TensorFlow Object Detection API

Akin Osman Kazakci
Maitre-Assistant
IHEIE, MINES ParisTech
akin.kazakci@mines-paristech.fr

This primer will shortly review some notions and resources that you will need to get started with the TensorFlow Object Detection (TFOD) API.

Introduction to TensorFlow

TensorFlow makes it easy for beginners and experts to create machine learning models for desktop, mobile, web, and cloud. See the sections below to get started.

TensorFlow

Learn the foundation of TensorFlow with tutorials for beginners and experts to help you create your next machine learning project.

Learn more

For JavaScript

Use TensorFlow.js to create new machine learning models and deploy existing models with JavaScript.

Learn more

For Mobile & IoT

Run inference with TensorFlow Lite on mobile and embedded devices like Android, iOS, Edge TPU, and Raspberry Pi.

Learn more

For Production

Deploy a production-ready ML pipeline for training and inference using TensorFlow Extended (TFX).

Learn more

README.md

TensorFlow Model Garden

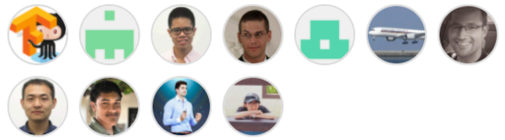
Welcome to the Model Garden for TensorFlow

The TensorFlow Model Garden is a repository with a number of different implementations of state-of-the-art (SOTA) models and modeling solutions for TensorFlow users. We aim to demonstrate the best practices for modeling so that TensorFlow users can take full advantage of TensorFlow for their research and product development.

To improve the transparency and reproducibility of our models, training logs on TensorBoard.dev are also provided for models to the extent possible though not all models are suitable.

Directory	Description
official	• A collection of example implementations for SOTA models using the latest TensorFlow 2's high-level APIs • Officially maintained, supported, and kept up to date with the latest TensorFlow 2 APIs by TensorFlow • Reasonably optimized for fast performance while still being easy to read
research	• A collection of research model implementations in TensorFlow 1 or 2 by researchers • Maintained and supported by researchers
community	• A curated list of the GitHub repositories with machine learning models and implementations powered by TensorFlow 2
orbit	• A flexible and lightweight library that users can easily use or fork when writing customized training loop code in TensorFlow 2.x. It seamlessly integrates with `tf.distribute` and supports running on different device types (CPU, GPU, and TPU).

Contributors 737



+ 726 contributors

Languages

- Python 90.2%
- Jupyter Notebook 7.0%
- C++ 1.8%
- Other 1.0%

TensorFlow 2 Detection Model Zoo

We provide a collection of detection models pre-trained on the COCO 2017 dataset. These models can be useful for out-of-the-box inference if you are interested in categories already in those datasets. You can try it in our inference colab

They are also useful for initializing your models when training on novel datasets. You can try this out on our few-shot training colab.

Please look at this guide for mobile inference.

Finally, if you would like to train these models from scratch, you can find the model configs in this directory (also in the linked `tar.gz`s).

Model name	Speed (ms)	COCO mAP	Outputs
CenterNet HourGlass104 512x512	70	41.9	Boxes
CenterNet HourGlass104 Keypoints 512x512	76	40.0/61.4	Boxes/Keypoints
CenterNet HourGlass104 1024x1024	197	44.5	Boxes
CenterNet HourGlass104 Keypoints 1024x1024	211	42.8/64.5	Boxes/Keypoints
CenterNet Resnet50 V1 FPN 512x512	27	31.2	Boxes
CenterNet Resnet50 V1 FPN Keypoints 512x512	30	29.3/50.7	Boxes/Keypoints
CenterNet Resnet101 V1 FPN 512x512	34	34.2	Boxes
CenterNet Resnet50 V2 512x512	27	29.5	Boxes
CenterNet Resnet50 V2 Keypoints 512x512	30	27.6/48.2	Boxes/Keypoints
CenterNet MobileNetV2 FPN 512x512	6	23.4	Boxes
CenterNet MobileNetV2 FPN Keypoints 512x512	6	41.7	Keypoints
EfficientDet D0 512x512	39	33.6	Boxes

TensorFlow 2 Object Detection API tutorial

Important

This tutorial is intended for TensorFlow 2.5, which (at the time of writing this tutorial) is the latest stable version of TensorFlow 2.x.

