

YOLOv3 Object Detection

05/31/2019

YOLOv3 Object Detector in Keras with TensorFlow backend. This notebook uses a YOLOv3 network, pre-trained on the ImageNet and Microsoft COCO datasets. OpenCV (CV2) used for reading and saving images.

Using libraries YOLOv3 from

- <https://pjreddie.com/darknet/yolo/> (<https://pjreddie.com/darknet/yolo/>)
- <https://github.com/experiencor/keras-yolo3> (<https://github.com/experiencor/keras-yolo3>)

YOLOv3 Concepts

Image Grid

Anchor boxes

Bounding boxes

Training data

```
In [1]: # Using TensorFlow 1.13.1
# YOLOv3 Keras code for TF 1.x; not compatible with TF 2.0 alpha

import numpy as np
import os
import matplotlib.pyplot as plt
from datetime import datetime

# Install CV2 with `pip install opencv-python`, rather than Anaconda
import cv2

import tensorflow as tf
import tensorflow.keras as keras

print("TensorFlow:", tf.__version__, " Keras:", keras.__version__, " CV2:", cv2.__version__)
```

TensorFlow: 1.13.1 Keras: 2.2.4-tf CV2: 4.1.0

Define and load the Keras YOLOv3 model (106 layers)

Import Keras model experiencor/keras-yolo3

- <https://github.com/experiencor/keras-yolo3> (<https://github.com/experiencor/keras-yolo3>)

Main functions used:

```
yolov3 = make_yolov3_model()
weight_reader = WeightReader(weights_path)
weight_reader.load_weights(yolov3)
```

It takes about 1 minute to load YOLO model (Intel i7 quad-core 2.8 GHz)

```
CPU times: user 54.4 s, sys: 1.97 s, total: 56.3 s
Wall time: 55.6 s
```

```
In [2]: # Import experiencor/keras-yolo3 library (some code changes for error debugging)
from keras_yolo3 import yolo3_one_file_to_detect_them_all as xp_yolov3

# YOLOv3 main() #1
# args.weights
weights_path = '/Users/nelson/dev/cv/yolov3-keras/models/yolov3.weights'
# args.image
image_path = './data/images/african_elephant_people.jpg' # sample image

# make the yolov3 model to predict 80 classes on COCO
print("make_yolov3_model:")
yolov3 = xp_yolov3.make_yolov3_model()
print()

# define weights reader
weight_reader = xp_yolov3.WeightReader(weights_path)
```

make_yolov3_model:

WARNING:tensorflow:From /Users/nelson/dev/anaconda3/lib/python3.7/site-packages/tensorflow/python/framework/op_def_library.py:263: colocate_with (from tensorflow.python.framework.ops) is deprecated and will be removed in a future version.

Instructions for updating:

Colocations handled automatically by placer.

Using TensorFlow backend.

YOLOv3 - load weights

```
In [3]: # YOLOv3 - load the weights trained on MS-COCO into the model
print("load_weights(yolov3):")
%time weight_reader.load_weights(yolov3)
```

```
load_weights(yolov3):
loading weights of convolution #0
loading weights of convolution #1
loading weights of convolution #2
loading weights of convolution #3
no convolution #4
loading weights of convolution #5
loading weights of convolution #6
loading weights of convolution #7
no convolution #8
loading weights of convolution #9
loading weights of convolution #10
no convolution #11
loading weights of convolution #12
loading weights of convolution #13
loading weights of convolution #14
no convolution #15
loading weights of convolution #16
loading weights of convolution #17
no convolution #18
loading weights of convolution #19
loading weights of convolution #20
no convolution #21
loading weights of convolution #22
loading weights of convolution #23
no convolution #24
loading weights of convolution #25
loading weights of convolution #26
no convolution #27
loading weights of convolution #28
loading weights of convolution #29
no convolution #30
loading weights of convolution #31
loading weights of convolution #32
no convolution #33
loading weights of convolution #34
loading weights of convolution #35
no convolution #36
loading weights of convolution #37
loading weights of convolution #38
loading weights of convolution #39
no convolution #40
loading weights of convolution #41
```

```
loading weights of convolution #42
no convolution #43
loading weights of convolution #44
loading weights of convolution #45
no convolution #46
loading weights of convolution #47
loading weights of convolution #48
no convolution #49
loading weights of convolution #50
loading weights of convolution #51
no convolution #52
loading weights of convolution #53
loading weights of convolution #54
no convolution #55
loading weights of convolution #56
loading weights of convolution #57
no convolution #58
loading weights of convolution #59
loading weights of convolution #60
no convolution #61
loading weights of convolution #62
loading weights of convolution #63
loading weights of convolution #64
no convolution #65
loading weights of convolution #66
loading weights of convolution #67
no convolution #68
loading weights of convolution #69
loading weights of convolution #70
no convolution #71
loading weights of convolution #72
loading weights of convolution #73
no convolution #74
loading weights of convolution #75
loading weights of convolution #76
loading weights of convolution #77
loading weights of convolution #78
loading weights of convolution #79
loading weights of convolution #80
loading weights of convolution #81
no convolution #82
no convolution #83
loading weights of convolution #84
```

```
no convolution #85
no convolution #86
loading weights of convolution #87
loading weights of convolution #88
loading weights of convolution #89
loading weights of convolution #90
loading weights of convolution #91
loading weights of convolution #92
loading weights of convolution #93
no convolution #94
no convolution #95
loading weights of convolution #96
no convolution #97
no convolution #98
loading weights of convolution #99
loading weights of convolution #100
loading weights of convolution #101
loading weights of convolution #102
loading weights of convolution #103
loading weights of convolution #104
loading weights of convolution #105
CPU times: user 54.4 s, sys: 2.04 s, total: 56.4 s
Wall time: 55.7 s
```

```
In [4]: # <tensorflow.python.keras.engine.training.Model at 0x14caaf898>
print ("Number of yolov3.layers:", len(yolov3.layers))
```

Number of yolov3.layers: 252

```
In [5]: # YOLOv3 Keras model summary  
yolov3.summary()
```

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	(None, None, None, 3 0)		
conv_0 (Conv2D)	(None, None, None, 3 864)		input_1[0][0]
bnorm_0 (BatchNormalization)	(None, None, None, 3 128)		conv_0[0][0]
leaky_0 (LeakyReLU)	(None, None, None, 3 0)		bnorm_0[0][0]
zero_padding2d_1 (ZeroPadding2D)	(None, None, None, 3 0)		leaky_0[0][0]
conv_1 (Conv2D)	(None, None, None, 6 18432)		zero_padding2d_1[0][0]
bnorm_1 (BatchNormalization)	(None, None, None, 6 256)		conv_1[0][0]
leaky_1 (LeakyReLU)	(None, None, None, 6 0)		bnorm_1[0][0]
conv_2 (Conv2D)	(None, None, None, 3 2048)		leaky_1[0][0]
bnorm_2 (BatchNormalization)	(None, None, None, 3 128)		conv_2[0][0]
leaky_2 (LeakyReLU)	(None, None, None, 3 0)		bnorm_2[0][0]
conv_3 (Conv2D)	(None, None, None, 6 18432)		leaky_2[0][0]
bnorm_3 (BatchNormalization)	(None, None, None, 6 256)		conv_3[0][0]
leaky_3 (LeakyReLU)	(None, None, None, 6 0)		bnorm_3[0][0]
add_1 (Add)	(None, None, None, 6 0)		leaky_1[0][0] leaky_3[0][0]
zero_padding2d_2 (ZeroPadding2D)	(None, None, None, 6 0)		add_1[0][0]
conv_5 (Conv2D)	(None, None, None, 1 73728)		zero_padding2d_2[0][0]
bnorm_5 (BatchNormalization)	(None, None, None, 1 512)		conv_5[0][0]
leaky_5 (LeakyReLU)	(None, None, None, 1 0)		bnorm_5[0][0]
conv_6 (Conv2D)	(None, None, None, 6 8192)		leaky_5[0][0]

bnorm_6 (BatchNormalization)	(None, None, None, 6 256	conv_6[0][0]
leaky_6 (LeakyReLU)	(None, None, None, 6 0	bnorm_6[0][0]
conv_7 (Conv2D)	(None, None, None, 1 73728	leaky_6[0][0]
bnorm_7 (BatchNormalization)	(None, None, None, 1 512	conv_7[0][0]
leaky_7 (LeakyReLU)	(None, None, None, 1 0	bnorm_7[0][0]
add_2 (Add)	(None, None, None, 1 0	leaky_5[0][0] leaky_7[0][0]
conv_9 (Conv2D)	(None, None, None, 6 8192	add_2[0][0]
bnorm_9 (BatchNormalization)	(None, None, None, 6 256	conv_9[0][0]
leaky_9 (LeakyReLU)	(None, None, None, 6 0	bnorm_9[0][0]
conv_10 (Conv2D)	(None, None, None, 1 73728	leaky_9[0][0]
bnorm_10 (BatchNormalization)	(None, None, None, 1 512	conv_10[0][0]
leaky_10 (LeakyReLU)	(None, None, None, 1 0	bnorm_10[0][0]
add_3 (Add)	(None, None, None, 1 0	add_2[0][0] leaky_10[0][0]
zero_padding2d_3 (ZeroPadding2D	(None, None, None, 1 0	add_3[0][0]
conv_12 (Conv2D)	(None, None, None, 2 294912	zero_padding2d_3[0][0]
bnorm_12 (BatchNormalization)	(None, None, None, 2 1024	conv_12[0][0]
leaky_12 (LeakyReLU)	(None, None, None, 2 0	bnorm_12[0][0]
conv_13 (Conv2D)	(None, None, None, 1 32768	leaky_12[0][0]
bnorm_13 (BatchNormalization)	(None, None, None, 1 512	conv_13[0][0]
leaky_13 (LeakyReLU)	(None, None, None, 1 0	bnorm_13[0][0]

