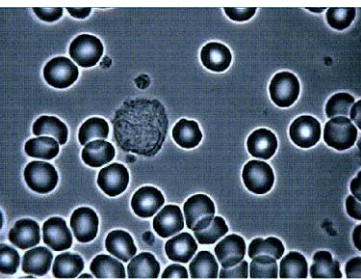


Сегментация изображений

Лекция 8

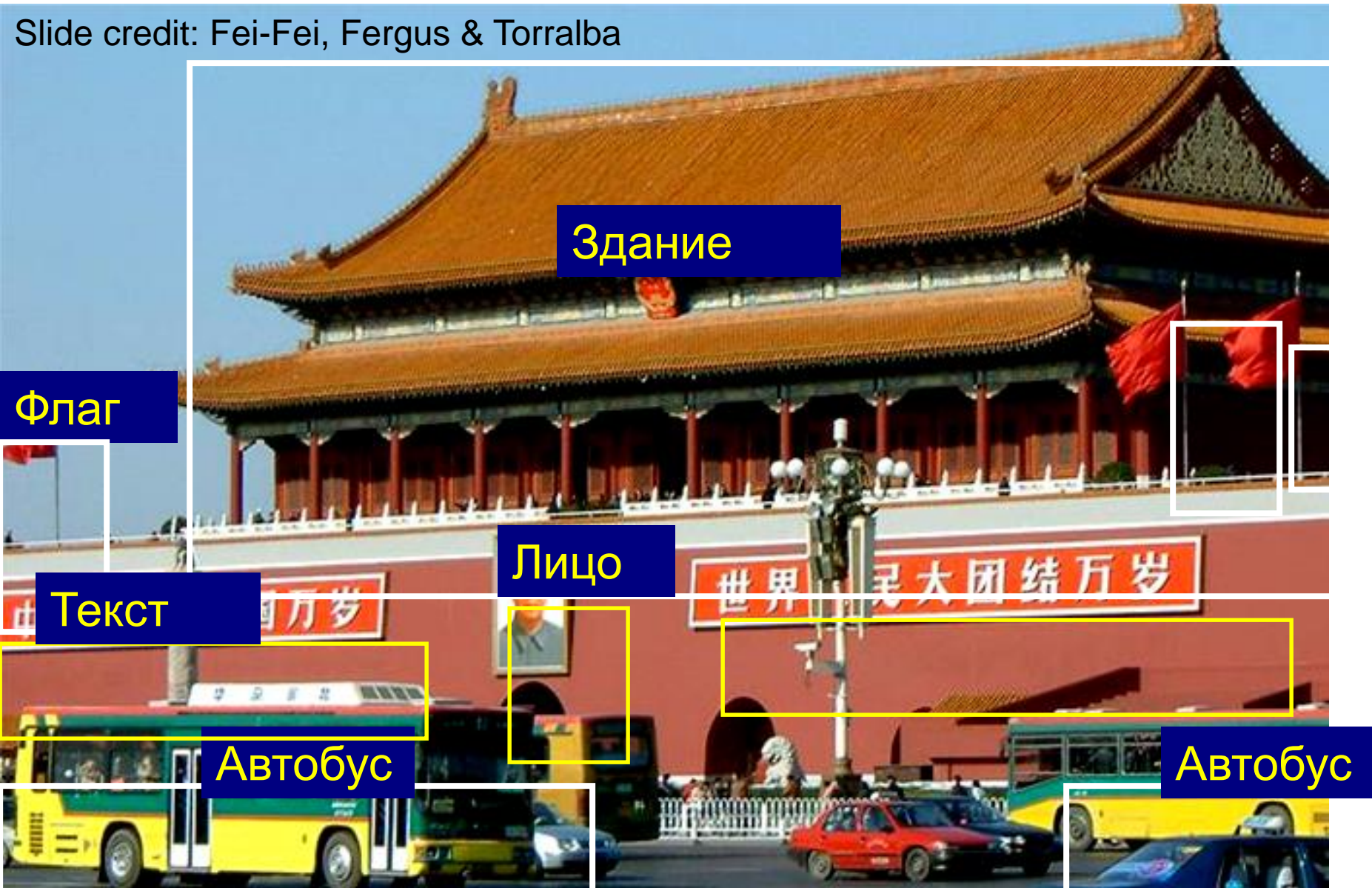


Сколько сегментов?



Такие сегменты?

Slide credit: Fei-Fei, Fergus & Torralba



Грубая сегментация

Небо

Строения

Автомобиль

Автомобиль

Где сегменты?

Голубое

наклонная

Ветер
справа
налево

Нежесткий,
движется

Жесткий,
движется

Мао

Жесткий,
движется

горизонтальный

Определение сегментации

- «Жесткая» сегментация
 - Разбиение изображения на неперекрывающиеся области, однородные по некоторому признаку и покрывающие всё изображение
- Формально:
 - Разбиение изображения на набор областей

$$S = \{S_i\}, i = \overline{1, N}$$

$$I = \bigcup_{i=1..N} S_i$$

$$\forall i, j = \overline{1, N} : i \neq j \quad S_i \cap S_j = \emptyset$$

$$\forall i = \overline{1, N}, P(S_i) = \text{истина}$$

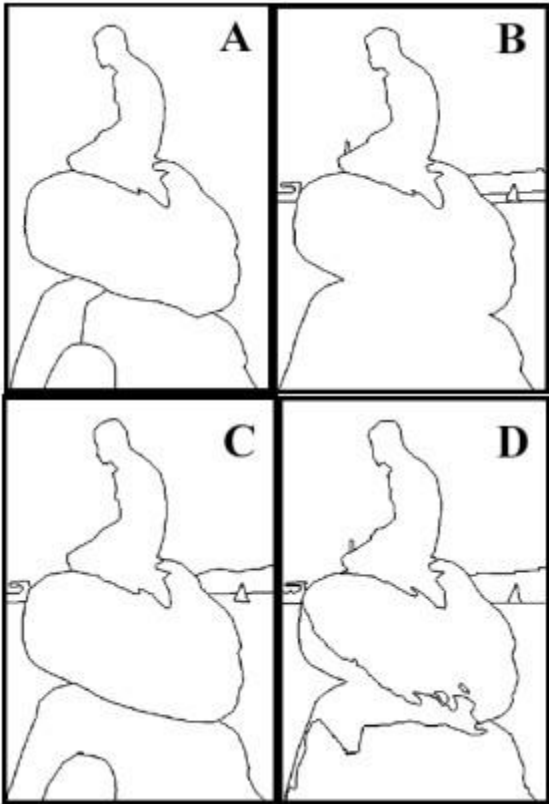
$$\forall i, j = \overline{1, N} : i \neq j \quad P(S_i \cup S_j) = \text{ложь}$$

Цель сегментации изображений

- Декомпозиция изображения на части
- Изменение формы описания изображения

Вопрос. Нужна ли информация из предметной области?

Варианты сегментации изображения. Какой верный?



Berkeley Segmentation Dataset

Оценка качества работы методов сегментации

- Целостность и однородность по некоторому признаку
- Отличие признака для смежных областей
- Отсутствие мелких отверстий внутри
- Гладкие границы

<http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench>

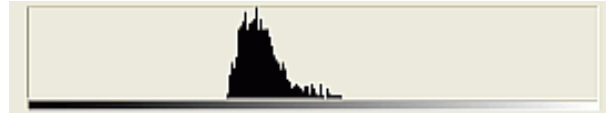
более 1000 изображений, отсегментированных вручную 30 разными людьми [18]

Критерии «похожести» пикселей

- По расстоянию
- По яркости
- По цвету
- По текстуре

Критерий однородности

- Гистограмма содержит не больше 1 значительного пика



- Отклонение любого пикселя от средней яркости $< T_{avg}$

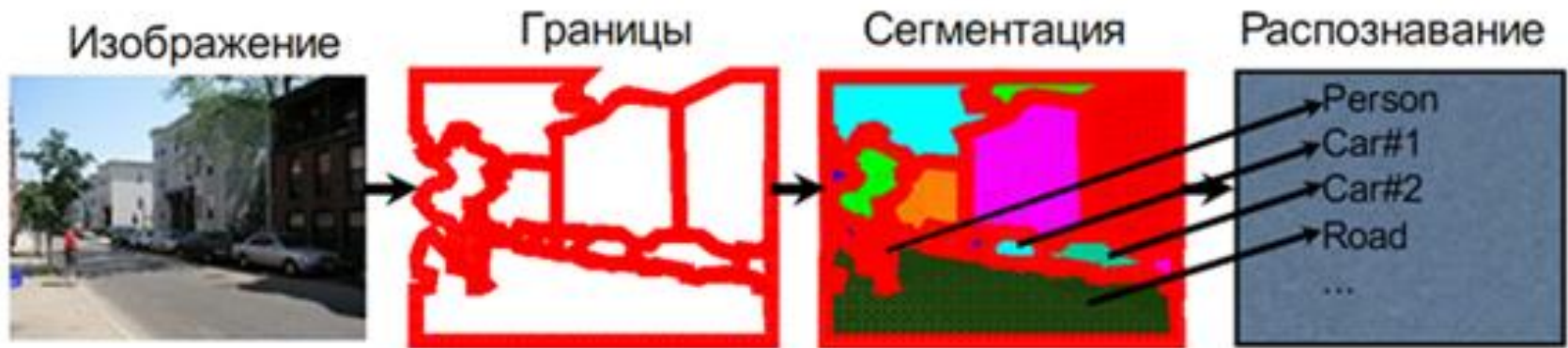
$$\forall p \in S \quad \left| I(p) - \frac{1}{N} \sum_{q \in S} I(q) \right| < T_{avg}$$

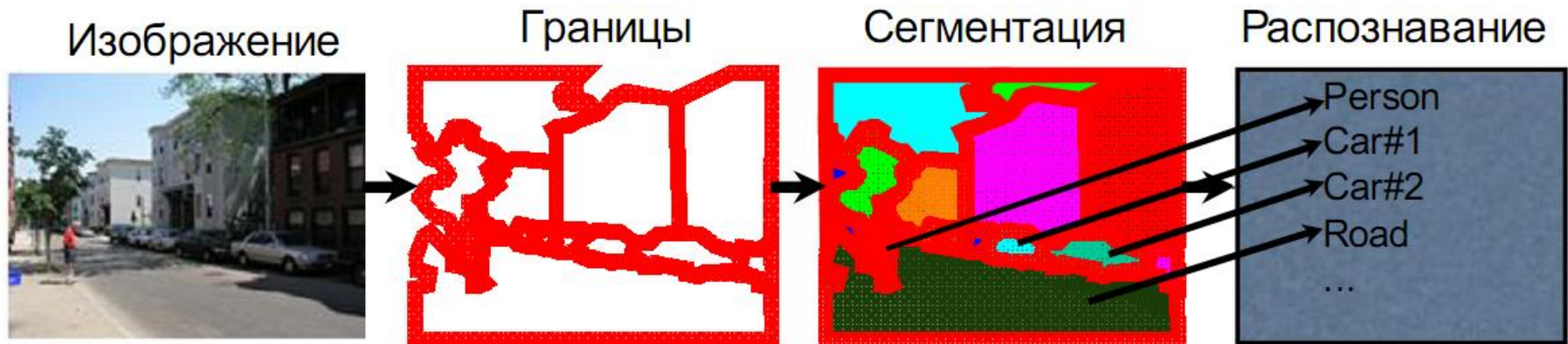
- Разница между соседними пикселями $< T_{diff}$

$$\forall p \in S, \forall q \in N(p) \quad |I(p) - I(q)| < T_{diff}$$

- «Слабая» граница между регионами (только для слияния)

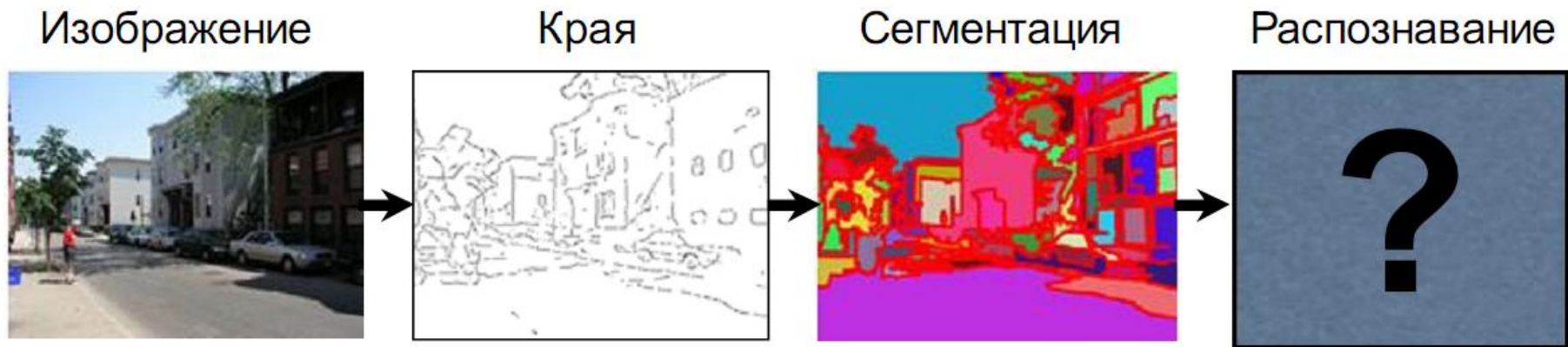
План





Теория ↑

Практика ↓



Суперпиксели

- Предобработка перед высокоуровневым анализом



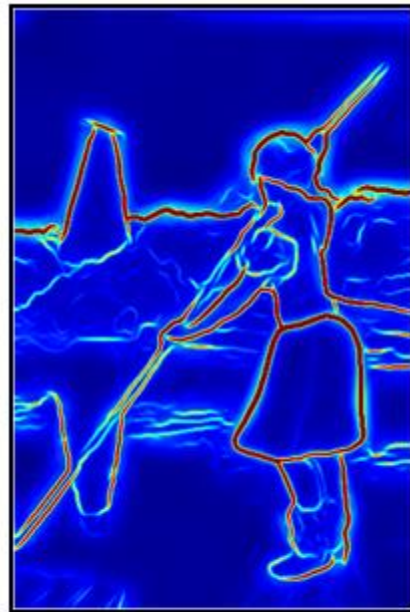
Два типа сегментации

- Автоматическая
 - Сегментация производимая без взаимодействия с пользователем
 - Картинка на входе, регионы на выходе
- Интерактивная
 - Сегментация, управляемая пользователем, допускающая и/или требующая ввода дополнительной информации
 - Пример – «волшебная палочка» в Photoshop

