**基于机器学习的视频风格迁移**

陈易磊 G161105 张为 G161140

**摘要：**风格迁移的目标是将一张图片的风格转化为另一张图片的风格，同时不损失原有图片的基本内容信息。将风格迁移的算法应用在视频上，我们可以创造出名画风格的视频。近期的研究着重于利用卷积神经网络进行图片的风格迁移，其中基于碎片的风格迁移算法在视觉效果和计算效率上相较其它算法更优。在本论文中我们将此算法应用于视频，进行逐帧的风格迁移，并采用超分辨率算法，改善了迁移后的视频的清晰度。我们详细阐释了Chen和Schmidt的风格迁移算法，对其并未解释清楚的部分进行了补充，也解释了我们采用的超分辨率算法。经测试，15秒的视频共需要约半小时进行处理。处理后的视频不仅细节信息没有过多丢失，视频的连贯性得以保证，噪点也较少，而且在一定程度上也解决了风格迁移后分辨率过低的问题。此结果也有一定的扩展性。我们可以考虑进行照片级真实的视频风格迁移，例如将白天拍摄的视频转化为夜景。

**关键词：**计算机视觉；卷积神经网络；视频风格迁移

**第1章 引言**

风格迁移的目标是将一张图片的风格转化为另一张图片的风格，同时不损失原有图片的基本内容信息。利用风格迁移我们可以模仿世界名画的绘画风格，利用自己的图片，创造出自己独有的画作，可以给人以视觉上的美感。而将风格迁移的算法应用在视频上，我们也可以创造出名画风格的视频，让画“动起来”。

在图片的风格迁移方面已有一些优良的算法。 最早的研究采用纹理合成算法进行纹理风格迁移[1]。Elad和Milanfar扩展了Kwatra等人的纹理合成算法[2]，也提出了利用纹理合成进行风格迁移的算法[3]。Hertzmann等人提出“图片类推”的方法[4]，通过预先给定的图片对的对应方式来合成新的图片，也可应用于纹理合成。Gatys等人于2015年最早采用卷积神经网络(Convolutional neural networks, CNN)进行风格迁移[5]，使用格拉姆矩阵表示图片的风格，取得了比以往更好的成果。之后也有许多围绕CNN展开的风格迁移算法。Li等人提出在CNN编码的特征图上进行白化和上色操作(whitening and coloring transforms, WCTs)，之后使用训练好的解码器还原图片，进行风格迁移[6]。后来Li等人改进了WCT算法，添加了平滑化操作，在相片级真实风格迁移上取得了优异的成果[7]。Luan等人也提出了相片级真实风格迁移的算法[8]，采用必要的正则项，对随机噪音图片进行优化，得到最终风格迁移的图片。但这两个算法都侧重于相片级真实风格迁移，与艺术图片风格迁移算法的侧重点不同，因此在下文将不再讨论这两个算法。Chen和Schmidt于2016年提出基于碎片的风格迁移算法[9]，没有采用传统的格拉姆矩阵表示图片的风格，而是直接在CNN输出的特征图上进行碎片的交换，得到新的特征图，之后可以选择使用反向CNN或基于梯度的优化的方式还原输出的图片。该算法在输出图片的美感和合成速度上都优于先前的算法。

在视频风格迁移方面，Ruder等人基于Gatys等人提出的图片风格迁移的算法[10]，提出了视频的风格迁移算法[11]。Ruder后来的改进算法不仅可以进行视频风格迁移，还可进行球形图片的风格迁移[12]。

在本研究中，我们采用Chen和Schmidt提出的图片风格迁移算法[9]，将其应用于视频上，将视频逐帧进行风格迁移，让画“动起来”，得到了画作风格的视频。由于受到标准的VGG-19神经网络[13]的限制，合成的逐帧图片分辨率较低，我们采用了Mao等人的超分辨率算法[14]，将像素数量扩大为原来的四倍，使得视频更加清晰。我们合成的视频几乎没有噪点，相邻的帧之间没有因合成而造成的不自然偏移，原视频的细节信息也得以在大体上保留。

本论文的结构如下：第2章阐释了Chen和Schmidt提出的对于图片的风格迁移算法[9]细节，并对两作者未清晰解释的部分进行补充。第3章阐释了Mao等人提出的超分辨率算法[14]的细节。第4章解释了我们将第2章和第3章应用于视频上的方式。第5章总结了全文并提出了我们的展望。

