A Neural Algorithm of Artistic Style

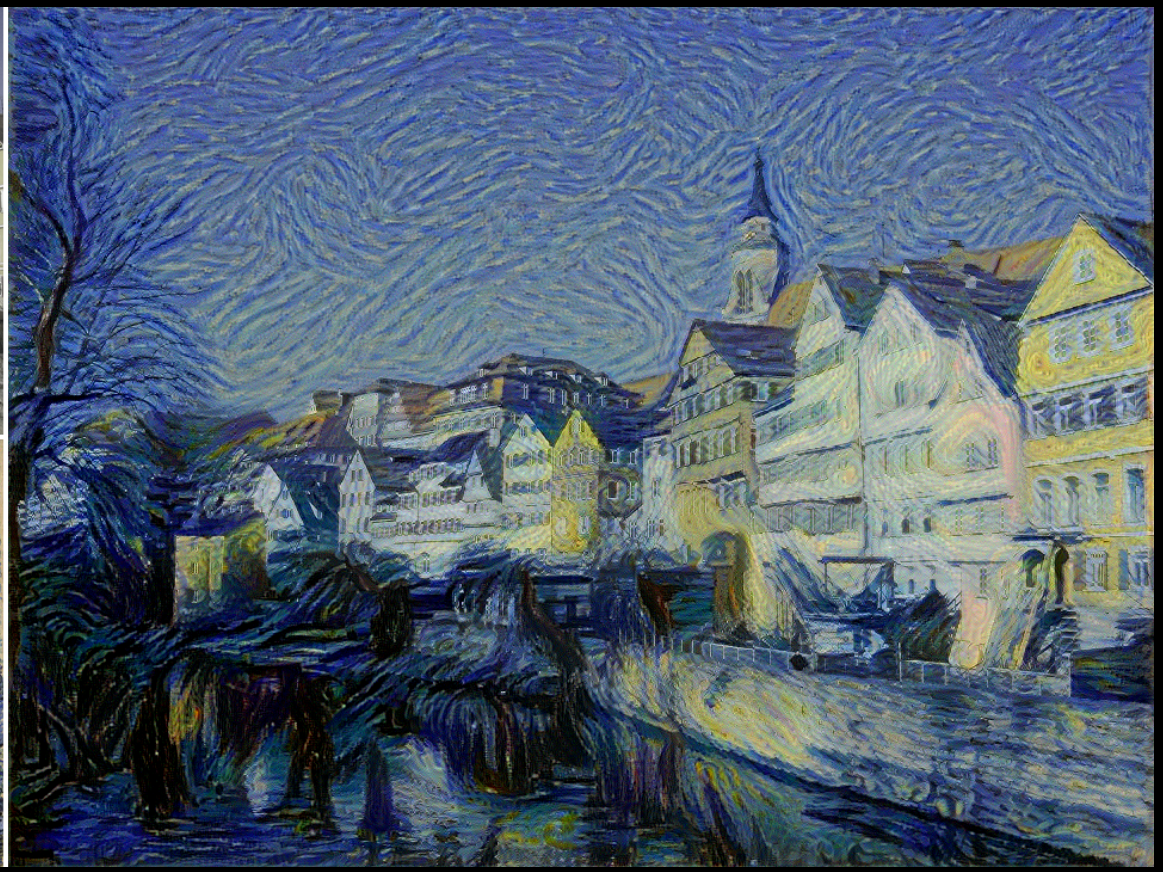
Leon A. Gatys, Alexander S. Ecker, Matthias Bethge

CSCI570 Theory Project

By: Ross Ring-Jarvi

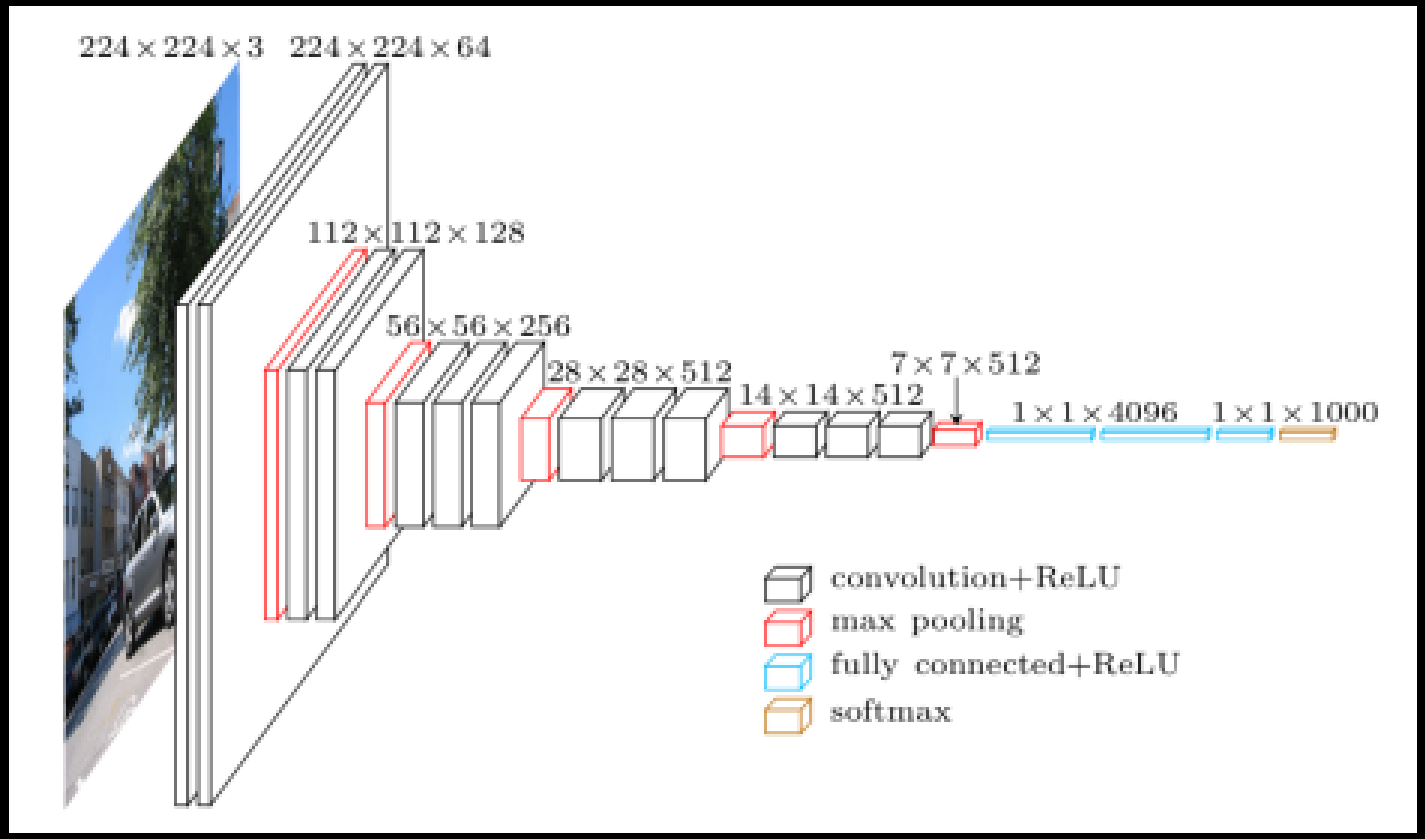
The Problem:

To generate an image that holds the content of an original image but applies the style/texture from a second image.