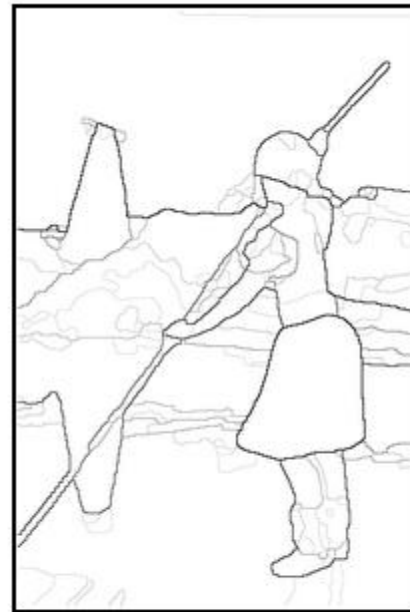
Автоматическая сегментация



Original Image



globalPb



Ultrametric Contour Map



Automatic Segmentation

Задачи автоматической сегментации делятся на два класса

- Выделение областей изображения с известными свойствами
- Разбиение изображения на однородные области

Семейства методов

- **Основанные на формировании однородных областей**
 - **без пространственных связей**
 - с учетом пространственных связей
- Основанные на поиске краев (края и области)
 - Canny
 - Рb-детектор краёв
- Методы на графах
 - Normalized cut
 - «Эффективный метод» Felzenszwalb & Huttenlocher
- Энергетические методы
 - Snakes
 - Методы уровня
 - ТурбоПиксели (TurboPixels)

Сегментация через поиск однородных областей

- **Сегментация без учёта пространственных связей**
 - Пороговая фильтрация
 - Кластеризация по цвету
 - K-средних
 - сдвиг среднего (Mean shift) и развитие
- Сегментация с учётом пространственных связей
 - Разрастание областей (region growing)
 - Слияние/разделение областей (region merging/splitting)
 - Методы водораздела

Этапы кластеризации цветового пространства

- Чтобы свести задачу сегментации к задаче кластеризации, достаточно
 - задать отображение точек изображения в некоторое пространство признаков
 - и ввести метрику (меру близости) на этом пространстве признаков
- Применение методов кластерного анализа
- Обычно после кластеризации точек - выделение связанных компонент

Кластеризация цветового пространства

Наиболее популярный метод кластеризации, используемый для сегментации изображений k -средних



6 кластеров

Методы кластеризации плохо работают на зашумленных изображениях

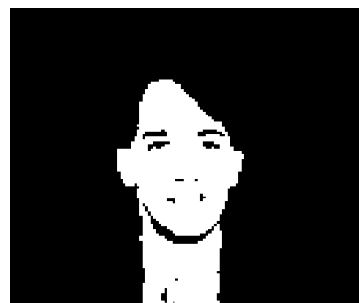
Сегментация по цвету



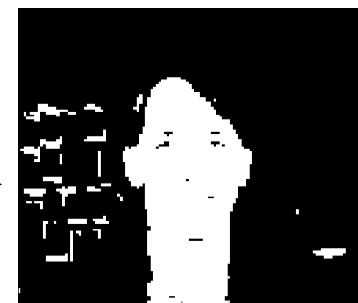
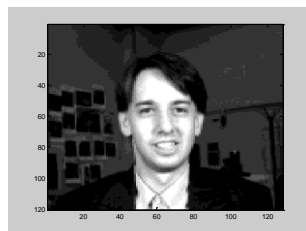
Сравнение k-средних с порогом по средней яркости

Чем отличается сегментация с помощью k-средних на 2 кластера от простейшей пороговой бинаризации по средней яркости изображения?

Пример:



k-средних



Порог по средней яркости

В причинах предлагается разобраться самостоятельно

Семейства методов

- **Основанные на формировании однородных областей**
 - без пространственных связей
 - **с учетом пространственных связей**
- Основанные на поиске краев (края и области)
 - Canny
 - Рb-детектор краёв
- Методы на графах
 - Normalized cut
 - «Эффективный метод» Felzenszwalb & Huttenlocher
- Энергетические методы
 - Snakes
 - Методы уровня
 - ТурбоПиксели (TurboPixels)

Сегментация через поиск однородных областей

- Сегментация без учёта пространственных связей
 - Пороговая фильтрация
 - Кластеризация по цвету
 - К-средних
 - сдвиг среднего (Mean shift) и развитие
- **Сегментация с учётом пространственных связей**
 - Разрастание областей (region growing)
 - Слияние/разделение областей (region merging/splitting)
 - Методы водораздела

Разрастание регионов (Region growing)

- Простая идея – начиная с некоторого “семени” обходить пиксели и присоединять к области пока выполняется условие однородности

Критерии присоединения к региону

- Близость точки к центру региона
- Близость к соседней точке, присоединенной к региону на предыдущем шаге
- Близость по некоторой статистике региона
- Стоимость кратчайшего пути от точки до центра региона

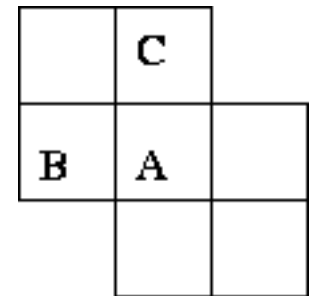
Существуют различные стратегии выбора зерен и выращивания регионов [14, 15, 16, 17]

Разрастание регионов

1. if $|I(A) - Cl_{avg}(B)| > \delta$ and $|I(A) - Cl_{avg}(C)| > \delta$ -
создаем новую область, присоединяем к ней пиксел А
2. if $|I(A) - Cl_{avg}(B)| \leq \delta$ xor $|I(A) - Cl_{avg}(C)| \leq \delta$ -
добавить А к одной из областей
3. if $|I(A) - Cl_{avg}(B)| \leq \delta$ and $|I(A) - Cl_{avg}(C)| \leq \delta$:

a) $|Cl_{avg}(B) - Cl_{avg}(C)| \leq \delta$ -
сливаем области В и С.

b) $|Cl_{avg}(B) - Cl_{avg}(C)| > \delta$ -
добавляем пиксел А к тому классу, отклонение от
которого минимально.



Алгоритм разрастания регионов 1

Среднее: 1

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

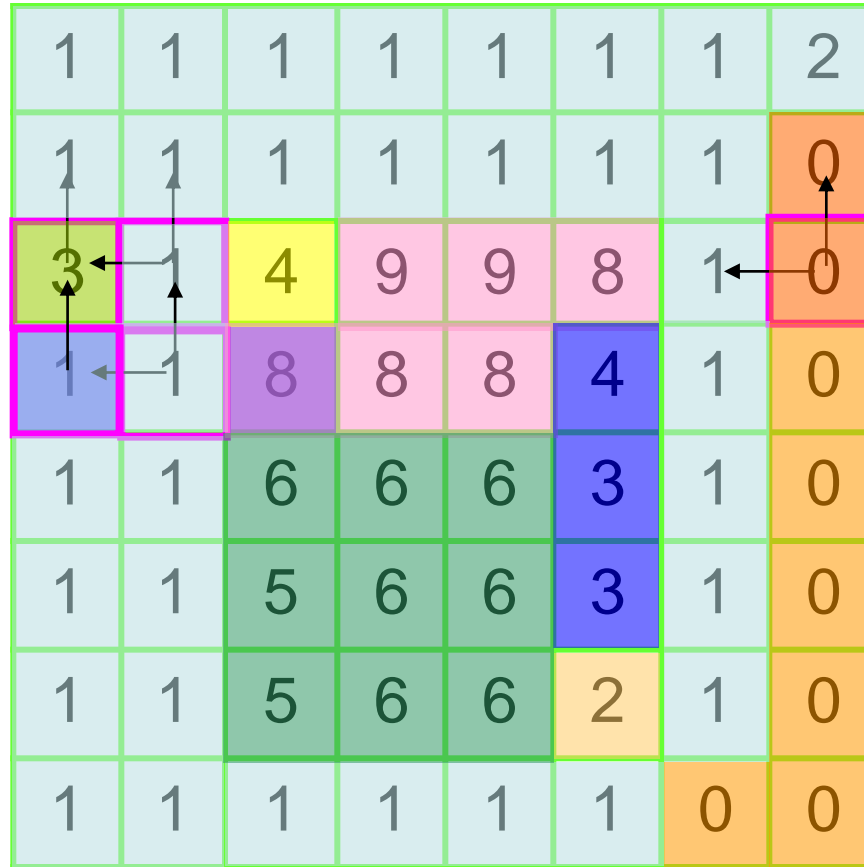
Пример $\delta = 1$

$$\forall p \in S \left| I(p) - \frac{1}{N} \sum_{q \in S} I(q) \right| \leq \delta$$

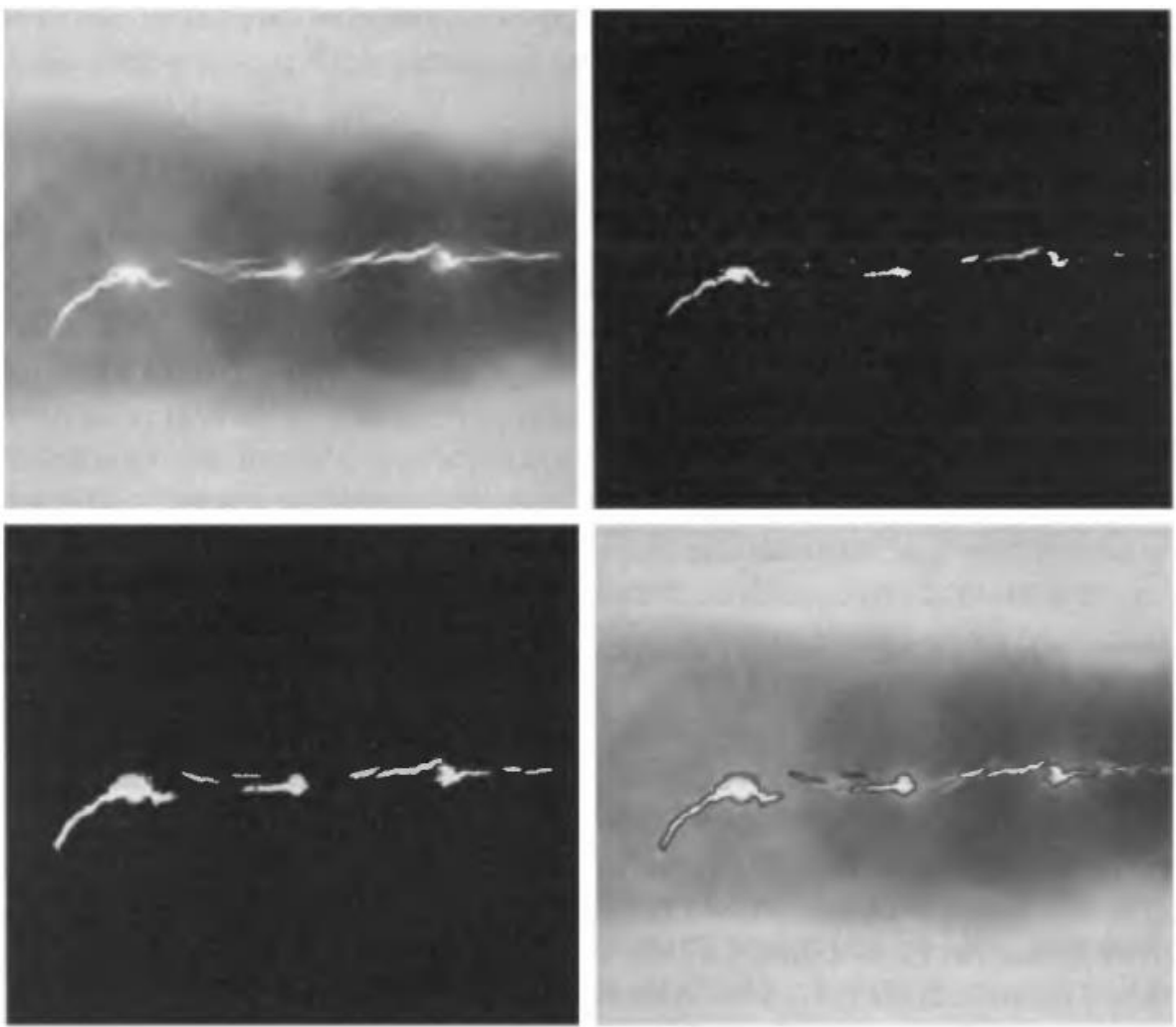
Алгоритм разрастания регионов 2

$$\forall p \in S \left| I(p) - \frac{1}{N} \sum_{q \in S} I(q) \right| \leq \delta$$

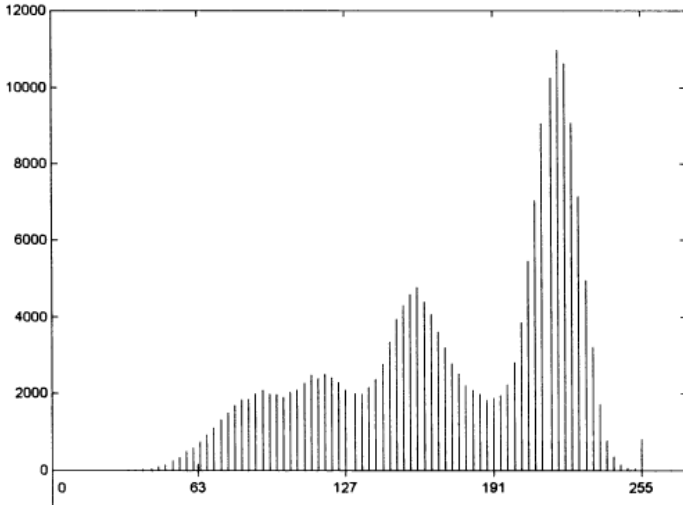
Пример $\delta = 1$



Сварочный шов с дефектами Центры кристаллизации



Гистограмма для сварного шва с дефектами



Границы дефектов, выявленных при сегментации

Результаты
выращивания
областей

Слияние областей

1. Первый шаг – каждый пиксель это отдельная область, поместить все области в стек
2. Пока стек не пуст
 - Взять область S из стека, для всех соседних областей S_i :
 - Проверить $S' = S \cup S_i$ на однородность
 - Если S' однородна -
 - Слить S и S_i , S' поместить в стек, S_i из стека удалить, перейти на 2
 - Если область не однородна
 - Пробуем другого соседа

Разделение областей

1. Первый шаг – всё изображение это одна область, поместить область в стек

2. Пока стек не пуст
 - Взять область S из стека
 - Проверить область на однородность
 - Если область неоднородна
 - разделить ее, новые области поместить в стек
 - Если область однородна
 - область больше не трогаем

Алгоритм разбиения (split)

Первое разбиение

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Алгоритм разбиения (split)

Второе разбиение

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

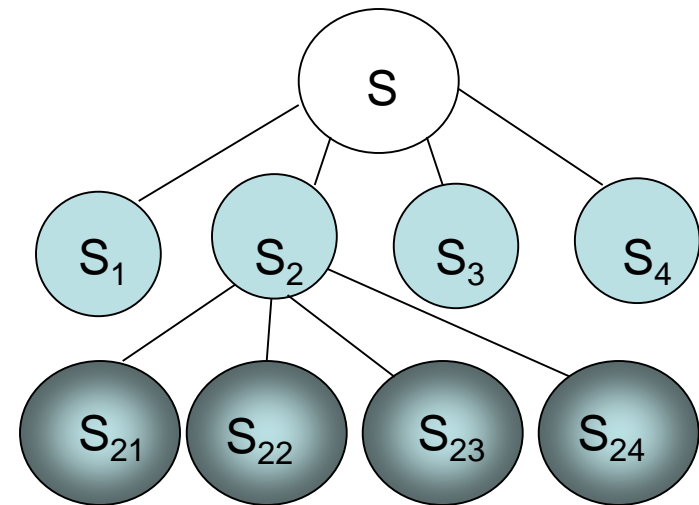
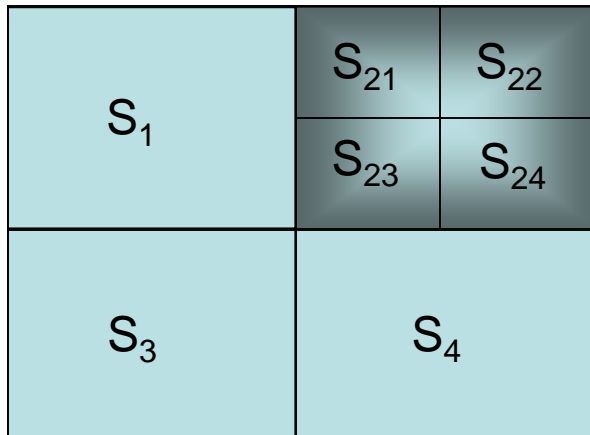
Алгоритм разбиения (split)

Третье разбиение

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Правило разделения областей 1

- Распространенный вариант – на 4 части, как квадродерево



Просто реализовать, но границы получившихся областей вряд ли будут соответствовать границам объектов

Недостатки алгоритмов разбиения и слияния

- Разбиение
 - Может дать слишком много регионов
 - Если использовать квадродерево, границы скорее всего будут неверны
- Слияние
 - Долго работает, если начинать с индивидуальных пикселей
- **Вывод — нужен комбинированный метод**

Алгоритм разбиения/слияния (split and merge)

- Идея [4, 6] :
 - Сначала провести разбиение (дробление, пересегментация) на небольшие однородные области
 - Обычно используется принцип квадродерева
 - Затем слить между собой те из них, которые вместе не нарушат требование однородности
 - Продолжать до тех пор, пока остаются регионы которые можно объединить

Третье разбиение из алгоритм разбиения (split)

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Слияние в алгоритме разбиения/слияния (split and merge)

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

Сравним результаты

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

разбиение/слияние

1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0
1	1	8	8	8	4	1	0
1	1	6	6	6	3	1	0
1	1	5	6	6	3	1	0
1	1	5	6	6	2	1	0
1	1	1	1	1	1	0	0

разрастание регионов

Сравнение результатов разделения-слияния и пороговой обработки



Исходное



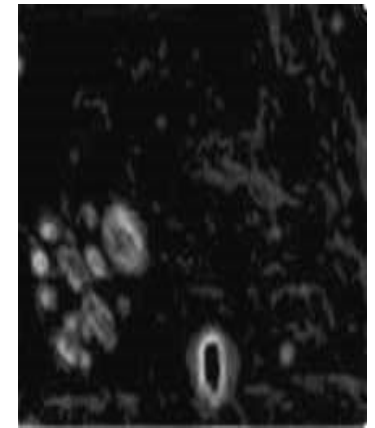
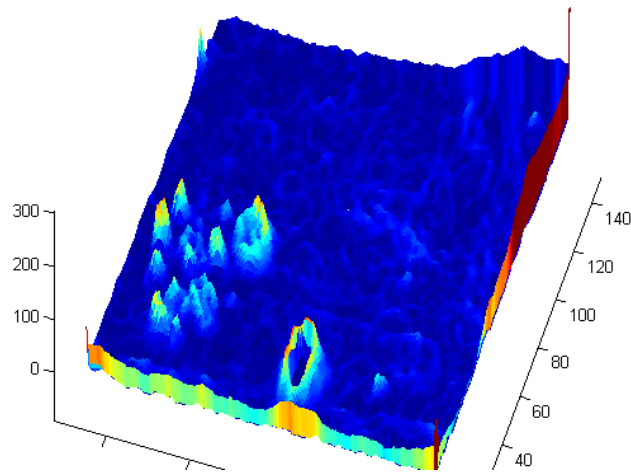
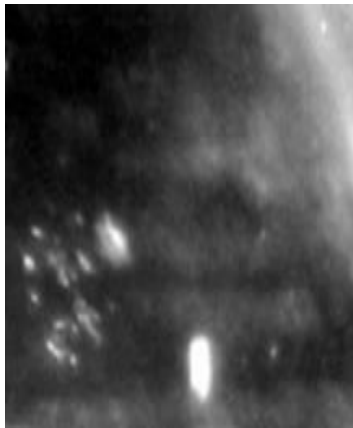
Разделение - слияние



Пороговая обработка

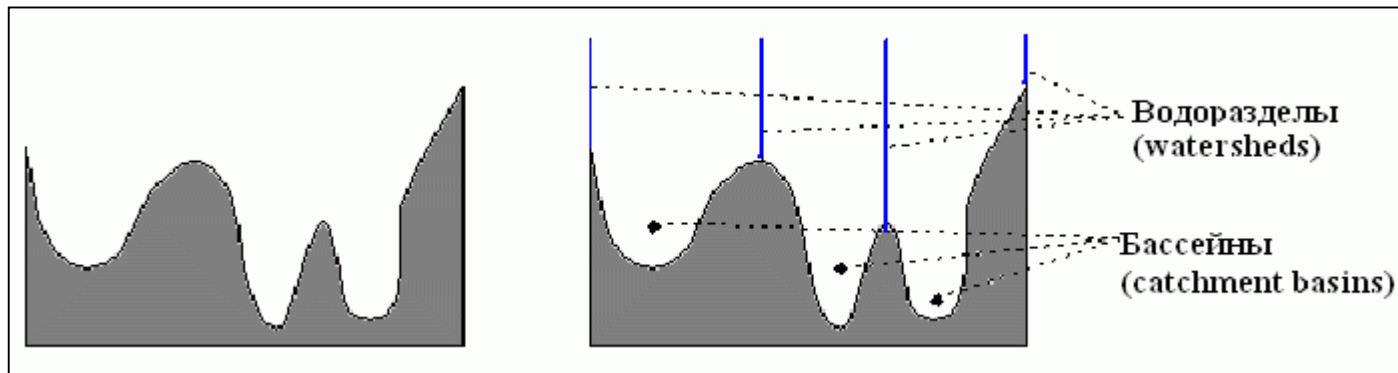
Алгоритмы водораздела (watershed)

Рассмотрим как карту высот ландшафта
– абсолютную величину градиента



На чём основаны алгоритмы водораздела

Область водораздела, бассейн (catchment basin): область, в которой поток из всех точек «стекает» к одной общей точке



Основные понятия алгоритмов водораздела

1. точки локального минимума
2. точки на склоне, с которых вода скатывается в один и тот же локальный минимум
3. точки на гребне, с которой вода с более-менее равной вероятностью скатывается в более чем один минимум

Бассейн (водосбор) локального минимума

Линии водораздела

Алгоритм «погружения» (immersion)

Начнем с самых «глубоких» (темных) пикселей
(они определяют начальные бассейны)

Для каждой яркости k :

Для каждой связной компоненты пикселей яркости k :

Если прилежит только к одному существующему бассейну

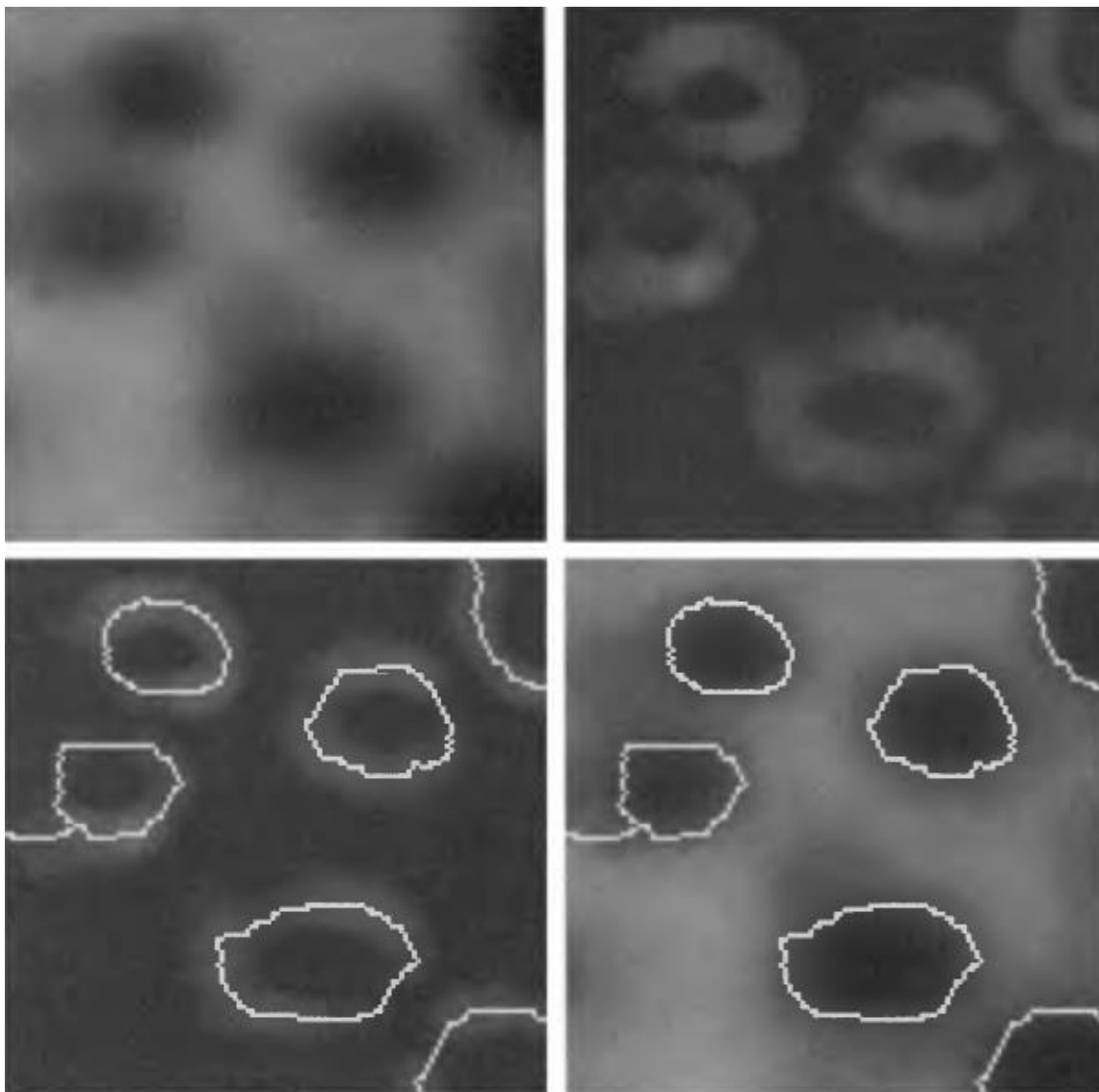
Добавить компоненту к бассейну

Иначе если прилежит более чем к одному существующему бассейну

Пометить как границу (водораздел)

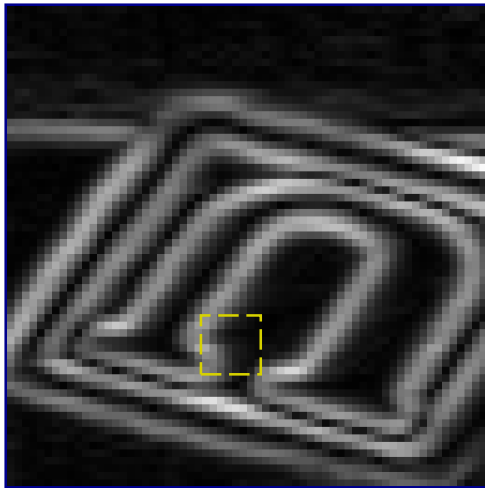
Иначе – создать новый бассейн

Иллюстрация работы алгоритма погружения



д.Бёше, СММ/Ecole
des Mines de Paris).

