

YOLO V3

Yolo v3 - train (0)

먼저, VOC 데이터로 train 잘 되는지 테스트함

```
./darknet detector train cfg/voc.data cfg/yolov3-voc.cfg darknet53.conv.74
```

- 학습 시, 과적합을 막는 세팅을 따로 해야함. 모든 값이 nan이 나와도 계속 학습되고 있음.
- Yolo 학습 자체는 잘 진행되고 있음을 확인.

```
./darknet detector train cfg/voc.data cfg/yolov3-voc.cfg darknet53.conv.74
Resizing
608
Loaded: 0.021483 seconds
Region 82 Avg IOU: 0.000000, Class: 0.000000, Obj: 0.000000, No Obj: 0.000000, .5R: 0.000000, .75R: 0.000000, count: 1
Region 94 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: 0.001644, .5R: -nan, .75R: -nan, count: 0
Region 106 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: 0.000037, .5R: -nan, .75R: -nan, count: 0
961: 628946752.000000, 17382625280.000000 avg, 0.000853 rate, 75.710204 seconds, 961 images
Loaded: 0.000067 seconds
Region 82 Avg IOU: 0.000000, Class: 1.000000, Obj: 0.000000, No Obj: 0.333333, .5R: 0.000000, .75R: 0.000000, count: 2
Region 94 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: 0.000842, .5R: -nan, .75R: -nan, count: 0
Region 106 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: 0.000025, .5R: -nan, .75R: -nan, count: 0
962: 115137339392.000000, 27158095872.000000 avg, 0.000856 rate, 75.212312 seconds, 962 images
Loaded: 0.000068 seconds
Region 82 Avg IOU: 0.000000, Class: 0.000000, Obj: 1.000000, No Obj: 0.333333, .5R: 0.000000, .75R: 0.000000, count: 1
Region 94 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: 0.001059, .5R: -nan, .75R: -nan, count: 0
Region 106 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: 0.000018, .5R: -nan, .75R: -nan, count: 0
963: 378535231172936136524824576.000000, 37853523578462215495221248.000000 avg, 0.000860 rate, 75.195539 seconds, 963 images
Loaded: 0.000079 seconds
Region 82 Avg IOU: -nan, Class: nan, Obj: nan, No Obj: nan, .5R: 0.000000, .75R: 0.000000, count: 1
Region 94 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: nan, .5R: -nan, .75R: -nan, count: 0
Region 106 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: nan, .5R: -nan, .75R: -nan, count: 0
964: -nan, -nan avg, 0.000864 rate, 75.158733 seconds, 964 images
Loaded: 0.000098 seconds
Region 82 Avg IOU: -nan, Class: nan, Obj: nan, No Obj: nan, .5R: 0.000000, .75R: 0.000000, count: 1
Region 94 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: nan, .5R: -nan, .75R: -nan, count: 0
Region 106 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: nan, .5R: -nan, .75R: -nan, count: 0
965: -nan, -nan avg, 0.000867 rate, 75.384323 seconds, 965 images
```

