

Компьютерная обработка результатов измерений

Лекция 7. Обработка изображений, часть 2

Емельянов Эдуард Владимирович

Специальная астрофизическая обсерватория РАН
Лаборатория физики оптических транзиентов



1 Вейвлеты

2 Морфологические операции

3 Сегментация изображений

4 Деконволюция

5 Обнаружение

Вейвлеты

Разложение функции по базису

Преобразование Фурье и любые другие преобразования $f(x)$ по базису $r(x, u)$ в 1-мерном варианте можно представить выражением

$$T(u) = \sum_{x=0}^{N-1} f(x)r(x, u), \quad f(x) = \sum_{u=0}^{N-1} T(u)s(x, u).$$

В матричном виде: $\mathbf{t} = \mathbf{R}\mathbf{f}$ и $\mathbf{f} = \mathbf{S}\mathbf{t}$. Очевидно, что $\mathbf{S} = \mathbf{R}^{-1}$ и обратно.
Если базис \mathbf{S} ортонормирован ($\mathbf{S}^T\mathbf{S} = \mathbf{I}$), то $\mathbf{R} = \mathbf{S}^T$.

В двумерном виде $s = s(x, y, u, v)$, положим, что ядро — разделяемое и симметричное, т.е. $s = s(x, y) \cdot s(u, v)$. В этом случае ядро можно записать в виде двумерной матрицы:

$$\mathbf{T} = \mathbf{S}\mathbf{F}\mathbf{S}^T, \quad \mathbf{F} = \mathbf{S}^T\mathbf{F}\mathbf{S}.$$



Вейвлеты

Рассмотрим простейший базис:

$$\mathbf{s}_0 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad \mathbf{s}_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}; \quad \mathbf{A} = (\mathbf{s}_0 \mathbf{s}_1)^T = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

Можно проверить, что данный базис — ортонормальный.

Для прямоугольных изображений размером $M \times N$ (M строк и N столбцов) пикселей ядра будут иметь размеры $M \times M$ и $N \times N$:

$$\mathbf{T} = \mathbf{A}_M \mathbf{F} \mathbf{A}_N^T, \quad \mathbf{F} = \mathbf{A}_M^T \mathbf{T} \mathbf{A}_N.$$

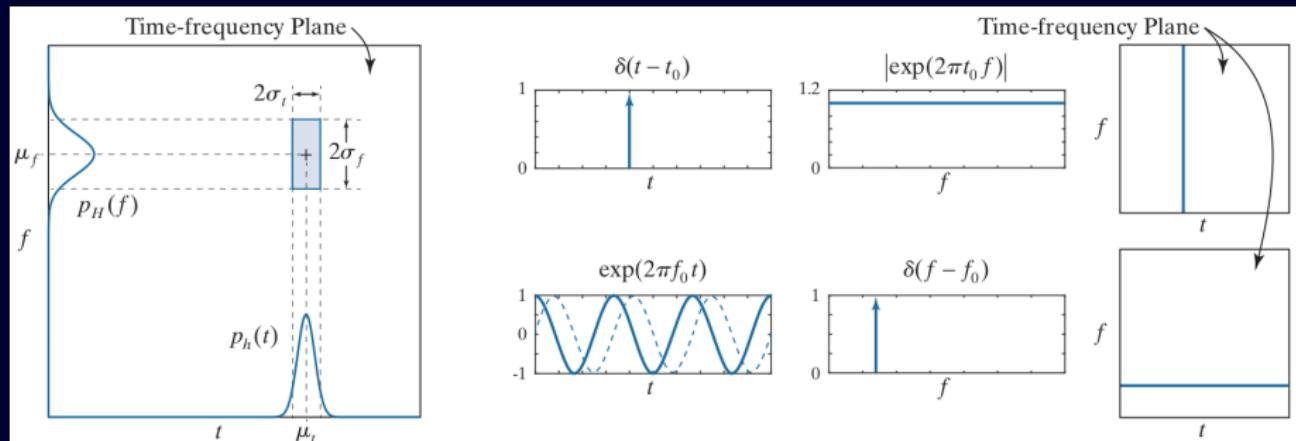
В случае комплексного базиса $\mathbf{T} = \mathbf{A} \mathbf{F} \mathbf{A}^T$, но $\mathbf{F} = \mathbf{A}^{*T} \mathbf{T} \mathbf{A}^*$.



Вейвлеты

Локализация

Дельта-функция локализована по времени, но не имеет локализации по частоте; комплексная синусоида — наоборот. **Вейвлет** имеет локализацию как по частоте, так и по времени.



Вейвлеты

Вейвлет

Имея **материнский вейвлет** $\psi(t)$, зададим **базис вейвлетов** как

$$\psi_{s,\tau} = 2^{s/2}\psi(2^s t - \tau).$$

Для дискретных изображений получаем набор **масштабирующих функций**:

$$\varphi_{j,k} = 2^{j/2}\varphi(2^j x - k),$$

k задает смещение вейвлета, j – его масштаб.

Базис вейвлетов и масштабирующих функций позволяет произвести декомпозицию изображений.

Одномерный случай:

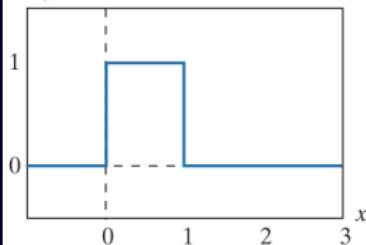
$$f(x) = \frac{1}{2} \left\{ T_\varphi(0, 0)\varphi(x) + T_\psi(0, 0)\psi_{0,0}(x) + T_\psi(1, 0)\psi_{1,0}(x) \dots \right\}.$$



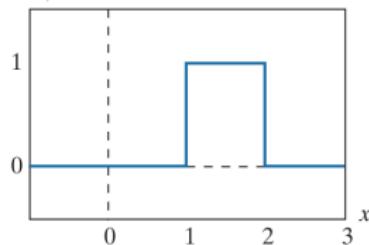
Вейвлеты

Масштабирующая функция Хаара: $\varphi(x) = 1$ при $0 \leq x \leq 1$.

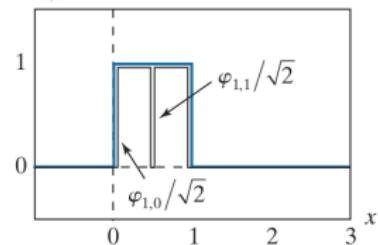
$$\varphi_{0,0}(x) = \varphi(x)$$



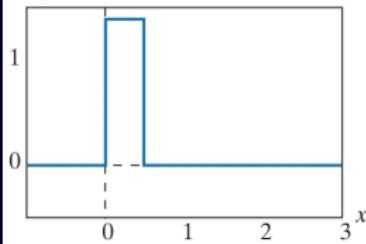
$$\varphi_{0,1}(x) = \varphi(x-1)$$



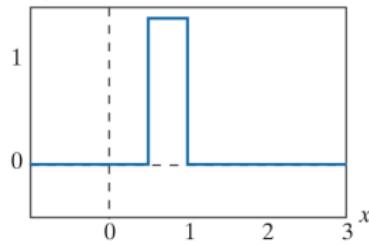
$$\varphi_{0,0}(x) \in V_1$$



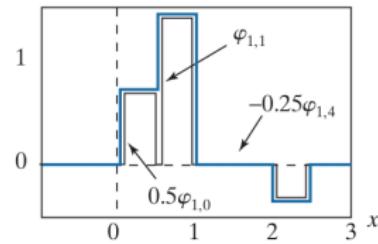
$$\varphi_{1,0}(x) = \sqrt{2}\varphi(2x)$$



$$\varphi_{1,1}(x) = \sqrt{2}\varphi(2x-1)$$



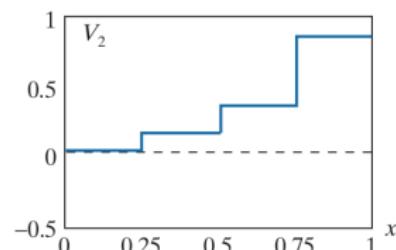
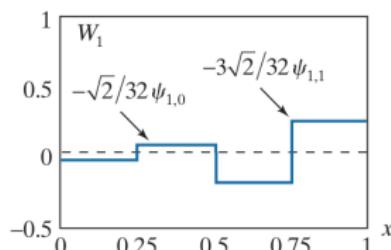
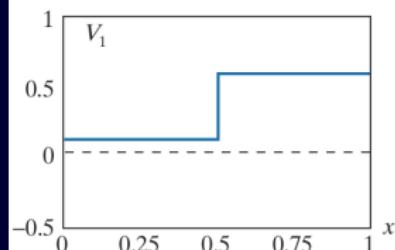
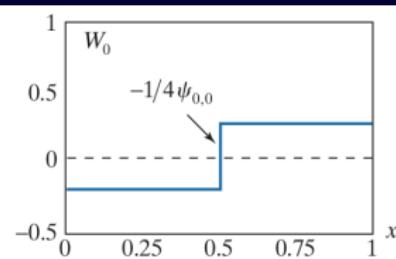
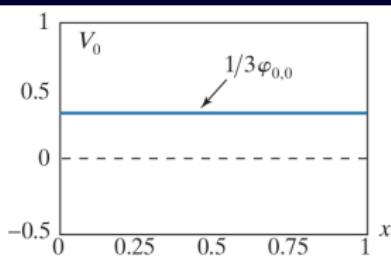
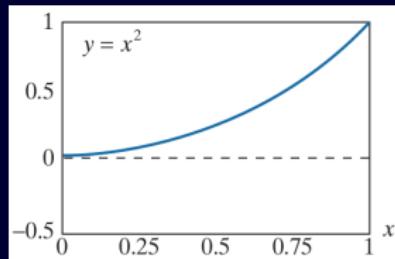
$$f(x) \in V_1$$



Вейвлеты

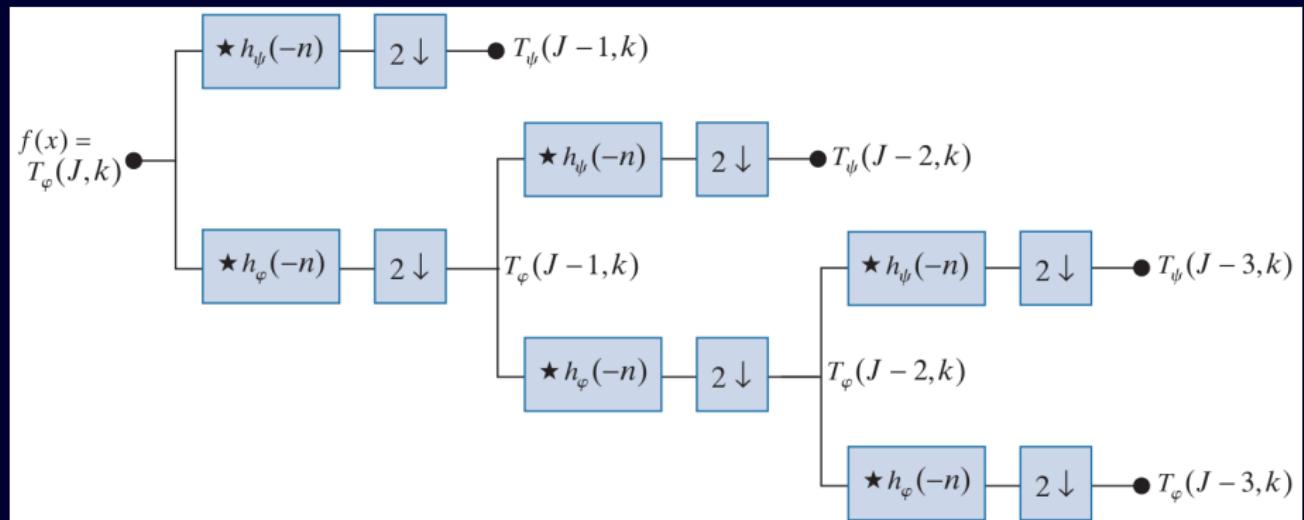
Вейвлет Хаара:

$$\psi(x) = \begin{cases} 1, & 0 \leq x < 0.5; \\ -1, & 0.5 \leq x < 1; \\ 0, & \text{в остальных случаях.} \end{cases}$$

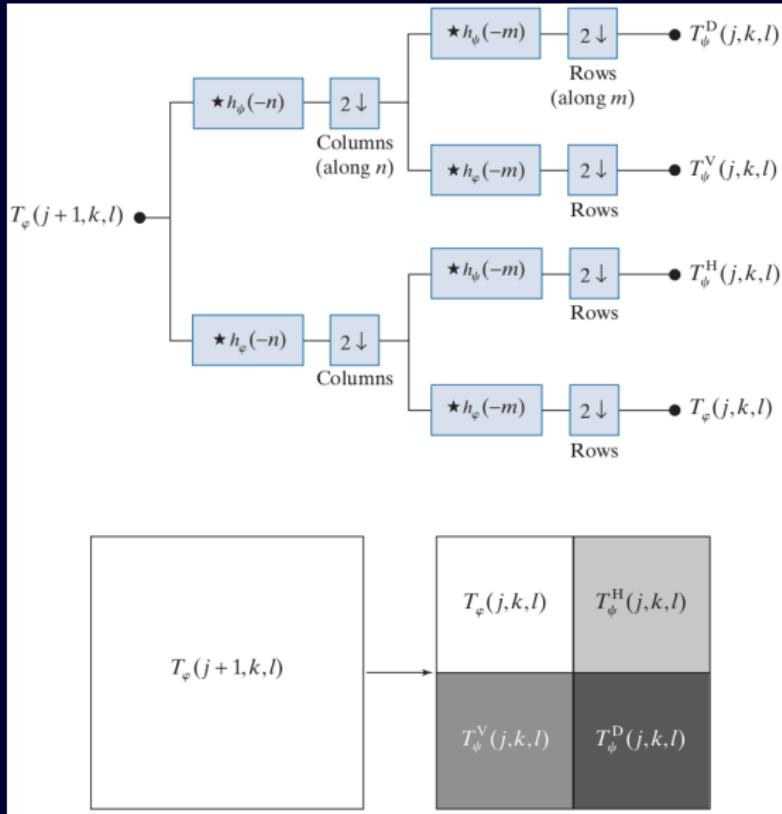


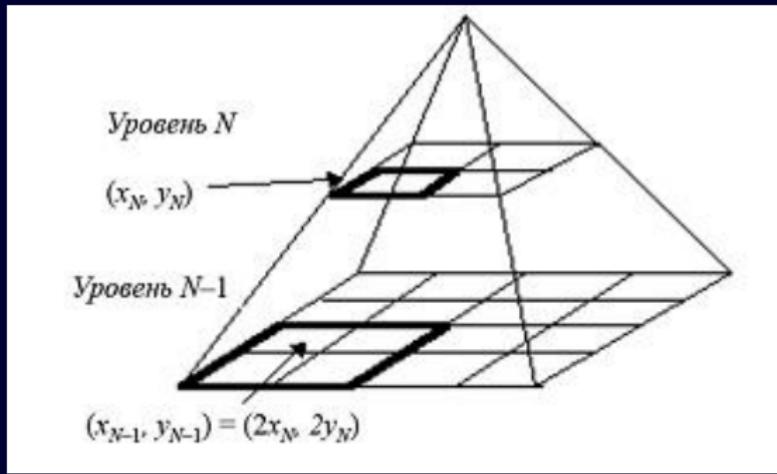
Вейвлеты

Пирамида преобразований



Вейвлеты

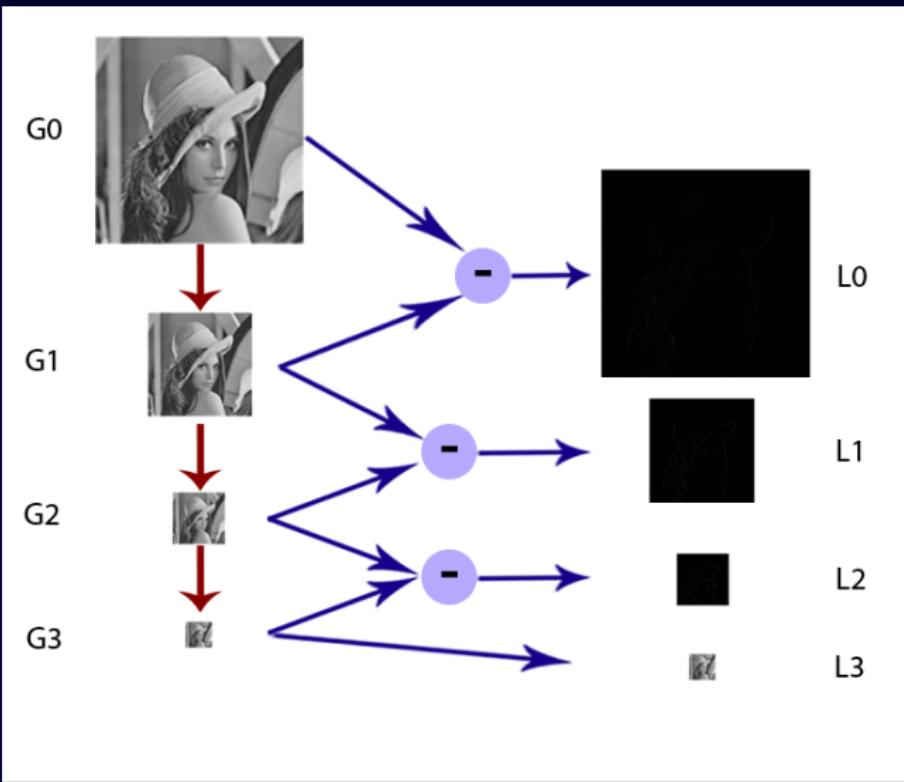


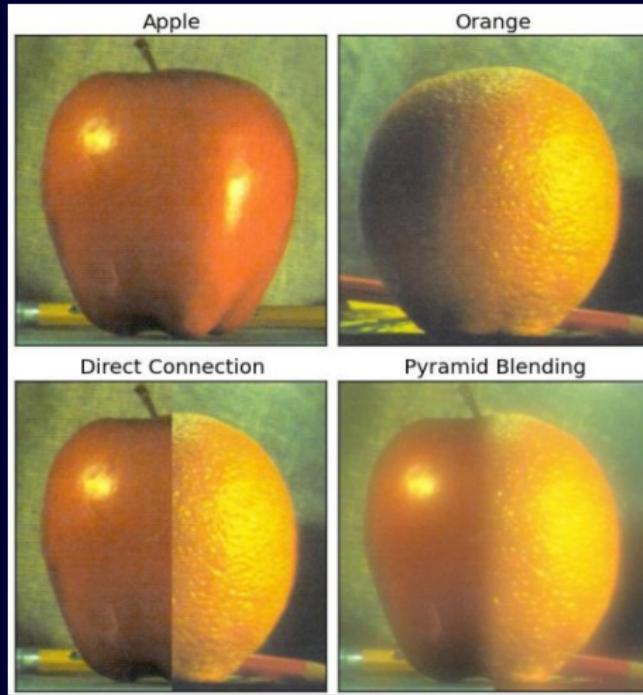


Пирамида изображений

Пирамида приближений (аппроксимирующие коэффициенты), пирамида ошибок (детализирующие коэффициенты). Пирамида Лапласа (только пирамида ошибок, компрессия); гауссова пирамида (только приближения, синтез текстур).





