Deep Learning- Final Course Project

Introduction

Style transfer is a technique that allows us to manipulate the style of an image without changing its content. It has been widely used in various domains, such as graphic design, photo editing, and art creation. In recent years, it has become a popular research topic in computer vision and machine learning, with the aim of generating high-quality artistic images that blend the content of one image with the style of another. In this project, we focus on transferring the style of real-world images to Monet's paintings. The motivation behind this task is to create visually appealing images that incorporate Monet's unique style and aesthetics. Monet's style is characterized by vibrant colors, loose brushstrokes, and a sense of light and atmosphere. By using style transfer, we can bring these artistic qualities to ordinary images, creating beautiful and unique visual compositions.

Related Works

Style transfer, also known as artistic style transfer, is a process of synthesizing a new image by combining the content of one image with the style of another. This technique has numerous applications in the field of computer vision, such as creating art, enhancing images, and generating realistic virtual environments. In recent years, deep learning approaches have been widely used to perform style transfer. In this section, we will discuss some of the key works in this field.

One of the most prominent works in style transfer is CycleGAN. CycleGAN is a deep learning model that can learn to transfer the style of one image to another without the need for paired training data. Instead, the model is trained on unpaired data, which makes it much easier to collect and use in practice. CycleGAN uses a pair of generative adversarial networks (GANs) to perform the style transfer. The first GAN maps the input image to the style of the target domain, while the second GAN maps the output image back to the original domain. By using two GANs and enforcing cycle consistency between the input and output images, CycleGAN is able to produce high-quality style transfer results.

Another important work in style transfer is the use of convolutional neural networks (CNNs). CNNs have been widely used in image processing tasks and have shown impressive performance in image style transfer. In this approach, the content of an image is represented by the activations of a deep neural network, while the style is represented by the correlations between the activations of different layers. The style transfer is achieved by minimizing a loss function that balances the content and style representations of the input image.

In summary, deep learning approaches have shown great potential in performing style transfer. CycleGAN and CNN-based methods are some of the key works in this field. These techniques have numerous applications in the fields of computer vision and image processing, and they continue to be an active area of research.

Solution

CycleGAN - General approach

This image to image model learns how to "translate" images into Monet style ones, in contrary to per image training in the Neural style transfer above.

The model architecture is very much based on Vanilla-GANs and gets it's name from it. Here we are using Generator-Discriminator as in standard GAN, but with several modifications.

The generator uses encoder-decoder first to generate style-image from the original image, and then a different encoder-decoder to generate back from this image the original content image. In the mean- time, the discriminator tries to guess from the style generated image and the train-data style image which is fake and which isn't. In addition a second discriminator is trying to guess if the output of the second encoder-decoder or the original input image are fake.

The process can be formalized also as:

Domain-A - Discriminator-A - [Real/Fake]

Domain-B - Generator-A - Discriminator-A - [Real/Fake]

Domain-B - Discriminator-B - [Real/Fake]

Domain-A - Generator-B - Discriminator-B - [Real/Fake]

Experimental results

Based on the experiments conducted in the project, several insights were obtained regarding the CycleGan method. Firstly, as expected, it was observed that using a small dataset of only 300 images for training the CycleGAN model produced suboptimal results. However, the performance of the model was significantly improved when the dataset was augmented to include more images. This highlights the importance of data augmentation in image-to-image translation tasks, especially when the available dataset is limited. Overall, these findings emphasize the importance of carefully selecting the appropriate methods and techniques for image-to-image translation tasks, and highlight the benefits of exploring different options to optimize the performance of the models. Additionally, the use of data augmentation can be highly effective in overcoming the challenges posed by small datasets, while choosing the right optimizer can also make a significant difference in the final results.

Augmentation

We generated from the 300 selected Monet images another 600 images, therefor tripling the train images, just by using augmentation! We've used random crop and horizontal flip only. The reason is that it's important to keep the augmented images in the latent space of the original ones. By adding some color filters (such as black-white or negative) or spinning them sideways and adding black padding, the images would no longer be Monet style!
Therefor we kept conservative augmenting only.

left- original, right- augmented image generated from it.

