

Цифровая обработка изображений

Базовая теория

Задачи

- обработка изображений с помощью фильтров
- детектирование границ
- удаление шума

Применение фильтров для обработки изображений

noisy lena



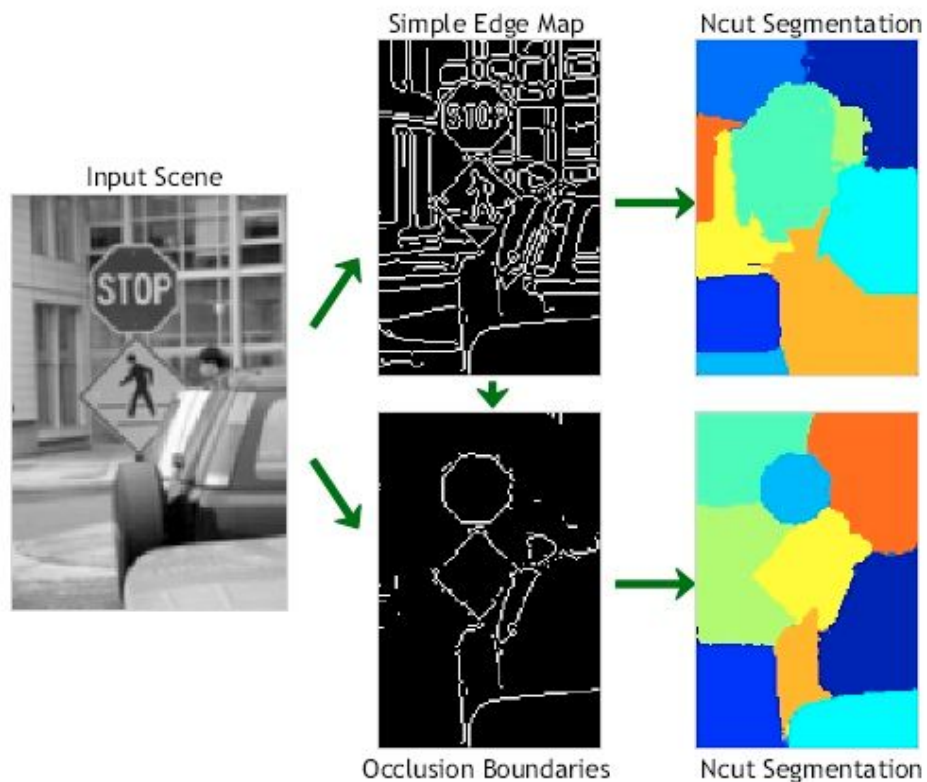
Gaussian filter



Удаление шума на изображении



Детектор границ объектов



Содержание

Цифровое представление изображений

- растр vs вектор

- разрешение

- пространство цветов

Растровые форматы, сжатие

Применение фильтров для обработки изображений

1.1 Цифровое представление изображений

Растр vs Вектор

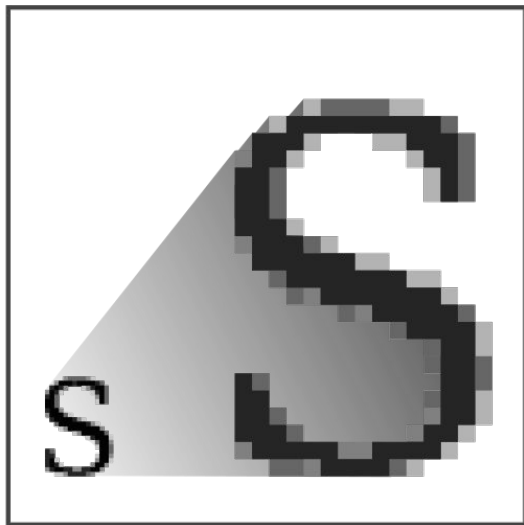
Растр

описание изображения на уровне точек (пикселей)
размер изображения ограничен числом пикселей

Вектор

описание изображения на уровне фигур и их свойств
размер изображения может быть произвольным

