Image-Segmentation-YOLOv8-Pytorch-GPL-Jupyter

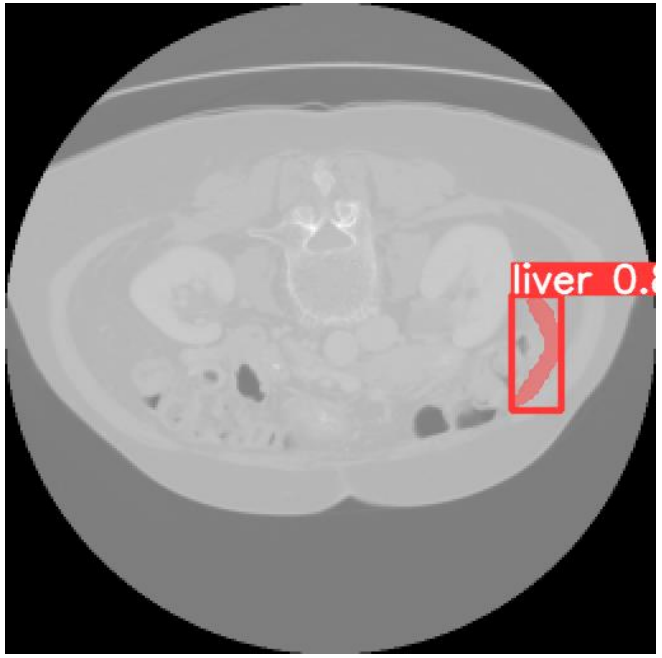
Ultralytics YOLOv8 is a cutting-edge, state-of-the-art (SOTA) model developed by Ultralytics.

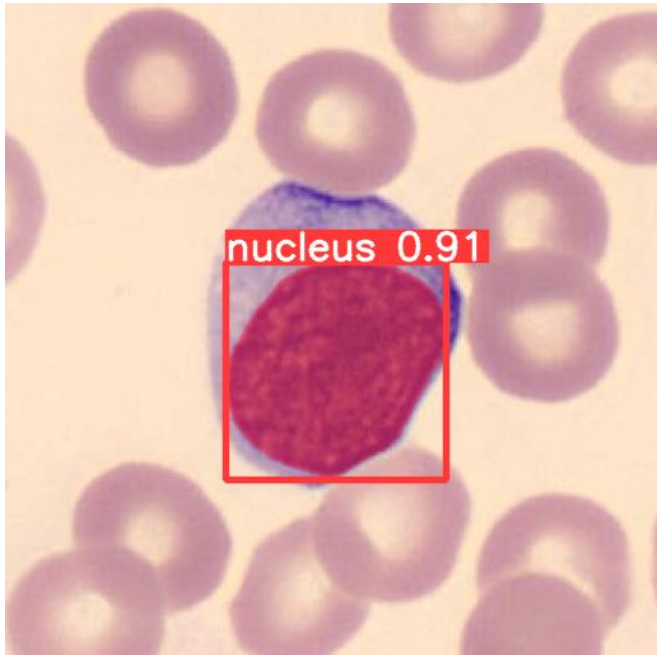
It builds on the previous successful version of YOLO, introducing new features and improvements that further enhance its performance and flexibility.

