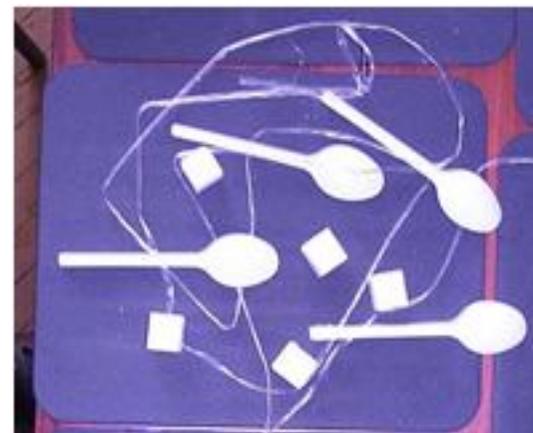
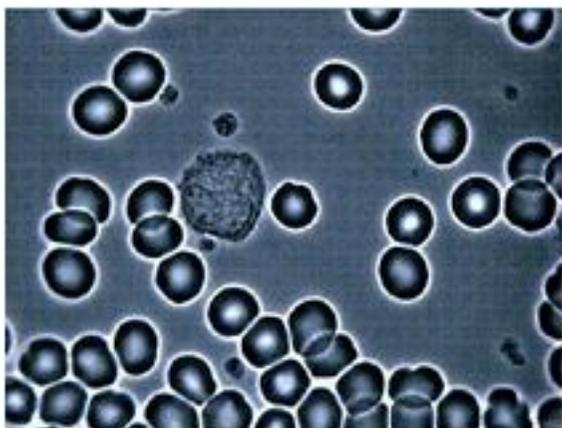




Простой анализ изображений



АНТОН КОНУШИН

Many slides adapted from Fei-Fei Li, Rob Fergus, Antonio Torralba, Jean Ponce and Svetlana Lazebnik



Выделение объектов



Необходимо определить, есть ли на изображении объекты заданного типа и если да, то определить их положение



Изменчивость изображений



Внешние факторы:

- Положение камеры
- Внутренние свойства камеры
- Освещение

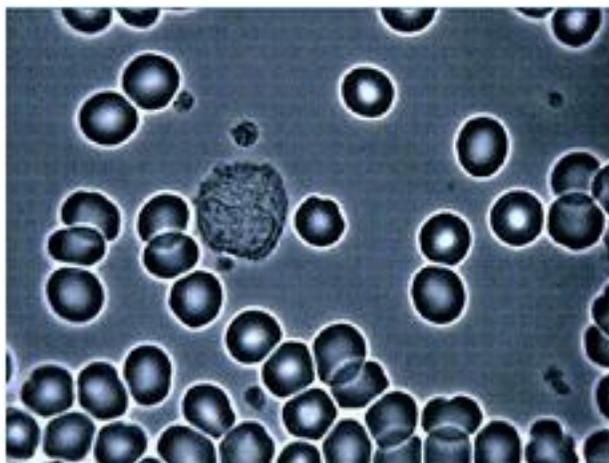
Внутренние факторы:

- Деформации объектов
- Внутриклассовая изменчивость

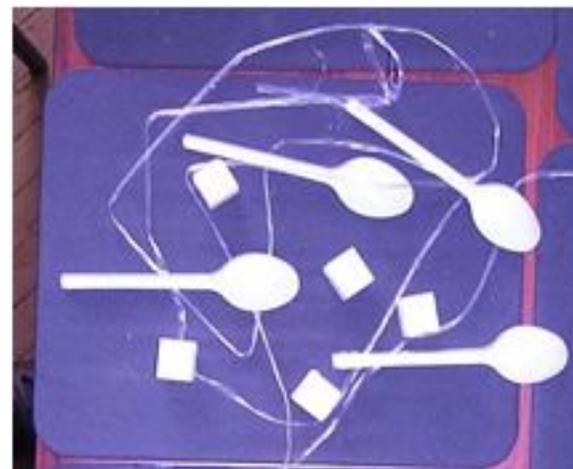
- Пока приходится задачу упрощать, вводя ограничения на ракурс съёмки, условия освещения, типы объектов
- Мы будем рассматривать простые случаи, когда все факторы варьируются незначительно



Примеры простых задач



Клетки крови



Ложки и сахар



Монеты и купюры



Номера



Какие параметры меняются мало?



Монеты и купюры



Номера

Внешние факторы:

- Положение камеры
- Внутренние свойства камеры
- Освещение

Внутренние факторы:

- Деформации объектов
- Внутриклассовая изменчивость

Да все!

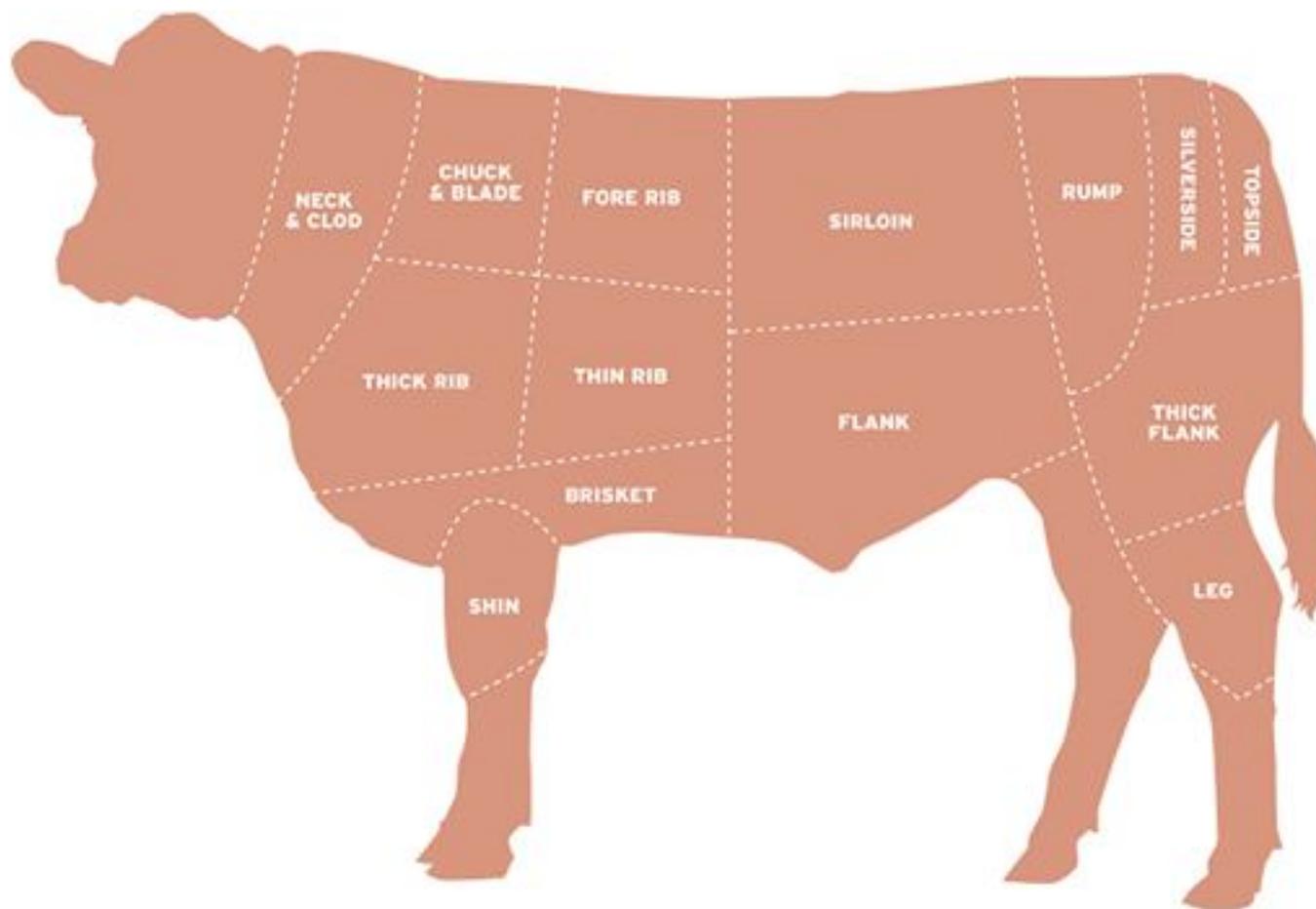


Из чего состоит изображение?





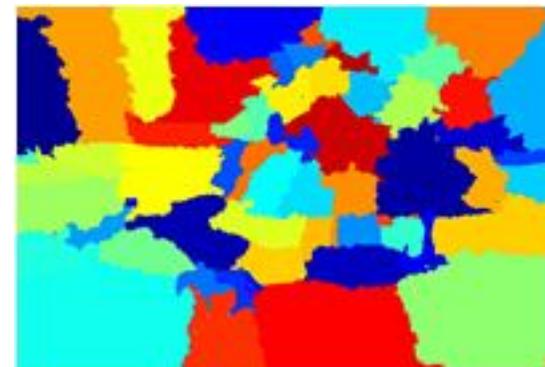
Из «кусков» - отдельных объектов





Сегментация

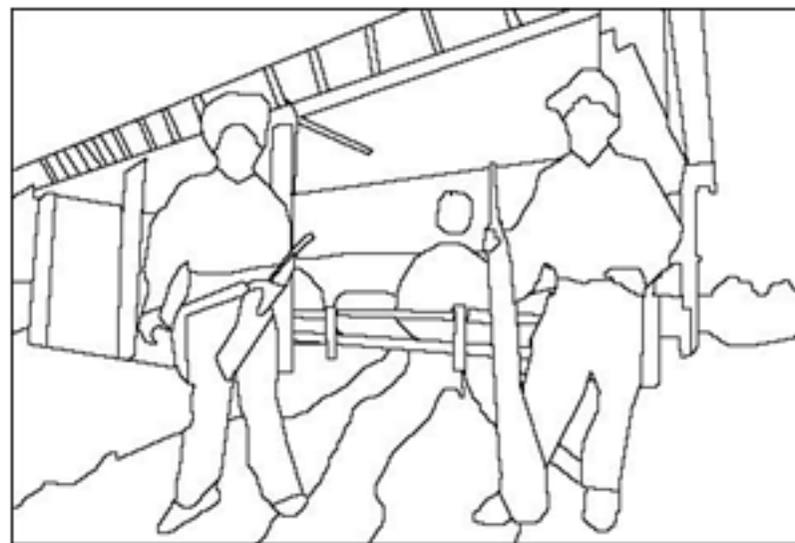
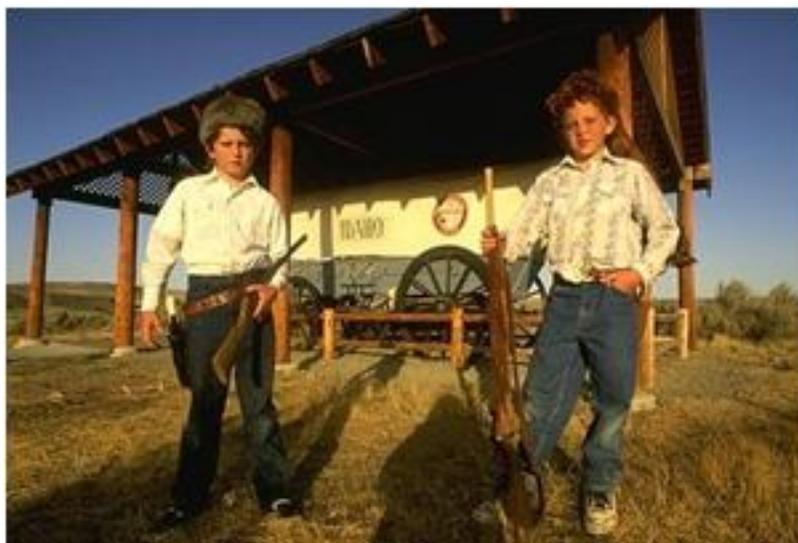
- Сегментация - это способ разделения сцены на «куски», с которыми проще работать
- Тесселяция - разбиение изображения на неперекрывающиеся области, покрывающие все изображение и однородные по некоторым признакам
- Можно и по другому сегментировать изображение
 - Пересекающиеся области
 - Иерархическое представление





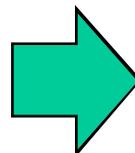
Требования к сегментации

- Сегментация - это способ разделения сцены на «куски», с которыми проще работать
- Границы сегментов должны соответствовать границам объектов





Результат сегментации

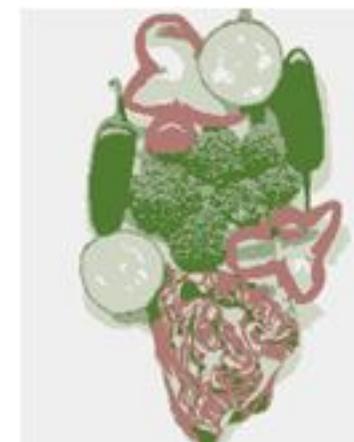


- Как мы будем записывать результат сегментации?
- Сделаем карту разметки – изображение, в каждом пикселе которого номер сегмента, которому принадлежит этот пиксель
- Визуализировать удобно каждый сегмент своим цветом



Алгоритмы к рассмотрению

- Бинаризация и выделения связанных компонент
- Последовательное сканирование
- Метод К-средних





Алгоритмы к рассмотрению

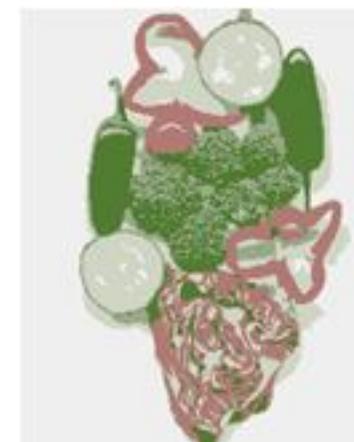
- Бинаризация и выделения связанных компонент



- Последовательное сканирование



- Метод К-средних





Простейшая сегментация

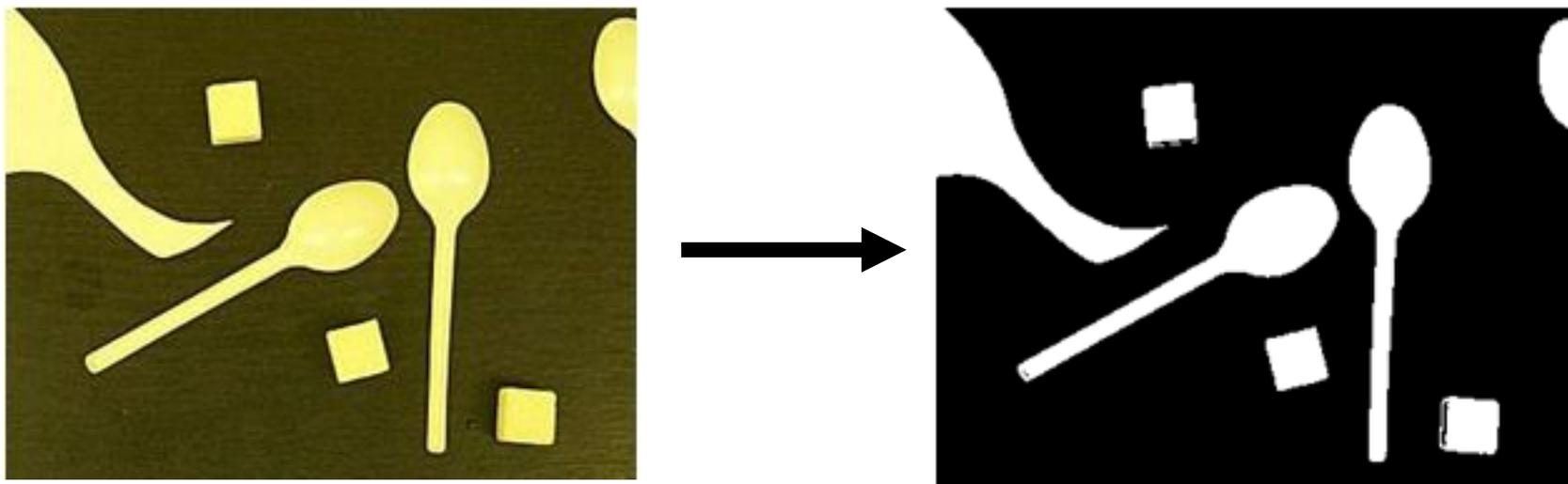
Чем отличаются объекты на этом изображении?



