**A Project Report**

on

# OPTICAL CHARACTER RECOGNITION AND TEXT TRANSLATION

***Submitted in partial fulfilment of the***

***requirement for the award of the degree of***

Bachelor of Technology

GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

**Under The Supervision of**

**Dr. S. Jerald Nirmal Kumar**

**Assistant Professor**

Submitted By

**ABHISHEK KUMAR CHAUDHARY (18SCSE1010661)**

**AKSHAY RAJ (18SCSE1010617)**

**SCHOOL OF COMPUTING SCIENCE AND ENGINEERING DEPARTMENT OF**

**COMPUTER SCIENCE AND ENGINEERING**

**GALGOTIAS UNIVERSITY, GREATER NOIDA**

**INDIA**

**December- 2021**

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

GALGOTIAS UNIVERSITY, GREATER NOIDA

# CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled **"OPTICAL CHARACTER RECOGNITION"** in partial fulfilment of the requirements for the award of the **Bachelor of Technology In Computer Science And Engineering** submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of **July,2021 to December, 2021**, under the supervision of **Dr. S. Jerald Nirmal Kumar**, Department of Computer Science and Engineering/Computer Application and Information and Science, of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by me/us for the award of any other degree of this or any other places.

ABHISHEK KUMAR CHAUDHARY - 18SCSE1010661

AKSHAY RAJ - 18SCSE1010617

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

**(Dr. S. Jerald Nirmal Kumar)**

**Assistant Professor**

## CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of **Abhishek Kumar Chaudhary (18SCSE1010661), Akshay Raj(18SCSE1010617)** has been held on __________________ and his/her work is recommended for the award of Bachelor of Technology~~.~~

**Signature of Examiner(s)** **Signature of Supervisor(s)**

**Signature of Project Coordinator** **Signature of Dean**

Date: December, 2021

Place: Greater Noida

# ACKNOWLEDGEMENT

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I am highly indebted to Dr. S. Jerald Nirmal Kumar for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

I would like to express my gratitude towards my parents & member of Galgotias University for their kind co-operation and encouragement which help me in completion of this project.

I would like to express my special gratitude and thanks to industry persons for giving me such attention and time.

My thanks and appreciations also go to my colleague in developing the project and people who have willingly helped me out with their abilities.

# ABSTRACT

In our day-to-day life the people are facing many problems in understand the languages. For example, if the people move from one state to the other, they don’t understand their language at that time this Mobile Application will help them. Existing system, having a separate application for each and every process like camera, Google translator and Optical Character Recognition (OCR) text scanner. But people expect the application consists of all the three facilities together. So, this proposed application provides a new idea to the people to translate the other language text into their known language.

This application contains three steps: -

1.Take a photo image of the unknown language text which you want to translate (either handwritten or printed material),

2.Tessaract is an open-source Optical Character Recognition (OCR) technology, which is used to extract the text from the image then Google API and Bing API is used for translation of language.

3.The translated text is generated in PDF format.

Keywords: - Text Extraction, Android, OCR, Tesseract.

# LIST OF FIGURES

# Table of Contents

CHAPTER-1

# INTRODUCTION

What is OCR?

OCR stands for "Optical Character Recognition." It is a technology that recognizes text within a digital image. It is commonly used to recognize text in scanned documents and images. OCR software can be used to convert a physical paper document, or an image into an accessible electronic version with text. For example, if you scan a paper document or photograph with a printer, the printer will most likely create a file with a digital image in it. The file could be a JPG/TIFF or PDF, but the new electronic file may still be only an image of the original document. You can then load this scanned electronic document it created, which contains the image, into an OCR program. The OCR program which will recognize the text and convert the document to an editable text file.

How Does OCR Work?

OCR software processes a digital image by locating and recognizing characters, such as letters, numbers, and symbols. Some OCR software will simply export the text, while other programs can convert the characters to editable text directly in the image. Advanced OCR software can export the size and formatting of the text as well as the layout of the text found on a page. Does OCR Create an Accessible Document?

The short answer is no, not really. Some OCR programs allow you to scan a document and convert it to a word processing document in a single step, but this still may not be an accessible document. Once you use the OCR to process your document, you must select the text and read it to verify the process was successful, and the text makes sense. You may have to spel lcheck it, add headings, add tags, reorder it and more. You can do this with your word processor such as Word, or Adobe Acrobat Pro.

Do I really need to Proofread and Correct an OCR output?

Yes! Think of it this way: If your original had really good contrast and readability a 99% success rate is possible with some OCR software, but what if the 1 % wrong was the tuition rate for the college? If the original image had poor contrast and readability the success rate could go down to 50% or even be unreadable. You won’t know until you check it!

What if My OCR Output is Really Bad?

If your original document or image has poor contrast, fuzzy characters, overlaps, etc. the OCR software may recognize the text, but the text may not be accurate and be hard to read to make your OCR output is more successful with easier to verify and correct output, make sure your original is not a fuzzy reprint. The original should have good contrast and sharp letters. If you cannot get your hands on a better version of the original, the printer, or printer software, may have settings on it that will produce a better scan.

Text extraction from image is one of the complicated areas in digital image processing. It is a complex process to detect and recognize the text from image. It’s possible of computer software can provide extracted text from image using most complicated algorithm. So it can’t be use anywhere in this existing environment. Here different types of language translators are available such as voice based translator, keyboard based translator etc. But those translators are not easy to use. The purpose of this work is to demonstrate that a tight dynamical connection may be made between text and interactive visualization imagery. The Android device camera can prove this type of extraction and also the algorithm will easily implemented using java language. Millions of mobile users in this world and they always have mobile in their hand, so simply they can capture the image to extract the text.

The purpose of this project is to implement text extraction from the image and translating the text. Captured text information from camera in natural scene images can serve as indicative marks in many image based applications such as assistive navigation, auxiliary reading, image retrieval, scene understanding, etc. Extracting text from natural scene images is a more challenging problem as compared to scanned document because of complex backgrounds and also large variations of text patterns such as font, color, scale, intensity and orientation.

Therefore, to extract text from camera captured images, text detection & extraction is an important and essential step which computes the subregions of the images containing text characters or strings. Once the image is captured from camera, the image went through various processes whose task is to detect the text within the image and extract those texts then translates that text.

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 as we know that we have number of newspapers and books which are in printed format related to different subjects. These days there is a huge demand in "storing the information available these paper documents in to a computer storage disk and then later reusing this information by searching process".

One simple way to store information in this 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 reason for this difficulty is the font characteristics of the characters in paper documents are different to font of the characters in computer system. As a result, computer is unable to recognize the characters while reading them. This concept of storing the contents of paper documents in computer storage place and then reading and searching the content is called DOCUMENT PROCESSING. Sometimes in this document processing we need to process the information that is related to languages other than the English in the world. For this document processing we need a software system called CHARACTER RECOGNITION SYSTEM. This process is also called DOCUMENT IMAGE ANALYSIS (DIA). Thus, our need is to develop character recognition software system to perform Document Image Analysis which transforms documents in paper format to electronic format. For this process there are various techniques in the world. Among all those techniques we have chosen Optical Character Recognition as main fundamental technique to recognize characters. The conversion of paper documents in to electronic format is an on-going task in many of the organizations particularly in Research and Development (R&D) area, in large business enterprises, in government institutions, so on. From our problem statement we can introduce the necessity of Optical Character Recognition in mobile electronic devices such a cell phones, digital cameras to acquire images and recognize them as a part of face recognition and validation.

To effectively use Optical Character Recognition for character recognition in-order to perform Document Image Analysis (DIA), we are using the information in grid format. This system is thus effective and useful in Virtual Digital Library's design and construction.

## What are we making and business use-case?

We are making an Optical Character Recognition and Text Translation.

- Optical Character Recognition
    - It is one of the most important day eye concept. It is used in mutliple AI project such as traffic number plate detection system and handwriting recognition system.
- Text Translation
    - It is also one of the most important AI concept used in many application for translation of text to particular target language.

The main purpose of Optical Character Recognition (OCR) system based on a grid infrastructure is to perform Document Image Analysis, document processing of electronic document formats converted from paper formats more effectively and efficiently. This improves the accuracy of recognizing the characters during document processing compared to various existing available character recognition methods. Here OCR technique derives the meaning of the characters, their font properties from their bit-mapped images.

> The primary objective 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.

**Use Case:** We can use it to identify images on your camera and gain more information about landmarks, places, plants, animals, products, and other objects. It can also be used to scan and translate text.

## Project Goals and Objectives

1. Basics and quick theory of optical character recognition and translation engine.
2. Create our own optical character recognizer from scratch.
3. Different ways of building OCR.

After these three steps we will be able to recognize text from images. Now next step involves translation.

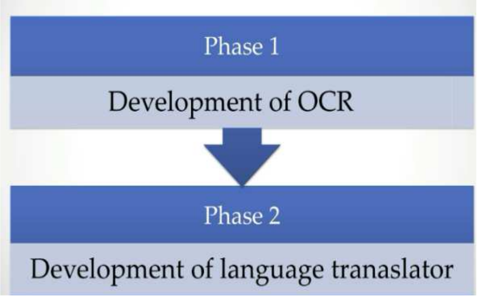
4. Here we convert the text into particular target language.

## PROJECT SCOPE

The scope of our product Optical Character Recognition on a grid infrastructure is to provide an efficient and enhanced software tool for the users to perform Document Image Analysis, document processing by reading and recognizing the characters in research, academic, governmental and business organizations that are having large pool of documented, scanned images. Irrespective of the size of documents and the type of characters in documents, the product is recognizing them, searching them and processing them faster according to the needs of the environment.

## EXISTING SYSTEM

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.



## DEEP LEARNING

Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behaviour of the human brain—albeit far from matching its ability—allowing it to “learn” from large amounts of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy.

Deep learning drives many artificial intelligence (AI) applications and services that improve automation, performing analytical and physical tasks without human intervention. Deep learning technology lies behind everyday products and services (such as digital assistants, voice-enabled TV remotes, and credit card fraud detection) as well as emerging technologies (such as self-driving cars). Machine learning and deep learning models are capable of different types of learning as well, which are usually categorized as supervised learning, unsupervised learning, and reinforcement learning. Supervised learning utilizes labeled datasets to categorize or make predictions; this requires some kind of human intervention to label input data correctly. In contrast, unsupervised learning doesn’t require labeled datasets, and instead, it detects patterns in the data, clustering them by any distinguishing characteristics. Reinforcement learning is a process in which a model learns to become more accurate for performing an action in an environment based on feedback in order to maximize the reward.

Deep neural networks consist of multiple layers of interconnected nodes, each building upon the previous layer to refine and optimize the prediction or categorization. This progression of computations through the network is called forward propagation. The input and output layers of a deep neural network are called visible layers. The input layer is where the deep learning model ingests the data for processing, and the output layer is where the final prediction or classification is made.

## FORMULATION OF PROBLEM

### I. RETYPING PROBLEMS

OCR technology enables scanned documents and images to be transformed into searchable and editable document formats. It will help you solve the problem of retyping.

A mobile scanner, can scan the document images or the photos captured and retrieve texts from them. You can edit the recognition results and save them. Wherever you are, you can search for the documents you need with entering few keywords.

The recognition accuracy reaches 99%, and it supports English, Simplified Chinese, Traditional Chinese, French, Spanish, German, Italian, Portuguese, Dutch, Danish, Swedish and Finnish. It is free.

### II. EXTRACTING DATA FROM STRUCTURED, UNSTRUCTURED DOCUMENTS AND HAND-WRITTEN DOCUMENTS

One of the major problems OCR technologies can help in solving is extracting data from structured, unstructured documents, and hand-written documents, making it easy for organizations to automate the data entry or document digitization tasks. Other problems that OCR technology helps in solving includes:

- OCR eliminates the chances of errors in data entry task that usually arise while performing it manually.
- It helps in the identity verification of individuals. In the verification process, OCR helps to extract the user information

from their identity document and match it with the one already provided by the user.

- OCR reduces the time required to convert manual documents into digital form.
- In addition to time, the cost of digitizing the documents also reduces.

III. DRAWING

The drawback in the early OCR systems is that they only have the capability to convert and recognize only the documents of English or a specific language only. That is, the older OCR system is uni-lingual.

## PROPOSED SYSTEM

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.

## BENEFIT OF PROPOSED SYSTEM

The benefit of proposed system that overcomes the drawback of the existing system is that it supports multiple functionalities such as editing and searching. It also adds benefit by providing heterogeneous characters recognition.

