

# Python에서 Tesseract 사용하기 for OCR


Jan 30, 2019

## Tesseract

이미지로부터 텍스트를 인식하고, 추출하는 소프트웨어를 일반적으로 OCR이라고 한다. Tesseract는 1984~1994년에 HP 연구소에서 개발된 오픈 소스 OCR 엔진이며, 현재까지도 LSTM과 같은 딥러닝 방식을 통해 텍스트 인식률을 지속적으로 개선하고 있다. 지금부터 Python 환경에서 Tesseract를 이용하여 이미지로부터 텍스트 추출하는 방법을 소개한다.

언어에 관계없이 Tesseract를 이용하기 위해서 우선 관련 프로그램을 설치해야 한다.

[Tesseract 다운로드](#)

 GitHub, Inc. [US] | <https://github.com/tesseract-ocr/tesseract/wiki>

### Windows

Installer for Windows for Tesseract 3.05-02 and Tesseract 4.00-beta are available from [Tesseract at UB Mannheim](#). These include the training tools. Both 32-bit and 64-bit installers are available.

An installer for the **OLD version 3.02** is available for Windows from our [download](#) page. This includes the English training data. If you want to use another language, [download the appropriate training data](#), unpack it using 7-zip, and copy the .traineddata file into the 'tessdata' directory, probably `C:\Program Files\Tesseract-OCR\tessdata`.

To access tesseract-OCR from any location you may have to add the directory where the tesseract-OCR binaries are located to the Path variables, probably `C:\Program Files\Tesseract-OCR`.

Experts can also get binaries build with Visual Studio from the build artifacts of the [Appveyor Continuous Integration](#).

### Cygnwin

Released version `>= 3.02` of tesseract-ocr [are part of Cygnwin](#)

The latest version available is 4.00. Please see [announcement](#).

### MSYS2

Install tesseract-OCR:

```
pacman -S mingw-w64-{i686,x86_64}-tesseract-ocr
```

and the data files:

각자 자신의 OS 환경에 맞춰서 tesseract를 설치하면 된다. 여기서는 Windows 64비트 환경으로 진행한다.

GitHub, Inc. [US] | https://github.com/UB-Mannheim/tesseract/wiki

jump to... Pull requests Issues Marketplace Explore

UB-Mannheim / tesseract  
forked from tesseract-ocr/tesseract

Watch 64

Code Issues 2 Pull requests 2 Projects 0 Wiki Insights

## Home

Stefan Weil edited this page on 30 Oct 2018 · 45 revisions

## Tesseract at UB Mannheim

The Mannheim University Library (UB Mannheim) uses Tesseract to perform OCR of historical German newspapers ([Allgemeine Preußische Staatszeitung](#), [Deutscher Reichsanzeiger](#)). The latest results with OCR from more than 360,000 scans are available [online](#).

Normally we run Tesseract on Debian GNU Linux, but there was also the need for a Windows version. That's why we have built a Tesseract installer for Windows.

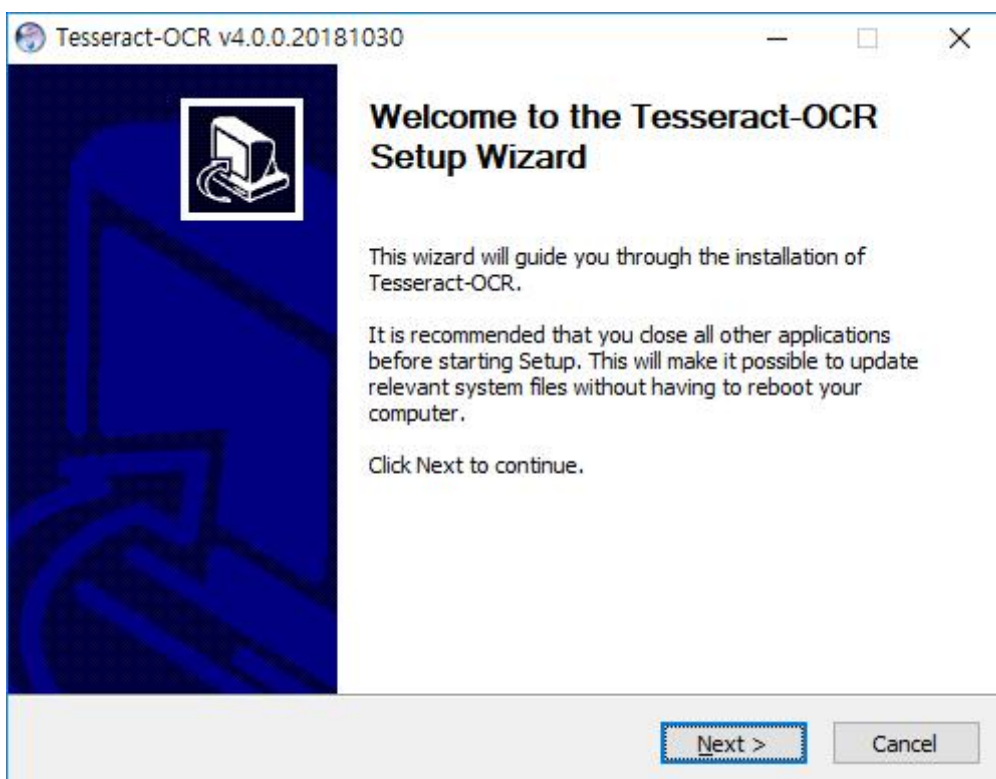
The latest installers can be downloaded here: [tesseract-ocr-setup-3.05.02-20180621.exe](#), [tesseract-ocr-w32-setup-v4.0.0.20181030.exe](#) (32 bit) and [tesseract-ocr-w64-setup-v4.0.0.20181030.exe](#) (64 bit). There are also [older versions](#) available.

