Photorealistic Image Style Transfer

Le Minh Viet

School of Electrical and Electronic Engineering

Associate Professor Yap Kim Hui

School of Electrical and Electronic Engineering

Abstract – Nowadays, social media is becoming more prevalent; people like to express themselves and share their opinions on the Internet. It is very common to use app to add effects into their photos, or even transfer them to a new style, to make them more personal and unique on social media. However, existing apps, such as Prisma, deepart.io, limit users to use only predefined styles. They are also not applicable for photo realism due to spatial distortion and unrealistic artifacts. This project aims to develop an application which allow user to do photo-realistic style transfer with any arbitrary style.

The first objective is to restudy Photorealistic Style Transfer via Wavelet Transforms method. Thanks to advantageous features of wavelets, the network can do stylization with arbitrary styles with minimal loss of structural information within short period of time.

The second objective is to develop style transfer application, which is applicable for photo-realism and support arbitrary styles. Different from existing stylization methods, this network is leveraged with the Wavelet Transform to achieve minimal loss of information, which is applicable for photo-realism. This network also utilizes progressive stylization, based on whitening and coloring transform (WCT), multiple times on the whole network to achieve better stylized result.

Keywords – Style Transfer, Photo-realism, Wavelet Transform, Computer vision, Deep Learning

1 Introduction

In the industry 4.0, using smart phone, accessing to social media is becoming prevalent worldwide. Social media users around the world is estimated to be nearly 3 billion; one of the most popular activities users do online is to share their photos. One study found that 60 percent of these uploaded photos are edited and applied filters. Users even transfer their photos to another artistic style to make it more unique and attractive. Prisma, an existing style transfer app, has more than 120 million users on IOS and Android. Thus, there is stunning demand for photo editing applications. However, these existing apps now are not applicable for photorealism and not having various styles for users to choose. To be photorealistic, a model should do stylization without generating too much the structural artifacts at output. To improve users experience, we re-study Photorealistic Style Transfer via Wavelet Transforms to build a stylization application, which allow users to define their own arbitrary pairs of content and style image for the stylization processing in a short time.

2 Literature review

2.1 Deep Photo Style Transfer

This method utilized the original work by Gatys et al. [6]. The original one shown that each layer of Convolutional Neural Network (CNN), a class of Deep Neural Network, contains multiple small computational units; from each unit, we can extract certain useful information about the image. Along the processing hierarchy of this network, lower layers capture the low-level content of input image, like exact pixels value, which can be used to capture content of input image. Meanwhile, higher layers capture the high-level information of input image, like textures, and object information, which can be used to capture style of input image. When both style and content image input to this model, both style and content information of 2 inputs are captured at the same time. Therefore, by combining together this information, this model can generate output, stylized image.

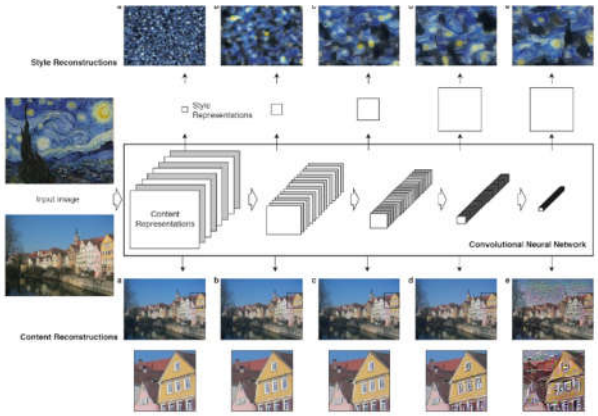

Figure 1: Model structure of Convolutional Neural Network (CNN) used in the paper : "A Neural Algorithm of Artistic Style"

Two loss functions defined by Gatys et al.

$$\mathcal{L}_c^\ell = \frac{1}{2N_\ell D_\ell} \sum_{ij} (F_\ell[O] - F_\ell[I])_{ij}^2$$
$$\mathcal{L}_s^\ell = \frac{1}{2N_\ell^2} \sum_{ij} (G_\ell[O] - G_\ell[S])_{ij}^2$$

,where F_l is result of activation function and G_l is Gram matrix at layer L of VGG network.

More than that, Deep Photo Style Transfer shown the techniques that can be applicable for photorealism. Their insight is that input is already photorealistic, so to generate realistic output, they try to preserve structure input image throughout the whole stylizing process. An affine function mapping the local input RGB values onto their output counterparts will help to preserve the input structural information during the whole process. Formally, they built based on the Matting Laplacian of Levin et al. [9]. Loss function of photorealism is defined as follow:

$$\mathcal{L}_m = \sum_{c=1}^{3} V_c[O]^T \mathcal{M}_I V_c[O]$$

(Input image *I, output O* with N pixels. M_i is $N \times N$, V_c [*O*] (N × 1) – vectorized version of O in channel *c.* Please refer to the original paper [5], [9] for detail.)

