Postprint: Image Synthesis Method for Paintings Based on Generative Adversarial Networks

Authors: Zhao Yuxin, crown

Date: 2020-09-28T00:00:00+00:00

Abstract

Image synthesis for artistic paintings aims to fuse images from two distinct sources as foreground and background, respectively, which typically necessitates local style transfer. Existing algorithms involve cumbersome procedures and are time-consuming, failing to achieve real-time image synthesis. To address this shortcoming, we propose PainterGAN, a forward generative model based on Generative Adversarial Networks (GAN). PainterGAN' s self-attention mechanism and U-net architecture preserve the semantic content of the foreground during the synthesis process. Simultaneously, adversarial learning guarantees realistic style transfer. In experiments, employing a pretrained model as PainterGAN' s generator substantially reduces computational time and cost. Experimental results indicate that, compared to prior methods, PainterGAN produces images of comparable or superior quality while achieving a 400-fold increase in generation speed, representing a high-quality, high-efficiency solution to local style transfer problems.

Full Text

Preamble

Vol. 38 No. 3
Application Research of Computers
ChinaXiv Partner Journal

Painterly Image Composition Based on Generative Adversarial Networks

Zhao Yuxin, Wang Guan
(School of Mathematics, Tianjin University, Tianjin 300354, China)

Abstract: Painterly image compositing aims to harmonize a foreground image inserted into a background painting through local style transfer. The chief

drawback of existing methods is their high computational cost, which makes real-time operation difficult. To overcome this limitation, this paper proposes a feed-forward model based on generative adversarial networks (GAN), called PainterGAN. PainterGAN introduces a self-attention mechanism and U-net structure to preserve the semantic content of the foreground during synthesis. Meanwhile, adversarial learning guarantees faithful style transfer. In experiments, a pre-trained model is used as the generator for PainterGAN, dramatically reducing computational time and cost. Experimental results demonstrate that PainterGAN generates images of comparable or superior quality to existing methods while achieving a 400× speedup, making it a high-quality, high-efficiency solution for local style transfer problems.

Keywords: image style transfer; generative adversarial network; image compositing; self-attention mechanism

0 Introduction

Image compositing belongs to the category of image transformation problems, where the goal is to transform a simple copy-paste composite into a harmoniously integrated image. For instance, when inserting a portrait (foreground) into a photograph (background), image compositing seeks to blend them seamlessly such that observers believe the portrait originally belonged in the scene. Because foreground and background images differ in stylistic features such as lighting, brightness, and texture, simple copy-pasting creates unnatural visual artifacts that are easily detected as fake. Therefore, a blending process is required to transfer some of the background' s style to the foreground, creating a visually unified composition.

For photographic image compositing, various approaches have achieved blending by matching statistical features between foreground and background, such as histograms, mean and variance statistics [**?**], and covariance matrices [**?**]. For painterly image compositing, Luan et al. [**?**] proposed a local style transfer model based on PatchMatch and neural networks. This paper presents a novel approach to this problem.

A closely related concept is image style transfer. With the advancement of deep learning [**?**, **?**], Gatys et al. [**?**] introduced neural style transfer (NST), which uses deep neural networks to transfer the stylistic features of oil paintings onto images while preserving their original content. Considering the time-consuming iterative optimization process of NST, Johnson [**?**] and Ulyanov [**?**] designed fast feed-forward generative models that significantly improved image generation speed. Numerous subsequent works [**?**, **?**] have further advanced this field. However, these existing methods address global style transfer problems and are unsuitable for painterly image compositing. For example, when pasting a bouquet of flowers into Van Gogh' s *Starry Night*, an ideal result would render the

flowers with a style similar to other vegetation in the painting, rather than combining styles from all elements including the night sky, mountains, and figures.

Generative adversarial networks (GAN) [**?**], proposed in 2014, have demonstrated impressive performance across many image processing tasks. A GAN consists of a generator and a discriminator, where the generator attempts to produce images similar to real data while the discriminator tries to identify generated images, eventually reaching a Nash equilibrium. In this state, the generator can produce sufficiently realistic data. Conditional GAN (cGAN) [**?**] constructs generators and discriminators using convolutional neural networks for image-related problems. IcGAN [**?**] combines GANs with encoders to edit image attributes in feature space for controlled image generation. CycleGAN [**?**] uses bidirectional mapping GAN models for image-to-image translation tasks. Zhang et al. [**?**] inserted self-attention mechanisms into GANs, substantially improving image generation quality. Unlike iterative optimization approaches, these models dramatically enhance generation speed, though generated images often lack sufficient detail and exhibit weak correlations between pixel regions.

This paper proposes PainterGAN, a novel model for painterly image compositing based on GANs. Through adversarial training, the loss function drives PainterGAN to learn stylistic features from the target background—including lighting, color, and texture—while preserving the semantic content of training data as much as possible. After training, PainterGAN can render any foreground image into the target background style. When the rendered foreground is pasted into the background, it blends completely, making it impossible for viewers to detect the composite. PainterGAN thus accomplishes local style transfer from background to foreground.

A key challenge is the trade-off between preserving original content and achieving realistic style transfer. When foreground content is heavily weighted, the transferred style often becomes inconsistent with the background. Conversely, emphasizing style transfer leads to information loss in the original content. Single-stage optimization schemes like PatchMatch [**?**] struggle to balance both aspects simultaneously. Two-stage optimization approaches, such as DPH [**?**], progressively refine generated images through coarse-to-fine stages but incur excessive computational costs.

PainterGAN improves upon the basic GAN architecture by introducing self-attention mechanisms and U-net to control preservation of foreground semantic content while adversarial training ensures realistic, background-consistent style transfer. During training, PainterGAN uses a pre-trained VGG network as the encoder within the generator, dramatically saving computational space and time. Experiments demonstrate that the proposed model generates images of comparable or superior quality to existing methods while achieving a 400× speedup.

1 Methodology

The fundamental principle of GANs is to transform a specific data distribution into a target distribution through mapping. During training, the adversarial loss function drives optimization of the entire model' s parameters until reaching a local optimum. For painterly image compositing, PainterGAN' s generator maps the foreground into the distribution of background images, endowing it with background stylistic features. This section provides detailed descriptions of the self-attention mechanism, PainterGAN' s network architecture, and the model' s loss functions.

1.1 Principles of Self-Attention Mechanism

Self-attention mechanisms establish correlations between different pixel regions during image generation, promoting complete object contours. In convolutional operations, individual kernels typically provide small receptive fields (e.g., 3$\times 3 or 4\times$4). Consequently, early convolutional layers capture fine-grained image information. As depth increases, the receptive field grows, enabling the model to grasp semantic content, but deep feature maps lose considerable information, making it difficult to effectively propagate regional correlations to shallow layers. Due to these convolutional limitations, existing style transfer methods tend to generate objects with broken edges. Self-attention offers a viable solution.

Originally used for contextual semantic understanding in natural language processing, Zhang et al. [**?**] first introduced self-attention to GANs for image classification. It has since proven effective in other computer vision tasks. Theoretically, self-attention responds more strongly to image regions that are more salient to human perception, thereby enhancing object prominence.

The self-attention network is inserted in PainterGAN' s generator after downsampling and before upsampling. The basic principle [**?**] can be summarized as follows: The feature maps produced by the encoder are fed into three independent convolutional layers. Assuming the input is x, with coefficient matrices for the three convolutional layers being W_f, W_g, and W_h respectively, where W_f and W_g measure the correlation between region i and region j in the image for generating attention weights. The output is a weighted sum of values, and considering that the self-attention network may not have converged to a local optimum initially, a parameter γ adjusts the output to allow the network to gradually influence image generation.

1.2 PainterGAN Network Architecture

As shown in [Figure 1: see original paper], PainterGAN primarily consists of two components: a generator and a discriminator. The generator comprises an encoder and decoder with symmetric structures that perform downsampling and reconstruction of input images. To save computational space and time,

PainterGAN uses a pre-trained VGG-19 as the encoder. VGG possesses powerful feature extraction capabilities that capture both pixel-level information and semantic content.

Before the decoder, the self-attention network computes correlations between different regions in the feature maps. Additionally, U-net connections link features from the same hierarchical level between encoder and decoder. Guided by the encoder, the decoder reconstructs the image. The generator structure is illustrated in [Figure 2: see original paper].

The discriminator is another crucial component of PainterGAN. Standard GAN-based models feed real and generated data to the discriminator, which performs simple downsampling and outputs a real (1) or fake (0) judgment. During training, both data types are split into smaller pixel patches before being fed to the discriminator. This approach reduces trainable parameters and provides greater flexibility, enabling the discriminator to accept images of any size as input.

In summary, the discriminator supervises realistic style transfer, while the self-attention mechanism and U-net preserve original semantic content. These components cooperate to maintain balance between style and content during transfer.

