

Лекция 3. Минимизация энергии с помощью разрезов графов.

Вадим Конушин

Спецкурс «Структурные методы анализа изображений и
сигналов»

План лекции

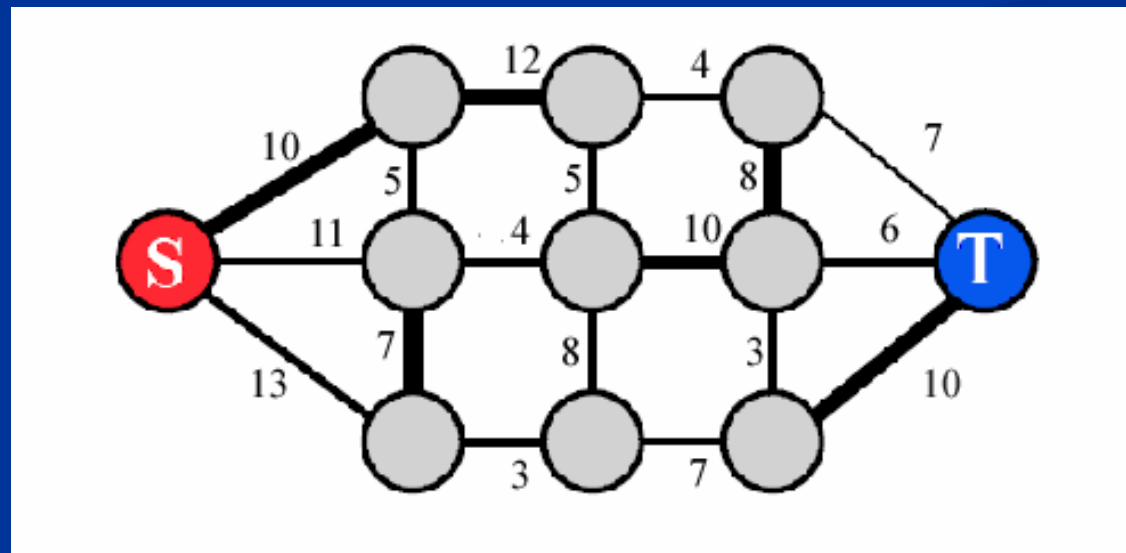


- Элементы теории графов
- Алгоритмы нахождения максимального потока в графе
- Интерактивная сегментация изображений
- Какую энергию можно минимизировать с помощью разрезов графов?
- Ускорения
 - Динамические марковские случайные поля
 - Многоуровневые разрезы графов
 - Алгоритм touch-expand
- Итерационные алгоритмы
- Приложения



Элементы теории графов (1)

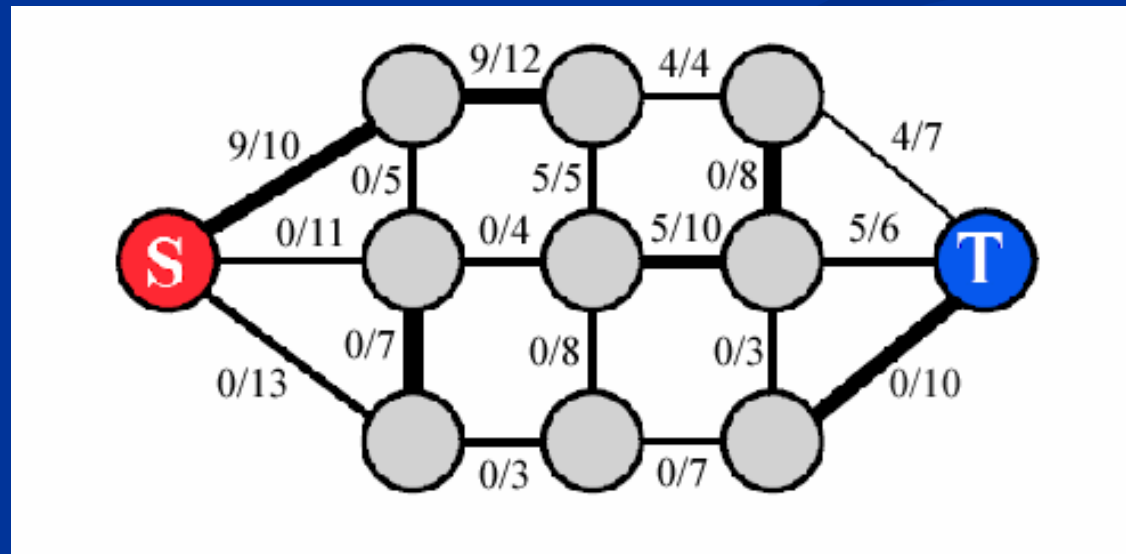
- Пусть $G = (V, E)$ - направленный граф. Каждому ребру (u, v) присваивается некоторый неотрицательный вес – пропускная способность ребра $c(u, v)$. В графе выделены 2 особые (терминальные) вершины – исход s и сток t . Такой граф называют сетью.





Элементы теории графов (2)

- **Опр.** Поток в графе G называется функция $f(u, v)$, определенная на ребрах графа, и удовлетворяющая следующим условиям:
 - $f(u, v) \geq 0 \quad \forall u, v$
 - $f(u, v) \leq c(u, v) \quad \forall u, v$
 - $\sum_{v:(v, u_0) \in E} f(v, u_0) - \sum_{v:(u_0, v) \in E} f(u_0, v) = 0 \quad \forall u_0 \in V, u_0 \neq s, u_0 \neq t$



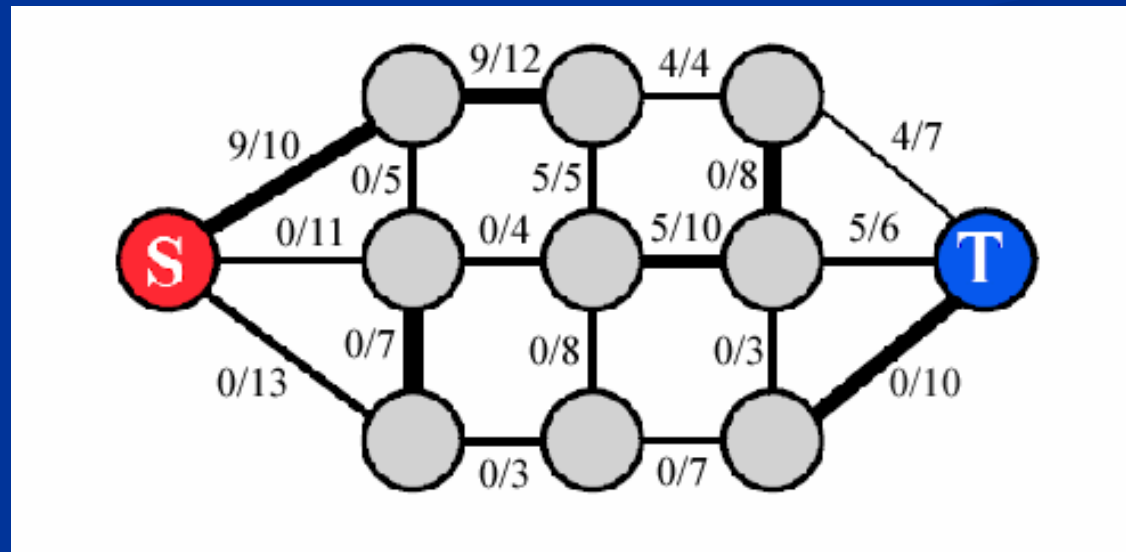


Элементы теории графов (3)

- **Опр.** Величиной потока называется величина

$$M(f) = \sum_{v:(p,v) \in E} f(p,v) - \sum_{v:(v,p) \in E} f(v,p) \quad p \in \{s,t\}$$

- **Задача о максимальном потоке**

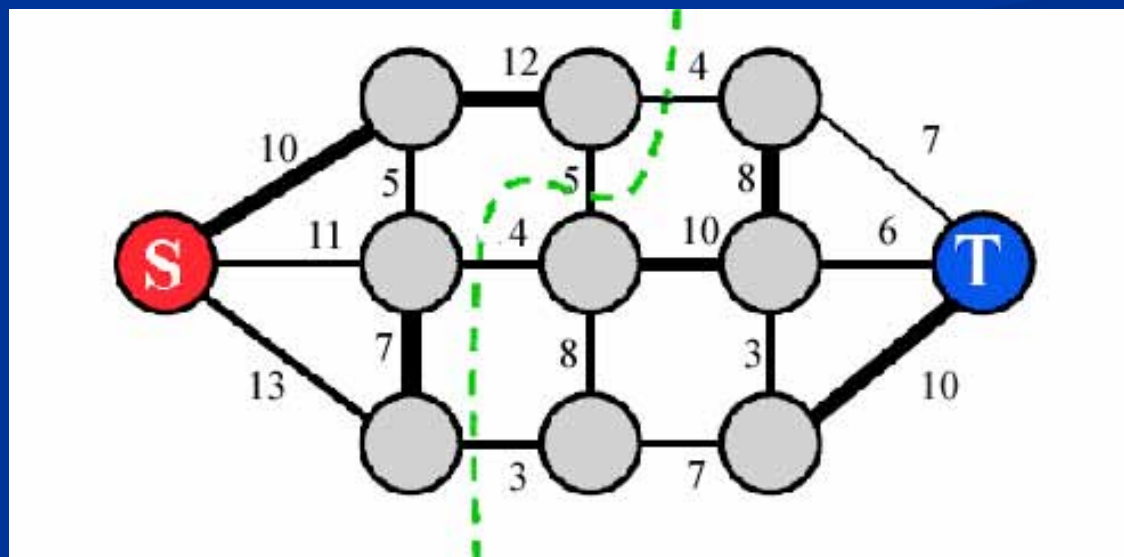




Элементы теории графов (4)

- **Опр.** $s - t$ -разрез $C = S, T$ - это разбиение вершин из V на 2 непересекающихся множества S и T , такие, что $s \in S, t \in T$
- **Опр.** Величиной разреза называется сумма весов всех ребер с началом в S и концом в T :

$$c(S, T) = \sum_{u \in S, v \in T, (u, v) \in E} c(u, v)$$

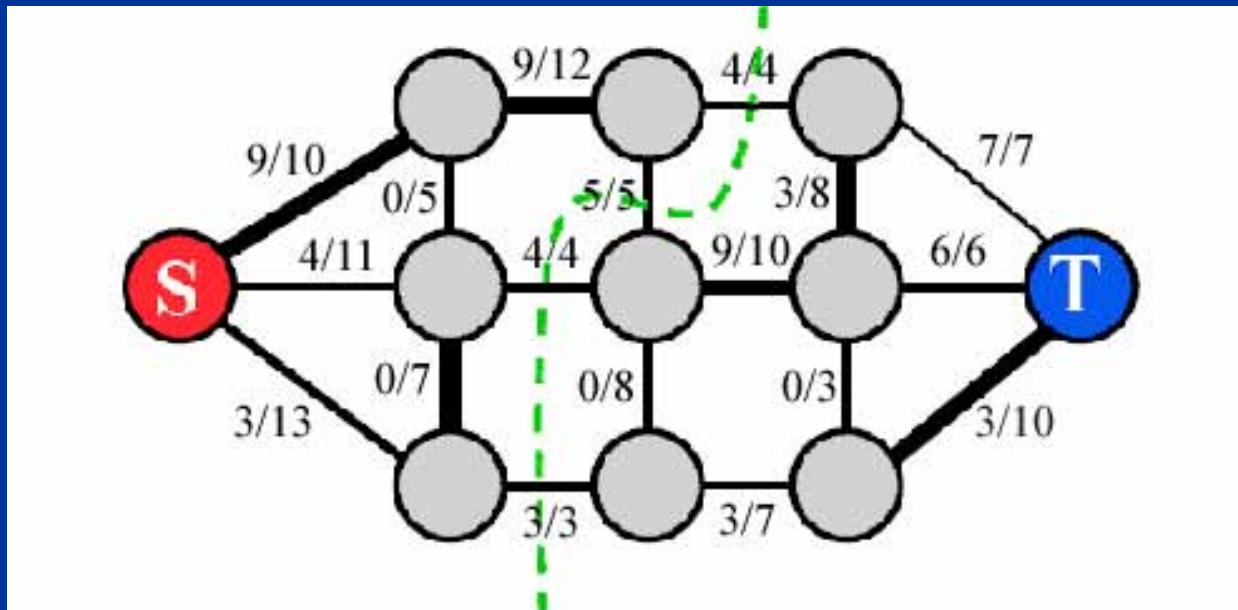


Элементы теории графов (5)



- Теорема Форда – Фалкерсона .

Величина максимального потока равна величине минимального разреза.



План лекции



- Элементы теории графов
- Алгоритмы нахождения максимального потока в графе
- Интерактивная сегментация изображений
- Какую энергию можно минимизировать с помощью разрезов графов?
- Ускорения
 - Динамические марковские случайные поля
 - Многоуровневые разрезы графов
 - Алгоритм touch-expand
- Итерационные алгоритмы
- Приложения

Алгоритмы нахождения максимального потока в графе (1)



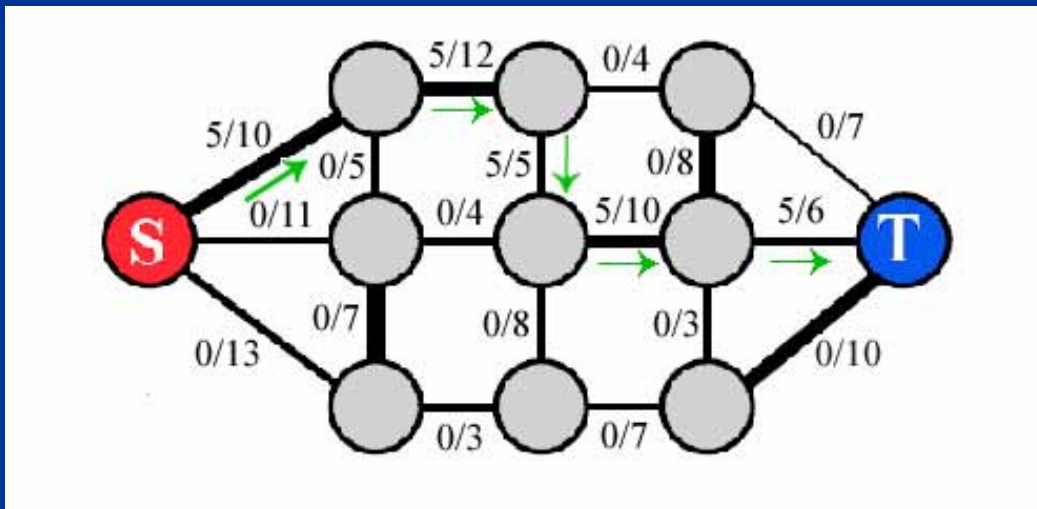
- Алгоритмы, основанные на дополняющих путях [Ford & Fulkerson, 1962]
- Алгоритмы, основанные на проталкивании предпотока (push-relabel) [Goldberg-Tarjan, 1986]

Алгоритмы нахождения максимального потока в графе (2)



■ Дополняющие пути

- Найти путь от S к T по ненасыщенным ребрам
- Увеличить поток вдоль данного пути до тех пор, пока какое-то ребро не станет насыщенным.

