
*Discrete Optimization Methods
in Computer Vision*

CSE 6389

Slides by: Boykov

Modified and Presented by: Mostafa Parchami

Basic overview of graph cuts

[Yuri Boykov, Olga Veksler, Ramin Zabih, Fast
Approximation Energy Minimization via
Graph Cuts, PAMI'01]

Disclaimer

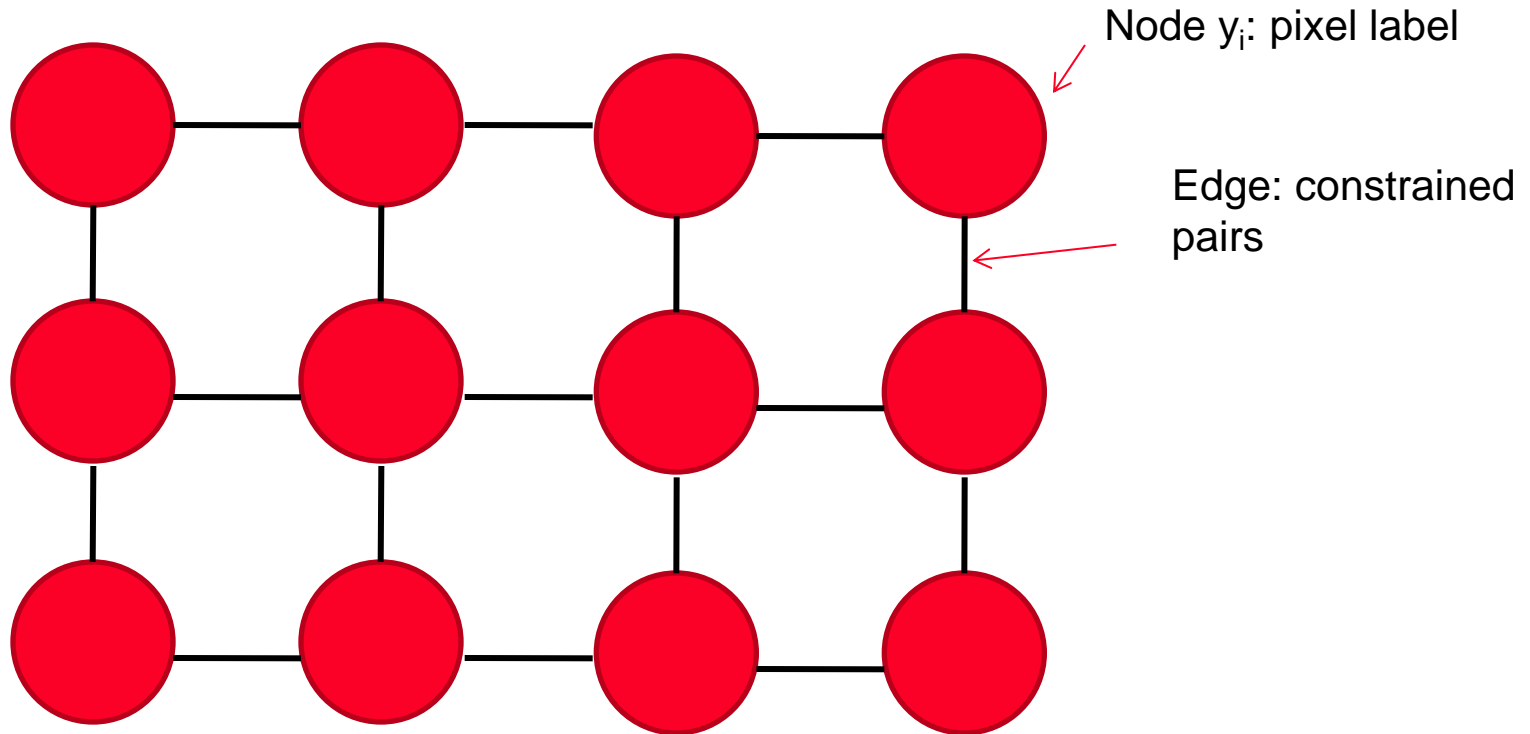
- Can not possibly cover all discrete optimization methods widely used in computer vision in the last 30 years
- Mainly concentrate on
 - Discrete energy minimization methods that can be applied to Markov Random Fields with binary or n-labels
 - applicable to a wide spectrum of problems in vision
 - Methods motivated by LP relaxations
 - good bounds on the solutions

Discrete Optimization Methods in Computer Vision

■ **Basic overview of graph cuts**

- binary labeling
 - a few basic examples
 - energy optimization
 - submodularity (**discrete view**)
 - continuous functionals (**geometric view**)
 - posterior MRF energy (**statistical view**)
- extensions to multi-label problems
 - interactions: convex, robust, metric
 - move-based optimization

Markov Random Fields in Computer Vision

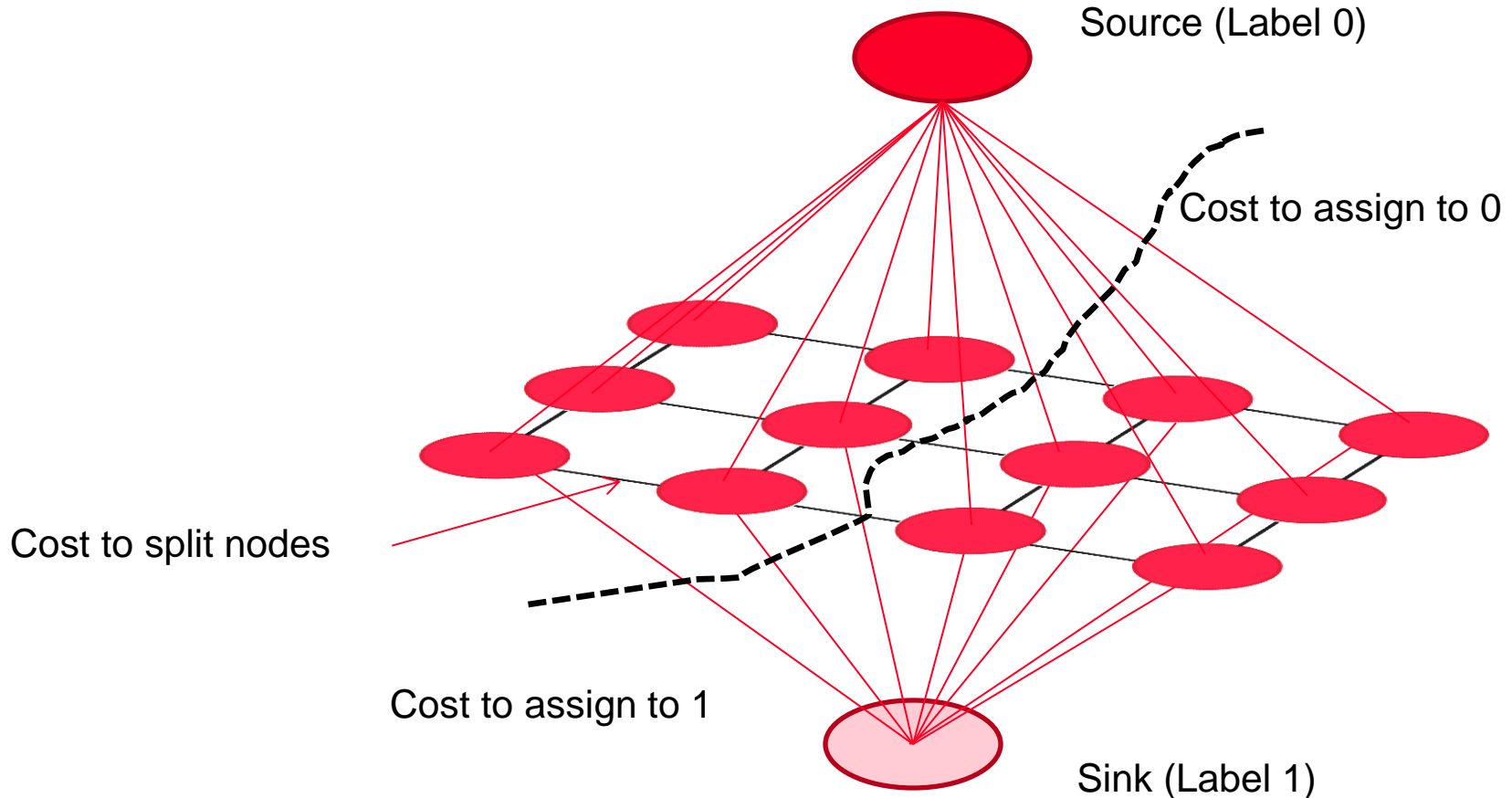


Cost to assign a label to each pixel

Cost to assign a pair of labels to connected pixels

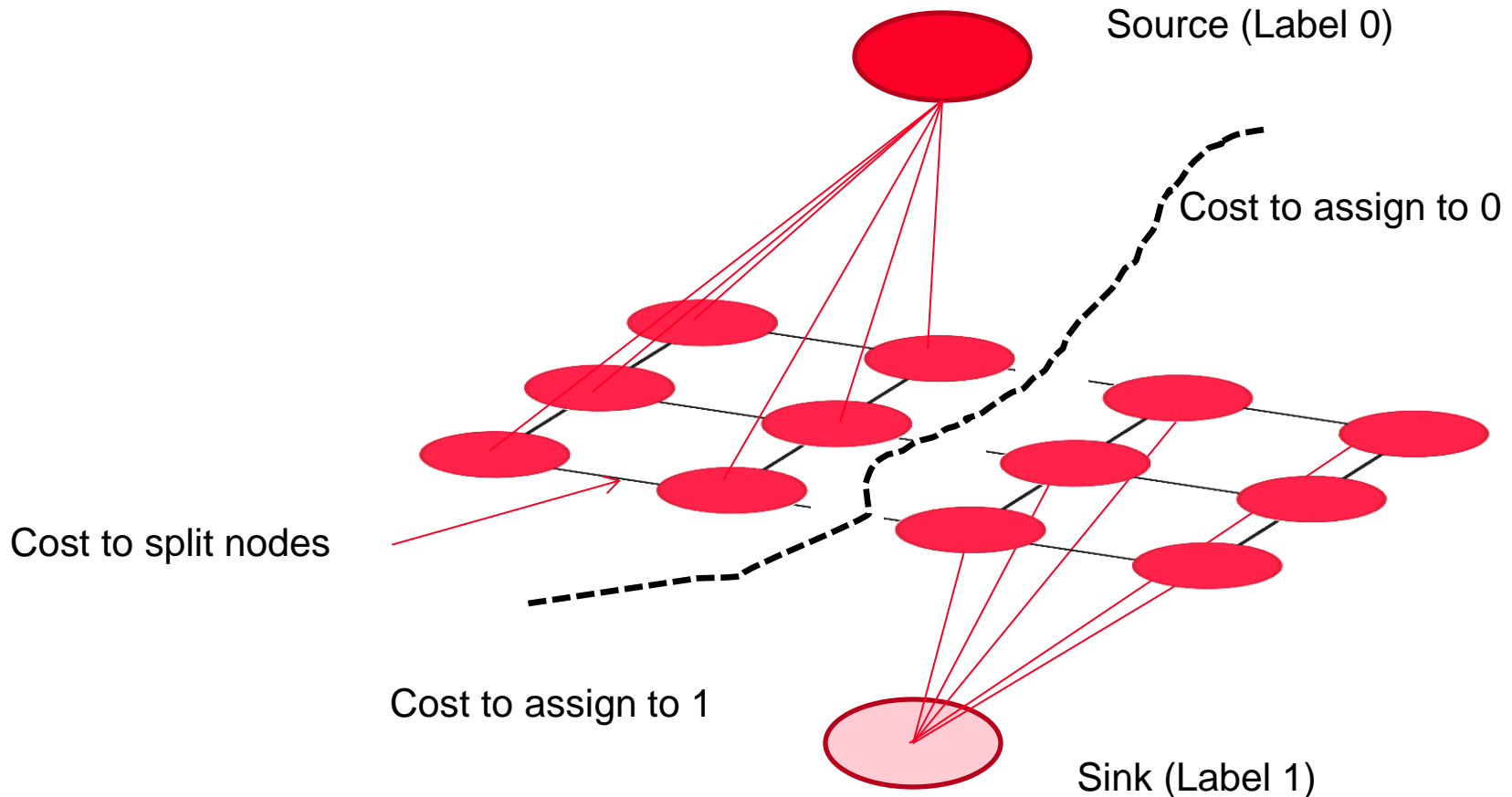
$$Energy(\mathbf{y}; \theta, data) = \sum_i \psi_1(y_i; \theta, data) \sum_{i,j \in edges} \psi_2(y_i, y_j; \theta, data)$$

Solving MRF with Graph Cuts



$$Energy(\mathbf{y}; \theta, data) = \sum_i \psi_1(y_i; \theta, data) + \sum_{i,j \in edges} \psi_2(y_i, y_j; \theta, data)$$

Solving MRF with Graph Cuts Cont.

