混合风格的快速图像风格转换

图像风格转换在于将原样本空间的图像转换到新样本空间，即保留高维的图像内容特征同时，在低维特征上渲染出不同的纹理、笔触和色彩。基于前馈网络的快速图像风格转换方法可以实现即时的风格渲染，但受限于网络模型与转换风格的设定关系。本文首先改进原快速图像风格转换网络，使用最新的零值填充、上采样和正则化等技术，以提高视觉质量。其次，在改进网络的基础上，通过训练网络来自适应地学习不同风格损失的权重参数，将单一样式扩展到多种混合风格样式。根据实验结果，该网络模型能够不但能够有效地转换单一风格，而且可以将多种风格混合渲染到原内容图像。

关键字

图像风格转换，快速，混合风格

1 引言

仿作是模仿一种艺术风格地作品。计算机视觉和最近的机器学习领域一直有研究如何自动化地学习图像的模式特征，即在一幅图像上渲染出特定的绘画风格。该研究任务被称作图形风格转移，该目标与纹理合成有密切关系，后者假定风格图像的像素在某种尺度上具有固定的分布，并试图捕获此种像素之间的统计关系，而前者图像风格转移在纹理合成的同时也试图保留内容图像的语义信息。

基于深度神经网络优化的图像风格转换方法，首次借助深度神经网络提取图像多维特征的能力，将图像内容与风格进行分离。通过数值化图像间的内容和风格差异，该方法迭代优化初始的噪声图像，以最大限度拟合风格图像的风格特征和内容图像的内容特征。在此基础上，基于前馈网络的图像风格转换方法，以前馈网络参数的形式，学习保留风格图像的风格特征。这样对于任意输入的内容图像，都可以及时对其渲染出特定的风格，克服了前一方法迭代优化效率低的问题。但问题是，该方法实现了高效转换的同时，却丧失了多种风格转换的可扩展性。

本文在复现改进基于前馈网络图像风格转换方法的技术上，尝试以更为通用的形式实现多种混合样式风格转移，并能够智能化地组合多种风格。本文的具体目标如下，

1） 首先复现基于前馈网络的快速风格转换方法，使用最新提出的零值填充、上采样和正则化等技术改进其网络结构，提高风格转换质量，以此来学习和掌握图像领域的深度学习技术。并通过观察不同超参数对风格转换的影响，理解该方法的特定。

2） 其次在原方法的基础上，设计一个多种风格混合的快速风格转换方法，使得算法可以更全面地理解和学习多种类型风格的图像，并优化模型参数以平衡转换的速度与风格的准确性。

2 相关工作

图像风格迁移问题源于图片渲染[1],以及与之相关的纹理合成和转换[2]。早期的一些方法包括基于线性滤波器的直方图匹配[3]，非参数采样[4]等。研究工作[15,16]试图通过全局扫描样本，生成在更大图像上看上去自然的纹理。这些方法行之有效，但依赖于低阶的图像统计信息而无法捕获到高维语义结构，且只适用于同质纹理。直到 Gatys 等人[17]提出使用神经网络算法的方法，从预训练网络中纹理提取特征并使用特征图（feature map）的 Gram 矩阵，将噪声转换为纹理。特征图的 Gram 矩阵包含有关纹理的信息，通过减少输

入高斯噪声与神经网络各层的纹理 Gram 矩阵之间的距离，最终可以实现将噪声转化成目标纹理样式。

除此之外，针对神经网络的特征重建的研究工作也为后续风格转移技术奠定了基础。Yosinksi 等[18]和 Nguyen 等[19]使用梯度上升来可视化不同层的神经网络对于不同目标类值的理解。Google 的 Deep Dream 项目[20]使用此种方法来最大化输入图片的某一特定特征。Mhendran 等人[21]使用输入图像的特征向量来产生另一张具有相似特征的图像，结果发现，处理后的输出图像在神经网络底层与原图非常相似。

基于以上发现与见解，Gatys 等人[5,6]在 15 年首次提出通过匹配深度神经网络（DNN）中卷积层的特征统计信息的方法，并取得了令人印象深刻的风格转移结果。Gatys 等人的框架[5,6]基于一个缓慢优化的过程，该过程将初始为噪声的图像、内容图像和风格图像作为预训练的损失函数计算网络（通常是 VGG）的输入。与特征重构和纹理合成类似，图像不同维度的特征分别从 VGG 网络的某些卷积层提取出来，以计算基于内容的损失值和基于风格的损失值。将损失值计算结果反向转播叠加到噪声图像上，迭代更新图像梯度来最小化这些损失。在 Gatys 等人的研究基础上，又有不少进一步的研究工作。Li 等人[7]在深度特征空间中引入基于马尔可夫随机场（MRF）的框架来强制局部模式的学习。Gatys 等人[8]提出控制颜色保留、空间位置和风格转移规模的方法。

