Guide for Converting COCO Formatted Datasets to KITTI Format

Author: Matthew Howlett
Document most recent update: September 24, 2019

1. Understanding the KITTI Format

Data Format Description
==========
Source/more info:
https://github.com/NVIDIA/DIGITS/blob/v4.0.0-rc.3/digits/extensions/data/objectDetection/README.md
==========

#Values	Name	Description
1	type	Describes the type of object: 'Car', 'Van', 'Truck', 'Pedestrian', 'Person_sitting', 'Cyclist', 'Tram', 'Misc' or 'DontCare'
1	truncated	Float from 0 (non-truncated) to 1 (truncated), where truncated refers to the object leaving image boundaries
1	occluded	Integer (0,1,2,3) indicating occlusion state: 0 = fully visible, 1 = partly occluded 2 = largely occluded, 3 = unknown
1	alpha	Observation angle of object, ranging [-pi..pi]
4	bbox	2D bounding box of object in the image (0-based index): contains left, top, right, bottom pixel coordinates
3	dimensions	3D object dimensions: height, width, length (in meters)
3	location	3D object location x,y,z in camera coordinates (in meters)
1	rotation_y	Rotation ry around Y-axis in camera coordinates [-pi..pi]
1	score	Only for results: Float, indicating confidence in detection, needed for p/r curves, higher is better.

Example KITTI label text file output:

stop_sign 0 0 0 803.5 77.5 911.5 189.5 0 0 0 0 0 0 0

2. Convert COCO to KITTI

The coco2kitti.py python conversion script is provided by dusty-nv. You can also find it in the python directory, and all the other files needed for this tutorial here: https://github.com/howl0893/custom-object-detection-datasets

Make appropriate changes to coco2kitti.py script in lines 52, 53, 54.
Also may want to change label file name in lines 32, 65, 68.

```
dataDir =
'C:/Users/Matt/Desktop/custom-object-detection-datasets/datasets/stop
_signs_by_hand'
dataType = 'road-signs'
annFile = '%s/annotations/coco_instances_%s.json' % (dataDir,
dataType)
```

After making the proper edits and saving the python script, run it by opening the command terminal. Navigate to the cocosynth-master directory within the command terminal and run the following command:

```
python ./datasets/road-signs/annotations/coco2kitti.py
```

Upon running this script, a new folder named labels should be within the cocosynth-master folder which contains a text file for each image/mask pair within your dataset.

3. KITTI dataset folder structure

Now that we have the label text files for each image, the last thing to do is to arrange the dataset with the correct folder structure. Here is how you should arrange your folder structure:

```
train/
├── images/
│   └── 000001.png
└── labels/
    └── 000001.txt
val/
├── images/
│   └── 000002.png
└── labels/
    └── 000002.txt
```