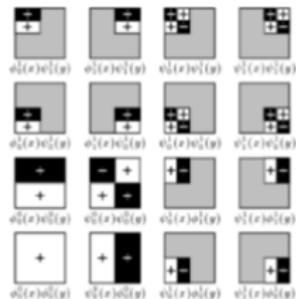


Объединение пирамид Лапласа.

2D Haar mother basis functions



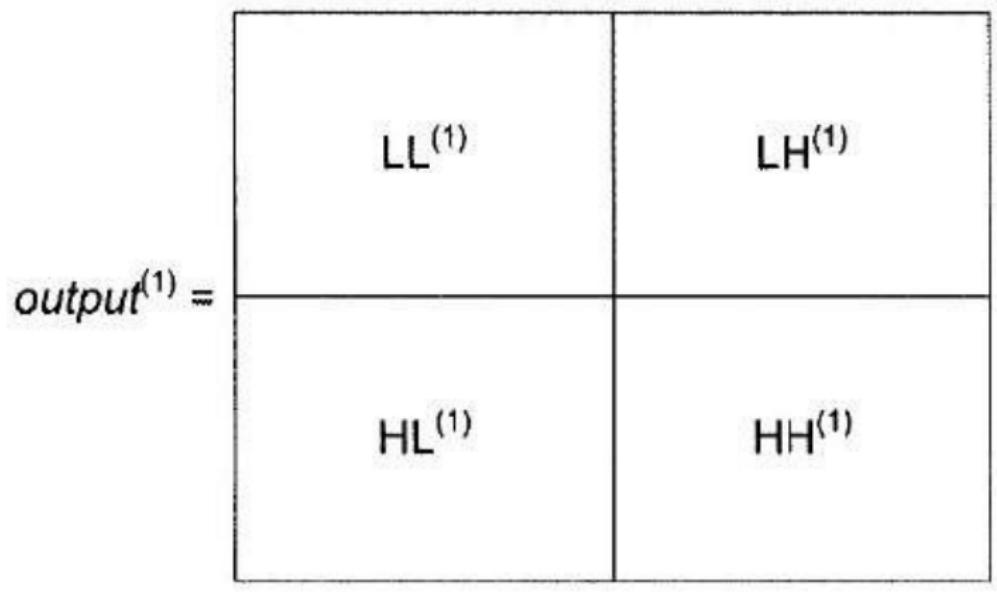
Non-standard Haar basis functions for 4x4 image



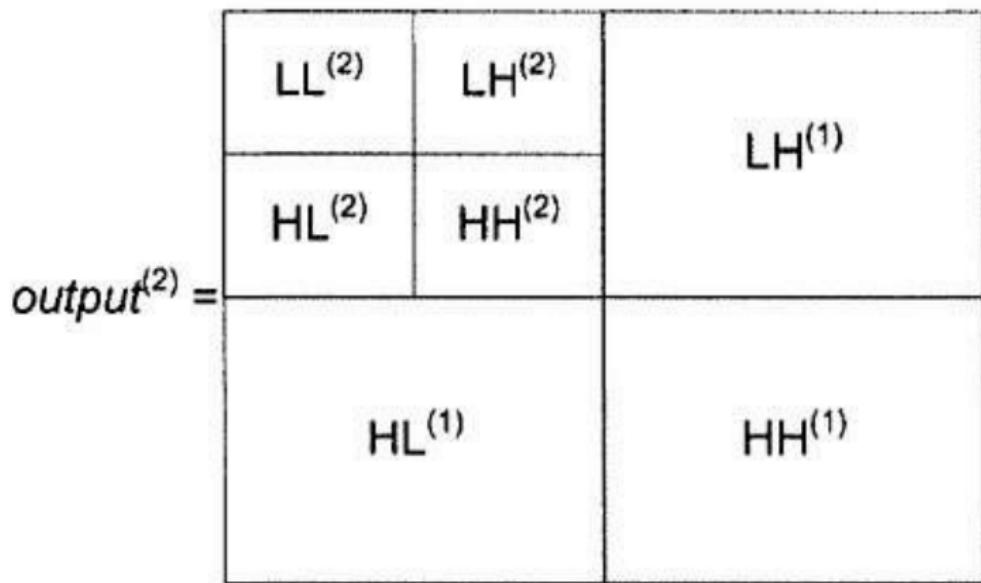
Sample images from **Wavelets for Computer Graphics: A Primer [1]**



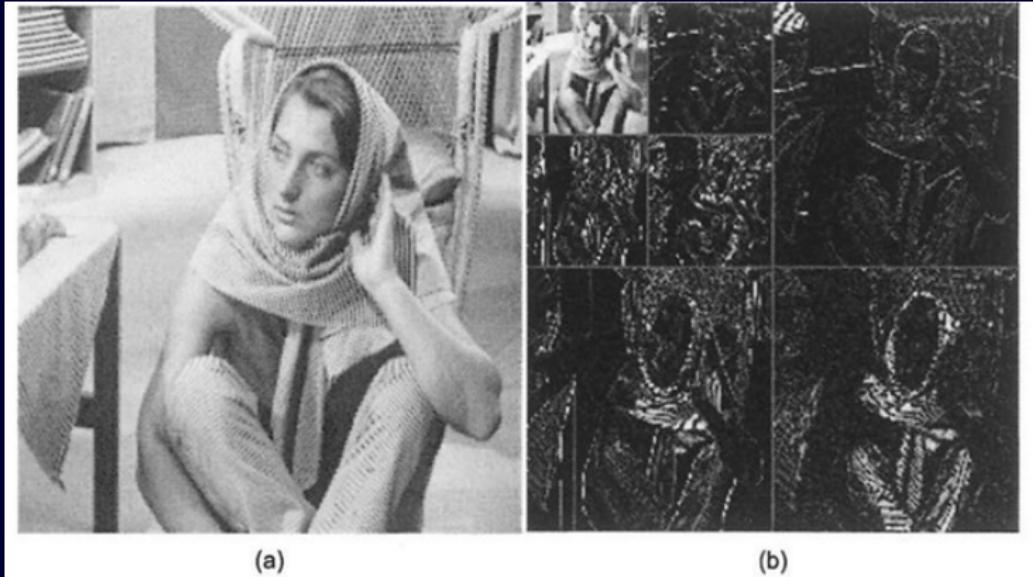
Вейвлеты



Вейвлеты



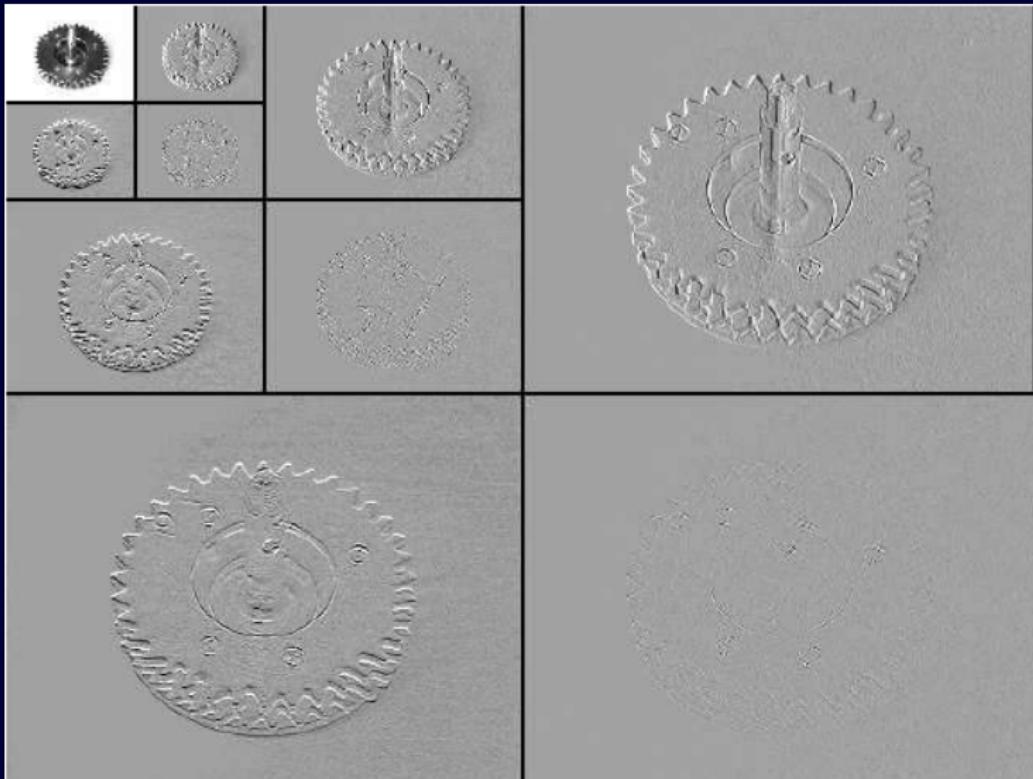
Вейвлеты



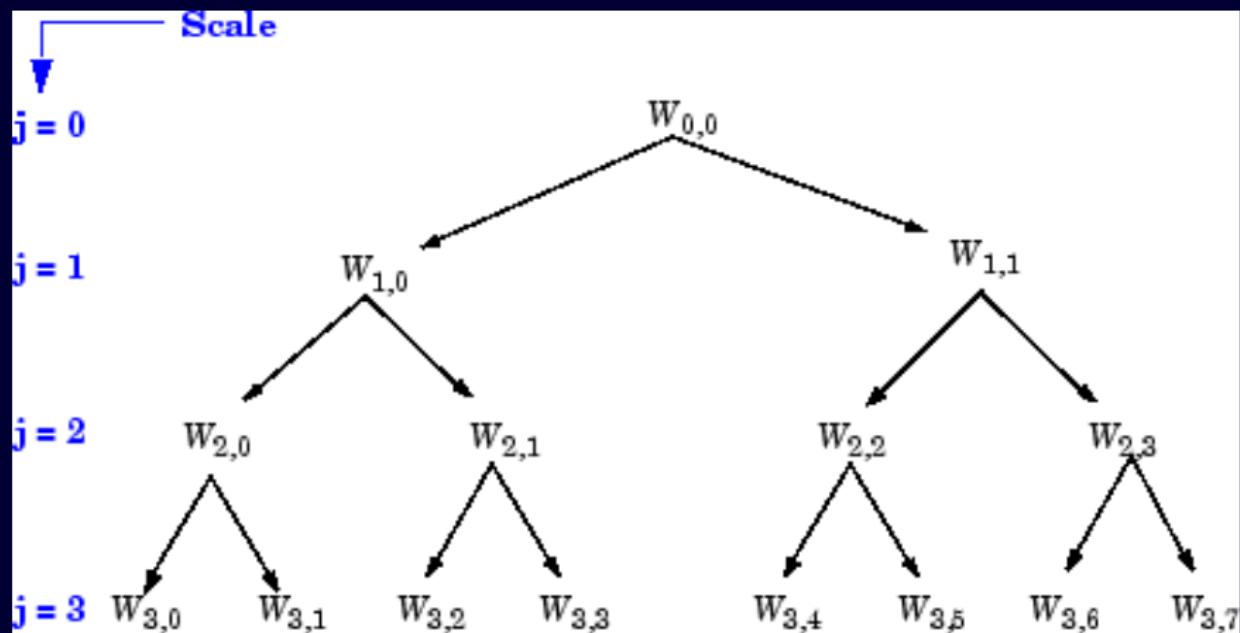
(a)

(b)

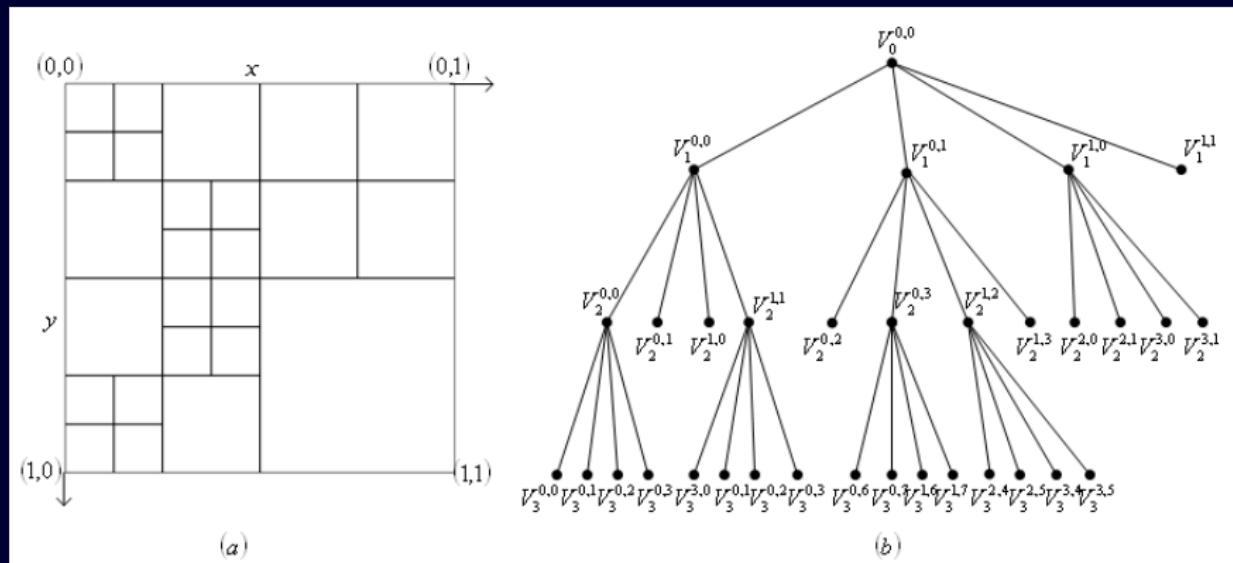
Вейвлеты



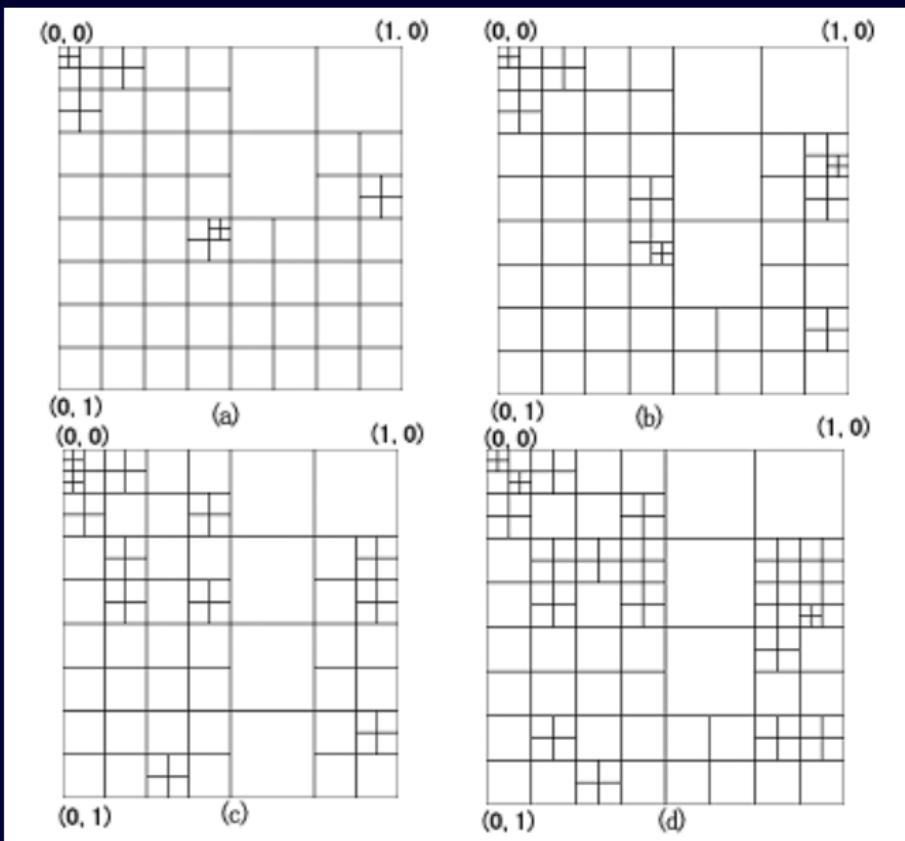
Пакеты вейвлетов



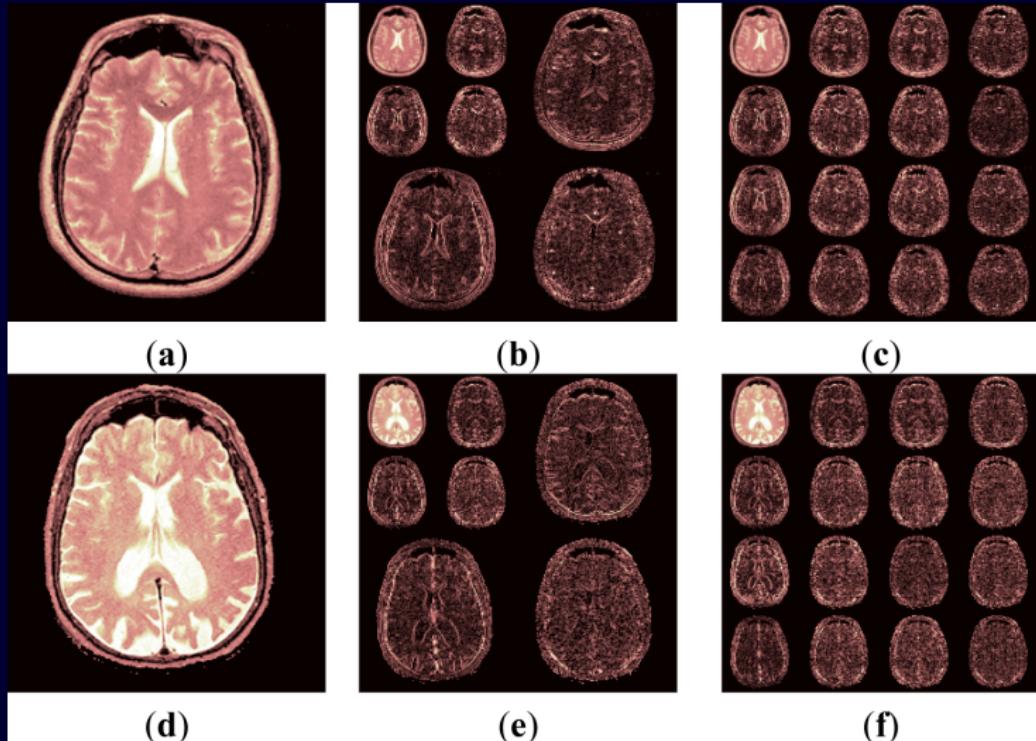
Пакеты вейвлетов



Пакеты вейвлетов



Пакеты вейвлетов



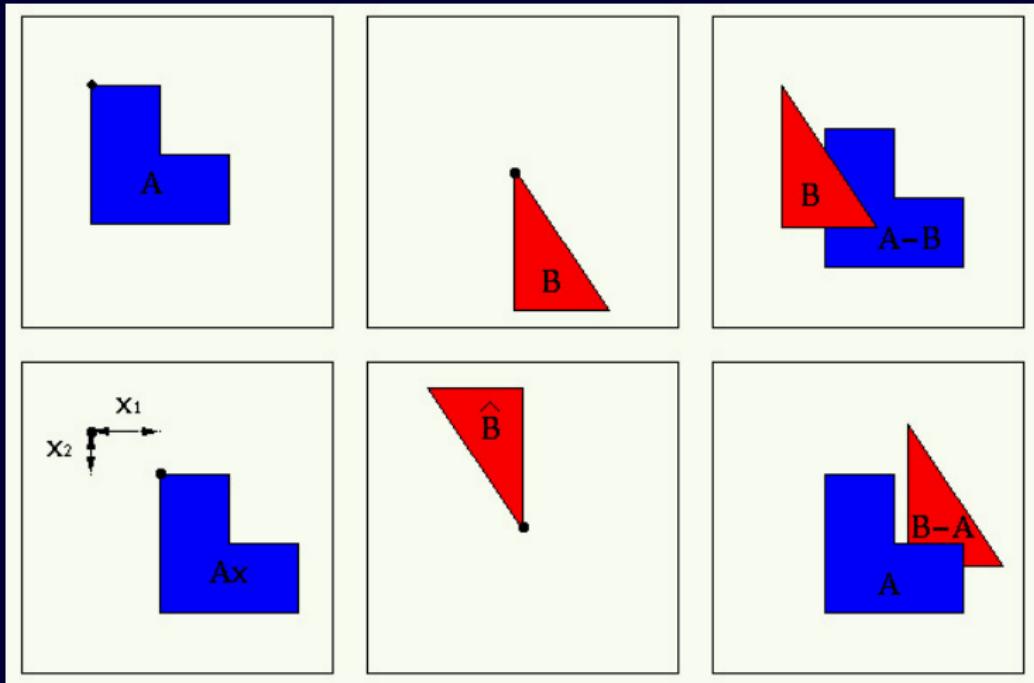
(a) normal brain; (b) 2-level DWT of normal brain; (c) 2-level DWPT of normal brain; (d) AD brain; (e) 2-level DWT of AD brain; (f) 2-level DWPT of AD brain.