尽管效果显著，上述这种基于优化的方法收敛相对缓慢。一个常见的解决方法是在增加一个前馈神经网络[9,10]，把学习到的信息保留在网络参数中，以训练时间换取泛化时间。这种基于前馈神经网络的方法比前种方案快上三个数量级。之后 Wang 等人[11]利用多分辨率架构增强了前馈式传输的粒度。Ulyanov 等人[12,22]发现在图像转换网络中使用 Instance Normalization 替换 Batch Normalization 产生了更好的结构，提出新的改善生成样品质量和多样性的网络结构。

这种前馈网络实现了实时样式传输，但仅限于每个网络转换固定一种风格样式。之后 Dumoulin 等人[13]发现图像转换网络中 Instance Normalization 的仿射参数 γ 和 β 是与风格样式有着显著关系，并提出 Conditional Instance Normalization 结构，其中的 γ 和 β 分别对应着风格矩阵的一行，每种风格对应一种 γ 和 β 组合，该实验结果表明前馈网络能够编码 32 种不同的风格样式及其插值。Yijun Li 等人[14]提出一种新的前馈结构，可以合成多达 300 种不同的纹理样式，并转换 16 种图像风格。

上述方法不但可以生成实时风格转换效果，还允许单个网络产生多个样式化，但需在训练阶段指定固定的样式数目，训练后的模型无法泛化任意风格。针对这一问题，Huang 等人[23]提出了用固定的 Adaptive Instance Normalization（AdaIN）函数来计算任意风格样式的 Instance Normalization 参数。然而，与神经方法相比，固定函数的方法显然不够灵活。另一种方法[24]将规范化参数从一个 Descriptive Network 单独抽取出来，再添加到前馈转换网络。这两种方法都存在的一个缺陷是，除前馈网络外，泛化过程需要额外网络（分别是 VGG 和 Descriptive Network）的介入，增加了空间负载。

本文首先尝试理解、复现、改进 Johnson 等人的研究工作[9]，并在其基础上，设计一种参数学习方式，让前馈网络自适应地学习不同风格损失的权重参数，以同时表征多种混合风格样式。

3 方法

3.1 基于优化的图像风格转换

图像风格转换任务可以被理解为，构造出一张新图像p，该图像描绘的内容与图像c一致，同时该图像的风格又与图像s相似。根据最初的工作[5]，该理解中的图像内容与图像风格对于神经网络算法可以被表达为：

1）在预训练的 VGG 网络中，如果两张图片被提取出的高层特征的欧氏距离接近，则这两张图片内容相似；

2）如果两张图片被提取出的低层特征，其 Gram 矩阵之间的欧氏距离接近，那么这两张图片的风格近似。

在 Gatys 等人提出的模型中（图 3-1），神经网络的风格转换算法将图像p初始化为随机高斯噪声，迭代优化该图像以最小化损失函数$\mathcal{L}(s,c,p)$，

$$\mathcal{L}(s,c,p) = \lambda_s \mathcal{L}_s(p) + \lambda_c \mathcal{L}_c(p)$$

其中，$\mathcal{L}_s(p)$表示基于图像风格的损失，$\mathcal{L}_c(p)$表示基于图像内容的损失，λ_s和λ_c分别对应这两种损失的权重。令$\phi_l(x)$表示损失函数网络（VGG）对于输入x在第l层上的特征激活，那么当第l层是卷积层时，$\phi_l(x)$就是该卷积层输出的大小为$C_l \times H_l \times W_l$的特征图。该层内容损失表示为该层输出特征图上的特征激活值的均方误差：

$$\mathcal{L}_c^l = \frac{1}{C_l H_l W_l} \|\phi(p) - \phi(c)\|_2^2$$

对于第l层上的风格损失$\mathcal{L}_c^l(p)$，需要首先定义一个$C_l \times C_l$大小的 Gram 矩阵G^l，

$$G^l(x)_{c,c'} = \frac{1}{C_l H_l W_l} \sum_{h=1}^{H_l} \sum_{w=1}^{W_l} \phi_l(x)_{h,w,c} \phi_l(x)_{h,w,c'}$$

