ЧАСТОТНОЕ ПРЕДСТАВЛЕНИЕ. МОРФОЛОГИЯ. Лекция 3.

Преподаватель: Сибирцева Елена
elsibirtseva@gmail.com

В предыдущих сериях…

Проблемы


Темное или слабоконтрастное

Неправильные цвета

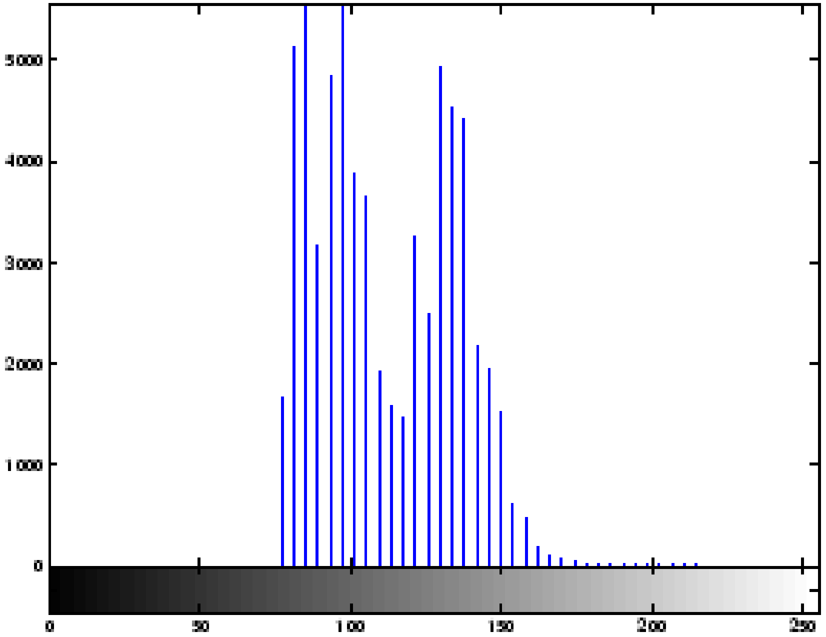
Шумное

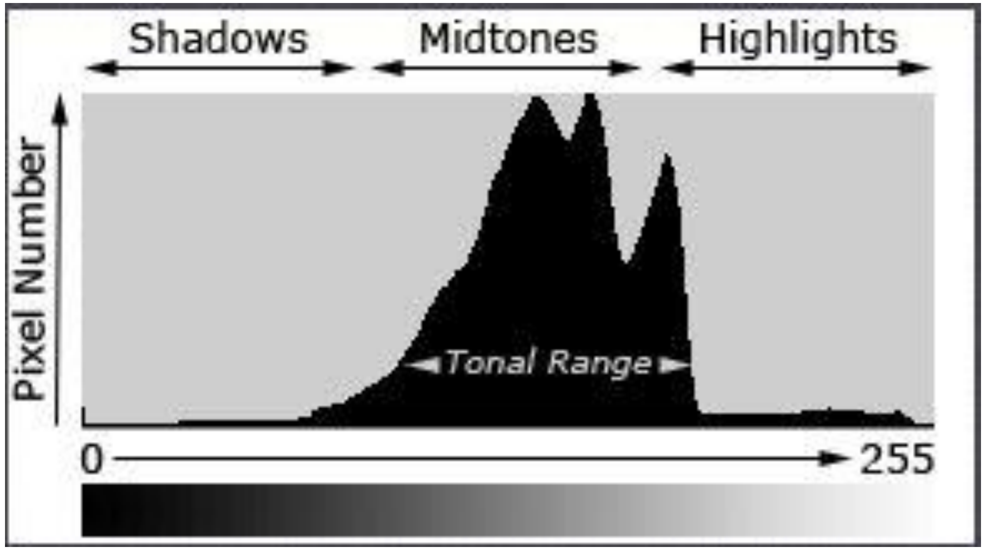
Нерезкое

Неравномерно освещённое

Гистограмма изображения

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





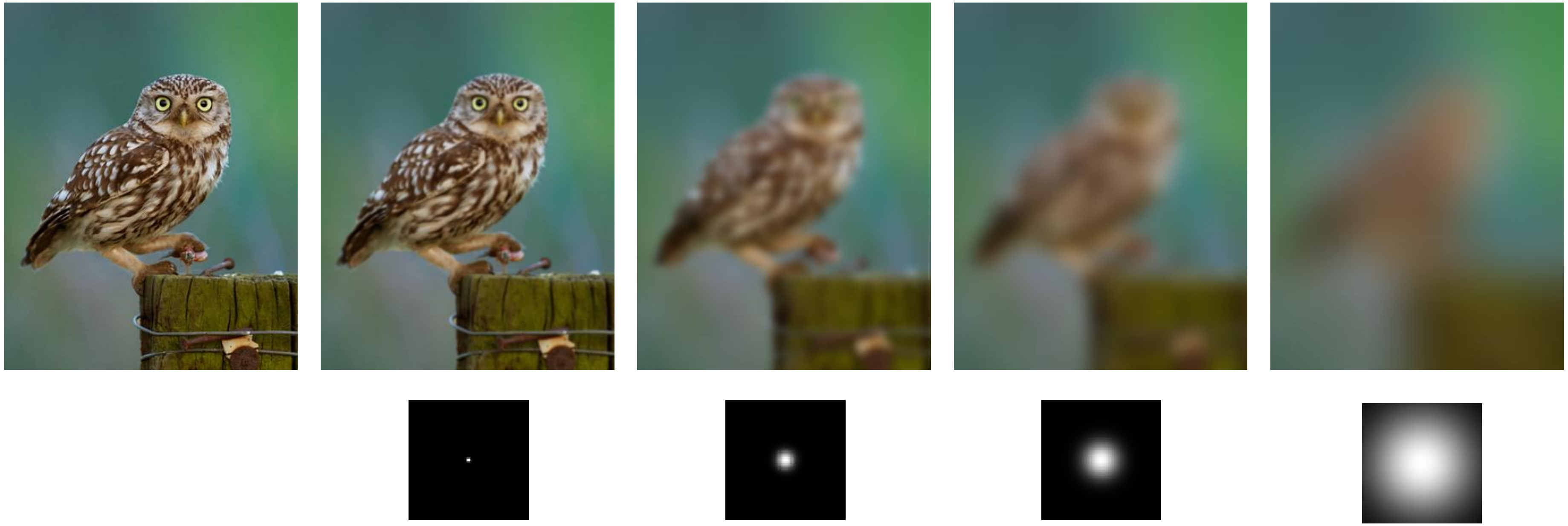
Свертка

$$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} H[u,v]F[i-u,j-v]$$

Это называется **оператором свертки**:

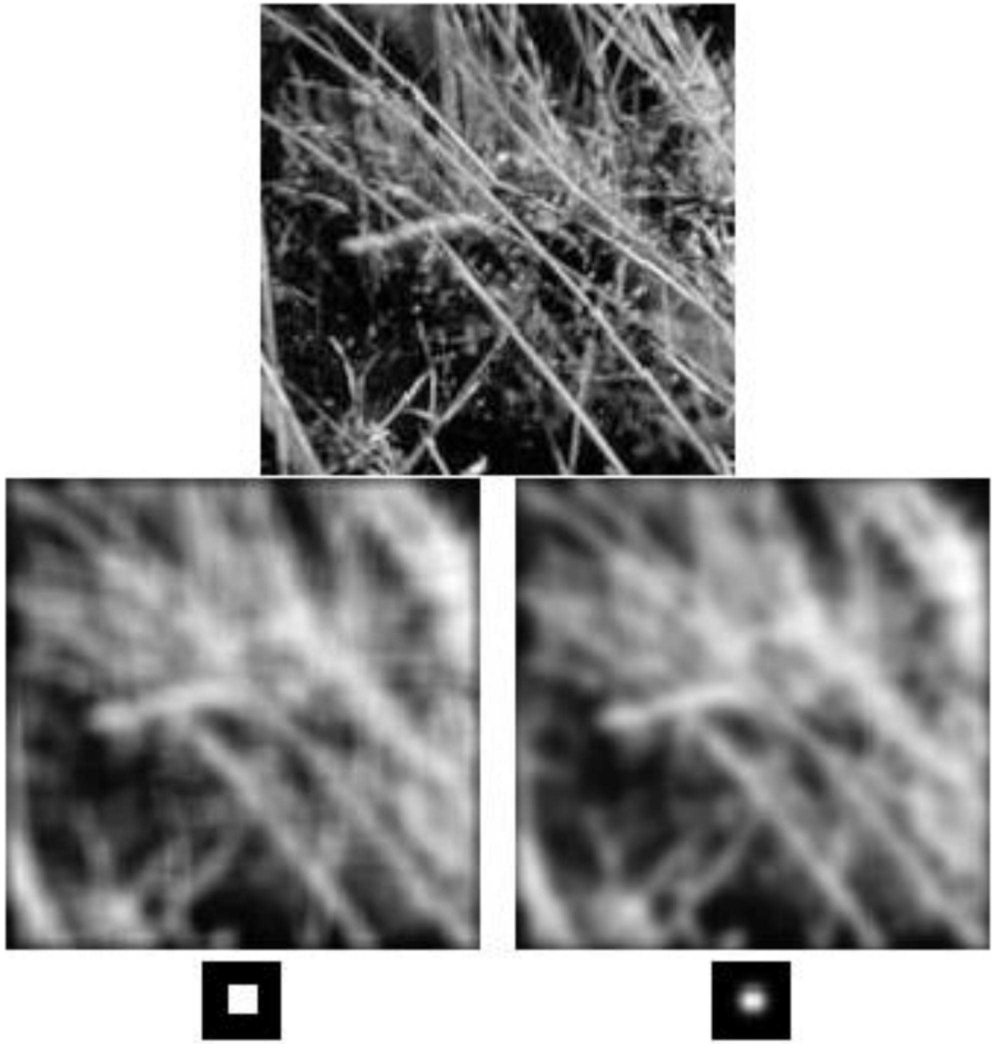
$$G = H * F$$

Гауссовы фильтры



σ = 1 pixel σ = 5 pixels σ = 10 pixels σ = 30 pixels

Среднее vs. Гауссов фильтр



ЧАСТОТНОЕ ПРЕДСТАВЛЕНИЕ ИЗОБРАЖЕНИЙ

Jean Baptiste Joseph Fourier

- Дикая идея (1807):

Любая периодическая функция может быть представлена как взвешенная сумма синусов и косинусов различной частоты

- Воспринята была не сразу:

Ни Лагранж, ни Лаплас, Пуассон не верили в это

Впервые переведена работа на английский в 1878 году

- Преобразование Фурье

Ряд Фурье

Идея: любая периодическая функция f(x) с периодом T может быть представлена в виде ряда:

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{+\infty} A_k \cos(2\pi \frac{k}{T} x + \theta_k)$$

где cos (k) – k-ая гармоника, A_k – амплитуда k-ой гармоники, θ_k – фаза k-ой гармоники