A version for TensorFlow 2.2 can be found here.

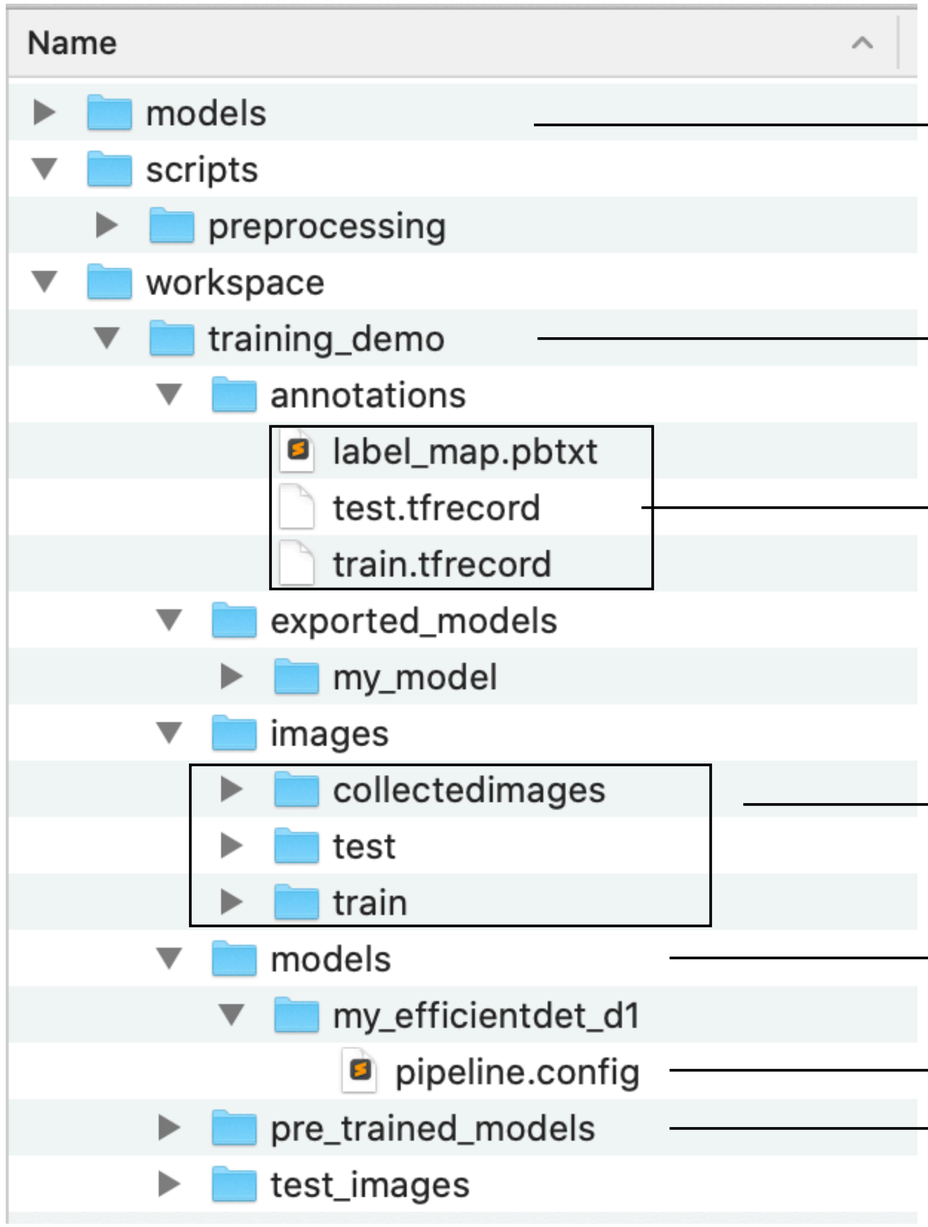
A version for TensorFlow 1.14 can be found here.

This is a step-by-step tutorial/guide to setting up and using TensorFlow's Object Detection API to perform, namely, object detection in images/video.