如果将$\phi_l(x)_{h,w,c}$理解为给定的$H_l \times W_l$特征图上第C_l维上的点，特征图上每一点都相互独立，那么$G^l(x)$就与第C_l维特征的非中心协方差矩阵（uncentered covariance）成正比，能表征关于那些特征被同时激活的信息。通过将$\phi_l(x)$形状重排列成$C_l \times H_l W_l$的二维矩阵ψ，Gram 矩阵可以被高效的计算得到，$G^l = \psi\psi^T / C_l H_l W_l$。基于 Gram 矩阵，$l$层上的风格损失就是输出图像和风格图像的 Gram 矩阵差异的平方范数：

$$\mathcal{L}_s^l = \|G^l(p) - G^l(s)\|_2^2$$

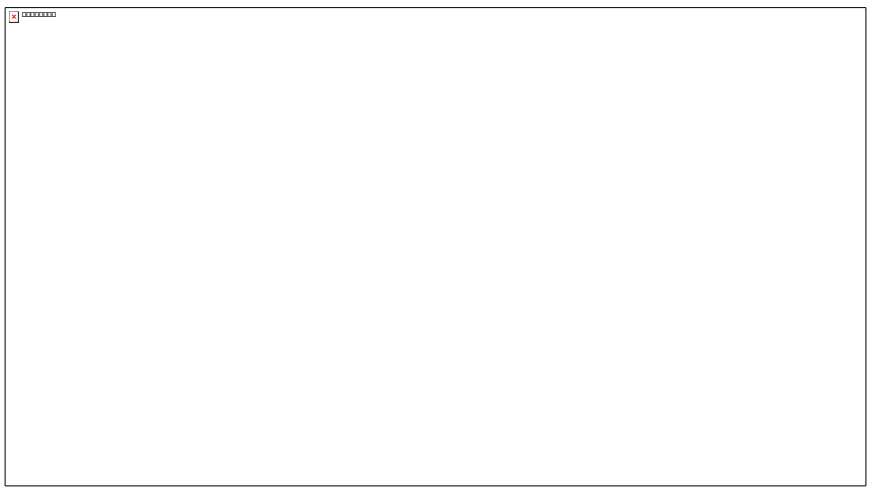

图. 3-1. 基于优化的图像风格转换

3.2 基于前馈网络的图像风格转换

为了加快上述基于优化的方法，前馈卷积网络（风格转移网络T）被添加在损失值计算网络之前，来学习风格转换参数[9,10]。该前馈网络将内容图像c作为输入并直接输出转换后的图像（图 3-2）。输入内容图像数据集，使用与上述相同的损失函数对网络进行，即

$$\mathcal{L}(s,c) = \lambda_s \mathcal{L}_s(T(c) + \lambda_c \mathcal{L}_c(T(c))$$

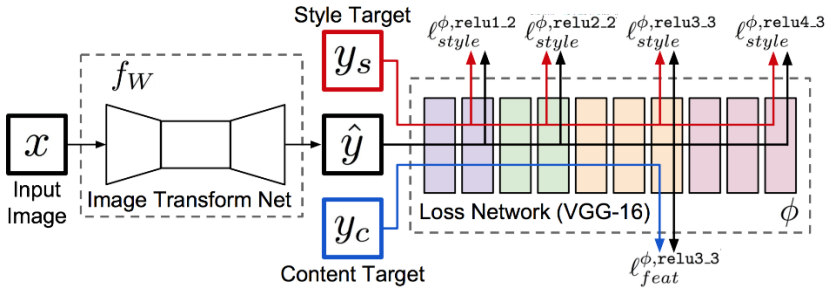

图. 3-2. 基于前馈网络的图像风格转换

虽然前馈式风格转换网络解决了泛化阶段的速度问题，但它们受到一个网络T只能转换一种特定图像风格的限制。这意味着必须针对每种要转换的风格样式训练单独的网络T.

