

# **Компьютерное зрение**

Астана. Лекция 5

# На прошлой лекции

- Цифровая обработка сигналов
- Сигналы и системы
- Свертка
- Преобразование Фурье
  - Спектр, высокие и низкие частоты
- Масштабирование изображений

# На лекции

- Информация в изображении
- Предобработка изображения
  - Удаление шума в цветных изображениях
  - Удаление шума в бинарных изображениях
- Математическая морфология
- Анализ информации, содержащейся в изображении

# Обработка и анализ изображений

- Обработка изображений
  - Image processing
- Компьютерное (машинное) зрение
  - Computer (machine) vision
- Компьютерная графика
  - Computer graphics

# Изображение

**Изображение оптическое** – картина, получаемая в результате прохождения через оптическую систему лучей, распространяющихся от объекта, и воспроизводящая его контуры и детали.

*Физический энциклопедический словарь.*

Компьютерное представление изображения:

Функция от двух переменных

Используется дискретное представление

$$I = g(x, y), \{x \in [x_0, x_1], y \in [y_0, y_1]\}$$

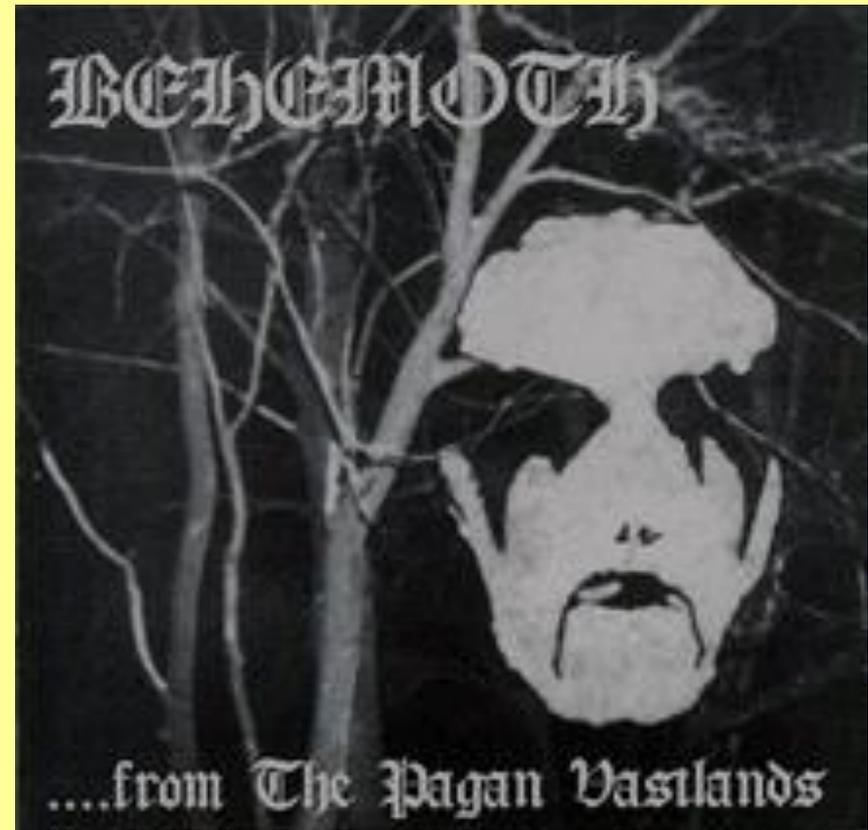
$$I = g(i, j), \{i = \overline{1, n}, j \in \overline{1, m}\}$$

# Компьютерное зрение

- Цель
  - Принятие решений о реальных физических объектах и сценах, основываясь на воспринимаемых изображениях.
- Метод
  - Получение информации из изображений
  - Анализ и интерпретация полученной информации

# Информация, содержащаяся в изображении

- Полезная информация
  - Информация низкого уровня
    - Области однородные по некому признаку
    - Границы однородных областей
    - Цвет, форма области
  - Информация высокого уровня
    - Объекты, содержащиеся на изображении
- Ошибочная информация (шум)



# Предобработка изображения

- Подготовка изображения к анализу
  - Подавление и устранение шума
  - Упрощение изображения
    - например, бинаризация
  - Усиление и выделение полезной информации

# Предобработка изображения

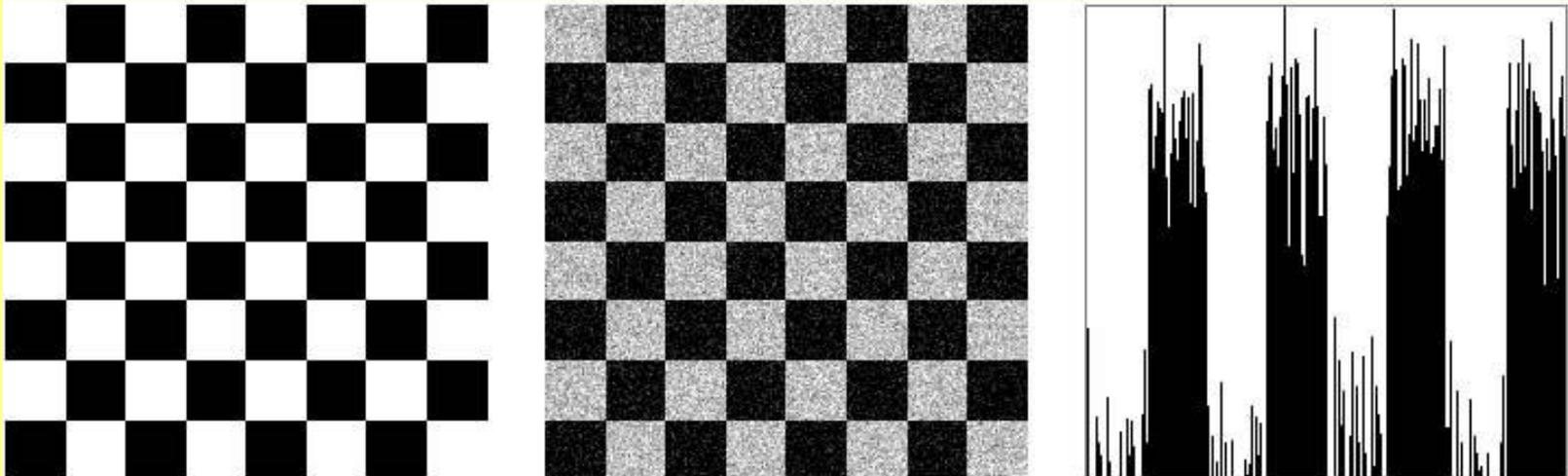
- Подавление и устранение шума
- Причины возникновения шума:
  - Несовершенство измерительных приборов
  - Хранение и передача изображений с потерей данных

# Устранение шума в полутоновых и цветных изображениях

Пример: изображение с равномерным шумом.

$$I(i, j) = g_r(i, j) + noise(i, j)$$

$noise(i, j)$  – нормально распределенная случайная величина.

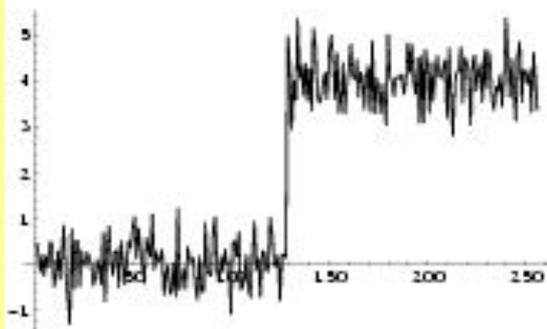


# Усреднение

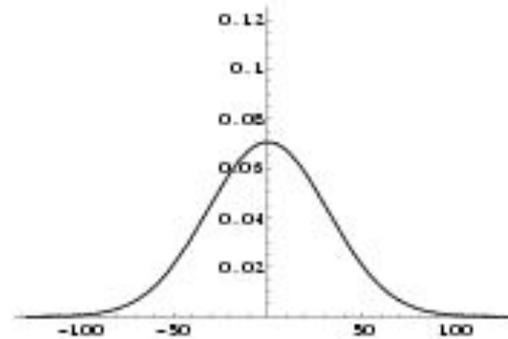
*OutputImage[r, c] = average of some neighborhood of InputImage[r, c]*

$$Out[r, c] = \left( \sum_{i=-2}^{+2} \sum_{j=-2}^{+2} In[r + i, c + j] \right) / 25$$

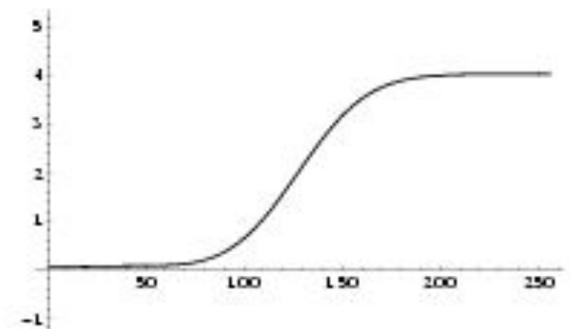
# Гауссиан (sigma=32)



