Standard Neural Style Transfer with Color Preservation optimization

Nikola Zubić
Artificial intelligence Laboratory, Department of Computer Science, Faculty of Technical Sciences, University of Novi Sad

ABSTRACT

Neural Style Transfer is a technique in Deep Learning that recieves two images as input. We will refer to one image as the content image (labeled with „C") and the other one we will call a style image (labeled with „S").

We aim to output the generated image (labeled with „G") so that G combines the content of C image with the artistic style of image S.

By using this approach we can, for example, generate our selfie in a style of the famous Serbian painter Paja Jovanovich, but without using the explicit pixel transformations (such as non-photorealistic rendering). Instead of explicit transformation, we are „learning" pixels for the G image.

Solution of the Neural Style Transfer problem can be used as subsystem of more complex system (such as FotoSketcher) or any filter application that is associated with photo editing.

CONTACT

Nikola Zubić
Faculty of Technical Sciences, Novi Sad
Email: nikolazubic97@yahoo.com
Website: github.com/NikolaZubic

INTRODUCTION

Original paper uses pre-trained Convolutional Neural Network (CNN) and solves a given problem based on it.

In this project we used VGG-19 neural network architecture (image on the right).
Substantially, approach is based on the minimization of the error function:

$$\mathcal{L}_{total}(\vec{p},\vec{a},\vec{x}) = \alpha\mathcal{L}_{content}(\vec{p},\vec{x}) + \beta\mathcal{L}_{style}(\vec{a},\vec{x})$$

where α and β are the weighting factors for content and style reconstruction respectively.

By minimizing the function above, our G image is produced in a manner in which its content will be similar to C image and its artistic style to S image.

We can directly visualize information that our network "learned" in current layer by reconstructing image up to that layer. Initial layers of the network, at the beginning of training process reproduce same pixel output for given input, and in deeper layers they start to blur pixels more. At the end, this blurring/modification process is expanded to complete regions of the image. We reconstruct the input image from layers 'conv1_1', 'conv2_1', 'conv3_1', 'conv4_1' and 'conv5_1' of the original VGG-network.

