
ImgAnn

Release 0.7.12

nipdep

Sep 12, 2022

CONTENTS

1	Contents	3
1.1	Usage	3
1.2	API	9
	Index	17



ImgAnn is a Python library for converting between popular image annotation formats and preview annotated dataset. Mainly, because of converting annotation file from one format to another is a tedious task to handle and also crucial as all other steps in the deep learning project. The target of this library is to provided convenient platform to convert bounding box image annotation file between popular formats [PascalVOC, COCO, csv], preview annotation dataset and get a summary of annotated dataset.

Check out the [Usage](#) section for further information, including how to [Installation](#) the project.

CONTENTS

1.1 Usage

1.1.1 Installation

To use imgann, you could install using [PyPi](#) :

```
(.venv) $ pip install imgann
```

Another option is to directly build the library from codebase :

```
% clone codebase
$ git clone https://github.com/nipdep/imgann.git
% for usual usage
$ pip install -e .
% for development
$ pip install -e .[dev]
```

1.1.2 Functionalities

Annotated Dataset Preview

It's import to make sure downloaded dataset image annotations are in proper / precise manner, And after all it's good to check the resulting annotations after custom annotation type conversion or annotation conversion provided by this library.

The following view function work in both *python* and *IPython* kernels. but ew encourage you to use in interactive python environment such as Jupyter notebooks. Also, dataset and annotation file paths could be in either relative or absolute formats. This function generates set or pseudo random images with their bounding + label on. Also, you could define resulting image shape and the seed to get consistent image outputs.

Note: The *image-shape* does not change the aspect ratio of the images in the dataset at any point. More explicitly, For example let say the images shape of original dataset is (246, 246); means the aspect ration of 1:1. So, even you have input `image_shape=(400, 500)` the resulting image in the shape (400,400) to preserve original aspect ratio.

Code Example:

```
from imgann import Sample
Sample.show_samples(data_path='../data/Hard Hat Sample.v5.voc/test',
                    ann_path='../data/Hard Hat Sample.v5.voc/test',
                    num_of_samples=5,
                    ann_type='voc',
                    seed=123,
                    image_shape=[500, 500])
```

Sample Output



Note: For further instruction follow to API page.

Convert Annotation Format

The library support converting between PascalVOC, COCO and CSV. In General, all the functions take parameter as image dataset directory and annotation file directory.

COCO to PascalVOC

Note:

the parameter 'center' defines the bounding box define formats;

[X_center, Y_center, Width, Height] < if center=True

[X_min, Y_min, Width, Height] < if center=False. i.e. roboflow annotated .json files saved in this format.

Code Example:

```
from imgann import Converter
Converter.coco2voc(dataset_dir='../data/Hard Hat Sample.v5i.coco/test',
                  coco_ann_dir='../data/Hard Hat Sample.v5i.coco/test/_annotations.coco.
↪ json',
                  save_dir='../data/coco2voc')
```

COCO to CSV

Note:

The library supports two CSV formats as the output.

The first format is directly applicable with any object detection work. hence the result contains bounding boxes.

In the second format it contains only number of distinct class contains in each image; which format directly supports to multi-class multi-label classification task.

Code Example:

```
from imgann import Converter
Converter.coco2csv(dataset_dir='../data/Hard Hat Sample.v5i.coco/test',
                  coco_ann_dir='../data/Hard Hat Sample.v5i.coco/test/_annotations.coco.
↪ json',
                  save_dir='../data/coco2csv.csv')
```

COCO to Yolo

Code Example:

```
from imgann import Converter
Converter.coco2yolo(dataset_dir='../data/Hard Hat Sample.v5i.coco/test',
                   coco_ann_dir='../data/Hard Hat Sample.v5i.coco/test/_annotations.
↪ coco.json',
                   save_dir='../data/coco2yolo')
```

PascalVOC to COCO

Code Example:

```
from imgann import Convertor
Convertor.voc2coco(dataset_dir='../data/Hard Hat Sample.v5i.coco/test',
                  voc_ann_dir='../data/coco2voc',
                  save_dir='../data/voc2coco.json')
```

PascalVOC to CSV

Code Example:

```
from imgann import Convertor
Convertor.voc2csv(dataset_dir='../data/Hard Hat Sample.v5.voc/test',
                  voc_ann_dir='../data/Hard Hat Sample.v5.voc/test',
                  save_dir='../data/voc2csv.csv')
```

PascalVOC to Yolo

Code Example:

```
from imgann import Convertor
Convertor.voc2yolo(dataset_dir='../data/Hard Hat Sample.v5.voc/test',
                   voc_ann_dir='../data/Hard Hat Sample.v5.voc/test',
                   save_dir='../data/voc2yolo')
```

