A Neural Algorithm of Artistic Style

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

Present by: Tianye Wang

A Neural Algorithm of Artistic Style

- Motivation
  What is neural style transfer and why.

- Background
  VGG

- Methods
  Cost function: Content loss and Style loss.

Arts

Vincent van Gogh The Starry Night
Saint Rémy, June 1889

Image source: https://www.moma.org/collection/works/79802

What is Neural Style Transfer?

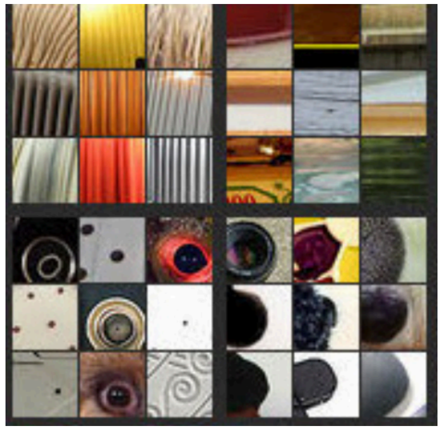
Style Image

Content Image

Generated Image

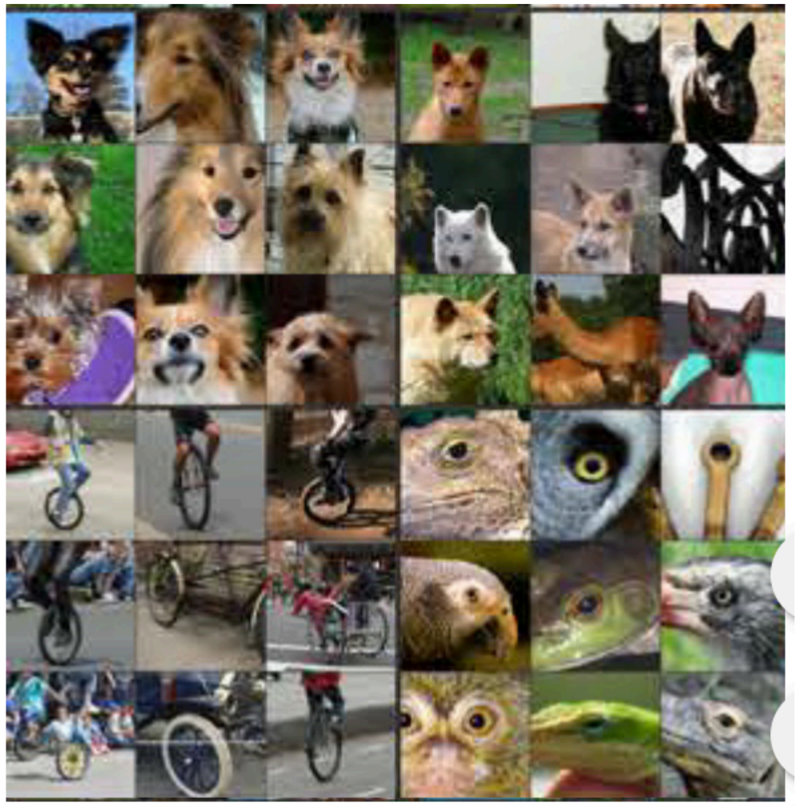
