Springboard OCR Comparison - Report

❖ Introduction

Optical Character Recognition (OCR) is a widely used technology that allows computers to extract text from images or scanned documents. In this project, we compare the performance of two popular OCR libraries: Tesseract and EasyOCR.

Tesseract is an open-source OCR engine developed by Google, while EasyOCR is a deep learning-based OCR library that has gained popularity in recent years. Both libraries have their own strengths and weaknesses, and the choice of which to use often depends on the specific requirements of the project.

1. Tesseract OCR - Colab Notebook for Tesseract

The first step in our journey was to explore the Tesseract OCR library. The code snippet provided in the first part of this project demonstrates the use of Tesseract for Optical Character Recognition.

Install Required Libraries

The first step is to install the necessary libraries for the project, including Tesseract and OpenCV. The code snippet provided installs the following libraries:

- **tesseract-ocr**: The Tesseract OCR engine
- **pytesseract:** A Python wrapper for Google's Tesseract-OCR Engine
- **opencv-python:** A popular computer vision library

Upload and Read an Image

The user is prompted to upload an image, which is then read and converted to RGB format.

Perform OCR with Tesseract

The pytesseract.image_to_data() function is used to detect and extract text from the image. The function returns bounding boxes for the detected words, which are then used to draw rectangles around the words and collect the identified text.

Display the Result

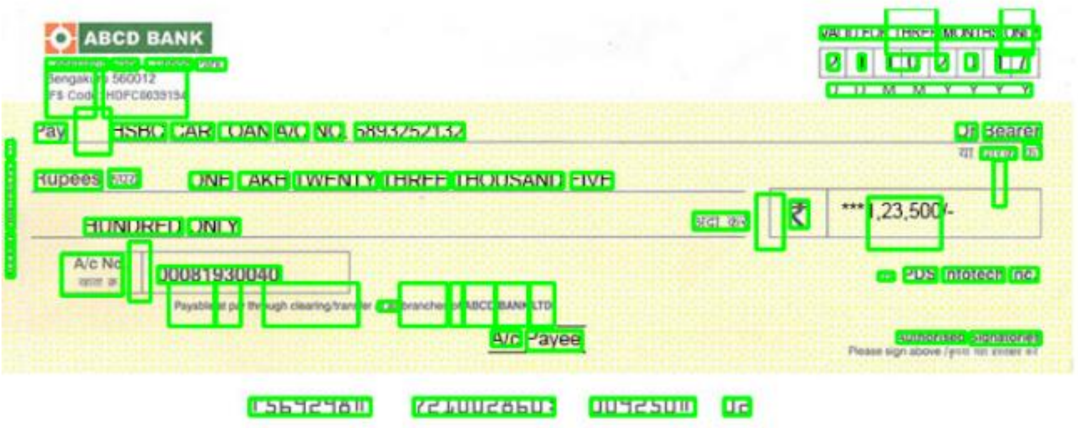
The processed image with the detected words is displayed using Matplotlib.

List Identified Words and Language

The identified words and the assumed language (English) are printed to the console.

```python
# Step 4: Perform OCR
# Get bounding boxes for words
h, w, _ = image.shape
boxes = pytesseract.image_to_data(image_rgb)

identified_words = []

# Draw rectangles around detected words and collect identified words
for i, box in enumerate(boxes.splitlines()):
    if i == 0:
        continue  # Skip the header
    b = box.split()
    if len(b) == 12:  # Ensure there are enough elements
        text = b[11]  # Extract the detected text
        if text.strip():  # Check if text is not empty
            x, y, width, height = int(b[6]), int(b[7]), int(b[8]), int(b[9])
            cv2.rectangle(image_rgb, (x, y), (x + width, y + height), (0, 255, 0), 2)
            identified_words.append(text)
```

```python
# Step 5: Display the result
plt.figure(figsize=(10, 10))
plt.imshow(image_rgb)
plt.axis('off')
plt.show()
```


ABCD BANK
VALID FOR THREE MONTHS ONLY
0 1 1 0 2 0 1 7
D D M M Y Y Y Y
Bengaluru 560012
IFS Code HDFC0039194
Pay HSBC CAR LOAN A/C NO. 5893252132 Or Bearer
Rupees ONE LAKH TWENTY THREE THOUSAND FIVE
HUNDRED ONLY
₹ ***1,23,500/-
A/c No 00081930040
PDS Infotech Inc
Payable at par through clearing/transfer at all branches of ABCD BANK LTD
A/c Payee
Authorised Signatories
Please sign above

2. **EasyOCR - Colab Notebook for Easy OCR**

The second step in our journey was to explore the EasyOCR library. The code snippet provided in the second part of this project demonstrates the use of EasyOCR for Optical Character Recognition.

