Artistic Style Transfer

J. Linkemeyer, M. Wodrich

April 3, 2021

Abstract

bla bla bla

Contents

1 Introduction

Artistic style transfer aims at converting the content of a given image into the style of a given painting. In 2015, Gatys et al. [1] proposed a deep neural network (DNN) to perform artistic style transform on any given image.

A Convolutional Neural Network (CNN) is a type of DNN that processes an image using convolution. Convolution describes the process of applying a filter kernel to an image, which can be seen as a feature extraction process. A CNN applies one or more distinct filter kernels in each layer to a given image. During the training process of a CNN, the values of those filter kernels, denoted *weights* in the context of neural networks, are adapted in order to learn a specific task, such as, for example, image classification. Gatys et al. [2] found that style and content of an image are separable. The key idea behind their approach is the finding that object information, or content of an image gets increasingly precise along the layers of a Convolutional Neural Network (CNN). Therefore, the content of an image is 'stored' in higher layers, while the style of an image can be obtained through the structure of an image, which can be represented by the correlation of its filter responses in all layers.

Separating style and content of an image allows to create new images with the content of one image and the style of another. The present work aims at replicating the approach by Gatys et al. Furthermore we will investigate and propose improvements to ensure a better separation between the content and the style of an image.

2 Related Work

Artistic style transfer using deep networks has first been introduced by Gatys et al. [2] in 2015. Since then, many researchers conducted various follow-up studies and investigated different mechanisms for accessing the style and the content of an image. Consequently, various solutions of how the combine the style of an image with the content of another image have been proposed.

In the present study we aim at replicating the findings from Gatys et al. Finally, we want to conduct ablation studies and compare our results to the improvements proposed by other researchers.

2.1 Image Style Transfer Using Convolutional Neural Networks by Gatys et al.

In their approach to artistic style transfer, Gatys et al. use the information that for CNNs trained for object recognition the different layers within the image processing hierarchy of a CNN represent distinct levels of abstraction and sub-information of a given image. Their approach is based on the VGG-19 achitecture pretrained on imagenet. [muss man das hier dann auch zitieren? oder VGG-19, imagenet] Visualizing a reconstructed image from different layers (see Figure 1), the researchers found that higher layers in the network carry information about the content of an image, instead of precise pixel values as early layers. The reconstruction of an artistic style has been investigated by creating a feature space consisting of the correlations of different filter responses from different layers. Results showed that each layer contains stylistic information and that we can precisely reconstruct artistic styles from the combination of the filter responses. Finally, the style representation only consists of texture information regardless of the content and its arrangement.

The researchers use their findings that style and content of an image are separable to create new images capturing the style of one image and the content of another. More precisely, Gatys et al. chose to apply the style of different historically relevant art pieces depicted in Figure 2 to a photograph of the neckarfront. The resulting image of this style transfer process carries the object information and global arrangement of the content image, combined with the color and local structure of the style image. In a nutshell, the style transfer net-