- Все объекты яркие, фон тёмный
- Объекты «контрастны» по отношению к простому (однотонному) фону
- Для сегментации такого изображения нам достаточно:
 - пороговая бинаризация
 - обработки шума
 - выделения связанных компонент



Пороговая бинаризация

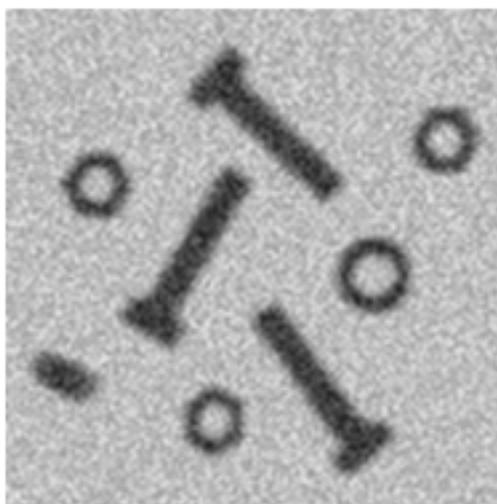


- Пороговая фильтрация (thresholding)
 - Пиксели, которых выше/ниже некоторого порога, заданного «извне», помечаются 1
 - Ниже порога помечаются 0
- Бинарное изображение – пиксели которого могут принимать только значения 0 и 1
- Бинаризация - построение бинарного изображения по полутоновому / цветному



Пороговая фильтрация

Более интересный способ – определение порога автоматически, по характеристикам изображения



- Нужно проанализировать распределение пикселей по яркости
- Анализ гистограммы яркости



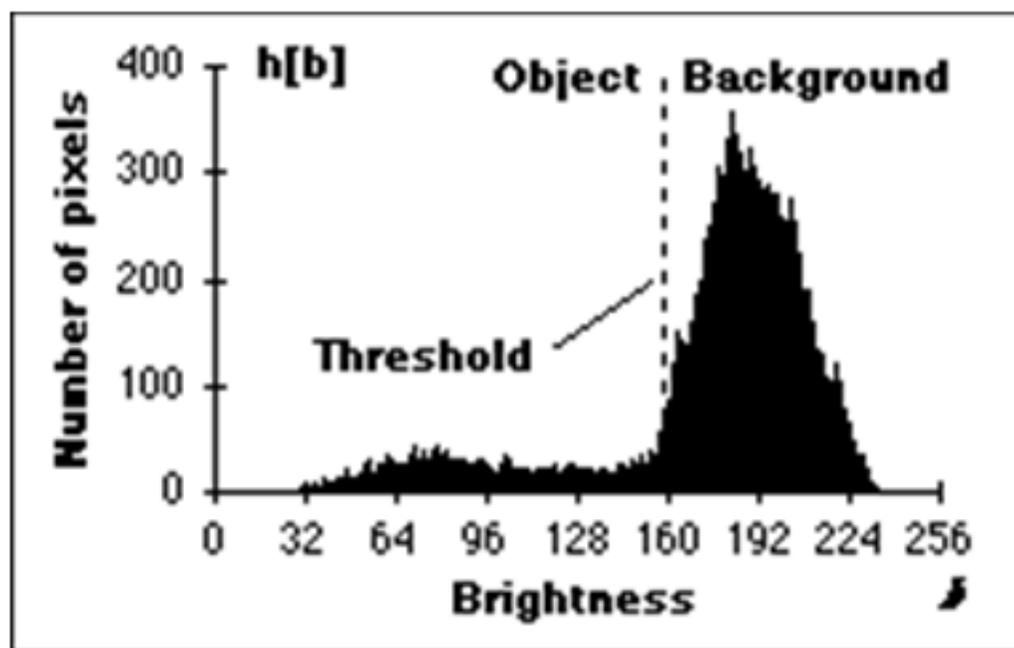
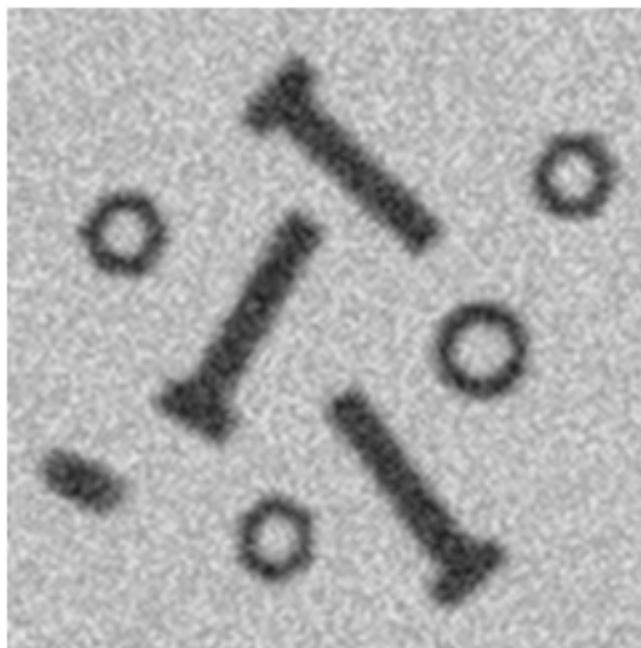
Гистограмма яркости

Гистограмма – это график распределения яркостей на изображении. На горизонтальной оси - шкала яркостей тонов от белого до черного, на вертикальной оси - число пикселей заданной яркости.





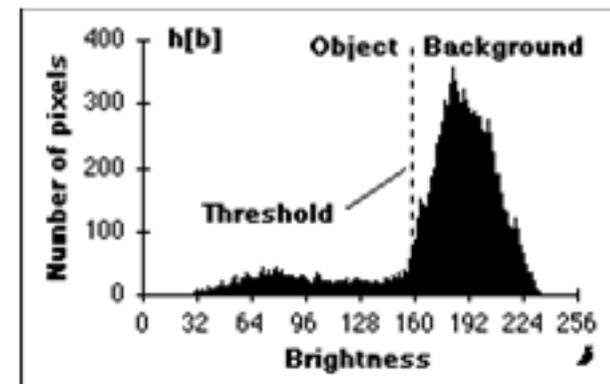
Гистограмма для нашего случая



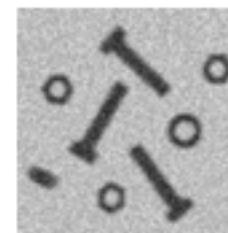


Анализ гистограммы

- Анализ симметричного пика гистограммы
- Применяется когда фон изображения дает отчетливый и доминирующий пик гистограммы, симметричный относительно своего центра.



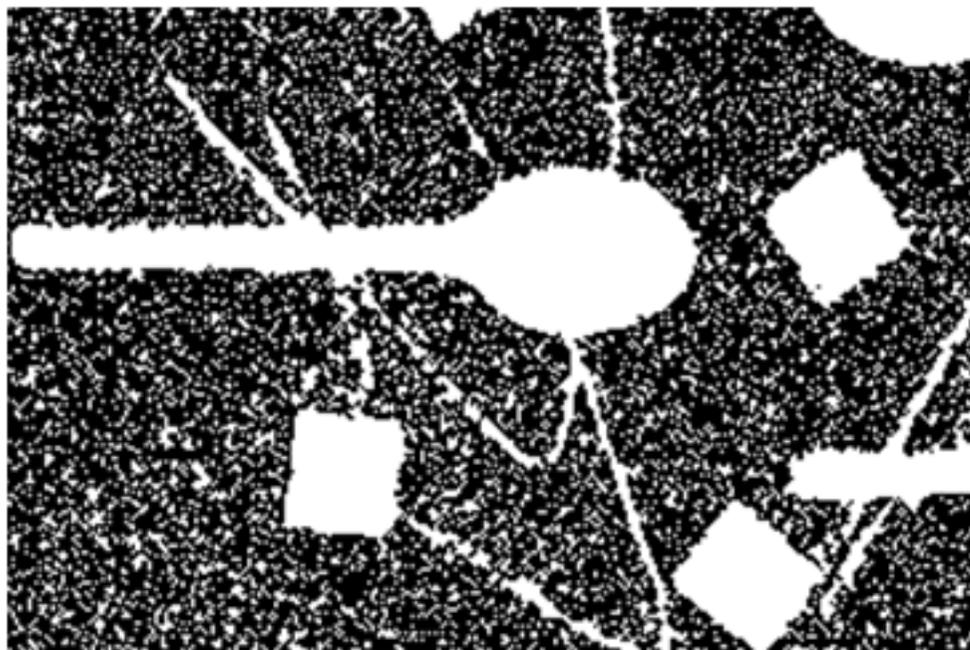
1. Сгладить гистограмму;
2. Найти ячейку гистограммы h_{\max} с максимальным значением;
3. На стороне гистограммы не относящейся к объекту (на примере – справа от пика фона) найти яркость h_p , количество пикселей с яркостью $\geq h_p$ равняется $p\%$ (например 5%) от пикселей яркости которых $\geq h_{\max}$;
4. Пересчитать порог $T = h_{\max} - (h_p - h_{\max})$;





Шум в бинарных изображениях

Пример бинарного изображения с сильным шумом

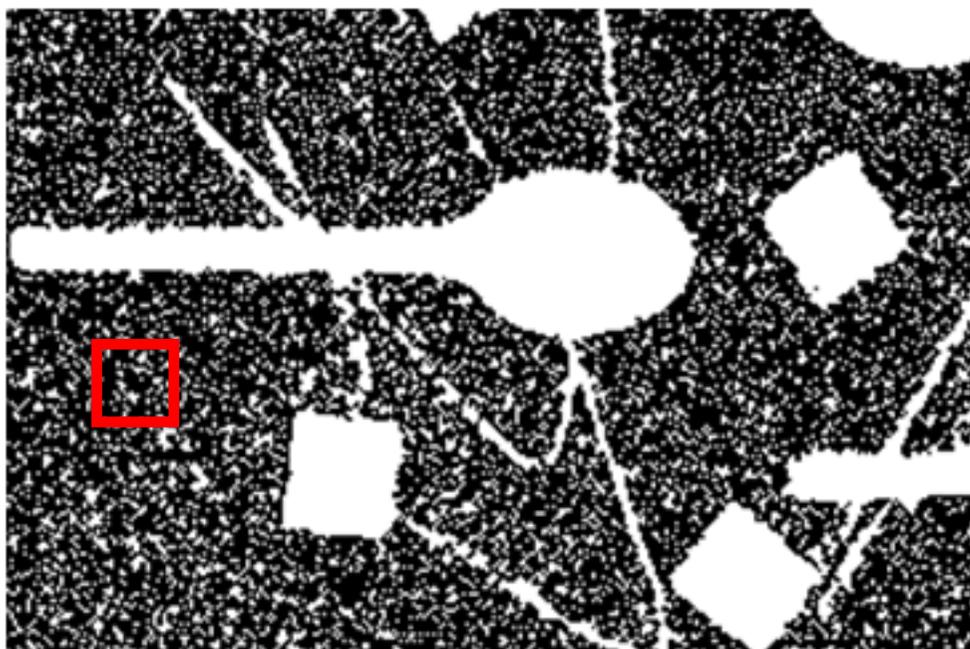


Часто возникает из-за невозможности полностью подавить шум в изображениях, недостаточной контрастности объектов и т.д.



Шум в бинарных изображениях

- ⑩ По одному пикселю невозможно определить – шум или объект?
- ⑩ Нужно рассматривать окрестность пикселя!





Подавление и устранение шума

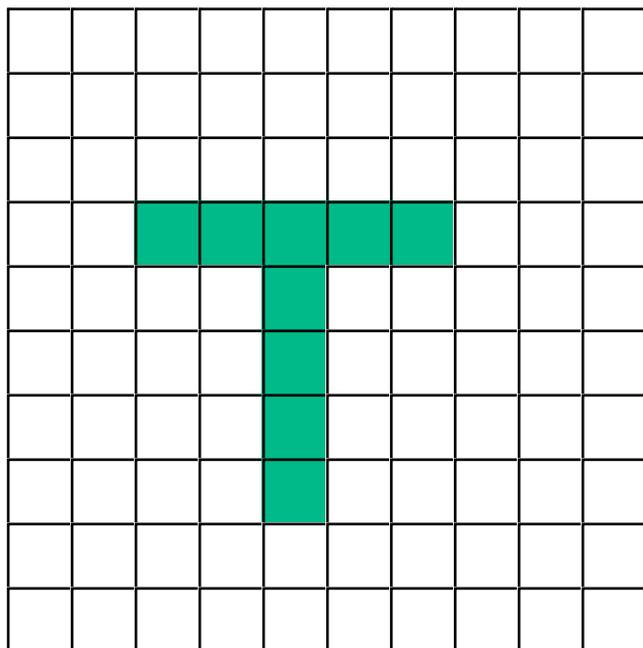
Широко известный способ - устранение шума с помощью операций математической морфологии:

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

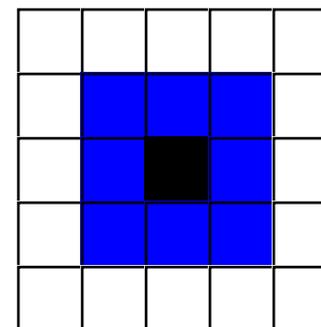


Математическая морфология

A



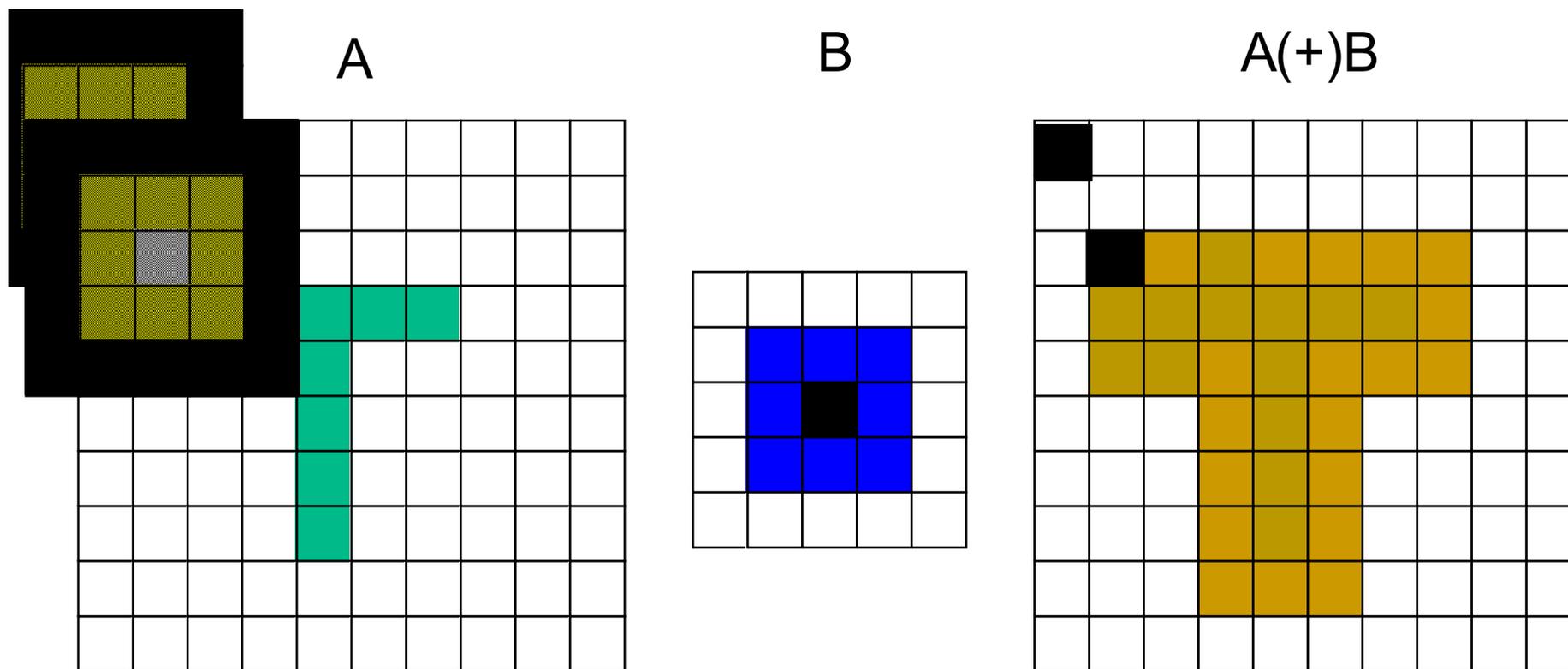
B



- Множество A обычно является объектом обработки
- Множество B (называемое структурным элементом) – инструмент обработки