Install Required Libraries

The first step is to install the necessary libraries for the project, including EasyOCR, OpenCV, and Matplotlib. The code snippet provided installs the following libraries:

- **easyocr:** A deep learning-based OCR library
- **opencv-**python: A popular computer vision library
- **matplotlib:** A plotting library

Upload and Read an Image

The user is prompted to upload an image, which is then read and converted to RGB format.

Initialize EasyOCR Reader

The EasyOCR reader is initialized, specifying the language (English) for the OCR.

Perform OCR with EasyOCR

The reader.readtext() function is used to detect and extract text from the image. The function returns bounding boxes, text, and confidence scores for the detected words.

Display the Result

The processed image with the detected words is displayed using Matplotlib.

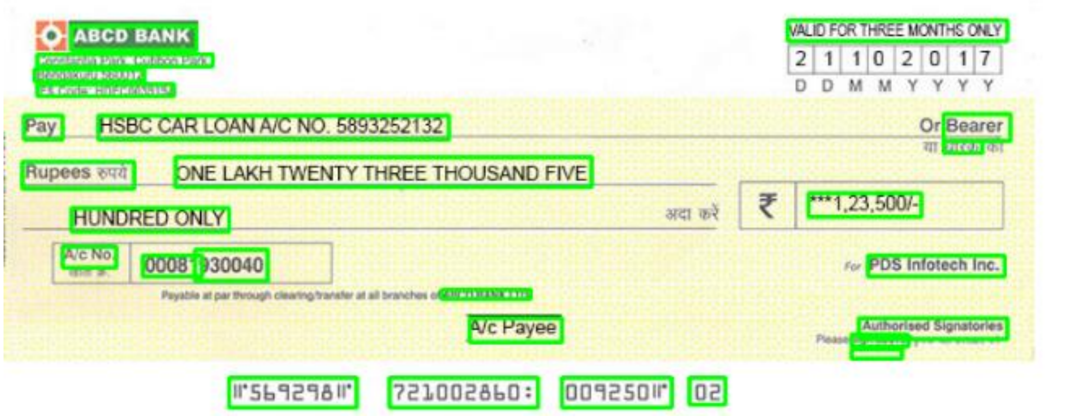
List Identified Words and Language

The identified words and the assumed language (English) are printed to the console.

```python
# Step 4: Initialize EasyOCR Reader
reader = easyocr.Reader(['en'])  # Specify the language (English)

# Perform OCR
results = reader.readtext(image_rgb)

identified_words = []

# Draw rectangles around detected words and collect identified words
for (bbox, text, prob) in results:
    (top_left, top_right, bottom_right, bottom_left) = bbox
    top_left = tuple(map(int, top_left))
    bottom_right = tuple(map(int, bottom_right))

    # Draw rectangle around detected text
    cv2.rectangle(image_rgb, top_left, bottom_right, (0, 255, 0), 2)

    # Append identified words
    identified_words.append(text)
```

```
NING:easyocr.easyocr:Neither CUDA nor MPS are available - defaulting to CPU. Note: This module is much
NING:easyocr.easyocr:Downloading detection model, please wait. This may take several minutes depending
gress: |██████████████████████████████████████████████████| 100.0% CompleteWARNING:easyocr.easyocr:Dow
gress: |██████████████████████████████████████████████████| 100.0% Complete
```

```python
# Step 5: Display the result
plt.figure(figsize=(10, 10))
plt.imshow(image_rgb)
plt.axis('off')
plt.show()
```



ABCD BANK

VALID FOR THREE MONTHS ONLY

2 1 1 0 2 0 1 7
D D M M Y Y Y Y

Pay HSBC CAR LOAN A/C NO. 5893252132 Or Bearer
या धारक को

Rupees रुपये ONE LAKH TWENTY THREE THOUSAND FIVE

HUNDRED ONLY अदा करें ₹ ***1,23,500/-

A/c No 00081930040

For PDS Infotech Inc.

