

计算机科学与技术学院神经网络与深度学习课程实验 报告

实验题目: style transfer		学号: 201918130222
日期: 2021/11/14	班级: 智能	姓名: 魏江峰
Email: 2257263015@qq.com		
实验目的: 使用 pretrained model 实现 style transfer		
实验软件和硬件环境: 硬件环境: 处理器: Intel core i7 9750-H 电脑: 神州 z7m-ct7nk 软件环境: Pycharm 与 jupyter notebook		
实验原理和方法: 使用 ImageNet 上已经训练好的模型, 实现 style transfer, 即将两张图片作为输入, 输出风格融合后的图片。		
实验步骤: (不要求罗列完整源代码) 1, 导入库和初始化 2, 加载 pretrained model 3, 实现三个 loss 函数:		

(1) Content loss

$$L_c = w_c \times \sum_{i,j} (F_{ij}^\ell - P_{ij}^\ell)^2$$

根据_____

可以写出: content_loss =

(content_curren-content_original).pow(2).sum().mul(content_weight)

(2) Style loss

根据: $G_{ij}^\ell = \sum_k F_{ik}^\ell F_{jk}^\ell$ $L_s^\ell = w_\ell \sum_{i,j} (G_{ij}^\ell - A_{ij}^\ell)^2$ $L_s = \sum_{\ell \in \mathcal{L}} L_s^\ell$

可写出:

N, C, H, W = features.shape

G = torch.zeros(N, C, C)

for i in range(N):

 G[i] = torch.mm(features[i].view(C, -1), features[i].view(C, -1).t())

if normalize:

 G.div_(C * H * W)

return G

(3) Total variation

style_loss = 0

for i in range(len(style_layers)):

 style_loss += style_weights[i] * (gram_matrix(feats[style_layers[i]]) -
 style_targets[i]).pow(2).sum()

4, 实现 total variation loss

根据提示可知：

```
loss = tv_weight*(torch.sum((img[:, :, 1:] - img[:, :, :-1]).pow(2)) +  
    torch.sum((img[:, 1:, :] - img[:, :-1, :]).pow(2)))
```

5, 调用 style_transfer 函数, 查看效果:

Content Source Img.



Style Source Img.



Iteration 100



Iteration 0



Iteration 199



Style Source Img.



Content Source Img.



Iteration 100



Iteration 0



Iteration 199



Iteration 100

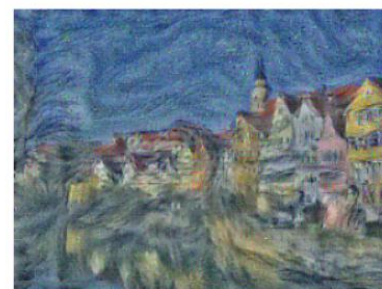
Style Source Img.



Content Source Img.



Iteration 0



Iteration 199



Iteration 100

效果不错~

结论分析与体会：

content loss 计算的时候只关心某一层的 loss，左右整体额 content loss，而 style loss 则是好几层的 layer 的 sytle loss 加起来作为最终的 sytle loss。这是二者之间在计算 loss 上的不同点。

训练好的模型很好用！

就实验过程中遇到和出现的问题，你是如何解决和处理的，自拟 1 - 3 道问答题：