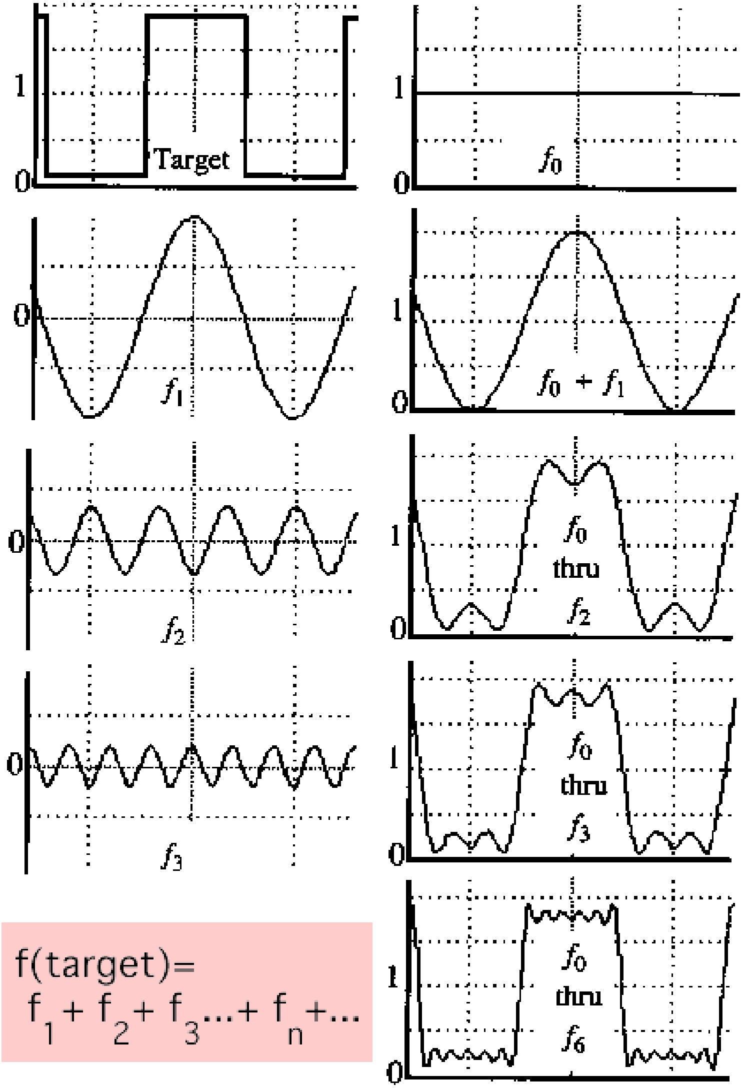
Сумма синусов

Структурный элемент:

$$A\sin(\omega x + \phi)$$

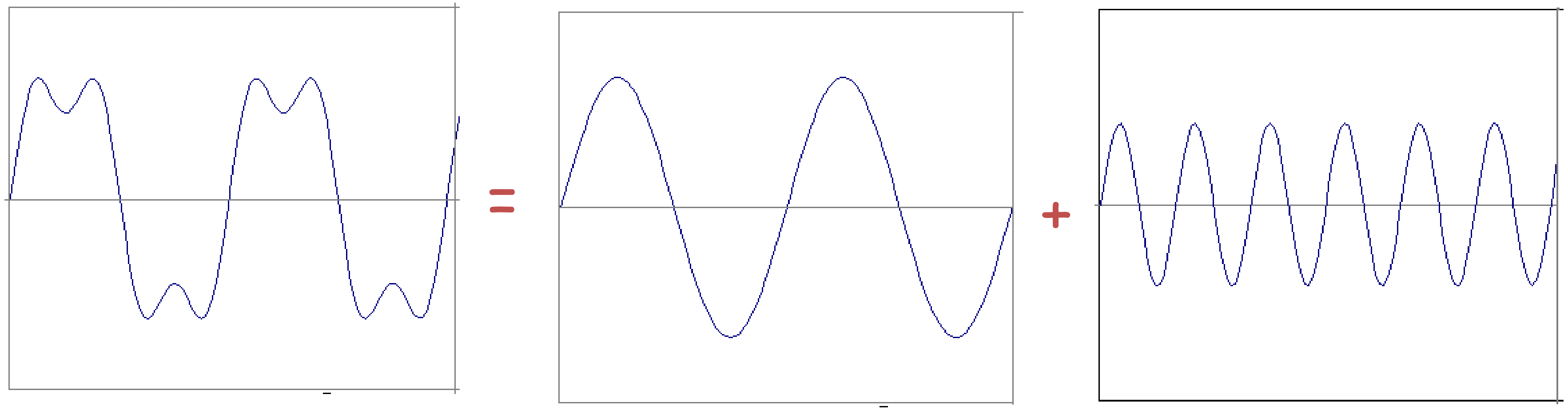
Если добавить их в достаточном количестве, то можно получить любой сигнал *g*(*x*)!

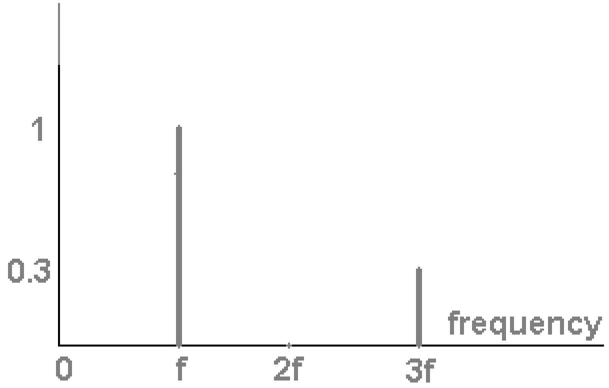
$A\sin(\omega x$



Частотный спектр

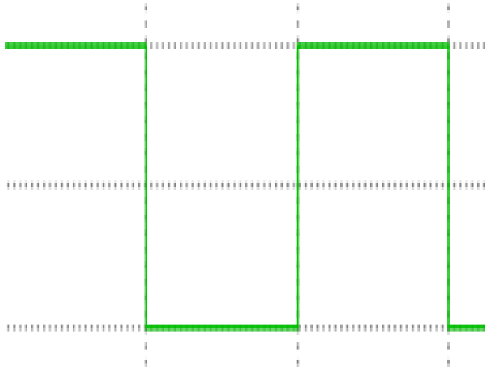
- Пример: $g(t) = \sin(2\pi f\, t) + (1/3)\sin(2\pi(3f)\, t)$



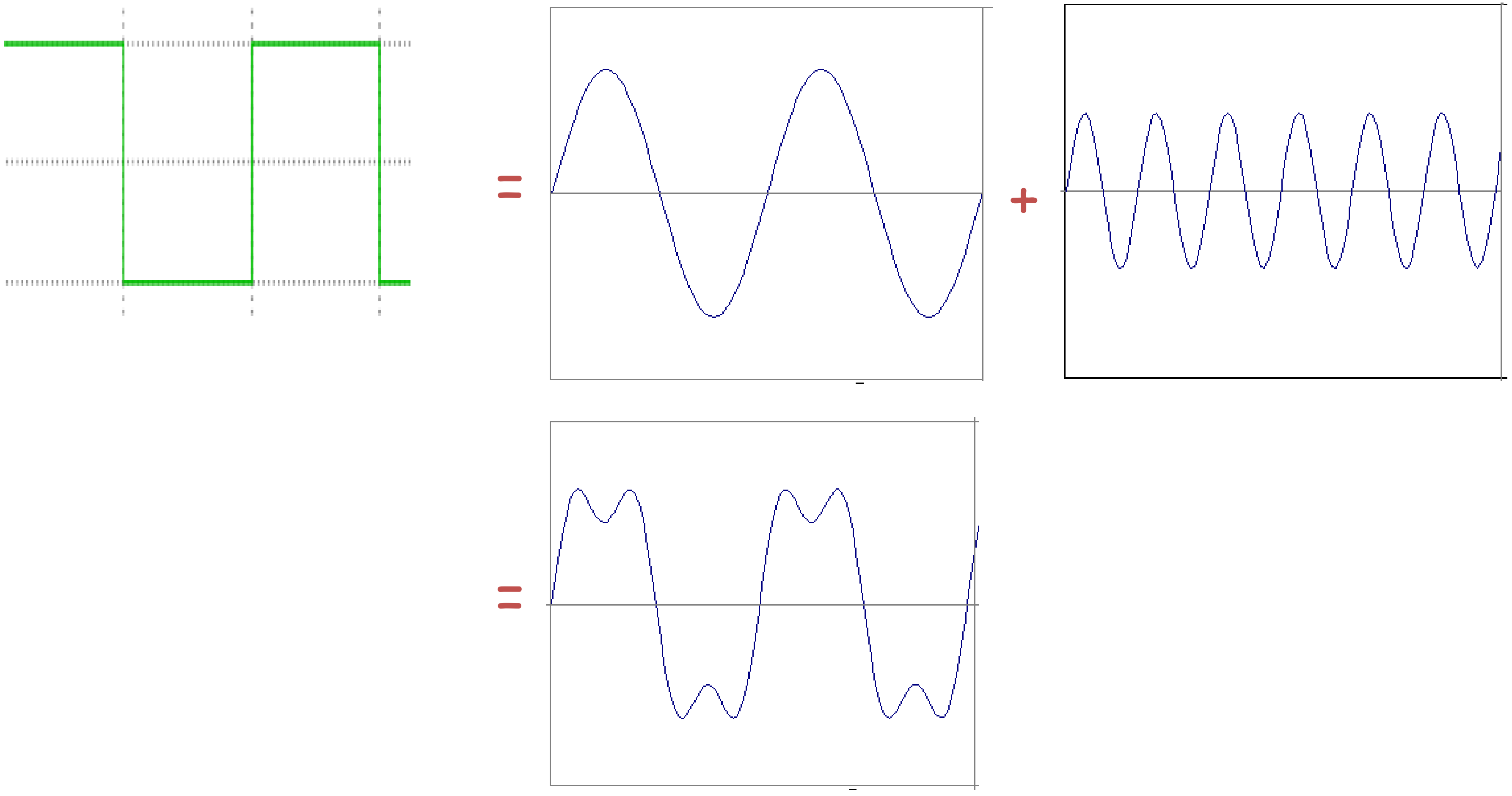


Спектр –график зависимости амплитуды от частоты и фазы.