conv_14 (Conv2D)	(None, None, None, 2 294912	leaky_13[0][0]
bnorm_14 (BatchNormalization)	(None, None, None, 2 1024	conv_14[0][0]
leaky_14 (LeakyReLU)	(None, None, None, 2 0	bnorm_14[0][0]
add_4 (Add)	(None, None, None, 2 0	leaky_12[0][0] leaky_14[0][0]
conv_16 (Conv2D)	(None, None, None, 1 32768	add_4[0][0]
bnorm_16 (BatchNormalization)	(None, None, None, 1 512	conv_16[0][0]
leaky_16 (LeakyReLU)	(None, None, None, 1 0	bnorm_16[0][0]
conv_17 (Conv2D)	(None, None, None, 2 294912	leaky_16[0][0]
bnorm_17 (BatchNormalization)	(None, None, None, 2 1024	conv_17[0][0]
leaky_17 (LeakyReLU)	(None, None, None, 2 0	bnorm_17[0][0]
add_5 (Add)	(None, None, None, 2 0	add_4[0][0] leaky_17[0][0]
conv_19 (Conv2D)	(None, None, None, 1 32768	add_5[0][0]
bnorm_19 (BatchNormalization)	(None, None, None, 1 512	conv_19[0][0]
leaky_19 (LeakyReLU)	(None, None, None, 1 0	bnorm_19[0][0]
conv_20 (Conv2D)	(None, None, None, 2 294912	leaky_19[0][0]
bnorm_20 (BatchNormalization)	(None, None, None, 2 1024	conv_20[0][0]
leaky_20 (LeakyReLU)	(None, None, None, 2 0	bnorm_20[0][0]
add_6 (Add)	(None, None, None, 2 0	add_5[0][0] leaky_20[0][0]
conv_22 (Conv2D)	(None, None, None, 1 32768	add_6[0][0]
bnorm_22 (BatchNormalization)	(None, None, None, 1 512	conv_22[0][0]

leaky_22 (LeakyReLU)	(None, None, None, 1 0	bnorm_22[0][0]
conv_23 (Conv2D)	(None, None, None, 2 294912	leaky_22[0][0]
bnorm_23 (BatchNormalization)	(None, None, None, 2 1024	conv_23[0][0]
leaky_23 (LeakyReLU)	(None, None, None, 2 0	bnorm_23[0][0]
add_7 (Add)	(None, None, None, 2 0	add_6[0][0] leaky_23[0][0]
conv_25 (Conv2D)	(None, None, None, 1 32768	add_7[0][0]
bnorm_25 (BatchNormalization)	(None, None, None, 1 512	conv_25[0][0]
leaky_25 (LeakyReLU)	(None, None, None, 1 0	bnorm_25[0][0]
conv_26 (Conv2D)	(None, None, None, 2 294912	leaky_25[0][0]
bnorm_26 (BatchNormalization)	(None, None, None, 2 1024	conv_26[0][0]
leaky_26 (LeakyReLU)	(None, None, None, 2 0	bnorm_26[0][0]
add_8 (Add)	(None, None, None, 2 0	add_7[0][0] leaky_26[0][0]
conv_28 (Conv2D)	(None, None, None, 1 32768	add_8[0][0]
bnorm_28 (BatchNormalization)	(None, None, None, 1 512	conv_28[0][0]
leaky_28 (LeakyReLU)	(None, None, None, 1 0	bnorm_28[0][0]
conv_29 (Conv2D)	(None, None, None, 2 294912	leaky_28[0][0]
bnorm_29 (BatchNormalization)	(None, None, None, 2 1024	conv_29[0][0]
leaky_29 (LeakyReLU)	(None, None, None, 2 0	bnorm_29[0][0]
add_9 (Add)	(None, None, None, 2 0	add_8[0][0] leaky_29[0][0]
conv_31 (Conv2D)	(None, None, None, 1 32768	add_9[0][0]

bnorm_31 (BatchNormalization)	(None, None, None, 1 512	conv_31[0][0]
leaky_31 (LeakyReLU)	(None, None, None, 1 0	bnorm_31[0][0]
conv_32 (Conv2D)	(None, None, None, 2 294912	leaky_31[0][0]
bnorm_32 (BatchNormalization)	(None, None, None, 2 1024	conv_32[0][0]
leaky_32 (LeakyReLU)	(None, None, None, 2 0	bnorm_32[0][0]
add_10 (Add)	(None, None, None, 2 0	add_9[0][0] leaky_32[0][0]
conv_34 (Conv2D)	(None, None, None, 1 32768	add_10[0][0]
bnorm_34 (BatchNormalization)	(None, None, None, 1 512	conv_34[0][0]
leaky_34 (LeakyReLU)	(None, None, None, 1 0	bnorm_34[0][0]
conv_35 (Conv2D)	(None, None, None, 2 294912	leaky_34[0][0]
bnorm_35 (BatchNormalization)	(None, None, None, 2 1024	conv_35[0][0]
leaky_35 (LeakyReLU)	(None, None, None, 2 0	bnorm_35[0][0]
add_11 (Add)	(None, None, None, 2 0	add_10[0][0] leaky_35[0][0]
zero_padding2d_4 (ZeroPadding2D	(None, None, None, 2 0	add_11[0][0]
conv_37 (Conv2D)	(None, None, None, 5 1179648	zero_padding2d_4[0][0]
bnorm_37 (BatchNormalization)	(None, None, None, 5 2048	conv_37[0][0]
leaky_37 (LeakyReLU)	(None, None, None, 5 0	bnorm_37[0][0]
conv_38 (Conv2D)	(None, None, None, 2 131072	leaky_37[0][0]
bnorm_38 (BatchNormalization)	(None, None, None, 2 1024	conv_38[0][0]
leaky_38 (LeakyReLU)	(None, None, None, 2 0	bnorm_38[0][0]
conv_39 (Conv2D)	(None, None, None, 5 1179648	leaky_38[0][0]

bnorm_39 (BatchNormalization)	(None, None, None, 5 2048)	conv_39[0][0]
leaky_39 (LeakyReLU)	(None, None, None, 5 0)	bnorm_39[0][0]
add_12 (Add)	(None, None, None, 5 0)	leaky_37[0][0] leaky_39[0][0]
conv_41 (Conv2D)	(None, None, None, 2 131072)	add_12[0][0]
bnorm_41 (BatchNormalization)	(None, None, None, 2 1024)	conv_41[0][0]
leaky_41 (LeakyReLU)	(None, None, None, 2 0)	bnorm_41[0][0]
conv_42 (Conv2D)	(None, None, None, 5 1179648)	leaky_41[0][0]
bnorm_42 (BatchNormalization)	(None, None, None, 5 2048)	conv_42[0][0]
leaky_42 (LeakyReLU)	(None, None, None, 5 0)	bnorm_42[0][0]
add_13 (Add)	(None, None, None, 5 0)	add_12[0][0] leaky_42[0][0]
conv_44 (Conv2D)	(None, None, None, 2 131072)	add_13[0][0]
bnorm_44 (BatchNormalization)	(None, None, None, 2 1024)	conv_44[0][0]
leaky_44 (LeakyReLU)	(None, None, None, 2 0)	bnorm_44[0][0]
conv_45 (Conv2D)	(None, None, None, 5 1179648)	leaky_44[0][0]
bnorm_45 (BatchNormalization)	(None, None, None, 5 2048)	conv_45[0][0]
leaky_45 (LeakyReLU)	(None, None, None, 5 0)	bnorm_45[0][0]
add_14 (Add)	(None, None, None, 5 0)	add_13[0][0] leaky_45[0][0]
conv_47 (Conv2D)	(None, None, None, 2 131072)	add_14[0][0]
bnorm_47 (BatchNormalization)	(None, None, None, 2 1024)	conv_47[0][0]
leaky_47 (LeakyReLU)	(None, None, None, 2 0)	bnorm_47[0][0]

