



Training Model 만들기

21.08.19

길다영

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



Index

1.

Estimating layout
with HorizonNet
실행 결과 비교

2.

Train img 밝기
조절하여 결과
도출

3.

Resnet50과
st3d.pth 비교

4.

Mydataset으로
training model
만들어보기

5.

6.

Mydataset에서 이미
지 양 끝의 높이 일치
여부에 따른 결과 비교

+ 1.

Estimating layout with

HorizonNet

실행 결과 비교

1. Estimating layout with HorizonNet 실행 결과 비교

Finetune_general/visual	Panos2d3d.pth	St3d.pth	resnet50
			
			
			

1. Estimating layout with HorizonNet 실행 결과 비교

Pretrained Models

- [resnet50_rnn__panos2d3d.pth](#)
 - Trained on PanoContext/Stanford2d3d 817 pano images.
 - Trained for 300 epoch
- [resnet50_rnn__st3d.pth](#)
 - Trained on Structured3D 18362 pano images with setting of original furniture and lighting.
 - Trained for 50 epoch.
 - Select 50th epoch according to loss function on validation set.

딱 보기에 finetune의 밝기가 다른 것들에 비해 더 밝고 resnet50과 가장 유사함.

Finetune과 resnet50의 결과를 바탕으로 layoutview를 비교해보고자 함.

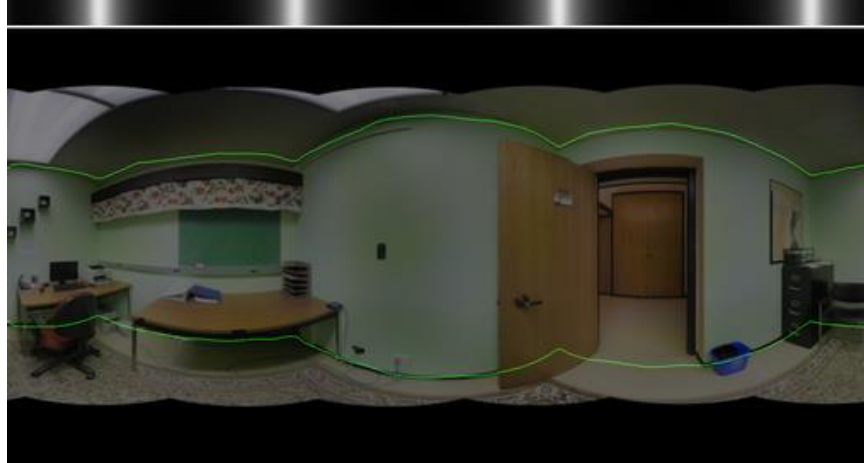
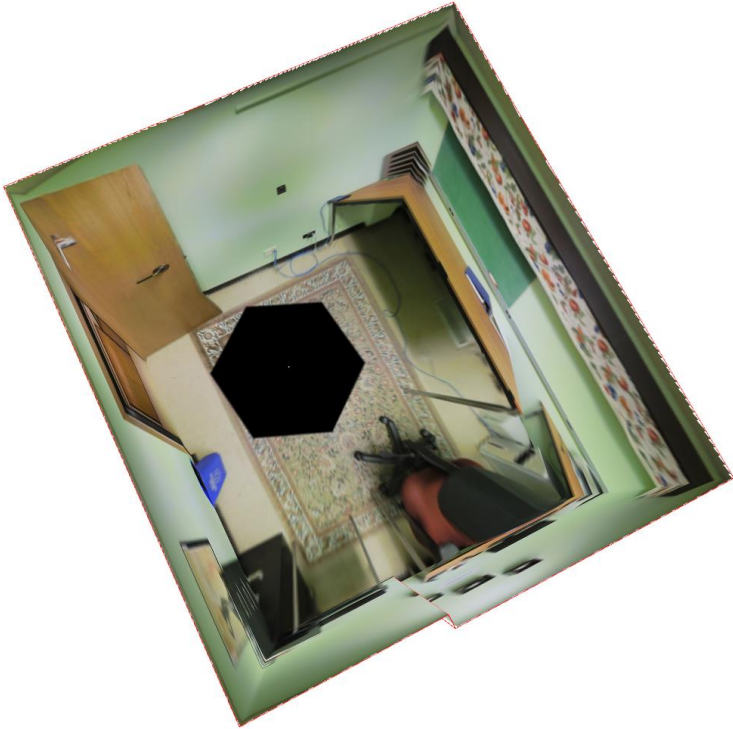
+ 2.

