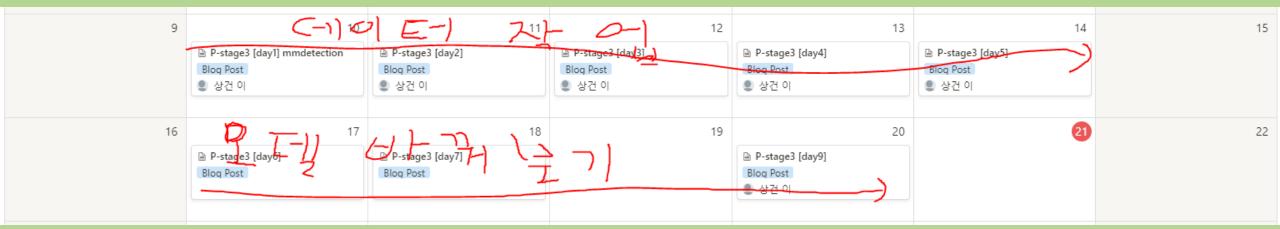
P-stage3 object detection 발표용

이상건



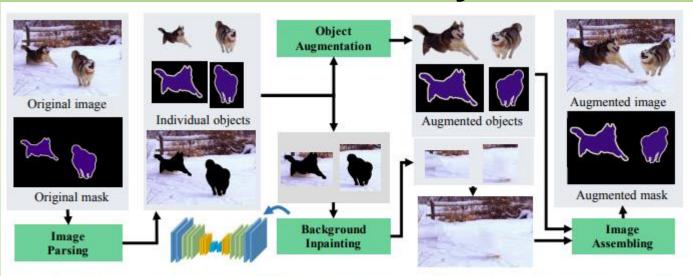
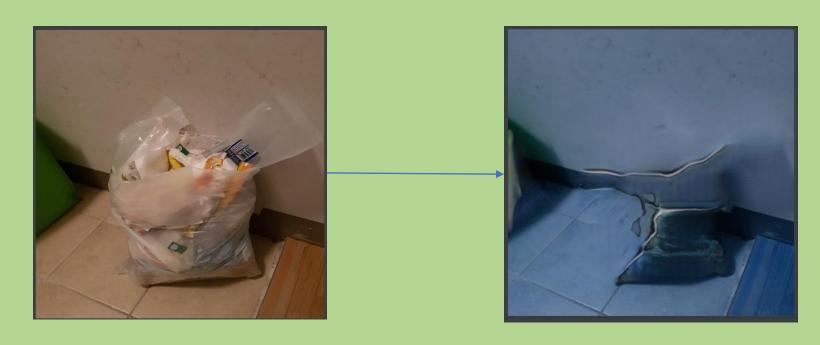
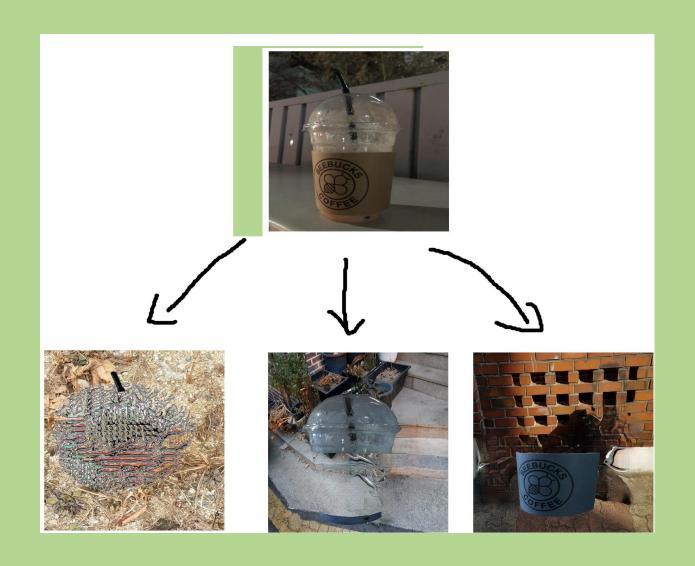


Fig. 3. Overall flow of ObjectAug. ObjectAug includes four modules: image parsing, object augmentation, background inpainting, and image assembling. First, image parsing decouples the image and extracts multiple objects in the image. Next, object augmentation adopts various data augmentation methods to augment the objects. At the same time, background inpainting restores the object-removed background to deal with the pixel artifacts. Finally, the augmented objects with the restored background and their masks are assembled to obtain the augmented image and mask.

