**YOLO\_Model Final Training & Prediction Document**

**Requirements and setup:**

First of all we need to install a couple of packages and download some more from GitHub

1) TensorFlow ( Install using pip )

2) DarkFlow – TensorFlow adaptation of Darknet network runner

3) Some configuration and weight files

1. Installing TensorFlow Install TensorFlow using PIP :

**$ sudo pip install tensorflow**

2. Installing DarkFlow

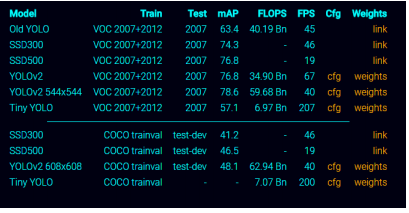
DarkFlow is a network builder adapted from Darknet, it allows building TensorFlow networks from cfg files and loading pre trained weights. We will use it to run YOLO.

a. Clone DarkFlow from : <https://github.com/thtrieu/darkflow>

b. Weights and cfg files can be found on : https://pjreddie.com/darknet/yolo/

c. We will use Yolo-v2: (Last One)

d. Download cfg and weights file and copy them to the DarkFlow folder



Setup the configuration and weights

1. Download the weights file

2. Download the configuration file

**Getting started**

You can choose one of the following three ways to get started with darkflow.

1. Just build the Cython extensions in place.

**python3 setup.py build\_ext --inplace**

1. Let pip install darkflow globally in dev mode (still globally accessible, but changes to the code immediately take effect)

**pip install -e .**

1. Install with pip globally

**pip install .**

**Clone Or Download Repository**

<https://github.com/thtrieu/darkflow>

### Training on your own dataset

The steps below assume we want to use tiny YOLO and our dataset has 3 classes

1. Create a copy of the configuration file tiny-yolo-voc.cfg and rename it according to your preference tiny-yolo-voc-3c.cfg (It is crucial that you leave the original tiny-yolo-voc.cfg file unchanged, see below for explanation).
2. In tiny-yolo-voc-3c.cfg, change classes in the [region] layer (the last layer) to the number of classes you are going to train for. In our case, classes are set to 3.
3. ...

**[region]**

**anchors = 1.08,1.19, 3.42,4.41, 6.63,11.38, 9.42,5.11, 16.62,10.52**

**bias\_match=1**

**classes=3**

**coords=4**

**num=5**

**softmax=1**

...

1. In tiny-yolo-voc-3c.cfg, change filters in the [convolutional] layer (the second to last layer) to num \* (classes + 5). In our case, num is 5 and classes are 3 so 5 \* (3 + 5) = 40 therefore filters are set to 40.

[**convolutional]**

**size=1**

**stride=1**

**pad=1**

**filters=40**

**activation=linear**

**[region]**

**anchors = 1.08,1.19, 3.42,4.41, 6.63,11.38, 9.42,5.11, 16.62,10.52**

...

Change labels.txt to include the label(s) you want to train on (number of labels should be the same as the number of classes you set in tiny-yolo-voc-3c.cfg file). In our case, labels.txt will contain 3 labels.

**label1**

**label2**

**label3**

1. Reference the tiny-yolo-voc-3c.cfg model when you train.

**flow --model /home/ubuntu/Music/YOLO\_WAJIHA/darkflow-master-1/darkflow-master/cfg**/**tiny-yolo-voc-1c.cfg --load bin/** **yolov2-tiny-voc.weights --train --annotation /home/ubuntu/Music/YOLO\_WAJIHA/darkflow-master-1/darkflow-master/images/train --dataset /home/ubuntu/Music/YOLO\_WAJIHA/darkflow-master-1/darkflow-master/images/train**

**OUTPUT:**

**After training the model getting the loss as 222.**