Train img

밝기 조절하여 결과 도출

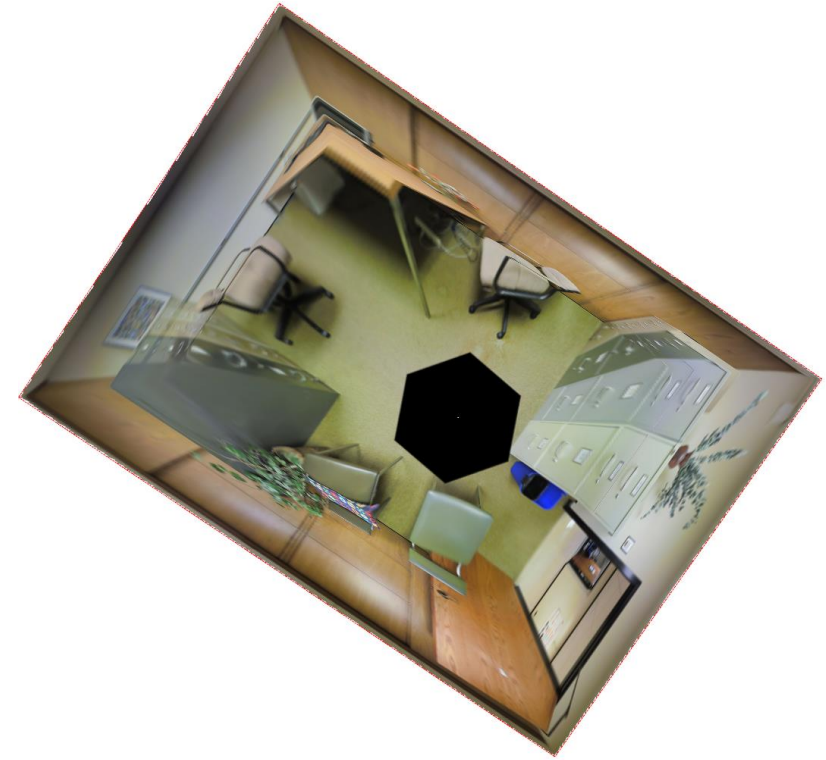
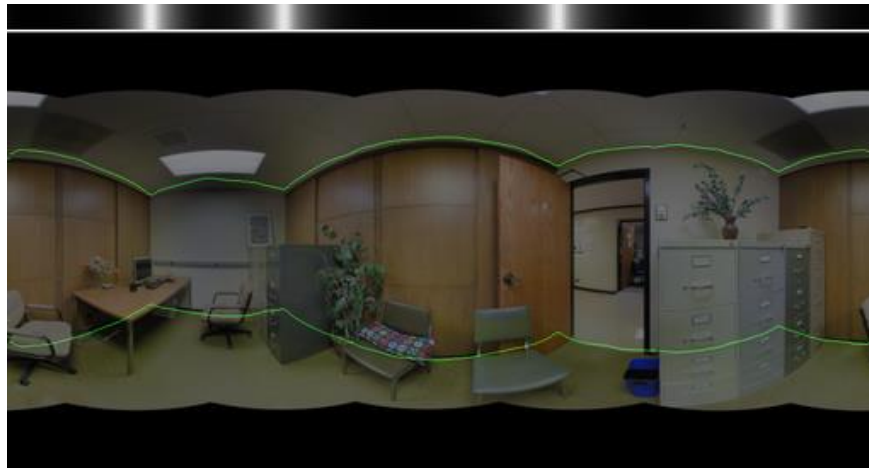
2. Train img 밝기 조절하여 결과 도출

- 밝기 조절 전과 별다른 차이가 없었음.



2. Train img 밝기 조절하여 결과 도출

- 밝기 조절 전과 별다른 차이가 없었음.



+ 3.