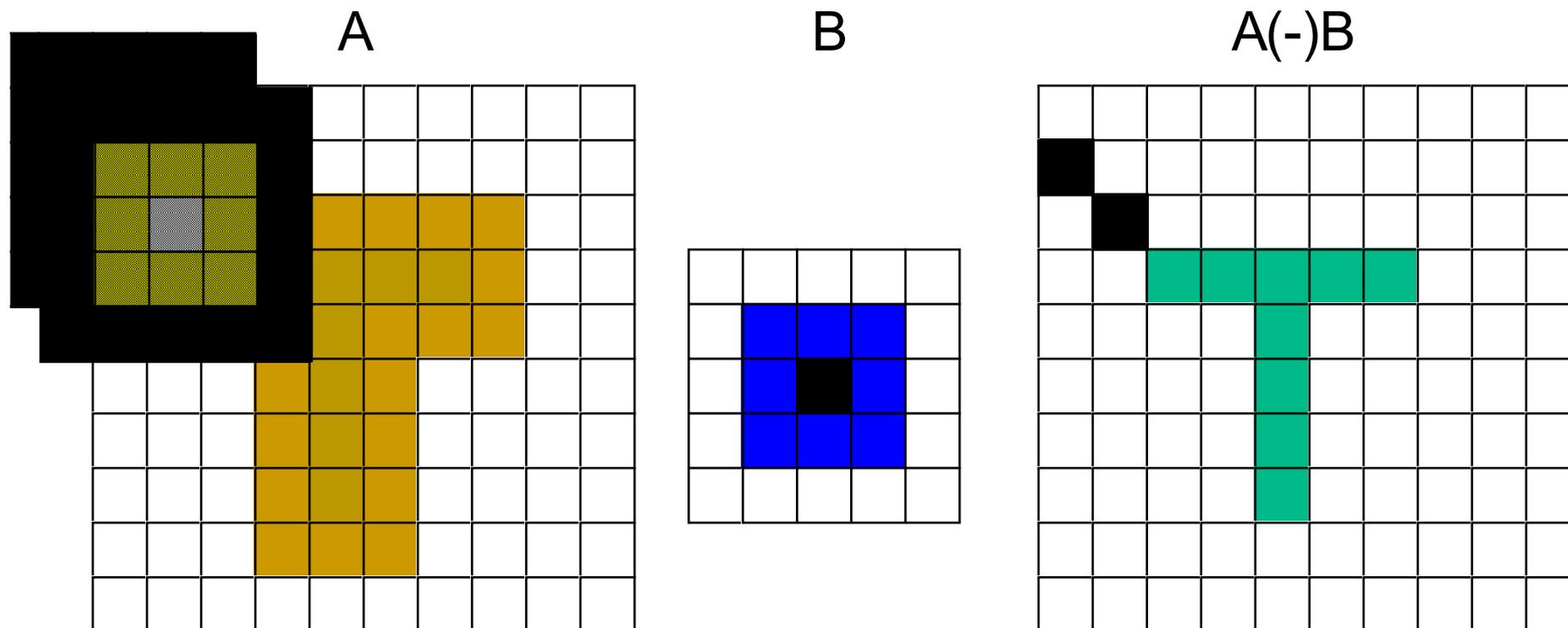
Расширение



Операция «расширение» - аналог логического «или»



Сужение



Операция «расширение» - аналог логического «и»



Результат операции сужения



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



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



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



Важное 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 скелета объекта
- Выделение сломанных зубьев на изображении шестерни



Операции раскрытия и закрытия

Морфологическое раскрытие (opening)

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

Морфологическое закрытие (closing)

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

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 изображения 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...

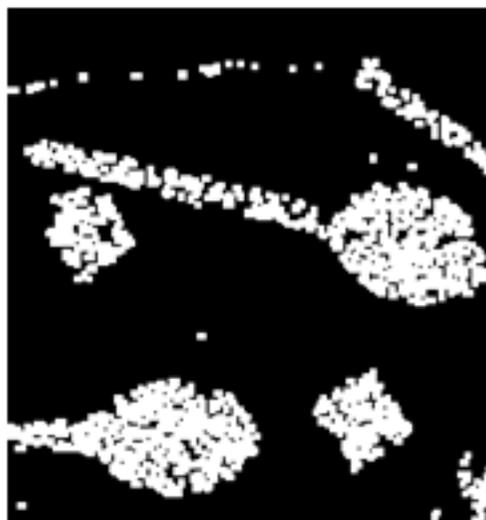




Применения операции открытия



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



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



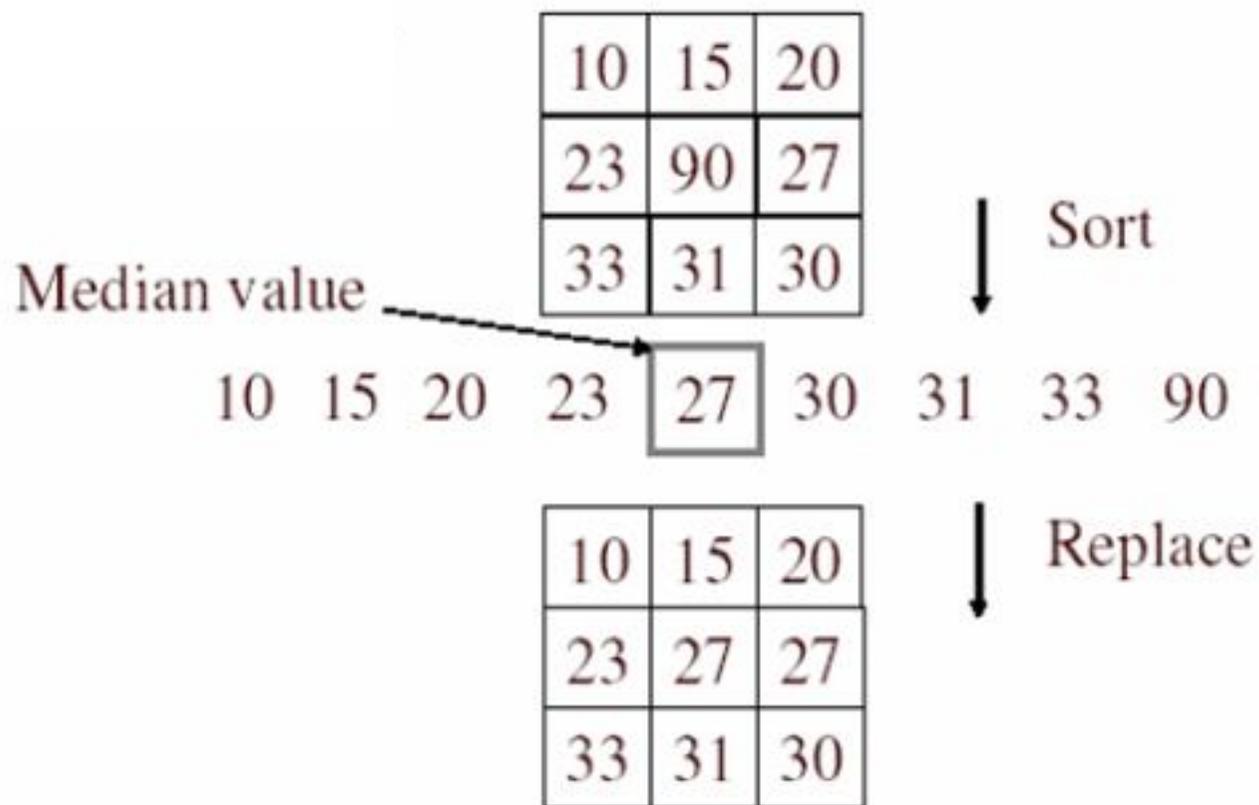
$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Часто помогает медианная фильтрация!



Медианный фильтр

- Посмотрим вначале в общем виде, на примере однотонного изображения (канал яркости)
- Строим выборку из пикселей по окрестности
- Сортируем и выбираем центральное значение («медиану»)





Для бинарных изображений

- Выбираем то значение, которого в окрестности больше

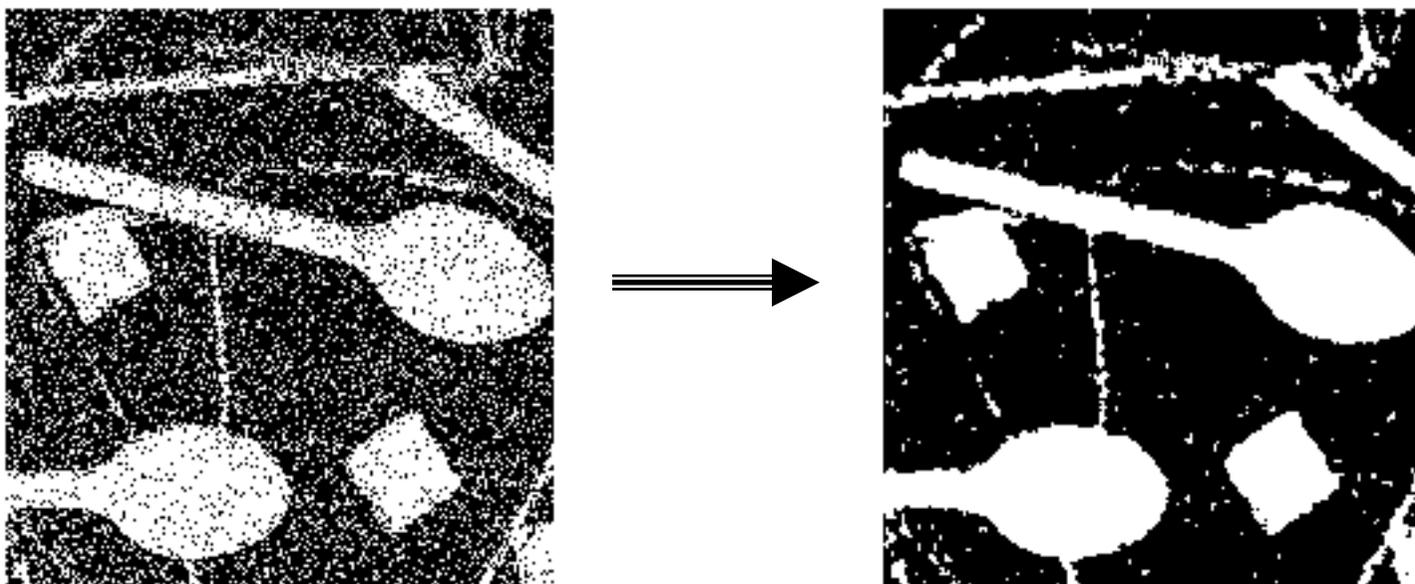
$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix}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

- Границы областей обычно остаются на месте
- Тонкие (по сравнению с размером фильтра) линии исчезнут
- Ещё про медианный фильтр мы вспомним на следующей лекции



Медианный фильтр

Фильтр с окрестностью 3x3



Теперь можем с помощью морфологии убрать оставшиеся точки, линии и т.д.



Что дальше?



Получили бинарное изображение



Нужна карта разметки



Выделение связных областей

Определение связной области:

- Множество пикселей, у каждого пикселя которого есть хотя бы один сосед, принадлежащий данному множеству.
- (или) Любые два пикселя которого связаны путём, проходящим только через пиксели множества



Соседи пикселей:

	1	
2	*	3
	4	

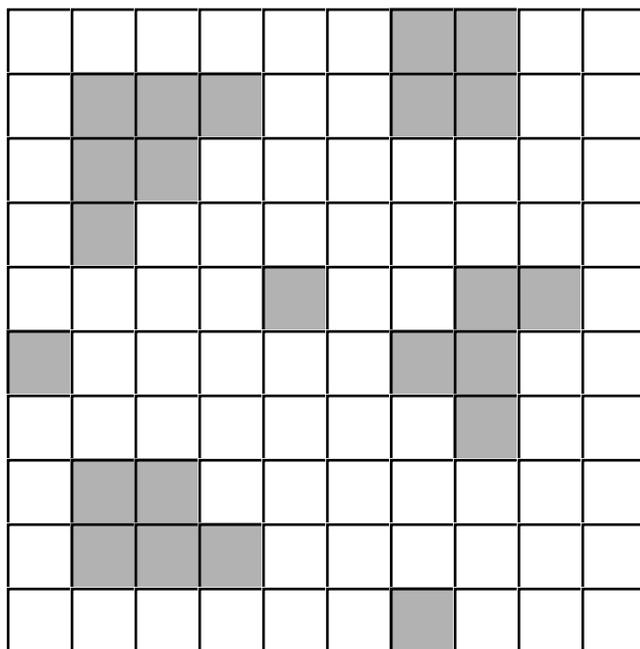
4-СВЯЗНОСТЬ

1	2	3
4	*	5
6	7	8

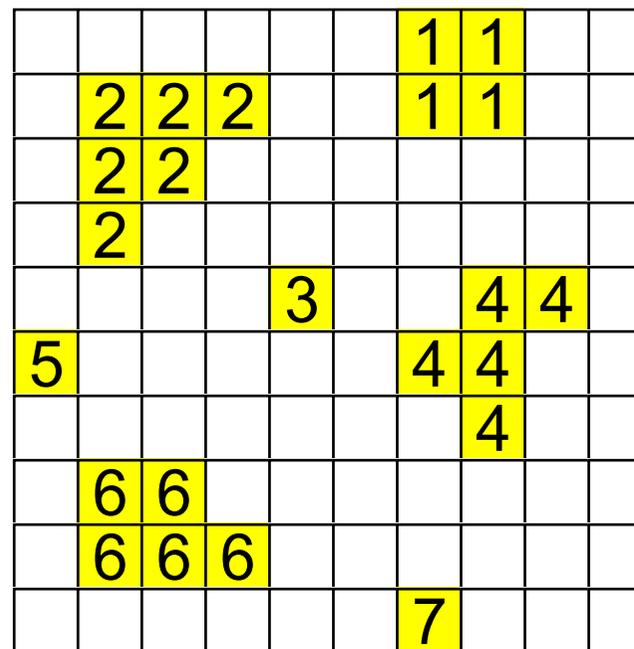
8-СВЯЗНОСТЬ



Разметка связанных областей



Бинарное изображение



Размеченное изображение



Рекурсивный алгоритм

```
void Labeling(BIT* img[], int* labels[])
{
    // labels должна быть обнулена
    L = 1;
    for(y = 0; y < H; y++)
        for(x = 0; x < W; x++)
        {
            Fill(img, labels, x, y, L++);
        }
}
```



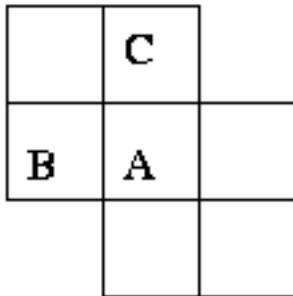
Рекурсивный алгоритм

```
void Fill(BIT* img[], int* labels[], int x, int y, int L)
{
    if( (labels[x][y] == 0) && (img[x][y] == 1) )
    {
        labels[x][y] = L;
        if( x > 0 )
            Fill(img, labels, x - 1, y, L);
        if( x < W - 1 )
            Fill(img, labels, x + 1, y, L);
        if( y > 0 )
            Fill(img, labels, x, y - 1, L);
        if( y < H - 1 )
            Fill(img, labels, x, y + 1, L);
    }
}
```



Последовательное сканирование

Последовательно, сканируем бинарное изображение сверху вниз, слева направо:



```
if A = 0
    do nothing

else if (not B labeled) and (not C labeled)
    increment label numbering and label A

else if B xor C labeled
    copy label to A

else if B and C labeled
    if B label = C label
        copy label to A
    else
        copy either B label or C label to A
        record equivalence of labels
```

За сколько операций мы разметим изображение?



Выделенные связанные компоненты





Алгоритмы к рассмотрению

- Бинаризация и выделения связанных компонент



- Последовательное сканирование



- Метод К-средних





Сегментация



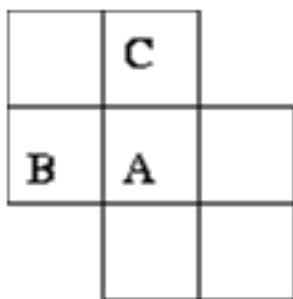
На основе
последовательного
сканирования можно
сделать и метод
сегментации изображений
на однородные области

Для простоты будем оценивать только яркость пикселов

Сегментация методом последовательного сканирования



Сканируем изображение сверху вниз, слева направо:



1. if $I(A) - I_{avg}(CI(B)) > \delta$ and $I(A) - I_{avg}(CI(C)) > \delta$ -
создаем новую область, присоединяем к ней пиксел А
2. if $I(A) - I_{avg}(CI(B)) < \delta$ xor $I(A) - I_{avg}(CI(C)) < \delta$ –
добавить А к одной из областей
3. if $I(A) - I_{avg}(CI(B)) < \delta$ and $I(A) - I_{avg}(CI(C)) < \delta$:
 1. $I_{avg}(CI(B)) - I_{avg}(CI(C)) < \delta$ –
сливаем области В и С.
 2. $I_{avg}(CI(B)) - I_{avg}(CI(C)) > \delta$ –
добавляем пиксел А к тому классу, отклонение от которого минимально.

