

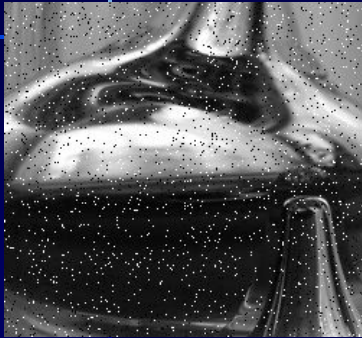


Лектор: Лукин Алексей Сергеевич

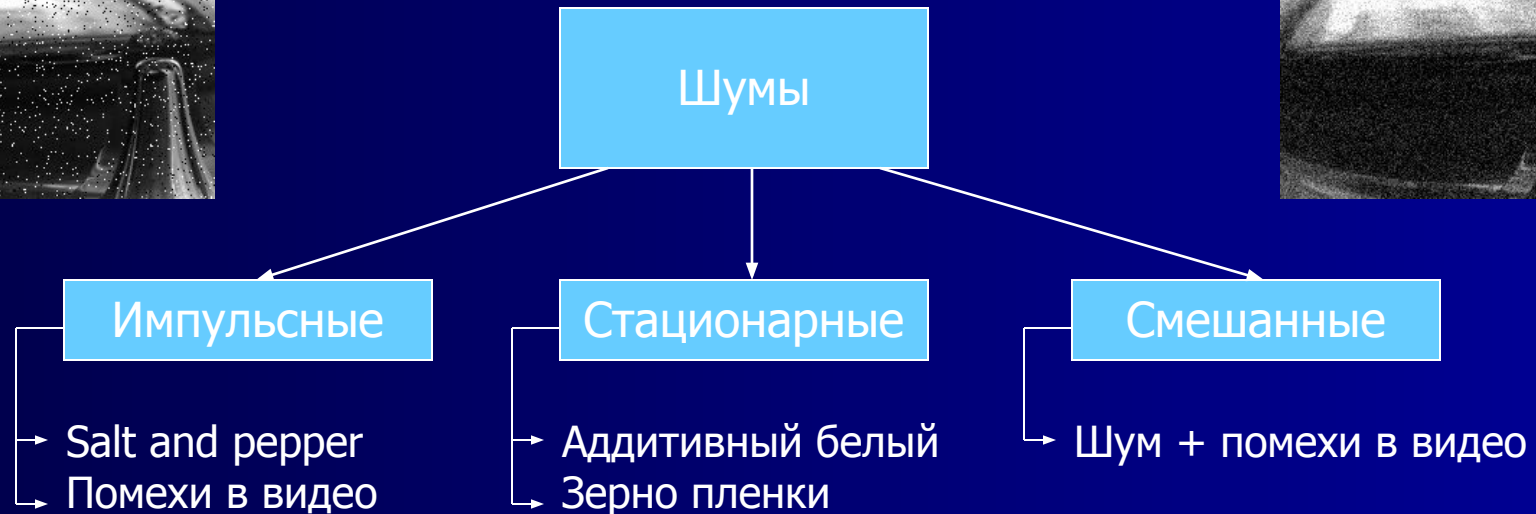
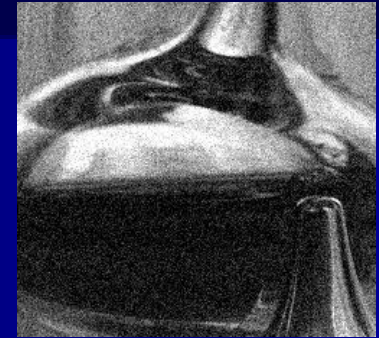
# Шумоподавление для изображений

# Виды и примеры шумов

Salt and pepper



AWGN

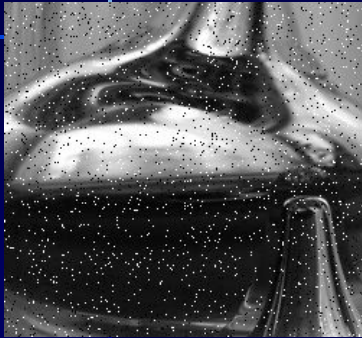


**Белый шум** – пиксели случайны, не коррелированы друг с другом  
**Гауссов/равномерный/треугольный шум** – закон распределения амплитуд пикселей

