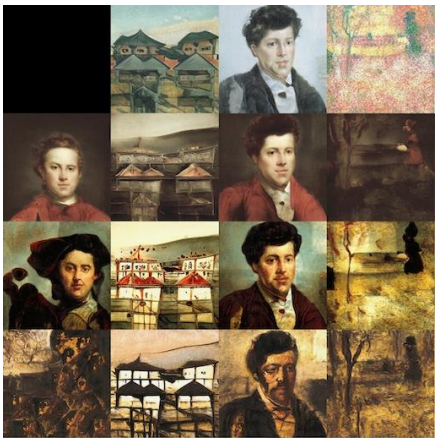
# Neural Networks for Images - Exercise #4

## Submission Date - 30/6/2022

**Programming Task:** Style Manipulation

In this exercise you will utilize a pre-trained StyleGAN model to manipulate the style of the images generated by it.

As we have seen in the lecture, one of the ways to characterize the style of an image is by extracting a set of Gram matrices, computed from activations resulting from feeding it image through a pre-trained feature extraction network. On the other hand, we have also seen that a target style may be induced by modifying the statistics of activation maps of a decoder/generator via the AdaIN mechanism, which is heavily used in the design of the StyleGAN generator.



Given a pretrained StyleGAN generator, and a *style input* image (possessing the **target style**), your goal is to modify the style of a *content input* image (which can be a synthetic image generated by the generator, or a real image that has been inverted into the generator's latent space) to match the target style. As usually done in style transfer tasks, the larger scale content of the content input should remain relatively unchanged. As an example, see the grid on the right, where the top row shows three different content inputs and the leftmost column shows three different style inputs.

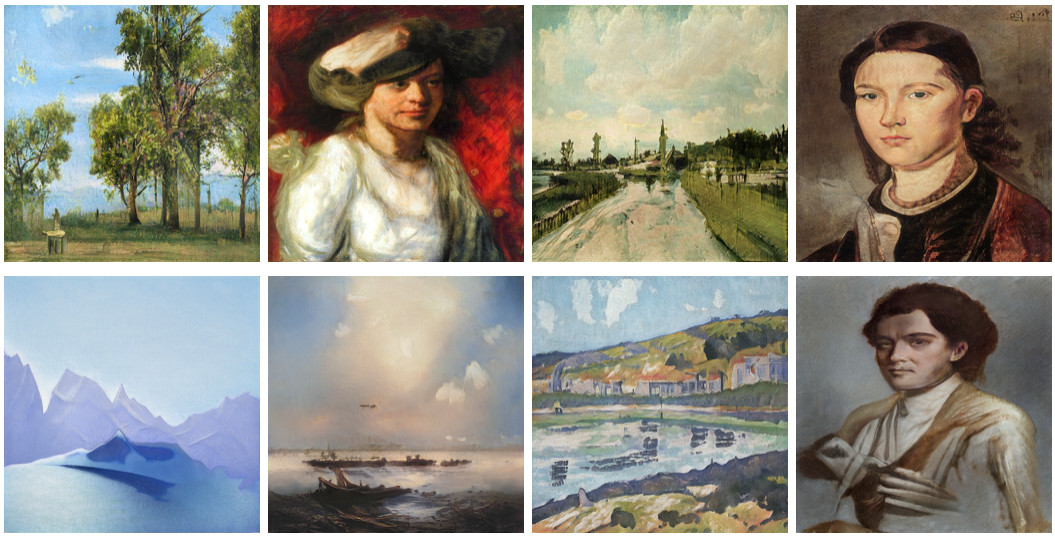
There are different ways to accomplish this task:

- If the style input can be faithfully inverted into the generator's latent space, one may attempt to achieve the style transfer using the style mixing mechanism (you'll need to experiment to decide which levels should be taken from the style input and which from the content input.
- Another option, that does not require inverting the style input, is to perform optimization in the latent space. Here, there are several choices as to which latent space to use. A common choice is the W+ extension of the intermediate latent space W, (a space containing multiple w latent vectors, one for each generator layer), or alternatively in the StyleSpace S (the space of all the style parameters). The loss may be evaluated using the loss network idea described in class. The latent optimization would need to be performed for each image.

- When performing latent optimization, one might consider using different losses, for example using the content loss and the Gram-based style loss, as was done by Johnson et al. Another option is to use the content loss together with AdaIN-based style loss, as was used by Huang et al. Implementing both kinds of losses is optional, you may choose one of the two.

~~Your task would be to experiment with these three different alternatives (style mixing, optimizing in W+, optimizing in S),~~
Your task would be to experiment with style mixing, and optimizing in W+, compare the results, and discuss them. Optimizing in S is optional and is left as a bonus, to receive points on the bonus you should submit it by the 23rd at 23:59. In the submission PDF provide results for 2 different content+style inputs that showcase your results and support your findings. Ideally you should compare how each approach acts on the same content+style inputs (i.e., have a fair comparison based on the same inputs). However, using the same inputs for some methods may not be possible, in such cases you can use other inputs (e.g., if one method takes in a real image while another uses a synthetic image as input.)

In this exercise you will be using a pre-trained GAN generator. We recommend using the official Nvidia pytorch implementation of StyleGAN2-ADA. A variety of pretrained StyleGAN2 models may be found here. Since the style inputs are typically famous paintings, it is recommended to use a StyleGAN generator that is capable of generating images in a compatible domain, such as this model pretrained on WikiArt. This pretrained model generates large images (1024x1024), which works fine in colab (at least when generating a small batch).
**Update**: we have trained a smaller model (256x256), which you may download from here. It's not perfect (but neither was the larger model), but you should be able to find decent images to play with. Try the following seeds, for example: 15, 21, 31, 83, 117, 139, 145.



As in previous exercises, in each of your experiments you will need to implement the network or test, rationalize the results and document your conclusions as to what happened and why. Your report should be submitted as a pdf file, as explained in the Submission Guidelines below.

**We expect you to report and elaborate on every practical task in the report, using your own words and your own analysis of what you've done. Include everything that you think is crucial for us to understand your way of thinking.**

For this exercise the submission will be done in **pairs.** Please submit a single zip file named "ex4_ID1_ID2.zip". This file should contain your code, along with an "ex4.pdf" file which should contain the description of your experiments with figures and analysis. Furthermore, include in this compressed file a README with your names, IDs, and CSE usernames.
Please write readable code, with documentation where needed, as the code will also be checked manually.