



May 09, 2021

Protocol_mask_RCNN_CT_tensorflow

Ching-Juei Yang^{1,2}, Jing-Yao Wang^{1,2}¹Department of BioMedical Engineering, National Cheng-Kung University, Tainan, Taiwan;²Division of Medical Radiology, E-Da Cancer Hospital, I-Shou University, Kaohsiung, Taiwan

In Development

dx.doi.org/10.17504/protocols.io.buusnwwe

Protocol_mask_RCNN_CT_tensorflow

Ching-Juei Yang

ABSTRACT

This document outlines the implementation of mask-RCNN of computed tomography under tensorflow

This testing protocol includes the public medical images from "The Cancer Imaging Archive" (TCIA) (<https://www.cancerimagingarchive.net/>), Google Brain's Tensorflow (<https://github.com/tensorflow>), TensorBoard, pre-trained COCO dataset, OpenCV library under Python 3 and NVIDIA GPU.

@ Ching-Juei Mark Yang @ Jing-Yao Wang May 9, 2021

DOI

dx.doi.org/10.17504/protocols.io.buusnwwe

PROTOCOL CITATION

Ching-Juei Yang, Jing-Yao Wang 2021. Protocol_mask_RCNN_CT_tensorflow. **protocols.io**
<https://dx.doi.org/10.17504/protocols.io.buusnwwe>

LICENSE

This is an open access protocol distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

CREATED

May 09, 2021

LAST MODIFIED

May 09, 2021

PROTOCOL INTEGER ID

49778

DISCLAIMER:

DISCLAIMER – FOR INFORMATIONAL PURPOSES ONLY; USE AT YOUR OWN RISK

The protocol content here is for informational purposes only and does not constitute legal, medical, clinical, or safety advice, or otherwise; content added to protocols.io is not peer reviewed and may not have undergone a formal approval of any kind. Information presented in this protocol should not substitute for independent professional judgment, advice, diagnosis, or treatment. Any action you take or refrain from taking using or relying upon the information presented here is strictly at your own risk. You agree that neither the Company nor any of the authors, contributors, administrators, or anyone else associated with protocols.io, can be held responsible for your use of the information contained in or linked to this protocol or any of our Sites/Apps and Services.

- 1 Protocol testing datasets were acquired from the collection of "TCGA-LIHC" (<https://wiki.cancerimagingarchive.net/display/Public/TCGA-LIHC>) of the public medical images from TCIA.
- 2 "dcmtojpgpnglTK-v3-subfolder.py" is designed for converting *.dcm to *.png under clinical desired window center and window level.

2.1 dcmtojpgpnglTK-v3-subfolder.py

- 3 "LabelMe" under python 3 was used for region of interest (ROI) labeling
- 4 The folder of ".tfrecordjson/json_test" contains the *.png and corresponding *.json files for program testing

4.1 json_test.7z

- 5 We used tensorflow-gpu 1.2 under python 3.6 version with NVIDIA GEFORCE RTX 2080 Ti
- 6 The details of setting environment was guided as followings mdfile under windows 10 "Environment setup.md"

6.1 Environment setup.md

- 7 The tfrecordjson folder contains files for tensorflow binary data transformation (https://www.tensorflow.org/tutorials/load_data/tfrecord) referring to @author: QingShui-Cheng (TFrecord_json.py) and @author: shirhe-lyh (read_pbtxt_file.py)

execute

```
python TFrecord_json.py --images_dir=./json_test --label_map_path=./liver_lesion_map.pbtxt --output_path=.
```

produce sample_train.record and sample_val.record datasets

7.1 tfrecordjson.7z

the ./tfrecord/json_test contains the *.png and *.json

7.2 read_pbtxt_file.py TFrecord_json.py readmeTFrecord.txt

8 Download tensorflow/models (<https://github.com/tensorflow/models>)

9 Download pipeline.config (mask_rcnn_resnet101_atrous_coco.config) and coco-trained models from tensorflow model zoo (https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf1_detection_zoo.md) eg. mask_rcnn_resnet101_atrous_coco.tar

10 building the folder of "liver_resnet"

```
models-master - official
- research - liver_resnet - dataset - sample_train/val.record
- model
  (model.ckpt.data-0000-of-0001
  model.ckpt.index
  model.ckpt.meta
  pipeline.config)
- trained_model
- *.pbtxt, pipeline.config
  (liver_lesion_map.pbtxt,
  mask_rcnn_resnet101_atrous_coco.config)
- sample
- tutorials
```

10.1  **liver_resnet.7z**

11 Modifying and setting your pipeline.config

1. Setting tf_record_input_reader with your local tfrecord path (include training and eval)
2. Setting label_map_path with your local label_map file path (include training and eval)
3. Setting your fine_tune_checkpoint path

12 execute the train process

```
at reaserch path
python ./object_detection/model_main.py --alsologtostderr --model_dir=./liver_resnet/trained_model --
pipeline_config_path=./liver_resnet/mask_rcnn_resnet101_atrous_coco.config
```

12.1  **readmetrain.txt**  **mask_rcnn_resnet101_atrous_coco.config**

13 execute the tensorboard

```
at research path
tensorboard --logdir=./liver_resnet/trained_model --port=####
(eg. tensorboard --logdir=./liver_resnet/trained_model --port=6006)
```

by guiding of tensorboard command line, please open chrome webBrowser and type
"https://*name_of_computer:####"

record the optimal training steps to avoid overfitting

14 export the trained model from optimal.ckpt to *.pb

```
at research path
python object_detection/export_inference_graph.py --input_type image_tensor --pipeline_config_path
liver_resnet/mask_rcnn_resnet101_atrous_coco.config --trained_checkpoint_prefix
liver_resnet/trained_model/model.ckpt-#### --output_directory liver_resnet/export_model/
```

15 Using inference_contour_LD.py, optimal_model.pb, and *.pbtxt(eg. liver_lesion_label_map.pbtxt) for detecting *.png

15.1  liver_lesion_label_map.pbtxt
 inference_contour_LD.py