Model

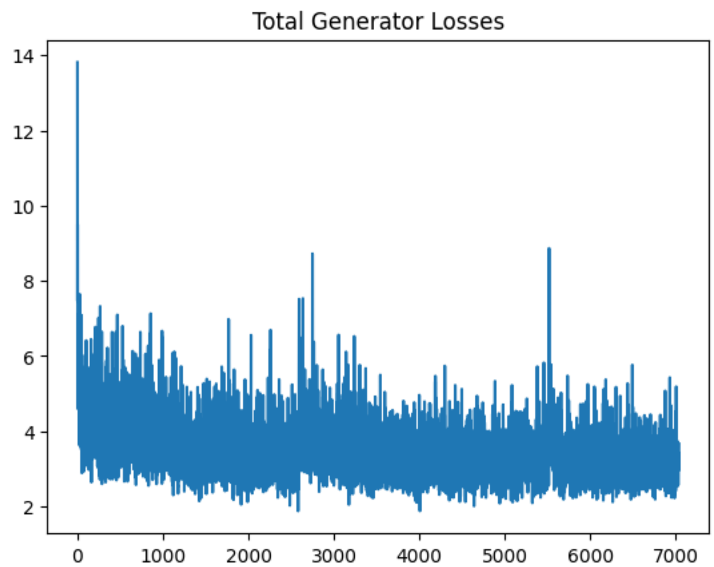
CycleGan

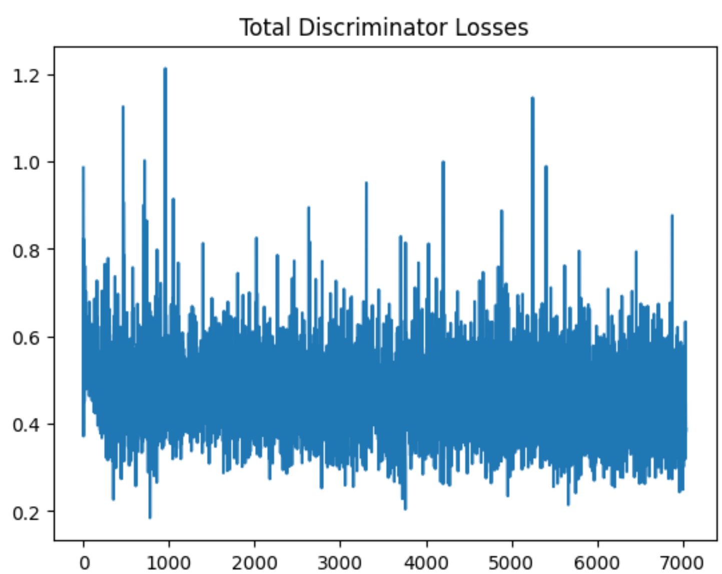
The cycle GAN is trained upon all dataset instead of one image. To train the generator to learn translating images of certain style to another a very log training loop must run on all images and train 2 discriminators and generators for each step in the loop. Therefor its training is much more time and resources consuming. We still explored several modifications in order to see their affect

Activation

We tried using Relu, Tanh and LekyRelu as activation for different placed in the discriminators and generators architecture. This modification have not shown clear importance to models loss, and there fore we early stooped the, to save computing time. In the end we've stacked to the Relu activation.

