Network 종류 변경하여 비교해보기

2021. 07. 29

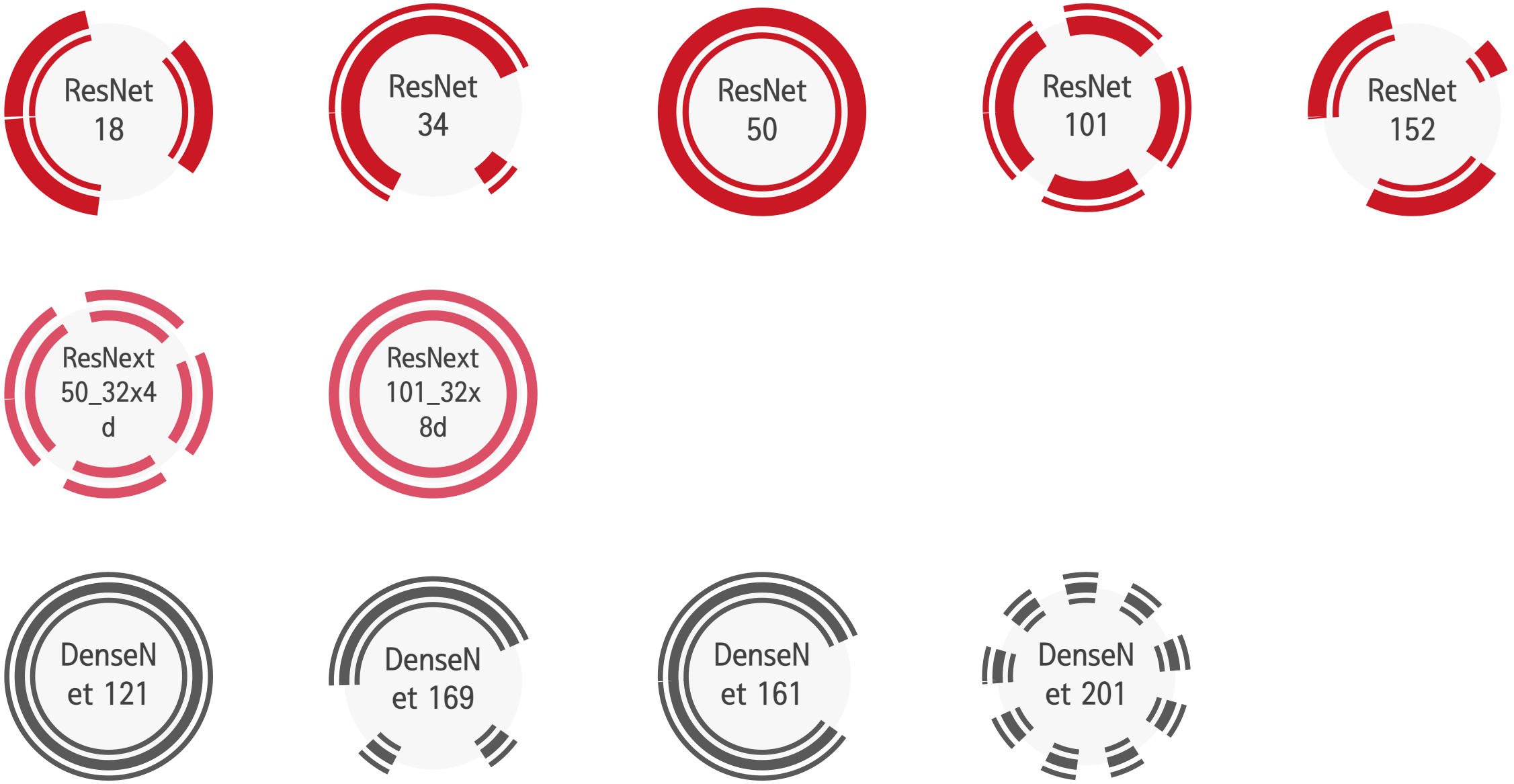
길다영

https://github.com/sunset1995/HorizonNet 참고.

