

中图分类号:

UDC:

学校代码: 10055

密级: 公开

南开大学  
硕士学位论文

结构保持的图像风格迁移

Structure Preserving Image Style Transfer

论文作者 刘笑畅 指导教师 程明明 教授  
申请学位 工学硕士 培养单位 计算机学院  
学科专业 计算机科学与技术 研究方向 计算机视觉  
答辩委员会主席 卫金茂 教授 评阅人 杨巨峰、卢少平 副教授

南开大学研究生院

二〇一九年五月

## 南开大学学位论文使用授权书

本人完全了解《南开大学关于研究生学位论文收藏和利用管理办法》关于南开大学(简称“学校”)研究生学位论文收藏和利用的管理规定,同意向南开大学提交本人的学位论文电子版及相应的纸质本。

本人了解南开大学拥有在《中华人民共和国著作权法》规定范围内的学位论文使用权,同意在以下几方面向学校授权。即:

1. 学校将学位论文编入《南开大学博硕士学位论文全文数据库》,并作为资料在学校图书馆等场所提供阅览,在校园网上提供论文目录检索、文摘及前 16 页的浏览等信息服务;
2. 学校可以采用影印、缩印或其他复制手段保存学位论文;学校根据规定向教育部指定的收藏和存档单位提交学位论文;
3. 非公开学位论文在解密后的使用权同公开论文。

本人承诺:本人的学位论文是在南开大学学习期间创作完成的作品,并已通过论文答辩;提交的学位论文电子版与纸质本论文的内容一致,如因不同造成不良后果由本人自负。

本人签署本授权书一份(此授权书为论文中一页),交图书馆留存。

学位论文作者暨授权人(亲笔)签字: \_\_\_\_\_

20 年 月 日

### 南开大学研究生学位论文作者信息

论 文 题 目	结构保持的图像风格迁移				
姓 名	刘笑畅	学号	2120160419	答辩日期	2019年5月30日
论 文 类 别	博士 <input type="checkbox"/> 学历硕士 <input checked="" type="checkbox"/> 专业学位硕士 <input type="checkbox"/> 同等学力硕士 <input type="checkbox"/> 划 <input checked="" type="checkbox"/> 选择				
学院(单位)	计算机学院	学科/专业(专业学位)名称		计算机科学与技术	
联系电话	18920377020	电子邮箱	liuxiaochang@mail.nankai.edu.cn		
通讯地址(邮编): 天津市海河教育园区同砚路 38 号 (300350)					
非公开论文编号		备注			

注:本授权书适用我校授予的所有博士、硕士的学位论文。如已批准为非公开学位论文,须向图书馆提供批准通过的《南开大学研究生申请非公开学位论文审批表》复印件和“非公开学位论文标注说明”页原件。

## 南开大学学位论文原创性声明

本人郑重声明：所提交的学位论文，是本人在导师指导下进行研究工作所取得的研究成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或者没有公开发表的作品的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

学位论文作者签名： \_\_\_\_\_ 年 月 日

-----

## 非公开学位论文标注说明

(本页表中填写内容须打印)

根据南开大学有关规定，非公开学位论文须经指导教师同意、作者本人申请和相关部门批准方能标注。未经批准的均为公开学位论文，公开学位论文本说明为空白。

论文题目			
申请密级	<input type="checkbox"/> 限制 (≤2 年)	<input type="checkbox"/> 秘密 (≤10 年)	<input type="checkbox"/> 机密 (≤20 年)
保密期限	20 年 月 日至 20 年 月 日		
审批表编号		批准日期	20 年 月 日

南开大学学位评定委员会办公室盖章 (有效)

注：限制 ★2 年 (可少于 2 年); 秘密 ★10 年 (可少于 10 年); 机密 ★20 年 (可少于 20 年)

## 摘要

近来，利用神经网络进行风格迁移展现出惊人的视觉效果，从而受到了广泛的关注。现有的基于神经网络的风格迁移方法按照策略大致可以分为两类：训练前馈卷积神经网络和使用迭代优化。尽管策略不同，但相同的是，这些方法中所用的图像表征都由两部分构成：风格表征和内容表征，而且这些图像表征都是从预训练的分类网络中提取得到。在这种图像生成类问题中，合适的图像表征是产生良好视觉效果的关键。然而，由于分类网络原本是用来做目标检测的，所以从中提取的图像特征通常会集中在中心对象上而忽略其他细节。这样以来，风格迁移的结果会呈现某些共性特点，比如：图像原有的结构遭到破坏，局部细节保留不完整，风格纹理倾向于散布在整幅结果图上，等等。

本文针对现有风格迁移方法不能很好地保留图像结构这一现象，提出用一种通用的策略解决上述问题：设计新的图像表征对原有的风格迁移系统进行修正和完善。并通过两个应用实例对提出的方案加以印证。内容要点包括：

1. 提出了一种基于图像结构表征的风格迁移算法。通过借助从深度预测网络提取得到的全局结构表征和从边缘检测网络提取得到的局部结构表征，该方法可以在进行风格迁移的同时保留图像原有的结构信息和局部细节。在一些对结构变化敏感的图片上（如人脸、建筑），该方法可以避免过度形变和过多杂乱纹理对外观造成的负面影响。用户调研表明，与其他方法相比，本方法的结果普遍更受欢迎。定量分析和比较表明，本方法可以更好地保持图像的结构一致性。
2. 提出了一种基于图像形状轮廓表征的风格迁移算法。通过借助从图像距离场计算得到的形状轮廓特征，该方法可以实现 2D/3D 表面特定区域风格迁移。在剪影图片和三维曲面上的实验结果表明，本方法可以消除指定区域外的纹理元素，将风格迁移的作用域控制在指定轮廓内。
3. 本文工作显示了图像表征对风格迁移结果的影响，凸显了其在风格迁移中的重要性，同时也为风格迁移提供了一种新的思路：通过靶向性地设计合适的图像表征使结果呈现特定特点。

**关键词：**风格迁移；结构保持；神经网络

## Abstract

State-of-the-art neural style transfer methods have demonstrated amazing results by training feed-forward convolutional neural networks or using an iterative optimization strategy. Although different in strategies, the image representations used in these methods all contain two components: style representation and content representation. And they are all extracted from pre-trained classification network. In such image generation task, proper image representations are key factors in generating expressive visual effects. However, because the classification networks are originally designed for object recognition, the extracted features often focus on the central object and neglect other details. As a result, the results will present some common characteristics: the original structure is disrupted, some local details are incomplete, the style textures tend to scatter over the stylized outputs, *etc.*

This paper aims at the phenomenon that current image style transfer methods can not well preserve the image structure, and presents a general strategy to solve the above problems: designing additional representations to correct and amend existing representations, then uses two cases to validate the corresponding schemes. The main contributions are:

1. Propose a structure representation based image style transfer algorithm. By using the global-structure representation extracted from the depth prediction network and local-structure representation extracted from the edge detection network, this method can preserve images' original structure and local details. The tests on images which are sensitive to structure distortion (*e.g.* human faces, buildings) show that this method can avoid the negative effects caused by excessive deformation and littery textures. User studies show that compared with other methods, this method is more popular. Quantitative analysis and comparisons indicate that this method can better keep the structure consistency.
2. Propose a contour shape representation based image style transfer algorithm. By using the contour shape representation computed from the image distance field,

this method can apply style transfer on 2D/3D surfaces. The tests on paper-cut silhouettes and 3D surfaces show that this method can eliminate the texture elements outside the designated area, and control the action scope within the designated contour.

3. The proposed methods demonstrate that image representations will highly influence the stylized results, and highlight the importance of them in style transfer. In the meantime, this paper provides a new idea in style transfer: changing the properties of stylized results by designing appropriate image representations.

**Key Words:** Style transfer; structure preserving; neural networks

## 目录

摘要	I
Abstract	II
第一章 绪论	1
第一节 背景介绍	1
第二节 风格迁移中的图像表征	3
第二章 结构保持的风格迁移	6
第一节 相关工作	6
2.1.1 深度图像表征	6
2.1.2 相似度量方法	9
2.1.3 图像生成网络	9
第二节 方法详述	10
2.2.1 生成网络	11
2.2.2 描述网络	12
2.2.3 网络学习	14
2.2.4 训练细节	14
第三节 实验结果	15
2.3.1 图像比较	15
2.3.2 敏感度分析	18
2.3.3 消融分析	19
2.3.4 速度与内存分析	19
2.3.5 用户调研	20
第四节 结构一致性分析	22
第五节 深入理解结构表征网络	22
2.5.1 权重调整	22
2.5.2 选择原因	23
2.5.3 方法局限	23

第六节 小结 .....	24
第三章 2D/3D 表面的风格迁移 .....	26
第一节 相关工作 .....	27
3.1.1 通用纹理合成 .....	27
3.1.2 结构化纹理编辑 .....	27
3.1.3 基于 CNN 的纹理表征 .....	28
第二节 算法概述 .....	29
3.2.1 深度图案描述 .....	30
3.2.2 轮廓形状表征 .....	31
3.2.3 目标损失函数 .....	33
3.2.4 多尺度合成策略 .....	35
第三节 实验结果 .....	36
3.3.1 区域合成 .....	36
3.3.2 沿曲线合成 .....	37
3.3.3 装饰表面 .....	37
第四节 分析和比较 .....	38
3.4.1 消融分析 .....	38
3.4.2 边缘过渡区 .....	40
3.4.3 缺陷 .....	43
第五节 小结 .....	43
第四章 总结 .....	45
参考文献 .....	46
致谢 .....	51
个人简历 .....	52

## 第一章 绪论

本文考虑的问题是利用神经网络对图像进行风格迁移。广义地说，风格迁移是将两幅图像进行融合，使得结果在风格上与其中一幅图像相似，同时在内容上与另一幅图像保持一致（如图1.1所示）。随着数字媒体行业的快速发展和创意文化产业的渐渐兴起，图像风格迁移技术已被广泛应用于包括艺术创作、游戏场景渲染、新闻广告传媒等在内的多个领域，取得了非常好的效果。这都得益于其在提高图像视觉表现力、增强作品艺术吸引力和丰富作品趣味性上的良好表现。

### 第一节 背景介绍

迄今已经有不少方法取得了令人印象深刻的结果。值得注意的是，在所有图像生成类问题中，使用合适的图像表征是取得良好视觉效果的关键。

受益于其强大的图像表征能力，深度神经网络在问世不久后迅速成为图像迁移的工具，促进了该领域的快速发展。Gatys 等人<sup>[1-3]</sup> 使用从预训练的分类网络 VGG<sup>[4]</sup> 提取得到的图像表征，通过迭代优化生成风格迁移图像。还有一些方法<sup>[5]</sup> 通过采用学习前馈网络的方式来避免缓慢的迭代优化过程。其他研究工作<sup>[6,7]</sup> 考虑了如何提高风格迁移模型的通用性。然而在效果上，所有这些方法很大程度地忽略了在图像原有场景中存在的一些个性特点，尤其是结构特征。

现有方法普遍存在的第一个问题是：结构信息保留不完整。当对具有复杂



图 1.1 风格迁移示例，图片来自 Gatys *et al.*<sup>[2]</sup>.

空间布局的图像进行风格迁移时，风格纹理倾向于均匀地分布在整个图像上，使得整体结构变得不易识别，对于那些场景深度范围较广的图像尤为如此。如图2.1(b)中所示，风格迁移的结果并不完全令人满意。而对于那些具有显著基本特征或对结构变形敏感的图像，均匀分布的纹理会掩盖本已弱化的细节，从而破坏原始结构。

这个问题要归因于图像表征的选择。当前的方法主要基于以下原理：预训练分类网络的中间层特征是一种强大的图像表征。然而，由于这些预训练的分类网络最初是为目标检测设计的，所以其中间层特征通常会集中在主要目标上而忽略其他细节。另外，如<sup>[8]</sup>中指出的，Gatys 等人使用的 VGG 架构是在小尺寸图像（ $224 \times 224$  像素）上训练得到的，所以相应地，其中间层特征尺寸也会偏小。因此，当前这些图像表征不足以表示图像细节和捕获图像结构（见图2.3）。然而，保持必要的细节和结构是达到良好的风格迁移效果所必需的。

现有方法普遍存在的另一个问题是：风格迁移的作用域不可控，当要对特定区域进行风格迁移时，现有方法会将整个风格迁移过程无差别地作用到整幅图像上，从而破坏图像的轮廓结构。这样使得用户无法对一些特定区域（比如 2D 剪影图像，3D 曲面）进行风格迁移，例如达到如图3.1的效果。这个现象仍能从图像表征的层面得到解释，原因在于现有的图像表征没有考虑到去保留这些剪影图像的形状轮廓特征。

本文接下来将用两个风格迁移算法证明，既然上述现象都可以在图像表征的层面得到解释，那就可以尝试用一个通用策略解决上述问题：针对现有图像表征在描述图像某些方面的缺陷，设计新的图像表征对原有系统进行修正和完善，靶向性地使风格迁移结果呈现特定特点。

第一个算法是通过设计结构表征（Structure Representation）以在风格迁移的同时保留图像的全局和局部结构。一些工作<sup>[9, 10]</sup>已经证明在对图像进行艺术化处理时保留图像原有结构可以产生很有吸引力的效果。考虑到深度图可以有效地捕捉场景的全局结构，先引入深度预测网络作为全局结构提取网络，以在风格迁移期间保持图像的整体布局。更进一步，引入边缘检测网络作为局部结构细化网络，与全局结构提取网络协调工作，组成新的结构增强图像表征来控制风格转移效果。这两个网络的共同作用可以使风格迁移效果在全局结构和局部结构之间达到平衡。实验结果表明，在进行风格迁移时，这样做可以有效地保留图像结构和关键细节，因此非常适合用来处理对结构失真敏感的图像（例

如人脸，建筑物)。

第二个算法是通过设计形状轮廓表征 (Contour Shape Representation) 以在 2D/3D 平面特定区域内进行风格迁移。现有方法在进行风格迁移时会将风格纹理无差别地合成到整幅图像上，用户无法自由地选择作用域。本文提出通过计算输入和输出图像之间的距离场，并最小化图像合成过程中的形状重建损失，以一种软约束的方式对形状不一致性进行惩罚。通过这种方式，迭代优化不仅可以保留所需的形状，还可以避免在指定区域外引入视觉瑕疵。实验结果表明，这样做实现了对风格迁移作用域的控制，可以将输入范例图案绘制到任意指定的区域 (或路径)，因此非常适合用来处理剪影图片和 3D 曲面。

总的来说，本文的贡献是：

- 引入结构增强图像表征进行图像风格迁移，在全局结构提取网络和局部结构细化网络的协同作用下保留图像的结构信息；
- 引入形状轮廓表征进行 2D/3D 表面的风格迁移，将风格迁移的作用域控制在指定的区域中；
- 本文工作表明，图像表征在风格迁移中起着非常重要的作用，通过设计合适的图像表征可以靶向地使风格迁移效果呈现特定特点。

## 第二节 风格迁移中的图像表征

上一节中我们提到，在所有图像生成类问题中，合适的图像表征是达到良好视觉效果的关键。本节将简要介绍现有基于神经网络的风格迁移方法中用到的图像表征，具体细节还会在接下来两章的相关工作部分再次提到。

首先再次回顾一下风格迁移要完成的任务：将图像  $A$  的风格迁移到图像  $B$  上得到图像  $C$ ，使得  $C$  在风格上与  $A$  相似，同时在内容上与  $B$  保持一致。

从对问题的描述中可以看出，进行风格迁移时我们至少需要解决以下几个基本问题：

1. 怎么表示一幅图像的内容？
2. 怎么表示一幅图像的风格？
3. 怎么定量计算两幅图像在内容/风格上的相似度？

首先考虑第三个问题，无论是对于内容还是风格，其相似度的计算最终都要归结到距离计算上。现有的相似度度量方法很多，比如计算欧几里得距离，余弦距离，曼哈顿距离等。但这些度量方法的有效性是建立在合适的特征表征上

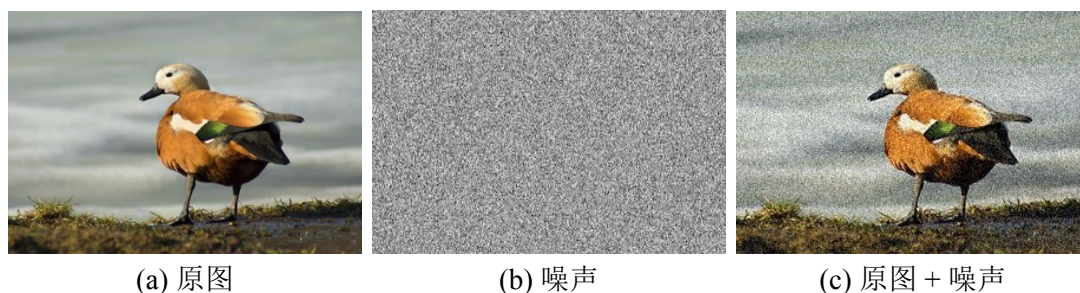


图 1.2 直接用图像本身表征图像内容是否可行，图片来自 PhotograFeed。

的，所以首先要保证所用的表示内容/风格的特征是有效的、稳定的。

拿图像内容的表示以及相似度计算举例来说。如果直接拿图像本身表征图像内容，并直接计算不同图像矩阵间的距离做为内容相似度，会怎么样？首先，这种做法虽然简单，但在一定程度上是可行的，因为毕竟内容一致的图片在外观上不会差太多，其像素的组成和分布也会高度相似。但这一做法的主要弊端是稳定性不够。如图1.2所示，(c)由(a)叠加了噪音(b)得到，(a)和(c)在内容上完全一致，但无论计算这两幅图像的哪种距离，结果都将会非常大，而且其对位移、旋转等也会非常敏感。由此可以看出，选择合适的图像表征是取得理性效果的前提。

实际应用中当然不会直接拿图像本身表征图像内容。在深度学习流行以前，人们通常会以手工设计的方式提取图像特征，这种做法是可行的，提取的特征

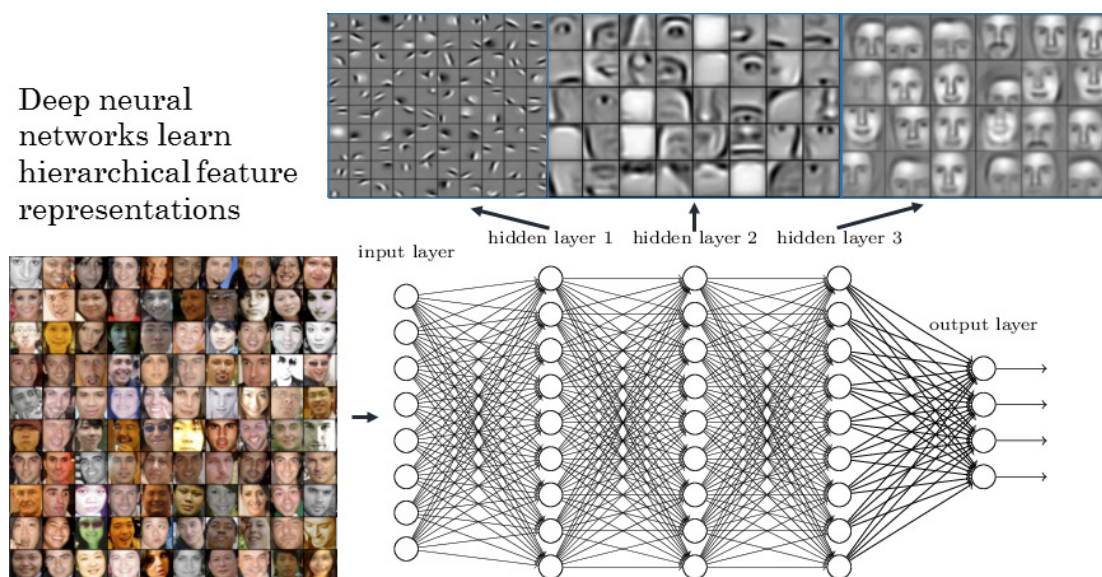


图 1.3 深度神经网络中包含了多层次多级别的图像特征，图片来自 Rspivision。

非常有效，也取得了不错的效果。但与从深度神经网络中提取的特征相比，人工设计的特征在稳定性和表现力上还是有一定差距。如图1.3所示，深度神经网络中包含了非常丰富和全面的多层次、多级别图像特征。正是由于这个特点，神经网络非常适合被拿来去做图像生成类任务，而风格迁移就是其中之一。

在基于神经网络的风格迁移方法中，用到的图像表征全都是从神经网络中提取得到，只是在具体运用上稍有差别。内容表征直接来自于神经网络的中间层特征，不需要进行任何后处理。而风格表征在提取了中间层特征后多了一步后处理：计算每组特征的 **Gram** 矩阵。在相似度度量上，内容相似度计算的是特征间的（平方，归一化）欧几里德距离，风格相似度计算的是两个 **Gram** 矩阵的 **Frobenius** 范数的平方。具体细节将会在后文详述。