Алгоритм tobogganing

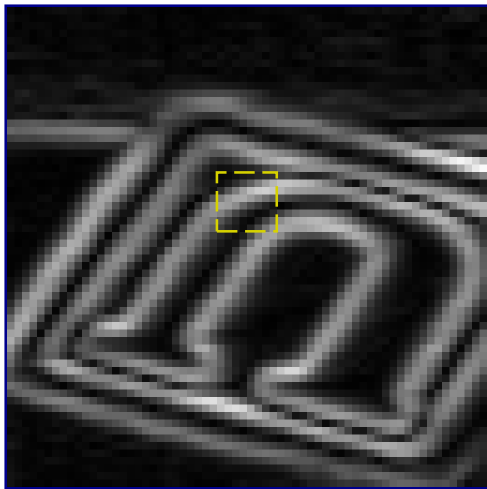


Идея

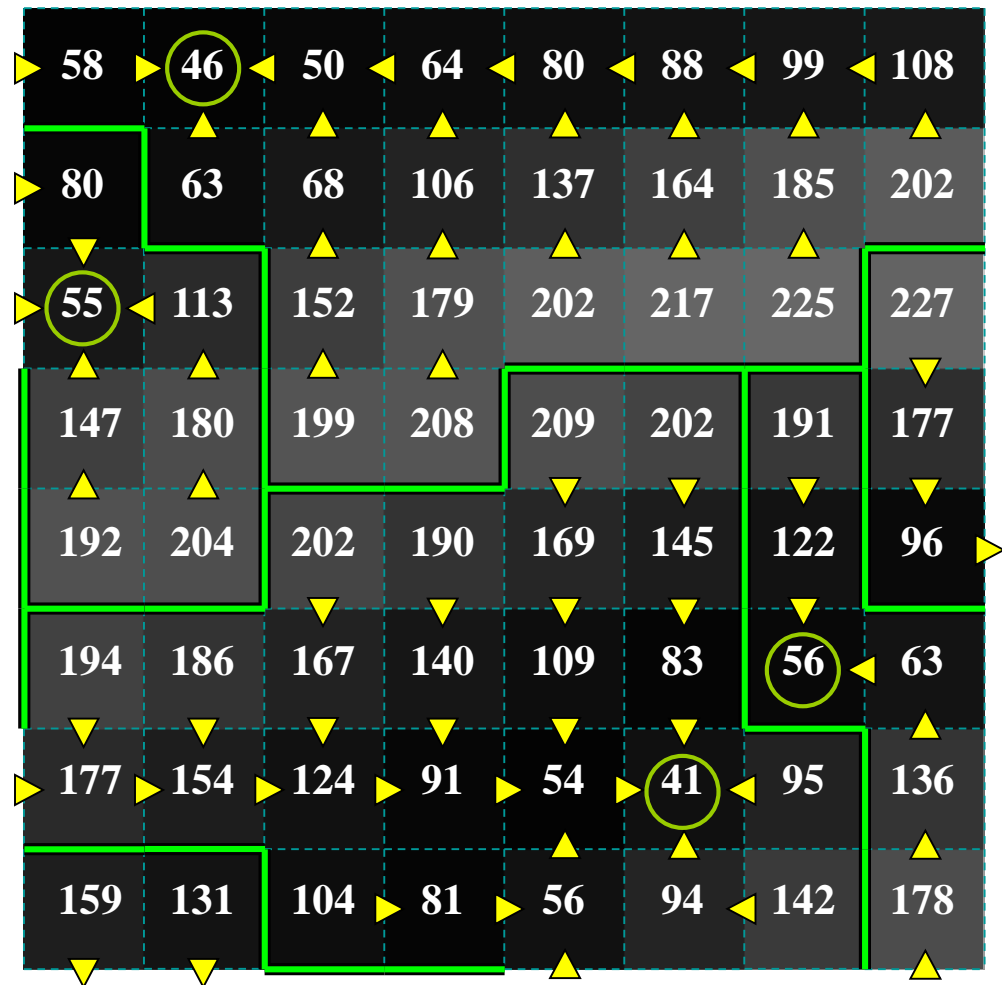
- Из каждого пикселя «спускаемся» в локальный минимум среди его соседей
- Спускаемся до тех пор, пока есть куда спускаться
- Пиксели «спустившиеся» в один минимум – одна область

58	46	50	64	80	88	99	108
80	63	68	106	137	164	185	202
55	113	152	179	202	217	225	227
147	180	199	208	209	202	191	177
				▼	▼		
192	204	202	190	169	145	122	96
		▼	▼	▼	▼		
194	186	167	140	109	83	56	63
▼	▼	▼	▼	▼	▼		
177	▶ 154	▶ 124	▶ 91	▶ 54	▶ 41	▶ 95	136
				▲	▲		
159	131	104	▶ 81	▶ 56	94	◀ 142	178

Сегментация алгоритмом tobogganing

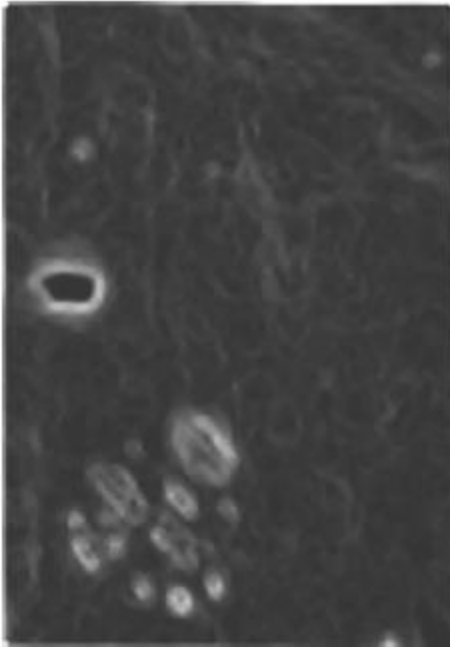


- Из каждого пикселя «спускаемся» в локальный минимум среди его соседей
- Спускаемся до тех пор, пока есть куда спускаться
- Пиксели «спустившиеся» в один минимум – одна область

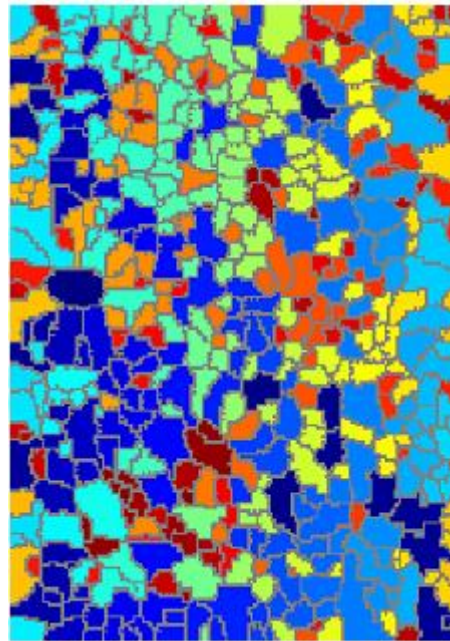


Проблемы алгоритмов водораздела

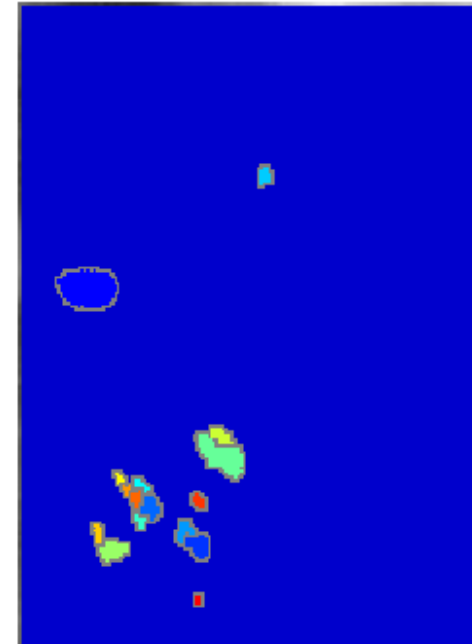
- Алгоритм дает множество небольших регионов
- Очень чувствителен к шуму – ищет все локальные минимумы
- Результат – избыточная сегментация



