Computational Visual Media
DOI 10.1007/s41095-xxx-xxxx-x Vol. x, No. x, month year, xx–xx

Article Type (Research/Review)

Reference-guided Structure-aware Deep Sketch Colorization for Cartoons

Xueting Liu[1], **Wenliang Wu**[2], **Chengze Li**[1] ✉, Yifan Li[2], and Huisi Wu[2]

Abstract Digital cartoon production requires extensive manual labor in colorizing sketches with visually pleasant color composition and color shading. During colorization, the artist usually takes an existing cartoon image as color guidance, especially when colorizing related characters or an animation sequence. Reference-guided colorization is also semantically more intuitive than colorization with other user hints, such as color points/scribbles and text-based hints. Unfortunately, reference-guided colorization is quite challenging since the style of the colorized image should be similar with the style of the reference image in terms of both global color composition and local color shading. In this paper, we propose a novel learning-based framework which colorizes a sketch based on the color style features extracted from a reference color image. Our framework contains a color style extractor to extract the color feature from a color image, a colorization network to generate multi-scale output images by combining a sketch and a color feature, and a multi-scale discriminator to improve the realness of the output image. Extensive qualitative and quantitative evaluations are conducted. Results show that our method outperforms existing methods in terms of both superior visual quality and style reference consistency in the task of reference-based colorization.

Keywords Sketch Colorization, Image Style Editing, Deep Feature Understanding, Reference-based Image Colorization.

1 Caritas Institute of Higher Education, Hong Kong SAR, China. E-mail: tliu@cihe.edu.hk, czli@cihe.edu.hk.

2 Shenzhen University, Shenzhen, 518060, China.

Manuscript received: 2014-12-31; accepted: 2015-01-30.

1 Introduction

With the increasing popularity of digital cartoons, various computer-assisted technologies for producing digital cartoons have been rapidly developed in recent years. Colorizing cartoon sketches is one of such technologies that has attracted extensive research focus in the field of computer graphics. Since the sketch itself contains no hint on the potential color to be colorized, the existing methods either colorize the sketch by purely guessing [16], [44], [42], which may lead to unnatural colors (Fig. 1(b)), or colorize based on user-provided hints, such as color points/scribbles [37, 40] and text hints [18]. However, manually crafting the color hints is time-consuming, especially for line drawings with complex content. For example, in Fig. 1(c), 62 color points are created by the user to obtain a relatively satisfactory colorized result. Besides, crafting proper color hints is also challenging, especially for amateur users, as it usually requires the user to have certain level of aesthetics to generate a visually pleasant color cartoon. Moreover, when coloring a cartoon animation, it would be extremely difficult for the user to achieve color consistency across frames when color hints are provided for every frame individually.

To resolve the above issues, we propose an automatic system that can colorize a cartoon line drawing based on a reference image. There are two key advantages in reference-guided colorization over the color hint-based one. Firstly, it saves the effort of user trial-and-errors in creating a proper set of user-hints for colorizing the sketches. Instead, the user only needs to provide a reference image that contains a similar color style as he/she targets for in the colorized image. The system can automatically learn the color style from this reference image and apply it to colorize the sketch. Secondly, when the user needs to colorize a set of sketches that contain similar contents, such as

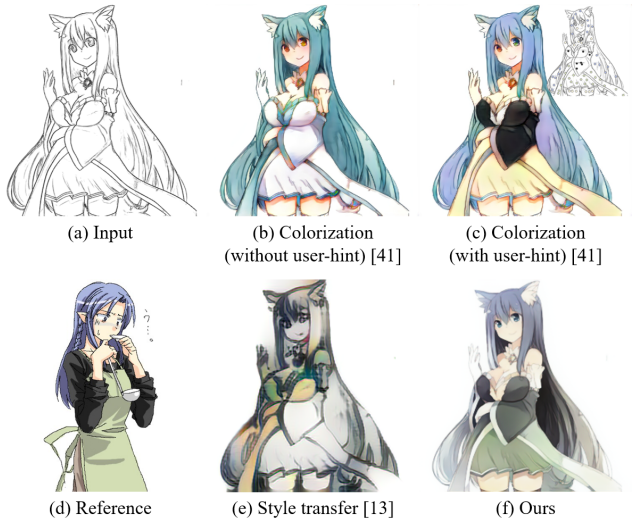


Fig. 1 Comparisons between our method and the existing sketch colorization and style transfer methods.

a sequence of sketchy frames for producing a cartoon animation or a set of sketchy character designs for the same cartoon character but with different poses, the user only needs to colorize one of the sketches in the set and then transfer the color of this image to the other images with ease.

In the current literature, very few works focus on reference-based colorization and mostly suffer from low colorization quality, as it is considered challenging to properly propagate the visual styles of the reference to the sketch input. However, various existing techniques tailored for other applications might be borrowed for this application. The style transfer methods [13, 24] take an input image and a style image as input and transfer the style of the style image to the input. While these methods may be adapted to apply the style of the reference image to the input sketch in our application, the structural lines in the sketch generally cannot be well preserved where the result image may exhibit obvious artifacts (Fig. 1(e)). The content-style disentanglement methods may separate the style from the content [7, 38], but they generally cannot ensure the the content component to be the exactly the structural lines in the sketch. As a result, the encoded style space is usually not perpendicular to the content space, which further affects the quality of the colorized result.