964번째부터 모든 값이 nan이 나오기 시작

```
./darknet detector train cfg/voc.data cfg/yolov3-voc.cfg darknet53.conv.74
Region 94 Avg IOU: -nan, Class: nan, Obj: nan, No Obj: nan, .5R: 0.000000, .75R: 0.000000, count: 1
Region 106 Avg IOU: -nan, Class: nan, Obj: nan, No Obj: nan, .5R: 0.000000, .75R: 0.000000, count: 1
1547: -nan, -nan avg, 0.001000 rate, 65.928801 seconds, 1547 images
Loaded: 0.000073 seconds
Region 82 Avg IOU: -nan, Class: nan, Obj: nan, No Obj: nan, .5R: 0.000000, .75R: 0.000000, count: 1
Region 94 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: nan, .5R: -nan, .75R: -nan, count: 0
Region 106 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: nan, .5R: -nan, .75R: -nan, count: 0
1548: -nan, -nan avg, 0.001000 rate, 65.925677 seconds, 1548 images
Loaded: 0.000075 seconds
Region 82 Avg IOU: -nan, Class: nan, Obj: nan, No Obj: nan, .5R: 0.000000, .75R: 0.000000, count: 2
Region 94 Avg IOU: -nan, Class: nan, Obj: nan, No Obj: nan, .5R: 0.000000, .75R: 0.000000, count: 1
Region 106 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: nan, .5R: -nan, .75R: -nan, count: 0
1549: -nan, -nan avg, 0.001000 rate, 65.931852 seconds, 1549 images
Loaded: 0.000069 seconds
Region 82 Avg IOU: -nan, Class: nan, Obj: nan, No Obj: nan, .5R: 0.000000, .75R: 0.000000, count: 1
Region 94 Avg IOU: -nan, Class: nan, Obj: nan, No Obj: nan, .5R: 0.000000, .75R: 0.000000, count: 1
Region 106 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: nan, .5R: -nan, .75R: -nan, count: 0
1550: -nan, -nan avg, 0.001000 rate, 69.222363 seconds, 1550 images
Resizing
544
Loaded: 0.019853 seconds
Region 82 Avg IOU: -nan, Class: nan, Obj: nan, No Obj: nan, .5R: 0.000000, .75R: 0.000000, count: 2
Region 94 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: nan, .5R: -nan, .75R: -nan, count: 0
Region 106 Avg IOU: -nan, Class: -nan, Obj: -nan, No Obj: nan, .5R: -nan, .75R: -nan, count: 0
1551: -nan, -nan avg, 0.001000 rate, 59.620458 seconds, 1551 images
Loaded: 0.000064 seconds
```

Train 24시간 경과 후

Yolo v3 - train (1)

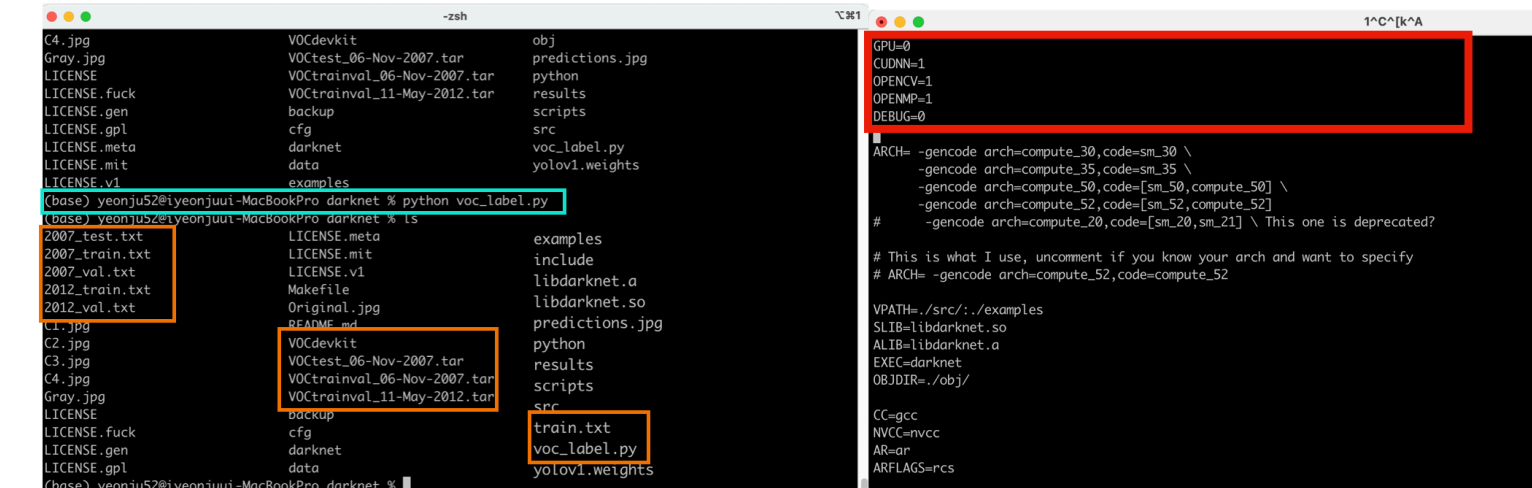
문제 발생 - 해결 시도

- (문제) Yolo v3 train 시도: (800 batch = 8시간)
-> Too Long !
 - (가정) Makefile에서 GPU=1로 설정했음에도 불구하고 GPU가 활성화가 안 된 걸까? Why?
 - (해결1) MakeFile과 Make명령어에 대한 오해가 있었음
 - Make 명령어 후, ./darknet (실행파일) 생성됨
 - MakeFile 바꾼 설정을 적용하기 위해선,
Make 명령어 실행 전에 MakeFile을 수정해야함.
 - (해결2) CUDA arch & gencode를 nvidia 버전에 매칭해야함
<https://arnon.dk/matching-sm-architectures-arch-and-gencode-for-various-nvidia-cards/>
- (현버전: `user ➡ nvidia-smi --query | fgrep 'Product Name'`
`Product Name : NVIDIA GeForce RTX 3090`)