Resnet50과 st3d.pth 비교

3. Resnet50과 st3d.pth 비교

St3d.pth	Resnet50 (epoch300, train img24개)	resnet50(epoch500, train img60개)
		
		

3. Resnet50과 st3d.pth 비교

St3d.pth	Resnet50 (epoch300, train img24개)	resnet50(epoch500, train img60개)
		
		

3. Resnet50과 st3d.pth 비교

St3d.pth	Resnet50 (epoch300, train img24개)	resnet50(epoch500, train img60개)
		
		





3. Resnet50과 st3d.pth 비교

St3d.pth	Resnet50 (epoch300, train img24개)	resnet50(epoch500, train img60개)
		
		



3. Resnet50과 st3d.pth 비교

St3d.pth	Resnet50 (epoch300, train img24개)	resnet50(epoch500, train img60개)
		

3. Resnet50과 st3d.pth 비교

St3d.pth	Resnet50 (epoch300, train img24개)	resnet50(epoch500, train img60개)
		
		

3. Resnet50과 st3d.pth 비교

St3d.pth	Resnet50 (epoch300, train img24개)	resnet50(epoch500, train img60개)
		
		

3. Resnet50과 st3d.pth 비교

St3d.pth	Resnet50 (epoch300, train img24개)	resnet50(epoch500, train img60개)
		
		



3. Resnet50과 st3d.pth 비교

St3d.pth	Resnet50 (epoch300, train img24개)	resnet50(epoch500, train img60개)
		
		





3. Resnet50과 st3d.pth 비교

St3d.pth	Resnet50 (epoch300, train img24개)	resnet50(epoch500, train img60개)






3. Resnet50과 st3d.pth 비교

St3d.pth	resnet50



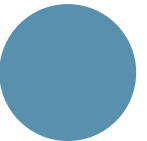
3. Resnet50과 st3d.pth 비교



눈으로 확인했을 때 이미지에 따라 각각 오차가 있어, 무엇이 더 좋다고 판단하기 어려움.



Resnet50, epoch=300, training data 개수 = 24장



Resnet50, epoch=500, training data 개수 = 80장
녹색선을 분명히 하고, 오류를 줄이기 위해 시도해봄.

+ 4.

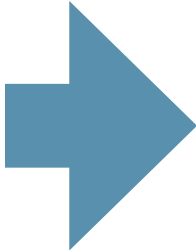
Mydataset으로

training model 만들어 보기

4. Mydataset으로 training model 만들어보기



pano_01 - Windows 메모장	
파일(F)	편집(E) 서식(O) 보기(V) 도움말(H)
629	435
629	937
899	443
899	943
941	387
941	987
1215	499
1215	929
1765	491
1765	947
2421	435
2421	937



*pano_01 - Windows 메모장	
파일(F)	편집(E) 서식(O) 보기(V) 도움말(H)
215	150
215	320
306	150
306	322
321	132
321	334
416	170
416	311
602	168
602	316
826	123
826	342

