CS256 HW 5

Artistic Style Transfer Using VGG

Aditya Singhania Pradeep Narayana

Introduction:

Rendering the content of an image in multiple different styles is a complex task. The major limitation in the existing approaches is the absence of explicit representation of semantic content. Here, we use the image depictions generated by convolutional neural networks which makes the content explicit. We use a neural algorithm that can distinguish and overlap the style and contents of two images. This method is also known as transfer learning.

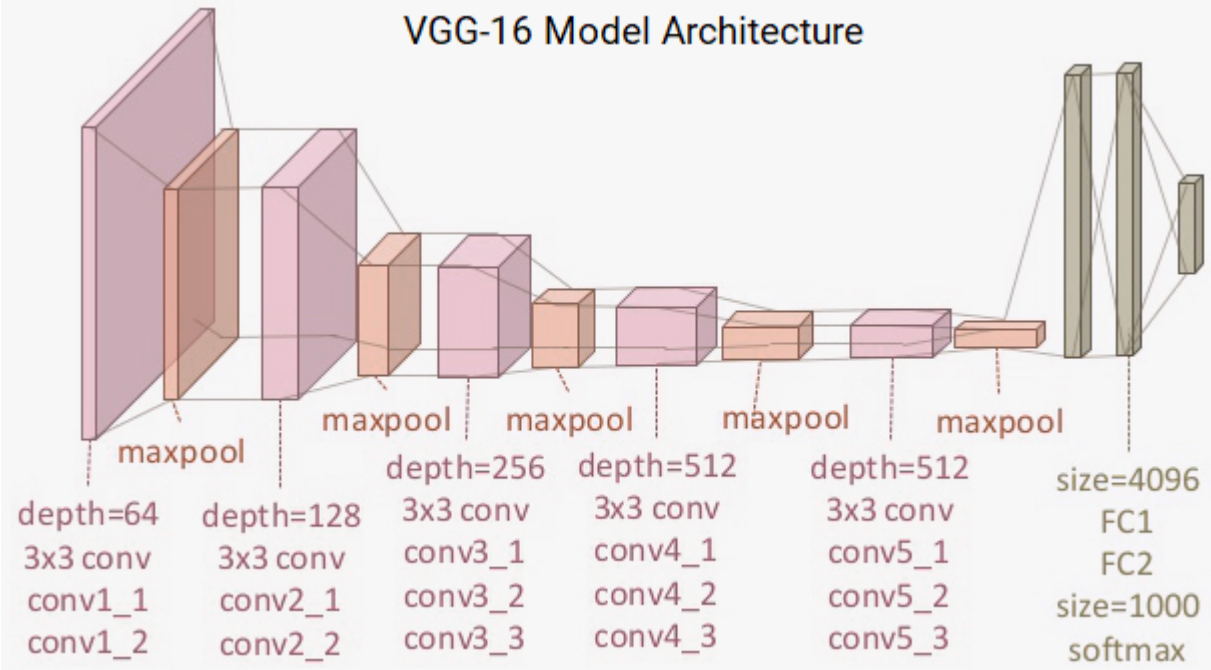
Transfer learning:

Transfer learning is a machine learning technique in which a model created for one job is utilized as the foundation for a model on a different task.
As huge compute and time resources are required to develop neural network models on these problems, transfer learning is a popular approach in deep learning to use pre-trained models as the starting point on computer vision and natural language processing tasks.
In this approach, we train a base network on a base dataset and task, then transfer the acquired features to a second target network to be trained on a target dataset and task. This method is more likely to succeed if the characteristics are generic, i.e., applicable to both the base and target tasks, rather than being specific to the base job.

In a traditional texture transfer method of transfer learning, the texture is extracted from the source image and the semantic content of the target image is preserved. This can be done by various methods such as resampling pixels, by using correspondence maps, etc. However, these methods only use low level features. Hence, Gatys et al.[2] suggested the method where conv1-2, conv2-2, conv3-2 and conv4-2 is used for content reconstruction and conv1-1, conv2-1, conv3-1, conv4-1 and conv5-1 is used for style reconstruction.

VGG:

Visual Geometry Group(VGG) is a multilayer deep Convolutional Neural Network (CNN) architecture where the term "deep" refers to the number of layers in VGG-16 or VGG-19, which have 16 or 19 convolutional layers respectively.
K. Simonyan and A. Zisserman[1] from the University of Oxford proposed the VGG16 convolutional neural network model. In ImageNet, a dataset of over 14 million images belonging to 1000 classes, the model achieves 92.7 percent top-5 test accuracy.
As shown below, VGG16 has 16 layers. In the last three fully connected layers, a succession of VGGs are identical. There are five sets of convolutional layers in total, followed by a MaxPool. The distinction is that in each of the five sets of convolutional layers, more and more cascaded convolutional layers are added.