当然，没有图像表征是完美的，即使取自神经网络，现有的图像表征在描述图像特性上也会有所欠缺，这也是本文的切入点。在大致了解了风格迁移需要解决的三个问题，以及选用神经网络进行图像表征的原因后，接下来正式介绍本文的第一个工作。

## 第二章 结构保持的风格迁移

上一章中已经提到，现有的基于神经网络的图像风格迁移方法可以产生不错的效果。然而，对于那些场景中包含了多个对象或有着复杂空间布局的输入图片，风格迁移效果并不能完全令人满意（参见图2.1（b）中的示例）。具体表现为风格迁移后的纹理倾向于均匀地分布在整幅图像上，图像布局会因此被破坏。而且前景和背景，以及不同的对象之间的边界变得模糊，从而使得场景中的对象变得不可识别。其中原因在于现有方法中用到的预训练网络最初是被设计用于做对象识别的，因此其对用于风格迁移的图像特征的提取能力也是有限的。为了对这一点进行补充，本章提出利用结构重构损失辅助训练图像生成网络。

考虑到深度图可以有效地反映场景的结构信息和整体布局，边缘图可以在很大程度上反映图片的细节和局部结构，并且在风格迁移后保留内容图像的这些特性有助于保留其语义信息，本章提出了一种新的风格迁移方法：将深度重建损失和边缘重建损失做为额外的损失函数，在保持图像结构的同时进行风格迁移。实验结果表明，所提方法实现了风格迁移并能够保留内容图像的基本布局 and 结构（见图2.1（c），图2.2（f））。

### 第一节 相关工作

如之前的一些工作所述<sup>[1, 2, 5]</sup>，风格迁移中的两个关键问题是：1）如何获得合适的图像风格表征和内容表征；2）如何定量计算两幅图像之间的风格相似度和内容相似度。另外一个影响性能的因素是图像生成网络的结构。

#### 2.1.1 深度图像表征

特征提取是图像表征的关键，如何提取尽可能完整地反映图像的特征尤为重要。传统方法，无论是参数方法<sup>[12-14]</sup>还是非参数方法<sup>[15-17]</sup>，都使用手工设计的图像表征。深度卷积神经网络的发展（CNN）<sup>[18]</sup>打破了传统图像表征的局限。Gatys 等人<sup>[1-3]</sup>使用从预训练的分类器 VGG<sup>[19]</sup>中提取的高级特征表征图像的内容，使用特征的 Gram 矩阵来表征图像风格，利用迭代优化过程将初始化白噪声变为风格迁移图像。其方法，尤其是所提出的图像表征，非常优雅而且有效，并



图 2.1 现有的一些风格迁移方法，Johnson *et al.*<sup>[5]</sup> 的方法倾向于将风格纹理均匀散布在输出结果上，忽略了图像中的重要结构信息。本章提出的方法可以更好地保留图像空间布局（第一行）和主要对象（第二行）的结构。（风格图：The Muse by Pablo Picasso，内容图来自 Pixabay）

且已被许多后续工作采用。但是，迭代过程意味着图片的生成时间会相对较长。

为了避免迭代优化过程带来的时间消耗，一些工作<sup>[5, 20, 21]</sup> 采用对每个风格学习一个对应的前馈网络的策略。除此之外，还有一些工作<sup>[22]</sup> 致力于提高风格迁移的稳定性。还有一些方法<sup>[6, 7, 23]</sup> 主要关注如何提高模型的通用性或处理速度<sup>[11, 24]</sup>，这些方法试图仅用一个模型实现多种风格迁移，并进一步加快处理速度。与此同时，一些真实感渲染方法<sup>[25-27]</sup> 也广泛使用深度图像表征。在所有方法中，Johnson 等人的方法<sup>[5]</sup> 因其快速的速度和令人满意的质量脱颖而出。通过预训练前馈网络而不是像<sup>[2]</sup> 中那样直接优化损失函数，其方法在对新输入图像进行风格迁移时效率比<sup>[2]</sup> 要高出 2 个数量级。最后，非常推荐有兴趣的读者参考一篇对基于深度学习的风格迁移方法的综合调研<sup>[28]</sup>。

这样看来，利用神经网络进行风格迁移似乎正变得越来越完善。但实际上，由于过分关注模型的性能，风格迁移结果的质量并没有得到显著提高，而当前所用的图像表征正是造成这一现象的一个主要原因。如图 2.3 所示，现有方法所

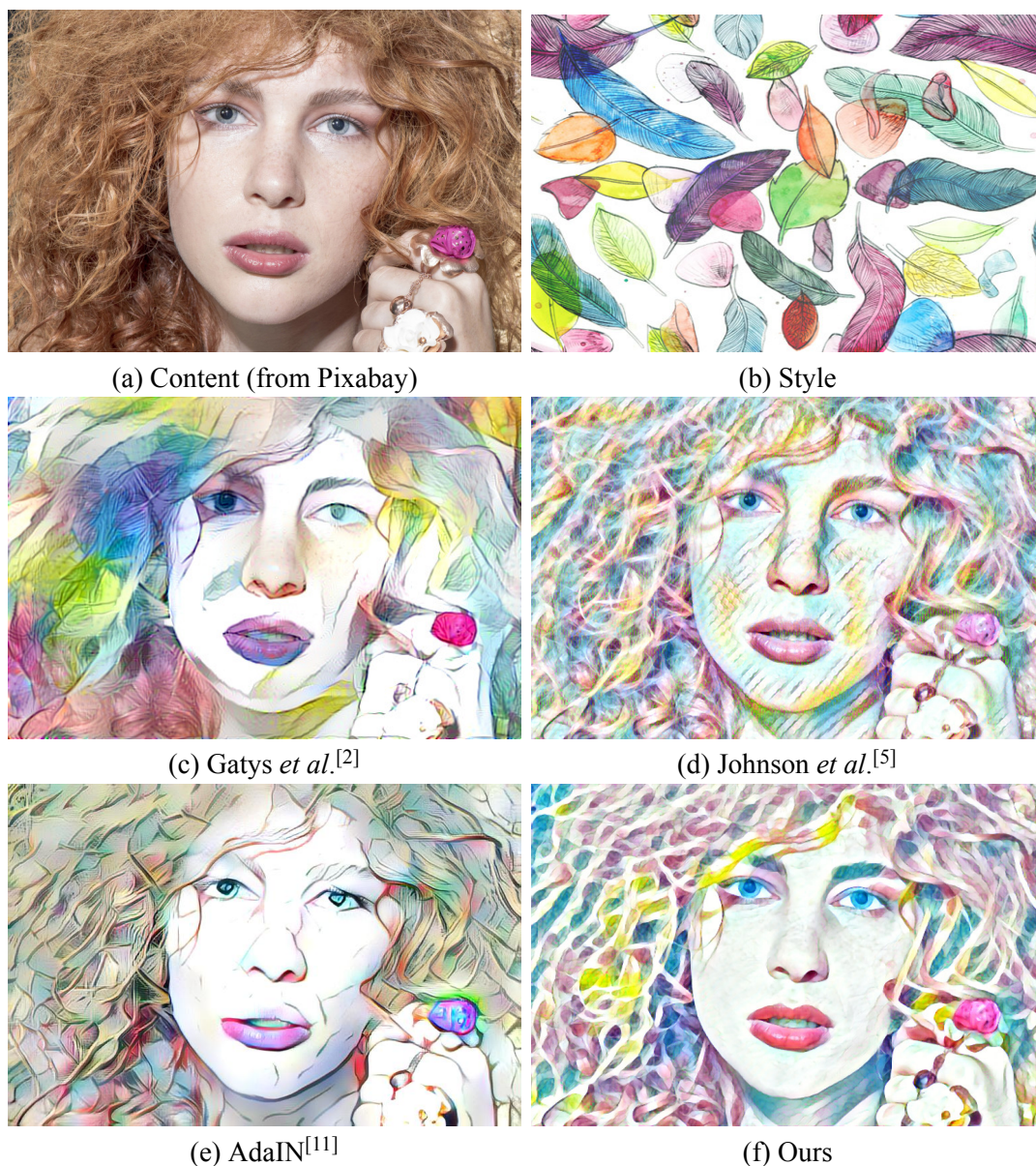


图 2.2 你更喜欢哪种风格迁移效果？人脸对结构变形非常敏感，缺失过多的结构信息甚至会丑化它们。(b) 中的结果由慢速优化过程生成。(c) 和 (e) 均属于快速方法。这三种方法都是仅在 VGG 分类网络的协助下得到图像表征，而它们也存在一些共性问题：风格纹理散布在结果上，破坏了图像的原始结构，而且也丢失了一些内容细节。(f) 显示的是本方法使用结构增强图像表征后的结果。

采用的图像表征并不能很好地捕捉图像细节和结构。于是本文提出用一种新的结构增强表征来弥补这点不足，并在<sup>[5]</sup>方法的基础上构建自己的风格迁移系统。

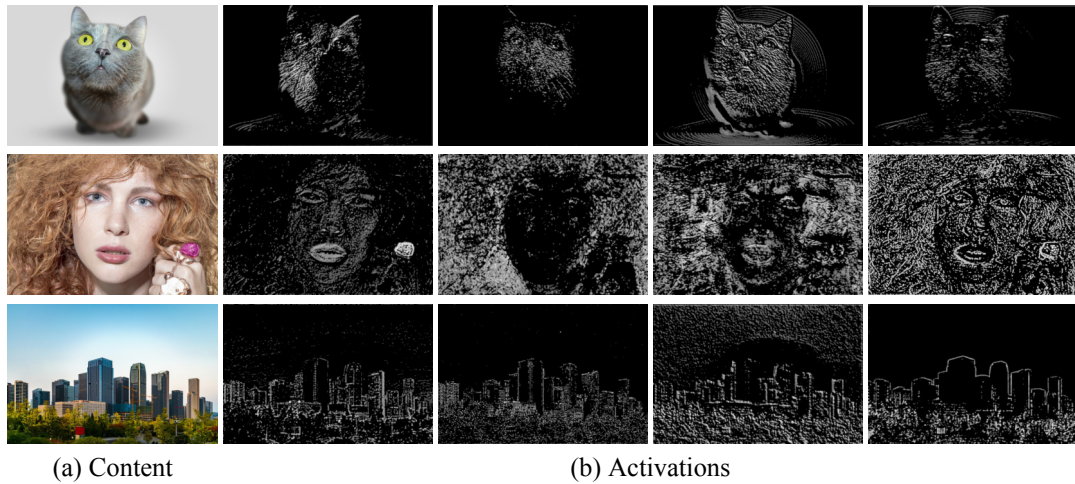


图 2.3 当前风格迁移方法所用的图像表征: 将图像送入 VGG-16 并对中间层响应 ( $relu1\_2$ ,  $relu2\_2$ ,  $relu3\_3$  和  $relu4\_3$ ) 进行可视化。查看这些激励可以发现它们并不能很好地捕捉图像的全局和局部结构。

### 2.1.2 相似度量方法

在获得了合适的图像表征后, 接下来是要找到可以定量计算相似度的方法。通常来说, 风格迁移中的相似性度量是计算图像表征的维度距离。距离越小则意味着相似度越高, 反之亦然。

目前有许多度量相似性的方法, 在风格迁移中, 欧几里德距离通常用来计算内容相似度, 这点也很符合人的直观感受。对于风格相似度, 大多数方法选择计算 Gram 矩阵之间的 Frobenius 范数。最近, Li 等人<sup>[29]</sup> 证明风格迁移可以看作是一个将内容图像的空间分布向风格图像的空间分布对齐的过程, 匹配图像的特征图可以看作是用二阶多项式内核最小化最大均值差异 (maximum mean discrepancy, MMD)。而我们刚才提到的两个相似性度量方法正是最小化 MMD 的两种特殊情况。当然还有使用其他损失函数的方法, 例如 MRF 损失<sup>[30, 31]</sup>, 直方图损失<sup>[32]</sup> 等。本文使用的相似性度量方法详见第二节。

### 2.1.3 图像生成网络

之前提到过, 基于迭代优化的风格迁移方法速度较慢。为了减少计算负担并加快处理速度, Johnson 等人<sup>[5]</sup> 提出构建一个前馈神经网络。通过预先训练生成网络而不是直接实时优化损失函数, 将对新输入图像进行风格迁移的效率提高了 2 个数量级。受其影响, 之后的工作<sup>[6, 11, 20]</sup> 大多采用类似的方法。

最近, 由 Iandola 等人设计的 SqueezeNet<sup>[33]</sup> 在 ImageNet 上实现了 AlexNet

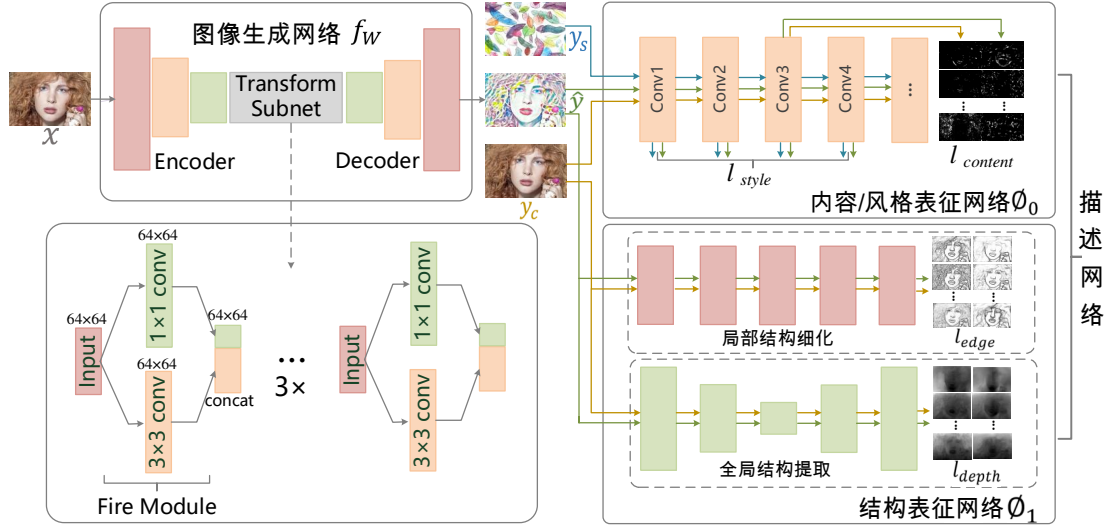


图 2.4 网络架构概述：左侧显示的是生成网络  $f_W$ ，其通过映射  $\hat{y} = f_W(x)$  将输入图像  $x$  转换为  $\hat{y}$ 。右侧显示的是描述网络，描述网络用于定义四种损失函数： $l_{style}$ 、 $l_{content}$ 、 $l_{edge}$  和  $l_{depth}$ ，其中  $l_{style}$  和  $l_{content}$  基于风格/内容表征网络  $\phi_0$ ，分别对应于风格损失和内容损失。 $l_{edge}$  和  $l_{depth}$  基于结构表征网络  $\phi_1$ ，分别对应于边缘损失和深度损失。 $y_s$  和  $y_c$  分别代表风格目标和内容目标。

级的精度，而参数量只有原有的 2%。这启发我们用与其类似的方式设计一个轻量级的图像生成网络，以使得提出的框架能够有效地用于视频的风格迁移。

## 第二节 方法详述

如图 2.4 所示，本文的系统主要由三个部分组成：两个图像表征网络  $\phi_0$  和  $\phi_1$ ，以及一个图像生成网络  $f_W$ 。表征网络用来定义四个损失函数： $l_1$ 、 $l_2$ 、 $l_3$  和  $l_4$ ，其中  $l_1$  和  $l_2$  基于  $\phi_0$ ，分别对应于风格损失和内容损失，表示为  $l_{style}$  和  $l_{content}$ 。 $l_3$  和  $l_4$  基于  $\phi_1$ ，分别对应于深度损失  $l_{depth}$  和边缘损失  $l_{edge}$ 。图像生成网络是由  $W$  参数化的卷积神经网络，它通过映射  $\hat{y} = f_W(x)$  将输入图像  $x$  变为输出图像  $\hat{y}$ 。每个损失函数计算一个标量值  $l_i(\hat{y}, y_i)$ ，用于测量输出图像  $\hat{y}$  和目标图像  $y_i$  ( $i = 1, 2, 3, 4$  对应于内容，风格，深度和边缘图像) 之间的相似度距离。最后使用随机梯度下降算法最小化目标损失函数（四部分的加权和）来训练图像生成网络：

$$W^* = \arg \min_W \mathbf{E}_{x, \{y_i\}} \left[ \sum_{i=1}^4 \lambda_i l_i(f_W(x), y_i) \right] \quad (2.1)$$

四种损失函数分为两类：感知损失 ( $l_{style}$  和  $l_{content}$ ) 和逐像素损失 ( $l_{depth}$  和  $l_{edge}$ )。感知损失函数基于从预训练网络提取的高级特征，用于衡量图像之间的高级感知和语义差异。与逐像素损失相比，感知损失可以更加有效稳定地衡量图像相似性。因为根据最近的一些工作<sup>[19, 34]</sup>，预训练的图像分类卷积网络已经学会对感知和语义信息进行编码。相反，当网络输出需要匹配一个明确的目标时，逐像素损失会更加合适，比如深度和边缘损失，因为我们可以得到内容和合成图像的相对深度和边缘图。在本方法中， $\phi_0$  是预训练的图像分类网络， $\phi_1$  由单图像深度感知网络<sup>[35]</sup> 和整体嵌套边缘检测网络<sup>[36]</sup> 组成。

在训练阶段，我们将每个输入图像  $x$  通过图像变换网络  $f_W$  并获得合成图像  $\hat{y}$ 。为了计算得到损失函数，输入图像  $x$  还被用做内容目标  $y_c$ ，风格图像被当做风格目标  $y_s$ 。通过在损失网络  $\phi_0$  中将每个  $\hat{y}$  与  $y_s$  进行比较得到风格重建损失  $l_{style}$ ，通过在相同的表征网络  $\phi_0$  中将每个  $\hat{y}$  与  $y_c$  进行比较得到内容重构损失  $l_{content}$ 。通过计算  $\hat{y}$  和  $y_c$  经过附加的深度预测和边缘检测网络  $\phi_1$  后的输出差异，得到深度重建损失  $l_{depth}$  和边缘重建损失  $l_{edge}$ ，以使得风格迁移后图像的深度和边缘与原内容保持一致。

### 2.2.1 生成网络

系统的三个部分构成两个网络：生成网络 (Generator Network) 和描述网络 (Descriptor network)。在全局结构提取网络和局部结构细化网络的共同作用下，本文使用基于描述网络内部数据统计的结构增强图像表征 (Structure-Enhanced Image Representation) 来训练生成网络。

生成网络包括三个子网：编码子网，变换子网和解码子网。我们使用生成网络来对输入图像进行风格迁移。通常来说，网络中的每个层等同于一个非线性滤波器组。而且随着层位置的增加，滤波器组的复杂性也随之增加。输入图像  $x$  也就被网络中每一层对该图像的滤波响应编码。

#### 输入和输出

在训练阶段，输入和输出是尺寸为  $256 \times 256$  的 3 通道彩色图像。由于图像变换网络是全卷积网络，在测试阶段，系统对输入图像的大小没有限制。

#### 编码器和解码器

编码器子网的主要功能是将高维输入图像映射到低维空间。这样以来可以大大减少后续的计算量。具体来说，我们使用一个 stride-1 卷积和两个 stride-2

卷积对输入图像进行下采样。对应地，在解码阶段，解码器子网将变换子网的输出还原为原始大小的图像。

经过这些处理步骤之后，图像的尺寸被保留，但随之带来两个优点：一方面，在下采样之后，在计算成本相同的情况下，我们可以使用更大的网络。例如，对尺寸为  $H \times W \times C$  的输入，一个有  $C$  个滤波的  $3 \times 3$  卷积的计算成本等于使用  $DC$  滤波器的  $3 \times 3$  卷积，其中  $D$  是下采样因子。另一方面，当具有相同的层数时，下采样可以提供更大的有效感受野。例如，如果没有下采样，每增加一个  $3 \times 3$  卷积层，有效感受野大小将增加 2。在下采样  $D$  倍后，有效感受野大小增加到  $2D$ 。而通常来说，感受野越大，风格迁移结果会越好。

### 转换子网

转换子网是生成器网络的核心，主要负责对图像进行转换编码。与现有方法不同，我们将 Fire 模块<sup>[33]</sup> 引入到架构中以减少计算并提高效率。通过使用 Fire 模块，Iandola 等人在参数减少 50 倍的情况下，在 ImageNet 上实现了 AlexNet 级精度。我们采用类似的方式来构建本方法的模型，以便其可以应用于视频。Fire 模块包括：挤压卷积层（squeeze convolution layer，仅包含  $1 \times 1$  滤波器），馈入由  $1 \times 1$  和  $3 \times 3$  卷积滤波器混合组成的扩展层（expand layer）。我们在图 2.4 中说明了这一点。设计这种架构的主要策略是：

