

StyleBank An Explicit Representation for Neural Image Style Transfer (神经图像风格传输的显式表示)

引入

文章发行时，风格迁移很火，大家都用cnn来得到内容和风格特征，但这时候特征依旧耦合，没能彻底分开，而且我们并没有对于风格的显式表示，因此，网络一次只能捕捉一种特定风格——对于新的风格，整个网络必须端到端地重新训练。

因此作者提出了style bank，它由多个滤波器组组成，每组代表一个风格，图片输入后将由编码器和对应的滤波器组进行卷积，这样，我们可以不去大改模型结构，只要为每个风格训练一个滤波器组就可以了，其他的地方不用变，而且因为这样的线性表示，我们可以将不同的风格融合在一起，产生区域特定的风格融合效果

具体实验

作者认为，风格迁移任务需要一种更明确的类似纹理合成中的纹理元那样的表示。

因此作者提出了一个前馈网络，使用编码器将图像（内容）转换到特征空间，引入stylebank作为风格表示，通过输入的风格进行学习，包含多个卷积滤波器组。每个滤波器组代表一种风格，每个通道视为一个小元素，通过与自动编码器生成的内容图像的中间特征图进行卷积，将风格映射到内容图像中，得到一个隐空间的表示。

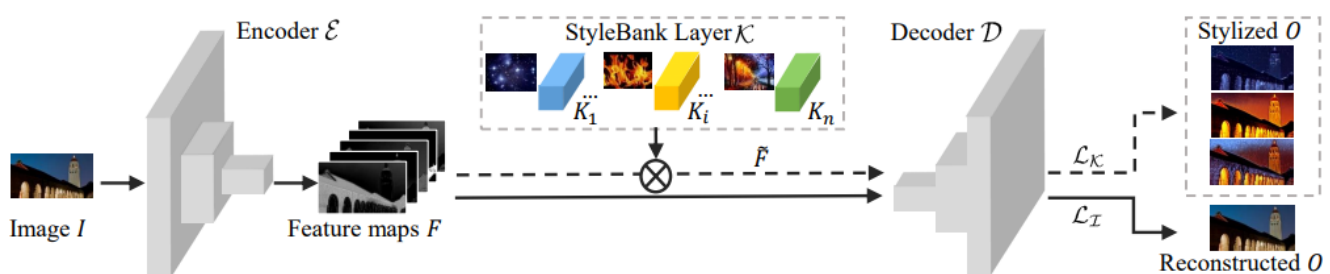


Figure 1. Our network architecture consists of three modules: image encoder \mathcal{E} , StyleBank layer \mathcal{K} and image decoder \mathcal{D}

具体模型结构如上图所示，模型中编码器和解码器要先放到一起训练（非风格化分支），编码器将图像映射到隐空间，解码器争取生成不变的图像，风格化分支中，中间加了一层stylebank，每个滤波器组都可以对隐变量进行卷积得到转换后的特征值，并输入解码器得到风格化的结果

在最初的训练中，我们可以在训练编码器和解码器的同时训练stylebank，在训练结束后，我们可以在固定编解码器后重新初始化新一组滤波器以训练新的风格迁移系统。

对于非风格化分支，我们使用输入和输出间的均方误差作为损失

对于风格化分支，我们融合内容损失 \mathcal{L}_c 、风格损失 \mathcal{L}_s 和变化正则化损失 $\mathcal{L}_{tv}(O_i)$

$$\mathcal{L}_{\mathcal{K}}(I, S_i, O_i) = \alpha \mathcal{L}_c(O_i, I) + \beta \mathcal{L}_s(O_i, S_i) + \gamma \mathcal{L}_{tv}(O_i) \quad (3)$$

$$\mathcal{L}_c(O_i, I) = \sum_{l \in \{l_c\}} \|F^l(O_i) - F^l(I)\|^2$$
$$\mathcal{L}_s(O_i, S) = \sum_{l \in \{l_s\}} \|G(F^l(O_i)) - G(F^l(S_i))\|^2 \quad (4)$$

作者使用了非风格化分支损失来解耦风格与内容，并用实验证明，没有非风格化分支而直接训练的话，编解码器无法完全重建原本的图像，损失了内容，这样的话可以证明，非风格化分支的损失促进了双方内容的解耦

另一方面，作者对内容特征进行了提取与聚类，发现编码器将内容图像分解为多层特征图，它们在空间上有特殊的意义，此外作者还发现，编码出的内容存在稀疏性，这证明模型还可以变得更小，更紧凑，但缩小模型大小后，效果会变差，这可能需要其他的方法进行帮助与修改

模型应用

作者的模型除了最基础的风格迁移功能外，还能玩两种花活：

区域风格迁移——对编码器内容进行聚类分解为 n 个不相交的区域，每个区域用一组风格滤波器卷积，就可以在不同的区域上进行迁移

风格线性融合——简单地线性融合StyleBank层中的滤波器组来线性融合多种风格。其中 m 是风格的数量， K_i 是第 i 种风格的滤波器组。 F 是内容特征（编码器编码出来的）

$$\tilde{F} = \left(\sum_{i=1}^m w_i * K_i \right) \otimes F \quad \sum_{i=1}^m w_i = 1,$$