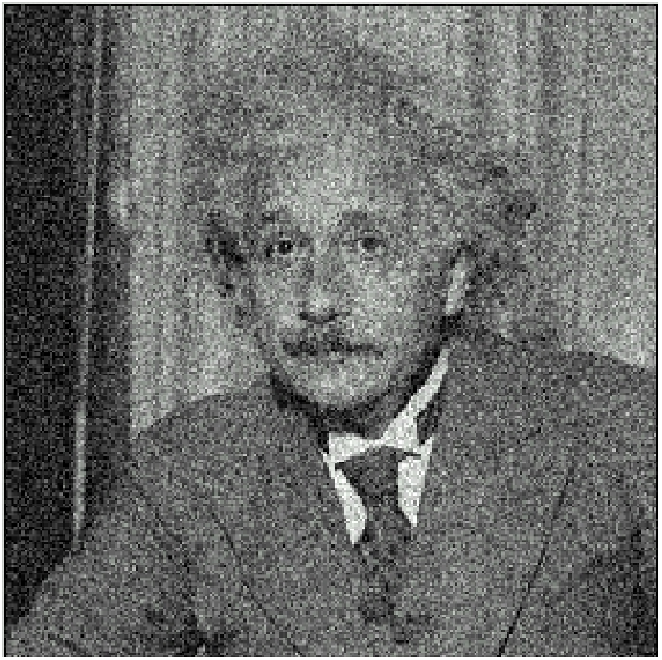
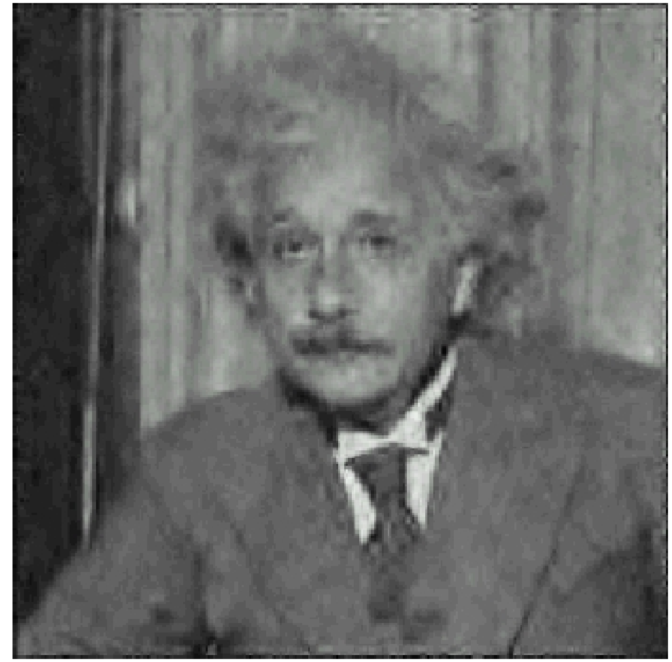




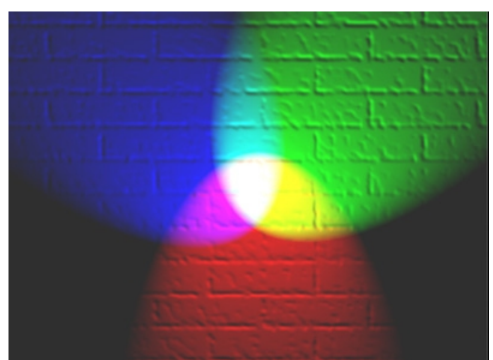
Анализ и обработка изображений

Представление растровых изображений

- Изображение разбивается на отдельные маленькие фрагменты - пиксели (pixel), причем хранятся их пространственные координаты, а каждому пикселю присваивается код его цвета.
- Чем больше глубина пикселя, тем шире диапазон доступных цветов.

Модель RGB

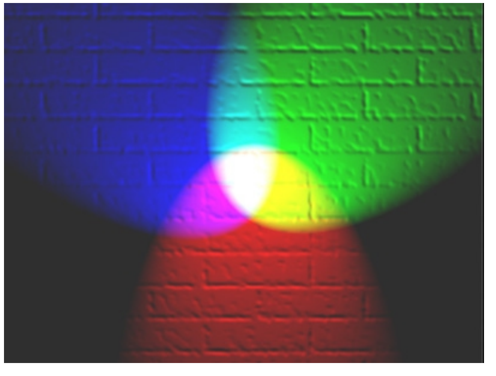
- Цвет можно представить в виде комбинации трех цветов: красного (Red, R), зеленого (Green, G), синего (Blue, B). Остальные цвета и их оттенки получаются за счет наличия или отсутствия этих составляющих.



https://ru.wikipedia.org/wiki/RGB

Модель RGB

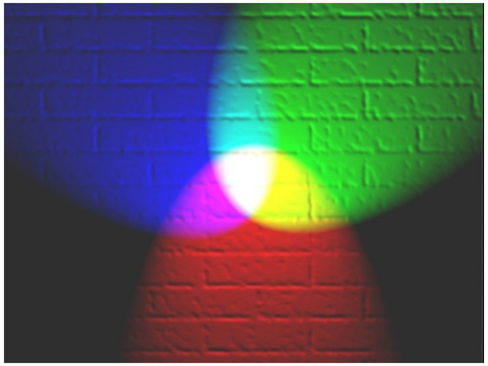
- Цвет для каждой точки кодируется 3 байтами (24 битами), и в десятичной системе счисления минимальные значения RGB (0,0,0) соответствуют черному цвету, а белому - максимальные с координатами (255, 255, 255).



https://ru.wikipedia.org/wiki/RGB

Модель RGB

- Чем больше значение байта цветовой составляющей, тем этот цвет ярче. Например, темно-синий кодируется тремя байтами (0, 0, 128), а ярко-синий (0, 0, 255).

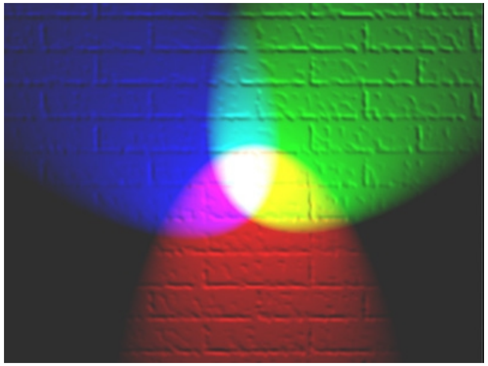


https://ru.wikipedia.org/wiki/RGB

Модель RGB

- Поскольку при кодировании цвета на пиксель приходится 24 бита, то число возможных оттенков составит

$$2^{24} = 16\ 777\ 216.$$



https://ru.wikipedia.org/wiki/RGB

Статистический анализ изображений

Статистический анализ изображений

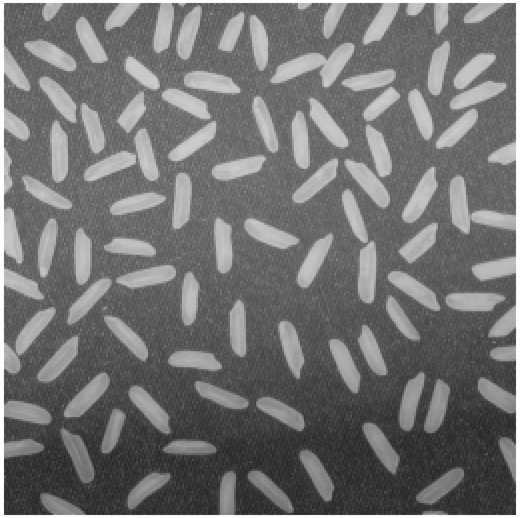
- Изображение – реализация случайной величины
- Использование статистических характеристик при улучшении изображений
- Контрастирование
- Попиксельные преобразования изображений

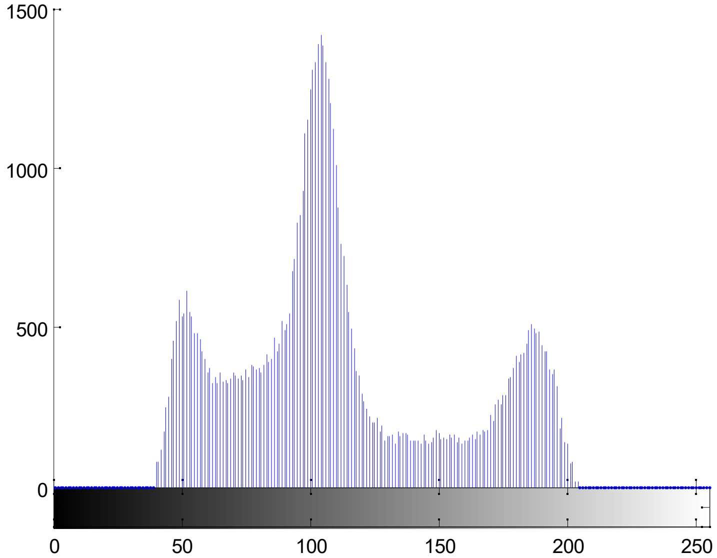
Изображение – реализация случайной величины

- Изображения могут рассматриваться как реализации случайной величины
- Цель статистического анализа изображений – оценивание по имеющейся реализации свойств случайной величины (закона распределения).

Гистограмма

- *Гистограмма* – таблица $H = \{H_l \mid l = 1,\dots,L\}$,
  H_l – количество пикселей с интенсивностью l.





H={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,78,80,115,174,25,282,399,460,521,587,532,5
43,616,547,535,482,483,464,427,401,360,373,325,345,327,358,332,336,327,341,359,350,340,349,335,370,343,382,37
6,369,372,360,383,414,391,403,467,424,449,521,491,508,543,677,714,827,854,926,1110,1150,1248,1307,1331,1389,1
419,1382,1330,1281,1203,1122,1008,876,762,724,636,547,494,435,363,348,293,267,245,221,203,200,218,171,192,14
5,159,157,164,135,172,161,170,170,163,147,147,147,143,136,164,146,134,140,155,179,170,148,154,150,166,155,162,
141,155,136,143,143,154,163,150,175,165,177,172,179,225,208,260,272,261,287,288,340,344,374,410,392,414,422,4
48,492,512,495,484,487,445,424,423,370,354,367,317,182,215,141,136,72,79,15,16,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}

Улучшение изображений с помощью попиксельных преобразований

Улучшение изображений

- **Цель**: преобразовать изображение таким образом, чтобы его было проще использовать при решении определенных задач.
- Общей теории улучшения изображений не существует.
- Набор специализированных средств, применяемых на усмотрение пользователя.
- Субъективность восприятия.

- **Методы**:
  – Попиксельные операции
    - Преобразование интенсивностей пикселей: не зависит от данных
    - Преобразование гистограмм: зависит от данных
    - Арифметические операции
  – Фильтрация

Преобразование интенсивностей пикселей

- Осуществляется попиксельно:

$$s = \mathrm{T}(r),$$

- где r – исходная интенсивность пикселя,

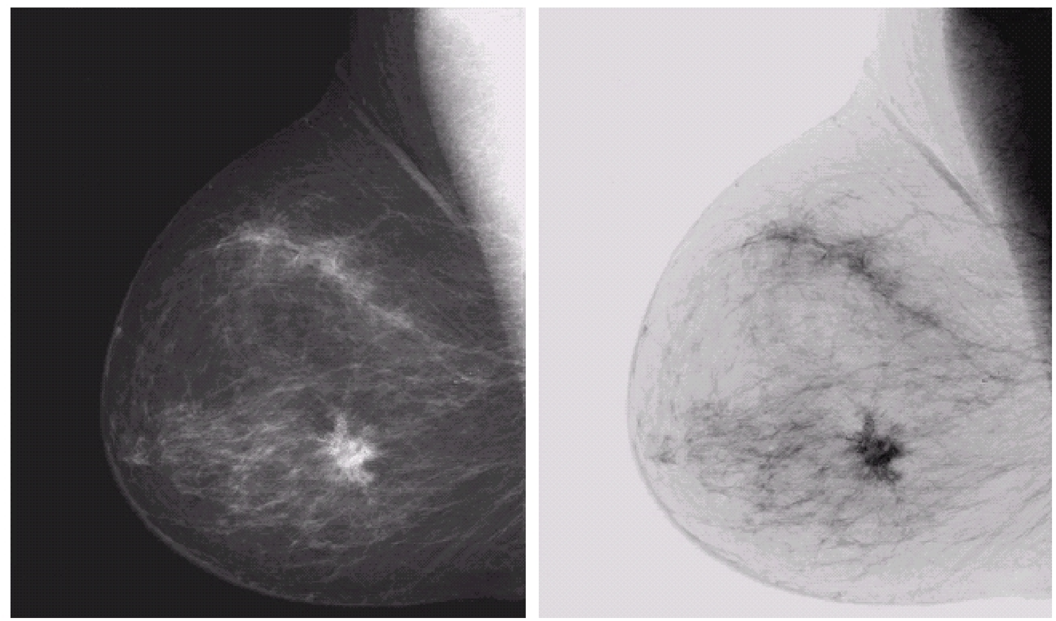
  s – результирующая интенсивность

Классы преобразований

- Обращение (негативизация)
- Логарифмические
- Степенные (полиномиальные)
- Кусочно-линейные

Негатив изображения

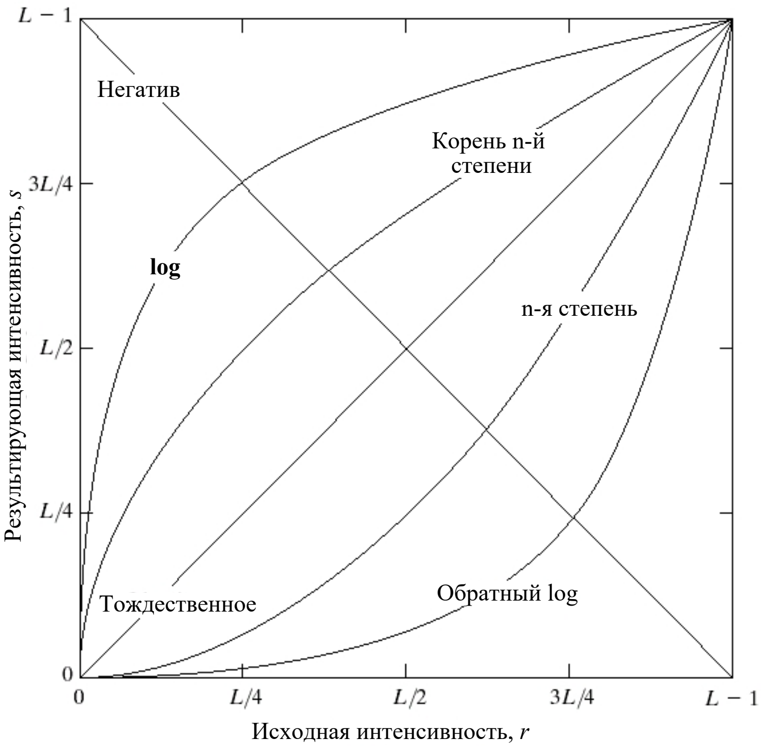
- $s = \mathrm{T}(r) = L-1-r$
- Аналогично фотонегативу.
- Применимо для улучшения белых или серых деталей на темном фоне.

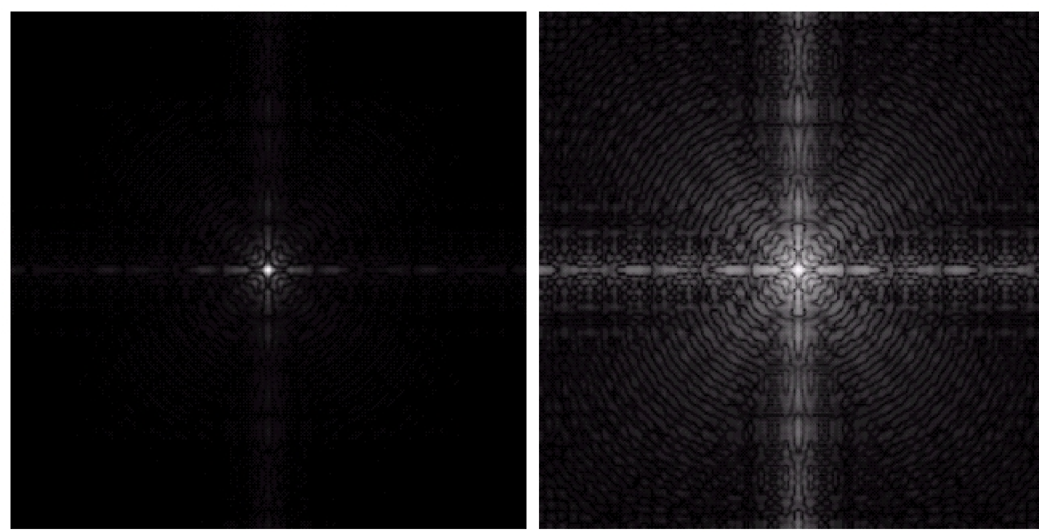


Логарифмические преобразования

- $s = \mathrm{T}(r) = c \log(1+r)$

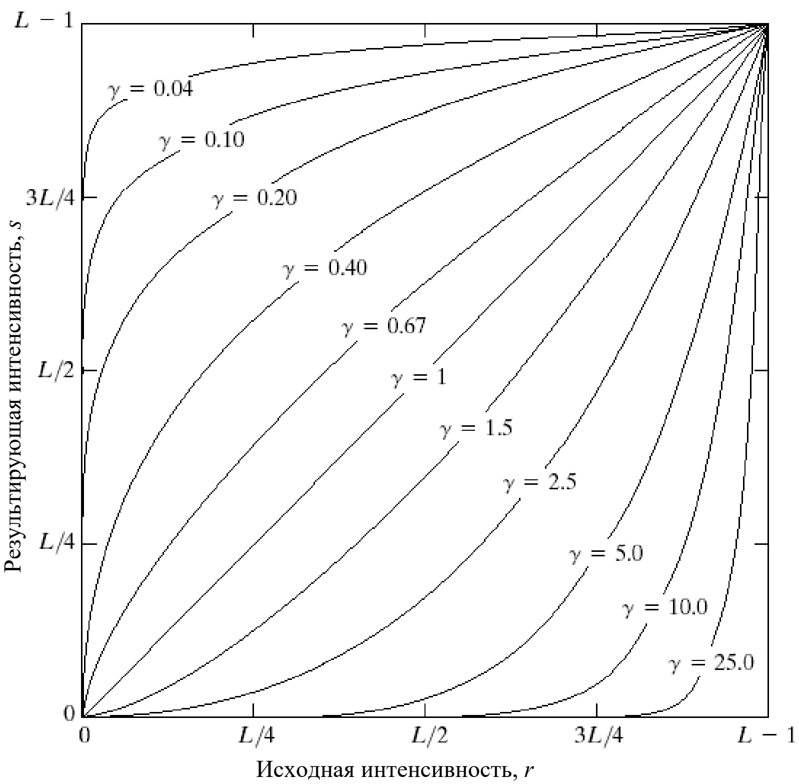
- Контрастирование «темных» значений интенсивностей улучшает восприятие деталей