(a) Input  $\phi(x)$



(b) a 1-D Gaussian kernel,  $g(x, \sigma)$   
where  $\sigma = 32$



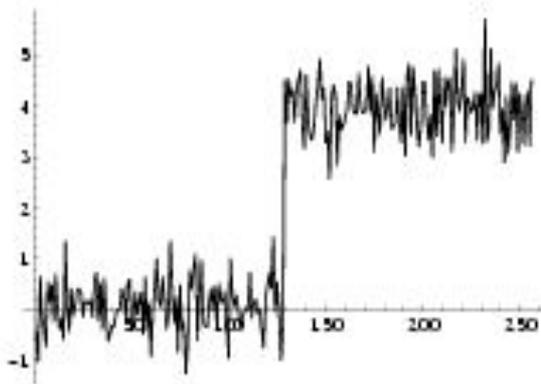
(c) output of  $\phi(x) \otimes g(x, 32)$

# Размытие

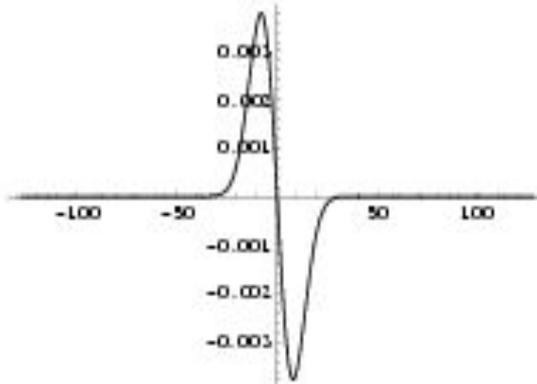


$$\frac{1}{74} \begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 4 & 5 & 4 & 2 \\ 3 & 5 & 6 & 5 & 3 \\ 2 & 4 & 5 & 4 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix}$$

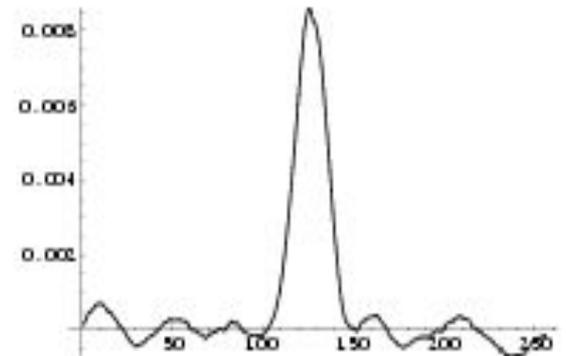
# Свертка - дифференцирование



(a) 1-D Input  $\phi(x)$



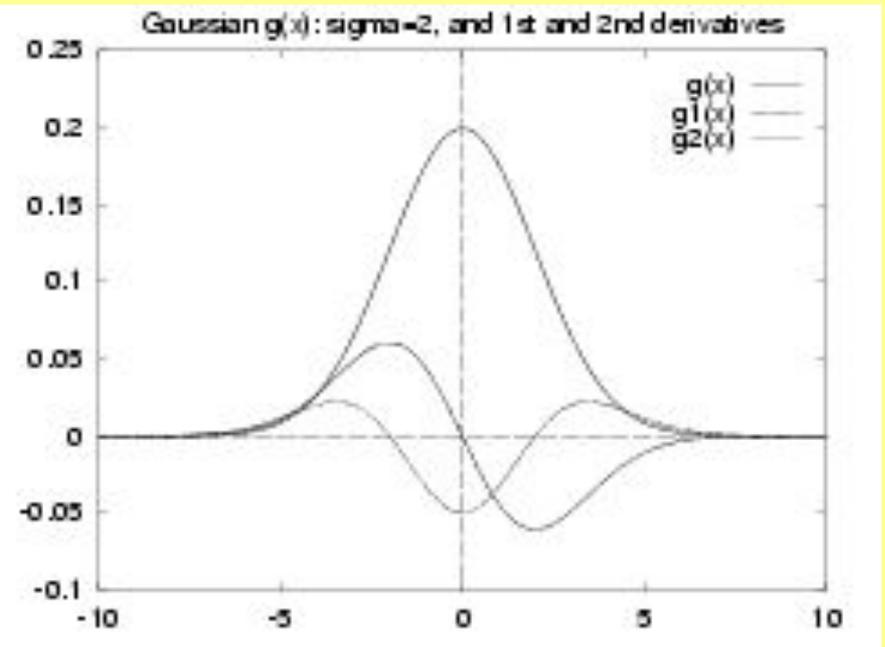
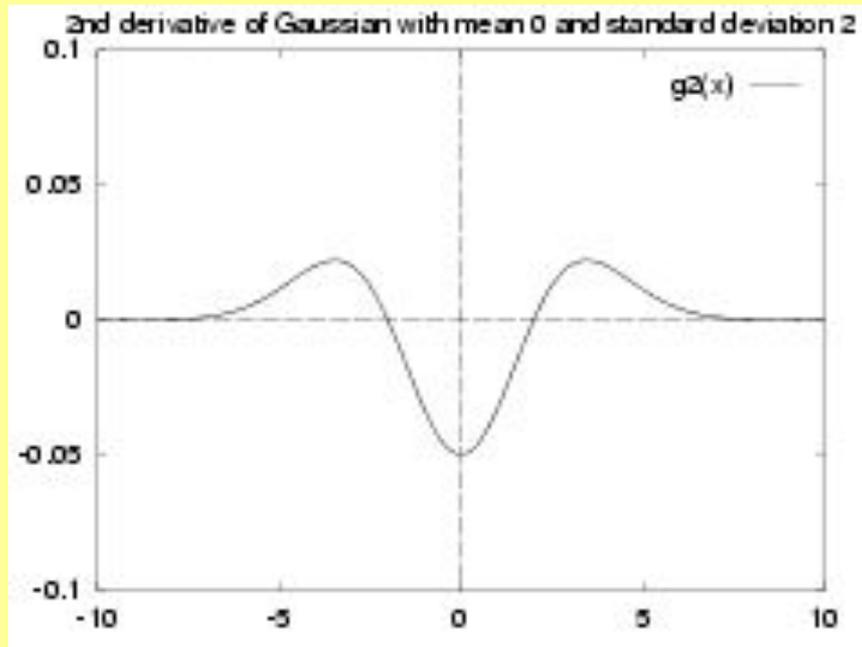
(b)  $\frac{\partial}{\partial x}g(x, \sigma)$  where  $\sigma = 3$



(c)  $\phi(x) \otimes \frac{\partial}{\partial x}g(x, \sigma)$

$$h(x, y, z) \otimes \frac{\partial}{\partial x} \phi(x, y, z) = \frac{\partial}{\partial x} h(x, y, z) \otimes \phi(x, y, z)$$

