



Лаборатория компьютерной
графики и мультимедиа
ВМК МГУ имени М.В. Ломоносова

Спецкурс «Введение в компьютерное зрение
и глубинное обучение»

Лекция №2 «Основы обработки изображений»

Антон Конушин

Заведующий лабораторией компьютерной графики и мультимедиа
ВМК МГУ

25 февраля 2019 года

Обработка изображений

- Семейство методов и задач, где входной и выходной информацией являются изображения
- Формально $Y = f(X)$, где $X, Y \in R^{n \times m \times k}$ - изображения с k -каналами, разрешением n на m пикселей

Зачем обрабатывать?

1. *Улучшение изображения для восприятия человеком*

- цель – чтобы стало «лучше» с субъективной точки зрения человека

2. *Улучшение изображения для восприятия компьютером*

- цель – упрощение последующего распознавания

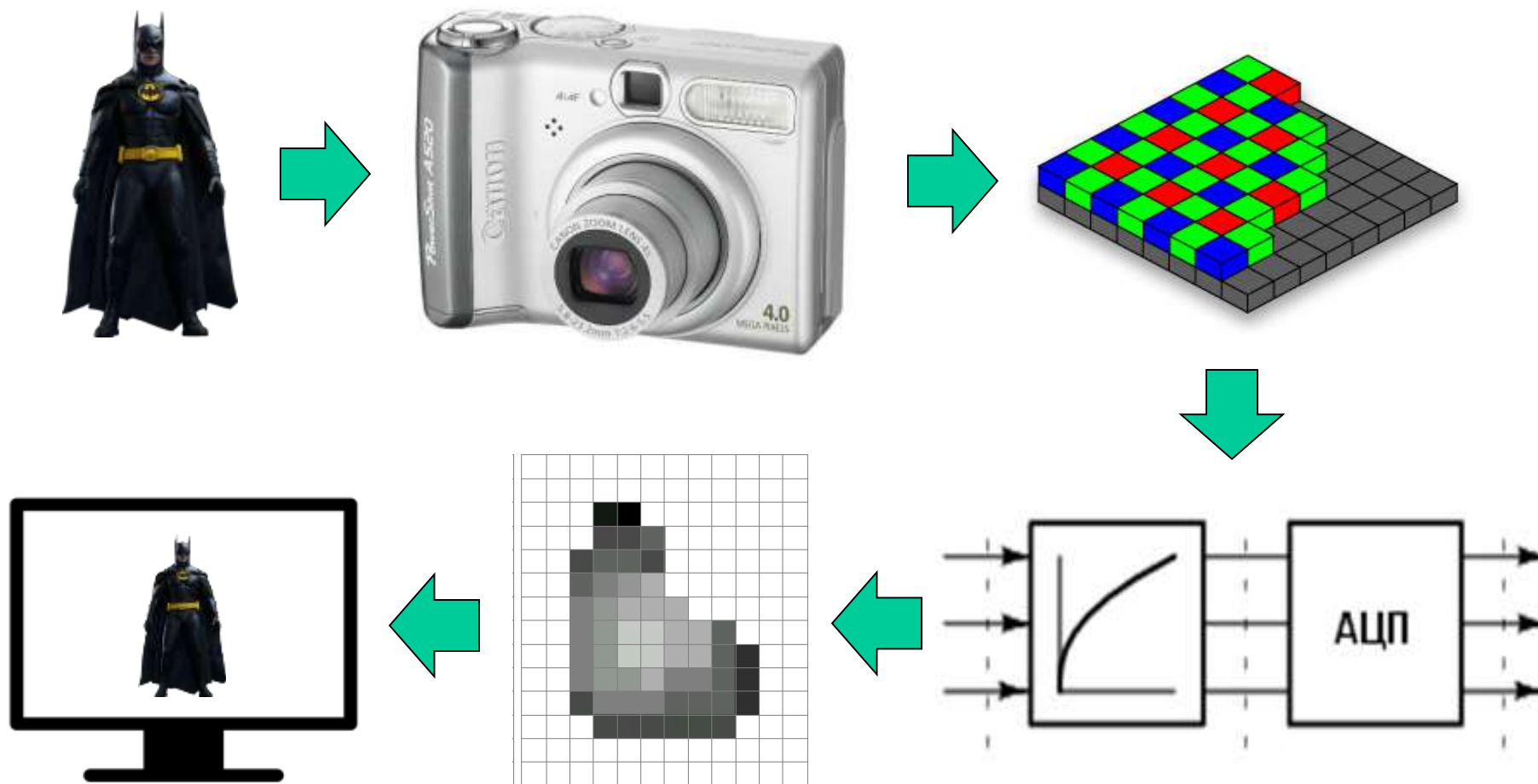
3. *Преобразование для технических нужд*

- например, изменение разрешения или пропорций для демонстрации на другом устройстве

4. *Развлечение (спецэффекты)*

- цель – получить эстетическое удовольствие от красивого эффекта

Тракт передачи изображений



Где кроются источники ошибок и дефектов?

Рассматриваемые дефекты



Темное или слабоконтрастное



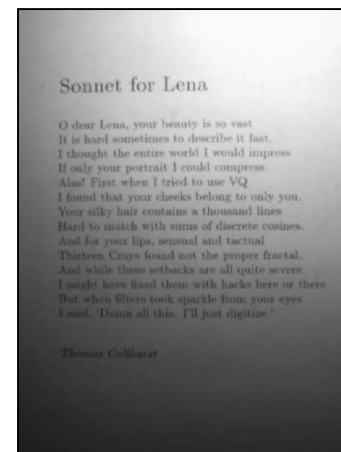
Неправильные цвета



Шумное



Нерезкое



Неравномерно
освещённое

Тональная коррекция

Постоянство цвета и освещенности



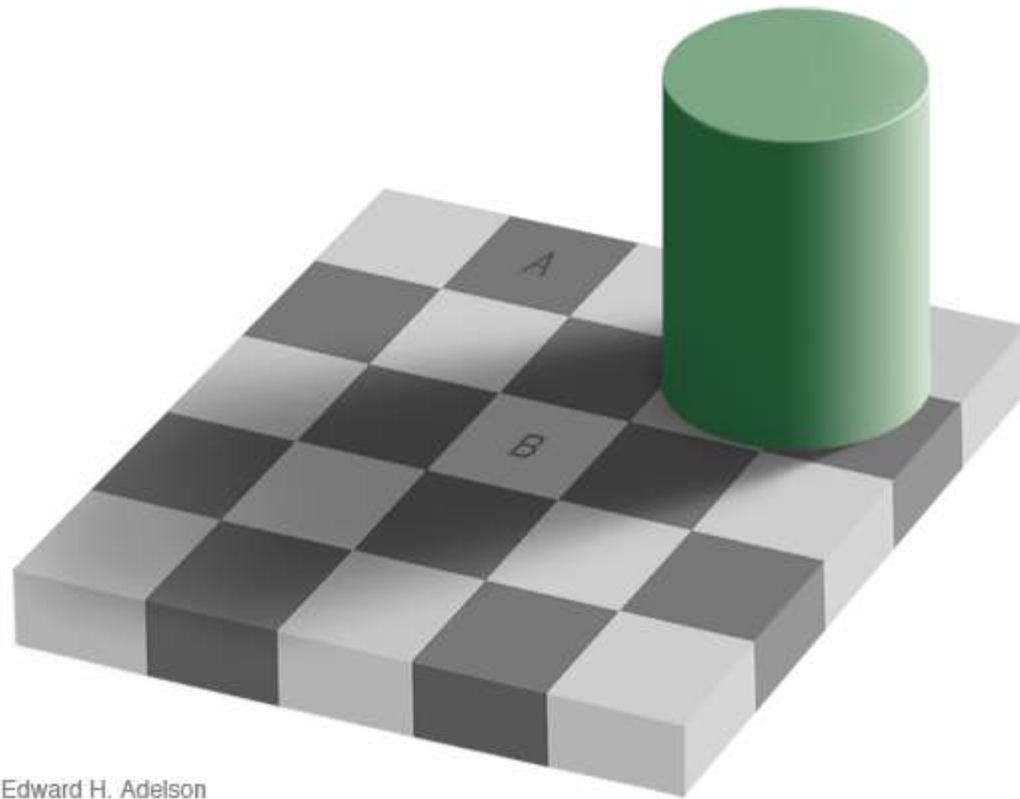
Способность зрительной системы человека оценивать собственные отражательные свойства поверхностей в не зависимости от условий освещенности

Освещённость:

- Полнолуние – 0.27 люкс
- Пасмурный день – 100-1000 люкс
- Яркий день (в тени) – 20000 люкс
- На солнце - 100000

J. S. Sargent, The Daughters of Edward D. Boit, 1882

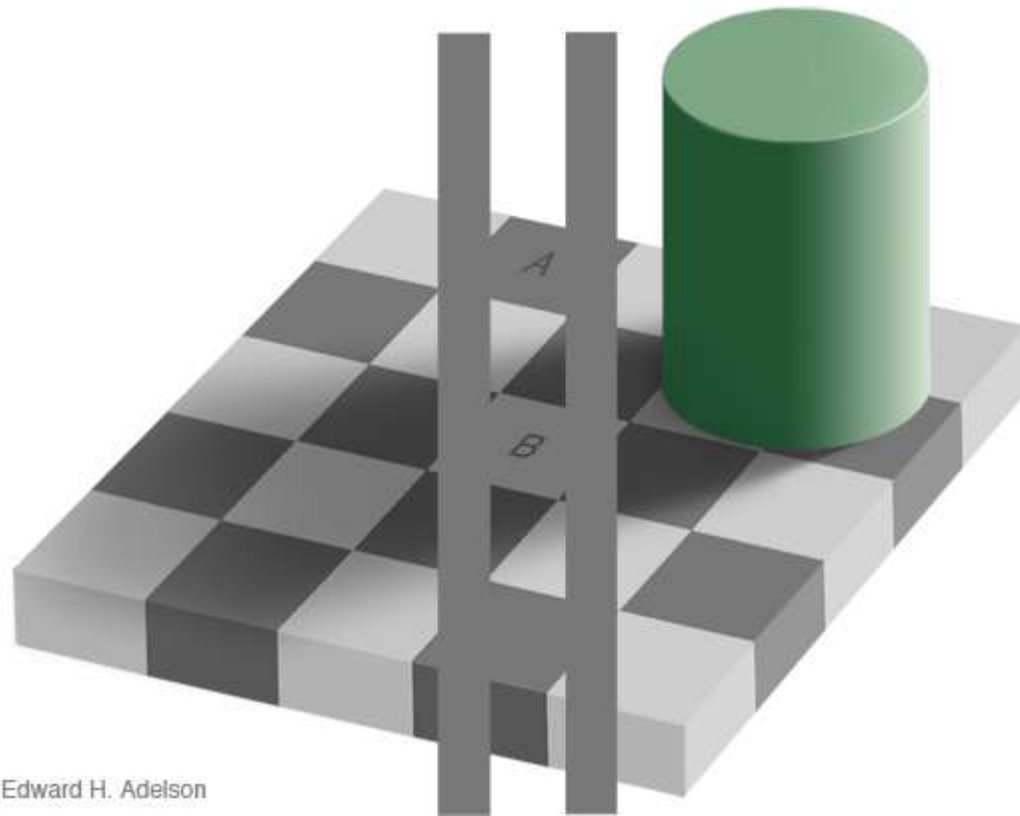
Постоянство яркости



Edward H. Adelson

http://web.mit.edu/persci/people/adelson/checkershadow_illusion.html

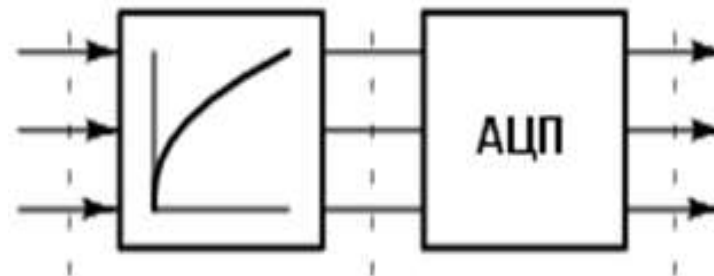
Постоянство яркости



http://web.mit.edu/persci/people/adelson/checkershadow_illusion.html

Причины плохой передачи яркости

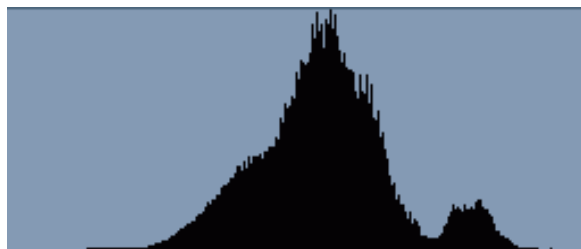
- Ограниченный диапазон чувствительности датчика
- “Плохая” функция передачи датчика



Нужно оценить качество передачи тонов в изображении

Гистограмма и её оценка

Гистограмма – это график распределения яркостей на изображении. На горизонтальной оси - шкала яркостей тонов от белого до черного, на вертикальной оси - число пикселей заданной яркости.



0

255



0

255

- Не полностью используется диапазон яркостей
- Концентрация яркостей вокруг определенных значений - неравномерное заполнение диапазона яркостей)

Точечные операторы

Оператор, который определяет значение выходного пиксела по значению только одного входного пиксела. Все пикселы обрабатываются независимо друг от друга.

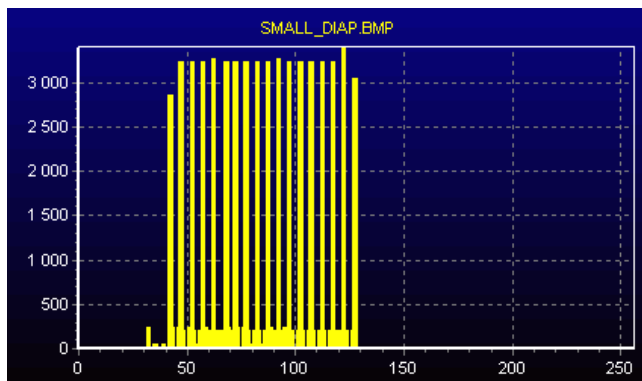
$$f^{-1}(y) = x$$

y – яркость пикселя на исходном изображении,
 x – яркость пикселя после коррекции.

Пишем f^{-1} , потому что «восстанавливаем» истинное значение яркости по неправильному

Линейная коррекция

Компенсация узкого диапазона яркостей –
линейное растяжение гистограммы
(Histogram equalization):



$$f^{-1}(y) = (y - y_{\min}) * \frac{(255 - 0)}{(y_{\max} - y_{\min})}$$

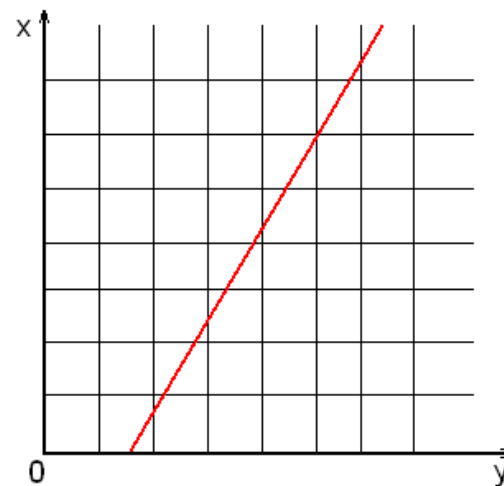
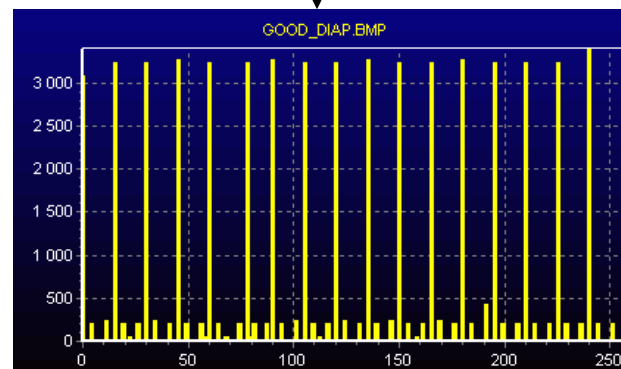
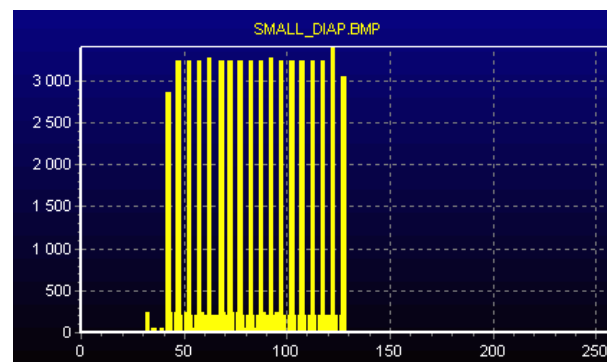


График функции $f^{-1}(y)$

Линейная коррекция

Компенсация узкого диапазона яркостей – линейное растяжение гистограммы:



Робастная линейная коррекция

Что будет при применении линейной коррекции к такой картинке?



Робастная (устойчивая) версия метода:

- Вычислим такую линейную коррекцию, чтобы 5% самых темных пикселов стали черными и 5% самых светлых стали белыми

Линейная коррекция

Линейное растяжение – «как AutoContrast в Photoshop»



Линейная коррекция



Линейная коррекция помогает не всегда!

К слову, в чём может быть причина дефекта такого изображения?

