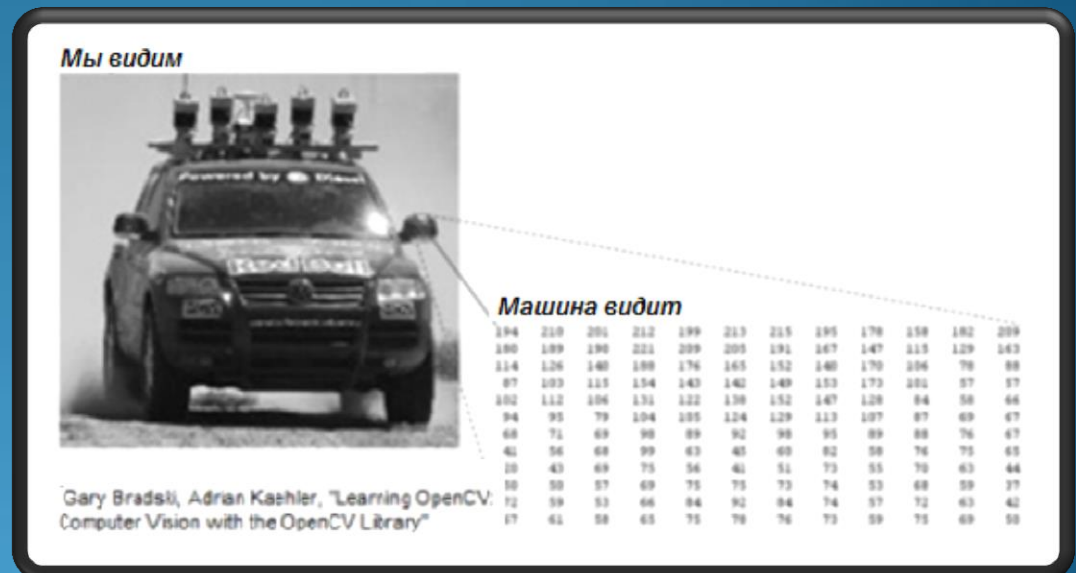


# ВВЕДЕНИЕ В КОМПЬЮТЕРНОЕ ЗРЕНИЕ



Бовырин Александр Владимирович,  
к.ф.-м.н., ст. преподаватель кафедры ЧиФА, ННГУ  
директор по разработке и исследованиям, ITSEEZ

# Введение

- Что такое зрение?

- 1) зрение — восприятие организмом объектов внешнего мира посредством улавливания отражаемого или излучаемого объектами света (*Большой Энциклопедический словарь*)
- 2) зрение — (visio, visus) восприятие света органом зрения и зрительным анализатором, благодаря чему организм получает информацию об объектах (*Большой медицинский словарь*)
- 3) зрение — восприятие организмом внешнего мира, т. е. получение информации о нём, посредством улавливания специальными зрением органами отражаемого или излучаемого объектами света (*Большая советская энциклопедия*)
- 4) зрение — Одно из внешних чувств человека и животного, органом которого является глаз; (*Толковый словарь Ожегова*)
- 5) «To see means to know what is where by looking, David Marr, Vision, 1982

- Что такое компьютерное зрение?

Компьютерное зрение — теория и технологии создания машин, которые способны извлекать полезную информацию из изображений.

# В чём сложность?

We perceive this:



picture from:  
Gary Bradski, Adrian Kaehler, "Learning OpenCV:  
Computer Vision with the OpenCV Library"

But the camera sees this:

194	210	201	212	199	213	215	196	176	198	182	209
180	189	190	221	209	205	191	167	147	115	129	163
114	126	140	188	176	165	152	140	170	106	76	88
87	103	115	154	143	142	149	153	173	101	87	87
102	112	106	171	122	138	152	147	129	84	56	66
94	93	79	104	105	124	129	113	107	87	69	67
68	71	69	98	89	92	96	95	89	80	76	67
41	56	68	89	63	43	80	82	58	76	73	63
20	43	69	75	56	41	51	73	55	70	63	44
30	50	37	69	70	73	73	74	53	68	59	37
72	59	33	66	84	92	84	74	57	72	63	42
47	61	58	65	75	78	76	73	59	75	69	58

## Неоднозначность восприятия



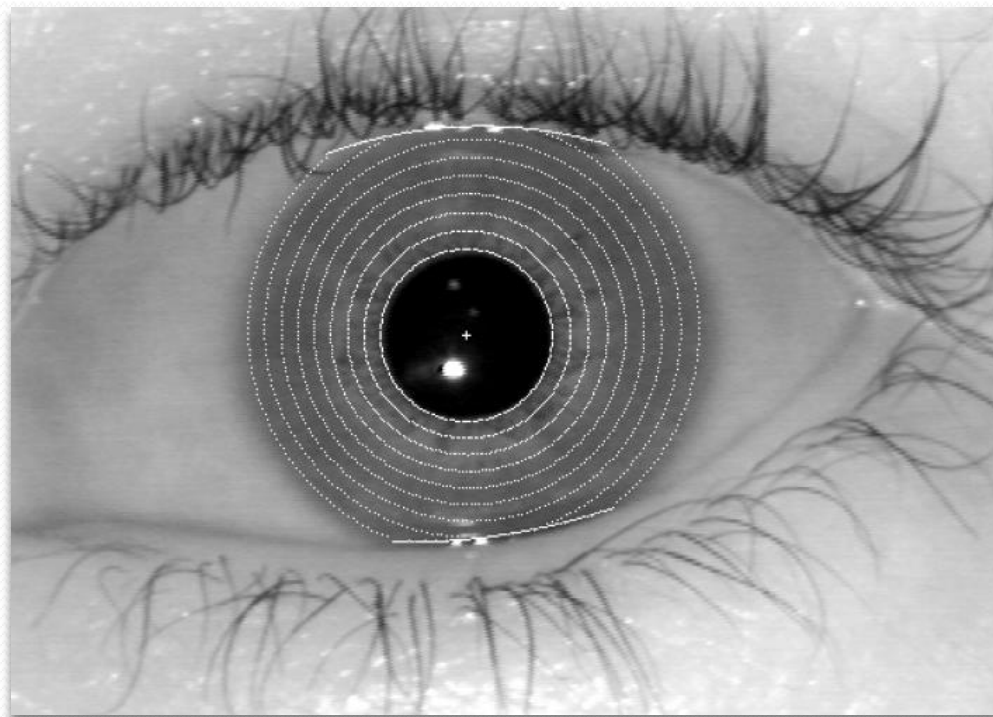
# Видеонаблюдение.

С  
Л  
О  
Ж  
Н  
О  
С  
Т  
Ь

- Детектирование движения и запись видео. (видеорегистраторы)
- Детектирование только определённых объектов. (метро)
- Охрана определённых зон. Охрана предметов. (музеи)
- Детектирование оставленных предметов/машин. (вокзалы)
- Подсчёт движущихся объектов на сцене. (магазины)
- Сопровождение объектов. Определение их скорости.
- Анализ траекторий движения. (рекомендации маркетингу).
- Сопровождение объекта используя несколько камер.
- Детектирование поскользнувшихся и упавших людей.
- Обнаружение воровства на кассе.  
(POS monitor)

# Биометрия

- Распознавание отпечатков пальцев
- Распознавание лица (2D, 3D)
- Идентификация по изображению роговицы



# Медицинские приложения

- Поиск аномалий на медицинских изображениях
- Помощь в проведении анализов
- Классификация рентгеновских снимков
- Детектирование эпи-припадков
- Автоматическая диагностика болезней глаз



# Улучшение качества изображения



- Удаление шумов.
- Улучшение цветопередачи.
- Улучшение контраста.
- Просветление тёмных фрагментов.
- Удаление нежелательных объектов.
- Удаление “красных глаз”



Google



Scre...47 PM.png x

описание изображения



Поиск

Результатов: примерно 23 (0,50 сек.)

Все результаты

Картинки

Карты

Видео

Новости

Ещё

Поиск по картинке

Похожие

Другие размеры

За всё время

За час

За 24 часа

За неделю

За месяц

За год

За период...



Размер изображения:  
405 × 275

Изображения других размеров не найдены.

