



Optical Character Reorganization

Using Machine Learning

DOCUMENTATION

INDEX

Sl. No.	Description	Page No.
1	Introduction to pyimageocr	1
2	Python	2
3	Machine Learning	2
4	Package Used	4
5	Installation	5
6	Algorithm	5
7	“pyimageocr” structure	7
8	Example Code	8
9	Output	8
10	Reference	9

PYIMAGEOCR

pyimageocr is an optical character reorganization library for python this is *underdeveloped* library based on machine learning. pyimageocr store edge filter image in user's folder and when pyimageocr scan any image it matches with previously present data in database in case of match it returns success otherwise it tests percentage match, in case of good accuracy it returns percentage match and the latter which is close to that. pyimageocr performs object matching, histogram analysis and many type of filter on image. pyimageocr can detect text in hazy background.

PYTHON

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Python source code is also available under the GNU General Public License (GPL). Python is processed at runtime by the interpreter.

A Hello World program in Python

```
#!/User/bin/python/hello.py  
print "Hello, Python!"
```

MACHINE LEARNING

Machine learning is a field of computer science that uses statistical techniques to give computer systems the ability to "learn" (i.e., progressively improve performance on a specific task) with data, without being explicitly programmed

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. **Machine learning focuses on the development of computer programs** that can access data and use it learn for themselves.

The process of learning begins with observations or data, such as examples, direct experience, or instruction, in order to look for patterns in data and make better decisions in the future based on the examples that we provide. **The primary aim is to allow the computers learn automatically** without human intervention or assistance and adjust actions accordingly.

SOME MACHINE LEARNING METHODS

Machine learning algorithms are often categorized as supervised or unsupervised.

- **Supervised machine learning algorithms** can apply what has been learned in the past to new data using labelled examples to predict future events. Starting from the analysis of a known training dataset, the learning algorithm produces an inferred function to make predictions about the output values. The system is able to provide

targets for any new input after sufficient training. The learning algorithm can also compare its output with the correct, intended output and find errors in order to modify the model accordingly.

- In contrast, **unsupervised machine learning algorithms** are used when the information used to train is neither classified nor labelled. Unsupervised learning studies how systems can infer a function to describe a hidden structure from unlabelled data. The system doesn't figure out the right output, but it explores the data and can draw inferences from datasets to describe hidden structures from unlabelled data.
- **Semi-supervised machine learning algorithms** fall somewhere in between supervised and unsupervised learning, since they use both labelled and unlabelled data for training – typically a small amount of labelled data and a large amount of unlabelled data. The systems that use this method are able to considerably improve learning accuracy. Usually, semi-supervised learning is chosen when the acquired labelled data requires skilled and relevant resources in order to train it / learn from it. Otherwise, acquiring unlabelled data generally doesn't require additional resources.
- **Reinforcement machine learning algorithms** is a learning method that interacts with its environment by producing actions and discovers errors or rewards. Trial and error search and delayed reward are the most relevant characteristics of reinforcement learning. This method allows machines and software agents to automatically determine the ideal behaviour within a specific context in order to maximize its performance. Simple reward feedback is required for the agent to learn which action is best; this is known as the reinforcement signal.

Machine learning enables analysis of massive quantities of data. While it generally delivers faster, more accurate results in order to identify profitable opportunities or dangerous risks, it may also require additional time and resources to train it properly. Combining machine learning with AI and cognitive technologies can make it even more effective in processing large volumes of information.

PACKAGE USED

OPEN CV (CV2)

This package contains only the OpenCV core modules without the optional contrib modules. If you are looking for a version which includes OpenCV contrib modules, please install `opencv-contrib-python` instead. The packages contain pre-compiled OpenCV binary with Python bindings. This enables super-fast (usually < 10 seconds) OpenCV installation for Python. If you need only OpenCV Python bindings, no separate OpenCV installation is required.

Installation and Usage

If you have previous/other version of OpenCV installed (e.g. `cv2` module in the root of Python's site-packages), remove it before installation to avoid conflicts.

1. If you have an existing `opencv-contrib-python` installation, run `pip uninstall opencv-contrib-python`
2. Install this package: `pip install opencv-python`

Import the package: `import cv2`

Documentation: <https://docs.opencv.org/master>

MATPLOTLIB

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter notebook, web application servers, and four graphical user interface toolkits.

