

## ▼ PyTorch YOLOv3

- YOLO는 객체 검출(Object Detection)의 대표적인 방법 중 하나
- 워싱턴대 대학원생이던 Joseph Redmon이 개발했으며, C언어로 구현
- 파이토치 버전 테스트

YOLOv3 : <https://pjreddie.com/darknet/yolo/>

PyTorch-YOLOv3 : <https://github.com/eriklindernoren/PyTorch-YOLOv3>

```
import warnings
warnings.filterwarnings('ignore')
```

## ▼ I. YOLOv3 설치

### ▼ 1) PyTorch YOLOv3 복사

```
!git clone https://github.com/eriklindernoren/PyTorch-YOLOv3
```



```
Cloning into 'PyTorch-YOLOv3'...
remote: Enumerating objects: 724, done.
remote: Total 724 (delta 0), reused 0 (delta 0), pack-reused 724
Receiving objects: 100% (724/724), 16.18 MiB | 46.53 MiB/s, done.
Resolving deltas: 100% (411/411), done.
```

### ▼ 2) 디렉토리 이동

```
%cd PyTorch-YOLOv3/
```

```
/content/PyTorch-YOLOv3
```

### ▼ 3) 파일 목록

```
!!ls -l
```

```
total 104
drwxr-xr-x 2 root root 4096 Jan  5 06:25 assets
drwxr-xr-x 2 root root 4096 Jan  5 06:25 config
drwxr-xr-x 4 root root 4096 Jan  5 06:25 data
-rw-r--r-- 1 root root 5354 Jan  5 06:25 detect.py
-rw-r--r-- 1 root root 35127 Jan  5 06:25 LICENSE
```

```
-rw-r--r-- 1 root root 14899 Jan  5 06:25 models.py
-rw-r--r-- 1 root root  6437 Jan  5 06:25 README.md
-rw-r--r-- 1 root root    90 Jan  5 06:25 requirements.txt
-rw-r--r-- 1 root root  4004 Jan  5 06:25 test.py
-rw-r--r-- 1 root root  6732 Jan  5 06:25 train.py
drwxr-xr-x 2 root root  4096 Jan  5 06:25 utils
drwxr-xr-x 2 root root  4096 Jan  5 06:25 weights
```

## ▼ 4) Package 설치

```
!pip install -r requirements.txt
```

```
Requirement already satisfied: numpy in /usr/local/lib/python3.6/dist-packages (from -r re
Requirement already satisfied: torch>=1.0 in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: torchvision in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: matplotlib in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: tensorflow in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: tensorboard in /usr/local/lib/python3.6/dist-packages (from
Collecting terminaltables
  Downloading https://files.pythonhosted.org/packages/9b/c4/4a21174f32f8a7e1104798c445dacc
Requirement already satisfied: pillow in /usr/local/lib/python3.6/dist-packages (from -r r
Requirement already satisfied: tqdm in /usr/local/lib/python3.6/dist-packages (from -r rec
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: future in /usr/local/lib/python3.6/dist-packages (from torc
Requirement already satisfied: dataclasses in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.6/dist-packa
Requirement already satisfied: cyclopy>=0.10 in /usr/local/lib/python3.6/dist-packages (fr
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: h5py~2.10.0 in /usr/local/lib/python3.6/dist-packages (fr
Requirement already satisfied: keras-preprocessing~1.1.2 in /usr/local/lib/python3.6/dist
Requirement already satisfied: termcolor~1.1.0 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: grpcio~1.32.0 in /usr/local/lib/python3.6/dist-packages (f
Requirement already satisfied: astunparse~1.6.3 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: opt-einsum~3.3.0 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: wheel~0.35 in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: google-pasta~0.2 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: absl-py~0.10 in /usr/local/lib/python3.6/dist-packages (fr
Requirement already satisfied: flatbuffers~1.12.0 in /usr/local/lib/python3.6/dist-packag
Requirement already satisfied: six~1.15.0 in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: protobuf>=3.9.2 in /usr/local/lib/python3.6/dist-packages (
Requirement already satisfied: gast==0.3.3 in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: tensorflow-estimator<2.5.0,>=2.4.0rc0 in /usr/local/lib/pyt
Requirement already satisfied: wrapt~1.12.1 in /usr/local/lib/python3.6/dist-packages (fr
Requirement already satisfied: google-auth-oauthlib<0.5,>=0.4.1 in /usr/local/lib/python3.
Requirement already satisfied: setuptools>=41.0.0 in /usr/local/lib/python3.6/dist-package
Requirement already satisfied: werkzeug>=0.11.15 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: google-auth<2,>=1.6.3 in /usr/local/lib/python3.6/dist-pack
Requirement already satisfied: requests<3,>=2.21.0 in /usr/local/lib/python3.6/dist-packag
Requirement already satisfied: tensorboard-plugin-wit>=1.6.0 in /usr/local/lib/python3.6/c
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.6/dist-packages (
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.6/dist-p
Requirement already satisfied: rsa<5,>=3.1.4; python_version >= "3" in /usr/local/lib/pyth
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.6/dist-pack
Requirement already satisfied: cachetools<5.0,>=2.0.0 in /usr/local/lib/python3.6/dist-pac
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/local/lib/p
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.6/dist-packages (fr
```

```
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: importlib-metadata; python_version < "3.8" in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: pyasn1>=0.1.3 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.6/dist-packages (from importlib-metadata)
Building wheels for collected packages: terminaltables
  Building wheel for terminaltables (setup.py) ... done
  Created wheel for terminaltables: filename=terminaltables-3.1.0-cp36-none-any.whl size=11511 sha256=11511
  Stored in directory: /root/.cache/pip/wheels/30/6b/50/6c75775b681fb36cdfac7f19799888ef9c
Successfully built terminaltables
Installing collected packages: terminaltables
Successfully installed terminaltables-3.1.0
```