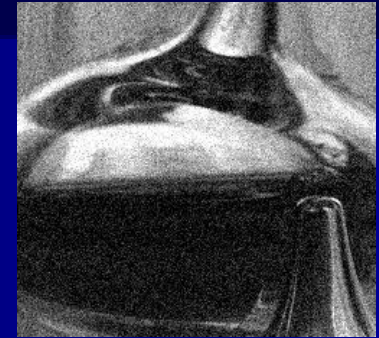
**Аддитивный шум** – прибавляется к «чистому» изображению и не зависит от него

# Методы шумоподавления

Salt and pepper



AWGN



Шумы

Импульсные

Стационарные

Смешанные

Медианный фильтр  
Взвешенная медиана  
Ранговые фильтры

Bilateral filter  
Non-Local Means  
Wavelet thresholding  
DCT, PCA, ICA  
Анизотропная диффузия  
Алгоритм BM3D

Ранговые фильтры  
Комбинированные методы

# Импульсные шумы

- Медианная фильтрация
  - ▶ Центральный элемент отсортированного массива яркостей
  - ▶ Как быть с цветными изображениями?



медиана 3x3

# Импульсные шумы

- Медианная фильтрация
  - ▶ Центральный элемент отсортированного массива яркостей
  - ▶ Как быть с цветными изображениями?



медиана 7x7

# Импульсные шумы

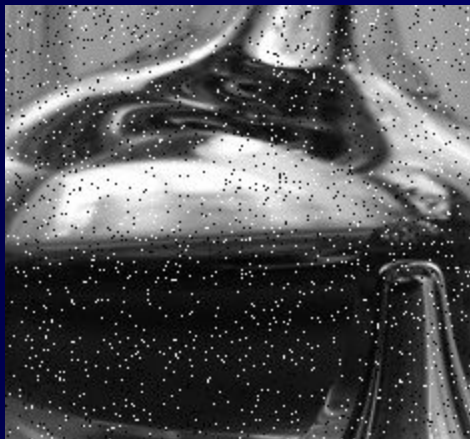
- Медианная фильтрация
  - ▶ Центральный элемент отсортированного массива яркостей
  - ▶ Как быть с цветными изображениями?



медиана 15x15

# Импульсные шумы

- Т.к. искажена лишь малая часть пикселей, то не надо фильтровать все изображение!
  1. Детектирование искаженных пикселей (простейший способ – анализ разности изображения с его медианной фильтрацией)
    - ▶ Фильтрация только искаженных пикселей



медиана 3x3



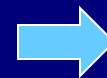
адаптивная медиана

# Стационарные шумы



- Простейшие методы

- ▶ Размытие изображения – вместе с шумом размывает детали



- ▶ Размытие в гладких областях – остается шум вблизи границ
- ▶ Медианная фильтрация – хорошо подавляет импульсный шум, но удаляет мелкие детали



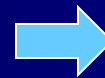
# Bilateral filter

- Адаптивные алгоритмы
  - ▶ Bilateral filter
    - усреднение окружающих пикселей
    - с весами

$$y_{i,j} = \sum_{k,m \in Q} x_{i+k,j+m} \cdot W(i,j,k,m)$$

$$W(i,j,k,m) \approx \exp\left(-\frac{(x_{i,j} - x_{i+k,j+m})^2}{h^2}\right) \cdot \exp\left(-\frac{k^2 + m^2}{\rho^2}\right)$$

фотометрическая близость      пространственная близость



# Bilateral filter



- Bilateral filter: художественное применение



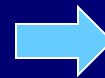
(при слишком сильном действии)

# Non-Local Means

- Адаптивные алгоритмы
  - ▶ Non-local means (NL-means) – веса зависят от близости целых блоков, а не отдельных пикселей

$$W(i, j, k, m) \approx \exp\left(-\frac{\|v(x_{i,j}) - v(x_{i+k, j+m})\|^2}{h^2}\right)$$