**第2章 风格迁移**

本节中将阐释Chen和Schmidt提出的对于图片的风格迁移算法[9]细节，并对两作者未清晰解释的部分进行补充。

**2.1 碎片交换**

输入的内容图片和风格图片（RGB格式）分别用和表示，用表示从输入的RGB图片到经过卷积神经网络处理的特征图的映射，通过VGG-19[13]的计算可得到内容图片的特征图和风格图片的特征图。碎片交换的过程即在这两个特征图上做如下操作：

1. 分别在内容图片和风格图片的特征图上提取和个碎片，表示为和，保证碎片的集合能够覆盖整个特征图，且提取的碎片之间有足够的重叠。
2. 对于每一个内容特征图的碎片，利用互相关算法，将其替换为与其最相近的风格特征图的碎片，作为输出特征图的碎片，如公式(1)所示：

(1)

1. 通过对中重叠部分求平均值得到，即CNN在输出图像上预期的特征图。

**2.2 并行处理优化**

为了简化运算，Chen和Schmidt将2.1中所述的碎片交换运算变换为以下三步：

1. 二维卷积；
2. 求argmax；
3. 二维卷积转置（又称反卷积）。

设的大小为，碎片大小为，表示在在位置提取的包含所有个频道的一个碎片，即。对于公式(1)，不变，因此公式(1)等价于公式(2)：

(2)

因此张量可以由一次以为卷积核，以作为输入的二维卷积操作完成。得到的张量K的大小为。

接下来，进行求argmax的操作，如公式(3)所示。

(3)

最后一步以为卷积核，以作为输入，进行二维卷积转置操作，即可达到公式(2)的效果。

然而卷积转置的操作将重叠部分的进行了加和，因此我们需要进行除以重叠数量的操作以达到平均的效果。因此我们也不需要保证公式(3)中的argmax有唯一解，因为取平均后argmax的多解可被视为添加了更多重叠的碎片。

**2.3 还原风格化图片**

在得到之后，需要还原输出图像的RGB表示，但由CNN的映射方式并不能严格推导出反向的映射，因此需要基于梯度的优化或训练反向CNN网络来得到风格化的图像。

2.3.1 基于梯度的优化

Chen和Schmidt使用了均方误差函数[9]并定义优化目标如公式(4)所示：

(4)

其中表示在图片的特征图误差和总变化量误差之间的平衡。项表示总变化量误差，如公式(5)所示：

(5)

这一正则项也在其它图片生成方法[15]中有所体现。

因为由CNN产生，因此公式(4)是一阶可微的，以其为目标的优化即可使用基于梯度的优化。

2.3.2 反向CNN

由于基于梯度的优化速度较慢，而反向CNN一经训练便可以更高的速度还原风格化的图片，因此在视频的逐帧风格迁移中较为必要。我们希望训练的反向CNN可以对于每个图片都能达到损失函数最小，即逼近优化的结果。反向CNN的结构采取VGG-19的镜像结构，即将卷积层替换为卷积转置，池化层替换为反池化。由于卷积转置和反池化并不与卷积和池化完全对应，因此我们需要使用已有图片对反向CNN网络中的参数进行预训练，得到最优的结果。

**第3章 超分辨率**

受到预训练的VGG-19网络的限制，输出图片的大小为像素，相对较不清晰。因此我们采用了Mao等人的超分辨率算法[14]，将图片长和宽均扩大为原来的两倍，得到更加清晰的图像。本节中将阐释此算法的细节。

Mao等人采用的是对称的CNN和反向CNN配合对称的跨层连接。对称的跨层连接可以使得反向CNN网络不仅从高层抽象的特征还原图片，由此去掉噪点，也能从低层更加细节化的特征还原图片的细节，达到保存图片细节的目的。网络由低清-高清图片对进行训练，可以达到输入低清图片，输出高清图片的效果。

**第4章 算法在视频上的应用**

在Ubuntu 16.04操作系统上，我们使用在视频处理方面广泛应用的ffmpeg插件，将视频按帧分离。接下来我们对于每一帧的图像应用第2章和第3章所述的算法，得到预期输出视频的逐帧图片。最后我们再次使用ffmpeg插件将逐帧图片合成输出的视频。

**第5章 结果**



(a)输入的内容图片 (b)输入的风格图片



(c)输出的风格化图片

图1 运用基于梯度的优化来进行风格迁移的结果

在运用基于梯度的优化（见2.3.1）来进行风格迁移后，已经可以得到具有画作风格的结果，如图1所示。在使用反向CNN（见2.3.2）进行处理后，图片风格的变化与前者相比出入不大，但处理速度得到大幅提升，平均0.5秒即可完成一张图片的风格转化。在采用了第3章所述的超分辨率算法后，图片的分辨率得到了大幅改善，而且并没有失真，使用肉眼无法区分两者（见图2）。通过快速风格迁移算法与超分辨率算法对视频进行处理后，我们得到了一段梵高风格的延时摄影[[1]](#footnote-1)。视频约15秒长，处理用时为半小时左右。



(a)处理前的图片 (b)处理后的图片

图2 使用超分辨率算法[14]处理前后的图片

**第6章 结论**

本论文展示了将图片风格迁移算法[9]应用于视频的风格迁移的结果。经过逐帧的风格迁移，我们得到了画作风格的视频。我们又采用逐帧的超分辨率算法[14]，使得视频更加清晰。合成的视频不仅细节信息没有过多丢失，视频的连贯性得以保证，噪点也较少，而且在一定程度上也解决了风格迁移后分辨率过低的问题。此结果也有一定的扩展性。我们可以考虑使用真实的照片作为风格图片，这样我们可以进行照片级真实的视频风格迁移，例如将白天拍摄的视频转化为夜景。

但本文所展示的结果仍存在不足。虽然风格迁移后分辨率过低的问题得到了一定程度上的解决，但是仍未达到一般的视频的720p或1080p的清晰度。我们考虑在未来的工作中通过调整所使用的卷积神经网络的超参数，或在从特征图还原回输出图片的过程中调整输出图片大小等方式，得到更加高清的画作风格化视频。

**参考文献**

[1] Efros A A, Freeman W T. Image quilting for texture synthesis and transfer[C]// Conference on Computer Graphics and Interactive Techniques. ACM, 2001:341-346.

[2] Kwatra V, Essa I, Bobick A, et al. Texture optimization for example-based synthesis[C]// ACM SIGGRAPH. ACM, 2005:795-802.

[3] Elad M, Milanfar P. Style Transfer via Texture Synthesis.[J]. IEEE Transactions on Image Processing, 2017, 26(5):2338-2351.

[4] Hertzmann A, Jacobs C E, Oliver N, et al. Image analogies[C]// Conference on Computer Graphics and Interactive Techniques. ACM, 2001:327-340.

[5] Gatys L A, Ecker A S, Bethge M. Texture Synthesis Using Convolutional Neural Networks[J]. 2015, 70(1):262-270.

[6] Li Y, Fang C, Yang J, et al. Universal Style Transfer via Feature Transforms[J]. 2017.

[7] Li Y, Liu M Y, Li X, et al. A Closed-form Solution to Photorealistic Image Stylization[J]. 2018.

[8] Luan F, Paris S, Shechtman E, et al. Deep Photo Style Transfer[J]. 2017:6997-7005.

[9] Chen T Q, Schmidt M. Fast Patch-based Style Transfer of Arbitrary Style[J]. 2016.

[10] Gatys L A, Ecker A S, Bethge M. Image Style Transfer Using Convolutional Neural Networks[C]// Computer Vision and Pattern Recognition. IEEE, 2016:2414-2423.

[11] Ruder M, Dosovitskiy A, Brox T. Artistic Style Transfer for Videos[J]. 2016:26-36.

[12] Ruder M, Dosovitskiy A, Brox T. Artistic Style Transfer for Videos and Spherical Images[J]. 2017.

[13] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2014.

[14] Mao X J, Shen C, Yang Y B. Image Restoration Using Convolutional Auto-encoders with Symmetric Skip Connections[J]. 2016.

[15] Johnson J, Alahi A, Li F F. Perceptual Losses for Real-Time Style Transfer and Super-Resolution[J]. 2016:694-711.

**致谢**

首先，在此要特别感谢指导老师张予瑶老师的热情关怀和悉心指导。从研究的选题、构思，到论文的完成的过程中，张老师提出了许多宝贵的意见和建议，使得我们少走了许多弯路，在此表示真诚的感谢。

在研究的进行过程中，我们也得到了许多同学的宝贵建议，在此一并致以诚挚的谢意。

感谢所有关心、支持、帮助过我们的良师益友。

最后，向在百忙中抽出时间对本文进行评审并提出宝贵意见的各位专家表示衷心的感谢！

1. 处理后的视频位于https://pan.baidu.com/s/16-bxijd3XKGylDGmCxW8lQ [↑](#footnote-ref-1)