Style Transfer with Content Preservation from Multiple Images

Dilin Liu(✉), Wei Yu, and Hongxun Yao

Harbin Institute of Technology, Harbin, China
{liudilin,h.yao}@hit.edu.cn, yuwei.hit@outlook.com

Abstract. Artistic style transfer is an image synthesis problem where the style of input image is reproduced with the style of given examples. Recent works show that artistic style transfer can be achieved by using hidden activations of a pretrained model. However, most existing methods only allow one example image representing style. In this work, we propose a framework based on neural patches matching that combines the content structure and style textures in a fusion layer of the network. Our method is capable to extract the style from a group of images, such as the paintings of specific painter. In particular, our method can preserve the original content information. Furthermore, by using multiple style images our approach can obtain desirable synthesis results in foreground objects.

Keywords: Texture synthesis · Style transfer
Convolutional neural network

1 Introduction

Famous artists are typically renowned for a particular artistic style and create literatures with their certain style. This motivates us to explore efficient computational strategies to create artistic images by examples. Many approaches focus on the problem of transferring the desired style from single painting to other images with similar content, mostly photographs. These are known as artistic style transfer in computer graphics and vision.

The key challenges of artistic style transfer problem are to capture the structure and color information of complex paintings. There have been many efforts to develop efficient methods for automatic style transfer. Many classic data-driven approaches [1–3] to the task are based on Markov Random Field that models characterize from images by statistics of local patches of pixels. Recently, deep neural networks models have shown exciting new perspective for image synthesis [4,5]. Subsequent work has improved both speed [6] and quality [7]. However, we found that most existing deep learning based approaches are designed to synthesis texture from only one single image. Their approaches only extract feature from one style image and the models are lack of ability to learn the common style of multiple paintings.

B. Zeng et al. (Eds.): PCM 2017, LNCS 10735, pp. 783–791, 2018.
https://doi.org/10.1007/978-3-319-77380-3_75

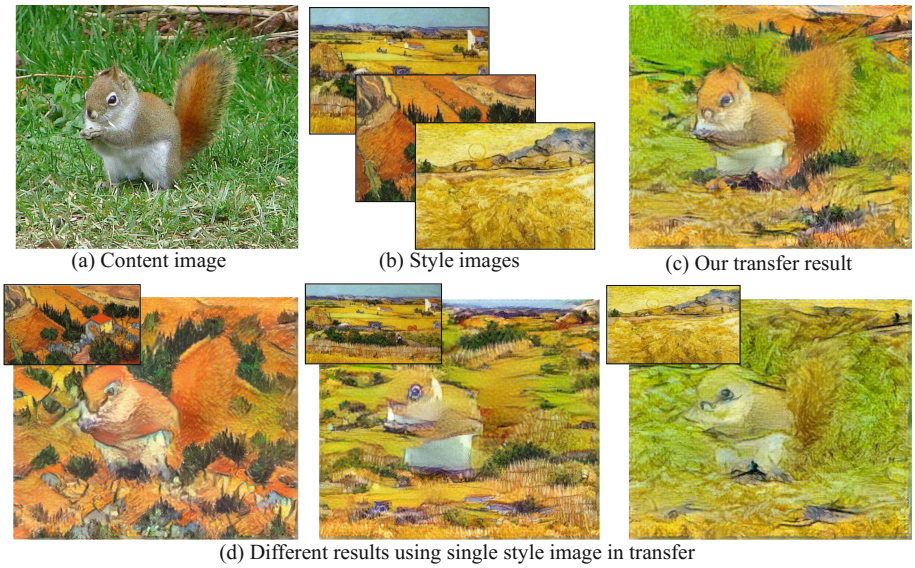

Fig. 1. Overview of our synthesis result using multiple example images. (a) Is the input content image. (b) Is three style images from *Vincent Willem van Gogh*. (c) Our result are generated with multiple style inputs where edges of the squirrel is well preserved. (d) Are the synthesis results using corresponding style image from content image.

In this paper, we propose an architecture of discriminative network which can extract the unified style from multiple images. Figure 1 shows the comparison of different synthesis results. Our idea is to evaluate the similarity between fused neural patch features sampled from the synthesis image and from different example images. The features are extracted from different networks with combination of both style and content information.

2 Related Work

Style transfer is an active topic in both academia and industry. Traditional methods mainly focus on generating new images by resampling either pixels [2,8] or whole patches [9,10] of the source texture images. Different methods were proposed to improve the quality of the synthesis image.

