Generating Artistic Styles using Neural Style Transfer

Esha Ghorpade, Nitisha Pradhan, Rahul Pal
U.G Student
Computer Engineering, Thakur College of Engineering and Technology

***Abstract* — Neural style transfer is an optimization technique used to take two images—a content image and a style reference image (such as an artwork by a famous painter)—and blend them together so the output image looks like the content image, but gives the result such that it seems painted in the style of the style reference image. Style transfer that we intend on showcasing in this paper is an extremely fun and creative concept which puts forth the capabilities and internal representations of neural networks particularly convolutional neural networks(CNN). Style Transfer is a classic example of image stylization which is basically an image processing and manipulation technique. This paper makes the use of a Convolutional Neural Network called VGG16 to achieve this task. Transfer learning approach is used in our paper wherein VGG16 is used as the network from which the content and style outputs are obtained which give the output by capturing style of the style image and transferring it over the content image.**

Keywords —Nneural style transfer, content image, style image, content loss, style loss, gram matrix

I. INTRODUCTION

Style transfer proves to be an ingenious tool which promises to change the ways we think about art, what originality means, and how we present art in the real world. Style transfer seems quite a relevant creative escape in today’s world allowing amateur artists and laymen to experiment with art and hence enable them to come up with their own artistic masterpiece.Artists can easily lend their creative aesthetic to others, allowing new and innovative representations of artistic styles to live alongside original masterpieces. Style transfer proves to be quite a useful tool especially in the fields of commercial art, photo,video editors, gaming and virtual reality.One of the most clear applications of style transfer is in photo and video editing software. Editing your pictures to add distinguished style, music videos that have famous art styles or to just add creativity in this form to images or video clips. This is the power that such creative tools promise.We can also imagine style transfer being used to create reproducible, high-quality prints for office buildings, or for large-scale advertising campaigns. To implement style transfer which proves to be such a fascinating concept, we have used two neural networks: a pre-trained feature extractor and a transfer network which is VGG16.

Neural Style Transfer (NST) is defined as a class of software algorithms that manipulate digital images, or videos, in order to adopt the appearance or visual style of another image. These image transformations use deep neural networks. As we need to deal with images, large convolutional Neural Networks are required. NST can be used for making digital art using photos, for example by transferring the appearance of famous paintings to user given photographs. This can be found in several mobile apps and even web apps or websites. This method has been used by artists and designers around the globe to develop new artwork based on existing style.

Neural Style Transfer was first introduced by Leon A Gatys in his paper in 2014. The paper [1] had made use of feature representations from the neural algorithm and a linear loss function.

Neural Style Transfer is a considerably artistic application of neural networks. Given a content image, NST can be used to “paint” the image in the style of some artwork or painting. It would stylize the content image in various factors like color scheme, patterns in style image or the brush strokes. NST basically makes any ordinary image an artwork.

Neural Style Transfer also has importance in the computer Vision field. Deep neural networks which are capable of deriving feature representations of images and fuse them appropriately. It is a remarkable use of a neural network which opened up a new branch of research.

The content representation and style representation are extracted and are reconstructed to get a combined image. The content loss and style loss are calculated individually and total loss is calculated by using a linear combination of these. It uses two parameters alpha and beta which are small numbers which are optimized to give results.

II. LITERATURE SURVEY

[1] introduced the concept of Neural Style Transfer and introduced “A Neural Algorithm of Artistic Style”, an algorithm to perform image style transfer. They have demonstrated how to use feature representations from Convolutional Neural Networks to transfer image style between images. On the other hand, [2] introduced a novel thought to integrate artist’s perception in the style transfer rather than the conventional colour or texture transfer. Van Gogh paintings and cubist artwork was used in capturing the overall style of the artist to create artwork productions. The architecture consists of encoder E, representation reconstruction module and decoder D.

In [3], to determine content representation and feature map of images, VGG-16 was used. Content loss and style loss of the generated image were calculated and their weighted sum as total loss. [4] compare the efficacy of relying on various network levels to define a fixed feature, and report novel results that significantly outperform the state-of-the-art on several important vision challenges. [5] introduced a network called VGG which is now used extensively for large scale image recognition as the representations from VGG generalise well to other datasets, where they achieve state-of-the-art results.

