

# Цифровая обработка изображений

## 2. Извлечение признаков и поиск

# Сегодня мы научимся

- решать задачу распознавания изображений методом понижения размерности
- выделять признаки для поиска и анализа изображений
- определять характерные точки на изображении
- строить систему поиска изображений

# План занятия

- Анализ главных компонент в задачах CV
- Представление свойств изображения с помощью гистограмм
  - Гистограммы цветов
  - Гистограммы градиентов

# План занятия

- Характерные точки
  - Поиск характерных областей на изображении
  - Выделение признаков (дескрипторов) характерных областей на изображении
  - Матчинг характерных точек на изображениях
  - Пример

# План занятия

- Поиск изображений по контенту CBIR
  - выделение признаков и индексация
  - обзор архитектуры
  - поиск по индексу

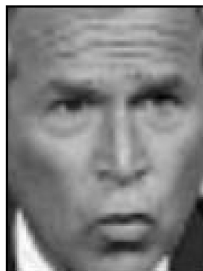
# Примеры задач компьютерного зрения

# Распознавание лиц

predicted: Powell  
true: Powell



predicted: Bush  
true: Bush



predicted: Chavez  
true: Chavez



predicted: Bush  
true: Bush



predicted: Rumsfeld  
true: Rumsfeld



predicted: Blair  
true: Schroeder



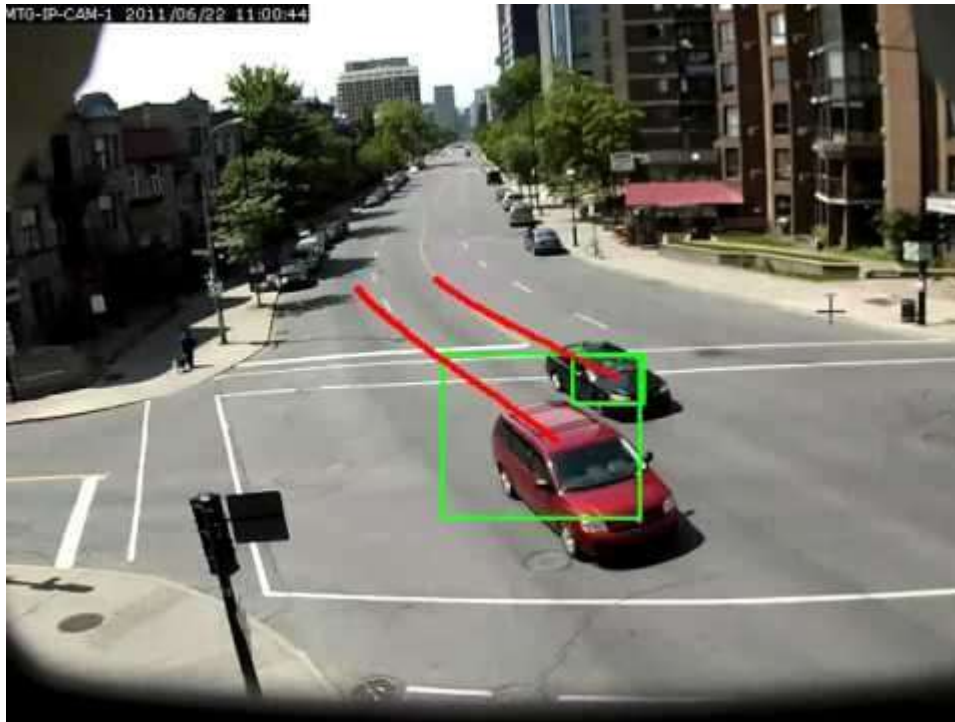
predicted: Sharon  
true: Sharon



predicted: Schroeder  
true: Schroeder



# Пример. Optical Flow





# Пример. Поиск похожих изображений (CBIR)

Query Image



Retrieved Results



РСА - анализ главных компонент (eigenface)

# РСА - анализ главных компонент

- изображение можно представить в виде вектора длины  $H \times W$
- большая размерность данных (число пикселей) затрудняет их обработку
- для сокращения размерности применяется метод РСА

# РСА - анализ главных компонент

- в результате преобразование РСА получаем представления изображений в базисе меньшей размерности
- полученное сжатое представление можно использовать для распознавания изображений