In addition, we also provide [documentation](#) which was generated by Doxygen.

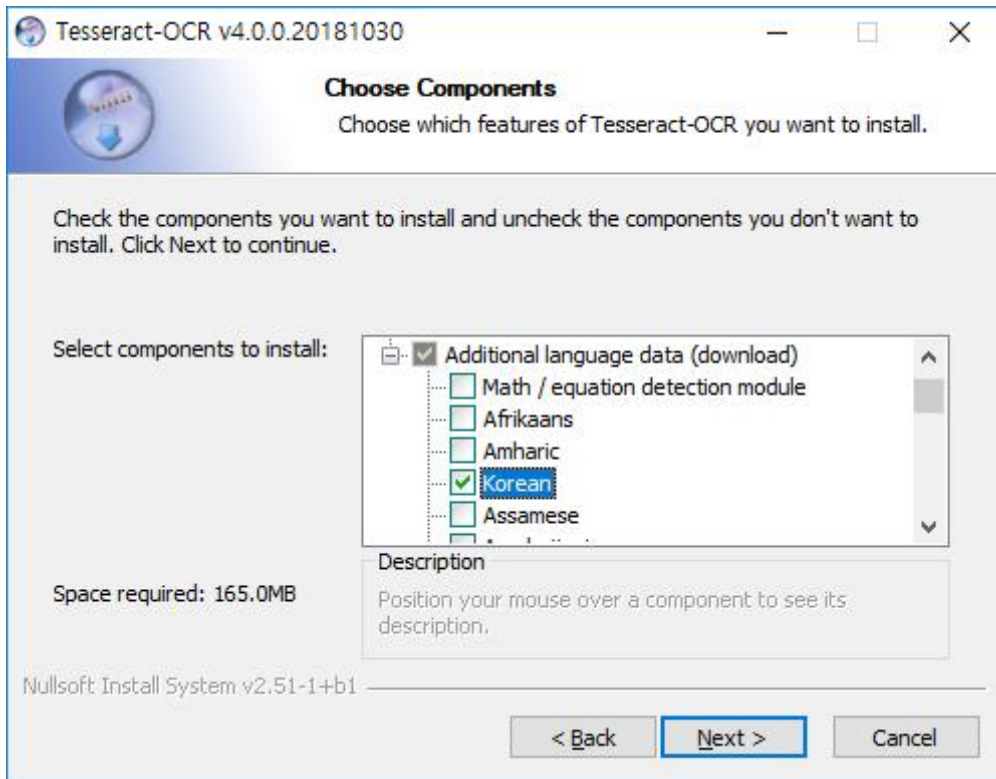
History:

- 2018-10-30 Update Tesseract 4.0.0.
- 2018-10-24 Update Tesseract 4.0.0 (RC4).
- 2018-10-14 Update Tesseract 4.0.0 (RC3).
- 2018-10-10 Update Tesseract 4.0.0 (RC2).
- 2018-10-02 Update Tesseract 4.0.0 (RC1).
- 2018-09-17 Fixed the previous 64 bit installer by adding two missing DLL files.
- 2018-09-12 Update Tesseract 4.0.0. Mainly bug fixes, see [list of commits](#). For the 64 bit installation, some executables don't work because of missing DLL files.

