

# A Closed-form Solution to Photorealistic Image Stylization

## ——照片级逼真的图像风格化

### 1. 摘要

照片级逼真的图像风格化涉及到将参考照片的风格转换为内容照片，其约束条件是，经过风格化的照片应保持照片级逼真程度。虽然存在多种逼真的图像风格化方法，但它们往往会产生具有明显伪影的空间不一致。在这篇论文中，提出一种解决这些问题的方法——FastPhotoStyle。

该方法由**风格化步骤**(stylization step)和**平滑步骤**(smoothing step)组成。当风格化步骤将引用照片的样式转换为内容照片时，平滑步骤确保空间上一致的样式化。每个步骤都有一个封闭的解决方案，可以有效地计算。作者进行了广泛的实验验证。结果表明，与其他方法相比，该方法生成的逼真风格输出更受受试者的青睐，同时运行速度更快，比传统方法快**49 倍**。

### 2. 背景

#### 2.1 图像风格化

图像风格化又可以称为风格迁移，即将一张具有艺术特色的图像的风格迁移到一张普通的图像上，使原有的图像保留原始内容的同时，具有独特的艺术风格，如卡通、漫画、油画、水彩、水墨等。

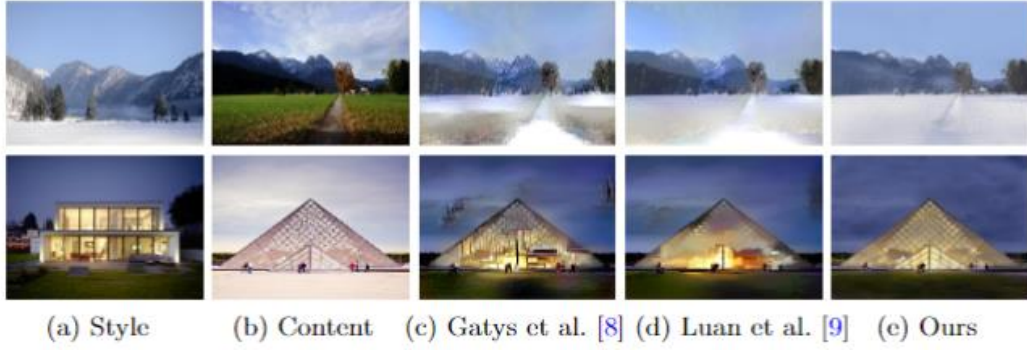


#### 2.2 图像风格化的目标

照片般逼真的图像样式转换的目标是将照片的样式更改为与另一个照片的样式相似。对于忠实的样式化，输出照片中的内容应保持不变，而输出照片的样式应与参考照片的样式相似。此外，输出照片应该看起来像是由相机拍摄的真实照片。

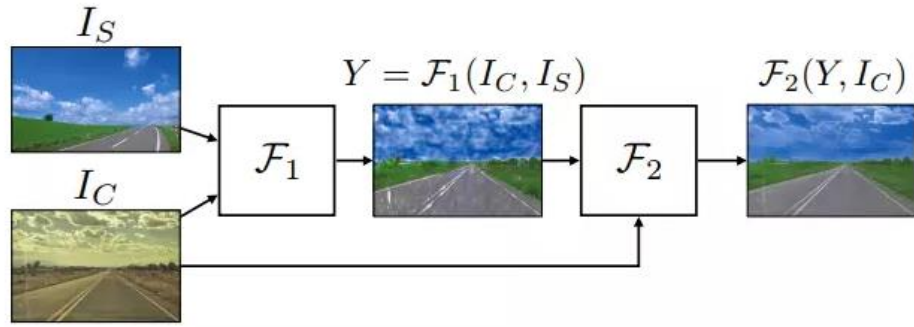
#### 2.3 前人工作

由于缺乏富有表现力的特征表示，经典的样式化方法基于匹配颜色统计（例如，颜色转移或色调转移）或仅限于特定场景（例如，季节和肖像）。最近，Gatys 等人。表明深度特征之间的相关性编码图像的视觉风格，并提出一种基于优化的方法，称为神经风格转移算法，用于图像样式转移。然而虽然该方法对于艺术风格转移（将图像转换为绘画）表现出令人印象深刻的性能，但是当应用于照片级真实图像样式转移任务时，它经常引入结构伪像和失真。在后续工作中，Luan 等人。建议将正则化项添加到神经样式转移算法的优化目标函数，并显示这减少了输出图像中的失真，然而，所得到的算法倾向于在图像中不一致地对语义上均匀的区域进行样式化。