conv_48 (Conv2D)	(None, None, None, 5 1179648	leaky_47[0][0]
bnorm_48 (BatchNormalization)	(None, None, None, 5 2048	conv_48[0][0]
leaky_48 (LeakyReLU)	(None, None, None, 5 0	bnorm_48[0][0]
add_15 (Add)	(None, None, None, 5 0	add_14[0][0] leaky_48[0][0]
conv_50 (Conv2D)	(None, None, None, 2 131072	add_15[0][0]
bnorm_50 (BatchNormalization)	(None, None, None, 2 1024	conv_50[0][0]
leaky_50 (LeakyReLU)	(None, None, None, 2 0	bnorm_50[0][0]
conv_51 (Conv2D)	(None, None, None, 5 1179648	leaky_50[0][0]
bnorm_51 (BatchNormalization)	(None, None, None, 5 2048	conv_51[0][0]
leaky_51 (LeakyReLU)	(None, None, None, 5 0	bnorm_51[0][0]
add_16 (Add)	(None, None, None, 5 0	add_15[0][0] leaky_51[0][0]
conv_53 (Conv2D)	(None, None, None, 2 131072	add_16[0][0]
bnorm_53 (BatchNormalization)	(None, None, None, 2 1024	conv_53[0][0]
leaky_53 (LeakyReLU)	(None, None, None, 2 0	bnorm_53[0][0]
conv_54 (Conv2D)	(None, None, None, 5 1179648	leaky_53[0][0]
bnorm_54 (BatchNormalization)	(None, None, None, 5 2048	conv_54[0][0]
leaky_54 (LeakyReLU)	(None, None, None, 5 0	bnorm_54[0][0]
add_17 (Add)	(None, None, None, 5 0	add_16[0][0] leaky_54[0][0]
conv_56 (Conv2D)	(None, None, None, 2 131072	add_17[0][0]
bnorm_56 (BatchNormalization)	(None, None, None, 2 1024	conv_56[0][0]

leaky_56 (LeakyReLU)	(None, None, None, 2 0	bnorm_56[0][0]
conv_57 (Conv2D)	(None, None, None, 5 1179648	leaky_56[0][0]
bnorm_57 (BatchNormalization)	(None, None, None, 5 2048	conv_57[0][0]
leaky_57 (LeakyReLU)	(None, None, None, 5 0	bnorm_57[0][0]
add_18 (Add)	(None, None, None, 5 0	add_17[0][0] leaky_57[0][0]
conv_59 (Conv2D)	(None, None, None, 2 131072	add_18[0][0]
bnorm_59 (BatchNormalization)	(None, None, None, 2 1024	conv_59[0][0]
leaky_59 (LeakyReLU)	(None, None, None, 2 0	bnorm_59[0][0]
conv_60 (Conv2D)	(None, None, None, 5 1179648	leaky_59[0][0]
bnorm_60 (BatchNormalization)	(None, None, None, 5 2048	conv_60[0][0]
leaky_60 (LeakyReLU)	(None, None, None, 5 0	bnorm_60[0][0]
add_19 (Add)	(None, None, None, 5 0	add_18[0][0] leaky_60[0][0]
zero_padding2d_5 (ZeroPadding2D	(None, None, None, 5 0	add_19[0][0]
conv_62 (Conv2D)	(None, None, None, 1 4718592	zero_padding2d_5[0][0]
bnorm_62 (BatchNormalization)	(None, None, None, 1 4096	conv_62[0][0]
leaky_62 (LeakyReLU)	(None, None, None, 1 0	bnorm_62[0][0]
conv_63 (Conv2D)	(None, None, None, 5 524288	leaky_62[0][0]
bnorm_63 (BatchNormalization)	(None, None, None, 5 2048	conv_63[0][0]
leaky_63 (LeakyReLU)	(None, None, None, 5 0	bnorm_63[0][0]
conv_64 (Conv2D)	(None, None, None, 1 4718592	leaky_63[0][0]

bnorm_64 (BatchNormalization)	(None, None, None, 1 4096	conv_64[0][0]
leaky_64 (LeakyReLU)	(None, None, None, 1 0	bnorm_64[0][0]
add_20 (Add)	(None, None, None, 1 0	leaky_62[0][0] leaky_64[0][0]
conv_66 (Conv2D)	(None, None, None, 5 524288	add_20[0][0]
bnorm_66 (BatchNormalization)	(None, None, None, 5 2048	conv_66[0][0]
leaky_66 (LeakyReLU)	(None, None, None, 5 0	bnorm_66[0][0]
conv_67 (Conv2D)	(None, None, None, 1 4718592	leaky_66[0][0]
bnorm_67 (BatchNormalization)	(None, None, None, 1 4096	conv_67[0][0]
leaky_67 (LeakyReLU)	(None, None, None, 1 0	bnorm_67[0][0]
add_21 (Add)	(None, None, None, 1 0	add_20[0][0] leaky_67[0][0]
conv_69 (Conv2D)	(None, None, None, 5 524288	add_21[0][0]
bnorm_69 (BatchNormalization)	(None, None, None, 5 2048	conv_69[0][0]
leaky_69 (LeakyReLU)	(None, None, None, 5 0	bnorm_69[0][0]
conv_70 (Conv2D)	(None, None, None, 1 4718592	leaky_69[0][0]
bnorm_70 (BatchNormalization)	(None, None, None, 1 4096	conv_70[0][0]
leaky_70 (LeakyReLU)	(None, None, None, 1 0	bnorm_70[0][0]
add_22 (Add)	(None, None, None, 1 0	add_21[0][0] leaky_70[0][0]
conv_72 (Conv2D)	(None, None, None, 5 524288	add_22[0][0]
bnorm_72 (BatchNormalization)	(None, None, None, 5 2048	conv_72[0][0]
leaky_72 (LeakyReLU)	(None, None, None, 5 0	bnorm_72[0][0]

conv_73 (Conv2D)	(None, None, None, 1 4718592	leaky_72[0][0]
bnorm_73 (BatchNormalization)	(None, None, None, 1 4096	conv_73[0][0]
leaky_73 (LeakyReLU)	(None, None, None, 1 0	bnorm_73[0][0]
add_23 (Add)	(None, None, None, 1 0	add_22[0][0] leaky_73[0][0]
conv_75 (Conv2D)	(None, None, None, 5 524288	add_23[0][0]
bnorm_75 (BatchNormalization)	(None, None, None, 5 2048	conv_75[0][0]
leaky_75 (LeakyReLU)	(None, None, None, 5 0	bnorm_75[0][0]
conv_76 (Conv2D)	(None, None, None, 1 4718592	leaky_75[0][0]
bnorm_76 (BatchNormalization)	(None, None, None, 1 4096	conv_76[0][0]
leaky_76 (LeakyReLU)	(None, None, None, 1 0	bnorm_76[0][0]
conv_77 (Conv2D)	(None, None, None, 5 524288	leaky_76[0][0]
bnorm_77 (BatchNormalization)	(None, None, None, 5 2048	conv_77[0][0]
leaky_77 (LeakyReLU)	(None, None, None, 5 0	bnorm_77[0][0]
conv_78 (Conv2D)	(None, None, None, 1 4718592	leaky_77[0][0]
bnorm_78 (BatchNormalization)	(None, None, None, 1 4096	conv_78[0][0]
leaky_78 (LeakyReLU)	(None, None, None, 1 0	bnorm_78[0][0]
conv_79 (Conv2D)	(None, None, None, 5 524288	leaky_78[0][0]
bnorm_79 (BatchNormalization)	(None, None, None, 5 2048	conv_79[0][0]
leaky_79 (LeakyReLU)	(None, None, None, 5 0	bnorm_79[0][0]
conv_84 (Conv2D)	(None, None, None, 2 131072	leaky_79[0][0]
bnorm_84 (BatchNormalization)	(None, None, None, 2 1024	conv_84[0][0]

leaky_84 (LeakyReLU)	(None, None, None, 2 0	bnorm_84[0][0]
up_sampling2d_1 (UpSampling2D)	(None, None, None, 2 0	leaky_84[0][0]
concatenate_1 (Concatenate)	(None, None, None, 7 0	up_sampling2d_1[0][0] add_19[0][0]
conv_87 (Conv2D)	(None, None, None, 2 196608	concatenate_1[0][0]
bnnorm_87 (BatchNormalization)	(None, None, None, 2 1024	conv_87[0][0]
leaky_87 (LeakyReLU)	(None, None, None, 2 0	bnnorm_87[0][0]
conv_88 (Conv2D)	(None, None, None, 5 1179648	leaky_87[0][0]
bnnorm_88 (BatchNormalization)	(None, None, None, 5 2048	conv_88[0][0]
leaky_88 (LeakyReLU)	(None, None, None, 5 0	bnnorm_88[0][0]
conv_89 (Conv2D)	(None, None, None, 2 131072	leaky_88[0][0]
bnnorm_89 (BatchNormalization)	(None, None, None, 2 1024	conv_89[0][0]
leaky_89 (LeakyReLU)	(None, None, None, 2 0	bnnorm_89[0][0]
conv_90 (Conv2D)	(None, None, None, 5 1179648	leaky_89[0][0]
bnnorm_90 (BatchNormalization)	(None, None, None, 5 2048	conv_90[0][0]
leaky_90 (LeakyReLU)	(None, None, None, 5 0	bnnorm_90[0][0]
conv_91 (Conv2D)	(None, None, None, 2 131072	leaky_90[0][0]
bnnorm_91 (BatchNormalization)	(None, None, None, 2 1024	conv_91[0][0]
leaky_91 (LeakyReLU)	(None, None, None, 2 0	bnnorm_91[0][0]
conv_96 (Conv2D)	(None, None, None, 1 32768	leaky_91[0][0]
bnnorm_96 (BatchNormalization)	(None, None, None, 1 512	conv_96[0][0]
leaky_96 (LeakyReLU)	(None, None, None, 1 0	bnnorm_96[0][0]