In this paper, we propose a novel learning-based solution for colorizing cartoon sketches based on reference images. A key observation of our application is that the structural lines in the sketch should be preserved during colorization. To preserve the structural lines, we formulate the colorization network as an image-to-image translation problem where the input is a cartoon sketch and the output is its colorized colorized version based on the color style of a reference image. To colorize the sketch based on a reference image, we extract the color style of the reference image via a color style extractor and then fuse the extracted color style feature into the deep hierarchical representations of the sketch via adaptive instance normalization (AdaIN) [13]. In order to improve the visual quality of the generated color output, we propose to adopt a multi-scale discriminator which can improve the realness of the output image in terms of both global color composition and local color shading. We apply our method on various images. convincing results are obtained in all cases.

The contributions of our method can be summarized as following:

- We propose a novel reference-based cartoon sketch colorization method via a deep learning approach where manual color hinting is not needed.
- We formulate sketch colorization as an image-to-image translation problem where the structural lines in the sketch can be faithfully preserved.
- We adopt a multi-scale discriminator to improve the visual realness of the generated cartoon in terms of both global color composition and local color shading.

2 Related Work

There are plenty of existing methods that can achieve reference-guided colorization. We can roughly categorize these methods into three major categories: image style transfer, conditional image-to-image generation, and style-content disentanglement.

Image Style Transfer Image style transfer aims at alternating the style of an image, such as colors and textures, based on a style exemplar. [12] proposed an style editing method by learning the texture mapping from the pyramidal features based on the exemplars. However, the paired supervision highly restricts the possible usage to our colorization task. Recently, thanks to the revamped construction of image features with deep neural network models, style transfer tasks are enhanced with higher output quality and style editing precision. Among these methods, [6] proposed to adopt a content loss and a style loss to handle different image components in the feature domain to achieve style transfer. [17] further proposed the perceptual loss as a universal metric for all style-related editing tasks. [13] extended the style transfer

method to perform on-the-fly style alternation with a single neural network by a feature rescaling technique named adaptive instance normalization. Later, more methods were proposed to achieve better visual quality [24, 28, 30, 34, 43], or efficiency [5, 23]. However, the style transfer methods are usually designed with global style objectives and cannot be directly applied to propagate local characteristics of the reference style. In other words, these methods cannot apply customized local color and textures according to the underlying structural constraints (e.g. Fig. 1(e)). In contrast, our method not only extracts the global color style to guide the colorization but also cares about detailed local color shading.

Deep Conditional Image-to-image Synthesis Deep generative models have been widely used to accomplish cross-domain image-to-image translation tasks. These methods facilitate the adversarial training scheme [8] to synthesize images with natural looks. The training can be either conducted under paired supervision [16] or cycle-consistency [19, 44]. While these methods achieved decent image translation quality and can be used for sketch colorization [42], these methods are generally deterministic, i.e., they cannot create diversified outputs based on different user requirements.

Conditional image generation methods extended the generative model to allow conditional user inputs. [37, 41] proposed to use scribbled or pointed color hints as the conditional input to guide the colorization. However, they cannot use full-page cartoon pictures as reference. Additionally, placing color hints is time-consuming and requires users' experiences. [18] proposed to read a list of text-level visual tags as the condition to decorate the sketch input with the given visual property. However, these visual tags may not well capture the color style the user targets for, as reference images do.[40] attempted to apply the style of the reference cartoon image to an arbitrary sketch input with a feature-based encoder and decoder design. However, the output quality is not promising with blurry artifacts. Besides, the color style of the result may not be consistent with that of the reference image. Different from the above conditional colorization methods that needs explicit visual cues, [45] proposed to encode the visual styles into a latent space with a style encoder. During the image translation process, the style of the reference image is extracted as the conditional input. However it lacks a strong style editing mechanism for complex datasets such as comics and cartoons, which contain almost unlimited color and texture combinations. More recently, [22] proposed a reference-based sketch colorization method based on an transformation-aware attention module. However, this method still shares the problem of color style inconsistency between the reference and the output. We will further demonstrate this weakness in Section 4.

Reference-based Photo Colorization Many works attempt to colorize photographs with reference-based priors. The pioneering work [36] proposed to transfer the chromatic information to the corresponding regions by matching the imminence and texture. Various correspondence techniques [2, 3, 9, 25, 32] have been proposed to improve the result of local color transfer by hand-crafted low-level features. Still, these correspondence methods are not robust to complex appearance variations of the same object because low-level features do not capture semantic information. With the deep learning development, recent studies [11, 39] proposed to compose semantically close source-reference pairs based on features extracted from the pretrained networks [11, 39] and exploit their semantic correlation for colorization. Although they may have good performance in the photo colorization task, their performance in sketch colorization is relatively weak. The problem is due to the abstract nature of sketches, which cannot offer enough visual semantic cues for dense color propagation, as widely used in photo colorization.

Image Style-content Disentanglement We may also achieve reference-guided colorization by a style-content disentanglement, in which images can be dismantled into a *content* space and a *style* space. The content space encodes the structural information, while the style space encodes the colors, textures, and other style-related information. With the disentanglement techniques, multi-modal [7, 14, 21] or multi-domain [38] image-to-image translation can be achieved with structure preservation. Moreover, these methods allow aligning the output image style to the reference by converging their style space representations. While the disentanglement methods manage to create smooth translations across different image categories or styles, they always encode the content (or the structural information) of the images into latent feature vectors, which are generally different from the structural information in the line art. However, in our task, the structural lines in the line arts should be exactly the same as the content of the colorized image so that the