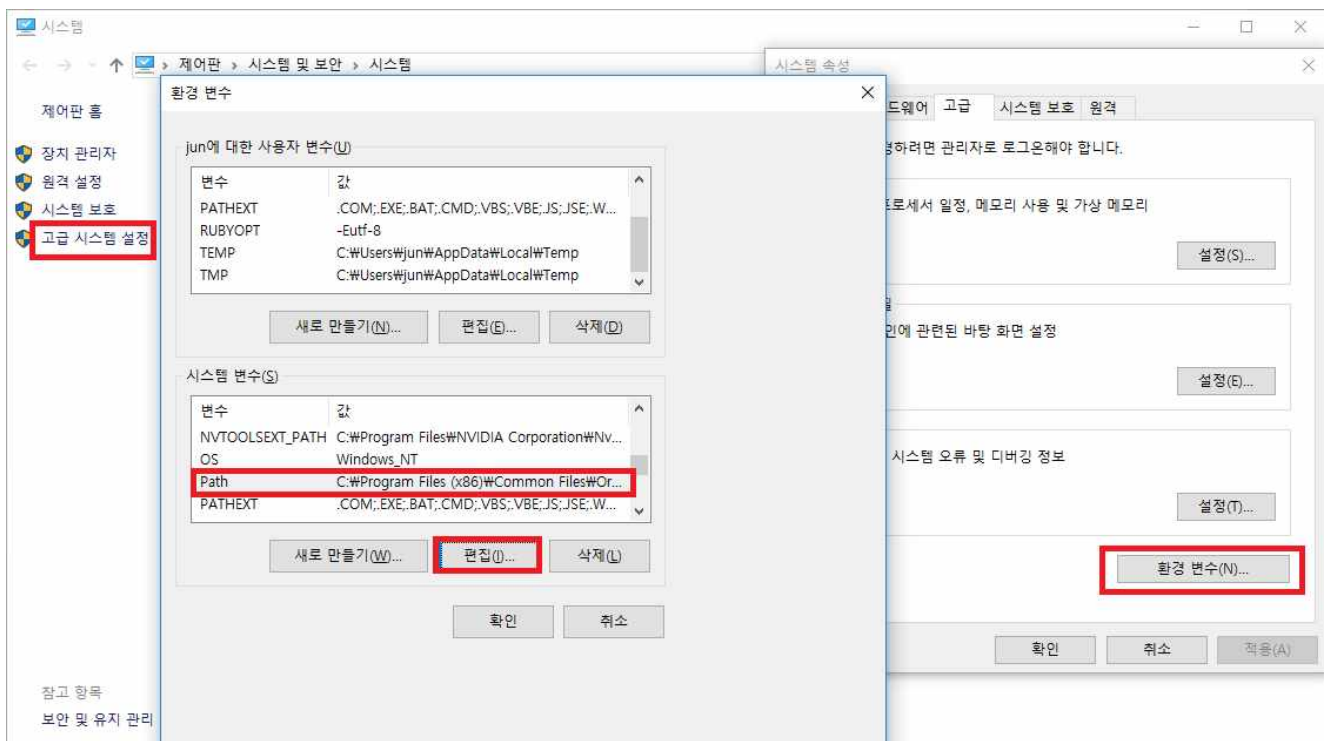
64비트 환경에서 tesseract-ocr-w64-setup-v4.0.0.20181030.exe 다운로드 후 설치를 진행한다.