Нелинейная коррекция

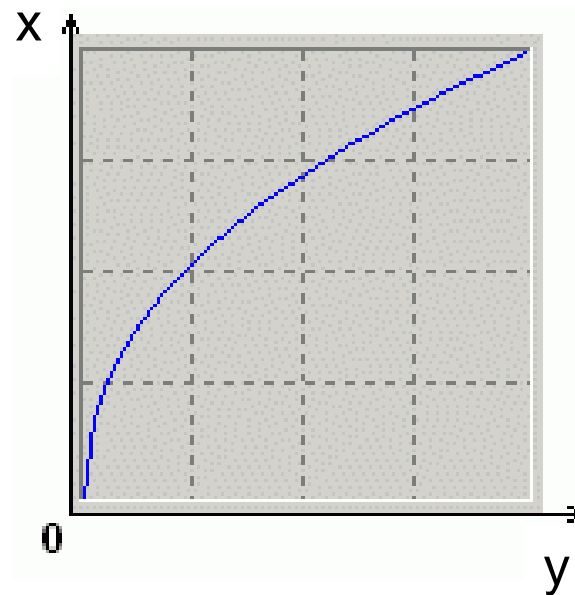
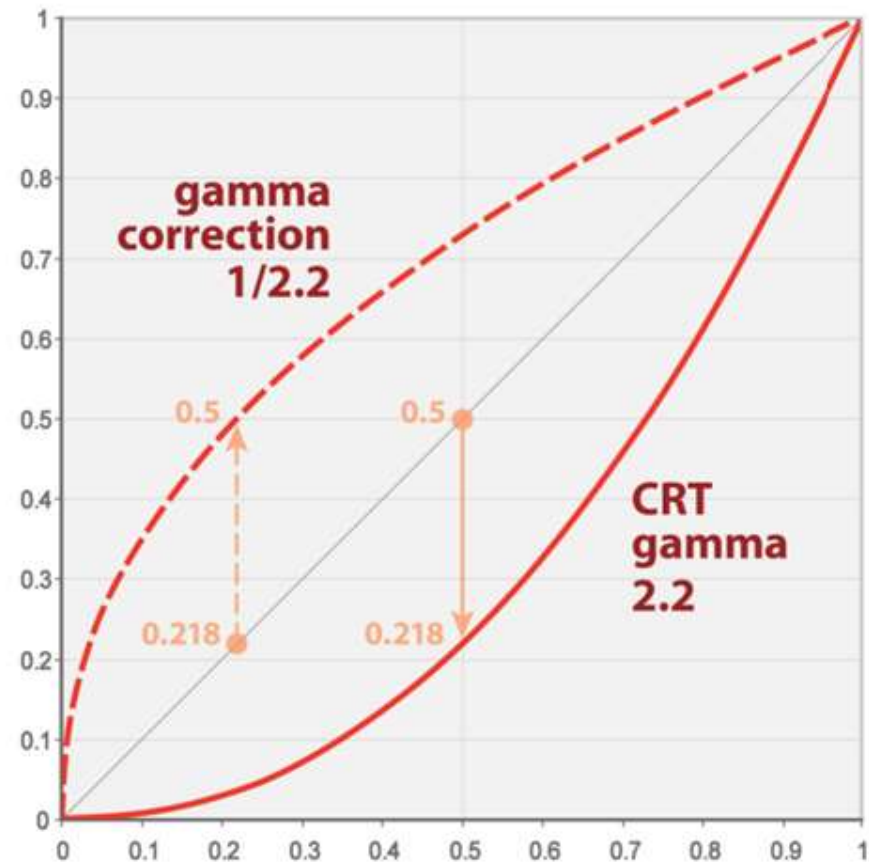


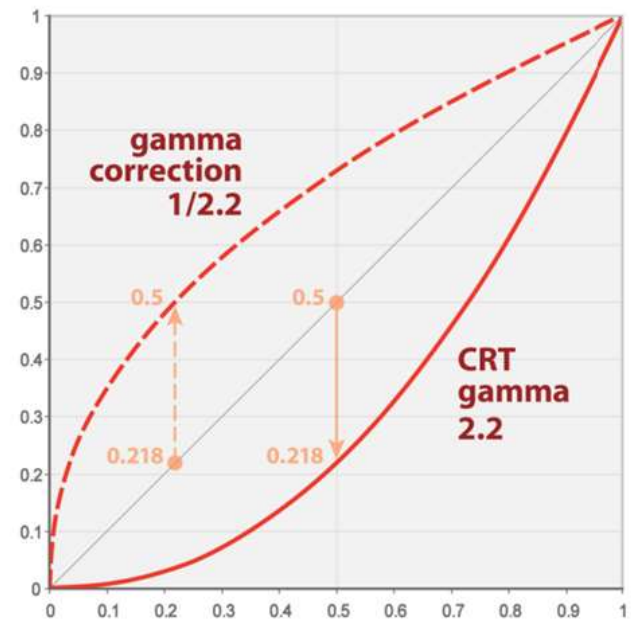
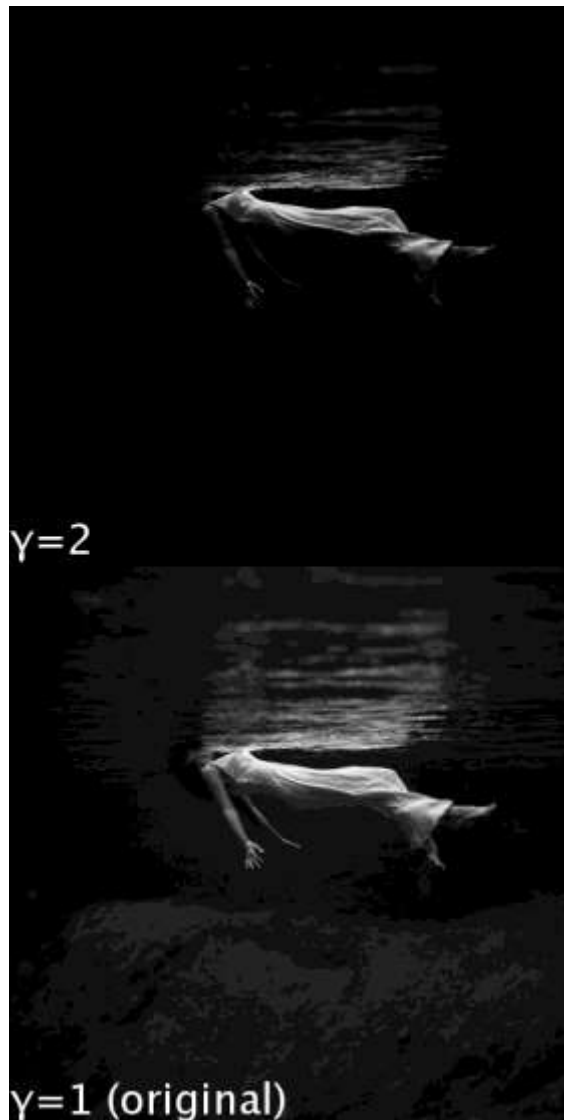
График функции $f^{-1}(y)$

Гамма-коррекция

- Преобразование вида $y = c \cdot x^\gamma$
- Изначальная цель – коррекция для правильного отображения на мониторе
 - на ЭЛТ-мониторах была нелинейная зависимость между напряжением и яркостью пиксела
- Сейчас важнее как раз лучшая контрастность изображения при ограничении диапазона (те же 8бит)



Пример



Произвольная нелинейная коррекция

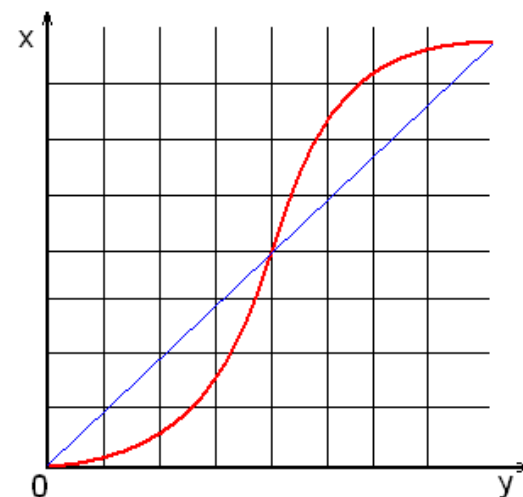
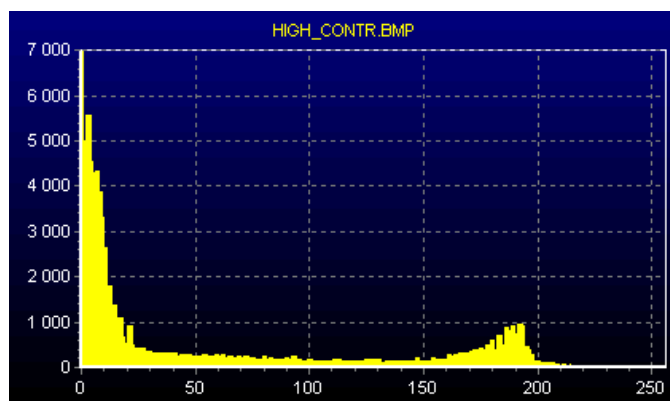
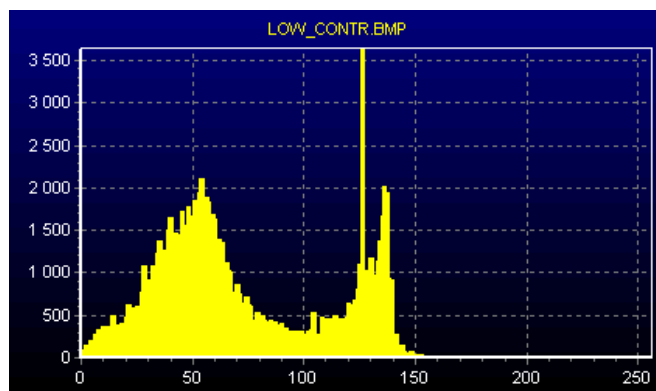
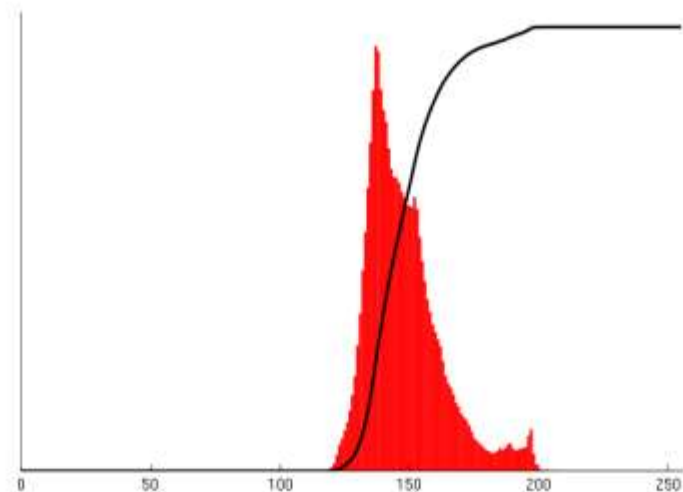


График функции $f^{-1}(y)$

Выравнивание гистограммы



https://en.wikipedia.org/wiki/Histogram_equalization

Цветокоррекция

Цветовой баланс («баланс белого»)

Когда мы смотрим на фотографию или монитор, глаза адаптируются к освещению в комнате, а не к освещению сцены на фотографии.

Если «баланс белого» неточен, цвета фотографии кажутся неестественными.

Неправильный баланс



Правильный баланс



Как определить,
что цвета
неправильные?

Как
скорректировать
изображение?

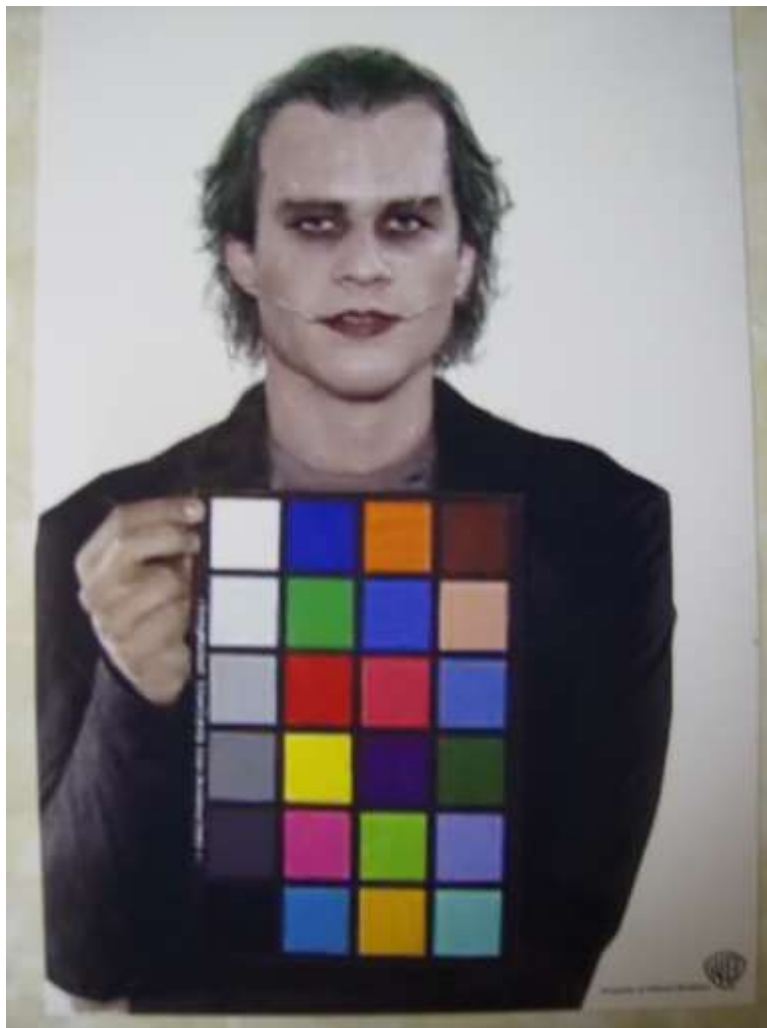
Коррекция по шаблону

- Разумный подход:
 - Сфотографировать объект с известным цветом (шаблон)
 - Вычислить цветовое преобразование, чтобы цвет объекта на фотографии совпал с нужным
- Простейшая реализация:
 - Возьмём однотонные карточки (белые)
 - Будем домножать каждый канал отдельно, чтобы цвет карточек стал белым
 - Вычисление коэффициентов - Если цвет объект записывается как r_w, g_w, b_w , тогда веса $1/r_w, 1/g_w, 1/b_w$



Насколько такое преобразование корректно, какие могут быть недостатки?

Профессиональная цветокоррекция



Source: The dark knight



Source: <http://x-rite.com>

Используем цветной шаблон с многими цветами

Какое преобразование в камере?

Сложные модели



<http://vision.middlebury.edu/color/>

- Авторы собрали большую коллекцию разных изображений для оценки различных моделей преобразования в камере
- Полиномиальная модель (24 параметра)

$$y_i = g_i([M_D k]_i)$$

A. Chakrabarti, D. Scharstein, and T. Zickler. [An empirical camera model for Internet color vision](#). BMVC 2009

Оценка параметров цветокоррекции

Если нет цветowych шаблонов, тогда нам нужно угадать (или оценить) коэффициенты усиления

Модель «Серого мира» (Grayworld)

- Средний уровень («серый») по каждому каналу должен быть одинаков для всех каналов
- Если цветовой баланс нарушен, тогда «серый» в этом канале больше «серого» других каналов
- Вычислим коэффициенты усиления так, чтобы среднее в каждом канале стало одинаковым:

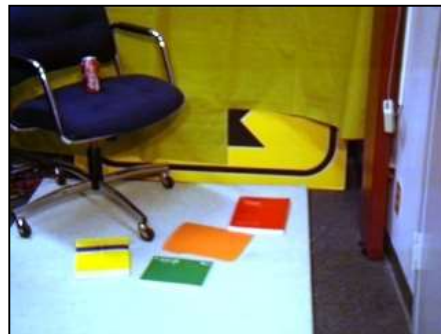
$$\bar{R} = \frac{1}{N} \sum R(x, y); \quad \bar{G} = \frac{1}{N} \sum G(x, y); \quad \bar{B} = \frac{1}{N} \sum B(x, y); \quad Avg = \frac{\bar{R} + \bar{G} + \bar{B}}{3};$$

$$R' = R \cdot \frac{Avg}{\bar{R}}; \quad G' = G \cdot \frac{Avg}{\bar{G}}; \quad B' = B \cdot \frac{Avg}{\bar{B}}$$

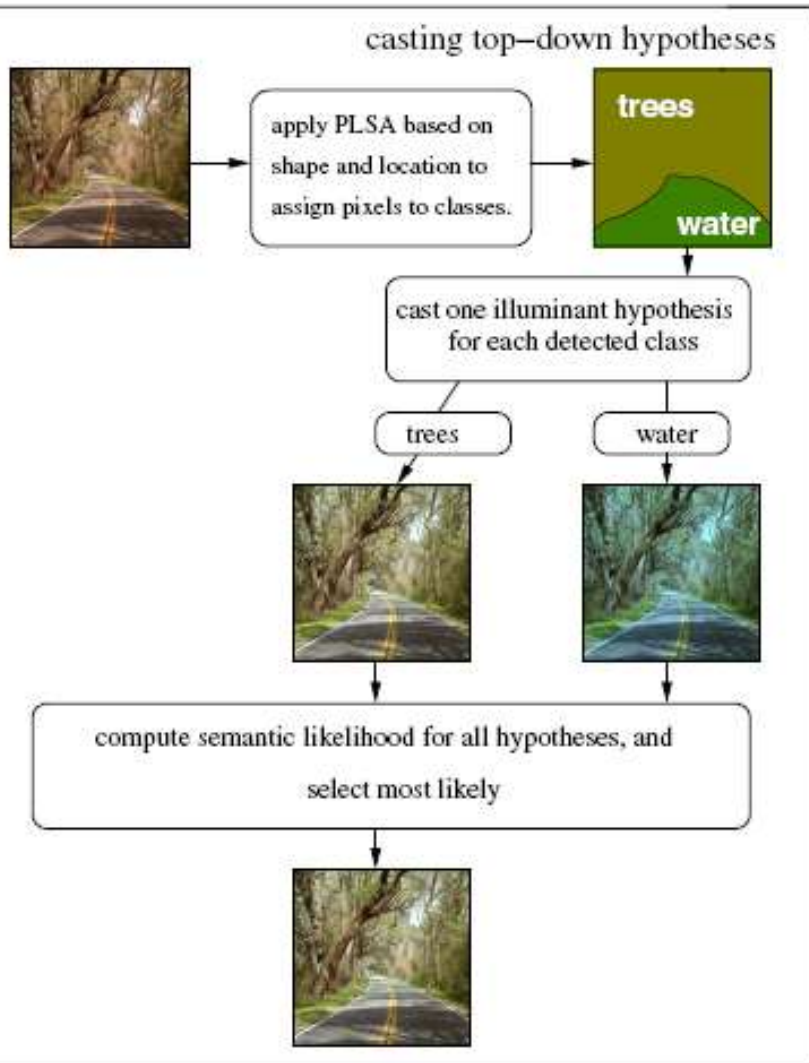
«Серый мир» - примеры



«Серый мир» - примеры



Распознавание баланса белого



Методы цветовой коррекции до сих пор развиваются.