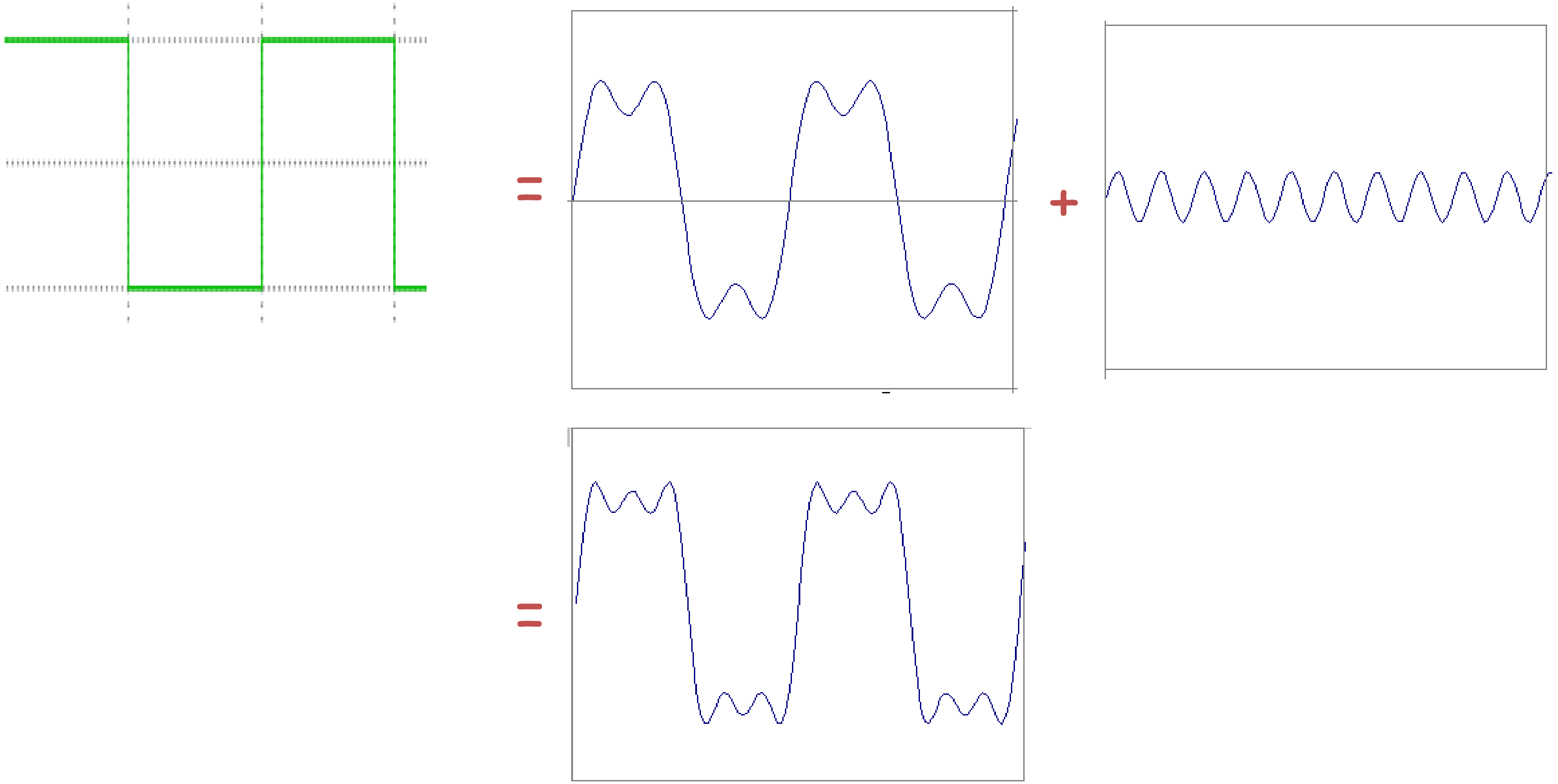
Частотный спектр



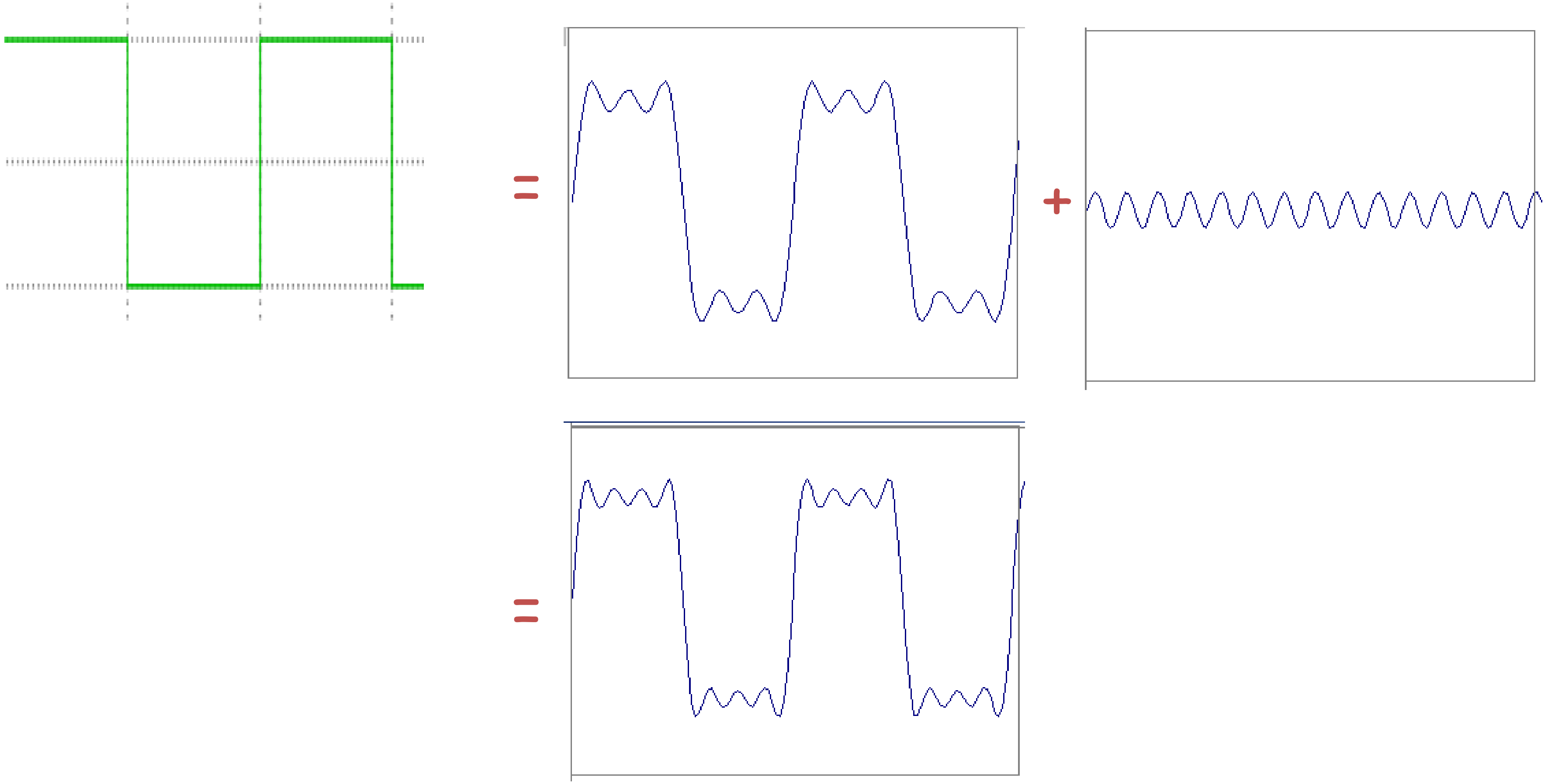
Частотный спектр



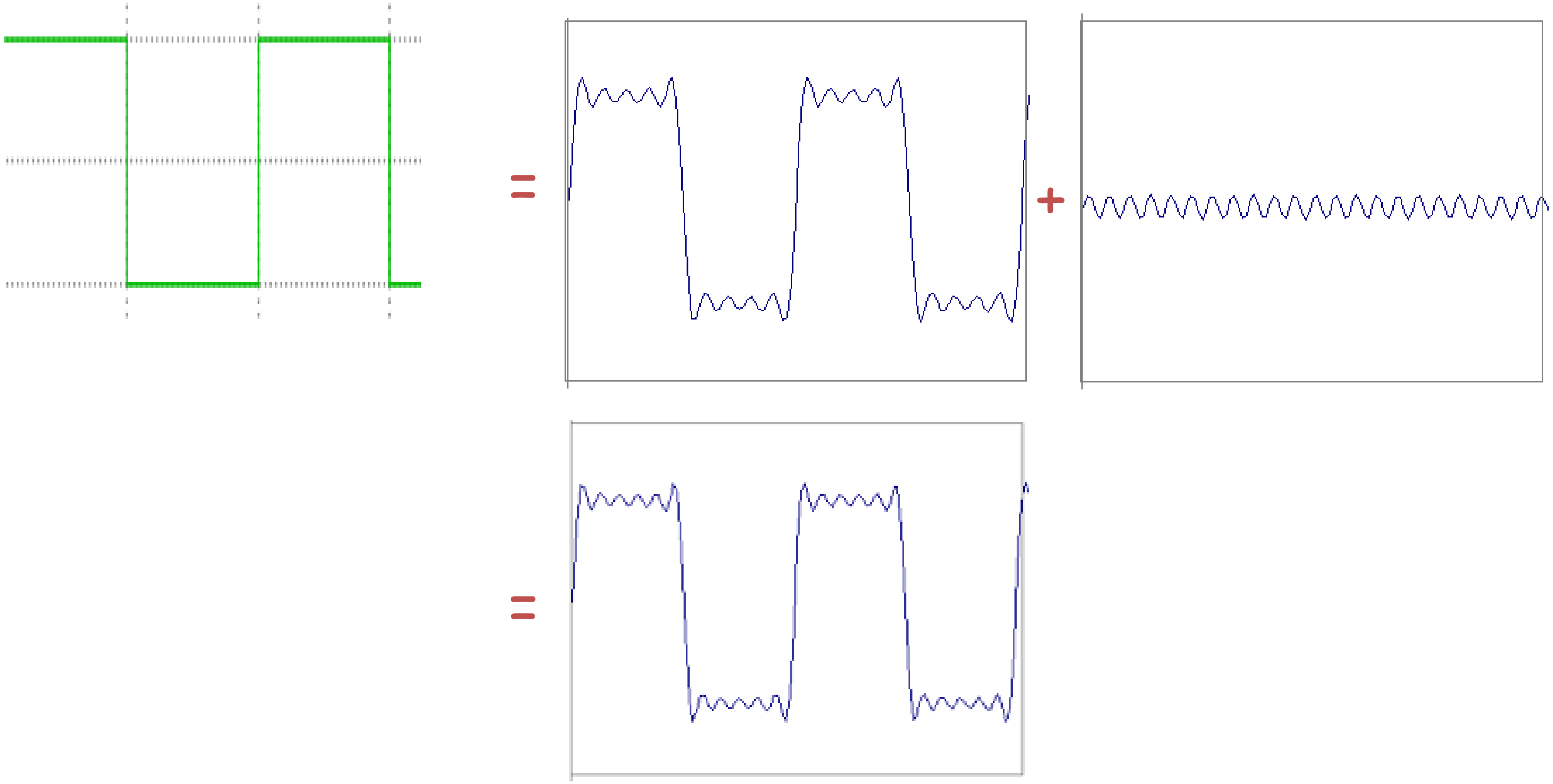
Частотный спектр



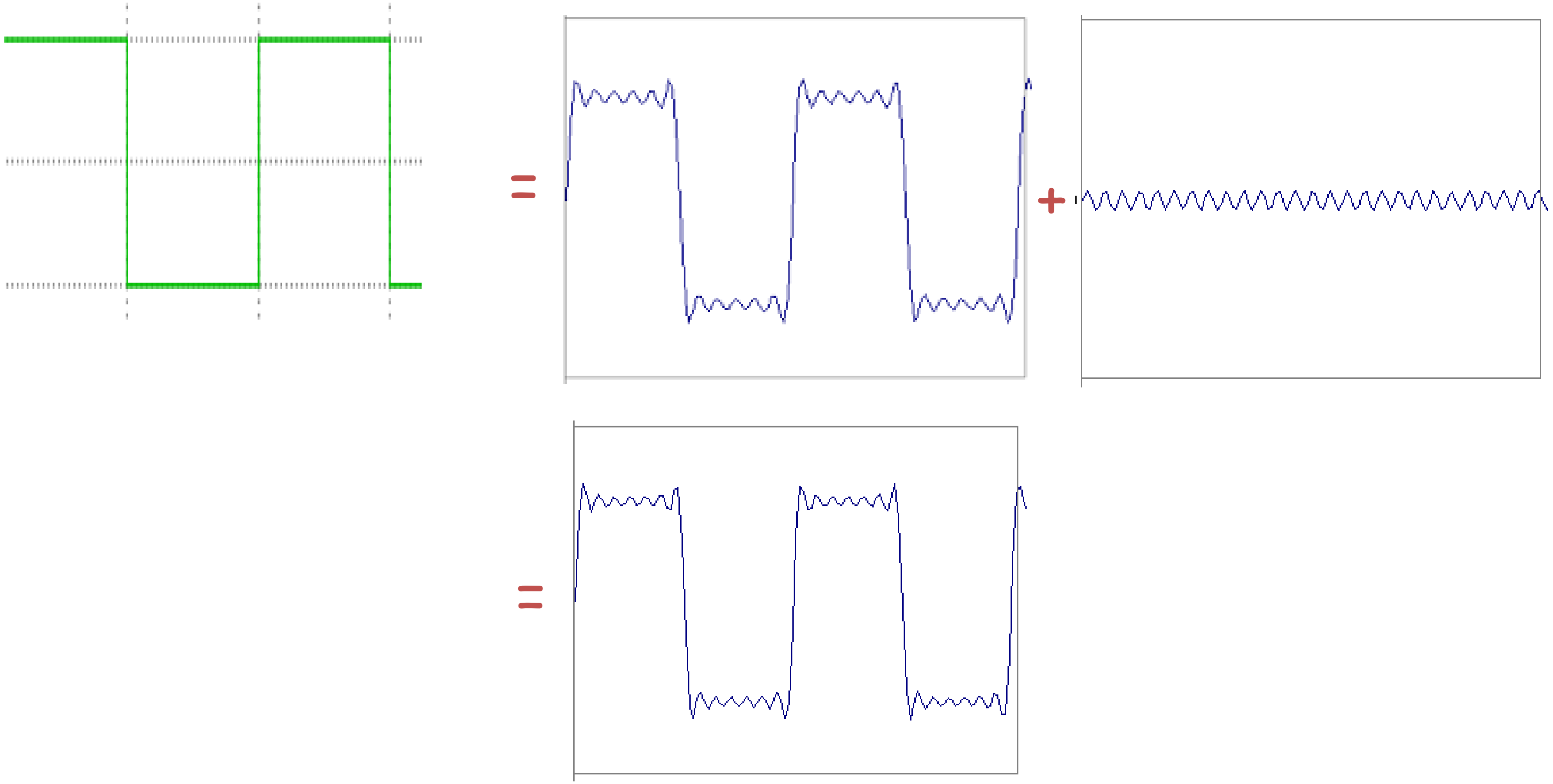
Частотный спектр



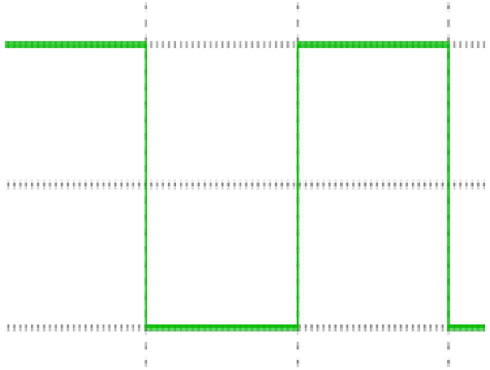
Частотный спектр



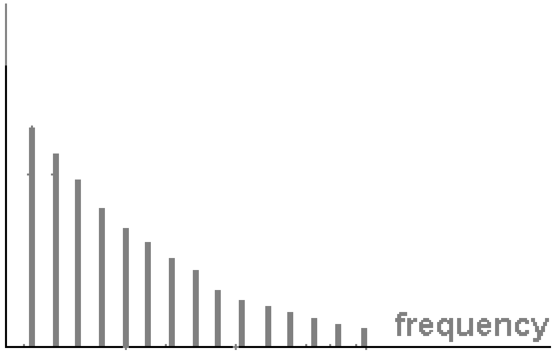
Частотный спектр



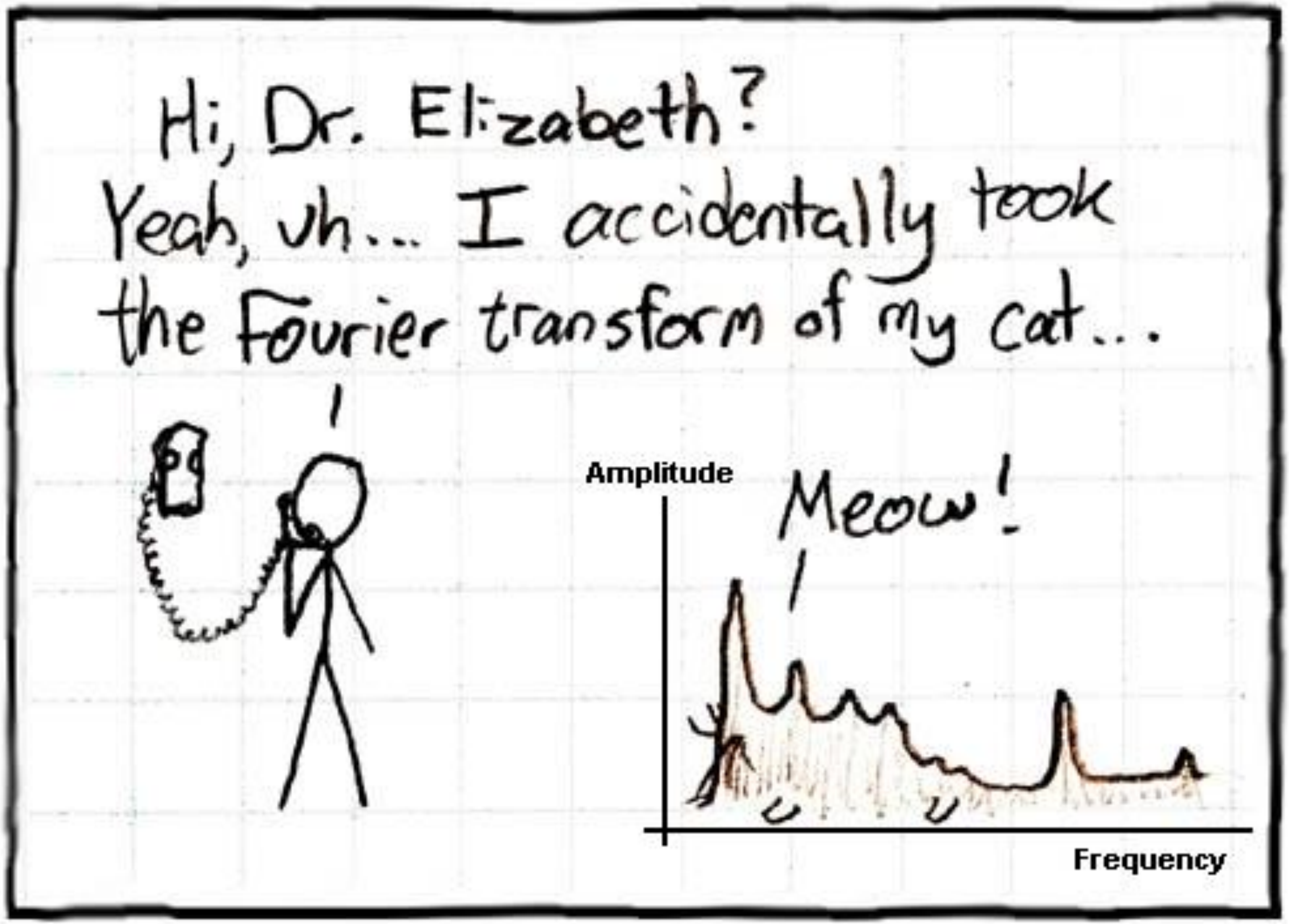
Частотный спектр



$$= \boxed{A\sum_{k=1}^{\infty}\frac{1}{k}\sin(2\pi kt)}$$



ПРИМЕРЫ СИГНАЛОВ?


Hi, Dr. Elizabeth?
Yeah, uh... I accidentally took the Fourier transform of my cat...
Amplitude
Meow!
Frequency

Дискретное преобразование Фурье

Для дискретных сигналов длиной N можно ввести прямое дискретное преобразование Фурье:

f(x) → Fourier Transform → $\boldsymbol{F(\omega)}$

Для каждой ω от 0 до N-1, $\boldsymbol{F(\omega)}$ содержит амплитуду A и фазу ϕ соответствующего синуса или косинуса

- Для удобной записи используются мнимые числа:

$$F(\omega) = R(\omega) + iI(\omega)$$

$$A = \pm\sqrt{R(\omega)^2 + I(\omega)^2} \qquad \phi = \tan^{-1}\frac{I(\omega)}{R(\omega)}$$

Преобразование Фурье

- Разложение Фурье обратимо, т.е. по коэффициентам разложения можно точно восстановить исходный дискретный сигнал.
- Обратное преобразование Фурье:

$$F(\omega) \longrightarrow \boxed{\text{Inverse Fourier Transform}} \longrightarrow f(x)$$

Вычисление ДПФ

- ДПФ является линейным преобразованием
- Базисные функции (косинусы или синусы) образуют N-мерный ортогональный базис в пространстве N-мерных векторов исходных сигналов.
- Весовые коэффициенты вычисляются как скалярное произведение сигнала на базисные функции

ДПФ: $$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N}kn} \qquad k = 0, \dots, N-1$$

Запись в матричной форме:

$$\vec{X} = \hat{A}\vec{x}$$