1.3 Loss Functions

1.3.1 Adversarial Loss Function As previously mentioned, the adversarial loss function drives the generator and discriminator toward equilibrium, with their parameters being alternately optimized during training. The generator loss is defined as $\mathcal{L}_{\text{adv}}^{G}$, while the discriminator loss is $\mathcal{L}_{\text{adv}}^{D}$. Here x_i^f, x_i^b, and $G(x_i^f)$ represent sampled foreground images, background images, and generated images respectively. The loss value indicates the extent to which generated images possess the target style.

1.3.2 Content Loss Function Beyond achieving appropriate style, generated images should preserve their original semantic content. DTN [**?**] discovered that when an image x_i passes through the generator to obtain $G(x_i)$, the mapping $f(G(x_i))$ remains consistent, where f maps images to feature space. This phenomenon, called "f-constancy," reflects that appearance-transformed images retain high-level semantic content features. While feasible, experiments show that "f-constancy" imposes overly strict constraints that suppress style diversity to some extent.

This paper adopts a pixel-level content loss function to measure differences between input and generated images. To obtain clearer image details, the L1 norm is used:

$$\mathcal{L}_{\text{con}} = \frac{1}{N}\sum_{i=1}^{N} \|x_i - G(x_i)\|_1$$

1.3.3 TV Regularization Term To encourage local smoothness in generated images, PainterGAN employs a total variation (TV) regularization term:

$$\mathcal{L}_{\text{TV}} = \sum_{i,j} \sqrt{(x_{i,j} - x_{i+1,j})^2 + (x_{i,j} - x_{i,j+1})^2}$$

where $x_{i,j}$ denotes the pixel value at position (i, j). The overall PainterGAN loss function combines these components:

$$\mathcal{L}(G, D) = \omega_1 \mathcal{L}_{\text{adv}} + \omega_2 \mathcal{L}_{\text{con}} + \omega_3 \mathcal{L}_{\text{TV}}$$

where ω_1, ω_2, and ω_3 represent the weights for adversarial, content, and regularization terms respectively.

2 Experiments

2.1 Experimental Platform

Experiments were conducted on an Ubuntu 16.04 system with an NVIDIA GTX 1080 Ti GPU, using Python and the TensorFlow framework. A pre-trained VGG-19 served as the encoder in the generator, producing feature maps from layer "conv4_1" as input to the self-attention network. U-net connections linked symmetric downsampling and upsampling convolutional layers. The entire network was trained for 200 epochs with a batch size of 64. The optimizer was Adam with an initial learning rate of 0.0002 and momentum of 0.5.

To accelerate convergence of PainterGAN' s generator, the network was initialized as a reconstruction function. Training with only the content loss function for 10 epochs enabled the generator to produce images close to the input. This same approach was used in [**?**] to speed up model optimization.

2.2 Experimental Data and Processing

The training data comprised two parts: grayscale images and paintings of various styles. The former served as generator input, while the latter and generated images were fed to the discriminator. Test data included only grayscale images.

Foreground: 3,482 grayscale images were extracted from the film *Loving Vincent*, with 3,070 used for training and the remainder for testing. These cropped images contained plants, buildings, figures, and other content.

Background: Background images came from four animated films: *Loving Vincent* (2,959 frames), *Father and Daughter* (2,548 frames), *The House of Small*

Cubes (1,570 frames), and *The Man Who Planted Trees* (4,104 frames). These four datasets belong to different artistic schools with distinct styles. All training data was cropped to 256$×$256 pixels and augmented through flipping and rotation.

2.4 Quantitative Performance Comparison

For training time, we compared PainterGAN with and without a pre-trained encoder. Results show that using a pre-trained encoder reduced neurons by 33.82% and training time by 46.49% while achieving the same performance.

For image generation time, among the four style transfer methods, NST performs global style transfer while Deep Analogy has strict requirements on foreground and background. Only DPH and PainterGAN are suitable for local style transfer of arbitrary foregrounds. Therefore, we compared their compositing speeds, with results shown in .

** Comparison of generation time between DPH and PainterGAN**

The results demonstrate that PainterGAN generates images in real time, being 400× faster than DPH. From this perspective, PainterGAN effectively learns image styles and can efficiently blend any foreground into backgrounds of that style.

2.5 Hyperparameter Tuning of Loss Functions

As mentioned in Section 1.3.3, PainterGAN' s loss function is $\mathcal{L}(G, D) = \omega_1\mathcal{L}_{\text{adv}} + \omega_2\mathcal{L}_{\text{con}} + \omega_3\mathcal{L}_{\text{TV}}$, where hyperparameters ω_1, ω_2, and ω_3 represent the importance of adversarial, content, and regularization terms in driving training. Through extensive experimentation and tuning, the final values were set to 1, 70, and 50 respectively. We selected three groups of hyperparameters for comparison, with results shown in [Figure 8: see original paper].

