MRP - Week 3/4

This week we will be looking into loss functions and how the style transfer problem can be viewed as an optimization problem

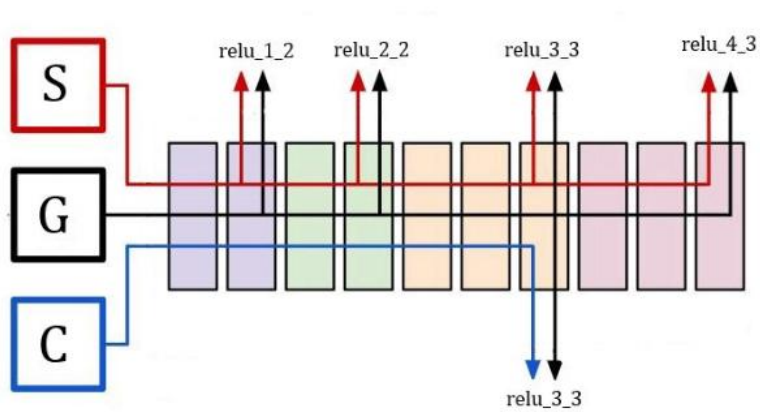
---

- We want to calculate loss between the generated image with respect to content and style image

  Loss function

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

- alpha and beta hyper parameters which are used to provide weights to each type of loss
- can be thought of simply as knobs to control how much of the content/style we want to inherit in the generated image



- during each iteration all the three images i.e. content image, style image and generated image are passed through the vgg16 model
- the value of the hidden unit's activation which encode feature representation of the given image at certain layers are taken as input to these loss functions
- no rules on selection of layers

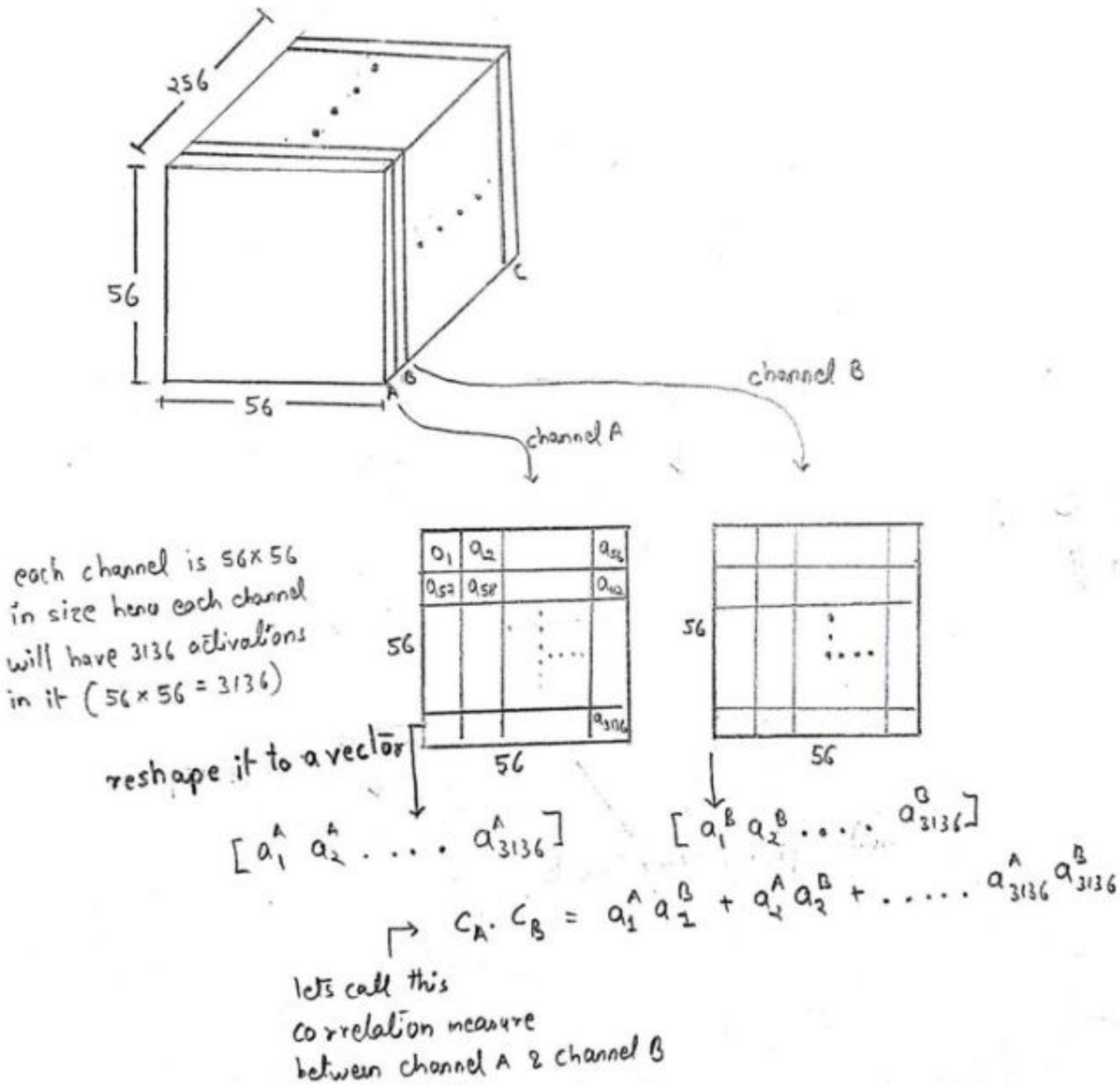
Content Loss

- content image ***C***, generated image ***G*** and the layer **L** whose activation's we are going to use to compute loss
- denote each activation layer of content image as ***a[L](C)*** and activation layer of generated image as ***a[L](G)***

