Neural Style Transfer Quiz Questions

Yashila Bordag
Division of Computing, Data Science, and Society
Department of Classics
University of California, Berkeley
Berkeley, CA 94720
yashila.bordag@berkeley.edu

William Brandon
Department of Computer Science
Department of Mathematics
University of California, Berkeley
Berkeley, CA 94720
williambrandon@berkeley.edu

Wangda Lei
Department of Computer Science
University of California, Berkeley
Berkeley, CA 94720
radarlwd@berkeley.edu

Let us consider the content loss function presented in the lecture notes and that we used in the assignment:

$$L^l_{content}(\vec{p}, \vec{x}) = \frac{1}{2} \sum_{i,j} (F^l_{ij} - P^l_{ij})^2 \tag{1}$$

$$L_{content}(\vec{p}, \vec{x}) = \sum_{l \in L} L^l_{content}(\vec{p}, \vec{x}) \tag{2}$$

1. What do we use the content loss defined by (1) and (2) for?

We use this loss to calculate the difference between the content features in the content image and the generated image. We can then use the calculated loss to minimize this difference in the neural style transfer algorithm which would allow us to emulate the desired content in the generated image.

2. Calculate the partial derivative of the layer content loss with respect to the activation $F^l_{i^*j^*}$.

$$\begin{aligned}
\frac{\partial L^l_{content}}{\partial F^l_{i^*j^*}} =& \frac{\partial}{\partial F^l_{i^*j^*}} \left(\frac{1}{2} \sum_{i,j} (F^l_{ij} - P^l_{ij})^2 \right) \\
=& \frac{1}{2} \sum_{i,j} \frac{\partial}{\partial F^l_{i^*j^*}} ((F^l_{ij} - P^l_{ij})^2) \\
=& \frac{1}{2} \frac{\partial}{\partial F^l_{i^*j^*}} ((F^l_{i^*j^*} - P^l_{i^*j^*})^2) \\
=& F^l_{i^*j^*} - P^l_{i^*j^*}
\end{aligned}$$

3. Now calculate the partial derivative of the total content loss with respect to the activation $F^l_{i^*j^*}$, using your answer from the last part.

$$
\begin{aligned}
\frac{\partial L_{content}}{\partial F^l_{i^*j^*}} =& \frac{\partial}{\partial F^l_{i^*j^*}} \left(\sum_{l \in L} L^l_{content} \right) \\
=& \sum_{l \in L} \frac{\partial}{\partial F^l_{i^*j^*}} (L^l_{content}) \\
=& \frac{\partial}{\partial F^l_{i^*j^*}} (L^l_{content}) \\
=& F^l_{i^*j^*} - P^l_{i^*j^*}
\end{aligned}
$$

Now let us consider the style loss function presented in the lecture notes and the that we used in the assignment. This loss is markedly different from the content loss defined earlier, as it involves both a Gram matrix and an expectation:

$$G^l_{ij} = \sum_k F^l_{ik} F^l_{jk}, \tag{3}$$

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

$$L_{style}(\vec{a}, \vec{x}) = \sum_{l \in L} w_l E_l \tag{5}$$

4. What do we use the style loss defined by (3), (4), and (5) for?

Similar to the content loss, the style loss is used to calculate the difference between the style features in the style image and the generated image. We can then use the calculated loss to minimize this difference in the neural style transfer algorithm which would allow us to emulate the desired style in the generated image. The style loss is more complicated than the content loss, because style is more heavily dependent on correlations between groups of pixels than the shape of the image as a whole.

Now we give you the partial derivative of the expectation with respect to the activation $F^l_{i^*j^*}$:

$$\frac{\partial E_l}{\partial F^l_{i^*j^*}} = \frac{1}{N_l^2 M_l^2} (F^l)^T (G^l - A^l), \tag{6}$$

5. Now using the derivative of the expectation with respect to $F^l_{i^*j^*}$ given above, calculate the partial derivative of the total style loss with respect to the activation $F^l_{i^*j^*}$.

$$
\begin{aligned}
\frac{\partial L_{style}}{\partial F^l_{i^*j^*}} =& \frac{\partial}{\partial F^l_{i^*j^*}} \left(\sum_{l \in L} w_l E_l \right) \\
=& \sum_{l \in L} \frac{\partial}{\partial F^l_{i^*j^*}} (w_l E_l) \\
=& w_l \frac{\partial}{\partial F^l_{i^*j^*}} (E_l) \\
=& w_1 \frac{1}{N_l^2 M_l^2} (F^l)^T (G^l - A^l)
\end{aligned}
$$

Now recall the total loss function:

$$L_{total}(\vec{p}, \vec{a}, \vec{x}) = \alpha L_{content}(\vec{p}, \vec{x}) + \beta L_{style}(\vec{a}, \vec{x}) \tag{7}$$

6. Finally, using problems 3. and 5. calculate the partial derivative of the total loss with respect to the activation $F^l_{i^*j^*}$.

