**深度学习训练中关于数据处理方式--原始样本采集以及数据增广**

https://blog.csdn.net/SMF0504/article/details/78695908

# 1绪言

        为啥要写数据，又说数据是深度学习中最关键的一步呢？算法是重要，但是呢，真正实际工程中，业务需求 > 数据  > 算法！根据业务需求来选择适合的数据，我们对数据的处理也是来源业务的需求！而数据选择何种增广方式也需要根据实际做的项目来确定的，举个例子！比如：做车型识别，大家可以想象到，车型识别是干嘛的？嗯，“型”是什么？就是形状有所差别吧！很多人可能已经想到了，对，处理数据是，我们就先从几何形状来增广数据！那么如果是车身颜色呢？又有小伙伴会想到是不是先从颜色来入手来增广数据呢？对了嘛，就是的呢！不过，你不能排除其他的影响，你可以先从你可以想到的角度入手，也要测试一下其他的变化方式！比如说，车型对形状敏感，做完形状之后，你来改变一下亮度或者对比度的变化等等！

        本文代码实现是用的Matlab！

# 2原始样本的采集制作

      训练的第一步就是采集数据制作样本！这个我觉得还是挺关键的，我看了很多视频课以及一些博客什么的，很多没有在这里写一些东西，来告知我们一些注意的问题，就是尽可能能保持现实生活中真实的模样，那么是增广的数据也需要注意模拟现实生活中数据模样，通熟一点就是，贴近现实生活场景！

举例子:

        我们需要从图a中截取出Bus，扣出来的图片，当我们送去训练的时候，还需要将所有样本调整到一样的大小！假定我们调整的到128\*128大小，如果我们不做任何补缺部分，我们就会出现图b中的左边的图，这个图是发生改变的，也就是说形状发生了改变与真实数据有了偏差！但是我们在确实部分有0来补缺，就会像图b中的左图！如果你来选择你会选择哪一张图呢？毋庸置疑，图b中的左图作为训练数据，同时这个数据还模拟了现实生活中那些被遮挡的数据，你说是不是呢？从这里你有没有得到一个启发，就是如果这是一个完整的车，你可不可以通过将部分置黑来起到数据增广的作用呢？



                                                                                                                                        图 a



图 b

# 3数据增广

       在不改变图像类别的情况下，增加数据量，能提高模型的泛化能力！自然图像的数据增广方式包括很多！从几何角度来看，常用的有：水平翻转（也就通差说的是镜像）， 一定程度的位移，裁剪，旋转等！从像素变换来看，常用的有：颜色抖动（color jittering），增加噪声，例如椒盐噪声，高斯噪声等。此外还可以尝试多种操作的组合， 例如同时做旋转和随机尺度变换，此外还可以把每个patch中所有像素在HSV颜色空间中的饱和度和明度提升0.25-4次幂方，乘以0.7-1.4之间的一个因子，再加一个-0.1-0.1之间的值。同样你可以在色调通道（H）对每张图片或patch的所有像素增加一个-0.1-0.1之间的值。

## 3.1几何变换

### 3.1.1翻转

            例如：水平翻转，竖直翻转！其实翻转也不是随意翻转的，要根据实际情况来翻转，比如，关于人脸，你上下翻转了就变成倒的人脸了，这个就没有了实际意义，如果是左右翻转了，也没有太大的实际意义，因为人脸是对称的，你翻转之后还是一样的！但是对于那些没有对称结构的物体图片，你做翻转还是挺有作用的，就相当于有两条数据了啊！不过你，你做测试的时候，可以看看对这个翻转数据敏感不敏感，如果敏感，你就把那些检测错误的数据添加到样本库，或者以低于某个概率的样本增加到样本库！这就起到它的作用了！

matlab代码实现：

**[html]** [view plain](https://blog.csdn.net/SMF0504/article/details/78695908) [copy](https://blog.csdn.net/SMF0504/article/details/78695908)

1. bmp = imread(filname);
2. I = fliplr(bmp);

filename是图片的文件名路径；镜像操作之后的图片如下图所示：



### 3.1.2平移

               这个位置的移动，模拟了现实生活中的那些图片不居中的情况，发生位置的扰动！

matlab代码实现：

**[plain]** [view plain](https://blog.csdn.net/SMF0504/article/details/78695908) [copy](https://blog.csdn.net/SMF0504/article/details/78695908)

