Neural Stylization of Traditional Asian Paintings on Duke University Buildings

Aaron Liu
Master in Interdisciplinary Data Science
Duke University
tl254@duke.edu

Christy Hu
Master in Interdisciplinary Data Science
Duke University
dh275@duke.edu

Abstract

Through disassociating and recombining content and style images, Gatys et al.[1] showed the capacity of VGG-19 [2] in generating high-quality artistic imagery. Such system of utilizing Deep Neural Networks (DNNs) for image transformation is widely known as Neural Style Transfer (NST). Since then, NST has become increasingly popular in both academic research and industrial applications. In this project, we further explore NST performance on different architectures, including VGG-11, VGG-19, VGG-11 and VGG-19 [2] with batch-normalization (BN) layers [3], and ResNet-18 [4]. Our results indicate that smaller model such as VGG-11 can yield comparable results with VGG-19. In addition, it seems ineffective to perform NST with ResNet architectures.

1 Background

1.1 VGG

The VGG architecture developed by Simonyan and Zisserman (2014) replaced large kernel-sized convolution filters in previous systems with a stack of 3×3 smaller filters, and demonstrated that concurrent placement of small-size filters can elicit the same effect of larger-size filters, adding the benefits of reduced computational complexity. The paper also shows that with increasing depth, the architecture presents significant improvement on localization and classification tasks.

1.2 Batch normalization

Loffe and Szegedy (2015) introduced batch normalization as an effective training technique to combat the internal covariate shift problem by standardizing the inputs to a layer for each mini-batch. It is believed that when incorporated into the model design, batch normalization stabilizes the learning process, making deep network training less sensitive to weight initialization with significantly reduced number of training epochs required.

1.3 ResNet

As network gets deeper and harder to train because of the vanishing gradients problem, He et al. proposed a residual learning framework called ResNet in 2015 that adopts a sequential approach using skip connections and batch normalization to train very deep neural networks efficiently with far less parameters and not hurting the performance.

ECE 590-07 (2021 Spring) final report.

1.4 Neural style transfer

Inspired by the remarkable success achieved in object and face recognition tasks using DNNs, Gatys et al. in 2015 proposed a way to generate high-quality artistic images by disassociating and recombing contents and styles of arbitrary images using VGG19. The study paves the way for a more algorithmic view of how people construct and interpret artistic imagery.

In this project, we aim to experiment NST with another variant of the VGG architecture, namely VGG-11, the smallest VGG model with only 11 layers and approximately 132M parameters to see if comparable results can be achieved by using a smaller model in the VGG family. In addition, we want to know if it is helpful to incorporate batch normalization when performing NST on VGG-based models. Finally, we select a member of the ResNet family, ResNet-18, which has comparably similar depth to VGG-19 to compare results. We want to find out if good performance of ResNet on visual recognition tasks will translate to similar superior performance on NST tasks.

2 Method

Our code implementation can be found on https://github.com/tienyuliu/ECE.590.07.Sp21-NST

2.1 Style model architecture

The models used for our experiments are based on VGG-11, VGG-19, VGG-11 and VGG-19 with BN layers, and ResNet-18. These models are pre-trained on ImageNet [5], and available on https://pytorch.org/vision/stable/models.html. Regarding the construction of the style models, we follow the setting in the original work by Gatys et al. [1]. The content reconstruction is obtained with the 'conv4_1' in the VGGs and ResNet-18. For style reconstructions, these are generated by matching the 'conv1_1', 'conv2_1', 'conv3_1', 'conv4_1' and 'conv5_1'. Because the objective of our study is to compare NST performance on different model architectures, we believe it is reasonable to use a similar setting proposed by the original author across all architectures to obtain comparable results.

2.2 Data and preprocessing

The dataset used in this project encompasses 12 content images (Duke University Buildings) and 7 style images (Asian paintings) with different image complexity. Specifically, we chose content images displaying buildings of varying sizes, colors, textures, taken from different angles and distance, have a different blend of objects and background scenery. In addition, the 7 style images are most representative of traditional Chinese paintings created in different periods by artists of a certain genre. Therefore, these images come in with drastically different styles, composition of visual elements, color and clarity, dominance and balance. We believe a combination of the above can provide us with a relatively fair view in comparing models and interpreting results. All the images are resized and cropped into 512×512. After that, they are normalized before being passed into our networks as they are pre-trained on ImageNet.