Пример: Для каждого класса объектов, присутствующих в сцене, вычисляем преобразование таким образом, чтобы диапазон цветов объекта совпадал со средним диапазоном объектов этого класса на «типичных» изображениях

Шумоподавление

Шумоподавление



Шум фотоаппарата

- Изображения обычно содержат «шум», т.е. значение пикселя отличается от истинного
- Нужно получить «чистое» изображение без шума («подавить шум»)

Виды шума



Original



Gaussian noise



Salt and pepper noise



Impulse noise

- **Гауссов:**
колебания яркости,
распределенные по
нормальному закону
 - Аддитивный шум
 - $\text{Image}(i,j) = \text{true}(i,j) + \text{noise}(i,j)$
 - $\text{Noise}(i,j) \sim N(\mu, \sigma)$
- **Потеря информации**
(*data drop-out noise*)
 - **Соль и перец:**
случайные черные и
белые пиксели
 - **Импульсный:**
случайные белые
пиксели

Усреднение нескольких кадров

«Временная фильтрация»



Зашумленные изображения

Усреднение по 10
изображениям

$$I(i, j) = g_r(i, j) + Err(i, j);$$

$$\bar{I}(i, j) = \frac{1}{N} \sum_{k=1}^N I_k(i, j);$$

$$E(\bar{I}(i, j)) = g_r(i, j);$$

Шумоподавление в одном кадре



Как быть, если
изображение
только одно?

Пространственная фильтрация

- Вычислим новое значение y_{ij} для пиксела x_{ij} как функцию от его локальной окрестности:

$$y_{ij} = f([x_{kl}]), x_{kl} \in \text{neighbour}(x_{ij})$$

- В простейшем случае мы возьмём просто взвешенное среднее по всем пикселям из окрестности
- Веса обозначаются как *ядро фильтра*
- Веса для усреднения задаются так:

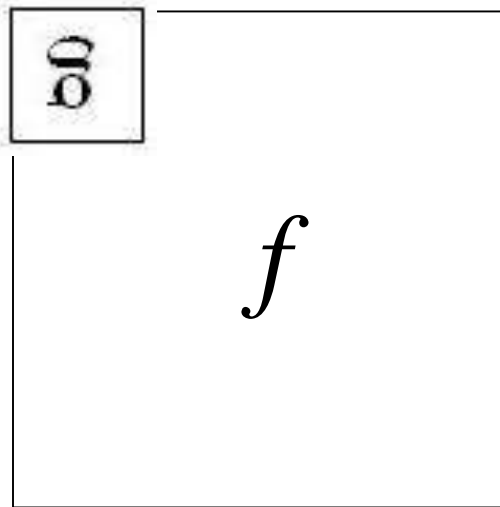
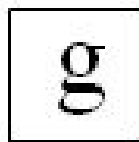
$\frac{1}{9}$	1	1	1
	1	1	1
	1	1	1

“box filter”

Свертка

- Пусть f – изображение, g – ядро. Свертка изображения f с помощью g обозначается как $f * g$ и называется:

$$(f * g)[m, n] = \sum_{k, l} f[m - k, n - l] g[k, l]$$



Свёртка – ядро фильтра «перевернуто»!

Примеры фильтров



Original

0	0	0
0	1	0
0	0	0

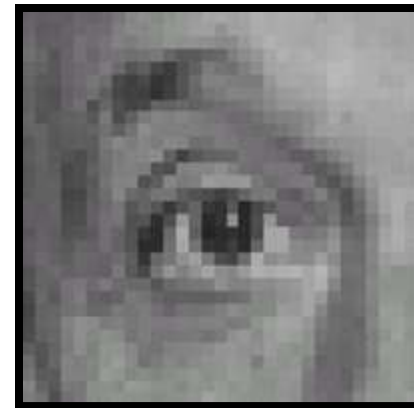
?

Примеры фильтров



Original

0	0	0
0	1	0
0	0	0



Filtered
(no change)

Примеры фильтров



Original

0	0	0
0	0	1
0	0	0

?

Примеры фильтров



Original

0	0	0
0	0	1
0	0	0



Shifted left
By 1 pixel

Примеры фильтров



Original

 $\frac{1}{9}$

1	1	1
1	1	1
1	1	1

?

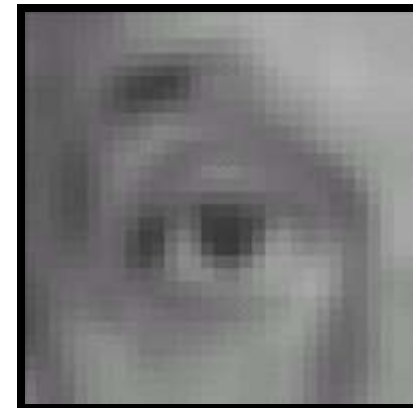
Примеры фильтров



Исходное

 $\frac{1}{9}$

1	1	1
1	1	1
1	1	1



Результат

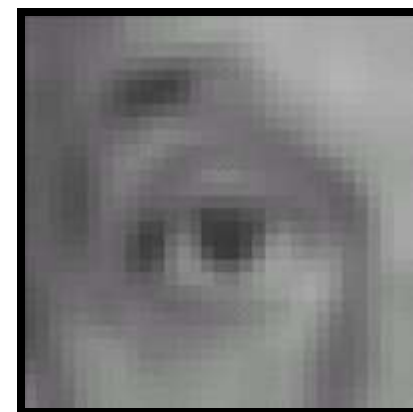
Примеры фильтров



Исходное

$$\frac{1}{16}$$

1	2	1
2	4	2
1	2	1



Результат

Любой фильтр со всеми положительными весами и суммой весов = 1 будет сглаживать изображение

Примеры фильтров



$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$



Примеры фильтров

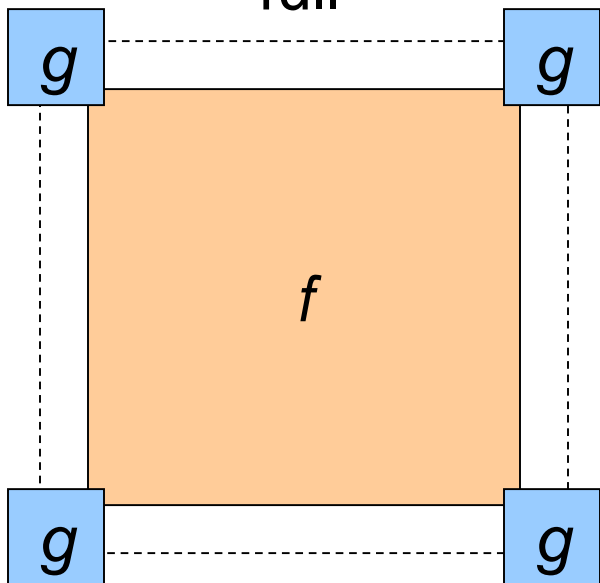


$$\frac{1}{10} \cdot \begin{vmatrix} -1 & -2 & -1 \\ -2 & 22 & -2 \\ -1 & -2 & -1 \end{vmatrix}$$

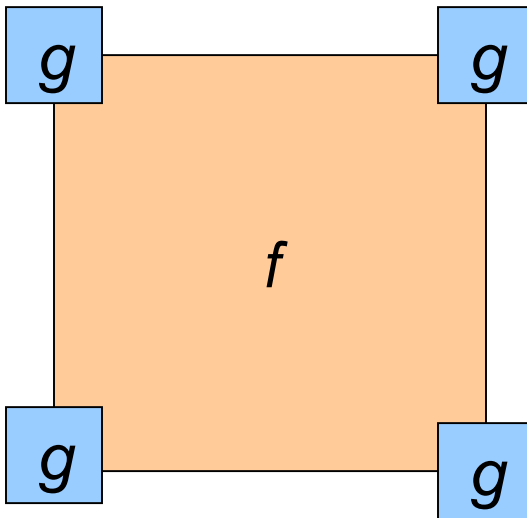


Детали реализации

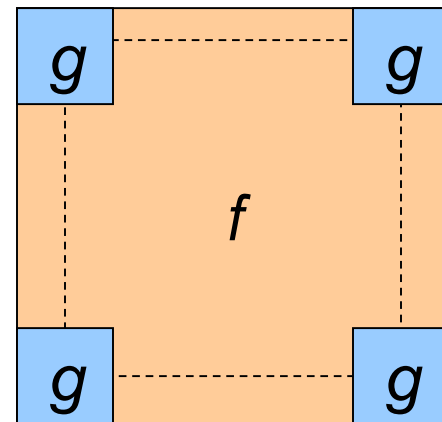
full



same



valid



Линейные фильтры и свёртка

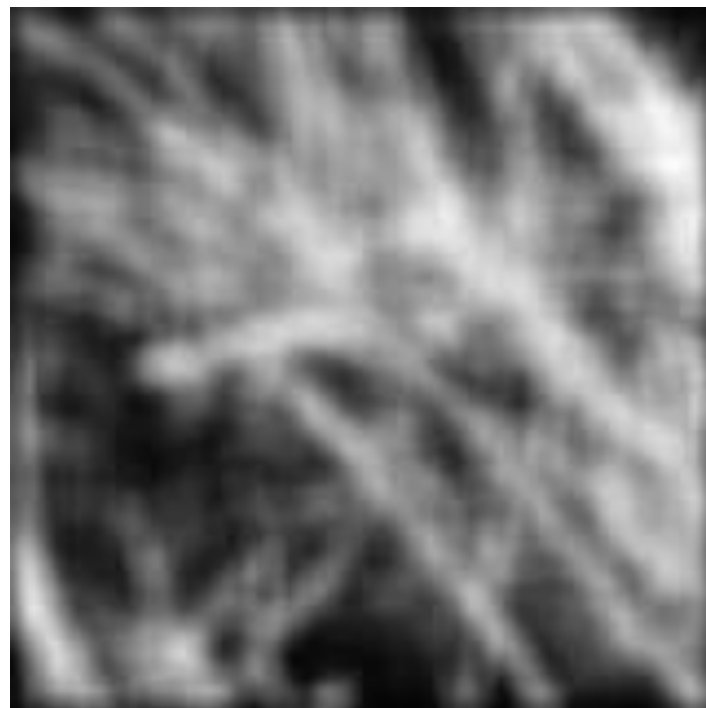
- Пространственный фильтр называется **линейным**, если выполняются свойства **линейности**:
 - $\text{filter}(f_1 + f_2) = \text{filter}(f_1) + \text{filter}(f_2)$
 - $\text{filter}(af_1) = a \text{filter}(f_1)$
- Фильтр называется инвариантным к сдвигу, если выполняется следующее условие:
$$\text{filter}(\text{shift}(f)) = \text{shift}(\text{filter}(f))$$
- Теорема: любой линейный оператор, инвариантный к сдвигу, может быть записан в виде свертки
- Чтобы доказать нелинейность фильтра, можно воспользоваться основными свойствами, и показать их не выполнение на примере

Полезные свойства

- Ассоциативность: $a * (b * c) = (a * b) * c$
 - Последовательное применение фильтров: $((a * b_1) * b_2) * b_3$
 - Эквивалентно применению такого фильтра: $a * (b_1 * b_2 * b_3)$
- Дистрибутивность по сложению:
$$a * (b + c) = (a * b) + (a * c)$$

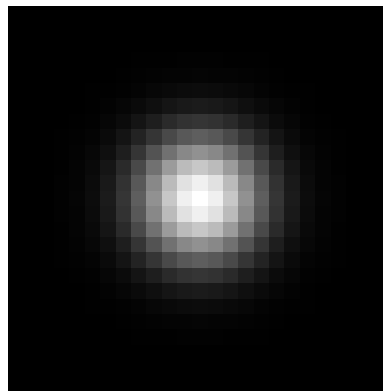
Сглаживание с box-фильтром

При сглаживании с box-фильтром на изображении могут образовываться паразитные линии



Сглаживание

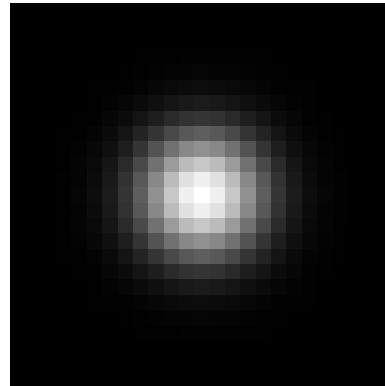
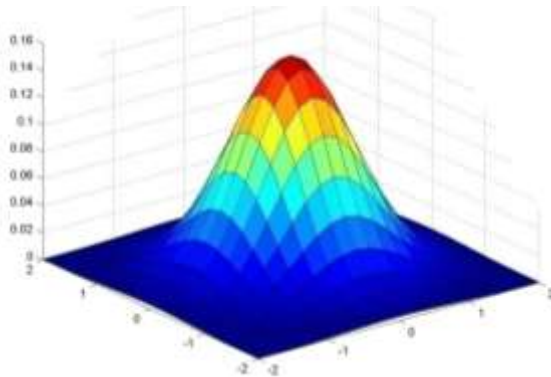
- Точка света, наблюдаемая с расфокусированного объектива, выглядит как кружок света, а усреднение (box) дает квадратик
- Другой способ: взвешиваем вклад пикселей по окрестности с учетом близости к центру



“размытый кружок”

Ядро фильтра гаусса

$$G_{\sigma} = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

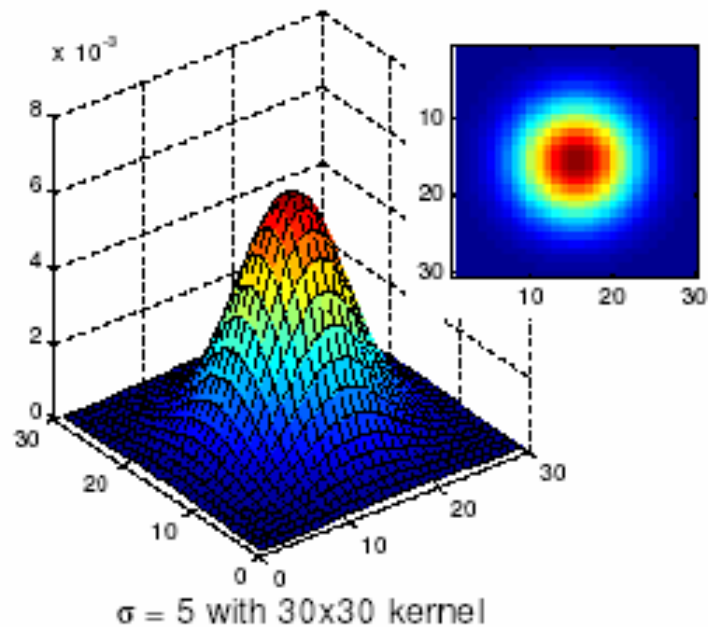
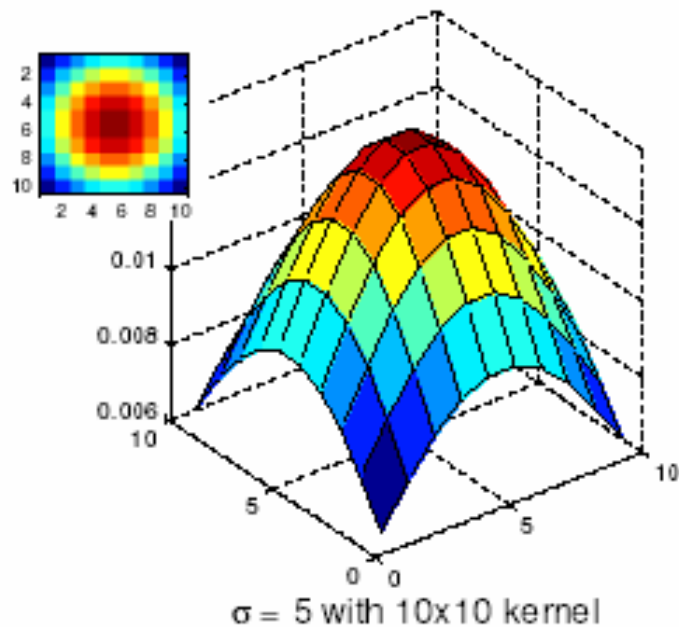


0.003	0.013	0.022	0.013	0.003
0.013	0.059	0.097	0.059	0.013
0.022	0.097	0.159	0.097	0.022
0.013	0.059	0.097	0.059	0.013
0.003	0.013	0.022	0.013	0.003

