

Лекция 3

Выделение контуров

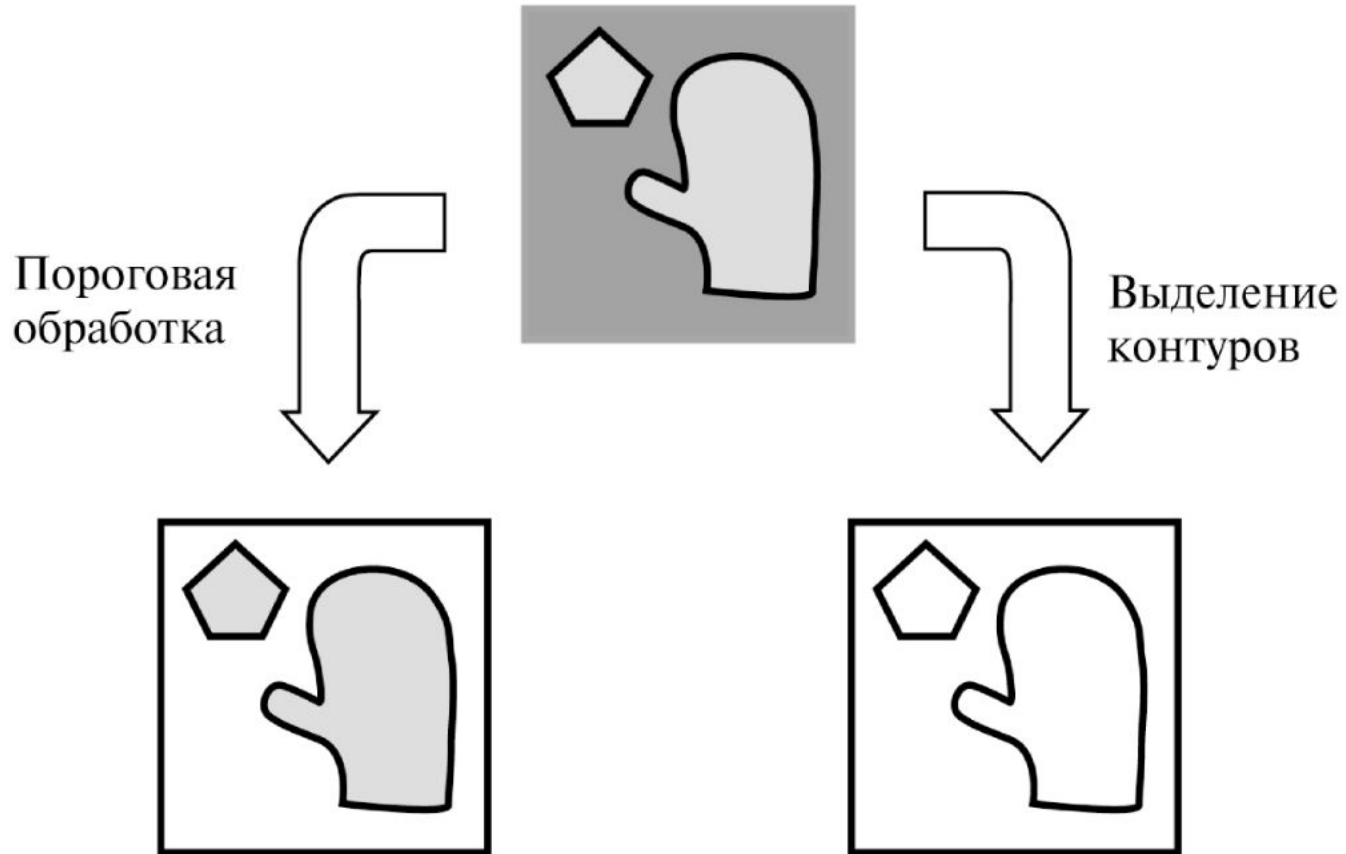
Контур vs. Граница

- Контур — в общем случае, замкнутая линия, очертание некоторой геометрической фигуры, предмета; силуэт.
- Граница – разделительная линия между чем-либо
- Контур замкнут всегда, граница не обязательно
- Замкнутая граница есть контур

