



# Перенос стиля фотографии с использованием глубоких нейронных сетей

Павел Пронченко

# 1. ПОСТАНОВКА ЗАДАЧИ



(a) Input image



(b) Reference image



(c) Our result



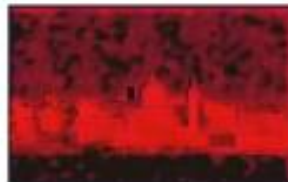
Content  
Image



Styling  
Image



Traditional  
ST

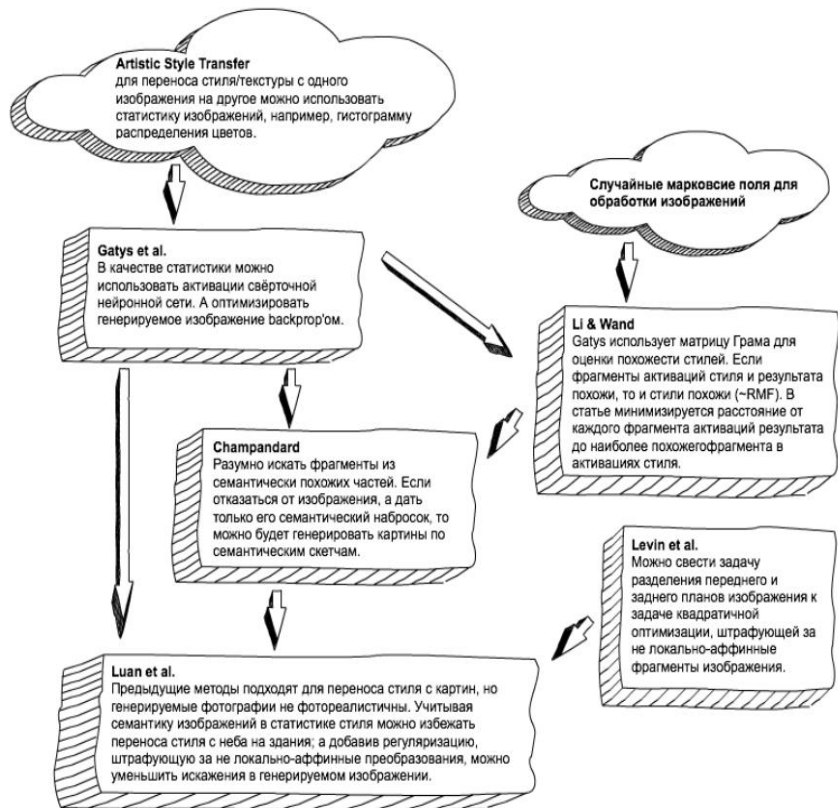


Using Seg-  
mentation



## 2. АНАЛИЗ

# КАКИЕ ЕСТЬ АНАЛОГИ



# —

## 3. МЕТОДИКА РЕШЕНИЯ

# АЛГОРИТМ ГЭТИСА

Style image



Output image



Content image



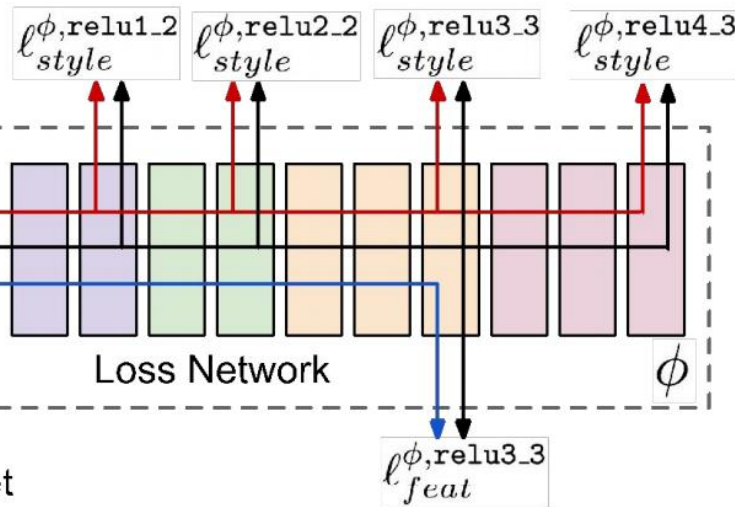
Style Target

$y_s$

$\hat{y}$

Content Target

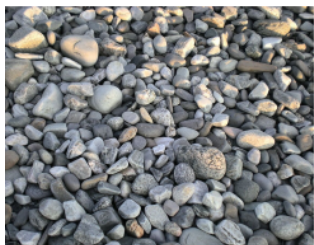
$y_c$



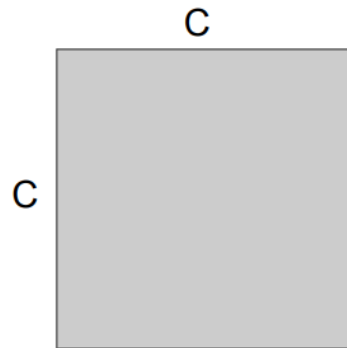
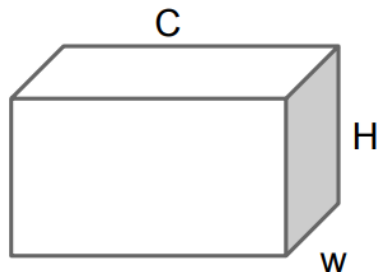
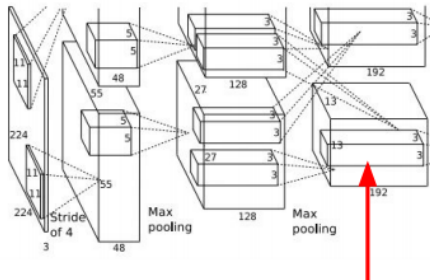
$$\mathcal{L}_{\text{content}}(\vec{p}, \vec{x}, \ell) = \sum_{i,j} (F_{i,j}^{\ell} - P_{i,j}^{\ell})^2$$

$$\mathcal{L}_{\text{style}}^{+} = \sum_{k \in \text{segments}} \sum_{l \in \text{layers}} \sum_{i,j} (G_k^{\ell} - A_k^{\ell})_{ij}^2$$

# МАТРИЦА ГРАМА



[This image](#) is in the public domain.



Каждый слой CNN представляет собой тензор размерности  $C \times H \times W$  ;

Его можно представить в виде матрицы  $H \times W$   $C$ -мерных векторов  
Произведение двух  $C$ -мерных векторов даст нам  $C \times C$  матрицу корреляции.

Среднее по всем парам  $HW$  векторов представляет собой  
Матрицу Грама размерностью  $C \times C$

Эффективно для вычисления;