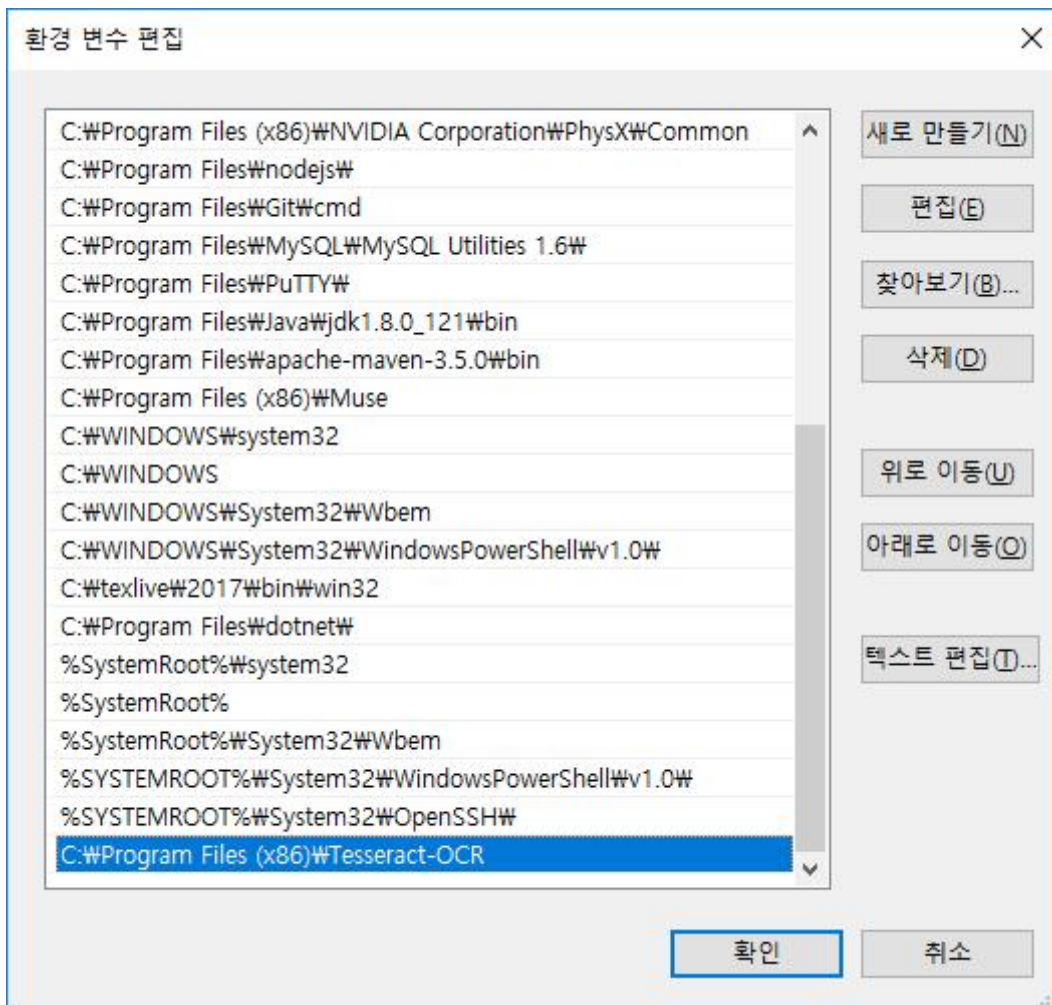


기본적으로 언어팩은 english가 선택되며, 다른 언어의 경우 추가적으로 language data를 설치해줘야 한다.



설치를 완료한 후 경로와 무관하게 tesseract를 실행하기 위해 환경변수를 등록해준다.





cmd창에서 tesseract 명령을 통해 정상적으로 설치되었음을 확인할 수 있다.

```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.17134.523]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Wjun>tesseract
Usage:
  tesseract --help | --help-extra | --version
  tesseract --list-langs
  tesseract imagename outputbase [options...] [configfile...]

OCR options:
  -l LANG[+LANG]      Specify language(s) used for OCR.
NOTE: These options must occur before any configfile.

Single options:
  --help              Show this help message.
  --help-extra        Show extra help for advanced users.
  --version           Show version information.
  --list-langs        List available languages for tesseract engine.

C:\Users\Wjun>

```

Python에서 tesseract를 사용하기 전에 직접 설치한 tesseract를 이용해 텍스트 인식 및 추출 (OCR)이 잘 되는지 테스트해 볼 수 있다.

cmd 창에서 아래 명령어를 통해 특정 이미지로부터 텍스트 추출을 수행하고, 결과를 txt 파일로 저장할 수 있다.

```
tesseract IMG_1.jpg stdout -l eng > IMG_1.txt
```

옵션 사용 시:

```
tesseract -c preserve_interword_spaces=1 IMG_5624.jpg stdout -l eng > IMG_5624.txt
```

```
tesseract --oem 1 --psm 7 IMG_5624.jpg stdout -l eng > IMG_5624.txt
```

- *OCR Engine modes(-oem):*
  - 0 - Legacy engine only.*
  - 1 - Neural nets LSTM engine only.*
  - 2 - Legacy + LSTM engines.*
  - 3 - Default, based on what is available.*
- *Page segmentation modes(-psm):*
  - 0 - Orientation and script detection (OSD) only.*
  - 1 - Automatic page segmentation with OSD.*
  - 2 - Automatic page segmentation, but no OSD, or OCR.*
  - 3 - Fully automatic page segmentation, but no OSD. (Default)*
  - 4 - Assume a single column of text of variable sizes.*
  - 5 - Assume a single uniform block of vertically aligned text.*
  - 6 - Assume a single uniform block of text.*
  - 7 - Treat the image as a single text line.*
  - 8 - Treat the image as a single word.*
  - 9 - Treat the image as a single word in a circle.*
  - 10 - Treat the image as a single character.*
  - 11 - Sparse text. Find as much text as possible in no particular order.*
  - 12 - Sparse text with OSD.*
  - 13 - Raw line. Treat the image as a single text line, bypassing hacks that are Tesseract-specific.*

여러 언어를 동시에 인식하고 싶을 경우에는 kor+eng 와 같이 옵션을 주면 된다.