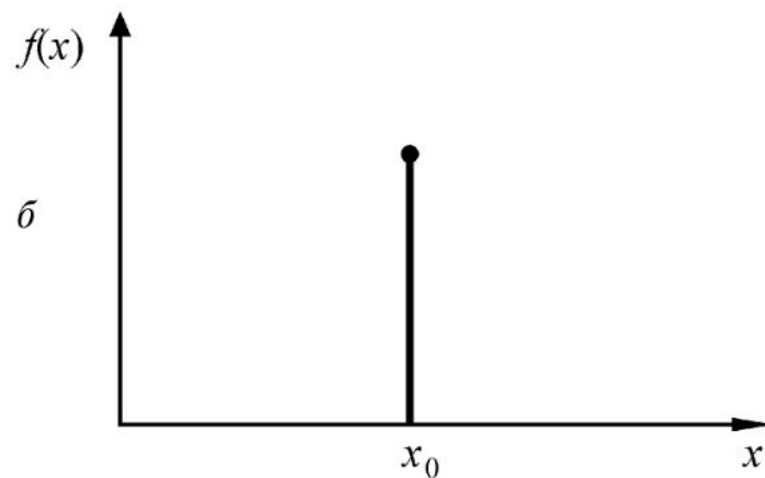
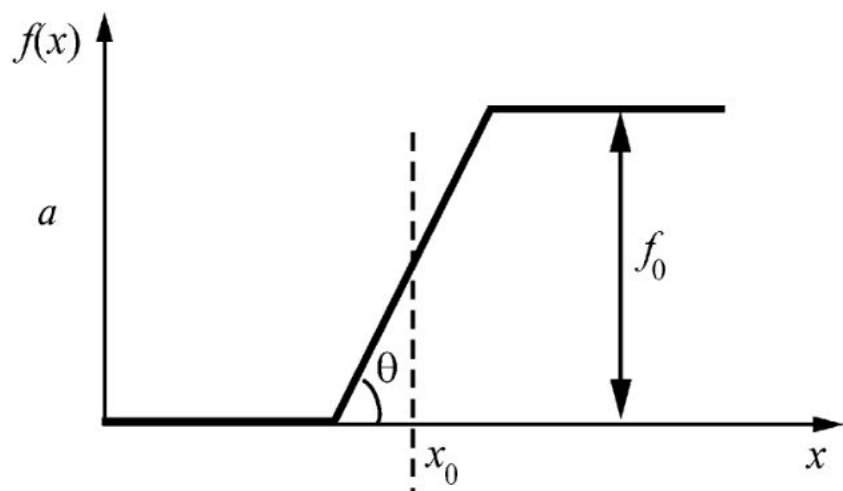
Назначение

- Предварительный этап для:
- Обнаружения объектов,
- Подсчета расстояний, поиска оптимального пути,
- Подсчета площади/объема
- Сегментации

Особенность

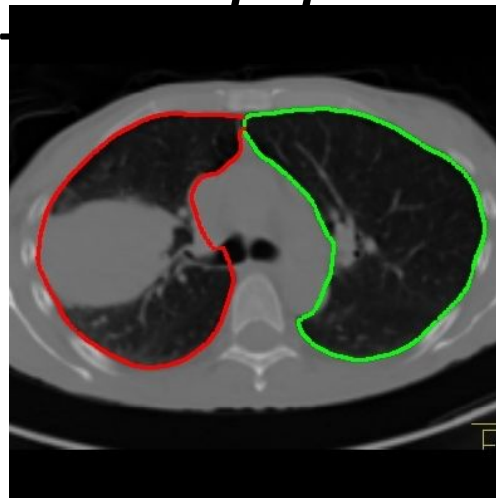


Детектор контура (идеальный)