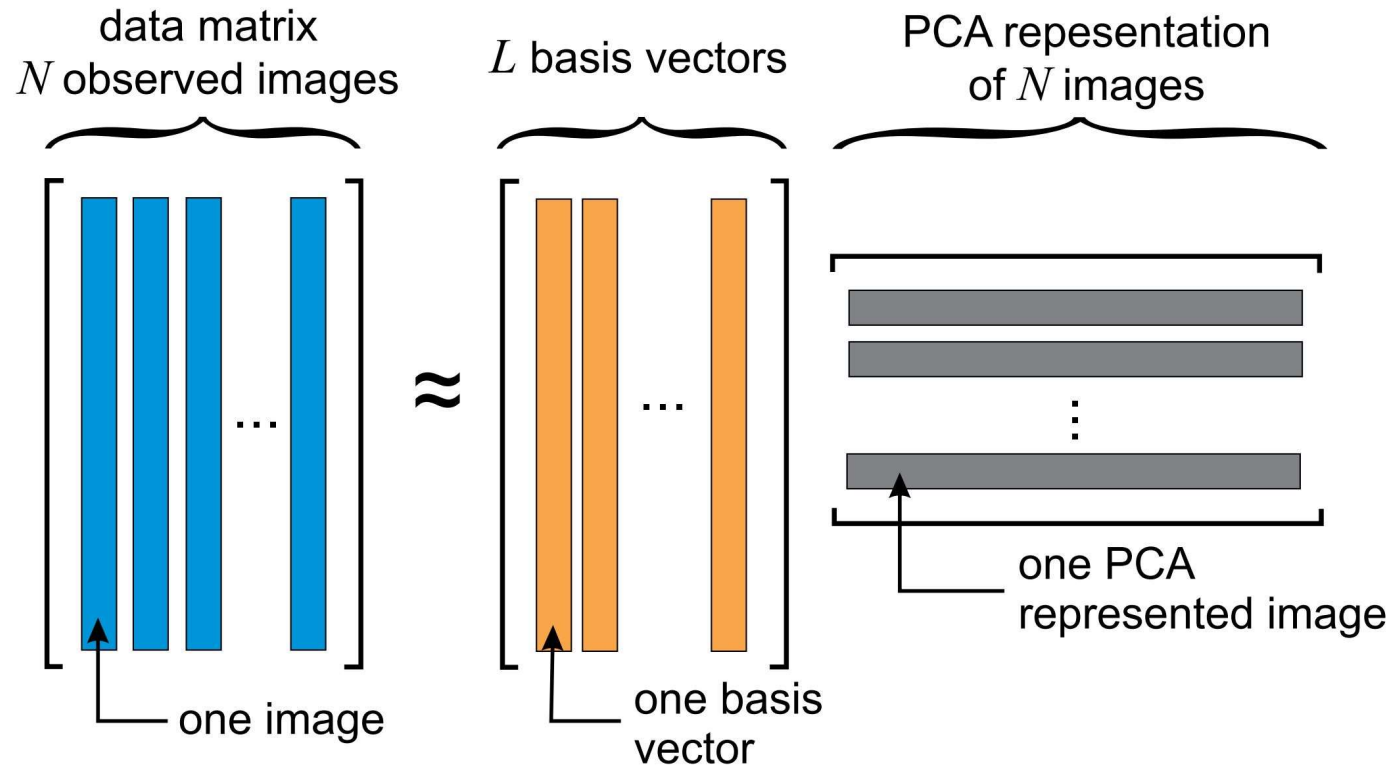
# РСА - анализ главных компонент



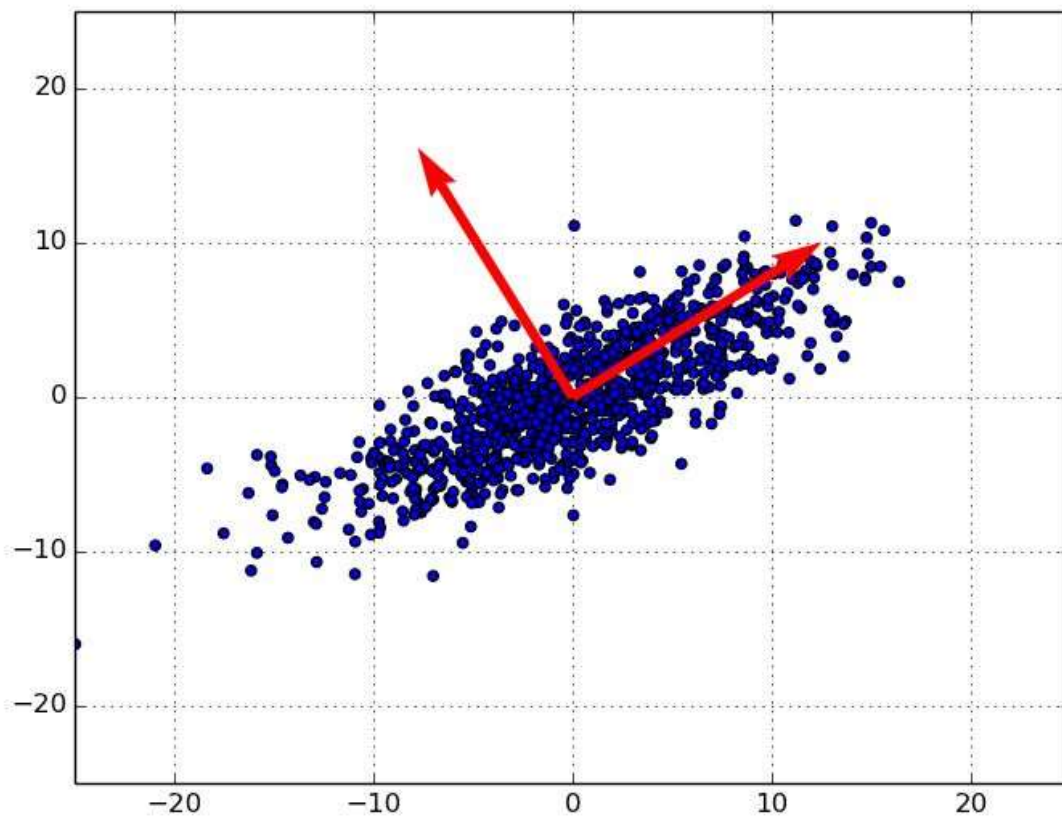
## РСА - анализ главных компонент



# РСА - анализ главных компонент



## РСА - анализ главных компонент



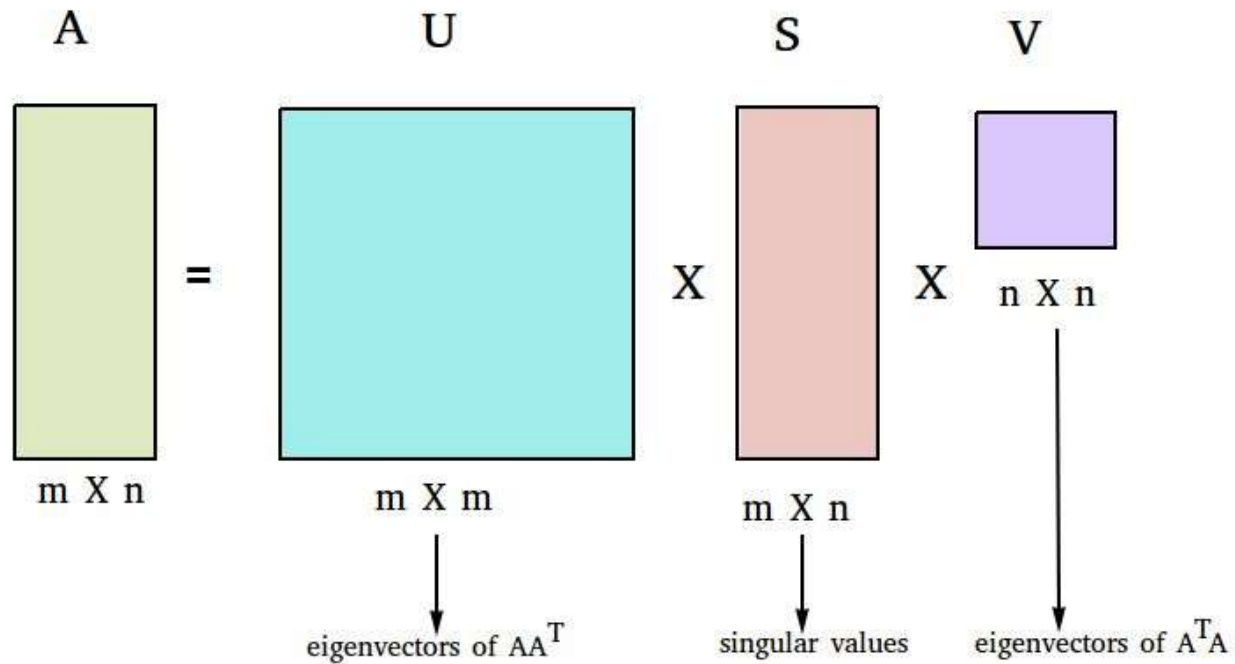


## РСА - матрица ковариации

$$\text{Cov}(A) = \begin{bmatrix} \frac{\sum (x_i - \bar{X})(x_i - \bar{X})}{N} & \frac{\sum (x_i - \bar{X})(y_i - \bar{Y})}{N} \\ \frac{\sum (x_i - \bar{X})(y_i - \bar{Y})}{N} & \frac{\sum (y_i - \bar{Y})(y_i - \bar{Y})}{N} \end{bmatrix}$$

