1. **Configuration파일과 model파일 다운로드 및 수정하기**

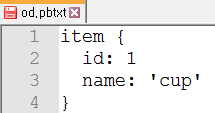
* Tesnorflow object detection을 트레이닝하기 위해서 configuration파일 (.config) 과 model 파일(.tar.gz)이 필요합니다.
* configuration파일은 아래 링크에서 다운 할 수 있습니다. Configuration과 model파일은 같은 이름을 다운 해야 합니다. 예를 들면 mobilenet은 속도가 빠르지만 정확도는 조금 떨어집니다. 다운 후 training폴더에 복사합니다.

<https://github.com/tensorflow/models/tree/master/research/object_detection/samples/configs>

* model파일은 아래 링크에서 다운 할 수 있습니다. 다운 후 od폴더에서 압축을 풉니다.

<https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md>

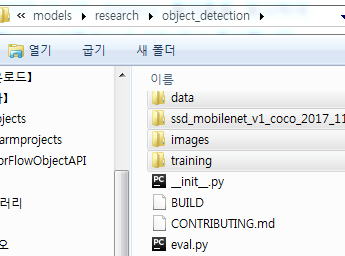
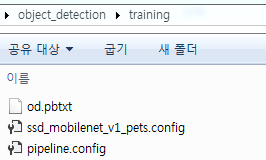
* training폴더에 label map와 비슷한 od.pbtxt를 만듭니다.



* configuration파일에서 우리가 원하는 트레이닝으로 class, path등 변수들을 고쳐야 합니다.
  + Line 9: num\_classes: 1
  + Line 143: 메모리에러가 뜨면 batch\_size를 24에서 줄일수 있습니다
  + Line 158: fine\_tune\_checkpoint: "ssd\_mobilenet\_v1\_coco\_2017\_11\_17/model.ckpt"
  + Line 177: input\_path: "data/train.record"
  + Line 179: label\_map\_path: " training/od.pbtxt"
  + Line 191: input\_path: "data/test.record"
  + Line 193: label\_map\_path: " training/od.pbtxt"
  + Line 164: num\_steps: 10000 트레이닝 스텝을 변경할 수 있습니다

1. **필요한 파일을 복사하기**

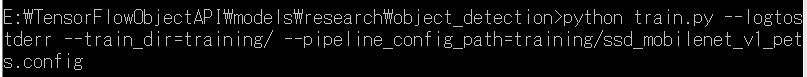
* ob폴더 속에는 트레이닝을 위해 준비한 파일들이 있습니다. 여기서 data, images, sssd\_mobilenet\_v1\_coco\_2017\_11\_17, training 폴더들을 복사하여 E:\TensorFlowObjectAPI\models\research\object\_detection복사합니다.

1. **트레이닝 하기**

* E:\TensorFlowObjectAPI\models\research\object\_detection에서 다음 명령1을 실행합니다.

python train.py --logtostderr --train\_dir=training/ --pipeline\_config\_path=training/ssd\_mobilenet\_v1\_pets.config



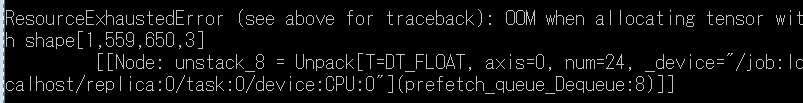
* 아래와 같이 ImportError가 뜹니다. 이때 windows에서는 E:\TensorFlowObjectAPI\models\research\slim폴더 안의 deployment와 nets 폴더를 Python35/Lib/site-packages폴더에 복사합니다. Linux에서는 tensorflow/models/research에서 아래 명령어를 실행합니다.

# From tensorflow/models/research/

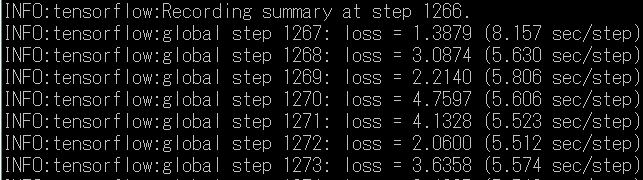
export PYTHONPATH=$PYTHONPATH:`pwd`:`pwd`/slim

C:\Users\Jane\AppData\Local\Temp\WeChat Files\ba72a38b4e1d30056cdecdffc5fcfb9.png