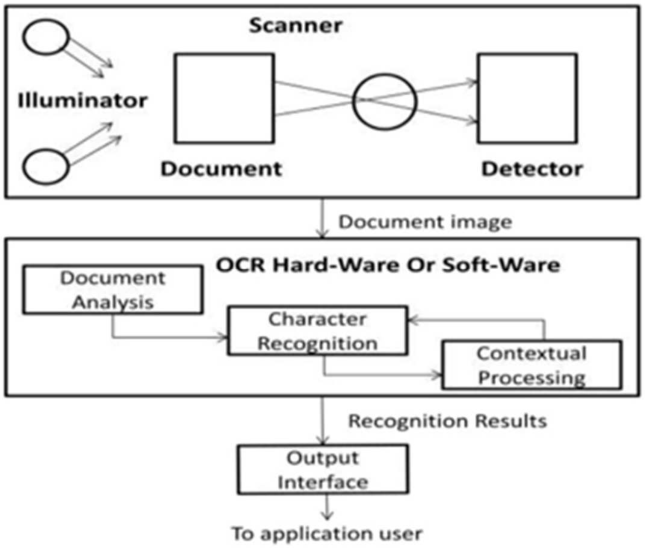
## ARCHITECTURE OF THE PROPOSED SYSTEM

The Architecture of the optical character recognition system on a grid infrastructure consists of the three main components. They are:

➤ Scanner

➤ OCR Hardware or Software

➤ Output Interface



OCR Architecture

# TOOLS AND TECHNOLOGY USED:

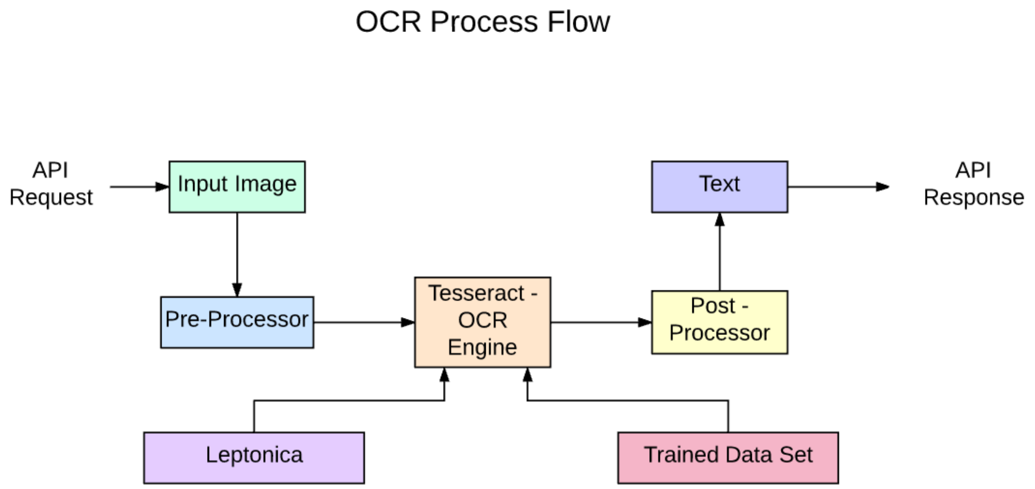
## Tesseract OCR

Tesseract is an open-source OCR engine that was developed at HP between 1984 and 1994. Like a supernova, it appeared from nowhere for the 1995 UNLV Annual Test of OCR Accuracy, shone brightly with its results, and then vanished back under the same cloak of secrecy under which it had been developed. Now for the first time, details of the architecture and algorithms can be revealed.

Tesseract began as a PhD research project in HP Labs, Bristol, and gained momentum as a possible software and/or hardware add-on for HP's line of flatbed scanners. Motivation was provided by the fact that the commercial OCR engines of the day were in their infancy, and failed miserably on anything but the best quality print.

After a joint project between HP Labs Bristol, and HP's scanner division in Colorado, Tesseract had a significant lead in accuracy over the commercial engines, but did not become a product. The next stage of its development was back in HP Labs Bristol as an investigation of OCR for compression. Work concentrated more on improving rejection efficiency than on base-level accuracy. At the end of this project, at the end of 1994, development ceased entirely. The engine was sent to UNLV for the 1995 Annual Test of OCR Accuracy, where it proved its worth against the commercial engines of the time. In late 2005, HP released Tesseract for open source. It is now available at http://code.google.com/p/tesseract-ocr.

Tesseract is an opensource text recognition (OCR) Engine, available under the Apache 2.0 license. It can be used directly, or (for programmers) using an API to extract printed text from images. It supports a wide variety of languages. Tesseract doesn't have a built-in GUI, but there are several available from the 3rd Party page. Tesseract is compatible with many programming languages and frameworks through wrappers that can be found here. It can be used with the existing layout analysis to recognize text within a large document, or it can be used in conjunction with an external text detector to recognize text from an image of a single text line.

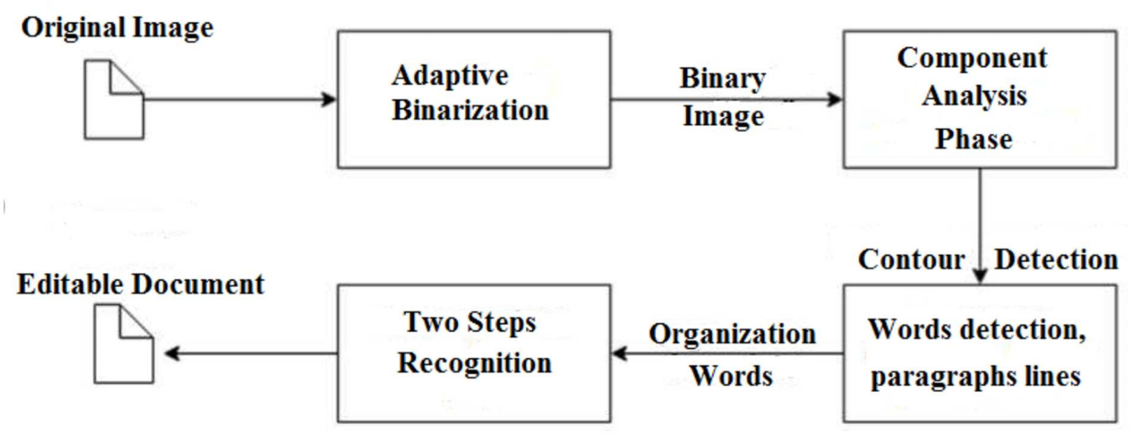


Tesseract 4.00 includes a new neural network subsystem configured as a text line recognizer. It has its origins in OCRopus' Python-based LSTM implementation but has been redesigned for Tesseract in C++. The neural network system in Tesseract pre-dates TensorFlow but is compatible with it, as there is a network description language called Variable Graph Specification Language (VGSL), that is also available for TensorFlow.

To recognize an image containing a single character, we typically use a Convolutional Neural Network (CNN). Text of arbitrary length is a sequence of characters, and such problems are solved using RNNs and LSTM is a popular form of RNN. Read this post to learn more about LSTM.

# Technology - How it works

LSTMs are great at learning sequences but slow down a lot when the number of states is too large. There are empirical results that suggest it is better to ask an LSTM to learn a long sequence than a short sequence of many classes. Tesseract developed from OCRopus model in Python which was a fork of a LSMT in C++, called CLSTM. CLSTM is an implementation of the LSTM recurrent neural network model in C++, using the Eigen library for numerical computations.



Legacy Tesseract 3.x was dependant on the multi-stage process where we can differentiate steps:

## Word Finding

Word finding was done by organizing text lines into blobs, and the lines and regions are analysed for fixed pitch or proportional text. Text lines are broken into words differently according to the kind of character spacing. Recognition then proceeds as a two-pass process. In the first pass, an attempt is made to recognize each word in turn. Each word that is satisfactory is passed to an adaptive classifier as training data. The adaptive classifier then gets a chance to more accurately recognize text lower down the page.