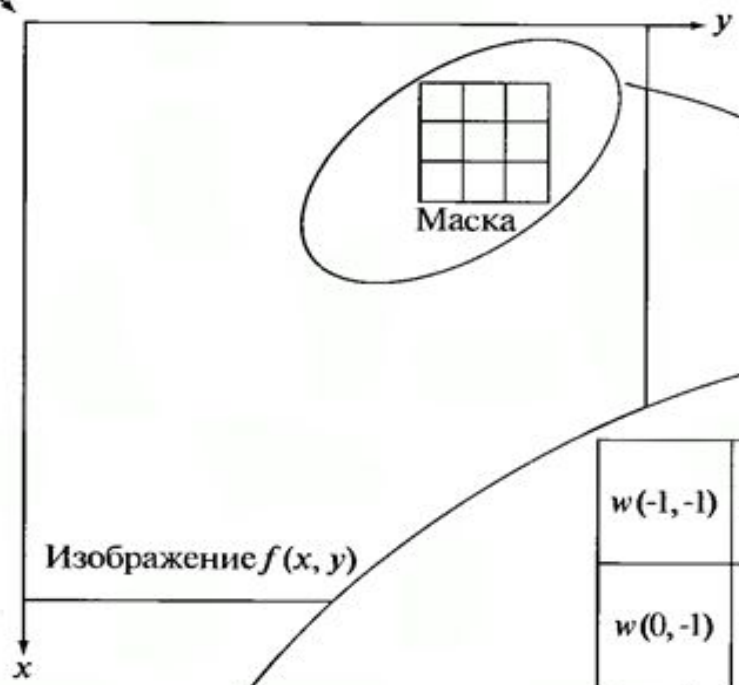


Классические «простые» методы выделения границ

- основываются на одном из базовых свойств сигнала яркости – *разрывности*.
- общим способом поиска разрывов является обработка с помощью *скользящей маски*.
- Элементы – *менты..*



Начало координат изображения



Маска

Изображение $f(x, y)$

$w(-1, -1)$	$w(-1, 0)$	$w(-1, 1)$
$w(0, -1)$	$w(0, 0)$	$w(0, 1)$
$w(1, -1)$	$w(1, 0)$	$w(1, 1)$

$f(x-1, y-1)$	$f(x-1, y)$	$f(x-1, y+1)$
$f(x, y-1)$	$f(x, y)$	$f(x, y+1)$
$f(x+1, y-1)$	$f(x+1, y)$	$f(x+1, y+1)$

Коэффициенты маски
с относительными
значениями
координат

Элементы области
изображения
под маской

Пространственная фильтрация

$$R = w(-1, -1)f(x - 1, y - 1) + w(-1, 0)f(x - 1, y) + \dots + w(0, 0)f(x, y) + \dots \\ + w(1, 0)f(x + 1, y) + w(1, 1)f(x + 1, y + 1)$$

$$\frac{\partial f}{\partial x} = f(x + 1) - f(x)$$

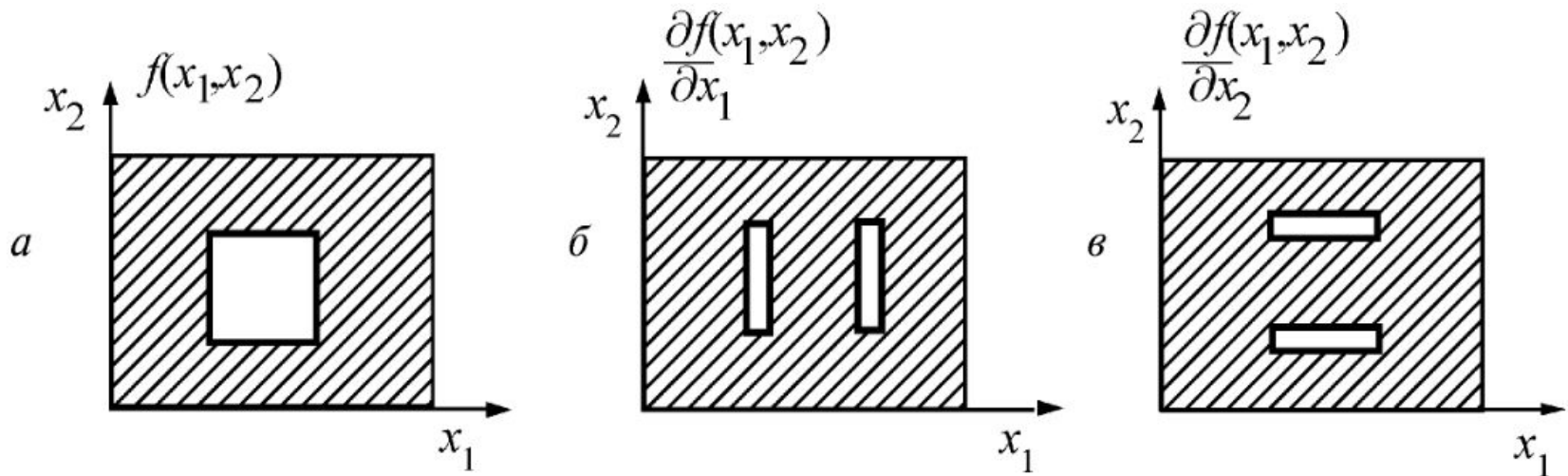
$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

