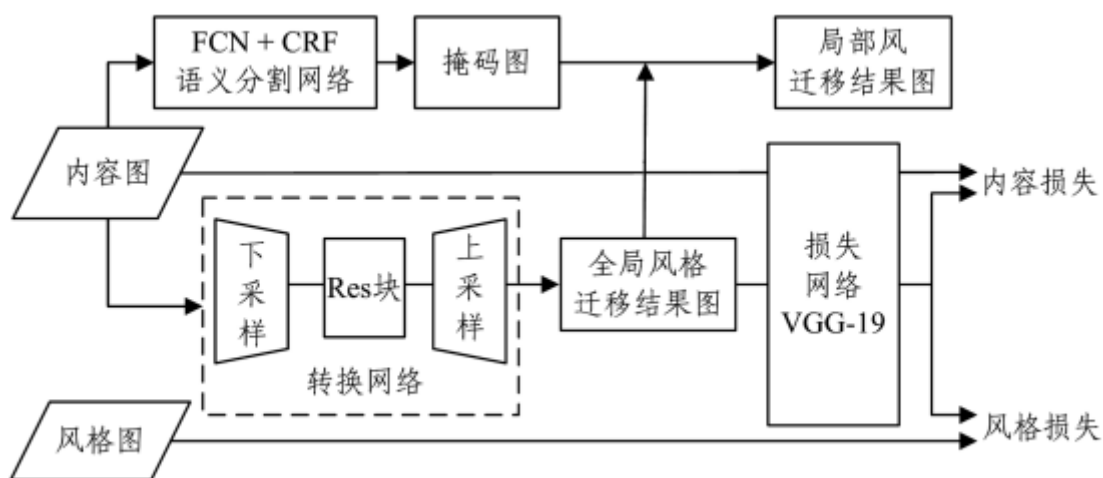


## 第三周论文总结（因为这周的论文不是特别的深，所以没做详细的笔记放到一起了，一共三篇）

### Image Localized Style Transfer Based on Convolutional Neural Network

看这篇论文主要是因为它的结果挺有意思，细看了以后发现emmmmmm，就是用语义风格区分下前后景，然后用cnn进行区域风格迁移

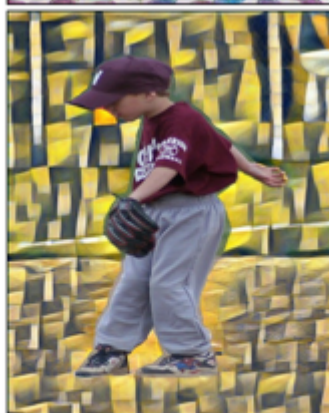
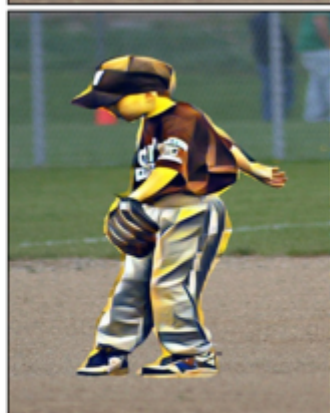
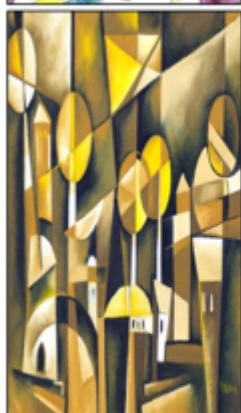


上面的图就可以很清晰的解释模型的运行过程，首先使用语义风格网络对图片进行切分得到掩码图，然后用风格迁移模型处理整张图片，然后把原图和迁移图通过掩码拼到一起，具体效果如

下:



(a) 内容图



(b) 风格图

(c) 前景风格迁移结果图

(d) 背景风格迁移结果图

## Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization

之前有个Instance Normalization的论文，那里提到用实例归一化的效果会更好，这是在那个论文的基础上进行的改进，作者在这里模仿实例归一化，提出了一个自适应实例归一化层，能将内容特征的均值和方差对应到风格特征（就是模仿条件实例归一化方法，但是把两个参数 $a$ 和 $b$ 变成了

依据风格输入变化的函数), 让模型风格迁移能在保持速度的同时, 拥有较强的灵活性  
整体而言就是为实例归一化的两个参数构建了个小网络, 变成这样:  $x$ 是输入,  $y$ 是类别 (风格)

$$\text{AdaIN}(x, y) = \sigma(y) \left( \frac{x - \mu(x)}{\sigma(x)} \right) + \mu(y)$$

## Texture Synthesis Using Convolutional Neural Networks

看这篇论文主要是之前的几个论文都提到了这个说是自己的灵感源泉, 就大概看了下, 这篇论文是专门整纹理合成的, 应该是gram矩阵评判风格的灵感来源 (? ), 整体而言是这样的: 简单的用cnn进行纹理合成, 初始化噪声图像, 用cnn提取模仿图和原始图的特征 (gram矩阵) 并计算差异, 最后用梯度下降减小双方的差距, 得到一个新的纹理图像