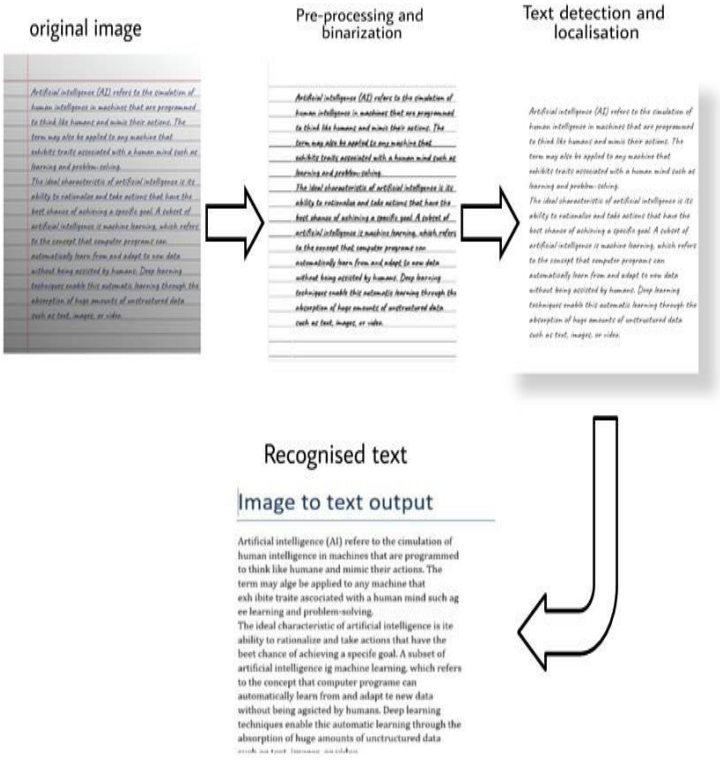
a) **Performance Measure:-**In order to get better performance of text extraction the combination of powerful algorithms is used; Edge Based Methods and Connected Component Method. These algorithms performance can be analyzed and evaluated by precision rate, recall rate, f-score and accuracy. The result of the measures differs based on the input image type means scale or size difference, lighting variance, and orientation or direction variances.Both edge based and connected component methods have their own strengths as well as weaknesses. In the proposed system these limitations would be decreased and the performance of text extraction from image documents. The table7.1.1 below shows the overall result of the two algorithms Edge based and Connected Component using different images with respective output comparing with proposed one.

Measures	Edge Based	Connected Components	Proposed
Precision Rate	80.78%	88.05%	96.11%
Recall Rate	86.87%	86.91%	93.58%
F-Score	83.71%	87.48%	94.83%
Accuracy	72.24%	76.55%	90.42%

Table 8.1 Comparison of all algorithms results

b) **Results:-**

Fig. 8.1 The result of the proposed System



c) **Result Analysis:-**In the proposed system system we have been using open source tesseract OCR engine.The paper document is generally scanned by the optical scanner and is converted in to the form of a picture . The picture is given as input to the system. Pre-processing and binarisation is done, then the image is fed to the OCR engine which includes sub process like feature extraction, classification, text region detection , text localization and text extraction . Finally the output text is display in the editable format.
The proposed system gives good performance in text extraction with high accuracy, it also givesuseful outputs on handwritten text and complexfont text. User can also edit the output text according to the convenience.

9. Technologies Used

- **Front End:- HTML, CSS, JavaScript :-** The front-end stack is made up of many different languages and libraries. While these vary from application to application, there are only a few generic languages understood by all web browsers. These three main front-end coding languages are HTML, CSS and JavaScript. HTML is the first layer of any website and creates the code version of a wireframe on a webpage. These wireframes exist for the styles in CSS and all the bells and whistles in JavaScript. Cascading Style Sheets, or CSS, is what gives our HTML visual appeal and draws in the user. To put it simply, style sheets dictate the presentation of HTML elements on a page. JavaScript is a runtime language for web browsers. This means that when you open a web page, the page will load both the foundational JavaScript that is standard with the page and any new JavaScript added to a page. The new JavaScript will load in parallel with it and can perform actions and make decisions.

- **Framework:- Django :-** Django is a free and open source web application framework written in Python. A framework is nothing more than a collection of modules that make development easier. They are grouped together, and allow you to create applications or websites from an existing source, instead of from scratch.

  The official project site describes Django as "a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It's free and open source.

- **Backend:- Python :-** Python can be used for either front-end or back-end development. That said, it's approachable syntax and widespread server-side use makes Python a core programming language for backend development. Python is open source andworks with the Flask and Django web frameworks. Because of its emphasis on codereadability and its vast librariesand frameworks, Python is anideal backend language when it comes to this area of technology. Moreover, Python has a vast set of libraries specific to machine learning, such as Keras, TensorFlow and Scikit-learn.

10. CONCLUSIONS

Extraction of text from image documents is very important in different areas nowadays. In this we proposed the algorithm which gives good performance in text extraction by combining two algorithms, Edge Based and Connected Components. The extracted text recognition done by OCR with better accuracy and finally audio output produced. The paper does not include handwritten and complex font text which can be future work.

This technology enables scan documents to become more than just files, turning into fully searchable & editable document with text content that is recognized by computers.

Image analysis involves the study of segmentation, feature extraction, & classification techniques. with the help of this technology, people no longer need to manually retype important documents.

11. REFERENCES

[1] K. Karthick, R. Francis , "A Text Recognistion" in IEEE Trans. Image Processing, Vol. 23, No. 9, Sep 2019.

[2] SuparnaKarmarkar, TridevChakraborty , "Text Recognition using image Processing" JoAIRA, STM Journal, 2017.

[3] K. Natei, S. Sasikumar , "Extracting text from document using connected component algorithm" , International Journal of Innovative Research in Computer and Communication Engineering, 2018.

[4] Jaswinder Kaur, Rupinder Kaur , "Chracter Recognition using", International Journal of Computer Application, NSFTICE, 2018.

[5] A. Panchal, Sh. Varde, "Comparative study of Image processing techniques used for Scene text detection and extraction", International Journal of Engineering Research and General Science (IJERGS), 2016.

[6] Shivani, D. Bansal, "Techniques of Text Detection and Recognition: A Survey", International Journal of Emerging Research in Management & Technology (IJERMT), 2017.

[7] N.Syal, N. K. Garg, "Text Extraction in Images Using DWT,Gradient Method And SVM Classifier", 2014.

[8] Ganesh, L. G. Malik, "Extraction of Text from Images of Big Data" IJARCSMS, 2014.