## 5) 가중치 폴더로 이동

```
%cd weights
```

```
/content/PyTorch-YOLOv3/weights
```

## 6) Google Drive Mount

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

## 7) 'yolo\_weight.zip' 압축해제

```
!ls -l '/content/drive/My Drive/Colab Notebooks/datasets/yolo_weight.zip'
```

```
-rw----- 1 root root 414658234 Oct 20 04:18 '/content/drive/My Drive/Colab Notebooks/datasets/yolo_weight.zip'
```

```
!unzip /content/drive/My Drive/Colab Notebooks/datasets/yolo_weight.zip
```

```
Archive: /content/drive/My Drive/Colab Notebooks/datasets/yolo_weight.zip
  inflating: darknet53.conv.74
  inflating: yolov3.weights
  inflating: yolov3-tiny.weights
```

## 8) YOLOv3 가중치 다운로드

- 약 60분

```
# !bash download_weights.sh
```

## ▼ II. 샘플 이미지 테스트

- 샘플 이미지 객체 검출

### ▼ 1) 상위 디렉토리 이동

```
%cd ..
```

```
/content/PyTorch-YOLOv3
```

### ▼ 2) 샘플 이미지 폴더 표시

```
!ls -l /content/PyTorch-YOLOv3/data/samples
```

```
total 1312
-rw-r--r-- 1 root root 163759 Jan  5 06:25 dog.jpg
-rw-r--r-- 1 root root 141886 Jan  5 06:25 eagle.jpg
-rw-r--r-- 1 root root 113880 Jan  5 06:25 field.jpg
-rw-r--r-- 1 root root 382965 Jan  5 06:25 giraffe.jpg
-rw-r--r-- 1 root root 133495 Jan  5 06:25 herd_of_horses.jpg
-rw-r--r-- 1 root root 126867 Jan  5 06:25 messi.jpg
-rw-r--r-- 1 root root  78771 Jan  5 06:25 person.jpg
-rw-r--r-- 1 root root  84988 Jan  5 06:25 room.jpg
-rw-r--r-- 1 root root 102770 Jan  5 06:25 street.jpg
```

### ▼ 3) 샘플 폴더 이미지 객체 검출 수행

```
!python detect.py --image_folder 'data/samples'
```

```
Namespace(batch_size=1, checkpoint_model=None, class_path='data/coco.names', conf_thres=0.8,
```

```
Performing object detection:
```

```
+ Batch 0, Inference Time: 0:00:00.243829
+ Batch 1, Inference Time: 0:00:00.064177
+ Batch 2, Inference Time: 0:00:00.058689
+ Batch 3, Inference Time: 0:00:00.057279
+ Batch 4, Inference Time: 0:00:00.060706
+ Batch 5, Inference Time: 0:00:00.078001
+ Batch 6, Inference Time: 0:00:00.043775
+ Batch 7, Inference Time: 0:00:00.043503
+ Batch 8, Inference Time: 0:00:00.047030
```

```
Saving images:
```

```
(0) Image: 'data/samples/dog.jpg'
```

```

+ Label: dog, Conf: 0.99335
+ Label: bicycle, Conf: 0.99981
+ Label: truck, Conf: 0.94229
(1) Image: 'data/samples/eagle.jpg'
+ Label: bird, Conf: 0.99703
(2) Image: 'data/samples/field.jpg'
+ Label: person, Conf: 0.99996
+ Label: horse, Conf: 0.99977
+ Label: dog, Conf: 0.99409
(3) Image: 'data/samples/giraffe.jpg'
+ Label: giraffe, Conf: 0.99959
+ Label: zebra, Conf: 0.97958
(4) Image: 'data/samples/herd_of_horses.jpg'
+ Label: horse, Conf: 0.99459
+ Label: horse, Conf: 0.99352
+ Label: horse, Conf: 0.96845
+ Label: horse, Conf: 0.99478
(5) Image: 'data/samples/messi.jpg'
+ Label: person, Conf: 0.99993
+ Label: person, Conf: 0.99984
+ Label: person, Conf: 0.99996
(6) Image: 'data/samples/person.jpg'
+ Label: person, Conf: 0.99883
+ Label: dog, Conf: 0.99275
(7) Image: 'data/samples/room.jpg'
+ Label: chair, Conf: 0.99906
+ Label: chair, Conf: 0.96942
+ Label: clock, Conf: 0.99971
(8) Image: 'data/samples/street.jpg'
+ Label: car, Conf: 0.99977
+ Label: car, Conf: 0.99402
+ Label: car, Conf: 0.99841
+ Label: car, Conf: 0.99785
+ Label: car, Conf: 0.97907
+ Label: car, Conf: 0.95370
+ Label: traffic light, Conf: 0.99995
+ Label: car, Conf: 0.62254

