2022 DeepLearning Term Project

Object Detection

Team F

201735802 Kang DooWon
201735820 Kim JaeUk
201735826 Rhew Changmeen
201935006 Ko HyoJin

Submission date: 2022.06.05
Course name: Deep Learning
Professor: Cha Youngwoon
Course id: 14448_01
Major: Dept. of Software

CONTENTS

1. Selected topic

Topic

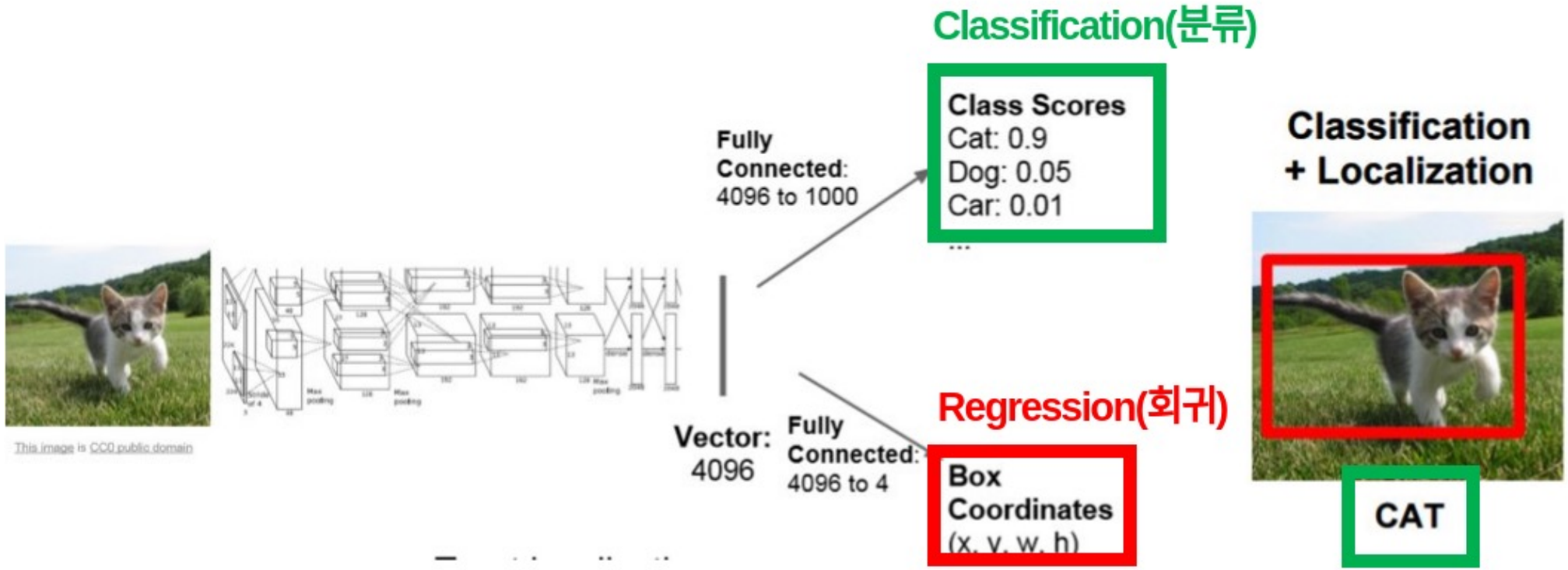
Object Detection

- **Classification**
  : Classification of object
- **Localization**:
  : Get the object coordinates from the image

https://wordbe.tistory.com/entry/Object-Detection-%EC%98%88%EC%A0%9C-%EC%84%A4%EB%AA%85-%EC%BD%94%EB%93%9C%ED%8F%AC%ED%95%A8-Pytorch