up_sampling2d_2 (UpSampling2D)	(None, None, None, 1 0	leaky_96[0][0]
concatenate_2 (Concatenate)	(None, None, None, 3 0	up_sampling2d_2[0][0] add_11[0][0]
conv_99 (Conv2D)	(None, None, None, 1 49152	concatenate_2[0][0]
bnorm_99 (BatchNormalization)	(None, None, None, 1 512	conv_99[0][0]
leaky_99 (LeakyReLU)	(None, None, None, 1 0	bnorm_99[0][0]
conv_100 (Conv2D)	(None, None, None, 2 294912	leaky_99[0][0]
bnorm_100 (BatchNormalization)	(None, None, None, 2 1024	conv_100[0][0]
leaky_100 (LeakyReLU)	(None, None, None, 2 0	bnorm_100[0][0]
conv_101 (Conv2D)	(None, None, None, 1 32768	leaky_100[0][0]
bnorm_101 (BatchNormalization)	(None, None, None, 1 512	conv_101[0][0]
leaky_101 (LeakyReLU)	(None, None, None, 1 0	bnorm_101[0][0]
conv_102 (Conv2D)	(None, None, None, 2 294912	leaky_101[0][0]
bnorm_102 (BatchNormalization)	(None, None, None, 2 1024	conv_102[0][0]
leaky_102 (LeakyReLU)	(None, None, None, 2 0	bnorm_102[0][0]
conv_103 (Conv2D)	(None, None, None, 1 32768	leaky_102[0][0]
bnorm_103 (BatchNormalization)	(None, None, None, 1 512	conv_103[0][0]
leaky_103 (LeakyReLU)	(None, None, None, 1 0	bnorm_103[0][0]
conv_80 (Conv2D)	(None, None, None, 1 4718592	leaky_79[0][0]
conv_92 (Conv2D)	(None, None, None, 5 1179648	leaky_91[0][0]
conv_104 (Conv2D)	(None, None, None, 2 294912	leaky_103[0][0]
bnorm_80 (BatchNormalization)	(None, None, None, 1 4096	conv_80[0][0]

bnorm_92 (BatchNormalization)	(None, None, None, 5 2048)	conv_92[0][0]
bnorm_104 (BatchNormalization)	(None, None, None, 2 1024)	conv_104[0][0]
leaky_80 (LeakyReLU)	(None, None, None, 1 0)	bnorm_80[0][0]
leaky_92 (LeakyReLU)	(None, None, None, 5 0)	bnorm_92[0][0]
leaky_104 (LeakyReLU)	(None, None, None, 2 0)	bnorm_104[0][0]
conv_81 (Conv2D)	(None, None, None, 2 261375)	leaky_80[0][0]
conv_93 (Conv2D)	(None, None, None, 2 130815)	leaky_92[0][0]
conv_105 (Conv2D)	(None, None, None, 2 65535)	leaky_104[0][0]
<hr/>		
Total params:	62,001,757	
Trainable params:	61,949,149	
Non-trainable params:	52,608	

YOLOv3 Notes

Total yolov3.layers: 252
Convolutional yolov3.layers: 106
Three image scales: layers 79-82 (stride 32), 91-94 (stride 16), 103-106 (stride 8)
Detection layer outputs: conv_81, conv_93, conv_105

Total params: 62,001,757
Trainable params: 61,949,149
Non-trainable params: 52,608

YOLOv3 Layers (0 - 105)

- See YOLOv3 Keras model definition in ./keras_yolo3/yolo.py

Detection layer outputs

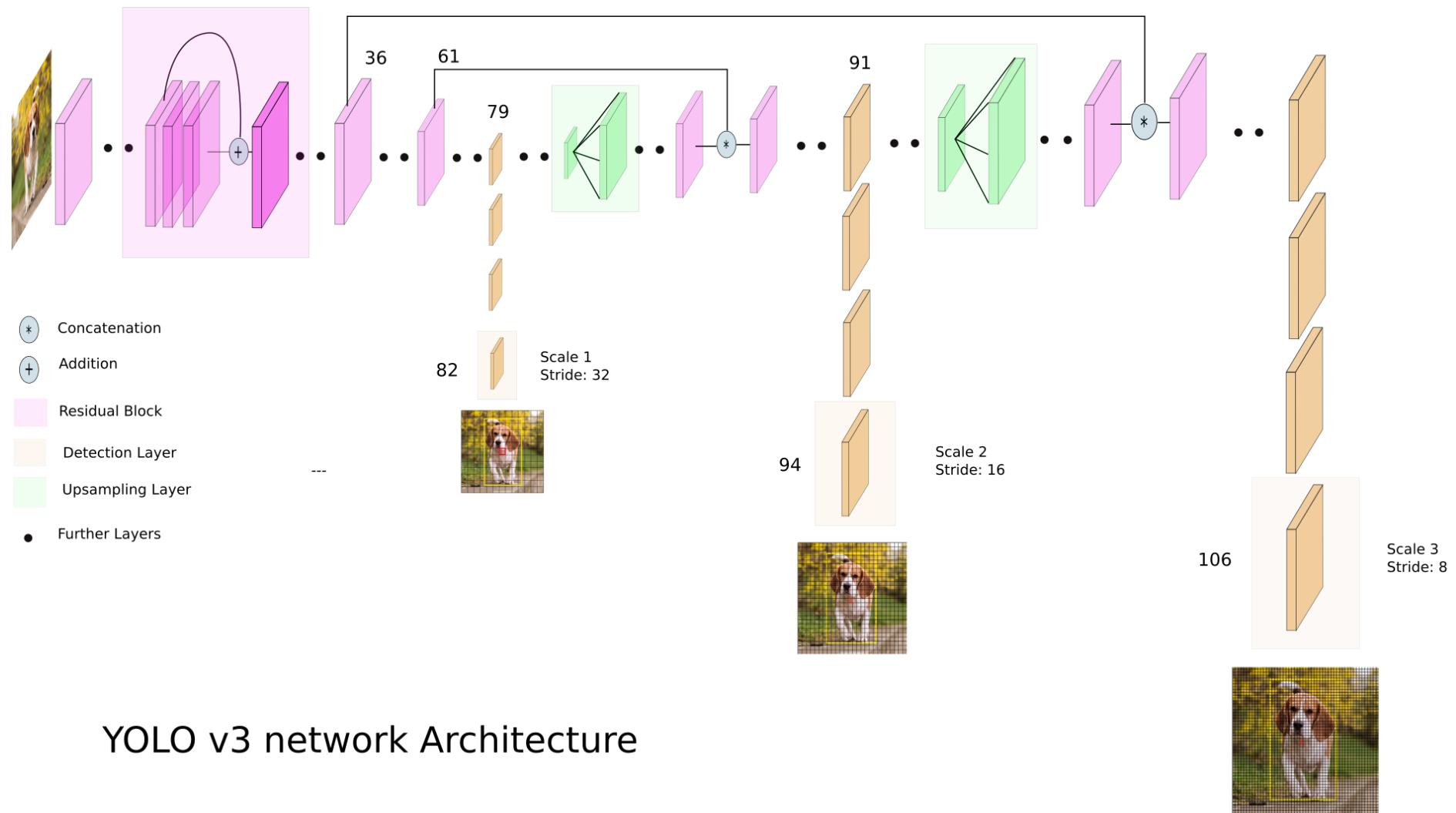
```
conv_81 (Conv2D)           (None, None, None, 2 261375      leaky_80[0][0]           conv_93 (Co
nv2D)                   (None, None, None, 2 130815      leaky_92[0][0]           conv_105 (Conv2D)
(None, None, None, 2 65535      leaky_104[0][0]
```

YOLOv3 Summary

Sample layers

```
<keras.engine.training.Model object at 0xb4a178a58>
Layer (type)          Output Shape         Param #     Connected to
input_3 (InputLayer)   (None, None, None, 3 0
conv_0 (Conv2D)        (None, None, None, 3 864      input_3[0][0]
bnorm_0 (BatchNormalization) (None, None, None, 3 128      conv_0[0][0]
leaky_0 (LeakyReLU)    (None, None, None, 3 0      bnorm_0[0][0]
zero_padding2d_11 (ZeroPadding2 (None, None, None, 3 0      leaky_0[0][0]
...
add_47 (Add)          (None, None, None, 6 0      leaky_1[0][0]
leaky_3[0][0]
zero_padding2d_12 (ZeroPadding2 (None, None, None, 6 0      add_47[0][0]
conv_5 (Conv2D)        (None, None, None, 1 73728      zero_padding2d_12[0][0]
bnorm_5 (BatchNormalization) (None, None, None, 1 512      conv_5[0][0]
leaky_5 (LeakyReLU)    (None, None, None, 1 0      bnorm_5[0][0]
...
conv_84 (Conv2D)       (None, None, None, 2 131072      leaky_79[0][0]      bnorm_84 (Batch
atchNormalization) (None, None, None, 2 1024      conv_84[0][0]      leaky_84 (LeakyReLU)
(None, None, None, 2 0      bnorm_84[0][0]      up_sampling2d_5 (UpSampling2D) (None, Non
e, None, 2 0      leaky_84[0][0]
```