Background: Visualization of CNN

Layer_1



Layer_2

Layer_3



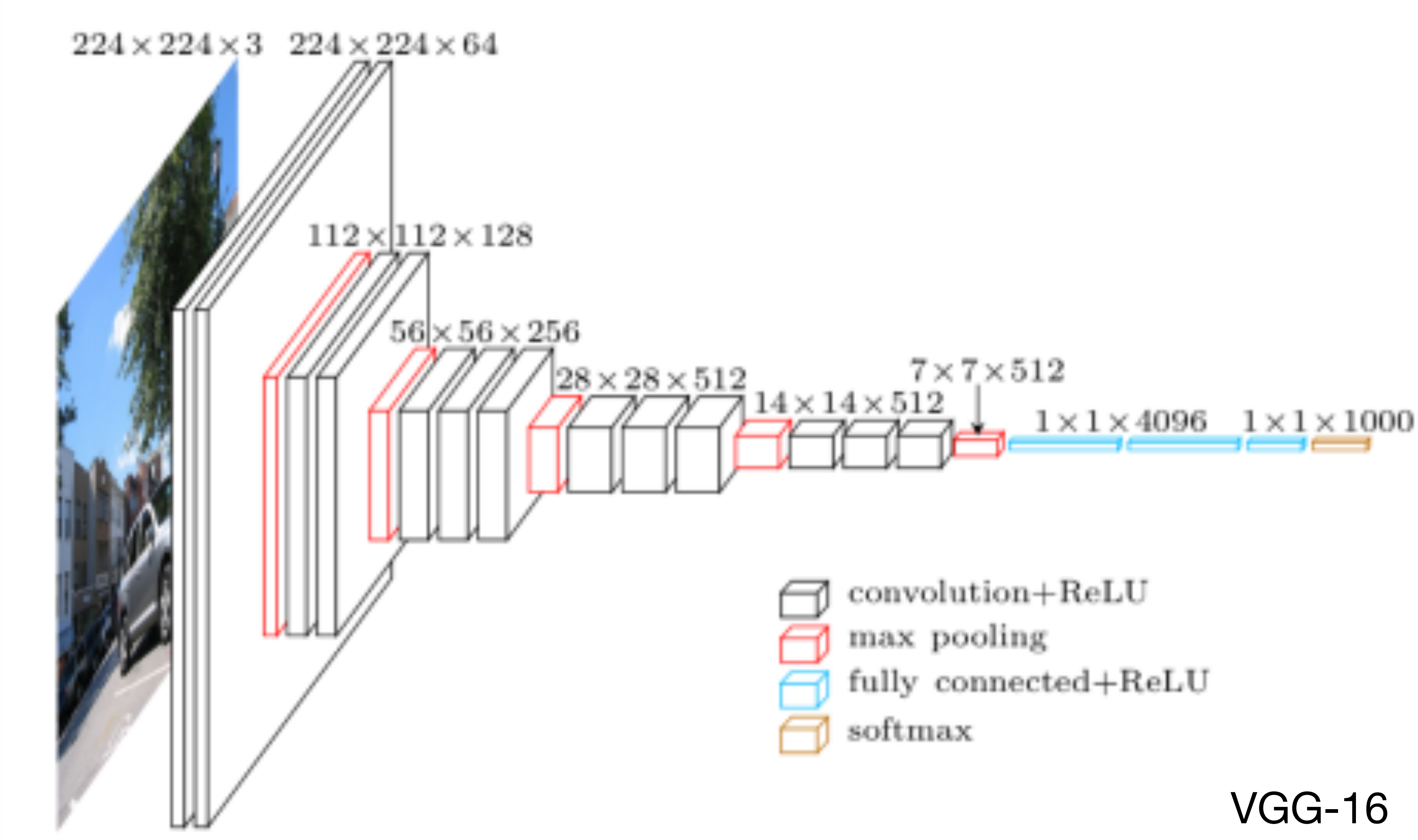
Layer_4

Image from Visualizing and Understanding Convolutional Networks , Matthew D. Zeiler and Rob Fergus

Background: VGG Network

Designed by: Visual Geometry Group

Application: object detection for 1000 object



VGG-16

Image Source: https://www.quora.com/What-is-the-VGG-neural-network

Total Loss Function:

Style Image (S)

Content Image (C)

Generated Image (G)

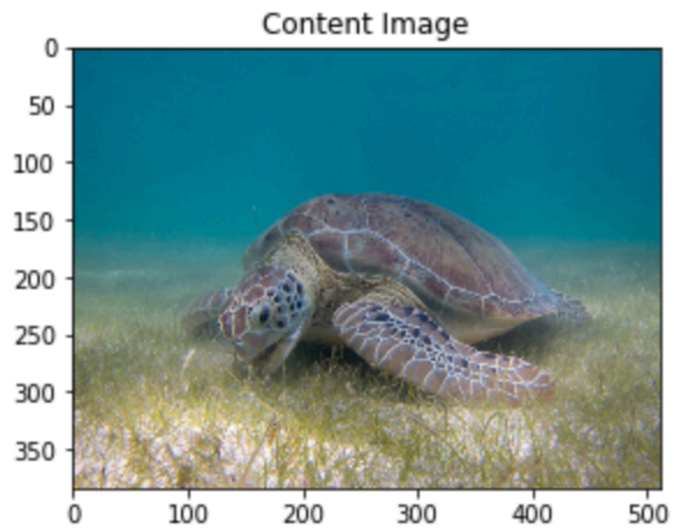
Loss Function: $L_{total}(S, C, G) = \; L_{content}(C, G) + \; L_{style}(S, G)$