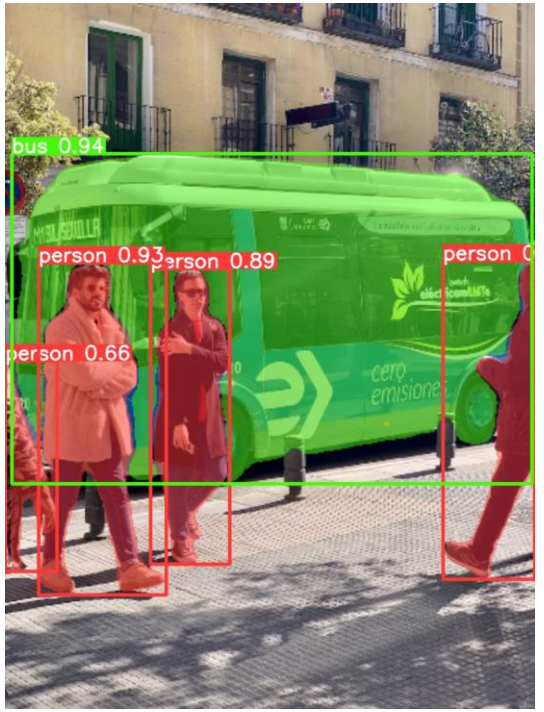
Version 20230223

Applications

- The YOLOv8 segmentation can be applied to factory defect detection,
- medical image analysis, biological image analysis, industrial safety
- image analysis, etc.



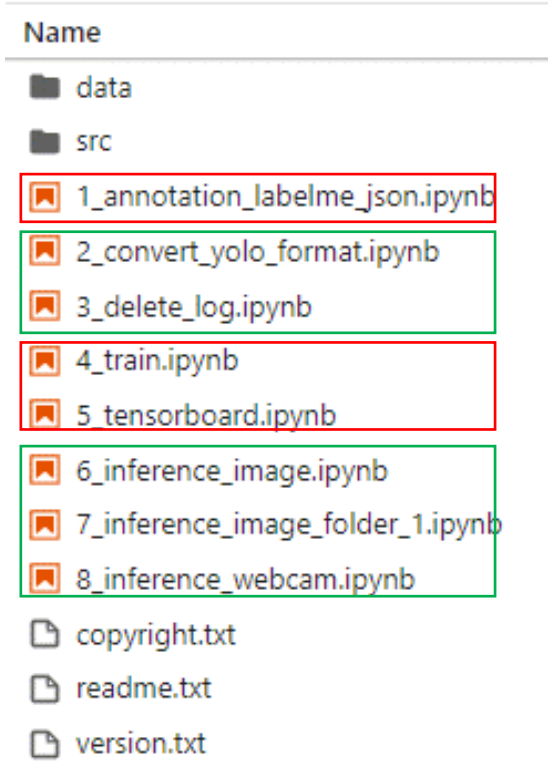




How to use

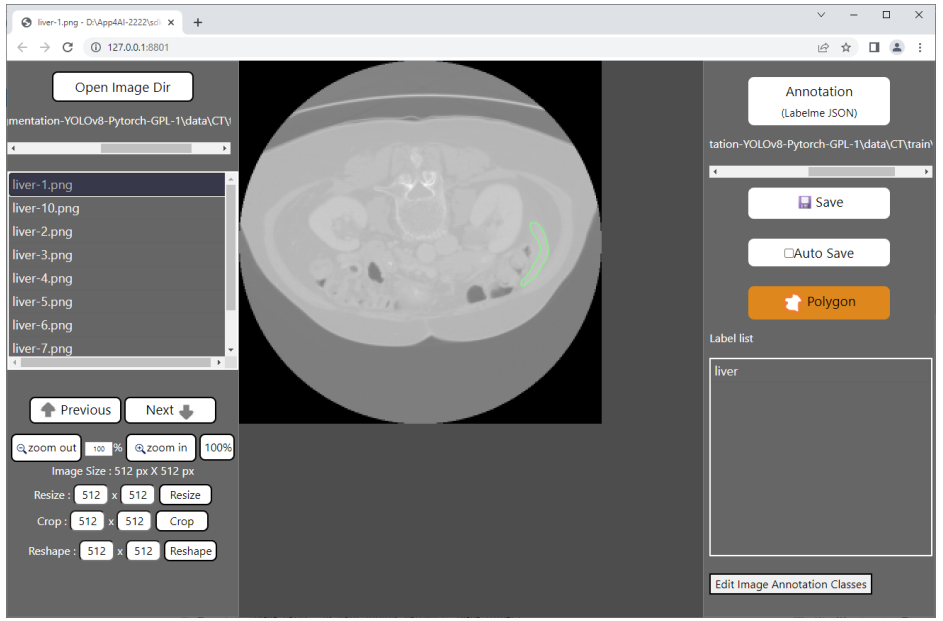
The main process is:

Annotate images -> Prepare files for training -> Training -> Inference



1_annotation_labelme_json.ipynb

Open the webpage for image annotation.



ipynb parameter:

- “port” is the port used by the webpage. If the port is occupied by the user, please change another port value by yourself.
- “dataset” is the dataset name
- “label_folder” is the image of the train folder, it can also be changed to “val” to label the image of the val folder.

