A Summer Project Report on

"IMAGE2TEXT"

Submitted in partial fulfillment of the requirement for
Degree in Bachelor of Engineering (Information Technology)

By

Joel Dsilva (5019116)
Aryan Koul (5019129)
Madhur Thakkar (5019156)

Guided by:

Prof. Archana Shirke

Department of Information Technology
Fr. Conceicao Rodrigues Institute of Technology
Sector 9A, Vashi, Navi Mumbai – 400703

University of Mumbai
2021-2022

CERTIFICATE

This is to certify that the project entitled

"IMAGE2TEXT"

Submitted By

Joel Dsilva (5019116)
Aryan Koul (5019129)
Madhur Thakkar (5019156)

In partial fulfillment of degree of **B.E.**in **Information Technology** for term work of the project is approved.

External Examiner

____

Internal Examiner

____

External Guide

____

Internal Guide

____

Head of the Department

____

Principal

____

Date: -

College Seal

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

(Name of student and Roll No.)

Date:

ABSTRACT

In the current world, there is a great increase in the utilization of digital technology and various methods are available for the people to capture images. Such images may contain important textual data that the user may need to edit or store digitally. This can be done using Optical Character Recognition with the help of Tesseract OCR Engine. The system will take an image as input which can be of any standard format such as .jpg, .png, etc, which will be then be pre-processed, then passed to the recognizer which will extract the text and output it on our website. The system will be able to recognize English of different fonts and will be able to convert the hardcopy to softcopy and save time.

INDEX

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

As the world is going online so should the printed media which is mostly available only offline. There is a need of a method for storing printed texts in online format and thus preserving it for a longer time and for the future generations. We could store many valuable texts which till now have been bounded to books only. OCR, or optical character recognition, scans document images and turns them into text documents. This intelligent conversion allows documents to be searched with relevant keywords.

1.1 Background

Textual information is available in many resources such as documents, newspapers, faxes, printed information, written notes, etc. Many people simply scan the document to store the data in the computers. When a document is scanned with a scanner, it is stored in the form of images. But these images are not editable and it is very difficult to find what the user requires as they will have to go through the whole image, reading each line and word to determine if it is relevant to their need. Images also take up more space than word files in the computer. It is essential to be able to store this information in such a way so that it becomes easier to search and edit the data.

There is a growing demand for applications that can recognize characters from scanned documents or captured images and make them editable and easily accessible. Artificial intelligence is an area of computer science where a machine is trained to think and behave like intelligent human beings. Optical Character Recognition (OCR) is a branch of artificial intelligence. It is used to detect and extract characters from scanned documents or images and convert them to editable form. Earlier methods of OCR used convolutional neural networks but they are complicated and usually best suitable for single characters. These methods also had a higher error rate.

Tesseract OCR Engine makes use of Long Short Term Memory (LSTM) which is a part of Recurrent Neural Networks. It is open-source and is more suitable for handwritten texts. It is also suitable at recognizing larger portion of text data instead of single characters. Tesseract OCR Engine significantly reduces errors created in the process of character recognition. Tesseract assumes that the input image is a binary image and processing takes place step-by-step. The first step is to recognize connected components. Outlines are nested into blobs. These blobs are organized into text lines. Text lines are broken according to the pitching. If there is a fixed pitch between the characters then recognition of text takes place which is a two-pass process. An adaptive classifier is used here. Words that are recognized in the first pass are given to the classifier so that it can learn from the data and use that information for the second pass to recognize the words that were left out in the previous pass. Words that are joined are chopped and words that are broken are recognized with the help of A* algorithm that maintains a priority queue which contains the best suitable characters. Then, the user can store this information in their computers by saving them in word documents or notepads that can be edited any time they want.

1.2 Motivation

Our project is cost effective as are all other similar projects. Regarding the Societal impact it will help to preserve old newspapers, research papers and records. Since we are converting papers to digital form, the scanned sheets can be promptly discarded and it also saves space which would otherwise be occupied by the heaps of these papers. Optical Character Recognition solves a lot of problems, one of which triggered me to write this article. I realized the ability to extract texts from an image ensures a lot of possibilities such as:

- **Regulation:** Every organization needs to regulate users' activities for some reasons. The regulation might be used to protect users' rights and secure them from threats or scams. Extracting texts from an image enables an organization to process textual information on an image for regulation, especially when the images are supplied by some of the users. For example, Facebook-like regulation of the number of texts on images used for ads can be achieved with OCR. Also, hiding sensitive content on Twitter is also made possible by OCR.

- **Searchability:** Searching is one of the most common activities, especially on the internet. Searching algorithms are mostly based on manipulating texts. With Optical Character Recognition, it is possible to recognize characters on images and use them to provide relevant image results to users. In short, images and videos are now searchable with the aid of OCR.

- **Accessibility:** Having texts on images has always been a challenge for accessibility and it is the rule of thumb to have few texts on an image. With OCR, screen readers can have access to texts on images to provide some necessary experience to its users.

- **Data Processing Automation:** The processing of data is mostly automated for scale. Having texts on images is a limitation to data processing because the texts cannot be processed except manually. Optical Character Recognition (OCR) makes it possible to extract texts on images programmatically thereby, ensuring data processing automation especially when it has to do with the processing of texts on images.

- **Digitization of Printed Materials:** Everything is going digital and there are still a lot of documents to be digitized. Cheques, certificates, and other physical documents can now be digitized with the use of Optical Character Recognition.

1.3 Problem Definition

The project aims at developing an online Image to Text converter which can take entire pages of text and convert them to editable text which is easy to copy. Essentially, we can also download it as a pdf and word file along with txt file.

1.4 Scope

- To digitally store data from old documents to preserve the data for a longer run.
- It will automatically recognize typed English Script accurately and efficiently and it can be used to convert typed English text into a machine editable format/electronic format instantly.
- It can also be used for various applications in sectors such as education, government, research, postal processing, script recognition etc.

1.5 Issues/Limitations

Although the application is well designed, a pre-trained dataset has been used. We have to replace it with a custom dataset. Along with this a newer technology, other than tesseract has to be used to make the model more efficient. Right now, we can only login and use the services. Nothing is stored or no unique profiles are available. We can add a profile per user and add storage option to store either the images or text for future use.

Chapter 2

Literature Survey and Analysis

2.1 Related Work

Text detection from images is useful in many real world applications. The data that is stored in text is huge and there is need to store this data in such a manner that can be searched easily whenever required. Elimination of the use of paper is one of the steps to progress towards a world of electronics. Also, data that can be converted to audio form is a way to ease the lives of visually impaired people.

[1] In [2007] Ray Smith published an overview of Tesseract OCR Engine. It stated that Tesseract started as a PhD project sponsored by HP in 1984. In 1987, a second person was assigned to help build it better. In 1988, HPLabs joint with Scanner Division project. In 1990, the scanner product was cancelled and four years ahead HPLabs project was cancelled too. From the year 1995 till the year 2005, Tesseract was in its dark ages. But in the year 2005, it was open sourced by HP. In 2006, Google took over it. In 2008, Tesseract expanded to support six languages. By the year 2016, it was developed further to makes use of LSTM for the purpose of OCR.

[2] In [2015] Pratik Madhukar Manwatkar and Dr. Kavita R. Singh published a technical review on text recognition from images. It emphasizes on the growing demand of OCR applications as it necessary in today's world to store information digitally so that it can be edited whenever required. This information can later be searched easily as it is in digital format. The system takes image as an input, processes on the image and the output is in the form of textual data.

[3] In [2016] Akhilesh A. Panchal, Shrugal Varde and M.S. Panse proposed a character detection and recognition system for visually impaired people. They focused on the need of people that are visually impaired as it is difficult for them to read text data. This system can be used to extract text data from shop boards or direction boards and convey this information to the user in audio form. The main challenges are the different fonts of the texts on the natural scene images.

[4] In [2017] Nada Farhani, Naim Terbeh and Mounir Zrigui published a paper that stated the conversion of different modalities. Human beings have different modalities such as gesture, sound, touch and images. It is vital that they can convert the information between these modalities. The paper focuses on conversion of image to text and also on text-to-speech so that the user can hear the information whenever required.

[5] In [2017] Azmi Can Özgen, Mandana Fasounaki and Hazım Kemal Ekenel published a paper that stated how text data can be extracted from both natural scene images and computer generated images. They make use of Maximally Stable External Regions for the purpose of text detection and recognition. This method eliminated the non-text part of the image so that OCR can be done more efficiently.

[6] In [2018] Sandeep Musale and Vikram Ghiye proposed a system that is a smart reader for visually impaired people. Using this system, they can convert the text information to audio format. This system has an audio interface that the people with visual problems can

use easily. It uses a combination of OTSU and Canny algorithms for the purpose for character recognition.

[7] In [2018] Christian Reul, Uwe Springmann, Christoph Wick and Frank Puppe proposed a method to reduce the errors generated during OCR process. It includes cross fold training and voting to recognize the words more accurately. As LSTM is introduced now, it is easier to recognize words of old printed books, handwritten words, blurry or uneven words with high accuracy. A combination of ground truths and confidence values are used in this method for optimal recognition of the characters.

[8] In [2013] Prachi Solanki and Malay Bhatt have presented on text based optical character recognition using classifiers. This follows the principle of segmentation for the sentence, word and character detection in the same order. The individual words are then sent to a trained Hopfield neural network classifier for the final output. They have achieved an accuracy of 93.2%.

[9] In [2012] Pranob K Charles, V.Harish, M.Swathi, CH. Deepthi presented ‘A review on different approaches used for Optical character recognition’. In this paper the authors have described two methods OCR using neural networks and OCR using correlation method. The correlation method works by preprocessing the in-put image first the image is converted to gray-scale and then it is converted to binary image subsequently noise filtration is applied. After pre-processing segmentation follows where individual characters are segmented from the complete image. Finally this image is compared with all the examples in the training set and the class with which the image shows the highest correlation score is considered as the class for that character. For performing OCR using neural networks the image is preprocessed as in the correlation method and then segmentation is performed to segment input images in to individual lines and further segmentation is done to extract individual characters from image. The neural network consists of two layers the training set consists of images with noise and without noise.

[10] In [2013] Ana Riza F. Quiros, Alexander Abad, Rhen Anjerome Bedruz, AaronChristian Uy, Elmer P. Dadios have a proposed "A genetic algorithm and artificial neural network-based approach for the machine vision of plate segmentation and character recognition". This paper proposes a genetic-algorithm and neural network-based approach in the optimization of the process of plate segmentation and character recognition respectively in intelligent transportation systems. Upon the detection of the vehicle’s plate from a captured image, it is necessary that the individual characters in the detected plate are distinguished. After the process of plate recognition, the recognized plate number can be crossed-referenced against a database to correctly identify the vehicle’s owner and ultimately penalize him for the traffic rule he violated. The segmentation algorithm captures the region of each character in the detected plate using genetic algorithm. After which, each plate character image is mapped against its corresponding sample character image. This is done by feeding sample character images into an artificial neural network and training the network.

2.2 Existing System

2.2.1 Connected components based method

It is a well known method used for text detection from images. The connected components are extracted with help of algorithm. The resulting components are then partitioned into clusters. This approach detects pixel differences between the text and the background of the text image. It can extract and recognize the characters too.

2.2.2 Sliding window based method

This method is also known as text binarisation process. It classifies individual pixels as text or background in the textual images. The method acts as bridge between localization and reorganization by OCR

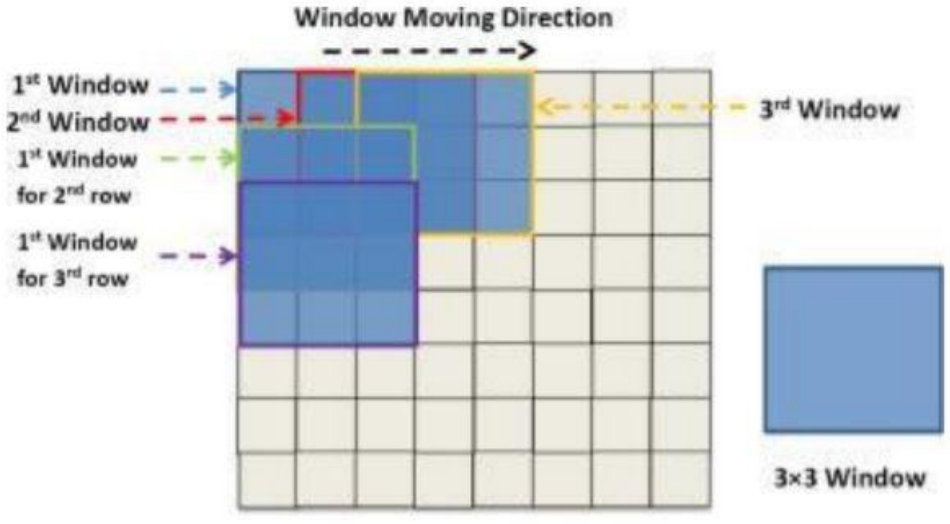


Fig -2.2.2: Sliding Window Based Method

2.2.3 Hybrid method

Hybrid method is used for text classification. This approach detects and recognizes texts in CAPTCHA images. The strength of CAPTCHA can be checked. This method efficiently detects and recognizes the text with a low false positive.

Fig -2.2.3: Captcha

2.2.4 Edge based method

This method is also known as image processing technique, which finds boundaries of the images or any other objects within the images. It works by detecting discontinuities in brightness. This approach is also used for image segmentation and data extraction in areas such as image processing, machine vision and computer vision.

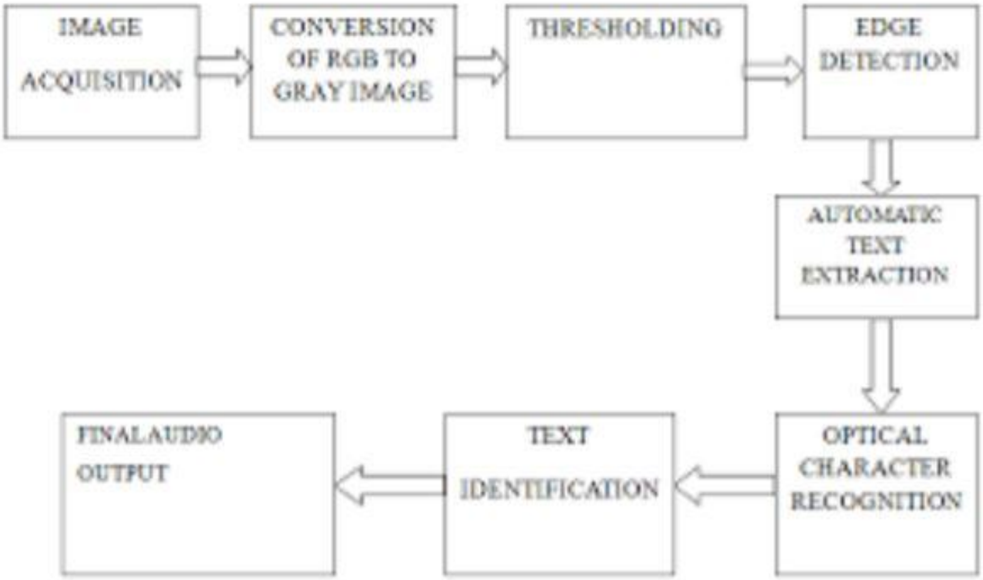


IMAGE ACQUISITION
CONVERSION OF RGB TO GRAY IMAGE
THRESHOLDING
EDGE DETECTION
AUTOMATIC TEXT EXTRACTION
OPTICAL CHARACTER RECOGNITION
TEXT IDENTIFICATION
FINALAUDIO OUTPUT

Fig -2.2.4: Edge Based Method

2.2.5 Color based method

Color based approach is used for clustering. It consists of two phases: text detection phase and text extraction phase. In text detection phase, two features are considered - homogeneous color and sharp edges, and color based clustering is used to decompose color edge map of image to several edge maps, which makes text detection more accurate. In text extraction phase, the difference between the text and background in image is considered.

2.2.6 Texture based method

It is another approach used for detecting texts in images. This approach uses Support Vector Machine (SVM) to analyze the textural properties of texts. This method also uses continuously adaptive mean shift algorithm (CAMSHIFT) that results in texture analysis. It combines both SVM and CAMSHIFT to provide robust and efficient text detection.

2.2.7 Corner based method

Corner based method is used for text extraction method. It has three stages - a) Computing corner response in multiscale space and thresholding it to get the candidate region of text; b)verifying candidate region by combining color and size range features and; c) locating the text line using bounding box. It is two-dimensional feature point which has high curvature in region boundary.

2.2.8 Stroke based method

This approach is used to detect and recognize text from the video. It uses text confidence using an edge orientation variance and opposite edge pair feature. The components are extracted and grouped into text lines based on text confidence maps. It can detect multilingual texts in video with high accuracy.

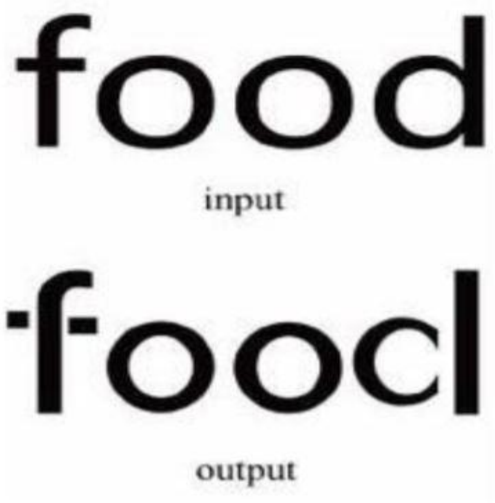


Fig -2.2.8: Stroke Based Method

2.2.9 Semi automatic ground truth generation method

This approach is also used for detecting and recognizing text from the videos. It can detect English and Chinese text of different orientations. It has attributes like: line index, word index, script type, area, content, type of text and many more. It is most efficient method to detect texts from the videos.

