Theme Article

Automatic Color Sketch Generation Using Deep Style Transfer

Wei Zhang
Baidu, Inc.

Guanbin Li
Sun Yat-Sen University

Haoyu Ma and Yizhou Yu
University of Hong Kong

***Abstract*—Recent advances in deep learning based algorithms have made it feasible to transfer image styles from an example image to other images. However, it is still hard to transfer the style of color sketches due to their unique texture statistics. In this paper, an automatic color sketch generation system is developed from existing real-time style transfer methods. We choose a suitable image from a set of carefully selected color sketch examples as the style target for every content image during training. We also propose a novel style transfer convolutional neural network with spatial refinement to realize high-resolution style transfer. Finally, gouache color is introduced to the generated images via a linear color transform followed by a guided filtering operation. Experimental results illustrate that our system can produce vivid color sketch images and greatly reduce artifacts compared to previous state-of-the-art methods.**

Characterized by line strokes and regional gouache paints, color sketching is one of the most popular artistic forms to depict visual scenes. As shown in Figure 1, skilled painters can create visually pleasant color sketches by combining their perceptual abstraction of natural scenes with painting styles. In computer graphics, generating color sketches from photographs also attracts a lot of research in the field of nonphotorealistic rendering (NPR). State-of-the-art research provides interactive systems to assist users in creating vivid color sketch painting works.[1,2] Recently, customers enjoy recording beautiful photographs and sharing them on social networks after using a variety of image editing software. Among them, automatic style transfer software, which provides easy-to-use artistic augmentations, are becoming much more frequently used than ever before.

Digital Object Identifier 10.1109/MCG.2019.2899089
Date of publication 12 February 2019; date of current version 22 March 2019.

Published by the IEEE Computer Society

Figure 1. Examples of color sketches created by artists.

This brings an increasing demand for novel automatic style transfer techniques in computer graphics and multimedia technology.

Recently, many neural network based algorithms have yielded impressive results on transferring the style from one example image to other images.[3,6] In their works, a deep convolutional neural network (CNN) pretrained for image classification is utilized as a feature extractor. Style transfer is then realized by minimizing the difference between the style representation of generated image and style target image as well as the difference between the high-level feature activations of generated image and content target image. These algorithms are inspired by the CNN-based texture synthesis,[7] in which the Gram matrix is used to characterize the image style. Their remarkable results demonstrate that a neural network based algorithm is good at transferring impressionist painting style to input images, imitating art works from famous artists like Vincent van Gogh and Pablo Picasso. Following this direction, we can think that neural network based algorithms[3,6] might also be efficient in transferring color sketch style to natural photographs. Here, we provide one example in Figure 2. Through the comparison between Figure 2(b) and (c), it can be observed that finding a style target image with similar texture statistics is crucial for a better quality color sketch style transfer. This is due to two reasons. First, there exists a great difference between color sketch artworks and impressionist oil paintings: oil paintings usually feature heavy textures, while color sketches are usually smooth and more sensitive to defects, such as redundant sketches and color variations. Second, models proposed in previous studies[3,6] capture image styles as multiscale texture representatives, in which sematic content and local textures are not separated. Thus, as shown in Figure 2(c), redundant sketches and color variations exist in the result when the style target is not suitable for the content image. This problem remains in the real-time style transfer approach,[6] since it also adopts one single image as the style target during the whole training phase.

Besides, both the original neural style transfer method[3] and the real-time approach[6] do not provide gouache colors, which are fundamental characteristics of color sketches. Therefore, it is extremely difficult to obtain pleasurable color sketch style transfer results with existing neural style transfer approaches.

In this paper, we present a CNN-based fully automatic system, which transforms natural photographs to color sketches. Our system extends the real-time style transfer[6] and is capable of converting an arbitrary natural photograph to color sketch. During the training phase, instead of using a single image as the style target, we build a set of color sketch examples with different texture statistics. We also develop an online style target selection method to pick up a suitable image as the style target for every training content image. Meanwhile, we improve the image transfer network by designing a novel CNN architecture with multiscale refinement and dilated residual learning. In the final step of our system, we perform the linear color transform which is proposed in one state-of-the-art interactive color-sketch drawing system[1] and a guided filtering operation to enhance gouache colors in our generated style transfer images. To our