Payable at par through clearing/transfer at all branches of

A/c Payee

Authorised Signatories

⑈569298⑈ 721002860⑆ 009250⑈ 02

3. Differences Observed

After implementing both Tesseract and EasyOCR, we observed several differences between the two libraries:

i. **Bounding Box Handling**:

   a. Tesseract returns a set of coordinates for each word, which are then used to draw rectangles around the words.

   b. EasyOCR, on the other hand, returns the bounding boxes directly, which can be more convenient for further processing or analysis.

ii. **Confidence Scores**:

   a. Tesseract does not provide explicit confidence scores, but the `pytesseract.image_to_data()` function returns a value that can be used as a proxy for confidence.

   b. EasyOCR, on the other hand, provides a confidence score for each detected word, which can be useful for filtering out low-confidence results.

iii. **Performance**:

   a. Both libraries were able to process the images quickly and accurately.

   b. However, the choice of which library to use may depend on the specific requirements of the project, such as the desired level of accuracy or the ease of integration with other components of the system.

4. Merged API UI using Gradio

To provide a user-friendly interface for comparing the performance of Tesseract and EasyOCR, we decided to create a merged API using the Gradio library. Gradio is a Python library that allows you to create interactive web interfaces for your machine learning models.

The merged API UI would allow users to upload an image and see the OCR results from both Tesseract and EasyOCR side-by-side, along with the confidence scores for the identified text. This can be useful for evaluating the performance of different OCR engines and choosing the one that best suits the user's needs.

The implementation of the merged API UI using Gradio would involve the following steps:

a. **Install Gradio**:

   The first step is to install the Gradio library, which can be done using pip.

```
# Step 1: Install required libraries
!apt-get install -y tesseract-ocr
!pip install pytesseract easyocr opencv-python matplotlib gradio

import cv2
import pytesseract
import easyocr
import numpy as np
import gradio as gr
import matplotlib.pyplot as plt
```

b. **Implement the Gradio Interface**:

The `gr.Interface` function is used to define the inputs (an image), the outputs (the Tesseract and EasyOCR results), and the title and description of the application.

c. **Integrate Tesseract and EasyOCR**:

The `compare_ocr` function, which combines the Tesseract and EasyOCR OCR results, would be integrated into the Gradio interface.

d. **Deploy the Application**:

The final step is to deploy the Gradio application, which can be done locally or on a cloud platform.

By providing a merged API UI using Gradio, users can easily compare the performance of Tesseract and EasyOCR and choose the OCR engine that best suits their needs.