5 x 5, $\sigma = 1$

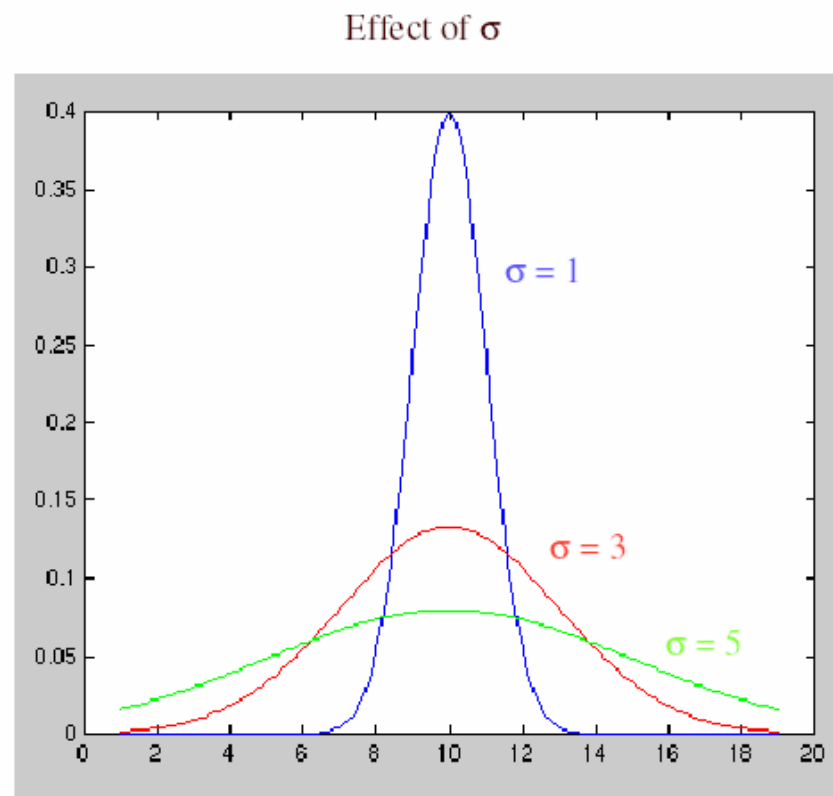
Выбор размера ядра

Размер ядра дискретного фильтра ограничен

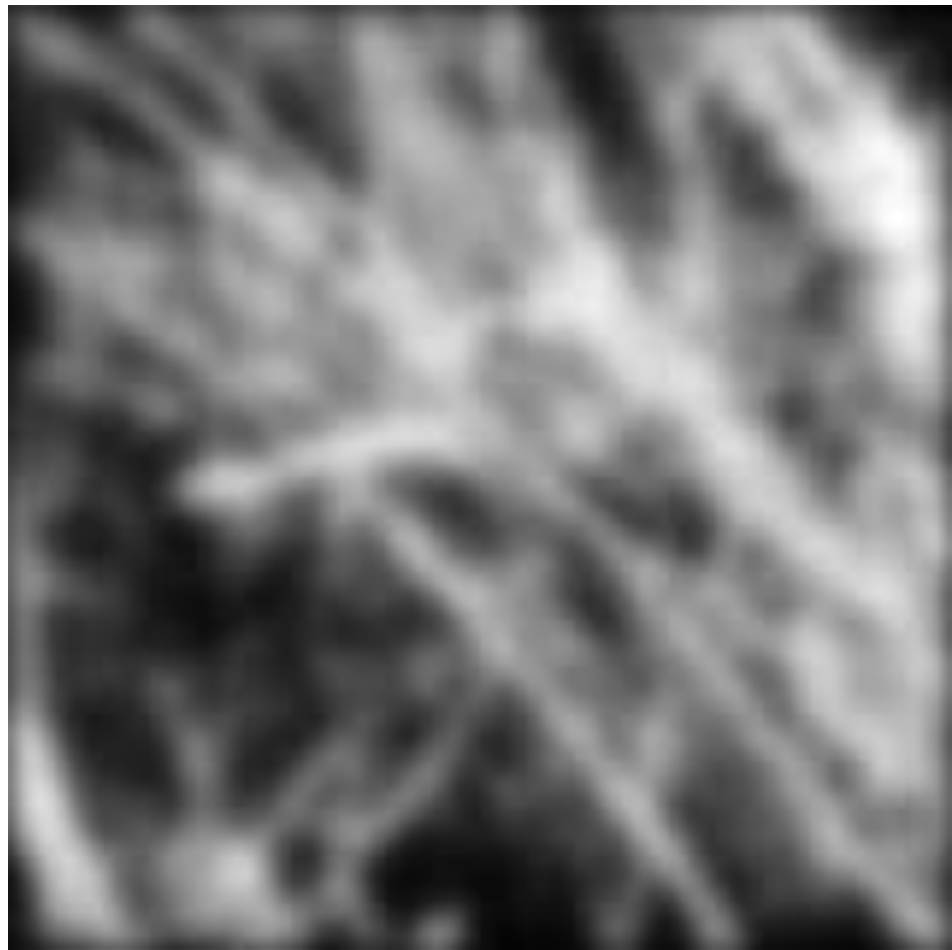
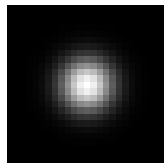


Выбор размера ядра

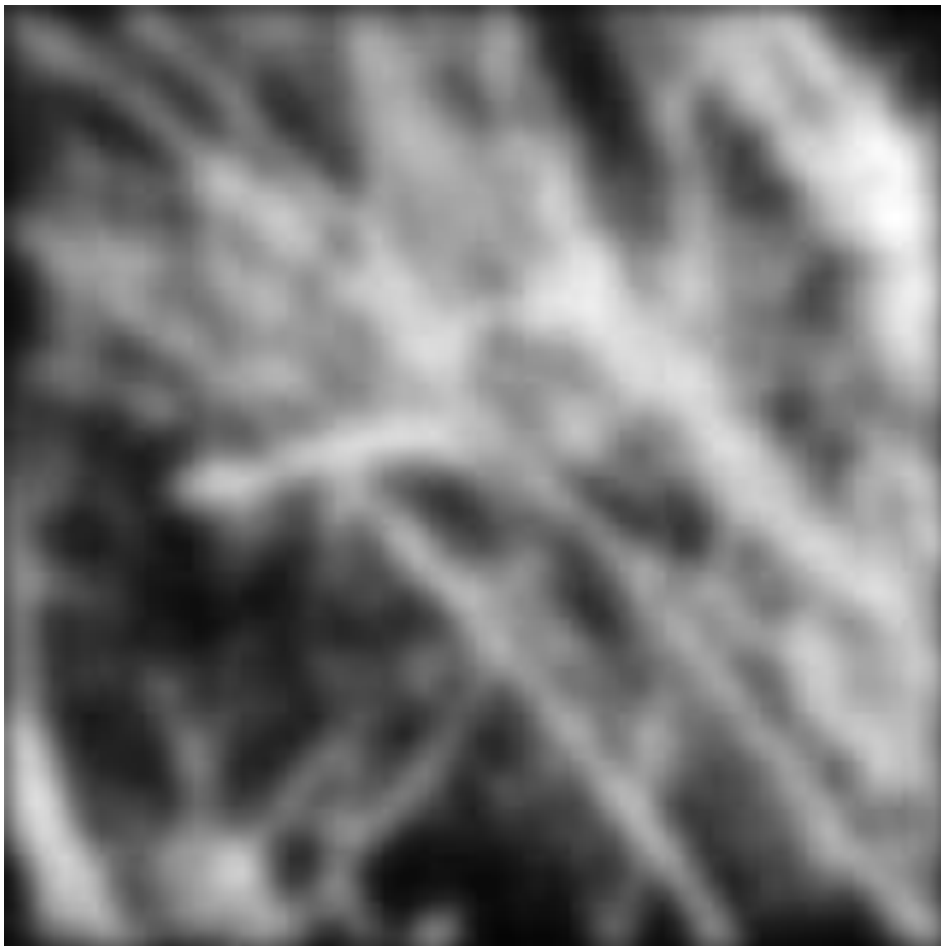
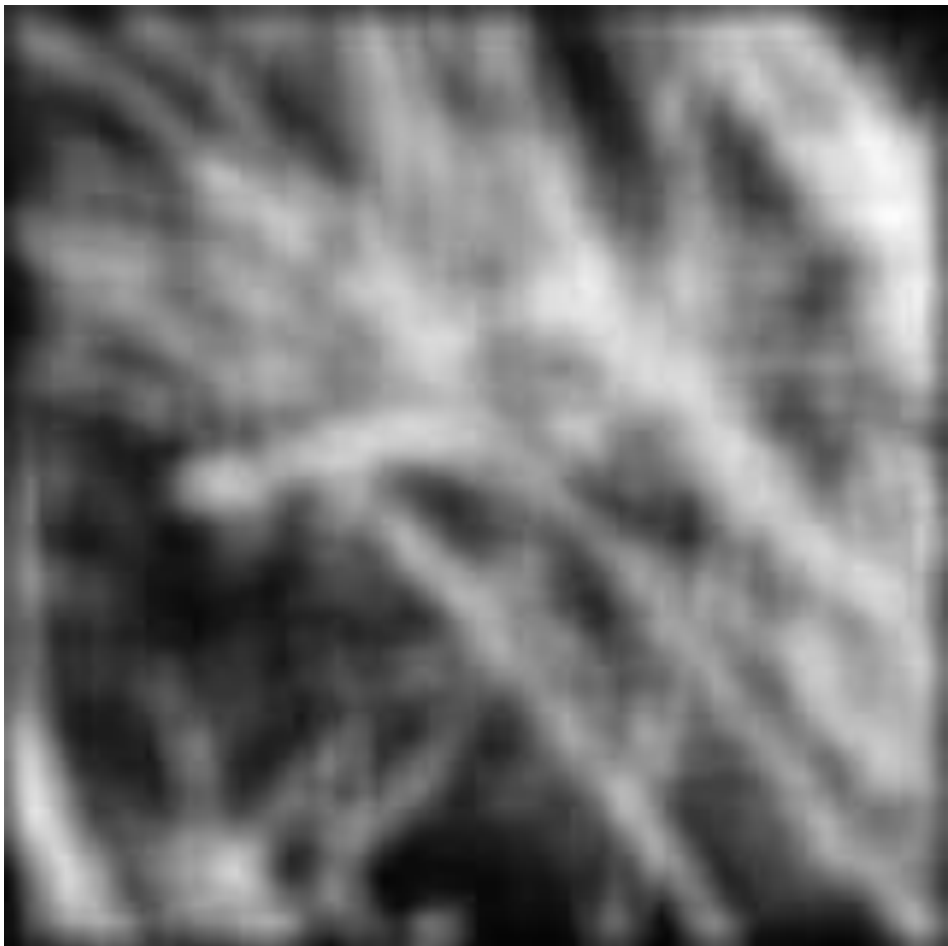
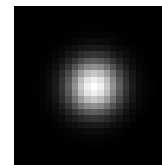
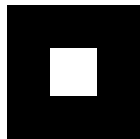
Эмпирика: полуразмер фильтра равен 3σ



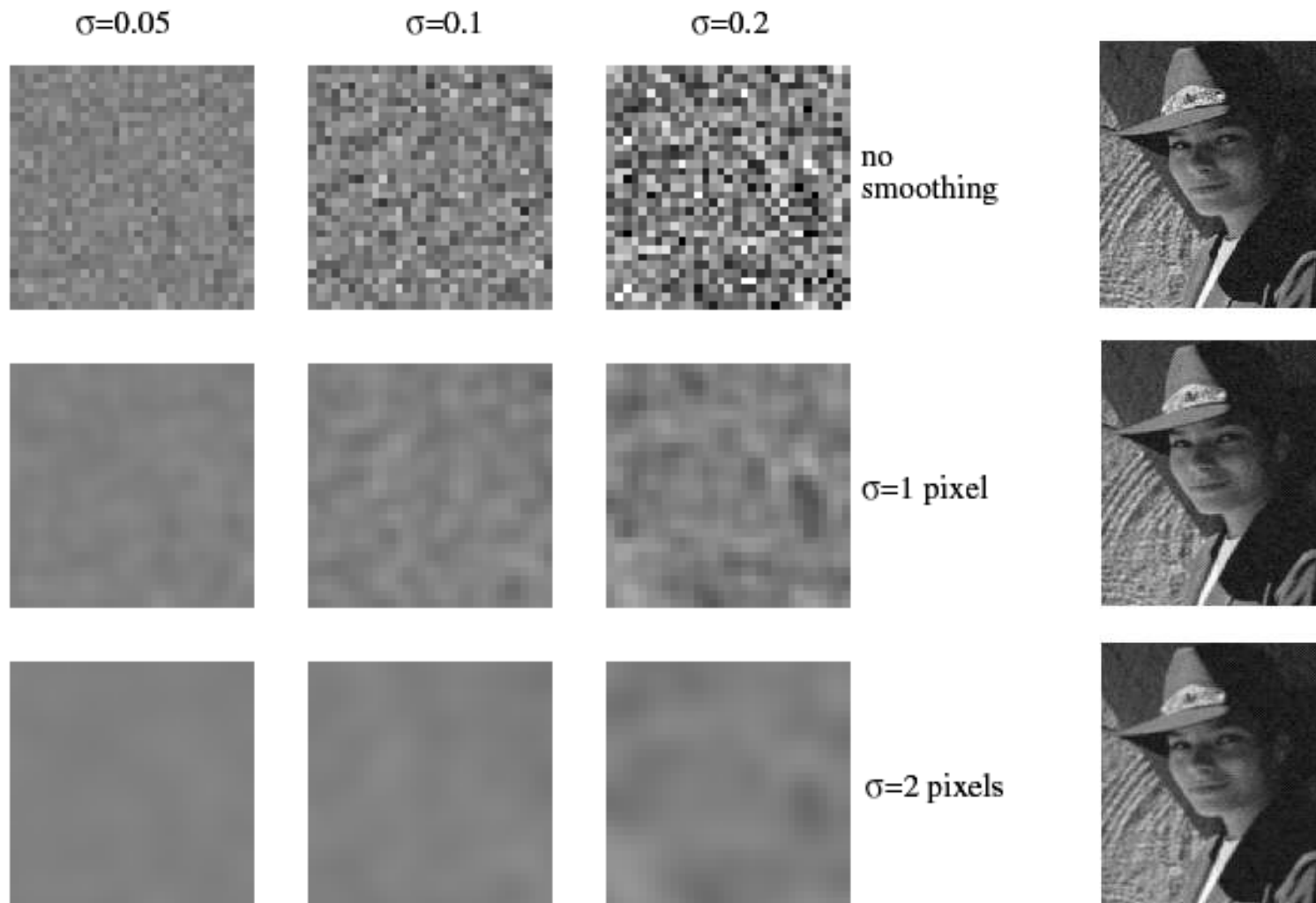
Сглаживание фильтром гаусса



Сравнение бокс и гаусс-фильтра



Подавление гауссова шума



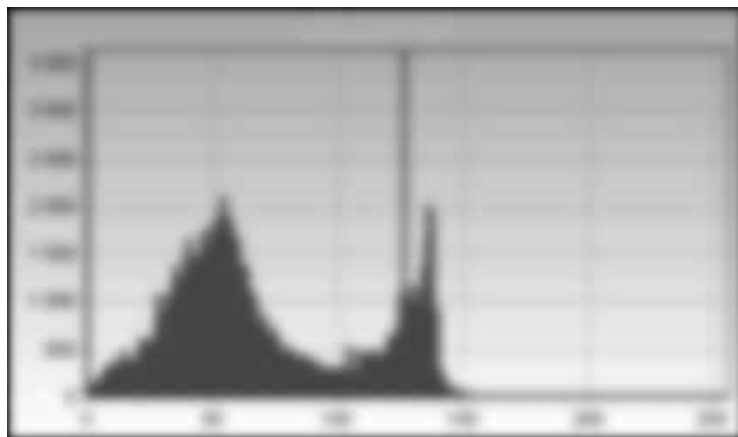
Сглаживание фильтрами большого радиуса подавляет шум, но размывает изображение

Свойства фильтра Гаусса

- Свертка с сами собой дает тоже фильтр гаусса
 - Сглаживание несколько раз фильтром с маленьким ядром дает результат, аналогичный свертке с большим ядром
 - Свертка 2 раза с фильтром радиуса σ дает тот же результат, что с фильтром радиуса $\sigma\sqrt{2}$
- Сепарабельное ядро
 - Раскладывается в произведение двух одномерных фильтров гаусса

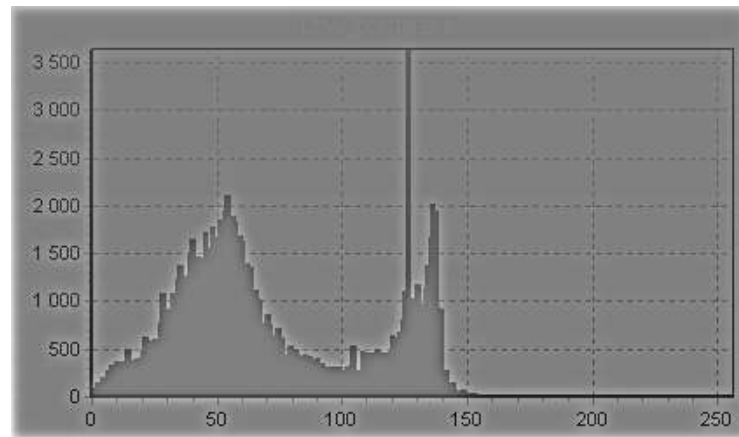
$$\begin{aligned} G_{\sigma}(x, y) &= \frac{1}{2\pi\sigma^2} \exp^{-\frac{x^2 + y^2}{2\sigma^2}} \\ &= \left(\frac{1}{\sqrt{2\pi}\sigma} \exp^{-\frac{x^2}{2\sigma^2}} \right) \left(\frac{1}{\sqrt{2\pi}\sigma} \exp^{-\frac{y^2}{2\sigma^2}} \right) \end{aligned}$$

Маленькая экскурсия к Фурье

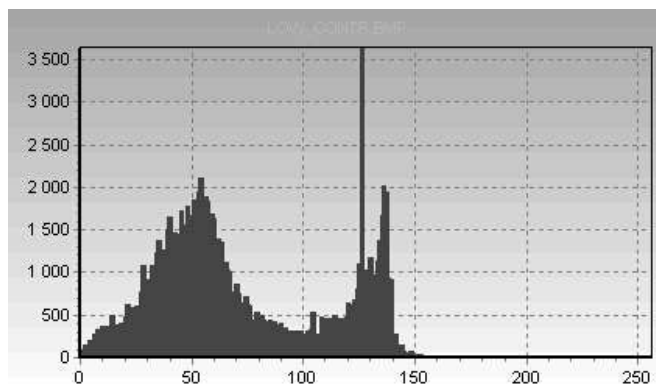


Низкие частоты

+

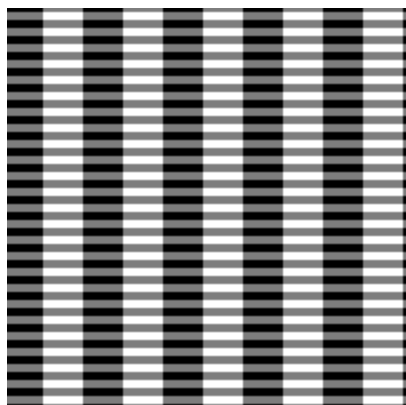


Высокие частоты

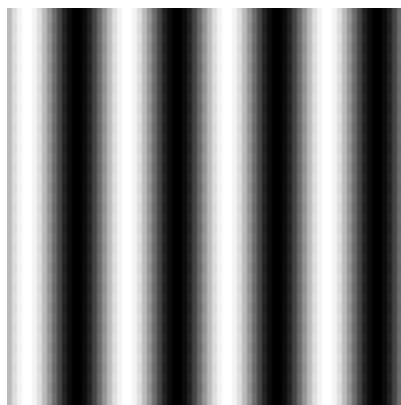


Фильтр Гаусса

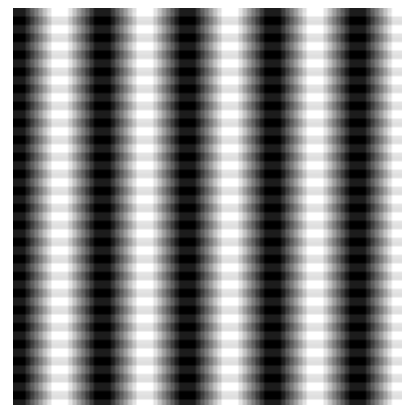
Результат свертки фильтром гаусса и усреднения



Исходное изображение



Фильтр Гаусса с
Sigma = 4



Усреднение по 49
пикселям (7x7)

Важное свойство фильтра Гаусса – он по сути является фильтром низких частот.