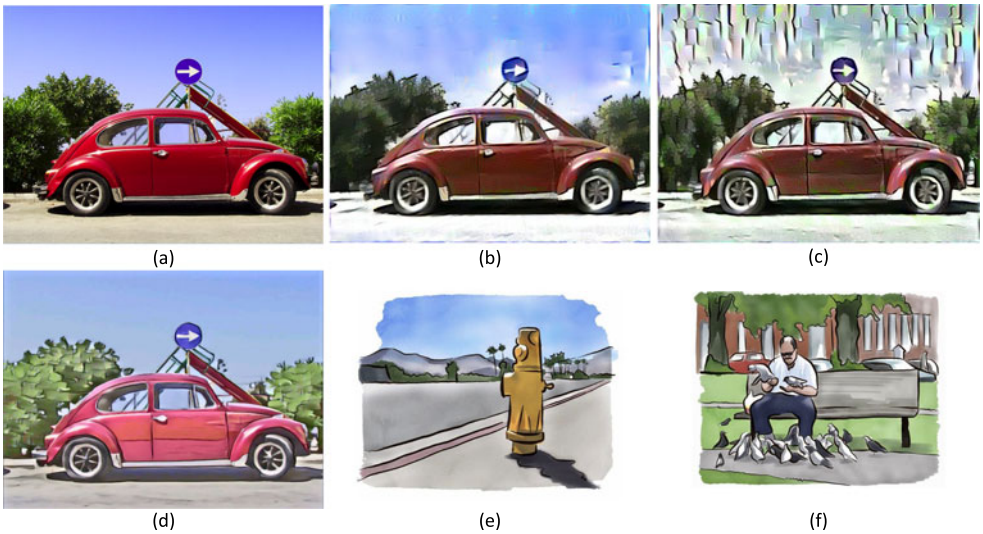

Figure 2. Example results for color sketch generation using the original neural style transfer method[3] and our proposed system. (a) Original image. (b) and (c) Style transfer results using the original neural style transfer method[3] but adopting (e) and (f) as the style target image, respectively. (d) Automatically generated color sketches using our proposed system. (e) and (f) Real color sketch artworks selected from our proposed style example set, both of them are drawn by an artist.

knowledge, our system is the first one to tackle this color sketch generation problem using deep CNN. In addition, it is also noteworthy that there are various kinds of painting styles in color sketch artworks, and our system aims to generate an image with typical and fundamental characteristics of color sketches, such as stylized line sketches and gouache style colors.

The major contributions of this paper can be summarized as follows.

- We set up a color sketch example set and develop an online style target selection method to overcome the shortcomings caused by using one single image as the style target in original neural network based style transfer methods.
- We design a novel style transfer network, called STSRnet, to generate high-quality color sketches. The proposed architecture integrates state-of-the-art techniques, such as top–down refinement, multiscale feature fusion, dilated convolution, and residual learning.
- The whole system could be trained in an end-to-end manner and generates pleasurable color sketches efficiently during the inference phase without any iteration.

RELATED WORK

Neural style transfer: Automatic image style transfer has been extensively studied in the computer graphics literature, and also has various applications in industry. Recently, Gatys *et al.*[3] drew inspirations from CNN-based texture synthesis[7] and achieved remarkable image style transfer results using an iterative optimization method.

Typically, a content image and a style target image are provided, and the Gram matrix is utilized to measure the correlations between different channels of feature activations and represent the image style. Feature activations in different layers are used to represent image content. A pretrained deep CNN[8] was used to extract the feature activations from its intermediate layers. During optimization, the transferred image was generated by simultaneously minimizing the style and content losses. More recently, there are many follow-up works. Gatys *et al.*[4,5] proposed an algorithm to preserve the color of content image and transfer different styles to different regions in input images according to the semantic information. Johnson *et al.*[6] sped up the time-consuming iterative optimization in original neural network-based style transfer by

building a feed-forward CNN to tackle the style transfer task. However, among existing CNN-based automatic style transfer methods and applications, none of them is designed specifically to tackle the problem of transferring gouache painting style to natural images.

Color sketch generation: Usually, generating pleasurable color sketches requires artistic abstractions of image content, which is challenging for fully automatic algorithms to extract. Thus, state-of-the-art style transfer systems for color sketch generation such as Li *et al.*[1] and Wen *et al.*,[2] all resort to users' interactions to perform scene parsing. Although significant efforts were made in their work to design an interface for efficient interactive scene parsing, it was still time consuming. In order to obtain high-quality color sketch results, it also demanded considerable users' sketching skills.

Lu *et al.*[9] converted natural images to vivid pencil drawings by rendering edges and adjusting tone automatically. However, since this algorithm directly transformed image gradients to pencil strokes, plenty of details still existed in the results, which led to messy sketches in complex scenes or textured regions.

CNN Architectures for image transform: Deep CNNs are wildly adopted in image transform tasks such as image colorization,[10] image to image translation,[11] real-time style transfer[6] mentioned above, and texture synthesis.[7] Although their architectures differ in detailed settings, basically they learn from CNN architectures built for image classification[8,12] and semantic segmentation[13,14] because all of them could be regarded as CNN-based dense pixelwise prediction tasks. Among them, consistent efforts have been made to improve the spatial resolution of the final prediction. Here, we conclude three noteworthy trends among recent CNN architectures designed for dense pixelwise prediction tasks: dilated convolutions,[14] residual learning,[6,12] and top–down refinement.[15]

Besides, instance normalization[16] developed from batch normalization[17] also benefits the training process by normalizing the contrast of each image separately. In this paper, we propose an image transform CNN architecture consisting of all above state-of-the-art techniques.