Contents

1 ResNet 종류

1 Network 종류

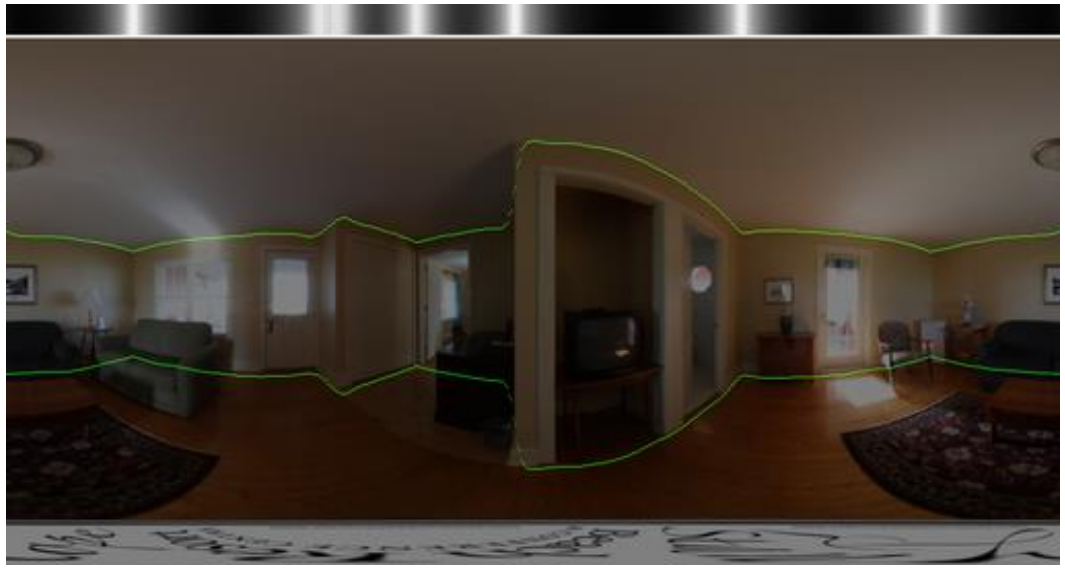
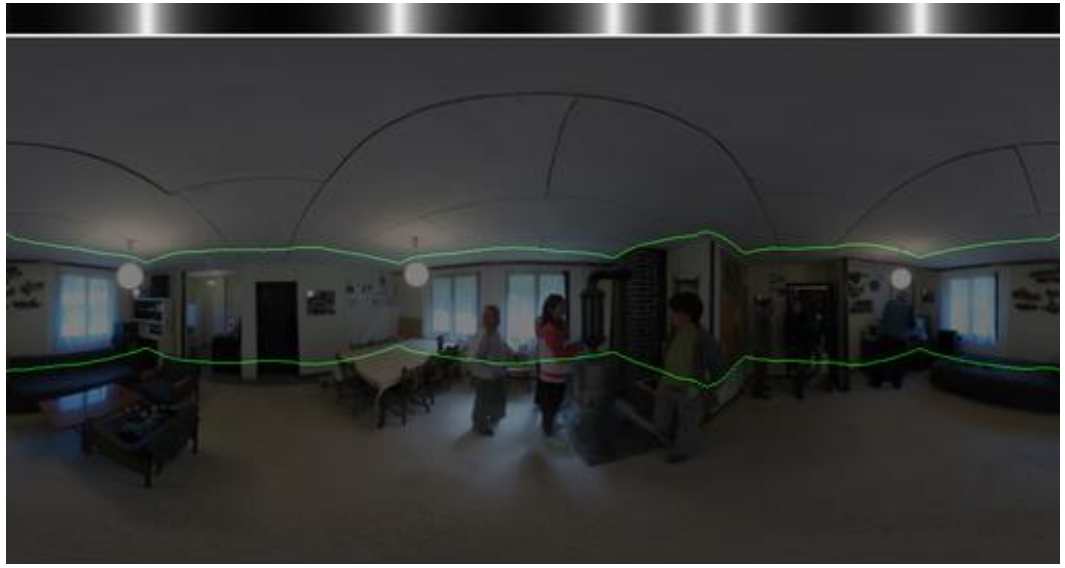


2 ResNet

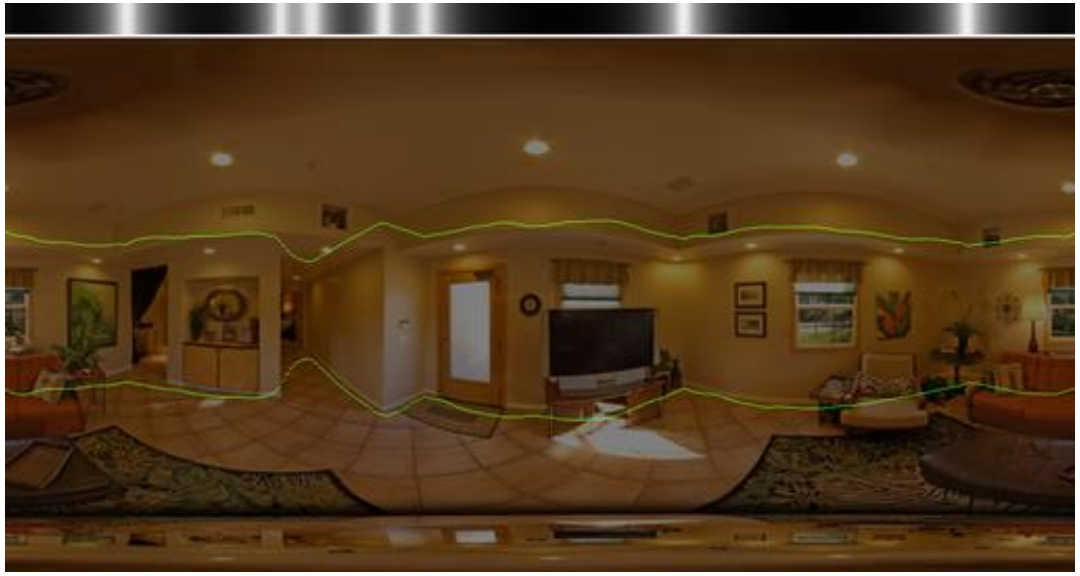
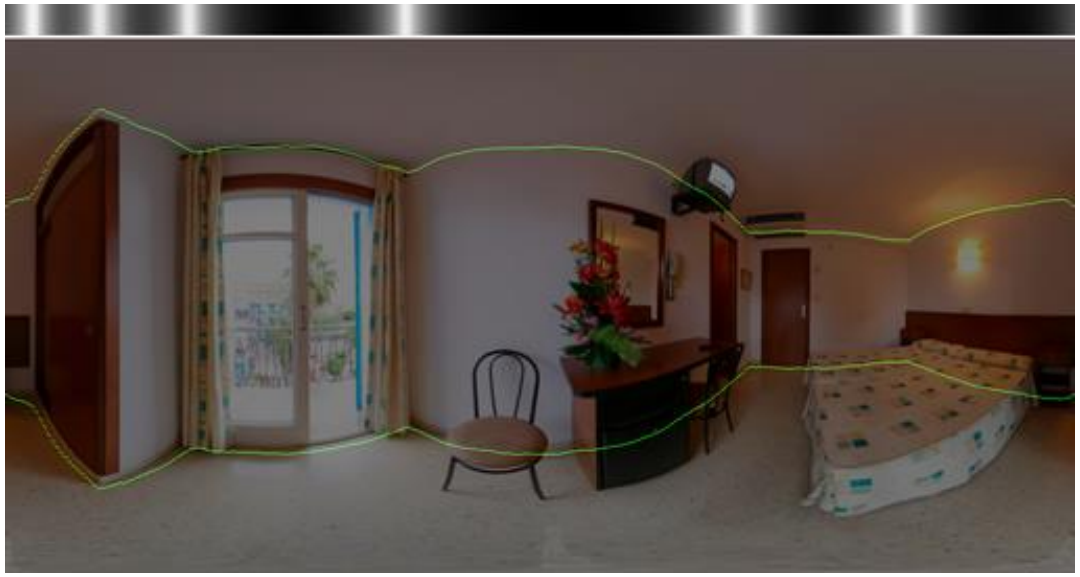
- **ResNet50**
- ResNet101
- ResNet34
- ResNet152