Solution

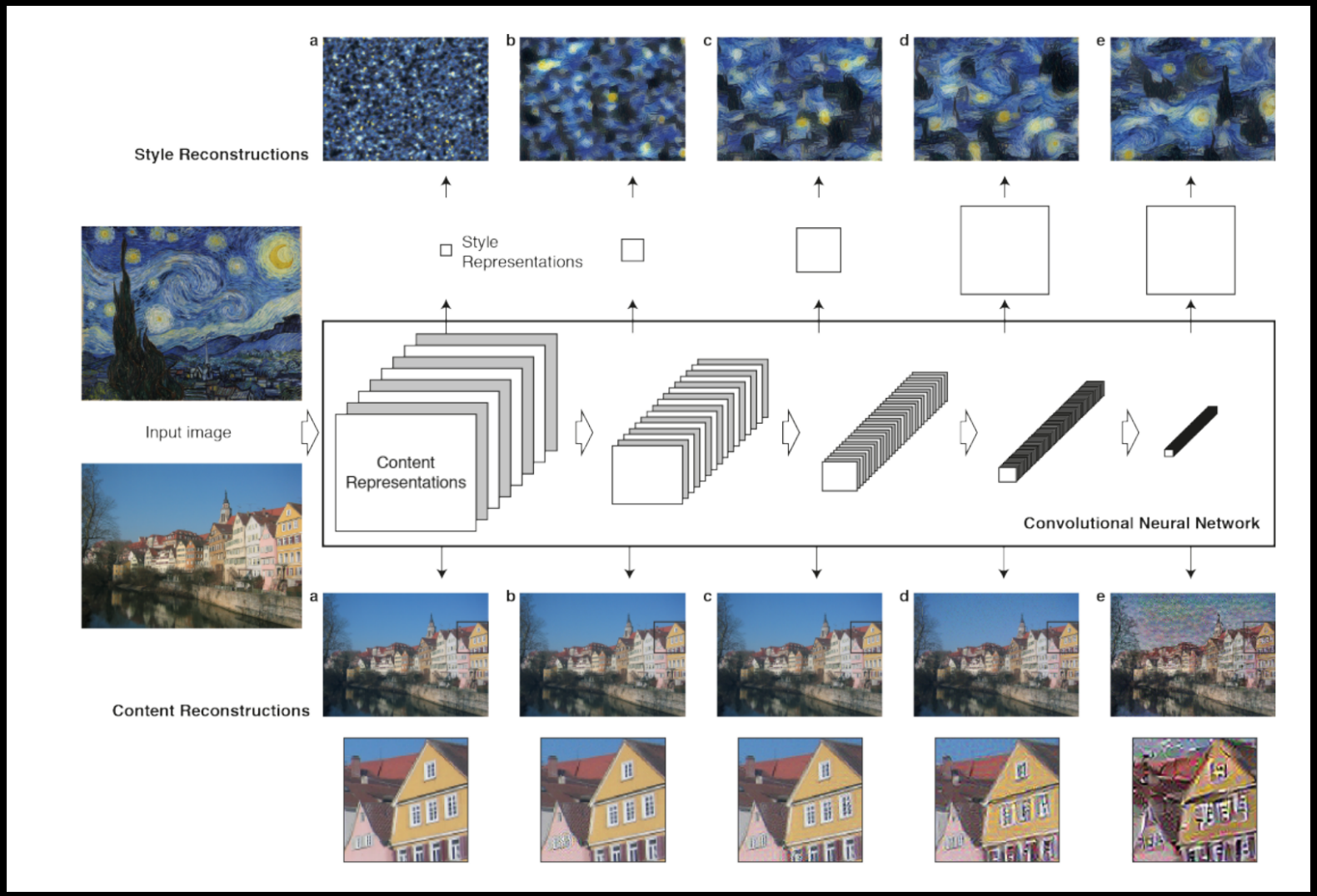
The authors use a convolutional neural network(CNN) to solve the problem. They choose to use VGG-19 which was train in a supervised fashion on object recognition.



VGG-16 Structure

Solution

Forward feed network structure showing how the data is used in the network.



Related Work

- Style Transfer witout Neural Networks
  - Stroke-Based Method [H98]
  - Region-Based Technique [K05][GCS02][SRHC08]
  - Example-Based Rendering [H01][ZZ11]
  - Image Processing and Filtering[HOG06][GRG04][TOG98]
- Visual Texture Modelling
  - Parametric Texture Modeling with Summary Statistics
    - Visual pattern discrimination [J62]
    - A parametric texture model based on join statistics of complex wavelet coefficients[PS95]
    - Texture synthesis using convolutional neural networks [GEB15]
    - Incorporating long-range consistency in cnn-based texture generation [BM17]
  - Non-parametric Texture Modelling with Markov Random Fields
    - Texture synthesis by non-parametric sampling[EL99]
- Image Reconstruction
  - Visualizing deep convolutional neural network using natural pre-images [MV15][MV16]
  - Inverting visual representations with convolutional networks [DB16]
  - Generative adversarial nets [GPMXWOCB14]

Content Loss

We start by passing the content image and our generated image into the network and return layer block5_conv2 from each and calculate the euclidean distance between those layers.
The content loss function describes the distance of content from our output image x and our content image, p.

$$L^{l}_{content}(p, x) = \sum_{i,j} (F^{l}_{ij}(x) - P^{l}_{ij}(p))^2$$

$F^{l}_{ij}(x) \in C_{nn}(x)$ and $P^{l}_{ij}(p) \in C_{nn}(p)$ describe the respective intermediate feature representation of the network with inputs x and p at layer l. $C_{nn}(X)$ is the network fed by any image X.

Style Loss

Similar to content loss, we start by passing the style image and our generated image into the network but now you compare gram matrices from layers block1_conv1, block2_conv1, block3_conv1, block4_conv1, and block5_conv1. The gram matrix is described as G^l_{ij} is the inner product between the vectorized feature map i and j in layer l.

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G^l_{ij} - A^l_{ij})^2$$

G^l_{ij} and A^l_{ij} are the respective style representation in layer l of x(generated image) and a(style image) at layer l. N_l are the number of feature maps and M_l represents the size of each. The total sytle loss:

$$L_{style}(a, x) = \sum_l w_l E_l$$

where each layer's contribution to the loss has a weight w_l assicated with it.

10 Iterations

Iterative process of style transfer.

(a) Style Image

(b) Content Image

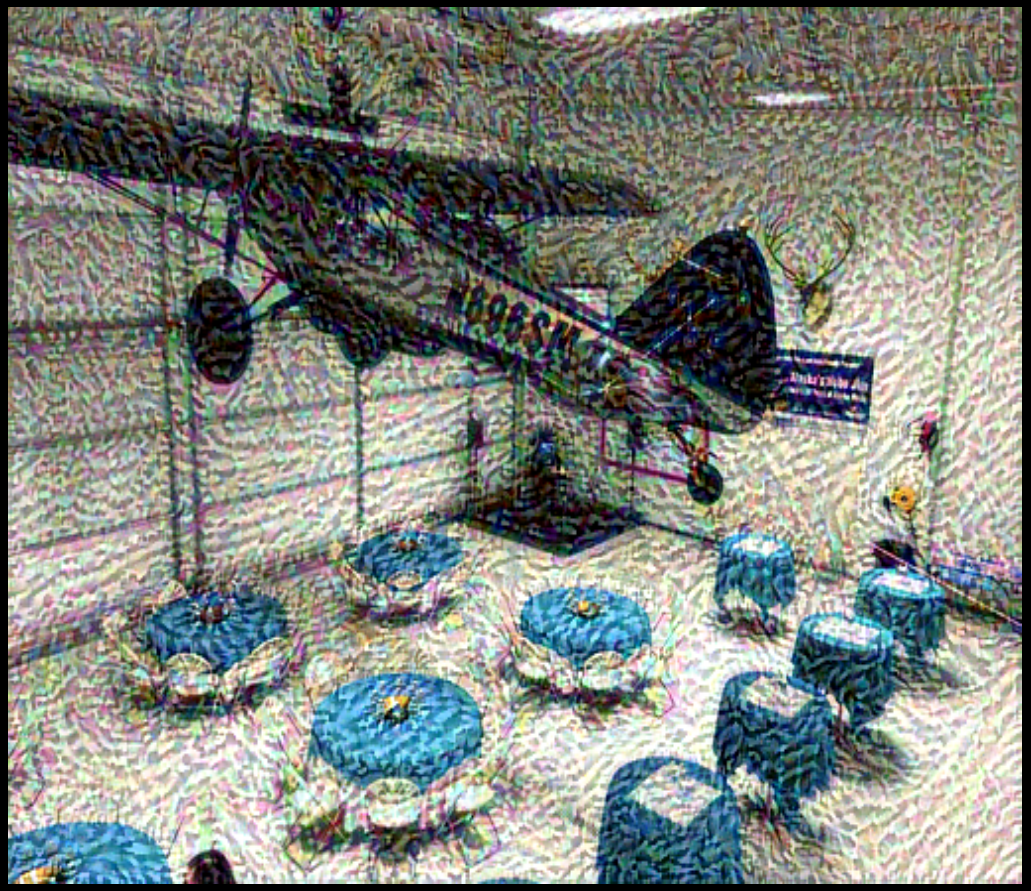
(c) Generated Image

20 Iterations

Iterative process of style transfer.

