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

Goal

Given $z$ and unknown (latent) variables $x$:

$$ P(x|z) = \frac{P(z|x) P(x)}{P(z)} \sim P(z|x) P(x) $$

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

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

Posterior Probability

 Likelihood (data-dependent)

 Prior (data-independent)

Maximum a Posteriori (MAP): $x^* = \arg\max_x P(x|z)$
Likelihood $P(x|z) \sim P(z|x) \ P(x)$
Likelihood\[P(x|z) \sim P(z|x) \cdot P(x)\]

Maximum likelihood:
\[x^* = \arg\max_x P(z|x) = \arg\max_x \prod_{x_i} P(z_i|x_i)\]
Prior $P(x|z) \sim P(z|x)$

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

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

“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: \( P(x) = \frac{1}{f} \prod_{i,j \in N_4} \exp\{-|x_i - x_j|\} \)

Faire Samples

Solutions with highest probability (mode)

\[ P(x) = 0.011 \]
\[ P(x) = 0.012 \]
\[ P(x) = 0.012 \]

Smoothness prior needs the likelihood
Posterior distribution

\[ P(x|z) \sim P(z|x) \ P(x) \]

“Gibbs” distribution:

\[ P(x|z) = \frac{1}{f(z,w)} \ exp\{-E(x,z,w)\} \]

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

\[ \theta_i (x_i, z_i) = -\log P(z_i|x_i=1) \ x_i - \log P(z_i|x_i=0) \ (1-x_i) \]

\[ \theta_{ij} (x_i, x_j) = |x_i - x_j| \]

"Gibbs" distribution:

- Likelihood
- Prior
- Energy
- Unary terms
- Pairwise terms
Energy minimization

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

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

\[ x^* = \arg\min_X E(x,z,w) \quad \text{MAP same as minimum Energy} \]

\[ E(x,z,w) = \sum_i \theta_i (x_i, z_i) + w \sum_{i,j \in N} \theta_{ij} (x_i, x_j) \]
Weight prior and likelihood

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

Factor graph

variables are in same factor.

unobserved
**Definition “Order”**

The arity (number of variables) of the largest factor

\[ E(X) = \theta(x_1, x_2, x_3) \theta(x_2, x_4) \theta(x_3, x_4) \theta(x_3, x_5) \]

**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

“Pairwise energy”

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

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

"Pairwise energy"

Order 2

higher(8)-connected; pairwise MRF

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

Order 2

Higher-order RF

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

"higher-order energy"

Order n
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**

- Observed variable
- Unobserved (latent) variable
Segmentation: Conditional Random Field

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

\[ \theta_{ij}(x_i, x_j, z_i, z_j) = |x_i - x_j| \left( -\exp\{-\beta ||z_i - z_j||\} \right) \]

\[ \beta = 2(\text{Mean}(||z_i - z_j||_2))^{-1} \]

**Conditional Random Field (CRF) no pure prior**

**Factor graph**

- Observed variable
- Unobserved (latent) variable

**MRF**

**CRF**
Stereo matching

- Images rectified
- Ignore occlusion for now

Energy:

\[ E(d) : \{0, ..., D-1\}^n \rightarrow \mathbb{R} \]

Labels: \( d \) (depth/shift)
Stereo matching - Energy

Energy:

\[ E(d): \{0, \ldots, D-1\}^n \rightarrow \mathbb{R} \]
\[ E(d) = \sum_i \theta_i(d_i) + \sum_{i,j \in N_4} \theta_{ij}(d_i, d_j) \]

Unary:

\[ \theta_i(d_i) = (l_j - r_{i-d_i}) \]
“SAD; Sum of absolute differences”
(many others possible, NCC, …)

Pairwise:

\[ \theta_{ij}(d_i, d_j) = g(|d_i - d_j|) \]
Stereo matching - prior

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

No truncation (global min.)