# Свертка - дифференцирование



# Маски дифференцирования

**Prewitt:**  $M_x =$ 

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

 ;  $M_y =$ 

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

**Sobel:**  $M_x =$ 

-1	0	1
-2	0	2
-1	0	1

 ;  $M_y =$ 

1	2	1
0	0	0
-1	-2	-1

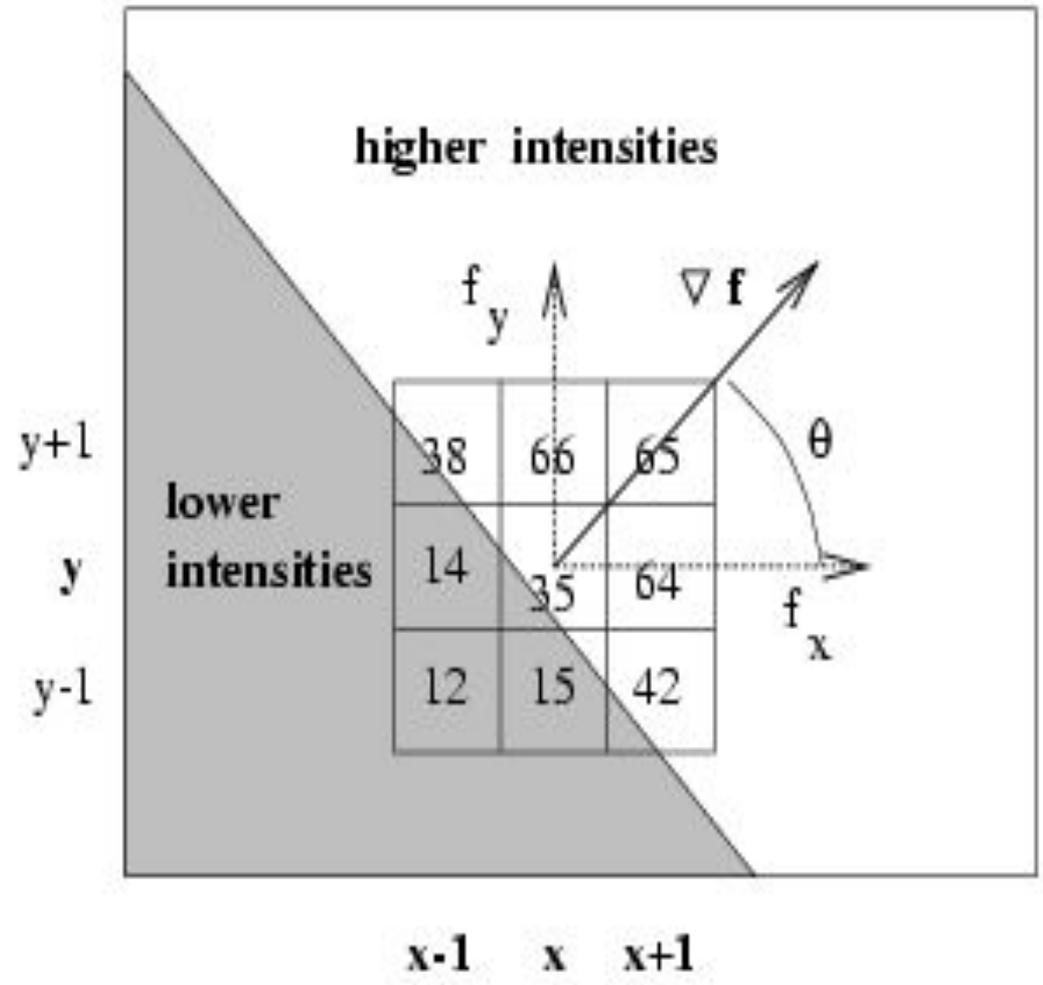
**Roberts:**  $M_x =$ 

0	1
-1	0

 ;  $M_y =$ 

1	0
0	-1

# Градиент



$$f_y = ((38-12)/2 + (66-15)/2 + (65-42)/2) / 3 = (13 + 25 + 11) / 3 = 16$$

$$f_x = ((65-38)/2 + (64-14)/2 + (42-12)/2) / 3 = (13 + 25 + 15) / 3 = 18$$

$$\theta = \tan^{-1}(16 / 18) = 0.727 \text{ rad} = 42 \text{ degrees}$$

$$|\nabla f| = (16^2 + 18^2)^{1/2} = 24$$

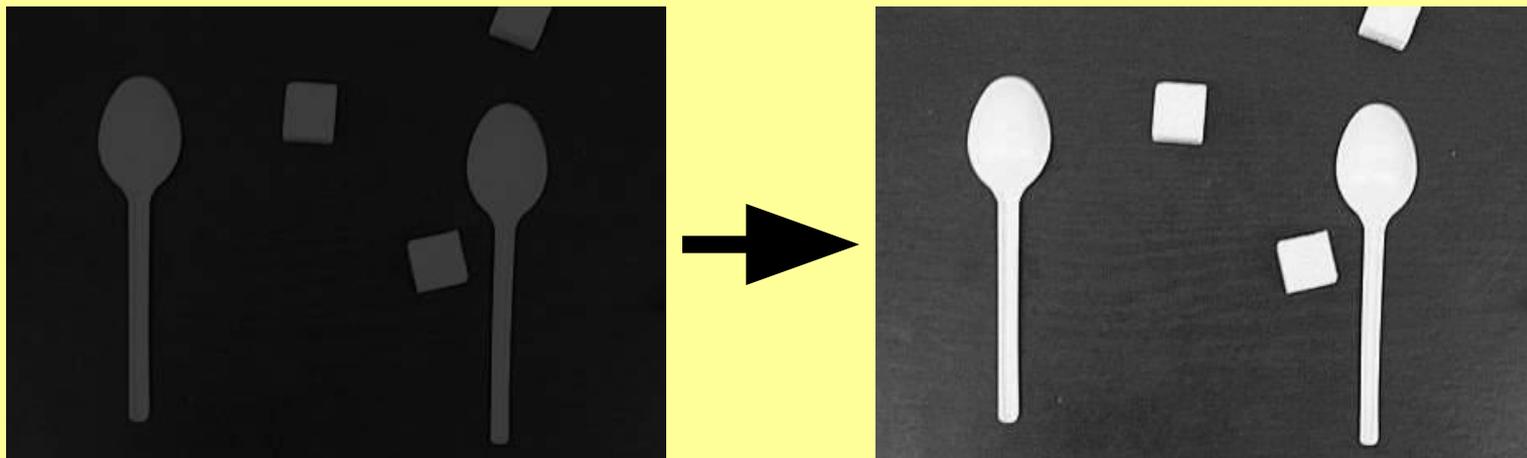
# Попиксельная обработка

- $\text{Out}[x, y] = f(\text{In}[x, y])$

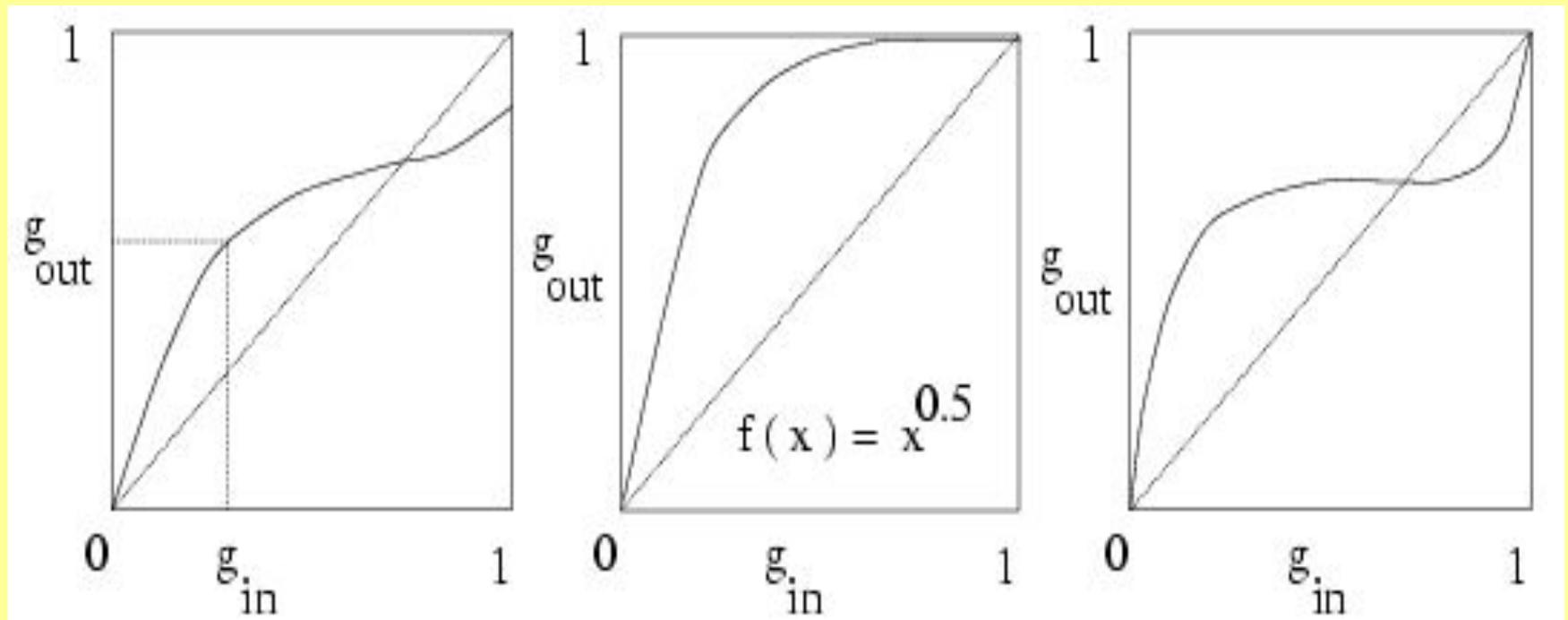
# Усиление и выделение полезной информации

Коррекция яркости изображения.  
Простое линейное преобразование:

$$g(i, j) = (g(i, j) - g_{\min}) * \frac{(255 - 0)}{(g_{\max} - g_{\min})}$$



# $g_{out} = f(g_{in})$



**gout = f(gin)**



# Выравнивание гистограмм

- Используются все  $n$  уровней серого цвета
- На каждый уровень попадает примерно одинаковое количество пикселей:  $q=(R*C)/n$

$$\sum_{i=1}^{t_1-1} H_{in}[i] \leq q_1 < \sum_{i=1}^{t_1} H_{in}[i].$$

$$\sum_{i=1}^{t_k-1} H_{in}[i] \leq (q_1 + q_2 + \dots + q_k) < \sum_{i=1}^{t_k} H_{in}[i].$$

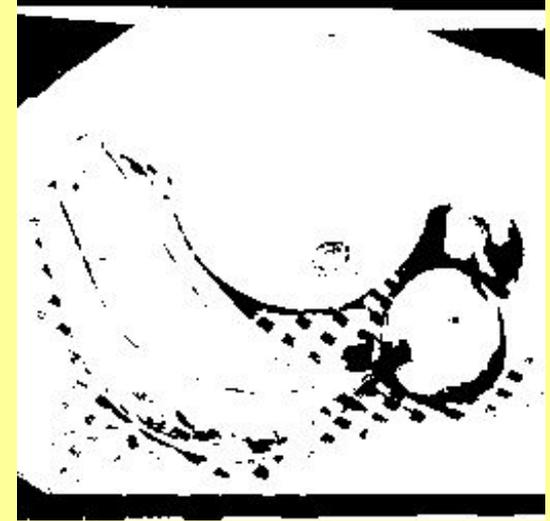
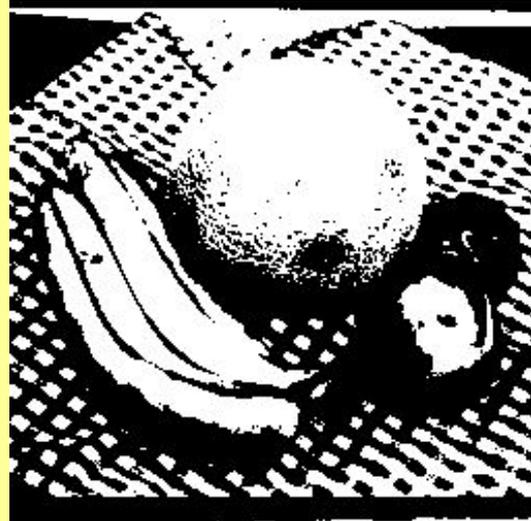
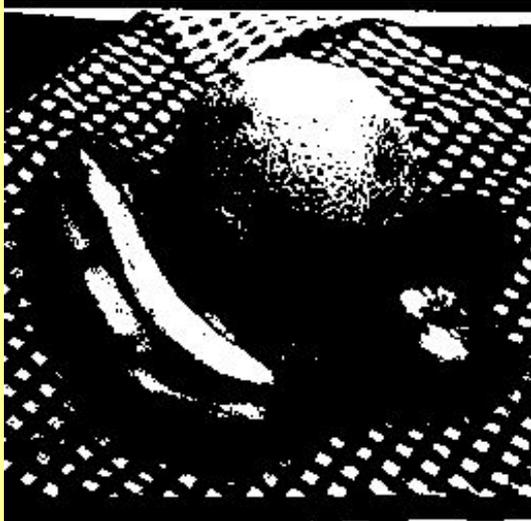
# Выравнивание гистограмм