Пример применения фильтра Гаусса

Sonnet for Lena

O dear Lena, your beauty is so vast
It is hard sometimes to describe it fast.
I thought the entire world I would impress
If only your portrait I could compress.
Alas! First when I tried to use VQ
I found that your cheeks belong to only you.
Your silky hair contains a thousand lines
Hard to match with sums of discrete cosines.
And for your lips, sensual and tactual
Thirteen Crays found not the proper fractal.
And while these setbacks are all quite severe
I might have fixed them with hacks here or there
But when filters took sparkle from your eyes
I said: 'Damn all this. I'll just digitize.'

Thomas Colthurst

Что можно сказать
про это изображение?

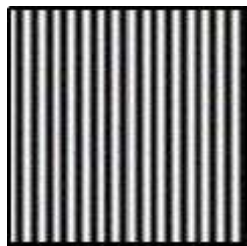
Компенсация разности освещения

Идея:

Формирование изображения:

$$I(i, j) = l(i, j) \cdot r(i, j)$$

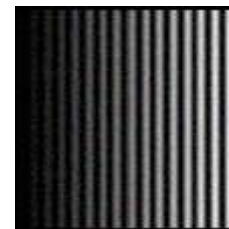
Плавные изменения яркости относятся к освещению, резкие - к объектам.



объект $r(i, j)$



освещение $l(i, j)$



Изображение
освещенного
объекта $I(i, j)$

Алгоритм Single scale retinex (SSR)

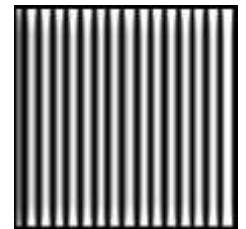
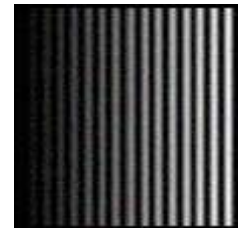
- Получить приближенное изображение освещения путем низочастотной фильтрации

$$\hat{l}(i, j) = G * I(i, j)$$

- Восстановить изображение по формуле

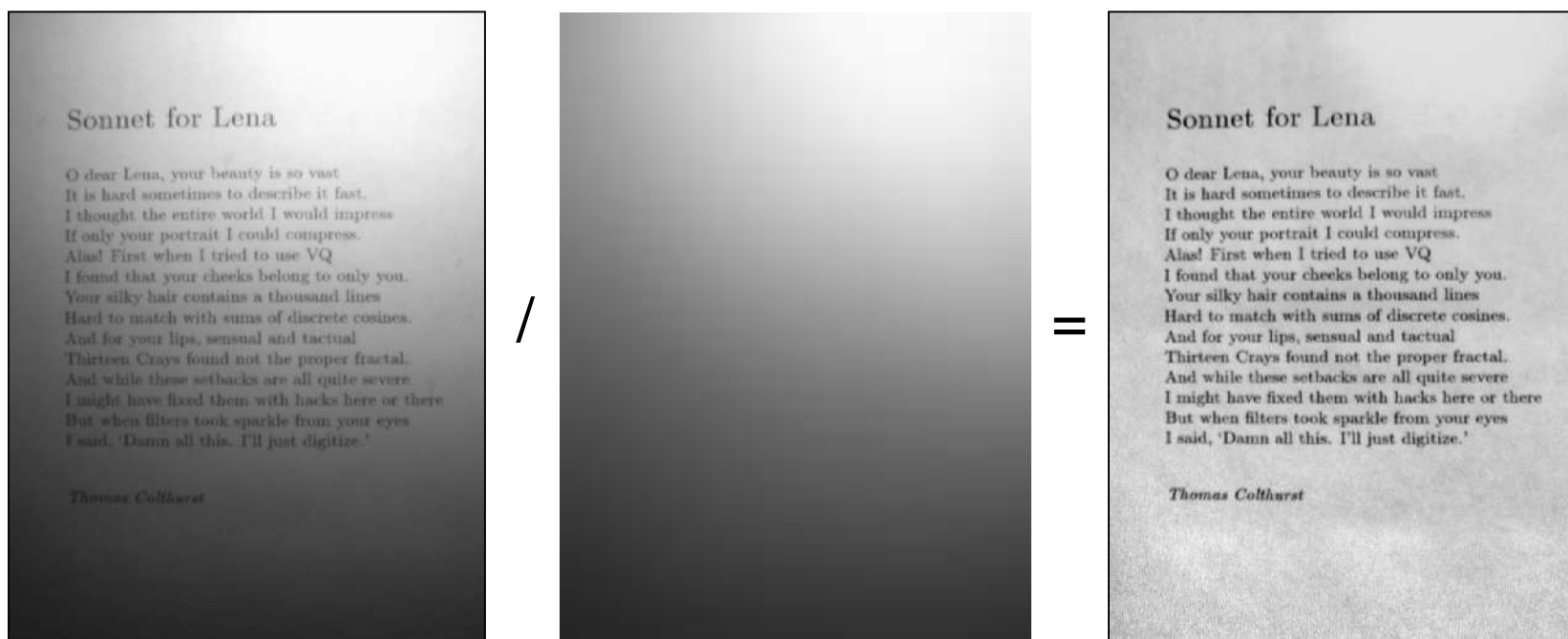
$$\hat{r}(i, j) = \frac{I(i, j)}{\hat{l}(i, j)}$$

- После преобразования потребуется применить тональную коррекцию и определить значения, которые будут соответствовать черному и белому



Компенсация разности освещения

Пример

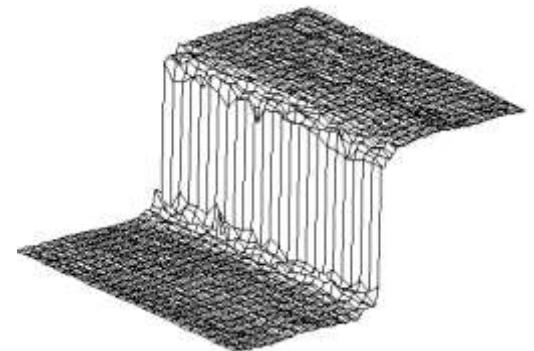
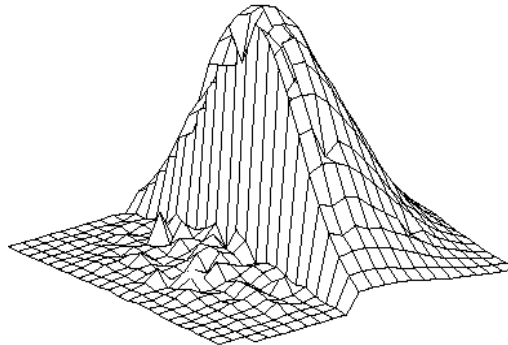
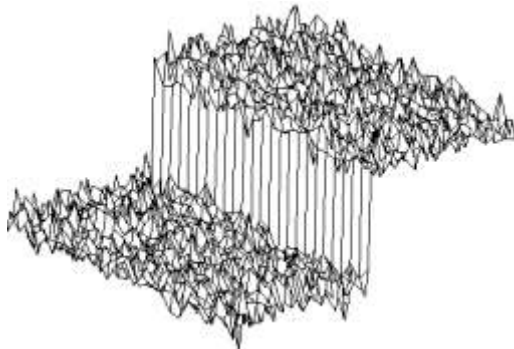


Gauss 14.7 пикселей

Билатеральный фильтр

- В операции свёртки каждый пиксель обрабатывается одинаково
- Но мы можем сделать параметры фильтра зависящими от изображения
- Билатеральный фильтр:

$$y_{ij} = \sum x_{i-k, j-l} \underbrace{w(x_{i-k, j-l} - x_{i,j})}_{\text{Вес, пропорциональный близости пикселей по цвет}} \underbrace{d(x_{i-k, j-l}, x_{i,j})}_{\text{Параметры фильтра Гаусса}}$$



Вернёмся к видам шума



Original



Gaussian noise



Salt and pepper noise



Impulse noise

- **Гауссов:**
колебания яркости,
распределенные по
нормальному закону
 - Аддитивный шум
 - $\text{Image}(i,j) = \text{true}(i,j) + \text{noise}(i,j)$
 - $\text{Noise}(i,j) \sim N(\mu, \sigma)$
- **Потеря информации**
(*data drop-out noise*)
 - **Соль и перец:**
случайные черные и
белые пиксели
 - **Импульсный:**
случайные белые
пиксели

Подавление шума «соль и перец»

3x3



5x5



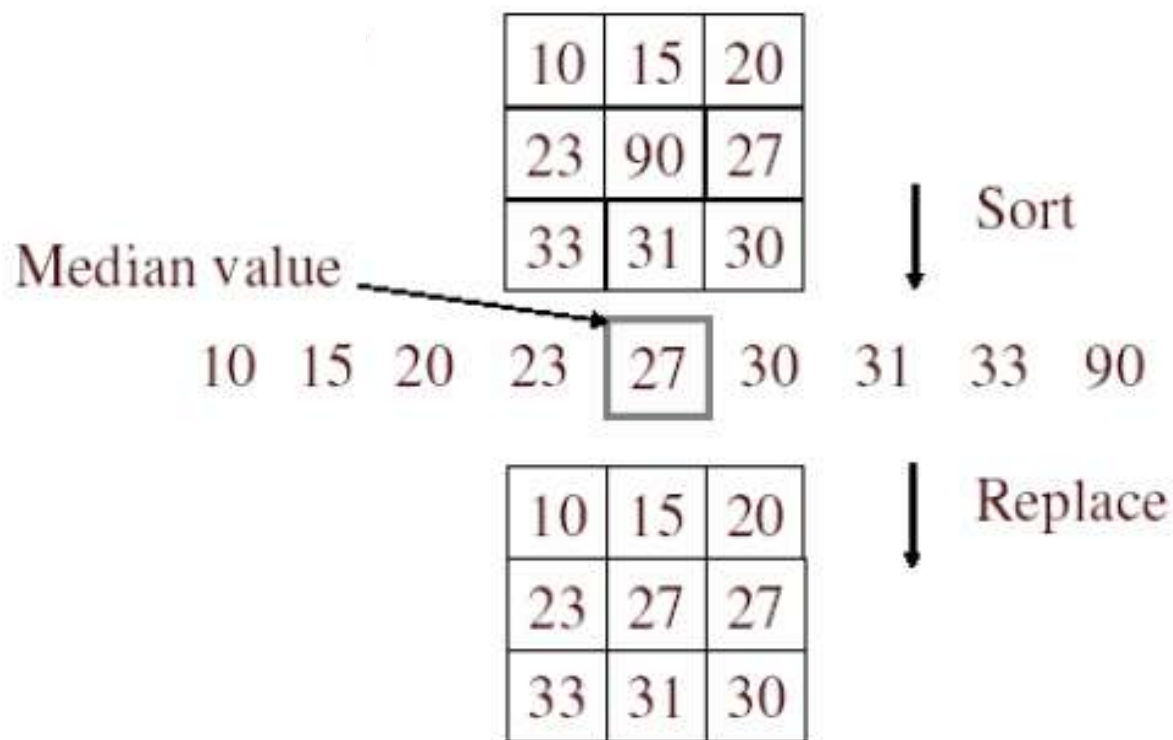
7x7



Применим фильтр Гаусса
Чем результат плох?

Медианный фильтр

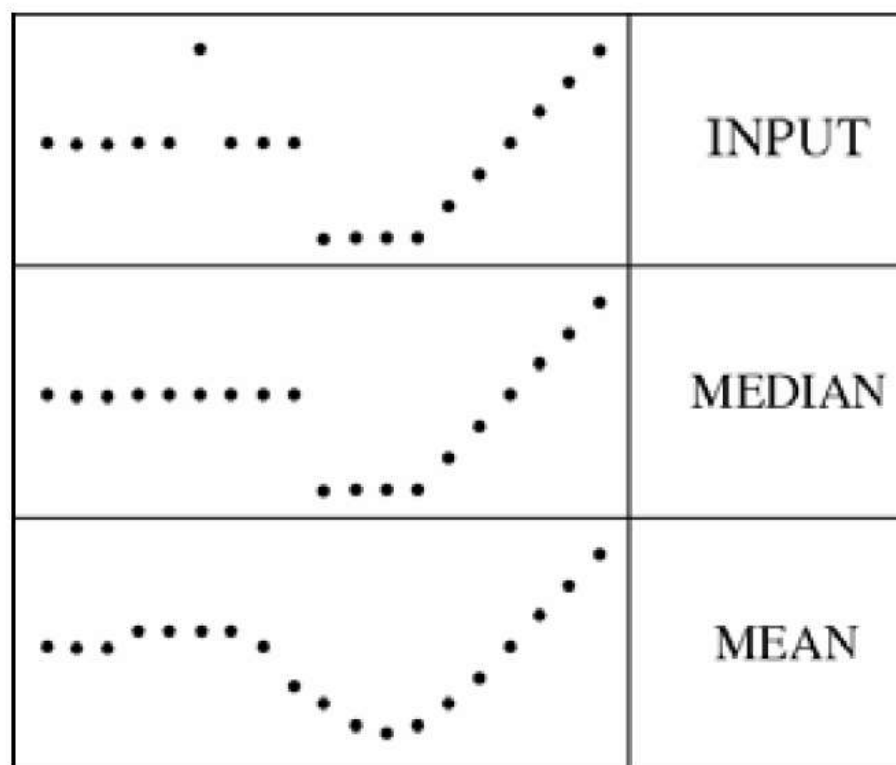
- Выбор медианы из выборки пикселей по окрестности данного



- Является ли фильтр линейным?

Медианный фильтр

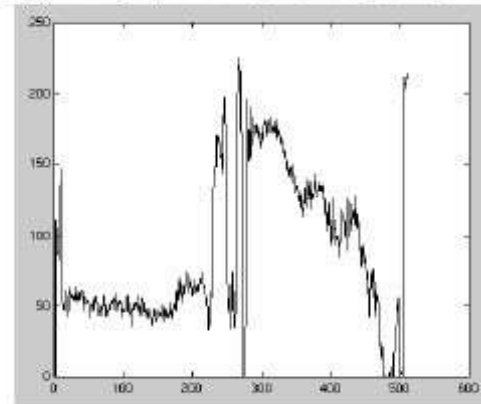
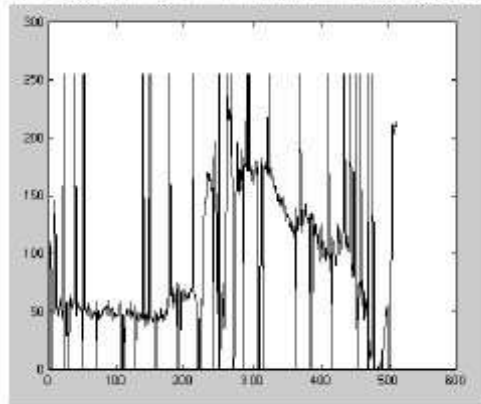
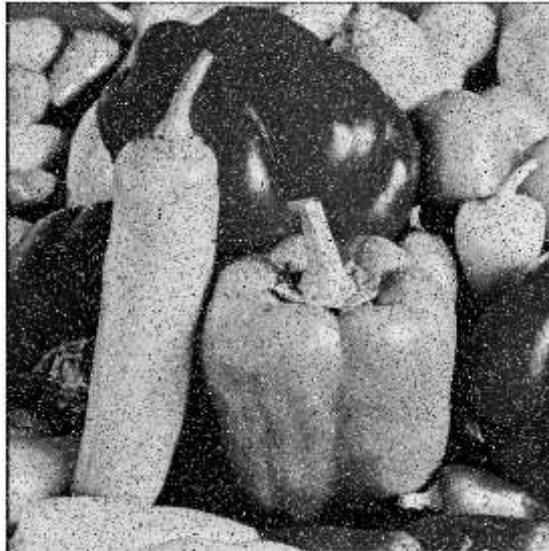
- В чем главное отличие медианного фильтра перед фильтром гаусса?
 - Устойчивость к выбросам (outliers)



Фильтры размером в 5 пикселей

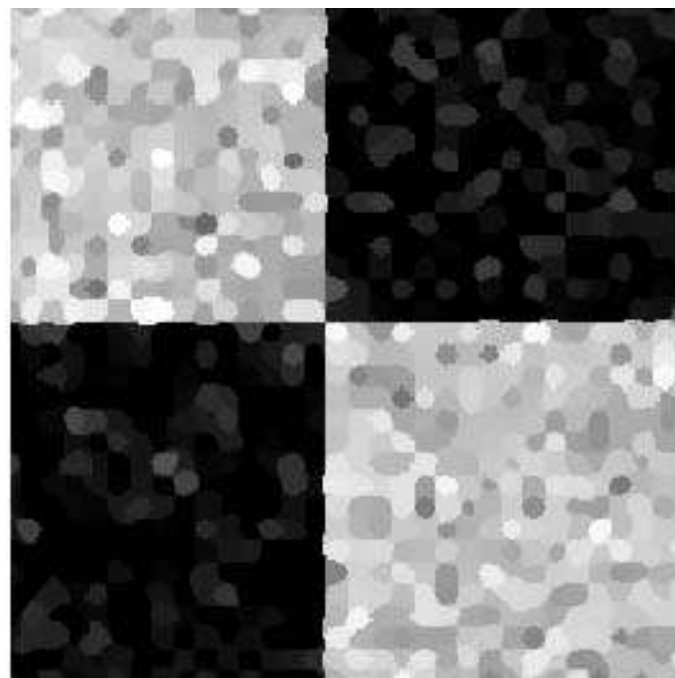
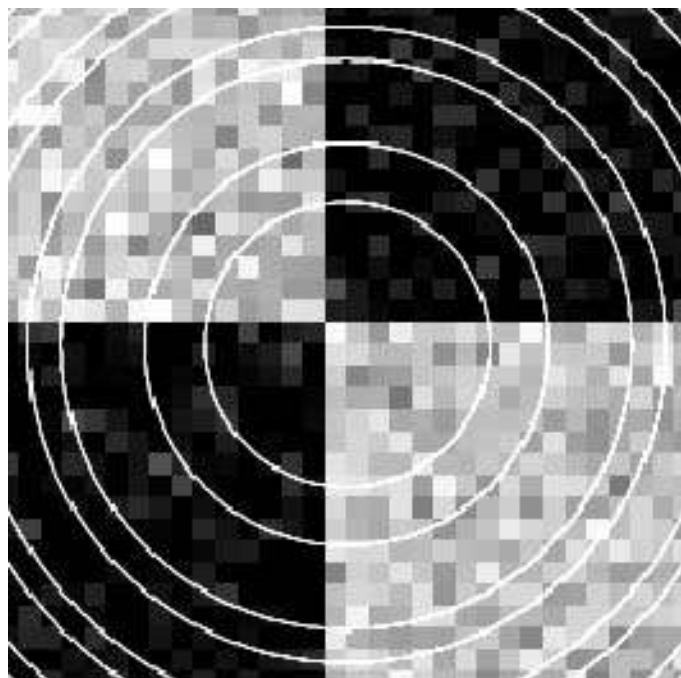
Медианный фильтр

Шум «соль и перец» Медианная фильтрация



Медианный фильтр

Результат применения медианного фильтра с радиусом в 7 пикселей к изображению с шумом и артефактами в виде тонких светлых окружностей.