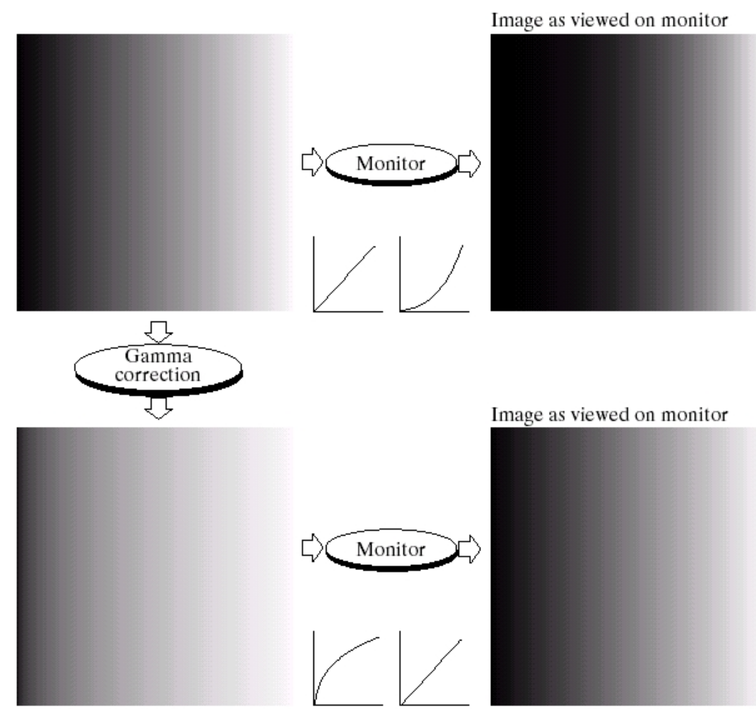


Степенное преобразование

- $s = T(r) = c\, r^{\gamma}$



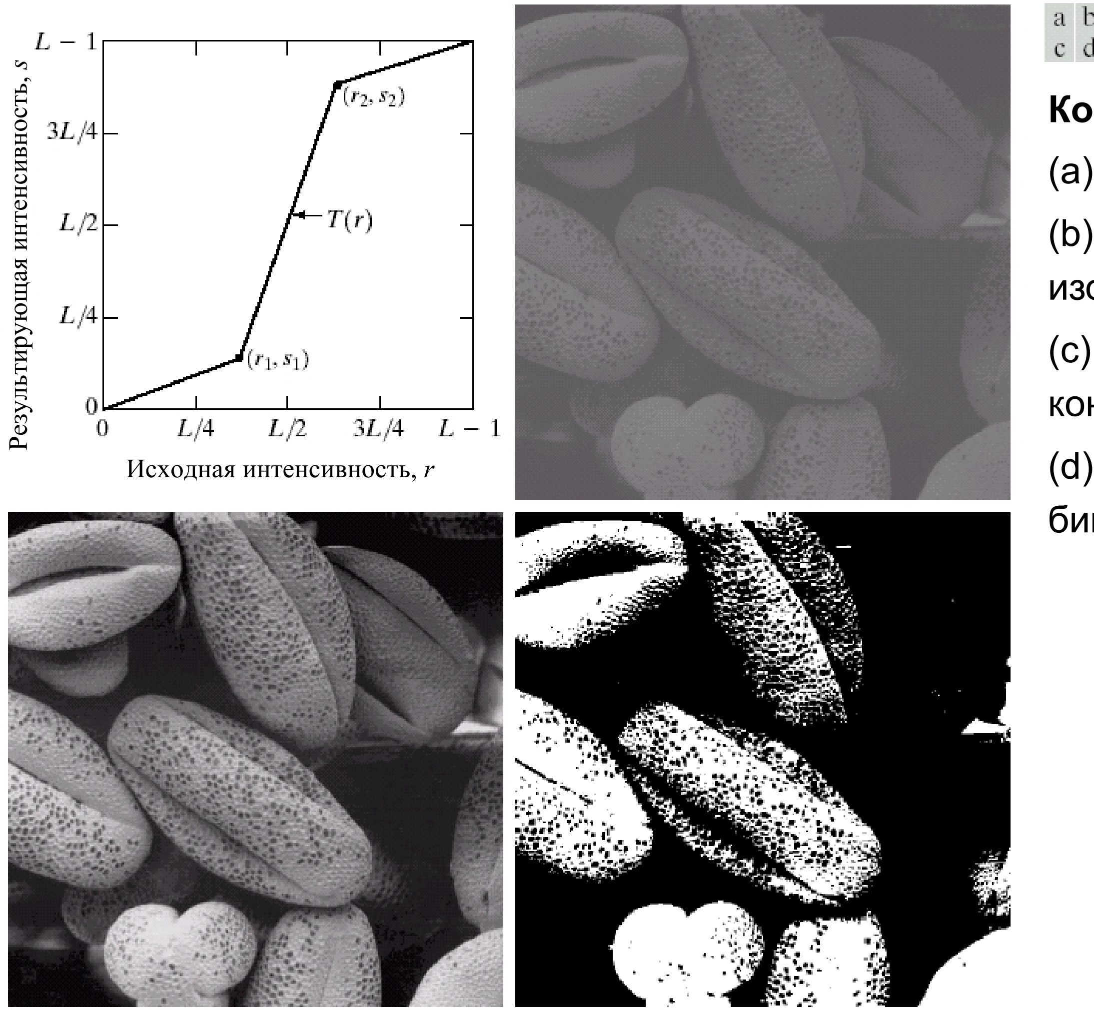
- Гамма-коррекция используется для адаптации изображений к особенностям дисплеев



Кусочно-линейные преобразования

- Позволяют конструировать сложные функции преобразования T(r).
  - Контрастирование
  - Выделение поддиапазона серого
  - Бинаризация
  - и т.д.

Контрастирование



Контрастирование

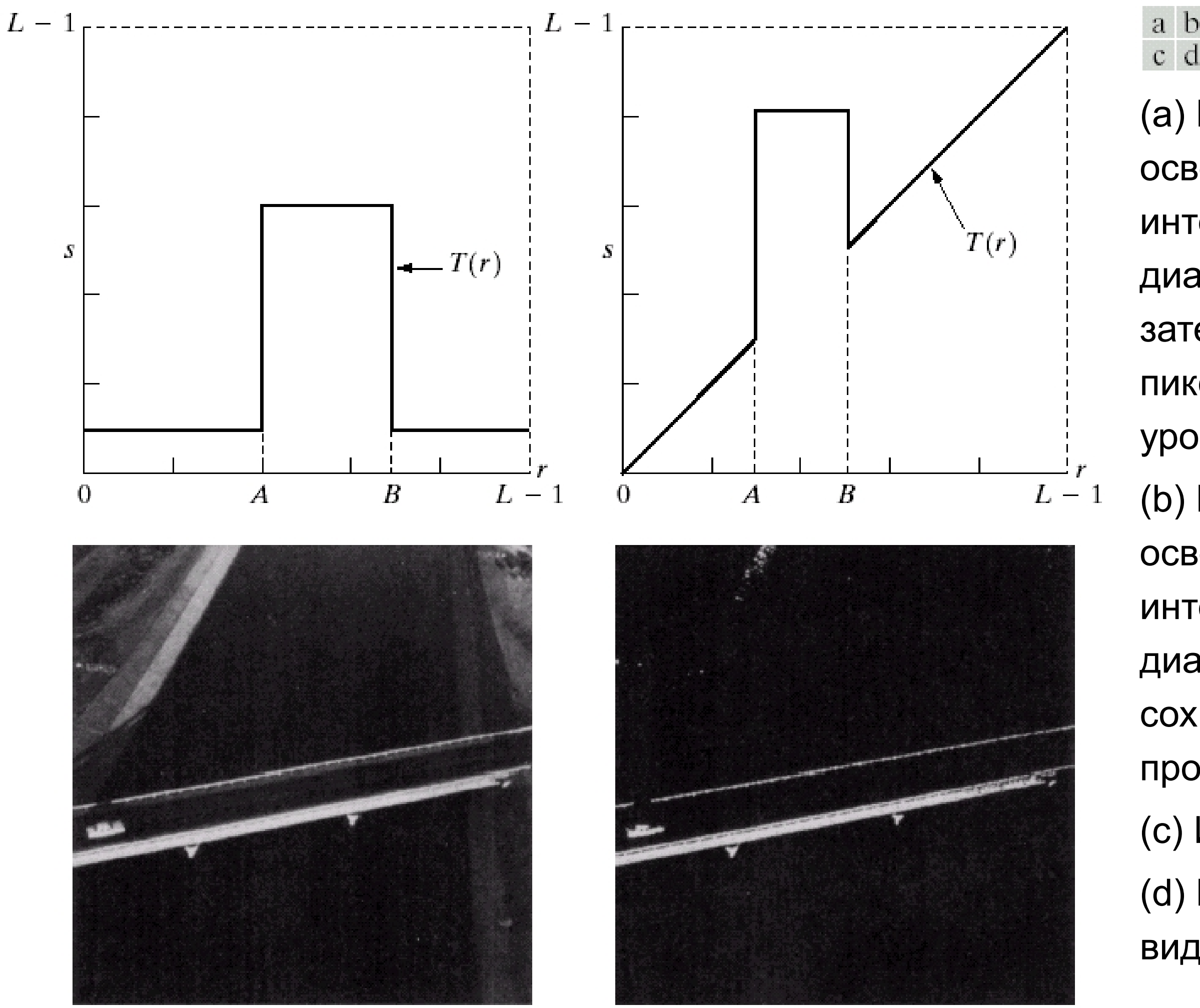
(a) Вид преобразования

(b) Неконтрастное изображение

(c) Результат контрастирования

(d) Результат пороговой бинаризации

Выделение поддиапазонов серого



(a) Преобразование, осветляющее пиксели с интенсивностью из диапазона $[A,B]$ и затемняющее все прочие пиксели до константного уровня

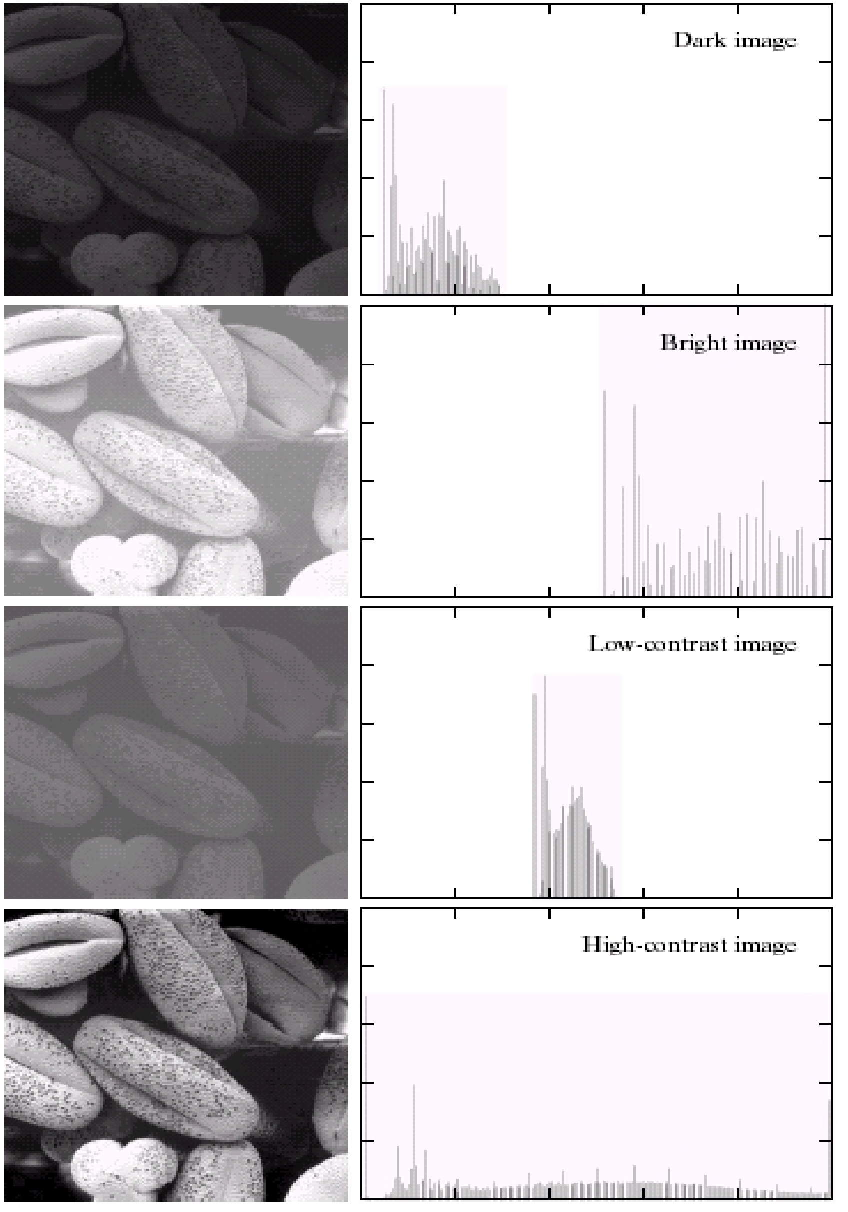
(b) Преобразование, осветляющее пиксели с интенсивностью из диапазона $[A,B]$ и сохраняющее значения прочих пикселей

(c) Исходное изображение

(d) Результат преобразования вида *(a)*

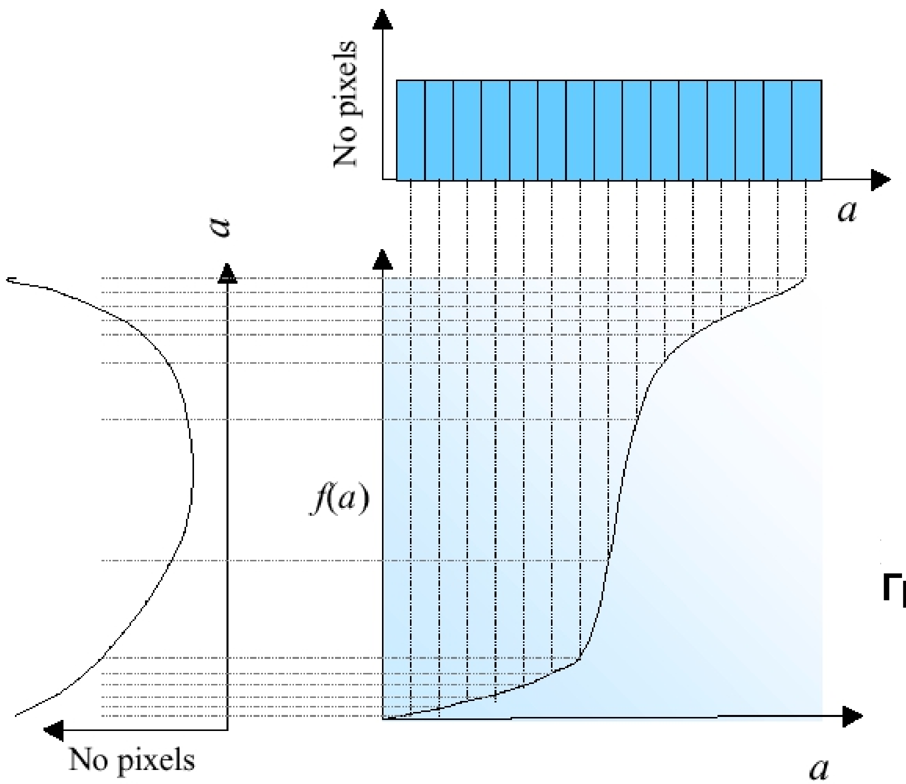
Преобразование гистограмм

- Попиксельный метод улучшения изображения зависит от данных.
- Гистограмма ≈ функция плотности распределения значений пикселей.
  - Предположение: значения пикселей изображения – одинаково распределенные случайные величины
  - На рисунках приведены примеры изображений и соответствующие им гистограммы



Преобразование гистограмм

Под влиянием попиксельного преобразования изменяется гистограмма изображения



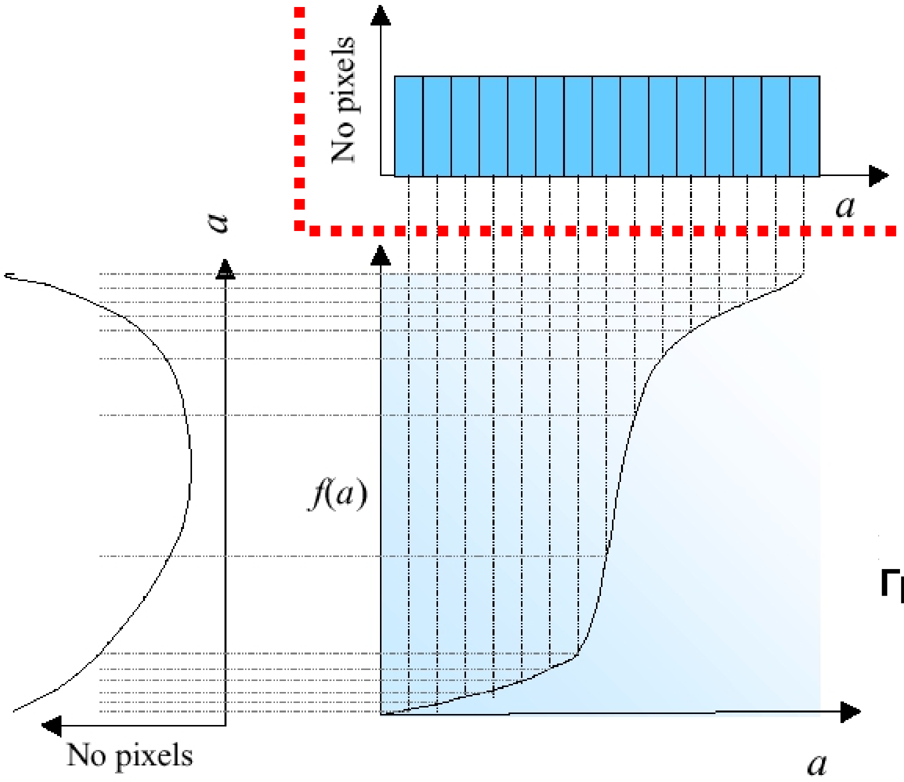
Исходная гистограмма

График преобразования *f*

Преобразованная гистограмма (лежит на боку)

Преобразование гистограмм

Под влиянием попиксельного преобразования изменяется гистограмма изображения



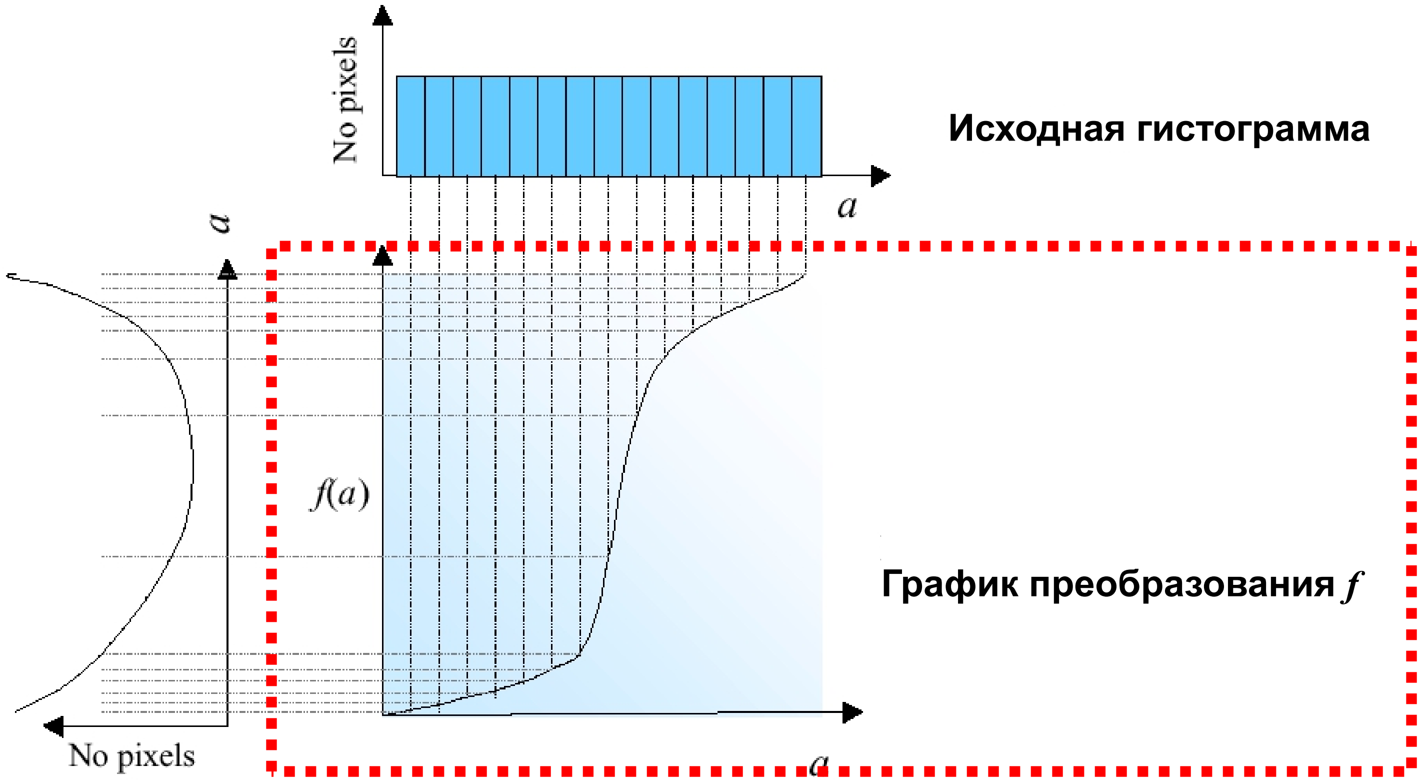
Исходная гистограмма

График преобразования *f*

Преобразованная гистограмма (лежит на боку)