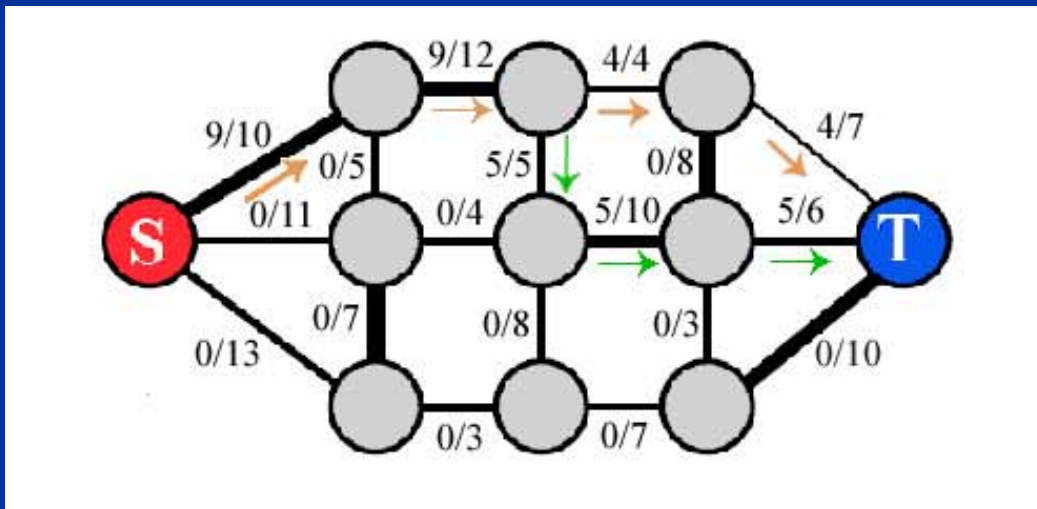


Алгоритмы нахождения максимального потока в графе (2)



■ Дополняющие пути

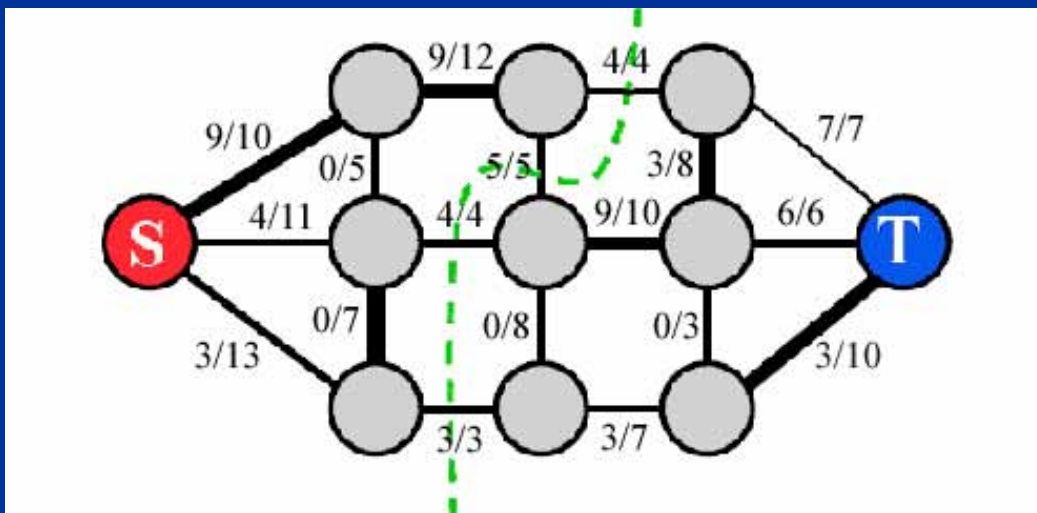
- Найти путь от S к T по ненасыщенным ребрам
- Увеличить поток вдоль данного пути до тех пор, пока какое-то ребро не станет насыщенным.



Алгоритмы нахождения максимального потока в графе (2)



■ Дополняющие пути



- Сложность: $O(E * \text{maxflow})$ (веса – только целые)

- Найти путь от S к T по ненасыщенным ребрам
- Увеличить поток вдоль данного пути до тех пор, пока какое-то ребро не станет насыщенным.
- Делать новые итерации до тех пор, пока каждый путь из S в T не будет содержать как минимум 1 насыщенное ребро.

Алгоритмы нахождения максимального потока в графе (3)



- Бойков и Колмогоров, EMMCVPR 2001
 - Эмпирическое сравнение различных версий алгоритмов, основанных на дополняющих путях и проталкивании предпотока на графах-решетках, типичных для задач компьютерного зрения
 - Сделали собственную версию алгоритма, основанного на дополняющих путях
 - Эмпирическая сложность близка к линейной относительно размера изображения.

Алгоритмы нахождения максимального потока в графе (4)



- Проталкивание предпотока
 - Может быть распараллелен на все вершины графа [Goldberg 86]
 - Распараллеленный алгоритм вместе с типичной для задач зрения структурой графа (решетка) отлично ложатся на GPU

План лекции



- Элементы теории графов
- Алгоритмы нахождения максимального потока в графе
- Интерактивная сегментация изображений
- Какую энергию можно минимизировать с помощью разрезов графов?
- Ускорения
 - Динамические марковские случайные поля
 - Многоуровневые разрезы графов
 - Алгоритм touch-expand
- Итерационные алгоритмы
- Приложения

Интерактивная сегментация изображений (1)



Интерактивная сегментация изображений (2)



- Входные данные:
 - Пользователь указывает т.н. семена объекта и фона – пиксели, которые должны принадлежать объекту (фону) в итоговой сегментации.
 - О – семена объекта
 - В – семена фон

Интерактивная сегментация изображений (3)



- $P = \{1, \dots, m\}$ – пиксели изображения
- Система соседства N - обычная 4 или 8-связность
- F – разметка изображения (“obj”, “bkg”)
- I – изображение

Интерактивная сегментация изображений (4)



$$E(F) = \lambda \cdot R(F) + B(F)$$

$R(F)$ (regional term) – отвечает за свойства областей сегментов (например, на сколько цвета пикселей объекта похожи на ожидаемый цвет объекта)

$B(F)$ (boundary term) – отвечает за свойства границы сегментов (например, на сколько граница сегментов контрастна)

Интерактивная сегментация изображений (5)



$$E(F) = \lambda \cdot R(F) + B(F)$$

$$R(F) = \sum_{p \in P} R_p(F)$$

$R_p(F)$ - индивидуальные штрафы за отнесение пикселя p к объекту или фону

Интерактивная сегментация изображений (6)



$$E(F) = \lambda \cdot R(F) + B(F)$$

$$R(F) = \sum_{p \in P} R_p(F)$$

$R_p(F)$ - индивидуальные штрафы за отнесение пикселя p к объекту или фону

Пример:

$R_p("obj") = -\ln \Pr(I_p | O)$ $\Pr(I_p | O)$ - вероятность того, что пиксель объекта примет цвет I_p

$R_p("bkg") = -\ln \Pr(I_p | B)$

Интерактивная сегментация изображений (7)



$$E(F) = \lambda \cdot R(F) + B(F)$$

$$B(F) = \sum_{\{p,q\} \in N} B_{\{p,q\}} \cdot \delta(F_p, F_q) \quad \delta(F_p, F_q) = \begin{cases} 1 & \text{if } F_p \neq F_q \\ 0 & \text{otherwise} \end{cases}$$

$B_{\{p,q\}}$ - штрафы за границу между пикселями p и q

Интерактивная сегментация изображений (8)



$$E(F) = \lambda \cdot R(F) + B(F)$$

$$B(F) = \sum_{\{p,q\} \in N} B_{\{p,q\}} \cdot \delta(F_p, F_q) \quad \delta(F_p, F_q) = \begin{cases} 1 & \text{if } F_p \neq F_q \\ 0 & \text{otherwise} \end{cases}$$

$B_{\{p,q\}}$ - штрафы за границу между пикселями p и q

Пример:

$$B_{\{p,q\}} \propto \exp\left(-\frac{(I_p - I_q)^2}{2\sigma^2}\right) \cdot \frac{1}{dist(p,q)} \quad dist(p,q) \text{ - расстояние между пикселями}$$

Интерактивная сегментация изображений (9)



$$F = \arg \min_F (E(F))$$

при условии:

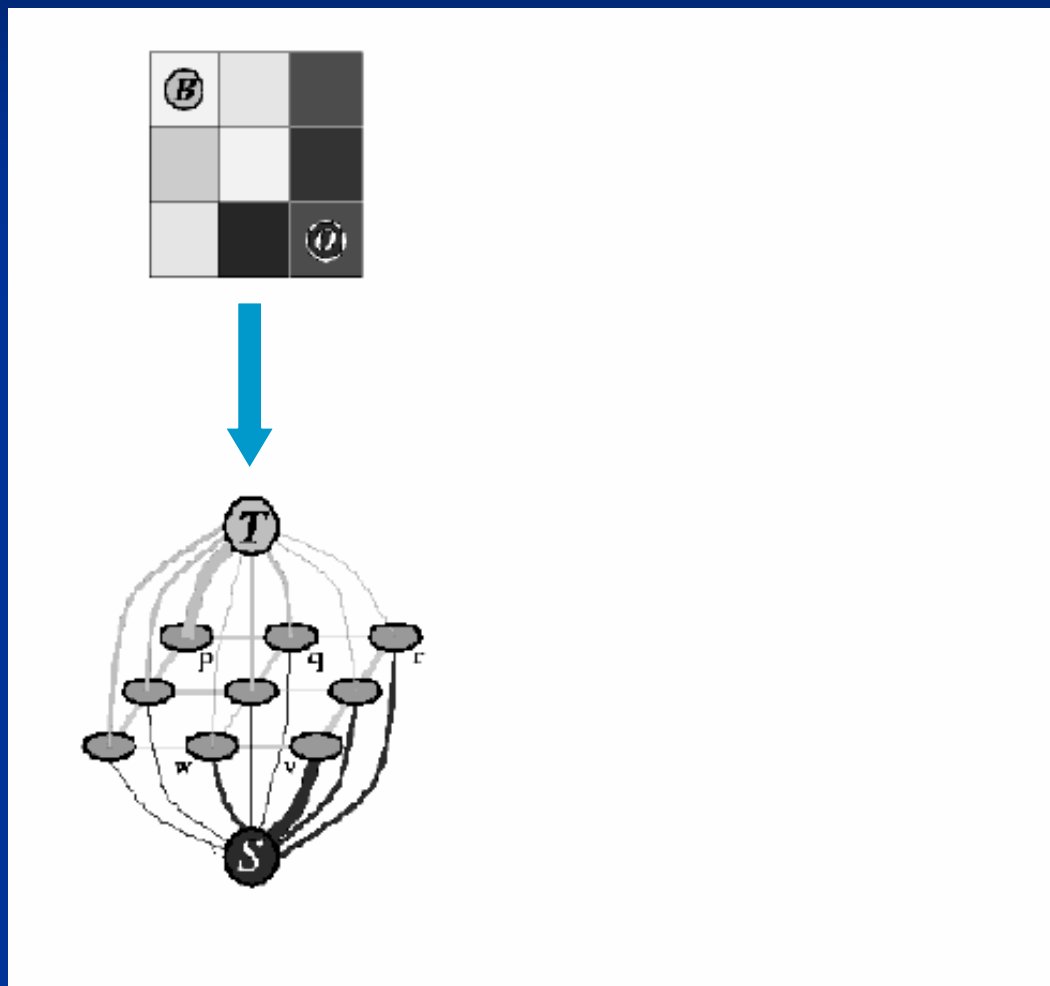
$$\forall p \in O, F_p = "obj"$$

$$\forall p \in B, F_p = "bkg"$$

Интерактивная сегментация изображений (10)



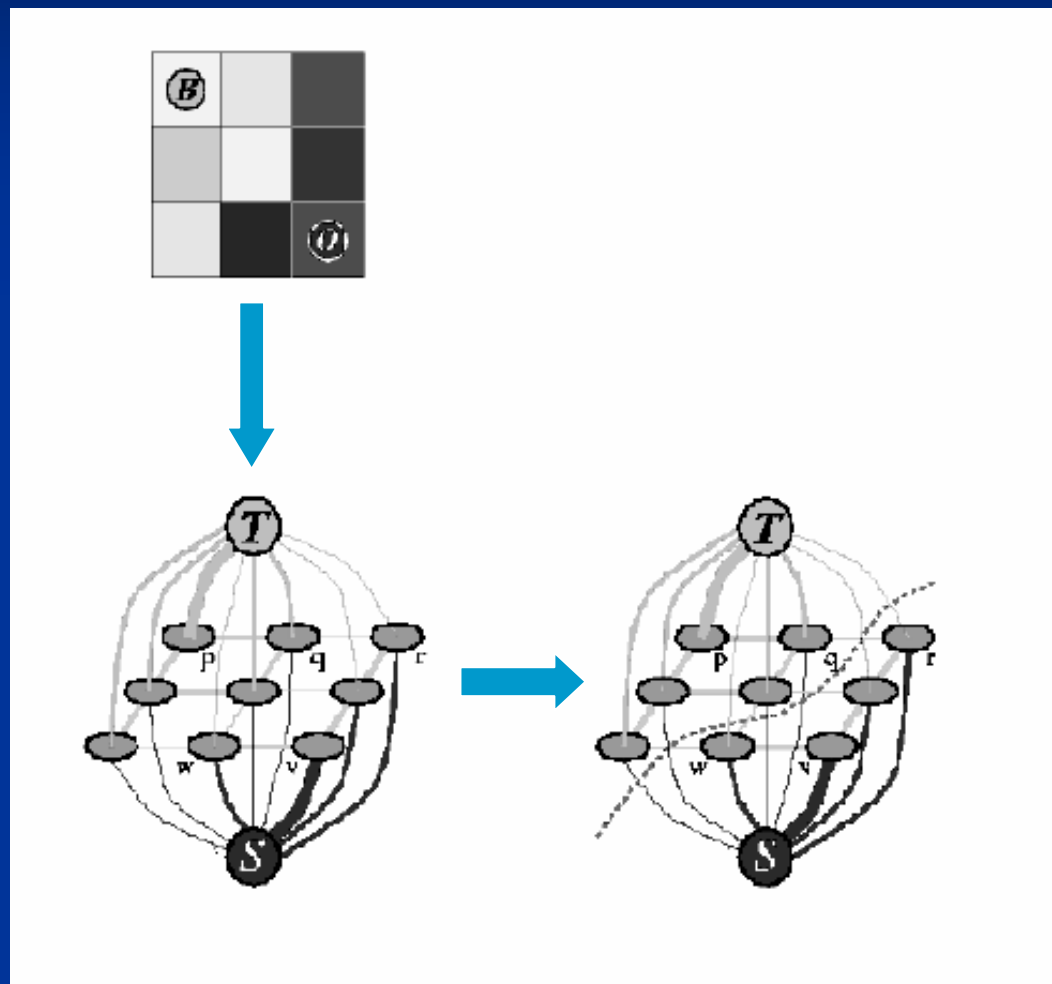
Интерактивная сегментация изображений (11)