$$\hat{A} = \begin{pmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & e^{-\frac{2\pi i}{N}} & e^{-\frac{4\pi i}{N}} & e^{-\frac{6\pi i}{N}} & \dots & e^{-\frac{2\pi i}{N}(N-1)} \\ 1 & e^{-\frac{4\pi i}{N}} & e^{-\frac{8\pi i}{N}} & e^{-\frac{12\pi i}{N}} & \dots & e^{-\frac{2\pi i}{N}2(N-1)} \\ 1 & e^{-\frac{6\pi i}{N}} & e^{-\frac{12\pi i}{N}} & e^{-\frac{18\pi i}{N}} & \dots & e^{-\frac{2\pi i}{N}3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-\frac{2\pi i}{N}(N-1)} & e^{-\frac{2\pi i}{N}2(N-1)} & e^{-\frac{2\pi i}{N}3(N-1)} & \dots & e^{-\frac{2\pi i}{N}(N-1)^2} \end{pmatrix}$$

Формула вычисления коэффициентов:

$$A(m,n) = \exp\left(-2\pi i \frac{(m-1)(n-1)}{N}\right)$$

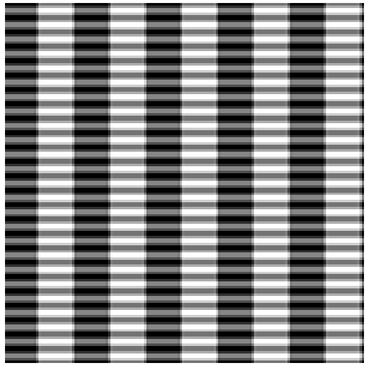
Быстрое преобразование Фурье (FFT)

- Для вычисления всех коэффициентов через скалярное произведение требуется примерно N^2 умножений: очень много при больших длинах сигнала N.
- Быстрое преобразование Фурье (БПФ, FFT) – ускоренный алгоритм вычисления ДПФ
  - Основан на периодичности базисных функций (много одинаковых множителей)
  - Математически точен (ошибки округления даже меньше, т.к. меньше число операций)
  - Число умножений порядка $N \cdot \log_2 N$, намного меньше, чем N^2
  - Ограничение: большинство реализаций FFT принимают только массивы длиной $N = 2^m$
- Есть и быстрое обратное преобразование

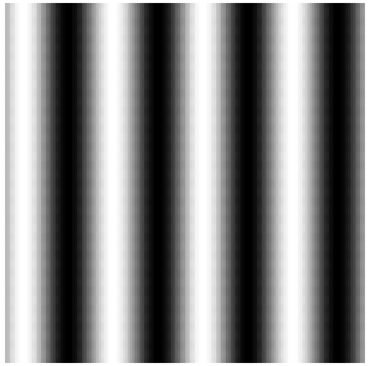
J. W. Cooley and J. W. Tukey, “An algorithm for the machine calculation of complex Fourier series,” *Mathematics of Computation, vol. 19*, pp. 297–301, 1965.

Фильтр Гаусса

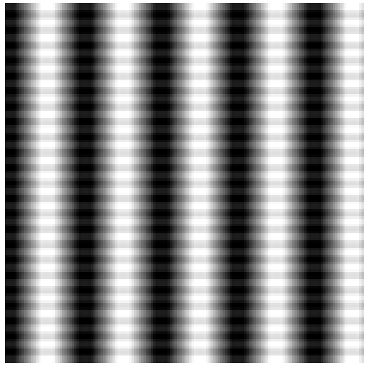
Результат свертки фильтром гаусса и усреднения



Исходное изображение



Фильтр Гаусса с Sigma = 4



Усреднение по 49 пикселям (7x7)

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

Выводы

- Переход от представления в виде регулярной сетки к частотному представлению позволяет учесть структуру изображения

Сжатие изображений по алгоритму JPEG

Использование теоремы о свёртке позволяет эффективнее фильтровать изображение

Фильтр Гаусса – фильтр низких частот

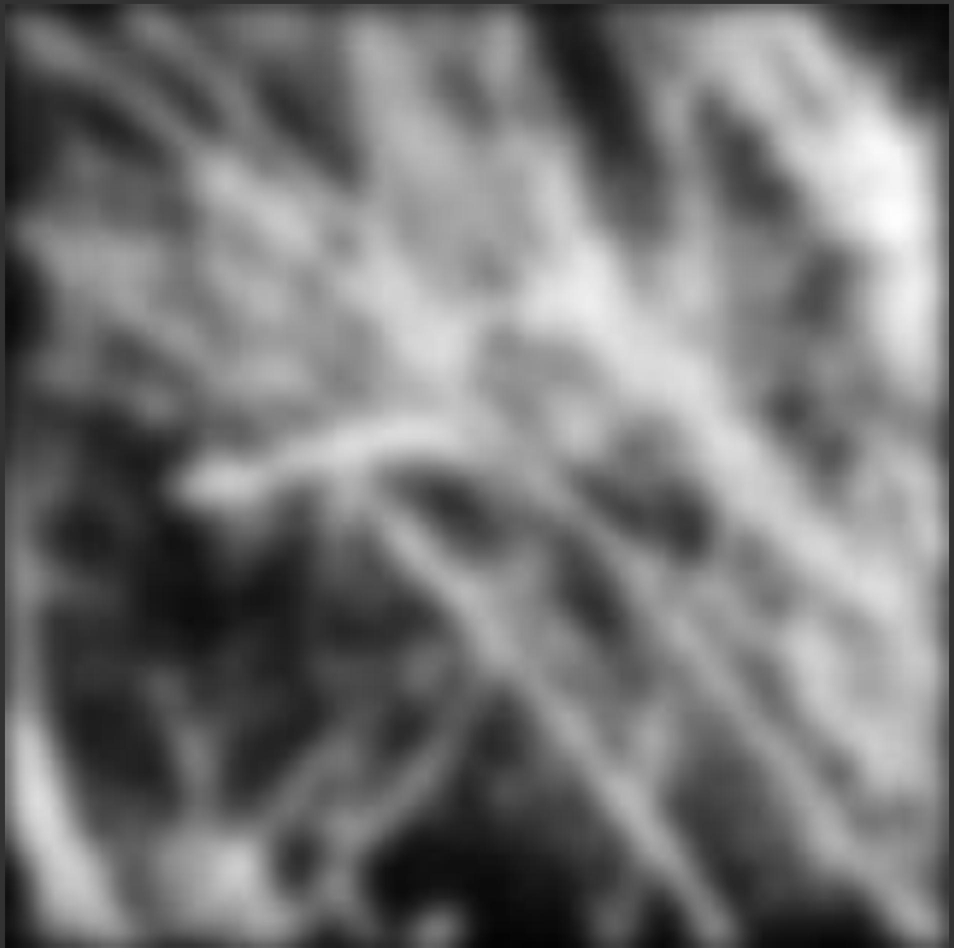
- Есть и другие виды представления изображений