Installation and Usage

Install this package: `pip install matplotlib`

Import the package: `import matplotlib`

Documentation: <https://matplotlib.org>

OS

This module provides a portable way of using operating system dependent functionality. If you just want to read or write a file see `open()`, if you want to manipulate paths, see the `os.path` module, and if you want to read all the lines in all the files on the command line see

the fileinput module. For creating temporary files and directories see the tempfile module, and for high-level file and directory handling see the shutil module.

Usage

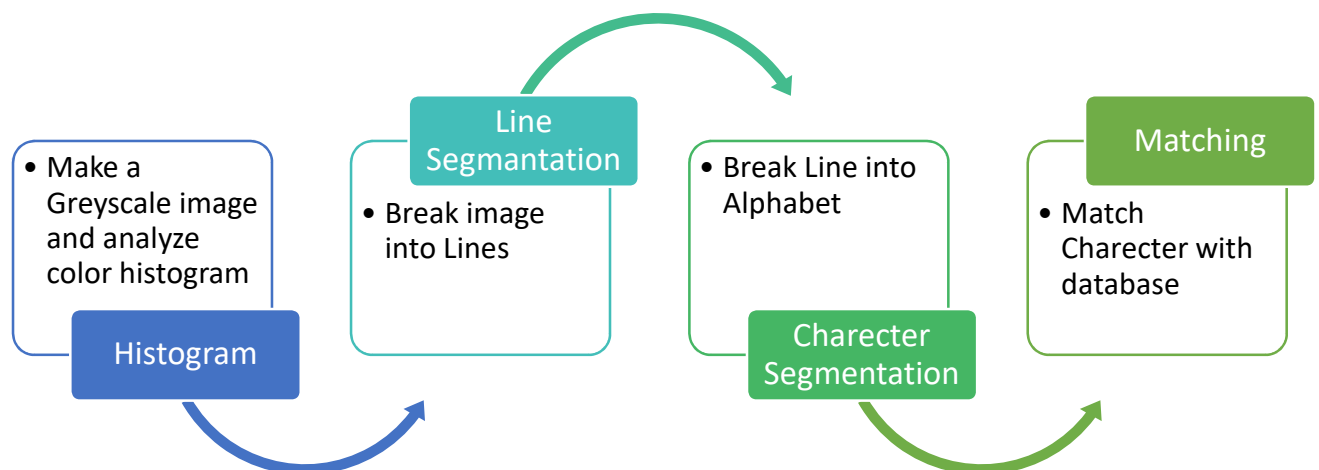
Import the package: `import os`

Documentation: <https://docs.python.org/2/library/os.html>

INSTALLATION