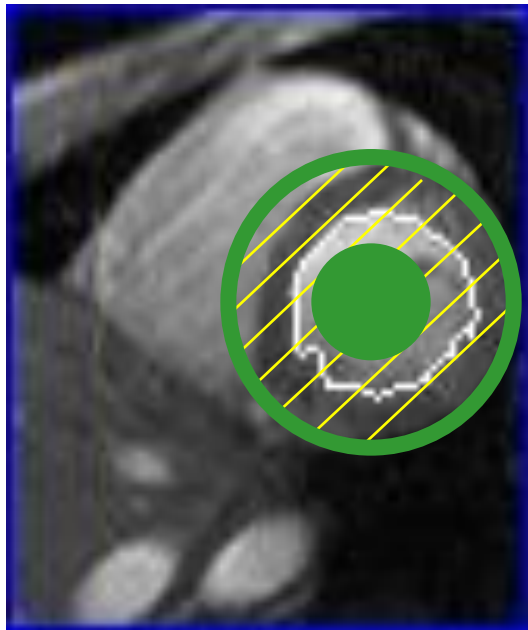


$$Energy(\mathbf{y}; \theta, data) = \sum_i \psi_1(y_i; \theta, data) + \sum_{i,j \in edges} \psi_2(y_i, y_j; \theta, data)$$

2D Graph cut \Leftrightarrow shortest path on a graph

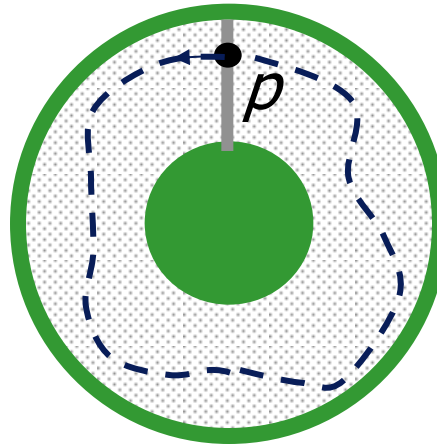
Example:

find the shortest closed contour in a given domain of a graph



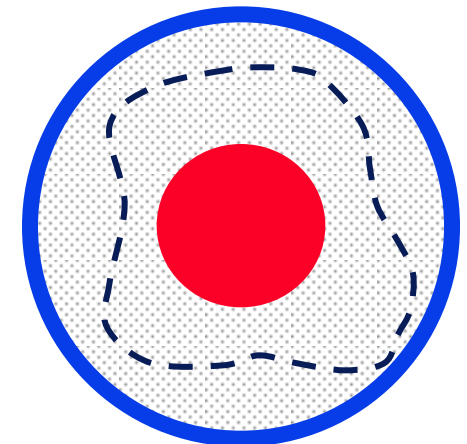
Shortest paths approach

(live wire, intelligent scissors)



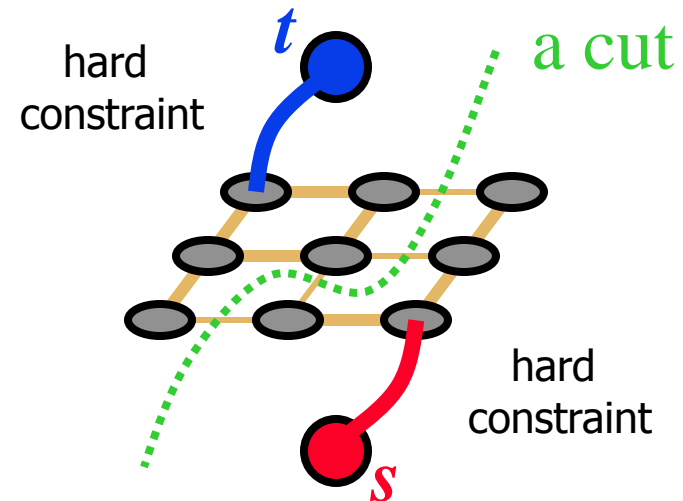
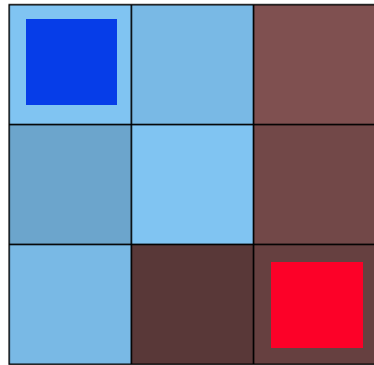
Compute the *shortest path* $p \rightarrow p$ for a point p .
Repeat for all points on the gray line. Then choose the optimal contour.

Graph Cuts approach

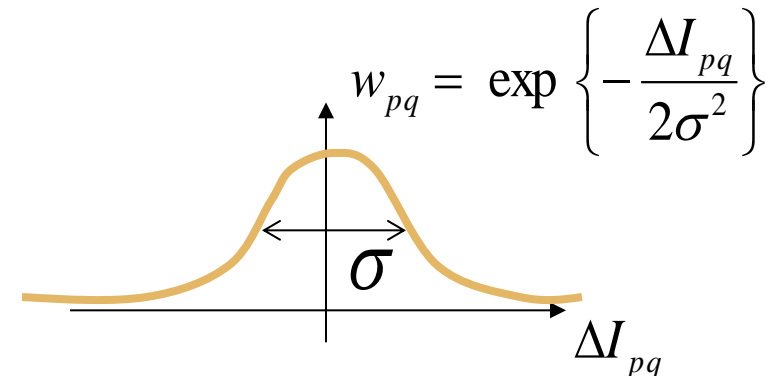


Compute the *minimum cut* that separates red region from blue region

Graph cuts for optimal boundary detection (B&J, ICCV'01)



Minimum cost cut can be
computed in polynomial time
(max-flow/min-cut algorithms)



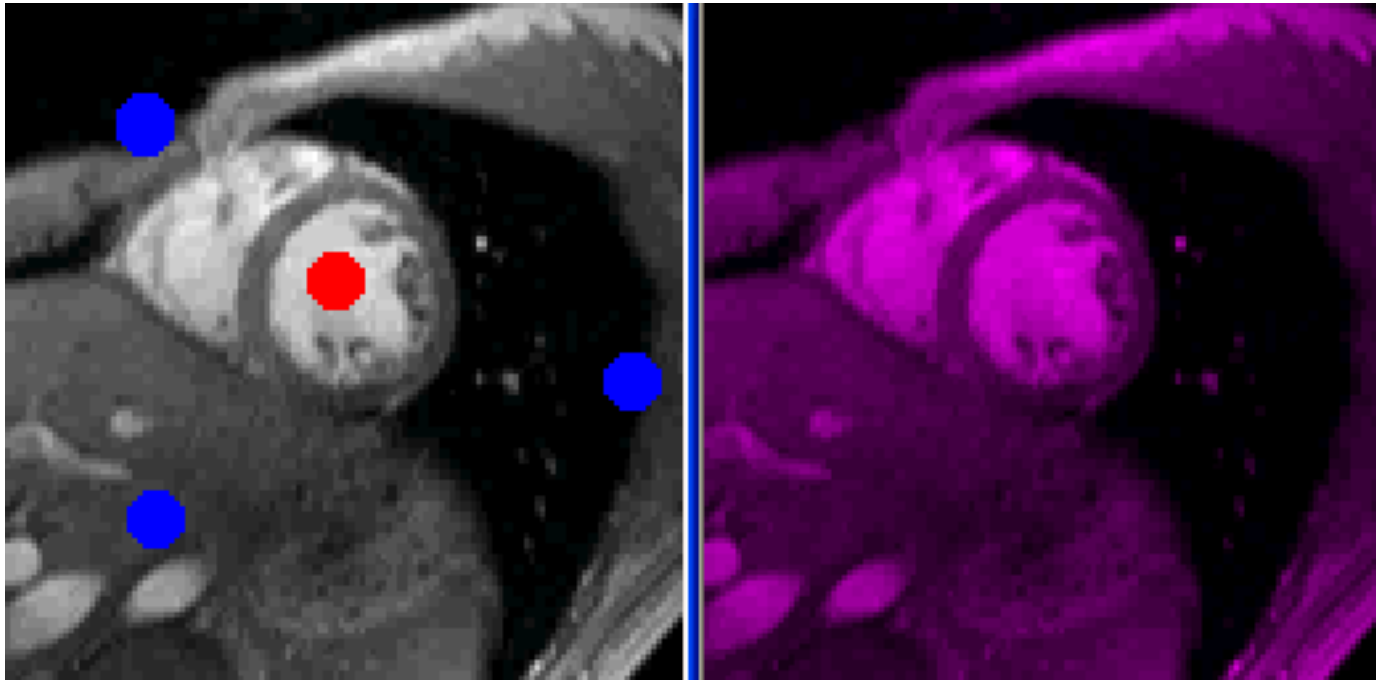
Standard minimum s - t cuts algorithms

- Augmenting paths [Ford & Fulkerson, 1962]
- Push-relabel [Goldberg-Tarjan, 1986]