Stylization is done by minimizing loss function, which is sum of loss in content, style, and photorealism:

$$\mathcal{L}_{\text{total}} = \sum_{l=1}^{L} \alpha_\ell \mathcal{L}_c^\ell + \Gamma \sum_{\ell=1}^{L} \beta_\ell \mathcal{L}_{s+}^\ell + \lambda \mathcal{L}_m$$

Limitation:

Despite of applicability for photorealism, this approach is slow as it is image-based algorithm. Whenever users want to apply new style, they have to wait for the training time of the whole VGG network to iteratively minimize the cost function. The higher resolution the image has, the longer time to do the optimization. Moreover, hyperparameter λ has to be tuned for each pair of content and style image, which is very impractical for our application.

2.2 Whitening and Coloring Transform (WCT)

Since the two following papers are built on top of WCT algorithm, we first brief about WCT [2]. WCT is an arbitrary style transfer model-based algorithm. The goal of WCT is to transform the VGG feature maps of content image to match that of style image. It comprises of two steps: whitening and coloring transform.

To achieve better stylization result, in the original paper, Li et al. [2] applies WCT multiple times at each specific layer (figure 2). At each layer, before applying WCT, inputs are first passed into an encoder (weights are kept fixed as VGG-19 model). After applying WCT, results are passed into a decoder, which inverts features back to RGB spaces. Decoder is simply trained as symmetrical to that of VGG-19 network.

This stylization method is model-based algorithm. Unlike the Deep photo style transfer method [3], this one shifts the burden of time from inference (testing) to training time. As a result, model-based algorithm is suitable for our application.

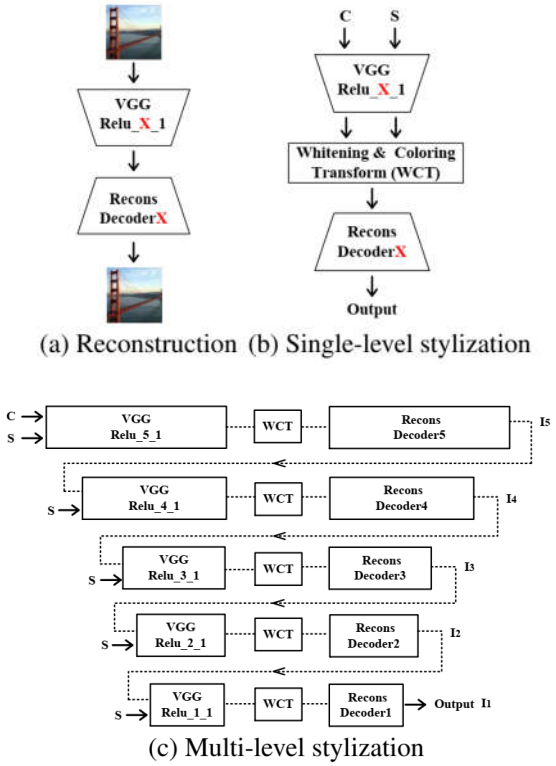

Figure 2: Model structure used in the paper [2].

Limitation:

The WCT generates good output in terms of applied style. Yet, the output contains many unrealistic features, like object distortion, which is inapplicable for photorealism.

2.2.1 A Closed-form Solution to Photorealistic Image Stylization

Based on the idea of WCT algorithm, Li et al. [5] proposed an improved method, which is applicable for photorealism. This method consists of 2 steps: Stylization (F_1) and Photorealistic Smoothing (F_2).

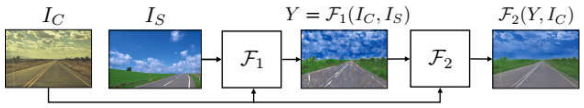

Figure 3: Steps used in the paper [4]

At stylization step, motivated by WCT method, Li et al. proposed PhotoWCT model. Observed that upsampling in the decoder fails to invert back to the original structures of the input, the unpooling layers is used to replace in PhotoWCT model, as the unpooling one is successful in preserving spatial information.