Морфологические операции

Основные понятия

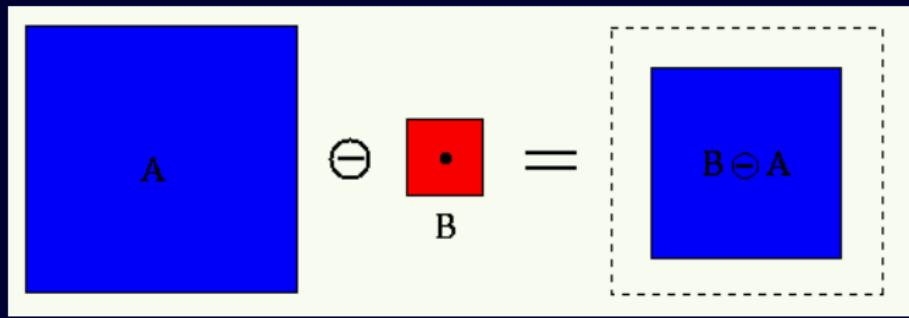
- Пусть A – некоторая область на бинарном изображении, $a = (a_1, a_2) \in A$ – точка, ей принадлежащая; интенсивность в точке a обозначим как $v(a)$.
- **Объект:** $A = \{a \mid v(a) == 1, \forall a \text{ 4/8-connected}\}$.
- **Фон:** $B = \{b \mid b == 0 \cup b \text{ not connected}\}$.
- **Сдвиг:** $A_x = \{c \mid c = a + x, \forall a \in A\}$.
- **Отражение:** $\hat{A} = \{c \mid c = -a, \forall a \in A\}$.
- **Дополнение:** $A^C = \{c \mid c \notin A\}$.
- **Сумма:** $A + B = \{c \mid c \in (A \cup B)\} = A \cup B$.
- **Разность:** $A - B = \{c \mid c \in A, c \notin B\} = A \cap B^C$.
- **Структурный элемент:** подобъект, по которому проводятся морфологические операции.

Морфологические операции



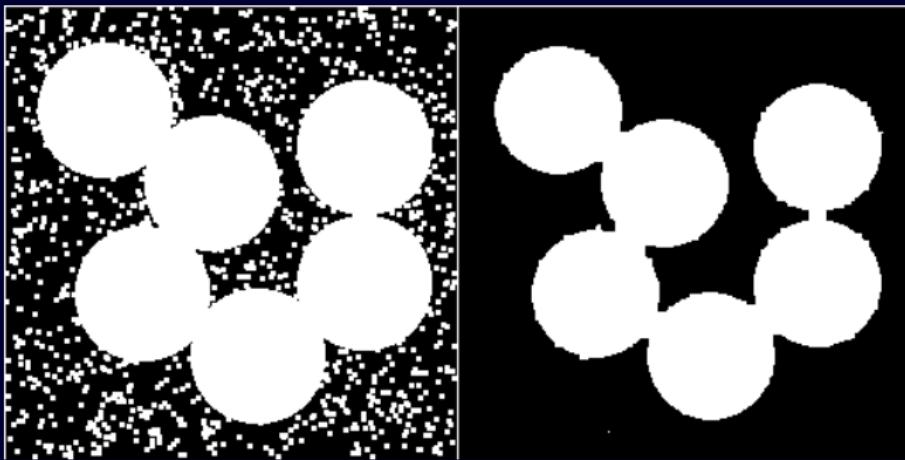
Эрозия (усечение)

$$A \ominus B = \{x \mid B_x \subseteq A\} \text{ или } A \ominus B = \{x \mid B_x \cap A^C = \emptyset\} \text{ или } A \ominus B = \bigcap_{b \in B} A_b$$



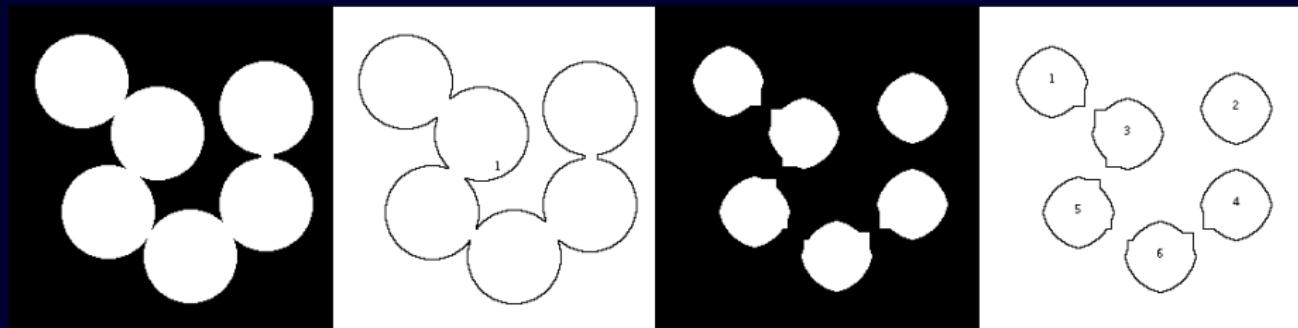
Эрозия (усечение)

$A \ominus B = \{x \mid B_x \subseteq A\}$ или $A \ominus B = \{x \mid B_x \cap A^C = \emptyset\}$ или $A \ominus B = \bigcap_{b \in B} A_b$



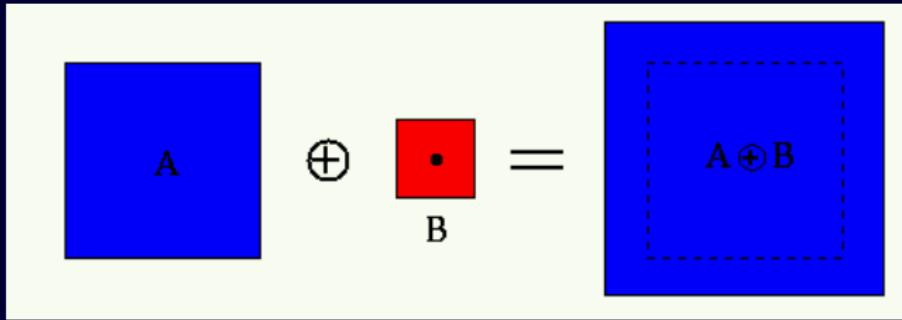
Эрозия (усечение)

$A \ominus B = \{x \mid B_x \subseteq A\}$ или $A \ominus B = \{x \mid B_x \cap A^C = \emptyset\}$ или $A \ominus B = \bigcap_{b \in B} A_b$



Дилатация (наращивание)

$$A \oplus B = \{x \mid \hat{B}_z \cap A \neq \emptyset\} \text{ или } A \oplus B = \bigcup_{b \in B} A_b = \bigcup_{a \in A} B_a$$



Дилатация (наращивание)

$$A \oplus B = \{x \mid \hat{B}_z \cap A \neq \emptyset\} \text{ или } A \oplus B = \bigcup_{b \in B} A_b = \bigcup_{a \in A} B_a$$

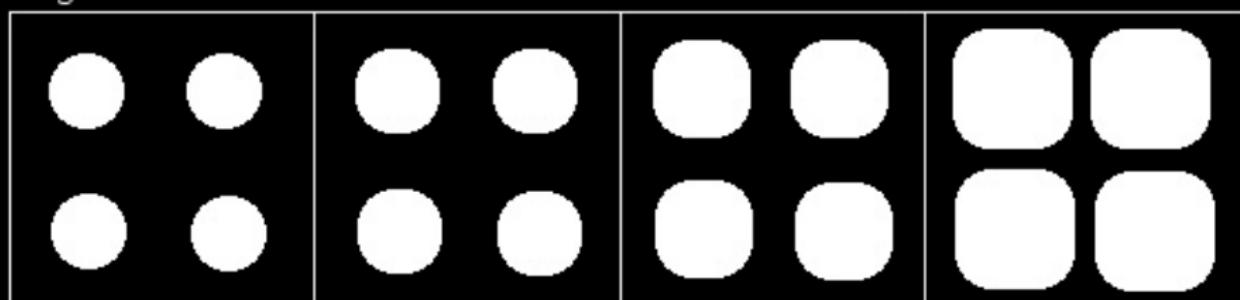
Dilation

Original

2

5

10



Свойства

Коммутативность:

$$A \oplus B = B \oplus A \quad A \ominus B \neq B \ominus A$$

Ассоциативность:

$$A \oplus (B \cup C) = (A \oplus B) \cup (A \oplus C) \quad A \ominus (B \cup C) = (A \ominus B) \cap (A \ominus C)$$

$$(A \ominus B) \ominus C = A \ominus (B \oplus C)$$

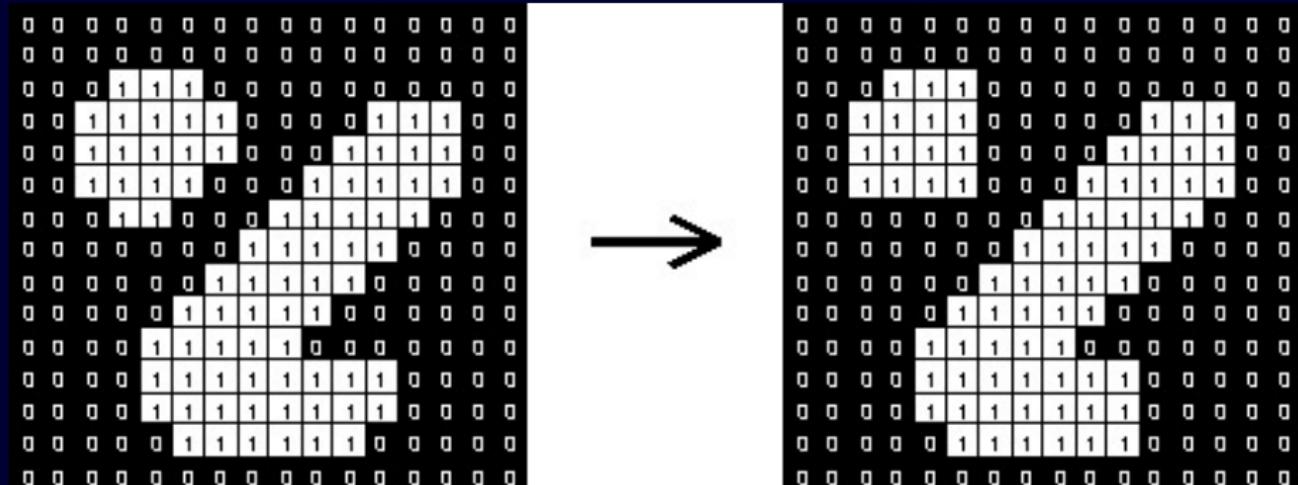
Двойственность:

$$(A \ominus B)^C = A^C \oplus \hat{B} \quad (A \oplus B)^C = A^C \ominus \hat{B}$$



Открытие (размыкание)

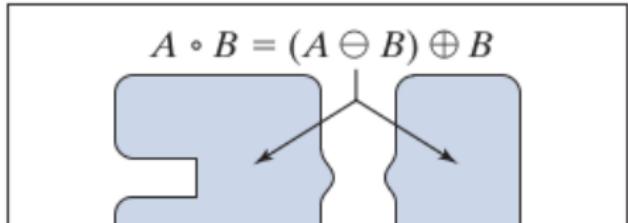
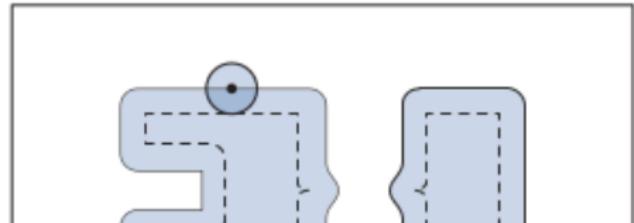
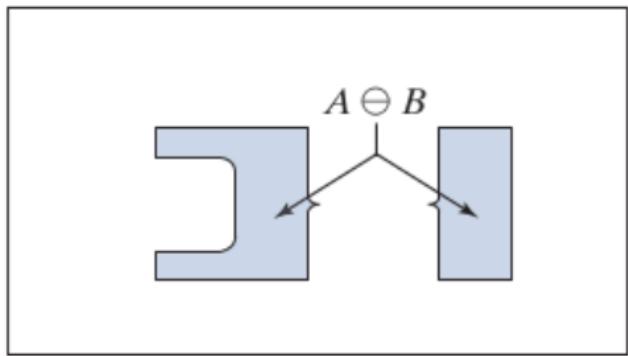
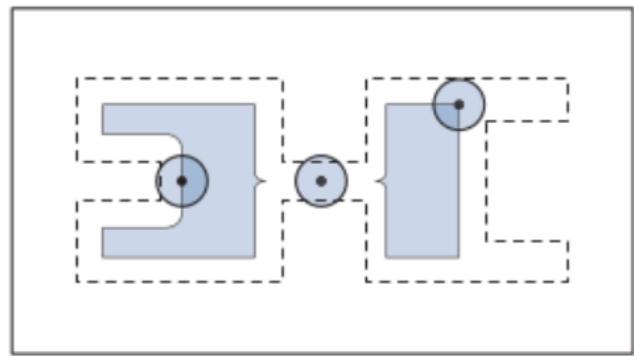
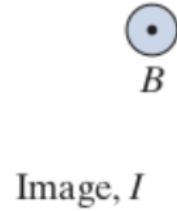
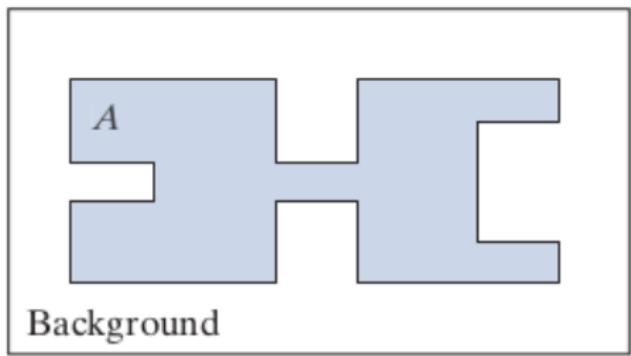
$$A \circ B = (A \ominus B) \oplus B$$