(d) Style Image

(e) Content Image



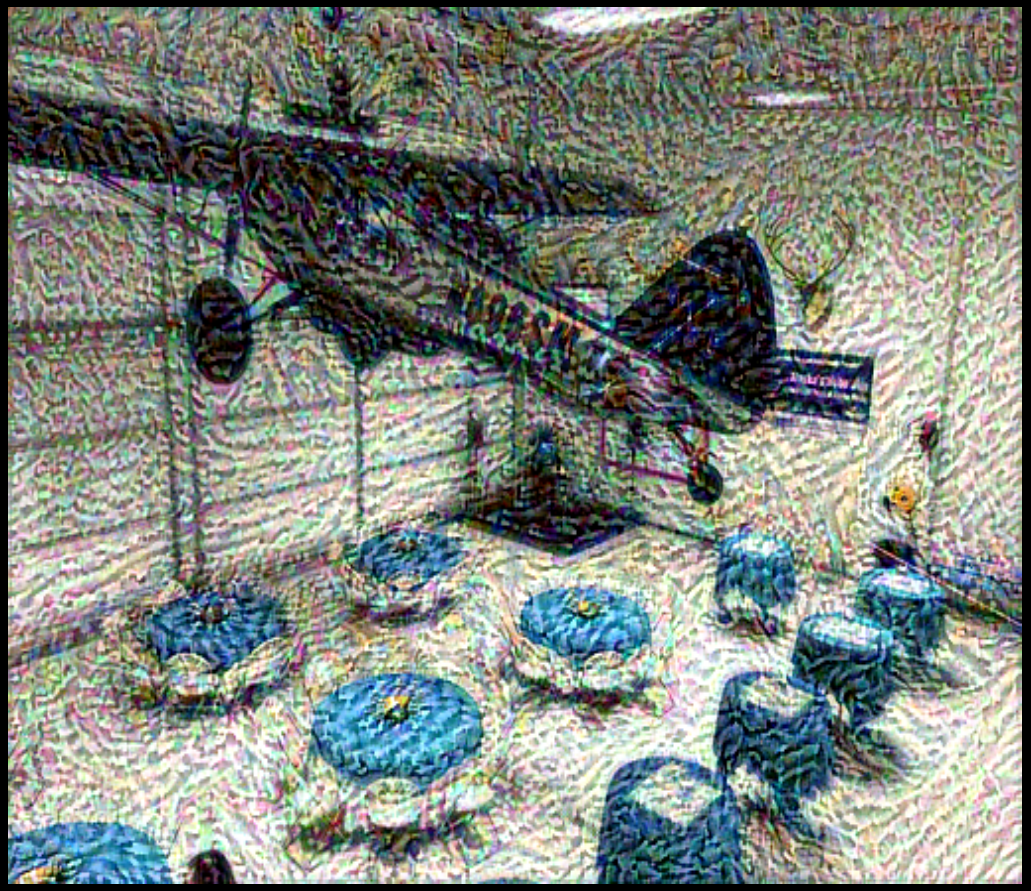
(f) Generated Image

30 Iterations

Iterative process of style transfer.

(g) Style Image

(h) Content Image



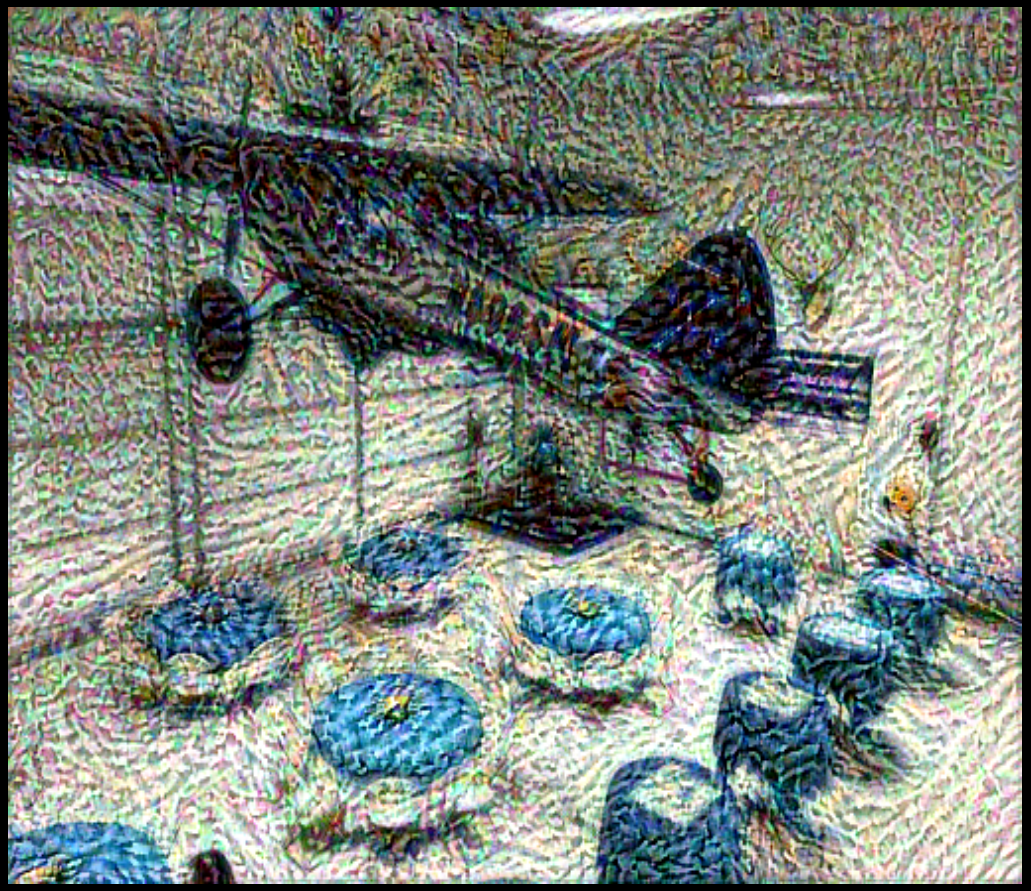
(i) Generated Image

40 Iterations

Iterative process of style transfer.

(j) Style Image

(k) Content Image



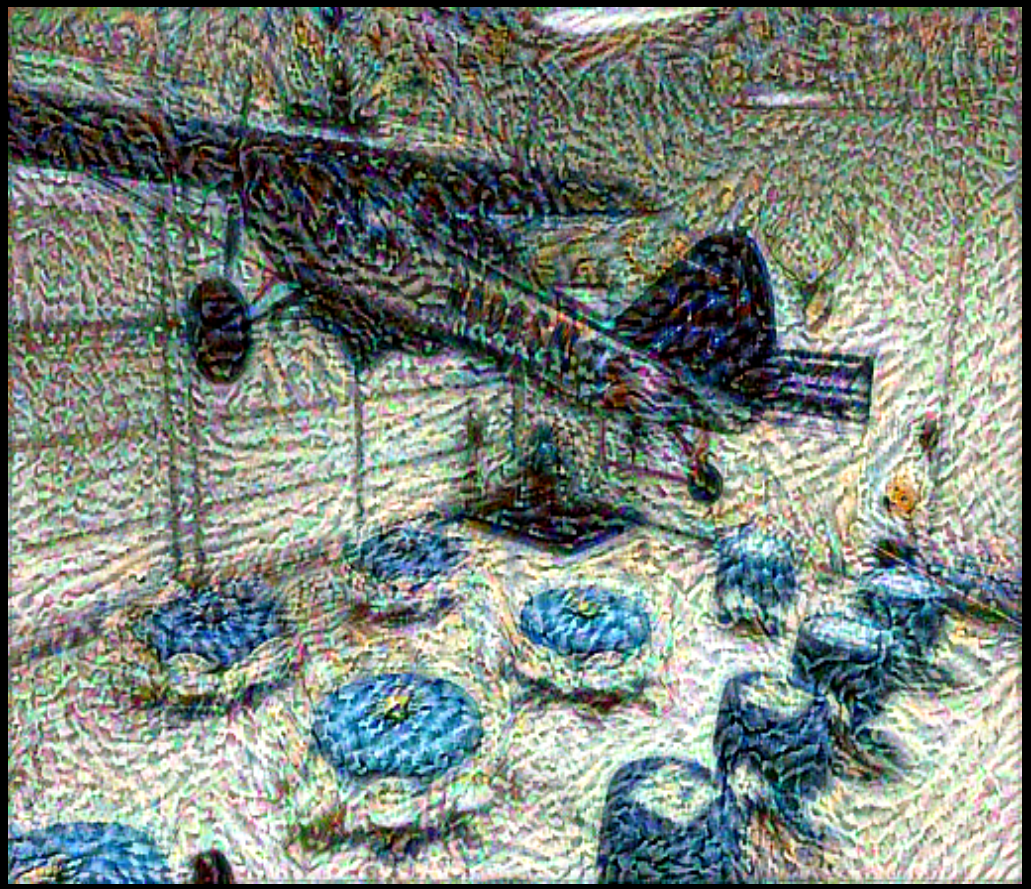
(l) Generated Image

50 Iterations

Iterative process of style transfer.