- 분석

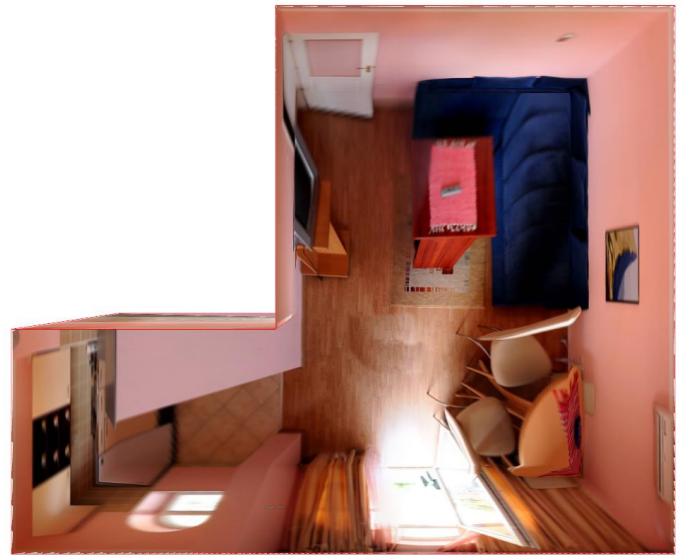
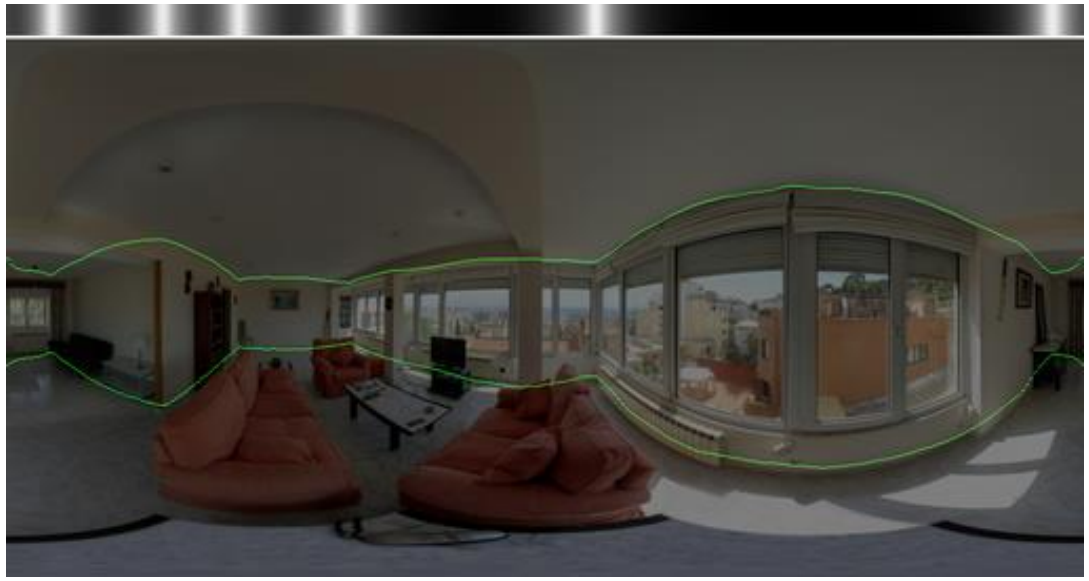
2 ResNet - resnet50