Страницы с подходящими изображениями



100 × 100

[ФотоПремия - Лучшие фотографии мира!](#)

[fotopremiya.ru/index](#)

Форум, Статистика, Последнее сообщение. О проекте. В этом форуме можно обсуждать любые темы касающиеся развития проекта FotoPremiya.ru ...



120 × 150

[Pilgrim - Просмотр](#)

[fotopremiya.ru/user/5-pilgrim](#)

Pilgrim. Регистрация:

-. Найдите темы Публикации

77 ...



366 × 366

[Творческая мастерская](#)

[mayart.ru/gallery/](#)

image audio video quote

Галереи · ФотоАльбомы

Image · Link ...



[Видео уроки фотографии](#)

# Поиск в архиве

## TinEye Services

Image recognition for your applications and enterprise.

[Documentation](#)

[Support](#)

[Contact Us](#)

[Login](#)

## MobileEngine

*Mobile image recognition and identification*

Build mobile application with fully integrated image recognition.

Connect users with your product information via a camera phone.

[Details](#)

[Plans & Pricing](#)



## MulticolorEngine



Let your users search through your image collection using color.

## MatchEngine



Finds duplicate, modified and even derivative images in your image collection.

## MobileEngine



Build mobile applications that use high-sensitivity image recognition.

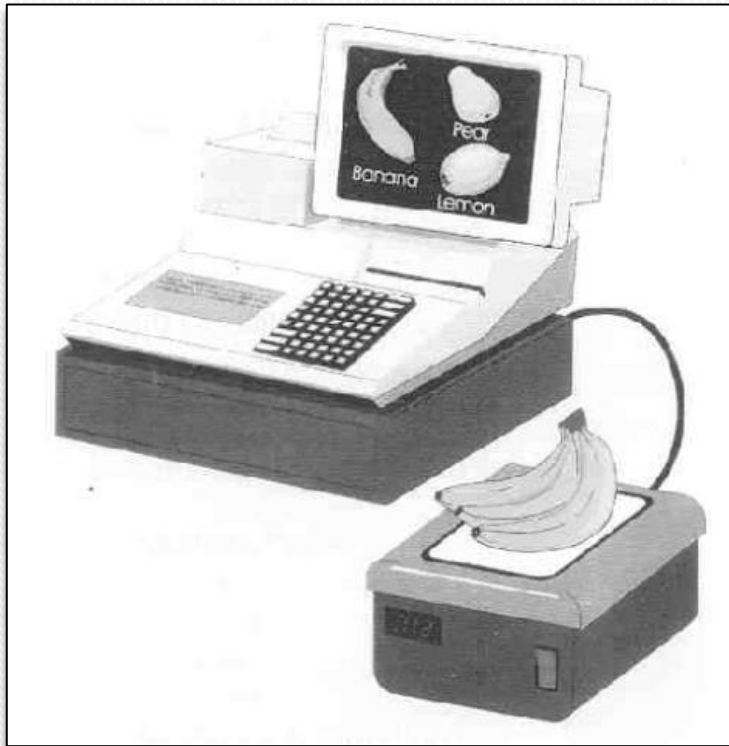
## TinEye API



Integrate TinEye with your own systems to automatically search the TinEye index.



# Распознавание объектов по образцу



Machine vision.  
Visual inspection.

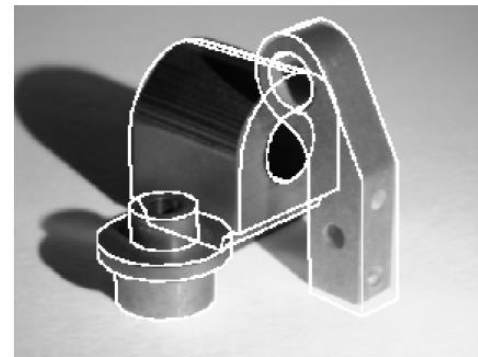


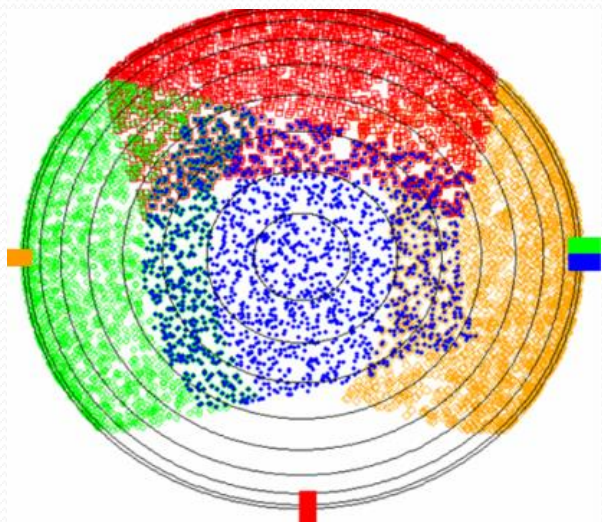
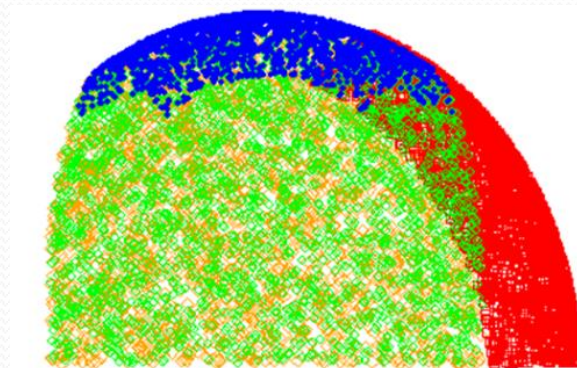
Figure 1.7: An inspection or assembly robot matches stored 3D models to a sensed 2D image (Courtesy of Mauro Costa).

# Помощь водителю

- Определение смены полосы движения.
- Определение расстояния до препятствия.
- Нахождение светофора и предупреждение о красном цвете.
- Локализация и распознавание знаков.
- Детектирование пешеходов.
- Детектирование закрытых глаз водителя.

# Взаимная калибровка камер и проекторов.

- Цифровой планетарий



Центр компьютерного зрения «Аргус»

Argus Planetarium



# Новый интерфейс. 3D зрение



# Часть II

- Формирование изображений
- Обработка бинарных изображений
- Обработка полутоновых изображений
- Цветовые пространства
- Частотный анализ изображений

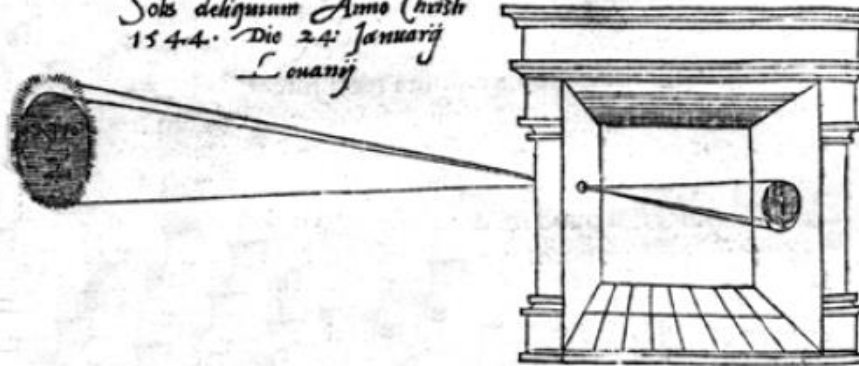


# Формирование изображения

## Camera Obscura

illum in tabula per radios Solis, quàm in cœlo contingit: hoc est, si in cœlo superior pars deliquiū patiatur, in radiis apparebit inferior deficere, vt ratio exigit optica.