(m) Style Image

(n) Content Image



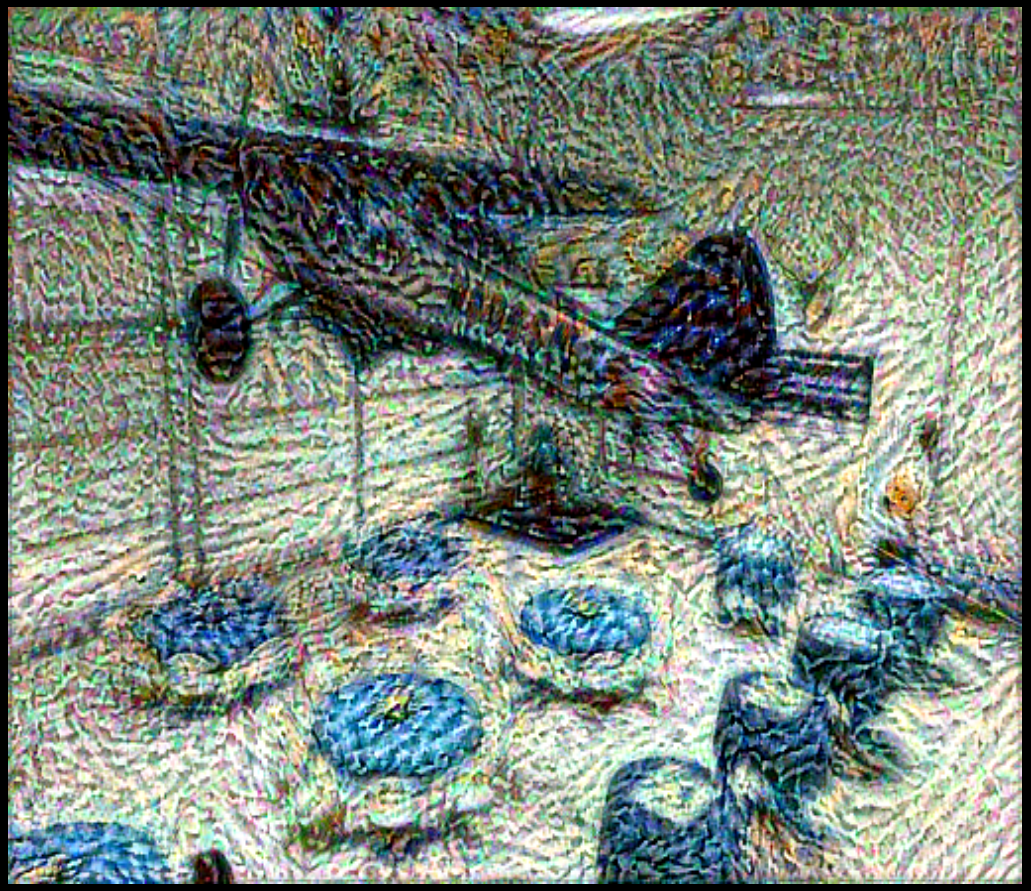
(o) Generated Image

60 Iterations

Iterative process of style transfer.

(p) Style Image

(q) Content Image



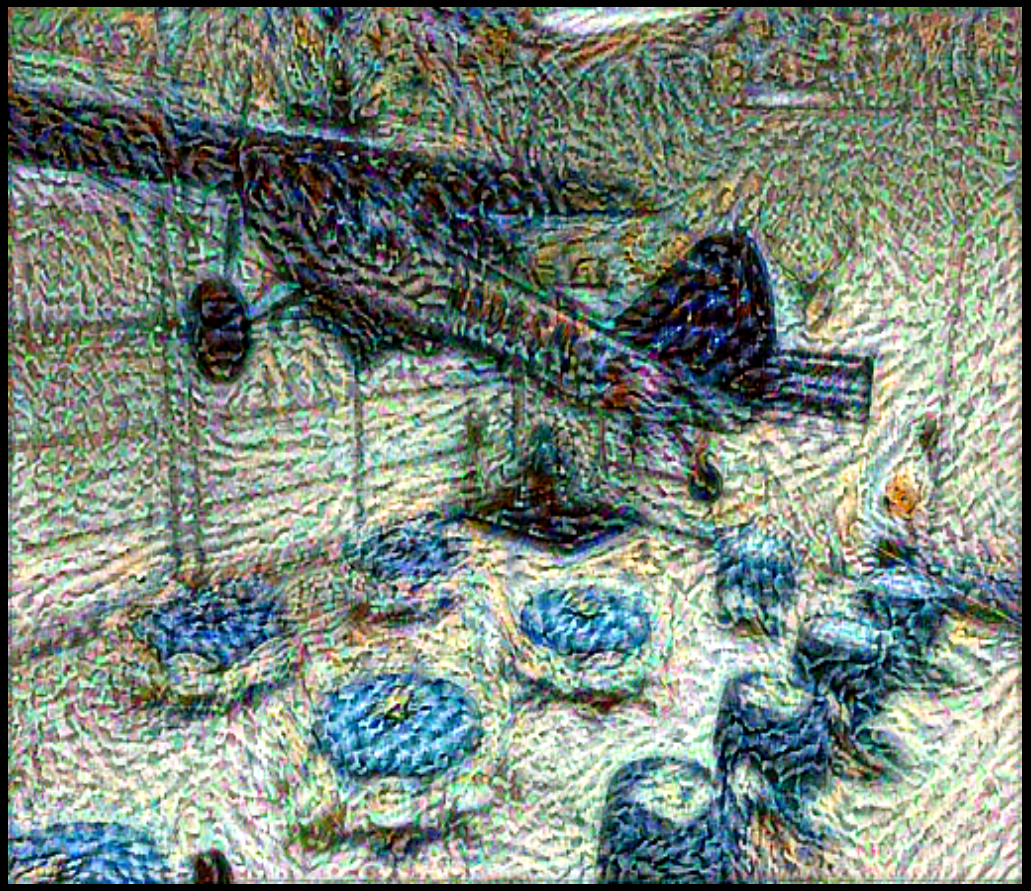
(r) Generated Image

70 Iterations

Iterative process of style transfer.

(s) Style Image

(t) Content Image



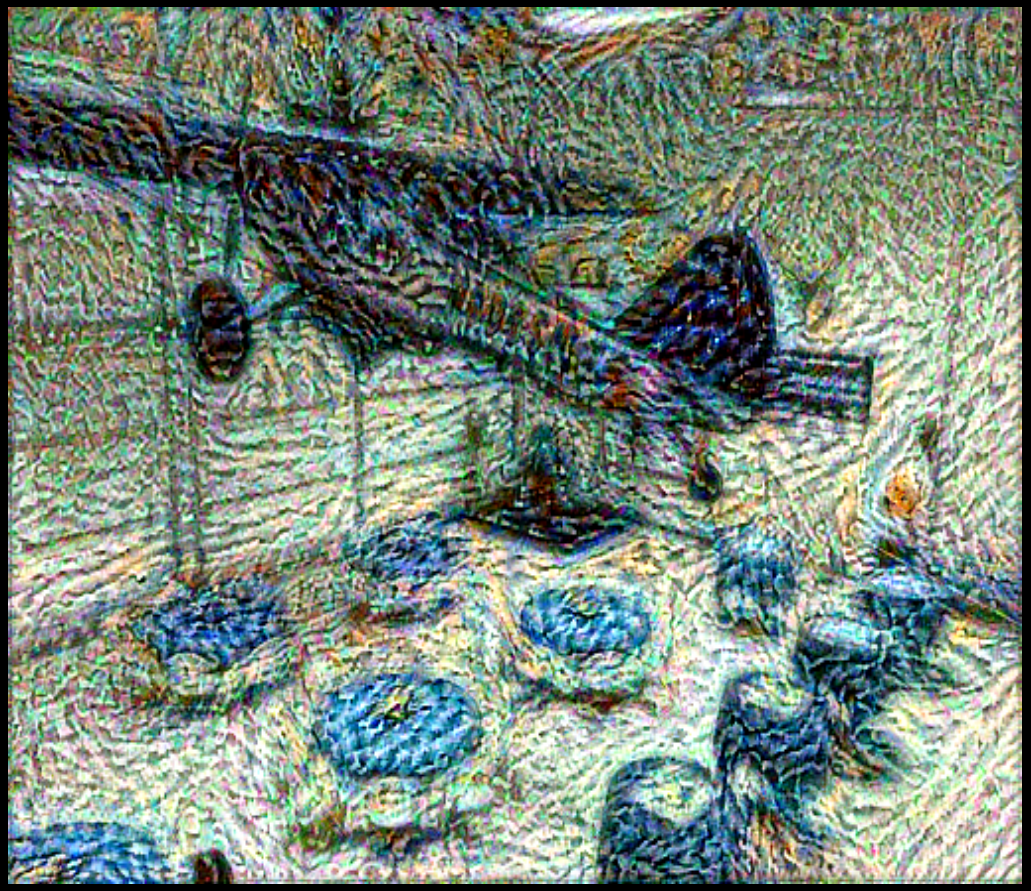
(u) Generated Image

80 Iterations

Iterative process of style transfer.