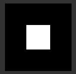
На основе вейвлет-разложения

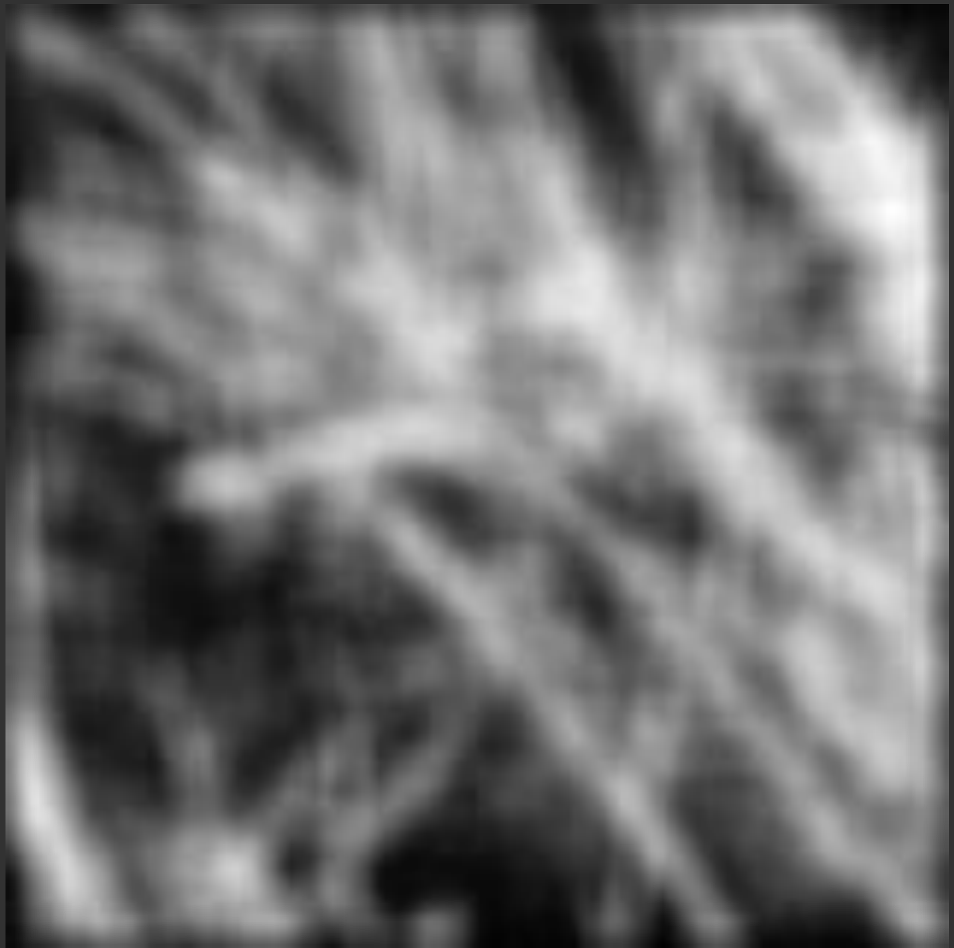
Разреженные представления на основе словаря

Почему Гаусс дает такой гладкий результат, а Box filter – нет?

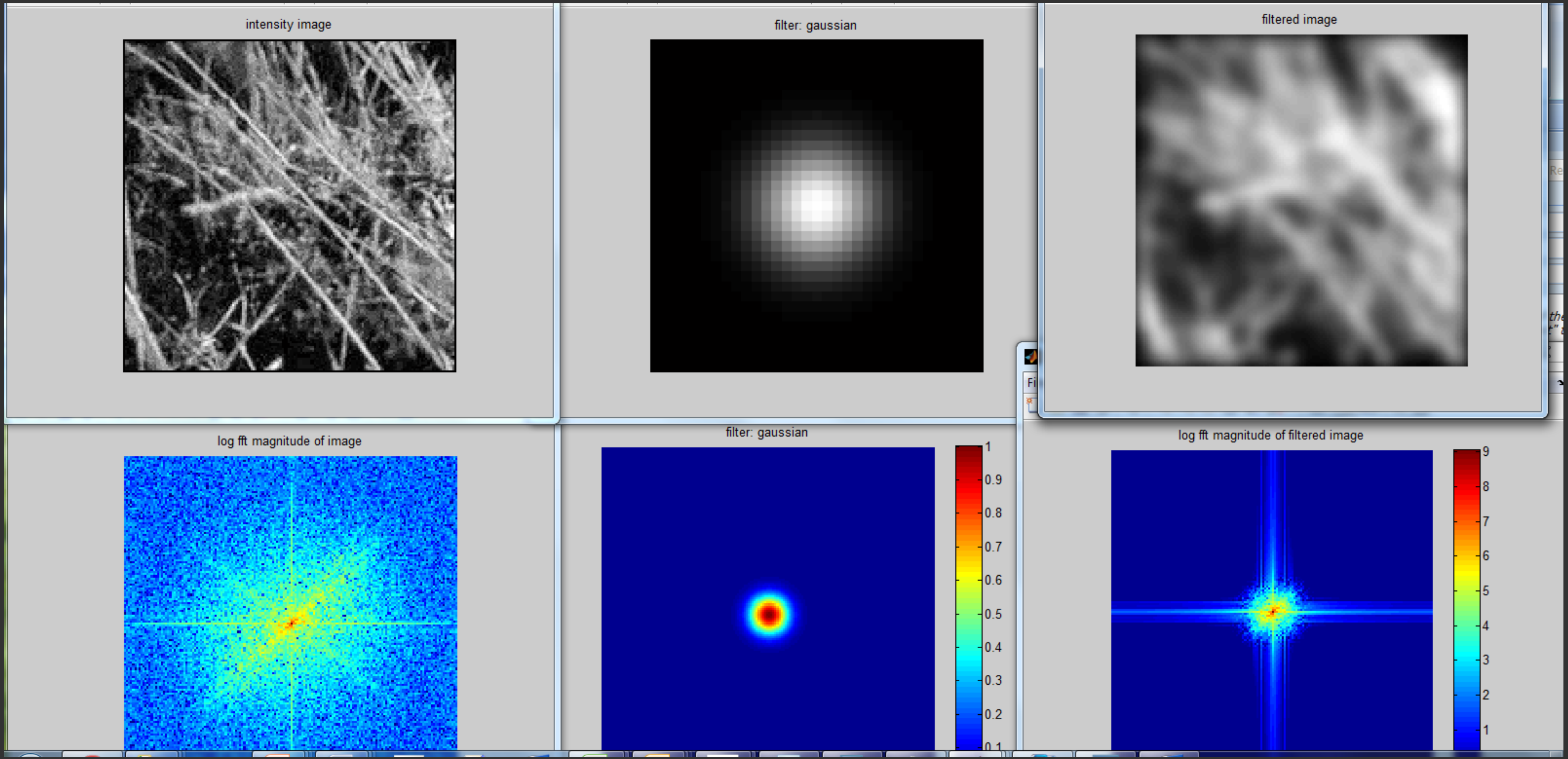
Gaussian 



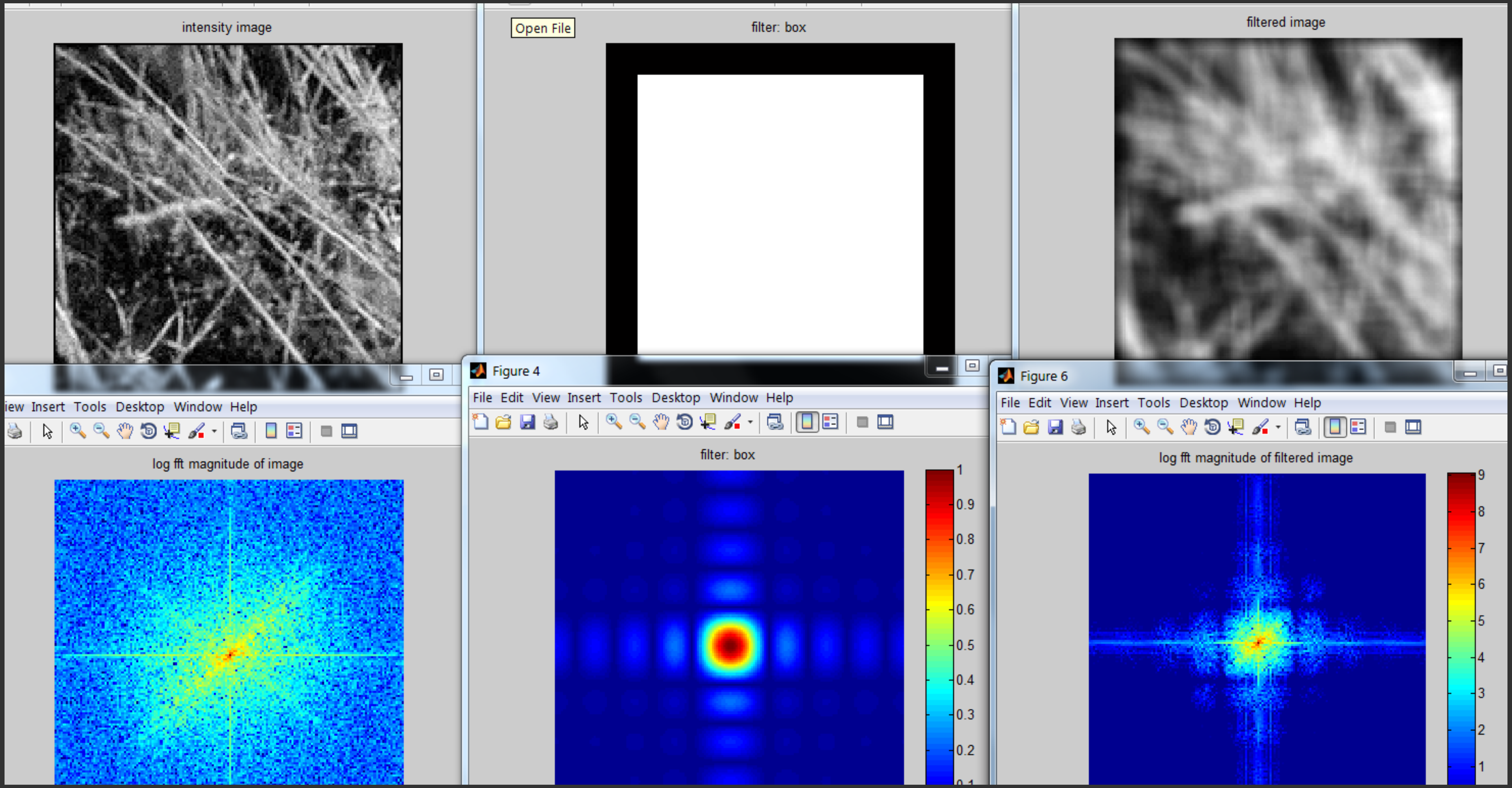
Box filter 



Гауссиан


intensity image
filter: gaussian
filtered image
log fft magnitude of image
filter: gaussian
log fft magnitude of filtered image

Box Filter



intensity image
Open File
filter: box
filtered image
Figure 4
Figure 6
File Edit View Insert Tools Desktop Window Help
log fft magnitude of image
filter: box
log fft magnitude of filtered image

СЕГМЕНТАЦИЯ ИЗОБРАЖЕНИЙ

Из чего состоит изображение?