More recently, neural style transfer [4] has demonstrated impressive results in example-based image stylisation. The method is based on deconvolution on a parametric texture model [11,12] by summarising global covariance statistics of feature vectors on higher network layers. Most prominently, during the stylisation it displays a greater flexibility than traditional models to reconstructs the content that are not present in the source images. However, the representation of image style within the parametric neural texture model allows far less intuitive control and extension than patch-based methods. Our model works on calculating the

similarities between neural patches to control the transformation in the guidance of content information. On the theory side, Xie [13] have proved that a generative random field model can be derived from the discriminative networks, and show some applications to unguided texture synthesis.

In very recent, there are a serious of approaches that employ specially trained auto-encoders as generative networks. Generative Adversarial Networks(GANs) use two different networks, one as the discriminator and the other as the generator, to iteratively improve the model by playing a minimax game [14]. The adversarial network models can offer perceptual metrics [15,16] that allows auto-encoders to be training more efficiently. Previous works have trained GANs on kinds of datasets, including discrete labels, text and images. Additionally, Li use GANs to obtain real-time texture synthesis in one-image style transfer [17]. As far as we know, we are the first to generate synthesis image from multiple style examples and learn discriminative features for a specific style.

3 Our Approach

We first introduce our model to perform style transfer on multiple example images. Our goal is to control the stylization to preserve the sematic information in the content image. The main component of our proposed method is a content guidance channel calculated from the content network and serving to texture neural patches matching.

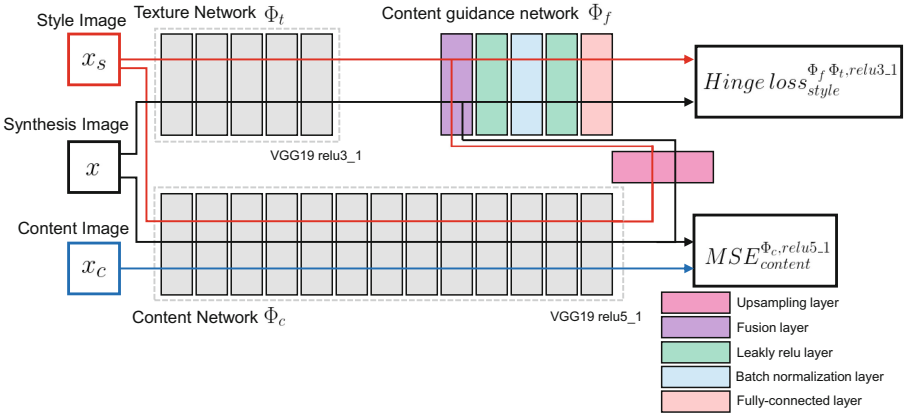

Fig. 2. Overview of our style transfer model. Our model contains a texture network, a content network, and the content guidance network. The content guidance network consists of fusion layer, batch normalization layer, leaky relu layers, and fully-connected layer.

The pipeline of our model is visualized in Fig. 2. The discriminative image synthesizer contains three parts: the texture network and the content network

work on different feature patches extraction and the content guidance network extracts the content guidance feature to the style patches.

The texture network is trained to distinguish between neural patches sampled from the synthesis image and sampled from the example images. It outputs a classification score for each neural patch, indicating the similarity of the patch matched with training data. For each patch sampled from the synthesized image, its similarity to patches from example images is maximized. The fusion layer consists of the output on layer relu3_1 of VGG_19 and the output of our proposed content guidance channel.

The content guidance network is connected to layer relu5_1 of VGG_19 and upsampling by using the nearest neighbour technique so that the output is the same size of the neural patches in texture network. The convolutional and upsampling layers ensure that features corresponding to neural patches contain the content information from a larger region than the representation of texture neural patches. Thus the output of the fusion layer for texture loss calculation is written as:

$$\Phi(x_f) = \sigma\left(\mathbf{b} + W\begin{bmatrix}\Phi(x)\\ \Phi(x_t)\end{bmatrix}\right) \tag{1}$$

where $\Phi(x_f)$ is the fused feature, $\Phi(x)$ represent the neural patches of the iterative image and $\Phi(x_t)$ is the corresponding content guided feature from layer relu5_1 of VGG_19. $\sigma(\cdot)$ is the Sigmoid non-linear transfer function and W is the weight matrix of the fusion layer, b is a bias. Here both W and b are learnable part of the network. Our style transfer approach concatenates the content and style information into the fused feature vector. The deconvolution progress back-propagates the loss to pixels in the synthesis image until the discriminative network recognize the patches as samples from the example images with similar content. We use batch normalization layer (BN) and leaky ReLU (LReLU) to decline the overfitting in the iteration progress as Radford [18] did.