(v) Style Image

(w) Content Image



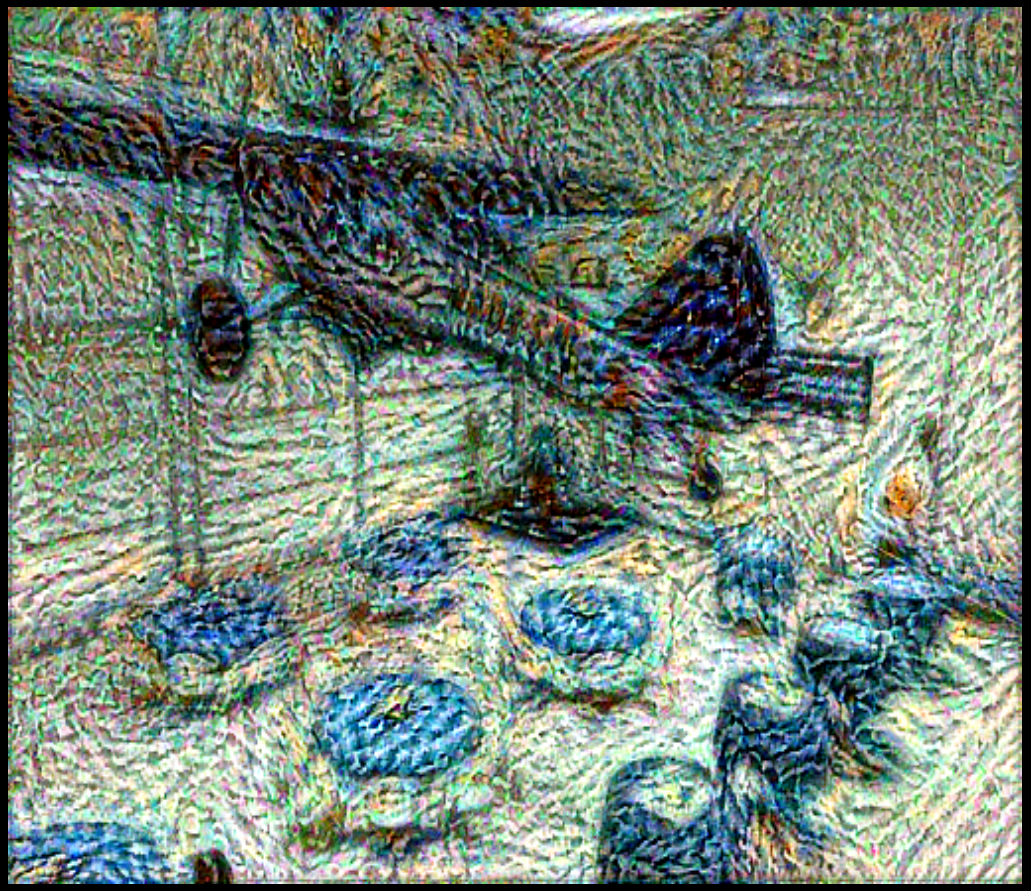
(x) Generated Image

90 Iterations

Iterative process of style transfer.

(y) Style Image

(z) Content Image



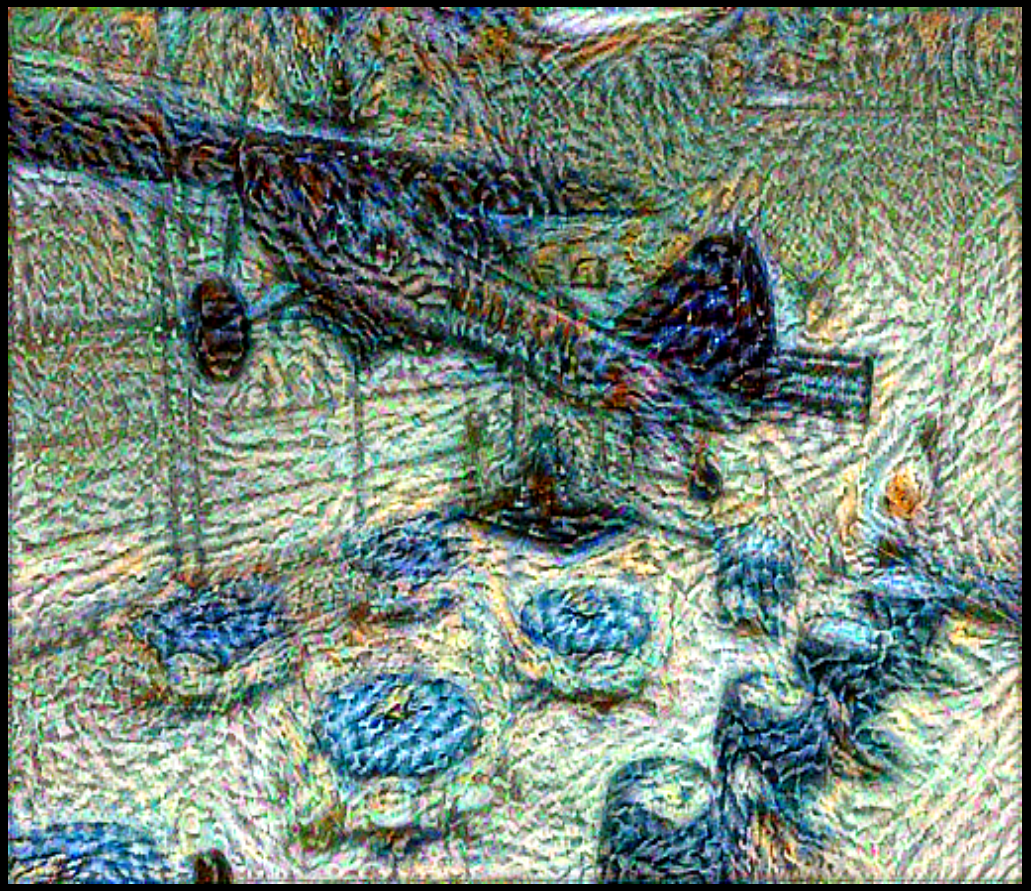
() Generated Image

100 Iterations

Iterative process of style transfer.

() Style Image

() Content Image



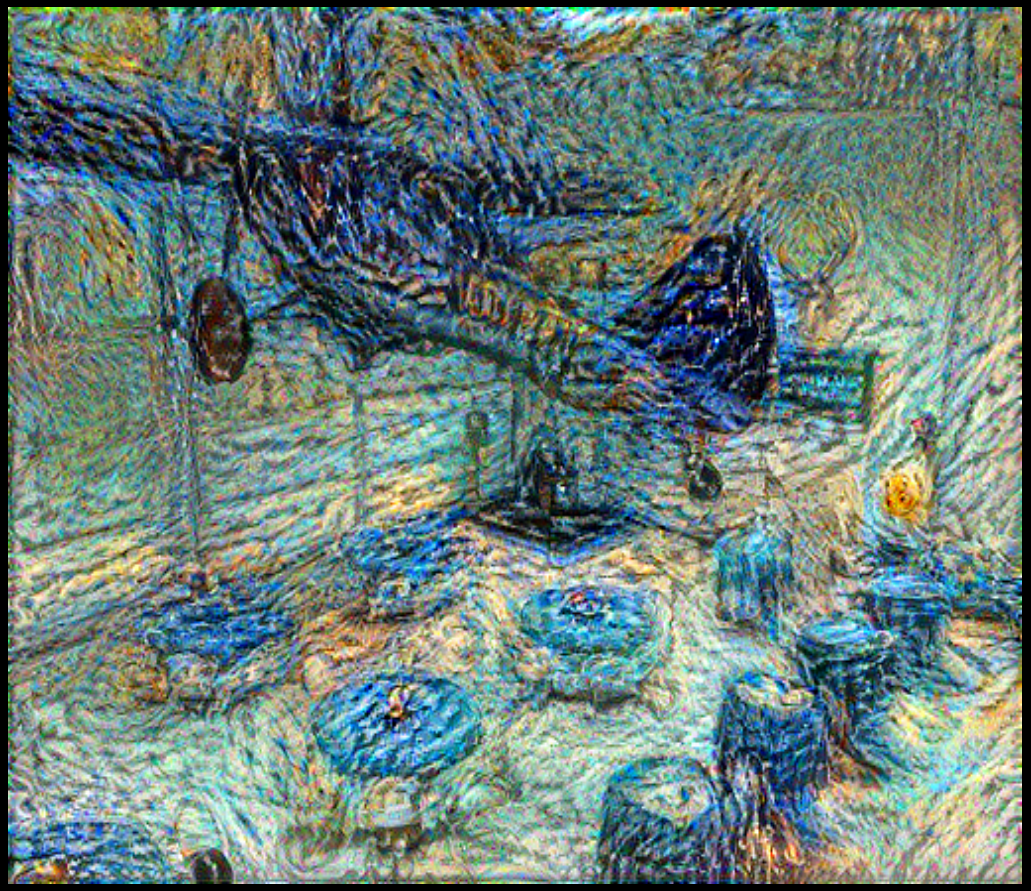
() Generated Image

500 Iterations

Iterative process of style transfer.

