Variational Style Transfer

Jan Schopohl Dominik Fuchsgruber

Abstract

We propose a variational approach to neural style transfer, where the style of an artistic image is transferred to another. By using variational autoencoders to encode the style of an image, the latent style space is enforced to be smooth. This allows for better interpolation between different styles as well as sampling new unseen styles from the style space.

1. Introduction

The task of artistic style transfer aims to extract the style of an (artistic) image and transfer it to the content of another picture and was first introduced by the seminal work of Gatys et al. in [1]. Their approach optimizes a random noise image to fit a particular style and content, which is very demanding computationally-wise. Ever since, many methods were developed that generalize the method proposed in [1] to arbitrary styles while also significantly alleviating the computational burden by introducing fully feed-forward neural networks. As has been shown in [2], using an encoder-decoder architecture, where the decoder layers are conditioned on a style image using using *Adaptive Instance Normalization* (AdaIn), can achieve visually pleasant results. Instead of using the same encoder module for style and content image, we follow Kotovenko et al. [4] and make use of distinct encoders for content and style images.

Furthermore, we extend their approach by using a variational autoencoder architecture [3] to encode the style from an image. This enforces smoothness to the latent style space, which allows for better interpolation between different style embeddings. Furthermore, we are able to draw samples from the latent style space, opening up a generative perspective to our approach.

2. Related Work

Artistic style transfer was introduced by Gatys et al. in [1], where the authors noticed, that the style of an image is captured by the gram matrices of different feature maps using the space of a VGG network. Optimizing a random noise image to minimize the perceptual loss with respect to the content image, while at the same time minimizing the discrepancy of the gram matrices with regards to a style image, leads to a visually appealing stylization. However, their approach suffered from heavy computational requirements, which lead to the development of fully feed-forward architectures for style transfer. [2] introduces an autoencoder-based approach that is applicable to transfer arbitrary styles the model was not explicitly trained on. They use *Adaptive Instance Normalization*, which manipulates the instance-wise statistics of a content image before each decoder layer, which surprisingly proofed to be sufficient to achieve high quality stylizations. Their method was expanded upon by Kotovenko et al. [4], which proposed to incorporate two distinct encoders for content and style images and furthermore introduced a disentanglement loss term that counteracts entanglement of the two latent spaces.

Adaptive Instance Normalization is also used to approach different problems, one of which being synthesizing photo-realistic video sequences of speech expressions for a particular individual [6]. They use AdaIn to condition a decoder module on an embedding of the facial characteristics of a person. In our project, we heavily rely on the architecture of their approach, namely residual blocks both in the encoders and the decoder as well as bottleneck layers between the two. We extend this model by introducing U-Net connections between the content encoder and the decoder module, which was first proposed in [5].

The concept of encoding images into a distribution space rather than just embedding them as simple points was first used in [3]. By sampling from those distributions during training, variations in the latent space can be captured, which leads to a smoother manifold. Furhtermore, the distributions provided by the encoder are regularized to resemble a standard normal, which allows for sampling from the latent space as well.

3. Our Approach

We denote with $c \in \mathbb{R}^{3\times H\times W}$ a content image and analogously with $y \in \mathbb{R}^{3\times H\times W}$ a style image.

3.1. Architecture

Our model consists of three parts: Two distinct encoders for content and style images as well as a decoder that outputs stylizations.

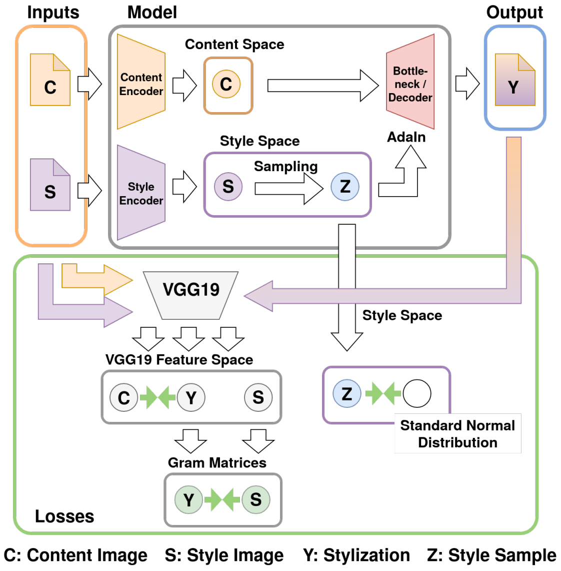

Figure 1. Visualization of the different components in our architecture as well as how the loss terms are implemented. Perceptual and style losses are computed in the VGG19 feature space.