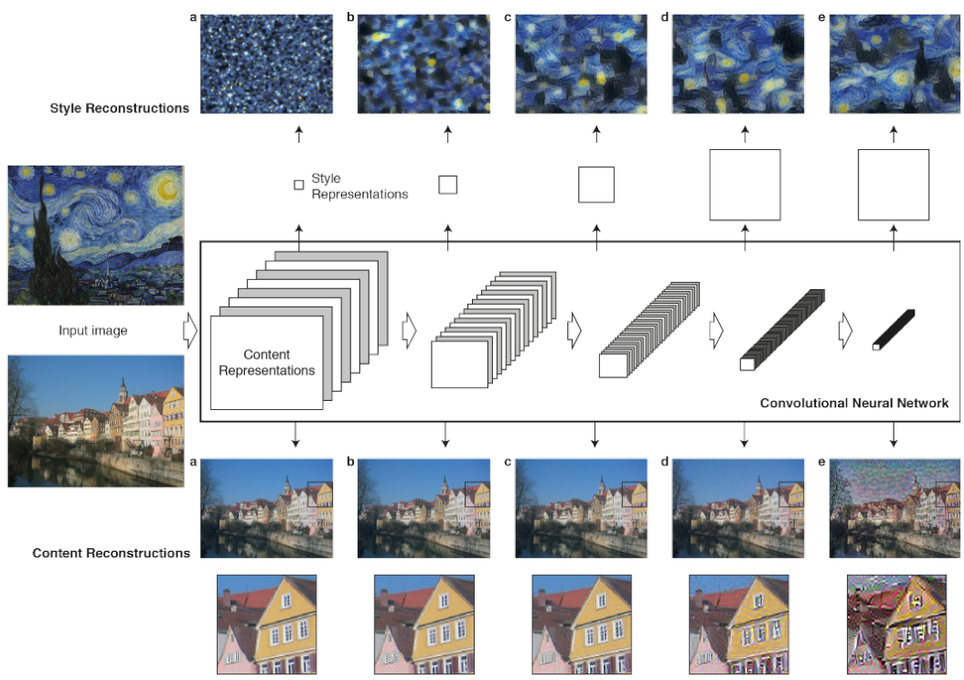

Figure 1: **Style and content reconstructions by Gatys et al.** The style and content input images have been fed into the CNN to investigate which layers should be used as output layers for the style transfer model. The upper row of images shows the **Style Reconstruction** from different subsets of layers from the VGG-19 model ('conv1 1' (a), 'conv1 1' and 'conv2 1' (b), 'conv1 1', 'conv2 1' and 'conv3 1' (c), 'conv1 1', 'conv2 1', 'conv3 1' and 'conv4 1' (d), 'conv1 1', 'conv2 1', 'conv3 1', 'conv4 1' and 'conv5 1' (e)). The more layers included, the bigger the scale of the style gets. The lower row of images shows the **Content Reconstruction** of the content image using different layer's responses ('conv1 2' (a), 'conv2 2' (b), 'conv3 2' (c), 'conv4 2' (d) and 'conv5 2' (e)). The resulting images indicate that lower the layer, the better the content can be reconstructed.

Figure 2: **Art pieces used for artistic style transfer in the paper by Gatys, Ecker and Bethge (2015).** From left to right: *The Shipwreck of the Minotaur* by J.M.W. Turner (1805), *The Starry Night* by Vincent van Gogh (1889), *The Scream* by Edvard Munch (1893), *Femme nue assise* by Pablo Picasso (1910), and *Composition VII* by Wassily Kandisky (1913).

work allows to apply the styles of impressionism (Vincent van Gogh), expressionism (Edvard Munch), cubism (Pablo Picasso), and various other art epochs to any given input image. Figure 2 shows the style images Gatys and colleagues chose in their paper. The neural network Gatys et al. introduced takes two images - one content image and one style image - as input. Because style and content of an image are not well-separable in general, the training of neural network for style transfer aims at finding a trade-off between a loss that is separated into a weighted style and a weighted content loss. For the image the network produces, the feature representation of the content is compared to the one of the content image (denoted in the content loss) using summed squared error. Formally, the content loss is defined as:

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

with $\vec{p}$, $\vec{x}$ being the original and the generated image and F^l, P^l being the feature representations of the images $\vec{p}$, $\vec{x}$ in the layer l.

For the style loss the feature representation of the style is compared to the one of the style image. The calculation here involves the filter outputs from several layers of which we calculate the correlation via a gram matrix. In order to do so, the feature maps from one layer are vectorized and multiplied via inner product. Finally, normalized summed squared error is used to obtain the overall loss of style. Formally, the entries of the Gram matrix G^l can be expressed as:

$$G_{ij}^l = \sum_k F_{ik}^l F_{jk}^l \tag{2}$$

where l denotes the layer and F_i^l, F_j^l represent the vectorised feature maps i and j of the respective layer l. To obtain the style loss of one layer, Gatys et al. compare the Gram matrices of the generated image and the original style image with summed squared error and multiply it with a normalisation term. The loss of a style layer l formally can be expressed as:

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G_{ij}^l - A_{ij}^l)^2 \tag{3}$$

with G_{ij}^l and A_{ij}^l denoting the Gram matrices' entries from the generated image and the input style image obtained through the calculation (2). In the normalisation term, N_l denotes the number of feature maps in layer l, whereas M_l describes the size of the feature maps of the respective layer, which can be calculated by multiplying its height and width.

Finally, the style loss can be obtained by weighting summing up the losses of all style layers. Formally this is defined as:

$$\mathcal{L}_{style}(\vec{a}, \vec{x}) = \sum_{l=0}^{L} w_l E_l \tag{4}$$

where w_l denotes the weight with which the respective layer's loss is counted towards the total loss.

To obtain the total loss of the network, both the content loss and the style loss need to be taken into account. This is done by simply weighting and adding them:

$$\mathcal{L}_{total} = \alpha \mathcal{L}_{content} + \beta \mathcal{L}_{style} \tag{5}$$

where α and β are the weights for content and style loss. While this loss is iteratively being improved, the network becomes better at both keeping the content from the original content and in keeping the style from the original style image.