3.1 基于前馈网络的混合风格转换

在前馈网络风格转换的基础上，对于每个网络，本文采用多种风格样式加以训练。每种输入的风格样式的风格损失，以加权平均的方式进行混合，计算得到总风格损失，

$$\mathcal{L}_s = \sum_{i=1}^{n} w_i \mathcal{L}_{si}$$

其中的混合权重w_i可以如下方式计算，

$$w_i = \frac{L_{s_i}}{\sum_{i=1}^{n} L_{s_i}}$$

本文采用与 Johnson 等人[9]相同的网络架构，并对其做如下的改进[22,26]：

1） 零值填充，用 reflection padding 替换 zero padding，避免在 SAME-padded 卷积层中由 zero-padding 引起的边界图案紊乱的问题。

2） 上采样，nearest-nerghbor upsampling 替换的 transposed upsampling，避免出现棋盘块式的风格图案。

3） 正则化，instance normalization 替换 batch normalization,显式地让网络中卷积层关注每一个输入样本的特征，而不是整一个批次地样本特征。

通过上述三项改进，风格转换网络可以不需要原先 Johnson 等人提出用于去除高频噪声的 total variation loss。

4 实验结果与分析

4.1 实验设置

本实验采用 pytorch 深度学习框架实现神经网络模型，以 Microsoft COCO(2014)数据集为训练集。该数据集包含约 8 万张图片，每张图片被放缩到 256×256 大小，以单批次 4 张图片的数量随机输入到风格转换网络。优化器选用 Adam 算法，学习率设置为1×10^{-4}，模型训练一个回合。对于所有的风格转换实验，选用预训练的 VGG-16 作为损失值计算网络，relu2_2 层处的激活表示内容特征，relu1_2、relu2_2、relu3_3、relu4_3 等层上的激活表示风格特征。单个网络在 GeForce GTX 1080 训练耗时约在 1.5 小时左右。

4.2 单一风格转换

以风格图像 wave 为例，当学习设定为1×10^{-4}时，可以看到损失曲线（图 4-1）在第 20 个间隔左右（即约 4 万张训练图片）趋向于拟合。

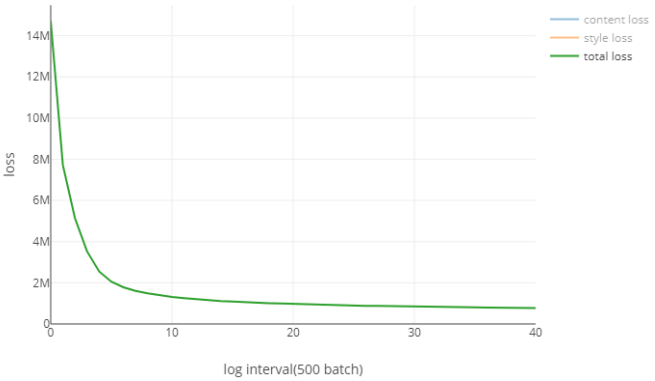

图. 4-1. Wave 风格的模型训练损失曲线

以图像 mosaic、seated-nude、shipwreck、starring-night 为输入的风格图像，训练得到 4 个风格转换模型。在泛化阶段，输入内容图像给转换模型，得到相应的风格转换图像。可以看到，输出的转换图像表达的内容与内容图像一致，同时在颜色、笔触和纹理等风格概念上，和模型的训练风格相似。

图. 4-2. 四种风格转换网络的泛化示例

固定内容损失权重$\lambda_c = 1e5$，调节风格损失权重$\lambda_s \in [1e10, 5e10, 1e11]$。从 4-3 损失曲线图可以看出，随着风格损失权重增大，内容损失值从原来的随训练逐渐较小，到不再变化甚至逐渐增加。这一现象可以理解为，训练目标针对的是总损失值，当风格函数的损失值权重远大于内容损失值权重，内容损失得不到足够训练，甚至在输入图像风格笔触趋向于风格图像时，二者内容之间的欧氏距离反而不断增大。

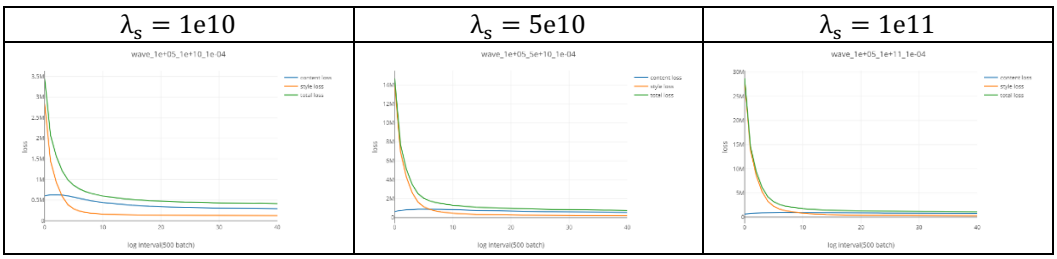

图. 4-3. 不同风格损失权重的损失曲线

如图 4-4，随着风格损失权重不断增加，输入内容图像与转换图像之间表达的内容不再一致，表征图像风格的笔触和纹理在风格转换图像上愈来愈明显（大），甚至甚于原风格图像（$\lambda_s = 1e11$）。

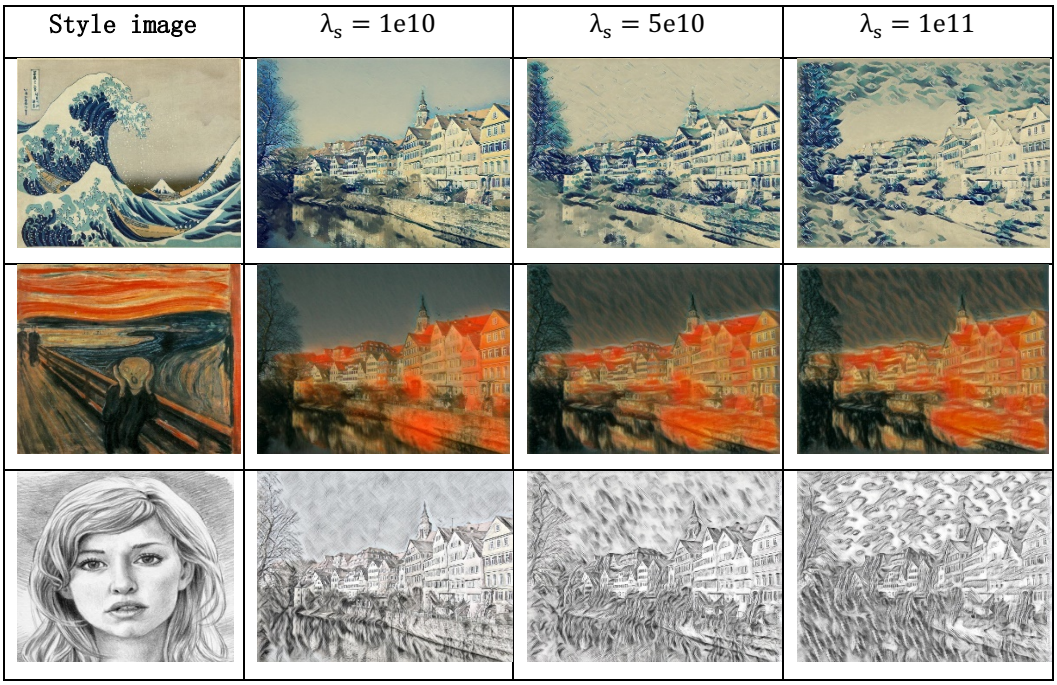

图. 4-4. 不同风格损失权重的风格转换网络的泛化示例

4.2 混合风格转换

对于混合风格转移，本实验修改 Johnson 等人[9]的前馈网络，以便能够同时学习多种混合样式风格，并且改进了其前馈网络的零值填充、上采用和正则化等技术细节。从混合风格转换的输出结果（图 4-5，4-6）可以看出，设计的风格混合方法能够较好的同时学习表征不同样式。