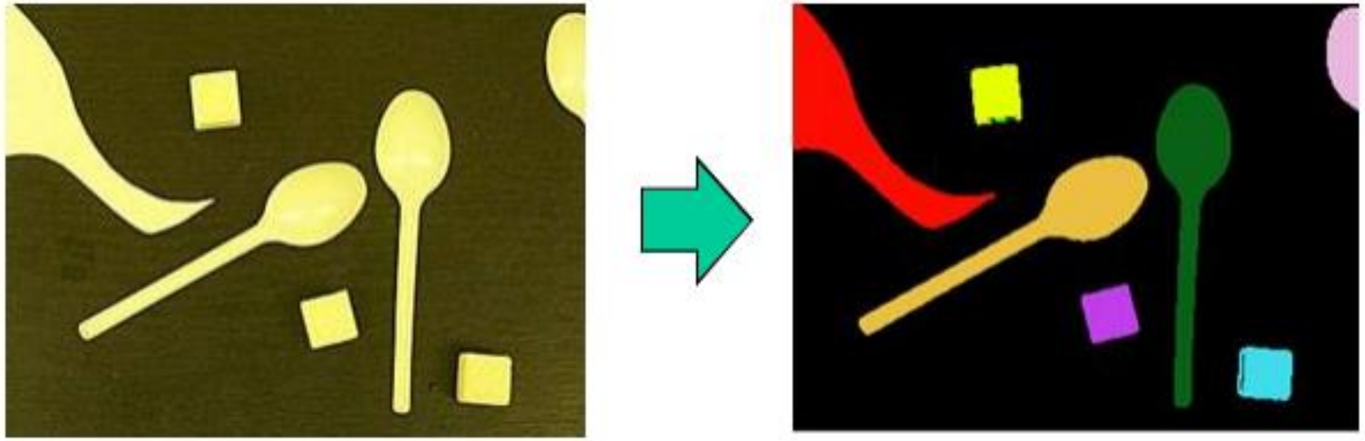
Из сегментов – отдельных объектов

Сегментация

- Сегментация - это способ разделения сцены на «куски», с которыми проще работать
- Тесселяция - разбиение изображения на неперекрывающиеся области, покрывающие все изображение и однородные по некоторым признакам
- Можно и по другому сегментировать изображение

• Пересекающиеся области

• Иерархическое представление

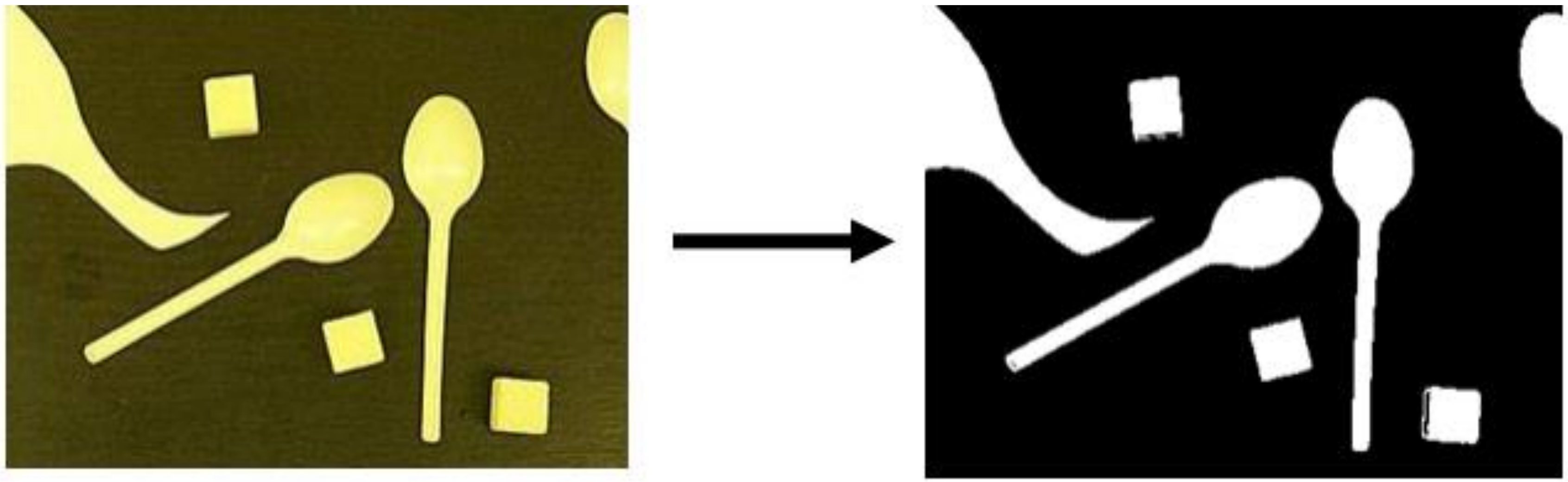


Простейшая сегментация

Чем отличаются объекты на этом изображении?

- Все объекты яркие, фон тёмный
- Для сегментации такого изображения нам достаточно:
  - пороговая бинаризация
  - обработки шума
  - выделения связанных компонент

Пороговая бинаризация



- **Пороговая фильтрация** (thresholding)

Пиксели, которых выше/ниже некоторого порога, заданного «извне», помечаются 1

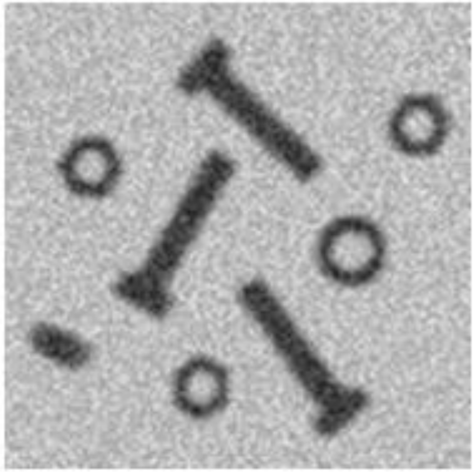
Ниже порога помечаются 0

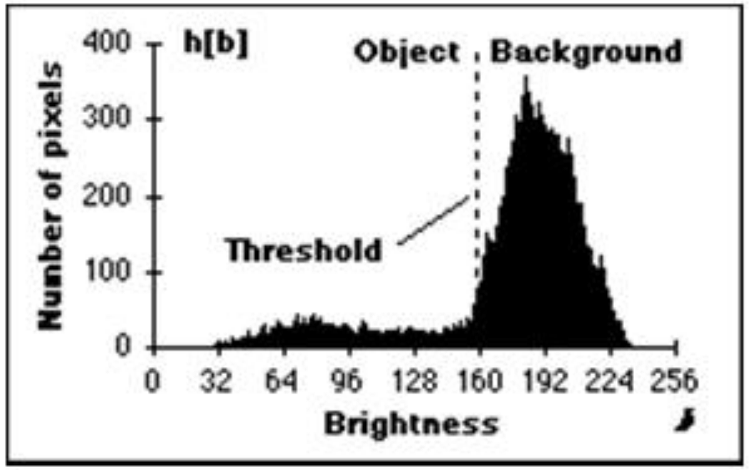
- Бинарное изображение – пиксели которого могут принимать только значения 0 и 1
- Бинаризация - построение бинарного изображения по полутоновому / цветному

Пороговая фильтрация

- Более интересный способ – определение порога автоматически, по характеристикам изображения

Анализ гистограммы





Анализ гистограммы

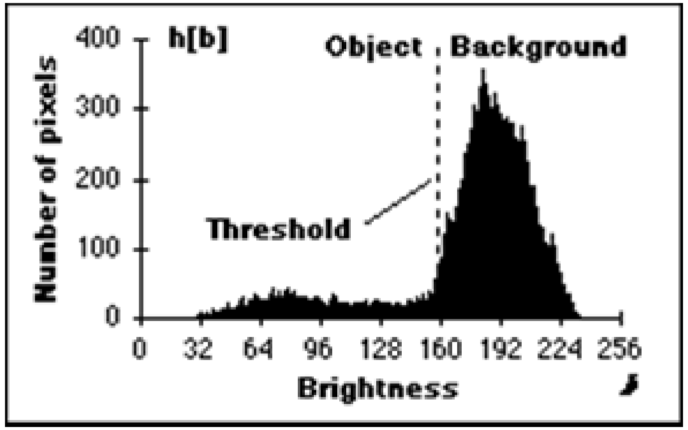
- Анализ симметричного пика гистограммы применяется, когда фон изображения дает отчетливый и доминирующий пик гистограммы, симметричный относительно своего центра.
- Что делать?

1. Сгладить гистограмму;

2. Найти ячейку гистограммы h_{max} с максимальным значением;

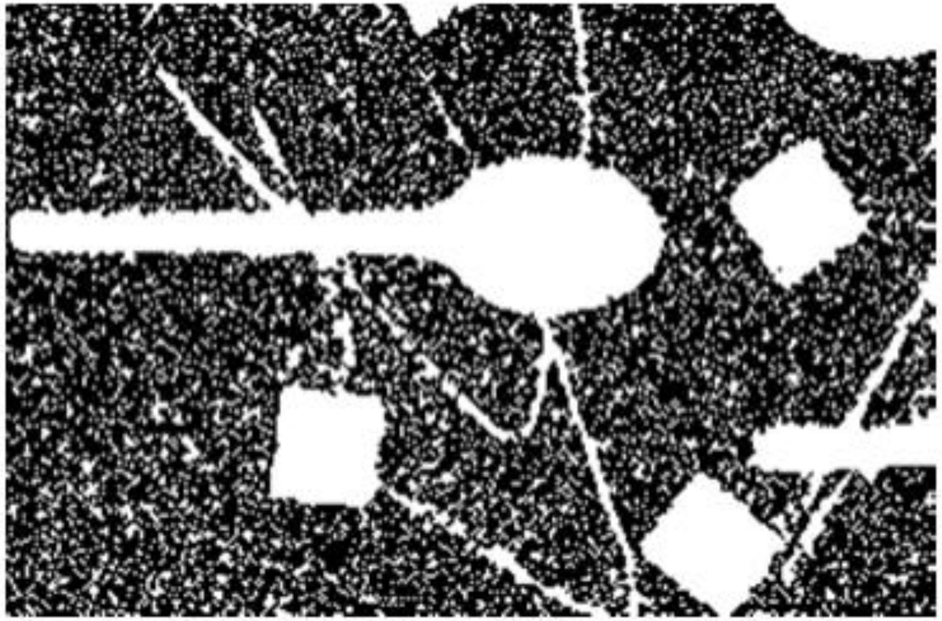
3. На стороне гистограммы не относящейся к объекту (на примере – справа от пика фона) найти яркость h_p, количество пикселей с яркостью >= h_p равняется p% (например 5%) от пикселей яркости которых >= h_{max};

4. Пересчитать порог $T = h_{max} - (h_p - h_{max})$;



Шум в бинарных изображениях

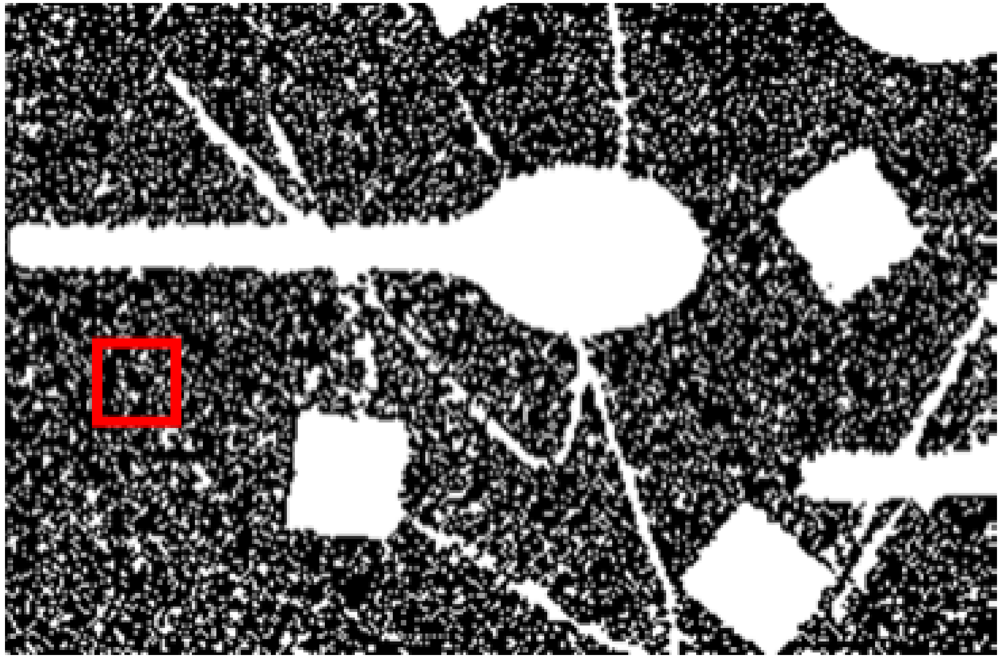
Пример бинарного изображению с сильным шумом



Часто возникает из-за невозможности полностью подавить шум в изображениях, недостаточной контрастности объектов и т.д.

Шум в бинарных изображениях

- По одному пикселю невозможно определить – шум или объект?
- Нужно рассматривать окрестность пикселя!