# БИНАРИЗАЦИЯ ИЗОБРАЖЕНИЯ (1)



# ПРИМЕРЫ ПОРОГОВОЙ БИНАРИЗАЦИИ



# Подавление и устранение шума

- Устранение шума в бинарных изображениях

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

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

# Устранение шума типа «соль и перец»

Задача – избавиться от одиночных темных пикселей в светлых областях и одиночных светлых пикселей в темных областях.

Применение масок к изображению

С учетом 8ми соседей

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

С учетом 4х соседей

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

# МОРФОЛОГИЧЕСКИЕ ФИЛЬТРЫ

- Сужение (Эрозия, Erosion)
- Расширение (Dilation)
- Закрытие (Closing)
- Раскрытие (Opening)

# Морфологические операции

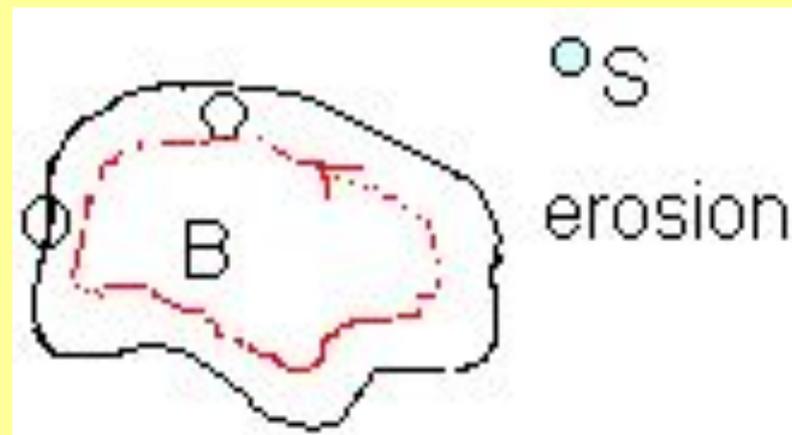
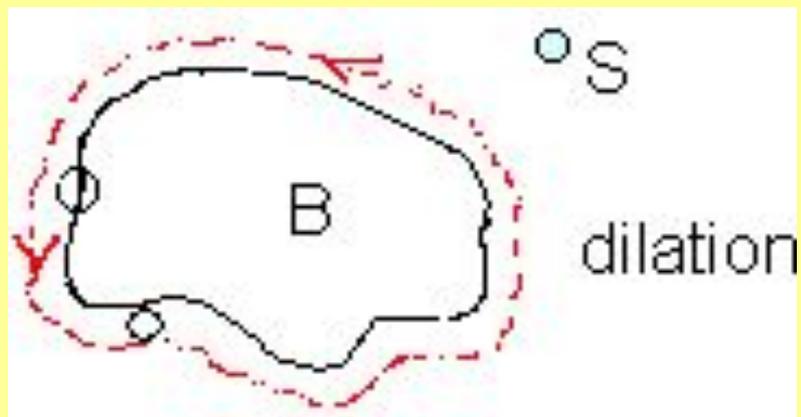
$$B \oplus S = \bigotimes_{b \in B} S_b$$

$$B(-)S = \{b \mid b + s \in B \forall s \in S\}$$

$$B \bullet S = (B \oplus S)(- )S$$

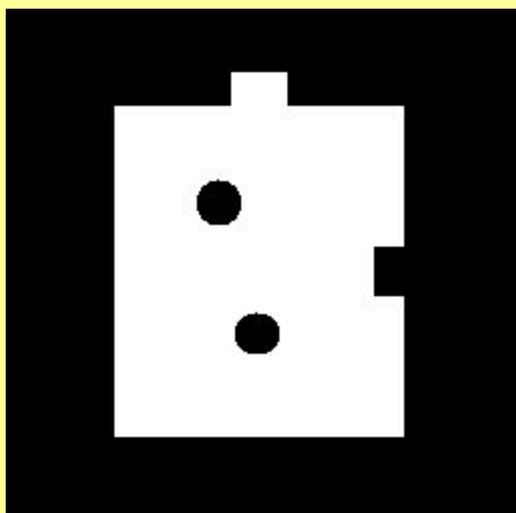
$$B \boxtimes S = (B(-)S) \oplus S$$

# Расширение & Сужение

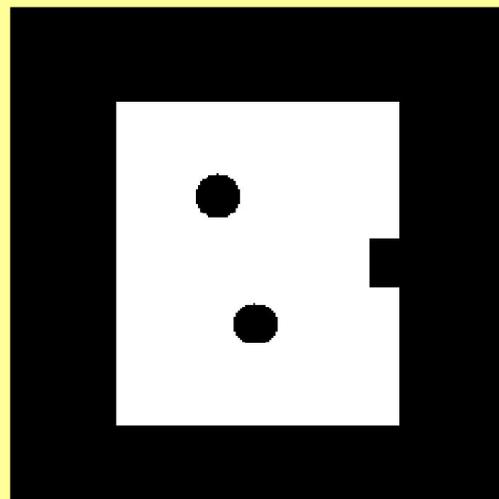


# ПРИМЕР МОРФОЛОГИЧЕСКОГО РАСКРЫТИЯ

**A**

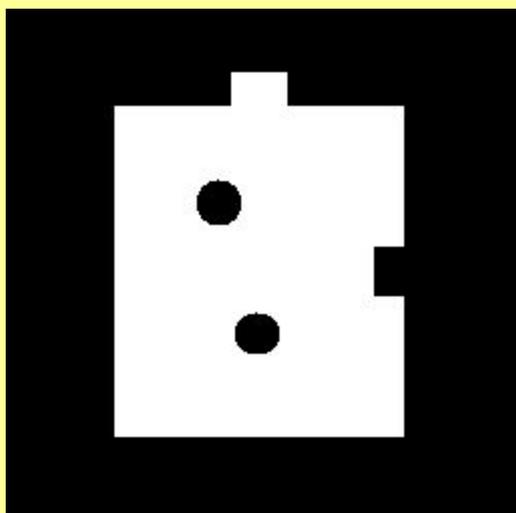


**open(A)**

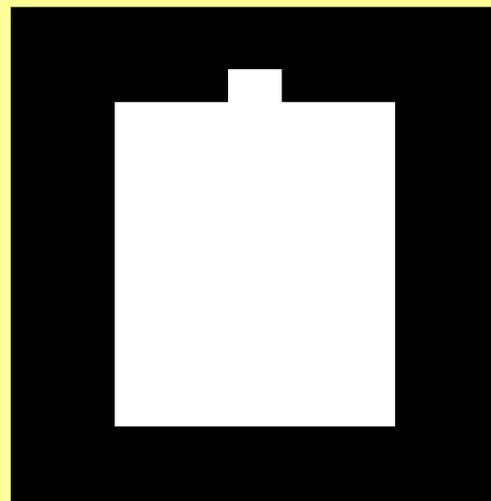


# ПРИМЕР МОРФОЛОГИЧЕСКОГО ЗАКРЫТИЯ

**A**



**close(A)**



# Морфологические операции

1	1	1	1	1	1	1	
			1	1	1	1	
			1	1	1	1	
		1	1	1	1	1	
			1	1	1	1	
		1	1				

a) Binary image  $B$

1	1	1
1	1	1
1	1	1

b) Structuring Element  $S$

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
	1	1	1	1	1	1	1
	1	1	1	1	1	1	1
	1	1	1	1	1	1	1
	1	1	1	1	1	1	1
	1	1	1	1			