adapted to N-D grids used in computer vision

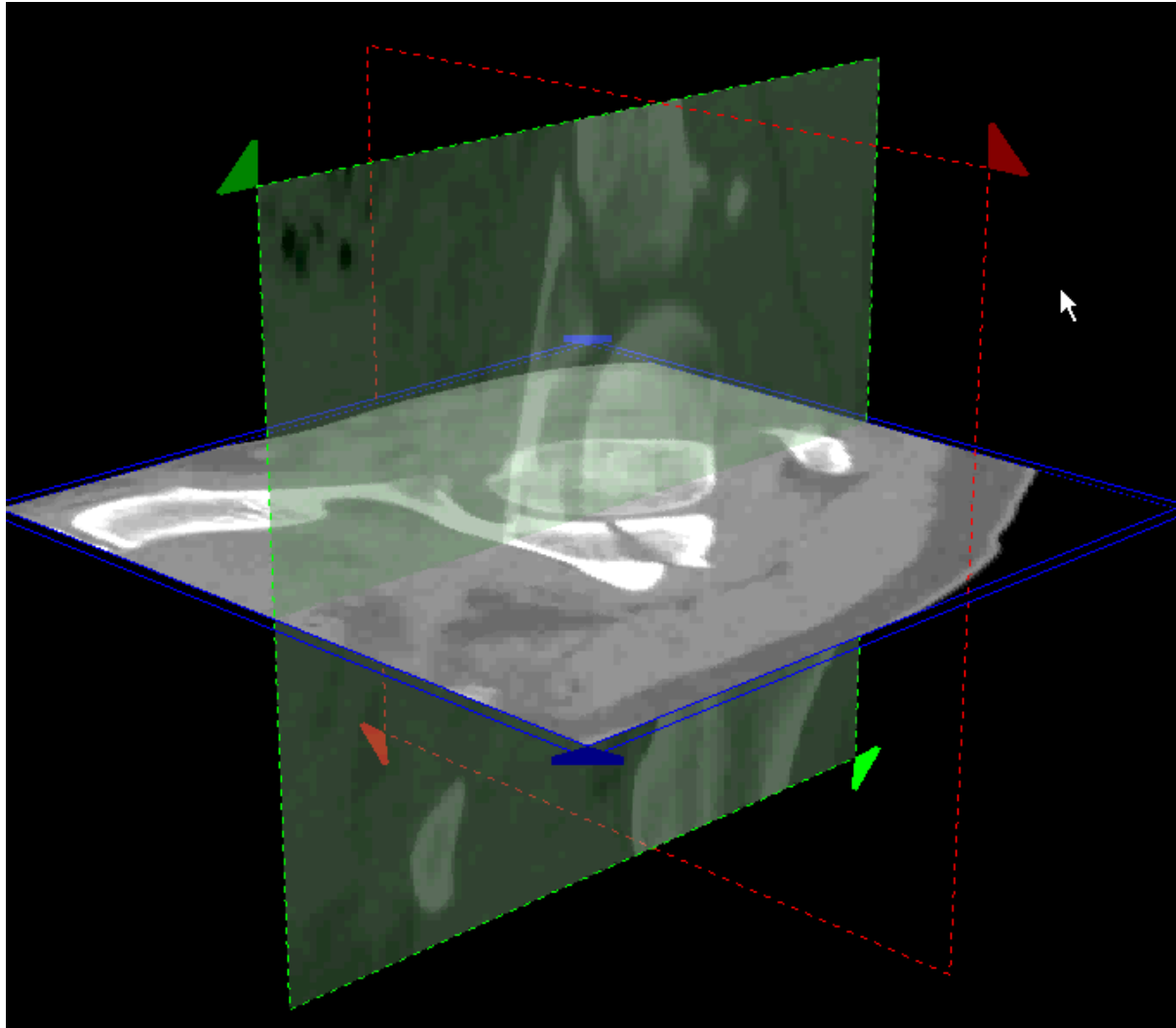
- Tree recycling (dynamic trees) [B&K, 2004]
- Flow recycling (*dynamic cuts*) [Kohli & Torr, 2005]
- Cut recycling (*active cuts*) [Juan & Boykov, 2006]
- Hierarchical methods
 - in search space [Lombaert et al., CVPR 2005]
 - in edge weights (*capacity scaling*) [Juan et al., ICCV07]

Optimal boundary in 2D



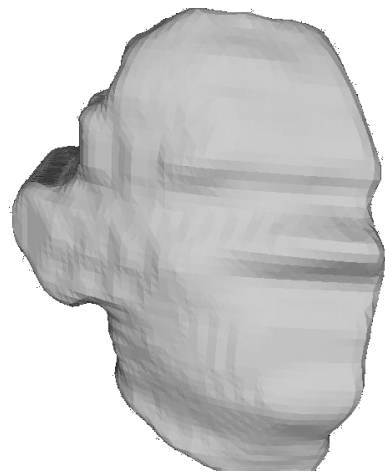
“max-flow = min-cut”

Optimal boundary in 3D

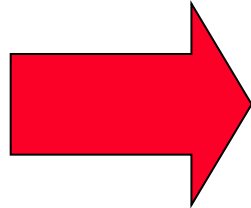


3D bone segmentation (real time screen capture)

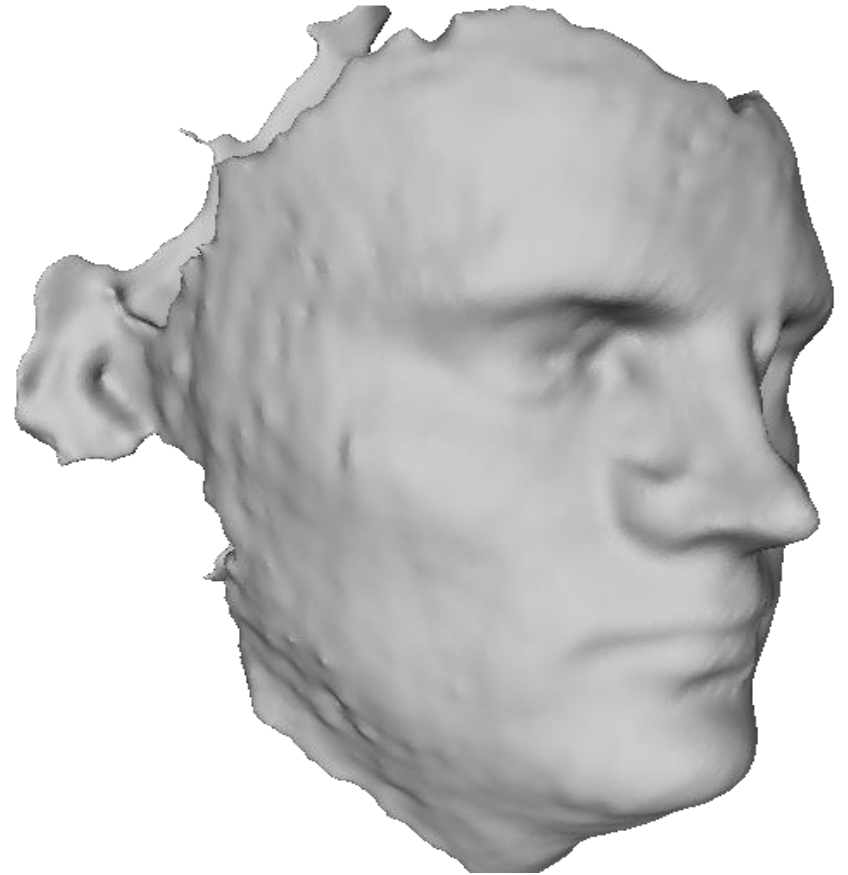
Graph cuts applied to multi-view reconstruction



visual hull
(silhouettes)

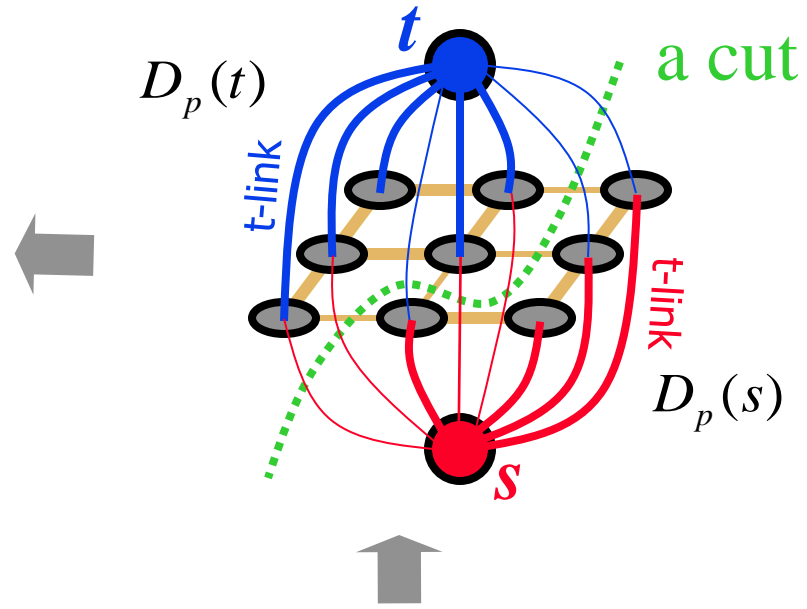
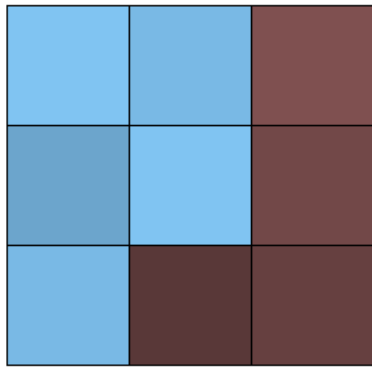


surface of good photoconsistency



CVPR'05 slides from Vogiatzis, Torr, Cippola

Adding regional properties (B&J, ICCV'01)



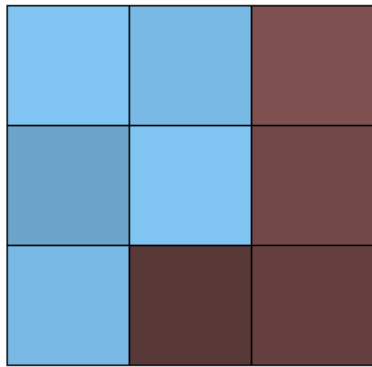
regional bias example

suppose I^s and I^t are given
"expected" intensities
of **object** and **background**

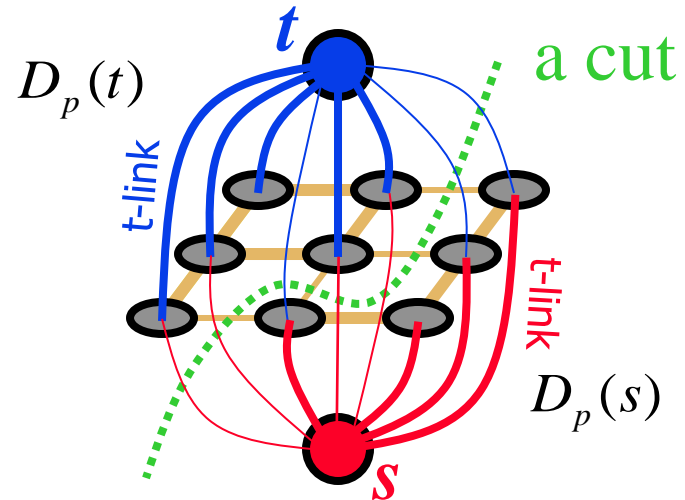
$$D_p(s) \propto \exp\left(-\|I_p - I^s\|^2 / 2\sigma^2\right)$$
$$D_p(t) \propto \exp\left(-\|I_p - I^t\|^2 / 2\sigma^2\right)$$

NOTE: hard constrains are not required, in general.

