IITB EE 782 Advanced Machine Learning Project Report
IHC Style Transfer to H&E-stained Histological Images

Amit Lohan
183079033
amitlohan@ee.iitb.ac.in

Pankaj Singh
183079036
pankajsingh@ee.iitb.ac.in

Sourabh Patil
183079002
sourabhpatil@ee.iitb.ac.in

***Abstract*—In the recent years, Convolutional Neural Networks (CNNs) have been demonstrated of being capable of creating artistic imagery by separating and recombining image content and style. This process of using CNNs to render a content image in different styles is referred to as Neural Style Transfer (NST). NST has become a trending topic both in academic literature and industrial applications. In this work, we aim to utilise the concept of Neural Style Transfer in medical images to transfer the Immunohistochemistry style to Haemotoxylin and Eosin stained histological images for the purpose of obtaining IHC stained images from low cost H&E stained histological images.**

***Index Terms*—Neural style transfer (NST), convolutional neural network (CNN)**

I. Introduction

Transferring the style from one image onto another can be considered a problem of texture transfer. In texture transfer the goal is to synthesise a texture from a source image while constraining the texture synthesis in order to preserve the semantic content of a target image. H&E and IHC stained images are one such pairs of images where the content and structure of images is almost similiar but with different textures. H&E staining cost around $5 a slide whereas producing IHC stained slides are quite costly around $300. CNNs have been used in recent years for achieving such texture and style transfer utilizing the concept of Neural style transfer. If we can synthesize IHC style images using the content and structure from H&E images and style from any IHC stained image then the cost of obtaining IHC style images can be reduced remarkably. In this work we present some such basic experiments and their results which we tried for obtaining the IHC images by transferring style to H&E slides.

II. Neural Style Transfer

A. CNNs and feature extraction

The class of Deep Neural Networks that are most powerful in image processing tasks are called Convolutional Neural Networks. CNN consist of layers of small computational units that process visual information hierarchically in a feed-forward manner. Each layers of units can be understood as a collection of image filters, each of which extracts certain feature from the input image. Thus, the output of a given layer consists of so-called feature maps.

At lower layers, using multiple filters the network may capture simple patterns, say a straight line or horizontal line which may not make sense to us, but is of immense importance to the next layers of the network. As we go more deep layer by layer, for example next layer has almost double the filter of the previous one, we get more and more complex features. Like some part of the shape of object. In more deep layers we might get a face of a dog or wheel of car. This capturing of different simple and complex features is called feature representation. The most important thing to know is that CNN does not know what the image is, but it learns to encode what a particular image represents. This encoding nature is exploited for the the Neural Style Transfer.

B. Capturing content and style of images

Each layer in the network defines a non-linear filter bank whose complexity increases with the position of the layer in the network. When Convolutional Neural Networks are trained for object recognition, they develop a representation of the image that makes object information increasingly explicit along the processing hierarchy. Therefore, along downstream layers of the network, the input image is transformed into representations that are increasingly sensitive to the actual content of the image, but become relatively invariant to its precise appearance or the style. Thus, higher layers in the network capture the high-level content in terms of objects and their arrangements in the input image but do not constrain the exact pixel values of the reconstruction to values in original image. On the other hand, reconstructions from the lower layers simply reproduce the exact pixel values of the original image, We therefore refer to the feature responses in higher layers of the network as the content representation.

To obtain representation of the style of an image, we use a feature space designed to capture texture information. This feature space can be built on top of the filter responses in any layer of the network. It consists of the correlations between the different filter responses, where the expectation is taken over the spatial extent of the feature maps. These feature correlations are given by the Gram matrix. Given a set $\mathbf{V}$ of m vectors, the Gram matrix $\mathbf{G}$ is the matrix of all possible inner products of $\mathbf{V}$, i.e.,

$$g_{\mathrm{ij}} = v_{\mathrm{i}}^{\mathrm{T}} v_j$$

By including the feature correlations of multiple layers, we obtain a stationary, multi-scale representation of the input image, which captures its texture information but not the global arrangement.