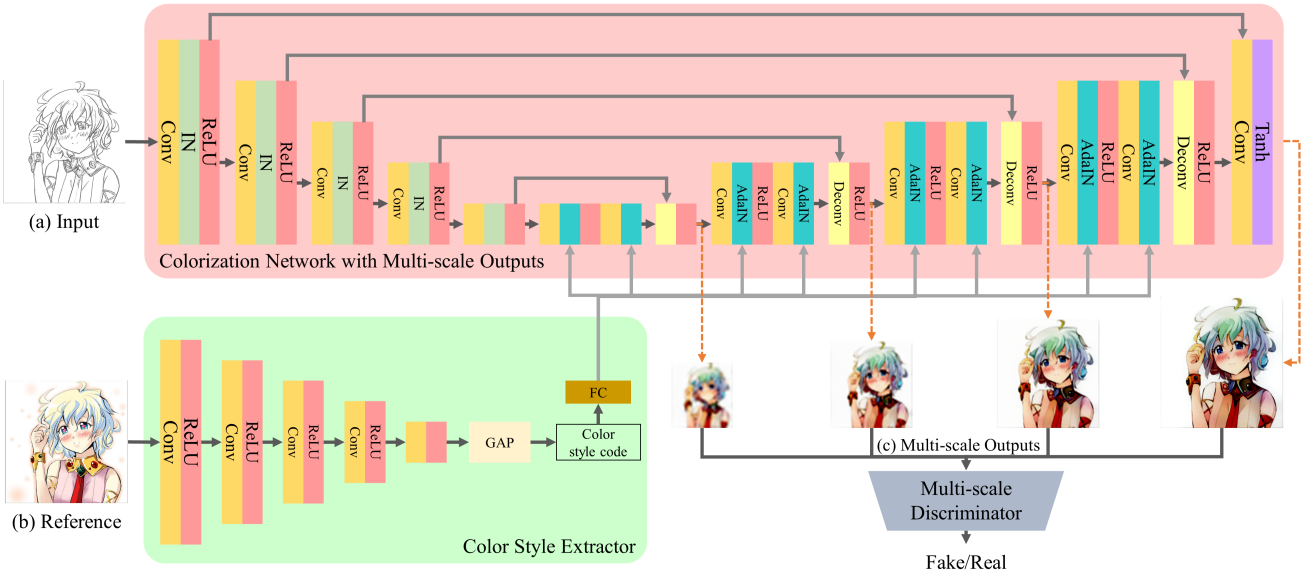


Fig. 2 Overview of our framework. Our framework takes an input image (a) and a reference image (b) as input, and outputs a color cartoon (c) that is consistent with the input in content and consistent with the reference in style.

structure of the line art can be preserved. Therefore, the style-content disentanglement methods generally cannot be directly applied in our application.

3 Method

3.1 Overview

The key insight of our proposed method is to achieve reference-guided sketch colorization by specifying the sketch as the content component in the colorized cartoon image. During the colorization process, we transfer the color style features of reference image into the sketch input to create our final color cartoon output. We propose a deep learning framework to tackle this challenging problem. In this section, we first present the detailed network design of our proposed sketch colorization framework. Then we discuss the training process of our colorization framework, including training dataset preparation, loss function design, and training configuration.

3.2 Network Design

Traditional colorization networks do not allow image-based color style reference or have limited abilities to read from users' color guidance. Our goal is to automatically recognize the color styles from the guidance image and apply the color style to the input sketch with consistency in color composition and shading. To achieve so, we first propose a style extraction network that takes the style guidance image as input and outputs a representative style code. The style code contains essential color style information of the guidance image. It will be further used during the colorization process to regularize the style of the output. We subsequently propose the colorization network for fusing the color style of the guidance image and the deep semantics of the input sketch to create the final color cartoon output with consistent styles to the guidance image and consistent content to the input sketch. We also propose a multi-scale discriminator to ensure realistic colorization in terms of both global color composition and local color shading. The overall network structure is shown in Fig. 2.

3.2.1 Color Style Extractor

First of all, we propose a color style extractor to capture a diversity of style variations of our training data into a unified color style space representation. The design of the color style space aims at collecting different color styles as much possible and excludes the structural information from the style representation.

To achieve so, we design a 4-block downscaling residual network [10] as the style extractor. We input the style guidance image into the style extractor, and use the global average pooling to output a style code with a dimension of 256. Our fully convolutional style extraction can extract the style code of reference image of any resolution. But for the convenience of training, we resize the image to 256×256 and input it to the style extractor. We can combine the extracted color style features with any sketch to generate a new color cartoon image.

3.2.2 Colorization Network with Multi-scale Outputs

The colorization network takes a cartoon sketch as input, colors it according to the color style code extracted from the reference image, and outputs a color cartoon with consistent content to the input and consistent style to the reference. Our colorization network is designed based on the U-Net structure, where a downscaling sketch encoder and an upscaling style-content fusion decoder. The encoder transforms a sketch image into deep feature maps with rich semantic information. The decoder fuses the color style code with the high-level feature maps of the input sketch to align the color style of the output to the reference. Different from existing colorization networks that usually produce only one output image, we propose a multi-scale output mechanism where the multi-scale outputs can help to train the discriminator to better distinguish realistic cartoon images in terms of both global and local color characteristics. In particular, low-resolution output images help more in improving the global color composition, while high-resolution images help more in improving local color shading.

