Title: README for Setup Mask RCNN

Document Number: 190l-1

CTI One Corporation

Table 1a. Document History

2022-05-2	Establish this document, document archive: /media/harry/easystore/backup-2020-2-15/CTI/3proejcts/3-8-smart-tech/3-8-4-CTI/3-8-4-6-products/AIV200/190-robots-health/190l-mask-rcnn	YY

Table 1b. Testing and Release Approval Form

2022-05-2	Test Status: and Release Status: approved for release by HL	Pending for testing and approval

Table 2. References

Number	Name and URL	Note
1.	Mask R-CNN for Object Detection and Segmentation https://github.com/buseyaren/Installation-MaskRCNN/tree/master/Mask_RCNN https://github.com/buseyaren/Installation-MaskRCNN	Installation
2.	Mask R-CNN for Object Detection and Segmentation	

	https://github.com/matterport/Mask_RCNN	
3.	matterport/Mask_RCNN/releases https://github.com/matterport/Mask_RCNN/releases	

Table 3. Prerequisite

Software Prerequisite No.	Description and Version	Note
1.	Ubuntu 18.04	
2.	Python version 3.6	On Ubuntu
3.	Anaconda version 4.7.12	On Ubuntu
Hardware Prerequisite No.	Description and Version	
1.	To be added	

1. Setup The Mask RCNN Environment

1.1. Create the Anaconda environment;

conda create -n maskrcnn python=3.6.12

1.2. Activate the Anaconda environment;

conda activate maskrcnn

1.3. Clone the GitHub folder;

git clone https://github.com/matterport/Mask_RCNN.git

1.4. Modify Mask_RCNN/requirements.txt

tensorflow==1.14.0

keras==2.0.5

opencv-python

h5py==2.10.0

1.5. Install the required Python package;

cd Mask_RCNN

pip install -r requirements.txt

1.6. Download the pre-trained model;

https://github.com/matterport/Mask_RCNN/releases

Download mask_rcnn_balloon.h5 from Mask_RCNN_2.1 Assets and mask_rcnn_coco.h5 from Mask_RCNN_2.0 Assets.

Copy the files to the “Mask_RCNN” folder.

1.7. Run setup.py;

python setup.py install

1.8. Loading the pycocotols module;

pip install git+https://github.com/philferriere/cocoapi.git#subdirectory=PythonAPI

