

YOLO V3

저번주 진행상황

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/opencl/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 \
  -D CMAKE_INSTALL_PREFIX=/usr/local \
  -D WITH_TBB=OFF \
  -D WITH_IPP=OFF \
  -D WITH_1394=OFF \
  -D BUILD_WITH_DEBUG_INFO=OFF \
  -D BUILD_DOCS=OFF \
  -D INSTALL_C_EXAMPLES=ON \
  -D INSTALL_PYTHON_EXAMPLES=ON \
  -D BUILD_EXAMPLES=OFF \
  -D BUILD_TESTS=OFF \
  -D WITH_QT=OFF \
  -D WITH_OPENGL=ON \
  -D WITH_OPENEXR=ON \
  -D WITH_XINE=ON \
  -D BUILD_NEW_PYTHON_SUPPORT=ON \
  -D OPENCV_GENERATE_PKGCONFIG=ON \
  -D PYTHON2_INCLUDE_DIR=/usr/include/python2.7 \
  -D PYTHON2_NUMPY_INCLUDE_DIRS=/usr/lib/python2.7/dist-packages/numpy/core/include \
  -D PYTHON2_PACKAGES_PATH=/usr/lib/python2.7/dist-packages \
  -D PYTHON2_LIBRARY=/usr/lib/x86_64-linux-gnu/libpython2.7.so \
  -D PYTHON3_INCLUDE_DIR=/usr/include/python3.10m \
  -D PYTHON3_NUMPY_INCLUDE_DIRS=/usr/lib/python3/dist-packages/numpy/core/include \
  -D PYTHON3_PACKAGES_PATH=/usr/lib/python3/dist-packages \
  -D PYTHON3_LIBRARY=/usr/lib/x86_64-linux-gnu/libpython3.10m.so
```

cmake 명령어 - option too much & 일일이 추가해야함

OpenCV

설치 시, 확인해야할 것

- 설치 방법: 2개
 1. Install OpenCV from Ubuntu Repository.

```
pip OR sudo apt-get install (python-opencv)
```
 - ✓ 2. Build OpenCV from source with CUDA support to enable OpenCV-DNN-CUDA module

```
wget opencv/opencv_contrib > 일일이 build 필요
```
- python, gcc 컴파일 둘 다 잘 실행되는지 확인
 - python은 따로 파일 생성해서 확인, gcc는 yolov3를 make 실행 시 확인
 - (이번주에 다룬 이슈) python에서는 잘 실행되지만, gcc 버전 호환 때문에 컴파일이 안 된 경우도 있음
- 오픈소스 Yolo v3와 버전 호환성 확인
 - make 실행 시 확인 가능
 - (이번주에 다룬 이슈) yolo v3는 opencv 4부터는 update 안 됨

Opencv 설치

4.5.5 ver

1. cuda & cudnn 설치

- cuda - 11.7 설치 (already installed)
- cudnn (8.6 deb (ubuntu_2204) ver) 설치함

```
user ~/cudnn/samples/v8/mnistCUDNN> ./mnistCUDNN
Executing: mnistCUDNN
cudnnGetVersion() : 8600 , CUDNN_VERSION from cudnn.h : 8600 (8.6.0)
Host compiler version : GCC 11.3.0

There are 1 CUDA capable devices on your machine :
device 0 : sms 82 Capabilities 8.6, SmClock 1695.0 Mhz, MemSize (Mb) 24267, MemClock 9751.0 Mhz, Ecc=0, boardGroupID=0
Using device 0

Testing single precision
Loading binary file data/conv1.bin
Loading binary file data/conv1.bias.bin
Loading binary file data/conv2.bin
Loading binary file data/conv2.bias.bin
Loading binary file data/ip1.bin
Loading binary file data/ip1.bias.bin
Loading binary file data/ip2.bin
Loading binary file data/ip2.bias.bin
Loading image data/one_28x28.pgm
Performing forward propagation ...
Testing cudnnGetConvolutionForwardAlgorithm_v7 ...
^^^^ CUDNN_STATUS_SUCCESS for Algo 1: -1.000000 time requiring 0 memory
```

