

Компьютерное зрение

Лекция 1. Базовая теория

14.01.2020

Руслан Алиев

План курса

1. Классическое компьютерное зрение

- a. Базовая теория
- b. Форматы представления изображения
- c. Морфологические операции
- d. Нахождение краев
- e. Ключевые точки - детекторы и дескрипторы
- f. Сегментация изображения и детекция объектов

План курса

2. Применение нейросетей для задач CV

- a. Принцип работы нейронных сетей
- b. Нейронные сети для обработки изображений
- c. Особенности обучения нейронных сетей
- d. Сегментация и детекция с помощью нейронных сетей
- e. Задача metric learning. Распознавание лиц.
- f. Латентное представление изображений. Автоэнкодеры.
- g. Генерация изображений. GAN, VAE.
- h. Читаем вместе статью с конференции
- i. Рекуррентные сети для изображений. Генерация описания по изображению.

Наш гитхаб

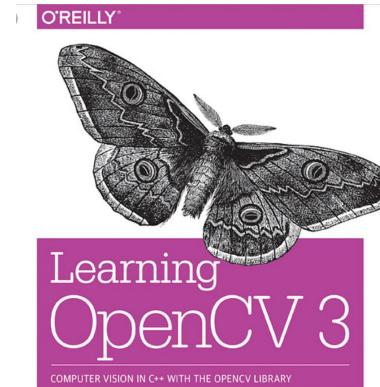
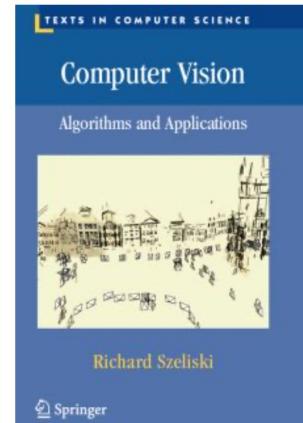
Ссылка

- Слайды с лекций
- Задания на семинар
- Домашки
- Доп. материалы

Рекомендуемые материалы

КНИГИ

- **Computer vision. Algorithms and Applications,**
Richard Szeliski
[Доступна онлайн](#)
- **Learning OpenCV3,**
Adrian Kaehler
[Осторожно C++!](#)



Adrian Kaehler & Gary Bradski

Рекомендуемые материалы

<http://cs231n.stanford.edu>

Крутые дипломные проекты!

<http://cs231n.stanford.edu/2017/reports.html>

CS231n: Convolutional Neural Networks for Visual Recognition

Spring 2019

Previous Years: [Winter 2015] [Winter 2016] [Spring 2017] [Spring 2018]



Рекомендуемые материалы

ods.ai

- #cv
- #deep_learning
- #article_essence
- #theory and practice



Рекомендуемые материалы

YouTube channels

- [ML Trainings](#)
- [Carnegie-Mellon University Deep Learning](#)

Рекомендуемые материалы

Twitter

- [Yann LeCun](#)
- [Andrej Karpathy](#)
- [Pieter Abbeel](#)
- [Ruslan Salakhutdinov](#)
- [Ian Goodfellow](#)
- [Jeremy Howard](#)

Рекомендуемые материалы

Twitter

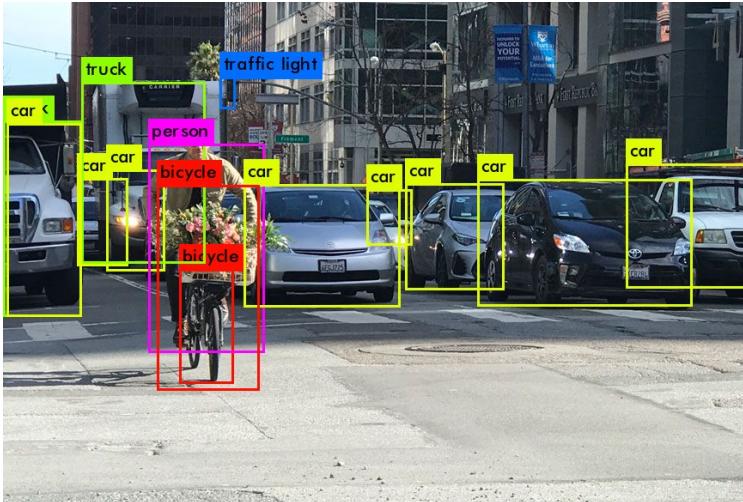
- [Yann LeCun](#)
- [Andrej Karpathy](#)
- [Pieter Abbeel](#)
- [Ruslan Salakhutdinov](#)
- [Ian Goodfellow](#)
- [Jeremy Howard](#)

The screenshot shows a Twitter thread from Andrej Karpathy (@karpathy). The first tweet is from him at 11:03 PM on Sep 20, 2019, on the Twitter Web App. He writes: "We see more significant improvements from training data distribution search (data splits + oversampling factor ratios) than neural architecture search. The latter is so overrated :)" This tweet has 317 Retweets and 1.7K Likes. Below it is a reply from Andrej Karpathy (@karpathy) at 2020-09-20, replying to @karpathy: "(likely an artifact of most of academia focused on finding models conditioned on standard datasets)". This reply has 11 replies, 16 retweets, and 253 likes. Next is a reply from Aderemi Adeola (@21st_title) at 2020-09-20, replying to @karpathy: "Could you explain how to go about this(training data distribution search) or point to a paper that does?". This reply has 5 replies, 1 retweet, and 15 likes. Finally, there is another reply from Andrej Karpathy (@karpathy) at 2020-09-20, replying to @karpathy: "There's no paper. That's exactly the problem.". This reply has 17 replies, 3 retweets, and 111 likes.

План лекции

1. Введение
 - a. Что такое компьютерное зрение
 - b. Какие задачи решает компьютерное зрение
 - c. Сложности компьютерного зрения
2. Как видит машина и как видит человек
 - a. Глаз человека
 - b. Цифровое изображение
 - c. Цвет
 - d. Форматы представления изображения
3. Морфологические операции с изображениями

Что видно на изображении?



Задача зрения: понять, что изображено на картинке

Компьютерное зрение: построение математической модели человеческого зрения (**искусственный интеллект!!!**)

«Тест Тьюринга» - компьютер должен ответить на любой вопрос об изображении, на который может ответить человек

«To see means to know what is where by looking»

David Marr, Vision, 1982

Немного терминологии

	Input	Output
Обработка сигналов (Signal processing)	Цифровой сигнал	Цифровой сигнал
Обработка изображений (Image processing)	Изображение	Изображение
Компьютерное зрение (Computer vision)	Изображение	Информация/признаки изображения
Машинное обучение (Machine learning)	Информация/признаки	Информация
Robot vision	Изображения	Действия

Немного терминологии

Наши
темы

	Input	Output
Обработка сигналов (Signal processing)	Цифровой сигнал	Цифровой сигнал
Обработка изображений (Image processing)	Изображение	Изображение
Компьютерное зрение (Computer vision)	Изображение	Информация/признаки изображения
Машинное обучение (Machine learning)	Информация/признаки	Информация
Robot vision	Изображение	Действия

Задачи компьютерного зрения

Классификация изображений



mite

container ship

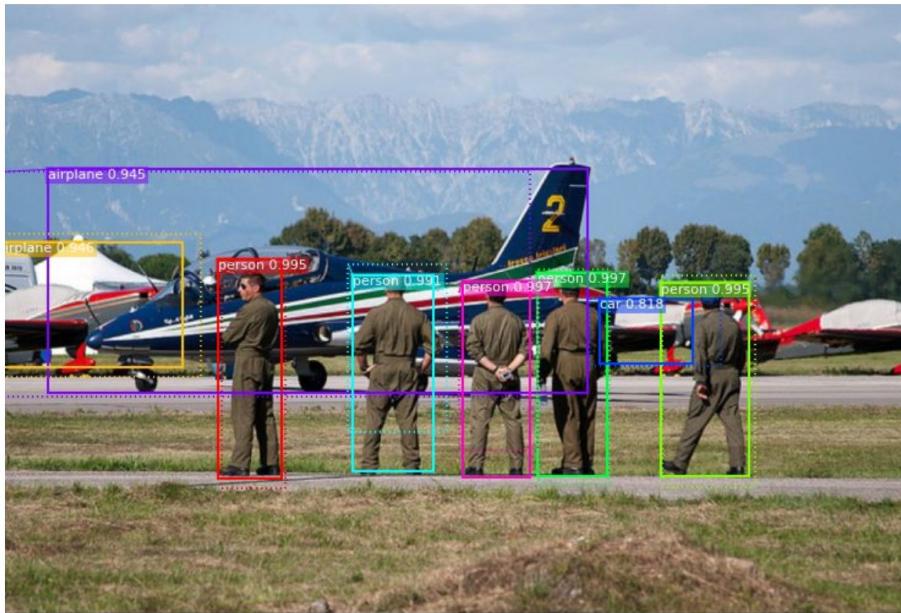
motor scooter

leopard



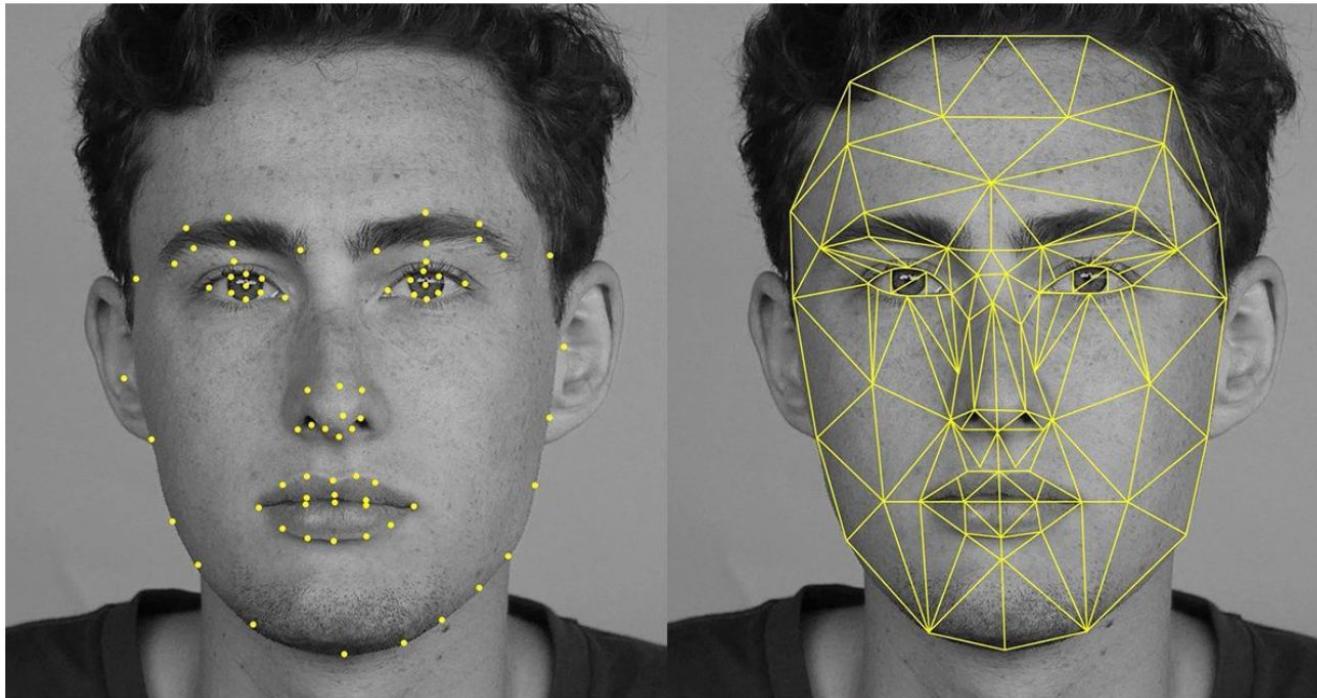
Источник: Krizhevsky et al. *ImageNet Classification with Deep Convolutional Neural Networks*

Детекция объектов



Источник: https://github.com/matterport/Mask_RCNN

Обнаружение ключевых точек

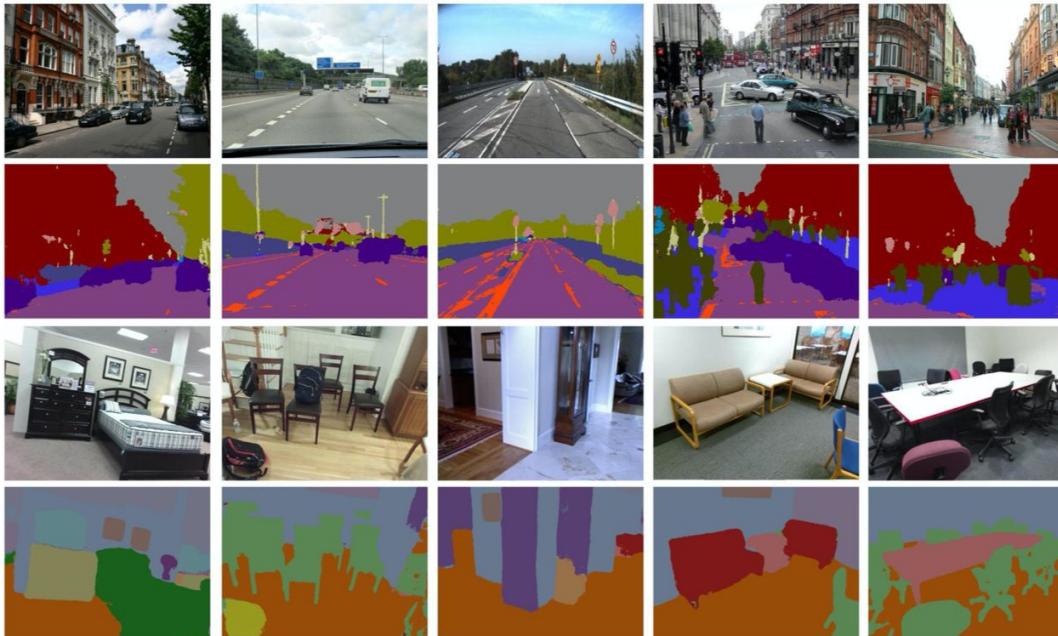


Сегментация изображения



Источник: <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html>

Семантическая сегментация изображения



Источник: <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html>

Аннотация изображений

A person riding a motorcycle on a dirt road.