### 3. 本文工作

#### 3.1 总体流程图



第一步是一个称为 PhotoWCT 风格化转换  $F_1$ 。给定样式照片  $I_S$ ， $F_1$  将样式传递给内容图片  $I_C$ ，同时最小化输出图像中的结构伪影。虽然  $F_1$  可以忠实地样式化，但它通常会在语义相似的区域中生成不一致的样式。因此，使用逼真的平滑函数  $F_2$  来消除这些伪影。整个算法可以写成两步映射函数：

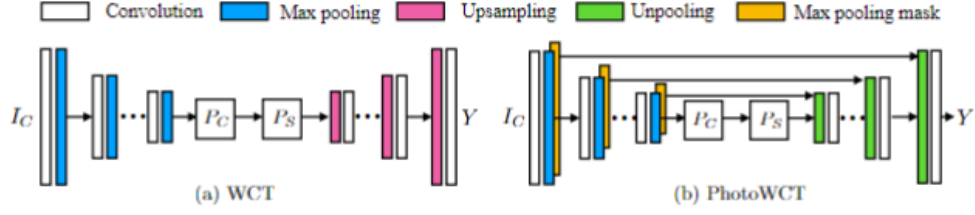
$$F_2(F_1(I_C, I_S), I_C)$$

#### 3.2 风格化

样式化步骤基于白化和着色变换（WCT）算法，并且被称为 PhotoWCT 步骤。

作者简单地回顾了 WCT 方法的步骤，WCT 将样式化表示为具有特征投影的图像重建问题。为了利用 WCT，首先训练用于一般图像重建的自动编码器。具体而言，它使用 VGG-19 模型作为编码器  $E$ （权重保持固定）并训练解码器  $D$  以重建输入图像。解码器与编码器对称，并使用上采样层来放大特征图的空间分辨率。一旦训练了自动编码器，就会在网络瓶颈中插入一对投影功能，以通过白化和着色变换执行样式化。WCT 背后的关键思想是通过双投影直接匹配内容图像与样式图像的相关性。WCT 在艺术图像风格化方面表现良好。然而，它产生用于照片级真实图像风格化的结构伪像（例如，对象边界上的扭曲）。而 PhotoWCT 目标旨在抑制结构伪影。

PhotoWCT 设计的动机是观察到 WCT 中的 max-pooling 操作减少了特征映射中的空间信息。解码器中的上采样特征映射无法恢复输入图像的详细结构。也就是说，我们需要将丢失的空间信息传递给解码器以便于重建这些精细的细节。