The content network calculate the Mean Squared Error (MSE) between the synthesis image and the content image from the output on the abstract layer Relu5_1 as the content loss function.

Formally, we denote the example texture image by x_t, and the synthesized image by x. We initialize x with random noise and iterate the image in the guidance from the content image x_c. The deconvolution progress iteratively updates x until the following energy is minimized:

$$x^* = \arg\min_x E_t(x, x_t) + \alpha_1 E_c(x, x_c) + \alpha_2 \gamma(x) \tag{2}$$

E_t denotes the texture loss, in which x_f is the output of fusion feature layer. We sample nerual patches from the synthesized image x, and compute E_t using the Hinge loss with their guidance features fixed to the most similar patch in content:

$$E_t(x, x_t) = \frac{1}{N}\sum_{i=1}^{N} \max(0, 1 - s(\Phi_i(x_f))) \tag{3}$$

where $s(\Phi_i(x_f))$ is the classification score of i-th fusion feature that consists of the texture neural patch and the content guidance feature. N is the total number of sampled patches.

The content loss is the Mean Squared Error between the two feature maps x and x_c. By minimizing an additional content loss E_c, the network can generate an image that is contextually related to a guidance image x_c.

To perform the training of proposed discriminative network, the parameters are randomly initialized and then updated after each iteration. The synthesis image will go through both three parts in each iteration then the networks will back-propagate the loss to pixels. We set the weights with $\alpha_1 = 1$ and $\alpha_2 = 0.0001$, and minimize Eq. 2 using back-propagation. The additional regularizer $\gamma(x)$ is a smoothness prior for the synthesis image [19].

4 Experimental Results and Analyses

In this section, we analyze the improvement of proposed content preserving style transfer model. We use the Torch7 framework [20] to implement our model and use existing open source implementations of prior works [1,4,17] for comparison. We choose two groups of *Vincent Willem van Gogh*'s painting in different theme ("night star" and "yellow field") as example images. For the transferred images producing we iterated the synthesis image from the content image as initialization.

First experiment is the comparison of transferring the style of Vincent's painting onto a photo with building content. We use each image in the group of example images as input to get the baseline synthesis image. For comparison we choose Vincent's another painting with building to simulate the condition of transferring from the style image with same content information. Figure 3 shows different results between using one example image and using multiple example images in our method. In the synthesis image of column (a), ceiling lights are almost unrecognized. In the results of column (b)(c), the shape of the building is messed up with the background. By using multiple example images, our style transfer result in column (d) obtain advantages in generating the edges of building where the ceiling lights is clear and the edges of the building is preserved as the original image. With the comparison to the synthesis image of column (e), our method can preserve the building content in style transfer. In Fig. 4 we show more qualitative examples using different groups of style images.

Secondly, in Fig. 5 we compare our synthesis result with existing neural style transfer methods. Since most approaches focus on using only one example images, we choose a group of Vincent's painting in the same theme as example images. The synthesis result from Li [17] is failed to preserve detail of the house content. Gatys's method [4] transfers some houses to black cypress in the style image and colorize the houses with cold color tone. More obviously, the synthesis image from Elad [1] is divided into two kinds of style where houses almost stay in the style of original image with only colorization. In our result, the houses stay the warm color tone as content image and obtain the style of given oil paintings. Our method choose to synthesis sky regions with uniform texture instead