```

#### ▼ 4) 객체 검출 결과 output 폴더에 저장

```
!ls -l output
```

```

total 2572
-rw-r--r-- 1 root root 329183 Jan  5 06:25 dog.png
-rw-r--r-- 1 root root 304230 Jan  5 06:25 eagle.png
-rw-r--r-- 1 root root 298366 Jan  5 06:25 field.png
-rw-r--r-- 1 root root 382965 Jan  5 06:25 giraffe.png
-rw-r--r-- 1 root root 264125 Jan  5 06:25 herd_of_horses.png
-rw-r--r-- 1 root root 244067 Jan  5 06:25 messi.png
-rw-r--r-- 1 root root 183522 Jan  5 06:25 person.png
-rw-r--r-- 1 root root 315045 Jan  5 06:25 room.png
-rw-r--r-- 1 root root 297985 Jan  5 06:25 street.png

```

#### ▼ 5) 객체 검출 결과 확인

```
from IPython.display import Image

Image('/content/PyTorch-YOLOv3/output/person.png')

Image('/content/PyTorch-YOLOv3/output/giraffe.png')
```

## ▼ II. 구글 드라이브 이미지

- 직접 선택한 사진으로 객체 검출
- 구글 드라이브에 폴더를 생성하고, 거기에 원하는 사진을 업로드
- 폴더 이름을 파라미터로 넣고 실행

### ▼ 1) Google Drive Mount

```
from google.colab import drive

drive.mount('/content/drive')
```

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/c

### ▼ 2) 객체 검출 수행

```
!python detect.py --image_folder '/content/drive/My Drive/Colab Notebooks/datasets/image'
```

Namespace(batch\_size=1, checkpoint\_model=None, class\_path='data/coco.names', conf\_thres=0.8,

Performing object detection:

```
+ Batch 0, Inference Time: 0:00:00.379619
+ Batch 1, Inference Time: 0:00:00.314112
+ Batch 2, Inference Time: 0:00:00.432718
```

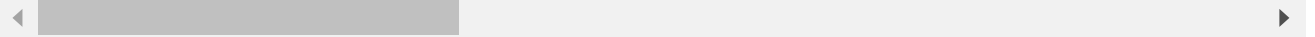
Saving images:

```
(0) Image: '/content/drive/My Drive/Colab Notebooks/datasets/image/001.jpg'
+ Label: person, Conf: 0.99994
+ Label: person, Conf: 0.99988
+ Label: person, Conf: 0.99993
+ Label: person, Conf: 0.99940
+ Label: person, Conf: 0.99916
+ Label: person, Conf: 0.99531
(1) Image: '/content/drive/My Drive/Colab Notebooks/datasets/image/002.jpg'
+ Label: person, Conf: 0.99769
+ Label: person, Conf: 0.99950
+ Label: person, Conf: 0.99978
+ Label: person, Conf: 0.99993
+ Label: person, Conf: 0.99621
```

```

+ Label: person, Conf: 0.99996
+ Label: person, Conf: 0.99999
+ Label: bus, Conf: 0.92784
+ Label: person, Conf: 0.99963
+ Label: person, Conf: 0.99903
+ Label: person, Conf: 0.99958
+ Label: person, Conf: 0.99881
(2) Image: '/content/drive/My Drive/Colab Notebooks/datasets/image/003.jpg'
+ Label: car, Conf: 0.99809
+ Label: car, Conf: 0.99508
+ Label: car, Conf: 0.97759
+ Label: car, Conf: 0.97114
+ Label: car, Conf: 0.96552
+ Label: car, Conf: 0.76919
+ Label: car, Conf: 0.90857
+ Label: car, Conf: 0.82868
+ Label: car, Conf: 0.68758

```



### 3) 객체 검출 결과 확인

Image( '/content/PyTorch-YOLOv3/output/001.png' )

Image( '/content/PyTorch-YOLOv3/output/002.png' )

Image( '/content/PyTorch-YOLOv3/output/003.png' )

## 출처

### < 인공지능 개발자 모임 >

- 페이스북 그룹에 가입하시면 인공지능에 대한 최신 정보를 쉽게 받으실 수 있습니다.
- <https://www.facebook.com/groups/AIDevKr/>

