ArtNet: Artistic Style Transfer Network

Mengying Bi (mybi), Cathy Jia (cathyjia), Geoffrey Li (geoffli)

Introduction

Have you ever wondered what Seattle's skyline might look like if Vincent van Gogh painted it in the style of his influential Starry Night? What if Claude Monet had visited Yosemite; what kind of painting would his famous impressionist style produce? What would a blend of Picasso and Hokusai's style look like? With the technique of neural style transfer, originally proposed by Gatys et al. [1], we are able to leverage a trained convolutional neural network's feature maps to derive the style representation of an image and transfer that style to the content of another image. In this project, we implement this technique, along with a few enhancements inspired by work done since then [4][5], and finally offer our own innovation: extending the technique by allowing multiple styles to be simultaneously transferred.

The project consists of two deliverables. The code deliverable is in the form of a PyTorch implementation on a Google Colab notebook, a cloud-GPU service that we leverage to speed up model training and inference. The other part is an interactive user dashboard that is currently hosted on a local machine with pre-stored style, content, and output images for the purpose of faster demonstrations during the poster session.

Related Work

As a baseline, our network leverages the techniques introduced in the paper A Neural Algorithm of Artistic Style by Gatys et al. [1]. These authors were pioneers in the technique of neural style transfer: extracting style features from one image to transfer them to another input image, while preserving the content of that input image. The key innovations they introduced was defining a style representation for a given image. They found that spatial correlations between feature maps across the deeper layers in a convolutional neural network (ones whose activations correspond with "higher-level features") hold the representation for the "style" or "texture" of an image. They also introduce a framework for a loss function that balances both content features of the original input image and the style features of another image; optimizing this combined loss function produces the style transfer. In later work, Gatys et al. also developed a technique for color preservation of the original input image [4].

Our baseline CNN leverages the VGG19 architecture [2] developed by Simonyan and Zisserman, pretrained on the ImageNet image classification dataset [3].

Datasets

We used a collection of master paintings with distinct artistic styles to use for artistic style transfer. Included are paintings from Van Gogh, Monet, Rousseau, Picasso, Baishi Qi, Beihong Xu, Aoyama, and Hokusai, chosen for breadth and distinction of style, and for their recognizability.

Methods

As mentioned before, our network leverages the techniques introduced by Gatys et al. [1]. For our preliminary baseline, we use the intermediate layers (feature-level) of a trained convolutional neural network to use to extract “content features” from the input content image. We leveraged the VGG19 architecture (Fig. 1) [2] as Gatys et al. [1] did, as it produced more visually pleasing results than other architectures like AlexNet and ResNet. We replace the max pooling layers with average pooling layers to produce better results and improve gradient flow, which was also inspired by Gatys et al. [1].

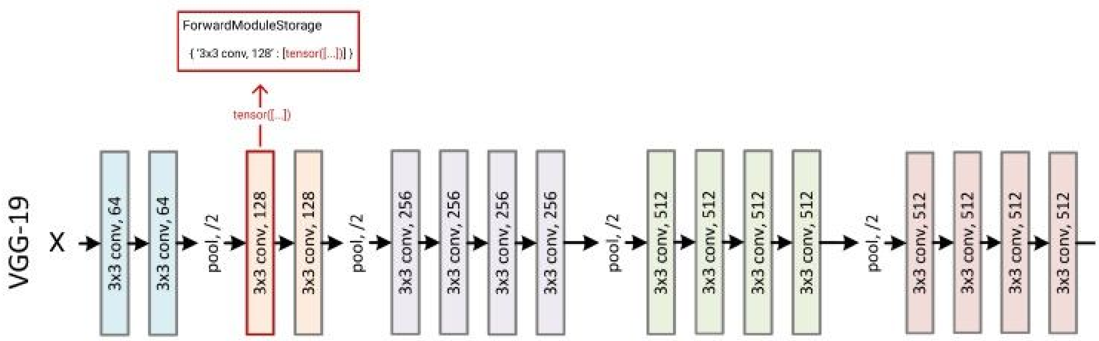


Fig. 1: VGG19 Network Architecture.