2. Run Mask R-CNN

2.1. Activate the Anaconda environment;

conda activate maskrcnn

2.2. Run Jupyter Notebook

jupyter notebook

2.3. Access http://localhost:8888/tree/ by a browser

http://localhost:8888/tree/

2.4. Navigate to samples folder on the browser

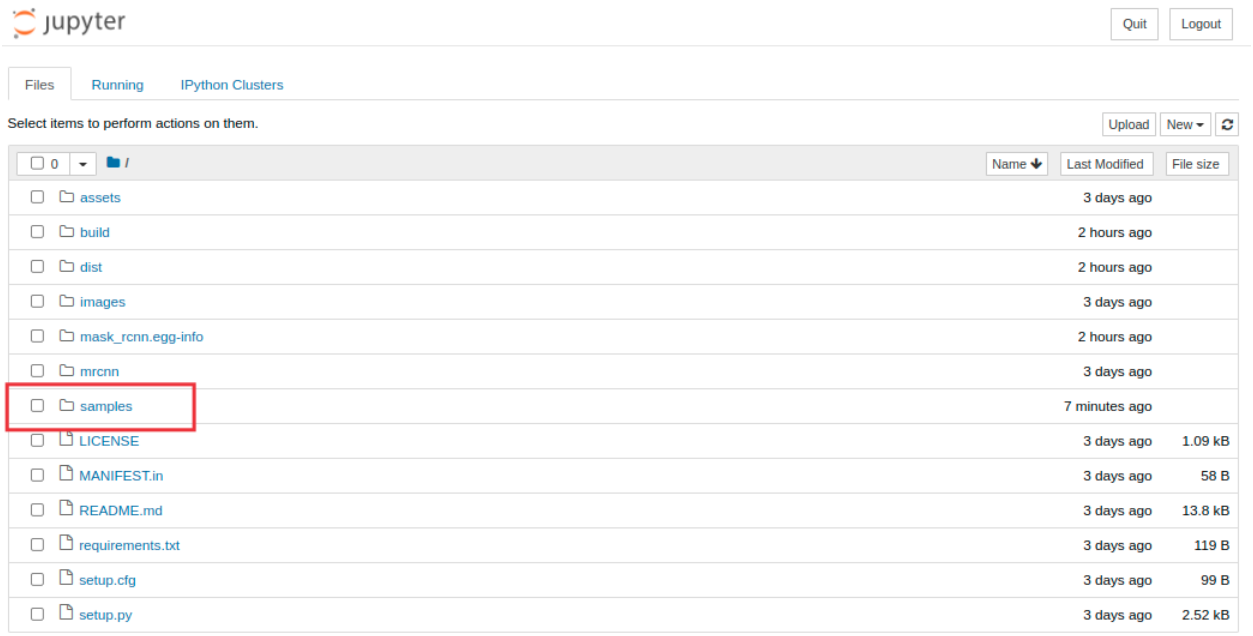

Figure 1. samples folder on Jupyter Notebook

2.5. Open demo.ipynb

Figure 2. demo.ipynb on Jupyter Notebook

2.6. Select a code cell and push the "Run" button

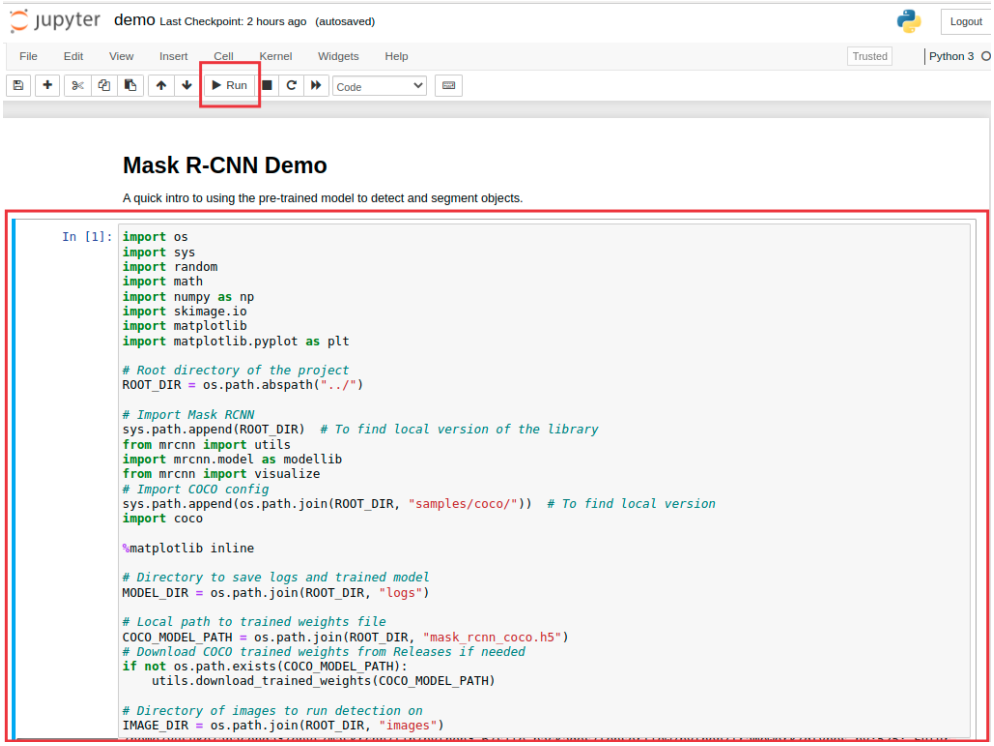

Figure 3. The first code cell on Jupyter Notebook

2.7. Run all code cells

The image is randomly chosen and the demo example is computed.

Run Object Detection

In [15]:
```
# Load a random image from the images folder
file_names = next(os.walk(IMAGE_DIR))[2]
image = skimage.io.imread(os.path.join(IMAGE_DIR, random.choice(file_names)))

# Run detection
results = model.detect([image], verbose=1)

# Visualize results
r = results[0]
visualize.display_instances(image, r['rois'], r['masks'], r['class_ids'],
                            class_names, r['scores'])
```

```
Processing 1 images
image                    shape: (640, 480, 3)         min:    0.00000  max:  255.00000  uint8
molded_images            shape: (1, 1024, 1024, 3)    min: -123.70000  max:  151.10000  float64
image_metas              shape: (1, 93)               min:    0.00000  max: 1024.00000  float64
anchors                  shape: (1, 261888, 4)        min:   -0.35390  max:    1.29134  float32
```

Figure 4. The result of on Jupyter Notebook

(END)