The software tools which we shall use throughout this tutorial are listed in the table below:

Target Software versions

OS	Windows, Linux
Python	3.9 [1]
TensorFlow	2.5.0
CUDA Toolkit	11.2
CuDNN	8.1.0
Anaconda	Python 3.8 (Optional)



TFOD API files

For each model you will train, you can start a folder under your workspace.

We will generate these files for each new detection task / dataset.

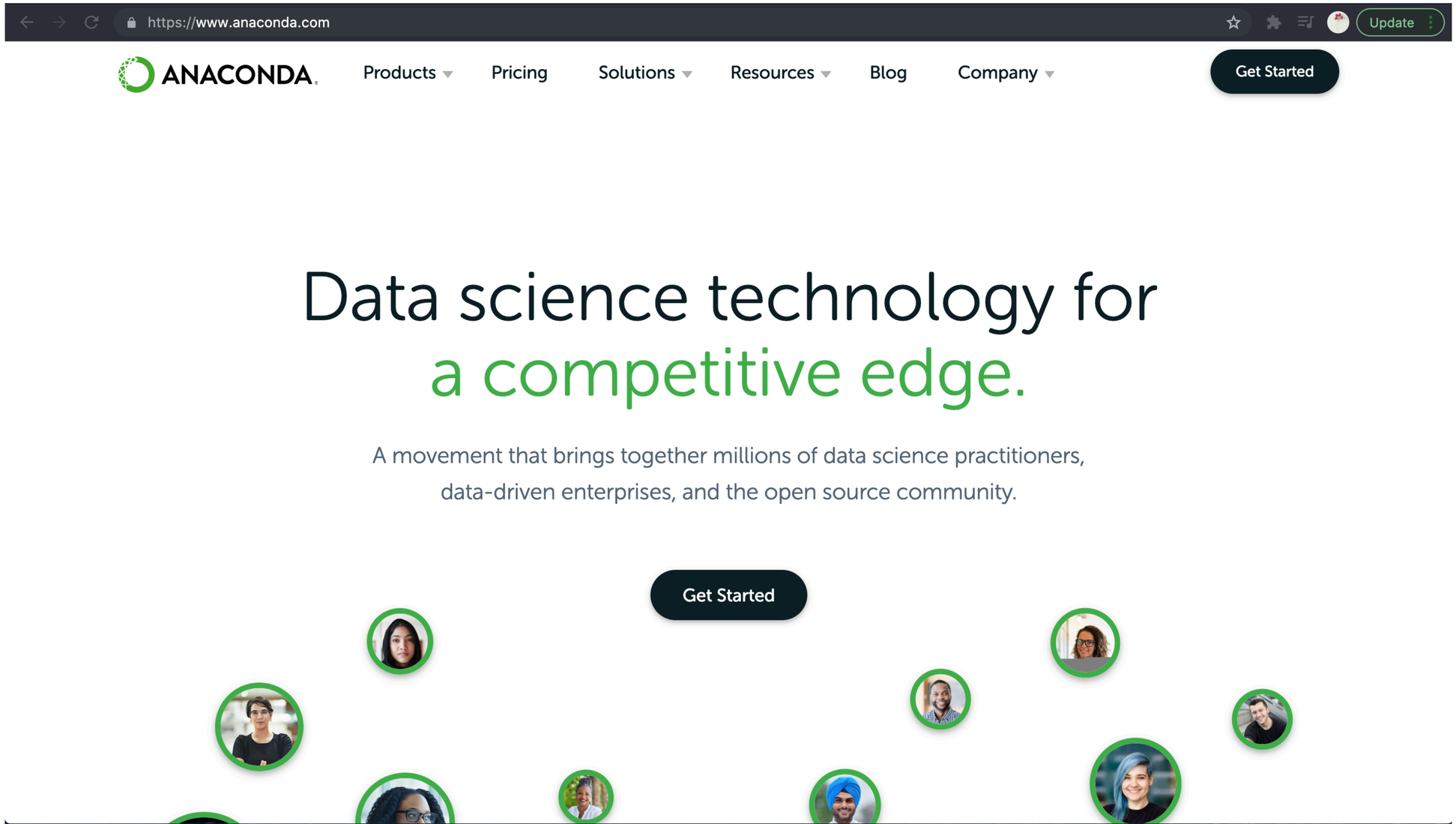
You can put images you'll collect and label under collected_images. You will need a script to split them into train and test folders.