[Figure 8: see original paper] shows test results with different hyperparameter settings using images from *Loving Vincent*. In (a), the input test image is shown. In (b), with weights set to 1:1:1, some original content is lost—petals in the upper image become distorted, and patterns on the vase neck in the lower image disappear. Therefore, in (c), we increased the content loss weight to 1:50:1, but this caused discontinuities in object edges (e.g., flower contours in the lower image). In (d), we correspondingly increased the TV term weight to 1:70:50, which suppressed distortion during generation while preserving complete semantic content, yielding the best rendering quality.

3 Conclusion

PainterGAN revisits the local style transfer problem in image compositing through adversarial training and an image-to-image feed-forward generation approach. It introduces self-attention mechanisms and U-net into GANs to

improve generation quality and further explores using a pre-trained VGG as the generator' s encoder, saving training time and memory while maintaining image quality. Experiments show that compared to existing models, PainterGAN achieves comparable or superior style transfer quality while dramatically improving generation speed, enabling real-time image compositing. However, applying this model to video local style transfer remains challenging and represents a promising direction for future research.

References

[1] Reinhard E, Ashikhmin M, Gooch B, et al. Color Transfer between Images. *IEEE Computer Graphics and Applications*, 2001, 21(5): 34-41.

[2] Li Yijun, Liu Mingyu, Li Xueting, et al. A Closed-Form Solution to Photorealistic Image Stylization. *Proc of the European Conference on Computer Vision*. Cham: Springer, 2018: 453-468.

[3] Luan Fujun, Paris S, Shechtman E, et al. Deep Photo Style Transfer. *Proc of IEEE Conference on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE Press, 2017: 6997-7005.

[4] Liu Jianwei, Liu Yuan, Luo Xionglin. Research and Development on Deep Learning. *Application Research of Computers*, 2014, 31(7): 1921-1930.

[5] Mao Yonghua, Gui Xiaolin, Li Qian, et al. Study on Application Technology of Deep Learning. *Application Research of Computer*, 2016, 33(11): 3201-3205.

[6] Gatys L A, Ecker A S, Bethge M. A Neural Algorithm of Artistic Style. arXiv preprint arXiv:1508.06576, 2015.

[7] Johnson J, Alahi A, Li Fei-Fei. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. *Proc of European Conference on Computer Vision*. Cham: Springer, 2016: 694-711.

[8] Ulyanov D, Lebedev V, Vedaldi A, et al. Texture Networks: Feed-Forward Synthesis of Textures and Stylized Images. *Proc of the 33rd International Conference on Machine Learning*. New York: ACM Press, 2016: 1349-1357.

[9] Ioffe S, Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *Proc of International Conference on Machine Learning*. New York: ACM Press, 2015: 448-456.

[10] Li Chuan, Wand M. Combining Markov Random Fields and Convolutional Neural Networks for Image Synthesis. *Proc of IEEE Conference on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE Press, 2016: 2479-2486.

[11] Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative Adversarial Nets. *International Conference on Neural Information Processing Systems*. Boston: MIT Press, 2014: 2672-2680.

[12] Mirza M, Osindero S. Conditional Generative Adversarial Nets. *Computer Science*, 2014, 5(32): 2672-2680.

[13] Isola P, Zhu Junyan, Zhou Tinghui, et al. Image-to-Image Translation with Conditional Adversarial Networks. *Proc of IEEE Conference on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE Press, 2016: 5967-5976.

[14] Zhu Junyan, Park T, Isola P, et al. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. *Proc of IEEE Conference on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE Press, 2017: 2242-2251.

[15] Connelly B, Eli S, Adam F, et al. PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing. *ACM Transactions on Graphics*, 2009, 28(3, article 24).

[16] Zhang Han, Goodfellow I, Metaxas D, et al. Self-Attention Generative Adversarial Networks. arXiv preprint arXiv:1805.08318, 2018.

[17] Taigman Y, Polyak A, Wolf L. Unsupervised Cross-Domain Image Generation. arXiv preprint arXiv:1611.02200, 2016.

[18] Chen Yang, Lai Yu-Kun, Liu Yong-Jin. CartoonGAN: Generative Adversarial Networks for Photo Cartoonization. *Proc of the IEEE Conference on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE Press, 2018: 9465-9474.

[19] Liao Jing, Yao Yuan, Yuan Lu, et al. Visual Attribute Transfer through Deep Image Analogy. *ACM Transactions on Graphics*, 2017, 36(4): 1-15.

Note: Figure translations are in progress. See original paper for figures.