2.2.10 Bing Visual Search

Bing Visual Search API lets you select a region of an image and then returns the insights and text present in an image. Bing can help you find similar images, it will let you copy the text on the image, learn where to buy the dress seen in the pic, explore a landmark, identify a dog’s breed, and more. All you need is an image file or any page.

2.2.11 Photo Scan

Photo Scan is a free Windows 10 OCR app you can download from the Microsoft Store. The app is created by Define Studios and it is an OCR scanner and a QR code reader rolled into one. Point the app to an image or a file printout. The recognized text is displayed in an adjacent window. The text to speech feature is a highlight. Click the speaker icon and the app will read aloud what it just scanned. When everything is done, you can save the OCR text in multiple formats like Text, HTML, Rich Text, XML, Log format etc.

2.2.12 Text Scanner [OCR]

It is an android app where, we can convert an image to text. It takes any type of image as an input and automatically recognizes the characters from an image. And let you copy or save the text in any format.

2.2.13 (a9t9) Free OCR Windows App

(a9t9) Free OCR software is a Universal Windows Platform app. So you can use it with any Windows device you own. (a9t9) supports 21 languages for parsing your images and PDF to text. The app is also free to use, and the ad support can be removed with an in-app purchase. Like most free OCR programs, it is idea for printed documents and not handwritten text.

2.3 Requirements Analysis

- A User-Friendly and minimal interface is required for easy and smooth handling of the website.
- Creation of a User Account for future login.
- Authentication of User's Email ID.
- Hashing of User's password for confidentiality.
- To provide a Free Trial for a period of few days for the users to explore the website.
- Making the process of uploading the image to the OCR Engine easier and quicker.
- Pre-processing which is to convert the image into a format suitable for tesseract.
- To recognize the text in an image and convert it in a digitally storable format.
- To make the output text available in various digital formats such as Text, PDF and Microsoft Word.

Chapter 3

System Design

3.1 Architectural Diagram/ block diagram

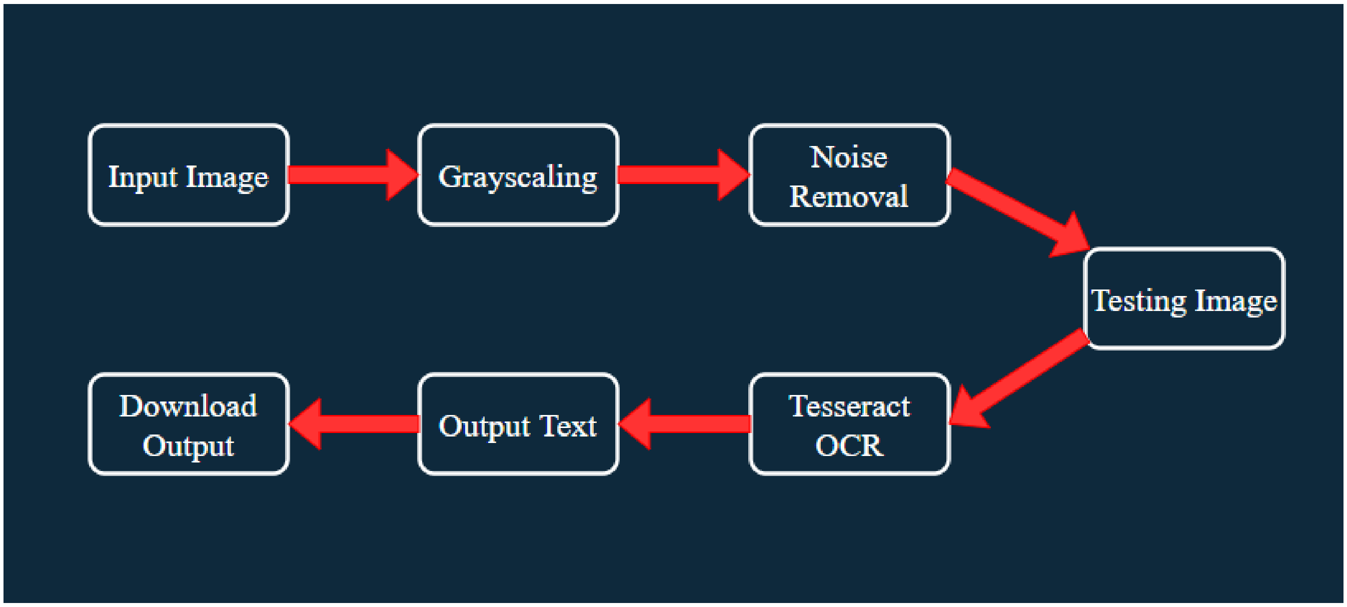


Fig -3.1.1: Architectural / Block diagram

In this section we briefly describe the overall architecture of text recognition system as shown in above figure. 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.

- **Image Acquisition:** Input image for OCR system might be acquire by scanning document or by capturing photograph of document. This is also known as digitization process.

- **Grayscaling:** It is the process of converting an image from other color spaces e.g. RGB, CMYK, HSV, etc. to shades of gray. It varies between complete black and complete white, which helps the OCR to better detect the characters.

- **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.

- **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.

- **Tesseract OCR Segmentation:** Character segmentation performs an operation of decomposition of an image into Sub images of individual symbols. It is one of the decision processes in a system for optical character recognition (OCR). It is done to make the separation between the individual characters of an image. Its decision that a pattern isolated from the image is that of a character or some other identifiable unit. Generally document is processed in hierarchical way. At first level lines are segmented using row histogram. From each row, words are extracted using column histogram and finally characters are extracted from words. Accuracy of final result is highly depends on accuracy of segmentation.

- **Feature Extraction:** Feature extraction is the important part of any pattern recognition application. 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. To store the different features of a character, the different classes are made. There are many technique used for feature extraction like Principle Component Analysis (PCA), Linear Discriminate Analysis (LDA), Independent Component Analysis (ICA), Chain Code (CC), zoning, Gradient Based features, Histogram etc. These features are also used to train the system.

- **Classification:** The classification is the process of identifying each character and assigning it to the correct character class, so that texts in images are converted in to computer understandable form. When image is provided as input to OCR system, its features are extracted and given as an input to the trained classifier like artificial neural network or Template Matching or support vector machine. Classifiers compare the input feature with stored pattern and find out the best matching class for input.

3.2 Flow Chart

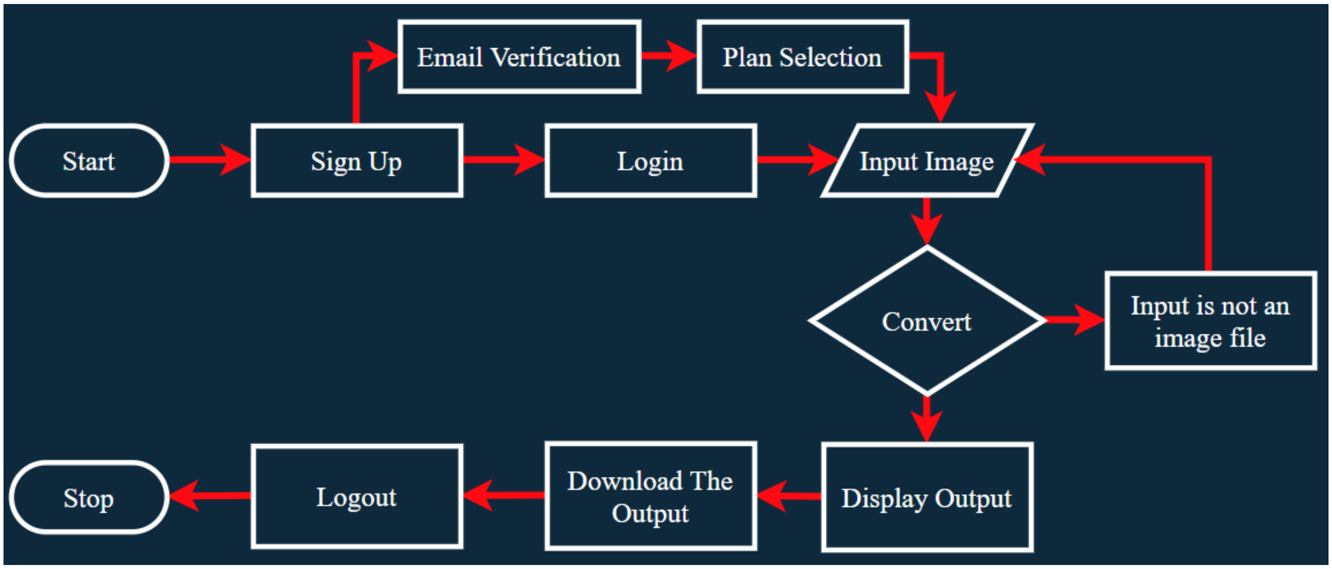


Fig -3.2.1: Flow Chart

- Start: On viewing the website the homepage will be visible where the option to signup/login will be given.
- Signup: User can signup and create an account.
  - Email Verification: For verifying the email and to create an account an otp verification system will be used.
  - Plan Selection: The user will be directed to the page containing the various plans available where the user can choose any one plan accordingly.
- Login: The user's account will be created and they can access it using their email and password.
- Input Image: The user can upload the image using either the button or using drag and drop.
- Convert: When the image is completely uploaded the convert button will be visible, which will redirect the user to download page.
  - Input is not an image file: If the file is not a valid image an alert will appear.
- Display Output: The output text will be displayed along with the input image.
- Download the Output: Output text can be downloaded in the form of text file, pdf file or word file.
- Logout: The user can logout using the logout option provided in the navigation bar.
- Stop: The user returns to the homepage.

3.3 Business Plan

Plans	Free-Trial	Basic	Pro	Premium
Duration	15 days	30 days	180 days	365 days
Price	Free	$5	$25	$40
Benefits	Downloadable Text file Ads	Downloadable Text, PDF and Docx file. No Ads	Downloadable Text, PDF and Docx file. No Ads	Downloadable Text, PDF and Docx file. No Ads

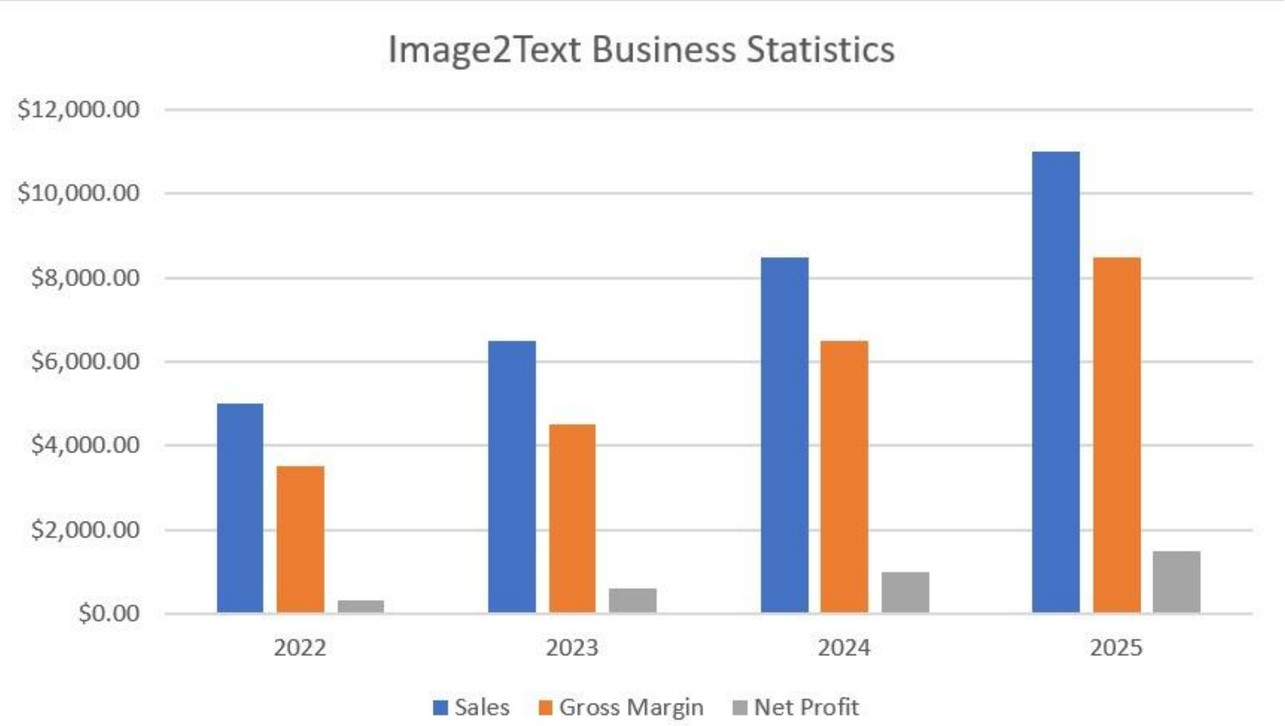


Fig -3.3.1: Business Statistics

‘

Chapter 4

Implementation Details

4.1 System Requirement

4.1.1 Hardware Requirement

The following describes the hardware needed in order to execute and develop the Image2Text application:

- **Computer Desktop or Laptop**
  The computer desktop or a laptop will be utilized to run the visual software in order to upload and download the files. A notebook which is a small, lightweight and inexpensive laptop computer is proposed to increase mobility.

  System will be using:-

Processor	:	intel i3
Main Memory	:	4GB RAM
Hard Disk	:	10 GB free space

4.1.2. Software Requirement

The following describes the software needed in-order to develop the Image2Text application:

- **Frontend:**
  - **Hyper Text Markup Language (HTML):** The Hyper Text Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript. HTML is the code that is used to structure a web page and its content.

  - **Cascading Style Sheets (CSS):** Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is a stylesheet language used to describe the presentation of a document written in HTML. CSS describes how elements should be rendered on screen, on paper, in speech, or on other media.

  - **JavaScript (JS):** JavaScript (JS) is a lightweight, interpreted, or just-in-time compiled programming language with first-class functions. JS is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled and multi-paradigm. It has dynamic typing, prototype-based object-orientation and first-class functions. It allows us to add dynamic behaviour to the webpage and add special effects to the webpage. On websites, it is mainly used for validation purposes. JavaScript helps us to execute complex actions and also enables the interaction of websites with visitors.

- **Database:**
  - **MySQL:** MySQL is an open-source relational database management system. MySQL is a database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL has stand-alone clients that allow users to interact directly with a MySQL database using SQL, but more often, MySQL is used with other programs to implement applications that need relational database capability.

- **Machine Learning:**
  - **Python3:** Python is an interpreter, high-level and general-purpose programming language. Python’s design philosophy emphasizes code readability with its notable use of significant. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. It supports multiple programming paradigms, including structured , object-oriented and functional programming.

  - **Tesseract (PyTesseract):** Tesseract is an OCR engine with support for unicode and the ability to recognize more than 100 languages out of the box. It can be trained to recognize other languages. Tesseract 4 adds a new neural net (LSTM) based OCR engine which is focused on line recognition, but also still supports the legacy Tesseract OCR engine of Tesseract 3 which works by recognizing character patterns. Pytesseract is a wrapper for Tesseract-OCR Engine. It is also useful as a stand-alone invocation script to tesseract, as it can read all image types supported by the Pillow and Leptonica imaging libraries, including jpeg, png etc.

  - **OpenCV:** OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms.

- **Backend:**
  - **Python Flask:** Flask is a micro web framework written in Python. It is classified as a micro-framework because it does not require particular tools or libraries. Flask aims to keep its core functionality small yet typically extensible to cover an array of small and large applications. Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

Software will be using:

OS	:	Window 10 64-bit
Language	:	Python
Tool Used	:	Tesseract, Any python supporting IDE

4.2 Solution Approach / Methodology

The scanned document is usually in the form of an image. The first step is preprocessing which is to convert the image into a format suitable for further processing. The text image may contain noise or it may be skewed. In this step the image is enhanced by noise removal and then converted to binary. The noise present in the image has a major role to play in successful text recognition. Noise removal increases the probability of accurate text recognition and generates more accurate output. Various filters such as Gaussian filter, mean filter can be used for noise removal. Then normalization is done to ensure uniformity which is followed by binarization to convert the gray image into a binary image.

After the preprocessing is done, the individual characters are separated using segmentation process. Then the vital data is retrieved from the raw data using the feature extraction step. Preprocessing consist series of operations and it is used to enhance an image and make it suitable for segmentation. Noise gets introduced during document generation. So Proper filter like mean filter, min-max filter, Gaussian filter etc. may be applied to remove noise from document. Binarization process converts gray scale or colored image to black and white image. To enhance visibility and structural information of character Binary morphological operations like opening, closing, thinning, hole filling etc. may be applied on image. If scanned image is not perfectly aligned, so we need to align it by performing slant angle correction. Input document may be resized if it is too large in size to reduce dimensions to improve speed of processing.

Character segmentation performs an operation of decomposition of an image into Sub images of individual symbols. It is one of the decision processes in a system for optical character recognition (OCR). Its decision that a pattern isolated from the image is that of a character or some other identifiable unit. Generally, document is processed in hierarchical way. At first level lines are segmented using row histogram. From each row, words are extracted using column histogram and finally characters are extracted from words. Accuracy of final result is highly depending on accuracy of segmentation.