[6] describes an approach to predict the style of the image and also presents two novel datasets on which their approach gave excellent classification results. [7] gives a comprehensive overview of the current methods in NST and presents several evaluation methods and compares different NST algorithms both qualitatively and quantitatively.

[8] considers the notion of image style as a local texture and proposes an adaptive patch method that outperformed existing methods. [9] has an interesting approach of a simple image-based method of generating novel visual appearance in which a new image is synthesized by stitching together small patches of existing images. [10] describes an effective way of initializing the weights that allows deep autoencoder networks to learn low-dimensional codes.

[11] is another Gatys paper which introduces a system based on Deep Neural Network that creates artistic images of high perceptual quality and also gives some insights on how humans perceive art. [12] aims to better reflect perceptual similarity of images by computing distances between image features extracted by deep neural networks. While, [13] says that NST should evolve as an interactive tool that considers the design aspects and mechanisms of artwork production.

[14] is another Gatys paper and this one introduces a new model of natural textures based on the feature spaces of convolutional neural networks. It is seen that across layers the texture representations increasingly capture the statistical properties of natural images while making object information more and more explicit. [15] introduces a new network structure, called SPP-net, can generate a fixed-length representation regardless of image size/scale, which sounds useful in a variety of applications.

[16] aims to learn more about the representations of images, by inverting them. Their findings show that several layers in CNNs retain photographically accurate information about the image. On similar lines, [17] uses the inverting technique on deep networks that has been trained on ImageNet and provides several insights into the properties of the feature representation learned by the network, especially colors and contours.

[18] proposes a novel Feature Guided Texture Synthesis (FGTS) algorithm which uses a new distance metric to provide better measures of perceptual similarity. [19] presents a residual learning framework to ease the training of networks that are substantially deeper than those used previously as they are easier to optimize, and can gain accuracy from considerably increased depth. [20] says that theoretically the essence of neural style transfer is to match the feature distributions between the style images and the generated images.

III. PROPOSED METHODOLOGY

The aim of this paper is to generate an image having the ‘style’ of the Style image having ‘contents’ of the Content image. This requires us to capture the ‘style’ as well as content so as to compare how close the generated image is to the desired artistic output.

If we try solving this task in a traditional supervised learning approach, it requires a pair of input images—both an original image and an artistic representation of that original image. Using these pairs, a machine learning model learns the transformation and can apply it to new original images. But this approach is highly impractical, as these kinds of image pairs rarely exist.

Neural Style Transfer uses deep neural networks. Neural networks are used to extract the statistical features of the images with respect to the content and styles so that we can quantify how well the style transfer is working and no explicit image pairs are required. In this approach, only a single style reference image is required for the neural network to apply it to original content images.

The convolution layers of deep networks like the VGG-16 are ideal for learning the style and content (generally deeper layers for content and multiple layers ranging from shallow to deeper layers for the style). So, we pass the content and style images through the VGG-16 and the intermediate hidden layers 'block1_conv1','block2_conv1','block3_conv1','block4_conv1','block5_conv1' for style and “block4_conv3” for the content and compare them with the generated image by calculating the total loss which comprise weighted sum of content loss and style loss as per the comparison with the generated image. The Adam Optimizer minimizes this total loss and updates the generated image until the convergence is met.

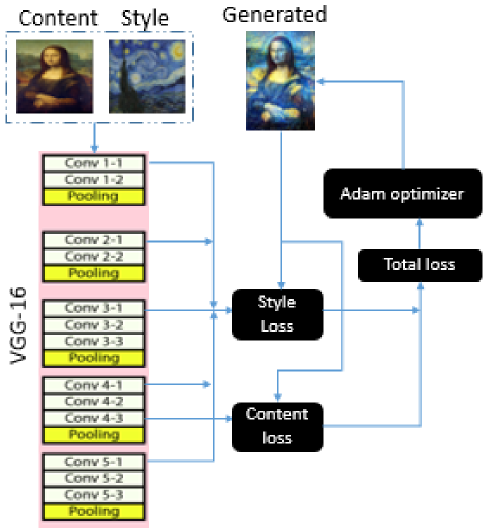

Fig.1. Flowchart

IV. IMPLEMENTATION

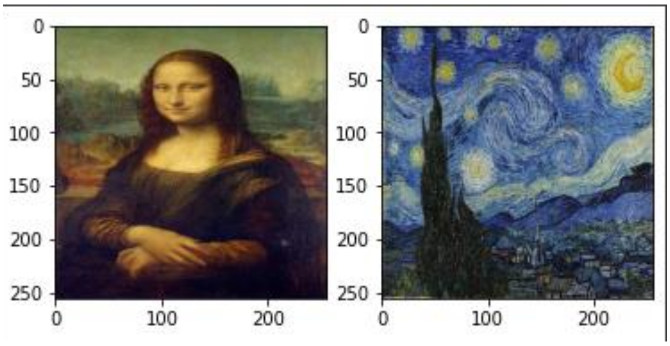

Fig. 2. Content Image (left) and Style Image (right)