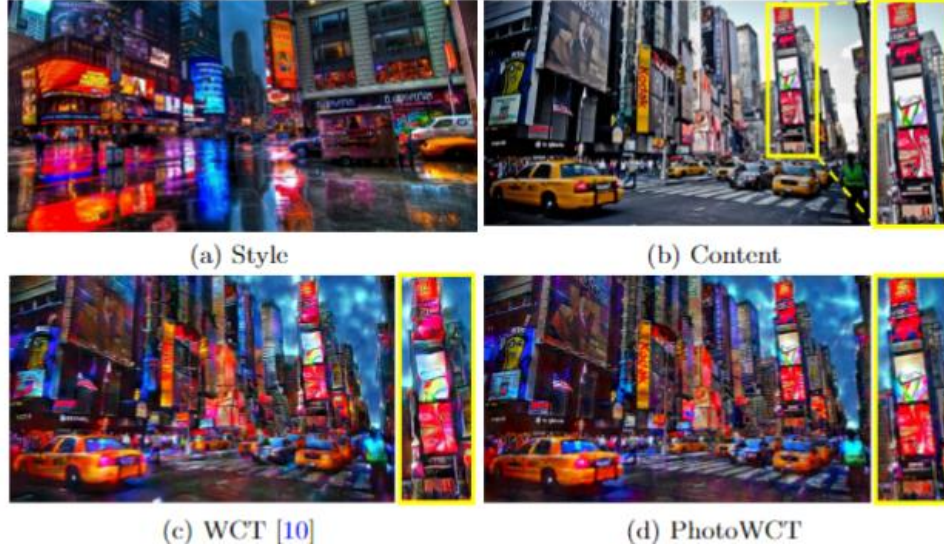


受到上池化层在保留空间信息方面的成功启发，PhotoWCT 使用上池化层来放置 WCT 中的上采样层。PhotoWCT 和 WCT 共享相同的编码器架构和投影步骤。在 PhotoWCT 中，作者用上池化层（绿色）替换上采样层（粉红色）。上池化层与池化掩码（黄色）一起使用，其记录了相应池化层中最大前端最大池区域。PhotoWCT 方法函数为

$$Y = F_1(I_C, I_S) = \bar{D}(P_S P_C H_C)$$

其中  $\bar{D}$  是解码器，包含 unpooling 层，被训练用于图像重建， $P_C$  为白化操作， $P_S$  为着色操作， $H_C$  为矢量化 VGG 特征。

下图比较了 WCT 和 PhotoWCT 的程式化结果。如特写中所示，内容图像中沿建筑物边界的直线在 WCT 样式化结果中变为锯齿形，但在 PhotoWCT 结果中保持笔直。PhotoWCT 风格化的图像具有很少的结构伪像。



### 3.3 平滑化

PhotoWCT 风格化的结果看起来仍然不像照片，因为语义相似的区域通常风格化不一致。如图 d 所示，当应用 PhotoWCT 使用夜间照片对日间照片进行样式化时，如果风格化的天空区域是均匀的深蓝色而不是部分黑暗而部分是浅蓝色，那么风格化的天空区域将更加逼真。基于这一观察，作者采用了内容照片中的像素亲和度光滑的 PhotoWCT 风格化结果。

为了在平滑步骤中实现两个目标。首先，在局部邻域中具有相似内容的像素应该类似地风格化。其次，输出不应与 PhotoWCT 结果显着偏离，以保持全局样式效果。作者首先将所有像素表示为图中的节点，并定义了一个亲和度矩阵  $W = \{w_{ij}\} \in \mathbb{R}^{N \times N}$  ( $N$  是像素数) 以描述像素相似性。作者定义了一个平滑项和一个拟合项，用于在以下优化问题中模拟这两个目标：

$$\argmin_r \frac{1}{2} \left( \sum_{i,j=1}^N w_{ij} \left\| \frac{r_i}{\sqrt{d_{ii}}} - \frac{r_j}{\sqrt{d_{jj}}} \right\|^2 + \lambda \sum_{i=1}^N \|r_i - y_i\|^2 \right)$$



其中 $y_i$ 是 PhotoWCT 样式化结果中的像素颜色  $Y$  和 $r_i$ 是所需平滑输出  $R$  中的像素颜色。变量 $d_{ii} = \sum_j w_{ij}$ 是  $W$  的度矩阵  $D$  中的对角元素即 $D = \text{diag}\{d_{11}, d_{22}, \dots, d_{NN}\}$ 。

该公式是基于图形的排名算法。在排名算法中， $Y$  是二进制输入，其中每个元素指示特定项是否是查询（如果 $y_i$ 是查询则 $y_i = 1$ ，否则 $y_i = 0$ ）。最优解  $R$  是基于其成对亲和度的所有项的排名值。在本文的算法中，其将  $Y$  设置为 PhotoWCT 样式化结果。最优解  $R$  是基于成对像素亲和度的  $Y$  的平滑版本，其促进在语义上相似的区域内的一致地样式化。

上述优化问题是一个简单的二次问题，采用闭式解决方案，由下式给出：

$$R^* = (1 - \alpha)(1 * \alpha S)^{-1}Y$$

其中 $I$ 是单位矩阵 $\alpha = \frac{1}{1+\lambda}$ 和 $S$ 是从 $I_c$ 计算的归一化拉普拉斯矩阵， $S = D^{-\frac{1}{2}}WD^{-\frac{1}{2}} \in$