1. Selected topic

Topic

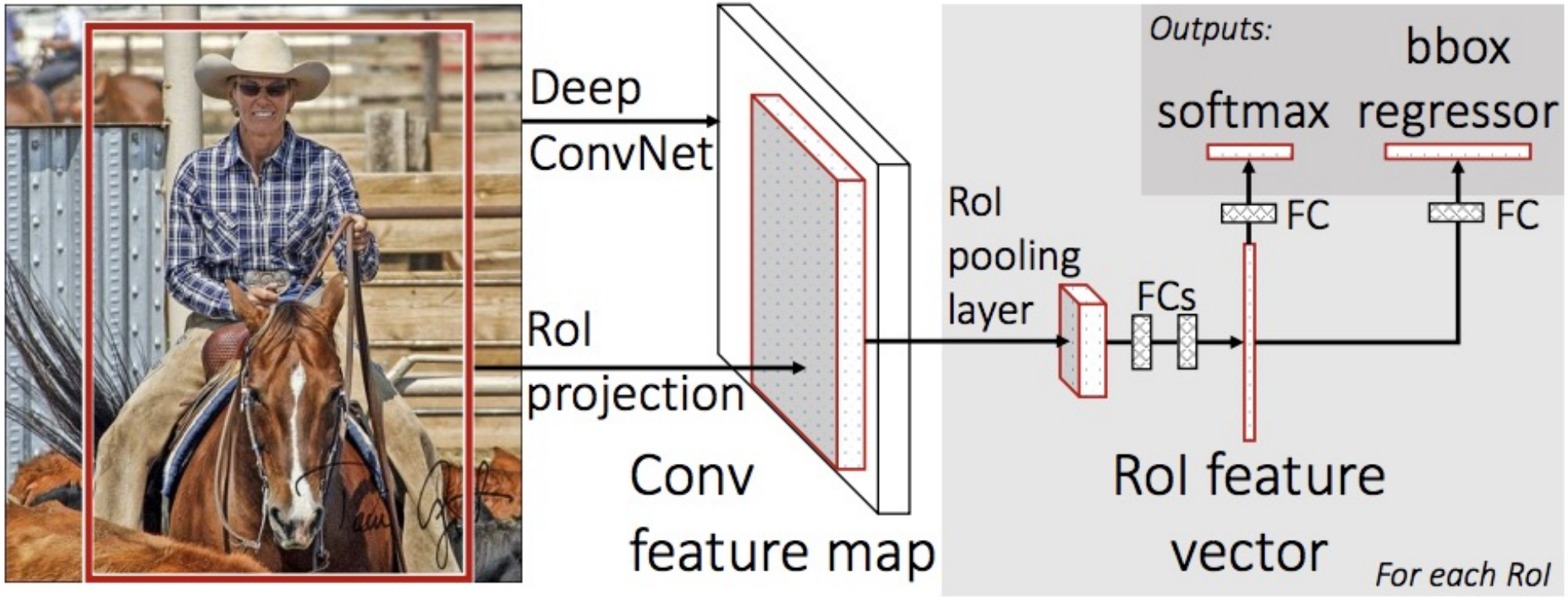


https://chacha95.github.io/2020-02-10-Object-Detection1/

1. Selected topic

Topic

Faster R-CNN

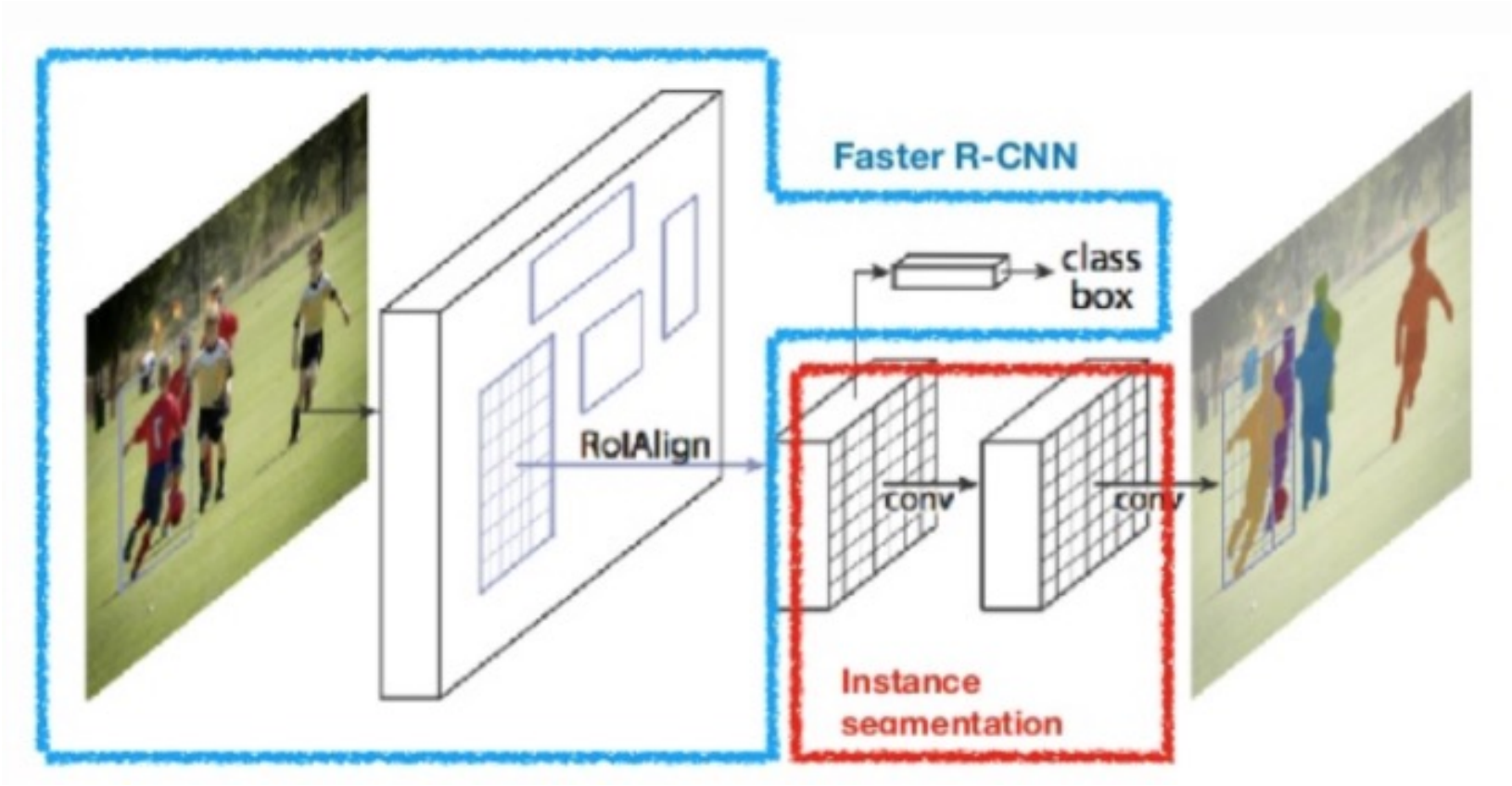


https://tutorials.pytorch.kr/intermediate/torchvision_tutorial.html

1. Selected topic

Topic

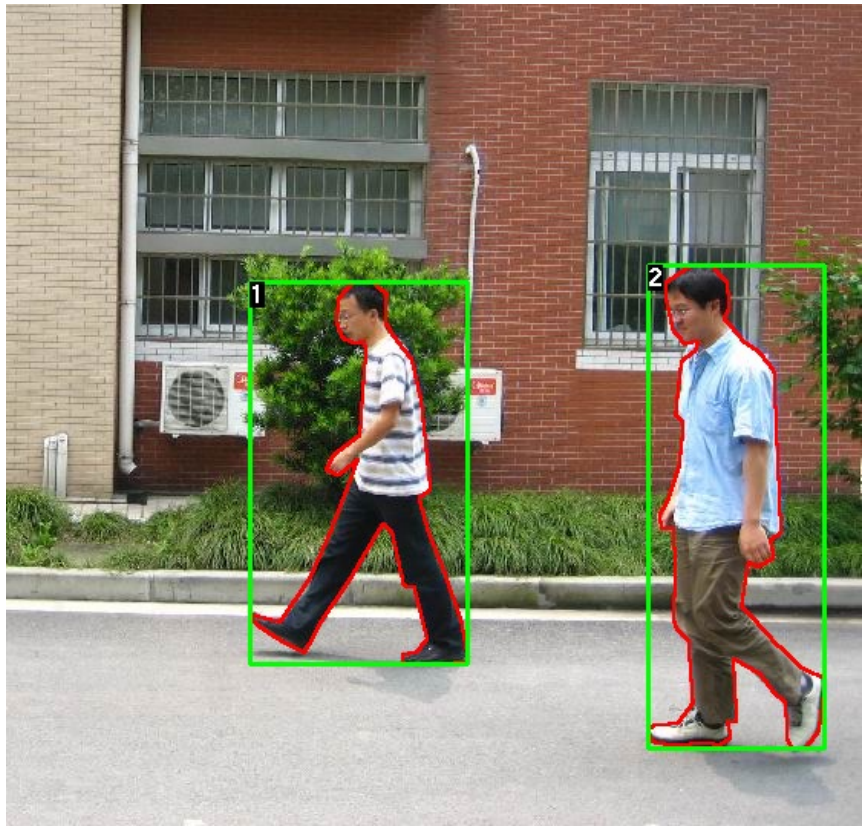
Mask R-CNN



https://wordbe.tistory.com/entry/Object-Detection-%EC%98%88%EC%A0%9C-%EC%84%A4%EB%AA%85-%EC%BD%94%EB%93%9C%ED%8F%AC%ED%95%A8-Pytorch

