

# How to train YOLOv3 to detect custom objects



Manivannan Murugavel [Follow](#)  
Jun 23, 2018 · 4 min read

This tutorial is how to train cat and dog object using Yolo-v3

## YOLO V3 Details—Feature Extractor:

We use a new network for performing feature extraction. Our new network is a hybrid approach between the network used in YOLOv2, Darknet-19, and that newfangled residual network stuff. Our network uses successive  $3 \times 3$  and  $1 \times 1$  convolutional layers but now has some shortcut connections as well and is significantly larger. It has 53 convolutional layers so we call it... wait for it..... Darknet-53! If you want to read about yolo v3 please click [here](#).

	Type	Filters	Size	Output
	Convolutional	32	$3 \times 3$	$256 \times 256$
	Convolutional	64	$3 \times 3 / 2$	$128 \times 128$
	Convolutional	32	$1 \times 1$	
1x	Convolutional	64	$3 \times 3$	
	Residual			$128 \times 128$
	Convolutional	128	$3 \times 3 / 2$	$64 \times 64$
	Convolutional	64	$1 \times 1$	
2x	Convolutional	128	$3 \times 3$	
	Residual			$64 \times 64$
	Convolutional	256	$3 \times 3 / 2$	$32 \times 32$
	Convolutional	128	$1 \times 1$	
8x	Convolutional	256	$3 \times 3$	
	Residual			$32 \times 32$
	Convolutional	512	$3 \times 3 / 2$	$16 \times 16$
	Convolutional	256	$1 \times 1$	
8x	Convolutional	512	$3 \times 3$	
	Residual			$16 \times 16$
	Convolutional	1024	$3 \times 3 / 2$	$8 \times 8$
	Convolutional	512	$1 \times 1$	
4x	Convolutional	1024	$3 \times 3$	
	Residual			$8 \times 8$
	Avrgpool			Global
	Connected			1000
	Softmax			

Table 1. Darknet-53.

Yolo v3 - Architecture

## Dataset Preparation:

The dataset preparation is similar to How to train YOLOv2 to detect custom objects blog in medium and here is the [link](#).

Please follow the above link for dataset preparation for yolo v3 and follow the link until before the Preparing YOLOv2 configuration files .

## Training:

Download Pretrained Convolutional Weights:

For training we use convolutional weights that are pre-trained on Imagenet. We use weights from the [darknet53](#) model. You can just download the weights for the convolutional layers [here \(76 MB\)](#).

## Preparing YOLOv3 configuration files

YOLOv3 needs certain specific files to know how and what to train. We'll be creating these three files (.data, .names and .cfg) and also explain the yolov3.cfg and yolov3-tiny.cfg.

- `cfg/cat-dog-obj.data`
- `cfg/cat-dog-obj.names`

First let's prepare the YOLOv3 .data and .names file. Let's start by creating `cat-dog-obj.data` and filling it with this content. This basically says that we are training one class, what the train and validation set files are and what file contains the names for the categories we want to detect.

```
classes= 2
train  = cat-dog-train.txt
valid  = cat-dog-test.txt
names  = cat-dog-obj.names
backup = backup/
```

The backup is where you want to store the yolo weights file.

The `cat-dog-obj.names` looks like this, plain and simple. Every new category should be on a new line, its line number should match the category number in the .txt label files we created earlier.

```
cat
dog
```

Now we go to create the .cfg for choose the yolo architecture. If you have less

Enter the following command into your terminal and watch your GPU do

Enter the following command into your terminal and watch your FPS go what it does best (copy your train.txt and test.txt to yolo\_darknet root folder):

```
manivannan@manivannan-whirlidatascience:~/YoloExample/darknet-v3$ ./darknet_detector train cfg/cat-dog-obj.data cfg/cat-dog-yolov3.cfg darknet53.conv.74
```

### Important Notes while training:

Weights only save every 100 iterations until 900, then saves every 10,000. If you want change the process please follow the [link](#).