Можно изменить размерность

$$C \times H \times W = C \times HW$$

$$\text{Вычислить } G = F \cdot F^T$$



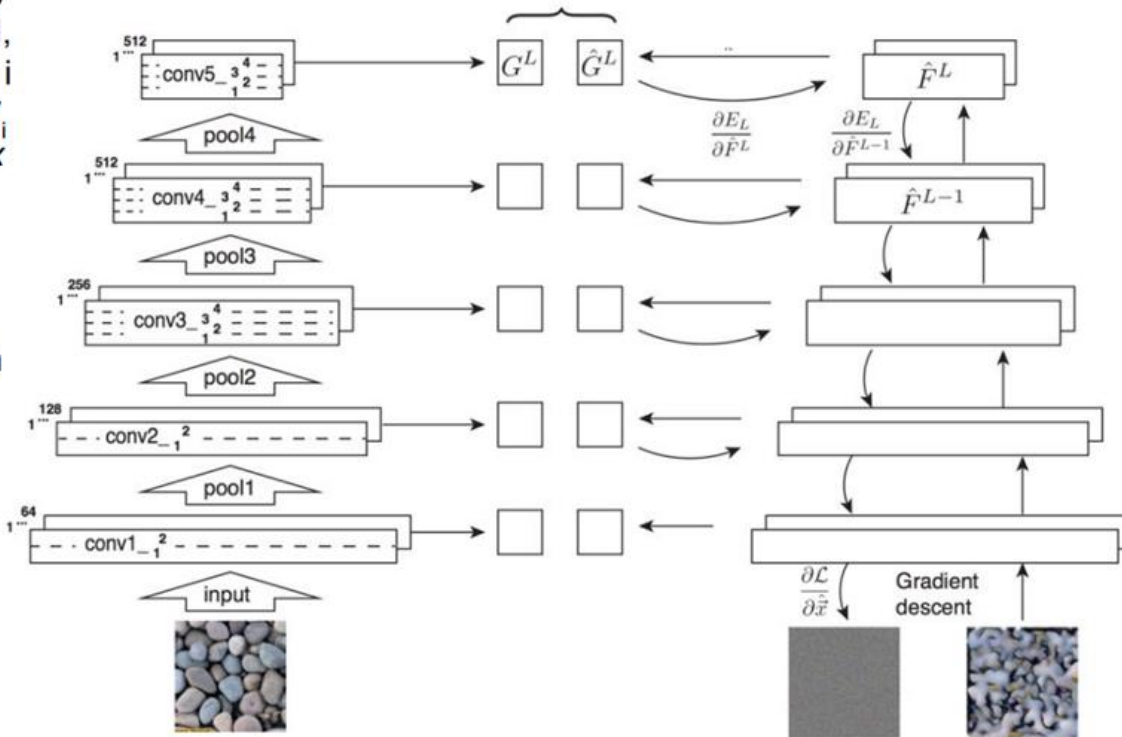
# NEURAL STYLE TRANSFER

1. Pretrain a CNN on ImageNet (VGG-19)
2. Run input texture forward through CNN, record activations on every layer; layer  $i$  gives feature map of shape  $C_i \times H_i \times W_i$
3. At each layer compute the *Gram matrix* giving outer product of features:

$$G_{ij}^l = \sum_k F_{ik}^l F_{jk}^l \text{ (shape } C_i \times C_i \text{)}$$

4. Initialize generated image from random noise
5. Pass generated image through CNN, compute Gram matrix on each layer
6. Compute loss: weighted sum of L2 distance between Gram matrices
