

01. Json file 을 읽고 Cat & Dog 바운딩 박스하기

Code

```
import os
import cv2
import json
import albumentations as A
from torch.utils.data import Dataset
from operator import itemgetter

BOX_COLOR = (255,0,0) # RED
TEXT_COLOR = (255, 255, 255) # WHITE

class MyCustomDataset(Dataset):
    def __init__(self, json_path, transform = None):
        self.json_file = json_path
        self.trnasform = transform

    def __getitem__(self, index):
        json_path = self.json_file[index]
        # open json file
        with open(json_path) as f:
            json_file = json.loads(f.read()) ## json 라이브러리 이용

        category_ids, category_id_list= [], []
        for item in json_file["categories"]:
            category_ids.append(item["id"])
            category_id_list.append(item["name"])

        # category dictionary
        category_id_to_name = dict(zip(category_ids, category_id_list))
        image_path= os.path.join("./2022.12/12.16_d53_image/data",
                                   json_file['images'][0]['file_name'])
        image = cv2.imread(image_path)

        bboxes = []
        annotations = sorted(json_file["annotations"], key=
itemgetter("category_id"))
        for i in range(len(annotations)):
            bboxes.append(annotations[i]["bbox"])
        return image, bboxes, category_ids, category_id_to_name

    def __len__(self):
        return len(self.json_file)
```

```

def visualize_bbox(image, bboxes, category_ids, category_id_to_name,
                  color=BOX_COLOR, thickness=2):
    img = image.copy()
    for bbox, category_id in zip(bboxes, category_ids):
        class_name = category_id_to_name[category_id]

        x_min, y_min, w, h = bbox
        x_min, x_max, y_min, y_max = int(x_min), int(
            x_min + w), int(y_min), int(y_min + h)

        cv2.rectangle(img, (x_min, y_min), (x_max, y_max),
                      color=color, thickness=thickness)
        cv2.putText(img, text=class_name, org=(x_min, y_min-15),
                    fontFace=cv2.FONT_HERSHEY_COMPLEX, fontScale=1,
color=color,
                    thickness=thickness)
    cv2.imshow("test", img)
    cv2.waitKey(0)

json_path= ["/2022.12/12.16_d53_image/data/instances_default.json"]
data = MyCustomDataset(json_path)

for item in data:
    image          = item[0]
    bboxes         = item[1]
    category_ids   = item[2]
    category_id_to_name = item[3]

# transforms
transfor = A.Compose([
    # A.RandomSizedBBBoxSafeCrop(width=448, height=336, erosion_rate=0.2),
    A.VerticalFlip(p=1),
    A.HorizontalFlip(p=1),
], bbox_params=A.BboxParams(format='coco', label_fields=['category_ids']))

transformed = transfor(image= image, bboxes= bboxes, category_ids=
category_ids)

visualize_bbox(transformed['image'], transformed['bboxes'],
transformed['category_ids'],
                category_id_to_name,
                color= BOX_COLOR, thickness=2)

```

Result

- Bounding box of Cat & Dog