■ Построение графа

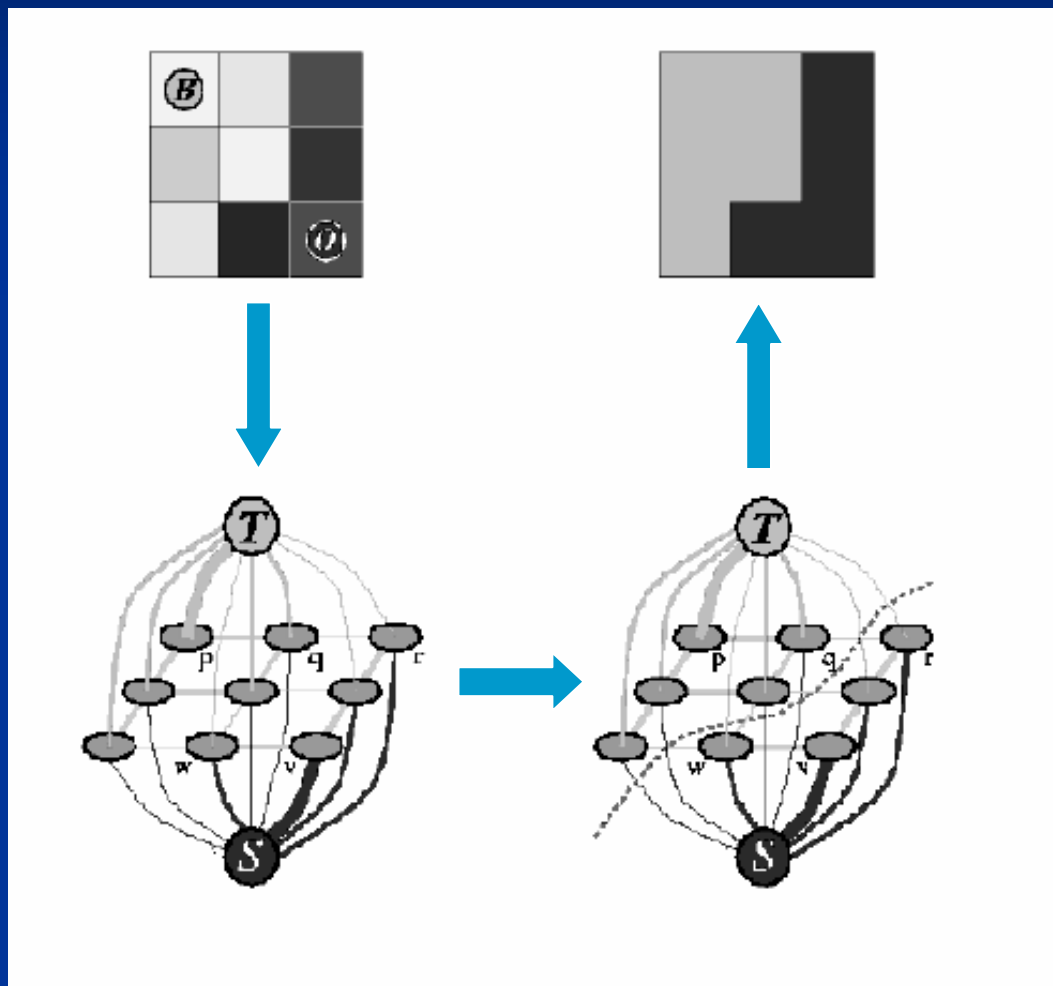
- t-связь – ребро между обычной вершиной и терминальной
- n-связь – ребро между двумя обычными вершинами

Интерактивная сегментация изображений (12)



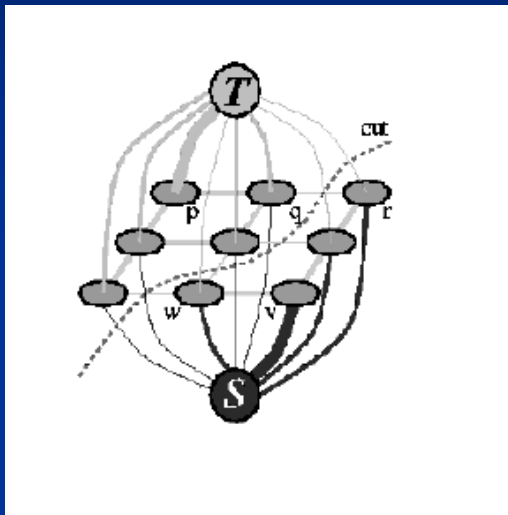
- Разрез C разрезает ровно 1 t -связь в каждом пикселе p
- $\{p, q\} \in C$ тогда и только тогда, когда p и q соединены t -связями с двумя разными терминальными вершинами

Интерактивная сегментация изображений (13)



- Существует взаимнооднозначное соответствие между множеством разрезов C и множеством возможных разметок F
 - $F(C)$

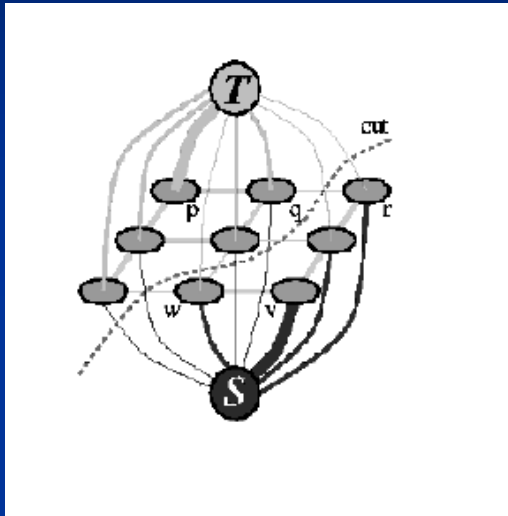
Интерактивная сегментация изображений (14)



$$K = 1 + \max_{p \in P} \sum_{q: \{p,q\} \in N} u_{\{p,q\}}$$

Ребро	Вес	В случае
$\{p, q\}$	$B_{\{p,q\}}$	$\{p, q\} \in N$
$\{p, S\}$	$\lambda \cdot R_p(\text{"bkg"})$	$p \in P, p \notin O \cup B$
	K	$p \in O$
	0	$p \in B$
$\{p, T\}$	$\lambda \cdot R_p(\text{"obj"})$	$p \in P, p \notin O \cup B$
	0	$p \in O$
	K	$p \in B$

Интерактивная сегментация изображений (15)

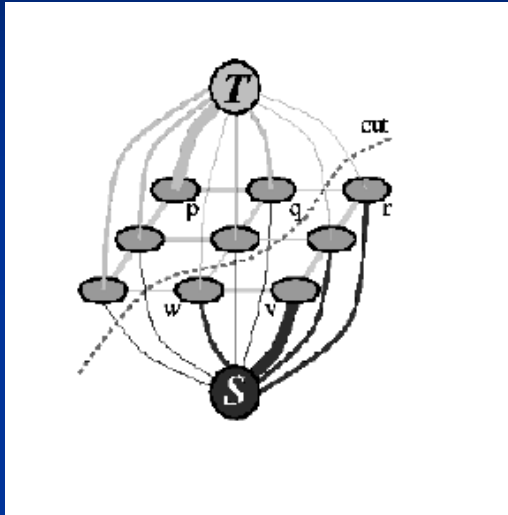


$$K = 1 + \max_{p \in P} \sum_{q: \{p,q\} \in N} u_{\{p,q\}}$$

Ребро	Вес	В случае
$\{p, q\}$	$B_{\{p,q\}}$	$\{p, q\} \in N$
$\{p, S\}$	$\lambda \cdot R_p(\text{"bkg"})$	$p \in P, p \notin O \cup B$
	K	$p \in O$
	0	$p \in B$
$\{p, T\}$	$\lambda \cdot R_p(\text{"obj"})$	$p \in P, p \notin O \cup B$
	0	$p \in O$
	K	$p \in B$

$$|C| = \sum_{p \notin O \cup B} \lambda \cdot R_p(F_p(C)) + \sum_{\{p,q\} \in N} B_{\{p,q\}} \cdot \delta(F_p(C), F_q(C)) = E(F(C)) - \sum_{p \in O} \lambda R_p(\text{"obj"}) - \sum_{p \in B} \lambda R_p(\text{"bkg"})$$

Интерактивная сегментация изображений (16)



$$K = 1 + \max_{p \in P} \sum_{q: \{p,q\} \in N} u_{\{p,q\}}$$

Ребро	Вес	В случае
$\{p, q\}$	$B_{\{p,q\}}$	$\{p, q\} \in N$
$\{p, S\}$	$\lambda \cdot R_p(\text{"bkg"})$	$p \in P, p \notin O \cup B$
	K	$p \in O$
	0	$p \in B$
$\{p, T\}$	$\lambda \cdot R_p(\text{"obj"})$	$p \in P, p \notin O \cup B$
	0	$p \in O$
	K	$p \in B$

$$|C| = E(F(C)) - const$$

Интерактивная сегментация изображений (17)



- Теорема. Сегментация $\hat{F} = F(\hat{C})$, заданная минимальным разрезом графа \hat{C} минимизирует энергию

$$E(F) = \lambda \cdot R(F) + B(F)$$

при условии жестких ограничений:

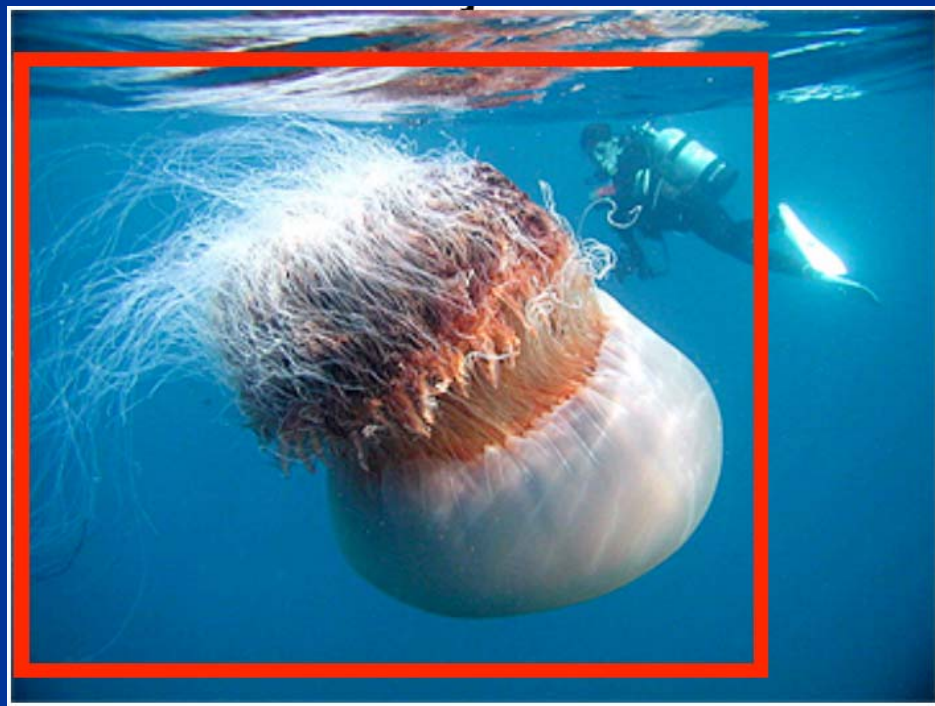
$$\forall p \in O, F_p = \text{"obj"}$$

$$\forall p \in B, F_p = \text{"bkg"}$$

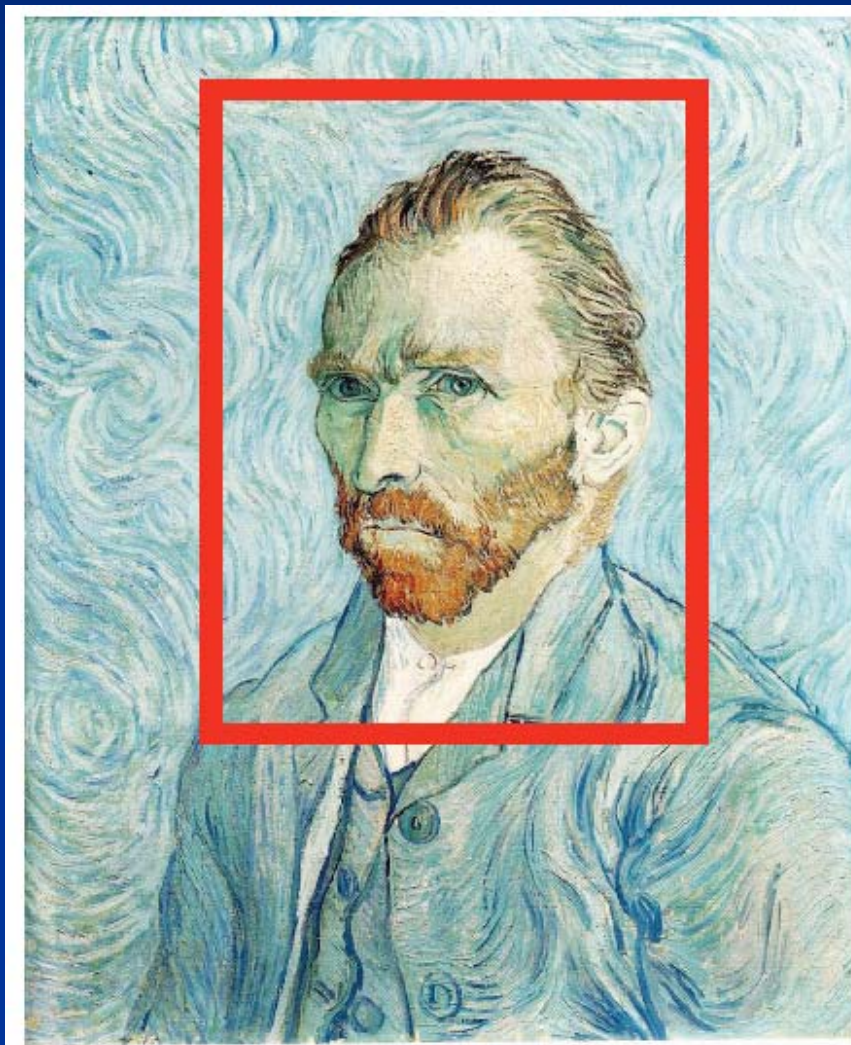
Интерактивная сегментация изображений (18)



Интерактивная сегментация изображений (19)



Интерактивная сегментация изображений (20)



Марковская сеть (1)



- I – наблюдаемые переменные
- F – скрытые переменные
- Задача нахождения наиболее вероятной конфигурации скрытых переменных $\Pr(F | I) \rightarrow \max$

Марковская сеть (2)



- Формула Байеса: $\Pr(F | I) \propto \Pr(I | F) \Pr(F)$

Марковская сеть (3)



- Формула Байеса: $\Pr(F | I) \propto \Pr(I | F) \Pr(F)$

←
правдоподобие

←
априорная вероятность

Марковская сеть (4)



- Формула Байеса: $\Pr(F | I) \propto \Pr(I | F) \Pr(F)$

$$\arg \max_F \Pr(F | I) = \arg \max_F \Pr(I | F) \Pr(F) =$$

$$\arg \max_F (\ln(\Pr(I | F)) + \ln(\Pr(F)))$$

Марковская сеть (5)



- Правдоподобие:

$$\Pr(I | F) = \prod_p \Pr(I_p | F_p)$$

$$\Rightarrow \ln(\Pr(I | F)) = \sum_p \ln(\Pr(I_p | F_p))$$

Марковская сеть (6)



- Правдоподобие:

$$\Pr(I | F) = \prod_p \Pr(I_p | F_p)$$

$$\Rightarrow \ln(\Pr(I | F)) = \sum_p \ln(\Pr(I_p | F_p))$$

$$R(F) = \sum_{p \in P} R_p(F)$$

$$R_p(\text{"obj"}) = -\ln \Pr(I_p | O)$$

$$R_p(\text{"bkg"}) = -\ln \Pr(I_p | B)$$



Марковская сеть (7)

- Теорема Хаммерсли-Клиффорда:
вероятность конфигурации F

$$\Pr(F) \propto \exp\left(-\sum_C V_C(F)\right)$$

V_C - потенциал клики, который описывает априорную вероятностью конкретной конфигурации клики C

$$B(F) = \sum_{\{p,q\} \in N} B_{\{p,q\}} \cdot \delta(F_p, F_q)$$

Марковская сеть (8)



Boykov et al. [ICCV 2001], Blake et al. [ECCV 2004]

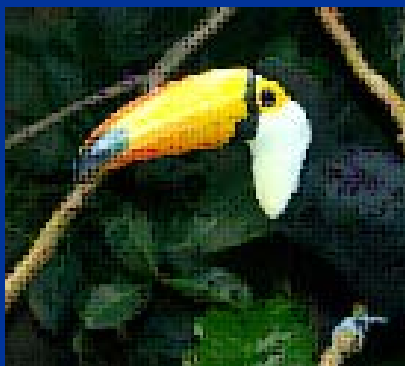
$$Energy_{MRF} = E(F) = \sum_p (\phi(I | F_p)) + \sum_{\{p,q\} \in N} (\phi(I | F_p, F_q) + \psi(F_p, F_q)) + const$$

Унарное правдоподобие
(Unary likelihood)

Член контраста
(Contrast Term)

Модель априорной
вероятности Поттса
(Potts Model Prior)

$$MAP \text{ solution } F^* = \arg \min_F E(F)$$



Данные (I)



Унарное правдоподобие
(Unary likelihood)



Бинарные члены
(Pair-wise Terms)



Решение максимальной
апостериорной
вероятности

(MAP Solution) Slide 43

План лекции



- Элементы теории графов
- Алгоритмы нахождения максимального потока в графе
- Интерактивная сегментация изображений
- Какую энергию можно минимизировать с помощью разрезов графов?
- Ускорения
 - Динамические марковские случайные поля
 - Многоуровневые разрезы графов
 - Алгоритм touch-expand
- Итерационные алгоритмы
- Приложения

Какую энергию можно минимизировать с помощью разрезов графов (1)



V. Kolmogorov [ECCV 2002], [PAMI 2004]

- В первых алгоритмах, использующих разрезы графов для минимизации энергии, граф строился индивидуально, под задачу.
- Уверенности, что построить нужный граф удастся – не было.