Подавление и устранение шума

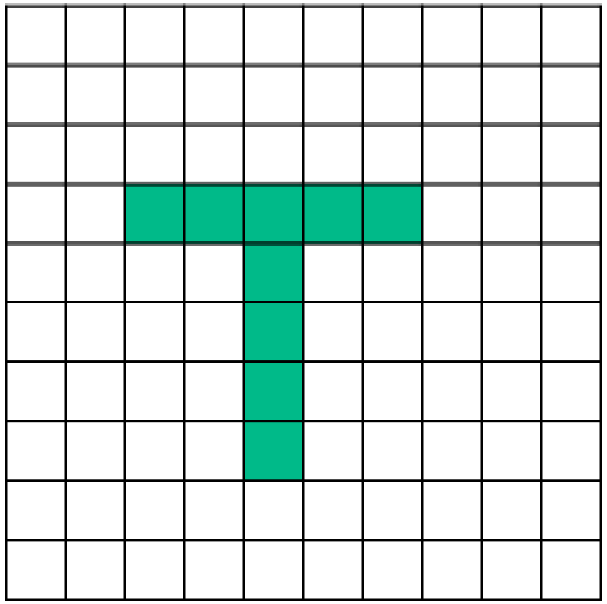
Широко известный способ - устранение шума с помощью операций математической **морфологии**:

- Сужение (erosion)
- Расширение (dilation)
- Закрытие (closing)
- Раскрытие (opening)

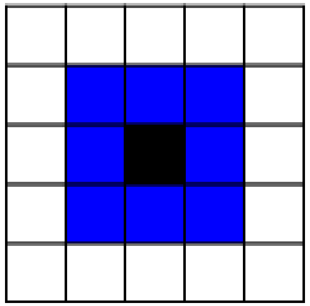
МАТЕМАТИЧЕСКАЯ МОРФОЛОГИЯ

Математическая морфология

A

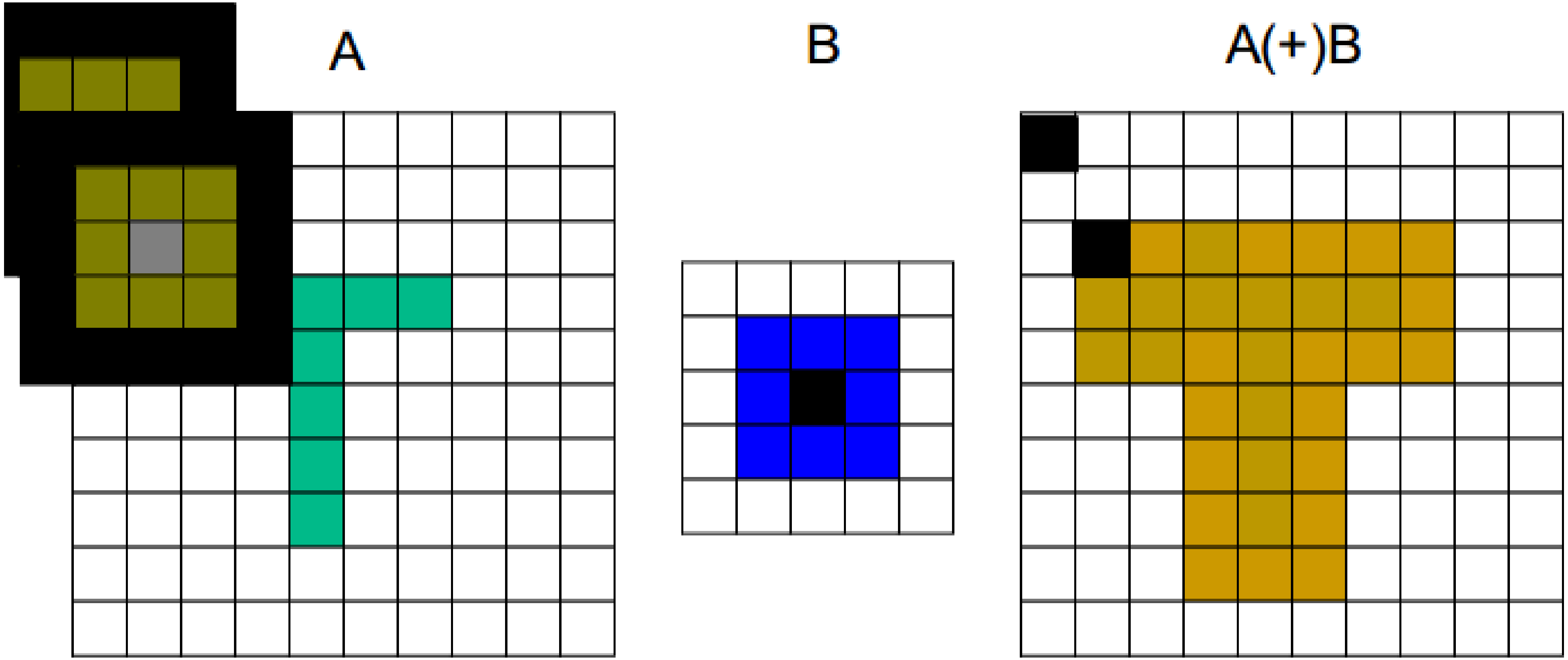


B



- Множество A обычно является объектом обработки
- Множество B (называемое структурным элементом) – инструмент обработки

Расширение в дискретном случае

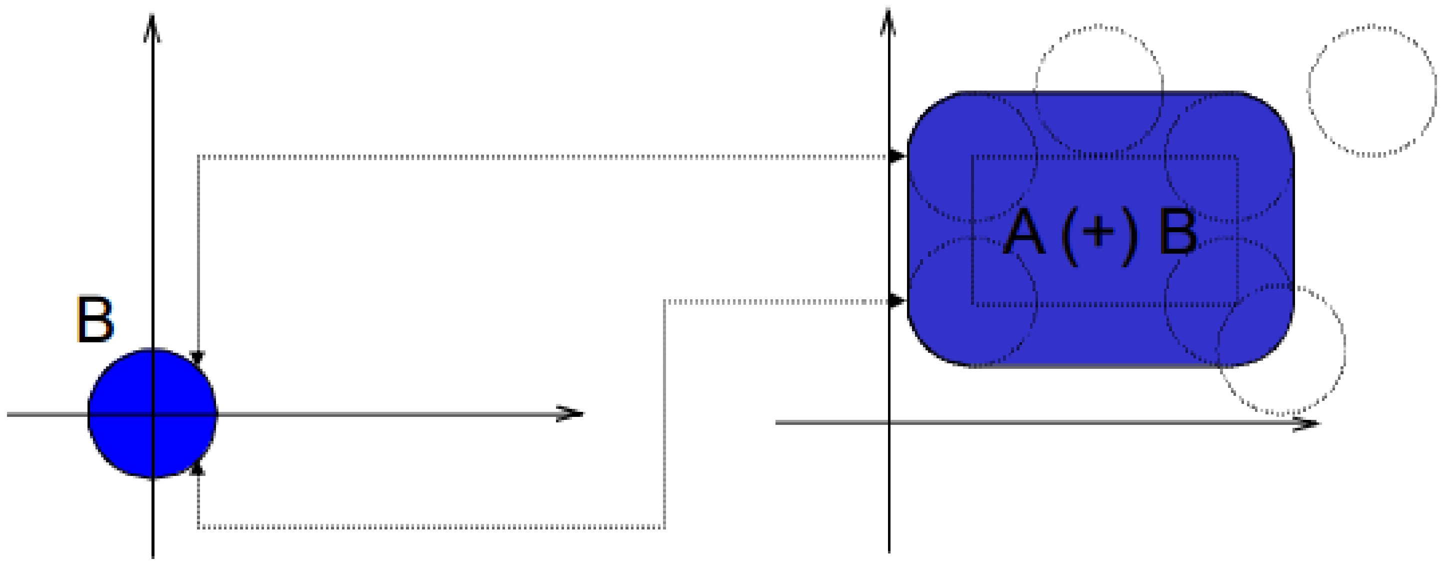


Операция «расширение» - аналог логического «или»

Расширение

Расширение (dilation)

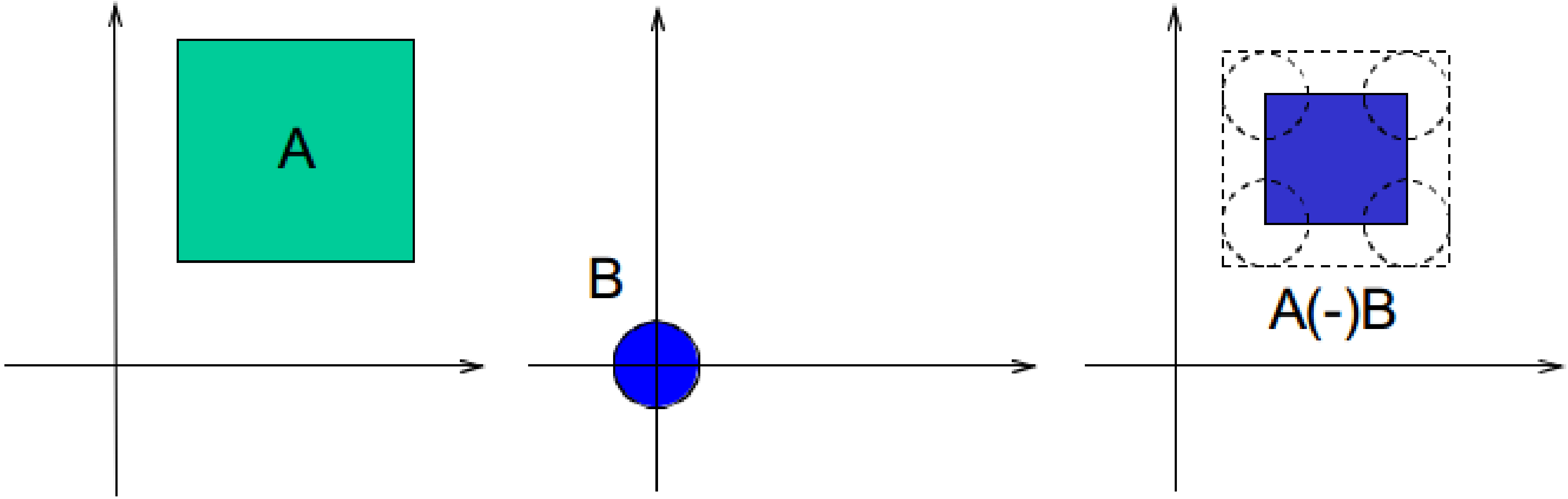
A (+) B = {t ∈ R^2: t = a + b, a ∈ A, b ∈ B}



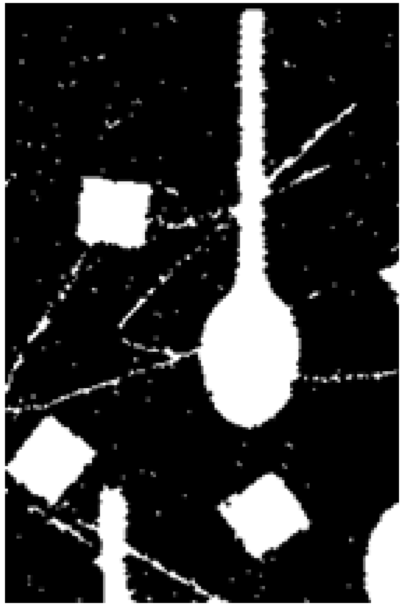
Сужение

Сужение (erosion)

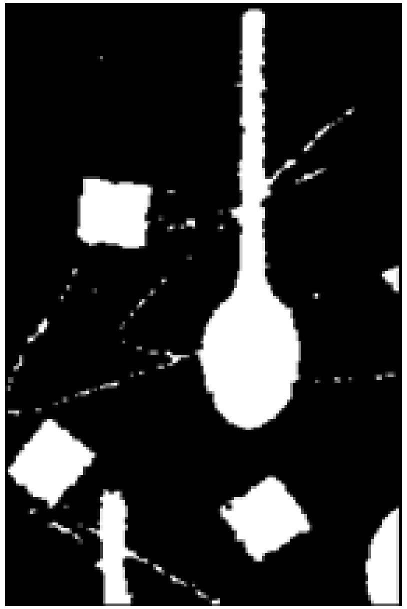
A (-) B = (A^C (+) B)^C, где A^C – дополнение A



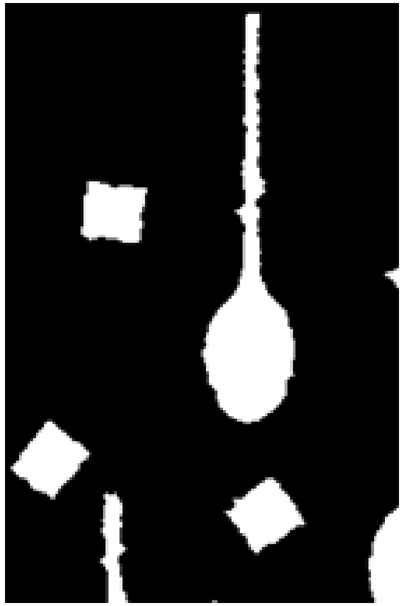
Результат



$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & [1] & 1 \\ 0 & 1 & 0 \end{bmatrix}$$



$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & [1] & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & [1] & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Achtung!