1. 减少使用  $3 \times 3$  filter，用  $1 \times 1$  对其进行替换；
2. 减少  $3 \times 3$  filter 的输入通道数。

采用这个策略的原因在于：一个由  $m$  个输入通道数为  $n$  的  $3 \times 3$  filter 组成的层的参数总量是： $m \times n \times 3 \times 3$ 。因此，为了减少参数数量，可以从减少  $m$  和  $n$  入手。借助此策略，在效果相同的情况下，网络的参数数量将大大减少。

### 2.2.2 描述网络

描述网络由两个子网组成：内容和风格表征网络  $\phi_0$  和结构表征网络  $\phi_1$ 。本方法中用到的图像表征均来自描述网络。

#### 内容和风格表征网络

按照<sup>[2, 5, 20]</sup> 中的方法，我们使用预训练的 VGG<sup>[19]</sup> 作为  $\phi_0$  来定义两个损失函数  $l_{content}$  和  $l_{style}$ ，分别对应内容表征和风格表征。具体地， $l_{content}$  被定义为从  $\phi_0$  所选层激活的（平方，归一化）欧几里德距离， $l_{style}$  是两个 Gram 矩阵的 Frobenius 范数的平方。假设 VGG 网络的第  $l$  层具有  $N_l$  个不同的滤波器，其中

每个特征响应的尺寸为  $H_l \times W_l$ ， $H_l$  和  $W_l$  分别是  $l$  层中的特征映射的高度和宽度。那么该层中的响应可以用如下矩阵表示：

$$F_l \in \mathcal{R}^{(H_l \times W_l) \times N_l} \quad (2.2)$$

矩阵的每个值  $F_{(i,j),k}^l$ ，是  $l$  层中第  $k$  个滤波器在位置  $(i,j)$  处的激活值。

于是  $l^{\text{th}}$  层中  $x_1$  和  $x_2$  之间的内容差异是：

$$l_{\text{content}}(x_1, x_2) = \frac{1}{H_l W_l N_l} \|F_l(x_1) - F_l(x_2)\|_2^2 \quad (2.3)$$

$x_1$  和  $x_2$  在该层的风格差异是：

$$l_{\text{style}}(x_1, x_2) = \|G_l(x_1) - G_l(x_2)\|_F^2 \quad (2.4)$$

其中  $G_l$  是 Gram 矩阵（尺寸为  $N_l \times N_l$  的对称矩阵）。 $G_{cc'}^l$  为  $l^{\text{th}}$  层中  $c$  和  $c'$ th 矢量化特征映射的归一化内积：

$$G_{c,c'}^l(x) = \frac{1}{H_l W_l N_l} \sum_{h=1}^{H_l} \sum_{w=1}^{W_l} F_{(h,w),c}^l(x) F_{(h,w),c'}^l(x) \quad (2.5)$$

$l_{\text{content}}$  和  $l_{\text{style}}$  这两种损失函数基于高级特征，也被称为感知损失<sup>[5]</sup>。根据最近的一些工作<sup>[4, 34]</sup>，图像分类网络已经学会了对感知和语义信息进行编码。所以与逐像素损失相比，感知损失在计算图像相似度上有更强的鲁棒性。

### 结构表征网络

结构表征网络  $\phi_1$  由两个子网络组成：全局结构提取（global structure extraction）网络和局部结构细化（local structure refinement）网络。它们旨在弥补内容和风格表征网络在捕获和保持图像结构上的不足。

边缘结构特别适合用来表征局部结构，因为它们可以说是无处不在。我们采用整体嵌套边缘检测（Holistically-nested edge detection, HED）系统<sup>[36]</sup>作为局部结构细化网络。HED 是一个端到端的边缘检测系统，它以图像作为输入并直接生成边缘图图像作为输出。HED 系统可以有效地生成感知多级特征，并且通过组合多尺度和多级视觉响应，在图像到图像学习方面显示出非常不错的效果。在本文构建的系统中，我们使用 HED 系统的边缘响应来表征图像的局部结构。 $l_{\text{edge}}$  代表两个图像  $x_1$  和  $x_2$  的局部结构差异，表示为输入图像在局部结构细化网络  $\mathcal{E}$  中第  $k$  层的激活的欧几里德距离：

$$l_{\text{edge}}(x_1, x_2) = \|\mathcal{E}_k(x_1) - \mathcal{E}_k(x_2)\|_2^2 \quad (2.6)$$

全局结构提取网络来自单图像深度感知网络<sup>[35]</sup>，该网络整个图像作为输入并直接预测每个像素的深度。深度图是图像的重要特征之一，其中包含了对象的3D特征信息，能够在很大程度上反映图像的全局结构。 $l_{depth}$ 的计算方法与 $l_{edge}$ 相同。结构重构损是两部分的加权和，具体如下所示：

$$l_{structure} = \alpha \cdot l_{depth} + \beta \cdot l_{edge} \quad (2.7)$$

在全局结构提取网络和局部结构细化网络的共同作用下，图像的局部和全局结构被有效地捕获，并且在最终结果中也有很明显的体现（见图2.6）。第五节中进一步提供了有关结构表征网络的更多信息。

### 2.2.3 网络学习

鉴于风格迁移没有特定的评判标准，本文选择最小化从描述网络得到的差值。

如之前所述，内容差异 $l_{content}$ 和风格差异 $l_{style}$ 由预训练的分类网络 $\phi_0$ 得到，结构重建损失 $l_{structure}$ 基于结构表征网络 $\phi_1$ 。

假设 $f$ 表示由 $W$ 参数化的生成网络，其通过映射 $\hat{y} = f_W(x_c)$ 将输入图像 $x_c$ 变换为输出图像 $\hat{y}$ 。通过使用随机梯度下降最小化加权和损失来调整生成网络的参数 $W$ ：

$$\begin{aligned} W^* = \operatorname{argmin}_W \mathbf{E}_{y_c, y_s} [ & \lambda_{content} l_{content}(\hat{y}, y_c) \\ & + \lambda_{style} l_{style}(\hat{y}, y_s) \\ & + \lambda_{structure} l_{structure}(\hat{y}, y_c) \\ & + \lambda_{TV} l_{TV}] \end{aligned} \quad (2.8)$$

其中 $y_c$ 和 $y_s$ 分别是内容目标和风格目标。 $l_{TV}$ 是总变差正则项，在许多工作<sup>[5, 6, 37]</sup>中都有使用，为的是增加生成图像的平滑度。

### 2.2.4 训练细节

本文选择Microsoft COCO<sup>[38]</sup>作为训练数据集，使用VGG-16的 $relu3\_2$ 层激活计算 $l_{content}$ ，选择 $relu1\_2$ ， $relu2\_2$ ， $relu3\_3$ 和 $relu4\_3$ 层计算 $l_{style}$ ，结构表征网络的输出层计算结构重建损失 $l_{structure}$ 。使用Adam算法<sup>[39]</sup>进行参数优化，学习率为 $1 \times 10^{-3}$ 。式2.7中 $\alpha$ 和 $\beta$ 都默认是5，式2.8中 $\lambda_{content}$ ， $\lambda_{style}$ ， $\lambda_{structure}$ 和 $\lambda_{TV}$ 分别是1， $5 \times 10^{-2}$ ， $1 \times 10^{-2}$ 和 $1 \times 10^{-3}$ 。如图2.4所示，全局

和局部网络是联合训练的，生成网络的内容和输出被馈送到结构表征子网中。在训练过程中，内容/风格表征子网和结构表征子网保持固定，生成网络的参数不断更新。在全局和局部子网的影响下，优化完成之后，生成网络的参数保持固定，而它也随之获得了在风格迁移的同时保持结构一致的能力。

### 第三节 实验结果

本节将详述训练细节，并将本文方法与其它基于 CNN 的风格迁移方法和一些真实感渲染方法<sup>[2, 5, 11, 21, 26, 27, 30]</sup> 进行比较。出于公平性考虑，一些结果直接来自所比较的论文。

#### 2.3.1 图像比较

图2.5, 2.6和2.7中展示了一些对比结果。从结果看来，这些方法各有各的特点，本方法结果给人第一印象应该是具有强烈的结构感。关键细节得到保留这个特点也得到了充分反映。

首先，本方法能够使图像前景和背景呈现相对较高的对比度，这是因为全局结构提取网络提供了丰富的全局结构信息，有助于区分不同深度的对象。需要特别指出的是，这个特点使得本方法非常适于处理特写镜头，如模特和鸟（见图2.6）。其他方法忽略了图像的原始结构，风格纹理会均匀地分布在结果上，场

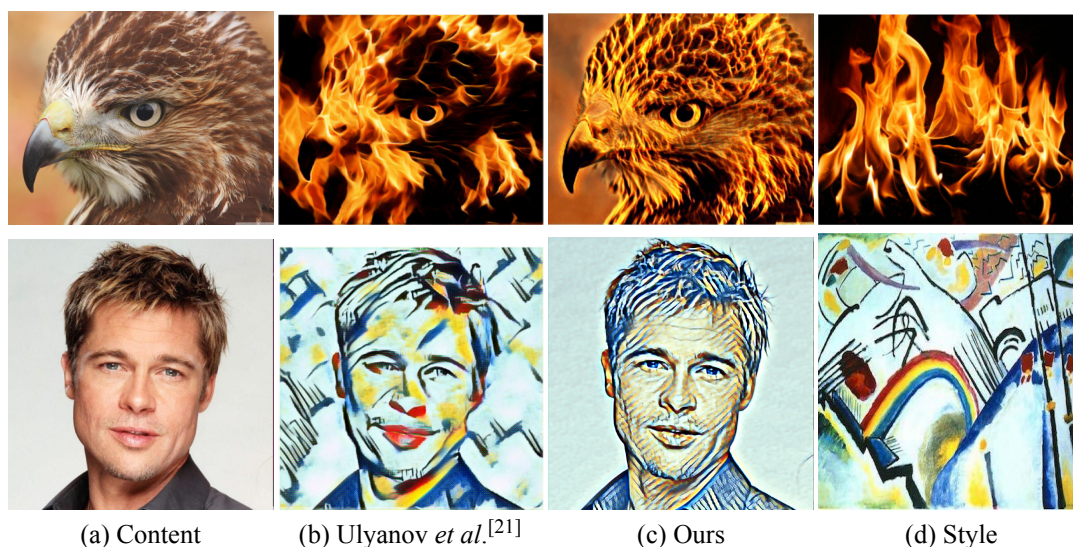


图 2.5 与 Ulyanov *et al.*<sup>[21]</sup> 比较。其专门引入 instance-normalization 层来提升网络性能并捕捉抽象特征。由于强调了结构信息，本方法表现抽象感觉的能力有所欠缺。

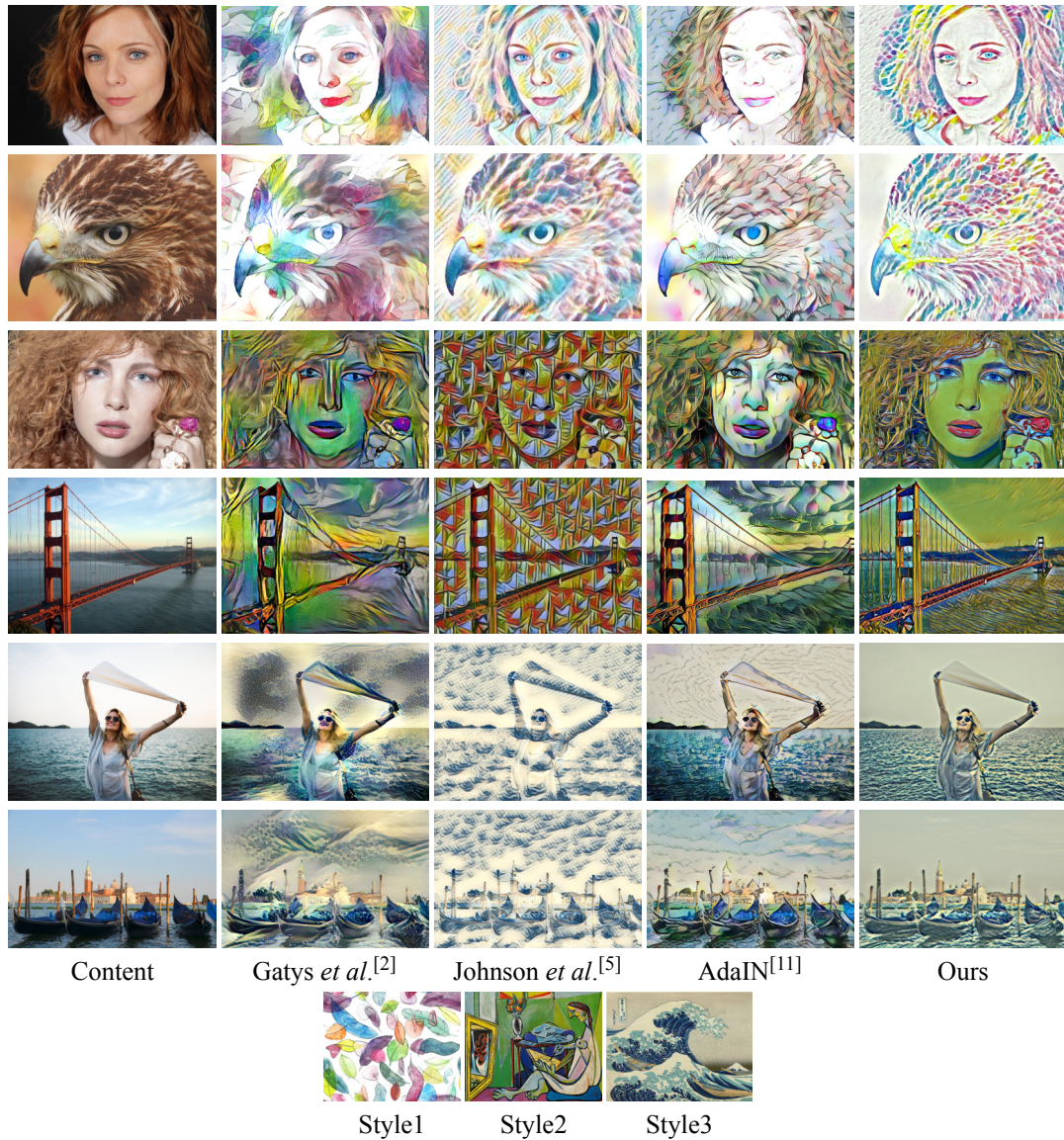


图 2.6 与其他风格转移方法的比较。与其他方法相比，本方法结果很好地保留了图像细节和结构。在人像上的效果更为明显，因为人脸对结构变形非常敏感。

景的结构感也随之弱化。这使得它们不适合处理那些包含丰富空间信息的图像。

其次，即使在同一片前景/背景区域内，本方法对不同区域的结构迁移效果也不同，使得结果呈现整体的平衡与和谐感。例如，本方法几乎没有在模特的脸部区域引入纹理（参见图2.2和图2.6），对五官特征（眼睛，嘴巴等）做了一些小的改动，但是格外强调了头发部分。这是因为这些区域的局部边缘细节彼此不同（头发部分丰富，脸部相对较少），局部结构很好地捕捉了这一特征。通过这种方式，本方法在保持肖像外观的同时产生风格迁移的效果。相反，在<sup>[5, 11]</sup>的

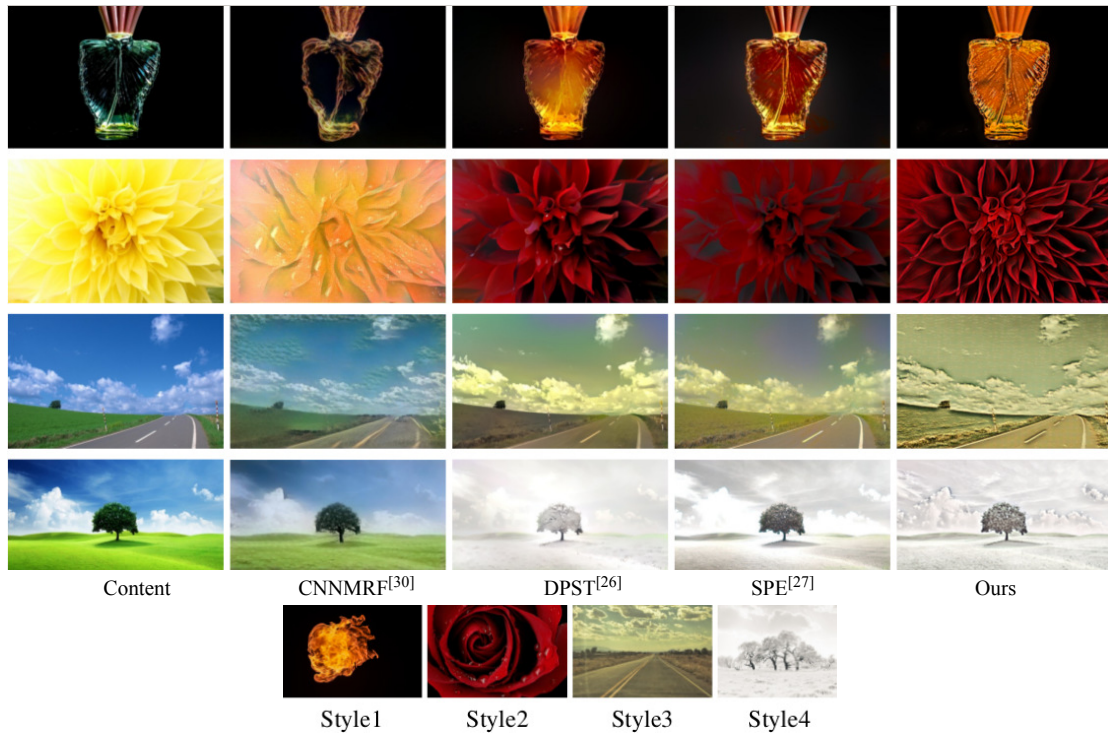


图 2.7 与真实感风格迁移 (photorealistic style transfer) 方法比较。方法<sup>[26, 27]</sup> 专为内容和样式图像共享高度相似的布局和语义信息这种特殊情况设计。本方法在这些情况下也表现良好。

结果中，密集覆盖的纹理使得人脸变得不整洁，甚至会丑化它。出于同样的原因，当处理其他图像时，如吊船（见图2.6），天空和海洋虽然都属于背景，但它们的处理方式却不不同：卷曲的波浪被渲染为不同的风格，但天空区域仍然清晰。与此同时，需要注意到，对结构的强调自然会导致抽象感的缺失。与一些专门为保存抽象感的方法相比，这一点变得更加明显（见图2.5）。本文在2.5.3节中进一步讨论了这个缺陷。

此外，本文还对比了一些真实感渲染方法（见图2.7）。其中<sup>[26, 27]</sup> 是专门为内容和风格图像共享高度相似的布局和语义成分这种情况设计的。结果表明，本文的方法也能很好地适用这种情况。以上所有特点都归结于使用了结构增强图像表征，其中两个部分分别起着不同的作用：

1. 全局结构有助于保持整体布局;
2. 局部结构主要关注局部信息并能很好地捕捉微小特征，可以在全局结构的基础上进一步改进局部细节。

这两种结构相互补充，正是它们的共同作用促进了视觉效果的提升。

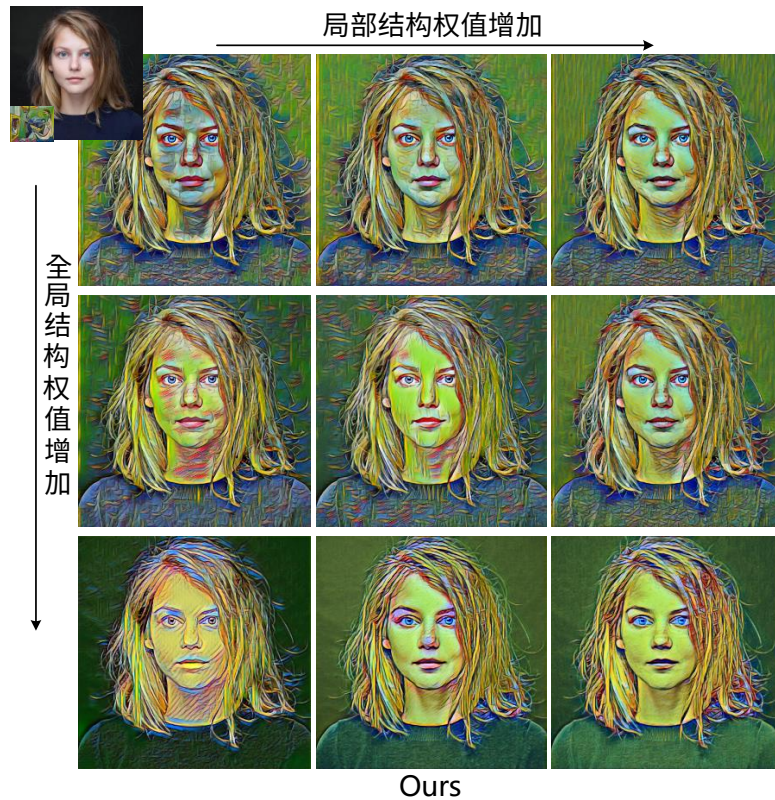


图 2.8 局部和全局结构不同组合下的结果。整体趋势是：随着局部结构权重的增加，面部变得更加干净和平滑，杂乱边缘的数量不断减少。随着全局结构权重的增加，图像前景变得越来越突出。