Какую энергию можно минимизировать с помощью разрезов графов (2)



- **Опр.** Пусть $E(x_1, \dots, x_n)$ - функция n двоичных переменных, и пусть I, J - непересекающееся разбиение множества индексов $\{1, \dots, n\}$: $I = \{i(1), \dots, i(m)\}$, $J = \{j(1), \dots, j(n-m)\}$. Пусть $\alpha_{i(1)}, \dots, \alpha_{i(m)}$ - какие-то двоичные константы. Проекцией $E' = E[x_{i(1)} = \alpha_{i(1)}, \dots, x_{i(m)} = \alpha_{i(m)}]$ называется функция $n-m$ переменных, определенная как

$$E'(x_{j(1)}, \dots, x_{j(n-m)}) = E(x_1, \dots, x_n)$$

где $x_i = \alpha_i$ для $i \in I$. Мы будем говорить, что зафиксировали переменные $x_{i(1)}, \dots, x_{i(m)}$.

Какую энергию можно минимизировать с помощью разрезов графов (3)



- **Опр.** (регулярной функции):
 - Все функции одной переменной являются регулярными.
 - Функция E двух переменных называется регулярной, если
$$E(0,0) + E(1,1) \leq E(0,1) + E(1,0)$$
.
 - Функция E более чем двух переменных называется регулярной, если все проекции E двух переменных являются регулярными.
- **Опр.** Назовем функцию энергии E граф-представимой, если для нее можно построить граф, минимальный разрез в котором будет соответствовать минимуму данной энергии.

Какую энергию можно минимизировать с помощью разрезов графов (4)



- **Теорема 1.** Пусть E - функция n двоичных переменных, которая может быть представлена в виде суммы

$$E(x_1, \dots, x_n) = \sum_i E^i(x_i) + \sum_{i < j} E^{i,j}(x_i, x_j) + \sum_{i < j < k} E^{i,j,k}(x_i, x_j, x_k)$$

Тогда E является граф-представимой тогда и только тогда, когда E является регулярной.

- **Теорема 2.** Пусть E функция двоичных переменных. Если E не является регулярной, то E не граф-представима.

План лекции



- Элементы теории графов
- Алгоритмы нахождения максимального потока в графе
- Интерактивная сегментация изображений
- Какую энергию можно минимизировать с помощью разрезов графов?
- Ускорения
 - Динамические марковские случайные поля
 - Многоуровневые разрезы графов
 - Алгоритм touch-expand
- Итерационные алгоритмы
- Приложения

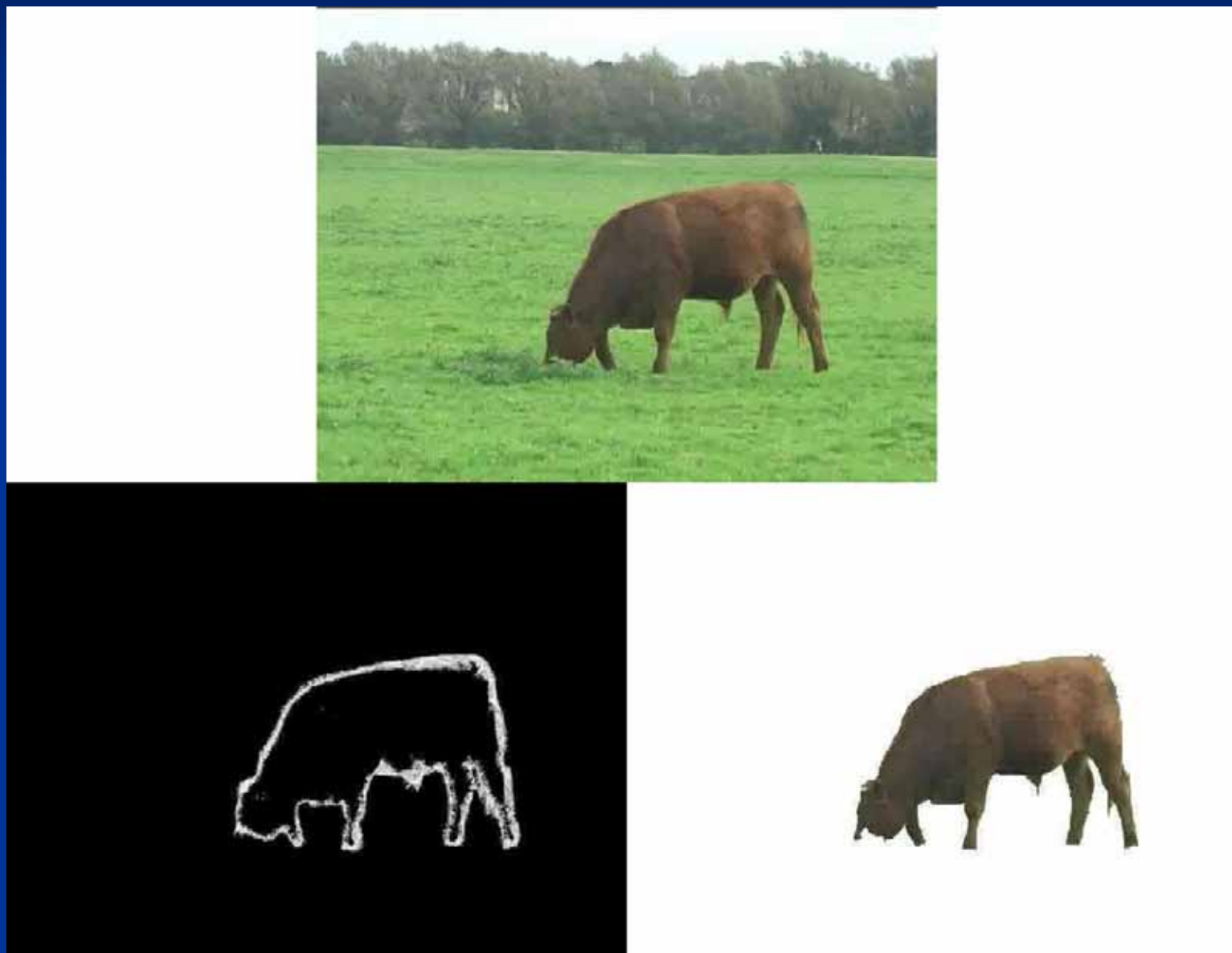
Динамические марковские случайные поля (1)



Р. Kohli, Р. Torr [ICCV 2005]

- Есть 2 похожих марковских случайных поля и решение максимальной апостериорной вероятности для одного из них. Можно ли это использовать для ускорения нахождения такого решения для второго поля?

Динамические марковские случайные поля (2)



Динамические марковские случайные поля (3)



Задача сегментации

Максимальный поток

Решение



Динамические марковские случайные поля (4)



- **Опр.** Поток в графе G называется функция $f(u, v)$, определенная на ребрах графа, и удовлетворяющая следующим условиям:
 - $f(u, v) \geq 0 \quad \forall u, v$
 - $f(u, v) \leq c(u, v) \quad \forall u, v$
 - $\sum_{v:(v, u_0) \in E} f(v, u_0) - \sum_{v:(u_0, v) \in E} f(u_0, v) = 0 \quad \forall u_0 \in V, u_0 \neq s, u_0 \neq t$

Динамические марковские случайные поля (5)



- Опр. Остаточная пропускная способность r_{ij} ребра $(i, j) \in E$ при потоке f - это максимальный дополнительный поток, который можно протолкнуть из вершины i в вершину j , используя ребра (i, j) и (j, i) .
- Остаточная пропускная способность состоит из двух компонент:
 - Неиспользованная пропускная способность ребра (i, j) : $c_{ij} - f_{ij}$
 - Текущий поток f_{ij} из вершины j в вершину i
- Опр. Остаточный граф $G(f)$ графа G состоит из множества вершин V и ребер с положительной остаточной пропускной способностью (относительно потока f)

Динамические марковские случайные поля (6)



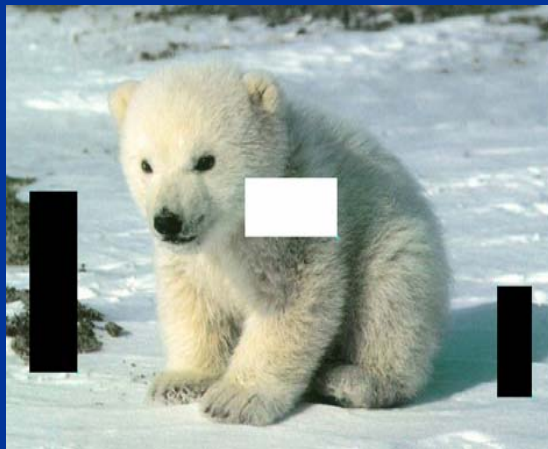
- Алгоритм Фалкерсона – последовательно ищем дополняющие пути в графе (пути от истока в сток, проходящие по ненасыщенным ребрам остаточного графа).
- При увеличении пропускных способностей ребер графа, можно использовать старый максимальный поток в качестве первого приближения.
 - Пропускные способности ребер могут даже уменьшиться, при условии что поток в каждом ребре не превышает его пропускной способности

Динамические марковские случайные поля (7)



$$Energy_{MRF} = \sum_p (\phi(I | F_p)) + \sum_{\{p,q\} \in N} (\phi(I | F_p, F_q) + \psi(F_p, F_q))$$

- Соответствует изменению пропускных способностей t-связей
- Приложения: интерактивная сегментация изображений



Исходное изображение

28.09.2009



Изначальная сегментация

<http://graphics.cs.msu.ru>



Улучшенная сегментация

Slide 56

Динамические марковские случайные поля (8)

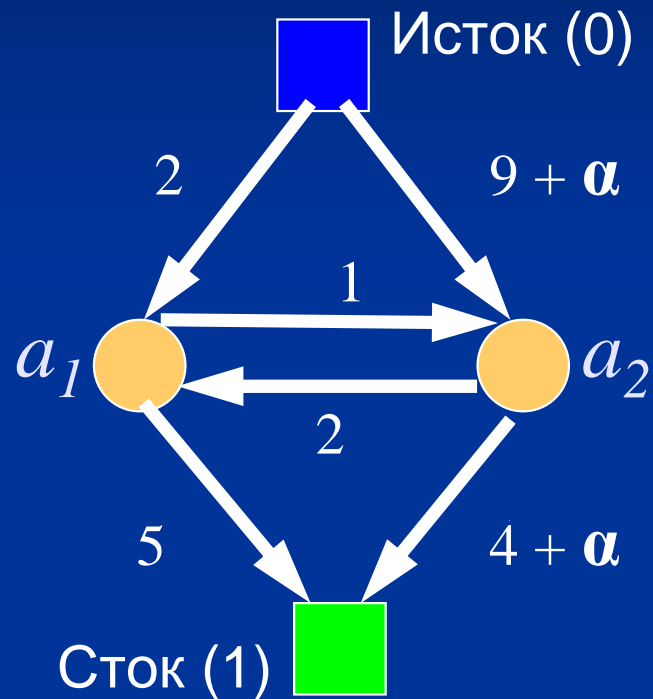


$$Energy_{MRF} = \sum_p (\phi(I | F_p)) + \sum_{\{p,q\} \in N} (\phi(I | F_p, F_q) + \psi(F_p, F_q))$$

- Соответствует изменению пропускных способностей n-связей
- Приложения: эффективная сегментация изображений в видео



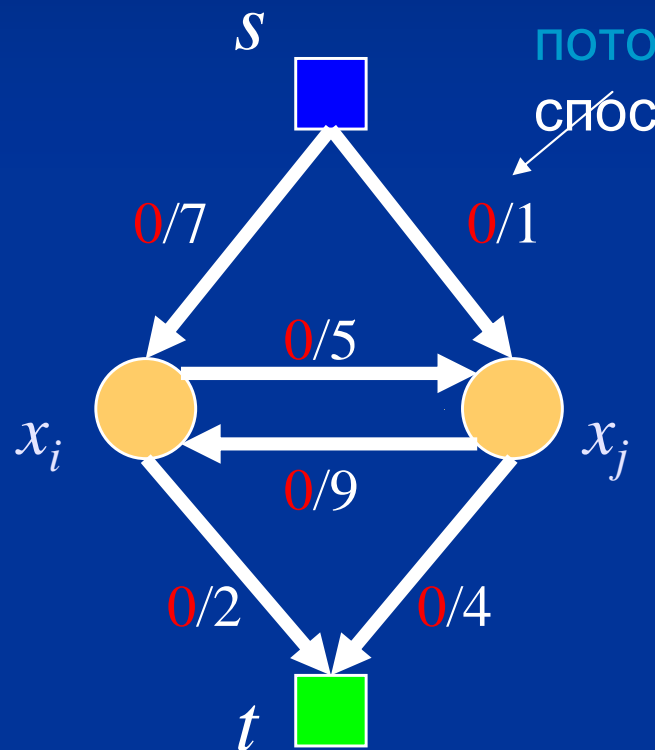
Динамические марковские случайные поля (9)



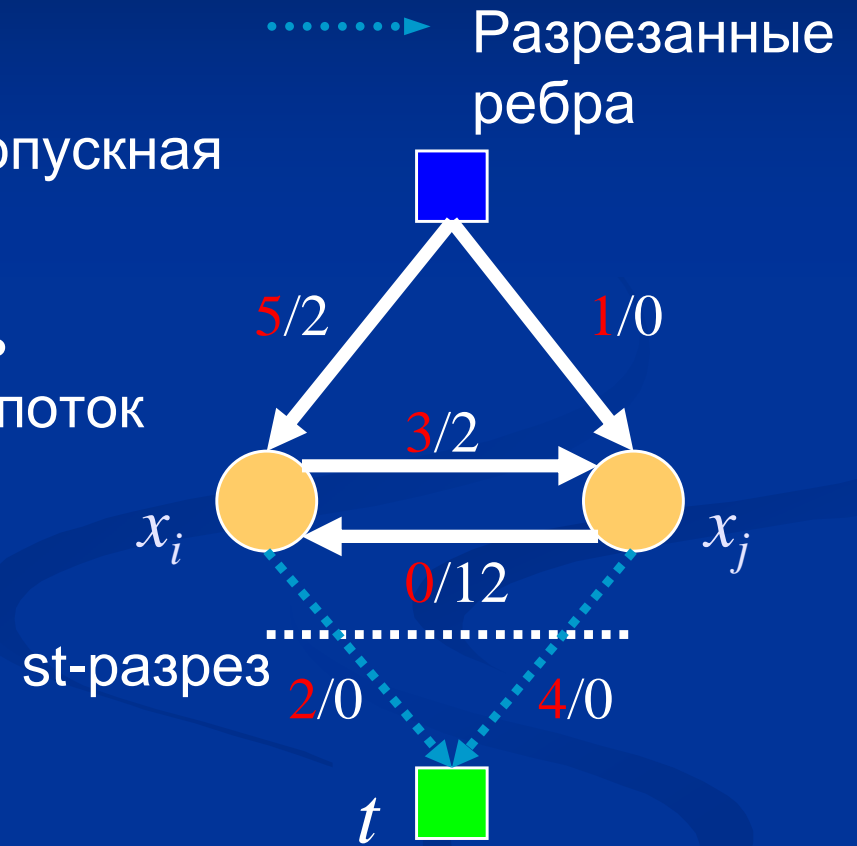
Ключевое наблюдение

Добавление константы к обеим t -связям вершины не меняет ребра, составляющие минимальный разрез.