7. Backprop to get gradient on image
8. Make gradient step on image
9. GOTO 5

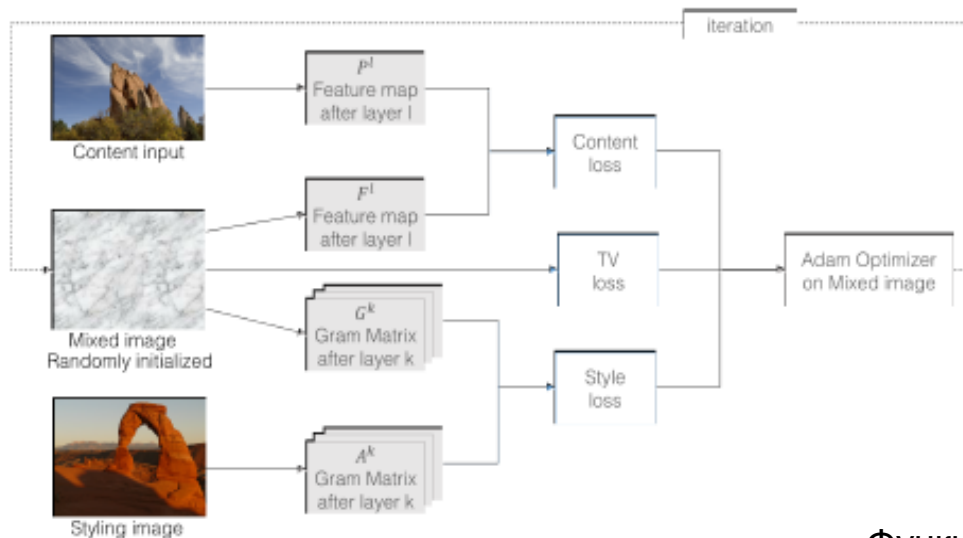
$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G_{ij}^l - \hat{G}_{ij}^l)^2 \quad \mathcal{L}(\vec{x}, \hat{\vec{x}}) = \sum_{l=0}^L w_l E_l$$



# АЛГОРИТМ LUAN

Общая функция потерь для  
фотографического переноса стиля:

$$\mathcal{L}_{\text{tot}}(\vec{p}, \vec{a}, \vec{x}) = \alpha \mathcal{L}_c(\vec{p}, \vec{x}) + \mathcal{L}_s^+(\vec{a}, \vec{x}) + \gamma \mathcal{L}_{\text{tv}}(\vec{x})$$



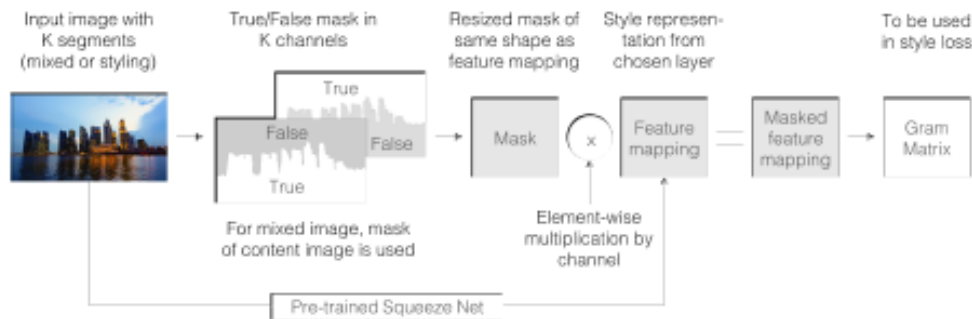
$$\mathcal{L}_{\text{content}}(\vec{p}, \vec{x}, \ell) = \sum_{i,j} (F_{i,j}^{\ell} - P_{i,j}^{\ell})^2$$