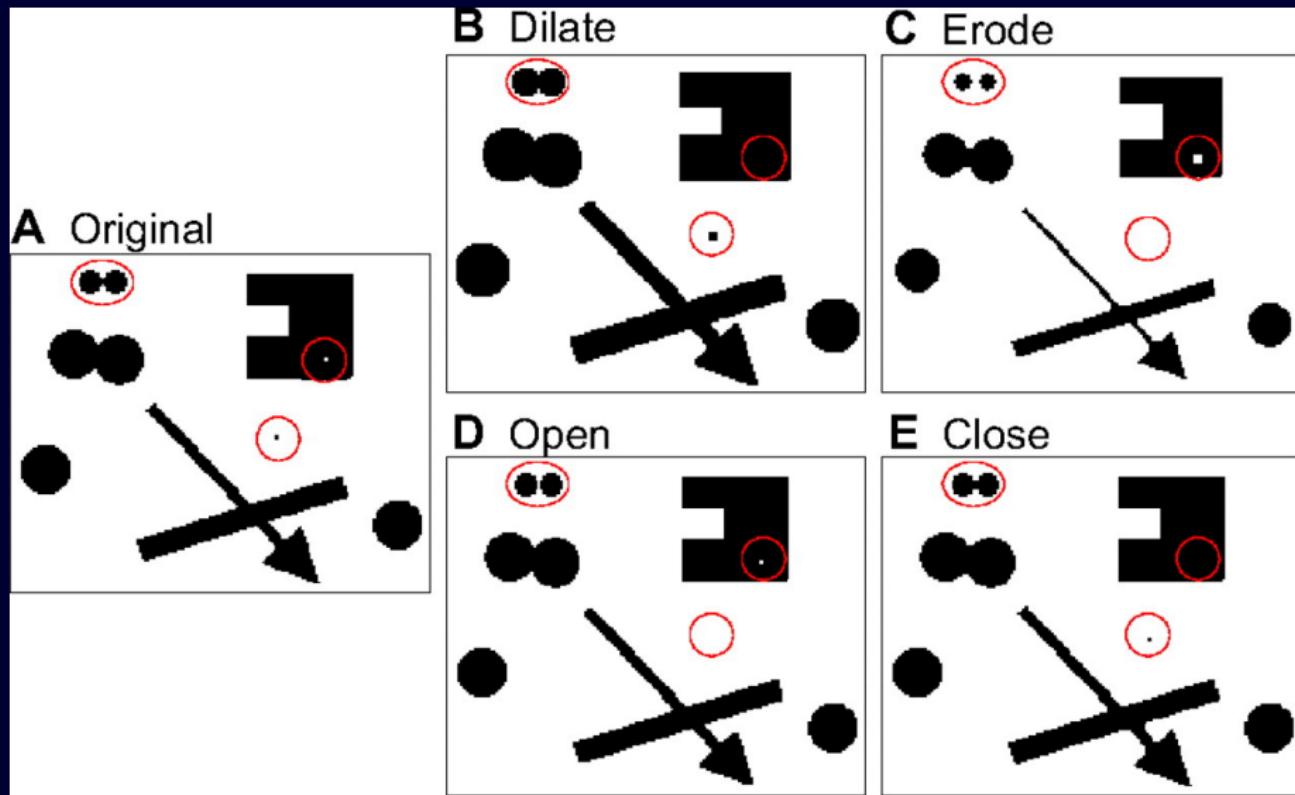


Закрытие (замыкание)

$$A \bullet B = (A \oplus B) \ominus B$$

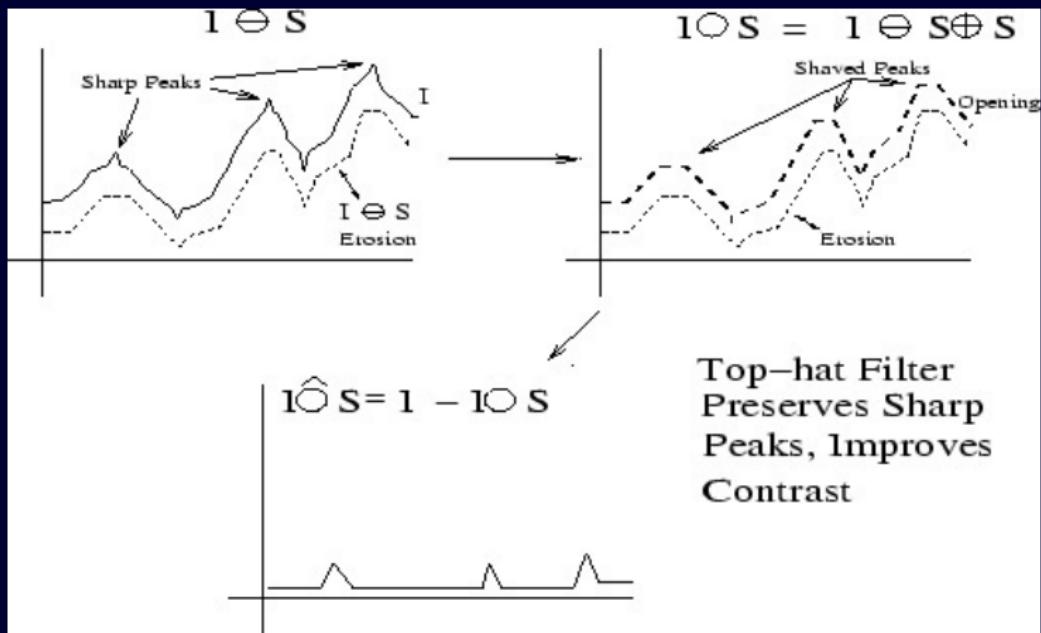






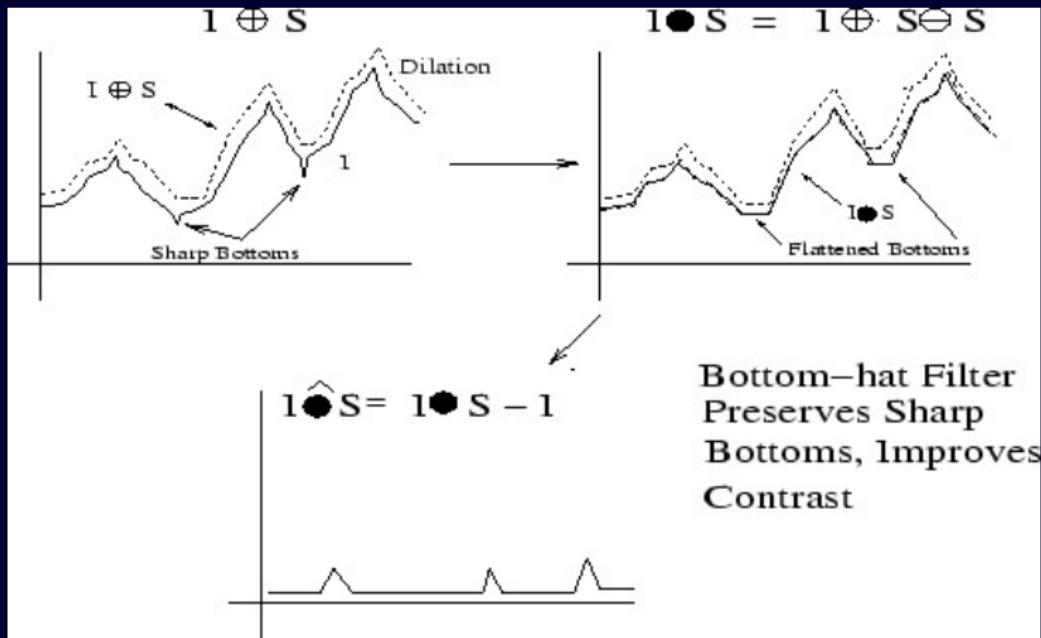
«Top hat» и «Bottom hat»

$$A \hat{\ominus} B = A \setminus (A \circ B), \quad A \hat{\bullet} B = (A \bullet B) \setminus A$$



«Top hat» и «Bottom hat»

$$A \hat{\ominus} B = A \setminus (A \circ B), \quad A \hat{\bullet} B = (A \bullet B) \setminus A$$



Hit-and-miss

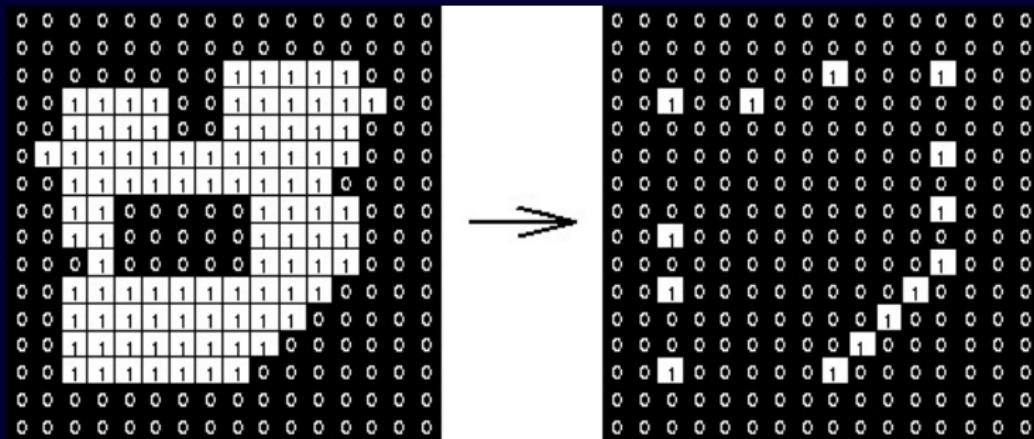
$$A \circledast B = (A \ominus B_1) \cap (A^C \ominus B_2), \quad \text{где}$$
$$B_1 = \{b \mid b \in B, b = 1\}, \quad B_2 = \{\tilde{b} \mid b \in B, b = 0\}$$

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

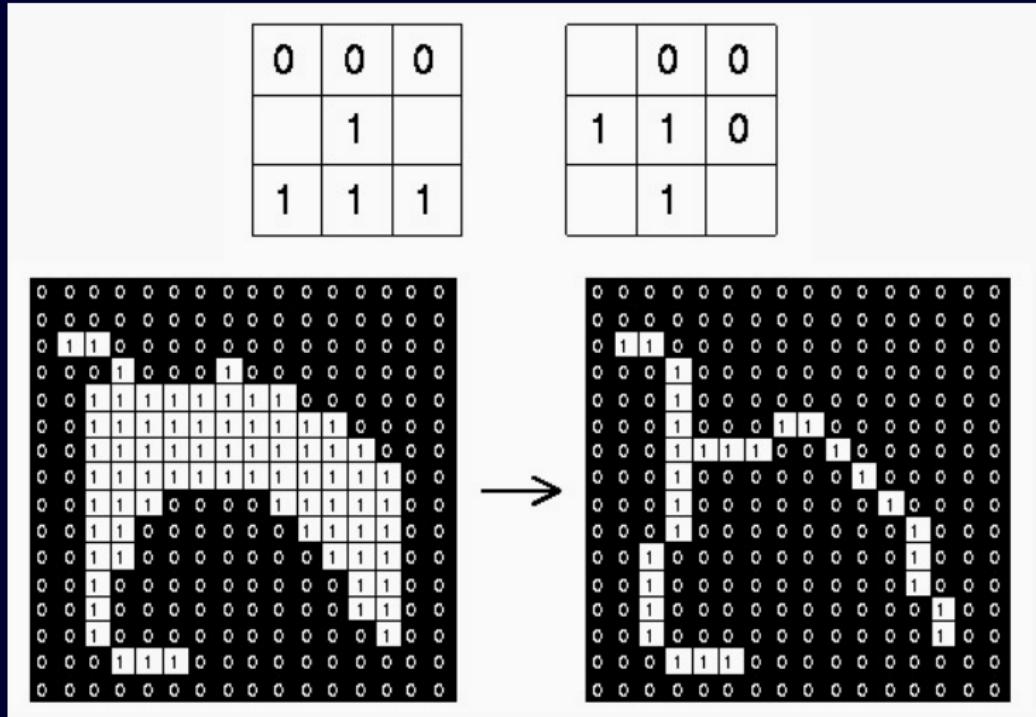
Hit-and-miss

$$A \circledast B = (A \ominus B_1) \cap (A^C \ominus B_2), \quad \text{где}$$

$$B_1 = \{b \mid b \in B, b = 1\}, \quad B_2 = \{\tilde{b} \mid b \in B, b = 0\}$$

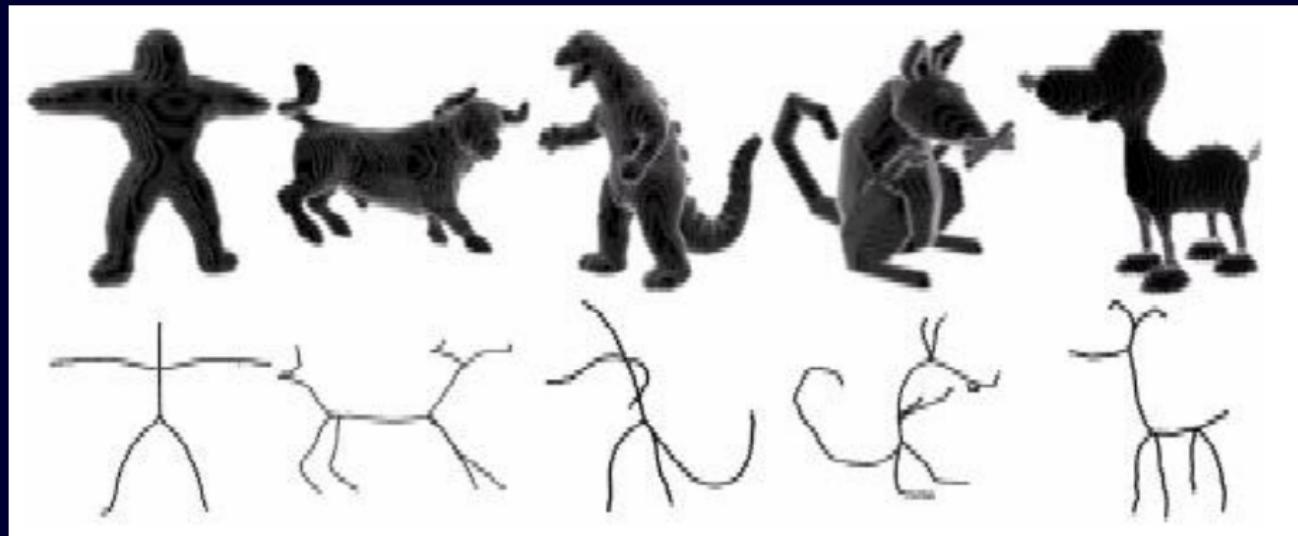


Hit-and-miss



$$S = A \setminus \bigcup_i (A \circledast B_i)$$

Hit-and-miss



Hit-and-miss



Сегментация изображений

Основы

- Сегментация: $\cup_{i=1}^n R_i \cup \cup_{i=1}^n B_i = R$, все R_i связные, B_i – фон.
- $R_i \cap R_j = \emptyset \forall i \neq j$.
- $Q(R_i) = 1, i = \overline{1, n}$, Q – логический предикат.
- $Q(R_i \cup R_j) = 0 \forall i \neq j$.

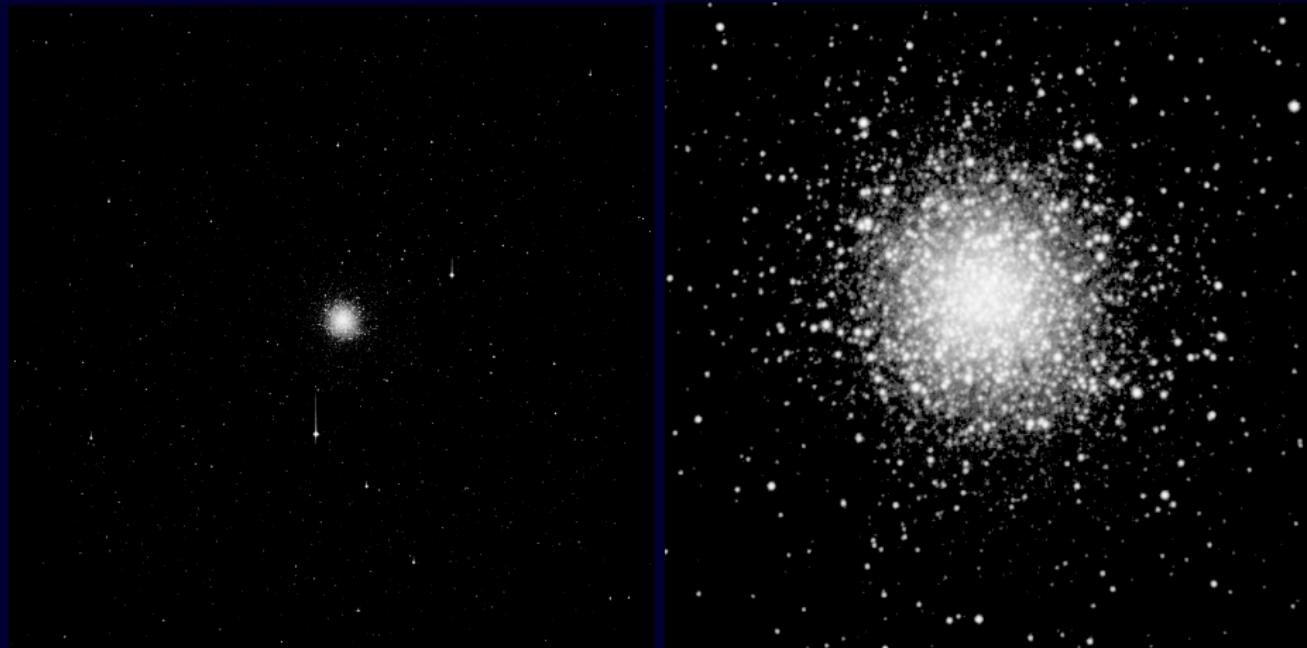
Производные

- $\frac{\partial f}{\partial x} \equiv f'_x = f(x+1) - f(x)$
- $\frac{\partial^2 f}{\partial x^2} \equiv f''_x = f'_x(x+1) - f'_x(x) = f(x+2) + f(x) - 2f(x+1)$
- $\nabla^2 f(x, y) = f''_x(x, y) + f''_y(x, y) \Rightarrow$
 $\nabla^2 f(x, y) = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)$



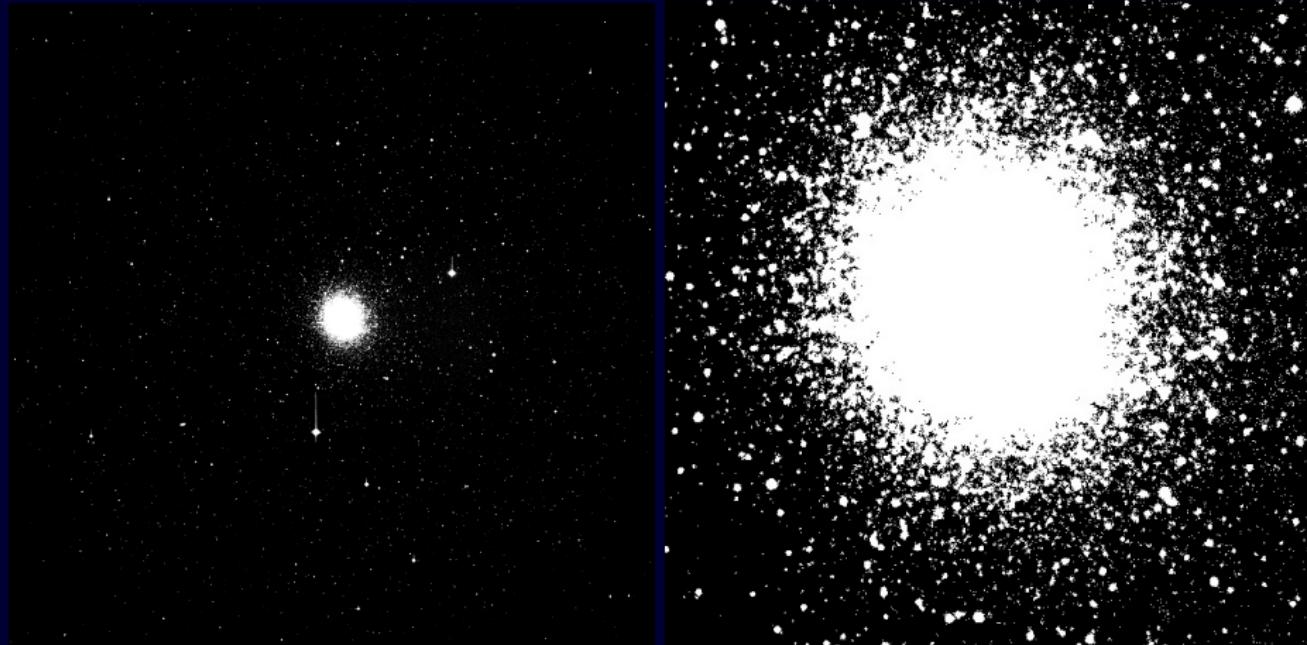
Примеры (М13)

Оригинал:



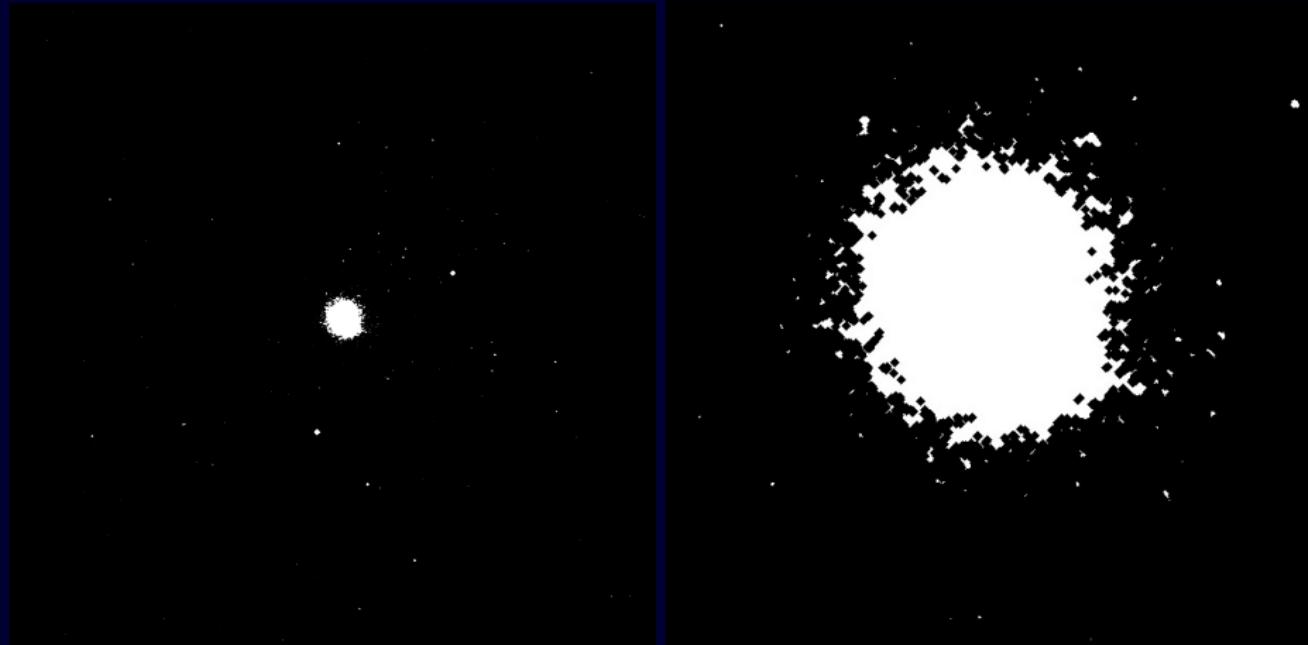
Примеры (M13)

Бинаризация по постоянному порогу:



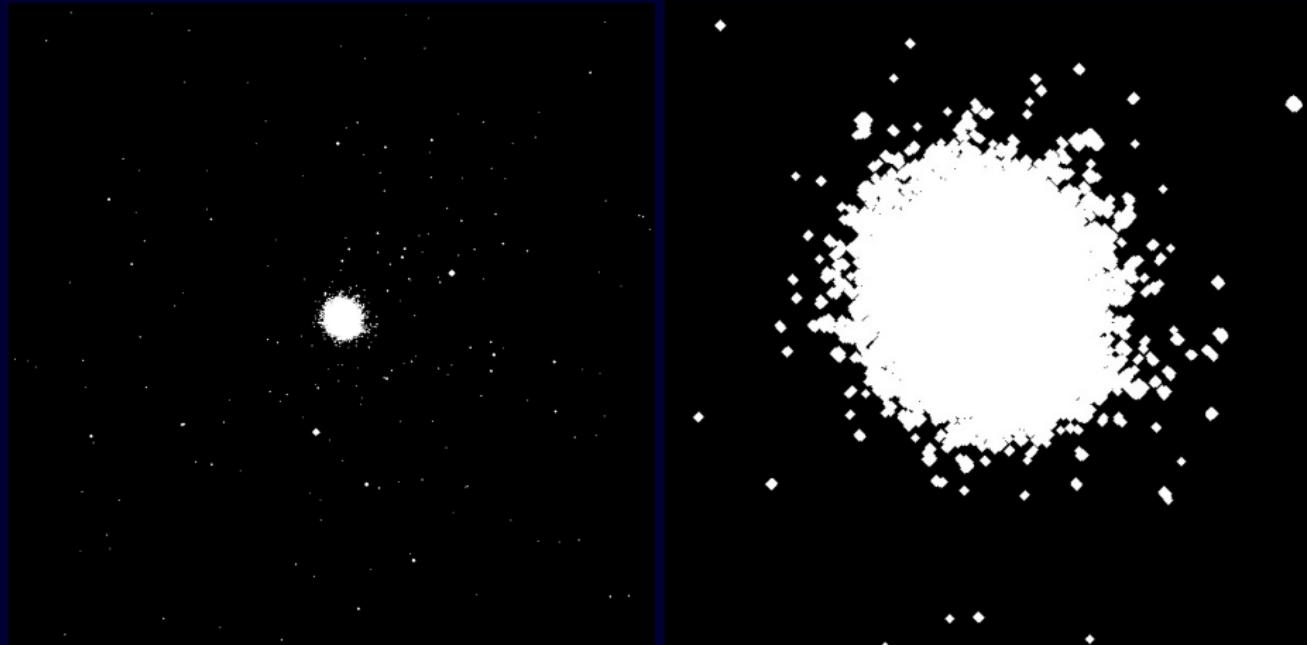
Примеры (М13)

Четырехкратная эрозия:



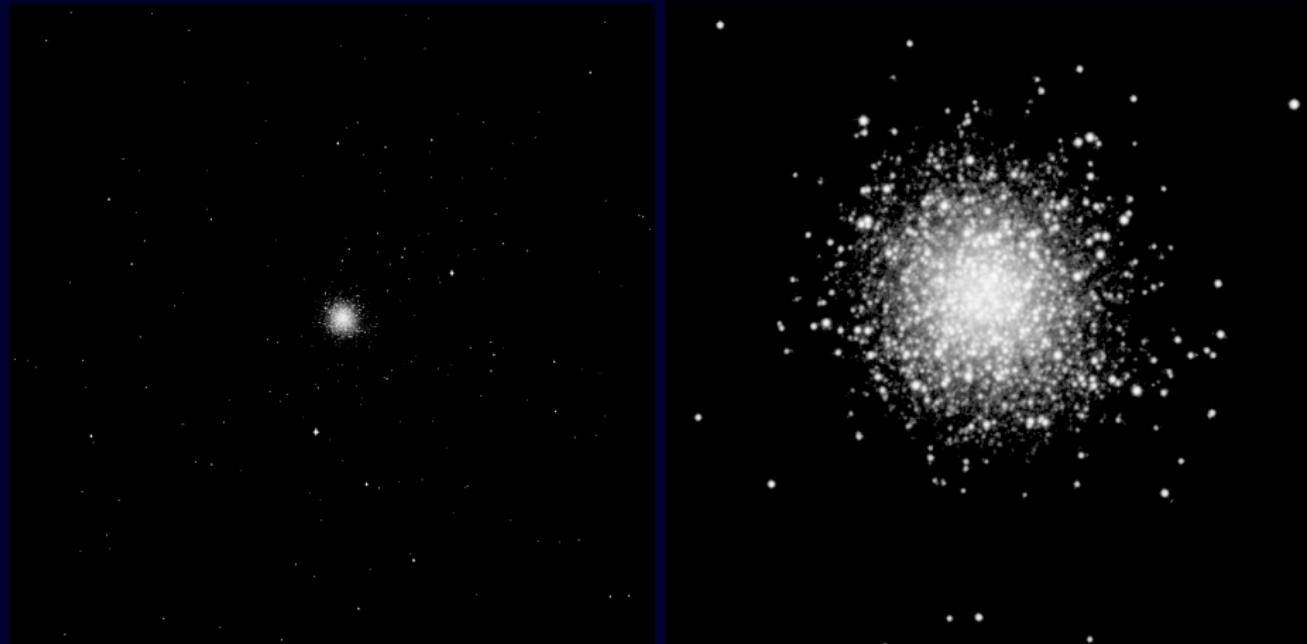
Примеры (М13)

Четырехкратное размыкание:



Примеры (M13)

Оригинал с предыдущей маской:



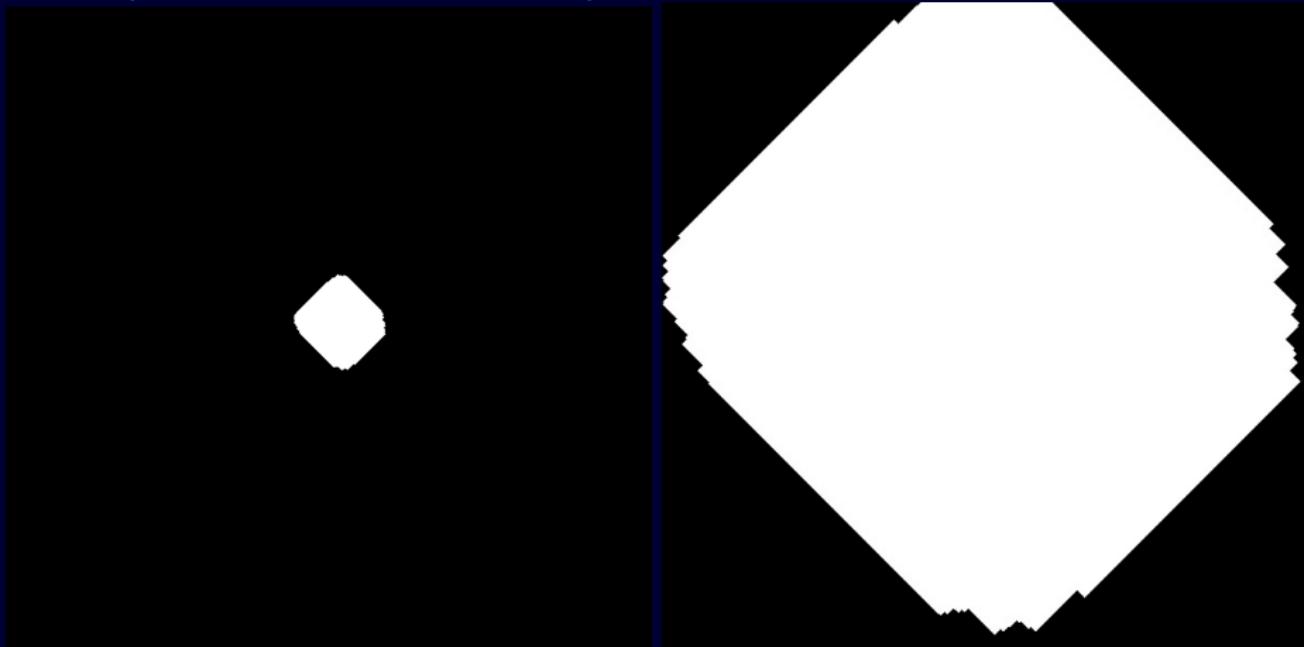
Примеры (М13)

Двадцатипятикратная эрозия:



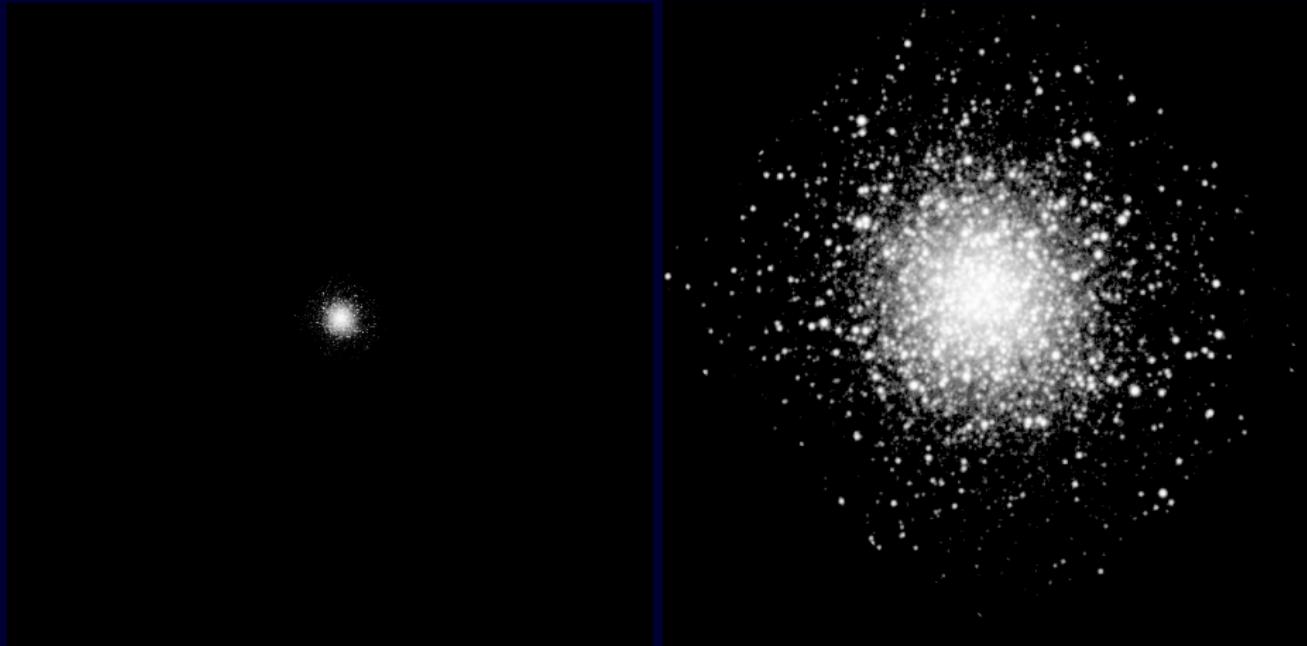
Примеры (М13)

Маска (25 эрозий и 200 дилатаций):



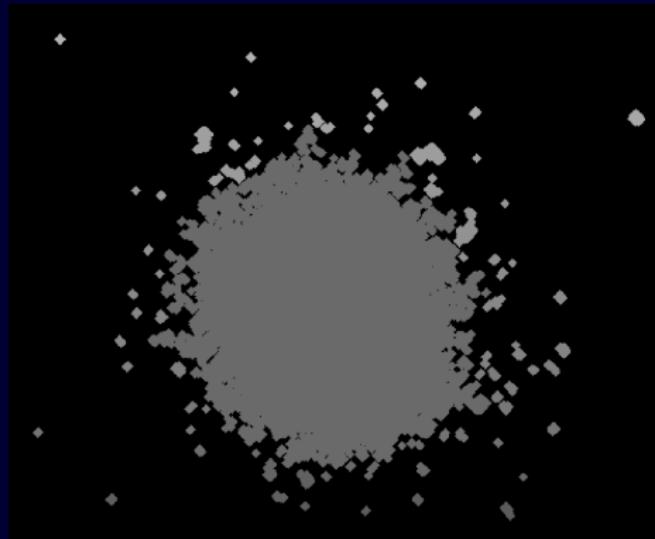
Примеры (М13)

Оригинал с предыдущей маской:



Примеры (М13)

Выделенные объекты (размыкание $\times 4$ и $\times 10$; 237 и 9 объектов в поле соответственно):



Обнаружение линий, точек и перепадов

Точки — лапласиан, линии, перепады — градиент

-1	+1	+1
-1	-2	+1
-1	+1	+1

0°

+1	+1	+1
-1	-2	+1
-1	-1	+1

45°

Prewitt

Обнаружение линий, точек и перепадов

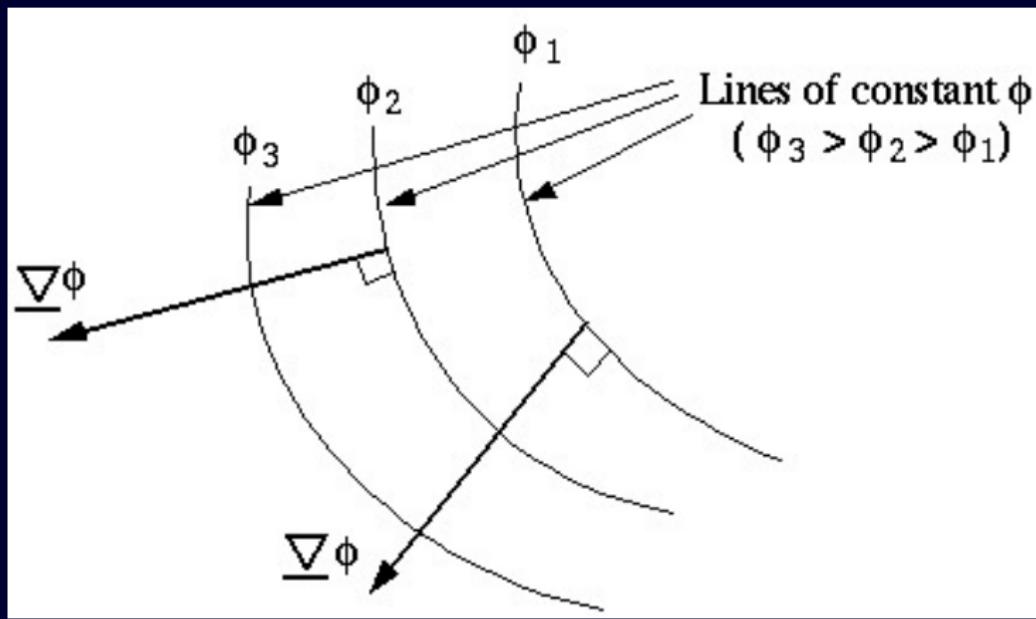
	0°			45°		
Sobel	-1	0	1	0	1	2
Kirsch	-3	-3	5	-3	5	5
Rabinsan	-1	0	1	0	1	1
	-1	0	1	-1	0	1
	-1	0	1	-1	-1	0



Обнаружение линий, точек и перепадов

Градиент

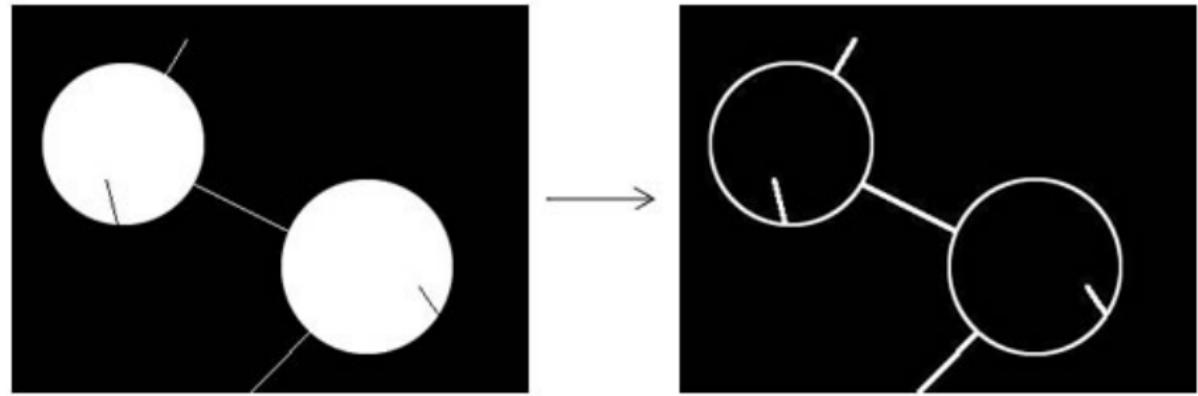
$$\nabla \vec{f} = (f'_x, f'_y) = (f(x+1, y) - f(x, y), f(x, y+1) - f(x, y))$$



