



Yolo Darknet Installation

Step 1.

Git clone <https://github.com/AlexeyAB/darknet> or download zip then unzip it

Step 2.

**Welcome to the Grammarly beta
for Google Docs!**

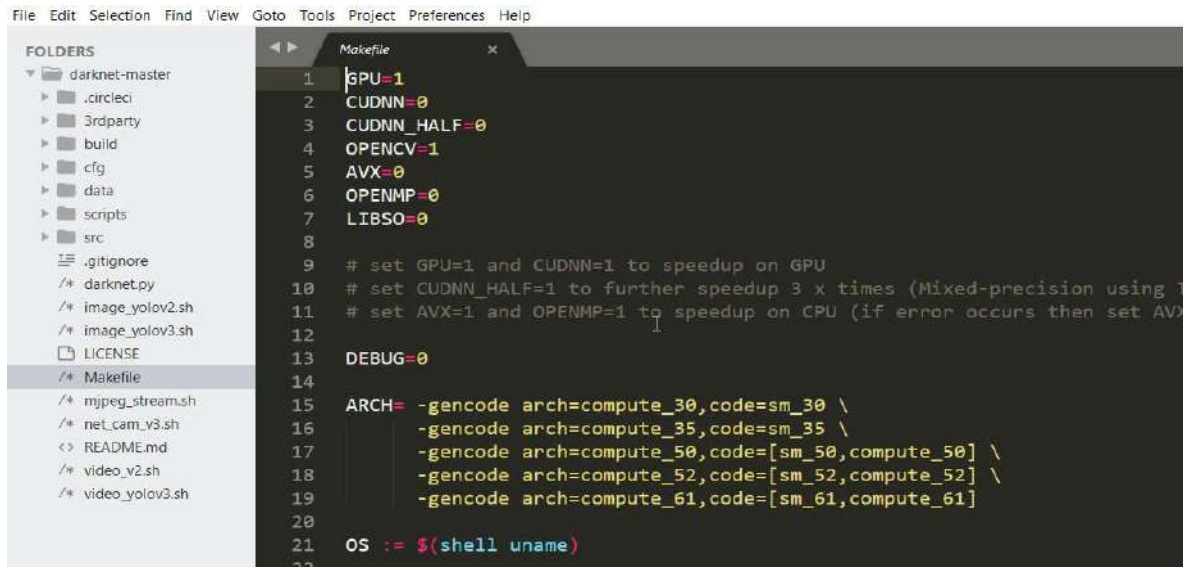
Sign up to turn on Grammarly
suggestions in your Google Docs.

Sign up

Already have an account? [Log in](#)



Open the Makefile in the unzipped folder **darknet-master**



In the Makefile change GPU = 1 and OPENCV = 1

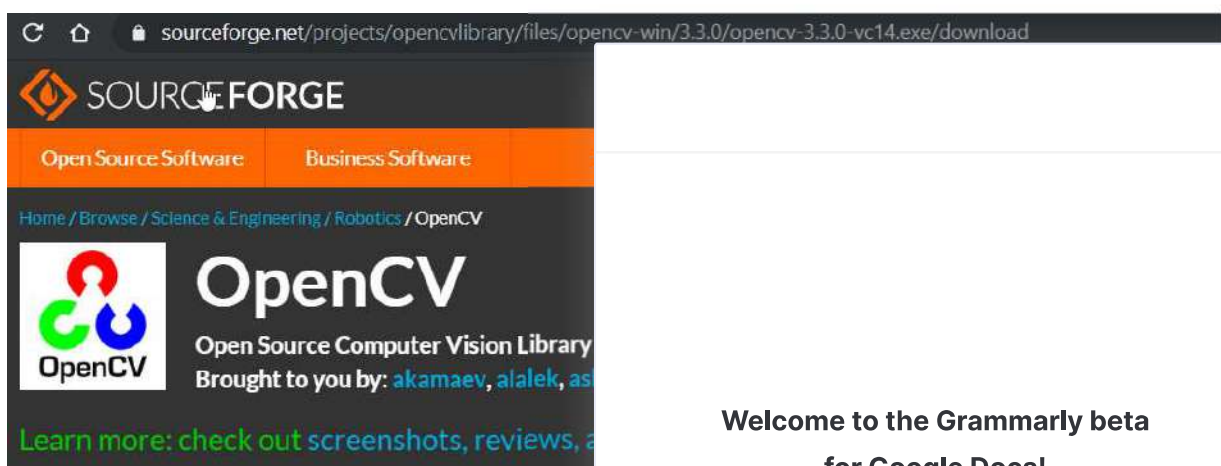
Step3.