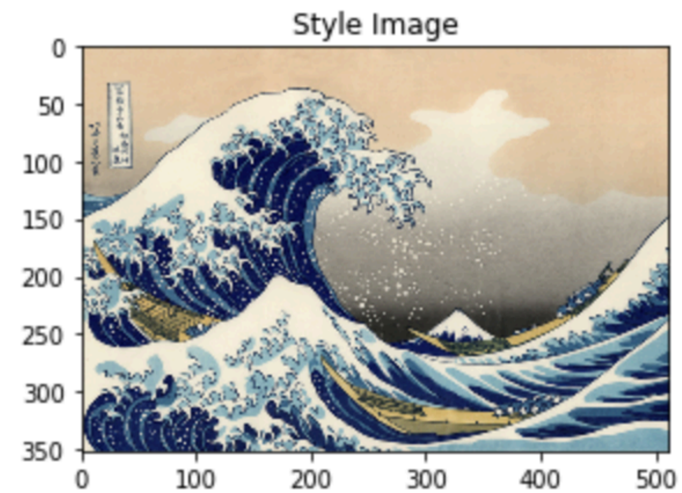
Neural Style Transfer Algorithm:

Step 1: Randomly initialize G, for example: G = 128*128*3

Step 2: Minimize $L_{total}(S, C, G)$ through gradient descent.

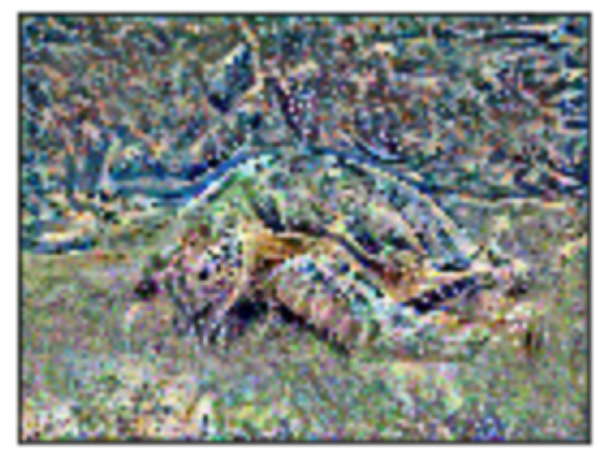
Update rule : $G = G - \frac{\partial}{\partial G} L_{total}(S, C, G)$





Generated Image

Iteration 100



Iteration 200

Iteration 400

Content Loss Function

Content Representation:

- Higher layers capture the high-level content of objects and their arrangement in the image.
- But they do not constrain the exact pixel values of the reconstruction image.