Выделение границ

Морфологический градиент

$$\beta(A) = A \setminus (A \oplus B) \quad \beta'(A) = (A \oplus B) \setminus A \quad \beta''(A) = (A \oplus B) \setminus (A \ominus B)$$



Morphological Gradient

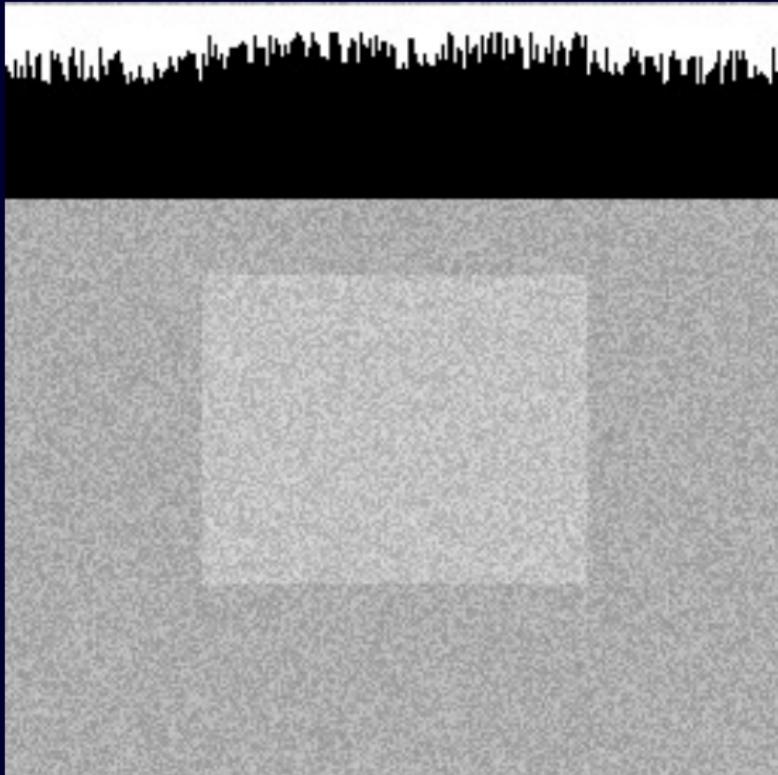
Выделение границ

Canny

- ① Размытие изображения гауссовым фильтром.
- ② Вычисление частных производных I'_x и I'_y (Робертс, Собель, Прюитт, LoG, DoG...) и компонентов градиента: $M = \sqrt{(I'_x)^2 + (I'_y)^2}$,
 $\theta = \arctg I'_y / I'_x$.
- ③ Пороговое преобразование M : $M_T = M$, если $M > T$, иначе $M_T = 0$.
- ④ Обнуление немаксимальных M_T по направлению θ (по двум соседям).
- ⑤ Получение двух пороговых значений: M_{T_1} и M_{T_2} ; $T_1 < T_2$.
- ⑥ Заполнение пропусков в M_{T_2} по соседним значениям в M_{T_1} .

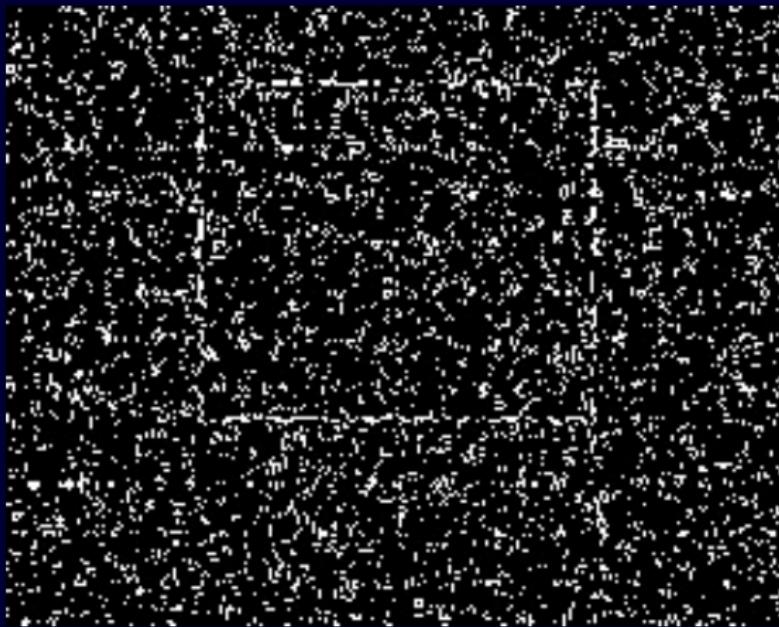


Выделение границ



Образец

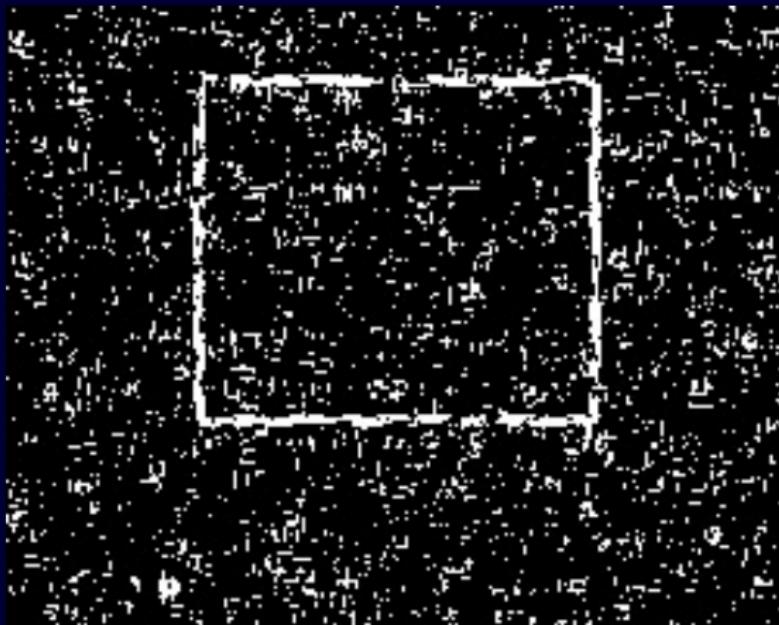
Выделение границ



Sobel

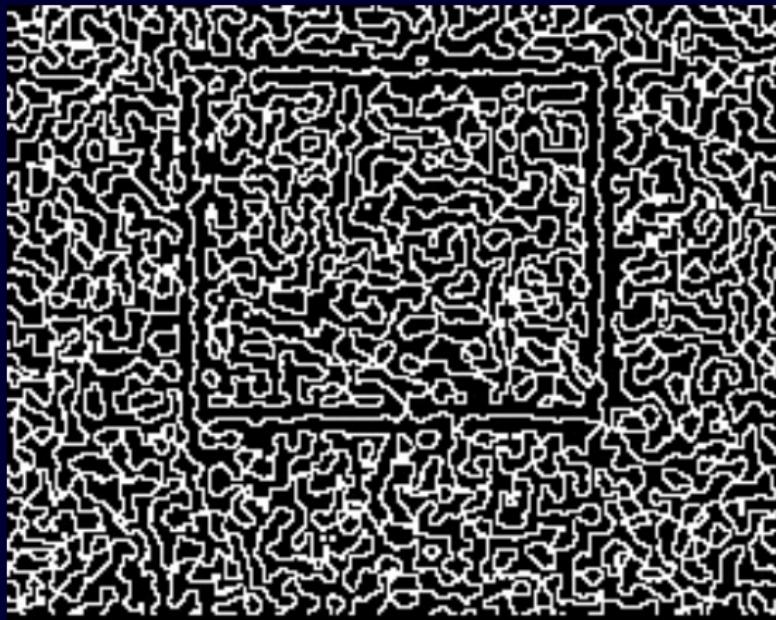


Выделение границ



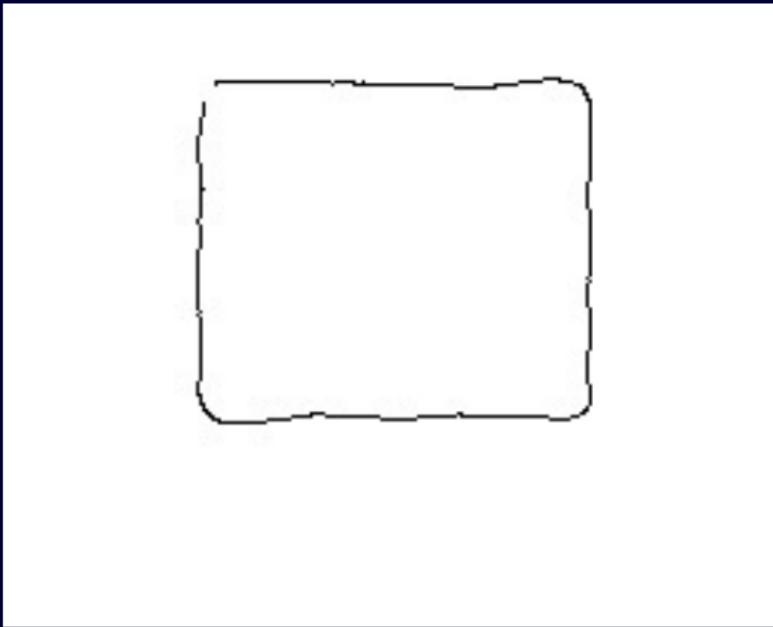
Prewitt

Выделение границ



DoG

Выделение границ



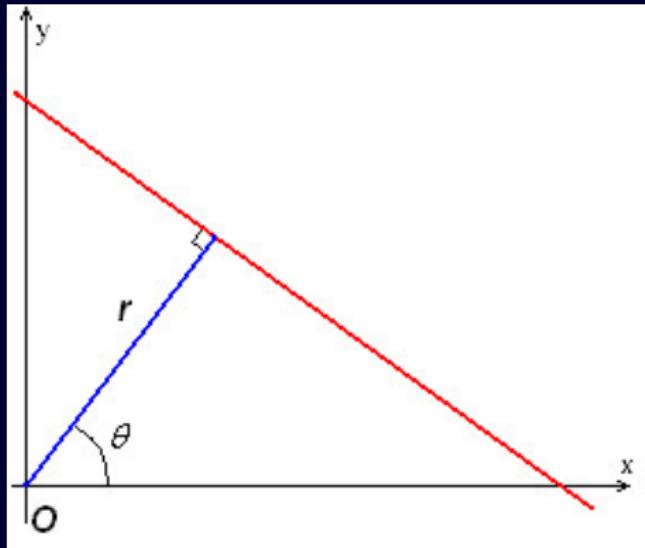
Canny, $\sigma = 5$, $T_1 = 0.8$, $T_2 = 0.95$



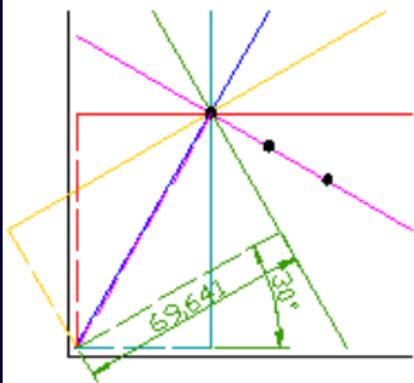
Обнаружение прямых и окружностей

Преобразование Хафа

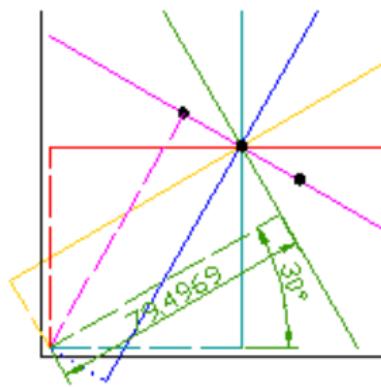
$$r = x \cos \theta + y \sin \theta$$



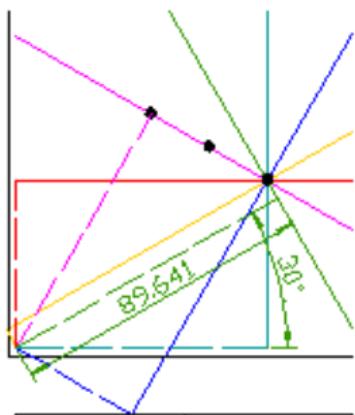
Обнаружение прямых и окружностей



Angle	Dist.
0	40
30	69.6
60	81.2
90	70
120	40.6
150	0.4

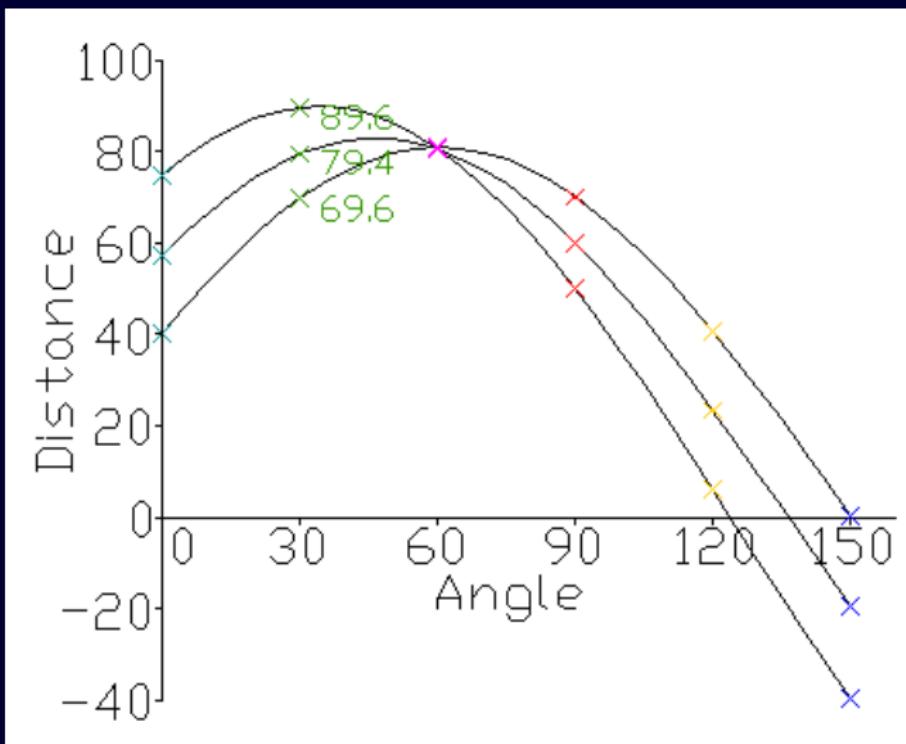


Angle	Dist.
0	57.1
30	79.5
60	80.5
90	60
120	23.4
150	-19.5



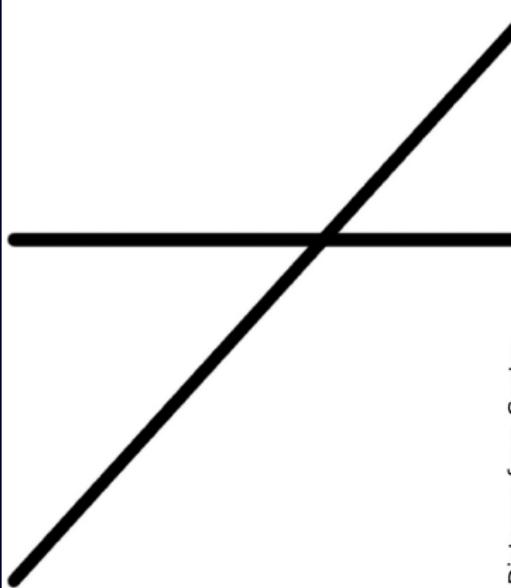
Angle	Dist.
0	74.6
30	89.6
60	80.6
90	50
120	6.0
150	-39.6