Two dogs play in the grass.



A skateboarder does a trick on a ramp.



A dog is jumping to catch a frisbee.



A group of young people playing a game of frisbee.



Two hockey players are fighting over the puck.



A little girl in a pink hat is blowing bubbles.



A refrigerator filled with lots of food and drinks.



A herd of elephants walking across a dry grass field.



A close up of a cat laying on a couch.



A red motorcycle parked on the side of the road.

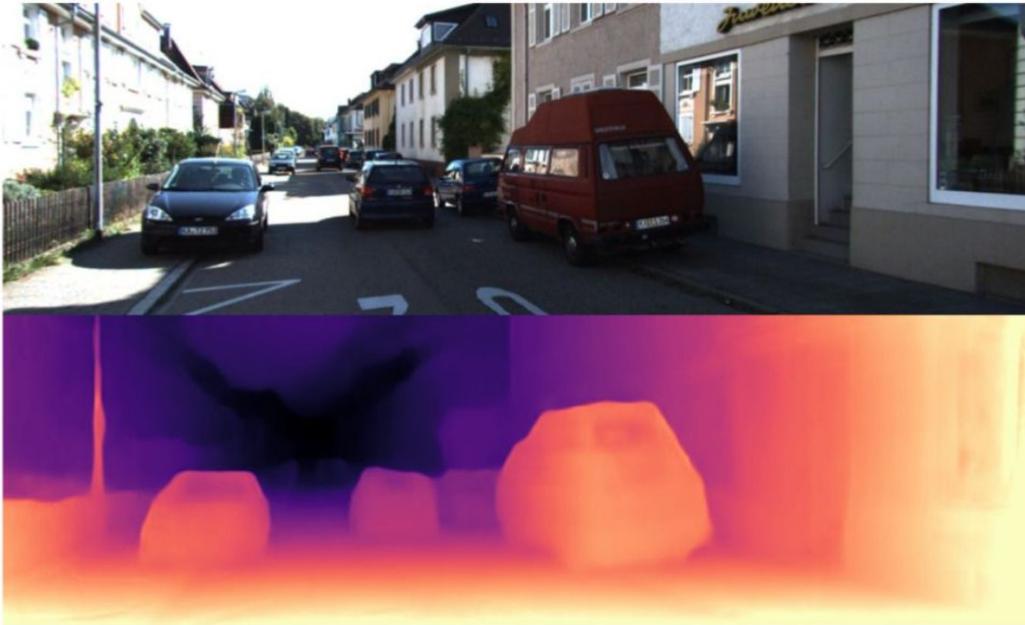


A yellow school bus parked in a parking lot.



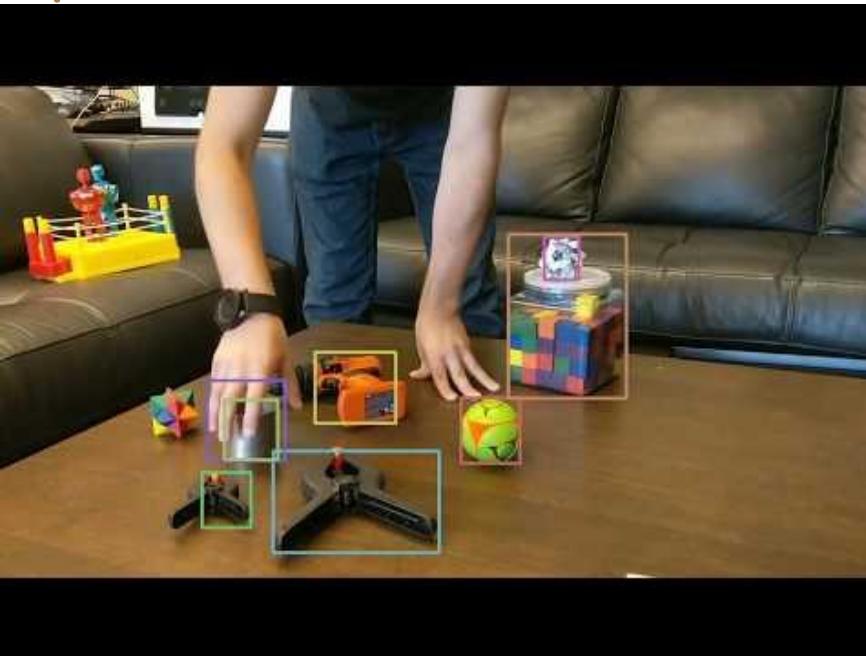
ИСТОЧНИК: Vinyals et al. *Show and Tell: A Neural Image Caption Generator*

Фотограмметрия (depth estimation)



Источник: <https://github.com/nianticlabs/monodepth2>

Трекинг объектов



Источник: <https://arxiv.org/abs/1705.06368>

Задачи обработки изображений

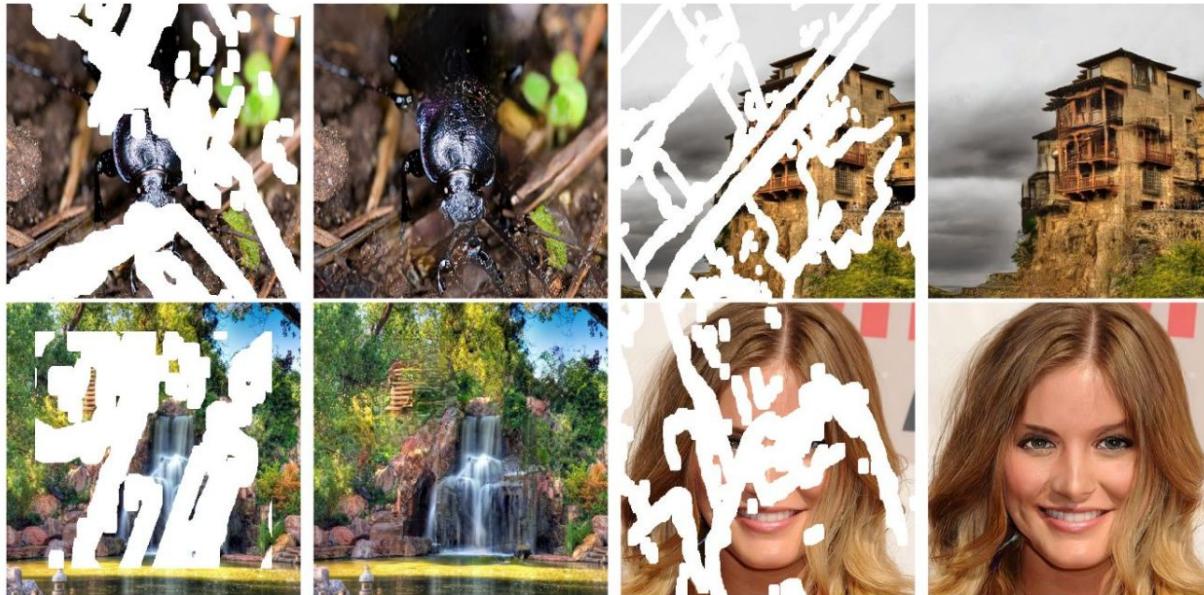
Superresolution



Figure 2: From left to right: bicubic interpolation, deep residual network optimized for MSE, deep residual generative adversarial network optimized for a loss more sensitive to human perception, original HR image. Corresponding PSNR and SSIM are shown in brackets. [4× upscaling]

Источник: Ledig et al. *Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network*

Восстановление изображения (inpainting)



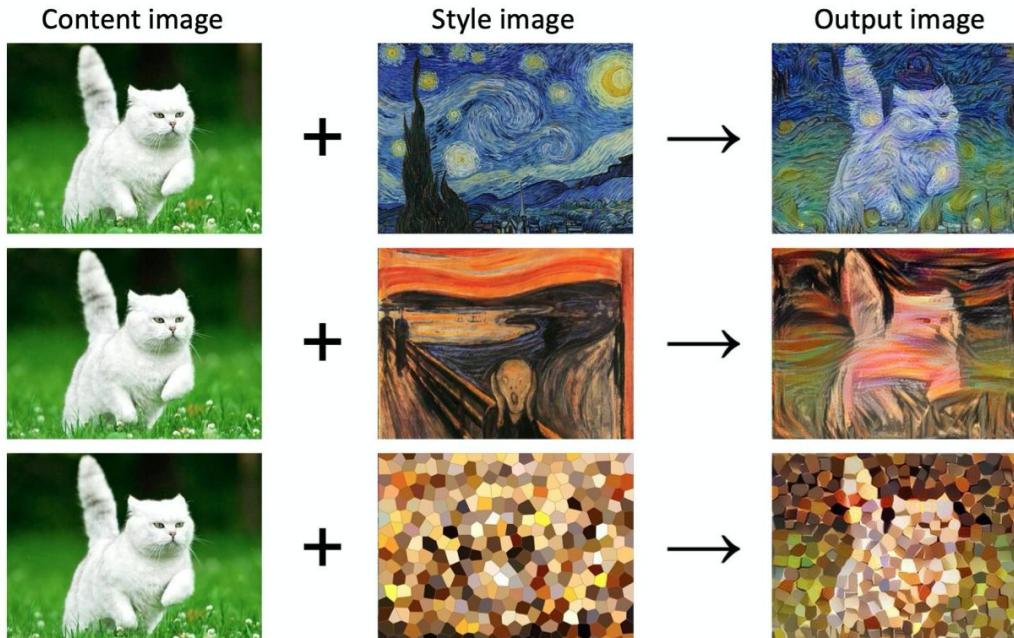
Источник: Liu et al. *Image Inpainting for Irregular Holes Using Partial Convolutions*

Image generation



Источник: Gravity, Jurassic Park, Avatar

Перенос стиля (style transfer)



Источник: <https://blog.godatadriven.com/images/how-to-style-transfer>

Face generation



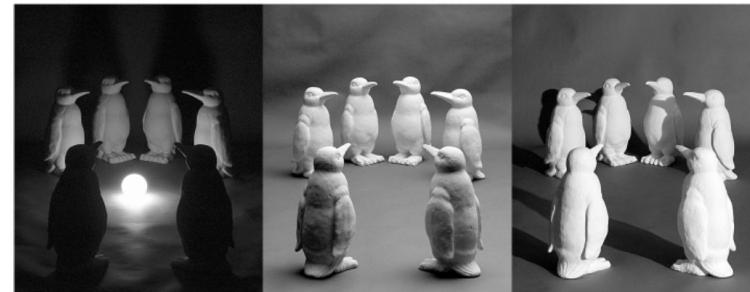
Источник: <https://arxiv.org/pdf/1812.04948.pdf>

Почему CV - это сложно?

Изменчивость изображений



Ракурс



Освещение



Внутриклассовые вариации



Размер

Изменчивость изображений



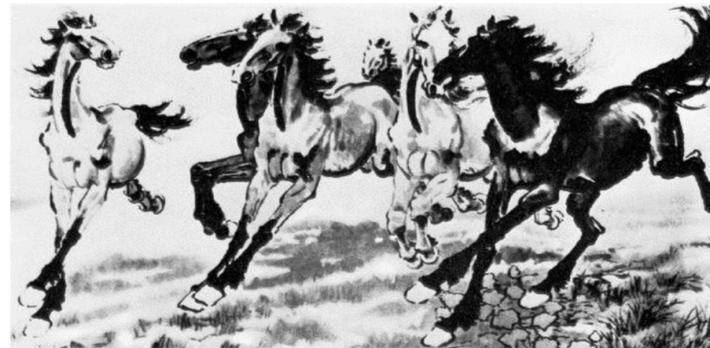
Перекрывание объектов



Маскировка



Движение



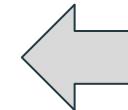
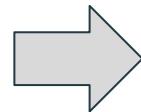
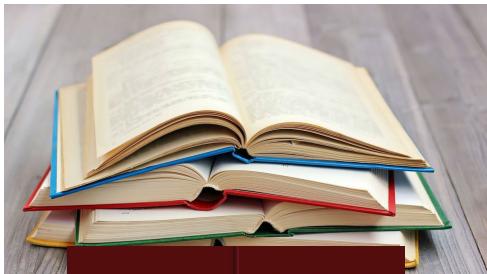
Деформация

Неоднозначность интерпретаций



Одно 2D изображения допускает несколько 3D-интерпретаций

Зрение человека

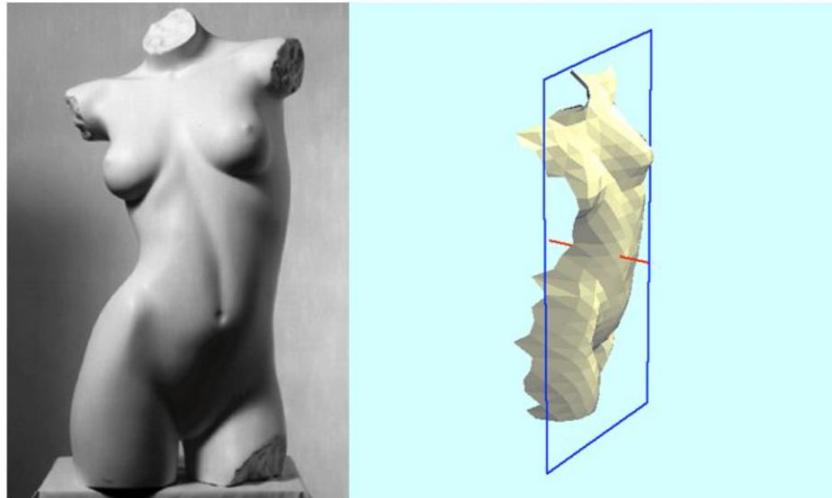


- Мы сопоставляем наблюдения (подсказки) и априорные знания для извлечения информации из того, что видим
- Этим занято 25 % мозга
- Какие подсказки вы можете придумать?