OUR APPROACH

Our system is extended from the perceptual loss based real-time style transfer method.[6] In the training phase, our system learns a convolutional image transform network $\mathrm{f}(\mathrm{w})$ by applying two complementary perceptual losses $l^{\Phi}_{\mathrm{content}}$ and $l^{\Phi}_{\mathrm{style}}$, which represent image content reconstruction and color sketch style transfer, respectively. Instead of directly calculate the pixelwise distance, each perceptual loss adopts a pretrained CNN to measure the perceptual difference between the generated color sketch image and target images. The image transform network converts the input image to color sketch which is then fed into different perceptual loss networks. The gradients of all perceptual losses are accumulated and backpropagated to update the parameters in the image transform net, thus the whole pipeline could be trained in an end-to-end manner.

Although integrating three perceptual loss networks increases the total size of our system during the training phase, our training is still efficient since the parameters in the perceptual loss networks do not need to be updated. More importantly, all the perceptual loss networks could be discarded during inference. The final output is generated with a single feed-forward path of a learned image transform network and a gouache color transform processing.

Style Transfer Network With Spatial Refinement

In this section, our style transfer network with spatial refinement (STSRnet) is introduced. STSRnet is a novel architecture of CNNs to realize the end-to-end style transfer. The detailed architecture of the proposed network is introduced and compared with the image transform network proposed by Johnson *et al.*[6] Johnson's network[6] mainly consists of a bottom-up feed-forward path stacked with several residual blocks. Dilated convolution layers are adopted in the residual blocks to aggregate the semantic information through the feed-forward path at the cost of sacrificing spatial resolution. Finally, with additional deconvolution layers stacked on the top of the residual blocks, the final output image of their network is up-sampled back to the same spatial resolution of the input image.

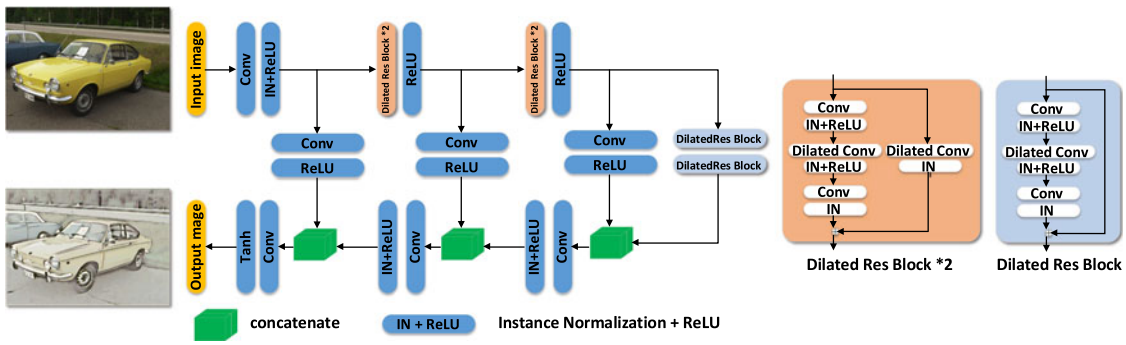

Figure 3. Architecture of the proposed STSRnet and dilated residual blocks.

Our proposed architecture improves Johnson's network[6] and is designed with two goals: one is to aggregate semantic information without sacrificing the spatial resolution; another goal is to fuse the feature activations of low-level layers with high-level semantic cues. We achieve both goals by building the whole architecture with a refinement strategy and employing dilated convolutions[14] in residual blocks. The proposed architecture is illustrated in Figure 3. Specifically, our architecture consists of two paths: a bottom–up path, which stacks dilated residual blocks with receptive fields of increasing size, and a top–down path, which gradually concatenates feature activations from low-level residual blocks in each refinement stage. The final output image has the same spatial size as the input image. Such refinement strategy has been proved effective for object segmentation.[15] To our knowledge, our system is the first one to adapt this refinement strategy for style transfer task. More importantly, comparing with the VGG-based[8] deep refinement networks proposed by Pinheiro *et al.*,[15] dilated residual blocks and instance normalizations are both employed in our architecture to improve the quality of transformed image.

Dilated Residual Blocks: As shown in the right part of Figure 3, two types of residual blocks are utilized in the proposed architecture. Both of them consist of a main path which contains three convolution layers and a shortcut path for residual learning. "Dilated Res Block *2" denotes the residual block which doubles the number of feature channel. It contains a dilated convolution layer[14] both in the main path and the shortcut path. "Dilated Res Block" denotes the residual block without changing the channel number of its input and output. In this case, the shortcut path is simply an identity mapping.

Instance Normalization: Following the suggestions given by Ulyanov *et al.*,[16] we adopt instance normalization layers in our architecture to incorporate instance-specific contrast normalization. The major difference between the instance normalization layer and batch normalization layer is that the former performs normalization for the feature activations of a single image instead of for the whole batch. It, thus, encourages the contrast of the stylized image to be similar to the contrast of the style target image and improves the quality of style transferred images.

Loss Design and Training Method

As discussed in the introduction, when training the style transform network, using a single image as style target usually leads to artifacts due to the structural and color mismatch between training images and style targets. We address this problem by building a style image set and applying an online style target selection method during the training phase.