1. Selected topic

Object Detection using Mask R-CNN



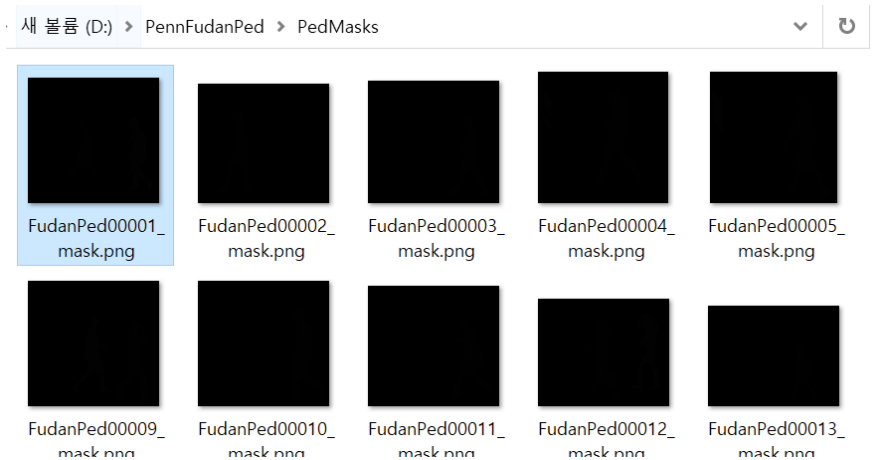
Mask R-CNN
(Kaiming He, Georgia Gkioxari, Piotr Dollar, R oss Girshick – Facebook AI Research)

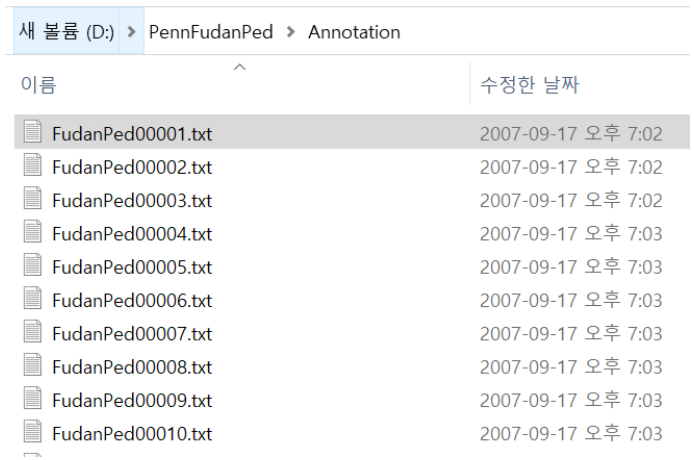
- Instance Segmentation and Object Detection
- Keypoint Detection
- Mask R-CNN for Human Pose Estimation

https://pytorch.org/tutorials/intermediate/torchvision_tutorial.html#torchvision-object-detection-finetuning-tutorial

1. Selected topic

Dataset





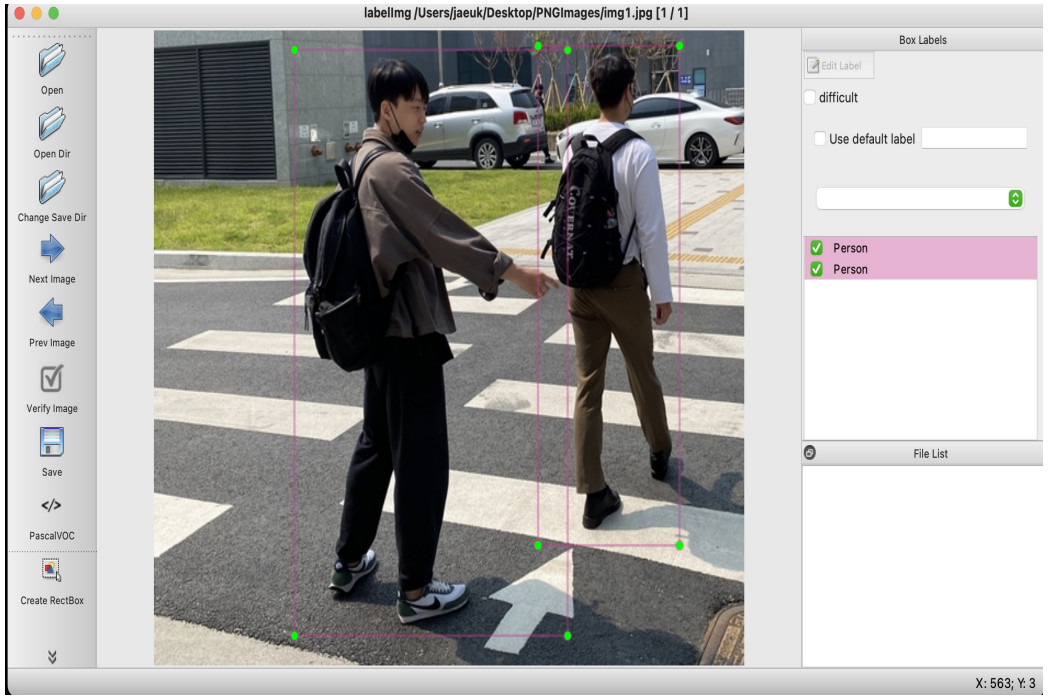
- **PNG Images**
  : Original images
- **PedMask**
  : Segmentation mask
- **Annotation**
  : Label information