$$\begin{aligned}
\frac{\partial L_total}{\partial F^l_{i^*j^*}} =& \frac{\partial}{\partial F^l_{i^*j^*}}(\alpha L_{content} + \beta L_{style}) \\
=& \alpha \frac{\partial}{\partial F^l_{i^*j^*}}(L_{content}) + \beta \frac{\partial}{\partial F^l_{i^*j^*}}(L_{style}) \\
=& \alpha(F^l_{i^*j^*} - P^l_{i^*j^*}) + \beta w_1 \frac{1}{N_l^2 M_l^2}(F^l)^T(G^l - A^l)
\end{aligned}$$

Recall the basic gradient descent step:

$$w^{t+1} = w^t - \eta_t \nabla Loss, \tag{8}$$

7. Now using your answer from problem 6. write the update step for $F^l_{i^*j^*}$ with update weight η and image $\vec{x}$.

$$\begin{aligned}
(F^l_{i^*j^*})^{t+1} =& (F^l_{i^*j^*})^t - \eta \frac{\partial L_total}{\partial F^l_{i^*j^*}} \\
=& (F^l_{i^*j^*})^t - \eta\alpha(F^l_{i^*j^*} - P^l_{i^*j^*}) + \beta w_1 \frac{1}{N_l^2 M_l^2}(F^l)^T(G^l - A^l)
\end{aligned}$$

8. Given a pretrained image classification model based on a convolutional neural network architecture, how do we extract the raw content and style features from an image? You don't need to describe how we measure similarity using those features.

Running the network in feed-forward mode on the image will produce a set of feature maps at each layer. The feature maps of earlier layers correspond to more style-like information about the image, while the feature maps of later layers correspond to more content-like information.

9. True or false: Obtaining a model suitable for running the Neural Style Transfer algorithm requires training on a large dataset of triples of the form (content image, style image, combined style-transferred image). Explain your answer.

False. The neural style transfer algorithm requires no additional training data beyond that required to train the pretrained model that it uses for style and content features. The pretrained model does not need to be designed explicitly for style transfer; it can in principle be a model built to perform any task which requires semantic featurization of an image, such as classification.

10. True or false: A Neural Style Transfer model can transfer style from a style source image that was not in its training data set. Explain your answer.

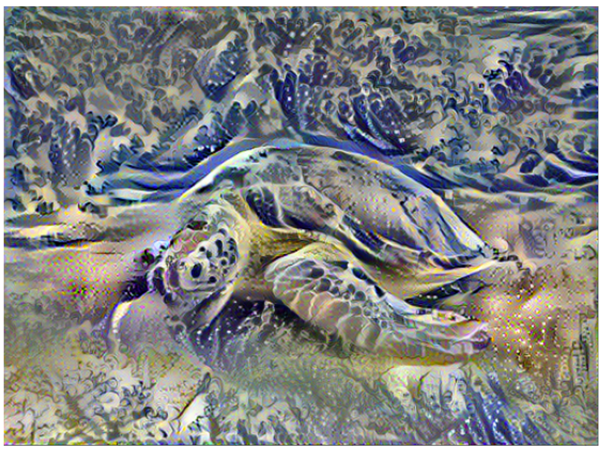
True. The Neural Style Transfer algorithm can use the pre-trained network to obtain style features from any style source image, using only a single example of that style, which can then be used to optimize the generated image to display a similar style.

11. At what steps in the Neural Style Transfer algorithm do we use the content source image?

The content source image is used both to initialize the generated style-transferred image before we optimize it, and at each iteration of the gradient descent optimization process, as part of computing the content loss and the gradient of that loss.

12. Both of the following images were generated by applying Neural Style Transfer to a content source image consisting of a photograph of a sea turtle, using style information from the woodblock print The Great Wave off Kanagawa. One of the images was generated using the first 3 layers of a pretrained convolutional neural network (CNN) to extract style features, while the other image was generated using the first 5 layers of the same network for style features. Which of the two images was generated using 3 layers, and which was generated using 5? Justify your answer.

The first image was generated using 5 layers, and the second was generated using 3. We can identify which image is which based on how "high-level" the style features which have been transferred are. The first image has transferred relatively high-level, large-scale style features such as wave crests, whereas the second image has transferred only small textural patterns. We conclude that the first image used style features which included features from later (and hence higher-level) layers of the pretrained network.



References

[1] Gatys, L. A., Ecker, A. S., Bethge, M. et al. "A neural algorithm of artistic style." arXiv preprint, 2015, `https://arxiv.org/abs/1508.06576`.

[2] Simonyan, K., Zisserman, A. "Very Deep Convolutional Networks for Large-Scale Image Recognition." arXiv preprint, 2015, `https://arxiv.org/pdf/1409.1556`.

[3] Yuan, R. "Neural Style Transfer: Creating Art with Deep Learning using tf.keras and eager execution." Medium, 3, Aug. 2018, `https://medium.com/tensorflow/neural-style-transfer-creating-art-with-deep-learning-using-tf-keras-and-eager-execution-7d541ac31398`.

[4] "Neural style transfer." TensorFlow.org, 2020, `https://www.tensorflow.org/tutorials/generative/style_transfer`.