Stereo matching - prior

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

- No truncation (global min.)
- With truncation (NP hard optimization)

Discontinuity preserving potentials
[Blake&Zisserman’83,’87]

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:

\[ E: \{0,1\}^n \rightarrow \mathbb{R} \]

\[ E(x) = \sum_{i,j \in \mathbb{N}_4} |x_i - x_j| \left[ |a_i - b_i| + |a_j - b_j| \right] \]

[Kwatra et. al. Siggraph '03]
Video Synthesis

Input

Output

Video (duplicated)
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
\[ E(x) = \sum_{i,j \in N_4} \theta_{ij}(x_i,x_j) \]
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 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

Training images

Test image

Test image (60% Noise)

Result MRF 4-connected (neighbours)

Result MRF 4-connected

Result MRF 9-connected (7 attractive; 2 repulsive)
Reason 2: Discretization artefacts

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

Length of the paths:

<table>
<thead>
<tr>
<th></th>
<th>Eucl.</th>
<th>4-con.</th>
<th>8-con.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.65</td>
<td>6.28</td>
<td>5.08</td>
</tr>
<tr>
<td>8</td>
<td>6.28</td>
<td>6.75</td>
<td></td>
</tr>
</tbody>
</table>

[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

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

Each pixel is connected to \( D \) pixels in the other image

\[ \theta_{lr}(d_l, d_r) = \]

\[ \begin{aligned}
\text{match} \\
\infty \\
\infty \\
1 \\
1 \\
d \\
d_l \\
d_r \\
D \\
D \\
\end{aligned} \]

\[ \text{Left view} \]

\[ \begin{aligned}
\text{right view} \\
\end{aligned} \]

\[ d=1 \text{ (}\infty\text{ cost)} \]

\[ d=10 \text{ (match)} \]

\[ d=20 \text{ (0 cost)} \]
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:

\[
E(x,w): \{0,1\}^n \times \{\text{GMMs}\} \rightarrow \mathbb{R}
\]

\[
E(x,w) = \sum_i \theta_i(x_i,w) + \sum_{i,j \in \mathbb{N}_4} \theta_{ij}(x_i,x_j)
\]

An object is a compact set of colors:

[Rother et al. Siggraph ’04]
Reason 4: Use Non-local parameters: Object recognition & segmentation

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

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

[TextonBoost; Shotton et al, ‘06]
Reason 4: Use Non-local parameters:
Object recognition & segmentation

(a) Class+ location
(b) 69.6%
(c) 70.3%
+ edges
(d) 72.2%
+ 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

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

[LayoutCRF Winn et al. ’06]
Random field models

4-connected; pairwise MRF

\[ E(x) = \sum_{i,j \in \mathbb{N}_4} \theta_{ij}(x_i, x_j) \]

Order 2

“Pairwise energy”

Higher(8)-connected; pairwise MRF

\[ E(x) = \sum_{i,j \in \mathbb{N}_8} \theta_{ij}(x_i, x_j) \]

Order 2

Higher-order RF

\[ E(x) = \sum_{i,j \in \mathbb{N}_4} \theta_{ij}(x_i, x_j) + \theta(x_1, \ldots, x_n) \]

Order n

“higher-order energy”
Why Higher-order Functions?

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

Reasons for higher-order MRFs:

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]
Reason 1: Better Texture Modelling

Higher Order Structure not Preserved

Training images

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 (-)

\[ E(x) = P(x) + h(x) \]

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

$E(x) = \sum_{i,j \in \mathbb{N}_4} \theta_{ij}(x_i, x_j)$

Order 2

“Pairwise energy”

higher(8)-connected; pairwise MRF

$E(x) = \sum_{i,j \in \mathbb{N}_8} \theta_{ij}(x_i, x_j)$

Order 2

Higher-order RF

$E(x) = \sum_{i,j \in \mathbb{N}_4} \theta_{ij}(x_i, x_j) + \theta(x_1, \ldots, x_n)$

Order n

“higher-order energy”

…. all useful models, but how do I optimize them?
Advanced CRF system
Outline

• Introduction
• MRFs and CRFs in Vision
• Optimisation techniques and Comparison
Why is good optimization important?

