Mid Evaluation

Photorealistic Style Transfer via Wavelet Transforms

Team : Framed

Method Overview

➔ Use the **VGG19** model’s feature extraction layers as the encoder and its corresponding mirror as the decoder

➔ Replace the max pooling and unpooling layers with **Haar Wavelet Pooling**.

➔ Apply **WCT** after each scale (conv1 X, conv2_X, conv3 X and conv4 X). This is referred to as **progressive stylisation**.

Progress so far :

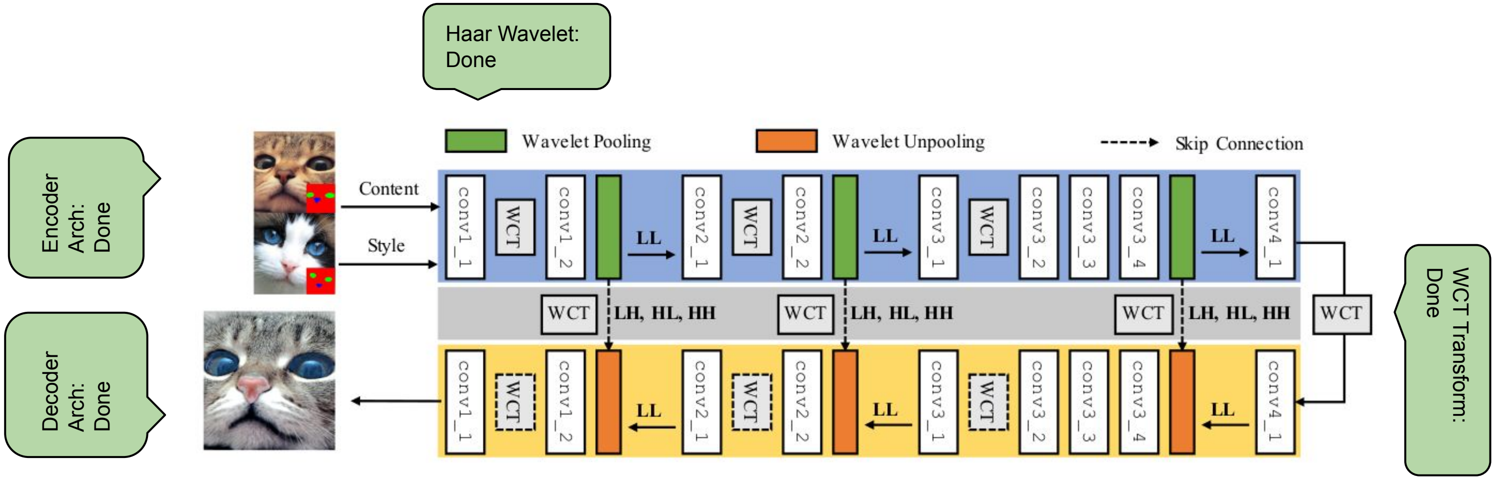


Figure 11: Overview of the proposed progressive stylization. For the encoder, we perform WCT on the output of `convX_1` layer and skip connections. For the decoder, we apply WCT on the output of `convX_2` layer, which is optional.

Upcoming tasks

Todo: Write transfer wrapper function

Todo: Assemble architecture

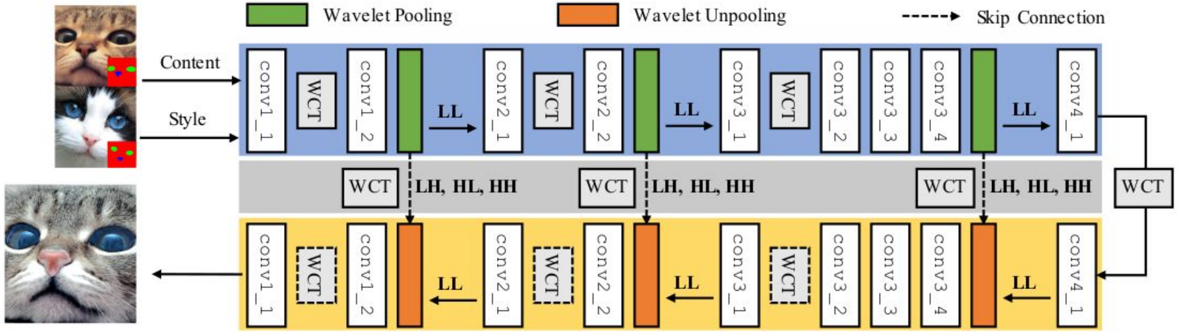


Figure 11: Overview of the proposed progressive stylization. For the encoder, we perform WCT on the output of `convX_1` layer and skip connections. For the decoder, we apply WCT on the output of `convX_2` layer, which is optional.