Цвет



Освещение

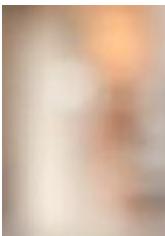


Источник: J. Koenderink

Текстура



Контекст

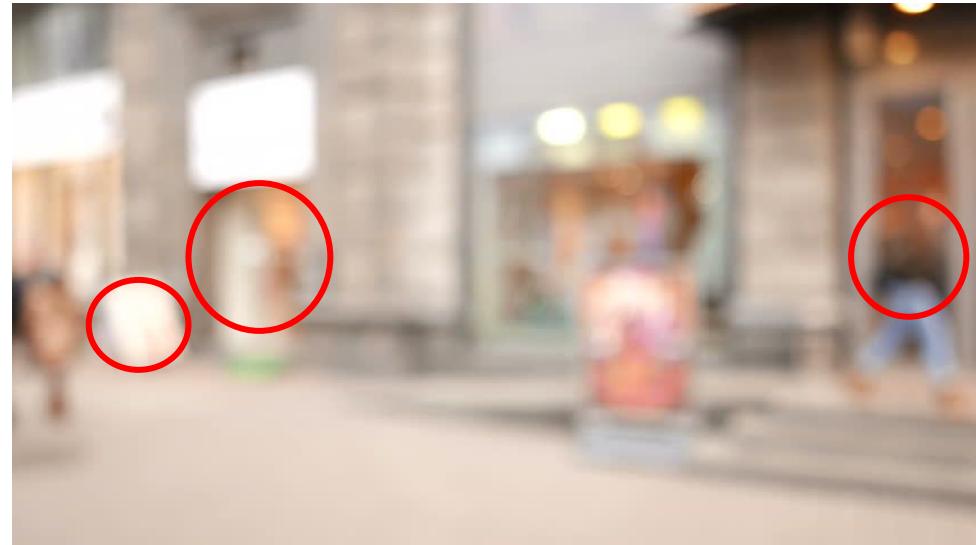


Можете
определить,
какие это
объекты?

Контекст



Можете
определить,
какие это
объекты?



Контекст

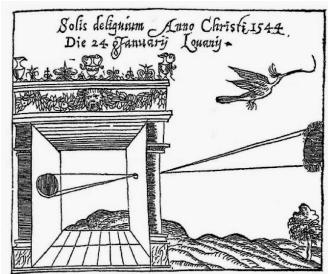


Источник: Fei-Fei, Fergus & Torralba

Небольшой экскурс в историю CV

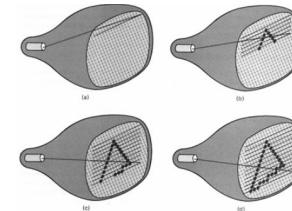
Начало начал

- Древняя Греция - камера обскура
Принцип был известен еще во времена Аристотеля
- Ренессанс
Снова появляется перспектива
Воспроизведение мельчайших деталей с помощью системы линз
Появились хорошее стекло для линз
- Камера-люцида (1807)
Художник, глядя на холст одним глазом, видит реально изображение, другим - рисунок и саму руку
- Первая фотография (1826)
Требовала 8 часов проявки
Появляется фотограмметрия (измерение расстояние между объектами по 2D-изображению)
- Первая скоростная съемка (1878), Eadweard Muybridge
- Первое кино на пленке (1888), Louis Le Prince

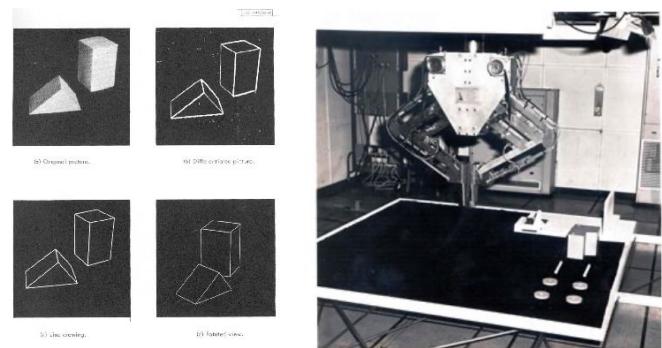


20 век

- Первый растровый дисплей (1927)
- WhirlWind, MIT, Первый компьютер, отображающий текст и графику в реальном времени на мониторе (1951)
- Первое цифровое изображение, Russell Kirsch, (1959)
176x176 пикселей
- L. G. Roberts, *Machine Perception of Three Dimensional Solids*, (1963)
- Freddy II, один из первых роботов с системой машинного зрения (1973)



Philo Farnsworth – 60-строчный растровый дисплей



Дэвид Марр

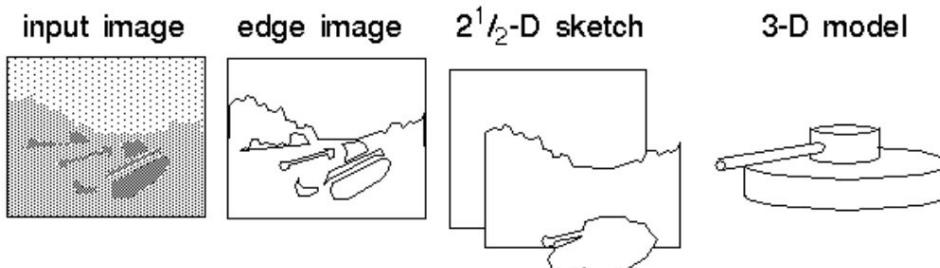
Компьютерное зрение - иерархическое. Главная цель - восстановить 3D-мир.

- «Primal sketch» - Низкоуровневые («low-level») свойства изображения: направленные края, отрезки и т.д.
- «2.5D sketch» - упорядочивание по глубине (бинокулярное стерео), учёт текстуры и т.д.
- «3D model» - распознавание объектов и представление о 3x мерном мире

“The Marr Prize” – главная премия в области компьютерного зрения



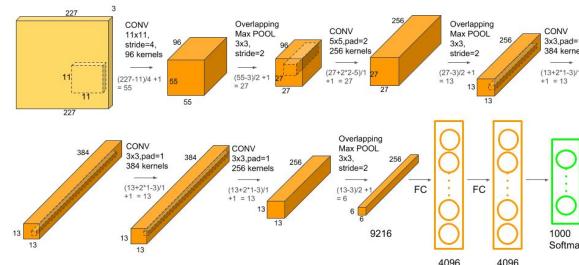
David Marr, 1982,
*Vision: A computational
investigation into the human
representation and processing of
visual information*



Источник: <http://www.doc.gold.ac.uk/~mas02fl/MSC101/Vision/Marr.html>

21 век

- SIFT Algorithm, (1999)
До сих пор не потерял актуальность
Самая цитируемая статья по CV (55000)
- Детектор лиц Viola-Jones (2001)
Первый быстрый и надежный алгоритм поиска лиц
- Pascal VOC Dataset (2006)
Первый масштабный датасет для классификации изображений
20 категорий
- AlexNet выигрывает соревнование ImageNet (2012)
CNN захватили компьютерное зрение

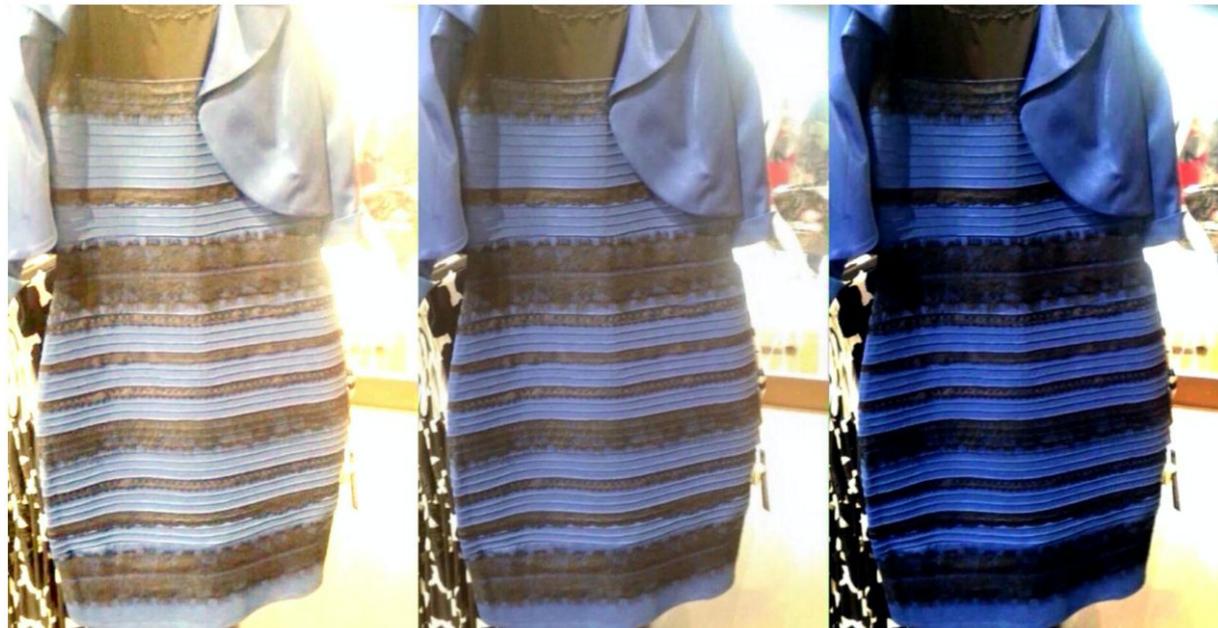


Как видит человек



Какого цвета платье? :)

Как видит человек



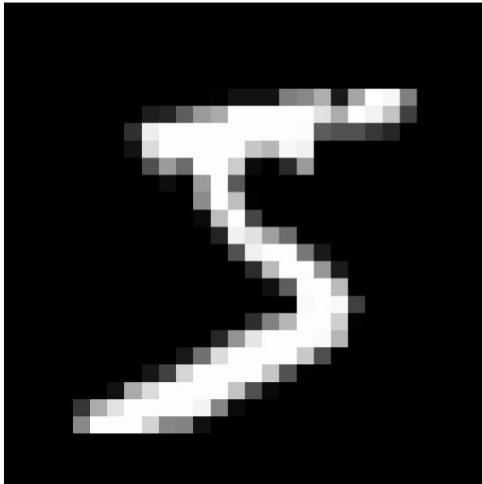
Человеческое зрение субъективно: кто-то видит бело-золотое, кто-то темно-синее платье

Как видит человек



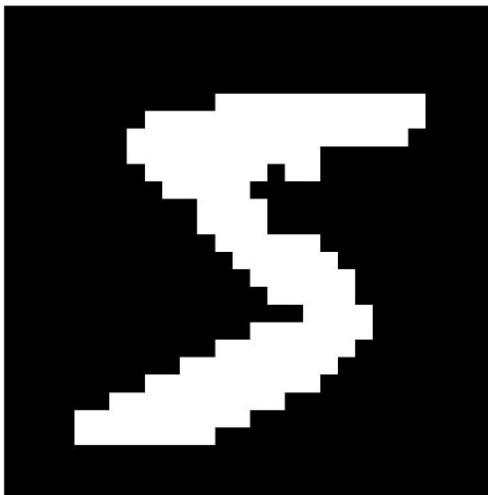
Человеческое зрение субъективно: кто-то видит старую женщину, кто-то молодую девушку

Как видит машина

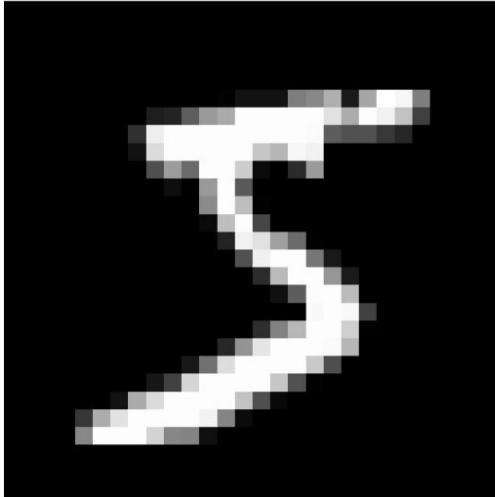


Матрица!

Как видит машина: бинарный формат



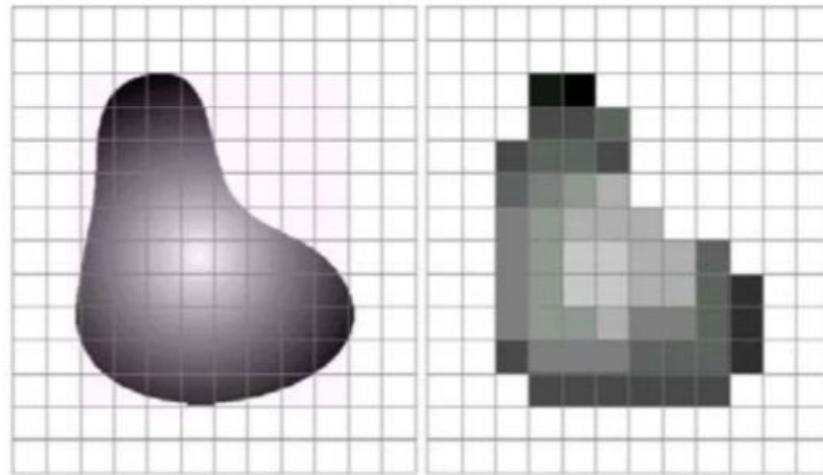
Как видит машина: байтовый формат



Целые значения в диапазоне 0-255 (1 байт на каждый пиксель)

Как видит машина

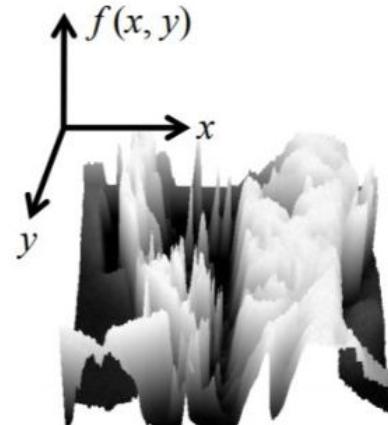
Мы можем рассматривать изображения, как дискретное представление 2D-функции.