1. bmp = imread(filename);  %读入图片
2. se= translate(strel(1), [y, x]);
3. Img = imdilate(bmp, se);

strel用来创建形态学结构元素;

translate（se, [y, x]）在原结构元素se上进行上下偏移，以及左右偏移，y>0 是向下移动，y<0是向上移动；x>0是向左，x<0是向右；

imdilate是形态学膨胀。

平移之后的图片，最左边是原图，中间是向下以及向左平移，最右边的是向上以及向左平移的结果！



### 3.1.3部分置黑

             这个数据模拟了那些被部分遮挡的数据样本

matlab代码实现：

**[plain]** [view plain](https://blog.csdn.net/SMF0504/article/details/78695908) [copy](https://blog.csdn.net/SMF0504/article/details/78695908)

1. I = imread(filename);
2. Img = I;
3. Img(1:up, left:right,:) = 0;
4. Img(down:row, left:right,:) = 0;
5. Img(up:down, 1:left,:) = 0;
6. Img(up:down, right:col,:) = 0;



### 3.1.4旋转

  matlab代码实现：

**[plain]** [view plain](https://blog.csdn.net/SMF0504/article/details/78695908) [copy](https://blog.csdn.net/SMF0504/article/details/78695908)

1. bmp = imread(filename);
2. I = imrotate(bmp, 10, 'bilinear', 'crop');

这里10只是一个数值，可以改为你想旋转的度数。正数向右，负数向左！

这个比较简单，如图所示：



### 3.1.5截切

matlab代码实现：

**[plain]** [view plain](https://blog.csdn.net/SMF0504/article/details/78695908) [copy](https://blog.csdn.net/SMF0504/article/details/78695908)

1. RGB=imread('1,jpg');
2. RGB1=imcrop(RGB,[60,255,400,425]);

注意，剪切的时候，在原始的数据上剪切！其实，我觉得这个功能有类似于平移的功能！发生一些位置的偏移！

### 3.1.6缩放

这个其实就是现实图片放大或者缩小功能！这里的缩放不同于训练前数据统一调整到一样大小的缩放！这只是在原始中稍微改动一些截取的区域，也就是增加或者较少背景的调整，最终还是需要将这些数据调整到你规定的训练数据大小的！

      几何变换暂时先写这么多，后续再进行补充!

## 3.2像素变换

### 3.2.1增加噪声和滤波

matlab代码实现：

**[plain]** [view plain](https://blog.csdn.net/SMF0504/article/details/78695908) [copy](https://blog.csdn.net/SMF0504/article/details/78695908)

1. bmp = imread(filename);
2. I1 = imnoise(bmp, 'salt & pepper'); %椒盐噪声
3. w = [1 2 1; 2 4 2; 1 2 1];
4. I2 = imfilter(bmp, w, 'corr', 'replicate'); %高斯噪声
5. w = [1 1 1; 1 1 1; 1 1 1] / 9;
6. I3 = imfilter(bmp, w, 'corr', 'replicate'); %平均平滑
7. I4 = medfilt2(bmp, [3, 3]);  %中值滤波

噪声的方式也比较多，比如椒盐噪声，高斯噪声，中值滤波等等！

### 3.2.2变换通道

            就是调整RBG三个通道的顺序

matlab代码实现：

**[plain]** [view plain](https://blog.csdn.net/SMF0504/article/details/78695908) [copy](https://blog.csdn.net/SMF0504/article/details/78695908)

1. bmp = imread(filename);
2. I1 = cat(3, bmp(:,:,3), bmp(:,:,2), bmp(:,:,1));
3. I2 = cat(3, bmp(:,:,1), bmp(:,:,3), bmp(:,:,2));

### 3.2.3调整对比度和亮度

         这个模拟了不同光照的影响！这个暂时先不讲了，百度太多了，再写就是重复工作（不过，不排除之后还是补充写一下，哈哈）

其实关于数据像素除了的也还有，就如我刚刚在开头写的都已经帮大家列出来了！

# 4结束语

       写这篇文章的目的其实为了梳理一下，在数据处理方便的方法有哪些，如果大家还有一些好的方法，希望大家可以指点我一下！知识在于分享，你知道一个给我一个，我知道一个分享一个给你，那么我们就有两个知识！