Style Image Set: In order to create a set of color sketch examples images with variants of structures, we select a set of representatives from the training data according to the semantic cues of each image. Specifically, for each training image, we first extract the feature activation F^l with N_l channels of the lth intermediate layer using a pretrained image classification CNN Φ. In our algorithm, VGG16[8] is used as the pretrained image classification CNN Φ. Then, the Gram matrix $G^{\Phi,l} \in R^{N_l \times N_l}$ is calculated according to F^l.[3] Each element of $G^{\Phi,l}$ represents the correlation between feature channels c and c' of $F^{\Phi,l}$ by

calculating inner product

$$G^{\Phi,l}_{c,c'} = F^{\Phi,l}_{c} F^{\Phi,l}_{c'}.$$

Note that $F^{\Phi,l}_{c}$ and $F^{\Phi,l}_{c'}$ are reshaped to one-dimensional (1-D) vectors before calculating the inner product.

Then, we cluster all the training images into C clusters using k-means based on the Gram matrix. To reduce the dimensionality, we perform clustering with only the diagonal elements of the Gram matrix. These elements are considered as the global structure and color descriptor because different channels of the high-level feature activations of the deep CNN already encodes rich semantic cues in these diagonal entries. Finally, we build the set of color sketch examples by letting a few artists draw the C center images. In our experiments, we find that a set of C = 10 color sketch images covers sufficient structure and color variants. The clustering results and our color sketch examples are provided in the supplemental material. We could observe that images in the same cluster are similar in structure, texture, and color, while the center images of different clusters cover certain variations.

Online style target selection: During the training stage, we search for nearest neighbor $n_{k_i^*}$ for each input image x_i in the set of cluster center images, and use the corresponding color sketch $S_{k_i^*}$ of the nearest neighbor as the style target image

$$k_i^* = \underset{k}{\operatorname{argmin}} \left\| f^{\Phi,l}(x_i) - f^{\Phi,l}(n_k) \right\|_2^2$$

where k_i^* represents the index of the style target in the proposed example set S for x_i. $f^{\Phi,l}(*)$ denotes the vector of the diagonal elements of gram matrix $G^{\Phi,l}$. We compute feature activations for this selected style target at layer relu4_2 in Φ and set l = 21 accordingly. To improve the efficiency of the training stage, the indices of the selected style target images for all training images could be obtained in advance and recorded in a lookup table.

Accordingly, the style loss function in our color sketch style transfer model is defined as follows:

$$l^{\Phi}_{\text{style}}\left(f_W, x_i, s_{k_i^*}\right) = \sum_{l \in L} \left\| G^{\Phi,l}(f_W(x_i)) - G^{\Phi,l}\left(S_{k_i^*}\right) \right\|_2^2.$$

We calculate the style loss at layer relu3_3 and relu4_3 in Φ and set $L = \{16, 23\}$ accordingly.

Combination of perceptual losses: We define the total loss function for our color sketch style transfer as a weighted summation of all the perceptual losses

$$L = \lambda_1 l^{\Phi}_{\text{content}}(f_W, x_i) + \lambda_2 l^{\Phi}_{\text{style}}\left(f_W, x_i, s_{k_i^*}\right).$$

We utilize the same content loss as in previous style transfer systems[3,6]

$$l^{\Phi}_{\text{content}}(f_W, x_i) = \sum_{l} \left\| F^{\Phi,l}(f_W(x_i)) - F^{\Phi,l}(x_i) \right\|_2^2.$$

During training phase, the total loss function is minimized using stochastic gradient descent and the gradients are backpropagated through the network to update parameters in the proposed STSRnet f_W. In our experiments, we set $\lambda_1 = 1$ and $\lambda_2 = 4$, respectively.

Gouache Color Transform

To generate a visually pleasurable color sketch artwork, it is also very important to convert existing pixel colors to gouache-style colors automatically. As discussed by Gatys *et al.,*[4] one potential shortcoming of the original neural network based style method is that the color of the style image is copied to the final result. As a result, it often leads to bizarre results when the color distribution of the style target image is not suitable for the input image. Although the style example set and the style target selection method proposed in our system alleviate such mismatch, unsatisfying colors still exist in the results. Also, the CNNs tend to learn a mean color (e.g., gray) during training. Several works have been proposed to extend the original methods to preserve the color distribution of the content image.[4,5] However, gouache-style colors are not provided in these methods. Therefore, we propose one postprocessing method, which can solve above shortcomings and adding vivid gouache-style color to the transferred results effectively.