## **Line Finding**

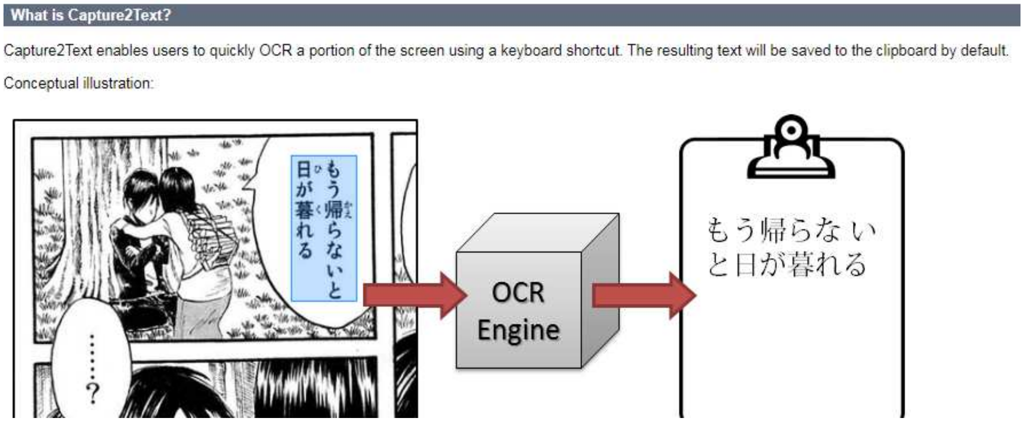
The line finding algorithm is one of the few parts of Tesseract that has previously been published. The line finding algorithm is designed so that a skewed page can be recognized without having to de-skew, thus saving loss of image quality. The key parts of the process are blob filtering and line construction.

Assuming that page layout analysis has already provided text regions of a roughly uniform text size, a simple percentile height filter removes drop-caps and vertically touching characters. The median height approximates the text size in the region, so it is safe to filter out blobs that are smaller than some fraction of the median height, being most likely punctuation, diacritical marks and noise.

The filtered blobs are more likely to fit a model of non-overlapping, parallel, but sloping lines. Sorting and processing the blobs by x-coordinate makes it possible to assign blobs to a unique text line, while tracking the slope across the page, with greatly reduced danger of assigning to an incorrect text line in the presence of skew. Once the filtered blobs have been assigned to lines, a least median of squares fit is used to estimate the baselines, and the filtered-out blobs are fitted back into the appropriate lines.

The final step of the line creation process merges blobs that overlap by at least half horizontally, putting diacritical marks together with the correct base and correctly associating parts of some broken characters.

Modernization of the Tesseract tool was an effort on code cleaning and adding a new LSTM model. The input image is processed in boxes (rectangle) line by line feeding into the LSTM model and giving output. In the image below we can visualize how it works.



What is Capture2Text?

Capture2Text enables users to quickly OCR a portion of the screen using a keyboard shortcut. The resulting text will be saved to the clipboard by default.

Conceptual illustration:

## Capture2Text

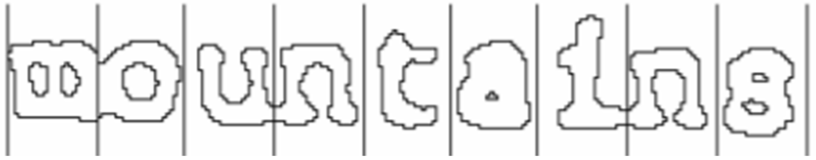
Capture2text enables users to quickly OCR a portion of the screen using a keyboard shortcut.

The resulting text will be saved to the clipboard by default.

In the above image the written text is extracted from the image through OCR system.

## Fixed Pitch Detection and Chopping

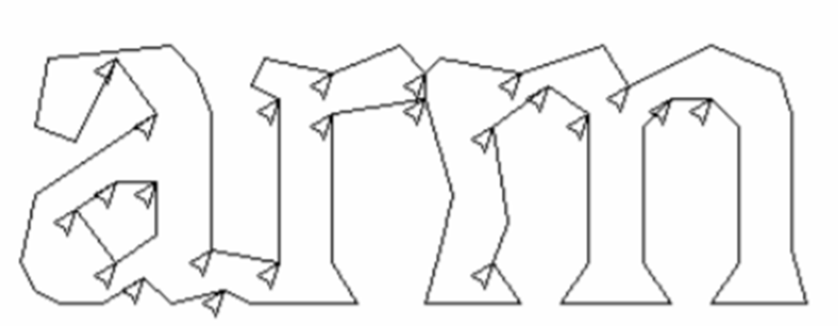
Tesseract tests the text lines to determine whether they are fixed pitch. Where it finds fixed pitch text, Tesseract chops the words into characters using the pitch, and disables the chopper and associator on these words for the word recognition step. Fig. 2 shows a typical example of a fixed-pitch word.



**A Fixed-Pitch Chopped Word**

### Chopping Joined Characters

While the result from a word  is unsatisfactory, Tesseract attempts to improve the result by chopping the blob with worst confidence from the character classifier. Candidate chop points are found from concave vertices of a polygonal approximation of the outline, and may have either another concave vertex opposite, or a line segment. It may take up to 3 pairs of chop points to successfully separate joined characters from the ASCII set.

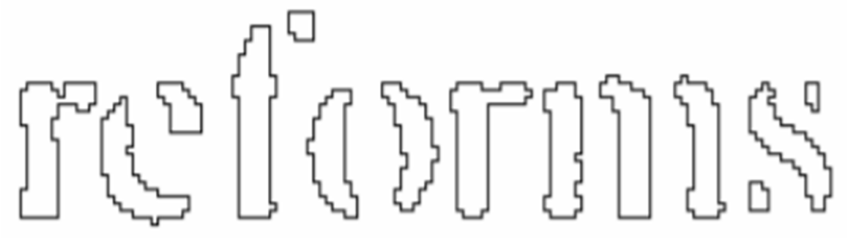


**Candidate Chop Points And Chop**

### Associating Broken Characters

When the potential chops have been exhausted, if the word is still not good enough, it is given to the associator. The associator makes an A* (best first) search of the segmentation graph of possible combinations of the maximally chopped blobs into candidate characters. It does this without actually building the segmentation graph, but instead maintains a hash table of visited states. The A* search proceeds by pulling candidate new states from a priority queue and evaluating them by classifying unclassified combinations of fragments.

It may be argued that this fully-chop-then-associate approach is at best inefficient, at worst liable to miss important chops, and that may well be the case. The advantage is that the chop-then-associate scheme simplifies the data structures that would be required to maintain the full segmentation graph.