() Style Image

() Content Image



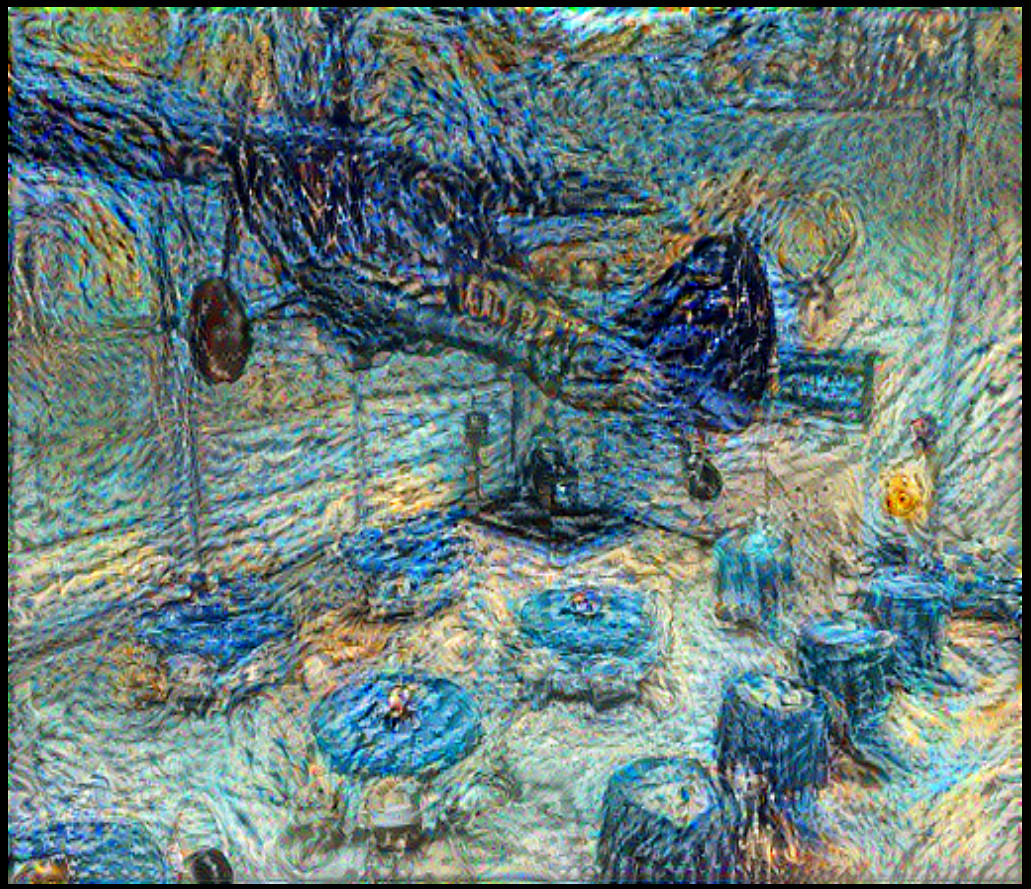
() Generated Image

1000 Iterations

Iterative process of style transfer.

() Style Image

() Content Image



() Generated Image

Style and Content Loss Plots over 1000 iterations

Notice the content loss increases then decreases but the scale is significantly lower compared to the change the style loss has.

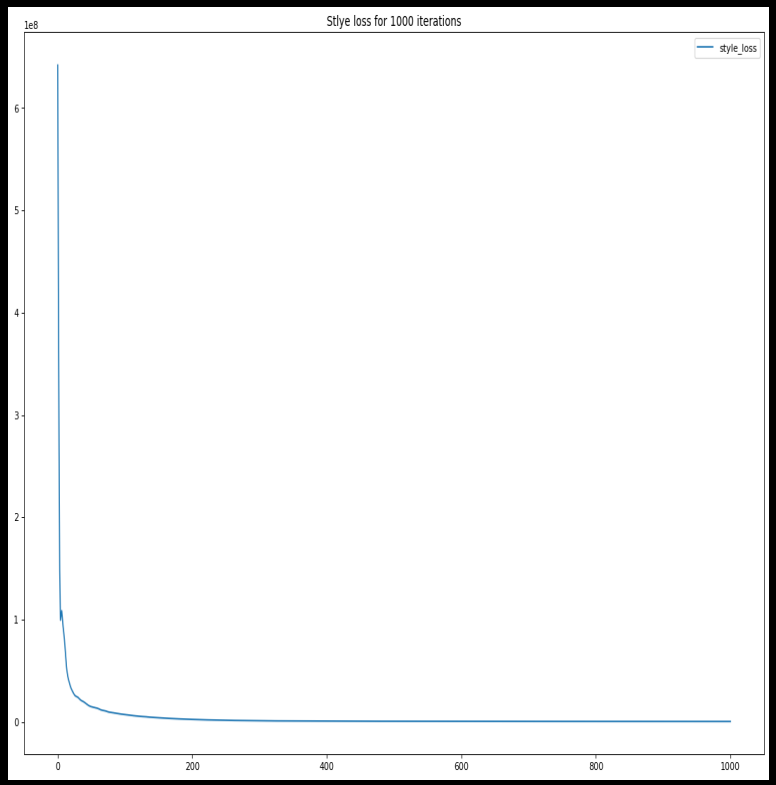


Figure: Style Loss

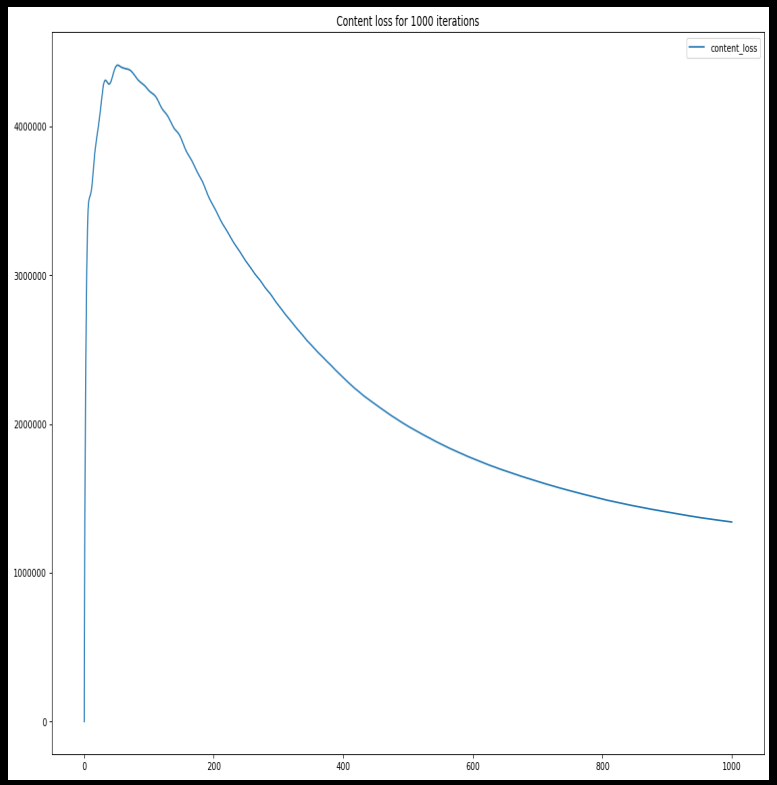


Figure: Content Loss

Way to ”judge” Style Transfer?

Total, Style, and Content loss plotted together. Notice that the total loss mirrors the style loss. After about 200 iterations the loss has plateaued, maybe this could be a determining factor is style transfer?