c) Dilation  $B \oplus S$

			1	1			
			1	1			
			1	1			

d) Erosion  $B \ominus S$

	1	1	1	1	1	1	
		1	1	1	1	1	
		1	1	1	1	1	
		1	1	1	1	1	
		1	1	1	1	1	
		1	1				

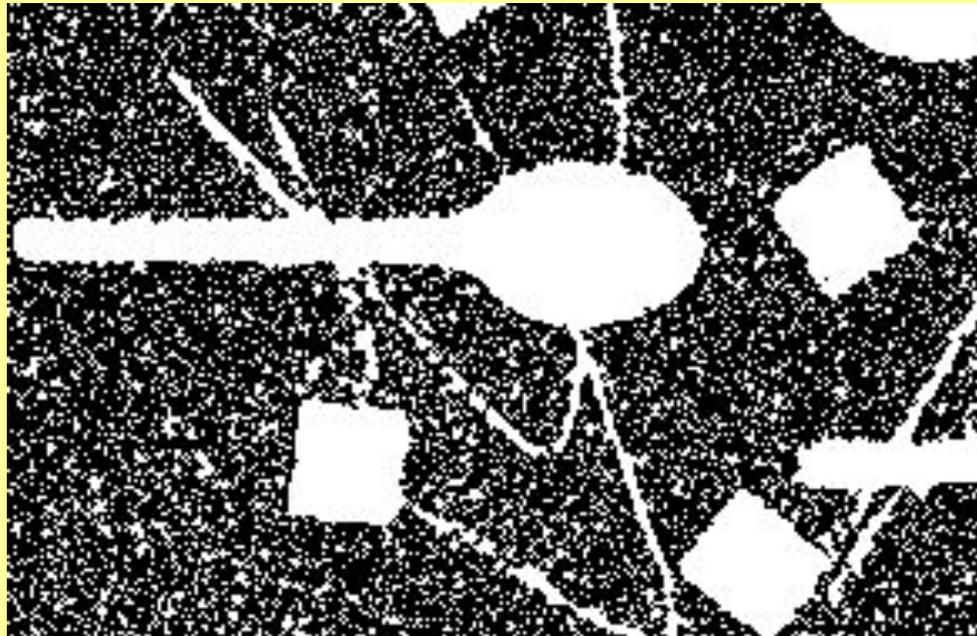
e) Closing  $B \bullet S$

		1	1	1	1		
		1	1	1	1		
		1	1	1	1		
		1	1	1	1		
		1	1	1	1		

f) Opening  $B \circ S$

# Устранение шума в бинарных изображениях

Пример бинарного изображению с сильным шумом



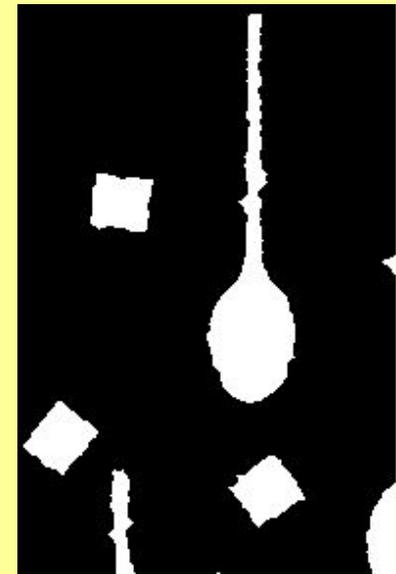
# Применения сужения к бинарному изображению с сильным шумом



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



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



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

# Устранение шума в бинарных изображениях

Пример бинарного изображения с дефектами  
распознаваемых объектов

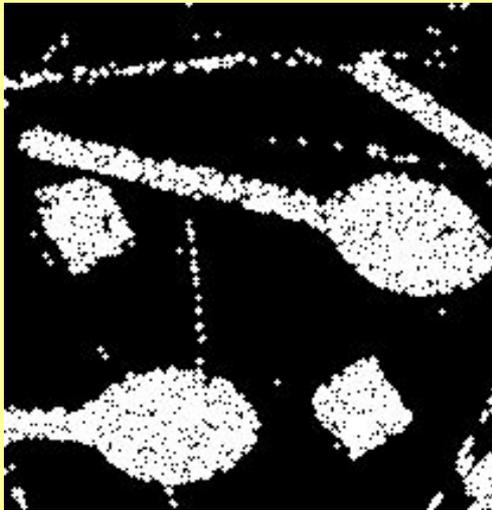


# Не лучший пример для морфологии

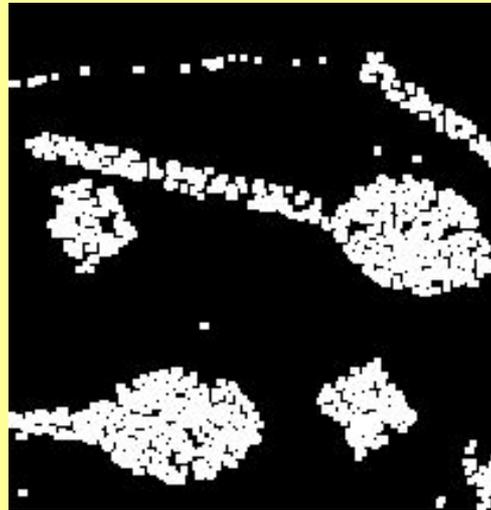
Не во всех случаях математическая морфология так легко убирает дефекты, как хотелось бы...



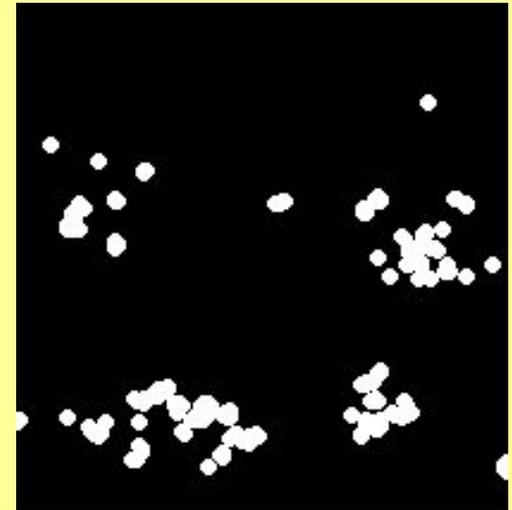
# Применения операции открытия



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



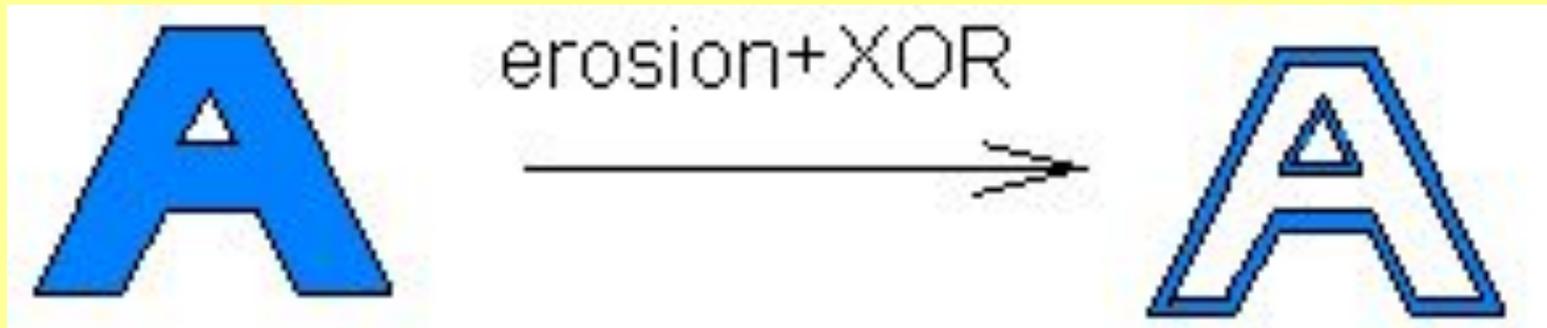
$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



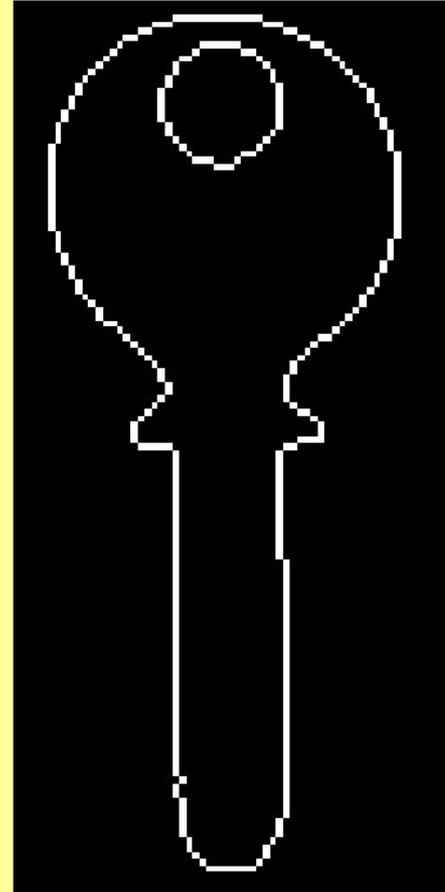
$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

# Выделение границ

- $B \setminus (B \ominus S)$  – внутренняя граница
- $(B \oplus S) \setminus B$  – внешняя граница