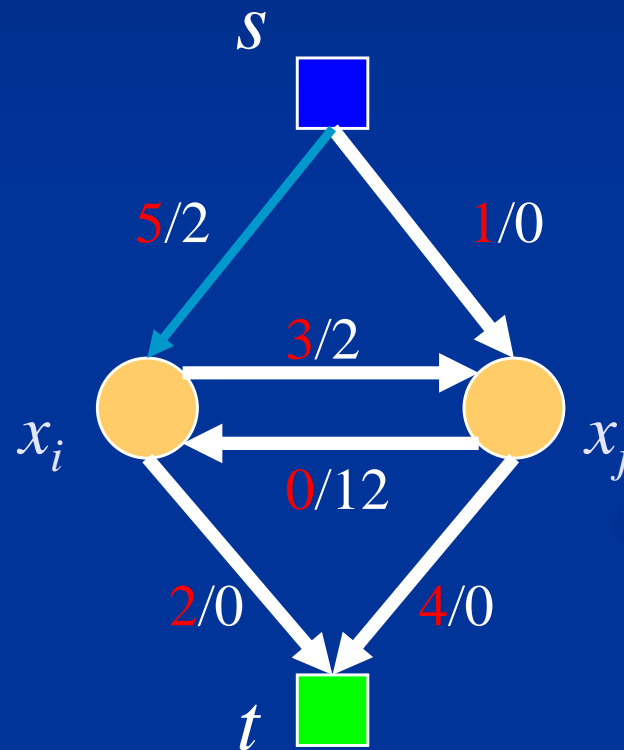
Динамические марковские случайные поля (10)



Вычислить
максимальный поток

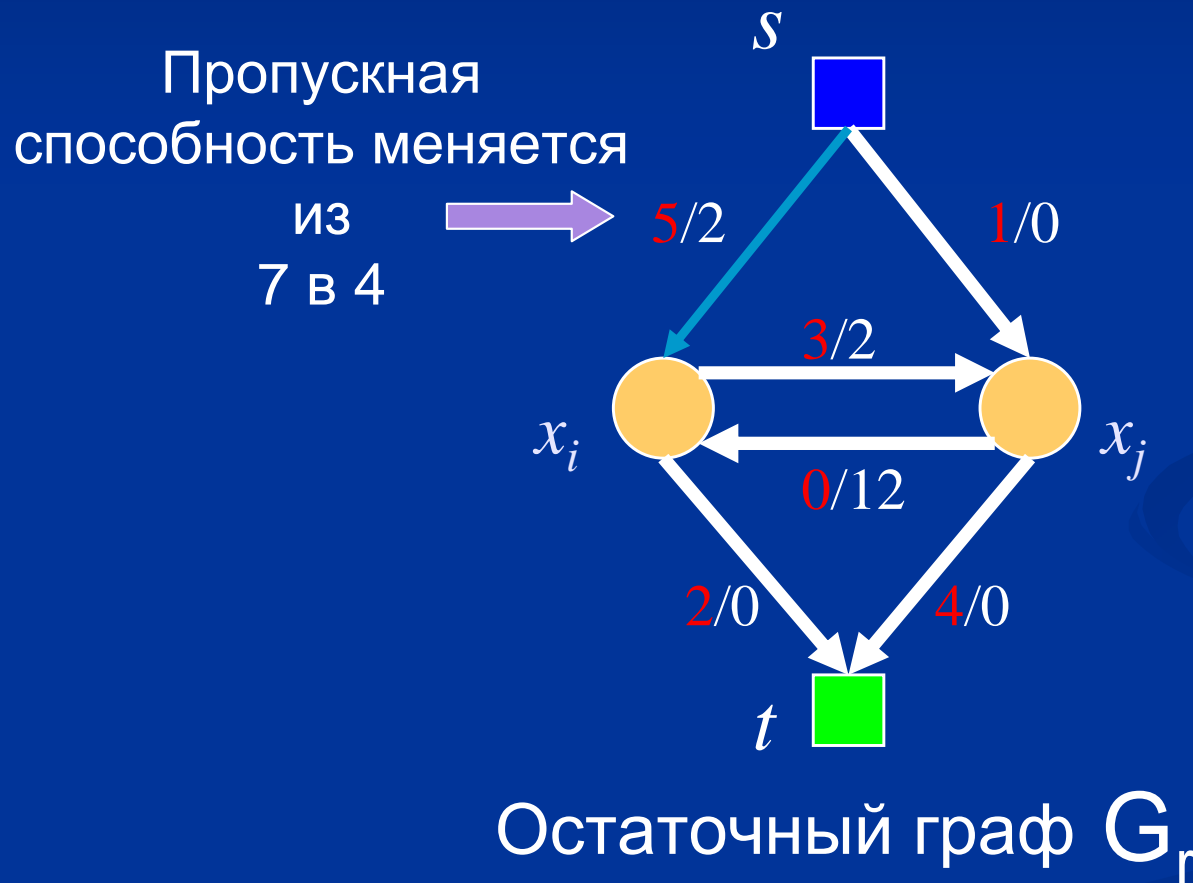


Динамические марковские случайные поля (11). Изменение t -связей.

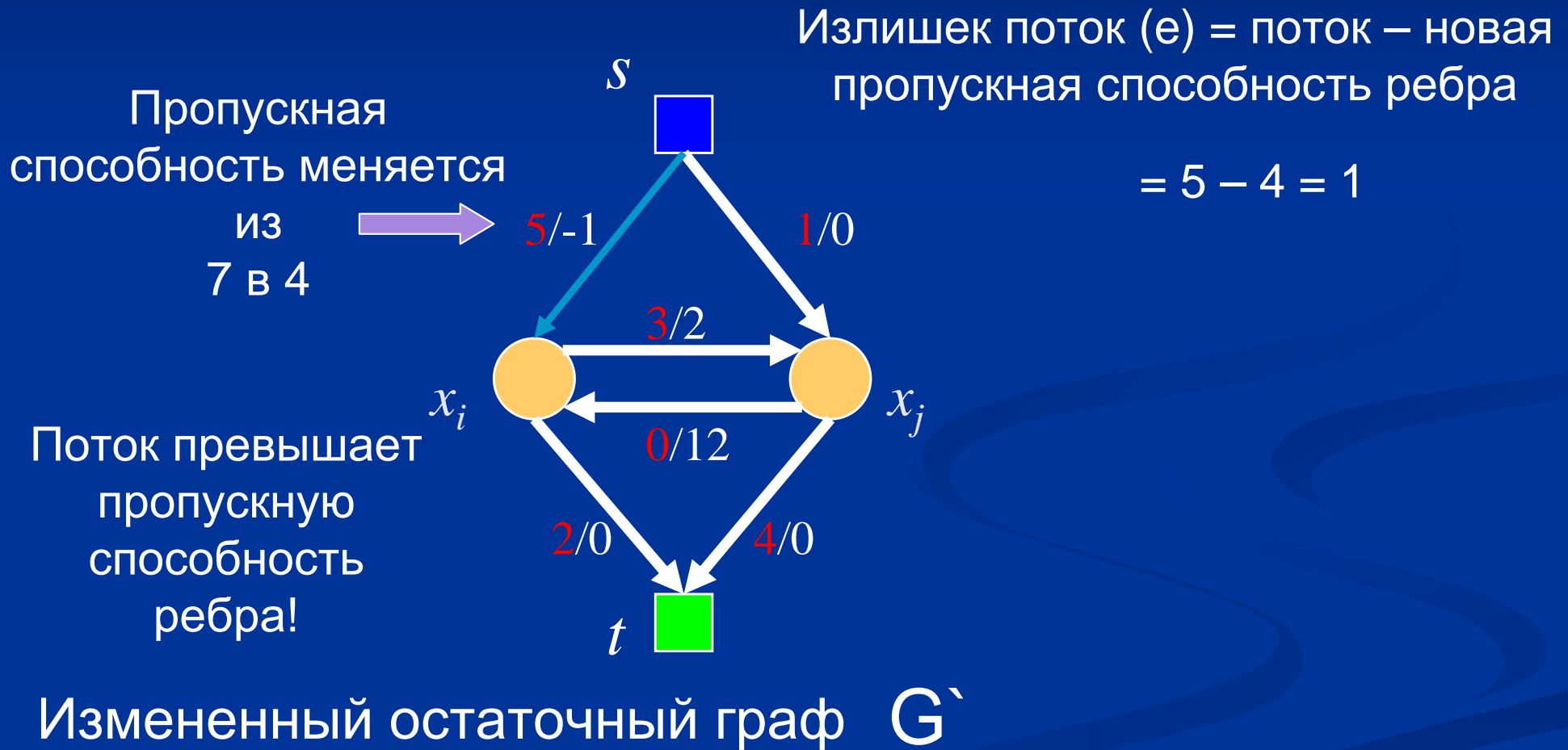


Остаточный граф G_r

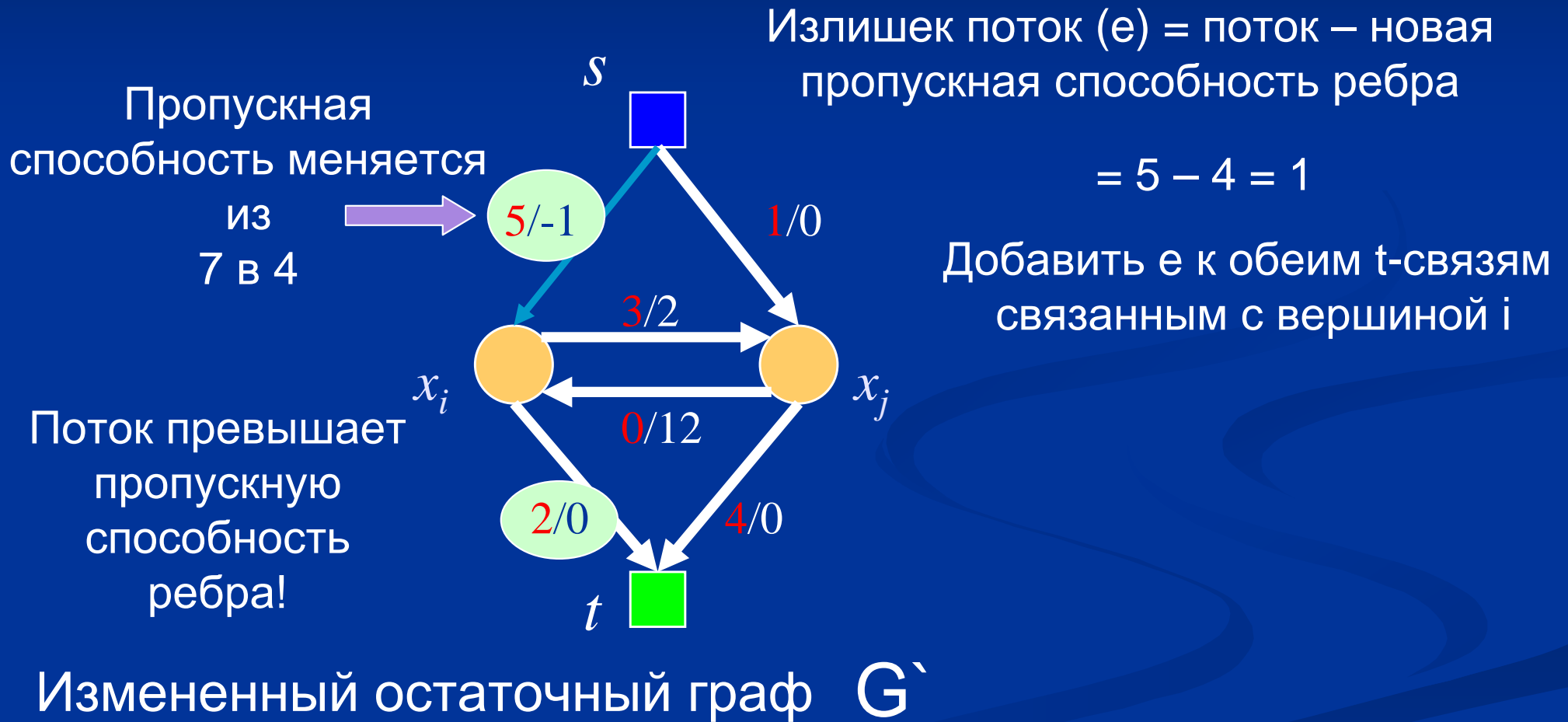
Динамические марковские случайные поля (11). Изменение t -связей.



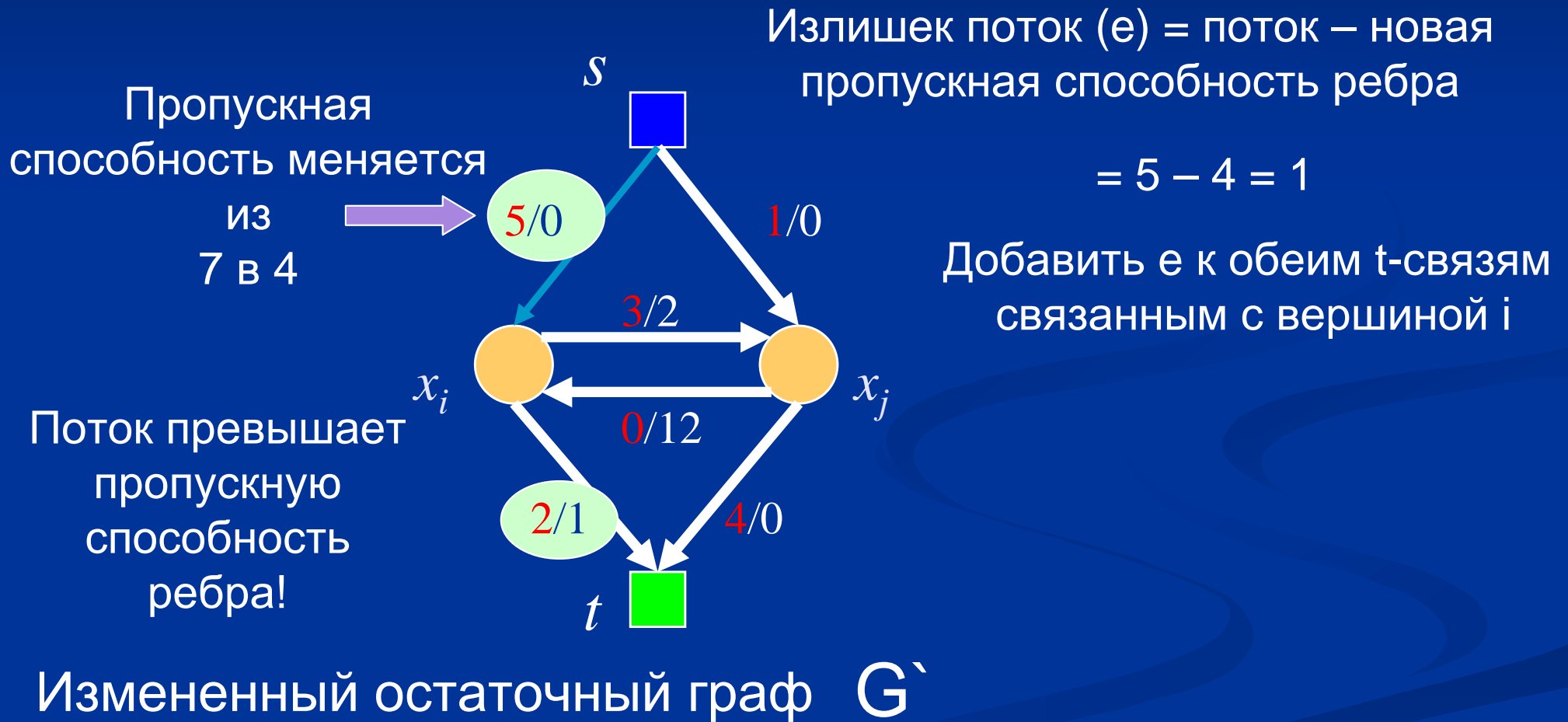
Динамические марковские случайные поля (11). Изменение t-связей.