See Annotation.pdf for how to use annotation pages.

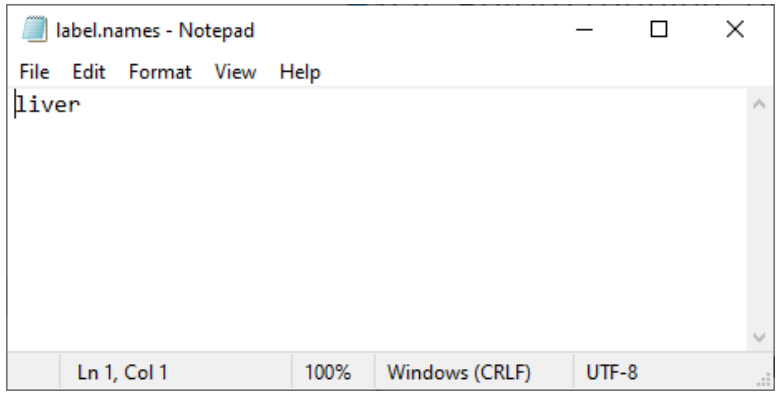
2_convert_yolo_format.ipynb

Convert the labelme json label file to the yolo format. Before running, please confirm label.names under the label_file path in #parameters and whether the content filled in the category is correct.

supplement:

The content of label.names is the category name without background.

If there are more than two category names, represent each category name with a line break.

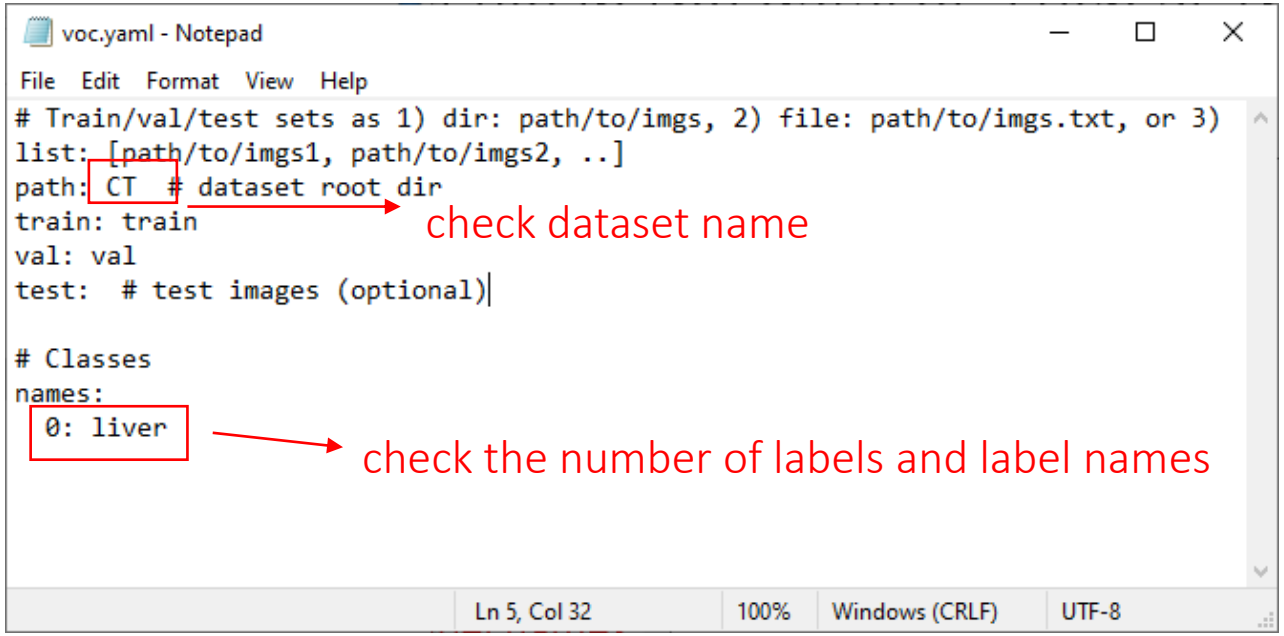


3_delete_log.ipynb

Delete the log files left over from previous training.

