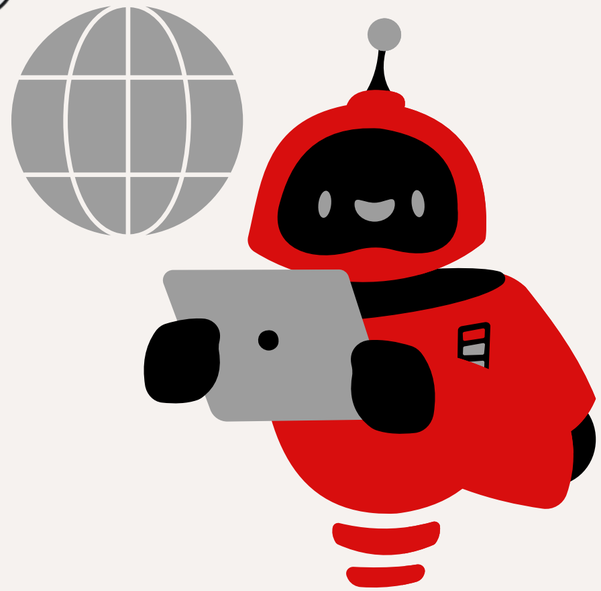
O . C . R

OPTICAL CHARACTER RECOGNITION

OCR

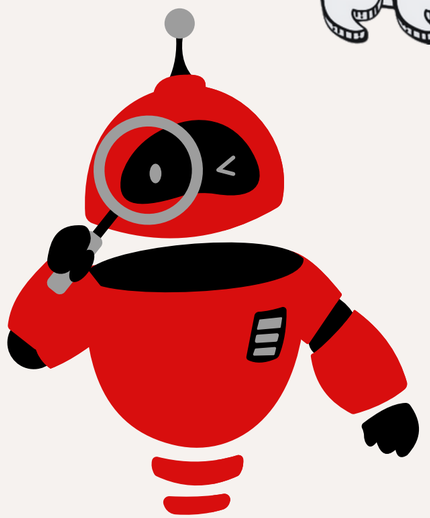
Struggling with tedious data entry?

OCR is here to simplify your workflow!



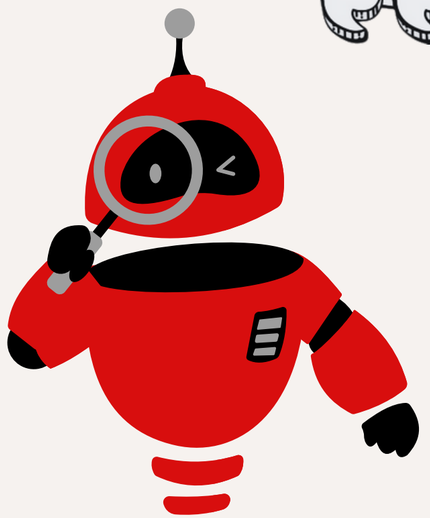
Hey, Nexus did ever tired of manual data entry? Its really hectic and stressful

Hello, Nova Yeah, I totally get it !
But what if I told you there’s a way to make it easier?



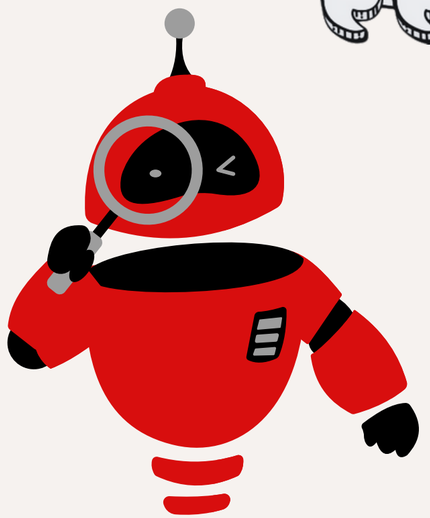
Is it so ?

with OCR, we quickly convert images to text



That means less time on paperwork and fewer errors.

Exactly! OCR streamlines everything, making work easier.



Introduction

OCR

Optical character recognition (OCR), a technology used for converting different types of documents, such as scanned paper documents, PDF files or images captured by a digital camera into editable and searchable data. OCR has played a significant role in digitizing printed texts to allow for editing, searching, and virtual storage.