**An Easily Recognised Word**

# CHAPTER-2

## LITERATURE REVIEW

This material serves as a guide and update for readers working in the Character Recognition area. Ayatullah Faruk Mollah, Nabamita Majumder, Subhadip Basu, and Mita Nasipuri(2011) presents a complete Optical Character Recognition(OCR) system for camera captured image textual documents for handheld devices. Firstly, text regions are extracted and skew corrected. Then, these text regions are binarized and segmented into lines and characters. Characters are passed into the recognition module. Pranob K Charles, V.Harish, M.Swathi(2012) describes the techniques for converting textual content from a paper document into machine readable form. The computer actually recognizes the characters in the document through a revolutionizing technique called Optical Character Recognition. Chirag Patel, Atul Patel, Dharmendra Patel (2012) recognize the characters in a given scanned documents and study the changes in the Models of Artificial Neural Network. It describes the behaviours of different Models of Neural Network used in Optical Character Recognition.

Neural network mostly uses the OCR. Dileep Kumar Patel, Tanmoy Som, Sushil Kumar Yadav, Manoj Kumar Singh [2012] gives the solution to the problem of handwritten character recognition. It has been tackled with multi resolution technique using Discrete wavelet transform (DWT) and Euclidean distance metric (EDM). The technique has been tested and found to be more accurate and faster. Characters is classified into 26 pattern classes based on appropriate properties. Chi et al. (2012) has proposed an effective algorithm to deal with bleed-through effects existing in the images of financial documents. Double-sided images scanned simultaneously are used as inputs, and the bleed-through effect is detected then removed after the registration of the side images. Satyajitsaha, Dnyaneshwar, Hagawane, Pravin C.Kulkarni, Swapni R.Dhamane (2013)[6] proposes the objective to recognize and extract the text from images captured by camera based mobile device, and once the text is recognized then information about the text can be obtain via Dictionary or via Web. Majida Ali Abed et al.(2013)[7] presents a new approach to simplify Handwritten Characters Recognition based on simulation of the behaviour of schools of fish and flocks of birds that is called the Particle Swarm Optimization Approach (PSOA).PSOA is convergent and more accurate in solutions that minimize the error recognition rate. Vijay Laxmi Sahu et al(2013)[8] explains that characteristics of the classification methods that have been successfully applied to character recognition and remaining problems that can be potentially solved by learning methods. Argha Roy, Diptam Dutta K Austav, Choudhury (2013)[9] explains the IRIS plant classification using Neural Network. It provides the adaptation of network weights using Particle Swarm Optimization (PSO) was proposed as a mechanism to improve the performance of Artificial Neural Network (ANN) in classification of IRIS dataset. Classification method is a machine learning technique used to predict group membership for data instances. Amir Bahador Bayat(2013)[10] proposes an efficient system that includes two main modules, the feature extraction module and the classifier module. In the first module, seven sets of discriminative features are extracted and used in the recognition system. In the second module,the adaptive neuro-fuzzy inference system is investigated. N.K.Gundu, S.M.Jadhav, T.S.Kulkarni, A.S.Kumbhar(2014)[11] explains the best ideas from the text extraction with the help of character description and stroke configuration, web context search and web mining with the help of semantic web and synaptic web at low entropy. Faisal Mohammad, Jyoti Anarase, Milan Shingote, Pratik Ghanwat(2014)[12] presents an algorithm for implementation of Optical Character Recognition (OCR) to translate images of typewritten or handwritten characters into electronically editable format by preserving font properties. OCR can easily do this by applying pattern matching algorithm. The recognized text characters are stored in editable format. Shalin A. Chopra, Amit A. Ghadge, Onkar A. Padwal, KaranS. Punjabi, and Prof. Gandhali S. Gurjar(2014) [13] presents a simple, efficient and minimum cost approach to construct OCR for reading any document that has fix font size and style or handwritten style. In this the systems have the ability to yield excellent results. It is mostly used with existing OCR methods, especially for English text. Sravan, ShivankuMahna, NirbhayKashyap (2015)[14] explains that problems being faced by the developers in using OCR as a technology on a large scale and give the solution to that problem. This system provides many features that require no typing, editing raw data, quick translation, and memory utilization.Surabhi Dusane, Monica Ahuja, Rucha Ghodke & Prathamesh Kothawade (2016)[15]The objective in this paper is to develop user friendly system which will extract text from images and convert the extracted text into user friendly language then it will convert it into audio which describes the text more efficiently.

## INTENDED AUDIENCE AND READING SUGGESTIONS

In this section, we identify the audience who are interested with the product and are involved in the implementation of the product either directly or indirectly. As from our research, the OCR system is mainly useful in R&D at various scientific organizations, in governmental institutes and in large business organizations, we identify the following as various interested audience in implementing OCR system:

➤ The scientists, the research scholars and the research fellows in telecommunication institutions are interested in using OCR system for processing the word document that contains base paper for their research.

➤ The Librarian to manage the information contents of the older books in building virtual digital library requires use of OCR system.

➤ Various sites that vendor e-books have a huge requirement of this OCR system in order to scan all the books in to electronic format and thus make money. The Amazon book world is largely using this concept to build their digital libraries.

Now we present the reading suggestions for the users or clients through which the user can better understand the various phases of the product. These suggestions may be effective and useful for the beginners of the product rather than the regular users such as research scholars, librarians and administrators of various web-sites. With these suggestions, the user need not waste his time in scrolling the documents up and down, browsing through the web. visiting libraries in search of different books and... The following are the various reading suggestions that the user can follow in order to completely understand about our product and to save time:

➤ It would help you if you start with Wikipedia.com. It lets you know the basic concept of every keyword you require. First leam from it what is OCR? And how does it work based on a Grid infrastructure?

➤ Now you can proceed your further reading with the introduction of our product we provided in our documentation. From these two steps you completely get an in depth idea of the use of our product and several processes involved in it.

➤ The more you need is the implementation of the product. For this you can visit FreeOCR.com where you can view how the sample OCR works and you can try it.

# SOFTWARE DESIGN

## DATA FLOW DIAGRAM

The DFD is also called as bubble chart. A data-flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFD's can also be used for the visualization of data processing. The flow of data in our system can be described in the form of dataflow diagram as follows:

1. Firstly, if the user is administrator be con initialise the following actions:

> Document processing

> Document search

> Document editing.

All the above actions come under 2cases. They are described as follows:

a) If the printed document is a new document that is not yet read into the system, then the document processing phase reads the scanned document as an image only and then produces the document image stored in computer memory as a result. Now the document processing phase has the document at its hand and can read the document at any point of time. Later the document processing phase proceeds with recognizing the document using OCR methodology and the grid infrastructures. Thus it produces the documents with the recognized characters as final output which can be later searched and edited by the end-user or administrator.

b) If the printed document is already scanned in and is held in system memory, then the document processing phase proceeds with document recognition using OCR methodology and grid infrastructure. And thus it finally produces the document with recognised documents as output.