Img _name	Estimating layout	결과
Pano_01		
Pano_07		

2 ResNet - resnet50

Img _name	Estimating layout	결과
Pano_13		
Pano_15		

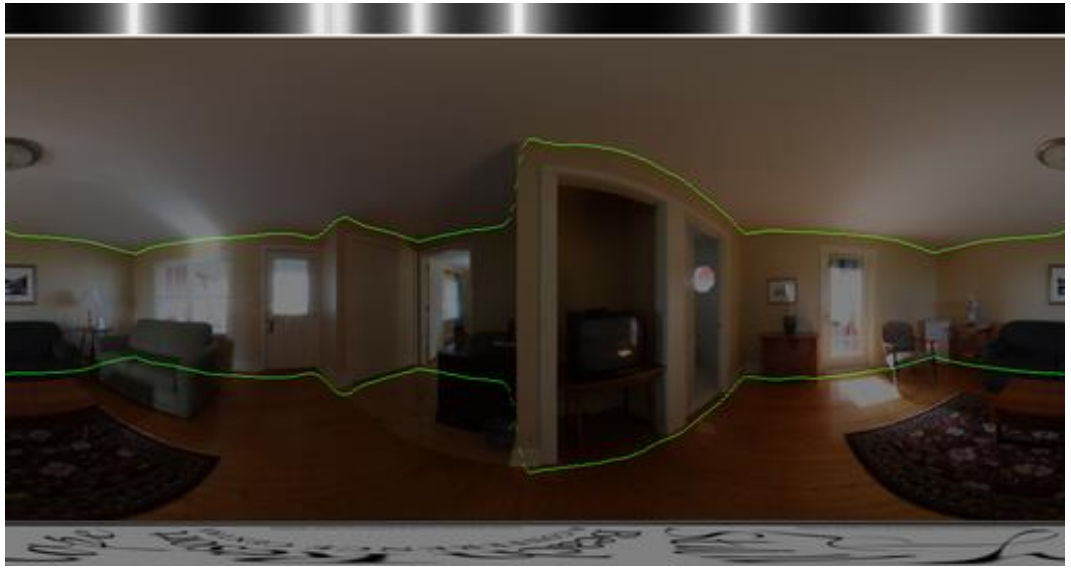
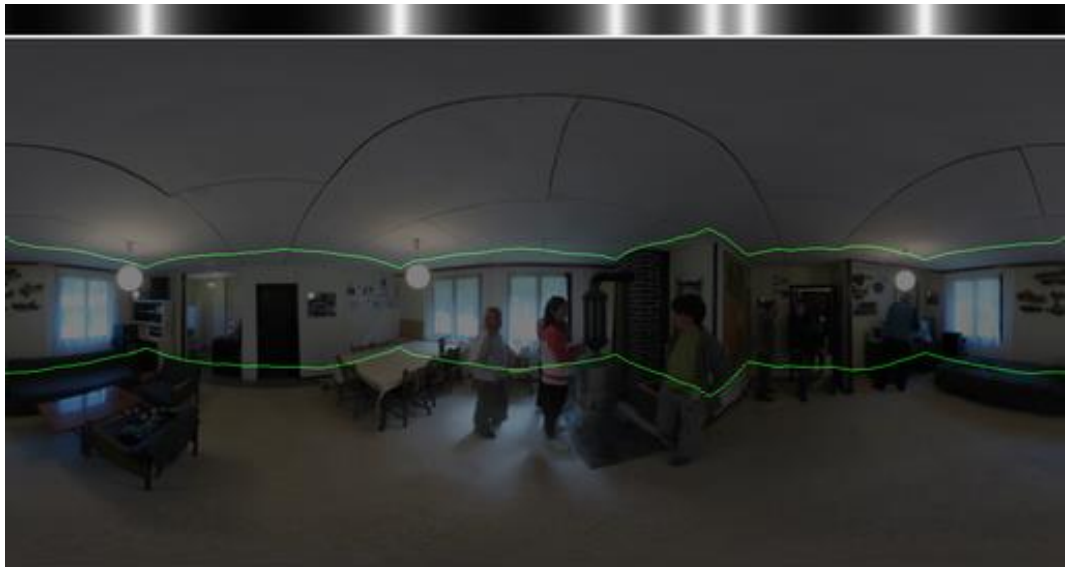
2 ResNet - resnet50

Img _name	Estimating layout	결과
Pano_18		
Pano_20		

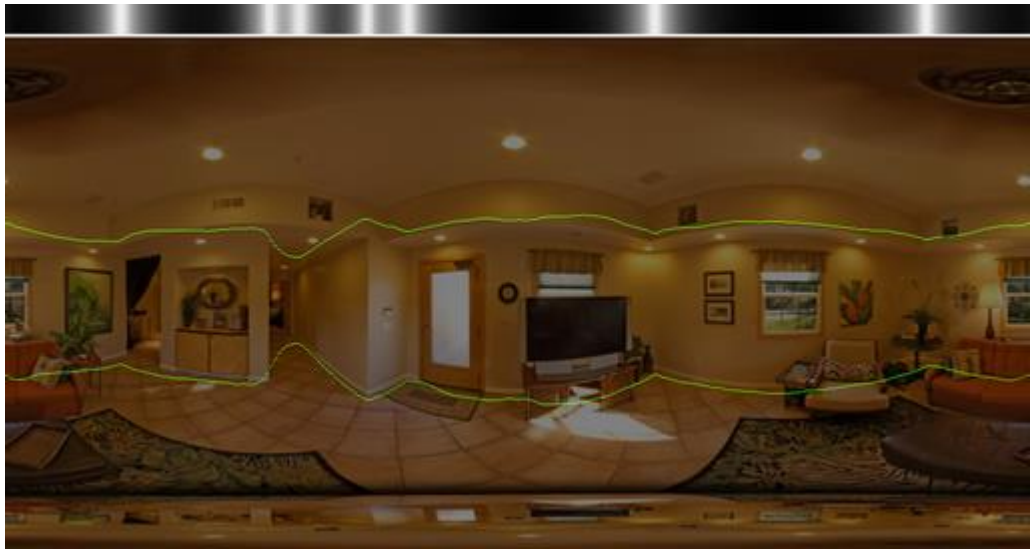
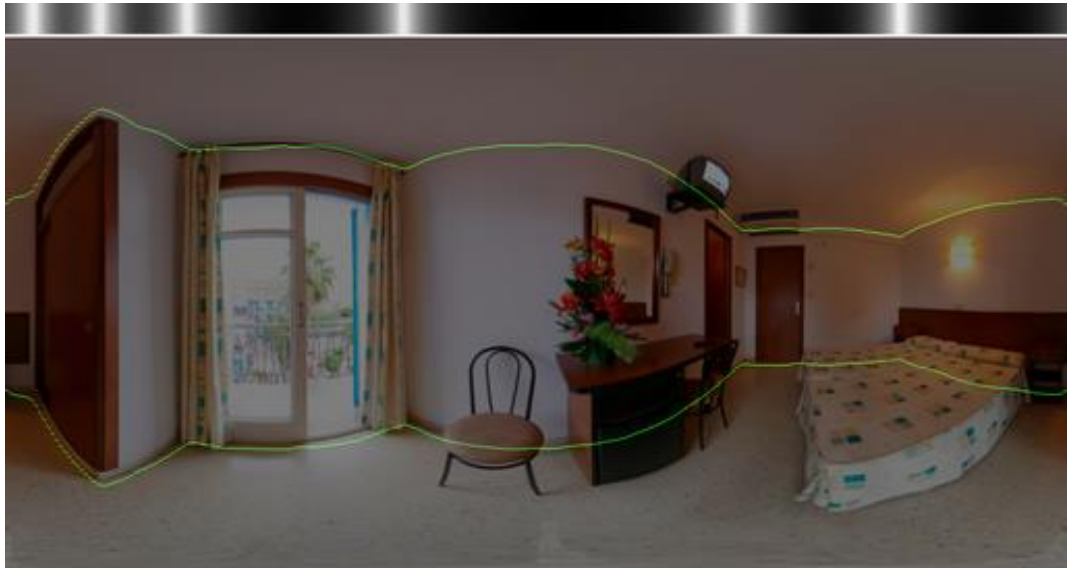
2 ResNet

