2020 International Conference on Computing and Data Science (CDS)

CNN-Based Image Style Transfer and Its Applications

Nanhao Jin
Department of Computer Science
Beijing University of Posts
and Telecommunications
Beijing, China
Jnh981103@163.com

***Abstract*—Convolutional neural network is a variant of deep neural network. It is widely used to extract image features and used in image classification, image generation and style transfer. In the style transfer task, we can extract the content from one picture through the convolutional neural network, extract the style from another picture, and generate a new picture by the combination. In this paper, we show the general steps of image style transfer based on convolutional neural networks through a specific example, and discuss the future possible applications.**

Keywords—convolutional neural network, image style transfer, deep learning

I. INTRODUCTION

Image style transfer refers to the technique of using machine learning algorithms to learn the style of artistic paintings and applying this style to another painting. Taking painting as an example, there are many styles, such as abstract, realism, impressionism, modernism, futurism, etc. The types of paintings include oil paintings, ink paintings, prints, etc. For each specific genre, different painters have their own style of expression, which is very characteristic in the brushstrokes, tones, textures and other characteristics of the picture. For example, at the beginning of his career, Van Gogh used gray and dark colors to paint. After receiving the influence of Impressionism, the choice of painting colors became more exaggerated and unrestrained. His texture was rougher, and the brush strokes were more casual. Cézanne's style is also a representative of the post-Impressionism. He tried to generalize things into geometry to make the painted objects have a solid sense of existence.

The image style transfer technology can be traced back to the image texture generation technology before 2000. Researchers used complex mathematical models and formulas to summarize and generate textures, but manual modeling was time-consuming and labor-intensive, and computer computing power was not strong at that time. Therefore, the development of image style transfer technology is very slow. With the development of deep learning [1], neural networks have achieved the state-of-the-art results on more and more problems, including traffic prediction [2], stock market prediction [3], time series classification [4] and so on.

As a type of neural network, the convolutional neural network extracts the features in the image through operations such as convolution and pooling, and has achieved great success in the field of image processing [5-7]. The development of convolutional neural networks has also promoted research related to image style transfer. In 2015, Gatys et al. proposed an image style transfer algorithm based on convolutional neural networks. Gatys et al. found that the use of convolutional neural networks can separate the latent content feature representation of the image from the latent style feature representation, and independently process these high-level latent feature representations to effectively achieve the transfer of image style, and achieved an ideal result [8].

In simple terms, the general style of the transfer process is as follows. Given a picture as the original content and another picture as the style, the pictures are entered into a convolution neural network so that the final image fusion is used to combine the original content and style. Different features can be extracted in different layers of the convolutional network, so that the difference between the output of the generated image in these layers between the content picture and the style picture is as small as possible, that is, the cost function is minimized to obtain any desired picture. The deeper the network used is, the more complex the patterns are.

In this paper, we reviewed the work related to image style transfer, and based on the pre-trained VGG model, showed a method of image style transfer based on convolutional neural networks. We also discussed the application of image style transfer in various scenarios. Our discussion can help subsequent researchers to further explore the application of image migration.

The following arrangements of this article are as follows. In Section 2, we discuss related work. In Section 3, we describe the specific process of style transfer and compare the weights of different losses. In Section 4, we discuss possible application scenarios. In Section 5, we give conclusions.

II. RELATED WORK

In this section, we review the related work of style transfer based on convolutional neural networks and adversarial generative networks. Since the breakthrough in 2015, image style transfer technology has been rapidly developed, and a series of deep learning models including convolutional neural networks and adversarial generation networks have been applied.

A. CNN-based Image Style Transfer

Convolutional Neural Network (CNN) is a deep feed-forward artificial neural network. Its artificial neurons can respond to some of the surrounding units in the coverage area

978-1-7281-7106-7/20/$31.00 ©2020 IEEE
DOI 10.1109/CDS49703.2020.00081

with the mechanisms of convolutional layers, pooling layers, and fully connected layers. The basic structure of CNN includes two layers: feature extraction layer and feature mapping layer. Convolutional neural networks have unique advantages in image processing, image recognition, and image segmentation due to their special structures shared by local weights. Weight sharing reduces the complexity of the network, and images of multi-dimensional input vectors can be directly input into the network, avoiding complex data reconstruction.