Преобразование гистограмм

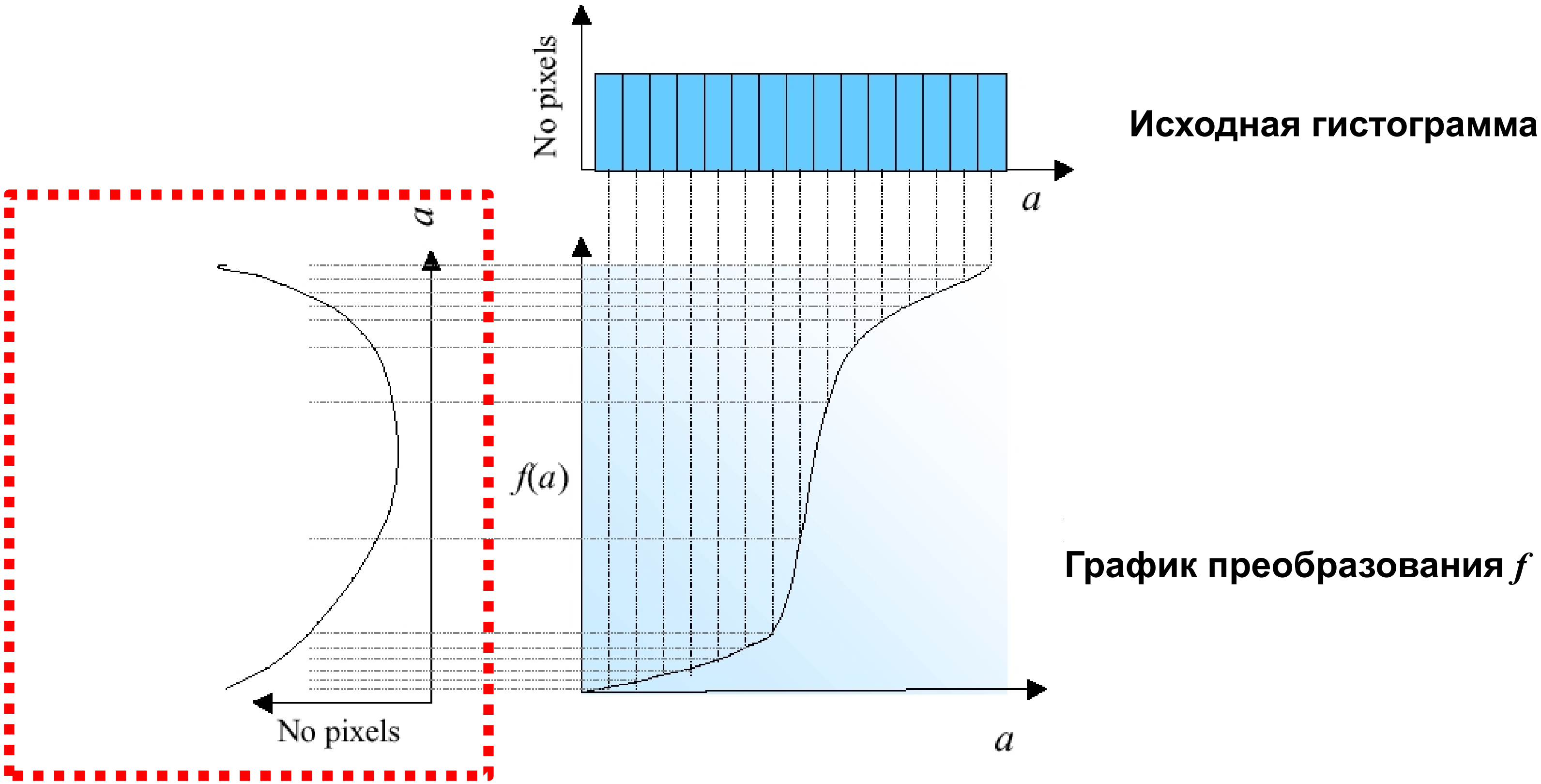
Под влиянием попиксельного преобразования изменяется гистограмма изображения



Преобразованная гистограмма (лежит на боку)

Преобразование гистограмм

Под влиянием попиксельного преобразования изменяется гистограмма изображения

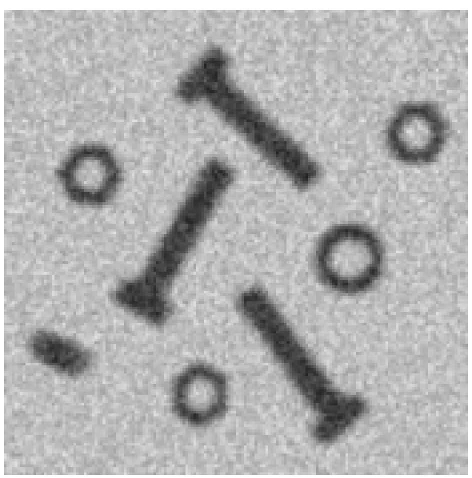


Преобразованная гистограмма (лежит на боку)

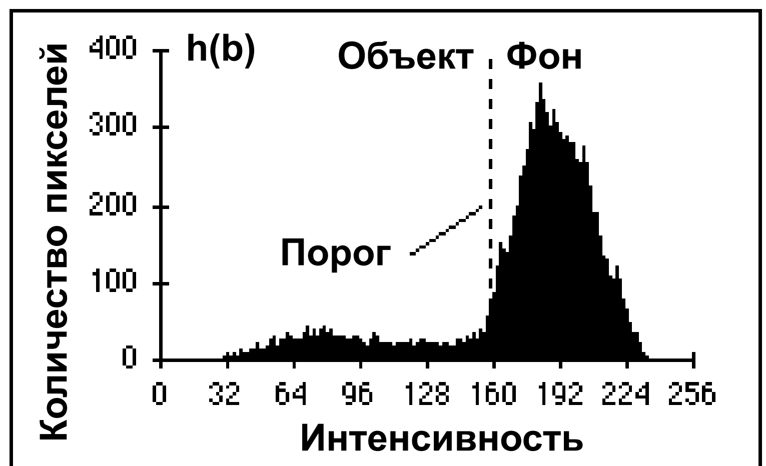
Пороговая бинаризация

- Пиксели с интенсивностью ниже порогового значения помечаются, как принадлежащие объекту, остальные пиксели – как принадлежащие фону.
- Здесь и далее рассматриваются изображения, содержащие темные объекты на светлом фоне.

Изображение

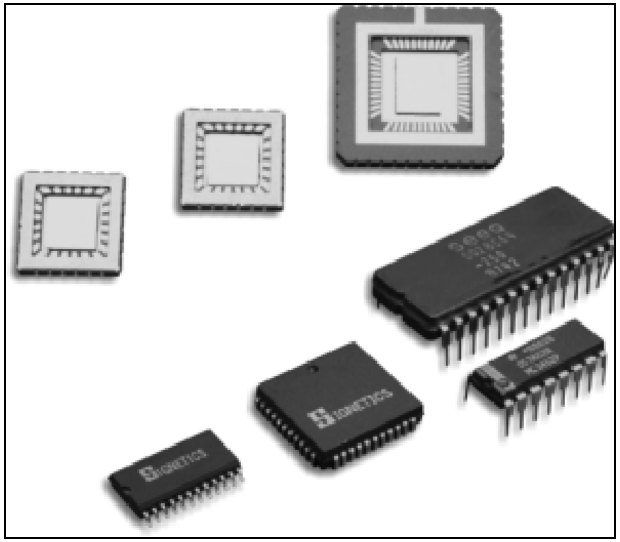


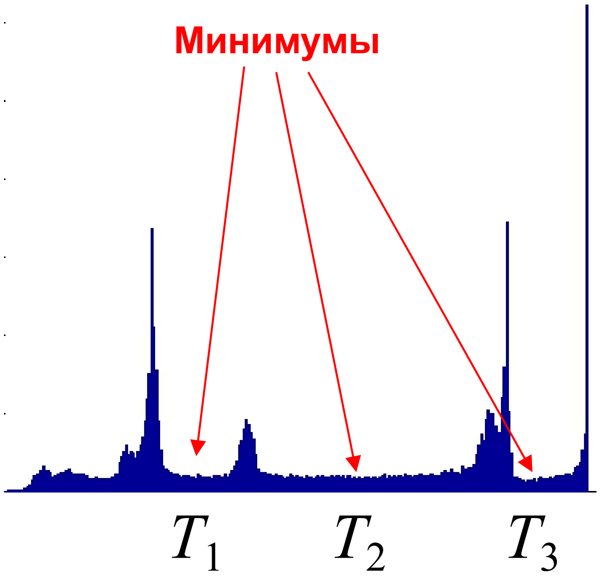
Гистограмма



Пороговая бинаризация

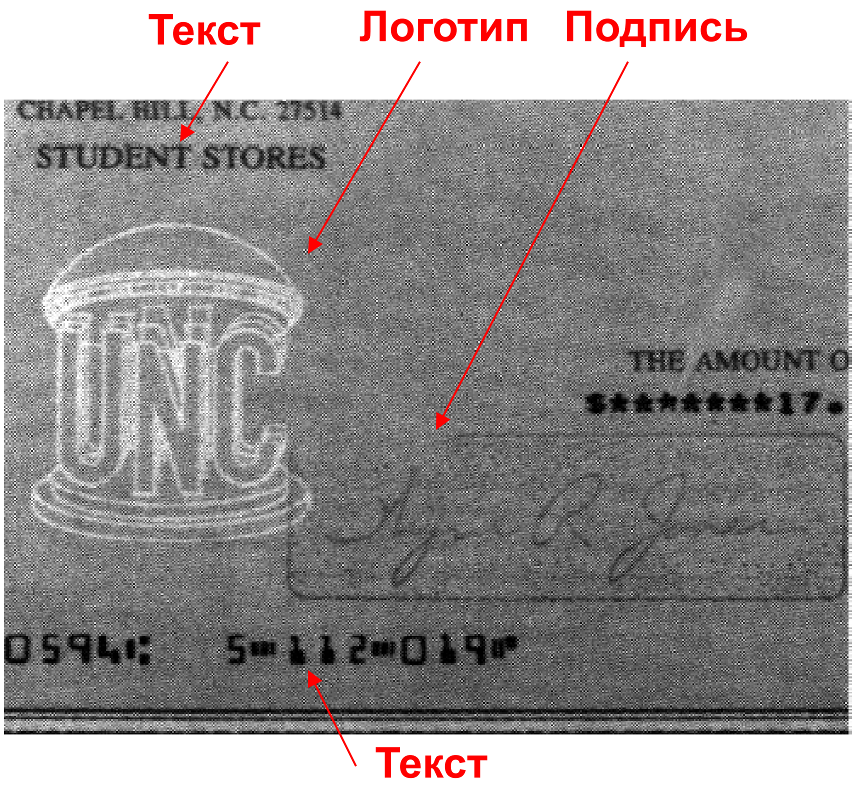
Возможно обобщение на случай многопороговой сегментации

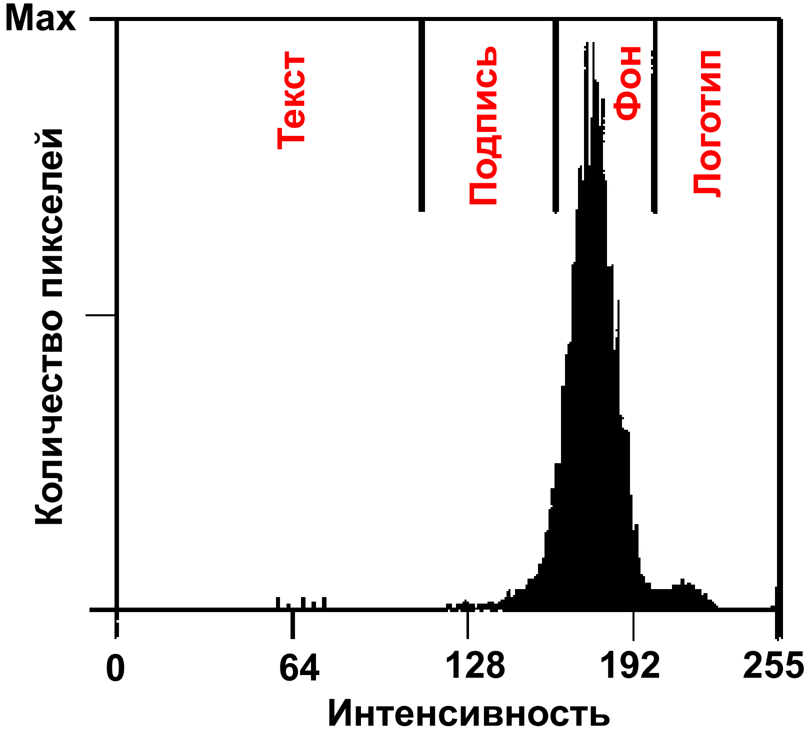




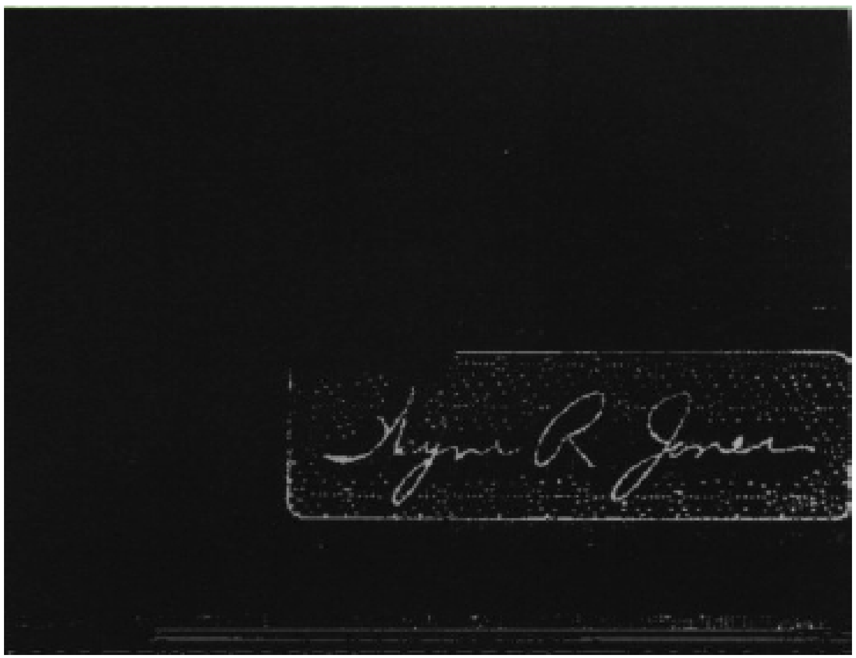
$$
\begin{array}{ll}
I(x,y) < T_1, & I(x,y) = 1 \ \text{(объект 1)} \\
T_1 < I(x,y) < T_2\ , & I(x,y) = 2 \ \text{(объект 2)} \\
T_2 < I(x,y), & I(x,y) = 3 \ \text{(фон)}
\end{array}
$$

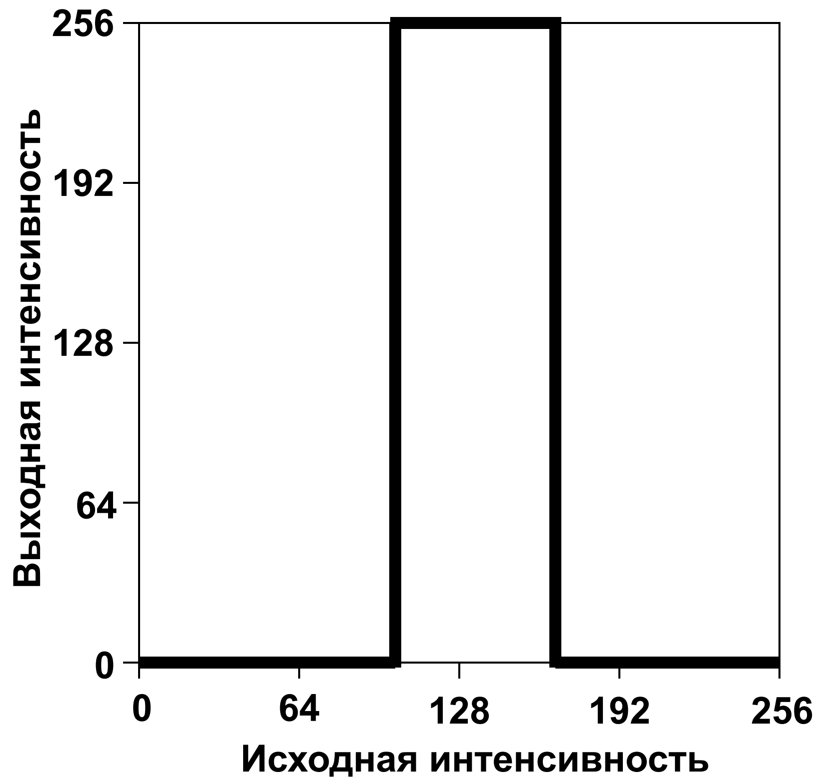
Пороговая сегментация для многопиковых гистограмм



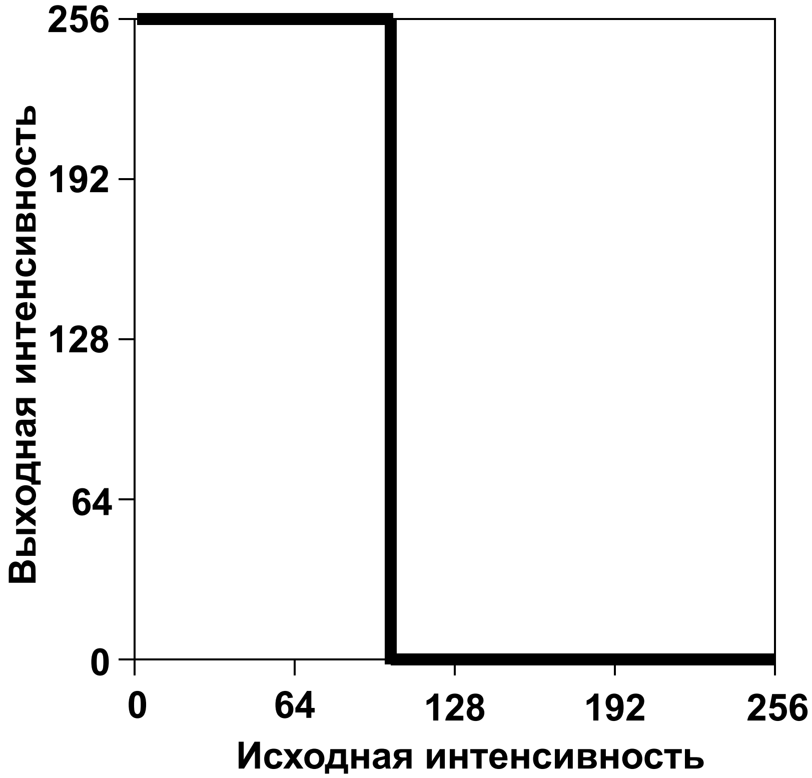


Препарирование изображения и пороговая сегментация

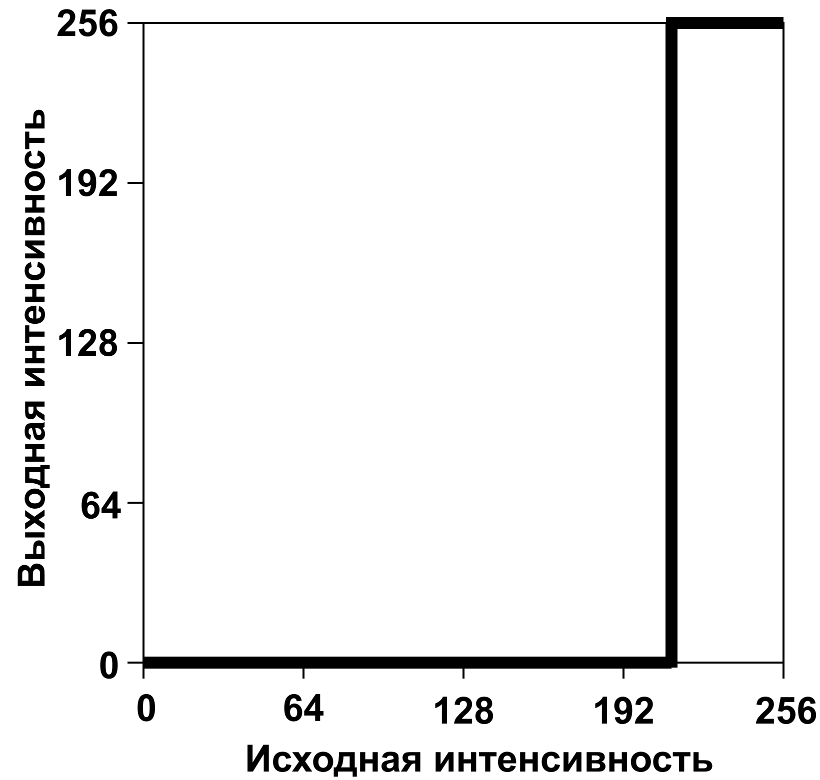




Препарирование изображения и пороговая сегментация



Препарирование изображения и пороговая сегментация



Фильтрация изображений

Что такое фильтрация изображений?

