SHRI VILE PARLE KELAVANI MANDAL'S
Institute of Technology, Dhule
Approved by AICTE | Affiliated to DBATU

A

Mini-Project Report

On

Extracting Text from Image Using Azure Cognitive Services

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Engineering in **Computer**

By

1) **Sukruta Prasanna Pardeshi (1954491245024)**
2) **Falguni Shashikant Shinde (1954491245026)**
3) **Priyanka Prashant Vibhandik (1954491245022)**
4) **Lina Hemantkumar Thakare (1954491245034)**
5) **Shalaka Sanjay Sharma (1954491245011)**
6) **Shraddha Anil Changune (1954491245037)**

Under the guidance

of

Ms. Vijaylaxmi Bittal

Department of Computer Engineering

SVKM'S Institute of Technology, Dhule
DHULE- 424002
Dr. Babasaheb Ambedkar Technological University, Lonere

Academic Year 2021 – 2022

SHRI VILE PARLE KELAVANI MANDAL'S
Institute of Technology, Dhule
Approved by AICTE | Affiliated to DBATU

SVKM's
Institute of Technology, Dhule
DHULE- 424002
Dr. Babasaheb Ambedkar Technological University, Lonere

Academic Year 2021 – 2022

MINI-PROJECT REPORT

Title of the Project:	Extracting Text from Image Using Azure Cognitive Services
Company Name:	AFLIPS
Student Name(s):	1) Shraddha Anil Changune (President) 2) Shalaka Sanjay Sharma (CEO) 3) Falguni Shashikant Shinde (CIO) 4) Priyanka Prashant Vibhandik (CTO) 5) Sukruta Prasanna Pardeshi (AI Engineer) 6) Lina Hemantkumar Thakare (AI Engineer)
Class:	T. E. (Computer Engineering)
College:	SVKM's Institute of Technology, Dhule
Semester:	VI
Name of the Guide:	Ms. Vijaylaxmi Bittal

Index

Table of Contents

Sr. No	Title	Page No.
1.	Abstract	1
2.	Introduction	1
3.	Aim and Objective	1
4.	Design and Methodology	1-5
5.	Results	5-8

1. Abstract:

This project is based on extraction of text present in an image with the help of Microsoft Azure Services; here the cognitive services (computer vision) is used for the same.

Optical character recognition (OCR) allows the extraction of printed or handwritten text from images, such as photos of street signs and products, as well as from documents—invoices, bills, financial reports, articles, and more. Microsoft's OCR technologies support extracting printed text in several languages.

The Computer Vision Read API is Azure's latest OCR technology that extracts printed text (in several languages), handwritten text (in several languages), digits, and currency symbols from images and multi-page PDF documents. It's optimized to extract text from text-heavy images and multi-page PDF documents with mixed languages. It supports extracting both printed and handwritten text in the same image or document.

2. Introduction:

Azure's Computer Vision service gives access to advanced algorithms that process images and return information based on the visual features one’s interested in. With the help of Optical character recognition (OCR) that allows the extraction of printed or handwritten text from images, the text from given image is extracted. OCR makes the use of deep-learning-based models and works with text on a variety of surfaces and backgrounds. Microsoft's OCR technologies support extracting printed text in several languages.

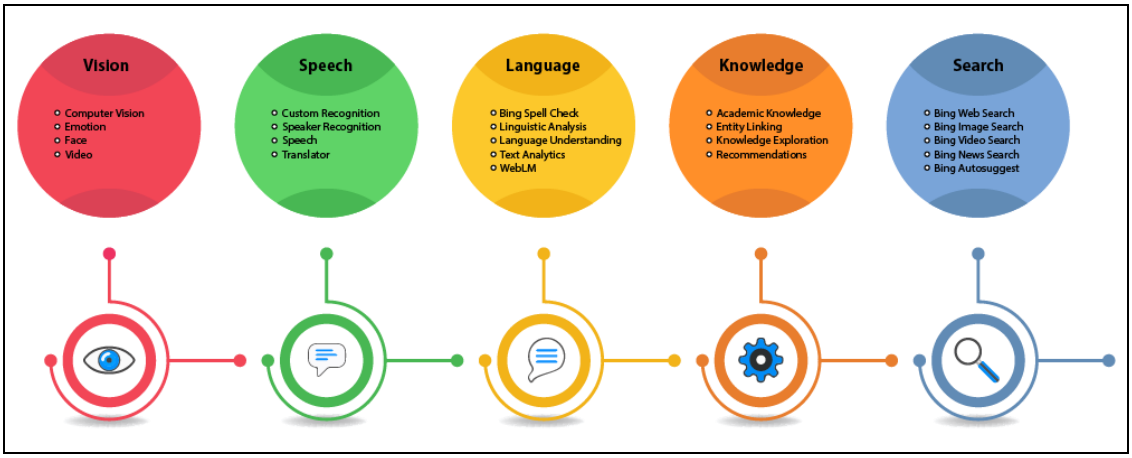
3. Aim and Objective:

The main objective of our project is to extract text from any given image. The text can be in any format like it can either be printed or handwritten. Many times we need to make use of text from images to create other new documents, in such cases text extraction can prove to be helpful since workload as well as time required for manually writing the text from image.