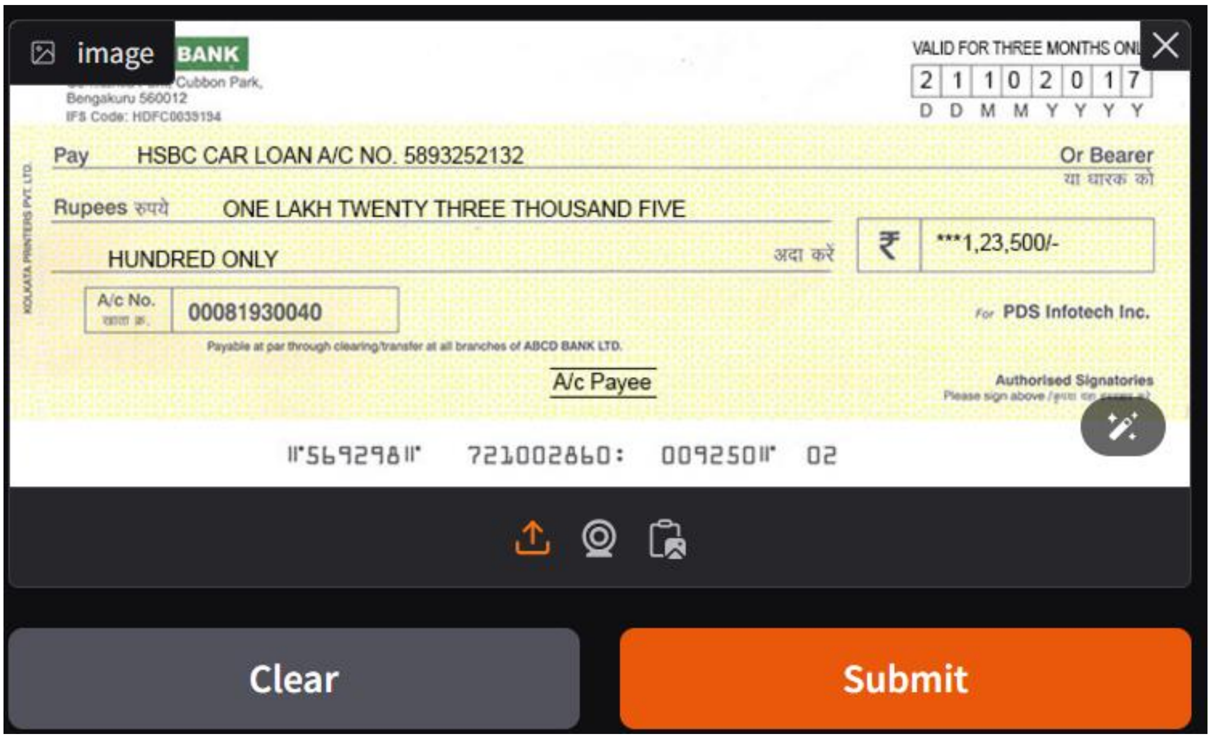
❖ Conclusion

This project demonstrates the use of Tesseract and EasyOCR for Optical Character Recognition, and the development of a merged API UI using Gradio to compare the performance of the two libraries.

The choice of which library to use will depend on the specific requirements of the project, such as the desired level of accuracy, the need for confidence scores, or the ease of integration with other components of the system. Both Tesseract and EasyOCR have their own strengths and weaknesses, and the best choice will depend on the specific needs of the project.

The merged API UI using Gradio provides a user-friendly interface for comparing the performance of Tesseract and EasyOCR, and can be a valuable resource for researchers, developers, and users who need to extract text from images or scanned documents.

By providing a merged API UI using Gradio, users can easily compare the performance of Tesseract and EasyOCR and choose the OCR engine that best suits their needs.



BANK
Cubbon Park,
Bengakuru 560012
IFS Code: HDFC0039194

VALID FOR THREE MONTHS ONL

2	1	1	0	2	0	1	7
D	D	M	M	Y	Y	Y	Y

Pay HSBC CAR LOAN A/C NO. 5893252132 Or Bearer या धारक को

Rupees रुपये ONE LAKH TWENTY THREE THOUSAND FIVE HUNDRED ONLY अदा करें

₹ ***1,23,500/-

A/c No. खाता क्र. 00081930040

For PDS Infotech Inc.