Абсолютная
величина
градиента

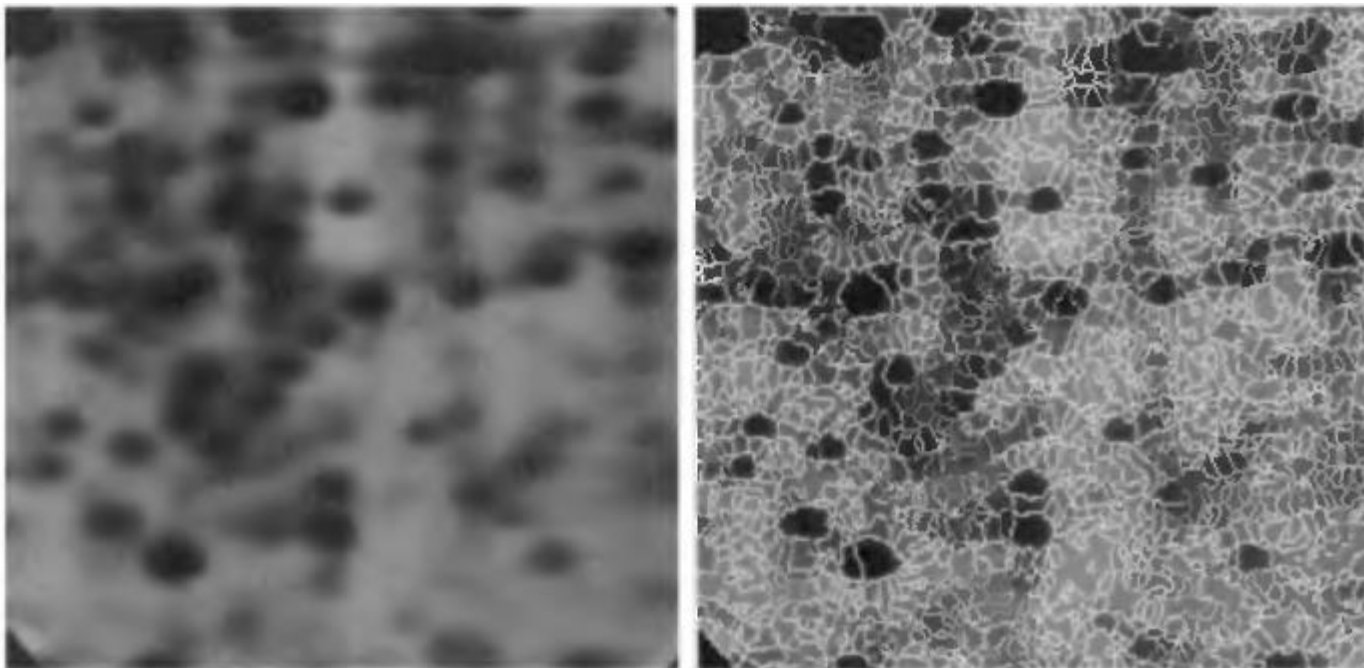


Сегментация по
данному
градиенту

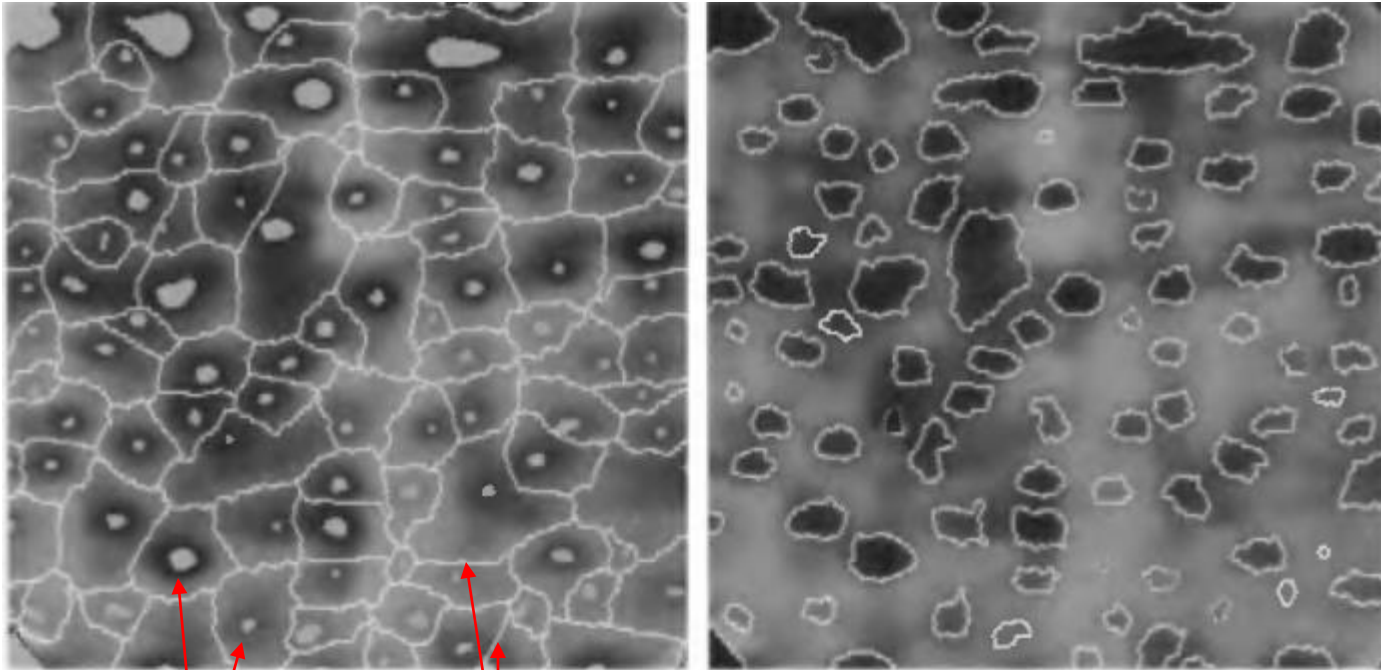


Градиент < 10
обращен в 0

Избыточная сегментация



Использование маркеров



Внутренние
маркеры

Внешние маркеры

Семейства методов

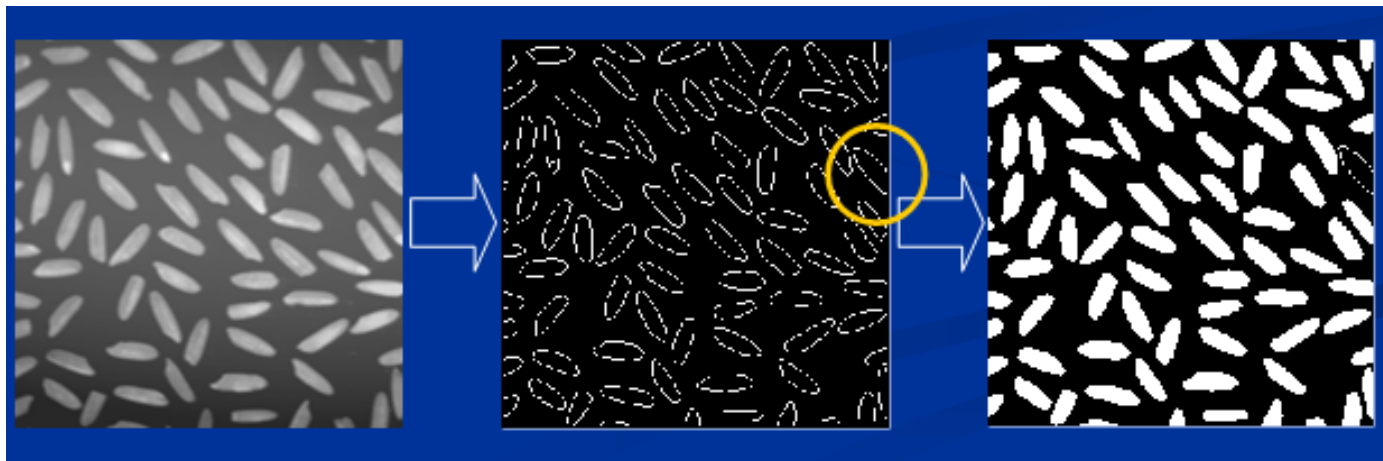
- Основанные на формировании однородных областей
 - без пространственных связей
 - с учетом пространственных связей
- **Основанные на поиске краев (края и области)**
 - Canny
 - Рb-детектор краёв
- Методы на графах
 - Normalized cut
 - «Эффективный метод» Felzenszwalb & Huttenlocher
- Энергетические методы
 - Snakes
 - Методы уровня
 - ТурбоПиксели (TurboPixels)

Методы, основанные на операторах выделения краёв



Сегментация посредством выделения контуров

1. Найдём все контура на изображении алгоритмом Canny;
2. Найдём все замкнутые контура;
3. «Внутренности» замкнутых контуров являются искомыми однородными областями;

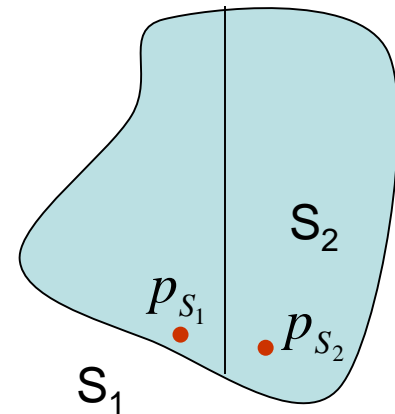


Алгоритм «фагоцита»

- Истаивание границ
 - Убирает слабые границы
- «Слабость границ» определяется по разности яркостей граничных пикселей

$$S(p_{S_1}, p_{S_2}) = |I(p_{S_1}) - I(p_{S_2})|$$

клетка способная захватывать и переваривать посторонние тела

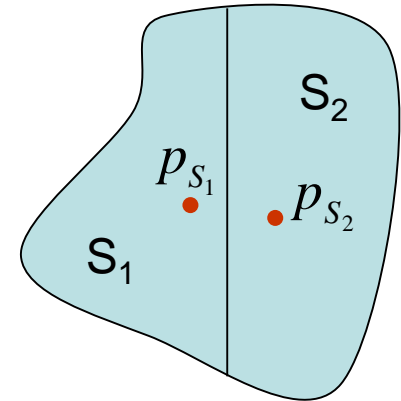


Алгоритм «фагоцита»

$$S(p_{S_1}, p_{S_2}) = |I(p_{S_1}) - I(p_{S_2})|$$

$$W(p_{S_1}, p_{S_2}) = \begin{cases} 1 & S(p_{S_1}, p_{S_2}) > T \\ 0 & \text{иначе} \end{cases}$$

$$W(S_1, S_2) = \sum_{p_{S_1} \in R_1 \wedge p_{S_2} \in R_2} W(p_{S_1}, p_{S_2})$$



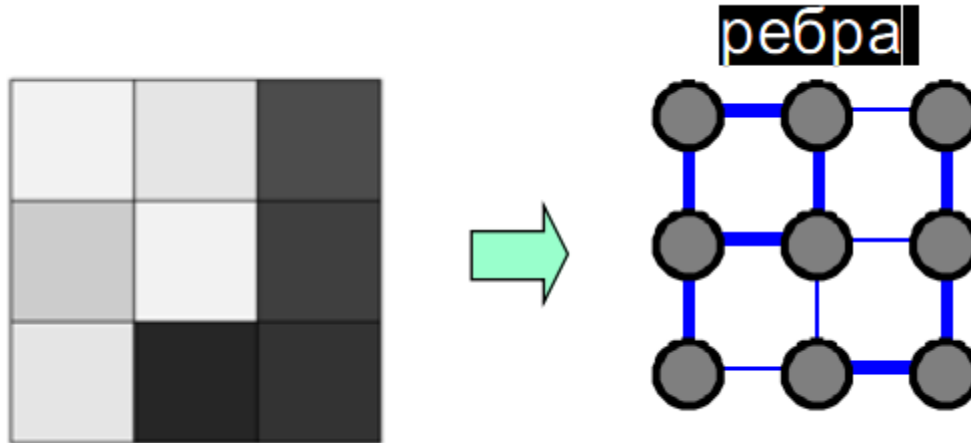
Слить две области если:

$$\frac{W(\text{граница})}{\text{кол - во Точек границы}} < T_3, \quad 0 < T_3 \leq 1$$

Семейства методов

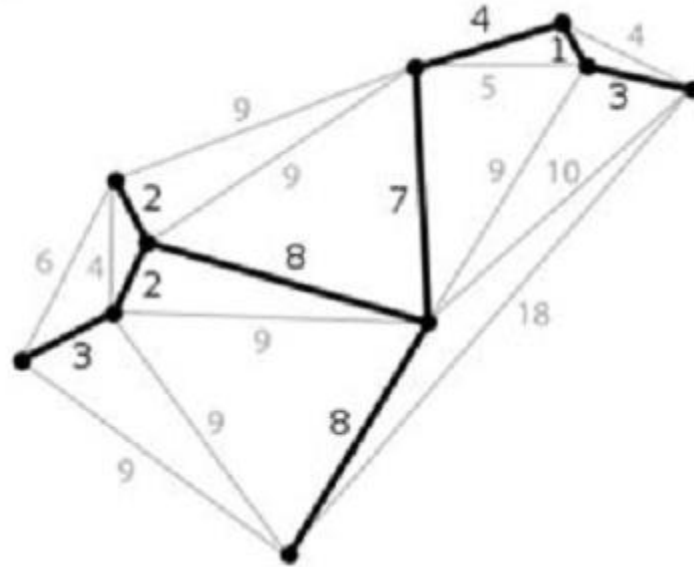
- Основанные на формировании однородных областей
 - без пространственных связей
 - с учетом пространственных связей
- Основанные на поиске краев (края и области)
 - Canny
 - Рb-детектор краёв
- **Методы на графах**
 - Normalized cut
 - «Эффективный метод» Felzenszwalb & Huttenlocher
- Энергетические методы
 - Snakes
 - Методы уровня
 - ТурбоПиксели (TurboPixels)

Представление в виде графа



- Изображение превращается во взвешенный неориентированный граф $G = (V, E)$
 - Вершины графа V – пиксели изображения
 - Ребра E – связи между соседними пикселями
 - Вес ребер пропорционален «похожести» пикселей

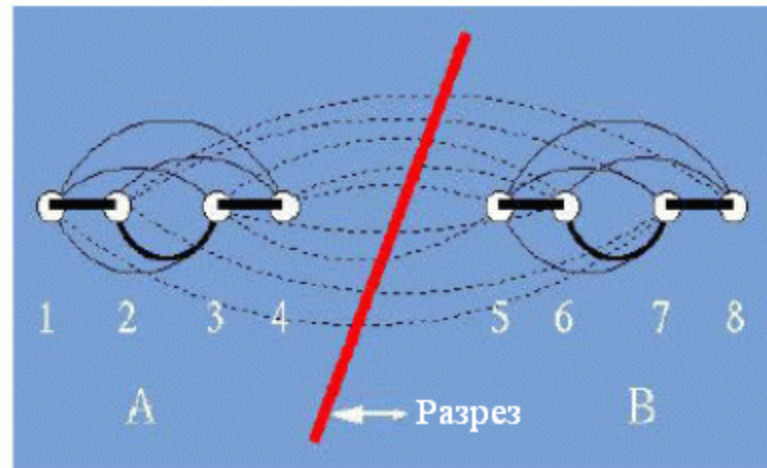
Minimum Spanning Tree (MST)



- Построим минимальное остовное дерево
 - Дерево, которое содержит все вершины графа, сумма весов ребёр которого минимальна
- MST для связанной компоненты (сегмента) содержит рёбра, обеспечивающие наибольшую «похожесть» пикселов в компоненте

Сегментация с помощью разрезов графа

- Создать граф
- Разрезать граф
- Каждую связную компоненту после разреза рассматривать как отдельную область

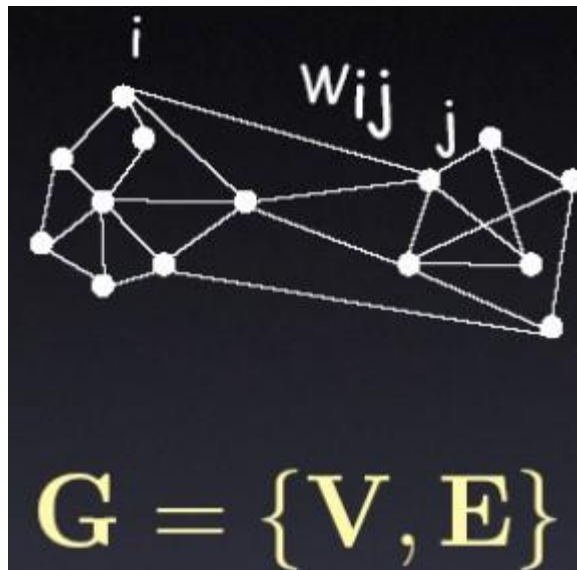


Разрез графа

- $G=(V,E)$
 - Непересекающиеся подмножества вершин A и B из V
 - Удаляем все ребра, связывающие A и B

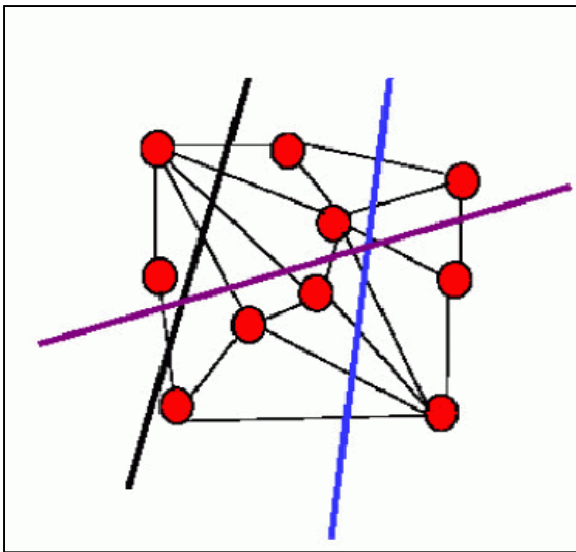
$$Cut(A, B) = \sum_{u \in A, v \in B} w(u, v)$$

- $Cut(A, B)$ – мера «силы связности» множеств A и B



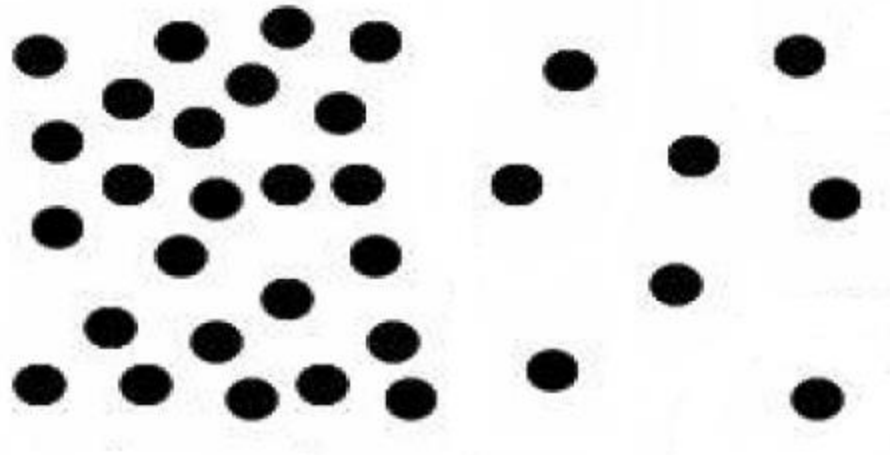
Минимальный разрез графа

- Если множества A и B не заданы заранее – разрезать граф можно по-разному:
 - **Минимальный разрез** – разрез, превращающий граф в несвязный, с минимальной суммой весов удаленных ребер