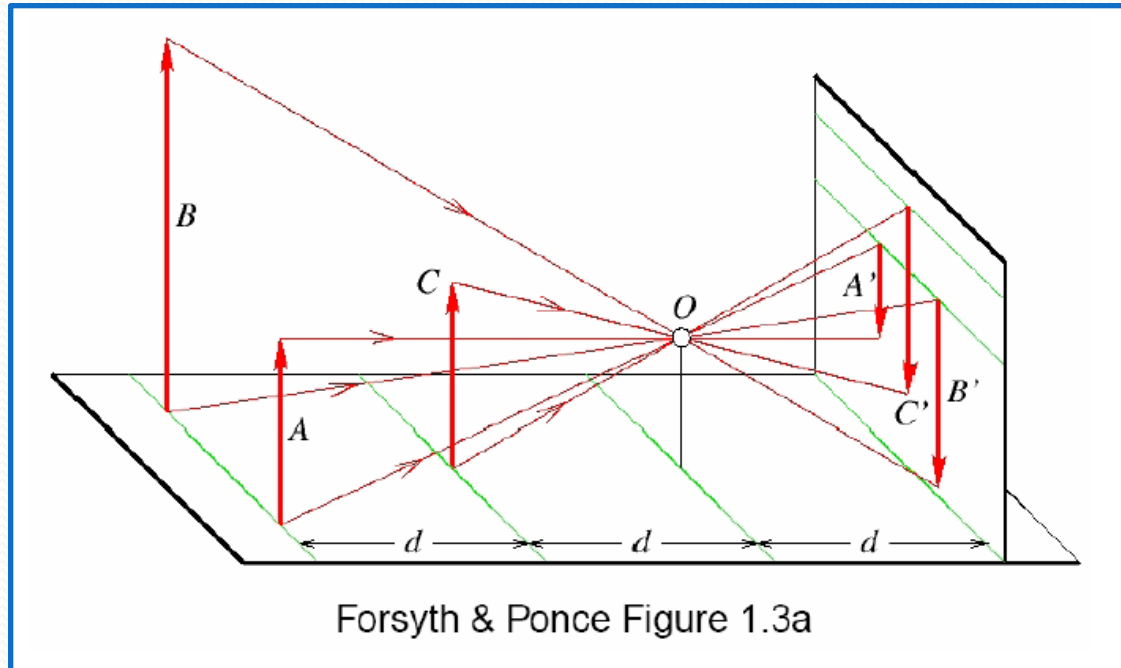
*Solis deliquium Anno Christi  
1544. Die 24. Januarij  
Louanij*



Sic nos exactè Anno .1544. Louanii eclipsim Solis obseruauimus, inuenimusq; deficere paulò plus q̃ dex-

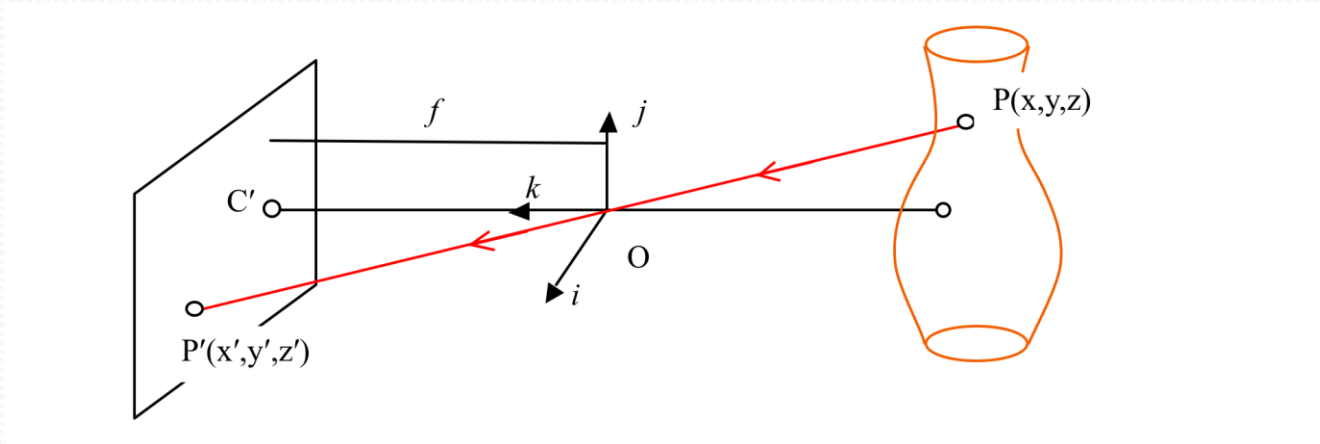
"When images of illuminated objects ... penetrate through a small hole into a very dark room ... you will see [on the opposite wall] these objects in their proper form and color, reduced in size ... in a reversed position, owing to the intersection of the rays". *Da Vinci*

# Формирование изображения – перспективная проекция



- Размер проекции объекта обратно пропорционален расстоянию до него.
- Точка проектируется в точку.
- Линия проектируется в линию.
- Углы не сохраняются.
- Параллельные линии могут пересекаться.

# Формирование изображения – перспективная проекция



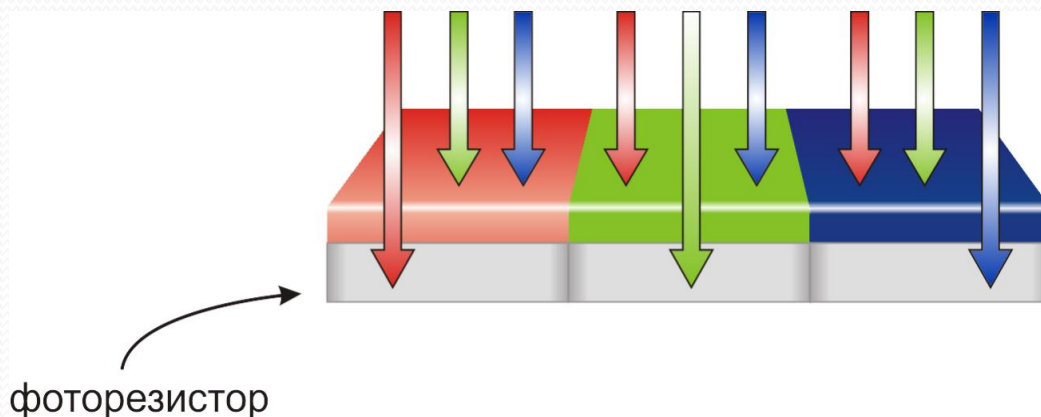
$$x' = f' \frac{x}{z}$$

$$y' = f' \frac{y}{z}$$

# Точка схода

- Линия проходящая через точку  $(X_0, Y_0, Z_0)$  в направлении  $(U, V, W)$  описывается уравнением  $(X_0 + kU, Y_0 + kV, Z_0 + kW)$
- Проекция этой линии записывается как  $P_k = (f(X_0 + kU)/(Z_0 + kW), f(Y_0 + kV)/(Z_0 + kW))$
- При  $k \rightarrow \infty$  получаем “точку схода”  
 $P_{\infty} = (fU/W, fV/W)$
- Видно, что точка схода не зависит от  $(X_0, Y_0, Z_0)$

# Формирование изображений



Формирование цифрового изображения в камере происходит следующим образом. Свет, отражённый от объектов сцены, проходит через линзу (или систему линз) объектива камеры и фокусируется на сенсоре, который состоит из фотоэлементов, покрытых светофильтрами. Таким образом, отдельные ячейки сенсора формируют разные цветовые каналы. Чаще всего используются RGB (red, green, blue) светофильтры.



# Цифровые изображения

Цифровое изображение – это двумерный массив  $I[r,c]$  элементов (пикселей), которые представляют собой одно или нескольких дискретных значений.

- *Определение. Бинарное изображение – это цифровое изображение  $B[r,c]$ , пиксели которого принимают значения 0 или 1.*
- *Определение. Полутонное (серое, монохромное, Ч/Б) изображение – это цифровое изображение  $I[r,c]$ , у которого каждому пикселю соответствует одно значение интенсивности (яркости).*
- *Определение. Мультиспектральное изображение – это цифровое изображение  $M[r,c]$ , у которого каждому пикселю соответствует вектор значений. (у цветных изображений размерность вектора = 3).*

# Бинарные изображения. Анализ и морфологические операции.

- Бинарные изображения.



- 4-ёх и 8-ми связанность

	1	
2	*	3
	4	

1	2	3
4	*	5
6	7	8

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

Сколько 4ёх связанных и  
8ми связанных компонент  
изображено ?

# Морфологические операции

- Эрозия:  $\text{dst}(x,y) = \min_{((x',y') \in [-3..3])} \text{src}(x+x',y+y')$

а а

- Наращивание:  $\text{dst}(x,y) = \max_{((x',y') \in [-3..3])} \text{src}(x+x',y+y')$

а а

- Размыкание:  $\text{dst} = \text{open}(\text{src}) = \text{dilate}(\text{erode}(\text{src}))$

а а а

- Замыкание:  $\text{dst} = \text{close}(\text{src}) = \text{erode}(\text{dilate}(\text{src}))$

а а а

# Анализ связанных компонент.

Свойства областей.