**Input:** Image sequence

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

**Output:** New view

[Data courtesy from Oliver Woodford]

[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

\[ E(x) = \sum_i f_i(x_i) + \sum_{ij} g_{ij}(x_i, x_j) + \sum_c h_c(x_c) \]

Unary: \( x_i \in \{0,1\} \)
Pairwise: \( x_i \in \{0,...,K\} \)
Higher Order: \( x_c \in \{0,...,K\} \)

Label-space:

Binary: \( x_i \in \{0,1\} \)
Multi-label: \( x_i \in \{0,...,K\} \)
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 $\geq 7$ 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?

• 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

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

\[ M_{p \rightarrow q}(L) = \min_{x_p} f(x_p) + g_{pq}(x_p, L) \]

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

\[ M_{p \rightarrow q}(L_1, L_2, L_3) = (3,1,2) \]
Message Passing Chain: Dynamic Programming

\[ f(x_p) + g_{pq}(x_p,x_q) \text{ with Potts model } g_{pq} = 2 (x_p \neq x_q) \]
Message Passing Chain: Dynamic Programming

\[ M_{q \rightarrow r}(L_i) = \min_{x_q} M_{p \rightarrow q} + f(x_q) + g_{qr}(x_q, L_i) \]

Get optimal labeling for \( x_r \):

\[ \min_{x_r} M_{q \rightarrow r} + f(x_r) \]

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

• Loopy graphs: many techniques: BP, TRW, TRW-S, Diffusion:
  – Message update rules differ
  – Compute (approximate) MAP or marginals $P(x_i \mid x_{\setminus \{i\}})$
  – 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

\[
f(A) + f(B) \geq f(A \lor B) + f(A \land B) \quad \text{for all } A, B \in \{0, 1\}^n
\]

(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?

Source

$\text{v}_1$  2  9  $\text{v}_2$

$\text{v}_1$

5  2  1  4

Sink
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 \(T\)

\[5 + 1 + 9 = 15\]
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 \(T\)

What is the st-mincut?
st-cut with the minimum cost

Source

\[
\begin{pmatrix}
2 & 2 & 4 \\
9 & & \\
& & \\
\end{pmatrix}
\]

Sink

\[
2 + 2 + 4 = 8
\]
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

Solution

[Hammer, 1965] [Kolmogorov and Zabih, 2002]
st-mincut and Energy Minimization

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

For all \( i,j \)

\[ \theta_{ij}(0,1) + \theta_{ij}(1,0) \geq \theta_{ij}(0,0) + \theta_{ij}(1,1) \]

Equivalent (transform to "normal form")

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

\[ c_i, c'_i \in \{0, p\} \text{ with } p \geq 0 \]

\[ c_{ij} \geq 0 \]

[Kolmogorov and Rother ‘07]
$E(v_1, v_2) = 2v_1 + 5(1-v_1) + 9v_2 + 4(1-v_2) + 2v_1(1-v_2) + (1-v_1)v_2$
Example

The diagram illustrates a flow network with a source and a sink. The flow function is given by:

\[ E(v_1, v_2) = 2v_1 + 5(1-v_1) + 9v_2 + 4(1-v_2) + 2v_1(1-v_2) + (1-v_1)v_2 \]

The optimal st-mincut is 8, with the variables set as:

\[ v_1 = 1 \quad v_2 = 0 \]

The equation for the optimal st-mincut is:

\[ E(1,0) = 8 \]
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

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

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

[Slide credit: Andrew Goldberg]
Minimizing general non-submodular functions is NP-hard.