Code Written for :

- Haar wavelet transform
- Encoder-Decoder architecture
- WCT

Work for the Next Phase

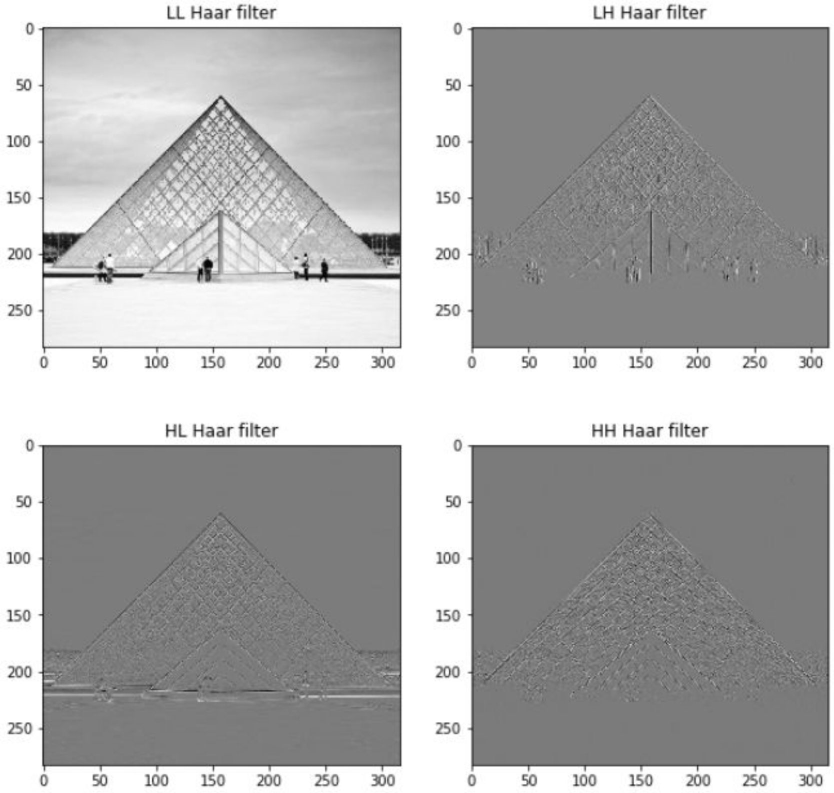
- Integrate the individual components of the architecture
- Write a wrapper function to perform the end-to-end style transfer pipeline
- Perform Experiments using the several examples of content and style images