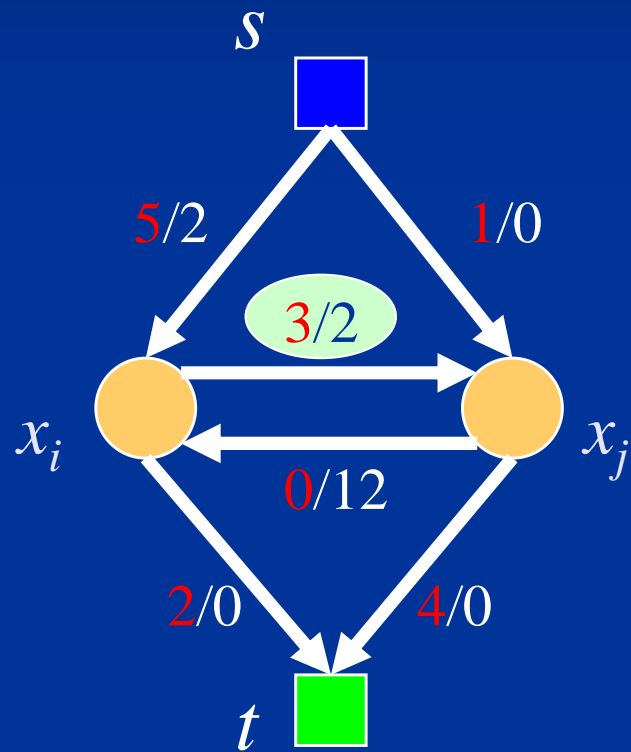
Динамические марковские случайные поля (11). Изменение t-связей.



Динамические марковские случайные поля (11). Изменение t-связей.



Динамические марковские случайные поля (12). Изменение n-связей.

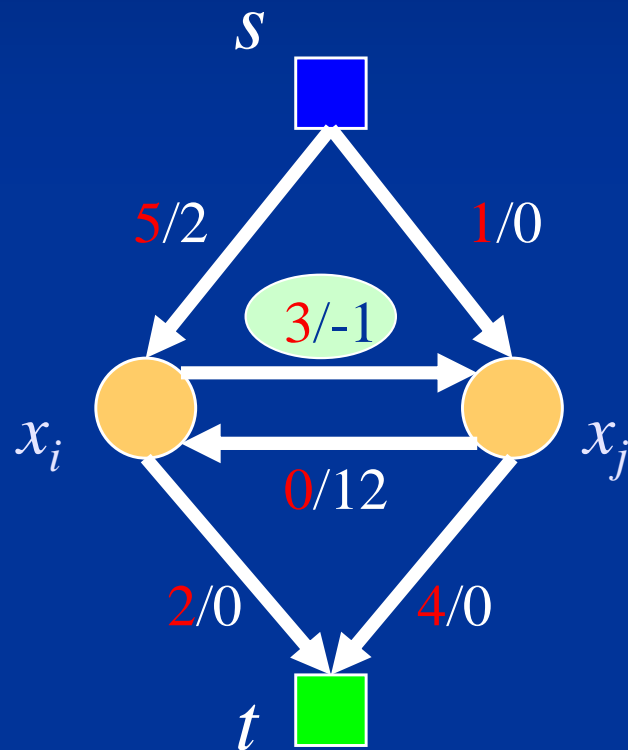


- Пропускная способность меняется из 5 в 2

Остаточный граф

G_r

Динамические марковские случайные поля (12). Изменение n-связей.

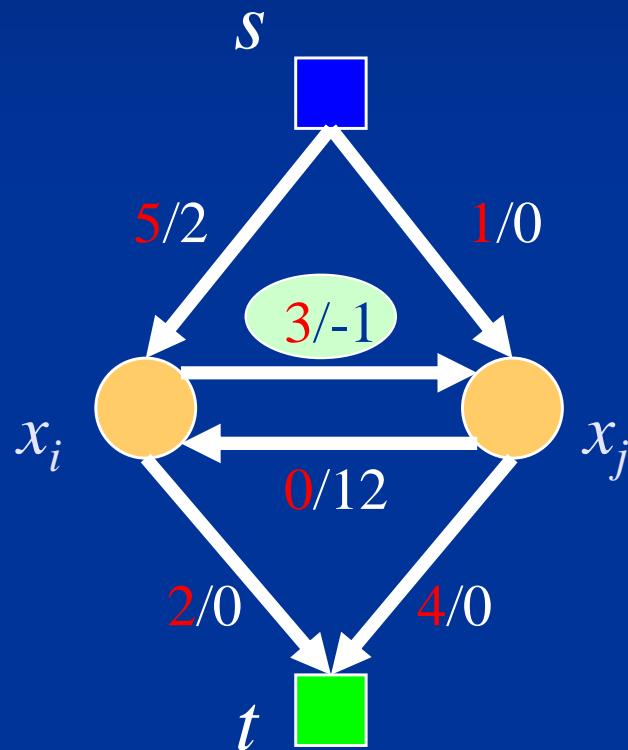


- Пропускная способность меняется из 5 в 2
- Поток превышает пропускную способность!

Измененный остаточный граф

G'

Динамические марковские случайные поля (12). Изменение n-связей.

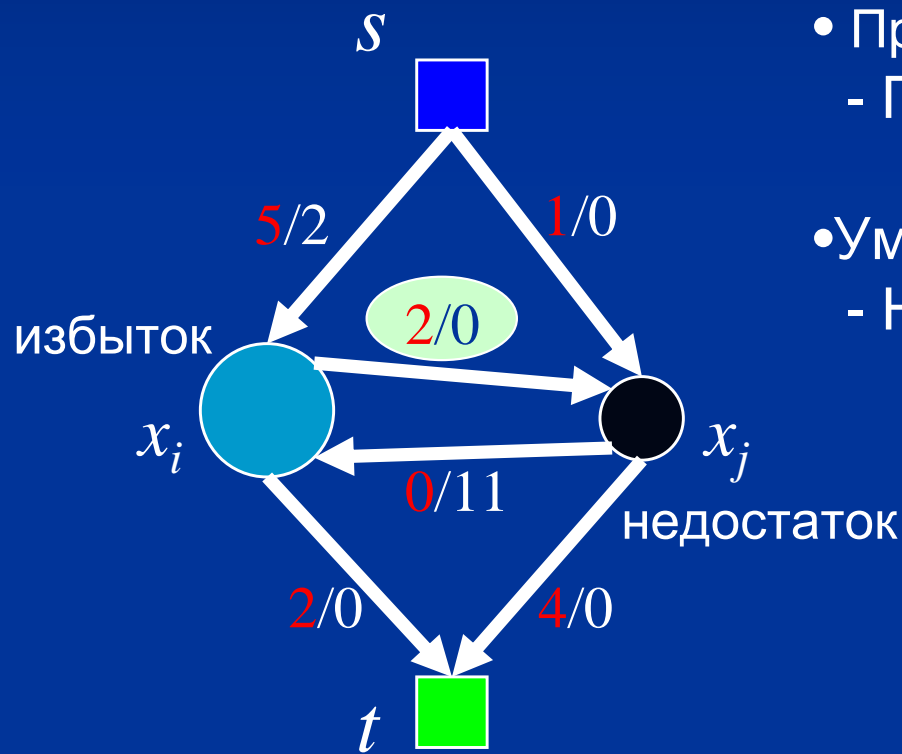


- Пропускная способность меняется из 5 в 2
- Поток превышает пропускную способность!
- Уменьшить поток до пропускной способности

Измененный остаточный граф

G'

Динамические марковские случайные поля (12). Изменение n-связей.

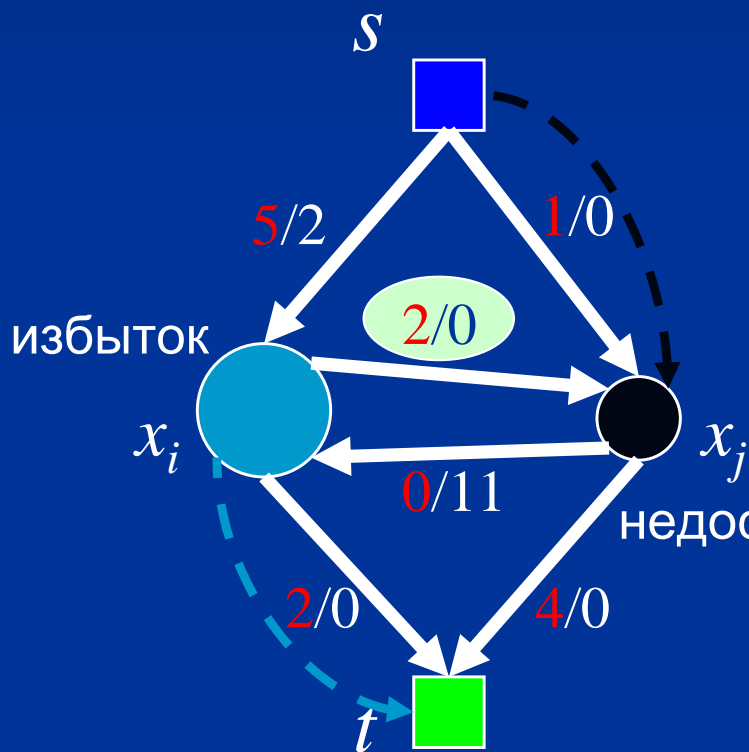


- Пропускная способность меняется из 5 в 2
- Поток превышает пропускную способность!
- Уменьшить поток до пропускной способности
- Нарушает условие сохранения поток

Измененный остаточный граф

G'

Динамические марковские случайные поля (12). Изменение n-связей.

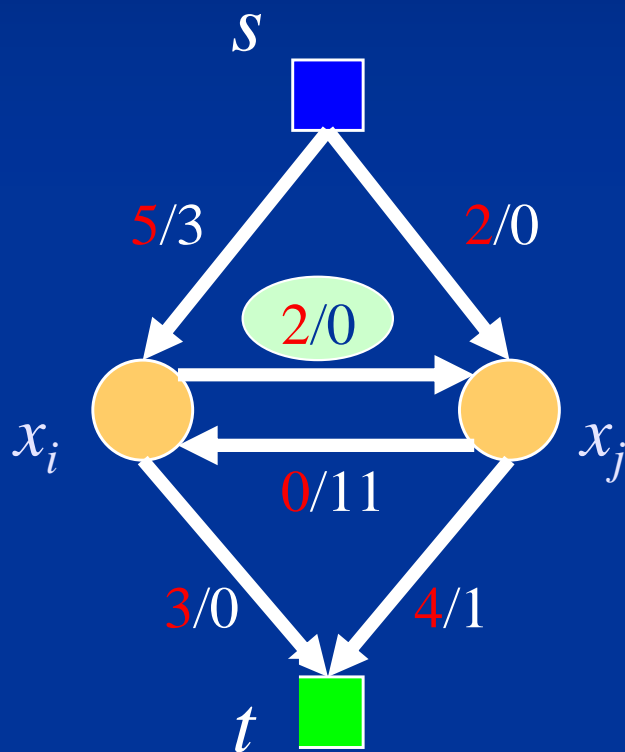


- Пропускная способность меняется из 5 в 2
- Поток превышает пропускную способность!
- Уменьшить поток до пропускной способности
- Нарушает условие сохранения поток
- Протолкнуть излишний поток в/из терминалов
- Создать необходимую пропускную способность, добавив $\alpha = \text{излишек}$ к обоим t -связям

Измененный остаточный граф

G'

Динамические марковские случайные поля (12). Изменение n -связей.



- Пропускная способность меняется из 5 в 2
- Поток превышает пропускную способность!
- Уменьшить поток до пропускной способности
- Нарушает условие сохранения поток
- Протолкнуть излишний поток в/из терминалов
- Создать необходимую пропускную способность, добавив α = излишек к обоим t -связям

Измененный остаточный граф

G'

Динамические марковские случайные поля (13). Результаты.

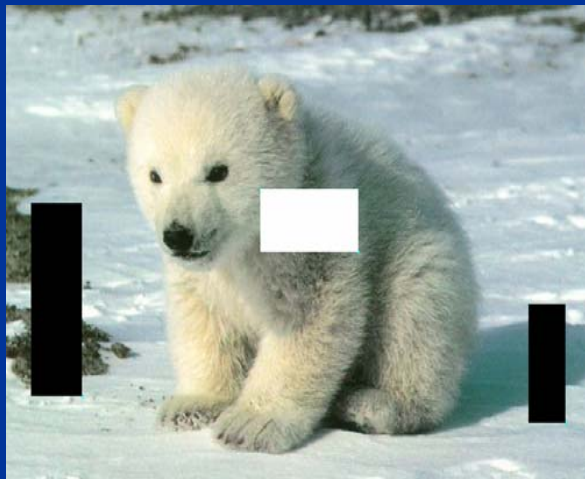


- На типичных видео-последовательностях наблюдается ускорение сегментации в 4-5 раз.

Динамические марковские случайные поля (14)



Интерактивная сегментация изображений
(изменение унарных членов)



Статический алгоритм: 150 мс.
динамический : 60 мс.
динамический (оптимизированный): 10 мс.

