MRFs and CRFs for Vision: Models & Optimization

Carsten Rother
Microsoft Research Cambridge
Grenoble Summer School
July 2010

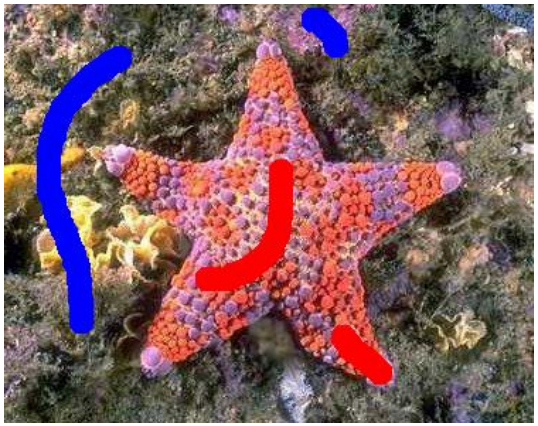
Outline

- Introduction
- MRFs and CRFs in Vision
- Optimisation techniques and Comparison

Outline

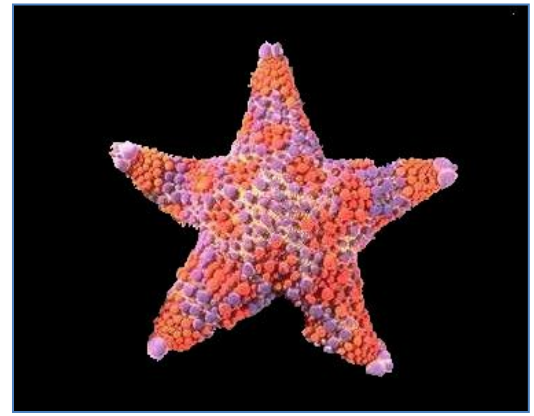
- Introduction
- MRFs and CRFs in Vision
- Optimisation techniques and Comparison

A gentle intro to Random Fields



$z = (R,G,B)^n$

Goal



$x = \{0,1\}^n$

Given $\mathbf{z}$ and unknown (latent) variables $\mathbf{x}$:

$$P(x|z) = P(z|x) \; P(x) \; / \; P(z) \sim P(z|x) \; P(x)$$

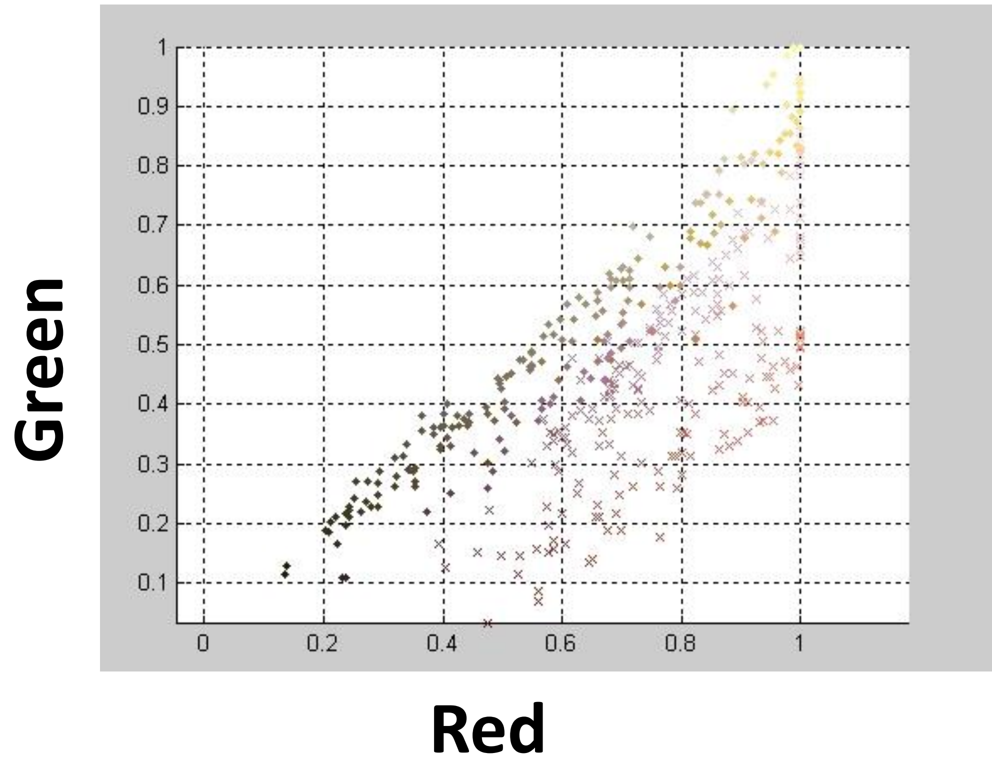
Posterior Probability

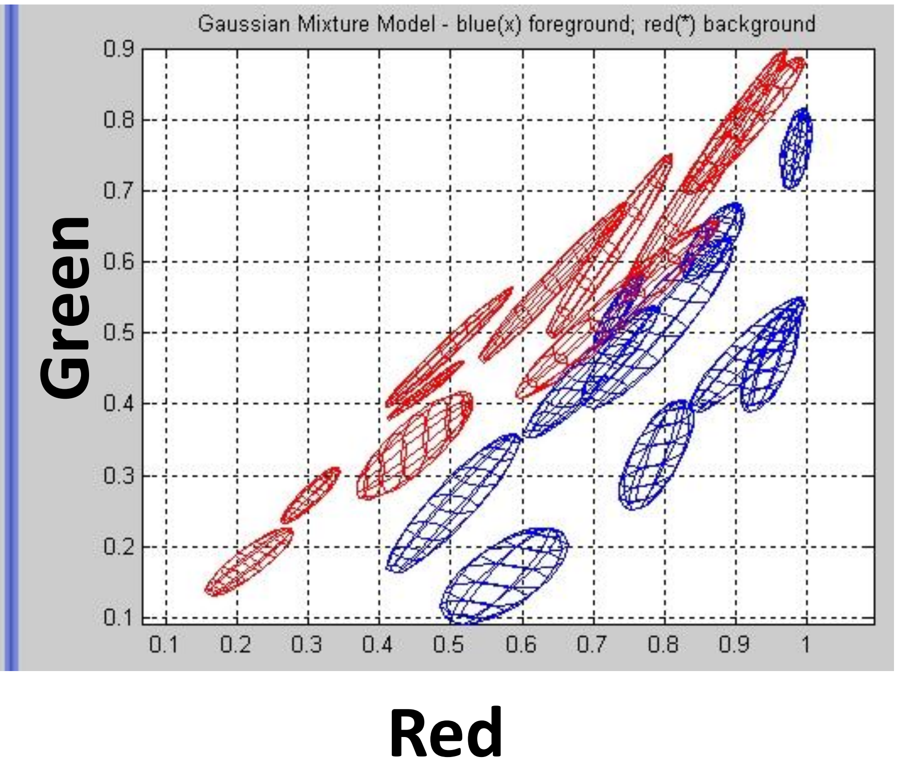
Likelihood (data-dependent)

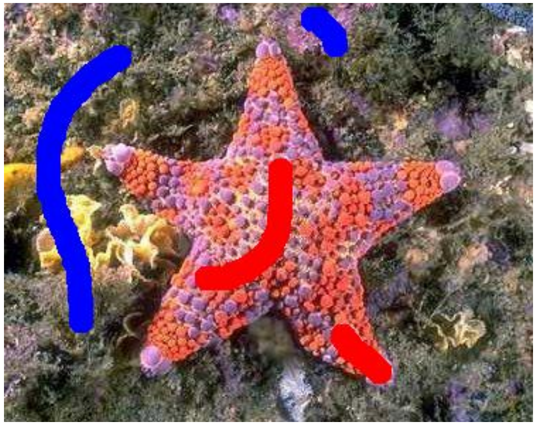
Prior (data-independent)

Maximium a Posteriori (MAP): $x^* = \underset{x}{\operatorname{argmax}} \; P(x|z)$

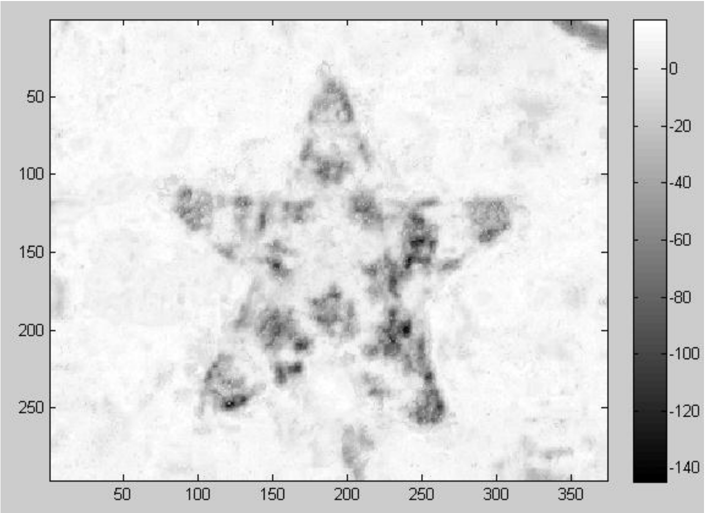
Likelihood $P(x|z) \sim P(z|x)\, P(x)$



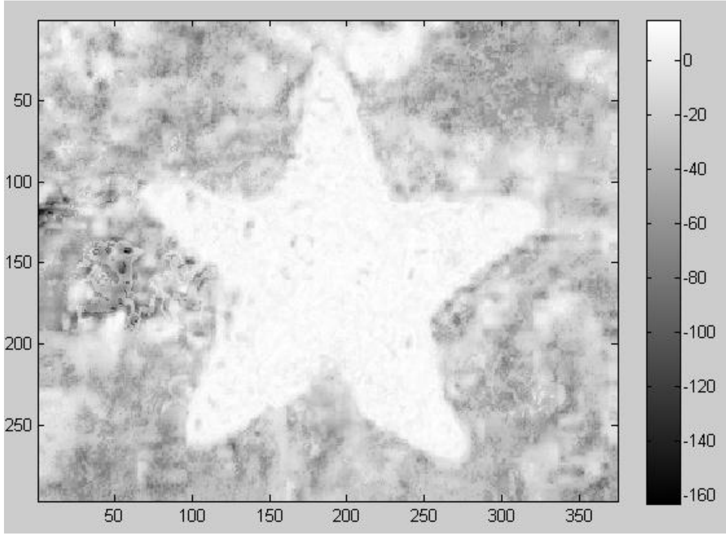




Likelihood $P(x|z) \sim P(z|x)\, P(x)$



$P(z_i|x_i=0)$

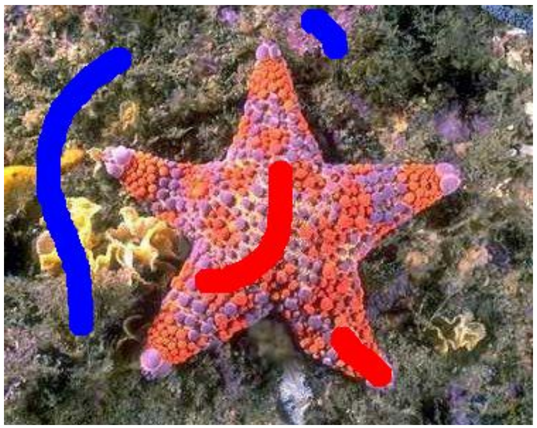


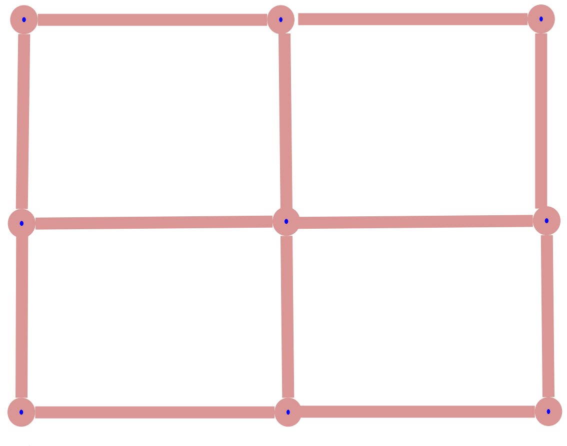
$P(z_i|x_i=1)$

Maximum likelihood:

$$x^* = \underset{x}{\operatorname{argmax}}\, P(z|x) = \underset{x}{\operatorname{argmax}} \prod_{x_i} P(z_i|x_i)$$

Prior $P(x|z) \sim P(z|x)\ P(x)$





$$P(x) = 1/f \prod_{i,j \in N_4} \theta_{ij}(x_i, x_j)$$

$$f = \sum_x \prod_{i,j \in N} \theta_{ij}(x_i, x_j)$$ "partition function"

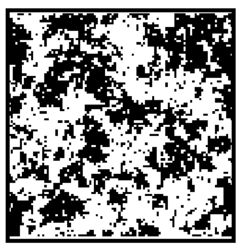
$$\theta_{ij}(x_i, x_j) = \exp\{-|x_i - x_j|\}$$ "ising prior"

(exp{-1}=0.36; exp{0}=1)

Prior

Pure Prior model: $$P(x) = 1/f \prod_{i,j \in N_4} \exp\{-|x_i - x_j|\}$$

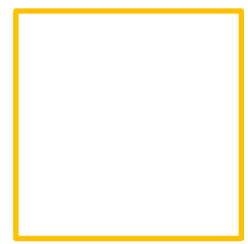
Faire Samples



P(x) = 0.011

Solutions with highest probability (mode)

P(x) = 0.012



P(x) = 0.012

Smoothness prior needs the likelihood

Posterior distribution

$P(x|z) \sim P(z|x)\, P(x)$

"Gibbs" distribution:

$P(x|z) = 1/f(z,w)\ \exp\{-E(x,z,w)\}$

$$E(x,z,w) = \sum_i \theta_i (x_i, z_i) + w \sum_{i,j \in N} \theta_{ij} (x_i, x_j)$$ **Energy**

Unary terms Pairwise terms

$\theta_i (x_i, z_i) = -\log P(z_i|x_i{=}1)\ x_i\ -\log P(z_i|x_i{=}0)\ (1-x_i)$ **Likelihood**

$\theta_{ij} (x_i, x_j) = |x_i - x_j|$ **prior**

Energy minimization

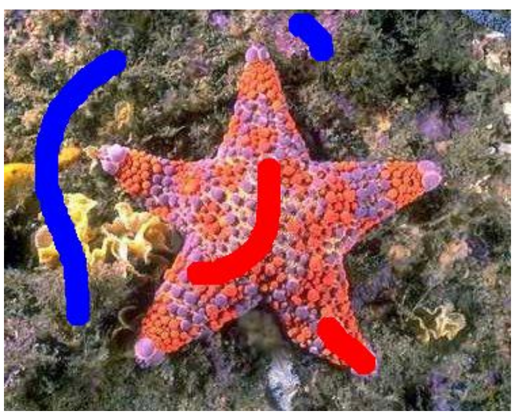
$$P(x|z) = 1/f(z,w)\ \exp\{-E(x,z,w)\}$$

$$f(z,w) = \sum_{X} \exp\{-E(x,z,w)\}$$

$$-\log P(x|z) = -\log\ (1/f(z,w)) + E(x,z,w)$$

$$x^* = \underset{X}{\operatorname{argmin}}\ E(x,z,w)$$ **MAP same as minimum Energy**

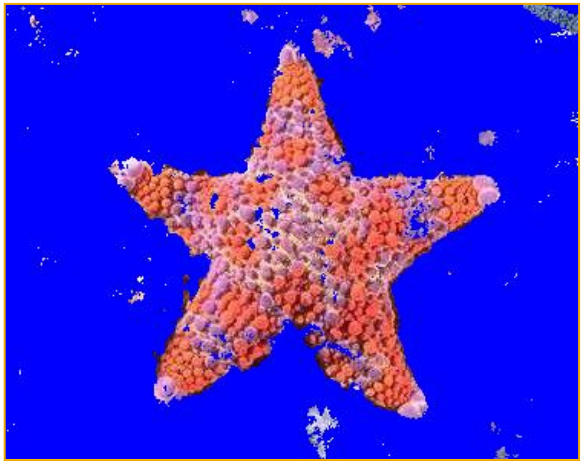
$$E(x,z,w) = \sum_{i} \theta_i\ (x_i,z_i) + w \sum_{i,j \in N} \theta_{ij}\ (x_i,x_j)$$



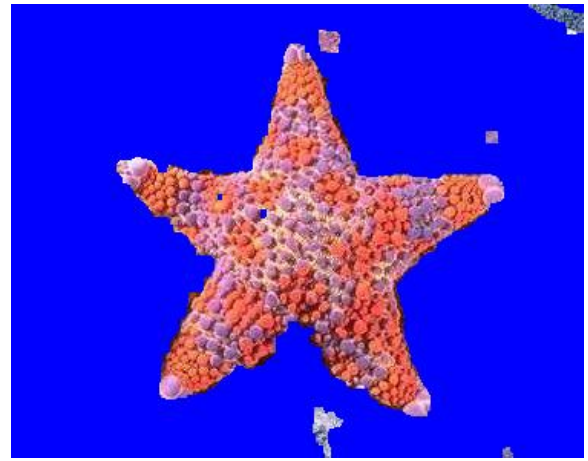
MAP; Global min E

ML

Weight prior and likelihood



w =0



w =10

w =40

w =200

$$E(x,z,w) = \sum \theta_i (x_i,z_i) + w\sum \theta_{ij} (x_i,x_j)$$

Outline

- Introduction
- MRFs and CRFs in Vision
- Optimisation techniques and Comparison

Random Field Models for Computer Vision

Model :

- discrete or continuous variables?
- discrete or continuous space?
- Dependence between variables?
- …

Inference/Optimisation

- Combinatorial optimization: e.g. Graph Cut
- Message Passing: e.g. BP, TRW
- Iterated Conditional Modes (ICM)
- LP-relaxation: e.g. Cutting-plane
- Problem decomposition + subgradient
- …

Applications:

- 2D/3D Image segmentation
- Object Recognition
- 3D reconstruction
- Stereo matching
- Image denoising
- Texture Synthesis
- Pose estimation
- Panoramic Stitching
- …

Learning:

- Exhaustive search (grid search)
- Pseudo-Likelihood approximation
- Training in Pieces
- Max-margin
- …

Introducing Factor Graphs

Write probability distributions as Graphical model:

- Direct graphical model
- Undirected graphical model *“traditionally used for MRFs”*
- Factor graphs *“best way to visualize the underlying energy”*

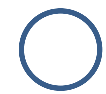
References:

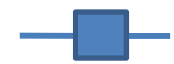
- Pattern Recognition and Machine Learning [Bishop ‘08, book, chapter 8]
- several lectures at the Machine Learning Summer School 2009
  (see video lectures)

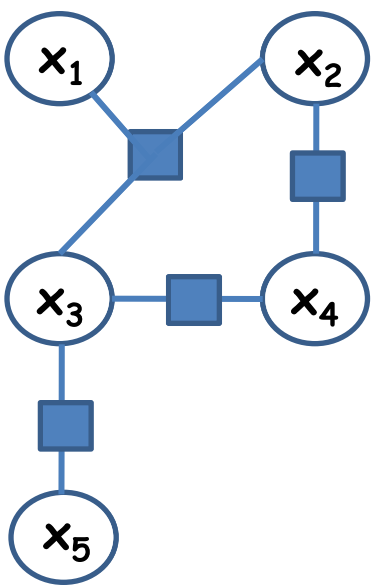
Factor Graphs

$P(x) \sim \exp\{-E(x)\}$
$E(x) = \theta(x_1,x_2,x_3) + \theta(x_2,x_4) + \theta(x_3,x_4) + \theta(x_3,x_5)$

Gibbs distribution
"4 factors"

unobserved 

variables are in same factor. 

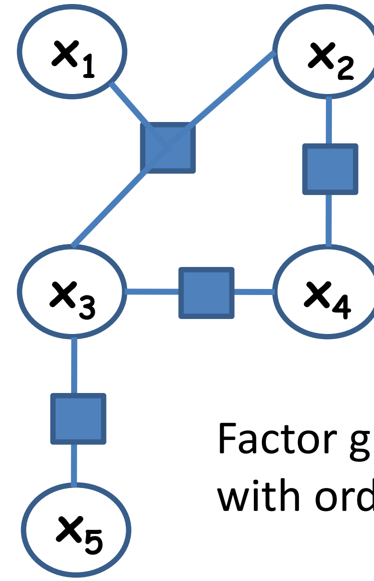


Factor graph

Definition “Order”

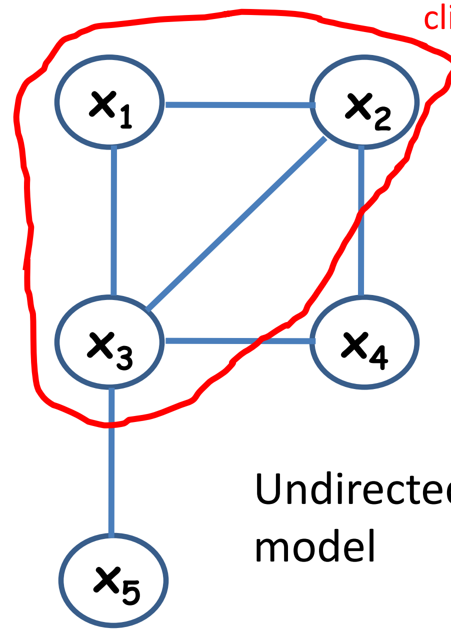
Definition “Order”:
The arity (number of variables) of the largest factor

$$E(X) = \underbrace{\theta(x_1,x_2,x_3)}_{\text{arity 3}} \; \underbrace{\theta(x_2,x_4)\;\theta(x_3,x_4)\;\theta(x_3,x_5)}_{\text{arity 2}}$$



Factor graph with order 3

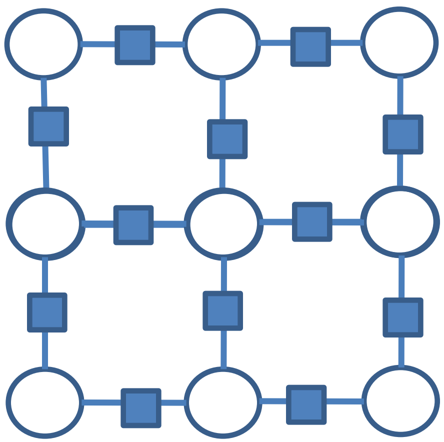
Triple clique



Undirected model

Extras:
- I will use “factor” and “clique” in the same way
- Not fully correct since clique may or may not decomposable
- Definition of “order” same for clique and factor (not always consistent in literature)
- Markov Random Field: Random Field with low-order factors/cliques.