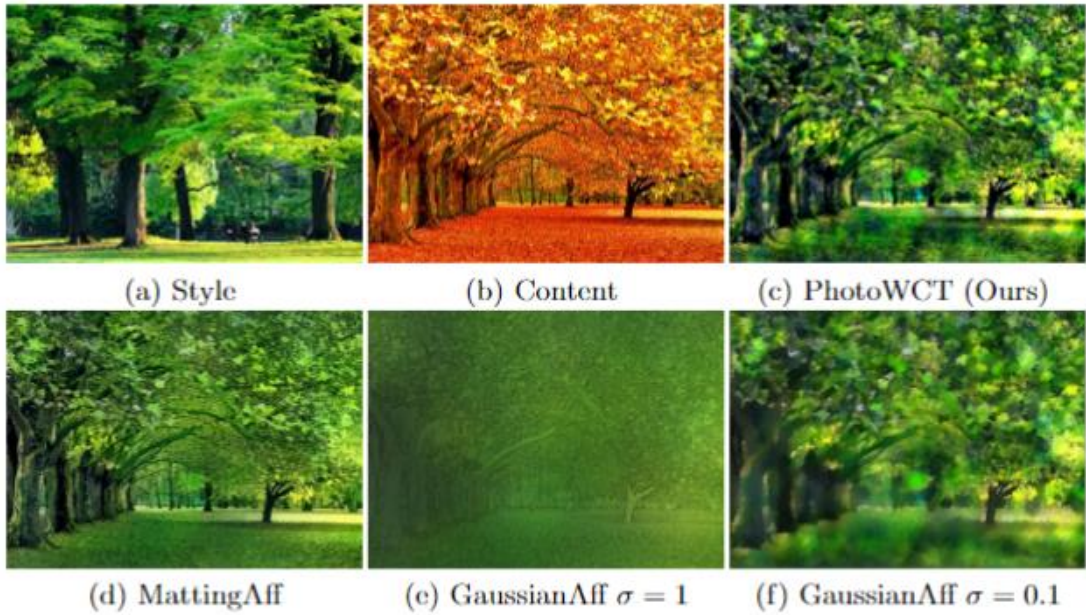
$R^{N \times N}$ ，由于构造的图形通常稀疏地连接（ $W$  中的大多数元素为零），上式中的逆操作在计算上是有效的。使用 close-form 解决方案，平滑步骤可以写为函数映射，由下式给出：

$$R^* = F_2(Y, I_c) = (1 - \alpha)(1 * \alpha S)^{-1}Y$$

### 3.3.1 亲和矩阵

使用基于 8 连接图像图形假设的内容照片来计算亲和度矩阵  $W$ 。虽然存在几种亲和度量选择，但是流行的是将亲和度（表示为高斯亲和度）定义为 $w_{ij} = e^{-\|I_i - I_j\|^2 / \sigma^2}$ 其中 $I_i$ ， $I_j$ 是相邻像素 $i, j$ 的 RGB 值，而 $\sigma$ 是全局缩放超参数。

但是，在实践中很难确定 $\sigma$ 值。它通常导致整张照片过度平滑或不一致地对照片进行样式化（下图 e）。为了避免选择一个全局缩放超参数，我们求助于 matting 亲和力[13,40]，其中两个像素之间的亲和度基于本地窗口中像素的均值和方差。下图 d 显示消光亲和力能够同时很好地平滑不同区域。



作者注意到平滑步骤还可以消除 WCT 中的结构伪影，如下图(e)所示。然而，它导致不令人满意的程式化。主要原因是内容照片和 WCT 结果由于空间扭曲而严重错位。例如，WCT

结果中的建筑物的程式化像素可以对应于内容照片中的天空的像素。因此，这会导致  $Y$  中的错误查询以进行平滑步骤。这就是为什么我们需要首先使用 PhotoWCT 来消除失真。下图(f)显示 PhotoWCT 和平滑的组合导致更好的照片级真实感，同时仍保持忠实的风格化。