Сравнение фильтров

3x3

5x5

7x7

Гауссов



Медианный



Как оценить качество шумоподавления?

- Оценить результат можно только сравнив исходное изображение без шума с результатом шумоподавления по зашумлённому изображению

исходное
изображение
без шума



Результат
шумоподавления

- Т.е. нужна тестовая выборка пар изображений и метрики для сравнения изображений

Метрика PSNR

- Пусть X – изображение без шума, Y – изображение с шумом после шумоподавления
- Среднеквадратичная ошибка (MSE)

$$MSE = \frac{1}{N} \sum_{i=1}^N (x_i - y_i)^2 \quad N - \text{число пикселей}$$

- Пиковое отношение сигнал/шум (PSNR)

$$PSNR_{dB} = 10 \lg \frac{M^2}{MSE} \quad M - \text{максимальное значение пикселя}$$

Ограничения метрики на примере Lena

PSNR и MSE не учитывают особенности человеческого восприятия, пусть и часто используются на практике



Оригинальное
изображение
Lena (Lenna)

- Обрезанная (512x512) часть изображения с разворота Playboy, Nov 1972
- Самый популярный, но не первый случай использования Playboy в обработке изображений (первый в 1961)
- Пригласили на 50ую конференцию [Society for Imaging Science and Technology](#) (IS&T) in 1997

Метрики качества

У этих изображений одинаковые PSNR с оригиналом (примерно 25 dB)



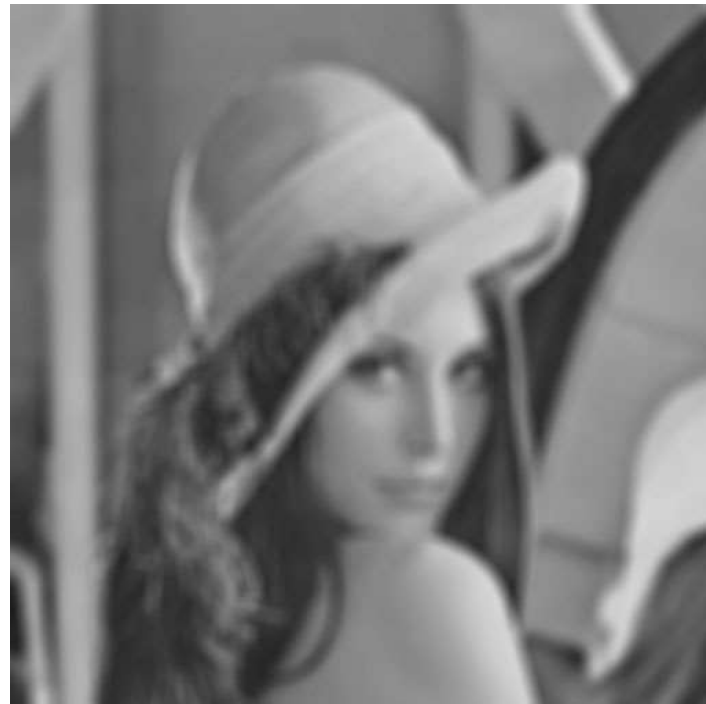
Повышена контрастность Добавлен белый гауссов шум

Метрики качества

И у этих – тоже примерно 25 dB!



Добавлен импульсный шум



Размытие

Метрики качества

- И у этого – тоже!

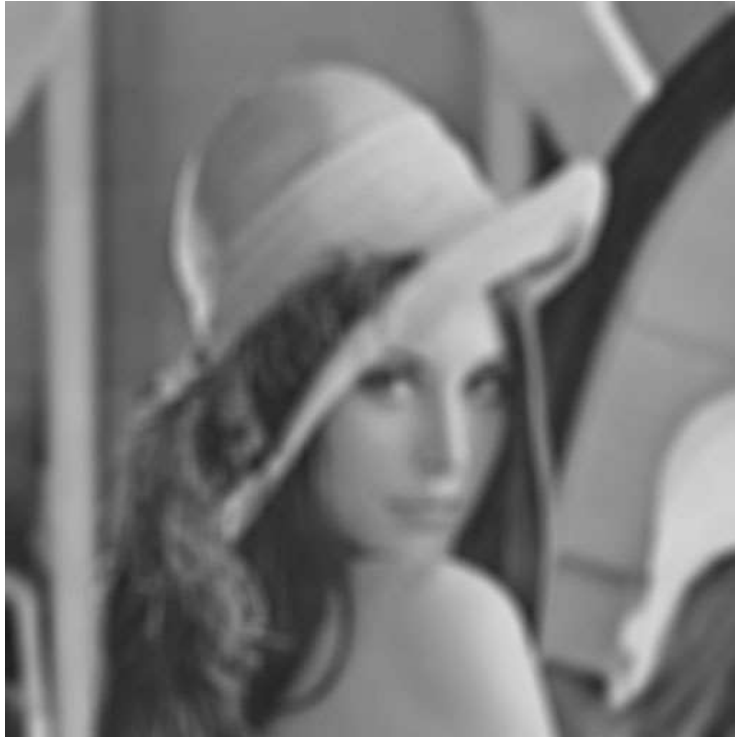


Артефакт блочности после JPEG

Метрики качества

- Вывод: PSNR не всегда отражает реальный видимый уровень искажений.
- Как улучшить?
 - Использовать функцию чувствительности глаза к различным частотам (CSF)
 - Использовать свойство маскировки
 - Использовать равномерные к восприятию цветовые пространства (CIE Lab, CIEDE2000)
 - Использовать perception loss посчитанный по нейросетевым признакам

Повышение резкости



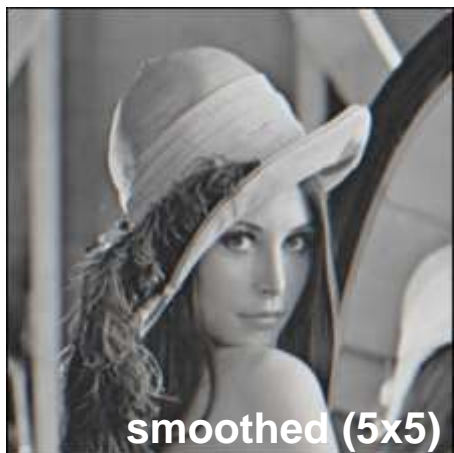
Иногда с помощью
фильтрации можно немного
повысить резкость
изображения (подчеркнуть
края)

Повышение резкости

Что теряется при сглаживании?



—



=



Добавим дополнительно высокие частоты:



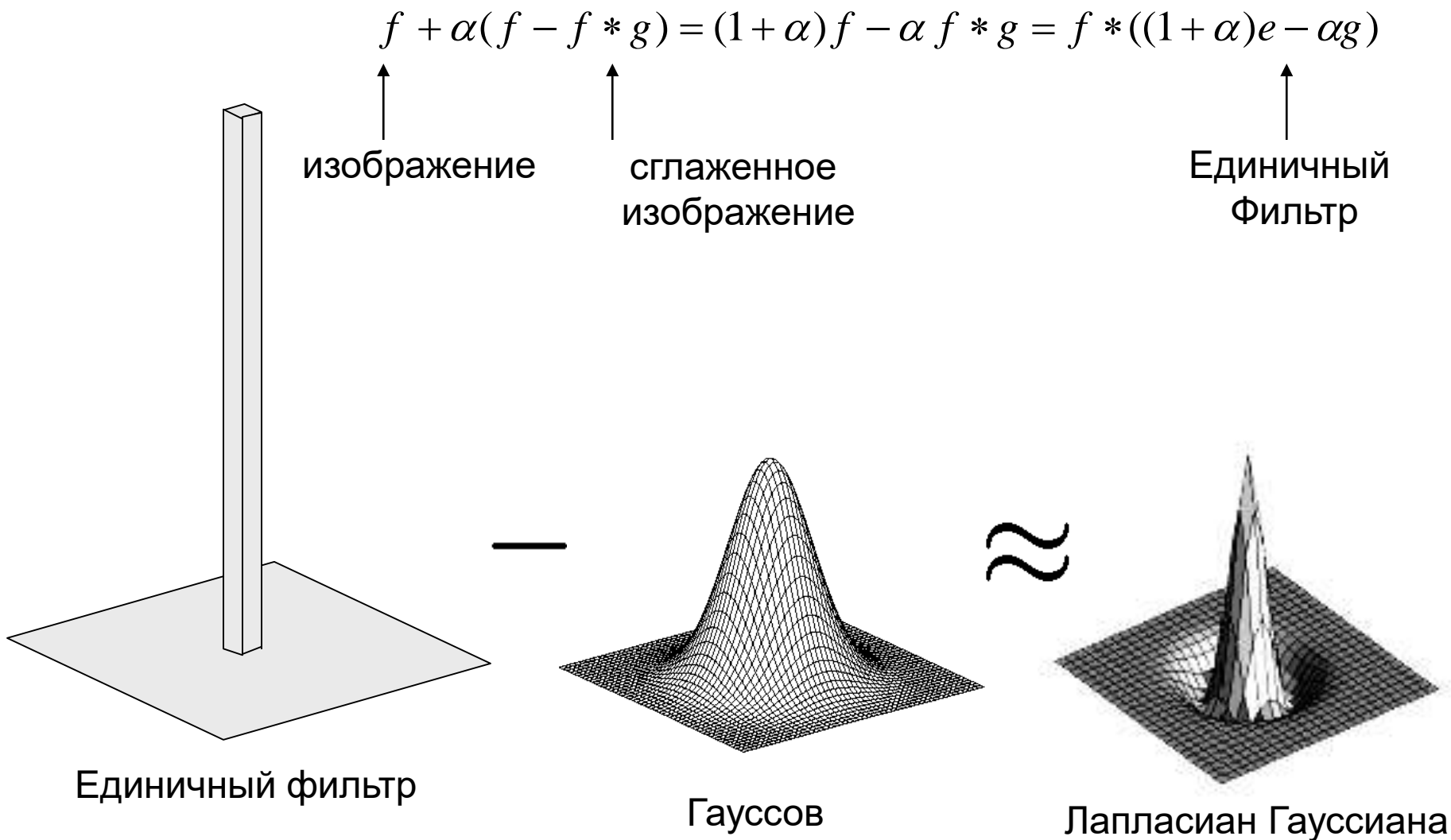
+ α



=



Фильтр Unsharp



Пример повышения резкости

Ядро
свертки

$$\frac{1}{10} \cdot \begin{vmatrix} -1 & -2 & -1 \\ -2 & 22 & -2 \\ -1 & -2 & -1 \end{vmatrix}$$



Выделение краёв

Пример



Что можно сказать про изображение?

Что такое край (edge)?

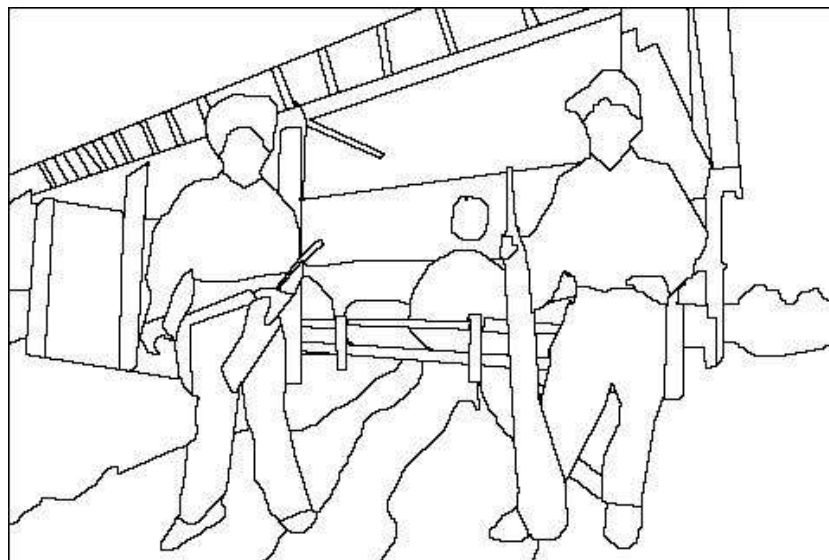


Разметка человеком

изображение



разметка вручную



Berkeley segmentation database:

<http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/>

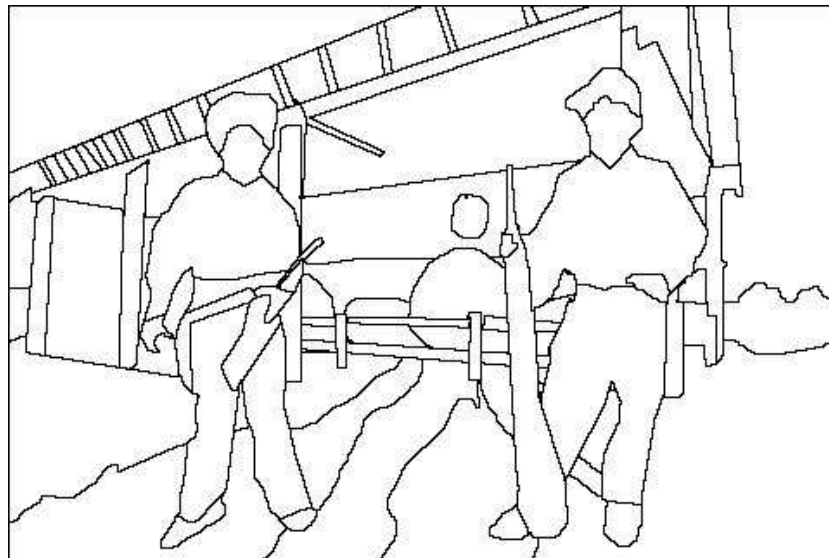
Края обычно соответствует границам визуально отличающихся друг от друга областей изображения

Края и области

изображение



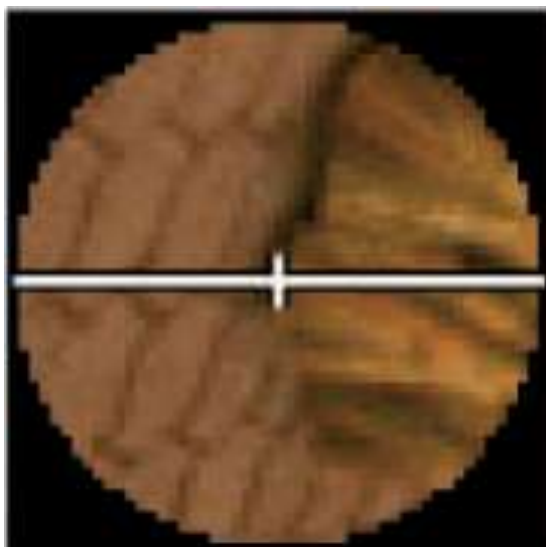
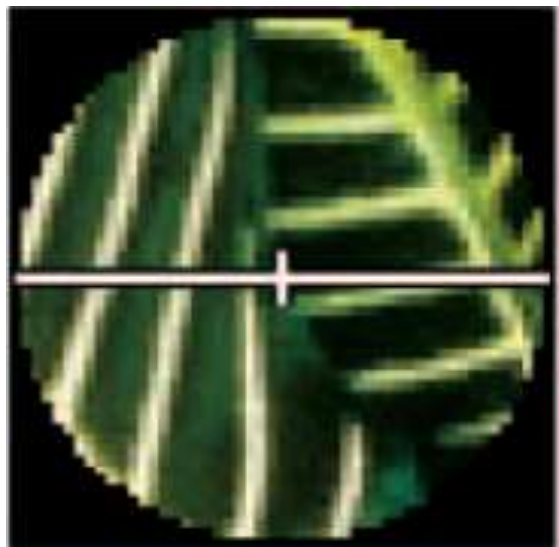
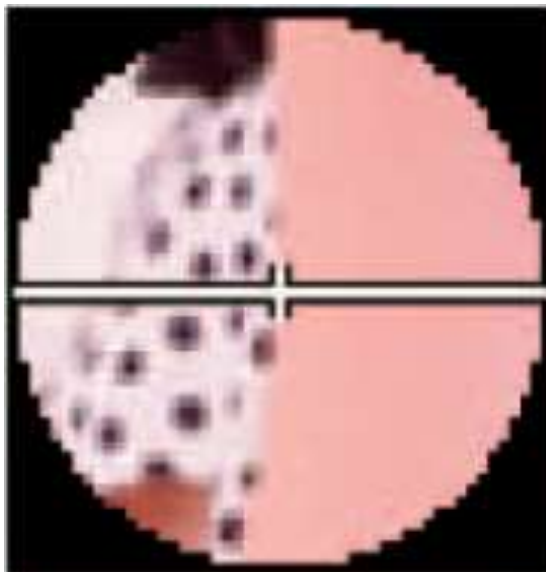
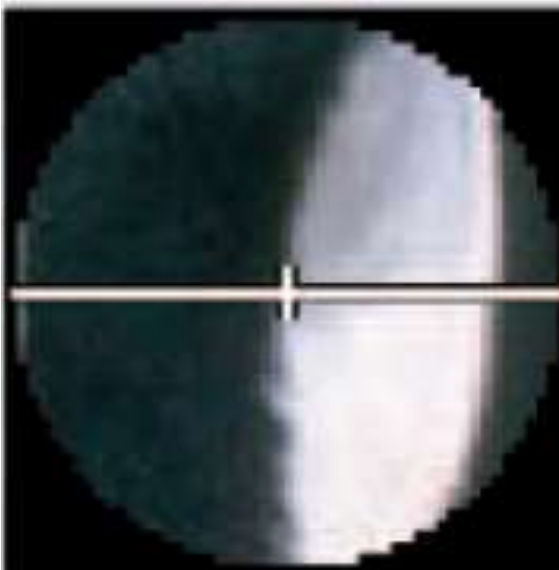
разметка вручную