$v(x_{i,j})$  – блок вокруг пикселя  $x_{i,j}$



# Non-Local Means

- Вычисление весов

Веса высоки для  $q1$  и  $q2$ ,  
но не для  $q3$

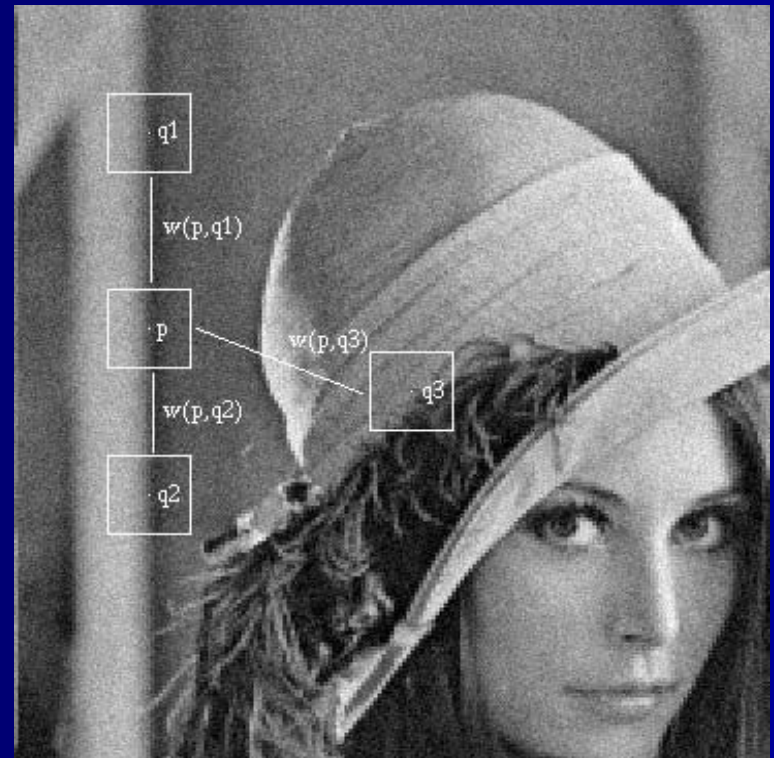


Иллюстрация из  
Buades et al 2005



▶ Способен сохранять текстуру изображения лучше,  
чем bilateral filter

# Non-Local Means

- Достоинства и недостатки:



- ▶ Высокое качество результирующего изображения



- ▶ В исходном варианте – очень высокая вычислительная сложность

- Ускоряющие расчет оптимизации:

- ▶ Использование команд MMX/SSE для сравнения блоков
- ▶ Разбиение изображения на блоки и усреднение целых блоков, а не отдельных пикселей
- ▶ Промежуточный вариант между усреднением блоков и усреднением пикселей: усреднение маленьких блоков

# Non-Local Means



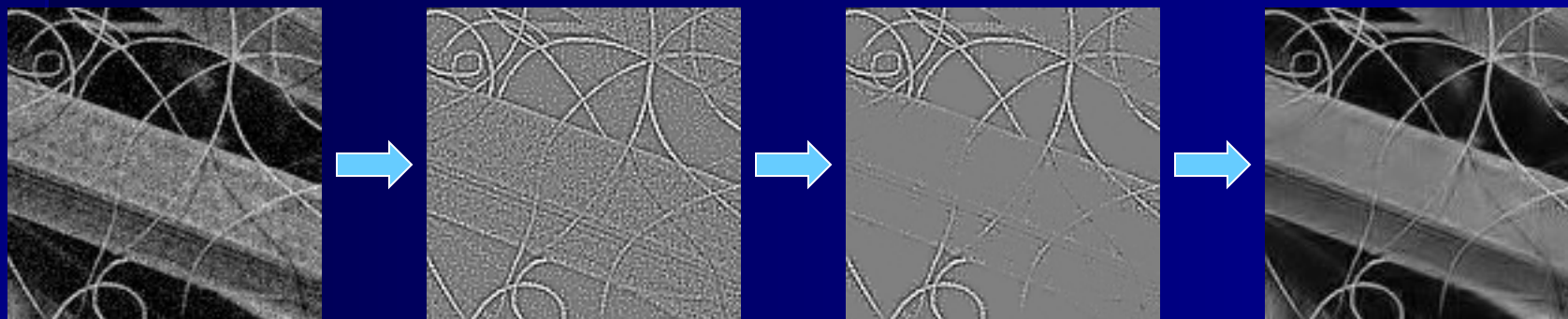
- Применение к видео
  - ▶ Область поиска блоков можно расширить на соседние кадры (сделать ее пространственно-временной)
  - ▶ Для ускорения просчета можно применять сравнение блоков только по Y-каналу в цветовой модели YCrCb (YUV)