Examples - Order

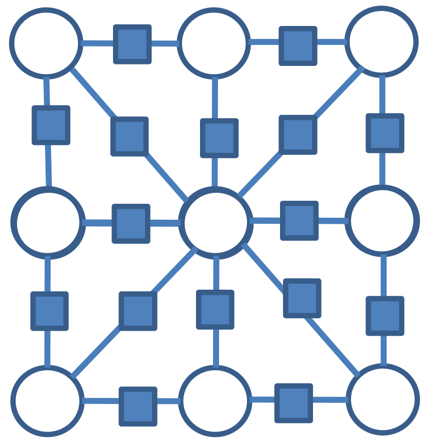


4-connected; pairwise MRF

$$E(x) = \sum_{i,j \in N_4} \theta_{ij}(x_i, x_j)$$

Order 2

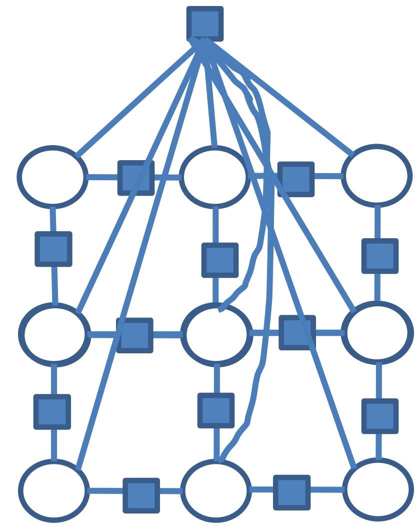
"Pairwise energy"



higher(8)-connected; pairwise MRF

$$E(x) = \sum_{i,j \in N_8} \theta_{ij}(x_i, x_j)$$

Order 2



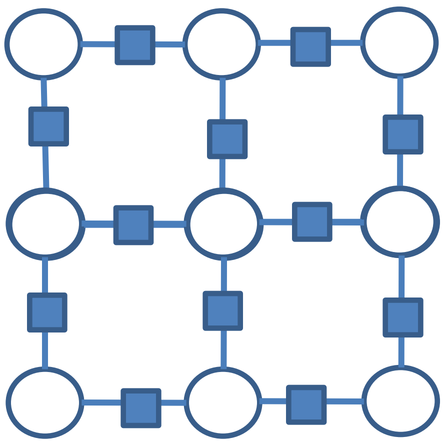
Higher-order RF

$$E(x) = \sum_{i,j \in N_4} \theta_{ij}(x_i, x_j) + \theta(x_1, \ldots, x_n)$$

Order n

"higher-order energy"

Random field models

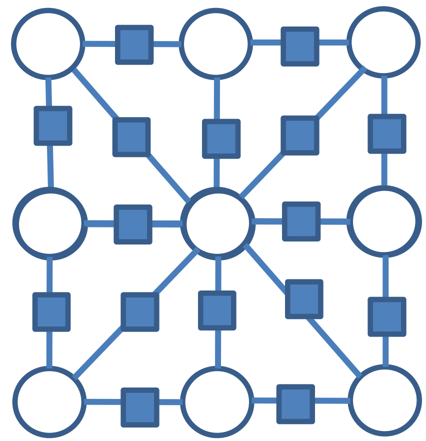


4-connected; pairwise MRF

$$E(x) = \sum_{i,j \,\epsilon\, N_4} \theta_{ij}\,(x_i,x_j)$$

Order 2

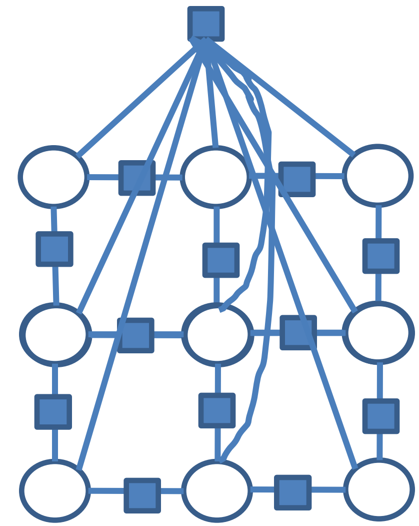
"Pairwise energy"



higher(8)-connected; pairwise MRF

$$E(x) = \sum_{i,j \,\epsilon\, N_8} \theta_{ij}\,(x_i,x_j)$$

Order 2



Higher-order RF

$$E(x) = \sum_{i,j \,\epsilon\, N_4} \theta_{ij}\,(x_i,x_j) + \theta(x_1,\dots,x_n)$$

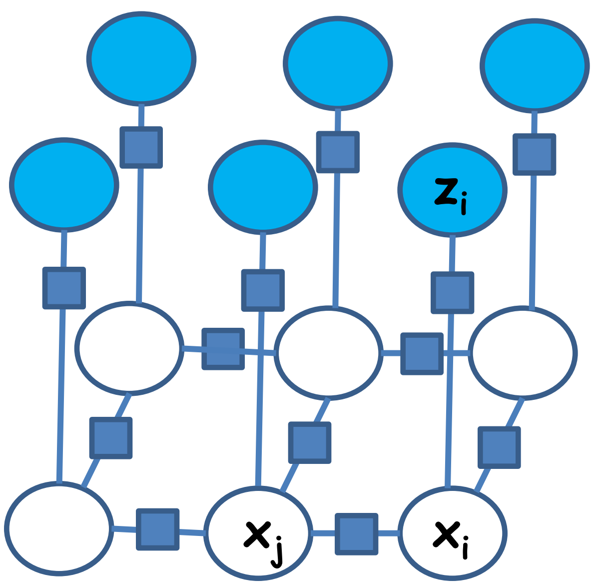
Order n

"higher-order energy"

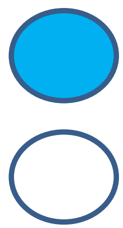
Example: Image segmentation

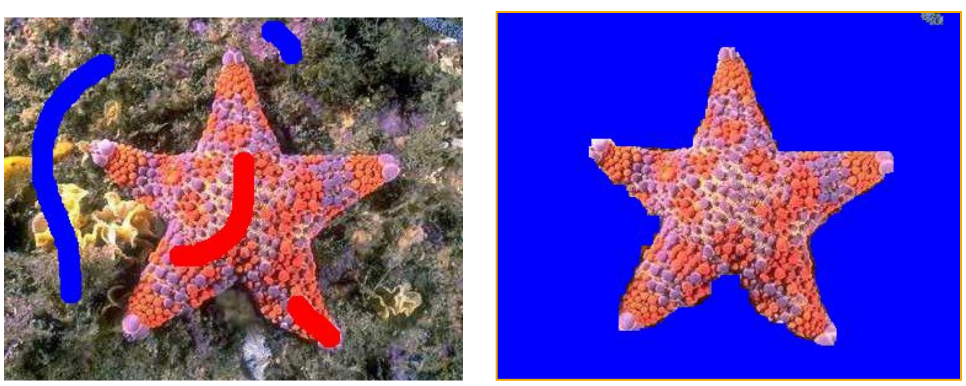
$$P(x|z) \sim \exp\{-E(x)\}$$

$$E(x) = \sum_{i} \theta_i (x_i, z_i) + \sum_{i,j \in N_4} \theta_{ij} (x_i, x_j)$$



Factor graph





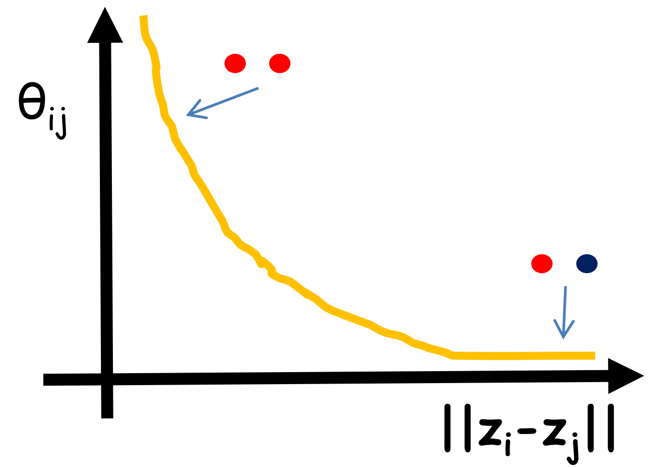
Segmentation: Conditional Random Field

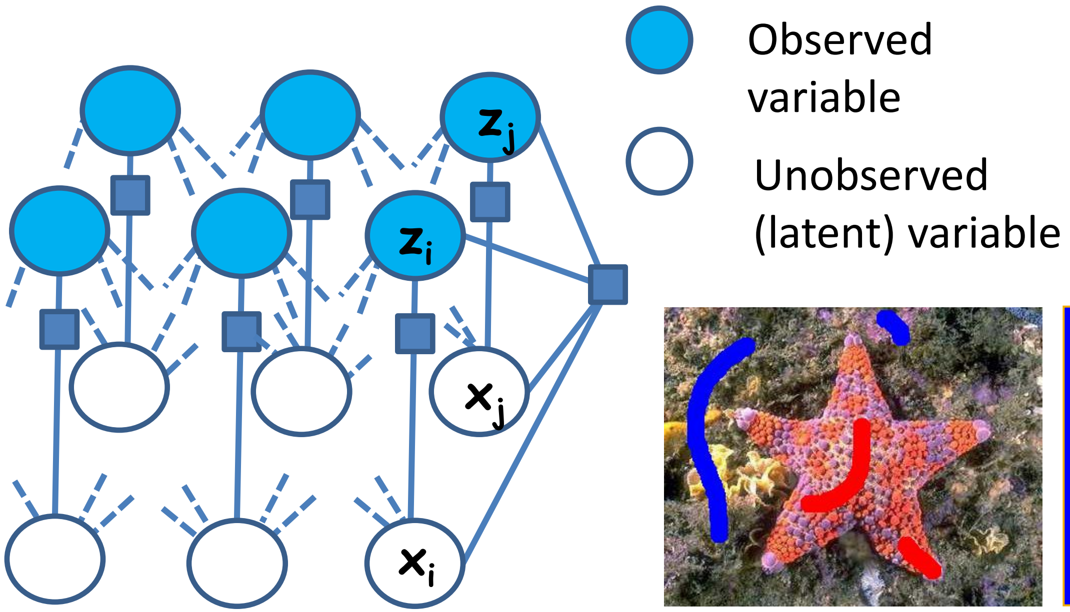
$$E(x) = \sum_{i} \theta_i (x_i, z_i) + \sum_{i,j \,\epsilon\, N_4} \theta_{ij} (x_i, x_j, z_i, z_j)$$

$$\theta_{ij} (x_i, x_j, z_i, z_j) = |x_i - x_j| \, (-\exp\{-\beta||z_i - z_j||\})$$

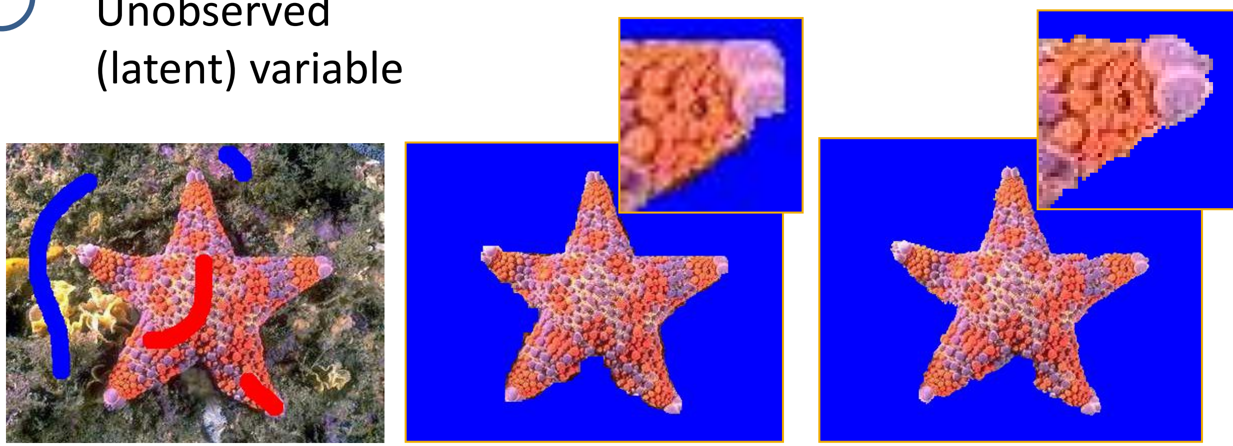
$$\beta = 2(\mathrm{Mean}(||z_i - z_j||_2))^{-1}$$

Conditional Random Field (CRF) no pure prior





Factor graph



MRF

CRF

Stereo matching

Image – left(a)

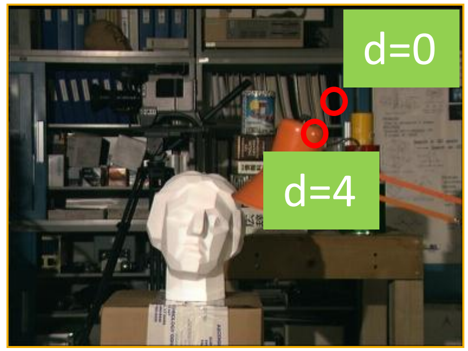


Image – right(b)

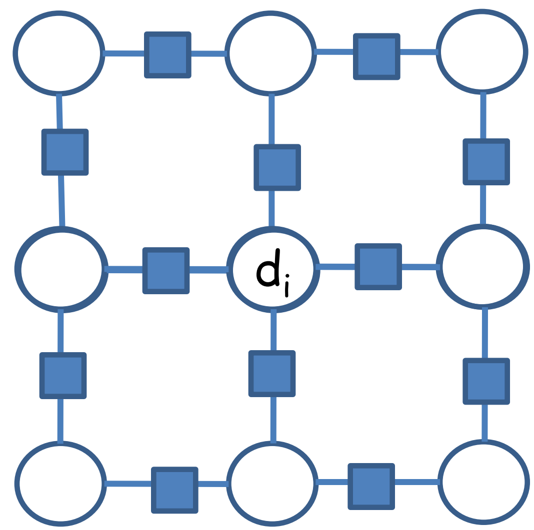
Ground truth depth

- Images rectified
- Ignore occlusion for now

Energy:

$E(d): \{0,\dots,D-1\}^n \to R$

Labels: d (depth/shift)



Stereo matching - Energy

Energy:

$$E(d): \{0,\dots,D\text{-}1\}^n \rightarrow R$$

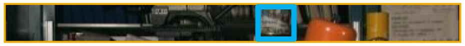
$$E(d) = \sum_i \theta_i(d_i) + \sum_{i,j \in N_4} \theta_{ij}(d_i,d_j)$$

Unary:

$$\theta_i(d_i) = (l_j - r_{i-d_i})$$

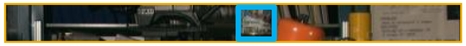
"SAD; Sum of absolute differences"

(many others possible, NCC,…)



left

i



right

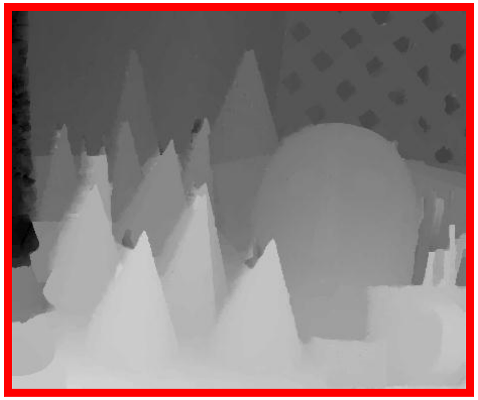
i-2

(d_i=2)

Pairwise:

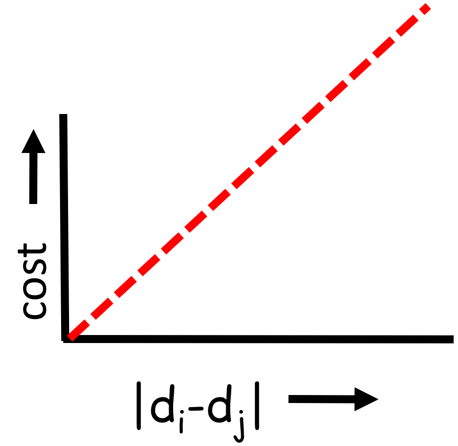
$$\theta_{ij}(d_i,d_j) = g(|d_i - d_j|)$$

Stereo matching - prior



No truncation
(global min.)

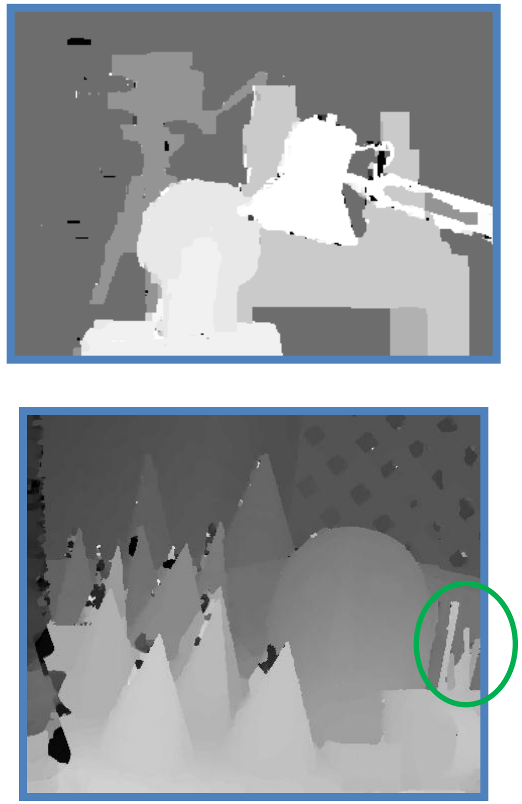
$\theta_{ij}\,(d_i,d_j) = g(|d_i\text{-}d_j|)$



[Olga Veksler PhD thesis,
Daniel Cremers et al.]

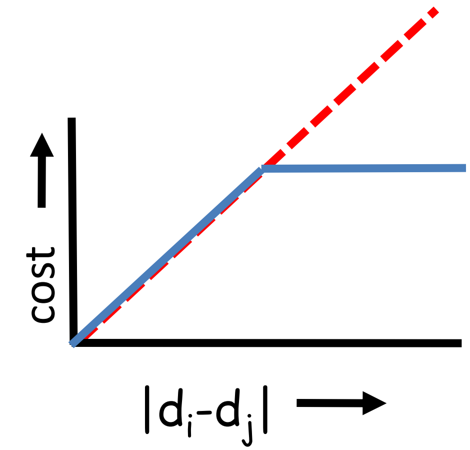
Stereo matching - prior

No truncation
(global min.)



with truncation
(NP hard optimization)

$$\theta_{ij}\,(d_i,d_j) = g(|d_i-d_j|)$$

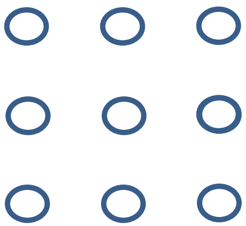


discontinuity preserving potentials
[Blake&Zisserman'83,'87]

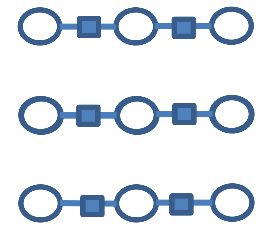
[Olga Veksler PhD thesis,
Daniel Cremers et al.]

Stereo matching

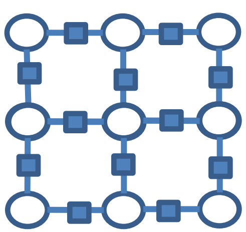
see http://vision.middlebury.edu/stereo/



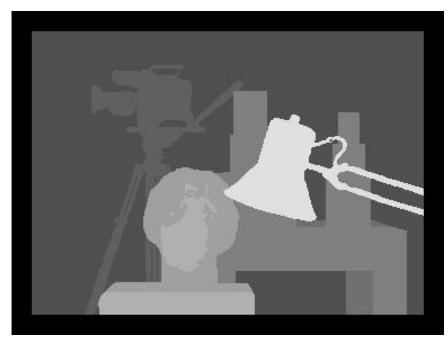
No MRF
Pixel independent (WTA)



No horizontal links
Efficient since independent chains

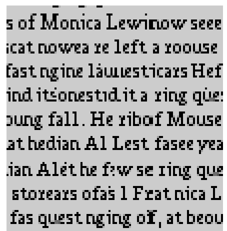


Pairwise MRF
[Boykov et al. ‘01]

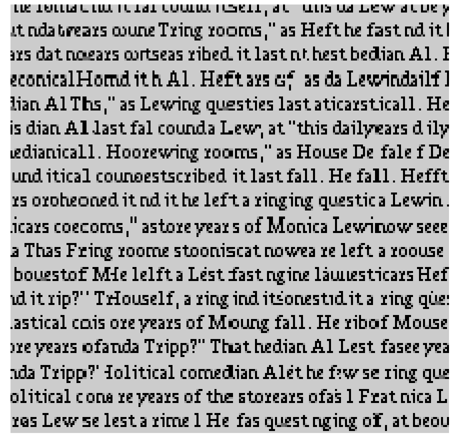


Ground truth

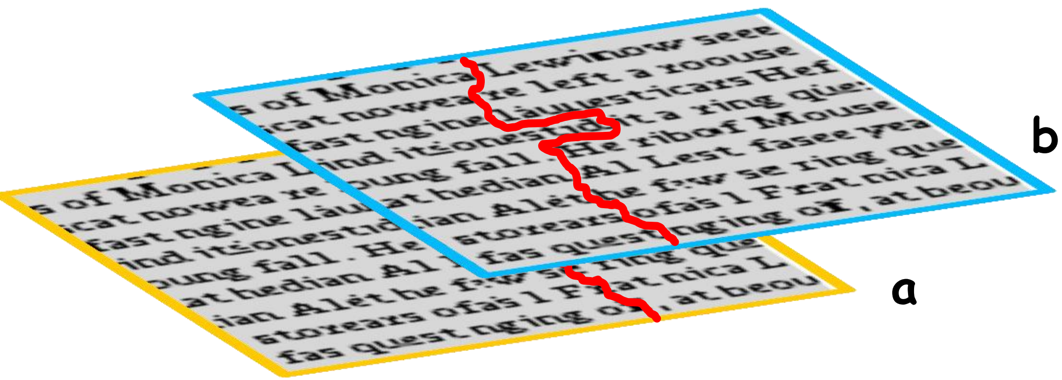
Texture synthesis



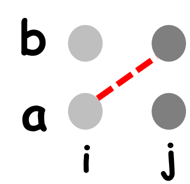
Input



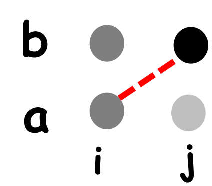
Output

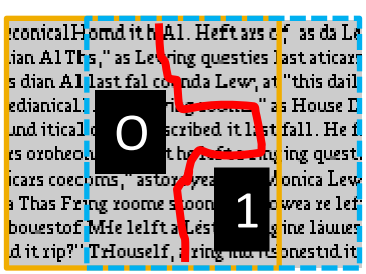


Good case:



Bad case:





$$E: \{0,1\}^n \rightarrow R$$

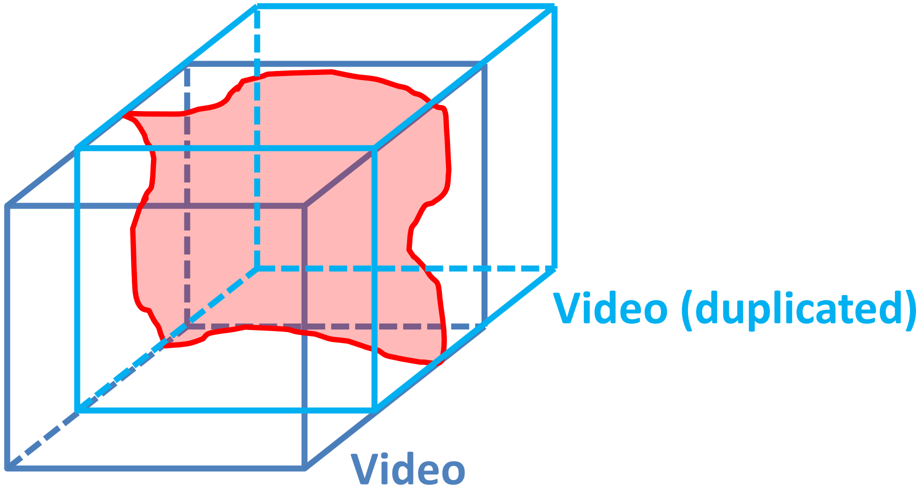
$$E(x) = \sum_{i,j \in N_4} |x_i - x_j| \; [\; |a_i - b_i| + |a_j - b_j| \;]$$

[Kwatra et. al. Siggraph '03]

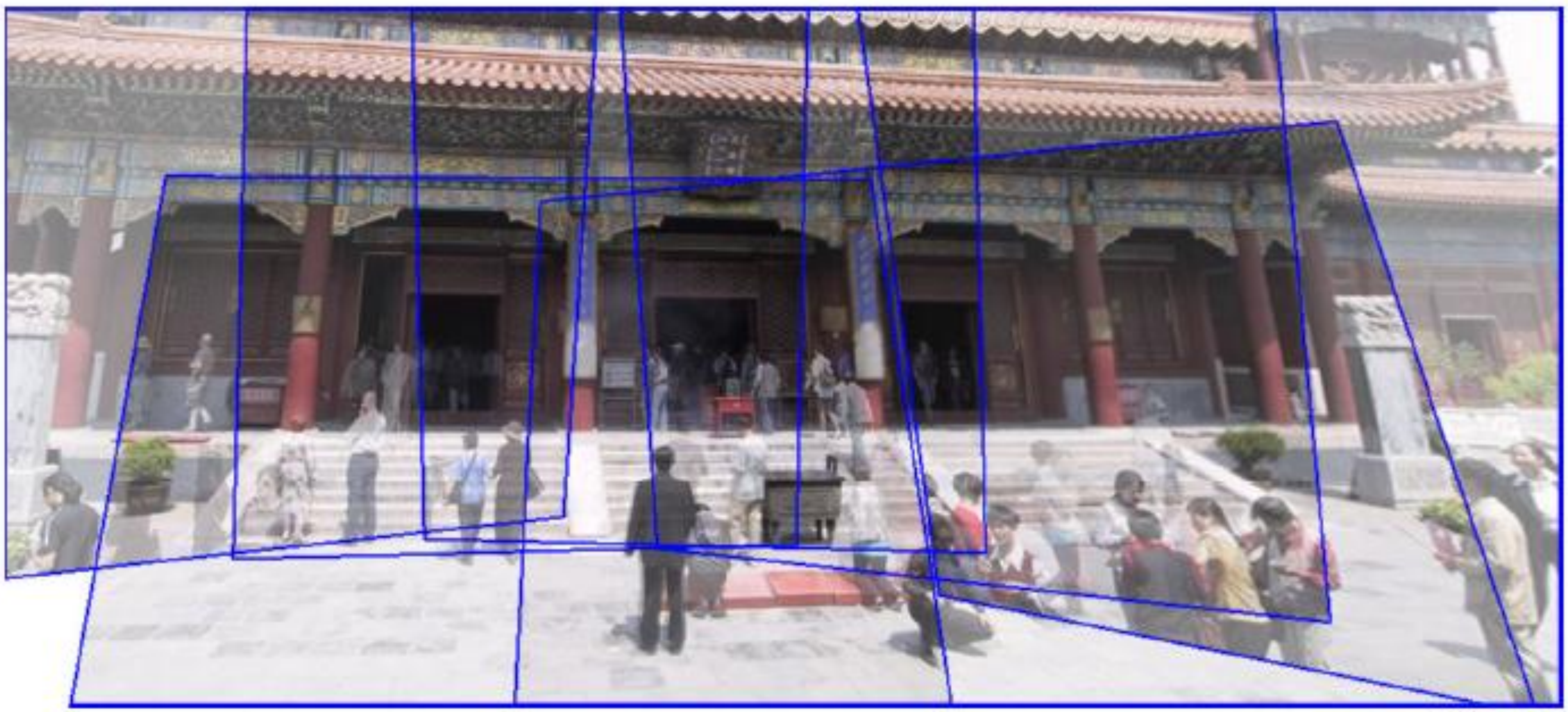
Video Synthesis

Input

Output



Panoramic stitching



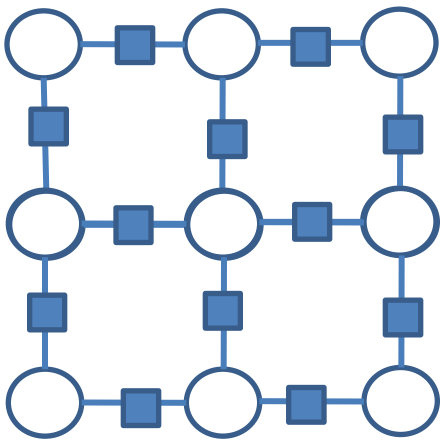
Panoramic stitching

Recap: 4-connected MRFs

- A lot of useful vision systems are based on 4-connected pairwise MRFs.