- Фильтрация изображений – это процесс вычленения сигнала или структуры изображения путем подавления нежелательных или неинтересующих вариаций на изображении.

- Главный механизм – изменение значений пикселей на изображении с помощью некоторой функции от значений пикселей в локальный окрестности.

Фрагмент изображения

Преобразованный фрагмент изображения

Что такое фильтрация изображений?

- Простейший случай: **линейная фильтрация** – замена каждого пикселя линейной комбинацией его соседей.
- Коэффициенты линейной комбинации называют «маской фильтра» («ядром фильтра», «ядром свертки»).

10	5	3
4	5	1
1	1	7

Фрагмент изображения

0	0	0
0	0.5	0
0	1	0.5

Ядро фильтра

	7	

Преобразованный фрагмент изображения

Что такое фильтрация изображений?

- Простейший случай: **линейная фильтрация** – замена каждого пикселя линейной комбинацией его соседей.
- Коэффициенты линейной комбинации называют «маской фильтра» («ядром фильтра», «ядром свертки»).

10	5	3
4	5	1
1	1	7

Фрагмент изображения

0	0	0
0	0.5	0
0	1	0.5

Ядро фильтра

	7	

Преобразованный фрагмент изображения

Свертка

- Действие фильтра g на сигнал f можно записать в виде свертки $f \otimes g$.
- В случае непрерывных сигналов f и g:

$$h(x) = f(x) \quad g(x) = \mathop{\ }\limits_{-}^{+} f(x-t)\, g(t)\, dt$$

- В дискретном случае f и g – сигналы, представленные конечным числом отсчетов: $f = \left\{ f_k \right\}_{k=K_0}^{K}, g = \left\{ g_l \right\}_{l=L_0}^{L}$

$$h_k = f \quad g = \mathop{\ }\limits_{l=L_0}^{L} f_{k-l}\, g_l$$

- Содержательный смысл: осреднение f с весами g.

Свертка для изображений

- Роль конечного дискретного двумерного сигнала f играет изображение I.

- Ядро двумерного линейного фильтра g с конечным носителем представляет собой матрицу.

- Результат фильтрации – результат двумерной свертки:

$$I^*[i,j]=I \quad g= \underset{k,l}{} I[i-k,j-l]g[k,l]$$

Виды фильтров

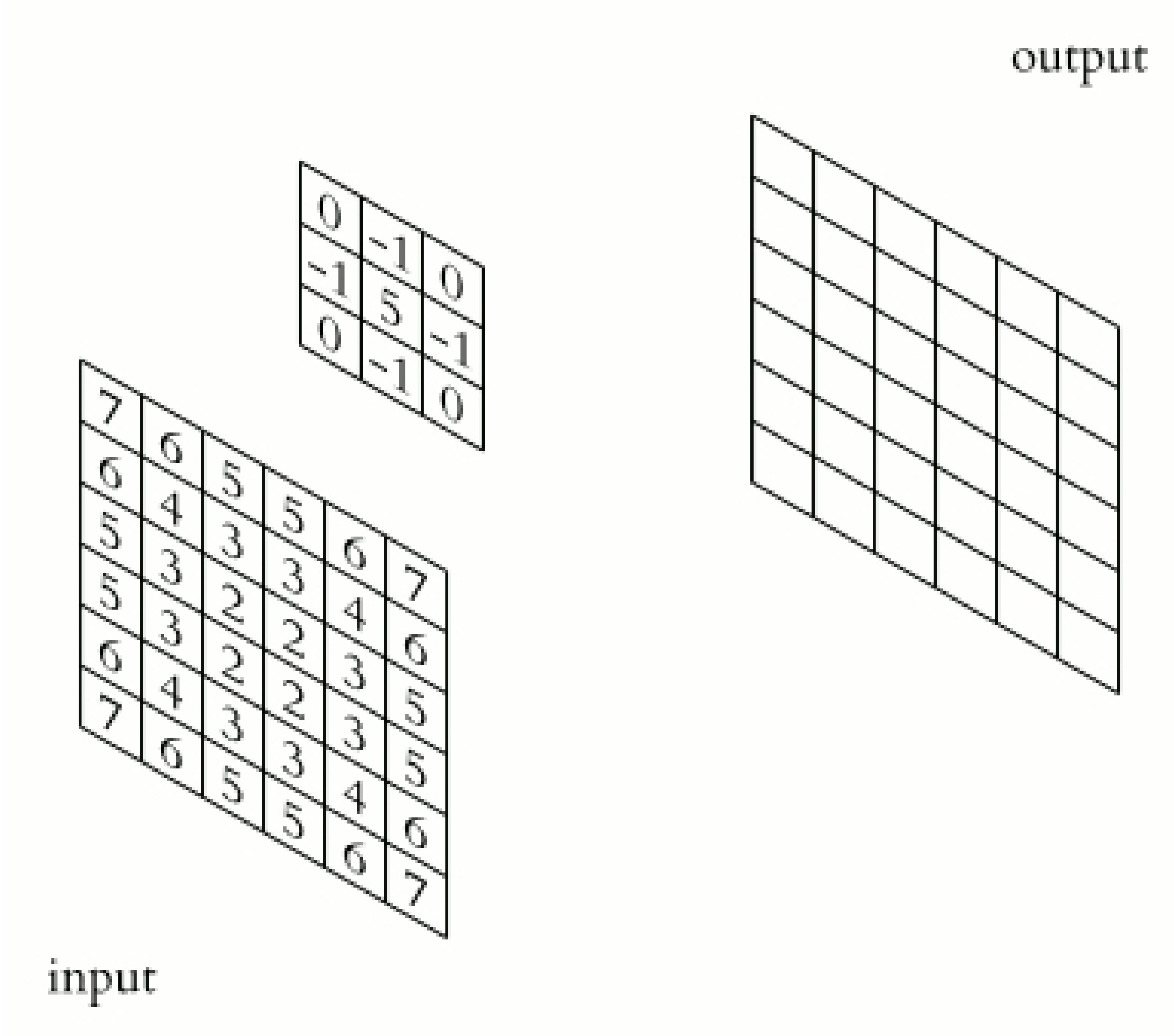
Фильтры подразделяют

- По виду функции, связывающей окрестность пикселя с откликом фильтра, на:
  - Линейные
  - Нелинейные
- По способу записи отклика фильтра в исходное или результирующее изображение на:
  - Рекурсивные
  - Нерекурсивные
- По независимости фильтрующей функции от координат на изображении на:
  - Стационарные
  - Нестационарные

Схемы перемещения маски фильтра

- **P–схема**: ни один элемент маски не выходит за пределы изображения.
- **S–схема**: центральный элемент маски не может выходить за пределы изображения (остальные элементы окна могут)
  - –Усекается окно
  - –Расширяется изображение
- **T–схема**: изображение «сворачивается» в тор так, чтобы его правый край примыкал к левому со сдвигом в один пиксел (конец второй строки к началу первой), а верхний край – к нижнему (последний столбец к первому).

Схемы перемещения маски фильтра



Фильтрация и зашумленные изображения

- Часто изображение содержит шум, затрудняющий дальнейшую обработку
  - Речь идет о **простом** шуме: флуктуации интенсивности; помехи, вносимые камерой; нежелательные эффекты квантования; эффекты, обусловленные конечной точностью компьютера.
  - Речь не идет о сложных эффектах: тени; посторонние объекты и т.п.

Распространенные типы шумов

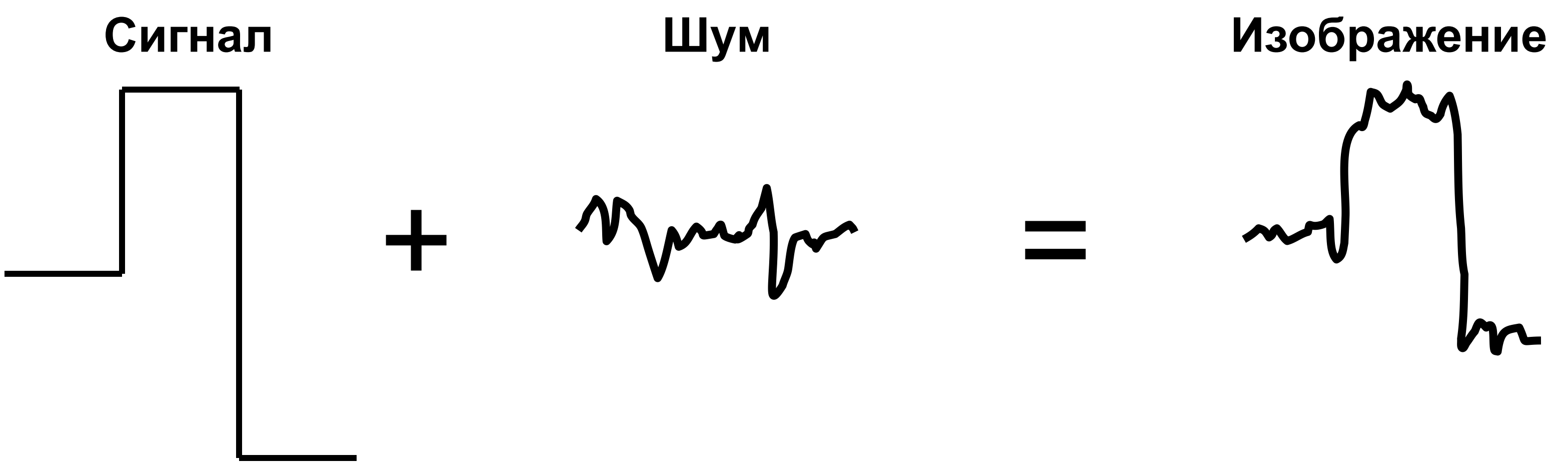
- Импульсный шум («соль и перец»)
  - Единичные светлые пикселы в темных областях изображения и темных пикселы в светлых областях изображения.
  - Обычно является результатом ошибочной классификации в виду вариации свойств объектов или освещения.

Распространенные типы шумов

- Гауссов шум
  - Нормально распределенные отклонения интенсивности.
  - Является результатом неточных измерений (или квантования) интенсивности.
- Белый шум
  - Равномерно распределенные отклонения интенсивности.
  - Является результатом неточных измерений (или квантования) интенсивности.

Модель шума

- Предполагается, что шум воздействует на сигнал аддитивно: $I = S + N$ (изображение = сигнал + шум).
- Сигнал — детерминированная величина
- Шум – случайная величина (с нулевым математическим ожиданием).
- Шум не зависит от сигнала.



Шумоподавляющие фильтры

- Соседи пикселя содержат информацию о его интенсивности.
- Более близкие точки содержат больше информации о сигнале, чем более удаленные.
- Усредняющие фильтры подавляют шумовые эффекты.

Усредняющий фильтр

- Ядро – матрица из положительных весов, в сумме дающих 1.
- Заменяет каждый пиксел средним значением интенсивности в его окрестности.
- Если все веса равны, то фильтр называется box–фильтром.

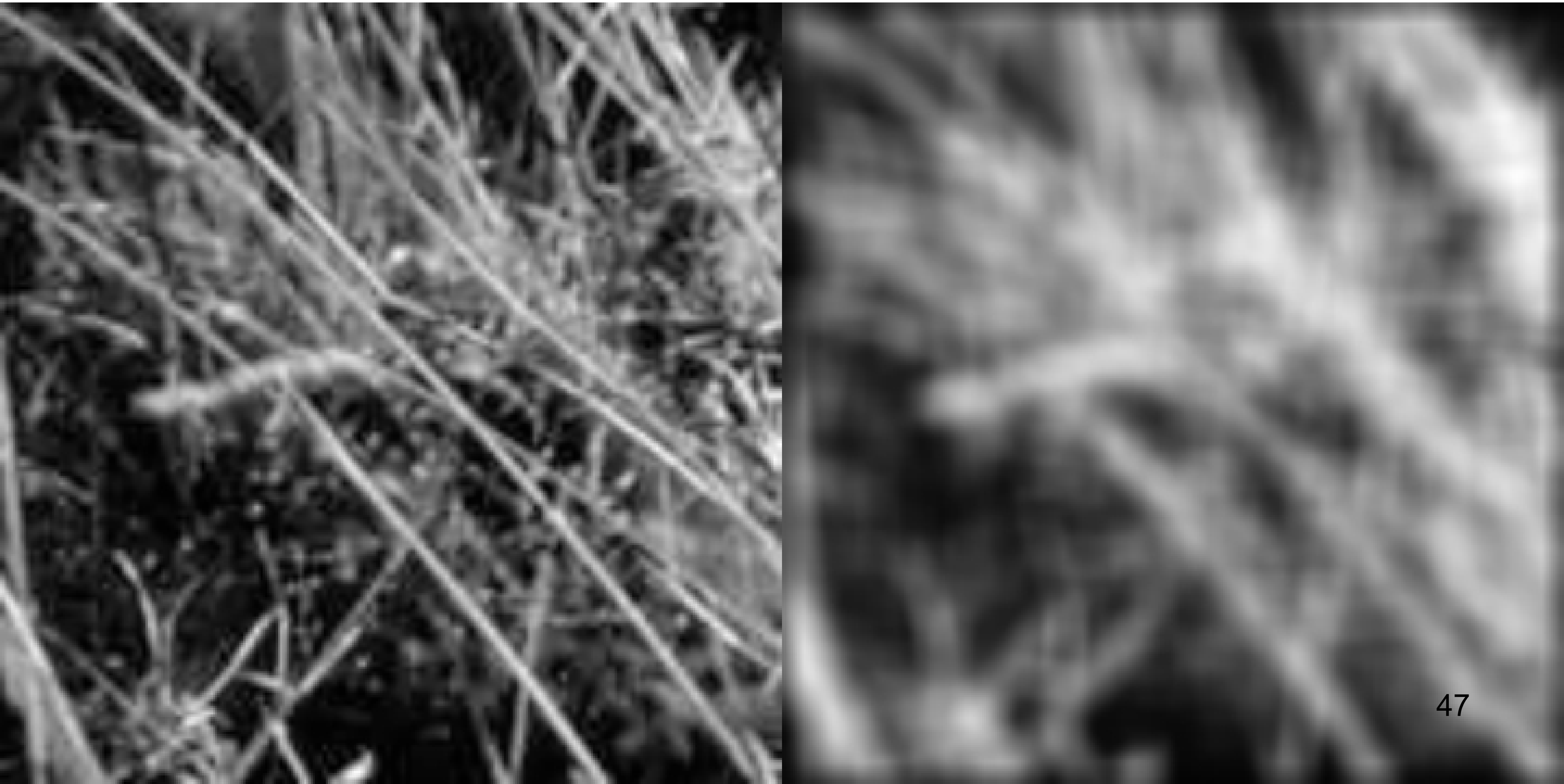
$$g$$

$$1/9 \cdot \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

Свойства сглаживающих масок

- Элементы сглаживающих масок положительны, их сумма равна единице, поскольку значение на выходе должно совпадать со значением на входе для областей постоянной интенсивности.
- Степень сглаживания и подавления шума пропорциональна размеру маски.
- Резкие перепады интенсивности сглаживаются пропорцинально размеру маски.

Пример: сглаживание усреднением

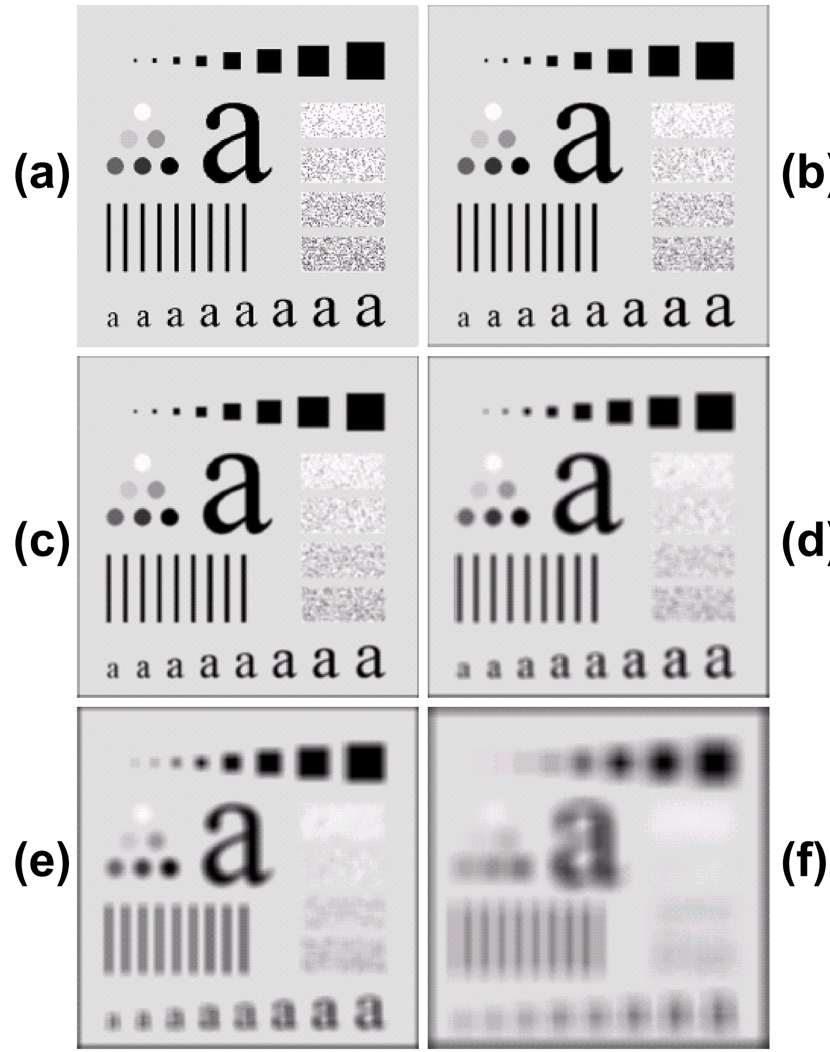


Пример: подавление шума усреднением

Использован box–фильтр с маской 5x5

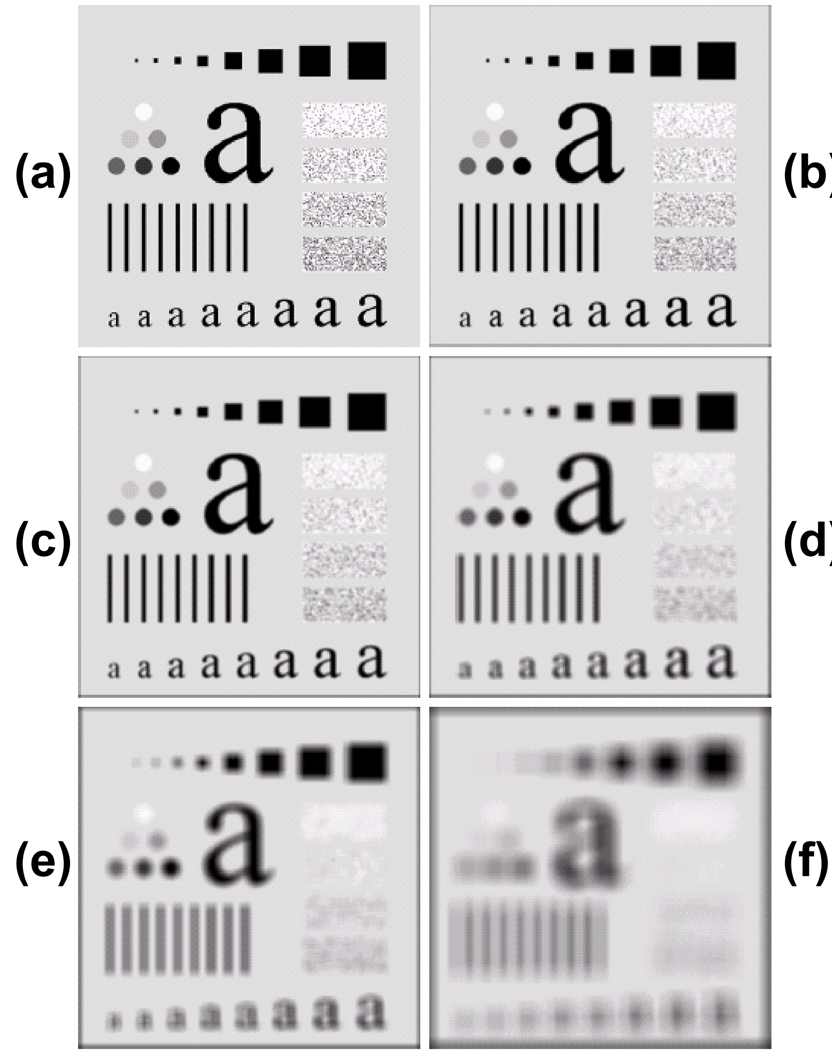
Пример: сглаживание усредняющими фильтрами

(a) Исходное изображение размера 500x500.
(b) – (f) Результаты сглаживания усредняющим фильтром с масками nxn, n=3, 5, 9, 15 и 35 соответственно. Длины сторон черных квадратов в верхней части равны 3, 5, 9, 15, 35, 45 и 55 пикселей соответственно; расстояние между квадратами равно 25 пикселям.