For their style transfer, Gatys et al. chose to use the only specific layers of the pretrained VGG-19 net to train the style and content transferal, as this produced the best results (as indicated in Figure 1). In other words, they build a model which takes the outputs of layer 'conv4 2' (for content) and layers 'conv1 1', 'conv2 1', 'conv3 1', 'conv4 1' and 'conv5 1' (for style) as input and trains their parameters to successfully perform the task of artistic style transfer. The weights for the style loss were all set to 1/5, meaning that each layer counted equally much towards the total style loss. Results have shown to be good for a ratio of style and content loss (α and β) that weights the content loss as more important, however clear number that are a universal good fit have not been proposed. [oder???]

2.2 Follow-up Studies and Other Related Work

Following the paper by Gatys et al. [2], several improvements and alternatives have been proposed to perform the task of artistic style transfer. While in the original paper the authors make use of the Gram matrix to calculate the style loss, Li et al. [3] combine the idea of using CNNS with Markov Random Fields. Opposed to the the Gram matrix which computes a global style, this allows to preserve local patterns of the style inputs which can be used and adapted throughout the whole output image.

As Gatys et al. based their approach on the pretrained object classifier VGG-19, the question arises if maybe other similar networks might perform equally good or even better, ideally being less complex and using less computational resources at the same time. Nikulin et al. [4] compared the performance of VGG-19, VGG-16, AlexNet and GoogLeNet. The results showed that the VGG-19 achitecture is better suited to perform this task compared to the other achitectures, however VGG-16 performs similarly. AlexNet and GoogLeNet loose a lot of fine details, possibly due to large kernels and strides, and therefore perform much worse at the task of artistic style transfer. Additionally, Nikulin et al. found out that it is possible to only partially transfer the style, meaning that structural and high-level information gets transferred, while the colors from the original image are being kept (see Figure 3). This is achieved via excluding the lower layers from the style loss.

The mechanism behind artistic style transfer proposed by Gatys et al. [2] is an iterative algorithm which can take many iterations, depending on the hyperparameters used, resulting in relativly slow stylization. In order to tackle this problem, ...

Figure 3: **Style Transfer vs. Partial Style Transfer by Nikulin et al.** Images produced by Nikulin et al. [4] using the upper left images as content image and the upper right image as style image. While normal artistic style transfer uses the colors of the style image (lower left), partial style transfer adapts the structural style of the style image, but uses the colors of the content image.

3 Our Approach

In the present work, we tried to replicate the findings of Gatys, Ecker and Bethge. For this purpose, we applied transfer learning and used the VGG19 architecture pretrained on the ImageNet data set as proposed by the original authors. This network is able to precisely perform image classification and is therefore well suited to extract high-level content information of an image.

3.1 Model Summary

4 Results and Discussion

We investigated different combinations of content and style images. We first chose to apply our style transfer to the images used in the original paper by Gatys and colleagues. The content image is an image of the Neckarfront in Tübingen, Germany, depicted in figure 4. The style images are the artworks presented in figure 2. Figure 5 shows the results on all combinations using our style transfer model. As in the original paper, we chose layer 'block4_conv2' of the VGG-19 model as the content layer and the following layers as style layers: 'block1_conv1', 'block2_conv1', 'block3_conv1', 'block4_conv1', 'block5_conv1'. For the despicted reults, we chose Adam optimizer with a learning rate of 4, content weight 1 and a style weight of 10^{-2}. We optimized the generated image for 2000 epochs. Overall, our results differ from the results by Gatys and colleagues by containing less content of the style image.

To see how the network reacts to other content images, we chose three more content images and performed the artistic style transfer on those as well. For the style images, we chose The Scream by Edvard Munch (1893) and The Starry Night by Vincent van Gogh (1889) as in the original paper. In addition to that, we also chose the image Caféterrasse am Abend by Vincent van Gogh (1888) and an image from the street artist James Rizzy. The original images as well as all possible style transfer combinations are documented in figure

Figure 4: **Photograph of the Neckarfront in Tübingen, Germany.** This image is used as the content image for the style transfer in the original paper by Gatys, Ecker and Bethge.

Figure 5: **The Neckarfront image in the styles from the original paper.** This figure depicts our results using VGG-19's 'block4_conv2' as the content layer and 'block1_conv1', 'block2_conv1', 'block3_conv1', 'block4_conv1', 'block5_conv1' as style layers.