Динамические марковские случайные поля (15)



Сегментация изображений в видео (унарные & бинарные члены)

Разрешение кадра: 720x576

Статический : 190 msec
динамический : 140 msec
динамический (оптимизированный): 60 msec

Динамические разрезы графов

Разрезы графов



План лекции



- Элементы теории графов
- Алгоритмы нахождения максимального потока в графе
- Интерактивная сегментация изображений
- Какую энергию можно минимизировать с помощью разрезов графов?
- **Ускорения**
 - Динамические марковские случайные поля
 - Многоуровневые разрезы графов
 - Алгоритм touch-expand
- Итерационные алгоритмы
- Приложения

Многоуровневые разрезы графов (1)



H. Lombaert [ICCV 2005]

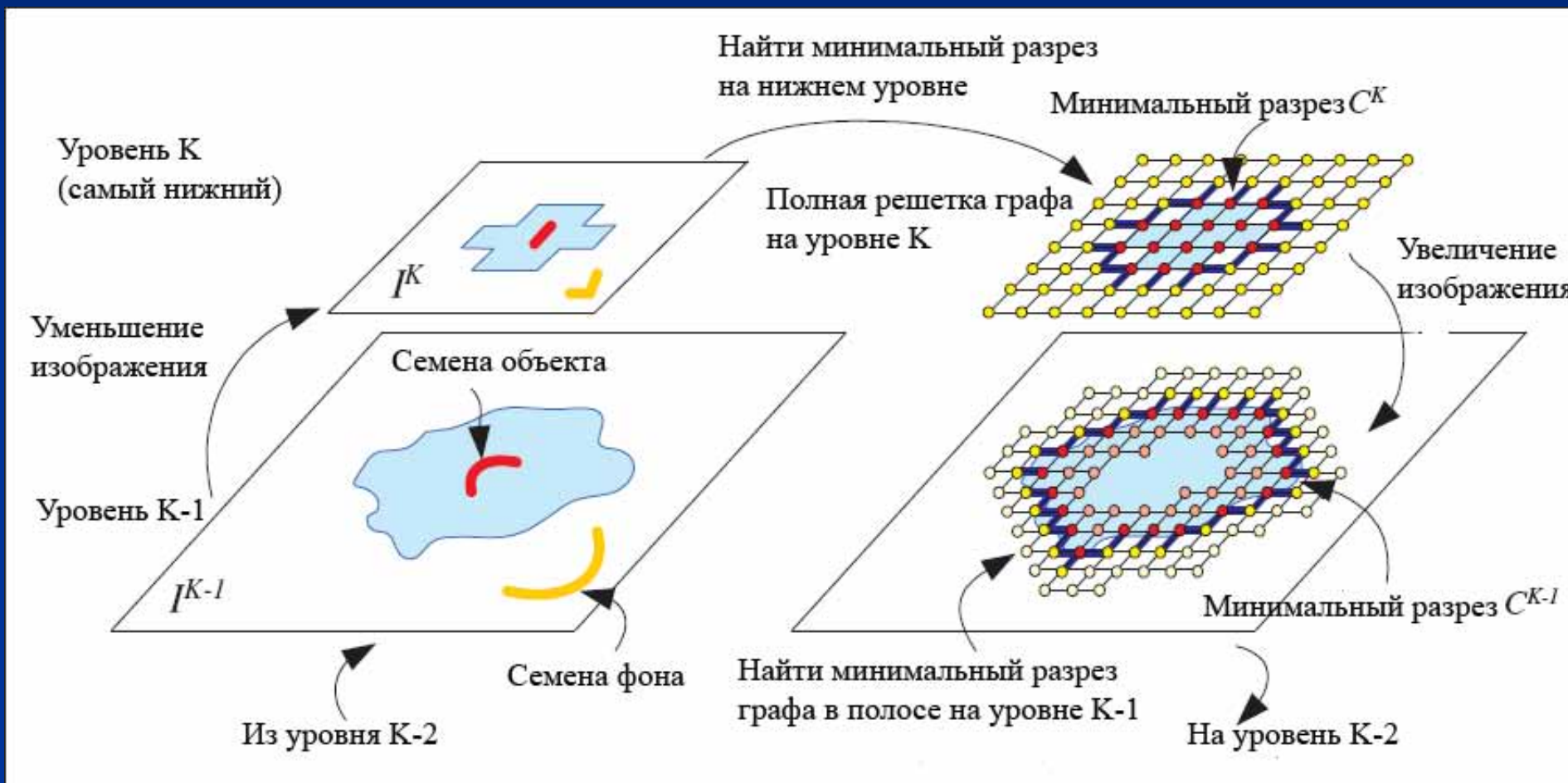
- Текущие алгоритмы нахождения максимального потока в графе требуют для своей работы очень много памяти. Алгоритм, предложенный Бойковым, требует $24|V| + 14|E|$ байт.
 - Для сегментации 3-мерного изображения 512^3 требуется более 8 Гб.
- Для сегментации больших изображений зачастую скорости данных алгоритмов тоже не хватает.

Многоуровневые разрезы графов (2)



- Идея: выполнить сегментацию уменьшенного изображения, после чего использовать полученный поток для уточнения сегментации на исходного разрешения.

Многоуровневые разрезы графов (3)



Многоуровневые разрезы графов (4)



- Не гарантирует глобального минимума энергии
 - На практике различия в сегментации составляют 2-3%

План лекции



- Элементы теории графов
- Алгоритмы нахождения максимального потока в графе
- Интерактивная сегментация изображений
- Какую энергию можно минимизировать с помощью разрезов графов?
- **Ускорения**
 - Динамические марковские случайные поля
 - Многоуровневые разрезы графов
 - Алгоритм touch-expand
- Итерационные алгоритмы
- Приложения

Ответы на возникшие вопросы (7)



Lempitsky, Boykov [CVPR 2007]

- Задача подгонки формы (shape fitting).
- Большие графы – слишком большие требования на память
- Многоуровневые разрезы графов не дают глобальный минимум



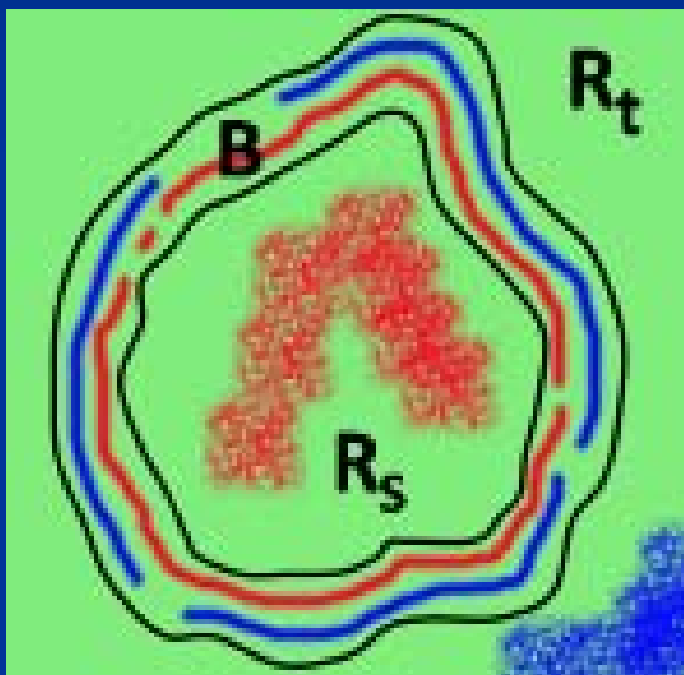
Слева-направо:

- Облако точек
- Результат с помощью многоуровневых разрезов графов
- Предложенный метод

Ответы на возникшие вопросы (8)



■ Алгоритм touch-expand

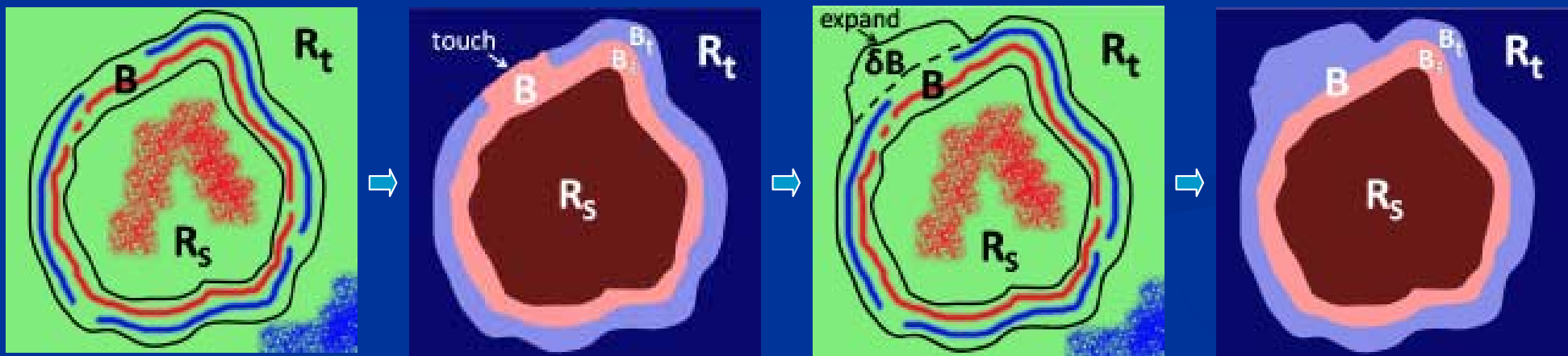


- Граф G разбивается на 3 части: B , R_s , R_t
 - B содержит s , t
 - Вершины из R_s (R_t) не связаны напрямую через t -связи с терминальной вершиной t (s)
 - R_s «не касается» R_t (посредством n -связей)

Ответы на возникшие вопросы (9)



- Алгоритм touch-expand



Гарантирует глобальный минимум!!!

Ответы на возникшие вопросы (10)



- Алгоритм touch-expand
 - Итоговый размер B – от 2% до 22% от всего графа G .
 - Нет гарантированного уменьшения графа.
 - Граф можно разложить на B , R_s , R_t (например в интерактивной сегментации – нельзя!!!)

План лекции



- Элементы теории графов
- Алгоритмы нахождения максимального потока в графе
- Интерактивная сегментация изображений
- Какую энергию можно минимизировать с помощью разрезов графов?
- Ускорения
 - Динамические марковские случайные поля
 - Многоуровневые разрезы графов
 - Алгоритм touch-expand
- Итерационные алгоритмы
- Приложения

Итерационные алгоритмы (Move algorithms) (1)



- Общий вид энергии:

$$E(F) = \sum_{p \in P} R_p(F_p) + \sum_{\{p,q\} \in N} V_{\{p,q\}}$$

$$F_p \in \{1, \dots, K\} \quad K > 2$$

$$\arg \min_F E(F) - ?$$



Итерационные алгоритмы (2)

- Разбиение пикселей изображения $P = \{P_l \mid l \in L\}$,
где $P_l = \{p \in P \mid f_p = l\}$ - подмножество пикселей с меткой l .
- V называется полу-метрикой на пространстве меток L , если
 - $V(\alpha, \beta) = V(\beta, \alpha) \geq 0 \quad \forall \alpha, \beta \in L$
 - $V(\alpha, \beta) = 0 \Leftrightarrow \alpha = \beta$
- V называется метрикой на пространстве меток L , если она является полу-метрикой, и
 - $V(\alpha, \beta) \leq V(\alpha, \gamma) + V(\gamma, \beta) \quad \forall \alpha, \beta, \gamma \in L$

Итерационные алгоритмы (3)



- **Опр. 1** Пусть α, β - метки. Изменение разбиения P (маркировки F) на новое разбиение P' (маркировку F') называется $\alpha - \beta$ заменой, если $P_l = P'_l$ для любой метки $l \neq \alpha, \beta$.
- **Опр. 2** Пусть имеется метка α . Изменения разбиения P (маркировки F) на новое разбиение P' (маркировку F') называется α -расширением, если $P_\alpha \subset P'_\alpha$ и $P'_l \subset P_l$ для любой метки $l \neq \alpha$.



Итерационные алгоритмы (4)

- Алгоритм, основанный на $\alpha - \beta$ замене.
 1. Начать с произвольной маркировки F
 2. Успех := 0
 3. Для каждой пары меток $\{\alpha, \beta\} \subset L$
 - a. Найти $\hat{F} = \arg \min E(F')$ среди всех F' , являющихся $\alpha - \beta$ заменой F
 - b. Если $E(\hat{F}) < E(F)$, установить $F := \hat{F}$ и Успех := 1
 4. Если Успех = 1, goto 2
 5. Вернуть F

- V должна быть полу-метрикой.



Итерационные алгоритмы (8)

- Алгоритм, основанный на α замене.
 1. Начать с произвольной маркировки F
 2. Успех := 0
 3. Для каждой метки $\alpha \in L$
 - a. Найти $\hat{F} = \arg \min E(F')$ среди всех F' , являющихся α расширением F
 - b. Если $E(\hat{F}) < E(F)$, установить $F := \hat{F}$ и Успех := 1
 4. Если Успех = 1, goto 2
 5. Вернуть F

- V должна быть метрикой!!!



Итерационные алгоритмы (9)

- Алгоритм, основанный на α замене.
- выдает маркировку F , для которой

$$E(F^*) \leq E(F) \leq 2k \cdot E(F^*)$$

где F^* - глобальный минимум,

$$k = \frac{\max\{V(\alpha, \beta) : \alpha \neq \beta\}}{\min\{V(\alpha, \beta) : \alpha \neq \beta\}}$$

Итерационные алгоритмы (10)



■ Примеры:

■ $E_1 : V_{\{p,q\}}(F_p, F_q) = \min(K, |F_p - F_q|^2)$, $K = const$
- полуметрика -> $\alpha - \beta$ замена

■ $E_2 : V_{\{p,q\}}(F_p, F_q) = \min(K, |F_p - F_q|)$

■ $E_3 : V_{\{p,q\}}(F_p, F_q) = \delta(F_p \neq F_q)$
- метрика -> α расширение

План лекции



- Элементы теории графов
- Алгоритмы нахождения максимального потока в графе
- Интерактивная сегментация изображений
- Какую энергию можно минимизировать с помощью разрезов графов?
- Ускорения
 - Динамические марковские случайные поля
 - Многоуровневые разрезы графов
 - Алгоритм touch-expand
- Итерационные алгоритмы
- Приложения

Приложения. Восстановление изображений (image restoration) (1)



Приложения. Восстановление изображений (2)



- Марковское случайное поле:
 - Вершины – пиксели.
 - Метки – цвет (яркость) пикселей

$$E(F) = \lambda \sum_{p \in P} R_p(F_p) + \sum_{\{p,q\} \in N} V_{\{p,q\}}(F_p, F_q)$$

- $R_p(F_p) = (F_p - I_p)^2$, где I_p - цвет пикселя p
- $V_{\{p,q\}}(F_p, F_q) = \min(K, |F_p - F_q|^2)$, $K = const$

Приложения. Восстановление изображений (3)



■ Исходное



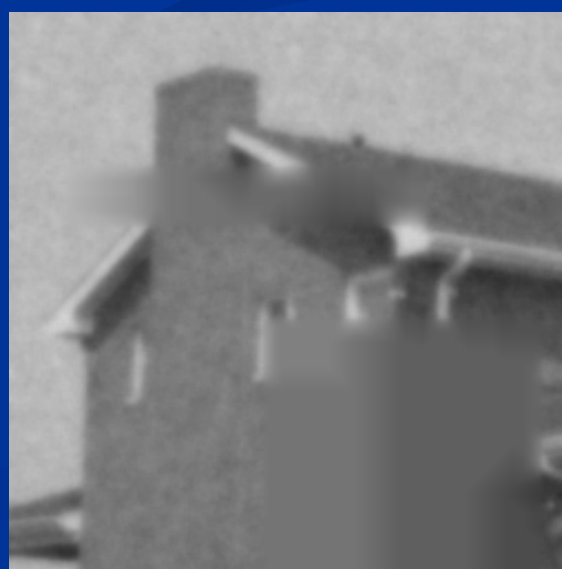
■ Входное



■ α расширение



■ $\alpha - \beta$ замена



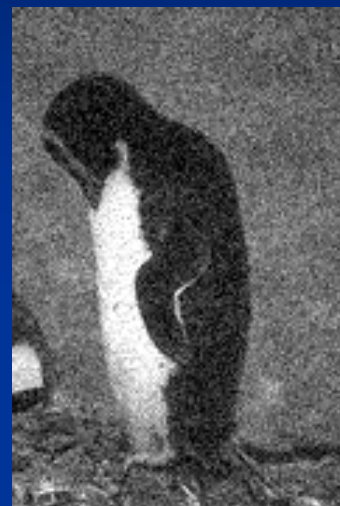
Приложения. Восстановление изображений (4)



■ Исходное



■ Входное



■ α расширение



■ $\alpha - \beta$ замена



Приложения. Сегментация изображений (1)



Приложения. Сегментация изображений (2)



- Марковское случайное поле:
 - Вершины – пиксели (суперпиксели).
 - Метки – класс объекта (в бинарном случае – ‘фон’/’объект’)