Обнаружение прямых и окружностей

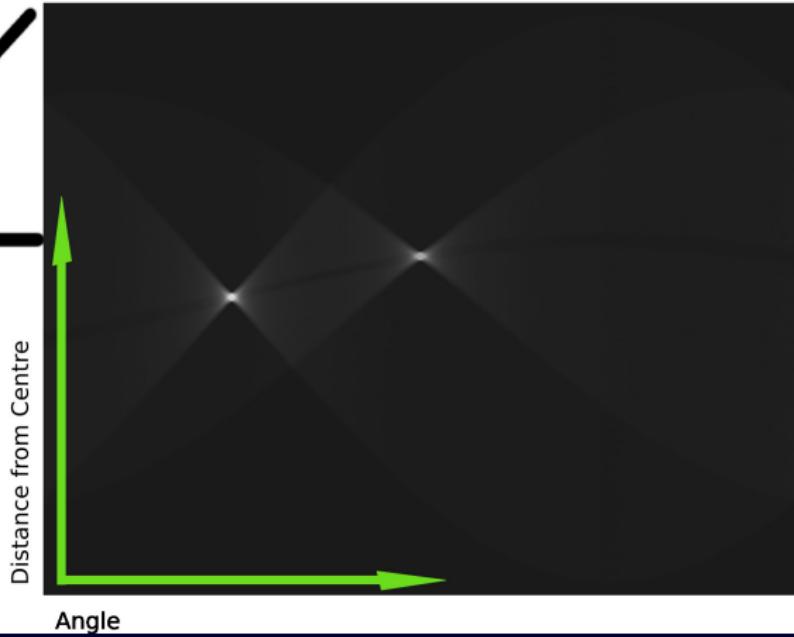


Обнаружение прямых и окружностей

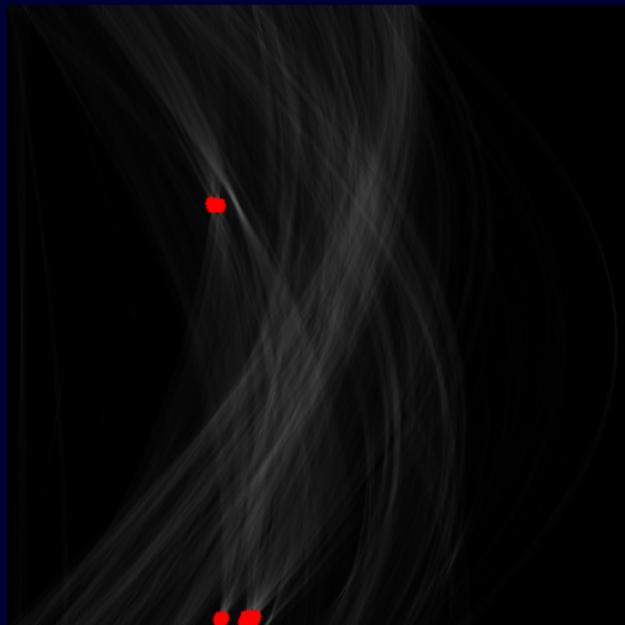
Input Image



Rendering of Transform Results



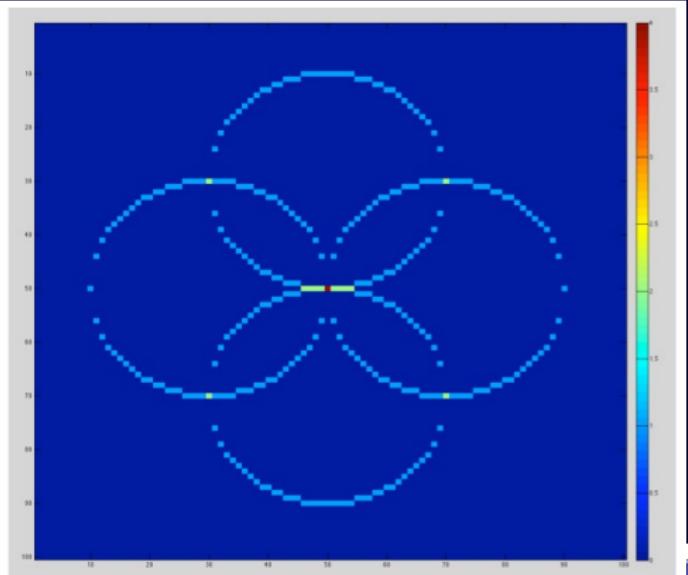
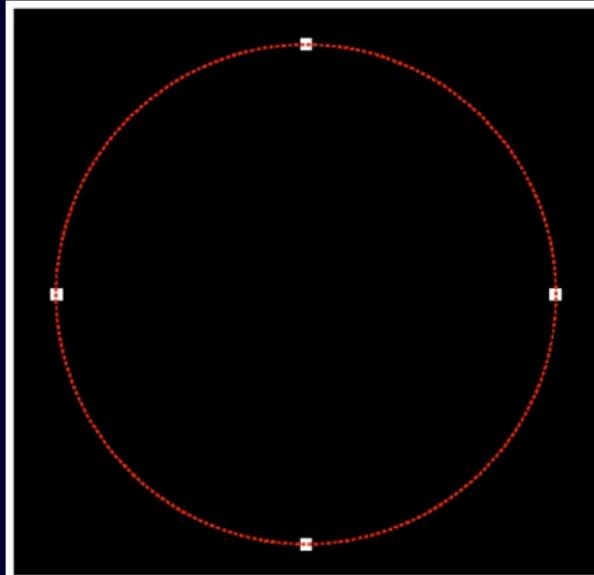
Обнаружение прямых и окружностей



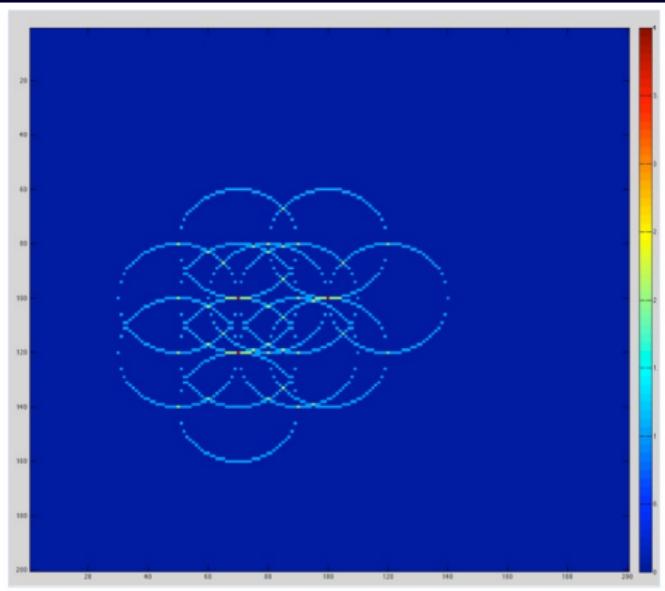
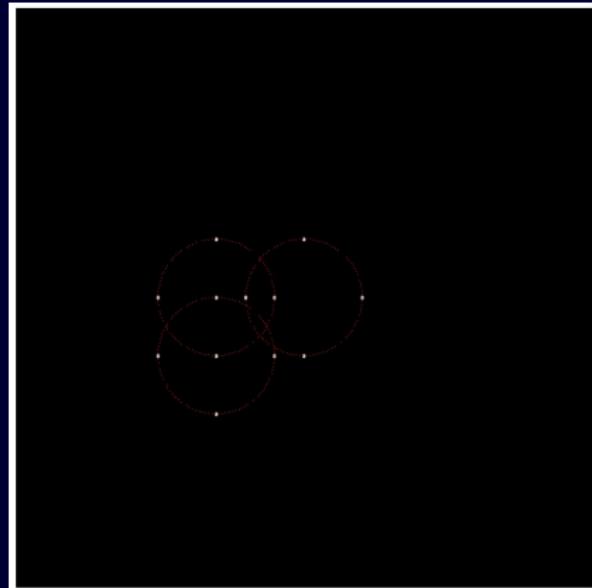
Обнаружение прямых и окружностей

Преобразование Хафа для поиска окружностей

$$(x - x_c)^2 + (y - y_c)^2 = r^2$$



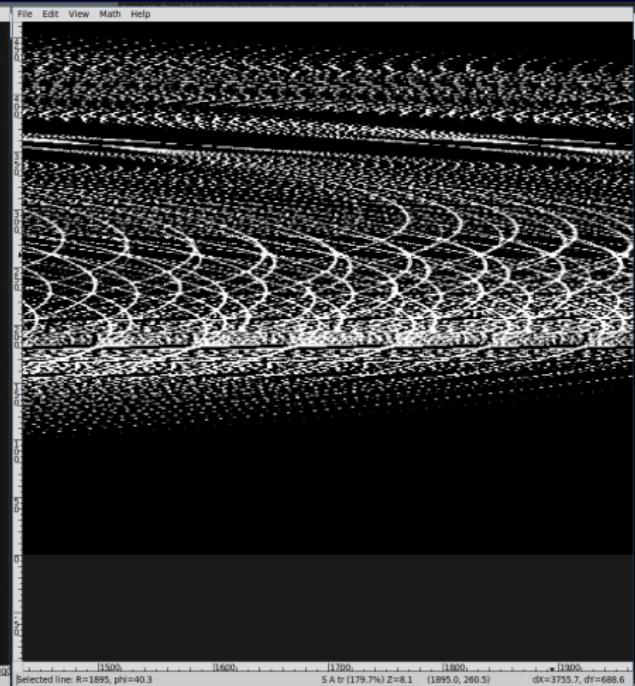
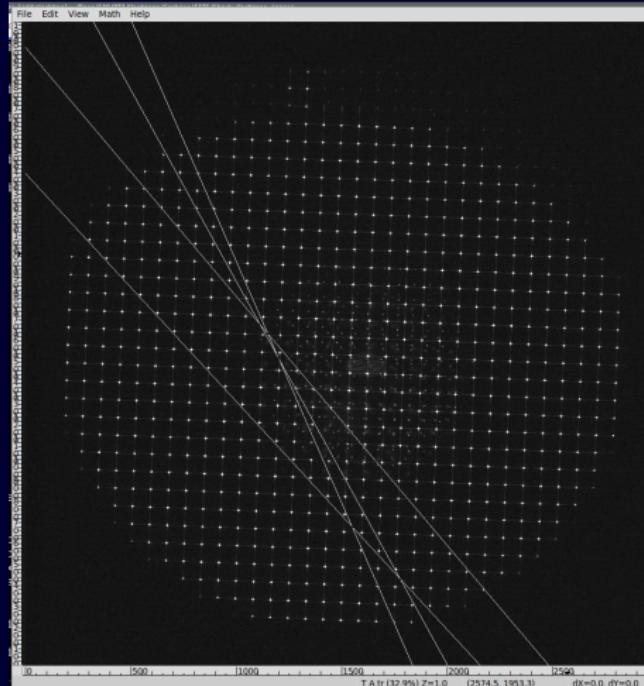
Обнаружение прямых и окружностей



Трехмерный массив в случае неизвестных центра и радиуса.

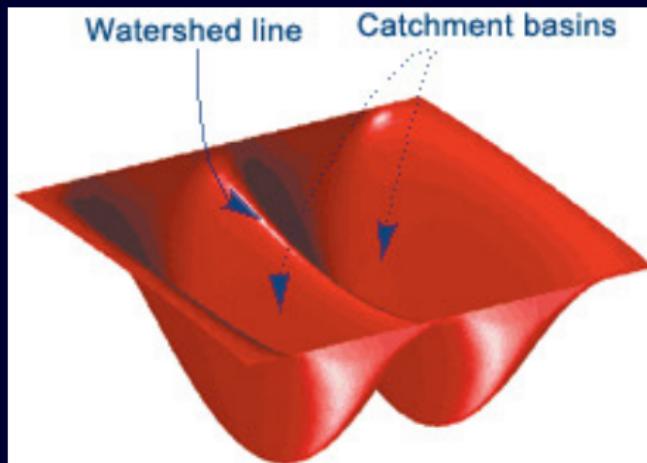


Пример: датчик волнового фронта



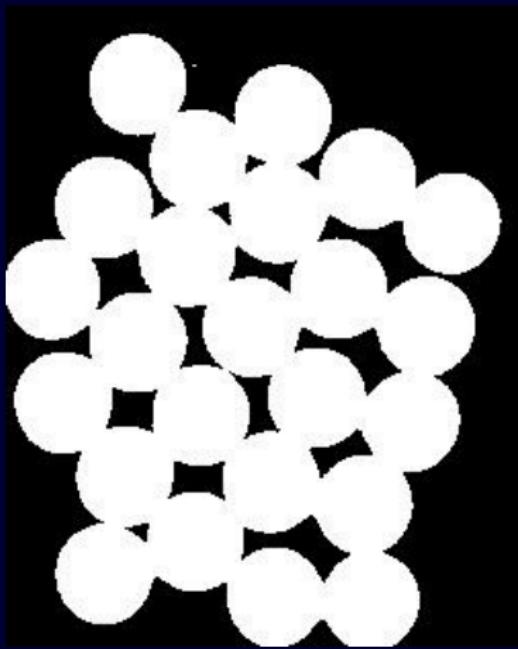
Сегментация по морфологическим водоразделам

Бинарные изображения: итеративные дилатации с построением перегородок в местах образовавшихся пересечений.



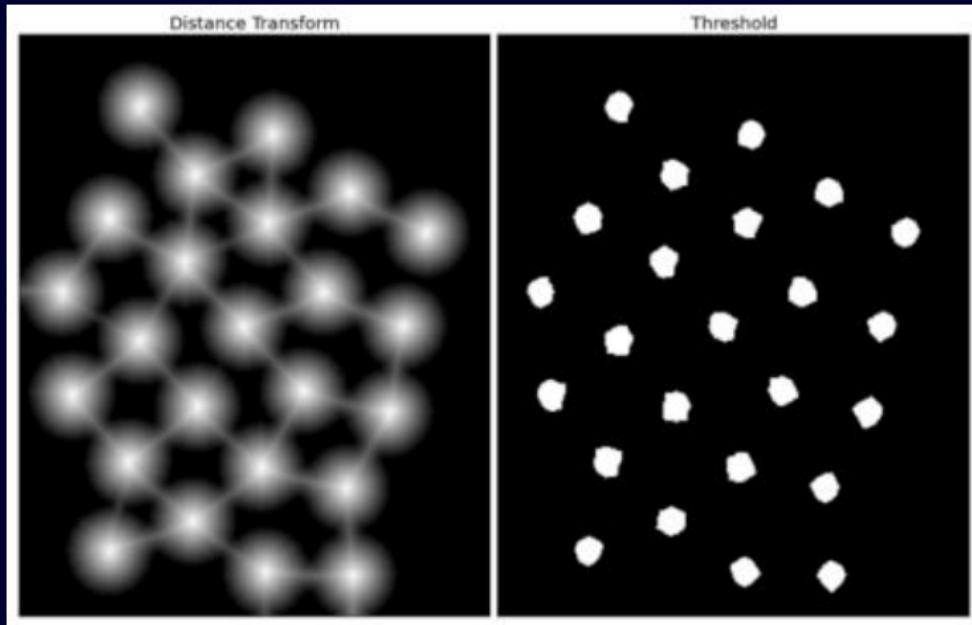
Сегментация по морфологическим водоразделам

Бинарные изображения: преобразования расстояний



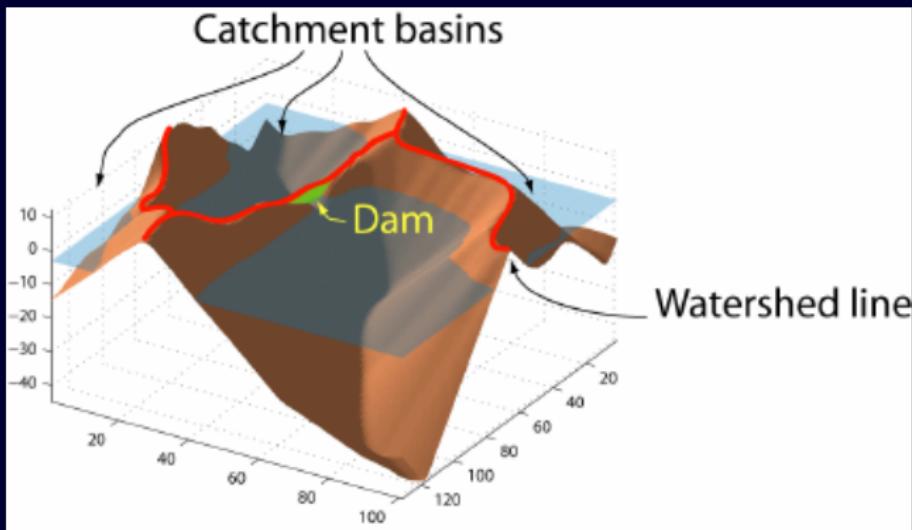
Сегментация по морфологическим водоразделам

Бинарные изображения: преобразования расстояний



Сегментация по морфологическим водоразделам

В общем случае: различные алгоритмы заполнения.



Деконволюция

$$I(x, y) = P(x, y) * O(x, y) + N(x, y), \quad P - \text{PSF} \quad \text{или}$$

$$\mathcal{F}(I) = \mathcal{F}(O) \cdot \mathcal{F}(P) + \mathcal{F}(N) \quad \Rightarrow \quad \mathcal{F}(O) = \frac{\mathcal{F}(I) - \mathcal{F}(N)}{\mathcal{F}(P)}$$

Наименьшие квадраты: $\mathcal{F}(O) = \frac{\mathcal{F}(P)^* \mathcal{F}(I)}{|\mathcal{F}(P)|^2}$

Регуляризация Тихонова, $\min(J_T)$ (H – HPF): $J_T = ||I - P * O|| - \lambda ||H * O||$,

$$\mathcal{F}(O) = \frac{\mathcal{F}(P)^* \mathcal{F}(I)}{|\mathcal{F}(P)|^2 + \lambda |\mathcal{F}(H)|^2}$$



Деконволюция

Регуляризация по Байесу

$$p(O|I) = \frac{p(I|O) \cdot p(O)}{p(I)}$$

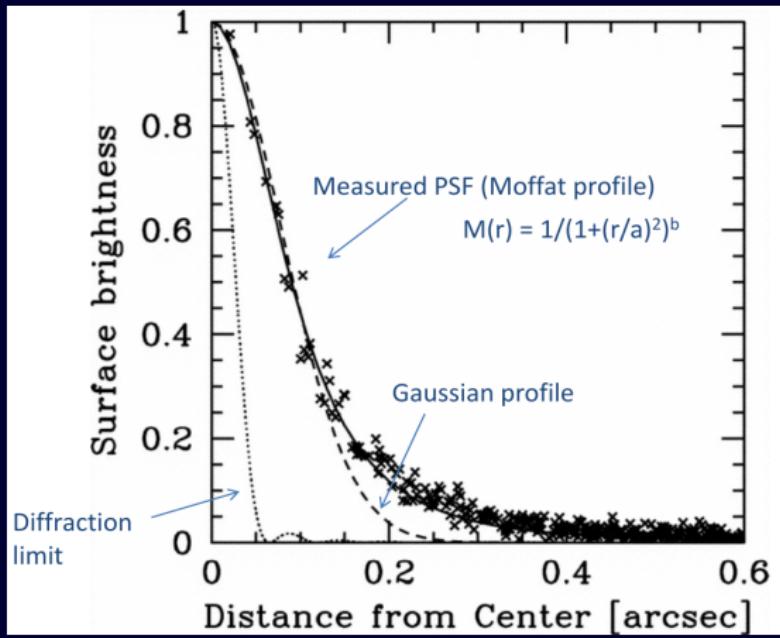
Maximum likelihood: $\text{ML}(O) = \max_O p(I|O)$

Maximum-a-posteriori solution: $\text{MAP}(O) = \max_O p(I|O) \cdot p(O)$

- Итерационная регуляризация
- Вейвлет-регуляризация
- ...



Функция рассеяния точки



Функция рассеяния точки

- Гаусс: $f(x) = f_0 \exp\left(\frac{-(x - x_0)^2}{2\sigma^2}\right)$, FWHM $\approx 2.355\sigma$
- Моффат: $f(x) = f_0 \left(1 + \frac{(x - x_0)^2}{\alpha^2}\right)^{-\beta}$, FWHM $\approx 2\alpha\sqrt{2^{1/\beta} - 1}$
- Фрид: $\mathcal{F}(f) \propto \exp\left[-(bu)^{5/3}\right]$, FWHM $\approx 2.921b$ (Моффат с $\beta = 4.765$, типичные же $\beta = 2.5 \dots 4.5$).

