Computer Vision Project

Photorealistic Style Transfer

Project Information

- Github Repository: Link
- Reference Paper: Deep Photo Style Transfer (link here)
- Team Members:
  - Shahbaz Shaik : 2018111025
  - Jayant Duneja : 2018102003
  - Tanvi Kamble : 2018114004
  - Nikhil E: 2018114019

TABLE OF CONTENTS

Approach - Base Neural Style Algorithm

The Neural Style algorithm that the paper attempts to improve upon has the objective function:

$$\mathcal{L}_{total} = \sum_{l=1}^{L} \alpha_l \mathcal{L}_c^l + \Gamma \sum_{l=1}^{L} \beta_l \mathcal{L}_s^l$$

Where,

$$\mathcal{L}_c^l = \frac{1}{2N_l D_l} \sum_{ij} (F_l[O] - F_l[I])_{ij}^2$$

$$\mathcal{L}_s^l = \frac{1}{2N_l^2} \sum_{ij} (G_l[O] - G_l[S])_{ij}^2$$

L is the total number of convolution layers, and l is used to indicate the l-th convolutional layer of the deep convolutional neural network. There are N_l filters in each layer, and each has a vectorized feature map of size D_l. $F_l[.]$ is the feature matrix, and G_l is the Gram matrix $G_l[.] = F_l[.]F_l[.]^T$. α_l and β_l are the weights to configure the layer preferences, with □ being a weight that balances content and style.

Approach - Augmentation to Neural Style

We define a new loss, $\mathcal{L}_m$, that we use to penalise outputs that cannot be explained by a locally affine transform. This is built upon the Matting Laplacian.

$$\mathcal{L}_m = \sum_{c=1}^{3} (V_c[O]^T M_I V_c[O])$$

Here, we have M_I as a matrix that depends only on the inputted image, I, and is of dimensions (m*n)x(m*n), where m and n are the dimensions of the image. $V_c[O]$ is the vectorized version of the output image O in channel c with dimensions (m*n)x1.

Approach - Augmented Style Loss

We define an augmented style loss, $\mathcal{L}_{s+}$. Segmented masks are added as additional channels to the input image and the augmented style loss is defined as:

$$\mathcal{L}_{s+}^{l} = \sum_{c=1}^{C} \frac{1}{2N_{l,c}^{2}} \sum_{ij} (G_{l,c}[O] - G_{l,c}[S])_{ij}^{2}$$

$F_{l,c}[O] = F_l[O]M_{l,c}[I]$, $F_{l,c}[S] = F_l[S] = F_l[S]M_{l,c}[S]$, where we define C as the number of channels within the segmentation mask, $M_{l,c}[.]$ denotes the channel c of the segmentation mask in layer l and $G_{l,c}[.]$ is the Gram matrix corresponding to the newly calculated $F_{l,c}[.]$

Approach - Augmented Objective Function

Combining the components that have been defined previously, we defined a new objective function for photorealistic style transfer, as:

$$\mathcal{L}_{total} = \sum_{l=1}^{L} \alpha_l \mathcal{L}_c^l + \Gamma \sum_{l=1}^{L} \beta_l \mathcal{L}_{s+}^l + \lambda \mathcal{L}_m$$

L is the total number of convolution layers, and l is used to indicate the l-th convolutional layer of the deep convolutional neural network. α_l and β_l are the weights to configure the layer preferences, with □ being a weight that balances content and style, and λ is the weight that we use for controlling photorealism regularisation. $\mathcal{L}_c^l$, $\mathcal{L}_{s+}^l$, $\mathcal{L}_m$ are the content loss, augmented style loss, and photorealism regularisation.

Approach Overview

- The algorithm takes in two images, an input image and a reference image, from which the style is to be extracted and applied to the input image.
- The proposed approach improves upon an a pre-existing Neural style algorithm has been augmented by introducing 2 core ideas:
    - In order to reduce distortions, a photorealism regularization term is introduced in the objective function, enabling the output image to be represented by locally affine color transformations of the input.
    - Introduction of an optimal guidance to the style transfer process, to prevent problems related to content-mismatch and to improve the photorealism of the results.

Input Image and Style

Output Images varying the weight of the photorealism regularization (Increasing photorealism)

Matting Laplacian

- The Matting Laplacian has its roots in the problem of image matting; in our approach, we use the Matting Laplacian to constrain the transformation from the input to the output to be locally affine in colorspace.
- In some other style transfer approaches, many painting like effects were noticed. These effects were prevented by preventing spatial distortions, and restricting the style transfer operation to only the color space of the image.
- For this, a transformational model that is locally affine in colorspace, which is expressed as a differentiable energy term inspired by the Matting Laplacian, was used.

