mxnet 实现 mtcnn 人脸检测和特征点定位

发表于 2016-10-22   |   分类于 [study](https://pangyupo.github.io/categories/study/)  |   [暂无评论](https://pangyupo.github.io/2016/10/22/mxnet-mtcnn/#comments)

处理人脸的时候，一直都希望有一个比较快而且又准的方法，这次看到mtcnn放出的模型，稍微试了几下感觉效果不错，就把模型转到了mxnet下，语言也换成了python，方便使用，记录一些转换的过程供以后参考。

**1 论文相关**

作者的[官方实现](https://github.com/kpzhang93/MTCNN_face_detection_alignment)以及作者的[官网](https://kpzhang93.github.io/)。

* 这个方法最大的好处当然是快了(在很准的前提下)，可以对比用mxnet实现的faster rcnn实现的[人脸检测](https://github.com/tornadomeet/mxnet-face" \t "_blank)，速度还是要慢很多的，当然本来faster rcnn就是奔着通用的检测方案去的，用在这里有点大材小用。
* 提速的最大来源还是和cascade一样的思路，用简单的分类器去掉大量的明显非人脸区域，再用逐渐复杂的分类器过滤余下的检测框。不过放弃了rpn对尺度不敏感的性质，也就意味着要生成图像金字塔，这里也是一比不小的开销。
* 只在最后的stage里做landmark的预测，既是节省时间，也是因为前面的人脸框本身就定位不准，那么做landmark噪声太大。

其它的信息参见论文，没有太多好说的。

**2 模型转换**

看了论文之后感觉还是挺简单的，没有对network做什么hack，应该都是现成的模块可以搞定的事情，中间逻辑也不复杂，没想到还是挺多需要注意的事情的。

* 都知道caffe因为使用opencv处理图片的缘故，一般训练出来的模型是读取BGR图像的，而mxnet是约定是RGB的。well，这次反过来了。作者由于是提供的matlab的代码，而matlab本身是RGB排列的，我观察到作者没有更换通道，那么这个模型其实是读取RGB的输入了。另一方面，由于skimage的resize速度没有opencv快，我希望使用opencv做image的io和resize，那么处于效率考虑当然最好用BGR输入。所以弄了一圈还是需要交换一次第一层conv的权重通道。
* 好久没用matlab的我，依稀记得matlab是列优先的内存存放。所以我看到作者直接把图片丢进去的时候一阵的莫名其妙，直到看到作者对x,y坐标的交换操作才明白。原来作者为了效率优先，就没有对内存做过多的修改，这样送给caffe相当于是在检测一张转置后的图像(不知道作者是训练的时候就交换了，还是发布模型的时候转置了权重，只是因为使用matlab就需要这么个额外的步骤，我再强烈安利一发mxnet啊)。那numpy以行为主，我们要拿到合理的输出，当然就是转置权重了。不过注意是在最后的x，y层面才转置，channel和num是要维持不动的。
* 经过上面的操作后，我们得到的feature map是转置后的，那么fc的连接也被我们也需要做对应的修改。如果feature map从  
  $$  
  \begin{vmatrix} 0 & 3 \\ 1 & 4 \\ 2 & 5 \end{vmatrix}  
  $$  
  变成了  
  $$  
  \begin{vmatrix}0 & 1 & 2 \\ 3 & 4 &5 \end{vmatrix}  
  $$  
  ​

fc不考虑它们的摆放关系，所以第一个变成了012345，而第二个的内存顺序是 031425，所以我们想要拿到同样的输出的时候，要把fc的权重也做这样的修改。其实也就是reshape回输出时候的(c, h, w)的shape，然后对后面两个维度转置一下，再flatten回来就好了。

经过上面这上个操作，最后检查一下。把x输入到caffe的model，和把x交换通道并且转置后输入到我们转换的mxnet的model，输出是不是一致的。其它一些需要注意的地方都是一些比较细节的玩意，好像matlab里面的1:4是包括4的，从1开始啊。总之太多约定上不一样的，小心一点就好。另外吐槽一下numpy的(n,)和(n,1)都不好用。

**结果**

[](http://7xsc78.com1.z0.glb.clouddn.com/face_mtcnn.png)

我大概检查了一下输出，和原版输出应该是差不多的撒，不过有时候浮点到整形之类的有一些些出入。

不过去掉matlab和caffe的依赖，可以用python和mxnet来做，实用性还是方便很多的，欢迎大家使用嘿嘿，有问题欢迎留言。