$$\mathcal{L}_{\text{style}}^+ = \sum_{k \in \text{segments}} \sum_{l \in \text{layers}} \sum_{i,j} (G_k^{\ell} - A_k^{\ell})_{ij}^2$$

Функция потерь вариаций. Обеспечивает плавность  
изменчивости между соседними пикселями:

$$\mathcal{L}_{\text{tv}}(\vec{x}) = \sum_{c=1}^3 \sum_{i=1}^{H-1} \sum_{j=1}^{W-1} (x_{i,j+1,c} - x_{i,j,c})^2 + (x_{i+1,j,c} - x_{i,j,c})^2$$

# СЕМАНТИЧЕСКАЯ СЕГМЕНТАЦИЯ



Content Image



Styling Image

Доп. функция фотореализма для регуляризации краев:

$$\mathcal{L}_m = \sum_{c=1}^3 V_c[O]^T \mathcal{M}_I V_c[O]$$

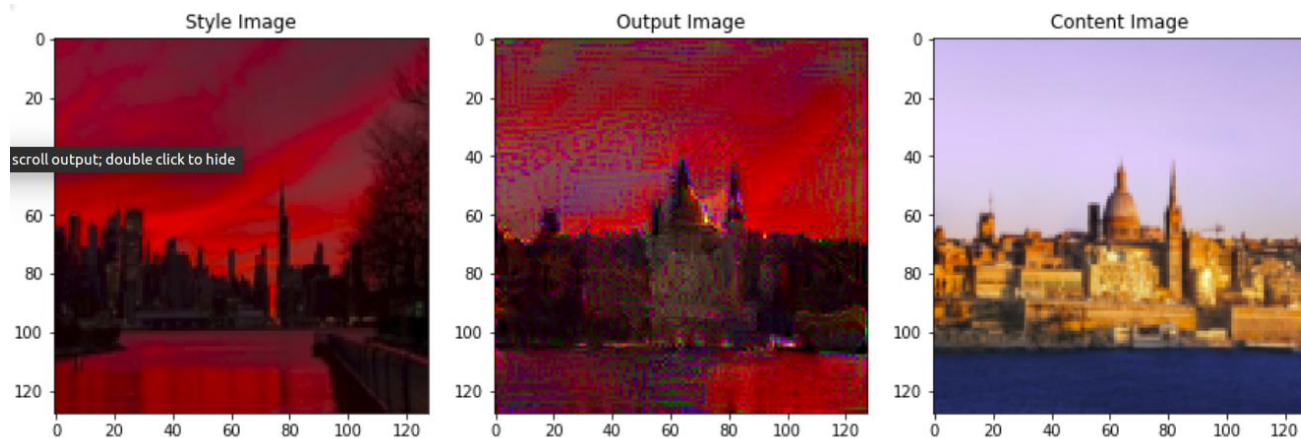
где  $V_c[O]$  - векторизованная версия текущего выходного изображения, а  $\mathcal{M}_I$  - матрица Лапласа ([Кирхгофа](#)) исходного изображения контента