- ResNet50
- **ResNet101**
- ResNet34
- ResNet152
- 분석

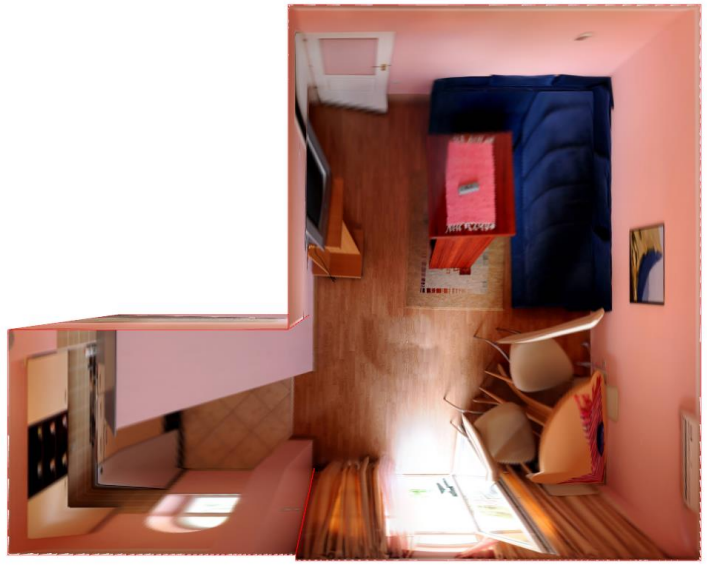
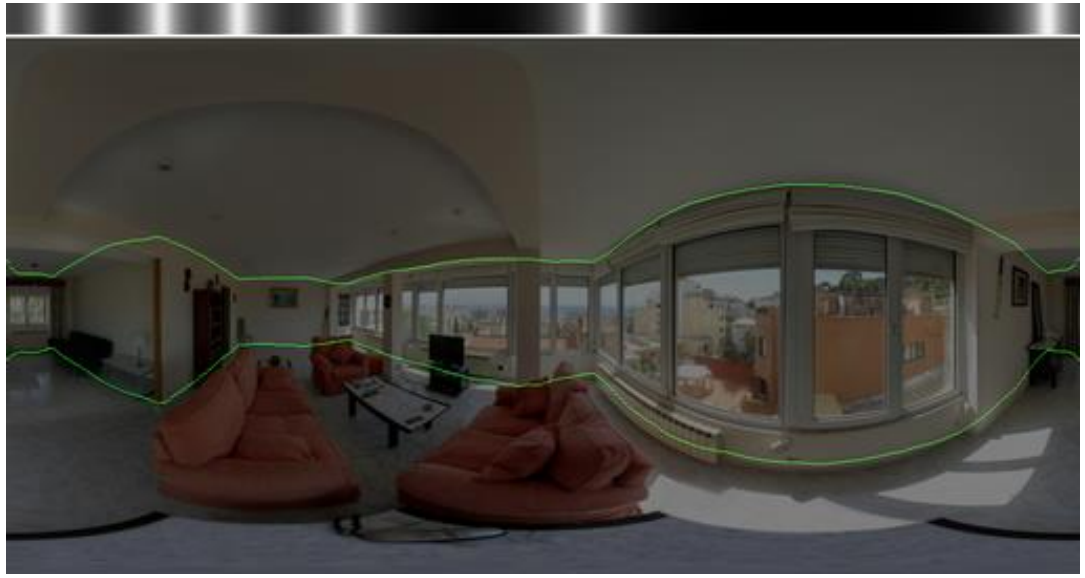
2 ResNet - resnet101