- Площадь

$$A = \sum_{(r,c) \in R} 1$$

- Центр тяжести

$$\bar{r} = \frac{1}{A} \sum_{(r,c) \in R} r$$

- Периметр

$$\bar{c} = \frac{1}{A} \sum_{(r,c) \in R} c$$

- Округлость

$$C_1 = \frac{|P|^2}{A}$$

- Описывающий прямоугольник (X,Y,W,H)

- Моменты(центральные и смешанные)

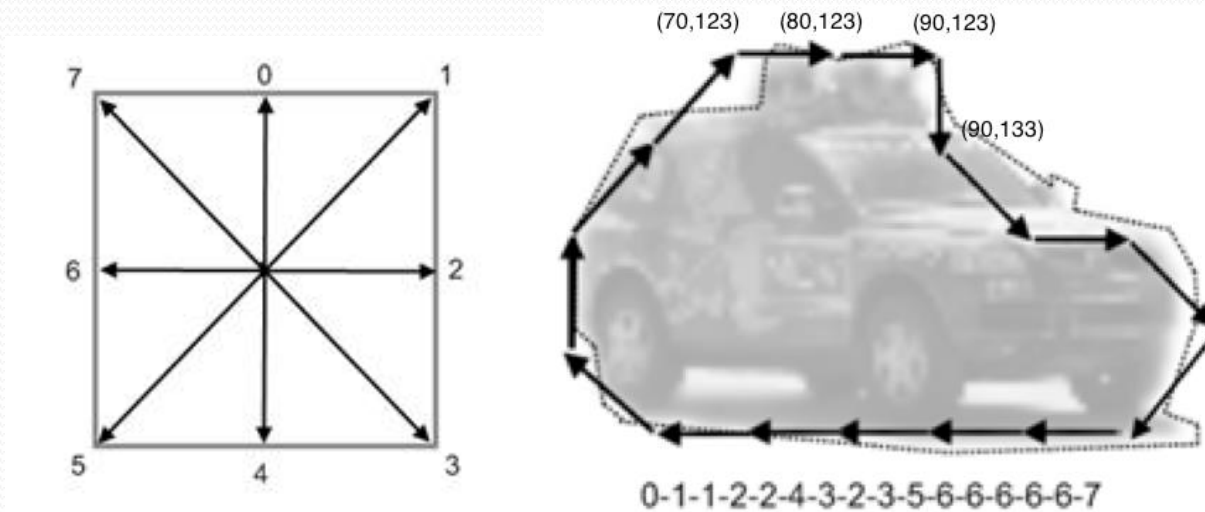
$$\mu_{rr} = \frac{1}{A} \sum_{(r,c) \in R} (r - \bar{r})^2 \quad \mu_{cc} = \frac{1}{A} \sum_{(r,c) \in R} (c - \bar{c})^2$$

- 

$$m_{p,q} = \sum_{i=1}^n I(x,y) x^p y^q$$

# Векторизация бинарных изображений.

- Векторизация – получение векторных данных из растра.
- Переход к векторному представлению позволяет значительно ускорить вычисления, а так же упростить некоторые алгоритмы.
- Пример 1 векторизации – представление границы связанной компоненты массивом двумерных точек  $\{X_i, Y_i\}$
- Пример 2 векторизации – код Фримана.





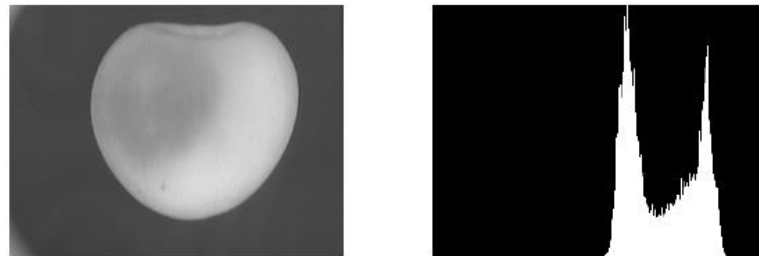
# Получение бинарных изображений из полутоновых.

- Пороговая бинаризация

$$g(x, y) = \begin{cases} 1 & \text{if } f(x, y) \geq T \\ 0 & \text{otherwise} \end{cases}$$

- Анализ гистограммы

$$h(m) = |\{(r, c) \mid I(r, c) = m\}|$$



- Метод Отсу (следующий слайд)

# Получение бинарных изображений из полутоновых. Метод Отсу.

- Метод основан на минимизации внутригрупповой дисперсии.  
Пусть  $P(0), \dots, P(I)$  наблюдаемые вероятности значений яркости

$$\sigma_{\text{Within}}^2(T) = n_B(T)\sigma_B^2(T) + n_O(T)\sigma_O^2(T) \xrightarrow[T]{\text{minimum}}$$

$$n_B(T) = \sum_{i=0}^{T-1} p(i)$$

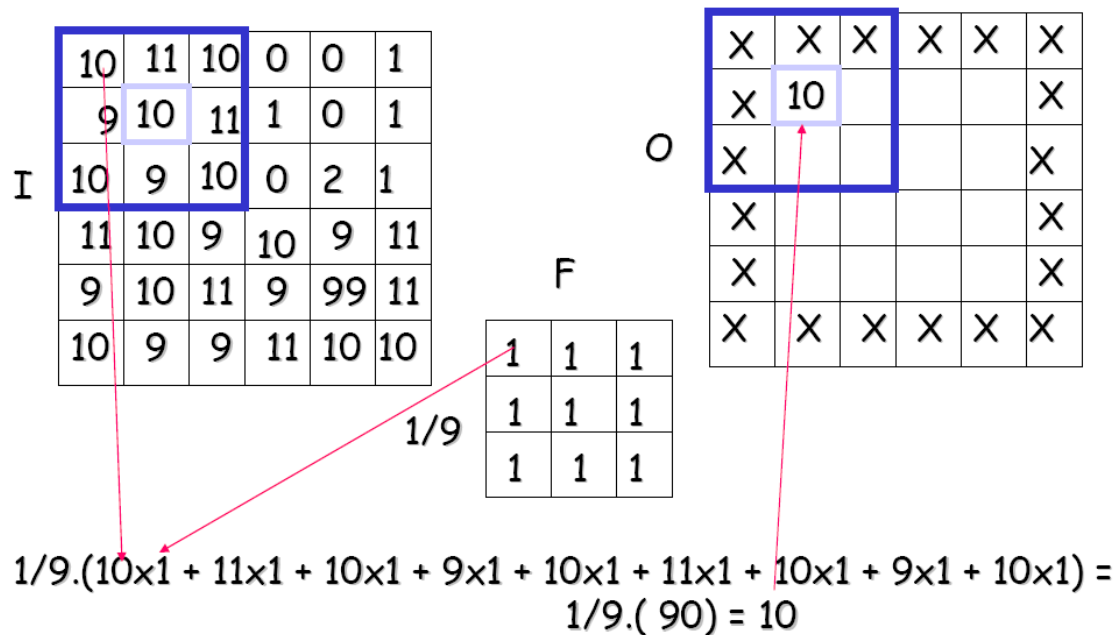
$$n_O(T) = \sum_{i=T}^{N-1} p(i)$$

$$\sigma_B^2(T) = \text{the variance of the pixels in the background (below threshold)}$$

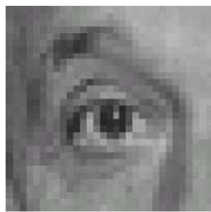
$$\sigma_O^2(T) = \text{the variance of the pixels in the foreground (above threshold)}$$

# Линейная фильтрация

$$I'(X, Y) = \sum_{j=-k}^K \sum_{i=-k}^K F(i, j) I(X - i, Y - j)$$



# Линейная фильтрация

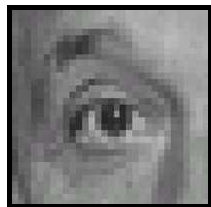


Original

0	0	0
0	1	0
0	0	0

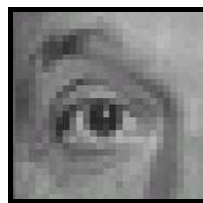


Filtered  
(no change)