$$= \begin{bmatrix} \text{Cov}(X, X) & \text{Cov}(Y, X) \\ \text{Cov}(X, Y) & \text{Cov}(X, Y) \end{bmatrix}$$

# PCA - разложение SVD



# РСА - последовательность вычислений

- подготавливаем данные, представляем изображения в виде векторов длиной  $H \times W$
- вычитаем среднее значение из каждой компоненты вектора
- получаем собственные вектора в результате SVD разложения ковариационной матрицы изображений
- выбираем размерность (число собственных векторов) на основе сингулярных значений

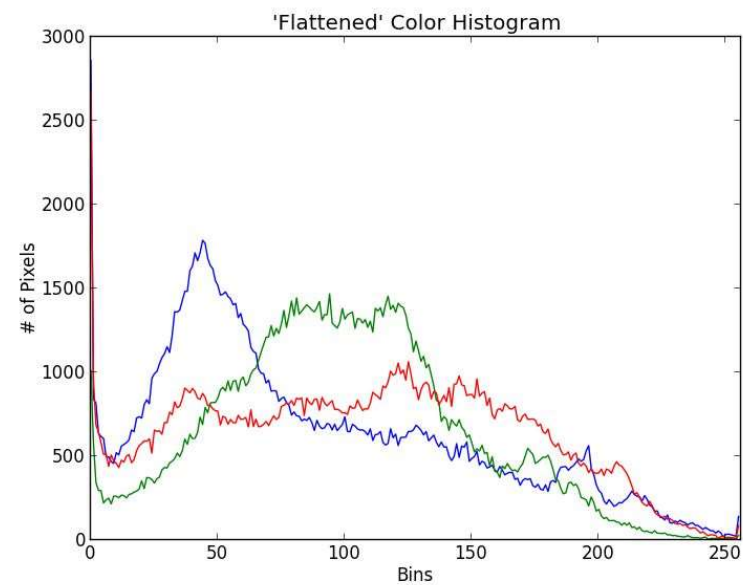
Выделение признаков изображения

# Гистограммы признаков изображения

- представляют собой обобщенное описание изображения
- как правило гистограммы инвариантны к масштабу и повороту изображений
- позволяют сравнивать изображения и находить похожие

# Гистограммы цветов

# Гистограммы цветов



# Гистограммы цветов

- разбиваем диапазон значений цвета (0..255) на фиксированное число ячеек (bins), например с шагом 1
- задаем в каких срезах (каналах) необходимо построить гистограмму
- для каждой ячейки считаем число соответствующих пикселей на изображении
- можно вычислять как в пространстве RGB, так и в других цветовых пространствах, например, HSV



