

Оптимизационный перенос стиля

Виктор Китов

v.v.kitov@yandex.ru

Содержание

- 1 Введение
- 2 Оптимизационный перенос стиля

Нейросетевой перенос стиля

- вход: изображение с контентом и стилем.
- Перенос стиля (style transfer) - перерисовка контентного изображения в стиле стилевого изображения.
- Реализуется с помощью сверточных нейронных сетей.

Пример



Приложения

- **Улучшить социальную коммуникацию**
 - добавить эмоции, личное восприятие в изображения
- **Расширение инструментов рисования**
 - добавлять сложные узоры к тексту и рисункам
 - полезно для художников, дизайнеров, архитекторов, рекламщиков.
- **Применение спецэффектов** к фильмам, компьютерным играм, дополненной реальности
 - интерактивный стиль, в зависимости от ситуации
- **Дешевое создание мультфильмов**
 - из отснятых сцен с участием актеров
- **Улучшенная подгонка модели**
 - увеличение данных путем случайной стилизации обучающих данных
- **Трансферное обучение**, например
 - детектирование лиц на картинах: фотографии->картины
 - сегментация: дневные фотографии -> ночные фотографии

Стилизация в других областях

Стилизация может применяться и в других областях:

- стилизация видео
- стилизация речи
- стилизация музыки (перевод в другой жанр, музыкальное оформление)
- стилизация текста (неформальный текст->в формальный)

Содержание

- 1 Введение
- 2 Оптимизационный перенос стиля

Оптимизационный перенос стиля¹

- Рассмотрим сверточную нейросеть VGG.
- Контент восстанавливается из активаций на слое I_c .
- Стиль восстанавливается по ковариациям на слоях $I \in L_s$

¹Gatys et al (2015). A Neural Algorithm of Artistic Style.

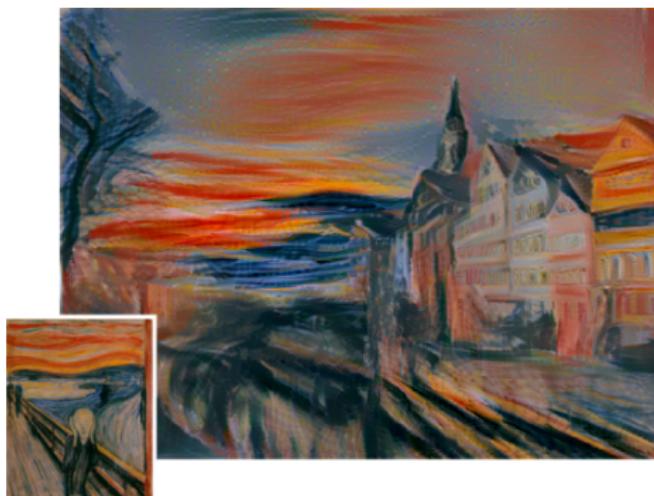
Изображение контента



Применение стиля



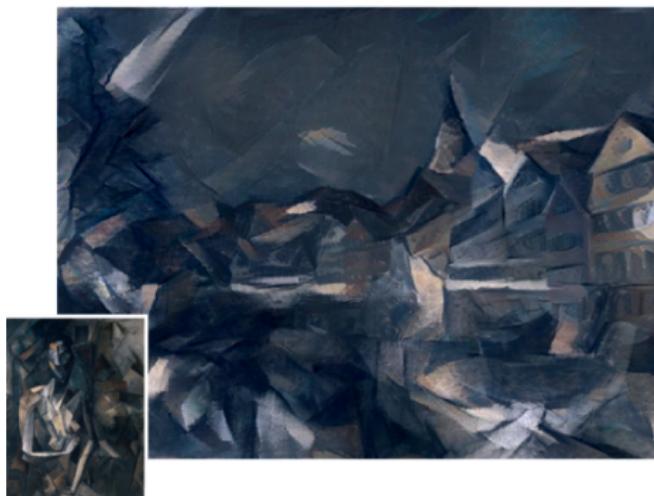
Применение стиля



Применение стиля



Применение стиля

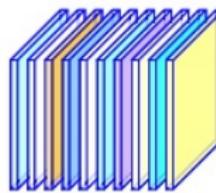


Обозначения

- Изображения: x_c - контент, x_s - стиль, x - стилизованное изображение
- $\Phi_{cij}^l(x) \in \mathbb{R}^{C_l \times H_l \times W_l}$ - промежуточное представление x на слое l .
- Определим матрицы Грамма $\in \mathbb{R}^{N_l \times N_l}$:

$$G_{uv}^l(x) = \frac{1}{H_l W_l} \langle \Phi_{u::}^l(x), \Phi_{v::}^l(x) \rangle = \frac{1}{H_l W_l} \sum_{i=1}^{H_l} \sum_{j=1}^{W_l} \Phi_{uji}^l(x) \Phi_{vij}^l(x)$$

- нечентрированные ковариации² между каналами
- пространственная информация агрегируется



² Центрирование приводит к аналогичным результатам.

Потери

- Потери контента (высокоуровневые признаки д. быть похожими):

$$\mathcal{L}_{content}^I(x, x_c) = \frac{1}{W_I H_I C_I} \left\| \Phi^I(x) - \Phi^I(x_c) \right\|_F^2$$

- Стилевые потери для одного слоя (ковариации между признаками д. быть похожи):

$$\mathcal{L}_{style}^I = \frac{1}{C_I^2} \left\| G^I(x) - G^I(x_s) \right\|_F^2$$

- Полные потери:

$$x = \arg \min_x \left\{ \mathcal{L}_{content}^{I_c}(x, x_c) + \alpha \sum_{I \in L_s} w_I \mathcal{L}_{style}^I(x, x_s) + \beta \mathcal{L}_{tv}(x) \right\} \quad (1)$$

Повышение важности контента

$\uparrow \alpha$ приводит к доминированию стиля над контентом.



1



2



3



4

Потери полной вариации

- Потери полной вариации (total variation) поощряют гладкость результата:

$$\mathcal{L}_{tv}(x) = \sum_{i,j} (x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2$$

- Альтернативы:

$$\mathcal{L}_{tv}(x) = \sum_{i,j} |x_{i+1,j} - x_{i,j}| + |x_{i,j+1} - x_{i,j}|$$

$$\mathcal{L}_{tv}(x) = \sum_{i,j} \left((x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2 \right)^\gamma, \quad \gamma > 0$$

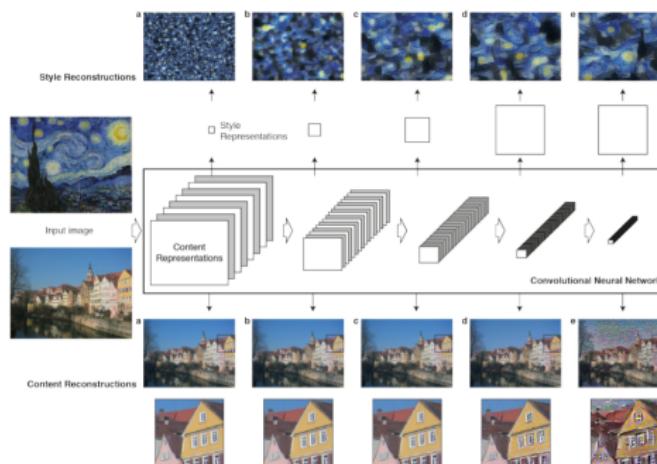
- В переносе стиля обращаем
max-pooling=>недоопределенная задача
- Добавление \mathcal{L}_{tv} делает ее более определенной, добавляя требование гладкости.

Обсуждение

- $\alpha > 0$: сила стилизации
- $\beta > 0$: сила сглаживания
- Выше $l \Rightarrow$ сохраняются более абстрактные свойства контента/стиля.
- Контентные потери используют один промежуточный слой I_c
 - компромисс между гибкостью и распознаваемостью контента
- Стилевые потери соответствуют нескольким слоям
 - ранние: низкоуровневая пиксельная информация
 - поздние: высокоуровневая информация
- Инициализация x :
 - случайная (может сходиться к немного разным решениям)
 - инициализировать $x = x_c$ (значительно быстрее сходится)

Восстановление по контентной/стилевой информации

Восстановление по контенту (нижний ряд) и стилю (верхний ряд) по слоям с $\uparrow l$:



Более глубокие слои содержат более общую информацию.