Outputs folder for training.

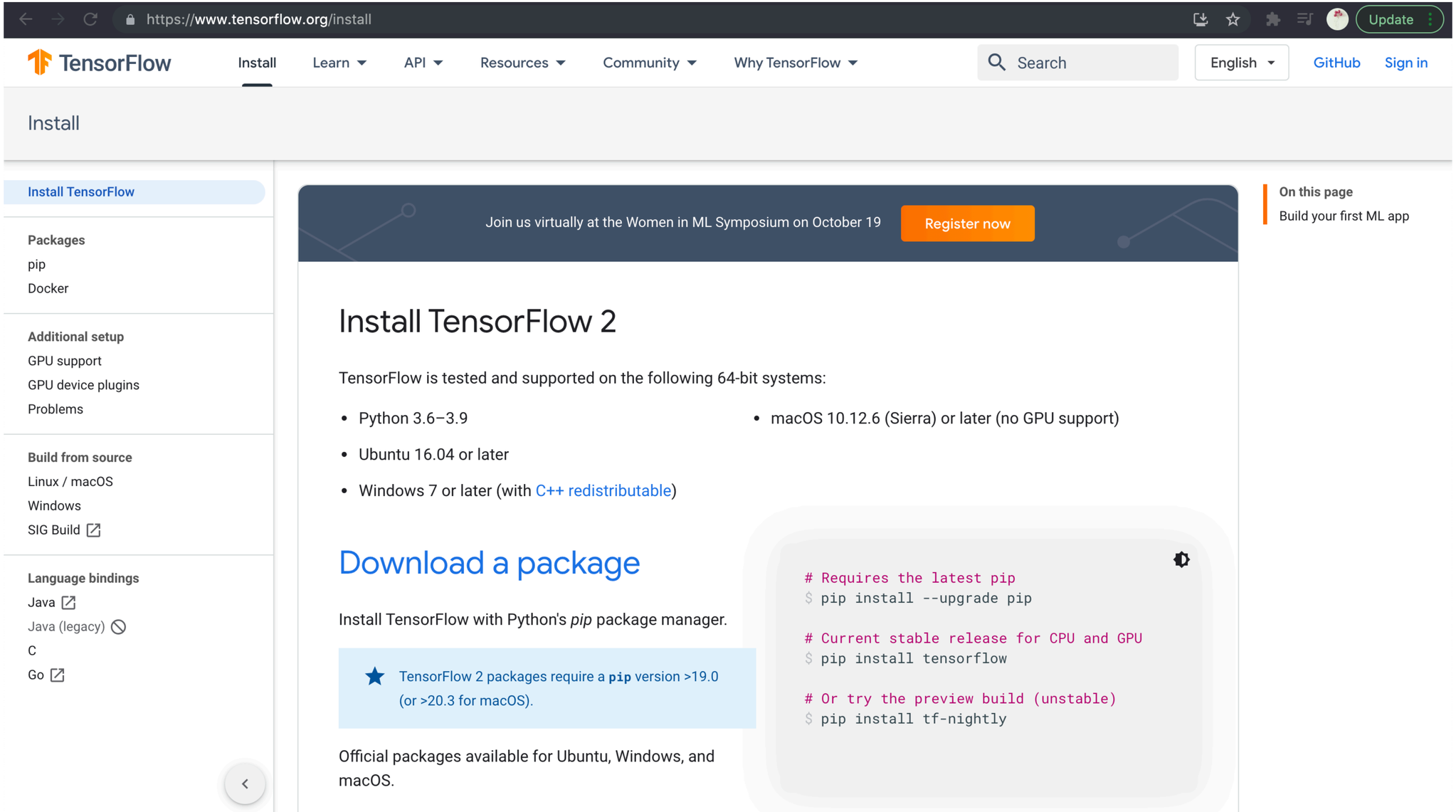
We will also need to provide a training configuration file.