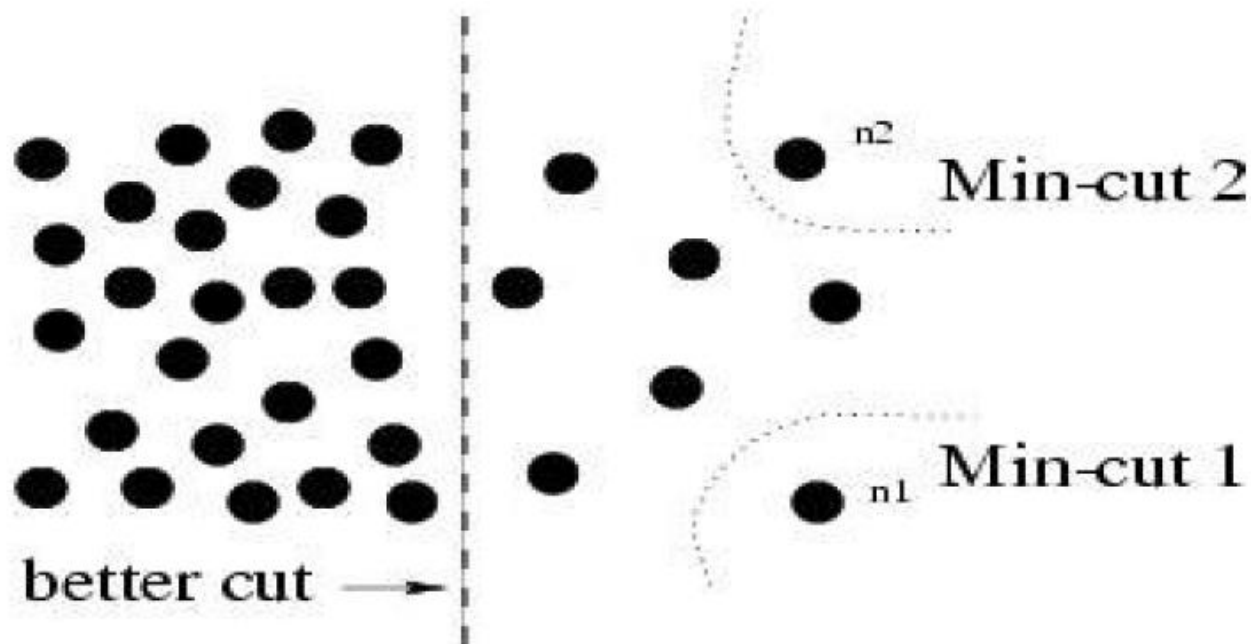
$$Cut(A, B) = \sum_{u \in A, v \in B} w(u, v)$$

А как разобьёшь ты?



Минимальный разрез хорош не всегда

- На данном рисунке вес ребер графа показан расстоянием между вершинами



Нормализованный разрез графа (Normalized cut)

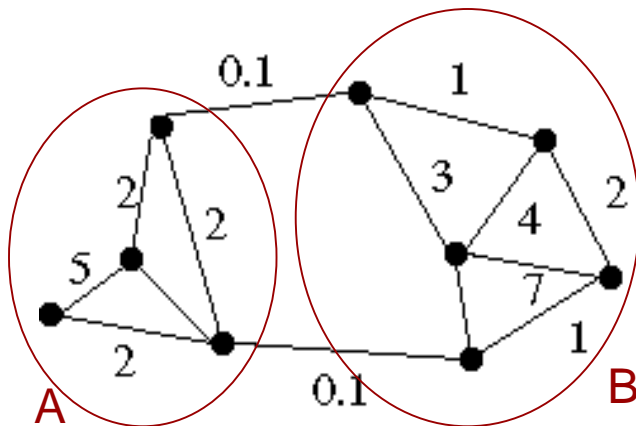
- **Другая мера разреза** – измеряет «похожесть» двух групп вершин, нормированную на «объем», занимаемый ими в графе

$$Ncut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)}$$

$$assoc(A, V) = \sum_{u \in A, t \in V} w(u, t)$$

Минимальный нормализованный разрез

- **Минимальный нормализованный разрез** – разрез, превращающий граф в несвязный, с минимальной величиной $NCut$
- *Как его найти?*



$$Ncut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)}$$

$$assoc(A, V) = \sum_{u \in A, t \in V} w(u, t)$$

Алгоритм сегментации с помощью normalized cuts (по Ши)

1. Задать граф на изображении

Вес ребра (по Ши)

$$w(i, j) = e^{\frac{-\|F(i) - F(j)\|_2}{\sigma_I}} * \begin{cases} e^{\frac{-\|X(i) - X(j)\|_2}{\sigma_X}} & \text{if } \|X(i) - X(j)\|_2 < r \\ 0 & \text{otherwise} \end{cases}$$

$$- F(i) = I(i)$$

$$- F(i) = [v, v \cdot s \cdot \sin(h), v \cdot s \cdot \cos(h)](i)$$

where h , s , and v are the HSV values

Алгоритм сегментации с помощью normalized cuts (по Ши)

2. Рассчитать матрицы W и D

$$D = \begin{bmatrix} \sum_j w(1, j) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sum_j w(N, j) \end{bmatrix}, \quad W = \begin{bmatrix} w(1,1) & \cdots & w(1,N) \\ \vdots & \ddots & \vdots \\ w(N,1) & \cdots & w(N,N) \end{bmatrix}$$

Можно вывести что:

$$\text{MinNcut}(G) = \min_y \frac{y^t (D - W)y}{y^t Dy}$$



NP-трудная задача

Если разрешить $y \in \mathfrak{R}$ задача сводится к задаче на собственные значения:

$$(D - W)y = \lambda Dy$$

Алгоритм сегментации с помощью normalized cuts

3. Решить задачу $(D-W)y = \lambda Dy$, найти вектора с наименьшими собственными значениями
4. По вектору со вторым наименьшим с.з. разрезать граф на две части
5. Рекурсивно разбить получившиеся области, если требуется

Jianbo Shi and Jitendra Malik "Normalized Cuts and Image Segmentation",
IEEE PAMI, 2000

Результат сегментации Ши



(a)



(b)



(c)



(d)



(e)



(f)

Результат сегментации Ши



Анализ алгоритма

- Метод требует хранения матрицы размером $n*n$, где n – число точек изображения, и потому в исходном виде неприменим к большим изображениям.
- Для данного метода предложены модификации [24, 25], позволяющие сократить сложность алгоритма и требования по памяти за счет аппроксимации матрицы расстояний. Такой подход дает выигрыш в скорости работы в 10-20 раз по сравнению с исходным методом.

«Эффективный метод»



- Идея: разница в интенсивности вдоль границы между областями должна быть существенной по сравнению с колебаниями интенсивности внутри одной из областей

P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. IJCV, 59(2):167–181, 2004.

Формализация

- «Внутренняя разница» в компоненте C :

$$Int(C) = \max_{e \in MST(C,E)} w(e) \quad \text{где } w(e) \text{ - вес ребра (мера различия двух пикселей)}$$

- Разница между областями:

$$Dif(C_1, C_2) = \min_{v_i \in C_1, v_j \in C_2, (v_i, v_j) \in E} w((v_i, v_j))$$

- Предикат присутствия границы между областями:

$$D(C_1, C_2) = \begin{cases} \text{true} & \text{if } Dif(C_1, C_2) > MInt(C_1, C_2) \\ \text{false} & \text{otherwise} \end{cases}$$

где $MInt$ – минимальные колебания интенсивности по областям

$$MInt(C_1, C_2) = \min(Int(C_1) + \tau(C_1), Int(C_2) + \tau(C_2))$$

и регуляризационный параметр $\tau(C) = k/|C|$

где $|C|$ - размер области, k – коэффициент регуляризации

Алгоритм

- **Схема «слияния регионов», но с учётом выбранных функций слияния:**
 - Сортируем все ребра по возрастанию веса
 - Инициализируем сегментацию максимальным разбиением (у каждого пиксела своя компонента)
 - Проходим по списку всех ребер (i,j)
 - Пусть C_i, C_j – компоненты, которым принадлежат вершины i, j
 - Тогда если не выполняется $D(C_i, C_j)$, тогда объединяем C_i и C_j

Сложность метода $O(N \log N)$

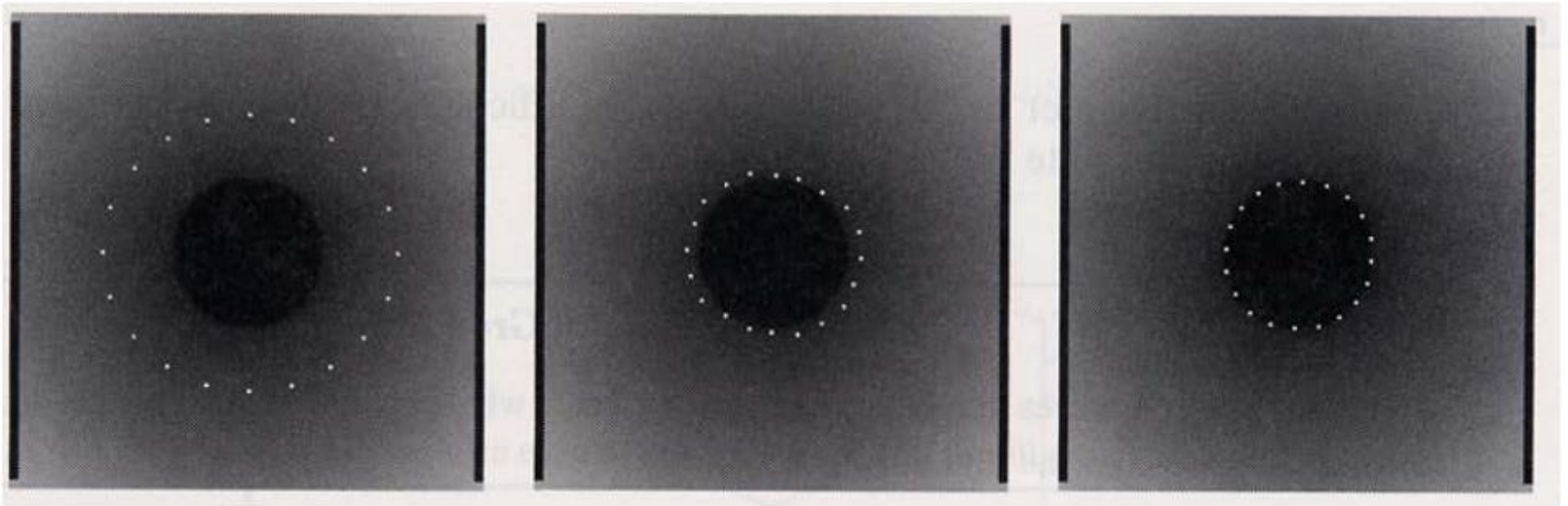
Результат



Семейства методов

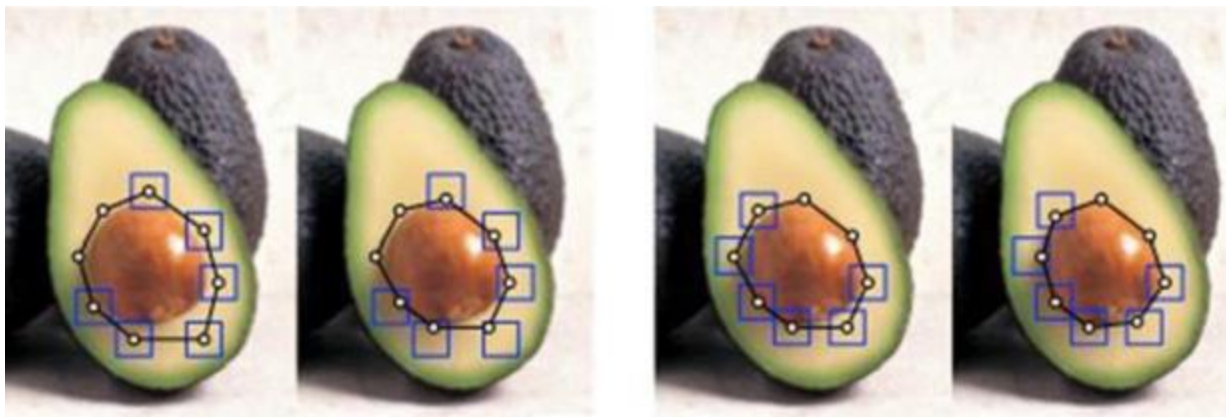
- Основанные на формировании однородных областей
 - без пространственных связей
 - с учетом пространственных связей
- Основанные на поиске краев (края и области)
 - Canny
 - Рb-детектор краёв
- Методы на графах
 - Normalized cut
 - «Эффективный метод» Felzenszwalb & Huttenlocher
- **Энергетические методы**
 - Snakes (активные контуры)
 - Методы уровня
 - ТурбоПиксели (TurboPixels)

Активный контур (Snakes)