### 2.3.2 结构重构损失的敏感度分析

结构重构损失包括两部分：局部结构损失和全局结构损失。它们的不同组合会产生不同的艺术效果。

我们尝试了几种不同的结构损失，具体结果如图2.8所示。 $l_{edge}$  的权重  $\alpha$  从左到右增加， $l_{depth}$  的权重  $\beta$  从上到下增加。可以发现：1) 随着  $\alpha$  的增加，面部和背景变得更加干净和光滑，杂乱边缘也随之减少；2) 随着  $\beta$  的增加，前景变得越来越明显。当  $\alpha$  和  $\beta$  的比例适当时，可以达到非常好的效果。

简而言之，本文方法提供了一种可调节的方式，以便在图像风格迁移时获得更好的控制。通常来说，有着更高  $l_{depth}$  权重的结构增强损失更适合用来处理风景和特写镜头；而  $l_{edge}$  权重较大的则更适合处理肖像。

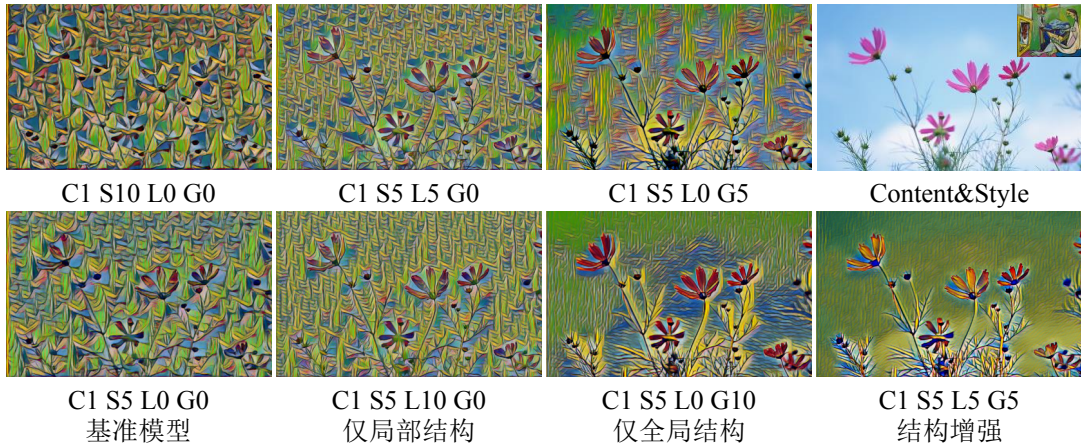


图 2.9 消融分析 (C: 内容 (content) S: 风格 (style) L: 局部结构 (local-structure) G: 全局结构 (global-structure), 数字代表权重)。第一列是基本模型的结果, 在图像表征中不使用任何结构表征。第二列仅使用局部结构, 第三列仅使用全局结构。最后一列使用了局部和全局结构。

### 2.3.3 消融分析

为了更直观地验证结构增强图像表征在结果中所起的作用, 本部分进行了消融分析, 结果见图2.9。由于为了确保处理速度, 模型在网络结构上做了一定牺牲, 因此不使用结构表征的基本模型的结果不尽人意。而这也在另一方面反映了结构增强表征所起的巨大作用。

如果只使用局部结构表征 (参见第二列), 局部的杂乱边界会减少, 但整体结果仍然给人一种很混乱的感觉。如果仅使用全局结构表征 (参见第三列), 结果的整体空间感会得到增强, 但一些局部地方的问题仍需要解决。如果同时结合局部和全局结构, 则整体效果会得到极大改善。

### 2.3.4 速度与内存分析

为了凸显本方法在处理速度和占用内存方面的优势, 本文将所提方法与其它几种<sup>[2, 5, 11, 20]</sup>进行了比较, 比较结果见表2.2。总的来说, 在这几种方法中, 本文方法的处理速度是最快的, 确保可以用来实时处理视频。同时表2.1展示了几

表 2.1 平均 GPU 内存消耗。在 Titan X GPU 上测试, batch-size 大小为 1 且输入图像尺寸为  $768 \times 768$ 。

Methods	[2]	[5]	[11]	Ours
GPU memory	3380 MiB	665 MiB	8869 MiB	<b>502 MiB</b>

表 2.2 速度比较。<sup>[2]</sup> 的速度比较慢,<sup>[5, 11]</sup> 可以实时处理尺寸为  $512 \times 512$  的图像。与这些方法相比, 本文方法相对更快。所有的结果都在 Titan X 12GB GPU 上测试得到。

Methods	Image Size		
	$256 \times 256$	$512 \times 512$	$1024 \times 1024$
Gatys <i>et al.</i> <sup>[2]</sup> (500 iterations)	15.86s	54.85s	214.44s
Johnson <i>et al.</i> <sup>[5]</sup>	0.015s	0.05s	0.21s
Ulyanov <i>et al.</i> <sup>[20]</sup>	0.021s	0.046s	0.145s
AdaIN <sup>[11]</sup>	0.018s	0.065s	0.275s
Ours	<b>0.008s</b>	<b>0.023s</b>	<b>0.12s</b>

种不同方法在处理单张  $768 \times 768$  像素图像时所需的内存。

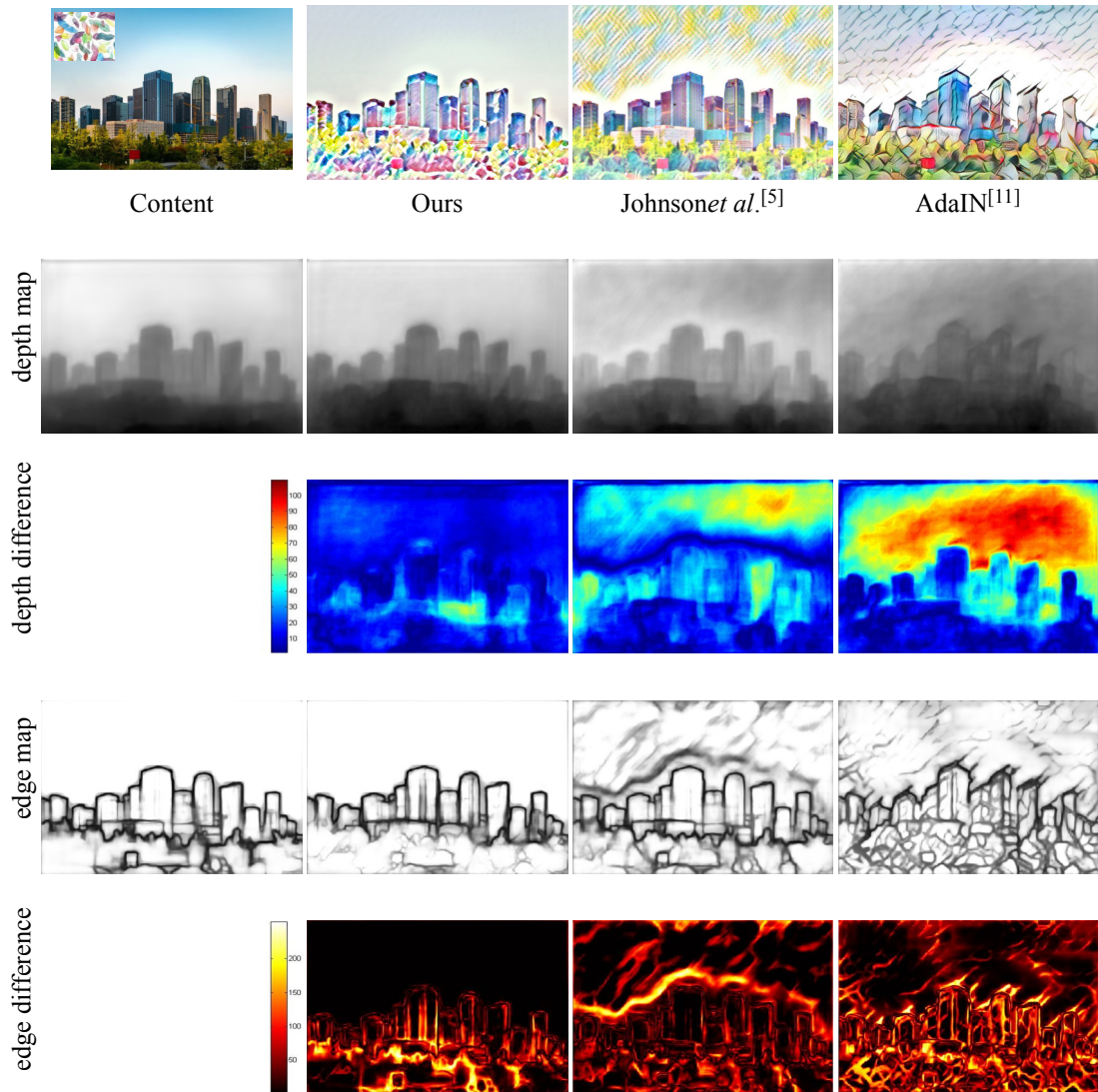
### 2.3.5 用户调研

因为对图像风格迁移进行美学评估是一项非常主观的任务, 本文选择进行用户调研以了解不同用户对不同效果的喜好情况。选择使用的是“Sojump”在线问卷和投票平台, 受访者包括两组: 12 岁以下的学生 (108 人) 和 18 至 30 岁的成年人 (112 人)。每个问卷包含 9 组图片, 要求参与者投票选出他们喜欢的风格迁移结果 (所有参与者) 和具有更强结构一致性的结果 (仅限成人, 因为我们发现大多数学生对此没有明确的概念)。最后计算每个算法被选中的百分比, 将其作为每个算法的偏好分数。

表 2.3 显示了用户调研的结果, 总体而言, 学生和成人的调查结果非常相似, 本方法的风格迁移效果更受欢迎, 而且成年参与者在在本方法可以更好地保持结构一致性这一点上基本达成了共识。

表 2.3 用户调研

Methods	Favorite Stylization			Structural Consistency
	Pupils	Adults	Overall	
Gatys <i>et al.</i> <sup>[2]</sup>	19.14%	12.70%	15.86%	12.81%
Johnson <i>et al.</i> <sup>[5]</sup>	20.99%	18.65%	19.80%	11.88%
AdaIN <sup>[11]</sup>	17.90%	21.83%	19.90%	10.00%
Ours	<b>41.98%</b>	<b>46.83%</b>	<b>44.44%</b>	<b>65.31%</b>



Methods	Content	Depth Map		Edge Map	
	SSIM	SSIM	RMSE	SSIM	RMSE
Ours	<b>0.629</b>	<b>0.956</b>	<b>16.716</b>	<b>0.808</b>	<b>43.656</b>
Johnson <i>et al.</i> <sup>[5]</sup>	0.392	0.939	33.141	0.755	50.576
AdaIN <i>et al.</i> <sup>[11]</sup>	0.394	0.895	53.906	0.582	56.813

图 2.10 结构一致性比较。第二行是使用<sup>[35]</sup>的深度预测结果，第三行是使用<sup>[40]</sup>的边缘检测结果，第二行和最后一行是相应的差异图。结果表明本方法的结果很好地保留了场景的整体深度结构。而其它结果则引入了一些微小的边缘，这些边缘遍布整幅图像。下方表格显示了结果在两种指标下的度量误差（SSIM<sup>[41]</sup>，RMSE）。SSIM（结构相似性）越高越好，RMSE 越低越好（均方根误差）。

## 第四节 结构一致性分析

需要指出，本方法并不是保留绝对精确的结构，而是尽量保持原始图像和风格迁移图像之间的结构一致性。这一点是在全局结构提取网络和局部结构细化网络的共同下实现的。在训练阶段，这些网络根据风格迁移效果提供实时反馈。

除了直观的视觉感受，更值得从数据层面验证所提出的网络是否能够保持结构一致性。目前，本方法的结果保持了合理的深度和边缘图，具有一致的空间和结构感，如图2.10示，通过比较原始内容和风格迁移后图像的深度和边缘图，可以发现本方法的结果比其他方法可以更好地恢复深度和边缘图，深度差和边缘差异图可以更直观地反映这一点。此外，本文还使用 SSIM<sup>[41]</sup> 和 RMSE 来计算结果图像和原始图像之间的相似性。SSIM 是用来测量两个图像之间的结构相似性的指标，当两个图像几乎相同时，它们的 SSIM 值接近 1。RMSE（均方根误差）是计算两个图像的绝对差值。可以看到，在这些度量指标下，本方法结果的深度图和边缘图很好地保持了结构的一致性，也充分地体现了本方法的有效性。

## 第五节 深入理解结构表征网络

本节要解释一下设计结构表征网络的动机和原因，以及方法的局限性：

### 2.5.1 现有方法能否通过调整 $l_{style}$ 和 $l_{content}$ 的权重比，达到与本方法相同的效果？

有些人可能会想是否可以通过增加  $l_{content}$  的权重来获得与本方法类似的结果。乍一听好像很合理，但事实并非如此。

首先需要注意的是， $l_{content}$  是通过计算 VGG 网络中的中间层特征距离得到，但由于 VGG 原本是被设计用于对象识别的。因此其特征将主要集中在主要目标上，背景和其他对象将无法被完全呈现。其次，即使在同一目标上，结构信息也会是多变的，例如肖像中的局部边缘细节（头发部分较丰富而脸部区域较少）或道路的深度值（随着道路的延展而变化）。因此，我们需要引入一些结构表征来保留这些细节。实验进一步验证了我们的分析。如图2.11所示，以<sup>[5]</sup>为例，其并不能并通过简单地调整  $l_{content}$  的权重来达到增强结构的效果。

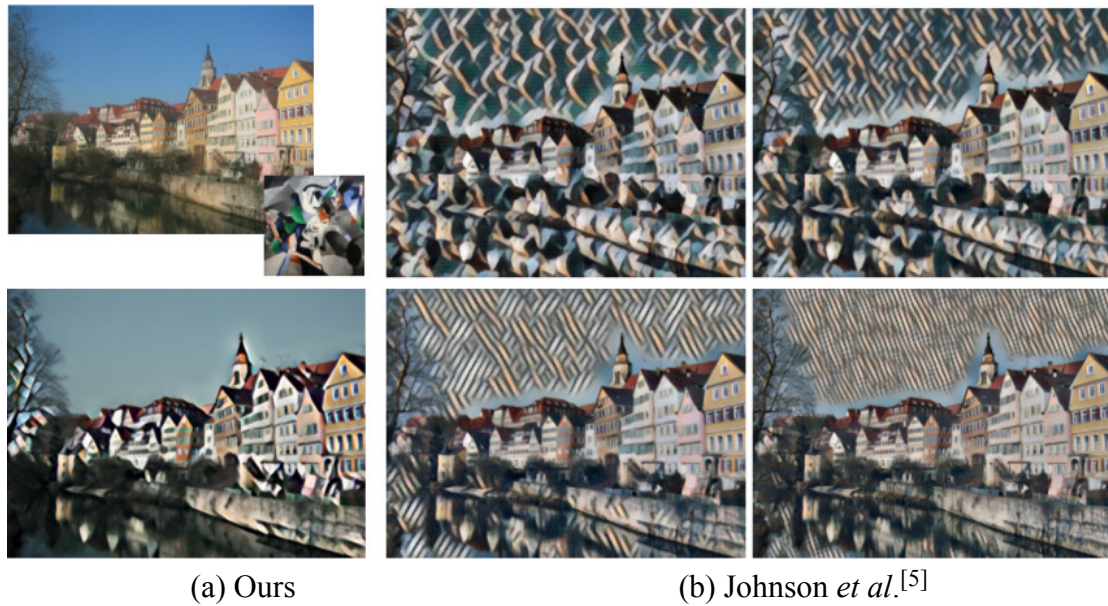


图 2.11 现有方法可以通过增加  $l_{content}$  的权重来达到与本方法类似的效果吗？右边的两列是<sup>[5]</sup>的结果： $l_{content}$  的权重从左到右，从上到下逐渐增加。可以看到，简单地增加  $l_{content}$  的权重只会使结果与内容图像越来越相似，风格纹理仍然均匀散布在整幅图像上。

### 2.5.2 为什么要选择局部和全局结构？

在分析一些现有方法的结果时，如图2.6所示，我们首先会注意到风格纹理的均匀分布是一个普遍问题，这使得前景和背景很难区分，这种现象可以总结为全局结构的缺失。考虑到深度图可以有效地反映图像的空间分布，那么如果可以在风格迁移的同时保留原始图像的深度信息，则可以得到更好的效果。

在保留了全局结构之后，如图2.9所示，上述问题确实得到了改善。可以看到，图像的整体结构得到了增强，但是风格纹理仍然均匀地分布在深度值大致相同的区域（如图2.9的背景）。因此，只考虑整体结构还是不能够完全解决问题。

为了弥补全局结构的局限性，我们增加了局部结构细化网络，因为在人类视觉感知中，过多的风格纹理会间接地引入杂乱边缘进而破坏局部结构。因此，如果迫使风格迁移图像的局部结构与原始内容图像保持一致，那么将在一定程度上消除上述现象，如图2.9所示。

### 2.5.3 方法局限

总的来说，本方法提供了一种可调节的方式，以在图像风格迁移时更好地保留或增强结构。由于每个人都有自己的喜好，每个图像都有自己的特点，本

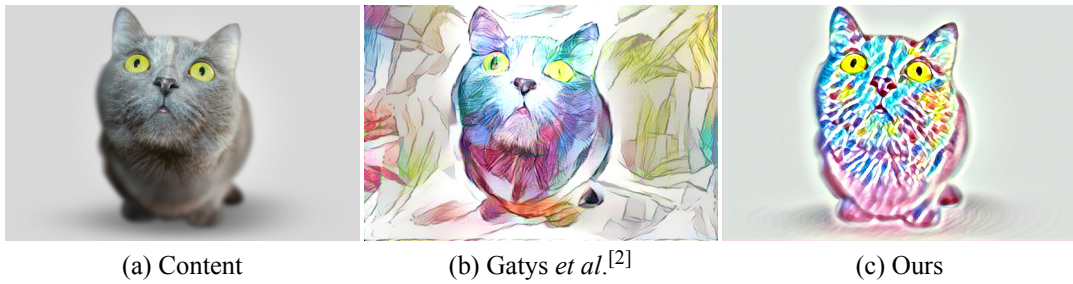


图 2.12 方法的局限性。本方法更适合处理那些对结构变化敏感的图像。对于上图中的这只猫，Gatys *et al.*<sup>[2]</sup> 方法的结构变形赋予其抽象的感觉，看起来很有趣。本方法的结果很好地保留了结构，但同时也丢失了一些抽象感。

方法可能不会令所有人都满意，也不肯适用于所有图像。一般来说，如果偏好抽象的感觉，而且图像内容本身对结构变化不敏感，那么 Gatys 等人<sup>[2]</sup> 的方法可能会更加有效（见图2.12）。

## 第六节 小结

本章提出了一种新的图像风格迁移方法，为的是在对图片进行艺术化渲染的同时保留其整体结构和局部细节。通过设计并使用结构增强表征，在全局结构提取网络和局部结构细化网络的共同作用下，提出的方法成功达到了这一目的。实验结果表明，该方法成功保留了内容图像的语义内容和布局，而且这一特点在处理对结构失真敏感的图像时体现得尤为明显。同时本文对不同方法的结果进行了分析比较，用户调研结果表明，本方法的效果总体更受欢迎。

如前所述，在对图像进行风格迁移后更改其深度的绝对值是可以接受的，但应保留相对深度，保留必要的局部细节以及对前景和背景的不同渲染，尤其是对那些深度覆盖范围较大、局部细节丰富的图像。同时，风格迁移也应该保留原始内容图像的空间布局，本章通过比较由风格迁移引起的深度和边缘图变化对这一特性进行了评估，参与调研的用户也在本方法可以更好地保持结构一致性这一点上达成了普遍共识。最后，实验结果同时也反映了图像表征在风格迁移中起着非常重要的作用。



图 2.13 更多结果。与 Johnson *et al.*<sup>[5]</sup> 的方法相比，本方法的结果更好地捕捉了图像的主要结构，而且有着更明显的轮廓边界。

