

Улучшение качества цифровых снимков с помощью глубоких сверточных нейронных сетей

Игнатов Андрей

Московский физико-технический институт
Факультет управления и прикладной математики
Кафедра интеллектуальных систем

Научный руководитель: д.ф.-м.н., профессор РАН К. В. Воронцов

Качество **мобильных** фотографий по-прежнему значительно уступает качеству фотографий с **зеркальных** фотоаппаратов:

Sony Xperia Z



BlackBerry Passport



iPhone 3GS



Canon DSLR




Возможно ли улучшить фотографии низкого качества и получить что-то вроде этого?





До



После



Улучшение изображений :

-  **super-resolution:** *восстановление изображений высокого разрешения*
-  **deblurring/dehazing:** *удаление искусственного размытия*
-  **denoising/sparse inpainting:** *удаление искусственных шумов*
-  **colorization:** *раскраска черно-белых изображений*

Почему бы поочередно не применить эти решения к изображению?

- Они убирают только конкретные **искусственно** добавленные артефакты
- Артефакты на реальных снимках значительно более сложные:

их моделирование в виде искажений не представляется возможным

Как сгенерировать данные для обучения?

- Не можем ухудшать хорошие изображения:

решение не будет обобщаться на реальные артефакты

- Не можем сгенерировать целевые улучшенные фото из имеющихся:

современные методы не способны сильно улучшить резкость и текстуру изображений, или добавить оттенки цветов потерянные сенсором камеры

В данной работе предлагается другой способ:

- фотография с **мобильного устройства** преобразуются в фотографию той же сцены снятой **зеркальным фотоаппаратом**

→ восстановление функции преобразования между:
исходным распределением, заданным **сенсором мобильной камеры**,
и целевым распределением, заданным **сенсором зеркального фотоаппарата**

Постановка задачи

- По исходной фотографии низкого качества \mathbf{I}_s требуется восстановить изображение \mathbf{I}_t , полученное зеркальной камерой
- Сверточная сеть (CNN) \mathbf{F}_W , параметризованная весами W , обучается восстановлению данной функции преобразования
- Пусть дана обучающая выборка $\{\mathbf{I}_s^j, \mathbf{I}_t^j\}_{j=1}^N$, состоящая из N пар данных изображений, тогда CNN обучается минимизировать:

$$W^* = \arg \min_W \sum_{j=1}^N \mathcal{L}(\mathbf{F}_W(\mathbf{I}_s^j), \mathbf{I}_t^j),$$

где \mathcal{L} обозначает функцию ошибки для данной задачи

Характеристики камер

Камера	Сенсор	Разрешение снимков	Качество снимков
<i>iPhone 3GS</i>	3 MP	2048 × 1536	Низкое (1)
<i>BlackBerry Passport</i>	13 MP	4160 × 3120	Ниже среднего (3)
<i>Sony Xperia Z</i>	13 MP	2592 × 1944	Выше среднего (4)
<i>Canon 70D DSLR</i>	20 MP	3648 × 2432	Отличное (5)





Технические детали

- Все устройства были смонтированы на одном штативе
- Активация камер происходила синхронно при помощи удаленной беспроводной системы управления

Обучающая выборка

- Две недели сбора данных
- В общей сложности более 22 тысяч снимков:

6064 пар для BlackBerry – Canon

6064 пар для iPhone – Canon

4549 пар для Sony – Canon



Выравнивание Фотографий

Объекты на фотографиях расположены по-разному:
у камер различные **углы обзора**, **положение** и **поле зрения**

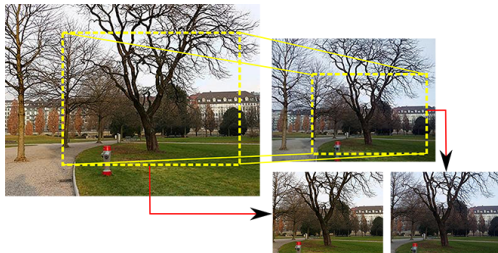


Двухшаговое выравнивание фотографий:

1. Примерное выравнивание с использованием признаков **SIFT**
2. Точное сопоставление патчей на основе **кросс-корреляции**