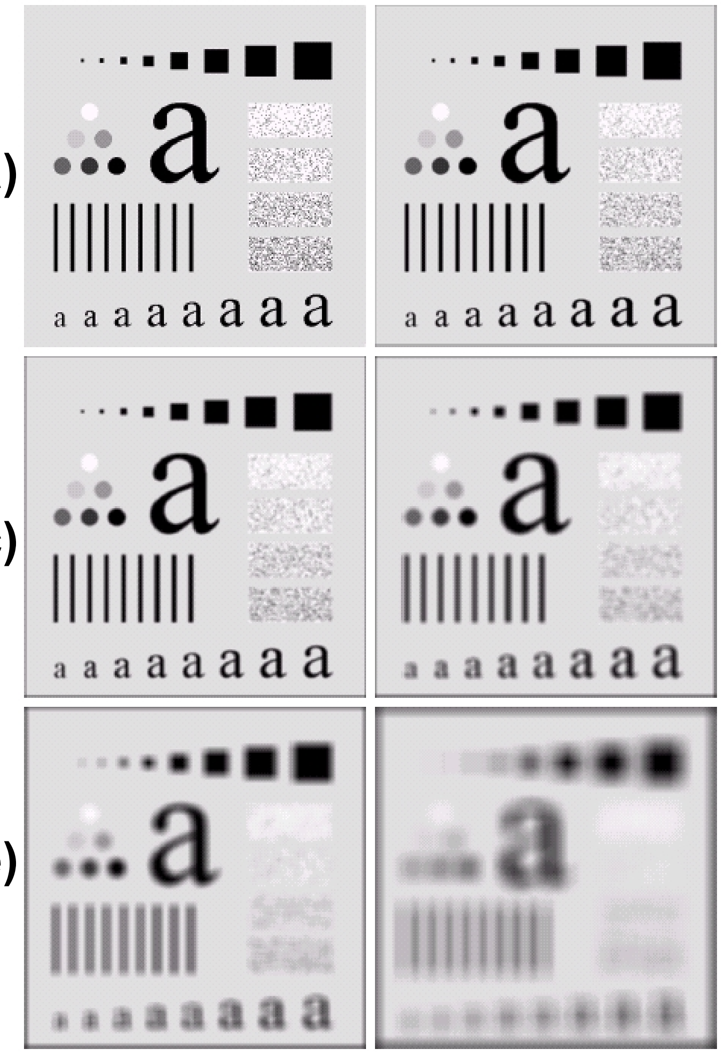
Пример: сглаживание усредняющими фильтрами

Буквы в нижней части набраны шрифтами от 10 до 24 пунктов с шагом в 2 пункта; большая буква наверху имеет размер 60 пунктов. Вертикальные прямоугольники имеют размеры 5x100 пикселей, расстояние между ними –20 пикселей.



Пример: сглаживание усредняющими фильтрами

Диаметр кругов – 25 пикселей, расстояние между ними – 15 пикселей; интенсивность кругов меняется от 0% до 100% с шагом в 20%. Интенсивность фона изображения – 10%. Зашумленные прямоугольники имеют размеры 50x120 пикселей.



(a) (b) (c) (d) (e) (f)

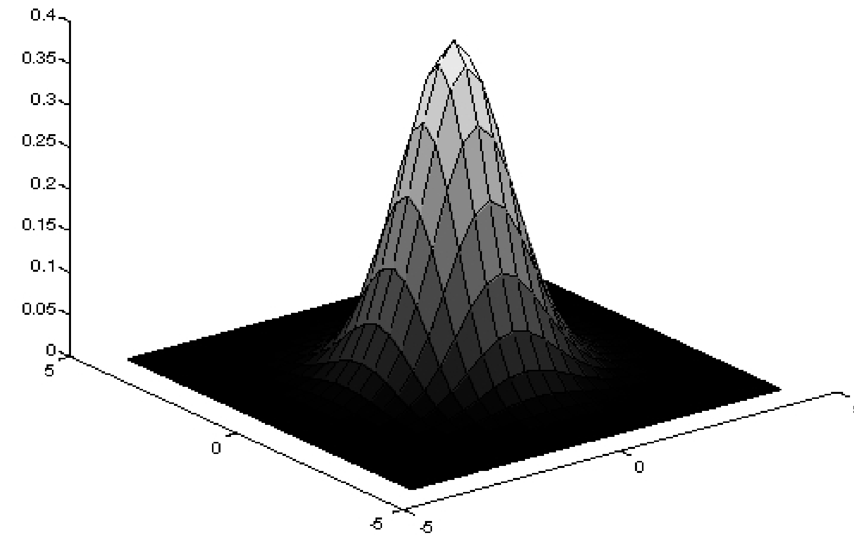
Сглаживание и сигнал

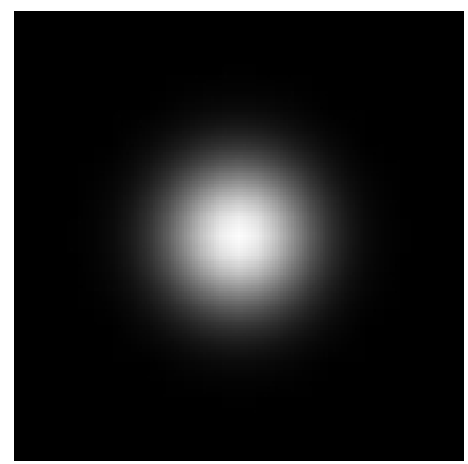
- Помимо подавления шума сглаживание делает более гладким и сигнал тоже.
- Сглаживание удаляет детали.
- Плюсы и минусы:
  – Минус: нельзя удалить шум без размывания форм объектов (границ).
  – Плюс: выделяется крупномасштабная структура изображения.

Гауссово усреднение

- Веса ближних пикселей больше весов дальних.
- Ядро симметрично относительно начала координат.
- Ядро пропорционально функции

$$g(x, y) = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$



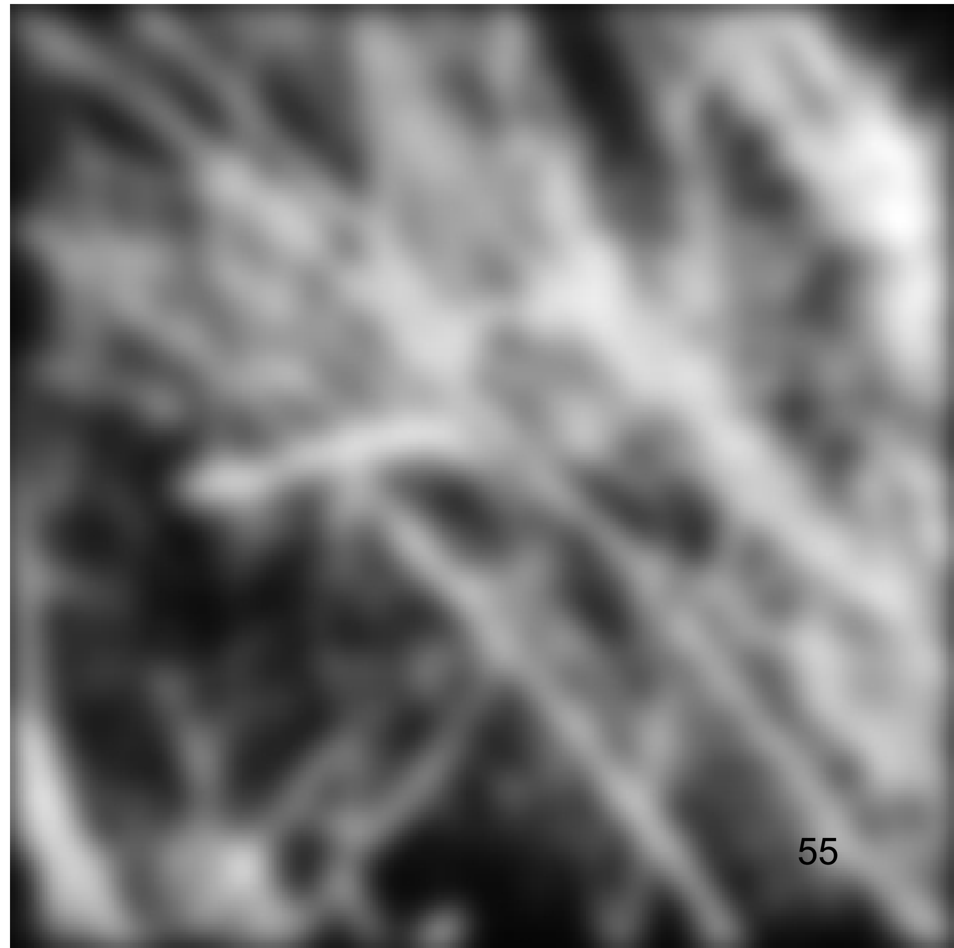


Примеры масок гауссовых фильтров

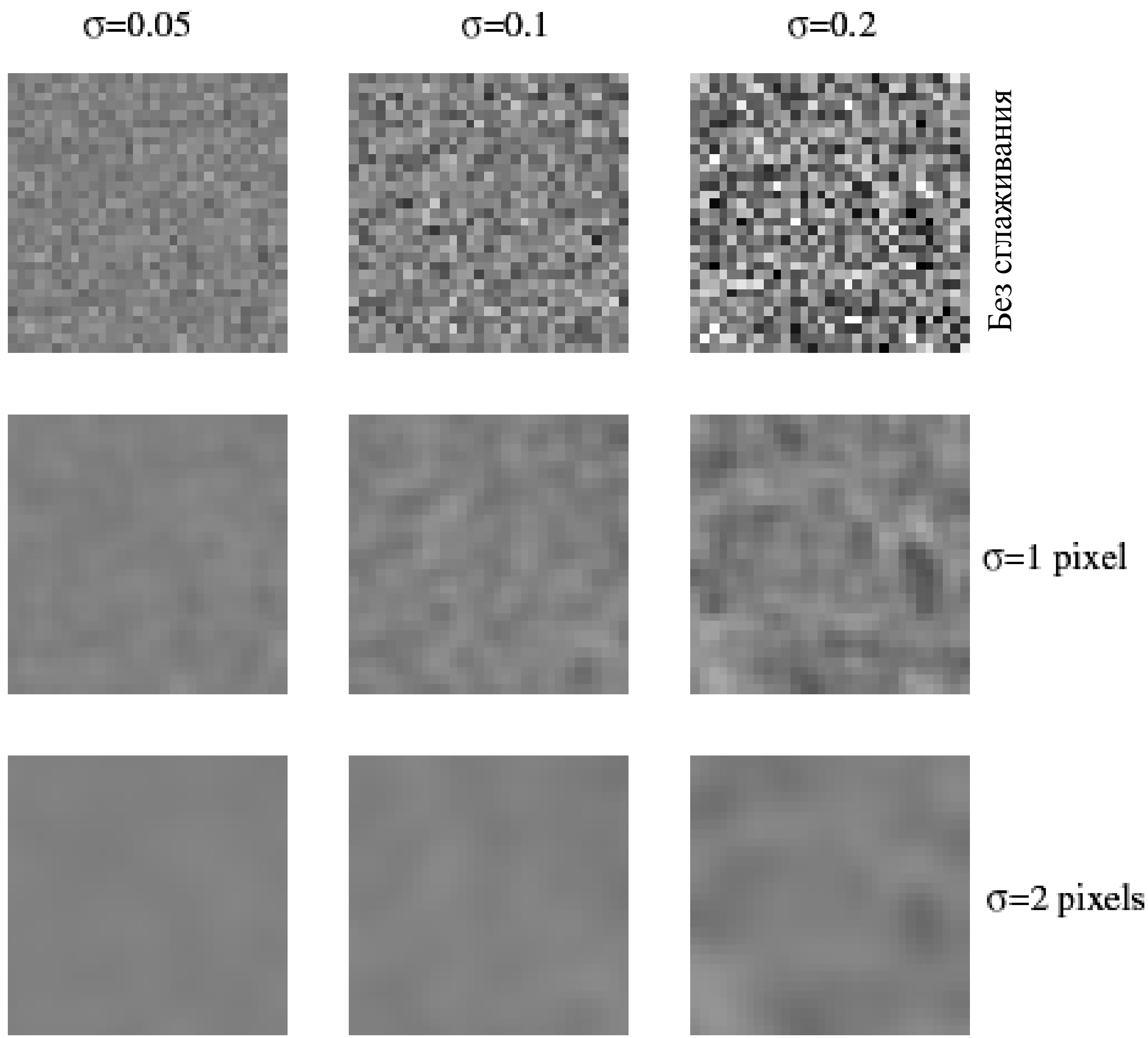
$$g_{3x3} = 1/16 \cdot \begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

$$g_{7x7} = 1/1098 \cdot \begin{array}{|c|c|c|c|c|c|c|}\hline 1 & 3 & 7 & 9 & 7 & 3 & 1 \\ \hline 3 & 12 & 26 & 33 & 26 & 12 & 3 \\ \hline 7 & 26 & 55 & 70 & 55 & 26 & 7 \\ \hline 9 & 33 & 70 & 90 & 70 & 33 & 9 \\ \hline 7 & 26 & 55 & 70 & 55 & 26 & 7 \\ \hline 3 & 12 & 26 & 33 & 26 & 12 & 3 \\ \hline 1 & 3 & 7 & 9 & 7 & 3 & 1 \\ \hline \end{array}$$

Пример: сглаживание гауссовым усреднением



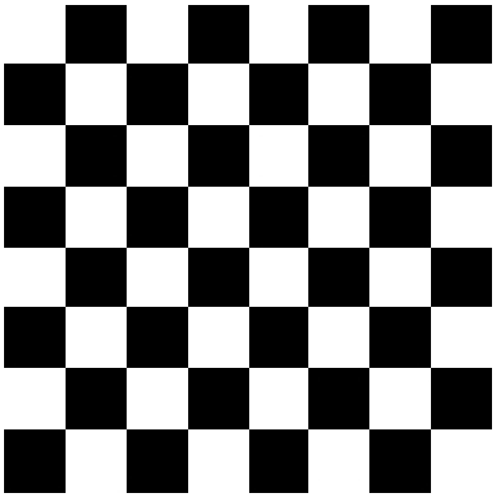
Эффекты сглаживания

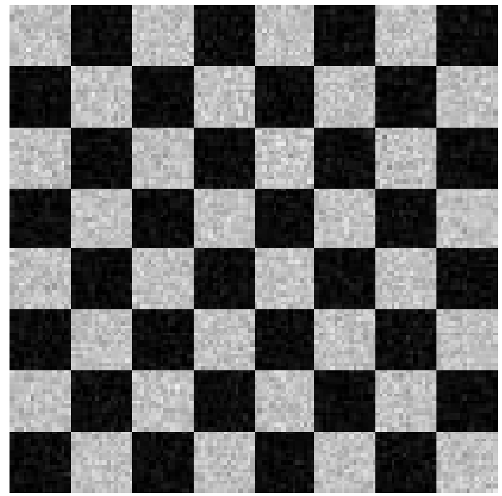


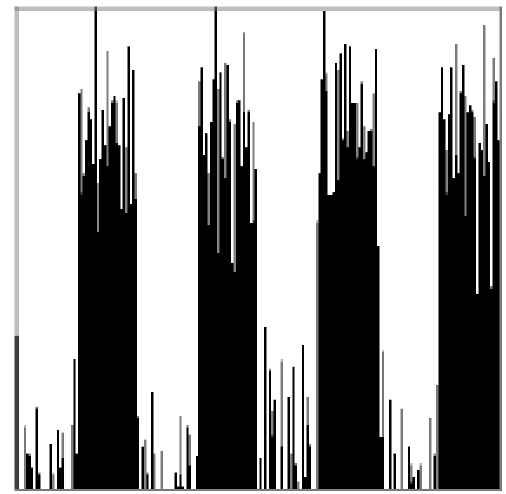
Изображения в каждой строке сглажены гауссовым фильтром с одним и тем же размером ядра.

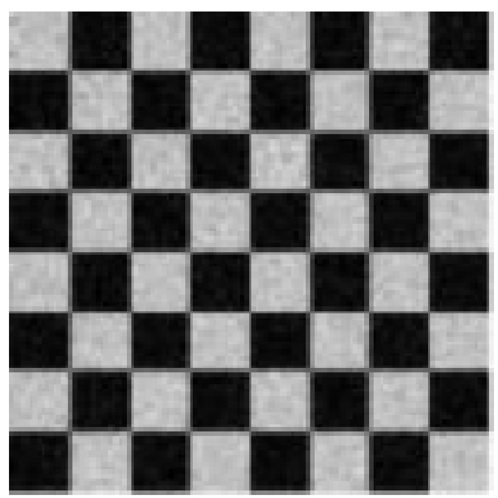
Изображения в каждом столбце являются реализациями гауссова шума с одной и той же дисперсией

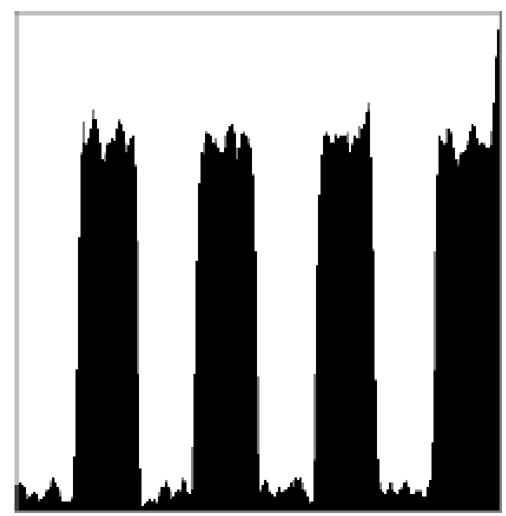
Изображение с гауссовым шумом











Пример: подавление шума гауссовым сглаживанием

Эффективная реализация линейных фильтров

- Большинство фильтров являются сепарабельными, в том числе box–фильтр и гауссов фильтр:

$$\begin{matrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{matrix} = \begin{matrix} 1/3 \\ 1/3 \\ 1/3 \end{matrix} \begin{pmatrix} 1/3 & 1/3 & 1/3 \end{pmatrix}$$

$$\begin{matrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{matrix} = \begin{matrix} 1 \\ 2 \\ 1 \end{matrix} \begin{pmatrix} 1 & 2 & 1 \end{pmatrix}$$

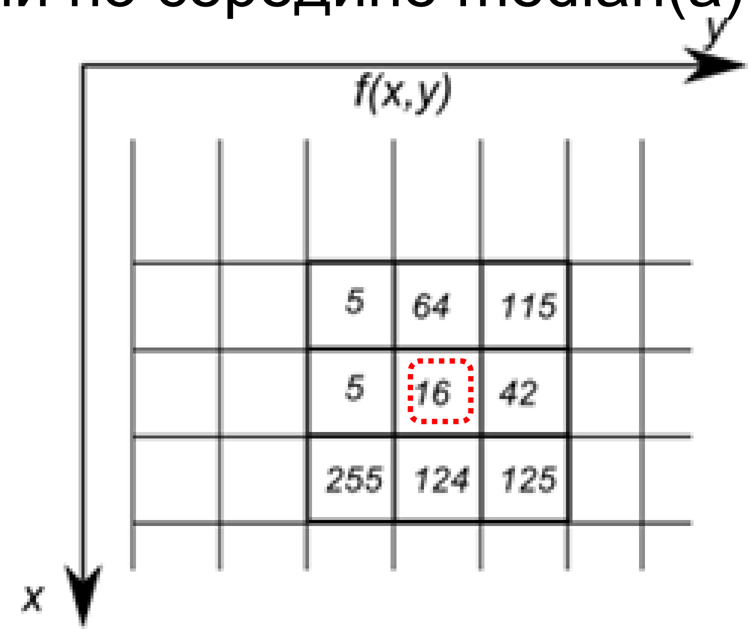
- Таким образом, фильтрация может быть осуществлена в два этапа:
  - Одномерная фильтрация столбцов;
  - Одномерная фильтрация строк.