2. Collected
New Dataset

New Image

2. Collected New Dataset

Label



https://github.com/tzutalin/labelImg

2. Collected
New Dataset

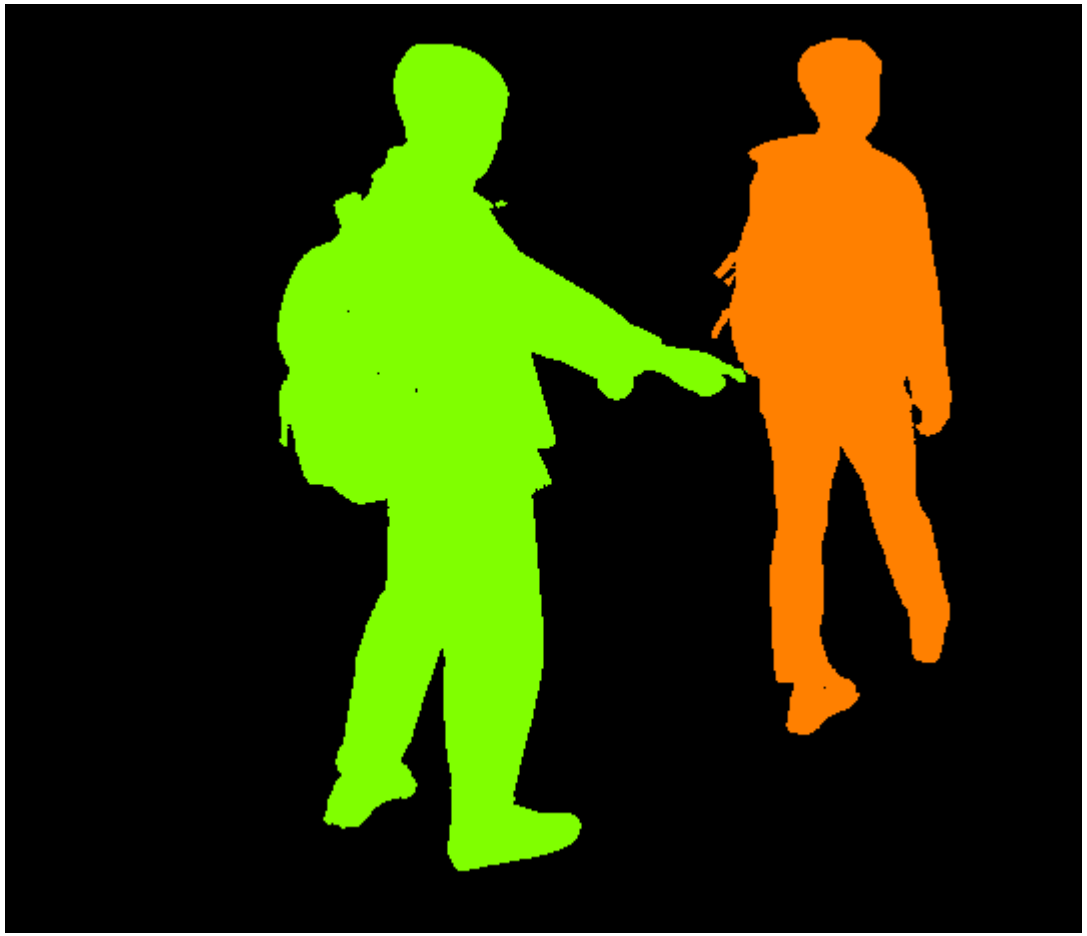
Mask

removebg

https://www.remove.bg/ko

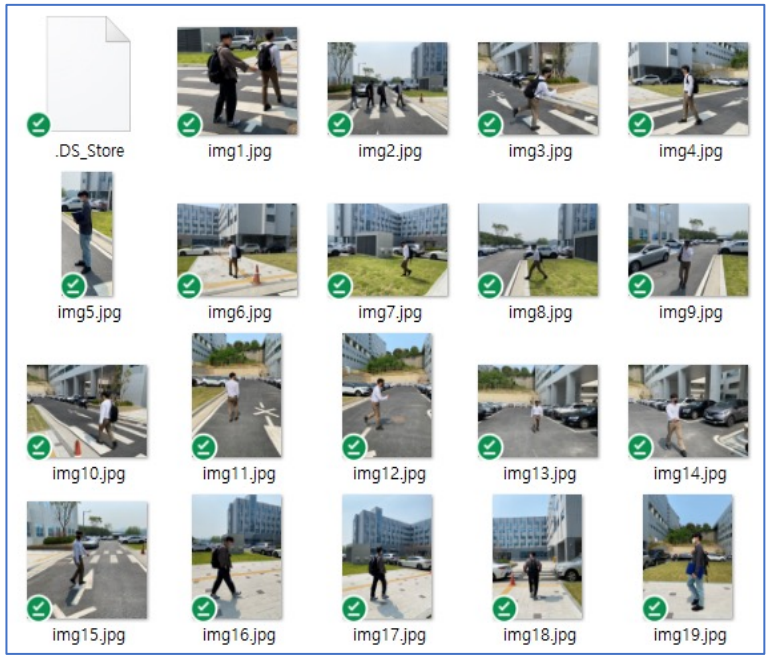
2. Collected
New Dataset

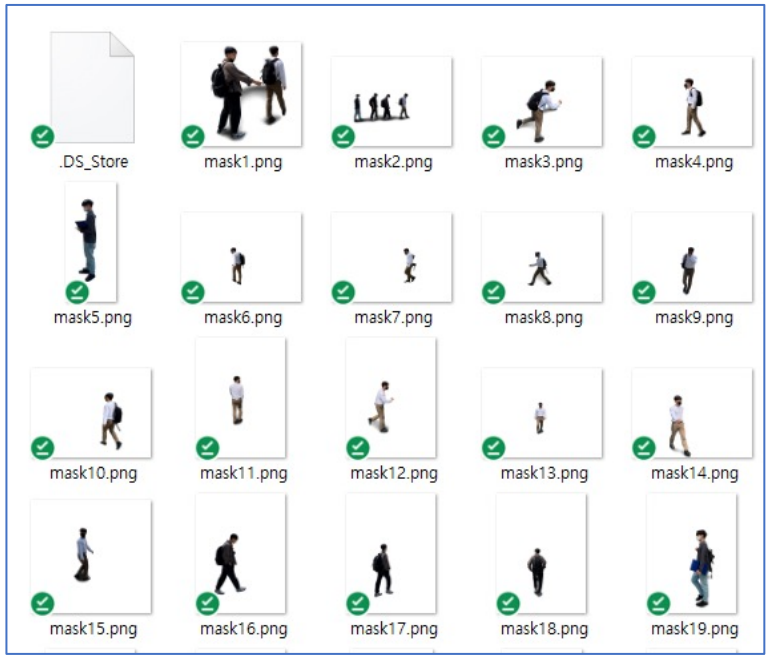
Label

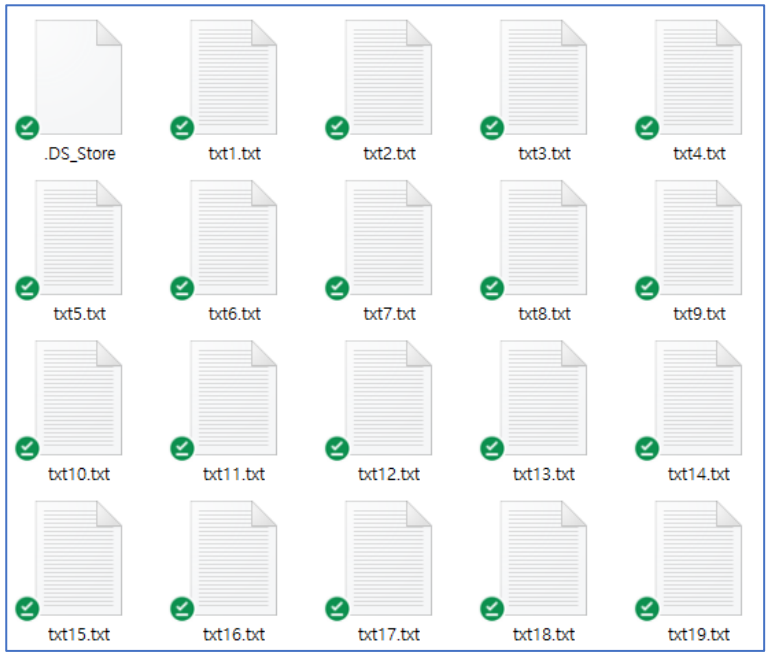


2. Collected New Dataset

Custom Dataset







- **PNG Images**
  : Original images

- **PedMask**
  : Segmentation mask