Adding regional properties (B&J, ICCV'01)



“expected” intensities of
object and **background**
 I^s and I^t
can be re-estimated



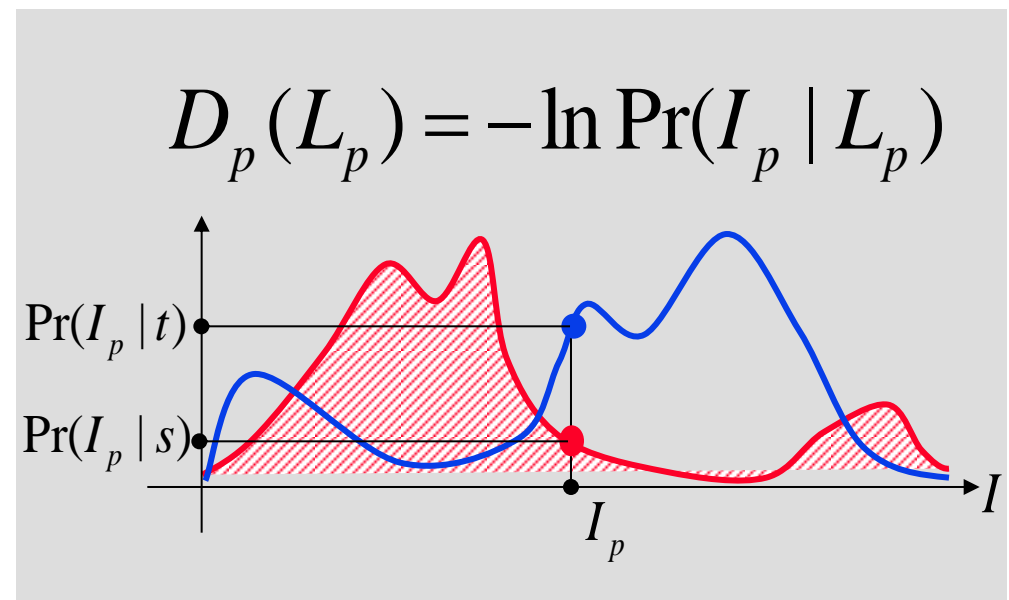
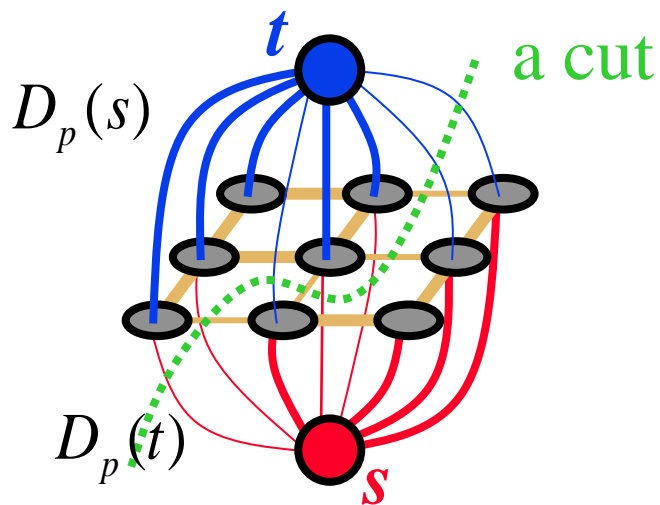
$$D_p(s) \propto \exp\left(-\|I_p - I^s\|^2 / 2\sigma^2\right)$$
$$D_p(t) \propto \exp\left(-\|I_p - I^t\|^2 / 2\sigma^2\right)$$

EM-style optimization of piece-wise constant *Mumford-Shah* model

Adding regional properties

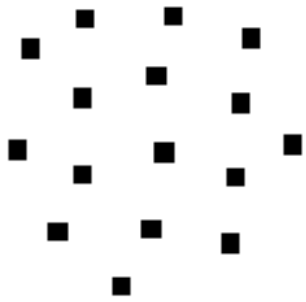
(B&J, ICCV'01)

More generally, regional bias can be based on any intensity models of object and background

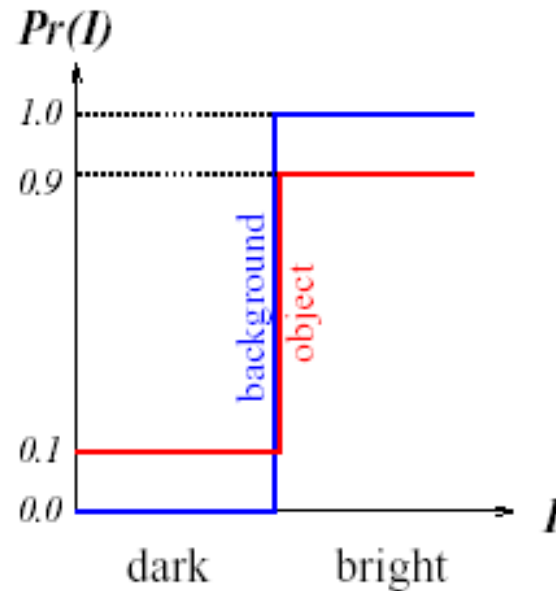


given object and background intensity histograms

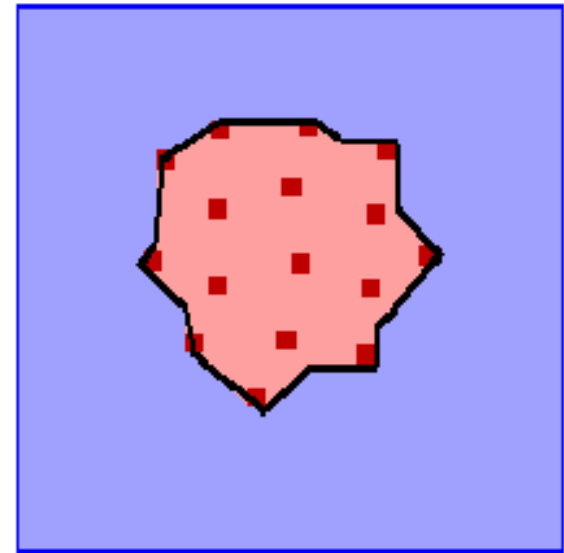
Adding regional properties (B&J, ICCV'01)



(a) Original image



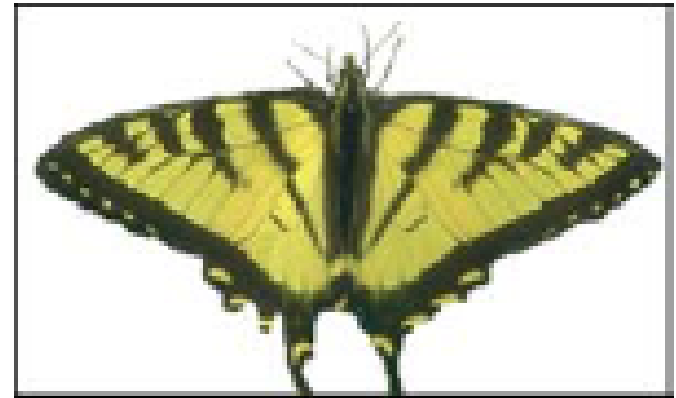
(b) Intensity histograms



(c) Optimal segmentation

Iterative learning of regional color-models

- GMMRF cuts (Blake et al., ECCV04)
- Grab-cut (Rother et al., SIGGRAPH 04)



parametric regional model – Gaussian Mixture (GM)
designed to guarantee convergence

At least three ways to look at energy of graph cuts

I: Binary submodular energy

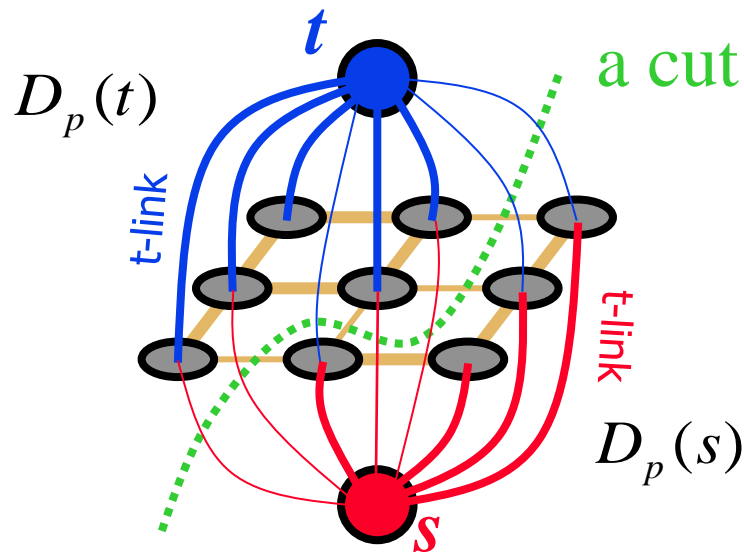
II: Approximating continuous surface functionals

III: Posterior energy (MAP-MRF)

Simple example of energy

$$E(L) = \sum_p \underbrace{D_p(L_p)}_{\text{t-links}} + \sum_{pq \in N} \underbrace{w_{pq} \cdot \delta(L_p \neq L_q)}_{\text{n-links}}$$

Regional term Boundary term



$$L_p \in \{s, t\}$$

**binary object
segmentation**

Graph cuts for minimization of submodular binary energies I

$$E(L) = \sum_p E_p(L_p) + \sum_{pq \in N} E(L_p, L_q)$$

t-links **n-links** $L_p \in \{s, t\}$

■ Characterization of **binary** energies that can be globally minimized by s - t graph cuts [Boros&Hummer, 2002, K&Z 2004]

$E(L)$ can be minimized by s - t graph cuts



$$E(s, s) + E(t, t) \leq E(s, t) + E(t, s)$$

Submodularity ("convexity")

■ **Non-submodular cases** can be addressed with some optimality guarantees, e.g. **QPBO** algorithm

- (see Boros&Hummer, 2002, Tavares et al. 06, Rother et al. 07)

Graph cuts for minimization of continuous surface functionals

II

$$E(C) = \int_C g(\cdot) ds + \int_C \langle \vec{\mathbf{N}}, \vec{\mathbf{v}}_x \rangle ds + \int_{\Omega(C)} f(x) dp$$

Geometric length

any convex,
symmetric metric \mathbf{g}
e.g. Riemannian

Flux

any vector field \mathbf{v}

Regional bias

any scalar function f

- Characterization of energies of **binary** cuts C as functionals of continuous surfaces

[B&K, ICCV 2003]

[K&B, ICCV 2005]

One extension using parametric max-flow methods

- optimization of ratio functionals

$$E(C) = \frac{\int \langle \vec{\mathbf{N}}, \vec{\mathbf{v}}_x \rangle ds}{\int_C g(\cdot) ds}$$

$$E(C) = \frac{\int g(\cdot) ds}{\int_{\Omega(C)} f(x) dp}$$

- In 2D can use DP [Cox et al'96, Jermyn&Ishikawa'01]
- In 3D, [Kolmogorov, Boykov, Rother, ICCV'07]

Graph cuts for minimization of posterior energy

III

■ Greig et al. [IJRSS, 1989]

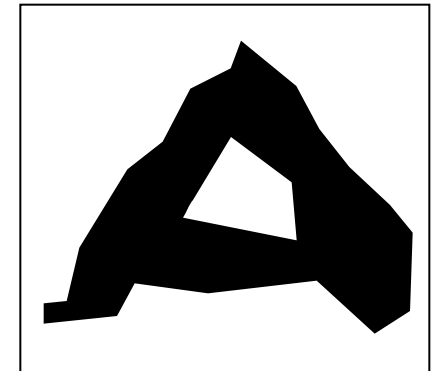
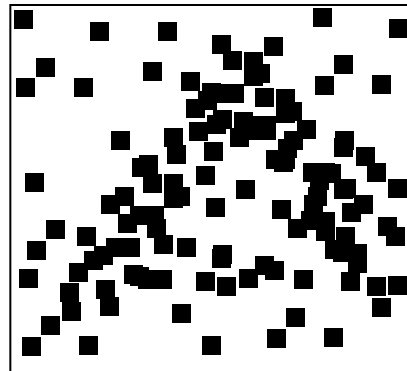
- Posterior energy (MRF, Ising model)