In the encoder, we use 5 levels of downsampling blocks to obtain a hierarchical feature map with rich semantic information of the input sketch. We added instance normalization [33] to the encoder to ensure the style information erasure [13] in the sketch. The feature map is then fed to the decoder. The decoder contains 5 upsampling blocks. We use concatenation operations to propagate information from the encoder to the decoder for better synthesis and reconstruction. The final output has the same resolution as the input sketch. In the decoder, we use the AdaIN layers [13] to control the statistics of the feature map to achieve style editing and alignment to the reference image. After each upsampling block of the decoder, we will produce an extra output image of a lower resolution, which will be fed to the multi-scale discriminator and are also useful in constraining the loss function of our system. More details will be explained in Sections 3.2.3 and 3.3.2.

3.2.3 Multi-scale Discriminator

We further employ a multi-scale discriminator to regularize the colorization network towards more realistic cartoon image generation. Different from the commonly used discriminator network that judges the generation quality based on a single input, we allows the discriminator to be compatible with different resolution of the generator output so that the receptive field of discrimination can be vastly improved. We find that such design may lead to better colorization quality, especially in global color composition. Figure 3 shows one visual comparison between adopting single-scale and multi-scale discriminators. In (c), without single-scale only, the discriminator fails to capture the global color composition of the output image, such as the color of the legs and the eyes.

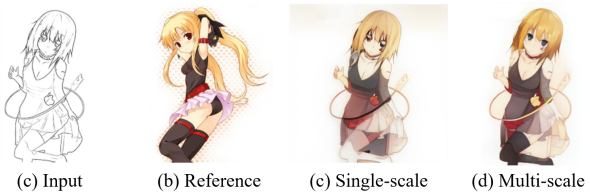


Fig. 3 Comparison between with and without multi-scale outputs and losses.

In particular, we construct our discriminator with 3 downscaling residual blocks. The discriminator is patch-based [16] and we will compute the mean of the output as the discriminator output. The adversarial objective will be introduced in Section 3.3.2.

3.3 Training

3.3.1 Training Dataset

To train our networks, we use the public available dataset [1] which contains paired data of color cartoons and their corresponding sketches. There are altogether 17,769 pairs. We use 14,224 pairs for training and 3,545 for evaluation.

During training, for each color cartoon and its corresponding sketch, we take the sketch as the input image and feed it to the colorization network. Then we take the color cartoon as the reference image and feed it to the color style extractor. Ideally, the output image should take the content from the sketch and take the color style from the color cartoon, i.e. the output image should be the same image as the color cartoon.

3.3.2 Loss Function

Our loss function contains two loss terms, a multi-scale reconstruction loss and a multi-scale adversarial loss.

Multi-scale Reconstruction Loss The reconstruction loss ensures the functionality of style extraction and style propagation by estimating the reconstruction ability of the colorization network. This is done by estimating the differences between a ground-truth color cartoon and the network output by the color cartoon as style guidance and its sketch counterpart as the input.

By minimizing the reconstruction error, the network can better learn the style encoding in a more precise way and obtain a color output similar with the ground truth. As mentioned before, our colorization network provides multi-scale versions of the colored output, we employ our reconstruction loss to each levels of the output so that the network can learn the colorization and style propagation in a coarse-to-fine manner and balance the learning load for all upscaling convolutions. To be specific, for each level of output resolution, the reconstruction loss contains the perceptual loss [17] and the pixel-wise mean square error (MSE). The final reconstruction loss will be applied to all output levels as a weighted sum, which is defined as:

$$\mathcal{L}_{rec} = \frac{1}{n}\sum_{i}^{n} \lambda_i \left\|\varphi(\hat{y}_i) - \varphi(y_i)\right\|_2^2 + \frac{\omega_1}{n}\sum_{i}^{n} \lambda_i \left\|\hat{y}_i - y_i\right\|_2^2 \tag{1}$$

Here, $\hat{y}_i$ is a predicted image of a certain level generated by the colorization network. y_i is the ground-truth image of the same resolution as $\hat{y}_i$, which can be obtained by rescaling the ground-truth image of original resolution via bilinear interpolation. λ_i is the weight for each level of output, which are set to [1, 2, 3, 10] respectively where output images with higher resolutions have higher weights. $\varphi()$ is the output of the VGG16 network [29]. ω_1 is the weighting factor and set to 5 in all our experiments.

Multi-scale Adversarial Loss We further adopt a multi-scale adversarial loss [27] for our multi-scale discriminator. To further improve stability, we apply gradient penalty regularization [26] to the discriminator. Our multi-scale adversarial loss is defined as:

$$\begin{aligned}\mathcal{L}_{adv} =& \frac{1}{n}\sum_{i}^{n} \lambda_i E_{y_i}[\min(0, -D(y) - 1)] \\ &+ \frac{1}{n}\sum_{i}^{n} \lambda_i E_{\hat{y}_i}[\min(0, D(\hat{y}_i) - 1)] \\ &+ \omega_2 E_y[(\|\nabla D(y)\|^2]\end{aligned} \tag{2}$$

Here, D is the discriminator. $\min(\cdot)$ is the minimum operator. $E[\cdot]$ is the expectation operator. ω_2 is a weighting factor for gradient penalty and is set to 10 in all our experiments. The multi-scale adversarial loss increases the realness of the colorization results in terms of both global color composition and local color shading, as shown in Figure 3. It also widens the diversity of different types of style references.