Source: N. Snavely

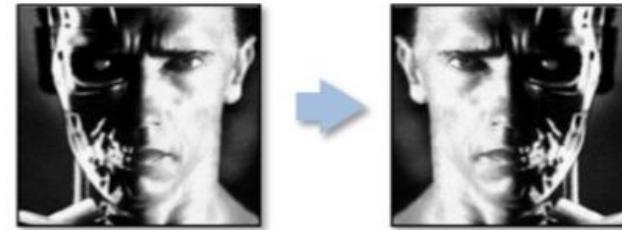
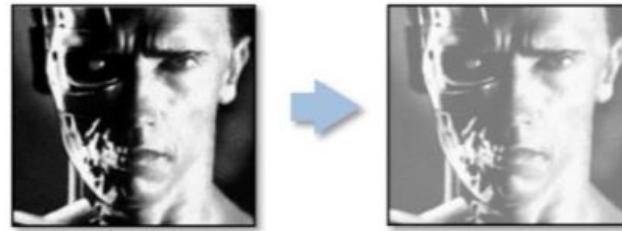
Как видит машина

Мы можем рассматривать изображения, как дискретное представление 2D-функции.



Как видит машина

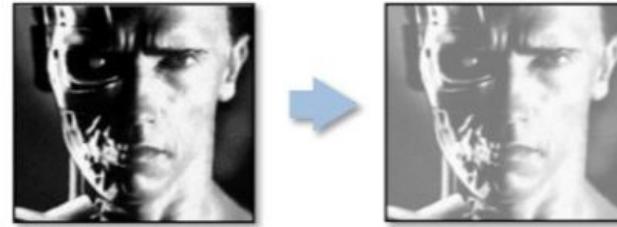
Применять к изображениям стандартные операции



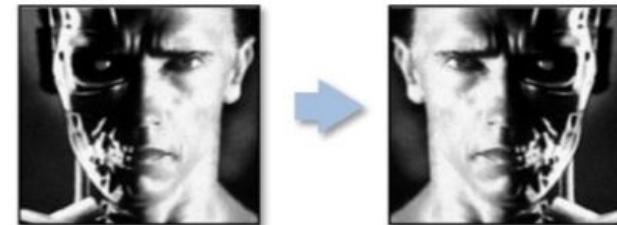
Source: N. Snavely

Как видит машина

Применять к изображениям стандартные операции



$$g(x,y) = f(x,y) + 20$$



$$g(x,y) = f(-x,y)$$

Source: N. Snavely

Цвет

Цветные фотографии



Что такое цвет?

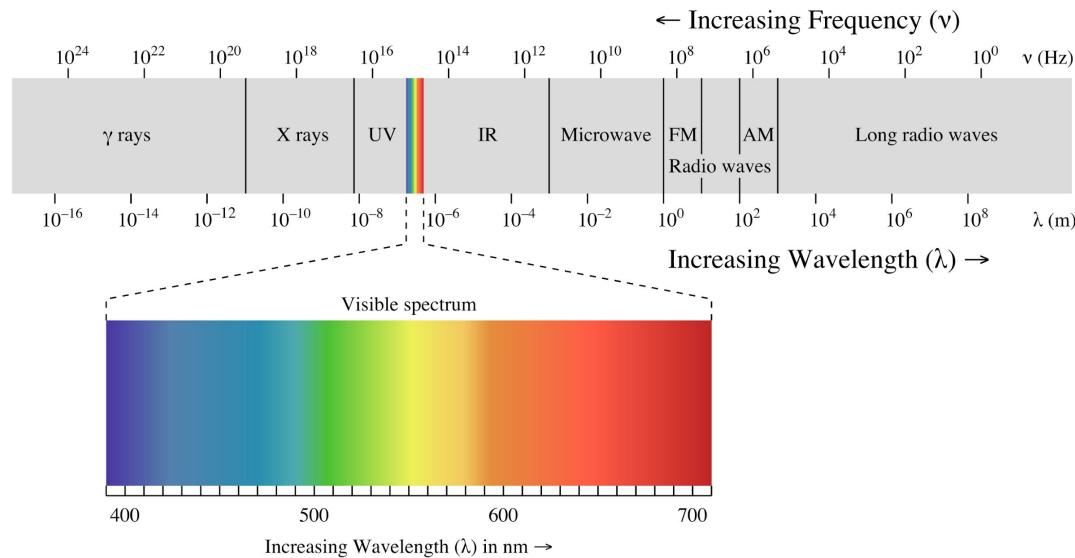
Свет - физическое понятие (электромагнитная волна)

Цвет - психологическое свойство нашего зрения, возникающие при наблюдении объектов и света.

Цвет - результат взаимодействия сцены, света и нашего мозга.

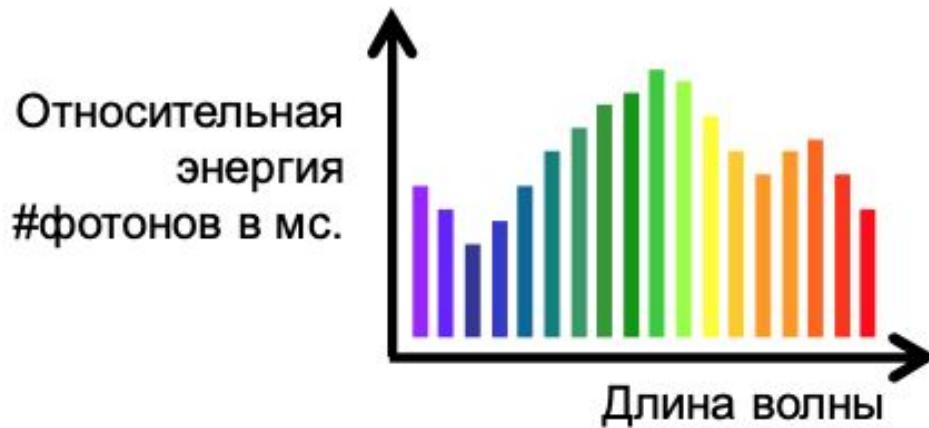


Видимый свет



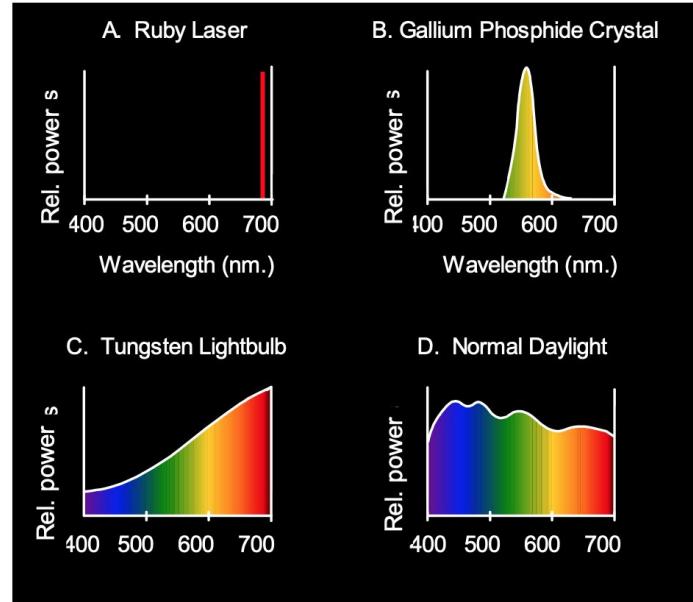
Мы видим электромагнитное излучение в районе 380нм-780нм

Физика света



Любой источник света можно полностью описать спектром: количеством излученной энергии в единицу времени для каждой длины волны (в интервале 380-780 нм)

Физика света

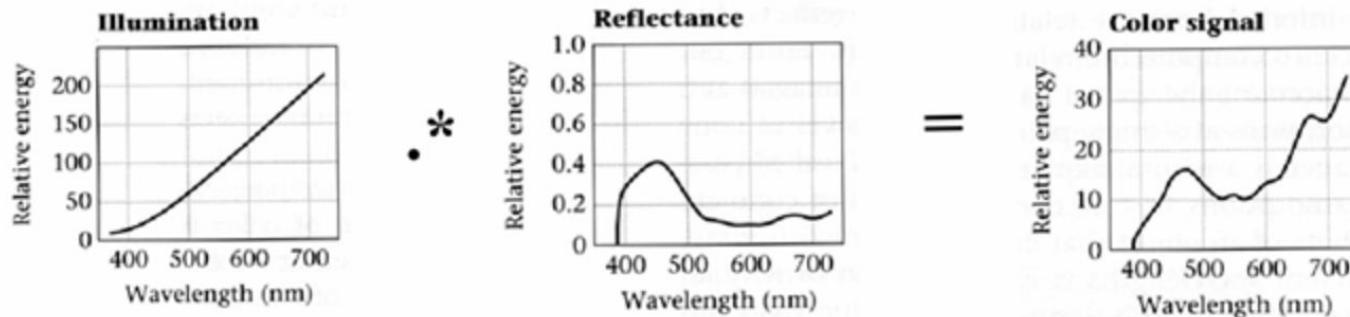


Примеры спектров разных источников света.

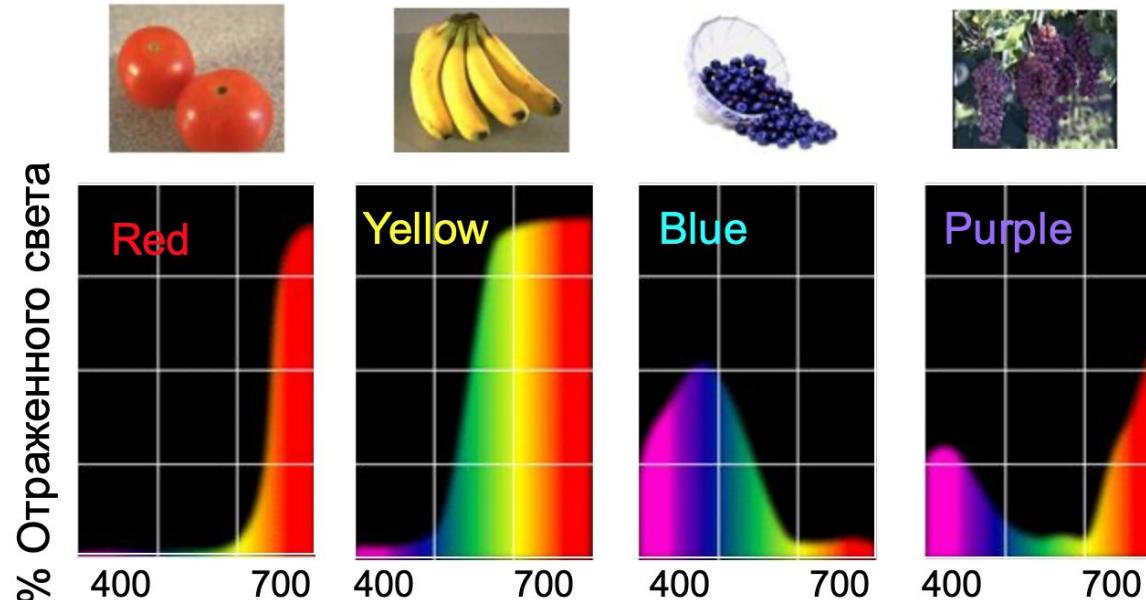
Взаимодействие света и объектов



Отраженный свет это результат взаимодействия излучаемого света и поверхности

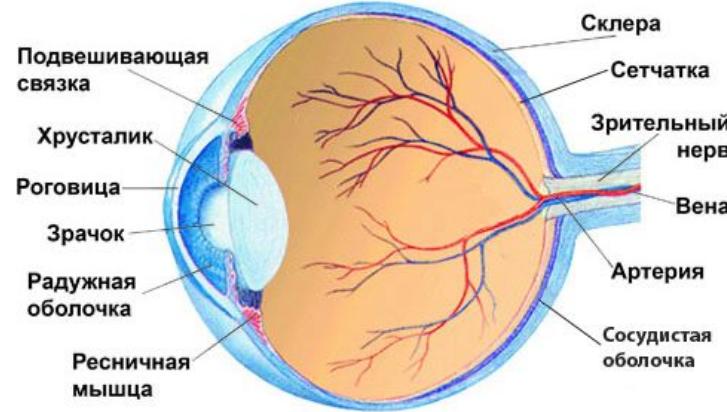
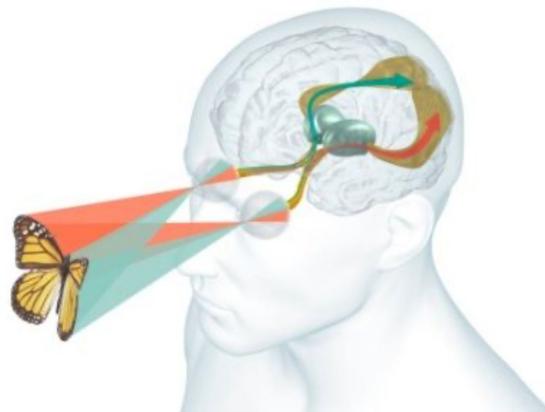


Спектры отраженного света для различных объектов



Источник: S. Lazebnik

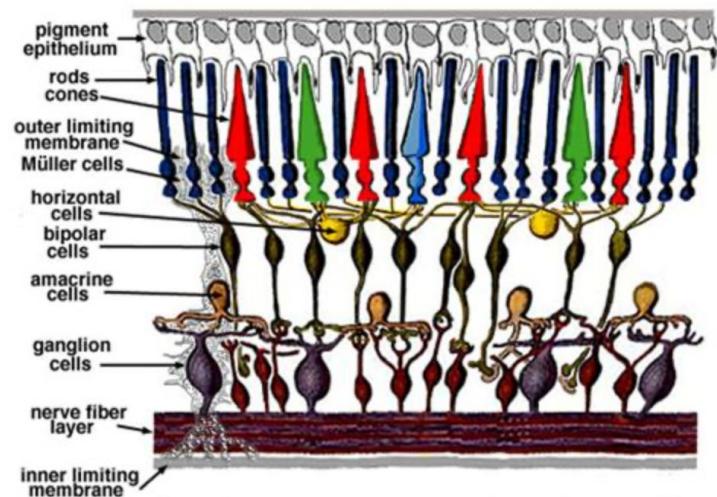
Как видит человек



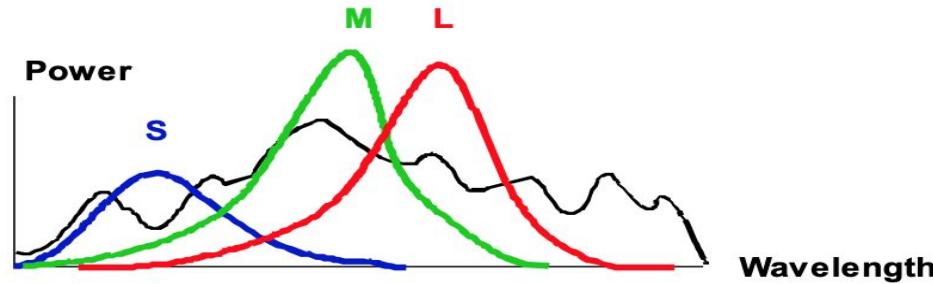
- **Радужка** – цветная пленка с радиальными мышцами
- **Зрачок** - отверстие, диаметр управляетяется радужкой
- **Хрусталик** – «линза», меняющая форму под действием мышц (позволяет менять фокус)
- Где матрица?
Клетки-фоторецепторы на сетчатке