```
...
^^^^ CUDNN_STATUS_NOT_SUPPORTED for Algo 3: -1.000000 time requiring 0 memory
Resulting weights from Softmax:
0.0000000 0.0000000 0.0000000 0.0000000 0.00000714 0.0000000 0.0000000 0.0000000 0.0000000
Loading image data/five_28x28.pgm
Performing forward propagation ...
Resulting weights from Softmax:
0.0000000 0.0000008 0.0000000 0.0000002 0.0000000 1.0000000 0.0000154 0.0000000 0.0000012 0.0000006

Result of classification: 1 3 5

Test passed!
```

(Test passed!) - 잘 설치됨

2. Install the necessary Ubuntu packages if not already installed.

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install build-essential cmake unzip pkg-config
sudo apt-get install libjpeg-dev libpng-dev libtiff-dev
sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev
sudo apt-get install libv4l-dev libxvidcore-dev libx264-dev
sudo apt-get install libgtk-3-dev
sudo apt-get install libblas-dev liblapack-dev gfortran
sudo apt-get install python3-dev
```

3. Download and unzip opencv & opencv_contrib

- wget & unzip

4. virtualenv & virtualenvwrapper 설치

- 가상환경 생성: opencv_dnn_cuda

- workon opencv_dnn_cuda (가상환경 활성화)

- ~/zshrc에서 virtualenv 환경변수 export

- 가상환경 설치: python3, numpy

Opencv 설치

4.5.5 ver

5. build 파일 생성 후, cmake & make 실행

(저번주 실패한 단계)

```
cmake -D CMAKE_BUILD_TYPE=RELEASE \
-D CMAKE_INSTALL_PREFIX=/usr/local \
-D INSTALL_PYTHON_EXAMPLES=ON \
-D INSTALL_C_EXAMPLES=OFF \
-D OPENCV_ENABLE_NONFREE=ON \
-D WITH_CUDA=ON \
-D WITH_CUDNN=ON \
-D OPENCV_DNN_CUDA=ON \
-D ENABLE_FAST_MATH=1 \
-D CUDA_FAST_MATH=1 \
-D CUDA_ARCH_BIN=6.1 \
-D WITH_CUBLAS=1 \
-D OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib/modules \
-D HAVE_opencv_python3=ON \
-D PYTHON_EXECUTABLE=~/virtualenvs/opencv_dnn_cuda/bin/python \
-D BUILD_EXAMPLES=..
```

(-D CUDA_ARCH_BIN=6.1 > 8.6으로 변경)

• **make -j16**

• 가상환경 site-packages에 cv2.so를 sym-link

6. opencv 설치 확인

- **python 실행 OK**

```
user ~ ➤ workon opencv_dnn_cuda
(opencv_dnn_cuda) user ~ ➤ python
Python 3.10.6 (main, Nov 14 2022, 16:10:14) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import cv2
>>> cv2.__version__
'4.5.5'
>>>
```



```
(opencv_dnn_cuda) user ~/Desktop ➤ python opencv_test.py
CUDA using GPU --- 1.026811122894287 seconds ---
CPU --- 1.552598237991333 seconds ---
```

- **gcc 컴파일 - yolov3에서 make할 때, gcc 컴파일 함**
 - **실패**
 - **원인파악: gcc 11ver. 과 opencv 4.5.5 호환성**
 - **원인파악2: yolov3는 opencv 4를 지원 x**

Opencv 설치

4.5.5 ver

5. build 파일 생성 후, cmake & make 실행

(저번주 실패한 단계)

```
cmake -D CMAKE_BUILD_TYPE=RELEASE \
-D CMAKE_INSTALL_PREFIX=/usr/local \
-D INSTALL_PYTHON_EXAMPLES=ON \
-D INSTALL_C_EXAMPLES=OFF \
-D OPENCV_ENABLE_NONFREE=ON \
-D WITH_CUDA=ON \
-D WITH_CUDNN=ON \
-D OPENCV_DNN_CUDA=ON \
-D ENABLE_FAST_MATH=1 \
-D CUDA_FAST_MATH=1 \
-D CUDA_ARCH_BIN=6.1 \
-D WITH_CUBLAS=1 \
-D OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib/modules \
-D HAVE_opencv_python3=ON \
-D
PYTHON_EXECUTABLE=~/virtualenvs/opencv_dnn_cuda/bin/python \
-D BUILD_EXAMPLES=ON ..
```