Img _name	Estimating layout	결과
Pano_01		
Pano_07		

2 ResNet - resnet101

Img _name	Estimating layout	결과
Pano_13		
Pano_15		

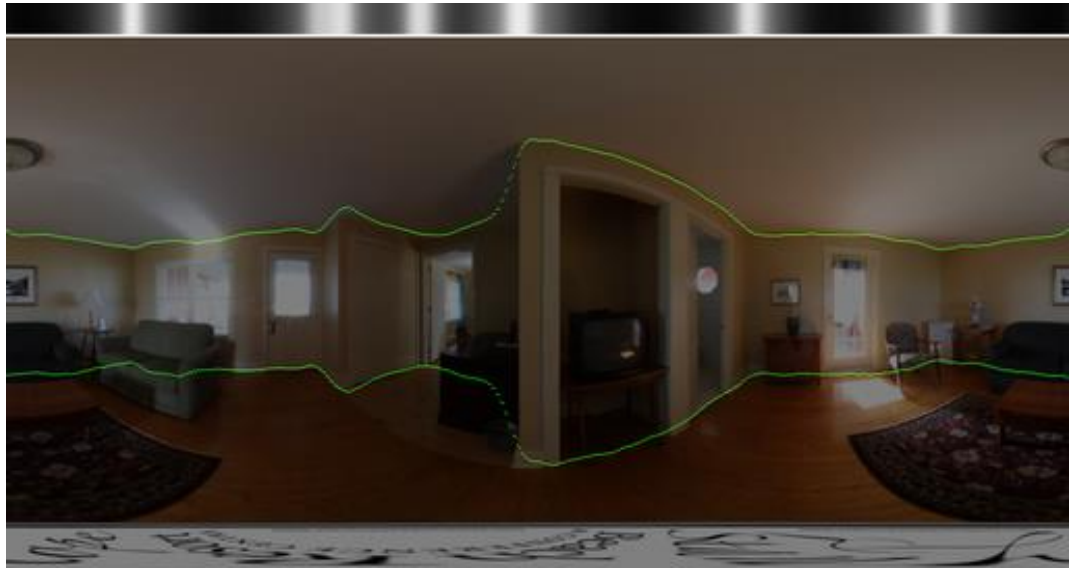
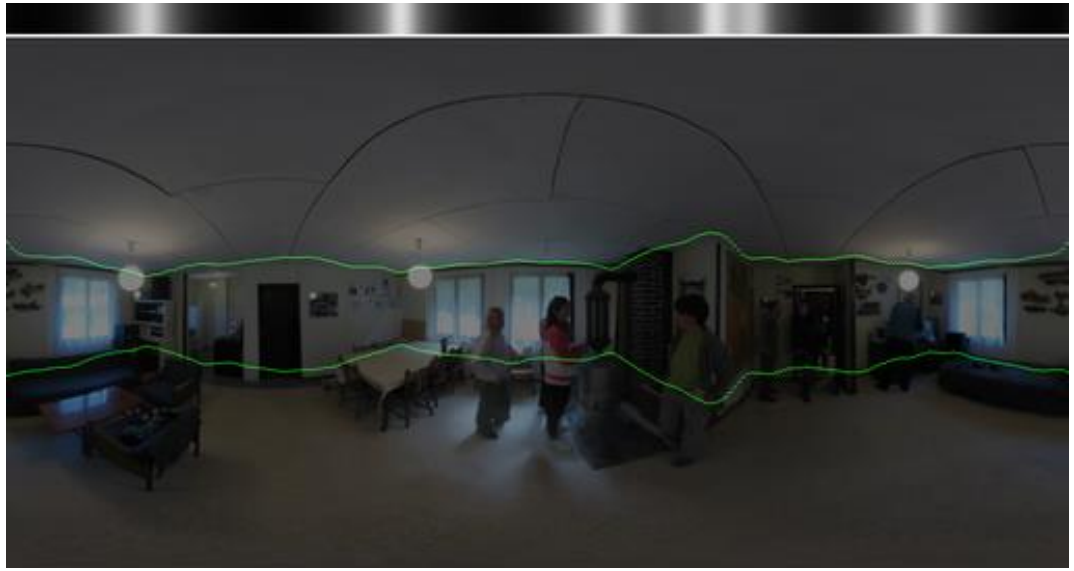
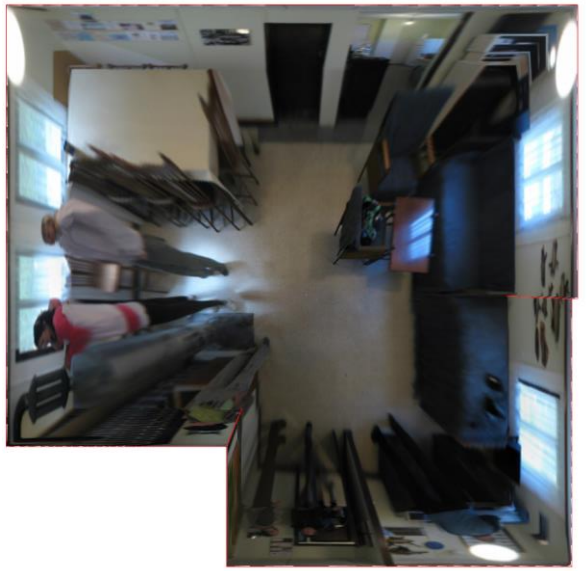
2 ResNet - resnet101

Img _name	Estimating layout	결과
Pano_18		
Pano_20		

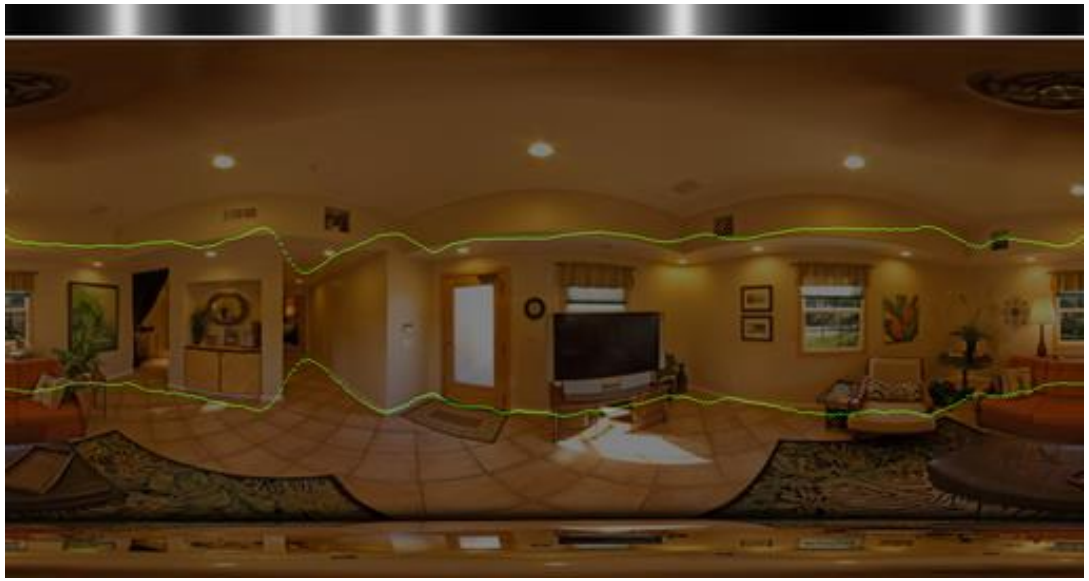
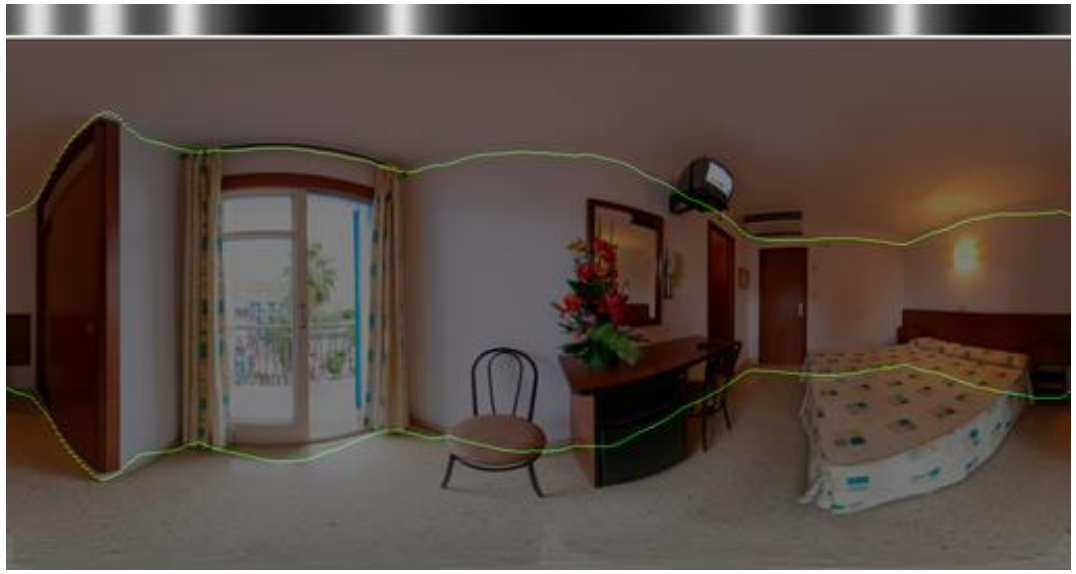
2 ResNet