Как видит человек

- Палочки (Rods) измеряют яркость
- Колбочки (Cones) измеряют цвет



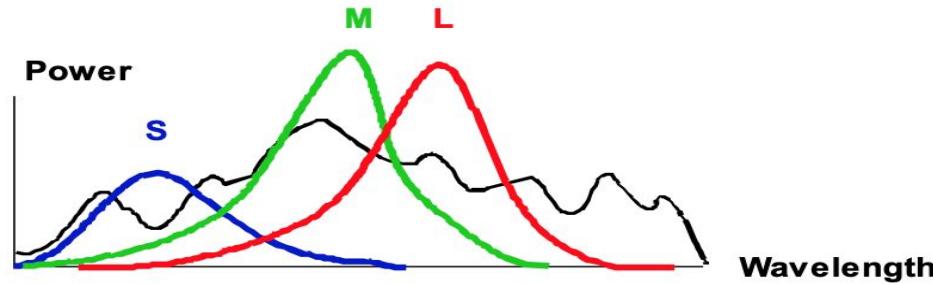
Как видит человек



Палочки и колбочки - фильтры спектра

Каждый тип колбочек дает 1 число

Как видит человек



Палочки и колбочки - фильтры спектра

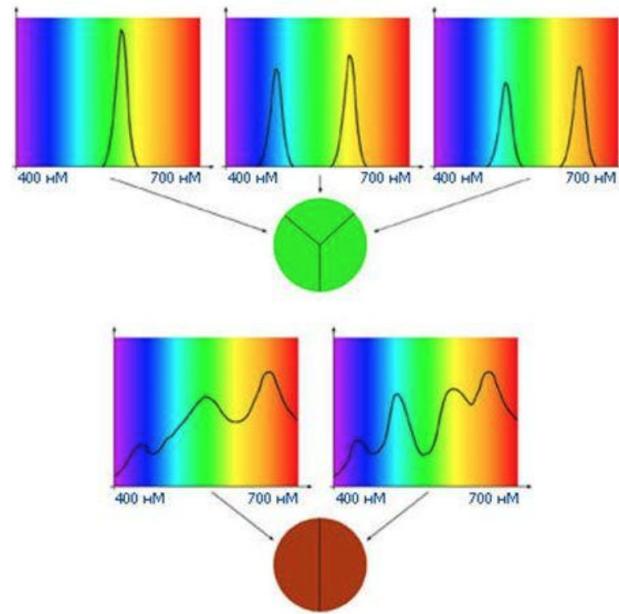
Каждый тип колбочек дает 1 число

Два разных спектра могут быть неотличимы (давать один “цвет”)

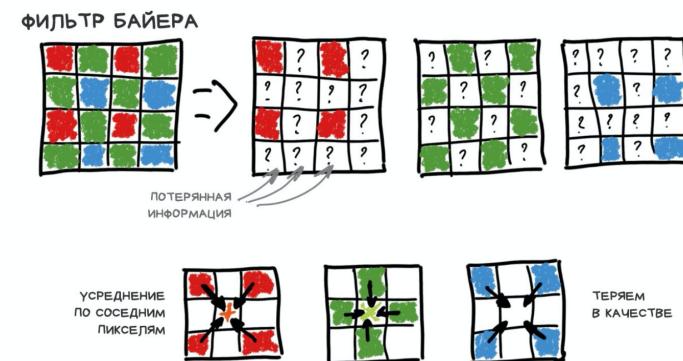
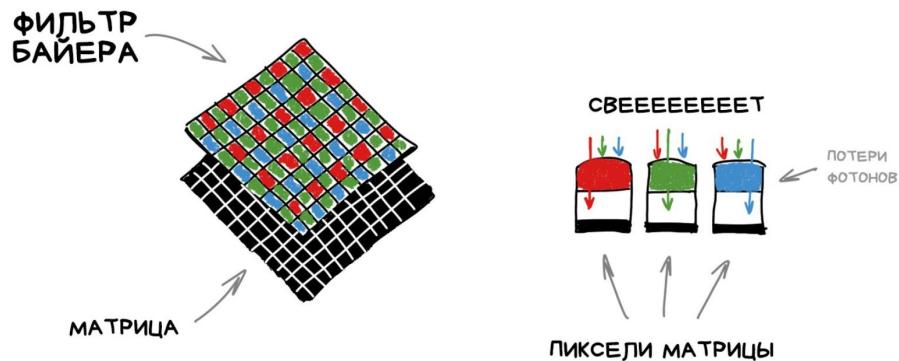
Метамеры

Источник: Steve Seitz

Метамеры



Фильтр байера



Источник:

https://vas3k.ru/blog/computational_photography/

Цветовые пространства

Цифровое цветное изображение

Трехмерная матрица (тензор)
размера NxMxC

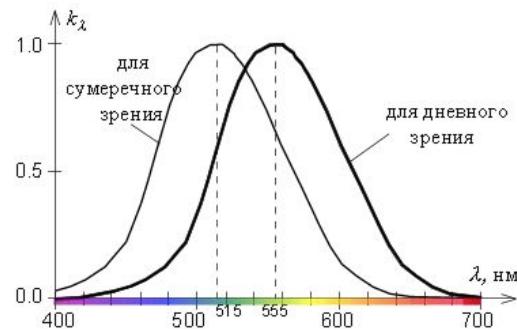
Каждое значение - 1 байт

		165	187	209	58	7
	14	125	233	201	98	159
253	144	120	251	41	147	204
67	100	32	241	23	165	30
209	118	124	27	59	201	79
210	236	105	169	19	218	156
35	178	199	197	4	14	218
115	104	34	111	19	196	
32	69	231	203	74		

Яркость и цвет

2 основные характеристики света, воспринимаемые человеком:

- Яркость (brightness)
- Цвет (chromasity)



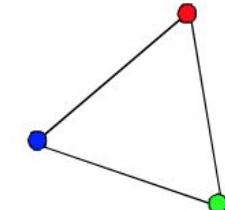
Функция чувствительности глаза
человеку к свету (воспринимаемая
яркость)

В цвете можно выделить тон (hue) и насыщенность (saturation)

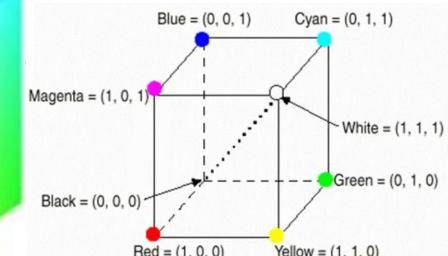
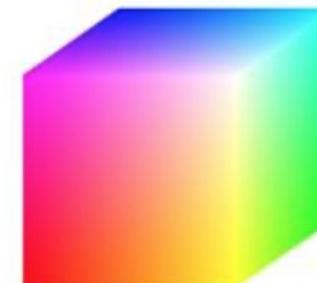
Illuminating and Reflecting Light

- Излучающие источники (первичный свет):
 - Излучают свет (солнце, монитор)
 - Воспринимаемый цвет зависит от частоты ЭМ волны
 - Аддитивное правило: $R+G+B=White$
- Отражающие источники (вторичный свет):
 - Отражают поступающий свет (краска, одежда)
 - Воспринимаемый цвет зависит от отраженного цвета ($= emitted freq - absorbed freq$)
 - Вычитающие правила: $R+G+B=Black$

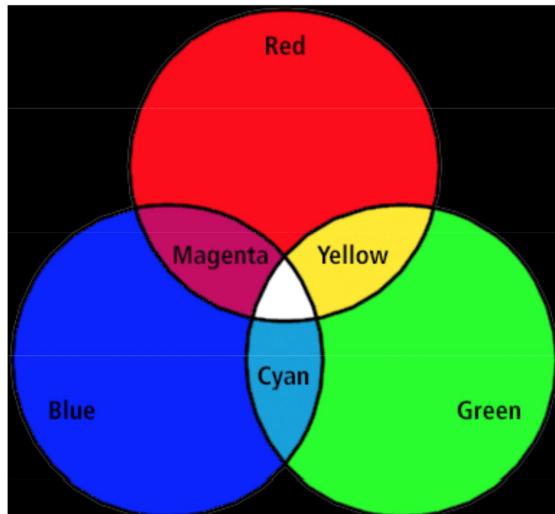
Трихроматическая теория цвета



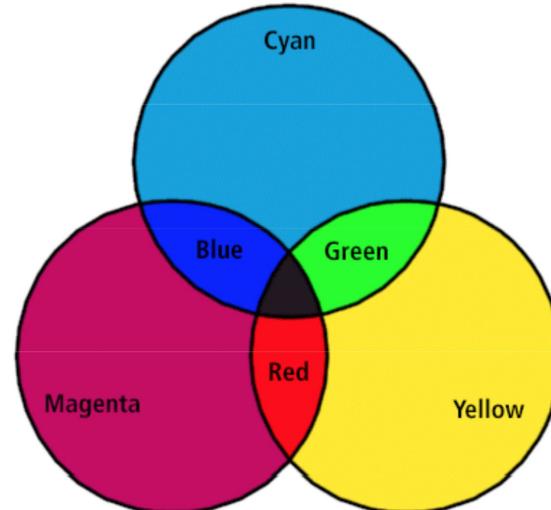
- Любой цвет может быть получен смешиванием трех основных цветов в правильном отношении
- Основные цвета для излучающих источников:
 - R, G, B
 - $R + G + B = \text{white}$
- Основные цвета для отражающих источников:
 - Cyan, Magenta, Yellow (CMY)
 - В принтерах CMYK (=CMY + black)
 - $R + G + B = \text{black}$



RGB vs CMY



Magenta = Red + Blue
Cyan = Blue + Green
Yellow = Green + Red



Magenta = White - Green
Cyan = White - Red
Yellow = White - Blue

Цветное изображение



Red

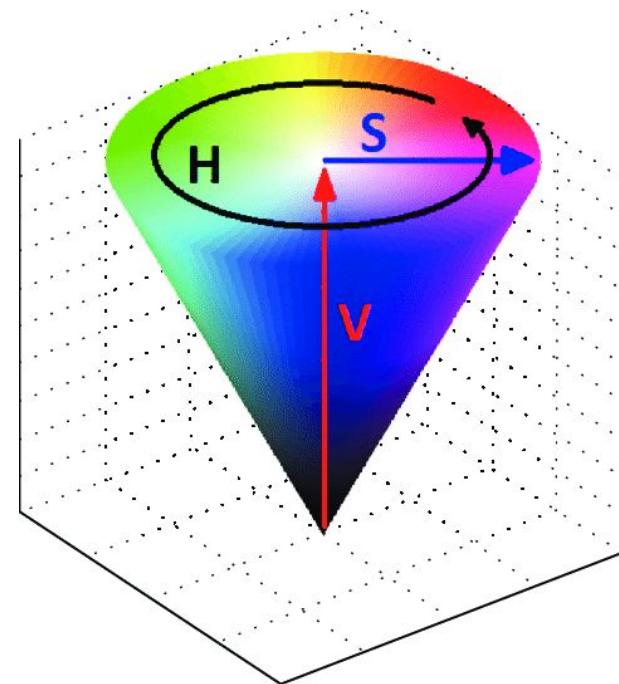
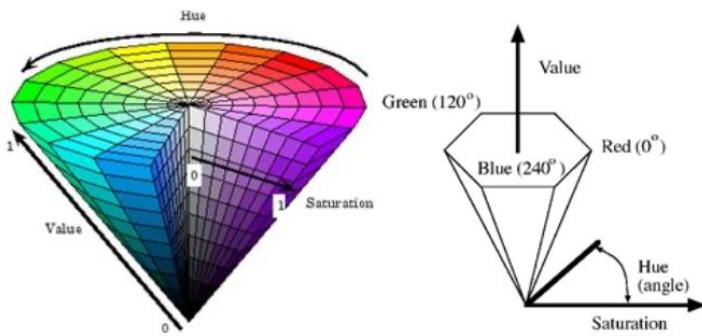
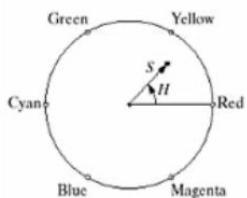
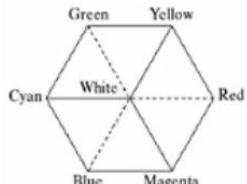


Green



Blue

Модель HSV (HIS)



Координаты выбраны с учетом человеческого восприятия:
Hue (тон), Saturation (насыщенность), Value/Intensity (насыщенность)

Модель цвета

- Определите 3 первичных или вторичных цвета
 - RGB
 - CMY
- Определите luminance и chrominance
 - HSB(=HSV) or HSL = hue, saturation, value
 - YIQ~YUV (аналоговое ТВ)
 - YCbCr (цифровое ТВ)
- Выберите амплитуду
 - 8 битов на компоненту или 24 бита на пиксель

RGB <-> CMY

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \quad \begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} C \\ M \\ Y \end{bmatrix}.$$