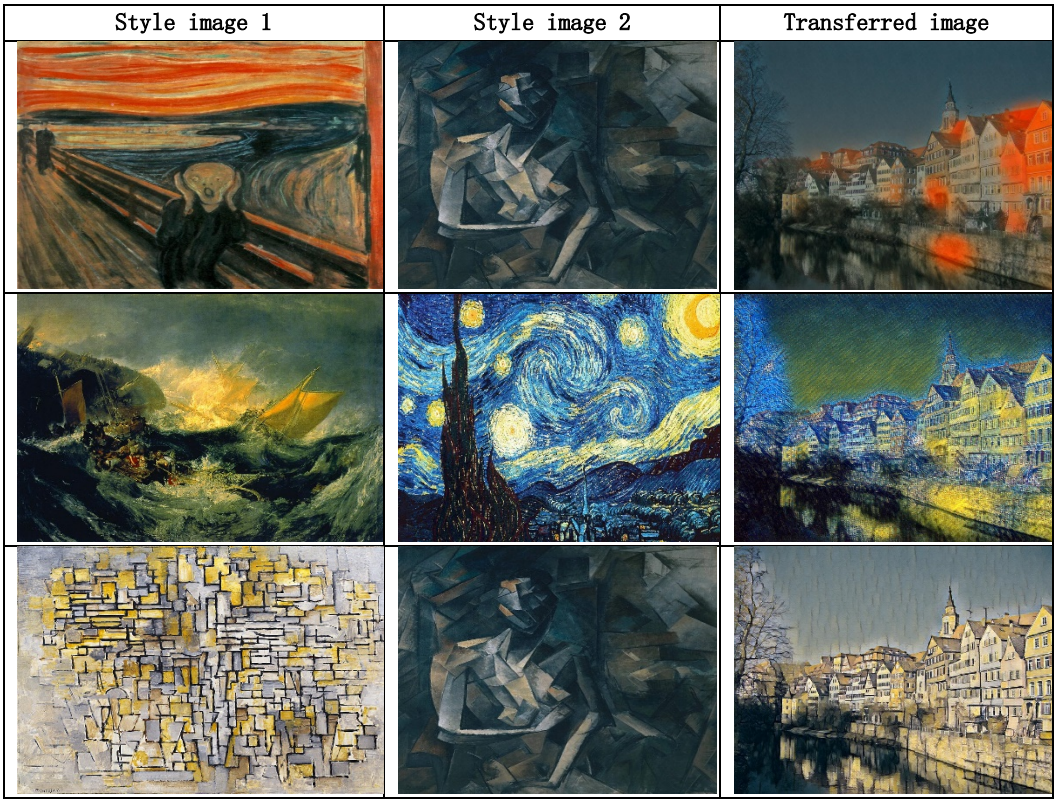

图. 4-5. 两种风格图像的混合学习泛化示例

需要提及的是，该方法使用可训练的混合权重，无需手动设定超参数来平衡不同风格，在训练过程中，模型自动选择最佳的权重组合。另外，虽然实验只实现和展示了两种及四种图像风格混合转换，但有理由相信，该方法可以有效扩展到更多图像风格。

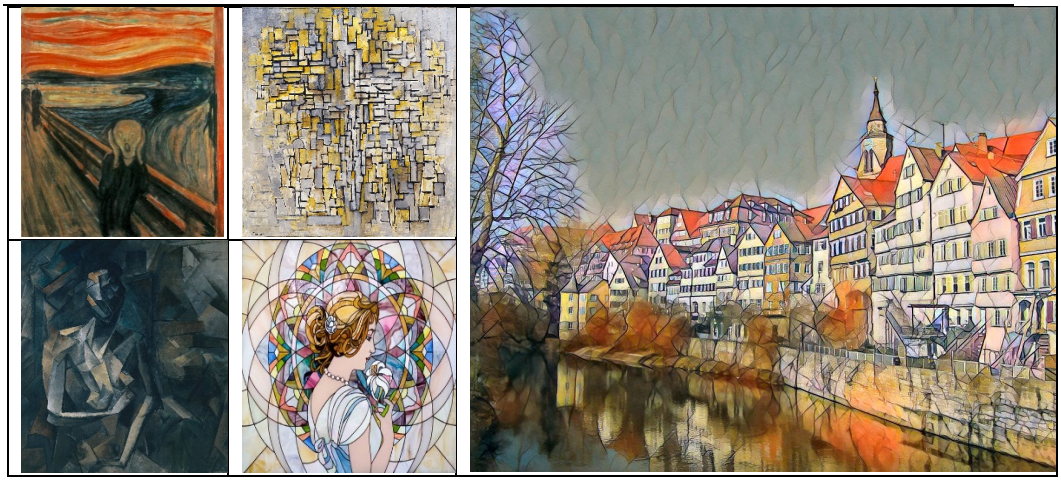

图. 4-6. 四种风格图像的混合学习泛化示例

5 结论

本文基于复现、改进和扩展基于前馈网络的快速图像风格转换方法，实现了快速的多种混合风格转换的方法。实验结果表明，该方法可以使神经网络模型理解学习表征多种类型的图像风格，高效快速地渲染输入内容图像。

虽然本文采用的方法足够通用并且可以同时表征多种图像风格，但仍限于事先设定的特定一种或多种风格图像，无法扩展到任意多种（在选用该方法之前，尝试过使用 AdaIN 结构设计一个新的快速表征任意图像风格的网络模型，但效果很不理想）。未来本文的方法可以被扩展以融入计算机视觉的其他技术，比如应用图像语义分割使风格转换网络能够识别并分别渲染图片的不同组成部分。

参考文献

[1] Kyprianidis, Jan Eric, et al. "State of the" Art": A Taxonomy of Artistic Stylization Techniques for Images and Video." IEEE transactions on visualization and computer graphics 19.5 (2013): 866-885.