• (-D CUDA_ARCH_BIN=6.1 > 8.6으로 변경)

• `make -j16`

• 가상환경 site-packages에 cv2.so를 sym-link

6. opencv 설치 확인

- python 실행 OK

```
user ➤ workon opencv_dnn_cuda
(opencv_dnn_cuda) user ➤ python
Python 3.10.6 (main, Nov 14 2022, 16:10:14) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import cv2
>>> cv2.__version__
'4.5.5'
>>>
(opencv_dnn_cuda) user ➤ ~/Desktop ➤ python opencv_test.py
CUDA using GPU --- 1.026811122894287 seconds ---
CPU --- 1.552598237991333 seconds ---
```

- gcc 컴파일 - yolov3에서 make할 때, gcc 컴파일 함

• 실패 -> 오류 해결 (자세한 내용은 다음 페이지부터)

- 문제1: gcc 11ver. 과 opencv 4.5.5 호환성
- 문제2: yolov3는 opencv 4를 지원 x
- 문제3: pkg_config_path 오류
- 문제4: Makefile 내 opencv link 안 맞음
- 문제5: cudnn 버전 호환성

make 시 오류 발생 !!

(~/Desktop/darknet) make

- **makefile 설정 (GPU=1, CUDNN=1, OPENCV=1) 후 make 명령어 실행함 -> 각종 오류 발생**

```
@@ -1,14 +1,15 @@
1 - GPU=0
2 - CUDNN=0
3 - OPENCV=0
4 OPENMP=0
5 DEBUG=0
6
7 - ARCH= -gencode arch=compute_30,code=sm_30 \
8 -     -gencode arch=compute_35,code=sm_35 \
9     -gencode arch=compute_50,code=[sm_50,compute_50] \
10    -gencode arch=compute_52,code=[sm_52,compute_52]
11   # -gencode arch=compute_20,code=[sm_20,sm_21] \ This one is deprecated?
12
13  # This is what I use, uncomment if you know your arch and want to specify
14  # ARCH= -gencode arch=compute_52,code=compute_52
15
+
+ GPU=1
+ CUDNN=1
+ OPENCV=1
4 OPENMP=0
5 DEBUG=0
6
7 + ARCH= -gencode arch=compute_35,code=sm_35 \
8     -gencode arch=compute_50,code=[sm_50,compute_50] \
9     + -gencode arch=compute_52,code=[sm_52,compute_52] \
10    + -gencode arch=compute_70,code=[sm_70,compute_70] \
11    + -gencode arch=compute_75,code=[sm_75,compute_75]\
12    + -gencode arch=compute_86,code=[sm_86,compute_86]
13
14  # This is what I use, uncomment if you know your arch and want to specify
15  # ARCH= -gencode arch=compute_52,code=compute_52
```

- **for 오류 해결, GPU=1, CUDNN=1, OPENCV=0 등 다양한 조합으로 바꾸며 make 실행**
- > 다음장부터 오류 해결 과정을 담음

make 시 오류 발생 1

(~/Desktop/darknet) make

(오류 캡처 x)