YOLOv3 Architecture



YOLO v3 network Architecture

Source: Ayoosh Kathuria, YOLOv3 YOLO: <https://pjreddie.com/yolo/> (<https://pjreddie.com/yolo/>)

```
In [6]: # YOLO parameters for image pre-processing and object detection

# set some parameters
net_h, net_w = 416, 416
obj_thresh, nms_thresh = 0.5, 0.45
anchors = [[116,90, 156,198, 373,326], [30,61, 62,45, 59,119], [10,13, 16,30, 33,23]]

# YOLOv3 object classes: 80 MSCOCO classes
labels = ["person", "bicycle", "car", "motorbike", "aeroplane", "bus", "train", "truck",
"boat", "traffic light", "fire hydrant", "stop sign", "parking meter", "bench",
"bird", "cat", "dog", "horse", "sheep", "cow", "elephant", "bear", "zebra", "giraffe",
"backpack", "umbrella", "handbag", "tie", "suitcase", "frisbee", "skis", "snowboard",
"sports ball", "kite", "baseball bat", "baseball glove", "skateboard", "surfboard",
"tennis racket", "bottle", "wine glass", "cup", "fork", "knife", "spoon", "bowl", "banana",
"apple", "sandwich", "orange", "broccoli", "carrot", "hot dog", "pizza", "donut", "cake",
"chair", "sofa", "pottedplant", "bed", "diningtable", "toilet", "tvmonitor", "laptop", "mouse",
"remote", "keyboard", "cell phone", "microwave", "oven", "toaster", "sink", "refrigerator",
"book", "clock", "vase", "scissors", "teddy bear", "hair drier", "toothbrush"]
```

```
In [7]: # Sample images
image_path = './keras_yolo3_datasets/BCCD_Dataset/example.jpg'
image_path = './images/zebra.jpg'

image_dir1 = './data/images/'
image_fns1 = ['african_elephant.jpg', 'african_elephant_people.jpg', 'african_elephant_jEEP',
              'zebra.jpg', 'dog.jpg', 'eagle.jpg', 'giraffe.jpg', 'horses.jpg',
              'kites_people.jpg', 'person.jpg']

# MS-COCO
image_dir2 = '../..../datasets/mscoco/val2017/'
image_fns2 = ['00000000285.jpg', '000000000785.jpg', '000000004765.jpg', '000000014380.jpg']

# West Palm Beach
image_dir3 = './data/images/wpb/'
image_fns3 = ['wpb_marina_01.jpg', 'wpb_clematis_01.jpg']

# Art
image_dir4 = './data/images/art/'
image_fns4 = ['dama_sentada.jpg', 'scream.jpg']

image_path = image_dir1+image_fns1[1]
print("image_path:", image_path)
```

```
image_path: ./data/images/african_elephant_people.jpg
```

YOLOv3 - image pre-processing

- yolov3.preprocess_input()
- Convert input image to defined width, height (416 x 416)

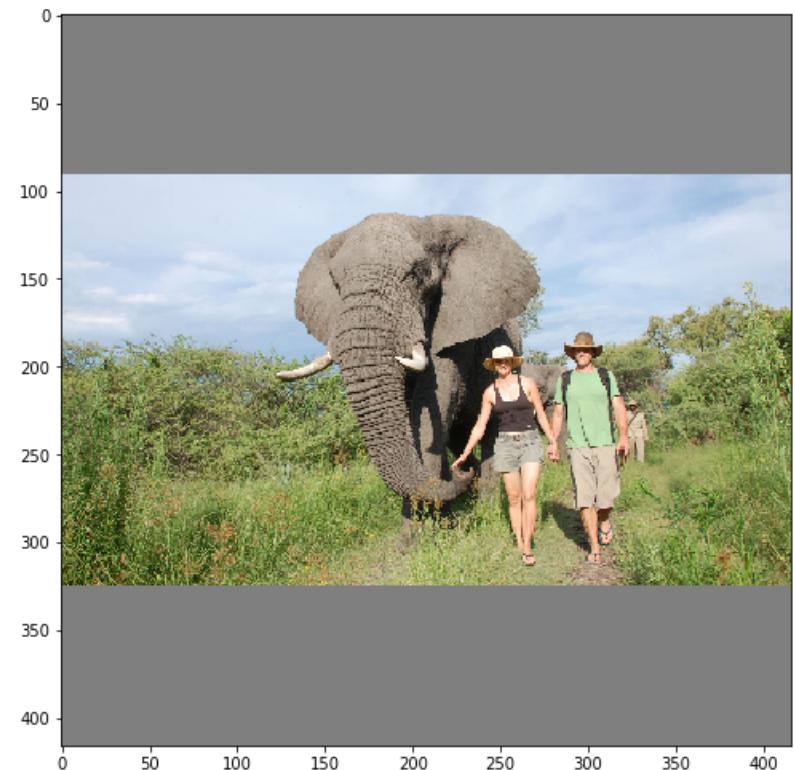
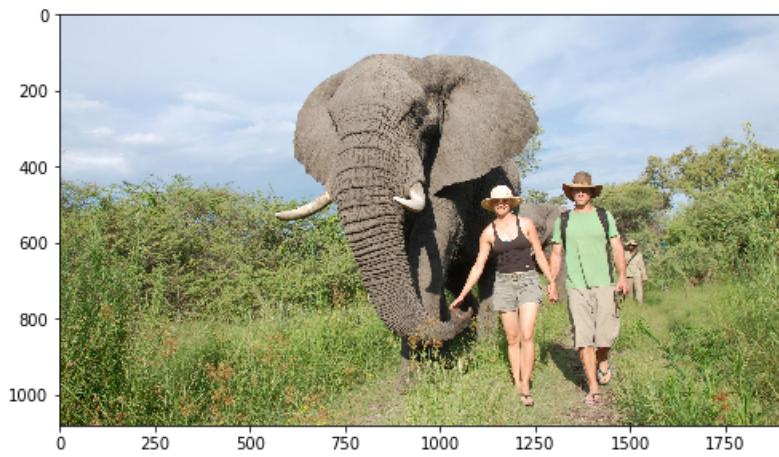
```
In [8]: # YOLOv3 read & preprocess the sample image
image = cv2.imread(image_path)

image_h, image_w, _ = image.shape
new_image = xp_yolov3.preprocess_input(image, net_h, net_w)
```

```
In [9]: # image, preprocessed new_image
```

```
print("image.shape:", image.shape, end=' - ')
print("new_image.shape:", new_image.shape)
plt.figure(figsize=(18,10))
plt.subplot(121)
plt.imshow(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
plt.subplot(122)
plt.imshow(new_image[0])
plt.show();
```

```
image.shape: (1080, 1920, 3) - new_image.shape: (1, 416, 416, 3)
```