RGB <-> HSV

RGB -> HSV

Считаем, что:

$$H \in [0, 360]$$

$$S, V, R, G, B \in [0, 1]$$

Пусть MAX — максимальное значение из R, G и B , а MIN — минимальное из них.

$$H = \begin{cases} 0, \text{ если } MAX = MIN \\ 60 \times \frac{G - B}{MAX - MIN} + 0, \text{ если } MAX = R \text{ и } G \geq B \\ 60 \times \frac{G - B}{MAX - MIN} + 360, \text{ если } MAX = R \text{ и } G < B \\ 60 \times \frac{B - R}{MAX - MIN} + 120, \text{ если } MAX = G \\ 60 \times \frac{R - G}{MAX - MIN} + 240, \text{ если } MAX = B \end{cases}$$

$$S = \begin{cases} 0, \text{ если } MAX = 0; \\ 1 - \frac{MIN}{MAX}, \text{ иначе} \end{cases}$$

$$V = MAX$$

HSV -> RGB

Для любых оттенков $H \in [0, 360]$, насыщенности $S \in [0, 100]$ и яркости $V \in [0, 100]$:

$$H_i = \left\lfloor \frac{H}{60} \right\rfloor \mod 6$$

$$V_{min} = \frac{(100 - S) * V}{100}$$

$$a = (V - V_{min}) * \frac{H \mod 60}{60}$$

$$V_{inc} = V_{min} + a$$

$$V_{dec} = V - a$$

H_i	R	G	B
0	V	V_{inc}	V_{min}
1	V_{dec}	V	V_{min}
2	V_{min}	V	V_{inc}
3	V_{min}	V_{dec}	V
4	V_{inc}	V_{min}	V
5	V	V_{min}	V_{dec}

Свертки

Свертка

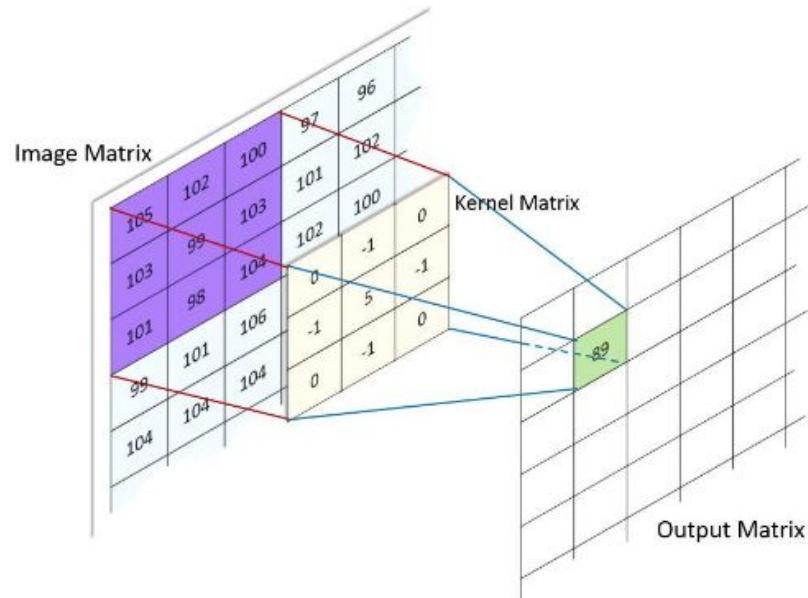
$$G[i, j] = \sum_{u=-k}^k \sum_{v=-k}^k H[u, v] F[i - u, j - v]$$

F - исходно изображение

H - фильтр размера k x k

G - результат на выходе свертки

i,j - координаты пикселя в области которого применяется операция свертки



Свертка

45	60	98	127	132	133	137	133
46	65	98	123	126	128	131	133
47	65	96	115	119	123	135	137
47	63	91	107	113	122	138	134
50	59	80	97	110	123	133	134
49	53	68	83	97	113	128	133
50	50	58	70	84	102	116	126
50	50	52	58	69	86	101	120

$f(x,y)$

*

0.1	0.1	0.1
0.1	0.2	0.1
0.1	0.1	0.1

=

69	95	116	125	129	132
68	92	110	120	126	132
66	86	104	114	124	132
62	78	94	108	120	129
57	69	83	98	112	124
53	60	71	85	100	114

$h(x,y)$

$g(x,y)$

Фильтры/свертки/ядра - названий много

Усредняющий фильтр

Какое ядро подходит для усреднения?

Усредняющий фильтр

Какое ядро подходит для усреднения?

$$\begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

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

Усредняющий фильтр

Что происходит с изображением после применения этого фильтра?

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

Усредняющий фильтр

Что происходит с изображением после применения этого фильтра?

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

Original image



Smoothed image



Фильтры

Что происходит с изображением после применения этого фильтра?

•0	•0	•0
•0	•1	•0
•0	•0	•0

Фильтры

Что происходит с изображением после применения этого фильтра?

•0	•0	•0
•0	•1	•0
•0	•0	•0



Original



•0	•0	•0
•0	•1	•0
•0	•0	•0



Filtered
(no change)

Фильтры

Что происходит с изображением после применения этого фильтра?

•0	•0	•0
•0	•0	•1
•0	•0	•0

Фильтры

Что происходит с изображением после применения этого фильтра?

•0	•0	•0
•0	•0	•1
•0	•0	•0



$$\begin{matrix} * & \begin{matrix} \bullet 0 & \bullet 0 & \bullet 0 \\ \bullet 0 & \bullet 0 & \bullet 1 \\ \bullet 0 & \bullet 0 & \bullet 0 \end{matrix} \end{matrix} =$$



Shifted right
By 1 pixel

Фильтры



Original

$$\begin{matrix} \bullet 0 & \bullet 0 & \bullet 0 \\ \bullet 0 & \bullet 2 & \bullet 0 \\ \bullet 0 & \bullet 0 & \bullet 0 \end{matrix} - \frac{1}{9} \begin{matrix} \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \end{matrix}$$

(Note that filter sums to 1)

= ?

“details of the image”

$$\begin{matrix} \bullet 0 & \bullet 0 & \bullet 0 \\ \bullet 0 & \bullet 1 & \bullet 0 \\ \bullet 0 & \bullet 0 & \bullet 0 \end{matrix} + \begin{matrix} \bullet 0 & \bullet 0 & \bullet 0 \\ \bullet 0 & \bullet 1 & \bullet 0 \\ \bullet 0 & \bullet 0 & \bullet 0 \end{matrix} - \frac{1}{9} \begin{matrix} \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \end{matrix}$$

Фильтры



Original

$$\begin{bmatrix} \bullet 0 & \bullet 0 & \bullet 0 \\ \bullet 0 & \bullet 2 & \bullet 0 \\ \bullet 0 & \bullet 0 & \bullet 0 \end{bmatrix}$$

$$- \frac{1}{9} \begin{bmatrix} \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \end{bmatrix}$$

= ?

(Note that filter sums to 1)

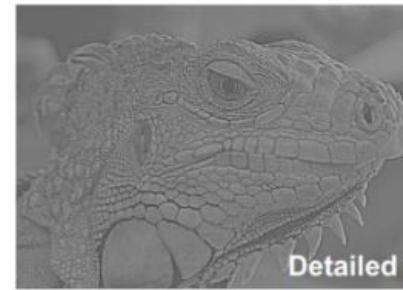
“details of the image”

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

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

$$- \frac{1}{9} \begin{bmatrix} \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \end{bmatrix}$$

Фильтры

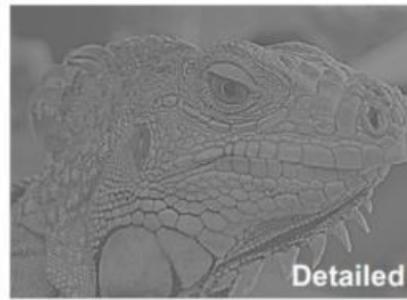


-

=



+



=



Фильтры



Original

$$\begin{bmatrix} \bullet 0 & \bullet 0 & \bullet 0 \\ \bullet 0 & \bullet 2 & \bullet 0 \\ \bullet 0 & \bullet 0 & \bullet 0 \end{bmatrix} - \frac{1}{9} \begin{bmatrix} \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \end{bmatrix}$$

(Note that filter sums to 1)

= ?



Original

$$\begin{bmatrix} \bullet 0 & \bullet 0 & \bullet 0 \\ \bullet 0 & \bullet 2 & \bullet 0 \\ \bullet 0 & \bullet 0 & \bullet 0 \end{bmatrix} - \frac{1}{9} \begin{bmatrix} \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \end{bmatrix} =$$



Фильтры



Original

$$\begin{matrix} \bullet 0 & \bullet 0 & \bullet 0 \\ \bullet 0 & \bullet 2 & \bullet 0 \\ \bullet 0 & \bullet 0 & \bullet 0 \end{matrix} - \frac{1}{9} \begin{matrix} \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \end{matrix} = ?$$

(Note that filter sums to 1)

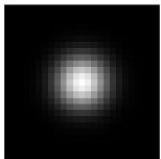
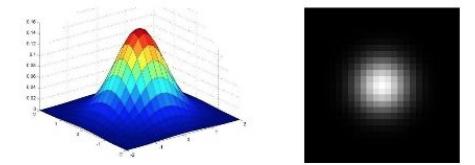


Original

$$\begin{matrix} \bullet 0 & \bullet 0 & \bullet 0 \\ \bullet 0 & \bullet 2 & \bullet 0 \\ \bullet 0 & \bullet 0 & \bullet 0 \end{matrix} - \frac{1}{9} \begin{matrix} \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \end{matrix} =$$

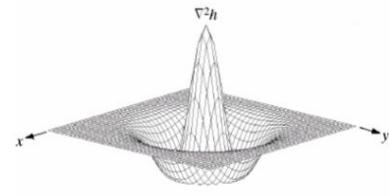


Размытие - фильтр Гаусса



$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

Выделение границ - оператор Лапласа



-1	-1	-1
-1	8	-1
-1	-1	-1



Отступы (padding)

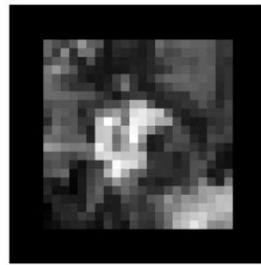
При свертке проблематично считать значения на краях изображения.

Какие варианты?

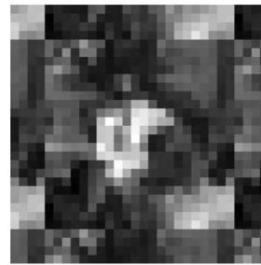
Отступы (padding)

При свертке проблематично считать значения на краях изображения.

Какие варианты?



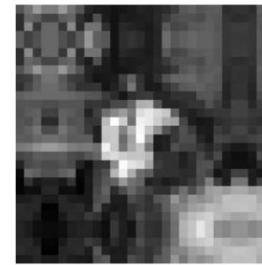
zero



wrap



clamp

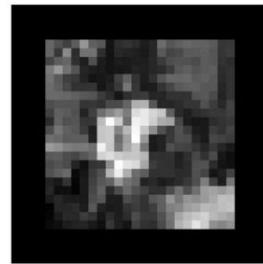


mirror

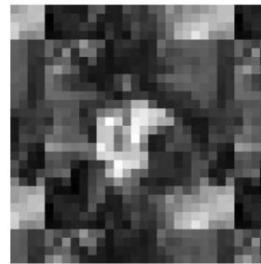
Отступы (padding)

При свертке проблематично считать значения на краях изображения.

Какие варианты?



zero



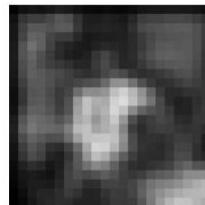
wrap



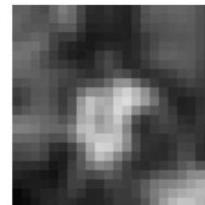
clamp



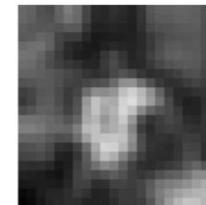
mirror



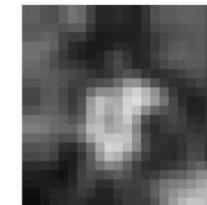
blurred zero



normalized zero



blurred clamp



blurred mirror

Сепарабельные фильтры

- сложность работы фильтра - K^2 на каждый пиксель изображения (K - размер ядра)
- сложность можно сократить до $2K$, в случае, если ядро фильтра можно представить в виде разложения на вертикальную и горизонтальную составляющие: $K=v*h^T$

$$\begin{matrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{matrix} = ?$$

Сепарабельные фильтры

- сложность работы фильтра - K^2 на каждый пиксель изображения (K - размер ядра)
- сложность можно сократить до $2K$, в случае, если ядро фильтра можно представить в виде разложения на вертикальную и горизонтальную составляющие: $K=v*h^T$

$$\begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array} = \begin{array}{|c|} \hline 1 \\ \hline 2 \\ \hline 1 \\ \hline \end{array} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline \end{array}$$

Сепарабельные фильтры

$$\begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array} * \begin{array}{|c|c|c|} \hline 2 & 3 & 3 \\ \hline 3 & 5 & 5 \\ \hline 4 & 4 & 6 \\ \hline \end{array} = \begin{array}{l} 2 + 6 + 3 = 11 \\ 6 + 20 + 10 = 36 \\ 4 + 8 + 6 = 18 \\ \hline 65 \end{array}$$

$$\begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array} = \begin{array}{|c|} \hline 1 \\ \hline 2 \\ \hline 1 \\ \hline \end{array} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline \end{array}$$

$$\begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline \end{array} * \begin{array}{|c|c|c|} \hline 2 & 3 & 3 \\ \hline 3 & 5 & 5 \\ \hline 4 & 4 & 6 \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 11 & & \\ \hline & 18 & \\ \hline & 18 & \\ \hline \end{array}$$
$$\begin{array}{|c|c|c|} \hline 1 & & \\ \hline 2 & & \\ \hline 1 & & \\ \hline \end{array} * \begin{array}{|c|c|c|} \hline 11 & & \\ \hline & 18 & \\ \hline & 18 & \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline & & \\ \hline & & \\ \hline & & \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline & & \\ \hline & & \\ \hline & & \\ \hline \end{array} = 65$$