# Гистограммы цветов

[cv2.calcHist](#)(images, channels, mask, histSize, ranges) → hist

**images** – набор входных изображений для оценки гистограммы

**channels** – каналы по которым оцениваются гистограммы

**mask** – маска ограничивает область оценки гистограммы

**histSize** – массив размеров гистограмм по каждому измерению

**ranges** – диапазоны значений каждого измерения

# Гистограммы цветов

- не зависят от изменения масштаба изображения
- устойчивы к повороту и перспективным искажениям
- в цветовых пространствах HSV и HSL менее чувствительны к изменению яркости

# Гистограммы градиентов (HOG)

# Гистограммы градиентов (HOG)

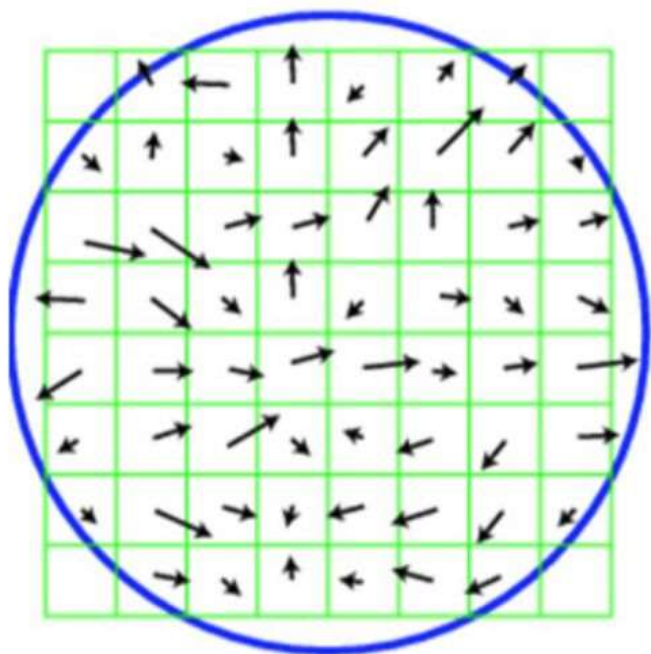
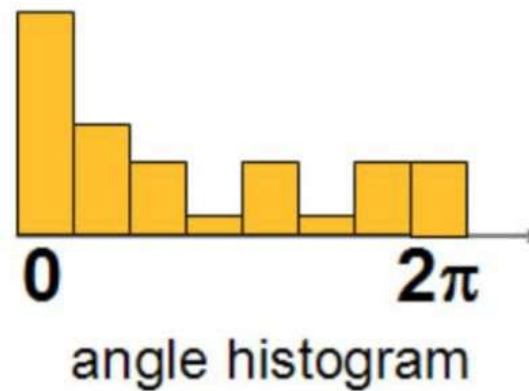


Image gradients



angle histogram



# Гистограммы градиентов (HOG)

- в каждой точке оцениваем составляющие градиента по осям  $x$  и  $y$
- определяем направление и длину вектора градиента
- оцениваем гистограмму градиентов
- полученные гистограммы нормализуют, таким образом, чтобы вектор признаков был единичной длины