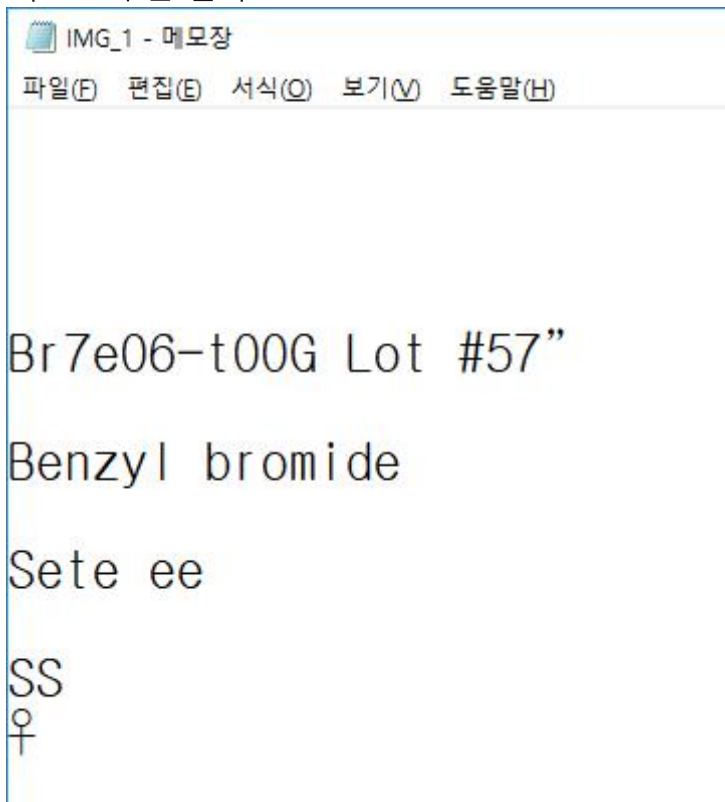
```
tesseract -c preserve_interword_spaces=1 IMG_5624.jpg stdout -l kor+eng > IMG_5624.txt
```





위의 IMG\_1.jpg 이미지로부터 텍스트를 추출하기 위해 아래와 같이 tesseract 명령을 수행한다.

텍스트 추출 결과:




텍스트 추출 결과가 아주 정확하지는 않다는 것을 확인 할 수 있다. 텍스트 추출 정확도는 이미지에 따라서 크게 차이 날 수 있다. 텍스트가 잘 정리되고 나열된 상태의 이미지라면 더 정확한 인식이 가능하다.

아래는 다른 시약병 이미지에 대한 tesseract OCR 텍스트 추출 결과이다.

IMG\_5624 - 메모장  
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)  
tetrahydrofuran  

310-4400 CAS No. 109-99-9

  
.  
ee  
.  
%  
~  
ts  
x eS in  
e We s  
t eQic NT)  
x ma NADAL)  
  
B Ad22 43 DANGER  
| i  
  
above 99.0 %  
1.406 - 1.409  
0.884 - 0.804  
  
below 0.2 %



앞선 이미지와 마찬가지로 제조사명은 추출되지 않았다. 이미지 내에 텍스트들이 다소 여러 부분에 분포되어 있기 때문에 전체 텍스트 구조를 분석하고 인식하는 과정에서 다양한 오차가 발생할 수 있다. 이와 같이 이미지에 노이즈가 많거나 Tesseract를 적용하기 전에 이미지가 제대로 사전 처리되지 않으면 성능이 크게 떨어질 수 있다.

이러한 문제를 개선할 수 있는 여러 가지 방법이 있는데, 그 중 한가지는 이미지에서 특정 텍스트 부분만 잘라내어 별도로 텍스트 추출을 수행하는 방법이다. 예를 들어, 원본 이미지에서 제조사명 부분만 잘라낸 이미지에 대해 텍스트 추출을 수행한 결과 아래와 같이 정확하게 제조사명을 출력한 것을 확인 할 수 있다.

IMG\_2 - 메모장  
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)  
DAEJUNG  
♀  


마찬가지로 제품번호가 포함된 부분만 잘라낸 이미지에 대해 추출한 결과 전체 이미지로부터 텍스트를 추출한 결과보다 결과가 개선되었음을 확인할 수 있다.



이러한 결과로부터 Tesseract를 이용해 이미지의 텍스트를 추출할 때, 텍스트 인식률을 높이기 위해서 이미지를 사전에 가공하는 절차가 포함되어야 한다는 것을 알 수 있다.

일단, 간단한 예를 통해 tesseract를 사용하는 방법과 그 결과를 확인하였으니, 이제 Python에서 tesseract를 이용한 OCR 텍스트 추출 방법을 살펴보자.

