

```
[net]
# Testing
# batch=1
# subdivisions=1
# Training
batch=64
subdivisions=16
width=608
height=608
channels=3
momentum=0.9
decay=0.0005
angle=0
saturation = 1.5
exposure = 1.5
hue=.1

learning_rate=0.01
burn_in=1000
max_batches = 500200
policy=steps
steps=400000,450000
scales=.1,.1
```

```
[convolutional]
batch_normalize=1
filters=32
size=3
stride=1
pad=1
activation=leaky
```

```
# Downsample
```

```
[convolutional]
batch_normalize=1
filters=64
size=3
stride=2
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=32
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=64
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
# Downsample
```

```
[convolutional]
batch_normalize=1
filters=128
size=3
stride=2
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=64
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=128
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=64
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=128
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
# Downsample
```

```
[convolutional]
batch_normalize=1
```

```
filters=256
size=3
stride=2
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=128
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=256
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=128
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=256
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=128
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=256
size=3
```

```
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=128
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=256
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=128
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=256
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=128
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
```

```
batch_normalize=1
filters=256
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=128
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=256
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=128
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=256
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
# Downsample
```

```
[convolutional]
batch_normalize=1
filters=512
size=3
stride=2
```

```
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=512
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=512
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=512
size=3
stride=1
```

```
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=512
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=512
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
```

```
batch_normalize=1
filters=512
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=512
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=512
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
# Downsample
```

```
[convolutional]
batch_normalize=1
filters=1024
size=3
```



```
stride=2
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=512
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=1024
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=512
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=1024
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=512
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=1024
size=3
stride=1
pad=1
```

```
activation=leaky

[shortcut]
from=-3
activation=linear
```

```
[convolutional]
batch_normalize=1
filters=512
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=1024
size=3
stride=1
pad=1
activation=leaky
```

```
[shortcut]
from=-3
activation=linear
```

```
#####
```

```
[convolutional]
batch_normalize=1
filters=512
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
size=3
stride=1
pad=1
filters=1024
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=512
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
size=3
stride=1
pad=1
filters=1024
```

activation=leaky

[convolutional]  
batch\_normalize=1  
filters=512  
size=1  
stride=1  
pad=1  
activation=leaky

[convolutional]  
batch\_normalize=1  
size=3  
stride=1  
pad=1  
filters=1024  
activation=leaky

[convolutional]  
size=1  
stride=1  
pad=1  
filters=255  
activation=linear

[yolo]  
mask = 6,7,8  
anchors = 10,13, 16,30, 33,23, 30,61, 62,45, 59,119, 116,90,  
156,198, 373,326  
classes=80  
num=9  
jitter=.3  
ignore\_thresh = .7  
truth\_thresh = 1  
random=1

[route]  
layers = -4

[convolutional]  
batch\_normalize=1  
filters=256  
size=1  
stride=1  
pad=1  
activation=leaky

[upsample]  
stride=2

[route]  
layers = -1, 61

[convolutional]

```
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
size=3
stride=1
pad=1
filters=512
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
size=3
stride=1
pad=1
filters=512
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
size=3
stride=1
pad=1
filters=512
activation=leaky
```

```
[convolutional]
size=1
stride=1
pad=1
filters=255
activation=linear
```

```
[yolo]
mask = 3,4,5
```

```
anchors = 10,13, 16,30, 33,23, 30,61, 62,45, 59,119, 116,90,  
156,198, 373,326  
classes=80  
num=9  
jitter=.3  
ignore_thresh = .7  
truth_thresh = 1  
random=1
```

```
[route]  
layers = -4
```

```
[convolutional]  
batch_normalize=1  
filters=128  
size=1  
stride=1  
pad=1  
activation=leaky
```

```
[upsample]  
stride=2
```

```
[route]  
layers = -1, 36
```

```
[convolutional]  
batch_normalize=1  
filters=128  
size=1  
stride=1  
pad=1  
activation=leaky
```

```
[convolutional]  
batch_normalize=1  
size=3  
stride=1  
pad=1  
filters=256  
activation=leaky
```

```
[convolutional]  
batch_normalize=1  
filters=128  
size=1  
stride=1  
pad=1  
activation=leaky
```

```
[convolutional]  
batch_normalize=1  
size=3  
stride=1
```

```
pad=1
filters=256
activation=leaky
```

```
[convolutional]
batch_normalize=1
filters=128
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1
size=3
stride=1
pad=1
filters=256
activation=leaky
```

```
[convolutional]
size=1
stride=1
pad=1
filters=255
activation=linear
```

```
[yolo]
mask = 0,1,2
anchors = 10,13, 16,30, 33,23, 30,61, 62,45, 59,119, 116,90,
156,198, 373,326
classes=80
num=9
jitter=.3
ignore_thresh = .7
truth_thresh = 1
random=1
```