Растр vs Вектор

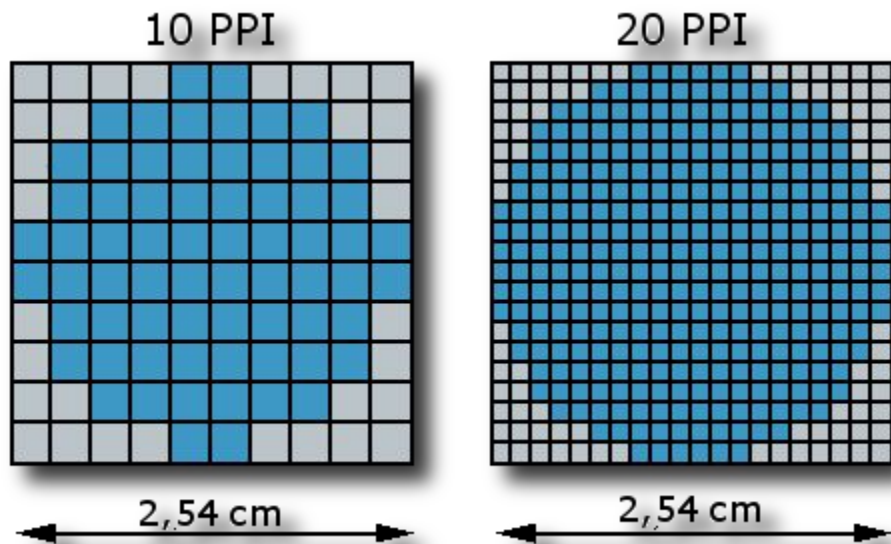


РАСТР
.jpeg .gif .png



ВЕКТОР
.svg

Разрешение



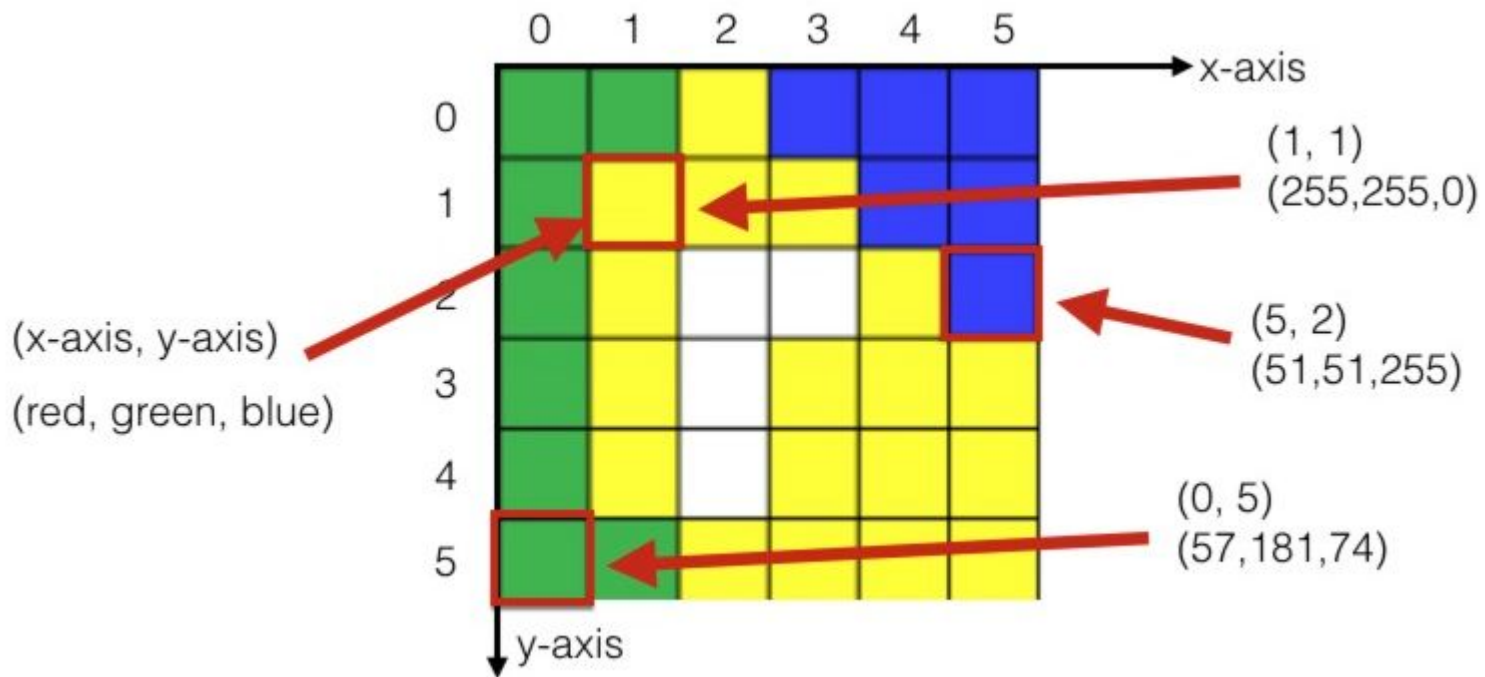
Каналы и динамический диапазон

- каждый пиксель изображения кодируется одним или несколькими значениями (каналами)
- стандартный диапазон значений в каждом канале: 0..255 (один байт или 8 бит)

Каналы и динамический диапазон

- для представления черно-белого изображения достаточно одного канала (передача яркости пикселя)
- цветные изображения, как правило, содержат 3 канала

Растровое представление изображения

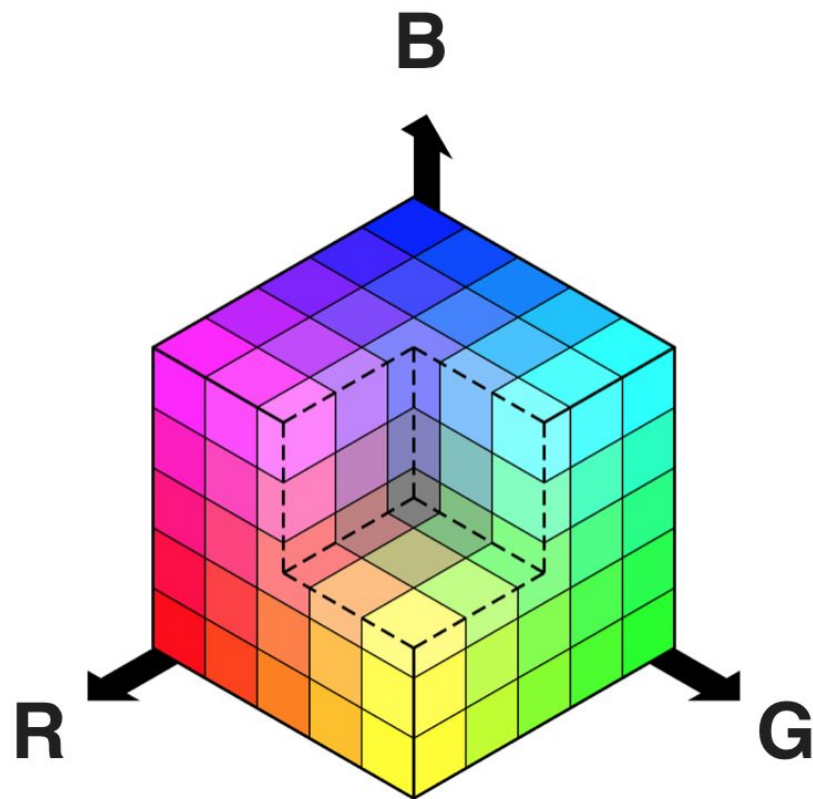


1.2 Пространства цветов

RGB

- RGB - **R**ed **G**reen **B**lue
 - наиболее распространенное представление цветного изображения
 - выбор основных цветов обусловлен восприятием цвета сетчатки глаза
 - 3 канала