- 문제3: **pkg_config_path** 오류

opencv.pc 없음 > 새로 생성

- ls /usr/local/lib/pkgconfig**
- vim opencv.pc**

The screenshot shows a terminal window with the title "vim /usr/local/lib/pkgconfig/opencv.pc". The file content is as follows:

```
1 prefix=/usr/local
2 exec_prefix=${prefix}
3 libdir=${exec_prefix}/lib
4 includedir_old=${prefix}/include/opencv4/opencv2
5 includedir_new=${prefix}/include/opencv4
6
7 Name: OpenCV
8 Description: Open Source Computer Vision Library
9 Version: 4.5.5
10 Libs: -L${exec_prefix}/lib -lopencv_gapi -lopencv_stitching -lopencv_aruco -lopencv_bgsegm -lopencv_bioinspir
     ed -lopencv_ccalib -lopencv_cvv -lopencv_dnn_objdetect -lopencv_dpm -lopencv_face -lopencv_freetype -lopencv_
     fuzzy -lopencv_hdf -lopencv_hfs -lopencv_img_hash -lopencv_line_descriptor -lopencv_quality -lopencv_reg -lop
     encv_rgbd -lopencv_saliency -lopencv_sfm -lopencv_stereo -lopencv_structured_light -lopencv_phase_unwrapping
     -lopencv_superres -lopencv_optflow -lopencv_surface_matching -lopencv_tracking -lopencv_datasets -lopencv_tex
     t -lopencv_dnn -lopencv_plot -lopencv_videostab -lopencv_video -lopencv_xfeatures2d -lopencv_shape -lopencv_m
     l -lopencv_ximgproc -lopencv_xobjdetect -lopencv_objdetect -lopencv_calib3d -lopencv_features2d -lopencv_high
     gui -lopencv_videoio -lopencv_imgcodecs -lopencv_flann -lopencv_xphoto -lopencv_photo -lopencv_imgproc -lop
     encv_core
11 Libs.private: -ldl -lm -lpthread -lrt -L/usr/lib/x86_64-linux-gnu -lGL -lGLU
12 Cflags: -I${includedir_old} -I${includedir_new}
```

The status bar at the bottom of the terminal window shows "/usr/local/lib/pkgconfig/opencv.pc [R0]" and "1,1 All".

make 시 오류 발생 2

(~/Desktop/darknet) make

```
(opencv_dnn_cuda) user ~/Desktop/darknet master ± make
g++ -Iinclude/ -Isrc/ -DOPENCV `pkg-config --cflags opencv` -DGPU -I/usr/local/cuda/include/ -DCUDNN -Wall -Wno-unused-result -Wno-unknown-pragmas -Wfatal-errors -fPIC -Ofast -DOPENCV -DGPU -DCUDNN -c ./src/image_opencv.cpp -o obj/image_opencv.o
./src/image_opencv.cpp:12:1: error: 'IplImage' does not name a type
  12 | IplImage *image_to_ipl(image im)
     | ^~~~~~
compilation terminated due to -Wfatal-errors.
make: *** [Makefile:90: obj/image_opencv.o] Error 1
```

- 문제1: gcc 11ver. 과 opencv 4.5.5 호환성
- 문제2: yolov3는 opencv 4를 지원 x
- 해결방법 2가지
 - 해결방법1: opencv 버전 다운그레이드 (3.4.0 ver)
-> cmake 단계에서 에러 발생
 - ✓ 해결방법2: 관련 코드 수정
(github issue - 사례 있음)

해결방법2

Fix compilation with latest OpenCV #1348

The screenshot shows a GitHub pull request titled "Fix compilation with latest OpenCV #1348". The pull request is from user "tiagoshibata" into branch "pjreddie:master". It contains 3 commits and 2 files changed. The changes are shown in a diff view across two files: Makefile and src/image_opencv.cpp.

Makefile Changes:

```
@@ -20,7 +20,7 @@ EXEC=darknet
20 OBJDIR=./obj/
21
22 CC=gcc
23 - CPP=g++
24 NVCC=nvcc
25 AR=ar
26 ARFLAGS=rcs
@@ -42,8 +42,8 @@ CFLAGS+=${OPTS}
42 ifeq ($(OPENCV), 1)
43 COMMON+= -DOPENCV
44 CFLAGS+= -DOPENCV
45 - LDFLAGS+= `pkg-config --libs opencv` -lstdc++
46 - COMMON+= `pkg-config --cflags opencv`
```