Feature extraction is the important part of any pattern recognition application. Feature extraction techniques like Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), Independent Component Analysis (ICA), Chain Code (CC), Scale Invariant Feature Extraction (SIFT), Gradient based features, Histogram might be applied to extract the features of individual characters. These features are used to train the system.

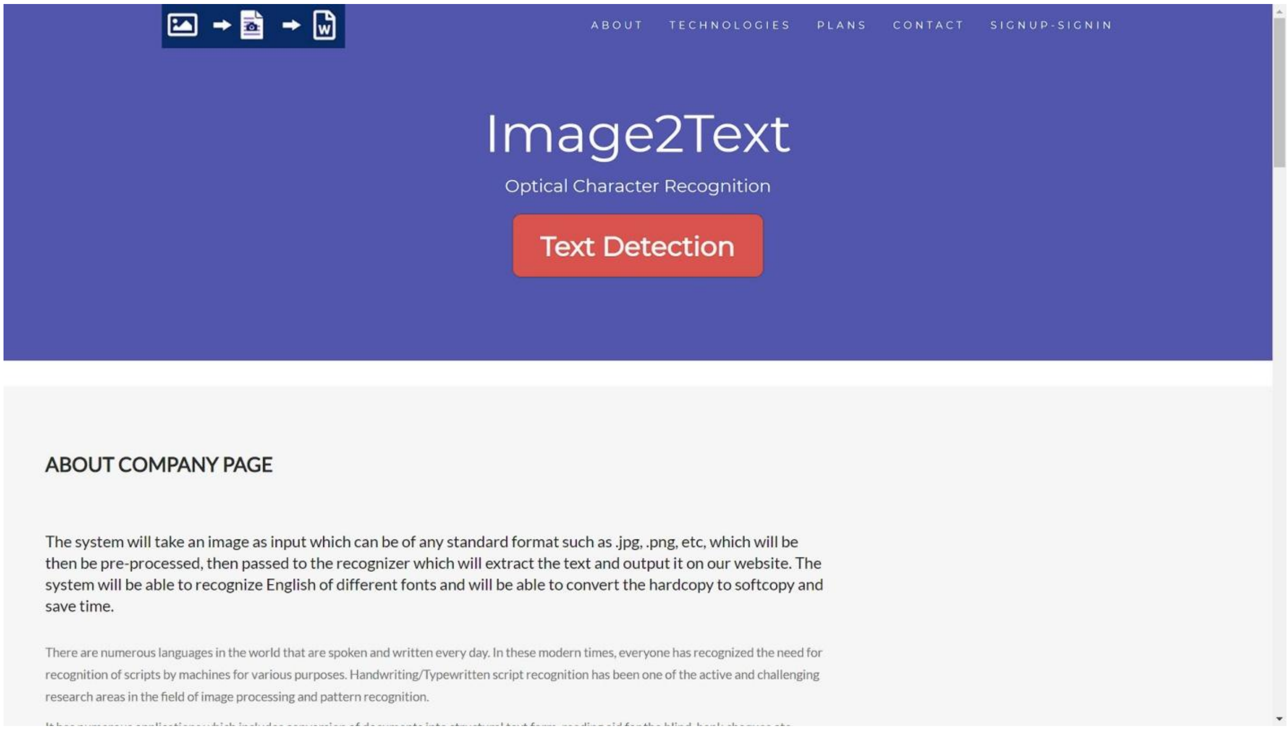
When image is provided as input to OCR system, its features are extracted and given as an input to the trained classifier like artificial neural network or support vector machine. Classifiers compare the input feature with stored pattern and find out the best matching class for input. Post processing it helps to improve the accuracy of recognition. Syntax analysis, semantic analysis kind of higher-level concepts might be applied to check the context of recognized character.

Finally, a preview is given of the text is given and options are displayed for downloading. These may include as a txt file, pdf file or a word file.

Chapter 5

Experimental Results

5.1 GUI (Graphical User Interface)



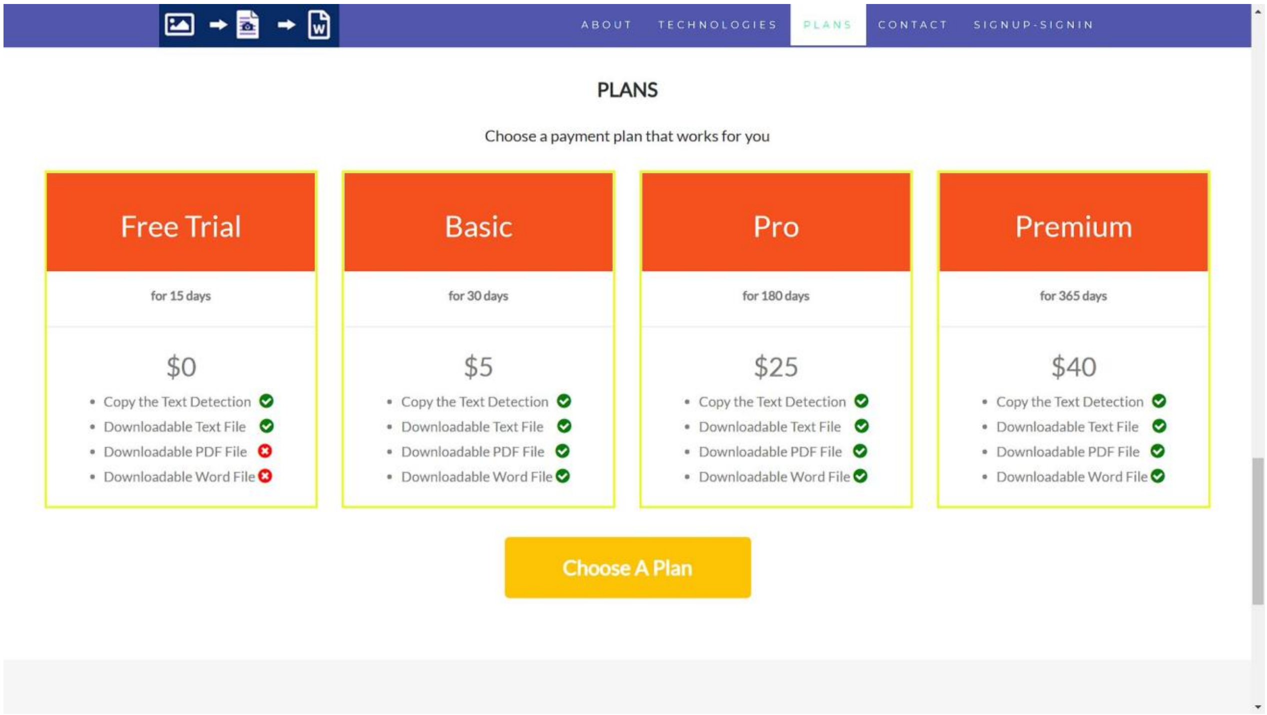
ABOUT COMPANY PAGE

The system will take an image as input which can be of any standard format such as .jpg, .png, etc, which will be then be pre-processed, then passed to the recognizer which will extract the text and output it on our website. The system will be able to recognize English of different fonts and will be able to convert the hardcopy to softcopy and save time.

There are numerous languages in the world that are spoken and written every day. In these modern times, everyone has recognized the need for recognition of scripts by machines for various purposes. Handwriting/Typewritten script recognition has been one of the active and challenging research areas in the field of image processing and pattern recognition.

It has numerous applications which includes conversion of documents into structural text form, reading aid for the blind, bank cheques etc.

Fig -5.1.1: Home Page

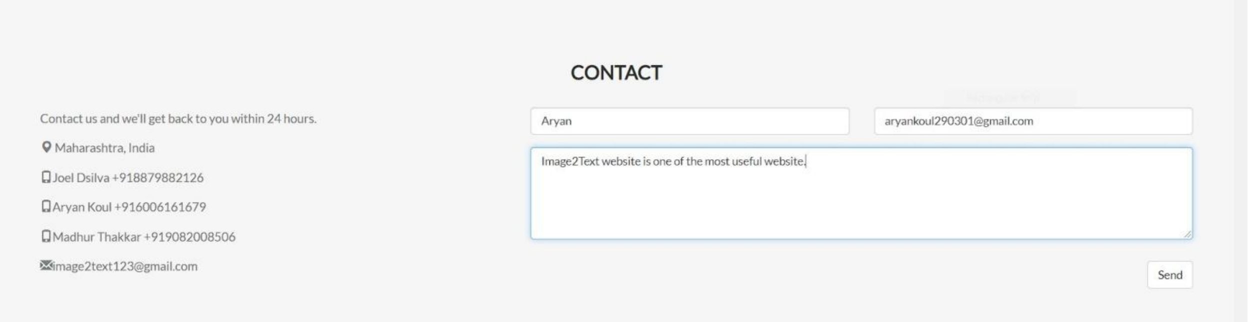


PLANS

Choose a payment plan that works for you

Free Trial	Basic	Pro	Premium
for 15 days	for 30 days	for 180 days	for 365 days
$0	$5	$25	$40
Copy the Text Detection ✓	Copy the Text Detection ✓	Copy the Text Detection ✓	Copy the Text Detection ✓
Downloadable Text File ✓	Downloadable Text File ✓	Downloadable Text File ✓	Downloadable Text File ✓
Downloadable PDF File ✗	Downloadable PDF File ✓	Downloadable PDF File ✓	Downloadable PDF File ✓
Downloadable Word File ✗	Downloadable Word File ✓	Downloadable Word File ✓	Downloadable Word File ✓

Fig -5.1.2: Plans Page



CONTACT

Contact us and we'll get back to you within 24 hours.

Maharashtra, India

Joel Dsilva +918879882126

Aryan Koul +916006161679

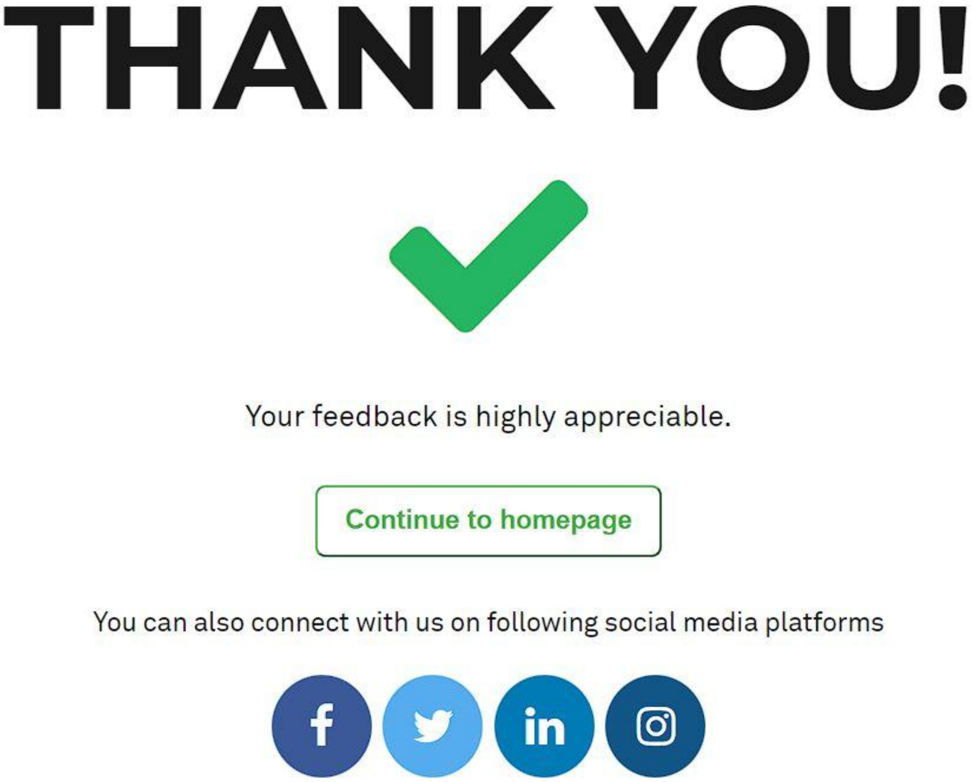
Madhur Thakkar +919082008506

image2text123@gmail.com

Aryan	aryankoul290301@gmail.com

Image2Text website is one of the most useful website.

Fig -5.1.3: Contact Page

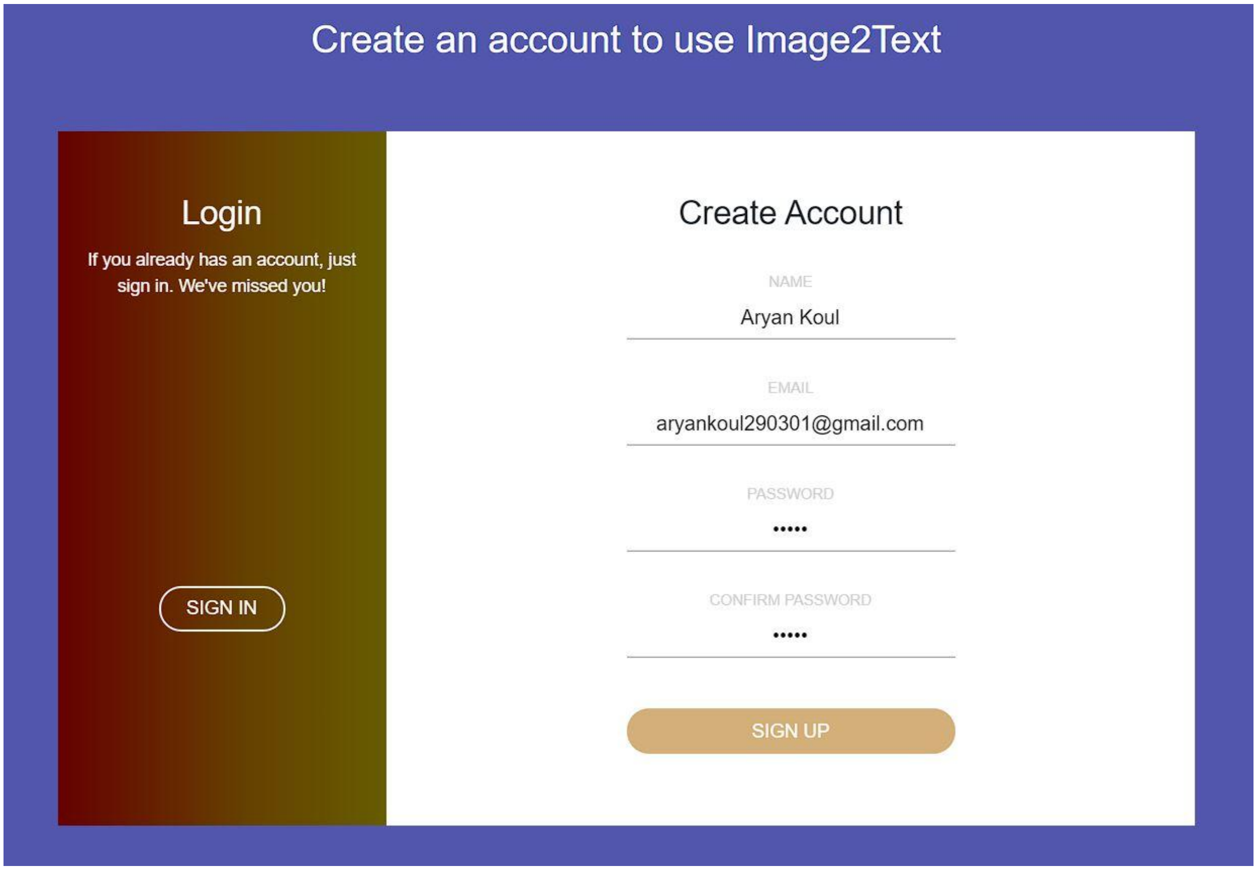


THANK YOU!

Your feedback is highly appreciable.

You can also connect with us on following social media platforms

Fig -5.1.4: Feedback Page



Create an account to use Image2Text

Login

If you already has an account, just sign in. We've missed you!