Setting & Get Pascal VOC Data

yolo v1 설치 후 데이터 처리

- train dataset (15680개): 2007 (train, val), 2012 (train, val)
- test dataset (4952개): 2007 (test)
- Makefile: GPU=1 -> GPU = 0으로 변경 (개인컴이므로 ! 서버컴에서는 GPU=1로 돌려야 함)



(좌) python voc_label.py (우) Makefile

[발표자료 230214 1p] 오개념

```
if(avg_loss == -1) avg_loss = loss;
avg_loss = avg_loss*.9 + loss*.1;
printf("%ld, %.3f: %f, %f avg, %f rate, %lf seconds, %ld images\n", get_current_batch(net), (float)(*net.seen),
free_data(train);
if(get_current_batch(net)%100 == 0){
    char buff[256];
    sprintf(buff, "%s/%s_batch_%ld.weights", backup_directory, base, get_current_batch(net));
    save_weights(net, buff);
}
if(*net.seen/N > epoch){
    epoch = *net.seen/N;
    char buff[256];
    sprintf(buff, "%s/%s_%d.weights", backup_directory, base, epoch);
    save_weights(net, buff);
}
```

batch라고 생각한 이유 (iteration, epoch x)

Yolo v3 - train (2)

Train을 위한 추가 설정

- (추가) yolov3.cfg
 - Test -> Train 모드로 변경 시, 수정 필요

```
[net]
# Testing
# batch=1
# subdivisions=1
# Training
batch=64
subdivisions=16
width=608
height=608
channels=3
momentum=0.9
decay=0.0005
angle=0
saturation = 1.5
exposure = 1.5
hue=.1

learning_rate=0.001
burn_in=1000
max_batches = 500200
policy=steps
steps=400000,450000
scales=.1,.1

"yolov3.cfg" 789L, 8342B
```

LICENSE	LICENSE.gen	LICENSE.meta	LICENSE.v1	README.md	data	include	scripts
LICENSE.fuck	LICENSE.gpl	LICENSE.mit	Makefile	cfg	examples	python	src

Make 명령어 실행 전 모습



```
(base) yeonju52@yeonjuui-MacBookPro yolov3 % ls
LICENSE          LICENSE.meta     README.md        examples         results
LICENSE.fuck     LICENSE.mit      backup           include          scripts
LICENSE.gen      LICENSE.v1       cfg              obj              src
LICENSE.gpl      Makefile         data             python
```

Make 명령어 실행 후 모습



```
(base) yeonju52@yeonjuui-MacBookPro darknet % ls
2007_test.txt    LICENSE.meta     examples
2007_train.txt   LICENSE.mit      include
2007_val.txt     LICENSE.v1       libdarknet.a
2012_train.txt   Makefile         libdarknet.so
2012_val.txt     Original.jpg     obj
C1.jpg           README.md        predictions.jpg
C2.jpg           VOCdevkit        python
C3.jpg           VOCtest_06-Nov-2007.tar results
C4.jpg           VOCtrainval_06-Nov-2007.tar scripts
Gray.jpg         VOCtrainval_11-May-2012.tar src
LICENSE          backup           voc_label.py
LICENSE.fuck     cfg              yolov1.weights
LICENSE.gen      darknet
LICENSE.gpl      data
(base) yeonju52@yeonjuui-MacBookPro darknet %
```

