

一种面向移动端的图像风格迁移模型压缩算法

裴斐, 刘进锋*, 李峻河

宁夏大学信息工程学院, 宁夏 银川 750021

摘要 基于 Johoson 等的图像风格转换网络模型, 在保证网络性能的前提下, 在原有的网络结构上, 通过运用更高效的网络计算方法对原有残差网络进行优化。实验结果表明, 改进后的方法在几乎不降低图像质量的前提下, 一定程度上克服了图像风格迁移模型存储量大、计算代价高、计算资源消耗大、难以移植到移动端的问题。

关键词 图像处理; 图像风格迁移; 卷积神经网络; 深度残差网络; 模型压缩

中图分类号 TP391.4

文献标志码 A

doi: 10.3788/LOP57.061021

An Image Style Transformation Model Compression Algorithm for Mobile Terminal

Pei Fei, Liu Jinfeng*, Li Xiaohu

College of Information Engineering, Ningxia University, Yinchuan, Ningxia 750021, China

Abstract In this study, we propose an efficient network computing method based on Johoson's image style transformation network model to optimize the original residual network for ensuring suitable network performance. The conducted experiments prove that the proposed method can solve the following problems: high storage and calculation cost associated with the image style transformation network model; massive consumption of the computing resources; and difficulty with respect to the transplantation to a mobile terminal without reducing the image quality.

Key words image processing; image style transformation; convolutional neural network; deep residual network; model compression

OCIS codes 100.4996; 100.4994; 100.2000

1 引言

图像风格迁移是在保持原始图像内容结构的同时, 将图像的艺术风格进行转换, 获得风格图像的质感与美学特征, 输出的生成图像能够完美地结合图像内容和特定的风格特征。在使用神经网络方法之前, 图像艺术风格渲染的方法主要有基于笔触渲染的方法^[1]、基于图像类比的方法^[2]以及基于纹理合成的方法^[3]等。这些基于物理模型提取图像底层特征进行绘制和纹理合成的方法, 在处理具有复杂纹理和颜色的图像时, 效果并不理想, 因此实际应用范围非常有限。

随着深度学习^[4]的兴起, 2015 年 Gatys 等^[5]提

出了基于卷积神经网络(CNN)的图像风格迁移方法, 他们研究发现使用卷积神经网络的深层卷积层可以得到图像的整体结构信息(内容), 利用浅层卷积层可以得到图像的纹理信息(风格), 从而可以将图像的内容与风格进行分离, 最后只需将两幅图像的结构信息和纹理信息结合起来, 就可成功实现图像风格迁移。这种方法能自动选取模仿图像的风格, 因此应用范围广泛。

Gatys 等^[5]的工作为图像风格迁移方面的研究提供了思路, 后来的研究者在其基础上提出了许多种图像风格迁移的算法, 主要包括基于图像迭代的描述性神经方法和基于模型迭代的生成式神经方法。Johnson 等^[6]最早提出迭代优化生成模型的图

收稿日期: 2019-09-11; 修回日期: 2019-11-10; 录用日期: 2019-11-19

基金项目: 国家自然科学基金(61762073)、宁夏高等学校科学研究项目(NGY2015044)

* E-mail: 850683631@qq.com

像风格迁移方法,他们使用感知损失函数为某种特定风格训练生成模型,为提高图像风格迁移效率提供了很好的思路。Ulyanov 等^[7]也采用了类似的网络架构,并通过实验证明在生成模型训练过程中,使用实例归一化可以显著提高生成图像的质量。Zhang 等^[8]构建了一个可以训练多种风格的生成模型,实现了多风格的快速迁移。杜振龙等^[9]利用改进的循环一致性对抗网络 CycleGAN 实现图像风格迁移,提升了网络性能,提高了风格迁移生成图像的质量。Yang 等^[10]提出一种包括简化导数链的计算和加快调整的计算的方法,可为不同的能量函数选择权重,以解决 VGG 网络在传输图像样式时计算负载大的问题。