Deep neural networks can separate and reorganize the content and style of any image, and provide a feasible algorithm for creating images. The most powerful one is the convolutional network. Each layer can extract specific features from the input image. Gatys et al. [8] found that the content and style of the picture can be separated and extracted in the convolutional network, which is also composed of two parts in the Loss function. The deeper the network, the more attention is paid to the image content, so the deep layer is selected to extract the image content. The style is represented by multiple scales, so that several layers in the network can be selected to define the style. In their work, 16 convolutional layers and 5 pooling layers in VGG-19 are used, instead of fully connected layers. Average pooling is used instead of maximum pooling. The content loss function describes the similarity between the generated picture and the original picture in content, and is defined as the L2 norm of the generated picture and the original picture. The style loss function describes the similarity between the style and the style picture of the generated picture, which is defined as the L2 norm of the gram matrix. The overall loss function is the sum of the products of the above two loss functions and the hyper parameters. The two hyper parameters determine the weights in the entire loss. An emphasize on the style will produce an image that matches the style image, effectively giving it texture and showing almost nothing of the original picture. When the content is emphasized, the original image content can be clearly recognized, but the style is not so matched. The output will get closer and closer to the final desired image.

Gatys et al. [9] proposed a method using a combination of content graphs and a trained forward network to convert a global optimal solution into a forward network, approaching the optimal solution to synthesize a network with a new texture. Johnson et al. [10] proposed to use the perceptual loss function instead of the pixel-by-pixel gap loss function to achieve real-time style conversion and image super-magnification reconstruction. Dumoulin et al. [11] proposed a way for N styles to share a model, alleviating the space consumption of model storage. With their work, real-time style interpolation can be applied to video. Luan et al. [12] strengthened and improved on the basis of Gatys' work and controlled the content details of style transfer. CNNMRF of Li et al. [13] is a combination algorithm of Markov Random Field (MRF) model and trained deep convolutional neural network (DCNN), which matches each input neural area with the most similar area in the style image, reducing inaccurate feature transfer and retaining the specific characteristics of the content image.

Huang et al. [14] accelerated the style transfer through the encoding-decoding structure. But due to the problem of decoder information loss in the self-encoder structure, the texture of the image migration result appeared "blocky" on the segmentation edge. Li et al. [15] added white migration (Whitening transform) and color migration (WCT) to Adaptive Instance Normalization (AdaIN). WCT makes style migration in feature selection and color to have some improvement. At CVPR 2019, Puy et al. [16] added runtime loss and regularization terms to AdaIN, making the network structure of style transfer more flexible. Park et al. [17] added low-level feature attention to AdaIN, that is, style attention, to solve the problem of consistency of texture distribution on the non-edge long-term domain. Yao et al. [18] added high-level feature attention to AdaIN, which is referred as the theme attention, to solve the problem of chaotic texture migration under multiple theme images.

B. *GAN-based Image Style Transfer*

Generative Adversarial Network (GAN) [19] contains two neural networks, a generator and a discriminator. Use the generator to generate the desired style of pictures, and then use the discriminator to determine whether the generated pictures are real. Through the mutual game between the two neural networks, the effect of "falsification" is achieved under equilibrium.

Zhu et al. [20] proposed CycleGAN, an unsupervised style conversion framework, which introduced the theory of cyclic consistency, using two sets of generators and discriminators to sequentially map an image in the source domain through the source domain to the target domain. After the mapping to the source domain, the reconstructed image of the source image is obtained, and the reconstructed image with the L1 norm constraint is as close as possible to the source image. Yi et al. [21] proposed DualGAN for unsupervised image style conversion based on dual learning and L1 norm. Xie et al. [22] proposed a method to generate high and low exposure images based on CycleGAN, and combined with the input image to achieve HDR style transfer. Lin et al. [23] proposed a gray-scale image colorization model based on pixel-level GAN. The objective function uses L1 norm as the colorization optimization item.

Compared with CNN, GAN has a more powerful ability to generate images with different styles, instead of the fixed style from the original image. However, GAN has a much higher requirement for time consumption of training. Besides, GAN models can be unstable in the training process and generate images that are far from expectation.

III. CNN-BASED IMAGE STYLE TRANSFER

In this section, we use a concrete example to demonstrate the style transfer process based on convolutional neural networks.