src/image_opencv.cpp Changes:

```
@@ -9,30 +9,34 @@ using namespace cv;
```

Makefile과 cpp 파일 수정

make 시 오류 발생 3

(~/Desktop/darknet) make

```
./src/gemm.c: In function 'time_gpu':  
./src/gemm.c:232: warning: 'cudaThreadSynchronize' is deprecated [-Wdeprecated-declarations]  
  232 |         cudaThreadSynchronize();  
      |  
In file included from /usr/local/cuda/include/cuda_runtime.h:96,  
                  from include/darknet.h:11,  
                  from ./src/utils.h:5,  
                  from ./src/gemm.c:2:  
/usr/local/cuda/include/cuda_runtime_api.h:1057:57: note: declared here  
  1057 | extern __CUDA_DEPRECATED __host__ cudaError_t CUDARTAPI cudaThreadSynchronize(void);  
      |  
gcc: warning: \<= linker input file unused because linking not done  
gcc: error: \<= linker input file not found: No such file or directory  
gcc: warning: You: linker input file unused because linking not done  
gcc: error: You: linker input file not found: No such file or directory  
gcc: warning: have: linker input file unused because linking not done  
gcc: error: have: linker input file not found: No such file or directory  
gcc: warning: to: linker input file unused because linking not done  
gcc: error: to: linker input file not found: No such file or directory  
gcc: warning: modify: linker input file unused because linking not done  
gcc: error: modify: linker input file not found: No such file or directory  
gcc: warning: this: linker input file unused because linking not done  
gcc: error: this: linker input file not found: No such file or directory  
gcc: warning: line: linker input file unused because linking not done  
gcc: error: line: linker input file not found: No such file or directory  
make: *** [Makefile:83: obj/gemm.o] Error 1
```

- 에러1:

cudaThreadSynchronize();

⇒ **cudaDeviceSynchronize();**

- 에러2:

gcc link 에러: opencv link가 안 맞음

pkg-config --cflags opencv 4

⇒ **pkg-config --cflags opencv**

- 문제4: Makefile 내 opencv link 안 맞음

make 시 오류 발생 4

(~/Desktop/darknet) make

```
./src/convolutional_layer.c:153:13: error: 'CUDNN_CONVOLUTION_FWD_SPECIFY_WORKSPACE_LIMIT' undeclared (first use in  
this function);
```

- 문제5: cudnn 버전 호환성
- 해결방법: ./src/convolutional_layer.c 파일 수정

```
diff --git a/src/convolutional_layer.c b/src/convolutional_layer.c  
@@ -8,6 +8,9 @@  
 8 #include <stdio.h>  
 9 #include <time.h>  
10  
11 #ifdef AI2  
12 #include "xnor_layer.h"  
13 #endif  
@@ -144,6 +147,76 @@ void cudnn_convolutional_setup(layer *l)  
144     error("CUDNN < 7 doesn't support groups, please upgrade!");  
145 }  
146 #endif  
147     error("CUDNN < 7 doesn't support groups, please upgrade!");  
148 }  
149 #endif  
150 + #if CUDNN_MAJOR >= 8  
151 + int returnedAlgoCount;  
152 + cudnnConvolutionFwdAlgoPerf_t fw_results[2 *  
CUDNN_CONVOLUTION_FWD_ALGO_COUNT];  
153 + cudnnConvolutionBwdDataAlgoPerf_t bd_results[2 *  
CUDNN_CONVOLUTION_BWD_DATA_ALGO_COUNT];  
154 + cudnnConvolutionBwdFilterAlgoPerf_t bf_results[2 *  
CUDNN_CONVOLUTION_BWD_FILTER_ALGO_COUNT];  
155 +  
156 + cudnnFindConvolutionForwardAlgorithm(cudnn_handle(),  
157 + l->srcTensorDesc,  
158 + l->weightDesc,  
159 + l->convDesc,  
160 + l->dstTensorDesc,  
161 + CUDNN_CONVOLUTION_FWD_ALGO_COUNT,  
162 + &returnedAlgoCount,  
163 + fw_results);  
164 + for(int algoIndex = 0; algoIndex < returnedAlgoCount; ++algoIndex){  
165 + #if PRINT_CUDNN_ALGO > 0  
166 + printf("^^^^ %s for Algo %d: %f time requiring %llu memory\n",  
cudnnGetErrorString(fw_results[algoIndex].status)
```

