Wine_mini_project_report 김영관

1. VOC 데이터(XML 포맷) Yolo 라벨 형식으로 변환 및 Train, Val, Test 로 분리

```
# VOC -> YOLO format
def xml_to_yolo_bbox(bbox, w, h):
    # xmin, ymin, xmax, ymax
    x_center = ((bbox[2] + bbox[0]) / 2) / w
    y_center = ((bbox[3] + bbox[1]) / 2) / h
    width = (bbox[2] - bbox[0]) / w
    height = (bbox[3] - bbox[1]) / h
    return [x_center, y_center, width, height]



for path in xml_path:
    name_temp = path.split('\\')[-1].split('.xml')[0]
    image_temp = name_temp + '.jpg'
    image_path = os.path.join('.\\dataset\\wine labels_voc_dataset', image_temp)
    image = cv2.imread(image_path)
    w = image.shape[1]
    h = image.shape[0]
    tree = ET.parse(path)
    root = tree.getroot()
    boxes = []
    labels = []
```

(VOC 에서 YOLO 형식으로 바꾸는 함수)

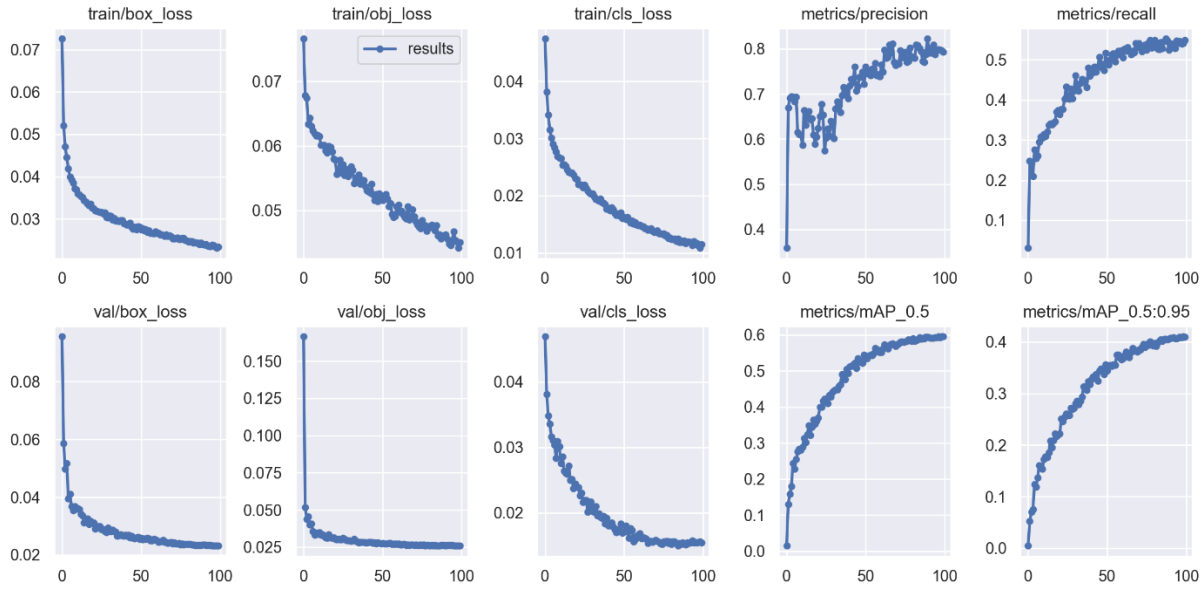
```
# data split

y_temp = []
jpg_path = []
for path in xml_path:
    y_data = path.split('\\')[-1].split('.xml')[0] + '.txt'
    y_temp.append(os.path.join('.\\dataset\\temp\\', y_data))
    jpg = path.split('\\')[-1].replace('.xml', '.jpg')
    jpg_path.append(os.path.join('.\\dataset\\wine labels_voc_dataset\\', jpg))

x_tr_temp, x_tp_temp, y_tr_temp, y_tp_temp = train_test_split(jpg_path, y_temp, train_size=0.8,
                                                              random_state=2324)
x_val_temp, x_te_temp, y_val_temp, y_te_temp = train_test_split(x_tp_temp, y_tp_temp, train_size=0.5,
                                                                random_state=2324)
for path, label in zip(x_tr_temp, y_tr_temp):
    img = cv2.imread(path)
    cv2.imwrite(f'.\\dataset\\train\\images\\{os.path.basename(path)}', img)
    shutil.move(label, f'.\\dataset\\train\\labels\\{os.path.basename(label)}')

for path, label in zip(x_val_temp, y_val_temp):
    img = cv2.imread(path)
    cv2.imwrite(f'.\\dataset\\valid\\images\\{os.path.basename(path)}', img)
    shutil.move(label, f'.\\dataset\\valid\\labels\\{os.path.basename(label)}')
xml_path
```