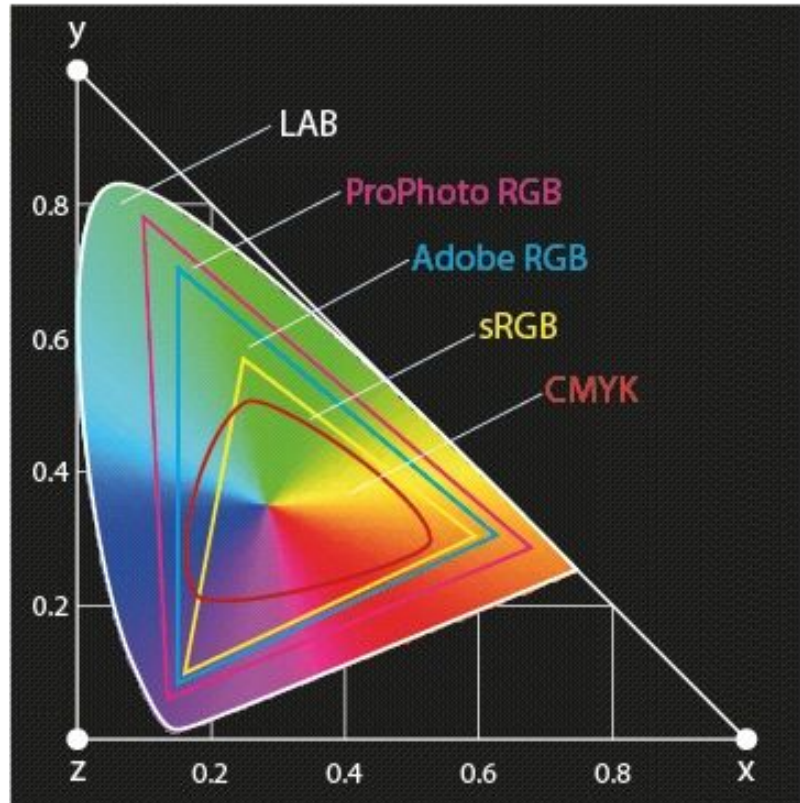
RGB



CMYK

- CMYK - **C**yan **M**agenta **Y**ellow **B**lack
 - 4 канала
 - в основном используется в полиграфии
 - цветовой охват меньше в сравнении с RGB

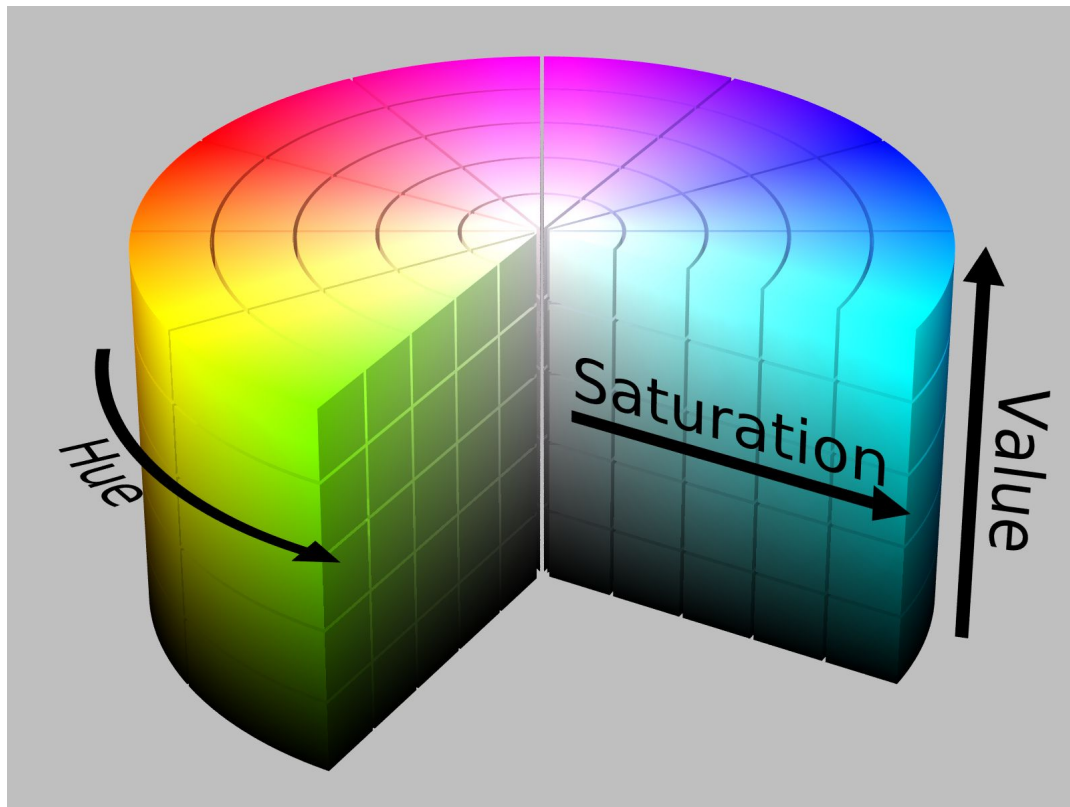
CMYK vs RGB



HSV

- HSV - **H**ue **S**aturation **V**alue
 - Hue - цветовой тон
 - Saturation - насыщенность цвета
 - Value - интенсивность