$$E(L) = \sum_p -\ln \Pr(D_p | L_p) + \sum_{pq \in N} V_{pq}(L_p, L_q)$$

**Likelihood
(data term)**

**Spatial prior
(regularization)**

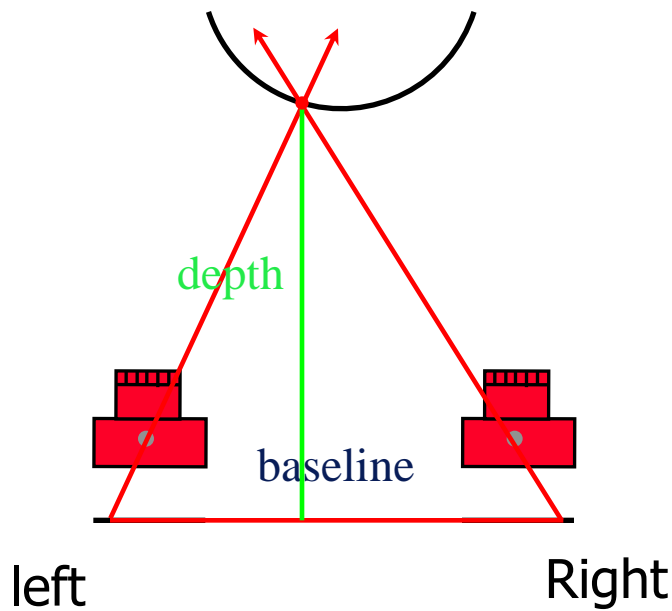
$L_p \in \{s, t\}$



Example: binary image restoration

Graph cuts algorithms can minimize
multi-label energies as well

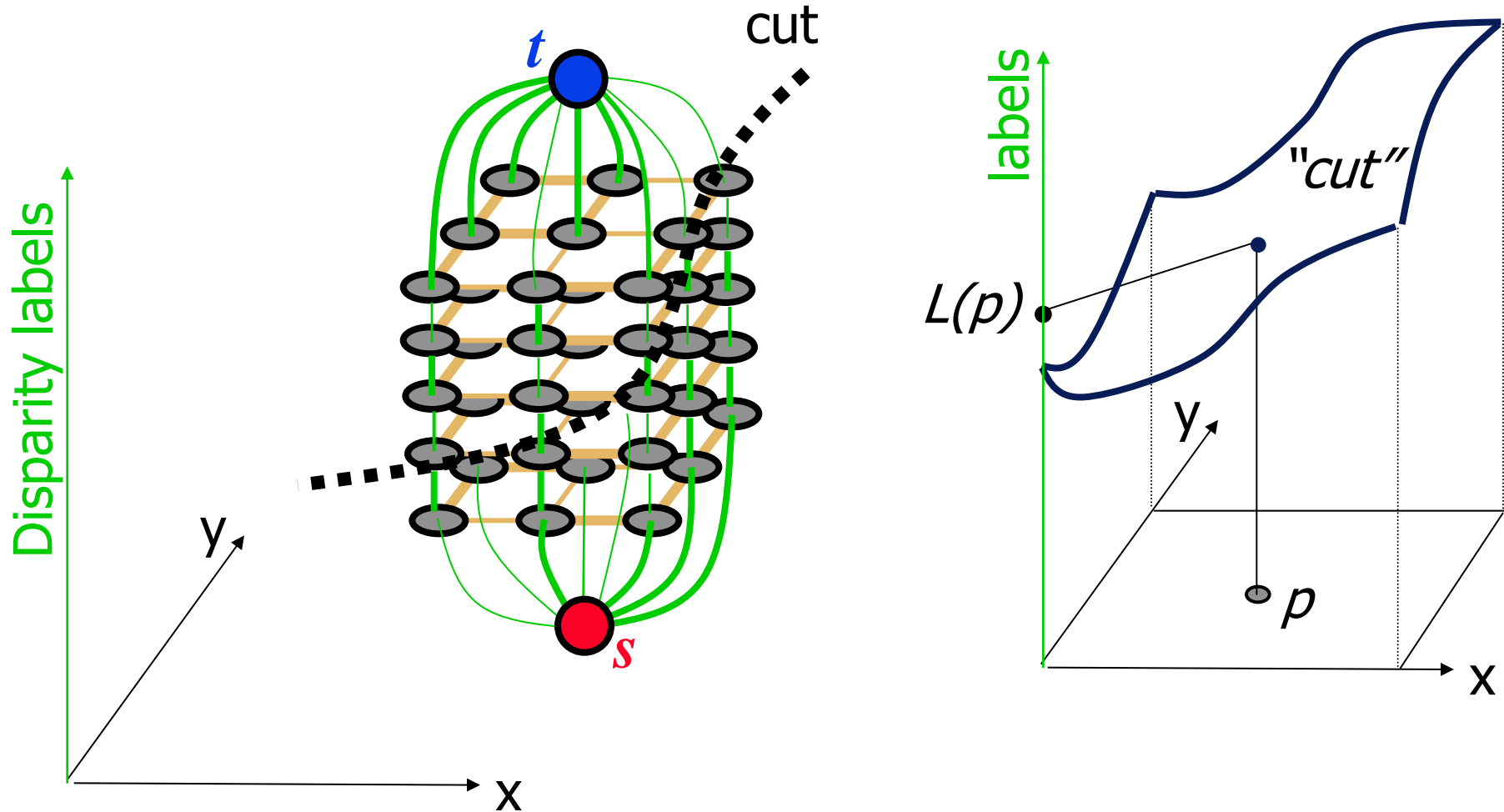
Problem Definition: Stereo Matching



Triangulate on two images of the same scene point to recover depth.

- Camera calibration
- Finding all correspondences
- Computing depth or surfaces

Multi-scan-line stereo with s - t graph cuts (Roy&Cox'98)



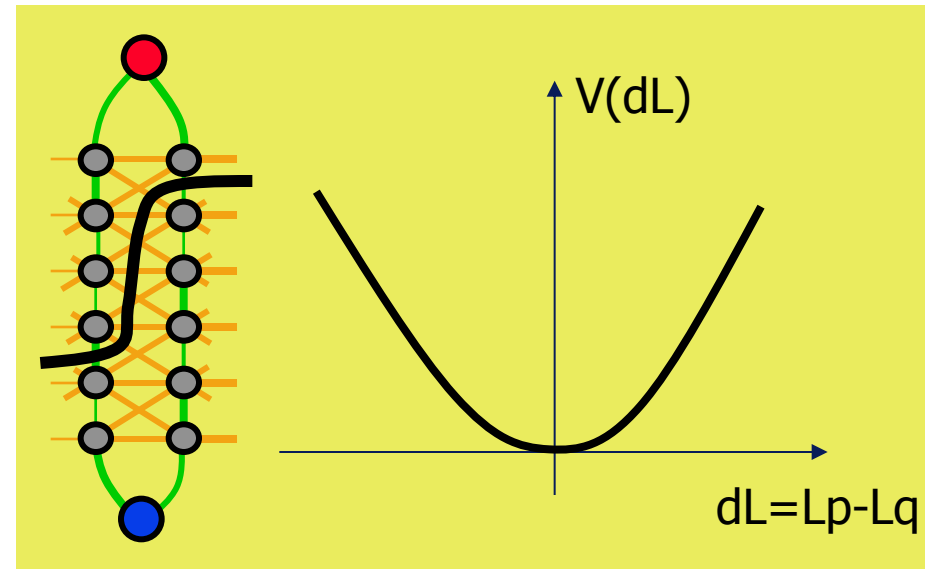
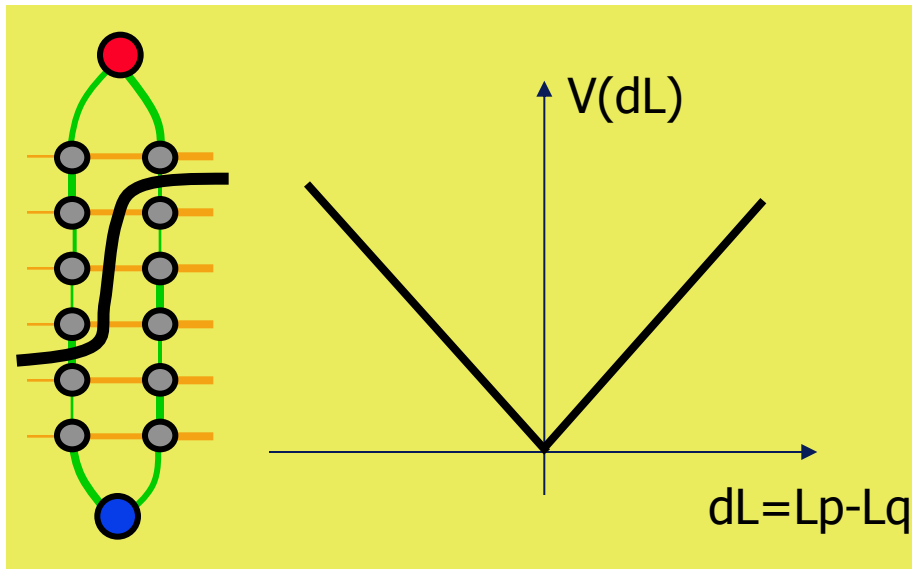
s - t graph-cuts for multi-label energy minimization

- Ishikawa 1998, 2000, 2003
- Generalization of construction by Roy&Cox 1998

$$E(L) = \sum_p D_p(L_p) + \sum_{pq \in N} V(L_p, L_q) \quad L_p \in R^1$$

Linear interactions

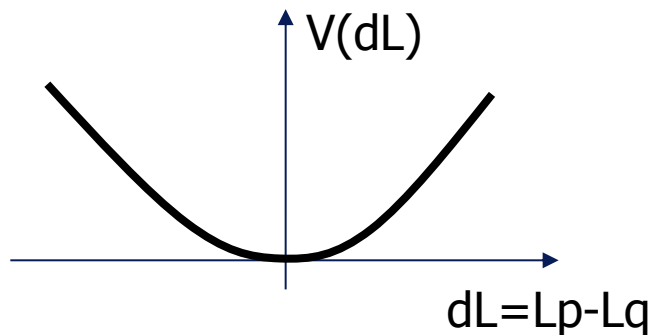
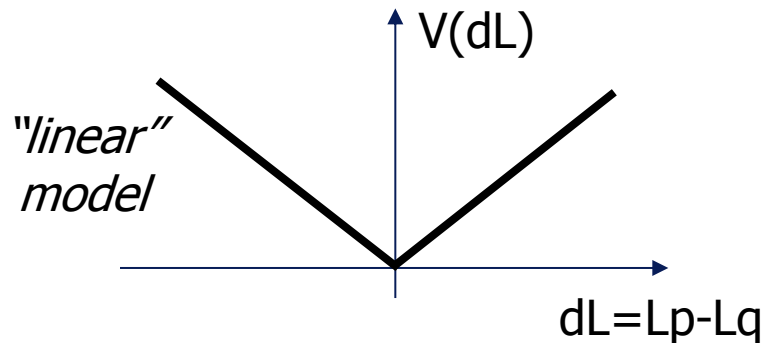
"Convex" interactions



Pixel interactions V :

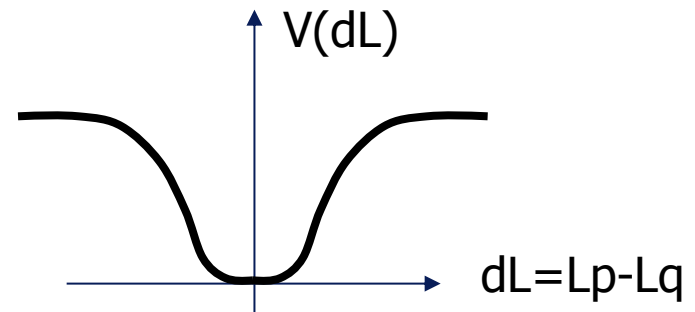
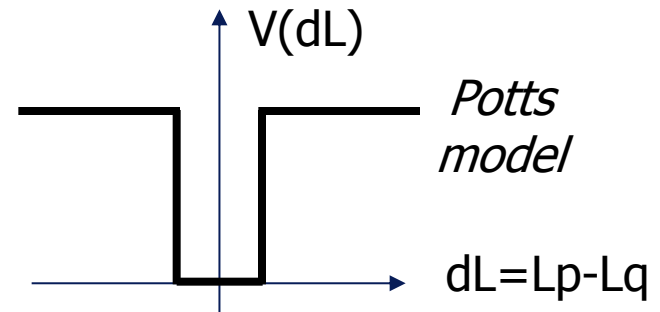
“convex” vs. “discontinuity-preserving”

“Convex”
Interactions V



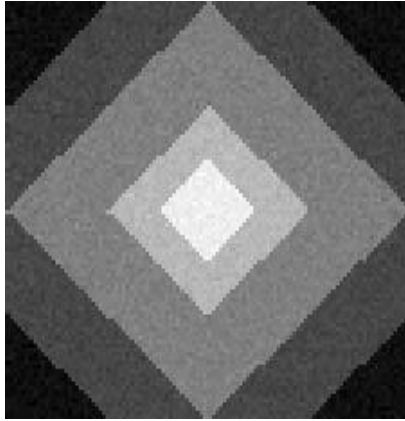
Robust or “discontinuity preserving”
Interactions V