In [10]: # YOLOv3 helper functions

```
# Read and preprocess image
def yolov3_read_and_show(image_path):
    # image_path = image_dir4+image_fns4[0]
    image = cv2.imread(image_path)
    print("image_path:", image_path)
    print("image.shape:", image.shape)
    # Show image
    plt.figure(figsize=(18,10))
    plt.imshow(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
    plt.show();
    return image

# Object detection and display of annotated image
def yolov3_detect_and_show(image, image_path):
    print("YOLOv3 predicting image_path:", image_path)
    image_h, image_w, _ = image.shape
    new_image = xp_yolov3.preprocess_input(image, net_h, net_w)
    yolos = yolov3.predict(new_image)
    boxes = []
    for i in range(len(yolos)):
        boxes += xp_yolov3.decode_netout(yolos[i][0], anchors[i],
                                         obj_thresh, nms_thresh, net_h, net_w)
    xp_yolov3.correct_yolo_boxes(boxes, image_h, image_w, net_h, net_w)
    xp_yolov3.do_nms(boxes, nms_thresh)
    xp_yolov3.draw_boxes(image, boxes, labels, obj_thresh)
    detected_image_path = image_path[:-4] + '_detected' + image_path[-4:]
    cv2.imwrite(detected_image_path, (image).astype('uint8'))
    # Show detected bounding boxes on image
    image = cv2.imread(detected_image_path)
    plt.figure(figsize=(18,10))
    plt.imshow(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
    plt.show()
    return boxes

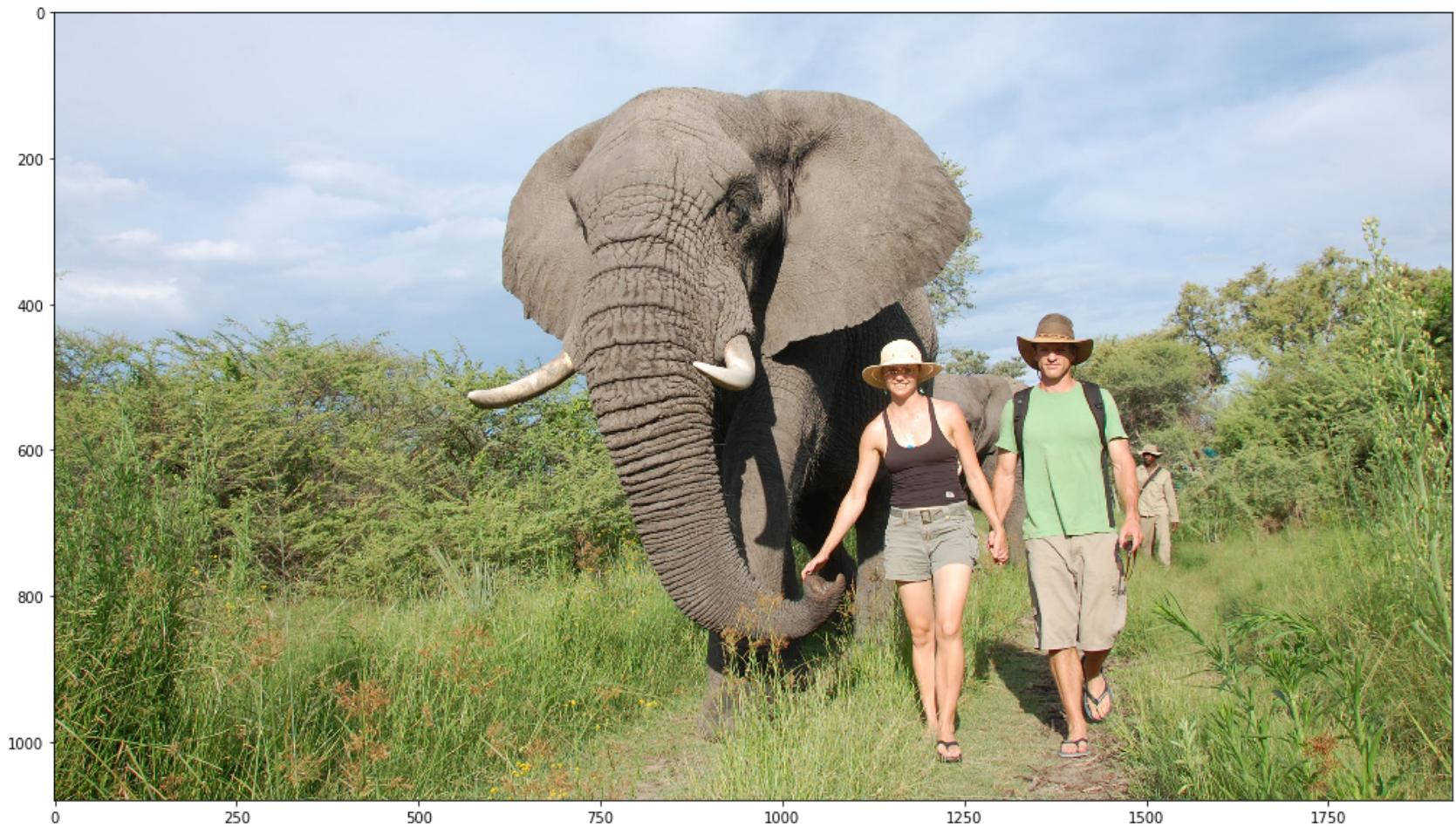
# Print object predictions and detected image; Multi-class, multi-object proposal per box
def print_predictions(boxes, labels, obj_thresh):
    print("Predictions on image (object threshold: {})".format(obj_thresh))
    obj_num = 0
    for b, box in enumerate(boxes):
        label_str = ''
```

```
label = -1
for i in range(len(labels)):
    if box.classes[i] > obj_thresh:
        obj_num += 1
        label_str += labels[i]
        label = i
        info_str = 'object ' + str(obj_num) + ' box ' + str(b)
        info_str += ' - ' + labels[i] + ' (class ' + str(i) + '): '
        print(info_str + str(box.classes[i]*100) + '%')
return obj_num
```

YOLOv3 Object Detection - Elephant and people

```
In [11]: # Elephant image  
image_path = image_dir1+image_fns1[1]  
# Object detection  
image = yolov3_read_and_show(image_path)
```

```
image_path: ./data/images/african_elephant_people.jpg  
image.shape: (1080, 1920, 3)
```



```
In [12]: # Object detection
```

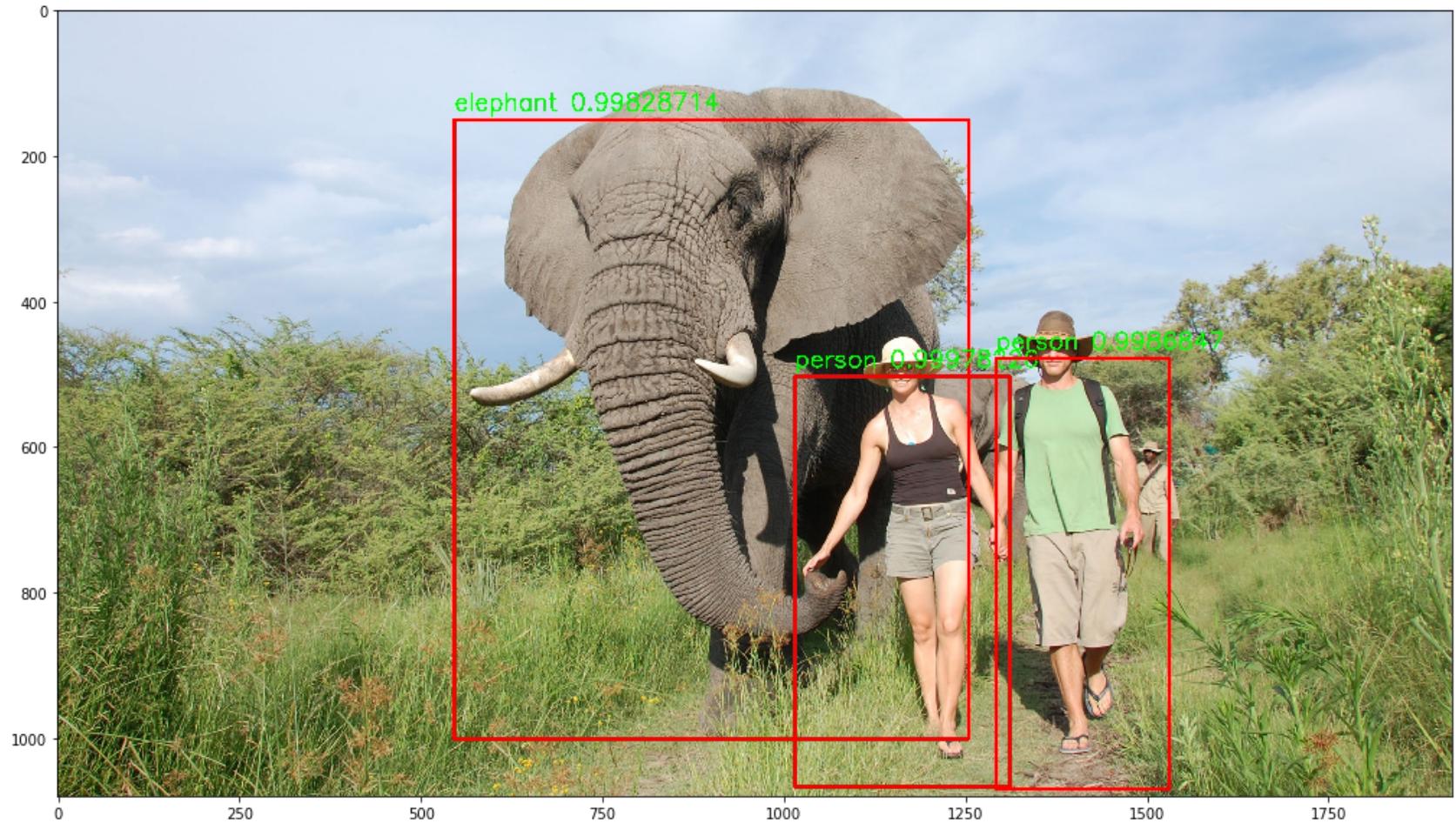
```
boxes = yolov3_detect_and_show(image, image_path)
```

YOLOv3 predicting image_path: ./data/images/african_elephant_people.jpg

elephant: 99.82871413230896%

person: 99.97822642326355%

person: 99.86847043037415%



```
In [13]: # Print yolo boxes and classes
objects_total = print_predictions(boxes, labels, obj_thresh)
print("\nobjects_total:", objects_total)
```

```
Predictions on image (object threshold: 0.5)
object 1 box 253 - elephant (class 20): 99.82871413230896%
object 2 box 1724 - person (class 0): 99.97822642326355%
object 3 box 1736 - person (class 0): 99.86847043037415%

objects_total: 3
```