### 第三章 2D/3D 表面的风格迁移

上一章展示了本文的第一个案例：通过设计和利用图像结构表征，在进行风格迁移的同时保留图像结构。沿着这一思路，本章将继续阐述如何同样通过设计图像表征这一策略来完成另一项任务：在 2D/3D 表面指定区域进行风格迁移。

本章考虑的问题是如何将给定的样本图案合成到用户指定的区域或路径。系统的输入是：1) 用户绘制的样例图案；以及 2) 用户绘制的路径或指定的区域。具体效果如图3.1所示。作为一种基本的视觉表现元素，结构化装饰图案被广泛用于多种艺术形式中，这些元素的合理使用会丰富艺术作品的内涵，同时增加其视觉吸引力。

本工作同样受到基于 CNN 的图像风格迁移工作的启发。如前面所述，预训练的 CNN 框架在特征提取方面有着突出的表现。然而，预训练的 CNN 很难直接从单个较小的图案中提取样式特征，其更适合处理大面积的连续纹理，而且合成效果将遍布整幅图像（见图3.10）。本方法使用的孤立单一图案和用户指定区域将更有挑战性，而且如果只是简单地从最终结果中剪裁出原有形状，将截断那些跨越边界的结构图案，使得结果看起来非常不自然（见图3.3）。本方法注

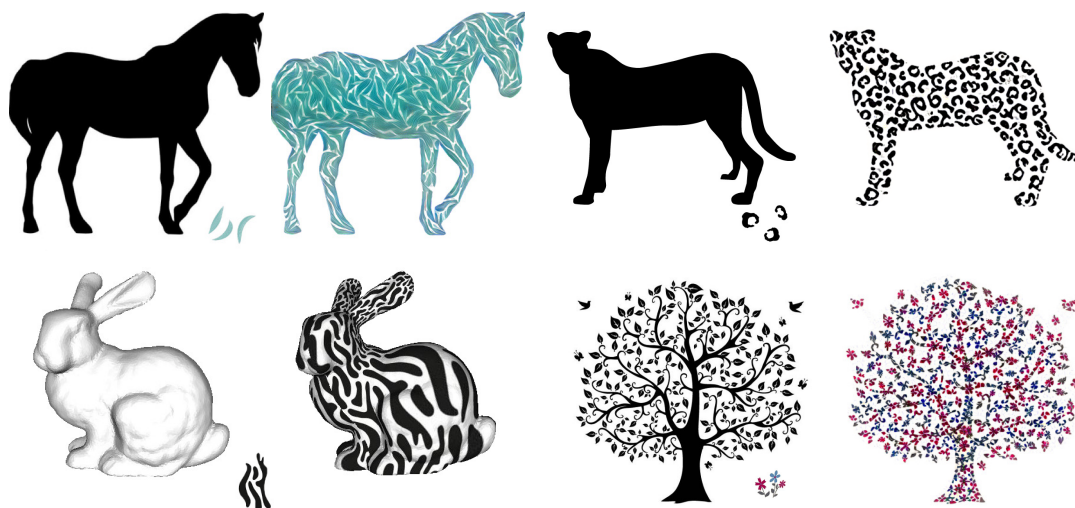


图 3.1 本章提出了一种使用卷积神经网络在 2D/3D 表面特定区域进行风格迁移的方法。该方法将指定路径或区域转换为与样本图案类似的纹理。

意到现有的图像表征没有办法直接地反映图像的形状轮廓，所以尝试通过设计形状轮廓表征的方式对现有的进行补充，进而解决上述问题。

系统首先使用 VGG 网络对输入样本进行预处理，利用中间层特征之间的相关性得到其深度表征。然后，计算得到用户指定图像的轮廓形状表征，并通过该表征消除指定区域外的图案以在优化过程中保持轮廓形状。此外，为了获得高质量高分辨率的图案，本文提出了一种多尺度纹理化策略，在优化的同时匹配金字塔的每一层的图案描述和轮廓形状表征。通过利用共形映射，本方法还可以用于装饰 3D 曲面。

## 第一节 相关工作

### 3.1.1 通用纹理合成

纹理合成指的是从给定的小样本生成更多可平铺的纹理，近年来受到很多关注，并且已广泛用于许多其他领域，而且有一些经典的代表性工作<sup>[15, 42-45]</sup>。GraphCut<sup>[46]</sup> 采用图形切割技术，将贴片或区域从样本图像扩展和缝合到新图像。还有不少其他相关研究，如 GPU 辅助实时实现<sup>[47]</sup> 和内容填充<sup>[48]</sup>。在<sup>[49]</sup> 中，纹理合成工作被扩展为图像风格迁移。除了 2D 纹理和图像合成外，面向 3D 曲面的映射和表面纹理的合成也是一个有趣的研究课题。例如，TextureMontage<sup>[50]</sup> 使用多个图像在任意 3D 模型上生成无缝纹理。为了打印单个 3D 物体，<sup>[51]</sup> 引入拓扑约束进行纹理合成，通过沿着来自输入矢量形状的曲线合成所需图案。此外，基于补丁的纹理合成也是图像处理和编辑中非常有潜力的方向。通过引入有效的近似最近邻策略，PatchMatch<sup>[52]</sup> 已被广泛用于图像修复以及许多其他相关领域。为了更好地理解和操纵结构化图像/视频内容，研究人员进一步引入了更复杂的图像片块查询解决方案，如 PatchNet<sup>[53]</sup> 和 PatchTable<sup>[54]</sup>。如果想要获得更多有关基于图像片块的综合概述，可以参考综述<sup>[55]</sup>。

### 3.1.2 结构化纹理编辑

侧重结构化图案或纹理的图像合成已被广泛应用于许多研究中。<sup>[56]</sup> 通过设计平滑的矢量场来增强虚幻强度和内容的动态形状，并通过一些静态纹理生成虚幻运动，也被称为自动动画图像 (self-animating image)。在合成图像时，将一些形状或对象隐藏到目标图像中使其不被察觉也是一个有趣的研究课题。伪装图像<sup>[57]</sup> 从输入形状中移除原始细微纹理，并使用周围的背景对其进行变换，从

而延迟观察者对隐藏图形的感知。<sup>[58]</sup> 先通过优化能量函数自动检测和放置要隐藏对象的边缘，然后使用泊松混合方法无缝嵌入纹理。在<sup>[59]</sup>的框架中，用户可以轻松地绘制任意示例图像的样式，并且构造一些画笔用于实时绘画或对矢量图像进行风格迁移。Brushables<sup>[60]</sup>是一种交互式纹理绘画工具，可以合成所需的内部纹理和区域边界。与<sup>[60]</sup>所用的传统贪婪搜索策略相比，本文引入了神经网络来自动提取输入纹理特征并组合区域目标形状。<sup>[61]</sup>采用形状匹配，轮廓变形和图像裁剪，组成一种特殊的二值图像，即模糊图形-地面图像（ambiguous figure-ground images）。类似地，<sup>[62]</sup>开发了自动 Escher 变形，将一个瓦片结构平滑地转换成具有双重感知的另一个瓦片结构。<sup>[63]</sup>引入了一小组富有表现力的几何和分组算子来描述复杂模式，然后通过重复和递归模式对结构化的笔墨 2D 风格进行了表征。<sup>[64]</sup>沿用户指定的路径合成装饰图案。一些其他工作如<sup>[9]</sup>模拟装饰马赛克瓦片，<sup>[65]</sup>使用任意形状的图像瓦片组成最终图片，<sup>[66]</sup>通过最小化间隙和重叠区域包装不规则和合理尺寸的形状，对指定区域进行填充。<sup>[67]</sup>通过创建视觉图像使一个单词呈现适合的 2D 形状。<sup>[68]</sup>引入了一种基于环的布局模型，能够捕获例如板和盆等物体的装饰成分。<sup>[69]</sup>提出了一种绘制装饰设计的技术。此外，几何纹理合成，例如重复结构生成<sup>[70]</sup>支持合成具有连续和离散结构的 3D 形状。

### 3.1.3 基于 CNN 的纹理表征

纹理合成可以为图片提供各种艺术效果，因此属于一种非真实感渲染（NPR）技术。艺术风格转换的一项开创性工作图像类比<sup>[17]</sup>，其对输入纹理根据另一个给定的示例图像进行风格转换。随着基于 CNN 的特征分析和图像表征技术的发展<sup>[18]</sup>，已经有了许多真实/非真实纹理重建的方法。例如，Gatys 等人<sup>[2]</sup>利用经过训练的分类器提取的高级特征来表示图像的内容，并基于特征的 gram 矩阵进一步对图像进行风格化。其他工作，如<sup>[5, 11, 20]</sup>的目的是为了加快专业风格学习神经网络的优化处理，而<sup>[7, 37]</sup>的重点是通过将多种风格集成到一个模型中来实现模型的灵活性。在<sup>[5]</sup>中，网络是预先训练的，而不是直接优化损失函数，因此提高了新输入图像样式化的效率。并且<sup>[71]</sup>试图理解基于 CNN 的纹理描述符捕捉到的不变性，<sup>[72, 73]</sup>提出在神经风格转换的现有损失函数上添加额外的损失，以达到不同的目的。另一方面，基于 CNN 的图像补全<sup>[74]</sup>和纹理合成<sup>[75]</sup>也可以产生逼真的面部或其他结构化图像。与这些基于学习的图像表征算法不同，本文工作专注于结构化图案或纹理，旨在将给定的纹理样本迁移到目标二维和三维形状上。

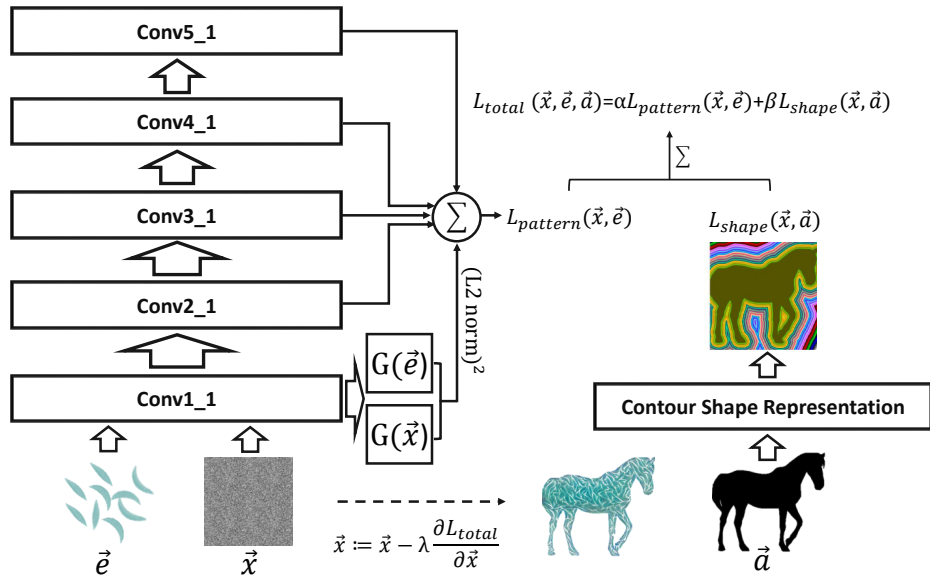


图 3.2 系统流程。左边显示了样本图案  $\vec{e}$  和优化图像  $\vec{x}$  的当前状态，其中特征相关 Gram 矩阵是从 VGG 网络的选定层中提取的。这些矩阵的平方差的总和对应于  $L_{pattern}$ 。右边是输入图像  $\vec{a}$ ，轮廓形状表征通过距离变换得到。总损失是  $L_{pattern}$  和形状重建损失  $L_{shape}$  的加权和。 $L_{total}$  通过在  $\vec{x}$  执行梯度下降进行优化。

## 第二节 算法概述

本算法在用户指定的图像中根据输入样例图案生成类似的图案，其合成结果通过基于 CNN 的迭代优化生成。算法的优化目标是 minimized 样本和合成图案之间的纹理数理统计差异。此外，还需要确保生成结果的轮廓形状与用户指定的图像保持一致。

在合成之前，系统先提取样例图案的特征描述。在系统中，输入样本是用户绘制的图案或指定的小纹理片块（如图3.1）。其特征是在 VGG 网络<sup>[76]</sup>的帮助下获得的，如之前所述，VGG 是一种高度精确的 CNN 架构，最初是用于物体识别和定位。我们使用了 19 层 VGG 网络的卷积层提供的特征映射（详见3.2.1）。通常来说，网络中的每个层定义了一个非线性滤波器组。因此，CNN 通过获取每个层对该图像的滤波器响应对输入图像进行编码，我们也随之获得合成图案的特征，并使它们在优化过程中与相应的样例图案特征保持一致。

接下来需要计算得到用户指定图像和合成结果的轮廓形状表征（细节见3.2.2）。需要注意的是，如果直接利用现有的基于 CNN 的图像风格迁移或编辑框架，可能会导致一些视觉瑕疵，而且目标形状将被破坏（如图3.10）。因此，本方法采用计算输入和输出图像之间的距离场，并最小化合成过程中的形状重

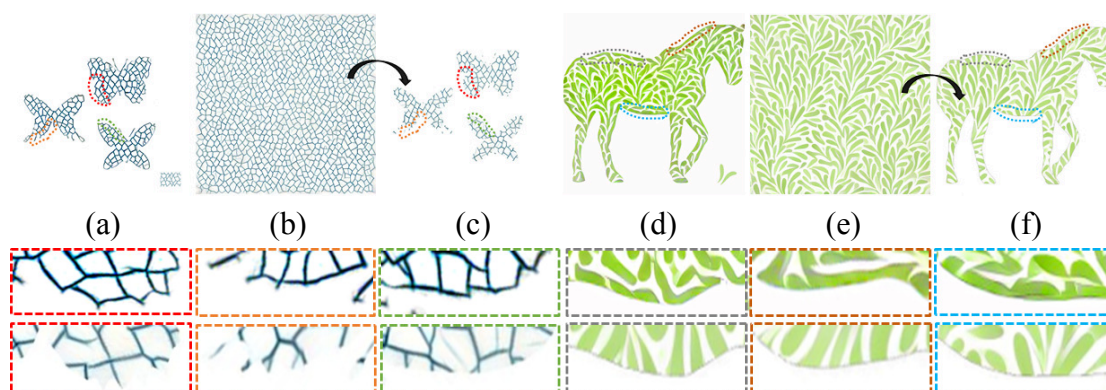


图 3.3 为什么不简单地裁切图案。在第一行 (a) 和 (d) 是联合优化下的结果。(b) 和 (e) 显示在整个图像上合成的纹理。(c) 和 (f) 显示通过直接掩蔽 (b) 和 (e) 中的结果而获得的结果。由于纹理是在优化过程中生成的，因此逐渐形成了许多结构。粗略地剪裁图像会破坏跨越边界的结构，使得过渡看起来不自然且没有吸引力。第二行和第三行分别是我们的结果中的相同边界区域的放大结果和粗略剪切结果。

建损失。通常来说，距离场是一种网格结构，其中的每个值表示从该单元到具有一些其他属性单元的最短距离。我们将每个图像视为二进制图像，距离场为每个像素分配一个值，代表其到指定区域最近像素的距离。这样以来，指定区域内的像素值将变为零。而对于轮廓外的像素值，越远则越大。使用轮廓形状表征，我们以一种软约束的方式对形状不一致性进行了惩罚。通过这种方式，迭代优化不仅可以保留所需的形状，还可以避免在指定区域外引入视觉瑕疵。

最后，为了有效地生成高分辨率图像，本方法设计了基于 CNN 框架的目标损失函数 (3.2.3)，并使用梯度下降法通过多尺度策略 (3.2.4) 最小化损失值。

### 3.2.1 深度图案描述

通常来说，外观类似的图案会共享一些局部统计规律。这个概念最初是由 Julesz<sup>[77]</sup> 提出的，他指出不同纹理图案之间的视觉差异可以通过图像滤波器的平均响应来量化。遵循这一原则，Zhu 等人<sup>[78]</sup> 进一步形式化了 Julesz 的想法，提出可以用一组非线性滤波器的统计规律来描述纹理特征。

巧合的是，CNN 由一系列的层组成，每个层可以看作是一组不同数量的滤波器。在基于 CNN 的图像处理系统中，网络的每个层通过图像的滤波响应对输入图像进行编码。因此，遵循 Gatys 等人提出的范式<sup>[2]</sup>，本文也选择了 VGG 网络中的某些中间层作为非线性滤波器组来获取图案的统计规律。

假设 VGG 网络的  $l$  层具有  $N_l$  个不同的滤波器，每个特征响应的大小是

$W_l \times H_l$ ，其中  $W_l$  和  $H_l$  分别是层  $l$  中的特征映射的宽度和高度。这种层中的响应可以用矩阵表示：

$$F^l \in \mathcal{R}^{N_l \times W_l \times H_l} \quad (3.1)$$

并且每个值  $F_{ijk}^l$  是层  $l$  中位置  $(j, k)$  处的第  $i$  个滤波器的激活。我们使用由 Gram 矩阵  $G^l \in \mathcal{R}^{N_l \times N_l}$  表示的若干层的滤波器响应之间的相关性作为纹理图案表示。在  $G^l$  中， $N_l$  是层  $l$  中的滤波器数， $G_{ij}^l$  是层  $l$  中第  $i$  个和第  $j$  个矢量化特征映射的内积：

$$G_{ij}^l = \sum_{w=1}^{W_l} \sum_{h=1}^{H_l} F_{iwh}^l F_{jwh}^l \quad (3.2)$$

简而言之，深度纹理图案是由来自 VGG 网络的滤波器组  $F = (F_1, \dots, F_n)$  进行描述的。通过匹配这些过滤器上的深度图案描述可以生成类似图案。

### 3.2.2 轮廓形状表征

当直接使用现有的神经网络处理图像时，一个明显的现象是该过程将作用于整幅图像（见图3.10）。相比之下，我们的重点是在用户指定的区域中生成结构化纹理，因此需要避免出现在所需区域之外的非预期元素。

但是，我们不能简单地使用轮廓图从最终的纹理结果中剪切出所需形状，因为这会截断那些跨越边界的结构，使得结果看起来不自然和人为（如图3.3，在3.4.2中会进一步分析这一点）。本文将整个合成过程归纳为一个联合优化问题，使用输入图像的距离变换，并用它来帮助在纹理合成优化过程中保持轮廓形状，从而直接消除伪影，而不是依赖于后处理。

一般而言，距离变换是从任何点到某种类型的最近点的距离的映射。考虑图像  $x$  的一组点  $P$ ，距离度量是：

$$DT(P)[x] = \min_{y \in P} \text{dist}(x, y) \quad (3.3)$$

需要注意的是，在该式中可以采用各种测量距离，例如欧几里德，曼哈顿和棋盘距离。本方法使用欧几里德距离度量。由于输入图像可以二值化，因此在  $DT$  中，轮廓形状内的像素值为零，相反，轮廓附近的值较小而远处的值较大。具体来说，可以用如下等式表示这三种不同的值：

$$d_{ij} = \begin{cases} 0 & \text{(inside the contour)} \\ \text{small} & \text{(near the contour)} \\ \text{large} & \text{(far from the contour)} \end{cases} \quad (3.4)$$