M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. IJCV, 1988

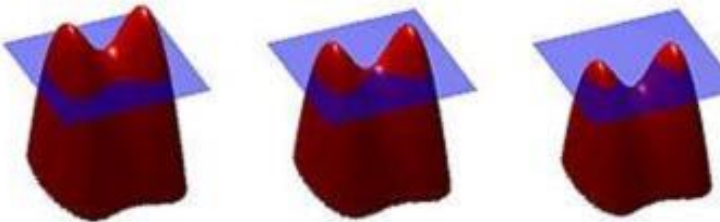
Вариант алгоритма Snakes



Метод фиксации уровня (Level set method)

- введён американскими математиками Стэнли Ошером и Джеймсом Сетьяном в 1980-е годы.
- Он стал популярным во многих дисциплинах, таких как
 - компьютерная графика,
 - обработка изображений,
 - вычислительная геометрия,
 - оптимизация,
 - вычислительная гидродинамика и
 - вычислительная биофизика

Level Sets (Линии уровня)



Визуализация представления контура с помощью функции вложения

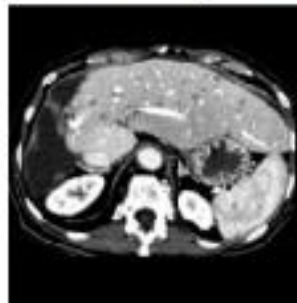
- Вместо явного представления контура зададим функцию вложения (embedding function) f
- $f > 0$ в области, $f < 0$ вне области
- Можем переформулировать задачу обновления контура как задачу обновления функции вложения

- Функция вложения позволяет моделировать изменения топологии области
- Работа с ней сводится к решению систем дифференциальных уравнений

Level set method

- Начальная маска. Это объект любой формы, находящийся внутри интересующего нас объекта. Чаще всего это прямоугольник, потому что его довольно просто задать.
- После определенного количества итераций функция модификации уровня меняется таким образом, чтобы ее пересечение с нулевой плоскостью давало интересующую нас кривую

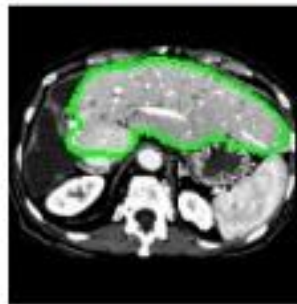
Входное изображение



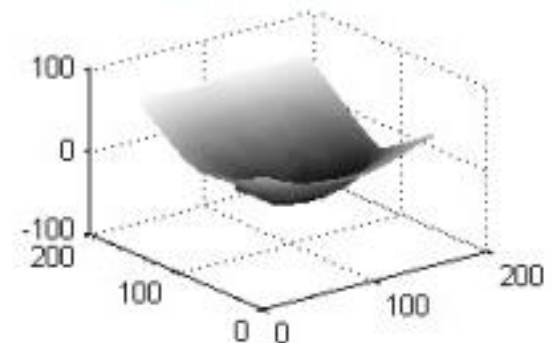
Маска



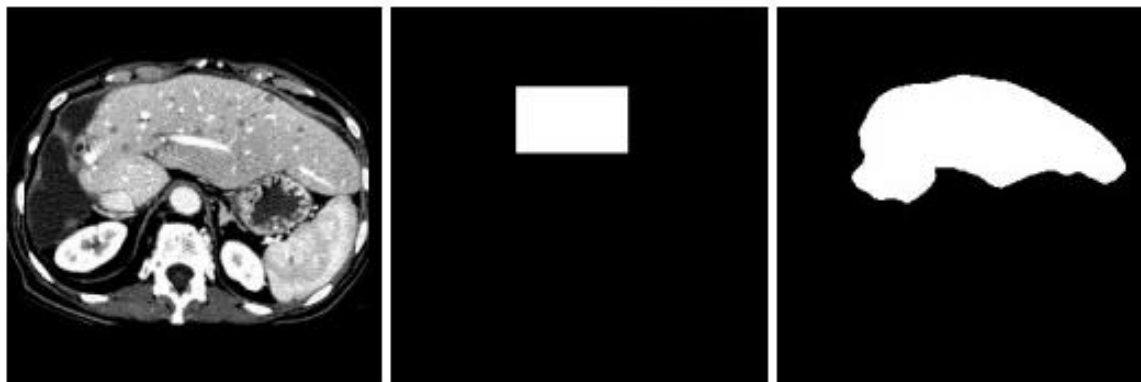
500 итераций



Функция фиксации уровня



Для медицинских изображений это работает вот так

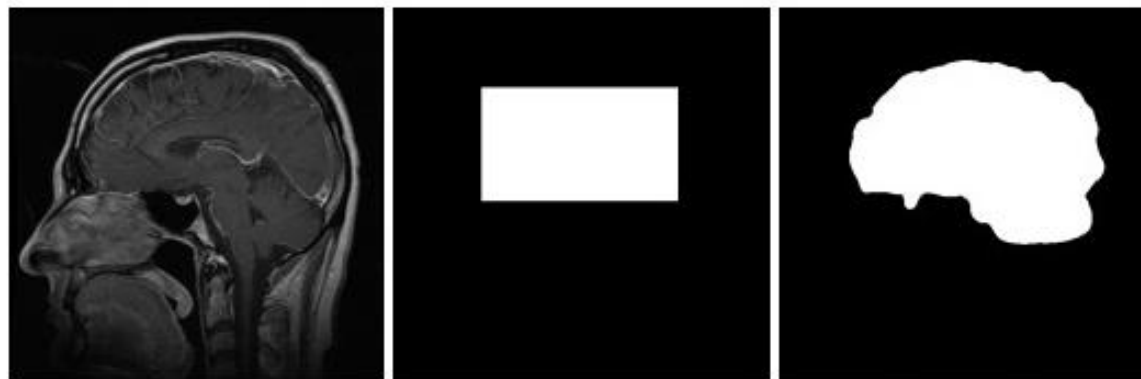


Входное изображение

Маска

Результат

Сегментация печени с параметрами $T = 180, \epsilon = 45, \alpha = 0.003$



Входное изображение

Маска

Результат

Сегментация мозга с параметрами $T = 45, \epsilon = 30, \alpha = 0.003$

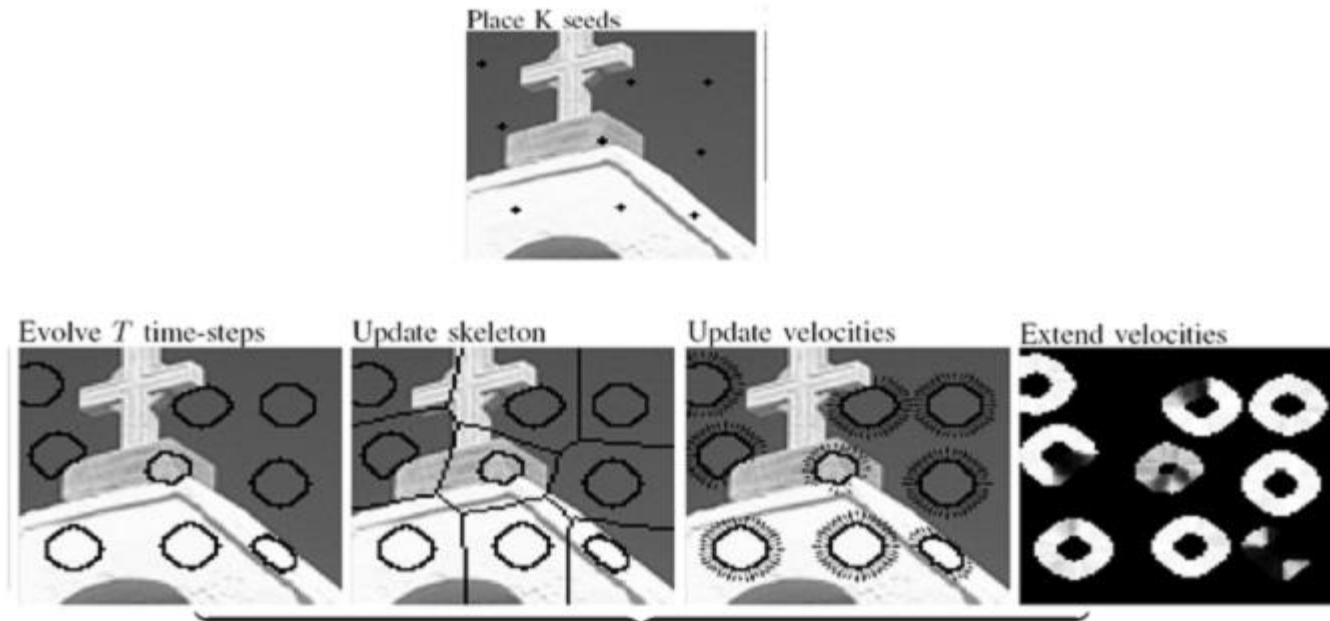
TurboPixel

- Алгоритм, специально нацеленный на сегментацию изображения на суперпиксели
- Даёт суперпиксели примерно одного
- размера, равномерно распределенные по изображению
- Использует подход линий уровня для сегментации
- Сложность $O(N)$, где N – пиксели
- При увеличении количества суперпикселей даже ускоряется



Alex Levinshtein, et. al, TurboPixels: Fast Superpixels Using Geometric Flows, PAMI 2009

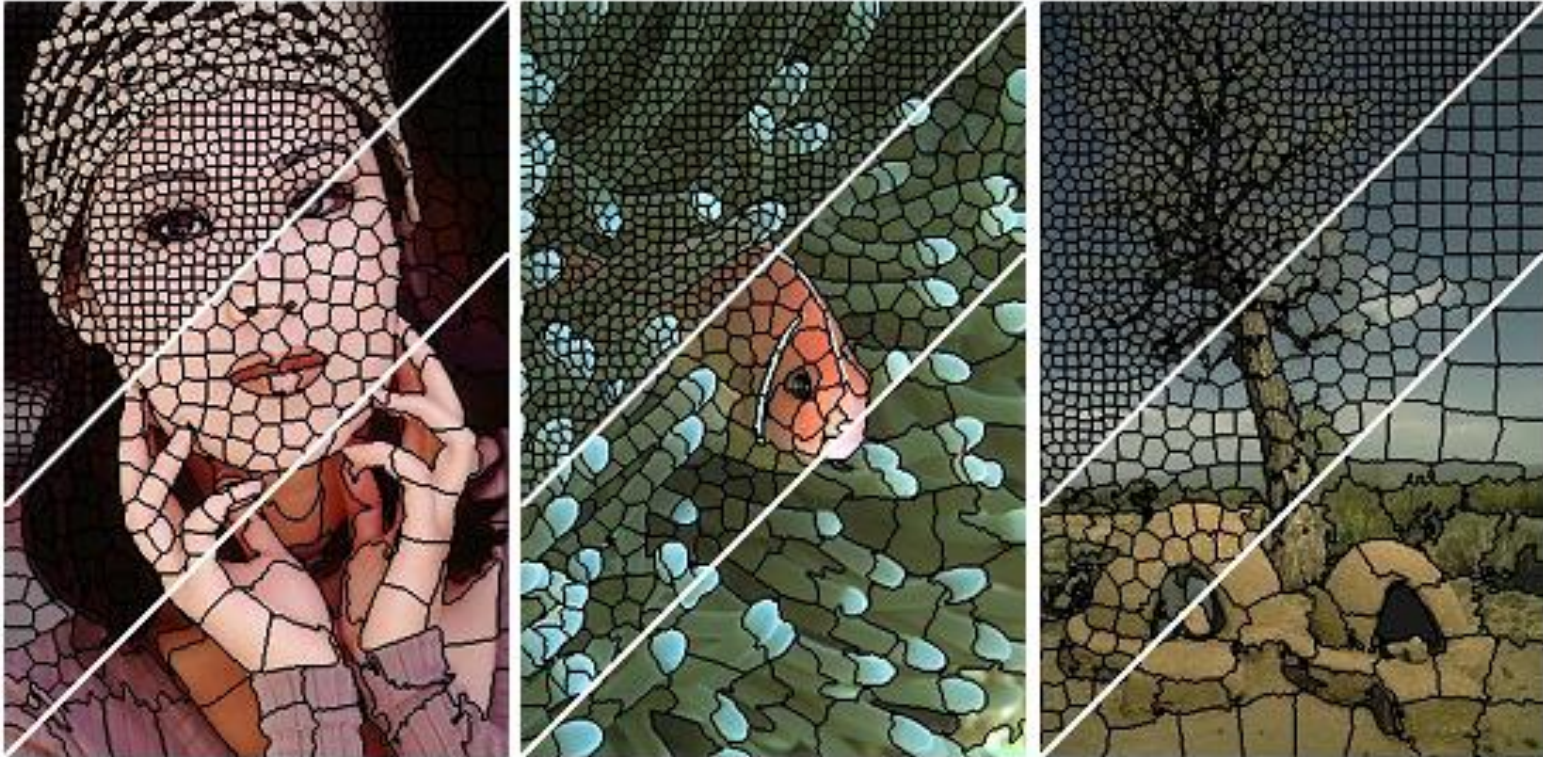
Схема алгоритма



- Идея: скорость движения «контура» зависит от градиента, близости к предполагаемой границе региона и т.д.
- Благодаря этому суперпиксели «тормозятся» у краёв изображения и делят его на фрагменты похожего размера



SLIC Superpixel



Images segmented using SLIC into superpixels of size 64, 256, 1024 pixels (approximately).

Algorithm 1 SLIC superpixel segmentation

/ Initialization */*

Initialize cluster centers $C_k = [l_k, a_k, b_k, x_k, y_k]^T$ by sampling pixels at regular grid steps S .

Move cluster centers to the lowest gradient position in a 3×3 neighborhood.

Set label $l(i) = -1$ for each pixel i .