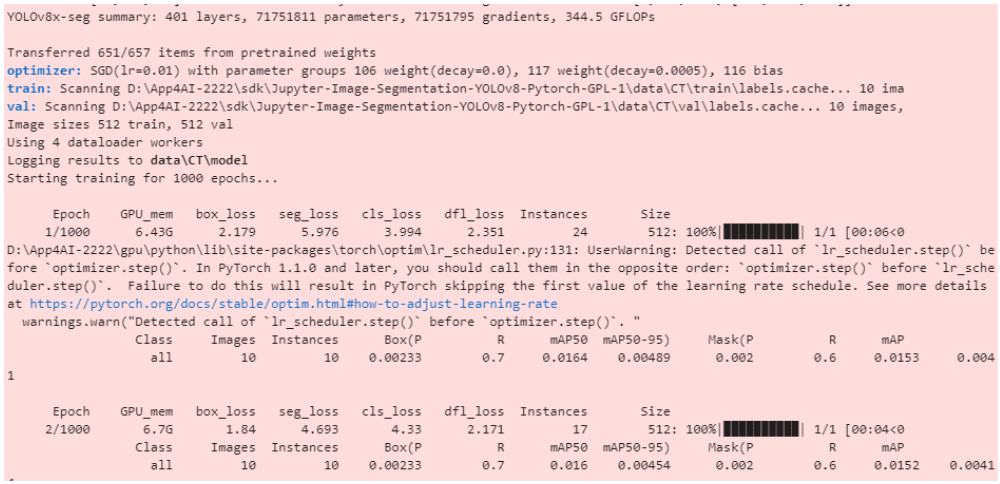
Set training related files

Confirm the content of the voc.yaml file in the dataset, such as the name of the dataset, the number of categories, and the name.



4_train.ipynb

Start training.



