# Pconv网络尝试

全名为partial convolution（部分卷积）

## 网络背景介绍

论文地址

<https://arxiv.org/pdf/1804.07723.pdf>

### 大体介绍：

就是解决普通卷积网络在训练时会利用掩膜中的不真实替代值（一般掩膜采用0或255的值）的做法，容易出现伪影、颜色模糊、灰影等，后处理操作麻烦且代价昂贵。

所以采取数据只在有效像素上进行卷积，自动化更新mask机制。

据说该方法在不规则mask上效果颇佳。

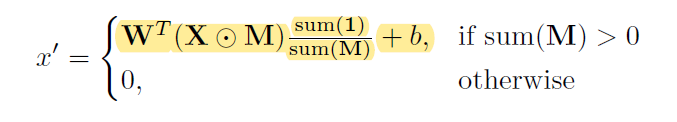
#### 创新点2个：

1.采用部分卷积替代所有卷积。

2.使用的损失函数较为新颖。

##### 详细介绍：

###### 新的卷积方式：

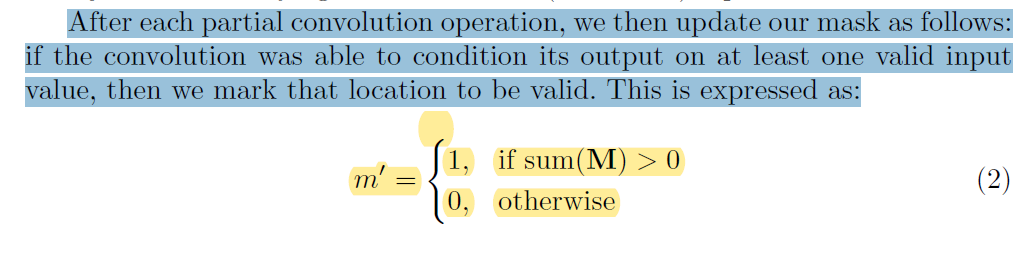


在当前滑动窗口上，选择有效的滑动值参与计算（选择方式是mask为判断标准），并加一个sum除法来规范化。

如果没有有效值，就直接置为0。

计算完成后，Mask更新：

对该点，如果周围没有有效像素值，置为0，有的话，将其置为有效像素值，即新mask里标记为有效值。



###### Loss函数：

（各种loss的公式不细述了）

多种loss函数结合：

loss\_hole (缺失区域的loss损失，L1loss计算)

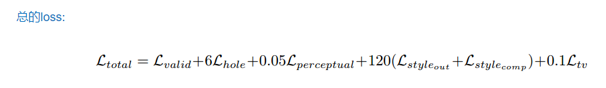
+ loss\_valid（有效区域的loss损失，L1loss计算）

+ loss\_perceptual（感知损失）

+ loss\_style（风格损失）

+ loss\_tv（平滑性损失）

上述每个loss都有一个系数（系数是先验算出来的），然后上述的所有种类loss求总和作为最终的loss函数。

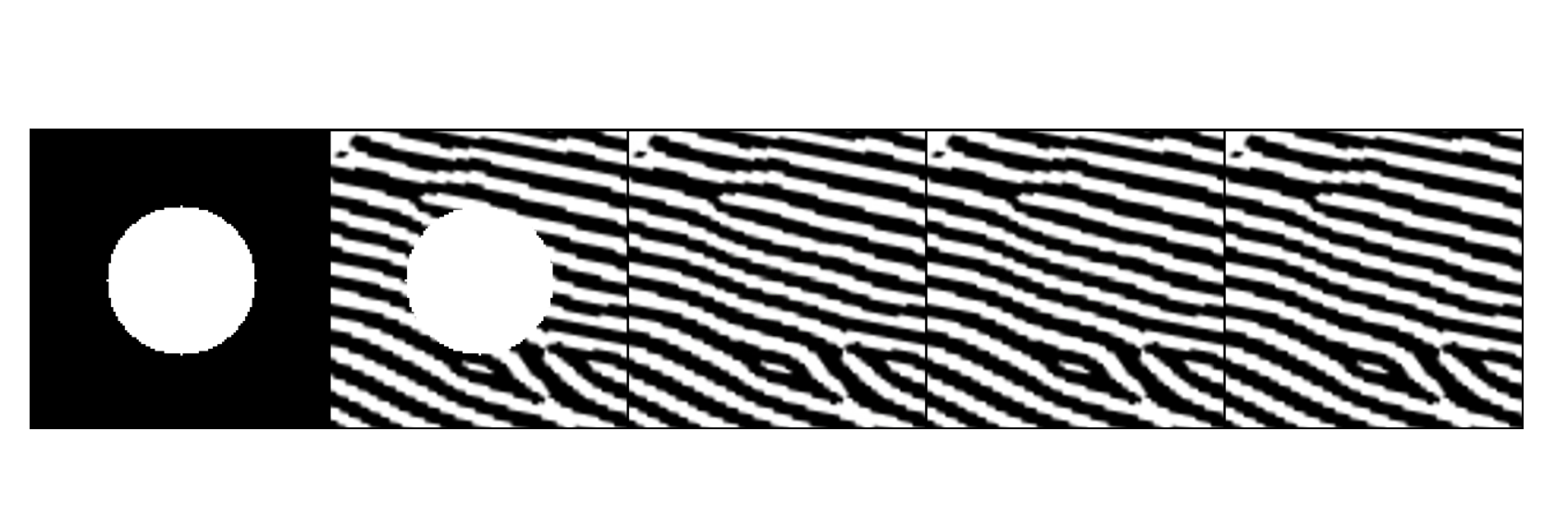


## 结果几个示例:

将1w张照片训练了230次的模型。

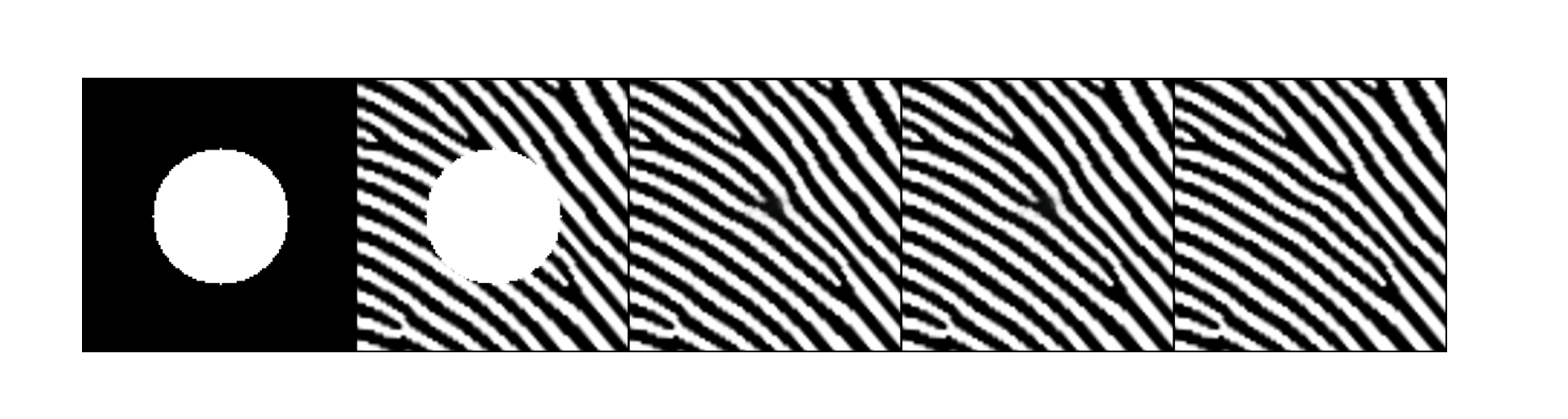
缺失部分没有特征点的图像修复效果已经足够好了。（因为本来就等价于直线连起来。）





### 例子1：

右1是真实照片，右2是补全的照片。



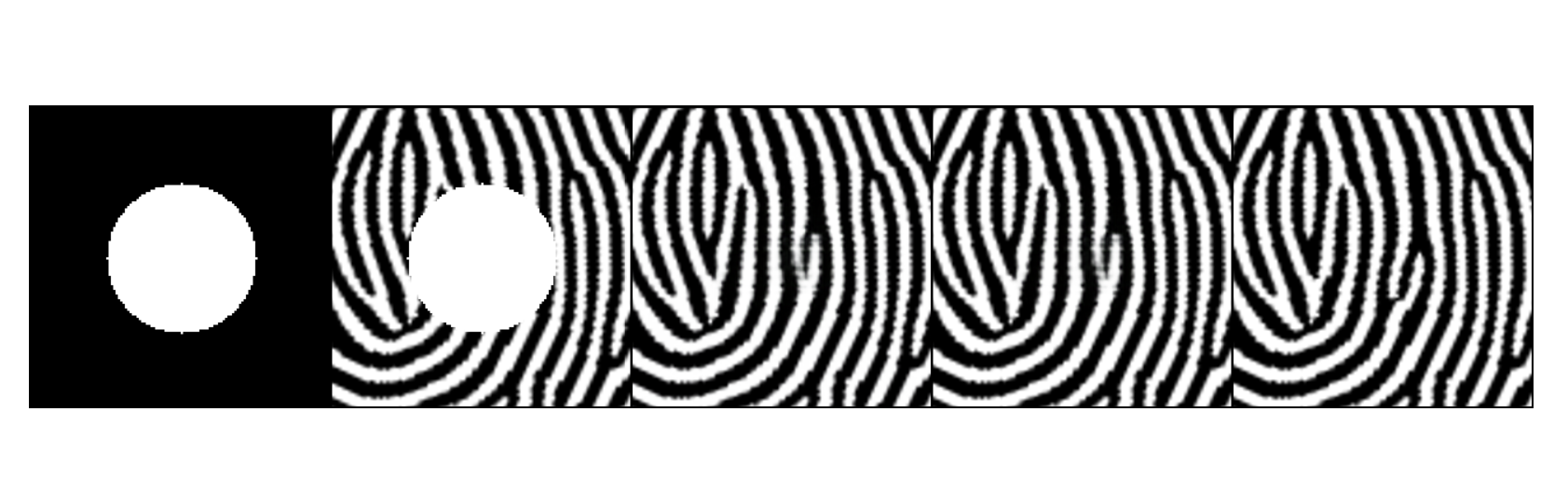
红色绿色的两个点的问题在于，错误预测了位置，但是可以判断正确的是，在该位置附近会出现特征点。

拿来进行特征点预测也是可行的。

同理类似



### 例子2：



右1图有2个特征点，2个分叉点，

右2图有2个特征点，1个分叉点，+1个端点

特征点的数目上还是符合的。

### 例子3：



例3类似于例1，在靠近的地方正确地预测出了特征点是存在的，虽然位置上有一点偏差。

其他例子效果类似

## 总结

效果还行，关于判断特征点种类判断错误的问题，似乎不太好解决，但是就特征点数目上来说，应该是差不多了，如果要用于实际应用，在缺损区域修复出来应该更看重数目，并非种类。

# 对细化图进行修复

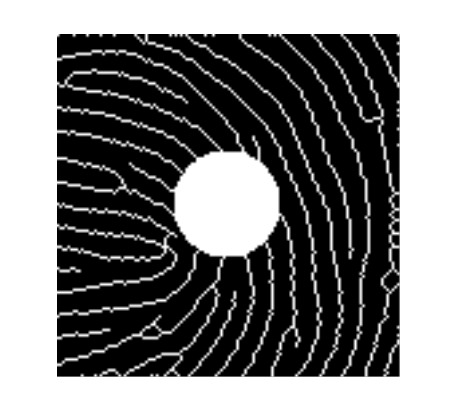
初步预测效果会更好一些，

细化算法用的是

hilditch thining

原图的细化图



缺损图

缺损的图的细化图

现予处理再训练



可是得到的二者不一致在mask周围附近有一些问题。

解决方法：

1. 增大训练时候的mask尺寸

但是要验证一下，缺失后的细化图其他区域的细化线等价于原图的细化线。

写个验证看一下，（不等价，周边的有一点点不一样///）

170次epoch

修复结果，我觉得不是很好，猜测原因在单像素宽度的01类对误差影响有点大。