Сепарабельные фильтры

$$\begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array} * \begin{array}{|c|c|c|} \hline 2 & 3 & 3 \\ \hline 3 & 5 & 5 \\ \hline 4 & 4 & 6 \\ \hline \end{array} = \begin{array}{l} 2 + 6 + 3 = 11 \\ 6 + 20 + 10 = 36 \\ 4 + 8 + 6 = 18 \\ \hline 65 \end{array}$$

$$\begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array} = \begin{array}{|c|} \hline 1 \\ \hline 2 \\ \hline 1 \\ \hline \end{array} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline \end{array}$$

$$\begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline \end{array} * \begin{array}{|c|c|c|} \hline 2 & 3 & 3 \\ \hline 3 & 5 & 5 \\ \hline 4 & 4 & 6 \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 11 & & \\ \hline & 18 & \\ \hline & 18 & \\ \hline \end{array}$$
$$\begin{array}{|c|c|c|} \hline 1 & & \\ \hline 2 & & \\ \hline 1 & & \\ \hline \end{array} * \begin{array}{|c|c|c|} \hline 11 & & \\ \hline & 18 & \\ \hline & 18 & \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline & & \\ \hline & & \\ \hline & & \\ \hline \end{array}$$
$$= 65$$

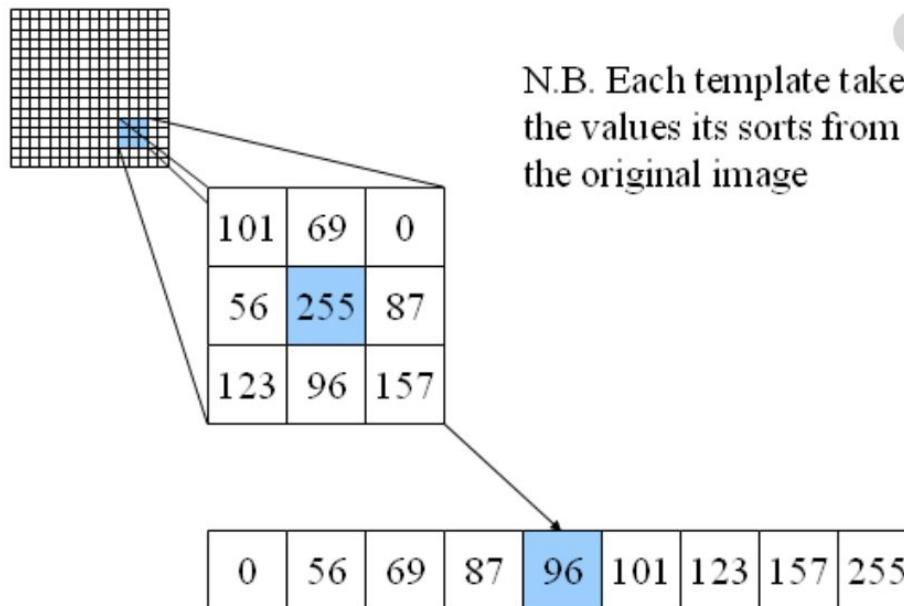
Сепарабельные фильтры

`cv2.sepFilter2D(src, ddepth, kernelX, kernelY)`

- **src** - исходное изображение
- **ddepth** - тип данных выходного изображения
- **kernelX** - горизонтальная компонента ядра
- **kernelY** - вертикальная компонента ядра

Нелинейные фильтры

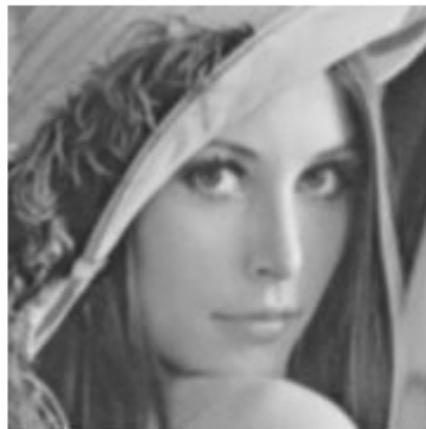
Медианный фильтр



Медианный фильтр



Билатеральный фильтр



Gaussian filter



Bilateral filter

Билатеральный фильтр

$$g(i, j) = \frac{\sum_{k,l} f(k, l)w(i, j, k, l)}{\sum_{k,l} w(i, j, k, l)}$$

$g(i,j)$ - результат применения фильтра в точке i,j

$f(k,l)$ - значение исходного изображения в точке k,l

$w(i,j,k,l)$ - вес, с которым учитывается значение $f(k,l)$

Билатеральный фильтр

$$d(i, j, k, l) = \exp\left(-\frac{(i - k)^2 + (j - l)^2}{2\sigma_d^2}\right)$$

$$r(i, j, k, l) = \exp\left(-\frac{\|f(i, j) - f(k, l)\|^2}{2\sigma_r^2}\right)$$

d - фактор расстояния от точки i,j до k,l

r - "похожесть" значений исходного изображения в точке i,j и k,l

sigma - параметры фильтра

Билатеральный фильтр

$$g(i, j) = \frac{\sum_{k,l} f(k, l) w(i, j, k, l)}{\sum_{k,l} w(i, j, k, l)}$$

$$w(i, j, k, l) = \exp \left(-\frac{(i - k)^2 + (j - l)^2}{2\sigma_d^2} - \frac{\|f(i, j) - f(k, l)\|^2}{2\sigma_r^2} \right)$$

w - учитывает как расстояние между точками (i,j) и (k,l), так и "похожесть" значений исходного изображения в точках (i,j) и (k,l)

Билатеральный фильтр

$$g(i, j) = \frac{\sum_{k,l} f(k, l) w(i, j, k, l)}{\sum_{k,l} w(i, j, k, l)}$$

$$w(i, j, k, l) = \exp \left(-\frac{(i - k)^2 + (j - l)^2}{2\sigma_d^2} - \frac{\|f(i, j) - f(k, l)\|^2}{2\sigma_r^2} \right)$$

w - учитывает как расстояние между точками (i,j) и (k,l), так и "похожесть" значений исходного изображения в точках (i,j) и (k,l)

Бинаризация изображений

Camping can be Fun and Adventurous in Our Sturdy Tents!

Camping is an exciting and adventurous activity for the entire family. Exploring the scenic views from your vacation, or exploring the sun and fun of the season, tent camping offers the ultimate combination with a lighter, more adjustable tent than ever before. All tents come with a variety of options and a range of accessories. There are numerous types of options and a range of accessories. There are numerous types of options and a range of accessories. There are numerous types of options and a range of accessories. There are numerous types of options and a range of accessories. There are numerous types of options and a range of accessories.

Select a model that meets your family's needs:

Our Tents are:

- Durable
- Weatherproof
- Made of the best fabrics
- Customized for 8' tents

	Small	Medium	Large
• Tent Type	Small	Medium	Large
• Durable	Yes	Yes	Yes
• Weatherproof	Yes	Yes	Yes
• Made of the best fabrics	Yes	Yes	Yes
• Customized for 8' tents	Yes	Yes	Yes

Camping can be Fun and Adventurous In Our Sturdy Tents!

Camping is an exciting and adventurous activity for the entire family. Exploring the scenic views from your vacation, or exploring the sun and fun of the season, tent camping offers the ultimate combination with a lighter, more adjustable tent than ever before. All tents come with a variety of options and a range of accessories. There are numerous types of options and a range of accessories. There are numerous types of options and a range of accessories. There are numerous types of options and a range of accessories. There are numerous types of options and a range of accessories.

Our Tents are:

- Durable
- Weatherproof
- Made of the best fabrics
- Customized for 8' tents

	Small	Medium	Large
• Tent Type	Small	Medium	Large
• Durable	Yes	Yes	Yes
• Weatherproof	Yes	Yes	Yes
• Made of the best fabrics	Yes	Yes	Yes
• Customized for 8' tents	Yes	Yes	Yes

Camping can be Fun and Adventurous In Our Sturdy Tents!

Camping is an exciting and adventurous activity for the entire family. Exploring the scenic views from your vacation, or exploring the sun and fun of the season, tent camping offers the ultimate combination with a lighter, more adjustable tent than ever before. All tents come with a variety of options and a range of accessories. There are numerous types of options and a range of accessories. There are numerous types of options and a range of accessories. There are numerous types of options and a range of accessories.

Our Tents are:

- Durable
- Weatherproof
- Made of the best fabrics
- Customized for 8' tents

	Small	Medium	Large
• Tent Type	Small	Medium	Large
• Durable	Yes	Yes	Yes
• Weatherproof	Yes	Yes	Yes
• Made of the best fabrics	Yes	Yes	Yes
• Customized for 8' tents	Yes	Yes	Yes

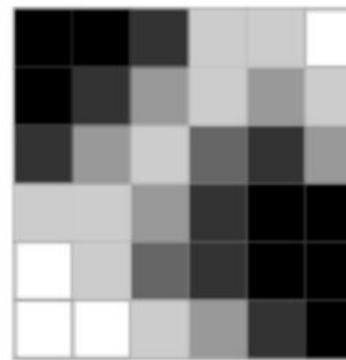
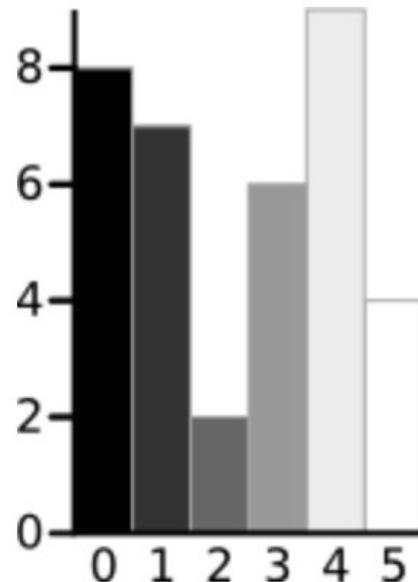
And remember, your investment in our products will be an investment in your family's happiness.

Бинаризация

$$\theta(f, t) = \begin{cases} 1 & \text{if } f \geq t, \\ 0 & \text{else,} \end{cases}$$

theta - оператор бинаризации
f - исходное изображение
t - значение порога

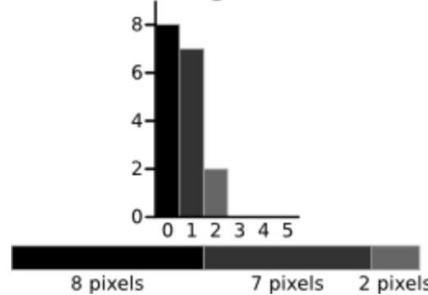
Otsu thresholding



Источник: <http://www.labbookpages.co.uk/software/imgProc/otsuThreshold.html>

Otsu thresholding

Background

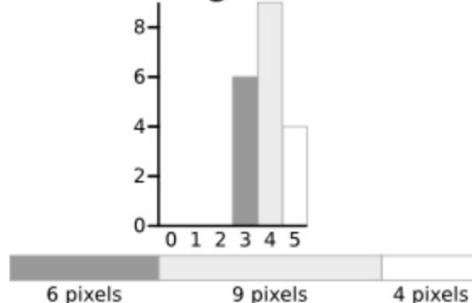


$$\text{Weight } W_b = \frac{8 + 7 + 2}{36} = 0.4722$$

$$\text{Mean } \mu_b = \frac{(0 \times 8) + (1 \times 7) + (2 \times 2)}{17} = 0.6471$$

$$\begin{aligned}\text{Variance } \sigma_b^2 &= \frac{((0 - 0.6471)^2 \times 8) + ((1 - 0.6471)^2 \times 7) + ((2 - 0.6471)^2 \times 2)}{17} \\ &= \frac{(0.4187 \times 8) + (0.1246 \times 7) + (1.8304 \times 2)}{17} \\ &= 0.4637\end{aligned}$$

Foreground



$$\text{Weight } W_f = \frac{6 + 9 + 4}{36} = 0.5278$$

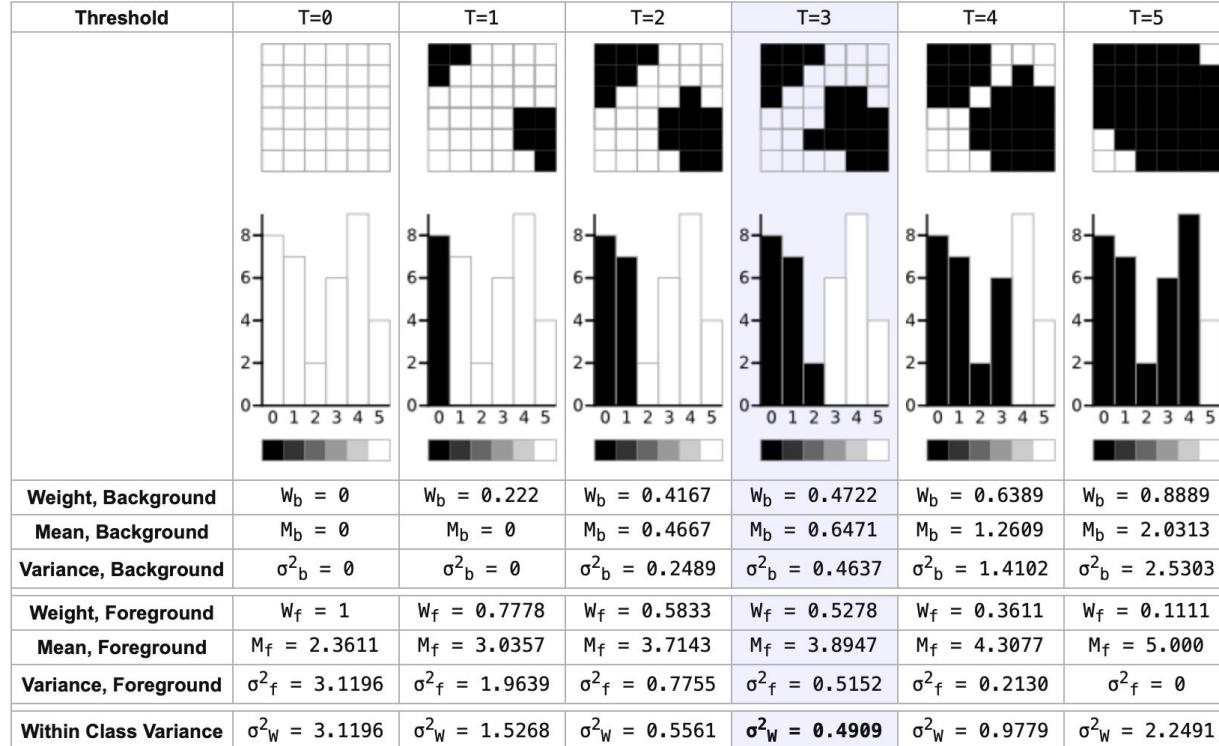
$$\text{Mean } \mu_f = \frac{(3 \times 6) + (4 \times 9) + (5 \times 4)}{19} = 3.8947$$

$$\begin{aligned}\text{Variance } \sigma_f^2 &= \frac{((3 - 3.8947)^2 \times 6) + ((4 - 3.8947)^2 \times 9) + ((5 - 3.8947)^2 \times 4)}{19} \\ &= \frac{(4.8033 \times 6) + (0.0997 \times 9) + (4.8864 \times 4)}{19} \\ &= 0.5152\end{aligned}$$

$$\begin{aligned}\text{Within Class Variance } \sigma_W^2 &= W_b \sigma_b^2 + W_f \sigma_f^2 = 0.4722 * 0.4637 + 0.5278 * 0.5152 \\ &= 0.4909\end{aligned}$$