- Possible Reason (see Inference part): a lot of fast and good (globally optimal) inference methods exist

Random field models

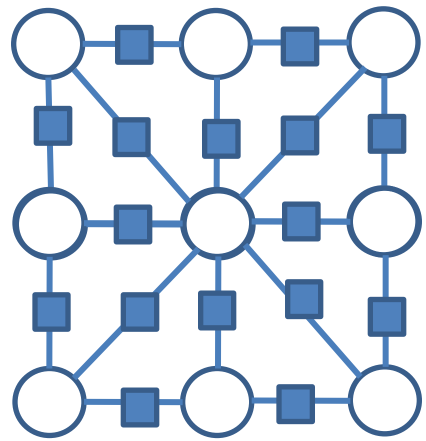


4-connected; pairwise MRF

$$E(x) = \sum_{i,j \,\epsilon\, N_4} \theta_{ij}(x_i, x_j)$$

Order 2

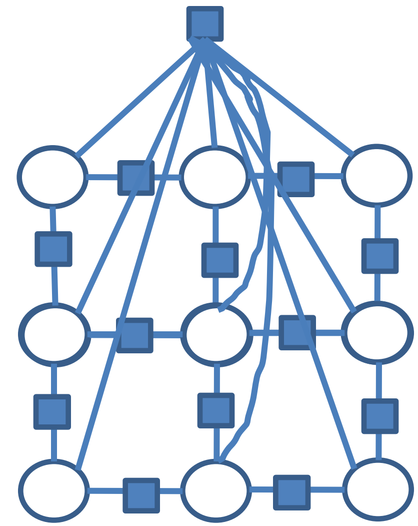
"Pairwise energy"



higher(8)-connected; pairwise MRF

$$E(x) = \sum_{i,j \,\epsilon\, N_8} \theta_{ij}(x_i, x_j)$$

Order 2



Higher-order RF

$$E(x) = \sum_{i,j \,\epsilon\, N_4} \theta_{ij}(x_i, x_j) + \theta(x_1, \ldots, x_n)$$

Order n

"higher-order energy"

Why larger connectivity?

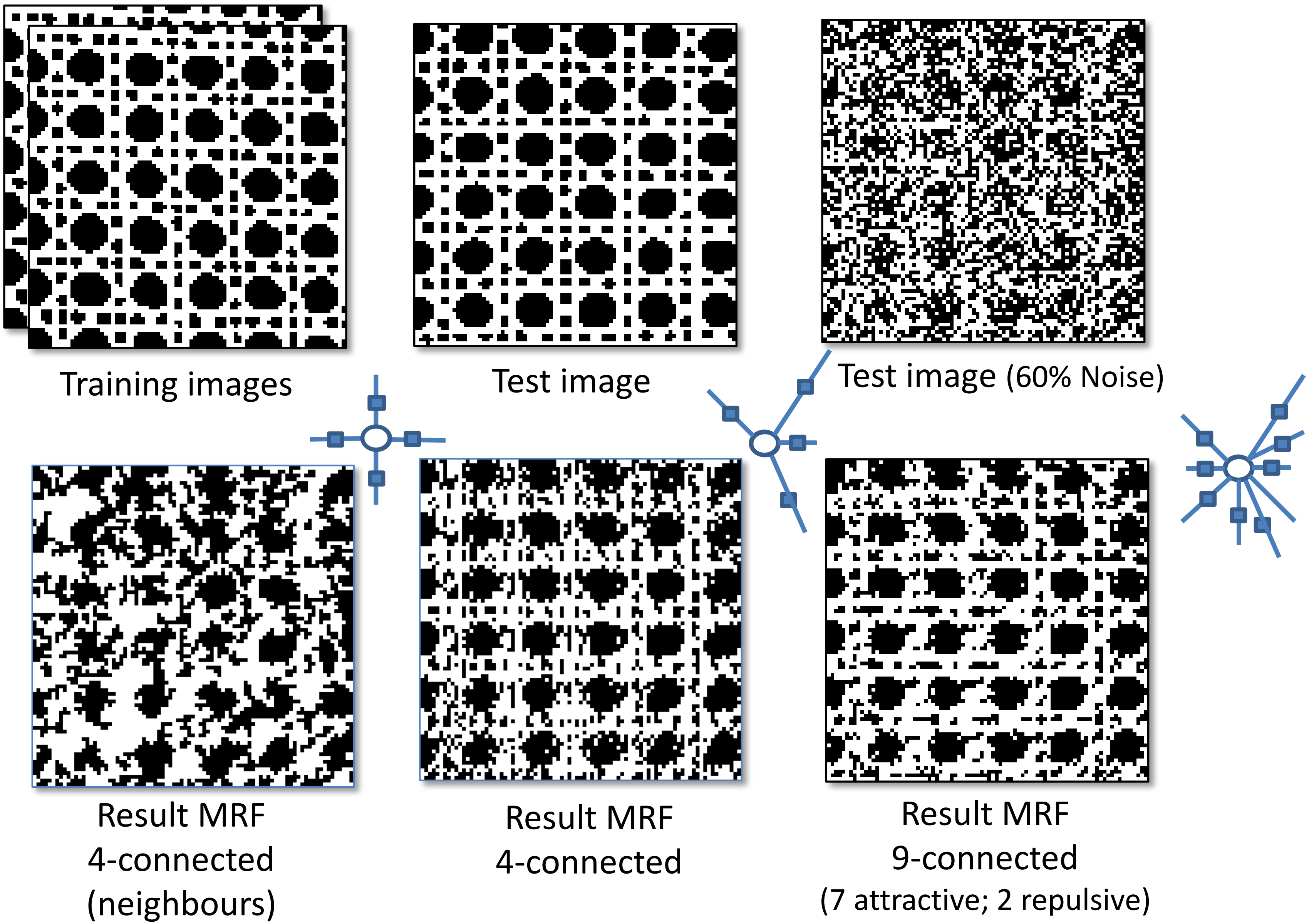
We have seen…

- “Knock-on” effect (each pixel influences each other pixel)
- Many good systems

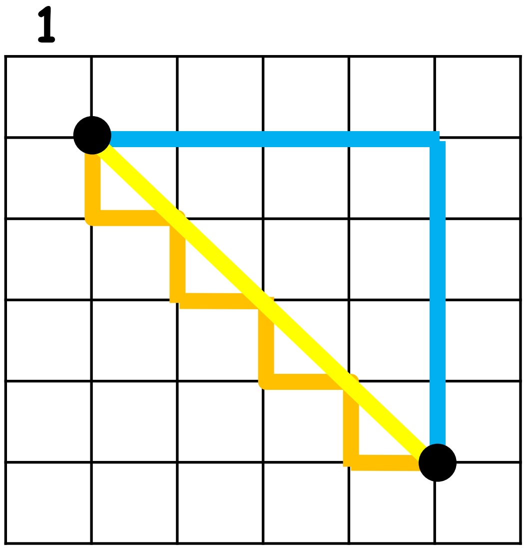
What is missing:

1. Modelling real-world texture (images)
2. Reduce discretization artefacts
3. Encode complex prior knowledge
4. Use non-local parameters

Reason 1: Texture modelling



Reason2: Discretization artefacts



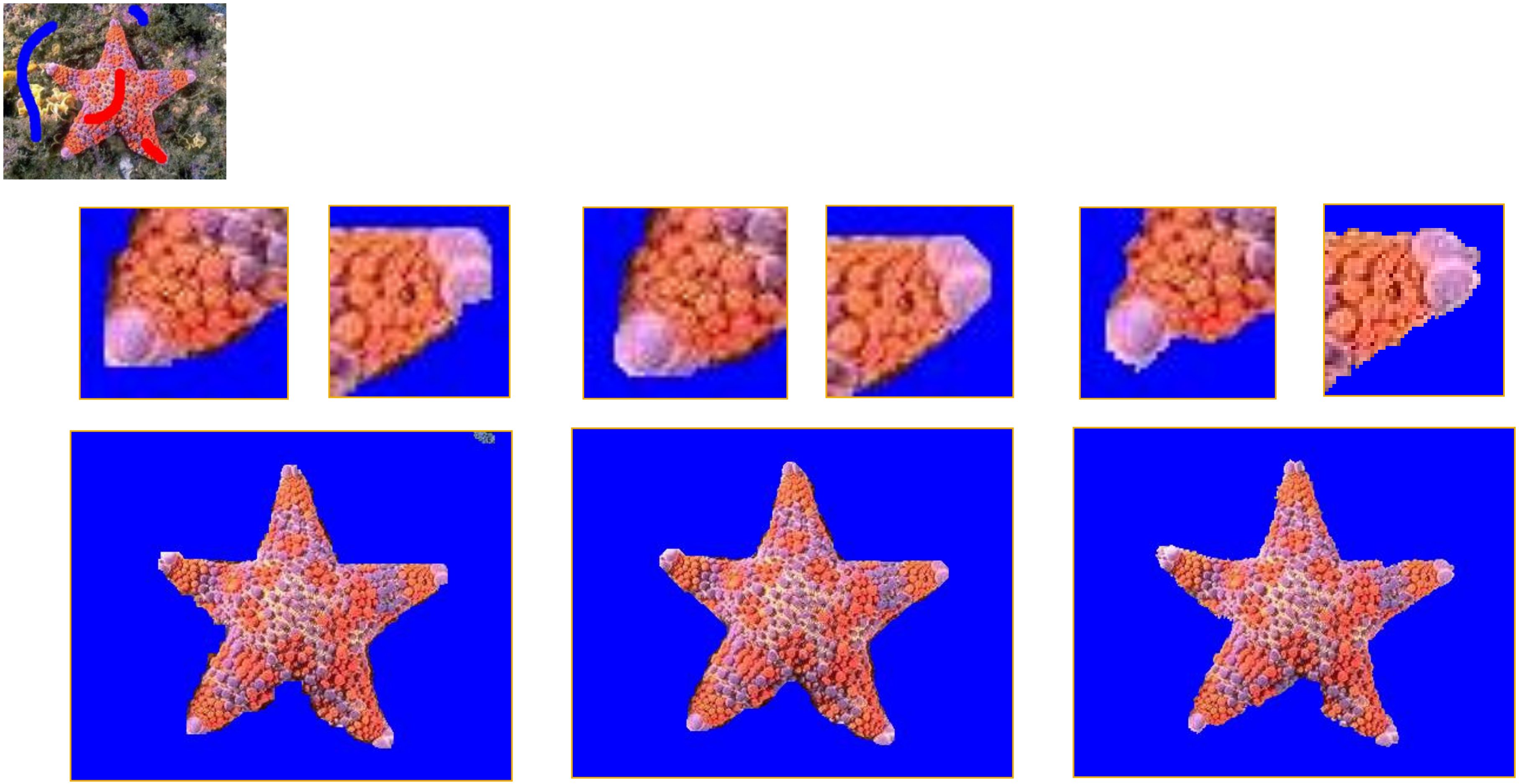
Length of the paths:

	Eucl.	4-con.	8-con.
	5.65	6.28	5.08
	8	6.28	6.75

Larger connectivity can model true Euclidean length (also other metric possible)

[Boykov et al. ‘03, ‘05]

Reason2: Discretization artefacts



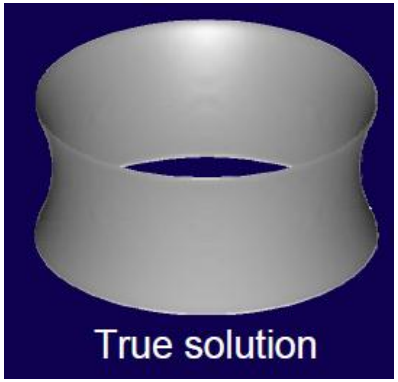
4-connected Euclidean

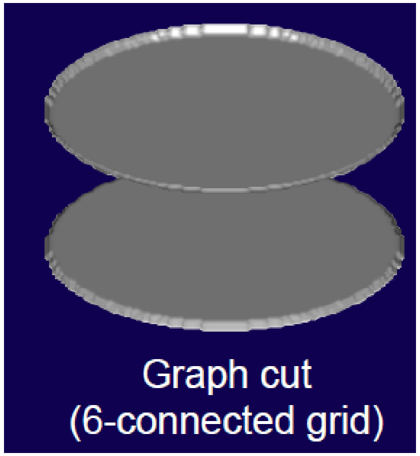
8-connected Euclidean (MRF)

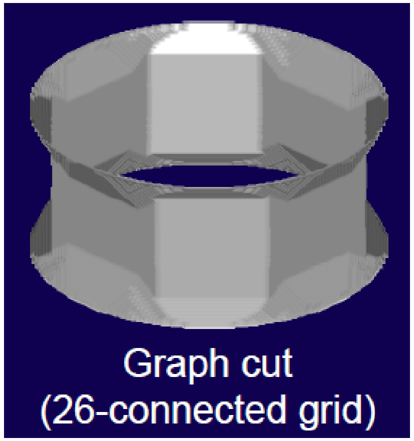
8-connected geodesic (CRF)

[Boykov et al. ‘03; ‘05]

3D reconstruction





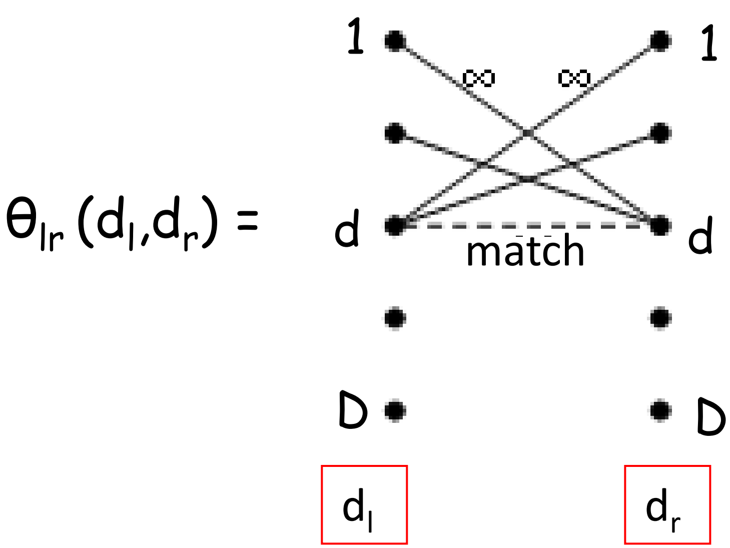


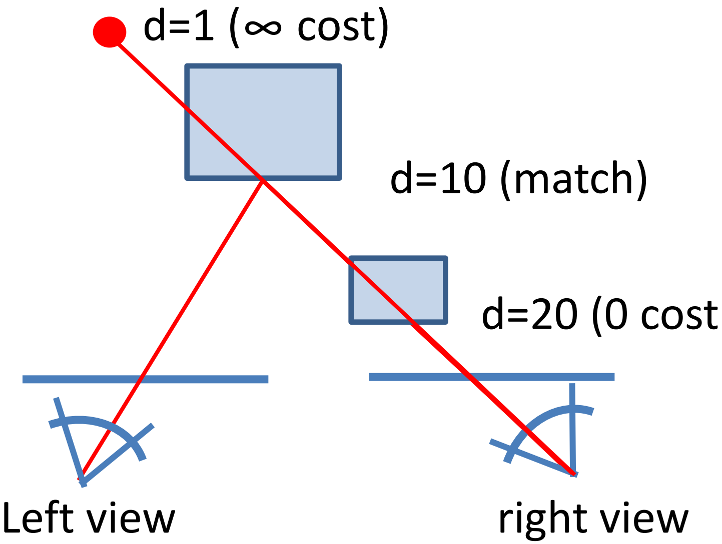
[Slide credits: Daniel Cremers]

Reason 3: Encode complex prior knowledge:
Stereo with occlusion

$E(d): \{1,\ldots,D\}^{2n} \rightarrow R$

Each pixel is connected to **D** pixels in the other image

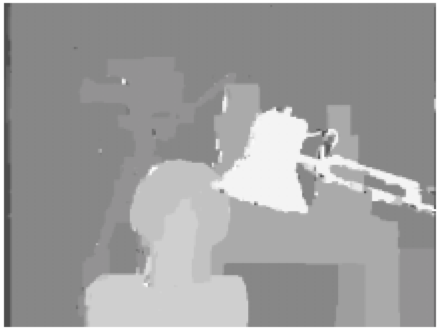




Stereo with occlusion

Ground truth

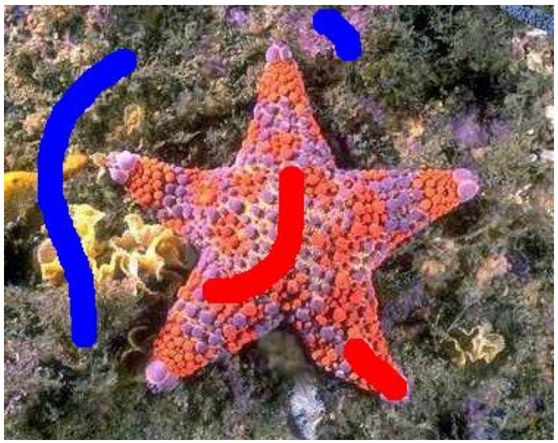
Stereo with occlusion
[Kolmogrov et al. ‘02]



Stereo without occlusion
[Boykov et al. ‘01]

Reason 4: Use Non-local parameters:

Interactive Segmentation (GrabCut)



[Boykov and Jolly ’01]

GrabCut [Rother et al. ’04]

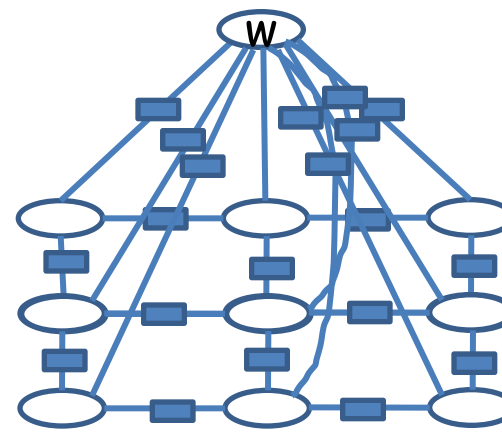
A meeting with the Queen

Reason 4: Use Non-local parameters:
Interactive Segmentation (GrabCut)

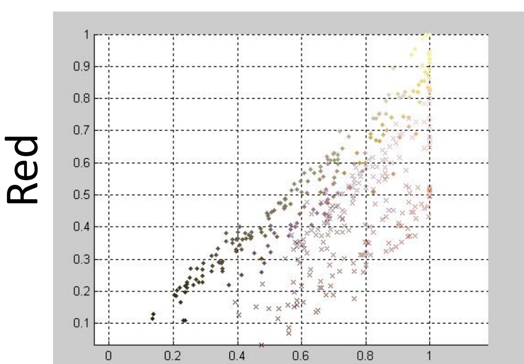

Model jointly segmentation and color model:

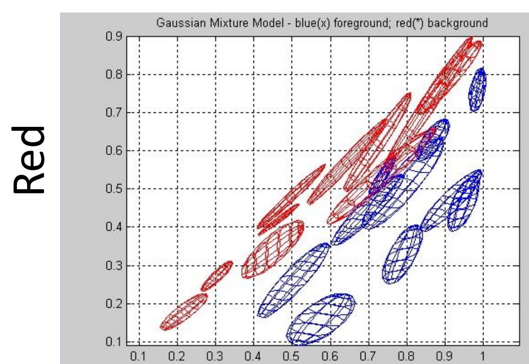
$$E(x,w): \{0,1\}^n \times \{GMMs\} \rightarrow R$$

$$E(x,w) = \sum_i \theta_i (x_i,w) + \sum_{i,j \in N_4} \theta_{ij} (x_i,x_j)$$



An object is a compact set of colors:

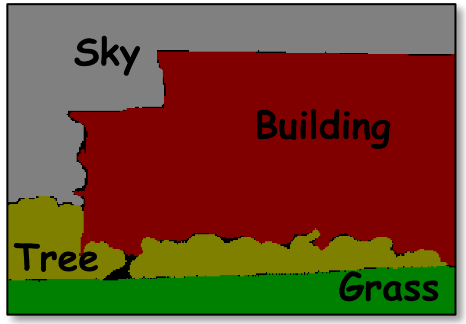




[Rother et al. Siggraph '04]

Reason 4: Use Non-local parameters:

Object recognition & segmentation



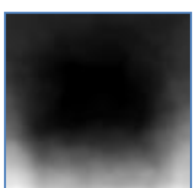
$$E(x,\omega) = \sum_i \theta_i(\omega, x_i) + \sum \theta_i(x_i) + \sum \theta_i(x_i) + \sum_{i,j} \theta_{ij}(x_i, x_j)$$

(color) (location) (class) (edge aware ising prior)

$x_i \in \{1,\ldots,K\}$ for K object classes

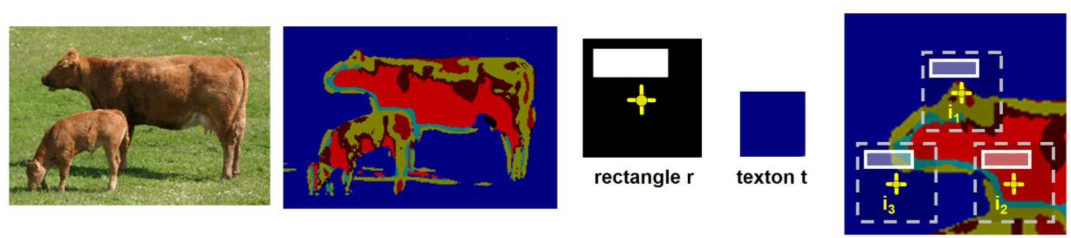
Location

sky



grass

Class (boosted textons)

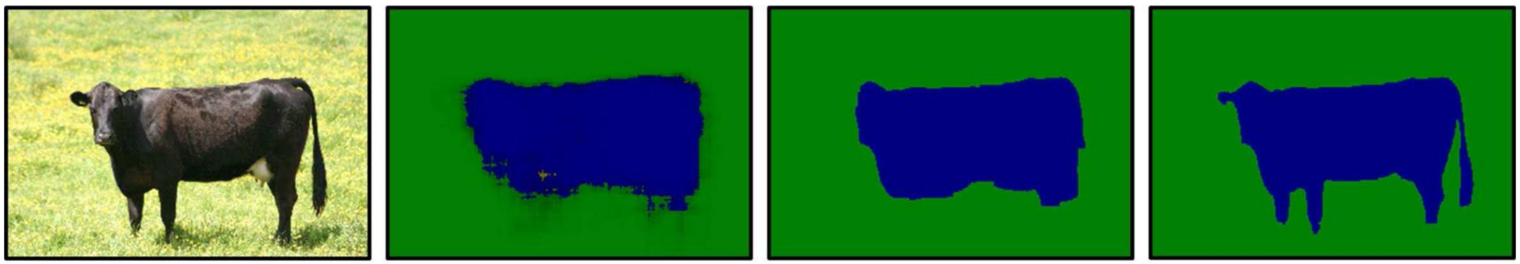


(a) Input image (b) Texton map (c) Feature pair = (r,t) (d) Superimposed rectangles

[TextonBoost; Shotton et al, '06]

Reason 4: Use Non-local parameters:

Object recognition & segmentation



(a) (b) 69.6% (c) 70.3% (d) 72.2%

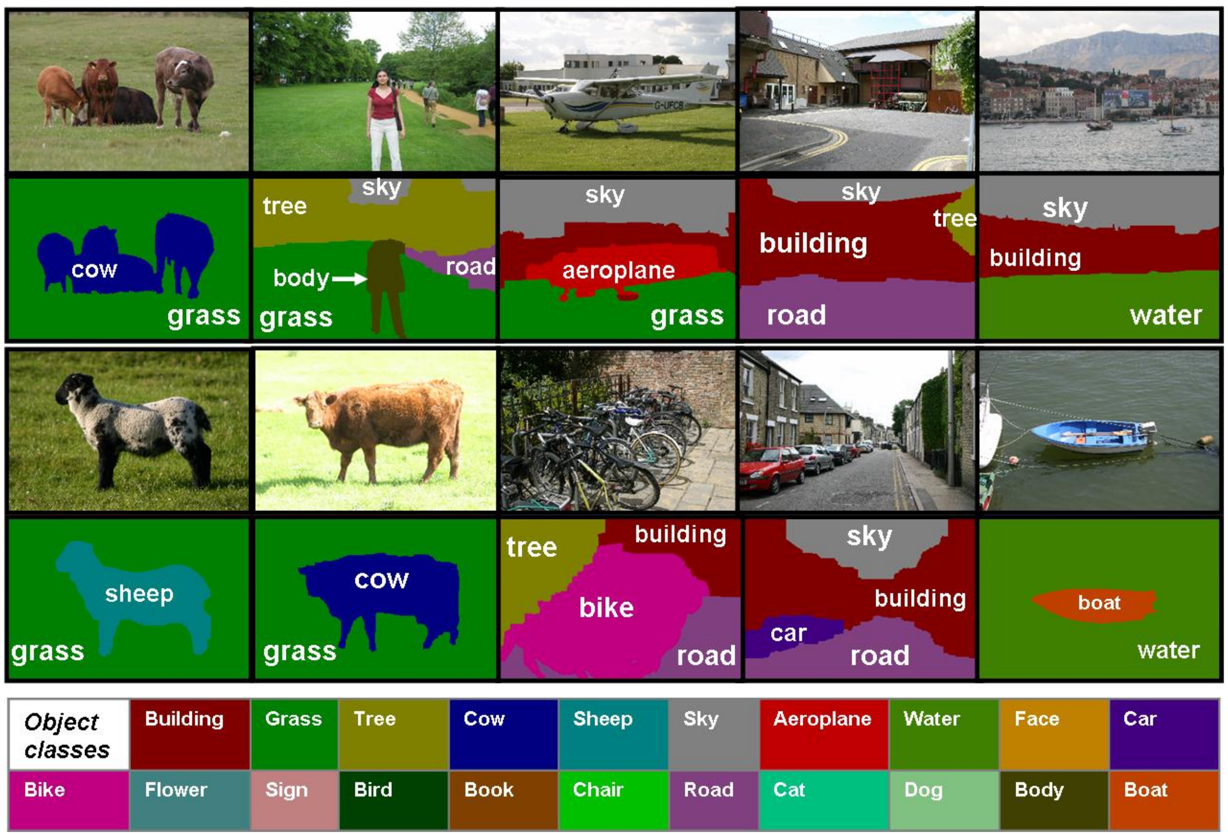
Class+ location | + edges | + color

[TextonBoost; Shotton et al, '06]