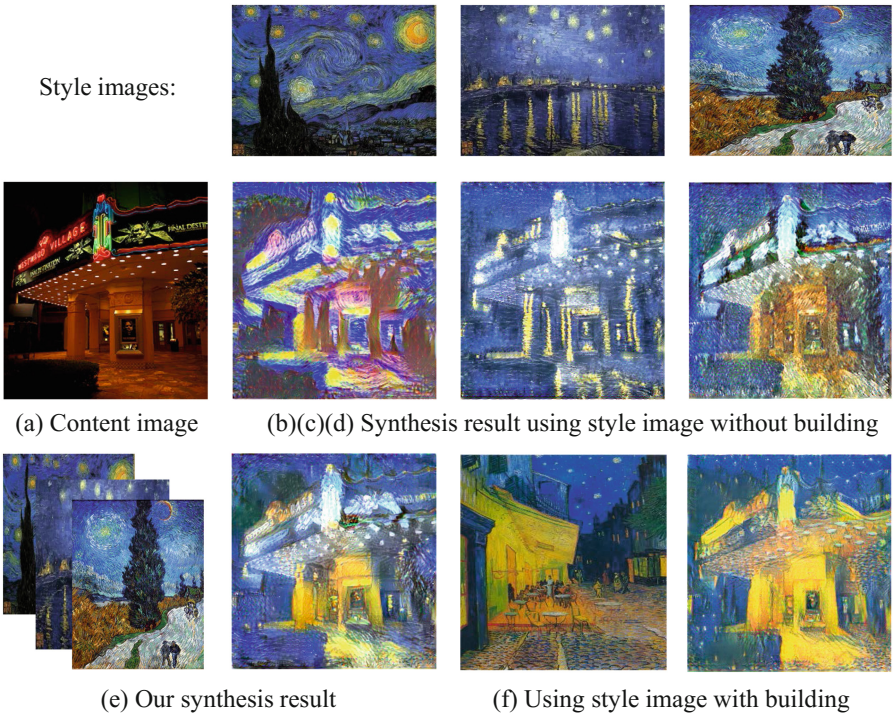

Fig. 3. The comparison of different synthesis strategies using corresponding style images. (b)(c)(d) Are the synthesis results using style images without building from (a) content image. (e) Our synthesis image is comparable to the synthesis image (f) using the style image with building content.

Fig. 4. More qualitative synthesis results using different groups of style inputs.

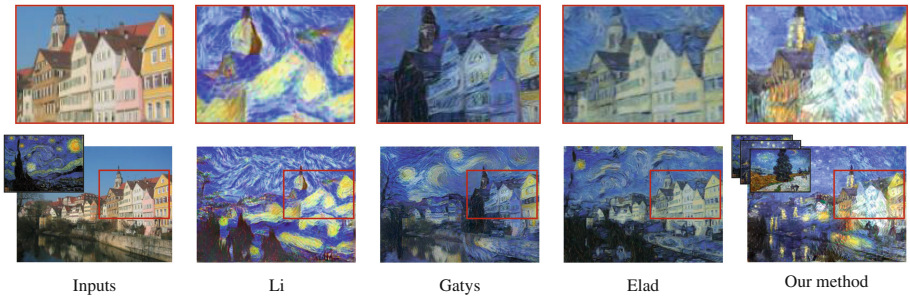

Fig. 5. The comparison of different synthesis strategies using our style transfer method and existing neural network based methods. The results in each column of images are given by (left-to-right): Li [17], Gatys [4], Elad [1] and our method. (Color figure online)

of stars because sky regions in original image match uniform style patches rather than patches sampled from stars. More comparisons of style transfer results are shown in Fig. 6.

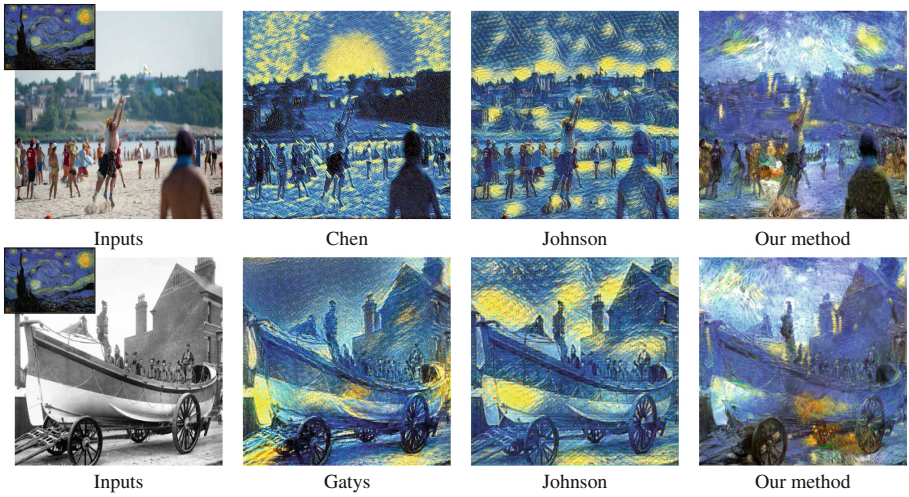

Fig. 6. More comparisons with style transfer methods: Chen [21], John [6] and Gatys [4] (Color figure online).

To explore deficiency in our synthesis method, we choose a photo of street cafe for content image because it is similar with Vincent's painting (*Night coffee shop*). We use three painting from Vincent in the same theme of "night star" as style input. The synthesis results are shown in Fig. 7. Our method can transfer the eave and tables to the one in real painting. The colorization of the coffee shop is not real enough. We considerate that cold color patches from style images occupy the majority and the network is in high probability to iterate the synthesis image with cold color tone building.