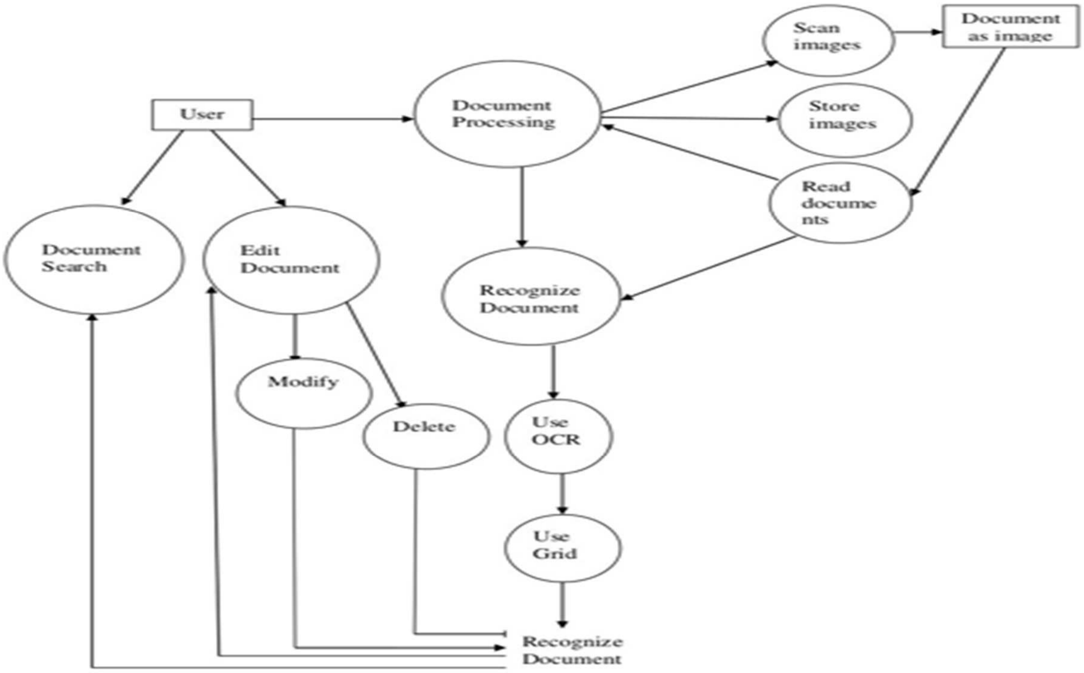
2. If the user using the OCR system is the end-user, then he can perform the following actions:

> Document searching

> Document editing

1. Document Searching: - The documents which are recognized can be searched by the user whenever required by requesting from the system database.

2. Document Editing: - The recognized documents can be edited by adding the specific content to the document, deleting specific content from the document and modifying the document.



## CLASS DIAGRAMS

The class diagram is the main building block in object-oriented modeling. The classes in a class diagram represent both the main objects and or interactions in the application and the objects to be programmed.

The class diagram of our OCR system consists of 9 classes. They are

1. Main Screen

2. Editor

3. Help Frame

4. Document

5. HEntry

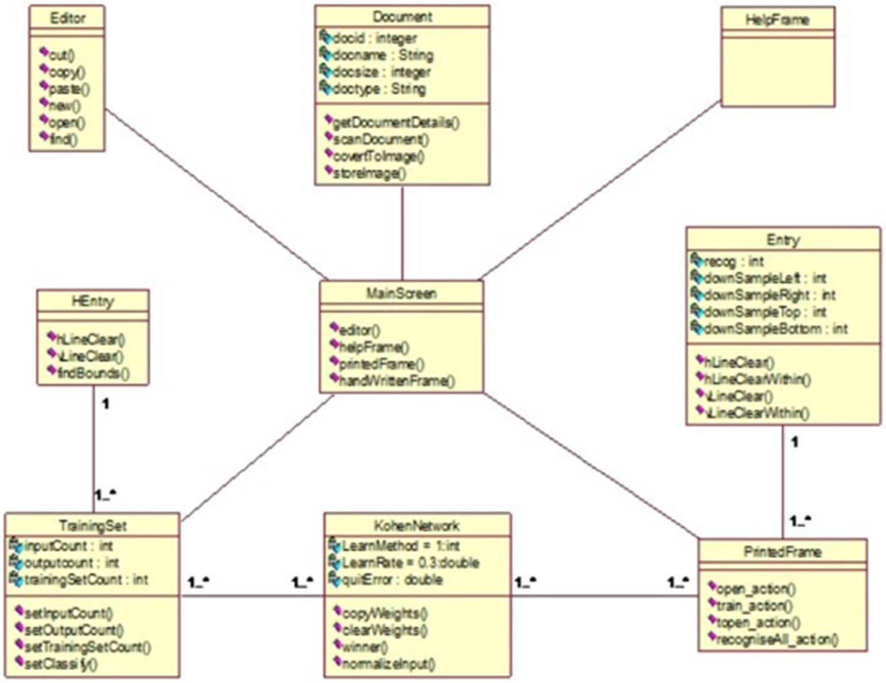
6. Entry

7. Training Set

8. Kohonen Network

9. Printed Frame.

Among all these classes the Main Screen is the main class that represents all the major functions carried out by our OCR system. The Main Screen class has an association with five classes viz, Editor, Help Frame, Document. Training Set, Printed Frame. And the Training Set class in-tum has an association with the HEntry and the Kohonen Network classes. The Printed Frame has an association with the Entry and Kohonen Network classes.



**Class Diagram**

# SEQUENCE DIAGRAMS

Sequence diagrams are sometimes called Event-trace diagrams, event scenarios, and timing diagrams. A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

In sequence diagram, the class objects that are used to describe the interaction between various classes vary from one function to another function. There are five sequence diagrams short-listed below for presenting the sequence of actions performed by each of the five modules. The key class object involved in all of these module functions is Main Screen class which controls the interaction among various class objects.