## Python Tesseract

Python에서 tesseract를 이용하기 위해 관련 모듈인 [Python-tesseract](#)를 설치해줘야 한다.

```
try:
    from PIL import Image
except ImportError:
    import Image
import pytesseract

# If you don't have tesseract executable in your PATH, include the following:
# pytesseract.pytesseract_cmd = r'<full_path_to_your_tesseract_executable>'
# Example pytesseract_cmd = r'C:\Program Files (x86)\Tesseract-OCR\tesseract'

# Simple image to string
print(pytesseract.image_to_string(Image.open('test.png')))
```

Python-tesseract는 Google의 [Tesseract-OCR Engine](#)에 대한 wrapper다.

pip를 이용한 설치:

```
pip install pytesseract
```



```
Anaconda Prompt
(base) C:\Users\jun>pip install pytesseract
Collecting pytesseract
  Downloading https://files.pythonhosted.org/packages/71/5a/d7600cad26276d991feecb27f3627ae2d0ee89aa1e3065fa4f9f1f2defb0/pytesseract-0.2.6.tar.gz (169kB)
    100% |#####| 174kB 410kB/s
Requirement already satisfied: Pillow in c:\Users\jun\anaconda3\lib\site-packages (from pytesseract) (5.0.0)
Building wheels for collected packages: pytesseract
  Building wheel for pytesseract (setup.py) ... done
  Stored in directory: C:\Users\jun\AppData\Local\pip\Cache\wheels\5\90\56\ab7b652592da86821293f7cad1c554aa376a0d57ce414d0a0
Successfully built pytesseract
Installing collected packages: pytesseract
Successfully installed pytesseract-0.2.6
(base) C:\Users\jun>
```

python에서 tesseract를 사용하는 방법은 아주 간단하다. pip를 통해 설치한 pytesseract를 import하고, 앞서 설치한 tesseract 경로만 명시적으로 등록해주기만 하면 된다.

이미지로부터 텍스트를 추출하는 함수는 pytesseract.image\_to\_string() 이다.

```
import pytesseract
pytesseract.pytesseract.tesseract_cmd = r'C:\Program Files (x86)\Tesseract-OCR\Tesseract.exe'

print(pytesseract.image_to_string('img.png'))
```

출력결과:

```
C:\Users\jun\Anaconda3\python.exe C:/Users/jun/PycharmProjects/Tesseract_Sample/main.py
S: 8178061006 Lot 519

Benzyl bromide

Sete ae

SS

Process finished with exit code 0
```

옵션을 주고 싶을 경우에는 아래와 같이 파라미터를 전달하여 함수를 호출 하면 된다.

```
pytesseract.image_to_string('img.png', lang='eng', config='--psm 1 -c preserve_in
```

더 자세한 기능 사용방법은 [Python-tesseract github](#)에서 확인 가능하다.