# ПРИМЕР ОКОНТУРИВАНИЯ ОБЪЕКТА



# МОРФОЛОГИЧЕСКАЯ СКЕЛЕТИЗАЦИЯ

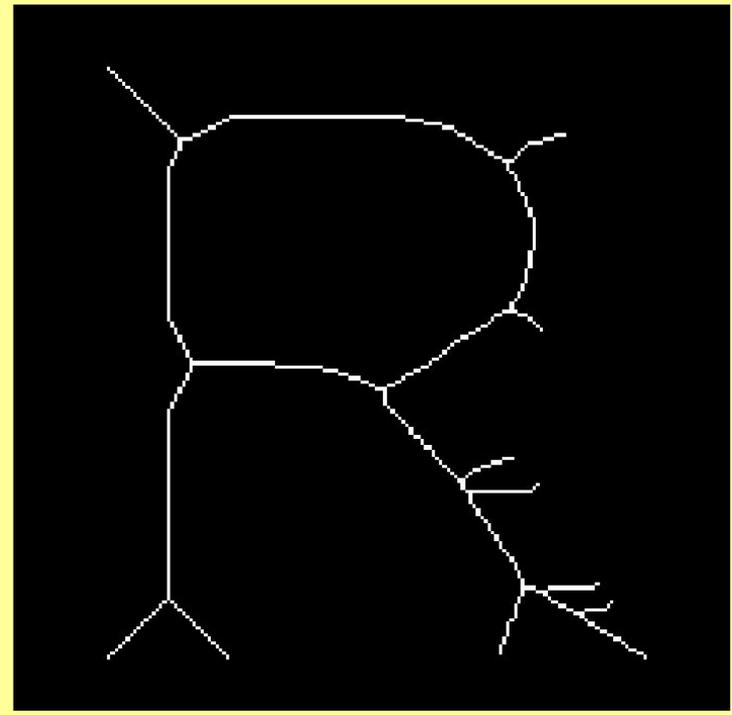
- Операция скелетонизации

$$S_n = (X - nQ) \setminus [(X - nQ) \circ Q], n = 0, 1, \dots, N$$

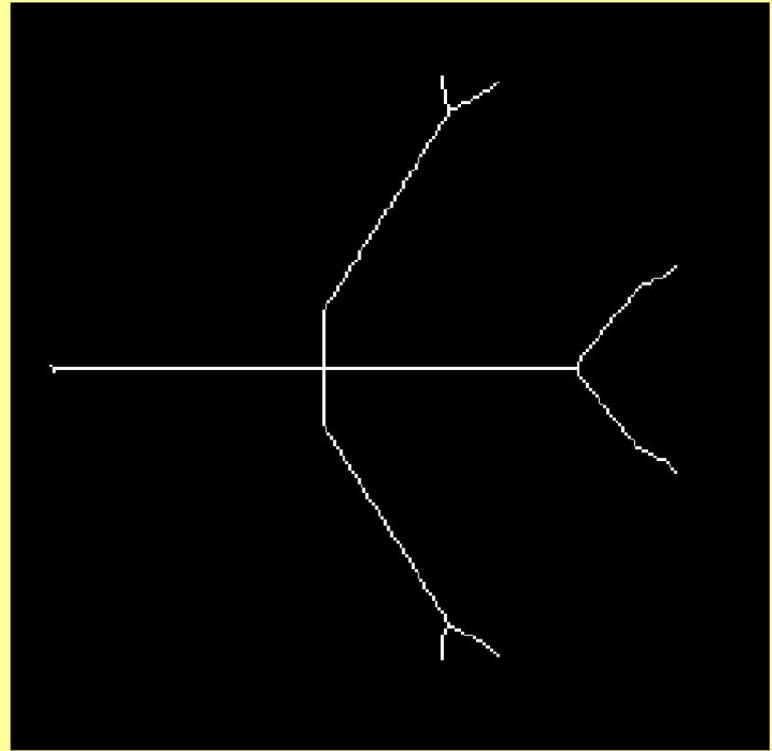
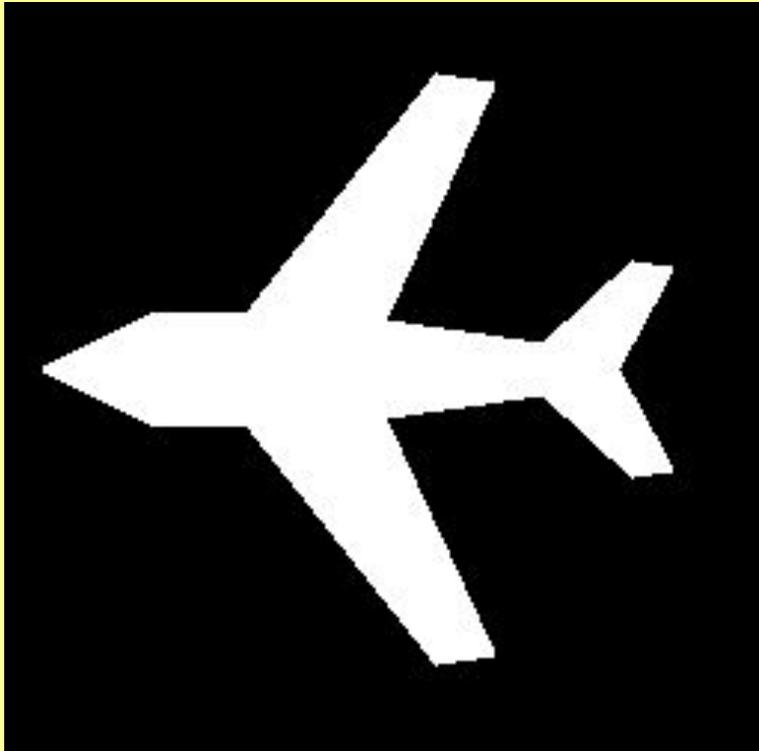
- Операция де-скелетонизации

$$X \circ kQ = \bigcup_{k \leq n \leq N} S_n \oplus nQ, 0 \leq k \leq N$$

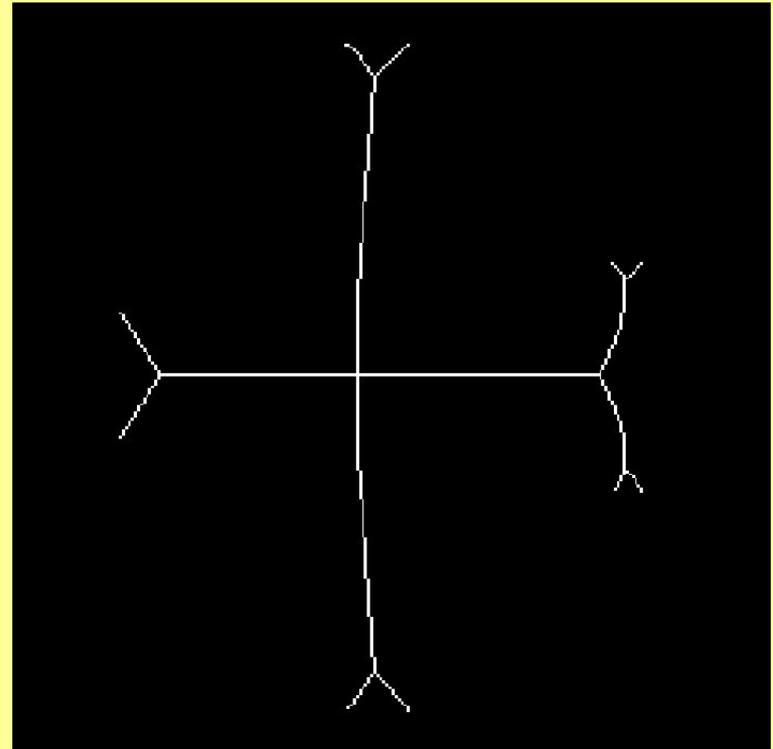
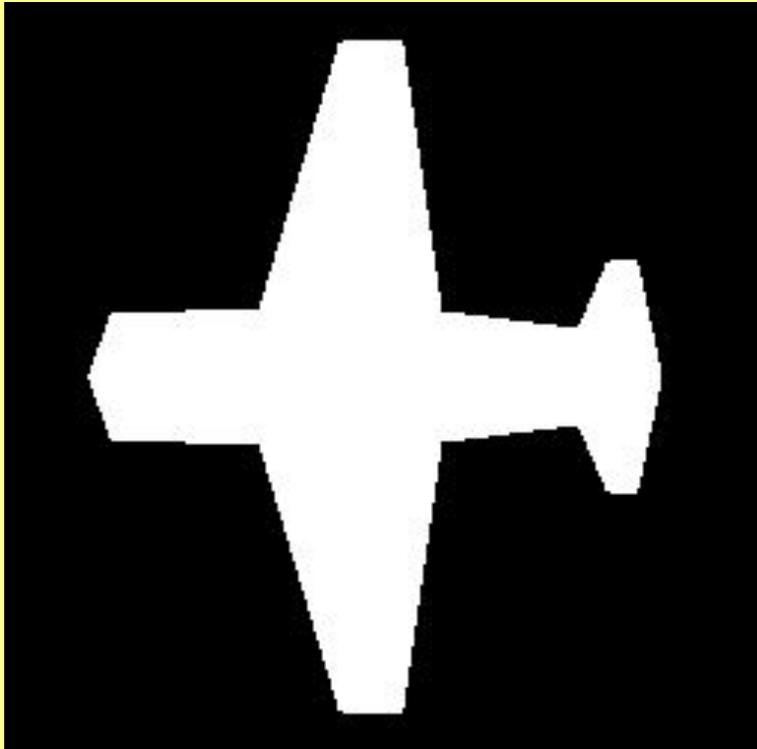
# МОРФОЛОГИЧЕСКАЯ СКЕЛЕТИЗАЦИЯ



# ПРИМЕРЫ МОРФОЛОГИЧЕСКОЙ СКЕЛЕТИЗАЦИИ (1)



# ПРИМЕРЫ МОРФОЛОГИЧЕСКОЙ СКЕЛЕТИЗАЦИИ (2)



# Медианный фильтр

Выбор медианы среди значений яркости пикселей в некоторой окрестности.