Specifically, we discard the color channels of the image generated by the STSRnet and apply the linear transform given by Li *et al.*[1] to predict the desired gouache color for each pixel. The detailed operations of the proposed color

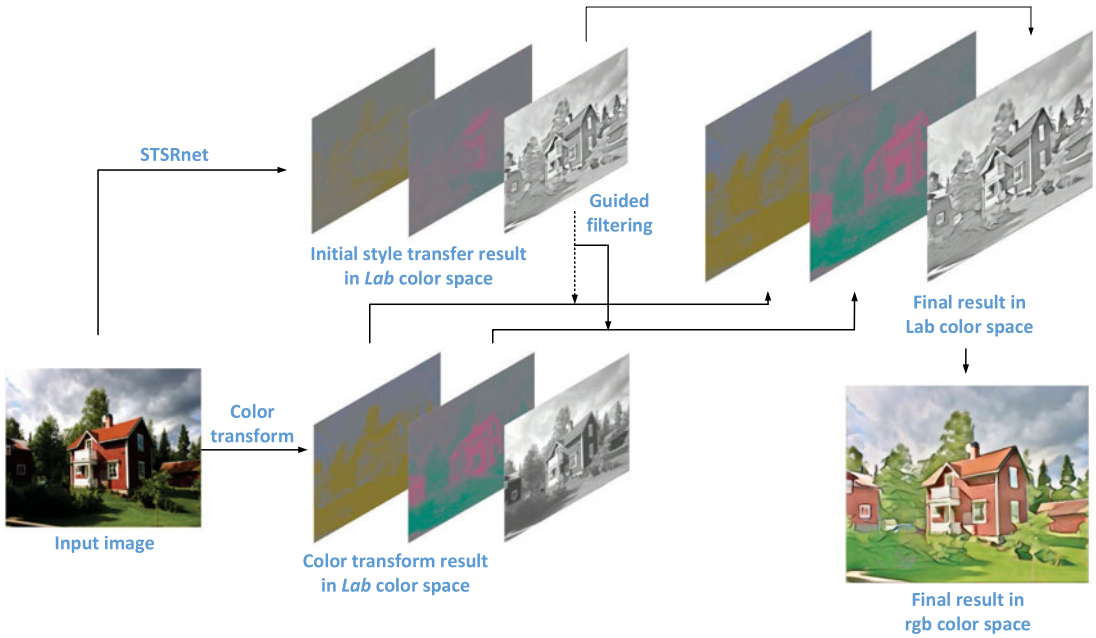

Figure 4. Pipeline of proposed gouache color transform.

transform are illustrated in Figure 4. Specifically, we obtain the initial style transfer result using the proposed STSRnet, discard the chromatic channels, and only keep the lightness channel. Meanwhile, the linear color transform given by Li *et al.*[1] is applied to the input image and the color transform result image is converted to CIE*Lab* color space. The *a*, *b* channels of the color transform result image are subsequently filtered with the lightness channel of the initial style transfer inference as a guidance. The benefits of this guided filtering here are twofold: it aligns the *a*, *b* channels with the lightness channel of the initial style transfer result in case the locations of the pixels on line sketches deviate from the original object boundaries during style transfer; it also removes fine chromatic variations of input image. Finally, we obtain the final color sketch by concatenating the lightness channel of the initial style transfer result with the filtered *a*, *b* channels and converting it back to RGB color space. Some examples are provided in the supplemental material.

RESULTS AND COMPARISON

We adopt the Microsoft COCO dataset[19] as the training data for the proposed STSRnet. In order to obtain the color sketch version of the images in our style image set, we invite an artist to manually draw 10 images using digital pen tablets and Photoshop. During the training phase, each image in the COCO dataset is resized to 256×256 and the minibatch size is set to 4. We use Adam to train the proposed STSRnet. The total number of iteration is set to 10K and the learning rate is 1×10^{-3}. Our proposed STSRnet has been extended from the code of real-time style transfer[6] on the popular deep learning framework Torch. An Nvidia Titan X GPU is used in our experiment. Our training takes around 8 h. During the inference phase, it only takes less than 0.5 s for an input image with 512×512 pixels. Galleries of color sketches generated by our system are provided in the supplemental material.

Ablation Study

To discover the effectiveness of our proposed style image set and STSRnet, we conduct an ablation study to compare generated images of our system under different settings.

In Figure 5, panel (b) shows the result of our system which employs the proposed style image set and the style target selection method during training. Panel (c) shows the result when one single image is used as the style