The overall loss function L of our framework can be defined as the sum of the reconstruction loss $\mathcal{L}_{rec}$ and the multi-scale adversarial loss $\mathcal{L}_{adv}$:

$$\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{adv} \tag{3}$$

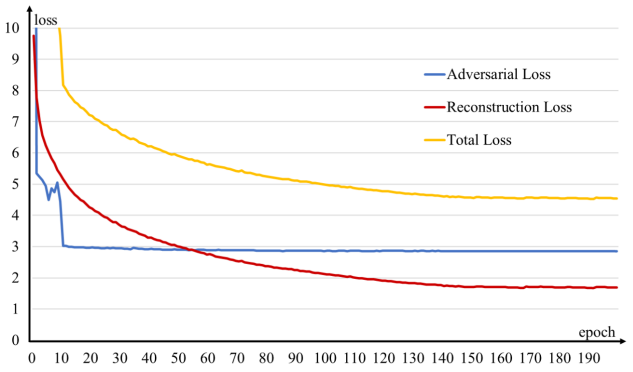


Fig. 4 Training loss curves for individual loss terms and the total loss.

3.3.3 Training Details

We use the Adam optimizer [20] to train our networks. All networks are jointly trained. The learning rate are initially set to $1e^{-4}$ and gradually decrease to $2e^{-6}$. we employ a learning rate adjustment policy where the initial learning rate is multiplied by $1-(\frac{iter}{max\_iter})^{power}$ with $power = 0.9$. The optimization converges after about 150 epochs. The training loss curve is shown in Fig. 4.

4 Results

In this section, we present an in-depth evaluation of our reference-guided sketch colorization framework. Firstly, we present visual comparisons between our method and several competitors in different categories to qualitatively evaluate the performance of colorization and style alignment of our framework. We further perform a quantitative comparison to the mainstream state-of-art colorization methods in terms of our sketch colorization task. Moreover, we present an ablation study to investigate the designs and contributions of each component in our framework.

We categorize our competitors into three major categories: image style transfer, style-content disentanglement, and conditional sketch colorization. We choose several state-of-the-art works in each category as the benchmark. For image style transfer, we choose three recent CNN-based methods, the Gatys method [6], WCT [24], and AdaIN [13] as our competitors. For style-content disentanglement, we compare with CCD [7] and DMIT [38]. Both methods are trained with our prepared cartoon dataset. For conditional sketch colorization, we choose two state-of-the-art reference-based colorization

methods [22, 40] and one state-of-the-art hint-based colorization method [41] as our competitors.

4.1 Qualitative Evaluations

Fig. 5 shows the visual comparisons between our method and the state-of-the-art image style transfer, style-content disentanglement, and reference-based sketch colorization methods. The Gatys' method [6] fails to colorize the sketch and only splash random color patterns. AdaIN [13] and WCT [24] achieve better results, but the structural lines are still not well preserved with obvious distortions and artifacts. The style-content disentanglement methods have better performance on preserving the structural lines of the sketch, but the color styles of the generated images are usually dissimilar with the reference cartoon. This is mainly because that they encode the content component via an implicit representation, which may lead to bias to the actual shape and introduce extra errors when encoding the color style. Similarly, the reference-based colorization method [22] fails to propagate the exact color from the reference during the colorization. In sharp contrast, our methods ensures the content component by faithfully preserving the structural lines in the sketch, so that both the style and the content are represented in our colorization framework with less bias.

Fig. 6 shows the visual comparisons between our method and the state-of-the-art hint-based sketch colorization methods. As shown in Fig. 6(c), the results of reference-based colorization [40] contain unexpected color mixing and obvious color discontinuity. Moreover, the results can not fully reflect the color characteristics of the reference image. The hint-based colorization method [41] acquires can also produce a style similar to the reference image with a certain amount of user hints (at least 25 color points), as shown in Fig.6(d)&(e). But the quality of the output highly depends on the user's experiences of color composition and expects extensive color indications. The colorization results often deviate from the reference image and the overall coloring procedure may not be convenient for amateur users. In comparison, our method faithfully propagates the reference color styles to the sketches with much less user efforts.

Besides, we investigate the possibilities to apply reference-based colorization methods in other domains to our sketch colorization task. Icon colorization [31] and photo colorization [39] are tested. A visual comparison is presented in fig. 7. We can observe that [31] somehow learns to propagate the colors from the references, but the output colors are too saturated for real-life sketch colorization purpose. [39] fails to obtain semantic correspondence in our sketch colorization case and thus outputs very dull colorization results. We find that the domain gap is still too large to directly adapt photo and icon colorization techniques to cartoon sketches.

Tab. 1 Statistics of quantitative evaluations.

Method	PSNR	SSIM	FID
Gatys	13.53	0.54	187.56
WCT	15.41	0.59	161.93
AdaIN	14.07	0.64	81.56
DMIT	12.33	0.69	46.95
CCD	13.61	0.69	104.85
Lee's method	16.00	0.76	50.12
Ours	**17.25**	**0.74**	**27.99**

4.2 Quantitative Evaluations

We also present a quantitative evaluation based on the quality of reconstruction. We randomly sample pairs of ground truth cartoon and sketch pairs from the evaluation dataset and use the color cartoon as the style guidance to colorize the sketch. Ideally, the colorized output image should be exactly the same with the color cartoon. To estimate the reconstruction quality, we measure the similarity between the reconstructed color image and the ground-truth color image with two commonly used similarity metrics, Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) [35]. The statistics is shown in the "PSNR" and "SSIM" columns in Tab. 1. As we can see in the table, our method outperforms all competitors in both metrics in the perspective of reconstruction.

Furthermore, to estimate the realness of our colorized outputs, we also present a quantitative study by calculating the Fréchet Inception Distance (FID) [4] score between our colorization results and ground truth color cartoons. In this case, the style reference and the input sketch do not need to be the same. The statistics is shown in the "FID" column of Tab.1. We can see that our method produces the closest colorization results to the ground truths, which is another proof of the superiority of our method in terms of output realness.