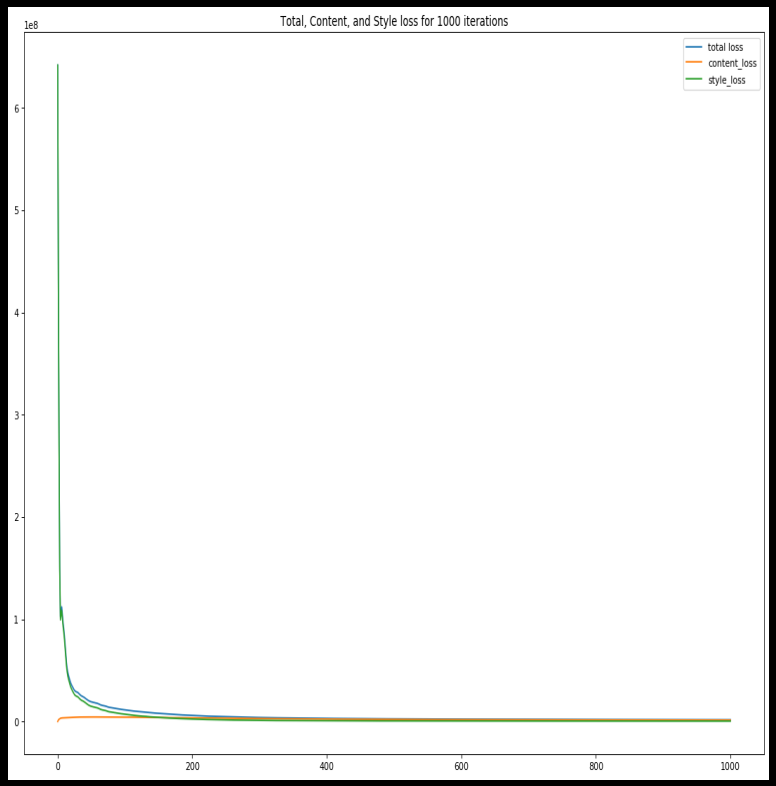


Figure: Total, Style, and Content Loss

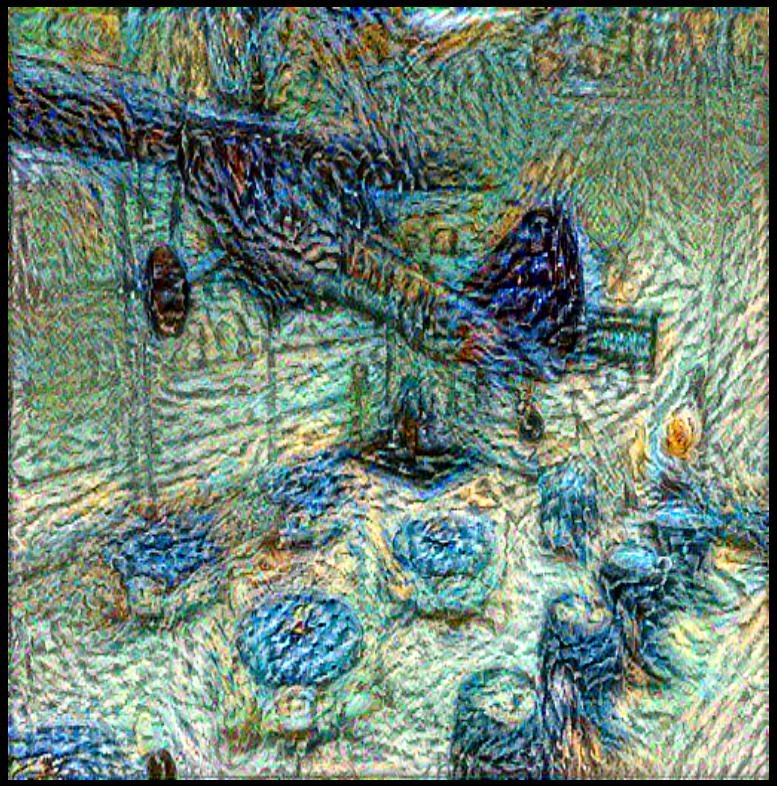
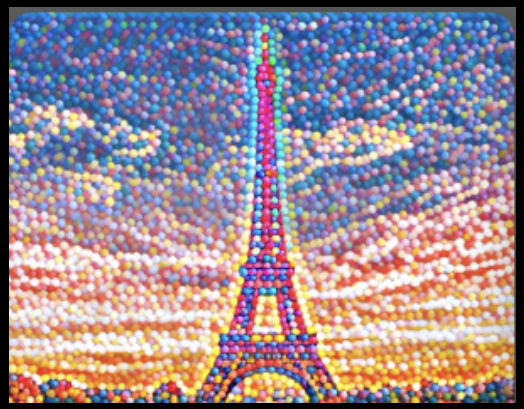
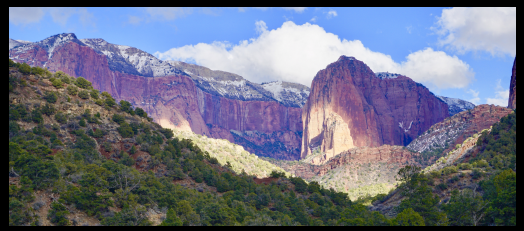