(바꾼 후 , Train, Validation, Test 데이터로 분리)

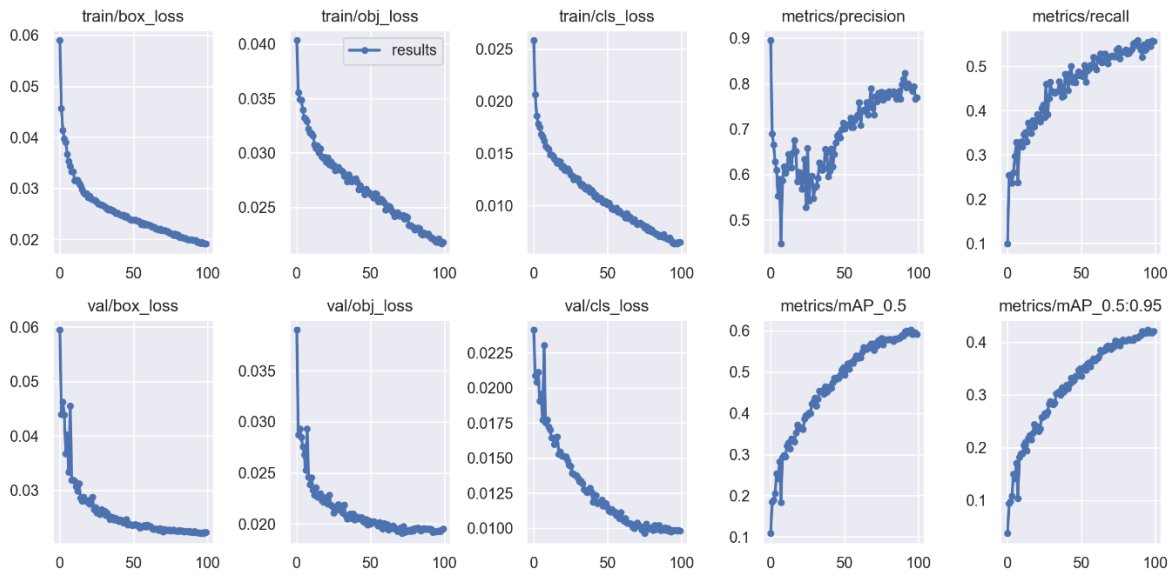
2. Yolo 모델을 활용하여 학습 진행 결과 제출

	YOLOV5S	YOLOV5M
Epoch	100	100
Batch_size	48	32
Learning_Rate	0.001	0.002
Optimizer	AdamW	AdamW
Scale_aug	0.6	0.9
Shear_aug	0.1	0.4
Mixup_aug	0.15	0.0

5s 모델 학습 후, 학습진행률이 더딘것같아 학습률을 높였고, 작고 기울어진 문자를 잘 잡아내지 못하여 Scale 과 Shear aug 를 높여주었다.



