Neural Style


CSC411/2515: Machine Learning and Data Mining, Winter 2018
Michael Guerzhoy and Lisa Zhang
Neural Style

• Task: given an input photo I and a painting P, produce a photo with the same contents as I, but with the style P

• Idea: use gradient descent again to change the input with the weights of the ConvNet constant, with a cost function that keeps x close to I (i.e., preserve the content) and makes the style of x close to the style of P

• Use a ConvNet that works for image classification
  • It knows how to represent images well
Cost function: content

• To keep the content close to the input I, make sure that $|x - I|^2$ stays small
  • Not the best idea! The pixels might be a completely different colour in the x, but x and I can still be similar in content

• Even better: make sure that all the activations in all the different layers for the original image and for x stay the same:

• Make sure that $L_{content}(x, I) = \sum_{l,i,j}(F(x)^l_{i,j} - F(I)^l_{i,j})^2$ is small
  • $F(y)^l_{(i,j)}$ is the activation at layer $l$ at location $j$ in feature map $i$, for input $y$

• Will make sure that the high-level features in the image stay the same, too
Style

• Define the Gram matrix at layer $l$ as

$$ G_{ij}^l(y) = \sum_k F(y)^l_{ik} F(y)^l_{jk} $$

• Discovery: the Gram matrix represents the style

  • $G_{ij}^l(y)$ is large if at layer $l$, the $i$-th feature and the $j$-th feature tend to be discovered together a lot

    • (i.e., a lot of $k$’s for which the product of both activations is large)
Style Reconstructions

- “Blue strokes at 40 degrees co-occur with blue strokes at 45 degrees?”
Cost function: style

\[ E_l(x, P) = \frac{1}{4N_l^2M_l^2} \sum_{i,j} (G_{ij}^l(x) - G_{ij}^l(P))^2 \]

\[ L_{\text{style}}(x, P) = \sum_l w_l E_l(x, P) \]

Divide by the size of the layer to make sure every layer influences things equally for now.
Cost function: overall

• \( \text{cost}(x, I, P) = \alpha L_{\text{style}}(x, P) + \beta L_{\text{content}}(x, I) \)