$I(A)$ – яркость пиксела А

$CI(B)$ – область к которой принадлежит пиксел В

$I_{avg}(CI(B))$ – средняя яркость области к которой принадлежит В



Сегментация методом последовательного сканирования

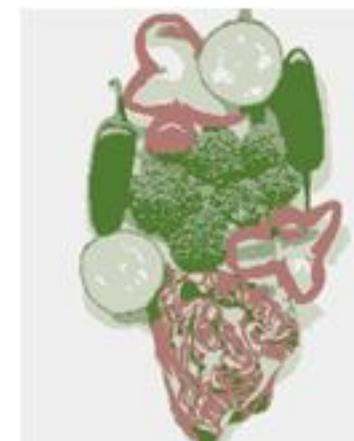
Пример:





Алгоритмы к рассмотрению

- Бинаризация и выделения связанных компонент
- Последовательное сканирование
- Метод К-средних

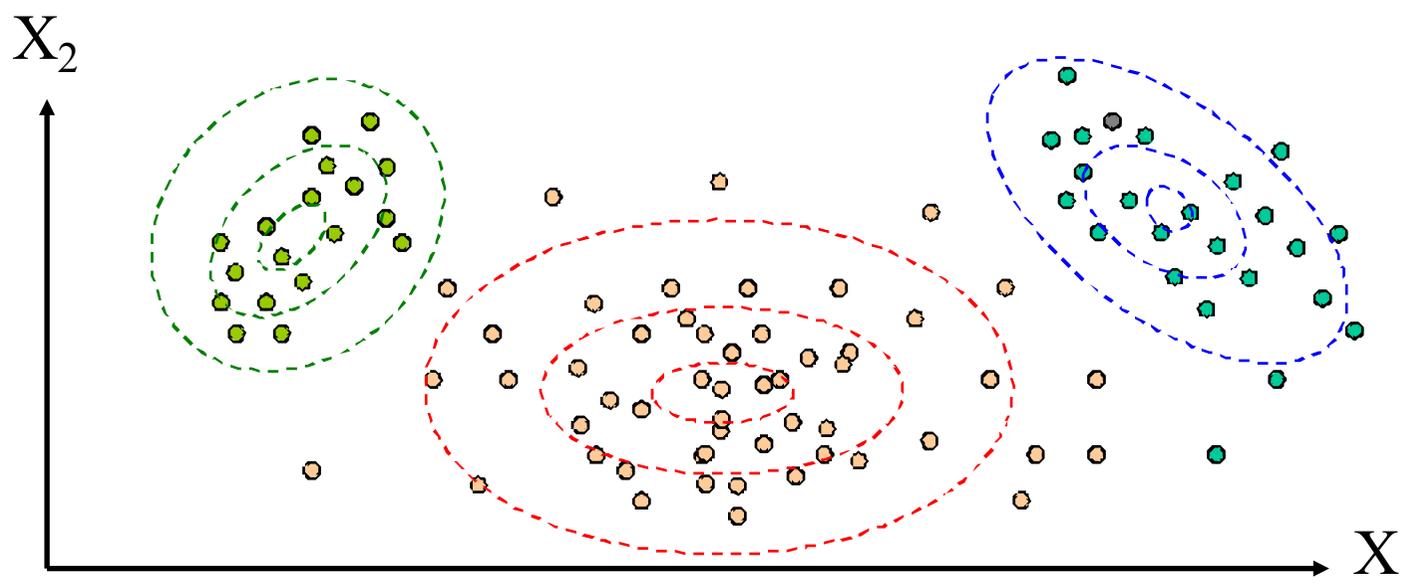




Кластеризация

Метод k -средних – метод **кластеризации** данных.

Целью *задачи кластеризации* является разбиение множества объектов на группы (кластеры) на основе некоторой меры сходства объектов.





Сегментация через кластеризацию

Что в случае сегментации изображения «объекты»?

- «Пиксели» изображения
- Мы «группируем» пиксели по схожести
- По каким признакам можем оценить схожесть пикселей?



Кластеризация K-средними

- Дано:
 - Набор векторов x_i $i=1, \dots, p$
 - k – число кластеров, на которые нужно разбить набор x_i
- Найти:
 - k векторов m_j , $j=1, \dots, k$ (центров кластеров)
 - Отнести каждый из векторов x_i к одному из k кластеров
 - При этом должен достигаться минимум суммы квадратов Евклидовых расстояний между точками x_i и назначенными им центрами кластеров m_j

$$D(X, M) = \sum_{\text{cluster } k} \sum_{\substack{\text{point } i \text{ in} \\ \text{cluster } k}} (x_i - m_j)^2$$

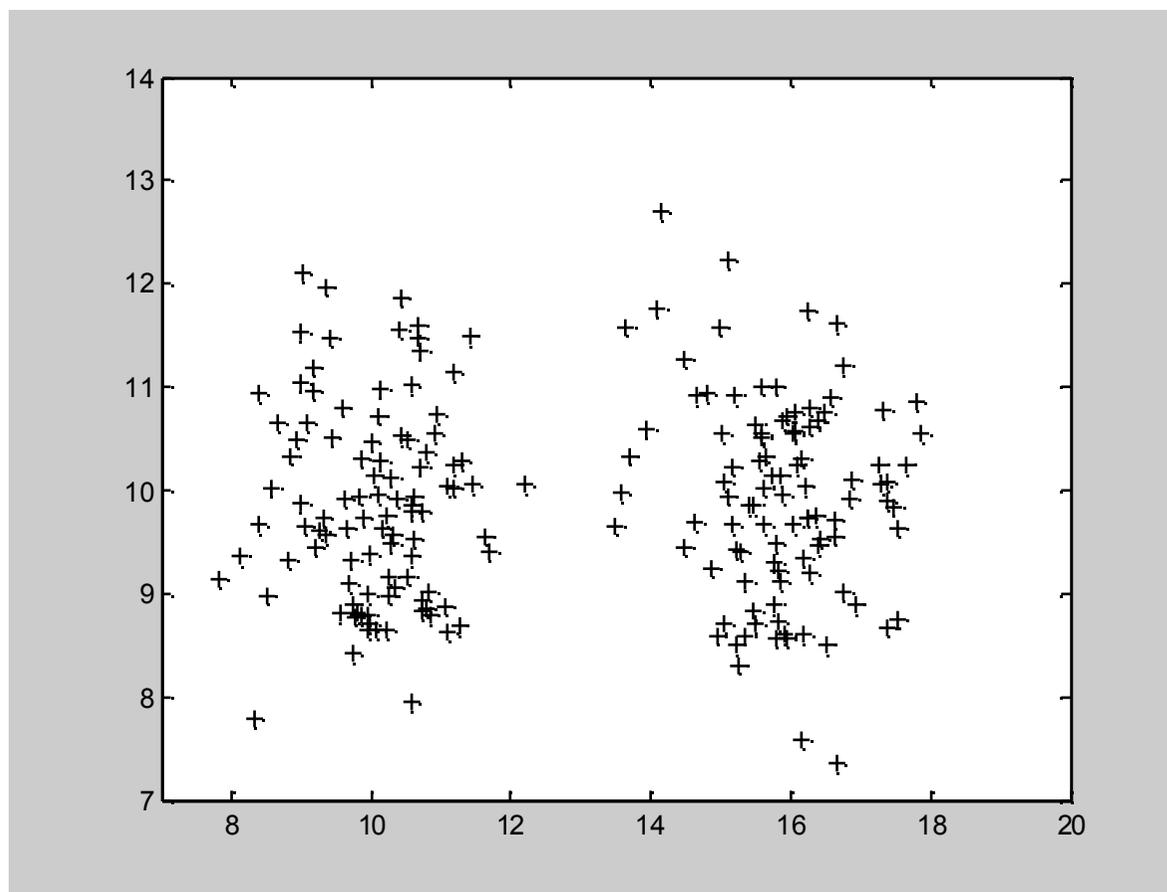


Алгоритм k-средних

1. Случайным образом выбрать k средних m_j
 $j=1, \dots, k$
2. Для каждого x_i $i=1, \dots, p$:
 - подсчитать расстояние от x_i до каждого из m_j
 $j=1, \dots, k$
 - Отнести (приписать) x_i к кластеру j' ,
расстояние до центра которого $m_{j'}$
минимально
3. Пересчитать средние m_j $j=1, \dots, k$ по всем кластерам
4. Повторять шаги 2, 3 пока кластеры не перестанут изменяться



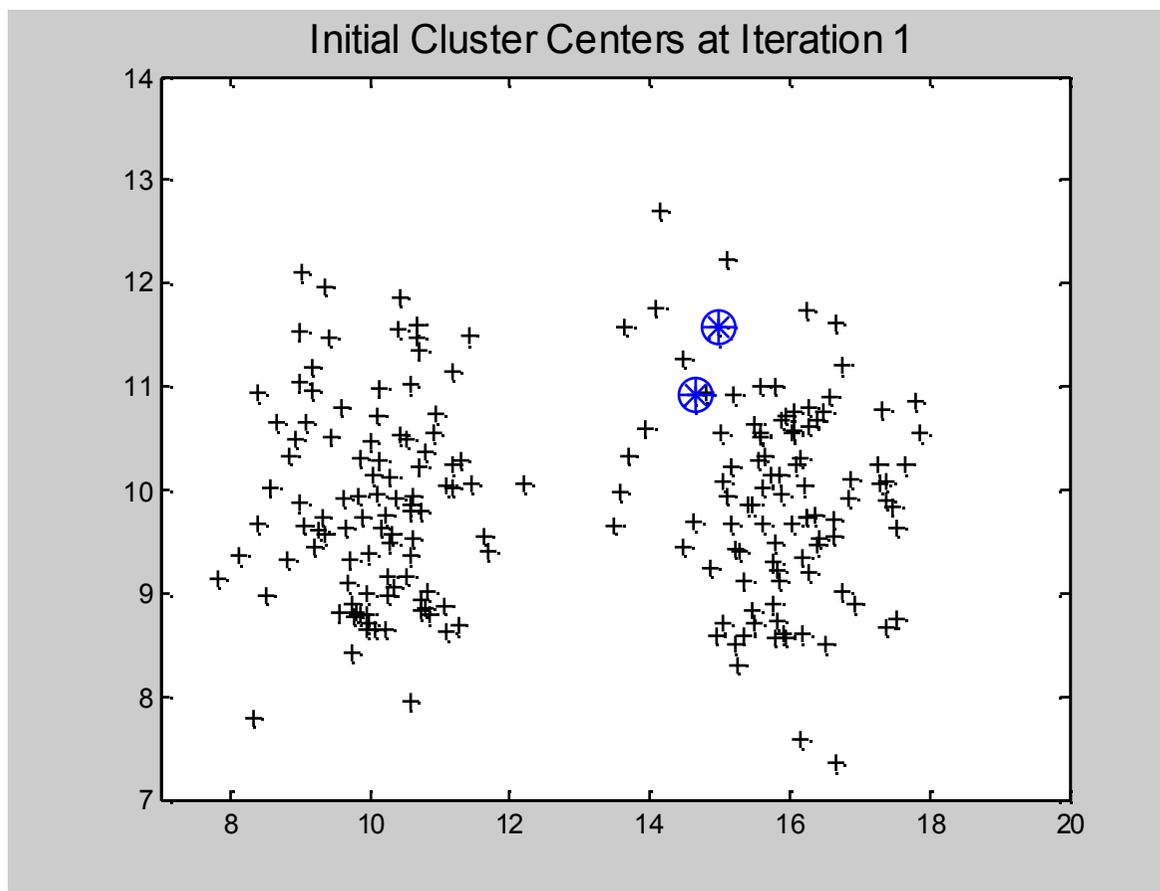
Пример кластеризации в 2D



Исходные данные



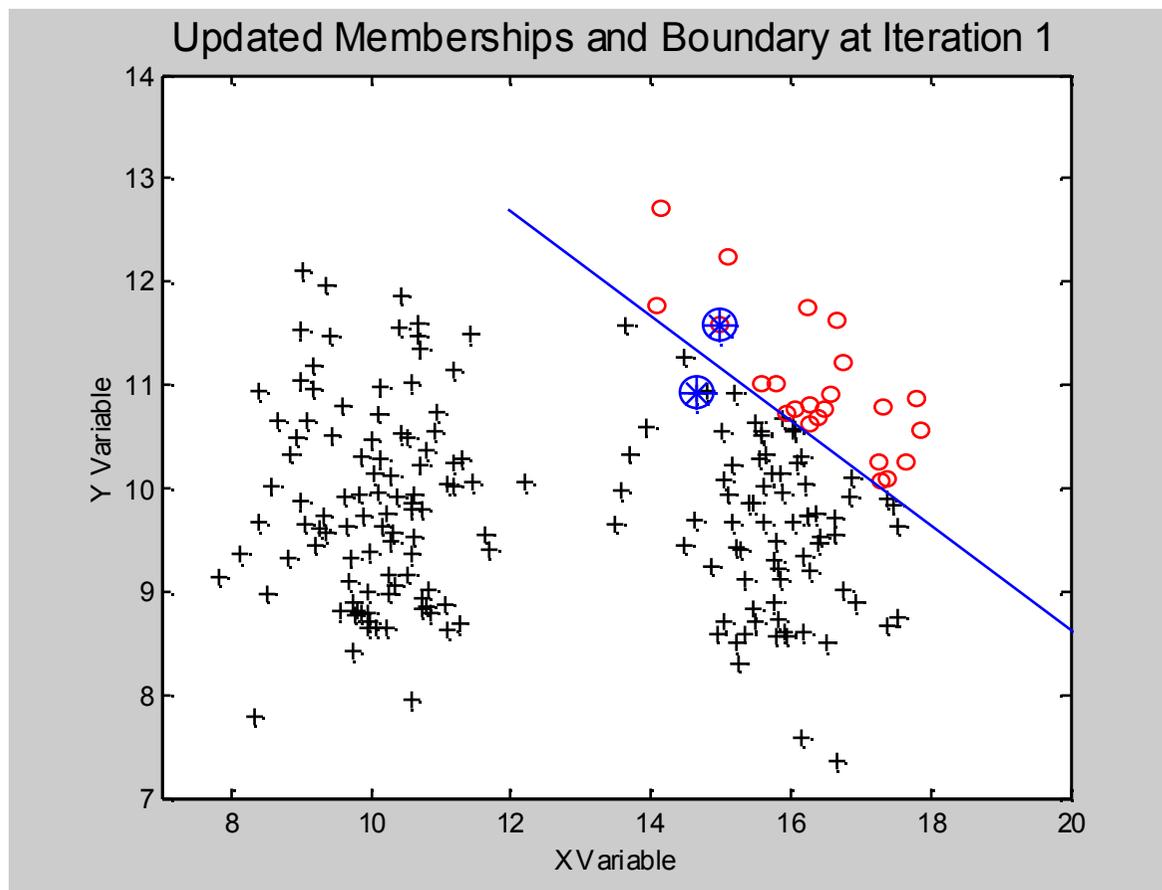
Пример кластеризации в 2D



Случайная инициализация центров кластеров (шаг 1)