Set distance $d(i) = \infty$ for each pixel i .

repeat

/ Assignment */*

for each cluster center C_k **do**

for each pixel i in a $2S \times 2S$ region around C_k **do**

 Compute the distance D between C_k and i .

if $D < d(i)$ **then**

 set $d(i) = D$

 set $l(i) = k$

end if

end for

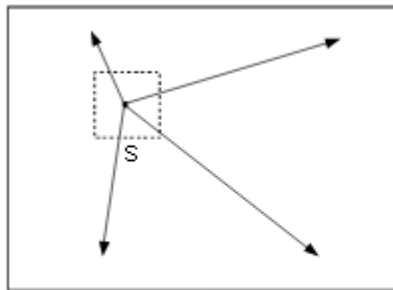
end for

/ Update */*

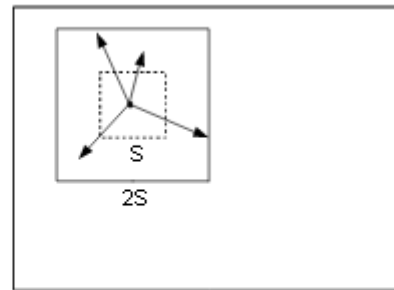
 Compute new cluster centers.

 Compute residual error E .

until $E \leq \text{threshold}$



(a) standard k -means searches the entire image

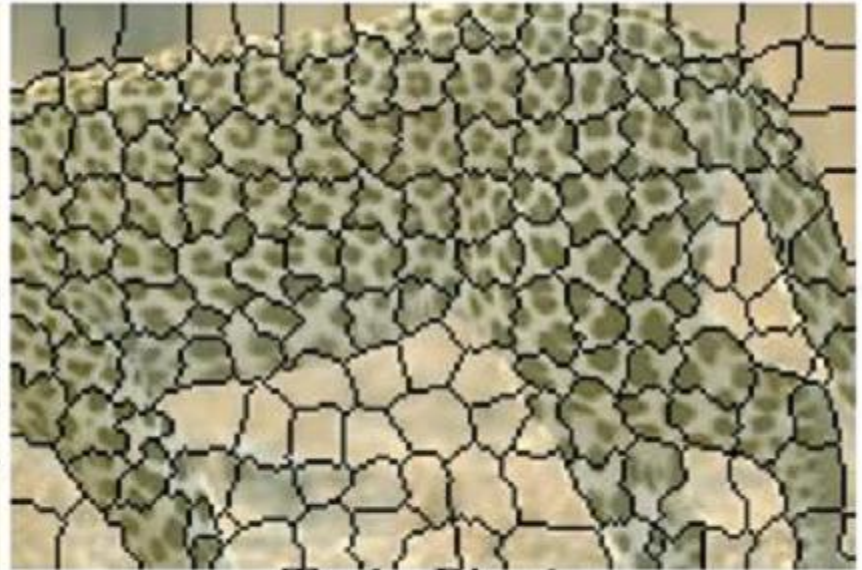


(b) SLIC searches a limited region







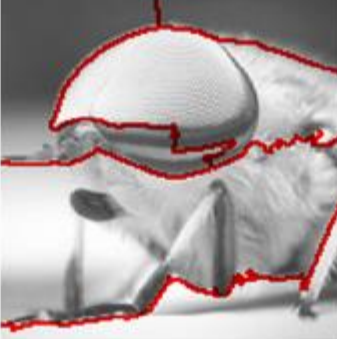



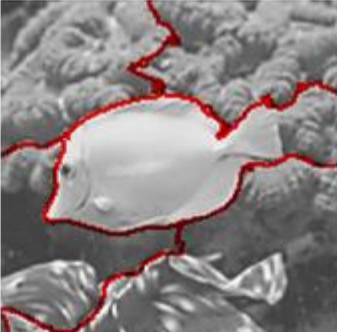
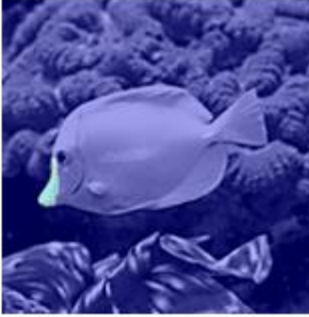
Сравнение







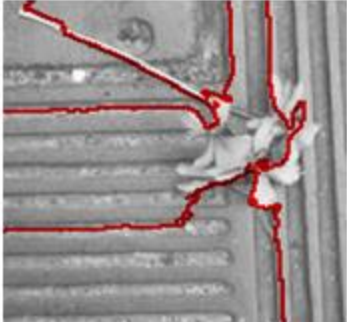
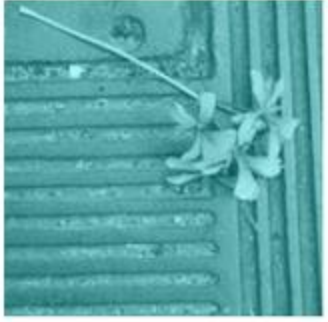


Ncuts

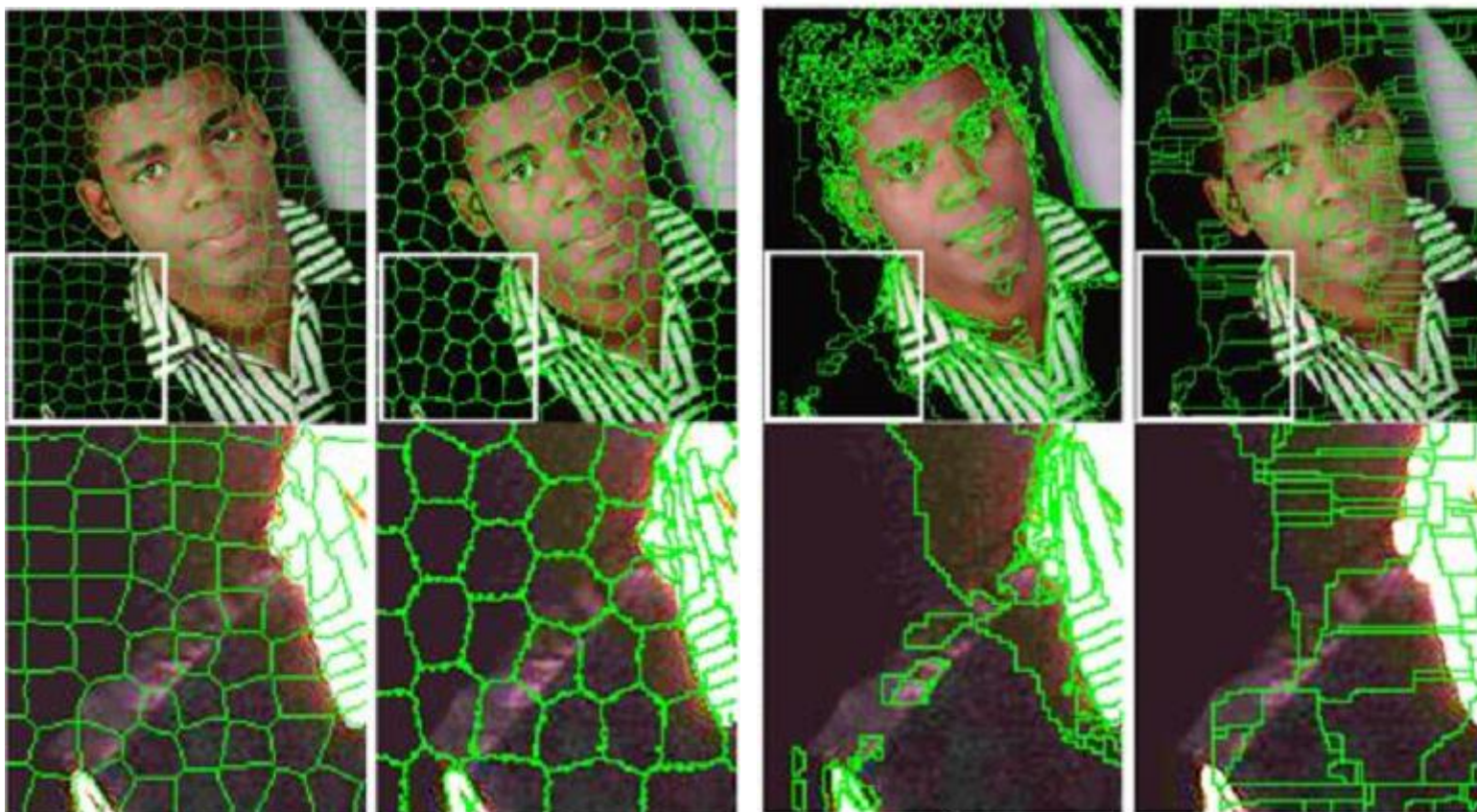


TurboPixels

Исходное изображение	Метод выращивания регионов	Метод нормальных разрезов	Метод водораздела
			
			
			

Исходное изображение	Метод выращивания регионов	Метод нормальных разрезов	Метод водораздела
			
			

Сравнение



TurboPixel

NCuts

Mean-shift

Watershed

Резюме лекции

- Задача сегментации – «разбора изображения» в общем случае является целью распознавания изображений
- Сегментация изображения на области по набору признаков – эффективная предобработка для решения других основных задач
- Понижение размерности задач (работа с областями, а не отдельными пикселями)
- Хорошая сегментация должна учитывать несколько признаков в совокупности
- Чаще всего используются в зависимости от задачи:
 - Методы на графах
 - Mean shift
 - Turbopixels

Литература

1. R. M. Haralick, L. G. Shapiro, "Image Segmentation Techniques," Computer Vision, Graphics, and Image Processing, Vol 29, No 1, 1985
2. K. S. Fu and J. k. Mui, "A Survey on Image Segmentation," Pattern Recognition, Vol. 13, 1981
3. N. R. Pal and S. K. Pal, "A Review on Image Segmentation Techniques," Pattern Recognition, Vol. 26, No 9, 1993
4. R. Jain, R. Kasturi, and B. G. Schunck, Machine Vision, 1995
5. L. Lucchese and S.K. Mitra "Color Image Segmentation: A State-of-the-Art Survey", 2001
6. S.-T. Bow, Pattern Recognition and Image Preprocessing, Marcel Dekker, Inc., New York, NY, 1992.
7. G. R. Cross and A. K. Jain, "Markov Random Field Texture Models," IEEE Trans. on Pattern Analysis and Machine Intelligence, 1983
8. S. German and D. German, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," IEEE Trans. on Pattern Analysis and Machine Intelligence, 1984
9. H. Digabel and C. Lantujoul, "Iterative Algorithms," Proc. of the 2nd European Symp. on Quantitative Analysis of Microstructures in Material Science, Biology and Medicine, 1977

Литература

10. M. Celenk, "Hierarchical Color Clustering for Segmentation of Textured Images," Proc. of the 29th Southeastern Symposium on System Theory, 1997
11. S. Ji and H. W. Park, "Image Segmentation of Color Image Based on Region Coherency," Proc. of ICIP'98
12. L. Shafarenko, M. Petrov, and J. Kittler, "Automatic Watershed Segmentation of Randomly Textured Color Images," IEEE Trans. on Image Processing, 1997
13. M. Barni, S. Rossi, and A. Mecocci, "A Fuzzy Expert System for Low Level Image Segmentation," EUSIPCO-96
14. A. Tremeau and N. Borel, "A Region Growing and Merging Algorithm to Color Segmentation," Pattern Recognition, 1997
15. Y. Kanai, "Image Segmentation Using Intensity and Color Information," SPIE – Visual Communications and Image Processing'98
16. B. Cramariuc, M. Gabbouj, and J. Astola, "Clustering Based Region Growing Algorithm for Color Image Segmentation," Int. Conf. on Digital Signal Processing, 1997
17. Y. Deng, B. S. Manjunath, and H. Shin, "Color Image Segmentation", CVPR 1999

Литература

18. A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics David Martin Charless Fowlkes Doron Tal Jitendra Malik Department of Electrical Engineering and Computer Sciences
19. M. Jacob, M. Unser, "Design of Steerable Filters for Feature Detection Using Canny-Like Criteria," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 8, pp. 1007-1019
20. Image Segmentation By Learning Approach Horacio Andrés Legal-Ayala, Jacques Facon, 2003
21. Bayesian learning, global competition and unsupervised image Segmentation Guodong Guo and Songde Ma
22. Edge Detection Using steerable Filters and CNN, Atilla Ozmen and Emir Tufan Akman, 2002
23. Normalized Cuts and Image Segmentation - J. Shi, J. Malik (1997) University of California at Berkeley
24. Efficient Spatiotemporal Grouping Using the Nystrom Method (CVPR 2001) - Charless Fowlkes , Serge Belongie, Jitendra Malik
25. Efficient Graph Cuts for Unsupervised Image Segmentation using Probabilistic Sampling and SVD-based Approximation (ICCV 2003) - J. Keuchel, C. Schnorr, University of Mannheim, Germany

Литература

26. Image Segmentation by Nested Cuts (2000) - Olga Veksler, NEC Research Institute
27. A New Graph-Theoretic Approach to Clustering and Segmentation (CVPR03)
28. Fast Multiscale Image Segmentation (CVPR 2000) - Eitan Sharon, Achi Brandt Ronen Basriy, Dept. of Applied Math The Weizmann Inst. of Science, Israel
29. Segmentation and Boundary Detection Using Multiscale Intensity Measurements (CVPR 2001) - Eitan Sharon, Achi Brandt_, Ronen Basri
30. Texture Segmentation by Multiscale Aggregation of Filter Responses and Shape Elements (ICCV 2003) - Meirav Galun, Eitan Sharon, Ronen Basri, Achi Brandt
31. Learning Affinity Functions for Image Segmentation: Combining Patch-based and Gradient-based Approaches, Charless Fowlkes, David Martin, Jitendra Malik, 2003
32. Boosting Margin Based Distance Functions for Clustering, Tomer Hertz, Aharon Bar-Hillel , 2004
33. Learning to Detect Natural Image Boundaries Using Brightness and Texture, Charless Fowlkes, David Martin, Jitendra Malik, 2003

Литература

34. How Much Does Globalization Help Segmentation, Charles Fowlkes and Jitendra Malik, 2003
35. Bishop, C. M. Neural Networks for Pattern Recognition. Oxford, England: Oxford University Press, 1995