Loss Plots



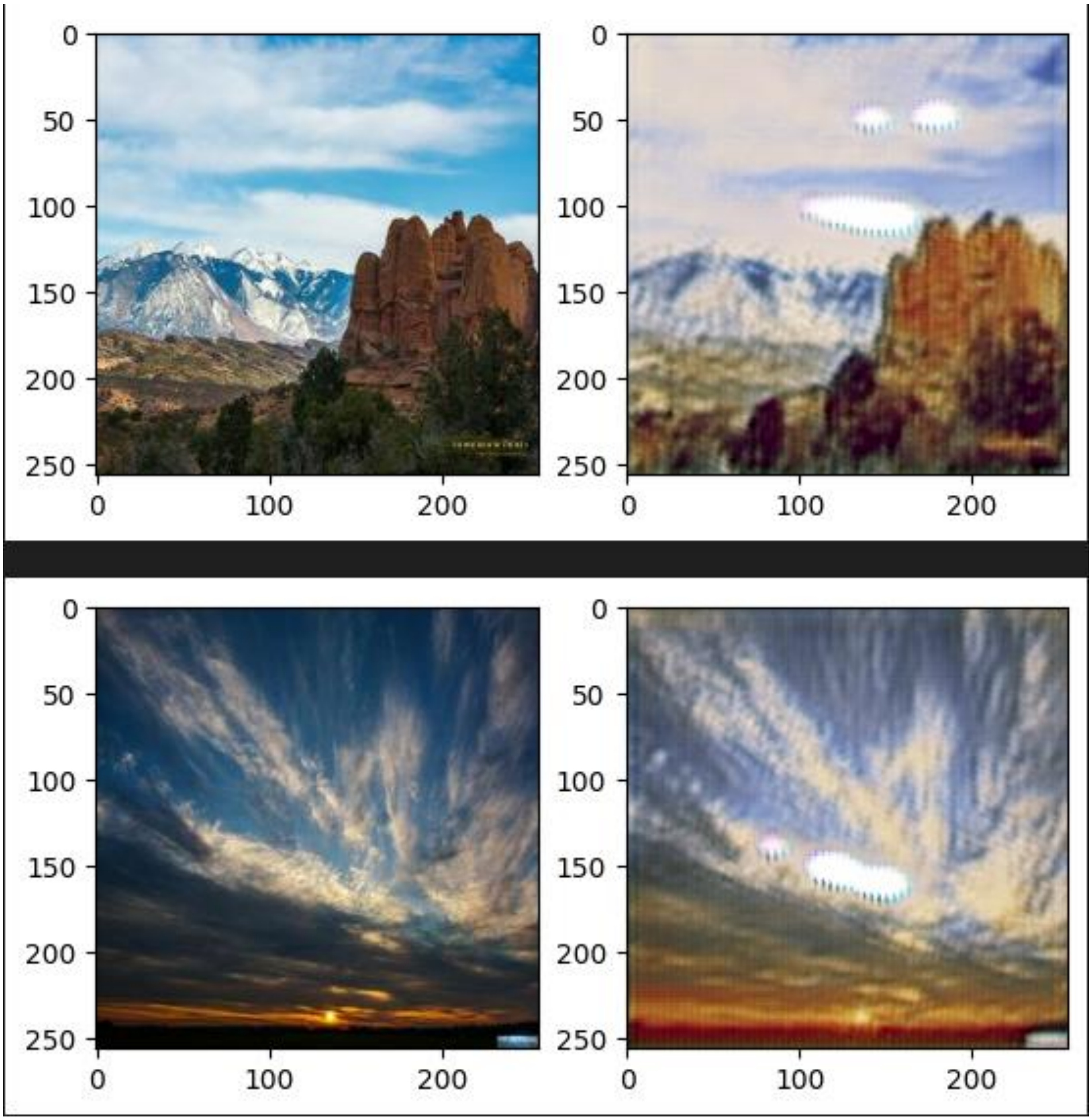


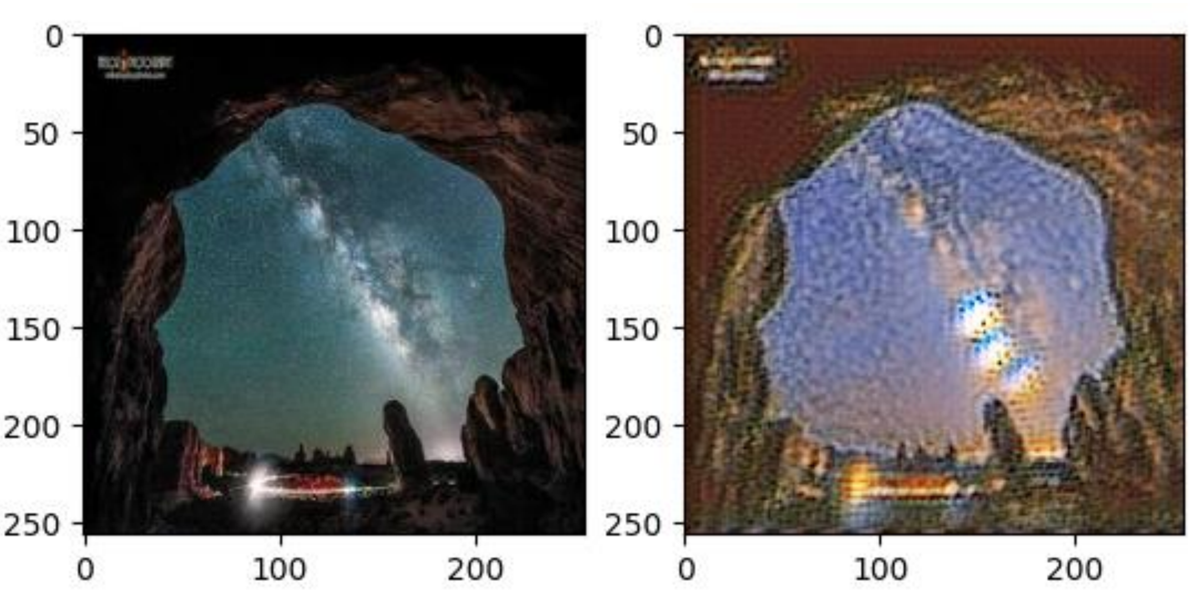
layers

The model architecture is very complex, and extensive search of all layers modification is not possible. yet we have tried adding another convolution layer to the discriminators, while decreasing kernel size of layers before. It showed no significant improvement, so we decided to use the version with the lower number of layers in sack of simplicity

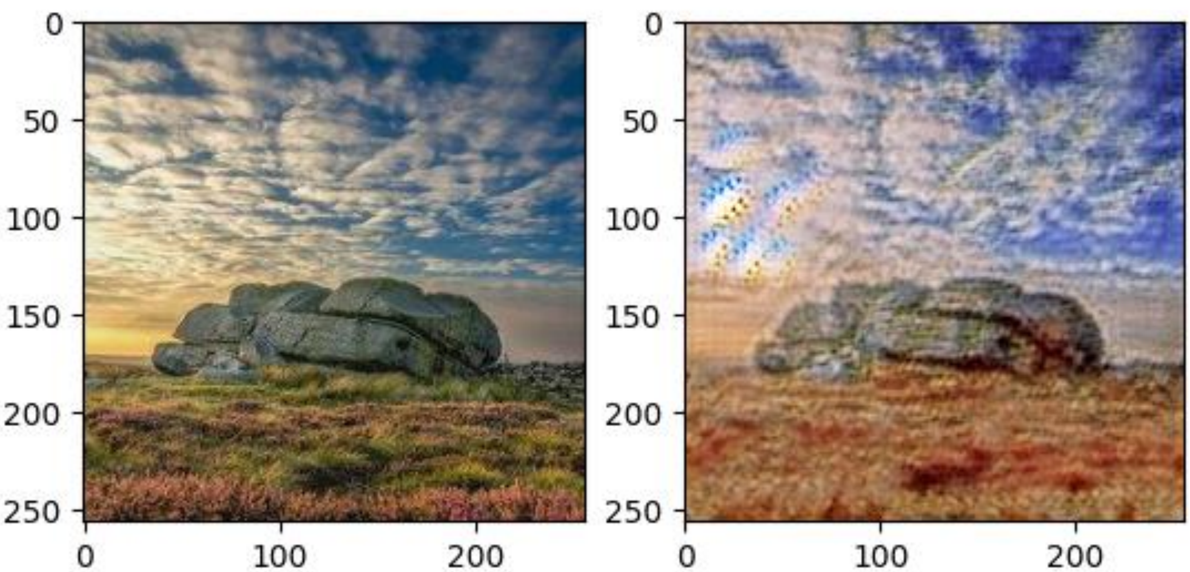
cycle gan output

Here we present output examples of the cycle Gan model, after trained for 1 epoch on 7038 image pairs:






```
WARNING:matplotlib.image:Clipping input data to the valid range for imshow wi
```



Summary and conclusion

We introduce research of generative models to change image styles, in our case to Monet style- but these models can be elevated into many other style transfers, and we predict this will continue to be prominent research field in the future with application to many other generative fields.