To optimize for the content features of our generated image, we simply change any arbitrarily-generated image using gradient descent until the feature responses of our generated image match those of the original image. We define the loss to be the squared error loss between the feature representations of the generated image and the original input image:

$$\mathcal{L}_{content}(\vec{p}, \vec{x}, l) = \frac{1}{2} \sum_{i,j} \left(F_{ij}^l - P_{ij}^l\right)^2$$

Where p is the original input image, x is the generated image, and P^l and F^l are the feature representations in layer l (of original and generated images respectively) at locations i and j.

From the input artistic images, we will extract the “style features” particular to that art piece by deriving a style representation, defined as the spatial correlations across a layer’s feature maps, as Gatys et al. have done [1]. This is done by computing the Gram matrices across all feature maps in a layer l:

$$G^l_{ij} = \sum_k F^l_{ik} F^l_{jk} \qquad E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} \left(G^l_{ij} - A^l_{ij}\right)^2 \qquad \mathcal{L}_{style}(\vec{a}, \vec{x}) = \sum_{l=0}^{L} w_l E_l$$

Then, we define layer l's contribution to the style loss, E_l, as the expectation of the squared loss between the feature map Gram matrices of the generated image (G^l) and the original image (A^l), across the spatial dimensions of the input image (M_l) and the feature maps (N_l). The total style loss is then computed as the weighted average of each style layer's loss contribution (E_l), a hyperparameter of our network.

Finally, the total loss we are optimizing for in our generated image can be summarized as the weighted average between the content loss and style loss:

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

Where α and β are content weight and style weight. They are hyperparameters for adjusting the relative importance given to optimizing towards content and style loss, respectively.

We offer an improvement on Gatys et al.'s approach by extending the ability to optimize towards multiple style images, instead of a single style image. We propose replacing the style loss with:

$$\mathcal{L}_{multi-style}(\vec{a}, \vec{x_1}, \vec{x_2}, ..., \vec{x_n}) = \sum_{s=1}^{n} u_s \sum_{l=0}^{L} w_l E_l$$

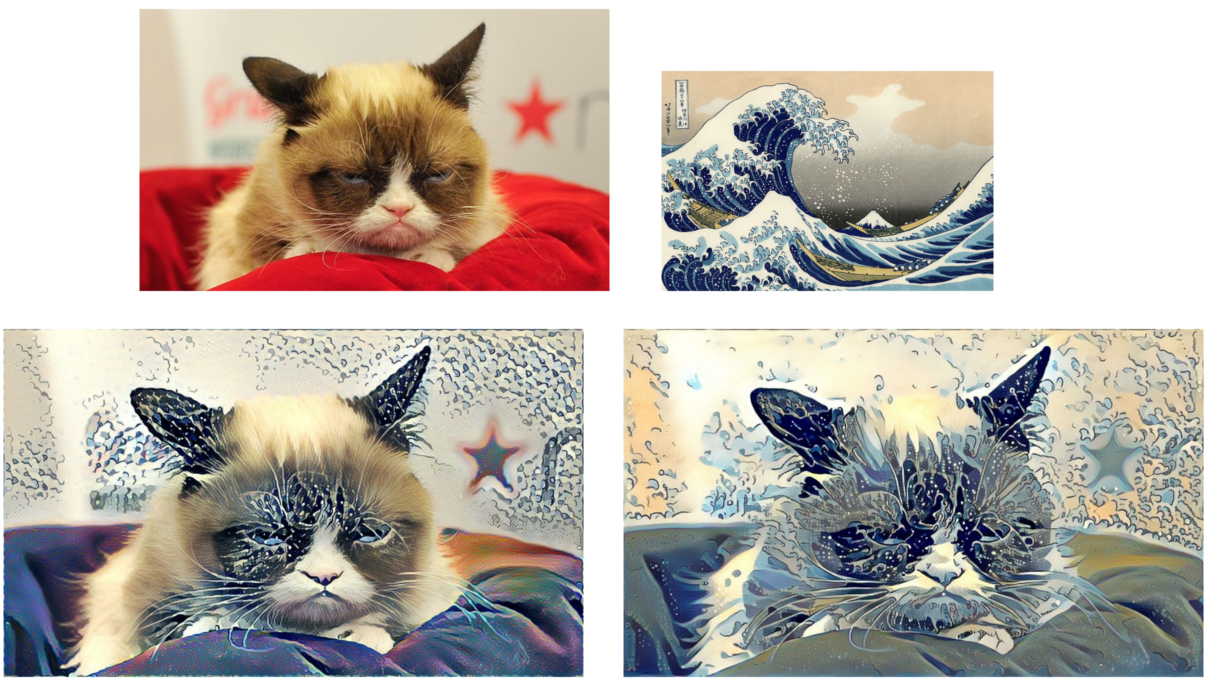
Where n is an arbitrary number of style images and u_s is the relative weight given to that style image (a user-chosen parameter). Our results have shown that this straightforward approach appears to work well. This multi-style loss function is successfully able to blend multiple artworks from a single artistic style (e.g. Monet impressionism) for an enhanced singular stylistic representation, and also is able to blend multiple distinct styles (though in practice we've found this works best when the number of distinct styles is small).