Смежные задачи:

- Нахождение краёв в изображении
- Разделение изображения на визуально однородные области (визуальная сегментация)

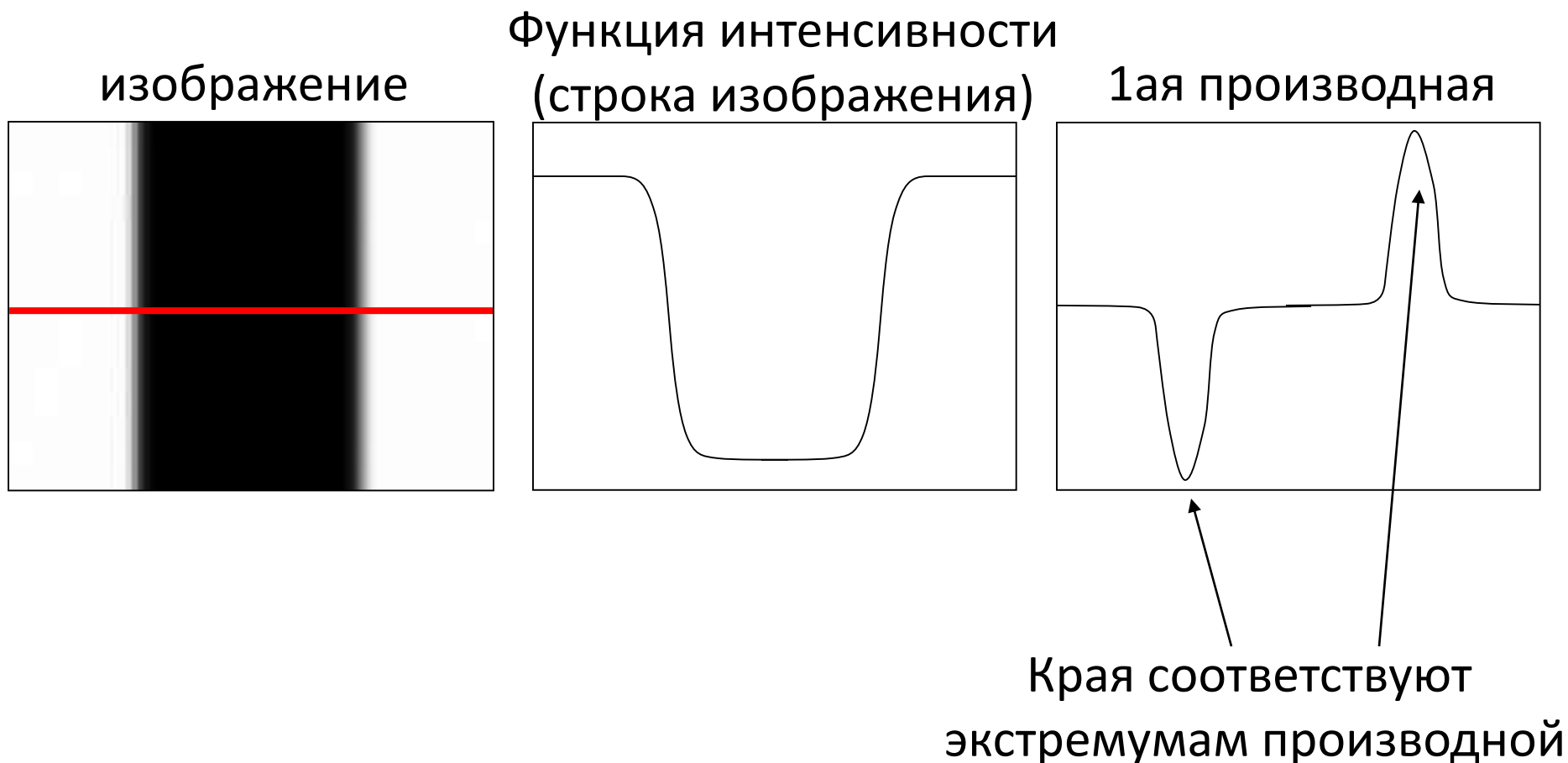
Примеры краёв



Какие признаки
используются?

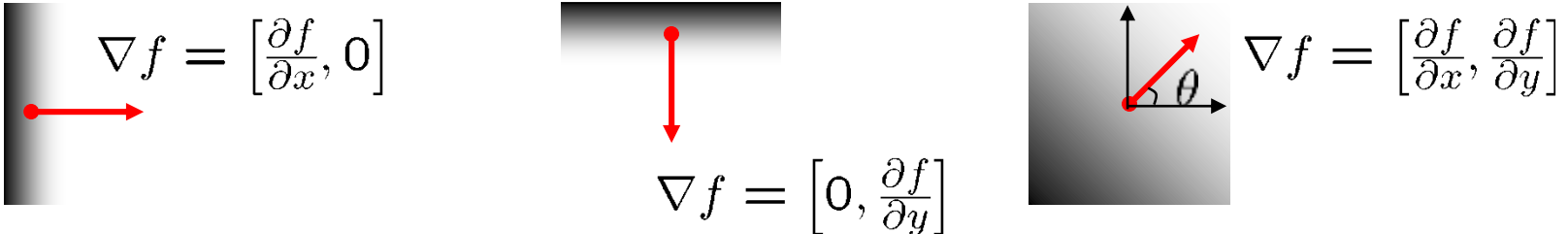
Базовое определение края

- Край – это точка резкого изменения значений функции интенсивности изображения



Градиент изображения

- Градиент изображения: $\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]$

- 

Градиент направлен в сторону наибольшего изменения интенсивности

Направления градиента задается как: $\theta = \tan^{-1} \left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x} \right)$

- Как направление градиента соответствует направлению края?
- Сила края* задается величиной (нормой) градиента:

$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Дифференцирование и свёртка

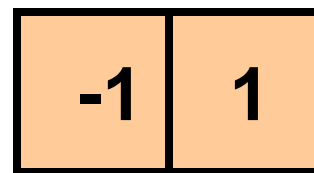
- Для функции 2х переменных, $f(x,y)$:

$$\frac{\partial f}{\partial x} = \lim_{\varepsilon \rightarrow 0} \left(\frac{f(x + \varepsilon, y) - f(x, y)}{\varepsilon} \right)$$

- Разностная производная:

$$\frac{\partial f}{\partial x} \approx \frac{f(x_{n+1}, y) - f(x_n, y)}{\Delta x}$$

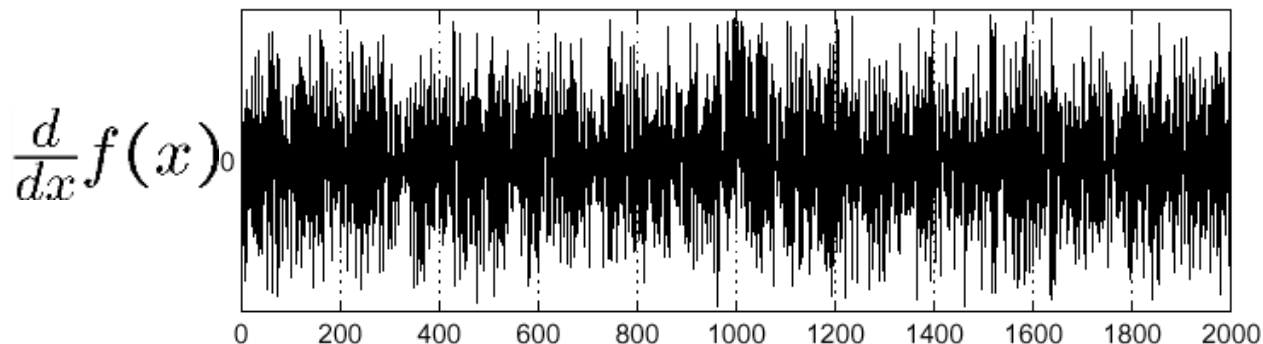
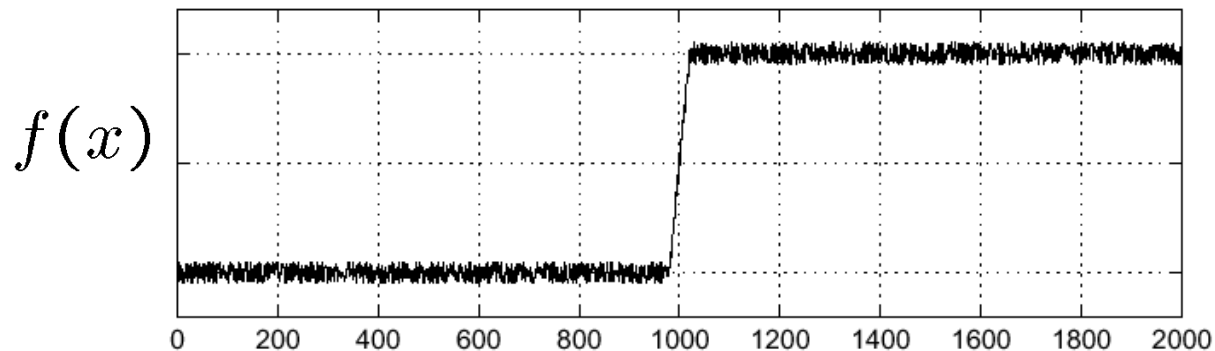
- Разностная производная - линейная и инвариантная к переносу
- Можно записать как свёртку



Простейший фильтр

Влияние шума

- Рассмотрим строку или столбец изображения
 - Интенсивность от положения можно рассматривать как сигнал

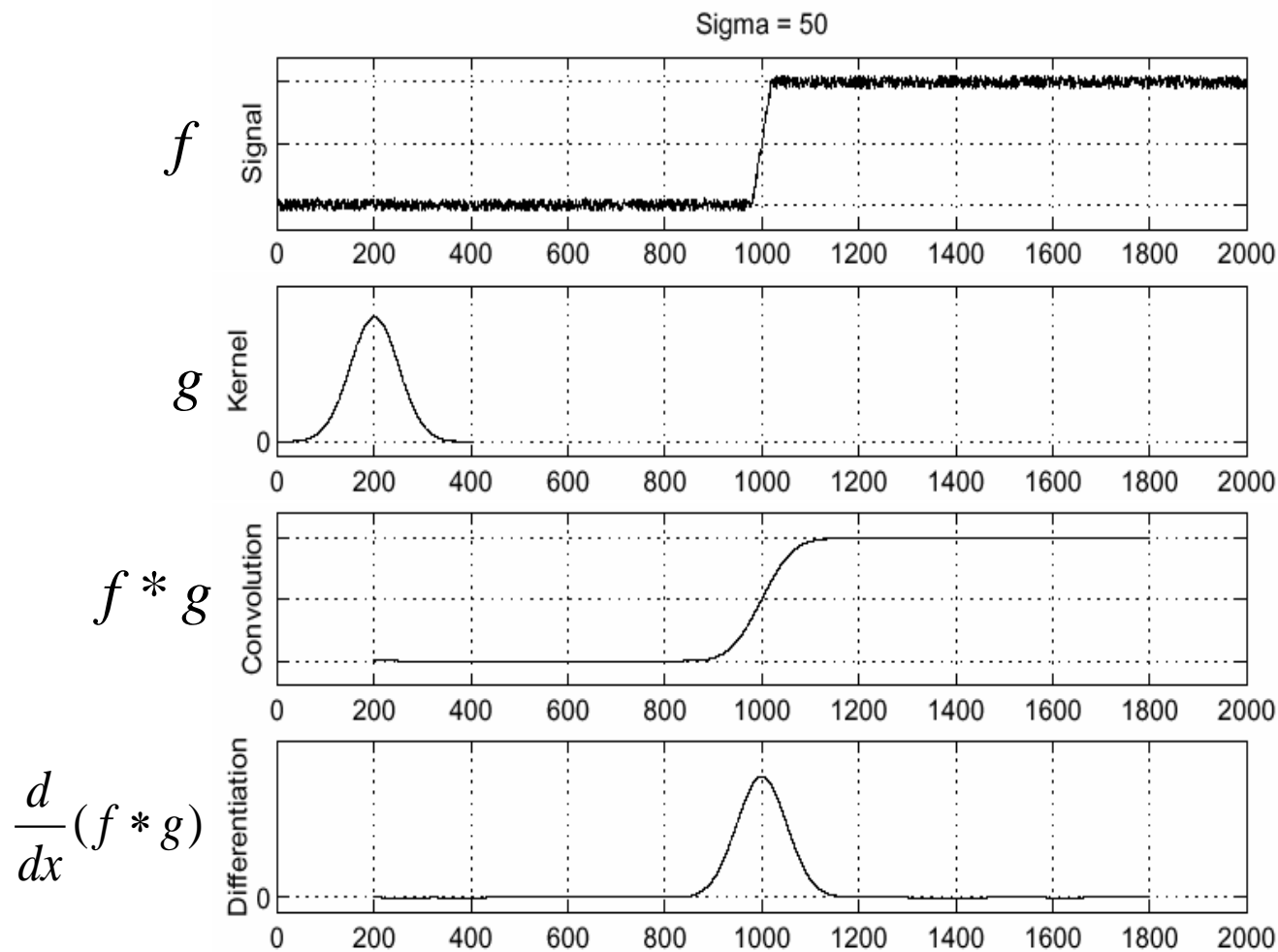


Край исчез

Влияние шума

- Разностные производные очень чувствительны к шуму
 - Зашумленные пиксели отличаются от соседей
 - Чем сильнее шум, тем выше отклик
- Сглаживание
 - Сглаживание делает все пиксели (зашумленные?) чуть более похожими на соседей

Предобработка (сглаживание)



- Для поиска краев ищем пики в: $\frac{d}{dx}(f * g)$

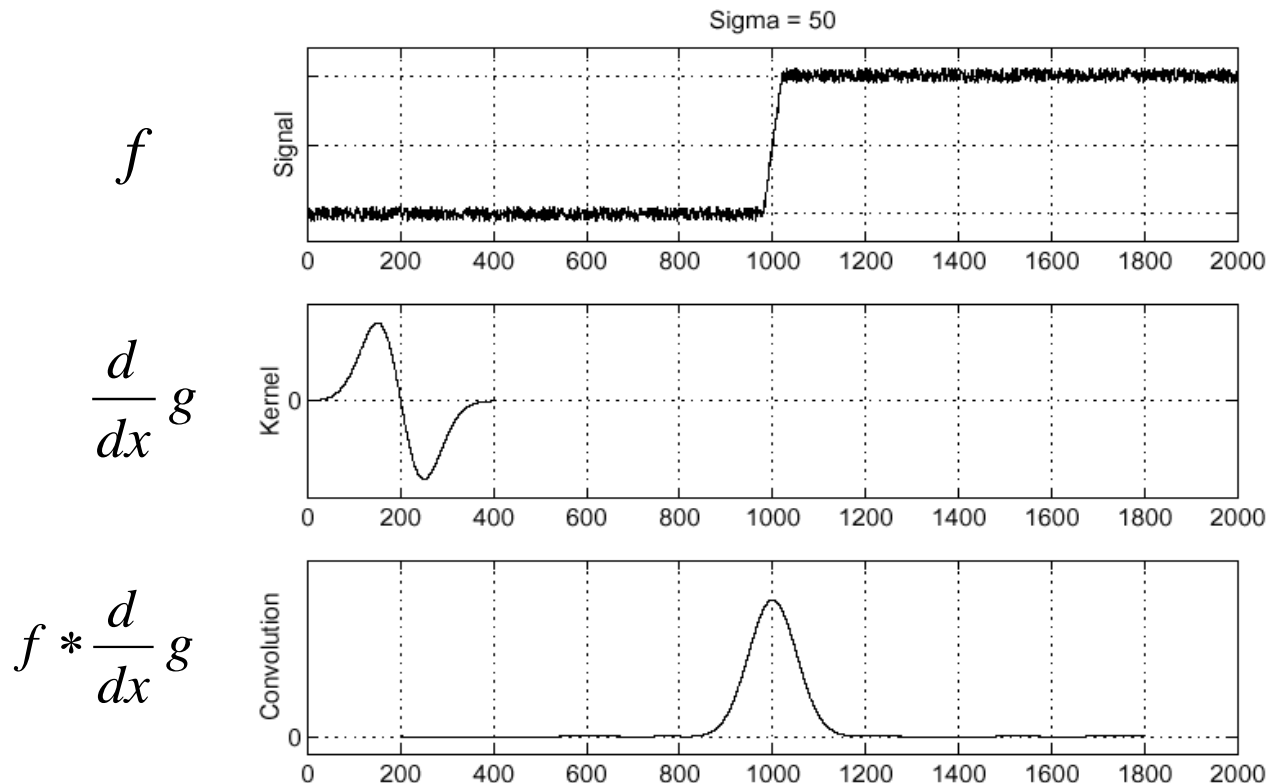
Свойства свертки

- Операции свертки и дифференцирования

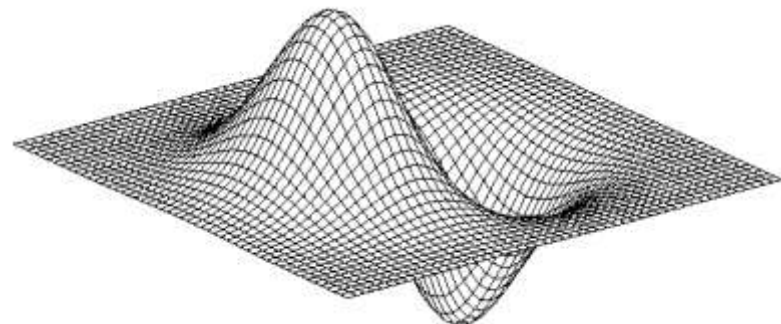
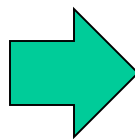
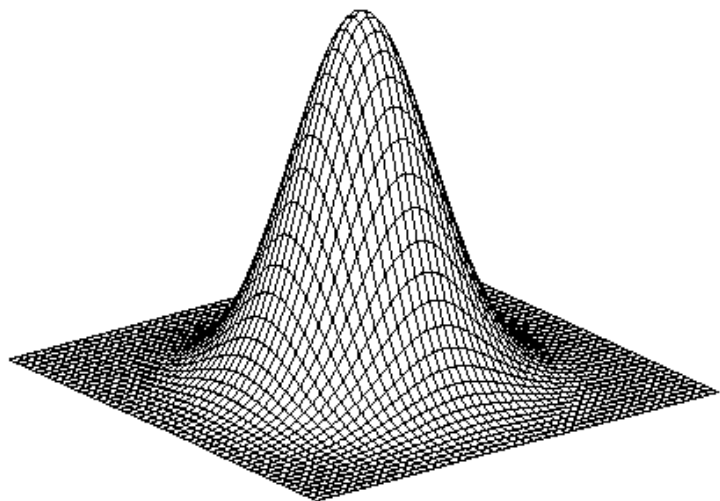
ассоциативны:

$$\frac{d}{dx}(f * g) = f * \frac{d}{dx}g$$

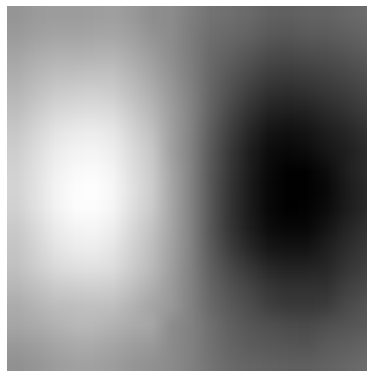
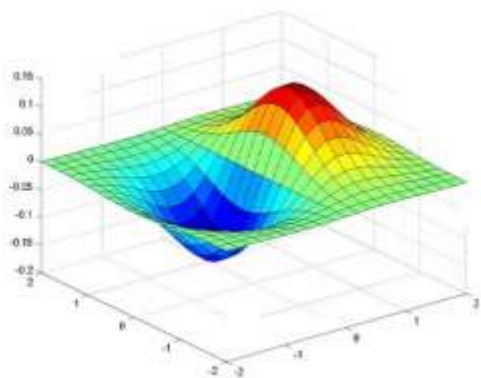
- Это экономит 1 операцию:



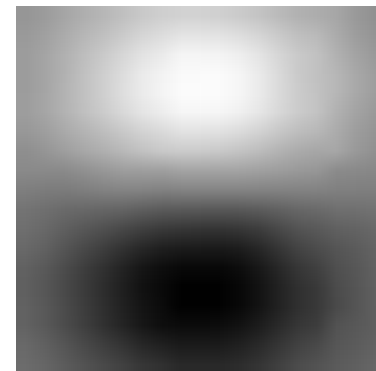
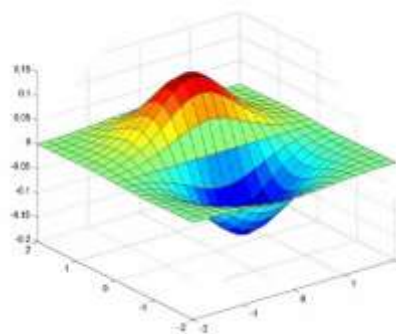
Производная фильтра Гаусса



По y



По x :



Известные фильтры

Несколько фильтров, по разному оценивающие производные по направлению и интегрирующие шумоподавление:

$$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

Робертса

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

Превитт

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Собель 3x3

$$\begin{bmatrix} +3 & +10 & +3 \\ 0 & 0 & 0 \\ -3 & -10 & -3 \end{bmatrix} \quad \begin{bmatrix} +3 & 0 & -3 \\ +10 & 0 & -10 \\ +3 & 0 & -3 \end{bmatrix}$$

Scharr фильтр

$$S_x = \begin{bmatrix} 1 & 2 & 0 & -2 & -1 \\ 4 & 8 & 0 & -8 & -4 \\ 6 & 12 & 0 & -12 & -6 \\ 4 & 8 & 0 & -8 & -4 \\ 1 & 2 & 0 & -2 & -1 \end{bmatrix} \quad S_y = \begin{bmatrix} -1 & -4 & -6 & -4 & -1 \\ -2 & -8 & -12 & -8 & -2 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 8 & 12 & 8 & 2 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

Собель 5x5

Карта силы краев

Примеры:



Робертса



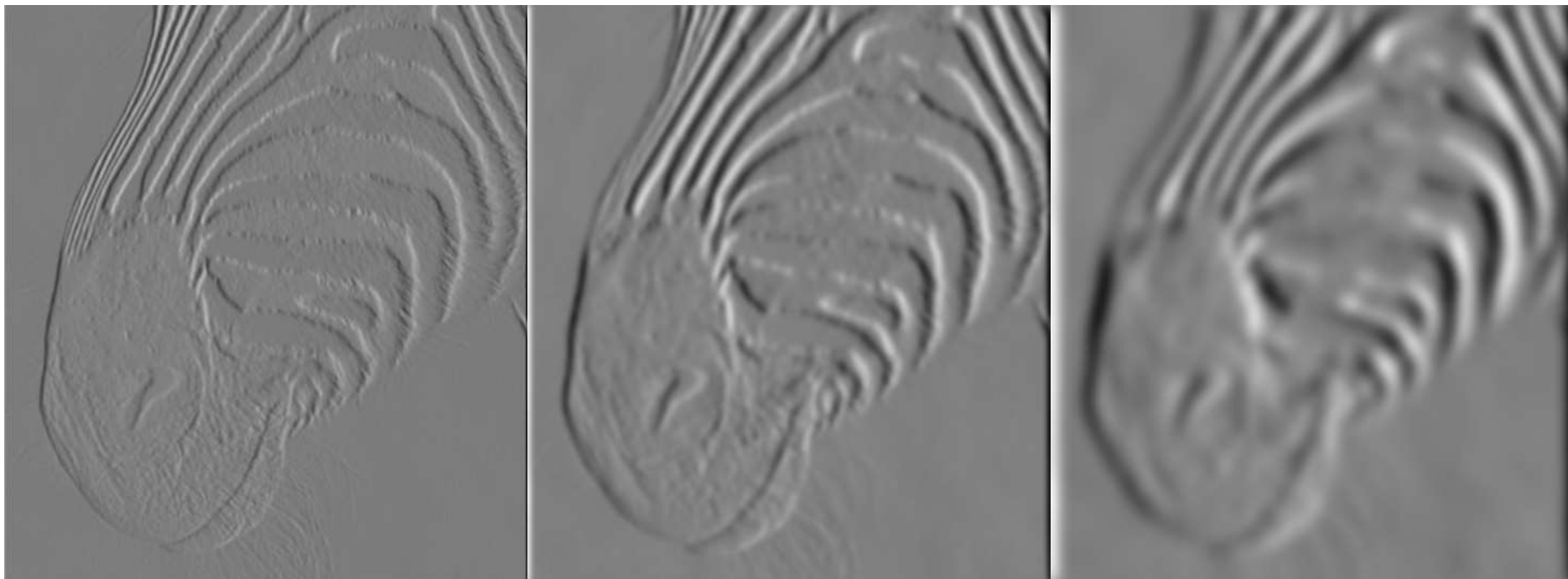
Превитт



Собея

Сглаживание и локализация

Применим сглаженные производные разного размера:



1 pixel

3 pixels

7 pixels

Сглаженные производные подавляют шум, но размывают края. Плюс края находятся на разных «масштабах»

Выделение краев

- Вычисление градиента – не идеальный метод для поиска краёв.



Исходное изображение



Карта силы краев

- Чего не хватает?
 - Точности – края получаются «толстые» и размытые
 - Ложные края, возникающие из-за шума

Детектор Canny

1. Свертка изображения с ядром – производной от фильтра гаусса
2. Поиск силы и направления градиента
3. Подавление не-максимумом / выделение локальных максимумов (Non-maximum suppression)
 - Утоньшение полос в несколько пикселей до одного пикселя
4. Связывание краев с использованием гистерезиса (двойного порога)
 - Выше верхнего порога – «сильные края»
 - Выше нижнего порога – «слабые края» (м.б. шумом)
 - Ниже нижнего порога – точно шум

Пример



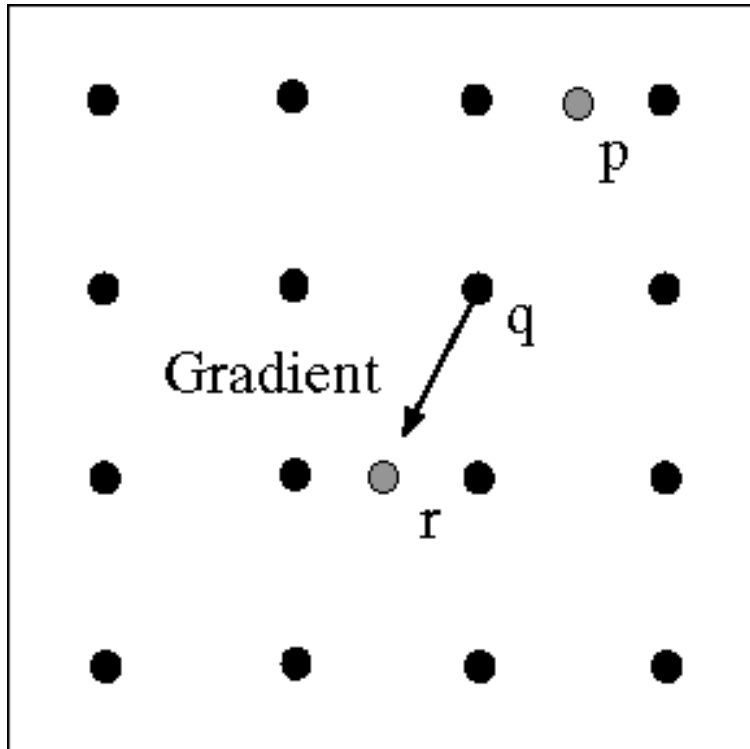
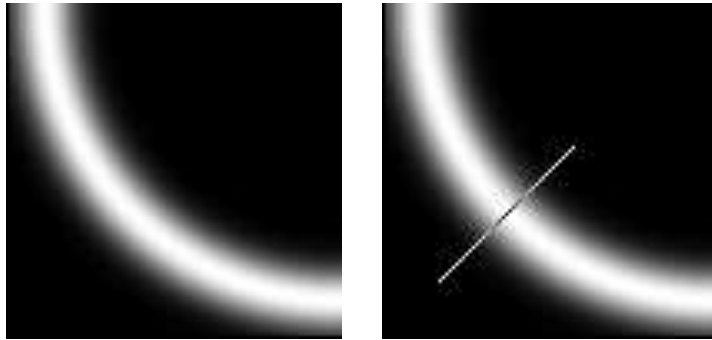
- Исходное изображение (Lena)

Пример



Норма градиента

Поиск локальных максимумов



Несколько вариантов фильтров:

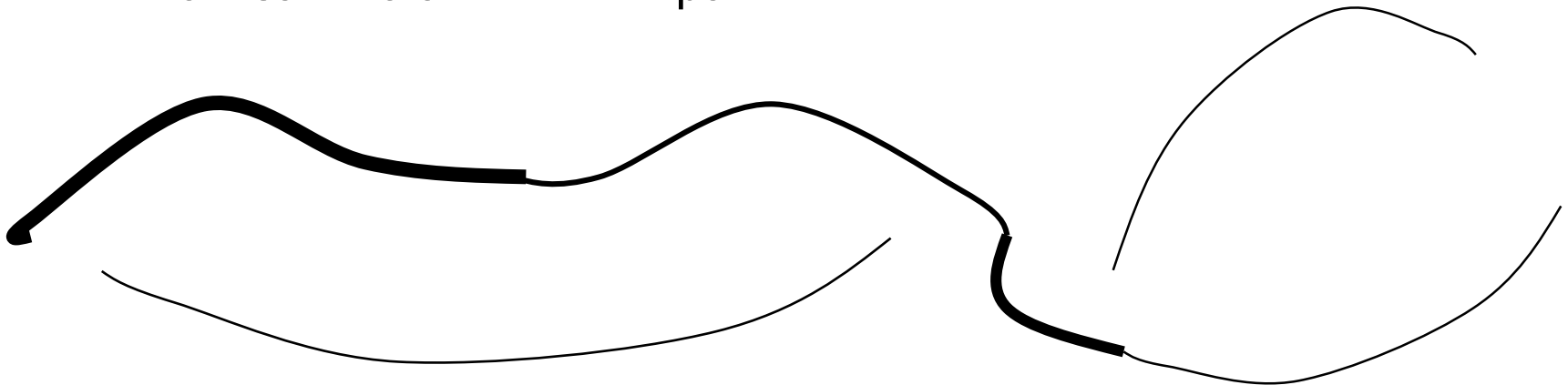
- Фильтр 3x3. Пиксель оставляем, если градиент в нём больше всех остальных в окрестности.
- Фильтр по направлению градиента. Оставляем градиент q , если значение больше p и r . Значения в p и r интерполируем.

Результат работы

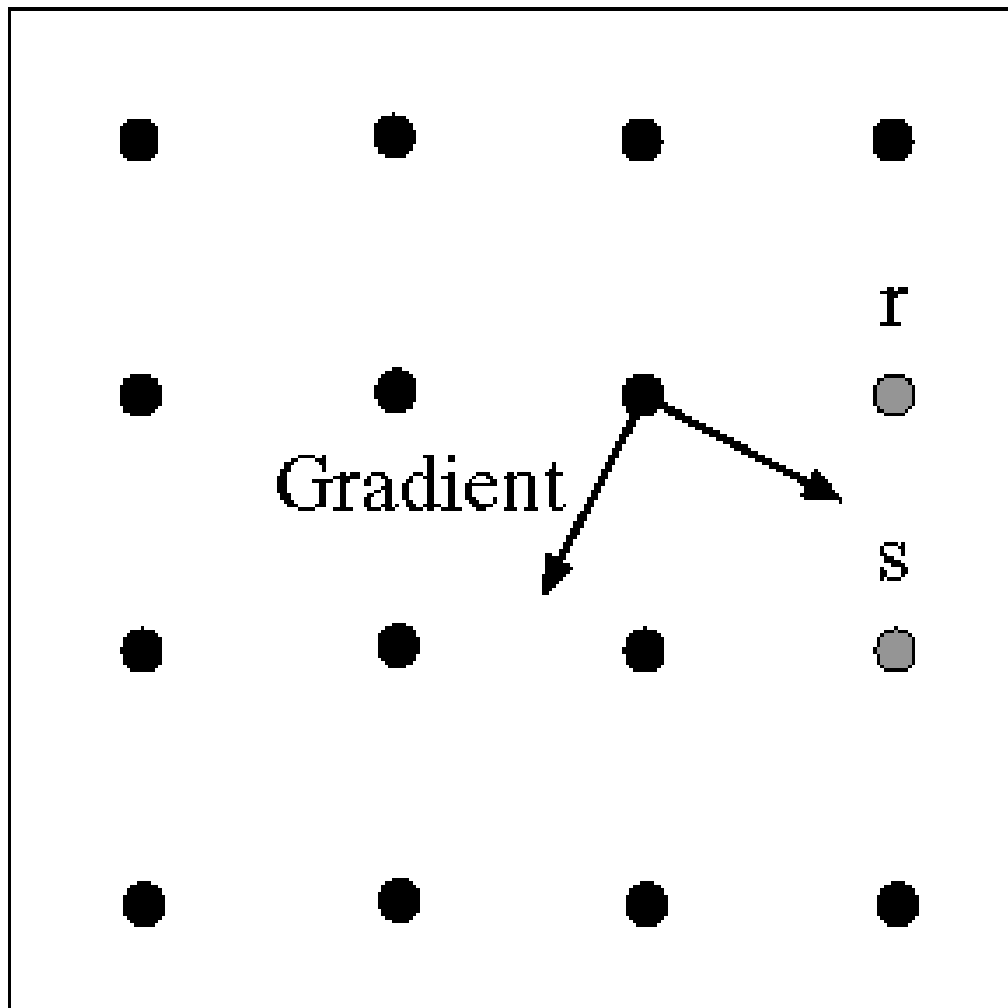


Прослеживание с гистерезисом

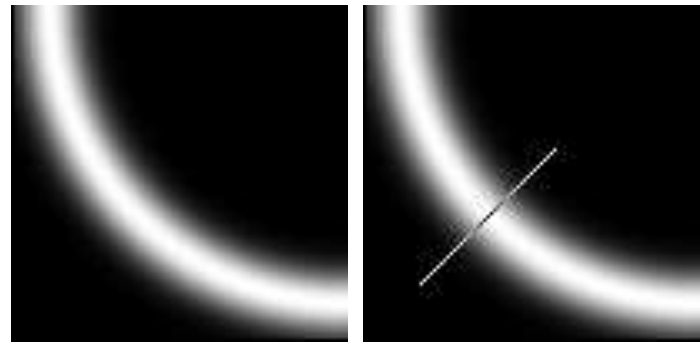
- Выбираем два порога «верхний» и «нижний»
- Размечаем все пиксели изображений (оставшиеся)
 - Выше верхнего – «сильные края»
 - Выше нижнего – «слабые края»
 - Ниже нижнего - шум
- Идея прослеживания:
 - Оставляем «слабые» края только в том случае, если они связаны с сильными краями



Прослеживание границ



- Пусть отмеченная точка – «слабый край», который ещё не проверен.
- Будем искать «сильного» соседа, переходя по соседним неподавленным краям вдоль границы



Эффект гистерезиса



Исходное изображение



Высокий порог
(сильные края)



Низкий порог
(слабые + сильные края)



Порог по гистерезису

Влияние σ (Размер ядра размытия)



original



Canny with $\sigma = 1$



Canny with $\sigma = 2$

Выбор σ зависит от задачи

- большое σ - поиск крупных границ
- маленькое σ - выделение мелких деталей

Резюме лекции

- Цветокоррекция изображения требует оценки до 24 параметров нелинейной модели
 - Калибровочный цветовой шаблон
 - Угадывание (оценка) параметров
- Линейная фильтрация (свёртка) изображения позволяет решать целый ряд задач – шумоподавление, повышение резкости, оценка градиента
- Выделение краёв изображения в простом случае можно достичь поиском локальных максимумов градиента яркости
- Всё это открытые задачи и сейчас активно продолжают исследоваться!