## Sequence Diagram for Document Processing

### 1. Objects

Administrator "a"

Main Screen - "m"

Document-"d"

System Memory- "s"

### 2. Links

1. Administrator object to Main Screen object.

2. Main Screen object to Document object.

3. Document object to System Memory object.

4. System Memory object to Administrator object.

**3. Messages**

1 Process documents

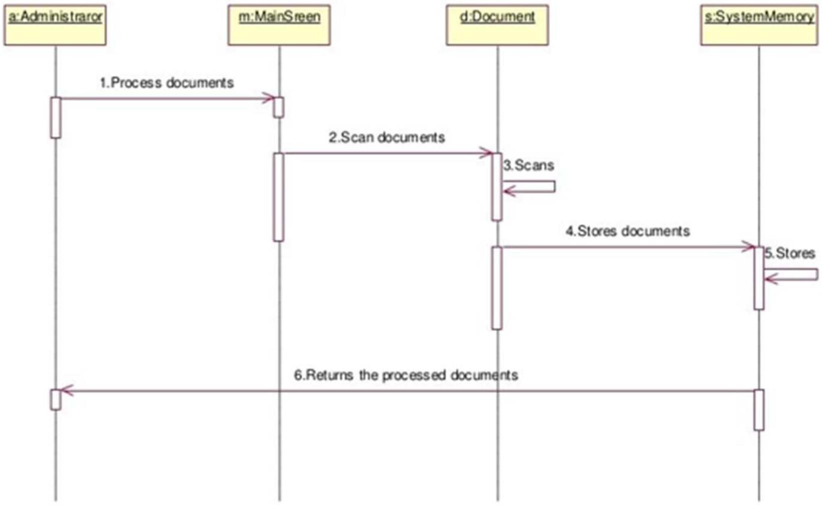
2. Scan documents

3. Scans

4. Stores documents

5. Stores

6. Returns the processed documents



**Sequence Diagram for Document Processing**

## Sequence Diagram for System Training

### 1. Objects

Administrator-"a"

System - ""

Training Set - ""

### 2. Links

1. Administrator object to System object

2. System object to Training Set object

3. Training Set object to System object

4. System object to Administrator object

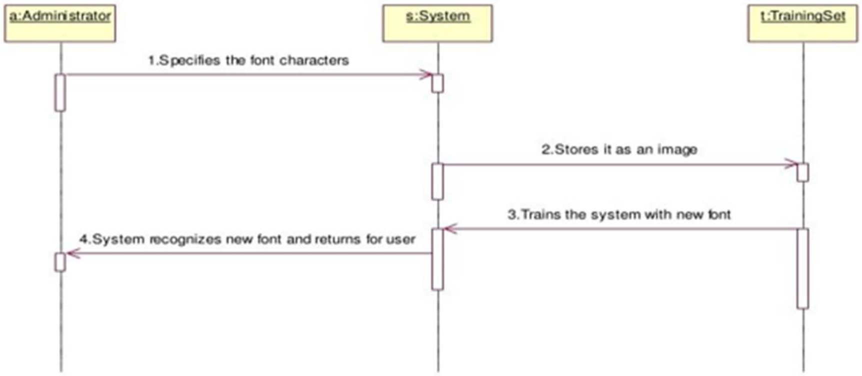
### 3. Messages

1. Specifies the font characters

2. Stores it as an image

3. Trains the system with new font

4. System recognizes new font and returns for user


a:Administrator
s:System
t:TrainingSet
1.Specifies the font characters
2.Stores it as an image
3.Trains the system with new font
4.System recognizes new font and returns for user

**Sequence Diagram For System Training**

## Sequence Diagram for Document Recognition

### 1. Objects

Administrator - "a"

MainScreen - "m"

SystemMemory - ""

TrainingSet - "

### 2. Links

1. Administrator object to MainScreen object

2. MainScreen object to System Memory object

3. SystemMemory object to MainScreen object

4. MainScreen object to Training Set object

5. Training Set object to MainScreen object

6. MainScreen object to Administrator object

**3. Messages**

1. Recognize documents.

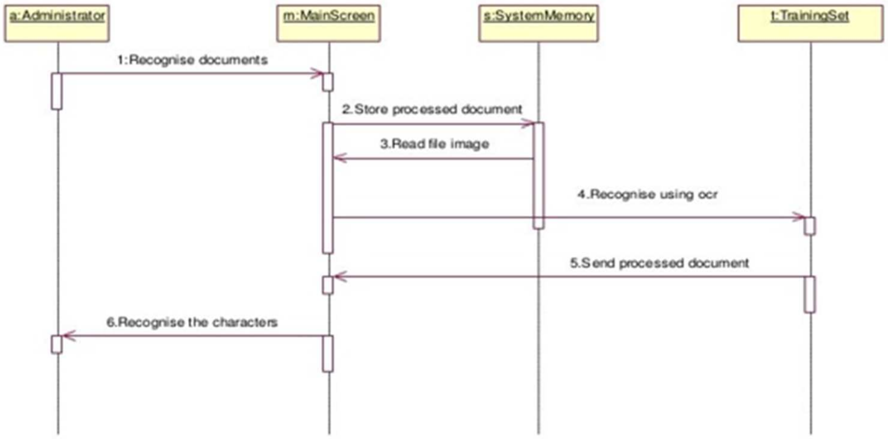
2. Store processed document

3. Read file image

4. Recognize using OCR

5. Send processed document

6. Recognize the characters.



**Sequence Diagram for Document Recognition**

## Sequence Diagram for Document Editing

### 1. Objects

Administrator - "a"

MainScreen - "m"

Document - "d"

SystemMemory- "s"

### 2. Links

1. Administrator object to MainScreen object.

2. MainScreen object to Document object.

3. MainScreen object to Document object

4. MainScreen object to Document object

5. Document object to System Memory object.

6. SystemMemory object to Administrator object.

### 3. Messages

1. Edit document.

2. Adding document

3. Adds

4. Deleting document

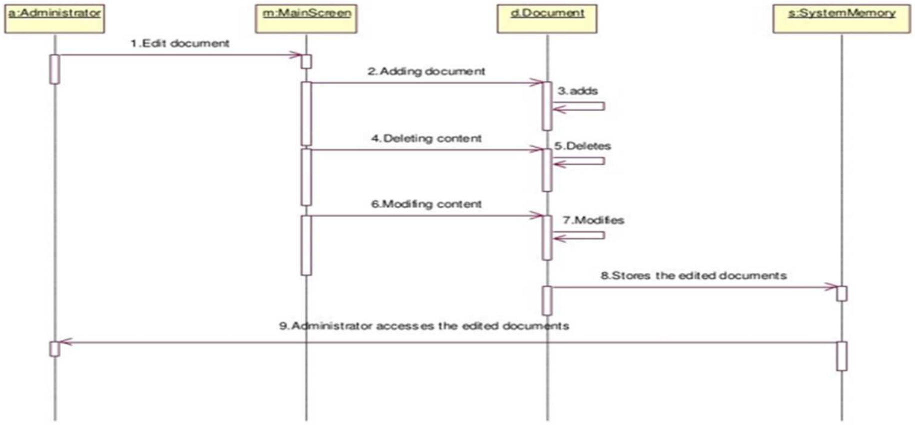
5. Deletes

6. Modifying document

7. Modifies

8. Stores the edited documents

9. Administrator accesses the edited documents



## Sequence Diagram for Document Searching

### 1.Objects

Administrator-"a"

MainScreen - "m"

Document -"d"