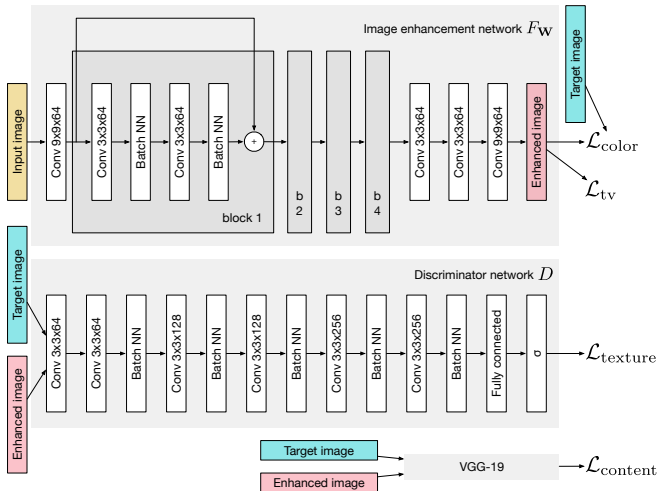
Выравнивание Фотографий

- Выравнивание на основе **SIFT**



- Выравнивание на основе **кросс-корреляции**
скользящее окно размером 100×100 px **параллельно движется** по полученной паре изображений
его положение на фотографии с зеркалки дополнительно регулируется с помощью **сдвигов и поворотов** исходя из максимизации **кросс-корреляции** для рассматриваемых фрагментов изображений (патчей)
- **65K, 92K и 97K** пар фрагментов (патчей) для Sony, BlackBerry и iPhone
- Эти патчи размером 100×100 px являются входными данными для CNN

Общая архитектура



- Даже после точного сопоставления патчи **не выровнены пиксель-в-пиксель**: различающаяся оптика и сенсоры привносят нелинейную локальную **дисторсию** и **абберацию**, которые приводят к сдвигу между изображениями

Стандартные по-пиксельные ошибки не подходят для данной задачи

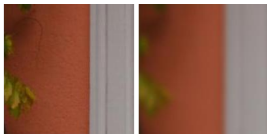
- Функция ошибки строится в предположении что общее качество изображения может быть разложено на три основных компонента:
 1. **Цвет**
 2. **Текстура**
 3. **Контент**
- Для каждого компонента предлагается своя функция ошибки, инвариантная к локальным сдвигам по построению

1. **Ошибка цвета:** **Гауссовское размытие** применяется к паре изображений, затем вычисляется **Евклидово расстояние** между полученными представлениями изображений:

$$\mathcal{L}_{\text{color}}(X, Y) = \|X^b - Y^b\|_2^2,$$

$$X_{i,j}^b = \sum_{k,l} X_{i+k,j+l} \cdot G_{k,l}, \quad G_{k,l} = A \exp\left(-\frac{(k - \mu_x)^2}{2\sigma_x} - \frac{(l - \mu_y)^2}{2\sigma_y}\right),$$

где X^b и Y^b размытые изображения X и Y , $G_{k,l}$ – Гауссовское размытие

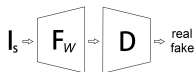


- **Идея:** оценить различие **яркости**, **контраста** и основных **цветов** между изображениями, исключая из сравнения текстуру и контент
- **Свойство:** **инвариантность** к локальным искажениям

2. **Ошибка текстуры:** метрика для оценки качества текстуры выучивается с помощью **Generative Adversarial Networks**¹

Обозначения: фото с зеркального фотоаппарата I_t – **настоящее**
изображение $F_W(I_s)$, улучшенное CNN – **фальшивое**

Discriminator CNN – предсказывает, является ли поданное изображение настоящим или фальшивым



Ошибка текстуры – стандартная ошибка сети-генератора:

$$\mathcal{L}_{\text{texture}} = - \sum \log D(F_W(I_s))$$

¹ Goodfellow et al., Generative adversarial nets, 2014

3. Ошибка контента: используются значения в слоях активации предобученной сверточной сети VGG-19

Идея – изображения должны иметь схожее признаковое описание, которое содержит различные аспекты их **контента** и **визуального качества**

Обозначения: $\mathbf{F}_W(\mathbf{I}_s)$ – улучшенное изображение, \mathbf{I}_t – целевое
 $\psi_j(\cdot)$ – признаковое описание, полученное после j -ого слоя сети VGG-19

$$\mathcal{L}_{\text{content}} = \frac{1}{C_j H_j W_j} \|\psi_j(\mathbf{F}_W(\mathbf{I}_s)) - \psi_j(\mathbf{I}_t)\|$$

- **Финальная ошибка:** взвешенная сумма предыдущих ошибок:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{color}} + \mathcal{L}_{\text{texture}} + 10^{-1} \cdot \mathcal{L}_{\text{content}} + 10^2 \cdot \mathcal{L}_{\text{tv}},$$

где \mathcal{L}_{tv} – ошибка полной вариабельности, используемая для регуляризации:

$$\mathcal{L}_{\text{tv}} = \|\nabla_x \mathbf{F}_W(\mathbf{I}_s) + \nabla_y \mathbf{F}_W(\mathbf{I}_s)\|$$

Исходное (iPhone)



Улучшенное



Исходное (BlackBerry)



Улучшенное



Исходное (Sony)



Улучшенное



Вычислительный эксперимент

Задача очень субъективна, трудно оценить количественно

- **Субъективная оценка:** опрос пользователей
- **Объективная оценка:** количественное сравнение с альтернативными подходами