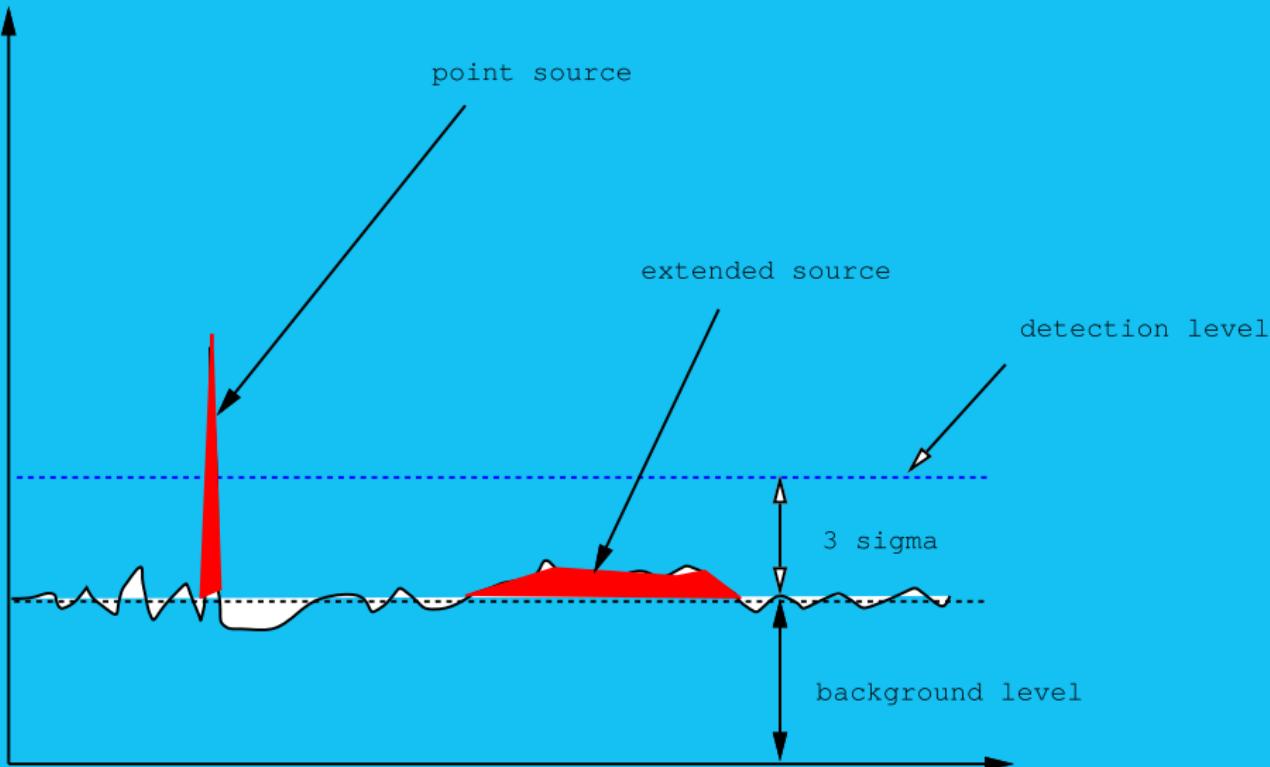
Обнаружение

Простейший алгоритм

- 1 Вычисление и вычитание фона
- 2 Свертка с маской и бинаризация
- 3 Обнаружение связных областей
- 4 Уточнение фона, goto 1
- 5 Классификация, фотометрия и т.п.



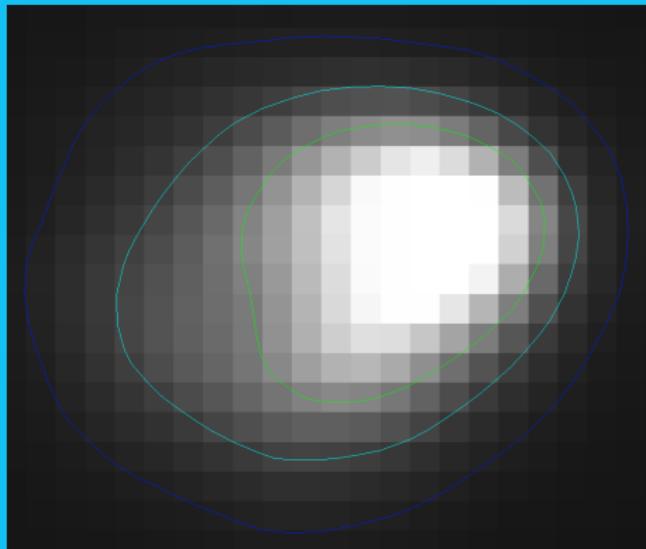
Flux



Изофоты

Метод шагающих квадратов

Бинаризуем изображение по заданному порогу. По соседям каждого пикселя вычисляем битовую маску ($0 \div 15$). От точки $1 \div 14$ строим изолинию, соответственно меняя значения в пикселях маски. Каждый узел изолинии — линейная или другая интерполяция интенсивности в пикселях оригинала.



Изофоты

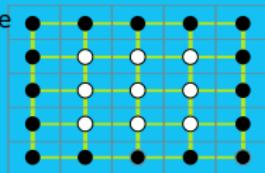
1	1	1	1	1
1	2	3	2	1
1	3	3	3	1
1	2	3	2	1
1	1	1	1	1

Threshold
with iso-value



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

Binary image
to cells



Give every cell a
number based on
which corners are
true/false

8
4
1
2



13	12	12	14	
9	0	0	6	
9	0	0	6	
11	3	3	7	

Look up the contour
lines in the database
and put them in
the cells



1	1	1	1	1
1	2	3	2	1
1	3	3	3	1
1	2	3	2	1
1	1	1	1	1

Look at the original
values and use linear
interpolation to
determine a
more accurate position
of all the line end-points

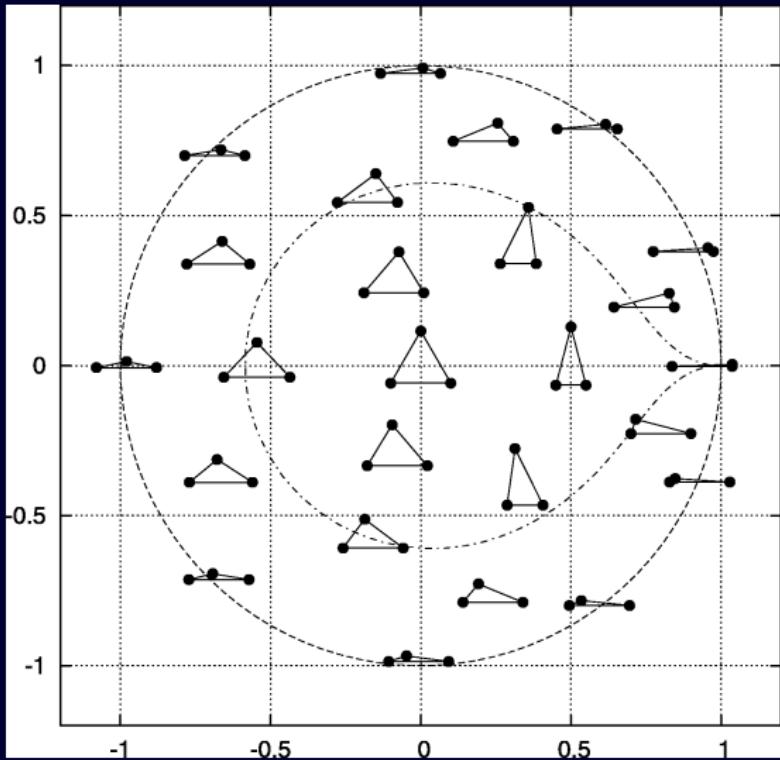


1	1	1	1	1
1	2	3	2	1
1	3	3	3	1
1	2	3	2	1
1	1	1	1	1

Look-up table contour lines

Case 0	Case 1	Case 2	Case 3
Case 4	Case 5	Case 6	Case 7
Case 8	Case 9	Case 10	Case 11
Case 12	Case 13	Case 14	Case 15

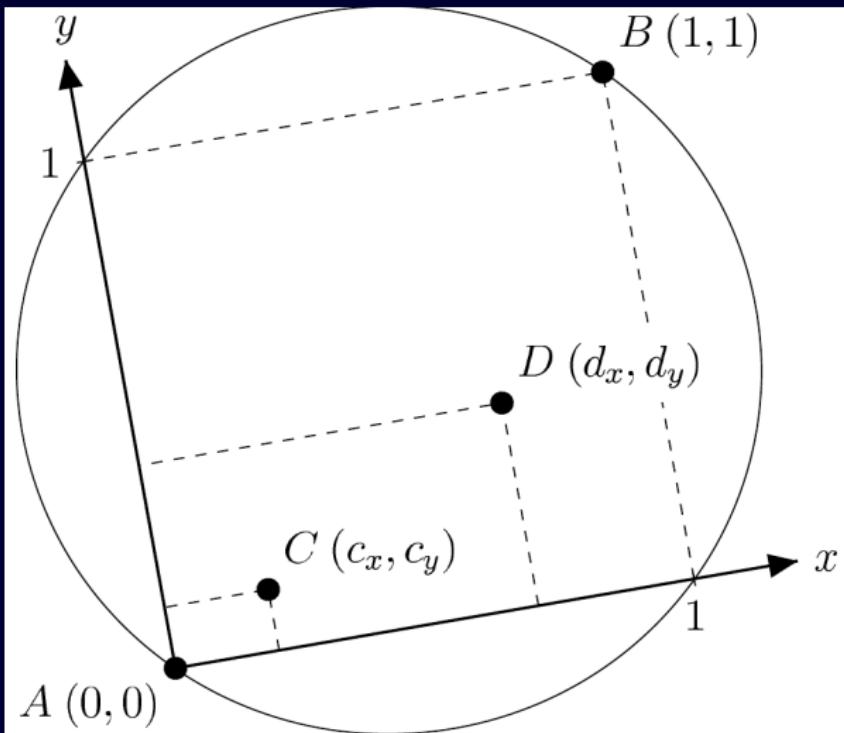
WCS-привязка



A. Pál, G.Á. Bakos. PASP 118: 1474–1483, 2006.



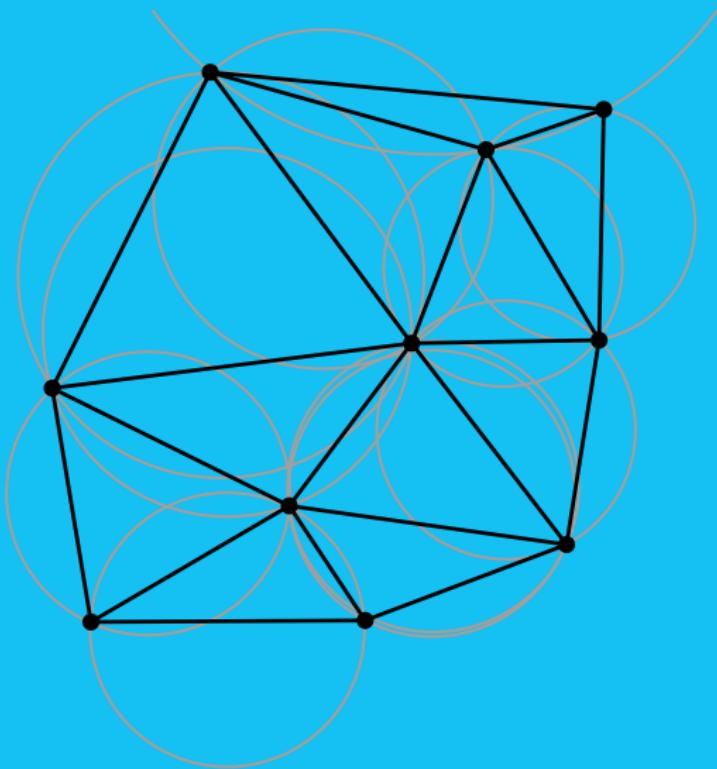
WCS-привязка



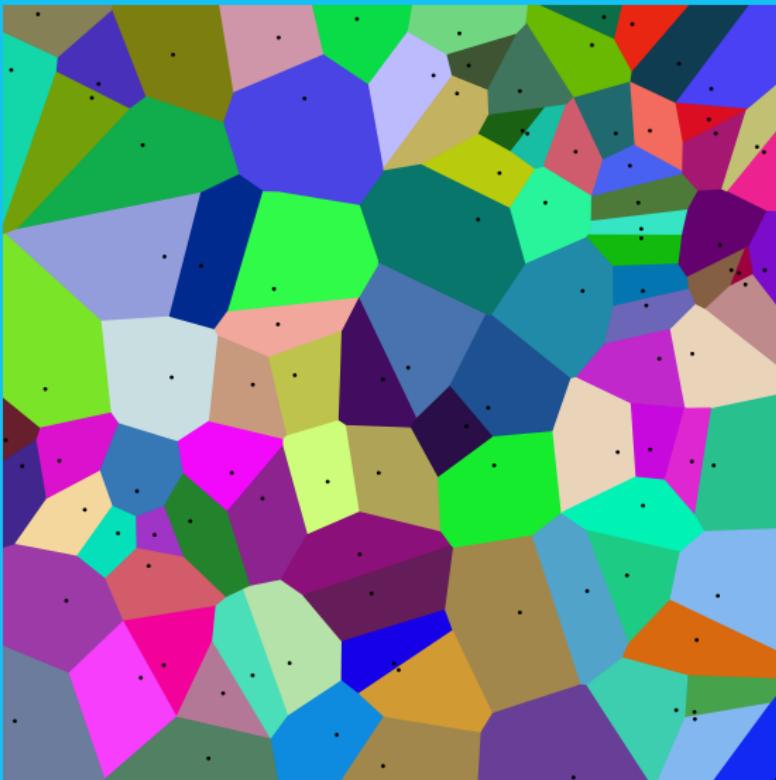
WCS-привязка

- Положение нескольких звезд характеризуется параметром, инвариантным к зеркалированию, масштабированию, вращению и переносу. Устойчивым к шуму.
- Квадрату ABCD соответствует четырехмерный код в относительных координатах точек C и D.
- Проблема вырождения: при смене порядка A, B или C, D код «отражается».
- На небе строится сетка с масштабируемым шагом, по каталожным данным в ее ячейках определяются квадраты с ниспадающей яркостью звезд.
- Полученный набор кодов позволяет идентифицировать участки неба вплоть до самых мелких масштабов (нужны хотя бы четыре звезды в кадре).
- Чем больше звезд на кадре, тем надежней будет идентификация.

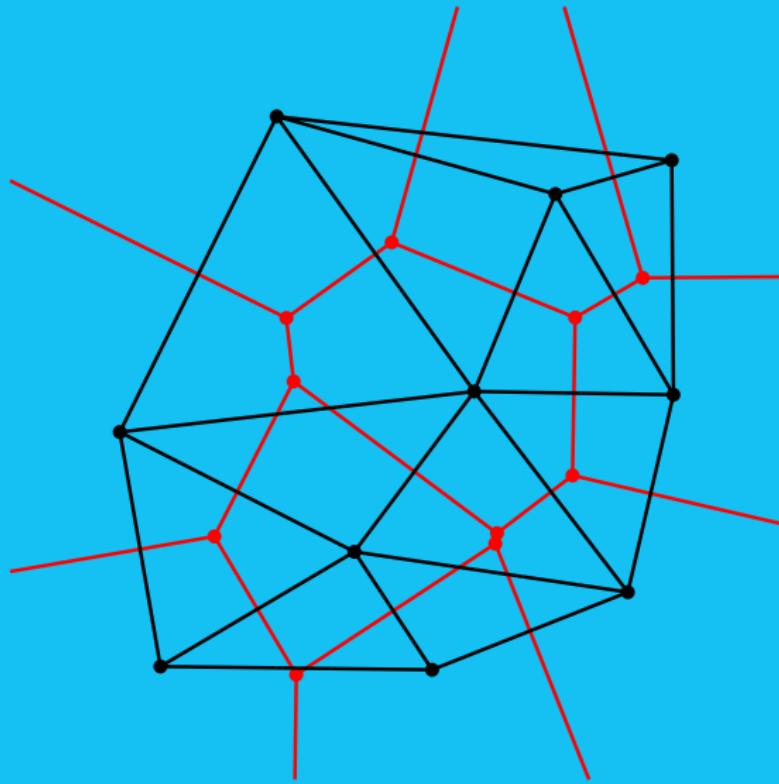
Триангуляция Делоне



Диаграммы Вороного



Диаграммы Вороного



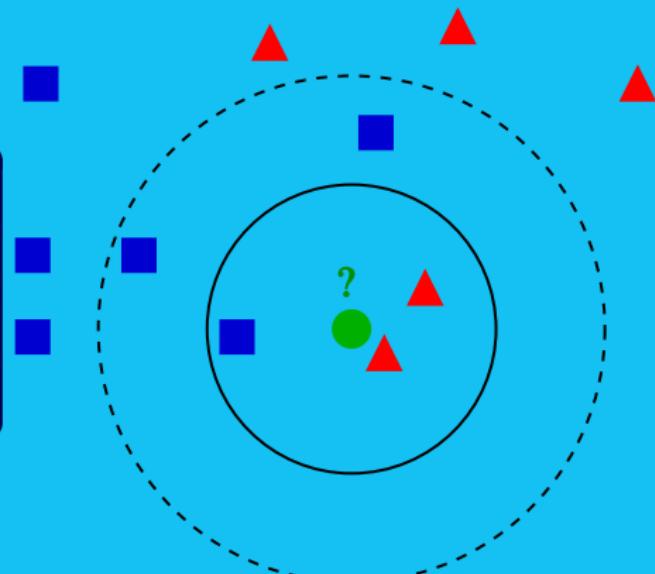
Свойства триангуляции Делоне

- ТД взаимно однозначно соответствует диаграмме Вороного для того же множества точек. Как следствие: если никакие четыре точки не лежат на одной окружности, ТД единственна.
- ТД максимизирует минимальный угол среди всех углов всех построенных треугольников, тем самым избегаются «тонкие» треугольники.
- ТД максимизирует сумму радиусов вписанных окружностей.
- ТД минимизирует максимальный радиус минимального объемлющего шара.
- ТД на плоскости обладает минимальной суммой радиусов окружностей, описанных около треугольников, среди всех возможных триангуляций.



K-nearest

Классификация объекта по k ближайшим соседям. В случае первой выборки — треугольник, в случае второй — квадрат.
 k может быть дробным, если применять взвешенные расстояния.



Программное обеспечение

<http://heasarc.gsfc.nasa.gov/docs/heasarc/astro-update/>

- ASTROPY: A single core package for Astronomy in Python
- Aladin: An interactive software sky atlas
- CFITSIO: FITS file access subroutine library
- GSL: GNU Scientific Library
- IDLAUL: IDL Astronomical Users Library
- IRAF: Image Reduction and Analysis Facility
- MIDAS: Munich Image Data Analysis System
- PyRAF: Run IRAF tasks in Python
- SAOImage ds9: FITS image viewer and analyzer
- SExtractor: Builds catalogue of objects from an astronomical image
- WCSLIB: World Coordinate System software library
- ... <http://tdc-www.harvard.edu/astro.software.html>

Литература

- W. Romanishin. An Introduction to Astronomical Photometry Using CCDs.
- Jean-Luc Starck and Fionn Murtagh. Handbook of Astronomical Data Analysis.
- Gonzalez & Woods. Digital Image Processing, 4th edition. 2018. ISBN 10: 1-292-22304-9
- Gonzalez & Woods & Eddins. Digital Image Processing Using MATLAB, 2nd edition. 2009.
- http://www.imageprocessingplace.com/root_files_V3/tutorials.htm



Спасибо за внимание!

mailto

eddy@sao.ru

edward.emelianoff@gmail.com