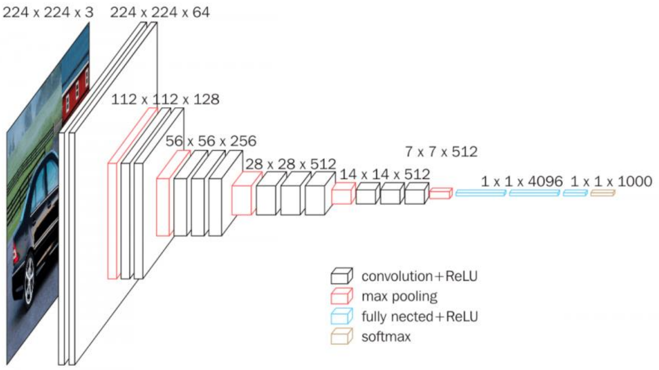

Fig. 3. VGG-16 Architecture

For implementing the neural style transfer, we used the VGG-Network, which is a convolutional neural network used for large scale image recognition, trained on the Imagenet dataset comprising over 14 million images belonging to 1000 classes. We have used the feature space provided by the 13 convolutional layers of the VGG 16 network. Here, we do not use the 3 fully connected layers of the vgg16 architecture. Generally, each layer in the network defines a non-linear filter bank whose capacity of detecting global context i.e., complicated features increase with the position of the layer in the network. Due to this, we create such a loss function for the content image such that the activations of the higher layers of the generated image match the content image. Since higher layers of the VGG 16 capture the content information, therefore we use the 'block4_conv3' layer of the VGG 16 for content reconstruction. Now for the content loss we use the root-mean squared error between the activations produced by the content and generated image. Let $A^l_{ij}(I)$ be the activation of the ith filter at position j which is the height times the width of the feature map, in layer l obtained using the image I. Then the content loss is defined as,

$$L_{content} = \tfrac{1}{2}\ \textstyle\sum_{ij}[A^l_{ij}(g) - A^l_{ij}(c)]^2 \qquad (3)$$

Now, for the style loss, a function is created to make sure that the correlation of activations in the layers are similar between the style image and the generated image. Since, the lower-level layers of the VGG 16 are more focused on the pixel level values which is required for capturing style, therefore 'block1_conv1','block2_conv1','block3_conv1','block4_conv1','block5_conv1' layers are used for style reconstruction. Since for the style representation we require to calculate the correlations between the different filter responses, we use the Gram Matrix which gives us the feature correlations, that is it essentially calculates how correlated are the different feature maps of a given layer. The Gram matrix formula is, wherein the G^l_{ij} is the inner product between the vectorized feature map i and j in layer l

$$G^l_{ij} = \textstyle\sum_k F^l_{ik} F^l_{jk}. \qquad (3)$$

Now the style loss is calculated by minimizing the mean-squared distance between the entries of the Gram matrix from the style image and the Gram matrix of the image to be generated. Therefore the the style loss is, wherein A^l and G^l are the style representations in layer l of the style image and the generated image respectively,

$$E_l = (1/4N_l^2M_l^2)\ \textstyle\sum_{ij}(G^l_{ij} - A^l_{ij})^2 \qquad (1)$$

Now, to generate the stylized image with the perfect balance of the content of a photograph with the style of a painting, we jointly minimize the distance of a white noise image from the content representation of the photograph in one layer of the network and the style representation of the painting in a number of layers of the CNN. The loss function we minimize is,

$$L_{total}(\vec{p},\vec{a},\vec{x}) = \alpha L_{content}(\vec{p},\vec{x}) \quad + \beta L_{style}(\vec{a},\vec{x}) \quad (2)$$

Where α and β are the weights for content and style reconstruction respectively which can be tuned for varied results. In our implementation, we have kept the ratio of α/β as either 1×10^{-3} or 1×10^{-4}. We have used the Adam optimizer to optimize the loss of the network having kept learning_rate as 0.01 and beta_1 as 0.99 while training the white noise input image. We also need to ensure that the generated image is trainable so that the weights keep getting updated. Also, at the start of the network we need to keep the pretrained weights and biases of the VGG 16 network frozen.