$$L_{content}(C, G, L) = \frac{1}{2} \sum_{ij} (a[L](C)_{ij} - a[L](G)_{ij})^2$$

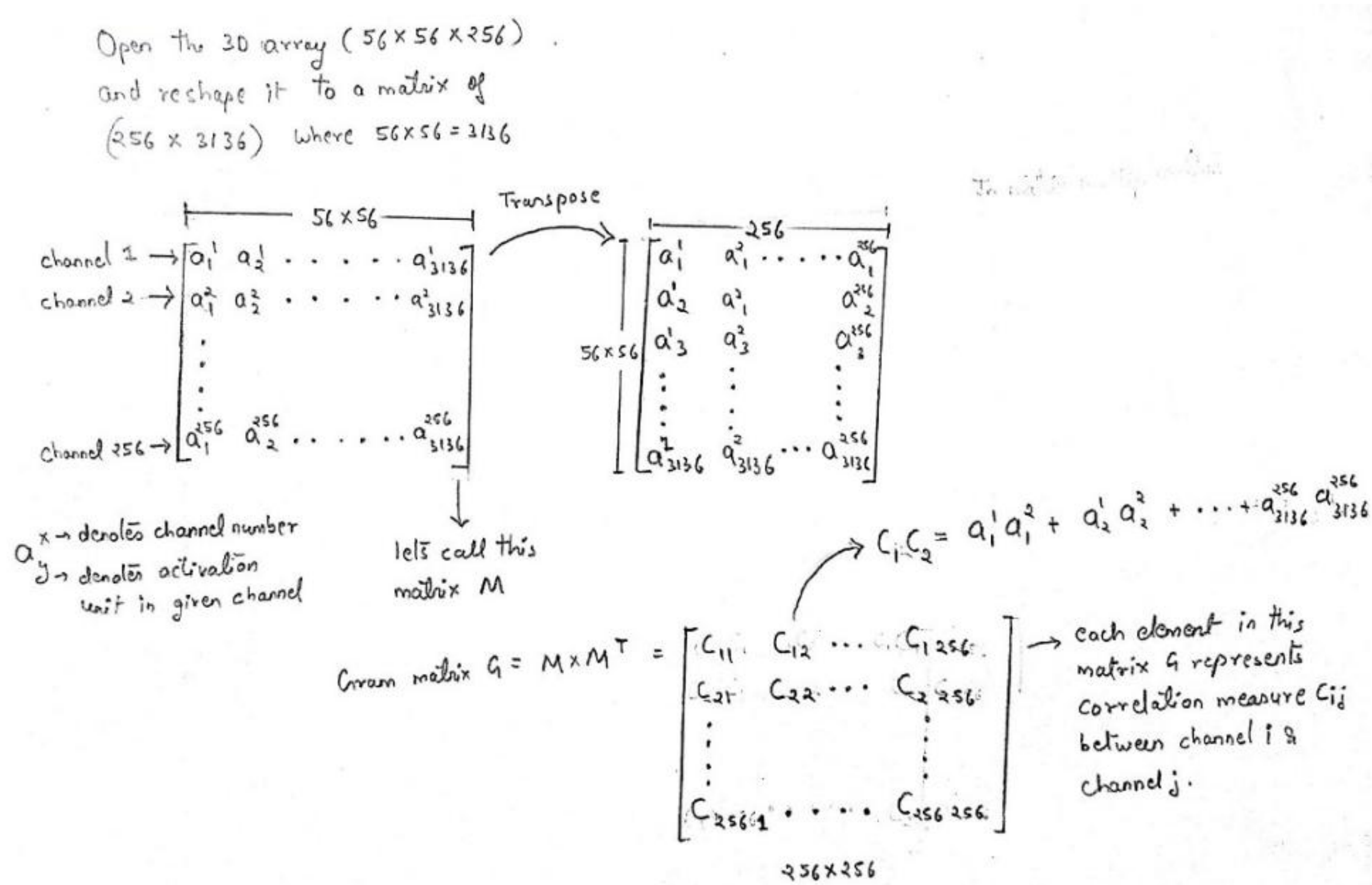
Style Loss

- find the correlation between these activations across different channels of the same layer and to do this we need something called as the Gram Matrix
- What is the **Gram Matrix**?
    - let's consider that we pass our style image though vgg16 and we get the activation values from 7th layer which generates the feature representation matrix of size 56x56x256



- example:
    - channel 'A' activates on receiving brown stripped patterns
    - channel 'B' activates on receiving a eye shape
    - if channel 'A' and 'B' get activated together, its' very likely that the input image is of a tiger
    - this implies channel 'A' and 'B' are highly correlated
    - the gram matrix allows us to find the correlation of all channels with respect to each other

- How to compute the Gram Matrix:



- denote the gram matrix of style image of layer L as GM[L](S)
- and gram matrix of generated image of same layer as GM[L](G)
- Now if we find sum of square difference or L2_norm of element subtraction of these two matrices and try to minimize it, Then this will eventually lead to minimizing the difference between the style of style image and the generated image

$$L_{GM}(S,G,l) = \frac{1}{4N_l^2 M_l^2} \sum_{ij} (GM[l](S)_{ij} - GM[l](G)_{ij})^2$$

Optimization Problem

- The aim is to setup an optimisation problem that aims to solve for a *combination image* that contains the content of the content image, while having the style of the style image.
- once you have both content and style loss add them up and use any optimizer to perform gradient descent to change generated image such that it decreases its loss after each iteration