In addition to the above, we also implemented two other enhancements. First, we implemented a feature to preserve the color of the original image. We used YIQ color space and combined IQ channels from content image with the luminance channels extracted from output image [4]. Second, we reduced high-frequency artifacts and improve spatial smoothness of output images by introducing a regularization term known as total variation loss [5].

Experiments and Outcomes

We optimized the total loss, which is the sum of content loss, style loss and total variation loss, to experiment with and evaluate how resultant images turn out. For each pair of content and style transfer, we saved the output image with the lowest total loss. There are many hyperparameters affecting the optimized total loss, such as initial image resizing, image normalization parameters and the number of iterations. The adjustment of the relative weights for content loss, style loss and total variation loss has significant impact on the images generated.

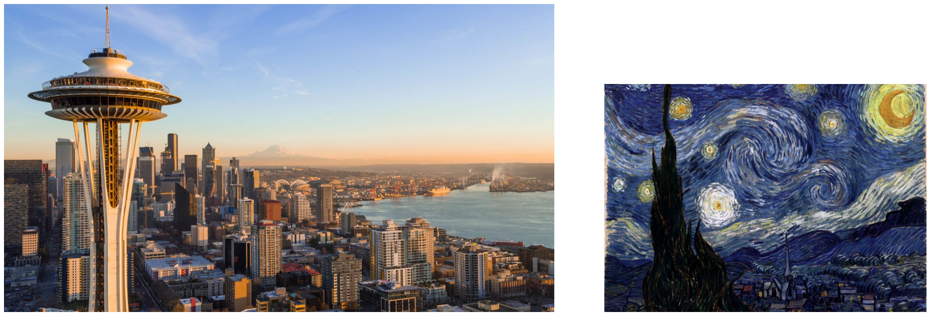
We performed experiments on different combinations of these hyper-parameters, for both single style transfer and multiple style transfer. We observed that as we increased the style weight, the output images have closer styles to the style image but lose some content features (e.g. edges). There is a trade-off between the content loss and style loss which significantly impacts how we use to judge the output image quality (Fig 2).

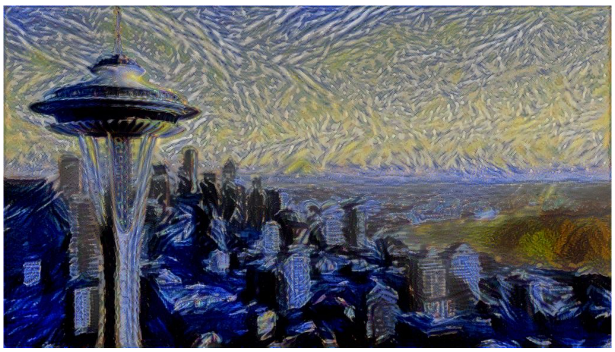


(content weight = 1, style weight = 100) (content weight = 1, style weight = 1000)

Fig. 2. Single-Style: Hokusai's The Great Wave off Kanagawa + Grumpy Cat.

For enhancement, we created output images while preserving the original color (Fig. 3) as well as adding the feature of creating multi-style images with respective style blend weights (Fig. 4).





(with color preservation) (without color preservation)

Fig. 3: Single-Style: Vincent van Gogh's Starry Light + Seattle.

Fig. 4: Multi-Style: Five Monet Paintings + Tulips → Monet's Tulips.

To demonstrate the outcome of the style transfer tool, we implemented an interactive user interface, which enables users to select a content image file and an artistic style to generate an output image. For enhancement features, users are able to choose preserve original image color or not and adjust the style weight for multiple style transfer.