CSV to COCO

Code Example:

```
from imgann import Convertor
Convertor.csv2coco(dataset_dir='../data/Hard Hat Sample.v5i.tensorflow/test/',
                   csv_ann_dir='../data/Hard Hat Sample.v5i.tensorflow/test/_annotations.
↳ csv',
                   save_dir='../data/csv2coco.json')
```

CSV to Yolo

Code Example:

```
from imgann import Convertor
Convertor.csv2yolo(dataset_dir='../data/Hard Hat Sample.v5i.tensorflow/test',
                   csv_ann_dir='../data/Hard Hat Sample.v5i.tensorflow/test/_annotations.
↳ csv',
                   save_dir='../data/csv2yolo')
```

CSV to PascalVOC

Code Example:

```
from imgann import Convertor
Convertor.csv2voc(dataset_dir='../data/Hard Hat Sample.v5i.tensorflow/test',
                  csv_ann_dir='../data/Hard Hat Sample.v5i.tensorflow/test/_annotations.
↳ csv',
                  save_dir='../data/csv2voc')
```

CSV Object Detection to Multi-class Multi-label

Code Example:

```
from imgann import Converter
Converter.csv2multilabel(csv_dir='../data/Hard Hat Sample.v5i.tensorflow/test/_
↳ annotations.csv',
                        save_dir='../data/csv2m.csv')
```

Note: For more info on functional parameters, acceptable input formats and output format refer API pages.

Yolo to COCO

Code Example:

```
Converter.yolo2coco(dataset_dir='../data/Hard Hat Sample.v5i.darknet/test',
                   yolo_ann_dir='../data/Hard Hat Sample.v5i.darknet/test',
                   save_dir='../data/yolo2coco.json',
                   center=True)
```

Yolo to PascalVOC

Code Example:

```
Converter.yolo2voc(dataset_dir='../data/Hard Hat Sample.v5i.darknet/test',
                  yolo_ann_dir='../data/Hard Hat Sample.v5i.darknet/test',
                  save_dir='../data/yolo2voc')
```

Yolo to CSV

Code Example:

```
Converter.yolo2csv(dataset_dir='../data/Hard Hat Sample.v5i.darknet/test',
                  yolo_ann_dir='../data/Hard Hat Sample.v5i.darknet/test',
                  save_dir='../data/yolo2csv.csv')
```

Describe Image Dataset

Get summary of stats of the input datasets and annotation is crucial, and can be considered as the EDA in object detection project.

The library supports summary generation under two levels.

Image Dataset Alone

This function analyse only images under the dataset. **Dataset** could combination set of folders.

Code Example:

```
from imgann import Sample
Sample.describe_data('../data/Hard Hat Sample.v5i.coco')
```

Sample Output:

```
INFO:imgann.sample:
                        IMAGE DATA SUMMARY
=====
```

(continues on next page)

(continued from previous page)

```
number of images : 240
number of folders : 3
folder image counts :
    > test : 10
    > train : 210
    > valid : 20
=====
```

Note: The Folder structure depth only supported down to single folder.

```
from imgann import Sample
Sample.describe_data('../data/Hard Hat Sample.v5i.coco/train')
```

Sample Output:

```
INFO:imgann.sample:
                        IMAGE DATA SUMMARY
=====
number of images      : 210
number of folders     : 1
folder image counts   :
    > train : 210
=====
```

Annotated Dataset

This function analyse the annotation stats in addition to the image stats generates in the above function.

Note: In the function also, supports the two annotation formats under COCO as stated under **PascalVOC to COCO** function.

```
from imgann import Sample
Sample.describe_ann(data_path='../data/Hard Hat Sample.v5i.coco/train',
                   ann_path='../data/Hard Hat Sample.v5i.coco/train/_annotations.coco.
↪ json',
                   ann_type='coco')
```

Sample Output:

```
INFO:imgann.sample:
                        IMAGE ANNOTATION SUMMARY
=====
number of images      : 210
folder image counts   :
    > train : 210
number of image sizes : 1
image_size            : 416 X 416
number of object classes : 4
object classes        : Workers | head | helmet | person
number of objects     : 760
```

(continues on next page)

(continued from previous page)

```
class object count      :
    > head    : 186
    > helmet : 553
    > person : 21
=====
```

1.2 API

1.2.1 ImgAnn.Sample

show_samples

Function Description

show set of random images from the dataset with annotations.

