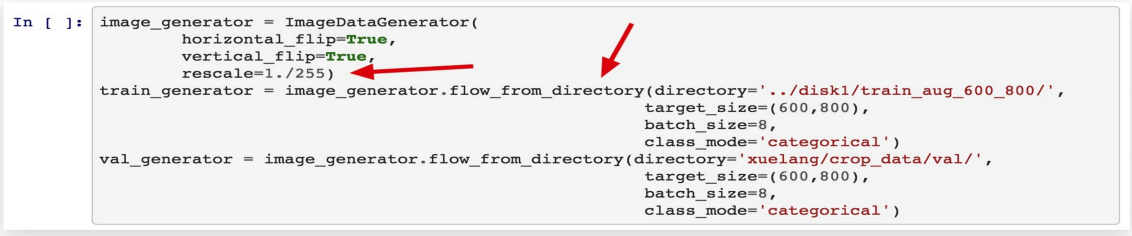
2018年7月30日 11:14

参考1:

https://github.com/effectivelbc/CV_Match 线上0.91

划重点：这里的resize比较特殊（全靠队友zp carry），因为我们发现大的布匹图片默认resize后会出现条状波纹，使用cv2.resize(img, (800, 600), interpolation=cv2.INTER_AREA)可以解决这个问题。



```
In [ ]: image_generator = ImageDataGenerator(
            horizontal_flip=True,
            vertical_flip=True,
            rescale=1./255)
        train_generator = image_generator.flow_from_directory(directory='../disk1/train_aug_600_800/',
                                                              target_size=(600,800),
                                                              batch_size=8,
                                                              class_mode='categorical')
        val_generator = image_generator.flow_from_directory(directory='xuelang/crop_data/val/',
                                                            target_size=(600,800),
                                                            batch_size=8,
                                                            class_mode='categorical')
```

两个问题:1、第一个问题是预处理应该用 InceptionResNetV2 的预处理函数，

From keras_applications.inception_resnet_v2 import preprocess_input

```
Defpreprocess_input(x):
                """Preprocesses a numpy array encoding a batch of images.
                # Arguments
                x: a 4D numpy array consists of RGB values within [0, 255].
                # Returns
                Preprocessed array.
                """
                return imagenet_utils.preprocess_input(x, mode='tf')
```

来自 <https://github.com/keras-team/keras-applications/blob/master/keras_applications/inception_resnet_v2.py#L39>

```
def preprocess_input(x, data_format=None, mode='caffe'):
                """Preprocesses a tensor or Numpy array encoding a batch of images.
                # Arguments
                x: Input Numpy or symbolic tensor, 3D or 4D.
                The preprocessed data is written over the input data
                if the data types are compatible. To avoid
```

this

behaviour, `numpy.copy(x)` can be used.

data_format: Data format of the image tensor/array.

mode: One of "caffe", "tf" or "torch".

- caffe: will convert the images from RGB to BGR,

then will zero-center each color channel with

respect to the ImageNet dataset,

without scaling.

- tf: will scale pixels between -1 and 1,

sample-wise.

- torch: will scale pixels between 0 and 1 and then

will normalize each channel with respect to the

ImageNet dataset.

Returns

Preprocessed tensor or Numpy array.

Raises

ValueError: In case of unknown `data_format` argument.

"""

来自 <https://github.com/keras-team/keras-applications/blob/master/keras_applications/imagenet_utils.py>

2、第二个问题是训练的时候插值没设置，应该设置一个较好的 interpolation(参考：https://github.com/keras-team/keras-preprocessing/blob/master/keras_preprocessing/image.py#L882)

```
def flow_from_directory(self, directory,
                        target_size=(256, 256), color_mode='rgb',
                        classes=None, class_mode='categorical',
                        batch_size=32, shuffle=True, seed=None,
                        save_to_dir=None,
                        save_prefix='',
                        save_format='png',
                        follow_links=False,
                        subset=None,
                        interpolation='nearest'):
```

interpolation： Supported methods are `"nearest"`, `"bilinear"`, and `"bicubic"`.

If PIL version 1.1.3 or newer is installed, `"lanczos"` is also supported,

If PIL version 3.4.0 or newer is installed,"box"` and `"hamming"` are also supported.By default, `"nearest"` is used.

来自 <https://github.com/keras-team/keras-preprocessing/blob/master/keras_preprocessing/image.py#L882>

keras对于数据归一化的操作：keras 有 tf、torch、caffe 三种预处理方式，不同预训练模型不一样。有的会自动归一化，有的不会。对于每一个keras.application的网络都有专有的process_input。在数据预处理时对于图像除于255的归一化操作，要先查看数据预处理函数是否有做归一化操作

参考2：线上得分0.929

https://github.com/bobo0810/XueLangTianchi

问题：有瑕疵图裁剪时候有设置IOU大于0之类的条件吗，还是只是按照那个橙色区域随机裁剪的 。

从有瑕疵图中裁剪420大小的无瑕疵样本时， 先将该图的所有bbox存储到一个list里，然后在有瑕疵原图上随机裁420，计算裁剪的样本与所有bbox的iou.当与所有bbox的iou都为0时，该裁剪样本才可用 。

训练时有瑕疵原图没有resize，测试时原图+按照420，步长为210裁剪，最后取概率最大值

参考3：

https://github.com/maozezhong/TIANCHI_XUELANG_AI