* 다시 명령1을 실행하면 ResourceExhaustedError가 뜹니다. 이때 configuration 파일 중 Line 143: batch\_size를 (12로) 줄이면 됩니다



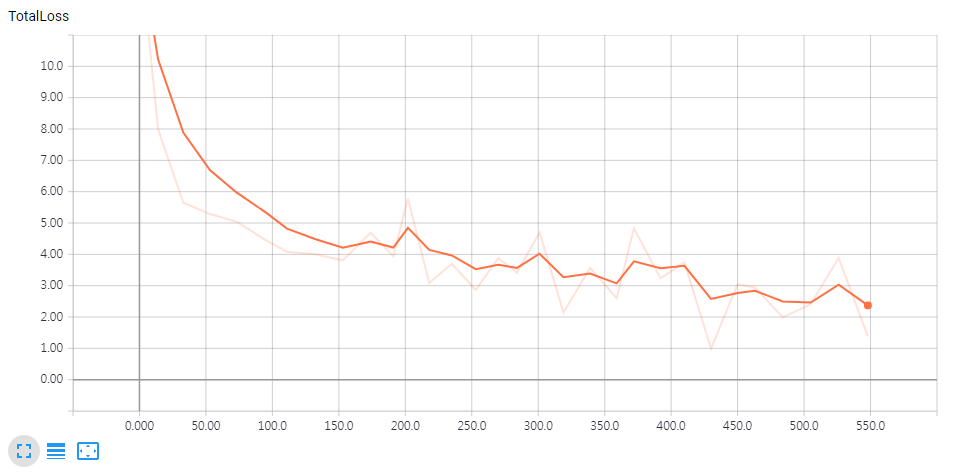
* 다시 명령1을 실행하면 정상적으로 트레이닝 됩니다.



* 트레이닝 과정을 tensorboard에서 확인할 수 있습니다. cmd에서 다음 명령을 실행합니다. 그리고 <http://Jane-PC:6006>에서 확인할 수 있습니다. TotalLoss가 거의 변하지 않으면 트레이닝 끝난 상태여서 트레이닝을 중단해도 됩니다.

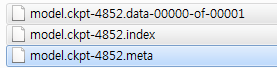
tensorboard --logdir=E:\TensorFlowObjectAPI\models\research\object\_detection\training



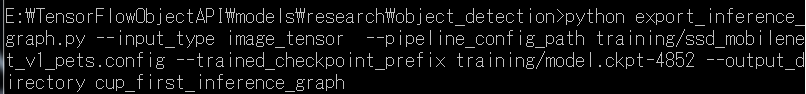


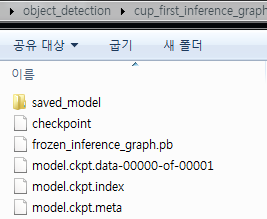
1. **Export & Frozen inference graph**

* 트레이닝을 하면 traning 폴더에 checkpoint들이 저장됩니다. Checkpoint 중에서 3개파일 다 갖고 있는 것을 선택하면 됩니다. 그리고 Frozen할 모듈은 cup\_first\_inference에 저장됩니다. object\_detection아래에서 cmd를 실행하고 다음 명령어를 실행하면 트레이닝한 모듈을 frozen할 수 있습니다.



python export\_inference\_graph.py --input\_type image\_tensor --pipeline\_config\_path **training/ssd\_mobilenet\_v1\_pets.config** --trained\_checkpoint\_prefix **training/model.ckpt-4852** --output\_directory **cup\_first\_inference\_graph**





1. **트레이닝 모듈로 테스트하기**

* object\_detection아래에서 cmd를 열고 jupyter notebook를 열고 다음 몇 개 부분을 수정합니다.

C:\Users\Jane\AppData\Local\Temp\1515475604.png

* + Variables 부분은 다음과 같이 수정:

MODEL\_NAME = 'cup\_first\_inference\_graph'

PATH\_TO\_CKPT = MODEL\_NAME + '/frozen\_inference\_graph.pb'

PATH\_TO\_LABELS = os.path.join('training', 'ob.pbtxt')

NUM\_CLASSES = 1

* + Download Model 부분 모두 삭제
  + Detection 에서 테스트 이미지 파일 위치, 개수 등을 변경
* 결과:

