Style Transfer using Deep Neural Networks

Content

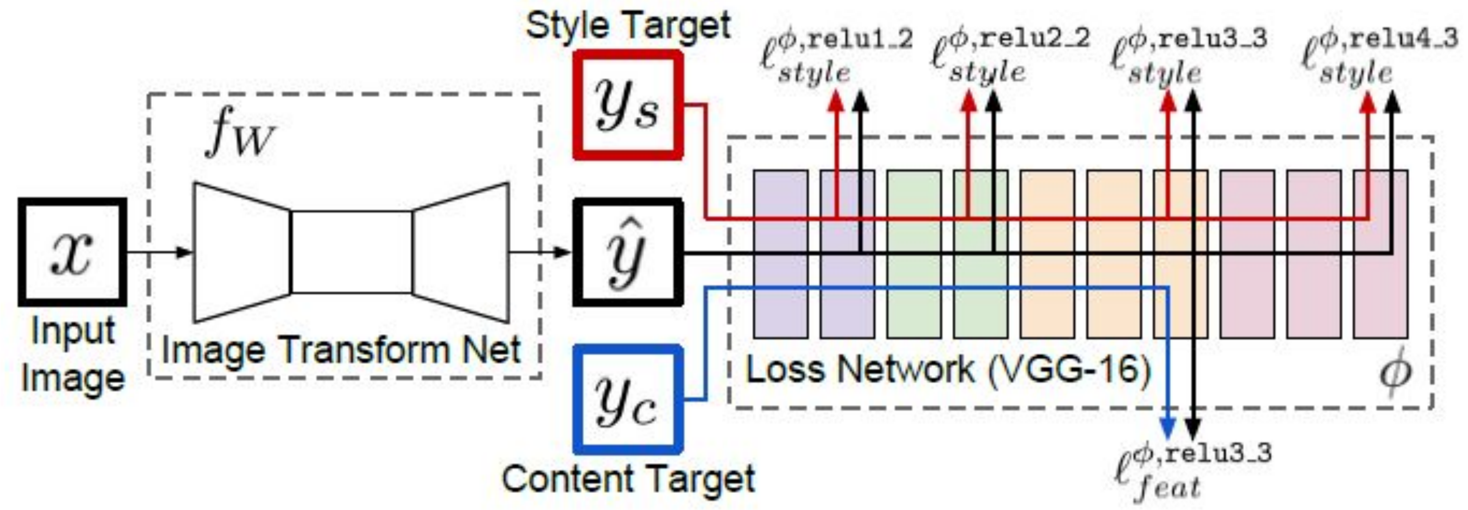
Problem Description

- Our aim is to re-interpret an image in the style of a given style image without losing too much content information of the input image.
- Thus, the problem address the trade-off between the loss of content information and gained style during training.

Methodology

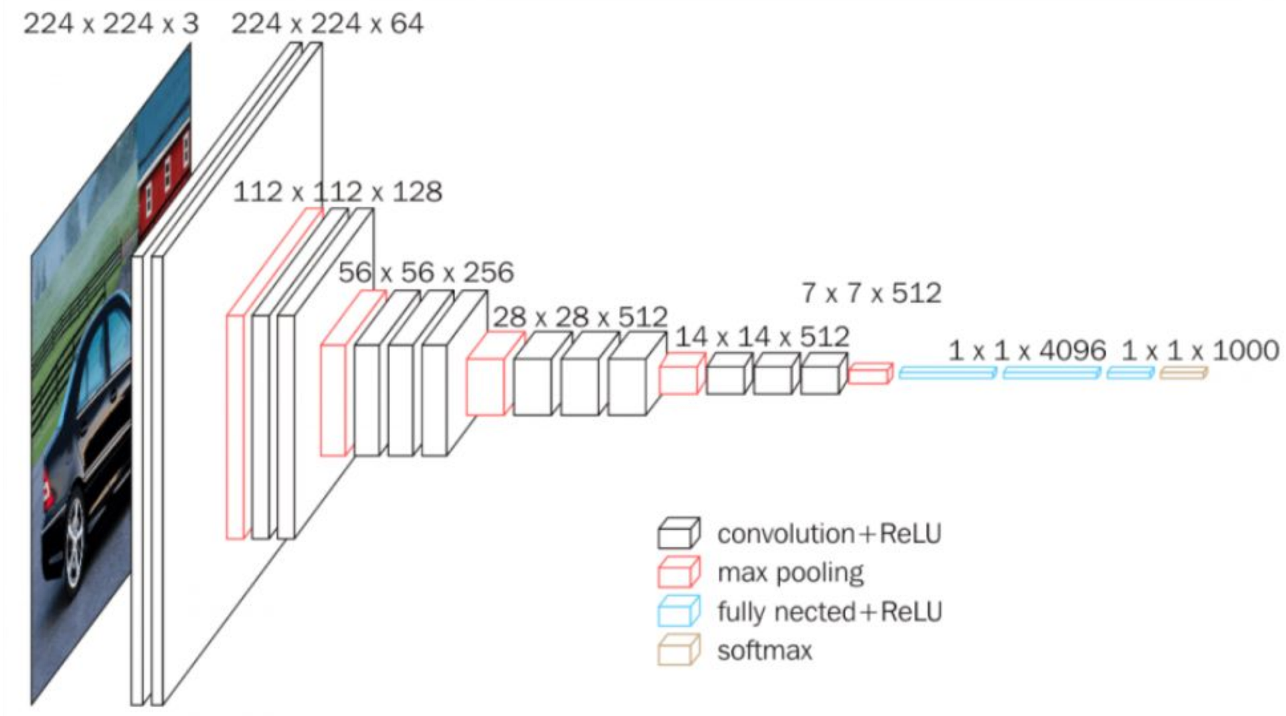
- The applied method relies on pre-trained network VGG16 for feature extraction where the features are used to implement a perceptual loss.
- Implemented loss is used for the training of deep style transformer network that trained with a specific style via back-propagation.
- Several models have been trained, i.e. for each style, on Flickr8K dataset.

System Overview



Feed-forward transformation networks with perceptual loss functions for style transformation.

Block diagram of pre-trained model VGG16



Mathematical background for style transformation

- vectorized version of extracted features ϕ_j and c_j is channel number

to computation of gram-matrices

$$\phi_j \;\rightarrow\; F_j \;:=\; [c_j,\; h_j,\; w_j] \;\rightarrow\; [c_j,\; h_j * w_j]$$

$$G_j := < F_j^T,\; F_j >$$

- Content Loss Function

$$L_{content}^{\phi_j}(y_c) = \frac{1}{C_j H_j W_j} ||\phi_j(\hat{y}) - \phi_j(y_c)||^2$$

- Computation of gram matricies

$$G_{c,c'}^{\phi_j}(x) = \frac{1}{C_j H_j W_j} \sum_{h=1}^{H_j} \sum_{w=1}^{W_j} \phi_j(x)_{h,w,c}\; \phi_j(x)_{h,w,c'}$$

Mathematical background for style transformation

- An element from computed gram matrix

$$G^{\phi_j}(x) = < F_j(x)^T,\ F_j(x) >$$

- Style loss

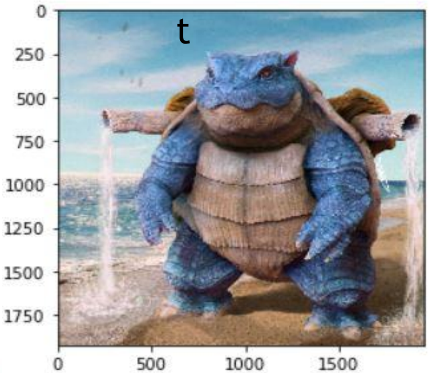
$$L_{style}^{G_j} = ||G_j(\hat{y}) - G_j(y_s)||^2$$

- Total loss which is weighted sum of content and style losses

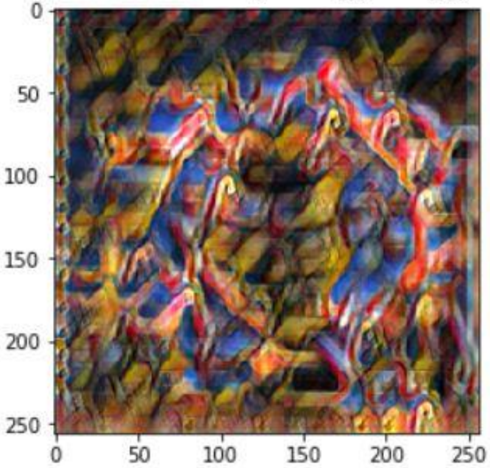
$$L_{total} = \alpha L_{content}^{\phi_2} + \beta(L_{style}^{G_1} + L_{style}^{G_2} + L_{style}^{G_3} + L_{style}^{G_4})$$
$$= \alpha L_{content}^{\phi_2} + \beta L_{style}^{G_{1,2,3,4}}$$

Applying style of Composition 7, by Vasiliy Kandinskiy, to realistic Pokemon image

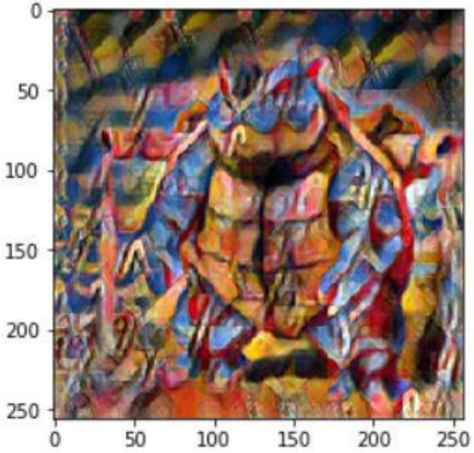
conten
t



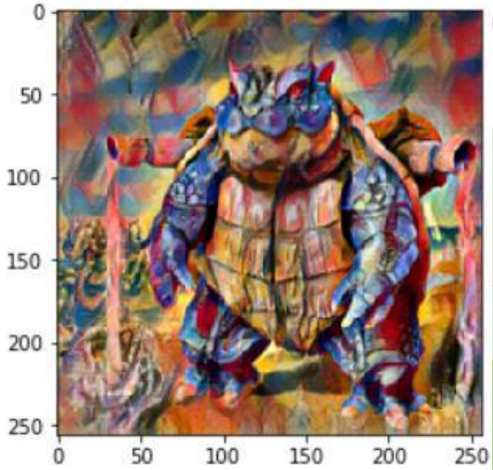
style



10%
training



20%
training



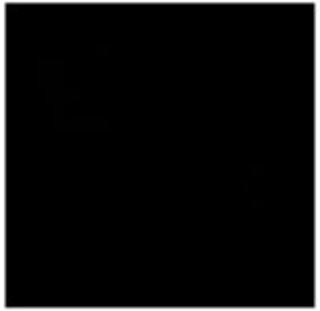
99%
training

Applying style of Starry Night, by Van Gogh, to Izmir Clock Tower

Content

Style

Initial



Epoch1

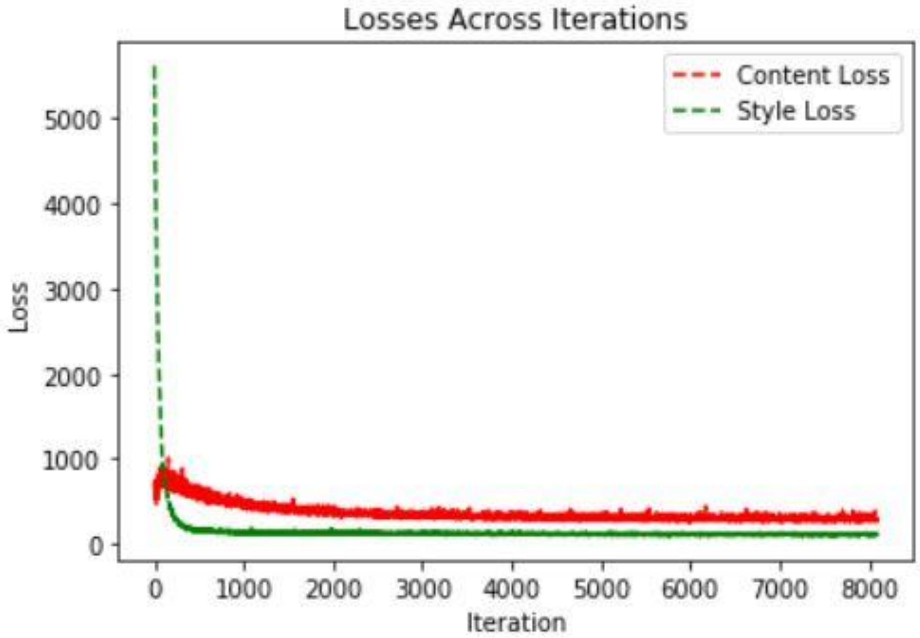
Epoch2

Epoch3

Epoch4

Epoch5

Model Performance for Van Gogh Style Transformation



Perceptual loss curves for starry night style transfer. The scales of hyper-parameters α and β are vivid on loss curves

Style transformation for Mosaic style on Big Ben

Content

Style

Initial

Epoch1

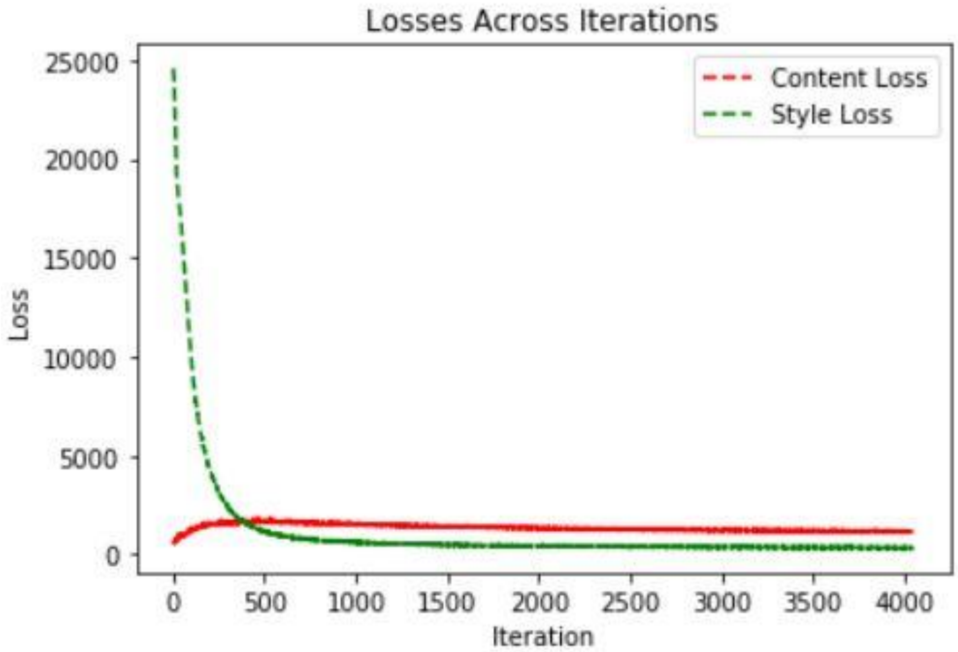
Epoch2

Epoch3

Epoch4

Epoch5

Model Performance for Mosaic Style Transformation



Picasso's Crying Women applied on Jon Snow from Game of Thrones

content

style

Initial

Epoch1

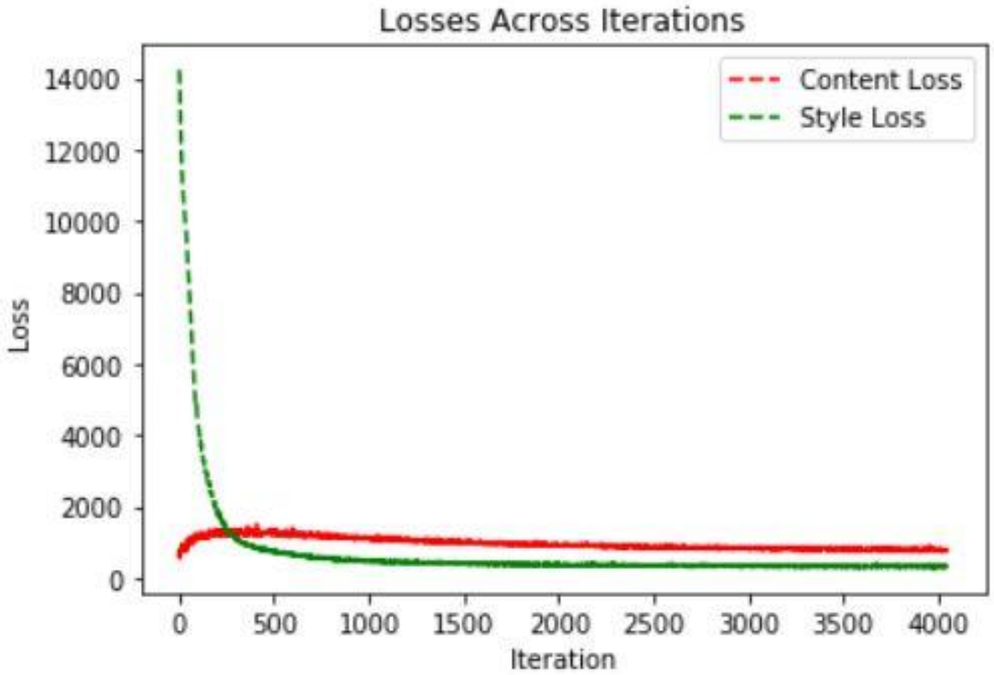
Epoch2

Epoch3

Epoch4

Epoch5

Model Performance for Picasso Style Transformation



Style Transformation for Pillar of Creations

content

style

Initial

Epoch1

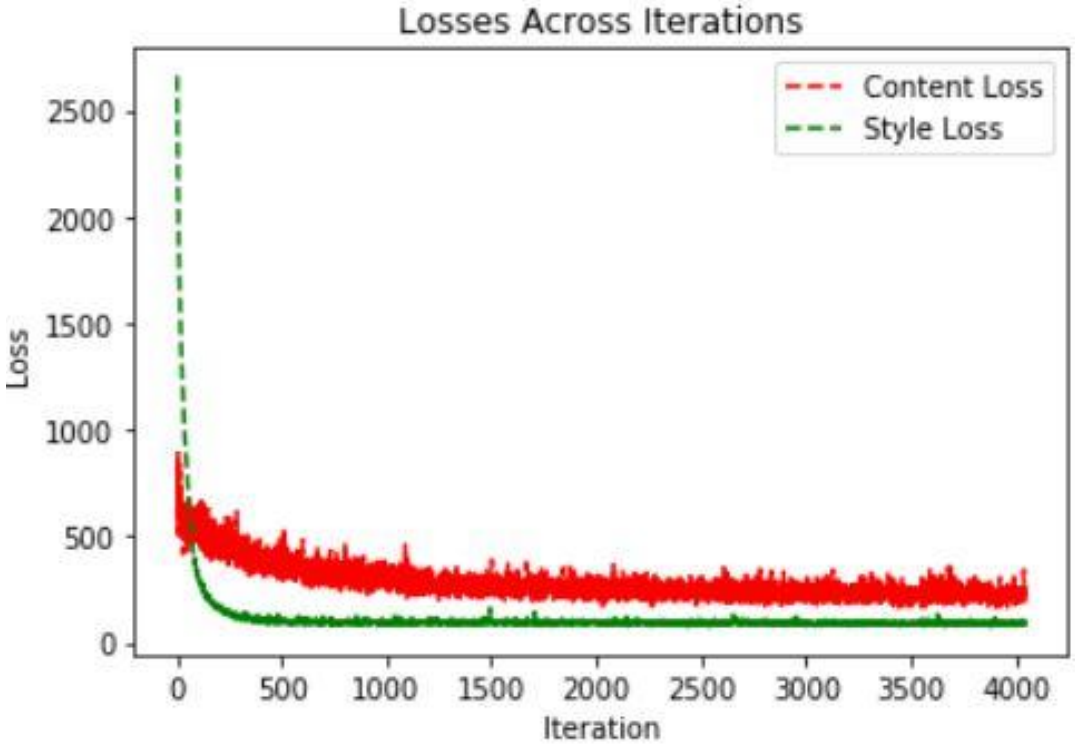
Epoch2

Epoch3

Epoch4

Epoch5

Model Performance for Pillar of Creations Style Transformation



Discussion on results

- Content image more adapts the style transformation through increasing the percentage of training.

Conclusion

- The results showed that, trained models could successfully apply the specific style to any input image.