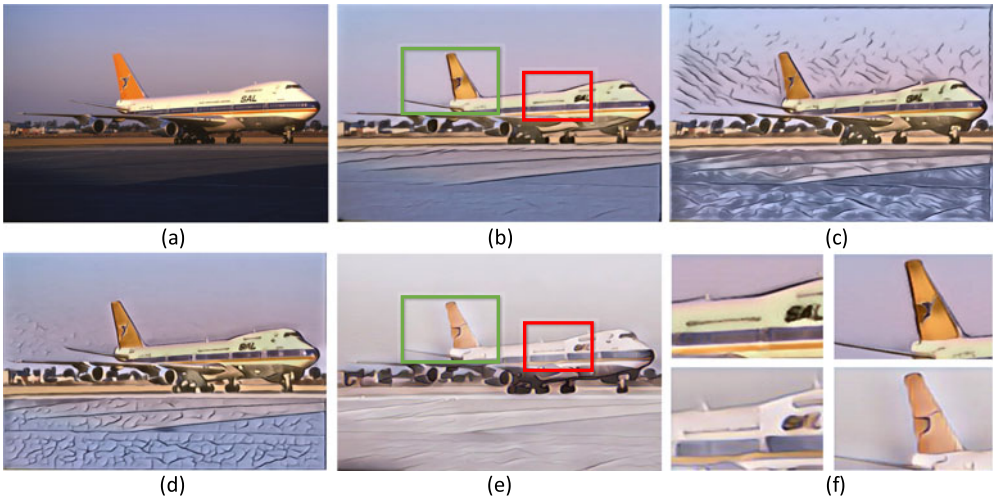

Figure 5. (a) Input image. (b) Our result. (c) Result obtained by our system but randomly selecting one style image from the proposed style image set and fixing the selected image as the style target during training. (d) Result obtained by our system but randomly selecting one image from the proposed style set for each input image as the style target during training. (e) Result obtained by our system but replacing our proposed STSRnet with the CNN architecture in the real-time style transfer method.[6] Gouache color transform is not involved. (f) First row shows enlarged regions in (b) while second row shows enlarged regions in (e).

target. Specifically, we randomly select one sketch image from the proposed style image set then fix the selected image as the style target for all input images during training. Since a single style image could not match the structure and color of each training image, it leads to severe artifacts in the style transfer results, such as a lot of meaningless sketches. Panel (d) shows the generated result when all the images in the proposed style image set are adopted but the proposed style target selection method is not utilized. Specifically, for each input image, one style image is randomly selected from the proposed color sketch image set then set as the style target for this input image during training. The result shows that the artifacts remain due to the mismatch between the training image and the style target image in terms of their structure and color. Through the comparison between panels (b) and (c), (d), we can observe that the proposed style image set and the style target selection method can greatly reduce the artifacts in the generated style transfer images and produce high-quality synthesized color sketch images. Our proposed STSRnet is used as the image transform model in this comparison. In addition, panel (e) shows result obtained by our system but replacing our proposed STSRnet with the CNN architecture in the real-time style transfer method.[6] Enlarged regions are shown in panel (f). Through the comparison between panels (b) and (e), we can observe that our proposed STSRnet contributes significantly to better resolution (e.g., patterns on the airplane) and line sketches (e.g., contour of the airplane).

Comparison With State-of-the-Art Neural Style Transfer Methods

Figure 6 compares color sketches generated by our system with the real-time style transfer method,[6] the color preserving neural style transfer method,[4] and the deep image analogy.[18] Since the real-time style transfer[6] only allows a single image as style target, in this experiment, we train the real-time style transfer model by randomly selecting one color sketch from our style target set and fix it as the style target for all training images. We also set a same ratio between the content loss and style loss in our system and the real-time style transfer to ensure a fair comparison. From column (b) in Figure 6, we could observe that our system consistently outperforms the original real-time style transfer method. Column (c) compares our results with the color preserving neural style transfer