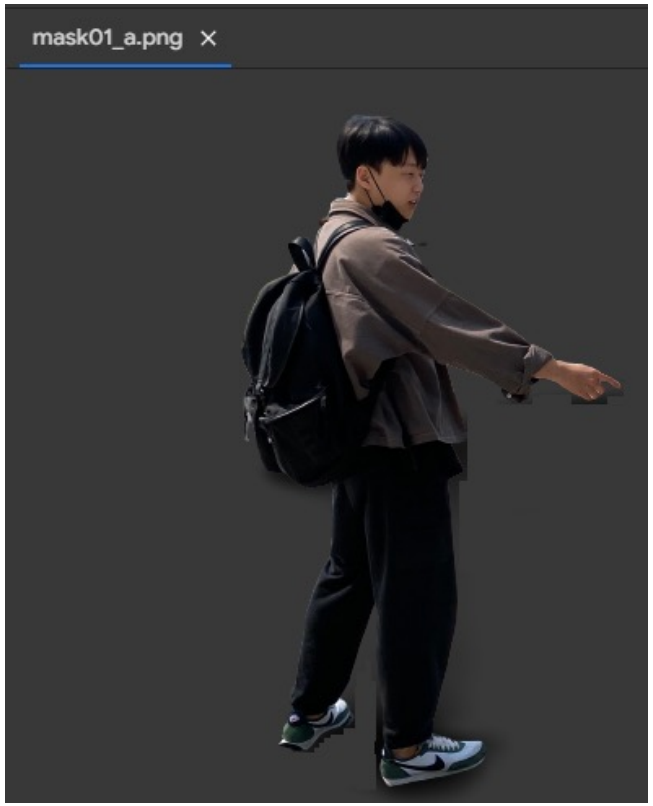
- **Annotation**
  : Label information

2. Collected New Dataset

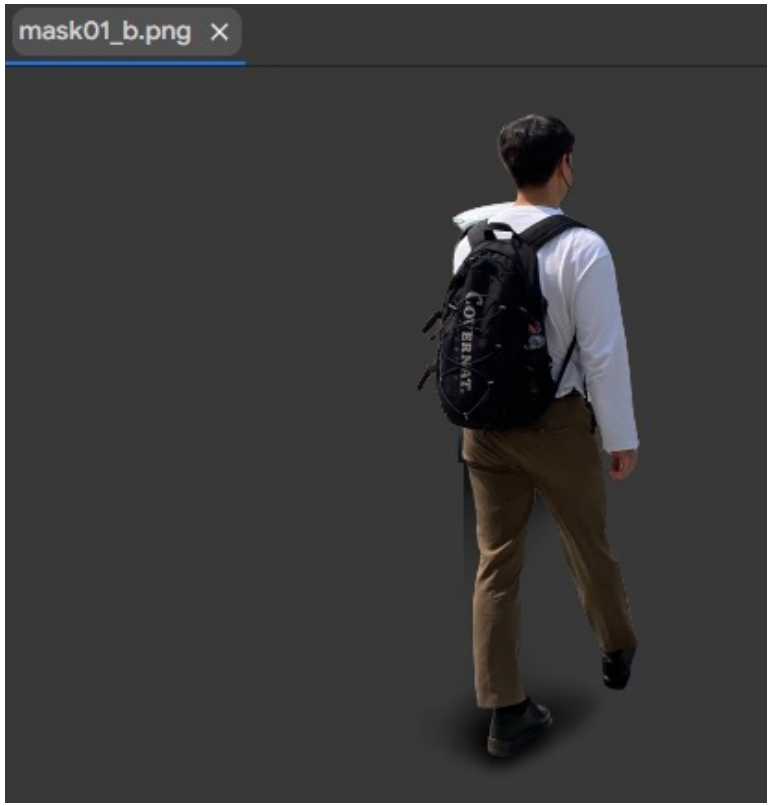
Image Visualization

maskimg01.png

maskimg01_a.png



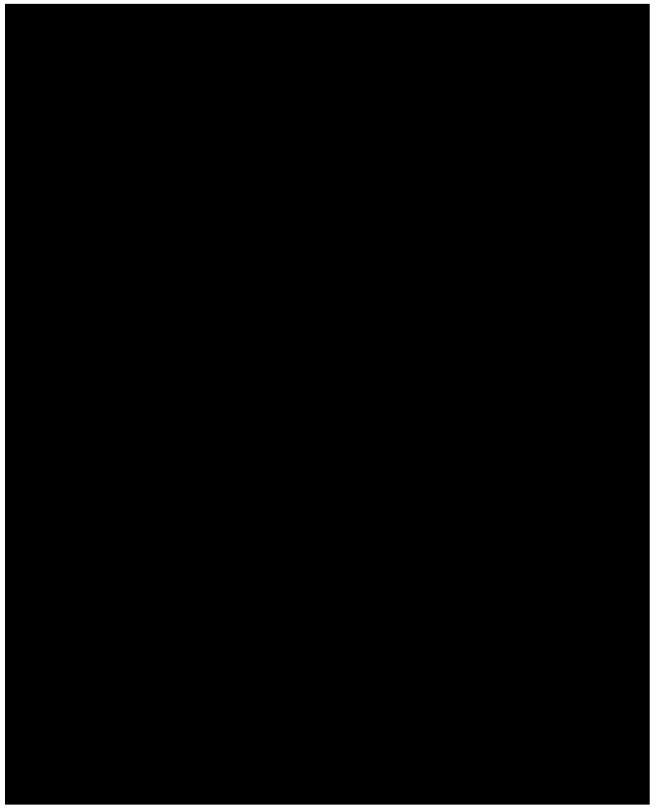
maskimg01_b.png



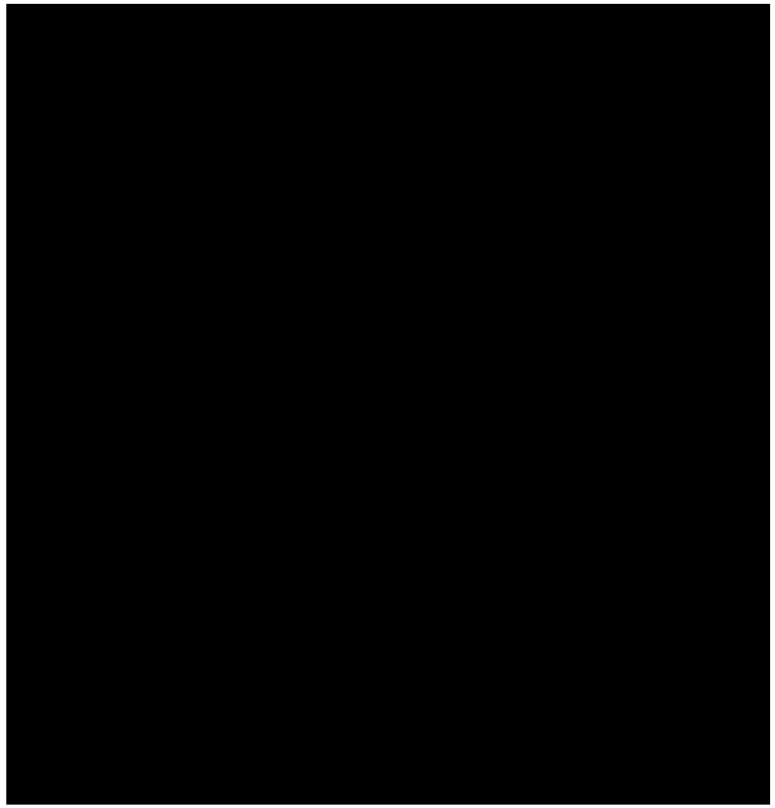
2. Collected New Dataset

Image Visualization

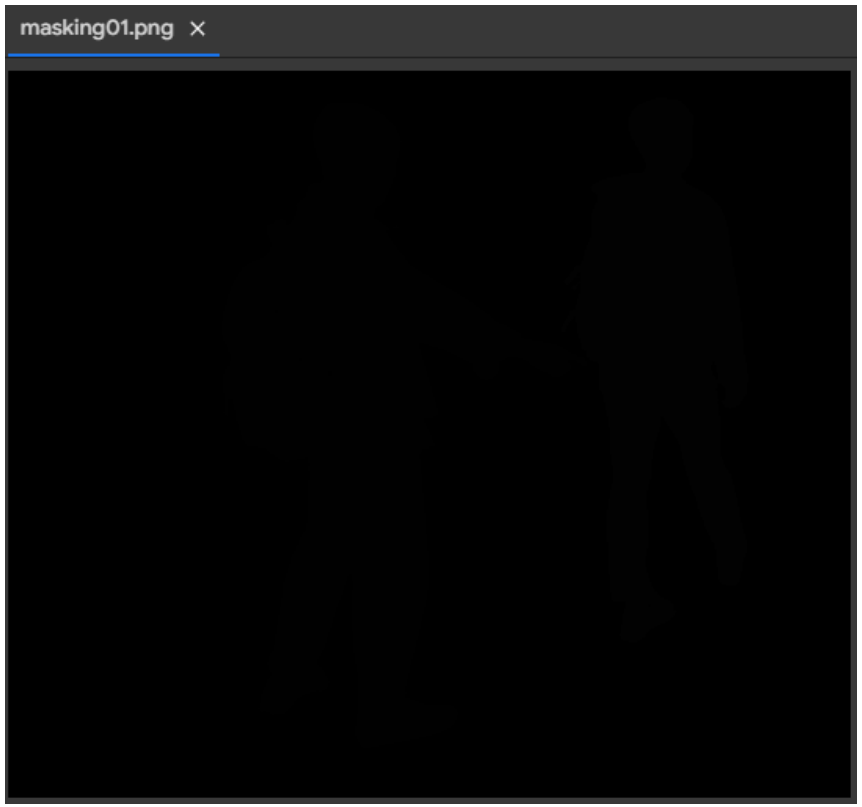
maskimg01_a.png



maskimg01_b.png



maskimg01.png



2. Collected New Dataset

Image Visualization

128
255
0

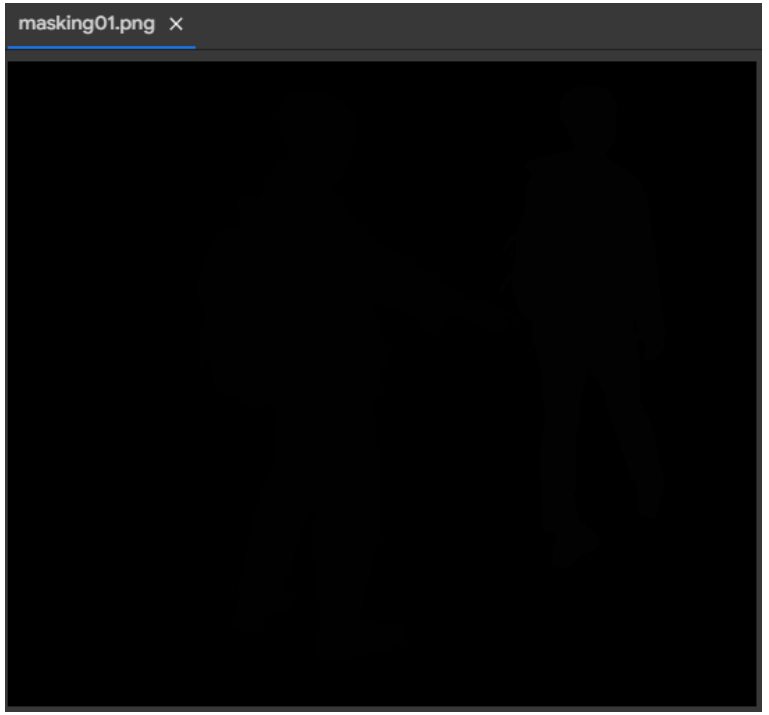
255
128
0

```python
def get_custom_palette():
    palette_unit = [128, 255, 0,
                    255, 128, 0,
                    0, 255, 128,
                    0, 128, 155,
                    64, 255, 64,
                    128, 64, 255,
                    128, 255, 255,
                    255, 0, 128,
                    255, 128, 255]
    palette = []
    for i in range(9):
        palette += [int(p/2**i) for p in palette_unit]

    palette = [0, 0, 0] + palette

    return palette
```