Save the Makefile.

Step 4:-

Go to

<https://sourceforge.net/projects/opencvlibrary/files/opencv-win/3.3.0/opencv-3.3.0-vc14.exe/download>
and Download OpenCV



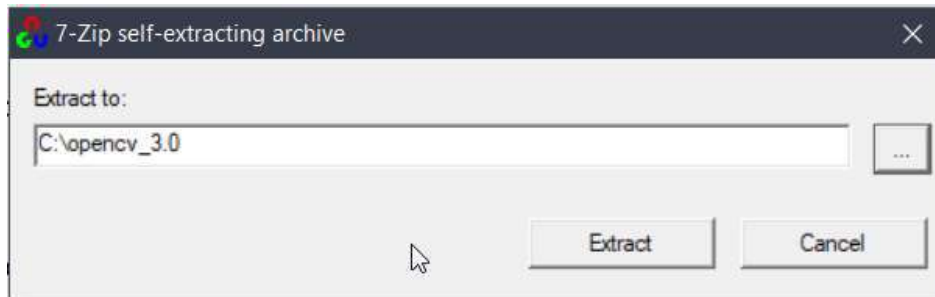
Create a new folder named **opencv_3.0** in C drive

Welcome to the Grammarly beta
for Google Docs!

Sign up to turn on Grammarly
suggestions in your Google Docs.

Sign up

Already have an account? [Log in](#)



