Photorealistic Style Transfer via Wavelet Transforms

Team : Framed

Objective

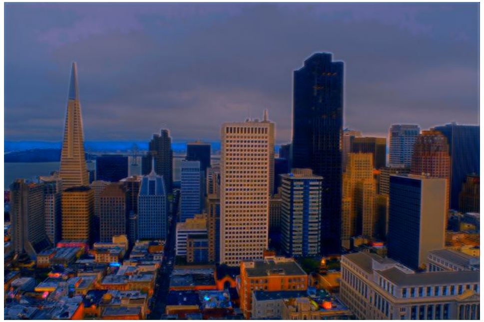
- Given a pair of images - S : Style image and C : Content image, perform photorealistic style transfer

- Leveraging wavelet transform to overcome limitations of spatial distortions, and introduction of unrealistic artifacts in the final image.

- The paper proposes to perform this via an end-to-end photorealistic style transfer model that allows to remove the additional post-processing steps

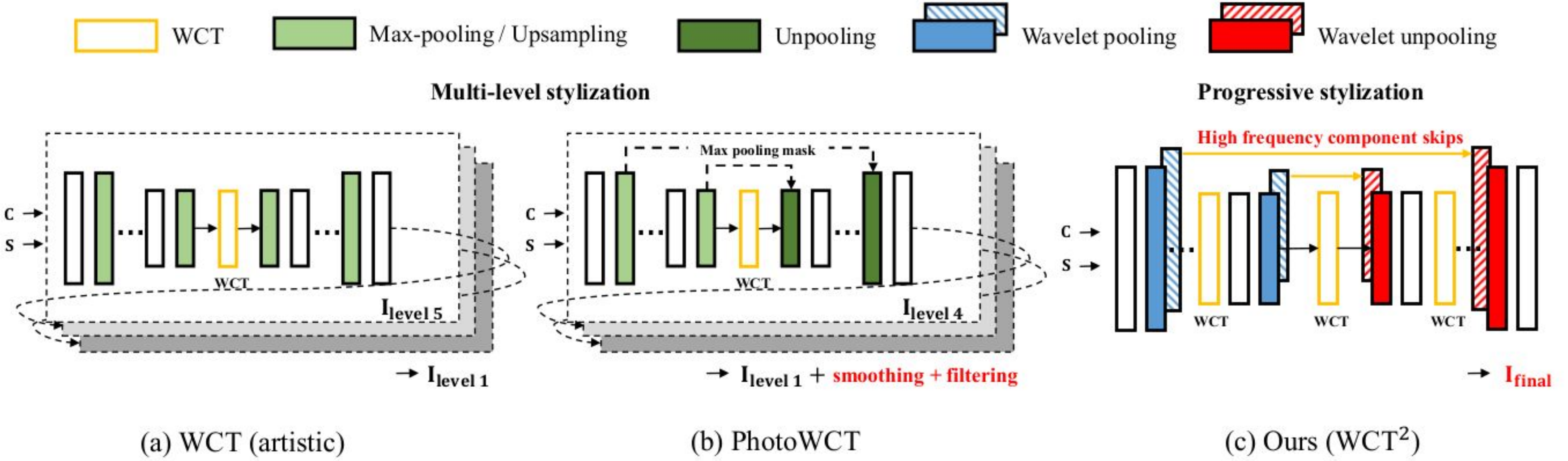
Content Image (C)

Style Image (S)



Style Transfer Output

How WCT^2 is Different



Goals

- Given a photograph as a reference style, transfer the style to a photo so that the stylized photo preserves the content of the original photo but carries the style of the reference photo.

Deliverables (March)

- Implement first part of Wavelet Corrected Transfer i.e. Haar Wavelet Pooling and Unpooling by March.

- Design model architecture of pre-trained VGG-19 network.

Method

Overview of the method

- ➔ 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**.

VGG-19 Model

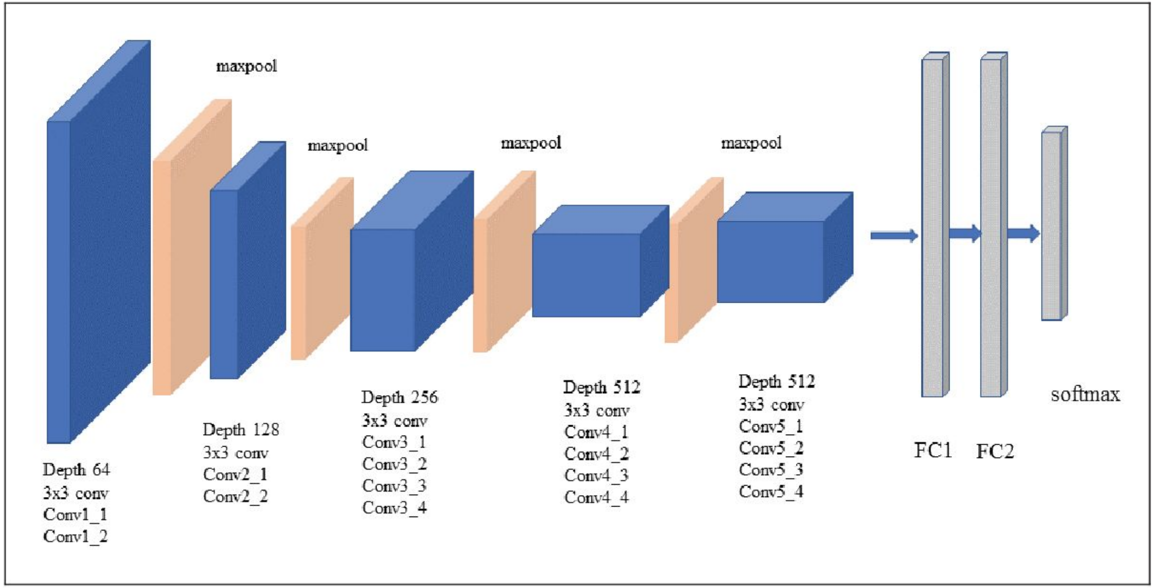


Fig. 3. VGG-19 network architecture

- For our use case, we use the layers upto conv4_1 for our encoder. We use the mirror of this model as our decoder.

WCT Procedure

- We first extract the features for both the content (c) and style (s) images. Let us denote this as f_c and f_s.
- We now apply WCT (Whitening and Coloring Transforms).
- We take the output (denoted by f_{cs}) of WCT and pass it to the next layer of the architecture

Whitening and Color Transforms

Whitening Transform

- We first center f_c by subtracting its mean vector.
- Then we 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

- We first center f_s by subtracting its mean vector.
- We now carry out the coloring transform as follows 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.

Haar Wavelet Pooling

- Haar wavelet pooling has four kernels, $\{LL^T\ LH^T\ HL^T\ HH^T\}$
- The low (L) and high (H) pass filters are-

$$L^{\top} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \end{bmatrix}, \quad H^{\top} = \frac{1}{\sqrt{2}}\begin{bmatrix} -1 & 1 \end{bmatrix}.$$

- Here, the low-pass filter captures smooth surface and texture while the highpass filters extract vertical, horizontal, and diagonal edgelike information
- One important property of the wavelet pooling is that the original signal can be exactly reconstructed by mirroring its operation
- Thanks to this favorable property, our proposed model can stylize an image with minimal information loss and noise amplification.

Final Model

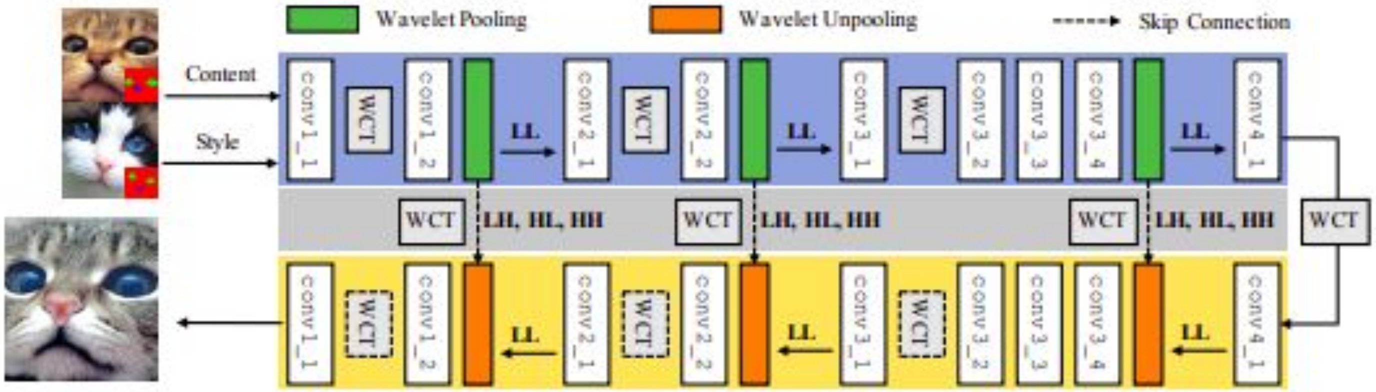


Figure 11: Overview of the proposed progressive stylization. For the encoder, we perform WCT on the output of `convX_1` layer and skip connections. For the decoder, we apply WCT on the output of `convX_2` layer, which is optional.