# Вейвлетный метод



- Вейвлетное шумоподавление для изображений

1. DWT
2. Оценка уровня и спектра шума
3. Подавление вейвлет-коэффициентов (thresholding, shrinkage)
4. Обратное DWT



Подавление шума различного масштаба



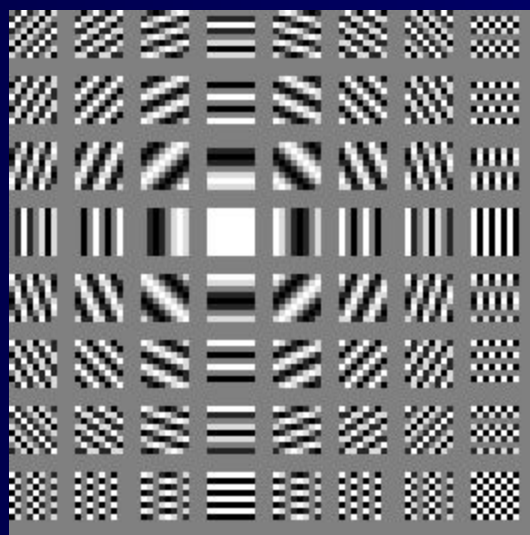
Отсутствие инвариантности к сдвигу

Плохая локализация энергии для наклонных границ

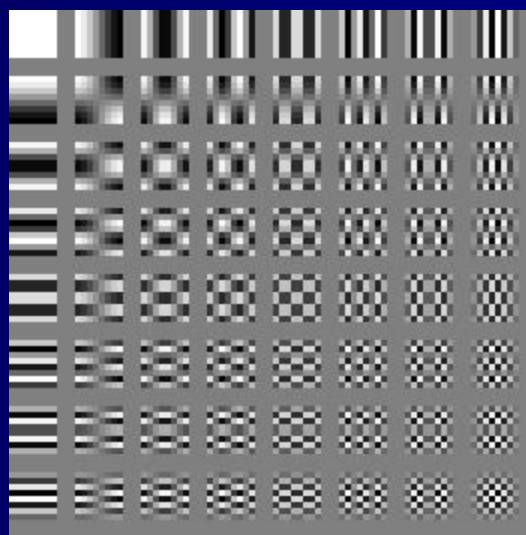
# Тригонометрические базисы



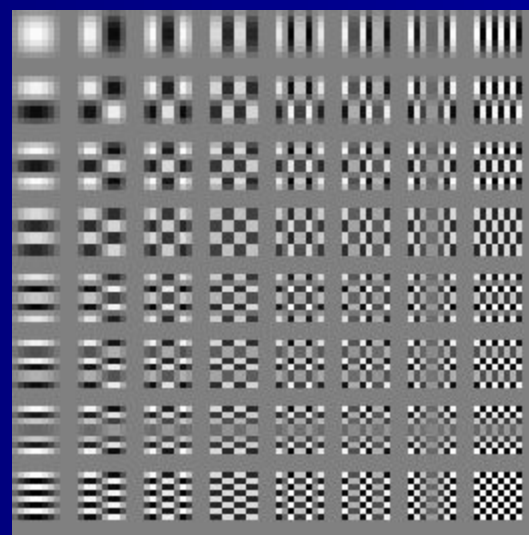
- Дискретные преобразования: Фурье, косинусное и синусное



DFT



DCT



DST



Быстро вычисляются



Зависимость от размера блока: эффект Гиббса либо плохое подавление НЧ компонент шума



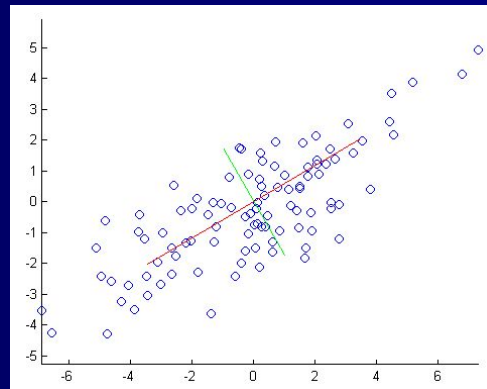
# Метод главных компонент

## КОМПОНЕНТ



### Principal Component Analysis (PCA, KLT)

- Идея: так выбрать базисные вектора, чтобы минимальным числом базисных векторов можно было наилучшим образом приблизить данные из некоторого набора.
- PCA – ортогональное линейное преобразование базиса, при котором первый вектор нового базиса соответствует направлению максимальной дисперсии данных, второй вектор – следующему направлению максимальной дисперсии и т.д.