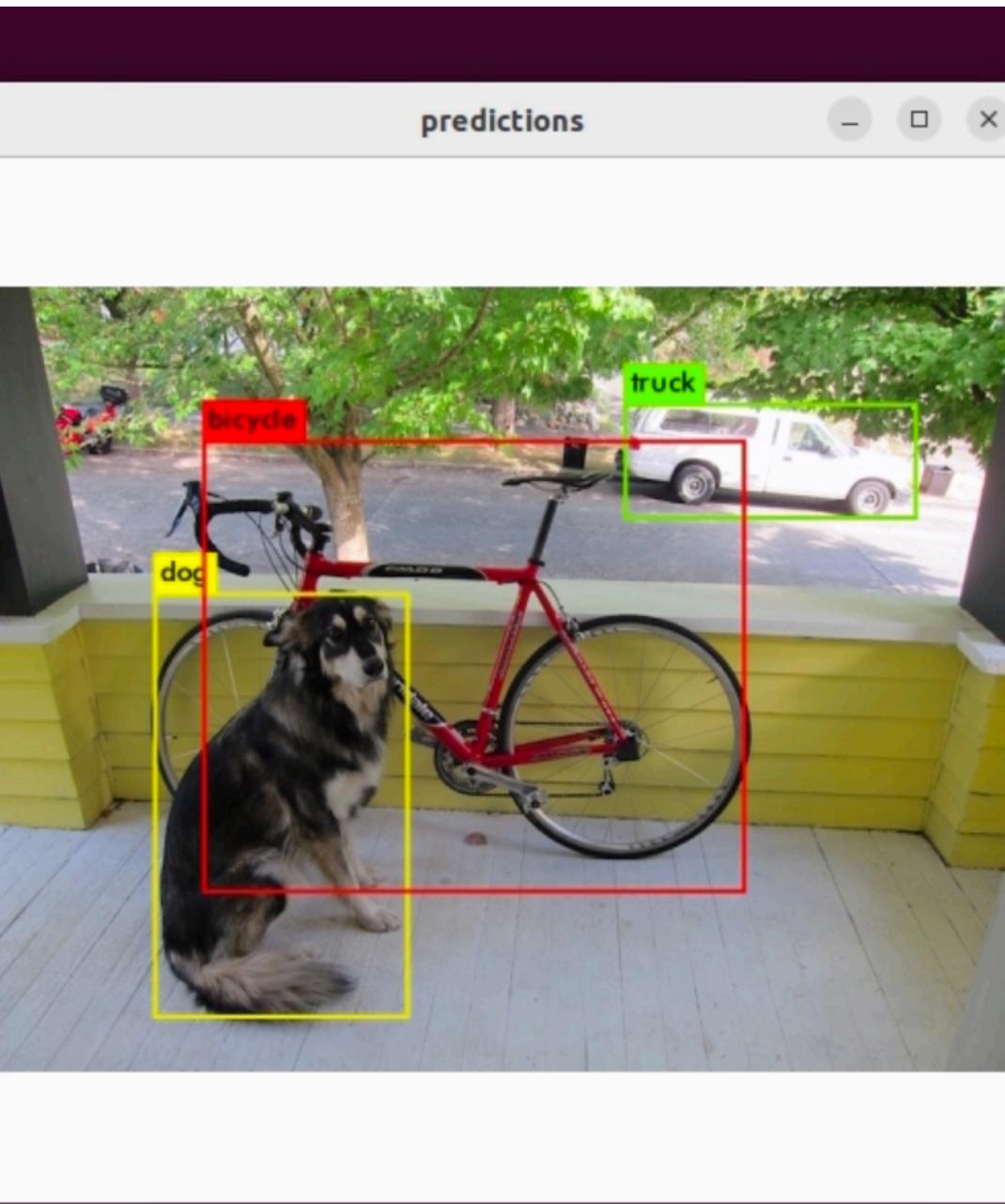
make 성공 !!

```
(~/Desktop/darknet) ./darknet detect cfg...
```

- ```
user ➤ ~ ➤ workon opencv_dnn_cuda
(opencv_dnn_cuda) user ➤ ~ ➤ cd Desktop/darknet
(opencv_dnn_cuda) user ➤ ~/Desktop/darknet ➤ master ➤ ./darknet detect cfg/yolov3.cfg yolov3.weights data/dog.jpg
```