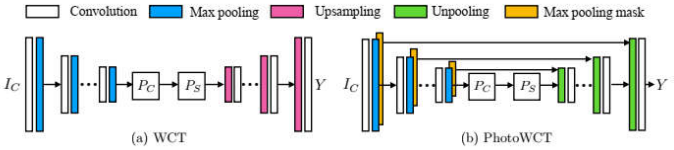

Figure 4 [4]: Comparison between WCT [2] and PhotoWCT [4]

The PhotoWCT-stylized result still contains structural artifacts. Authors propose a Photorealistic Smoothing as post-processing step to fix these artifacts. This step aims to achieve two goals. Firstly, similar stylization should be done on pixels with similar content in local neighborhood. Secondly, to maintain global stylization, the output should not deviate drastically from the PhotoWCT result.

Authors used affinity matrix to describe the pixel similarities.

$$W = \{w_{ij}\} \in \mathbb{R}^{N \times N}$$

(N is the number of pixels)

Authors achieve these two goals by optimizing the following:

$$\underset{r}{\operatorname{argmin}} \frac{1}{2}\left(\sum_{i,j=1}^{N} w_{ij}\left\|\frac{r_i}{\sqrt{d_{ii}}} - \frac{r_j}{\sqrt{d_{jj}}}\right\|^2 + \lambda \sum_{i=1}^{N} \|r_i - y_i\|^2\right),$$

y_i :pixel value of the stylized image Y, after F_1.

r_i :pixel value of the smoothed image R, after F_2.

This model supports arbitrary styles and produces good results, which is applicable for photorealistic. Yet, the performance of this model come mostly from the post-processing step, not from the PhotoWCT stylization step.

Limitation:

Although this method is model-based algorithm, the post-processing step takes long time and high memory usage. Especially for higher-resolution images, post-processing step might lead to out of memory usage on GPU. Moreover, the hyper-parameters are also manually tuned.

2.2.2 Photorealistic Style Transfer via Wavelet Transforms

Yoo et al. [1] proposed "wavelet corrected transfer" as another solution to fix the artifacts of WCT algorithm. Authors substituted the pooling and unpooling operations with the wavelet pooling and unpooling in the encoder and decoder. Utilizing WCT model, authors proposed WCT2 as an improved model, which can successfully reconstruct the images without any fixing artifacts steps used in the PhotoWCT one.

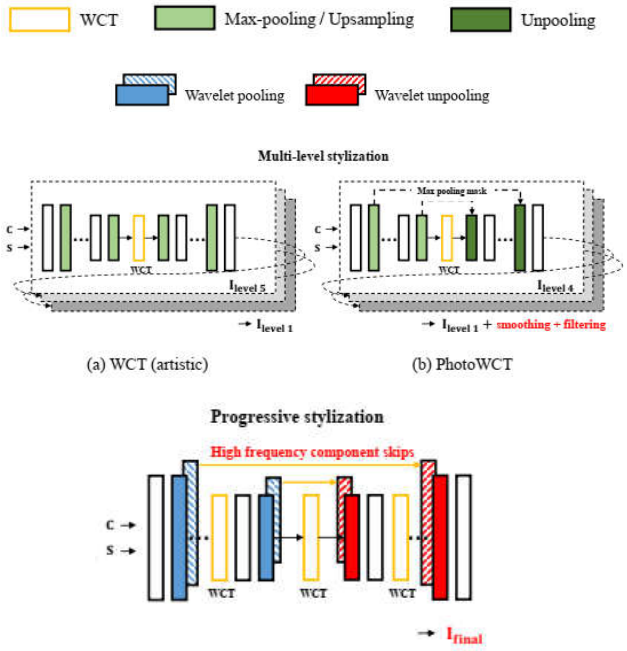

Figure 5 [1]: Comparison between WCT [2], PhotoWCT [4], and WCT^2 [1].

Additionally, authors proposed a progressive stylization (refer to figure 6) instead of the multi-level one. This enables the model to be more simple, efficient, and less artifacts. While multi-level strategy requires multiple pairs of encoder/ decoder for layers without sharing parameters during training and inference time, progressive stylization only requires one. As a result, the progressive one reduces parameters in both training and inference time. Moreover, when applying multi-level strategy, encoding/ decoding the signal recursively generates more artifacts.

In details, the "wavelet corrected transfer" is based on Haar wavelets so called wavelet pooling and unpooling in [1]. It has four kernels {LL^T, LH^T, HL^T, HH^T}. For simplicity, authors denoted them as LL, LH, HL, and HH, respectively in the figure 6. The low-pass filter (L) captures style information, like texture, color. Meanwhile, the high-pass one (H) contains structural information like, vertical, horizontal edges. As a result, the model can actively control the applied effects on each specific part of images.