Losses

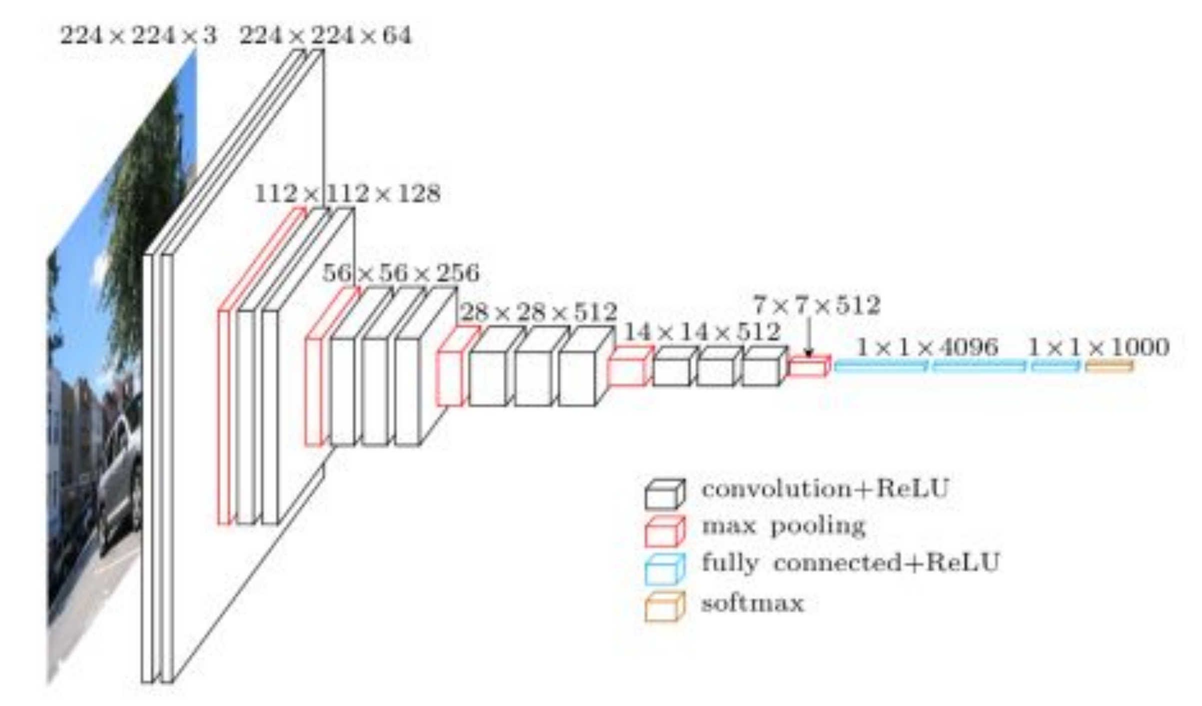
We have defined 4 losses to optimise our Loss function.

- Content Loss: This function is essentially the mean squared error between the feature maps produced by the generated output image and the content image since the two images should have similar feature maps in the higher layers.
- Augmented Style Loss: This is a modified style loss function. For this, we use segmentation masks concatenated on different channels, to generate image segmentation. It miscalculated as the mead square difference between the Gram Matrices of the feature Maps of masked output and style images.

Losses-Continued

- Affine Loss: Using the Matting Laplacian, that has its origins in alpha matting, and the output image, we define a loss called Affine Loss that penalizes output images which are not well explained by locally affine transformations.
- Total-Variation Loss: The total variation loss is the sum of absolute differences for neighbouring pixel-values in the input images.This is used to measure how much noise is present in the images. We have used this as a Loss function during Optimization to suppress noise in images.

VGG-19



224 × 224 × 3
224 × 224 × 64
112 × 112 × 128
56 × 56 × 256
28 × 28 × 512
14 × 14 × 512
7 × 7 × 512
1 × 1 × 4096
1 × 1 × 1000
convolution+ReLU
max pooling
fully connected+ReLU
softmax

VGG-19

- VGG-19 is a Convolutional Neural Network which has 19 layers.
- It has been trained using more than 1 million images from the ImageNet Database
- We have used the VGG-19 as a pre-trained model and fine tuned that model by training on our Dataset.
- We downloaded the pretrained VGG-19 weights that they could be fine-tuned.
- This model helps us calculate the style and loss.

Future-Goals

- Combining the defined functions and the losses, to achieve the desired output.
- Testing the final product on various images and comparing obtained results upon varying parameters

THANK YOU