Install using pip command: `pip install pyimageocr`

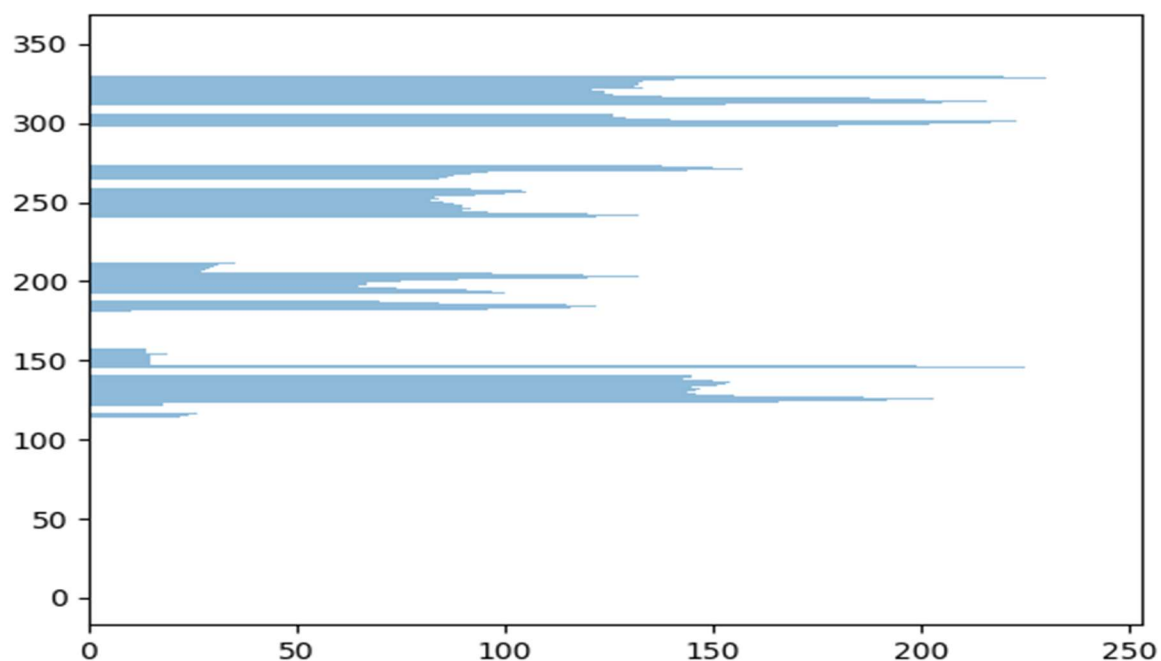
ALGORITHM



Test Image

ABCDEFGHIJKLMNOP
 QRSTUVWXYZ
 abcdefghi
 jklmnopqrstuvwxyz

Histogram Analysis



Row Segmentation

ABCDEFGHIJKLMNOP
 QRSTUVWXYZ
 abcdefghi
 jklmnopqrstuvwxyz

Character Segmentation

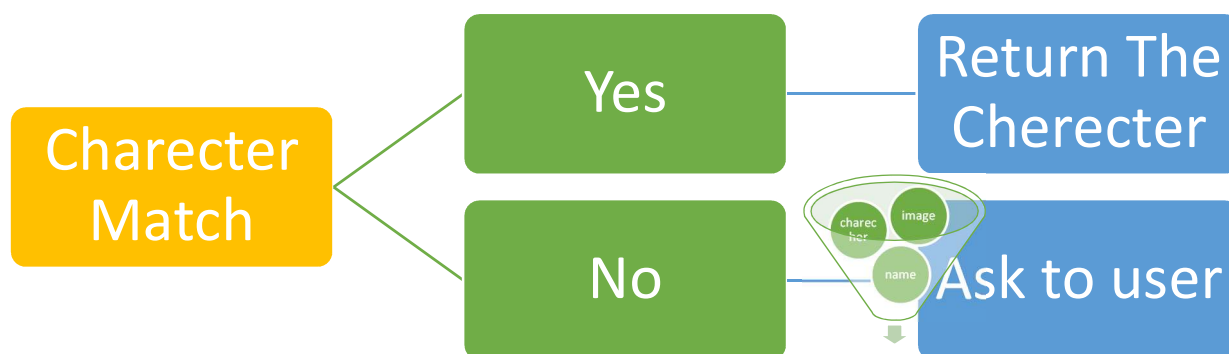
ABCDEFGHIJKLMNOP

Character Extracting

A

Character Matching

A = B 52.25 %
 C 26.74 %
 D 45.14 %
 A 95.89 %
 E 59.87 %



“PYIMAGEOCR” STRUCTURE

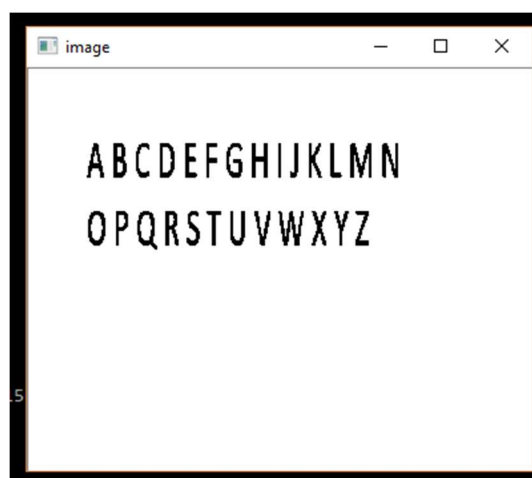
Method	Discription
OCR()	Create an instance of scanner. Input: mode: Language to be scan accuracy: accuracy percentage of scanning
getImageFormFile()	Load the image into OCR scanner Input: Image Name: name of the image to be loaded with path