HSV



RGB -> HSV

$$V \leftarrow \max(R, G, B)$$

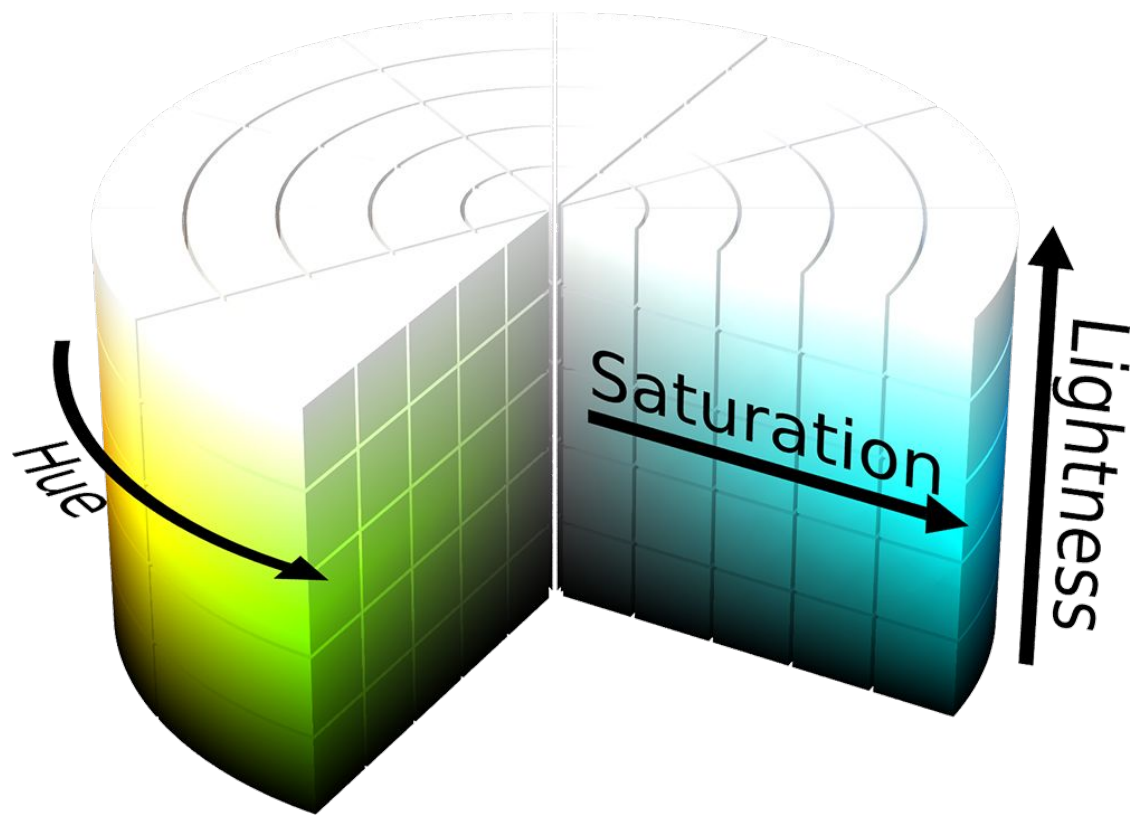
$$S \leftarrow \begin{cases} \frac{V - \min(R, G, B)}{V} & \text{if } V \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$H \leftarrow \begin{cases} 60(G - B)/(V - \min(R, G, B)) & \text{if } V = R \\ 120 + 60(B - R)/(V - \min(R, G, B)) & \text{if } V = G \\ 240 + 60(R - G)/(V - \min(R, G, B)) & \text{if } V = B \end{cases}$$

HSL

- HLS - **H**ue **S**aturation **L**ightness
 - Hue - цветовой тон
 - Saturation - насыщенность цвета
 - Lightness - яркость