Hypothesis :

The initial layers have the information of style and deeper layers in the network architecture contain the information regarding the content. By choosing the deeper layers for the content, the output image will be clear with all prominent features of the 'content image' along with the style of the selected 'style image'.

Conducting the experiment:

First we select one image for style and another image for the content as shown below.

Style Image

Content Image

Next, we conduct the experiments by choosing different layers for Style and Content.

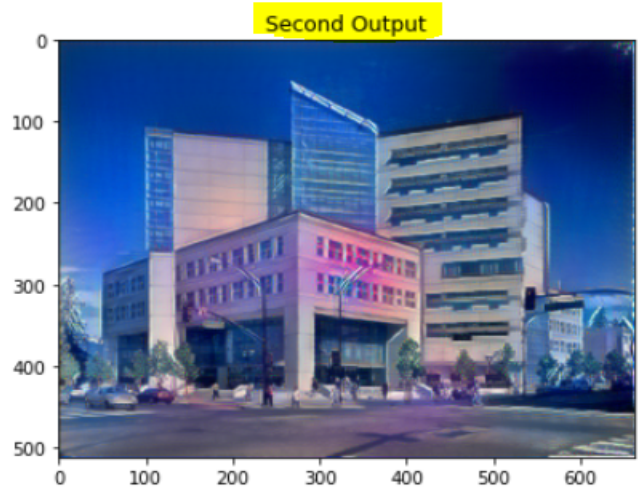
- Firstly we choose 'r11' and 'r21' for style and 'r42' for content.
  style_layers = ['r11','r21']
  content_layers = ['r42']

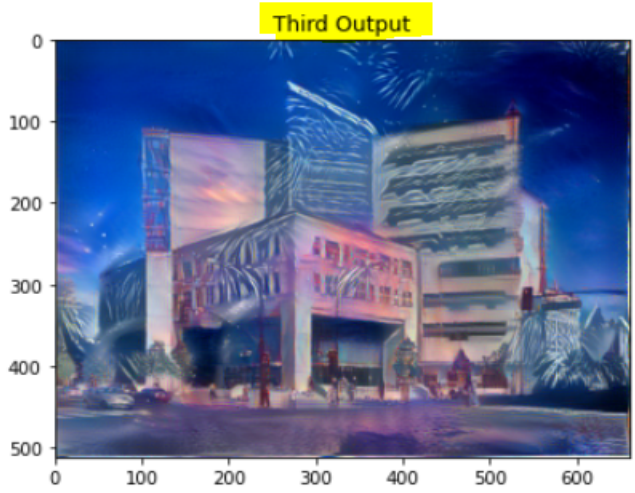
- Next, we increase the number of layers used for extracting style and content. Now we choose r11, r21 and r31 for extracting style and r42 and r32 for extracting content respectively.
  style_layers = ['r11','r21','r31']
  content_layers = ['r42','r32']

- For the third experiment, we use 5 layers r11, r21, r31, r41 and r51 for extracting the style and r42, r32 and r22 for extracting the content.
  style_layers = ['r11','r21','r31',’r41’,’r51’]
  content_layers = ['r42','r32',’r22’]

The corresponding outputs obtained are shown below:

Style layers	Content Layers	Output
'r11','r21'	'r42'	First Output
'r11','r21','r31'	'r42','r32'	Second Output
'r11','r21','r31','r41','r51'	'r42','r32','r22'	Third Output





Observation:

The style of the selected “style image” has been transferred clearly to the output of the first experiment where we use the content of the deep layer. As we use the contents from the middle layers in the second and third experiments, there are more distortions in the output image.

Conclusion :

We can see that in terms of transferring the style, the first output is the best one. The second output would come next and the third output has a lot of distortions.
This is because the initial layers have the information of style and deeper in the network architecture, the hidden layers contain the information regarding the content.
We observe that when we extracted content from the "deep" layers, the building edges were more prominent and similar to the original picture. As in the second and third image, we use layers from the middle as well as deeper end of the network and hence the resultant output had a lot of distortions.

Pre trained vgg weights link - https://www.kaggle.com/yujinozaki/models/version/1

REFERENCES

[1] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” Sep. 2014 [Online]. Available: https://arxiv.org/abs/1409.1556

[2] L. A. Gatys, A. S. Ecker, and M. Bethge, “Image Style Transfer Using Convolutional Neural Networks,” 2016, pp. 2414–2423, doi: 10.1109/CVPR.2016.265 [Online]. Available: https://ieeexplore.ieee.org/document/7780634