ДО



ПОСЛЕ



- **Apple Photo Enhancer (APE)** коммерческий продукт, дающий одни из наиболее качественных результатов, алгоритм не опубликован
- **Dong et al.²** фундаментальная работа по super-resolution, основанная на 3-х слойной сверточной сети и среднеквадратичной функции ошибки
- **Johnson et al.³** один из наиболее современных подходов к задачам super-resolution и image style transfer. Метод основан на сверточной сети с четырьмя остаточными блоками, которая обучается минимизировать функцию ошибки на основе активаций сети VGG-19
- **Ручное улучшение.** Графического художник улучшал изображения с помощью профессионального программного обеспечения (Adobe Photoshop CS6). Требовалось улучшить цвета, четкость и общий вид фотографий

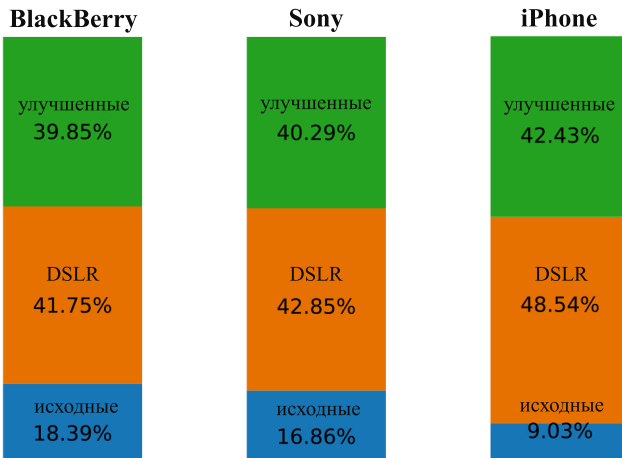
²Learning a Deep CNN for Image Super-Resolution. ECCV 2014

³Perceptual Losses for Real-Time Style Transfer and Super-Resolution. ECCV 2016

Опрос пользователей

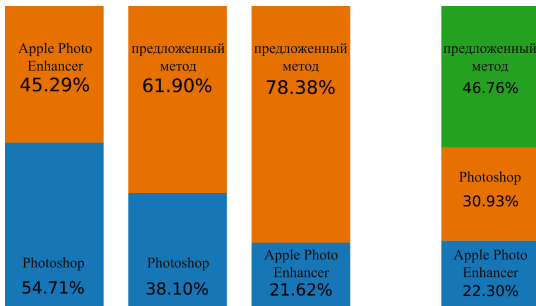
- Пользователю на экране показывались две фотографии, из которой он должен выбрать лучшую
- Рассматривались попарные сравнения для трех классов:
 - 1) исходные фотографии с телефонов
 - 2) фотографии, улучшенные предложенным методом
 - 3) фотографии с зеркальной камеры
- 9 фотографий, все возможные попарные комбинации, 81 вопрос
- 42 участника





Для фотографий с **iPhone** тем же 42 пользователям требовалось дополнительно сравнить фотографии, улучшенные:

- предложенным методом
- вручную, с помощью Photoshop
- с помощью Apple Photo Enhancer



Количественная оценка

Исходное изображение	Apple Photos Enhancer	Результаты Dong et al.
Результаты Johnson et al.	Результаты предложенного метода	Фотография с зеркалки



Средние значения индексов **PSNR** (peak signal-to-noise ratio) и **SSIM** (structural similarity), полученные на тестовых изображениях

Камера	APE		Dong et al.		Johnson et al.		Наш метод	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
iPhone	17.28	0.8631	19.90	0.8813	19.65	0.9003	19.16	0.9114
BlackBerry	18.91	0.8922	19.34	0.8998	19.31	0.9229	19.51	0.9259
Sony	19.45	0.9168	22.00	0.9291	21.07	0.9378	21.97	0.9400

PSNR: сопоставимо с Dong et al.

SSIM: превосходит все остальные методы

Типичные артефакты:

- отклонения цветов
- слишком высокая контрастность
- усиление шума



Исходное (iPhone)



Улучшенное



Исходное (iPhone)



Улучшенное



Исходное (BlackBerry)



Улучшенное



Исходное (BlackBerry)



Улучшенное



Исходное (Sony)



Улучшенное



Исходное (Sony)



Улучшенное



- Новый **подход**⁴ к улучшению качества фотографий: восстановление функции преобразования между фотографиями с мобильных устройств и фотографиями с зеркальных фотоаппаратов
- **Датасет**, состоящий большого числа фотографии, синхронно снятых зеркальным фотоаппаратом и камерами трех мобильных устройств в различных условиях
- Составная **функция ошибки** для эффективной оценки качества изображений, состоящая из ошибок цвета, текстуры и контента
- Вычислительные эксперименты, демонстрирующие преимущество улучшенных фотографий над исходными и паритет с фотографиями с зеркальных камер

⁴Статья на основе данной работы отправлена на ICCV 2017