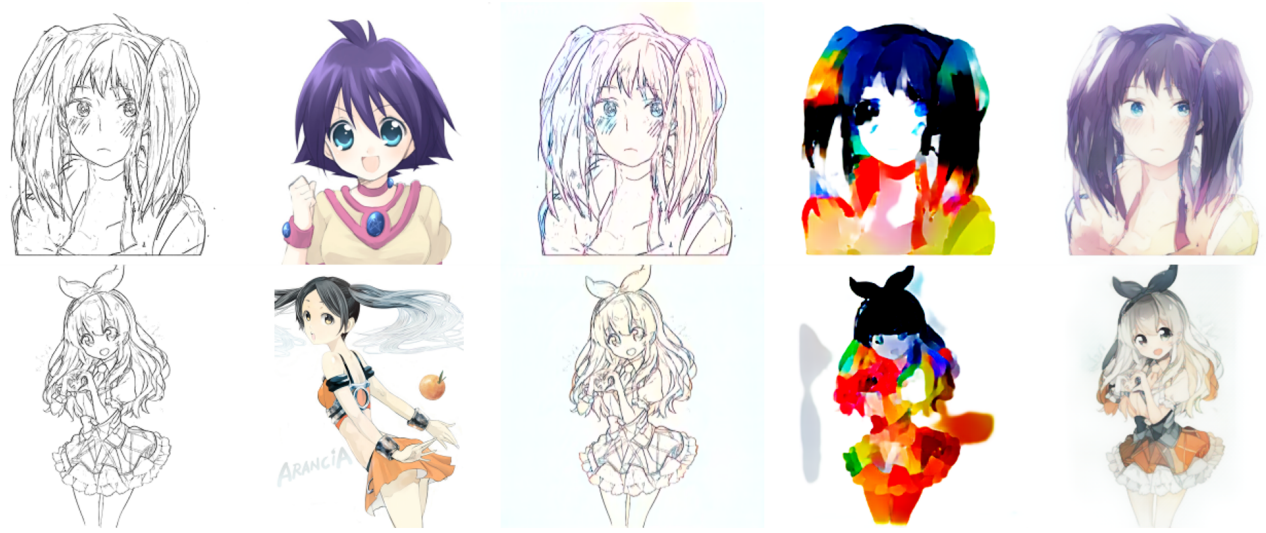
4.3 Ablation Study

To validate the impact of each component in our network design, we perform an ablation study by estimating the reconstruction metrics and the FID metrics with different network designs. The statistics is

Fig. 5 Visual comparisons between our method and the state-of-the art image style transfer methods, style-content disentanglement methods, and conditional sketch colorization.

(a) Input (b) Reference (c) Zheng et al. [40] (d) Zheng et al. [41] (without hint) (e) Zheng et al. [41] (with hint) (f) Ours

Fig. 6 Comparisons between our method and the state-of-the-art conditional sketch colorization methods.



(a) Input (b) Reference (c) Zhang et al. [39] (d) Sun et al. [31] (e) Ours

Fig. 7 Comparisons our method with method of icon colorization and photo colorization.

shown in table 2. Note that the PSNR/SSIM metrics are evaluated with paired sketch/reference input and the FID is evaluated with random references.

Firstly, without the multi-scale output design, both PSNR and SSIM values are significantly dropped, which shows the effectiveness of our multi-scale output design. We also study the use of the multi-scale discriminator by only feeding a single full-resolution output to the discriminator. The statistics show that the multi-scale discriminator design is essential to the output quality. The multi-scale discriminator design ensembles the adversarial learning in different scales and can be seen as a general case of [15], which fuses global and local information together to improve the generation quality. With the multi-scale network design, the colorization can take care of both global color composition and the local texture synthesis. In addition, we find that the multi-scale reconstruction loss is also very important for generating visually pleasant results. We also test to replace the adaptive instance normalization with an alternative style injection approach where the style codes of the same spatial size are reshaped as feature maps and concatenated together. However, as shown in the second row, both measurements dropped a lot with the concatenation design. We also investigate the appropriate dimension of the color style code to encode the color style. Higher-dimensional encoding has a larger capacity but may encode some extra information that may not be directly related to styles. Also, using higher dimensions may reduce the generalization ability in the style encoding. On the opposite side, lower-dimensional encoding has better generalization but may lead to a shift of color style in the reconstructed image. In our experiments, we find 256-D is optimal for the color style code, which best balances between reconstruction and generalization.

We have also explored the effectiveness of the discriminator by a visual ablation, as shown in Fig. 8. Without the discriminator, the diversity of the colors in the output image might be restricted, and the color styles might be overfitted. For example, in the first row, without adversarial learning, the colorization of the girl's eyes lose symmetry, which is considered unnatural. The same issue is also observed in the last row, where the network without adversarial learning may falsely propagate the white colors to the top-right of the canvas due to similar modes in the style reference. The adversarial learning minimizes the impacts of these style-unrelated modes and ensures natural colorization in these cases. On the other hand, adversarial learning also effectively alleviates the problem of color-bleeding, such as the girl's chest in the second row and the girl's arms and hair in the third row.

5 Limitation

Although our method can be applied to most sketch images, our output results may still be affected by the color and texture composition of the style reference. If the style reference contains very few color information, the output quality may not be satisfying. Additionally, if the style reference and the input sketch are too

Tab. 2 Statistics of our ablation study.