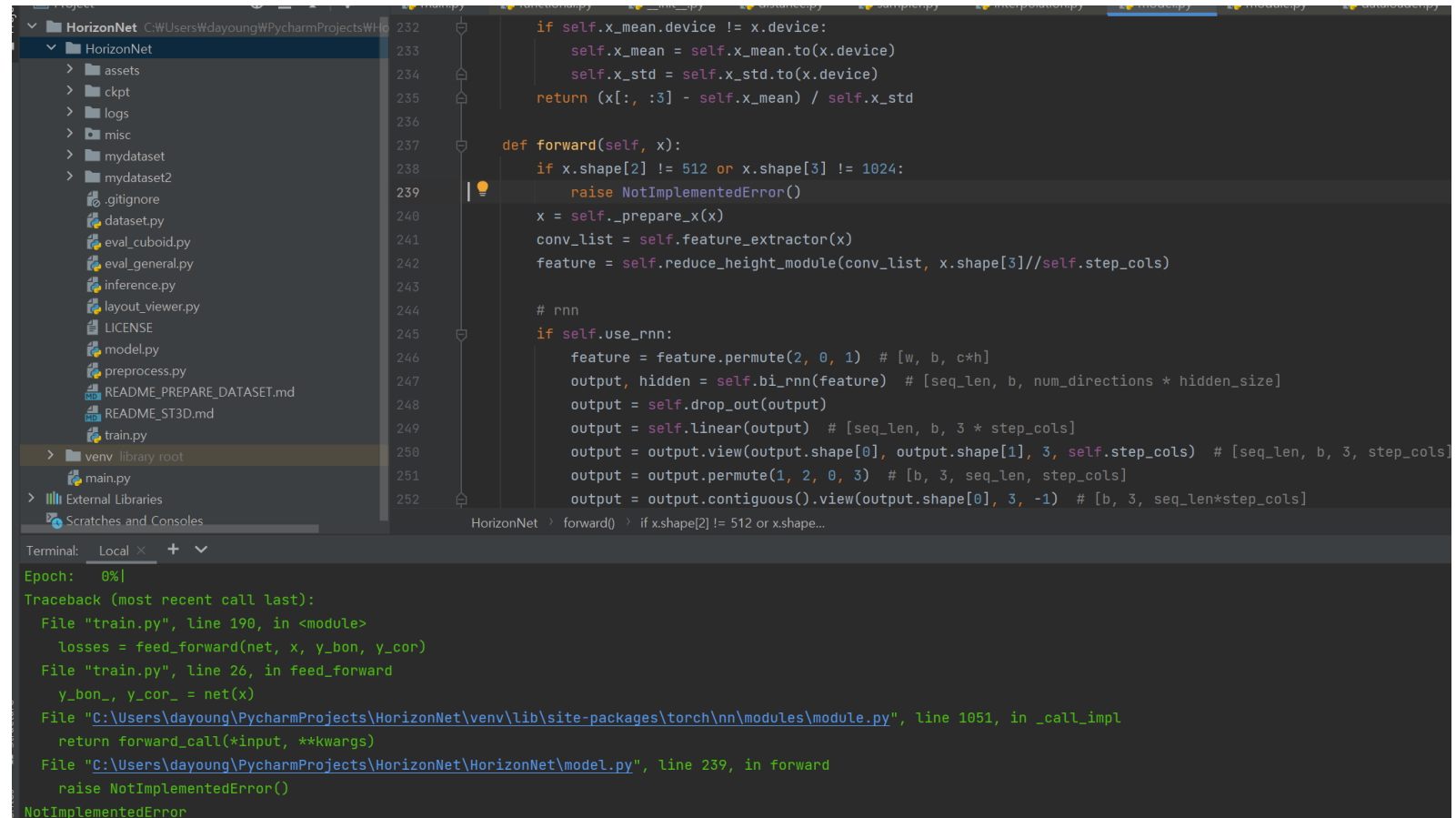
4. Mydataset으로 training model 만들어보기

<https://www.pinterest.co.kr/pin/17240411063757029/>

파노라마 데이터 셋을 구해 직접 모서리를 찍어 봤으나, preprocess까지만 되고 train에서 오류가 남 🤖

1024 x 512 범위 안에 모서리를 찍어야 하는데, 이 범위를 벗어나서 그런 것 같음.

🤖👉 Preprocess 된 이미지에 모서리를 다시 찍어보도록 하겠음



```
HorizonNet
├── assets
├── ckpt
├── logs
├── misc
├── mydataset
├── mydataset2
├── .gitignore
├── dataset.py
├── eval_cuboid.py
├── eval_general.py
├── inference.py
├── layout_viewer.py
├── LICENSE
├── model.py
├── preprocess.py
├── README_PREPARE_DATASET.md
├── README_ST3D.md
├── train.py
├── venv
├── library root
├── main.py
├── External Libraries
└── Scratches and Consoles

232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252

if self.x_mean.device != x.device:
    self.x_mean = self.x_mean.to(x.device)
    self.x_std = self.x_std.to(x.device)
    return (x[:, :3] - self.x_mean) / self.x_std

def forward(self, x):
    if x.shape[2] != 512 or x.shape[3] != 1024:
        raise NotImplementedError()
    x = self._prepare_x(x)
    conv_list = self.feature_extractor(x)
    feature = self.reduce_height_module(conv_list, x.shape[3]//self.step_cols)

    # rnn
    if self.use_rnn:
        feature = feature.permute(2, 0, 1) # [w, b, c*h]
        output, hidden = self.bi_rnn(feature) # [seq_len, b, num_directions * hidden_size]
        output = self.drop_out(output)
        output = self.linear(output) # [seq_len, b, 3 * step_cols]
        output = output.view(output.shape[0], output.shape[1], 3, self.step_cols) # [seq_len, b, 3, step_cols]
        output = output.permute(1, 2, 0, 3) # [b, 3, seq_len, step_cols]
        output = output.contiguous().view(output.shape[0], 3, -1) # [b, 3, seq_len*step_cols]

HorizonNet > forward() > if x.shape[2] != 512 or x.shape...

Terminal: Local x + v
Epoch: 0%|
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\dayoung\PycharmProjects\HorizonNet\venv\lib\site-packages\torch\nn\modules\module.py", line 1051, in _call_impl
    return forward_call(*input, **kwargs)
  File "C:\Users\dayoung\PycharmProjects\HorizonNet\HorizonNet\model.py", line 239, in forward
    raise NotImplementedError()
NotImplementedError
```