$$\frac{\partial^2 f}{\partial x^2} = f(x + 1) + f(x - 1) - 2f(x)$$

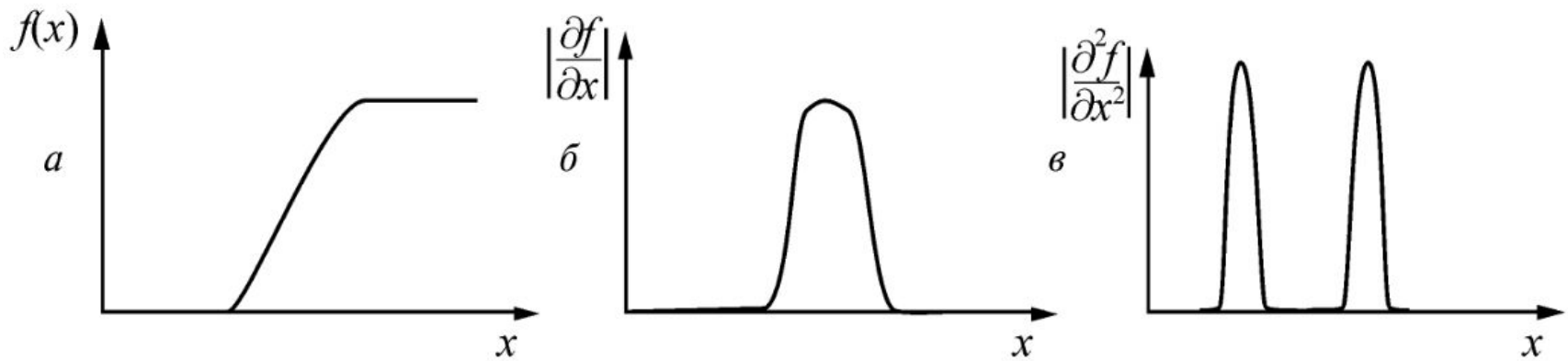
$$|\nabla f| = \sqrt{G_x^2 + G_y^2}$$

$$\alpha(x, y) = \operatorname{arctg} \left(\frac{G_y}{G_x} \right)$$

Пространственная фильтрация



f' vs. f''



Оператор Робертса

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Окрестность 3x3 внутри
изображения