Определение медианы:

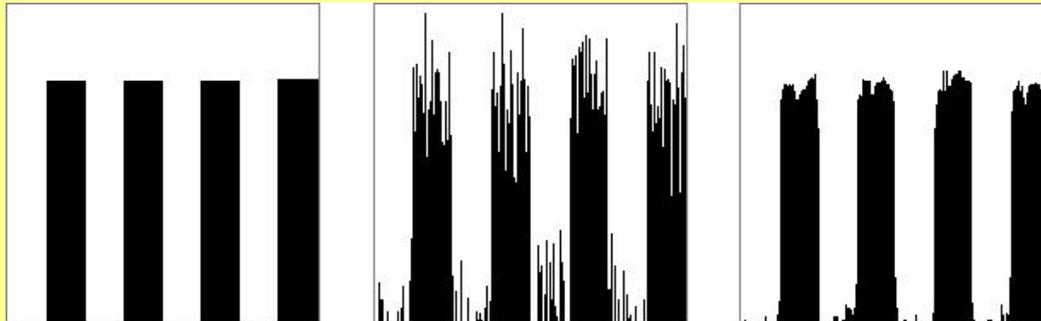
$A_i, i = \overline{1, n}$ ; - отсортированный набор чисел,

$A_{[n/2]}$  – медиана набора.

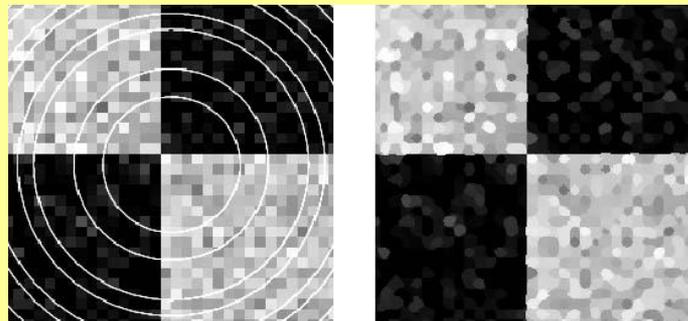
Медианный фильтр радиусом  $r$  – выбор медианы среди пикселей в окрестности  $[-r, r]$ .

# Применение медианного фильтра

Результат применения медианного фильтра с радиусом 5 пикселей

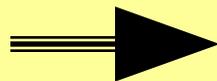


Результат применения медианного фильтра с радиусом в 7 пикселей к изображению с шумом и артефактами в виде тонких светлых окружностей.



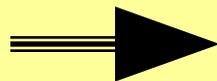
# Очистка изображения с помощью медианного фильтра

Фильтр с радиусом 3x3



# Очистка изображения с помощью медианного фильтра

Фильтр с радиусом 3x3



# Быстрая реализация медианного фильтра

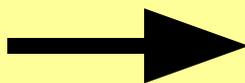
Медианный фильтр считается дольше, чем операция свертки, поскольку требует частичной сортировки массива яркостей окрестных пикселей.

Возможности ускорения:

- Делать несколько шагов быстрой сортировки до получения медианы
- Конкретная реализация для каждого радиуса (3x3, 5x5)

# Анализ информации, содержащейся в изображении

Нас интересуют области однородные по некоторому признаку - например по яркости. Простейший случай – области, яркость которых выше/ниже некоторого порога



# Выделение связанных областей

Определение связанной области:

Область, каждый пиксель которой связан с одним из других пикселей, принадлежащих данной области.

Связность пикселей:

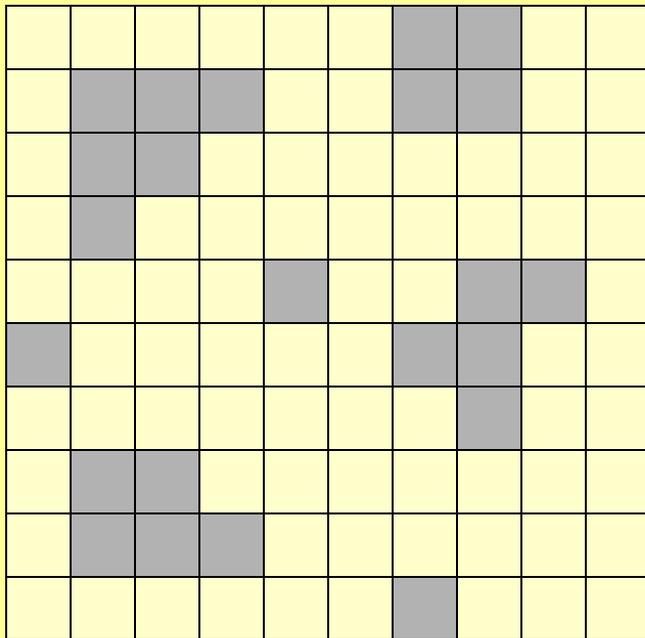
	1	
2	*	3
	4	

4-связность

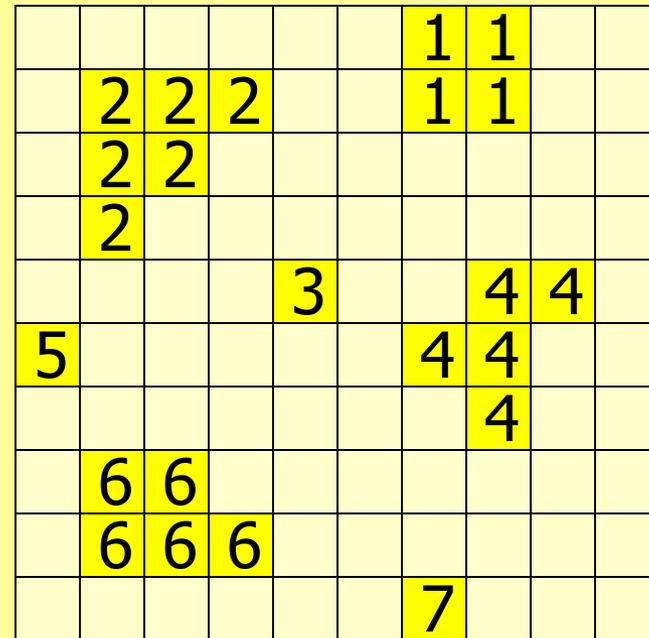
1	2	3
4	*	5
6	7	8

8-связность

# Разметка связанных областей



Бинарное изображение



Размеченное изображение

# Рекурсивная разметка СВЯЗНЫХ областей (1)

```
void Labeling(BIT* img[], int* labels[])
{
    // labels должна быть обнудена
    L = 1;
    for(y = 0; y < H; y++)
        for(x = 0; x < W; x++)
            if( (img[x][y] == 1)
                && (labels[x][y] == 0) )
            {
                Fill(img, labels, x, y, L++);
            }
}
```

# Рекурсивная разметка СВЯЗНЫХ областей (2)

```
void Fill(BIT* img[], int* labels[], int x, int y, int L)
{
    if( (labels[x][y] == 0) && (img[x][y] == 1) )
    {
        labels[x][y] = L;
        if( x > 0 )
            Fill(img, labels, x - 1, y, L);
        if( x < W - 1 )
            Fill(img, labels, x + 1, y, L);
        if( y > 0 )
            Fill(img, labels, x, y - 1, L);
        if( y < H - 1 )
            Fill(img, labels, x, y + 1, L);
    }
}
```

# Разметка связанных областей путем последовательного сканирования

Последовательно, сканируем бинарное изображение сверху вниз, слева направо:

	<b>C</b>	
<b>B</b>	<b>A</b>	

```
if A = 0
    do nothing

else if (not B labeled) and (not C labeled)
    increment label numbering and label A

else if B xor C labeled
    copy label to A

else if B and C labeled
    if B label = C label
        copy label to A
    else
        copy either B label or C label to
A

    record equivalence of labels
```

Постобработка - переразметка с учетом эквивалентностей областей

# Разметка связанных областей путем последовательного сканирования

Случай конфликта:

						1
					1	1
2				1	1	1
2	2		1	1	1	1
2	2	2	?			

Постобработка - переразметка с учетом эквивалентностей областей

# Анализ формы связанных областей

Для каждой области можно подсчитать некий набор простейших числовых характеристик:

- Площадь
- Периметр
- Компактность
- Ориентацию главной оси инерции
- Удлиненность (эксцентриситет)

На основе этих характеристик можно классифицировать получаемые области.

# Анализ формы связных областей

**Площадь** – количество пикселей в области;

**Периметр** – количество пикселей принадлежащих границе области;

**Компактность** – отношение квадрата периметра к площади;

$$C = \frac{P^2}{A}$$

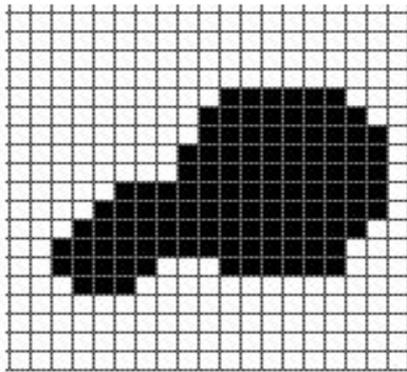
Наиболее компактная фигура – круг,  
 $C = 4\pi$ .