4. Design and Methodology:

4.1 Cognitive Services:

Cognitive Services are a set of machine learning algorithms that Microsoft has developed to solve problems in the field of Artificial Intelligence (AI). The goal of Cognitive Services is to democratize AI by packaging it into discrete components that are easy for developers to use in their own apps. Web and Universal Windows Platform developers can consume these algorithms through standard REST calls over the Internet to the Cognitive Services APIs.



The Cognitive Services APIs are grouped into five categories…

- Vision—analyze images and videos for content and other useful information.
- Speech—tools to improve speech recognition and identify the speaker.
- Language—understanding sentences and intent rather than just words.
- Knowledge—tracks down research from scientific journals for you.
- Search—applies machine learning to web searches.

The Vision APIs are broken out into five groups of tasks…

- Computer Vision—Distill actionable information from images.
- Content Moderator—Automatically moderate text, images and videos for profanity and inappropriate content.
- Emotion—Analyze faces to detect a range of moods.
- Face—identify faces and similarities between faces.
- Video—Analyze, edit and process videos within your app.

4.2 What is Computer Vision?

Azure's Computer Vision service gives you access to advanced algorithms that process images and return information based on the visual features you are interested in.

4.2.1 Optical Character Recognition (OCR):

The Optical Character Recognition (OCR) service extracts text from images. You can use the new Read API to extract printed and handwritten text from photos and documents. It uses deep-learning-based models and works with text on a variety of surfaces and backgrounds. These include business documents, invoices, receipts, posters, business cards, letters, and whiteboards. The OCR APIs support extracting printed text in several languages. Follow the OCR quickstart to get started.

4.2.2 Image Analysis:

The Image Analysis service extracts many visual features from images, such as objects, faces, adult content, and auto-generated text descriptions. Follow the Image Analysis quickstart to get started.

4.2.3 Face:

The Face service provides AI algorithms that detect, recognize, and analyze human faces in images. Facial recognition software is important in many different scenarios, such as identity verification, touchless access control, and face blurring for privacy. Follow the Face quickstart to get started.

4.2.4 Spatial Analysis:

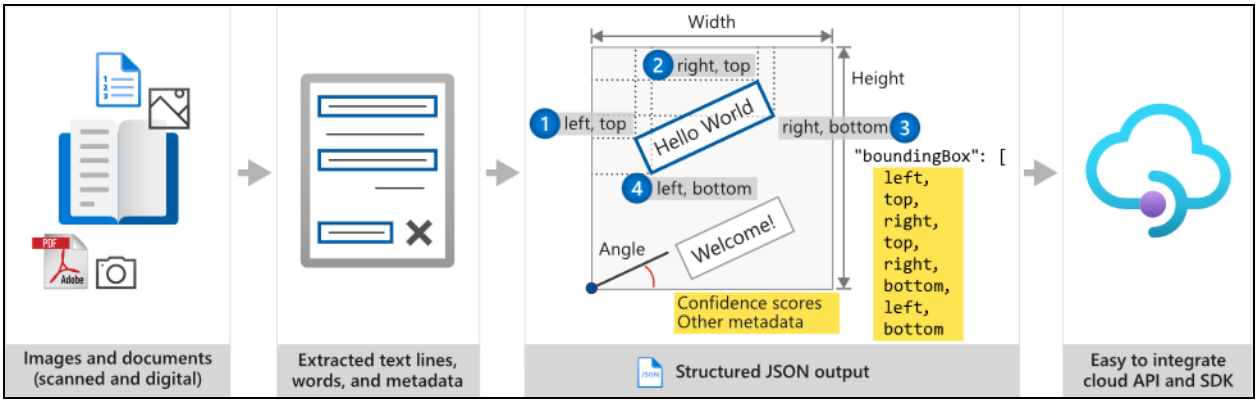
The Spatial Analysis service analyzes the presence and movement of people on a video feed and produces events that other systems can respond to. Install the Spatial Analysis container to get started.

4.3 What is Optical character recognition?

Optical character recognition (OCR) allows you to extract printed or handwritten text from images, such as photos of street signs and products, as well as from documents—invoices, bills, financial reports, articles, and more. Microsoft's OCR technologies support extracting printed text in several languages.