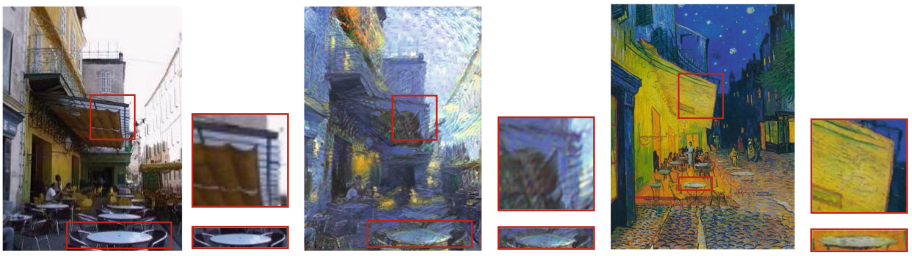

Fig. 7. The result of matching the content representation through the network. Texture of the painting and content of the photograph are merged together such as the real painting. (Color figure online)

5 Conclusion

We present a new CNN-based method of artistic style transfer that focus on preserving the content information and adaptability to arbitrary content. Our model concatenates both content and style information into a fusion layer by upsampling features from the content network. It is capable to transfer the style from a group of example images with preserving complex content. Our method is only one step in the direction of learning the relationship between style and content from images. An important avenue for future work would be to study the semantic representation of the style in some certain theme.

Acknowledgement. This work was supported by the National Natural Science Foundation of China under Project No. 61472103.

References

1. Elad, M., Milanfar, P.: Style-transfer via texture-synthesis, arXiv preprint arXiv:1609.03057
2. Efros, A.A., Leung, T.K.: Texture synthesis by non-parametric sampling. In: The Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 2, pp. 1033–1038. IEEE (1999)
3. Paget, R., Longstaff, I.D.: Texture synthesis via a noncausal nonparametric multiscale markov random field. IEEE Trans. Image Process. **7**(6), 925–931 (1998)
4. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2414–2423 (2016)
5. Li, C., Wand, M.: Combining Markov random fields and convolutional neural networks for image synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2479–2486 (2016)
6. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016 Part II. LNCS, vol. 9906, pp. 694–711. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_43

7. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Improved texture networks: maximizing quality and diversity in feed-forward stylization and texture synthesis, arXiv preprint arXiv:1701.02096
8. Wei, L.-Y., Levoy, M.: Fast texture synthesis using tree-structured vector quantization. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, pp. 479–488. ACM Press/Addison-Wesley Publishing Co. (2000)
9. Efros, A.A., Freeman, W.T.: Image quilting for texture synthesis and transfer. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, pp. 341–346. ACM (2001)
10. Kwatra, V., Schödl, A., Essa, I., Turk, G., Bobick, A.: Graphcut textures: image and video synthesis using graph cuts. In: ACM Transactions on Graphics (ToG), vol. 22, pp. 277–286. ACM (2003)
11. Heeger, D.J., Bergen, J.R.: Pyramid-based texture analysis/synthesis. In: Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques, pp. 229–238. ACM (1995)
12. Portilla, J., Simoncelli, E.P.: A parametric texture model based on joint statistics of complex wavelet coefficients. Int. J. Comput. Vis. **40**(1), 49–70 (2000)
13. Xie, J., Lu, Y., Zhu, S.-C., Wu, Y.N.: A theory of generative convnet, arXiv preprint arXiv:1602.03264
14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems, pp. 2672–26800 (2014)
15. Dosovitskiy, A., Brox, T.: Generating images with perceptual similarity metrics based on deep networks. In: Advances in Neural Information Processing Systems, pp. 658–666 (2016)
16. Larsen, A.B.L., Sønderby, S.K., Larochelle, H., Winther, O.: Autoencoding beyond pixels using a learned similarity metric, arXiv preprint arXiv:1512.09300
17. Li, C., Wand, M.: Precomputed real-time texture synthesis with markovian generative adversarial networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016 Part III. LNCS, vol. 9907, pp. 702–716. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_43
18. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks, arXiv preprint arXiv:1511.06434
19. Mahendran, A. Vedaldi, A.: Understanding deep image representations by inverting them. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5188–5196 (2015)
20. Collobert, R., Kavukcuoglu, K., Farabet, C.: Torch7: a Matlab-like environment for machine learning. In: BigLearn, NIPS Workshop, no. EPFL-CONF-192376 (2011)
21. Chen, D., Yuan, L., Liao, J., Yu, N., Hua, G.: StyleBank: an explicit representation for neural image style transfer, arXiv preprint arXiv:1703.09210