(weak membrane models,
see a book by [Blake and Zisserman, 87](#))

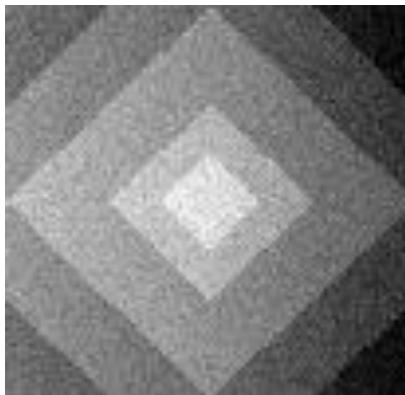
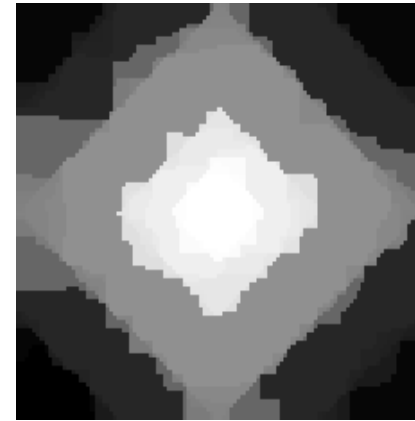
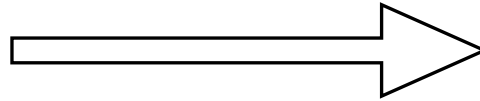


Pixel interactions:

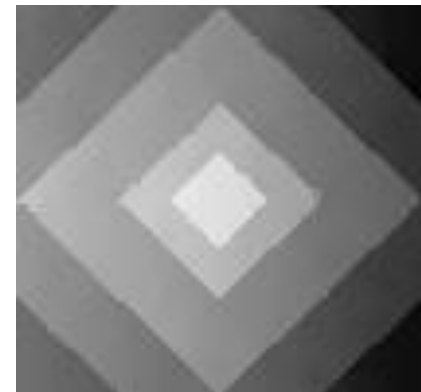
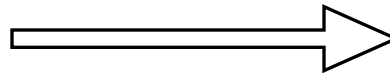
“convex” vs. *“discontinuity-preserving”*



“linear” V



truncated
“linear” V



Fast Approximate Energy Minimization

$$E(f) = \sum_{\{p,q\} \in \mathcal{N}} V_{p,q}(f_p, f_q) + \sum_{p \in \mathcal{P}} D_p(f_p)$$

Simple Way to Optimize - ICM

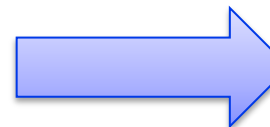
1. Choose a pixel
2. Fix the disparities at the rest of the pixels
3. Set the pixel to the optimal disparity
4. Iterate

- Advantages

- Energy always guaranteed to decrease
- Easy to code

- Disadvantages

- Convergence
- Scalability



Make Bigger Moves!

Fast Approx. Energy Minimization Cont.

- The problem with ICM is that you can only modify one pixel at a time
- Get stuck in local minima easily
- We need a way of moving multiple pixels at once
- Boykov, Veksler and Zabih introduced two types of moves:
 - alpha-beta swap
 - alpha-expansion

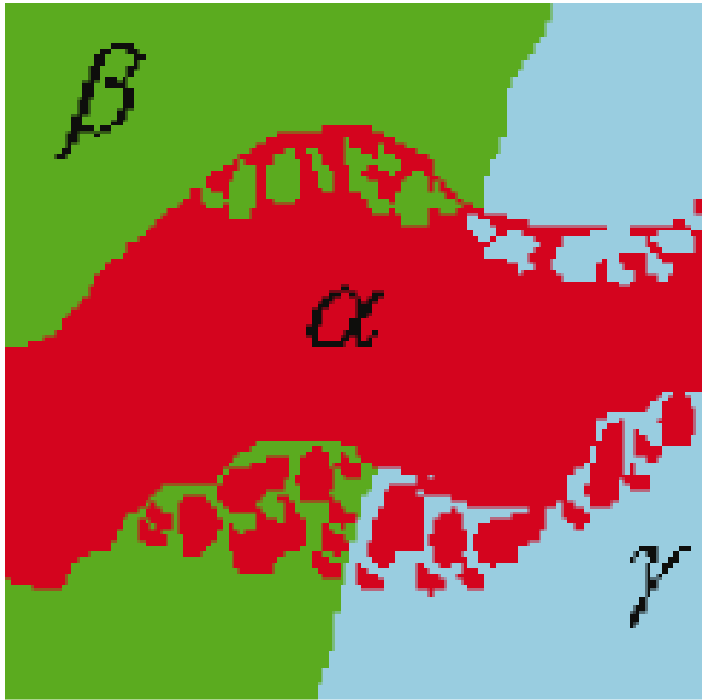
Moving Policies



- ICM – One pixel moves

(From BKZ-PAMI 01)

Moving Policies



(From BKZ-PAMI 01)

- Fix all nodes that aren't labelled alpha or beta
- With remaining nodes, find optimal swap
 - Some alpha nodes change to beta
 - Vice Versa
 - Some stay the same

alpha-beta swap

Moving Policies



- Any node can change to alpha

(From BKZ-PAMI 01)

alpha-expansion

a-expansion & a-b swap overview

1. Start with an arbitrary labeling f
2. Set $\text{success} := 0$
3. For each pair of labels $\{\alpha, \beta\} \subset \mathcal{L}$
 - 3.1. Find $\hat{f} = \operatorname{argmin} E(f')$ among f' within one α - β swap of f
 - 3.2. If $E(\hat{f}) < E(f)$, set $f := \hat{f}$ and $\text{success} := 1$
4. If $\text{success} = 1$ goto 2
5. Return f

-
1. Start with an arbitrary labeling f
 2. Set $\text{success} := 0$
 3. For each label $\alpha \in \mathcal{L}$
 - 3.1. Find $\hat{f} = \operatorname{argmin} E(f')$ among f' within one α -expansion of f
 - 3.2. If $E(\hat{f}) < E(f)$, set $f := \hat{f}$ and $\text{success} := 1$
 4. If $\text{success} = 1$ goto 2
 5. Return f

alpha-beta graphs

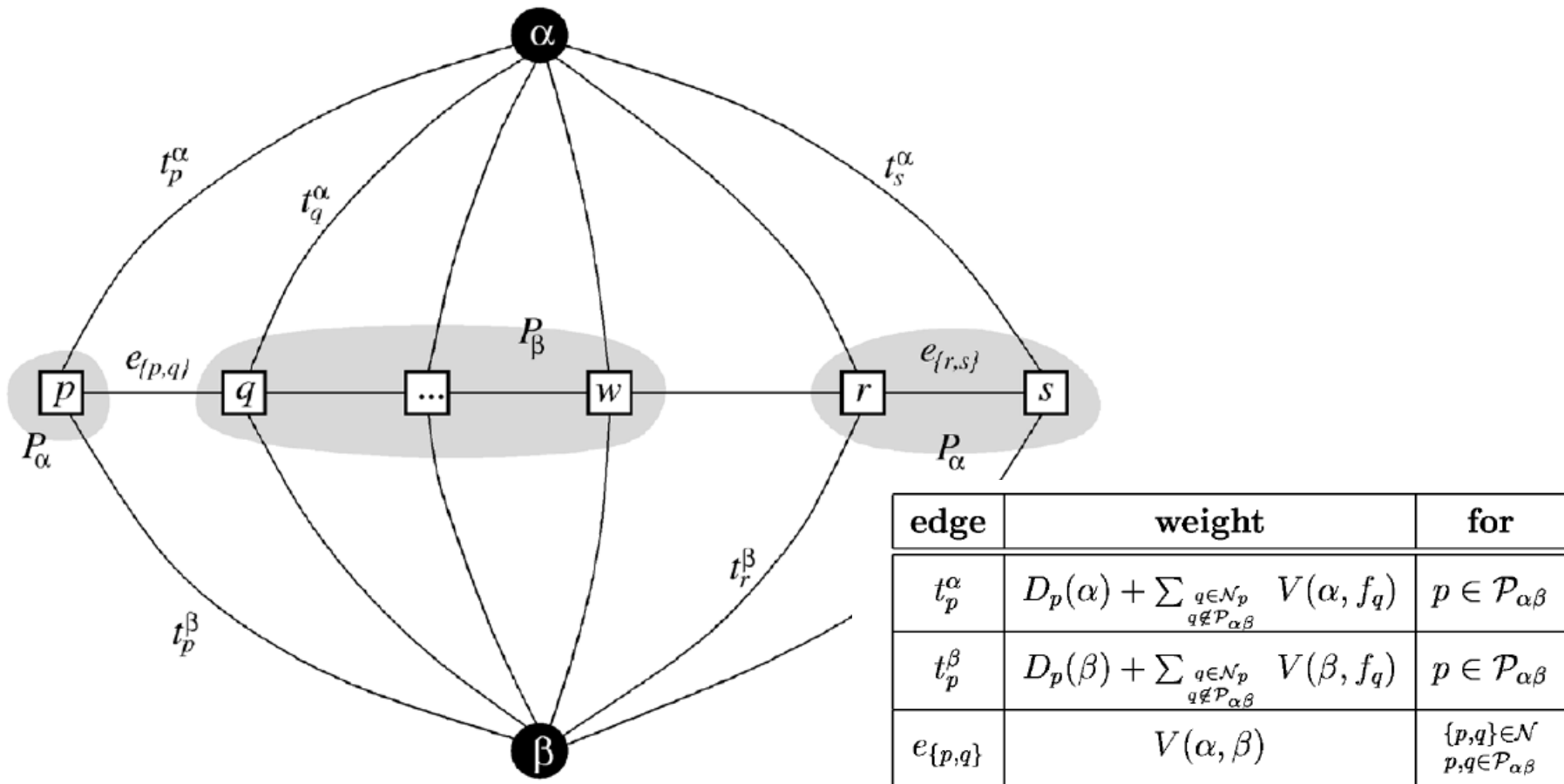
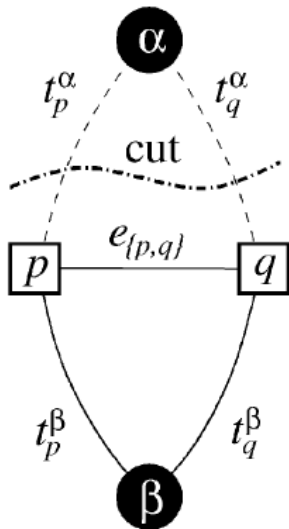


Fig. 4. An example of the graph $\mathcal{G}_{\alpha\beta}$ for a 1D image. The set of pixels in the image is $\mathcal{P}_{\alpha\beta} = \mathcal{P}_\alpha \cup \mathcal{P}_\beta$, where $\mathcal{P}_\alpha = \{p, r, s\}$ and $\mathcal{P}_\beta = \{q, \dots, w\}$.

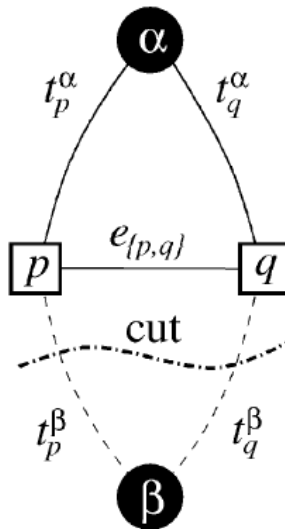
alpha-beta swap properties

Property 4.2. For any cut C and for any n -link $e_{\{p,q\}}$:

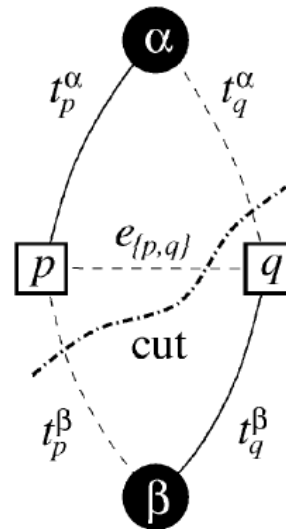
- (a) If $t_p^\alpha, t_q^\alpha \in C$ then $e_{\{p,q\}} \notin C$.
- (b) If $t_p^\beta, t_q^\beta \in C$ then $e_{\{p,q\}} \notin C$.
- (c) If $t_p^\beta, t_q^\alpha \in C$ then $e_{\{p,q\}} \in C$.
- (d) If $t_p^\alpha, t_q^\beta \in C$ then $e_{\{p,q\}} \in C$.