## Quickstart

```
try:
    from PIL import Image
except ImportError:
    import Image
import pytesseract

# If you don't have tesseract executable in your PATH, include the following:
pytesseract.pytesseract.tesseract_cmd = r'<full_path_to_your_tesseract_executable>'
# Example tesseract_cmd = r'C:\Program Files (x86)\Tesseract-OCR\tesseract'

# Simple image to string
print(pytesseract.image_to_string(Image.open('test.png')))

# French text image to string
print(pytesseract.image_to_string(Image.open('test-european.jpg'), lang='fra'))

# In order to bypass the image conversions of pytesseract, just use relative or absolute image path
# NOTE: In this case you should provide tesseract supported images or tesseract will return error
print(pytesseract.image_to_string('test.png'))

# Batch processing with a single file containing the list of multiple image file paths
print(pytesseract.image_to_string('images.txt'))

# Get bounding box estimates
print(pytesseract.image_to_boxes(Image.open('test.png')))

# Get verbose data including boxes, confidences, line and page numbers
print(pytesseract.image_to_data(Image.open('test.png')))

# Get information about orientation and script detection
print(pytesseract.image_to_osd(Image.open('test.png')))

# Get a searchable PDF
pdf = pytesseract.image_to_pdf_or_hocr('test.png', extension='pdf')

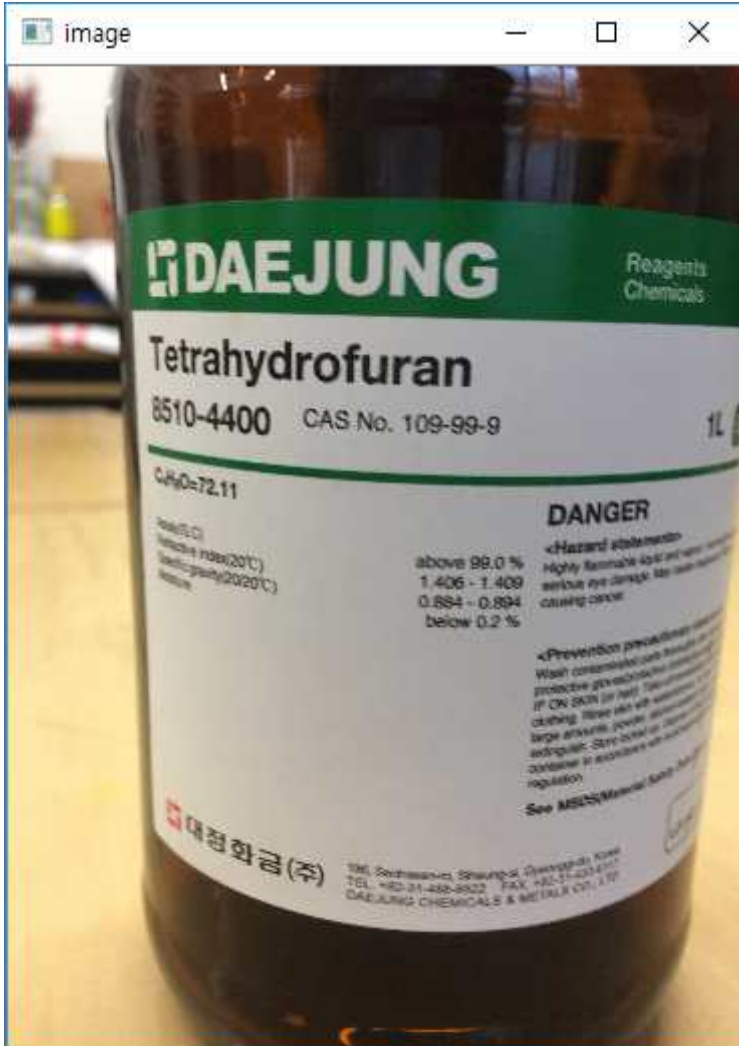
# Get HOCR output
hocr = pytesseract.image_to_pdf_or_hocr('test.png', extension='hocr')
```

## OCR 정확도 개선

Tesseract는 배경으로부터 전경 텍스트가 깨끗히 세분화가있을 때 가장 효과적인 결과를 얻을 수 있다. 따라서, 보다 높은 텍스트 인식률을 달성하기 위해서는 OCR 수행에 앞서 반드시 원본 이미지에 대한 전처리 과정이 포함되어야 한다.

다음은 OpenCV를 이용하여 Tesseract OCR 인식률을 높이기 위한 전처리 방법들 중 일부를 소개한다.

원본 이미지:

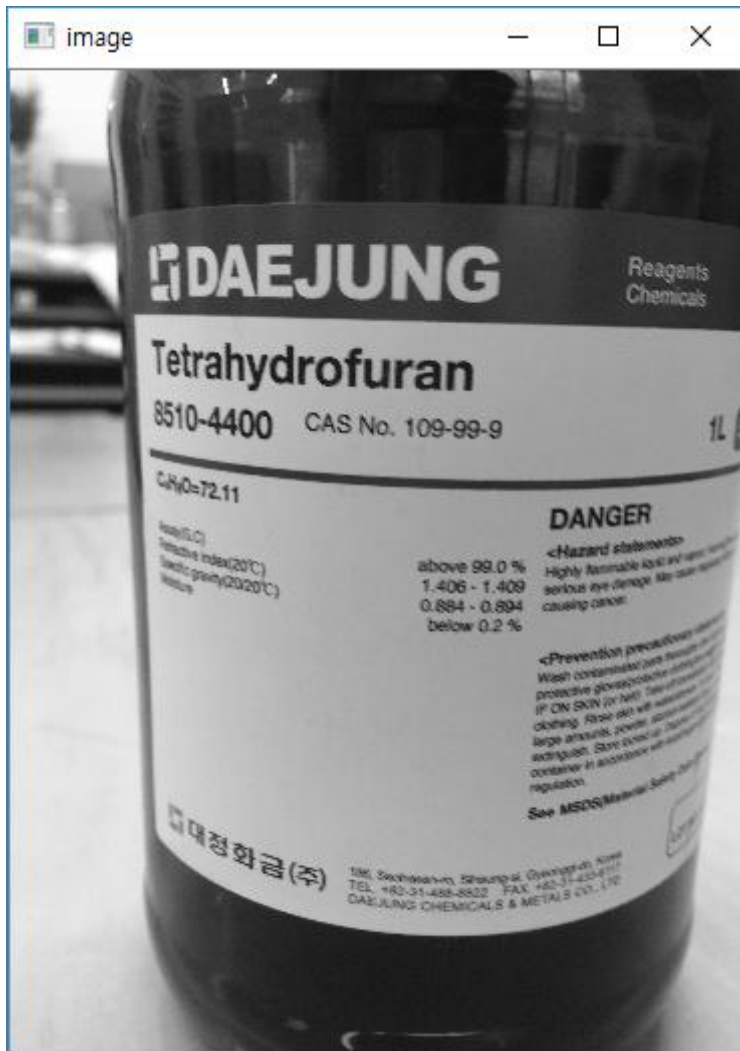


```
import cv2
```

```
image = cv2.imread('IMG_1.jpg')
```