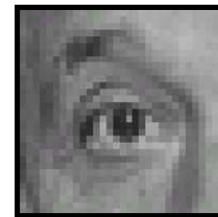


Original

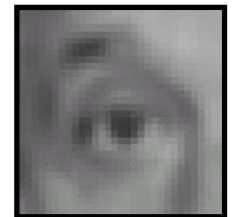
0	0	0
0	0	1
0	0	0



Shifted left  
By 1 pixel



Original

$$\frac{1}{9} \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$


Blur (with a  
box filter)



Original

$$\begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 0 & 2 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} - \frac{1}{9} \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$


## Sharpening filter

- Accentuates differences with local average
- Also known as Laplacian

# Подсчёт градиентов

$$\frac{\partial f}{\partial x} = \lim_{\varepsilon \rightarrow 0} \left( \frac{f(x + \varepsilon, y) - f(x, y)}{\varepsilon} \right) \rightarrow \frac{\partial f}{\partial x} \approx \frac{f(x_{n+1}, y) - f(x_n, y)}{\Delta x} \rightarrow$$

-1	1
----	---

Фильтры Собеля позволяют получить более устойчивые к шуму градиенты

 $\frac{1}{8}$ 

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

 $\frac{1}{8}$ 

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

$[I_x, I_y]$



# Лапласиан

$$\text{dst}(x,y) = d^2\text{src}/dx^2 + d^2\text{src}/dy^2$$

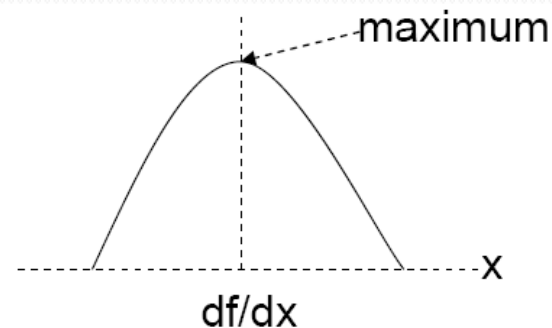
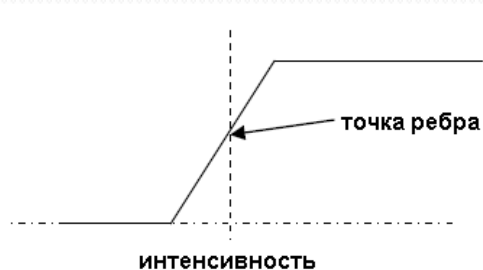
$$\begin{vmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{vmatrix}$$



picture from:  
Gary Bradski, Adrian Kaehler, "Learning OpenCV:  
Computer Vision with the OpenCV Library"

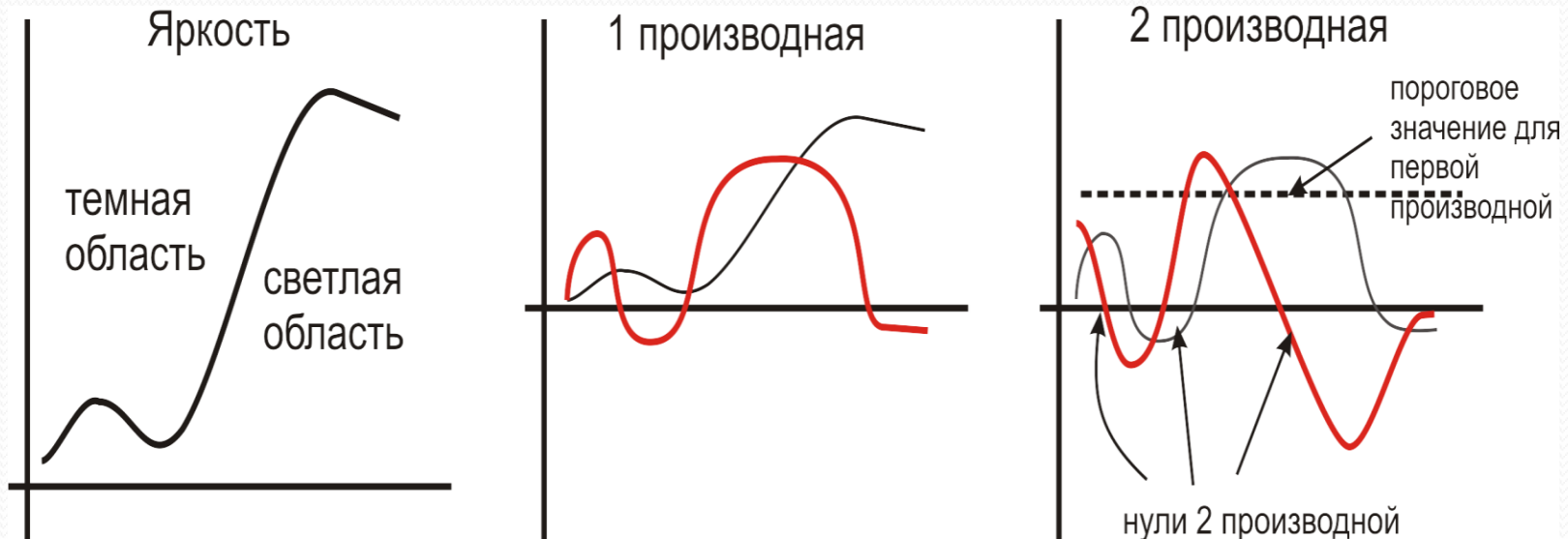
# Выделение краёв/рёбер объекта

Цель: найти резкие не шумовые перепады интенсивности, соответствующие границам объектов на сцене.

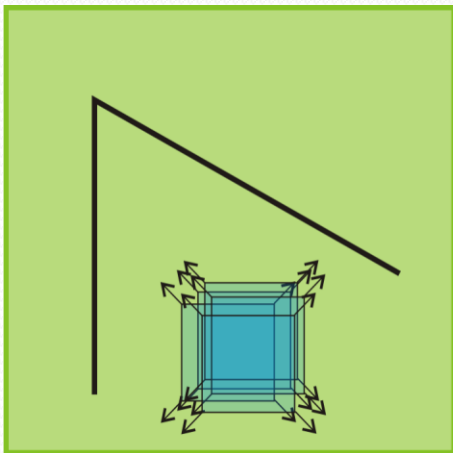


# Выделение краёв с помощью Лапласиана

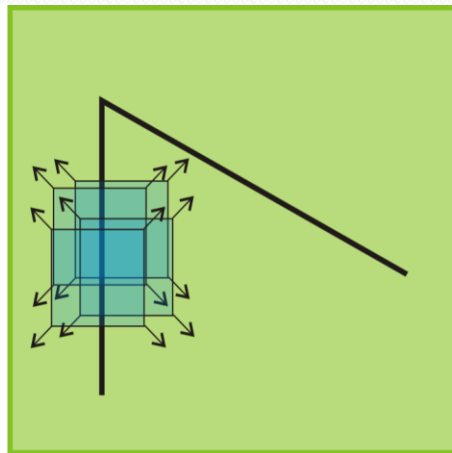
- Зная градиенты функции интенсивности и лапласиан, мы можем найти границы объектов, определяя их как точки с высокой нормой градиента (интенсивность резко изменяется на границе объекта) и нулевой второй производной (лапласиан = 0).