Intermediate Results

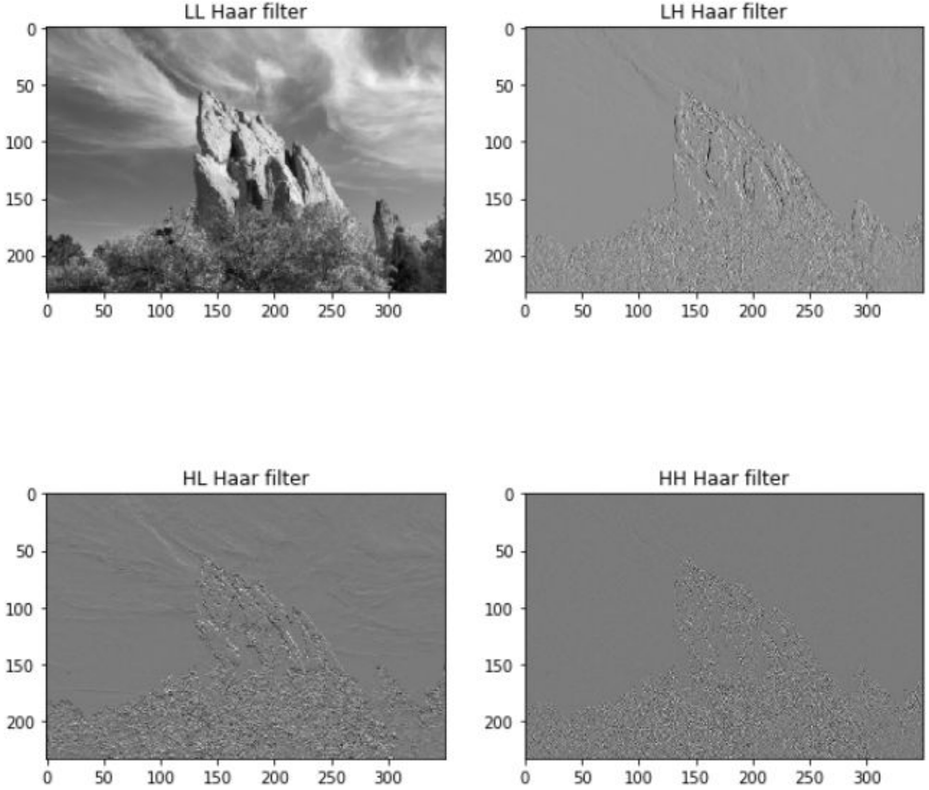
Haar-wavelet filter

- To test the working of the Haar wavelet convolution filters, we generate the 4 kernels - LL, LH, HL, HH as mentioned in the paper.
- We apply the wavelet filters to two experiment images and show our observations in the next slides.
- It is observed that :
    - The LL filter convolution has little to no change on the image. This is because it is a low pass filter and captures only the lower frequencies in the image
    - The HH filter on the other hand captures the higher frequency, intricate edges of the image as we can see in the case of the Louvre museum image’s glass edges

Experiment Image : 1



Experiment Image : 2



Testing encoder-decoder with image reconstruction

- For the encoder, we have used the feature extraction layers (upto conv4_x) as the encoder and its corresponding mirror architecture as the decoder.

- The authors then trained this architecture for image reconstruction, i.e, given an image as the input, the encode should encode the image and the decoder should output the original image.

- In order to test if our implementation of the encoder-decoder is working, we shall see if the above case is working.

- We will also see how the architecture performs if we remove certain layers.

All Layers included (conv1_x, conv2_x, conv3_x, conv4_x)

Experiment Image : 1

Input Image

Output Image

We observe that the reconstructed image is very similar to the input image but is slightly blurrier.

All Layers included (conv1_x, conv2_x, conv3_x, conv4_x)

Experiment Image : 2

Input Image **Output Image**

We observe that the reconstructed image is very similar to the input image but is slightly blurrier.

Last layer omitted (conv1_x, conv2_x, conv3_x)

Experiment Image : 1

Input Image **Output Image**

We observe that the reconstructed image is mostly gray, however some resemblance of the original image is seen.

Last layer omitted (conv1_x, conv2_x, conv3_x)

Experiment Image : 2

Input Image **Output Image**

We observe that the reconstructed image is mostly gray, however some resemblance of the original image is seen.

WCT Procedure

- We first extract the features for both the content (cF) and style images (sF).

- We now apply WCT (Whitening and Coloring Transforms).

- We take the output of WCT and pass it to the next layer of the architecture.

Whitening and Color Transforms

Whitening Transform

- Center f_c by subtracting its mean vector.

- Transform f_c linearly such that the feature maps are uncorrelated.

$$\hat{f}_c = E_c\, D_c^{-\frac{1}{2}}\, E_c^\top\, f_c$$

- D_c is a diagonal matrix with the eigenvalues of the covariance matrix. E_c is the corresponding orthogonal matrix of eigenvectors.

Color Transform

- Center f_s by subtracting its mean vector.

- Carry out the coloring transform to obtain f_{cs} .

$$\hat{f_{cs}} = E_s\, D_s^{\frac{1}{2}}\, E_s^\top\, \hat{f}_c$$

- D_s is a diagonal matrix with the eigenvalues of the covariance matrix. E_s is the corresponding orthogonal matrix of eigenvectors.

Experiment Image : 1

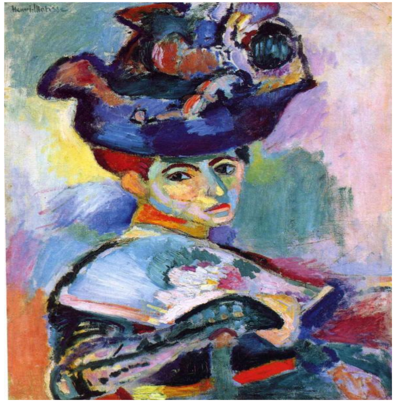
Style Image

Content Image

With both VGG and Decoder fixed, and given the content image C and style image S, our method performs the style transfer through whitening and coloring transforms.

Single-level Stylization

Experiment Image : 2



Style Image

Content Image

Single-level Stylization