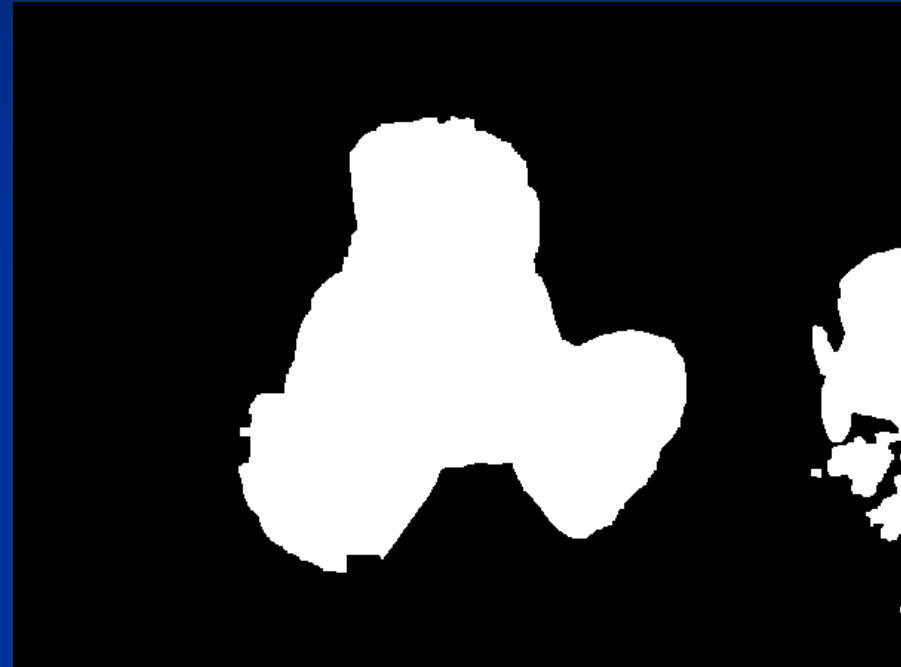
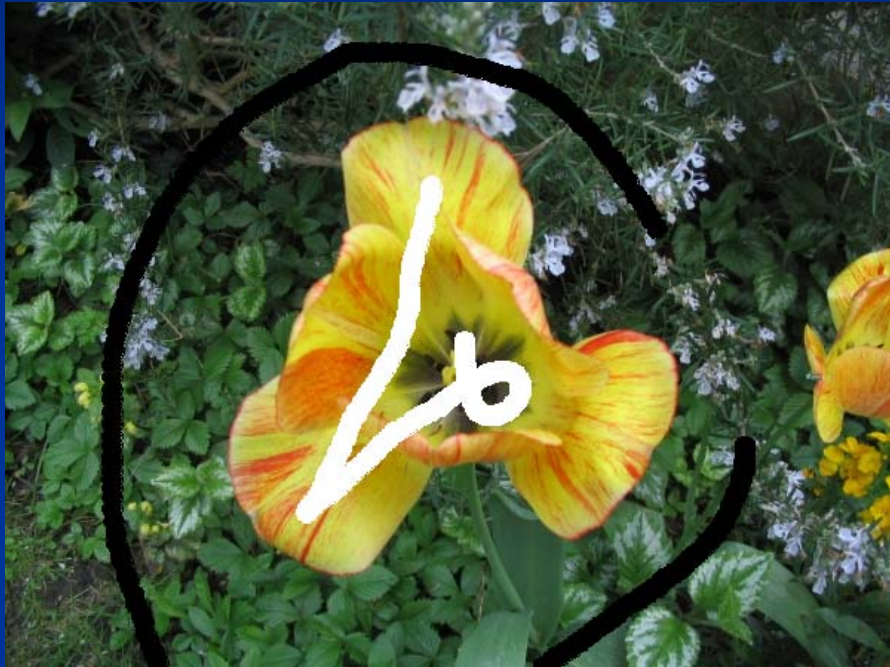
$$E(F) = \lambda \sum_{p \in P} R_p(F_p) + \sum_{\{p,q\} \in N} V_{\{p,q\}}(F_p, F_q)$$

- $R_p(F_p) = -\ln \Pr(I_p | F_p)$, где I_p - цвет пикселя

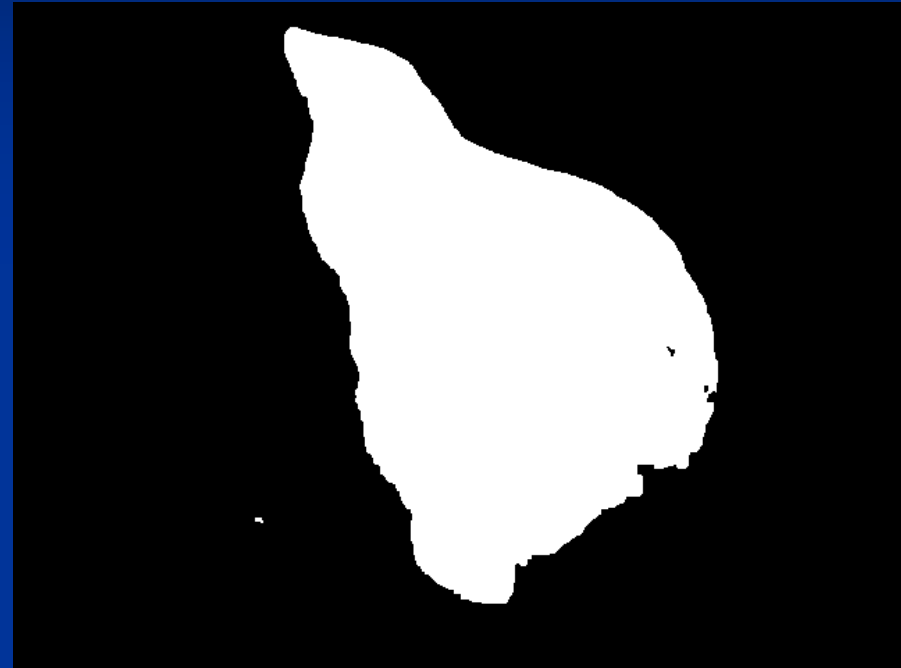
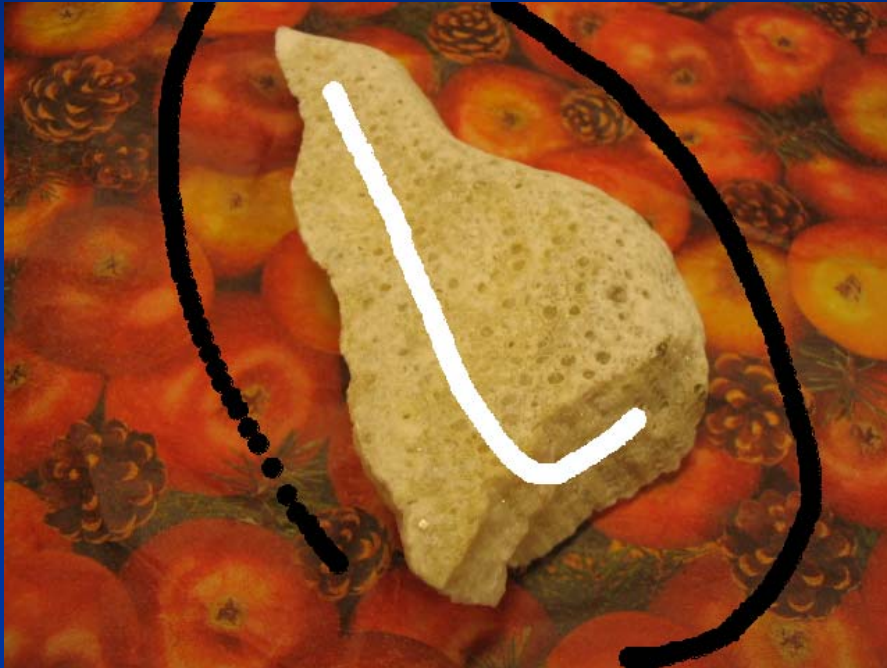
- $V_{\{p,q\}}(F_p, F_q) = \exp\left(-\frac{(I_p - I_q)^2}{2\sigma^2}\right) \cdot \frac{1}{\text{dist}(p, q)} \cdot \delta(F_p, F_q)$,

$$\sigma = \text{const}$$

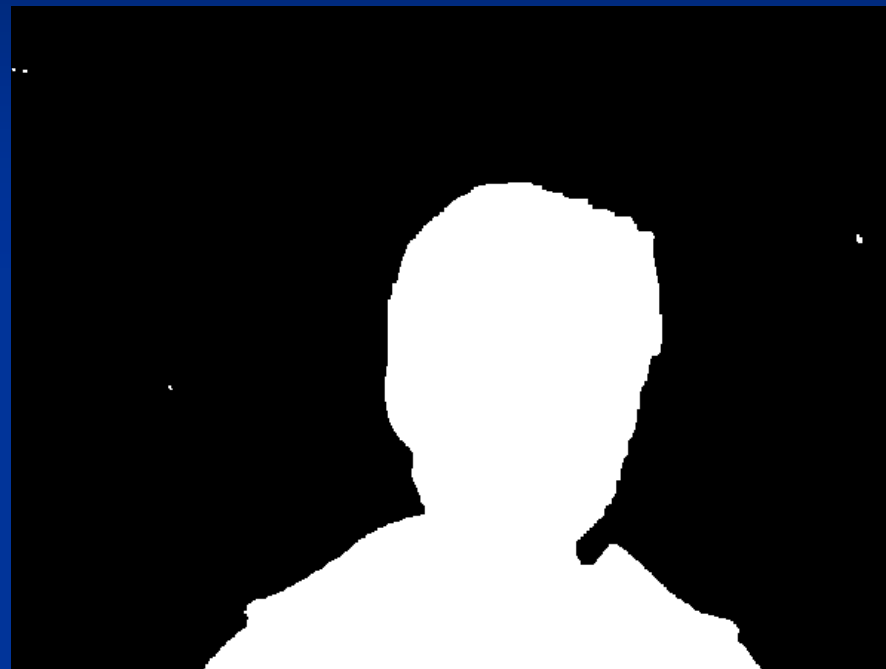
Приложения. Сегментация изображений (3)



Приложения. Сегментация изображений (3)



Приложения. Сегментация изображений (3)



Приложения. Сегментация видео (1)

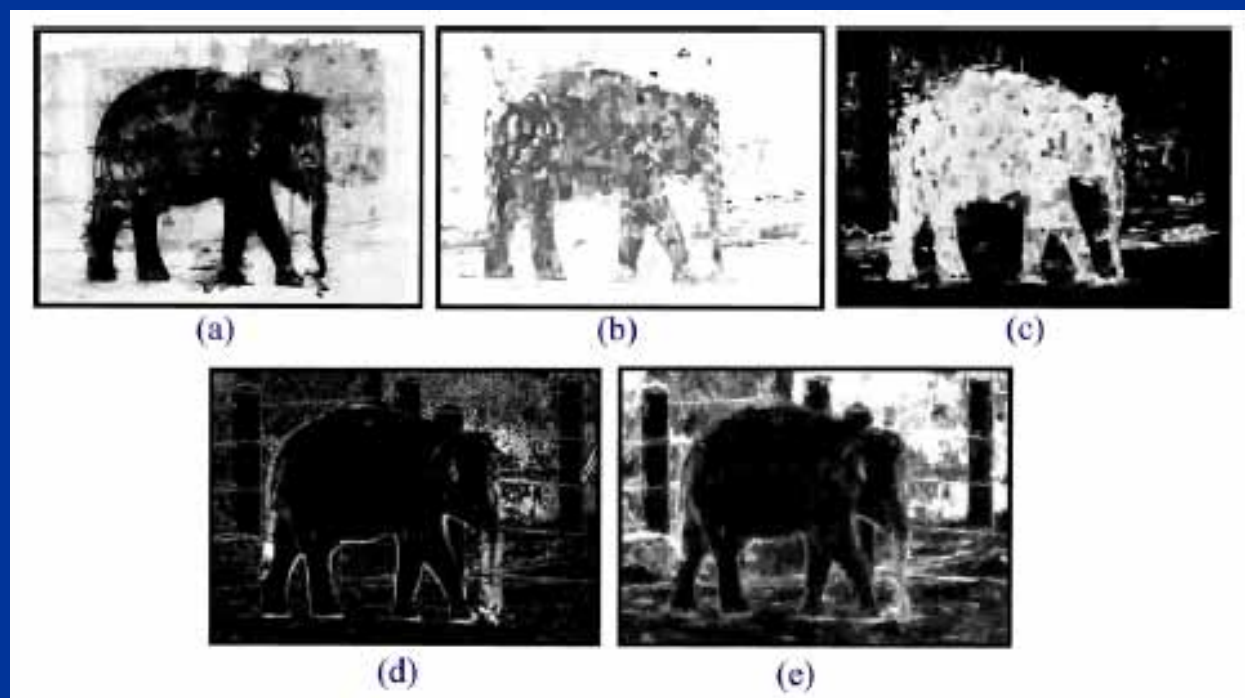


Приложения. Сегментация видео

(2)



- Марковское случайное поле:
 - Вершины – суперпиксели.
 - Метки – класс объекта (в бинарном случае – ‘фон’/’объект’)



- (a) Локальная цветовая статистика фона
- (b) Глобальная цветовая статистика фона
- (c) Глобальная цветовая статистика объекта
- (d) Локальная оценка границ
- (e) Глобальная оценка границ

Приложения. Сегментация видео (3)



Приложения. Сшивка изображений. Фотомонтаж. (1)



Приложения. Сшивка изображений. Фотомонтаж. (2)



- Марковское случайное поле:
 - Вершины – пиксели.
 - Метки – номер изображения, из которого берется данный пиксель.

$$E(F) = \lambda \sum_{p \in P} R_p(F_p) + \sum_{\{p,q\} \in N} V_{\{p,q\}}(F_p, F_q)$$

- $R_p(F_p)$ - зависит от задачи, может отсутствовать

- $V(F_s, F_t) = \left\| I_{F_s}(s) - I_{F_t}(s) \right\| + \left\| I_{F_s}(t) - I_{F_t}(t) \right\|$, где

$I_{F_s}(t)$ - цвет изображения номер F_s в пикселе t

Приложения. Сшивка изображений. Фотомонтаж. (3)



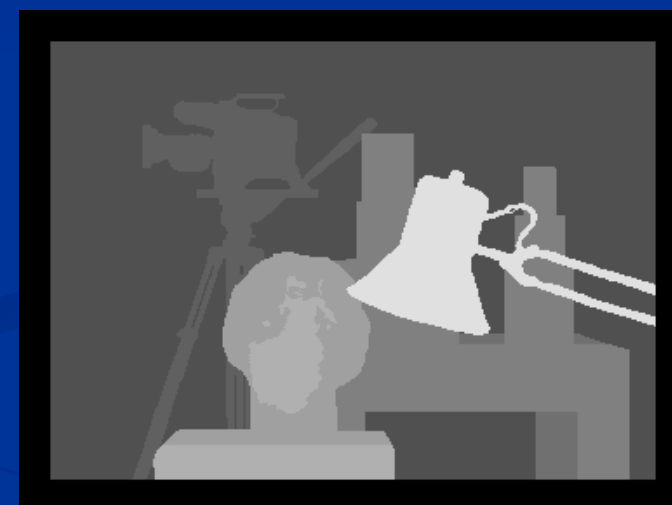
Приложения. Сшивка изображений. Фотомонтаж. (4)



Приложения. Сшивка изображений. Фотомонтаж. (5)



Приложения. Стерео (1)



Приложения. Стереос (2)



- Марковское случайное поле:
 - Вершины – пиксели первого изображения.
 - Метки – диспаритет пикселя.

$$E(F) = \lambda \sum_{p \in P} R_p(F_p) + \sum_{\{p,q\} \in N} V_{\{p,q\}}(F_p, F_q)$$

- $R_p(F_p) = \min(|I_2(p + F_p) - I_1(p)|, K_1), K_1 = const$
- $V_{\{p,q\}}(F_p, F_q) = \min((F_p - F_q)^2, K_2), K_2 = const$

$I_m(p)$ - цвет изображения номер m в пикселе p

Приложения. Стерео (3)