Create Account

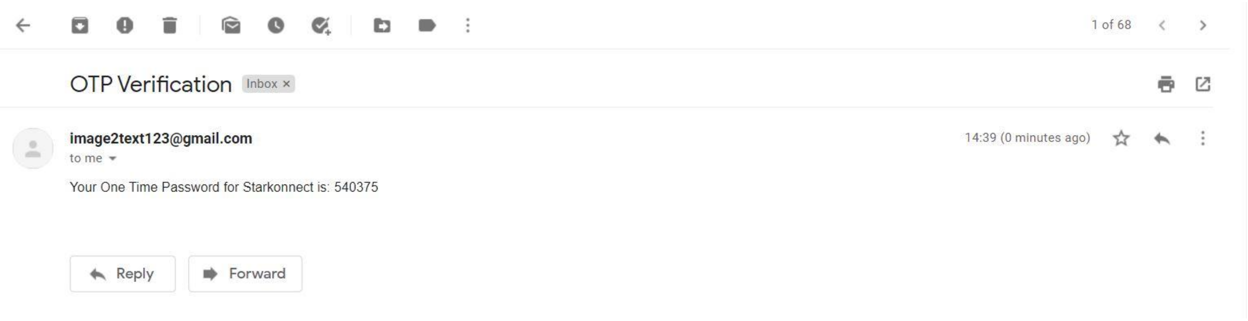
NAME: Aryan Koul

EMAIL: aryankoul290301@gmail.com

PASSWORD: •••••

CONFIRM PASSWORD: •••••

Fig -5.1.5: Sign Up



OTP Verification

image2text123@gmail.com

to me

14:39 (0 minutes ago)

Your One Time Password for Starkonnect is: 540375

Fig -5.1.6: OTP Mail

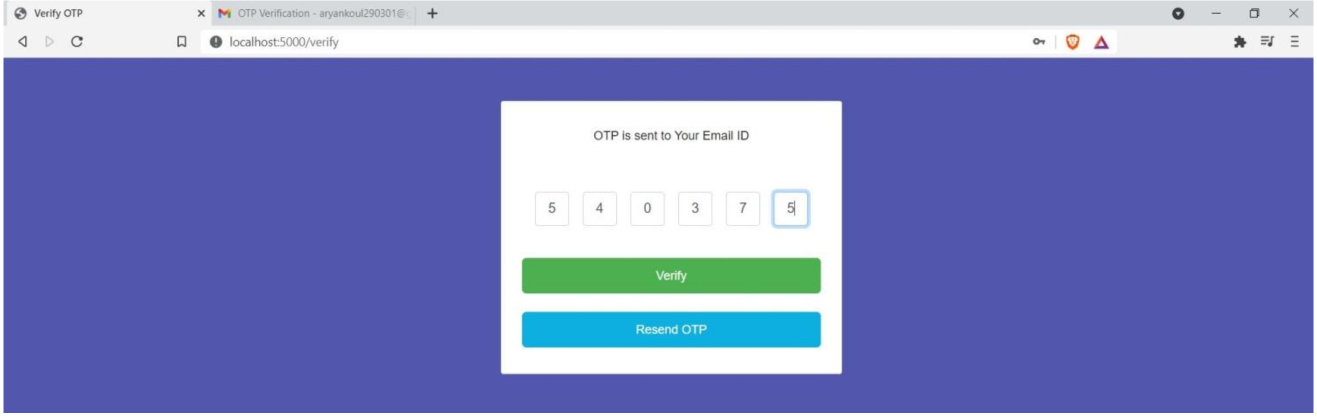

Verify OTP
OTP Verification - aryankoul290301@
localhost:5000/verify
OTP is sent to Your Email ID
5
4
0
3
7
5
Verify
Resend OTP

Fig -5.1.7: OTP Page

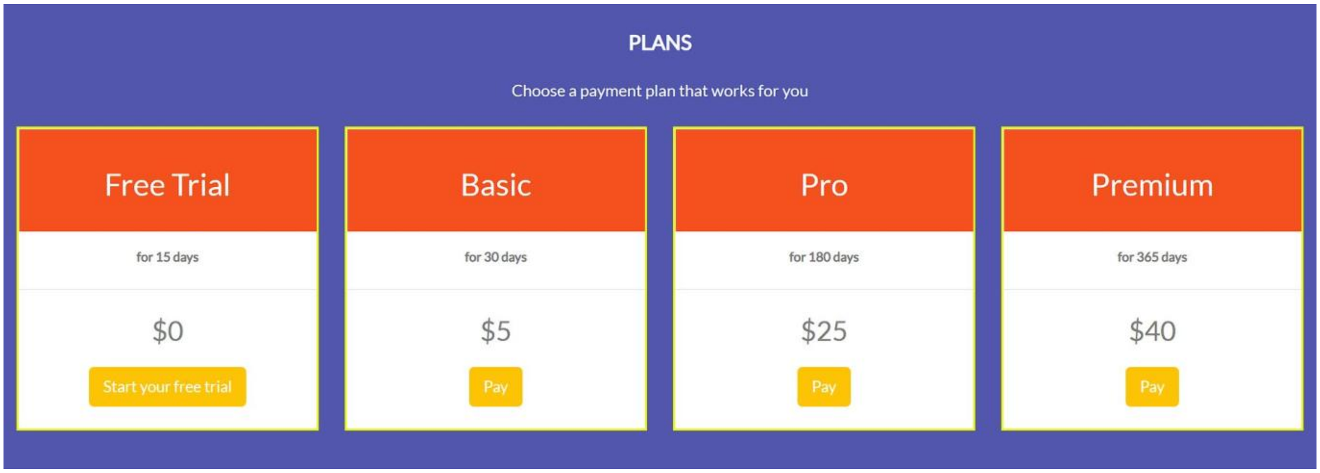
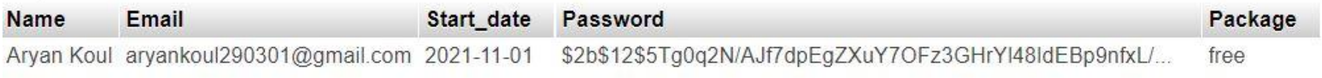


Fig -5.1.8: Choose a Plan



Name	Email	Start_date	Password	Package
Aryan Koul	aryankoul290301@gmail.com	2021-11-01	$2b$12$5Tg0q2N/AJf7dpEgZXuY7OFz3GHrYl48ldEBp9nfxL/...	free

Fig -5.1.9: Password Hashing

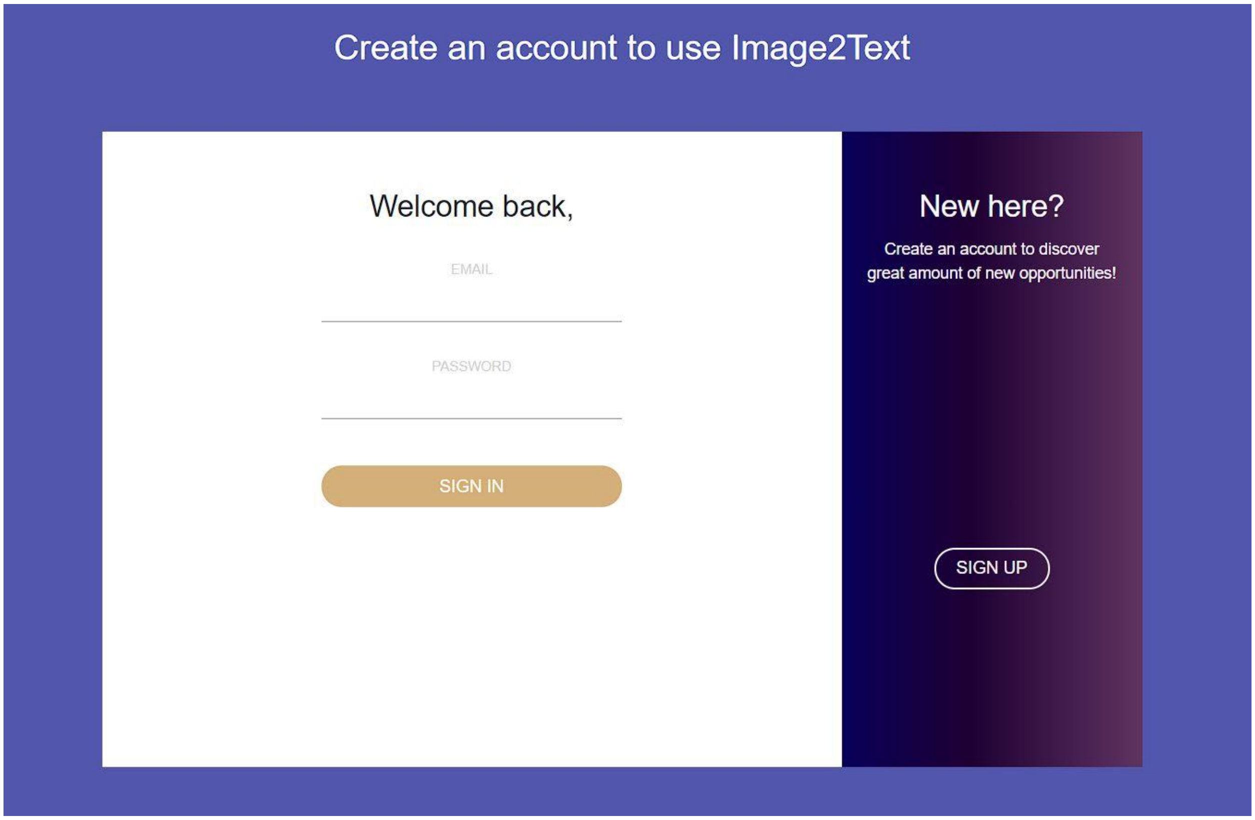
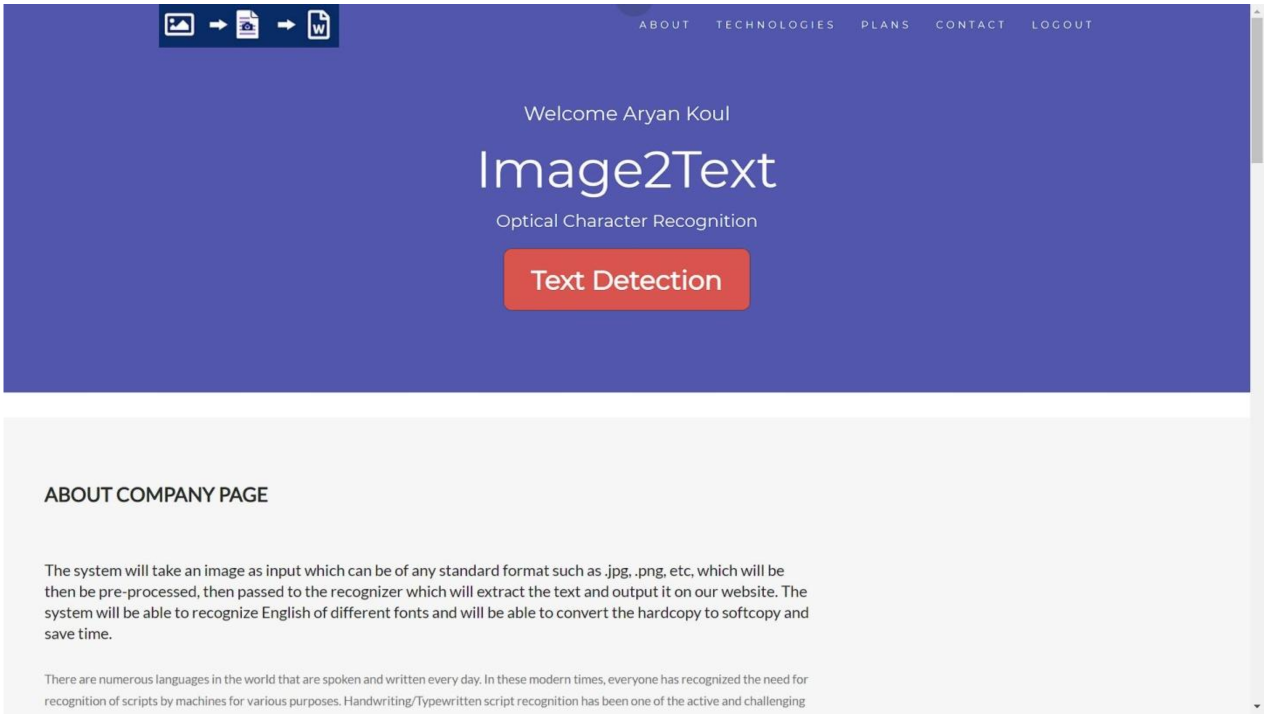

Create an account to use Image2Text
Welcome back,
EMAIL
PASSWORD
SIGN IN
New here?
Create an account to discover great amount of new opportunities!
SIGN UP

Fig -5.1.10: Login Page



ABOUT COMPANY PAGE

The system will take an image as input which can be of any standard format such as .jpg, .png, etc, which will be then be pre-processed, then passed to the recognizer which will extract the text and output it on our website. The system will be able to recognize English of different fonts and will be able to convert the hardcopy to softcopy and save time.

There are numerous languages in the world that are spoken and written every day. In these modern times, everyone has recognized the need for recognition of scripts by machines for various purposes. Handwriting/Typewritten script recognition has been one of the active and challenging

Fig -5.1.11: Home Page After Login

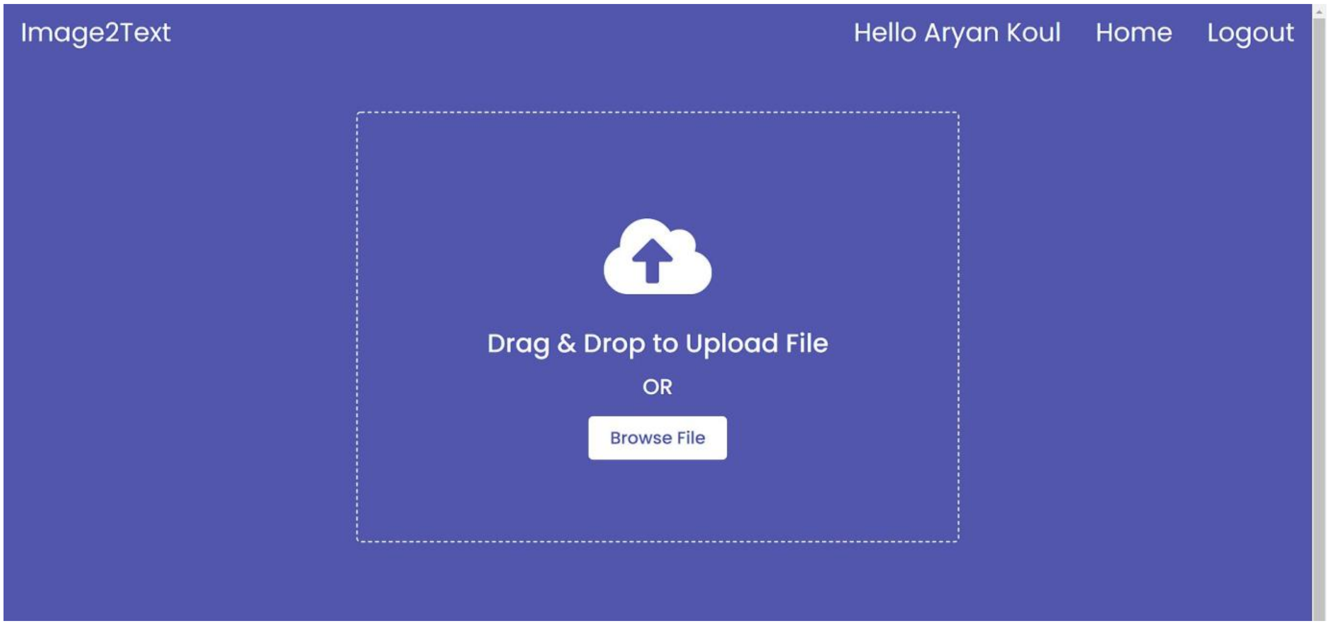
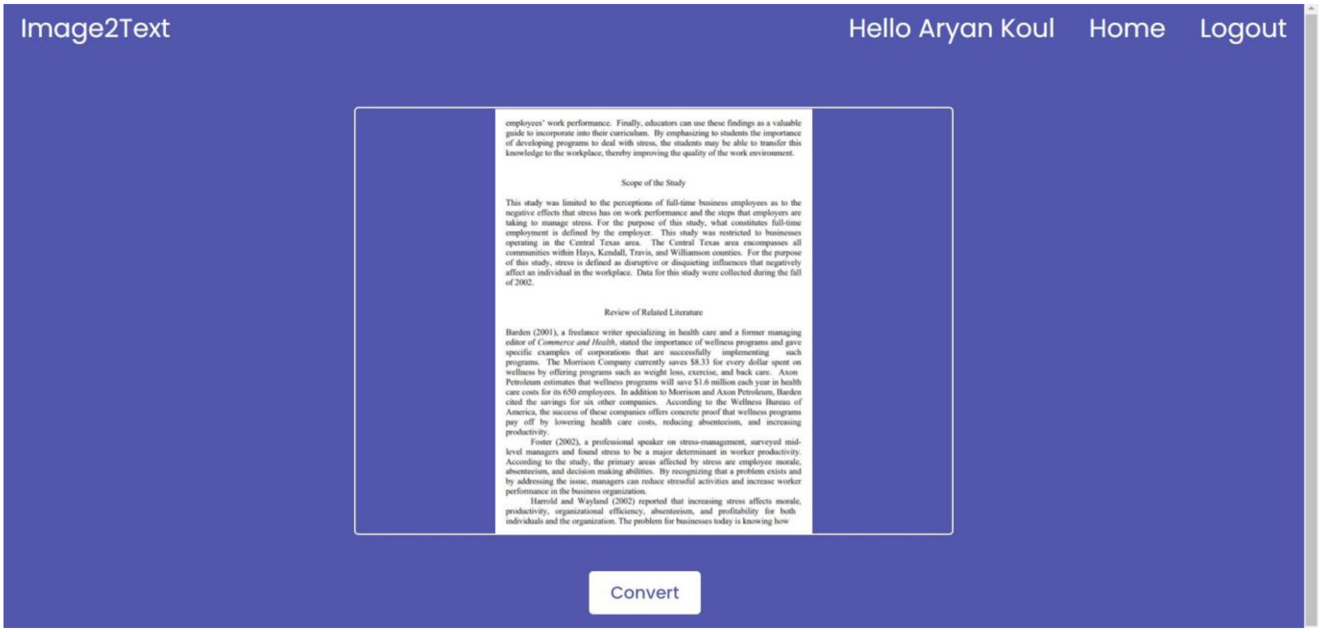


Fig -5.1.12: Upload Page



employees' work performance. Finally, educators can use these findings as a valuable guide to incorporate into their curriculum. By emphasizing to students the importance of developing programs to deal with stress, the students may be able to transfer this knowledge to the workplace, thereby improving the quality of the work environment.