maskimg01.png



maskimg01.png

3. Results on Transfer Learning

Train, Validation, Test

```
# define training and validation data loaders
dl = torch.utils.data.DataLoader(
    dataset_train, batch_size=8, shuffle=False, collate_fn=utils.collate_fn)

custom_dl = torch.utils.data.DataLoader(
    custom_dataset_train, batch_size=8, shuffle=False, collate_fn=utils.collate_fn)

dl_val = torch.utils.data.DataLoader(
    dataset_val, batch_size=1, shuffle=False, collate_fn=utils.collate_fn)

custom_dl_val = torch.utils.data.DataLoader(
    custom_dataset_val, batch_size=1, shuffle=False, collate_fn=utils.collate_fn)

dl_test = torch.utils.data.DataLoader(
    dataset_test, batch_size=1, shuffle=False, collate_fn=utils.collate_fn)

custom_dl_test = torch.utils.data.DataLoader(
    custom_dataset_test, batch_size=1, shuffle=False, collate_fn=utils.collate_fn)
```

Train

Validation

Test

3. Results on Transfer Learning

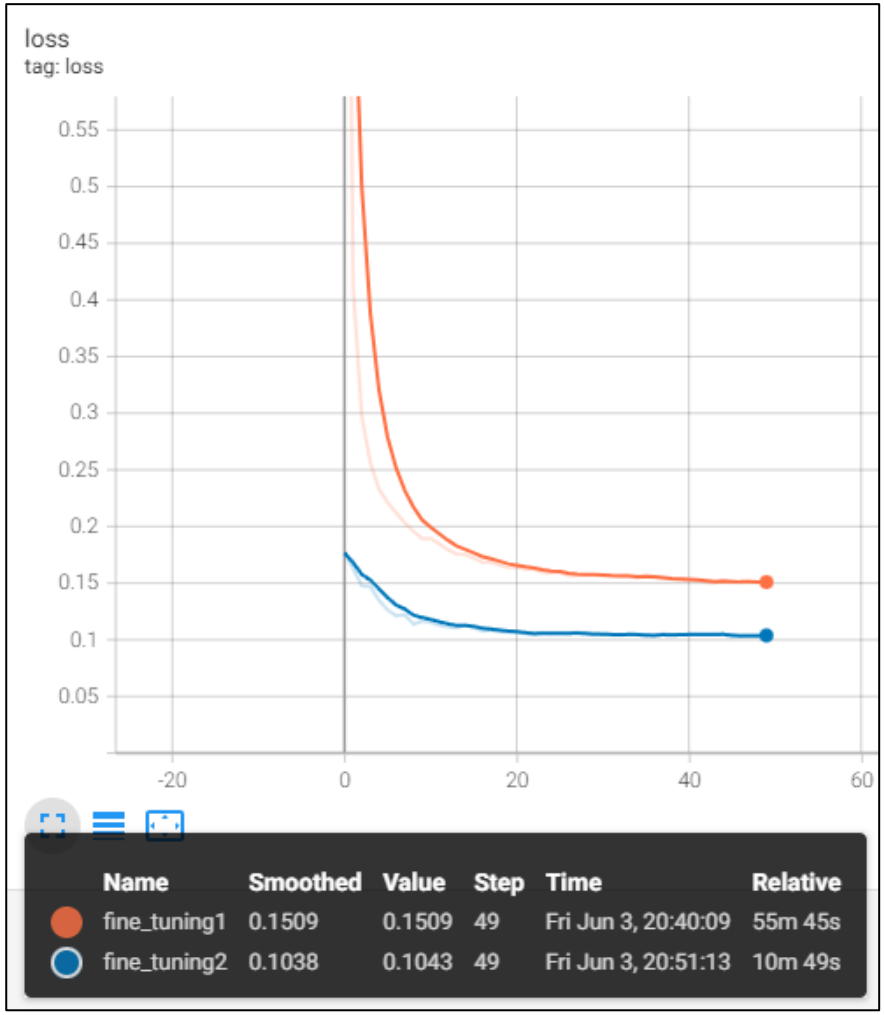
Test

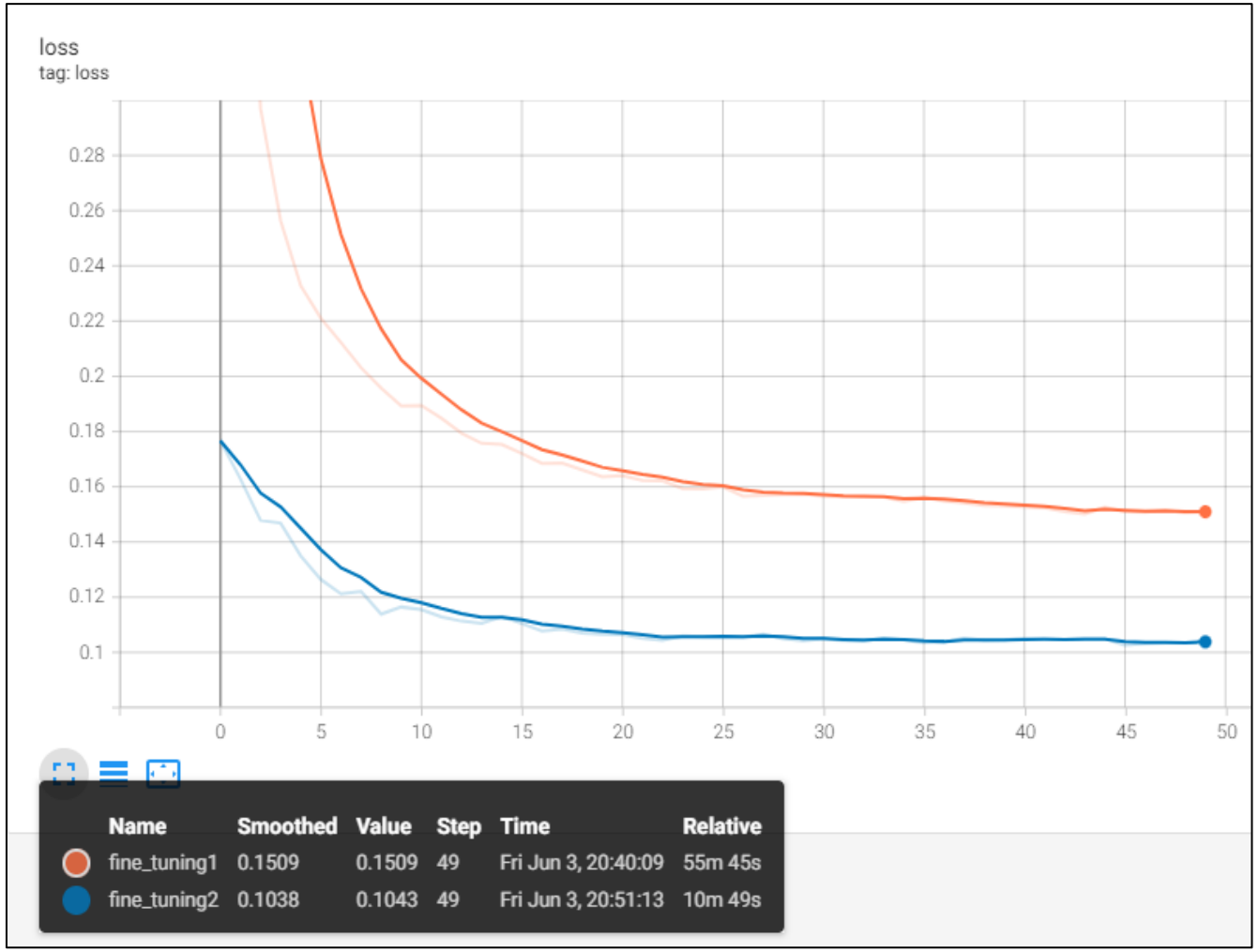
Test for 3 cases

```
test(model, dl_test, device, "before_train")
test(model, custom_dl_test, device, "before_train_costum")
fine_tuning(model, optimizer, scheduler, dl, dl_val, device, "fine_tuning1", num_epochs=num_epochs)
test(model, dl_test, device, "after_finetuning1")