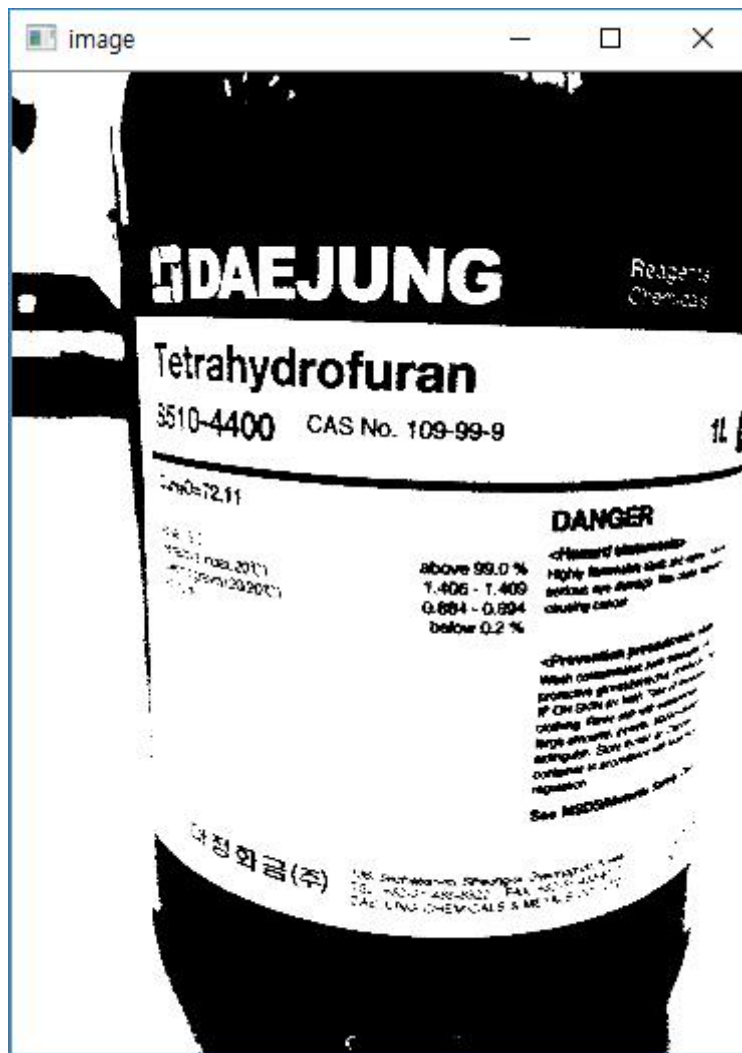
- 그레이 스케일로 변환하기

```
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
```



- 배경에서 전경 텍스트를 분할하기 위해 임계값을 사용한다. 플래그 값에 대한 자세한 내용은 공식 [OpenCV 설명서](#)를 참조하자. 이러한 임계값 사용은 회색 배경 위에 겹쳐진 검은색 텍스트를 읽는 데 유용하다.

```
gray = cv2.threshold(gray, 0, 255, cv2.THRESH_BINARY | cv2.THRESH_OTSU)[1]
```



임계값 사용 대신, 블러(Blur) 처리를 적용될 수 있다. medianBlur를 적용하면 이미지의 노이즈를 줄일 수 있다.

```
gray = cv2.medianBlur(gray, 10)
```



## References

- [pyimagesearch - Using Tesseract OCR with Python](#)
- [pyimagesearch - OpenCV OCR and text recognition with Tesseract](#)
- [EAST Github](#)
- [EAST paper - EAST: An Efficient and Accurate Scene Text Detector](#)

JY Kang's Blog

JY Kang's Blog

[msmapark2@gmail.com](mailto:msmapark2@gmail.com)

 [junyoung-jamong](#)  
 [junyoung](#)

JY Kang's technical blog about computer science, data mining, machine learning, deep learning...