HSL



RGB -> HSL

$$V_{max} \leftarrow \max(R, G, B)$$

$$V_{min} \leftarrow \min(R, G, B)$$

$$L \leftarrow \frac{V_{max} + V_{min}}{2}$$

$$S \leftarrow \begin{cases} \frac{V_{max} - V_{min}}{V_{max} + V_{min}} & \text{if } L < 0.5 \\ \frac{V_{max} - V_{min}}{2 - (V_{max} + V_{min})} & \text{if } L \geq 0.5 \end{cases}$$

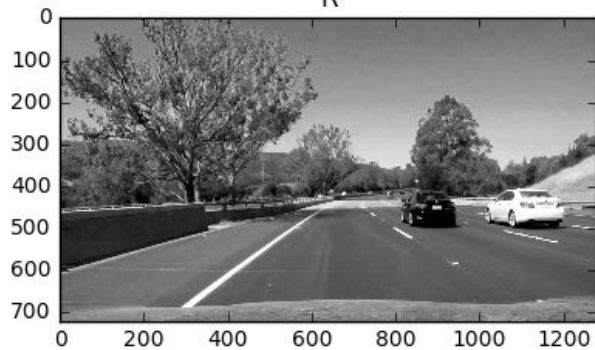
$$H \leftarrow \begin{cases} 60(G - B)/(V_{max} - V_{min}) & \text{if } V_{max} = R \\ 120 + 60(B - R)/(V_{max} - V_{min}) & \text{if } V_{max} = G \\ 240 + 60(R - G)/(V_{max} - V_{min}) & \text{if } V_{max} = B \end{cases}$$