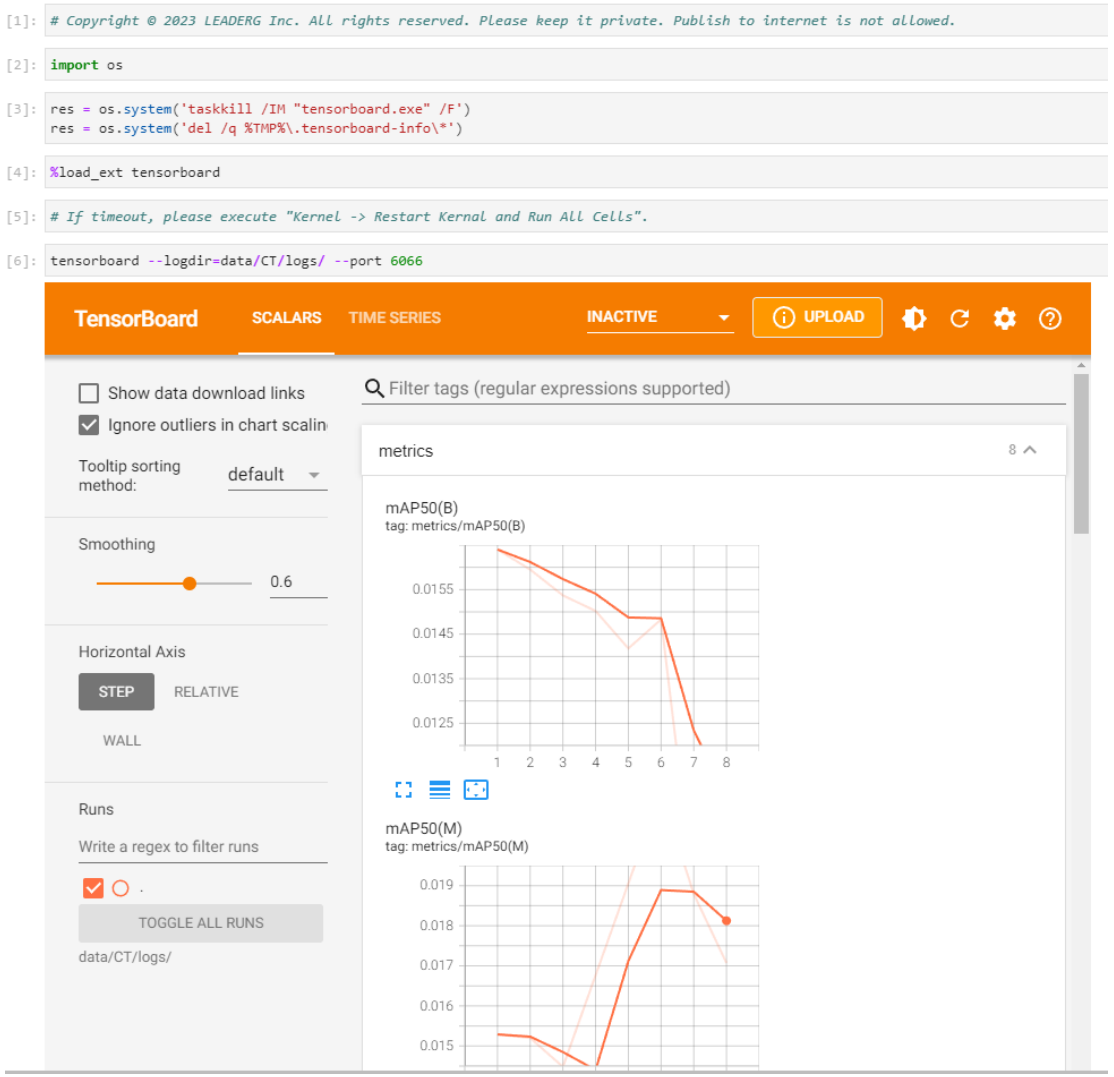
ipynb parameter:

- dataset is the dataset name.
- weights_file is the pretrained model path used,

  None means not to use the pretrained model for training.
- devices is the GPU id used.
- epochs is the number of training epochs.

5_tensorboard.ipynb

You can view the training loss curve and other related information through TensorBoard.



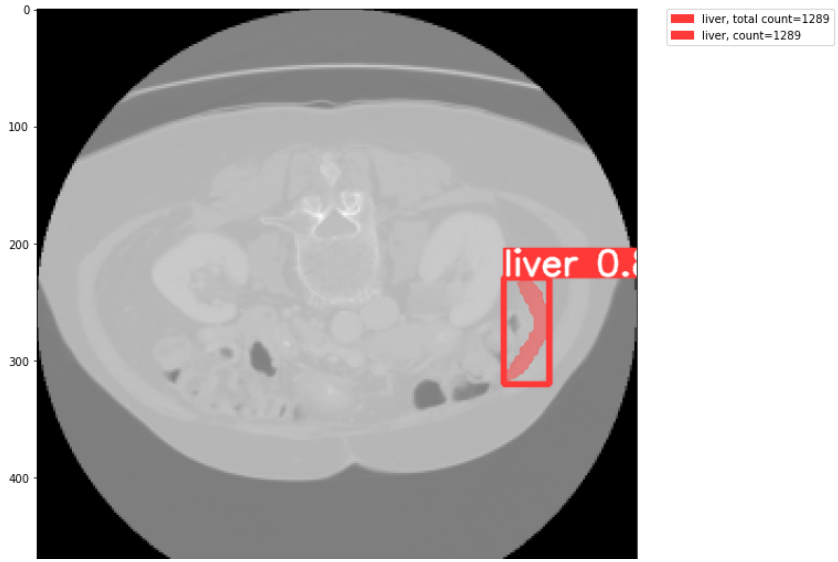
6_inference_image.ipynb

Infer a single image.