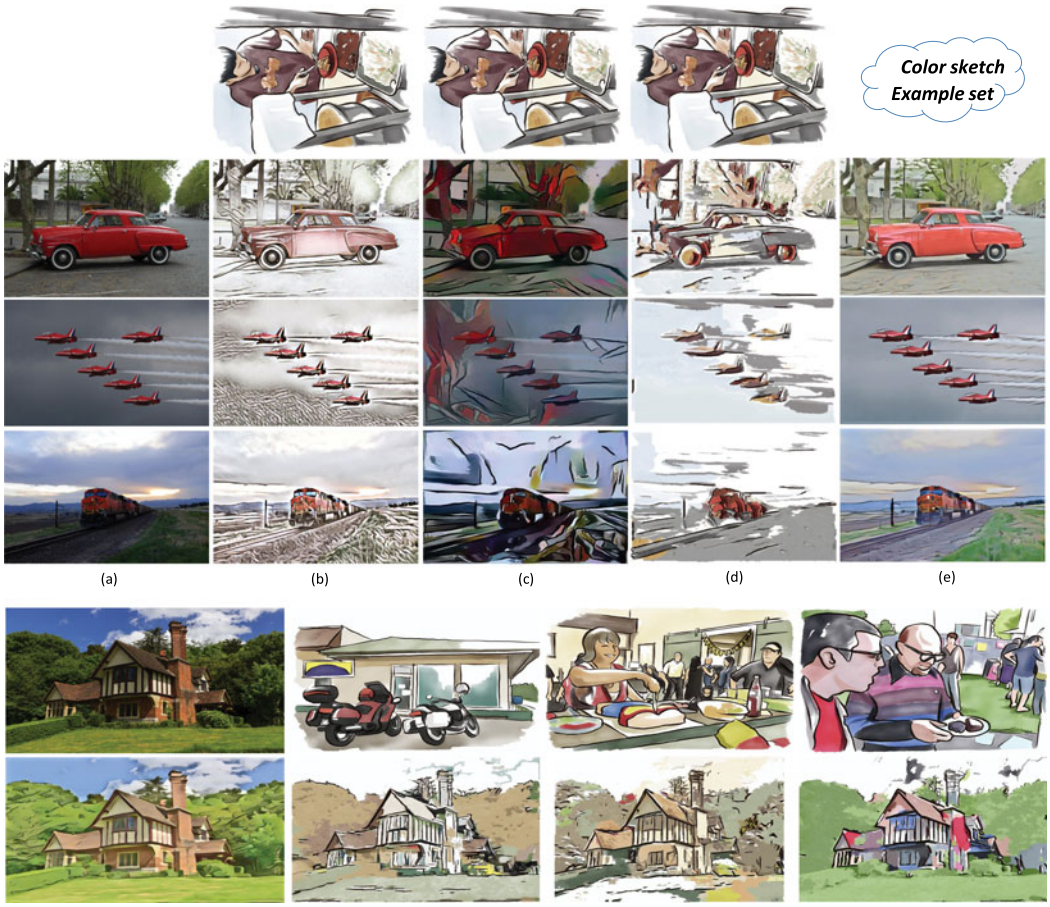

Figure 6. Comparison with state-of-the-art neural style transfer methods. In the upper part, first row shows the corresponding style target image or image set used to obtain style transfer results. For the remaining rows, column (a) shows the original photographs. Columns (b)–(e) show results generated by the real-time style transfer method,[6] the color preserving neural style transfer method,[4] deep image analogy,[18] and our system. Our system demonstrates superior performance. In the lower part, we further provide one detailed comparison between our method and the deep image analogy method. In the first column, original content image is shown in the first row and our result is shown in the second row. In the second to fourth columns, the style target images are shown in the first row while the corresponding results generated by the deep image analogy method are shown in the second row. All these style target images are randomly selected from our proposed style target image set. As can be observed, although promising results might be obtained by picking up a more suitable style target image, there usually exist artifacts in the results of the deep image analogy method. Besides, it is hard to generate pleasurable gouache colors by only using the deep image analogy method.

method.[4] As suggested in the color preserving neural style transfer method,[4] the colors of the style image are transformed using the 3-D color matching to match the colors of the content image before doing neural style transfer. For all results in column (c), we use the same style target image as in column (b) and set the ratio between the content loss and style loss as suggested in the original neural style transfer method.[3] All results in column (d) are obtained by deep image analogy.[18] Our results also demonstrate significant improvement compared to the results of the color preserving neural style transfer method[4] and deep image analogy.[18]

Comparison With a State-of-the-Art Interactive System

As shown in Figure 7, we compare color sketches generated by our system with the results obtained by Li's interactive system.[1] Our

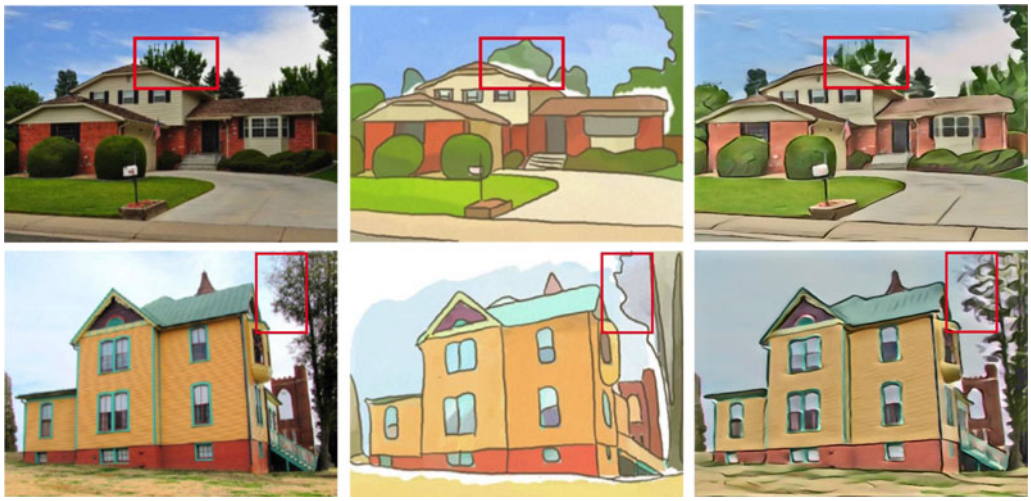

Figure 7. Comparison with a state-of-the-art interactive color sketch generation system.

Figure 8. Comparison with state-of-the-art NPR methods.

results present an overall similar color sketch style as those created manually using the interactive system. The average time consumption for a novice user to create one color sketch using Li's system[1] is 6 min, while our system is fully automatic and highly efficient: it only takes less than 0.5 s for an input image with 512 × 512 pixels.

We also note that in regions with heavy textures, for example, tree leaves highlighted in Figure 7 with red boxes, the style of sketchy lines generated by our system is different from those created based on users' input: dense sketches usually exist in our results, while Li's results prefer to depict tree leaves using very sparse and long curves. This is because Li's system encourages users to perform interactive image segmentation with sparse line strokes, but our system learns image stylization according to perceptual losses rather than performing image segmentation explicitly.

Comparison With NPR Methods

We conduct a comparison with NPR methods which produce stylized images that look similar to color sketches. A comparison is shown in Figure 8, in which the NC filter based image stylization results are generated by superimposing the magnitude of filtered images to the filtered images themselves. Compared with the results of NC filter[20] and Pencil Drawing,[9] our results both highlight the object contours with sparser

sketches and produce stylized regional abstraction and gouache colors.