Reason 4: Use Non-local parameters:

Object recognition & segmentation

Good results …



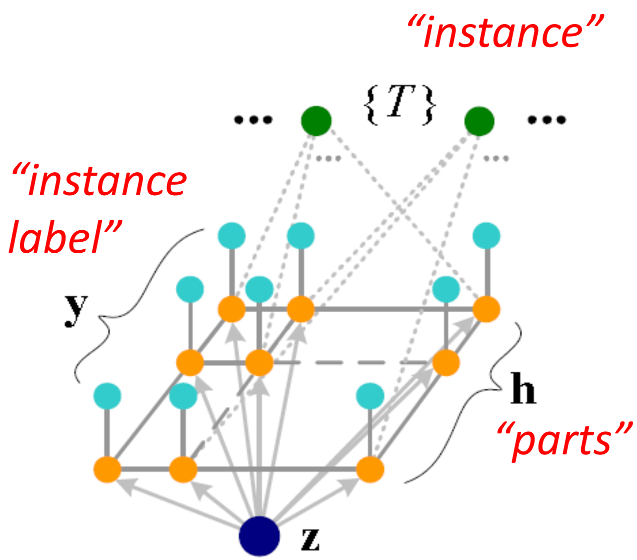
[TextonBoost; Shotton et al, '06]

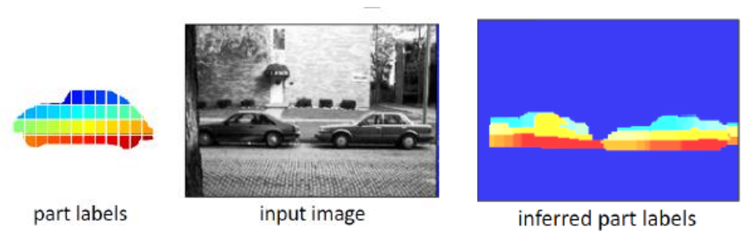
Reason 4: Use Non-local parameters:

Object recognition & segmentation

Failure cases…

Reason 4: Use Non-local parameters: Recognition with Latent/Hidden CRFs



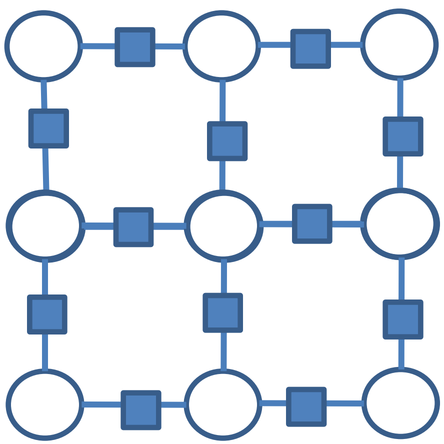


[LayoutCRF Winn et al. ’06]

- Many other examples: ObjCut Kumar et. al. ’05; Deformable Part Model Felzenszwalb et al.; CVPR ’08; PoseCut Bray et al. ’06, LayoutCRF Winn et al. ’06

- Maximizing over hidden variables

Random field models

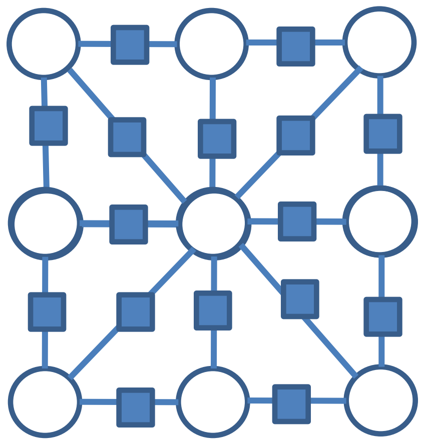


4-connected; pairwise MRF

$$E(x) = \sum_{i,j \in N_4} \theta_{ij}(x_i, x_j)$$

Order 2

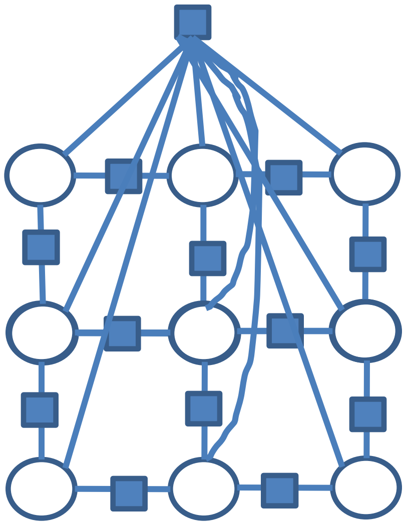
"Pairwise energy"



higher(8)-connected; pairwise MRF

$$E(x) = \sum_{i,j \in N_8} \theta_{ij}(x_i, x_j)$$

Order 2



Higher-order RF

$$E(x) = \sum_{i,j \in N_4} \theta_{ij}(x_i, x_j) + \theta(x_1, \ldots, x_n)$$

Order n

"higher-order energy"

Why Higher-order Functions?

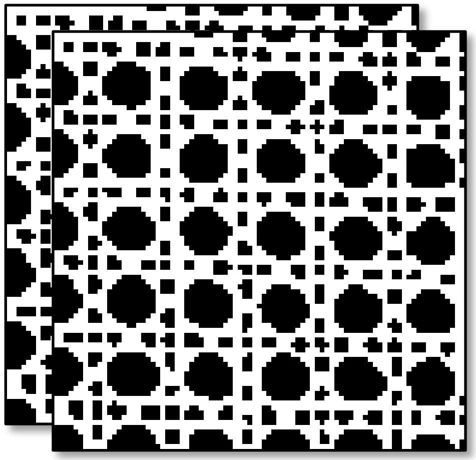
In general $\theta(x_1,x_2,x_3) \neq \theta(x_1,x_2) + \theta(x_1,x_3) + \theta(x_2,x_3)$

<u>Reasons for higher-order MRFs:</u>

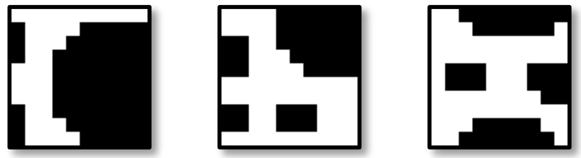
1. Even better image(texture) models:
   - Field-of Expert [FoE, Roth et al. ‘05]
   - Curvature [Woodford et al. ‘08]

2. Use **global** Priors:
   - Connectivity [Vicente et al. ‘08, Nowizin et al. ‘09]
   - Encode better training statistics [Woodford et al. ‘09]
   - Convert global variables to global factors [Vicente et al. ‘09]

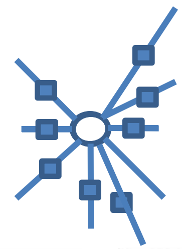
Reason1: Better Texture Modelling

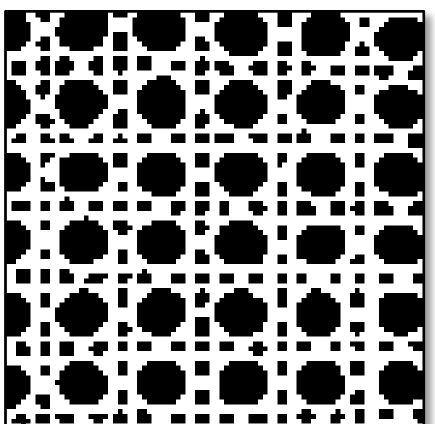


Training images



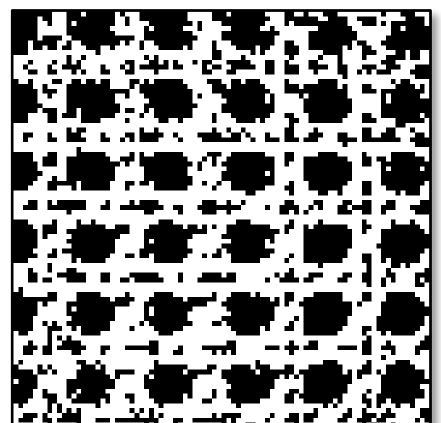
Higher Order Structure not Preserved



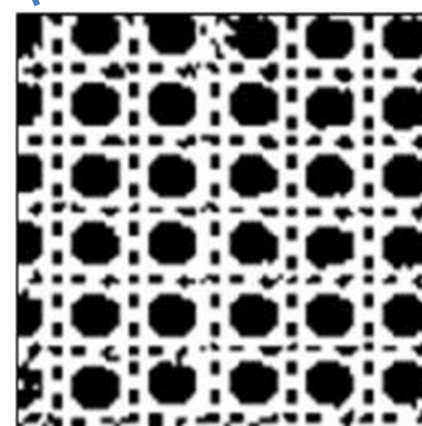


Test Image

Test Image (60% Noise)



Result pairwise MRF
9-connected

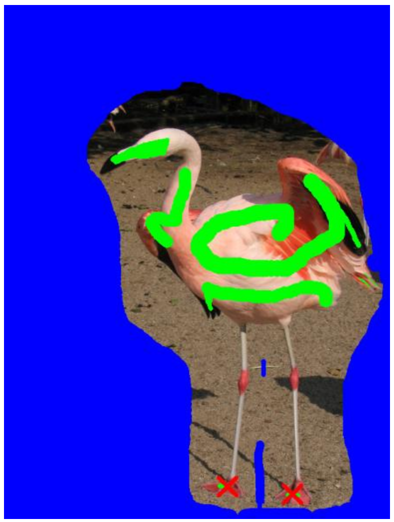


Higher-order MRF

[Rother et al. CVPR ‘09]

Reason 2: Use global Prior

Foreground object must be connected:



User input

Standard MRF:
Removes noise (+)
Shrinks boundary (-)

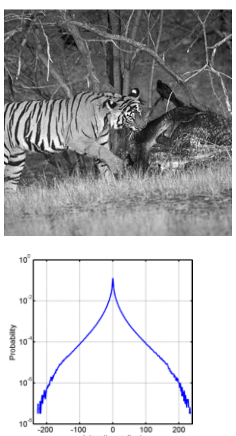
with connectivity

$$E(x) = P(x) + h(x) \qquad \text{with } h(x) = \begin{cases} \infty & \text{if not 4-connected} \\ 0 & \text{otherwise} \end{cases}$$

[Vicente et. al. '08
Nowizin et al '09]

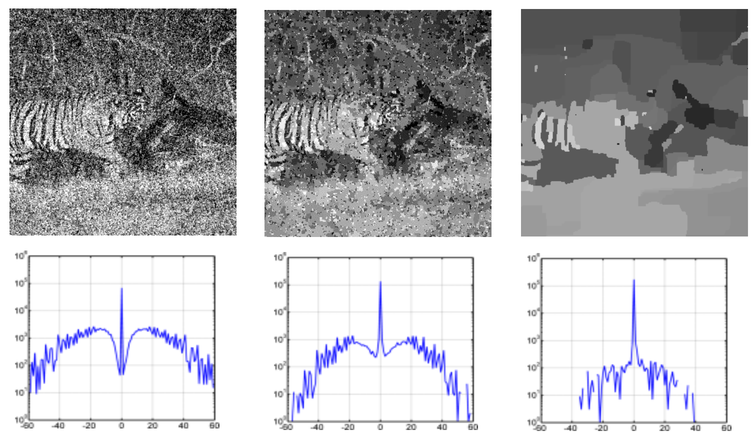
Reason 2: Use global Prior

Introduce a global term, which controls global statistic:

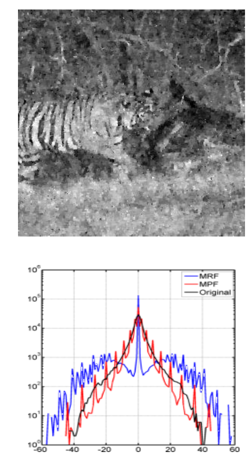


Ground truth

Noisy input

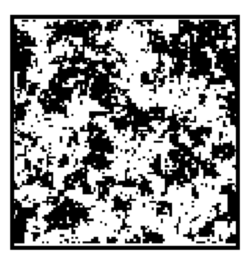


Pairwise MRF –
Increase Prior strength



Global gradient prior

Remember:

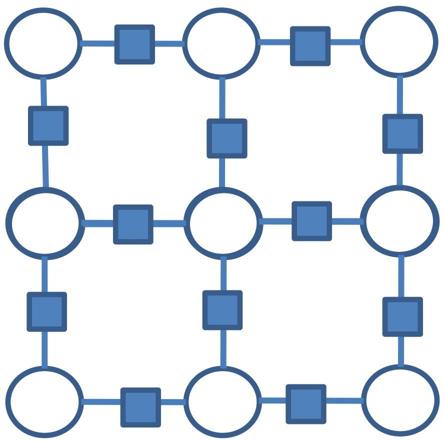


P(x) = 0.011

P(x) = 0.012

[Woodford et. al. ICCV ‘09]

Random field models

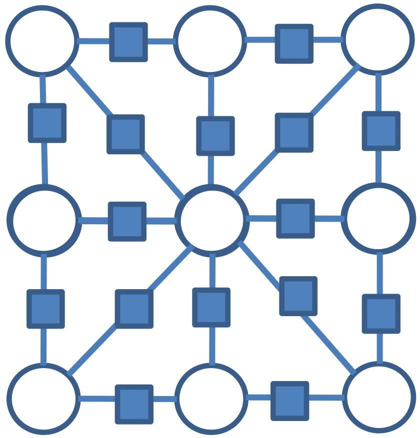


4-connected;
pairwise MRF

$$E(x) = \sum_{i,j \in N_4} \theta_{ij}(x_i, x_j)$$

Order 2

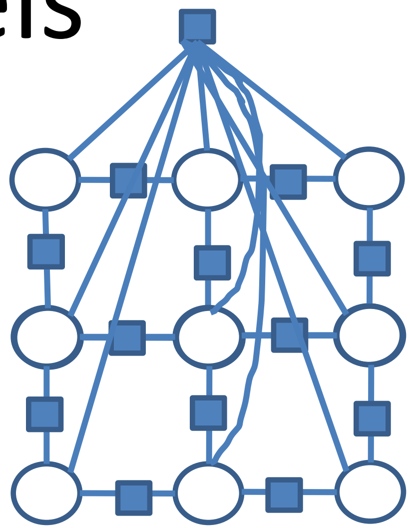
"Pairwise energy"



higher(8)-connected;
pairwise MRF

$$E(x) = \sum_{i,j \in N_8} \theta_{ij}(x_i, x_j)$$

Order 2



Higher-order RF

$$E(x) = \sum_{i,j \in N_4} \theta_{ij}(x_i, x_j) + \theta(x_1, \ldots, x_n)$$

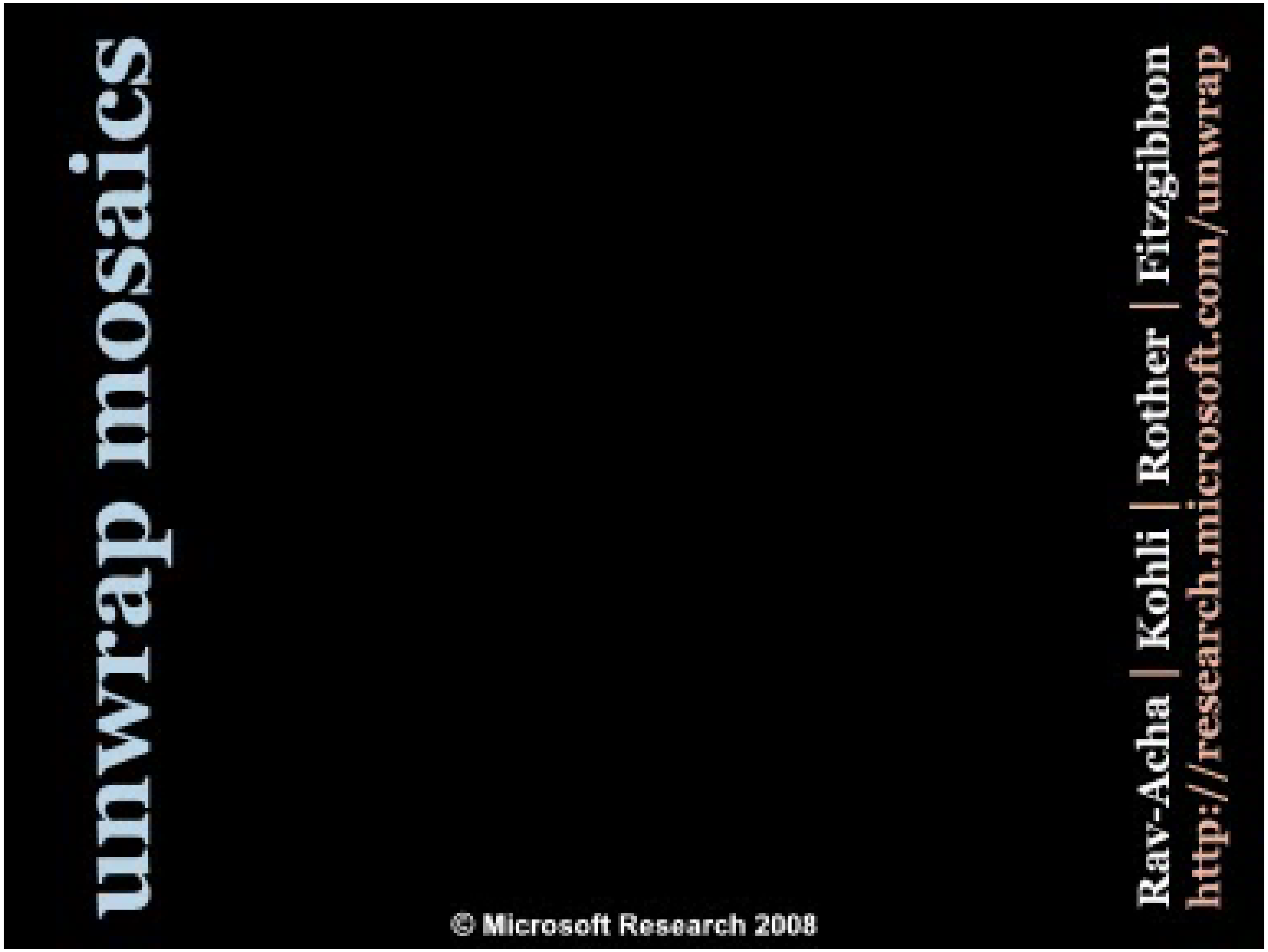
Order n

"higher-order energy"

.... all useful models,

but how do I optimize them?

Advanced CRF system



[Unwrap Mosaic, Rav-Acha et al. Siggraph ’08]

Outline

- Introduction
- MRFs and CRFs in Vision
- Optimisation techniques and Comparison

Why is good optimization important?

Input: Image sequence

[Data courtesy from Oliver Woodford]

Output: New view

Problem: Minimize a binary 4-connected pair-wise MRF
(choose a colour-mode at each pixel)

[Fitzgibbon et al. ‘03]

Why is good optimization important?

Ground Truth

Graph Cut with truncation
[Rother et al. ‘05]

Belief Propagation

ICM, Simulated
Annealing

QPBOP [Boros et al. ’06, Rother et al. ‘07]
Global Minimum

Recap

$$E(\mathbf{x}) = \boxed{\sum_i f_i(x_i)} + \boxed{\sum_{ij} g_{ij}(x_i, x_j)} + \boxed{\sum_c h_c(\mathbf{x}_c)}$$

Unary *Pairwise* *Higher Order*

Label-space:

Binary: $x_i \in \{0,1\}$
Multi-label: $x_i \in \{0,\ldots,K\}$

Inference – Big Picture

- Combinatorial Optimization (main part)
  - Binary, pairwise MRF: Graph cut, BHS (QPBO)
  - Multiple label, pairwise: move-making; transformation
  - Binary, higher-order factors: transformation
  - Multi-label, higher-order factors: move-making + transformation
- Dual/Problem Decomposition
  - Decompose (NP-)hard problem into tractable once. Solve with e.g. sub-gradient technique
- Local search / Genetic algorithms
  - ICM, simulated annealing

Inference – Big Picture

- Message Passing Techniques
  - Methods can be applied to any model in theory (higher order, multi-label, etc.)
  - BP, TRW, TRW-S

- LP-relaxation (not covered)
  - Relax original problem (e.g. {0,1} to [0,1]) and solve with existing techniques (e.g. sub-gradient)
  - Can be applied any model (dep. on solver used)
  - Connections to message passing (TRW) and combinatorial optimization (QPBO)

Inference – Big Picture: Higher-order models

- Arbitrary potentials are only tractable for order <7 (memory, computation time)

- For ≥ 7 potentials need some structure to be exploited in order to make them tractable
  (e.g. cost over number of labels)

Function Minimization: The Problems

- Which functions are exactly solvable?

- Approximate solutions of NP-hard problems

Function Minimization: The Problems

- ## Which functions are exactly solvable?

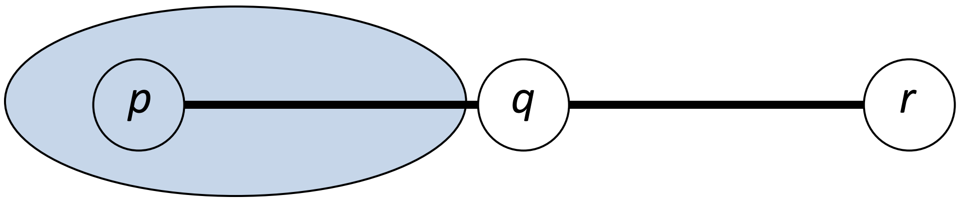
  Boros Hammer [1965], Kolmogorov Zabih [ECCV 2002, PAMI 2004] , Ishikawa [PAMI 2003], Schlesinger [EMMCVPR 2007], Kohli Kumar Torr [CVPR2007, PAMI 2008] , Ramalingam Kohli Alahari Torr [CVPR 2008] , Kohli Ladicky Torr [CVPR 2008, IJCV 2009] , Zivny Jeavons [CP 2008]

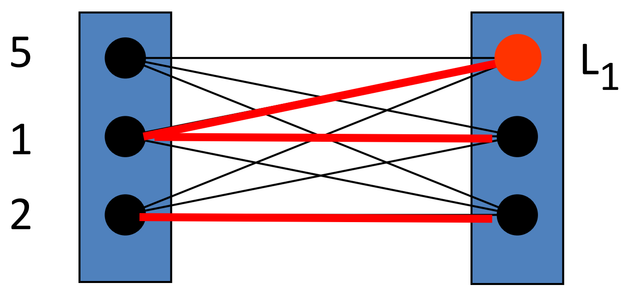
- ## Approximate solutions of NP-hard problems

  Schlesinger [1976], Kleinberg and Tardos [FOCS 99], Chekuri et al. [2001], Boykov et al. [PAMI 2001], Wainwright et al. [NIPS 2001], Werner [PAMI 2007], Komodakis [PAMI 2005], Lempitsky et al. [ICCV 2007], Kumar et al. [NIPS 2007], Kumar et al. [ICML 2008], Sontag and Jakkola [NIPS 2007], Kohli et al. [ICML 2008], Kohli et al. [CVPR 2008, IJCV 2009], Rother et al. [2009]

Message Passing Chain: Dynamic Programming

$f(x_p) + g_{pq}(x_p, x_q)$ with Potts model $g_{pq} = 2\,(x_p \neq x_q)$





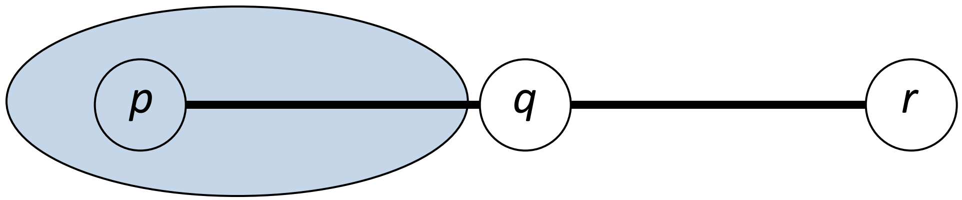
$$M_{p\text{->}q}(L_1) = \min_{x_p} f(x_p) + g_{pq}(x_p, L_1)$$

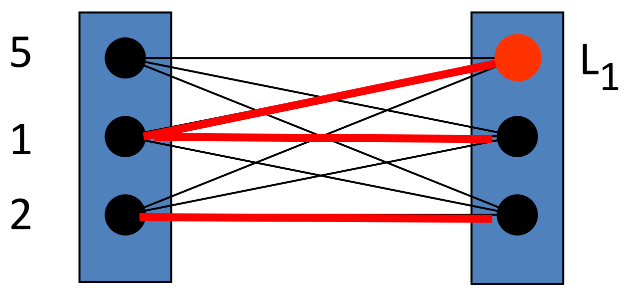
$$= \min(5+0,\ 1+2,\ 2+2)$$

$$M_{p\text{->}q}(L_1, L_2, L_3) = (3,1,2)$$

Message Passing Chain: Dynamic Programming

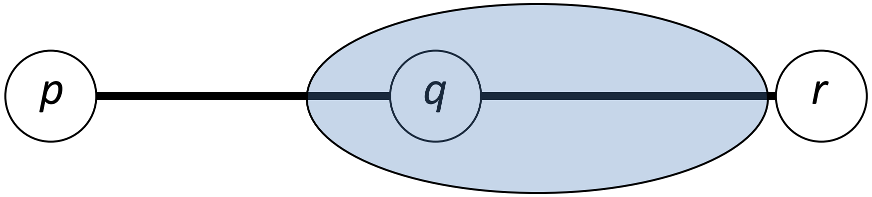
$f(x_p) + g_{pq}(x_p, x_q)$ with Potts model $g_{pq} = 2\,(x_p \neq x_q)$