Property 4.2(a)



Property 4.2(b)



Property 4.2(c,d)

alpha-beta swap: proof

Lemma 4.1. *A labeling $f^{\mathcal{C}}$ corresponding to a cut \mathcal{C} on $\mathcal{G}_{\alpha\beta}$ is one α - β swap away from the initial labeling f .*

Lemma 4.3. *For any cut \mathcal{C} and for any n -link $e_{\{p,q\}}$*

$$|\mathcal{C} \cap e_{\{p,q\}}| = V(f_p^{\mathcal{C}}, f_q^{\mathcal{C}}).$$

Theorem 4.4. *There is a one to one correspondence between cuts \mathcal{C} on $\mathcal{G}_{\alpha\beta}$ and labelings that are one α - β swap from f . Moreover, the cost of a cut \mathcal{C} on $\mathcal{G}_{\alpha\beta}$ is $|\mathcal{C}| = E(f^{\mathcal{C}})$ plus a constant.*

Corollary 4.5. *The lowest energy labeling within a single α - β swap move from f is $\hat{f} = f^{\mathcal{C}}$, where \mathcal{C} is the minimum cut on $\mathcal{G}_{\alpha\beta}$.*

a-expansion

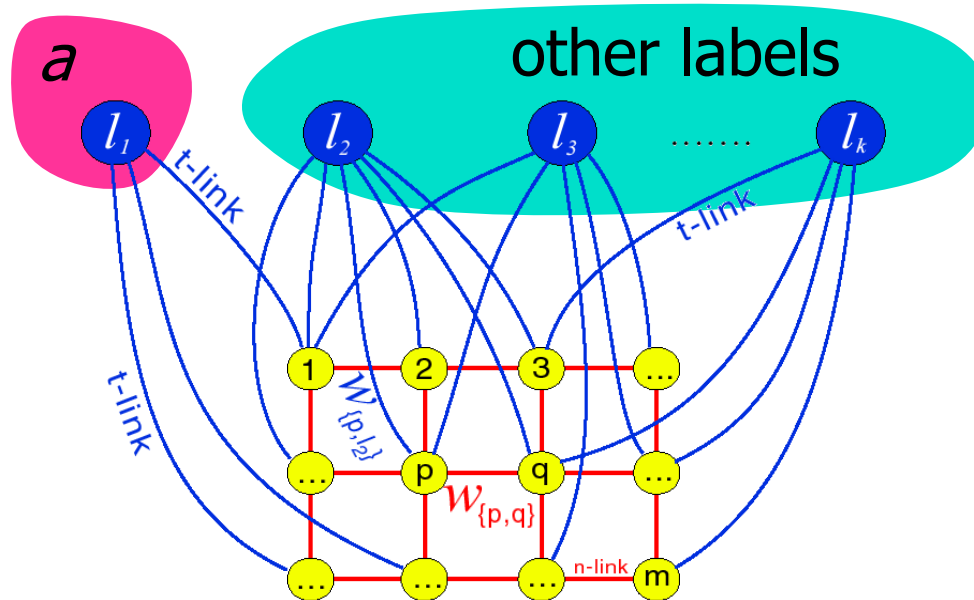
- NP-hard problem (3 or more labels)
 - two labels can be solved via $s-t$ cuts (Greig et al., 1989)
- *a-expansion* approximation algorithm
(Boykov, Veksler, Zabih 1998, 2001)
 - guaranteed approximation quality (Veksler, thesis 2001)
 - within a factor of 2 from the global minima (Potts model)
 - applies to a wide class of energies with robust interactions
 - Potts model (BVZ 1989)
 - “metric” interactions (BVZ 2001)
 - can be extended to arbitrary interactions with weaker guarantees
 - truncation (Kolmogorov et al. 2005)
 - QPBO (Boros and Hummer, 2002)
- Other “move” algorithms (e.g. $a-b$ swap, jump-moves)

a-expansion algorithm

1. Start with any initial solution
2. For each label "*a*" in any (e.g. random) order
 1. *Compute optimal a-expansion move (s-t graph cuts)*
 2. *Decline the move if there is no energy decrease*
3. *Stop when no expansion move would decrease energy*

a-expansion move

Basic idea: break multi-way cut computation into a **sequence of binary *s-t* cuts**



a -expansion moves

In each a -expansion a given label " a " grabs space from other labels



initial solution

● -expansion

● -expansion

● -expansion

● -expansion

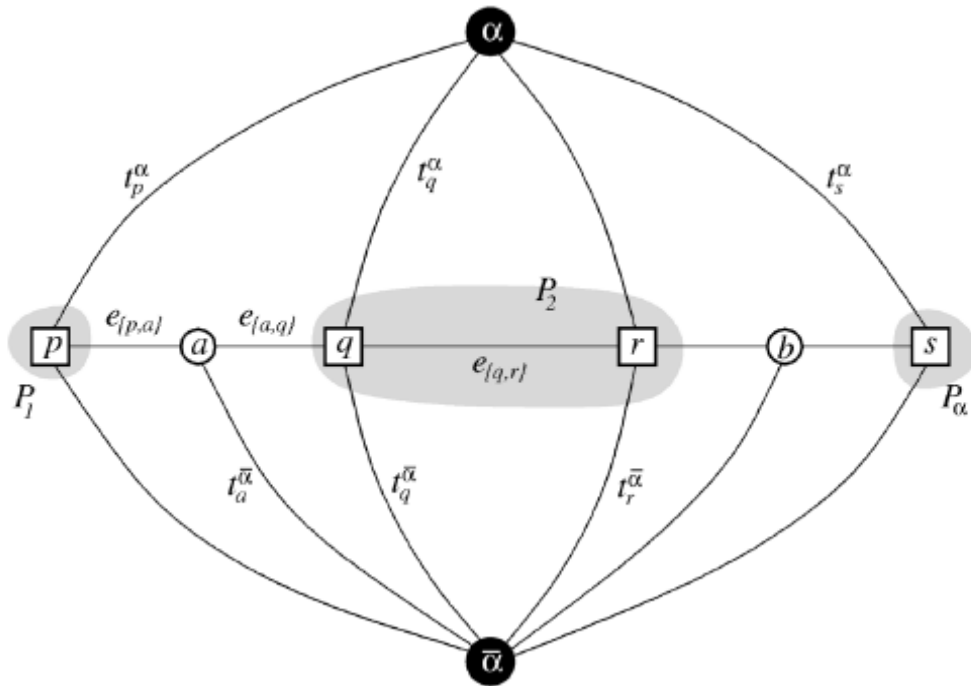
● -expansion

● -expansion

● -expansion

For each move we choose expansion that gives the largest decrease in the energy: **binary optimization problem**

Expansion move



The weights assigned to the edges are

edge	weight	for
$t_p^{\bar{\alpha}}$	∞	$p \in \mathcal{P}_\alpha$
t_p^{α}	$D_p(f_p)$	$p \notin \mathcal{P}_\alpha$
t_p^{α}	$D_p(\alpha)$	$p \in \mathcal{P}$
$e_{\{p,a\}}$	$V(f_p, \alpha)$	$\{p, q\} \in \mathcal{N}, f_p \neq f_q$
$e_{\{a,q\}}$	$V(\alpha, f_q)$	
$t_a^{\bar{\alpha}}$	$V(f_p, f_q)$	
$e_{\{p,q\}}$	$V(f_p, \alpha)$	$\{p, q\} \in \mathcal{N}, f_p = f_q$

As in Section 4, any cut \mathcal{C} on \mathcal{G}_α must sever

- The difference between the optimal solution and the solution from the expansion move is bounded

Metric interactions

$$V(a,b)=0 \text{ iff } a=b$$

$$V(a,b) = V(b,a) \geq 0$$

$$V(a,c) \leq V(a,b)+V(b,c)$$

Triangular
inequality

Implies that every expansion move (a binary problem)
is submodular

- V is called a metric if it obeys all 3
 - Can use Expansion Moves
- V is called a semi-metric if it obeys the last two
 - Can only use Swap-Moves
- Subject of recent research.

alpha expansion algorithm

Property 5.2. *If $\{p, q\} \in \mathcal{N}$ and $f_p \neq f_q$, then a minimum cut \mathcal{C} on \mathcal{G}_α satisfies:*

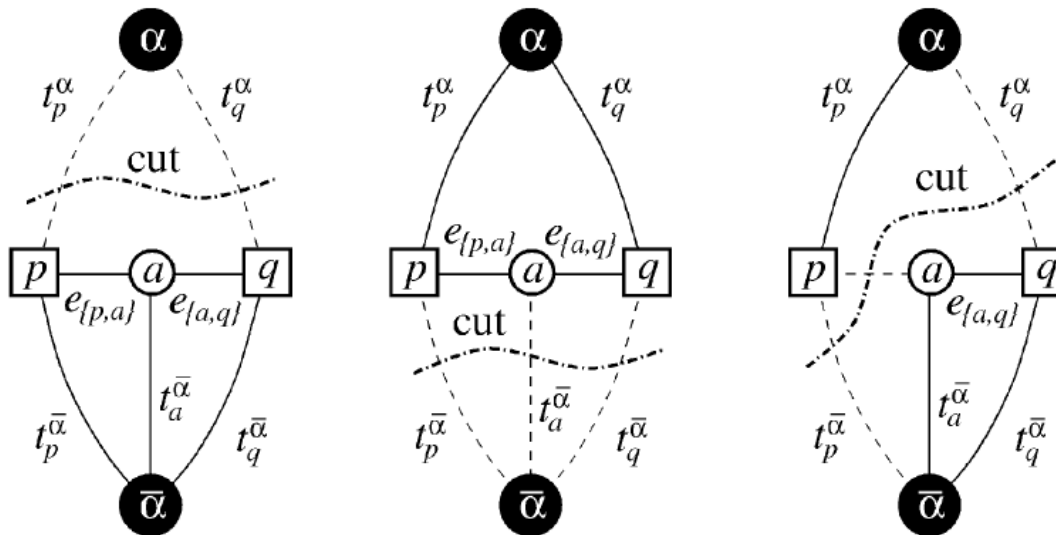
(a) *If $t_p^\alpha, t_q^\alpha \in \mathcal{C}$ then $\mathcal{C} \cap \mathcal{E}_{\{p,q\}} = \emptyset$.*

(b) *If $t_p^{\bar{\alpha}}, t_q^{\bar{\alpha}} \in \mathcal{C}$ then $\mathcal{C} \cap \mathcal{E}_{\{p,q\}} = t_a^{\bar{\alpha}}$.*

(c) *If $t_p^{\bar{\alpha}}, t_q^\alpha \in \mathcal{C}$ then $\mathcal{C} \cap \mathcal{E}_{\{p,q\}} = e_{\{p,a\}}$.*