$$G_x = (z_9 - z_5)$$

$$G_y = (z_8 - z_6)$$

Маски оператора
Робертса

-1	0
0	1

0	-1
1	0

Оператор Превитта

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

$$G_x = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)$$

$$G_y = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)$$

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

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

Оператор Собеля

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

$$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

$$G_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

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

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

$$\nabla f \approx |G_x| + |G_y|$$

$$f = \sqrt{G_x^2 + G_y^2}$$

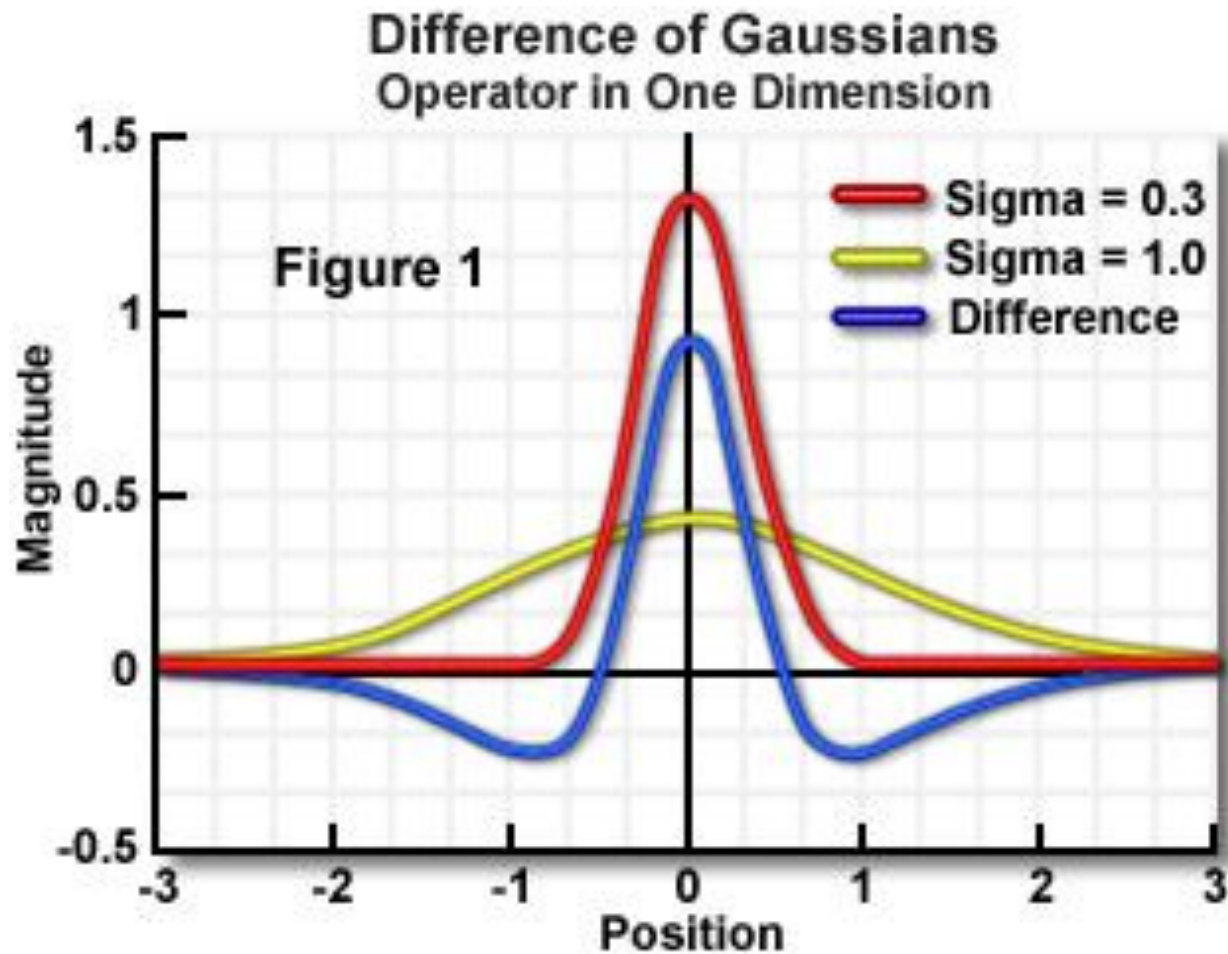
Примеры

- <http://matlabtricks.com/post-5/3x3-convolution-kernels-with-online-demo>

DoG - Diff. of Gaussian

- DoG обнаруживает края путем применения размытия изображения по Гауссу с указанным значением θ . Результирующее изображение является размытой версией исходного изображения.
- Далее применяется другое размытие с "более резким" значением θ , которое размывает изображение меньше чем в предыдущем случае.
- Финальное изображение вычисляется путем замены каждого пикселя разностью между двумя размытыми изображениями