V. Results

Results produced by style transfer are usually very subjective and evaluation of the neural style transfer output is highly qualitative. In our implementation we tried to compare the visual appeal of the results obtained by using two different image recognition CNN networks i.e., the VGG 19 as our baseline and VGG 16 as our improved model. The results were very subjective since style in itself is very abstract and depends a lot on personal preferences. Though we could compare the extent of content preserved in the stylized image which was better for the VGG 19 model, the results were appealing for both the networks. The convergence speed i.e., the time taken for the loss function to converge was identical, when trained for 450 epochs for both the networks. Comparing the final loss values of the two methods, clearly the loss using VGG 16 was quite less as compared to the los outputted by VGG 19 method. Though we need to keep in mind that the values of the loss function do not necessarily correspond to the quality of the output image.

The output by VGG 16 and the output by VGG 19:

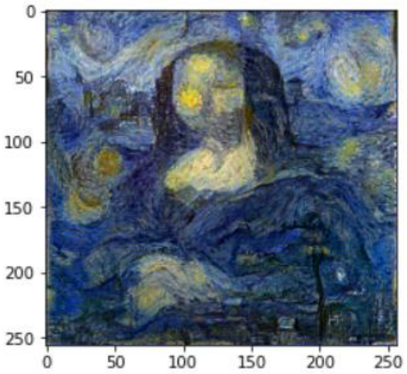

Fig. 4. VGG-16 Output

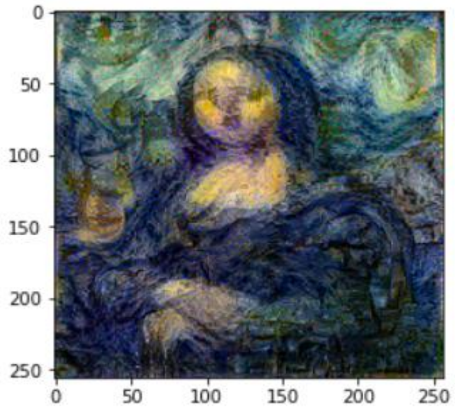

Fig. 5. VGG-19 Output

VI. CONCLUSION

Neural Style Transfer can separate and recombine the image content and style of natural images and create a new image that is a combination of the content and style image. The content image is represented in the manner of the style image with respect to color pattern, brush stroke of painting, some dominant pattern or shape, the composition of a scene and many more. In other words, it captures the style of one image and transfers it onto the other.

The criteria to judge the result of Neural Style Transfer is not mathematically precise or universally agreed upon. The result is judged on the basis of how it looks and how the mixture of the content and style has turned out to be. This can be fairly subjective. Some parameters that can be used to bring clarity to evaluate the resultant image is to see how much has the color map from the style image been transferred, how dominantly or solidly has the content image retained its objects and outlines and so on.

In the method proposed in this paper, the network used is VGG16 which is a convolutional neural network model proposed by K. Simonyan and A. Zisserman from the University of Oxford in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition" [5].

The layers that were used to get content and style output were "block4_conv3" for the content and 'block1_conv1','block2_conv1','block3_conv1','block4_conv1','block5_conv1' for style.