(d) *If $t_p^\alpha, t_q^{\bar{\alpha}} \in \mathcal{C}$ then $\mathcal{C} \cap \mathcal{E}_{\{p,q\}} = e_{\{a,q\}}$.*

alpha expansion algorithm

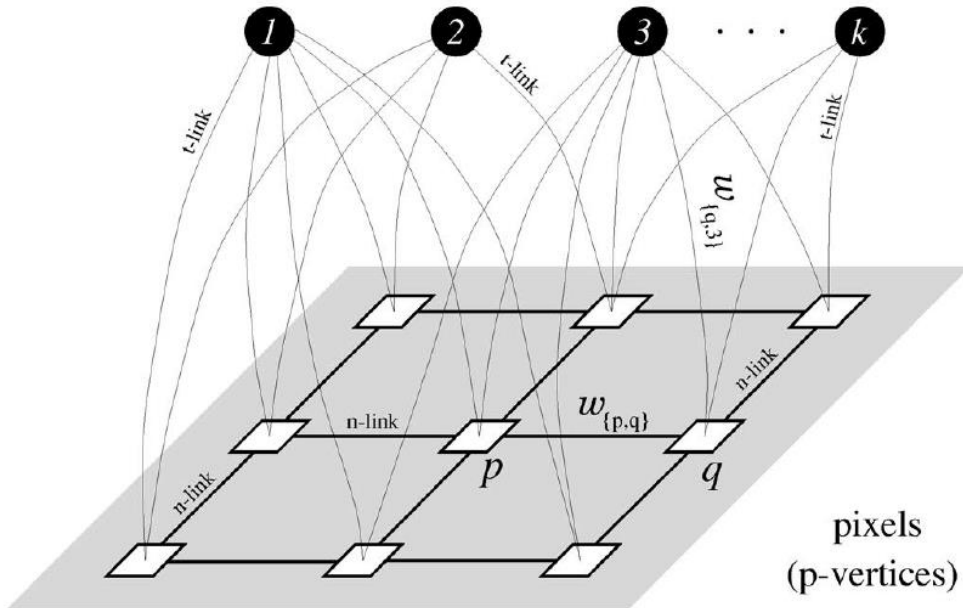


Property 5.2(a) Property 5.2(b) Property 5.2(c,d)

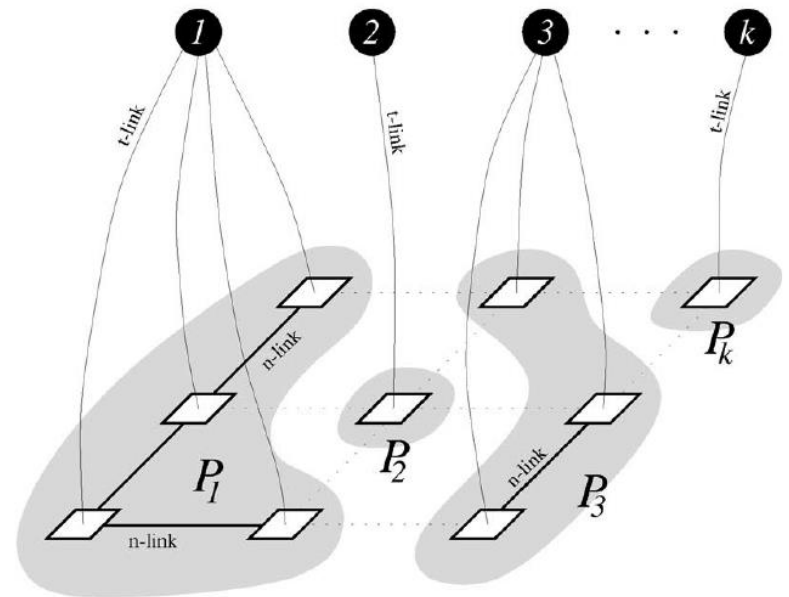
Fig. 7. Properties of a minimum cut \mathcal{C} on \mathcal{G}_α for two pixel $p, q \in \mathcal{N}$ such that $f_p \neq f_q$. Dotted lines show the edges cut by \mathcal{C} and solid lines show the edges in the induced graph $\mathcal{G}(\mathcal{C}) = \langle \mathcal{V}, \mathcal{E} - \mathcal{C} \rangle$.

2-D alpha expansion

terminals (l-vertices or labels)

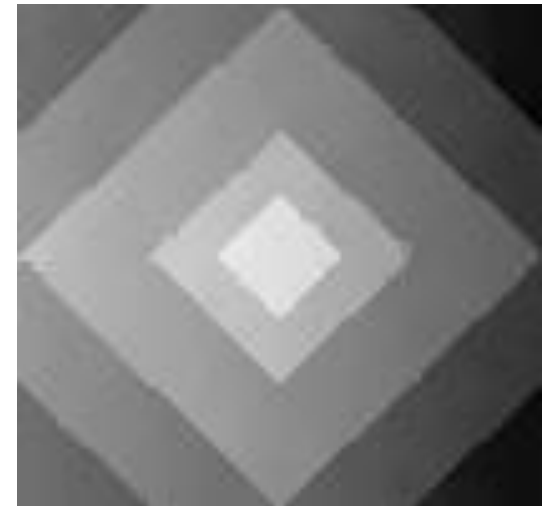
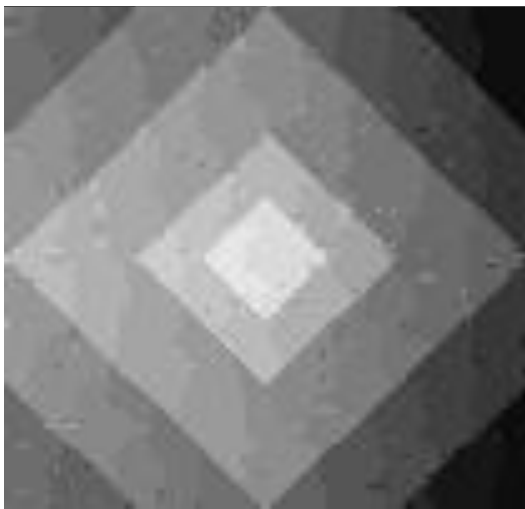
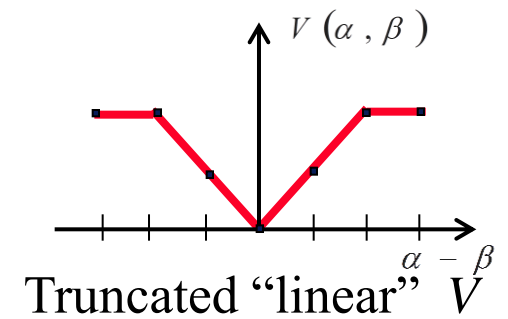
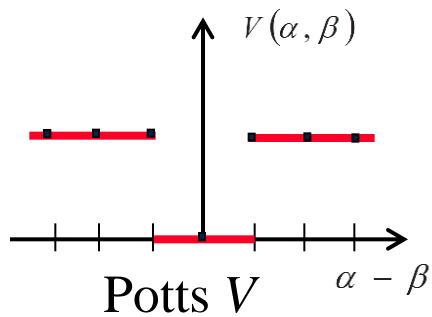


(a)



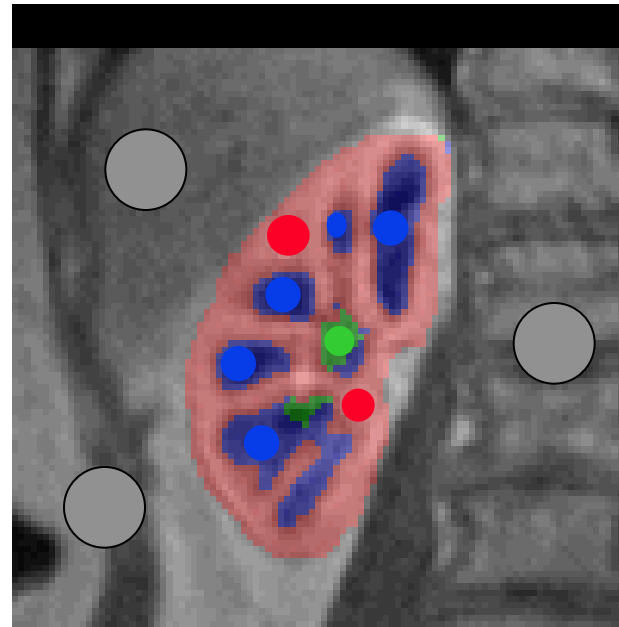
(b)

α -expansions: examples of *metric* interactions



Multi-way graph cuts

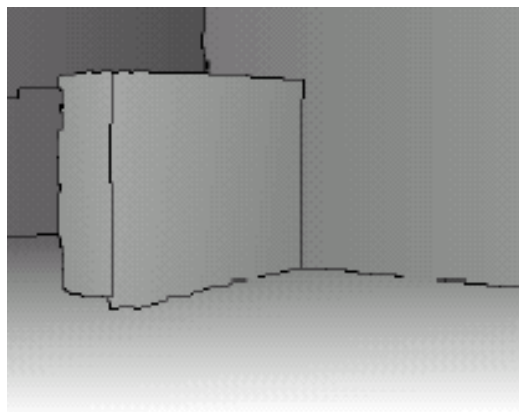
Multi-object Extraction



Multi-way graph cuts

Stereo/Motion with slanted surfaces

(Birchfield & Tomasi 1999)



Labels = parameterized surfaces

EM based: E step = compute surface boundaries

M step = re-estimate surface parameters

Multi-way graph cuts

stereo vision



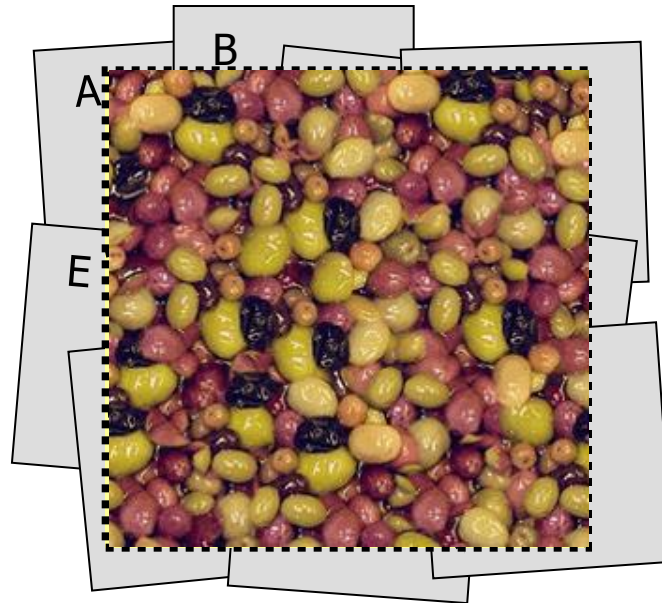
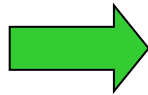
depth map

original pair of “stereo” images

Multi-way graph cuts

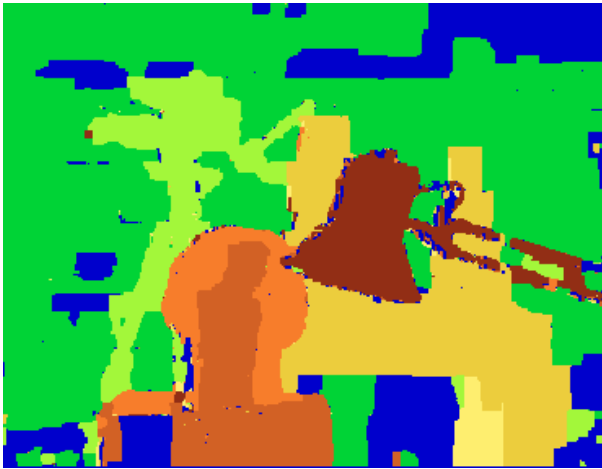
Graph-cut textures

(Kwatra, Schodl, Essa, Bobick 2003)



similar to “**image-quilting**” (Efros & Freeman, 2001)

a-expansions vs. simulated annealing



simulated annealing,
start for hours, 20.3% err
24.7% err



a-expansions (BVZ 89,01)
90 seconds, 5.8% err