A. *Image Preparation*

We need two input images, one is a content image and the other is a style image. We will use a convolutional neural network to modify the content image so that it is close to the style image in style. The content image is shown in Figure 1, which is a landscape shot at Cannes, France. The style image is shown in Figure 2, which is a famous painting of Vincent Willem van Gogh. The painting of van Gogh is famous for its own style, that’s why we choose it as the style image.

Fig. 1. The content image.

Fig. 2. The style image.

B. Image Preprocessing and Postprocessing

We will standardize the pictures and conduct other preprocessing steps, so that the input pictures meet the input format requirements of the model. The preprocessing function normalizes the input image in the three channels of RGB, and transforms the result into the input format accepted by the convolutional neural network. The postprocessing function restores the pixel values in the output image back to the values before normalization. Since the image printing function requires that the floating-point value of each pixel is between 0 and 1, we use the clip function to take values 0 and 1 for values less than 0 and greater than 1, respectively.

C. Model Preparation

In this part, we use the pre-trained VGG-19 model based on the famous ImageNet dataset to extract image features. Through training on a large number of pictures, this VGG-19 model already has the ability to extract picture styles. Even though the model has 19 layers, we only keep and use some hidden layers instead of the whole network.

D. Loss Function Definition

In the style transfer task, loss function is the main means to adjust the final transfer effect. Through the weighted sum of the three types of Loss and the control of the weight coefficients, we can adjust the final output image effect. That is, by adjusting these weight hyper parameters, we can change the relative importance of retention content, migration style, and noise reduction in the composite image.

The three loss functions we use are as follows:

1) Content Loss: Content loss measures the difference in content characteristics between the synthesized image and the content image through the squared error function.

2) Style Loss: Style loss also measures the difference in style between the synthesized image and the style image through the square error function. In order to express the style output by the style layer, we first calculate the output of the style layer through the feature extraction function. We use the Gram matrix to express the style output by the style layer. It should be noted that when the image is large, the elements in the Gram matrix are prone to have large values. In addition, the height and width of the Gram matrix are the number of channels. In order to make the style loss not affected by the size of these values, the gram matrix is divided by the size of the image. The two Gram matrix inputs of the square error function of pattern loss are respectively based on the pattern layer output of the composite image and the pattern image. It is assumed here that the Gra3m matrix of the style image has been calculated in advance.

3) Total Variation Loss: Sometimes, there are a lot of high-frequency noise in the synthesized image we learned, that is, there are particularly bright or dark particles. A commonly used noise reduction method is the total variation denoising. Suppose $x_{i,j}$ represents the pixel value with the coordinate of (i,j), reducing the total variance loss $sum_{i,j}\left|x_{i,j}-x_{i+1,j}\right|+\left|x_{i,j}-x_{i,j+1}\right|$ can make adjacent pixel values as similar as possible.

E. New Image Generation

Fig. 3. The output image when the resolution is set to 150×225.

Fig. 4. The output image when the resolution is set to 300×450.

Next, we can define a new composite image and treat the composite image as a model parameter. When training the model, we continuously extract the content features and style features of the synthesized image, and calculate the loss function. During the training process, we constantly update the synthesized new images to make the loss function smaller. We first set the weights of three different losses to 1, 1000 and 10. The synthesized image is trained for 500 epochs with a learning rate of 0.1. Figures 3 and 4 show the results when the resolution of the generated image is set to 150 × 225 and 300 × 450, respectively. The composite image of Figure 5 retains more details because of its larger size.

IV. Applications of Style Transfer

In this part, we discuss some potential applications of style transfer.

The first application is the mobile APPs with photo filters. Nowadays, the photo filter function in mobile phones can only simply change the color tone of the original photo (black and white, dark color, bright color, etc.) or perform simple spatial transformation of the content (perspective, fisheye, symmetry, etc.), but not the photo texture. Style transfer allows us to add some photo styles to the phone, users can transform the original photos into the selected style and even choose the degree of change in the generated pictures.

Style transfer can also be used in movies. The full movie Loving Vincent, released in 2017, uses characters from Van Gogh's paintings, and the style of the entire film adopts Van Gogh's painting style. For this reason, 125 artists spent nearly six years painting and painting each frame to make the film coherent and shocking. If you use style transfer to build props for framing and transfer Van Gogh's style to the image, you will definitely save a lot of time. In the film and television works, there will be a style change before and after the same picture, for example, the scene of the restoration of the planet before the destruction in the Revengers. It takes a long time to render, which consumes time and money. After using style transfer to obtain special effects that are close to the target, careful modification can save part of the cost.