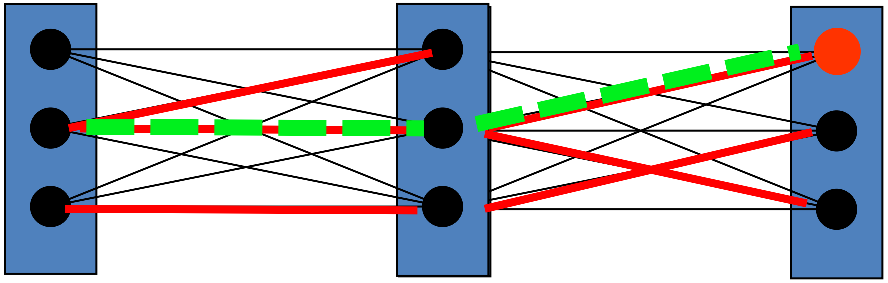




Message Passing Chain: Dynamic Programming

$$M_{q\text{->}r}(L_i) = \min_{x_q} M_{p\text{->}q} + f(x_q) + g_{qr}(x_q, L_i)$$





Get optimal labeling for x_r :

$$\min_{x_r} M_{q\text{->}r} + f(x_r)$$

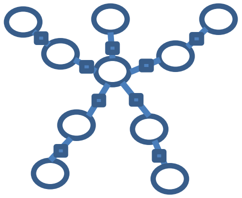
This gives min E

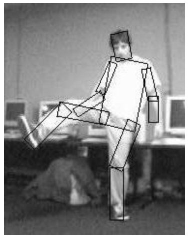
Trace back path to get minimum cost labeling **x**

Global minimum in linear time ☺

Message Passing Techniques

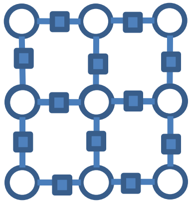
- Exact on Trees, e.g. chain



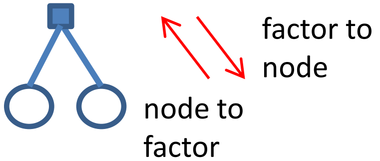


[Felzenschwalb et al '01]

- Loopy graphs: many techniques: BP, TRW, TRW-S, Diffusion:
  - Message update rules differ
  - Compute (approximate) MAP or marginals $P(x_i \mid x_{V\setminus\{i\}})$
  - Connections to LP-relaxation (TRW tries to solve MAP LP)



- Higher-order MRFs: Factor graph BP



[See details in tutorial ICCV '09, CVPR '10]

Combinatorial Optimization

- Binary, pairwise
  - Solvable problems
  - NP-hard
- Multi-label, pairwise
  - Transformation to binary
  - move-making
- Binary, higher-order
  - Transformation to pairwise
  - Problem decomposition

Binary functions that can be solved exactly

Pseudo-boolean function $f:\{0,1\}^n \rightarrow \mathbb{R}$ is submodular if

$$f(A) + f(B) \geq f(A \vee B) + f(A \wedge B)$$

for all $A,B \in \{0,1\}^n$

(OR) (AND)

Example: $n = 2$, $A = [1,0]$, $B = [0,1]$

$$f([1,0]) + f([0,1]) \geq f([1,1]) + f([0,0])$$

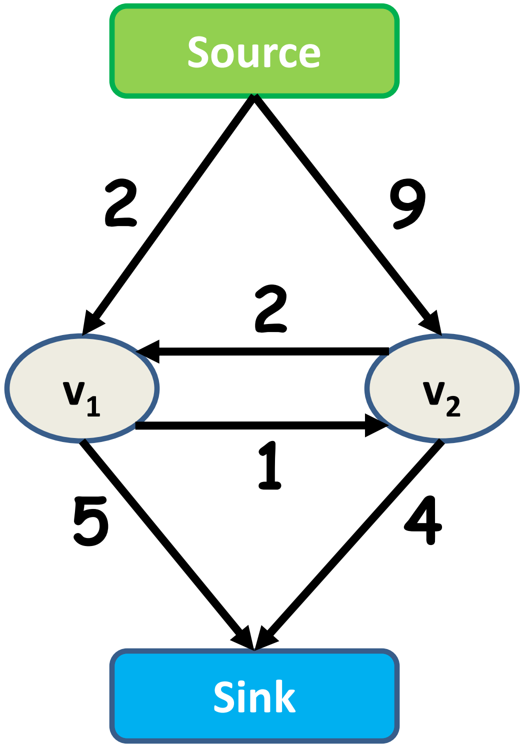
Property : Sum of **submodular** functions is **submodular**

Binary Image Segmentation Energy is submodular

$$E(x) = \sum_i c_i x_i + \sum_{i,j} d_{ij} |x_i - x_j|$$

Submodular binary, pairwise MRFs:

Maxflow-MinCut or GraphCut algorithm [Hammer et al. '65]



Graph (V, E, C)

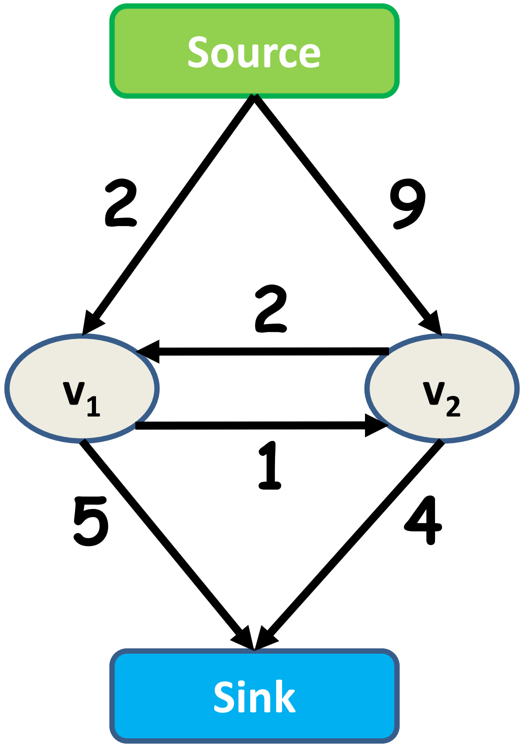
Vertices V = $\{v_1, v_2 \ldots v_n\}$

Edges E = $\{(v_1, v_2) \ldots\}$

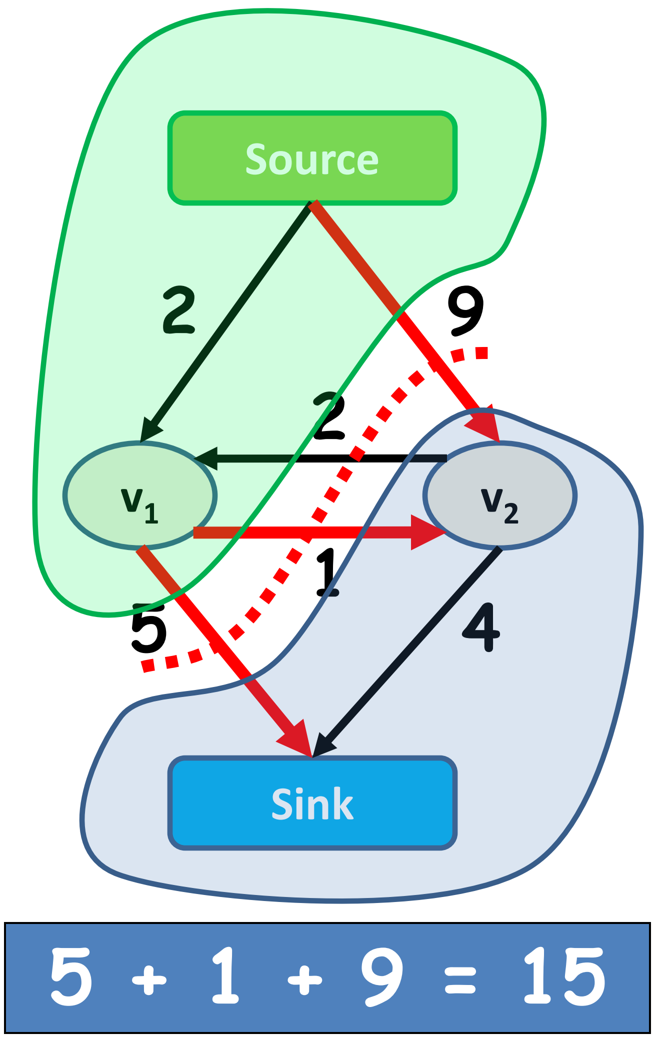
Costs C = $\{c_{(1,2)} \ldots\}$

The st-Mincut Problem

What is a st-cut?



The st-Mincut Problem



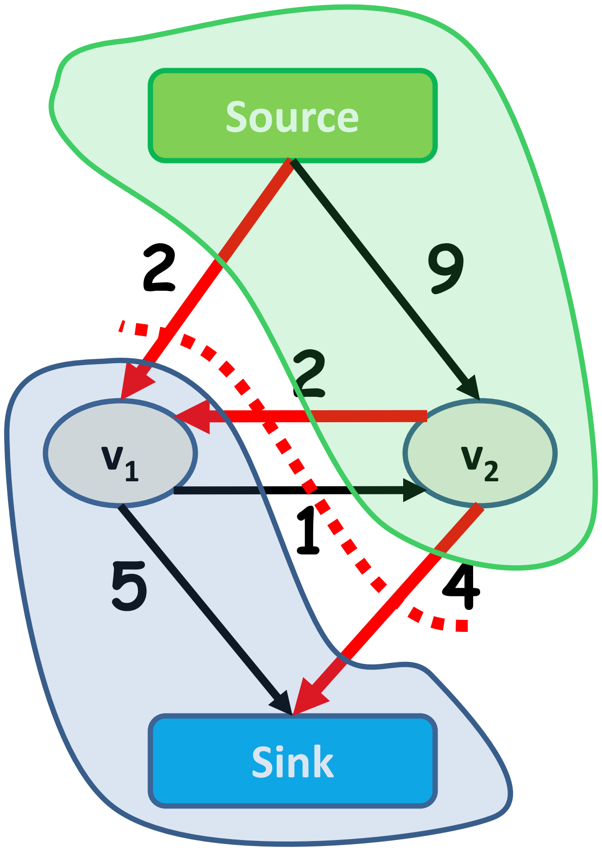
What is a st-cut?

An st-cut (**S**,**T**) divides the nodes between source and sink.

What is the cost of a st-cut?

Sum of cost of all edges going from S to T

The st-Mincut Problem



$2 + 2 + 4 = 8$

What is a st-cut?

An st-cut (**S**,**T**) divides the nodes between source and sink.

What is the cost of a st-cut?

Sum of cost of all edges going from S to T

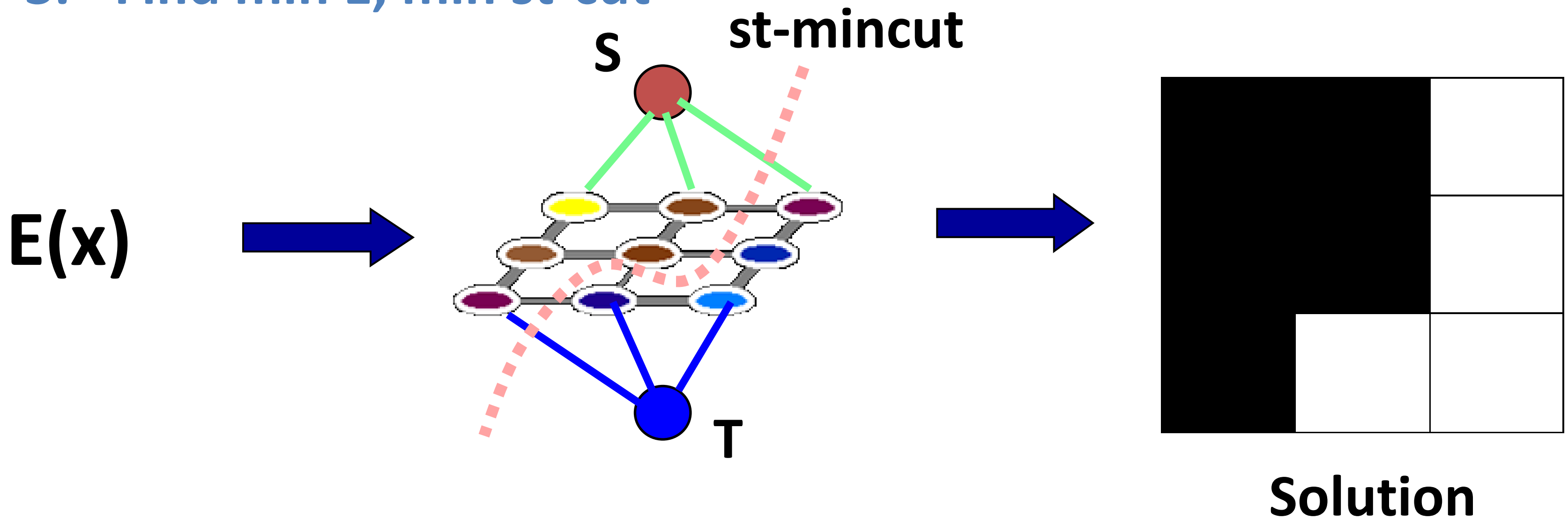
What is the st-mincut?

st-cut with the minimum cost

So how does this work?

Construct a graph such that:

1. **Any st-cut corresponds to an assignment of x**
2. **The cost of the cut is equal to the energy of x : E(x)**
3. **Find min E, min st-cut**



[Hammer, 1965] [Kolmogorov and Zabih, 2002]

st-mincut and Energy Minimization

$$E(x) = \sum_i \theta_i(x_i) + \sum_{i,j} \theta_{ij}(x_i, x_j)$$

For all ij $\quad \theta_{ij}(0,1) + \theta_{ij}(1,0) \geq \theta_{ij}(0,0) + \theta_{ij}(1,1)$

Equivalent (transform to "normal form")

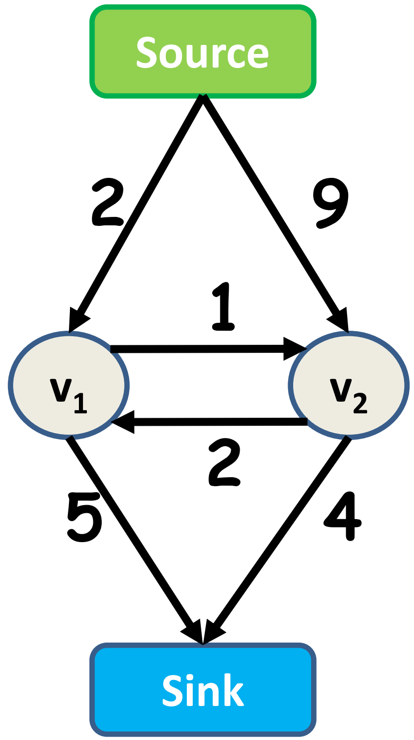
$$E(x) = \sum_i c_i x_i + c'_i(1-x_i) + \sum_{i,j} c_{ij}\, x_i(1-x_j)$$

$c_i, c'_i \in \{0,p\}$ with $p \geq 0$

$c_{ij} \geq 0$

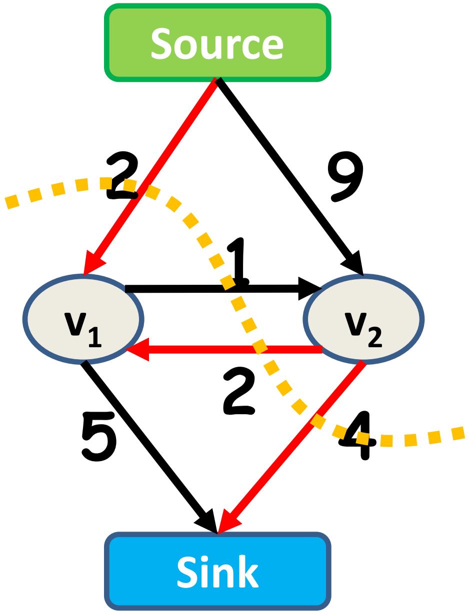
[Kolmogorov and Rother '07]

Example



$$E(v_1,v_2) = 2v_1 + 5(1-v_1) + 9v_2 + 4(1-v_2) + 2v_1(1-v_2) + (1-v_1)v_2$$

Example



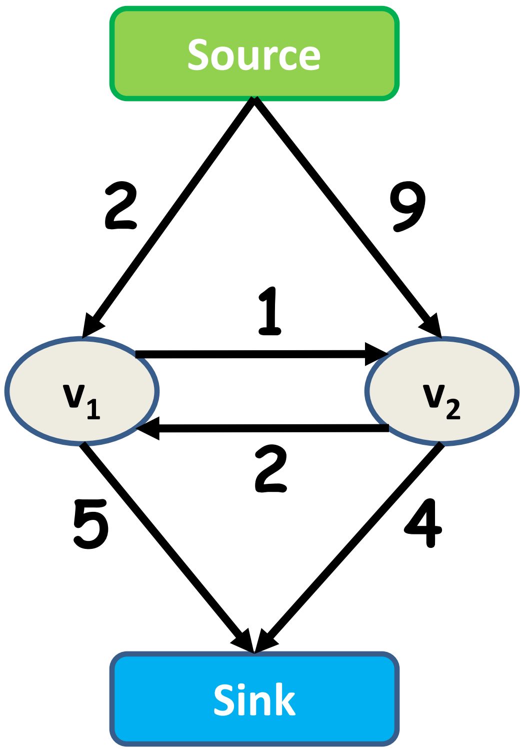
optimal st-mincut: 8

$v_1 = 1 \quad v_2 = 0$

$E(1,0) = 8$

$$E(v_1,v_2) = 2v_1 + 5(1-v_1) + 9v_2 + 4(1-v_2) + 2v_1(1-v_2) + (1-v_1)v_2$$

How to compute the st-mincut?



Min-cut\Max-flow Theorem

In every network, the maximum flow equals the cost of the st-mincut

Solve the **maximum flow** problem

Compute the maximum flow between Source and Sink s.t.

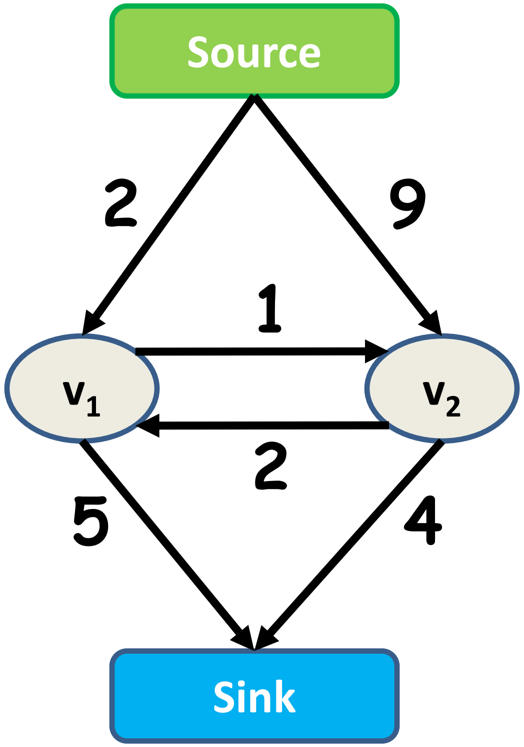
Edges: Flow < Capacity

Nodes: Flow in = Flow out

Assuming non-negative capacity

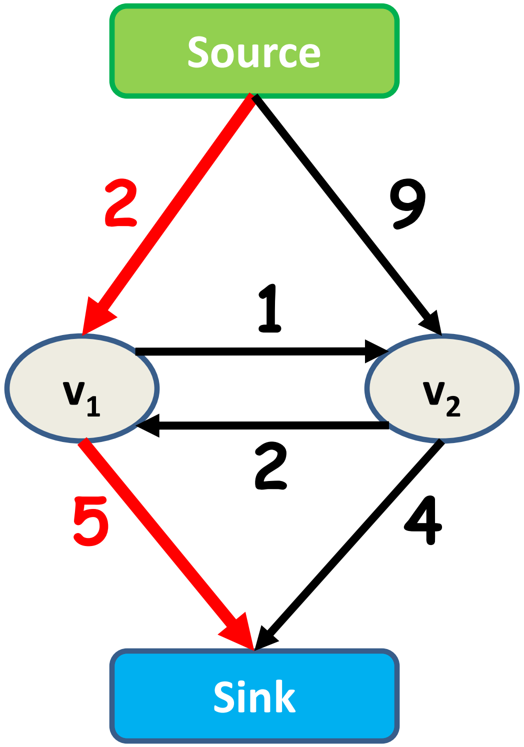
Augmenting Path Based Algorithms

Flow = 0



Augmenting Path Based Algorithms

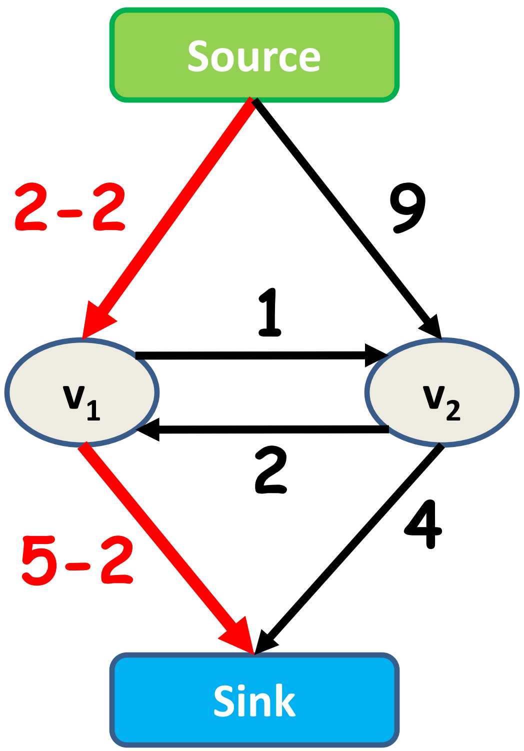
Flow = 0



1. Find path from source to sink with positive capacity

Augmenting Path Based Algorithms

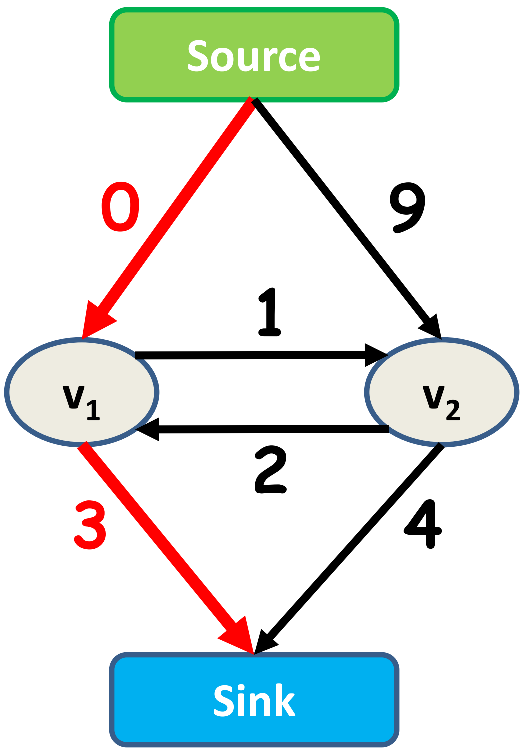
Flow = 0 + 2



1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path

Augmenting Path Based Algorithms

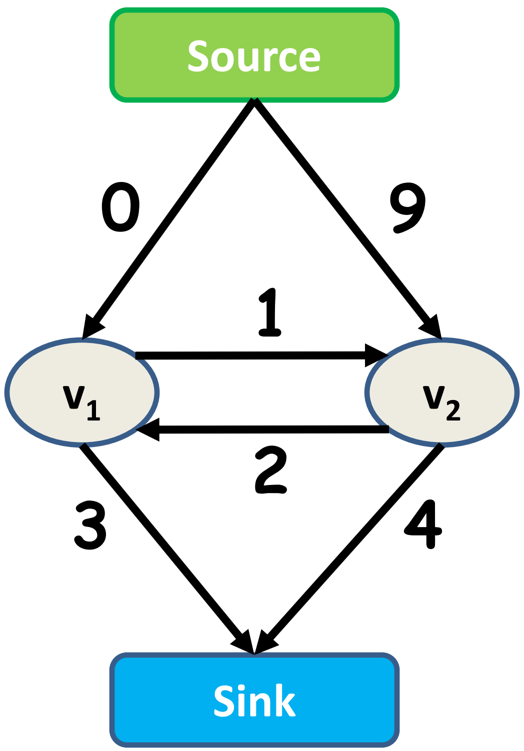
Flow = 2



1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path

Augmenting Path Based Algorithms

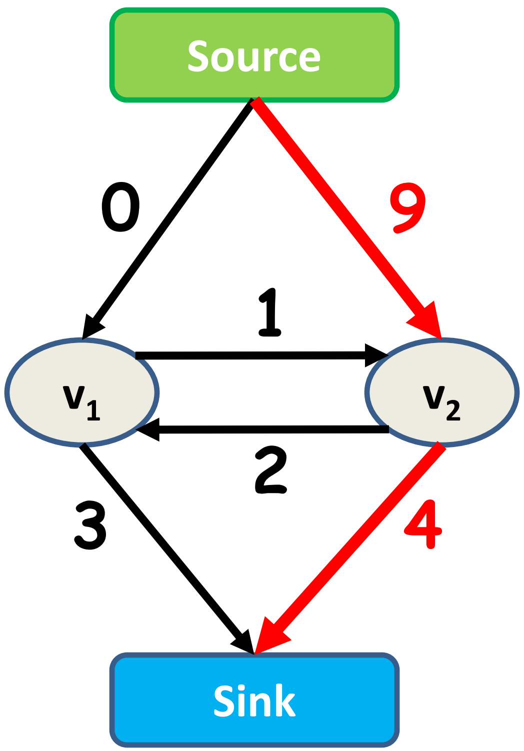
Flow = 2



1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Augmenting Path Based Algorithms