Layer_1

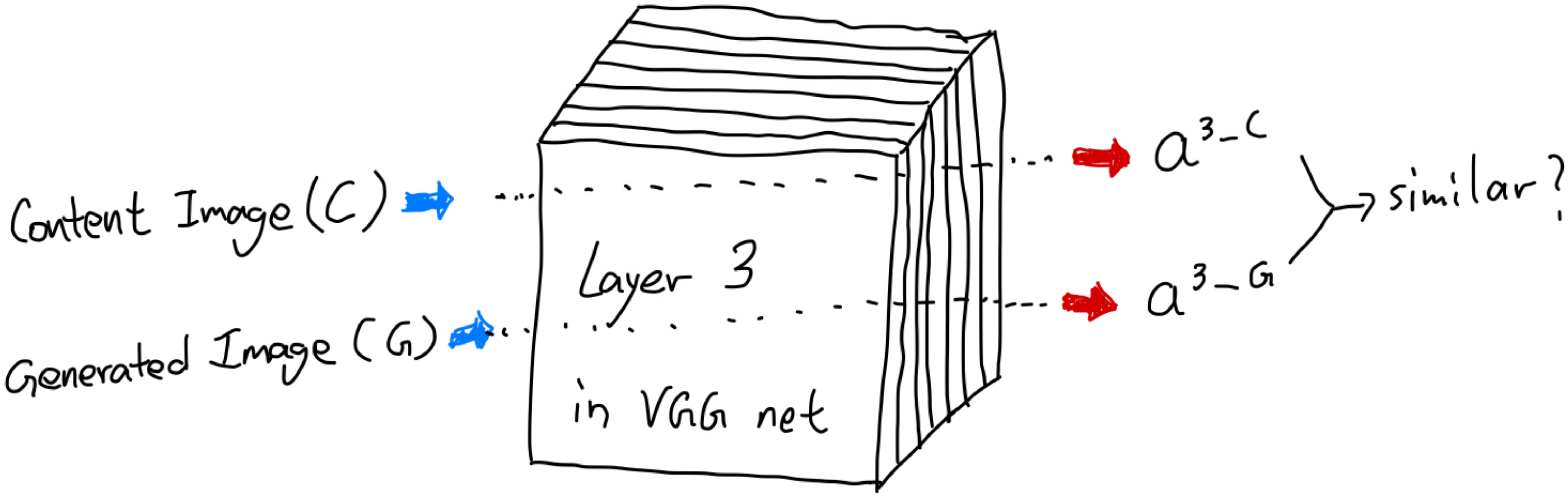
Layer_2

Layer_3

Layer_4

Layer_5

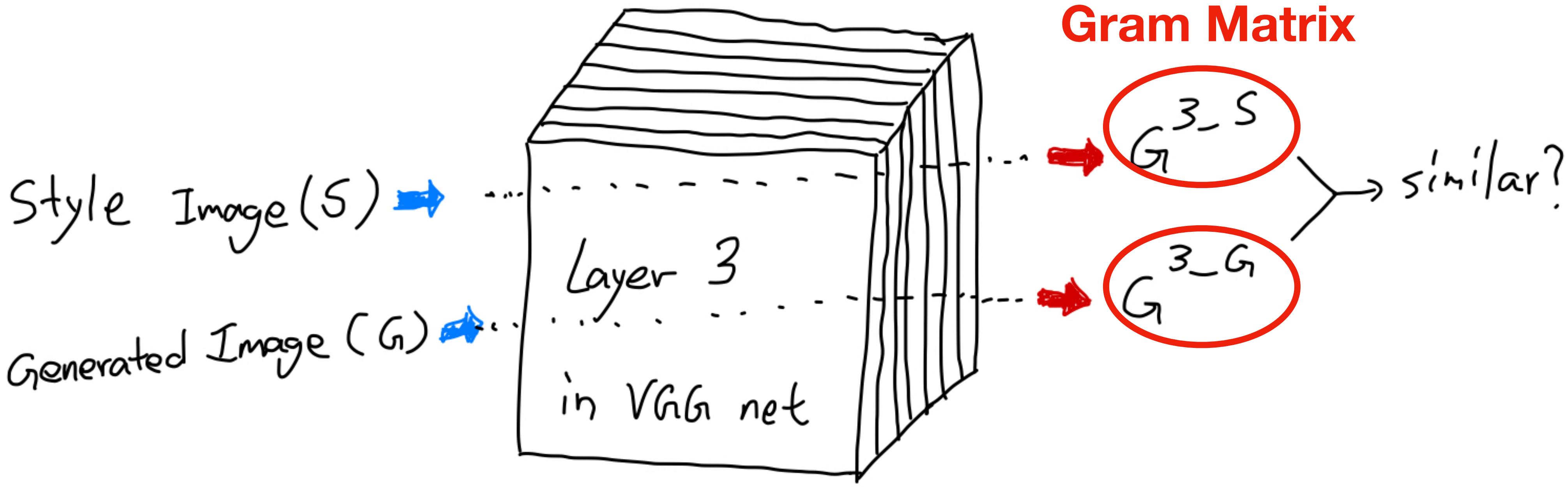
Content Loss Function



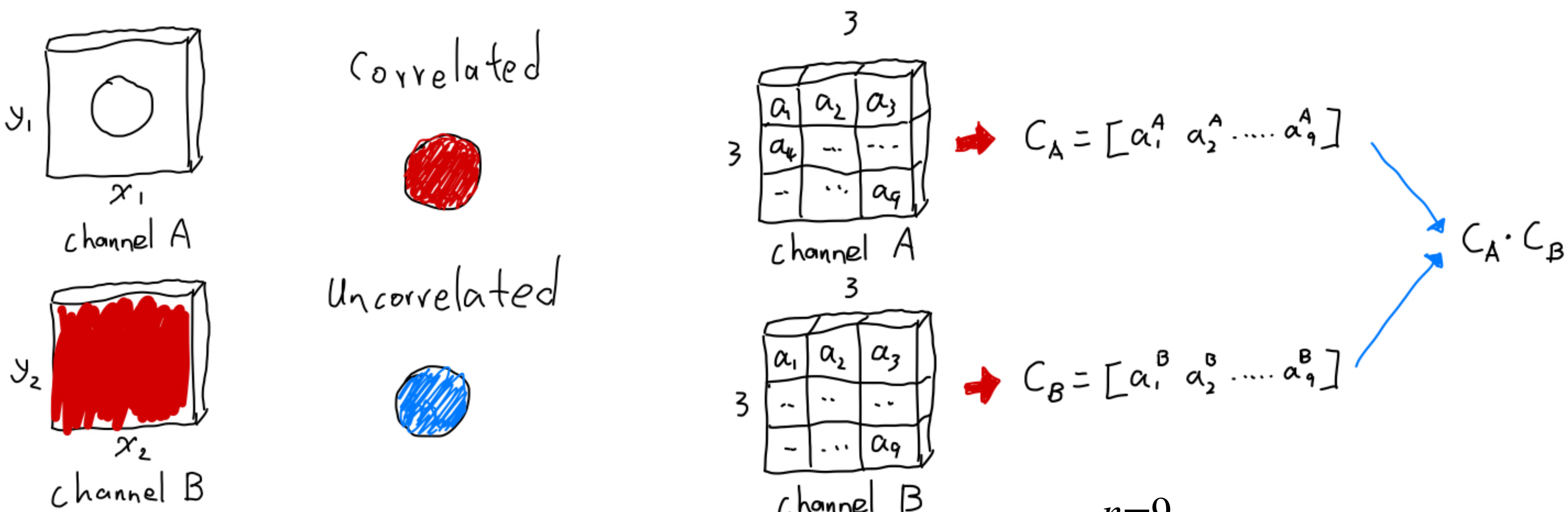
$$L_{content}(C, G) = \frac{1}{2} \sum_{ij} (a_{ij}^{l\_C} - a_{ij}^{l\_G})^2$$