- ResNet50
- ResNet101
- **ResNet34**
- ResNet152
- 분석

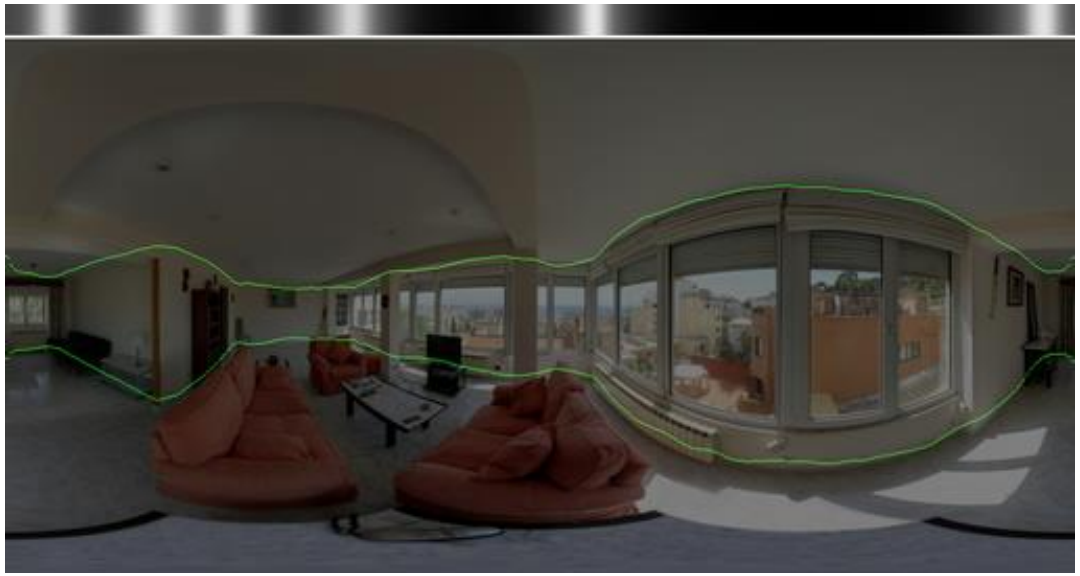
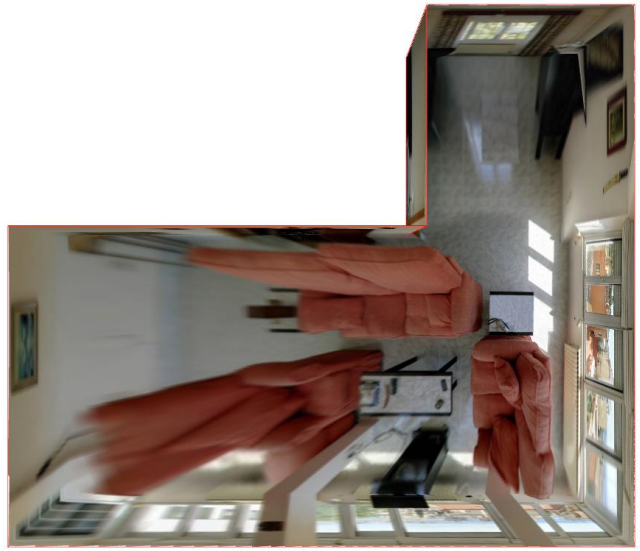
2 ResNet - resnet34