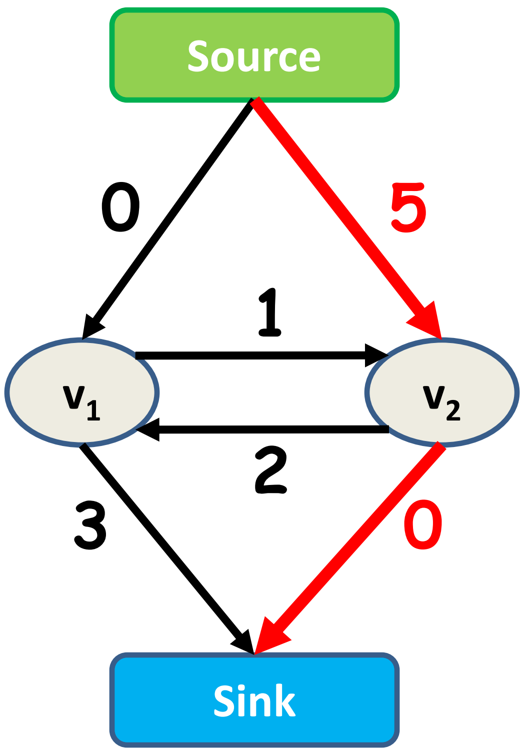
Flow = 2



1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Augmenting Path Based Algorithms

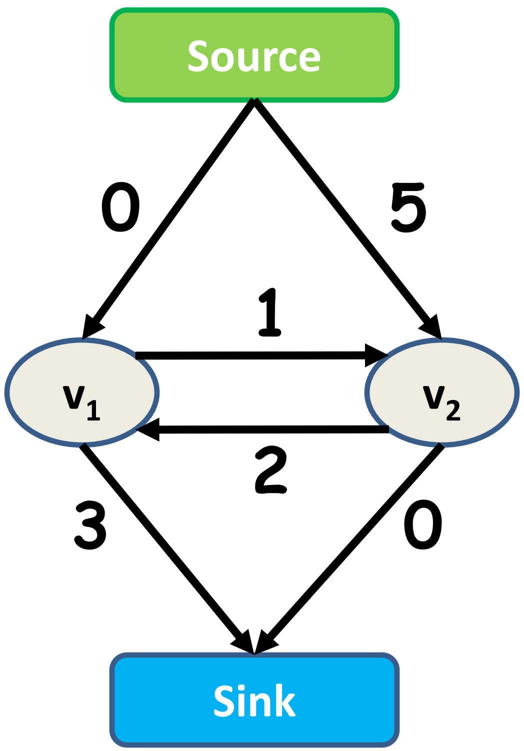
Flow = 2 + 4



1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Augmenting Path Based Algorithms

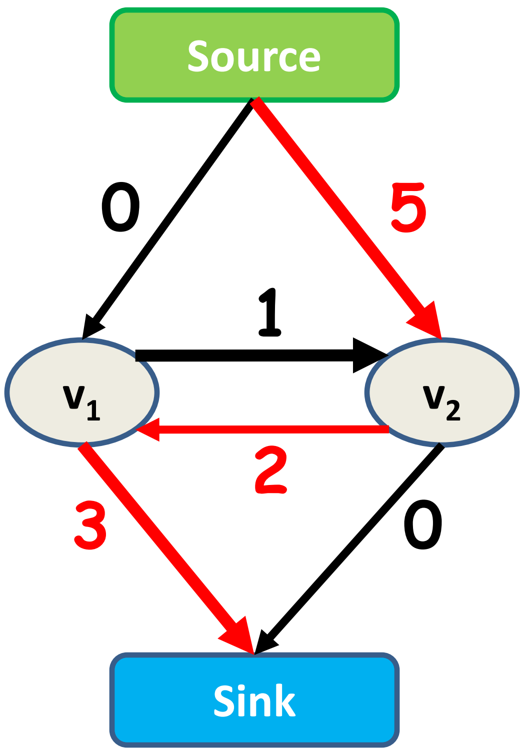
Flow = 6



1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Augmenting Path Based Algorithms

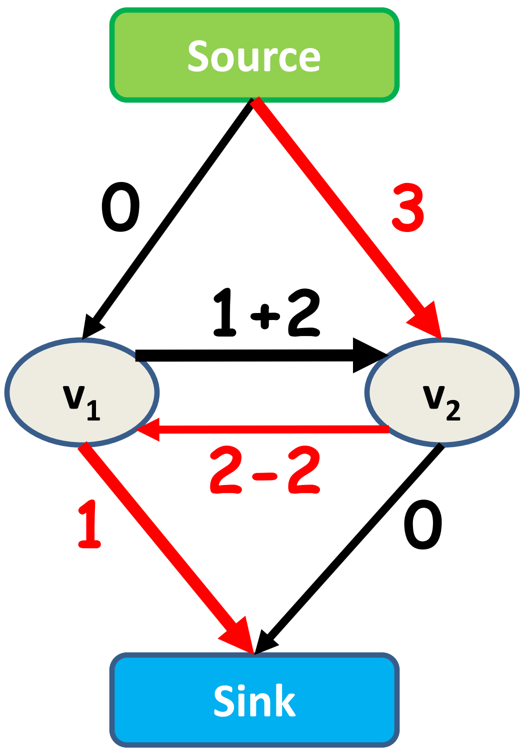
Flow = 6



1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Augmenting Path Based Algorithms

Flow = 6 + 2



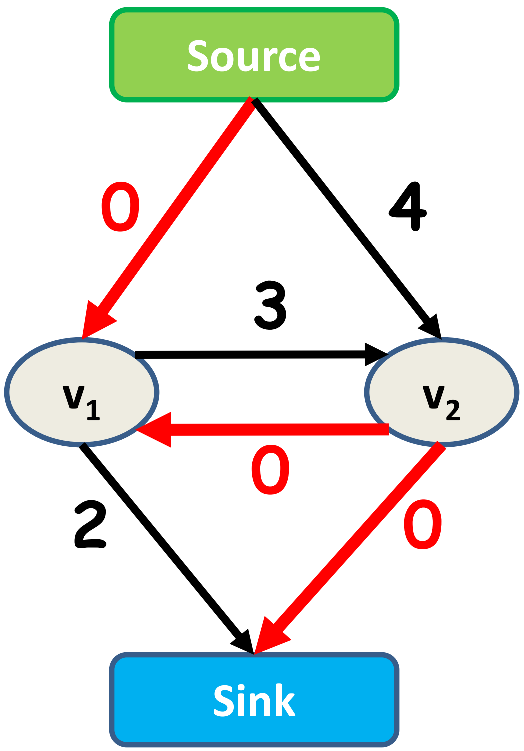
1. Find path from source to sink with positive capacity

2. Push maximum possible flow through this path

3. Repeat until no path can be found

Augmenting Path Based Algorithms

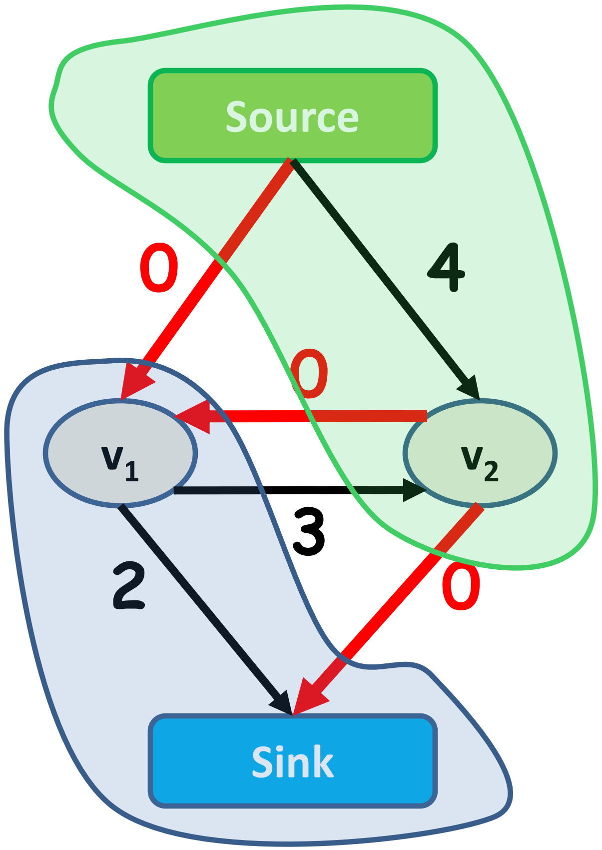
Flow = 8



1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Augmenting Path Based Algorithms

Flow = 8



1. Find path from source to sink with positive capacity

2. Push maximum possible flow through this path

3. Repeat until no path can be found

Saturated edges give the minimum cut. Also flow is min E.

History of Maxflow Algorithms

Augmenting Path and Push-Relabel

year	discoverer(s)	bound
1951	Dantzig	$O(n^2 mU)$
1955	Ford & Fulkerson	$O(m^2 U)$
1970	Dinitz	$O(n^2 m)$
1972	Edmonds & Karp	$O(m^2 \log U)$
1973	Dinitz	$O(nm \log U)$
1974	Karzanov	$O(n^3)$
1977	Cherkassky	$O(n^2 m^{1/2})$
1980	Galil & Naamad	$O(nm \log^2 n)$
1983	Sleator & Tarjan	$O(nm \log n)$
1986	Goldberg & Tarjan	$O(nm \log(n^2/m))$
1987	Ahuja & Orlin	$O(nm + n^2 \log U)$
1987	Ahuja et al.	$O(nm \log(n\sqrt{\log U}/m))$
1989	Cheriyan & Hagerup	$E(nm + n^2 \log^2 n)$
1990	Cheriyan et al.	$O(n^3 / \log n)$
1990	Alon	$O(nm + n^{8/3} \log n)$
1992	King et al.	$O(nm + n^{2+\epsilon})$
1993	Phillips & Westbrook	$O(nm(\log_{m/n} n + \log^{2+\epsilon} n))$
1994	King et al.	$O(nm \log_{m/(n \log n)} n)$
1997	Goldberg & Rao	$O(m^{3/2} \log(n^2/m) \log U)$ $O(n^{2/3} m \log(n^2/m) \log U)$

n: #nodes

m: #edges

U: maximum edge weight

Computer Vision problems: efficient dual search tree augmenting path algorithm [Boykov and Kolmogorov PAMI 04] O(mn^2|C|) … but fast in practice: 1.5MPixel per sec.

[Slide credit: Andrew Goldberg]

Minimizing Non-Submodular Functions

$$E(x) = \sum_{i} \theta_i (x_i) + \sum_{i,j} \theta_{ij} (x_i, x_j)$$

$\theta_{ij}(0,1) + \theta_{ij}(1,0) < \theta_{ij}(0,0) + \theta_{ij}(1,1)$ for some ij

- **Minimizing general non-submodular functions is NP-hard.**
- **Commonly used method is to solve a relaxation of the problem**

Minimization using Roof-dual Relaxation

$$E(\{x_p\}) = \sum \theta_p(x_p)$$

unary

$$+ \sum \theta_{pq}(x_p, x_q)$$

$$\theta_{pq}(0,0) + \theta_{pq}(1,1) \leq \theta_{pq}(0,1) + \theta_{pq}(1,0)$$

pairwise submodular

$$+ \sum \tilde{\theta}_{pq}(x_p, x_q)$$

$$\tilde{\theta}_{pq}(0,0) + \tilde{\theta}_{pq}(1,1) \geq \tilde{\theta}_{pq}(0,1) + \tilde{\theta}_{pq}(1,0)$$

pairwise nonsubmodular

[Boros, Hammer, Sun ’91; Kolmogorov, Rother ‘07]

Minimization using Roof-dual Relaxation (QPBO, BHS-algorithm)

Double number of variables: $x_p \rightarrow x_p, x_{\bar{p}}$

$$E(\{x_p\}) = \sum \theta_p(x_p) + \sum \theta_{pq}(x_p, x_q) + \sum \tilde{\theta}_{pq}(x_p, x_q)$$

$$E'(\{x_p\},\{x_{\bar{p}}\}) = \sum \frac{\theta_p(x_p) + \theta_p(1-x_{\bar{p}})}{2} + \sum \frac{\theta_{pq}(x_p, x_q) + \theta_{pq}(1-x_{\bar{p}}, 1-x_{\bar{q}})}{2} + \sum \frac{\tilde{\theta}_{pq}(x_p, 1-x_{\bar{q}}) + \tilde{\theta}_{pq}(1-x_{\bar{p}}, x_q)}{2}$$

$$E(\{x_p\}) = E'(\{x_p\},\{x_{\bar{p}}\}) \text{ if } x_{\bar{p}} = 1 - x_p$$

- **E’ is submodular**
- **Ignore constraint and solve anyway**

[Boros, Hammer, Sun ’91; Kolmogorov, Rother ‘07]

Minimization using Roof-dual Relaxation (QPBO, BHS-algorithm)

- Output: original $x_p \in \{0,1,?\}$ (partial optimality)

$$\boxed{x_p = 1 - x_{\bar{p}}} \Longrightarrow \boxed{x_p} \text{ is the optimal label}$$

- Solves the LP relaxation for binary pairwise MRFs
- Extensions possible QPBO-P/I [Rother et al. '07]

Combinatorial Optimization

- Binary, pairwise
  - Solvable problems
  - NP-hard
- Multi-label, pairwise
  - Transformation to binary
  - move-making
- Binary, higher-order
  - Transformation to pairwise
  - Problem decomposition

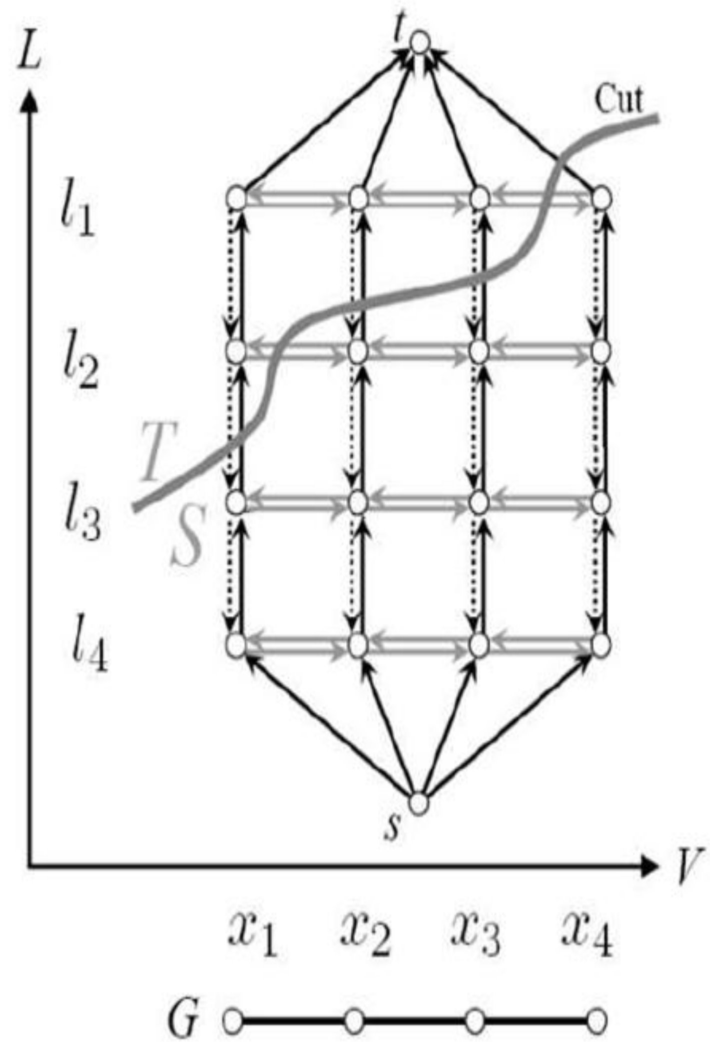
Example: transformation approach

Transform exactly: multi-label to binary

Labels: l_1 …. l_k

variables: x_1 …. x_n

New nodes: n * k



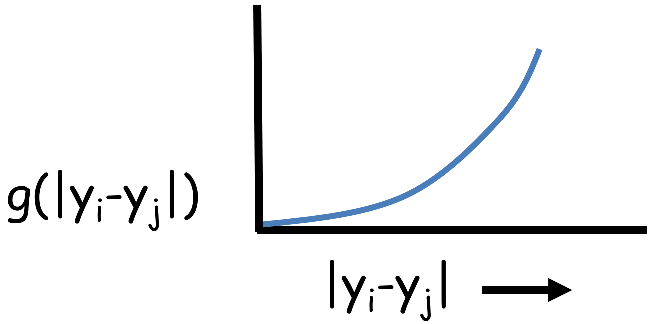
$x_1 = l_3$	$x_2 = l_2$
$x_3 = l_2$	$x_4 = l_1$

[Ishikawa PAMI '03]

Example transformation approach

$$E(\mathbf{y}) = \sum_i \theta_i(y_i) + \sum_{i,j} g(|y_i - y_j|)$$

Exact if g convex:

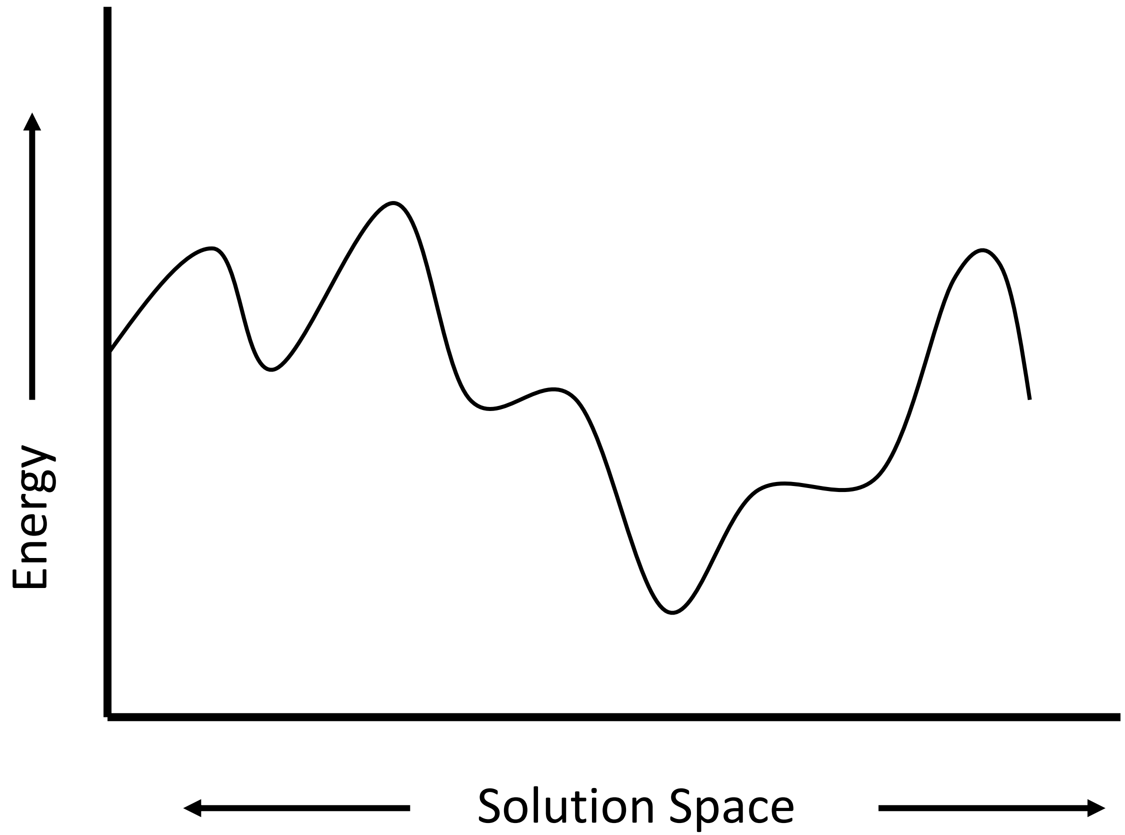


Problem: not discontinuity preserving

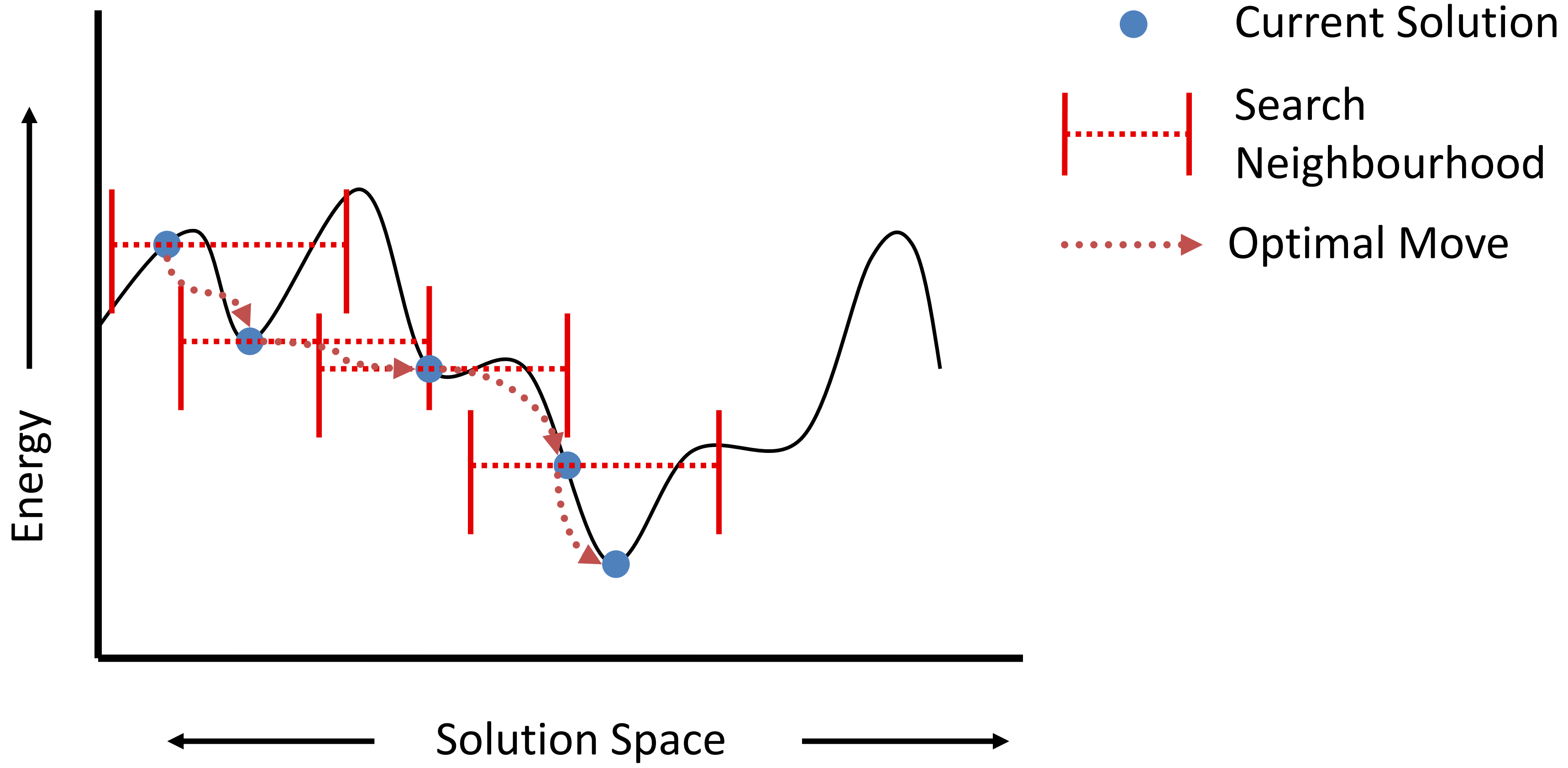
other encoding scheme:

[Roy and Cox '98, Schlesinger & Flach '06]

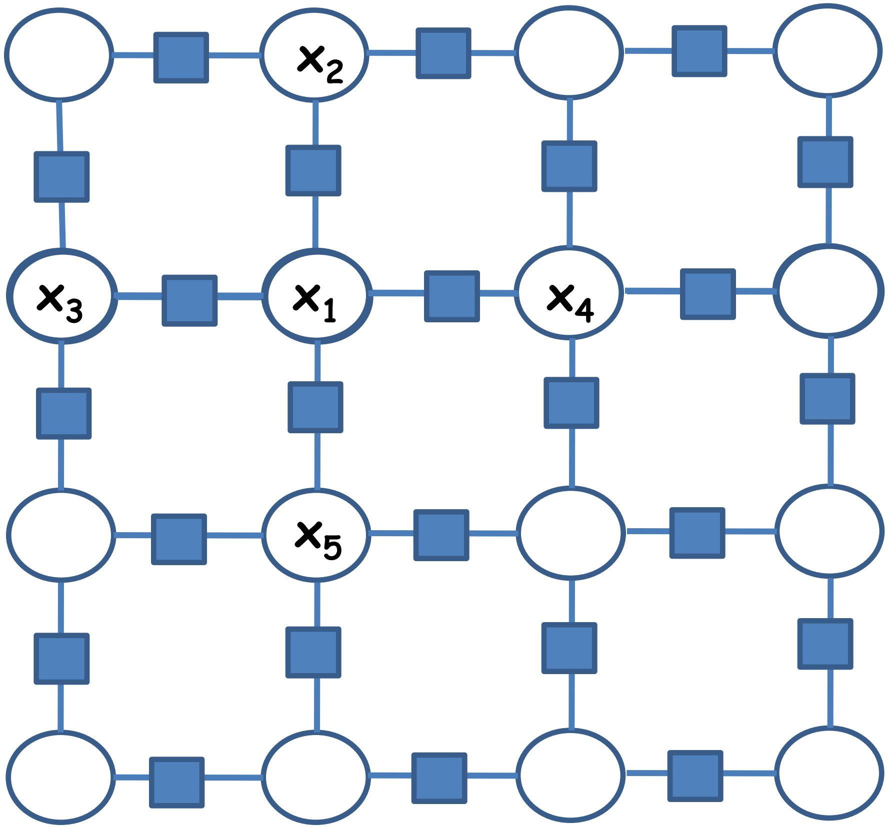
Move Making Algorithms



Move Making Algorithms



Iterative Conditional Mode (ICM)



$$E(x) = \theta_{12}(x_1,x_2) + \theta_{13}(x_1,x_3) + \theta_{14}(x_1,x_4) + \theta_{15}(x_1,x_5) + \ldots$$

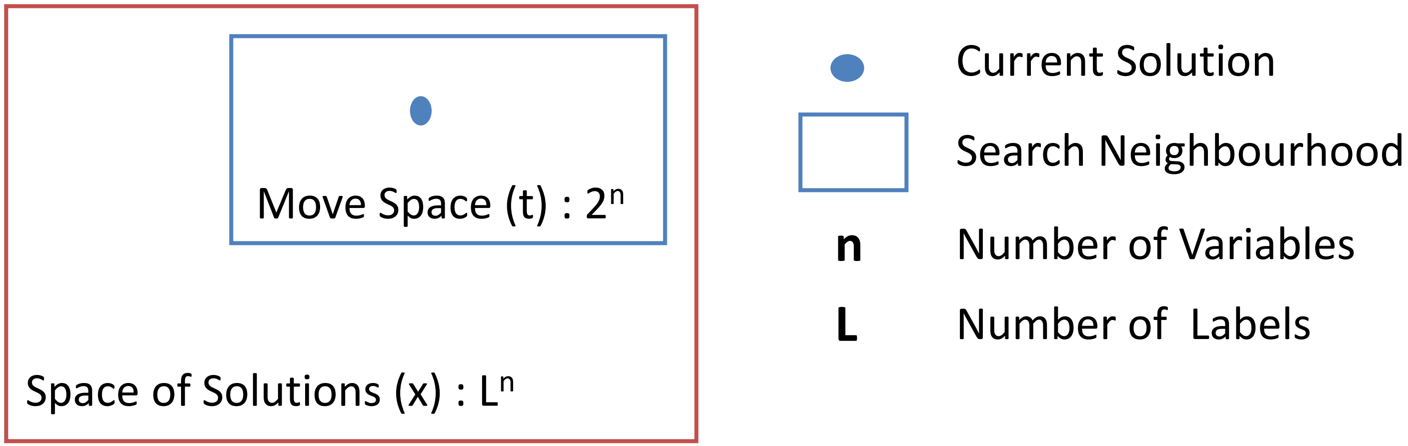
ICM: Very local moves get stuck in local minima

ICM

Global min.