# Подсчет периметра области

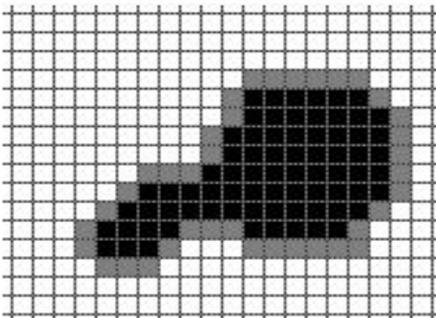
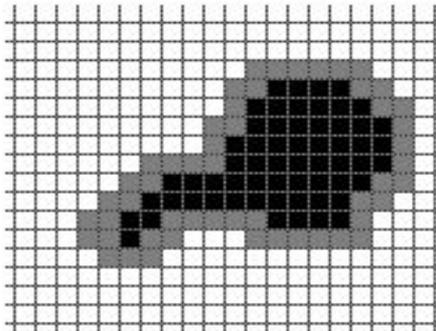
1. Пиксель лежит на границе области, если он сам принадлежит области и хотя бы один из его соседей области не принадлежит.  
(внутренняя граница)
2. Пиксель лежит на границе области, если он сам **не** принадлежит области и хотя бы один из его соседей области принадлежит.  
(внешняя граница)

Периметр зависит также от того 4-х или 8-ми связность используется для определения соседей.

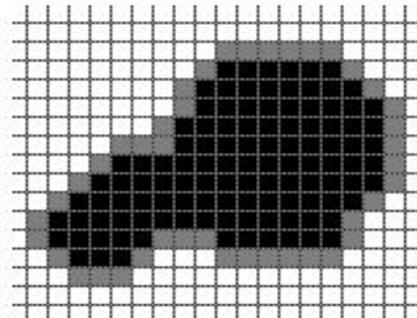
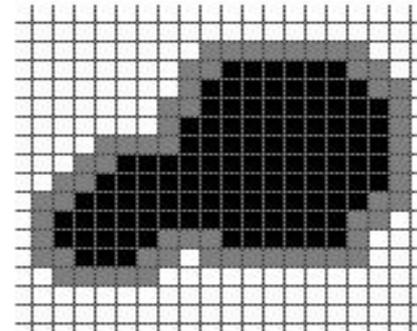
# Пример периметров области



Область



Внутренняя граница



Внешняя граница

# Статистические моменты области

Дискретный центральный момент  $m_{ij}$  области определяется следующим образом:

$$m_{ij} = \sum_{(x,y) \in Reg} (x - \bar{x})^i (y - \bar{y})^j;$$

$$\bar{x} = \frac{1}{n} \sum_{(x,y) \in Reg} x; \quad \bar{y} = \frac{1}{n} \sum_{(x,y) \in Reg} y;$$

$n$  – общее количество пикселей в области

# Инвариантные характеристики области

Для распознавания нас интересуют характеристики инвариантные по отношению к масштабированию, переносу, повороту:

- **Удлиненность, нецентрированность (эксцентриситет)**

$$elongation = \frac{m_{20} + m_{02} + \sqrt{(m_{20} - m_{02})^2 + 4m_{11}^2}}{m_{20} + m_{02} - \sqrt{(m_{20} - m_{02})^2 + 4m_{11}^2}}$$

- **Компактность**

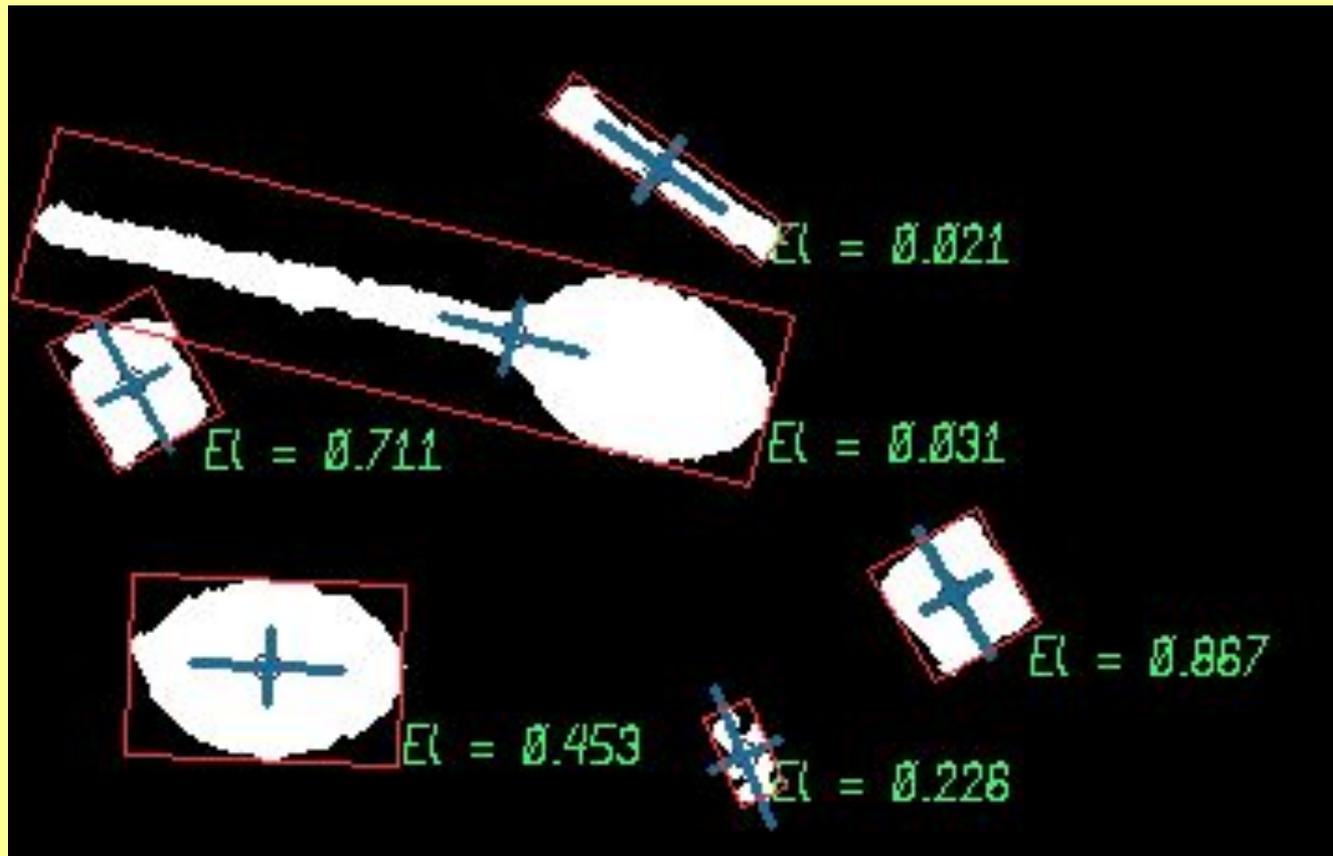
$$C = \frac{P^2}{A}$$

# Ориентация главной оси инерции

Не является инвариантной к повороту, но в ряде случаев предоставляет полезную информацию об ориентации объекта:

$$\theta = \frac{1}{2} \tan^{-1} \left( \frac{2m_{11}}{m_{20} - m_{02}} \right)$$

# Пример изображения с подсчитанными характеристиками областей



# Другие инвариантные характеристики области

$$M_1 = m_{20} + m_{02}$$

$$M_2 = (m_{20} - m_{02})^2 + 4m_{11}^2$$

$$M_3 = (m_{30} - 3m_{12})^2 + (3m_{21} - m_{03})^2$$

$$M_4 = (m_{30} + m_{12})^2 + (m_{21} + m_{03})^2$$

$$M_5 = (m_{30} - 3m_{12})(m_{30} + m_{12}) [(m_{30} + m_{12})^2 - 3(m_{21} + m_{03})^2] \\ + (3m_{21} - m_{03})(m_{21} + m_{03}) [3(m_{30} + m_{12})^2 - (m_{21} + m_{03})^2]$$

$$M_6 = (m_{20} + m_{02}) [(m_{30} + m_{12})^2 - 3(m_{21} + m_{03})^2] \\ + 4m_{11}(m_{30} + m_{12})(m_{03} + m_{21})$$

$$M_7 = (3m_{21} - m_{03})(m_{12} + m_{30}) [(m_{30} + m_{12})^2 - 3(m_{21} + m_{03})^2] \\ - (m_{30} - 3m_{12})(m_{12} + m_{03}) [3(m_{30} + m_{12})^2 - (m_{21} + m_{03})^2]$$

(54)

# Пример изображения для обработки



# Литература

[http://graphics.cs.msu.su/ru/library/cv/cv\\_intro.html](http://graphics.cs.msu.su/ru/library/cv/cv_intro.html)

[http://graphics.cs.msu.su/courses/cg2000b/notes/lect\\_1.doc](http://graphics.cs.msu.su/courses/cg2000b/notes/lect_1.doc)

# Задания

<http://graphics.cs.msu.su/courses/cg02b/assigns/hw-3/hw3.html>

<http://graphics.cs.msu.su/courses/cg01b/hw4/hw-4.htm>

<http://graphics.cs.msu.su/courses/cg2000b/hw2/hw-2.htm>