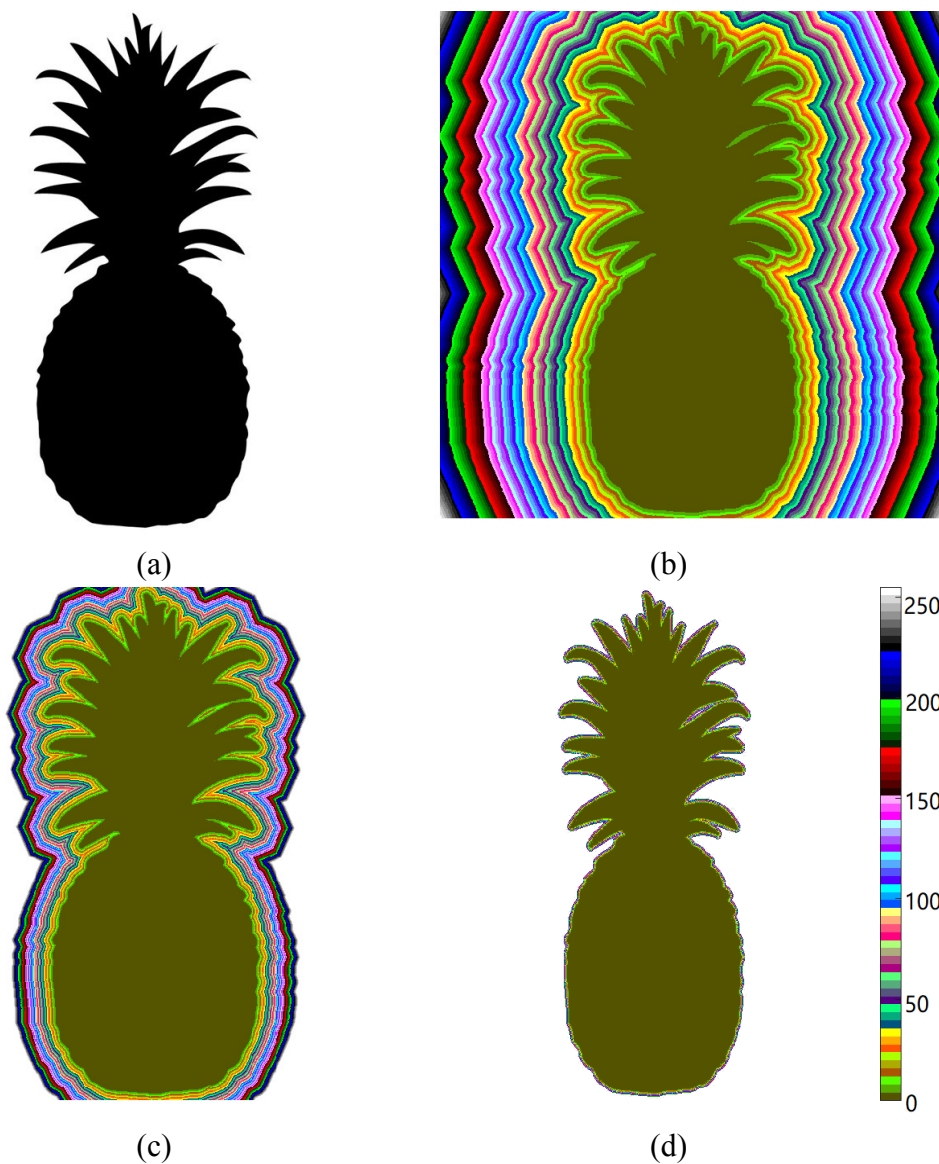


图 3.4 轮廓形状表征。(a) 是目标轮廓，(b) 是可视化后的轮廓形状表征。可以看到，轮廓形状内的值为零，而轮廓外的像素离得越远值越大。将 (b) 平方后得到 (c)，(d) 为 (b) 的三次方。与轮廓区域附近的那些像素相比，远离轮廓的像素值变化幅度更大。

图3.4 (b) 显示了可视化的一个例子。

在获得内容图像的距离图  $DT$  之后，我们可以使用它来不同程度地强调结果中的指定区域。具体来说，我们采用  $DT$  的像素级幂运算。在取得  $DT$  的幂之后，如图3.4 (c) 和 (d) 所示，远离轮廓形状的像素值变大，而附近像素的值变化不大，这样就为前景和背景提供了高对比度。

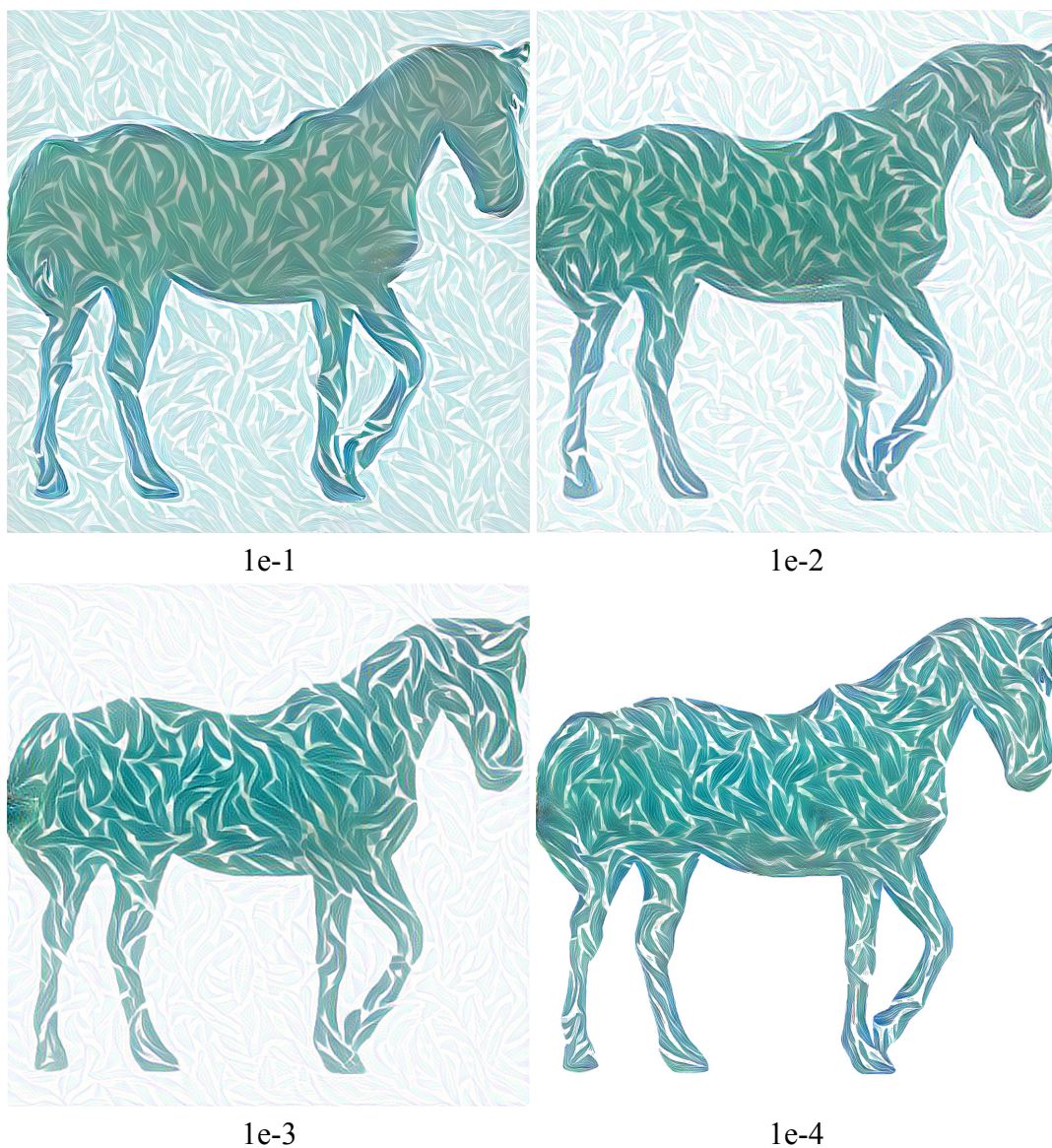


图 3.5 图案和形状重建损失不同组合的结果。每幅图像下的数字代表样本图案深度表征和目标图像轮廓形状表征之间的权重比  $\alpha/\beta$  (详见 Sec.3.2.3)。

### 3.2.3 目标损失函数

为了将示例图案  $\vec{e}$  转移到指定区域  $\vec{a}$ ，我们选择对白噪声图像执行梯度下降算法，以找到与  $\vec{e}$  的深度图案描述和  $\vec{a}$  的轮廓形状表征匹配的另一图像。

设  $\vec{x}$  为生成的图像，本文定义的目标损失函数由两部分组成，如下所示：

$$\mathcal{L}_{total}(\vec{x}, \vec{e}, \vec{a}) = \alpha \cdot \mathcal{L}_{pattern}(\vec{x}, \vec{e}) + \beta \cdot \mathcal{L}_{shape}(\vec{x}, \vec{a}) \quad (3.5)$$

其中  $\alpha$  和  $\beta$  分别是图案和形状重建的加权因子。

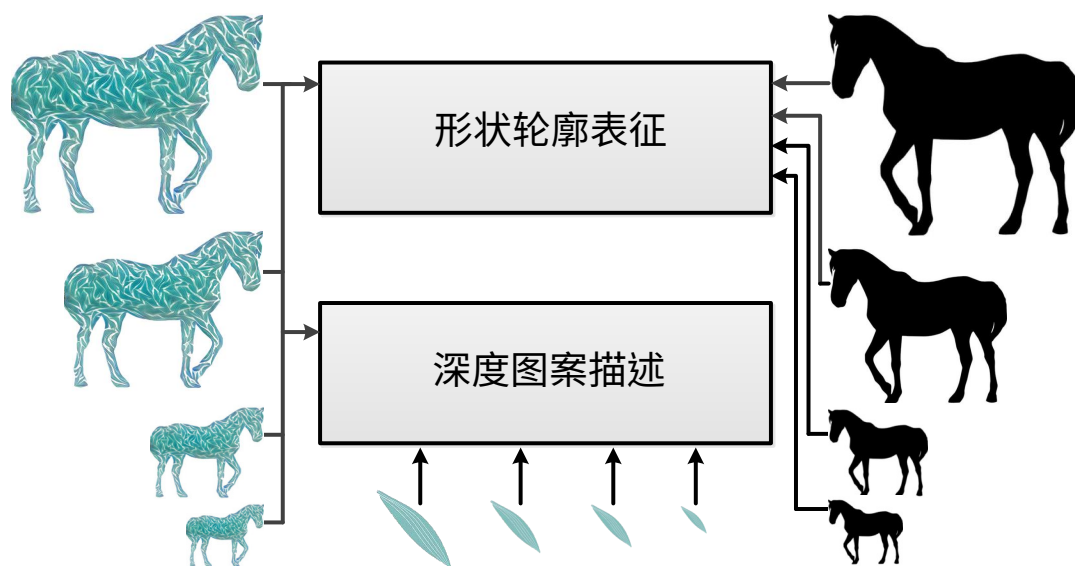


图 3.6 为了克服现有基于 CNN 的框架会引入不令人满意的结果这种局限，本文使用多尺度策略来合成模式，通过这种方法可以有效地合成各种高质量的高分辨率图像。

方程的第一部分损失函数  $\mathcal{L}_{pattern}$  是 VGG 网络所选的滤波器组  $F = (F_1, \dots, F_n)$  中每个响应的 Gram 矩阵的欧几里德距离的总和（见第 3.2.1 节）。其中  $n$  是所选层的数量， $G_l$  是 3.2 的形式。具体实现中，本文选择匹配层 ‘conv1\_1’，‘conv2\_1’，‘conv3\_1’，‘conv4\_1’ 和 ‘conv5\_1’ 上的图案表征。

用于保护形状轮廓的损失函数是：

$$\mathcal{L}_{shape}(\vec{x}, \vec{a}) = \frac{1}{2} \|DT(\vec{a})^n \circ (\vec{x} - \vec{a})\|_2^2, \quad (3.6)$$

其中  $n \geq 2$  且  $DT(\vec{a})$  是  $\vec{a}$  的轮廓形状表征。 $\circ$  代表 Hadamard 积，其作用于两个相同尺寸的矩阵并产生同样尺寸的第三矩阵。如第 3.2.2 节所述，距离轮廓区域较远的像素具有相对较大的  $DT$  值，轮廓区域内的像素  $DT$  值为 0（如图 3.4 所示）。换句话说， $\mathcal{L}_{shape}$  确保轮廓形状周围像素的惩罚总是小于其他像素的像素。所以通过减小  $\mathcal{L}_{shape}$  的值，我们既能保留图像原始形状，同时又可以避免在指定区域外引入元素。

对于文中展示的所有图像，匹配的是层 ‘conv1\_1’，‘conv2\_1’，‘conv3\_1’，‘conv4\_1’ 和 ‘conv5\_1’ 上的图案表征。比率  $\alpha/\beta$  为  $1 \times 10^{-4}$ 。图 3.5 显示了不同的图案和形状重建损失权重对结果的影响。

### 3.2.4 多尺度合成策略

当使用神经网络处理图像时，如之前<sup>[79]</sup>指出的那样，网络中神经元的有效感受域相对较小。这就限制了可以合成的图案特征的最大尺寸。具体来说，如果尝试使用神经网络直接合成高分辨率图像，结果将不尽人意（参见图 3.11 中的示例）。

Portilla 等人<sup>[14]</sup>利用了一组称为“可操纵金字塔”的多尺度线性滤波器进行纹理合成，受该工作启发，本文提出了一种多尺度合成策略，如图3.6所示。通过使用这种策略可以合成外观优美的高分辨率图案。下文将给出该策略的更多细节。

我们首先将图像送到“金字塔”的层中，其中每个层通过对前一层进行模糊和插值来得到该层输出。整个优化过程会同时匹配图像纹理在多个尺度上的特征。具体来说，设 CNN 的  $s^{th}$  和  $l^{th}$  层的特征矩阵为  $F^{l,s}$ 。如第3.2.1节中所定义，相应的 Gram 矩阵  $G^{l,s}$  的元素重新表述为：

$$G_{ij}^{l,s} = \sum_{w=1}^{W_l} \sum_{h=1}^{H_l} F_{iwh}^{l,s} F_{jwh}^{l,s} \quad (3.7)$$

对应地，如第3.2.2节中所定义，尺度为  $s$  时的  $\mathcal{L}_{pattern}$  为：

$$\mathcal{L}_{pattern}^s(\vec{x}, \vec{e}) = \sum_{l=1}^n \|G_l(\vec{x}^s) - G_l(\vec{e}^s)\|_2^2 \quad (3.8)$$

而  $\mathcal{L}_{shape}$  变为：

$$\mathcal{L}_{shape}^s(\vec{x}, \vec{a}) = \frac{1}{2} \|DT(\vec{a}^s)^n \circ (\vec{x}^s - \vec{a}^s)\|_2^2 \quad (3.9)$$

其中  $\vec{x}^s, \vec{e}^s$  和  $\vec{a}^s$  分别是生成的图像，输入范例和尺度  $s$  的指定区域。

最后，我们可以利用多尺度公式修正原有的目标损失函数：

$$\mathcal{L}_{total} = \sum_{s=1}^S (\alpha \cdot \mathcal{L}_{pattern}^s + \beta \cdot \mathcal{L}_{shape}^s) \quad (3.10)$$

其中  $S$  代表金字塔的总层数。

使用不同数量的尺度会产生不同的结果，如图3.11所示，本文将在第第四节对这一点做更进一步的分析。

### 第三节 实验结果

图3.7, 3.8和3.9展示了对各种样例图案的合成结果。总的来说, 本方法会在用户指定的区域中连贯地合成纹理。在效果上, 合成的图案与示例图案相似但并不完全相同。接下来将展示更多在 2D 区域, 曲线和 3D 表面上的结果。

#### 3.3.1 区域合成

图3.7展示了将本方法应用于平面区域时的结果。总的来说, 指定的区域内充满了类似示例图案的元素。仔细观察会发现, 这些范例也有一些变化和衍生物。这是由卷积网络的使用引起的。因为与那些为样本的拓扑结构构建精确模型的传统方法不同, 本文基于 CNN 的框架倾向于提取语义级别的高级抽象特征, 尤其是在处理单个分离图案时。因此, 合成结果会将样本图案合理地平铺, 并且这些样本会自动变形而不是简单地复制。

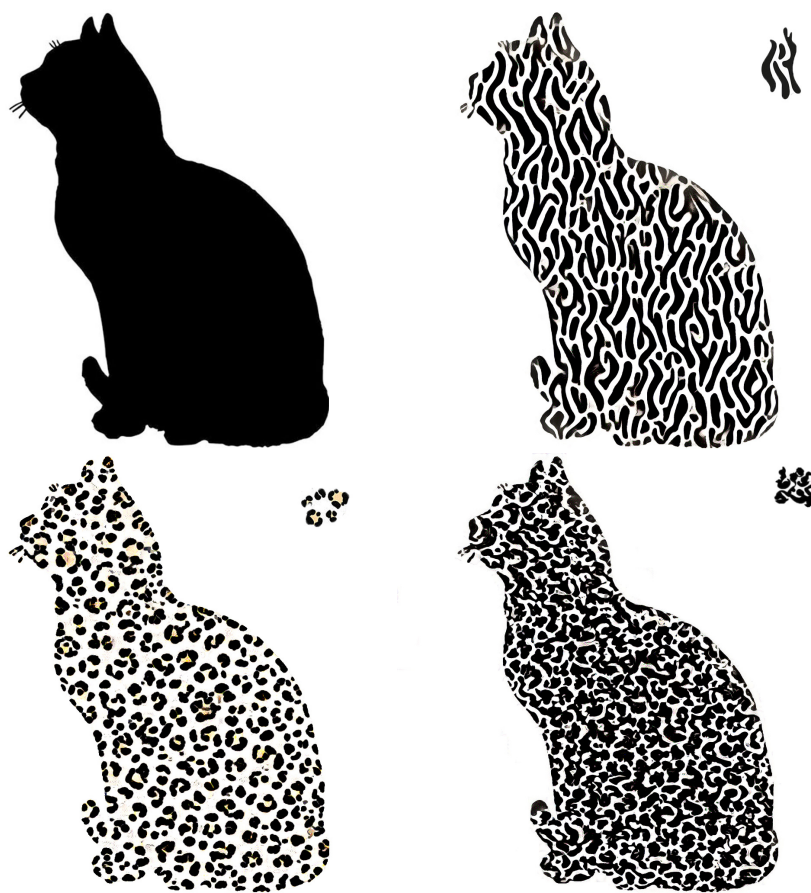


图 3.7 在平面区域中合成。每幅图对应的示例图案在右上角。



图 3.8 沿着指定路径合成。每幅图对应的示例图案在右下角。

### 3.3.2 沿曲线合成

如图3.8所示，本方法也可以应用于曲线合成。实际上，曲线可以被视为区域的一种特例，只是有着带状和狭窄区域。可以先通过基本的图像处理技术对曲线进行膨胀和腐蚀，之后再使用所提出的框架，从而实现沿着曲线合成纹理这一目的。

### 3.3.3 装饰表面

借助共形映射<sup>[80, 81]</sup>，本方法也可以应用于装饰 3D 表面。共形映射具有保角特性，等效来说，其可以将表面上的圆映射到平面上。假设  $S$  是以 Euclidean 度量  $\mathbf{g}$  嵌入欧几里德空间  $R^3$  中的可定向表面，我们可以通过共形映射  $\phi: (S, \mathbf{g}) \rightarrow (D, \mathbf{h})$ 。将  $S$  映射到具有黎曼度量  $\mathbf{h}$  的平面盘  $D$ ，通过在合成后将  $D$  映射回表面，如图3.9所示，我们在曲面上精确地合成图案。用户还可以交

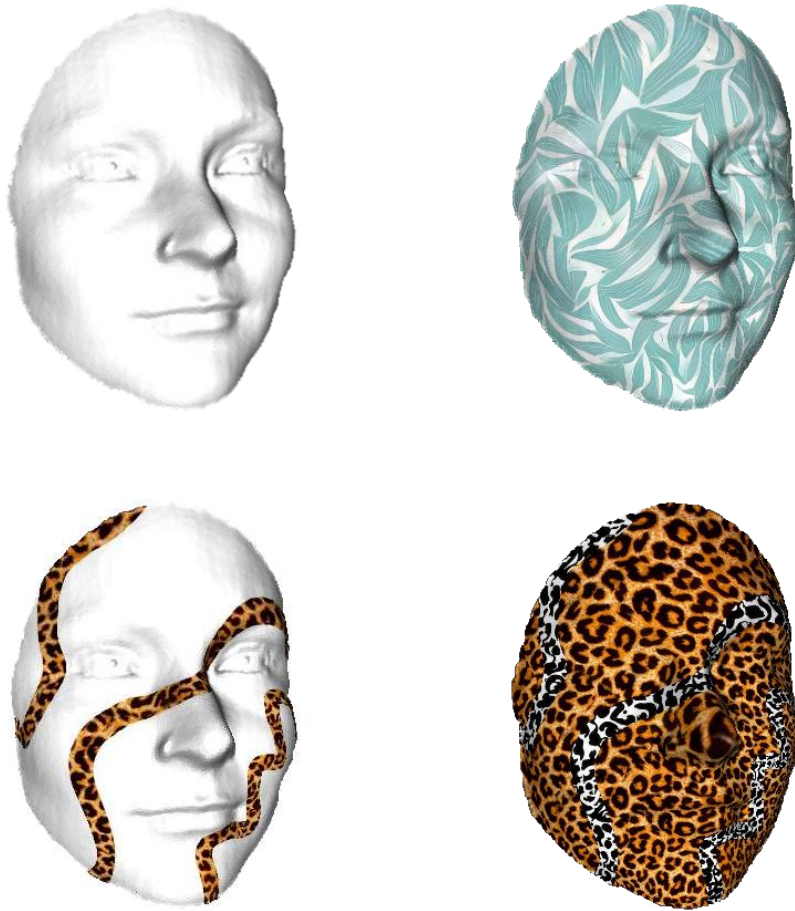


图 3.9 装饰 3D 曲面。通过共形映射可以实现 3D 曲面上的图案合成。而且用户可以交互式地为不同区域选择不同图案。

交互式地为不同区域选择不同的图案。

## 第四节 分析和比较

如第二节所述，本算法由轮廓形状表征和多尺度纹理合成两部分组成。现在具体分析它们分别在结果中所起的作用。

### 3.4.1 消融分析

首先值得注意的是我们引入了轮廓形状表征。与现有的基于 CNN 的风格迁移方法（过程将应用于整幅图像）相比，轮廓形状表征有助于保留原始轮廓结构并移除指定区域外部的元素。如图3.10所示，像 Gatys 等人的结果，由于没有