데이터 세팅 및 ./darknet 실행 후 모습

OpenCV 설치

make 명령어 실행 전

- **sudo apt-get install g++**
- **sudo apt-get install cmake**
- **기존 설치된 패키지 업그레이드**
 - **sudo apt-get update**
 - **sudo apt-get upgrade**
- **opencv 설치 방법 2개 중 부가 패키지 일일히 설치**

```
1026 sudo apt-get install build-essential cmake
1027 sudo apt-get install pkg-config
1028 sudo apt-get install libjpeg-dev libtiff5-dev libpng-dev
1029 sudo apt-get install ffmpeg libavcodec-dev libavformat-dev libswscale-dev
libxvidcore-dev libx264-dev libxine2-dev
1030 sudo apt-get install libv4l-dev v4l-utils
1031 sudo apt-get install libgstreamer1.0-dev libgstreamer-plugins-base1.0-dev
1032 sudo apt-get install libgtk-3-dev
1033 sudo apt-get install libatlas-base-dev gfortran libeigen3-dev
1034 sudo apt-get install python3-dev python3-numpy
1035 sudo apt install unzip
```

```
1038 mkdir opencv-342
1039 cd opencv-342
1040 ls
1041 wget -O opencv.zip https://github.com/opencv/opencv/archive/3.4.2.zip
1042 unzip opencv.zip
1043 wget -O opencv_contrib.zip https://github.com/opencv/opencv_contrib/archive/3.4.2.zip
1044 unzip opencv_contrib.zip
```

cmake 명령어 실패

```
--
--   OpenCL:                      YES (no extra features)
--   Include path:                /home/user/opencv-342/opencv-3.4.2/3rdparty/include/opencvcl/1.2
--   Link libraries:              Dynamic load
--
--   Python 3:
--   Interpreter:                 /usr/bin/python3 (ver 3.10.6)
--   Libraries:                   /usr/lib/x86_64-linux-gnu/libpython3.10m.so
--   numpy:                       /usr/lib/python3/dist-packages/numpy/core/include (ver 1.21.5)
--   packages path:               /usr/lib/python3/dist-packages
--
--   Python (for build):          /usr/bin/python3
--
--   Java:
--   ant:                         NO
--   JNI:                         NO
--   Java wrappers:               NO
--   Java tests:                  NO
--
--   Matlab:                      NO
--
--   Install to:                  /usr/local
-- -----
--
-- Configuring done
-- Generating done
-- Build files have been written to: /home/user/opencv-342/opencv-3.4.2/build
```

```
1106  cmake -D CMAKE_BUILD_TYPE=RELEASE \\\n-D CMAKE_INSTALL_PREFIX=/usr/local \\\n-D WITH_TBB=OFF \\\n-D WITH_IPP=OFF \\\n-D WITH_1
394=OFF \\\n-D BUILD_WITH_DEBUG_INFO=OFF \\\n-D BUILD_DOCS=OFF \\\n-D INSTALL_C_EXAMPLES=ON \\\n-D INSTALL_PYTHON_EXAMPLES=ON \\\n-D B
UILD_EXAMPLES=OFF \\\n-D BUILD_TESTS=OFF \\\n-D BUILD_PERF_TESTS=OFF \\\n-D WITH_QT=OFF \\\n-D WITH_GTK=ON \\\n-D WITH_OPENGL=ON \\\n-D
OPENCV_EXTRA_MODULES_PATH=../../opencv_contrib-3.4.2/modules \\\n-D WITH_V4L=ON \\\n-D WITH_FFMPEG=ON \\\n-D WITH_XINE=ON \\\n-D BU
ILD_NEW_PYTHON_SUPPORT=ON \\\n-D OPENCV_GENERATE_PKGCONFIG=ON \\\n-D PYTHON2_INCLUDE_DIR=/usr/include/python2.7 \\\n-D PYTHON2_NUMPY
_INCLUDE_DIRS=/usr/lib/python2.7/dist-packages/numpy/core/include/ \\\n-D PYTHON2_PACKAGES_PATH=/usr/lib/python2.7/dist-packages \
\\n-D PYTHON2_LIBRARY=/usr/lib/x86_64-linux-gnu/libpython2.7.so \\\n-D PYTHON3_INCLUDE_DIR=/usr/include/python3.10m \\\n-D PYTHON3_N
UMPY_INCLUDE_DIRS=/usr/lib/python3/dist-packages/numpy/core/include/ \\\n-D PYTHON3_PACKAGES_PATH=/usr/lib/python3/dist-packages
\\n-D PYTHON3_LIBRARY=/usr/lib/x86_64-linux-gnu/libpython3.10m.so \\\n../
```

cmake 명령어 - option too much & 일일이 추가해야함