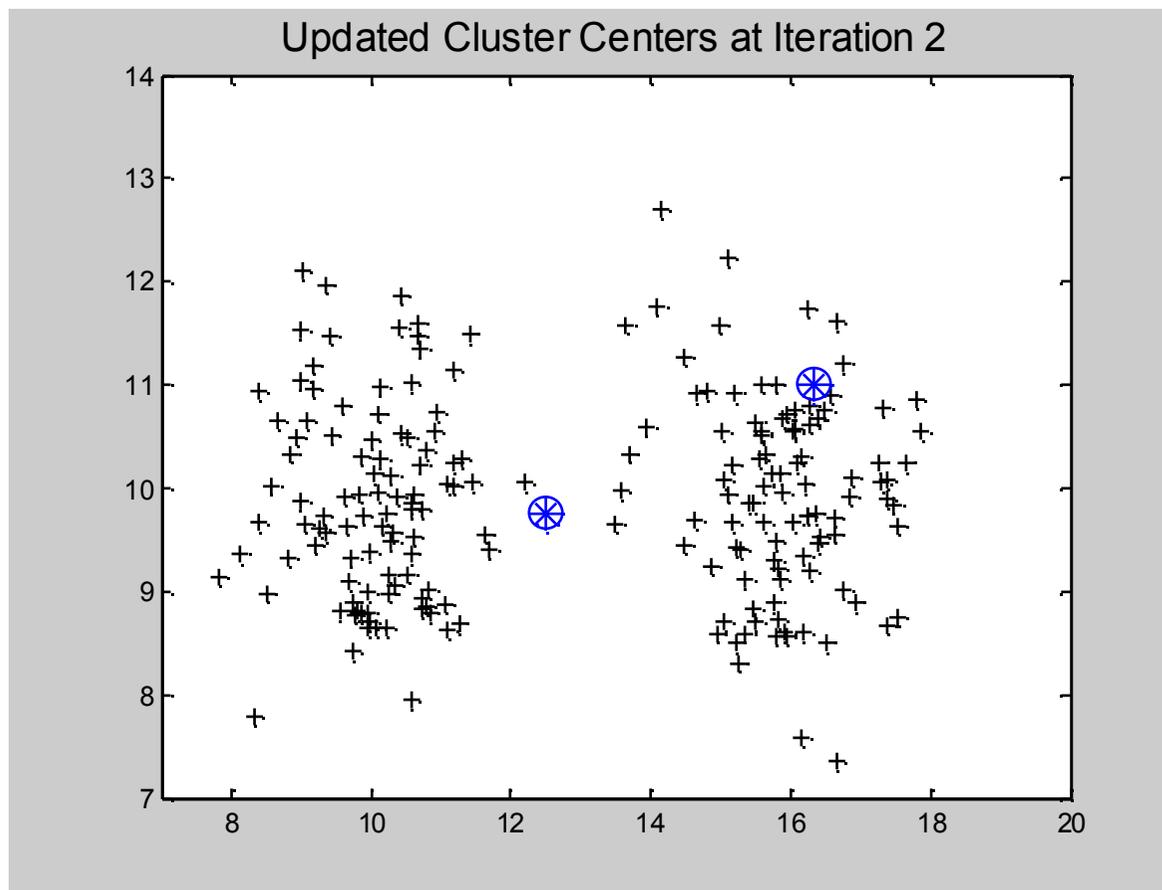
Пример кластеризации в 2D



Кластеры после первой итерации (шаг 2)



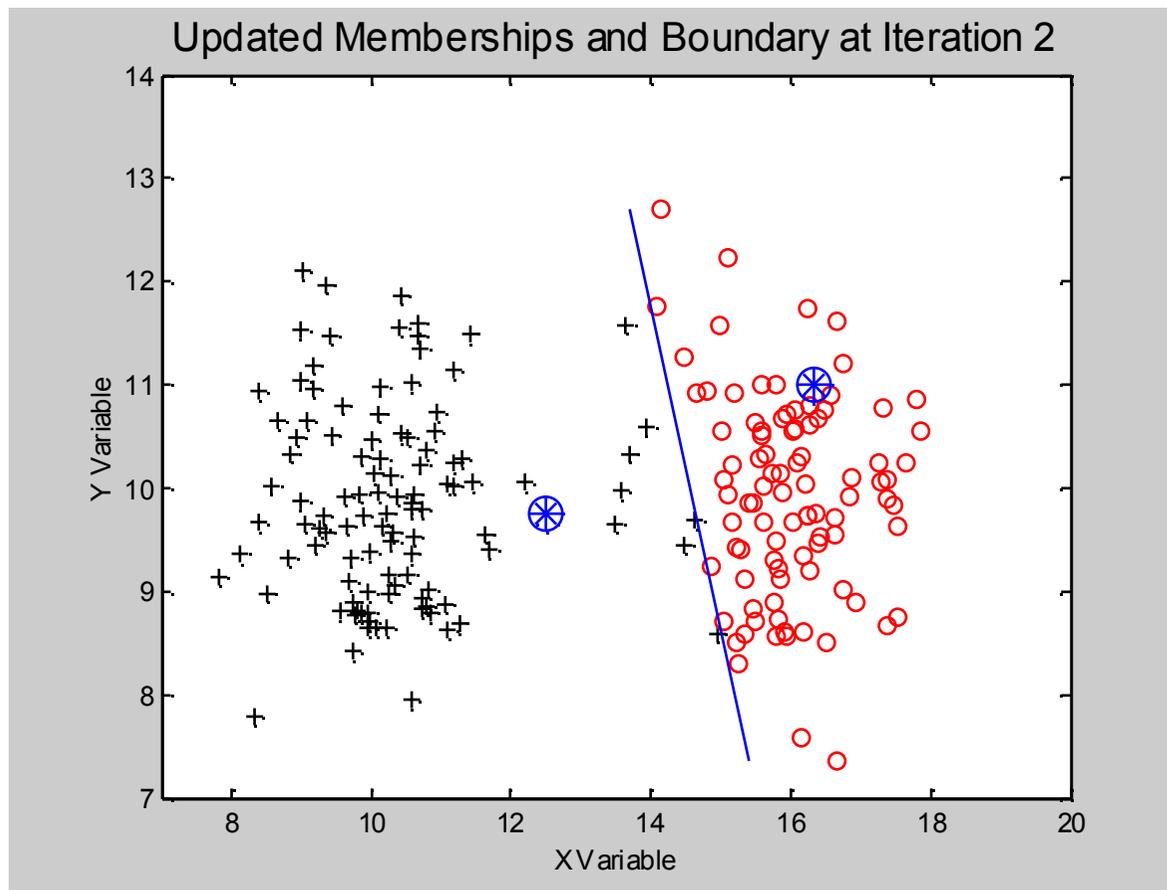
Пример кластеризации в 2D



Пересчет центров кластеров после первой итерации (шаг 3)



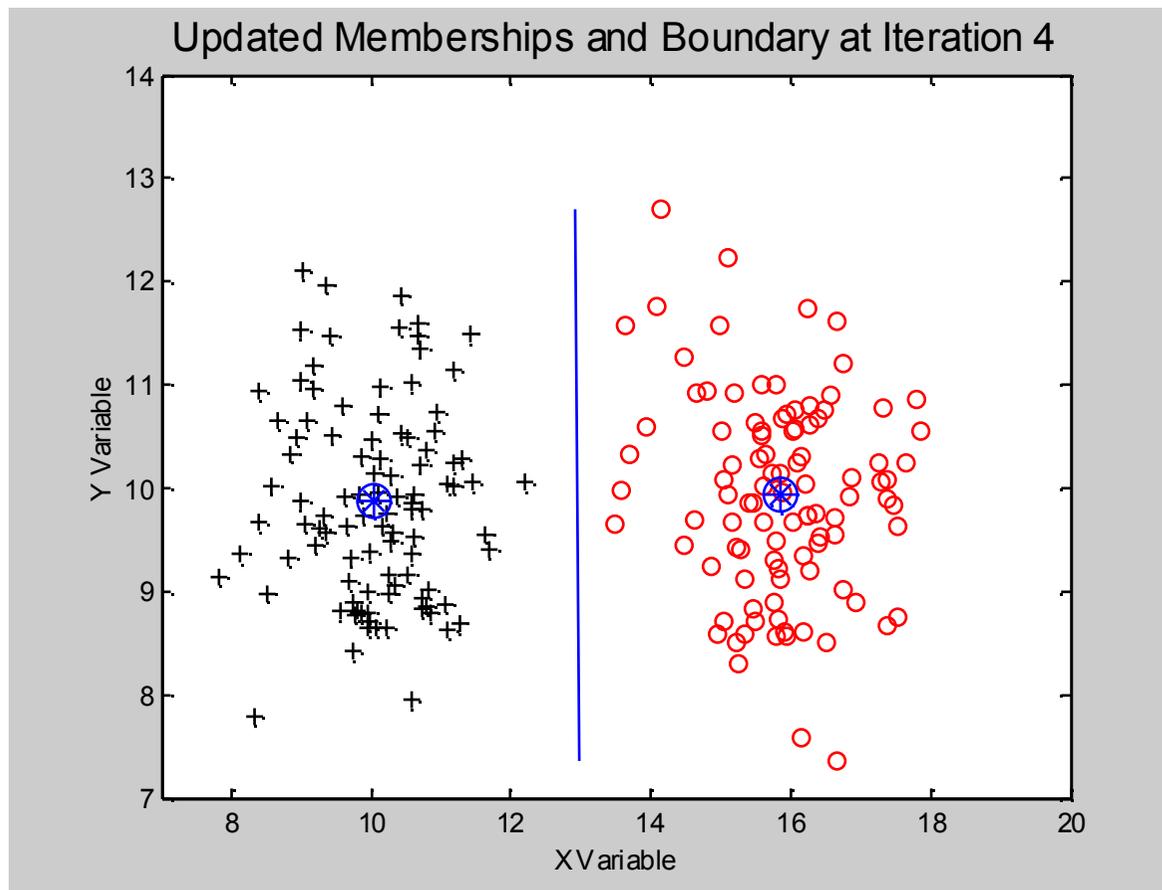
Пример кластеризации в 2D



Кластеры после второй итерации (шаг 2)



Пример кластеризации в 2D

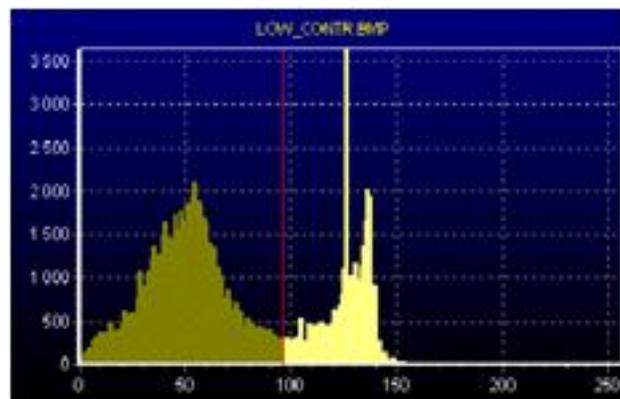


Стабильная конфигурация после четвертой итерации



Сегментации изображения по яркости

Рассматриваем *одномерное пространство яркостей пикселей* и производим в нем кластеризацию с помощью *k-средних*. Это дает автоматическое вычисление яркостных порогов.



(Для получения бинарного изображения $k=2$)



Алгоритм К-средних. Примеры.



Исходное изображение



Кластеры по яркости

Вектор признак – {яркость}



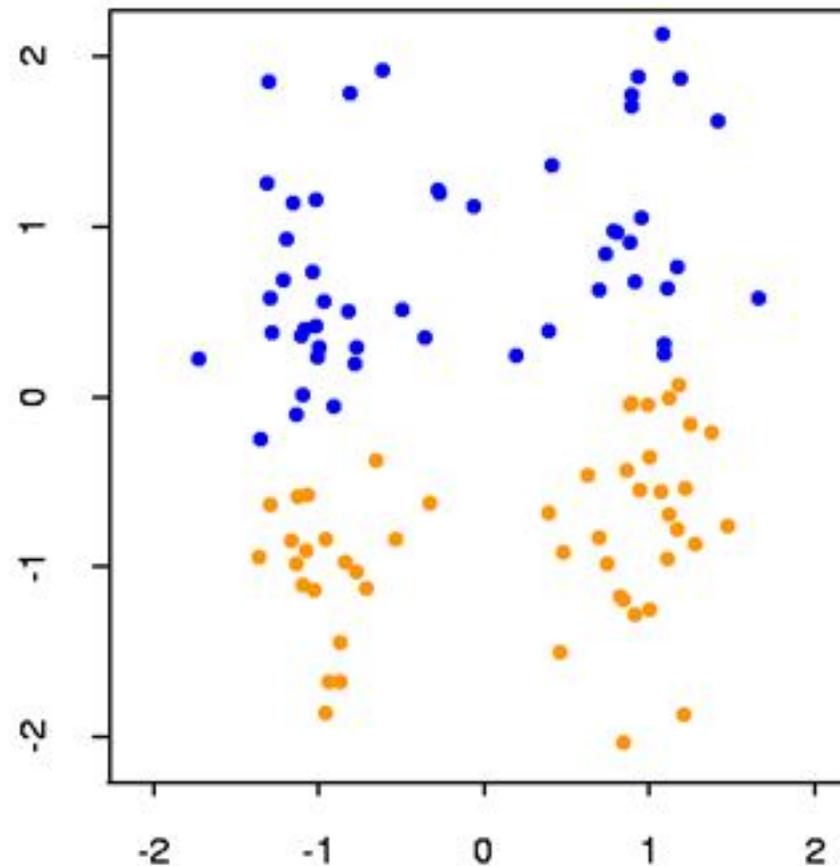
Кластеры по цвету

Вектор признак – {R, G, B}



Алгоритм K-средних

- Однопараметрический
 - Требуется знание только о количестве кластеров
- Рандомизирован
 - Зависит от начального приближения
- Не учитывает строения самих кластеров



Есть целый ряд других, более совершенных методов кластеризации!



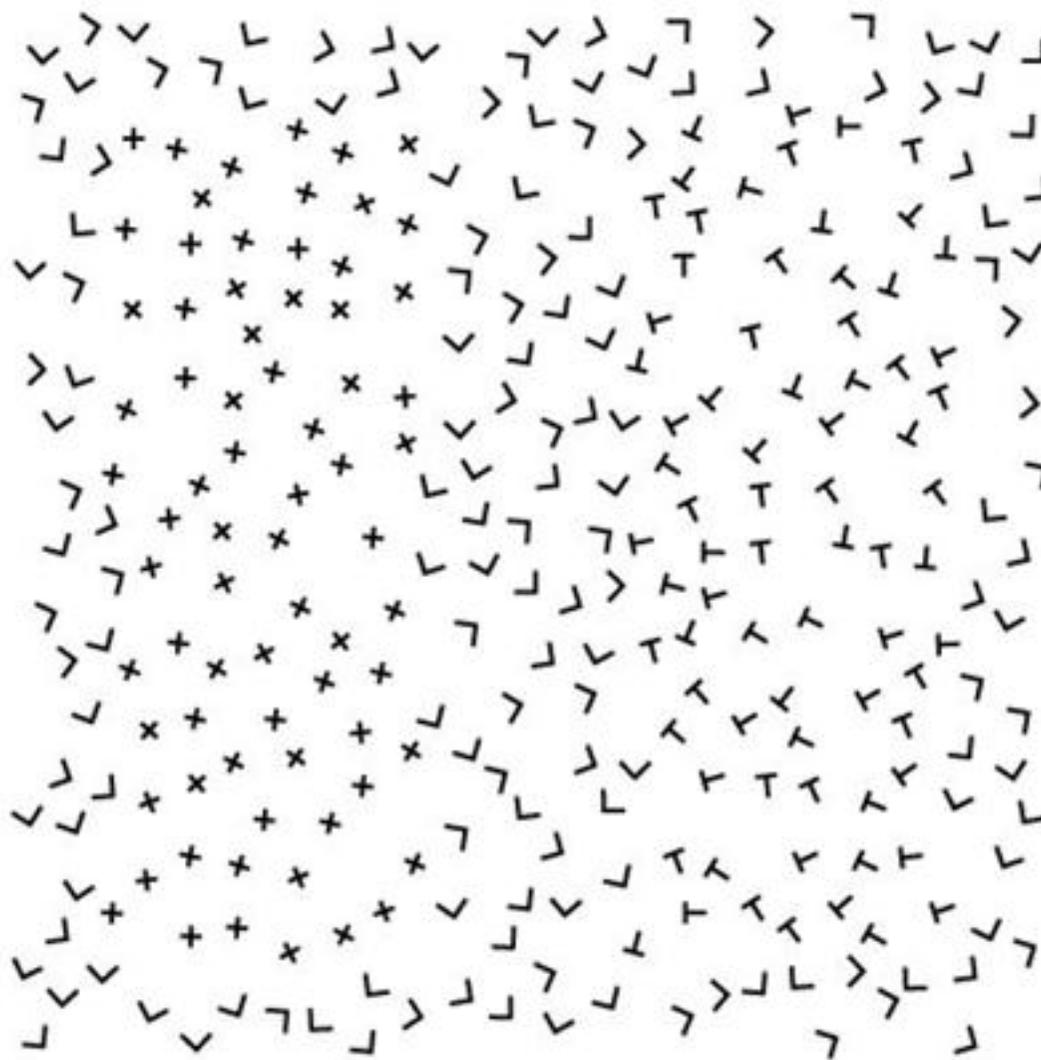
Признаки изображения

Какие признаки мы можем использовать для сравнения пикселей и регионов?

- Яркость
- Цвет
- ?