Simulated Annealing: accept move even if energy increases (with certain probability)

Graph Cut-based Move Making Algorithms



A series of globally optimal large moves

[Boykov , Veksler and Zabih 2001]

Expansion Move

- Variables take label α or retain current label

Tree
Ground
House
Sky

Status:

[Boykov , Veksler and Zabih 2001]

Expansion Move

- Move energy is submodular if:
  - Unary Potentials: Arbitrary
  - Pairwise potentials: Metric

$$\theta_{ij}(l_a, l_b) = 0 \text{ iff } l_a = l_b$$

$$\theta_{ij}(l_a, l_b) = \theta_{ij}(l_b, l_a) \geq 0$$

$$\theta_{ij}(l_a, l_b) + \theta_{ij}(l_b, l_c) \geq \theta_{ij}(l_a, l_c)$$

Examples: Potts model, Truncated linear
(not truncated quadratic)

Other moves: alpha-beta swap, range move, etc.

[Boykov , Veksler and Zabih 2001]

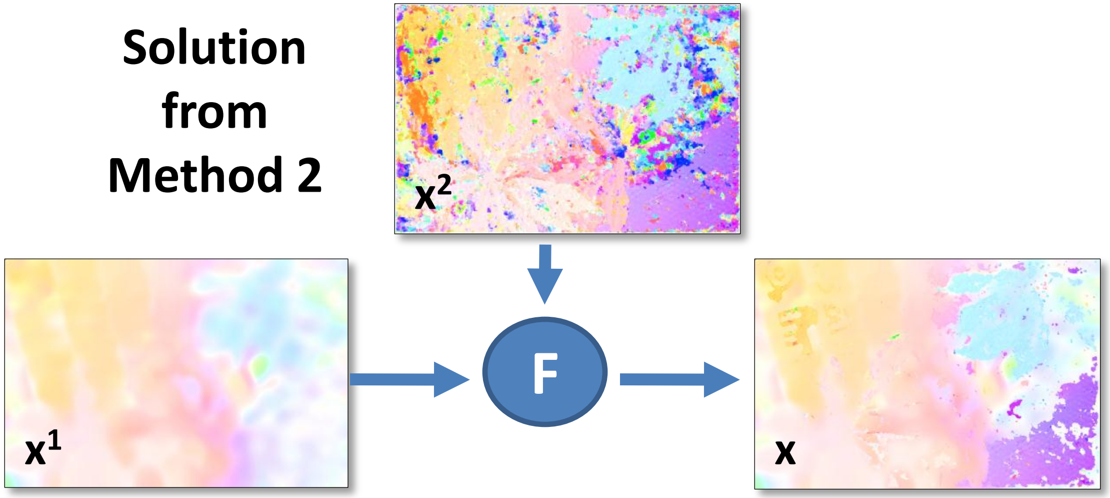
Fusion Move:
Solving Continuous Problems using

$$x = t\, x^1 + (1-t)\, x^2$$

x^1, x^2 can be continuous

Optical Flow Example

Solution from Method 2



Solution from Method 1

Final Solution

[Woodford, Fitzgibbon, Reid, Torr, 2008] [Lempitsky, Rother, Blake, 2008]

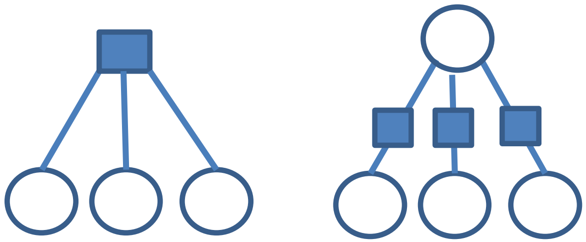
Combinatorial Optimization

- Binary, pairwise
  - Solvable problems
  - NP-hard
- Multi-label, pairwise
  - Transformation to binary
  - move-making
- Binary, higher-order
  - Transformation to pairwise (arbitrary < 7, and special potentials)
  - Problem decomposition

Example: Transformation with factor size 3

$$f(x_1,x_2,x_3) = \theta_{111}x_1x_2x_3 + \theta_{110}x_1x_2(1-x_3) + \theta_{101}x_1(1-x_2)x_3 + \dots$$

$$f(x_1,x_2,x_3) = ax_1x_2x_3 \underbrace{+ bx_1x_2 + cx_2x_3 \dots + 1}_{\text{Quadratic polynomial can be done}}$$



Idea: transform 2+ order terms into 2nd order terms

Many Methods for exact transformation:

Worst case exponential number of auxiliary nodes

(e.g. factor size 5 gives 15 new variables

-see [Ishikawa PAMI ‘09])

Problem: often non-submodular pairwise MRF

Special Potential: Label-Cost Potential

[Hoiem et al. '07, Delong et al. '10, Bleyer et al. '10]

Image

Grabcut-style result

With cost for each new label [Delong et al. '10]

(Same function as [Zhu and Yuille '96])

$\boxed{\text{Label cost} = 10c}$

$\boxed{\text{Label cost} = 4c}$

$$E(x) = \underbrace{P(x)}_{\text{"pairwise MRF"}} + \sum_{l \in L} \underbrace{c_l \, [\exists p : x_p = l\,]}_{\text{"Label cost"}} \qquad E: \{1,\ldots,L\}^n \rightarrow R$$

Transform to pairwise MRF with one extra node (use alpha-expansion)

Basic idea: penalize the complexity of the model
- Minimum description length (MDL)
- Bayesian information criterion (BIC)

[Many more special higher-order potentials in tutorial CVPR '10]

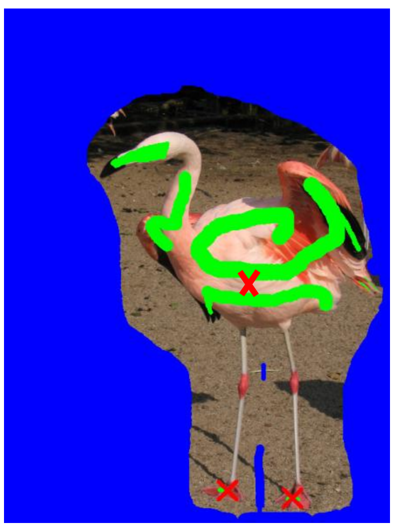
From [Delong et al. '10]

Problem decomposition: Segmentation and Connectivity

Foreground object must be connected:

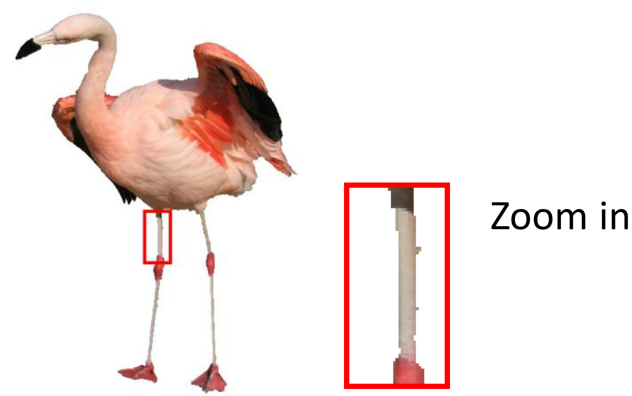
$$E(x) = \sum \theta_i (x_i) + \sum \theta_{ij} (x_i,x_j) + h(x)$$

$$h(x) = \begin{cases} \infty & \text{if } x \text{ not 4-connected} \\ 0 & \text{otherwise} \end{cases}$$



User input

Standard MRF



Standard MRF +**h**

[Vicente et al ’08]

Problem decomposition: Segmentation and Connectivity

$$E(x) = \underbrace{\sum \theta_i (x_i) + \sum \theta_{ij} (x_i,x_j)}_{E_1(x)} + h(x) \qquad h(x) = \begin{cases} \infty & \text{if } x \text{ not 4-connected} \\ 0 & \text{otherwise} \end{cases}$$

Derive Lower bound:

$$\min_x E(x) = \min_x [\, E_1(x) + \theta^T x + h(x) - \theta^T x \,]$$

$$\geq \min_{x_1} [E_1(x_1) + \theta^T x_1] + \min_{x_2} [h(x_2) + \theta^T x_2] = L(\theta)$$

Subproblem 1:

Unary terms + pairwise terms

Global minimum: GraphCut

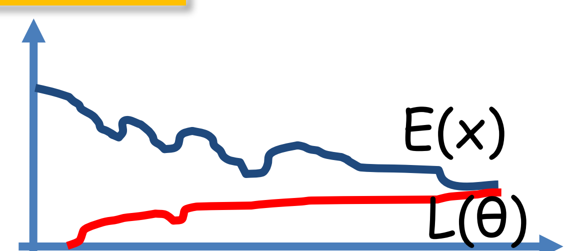
Subproblem 2:

Unary terms + Connectivity constraint

Global minimum: Dijkstra

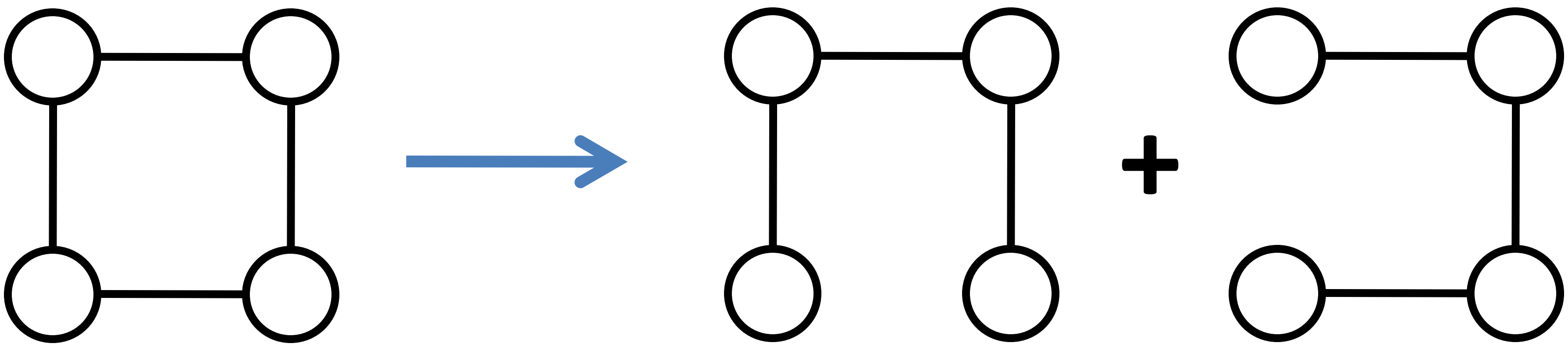
Goal: - maximize concave function $L(\theta)$ using sub-gradient

- no guarantees on E (NP-hard)



Problem decomposition approach:

Tree-reweighted message passing (TRW-S)



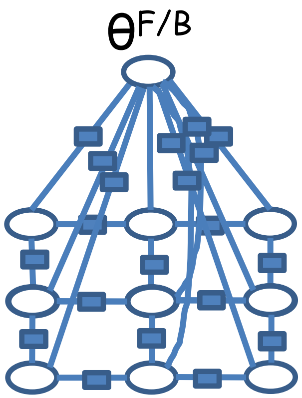
- Each chain provides a global optimum
- Combine these solutions to solve the original problem (different messages update from sub-gradient)
- Try to solve a LP relaxation of the MAP problem

[Kolmogorov, Wainwright et al.; Komodiakis et al '07]

MRF with global potential

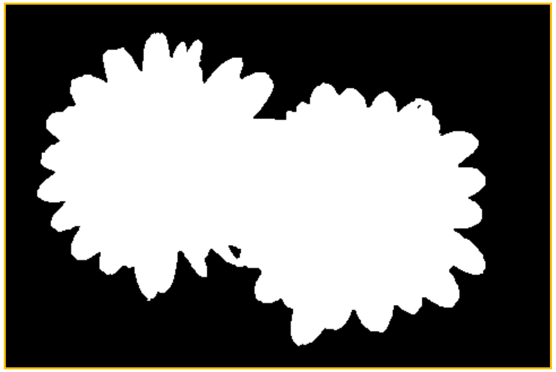
GrabCut model [Rother et. al. '04]

$$E(x,\theta^F,\theta^B) = \sum_i F_i(\theta^F)x_i + B_i(\theta^B)(1-x_i) \quad + \sum_{i,j \in N} |x_i - x_j|$$

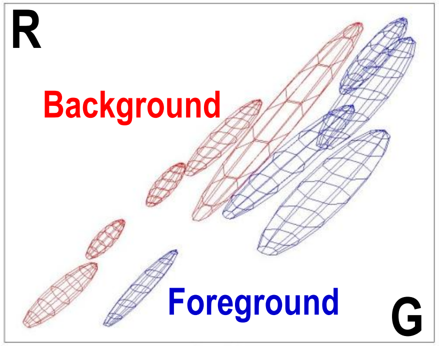


$$F_i = -\log \Pr(z_i|\theta^F) \qquad B_i = -\log \Pr(z_i|\theta^B)$$

Image **z**



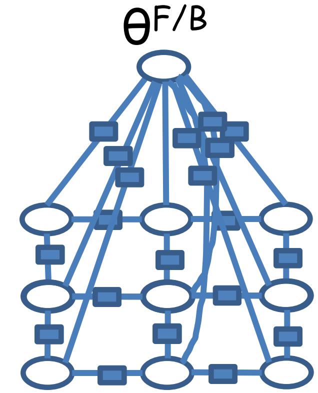
Output x



$\theta^{F/B}$ Gaussian Mixture models

Problem: for unknown x,θ^F,θ^B the optimization is NP-hard! [Vicente et al. '09]

MRF with global potential:
GrabCut - Iterated Graph Cuts



$$\min_{\theta^F,\theta^B} E(x, \theta^F, \theta^B)$$

$$\min_{x} E(x, \theta^F, \theta^B)$$

Learning of the colour distributions

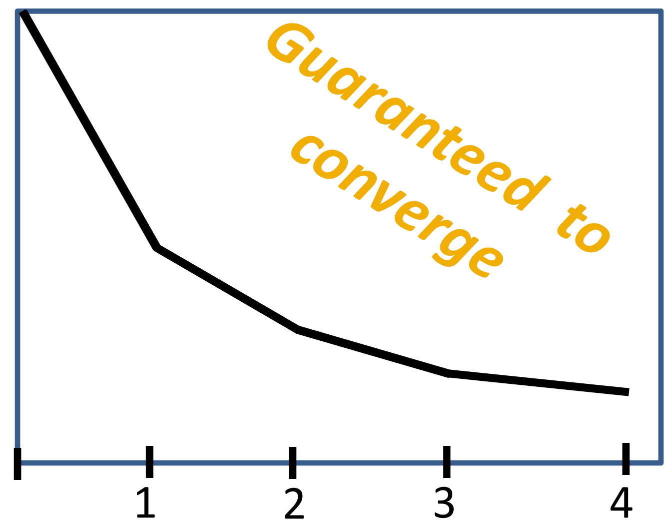
Graph cut to infer segmentation

Most systems with global variables work like that
e.g. [ObjCut Kumar et. al. ‘05, PoseCut Bray et al. ’06, LayoutCRF Winn et al. ’06]

More sophisticated methods: [Lempitsky et al ‘08, Vicente et al ‘09]

MRF with global potential:
GrabCut - Iterated Graph Cuts

Result



Energy after each Iteration

Outline

- Introduction
- MRFs and CRFs in Vision
- Optimisation techniques and Comparison

Comparison papers

- Binary, highly-connected MRFs [Rother et al. ‘07]
- Multi-label, 4-connected MRFs [Szeliski et al. ‘06,‘08]
  all online: http://vision.middlebury.edu/MRF/
- Multi-label, highly-connected MRFs [Kolmogorov et al. ‘06]

Comparison papers

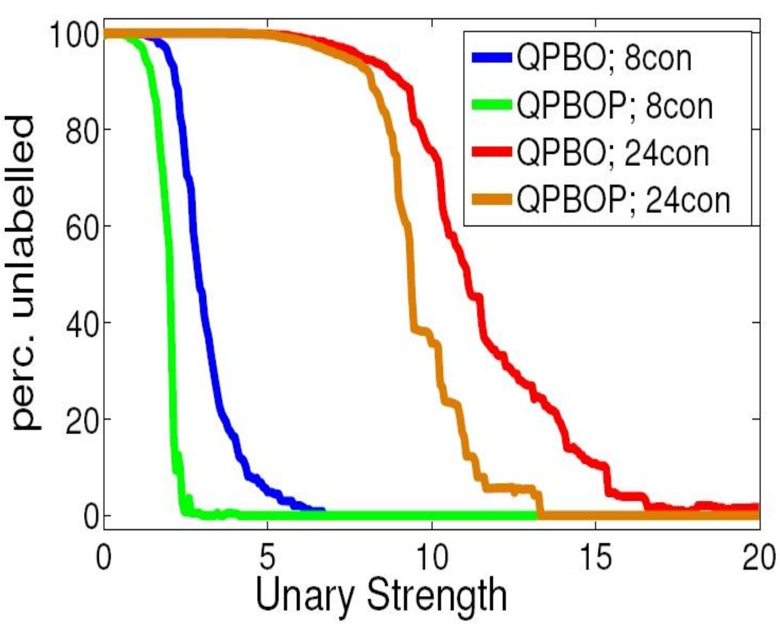
- Binary, highly-connected MRFs [Rother et al. ‘07]

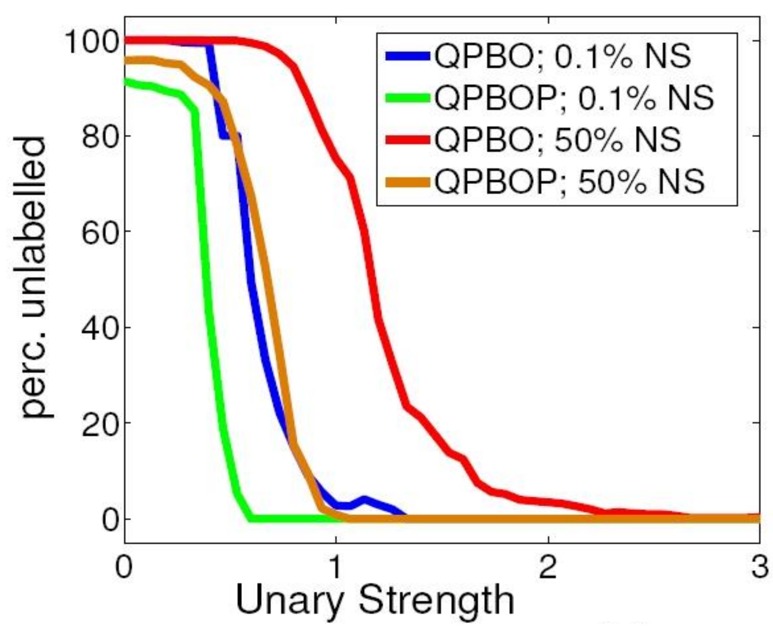
- Multi-label, 4-connected MRFs [Szeliski et al. ‘06,‘08]
  all online: http://vision.middlebury.edu/MRF/

- Multi-label, highly-connected MRFs [Kolmogorov et al. ‘06]

Random MRFs

- Three important factors:
  - Unary strength: $E(x) = w \sum \theta_i (x_i) + \sum \theta_{ij} (x_i, x_j)$
  - Connectivity (av. degree of a node)
  - Percentage of non-submodular terms (NS)





Computer Vision Problems

perc. unlabeled (sec) Energy $\in [0, 999]$ (sec)

Applications	QPBO	QPBOP	P+BP+I	Sim. An.	ICM	GC	BP
Diagram recognition (4.8con)	56.3% (0s)	0% (0s) **GM**	0 (0s)	0 (0.28s)	999 (0s)	119 (0s)	25 (0s)
New View Synthesis (8con)	3.9%(0.7s)	0% (1.4s) **GM**	0 (1.2s)	- (-s)	999 (0.2s)	2 (0.3s)	18 (0.6s)
Super-resolution (8con)	0.5% (0.016s)	0% (0.047s) **GM**	0 (0.03s)	7 (52s)	68 (0.02s)	999 (0s)	0.03 (0.01s)
Image Segm. 9BC + 1 Fgd Pixel (4con)	99.9% (0.08s)	0% (10.5s) **GM**	0 (10.5s)	983 (50s)	999 (0.07s)	0 (28s)	28 (0.2s)
Image Segm. 9BC; 4RC (4con)	1% (1.46s)	0% (1.48s) **GM**	0 (1.48s)	900 (50s)	999 (0.04s)	0 (14s)	24 (0.2s)
Texture restoration (15con)	16.5% (1.4s)	0% (14s) **GM**	0 (14s)	15 (165s)	636 (0.26)	999 (0.05s)	19 (0.18s)
Deconvolution 3×3 kernel (24con)	45% (0.01s)	43% (0.4s)	0 (0.4s)	0 (0.4s)	14 (0s)	999 (0s)	5 (0.5s)
Deconvolution 5×5 kernel (80con)	80% (0.1s)	80% (9s)	8.1 (31s)	0 (1.3s)	6 (0.03s)	999 (0s)	71 (0.9s)

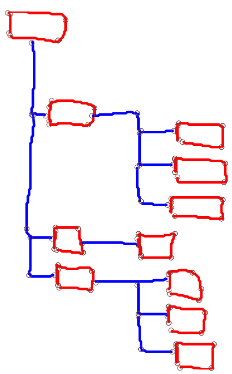
Conclusions:

- Connectivity is a crucial factor
- Simple methods like Simulated Annealing sometimes best

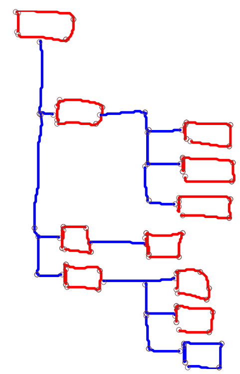
Diagram Recognition **[Szummer et al ‘04]**

71 nodes; 4.8 con.; 28% non-sub; 0.5 unary strength

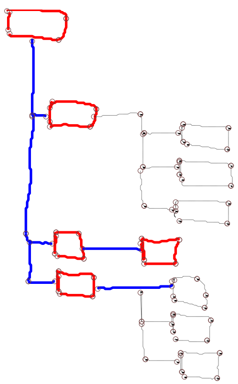
- **2700** test cases: QPBO solved nearly all (QPBOP solves all)



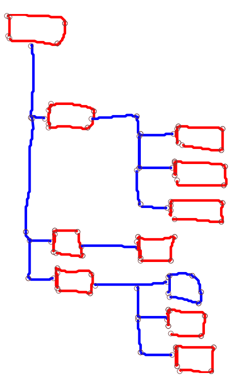
Ground truth



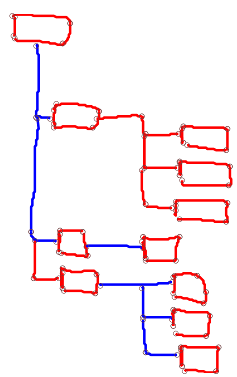
QPBOP (0sec) - Global Min.
Sim. Ann. E=0 (0.28sec)



