Perceptual loss

Состоит из **взвешенной суммы контентного и стилевого** лоссов

$$L(C,S,X) = \alpha L_{\text{content}}(C,X) + \beta L_{\text{style}}(S,X)$$

Контентный лосс

- позволяет определить схожесть двух картинок по контенту
- это сумма всех квадратов разниц пикселей выходов слоя l по двум картинкам (

1) То есть мы прогоняем картинки A и B через нейросеть и берем выходы по обоим картинкам на слое l
2) Далее смотрим на каждый пиксель j каждого фильтра i обеих картинок
3) Получаем сумму квадратов попарных разниц)

Почему это работает? Потому что на выходе l благодаря фильтрам каждый пиксель содержит информацию о соседних пикселях. И даже если у нас объект на первой картинке отодвинулся в сторону относительно второй картинки (тогда попиксельное сравнение не скажет, что у нас схожие объекты), то контентный лосс будет низким, так как на слое l выход по первой картинке имеет схожие пиксели относительно второй картинки

$$L^{\ell}_{\text{content}}(A,B) = \sum_{i,j}\left(A^{\ell}_{ij} - B^{\ell}_{ij}\right)^2$$

Стилевой лосс

- позволяет определить схожесть двух картинок по стилю
    1) Берем скалярные произведения выходов всех возможных пар каналов для одной картинки это G(A)
    2) Для второй - G(B)
    3) Считаем сумму их разниц
    4) Далее суммируем по каждому слою взвешенно

Почему это работает? Мы хотим чтобы элементы, которые улавливали фильтры встречались в обеих картинках, поэтому если оба фильтра что-то уловили в 1 картинке, но только один уловил во 2 картинке, то лосс будет больше, чем в ситуации, когда в оба фильтра уловили информацию и в 1, и во 2 картинке

- Разница в стиле:

$$L^{\ell}_{\text{style}}(A,B) = \sum_{i,j}\left(G^{\ell}_{ij}(A) - G^{\ell}_{ij}(B)\right)^2$$

- $G^{\ell}_{ij}(A) = \sum_k A^{\ell}_{ik}A^{\ell}_{jk}$ — скалярное произведение i-го и j-го каналов в слое ℓ для изображения A

$$L_{\text{style}}(A,B) = \sum_{\ell=0}^{L} w_{\ell} L^{\ell}_{\text{style}}(A,B)$$

Как перенести стиль с S картинки на C?

$$L(C,S,X) = \alpha L_{\text{content}}(C,X) + \beta L_{\text{style}}(S,X)$$

S - стилевое изображение
C - наша картинка
X - предсказание

1) Берем предубоченную модель (VGG-16)
2) Обучаем модель(C) чтобы минимизировать L(C, S, X)
3) Lcontent отвечает за то, чтобы не потеярлись объекты с нашей картинки
4) Lstyle отвечает за то, чтобы стиль S и нашей картинки были как можно ближе

Как ускорить?

- стиль будет одним и тем же

1) С помощью способа выше можно собрать обучающую выборку X по стилю S
2) Далее обучаем простую модель где на вход подаем C и получаем X