Thank You.

Enter the title

ENTER THE CONTENTS

I believe that someone like you. I was broken my heart. But now, I am standing again.



Something

I believe that someone like you.
Enter something here.

Something

I believe that someone like you.
Enter something here.

Something

I believe that someone like you.
Enter something here.

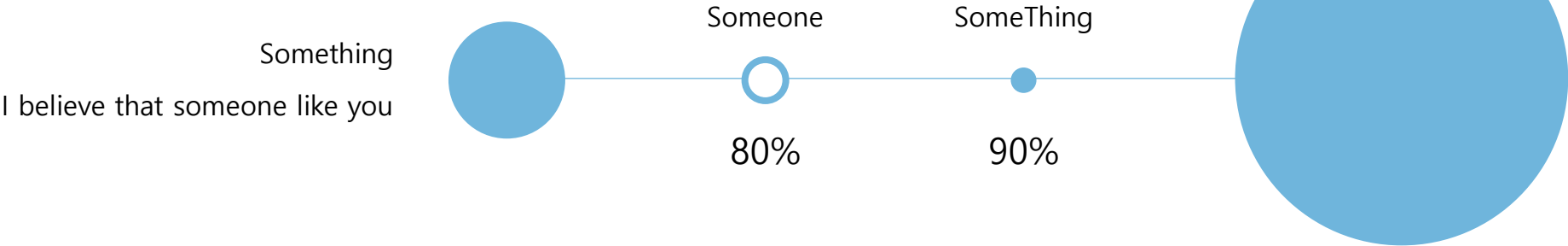


Enter the title

ENTER THE CONTENTS

I believe that someone like you

ENTER



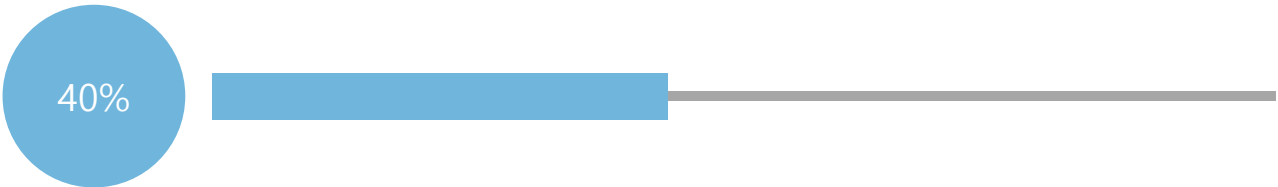


Enter the title

ENTER
CONTENTS



ENTER
CONTENTS



ENTER
CONTENTS



I want you to use this template for free and to remember slug and CREBUGS for me.



Enter the title

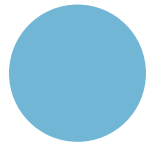
ENTER THE CONTENTS

I believe that someone like you. I was broken my heart. But now, I am standing again.



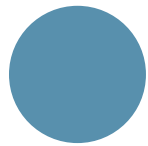
Someone

I believe that someone like you.
Enter something here.



Someone

I believe that someone like you.
Enter something here.



Someone

I believe that someone like you.
Enter something here.