Fig. 1. Content Image (IIT Bombay main building)

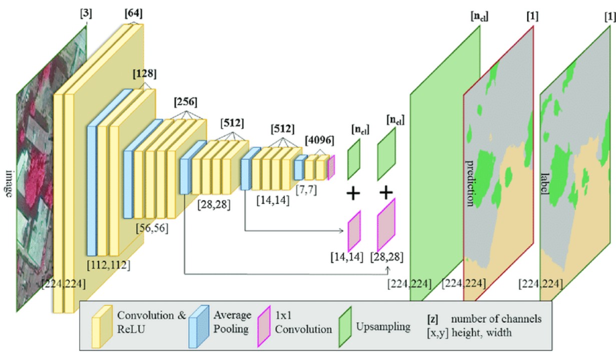
Fig. 2. Style Image (Starry Night by Vincent van Gogh)

Fig. 3. Style Transferred Image

III. Training

For neural style transfer we use a pretrained convolution neural network. Then to define a loss function which blends two images to create an image which has content and style form content and style images respectively, NST defines the following inputs:
A content image (C) — the image we want to transfer a style to
A style image (S) — the image we want to transfer the style from
An input (generated) image (G) — the image that contains the final result (the only trainable variable)

The architecture of the VGG19 model used for neural style transfer in this project is shown below:



A. *Content and Style losses*

1) Content Loss: To calculate the content loss, we take the feature representation of only one of the layers, let's consider 4th convolutional layer of VGG19. To calculate the content loss we pass both content image and generated image through VGG19 and get the activation values (i.e outputs) of 4th convolutional layer for both of these images which has Relu for its activation. Finally we find the L2 Norm of element wise subtraction between these two activation matrices as,

$$\mathbb{L}(C, G, l) = 1/2 \sum (L_l(G) - L_l(C))^2$$

where $L_4(A)$ represents features matrix of image A at the output of Layer l.

This will help to preserve the original content in the generated image by making sure to minimize the difference in feature representation which logically focuses on the difference between content of both the images

2) Style Loss: To calculate style loss we first calculate the gram matrices for outputs of different layers for style and generated images. Style loss for a single layer is given by the following equation.

$$\mathbb{L}_{\mathbb{GM}}(S, G) = (1/4N_l^2 M_l^2) \sum (GM_l(G) - GM_l(S))^2$$

where $GM_l(A)$ represents Gram matrix of image A at the output of Layer l.

While computing style loss we use multiple activation layers, that gives us the flexibility of assigning different weights to each sub loss provided by different layers. So the final style loss is given by:

$$\mathbb{L}_{style}(S, G) = \sum_{i=0}^{L} w_i * L_{GM}(S, G, i)$$

where $GM_l(A)$ represents Gram matrix of image A at the output of Layer l.

IV. Results

Following are some of the results obtained with various settings of hyperparameters and loss functions. Original HE image is shown in Fig. 4 and corresponding IHC image in Fig. 5. Fig. 6 shows the image obtained when weight of content loss while calculating total loss is too high as comared to the style loss. We can see in Fig. 6 that loss of structure occurs when a low weight is given to the content loss. Fig. 7 show the image obtained when weight of style loss is too high as compared to the content loss. Fig. 8 shows the results of using Mean Absolute Error as content loss. We can observe that results obtained with MAE are inferior to resuts obtained with Mean Square Error as content loss. Fig. 9 contains the results when content loss is calculated at higher layers of CNN. In this case also we can see that content of the image is quite different from the original HE or IHC image. In Fig. 10, results obtained when high weight is given to outputs of higher layers for calculating the style loss. Finally, In Fig. 11 is the result of calculating content loss at 4th convolutional layer of CNN and the image obtained is most closest looking to actual IHC image and the best of all the result obtained in our experiment.