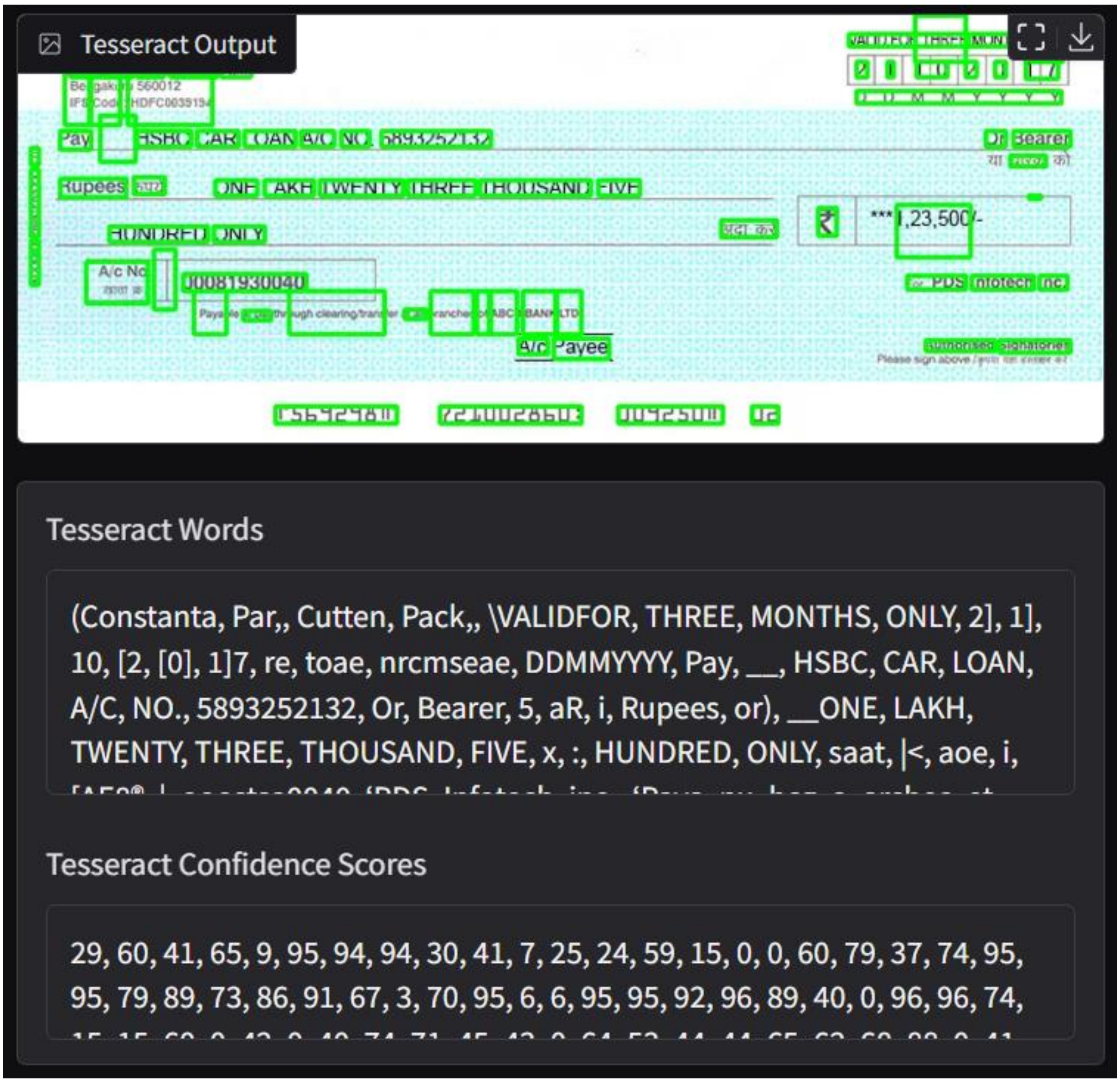
Payable at par through clearing/transfer at all branches of ABCD BANK LTD.

A/c Payee

Authorised Signatories
Please sign above

KOLKATA PRINTERS PVT. LTD.