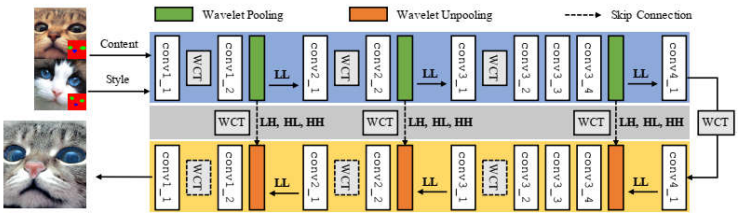

Figure 6 : Network used in the original paper [1]

Network architecture:

- The encoder (blue region) is based on pre-trained Image-Net VGG-19 from layer conv1_1 to conv4_1.
- The trained decoder (yellow region) has a symmetrical structure of the encoder.
- Progressive stylization based on WCT. Stylization at decoder is optional. It creates more vivid result; yet, style effects come with the cost of structural artifacts.
- Only the low component passes to the next layer; the high are directly skipped to the corresponding layer of the decoder. All the components are aggregated at the decoder.

Advantages:

This model allows us to do style transfer with arbitrary styles with minimum loss of information thanks to favorable properties of wavelets. Even with high resolution image, because of the simplicity in stylization steps, this model still performances well with satisfactory results.

Yet, like all of the mentioned above papers, this model still depends on semantic segmentation mask during stylization step.

3 Experiments

In order to achieve all the functions of out style transfer application, we choose "Photorealistic Style Transfer via Wavelet Transforms" method, as this enables us to do photorealistic stylization with

arbitrary styles. This model also performs well and quickly even with the high-resolution images.

Implementation details

We run the pre-trained network WCT2 [1] as the stylization model. We do not do stylization at decoder as it will make the result look unrealistic. For unpooling, we use concatenation version of WCT2 as it reconstructs the image better and clearer than summation version.

Like any stylization models, this model also realizes a lot on semantic segmentation maps. We utilize the semantic segmentation pretrained network provided from CSAILVision/semantic-segmentation-pytorch.

Content Style Result

Figure 7: Results generated of our application.

Using Flask framework to deploy all of these models as a web application, we provide users an end-to-end application, photorealistic style transfer. My code are available at leminhviett/Photo-realistic-Neural-style-transfer.

GPU used is Nvidia GTX 1050; the stylization process takes around 65 seconds for image size of 512 x 512. The whole process including segmentation and stylization takes around 90 seconds.

4 Conclusion

In conclusion, we have accomplished two objectives of this project by implementing and analyzing the Style Transfer technique via Wavelet Transforms. Regarding the comparison with other methods and the results generated, we can understand that this technique currently is the most efficient and suitable for this project. This model enables users to generate stylized results quickly and photo-realistically. Yet, like any others model which is used for photorealism, this one also relies heavily on semantic segmentation maps. Future works can be mitigating this dependency.

ACKNOWLEDGMENT

This project is highly supported by my URECA project supervisor, Associate Professor Yap Kim Hui. It must be acknowledged that some of my implementation code is based on [1]'s implementation.

We wish to acknowledge the funding support for this project from Nanyang Technological University under the Undergraduate Research Experience on Campus (URECA) programme.

REFERENCES

[1] Jaejun Yoo, Youngjung Uh, Sanghyuk Chun, Byeongkyu Kang, and Jung-Woo Ha. "Photorealistic Style Transfer via Wavelet Transforms", ICCV 2019

[2] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. "Universal style transfer via feature transforms". *In Advances in Neural Information Processing Systems, pages 386–396*, 2017

[3] F. Luan, S. Paris, E. Shechtman, and K. Bala, "Deep photo style transfer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017, pp. 6997–7005

[4] Y. Li, M.-Y. Liu, X. Li, M.-H. Yang, and J. Kautz, "A closed-form solution to photorealistic image stylization," in *European Conference on Computer Vision*, 2018.

[5] A. Levin, D. Lischinski, and Y. Weiss. "A closed-form solution to natural image matting". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):228–242, 2008

[6] Leon A. Gatys, Alexander S. Ecker and Matthias Bethge, "A Neural Algorithm of Artistic Style", Aug 2015.

[7] Statista, “Number of social media users worldwide from 2010 to 2021 (in billions),” July 2017; https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/.

[8] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso and A. Torralba. “Semantic Understanding of Scenes through ADE20K Dataset*,” in International Journal on Computer Vision (IJCV),* 2018.

[9] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso and A. Torralba. “Scene Parsing through ADE20K Dataset, ” *in Computer Vision and Pattern Recognition (CVPR),* 2017.

[10] Jing, Yongcheng and Yang, Yezhou and Feng, Zunlei and Ye, Jingwen and Yu, Yizhou and Song, Mingli. “Neural Style Transfer: A Review,” *in IEEE Transactions on Visualization and Computer Graphics,* 2019.