HSL vs RGB

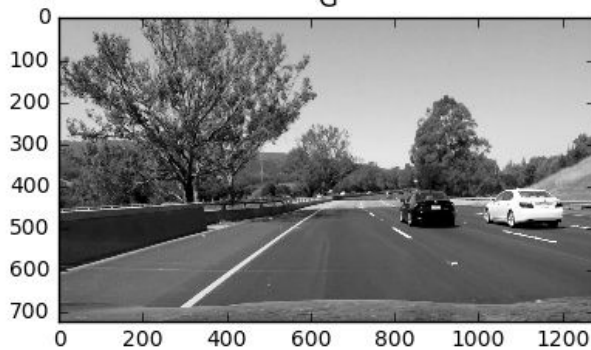


HSL vs RGB

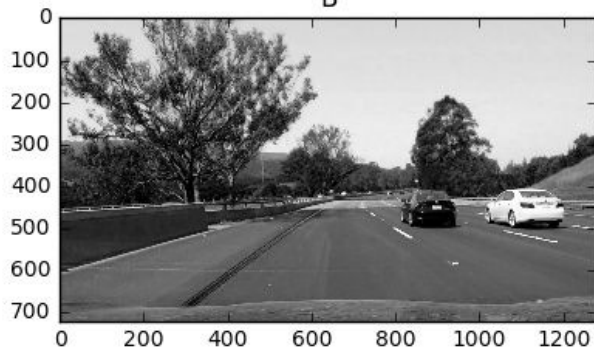
R



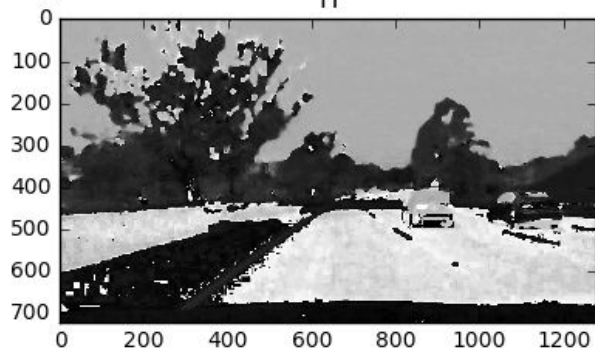
G



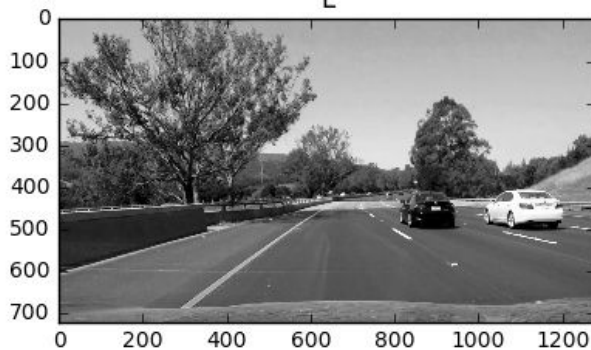
B



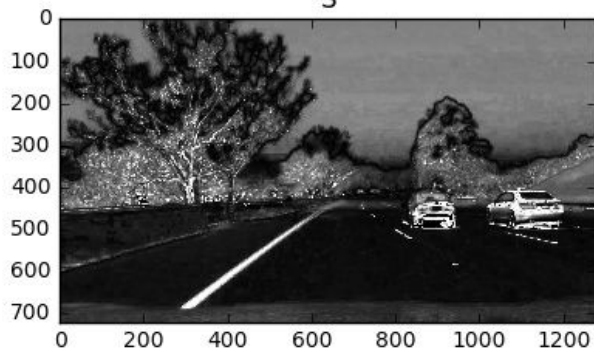
H



L



S



1.3 Растровые форматы. Сжатие

Сжатие изображений

- Матричное представление RGB требовательно к ресурсам памяти
- Каждый пиксель занимает $3 \times 8\text{bit} = 24\text{bit}$ памяти
- Изображение 1024×768 занимает 2,4Mb памяти

Сжатие изображений

- Представление изображения в виде матриц RGB, как правило, содержит избыточную информацию
- Сжатие изображений осуществляется за счет уменьшения объема избыточной информации

Алгоритмы сжатия изображений

- Сжатие с потерями
 - восстановленное после сжатия изображение может отличаться от исходного
- Сжатие без потерь