test(model, custom_dl_test, device, "after_finetuning1_custom")
fine_tuning(model, optimizer2, scheduler2, custom_dl, custom_dl_val, device, "fine_tuning2", num_epochs=num_epochs)
test(model, dl_test, device, "after_finetuning2")
test(model, custom_dl_test, device, "after_finetuning2_custom")
```

3. Results on Transfer Learning

Train

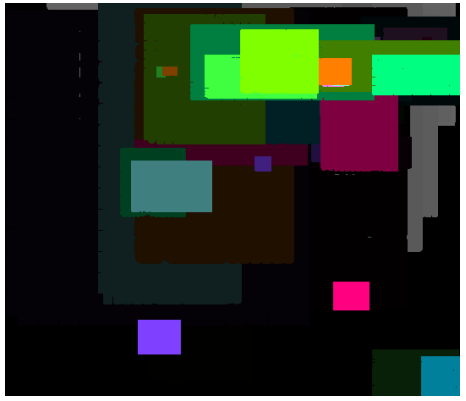


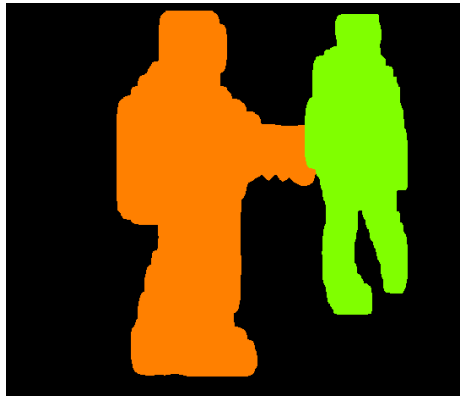


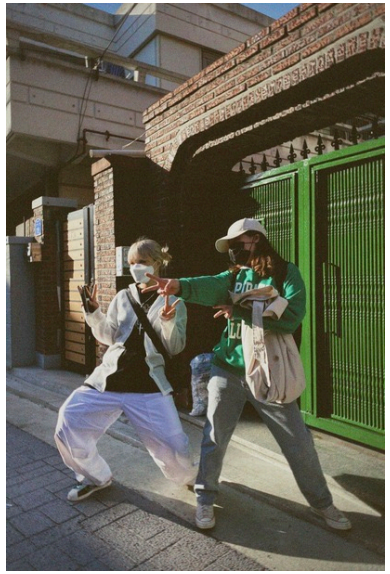
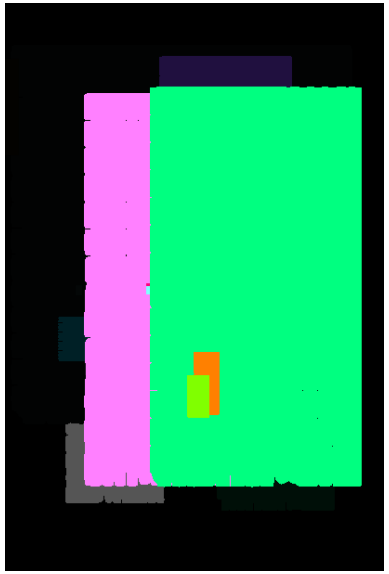
3. Results on Transfer Learning

Test Result

Original Data | **Before Train** | **Fine Tuning1** | **Fine Tuning2**



4. Reference

Reference

https://chacha95.github.io/2020-02-10-Object-Detection1/

”Mask R-CNN”, Kaiming He, Georgia Gkioxari, Piotr Dollar, Ross Girshick, arXiv:1703.06870v3 [cs.CV] 24 Jan 2018

https://wordbe.tistory.com/entry/Object-Detection-%EC%98%88%EC%A0%9C-%EC%84%A4%EB%AA%85-%EC%BD%94%EB%93%9C%ED%8F%AC%ED%95%A8-Pytorch

https://colab.research.google.com/drive/1pkGJud_G6V706qEzP5leJt85R7RTx_xt#scrollTo=5v5S3bm07SO1

https://colab.research.google.com/drive/1pkGJud_G6V706qEzP5leJt85R7RTx_xt#scrollTo=5v5S3bm07SO1

https://www.thecrimson.com/article/2019/9/27/reevaluating-pedestrian-safety/

Dataset: Penn-Fudan Database for Pedestrian Detection and Segmentation
https://www.cis.upenn.edu/~jshi/ped_html/

Thank you