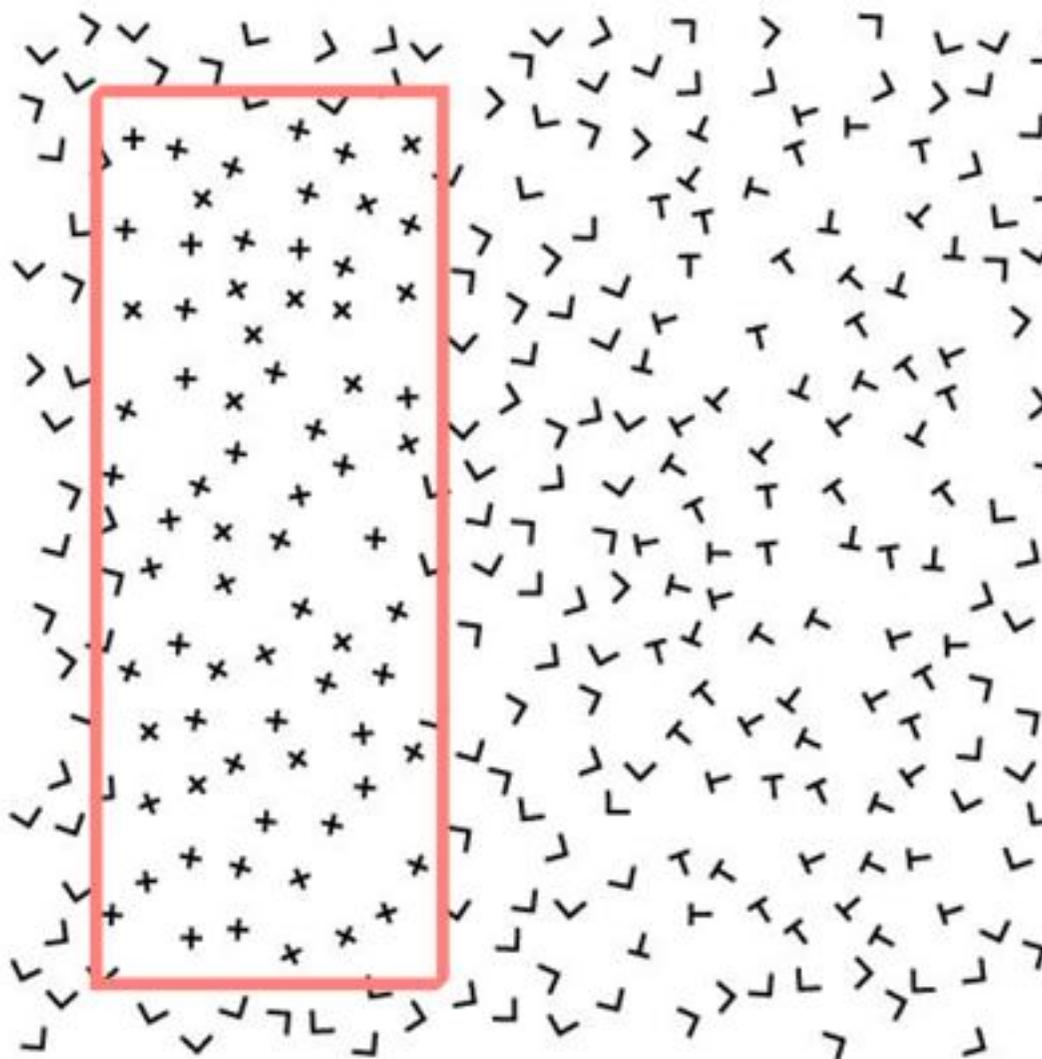
Пример



Видите отдельные области?

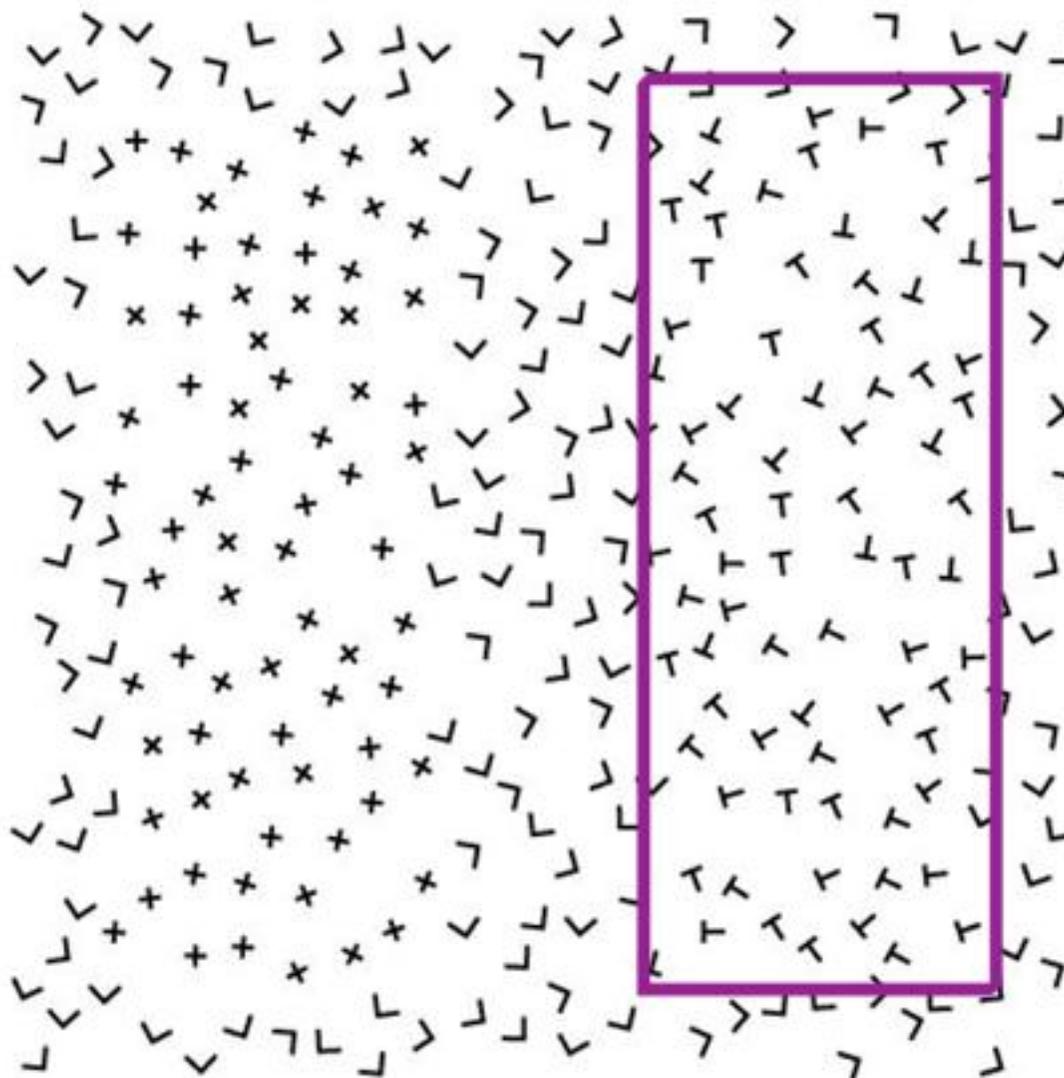


Область 1



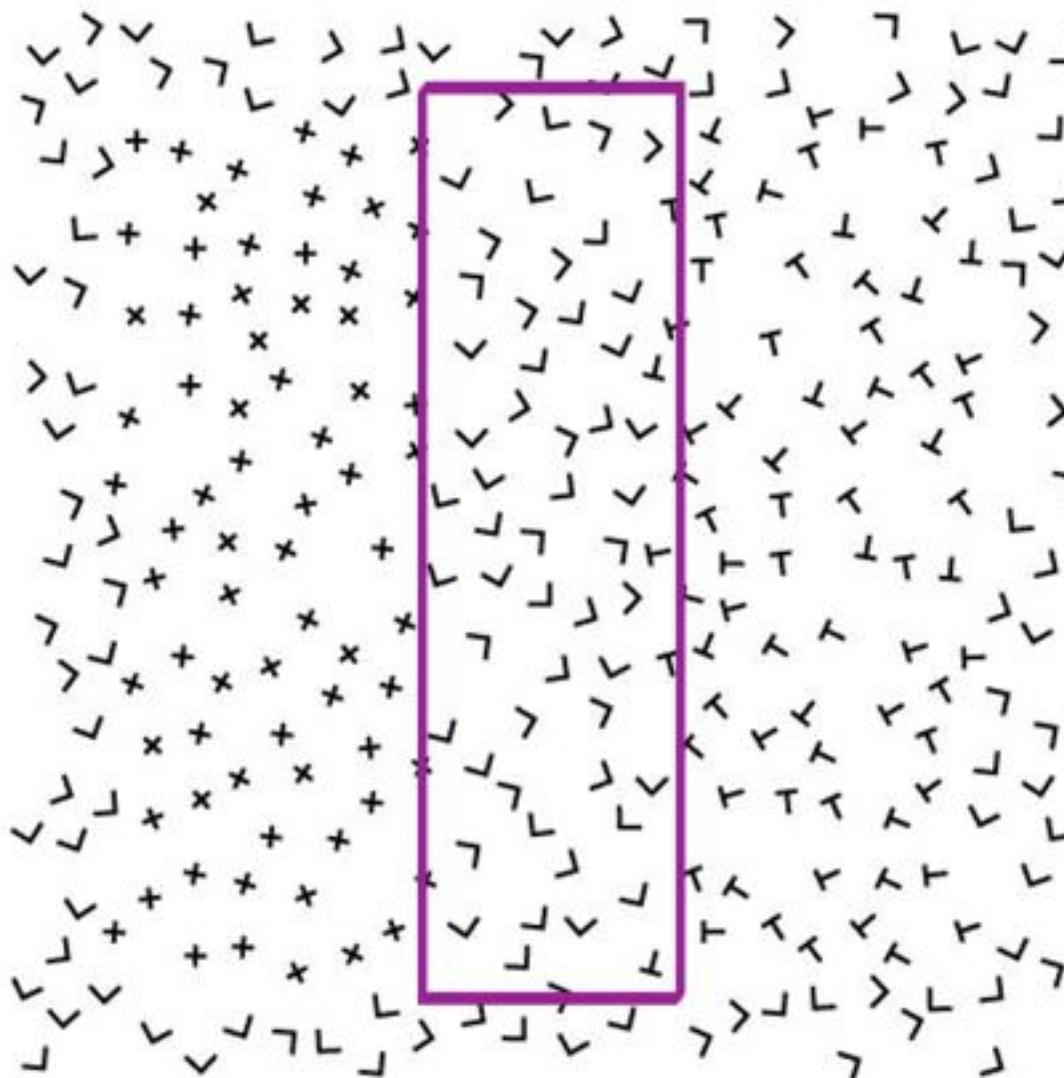


Область 2



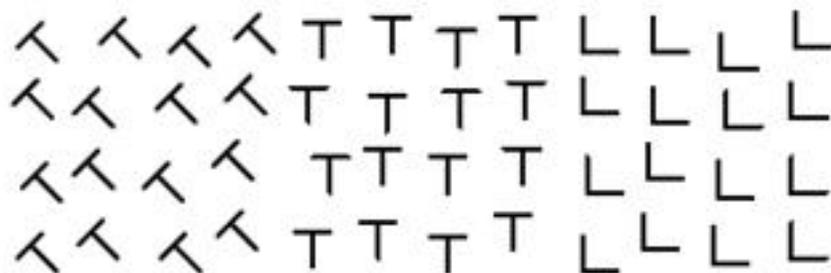


Область 2





Текстура



(a)

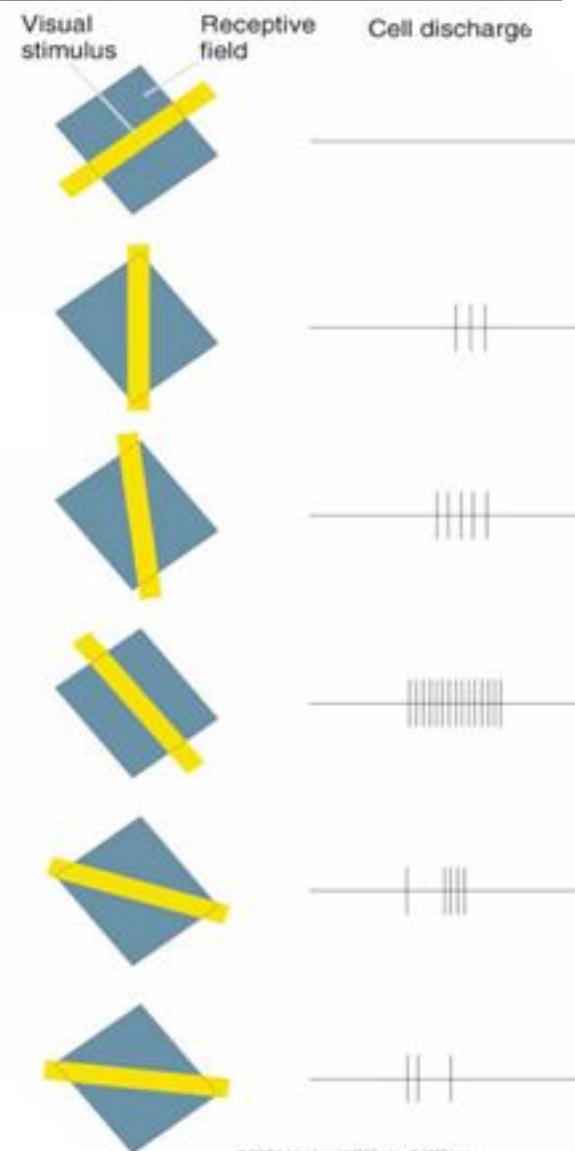
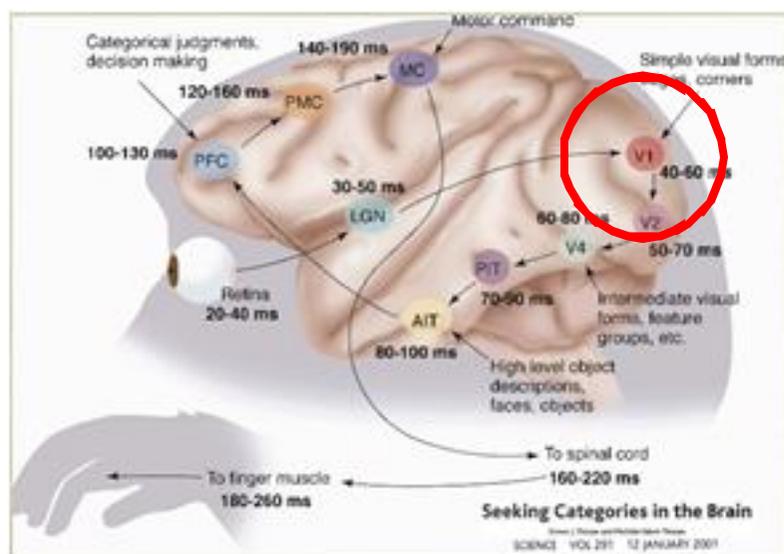


(b)

- **Текстура** — преимущественная ориентация элементов, составляющих материал (одно из определений)
- (b) - типичные примеры шаблонов (текстурных элементов) для исследований психофизиологического восприятия изображений
- Человек явно учитывает свойства текстуры фрагмента при распознавании изображений и выделении областей



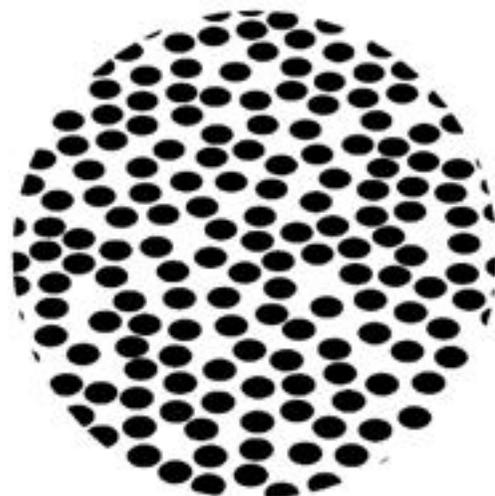
«Простые клетки» V1



- В первичной визуальной коре головного мозга есть клетки, чувствительные к краям определенной ориентации
- Для каждой области есть набор таких клеток, чувствительные к ориентации элементов текстуры
- Как их измерять – в следующей лекции



Психологическое свойство текстуры





Форма из текстуры

- Человек интуитивно считает текстуру **изотропной**, т.е. с постоянными свойствами на поверхности объекта
- Shape from texture: Исходя из предположения об изотропности шаблона текстуры, можно определить наклон поверхности