 - гарантируется что восстановленное после сжатия изображение совпадает с исходным

Форматы изображений

- JPEG [Joint Photographic Experts Group](#)
 - сжатие с потерями - восстановленное изображение не является точной копией исходного
 - уровень сжатия является параметром алгоритма
 - ориентировочный коэффициент сжатия цветного изображения: 10:1 - 20:1
 - использует особенность восприятия изображения человеческим глазом, связанную с большей чувствительностью к изменению яркости пикселей и меньшей чувствительностью к небольшому изменению цвета

Форматы изображений

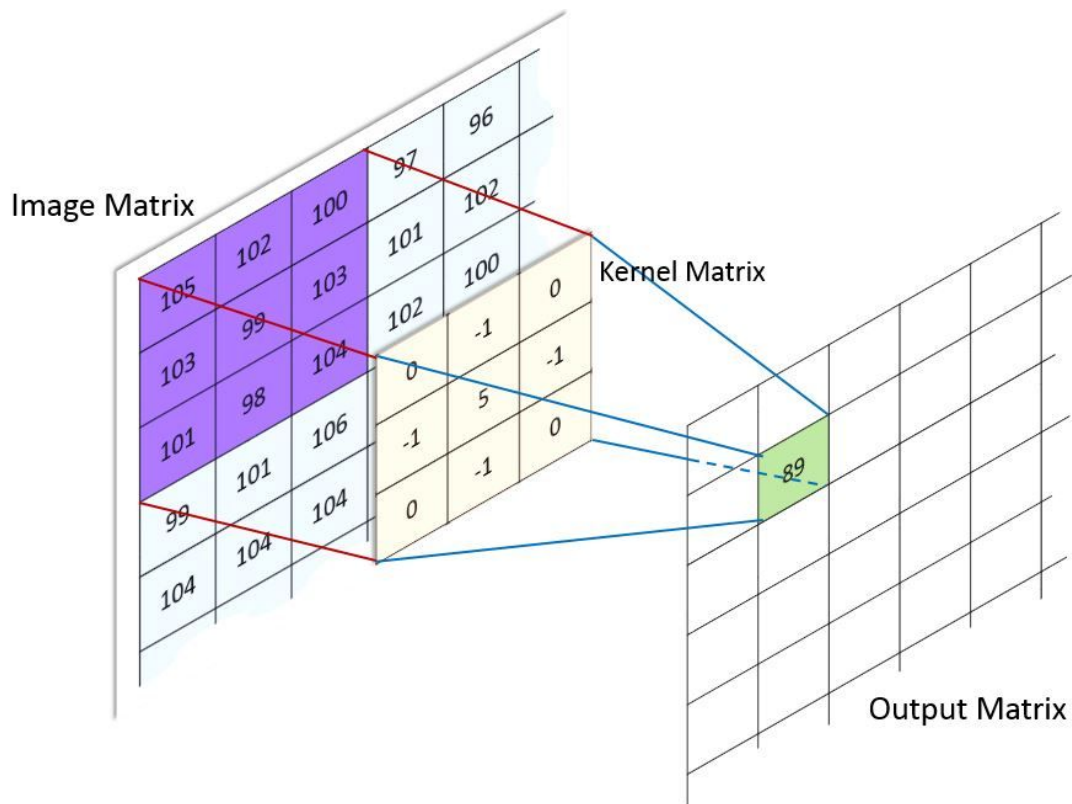
- PNG [Portable Network Graphic](#)
 - сжатие без потерь
 - палитра цветов изображения хранятся в таблице
 - для каждого пикселя указывается индекс цвета из палитры
 - ориентировочный коэффициент сжатия цветного изображения: 2,5:1