Table 1: Parameter Description

Parameter	Default value	Description
data_path	None	relative path current folder, or absolute path to the main folder of the image dataset
ann_path	None	relative path current folder, or absolute path to the main folder of the annotated file
num_of_samples	5	number of sample images to view in integer format
ann_type	'coco'	annotation type of the file that given in the 'ann_path'. supported names : ['coco', 'voc', 'csv', 'yolo']
center	True	Only applicable in 'coco' annotation format. define in following note.
image_shape	[300, 300]	image size in pixels for resulting images.
seed	0	if seed=0; resulting images will be always random. if seed>0, the resulting images will be same at each execution.

Note:

the parameter 'center' defines the bounding box define formats;

[X_center, Y_center, Width, Height] < if center=True

[X_min, Y_min, Width, Height] < if center=False. i.e. [roboflow](#) annotated .json files saved in this format.

describe_data

Function Description

show the summary of image dataset.

Table 2: Parameter Description

Parameter	Default value	Description
data_path	None	absolute or relative path to image dataset directory. effect of directory illustrated in the Usage section

describe_ann

Function Description

show the summary of image dataset with annotations.

Table 3: Parameter Description

Parameter	Default value	Description
data_path	None	absolute or relative path to image dataset main folder
ann_path	None	absolute or relative path to image annotation file or folder
ann_type	'coco'	annotation format [coco, voc, csv, yolo]
center	True	Only applicable in 'coco' annotation format

1.2.2 ImgAnn.Convertor

coco2csv

Function Description

convert COCO annotation into CSV.

Table 4: Parameter Description

Parameter	Default value	Description
dataset_dir	None	relative path current folder, or absolute path to the main folder of the image dataset
coco_ann_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. '../data/annotations.json'
save_dir	None	annotation file saving location. Ex. '../data/annotations.csv'
center	True	defined in the above 'Note'
is_multilabel	False	if 'True' output will be in <i>CSV (Multi-label)</i> format, else in <i>CSV (Object Detection)</i> format

coco2voc

Function Description

convert COCO annotation into PascalVOC.

Table 5: Parameter Description

Parameter	Default value	Description
dataset_dir	None	relative path current folder, or absolute path to the main folder of the image dataset
coco_ann_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. '../data/annotations.json'
save_dir	None	annotation file saving location. Ex. '../data/annotations/'
center	True	defined in the above 'Note'

coco2yolo

Function Description

convert COCO annotation into Yolo.

Table 6: Parameter Description

Parameter	Default value	Description
dataset_dir	None	relative path current folder, or absolute path to the main folder of the image dataset
coco_ann_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. <code>../data/annotations.json</code>
save_dir	None	annotation file saving location. Ex. <code>../data/annotations/</code>
center	True	defined in the above 'Note'

voc2coco

Function Description

convert PascalVOC annotation into COCO.

Table 7: Parameter Description

Parameter	Default value	Description
dataset_dir	None	relative path current folder, or absolute path to the main folder of the image dataset
voc_ann_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. <code>../data/annotations/</code>
save_dir	None	annotation file saving location. Ex. <code>../data/annotations.json</code>
center	True	defined in the above 'Note'

voc2csv

Function Description

convert PascalVOC annotation into CSV.

Table 8: Parameter Description

Parameter	Default value	Description
dataset_dir	None	relative path current folder, or absolute path to the main folder of the image dataset
voc_ann_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. <code>../data/annotations/</code>
save_dir	None	annotation file saving location. Ex. <code>../data/annotations.csv</code>
is_multilabel	False	if 'True' output will be in <i>CSV (Multi-label)</i> format, else in <i>CSV (Object Detection)</i> format

voc2yolo

Function Description

convert PascalVOC annotation into Yolo.

Table 9: Parameter Description

Parameter	Default value	Description
dataset_dir	None	relative path current folder, or absolute path to the main folder of the image dataset
voc_ann_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. ‘../data/annotations/’
save_dir	None	annotation file saving location. Ex. ‘../data/annotations/’

csv2coco

Function Description

convert CSV annotation into COCO.

Table 10: Parameter Description

Parameter	Default value	Description
dataset_dir	None	relative path current folder, or absolute path to the main folder of the image dataset
csv_ann_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. ‘../data/annotations.csv’
save_dir	None	annotation file saving location. Ex. ‘../data/annotations.json’
center	True	defined in the above ‘Note’

csv2voc

Function Description

convert CSV annotation into PascalVOC.

Table 11: Parameter Description

Parameter	Default value	Description
dataset_dir	None	relative path current folder, or absolute path to the main folder of the image dataset
csv_ann_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. ‘../data/annotations.csv’
save_dir	None	annotation file saving location. Ex. ‘../data/annotations/’

csv2yolo

Function Description

convert CSV annotation into Yolo.