6. We observed that different style images react differently to specific style weights. Using The Scream and The Starry Night as a style images, the ratio of content and style weight (α/β) was larger compared to the ratio used to transfer to content images to the other two styles. We found that for all variability of the content images - large unicolor areas for the mountain image, the very small details in the sunflower image and the unsharp background of the puppy image - the style transfer works equally well. We also observe that our model captures edges extremely well, especially for the mountain image. For all images, the colors of the style images are captured very well and the brushstroke matches that of the artist.

In a further analysis, we captured the effects of different style weights. The results can be found in figure 7.

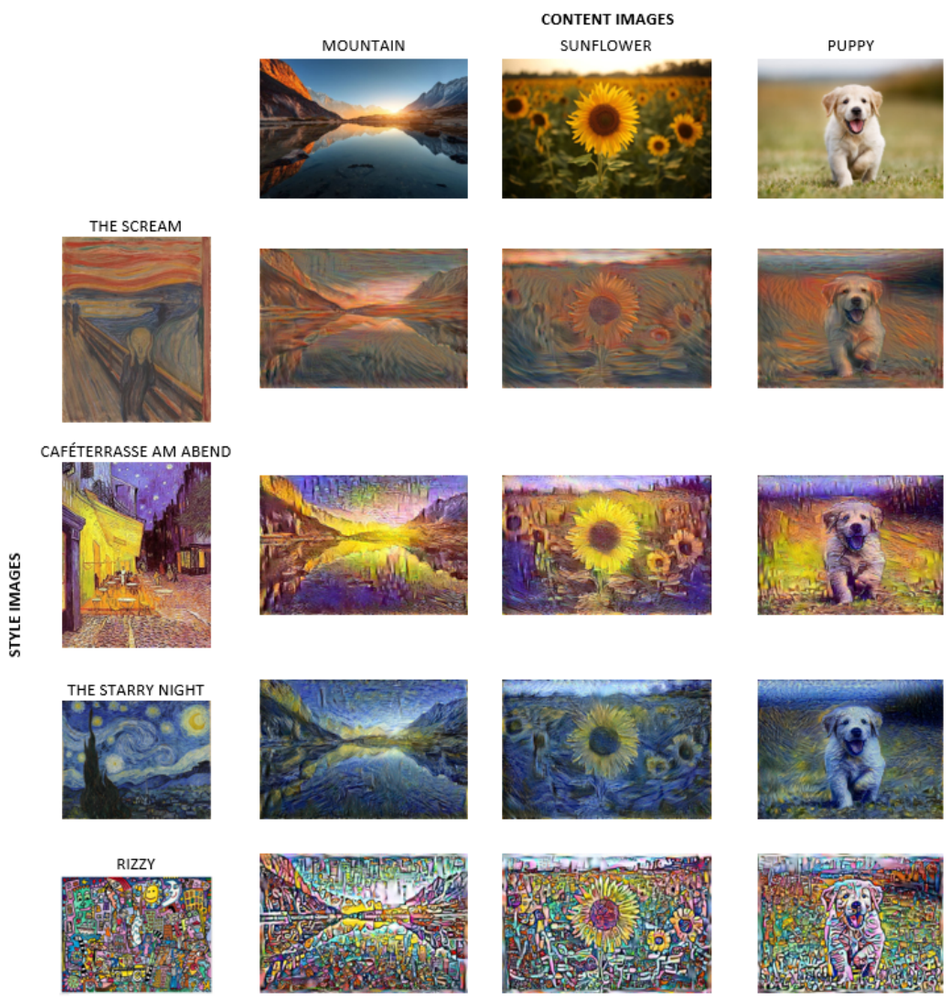

Figure 6: **All combinations of various content and style images.** The content images presented in the first row were obtained through a google search. The style images depicted in the very left column are *The Scream* by Edvard Munch (1893), *Caféterrasse am Abend* by Vincent van Gogh (1888), *The Starry Night* by Vincent van Gogh (1889), and an image from the street artist James Rizzy.

Figure 7: **Analysis of different style weights.** The combination of the Neckarfront image as the content image in the style of Kadinsky's *Composition VII* is examined. All images are created with a content weight $\alpha = 1$, learning rate = 4 and 5000 iterations. The different style weights (β) are documented above each image. It is observable that for high style weights, the image is increasingly abstract and very close to the original style image. For $\beta <= 0.00001$ the content of the Neckarfront image overpowers the style of *Composition VII* extremely.

References

[1] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," *arXiv preprint arXiv:1508.06576*, 2015.

[2] ——, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2414–2423.

[3] C. Li and M. Wand, "Combining markov random fields and convolutional neural networks for image synthesis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2479–2486.

[4] Y. Nikulin and R. Novak, "Exploring the neural algorithm of artistic style," *arXiv preprint arXiv:1602.07188*, 2016.