Different Libraries

Tesseract

PyTesseract

EasyOCR

OCR.space

Amazon Textract

Google Cloud Vision

Adobe SDK

ABBYY SDKs

Keras OCR

GOCR

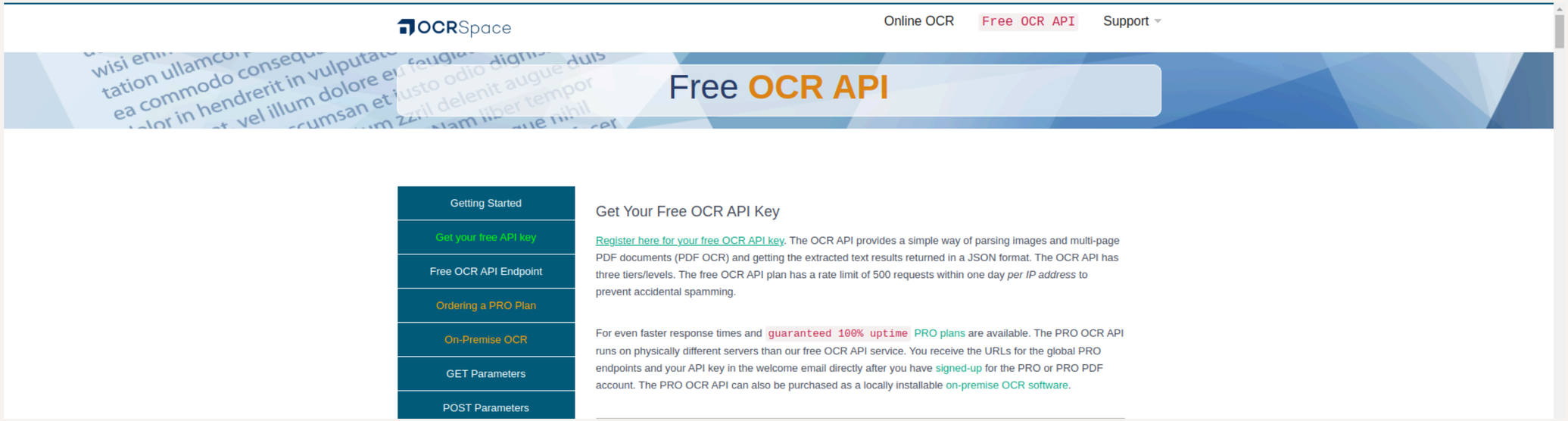
PDFMiner

OCRopus

Tutorial Using OCR.Space

This is an OCR API that allows developers to extract text from images. It offers a Python wrapper for the API, which is efficient and easy to use.

Get your API key from OCR.space. You get this key by signing up on OCR.space



- **click on Free OCR API**
- **Login using mail (Confirm ur mail)**
- **You will recieve the Free API In mail**

- **You need the requests library in Python installed. Install it using pip if you don’t have it already.**
- Use the API key you recieved in mail.
- **Write a Python program. In the program, use the requests library to send HTTP requests to OCR.space. The API endpoint is https://api.ocr.space/parse/image.**
- In your request, send your API key as a parameter along with your image file on which you want to perform the OCR.
- **After sending the request, OCR.space will process the image and return the recognised text in the response. You can integrate with Frontend for better interface.**

CAPTURE

Utilize cutting-edge OCR technology to extract and digitize text from images and documents with precision and reliability.

PROCESS

Implement OCR systems to efficiently convert scanned text into editable and searchable formats, streamlining data workflows.

OPTIMIZE

Continuously improve OCR accuracy by fine-tuning models and adapting to new fonts and formats, ensuring seamless text recognition.

Challenges

Accuracy and quality of OCR can be influenced by several factors such as

1. Poor image quality and skewing.
2. Complex layouts of the document to be scanned.
3. Variations in text size, font, style.
4. In case of handwriting recognition, variability in handwriting styles poses a significant challenge.

Conclusion

OCR as a field has advanced significantly over the years due to advancements in computer vision and machine learning. However, the challenges in OCR provide opportunities for further research and improvement. Python's contribution with its efficient libraries has extended the potential of OCR implementations. The applications of OCR are broad, and they are becoming increasingly integrated into our digital experiences.

Thank you