当前,移动设备正在朝着更加智能化的方向发展,智能化的应用程序也在改变着人类生活的各个方面,利用图像风格迁移的算法训练出的模型虽然性能强大,然而由于参数量庞大、存储和计算代价高等缺点,其相关的应用大多基于 PC 端,难以将其部署在受限的硬件平台上(如移动设备)。虽然云计算可以将一部分计算需求转移到云端,但是对于图像风格迁移这种高实时性的计算场景而言,云计算的带宽、延迟和全时可用性均面临着严峻的挑战,因此,对于图像风格迁移而言,虽然深度学习算法带来了性能方面巨大的提升,却因计算瓶颈而无法得到有效应用。本文基于 Johnson 等^[6]提出的实时风格转换网络模型,在保证网络性能的前提下,在原算法网络结构的基础上,通过设计更高效的网络计算方式来改进原有残差网络的单元结构,实验结果表明,所提出的改进方法在一定程度上克服了图像风格迁移模型存储和计算代价高、计算资源巨大、难以移植到移动端等问题。从实时响应的需求来看,本文提出的图像风格迁移系统因使用本地化运行处理风格模型,无需将相关图像、模型参数等传输到服务器,不仅可以节省传输时间,加快运行速度,还可以保护用户的隐私。

2 图像风格迁移工作原理

图像风格迁移的难点是分别提取图像的内容和风格,并把它们结合起来。期望能找到一种图像表示方法,可以将图像的内容结构和艺术风格进行分离,从而提取出图像的语义内容,然后用不同风格渲染图像的语义内容。将卷积神经网络引入图像风格迁移领域的基础在于深度卷积神经网络具有较强的特征提取能力,经过训练的神经网络可视为良好的

特征提取器,利用深层卷积层可以得到图像的整体结构信息(内容信息),利用浅层卷积层可以得到图像的纹理信息(风格信息),这些信息正是实现风格迁移的基础,并且各层次之间的紧密联系和空间信息特别适用于图像的处理和理解。

图像风格迁移实质上涉及纹理建模与图像重建两个领域,这两个领域的处理方法组合产生了多种图像风格迁移算法,比较典型的有 Gatys 等^[5]提出的基于在线图像优化的慢速算法和 Johnson 等^[6]提出的基于模型迭代的快速算法。

基于在线图像优化的慢速算法的核心思想是在图像经过预训练的 VGG 网络^[11]的特征图上计算 Gram 矩阵,利用得到的 Gram 矩阵来表示纹理,其实质是利用了二阶统计量来对纹理进行建模。通过 Gram 矩阵对图像中的风格进行建模和提取后,再利用图像重建方法让重建后的图像以梯度下降的方式更新像素值,使其 Gram 矩阵接近风格图的 Gram 矩阵(即风格相似),其 VGG 网络的高层特征表达接近内容图的特征表达(即内容相似),最终重建出来的结果图就既拥有风格图的风格,又有内容图的内容。该方法速度慢、资源占用量大。

基于模型迭代的快速算法主要为了解决速度慢的问题,核心思想是预先训练前向网络来解决计算量大、速度慢的问题,利用基于离线模型优化的快速图像重建方法对风格化结果进行重建。因为本文方法是在这种方法的基础上改进得到的,所以本节重点介绍这种算法的构成及网络模型。

图 1 为模型迭代优化风格迁移算法的整体结构图,整个结构包括图像转换网络和损失计算网络。图像转换网络是多层卷积神经网络,其主要作用是将一张输入的原始图片转换成为一张生成图片。损失计算网络本质上是 VGG-16 网络,主要用于计算图像转换网络生成的图像与输入的风格图像和内容图像之间的损失大小。通过计算生成图像与内容图像的内容损失以及生成图像与风格图像的风格损失大小来判断生成图像的质量。通过不断计算来减少损失,反向传播到图像转换网络并对其进行优化,最终得到合格的图像风格转换模型,而这个模型就可以将图像转换到指定的风格。

图像转换网络的目的是对于给定的某种艺术风格,输入任何一张内容图像,都可以转换出带有该艺术风格的合成图像。因此在训练图像转换网络时,需要大量的内容图片和一张风格图片,然后训练出能把任何图片变成这种风格的网络。图像转换网

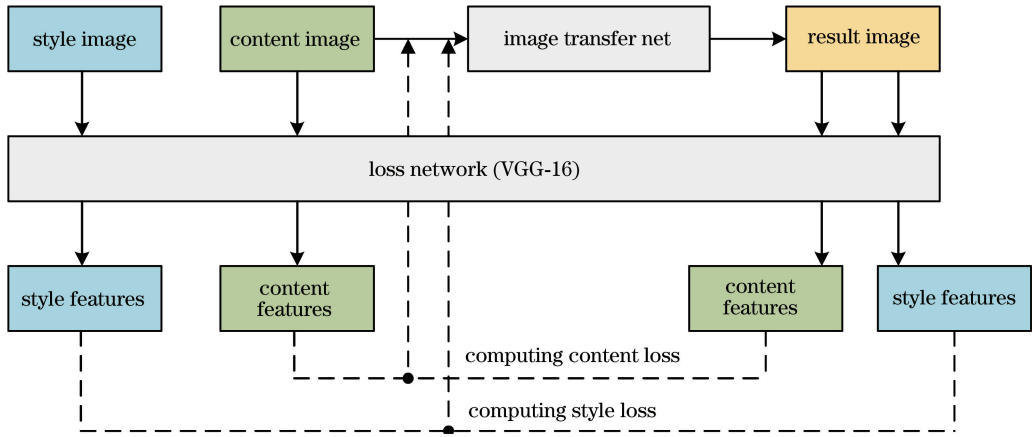


图 1 模型迭代优化风格迁移算法整体结构

Fig. 1 Overall structure of model iterative optimization style transformation algorithm

络的结构如图 2 所示,主要包含三部分:下采样卷积层(conv)、深度残差层(res)和上采样卷积层(deconv)。深度残差层是图像转换网络中的重要部分。

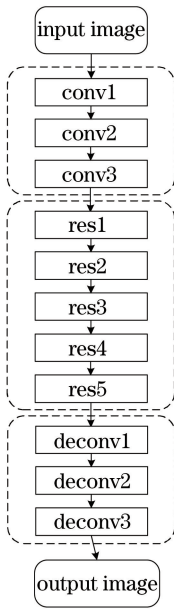


图 2 图像风格迁移网络结构

Fig. 2 Image style transformation network structure

通过上述方法训练出来的单个风格神经网络模型大小约为 20 MB,只能将少量模型部署在移动端,而风格模型部署数量增多,将对移动设备的功率、存储能力、计算能力等产生巨大的挑战。

如表 1 所示,通过分析训练出的对应风格网络模型,发现整个网络模型的参数主要集中在残差层(res),单层的残差层参数量在整个网络模型结构参数总量中的占比为 17.6%,五层残差网络的参数量在整个网络模型结构参数总量中的占比

表 1 各层网络的参数量

Table 1 Number of network parameters at each layer

Layer	Number of parameters	Ratio of number of parameters / %
Conv1	7776	0.4
Conv2	18432	1.1
Conv3	73728	4.4
Res1	294912	17.6
Res2	294912	17.6
Res3	294912	17.6
Res4	294912	17.6
Res5	294912	17.6
Deconv1	73728	4.4
Deconv2	18432	1.1
Deconv3	7776	0.4

为 88%,原有残差层单元结构的网络计算方式(卷积方式)导致残差层参数量庞大,庞大的参数量不仅影响模型推理的速度,而且会占用移动端的存储资源。在这样的形势下,模型小型化与加速成为亟待解决问题。

3 图像转换网络结构改进

本文在 Johnson 等^[6]提出的基于模型迭代的快速算法网络结构基础上,以服务于移动端推理、降低参数量和计算量、保证网络性能为原则,对图像风格迁移模型按移动端推理的实际需求进行一定的改进,以解决模型压缩的问题。图 3(a)所示为原始网络结构中使用的 ResNet^[12]单元结构,其主要由两层 3×3 的卷积层组成。图 3(b)为 ResNet 的另一种三层的网络结构图,其主要由两层 1×1 的卷积层和一层 3×3 的卷积层堆叠组成。将图 3(b)所示的三层网络结构替代原来的网

络结构,简化了网络的设计,在增加网络深度的同时,显著减少计算量。网络参数量的计算可表示为

$$P = NC \cdot n^2, \quad (1)$$

式中: P 为参数总量; N 为卷积核的数量,即输出通道数; C 为输入通道数; n^2 为卷积核大小。假设输入的通道数为256,卷积核大小为 3×3 ,输出通道数为256,由(1)式计算的整个单元结构的参数总量(因为有两层,所以要乘以2)可知,整体参数总量为 $3 \times 3 \times 256 \times 256 \times 2 = 1179648$ 。

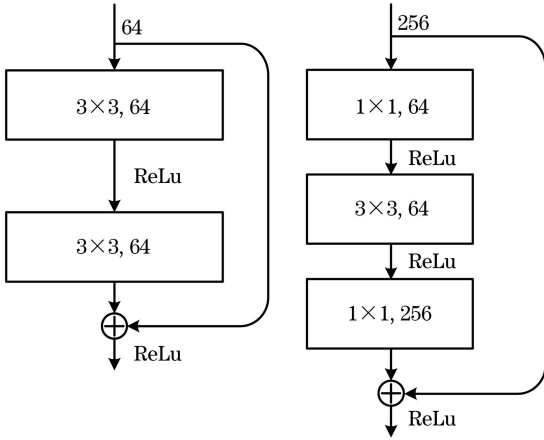


图3 改进前、后的网络结构。(a)改进前;(b)改进后

Fig. 3 Network architecture before and after improvement.

(a) Before improvement; (b) after improvement

减小网络参数的常用方法是对残差层进行计算优化,减少输入到 3×3 卷积核的通道数量,即(1)式中 C 的数值。如图3(b)所示,首先在一个 1×1 卷积层中把256维降低到64维,然后再通过 1×1 的卷积层增大到256维。 1×1 的卷积核主要用于调节通道数,对不同的通道上的像素点进行线性组合,然后进行非线性化操作,可以实现升维和降维的功能。当输入和输出的通道数很大时,图3(a)所示结构的卷积核参数会变得很大,而如图3(b)所示,在加入 1×1 卷积层后可以降低输入通道的数量,卷积核参数、运算复杂度也相应降低。改进后的结构既保持了精度又减少了计算量,由(1)式可计算整体的参数量: $1 \times 1 \times 64 \times 256 + 3 \times 3 \times 64 \times 64 + 1 \times 1 \times 64 \times 256 = 69632$ 。

通过上述理论分析结果可以得出,采用图3(b)所示的残差单元结构,可使得每个单元的参数量大约减少了94%。当输入通道数为128时,网络压缩前输出通道数仍为128,压缩后输出通道数降低到64,分别统计压缩前、后每一层的参数量,结果如表2所示。

表2 模型压缩前、后参数数量的对比

Table 2 Comparison of number of parameters before and after model compression

Layer	Number of parameters		Ratio of compression / %
	Before compression	After compression	
Conv1	7776	7776	0
Conv2	18432	18432	0
Conv3	73728	73728	0
Res1	294912	53248	81.9
Res2	294912	53248	81.9
Res3	294912	53248	81.9
Res4	294912	53248	81.9
Res5	294912	53248	81.9
Deconv1	73728	73728	0
Deconv2	18432	18432	0
Deconv3	7776	7776	0
Total	1674432	363712	78.3

4 实现结果及分析

4.1 压缩模型的性能

在训练参数和实验环境与原有方法一致的情况下,本实验通过使用改进的网络结构训练2个周期后,实现了对风格模型压缩的目标,模型压缩前、后的实验结果如表3所示。移动设备的计算资源和能耗预算都有限,因此模型压缩对于在移动设备上部署神经网络模型至关重要,深度神经网络压缩与加速性能的主要测评标准是压缩比和加速比^[13]。假设 λ 和 λ^* 分别为原始模型 M 和压缩后模型 M^* 所有参数所占的内存开销,那么模型的压缩比 C_r 可表示为

$$C_r(M, M^*) = \frac{\lambda}{\lambda^*}. \quad (2)$$

假设 γ 和 γ^* 分别为原始模型 M 和压缩后模型 M^* 推测整个网络的时间,那么模型的加速比 S_r 可表示为

$$S_r(M, M^*) = \frac{\gamma}{\gamma^*}. \quad (3)$$

为了评价风格模型的压缩性能,将模型的压缩比与加速比作为深度神经网络改进前、后的压缩与加速性能的评价指标。表3分别从压缩前、后的风格模型中随机选取了其中4种风格模型的相关实验数据,压缩前wave、feathers、mosaic 3种模型的尺寸为21.1 MB,模型推理耗时为6~7 s,压缩后模型尺寸减小到4.4 MB,模型推理耗时范围为4~5 s,压缩后的模型在运行效率上有着绝对的优势,

表3 压缩前、后模型相关数据
Table 3 Data related to the compressed model

Model	Before compression		After compression		Compression ratio	Speed-up ratio
	Size /MB	Time /ms	Size /MB	Time /ms		
Wave	21.1	6937	4.4	4590	4.56	1.51
Feathers	21.1	6851	4.4	5314	4.56	1.29
Mosaic	21.1	6950	4.4	4421	4.56	1.57
Denoised	21.1	4420	4.4	3571	4.56	1.23

完全可以在产品端落地。

为了验证本文方法的可行性,开发了一个图像风格迁移移动端 App,并采用服务器端训练+移动端推理的模式。在服务器端进行训练,待模型训练好后,将其下载到本地手机,即可在手机上实现本地图像的风格转换。

用户选择本机相册或拍摄内容图像,系统提示用户选择想要迁移的风格类型,如果是新的风格类型,需要在服务器端进行网络模型的训练,训练好后下载到移动端,这个过程自动实时完成。移动端获取选定风格模型参数后进行风格迁移,最后显示生成的图像。

本研究选择一些常见品牌的手机,分别选用毕加索、莫奈等典型风格对实际运行效果进行测试,测试指标有安装耗时、CPU 占用率等,部分测试数据如表 4 所示。

表 4 移动端测试结果

Table 4 Test results at mobile terminal

Brand	Model	Installation	Initialization	CPU
		time /s	time /s	usage /%
Huawei	P10 Plus	3.85	0.83	1.00
Huawei	Honor 9	3.31	0.72	0.13
Huawei	Honor 7X	4.38	1.16	0.82
	360 N5S	5.71	0.97	1.07
Vivo	V173CA	5.15	1.56	1.78
Sony	Xperia XA	9.47	1.38	4.27

4.2 图像质量评估

对图像质量的评价分为客观评价和主观评价。目前对图像风格迁移技术合成图像的质量评价尚处于起步阶段,主要是因为生成的风格化图像不仅反映了风格图像作者的思想倾向、美术主张和创作方法,还反映了图像的内容、结构信息。主观评价的方法主要依赖于人的知识与理解,每个人都有自己的品味,对同一张图像的评价各不相同,导致误差较大。客观评价是借助于某些能精确、自动感知图像质量的数学模型给出的参数指标来对图像质量作出

评价。

对压缩后模型的图像质量进行评价是衡量压缩后模型性能非常重要的指标,以此来验证压缩后模型的保真情况。本研究利用 NIMA 方法^[14]对图像质量进行评测。NIMA 是谷歌发布的一种客观图像质量评测方法,能够从直接观感(技术角度)与吸引程度(美学角度)评测图像质量。NIMA 方法由两个卷积神经网络训练的模型组成,主要用来预测图像的美学质量和技术质量,这种方法既考虑了风格化图像所表达的情感,又兼顾了图像的语义信息,从定性、定量以及算法时间三个角度对图像质量进行评价,很好地契合了图像质量的评价思路。NIMA 对图像的评分在 1~10 分的范围内,得分越高,表示图像质量越好。

本研究分别对网络模型压缩前和压缩后的 45 幅风格迁移图像进行评分,图 4 所示为相同图像网络模型压缩前、后的评分,图 5 所示为不同风格化图像的评分。从图 4 和图 5 可以看出,由 NIMA 方法给出的图像质量评分范围为 4.0~5.0,而 AVA Dataset 标准数据集^[15]的图像质量评分范围为 5.0~6.0。综上所述,网络模型压缩前和压缩后的图像评分十分接近,说明与改进前模型相比,改进后网络模型的保真能力基本没有损失;经过图像风格迁移生成的图像质量已经很接近由专业摄影师拍摄的图像,从侧面证明了本文算法的可靠性。

5 总结

随着深度学习与图像处理技术的日益发展,采用深度神经网络处理图像问题的方法已经在越来越多的领域中得到应用。本文实现了基于卷积神经网络图像风格化模型的压缩,与目前主流的风格化模型进行对比,发现所提出的压缩后模型在网络性能几乎没有损失的情况下,极大地降低了网络的参数量,为向移动端的移植提供了参考。

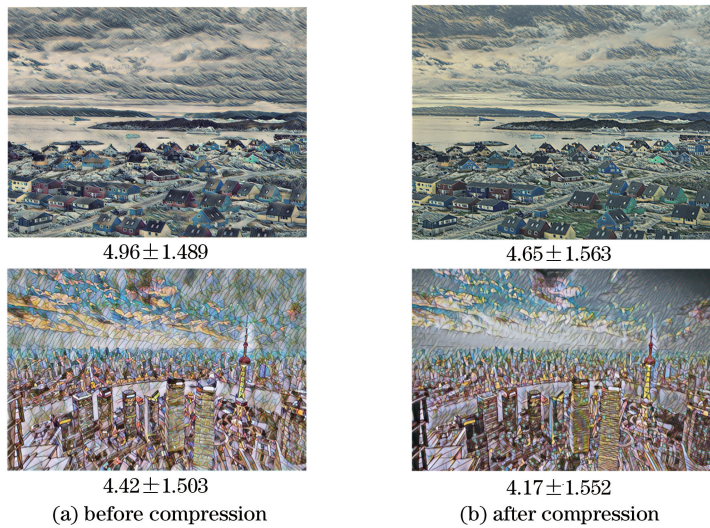


图 4 网络模型压缩前、后风格迁移评分。(a)压缩前;(b)压缩后

Fig. 4 Style transformation scores before and after network model compression. (a) Before compression; (b) after compression

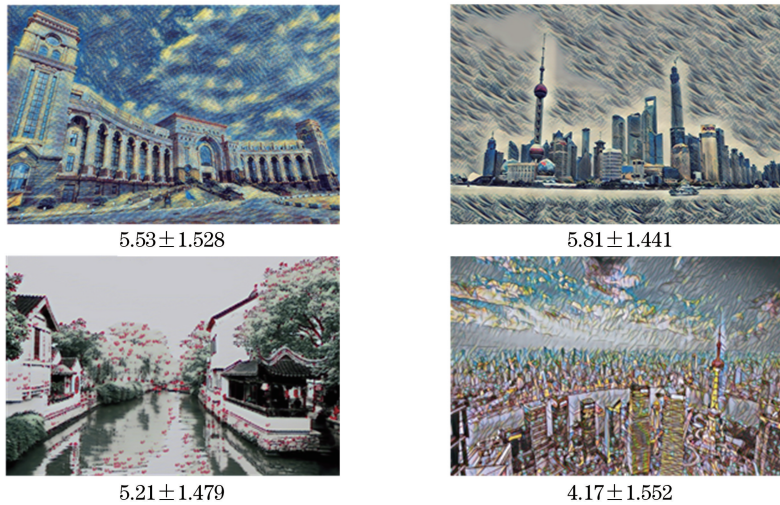


图 5 网络模型压缩后不同风格的图像迁移评分

Fig. 5 Transformation scores of image with different styles after network model compression

参 考 文 献

[1] Hertzmann A. Painterly rendering with curved brush strokes of multiple sizes[C]//Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, July 19-24, 1998, Orlando, Florida, USA. New York: ACM, 1998: 453-460.