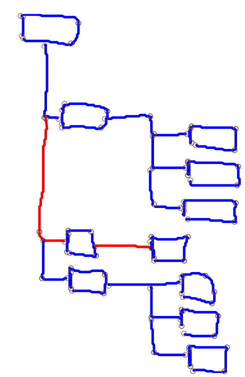
QPBO: 56.3% unlabeled (0 sec)



BP E=25 (0 sec)



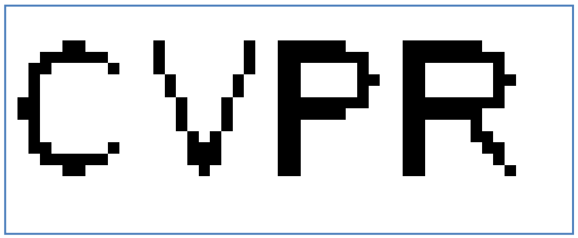
GrapCut E= 119 (0 sec)



ICM E=999 (0 sec)

Binary Image Deconvolution

50x20 nodes; 80con; 100% non-sub; 109 unary strength

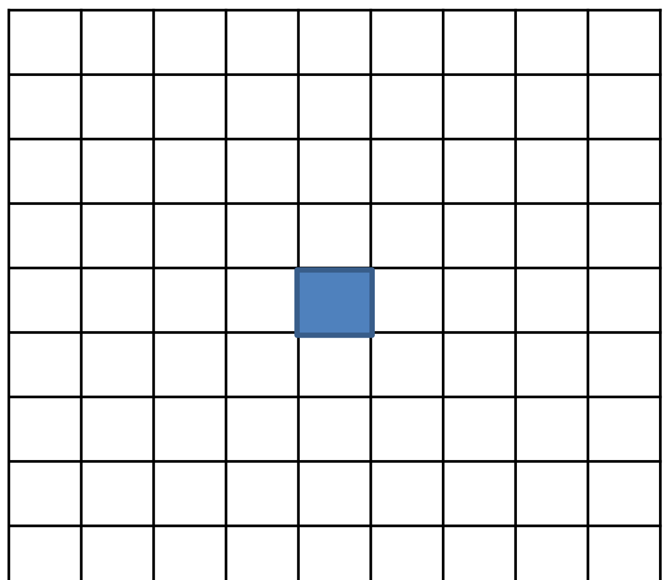


Ground Truth

Input

0.2	0.2	0.2	0.2	0.2
0.2	0.2	0.2	0.2	0.2
0.2	0.2	0.2	0.2	0.2
0.2	0.2	0.2	0.2	0.2
0.2	0.2	0.2	0.2	0.2

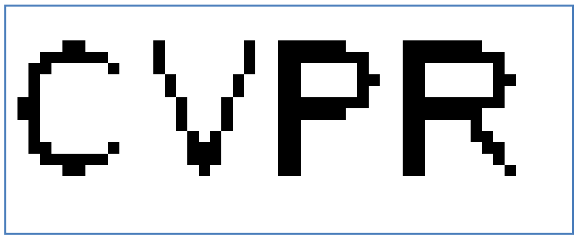
5x5 blur kernel



MRF: 80 connectivity - illustration

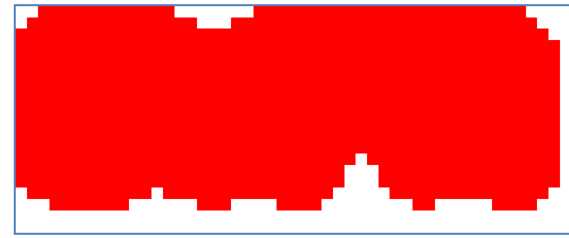
Binary Image Deconvolution

50x20 nodes; 80con; 100% non-sub; 109 unary strength

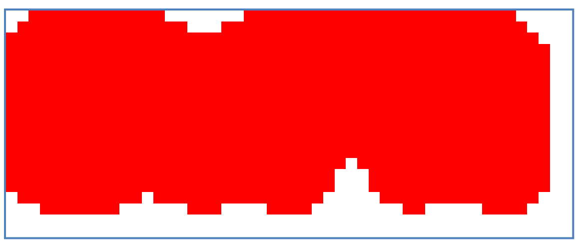


Ground Truth

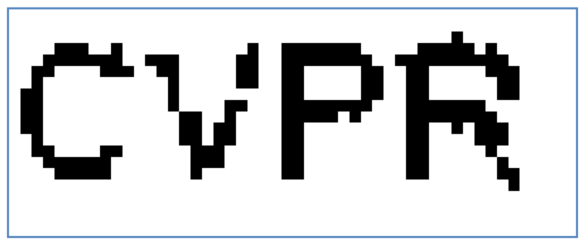
Input



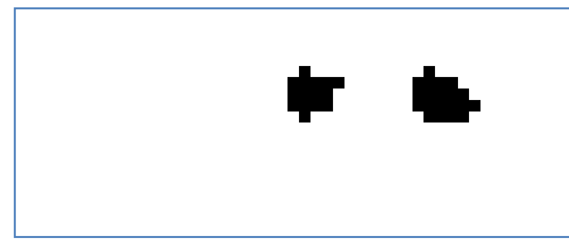
QPBO 80% unlab. (0.1sec)



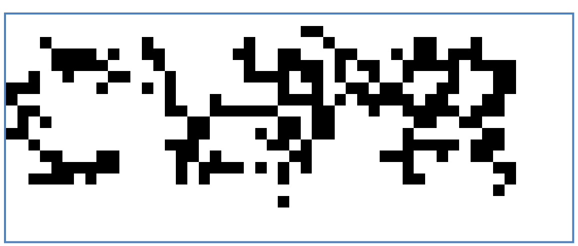
QPBOP 80% unlab. (0.9sec)



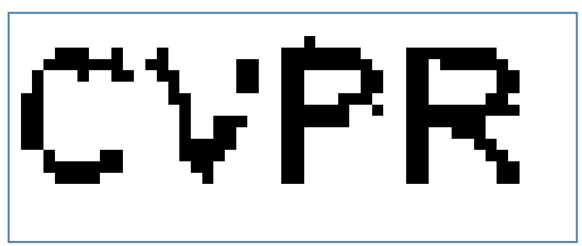
ICM E=6 (0.03sec)



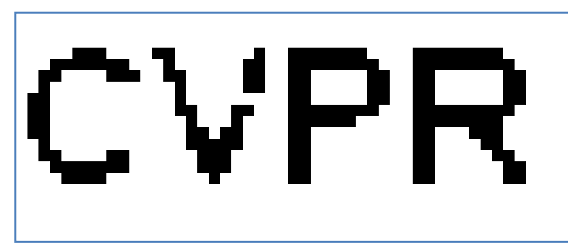
GC E=999 (0sec)



BP E=71 (0.9sec)



QPBOP+BP+I, E=8.1 (31sec)



Sim. Ann. E=0 (1.3sec)

Comparison papers

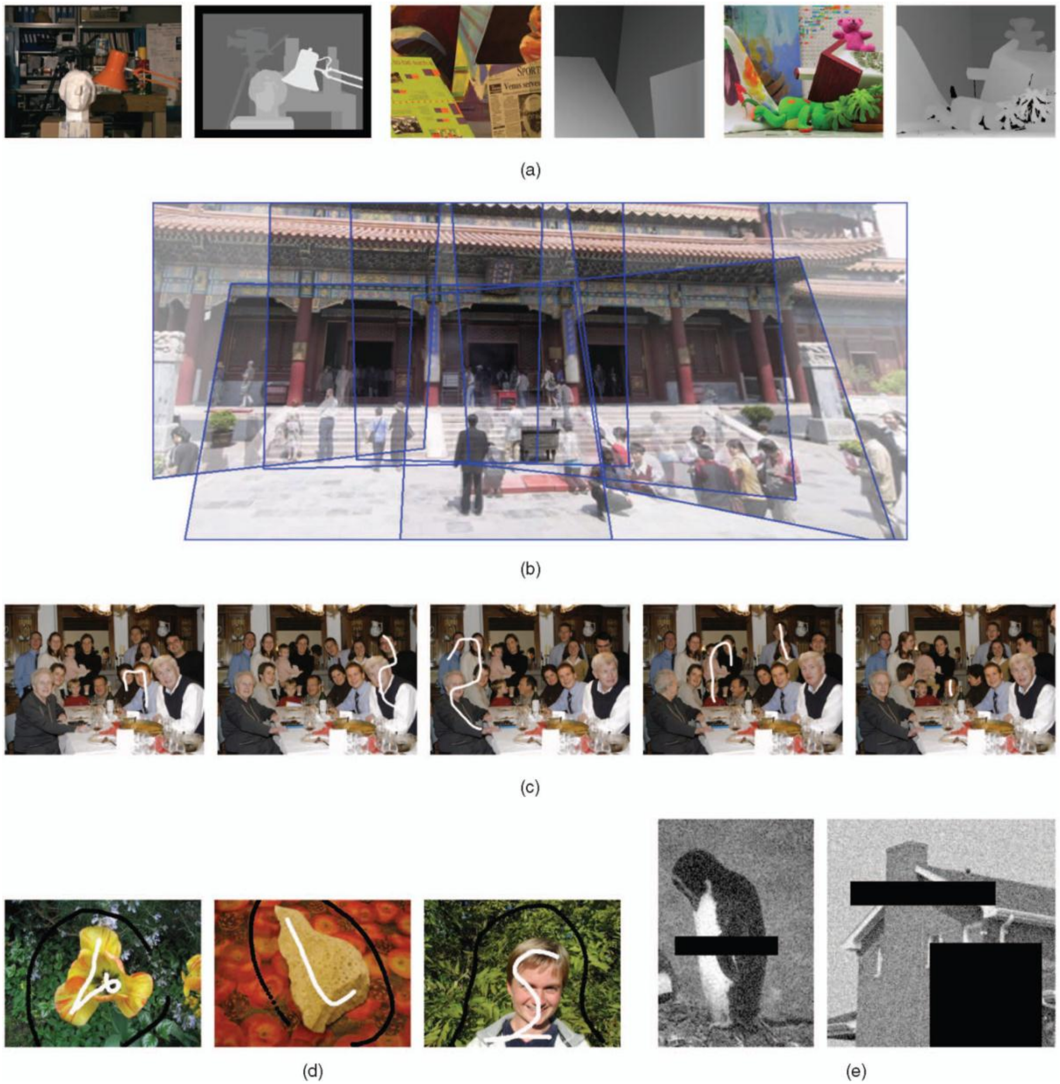
- Binary, highly-connected MRFs [Rother et al. ‘07]
  Conclusion: low-connectivity tractable: QPBO(P)

- Multi-label, 4-connected MRFs [Szeliski et al ‘06,‘08]
  all online: http://vision.middlebury.edu/MRF/

- Multi-label, highly-connected MRFs [Kolmogorov et al ‘06]

Multiple labels – 4 connected

"Attractive Potentials"



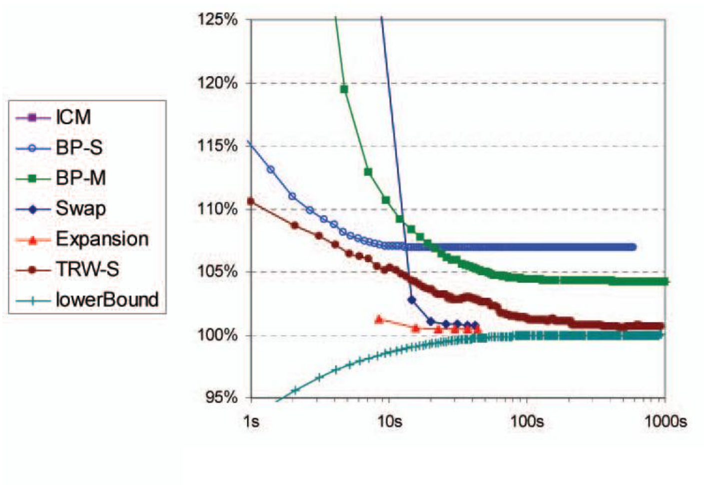
stereo

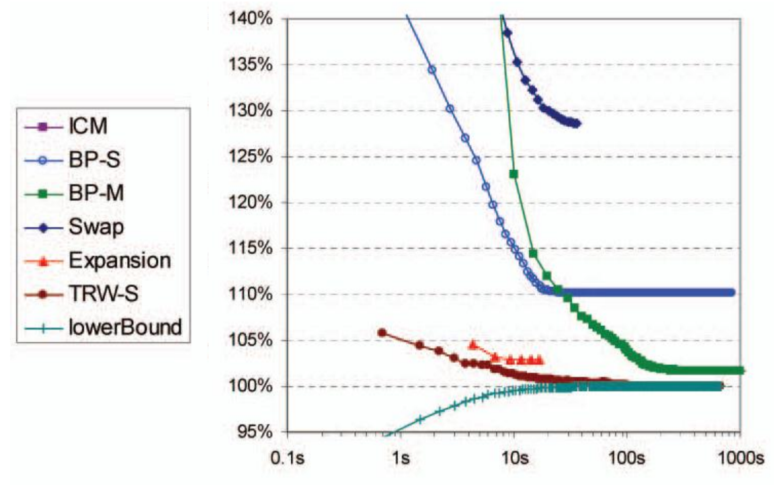
Panoramic stitching

Image Segmentation; de-noising; in-painting

[Szelsiki et al '06,08]

Stereo





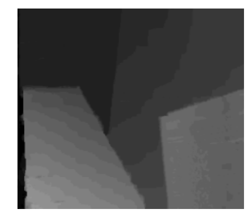

image


Ground truth


TRW-S


image


Ground truth

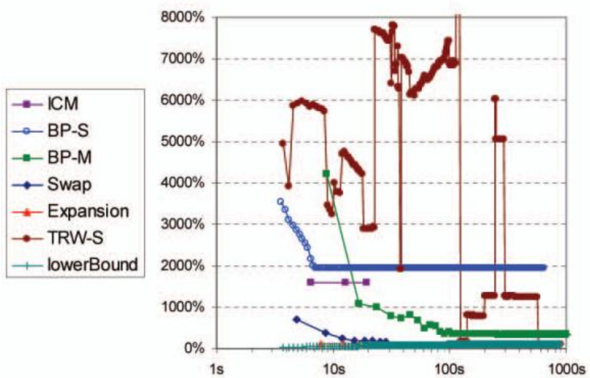

TRW-S

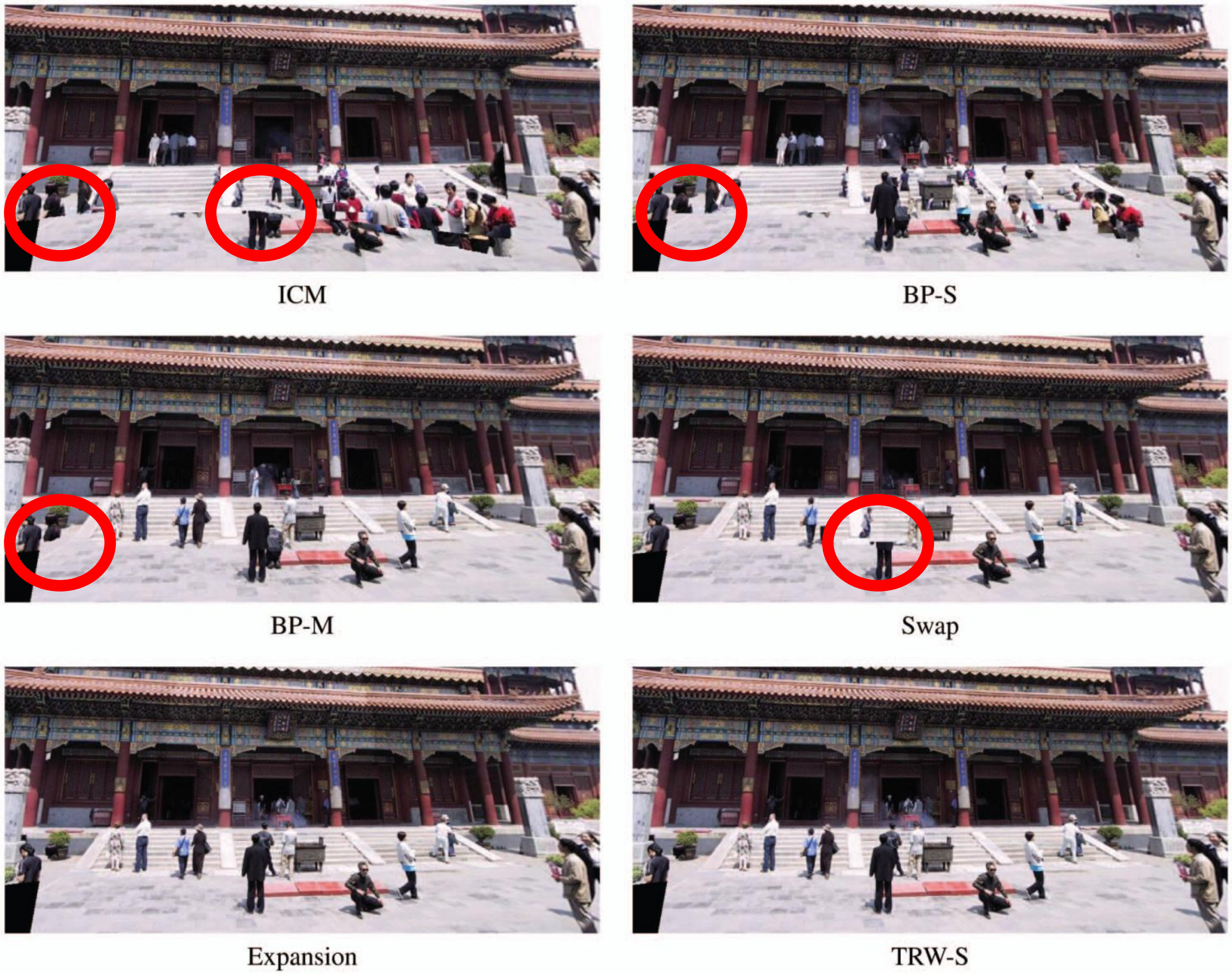
Conclusions:

- Solved by alpha-exp. and TRW-S
  (within 0.01%-0.9% of lower bound – true for all tests!)

Panoramic stitching

- Unordered labels are (slightly) more challenging





Comparison papers

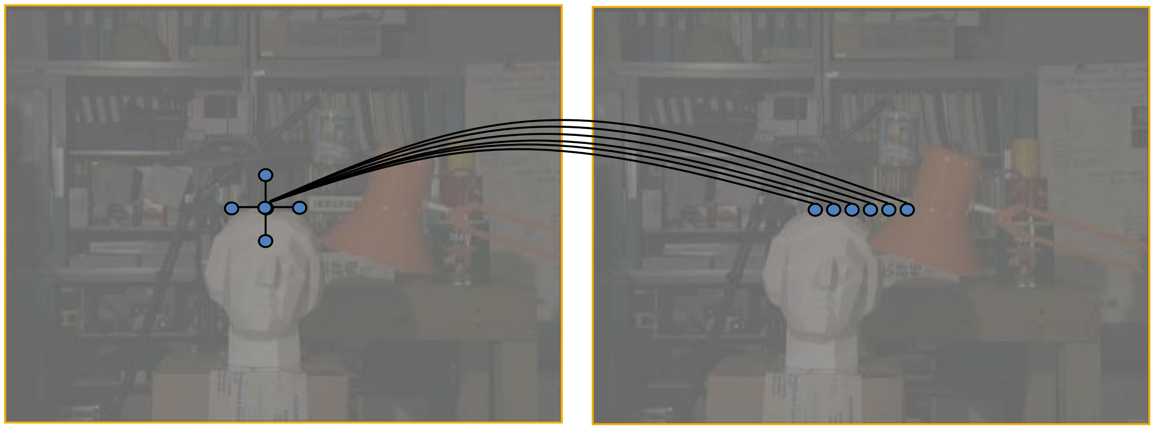
- Binary, highly-connected MRFs [Rother et al. ‘07]
  Conclusion: low-connectivity tractable (QPBO)

- Multi-label, 4-connected MRFs [Szeliski et al ‘06,‘08]
  all online: http://vision.middlebury.edu/MRF/
  Conclusion: solved by expansion-move; TRW-S
  (within 0.01 - 0.9% of lower bound)

- Multi-label, highly-connected MRFs [Kolmogorov et al ‘06]

Multiple labels – highly connected

Stereo with occlusion:



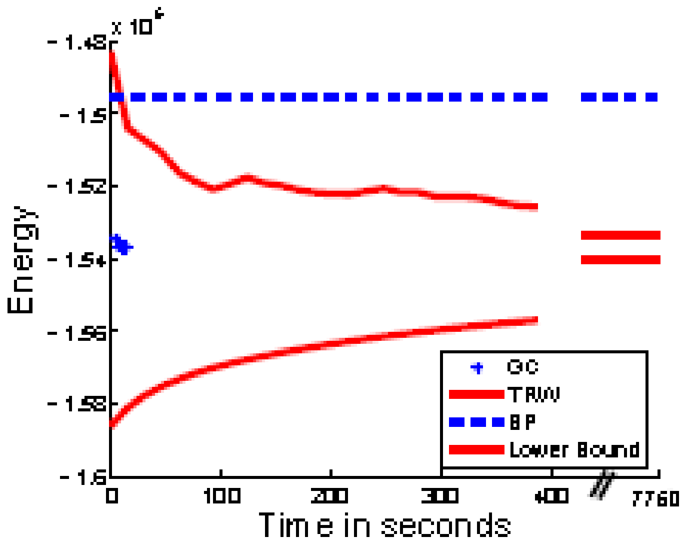
E(d): $\{1,\dots,D\}^{2n} \rightarrow R$

Each pixel is connected to **D** pixels in the other image

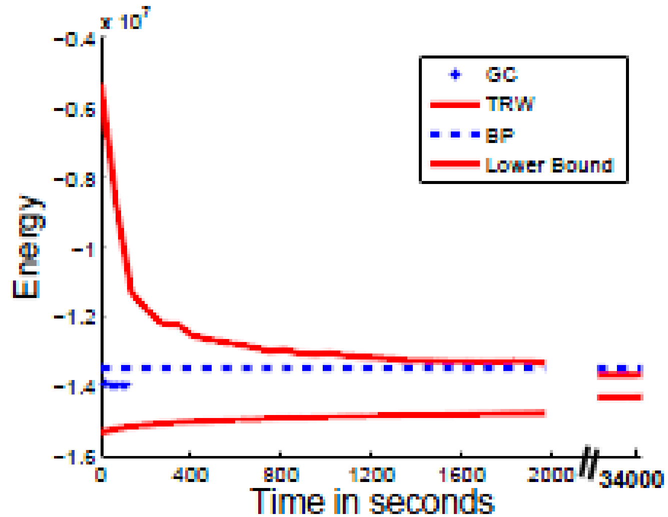
[Kolmogorov et al. '06]

Multiple labels – highly connected

Tsukuba: 16 labels

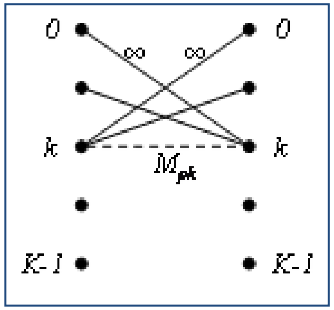


Cones: 56 labels



- Alpha-exp. considerably better than message passing

Potential reason: smaller connectivity in one expansion-move



Comparison papers

- binary, highly-connected MRFs [Rother et al. ‘07]
  Conclusion: low-connectivity tractable (QPBO)

- Multi-label, 4-connected MRFs [Szeliski et al ‘06,‘08]
  all online: http://vision.middlebury.edu/MRF/
  Conclusion: solved by alpha-exp.; TRW
  (within 0.9% to lower bound)

- Multi-label, highly-connected MRFs [Kolmogorov et al ‘06]
  Conclusion: challenging optimization (alpha-exp. best)

How to efficiently optimize general highly-connected (higher-order) MRFs is still an open question

Forthcoming book!

Advances in Markov Random Fields for Computer Vision
(Blake, Kohli, Rother)

- MIT Press (Spring 2011)
- Most topics of this tutorial and much, much more
- Contributors: usual suspects: Editors + Boykov, Kolmogorov, Weiss, Freeman, Komodiakis, ...

Other sources of references:
Tutorials at recent conferences: CVPR ‘10, ICCV 09, ECCV ’08, ICCV ‘07, etc.

IMPORTANT

Tea break!

unused slides

What is the LP relaxation approach?

- Write MAP as Integer Program (IP)
- Relax to Liner Program (LP relaxation)
- Solve LP (polynomial time algorithms)
- Round LP to get best IP solution (no guarantees)

MAP Inference as an IP

$$\min \left[\sum_{a \in L} V_p(a) x_{p,a} \qquad + \sum_{a,b \in L} V_{pq}(a,b) x_{pq,ab} \qquad \right]$$

$$\begin{aligned} \text{s.t.} \sum\nolimits_{a \in L} x_{p,a} &= 1 \\ \sum\nolimits_{a \in L} x_{pq,ab} &= x_{q,b} \\ \sum\nolimits_{b \in L} x_{pq,ab} &= x_{p,a} \\ x_{p,a},\ x_{pq,ab} &\in \{0,1\} \end{aligned}$$

Integer Program

Relax to LP

$$\min \left[\sum_{a \in L} V_p(a) x_{p,a} \qquad + \sum_{a,b \in L} V_{pq}(a,b) x_{pq,ab} \qquad \right]$$

$$\begin{aligned} \text{s.t.} \sum\nolimits_{a \in L} x_{p,a} &= 1 \\ \sum\nolimits_{a \in L} x_{pq,ab} &= x_{q,b} \\ \sum\nolimits_{b \in L} x_{pq,ab} &= x_{p,a} \\ x_{p,a} \geq 0,\ x_{pq,ab} &\geq 0 \end{aligned}$$

Linear Program

- **Solve it**: Simplex, Interior Point methods, Message Passing, QPBO, etc.
- **Round** continuous solution