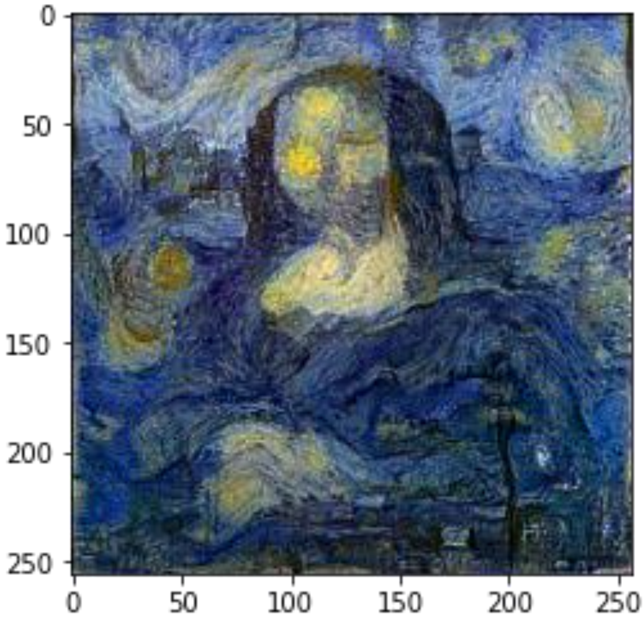

Fig. 6. Resultant Image

The resultant image is a combination of style of the Starry Night image and the base content image is the famous Mona Lisa. The result by this proposed method shows a fair amount of style being captured. The result can be varied by changing a few parameters. Using different layers for content and style outputs will give varied amounts of content and style in the output. Changing the alpha and beta parameters which are used in calculating loss will also give a varied result.

REFERENCES

[1] L. A. Gatys, A. S. Ecker, and M. Bethge on Image Style Transfer Using Convolutional Neural Networks , CVPR 2016.

[2] Zhuoqi Ma, Nannan Wang, Xinbo Gao, Jie Li on Genre-Based Neural Image Style Transfer, International Joint Conference on Artificial Intelligence, 2018.

[3] G. Atarsaikhan, B. K. Iwana, A. Narusawa, K. Yanai and S. Uchida on Neural Font Style Transfer, 14th IAPR International Conference on Document Analysis and Recognition, 2017.

[4] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition. arXiv:1310.1531 [cs], Oct. 2013. arXiv: 1310.153.

[5] K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv:1409.1556 [cs], Sept. 2014. arXiv: 1409.1556.

[6] S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann, and H. Winnemoeller. Recognizing image style. arXiv preprint arXiv:1311.3715, 2013.

[7] Y. Jing, Y. Yang, Z. Feng, J. Ye, and M. Song. Neural style transfer: A review. arXiv preprint arXiv:1705.04058, 2017.

[8] O. Frigo, N. Sabater, J. Delon, and P. Hellier. Split and match: example-based adaptive patch sampling for unsupervised style transfer. In CVPR, 2016.

[9] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in Proceedings of the 28th annual conference on Computer graphics and interactive techniques. ACM, 2001, pp. 341–346.

[10] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. Science, 313(5786):504–507, 2006.

[11] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," ArXiv e-prints, Aug. 2015.

[12] A. Dosovitskiy and T. Brox. Generating images with perceptual similarity metrics based on deep networks. arXiv preprint arXiv:1602.02644, 2016.

[13] A. Semmo, T. Isenberg, and J. Dollner, "Neural style transfer: A ¨ paradigm shift for image-based artistic rendering?" in Proceedings of the Symposium on Non-Photorealistic Animation and Rendering. ACM, 2017, pp. 5:1–5:13.

[14] L. A. Gatys, A. S. Ecker, and M. Bethge, "Texture synthesis using convolutional neural networks," in Advances in Neural Information Processing Systems, 2015, pp. 262–270.

[15] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. arXiv preprint arXiv:1406.4729, 2014.

[16] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5188–5196.

[17] A. Dosovitskiy and T. Brox, "Inverting visual representations with convolutional networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4829– 4837.

[18] X. Xie, F. Tian, and H. S. Seah, "Feature guided texture synthesis (fgts) for artistic style transfer," in Proceedings of the 2nd international conference on Digital interactive media in entertainment and arts. ACM, 2007, pp. 44–49.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[20] Y. Li, N. Wang, J. Liu, and X. Hou, "Demystifying neural style transfer," in Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17, 2017, pp. 2230–2236. [Online]. Available: https://doi.org/10.24963/ijcai.2017/310

[21] https://www.tensorflow.org/tutorials/generative/style_transfer

[22] https://keras.io/examples/generative/neural_style_transfer/

[23] https://en.wikipedia.org/wiki/Neural_Style_Transfer

[24] https://www.analyticsvidhya.com/blog/2020/10/introduction-and-implementation-to-neural-style-transfer-deep-learning/
https://towardsdatascience.com/neural-style-transfer-applications-data-augmentation-43d1dc1aeecc

[25] G. Atarsaikhan, B. K. Iwana, A. Narusawa, K. Yanai and S. Uchida on Neural Font Style Transfer, 14th IAPR International Conference on Document Analysis and Recognition, 2017.