- **darknet detect 실행 결과**

```
76 conv 1024 3 x 3 / 1 13 x 13 x 512 -> 13 x 13 x 1024 1.595 BFLOPs
77 conv 512 1 x 1 / 1 13 x 13 x 1024 -> 13 x 13 x 512 0.177 BFLOPs
78 conv 1024 3 x 3 / 1 13 x 13 x 512 -> 13 x 13 x 1024 1.595 BFLOPs
79 conv 512 1 x 1 / 1 13 x 13 x 1024 -> 13 x 13 x 512 0.177 BFLOPs
80 conv 1024 3 x 3 / 1 13 x 13 x 512 -> 13 x 13 x 1024 1.595 BFLOPs
81 conv 255 1 x 1 / 1 13 x 13 x 1024 -> 13 x 13 x 255 0.088 BFLOPs
82 yolo
83 route 79
84 conv 256 1 x 1 / 1 13 x 13 x 512 -> 13 x 13 x 256 0.044 BFLOPs
85 upsample 2x 13 x 13 x 256 -> 26 x 26 x 256
86 route 85 61
87 conv 256 1 x 1 / 1 26 x 26 x 768 -> 26 x 26 x 256 0.266 BFLOPs
88 conv 512 3 x 3 / 1 26 x 26 x 256 -> 26 x 26 x 512 1.595 BFLOPs
89 conv 256 1 x 1 / 1 26 x 26 x 512 -> 26 x 26 x 256 0.177 BFLOPs
90 conv 512 3 x 3 / 1 26 x 26 x 256 -> 26 x 26 x 512 1.595 BFLOPs
91 conv 256 1 x 1 / 1 26 x 26 x 512 -> 26 x 26 x 256 0.177 BFLOPs
92 conv 512 3 x 3 / 1 26 x 26 x 256 -> 26 x 26 x 512 1.595 BFLOPs
93 conv 255 1 x 1 / 1 26 x 26 x 512 -> 26 x 26 x 255 0.177 BFLOPs
94 yolo
95 route 91
96 conv 128 1 x 1 / 1 26 x 26 x 256 -> 26 x 26 x 128 0.044 BFLOPs
97 upsample 2x 26 x 26 x 128 -> 52 x 52 x 128
98 route 97 36
99 conv 128 1 x 1 / 1 52 x 52 x 384 -> 52 x 52 x 128 0.266 BFLOPs
100 conv 256 3 x 3 / 1 52 x 52 x 128 -> 52 x 52 x 256 1.595 BFLOPs
101 conv 128 1 x 1 / 1 52 x 52 x 256 -> 52 x 52 x 128 0.177 BFLOPs
102 conv 256 3 x 3 / 1 52 x 52 x 128 -> 52 x 52 x 256 1.595 BFLOPs
103 conv 128 1 x 1 / 1 52 x 52 x 256 -> 52 x 52 x 128 0.177 BFLOPs
104 conv 256 3 x 3 / 1 52 x 52 x 128 -> 52 x 52 x 256 1.595 BFLOPs
105 conv 255 1 x 1 / 1 52 x 52 x 256 -> 52 x 52 x 255 0.353 BFLOPs
106 yolo
Loading weights from yolov3.weights...Done!
data/dog.jpg: Predicted in 1.106807 seconds.
dog: 99%
truck: 93%
bicycle: 99%
```



# GPU, CUDNN, OPENCV (makefile)

```
./darknet detect cfg/yolov3.cfg yolov3.weights data/dog.jpg
```

- **gpu=0, cudnn=0, opencv=0**

```
Loading weights from yolov3.weights...Done!
data/dog.jpg: Predicted in 10.088496 seconds.
dog: 99%
truck: 92%
bicycle: 99%
```

- **gpu=1, cudnn=0, opencv=1**

```
Loading weights from yolov3.weights...Done!
data/dog.jpg: Predicted in 0.982864 seconds.
dog: 99%
truck: 92%
bicycle: 99%
```

- **gpu=1, cudnn=1, opencv=1**

```
Loading weights from yolov3.weights...Done!
data/dog.jpg: Predicted in 0.013085 seconds.
dog: 99%
truck: 93%
bicycle: 99%
```

- 성공한줄 알았으나, cudnn=1의 경우, detection (classification)이 안 되는 경우를 다수 발견

```
Loading weights from yolov3.weights...Done!
data/dog.jpg: Predicted in 0.012031 seconds. Loading weights from yolov3.weights...Done!
Loading weights from yolov3.weights...Done! truck: 79% data/dog.jpg: Predicted in 0.012506 seconds.
data/dog.jpg: Predicted in 0.012705 seconds. bicycle: 93% bicycle: 96%
```