Method	PSNR	SSIM	FID
Without multi-scale outputs	18.89	0.81	33.50
With multi-scale discriminator	19.99	0.84	36.80
With multi-scale reconstruction	22.01	0.84	32.35
AdaIn→feature reshape & concat	20.37	0.68	66.48
Full method with 128D style	21.88	0.84	30.38
Full method with 512D style	19.52	0.72	39.82
Our full method (256D style)	**22.43**	**0.85**	**27.99**

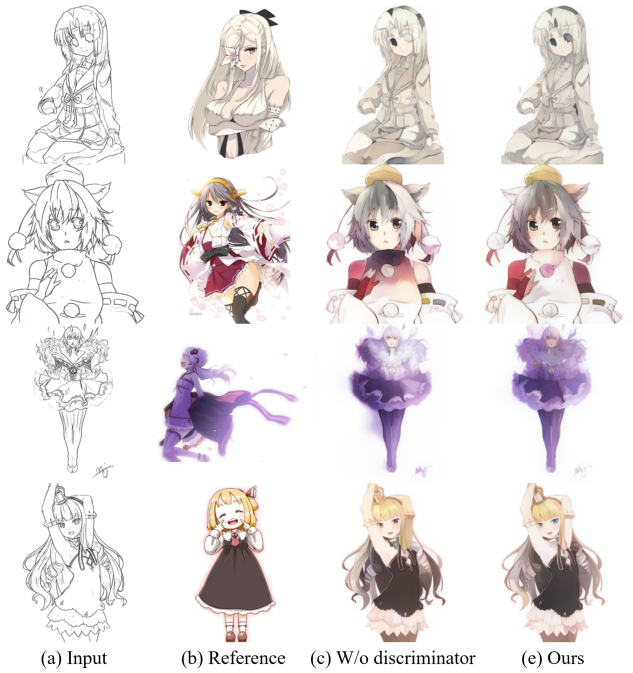


Fig. 8 Comparisons between with and without the discriminator.

different in structure composition, our method may find difficulties in propagating the color styles from the color regions of the reference style to the input sketch, as shown in Fig. 9.

6 Conclusion

In this paper, we proposed a novel deep learning approach for the reference-guided cartoon sketch colorization task. Our system consists of a color style extractor that extracts the color style code from a color cartoon image, a colorization network that fuse a sketch with a color style code to generate a set of multi-scale color cartoon outputs, and a multi-scale discriminator that improves the realness of the generated color cartoon in terms of both global color composition and local color shading. Experiments show that our method

Fig. 9 Our method may not work well when the style reference contains very few color information or when the style reference and the input sketch are too different in structure composition.

significantly outperforms the the existing methods in preserving content and style consistency with the input and reference images respectively. As for the future work, we intend to explore the potential of colorizing an animation sequence by incorporating the temporal information.

Acknowledgements

This work was supported in part by grants from the CIHE Institutional Development Grant IDG200107, the National Natural Science Foundation of China under Grant 61973221, the Natural Science Foundation of Guangdong Province of China under Grant 2018A030313381 and Grant 2019A1515011165.

References

[1] https://www.kaggle.com/ktaebum/anime-sketch-colorization-pair.
[2] A. Bugeau, V.-T. Ta, and N. Papadakis. Variational exemplar-based image colorization. *IEEE Transactions on Image Processing*, 23(1):298–307, 2013.
[3] A. Y.-S. Chia, S. Zhuo, R. K. Gupta, Y.-W. Tai, S.-Y. Cho, P. Tan, and S. Lin. Semantic colorization with internet images. *ACM Transactions on Graphics (TOG)*, 30(6):1–8, 2011.
[4] D. Dowson and B. Landau. The fréchet distance between multivariate normal distributions. *Journal of multivariate analysis*, 12(3):450–455, 1982.
[5] W. Gao, Y. Li, Y. Yin, and M. Yang. Fast video

multi-style transfer. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 3211–3219, 2020.

[6] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016.

[7] A. Gonzalez-Garcia, J. van de Weijer, and Y. Bengio. Image-to-image translation for cross-domain disentanglement. In *Advances in Neural Information Processing Systems*, pages 1294–1305, 2018.

[8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative adversarial networks. *CoRR*, abs/1406.2661, 2014.

[9] R. K. Gupta, A. Y.-S. Chia, D. Rajan, E. S. Ng, and H. Zhiyong. Image colorization using similar images. In *Proceedings of the 20th ACM international conference on Multimedia*, pages 369–378, 2012.

[10] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778, 2016.

[11] M. He, D. Chen, J. Liao, P. V. Sander, and L. Yuan. Deep exemplar-based colorization. *ACM Transactions on Graphics (TOG)*, 37(4):1–16, 2018.

[12] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. Salesin. Image analogies. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, pages 327–340. ACM, 2001.

[13] X. Huang and S. J. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1510–1519, 2017.

[14] X. Huang, M. Liu, S. J. Belongie, and J. Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European Conference on Computer Vision*, volume 11207, pages 179–196, 2018.

[15] S. Iizuka, E. Simo-Serra, and H. Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics (ToG)*, 36(4):1–14, 2017.

[16] P. Isola, J. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5967–5976, 2017.

[17] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711, 2016.

[18] H. Kim, H. Y. Jhoo, E. Park, and S. Yoo. Tag2pix: Line art colorization using text tag with secat and changing loss. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 9055–9064, 2019.

[19] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim. Learning to discover cross-domain relations with generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70, pages 1857–1865, 2017.

[20] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In Y. Bengio and Y. LeCun, editors, *Proceedings of the 3rd International Conference on Learning Representations*, 2015.