Img _name	Estimating layout	결과
Pano_01		
Pano_07		

2 ResNet - resnet34

Img _name	Estimating layout	결과
Pano_13		
Pano_15		

2 ResNet - resnet34

Img _name	Estimating layout	결과
Pano_18		
Pano_20		

2 ResNet

- ResNet50
- ResNet101
- ResNet34
- **ResNet152**

- 분석

2 ResNet - resnet152

- 다음의 에러가 발생하여 진행하지 못함. 메모리 문제인가..?😢

```
C:\Users\user\PycharmProjects\HorizonNet-resnet152\venv\lib\site-packages\torch\nn\functional.py:718: UserWarning: Named tensors and all their associated APIs are an experimental feature and subject to change. Please do not use them fo
r anything important until they are released as stable. (Triggered internally at  ..\c10/core/TensorImpl.h:1156.)
  return torch.max_pool2d(input, kernel_size, stride, padding, dilation, ceil_mode)

Epoch:   0%|                                                                                                                                                                                                | 0/500 [00:01<?, ?ep/s]
Traceback (most recent call last):
  File "train.py", line 190, in <module>
    losses = feed_forward(net, x, y_bon, y_cor)
  File "train.py", line 26, in feed_forward
    y_bon_, y_cor_ = net(x)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\venv\lib\site-packages\torch\nn\modules\module.py", line 1051, in _call_impl
    return forward_call(*input, **kwargs)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\HorizonNet\model.py", line 242, in forward
    feature = self.reduce_height_module(conv_list, x.shape[3]//self.step_cols)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\venv\lib\site-packages\torch\nn\modules\module.py", line 1051, in _call_impl
    return forward_call(*input, **kwargs)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\HorizonNet\model.py", line 164, in forward
    feature = torch.cat([
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\HorizonNet\model.py", line 165, in <listcomp>
    f(x, out_w).reshape(bs, -1, out_w)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\venv\lib\site-packages\torch\nn\modules\module.py", line 1051, in _call_impl
    return forward_call(*input, **kwargs)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\HorizonNet\model.py", line 138, in forward
    x = self.layer(x)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\venv\lib\site-packages\torch\nn\modules\module.py", line 1051, in _call_impl
    return forward_call(*input, **kwargs)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\venv\lib\site-packages\torch\nn\modules\container.py", line 139, in forward
    input = module(input)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\venv\lib\site-packages\torch\nn\modules\module.py", line 1051, in _call_impl
    return forward_call(*input, **kwargs)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\HorizonNet\model.py", line 124, in forward
    return self.layers(x)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\venv\lib\site-packages\torch\nn\modules\module.py", line 1051, in _call_impl
    return forward_call(*input, **kwargs)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\venv\lib\site-packages\torch\nn\modules\container.py", line 139, in forward
    input = module(input)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\venv\lib\site-packages\torch\nn\modules\module.py", line 1051, in _call_impl
    return forward_call(*input, **kwargs)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\HorizonNet\model.py", line 31, in forward
    return lr_pad(x, self.padding)
  File "C:\Users\user\PycharmProjects\HorizonNet-resnet152\HorizonNet\model.py", line 21, in lr_pad
    return torch.cat([x[..., -padding:], x, x[..., :padding]], dim=3)
RuntimeError: CUDA out of memory. Tried to allocate 34.00 MiB (GPU 0; 12.00 GiB total capacity; 2.33 GiB already allocated; 27.40 MiB free; 2.41 GiB reserved in total by PyTorch)
```

2 ResNet

- ResNet50
- ResNet101
- ResNet34
- ResNet152

- **분석**

2 ResNet - 분석

설정

- Batch_size = 1, Epoch = 300으로 고정.
- Epoch를 늘리고 싶었으나, 시간이 너무 오래 걸려 하지 못함.
- resnet 50과 resnet34의 경우, 총 65개의 이미지를 학습시킴. 그러나 시간이 너무 오래 걸려 그 외 나머지는 총 20개의 이미지만 학습시킴.

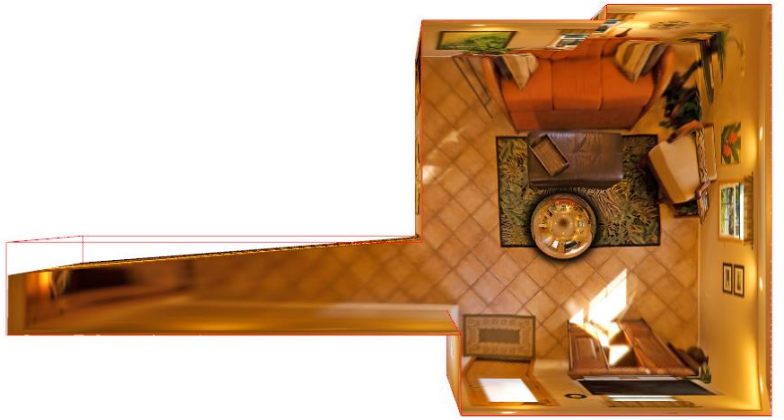
결과 분석

- ResNet101이 대체적으로 성능이 떨어져 보임.
- ResNet50과 ResNet34는 눈으로 성능을 비교하기 어려움.
- ResNet 종류마다 사진의 가로 세로 비율이 다름.
- ResNet152의 경우, 코드가 실행되지 않음.

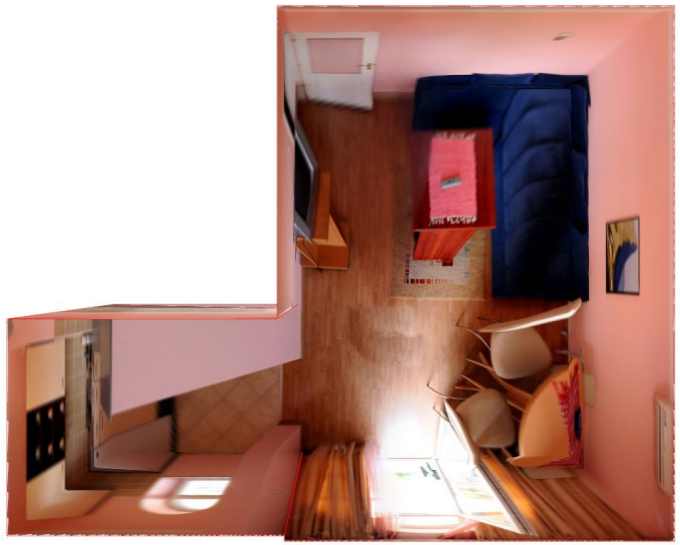
2 ResNet - resnet34

Img _name	resnet50	resnet101	resnet34
Pano_01			
Pano_07			

2 ResNet - resnet34

Img _name	resnet50	resnet101	resnet34
Pano_13			
Pano_15			

2 ResNet - resnet34

Img _name	resnet50	resnet101	resnet34
Pano_18			
Pano_20			

THANK YOU

The End