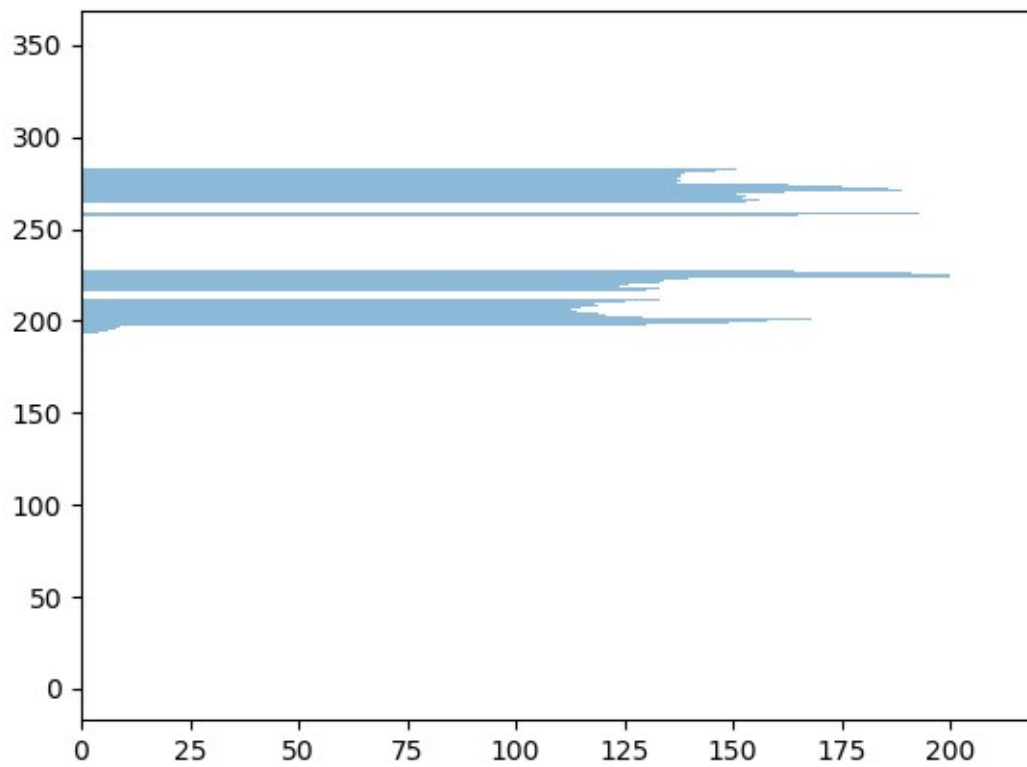
	Return: Exception image
thresoldImage()	thresoldImage(), Throsold the loaded image and generates the black and white edge filtered image. Input: None Return None Exception
imageShow()	Show the image which is loaded. Input: None Return: None

EXAMPLE CODE

```
#!/ /User/bin/python/hello.py
import pyimageocr
print('Character : Accuracy')
ocr = pyimageocr.OCR(mode='en', accurecy=70)
ocr.getImageFormFile("test_image.jpg")
ocr.thresoldImage()
ocr.getimageHistogram()
ocr.imageShow()
```

OUTPUT





```

C:\Python\python.exe
Character : Accuracy
['A', 100.0]
['B', 100.0]
['C', 100.0]
['D', 100.0]
['E', 100.0]
['F', 100.0]
['G', 100.0]
['H', 100.0]
['I', 100.0]
['J', 100.0]
['K', 100.0]
['L', 100.0]
['M', 100.0]
['N', 100.0]
['O', 100.0]
['P', 100.0]
['Q', 100.0]
['R', 100.0]
['S', 100.0]
['T', 100.0]
['U', 100.0]
['V', 100.0]
['W', 95.9228515625]
['X', 100.0]
['Y', 100.0]
['Z', 100.0]

```

REFERENCE

<https://pypi.org/project/pyimageocr>

<https://www.python.org>

<https://docs.opencv.org/master>

<https://matplotlib.org>