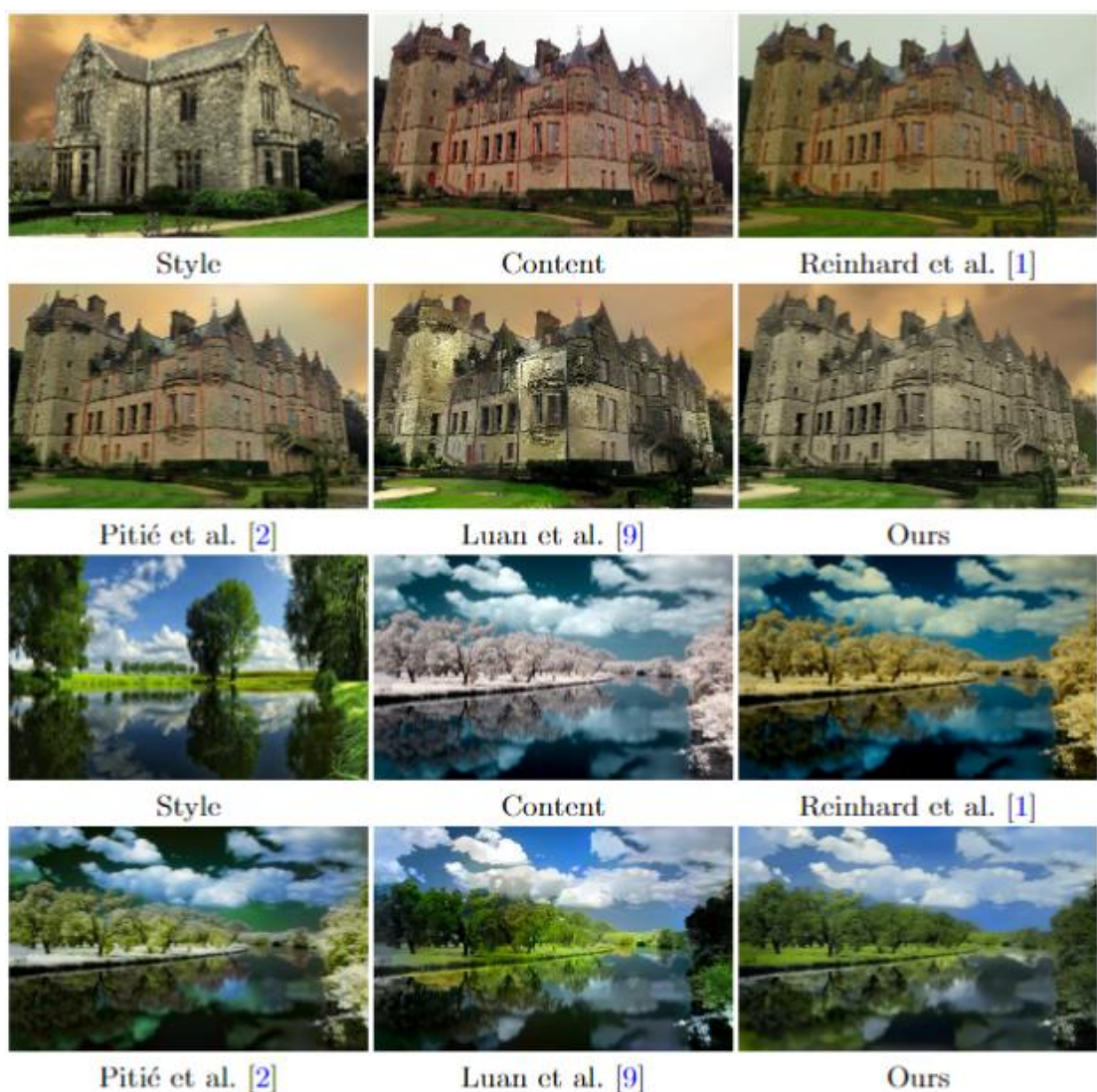


## 4. 实现

代码在 <https://github.com/NVIDIA/FastPhotoStyle> 进行了开源。作者分别阐述了实现的细节，此处不再赘述。

## 5. 结果

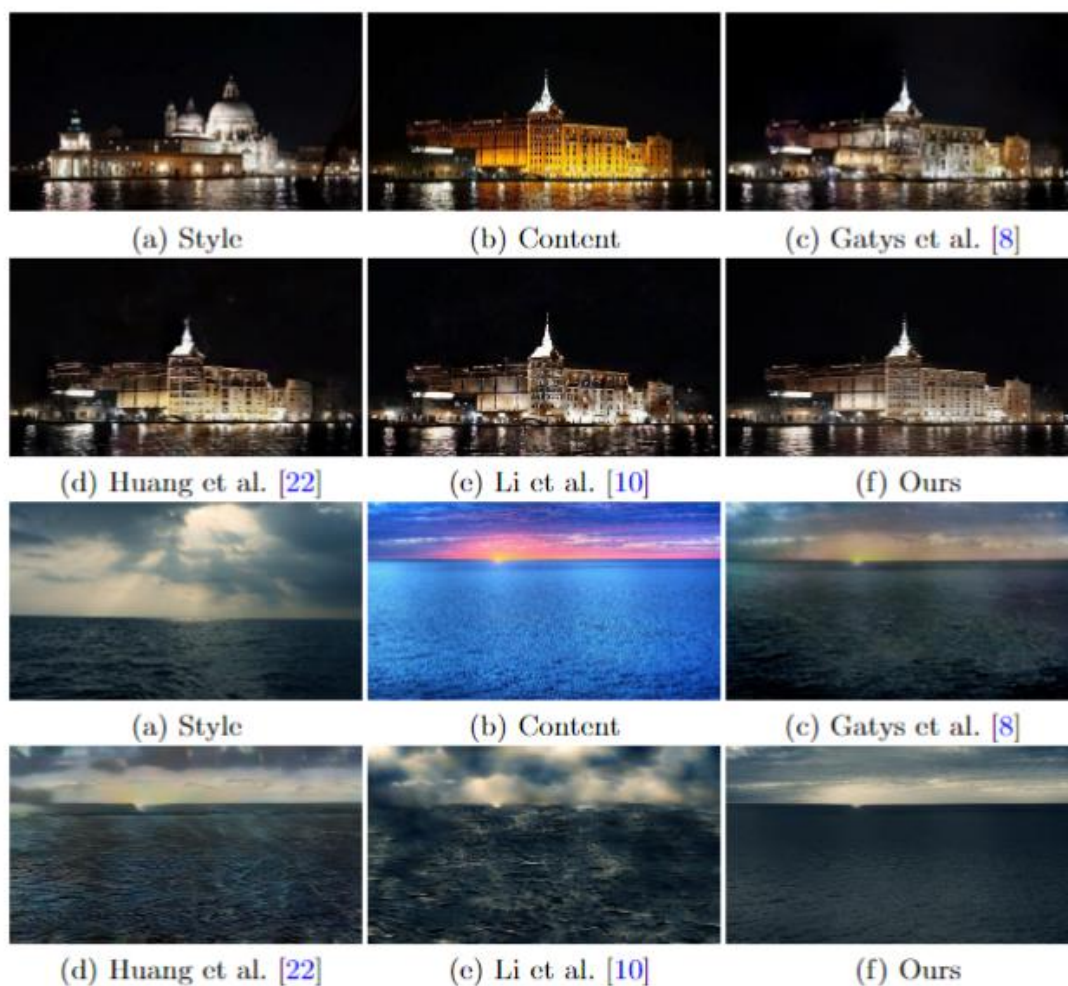
### 5.1 不同照片级逼真的图像风格化







## 5.1 艺术风格化比较



## 6. 性能评估

### 6.1 竞争算法运行时间

Table 3: Run-time comparison. We compute the average run time (in seconds) of the evaluated algorithms across various image resolutions.

Image resolution	Luan et al. [9]	proposed	PhotoWCT	smoothing	<b>approx</b>
256×128	79.61	0.96	0.40	0.56	0.41
512×256	186.52	2.95	0.42	2.53	0.47
768×384	380.82	7.05	0.53	6.52	0.55
1024×512	650.45	13.16	0.56	12.60	0.64

## 6.2 平滑效果

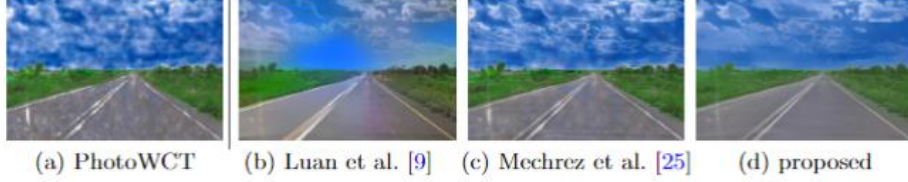


Fig. 9: Comparison between using our photorealistic smoothing step and other refinement methods (b)-(d).