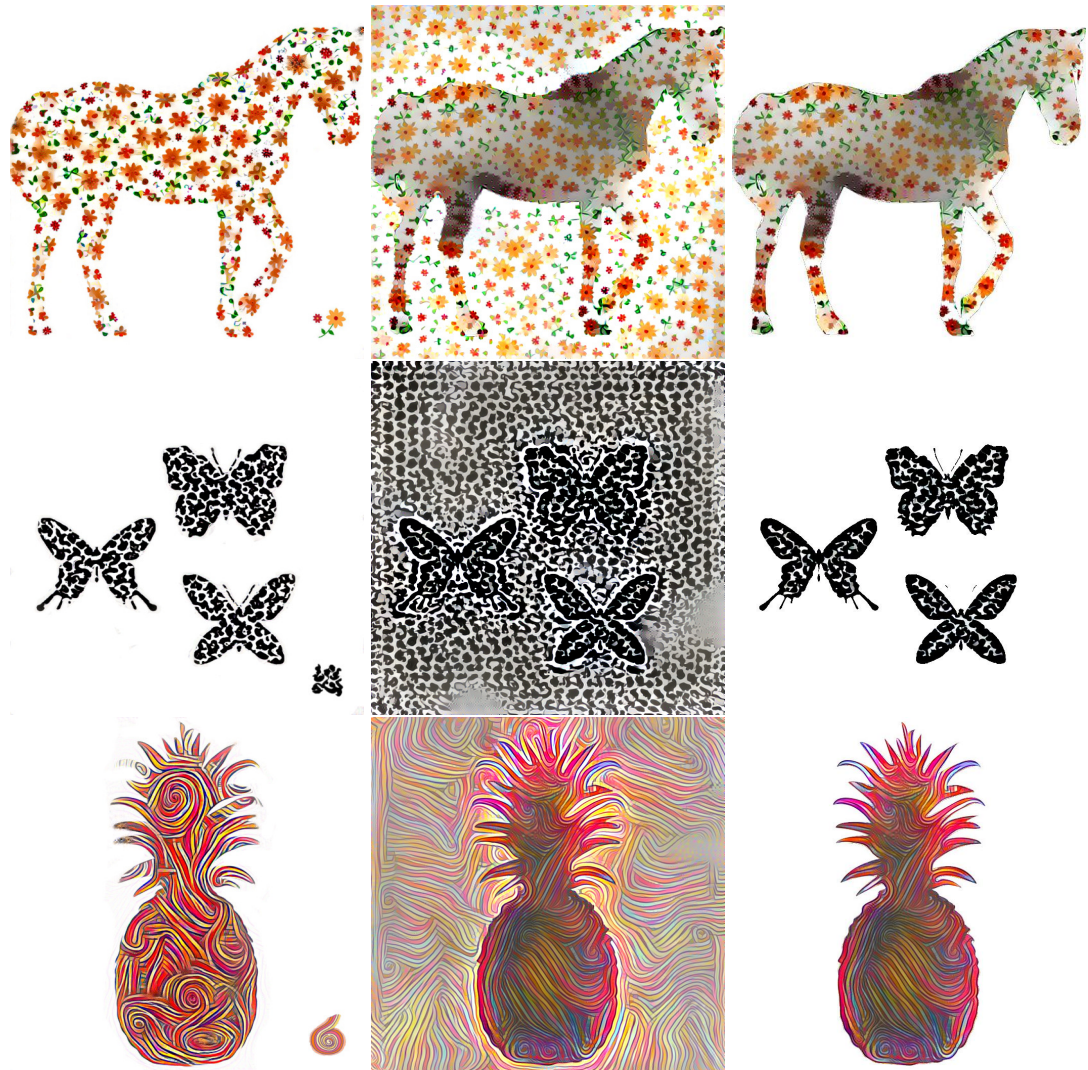


图 3.10 左：本文结果（样例图案位于右下角）。中：Gatys *et al.*<sup>[2]</sup> 方法的结果。右：直接裁剪 Gatys *et al.*<sup>[2]</sup> 的结果。可以看到，如果没有使用轮廓形状表征，图案会散布在整幅图像上，而且很容易产生瑕疵。

使用轮廓形状表征，纹理图案将散布在整幅图上，并且很容易产生瑕疵。

另一种对结果有帮助的策略是多尺度合成。图3.11展示了不同数量尺度之间的比较结果。当合成相同尺寸的图像时，如果不使用多尺度策略，纹理细节将过度模糊并且很易于引入瑕疵。有趣的是，瑕疵的出现主要归因于收敛的不稳定，并且图像越大瑕疵将越容易出现。相反，多尺度策略迫使优化过程匹配金字塔每一层的图案描述，在消除大块的瑕疵同时阻止小块瑕疵的蔓延。通常来说，层数越多将有助于获得更好的结果，但也会导致更大的计算量。考虑到计算性能和视觉质量，本文在大多数情况下使用 4 层。

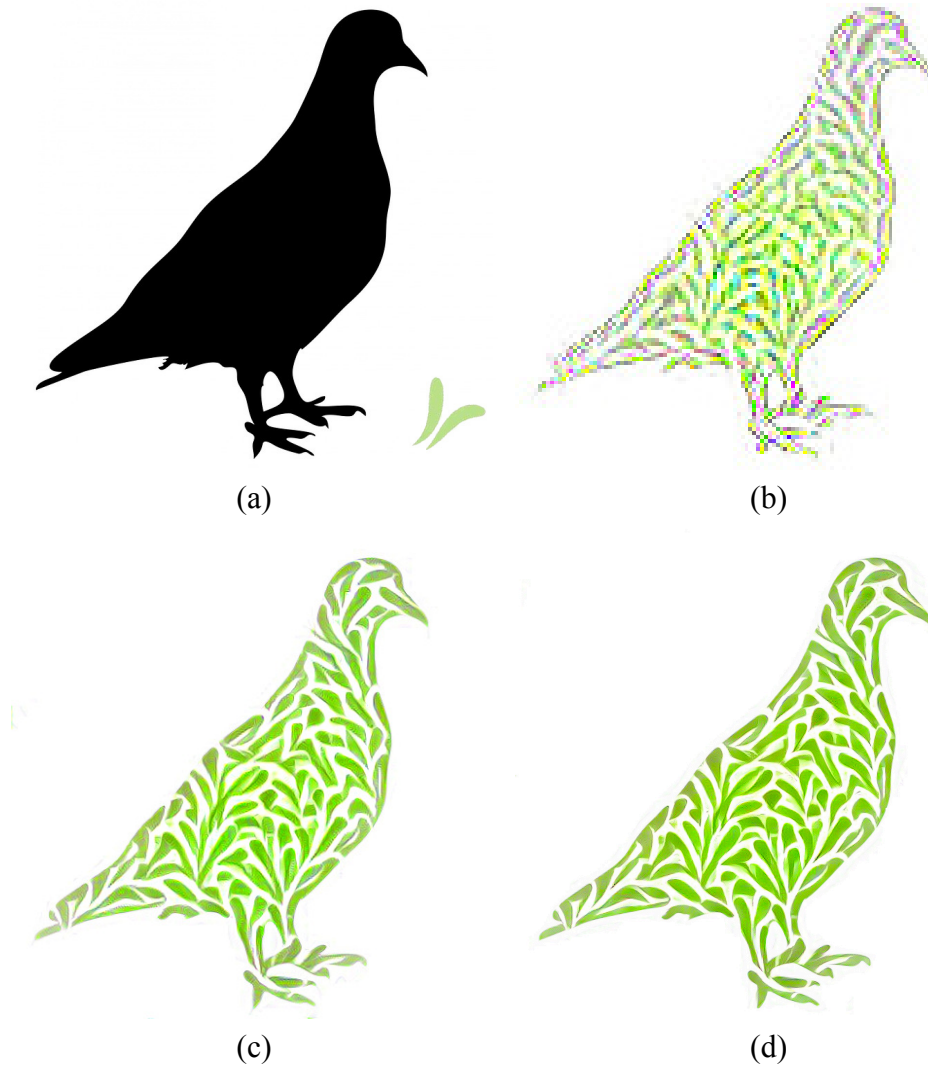


图 3.11 在框架中使用不同数量的尺度会产生不同清晰度的结果。输入图像的尺寸是  $1024 \times 1024$ ，在 (b)，(c) 和 (d) 中分别使用 2，4，6 层尺度。随着尺度数量的增加，局部细节逐渐得到保留，视觉伪像也随之减少。

此外，如图3.12所示，我们为同种图案提供了多种视觉选择，例如不同的比例和方向。这种视觉多样性使得方法可以产生更丰富的效果。

### 3.4.2 边缘过渡区

与简单地使用轮廓图从最终结果中切出所需形状相比，本文结果的明显优势是边界区域附近的过渡将更加自然。具体而言，在本文结果中，靠近边界区域的纹理会倾向于沿着边界分布。相反，如果直接切割，边界纹理的完整性将会遭到破坏（参见图3.3）。

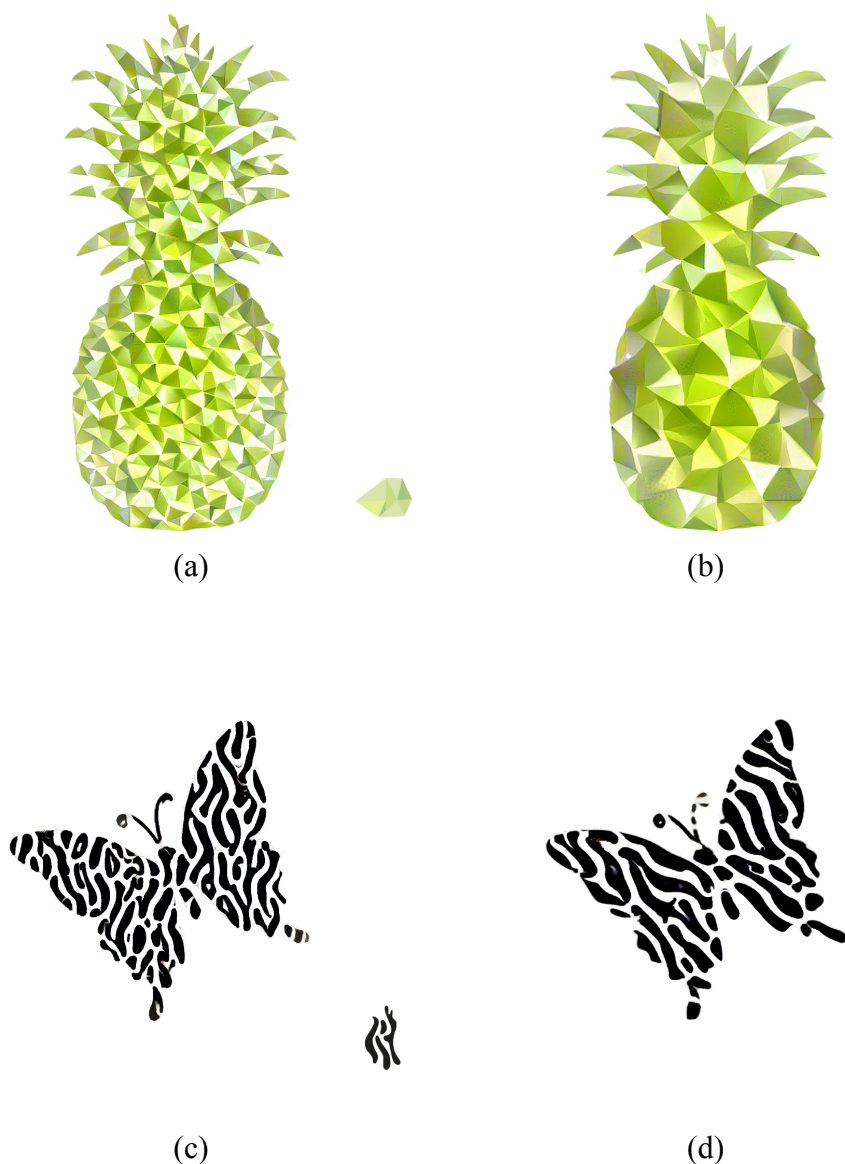


图 3.12 本方法可以为相同的图案提供视觉上的多样选择，比如不同的尺度和方向。与 (a) 和 (c) 相比，(b) 和 (d) 有着更大的尺度。而且 (d) 的图案旋转了 45 度。

本方法结果中边缘区域附近的自然过渡主要归功于轮廓形状表征的使用。参见图3.13，考虑边界附近的两种不同的纹理元素分布状态，一种是沿着边界，另一种是跨越边界。为了方便分析，可以将每个纹理元素分成两部分：轮廓内的像素  $P_{in}$  和轮廓外的像素  $P_{out}$ 。

这种情况下，在对结果进行轮廓形状表征  $DT$  操作后，形状重建损失，如式3.6所示，变为：

$$\mathcal{L}_{shape} = \frac{1}{2} \|DT(\vec{a})^n \circ P_{out}\|_2^2. \quad (3.11)$$

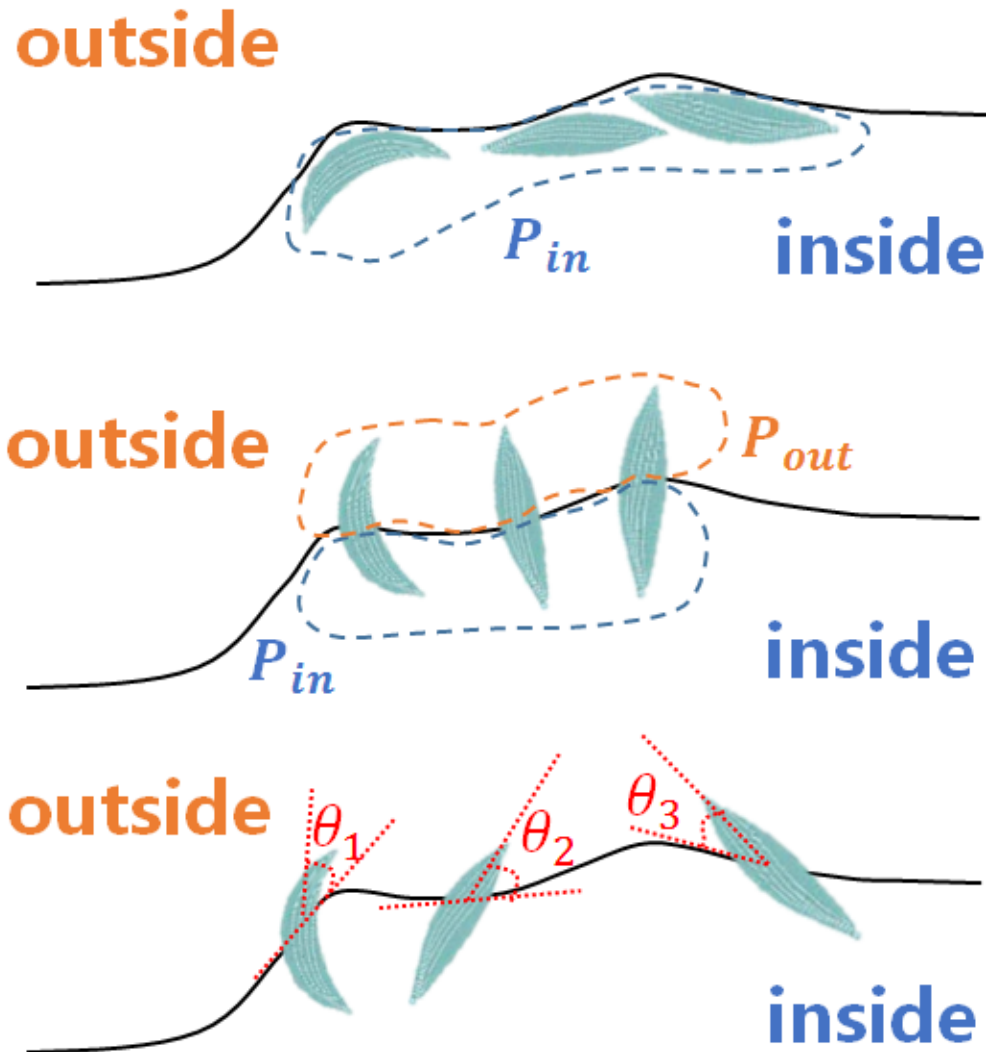


图 3.13 区域边缘部分的几种纹理分布情况。上: case 1 纹理元素主方向沿着边缘方向; 中: case 2 纹理元素跨越边缘分布; 下:  $P_{out}$  与纹理元素的基本方向和边界的切线之间的夹角密切相关。通常来说,  $\theta$  越小,  $P_{out}$  越小。

这意味着只有指定区域之外的元素才会对  $\mathcal{L}_{shape}$  有贡献。在 case1 中, 所有像素都在轮廓内, 而 case2 中的部分纹理在轮廓之外。因此 case1 的  $\mathcal{L}_{shape}$  较小。虽然实际情况中纹理元素不会完全像结果中那样分布, 但是可以在很大程度上避免出现类似 case2 的分布。

如图3.13底部所示, 显然  $P_{out}$  与纹理元素主方向和边界切线之间的角度密切相关。通常来说,  $\theta$  越小, 则  $P_{out}$  越小。因此边界附近的纹理将尽可能地像

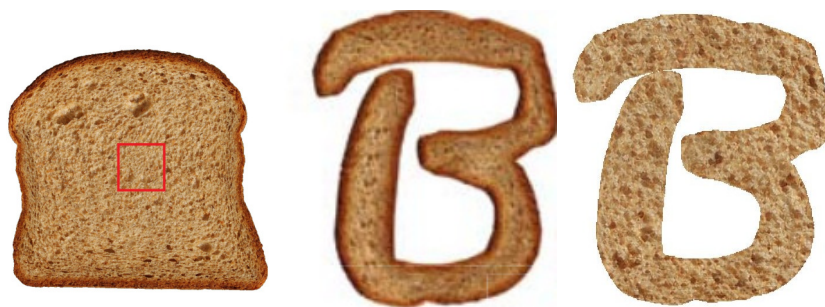


图 3.14 左：样例图案。中：Lukáč *et al.*<sup>[60]</sup> 的结果，该方法专门设计用于同时合成区域的内部和边界。右：本方法结果（以红框内的纹理片块作为输入图案）。本方法可以在目标区域中合成连续纹理，但不能以不同方式处理区域边界。

case1 一样分布，并试图保持一个较小的  $\theta$  值。

### 3.4.3 缺陷

我们的系统仍存在一些缺陷，这在一定程度上要归结于 CNN 的使用。以图3.14为例，我们的方法可以合成目标区域中的连续纹理，但是我们当前的解决方案不能同时合成具有不同样式的内部和边界，如图3.14所示。这是因为，与传统方法不同，我们当前的解决方案倾向于提取具有语义意义和高级别的抽象特征。这意味着我们的解决方案特别适合处理相同风格的图案。此外，如何在提高计算性能和避免出现瑕疵之间取得平衡是我们未来工作的另一个方向。

## 第五节 小结

本章介绍了一个基于 CNN 的风格迁移系统“DeepBrush”，该系统可以将示例图案合成到 2D 和 3D 表面，同时将风格迁移的作用域控制在用户指定形状轮廓内。

与上一章方法所用的策略一样，本方法仍然是通过设计新的图像表征对现有的进行补充和完善，使生成图像达到想要的效果。我们在图像生成过程中引入轮廓形状表征，确保可以同时保证风格图案和轮廓形状。多尺度合成策略的运用确保了生成结果的高质量 and 多样性。各种实验结果表明，该方法成功保护了图像的轮廓结构，消除了不必要的元素。再一次印证了通过设计合适的图像表征靶向地使风格迁移效果呈现特定特点这一策略是可行的。