# Оператор Собеля

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

x filter

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

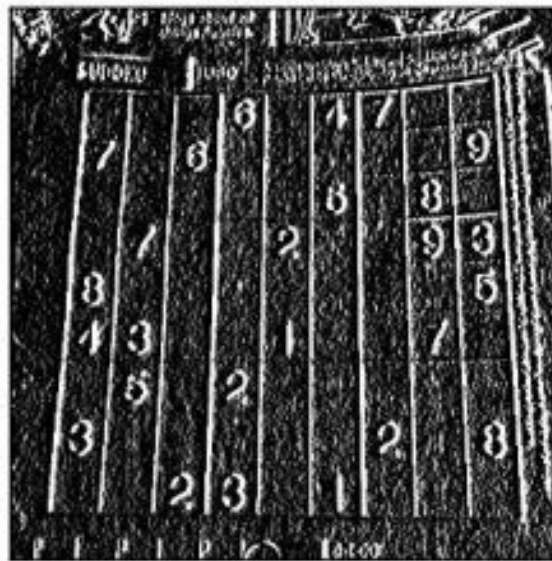
y filter

# Оператор Собеля

Original



Sobel X



Sobel Y



## Гистограммы градиентов (HOG)

$$g = \sqrt{g_x^2 + g_y^2}$$

$$\theta = \arctan \frac{g_y}{g_x}$$

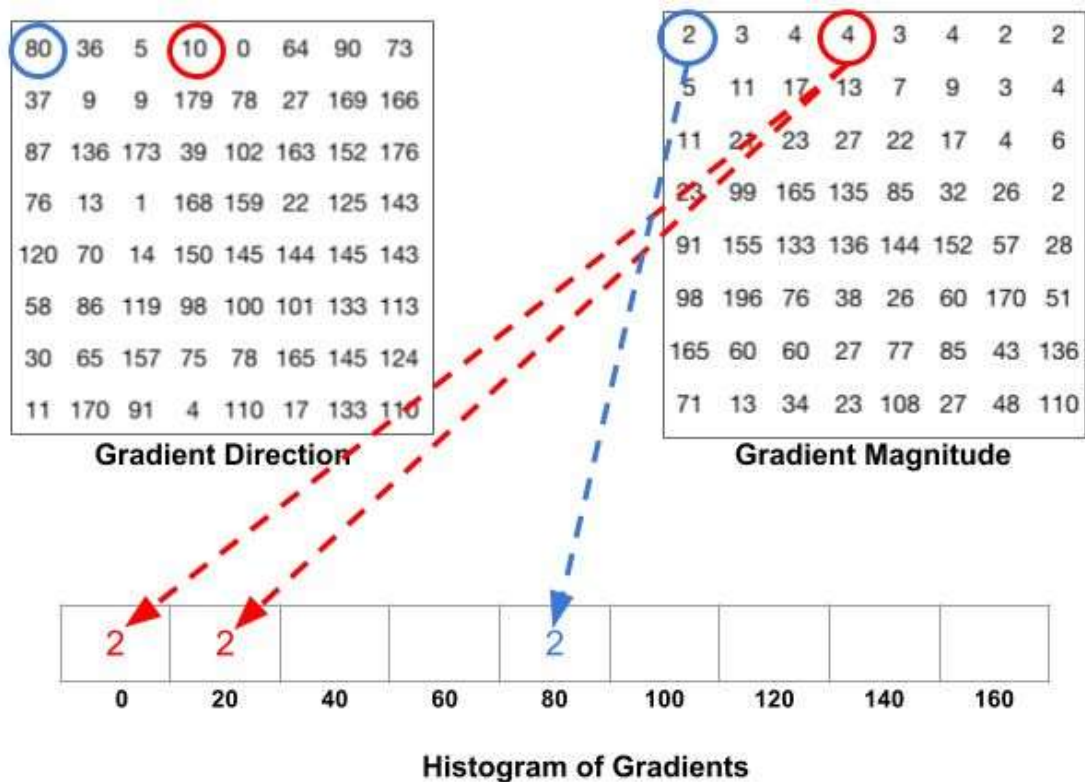
$g, g_x, g_y$  - длина вектора градиента и его составляющих  
 $\theta$  - угол наклона градиента в полярной системе координат



# Гистограммы градиентов (HOG)

- как правило гистограмму градиентов строят для диапазона углов  $0..180$
- при оценке гистограммы градиентов учитывается как угол, так и длина вектора
- чем больше длина вектора, тем больший вклад вносится в соответствующую ячейку гистограммы

# Гистограммы градиентов (HOG)



# Гистограммы градиентов (HOG)

Input image



Histogram of Oriented Gradients



# Гистограммы градиентов (HOG)

`cv2.Sobel`(src, ddepth, dx, dy[, dst[, ksize]]) → dst

**src** – входное изображение

**ddepth** - тип данных для вычисления производной, например, `cv2.CV_64F`

**dx/dy** - порядок производной по осям, как правило 0 или 1

**dst** - выходное изображение

**ksize** – размер ядра фильтра 1, 3, 5, или 7

`cv2.cartToPolar`(x, y) → magnitude, angle

**x,y** – вектора с координатами x и y

**magnitude** - длины векторов

**angle** - соответствующие углы

# Гистограммы градиентов (HOG)

- не чувствительны к изменению цвета
- устойчивы к изменению яркости
- устойчивы к изменению масштаба