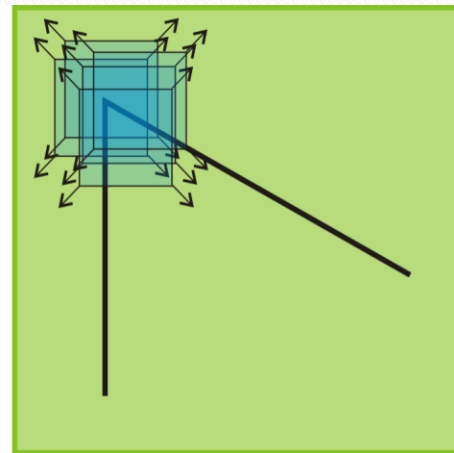
# Нахождение углов на изображении



Однородная  
область –  
нет  
изменений в  
любом  
направлении



Ребро –  
нет  
изменений в  
направлении  
ребра



Угловая  
точка –  
есть  
изменения в  
любом  
направлении

# Нахождение угловых точек (Harris)

$$S(x, y) = \sum_u \sum_v (I(u, v) - I(u + x, v + y))^2$$

$$I(u + x, v + y) \approx I(u, v) + I_x(u, v)x + I_y(u, v)y$$

$$S(x, y) \approx \sum_u \sum_v (I_x(u, v)x + I_y(u, v)y)^2 \quad S(x, y) \approx \begin{pmatrix} x & y \end{pmatrix} A \begin{pmatrix} x \\ y \end{pmatrix}$$

$$A = \sum_u \sum_v \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = \begin{bmatrix} \langle I_x^2 \rangle & \langle I_x I_y \rangle \\ \langle I_x I_y \rangle & \langle I_y^2 \rangle \end{bmatrix}$$

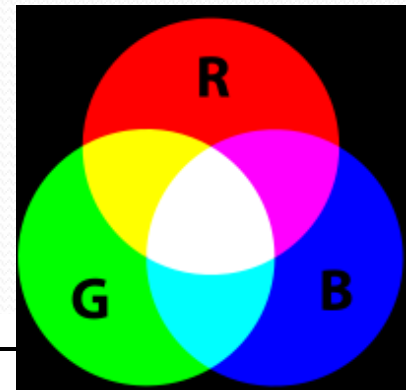
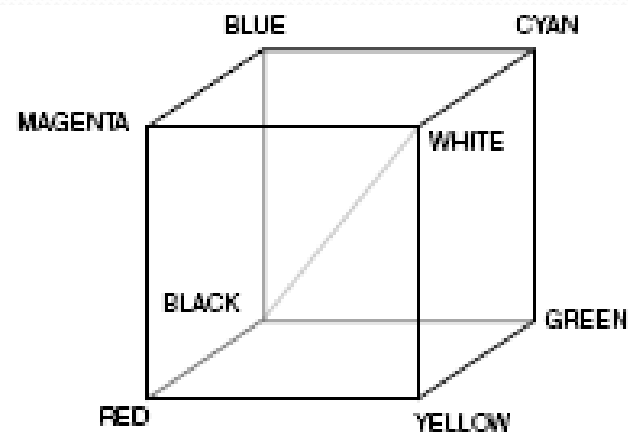
У матрицы  $A$  вычисляются собственные числа  $\lambda_1, \lambda_2$

1. Если  $\lambda_1 \sim 0$  &  $\lambda_2 \sim 0$  – однородная область.
2. Если  $\lambda_1 \sim 0$  &  $\lambda_2$  велико, то точка на ребре.
3. Если  $\lambda_1$  велико &  $\lambda_2$  велико, то точка является угловой.

# Цветные изображения.

## Пространство RGB.

- Современные устройства формирования изображений обычно выдают изображение в формате RGB.
  - RGB является доминантным цветовым пространством в графике.
- Однако RGB не всегда удобно для построения систем компьютерного зрения.



	Nominal Range	White	Yellow	Cyan	Green	Magenta	Red	Blue	Black
R	0 to 255	255	255	0	0	255	255	0	0
G	0 to 255	255	255	255	255	0	0	0	0
B	0 to 255	255	0	255	0	255	0	255	0

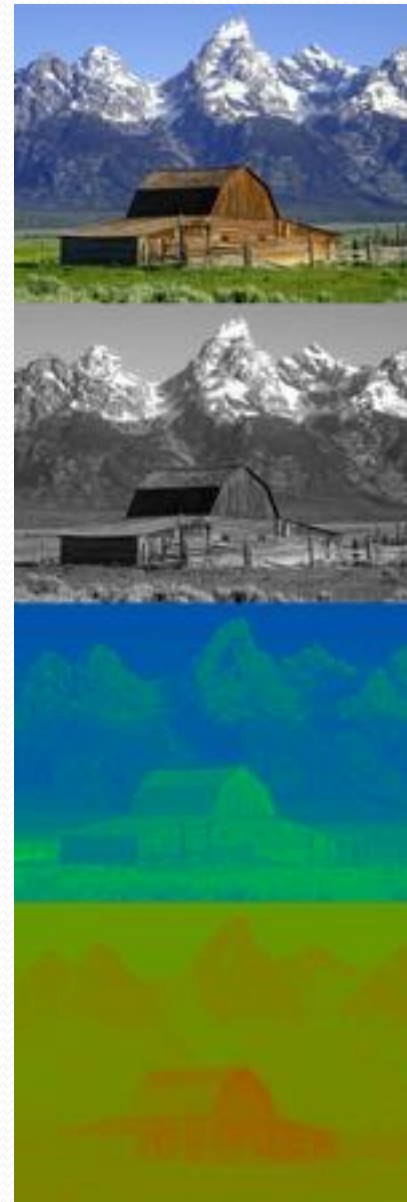


# Пространство YUV

Y – яркость,  
U,V – компоненты отвечающие за цвет

В телевидении и при сжатии часто применяют  
схемы 4:2:2 или 4:1:1

$$\begin{aligned} Y &= 0.299 \times R + 0.587 \times G + 0.114 \times B \\ U &= -0.147 \times R - 0.289 \times G + 0.436 \times B \\ V &= 0.615 \times R - 0.515 \times G - 0.100 \times B \end{aligned}$$



# Нелинейные цветовые пр-ва (HSV)

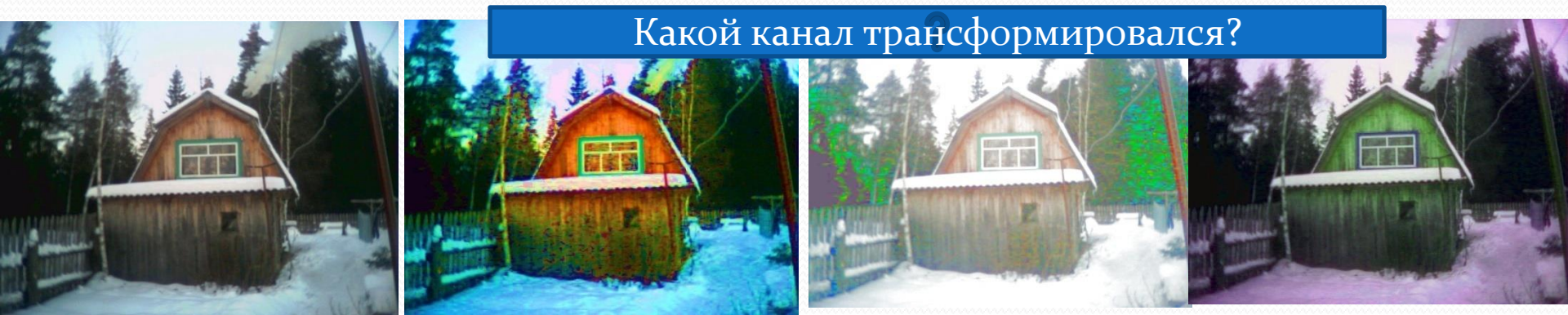
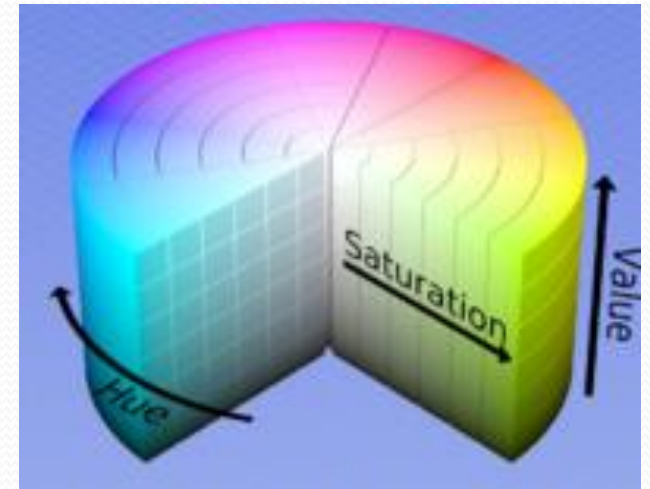
Более понятное описание цвета –  
интенсивность (яркость)+ оттенок (Hue/тон)+ насыщенность (saturation).

```
V <- max(R,G,B)
S <- (V-min(R,G,B))/V    if V≠0, 0 otherwise

      (G - B)*60/S,    if V=R
H <- 180+(B - R)*60/S,  if V=G
      240+(R - G)*60/S,  if V=B

if H<0 then H<-H+360

On output  $0 \leq V \leq 1$ ,  $0 \leq S \leq 1$ ,  $0 \leq H \leq 360$ .
```



# Частотный анализ изображений

- Преобразование Фурье. Разложение изображения по гармоникам.