4.3.1 Read API

The Computer Vision Read API is Azure's latest OCR technology (learn what's new) that extracts printed text (in several languages), handwritten text (in several languages), digits, and currency symbols from images and multi-page PDF documents. It's optimized to extract text from text-heavy images and multi-page PDF documents with mixed languages. It supports extracting both printed and handwritten text in the same image or document.



Input requirements: The Read call takes images and documents as its input. They have the following requirements:

- Supported file formats: JPEG, PNG, BMP, PDF, and TIFF
- For PDF and TIFF files, up to 2000 pages (only the first two pages for the free tier) are processed.
- The file size of images must be less than 500 MB (4 MB for the free tier) and dimensions at least 50 x 50 pixels and at most 10000 x 10000 pixels. PDF files do not have a size limit.
- The minimum height of the text to be extracted is 12 pixels for a 1024 x 768 image. This corresponds to about 8 font point text at 150 DPI.

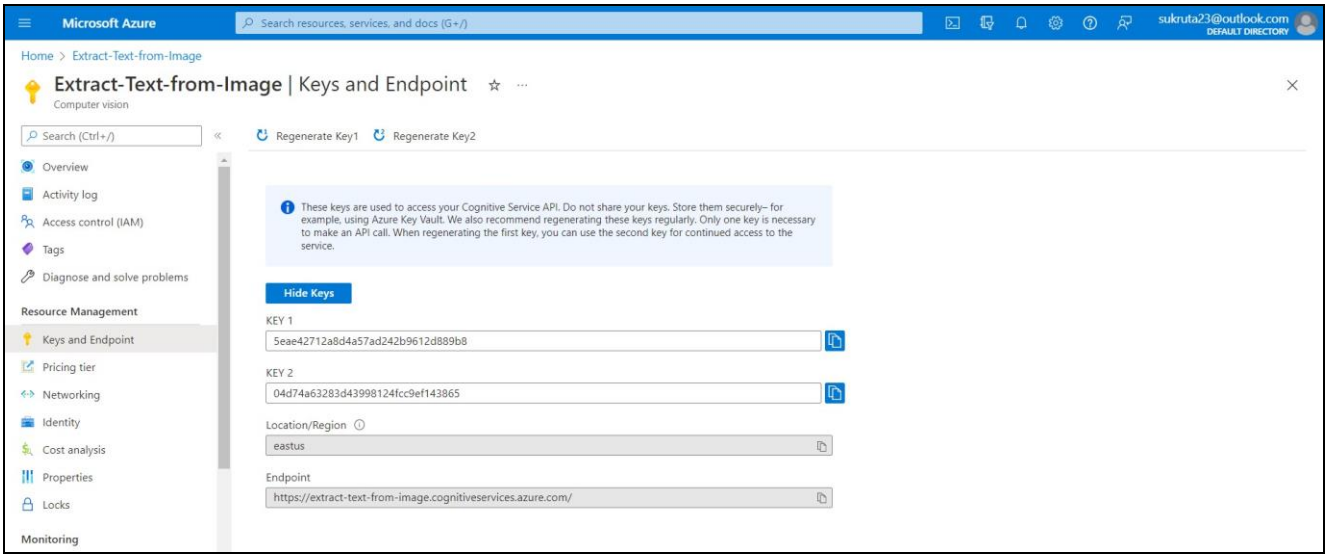
Supported languages: The Read API latest generally available (GA) model supports 164 languages for print text and 9 languages for handwritten text. OCR for print text includes support for English, French, German, Italian, Portuguese, Spanish, Chinese, Japanese, Korean, Russian, Arabic, Hindi, and other international languages that use Latin, Cyrillic, Arabic, and Devanagari scripts. OCR for handwritten text includes support for English, Chinese Simplified, French, German, Italian, Japanese, Korean, Portuguese, Spanish languages.

Key features: The Read API includes the following features:

- Print text extraction in 164 languages
- Handwritten text extraction in nine languages
- Text lines and words with location and confidence scores
- No language identification required
- Support for mixed languages, mixed mode (print and handwritten)
- Select pages and page ranges from large, multi-page documents
- Natural reading order option for text line output (Latin only)
- Handwriting classification for text lines (Latin only)
- Available as Distroless Docker container for on-premises deployment