- The *content encoder* E_c takes a content image c and applies several convolutions and pooling operations to a content image. It is composed of several residual blocks, and the output still retains a spatial shape, i.e. $E_c(c) \in \mathbb{R}^{D_c \times H' \times W'}$.

- The *style encoder* $E_s(s)$ is designed analogously to the content encoder, with the only difference being the use of adaptive pooling after the last convolution. That is, the style encoder yields a fixed-size embedding distribution by outputting the mean and variance of a normal distribution: $E_s(s) \in \mathbb{R}^{2D_s}$.

- The *decoder* mirrors the design of the two encoders. It is preceded by several bottleneck layers, which do not change the resolution of the input image. Both the bottleneck layers as well as the upsamling convolutions are conditioned on the style embedding using *Adaptive Instance Normalization*. Additionally, we use U-Net connections that connect the output of content encoder layers to the respective components of the decoder. We denote its output with $y = D(E_c(c), E_s(s))$.

3.2. Adaptive Instance Normalization

Before each convolution in the decoder, the instance wise mean μ and variance σ^2 of an image is shifted according to a learnable, linear transformation the style encoding:

$$\{\mu, \sigma^2\} = \mathbf{W}E_s(s) + b \tag{1}$$

3.3. Losses

Our network is trained with respect to three different objectives, two of which are computed in the features space of a pre-trained VGG19 network. Let $VGG^{\{i\}}$ denote the feature maps of the i-th layer and $G(x)$ the gram matrix of an activation x.

- The content loss $\mathcal{L}_{content}$ penalizes deviations of the stylizations regarding the content image and is implemented as a perceptual loss. That is, we consider the feature maps of the $relu_{4,2}$ layer of the VGG19 network.

- The style loss $\mathcal{L}_{style}$ penalizes deviations from the style image by comparing the gram matrices of different VGG19 activation maps. We include all layers up to $relu_{4,2}$ into the style loss term.

- The variational loss $\mathcal{L}_{KL}$ regularizes the latent space to resemble a standard normal distribution. It uses the Kullback-Leibler divergence to compare the two distributions.

$$\begin{aligned} \mathcal{L}_{content} &= \sum_i \|VGG^{\{i\}}(c) - VGG^{\{i\}}(y)\|_2^2 \\ \mathcal{L}_{style} &= \sum_i \|G(VGG^{\{i\}}(c)) - G(VGG^{\{i\}}(y))\|_2^2 \\ \mathcal{L}_{KL} &= KL(\mathcal{N}(E_s(s)_{:D_s}, E_s(s)_{D_s:})\|\mathcal{N}(0, I)) \end{aligned} \tag{2}$$

We assign different weights λ to each component of the content and style loss terms, as well as the variational loss.

3.4. Datasets

We use images from the *Places365* dataset [7], containing 1.8 million scenic photographs as content images. A subset of 100 images from the *WikiArt* dataset is selected that exhibits a broad range of abstract and artistic styles. We explicitly exclude realistic paintings as they do not contribute to the learning of styles. For both datasets, images are downsampled and cropped to a resolution of 96x96 pixels.

4. Results

5. Conclusion and Future Work

References

[1] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A neural algorithm of artistic style. *CoRR*, abs/1508.06576, 2015.

[2] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[3] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013.

[4] Dmytro Kotovenko, Artsiom Sanakoyeu, Sabine Lang, and Bjorn Ommer. Content and style disentanglement for artistic style transfer. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[5] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.

[6] Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor S. Lempitsky. Few-shot adversarial learning of realistic neural talking head models. *CoRR*, abs/1905.08233, 2019.

[7] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 487–495. Curran Associates, Inc., 2014.