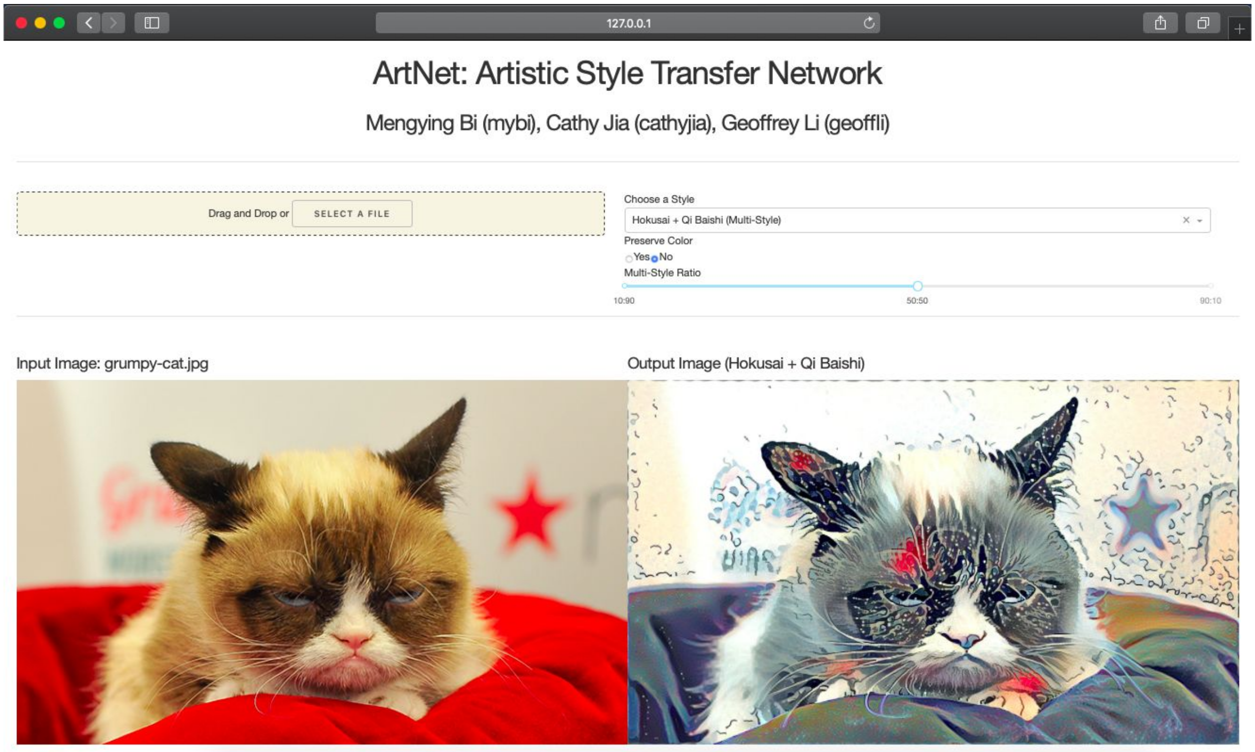


Fig. 5: An Interactive User Dashboard for Style Transfer

Conclusion

Our deep convolutional neural network, ArtNet, in an implementation and improvement on neural style transfer networks originally proposed by Gatys et al. [1]. The network is able to take the content of an input image, and transfer the “artistic style” of another image onto the content input image by extracting the content and stylistic features of the respective images. Further, we have implemented enhancements that allow the original colors of the content image to be preserved, transferring only stylistic texture of the style image. Finally, we propose a straightforward but effective means of transferring the blended style of multiple style images onto the content image. This innovation allows the user to either magnify one distinctive style, or evaluate what a hypothetical collaboration between distinct style might look like when transferred to a given input image.

References

[1] Gatys, L.A., Ecker, A.S., & Bethge, M. (2015). A Neural Algorithm of Artistic Style. ArXiv, abs/1508.06576.

[2] Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. CoRR, abs/1409.1556.

[3] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proc. CVPR, 2009.

[4] Gatys, L.A., Bethge, M., Hertzmann, A., & Shechtman, E. (2016). Preserving Color in Neural Artistic Style Transfer. ArXiv, abs/1606.05897.

[5] Johnson, J.M., Alahi, A., & Fei-Fei, L. (2016). Perceptual Losses for Real-Time Style Transfer and Super-Resolution. ECCV.