Scope of the Study

This study was limited to the perceptions of full-time business employees as to the negative effects that stress has on work performance and the steps that employers are taking to manage stress. For the purpose of this study, what constitutes full-time employment is defined by the employer. This study was restricted to businesses operating in the Central Texas area. The Central Texas area encompasses all communities within Hays, Kendall, Travis, and Williamson counties. For the purpose of this study, stress is defined as disruptive or disquieting influences that negatively affect an individual in the workplace. Data for this study were collected during the fall of 2002.

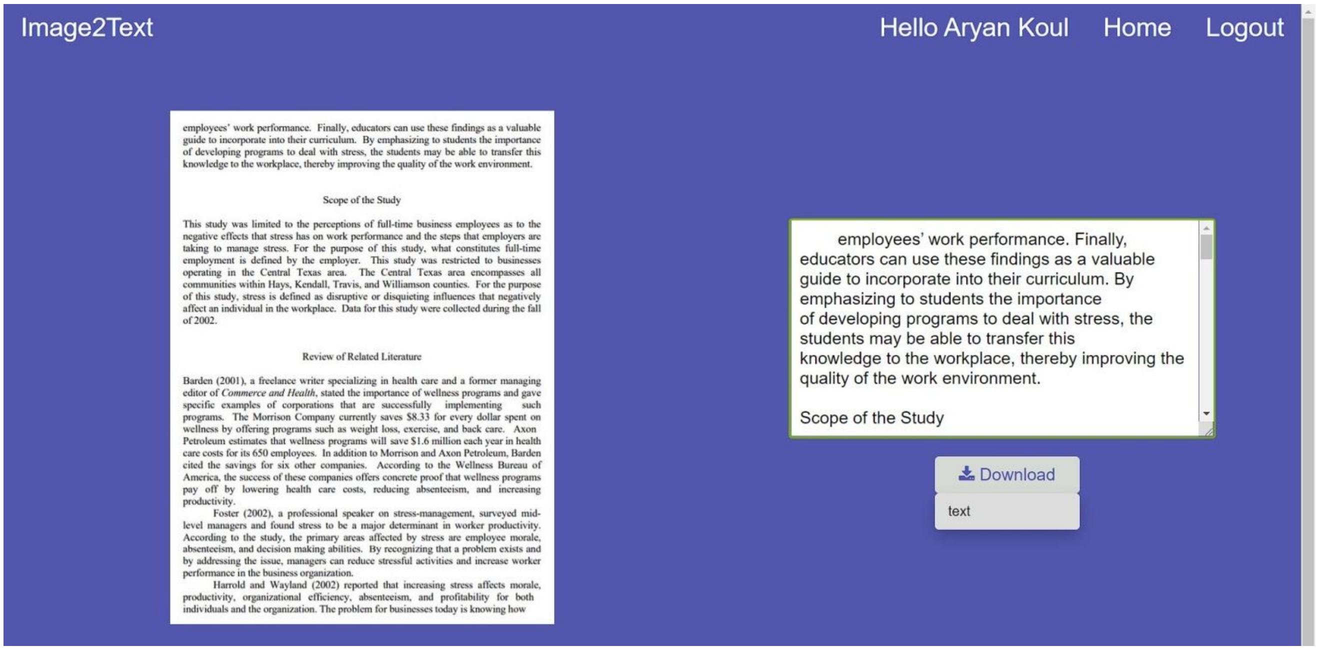
Review of Related Literature

Barden (2001), a freelance writer specializing in health care and a former managing editor of *Commerce and Health*, stated the importance of wellness programs and gave specific examples of corporations that are successfully implementing such programs. The Morrison Company currently saves $8.33 for every dollar spent on wellness by offering programs such as weight loss, exercise, and back care. Axon Petroleum estimates that wellness programs will save $1.6 million each year in health care costs for its 650 employees. In addition to Morrison and Axon Petroleum, Barden cited the savings for six other companies. According to the Wellness Bureau of America, the success of these companies offers concrete proof that wellness programs pay off by lowering health care costs, reducing absenteeism, and increasing productivity.

Foster (2002), a professional speaker on stress-management, surveyed mid-level managers and found stress to be a major determinant in worker productivity. According to the study, the primary areas affected by stress are employee morale, absenteeism, and decision making abilities. By recognizing that a problem exists and by addressing the issue, managers can reduce stressful activities and increase worker performance in the business organization.

Harrold and Wayland (2002) reported that increasing stress affects morale, productivity, organizational efficiency, absenteeism, and profitability for both individuals and the organization. The problem for businesses today is knowing how

Fig -5.1.13: Convert Page



employees' work performance. Finally, educators can use these findings as a valuable guide to incorporate into their curriculum. By emphasizing to students the importance of developing programs to deal with stress, the students may be able to transfer this knowledge to the workplace, thereby improving the quality of the work environment.

Scope of the Study

This study was limited to the perceptions of full-time business employees as to the negative effects that stress has on work performance and the steps that employers are taking to manage stress. For the purpose of this study, what constitutes full-time employment is defined by the employer. This study was restricted to businesses operating in the Central Texas area. The Central Texas area encompasses all communities within Hays, Kendall, Travis, and Williamson counties. For the purpose of this study, stress is defined as disruptive or disquieting influences that negatively affect an individual in the workplace. Data for this study were collected during the fall of 2002.

Review of Related Literature

Barden (2001), a freelance writer specializing in health care and a former managing editor of *Commerce and Health*, stated the importance of wellness programs and gave specific examples of corporations that are successfully implementing such programs. The Morrison Company currently saves $8.33 for every dollar spent on wellness by offering programs such as weight loss, exercise, and back care. Axon Petroleum estimates that wellness programs will save $1.6 million each year in health care costs for its 650 employees. In addition to Morrison and Axon Petroleum, Barden cited the savings for six other companies. According to the Wellness Bureau of America, the success of these companies offers concrete proof that wellness programs pay off by lowering health care costs, reducing absenteeism, and increasing productivity.

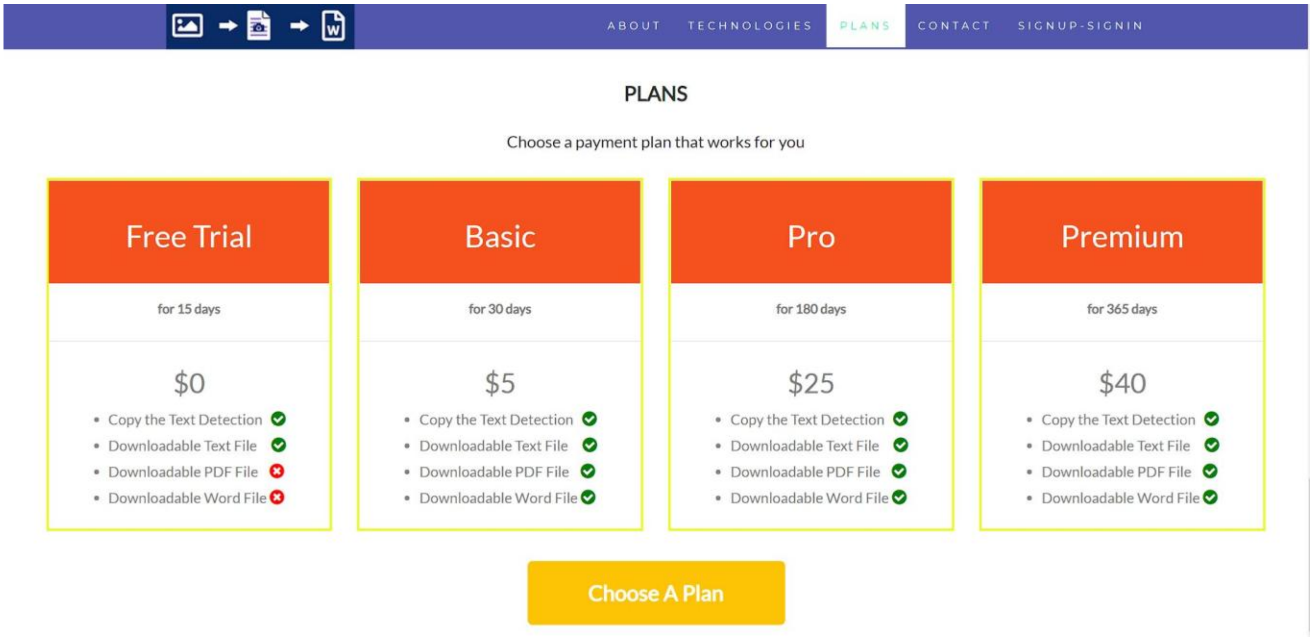
Foster (2002), a professional speaker on stress-management, surveyed mid-level managers and found stress to be a major determinant in worker productivity. According to the study, the primary areas affected by stress are employee morale, absenteeism, and decision making abilities. By recognizing that a problem exists and by addressing the issue, managers can reduce stressful activities and increase worker performance in the business organization.

Harrold and Wayland (2002) reported that increasing stress affects morale, productivity, organizational efficiency, absenteeism, and profitability for both individuals and the organization. The problem for businesses today is knowing how

employees' work performance. Finally,
educators can use these findings as a valuable
guide to incorporate into their curriculum. By
emphasizing to students the importance
of developing programs to deal with stress, the
students may be able to transfer this
knowledge to the workplace, thereby improving the
quality of the work environment.

Scope of the Study

Fig -5.1.14: Download Page (Free Trial)

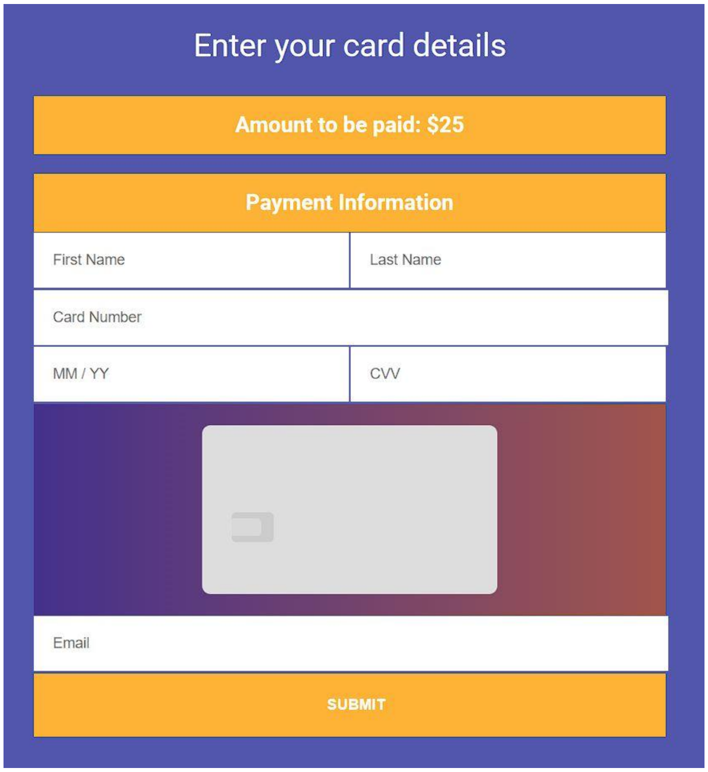


PLANS

Choose a payment plan that works for you

Free Trial	Basic	Pro	Premium
for 15 days	for 30 days	for 180 days	for 365 days
$0	$5	$25	$40
Copy the Text Detection ✓	Copy the Text Detection ✓	Copy the Text Detection ✓	Copy the Text Detection ✓
Downloadable Text File ✓	Downloadable Text File ✓	Downloadable Text File ✓	Downloadable Text File ✓
Downloadable PDF File ✗	Downloadable PDF File ✓	Downloadable PDF File ✓	Downloadable PDF File ✓
Downloadable Word File ✗	Downloadable Word File ✓	Downloadable Word File ✓	Downloadable Word File ✓

Fig -5.1.15: Plan Change Page



Enter your card details

Amount to be paid: $25

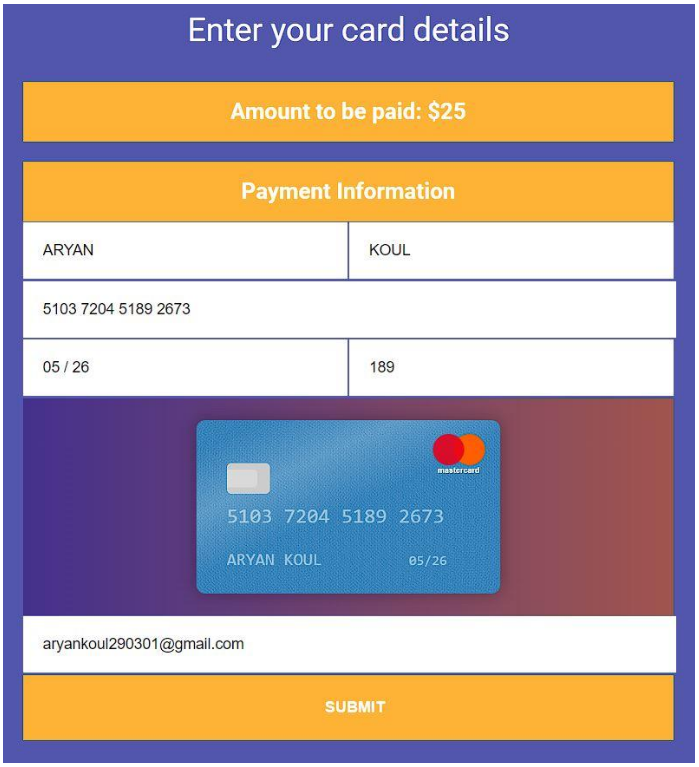
Payment Information

First Name	Last Name
Card Number	
MM / YY	CVV

Email

SUBMIT

Fig -5.1.16: Payment Page



Enter your card details

Amount to be paid: $25

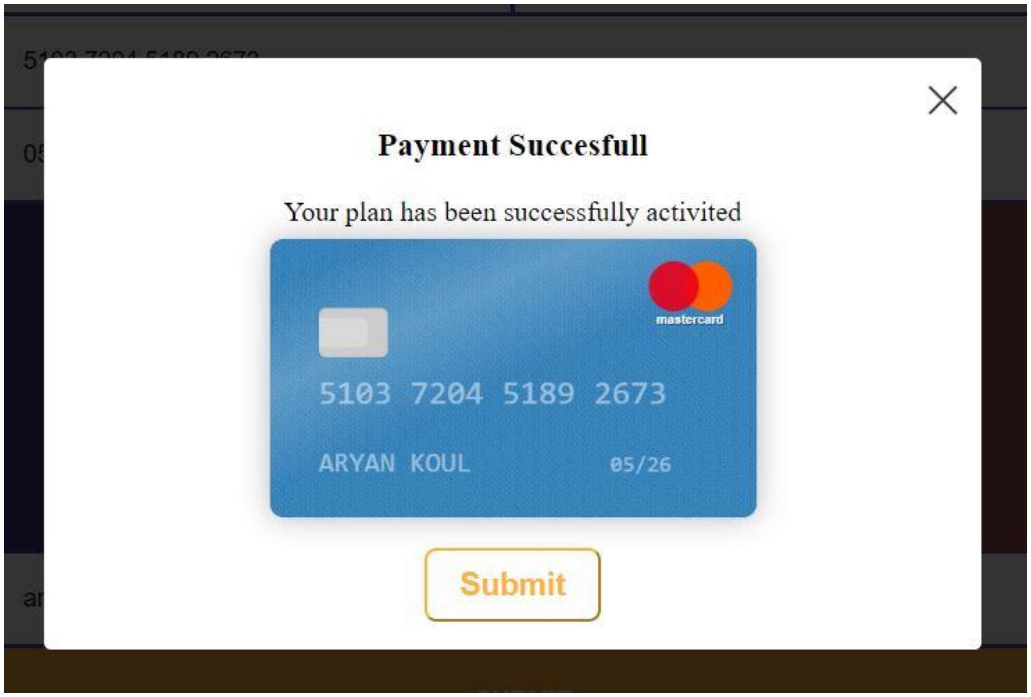
Payment Information

ARYAN	KOUL
5103 7204 5189 2673	
05 / 26	189

aryankoul290301@gmail.com

SUBMIT

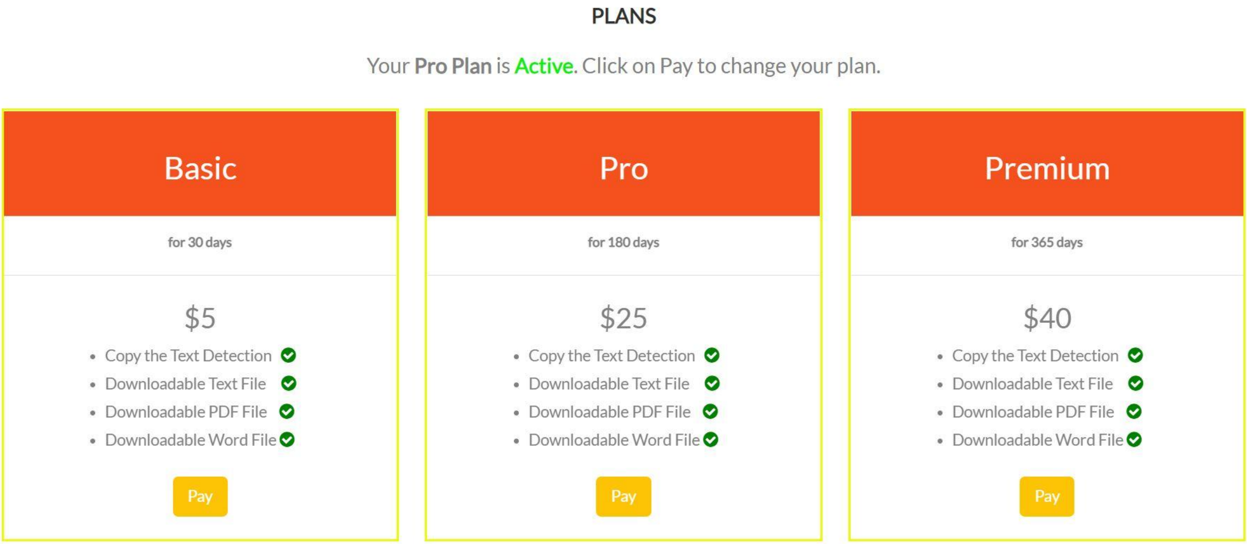
Fig -5.1.17: Payment Page



Payment Succesfull

Your plan has been successfully activited

Fig -5.1.18: Confirmation Page



PLANS

Your **Pro Plan** is **Active**. Click on Pay to change your plan.

Basic

for 30 days

$5

- Copy the Text Detection ✓
- Downloadable Text File ✓
- Downloadable PDF File ✓
- Downloadable Word File ✓

Pay

Pro

for 180 days

$25

- Copy the Text Detection ✓
- Downloadable Text File ✓
- Downloadable PDF File ✓
- Downloadable Word File ✓

Pay

Premium

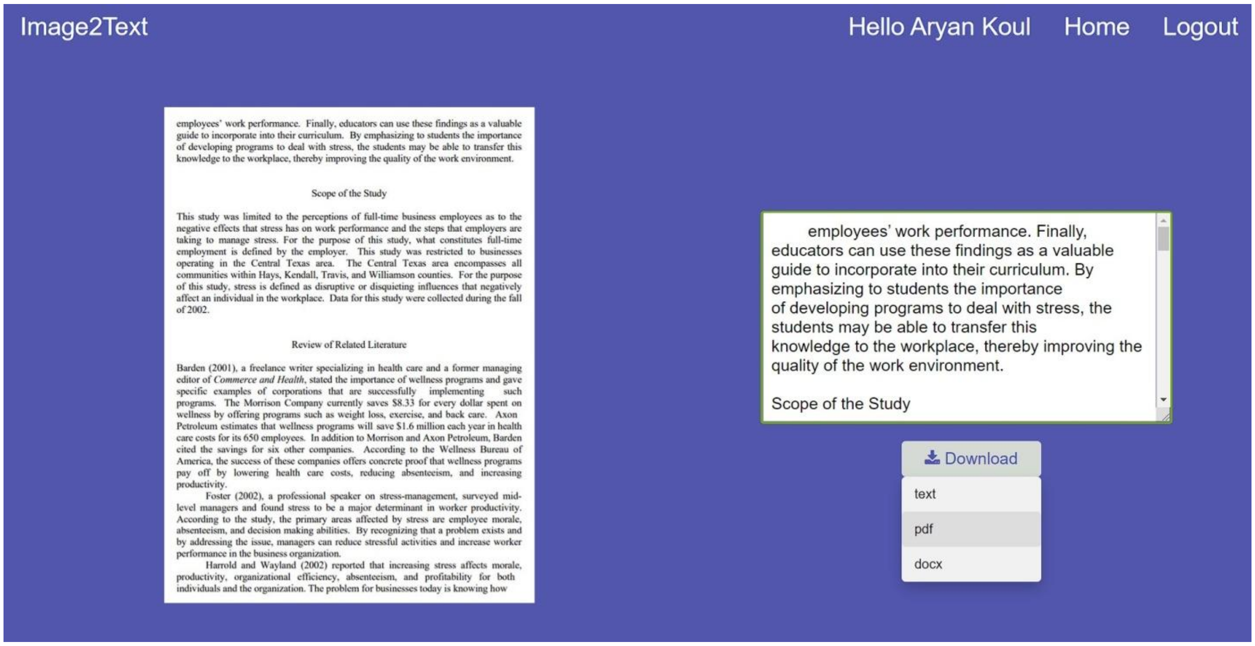
for 365 days

$40

- Copy the Text Detection ✓
- Downloadable Text File ✓
- Downloadable PDF File ✓
- Downloadable Word File ✓

Pay

Fig -5.1.19: Active Plan



employees' work performance. Finally, educators can use these findings as a valuable guide to incorporate into their curriculum. By emphasizing to students the importance of developing programs to deal with stress, the students may be able to transfer this knowledge to the workplace, thereby improving the quality of the work environment.

Scope of the Study

This study was limited to the perceptions of full-time business employees as to the negative effects that stress has on work performance and the steps that employers are taking to manage stress. For the purpose of this study, what constitutes full-time employment is defined by the employer. This study was restricted to businesses operating in the Central Texas area. The Central Texas area encompasses all communities within Hays, Kendall, Travis, and Williamson counties. For the purpose of this study, stress is defined as disruptive or disquieting influences that negatively affect an individual in the workplace. Data for this study were collected during the fall of 2002.

Review of Related Literature

Barden (2001), a freelance writer specializing in health care and a former managing editor of *Commerce and Health*, stated the importance of wellness programs and gave specific examples of corporations that are successfully implementing such programs. The Morrison Company currently saves $8.33 for every dollar spent on wellness by offering programs such as weight loss, exercise, and back care. Axon Petroleum estimates that wellness programs will save $1.6 million each year in health care costs for its 650 employees. In addition to Morrison and Axon Petroleum, Barden cited the savings for six other companies. According to the Wellness Bureau of America, the success of these companies offers concrete proof that wellness programs pay off by lowering health care costs, reducing absenteeism, and increasing productivity.

Foster (2002), a professional speaker on stress-management, surveyed mid-level managers and found stress to be a major determinant in worker productivity. According to the study, the primary areas affected by stress are employee morale, absenteeism, and decision making abilities. By recognizing that a problem exists and by addressing the issue, managers can reduce stressful activities and increase worker performance in the business organization.

Harrold and Wayland (2002) reported that increasing stress affects morale, productivity, organizational efficiency, absenteeism, and profitability for both individuals and the organization. The problem for businesses today is knowing how

employees' work performance. Finally,
educators can use these findings as a valuable
guide to incorporate into their curriculum. By
emphasizing to students the importance
of developing programs to deal with stress, the
students may be able to transfer this
knowledge to the workplace, thereby improving the
quality of the work environment.

Scope of the Study

Fig -5.1.20: Download Page (Paid User)

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

As the world is going online so should the printed media which is mostly available only offline. Our project represents a step or rather a method in storing printed texts in online format and thus preserving it for a longer time and for the future generations. Using this method we can store many valuable texts which till now have been bounded to books only. The recognition of any script is a complicated task. Literature survey and a thorough study of the existing systems was done successfully. From it we learnt about different feature extraction and recognition techniques, alternative strategies that we can apply to approach the solution and noted the pros and cons of different methods. We have also worked out different pre-processing algorithms which can be applied to images for better results. Moreover the detailed study of various English fonts was done which helped us to understand the complexity in the character set of the language. The website for the proposed system has also been implemented and tested successfully.

6.2 Future Scope

- **Adding a unique user profile for every user:** Every account will have a verified profile where, the user can keep a track of their current subscription plan, their personal information and their allotted storage space. All these will further enhance the user experience.

- **Storage of uploaded image and converted text output:** A personal profile will be created for every user, in which user will be allotted storage according to the subscription plan for keeping the uploaded images and the generated output texts saved in the database.

- **Utilizing newer technologies like YOLO:** Using modern technologies such as YOLO (You Only Look Once) will increase the performance of the OCR engine and will also improve its accuracy.

- **Adding a payment gateway:** Using a payment gateway such as PayTM will facilitate different payment methods for users such as NetBanking, UPI, Digital Wallets, etc.

- **Text-to-speech:** Adding a text to speech feature to convert the output text in an audio format to help visually impaired people to hear the detected output text.

Chapter 7

References

REFERENCES

[1] Ray Smith, “An overview of the tesseract OCR engine,” 2005.

[2] Pratik Madhukar Manwatkar, Dr. Kavita R. Singh, “A technical review on text recognition from images,” IEEE Sponsored 9th International Conference on Intelligent Systems and Control (ISCO), 2015.

[3] Akhilesh A. Panchal, Shrugal Varde, M.S. Panse, “Character detection and recognition system for visually impaired people,” IEEE International Conference On Recent Trends In Electronics Information Communication Technology, May 20-21, 2016.

[4] Nada Farhani, Naim Terbeh, Mounir Zrigui, “Image to text conversion: state of the art and extended work,” IEEE/ACS 14th International Conference on Computer Systems and Applications, 2017.

[5] Azmi Can Özgen, Mandana Fasounaki, Hazım Kemal Ekenel, “Text detection in natural and computer generated images,” 2017.

[6] Sandeep Musale, Vikram Ghiye, “Smart reader for visually impaired,” Proceedings of the Second International Conference on Inventive Systems and Control (ICISC 2018) IEEE Xplore Compliant - Part Number:CFP18J06-ART, ISBN:978-1-5386-0807-4; DVD Part Number:CFP18J06DVD, ISBN:978-1-5386-0806-7.

[7] Christian Reul, Uwe Springmann, Christoph Wick, Frank Puppe, “Improving OCR accuracy on early printed books by utilizing cross fold training and voting,” 13th IAPR International Workshop on Document Analysis Systems, 2018.

[8] Prachi Solanki and Malay Bhatt, "Printed Script Recognition OCR using Hopfield Neural Network", International Journal of Computer Applications.

[9] Pranob K Charles , V. Harish M. Swathi , Ch. Deepthi, "A Review on the Various Techniques used for Optical Character Recognition",International Journal of Engineering Research and Applications.

[10] Ana Riza F. Quiros, Alexander Abad, Rhen Anjerome Bedruz, Aaron Christian Uy, Elmer P. Dadios, "A genetic algorithm and artificial neural network-based approach for the machine vision of plate segmentation and character recognition", 2015 International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management(HNICEM).

Chapter 8

Appendix A: Code Sample

- **App.py**

```
from flask import *
import cv2
import pytesseract
import shutil, os
import base64
from flask_mail import Mail, Message
import random
import secrets
from flask_mysqldb import MySQL
from flask_session import Session
from datetime import datetime
from docx import Document
import re
from docx.shared import Pt
from fpdf import FPDF
import bcrypt

app = Flask(__name__)
app.config['MYSQL_HOST'] = 'localhost'
app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = ''
app.config['MYSQL_DB'] = 'newspaper_recog_userdb'

mysql = MySQL(app)
app.secret_key = secrets.token_bytes(16)
app.config['SESSION_PERMANENT'] = False
app.config['SESSION_TYPE'] = "filesystem"
Session(app)
app.config['MAIL_SERVER'] = 'smtp.gmail.com'
app.config['MAIL_PORT'] = 465
app.config['MAIL_USERNAME'] = 'image2text123@gmail.com'
app.config['MAIL_PASSWORD'] = 'Image2Text@123'
app.config['MAIL_USE-TLS'] = False
app.config['MAIL_USE_SSL'] = True
mail = Mail(app)


def createdocx(output):
    document = Document()
    myfile = re.sub(r'[^\x00-\x7F]+|\x0c', ' ', output) # remove all non-XML-compatible
characters
```

```
    para = document.add_paragraph().add_run(myfile)
    para.font.size = Pt(12)
    document.save('.\static\output\output.docx')

def createpdf(output):

    pdf = FPDF()

    pdf.add_page()


    pdf.set_font("Times", size=12)


    f = open(".\static\output\output.txt", "r")

    for x in f:
        x = re.sub(r'[^\x00-\x7F]+|\x0c', ' ', x)
        pdf.cell(200, 10, txt=x, ln=1, align='L')

    pdf.output(".\static\output\output.pdf")


def predict_txt(img_path):

    pytesseract.pytesseract.tesseract_cmd = 'C:\Program Files\Tesseract-
OCR\\tesseract.exe'

    img1 = cv2.imread(img_path)
    img = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)
    # img = cv2.adaptiveThreshold(img1, 255,
cv2.ADAPTIVE_THRESH_GAUSSIAN_C, cv2.THRESH_BINARY, 101, 5)

    output = pytesseract.image_to_string(img)

    file = open(".\static\output\output.txt", "w")
    file.writelines(output)
    file.close()

    createdocx(output)

    createpdf(output)

    return output

@app.route('/')
```

```
def index():
    session.clear()
    session['login_status'] = False
    return render_template("index.html", status=session.get('login_status'))

@app.route('/login_index')
def login_index():
    if session.get('login_status'):
        return render_template("index.html",
status=session.get('login_status'),user=session.get('name'), plan = session.get('pack'))
    else:
        return render_template("index.html", status=session.get('login_status'))

@app.route('/login/<dest>')
def login(dest):
    session['login_dest'] = dest
    return render_template("login.html")

@app.route('/verify', methods=['GET','POST'])
def verify():
    if request.method == "POST":
        if request.form['name']!="" and request.form['email']!="" and
request.form['password']!="" and request.form['con_password']!="":
            if request.form['password']==request.form['con_password']:
                ver_email = request.form['email']
                session['email'] = ver_email

                reg = None
                try:
                    cur = mysql.connection.cursor()
                    cur.execute("SELECT * FROM users WHERE email =
'"+session.get('email')+"'")
                    reg = cur.fetchone()
                    print(reg,session.get('email'))
                    mysql.connection.commit()
                    cur.close()
                except:
                    print(reg,session.get('email'))
                if reg is None:
                    otp_gen = random.randint(100000,999999)
                    session['otp'] = otp_gen
                    print("verify")
                    print(session.get('otp'))
                    msg="Your One Time Password for Starkonnect is: "+str(session.get('otp'))
                    message = Message("OTP
Verification",sender="email",recipients=[session.get('email')])
```

```
                    message.body = msg

                    mail.send(message)

                    success = "Message Sent"

                    session['name'] = request.form.get('name')
                    session['hashed'] = bcrypt.hashpw(request.form['password'].encode('utf-8'),
bcrypt.gensalt())
                    print("hash",session.get('hashed'))
                    return render_template("otp.html",success=success)
                else:
                    flash('This Email is already registered')
                    return render_template("login.html")
            else:
                flash('Password and Confirm Password Do not Match!')
                return render_template("login.html")
        else:
            flash('Please enter all the details properly!')
            return render_template("login.html")

@app.route('/authenticate', methods=['GET','POST'])
def authenticate():
    if request.method == "POST":
        otp_get = request.form.getlist('otp')
        otp_enter = ""
        for ele in otp_get:
            otp_enter+=str(ele)
        if len(otp_enter)<1:
            otp_enter = -1
        if int(otp_enter)==session.get('otp'):
            return render_template("pack.html")
        else:
            flash(u"Incorrect OTP\nTry Again!")
            return render_template("otp.html")

@app.route('/payment/<plan>', methods=['GET','POST'])
def payment(plan):
    if request.method == "POST":
        if plan=="basic":
            session['pack'] = "basic"
            return render_template("payment.html",amount="5")
        elif plan=="pro":
            session['pack'] = "pro"
            return render_template("payment.html",amount="25")
        elif plan=="premium":
            session['pack'] = "premium"
            return render_template("payment.html",amount="40")
        elif plan=="free":
            session['pack'] = "free"
            reg = None
            try:
                cur = mysql.connection.cursor()
                cur.execute("SELECT * FROM users WHERE email = 
'"+session.get('email')+"'")
                reg = cur.fetchone()
                print(reg)
                mysql.connection.commit()
                cur.close()
            except:
                print(reg,session.get('email'))
            if reg is None:
                cur = mysql.connection.cursor()
                print(str(session.get('hashed')))
                cur.execute("INSERT INTO users 
VALUES('"+session.get('name')+"','"+session.get('email')+"','"+str(datetime.now())+"','"
+str(session.get('hashed'))[2:-1]+"','"+session.get('pack')+"')")
                mysql.connection.commit()
                cur.close()
            else:
                cur = mysql.connection.cursor()
                cur.execute("UPDATE users SET Start_date='"+str(datetime.now())+"', 
Package='"+session.get('pack')+"' WHERE Email='"+session.get('email')+"'")
                mysql.connection.commit()
                cur.close()
            session['login_status'] = True
            if session.get('login_dest')=="index" or session.get('login_dest')=="pack":
                return 
render_template("index.html",status=session.get('login_status'),user=session.get('name'),
plan = session.get('pack'))
            elif session.get('login_dest')=="upload":
                return render_template("upload.html",user=session.get('name'))

@app.route('/contact',methods=['POST','GET'])
def contact():
    if request.method=="POST":
        name = request.form['feed_name']
        email = request.form['feed_email']
        comment = request.form['comments']
        cur = mysql.connection.cursor()
        cur.execute("INSERT INTO feedback
VALUES('"+name+"','"+email+"','"+comment+"')")
        mysql.connection.commit()
        cur.close()
        return render_template("feedback.html")

@app.route('/signup_index',methods=['GET','POST'])
def signup_index():
    if request.method == "POST":
        reg = None
        try:
            cur = mysql.connection.cursor()
            cur.execute("SELECT * FROM users WHERE email =
'"+session.get('email')+"'")
            reg = cur.fetchone()
            print(reg)
            mysql.connection.commit()
            cur.close()
        except:
            print(reg,session.get('email'))
        if reg is None:
            cur = mysql.connection.cursor()
            cur.execute("INSERT INTO users
VALUES('"+session.get('name')+"','"+session.get('email')+"','"+str(datetime.now())+"','"
+str(session.get('hashed'))[2:-1]+"','"+session.get('pack')+"')")
            mysql.connection.commit()
            cur.close()
        else:
            cur = mysql.connection.cursor()
            cur.execute("UPDATE users SET Start_date='"+str(datetime.now())+"',
Package='"+session.get('pack')+"' WHERE Email='"+session.get('email')+"'")
            mysql.connection.commit()
            cur.close()
        session['login_status'] = True
        if session.get('login_dest')=="index" or session.get('login_dest')=="pack":
            return
render_template("index.html",status=session.get('login_status'),user=session.get('name'),
plan = session.get('pack'))
        elif session.get('login_dest')=="upload":
            return render_template("upload.html",user=session.get('name'))

@app.route('/resend_verify', methods=['GET','POST'])
def resend_verify():
    if request.method == "POST":
        #email = request.form['email']
        otp_gen = random.randint(100000,999999)
        session['otp'] = otp_gen
        print("resend")
        print(session.get('otp'))
        msg="Your One Time Password for Starkonnect is: "+str(session.get('otp'))
        message = Message("OTP
Verification",sender='image2text123@gmail.com',recipients=[session.get('email')])

        message.body = msg

        mail.send(message)

        success = "Message Sent"

        flash(u"New OTP is sent to {0}".format(session.get('email')))
        return render_template("otp.html")

# @app.route('/selection',methods=['GET','POST'])
# def selection():
#     if request.method=='POST':
#         chkbox = request.form.getlist('chkbox')
#         session['profile'] = chkbox[0]
#         cur = mysql.connection.cursor()
#         cur.execute("INSERT INTO users
VALUES('"+session.get('name')+"','"+session.get('email')+"','"+str(datetime.now())+"','"
+session.get('password')+"','"+session.get('profile')+"')")
#         mysql.connection.commit()
#         cur.close()
#         if session.get('profile')=="Free":
#             return
render_template("index.html",user_name=session.get('name'),prof=session.get('profile'))
#         elif session.get('profile')=="Paid":
#             return
render_template("index.html",user_name=session.get('name'),prof=session.get('profile'))

@app.route('/userlogin',methods=['GET','POST'])
def userlogin():
    if request.method=='POST':
        if request.form['loginemail'] != "" and request.form['loginpass'] != "":
            email = request.form['loginemail']
            password = request.form['loginpass']
            reg = None
            try:
                cur = mysql.connection.cursor()
                cur.execute("SELECT * FROM users WHERE email = '"+email+"'")
                reg = cur.fetchone()
                session['hashed'] = reg[3].encode("utf-8")
                session['email'] = reg[1]
                session['name'] = reg[0]
                session['pack'] = reg[4]
                session['Start_date'] = reg[2]
                mysql.connection.commit()
                cur.close()
            except:
                pass
            if reg is None:
                flash('This Email is not registered')
                return render_template("login.html")
            else:
                if bcrypt.checkpw(password.encode('utf-8'), session.get('hashed')):
                    pack = session.get('pack')
                    print(pack)
                    today_date = str(datetime.now())[:10]
                    date_format = "%Y-%m-%d"
                    a = datetime.strptime(today_date, date_format)
                    b = datetime.strptime(session.get('Start_date'), date_format)
                    delta = a-b

                    if pack=="free":
                        if delta.days>15:
                            flash('Your Free Trial has Ended')
                            return render_template("pack2.html")
                        else:
                            session['login_status'] = True
                            if session.get('login_dest')=="index":
                                return
render_template("index.html",status=session.get('login_status'),user=session.get('name'),
plan = session.get('pack'))
                            elif session.get('login_dest')=="upload":
                                return render_template("upload.html",user=session.get('name'))
                            elif session.get('login_dest')=="pack":
                                flash("Your Free Trial is active")
                                return render_template("pack2.html")

                    elif pack=="basic":
                        if delta.days>30:
                            flash('Your Plan has Ended')
                            return render_template("pack2.html")
                        else:
                            session['login_status'] = True
                            if session.get('login_dest')=="index":
                                return
render_template("index.html",status=session.get('login_status'),user=session.get('name'),
plan = session.get('pack'))
```

```
                            elif session.get('login_dest')=="upload":
                                return render_template("upload.html",user=session.get('name'))
                            elif session.get('login_dest')=="pack":
                                flash("Your Basic Plan is active")
                                return render_template("pack2.html")

                    elif pack=="pro":
                        if delta.days>180:
                            flash('Your Plan has Ended')
                            return render_template("pack2.html")
                        else:
                            session['login_status'] = True
                            if session.get('login_dest')=="index":
                                return
render_template("index.html",status=session.get('login_status'),user=session.get('name'),
plan = session.get('pack'))
                            elif session.get('login_dest')=="upload":
                                return render_template("upload.html",user=session.get('name'))
                            elif session.get('login_dest')=="pack":
                                flash("Your Pro Plan is active")
                                return render_template("pack2.html")

                    elif pack=="premium":
                        if delta.days>365:
                            flash('Your Plan has Ended')
                            return render_template("pack2.html")
                        else:
                            session['login_status'] = True
                            if session.get('login_dest')=="index":
                                return
render_template("index.html",status=session.get('login_status'),user=session.get('name'),
plan = session.get('pack'))
                            elif session.get('login_dest')=="upload":
                                return render_template("upload.html",user=session.get('name'))
                            elif session.get('login_dest')=="pack":
                                flash("Your Premium Plan is active")
                                return render_template("pack2.html")
                else:
                    flash('Incorrect Password!')
                    return render_template("login.html")
        else:
            flash('Please enter all the details properly!')
            return render_template("login.html")

@app.route('/upload',methods=['GET','POST'])
def upload():
    if request.method=="POST":
        return render_template("upload.html",user=session.get('name'))

@app.route('/convert', methods=['GET', 'POST'])
def convert():
    if request.method == 'POST':

        file = request.files['file']
        file1 = request.form['base64data']

        if file:
            # print("path" + os.path.join(os.path.curdir, secure_filename(file.filename)))
            shutil.rmtree('.\static\images1')
            os.makedirs('.\static\images1')

            img_path = os.path.join('.\static\images1', "input_image.png")

            file.save(img_path)

            output_text = predict_txt(img_path)
            return render_template("download.html", img=str(img_path),
op_text=str(output_text), user=session.get('name'), pack=session.get('pack'))

        elif file1:
            file1 += '=' * (-len(file1) % 4)
            with open(".\static\images1\input_image.png", "wb") as fh:
                fh.write(base64.decodebytes(file1.split(',')[1].encode()))
            img_path = ".\static\images1\input_image.png"

            output_text = predict_txt(img_path)

            return render_template("download.html", img=str(img_path),
op_text=str(output_text), user=session.get('name'), pack=session.get('pack'))

        else:
            return "No Image File"

if __name__ == '__main__':
    app.run()
```

- **Index.html**

```
<!DOCTYPE html>
<html lang="en">

<head>
  <title>Image2Text</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <link href="https://fonts.googleapis.com/css?family=Montserrat" rel="stylesheet"
type="text/css">
  <link href="https://fonts.googleapis.com/css?family=Lato" rel="stylesheet"
type="text/css">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
  <style>
    body {
      font: 400 15px Lato, sans-serif;
      line-height: 1.8;
      color: #818181;
    }

    h2 {
      font-size: 24px;
      text-transform: uppercase;
      color: #303030;
      font-weight: 600;
      margin-bottom: 30px;
    }

    h4 {
      font-size: 19px;
      line-height: 1.375em;
      color: #303030;
      font-weight: 400;
      margin-bottom: 30px;
    }

    .jumbotron {
      background-color: #5256ad;
    color: #fff;
    padding: 100px 25px;
    font-family: Montserrat, sans-serif;
  }

  .container-fluid {
    padding: 60px 50px;
  }

  .bg-grey {
    background-color: #f6f6f6;
  }

  .logo-small {
    color: #69d80e;
    font-size: 50px;
  }

  .logo {
    color: #e60965;
    font-size: 200px;
  }

  .thumbnail {
    padding: 0 0 15px 0;
    border: none;
    border-radius: 0;
  }

  /* .thumbnail:before {
 content: ' ';
 display: block;
 position: absolute;
 left: 0;
 top: 0;
 width: 100%;
 height: 100%;
 background-image: url('OCR2.png');
 background-repeat: no-repeat;
 background-size: cover;
}*/

  .thumbnail img {
    width: 100%;
    height: 100%;
    margin-bottom: 10px;
```

```
}

.carousel-control.right,
.carousel-control.left {
   background-image: none;
   color: #f108bf;
}

.carousel-indicators li {
   border-color: #f106ea;
}

.carousel-indicators li.active {
   background-color: #ec0aa8;
}

.item h4 {
   font-size: 20px;
   line-height: 1.375em;
   font-weight: 400;
   font-style: italic;
   margin: 70px 0;
}

.item span {
   font-style: normal;
}

.panel {
   border: 3px solid #ebfc04;
   border-radius: 0 !important;
   transition: box-shadow 0.5s;
}

.panel:hover {
   box-shadow: 5px 0px 40px rgba(0, 0, 0, .2);
}

.panel-footer .btn:hover {
   border: 1px solid #06ebf3;
   background-color: #fff !important;
   color: #1af507;
}

.panel-heading {
   color: #fff !important;
   background-color: #f4511e !important;
   padding: 25px;
   border-bottom: 1px solid transparent;
   border-top-left-radius: 0px;
   border-top-right-radius: 0px;
   border-bottom-left-radius: 0px;
   border-bottom-right-radius: 0px;
}

.panel-footer {
   background-color: white !important;
}

.panel-footer h3 {
   font-size: 32px;
}

.panel-footer h4 {
   color: #aaa;
   font-size: 14px;
}

.panel-footer .btn {
   margin: 15px 0;
   background-color: #fdc305;
   color: #fff;
}

.btn56 {
   margin: 15px 0;
   background-color: #fdc305;
   color: #fff;
   margin-left: 575px;
   padding: 20px;
   width: 300px;
   height: max-content;
   font-size: 25px;
   font-weight: 900;
}

.btn56:hover {
   border: 1px solid #06ebf3;
   background-color: #fff !important;
   color: #1af507;
}
```

```
.navbar {
   margin-bottom: 0;
   background-color: #5256ad;
   z-index: 9999;
   border: 0;
   font-size: 12px !important;
   line-height: 1.42857143 !important;
   letter-spacing: 4px;
   border-radius: 0;
   font-family: Montserrat, sans-serif;
}

.navbar li a,
.navbar .navbar-brand {
   color: #fff !important;
}

.navbar-nav li a:hover,
.navbar-nav li.active a {
   color: #06f096 !important;
   background-color: #fff !important;
}

.navbar-default .navbar-toggle {
   border-color: transparent;
   color: #fff !important;
}

footer .glyphicon {
   font-size: 20px;
   margin-bottom: 20px;
   color: #f41e1e;
}

.slideanim {
   visibility: hidden;
}

.slide {
   animation-name: slide;
   -webkit-animation-name: slide;
   animation-duration: 1s;
   -webkit-animation-duration: 1s;
   visibility: visible;
}
```

```
@keyframes slide {
    0% {
        opacity: 0;
        transform: translateY(70%);
    }

    100% {
        opacity: 1;
        transform: translateY(0%);
    }
}

@-webkit-keyframes slide {
    0% {
        opacity: 0;
        -webkit-transform: translateY(70%);
    }

    100% {
        opacity: 1;
        -webkit-transform: translateY(0%);
    }
}

@media screen and (max-width: 768px) {
    .col-sm-4 {
        text-align: center;
        margin: 25px 0;
    }

    .btn-lg {
        width: 100%;
        margin-bottom: 35px;
    }
}

@media screen and (max-width: 480px) {
    .logo {
        font-size: 150px;
    }
}

.text-detection {
    width: 300px;
    height: 75px;
    font-size: xx-large;
```

```
            font-weight: bold;
        }

        #myCarousel2 {
            margin-left: 50px;
            margin-right: 50px;
        }

        .tech-left {
            margin-left: -150px;
        }

        .tech-right {
            margin-right: -150px;
        }
    </style>
</head>

<body id="myPage" data-spy="scroll" data-target=".navbar" data-offset="60">

    <nav class="navbar navbar-default navbar-fixed-top">
        <div class="container">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                <img class="collapse navbar-collapse" src="../static/images/logo.jpg"></img>
            </div>
            <div class="collapse navbar-collapse" id="myNavbar">
                <ul class="nav navbar-nav navbar-right">
                    <li><a href="#about">ABOUT</a></li>
                    <li><a href="#portfolio">TECHNOLOGIES</a></li>
                    <li><a href="#pricing">PLANS</a></li>
                    <li><a href="#contact">CONTACT</a></li>
                    {% if status==False %}
                    <li><a href="http://localhost:5000/login/index">SIGNUP-SIGNIN</a></li>
                    {% else %}
                    <li><a href="http://localhost:5000/">LOGOUT</a></li>
                    {% endif %}
                </ul>
            </div>
        </div>
    </nav>
    <div class="jumbotron text-center">
        {% if status==True %}
        <h3>Welcome {{ user }}</h3>
        {% endif %}
        <h1>Image2Text</h1>
        <p>Optical Character Recognition</p>
        <form id="text_det_form">
            <div class="input-group">
                <!-- <input type="email" class="form-control" size="50" placeholder="Email
Address" required> -->
                <div class="input-group-btn">
                    {% if status==False %}
                    <button class="btn btn-danger text-detection" style="border-radius: 10px;"

onclick="document.getElementById('text_det_form').action='http://localhost:5000/login/
upload'; document.getElementById('text_det_form').method='GET'">Text
                        Detection</button>
                    {% else %}
                    <button class="btn btn-danger text-detection" style="border-radius: 10px;"

onclick="document.getElementById('text_det_form').action='http://localhost:5000/uploa
d'; document.getElementById('text_det_form').method='POST'">Text
                        Detection</button>
                    {% endif %}
                </div>
            </div>
        </form>
    </div>

    <!-- Container (About Section) -->
    <div id="about" class="container-fluid bg-grey">
        <div class="row">
            <div class="col-sm-8">
                <h2>About Company Page</h2><br>
                <h4>The system will take an image as input which can be of any standard
format such as .jpg, .png, etc,
                    which will be then be pre-processed, then passed to the recognizer which
will extract the text and
                    output
                    it on our website. The system will be able to recognize English of different
fonts and will be able
                    to convert
                    the hardcopy to softcopy and save time.</h4>

                <p>There are numerous languages in the world that are spoken and written
```

```
every day. In these modern
                    times, everyone has recognized the need for recognition of scripts by
machines for various purposes.
                    Handwriting/Typewritten script recognition has been one of the active and
challenging research areas
                    in the field of image processing and pattern recognition.</p>

                <p>It has numerous applications which includes conversion of documents into
structural text form,
                    reading aid for the blind, bank cheques etc. Script recognition can also play
an important role in
                    preserving
                    the culture, famous works and literature from ancient times. With the amount
of variation in the
                    handwriting of
                    people and the curves involved, processing becomes a challenge. For this
reason, it is very
                    important to have a
                    script recognition system that can be used irrespective of the document
type.</p>

                <br><button class="btn btn-default btn-lg">Like</button>
            </div>
            <div class="col-sm-4">
                <span class="glyphicon glyphicon-signal logo slideanim"></span>
            </div>
        </div>
    </div>

    <div class="container-fluid">
        <div class="row">
            <div class="col-sm-4">
                <span class="glyphicon glyphicon-globe logo slideanim"></span>
            </div>
            <div class="col-sm-8">
                <h2>Our Values</h2><br>
                <h4><strong>MISSION:</strong> Our mission aims at developing an online
English Script Recognition system
                    which can be effectively used for recognizing various English fonts and
convert them to text file.
                </h4>
                <br>
                <p><strong>VISION:</strong>As the world is going online so should the
printed media which is mostly
                    available only offline. Our project represents a step or rather a method in
storing printed texts in
```

```
                    online format and thus preserving it for a longer time and for the future
generations. Using this
                    method we
                    can store many valuable texts which till now have been bounded to books
only.</p>
            </div>
        </div>
    </div>

    <!-- Container (Portfolio Section) -->
    <div id="portfolio" class="container-fluid text-center bg-grey">
        <h2>TECHNOLOGIES</h2>
        <h4>What technologies we have used</h4>
        <div id="myCarousel2" class="carousel slide text-center" data-ride="carousel"
            style="width: 700px; margin-left: auto; margin-right: auto;color: black">
            <!-- Indicators -->
            <ol class="carousel-indicators">
                <li data-target="#myCarousel2" data-slide-to="0" class="active"></li>
                <li data-target="#myCarousel2" data-slide-to="1"></li>
                <li data-target="#myCarousel2" data-slide-to="2"></li>
            </ol>

            <!-- Wrapper for slides -->
            <div class="carousel-inner" role="listbox">
                <div class="row-sm-4 item active" style="width: 700px; height: 400px;
margin: auto;color: black">
                    <div class="thumbnail"
                        style="background-image: url('../static/images/new-OCR.jpg'); width:
100%; height: 100%; background-repeat: no-repeat; background-size: cover">
                        <p style="font-size: 25px"><strong><br>OCR</strong></p>

                        <div class="text-left slideanim">
                            <ul style="font-weight: bold">
                                <li>OCR stands for "Optical Character Recognition." </li>
                                <br>
                                <li>It is a technology that recognizes
                                    text within a digital image. It is commonly used to recognize text
in scanned
                                    documents and images.</li>
                                <br>
                                <li>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.
                                </li>
                                <br>
                                <li>Whereas, 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 inthe
                                    image.</li>
                            </ul>
                        </div>
                    </div>
                </div>
                <div class="row-sm-4 item" style="width: 700px; height: 400px; margin:
auto;color: black">
                    <div class="thumbnail"
                        style="background-image:url('../static/images/new-OCR.jpg'); width:
100%; height: 100%; background-repeat: no-repeat; background-size: cover">
                        <p style="font-size: 25px"><strong><br>Tesseract</strong></p>
                        <div class="text-left slideanim">
                            <ul style="font-weight: bold">
                                <li>Tesseract is an open source text recognition (OCR) Engine,
available under the
                                    Apache
                                    2.0 license.</li>
                                <br>
                                <li>The current official release is 4.1.1.</li>
                                <br>
                                <li>The master branch on Github can be used by those who want the
latest code for LSTM
                                    (–oem
                                    1) and legacy (–oem 0) Tesseract. The master branch is using 5.0.0
versioning
                                    because
                                    code modernization caused API compatibility issues with 4.x
release.</li>
                                <br>
                                <li>Tesseract can be used directly via command line, or (for
programmers) by using an
                                    API to
                                    extract printed text from images.</li>
                                <br>
```

```
                  <li>Tesseract 4.0 added a new OCR engine based on LSTM neural
networks.</li>
                </ul>
              </div>
            </div>
          </div>
          <div class="row-sm-4 item" style="width: 700px; height: 400px; margin:
auto;color: black">
            <div class="thumbnail"
              style="background-image: url('../static/images/new-OCR.jpg'); width:
100%; height: 100%; background-repeat: no-repeat; background-size: cover">
              <!-- <img src="sanfran.jpg" alt="San Francisco" width="400"
height="300"> -->
              <p style="font-size: 25px"><strong><br>Flask</strong></p>
              <div class="text-left slideanim">
                <ul style="font-weight: bold">
                  <li>Flask is a micro web framework written in Python.</li>
                  <br>
                  <li>It is classified as a microframework because it does not require
particular tools or
                    libraries.</li>
                  <br>
                  <li>It has no database abstraction layer, form validation, or any other
components where
                    pre-existing third-party libraries provide common functions.</li>
                  <br>
                  <li>However, Flask supports extensions that can add application
features as if they were
                    implemented in Flask itself.</li>
                  <br>
                  <li>Extensions exist for object-relational mappers, form validation,
upload handling,
                    various open authentication technologies and several common
framework related tools.
                  </li>
                </ul>
              </div>
            </div>
          </div>
        </div>

        <!-- Left and right controls -->
        <a class="left carousel-control tech-left" href="#myCarousel2" role="button"
data-slide="prev">
          <span class="glyphicon glyphicon-chevron-left" aria-hidden="true"></span>
          <span class="sr-only">Previous</span>
```

```
      </a>
      <a class="right carousel-control tech-right" href="#myCarousel2" role="button" data-slide="next">
        <span class="glyphicon glyphicon-chevron-right" aria-hidden="true"></span>
        <span class="sr-only">Next</span>
      </a>
    </div>
    <br><br><br><br><br>
    <h2>What are the libraries we used</h2>
    <div id="myCarousel" class="carousel slide text-center" data-ride="carousel">
      <!-- Indicators -->
      <ol class="carousel-indicators" style="margin-top:100px">
        <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
        <li data-target="#myCarousel" data-slide-to="1"></li>
        <li data-target="#myCarousel" data-slide-to="2"></li>
        <li data-target="#myCarousel" data-slide-to="3"></li>
      </ol>

      <!-- Wrapper for slides -->
      <div class="carousel-inner" role="listbox">
        <div class="item active">
          <h4>Frontend<br><br><span>HTML, CSS, JS</span></h4>
        </div>
        <div class="item">
          <h4>Machine Learning<br><br><span>Python3, Pytesseract, OpenCV</span></h4>
        </div>
        <div class="item">
          <h4>Database<br><br><span>MySQL</span></h4>
        </div>
        <div class="item">
          <h4>Backend<br><br><span>Flask</span></h4>
        </div>
      </div>

      <!-- Left and right controls -->
      <a class="left carousel-control" href="#myCarousel" role="button" data-slide="prev">
        <span class="glyphicon glyphicon-chevron-left" aria-hidden="true"></span>
        <span class="sr-only">Previous</span>
      </a>
      <a class="right carousel-control" href="#myCarousel" role="button" data-slide="next">
        <span class="glyphicon glyphicon-chevron-right" aria-hidden="true"></span>
        <span class="sr-only">Next</span>
      </a>
```

```
        </div>
    </div>

    <!-- Container (Pricing Section) -->
    {% if status==False %}
    <div id="pricing" class="container-fluid">
        <div class="text-center">
            <h2>PLANS</h2>
            <h4>Choose a payment plan that works for you</h4>
        </div>
        <div class="row slideanim" \>
            <div class="col-sm-3 col-xs-12">
                <div class="panel panel-default text-center">
                    <div class="panel-heading">
                        <h1>Free Trial</h1>
                    </div>
                    <div class="panel-body">
                        <p><strong>for 15 days</strong></p>
                    </div>
                    <div class="panel-footer">
                        <h3>$0</h3>
                        <ul style="text-align: left; margin-left:14px; font-size: 17px">
                            <li>Copy the Text Detection &nbsp;&nbsp;<i class='fa fa-check-circle'
                                    style='font-size:20px;color:green'></i></li>
                            <li>Downloadable Text File &nbsp;&nbsp;&nbsp;&nbsp;<i class='fa
fa-check-circle'
                                    style='font-size:20px;color:green'></i></li>
                            <li>Downloadable PDF File &nbsp;&nbsp; <i class='fa fa-times-circle'
                                    style='font-size:20px;color:red'></i></li>
                            <li>Downloadable Word File <i class='fa fa-times-circle'
                                    style='font-size:20px;color:red'></i></li>
                        </ul>
                        <!-- <a href='#'>
                            <button class="btn btn-lg">Start your free trial</button></a> -->
                    </div>
                </div>
            </div>
            <div class="col-sm-3 col-xs-12">
                <div class="panel panel-default text-center">
                    <div class="panel-heading">
                        <h1>Basic</h1>
                    </div>
                    <div class="panel-body">
                        <p><strong>for 30 days</strong></p>
                    </div>
                    <div class="panel-footer">
            <h3>$5</h3>
            <ul style="text-align: left; margin-left:14px; font-size: 17px">
              <li>Copy the Text Detection &nbsp;&nbsp;<i class='fa fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
              <li>Downloadable Text File &nbsp;&nbsp;&nbsp;&nbsp;<i class='fa
fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
              <li>Downloadable PDF File &nbsp;&nbsp; <i class='fa fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
              <li>Downloadable Word File <i class='fa fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
            </ul>
            <!-- <a href='#'>
              <button class="btn btn-lg">Start your free trial</button></a> -->
          </div>
        </div>
      </div>
      <div class="col-sm-3 col-xs-12">
        <div class="panel panel-default text-center">
          <div class="panel-heading">
            <h1>Pro</h1>
          </div>
          <div class="panel-body">
            <p><strong>for 180 days</strong></p>
          </div>
          <div class="panel-footer">
            <h3>$25</h3>
            <ul style="text-align: left; margin-left:14px; font-size: 17px">
              <li>Copy the Text Detection &nbsp;&nbsp;<i class='fa fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
              <li>Downloadable Text File &nbsp;&nbsp;&nbsp;&nbsp;<i class='fa
fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
              <li>Downloadable PDF File &nbsp;&nbsp; <i class='fa fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
              <li>Downloadable Word File <i class='fa fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
            </ul>
            <!-- <a href='#'>
              <button class="btn btn-lg">Start your free trial</button></a> -->
          </div>
        </div>
      </div>
      <div class="col-sm-3 col-xs-12">
        <div class="panel panel-default text-center">
          <div class="panel-heading">
```

```
            <h1>Premium</h1>
          </div>
          <div class="panel-body">
            <p><strong>for 365 days</strong></p>
          </div>
          <div class="panel-footer">
            <h3>$40</h3>
            <ul style="text-align: left; margin-left:14px; font-size: 17px">
              <li>Copy the Text Detection &nbsp;&nbsp;<i class='fa fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
              <li>Downloadable Text File &nbsp;&nbsp;&nbsp;&nbsp;<i class='fa
fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
              <li>Downloadable PDF File &nbsp;&nbsp; <i class='fa fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
              <li>Downloadable Word File <i class='fa fa-check-circle'
                  style='font-size:20px;color:green'></i></li>
            </ul>
            <!-- <a href='#'>
              <button class="btn btn-lg">Start your free trial</button></a> -->
          </div>
        </div>
      </div>
      <a href="http://localhost:5000/login/pack">
        <button class="btn btn56 btn-lg">Choose A Plan</button>
      </a>
    </div>
  </div>
  {% else %}
  <div id="pricing" class="container-fluid">
    <div class="text-center">
      <h2>PLANS</h2>
      {% if plan=="free" %}
      <h3>Your <strong>Free Trial</strong> is <span style="color: rgb(13, 240,
13);"><strong>Active</strong></span>. Click on Pay to change your plan.</h3>
      {% elif plan=="basic" %}
      <h3>Your <strong> Basic Plan</strong> is <span style="color: rgb(13, 240,
13);"><strong>Active</strong></span>. Click on Pay to change your plan.</h3>
      {% elif plan=="pro" %}
      <h3>Your <strong>Pro Plan</strong> is <span style="color: rgb(13, 240,
13);"><strong>Active</strong></span>. Click on Pay to change your plan.</h3>
      {% elif plan=="premium" %}
      <h3>Your <strong>Premium Plan</strong> is <span style="color: rgb(13, 240,
13);"><strong>Active</strong></span>. Click on Pay to change your plan.</h3>
      {% endif %}
      <br>
```

```
        </div>
        <form class="row slideanim" method="POST" id="pay_form">
            <div class="col-sm-4 col-xs-12">
                <div class="panel panel-default text-center">
                    <div class="panel-heading">
                        <h1>Basic</h1>
                    </div>
                    <div class="panel-body">
                        <p><strong>for 30 days</strong></p>
                    </div>
                    <div class="panel-footer">
                        <h3>$5</h3>
                        <ul style="text-align: left; margin-left:60px; font-size: 18px">
                            <li>Copy the Text Detection &nbsp;&nbsp;<i class='fa fa-check-circle'
                                    style='font-size:20px;color:green'></i></li>
                            <li>Downloadable Text File &nbsp;&nbsp;&nbsp;&nbsp;<i class='fa
fa-check-circle'
                                    style='font-size:20px;color:green'></i></li>
                            <li>Downloadable PDF File &nbsp;&nbsp; <i class='fa fa-check-circle'
                                    style='font-size:20px;color:green'></i></li>
                            <li>Downloadable Word File <i class='fa fa-check-circle'
                                    style='font-size:20px;color:green'></i></li>
                        </ul>
                        <a href='#'>
                            <button class="btn btn-lg"
                                onclick="document.getElementById('pay_form').action =
'http://localhost:5000/payment/basic'">Pay</button></a>
                    </div>
                </div>
            </div>
            <div class="col-sm-4 col-xs-12">
                <div class="panel panel-default text-center">
                    <div class="panel-heading">
                        <h1>Pro</h1>
                    </div>
                    <div class="panel-body">
                        <p><strong>for 180 days</strong></p>
                    </div>
                    <div class="panel-footer">
                        <h3>$25</h3>
                        <ul style="text-align: left; margin-left:60px; font-size: 18px">
                            <li>Copy the Text Detection &nbsp;&nbsp;<i class='fa fa-check-circle'
                                    style='font-size:20px;color:green'></i></li>
                            <li>Downloadable Text File &nbsp;&nbsp;&nbsp;&nbsp;<i class='fa
fa-check-circle'
                                    style='font-size:20px;color:green'></i></li>
                <li>Downloadable PDF File &nbsp;&nbsp; <i class='fa fa-check-circle'
                    style='font-size:20px;color:green'></i></li>
                <li>Downloadable Word File <i class='fa fa-check-circle'
                    style='font-size:20px;color:green'></i></li>
              </ul>
              <a href='#'>
                <button class="btn btn-lg"
                  onclick="document.getElementById('pay_form').action =
'http://localhost:5000/payment/pro'">Pay</button></a>
            </div>
          </div>
        </div>
        <div class="col-sm-4 col-xs-12">
          <div class="panel panel-default text-center">
            <div class="panel-heading">
              <h1>Premium</h1>
            </div>
            <div class="panel-body">
              <p><strong>for 365 days</strong></p>
            </div>
            <div class="panel-footer">
              <h3>$40</h3>
              <ul style="text-align: left; margin-left:60px; font-size: 18px">
                <li>Copy the Text Detection &nbsp;&nbsp;<i class='fa fa-check-circle'
                    style='font-size:20px;color:green'></i></li>
                <li>Downloadable Text File &nbsp;&nbsp;&nbsp;&nbsp;<i class='fa
fa-check-circle'
                    style='font-size:20px;color:green'></i></li>
                <li>Downloadable PDF File &nbsp;&nbsp; <i class='fa fa-check-circle'
                    style='font-size:20px;color:green'></i></li>
                <li>Downloadable Word File <i class='fa fa-check-circle'
                    style='font-size:20px;color:green'></i></li>
              </ul>
              <a href='#'>
                <button class="btn btn-lg"
                  onclick="document.getElementById('pay_form').action =
'http://localhost:5000/payment/premium'">Pay</button></a>
            </div>
          </div>
        </div>
        <!-- <a href="http://localhost:5000/login/index">
          <button class="btn btn-lg">Choose A Plan</button>
        </a> -->
      </form>
    </div>
    {% endif %}
    <!-- Container (Contact Section) -->
    <div id="contact" class="container-fluid bg-grey">
        <h2 class="text-center">CONTACT</h2>
        <div class="row">
            <div class="col-sm-5">
                <p>Contact us and we'll get back to you within 24 hours.</p>
                <p><span class="glyphicon glyphicon-map-marker"></span> Maharashtra,
India</p>
                <p><span class="glyphicon glyphicon-phone"></span>Joel Dsilva
+918879882126</p>
                <p><span class="glyphicon glyphicon-phone"></span>Aryan Koul
+916006161679</p>
                <p><span class="glyphicon glyphicon-phone"></span>Madhur Thakkar
+919082008506</p>
                <p><span class="glyphicon glyphicon-
envelope"></span>image2text123@gmail.com</p>
            </div>
            <form class="col-sm-7 slideanim" action="http://localhost:5000/contact"
method="POST">
                <div class="row">
                    <div class="col-sm-6 form-group">
                        <input class="form-control" id="name" name="feed_name"
placeholder="Name" type="text" required
                            important>
                    </div>
                    <div class="col-sm-6 form-group">
                        <input class="form-control" id="email" name="feed_email"
placeholder="Email" type="email"
                            required important>
                    </div>
                </div>
                <textarea class="form-control" id="comments" name="comments"
placeholder="Comment" rows="5" required
                    important></textarea><br>
                <div class="row">
                    <div class="col-sm-12 form-group">
                        <button class="btn btn-default pull-right" type="submit">Send</button>
                    </div>
                </div>
            </form>
        </div>
    </div>
    <!-- Image of location/map -->
    <!-- <img src="/w3images/map.jpg" class="w3-image w3-greyscale-min"
style="width:100%"> -->
    <footer class="container-fluid text-center">
    <a href="#myPage" title="To Top">
      <span class="glyphicon glyphicon-chevron-up"></span>
    </a>
    <p>Made By Agnelites
      <!-- <a href="https://www.google.com" title="Visit
Google">www.google.com</a> -->
    </p>
  </footer>
  <script>
    $(document).ready(function () {
      // Add smooth scrolling to all links in navbar + footer link
      $(".navbar a, footer a[href='#myPage']").on('click', function (event) {
        // Make sure this.hash has a value before overriding default behavior
        if (this.hash !== "") {
          // Prevent default anchor click behavior
          event.preventDefault();

          // Store hash
          var hash = this.hash;
          // Using jQuery's animate() method to add smooth page scroll
          // The optional number (900) specifies the number of milliseconds it takes to
scroll to the specified area
          $('html, body').animate({
            scrollTop: $(hash).offset().top
          }, 900, function () {

            // Add hash (#) to URL when done scrolling (default click behavior)
            window.location.hash = hash;
          });
        } // End if
      });
      $(window).scroll(function () {
        $(".slideanim").each(function () {
          var pos = $(this).offset().top;

          var winTop = $(window).scrollTop();
          if (pos < winTop + 600) {
            $(this).addClass("slide");
          }
        });
      });
    })
  </script>

</body>
</html>
```

Chapter 9

Acknowledgements

ACKNOWLEDGEMENTS

The satisfaction that accompanies the successful completion of this project would be incomplete without the mention of the people who made it possible, without whose constant guidance and encouragement would have made efforts go in vain. We consider our self privileged to express gratitude and respect towards all those who have guided us through the completion of this project.

We convey thanks to all our teachers of Information Technology Department for providing encouragement, constant support and guidance which was of a great help to complete this project successfully.