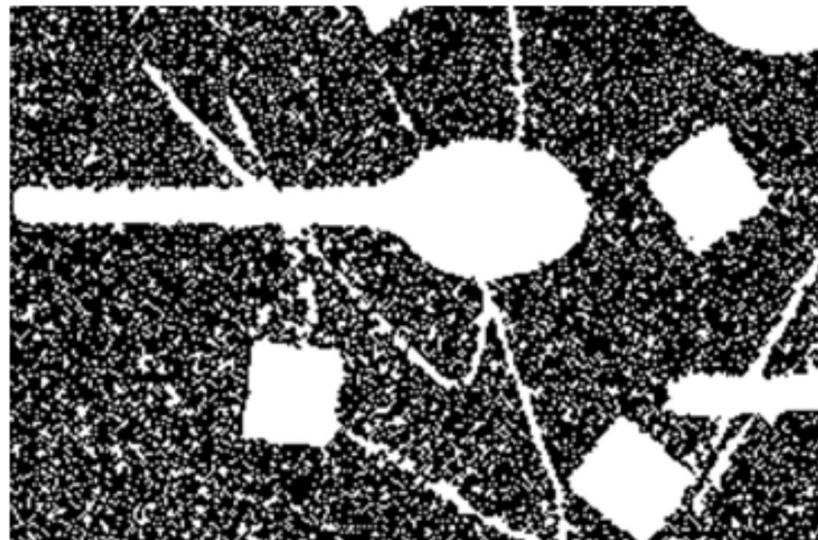
Источник: <http://www.labbookpages.co.uk/software/imgProc/otsuThreshold.html>

Otsu thresholding

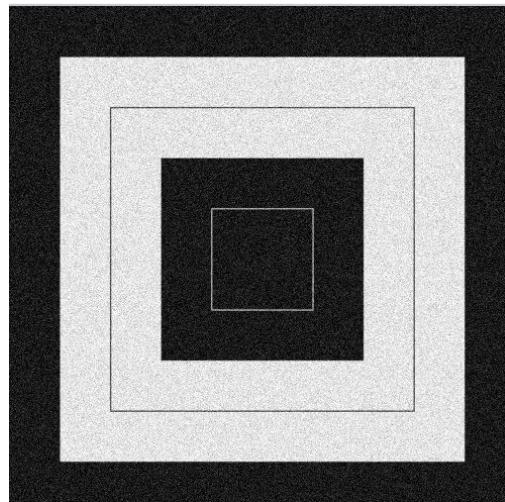


Шум в бинарных изображениях

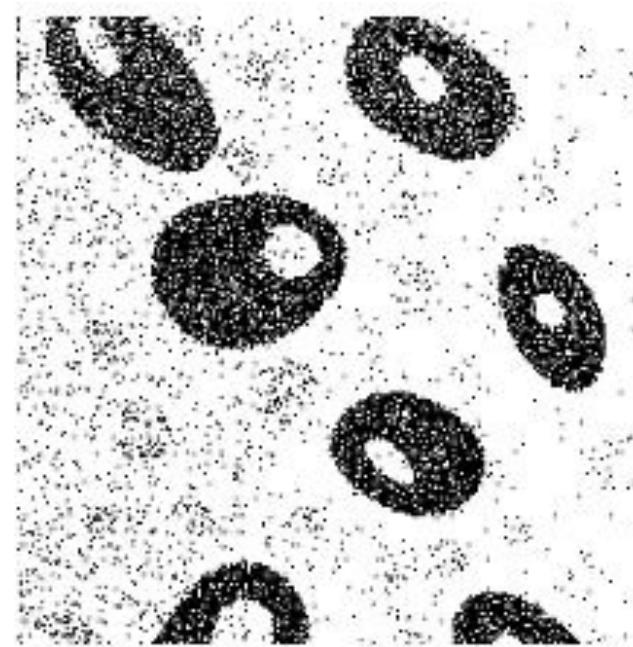
Часто возникает из-за невозможности полностью подавить шум в изображениях, недостаточной контрастности объектов и т.д.



Типы шумов



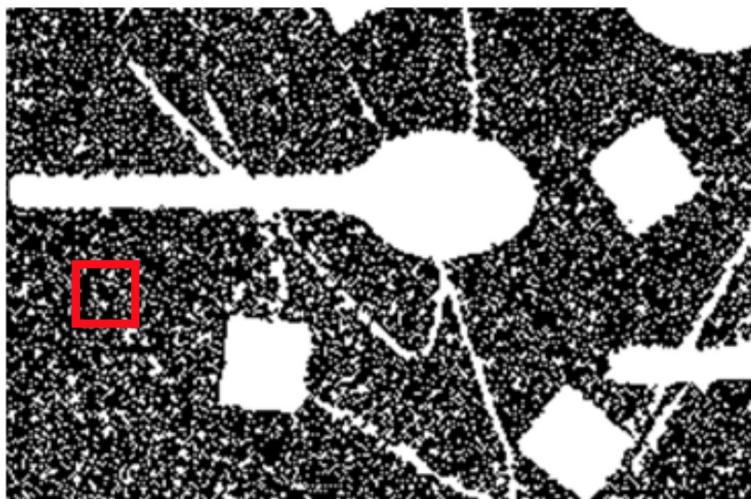
Гауссов шум



Шум “Соль и перец”

Шум в бинарных изображениях

- По одному пикслю невозможно определить – шум или объект?
- Нужно рассматривать окрестность пикселя!



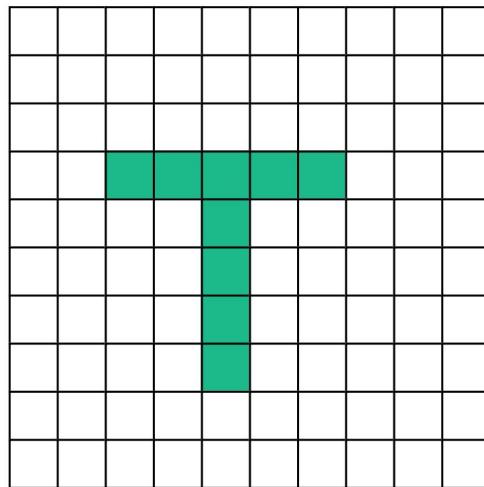
Подавление и устранение шума

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

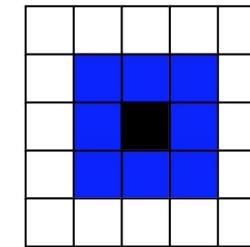
- Сужение (erosion)
- Расширение (dilation)
- Закрытие (closing)
- Раскрытие (opening)

Математическая морфология

A



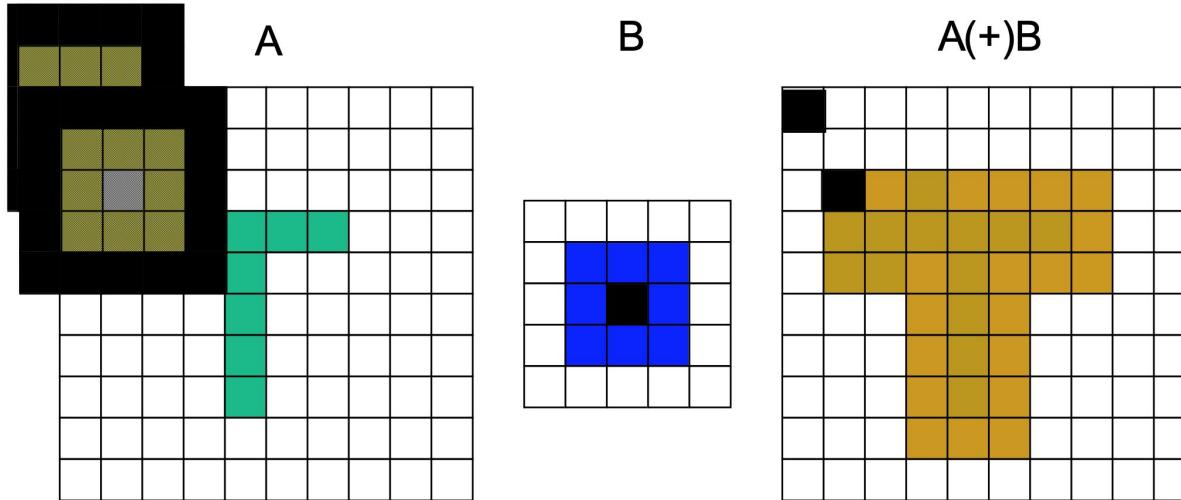
B



Множество А обычно является объектом обработки

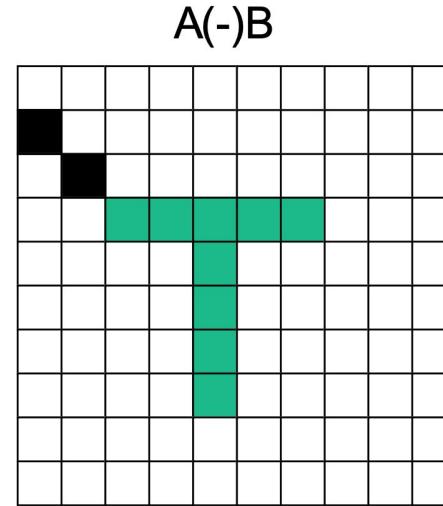
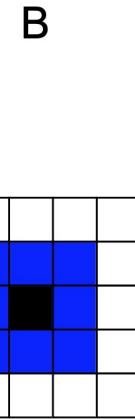
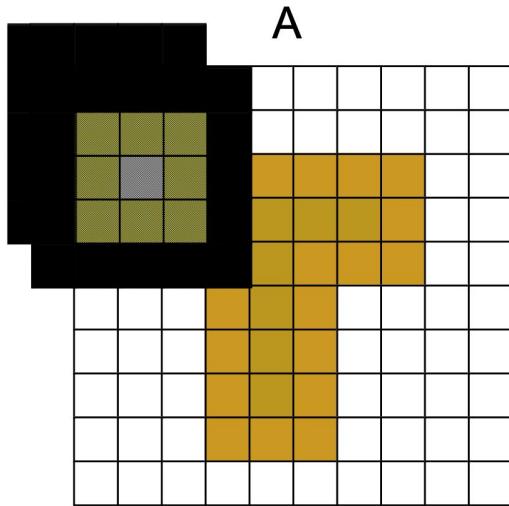
Множество В (называемое структурным элементом) – инструмент обработки

Расширение



Операция «расширение» - аналог логического «или»

Сужение



Операция «расширение» - аналог логического «и»

Результат операции сужения



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



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



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

Замечание

Результат морфологических операций во многом определяется применяемым структурным элементом. Выбирая различный структурный элемент можно решать разные задачи обработки изображений:

- Шумоподавление
- Выделение границ объекта
- Выделение особенностей на медицинских снимках

Операции раскрытия и закрытия

Морфологическое раскрытие (opening) - вначале сужение, потом расширение

- $\text{open}(A,B)=(A \ (-) \ B) \ (+) \ B$

Морфологическое закрытие (closing) - вначале расширение, потом сужение

- $\text{close}(A,B)=(A \ (+) \ B) \ (-) \ B$

Попробуйте догадаться, что эти операции делают?

Применение открытия

Применим операцию открытия к изображению с сильным шумом:



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



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



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

Сужение vs Открытие



Сужение



Открытие

Дефекты бинаризации

Пример бинарного изображению с дефектами распознаваемых объектов



Применение закрытия

Применим операцию закрытия к изображению с дефектами объектов:



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



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



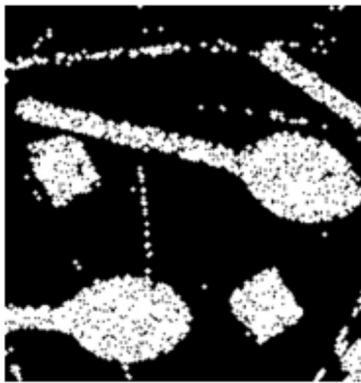
$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Не лучший пример для морфологии

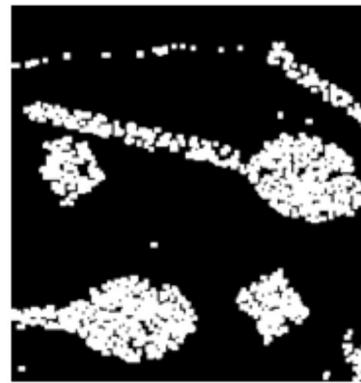
Не во всех случаях математическая морфология так легко убирает дефекты, как хотелось бы...



Применение операции открытия



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



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



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

Часто помогает медианная фильтрация!

Let's code!