NEURAL NETWORK ARCHITECTURE

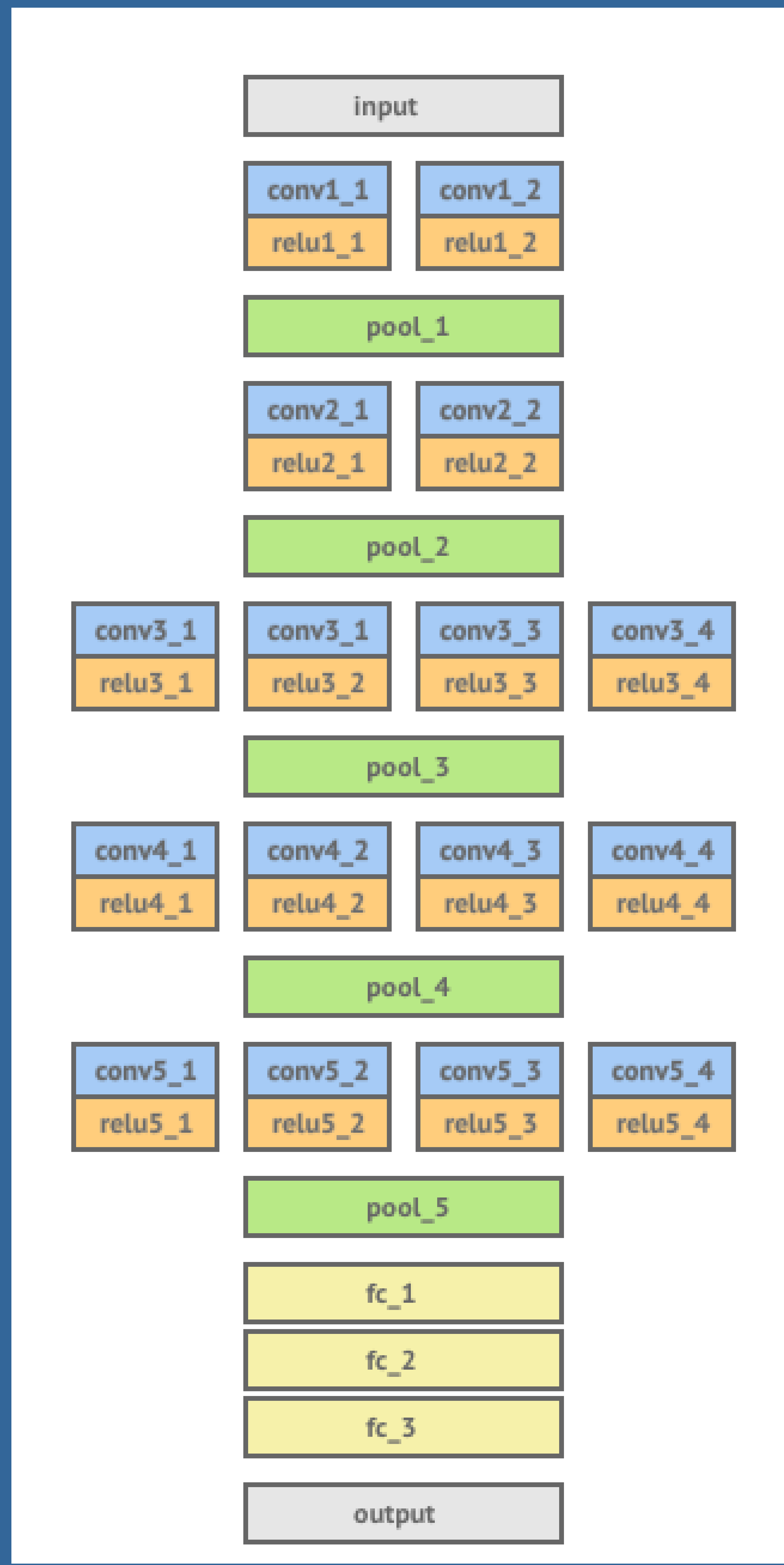
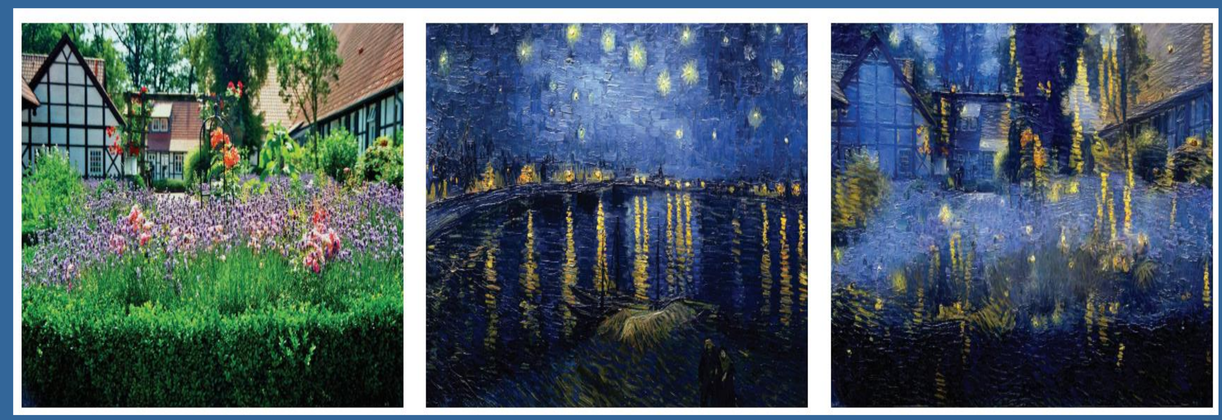


Figure 1. VGG-19

RESULTS without optimization



EVALUATION

By assuming that frequent problem when evaluating soft computing and fuzzy systems is the creation of proper metric for assessment of their quality, this project was evaluated by comparing it with scientific papers and other projects that solved the same problem, and also, by subjective visual perception.