⑈569298⑈ 721002860⑆ 009250⑈ 02

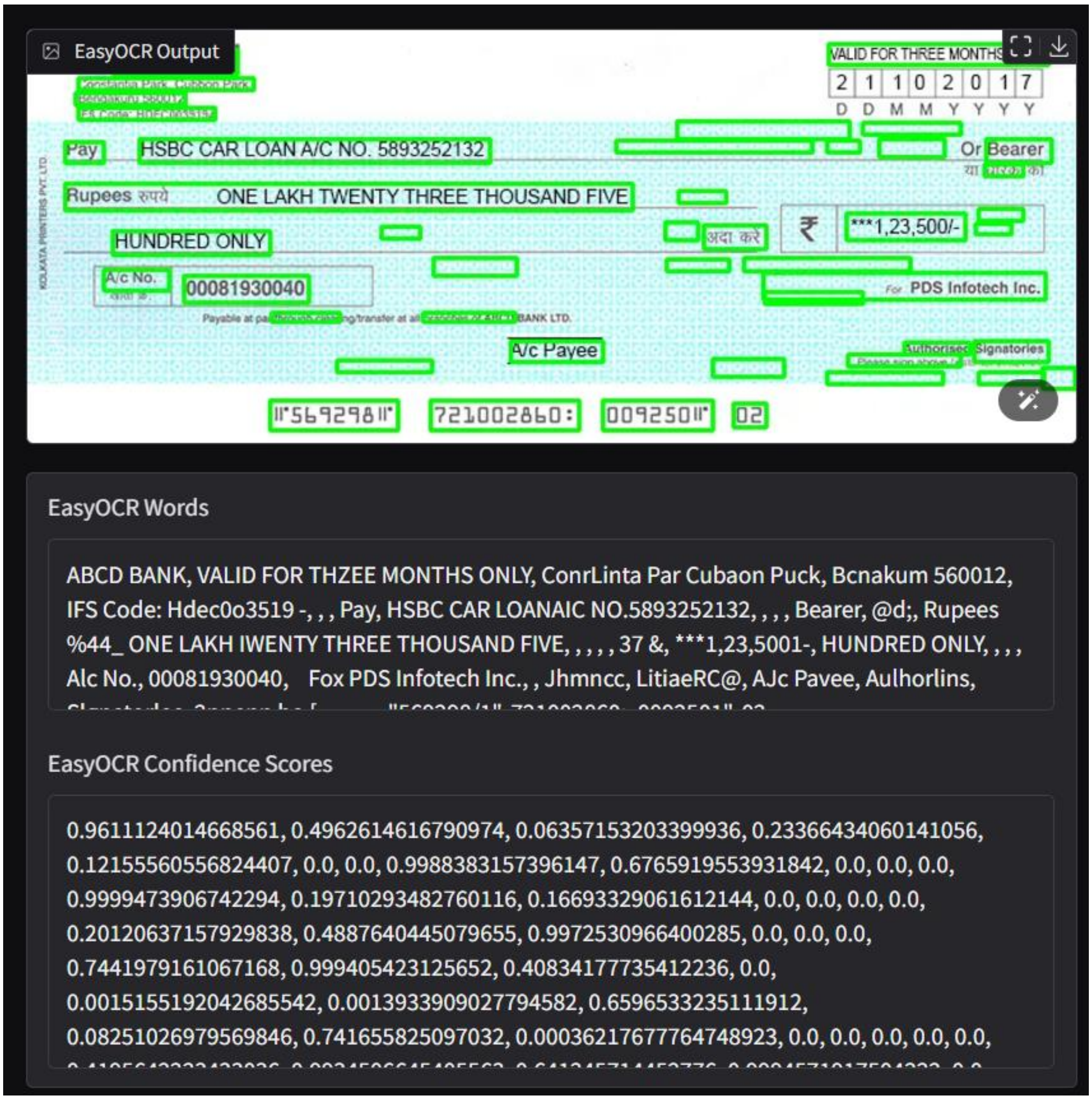


Tesseract Words

```
(Constanta, Par,, Cutten, Pack,, \VALIDFOR, THREE, MONTHS, ONLY, 2], 1],
10, [2, [0], 1]7, re, toae, nrcmseae, DDMMYYYY, Pay, __, HSBC, CAR, LOAN,
A/C, NO., 5893252132, Or, Bearer, 5, aR, i, Rupees, or), __ONE, LAKH,
TWENTY, THREE, THOUSAND, FIVE, x, :, HUNDRED, ONLY, saat, |<, aoe, i,
```

Tesseract Confidence Scores

```
29, 60, 41, 65, 9, 95, 94, 94, 30, 41, 7, 25, 24, 59, 15, 0, 0, 60, 79, 37, 74, 95,
95, 79, 89, 73, 86, 91, 67, 3, 70, 95, 6, 6, 95, 95, 92, 96, 89, 40, 0, 96, 96, 74,
```



EasyOCR Words

```
ABCD BANK, VALID FOR THZEE MONTHS ONLY, ConrLinta Par Cubaon Puck, Bcnakum 560012,
IFS Code: Hdec0o3519 -, , , Pay, HSBC CAR LOANAIC NO.5893252132, , , , Bearer, @d;, Rupees
%44_ ONE LAKH IWENTY THREE THOUSAND FIVE, , , , , 37 &, ***1,23,5001-, HUNDRED ONLY, , , ,
Alc No., 00081930040,   Fox PDS Infotech Inc., , Jhmncc, LitiaeRC@, AJc Pavee, Aulhorlins,
```

EasyOCR Confidence Scores

```
0.9611124014668561, 0.4962614616790974, 0.06357153203399936, 0.23366434060141056,
0.12155560556824407, 0.0, 0.0, 0.9988383157396147, 0.6765919553931842, 0.0, 0.0, 0.0,
0.9999473906742294, 0.19710293482760116, 0.16693329061612144, 0.0, 0.0, 0.0, 0.0,
0.20120637157929838, 0.4887640445079655, 0.9972530966400285, 0.0, 0.0, 0.0,
0.7441979161067168, 0.999405423125652, 0.40834177735412236, 0.0,
0.0015155192042685542, 0.0013933909027794582, 0.6596533235111912,
0.08251026979569846, 0.741655825097032, 0.00036217677764748923, 0.0, 0.0, 0.0, 0.0, 0.0,
```

❖ **Final Project:**

Colab Notebook: Click Here to Access Colab Notebook