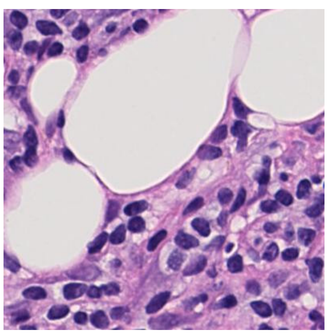


Fig. 4. Original HE image

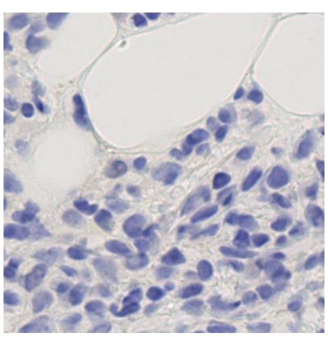


Fig. 5. Corresponding IHC Image

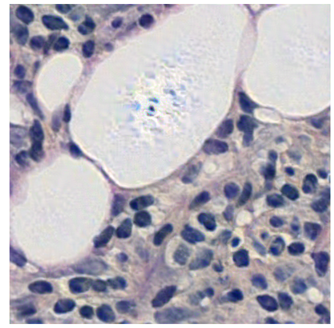


Fig. 6. Low weight of content loss

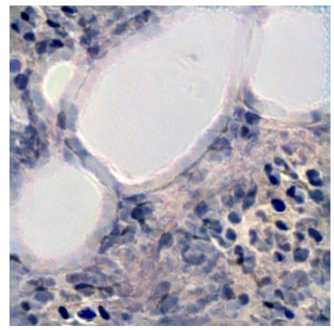


Fig. 7. High weight of style loss

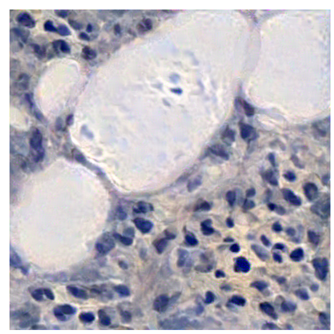


Fig. 8. MAE as content loss

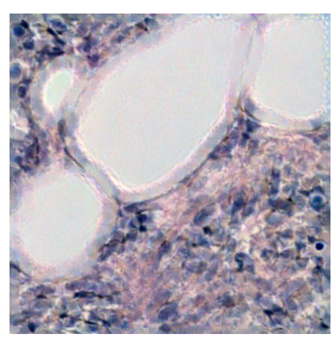


Fig. 9. Content loss at higher layers

V. Conclusion and Future Work

Although the results obtained are not quite usable for any medical application, it can be inferred from results of various experiments that with more efforts and experiments, a style transferred image which looks very much like its HE counterpart can be obtained as results were improved with mean square content loss at middle convolutional layers.

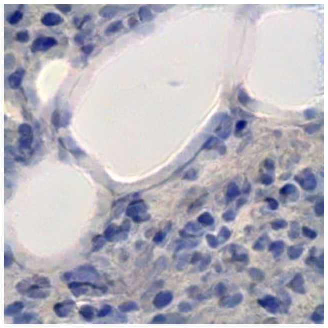


Fig. 10. High weight of style loss at higher layers

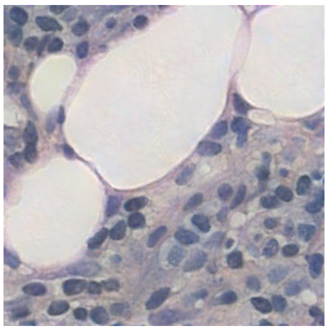


Fig. 11. Content loss at 4th convolutional layer

Also it seems that obtaining IHC style image from HE style image requires some kind of already learnt bias, so network optimization neural style transfer might also get good results.

VI. Link to Code

https://github.com/amitlohan/AML_Project

References

[1] L. A. Gatys, A. S. Ecker and M. Bethge, ”Image Style Transfer Using Convolutional Neural Networks,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016 doi: 10.1109/CVPR.2016.265

[2] Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu and M. Song, ”Neural Style Transfer: A Review,” in IEEE Transactions on Visualization and Computer Graphics. doi: 10.1109/TVCG.2019.2921336

[3] Neural Style Transfer tutorial https://towardsdatascience.com/neural-style-transfer-tutorial-part-1-f5cd3315fa7f

[4] Staining- https://en.wikipedia.org/wiki/Staining

[5] Key Differences of HE and Special Stains for Immunohistochemistry https://info.gbiosciences.com/blog/h-and-e-stain-for-immunohistochemistry