### 2. Links

1. Administrator object to Main Screen object

2. Main Screen object to Document object

3. Document object to Administrator object

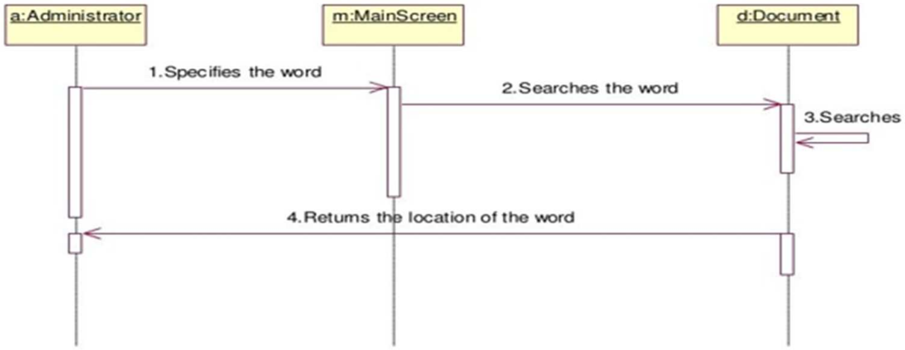
**3. Messages**

1. Specifies the word

2. Searches the word

3. Searches

4. Returns the location of the word



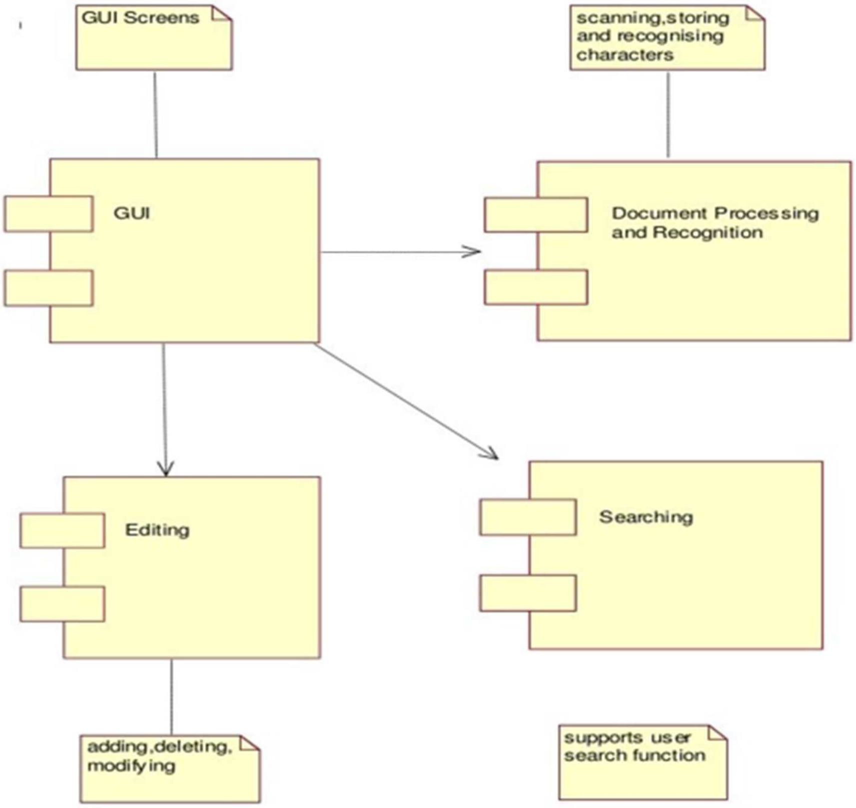
**Sequence Diagram for Document Searching**

# COMPONENT DIAGRAM

The crucial component in our component diagram that plays a major role in implementing the OCR system is the GUI component. All other components that is Document processing and recognition, Document editing and Document Searching depends on it. They are as follows:

> GUI Component that is used to design GUI screens for interacting with the end-user and administrator.

From the GUI component other components functionalities are carried out. The functionalities include Document processing and recognition, Document editing and Document Searching.


GUI Screens
scanning,storing and recognising characters
GUI
Document Processing and Recognition
Editing
Searching
adding,deleting, modifying
supports user search function

**Component Diagram**

# CONCLUSION AND FUTURE SCOPE

What does the future hold for OCR? Given enough entrepreneurial designers and sufficient research and development dollars, OCR can become a powerful tool for future data entry applications. However, the limited availability of funds in a capital-short environment could restrict the growth of this technology. But, given the proper impetus and encouragement, a lot of benefits can be provided by the OCR system. They are:

>The automated entry of data by OCR is one of the most attractive, labor reducing technology The recognition of new font characters by the system is very easy and quick.

> We can edit the information of the documents more conveniently and we can reuse the edited information as and when required.

> The extension to software other than editing and searching is topic for future works. The Grid infrastructure used in the implementation of Optical Character Recognition system can be efficiently used to speed up the translation of image based documents into structured documents that are currently easy to discover, search and process.

## FUTURE ENHANCEMENTS

The Optical Character Recognition software can be enhanced in the future in different kinds of ways such as:

Training and recognition speeds can be increased greater and greater by making it more user-friendly.

Many applications exist where it would be desirable to read handwritten entries. Reading handwriting is a very difficult task considering the diversities that exist in ordinary penmanship. However, progress is being made.

# REFERENCES

Under this references section, we have mentioned various references from which we collected our problem and several others that supported us to design the solution for our problem. These references include either books, papers published through some standards and several websites links with URL's:

- For the complete reference and understanding of neural networks refer jeff heaton's chapter 1 from www.jeffheaton.com
- For the complete reference and understanding of OCR refer jeff heaton's chapter 7 from www.jeffheaton.com
- The IEEE standard reference paper from which we collected our problem statement is authorized by Dana Petcu, Silviu Panica, Viorel Negru and Andrei Eckstein of Computer Science Department who are from West University of Timisoara, Romania.
- The reference paper is also authorized by Doina Banciu from National Institute for Research and Development in Informatics, Romania.
- You can refer the IEEE standard paper written by D. Andrews, R. Brown, C. Caldwell, et al., "A Parallel Architecture for Performing Real Time Multi-Line Optical Character Recognition"
- You can refer the IEEE standard paper written by H. Goto, "OCR Grid: A Platform for Distributed and Cooperative OCR Systems".

# APPENDIX

## Appendix A: Glossary

### TERMS

All the terms and abbreviations in the project are specified clearly.

For further development of project evolved definition will be specified.

### ACRONYMS

IEEE Institute of Electrical and Electronics Engineers

DFD: Data Flow Diagram

UML: Unified Modeling Language

GUI: Graphical User Interface

OCR: Optical Character Recognition

GOCR: Grid OCR

## Appendix B: Analysis Mode

This includes all the pertinent analysis models, such as data flow diagrams, class diagrams, use case diagrams, interaction diagrams and state-chart diagrams.