1.4 Применение фильтров для обработки изображений

Фильтры

- Свертка
- Размытие (blur)
- Выделение границ (sharpen)
- Удаление шума (denoise)

Свертка



Свертка

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

Свертка

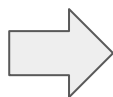
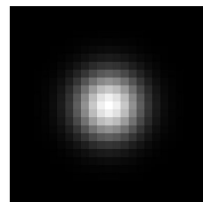
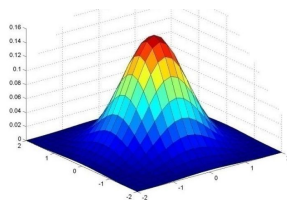
| | | |
|----|----|----|
| x1 | x0 | x1 |
| x0 | x1 | x0 |
| x1 | x0 | x1 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |



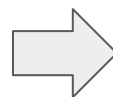
| | | | | |
|---|---|---|---|---|
| 2 | 2 | 3 | 1 | 1 |
| 1 | 4 | 3 | 4 | 1 |
| 2 | 2 | 4 | 3 | 3 |
| 1 | 2 | 3 | 4 | 1 |
| 1 | 2 | 3 | 1 | 1 |

Размытие - фильтр Гаусса

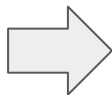
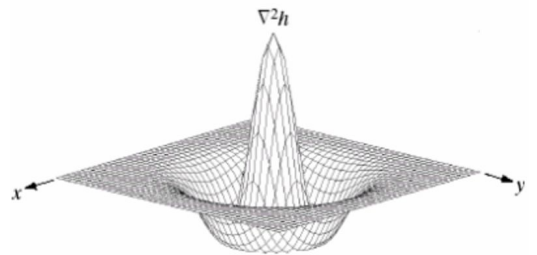


$$\frac{1}{273}$$

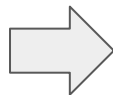
| | | | | |
|---|----|----|----|---|
| 1 | 4 | 7 | 4 | 1 |
| 4 | 16 | 26 | 16 | 4 |
| 7 | 26 | 41 | 26 | 7 |
| 4 | 16 | 26 | 16 | 4 |
| 1 | 4 | 7 | 4 | 1 |



Выделение границ - оператор Лапласа



| | | |
|----|----|----|
| -1 | -1 | -1 |
| -1 | 8 | -1 |
| -1 | -1 | -1 |



Медианный фильтр



Резюме

- изображения в цифровом виде представляются в виде матриц
- каждый пиксель матрицы цветного изображения содержит несколько значения
- значения пекселя зависят от пространства цветов RGB, HSV, HSL

Резюме

- Сжатие изображений может быть как с потерями в качестве, так и без потерь
- В случае сжатия с потерями размер цветного изображения может быть уменьшен в 10 - 20 раз

Резюме

- фильтрация изображений осуществляется с помощью сверточной операции
- фильтр Гаусса усредняет значение пикселя и размывает изображение
- оператор Лапласа вычисляет производную и выделяет грани на изображении
- медианный фильтр позволяет убрать шум на изображении

Полезные материалы

https://en.wikipedia.org/wiki/HSL_and_HSV

https://en.wikipedia.org/wiki/Image_file_formats

https://en.wikipedia.org/wiki/Digital_image_processing

Computer Vision: Algorithms and Applications <http://szeliski.org/Book/> (chapter 3)