Figure: 200 iterations

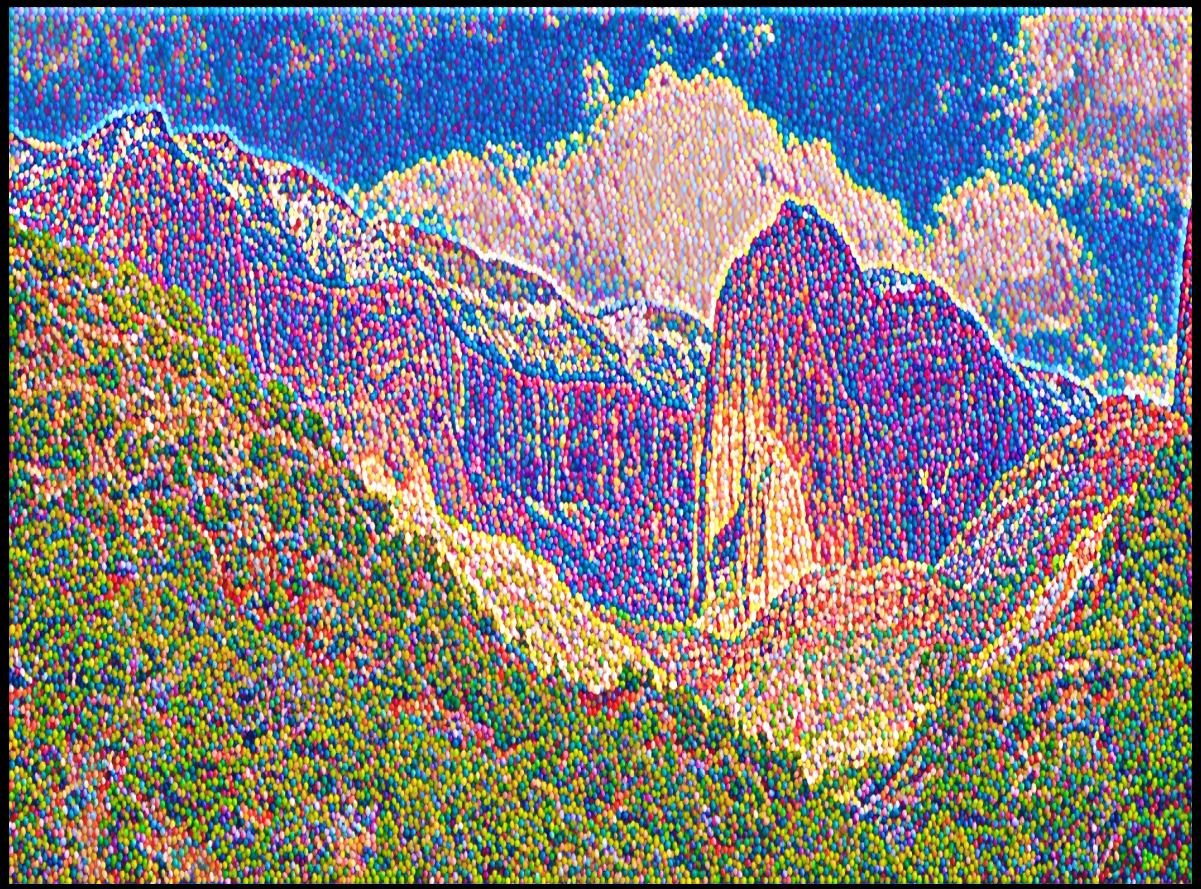
National Parks Image Data set



(a) Style Image



(b) Content Image



(c) Generated Image

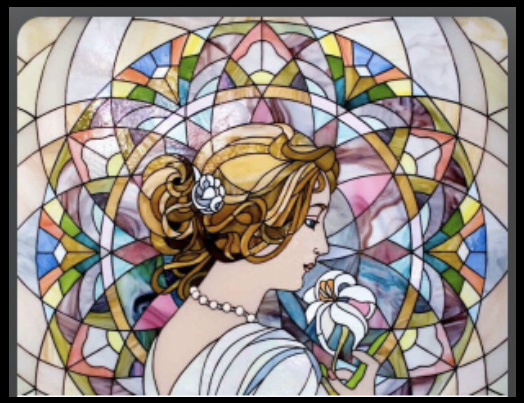
National Parks Image Data set

(d) Style Image

(e) Content Image

(f) Generated Image

National Parks Image Data set



(g) Style Image

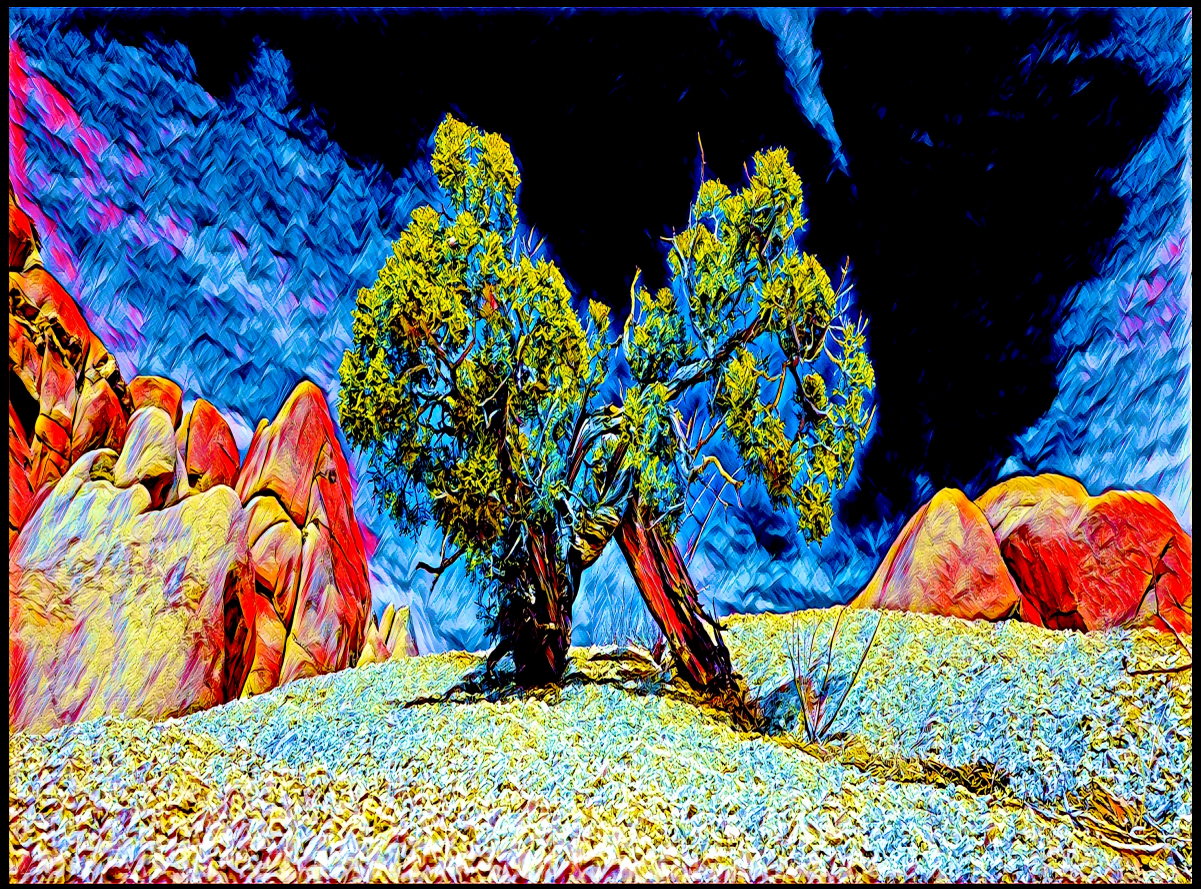
(h) Content Image

(i) Generated Image

National Parks Image Data set

(j) Style Image

(k) Content Image



(l) Generated Image

National Parks Image Data set

(m) Style Image

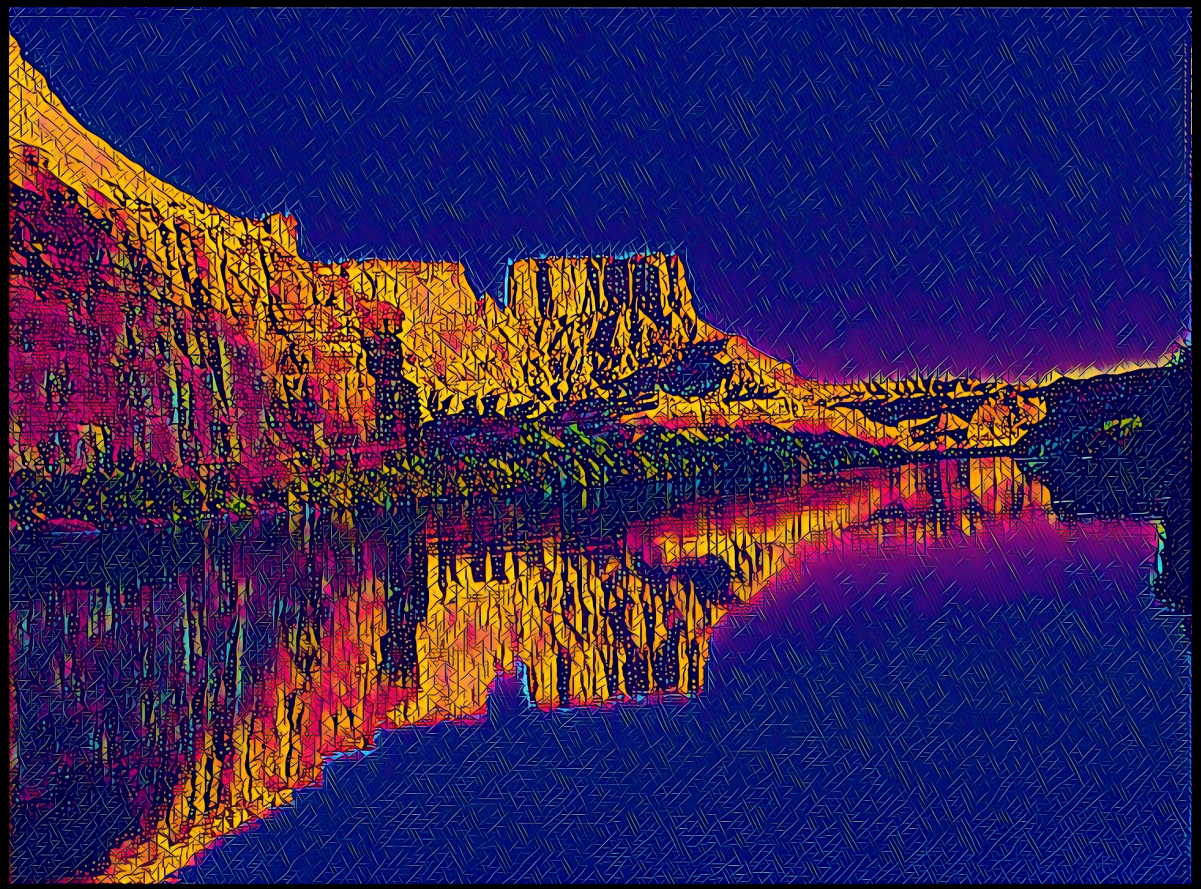
(n) Content Image

(o) Generated Image

National Parks Image Data set

(p) Style Image

(q) Content Image



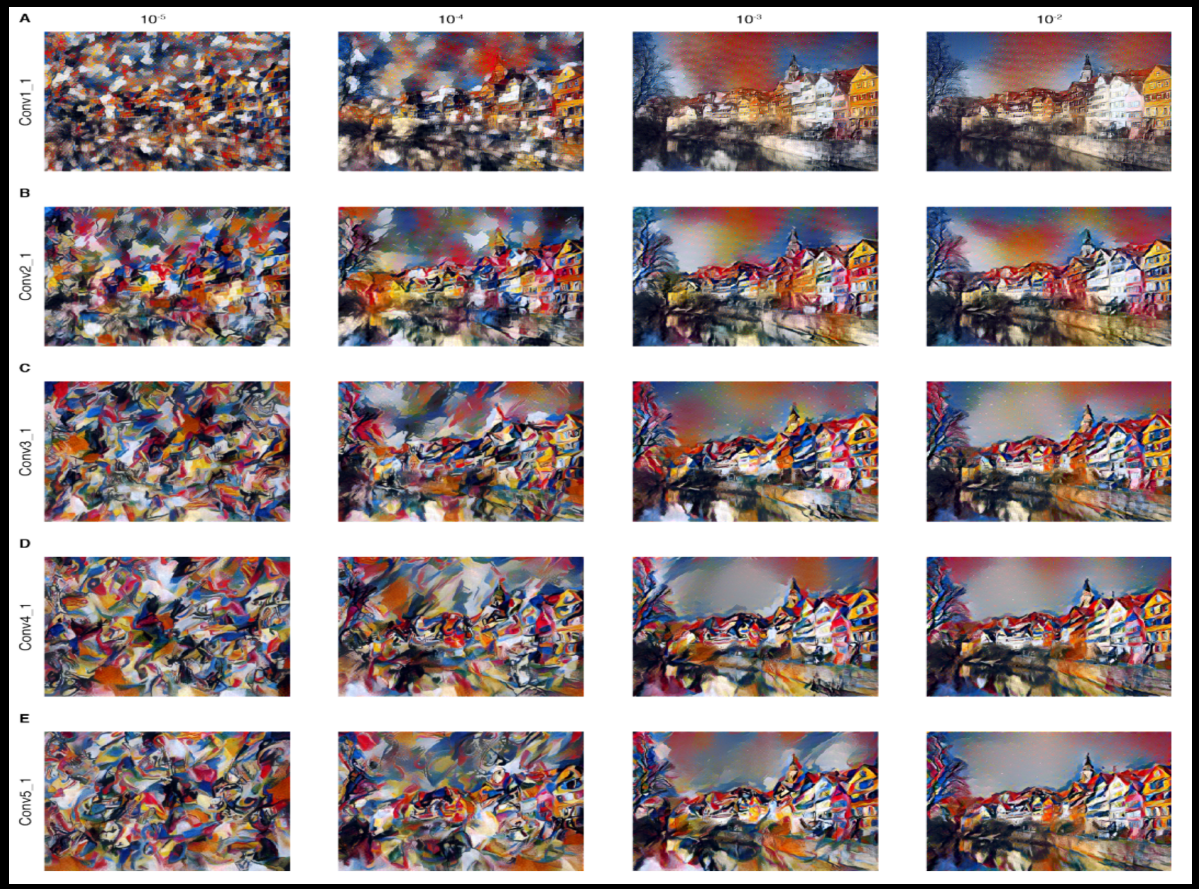
(r) Generated Image

Content layer Loss Ratios

Rows show different layers from VGG-19 for content loss comparison. Columns have different ratio for weights associated content loss and style loss.

(s) Style Image

(t) Content Image



(u) Generated Images

References

Gatys, L. A., Ecker, A. S. Bethge (2015)

M. Texture synthesis and the controlled generation of natural stimuli using convolutional neural networks *arXiv:1505.07376* [cs, q-bio] (2015)

Jing, Yongcheng Yang, Yezhou Feng, Zunlei Ye, Jingwen Yu, Yizhou Song, Mingli. (2017)

Neural Style Transfer: A Review.

Gatys, L.A., Ecker, A.S., Bethge, M.(2015)

A Neural Algorithm of Artistic Style.
ArXiv, abs/1508.06576. abs/1508.06576. (2015)

Questions?

Sophie!