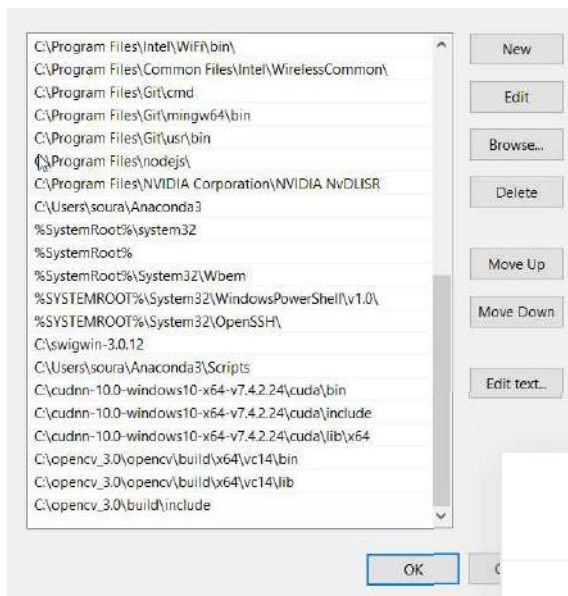
After Successful installation

Add to path in environment variables

C:\opencv_3.0\opencv\build\include

C:\opencv_3.0\opencv\build\x64\vc14\lib

C:\opencv_3.0\opencv\build\x64\vc14\bin



Step 5:-

Download and Install Microsoft Visual Studio

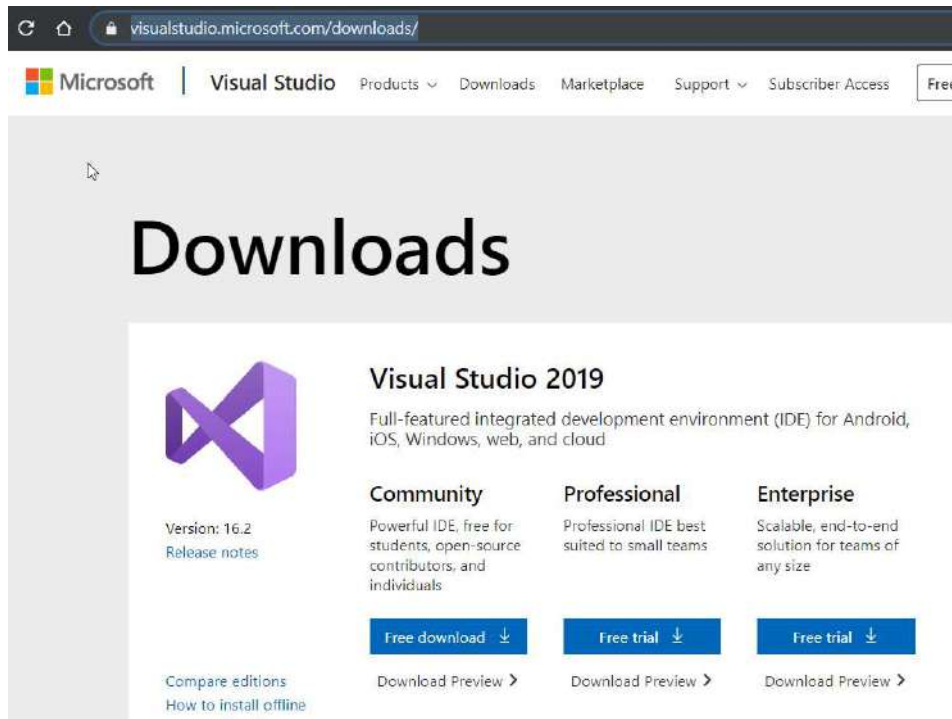
<https://visualstudio.microsoft.com/download>

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

Sign up

Already have an account? [Log in](#)



Step 6 :-

Open Visual Studio Community Edition 2017

Open New Project

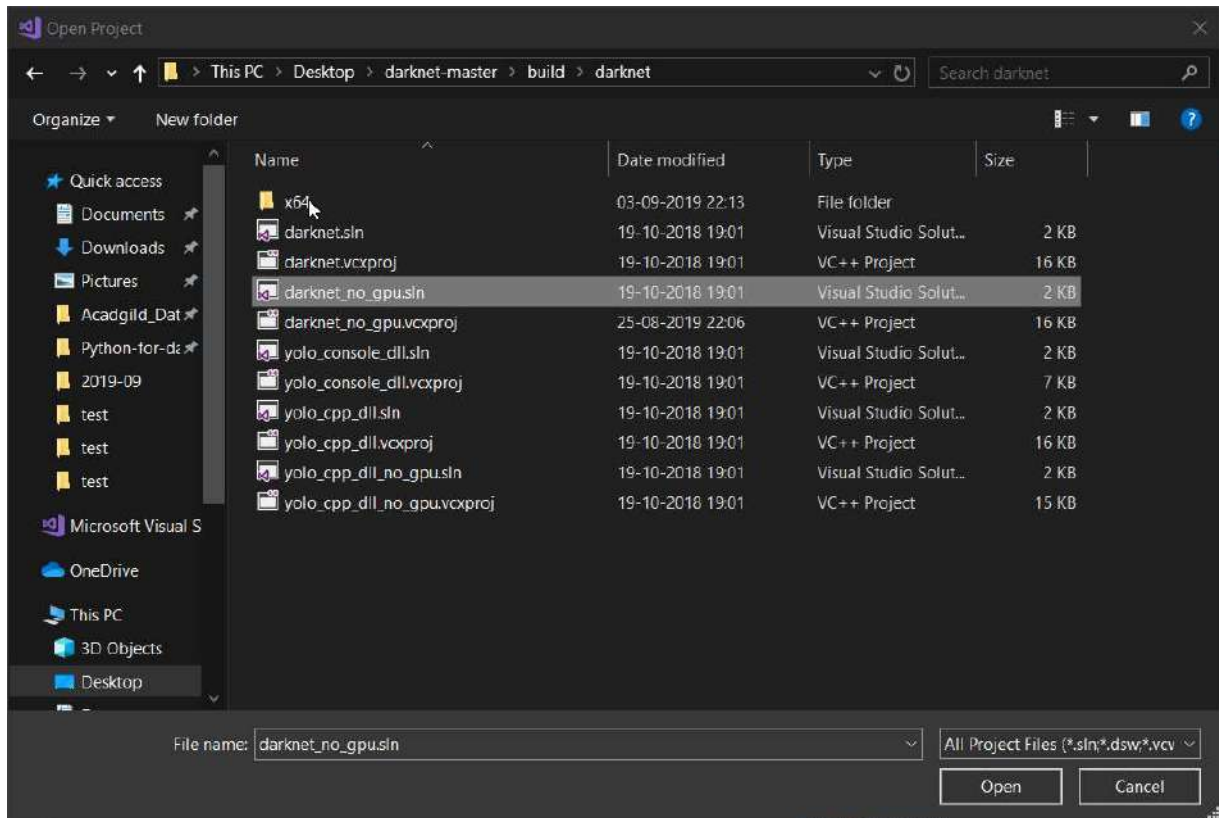
C:\Users\soura\Desktop\darknet-master\build\darknet_no_gpu.sln