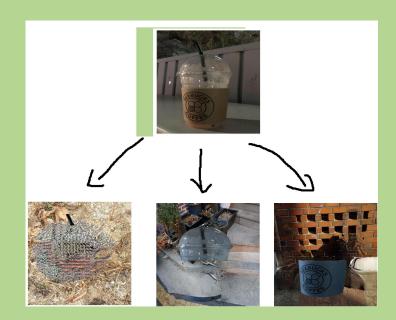
- 저번 대회에서 object augmentation이 잘 되었다고 해서 따라하고 싶었다.
- https://arxiv.org/pdf/2102.00221.pdf
- 라이브러리 문제와 실력으로 완벽하게 따라하진 못하고 배경따로, 각 물체 따로 분리해서 data, json을 만듬.



• 외부 pretrained model 을 사용 (https://github.com/Atlas200dk/sample-imageinpainting-HiFill)







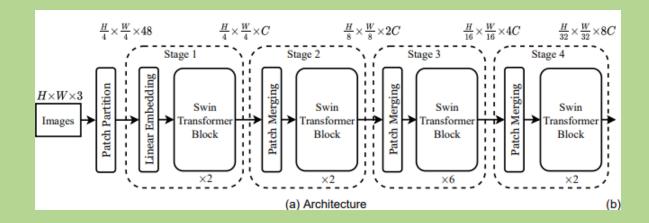
- 훈련데이터와 섞어서 사용햇을 때 성능변화
- Public mAP50: 0.3377 -> 0.2912
- val mAP50: 0.27 -> 0.24
- val mAP_s: 0.04 -> 0.035
- 로 모두 감소! 괜히 했다!

모델 작업 (day 6~9)

Rank	Model	box † AP	AP50	AP75	APS	АРМ	APL	Extra Training Data	Paper	Code	Result	Year
1	Swin-L (HTC++, multi scale)	58.7						×	Swin Transformer: Hierarchical Vision Transformer using Shifted Windows	O	Ð	2021
2	Swin-L (HTC++, single scale)	57.7						×	Swin Transformer: Hierarchical Vision Transformer using Shifted Windows	n	Ð	2021
									C 1 C D 1 .			

- 팀원분이 사용한 sota를 찍은 swin transformer를 사용 -> PB 0.5304로 제일 높게 나옴
- 얘를 쓰자.

모델 작업 (day 6~9)



```
model = dict(
    backbone=dict(
    embed_dim=96,
    depths=[2, 2, 6, 2],
    num_heads=[3, 6, 12, 24],
    window_size=7,
    ape=False,
    drop_path_rate=0.1,
    patch_norm=True,
    use_checkpoint=False
```

```
model = dict(
    backbone=dict(
    embed_dim=96,
    depths=[2, 2, 6, 4],
    num_heads=[3, 6, 12, 24],
    window_size=7,
    ape=False,|
    drop_path_rate=0.1,
    patch_norm=True,
    use_checkpoint=False
),
```

- swin tiny model 에서 depths를
- [2,2,6,2] -> [4,4,4,6,2]로 바꿔봄
- PB 0.01 떨어짐
- [2,2,6,2] -> [2,2,6,4]로 바꿔봄
- PB 0.03 증가했지만 다른분은 감 소함
- 그냥 있는것만 잘 쓰자. 아마 새로 생긴 빈 weight가 노이즈 역할을 하는 듯.

모델 작업 (day 6~9)



- https://github.com/ZFTurbo/Weighted-Boxes-Fusion 라이브러리와 팀원의 도움으로 각 자 kfold 훈련한 모델을 앙상블.
- 잘 나온 결과를 가지고 또 앙상블, 더 잘나오면 또 앙상블, 또 앙상블, 또 앙상블..
- 각 앙상블 결과마다 모델 예측을 보며 판단.



- 토론게시판에 나온 논문 그대로 구현한걸 중간까지만 학습하고 앙상블하여
- 최종 PB 0.5878, LB 0.4771

기타 하고싶은 말들...

- Incompatible version of pycocotools is installed, ~~~ is not in the ~~~ registry. ...등등의 여러 에러들.
- 편리하고 강력한 mmdetection 라이브러리이지만 에러 버그 덩어리...
- 디버그에 많은 시간을 썼다.
- 다음엔 중요한 부분은 한줄 한줄 보면서 판단하자.