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j 2\pi (ux + vy)} dx dy$$

- Дискретный вариант

$$F[u, v] \equiv \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} I[x, y] e^{\frac{-2\pi j}{N} (xu + yv)}$$

$$I[x, y] \equiv \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F[u, v] e^{\frac{+2\pi j}{N} (ux + vy)}$$

- $F(0,0)$ ?

# Спектр и фаза

- Спектр

$$e^{-j 2\pi(ux+vy)} = \cos(2\pi(ux + vy)) - j\sin(2\pi(ux + vy))$$

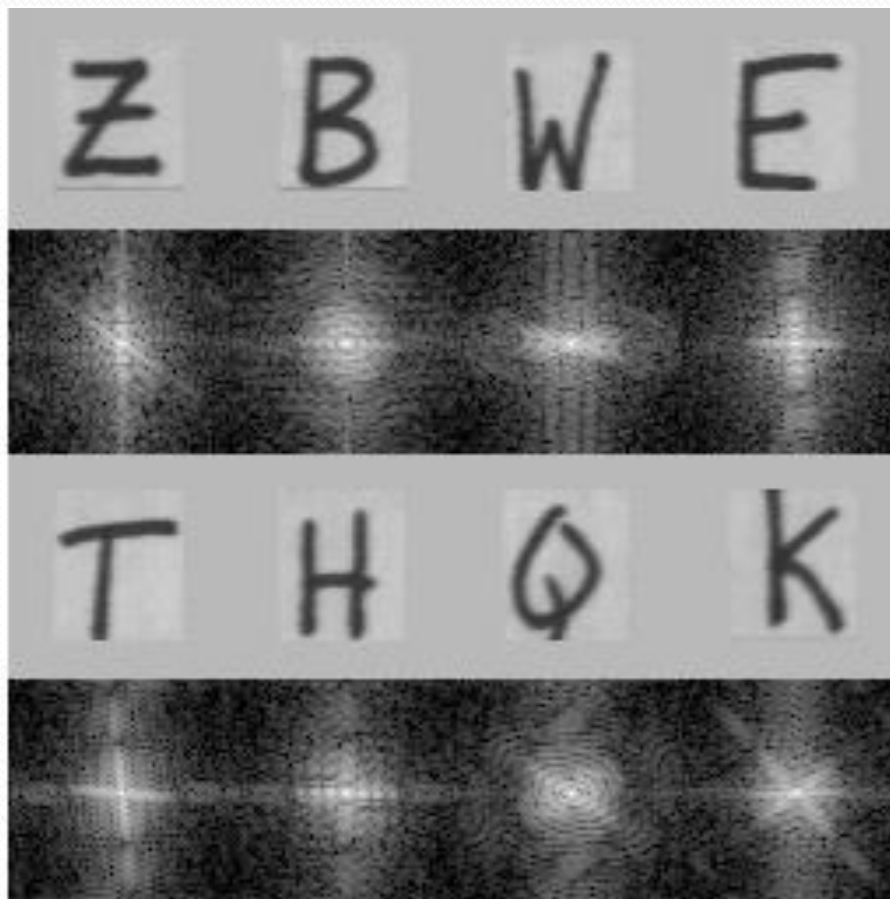
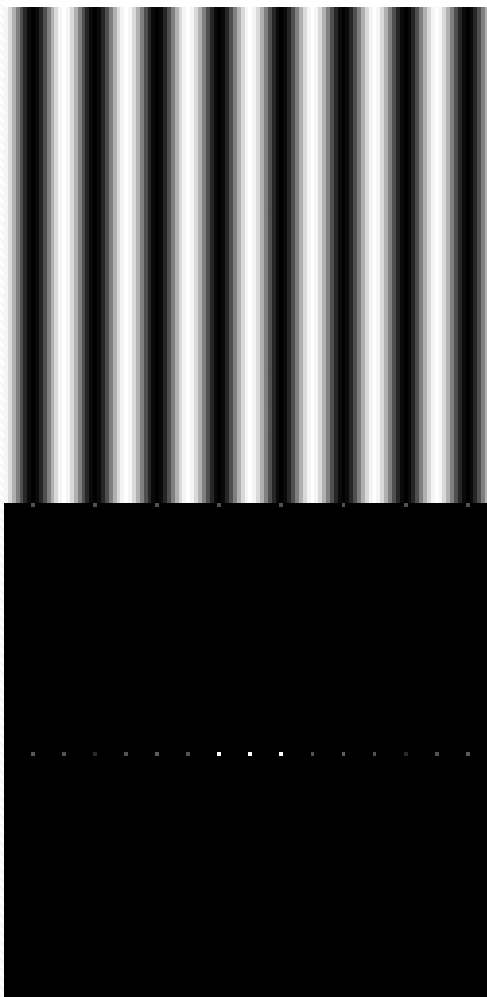
$$P(u, v) \equiv ( \text{Real}(F(u, v))^2 + \text{Imaginary}(F(u, v))^2 )^{1/2}$$

$$P(-u, -v) = P(u, v)$$

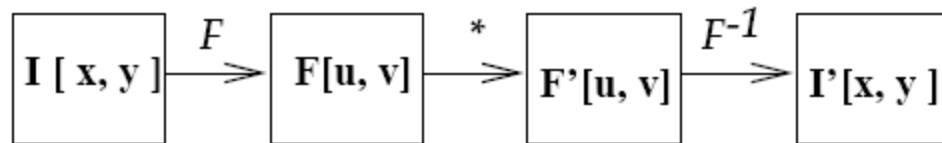
- Фаза

$$\text{PHASE}(F) = \text{ATAN}( \text{IMAGINARY}(F)/\text{REAL}(F) )$$

# Примеры спектра



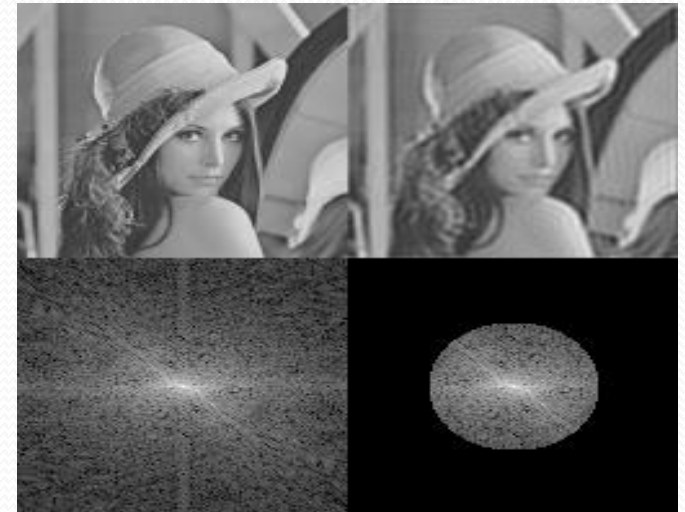
# Полосно-пропускающий фильтр



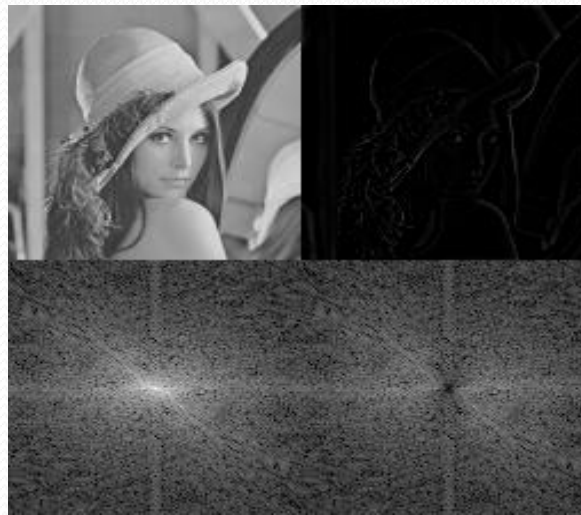
- Подавление высоких частот

$$G(k, l) = F(k, l)H(k, l)$$

$$H(k, l) = \begin{cases} 1 & \text{if } \sqrt{k^2 + l^2} < D_0 \\ 0 & \text{if } \sqrt{k^2 + l^2} > D_0 \end{cases}$$