2.3 Training objective and optimization

The initialization uses the content images. Regarding optimization, we adopt Limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) to minimize the loss used in [1] shown below with $\alpha = 1$ and $\beta = 10^6$ for 500 steps to achieve the stylization.

$$\mathcal{L}_{total}(\vec{p}, \vec{a}, \vec{x}) = \alpha \mathcal{L}_{content}(\vec{p}, \vec{x}) + \beta \mathcal{L}_{style}(\vec{a}, \vec{x})$$

3 Experiment results

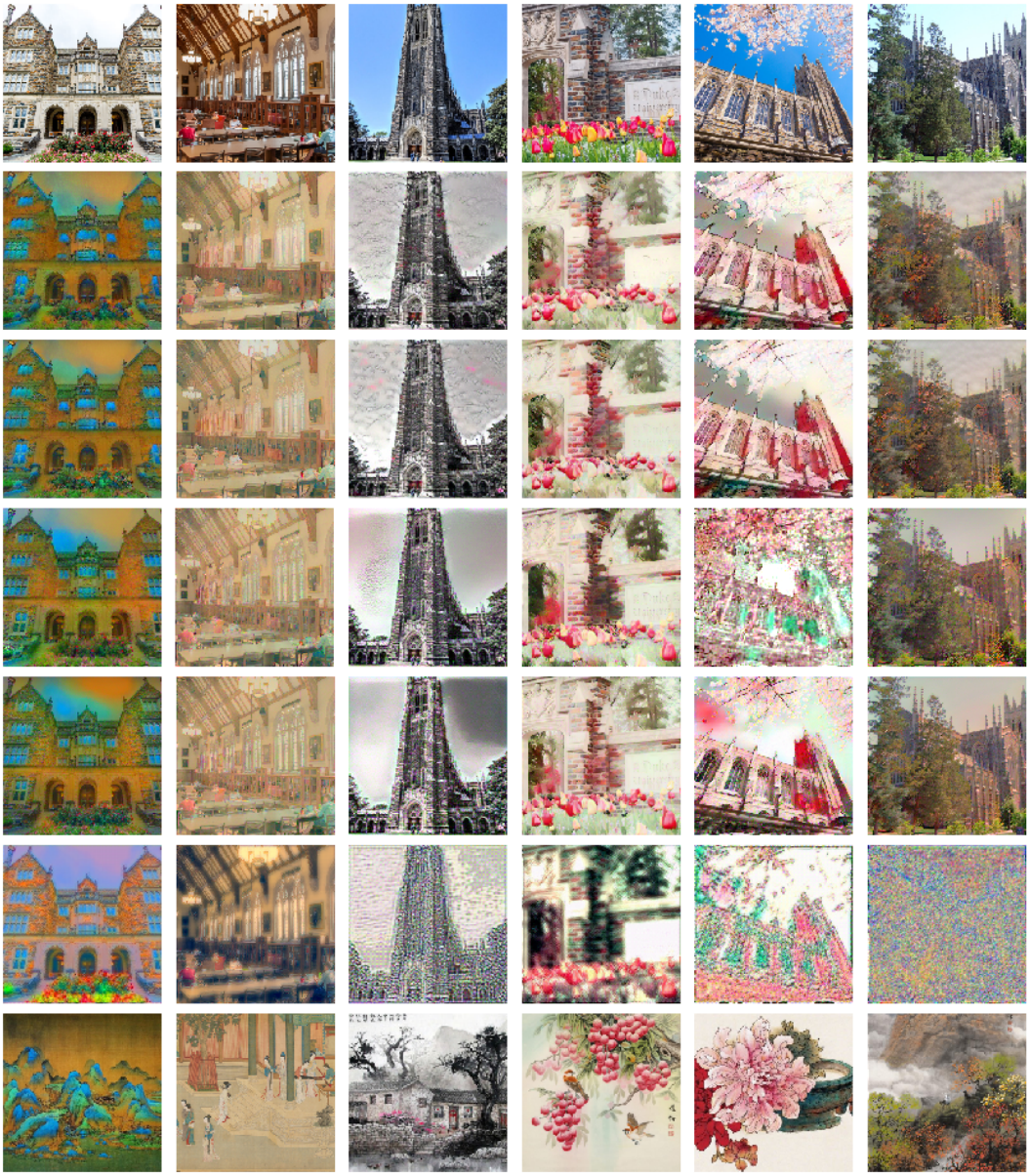

Figure 1: content image, VGG-11, VGG-19, VGG-11_bn, VGG-19_bn, ResNet-18, style image (from top to bottom)

Figure 1 shows our experimental outputs generated by the aforementioned architectures with 6 images each for content and style of varying angles, colors, textures and complexities. When comparing VGG-11 to VGG-19, we see that both models are able to generate very similar visually appealing stylized images across all combinations. The output images show a good blend of content and style of the two original images, appearing unique in style while preserving the original content. When BN is incorporated into the VGG models, we find that the output images seem to put more emphasize on preserving the original content. For instance, in the first column of Figure 1, the color looks much brighter on the VGG-11 and VGG-19 generated outputs compared to their counterparts with BN. We can also see from the chapel hill images shown in the third column, where the BN versions have a much more smoother and clearer sky background compared to the original models. Interestingly, in the set of experiments we ran, VGG-11 with BN seems to have corrupted one image and indicate that

it may not be able to handle certain complexity level of style images as well as the original models and VGG-19 with BN. Finally, we observe that ResNet-18 cannot perform comparable results to the VGG models across all combinations of our selected content and style images. The stylized images being generated by ResNet-18 are color-variant, blurry, pixelated, and contain a lot of noises. Particularly, when the input images get more complex, ResNet-18 does not seem to generate anything that's meaningful or visually perceptible.

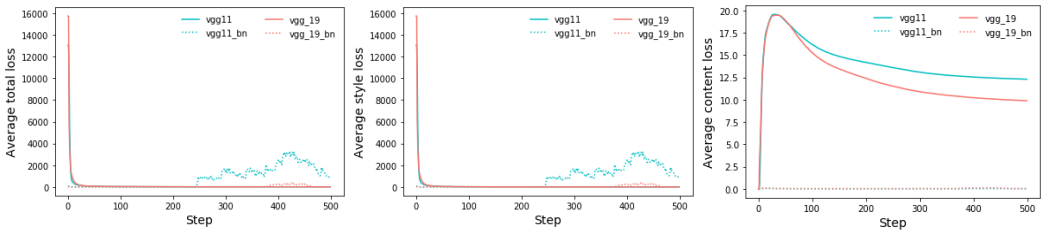

Figure 2: Histories of average training losses using different models. Average total loss, average style loss, and average content loss (From left to right)

In Figure 2, the results validate our hyperparameter setting of the weights on the content and style losses. Regarding the style loss, we see that the VGGs with BN layers start to fluctuate after 250 steps for VGG-11 BN and after 350 steps for VGG-19 BN, while the training histories of the regular VGGs are comparably stable. This contradicts with the common understanding that BN layers stabilize the training process, and suggests that BN layer has different impact to style models. In addition, the histories of the content loss in regular VGGs increases as the number of steps increases, reaches the peak around 50 steps, and start to decrease after that. On the contrary, the histories of using the VGGs with BN layers are flat. The content loss histories echo with the finding in Figure 1 that the VGGs with BN layers tend to preserve the original content, especially VGG-19 as its content loss is lower than that of VGG-11.

4 Conclusion

Our results show that VGG-11 can achieve comparable stylization performance with VGG-19. This implies that we can perform stylization with less computational cost. In addition, the adopted kernels in VGG-11 and VGG-19 seem to be able capture similar patterns in input images. Our results also indicate that ResNet-18 may be ineffective for stylization. This finding suggests that, to achieve successful stylization, it is recommended to adopt VGGs-based architecture instead of ResNets, which echoes with the hypotheses reviewed in [6]. On top of that, we hypothesize that the residual connections may blend low-level and high-level features to certain degree that can lead to performance increase in classification task but not in stylization task. However, it is possible that other larger ResNet models may be able to generate high-quality stylization images by carefully choosing kernels for content or style reconstructions or weights of the loss.

References

[1] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A neural algorithm of artistic style, 2015.

[2] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[3] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[5] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge, 2015.

[6] Alexander Mordvintsev, Nicola Pezzotti, Ludwig Schubert, and Chris Olah. Differentiable image parameterizations. *Distill*, 2018. https://distill.pub/2018/differentiable-parameterizations.

A Timeline and task allocation

This project is jointly developed by Aaron Liu and Christy Hu. Both authors have made equal contribution to all components of the project, including selecting the project topic, doing the literature review, writing code to implement NST on different architectures and preparing for final poster presentation and the written report. The specific timeline of the project is shown in Figure 3 below.

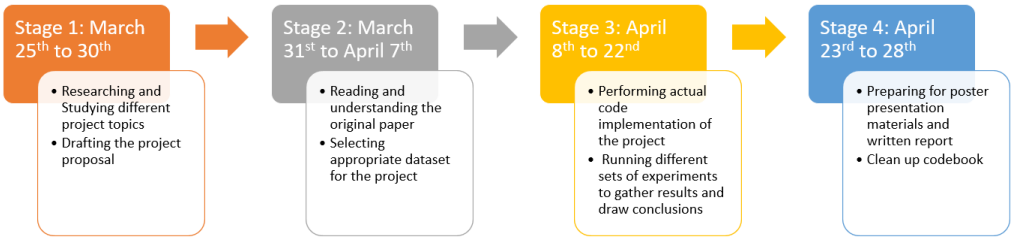

Figure 3: Project timeline