---

## 4. РЕЗУЛЬТАТЫ

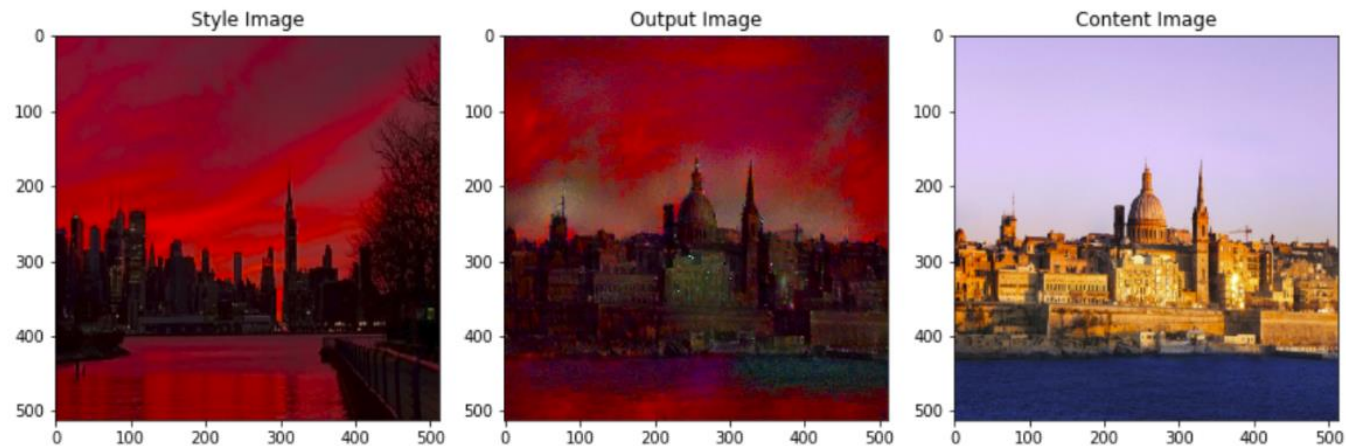
# РЕЗУЛЬТАТЫ РАБОТЫ МОДЕЛИ



Подбор параметров  
обучения на  
картинках с низким  
разрешением 128  
на 128



# РЕЗУЛЬТАТЫ РАБОТЫ МОДЕЛИ



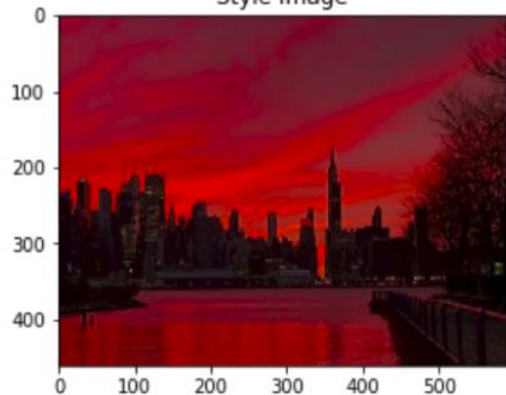
Обучение  
2000 эпох, 512  
на 512



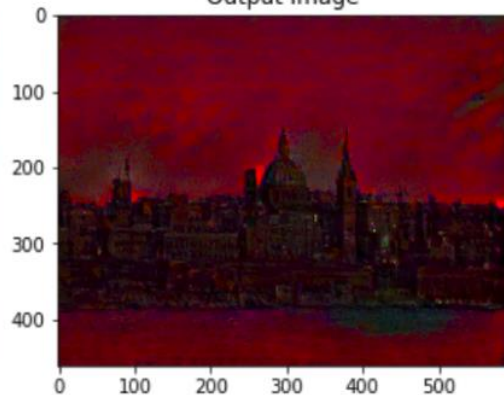


# РЕЗУЛЬТАТЫ РАБОТЫ МОДЕЛИ

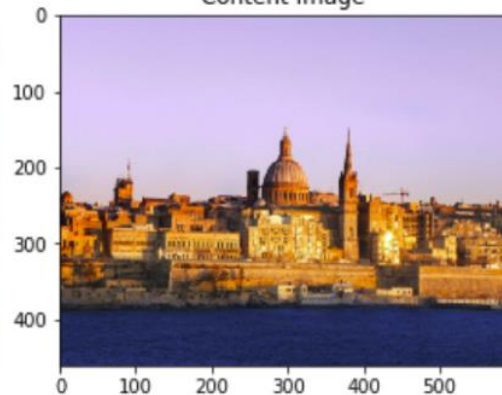
Style Image



Output Image



Content Image



Обучение  
15000 эпох,  
461 на 590.

Время  
обучения на  
GTX 1050Ti:  
3 часа



---

## 5. ЗАКЛЮЧЕНИЕ



# ВЫВОДЫ

Удалось решить задачу переноса стиля фотографии на PyTorch:

<https://github.com/ppronchenko/study/blob/master/Diplom-Project/>

Фотографический перенос стиля требует долгого подбора параметров и особой стратегии тренировки

Валидация «на глаз»

Перенос стиля довольно ресурсоемкий процесс и для картинок высокого разрешения требует производительных видеокарт

# КУДА ДАЛЬШЕ

- Попробовать обучать в облаке с высоким разрешением картинок на более производительных GPU
- Ускорение переноса стиля: Fast Style Transfer
- Генерация стиля из нескольких изображений посредством вариационных автоэнкодеров (VAE)
- Генерация изображений генеративными состязательными сетями (GANs)



**Спасибо за внимание!**

Павел Пронченко