Public models from which you can start training your own models.

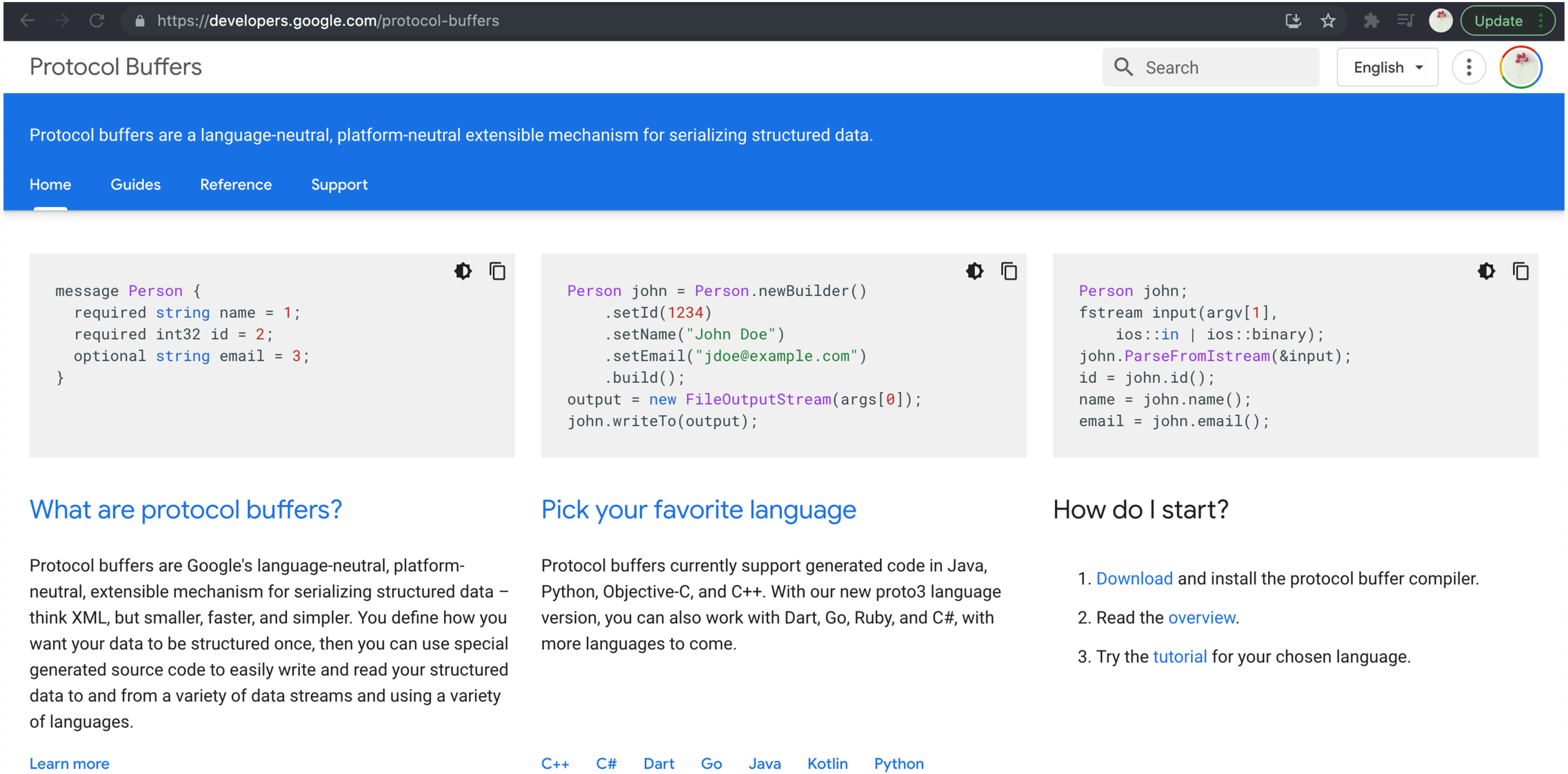
What to install



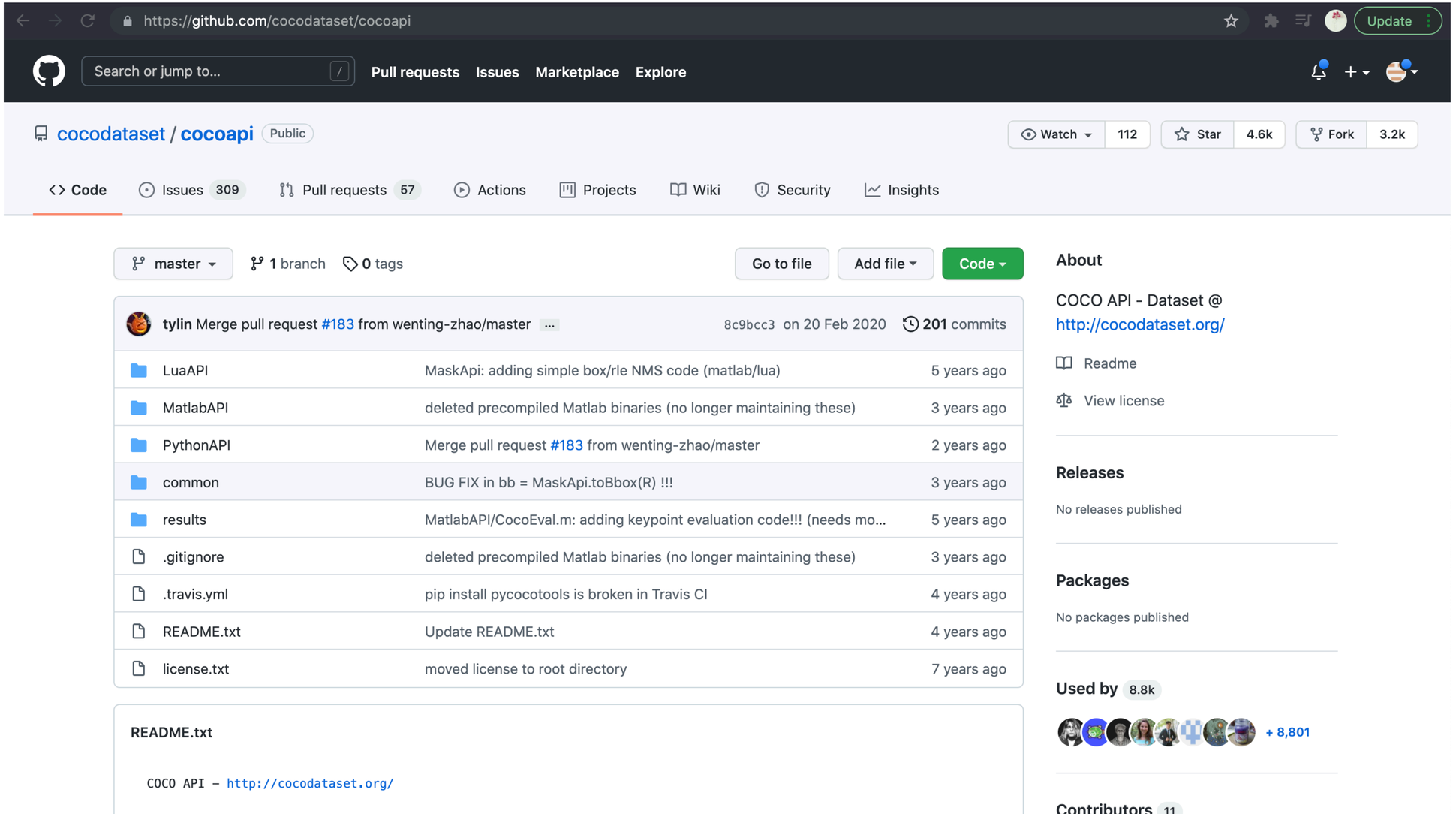
What to install



What to install



What to install



Things you will need to do

1. Organise your workspace (folder structure)
2. (Create a label map and annotate images)
3. Generate (Tensorflow record files) tfrecords
4. Provide a configuration file for the training pipeline
5. Train your model and check its progress
6. Export the resulting model and use it to detect objects