[21] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, and M.-H. Yang. Diverse image-to-image translation via disentangled representations. In *Proceedings of the European conference on computer vision (ECCV)*, pages 35–51, 2018.

[22] J. Lee, E. Kim, Y. Lee, D. Kim, J. Chang, and J. Choo. Reference-based sketch image colorization using augmented-self reference and dense semantic correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5801–5810, 2020.

[23] X. Li, S. Liu, J. Kautz, and M. Yang. Learning linear transformations for fast arbitrary style transfer. *CoRR*, abs/1808.04537, 2018.

[24] Y. Li, C. Fang, J. Yang, Z. Wang, X. Lu, and M. Yang. Universal style transfer via feature transforms. In *Advances in Neural Information Processing Systems*, pages 386–396, 2017.

[25] X. Liu, L. Wan, Y. Qu, T.-T. Wong, S. Lin, C.-S. Leung, and P.-A. Heng. Intrinsic colorization. In *ACM SIGGRAPH Asia 2008 papers*, pages 1–9. 2008.

[26] L. M. Mescheder, A. Geiger, and S. Nowozin. Which training methods for gans do actually converge? In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 3478–3487, 2018.

[27] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *Proceedings of the 6th International Conference on Learning Representations*, 2018.

[28] A. Sanakoyeu, D. Kotovenko, S. Lang, and B. Ommer. A style-aware content loss for real-time HD style transfer. In *Proceedings of the European Conference on Computer Vision*, volume 11212, pages 715–731, 2018.

[29] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[30] C. Song, Z. Wu, Y. Zhou, M. Gong, and H. Huang. Etnet: Error transition network for arbitrary style transfer. In *Advances in Neural Information Processing Systems*, pages 668–677, 2019.

[31] T.-H. Sun, C.-H. Lai, S.-K. Wong, and Y.-S. Wang. Adversarial colorization of icons based on contour and color conditions. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 683–691, 2019.

[32] Y.-W. Tai, J. Jia, and C.-K. Tang. Local color transfer via probabilistic segmentation by

expectation-maximization. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 747–754. IEEE, 2005.

[33] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[34] H. Wang, Y. Li, Y. Wang, H. Hu, and M. Yang. Collaborative distillation for ultra-resolution universal style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1857–1866, 2020.

[35] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4):600–612, 2004.

[36] T. Welsh, M. Ashikhmin, and K. Mueller. Transferring color to greyscale images. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 277–280, 2002.

[37] T. Yonetsuji. Paints chainer. *github.com/pfnet/Paintschainer*, 2017.

[38] X. Yu, Y. Chen, S. Liu, T. H. Li, and G. Li. Multi-mapping image-to-image translation via learning disentanglement. In *Advances in Neural Information Processing Systems*, pages 2990–2999, 2019.

[39] B. Zhang, M. He, J. Liao, P. V. Sander, L. Yuan, A. Bermak, and D. Chen. Deep exemplar-based video colorization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8052–8061, 2019.

[40] L. Zhang, Y. Ji, X. Lin, and C. Liu. Style transfer for anime sketches with enhanced residual u-net and auxiliary classifier gan. In *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 506–511. IEEE, 2017.

[41] L. Zhang, C. Li, T.-T. Wong, Y. Ji, and C. Liu. Two-stage sketch colorization. *ACM Transactions on Graphics (TOG)*, 37(6):1–14, 2018.

[42] R. Zhang, P. Isola, and A. A. Efros. Colorful image colorization. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III*, volume 9907, pages 649–666, 2016.

[43] Y. Zhang, C. Fang, Y. Wang, Z. Wang, Z. Lin, Y. Fu, and J. Yang. Multimodal style transfer via graph cuts. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5942–5950, 2019.

[44] J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2242–2251, 2017.

[45] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman. Toward multimodal image-to-image translation. In *Advances in neural information processing systems*, pages 465–476, 2017.

Xueting Liu received her B.Eng. degree in Computer Science and Technology from Tsinghua University and Ph.D. degree in Computer Science and Engineering from The Chinese University of Hong Kong in 2009 and 2014 respectively. She is currently an Assistant Professor in the School of Computing and Information Sciences, Caritas Institute of Higher Education. Her research interests include computational art, intelligent art, computer vision, and computer graphics.

Wenliang Wu received his B.Sc. degree from Guangdong Ocean University of Science and Technology in 2019. He is currently a graduate student in the College of Computer Science and Software Engineering, Shenzhen University. His research interests include computer vision and computer graphics.

Chengze Li received their B.Eng. degree from University of Science and Technology of China in 2013, and Ph.D. degree in Computer Science and Engineering from the Chinese University of Hong Kong in 2020. They is currently an Assistant Professor in the School of Computing and Information Sciences, Caritas Institute of Higher Education. Their research interests include 2D non-photorealistic media analysis and processing, computational photography, and computer graphics.

Yifan Li received his B.Sc. degree from Jiangxi University of Science and Technology in 2018 and is now a graduate student in the College of Computer Science and Software Engineering, Shenzhen University. His research interests include computer graphics, computer vision, machine learning and deep learning.

Huisi Wu received his B.E. and M.E. degrees both in Computer Science from the Xi'an Jiaotong University (XJTU) in 2004 and 2007, respectively. He obtained his Ph.D. degree in Computer Science from The Chinese University of Hong Kong (CUHK) in 2011. He is currently an Associate Professor in the College of Computer Science and Software Engineering, Shenzhen University. His research interests include in computer graphics, image processing, and medical imaging.