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

[Sign up](#)

Already have an account? [Log in](#)



Step 7:-

Select OK for retargeted objects if POP UP

In the Solutions Explorer collapse the menu

iNeuron.ai

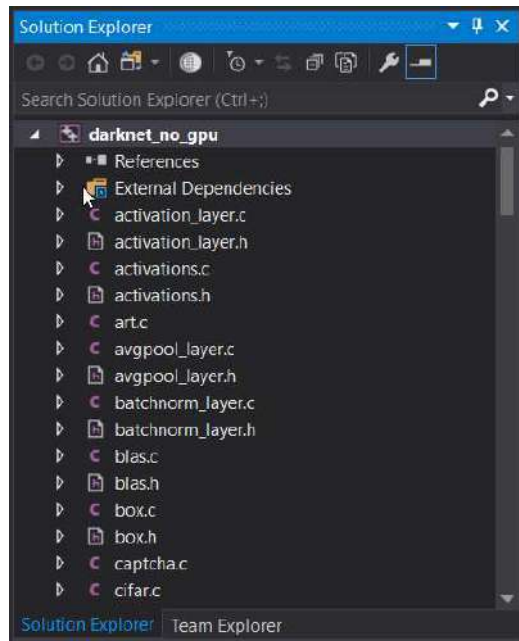
Pa

Welcome to the Grammarly beta for Google Docs!

Sign up to turn on Grammarly
suggestions in your Google Docs.

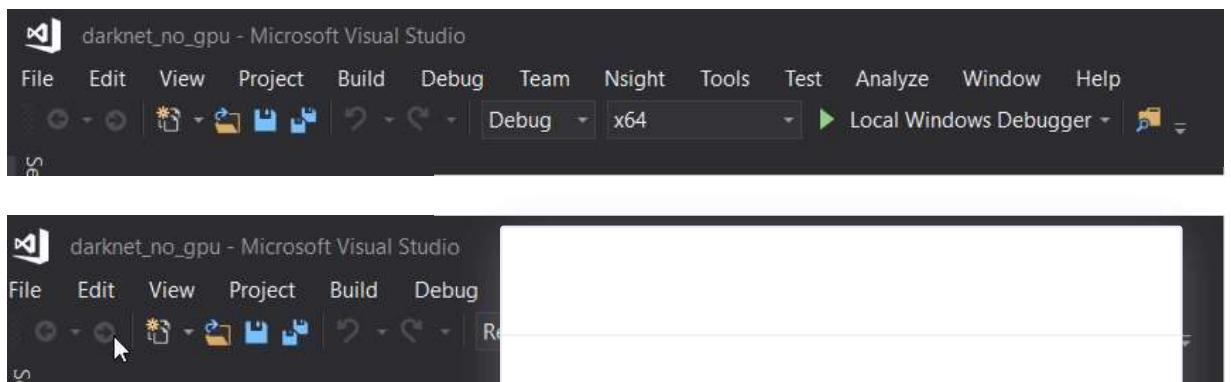
[Sign up](#)

Already have an account? [Log in](#)



Step 8:-

Change the **Debug** option to **Release**



Step 9 :

Then go to **Build** Options and select **Build**

iNeuron.ai

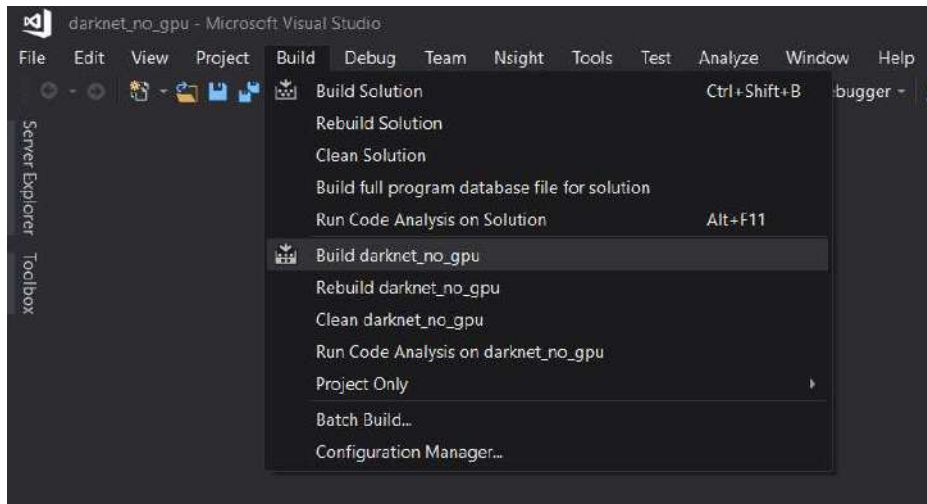
Pa

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

[Sign up](#)

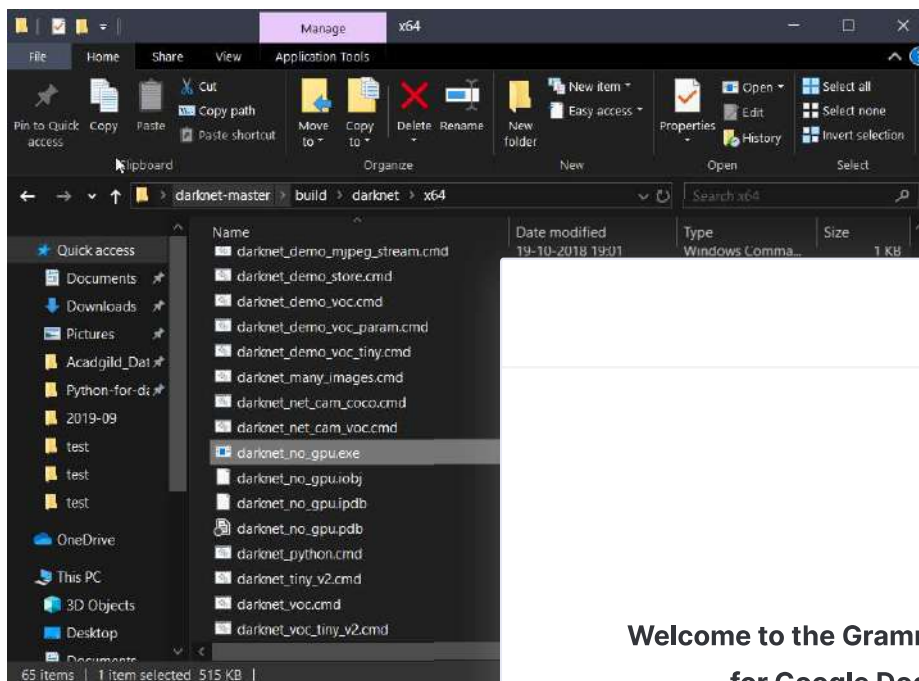
Already have an account? [Log in](#)



at end it will show **Build 1 succeeded.**

Step 11:-

Now go to C:\Users\soura\Desktop\darknet-master\build\darknet\x64 and you will see darknet_no_gpu.exe



Step 12:-

iNeuron.ai

Pa

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

[Sign up](#)

Already have an account? [Log in](#)



Download the yolo weights from <https://pjreddie.com/media/files/yolov3.weights> and paste in x64 folder

Step 13:-

Open CMD here or in the x64 folder

Step 14:-

For Images

Run YOLOv3 to **detect images**:

```
darknet_no_gpu.exe detect cfg/yolov3.cfg yolov3.weights data/horses.jpg
```

For Videos

Run YOLOv3 to **process videos**:

```
darknet_no_gpu.exe detector demo data/coco.data cfg/yolov3.cfg yolov3.weights testvideo.mp4  
-out_filename result.avi
```

```
C:\Windows\System32\cmd.exe  
Microsoft Windows [Version 10.0.18362.295]  
(c) 2019 Microsoft Corporation. All rights reserved.  
C:\Users\soura\Desktop\darknet-master\build\darknet\x64>darknet_no_gpu.exe  
detect cfg/yolov3.cfg yolov3.weights data/horses.jpg
```

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

Sign up

Already have an account? [Log in](#)



Collect images and label

Step 1:

Collect images using a **scraper**



Or use

<https://github.com/jiwitesh/ImageScraper/blob/master/ImageScraper/imagescrapperservice/test2.py>

Step 2:

Collect all the images in a single folder

Step 3:

Do the labelling using Labellmg or your preferred tool

Open Labellmg and change the format from default to yolo

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

Sign up

Already have an account? [Log in](#)



Open the directory where all images are present and start labelling

All the annotations will be saved in the .txt format with the label id as generated in classes.txt

Step 3:

Create a new folder named labels

Step 4:

Inside that create another new folder called

all corresponding .txt in a separate folder c

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

Sign up

Already have an account? [Log in](#)

iNeuron.ai

Pa



Step 5:

use the below script and save it in a new file called train.py

run the train.py file and

train.txt will be generated.....***Keep this file for future use***

```
import os
path='data/obj/'

imgList=os.listdir('images')
print(imgList)
textFile=open('train.txt','w')

for img in imgList:
    imgPath=path+ img +'\n'
    textFile.write(imgPath)
```

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

Sign up

Already have an account? [Log in](#)



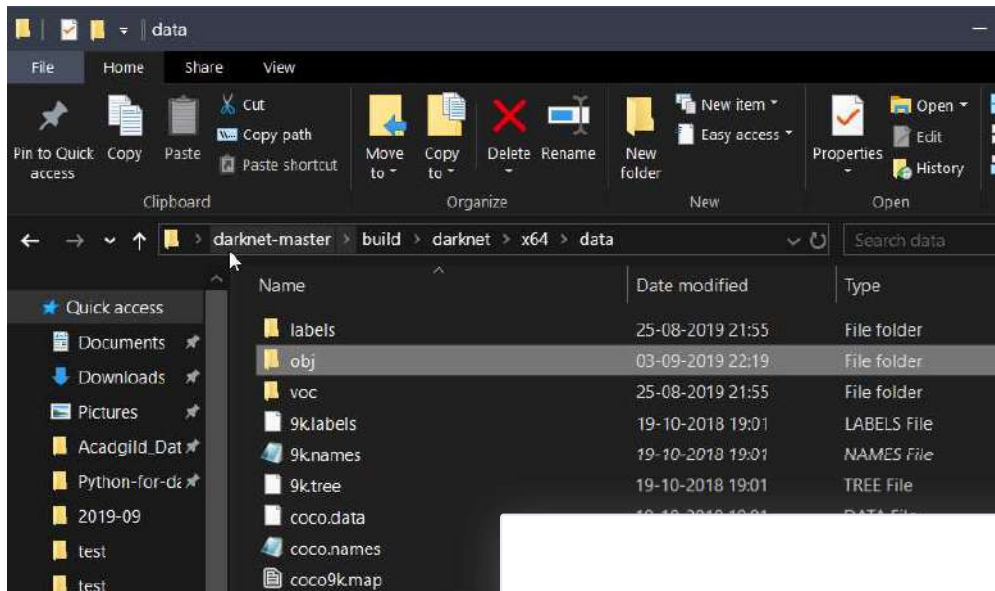
Custom Training of Yolo(Tiny)

Step 1:

Inside the X64 folder...There will be data folder.

Go to the data folder

Create a new folder named obj



Step 2:

train.txt was created in **labelling step**....so

Step 3:

Move all images from **images** folder and p

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

Sign up

Already have an account? [Log in](#)



Move all .txt files from **txt** folder and paste in **obj** folder

Step 4:

Create a new file called obj.names and enter all the class names acc to urs

A screenshot of a Notepad window titled "obj.names - Notepad". The window has a menu bar with "File", "Edit", "Format", "View", and "Help". The text content lists 16 class names, one per line: dog, person, cat, tv, car, meatballs, marinara sauce, tomato soup, chicken noodle soup, french onion soup, chicken breast, ribs, pulled pork, hamburger, cavity, and closehand.

```
obj.names - Notepad
File Edit Format View Help
dog
person
cat
tv
car
meatballs
marinara sauce
tomato soup
chicken noodle soup
french onion soup
chicken breast
ribs
pulled pork
hamburger
cavity
closehand
```

Step 5:

Create a new file called object.data

A screenshot of a code editor window titled "obj.data". The window shows a list of configuration parameters for training, numbered 1 through 6. The parameters are: classes=16, train=data/t, valid=data/t, names=data/o, backup=backu, and an empty line 6.

```
obj.data
1 classes=16
2 train=data/t
3 valid=data/t
4 names=data/o
5 backup=backu
6
```

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

[Sign up](#)

Already have an account? [Log in](#)



Step 6:

Go the cfg directory in x64 folder

and copy the file yolov3-tiny_obj.cfg to x64 folder.

Step 7:

Rename the file in x64 folder from yolov3-tiny_obj.cfg to yolov3-tiny-obj.cfg

Step 8 :

We will modify this cfg file now...Open in text editor and change the lines

1. **batch = 64**
2. **subdivisions = 8**

A screenshot of a text editor window titled "yolov3-tiny-obj.cfg". The editor shows a configuration file with the following content:

```
1 [net]
2 # Testing
3 #batch=64
4 #subdivisions=8
5 # Training
6 batch=64
7 subdivisions=8
8 width=416
9 height=416
10 channels=3
11 momentum=0.
12 decay=0.000
```

The lines for "batch=64" and "subdivisions=8" are highlighted in a darker shade of gray.

3. change line **classes=80** to your number

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

[Sign up](#)

Already have an account? [Log in](#)



```

132 [yolo]
133 mask = 3,4,5
134 anchors = 44.8535,69.1
135 classes=16
136 num=6
137 jitter=.3
138 ignore_thresh = .7
139 truth_thresh = 1
140 random=1
141

```

4. change **[filters=255]** to **filters=(classes + 5)x3** in the 3 [convolutional] before each [yolo] layer

```

123 [convolutional]
124 size=1
125 stride=1
126 pad=1
127 filters=63
128 activation=linear
129

```

Step 9:

Run this command to get the anchors value
darknet_no_gpu.exe detector calc_anchors data/obj

```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.18362.295]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\soura\Desktop\darknet-master\build>darknet_no_gpu.exe detector calc_anchors data/obj.data -num_of_clusters 16

```

Copy the result of anchors value from console and

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

[Sign up](#)

Already have an account? [Log in](#)



```
[yolo]
mask = 3,4,5
anchors = 44.8535,69.1571, 119.8065,158.4406, 180.8063,245.7598, 249.9994,184.7015, 228.8867,344.1920, 331.1044,309.4365
classes=16
```

******Note: if error Out of memory occurs then in .cfg-file you should increase subdivisions=16, 32 or 64******

Step 10:

Download default weights file for yolov3-tiny:

<https://pireddie.com/media/files/yolov3-tiny.weights>

Step 11:

Get pre-trained weights yolov3-tiny.conv.15 using command:

darknet_no_gpu.exe partial cfg/yolov3-tiny.cfg yolov3-tiny.weights yolov3-tiny.conv.15 15

Step 12:

Start training:

darknet_no_gpu.exe detector train data/obj.data
-dont_show

iNeuron.ai

Pa

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

Sign up

Already have an account? [Log in](#)

A screenshot of a Windows command prompt window. The title bar reads "C:\Windows\System32\cmd.exe". The text inside the window is as follows:

```
Microsoft Windows [Version 10.0.18362.295]  
(c) 2019 Microsoft Corporation. All rights reserved.  
C:\Users\soura\Desktop\darknet-master\build\darknet\x64>darknet_no_gpu.exe detector  
train data/obj.data yolov3-tiny-obj.cfg yolov3-tiny.conv.15 -dont_show
```

**Welcome to the Grammarly beta
for Google Docs!**

Sign up to turn on Grammarly
suggestions in your Google Docs.

Sign up

Already have an account? [Log in](#)