5. Results:

5.1 Creating Computer Vision API:



5.2 Code:

1. Using Image URL:

```
#using image url
import os
import sys
import time
from PIL import Image
from array import array
from azure.cognitiveservices.vision.computervision import
ComputerVisionClient
from azure.cognitiveservices.vision.computervision.models import
OperationStatusCodes
```

```
from azure.cognitiveservices.vision.computervision.models import
VisualFeatureTypes
from msrest.authentication import CognitiveServicesCredentials

API_KEY = "5eae42712a8d4a57ad242b9612d889b8"
ENDPOINT = "https://extract-text-from-image.cognitiveservices.azure.com/"
computervision_client = ComputerVisionClient(ENDPOINT,
CognitiveServicesCredentials(API_KEY))

print("===== START - Read File - remote =====")
read_image_url = "https://gtelocalize.com/wp-content/uploads/2020/12/OCR-
Software.png"
read_response = computervision_client.read(read_image_url,  raw=True)
read_operation_location = read_response.headers["Operation-Location"]
operation_id = read_operation_location.split("/")[-1]

while True:
    read_result = computervision_client.get_read_result(operation_id)
    if read_result.status not in ['notStarted', 'running']:
        break
    time.sleep(1)

if read_result.status == OperationStatusCodes.succeeded:
    for text_result in read_result.analyze_result.read_results:
        for line in text_result.lines:
            print(line.text)
print()
print("===== END - Read File - remote =====")
```

2. Using Local Image File:

```
#using local image file
import os
import io
import json
import time
from array import array
from azure.cognitiveservices.vision.computervision import
ComputerVisionClient
from azure.cognitiveservices.vision.computervision.models import
OperationStatusCodes
from azure.cognitiveservices.vision.computervision.models import
VisualFeatureTypes
from msrest.authentication import CognitiveServicesCredentials
import requests
from PIL import Image, ImageDraw, ImageFont
```

```
API_KEY = "5eae42712a8d4a57ad242b9612d889b8"
ENDPOINT = "https://extract-text-from-image.cognitiveservices.azure.com/"
computervision_client = ComputerVisionClient(ENDPOINT,
CognitiveServicesCredentials(API_KEY))

print("===== START - Read File - remote =====")
# read_image_url = "https://twistarticle.com/wp-
content/uploads/2021/02/Improving-Workflow-Efficiency-by-Using-OCR-
Technologies.jpg"
# read_response = computervision_client.read(read_image_url,  raw=True)
local_image  = './images/img1.jpg'
read_response = computervision_client.read_in_stream(open(local_image, 'rb'),
language ='en', raw=True)
read_operation_location = read_response.headers["Operation-Location"]
operation_id = read_operation_location.split("/")[-1]

while True:
    read_result = computervision_client.get_read_result(operation_id)
    if read_result.status not in ['notStarted', 'running']:
        break
    time.sleep(2)

if read_result.status == OperationStatusCodes.succeeded:
    text = ''
    for text_result in read_result.analyze_result.read_results:
        for line in text_result.lines:
                text += line.text + ' '
                text += '\n'
        open('output.txt', 'w').write(text)
        print()
    print("Text successfully written in output file")
print()
print("===== END - Read File - remote =====")
```

5.3 Output:

1. Using Image URL:

Image:

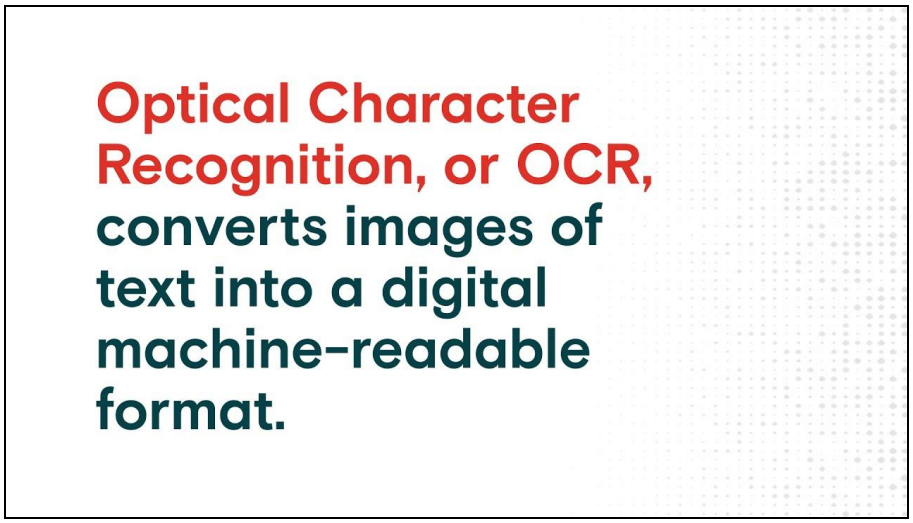
Output:

```
===== START - Read File - remote =====
Text
SCANNED FILE
OCR SOFTWARE
EDITABLE FILE

===== END - Read File - remote =====
```

2. Using Local Image File:

Image:



Output:

```
===== START - Read File - remote =====

Text successfully written in output file

===== END - Read File - remote =====
```

output.txt: