

This repository | Search

Pull requests Issues Marketplace Gist

**bityangke / Speed-accuracy-trade-offs-for-modern-convolutional-object-detectors**forked from [rayanelleuch/Speed-accuracy-trade-offs-for-modern-convolutional-object-detectors](#)

Watch ▾

1

★ Star

0

Fork

1

&lt;&gt; Code

Pull requests 0

Projects 0

Wiki

Insights ▾

The purpose of this project is to reimplement the following paper: <https://arxiv.org/pdf/1611.10012.pdf>

2 commits

1 branch

0 releases

1 contributor

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download

This branch is even with rayanelleuch:master.

Pull request Compare

**rayanelleuch** add some models from tfSlim

Latest commit f545a02 on 22 Mar

<a href="#">nets</a>	add some models from tfSlim	2 months ago
<a href="#">LICENSE</a>	Initial commit	2 months ago
<a href="#">README.md</a>	add some models from tfSlim	2 months ago

[README.md](#)

# Speed-accuracy-trade-offs-for-modern-convolutional-object-detectors

The purpose of this project is to reimplement the following paper: <https://arxiv.org/pdf/1611.10012.pdf>

## Pre-trained Models

Neural nets work best when they have many parameters, making them powerful function approximators. However, this means they must be trained on very large datasets. Because training models from scratch can be a very computationally intensive process requiring days or even weeks, we provide various pre-trained models, as listed below. These CNNs have been trained on the [ILSVRC-2012-CLS](#) image classification dataset.

In the table below, we list each model, the corresponding TensorFlow model file, the link to the model checkpoint, and the top 1 and top 5 accuracy (on the imagenet test set). Note that the VGG and ResNet parameters have been converted from their original caffe formats ([here](#) and [here](#)), whereas the Inception parameters have been trained internally at Google. Also be aware that these accuracies were computed by evaluating using a single image crop. Some academic papers report higher accuracy by using multiple crops at multiple scales.

Model	TF-Slim File	Checkpoint	Top-1 Accuracy	Top-5 Accuracy
<a href="#">Inception V1</a>	<a href="#">Code</a>	<a href="#">inception_v1_2016_08_28.tar.gz</a>	69.8	89.6
<a href="#">Inception V2</a>	<a href="#">Code</a>	<a href="#">inception_v2_2016_08_28.tar.gz</a>	73.9	91.8
<a href="#">Inception V3</a>	<a href="#">Code</a>	<a href="#">inception_v3_2016_08_28.tar.gz</a>	78.0	93.9
<a href="#">Inception V4</a>	<a href="#">Code</a>	<a href="#">inception_v4_2016_09_09.tar.gz</a>	80.2	95.2
<a href="#">Inception-ResNet-v2</a>	<a href="#">Code</a>	<a href="#">inception_resnet_v2.tar.gz</a>	80.4	95.3
<a href="#">ResNet 50</a>	<a href="#">Code</a>	<a href="#">resnet_v1_50.tar.gz</a>	75.2	92.2
<a href="#">ResNet 101</a>	<a href="#">Code</a>	<a href="#">resnet_v1_101.tar.gz</a>	76.4	92.9

Model	TF-Slim File	Checkpoint	Top-1 Accuracy	Top-5 Accuracy
<a href="#">ResNet 152</a>	<a href="#">Code</a>	<a href="#">resnet_v1_152.tar.gz</a>	76.8	93.2
<a href="#">VGG 16</a>	<a href="#">Code</a>	<a href="#">vgg_16.tar.gz</a>	71.5	89.8
<a href="#">VGG 19</a>	<a href="#">Code</a>	<a href="#">vgg_19.tar.gz</a>	71.1	89.8

Here is an example of how to download the Inception V3 checkpoint:

```
$ CHECKPOINT_DIR=/tmp/checkpoints
$ mkdir ${CHECKPOINT_DIR}
$ wget http://download.tensorflow.org/models/inception_v3_2016_08_28.tar.gz
$ tar -xvf inception_v3_2016_08_28.tar.gz
$ mv inception_v3.ckpt ${CHECKPOINT_DIR}
$ rm inception_v3_2016_08_28.tar.gz
```