Медианный фильтр

Из исходного изображения выделим область размером 3х3 с 9 значениями яркостей пикселей. Получим упорядоченную по возрастанию последовательность

a= [5, 5, 16, 42, 64, 115, 124, 125, 255],

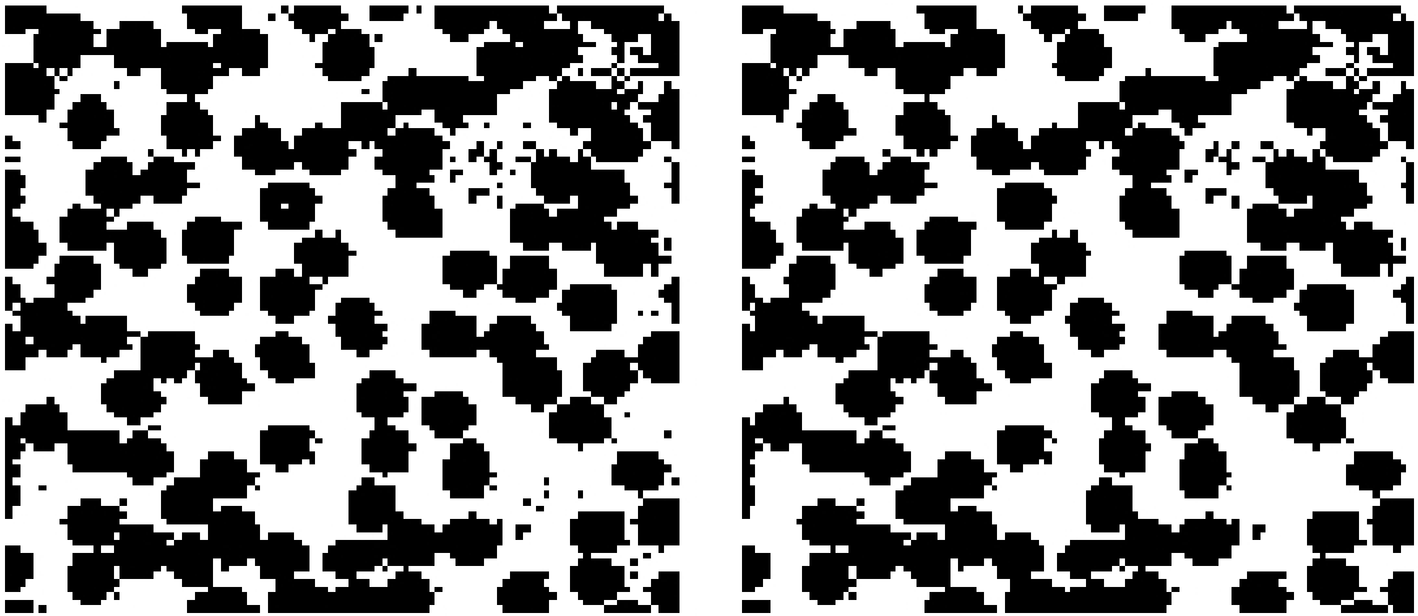
выберем из данной последовательности элемент, расположенный по середине median(a)=64.



Медианный фильтр

- Усредняющий фильтр обычно работает, если области изображения более или менее однородны.
- **Медианный фильтр** заменяет центральный пиксел в маске медианой пикселей из некоторой окрестности.
  - Работает для многих типов шума.
  - Сохраняет границы в отличие от усредняющего фильтра.
  - Вычислительно сложен (из-за сортировки).
- Медианный фильтр – нелинейный!

Медианный фильтр



1	1	1
1	0	1
1	1	1

⇒

1	1	1
1	1	1
1	1	1

;

0	0	0
0	1	0
0	0	0

⇒

0	0	0
0	0	0
0	0	0

X	X	X
X	L	X
X	X	X

⇒

X	X	X
X	X	X
X	X	X

;

	X	
X	L	X
	X	

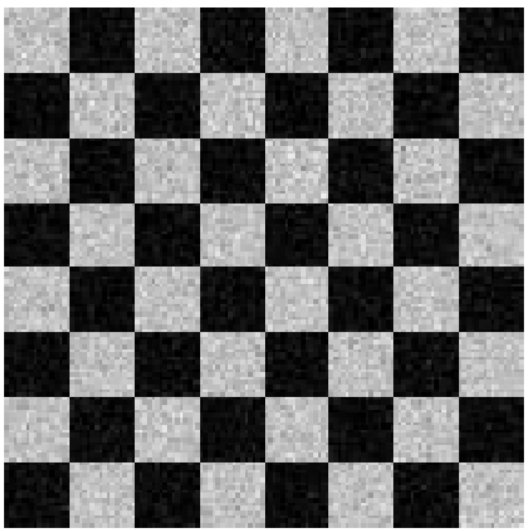
⇒

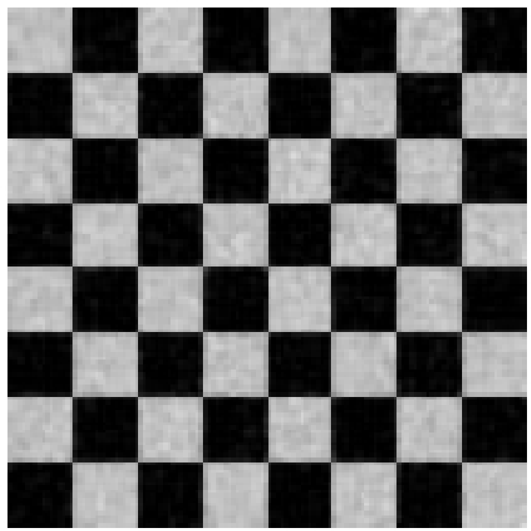
	X	
X	X	X
	X	

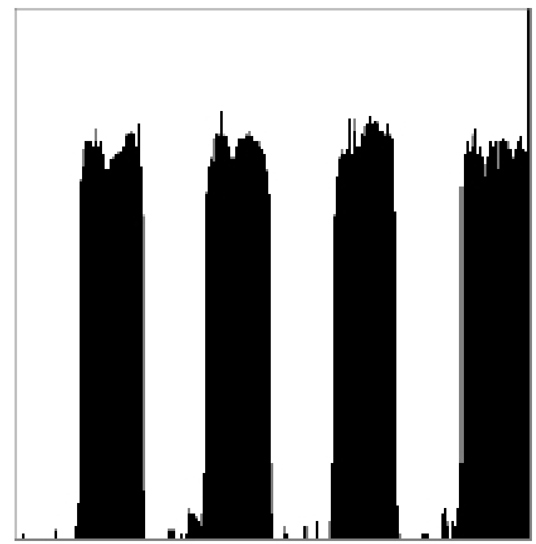
Пример: устранение импульсного шума медианной фильтрацией

Пример: устранение гауссова шума медианной фильтрацией

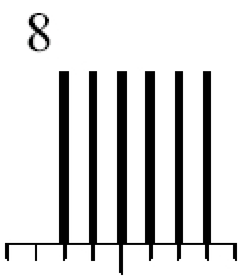
Использован медианный фильтр с маской 5x5



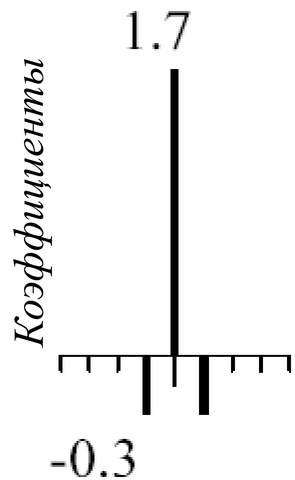




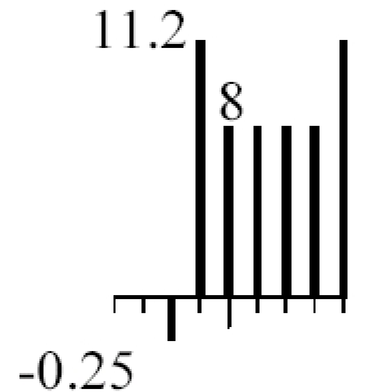
Пример действия фильтра увеличения резкости



Исходное изображение

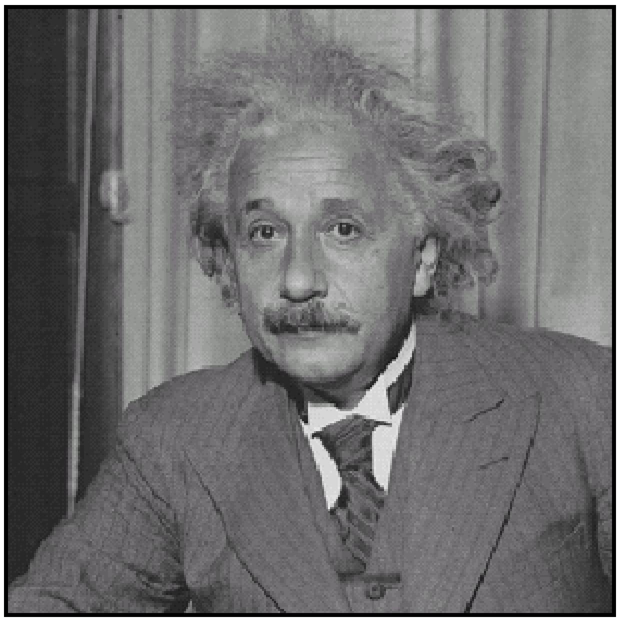


Ядро фильтра

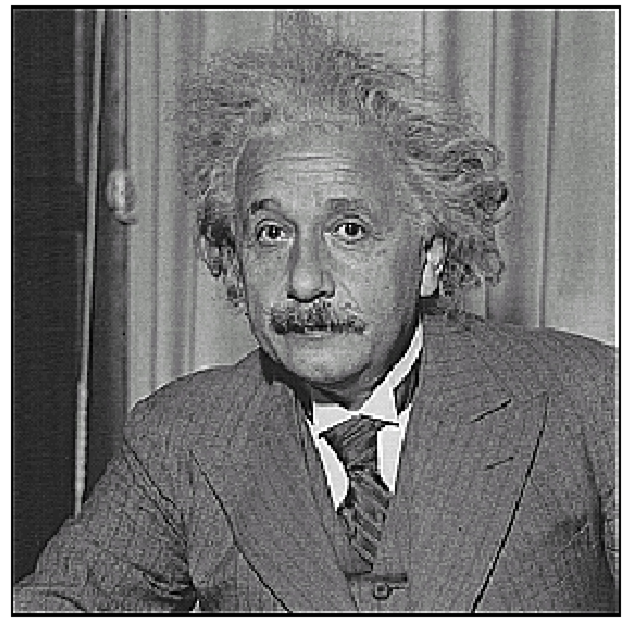


Результирующее изображение (перепады подчеркнуты; области постоянной интенсивности неизменны)

Пример действия фильтра увеличения резкости



Исходное изображение



Результирующее изображение

Линейные фильтры, подчеркивающие линии определенного направления

$$\frac{1}{16}\cdot\begin{bmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{bmatrix}$$

Подчеркиваются четырехсвязные элементы исходного изображения, т.е. горизонтальные и вертикальные линии.

$$\frac{1}{16}\cdot\begin{bmatrix}2 & 1 & 2\\ 1 & 4 & 1\\ 2 & 1 & 2\end{bmatrix}$$

Подчеркиваются восьмисвязные элементы, не являющиеся четырехсвязными элементами исходного изображения.

Линейные фильтры, подчеркивающие контуры

Общий вид (A ≥ 1)

0	–1	0
–1	A+4	–1
0	–1	0

–1	–1	–1
–1	A+8	–1
–1	–1	–1

Примеры

0	–1	0
–1	5	–1
0	–1	0

–1	–1	–1
–1	9	–1
–1	–1	–1

- Подчеркиваются контуры независимо от их направления.
- Сумма весов фильтра равна **единице**.

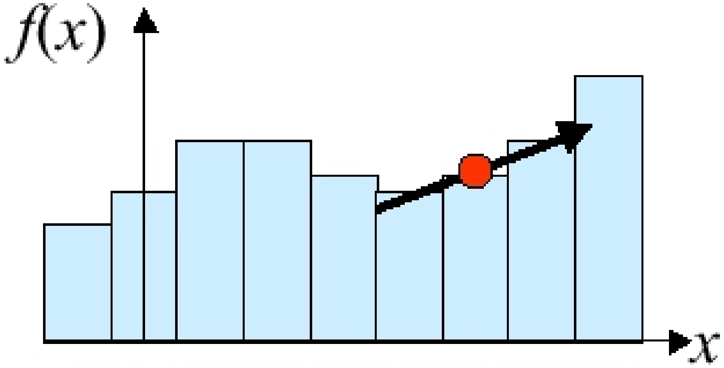
Линейные фильтры, выделяющие контуры

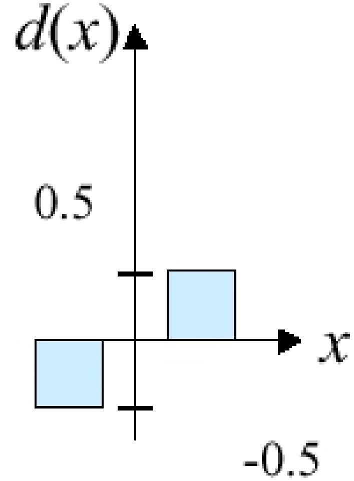
0	–1	0
–1	4	–1
0	–1	0

–1	–1	–1
–1	8	–1
–1	–1	–1

- Выделяются контуры независимо от их направления.
- Сумма весов фильтра равна **нулю**.
- Фильтры выделения контуров – дискретные **дифференциальные операторы.**

Численное дифференцирование





$$\mathbf{d} = \begin{bmatrix} -0.5 \\ 0 \\ 0.5 \end{bmatrix}$$

Ядро одномерного дифференциального фильтра

- Производную дискретной функции одной переменной можно численно оценить следующим образом:

$$\frac{df}{dx} \approx \frac{f(x+1) - f(x-1)}{2}$$

- Это равносильно свертке функции $f(x)$ с функцией $d(x)$ или действию дифференциального фильтра с ядром $\boldsymbol{d}$.

Дифференцирование изображения

- Производная изображения вычисляется путем фильтрации изображения дифференциальным фильтром.
- Один из возможных дифференциальных фильтров – фильтр Превитта 3x3:

-1	0	1
-1	0	1
-1	0	1

K

Пример дифференцирования изображения

- Как это работает?

-1	0	1
-1	0	1
-1	0	1

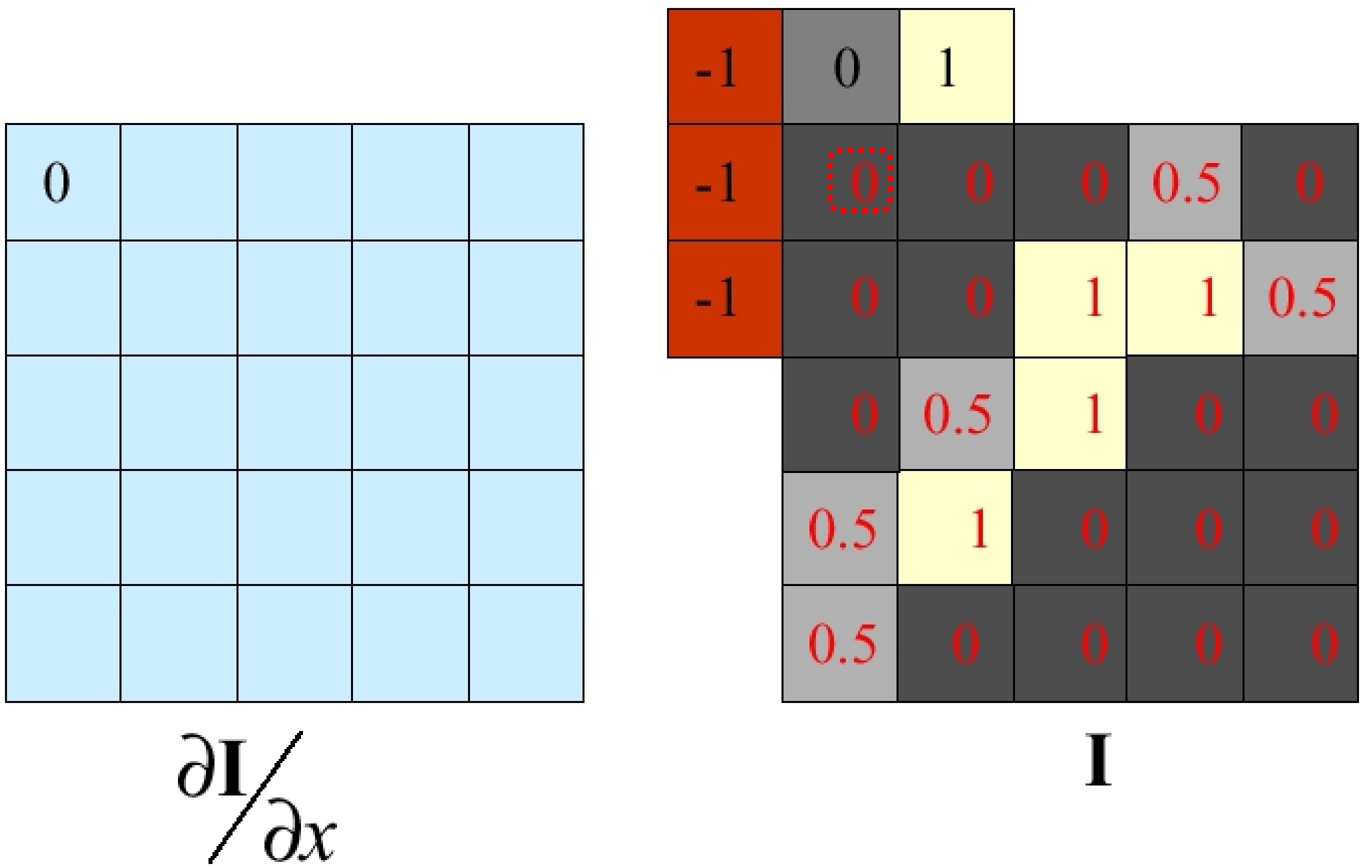
K

0	0	0	0.5	0
0	0	1	1	0.5
0	0.5	1	0	0
0.5	1	0	0	0
0.5	0	0	0	0

I

Пример дифференцирования изображения

- Хотим получить результат тех же размеров, что и изображение



Пример дифференцирования изображения

- Хотим получить результат тех же размеров, что и изображение

0	1			

$\partial \mathbf{I} / \partial x$

-1	0	1

0	0	0	0.5	0
0	0	1	1	0.5
0	0.5	1	0	0
0.5	1	0	0	0
0.5	0	0	0	0

$\mathbf{I}$

Пример дифференцирования изображения

- Хотим получить результат тех же размеров, что и изображение

0	1	1.5		

$$\partial \mathbf{I} / \partial x$$

	-1	0	1	

0	0	0	0.5	0
0	0	1	1	0.5
0	0.5	1	0	0
0.5	1	0	0	0
0.5	0	0	0	0

$$\mathbf{I}$$

Пример дифференцирования изображения

- Хотим получить результат тех же размеров, что и изображение

-1	0	1

0	1	1.5	-0.5	

$\partial \mathbf{I} / \partial x$

0	0	0	0.5	0
0	0	1	1	0.5
0	0.5	1	0	0
0.5	1	0	0	0
0.5	0	0	0	0

$\mathbf{I}$

Пример дифференцирования изображения

- Хотим получить результат тех же размеров, что и изображение

0	1	1.5	-0.5	-1.5

$\partial \mathbf{I} / \partial x$

			-1	0	1
0	0	0	0.5	0	1
0	0	1	1	0.5	1
0	0.5	1	0	0	
0.5	1	0	0	0	
0.5	0	0	0	0	

$\mathbf{I}$

Пример дифференцирования изображения

- Хотим получить результат тех же размеров, что и изображение

0	1	1.5	-0.5	-1.5
0.5				

$\partial \mathbf{I} / \partial x$

-1	0	0	0	0.5	0
-1	$\boxed{0}$	0	1	1	0.5
-1	0	0.5	1	0	0
	0.5	1	0	0	0
	0.5	0	0	0	0

$\mathbf{I}$

Пример дифференцирования изображения

- Хотим получить результат тех же размеров, что и изображение

0	1	1.5	-0.5	-1.5
0.5	2			

$$\partial\mathbf{I}/\partial x$$

0	0	0	0.5	0
0	0	1	1	0.5
0	0.5	1	0	0
0.5	1	0	0	0
0.5	0	0	0	0

$$\mathbf{I}$$

Пример дифференцирования изображения

- Хотим получить результат тех же размеров, что и изображение

0	1	1.5	-0.5	-1.5
0.5	2	1		

$\partial \mathbf{I} / \partial x$

0	0	0	0.5	0
0	0	$\boxed{1}$	1	0.5
0	0.5	1	0	0
0.5	1	0	0	0
0.5	0	0	0	0

$\mathbf{I}$

Пример дифференцирования изображения

- Фильтрация завершена
- Вообще говоря, значения отклика могут не укладываться в диапазон квантования.