размерность данных = 2

# Метод главных компонент

## КОМПОНЕНТ



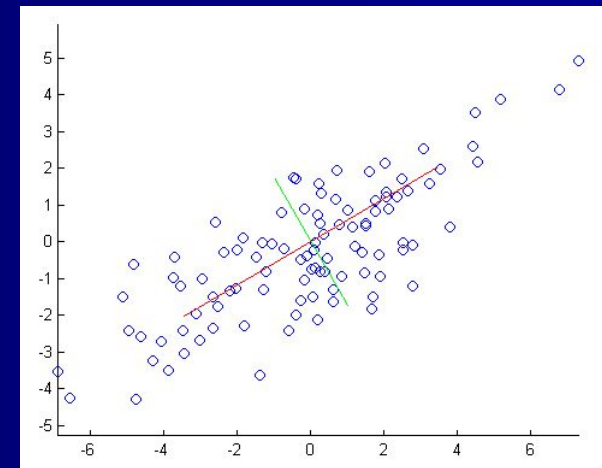
### Principal Component Analysis (PCA, KLT)

- Решение: базисные вектора – собственные вектора  $e_i$ , матрицы ковариации  $C_x$  исходных данных  $x$ :

$$x = (x_1, \dots, x_n)^T$$

$$\mu_x = E\{x\}$$

$$C_x = E\{(x - \mu_x)(x - \mu_x)^T\}$$



на рисунке размерность  $n = 2$

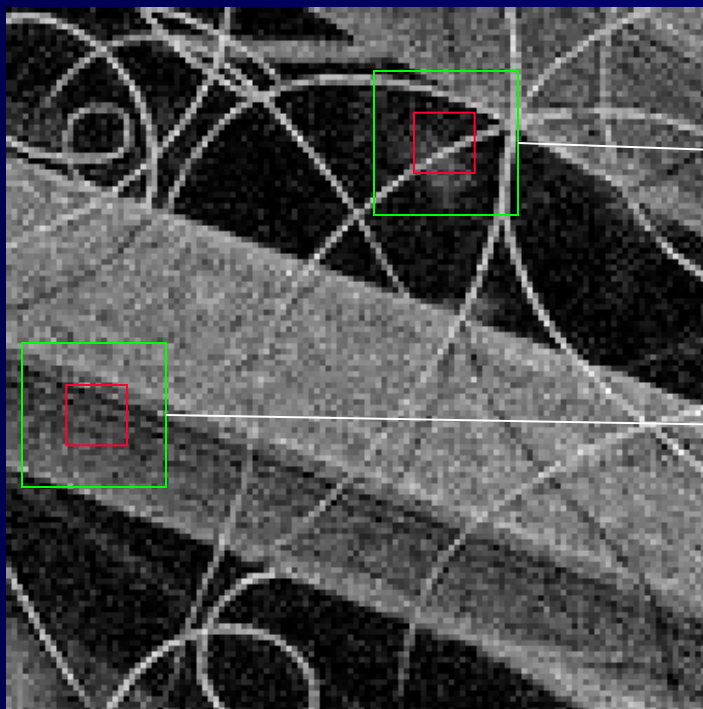
- Альтернативное решение – с помощью сингулярного разложения матрицы исходных данных

# Шумоподавление

## Метод главных компонент

- Применение к блокам изображения 8x8:

$$x = (x_1, \dots, x_{64})^T$$

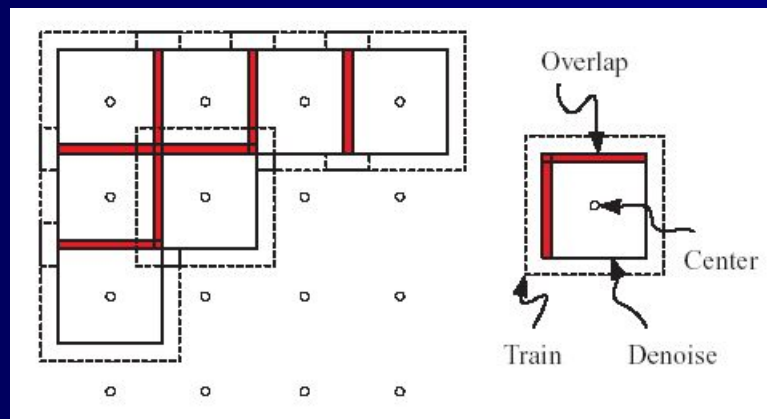


64 базисных вектора

# Шумоподавление

Существующие подходы (*Muresan/Parks, 2003*)

- PCA-денойзинг изображений
  1. Блочное преобразование PCA
  2. Подавление коэффициентов в новом базисе
  3. Обратное преобразование и наложение блоков с перекрытием



Адаптация к деталям изображения (*в т.ч. – к линиям любого наклона*)



При больших блоках – эффект Гиббса,  
при маленьких – не подавляется крупный шум

# Шумоподавление

Результаты



Зашумленное изображение



Вейвлетное шумоподавление

# Шумоподавление

Результаты



Шумоподавление с помощью PCA



Разработанный нами метод