- L2 Norm of element wise subtraction between these two activation matrices

Style Representation



We define the style as the correlation between activations across channels.



$$C_A \cdot C_B = a_1^A a_1^B + a_2^A a_2^B + \ldots a_9^A a_9^B = \sum_{i=1}^{n=9} a_i^A a_i^B$$

Gram Matrix Example:

Layer 3:
size 3x3
64 channels

$$\begin{array}{c} \\ C_1 \\ C_2 \\ \vdots \\ C_{64} \end{array} \overset{3\times 3}{\begin{bmatrix} a_1^1 & a_2^1 & \cdots & a_9^1 \\ a_1^2 & a_2^2 & \cdots & a_9^2 \\ & \vdots & & \\ a_1^{64} & a_2^{64} & \cdots & a_9^{64} \end{bmatrix}} \xrightarrow{\text{Transpose}} \overset{64}{\begin{bmatrix} a_1^1 & a_1^2 & \cdots & a_1^{64} \\ a_2^1 & a_2^2 & \cdots & a_2^{64} \\ \vdots & \vdots & & \vdots \\ a_9^1 & a_9^2 & & a_9^{64} \end{bmatrix}} \; 3\times 3$$

$$M \qquad\qquad\qquad M^T$$

$$C_1 \cdot C_2 = a_1^1 a_1^2 + a_2^1 a_2^2 + \ldots a_9^1 a_9^2$$

$$G = M \cdot M^T \begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1\,64} \\ C_{21} & C_{22} & \cdots & C_{2\,64} \\ & \vdots & & \\ C_{64\,1} & C_{64\,2} & \cdots & C_{64\,64} \end{bmatrix}$$

Feature correlations(Gram Matrix):

$$G_{ij}^{l\_S} = \sum_k a_{ik} a_{jk'}$$

$$G_{ij}^{l\_G} = \sum_k a_{ik} a_{jk'}$$

Note: Gram matrix is the matrix of all possible inner products of the activations.

Style Loss Function

The contribution of layer l:

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{ij} (G_{ij}{}^{l\_S} - G_{ij}{}^{l\_G})^2$$

Where N_l : number of feature maps, M_l is the height times the width of the feature map.

Total style loss:

$$L_{style}(S, G) = \sum_l w_l E_l$$

Total loss:

$$L_{total}(S, C, G) = \alpha L_{content}(C, G) + \beta L_{style}(S, G)$$

Generated Image:

Re-cap:

$$L_{total}(S, C, G) = \alpha L_{content}(C, G) + \beta L_{style}(S, G)$$

Key findings of the paper:

- The representations of content and style in the Convolutional Neural Network are separable.
- We can manipulate both representations independently to produce new, perceptually meaningful images.

For my project:

- Train using style loss from one layer at a time.
- Train using content loss from one layer at a time.
- Train using different sizes of style image

References:

- https://gaussian37.github.io/vision-concept-gram-matrix/
- https://www.moma.org/collection/works/79802
- https://arxiv.org/pdf/1508.06576.pdf
- https://towardsdatascience.com/neural-networks-intuitions-2-dot-product-gram-matrix-and-neural-style-transfer-5d39653e7916