ipynb parameter:

- dataset is the dataset name.
- source is the inferred image path.
- weights_file is the inference model path.

```
[1]: # Copyright © 2023 LEADERG Inc. All rights reserved. Please keep it private. Publish to internet is not allowed.

[2]: %matplotlib inline

[3]: import matplotlib.pyplot as plt
     plt.rcParams['figure.figsize'] = (10.0, 10.0)

[4]: dataset = "CT"
     dataset_path = "data/%s" %(dataset)
     source = "data/%s/test/images/liver-1.png" %(dataset)

     image_size = 512

     weights_file = "data/%s/model/best.pt" %(dataset)

     device = "0" # 0, 1, 2, … for Nvidia GPU or cpu for CPU
     threshold = 0.5

[5]: %run src/predict.py model=$weights_file conf=$threshold source=$source imgsz=$image_size show=True device=$device project=$dataset_

Ultralytics YOLOv8.0.6  Python-3.9.12 torch-1.12.0+cu113 CUDA:0 (NVIDIA TITAN RTX, 24576MiB)
Fusing layers...
YOLOv8x-seg summary: 295 layers, 71721619 parameters, 0 gradients, 343.7 GFLOPs
```

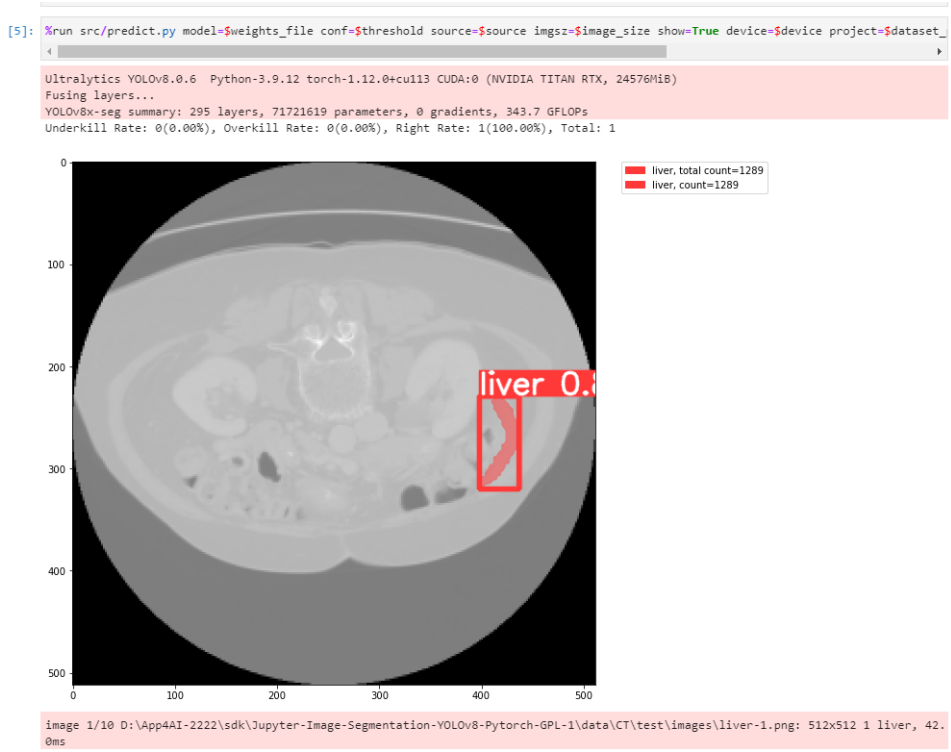


7_inference_image_folder_1.ipynb

Infer all images in the folder.

ipynb parameter:

- dataset is the dataset name.
- source is the inferred image path.
- weights_file is the inference model path.



8_inference_webcam.ipynb

Infer the image of the webcam. Press “q” on the display to turn the webcam off.

It takes a while to display after running

Reference

- Please refer to the readme.txt in the SDK folder.
- LEADERG AppForAI: https://www.leaderg.com/appforai-windows
-