Lijst van Python OCR libraries:

* pytesseract (heel bekend en veel gebruikt)

Python-tesseract is a python wrapper for Google's Tesseract-OCR.

* Tesserocr

(weinig informatie over te vinden, in github repo staat niks over het gekozen algortime, wel stond erin dat deze library ook een andere tesseract libray import)

* ReadBot

(net als tesserocr gebruikt deze library ook een geïmporteerde library en is in principe een wrapper die alleen maar fungeert als een makkelijke interface voor de oorspronkelijke tesseract)

* pyocr

(pyocr maakt gebruikt van een “zelgemaakt”-algortime, dat goed van toepassing is bij tekst herkennen vanaf foto’s. Hiervoor is een bijlage te vinden in de map bijlagen onder het bestand “Detecting Text in Natural Scenes with Stroke Width Transform”. Deze methode van tekst herkenning is uitzonderlijk goed voor de toepassing in natuurlijke foto’s en om daar tekst uit te herkennen.)

* tesserwrap

(ook weer een wrapper van de tesseract-ocr, geen algoritmesover te vinden)

* TesseractTrainer

(is een API die de taak van het trainen van tesseract-ocr’s Tesseract3 niet meer nodig maakt. Volgens de GitHub heeft deze library een nieuwe eigenaar die al sinds juni 2016 geen nieuwe commits heeft gemaakt, de laatste commit was een aanpassing in de readMe.md file)

* tesseract\_sip

(nog een wrapper die gebaseerd is op tesseract, maakt nu gebruik van een koppelings programma genaamd SIP dat mogelijk maakt om C en C++ libraries met Python te koppelen)

* tesseract-ocr

//de engine OCR engine van goolge

* tightocr

(een tight wrapper for the tesseract-ocr, bedoeld voor de toepassing van het OCR’en van documenten)

* tesserpy

(Python interface voor tesseract-ocr)

* [textract](https://textract.readthedocs.io/en/stable/)
* Openalpr

In na een aantal paper gelezen te hebben, begin ik het gevoel te hebben dat de OCR niet persé afhankelijk is van een algoritme, de methode van het zoeken van letters en cijfers op een foto in combinatie met een goed algoritme is wat uiteindelijk de performance bepaald voor een OCR.

bronnen: <https://www.reddit.com/r/MachineLearning/comments/3q1wkz/your_favorite_python_ocr_library/>,<https://textract.readthedocs.io/en/stable/>, <https://github.com/mwmeyer/readbot>, <https://github.com/sirfz/tesserocr>, <https://gitlab.gnome.org/World/OpenPaperwork/pyocr>, <https://pypi.org/project/tesserwrap/>, <https://pypi.org/project/TesseractTrainer/>, <https://pypi.org/project/tesseract_sip/>, <https://github.com/amazon-archives/tesserpy>, <https://pypi.org/project/tightocr/>, <https://scikit-image.org/docs/dev/auto_examples/segmentation/plot_niblack_sauvola.html>,