Осторожно...

Результат морфологических операций во многом определяется применяемым структурным элементом.

Выбирая различный структурный элемент можно решать разные задачи обработки изображений:

- Шумоподавление
- Выделение границ объекта
- Выделение скелета объекта
- Выделение сломанных зубьев на изображении шестерни

- Морфологическое раскрытие (opening)

  open(A, B) = (A (-) B) (+) B

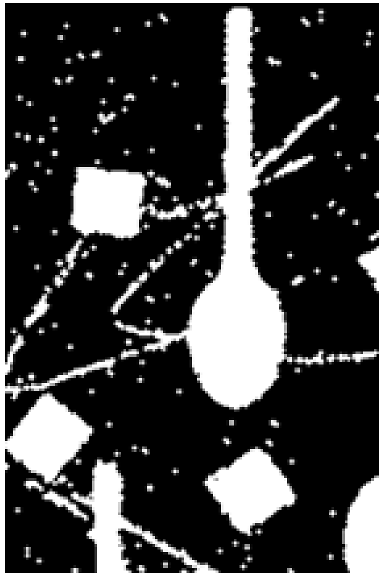
- Морфологическое закрытие (closing)

  close(A, B) = (A (+) B) (-) B

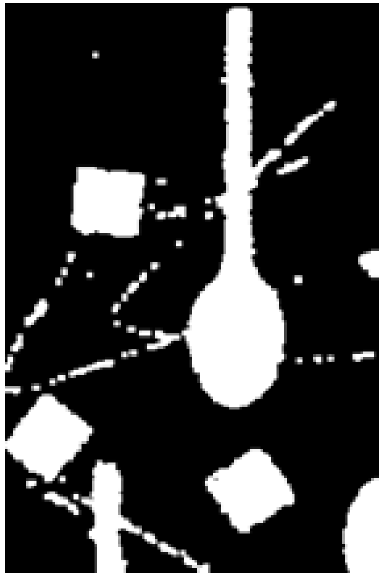
- Что эти операции делают?

Применение открытия

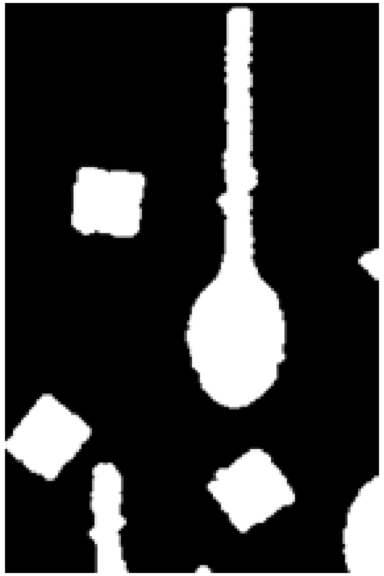
Применим операцию открытия к изображению с сильным шумом:



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

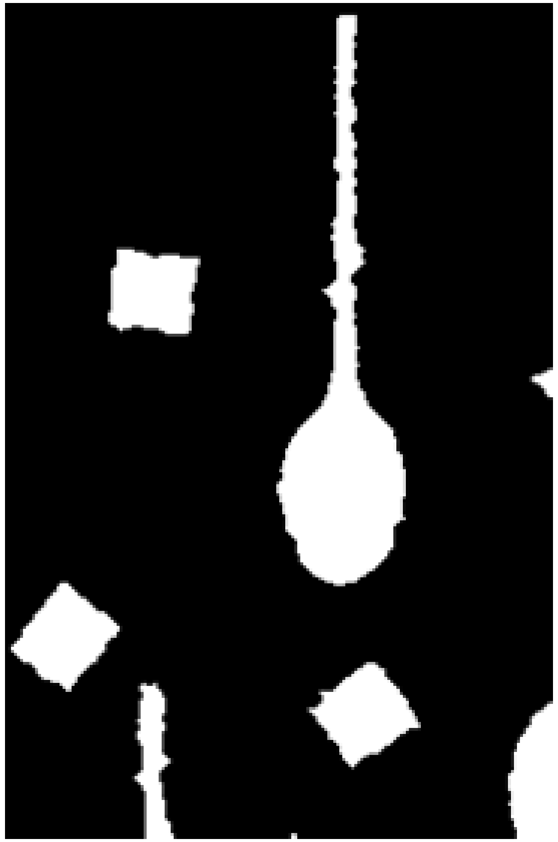


$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



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

Сужение VS Открытие



Сужение

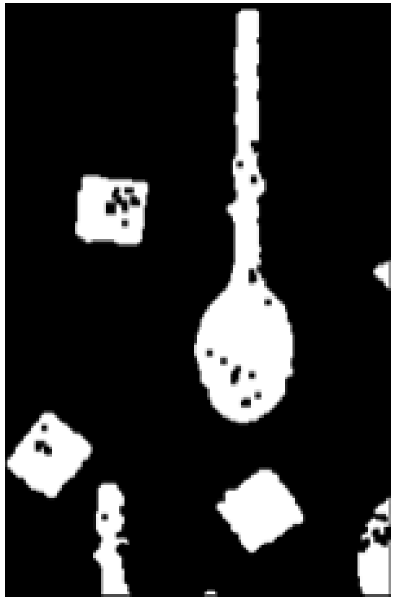
Открытие

Дефекты бинаризации

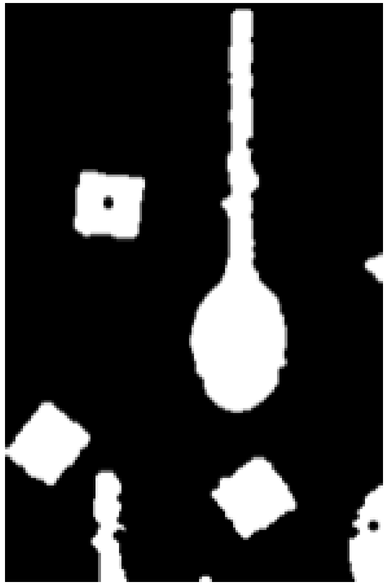
Пример бинарного изображению с дефектами распознаваемых объектов

Применение закрытия

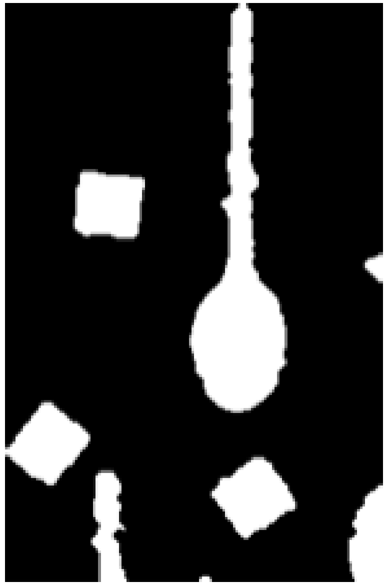
Применим операцию закрытия к изображению с дефекиами объектов:



$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



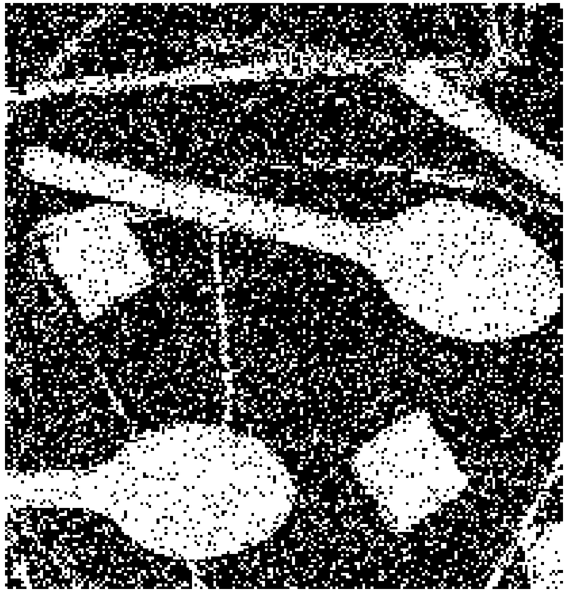
$$\begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$



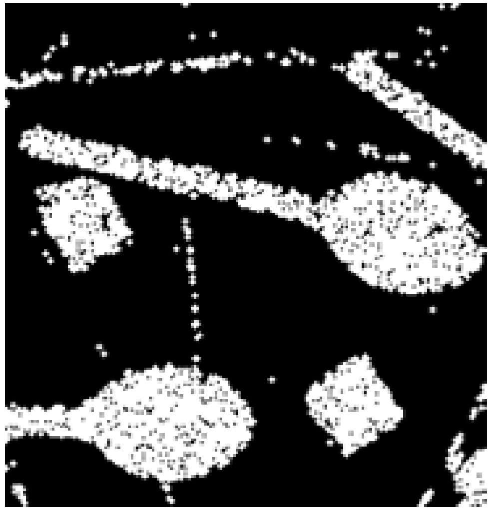
$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

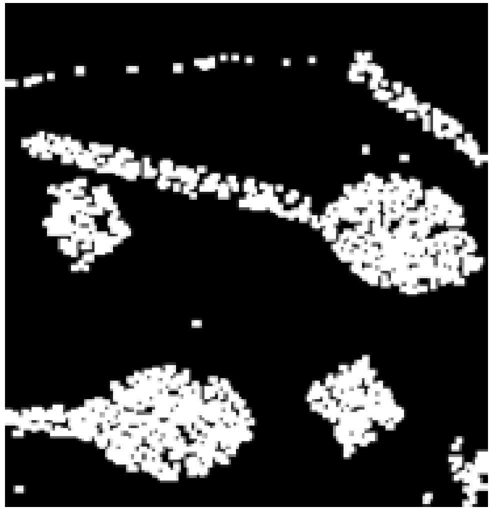
Не лучший день для морфологии…

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



FAIL операции открытия



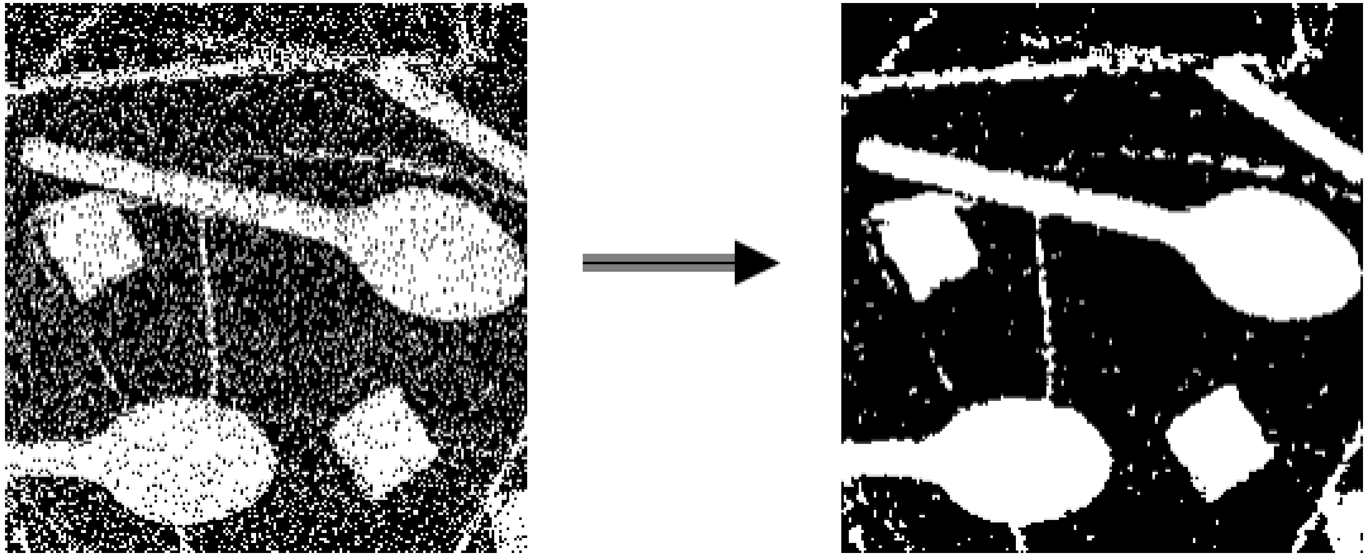


$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 &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

Часто помогает медианная фильтрация!

Медианный фильтр

Фильтр с окрестностью 3x3



Теперь можем с помощью морфологии убрать оставшиеся точки, тонкие линии и т.д.

В следующих сериях…

Выделение однородных областей (реальная сегментация)