(yolov5s 의 학습 결과)



(yolov5m 의 학습 결과)

5m 모델이 5s 에 비해 파라미터 수가 많아서 인지 mean average precision 이 조금이라도 더 높은 모습을 보였다. 하지만 두 모델 모두 mAP 가 0.4 초반대로 낮은 정확도를 보여준다.

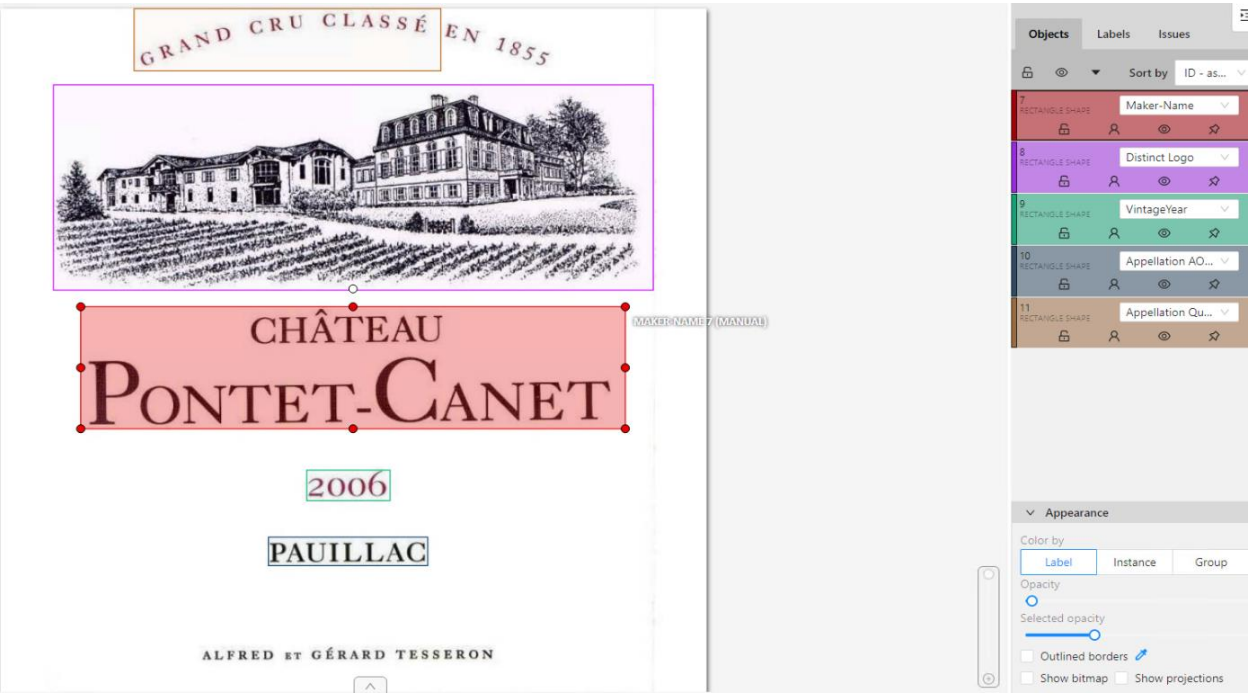
(왼쪽이 yolov5s 모델을 적용시킨 이미지, 오른쪽이 yolov5m 모델을 적용시킨 이미지)