0	1	1.5	-0.5	-1.5
0.5	2	1	-1.5	-1.5
1.5	1.5	-0.5	-1.5	-1
1.5	0	-1.5	-1	0
1	-1	-1	0	0

$\partial \mathbf{I} / \partial x$

0	0	0	0.5	0
0	0	1	1	0.5
0	0.5	1	0	0
0.5	1	0	0	0
0.5	0	0	0	0

$\mathbf{I}$

Пример дифференцирования изображения

- Представляем результат как изображение

0	1	1.5	-0.5	-1.5
0.5	2	1	-1.5	-1.5
1.5	1.5	-0.5	-1.5	-1
1.5	0	-1.5	-1	0
1	-1	-1	0	0

$\partial \mathbf{I} / \partial x$

0	0	0	0.5	0
0	0	1	1	0.5
0	0.5	1	0	0
0.5	1	0	0	0
0.5	0	0	0	0

$\mathbf{I}$

Пример дифференцирования изображения

- Вычислили

$$\partial I / \partial x = I \quad K$$

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

- Используем транспонированное ядро фильтра для вычисления

$$\partial I / \partial y = I \quad K^T$$

$$\mathbf{K}^{\mathbf{T}} \quad \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

Пример дифференцирования изображения

- Используем ту же процедуру, что и при вычислении производной по x

0	1	2	2.5	1.5
0.5	1.5	1	0.5	-0.5
1.5	0.5	-1	-2.5	-1.5
0	-1	-1.5	-1	0
-1.5	-1.5	-1	0	0

$\partial \mathbf{I} / \partial y$

0	0	0	0.5	0
0	0	1	1	0.5
0	0.5	1	0	0
0.5	1	0	0	0
0.5	0	0	0	0

$\mathbf{I}$

Пример дифференцирования изображения

- Используем ту же процедуру, что и при вычислении производной по x

0	1	2	2.5	1.5
0.5	1.5	1	0.5	-0.5
1.5	0.5	-1		-1.5
0	-1	-1.5	-1	0
-1.5	-1.5	-1	0	0

$\partial \mathbf{I} / \partial y$

0	0	0	0.5	0
0	0	1	1	0.5
0	0.5	1	0	0
0.5	1	0	0	0
0.5	0	0	0	0

$\mathbf{I}$

Пример дифференцирования изображения

- Таким образом, мы вычислили градиент изображения $\boldsymbol{I}$ (используя $\boldsymbol{K}$ — оператор Превитта с размерами 3x3)

$$I = \left[\frac{\partial I}{\partial x} \quad \frac{\partial I}{\partial y} \right]^T$$

Дифференциальные фильтры

- Существует много других дифференциальных фильтров. Наиболее известны из них:

Фильтр Превитта

–1	0	1
–1	0	1
–1	0	1

Фильтр Собеля

–1	0	1
–2	0	2
–1	0	1

Фильтр Робертса

0	1
–1	0

1	0
0	–1

y–составляющая не может быть получена транспонированием!

- Используются маски бо́льших размеров
- Компоненты градиента могут быть измерены в двух любых ортогональных направлениях. Фильтр Робертса использует диагональные направления.

Свойства дифференцирующих фильтров

- Дифференцирующие фильтры содержат отрицательные элементы.
- Сумма весов дифференцирующего фильтра равна нулю, т.е. в областях постоянной интенсивности отклик фильтра равен нулю.
- Фильтры, вычисляющие первую производную, в точках больших перепадов интенсивности порождают большие по абсолютной величине значения откликов.
- Фильтры, вычисляющие вторую производную, в точках перепада интенсивности порождают нулевой отклик.

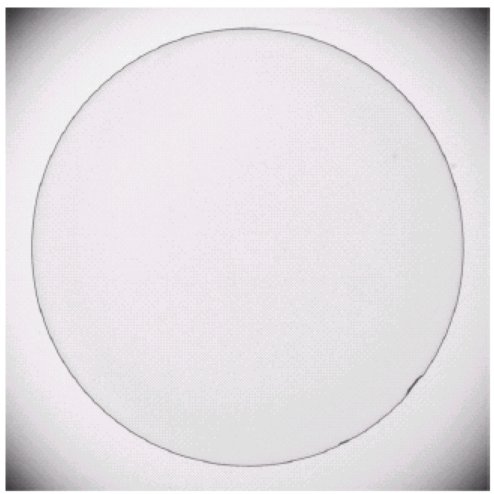
Замечания по использованию дифференциальных фильтров

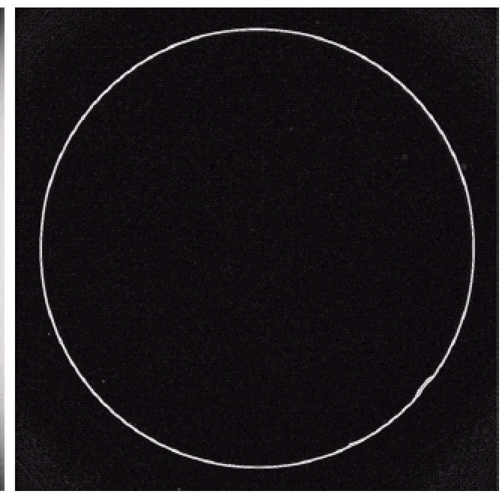
- Фильтр Робертса вычислительно более эффективен, благодаря маленькому количеству задействуемых соседей.
- Все локальные дифференциальные операторы «теряют» многие из границ.
- Вычисление квадратов и квадратных корней вычислительно слишком дорого. Можно заменять их вычислением абсолютных значений, максимумов и т.п.

Пример действия фильтра Собеля

Изображение контактной линзы с дефектами на краю (в районе 4–5 часов)

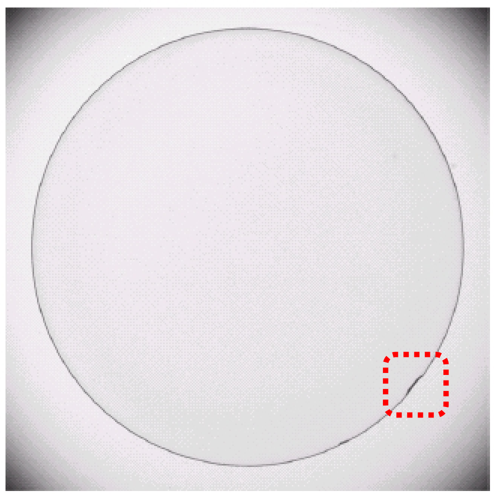
Результат действия фильтра Собеля



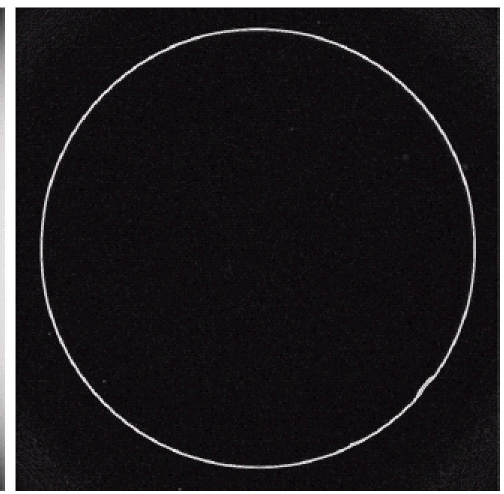


Пример действия фильтра Собеля

Изображение контактной линзы с дефектами на краю (в районе 4–5 часов)

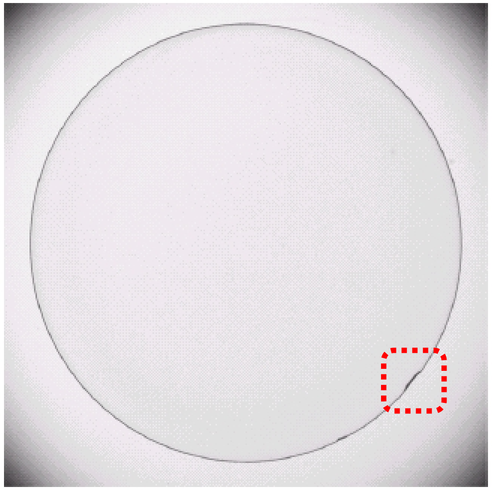


Результат действия фильтра Собеля

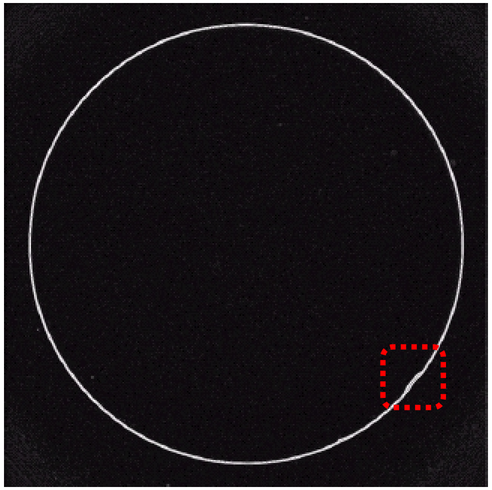


Пример действия фильтра Собеля

Изображение контактной линзы с дефектами на краю (в районе 4–5 часов)



Результат действия фильтра Собеля



Анализ бинарных изображений

Бинарные изображения

- Бинарные изображения содержат пикселы только с двумя значениями интенсивности – максимальным и минимальным.
- Бинарные изображения особенно полезны для
  – Идентификации объектов с заданным силуэтом (например, деталей, движущихся по конвейеру);
  – Распознавания символов и текста (например, при обработке документов, интерпретации дорожных знаков);
  – Определения ориентации объектов в пространстве.
- Бинарные изображения просты для анализа.

Операции математической морфологии

- Операции математической морфологии позволяют:
  – утолщать объекты
  – утоньчать объекты
  – отыскивать границы объектов
  – подавлять бинарный шум (мелкие объекты)
  – и пр.

Множества точек и обозначения

- Бинарные объекты рассматриваются как множества точек
- Для множеств точек A и B обозначим
  - **сдвиг** A на x через $A_x = \{a_i + x \mid a_i \in A\}$
  - **отражение** B через $B^r = \{-b_i \mid b_i \in B\}$
  - **дополнение** A через $A^c = \{a_i \mid a_i \notin A\}$
  - **разность** A и B через $A - B = \{c_i \mid (c_i \in A) \text{ XOR } (c_i \in B)\}$

Дилатация и эрозия

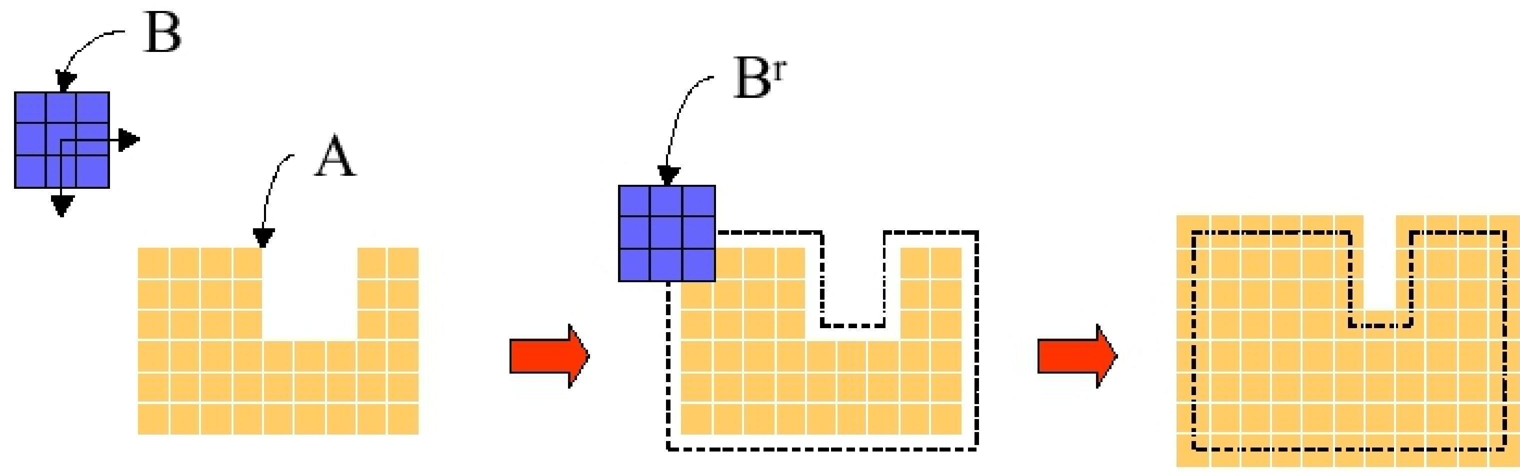
- Двумя основными операциями математической морфологии являются:
  - Дилатация (расширение)
  - Эрозия (сужение)
- Каждая из этих операций бинарна, т.е. задействует два множества:
  - Интересующий объект
  - Структурный элемент

Дилатация

- Дилатация множества A структурным элементом B есть множество

$$\mathrm{dil}(A,B) = \{x_i \mid (B_x^r \cap A) \neq \varnothing\}$$

- Пример. Дилатация множества A структурным элементом B размера 3x3, с центром в $(0,0)$.

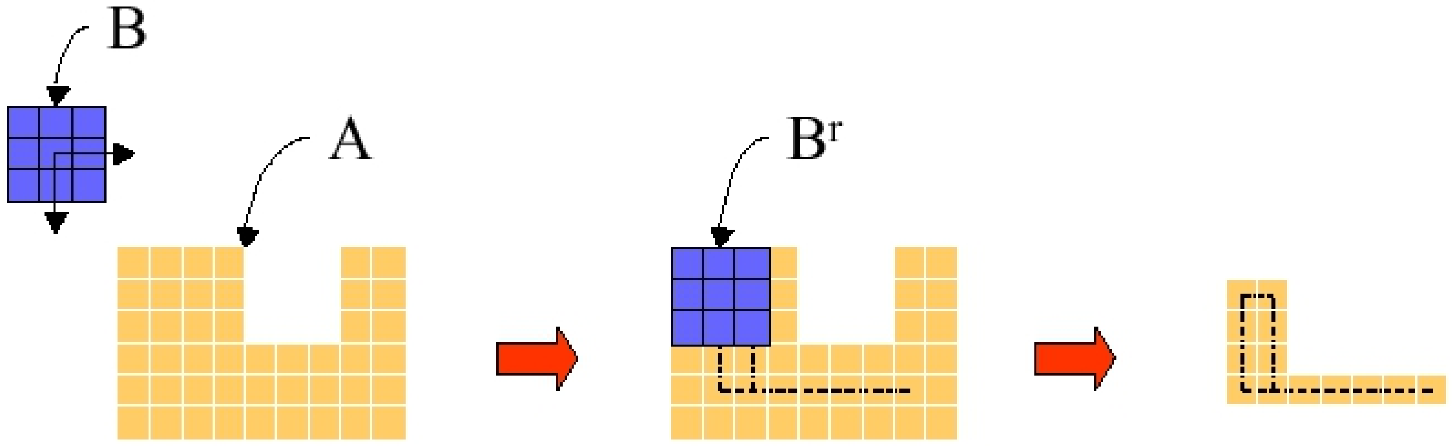


Эрозия

- Эрозия множества A структурным элементом B есть множество

$$er(A,B) = \{x_i \mid B_x \subseteq A\}$$

- Пример. Эрозия множества A структурным элементом B размера 3x3, с центром в (0,0).



Комбинации дилатации и эрозии

- На базе основных операций математической морфологии, получают операции
  - Заполнение, размыкание, открытие (opening)
  - Пополнение, замыкание, закрытие (closing)
  - Утончение (thinning)
  - Утолщение (thickening)
  - …

Наиболее важны

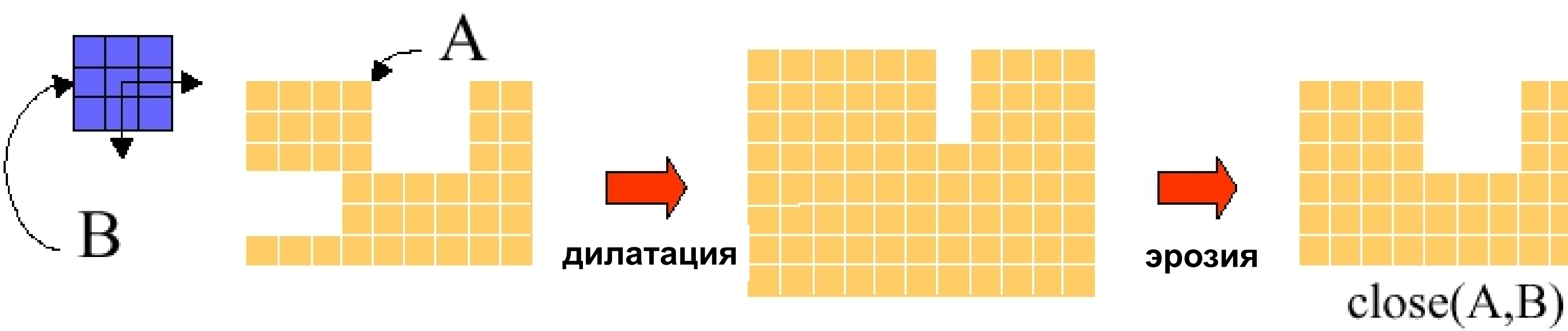
Интуитивная интерпретация

- Дилатация
  - Расширяет объект
- Эрозия
  - Сжимает объект
- Открытие (open)
  - Сглаживает контуры
  - Увеличивает узкие разрывы
  - Ликвидирует тонкие выступы
- Закрытие (close)
  - Заполняет небольшие разрывы, дырки.