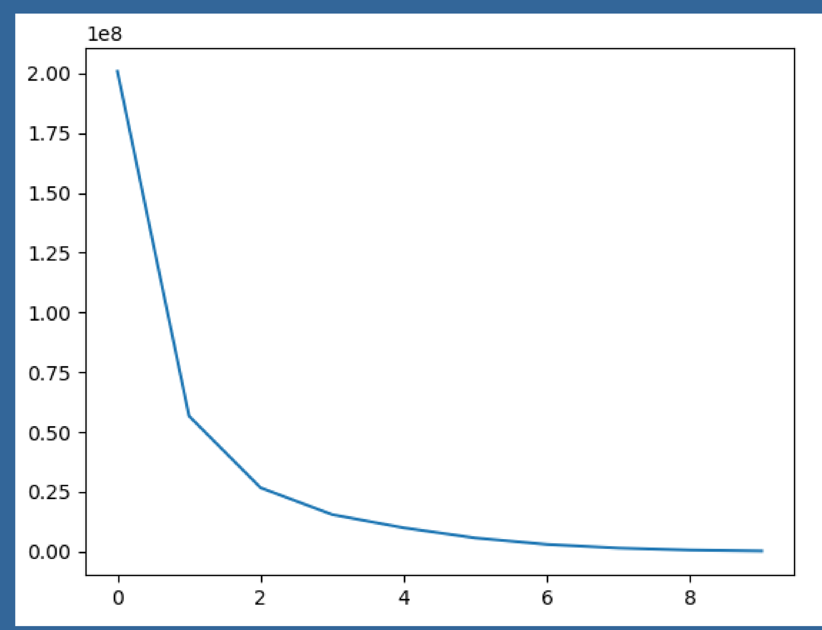


Figure 2. Loss function through time steps

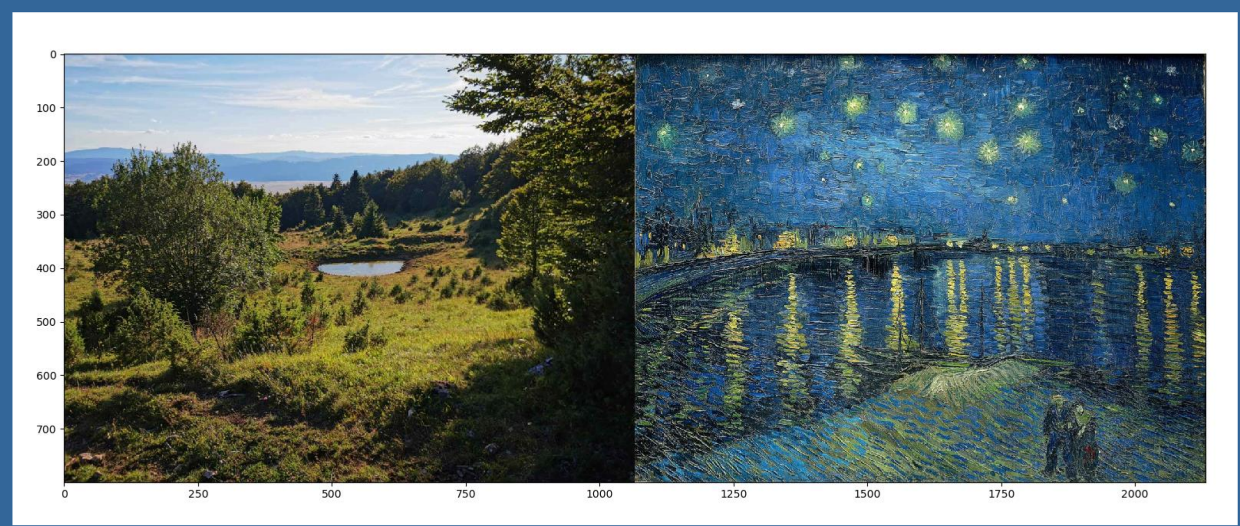
COLOR PRESERVATION METHODS

The main problem with standard NST technique is that we can't effectively preserve colors on G image from C image. When we have C and S images as inputs, when we generate G image, its content is preserved, but dominant colors/palettes are taken from S image. Color Preservation enables us to preserve content of C image, but also the colors from it and only style characteristics (texture, luminosity, etc.) are like in S image.

This optimization can be done by using one of these two approaches:

1. *Color Histogram Matching*: Given the S and C images, the S image's colors are transformed to match the colors of the content image. This produces a new style image S' that replaces S as input to the NST algorithm. Color transfer procedure that we used is based on linear methods, where solutions are produced by using Cholesky decomposition.
2. *Luminance-only transfer:* Perform style transfer only in the luminance channel. This is motivated by the observation that visual perception is far more sensitive to changes in luminance than in color.

RESULTS with optimization



CONCLUSION

Since the problem of standard NST was solved by using Color Preservation methods we are going to discuss their potential disadvantages.

Color Histogram Matching is naturally limited by how well the color transfer from the content image onto the style image works. The color distribution often cannot be matched perfectly, leading to a mismatch between the colors of the output image and that of the content image.

Luminance-only transfer perfectly preserves the colors of the content image. However, dependencies between the luminance and color channels are lost in the output image. This is particularly apparent for styles with prominent brushstrokes. As the result of this colors are no longer aligned to strokes and single brushstroke can have multiple colors, which is nonexistent in real paintings.

In the future, it would be interesting to explore how these two models might be unified, and also to explore more sophisticated methods that can work very well with texture transfer, because standard NST has slight problems with it.