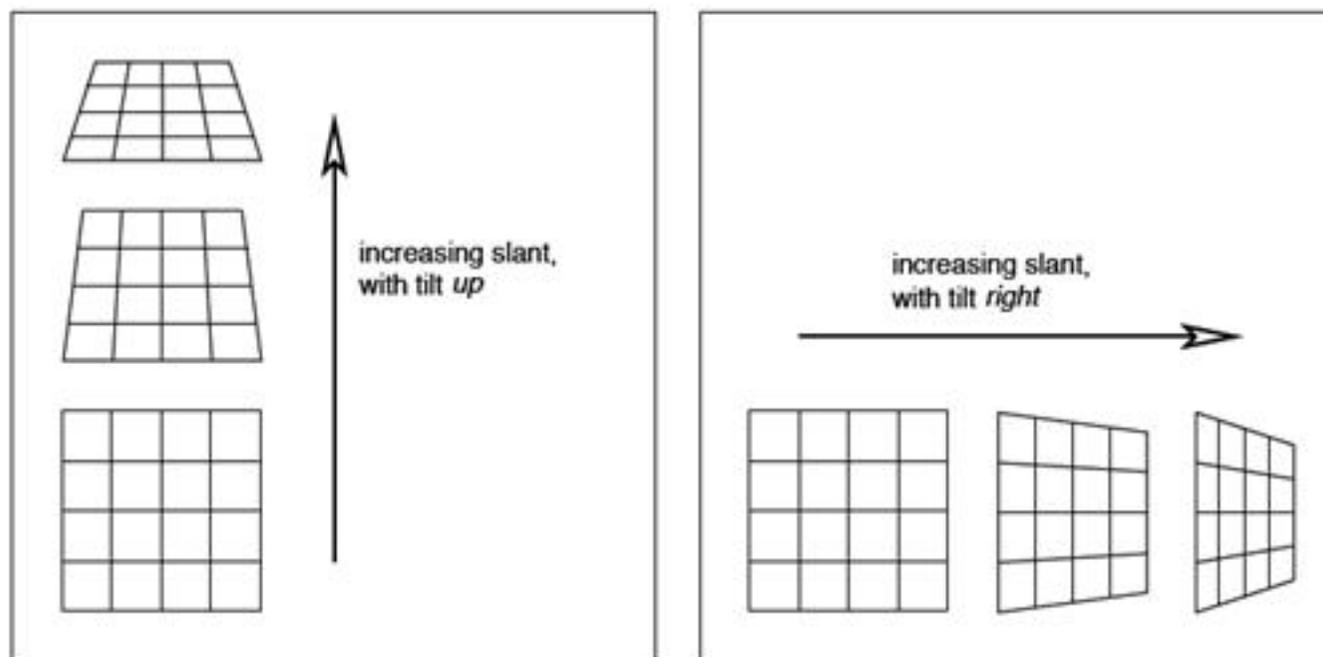
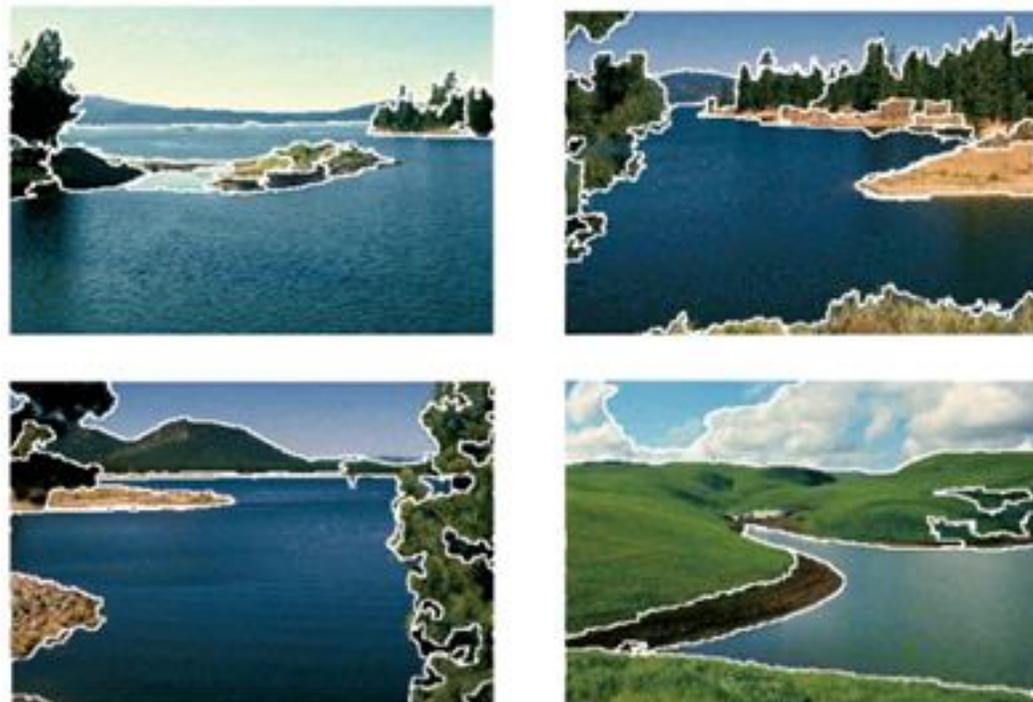


Figure 8.7. Surface orientation is often characterized in terms of *slant* and *tilt*.



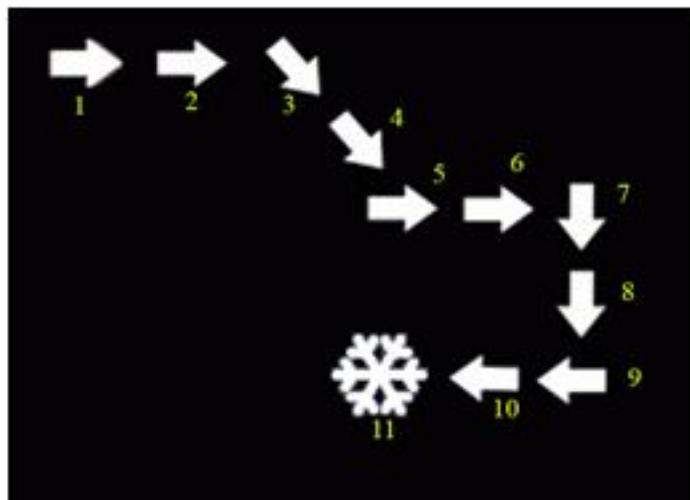
Современные алгоритмы



- Задача сегментации изображений продолжает активно исследоваться.
- Иногда используется один признак, иногда множество
- Подробнее алгоритмы сегментации рассматриваются в с\к «Доп. главы компьютерного зрения»



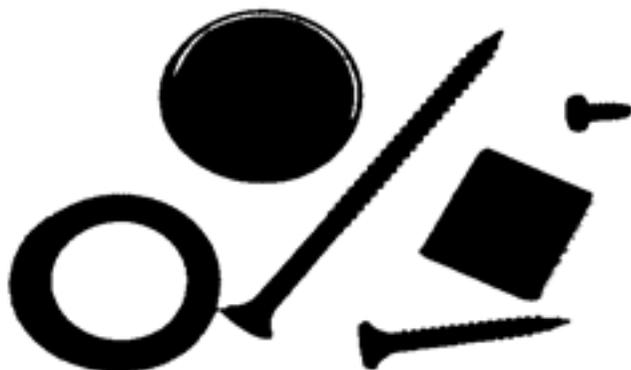
Анализ выделенных областей



Для анализа требуется вычислить некоторые числовые характеристики (признаки) областей:

- геометрические признаки
- фотометрические признаки

На основе этих характеристик можно классифицировать получаемые области





Геометрические признаки

Для каждой области можно подсчитать некий набор простейших числовых характеристик:

- Площадь
- Центр масс
- Периметр
- Компактность
- Ориентацию главной оси инерц
- Удлиненность (эксцентриситет)

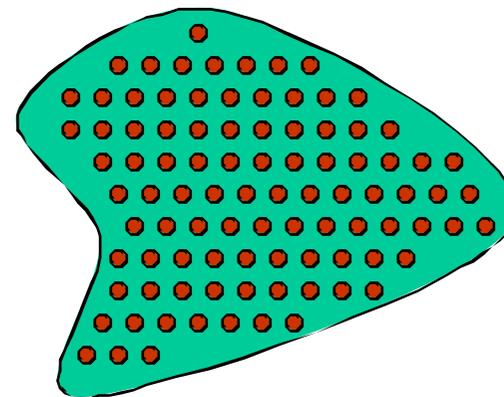




Площадь и центр масс

- Площадь – количество пикселей в области;

$$A = \sum_{x=0}^m \sum_{y=0}^n I(x, y)$$



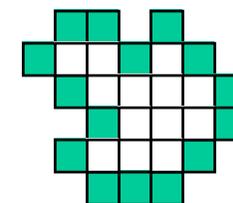
- Центр масс

$$\bar{x} = \frac{\sum_{x=0}^m \sum_{y=0}^n xI(x, y)}{A}; \bar{y} = \frac{\sum_{x=0}^m \sum_{y=0}^n yI(x, y)}{A}$$



Периметр и компактность

- Периметр – количество пикселей принадлежащих границе области;



- Компактность – отношение квадрата периметра к площади;

$$C = \frac{P^2}{A}$$

Наиболее компактная фигура – $C = 4\pi$
круг:





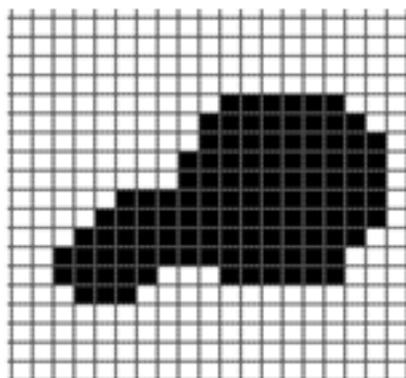
Подсчет периметра области

1. Пиксель лежит на границе области, если он сам принадлежит области и хотя бы один из его соседей области не принадлежит.
(внутренняя граница)
2. Пиксель лежит на границе области, если он сам не принадлежит области и хотя бы один из его соседей области принадлежит.
(внешняя граница)

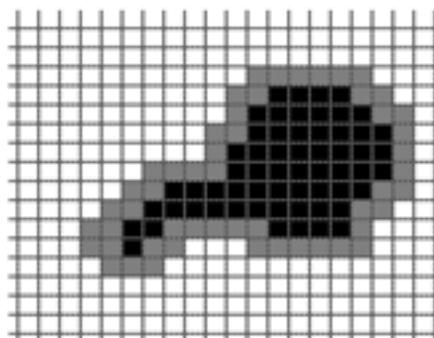
Периметр зависит также от того 4-х или 8-ми связность используется для определения соседей.



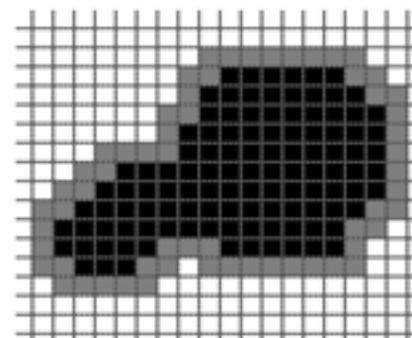
Пример периметров области



Область



Внутренняя граница



Внешняя граница



Операция оконтуривания объекта

При работе с бинарными изображениями контуры объекта можно получить с помощью операций математической морфологии

Внутреннее оконтуривание

- $C_1 = A - (A (-) B)$

Внешнее оконтуривание

- $C_0 = (A (+) B) - A$



Пример оконтуривания объекта





Статистические моменты области

Дискретный центральный момент m_{ij} области определяется следующим образом:

$$m_{ij} = \sum_{x,y \in S} (x - \bar{x})^i (y - \bar{y})^j I(x, y)$$

Центр масс области



Инвариантные характеристики

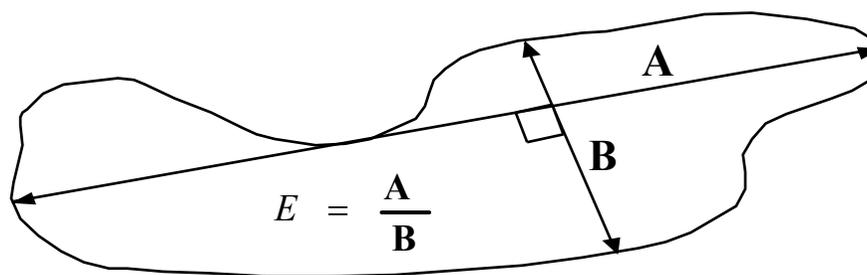
Для распознавания нас интересуют характеристики инвариантные по отношению к масштабированию, переносу, повороту:

- Удлиненность, нецентрированность (эксцентриситет)

$$elongation = \frac{m_{20} + m_{02} + \sqrt{(m_{20} - m_{02})^2 + 4m_{11}^2}}{m_{20} + m_{02} - \sqrt{(m_{20} - m_{02})^2 + 4m_{11}^2}}$$

- Компактность

$$C = \frac{P^2}{A}$$

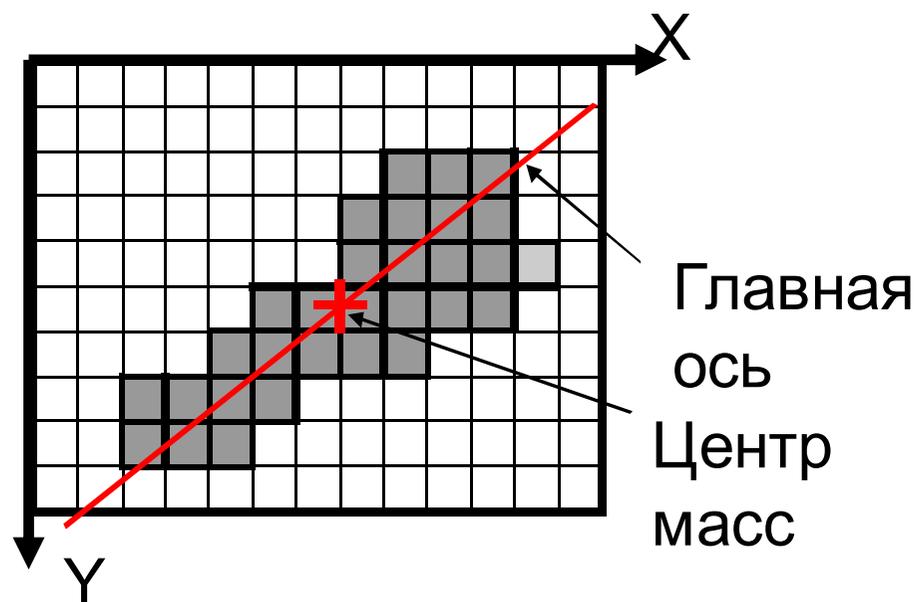




Ориентация главной оси инерции

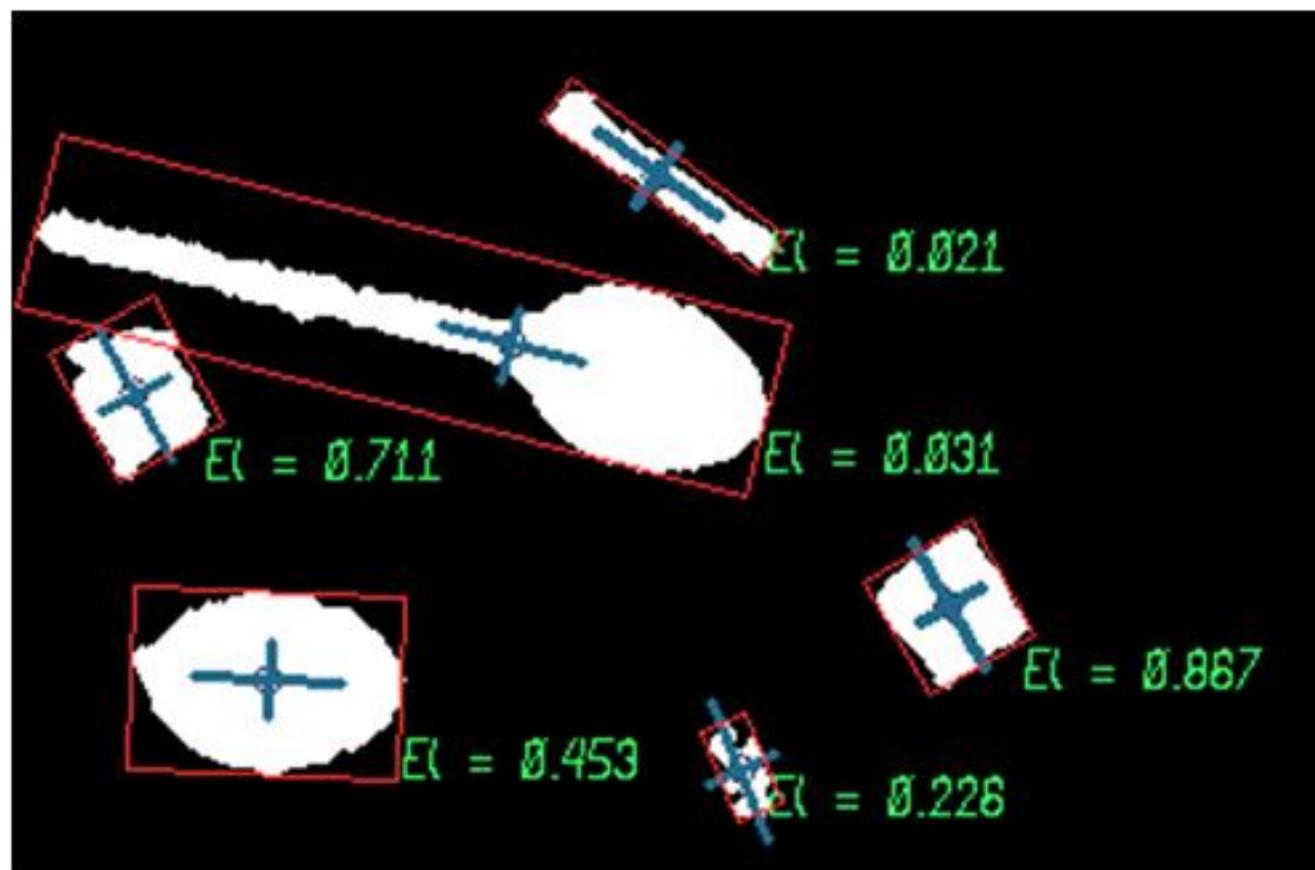
Не является инвариантной к повороту, но в ряде случаев предоставляет полезную информацию об ориентации объекта:

$$\theta = \frac{1}{2} \arctan \left(\frac{2 m_{11}}{m_{20} - m_{02}} \right)$$





Пример



Вычисленные значения признаков



Фотометрические признаки

Для каждой области можно подсчитать некий набор простейших числовых характеристик:

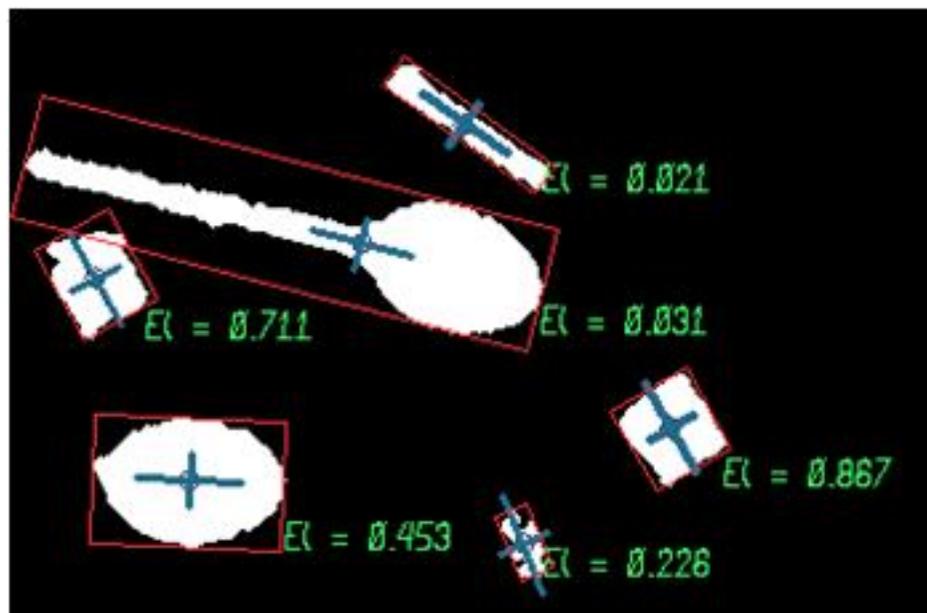
- Средняя яркость
- Средний цвет (если изображение цветное)
- Распределение яркости или цвета
- Признаки текстуры
 - Про распределение цветов и текстурные признаки будем говорить отдельно дальше

Разумеется, все это считается по исходному, а не бинарному изображению!



Как анализировать признаки

- Пример – ложки и сахар





Как анализировать признаки

- Как воспользоваться признаками для классификации?
 - Подобрать диапазоны значений для разных классов вручную, экспериментально
(может быть весьма трудоемко)
 - Подобрать диапазоны значений графически
(нужна база для тренировки, трудно, если признаков много)
 - Обучить классификатор с помощью машинного обучения

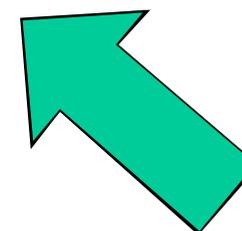
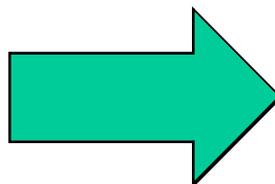
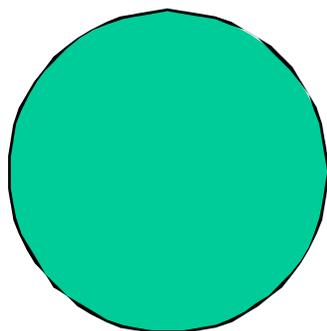
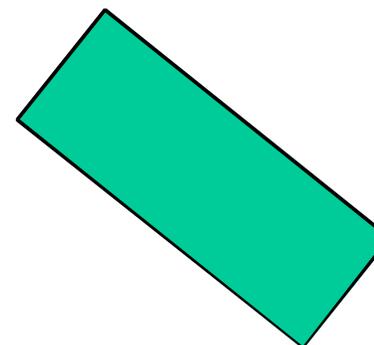
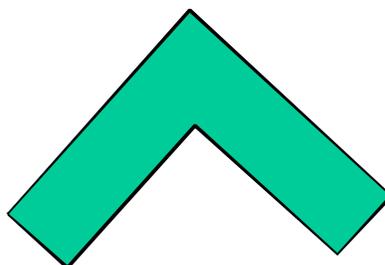
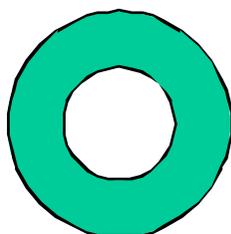
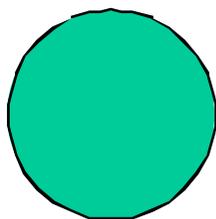


Ручной подбор

- Из общих соображений:
 - Ложки более вытянутые, чем сахарные кусочки
 - Ложки больше чем сахарные кусочки
 - Сахарные кусочки квадратные
 - Области появляющиеся из-за шума обычно небольшие и неквадратные
- Пытаемся сконструировать решающее правило, проверяем экспериментально
- Может быть весьма утомительно



Примеры





Графический анализ

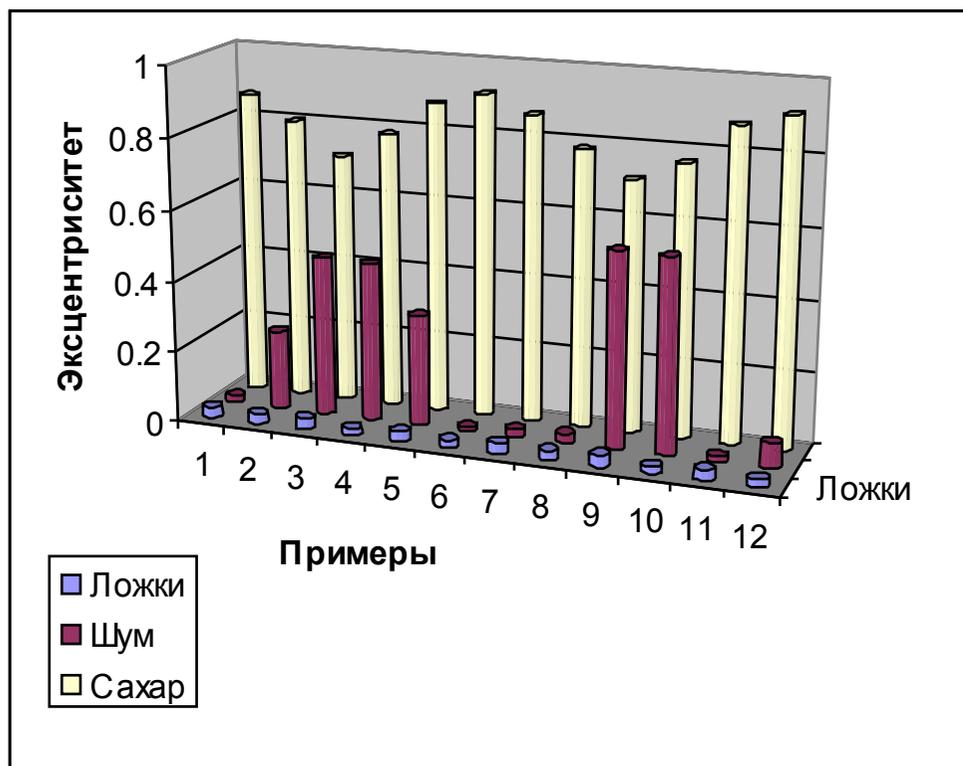
- Собрать тренировочную базу изображений
 - Где только ложки
 - Где только сахар
 - Где только шум

Как получить такие? Да просто закрасить все остальное.
- Брать признаки и строить графики



Графический анализ

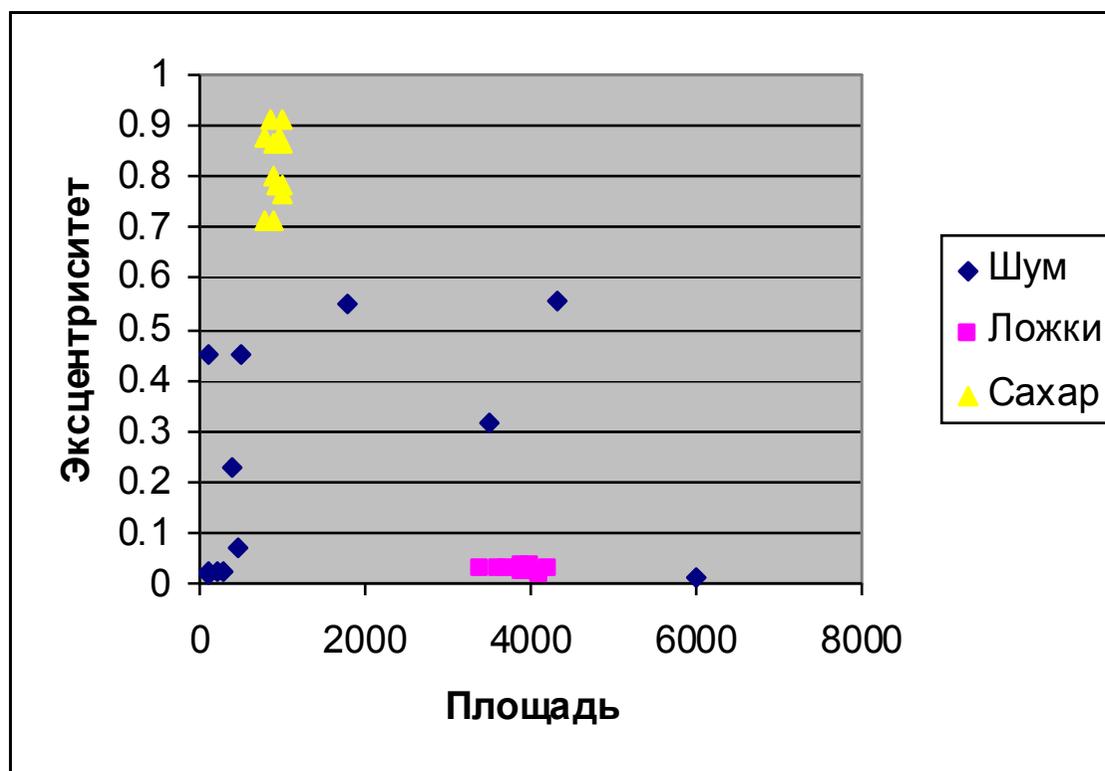
- Диаграмма распределения эксцентриситета (проблема – не получается отличить шум от ложек)





Графический анализ

- График распределения эксцентриситета и площади (гораздо лучше – можем подобрать значения порогов)



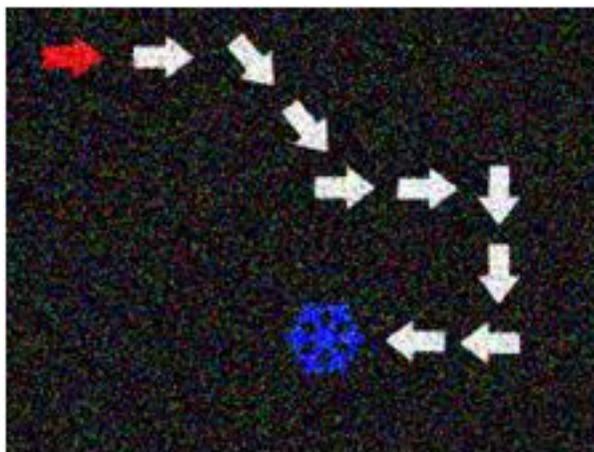


Машинное обучение

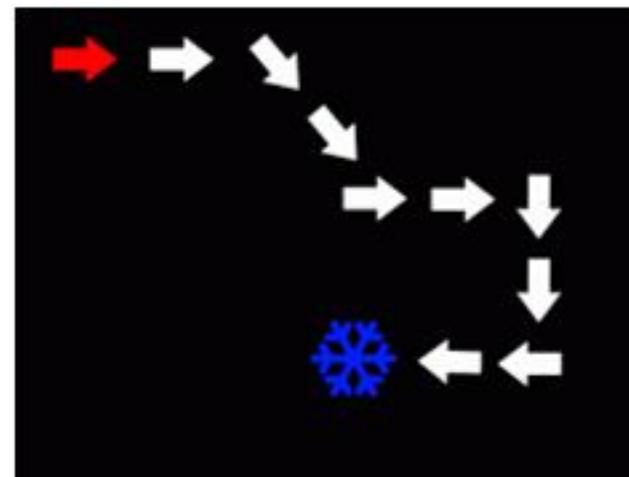
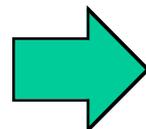
- Причина бурного развития компьютерного зрения в последние годы.
- Требуются большие коллекции примеров для обучения.
- Посмотрим один из методов на 4ой лекции



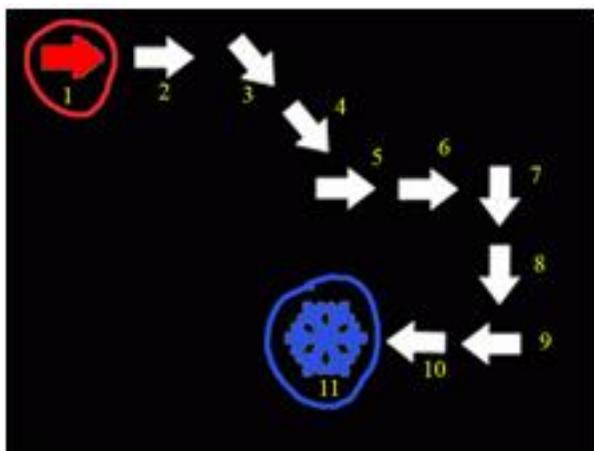
Схема простого алгоритма



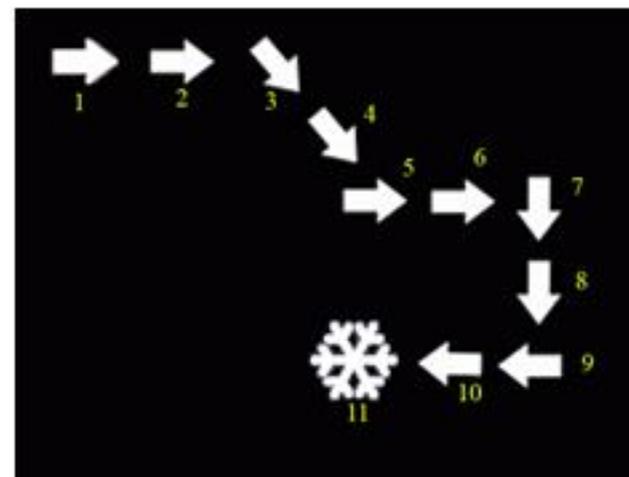
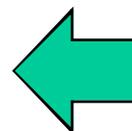
Предобработка
изображения



Сегментация
изображения



Вычисление
признаков
сегментов и
классификация





Резюме лекции

- Сегментация изображения позволяет работать не со всем изображением в целом, а с отдельными значимыми фрагментами
- Сегменты могут выбираться по критериям однородности по яркости, цвету, текстуре и по комбинации этих признаков
- В отдельных случаях мы можем решить задачу распознавания, анализируя геометрические и фотометрические признаки сегментов