Закрытие

- Закрытие можно рассматривать как «сглаживание снаружи»

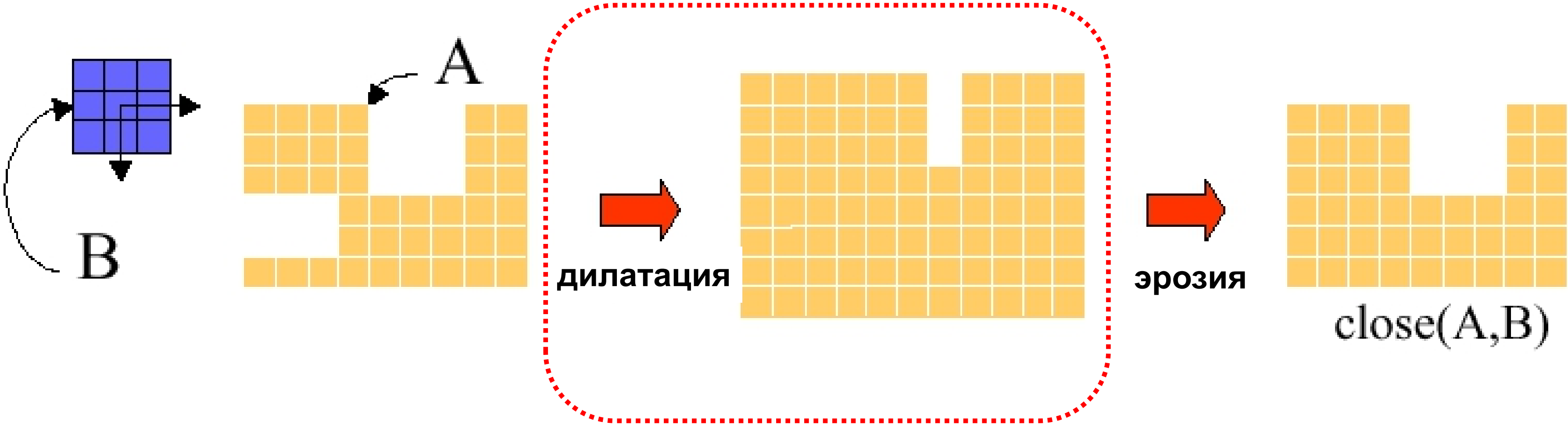
$$close(A,B) = er(dil(A,B),B)$$



Закрытие

- Закрытие можно рассматривать как «сглаживание снаружи»

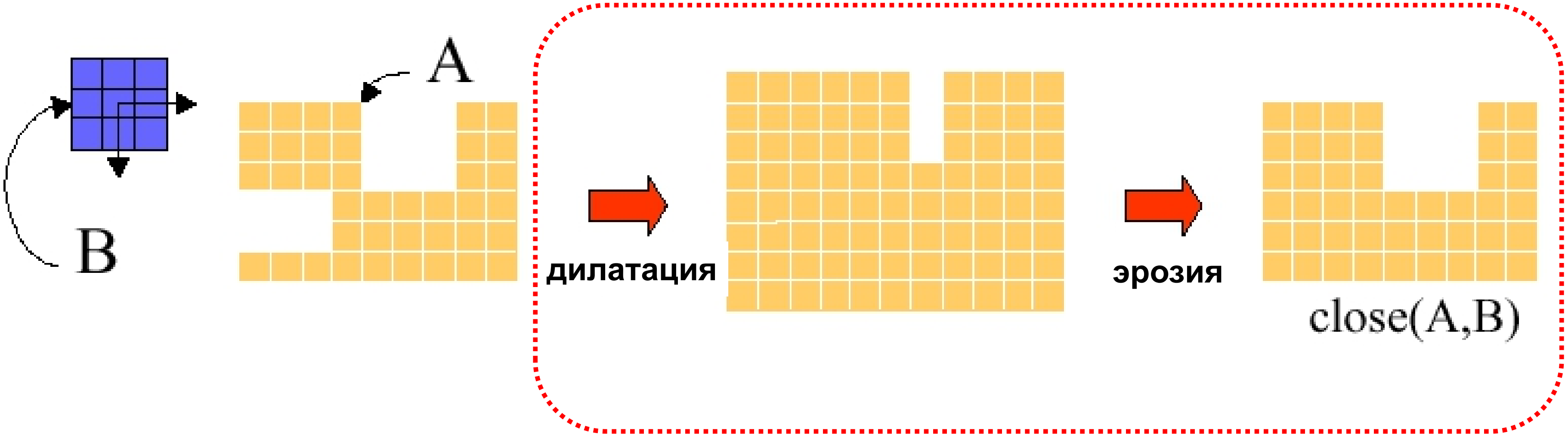
$$close(A,B) = er(\boxed{dil(A,B)},B)$$



Закрытие

- Закрытие можно рассматривать как «сглаживание снаружи»

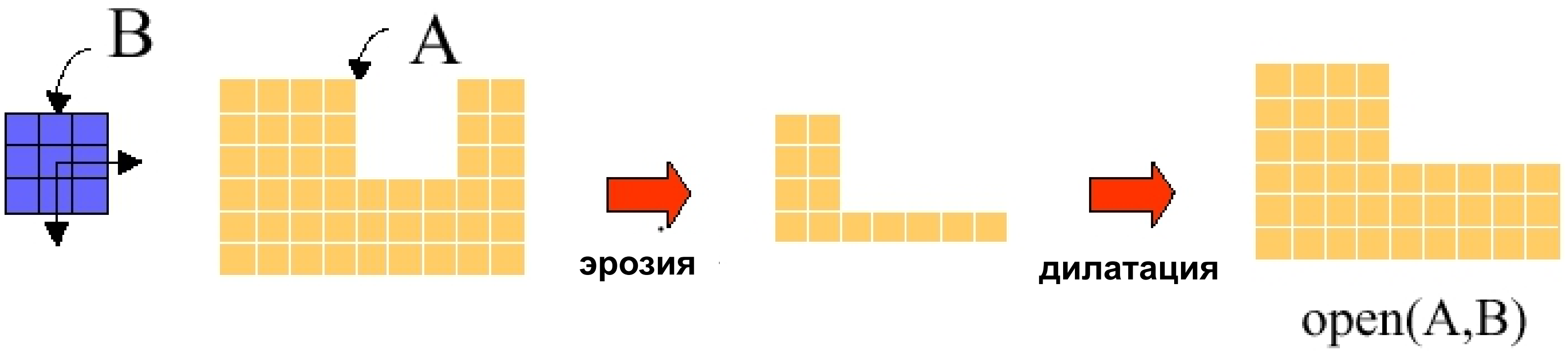
$$close(A,B) = \boxed{er(dil(A,B),B)}$$



Открытие

- Открытие можно рассматривать как «сглаживание изнутри»

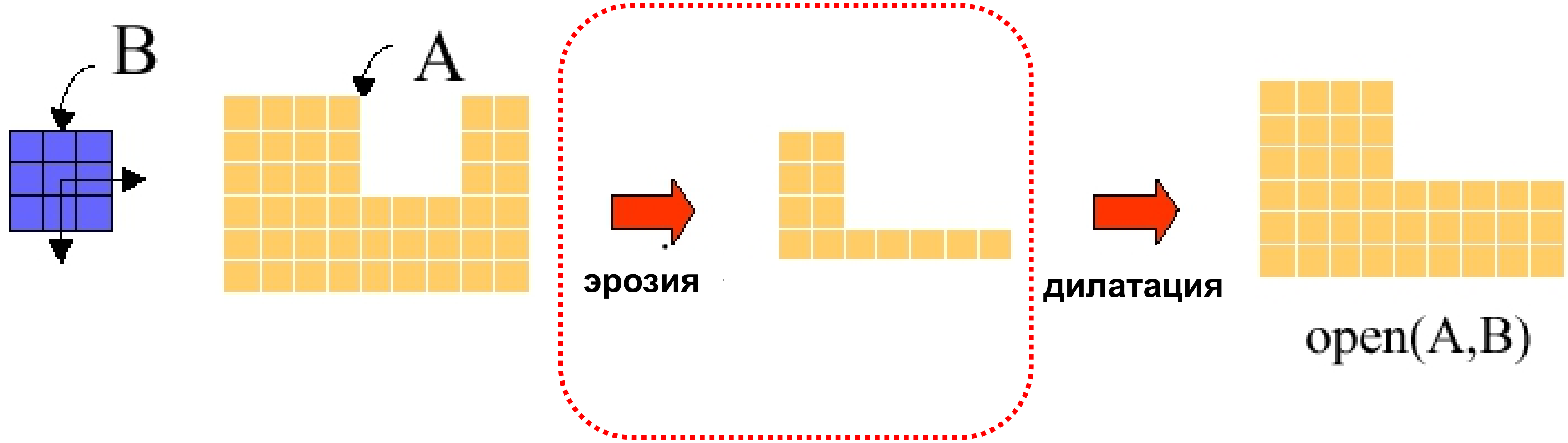
$$open(A,B) = dil\,(er\,(A,B),B) = \cup\{B_x \mid B_x \subseteq A\}$$



Открытие

- Открытие можно рассматривать как «сглаживание изнутри»

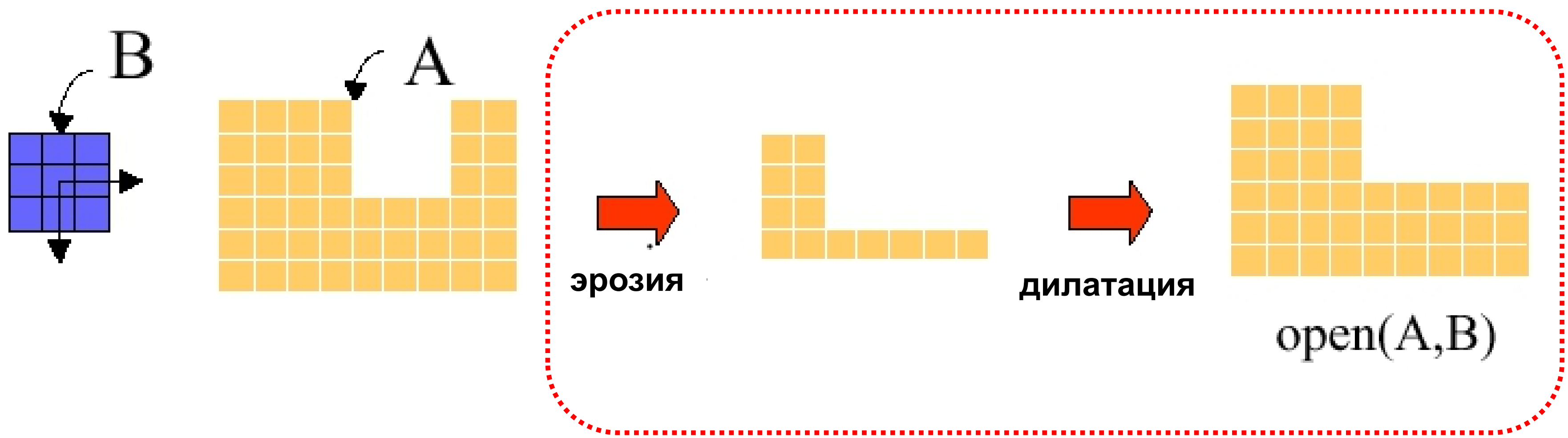
$$open(A,B) = dil\,(er\,(A,B),B) = \cup\{B_x |\ B_x \subseteq A\}$$



Открытие

- Открытие можно рассматривать как «сглаживание изнутри»

$$\mathrm{open}(A,B) = \boxed{\mathrm{dil}\,(\mathrm{er}\,(A,B),B)} = \cup\{B_x \mid B_x \subseteq A\}$$

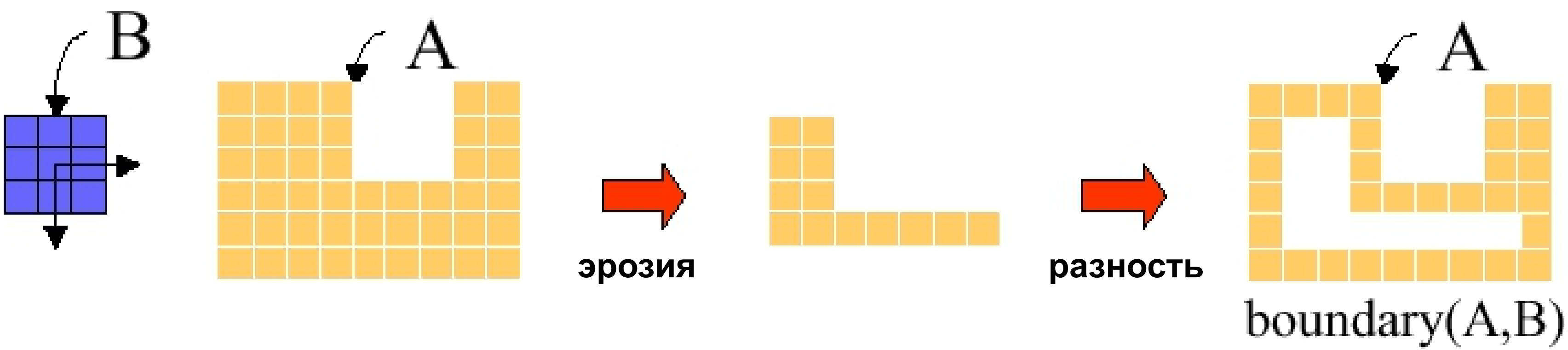


Идемпотентность

- Повторные применения открытия и закрытия не имеют эффекта, т.е.
  - open(open(A,B),B) = open(A,B)
  - close(close(A,B),B) = close (A,B)

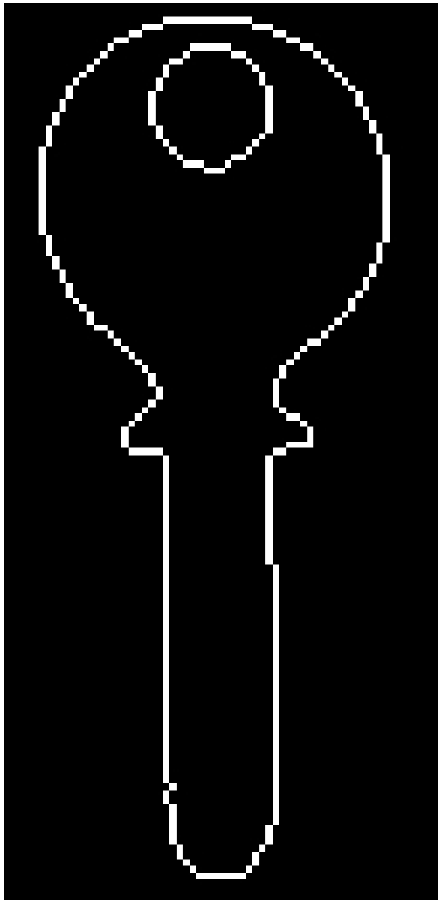
Поиск границы объекта

$$\text{boundary}(A,B) = A - \text{er}(A,B)$$



Поиск границы объекта

- Пример

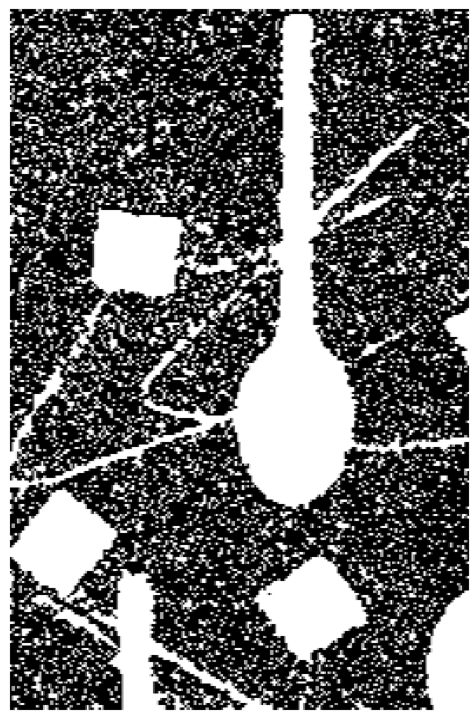


Морфологическое подавление шума

- Широко известный способ включает следующие шаги:
  - Эрозия (erosion)
  - Дилатация (dilation)
  - Закрытие (closing)
  - Открытие (opening)

Морфологическое подавление шума

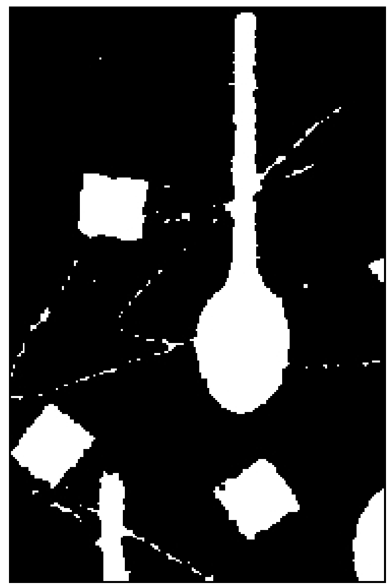
- Пример бинарного изображения с сильным шумом



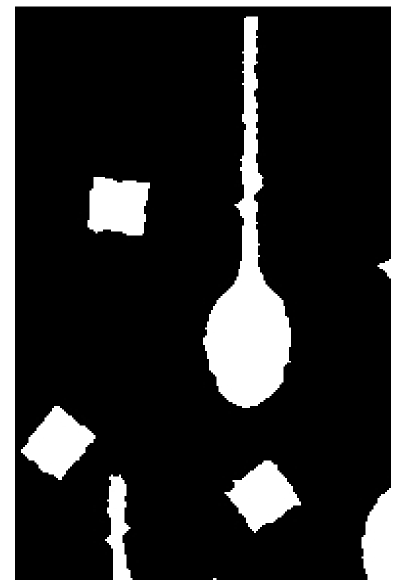
Морфологическое подавление шума

- Применение эрозии с различными структурными элементами

0	1	0
1	1	1
0	1	0



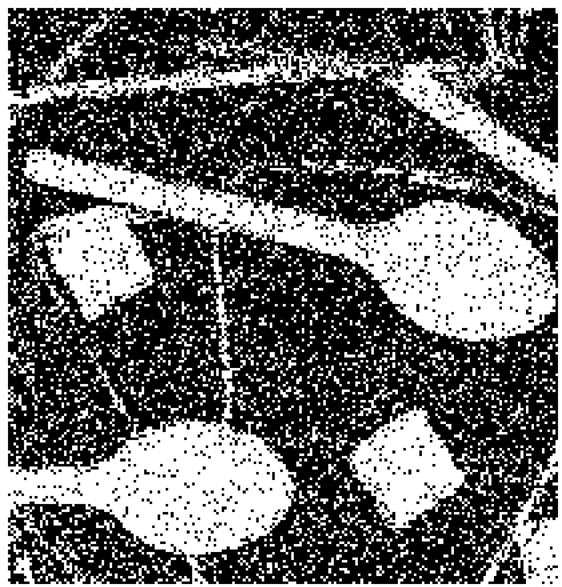
1	1	1
1	1	1
1	1	1



0	0	1	1	1	0	0
0	1	1	1	1	1	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	1	1	1	1	1	0
0	0	1	1	1	0	0

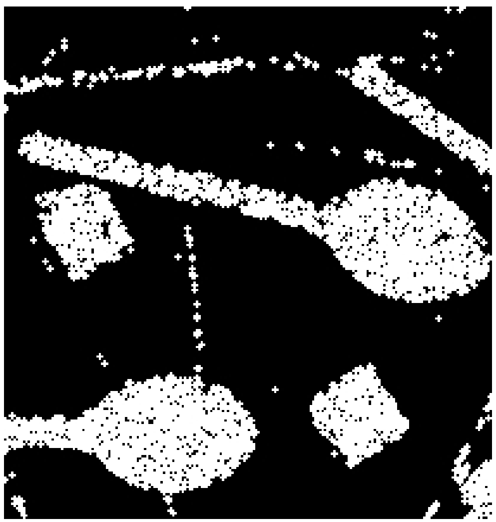
Морфологическое подавление шума

- Не во всех случаях математическая морфология так легко убирает дефекты, как хотелось бы

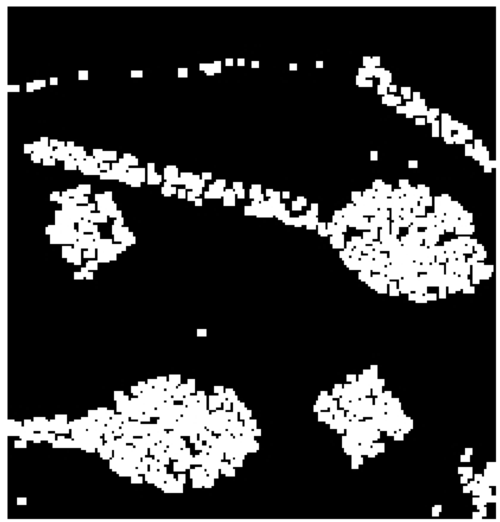


Морфологическое подавление шума

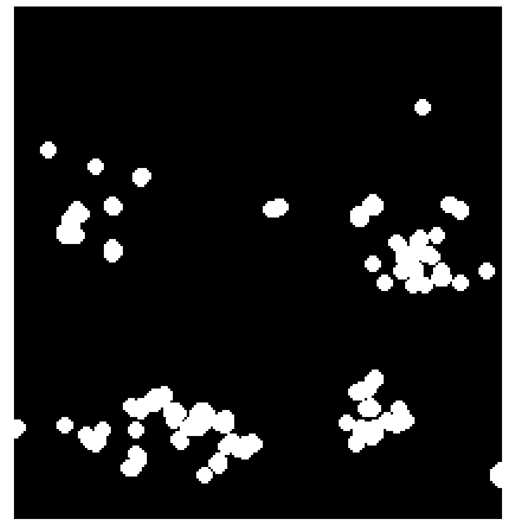
- Примеры применения операции открытия (пополнения) с различными структурными элементами



0	1	0
1	1	1
0	1	0



1	1	1
1	1	1
1	1	1



0	0	1	1	1	0	0
0	1	1	1	1	1	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	1	1	1	1	1	0
0	0	1	1	1	0	0