[2] Hertzmann A, Jacobs C E, Oliver N, et al. Image analogies [C] // Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, August 12-17, 2001, Los Angeles, California, USA. New York: ACM, 2001: 327-340.

[3] Portilla J, Simoncelli E P. A parametric texture model based on joint statistics of complex wavelet coefficients[J]. International Journal of Computer Vision, 2000, 40(1): 49-70.

[4] Mao Y H, Gui X L, Li Q, et al. Study on application technology of deep learning[J]. Application Research of Computers, 2016, 33(11): 3201-3205.

毛勇华, 桂小林, 李前, 等. 深度学习应用技术研究[J]. 计算机应用研究, 2016, 33(11): 3201-3205.

[5] Gatys L A, Ecker A S, Bethge M. A neural algorithm of artistic style[EB/OL]. (2015-09-02) [2019-09-10]. <https://arxiv.xileisou.top/abs/1508.06576>.

[6] Johnson J, Alahi A, Li F F. Perceptual losses for real-time style transfer and super-resolution[M] // Leibe B, Matas J, Sebe N, et al. Computer vision-

- ECCV 2016. Lecture notes in computer science. Cham: Springer, 2016, 9906: 694-711.
- [7] Ulyanov D, Vedaldi A, Lempitsky V. Instance normalization: the missing ingredient for fast stylization [EB/OL]. (2017-11-06) [2019-09-10]. <https://arxiv.xilesou.top/abs/1607.08022>.
- [8] Zhang H, Dana K. Multi-style generative network for real-time transfer [EB/OL]. (2017-11-16) [2019-09-10]. <https://arxiv.xilesou.top/abs/1703.06953>.
- [9] Du Z L, Shen H Y, Song G M, et al. Image style transfer based on improved CycleGAN [J]. Optics and Precision Engineering, 2019, 27(8): 1836-1844. 杜振龙, 沈海洋, 宋国美, 等. 基于改进 CycleGAN 的图像风格迁移 [J]. 光学精密工程, 2019, 27(8): 1836-1844.
- [10] Yang F W, Lin H J, Yen S H, et al. A study on the convolutional neural algorithm of image style transfer [J]. International Journal of Pattern Recognition and Artificial Intelligence, 2019, 33(5): 1954020.
- [11] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [EB/OL]. (2015-04-10) [2019-09-10]. <https://arxiv.xilesou.top/abs/1409.1556>.
- [12] He K M, Zhang X Y, Ren S Q, et al. Deep residual learning for image recognition [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, June 26-July 1, 2016, Las Vegas, Nevada. New York: IEEE, 2016: 770-778.
- [13] Ji R R, Lin S H, Chao F, et al. Deep neural network compression and acceleration: a review [J]. Journal of Computer Research and Development, 2018, 55(9): 1871-1888. 纪荣嵘, 林绍辉, 晁飞, 等. 深度神经网络压缩与加速综述 [J]. 计算机研究与发展, 2018, 55(9): 1871-1888.
- [14] Talebi H, Milanfar P. NIMA: neural image assessment [J]. IEEE Transactions on Image Processing, 2018, 27(8): 3998-4011.
- [15] Murray N, Marchesotti L, Perronnin F. AVA: a large-scale database for aesthetic visual analysis [C] // 2012 IEEE Conference on Computer Vision and Pattern Recognition, June 16-21, 2012, Providence, RI, USA. New York: IEEE, 2012: 2408-2415.