datitran / object_detector_app

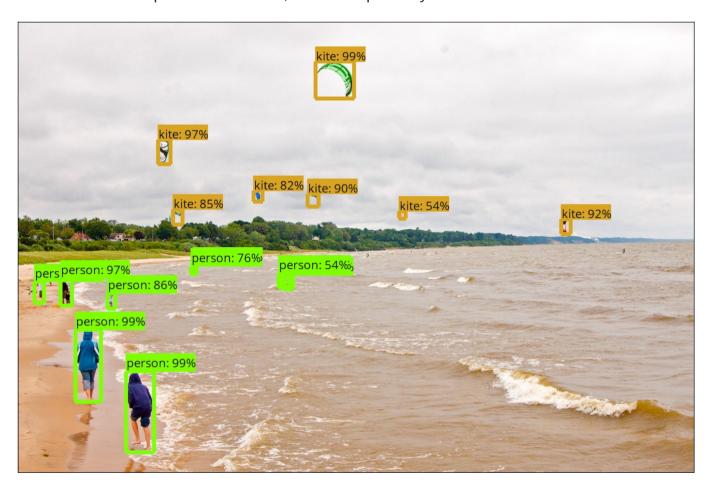
Branch: master ▼ object_detector_app /	object_detection /	Create new file	Upload files	Find file	History	
datitran Remove process and terminate pool explicitly and also remove unnecess			Latest commit 13adb69 on 17 Ju			
anchor_generators	Add working version without opency.	6 months ago				
box_coders	Add working version without opency.	6 months ago				
b uilders	Add working version without opency.			6 mo	nths ago	
core	Add working version without opency.			6 mo	nths ago	
data data	Add working version without opency.			6 mo	nths ago	
data_decoders	Add working version without opency.			6 mo	nths ago	
■ g3doc	Add working version without opency.			6 mo	nths ago	
matchers	Add working version without opency.			6 mo	nths ag	
meta_architectures	Add working version without opency.			6 mo	nths ag	
models	Add working version without opency.			6 mo	nths ago	
protos	Add working version without opency.			6 mo	nths ago	
a samples	Add working version without opency.			6 mo	nths ago	
ssd_mobilenet_v1_coco_11_06_2017	Add working version without opency.			6 mo	nths ago	
test_images	Add working version without opency.			6 mo	nths ago	
utils	Add working version without opency.			6 mo	nths ag	

BUILD	Add working version without opency.	6 months ago
CONTRIBUTING.md	Add working version without opency.	6 months ago
■ README.md	Add working version without opency.	6 months ago
initpy	Add working version without opencv.	6 months ago
create_pascal_tf_record.py	Add working version without opency.	6 months ago
create_pascal_tf_record_test.py	Add working version without opency.	6 months ago
create_pet_tf_record.py	Add working version without opency.	6 months ago
eval.py	Add working version without opency.	6 months ago
eval_util.py	Add working version without opency.	6 months ago
evaluator.py	Add working version without opency.	6 months ago
export_inference_graph.py	Add working version without opency.	6 months ago
exporter.py	Add working version without opency.	6 months ago
exporter_test.py	Add working version without opency.	6 months ago
bject_detection_tutorial.ipynb	Add working version without opency.	6 months ago
train.py	Add working version without opency.	6 months ago
trainer.py	Add working version without opency.	6 months ago
trainer_test.py	Add working version without opency.	6 months ago

README.md

Tensorflow Object Detection API

Creating accurate machine learning models capable of localizing and identifying multiple objects in a single image remains a core challenge in computer vision. The TensorFlow Object Detection API is an open source framework built on top of TensorFlow that makes it easy to construct, train and deploy object detection models. At Google we've certainly found this codebase to be useful for our computer vision needs, and we hope that you will as well.



Contributions to the codebase are welcome and we would love to hear back from you if you find this API useful. Finally if you use the Tensorflow Object Detection API for a research publication, please consider citing:

"Speed/accuracy trade-offs for modern convolutional object detectors."
Huang J, Rathod V, Sun C, Zhu M, Korattikara A, Fathi A, Fischer I, Wojna Z,
Song Y, Guadarrama S, Murphy K, CVPR 2017

[link][bibtex]

Maintainers

• Jonathan Huang, github: jch1

• Vivek Rathod, github: tombstone

• Derek Chow, github: derekjchow

• Chen Sun, github: jesu9

• Menglong Zhu, github: dreamdragon

Table of contents

Quick Start:

- Quick Start: Jupyter notebook for off-the-shelf inference
- · Quick Start: Training a pet detector

Setup:

- Installation
- Configuring an object detection pipeline
- Preparing inputs

Running:

- Running locally
- Running on the cloud

Extras:

- Tensorflow detection model zoo
- Exporting a trained model for inference
- Defining your own model architecture

Release information

June 15, 2017

In addition to our base Tensorflow detection model definitions, this release includes:

- A selection of trainable detection models, including:
 - o Single Shot Multibox Detector (SSD) with MobileNet,
 - SSD with Inception V2,
 - Region-Based Fully Convolutional Networks (R-FCN) with Resnet 101,
 - Faster RCNN with Resnet 101,
 - Faster RCNN with Inception Resnet v2
- Frozen weights (trained on the COCO dataset) for each of the above models to be used for out-of-the-box inference purposes.
- A Jupyter notebook for performing out-of-the-box inference with one of our released models
- Convenient local training scripts as well as distributed training and evaluation pipelines via Google Cloud.

Thanks to contributors: Jonathan Huang, Vivek Rathod, Derek Chow, Chen Sun, Menglong Zhu, Matthew Tang, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, Jasper Uijlings, Viacheslav Kovalevskyi, Kevin Murphy