3. 학습한 모델 인퍼런스 코드 작성 및 CVAT 양식 XML 코드 작성 하여 XML 추출

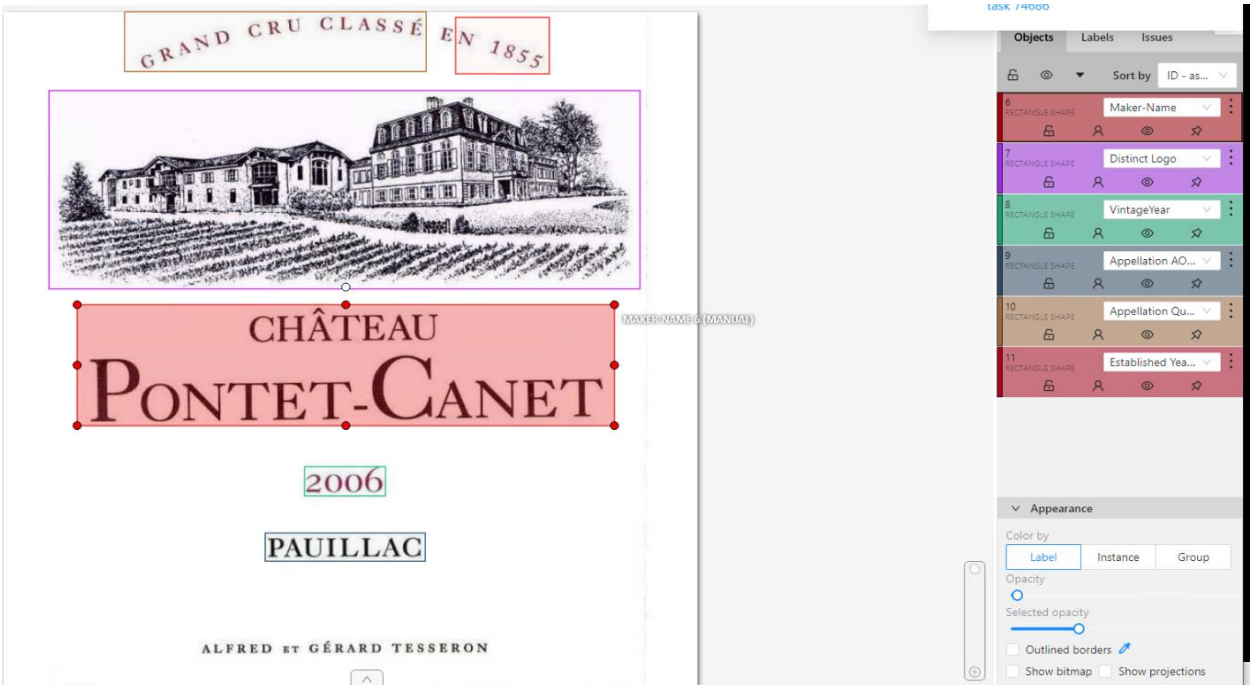
```python
result = model(img, size=640)
# print(result.print())
# print(result.xyxy)  # bbox


bbox = result.xyxy[0]
for bbox_info in bbox:
    # print(bbox_info)
    x1 = str(round(bbox_info[0].item(),6))
    y1 = str(round(bbox_info[1].item(),6))
    x2 = str(round(bbox_info[2].item(),6))
    y2 = str(round(bbox_info[3].item(),6))
    sc = round(bbox_info[4].item(),6)
    label_number = bbox_info[5].item()
    label = label_dict[int(label_number)]
    # print(x1, y1, x2, y2, sc, label_number)

    if sc >= model.conf:
        ET.SubElement(xml_frame, 'box', label=label, occluded='0', source='manual',
                      xtl=x1, ytl=y1, xbr=x2, ybr=y2, z_order='0')
```

(테스트 이미지에 모델을 적용하여 얻은 BoundingBox 좌표를 xml 형식으로 저장하는 코드)

4. CVAT Test 파일 업로드 하여 Auto 라벨 체크



(yolov5s 모델)


GRAND CRU CLASSÉ EN 1855
CHÂTEAU
PONTET-CANET
2006
PAUILLAC
ALFRED ET GÉRARD TESSERON
Objects
Labels
Issues
Sort by
Maker-Name
Distinct Logo
VintageYear
Appellation AO...
Appellation Qu...
Established Yea...
Appearance
Color by
Label
Instance
Group
Opacity
Selected opacity
Outlined borders
Show bitmap
Show projections

(yolov5m 모델)