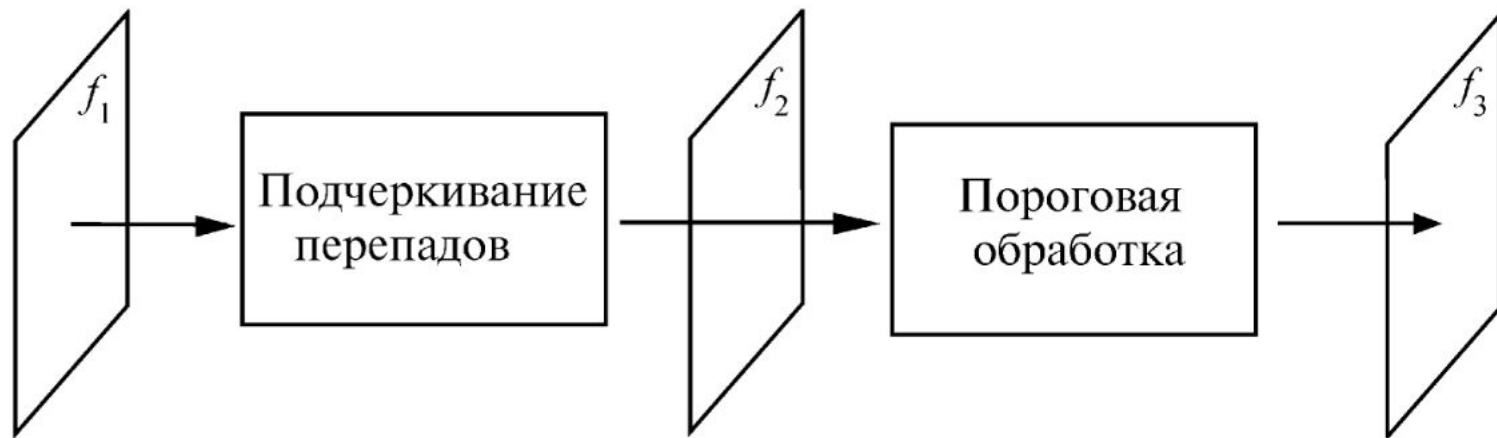
DoG - Diff. of Gaussian



DoG. An interactive tutorial

- <http://micro.magnet.fsu.edu/primer/java/digitalimaging/processing/diffgaussians/index.html>

Ожидание vs. Реальность



Алгоритм Сэнны

Целью Кэнни было разработать оптимальный алгоритм обнаружения границ, удовлетворяющий трём критериям:

- хорошее обнаружение (Кэнни трактовал это свойство как повышение отношения сигнал/шум);
- хорошая локализация (правильное определение положения границы);
- единственный отклик на одну границу.

Алгоритм Сапну. Этапы

Сглаживание

$$H_{ij} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(i - (k+1))^2 + (j - (k+1))^2}{2\sigma^2}\right); 1 \leq i, j \leq (2k+1)$$

$$\sigma = 1.4$$

$$\mathbf{B} = \frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} * \mathbf{A}.$$

Алгоритм Сэнны. Этапы

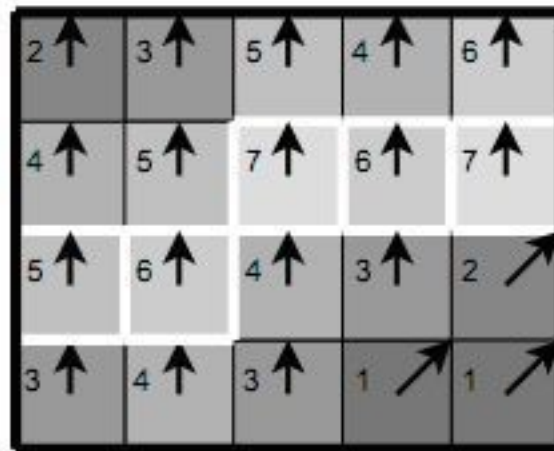
Поиск градиентов.

Границы отмечаются там, где градиент изображения приобретает максимальное значение. Они могут иметь различное направление, поэтому алгоритм Кэнни использует четыре фильтра для обнаружения горизонтальных, вертикальных и диагональных ребер в размытом изображении.

Алгоритм Canny. Этапы

Удаление не-максимумов

Пикселями границ объявляются пиксели, в которых достигается локальный максимум градиента в направлении вектора градиента. Значение направления должно быть кратно 45°



Алгоритм Canny. Этапы

Двойная пороговая фильтрация

- если значение пикселя выше верхней границы – он принимает максимальное значение (граница считается достоверной),
- если ниже – пиксель подавляется,
- точки со значением, попадающим в диапазон между порогов, принимают фиксированное среднее значение (они будут уточнены на следующем этапе).

Алгоритм Саппу. Этапы

Трассировка области неоднозначности
задача сводится к выделению групп пикселей, получивших на предыдущем этапе промежуточное значение, и отнесению их к границе (если они соединены с одной из установленных границ) или их подавлению (в противном случае).

Пиксель добавляется к группе, если он соприкасается с ней по одному из 8-ми направлений.

Алгоритм Sanny. Этапы

Пример с Хабра

<https://habrahabr.ru/post/114589/>

Алгоритм Canny. Online

- <http://homepages.inf.ed.ac.uk/rbf/HIPR2/cannydemo.htm>