CONCLUSION

This paper extends the deep neural network based style transfer for color sketch image generation. We have integrated the benefits of image transform network with spatial refinement and automatic target image selection method. Experimental results demonstrate that our system overcomes the limitations of the original neural network based style transfer methods and produces vivid color sketches. Besides color sketch, our ideas of building example style target set and online style target selection method will potentially benefit other noise-sensitive image style transfer tasks.

In the future, one potential direction is to combine the automatic generation pipeline with humans' interactions and investigate how such a data-driven approach could contribute to more smart software to help artists' creation. In addition, another potential direction is to extend our system for video stylization applications.

ACKNOWLEDGMENT

This paper has supplementary downloadable material at http://ieeexplore.ieee.org, provided by the authors.

REFERENCES

1. G. Li, S. Bi, J. Wang, Y. Xu, and Y. Yu, "ColorSketch: A drawing assistant for generating color sketches from photos," *IEEE Comput. Graph. Appl.*, vol. 37, no. 3, pp. 70–81, May/Jun. 2016.
2. F. Wen, Q. Luan, L. Liang, Y.-Q. Xu, and H. Y. Shum, "Color sketch generation," in *Proc. 4th Int. Symp. Non-Photorealistic Animation Rendering*, 2006, pp. 47–54.
3. L. A Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2414–2423.
4. L. A. Gatys, M. Bethge, A. Hertzmann, and E. Shechtman, "Preserving color in neural artistic style transfer," 2016, *arXiv: 1606.05897*.
5. L. A. Gatys, A. S. Ecker, M. Bethge, A. H. Mann, and E. Shechtman, "Controlling perceptual factors in neural style transfer," 2016, *arXiv: 1611.07865*.
6. J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 694–711.
7. L. Gatys, A. S Ecker, and M. Bethge, "Texture synthesis using convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 262–270.
8. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv: 1409.1556*.
9. C. Lu, L. Xu, and J. Jia, "Combining sketch and tone for pencil drawing production," in *Proc. Symp. Non-Photorealistic Animation Rendering*, 2012, pp. 65–73.
10. S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Let there be color!: Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification," *ACM Trans. Graph.*, vol. 35, no. 4, 2016, Art. no. 110.
11. P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," 2016, *arXiv: 1611.07004*.
12. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
13. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.
14. F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv: 1511.07122*.
15. P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár, "Learning to refine object segments," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 75–91.
16. D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," 2016, *arXiv: 1607.08022*.
17. S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, vol. 37, pp. 448–456.
18. J. Liao, Y. Yao, L. Yuan, G. Hua, and S. B. Kang, "Visual attribute transfer through deep image analogy," *ACM Trans. Graph.*, vol. 36, no. 4, 2017, Art. no. 120.
19. T.-Y. Lin *et al.*, "Microsoft coco: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.
20. E. S. Gastal and M. M. Oliveira, "Domain transform for edge-aware image and video processing," *ACM Trans. Graph.*, vol. 30, 2011, Art. no. 69.

Wei Zhang is a Senior R&D Engineer with Baidu, Inc. His current research interests cover computer vision, deep learning, and image processing. He received the B.E. degree from Chongqing University in 2010, the M.S. degree from the Huazhong University of Science and Technology in 2013, and the Ph.D. degree from the University of Hong Kong in 2017. Contact him at zhangwei.hi@gmail.com.

Guanbin Li is a Research Associate Professor with the School of Data and Computer Science, Sun Yat-Sen University. His current research interests include computer vision, image processing, and deep learning. He received the Ph.D. degree from the University of Hong Kong. He is a recipient of a Hong Kong Postgraduate Fellowship. He is a member of the IEEE. Contact him at liguanbin@mail.sysu.edu.cn.

Haoyu Ma is a postgraduate student with the University of Hong Kong, People's Republic of China. His research interests include image processing, computer vision, and signal processing and related areas. He received the B.E. and M.S. degrees from the Department of Information Engineering, Zhejiang University, China, in 2015 and 2018, respectively, and the M.S. degree from the Department of Electrical and Electronic Engineering, Imperial College London, London, U.K., in 2016. Contact him at 21530059@zju.edu.cn.

Yizhou Yu (M'10–SM'12–F'19) is a Professor with The University of Hong Kong and was a Faculty Member with the University of Illinois at Urbana-Champaign for 12 years. He is a recipient of the 2002 U.S. National Science Foundation CAREER Award and the ACCV 2018 Best Application Paper Award. He has served on the editorial board of *IET Computer Vision*, *The Visual Computer*, and the IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS. He has also served on the program committee of many leading international conferences, including ICCV, SIGGRAPH, and SIGGRAPH Asia. His current research interests include computer vision, deep learning, biomedical data analysis, computational visual media, and geometric computing. He received the Ph.D. degree from the University of California at Berkeley in 2000. Contact him at yzyu@cs.hku.hk.