## 6.3 用户偏好

Table 4: User preference score comparison: comparing **approx** (the fast approximation of the proposed algorithm) to the proposed algorithm as well as other photorealistic stylization algorithms.

	proposed/ <b>approx</b>	Luan et al. [9]/ <b>approx</b>	Pitić et al. [2]/ <b>approx</b>
Better stylization	<b>59.6%</b> / 40.4	36.4 / <b>63.6%</b>	46.0 / <b>54.0%</b>
Fewer artifacts	<b>52.8%</b> / 47.2	20.8 / <b>79.2%</b>	46.8 / <b>53.2%</b>

## 6.4 $\lambda$ 灵敏性分析

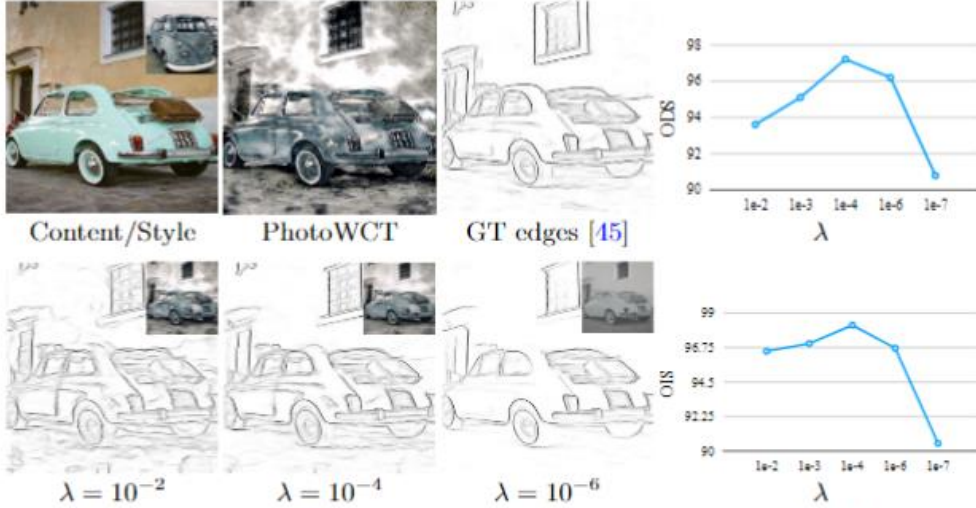


Fig. 8: Visualization of effects of using different  $\lambda$  values in the photorealistic smoothing step. We show the edge maps of different stylization results (inset) at bottom and compare them with the edge map of the content in terms of the ODS and OIS metric (rightmost).

## 7. 结论

### 7.1 概要总结

这个来自英伟达(NVIDIA)和加州大学默塞德分校的研究团队提出了一种新的解决照片级图像风格化的方法——FastPhotoStyle。该方法包括两个步骤：风格化和平滑化。大量的实验表明，该方法生成的图像比以前的最先进的方法更真实、更引人注目。更重要的是，由于采用封闭式的解决方案，FastPhotoStyle 生成风格化图像的速度比传统方法快 49 倍。

## 7.2 研究成果

- 提出了一种新的图像风格化方法：FastPhotoSyle，其中：
- 通过渲染更少的结构伪影和不一致样式，从而比艺术风格化算法表现更好；
- 通过不仅合成风格照片中色彩，而且合成风格照片的图案，从而优于照片级真实的风格化算法。
- 实验表明，在风格化效果(63.1%)和光真实感(73.5%)方面，用户更喜欢 FastPhotoSyle 的结果，而不是之前的最先进的技术。
- FastPhotoSyle 可以在 13 秒内合成一张分辨率为 1024 x 512 的图像，而之前最先进的的方法需要 650 秒才能完成相同的任务。

## 7.3 应用前景

- 内容创建者可以从照片级真实的图像风格化技术中获得很大的好处，因为该技术基本上允许你根据适合的内容自动更改任何照片的风格。
- 摄影师们也将受到这项技术的影响。