Table 12: Parameter Description

Parameter	Default value	Description
dataset_dir	None	relative path current folder, or absolute path to the main folder of the image dataset
csv_ann_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. ‘../data/annotations.csv’
save_dir	None	annotation file saving location. Ex. ‘../data/annotations/’

csv2multilabel

Function Description

convert CSV (Object Detection)annotation into CSV (Multi-label).

Table 13: Parameter Description

Parameter	Default value	Description
csv_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. ‘../data/annotations.csv’
save_dir	None	annotation file saving location. Ex. ‘../data/annotations_m.csv’

yolo2coco

Function Description

convert Yolo annotation into COCO.

yolo2voc

Function Description

convert Yolo annotation into PascalVOC.

Table 14: Parameter Description

Parameter	Default value	Description
dataset_dir	None	relative path current folder, or absolute path to the main folder of the image dataset
yolo_ann_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. ‘../data/annotations/’
save_dir	None	annotation file saving location. Ex. ‘../data/annotations/’

yolo2csv

Function Description

convert Yolo annotation into CSV.

Table 15: Parameter Description

Parameter	Default value	Description
dataset_dir	None	relative path current folder, or absolute path to the main folder of the image dataset
yolo_ann_dir	None	relative path current folder, or absolute path to the main folder of the annotated file. Ex. ‘../data/annotations/’
save_dir	None	annotation file saving location. Ex. ‘../data/annotations.csv’
is_multilabel	False	if ‘True’ output will be in <i>CSV (Multi-label)</i> format, else in <i>CSV (Object Detection)</i> format

1.2.3 Supporting Annotation File Examples

COCO

```
{
  "annotations": [
    {
      "id": "1",
      "image_id": "1",
      "category_id": 1,
      "area": 22165,
      "bbox": [170, 114, 313, 269],
      "ignore": "0",
      "iscrowd": "0"
    },
    .
    .
  ],
  "images": [
    {
      "file_name": "1.jpg",
      "height": 413,
      "width": 413,
      "id": "1"
    },
    .
    .
  ],
  "categories": [
    {
      "id": 1,
      "name": 1,
      "supercategory": "none"
    },
    .
  ],
}
```

PascalVOC

```

<annotation>
  <folder></folder>
  <filename>000008_jpg.rf.d00174cb69229a352e8677a640ec2d86.jpg</filename>
  <path>000008_jpg.rf.d00174cb69229a352e8677a640ec2d86.jpg</path>
  <source>
    <database>roboflow.ai</database>
  </source>
  <size>
    <width>416</width>
    <height>416</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
  <object>
    <name>helmet</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <occluded>0</occluded>
    <bndbox>
      <xmin>201</xmin>
      <xmax>241</xmax>
      <ymin>115</ymin>
      <ymax>142</ymax>
    </bndbox>
  </object>
  <object>
    <name>head</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <occluded>0</occluded>
    <bndbox>
      <xmin>128</xmin>
      <xmax>164</xmax>
      <ymin>151</ymin>
      <ymax>180</ymax>
    </bndbox>
  </object>
</annotation>

```

CSV(*Object Detection*)

Table 16: train.csv

filename	width	height	class	xmin	ymin	xmax	ymax
1.png	416	416	helmet	234	136	265	197
1.png	416	416	head	109	135	145	164

CSV(*Multi-label*)

Description : one-hot encoded format of the all the classes presents in the annotation

Table 17: train.csv

filename	head	helmet
1.png	1	0
2.png	0	1

Yolo

Description : YoloV3 annotation format where save .txt file for each image. in each text file, bounding boxes recorded as;

<label> <x-center> <y-center> <width> <height> format.

where all the values are normalized by the image width&height sizes.

INDEX

C

COCO (*built-in variable*), 14
coco2csv (*built-in variable*), 10
coco2voc (*built-in variable*), 10
coco2yolo (*built-in variable*), 10
CSV (*built-in variable*), 15
csv2coco (*built-in variable*), 12
csv2multilabel (*built-in variable*), 13
csv2voc (*built-in variable*), 12
csv2yolo (*built-in variable*), 12

D

describe_ann (*built-in variable*), 9
describe_data (*built-in variable*), 9

P

PascalVOC (*built-in variable*), 14

S

show_samples (*built-in variable*), 9

V

voc2coco (*built-in variable*), 11
voc2csv (*built-in variable*), 11
voc2yolo (*built-in variable*), 11

Y

Yolo (*built-in variable*), 16
yolo2coco (*built-in variable*), 13
yolo2csv (*built-in variable*), 13
yolo2voc (*built-in variable*), 13