- Подавление низких частот:





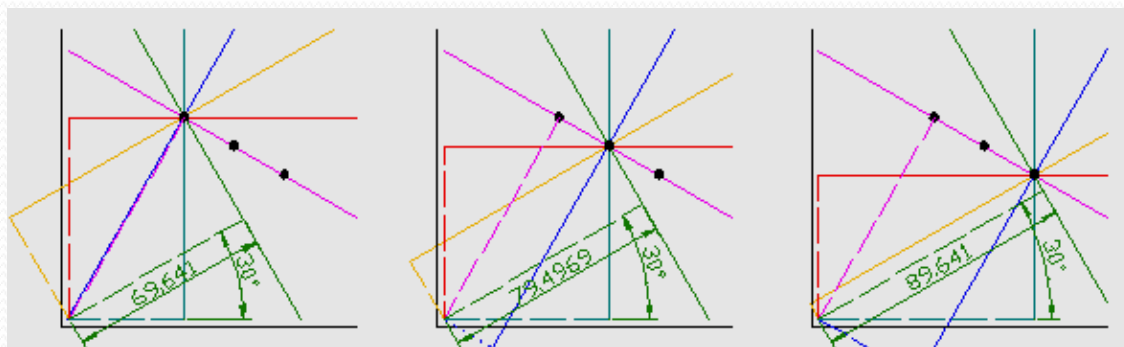
# Поиск примитивов.

## Преобразование Хафа

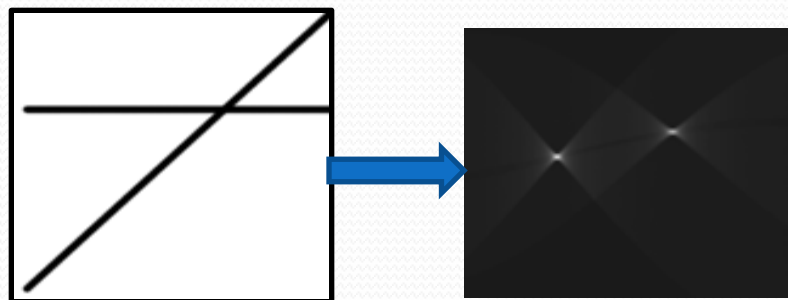
- Задача - найти линии на изображении формируемые точками бинарного изображения

$$y = \left( -\frac{\cos \theta}{\sin \theta} \right) x + \left( \frac{r}{\sin \theta} \right), \quad \rho = x \cos \theta + y \sin \theta$$

- Каждая точка бинарного изображения формирует множество гипотез о том какие линии через неё могут проходить.



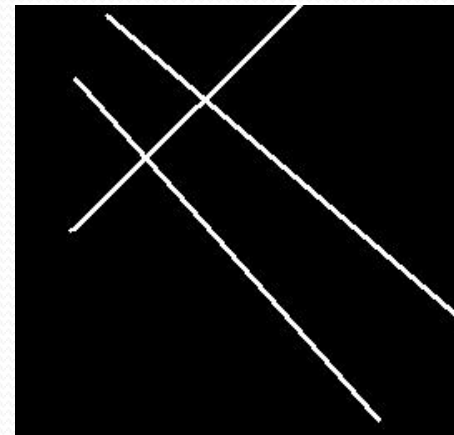
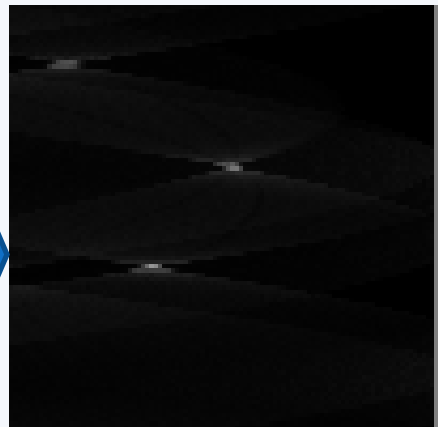
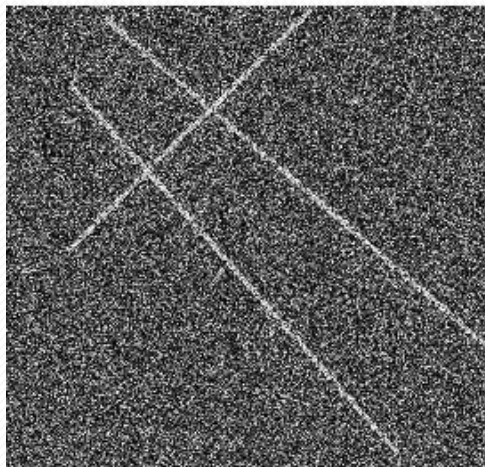
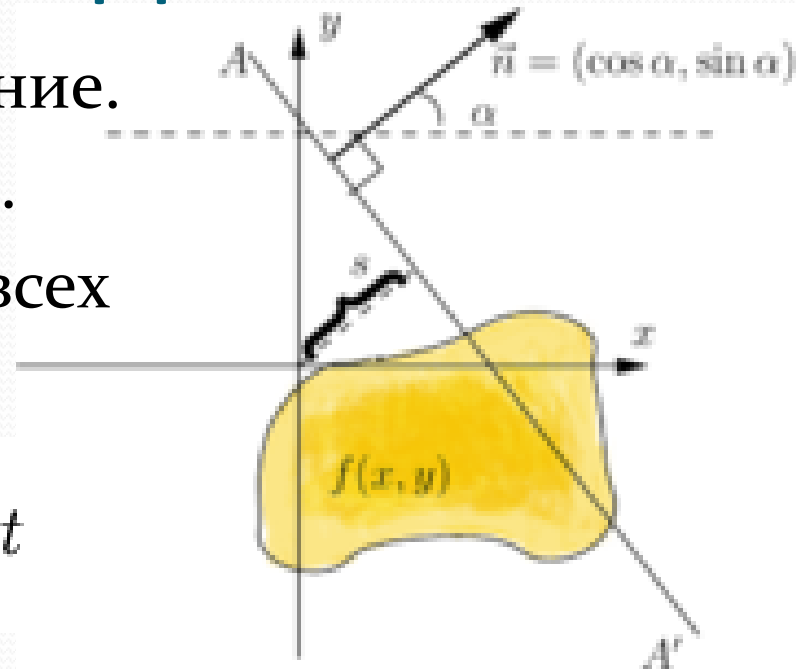
- Все гипотезы аккумулируются.
- Выбираются локальные максимумы в пространстве гипотез.



# Преобразование Радона

- Вход – полутоновое изображение.
- Задача - найти светлые линии.
- Вычисляется интеграл вдоль всех прямых

$$\mathcal{R}[f](\alpha, s) = \int_{-\infty}^{\infty} f(x(t), y(t)) dt$$



# Теорема о свёртке

- Преобразование Фурье над двумя свернутыми функциями пропорционально результатам преобразований Фурье над каждой функцией в отдельности

$$\mathbf{F}(f(x,y) \star h(x,y)) \cong \mathbf{F}(f(x,y))\mathbf{F}(h(x,y)) = \mathbf{F}(u,v)\mathbf{H}(u,v)$$

- Алгоритм фильтрации изображения  $f(x,y)$  фильтром  $h(x,y)$ 
  1. Получить преобразования Фурье  $F(u,v)$ ,  $H(u,v)$
  2. Поточечно перемножить  $F(u,v)$  и  $H(u,v)$  получить  $F'(x,y)$
  3. Применить обратное преобразование Фурье для получения сглаженного изображения.

# Интегральное изображение

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y'),$$

1	2	5	1	2		1	3	8	9	11
2	20	50	20	5		3	25	80	101	108
5	50	100	50	2		8	80	235	306	315
2	20	50	20	1		10	102	307	398	408
1	5	25	1	2		11	108	338	430	442
5	2	25	2	5		16	115	370	464	481
2	1	5	2	1		18	118	378	474	492