[2] Elad, Michael, and Peyman Milanfar. "Style Transfer Via Texture Synthesis." IEEE Trans. Image Processing 26.5 (2017): 2338-2351.

[3] Heeger, David J., and James R. Bergen. "Pyramid-based texture analysis/synthesis." Proceedings of the 22nd annual conference on Computer graphics and interactive techniques. ACM, 1995.

[4] Frigo, Oriel, et al. "Split and match: Example-based adaptive patch sampling for unsupervised style transfer." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.

[5] Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge. "Image style transfer using convolutional neural networks." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.

[6] Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge. "A neural algorithm of artistic style." arXiv preprint arXiv:1508.06576 (2015).

[7] Li, Chuan, and Michael Wand. "Combining markov random fields and convolutional neural networks for image synthesis." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.

[8] Gatys, Leon A., et al. "Controlling perceptual factors in neural style transfer." IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017.

[9] Johnson, Justin, Alexandre Alahi, and Li Fei-Fei. "Perceptual losses for real-time style transfer and super-resolution." European Conference on Computer Vision. Springer, Cham, 2016.

[10] Ulyanov, Dmitry, et al. "Texture Networks: Feed-forward Synthesis of Textures and Stylized Images." ICML. 2016.

[11] Wang, Xin, et al. "Multimodal transfer: A hierarchical deep convolutional neural network for fast artistic style transfer." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Vol. 2. No. 6. 2017.

[12] Ulyanov, Dmitry, Andrea Vedaldi, and Victor S. Lempitsky. "Improved Texture Networks: Maximizing Quality and Diversity in Feed-forward Stylization and Texture Synthesis." CVPR. Vol. 1. No. 2. 2017.

[13] Dumoulin, Vincent, Jonathon Shlens, and Manjunath Kudlur. "A learned representation for artistic style." Proc. of ICLR (2017).

[14] Li, Yijun, et al. "Diversified texture synthesis with feed-forward networks." Proc. CVPR. 2017.

[15] Efros, Alexei A., and Thomas K. Leung. "Texture synthesis by non-parametric sampling." iccv. IEEE, 1999.

[16] Wei, Li-Yi, and Marc Levoy. "Fast texture synthesis using tree-structured vector quantization." Proceedings of the 27th annual conference on Computer graphics and interactive techniques. ACM Press/Addison-Wesley Publishing Co., 2000.

[17] Gatys, Leon, Alexander S. Ecker, and Matthias Bethge. "Texture synthesis using convolutional neural networks." Advances in Neural Information Processing Systems. 2015.

[18] Yosinski, Jason, et al. "Understanding neural networks through deep visualization." arXiv preprint arXiv:1506.06579 (2015).

[19] Nguyen, Anh, Jason Yosinski, and Jeff Clune. "Multifaceted feature visualization: Uncovering the different types of features learned by each neuron in deep neural networks." arXiv preprint arXiv:1602.03616 (2016).

[20] Mordvintsev, Alexander, Michael Tyka, and Christopher Olah. "Deepdream." (2015).

[21] Mahendran, Aravindh, and Andrea Vedaldi. "Understanding deep image representations by inverting them." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[22] Vedaldi, Victor Lempitsky Dmitry Ulyanov Andrea. "Instance Normalization: The Missing Ingredient for Fast Stylization." arXiv preprint arXiv:1607.08022 (2016).

[23] Huang, Xun, and Serge J. Belongie. "Arbitrary Style Transfer in Real-Time with Adaptive Instance Normalization." ICCV. 2017.

[24] Zhang, Hang, and Kristin Dana. "Multi-style generative network for real-time transfer." arXiv preprint arXiv:1703.06953 (2017).

[25] Chen, Dongdong, et al. "Stylebank: An explicit representation for neural image style transfer." Proc. CVPR. Vol. 1. No. 3. 2017.

[26] Odena, Augustus, Vincent Dumoulin, and Chris Olah. "Deconvolution and checkerboard artifacts." Distill 1.10 (2016): e3.