YOLOv3 Object Detection - West Palm Beach

```
In [14]: # WPB downtown image
```

```
image_path = image_dir3+image_fns3[1]
image = yolov3_read_and_show(image_path)
```

```
image_path: ./data/images/wpb/wpb_clematis_01.jpg
image.shape: (500, 800, 3)
```



In [15]: # Object detection

```
boxes = yolov3_detect_and_show(image, image_path)
```

```
YOLOv3 predicting image_path: ./data/images/wpb/wpb_clematis_01.jpg
bus: 99.76127743721008%
person: 99.40769672393799%
person: 95.56765556335449%
bicycle: 99.69530701637268%
traffic light: 87.71517276763916%
traffic light: 84.3330442905426%
bus: 63.52843642234802%
car: 68.71739029884338%
car: 97.94415235519409%
car: 91.70721173286438%
motorbike: 67.31112599372864%
```



```
In [16]: # Print yolo boxes and classes
objects_total = print_predictions(boxes, labels, obj_thresh)
print("\nobjects_total:", objects_total)
```

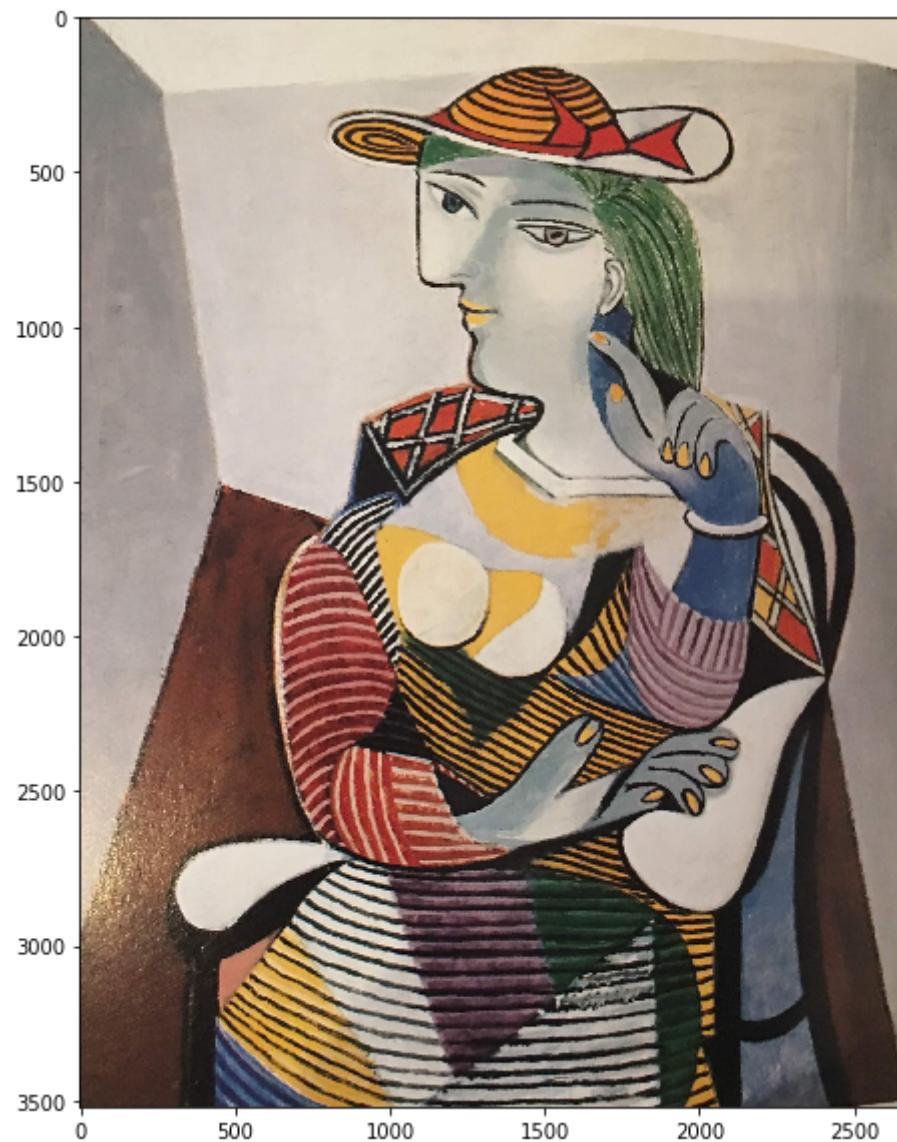
```
Predictions on image (object threshold: 0.5)
object 1 box 282 - bus (class 5): 99.76127743721008%
object 2 box 1878 - person (class 0): 99.40769672393799%
object 3 box 1893 - person (class 0): 95.56765556335449%
object 4 box 1957 - bicycle (class 1): 99.69530701637268%
object 5 box 4661 - traffic light (class 9): 87.71517276763916%
object 6 box 4847 - traffic light (class 9): 84.3330442905426%
object 7 box 6755 - bus (class 5): 63.52843642234802%
object 8 box 7078 - car (class 2): 68.71739029884338%
object 9 box 7081 - car (class 2): 97.94415235519409%
object 10 box 7286 - car (class 2): 91.70721173286438%
object 11 box 7454 - motorbike (class 3): 67.31112599372864%

objects_total: 11
```

YOLOv3 Object Detection - Dama sentada 1937

```
In [17]: # Dama image (Picasso)
image_path = image_dir4+image_fns4[0]
# Object detection
image = yolov3_read_and_show(image_path)
```

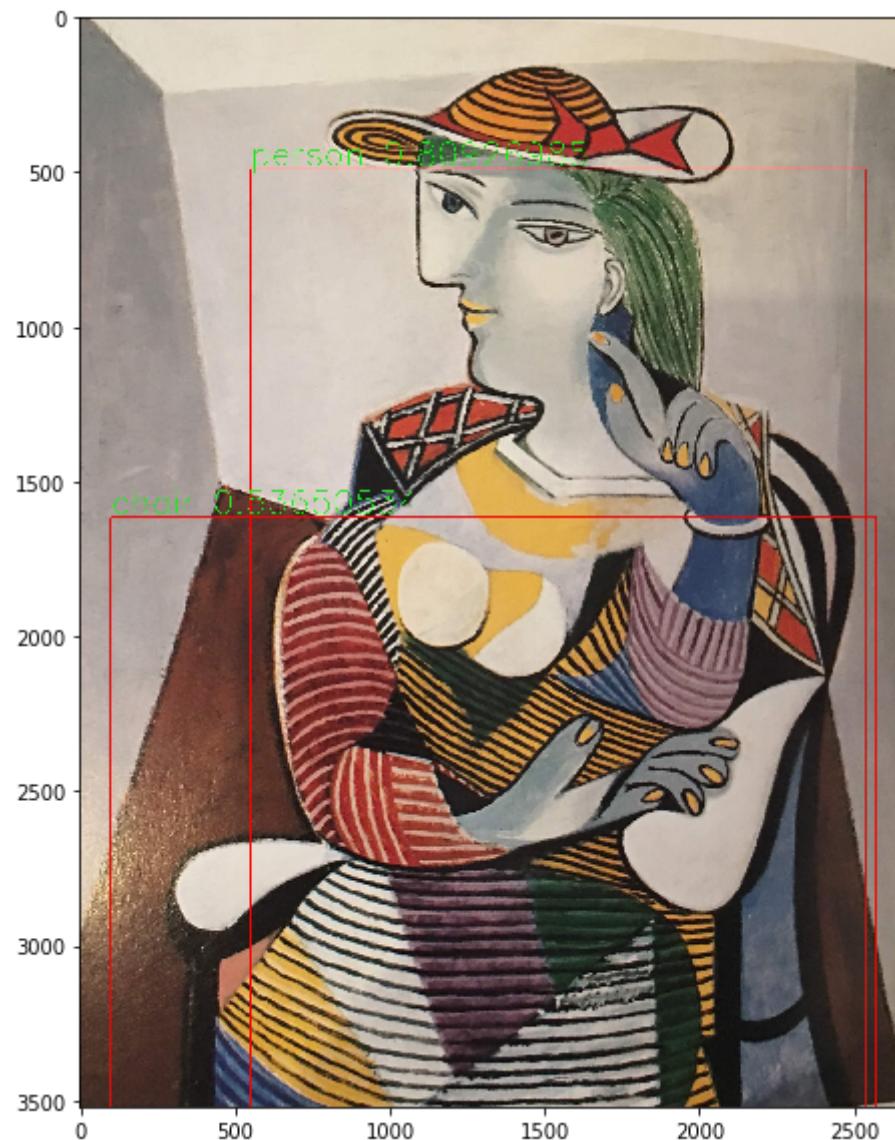
```
image_path: ./data/images/art/dama_sentada.jpg
image.shape: (3520, 2666, 3)
```



```
In [18]: # Object detection
```

```
boxes = yolov3_detect_and_show(image, image_path)
```

```
YOLOv3 predicting image_path: ./data/images/art/dama_sentada.jpg
person: 80.92698454856873%
chair: 53.65053415298462%
```



```
In [19]: # Print yolo boxes and classes
objects_total = print_predictions(boxes, labels, obj_thresh)
print("\nobjects_total:", objects_total)
```