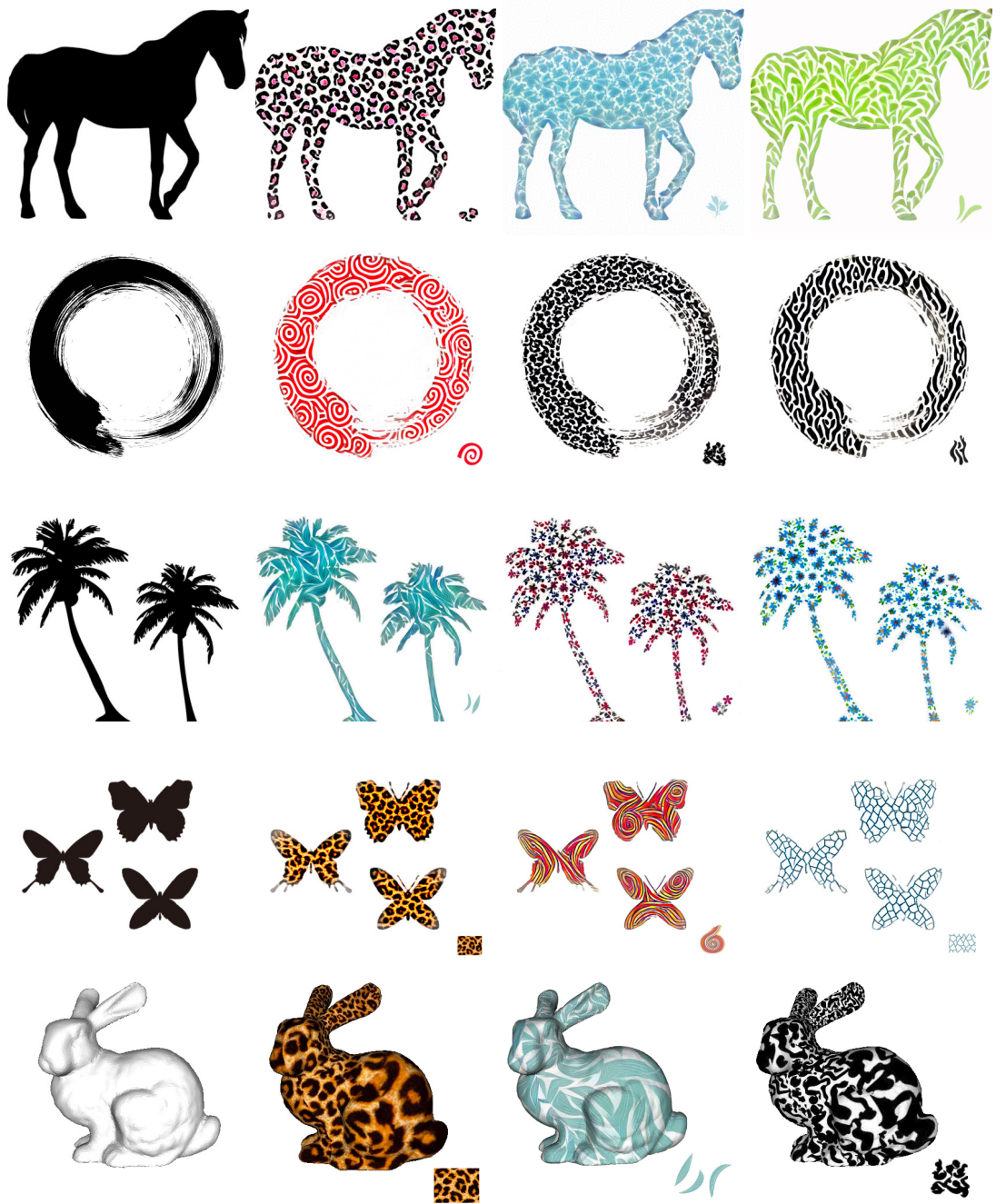


图 3.15 更多示例。

## 第四章 总结

本文关注了利用神经网络进行风格迁移这一任务，采用了一个通用策略解决了现有方法在结构保持方面存在的一些共性问题：整体结构信息保留不完整，局部细节缺失，风格迁移作用域不可控等等。

文章先简要介绍了风格迁移需要解决的三个基本问题：风格表征、内容表征、和相似度计算方法，并说明了图像表征在这类图像生成任务中的重要性。紧接着指出了现有的基于神经网络的风格迁移方法存在的一些共性问题，然后从图像表征入手，通过具体分析归纳出这些问题都可以归结到现有的图像表征上：虽然现有的取自神经网络的图像表征在描述图像风格和内容上已经足够强大，但在表征图像的一些个性特点上还是有所欠缺。进而提出可以用一种通用策略解决上述问题：设计新的图像表征对原有系统进行修正和完善，靶向地使风格迁移结果呈现某些特定的特点。

为了验证所提策略的有效性，文章通过两个方法对其进行了具体阐述：结构保持的风格迁移和 2D/3D 表面的风格迁移。既然现有方法无法保留图像整体结构和局部细节，那就人为设计全局结构表征（global-structure representation）和局部结构表征（local-structure representation）对风格迁移效果进行修正；既然神经网络风格迁移作用域不可控，那就人为设计形状轮廓表征（contour-shape representation）对作用域进行软约束。

实验结果表明，本文提出的策略是可行的，而且一系列定性比较和量化分析也对所提策略的有效性加以了印证。在全局结构提取网络和局部结构细化网络的共同作用下，生成结果成功地保留了图像结构，同时达到了艺术风格效果。在轮廓形状表征的作用下，风格迁移的作用域也得到了控制。但本文展示的方法也存在其局限性，这也将是后续工作的研究方向。同时需要指出，本文所提策略并非解决当前问题唯一方案，或许还有很多其他策略可以同样有效或者更好地解决问题。本文是想以一种通用的，统一的问题解决方式，凸显图像表征在风格迁移中所起的作用，同时希望这种问题解读方式和分析方法可以给读者更多启发。

## 参考文献

- [1] GATYS L, ECKER A S, BETHGE M. Texture synthesis using convolutional neural networks. [C] // Adv. Neural Inform. Process. Syst. 2015.
- [2] GATYS L A, ECKER A S, BETHGE M. Image style transfer using convolutional neural networks. [C] // IEEE Conf. Comput. Vis. Pattern Recog. 2016.
- [3] GATYS L, BETHGE M, HERTZMANN A, et al. Preserving color in neural artistic style transfer. [J]. *arXiv:1606.05897 [cs.CV]*, 2016.
- [4] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition. [C] // Int. Conf. Learn. Represent. 2015.
- [5] JOHNSON J, ALAHI A, FEI-FEI L. Perceptual losses for real-time style transfer and super-resolution. [C] // Eur. Conf. Comput. Vis. 2016.
- [6] CHEN D, YUAN L, LIAO J, et al. Stylebank: An explicit representation for neural image style transfer. [C] // IEEE Conf. Comput. Vis. Pattern Recog. 2017.
- [7] LI Y, FANG C, YANG J, et al. Diversified texture synthesis with feed-forward networks. [C] // IEEE Conf. Comput. Vis. Pattern Recog. 2017.
- [8] SNELGROVE X. High-Resolution Multi-Scale Neural Texture Synthesis. [C] // SIGGRAPH ASIA Technical Briefs. ACM, 2017.
- [9] HAUSNER A. Simulating decorative mosaics. [C] // ACM SIGGRAPH. 2001.
- [10] XU J, KAPLAN C S. Calligraphic packing. [C] // Proceedings of Graphics Interface. ACM. 2007: 43–50.
- [11] HUANG X, BELONGIE S. Arbitrary Style Transfer in Real-Time With Adaptive Instance Normalization. [C] // Int. Conf. Comput. Vis. 2017.
- [12] HEEGER D J, BERGEN J R. Pyramid-based texture analysis/synthesis. [C] // ACM SIGGRAPH. 1995.
- [13] JULESZ B. Visual pattern discrimination. [J]. *IRE transactions on Information Theory*, 1962, 8 (2): 84–92.
- [14] PORTILLA J, SIMONCELLI E P. A parametric texture model based on joint statistics of complex wavelet coefficients. [J]. *Int. J. Comput. Vis.*, 2000, 40 (1): 49–70.
- [15] EFROS A A, LEUNG T K. Texture synthesis by non-parametric sampling. [C] // Int. Conf. Comput. Vis. 1999.
- [16] EFROS A A, FREEMAN W T. Image quilting for texture synthesis and transfer. [C] // ACM SIGGRAPH. 2001.
- [17] HERTZMANN A, JACOBS C E, OLIVER N, et al. Image analogies. [C] // ACM SIGGRAPH. 2001.

- 
- [18] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks. [C] // Adv. Neural Inform. Process. Syst. 2012.
- [19] SIMONYAN K, VEDALDI A, ZISSERMAN A. Deep inside convolutional networks: Visualising image classification models and saliency maps. [C] // Int. Conf. Learn. Represent. 2014.
- [20] ULYANOV D, LEBEDEV V, VEDALDI A, et al. Texture Networks: Feed-forward Synthesis of Textures and Stylized Images. [C] // Int. Conf. Mach. Learn. 2016.
- [21] ULYANOV D, VEDALDI A, LEMPITSKY V. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. [C] // IEEE Conf. Comput. Vis. Pattern Recog. 2017.
- [22] GUPTA A, JOHNSON J, ALAHI A, et al. Characterizing and Improving Stability in Neural Style Transfer. [C] // Int. Conf. Comput. Vis. 2017.
- [23] DUMOULIN V, SHLENS J, KUDLUR M, et al. A learned representation for artistic style. [C] // Int. Conf. Learn. Represent. 2017.
- [24] CHEN T Q, SCHMIDT M. Fast patch-based style transfer of arbitrary style. [C] // Adv. Neural Inform. Process. Syst. Worksh. 2016.
- [25] LI Y, LIU M.-Y, LI X, et al. A closed-form solution to photorealistic image stylization. [C] // Eur. Conf. Comput. Vis. 2018.
- [26] LUAN F, PARIS S, SHECHTMAN E, et al. Deep photo style transfer. [C] // IEEE Conf. Comput. Vis. Pattern Recog. 2017.
- [27] MECHREZ R, SHECHTMAN E, ZELNIK-MANOR L. Photorealistic Style Transfer with Screened Poisson Equation. [C] // Brit. Mach. Vis. Conf. 2017.
- [28] JING Y, YANG Y, FENG Z, et al. Neural Style Transfer: A Review. [J]. arXiv:1705.04058 [cs.CV], 2017.
- [29] LI Y, WANG N, LIU J, et al. Demystifying neural style transfer. [C] // Int. Joint Conf. Artifi. Intell. 2017.
- [30] LI C, WAND M. Combining markov random fields and convolutional neural networks for image synthesis. [C] // IEEE Conf. Comput. Vis. Pattern Recog. 2016.
- [31] LI C, WAND M. Precomputed real-time texture synthesis with markovian generative adversarial networks. [C] // Eur. Conf. Comput. Vis. 2016.
- [32] WILMOT P, RISSER E, BARNES C. Stable and Controllable Neural Texture Synthesis and Style Transfer Using Histogram Losses. [J]. arXiv:1701.08893 [cs.GR], 2017.
- [33] IANDOLA F N, HANS, MOSKEWICZ M W, et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size. [J]. preprint arXiv:1602.07360 [cs.CV], 2016.
- [34] MAHENDRAN A, VEDALDI A. Understanding deep image representations by inverting them. [C] // IEEE Conf. Comput. Vis. Pattern Recog. 2015.
- [35] CHEN W, FU Z, YANG D, et al. Single-image depth perception in the wild. [C] // Adv. Neural Inform. Process. Syst. 2016.

- 
- [36] XIE S, TU Z. Holistically-nested edge detection. [J]. *Int. J. Comput. Vis.*, 2017.
- [37] ZHANG H, DANA K. Multi-style generative network for real-time transfer. [J]. preprint arXiv:1703.06953 [cs.CV], 2017.
- [38] LIN T.-Y, MAIRE M, BELONGIE S, et al. Microsoft COCO: Common objects in context. [C] // *Eur. Conf. Comput. Vis.* 2014.
- [39] KINGMA D, BA J. Adam: A method for stochastic optimization. [C] // *Int. Conf. Learn. Represent.* 2014.
- [40] LIU Y, CHENG M.-M, HU X, et al. Richer Convolutional Features for Edge Detection. [J]. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019.
- [41] WANG Z, BOVIK A C, SHEIKH H R, et al. Image quality assessment: from error visibility to structural similarity. [J]. *IEEE Trans. Image Process.*, 2004, 13 (4): 600–612.
- [42] KWATRA V, ESSA I, BOBICK A, et al. Texture optimization for example-based synthesis. [C] // *ACM Trans. Graph. Vol. 24. 3. ACM. 2005: 795–802.*
- [43] LIANG L, LIU C, XU Y.-Q, et al. Real-time texture synthesis by patch-based sampling. [J]. *ACM Trans. Graph.*, 2001, 20 (3): 127–150.
- [44] SOLER C, CANI M.-P, ANGELIDIS A. Hierarchical pattern mapping. [J]. *ACM Trans. Graph.*, 2002, 21 (3): 673–680.
- [45] WEI L.-Y, LEVOY M. Fast texture synthesis using tree-structured vector quantization. [C] // *ACM Trans. Graph.* 2000: 479–488.
- [46] KWATRA V, SCHÖDL A, ESSA I, et al. Graphcut textures: image and video synthesis using graph cuts. [J]. *ACM Trans. Graph.*, 2003, 22 (3): 277–286.
- [47] LEFEBVRE S, HOPPE H. Parallel controllable texture synthesis. [J]. *TOG*, 2005.
- [48] Jia-Bin Huang, Sing Bing Kang, Narendra Ahuja, and Johannes Kopf. Image Completion using Planar Structure Guidance. [J]. *TOG*, 2014.
- [49] ELAD M, MILANFAR P. Style Transfer Via Texture Synthesis. [J]. *IEEE Trans. Image Processing*, 2017, 26 (5): 2338–2351.
- [50] ZHOU K, WANG X, TONG Y, et al. TextureMontage. [J]. *ACM Trans. Graph.*, 2005, 24 (3): 1148–1155.
- [51] ZHOU S, CHANGYUN J, LEFEBVRE S. Topology-constrained synthesis of vector patterns. [J]. *ACM Trans. Graph.*, 2014, 33 (6): 215:1–11.
- [52] BARNES C, SHECHTMAN E, FINKELSTEIN A, et al. PatchMatch: a randomized correspondence algorithm for structural image editing. [J]. *ACM Trans. Graph.*, 2009, 28 (3): 24.
- [53] HU S.-M, ZHANG F.-L, WANG M, et al. PatchNet: a patch-based image representation for interactive library-driven image editing. [J]. *ACM Trans. Graph.*, 2013, 32 (6): 196.
- [54] BARNES C, ZHANG F.-L, LOU L, et al. PatchTable: efficient patch queries for large datasets and applications. [J]. *ACM Trans. Graph.*, 2015, 34 (4): 97:1–10.
- [55] BARNES C, ZHANG F.-L. A survey of the state-of-the-art in patch-based synthesis. [J]. *Computational Visual Media*, 2017, 3 (1): 3–20.

- 
- [56] CHI M.-T, LEE T.-Y, QU Y, et al. Self-animating images: illusory motion using repeated asymmetric patterns. [J]. *ACM Trans. Graph.*, 2008, 27 (3): 62.
- [57] CHU H.-K, HSU W.-H, MITRA N J, et al. Camouflage Images. [J]. *ACM Trans. Graph.*, 2010, 29 (3).
- [58] TONG Q, ZHANG S.-H, HU S.-M, et al. Hidden images. [C] // *Proc. NPAR. ACM.* 2011: 27–34.
- [59] LUKÁČ M, FIŠER J, BAZIN J.-C, et al. Painting by feature: texture boundaries for example-based image creation. [J]. *ACM Trans. Graph.*, 2013, 32 (4): 116.
- [60] LUKÁČ M, FIŠER J, ASENTE P, et al. Brushables: Example-based Edge-aware Directional Texture Painting. [J]. *Comput. Graph. Forum*, 2015, 34 (7): 257–267.
- [61] KUO Y.-M, CHU H.-K, CHI M.-T, et al. Generating Ambiguous Figure-Ground Images. [J]. *IEEE Trans. Vis. Comput. Graphics*, 2017, 23 (5): 1534–1545.
- [62] LIN S.-S, MORACE C C, LIN C.-H, et al. Generation of Escher Arts with Dual Perception. [J]. *IEEE Trans. Vis. Comput. Graphics*, 2018, 24 (2): 1103–1113.
- [63] SANTONI C, PELLACINI F. GTangle: a grammar for the procedural generation of tangle patterns. [J]. *ACM Trans. Graph.*, 2016, 35 (6): 182.
- [64] LU J, BARNES C, WAN C, et al. DecoBrush: drawing structured decorative patterns by example. [J]. *ACM Transactions on Graphics (TOG)*, 2014, 33 (4): 90.
- [65] KIM J, PELLACINI F. Jigsaw image mosaics. [J]. *ACM Transactions on Graphics*, 2002, 21 (3): 657–664.
- [66] KWAN K C, SINN L T, HAN C, et al. Pyramid of arclength descriptor for generating collage of shapes. [J]. *ACM Trans. Graph.*, 2016, 35 (6): 229–1.
- [67] ZOU C, CAO J, RANAWEERA W, et al. Legible compact calligrams. [J]. *ACM Transactions on Graphics (TOG)*, 2016, 35 (4): 122.
- [68] PHAN H Q, LU J, ASENTE P, et al. Patternista: learning element style compatibility and spatial composition for ring-based layout decoration. [C] // *Proceedings of the Joint Symposium on Computational Aesthetics and Sketch Based Interfaces and Modeling and Non-Photorealistic Animation and Rendering. Eurographics Association.* 2016: 79–88.
- [69] SAPUTRA R A, KAPLAN C S, ASENTE P, et al. FLOWPAK: Flow-based Ornamental Element Packing. [C] // *Proceedings of the 43rd Graphics Interface Conference. Canadian Human-Computer Communications Society.* 2017: 8–15.
- [70] ROVERI R, ÖZTIRELI A C, MARTIN S, et al. Example based repetitive structure synthesis. [J]. *Comput. Graph. Forum*, 2015, 34 (5): 39–52.
- [71] LIN T.-Y, MAJI S. Visualizing and understanding deep texture representations. [C] // *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2016: 2791–2799.
- [72] ATARSAIKHAN G, IWANA B K, UCHIDA S. Contained Neural Style Transfer for Decorated Logo Generation. [J]. 2018.

- 
- [73] LIU X.-C, CHENG M.-M, LAI Y.-K, et al. Depth-aware Neural Style Transfer. [C] // Non-Photorealistic Animation and Rendering (NPAR). 2017: 4:1–4:10. ISBN: 978-1-4503-5081-5. DOI: 10.1145/3092919.3092924.
- [74] IIZUKA S, SIMO-SERRA E, ISHIKAWA H. Globally and locally consistent image completion. [J]. ACM Trans. Graph., 2017, 36 (4): 107.
- [75] ZHOU Y, ZHU Z, BAI X, et al. Non-stationary Texture Synthesis by Adversarial Expansion. [J]. ACM Transactions on Graphics (Proc. SIGGRAPH), 2018, 37 (4).
- [76] SIMONYAN K, ZISSERMAN A. Very Deep Convolutional Networks for Large-Scale Image Recognition. [C] // International Conference on Learning Representations. 2015.
- [77] JULESZ B. Textons, the elements of texture perception, and their interactions. [J]. Nature, 1981, 290 (5802): 91.
- [78] ZHU S C, LIU X W, WU Y N. Exploring texture ensembles by efficient Markov chain Monte Carlo-Toward a “trichromacy” theory of texture. [J]. IEEE Trans. Pattern Anal. Mach. Intell., 2000, 22 (6): 554–569.
- [79] LUO W, LI Y, URTASUN R, et al. Understanding the effective receptive field in deep convolutional neural networks. [C] // Advances in Neural Information Processing Systems. 2016: 4898–4906.
- [80] GU X, YAU S.-T. Global conformal surface parameterization. [C] // Pro. SGP. 2003: 127–137.
- [81] LÉVY B, PETITJEAN S, RAY N, et al. Least squares conformal maps for automatic texture atlas generation. [J]. ACM Trans. Graph., 2002, 21 (3): 362–371.

## 致谢

与你同行的人，比你到达的方向更重要。

— 特拉克尔

感谢我的导师、挚友、和家人，

我们同行的那段岁月，

会在时光的过滤下越发动人。

## 个人简历

### 教育背景:

- 南开大学, 计算机科学与技术, 硕士, 2016/09-2019/06
- 华中科技大学, 电子信息工程, 本科, 2012/09-2016/06

### 研究生期间发表论文:

- [1] **Xiao-Chang Liu**, Ming-Ming Cheng, Yu-Kun Lai, and Paul L. Rosin. Depth-aware Neural Style Transfer. In Proceedings of NPAR17, Los Angeles, CA, USA, 2017, 10 pages.
- [2] **Xiao-Chang Liu**, Shao-Ping Lu, Jie Wang, and Ming-Ming Cheng. DeepBrush: CNN-based Structured Style Transfer on 2D/3D Surfaces. The 7th international conference on Computational Visual Media (CVM 2019), Bath, UK, 2019, 12 pages.
- [3] Ming-Ming Cheng, **Xiao-Chang Liu**, Jie Wang, Shao-Ping Lu, Yu-Kun Lai, and Paul L. Rosin. Structure-Preserving Neural Style Transfer. IEEE Transactions on Image Processing. (二轮审稿中)

### 所获荣誉:

- 硕士研究生国家奖学金, 2017 年