Furthermore, style transfer can be used for entertainment. The facial features of the couple are extracted, and the predicted face of the child is generated by GAN. But in fact, most of the characteristics are determined by genetic factors, and it will not happen that the mother's eyes are very big, the father's eyes are very small, and the child's eyes are averaged, so it is only for entertainment.

Last but not the least, style transfer has the potential of being used for historical document repair. For some calligraphy, painting monuments, and old photos, some of them may be incomplete due to age. You can take a complete object and extract the style of the work to be restored to achieve restoration.

V. Conclusion

In this paper, we use a real-world landscape photo and a pre-trained VGG-19 model to show the process of style transfer. Our results demonstrate both the simplicity and feasibility of style transfer in a series of applications. Our discussion can help subsequent researchers to further explore the application of image style transfer in various scenarios.

References

[1] LeCun Y, Bengio Y, Hinton G. Deep learning[J]. nature, 2015, 521(7553): 436-444.

[2] Jiang W, Zhang L. Geospatial data to images: A deep-learning framework for traffic forecasting[J]. Tsinghua Science and Technology, 2018, 24(1): 52-64.

[3] Jiang W. Applications of deep learning in stock market prediction: recent progress[J]. arXiv preprint arXiv:2003.01859, 2020.

[4] Jiang W. Time series classification: nearest neighbor versus deep learning models[J]. SN Applied Sciences, 2020, 2(4): 1-17.

[5] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]//Advances in neural information processing systems. 2012: 1097-1105.

[6] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[7] Jiang W, Zhang L. Edge-siamnet and edge-triplenet: New deep learning models for handwritten numeral recognition[J]. IEICE Transactions on Information and Systems, 2020, 103(3): 720-723.

[8] Gatys L A, Ecker A S, Bethge M. Image style transfer using convolutional neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2414-2423.

[9] Gatys L A, Ecker A S, Bethge M. Texture and art with deep neural networks[J]. Current opinion in neurobiology, 2017, 46: 178-186.

[10] Johnson J, Alahi A, Fei-Fei L. Perceptual losses for real-time style transfer and super-resolution[C]//European conference on computer vision. Springer, Cham, 2016: 694-711.

[11] Dumoulin V, Shlens J, Kudlur M. A learned representation for artistic style[J]. arXiv preprint arXiv:1610.07629, 2016.

[12] Luan F, Paris S, Shechtman E, et al. Deep photo style transfer[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 4990-4998.

[13] Li C, Wand M. Combining markov random fields and convolutional neural networks for image synthesis[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2479-2486.

[14] Huang X, Belongie S. Arbitrary style transfer in real-time with adaptive instance normalization[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 1501-1510.

[15] Li Y, Fang C, Yang J, et al. Universal style transfer via feature transforms[C]//Advances in neural information processing systems. 2017: 386-396.

[16] Puy G, Pérez P. A Flexible Convolutional Solver for Fast Style Transfers[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 8963-8972.

[17] Park D Y, Lee K H. Arbitrary Style Transfer With Style-Attentional Networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 5880-5888.

[18] Yao Y, Ren J, Xie X, et al. Attention-aware multi-stroke style transfer[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 1467-1475.

[19] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[C]//Advances in neural information processing systems. 2014: 2672-2680.

[20] Zhu J Y, Park T, Isola P, et al. Unpaired image-to-image translation using cycle-consistent adversarial networks[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2223-2232.

[21] Yi Z, Zhang H, Tan P, et al. Dualgan: Unsupervised dual learning for image-to-image translation[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2849-2857.

[22] Xie Z, Ye G, Yan S, et al. HDR image style transfer technique based on generative adversarial networks[J]// Journal of Shanghai University (natural science). 2018, 024(004):524-534.

[23] Lin J, Zhuge J, Zhang Q. Colorization of Complex Scene Grayscale Images with Pixel-Wise Generative Adversarial Networks[J]// Journal of Computer-Aided Design & Computer Graphics. 2019, 31(03):93-100.