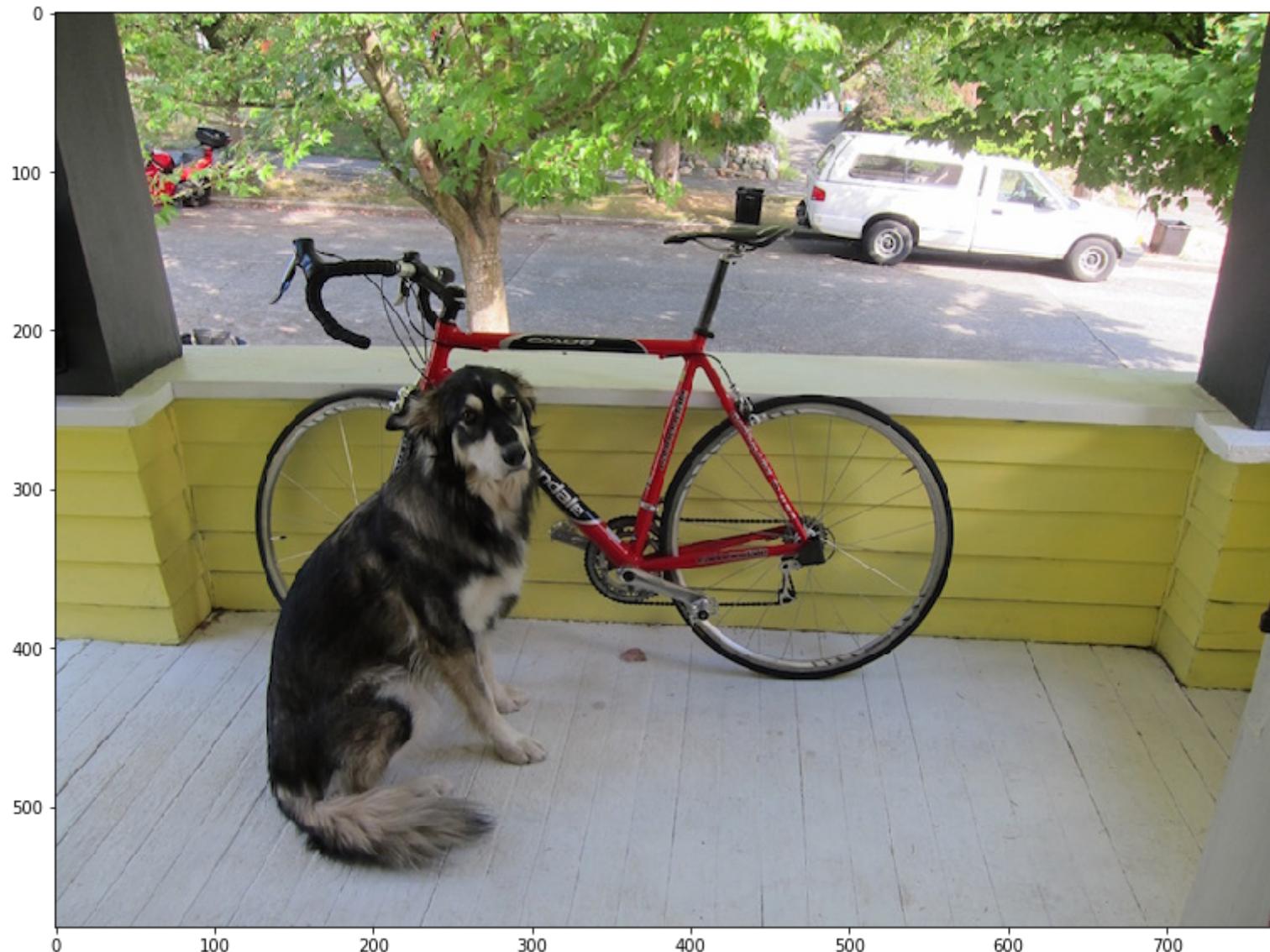
```
Predictions on image (object threshold: 0.5)
object 1 box 296 - person (class 0): 80.92698454856873%
object 2 box 371 - chair (class 56): 53.65053415298462%

objects_total: 2
```

Run YOLOv3 Object Detection - Dog

```
In [20]: # Dog image (Redmon)
image_path = image_dir1+image_fns1[4]
# Object detection
image = yolov3_read_and_show(image_path)
```

```
image_path: ./data/images/dog.jpg
image.shape: (576, 768, 3)
```



```
In [21]: # Object detection
```

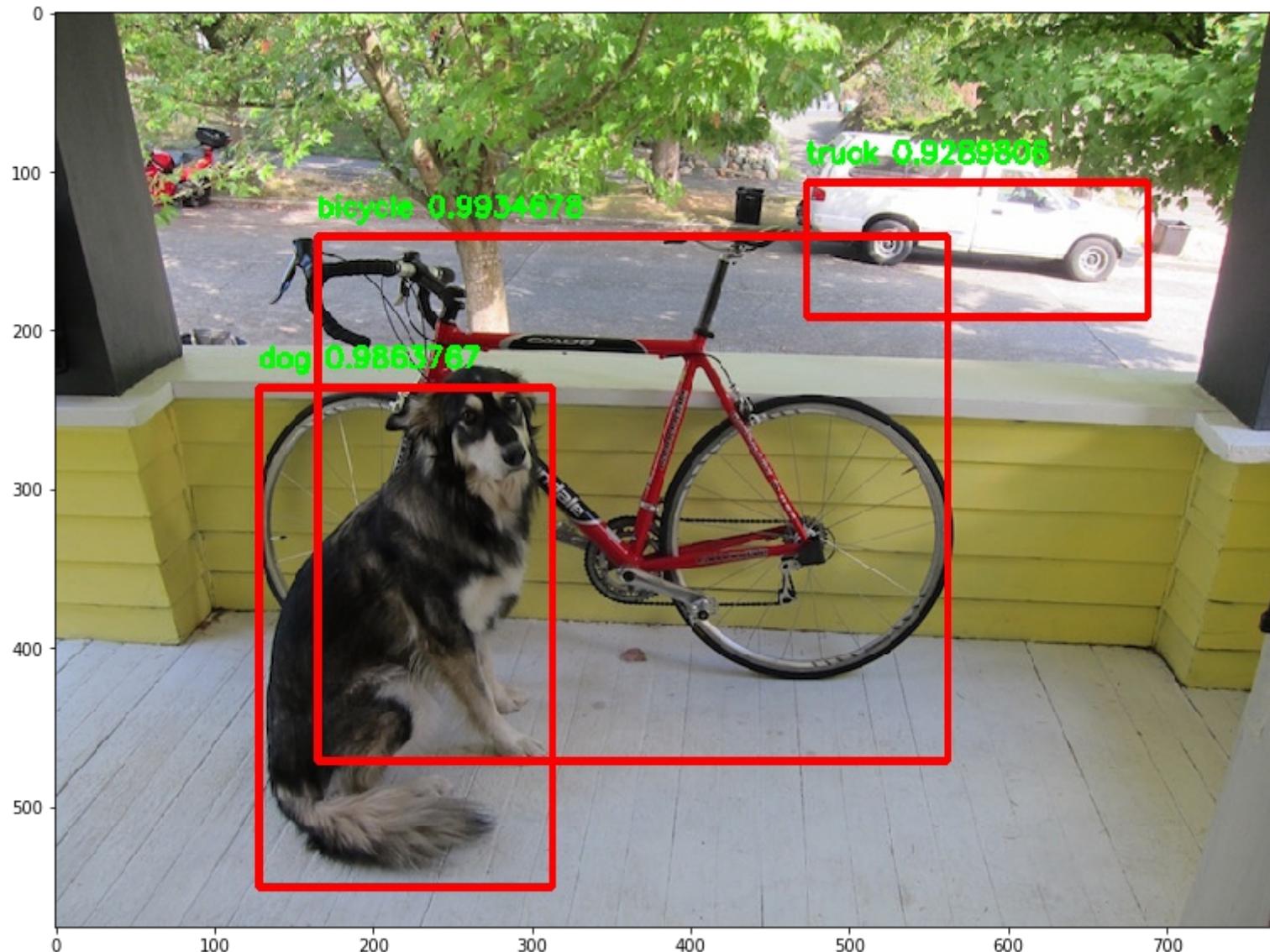
```
boxes = yolov3_detect_and_show(image, image_path)
```

YOLOv3 predicting image_path: ./data/images/dog.jpg

bicycle: 99.34678077697754%

dog: 98.6376702785492%

truck: 92.89805889129639%



Notes

YOLOv3 Training is done in a separate Jupyter notebook.

- See: `yolov3_object_detector_train.ipynb`.

YOLOv3 model trained using the DarkNet code base on the MSCOCO dataset.

Output of the model:

- encoded candidate bounding boxes from three different grid scale sizes
- bounding boxes defined in the context of anchor boxes,
chosen based on an analysis of the size of objects in the MSCOCO dataset.

In []: