

Perceptual Losses for Real-Time Style Transfer (风格转移的感知损失)

引言

图像到图像的转换（去噪等操作）在当时的情况中常用像素差异作为损失函数，但这种损失没有考虑到语义与感知上的问题，有时我们感知上看起来没问题，但逐像素测量上损失值极大；因此有人提出用预训练卷积网络的高级图像特征计算差异，这样确实防止了之前所说的感知问题，但是因为由于推断需要解决优化问题（卷积要计算的东西太多了），导致速度较慢。

因此作者提出了一种新的方法，使用依赖于预训练损失网络的高级特征的感知损失函数训练图像转换网络。并在两种应用上进行测试——也就是风格转移和单图像超分辨率，在图像风格迁移任务上，训练速度变得更快，单图像超分辨率任务上，效果更好。

相关工作

前馈图像转换

语义分割方法：以完全卷积方式，使用像素差异作为损失训练模型，或将CRF框架作为与网络其余部分共同训练的循环层一起训练，以超越像素差异损失。（需要看看相关论文）

深度和表面法线估计方法训练有逐像素回归或分类损失的前馈卷积网络，将彩色输入图像转换为具有几何意义的输出图像。通过惩罚图像梯度或使用CRF损失层来强制输出图像的局部一致性，以超越像素差异损失。

感知优化

使用卷积网络来提取感知信息，并以此为基础构建感知优化的模型，以理解训练网络中编码的功能或用于生成高置信度的欺骗图像等技术

Mahendran和Vedaldi通过最小化特征重建损失来反转卷积网络的特征，以理解不同网络层保留的图像信息

Dosovitskiy和Brox训练一个前馈神经网络来反转卷积特征。

风格迁移

Gatys等人通过联合最小化来结合一幅图像的内容和另一幅图像的风格特征重建损失和同样基于预训练卷积网络提取的特征的风格重建损失

但由于优化问题的每个步骤都需要通过预训练网络进行前向和后向传递，因此计算成本高。

图像超分辨率

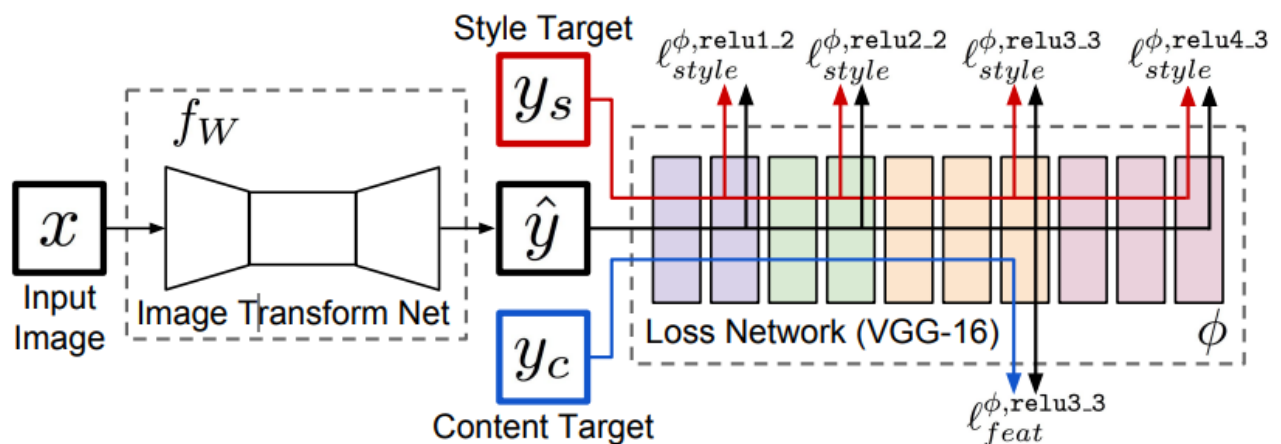
超分辨率技术在卷积神经网络广泛采用之前分为基于预测的方法（双线性、双立方、Lanczos等）、基于边缘的方法、统计方法、基于补丁的方法和稀疏字典方法。

现阶段人们使用三层卷积神经网络训练，并且具有逐像素欧几里得损失，实现了单图像超分辨率上的卓越表现

具体工作

网络结构

模型整体分为两部分：用于实现功能的图像转换网络 f_W （深度卷积转换网络），与计算损失的损失网络 ϕ （为图像分类预训练的网络）



目标有两种——风格目标与内容目标，风格迁移使用这两种目标结合而得到的损失，为每个风格目标训练一个网络，超分辨率仅用内容目标计算损失为每个超分辨率因子（？）训练一个网络。

图像转换网络

整体结构

使用有步长和分数步长的卷积进行网络内的下采样和上采样，网络主体由五个残差块组成，所有非残差卷积层后接空间批量归一化和ReLU非线性激活，输出层使用缩放的tanh函数以确保输出图像的像素在范围内。除了第一层和最后一层使用9×9的卷积核外，所有卷积层都使用3×3的卷积核。

输入输出大小

风格迁移输入输出都是形状为3×256×256的彩色图像。

超分辨率输出是形状为3×288×288的高分辨率图像块，输入是形状为3×288/f×288/f的低分辨率块（因没有全连接层，输入啥它都能管）。

上下采样

使用若干个残差块，然后是步长为1/2的 $\log_2(f)$ 个卷积层（f为超分辨率），依赖固定的上采样函数，分数步长卷积允许与网络其余部分一起学习上采样函数。

损失函数网络

作者在 ImageNet 数据集上预训练了一个16层 VGG 网络（图像分类网络）作为损失感知网络，并借用这个网络定义了两种感知损失函数，以衡量图像之间的感知和语义差异。

特征重建损失

如下图所示，特征重建损失是计算各感知层卷积结果中，目标和输出特征图间的欧式距离，使用这种方法，当我们从更高层重建时，图像内容和整体空间结构得以保留，但颜色、纹理和确切的形状则不会。鼓励输出与目标图像相似，却不需完全匹配

$$\ell_{feat}^{\phi,j}(\hat{y}, y) = \frac{1}{C_j H_j W_j} \|\phi_j(\hat{y}) - \phi_j(y)\|_2^2$$

风格重建损失

计算每层的Gram矩阵，以GRAM矩阵之间的差异作为损失值，gram矩阵可以通过塑性为C*HW的格式来高效计算

$$G_j^\phi(x)_{c,c'} = \frac{1}{C_j H_j W_j} \sum_{h=1}^{H_j} \sum_{w=1}^{W_j} \phi_j(x)_{h,w,c} \phi_j(x)_{h,w,c'}.$$

$$\ell_{style}^{\phi,j}(\hat{y}, y) = \|G_j^\phi(\hat{y}) - G_j^\phi(y)\|_F^2.$$

简单损失

除上述内容外，本文还有两个其他的损失—像素损失 l_{pixel} 与总变异正则化 l_{TV}

实验（具体应用）

风格迁移

使用 $relu1_2, relu2_2, relu3_3, relu4_3$ 计算风格重建损失，使用 $relu2_2$ 计算特征重建损失，整体损失函数如下

$$\hat{y} = \arg \min_y \lambda_c \ell_{feat}^{\phi,j}(y, y_c) + \lambda_s \ell_{style}^{\phi,J}(y, y_s) + \lambda_{TV} \ell_{TV}(y)$$

在对比中表明，这样训练的风格迁移模型能够识别语义模型，让主体（如人，猫）足够清晰，其他的被模糊，更好地保留中心内容

而且，本文的模型训练速度更快，迭代次数更小

图像超分辨率

图像超分辨率这一工作有一个问题是，一个低分辨率图像可能有多个超分辨率图像，因此我们更专注于特征重建损失，而非像素损失

因此作者使用VGG-16中的 $relu2_2$ 层最小化特征重建损失来训练*4与*8的超分辨率模型，训练的特征重建模型在重建锐利边缘和细节方面做得非常好，作者的模型能锐化描绘主体的细节（人）而模糊背景的细节，它更能意识到图像的语义