

# 批量文件格式互相转换

png到jpg

```
from PIL import Image
import os

# 设置需要转换的图片目录
img_dir = r"C:\Users\jajnw\Downloads\archive\images\train"

# 遍历目录下所有文件
for filename in os.listdir(img_dir):
    # 判断文件是否为png格式
    if filename.endswith(".png"):
        # 构造新的文件名, 将后缀改为jpg
        new_filename = os.path.splitext(filename)[0] + ".jpg"
        # 打开图片
        img = Image.open(os.path.join(img_dir, filename))
        # 如果图片是RGBA模式, 转换为RGB模式
        if img.mode == "RGBA":
            img = img.convert("RGB")
        # 如果图片是P模式, 转换为RGB模式
        if img.mode == "P":
            img = img.convert("RGB")
        # 保存为新的jpg格式图片
        img.save(os.path.join(img_dir, new_filename))
        # 关闭图片
        img.close()
        # 删除原来的png格式图片
        os.remove(os.path.join(img_dir, filename))
```

文件重命名

```
import os

def rename_files_in_directory(directory):
    # 获取指定目录下的所有文件名
    for filename in os.listdir(directory):
        # 检查文件名是否以"annotations"开头
        if filename.startswith("annotations"):
            # 构造新的文件名, 去掉"annotations"
            new_filename = filename.replace("annotations", "", 1)
            # 获取文件的完整路径
            old_file_path = os.path.join(directory, filename)
            new_file_path = os.path.join(directory, new_filename)
            # 重命名文件
            os.rename(old_file_path, new_file_path)
            print(f'Renamed: {filename} -> {new_filename}')

# 指定要重命名文件的目录
```

```
directory_path = r'C:\Users\jajnw\Downloads\archive\labels\train' # 请替换为你的目录
# 路径
rename_files_in_directory(directory_path)
```

xml到对应yolo能识别的txt

```
import xml.etree.ElementTree as ET
import os
from os import getcwd
from os.path import join
import glob

sets = ['train', 'test'] # 分别保存训练集和测试集的文件夹名称

# 需要修改的地方1
# 将里面的内容换成自己的标签
classes = ['without_mask', 'with_mask'] # 标注时的标签

'''
xml中框的左上角坐标和右下角坐标(x1,y1,x2,y2)
》》txt中的中心点坐标和宽和高(x,y,w,h)，并且归一化
'''

def convert(size, box):
    dw = 1. / size[0]
    dh = 1. / size[1]
    x = (box[0] + box[1]) / 2.0
    y = (box[2] + box[3]) / 2.0
    w = box[1] - box[0]
    h = box[3] - box[2]
    x = x * dw
    w = w * dw
    y = y * dh
    h = h * dh
    return (x, y, w, h)

def convert_annotation(data_dir, imageset, image_id):
    in_file = open(data_dir + '/%s.xml' % (image_id)) # 读取xml
    out_file = open(data_dir + '%s.txt' % (image_id), 'w') # 保存txt

    tree = ET.parse(in_file)
    root = tree.getroot()
    size = root.find('size')
    w = int(size.find('width').text)
    h = int(size.find('height').text)
    for obj in root.iter('object'):
        difficult = obj.find('difficult').text
        cls = obj.find('name').text
        if cls not in classes or int(difficult) == 1:
            continue
        cls_id = classes.index(cls) # 获取类别索引
```

```

xmlbox = obj.find('bndbox')
b = (float(xmlbox.find('xmin').text), float(xmlbox.find('xmax').text),
float(xmlbox.find('ymin').text),
float(xmlbox.find('ymax').text))
bb = convert((w, h), b)
out_file.write(str(cls_id) + " " + " ".join([str('%.6f' % a) for a in
bb])) + '\n')

wd = getcwd()
print(wd) # 当前路径

# 需要修改的地方2
# 复制文件夹的相对路径到这里来
data_dir = r'C:\Users\jajnw\Downloads\archive\annotations'

for image_set in sets:
    image_ids = []
    for x in glob.glob(data_dir + '/*.xml'):
        print(x)
        image_ids.append(os.path.basename(x)[:4])
    print('\n%s数量:' % image_set, len(image_ids)) # 确认数量
    i = 0
    for image_id in image_ids:
        i = i + 1
        convert_annotation(data_dir, image_set, image_id)
        print("%s 数据:%s/%s文件完成!" % (image_set, i, len(image_ids)))

print("Done!!!")

```

jsontotxt

```

import json
import os

name2id = {'cat': 0} # 标签名称

def convert(img_size, box):
    dw = 1. / (img_size[0])
    dh = 1. / (img_size[1])
    x = (box[0] + box[2]) / 2.0 - 1
    y = (box[1] + box[3]) / 2.0 - 1
    w = box[2] - box[0]
    h = box[3] - box[1]
    x = x * dw
    w = w * dw
    y = y * dh
    h = h * dh
    return (x, y, w, h)

```

```

def decode_json(json_folder_path, json_name):
    txt_name = 'C:/Users/jajnw/Desktop/yolov5-master/catndog/labels/text' +
    json_name[0:-5] + '.txt'
    # txt文件夹的绝对路径
    txt_file = open(txt_name, 'w')

    json_path = os.path.join(json_folder_path, json_name)
    data = json.load(open(json_path, 'r', encoding='gb2312', errors='ignore'))

    img_w = data['imageWidth']
    img_h = data['imageHeight']

    for i in data['shapes']:

        label_name = i['label']
        if (i['shape_type'] == 'rectangle'):
            x1 = int(i['points'][0][0])
            y1 = int(i['points'][0][1])
            x2 = int(i['points'][1][0])
            y2 = int(i['points'][1][1])

            bb = (x1, y1, x2, y2)
            bbox = convert((img_w, img_h), bb)
            txt_file.write(str(name2id[label_name]) + " " + " ".join([str(a) for
a in bbox]) + '\n')

if __name__ == "__main__":

    json_folder_path = 'C:/Users/jajnw/Desktop/yolov5-master/catndog/labels/json'
    # json文件夹的绝对路径
    json_names = os.listdir(json_folder_path)
    for json_name in json_names:
        decode_json(json_folder_path, json_name)

```