Commonly used method is to solve a relaxation of the problem

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

\[ \theta_{ij}(0,1) + \theta_{ij}(1,0) < \theta_{ij}(0,0) + \theta_{ij}(1,1) \text{ for some } i, j \]
Minimization using Roof-dual Relaxation

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

- Unary

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

- Pairwise submodular

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

- Pairwise nonsubmodular

[ Boros, Hammer, Sun ’91; Kolmogorov, Rother ‘07 ]
Minimization using Roof-dual Relaxation (QPBO, BHS-algorithm)

Double number of variables: \( x_p \rightarrow x_p, x_{\overline{p}} \)

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

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

\[ E(\{x_p\}) = E'(\{x_p\}, \{x_{\overline{p}}\}) \text{ if } x_{\overline{p}} = 1 - x_p \]

- 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)

\[ x_p = 1 - x_{\overline{p}} \quad \rightarrow \quad x_p \quad \text{is the optimal label} \]

- 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: $n \times k$

\[
\begin{align*}
  x_1 &= l_3 \\
  x_2 &= l_2 \\
  x_3 &= l_2 \\
  x_4 &= l_1
\end{align*}
\]

[Ishikawa PAMI '03]
Example transformation approach

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

Exact if \(g\) convex:

other encoding scheme:
[Roy and Cox '98, Schlesinger & Flach '06]

Problem: not discontinuity preserving
Move Making Algorithms

Energy

Solution Space
Move Making Algorithms

- Energy
- Solution Space
- Current Solution
- Search Neighbourhood
- Optimal Move
Iterative Conditional Mode (ICM)

\[
E(x) = \theta_{12}(x_1, x_2) + \theta_{13}(x_1, x_3) + \theta_{14}(x_1, x_4) + \theta_{15}(x_1, x_5) + \ldots
\]
Graph Cut-based Move Making Algorithms

A series of globally optimal large moves

$\text{Space of Solutions (x)} : L^n$

$\text{Move Space (t)} : 2^n$

$n \quad \text{Number of Variables}$

$L \quad \text{Number of Labels}$

[Boykov, Veksler and Zabih 2001]
Expansion Move

- Variables take label $\alpha$ or retain current label

Status: Expand House, Tree

[Boykov, Veksler, Zabih 2001]
Expansion Move

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

\[
\begin{align*}
\theta_{ij}(l_a,l_b) &= 0 \text{ iff } l_a = l_b \\
\theta_{ij}(l_a,l_b) &= \theta_{ij}(l_b,l_a) \geq 0 \\
\theta_{ij}(l_a,l_b) + \theta_{ij}(l_b,l_c) &\geq \theta_{ij}(l_a,l_c)
\end{align*}
\]

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 1

Solution from Method 2

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

\[ f(x_1, x_2, x_3) = \theta_{111} x_1 x_2 x_3 + \theta_{110} x_1 x_2 (1-x_3) + \theta_{101} x_1 (1-x_2) x_3 + \ldots \]

\[ f(x_1, x_2, x_3) = a x_1 x_2 x_3 + b x_1 x_2 + c x_2 x_3 \ldots + 1 \]

Quadratic polynomial can be done

**Idea:** transform 2+ order terms into 2\textsuperscript{nd} 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]

\[ E(x) = P(x) + \sum_{l \in L} c_l \left[ \exists p: x_p = l \right] \]

"pairwise MRF"

"Label cost"

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)

Label cost = 10c

Label cost = 4c

From [Delong et al. ‘10]

[Many more special higher-order potentials in tutorial CVPR ’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

\[ E(x) = E_1(x) + \sum \theta_i (x_i) + \sum \theta_{ij} (x_i, x_j) + h(x) \]

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

Derive Lower bound:

\[ \min_x E(x) = \min_x \left[ E_1(x) + \theta^T x + h(x) - \theta^T x \right] \]
\[ \geq \min_{x_1} [E_1(x_1) + \theta^T x_1] + \min_{x_2} [h(x_2) + \theta^T x_2] = L(\theta) \]

**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]

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

\[ F_i = -\log \Pr(z^i|\theta^F) \quad B_i = -\log \Pr(z^i|\theta^B) \]

**Problem:** for unknown \( x, \theta^F, \theta^B \) the optimization is NP-hard! [Vicente et al. '09]
MRF with global potential: GrabCut - Iterated Graph Cuts

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

Guaranteed to converge

1 2 3 4
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

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
  (QPBO-P solves all)

Ground truth

QPBO (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; 80 con; 100% non-sub; 109 unary strength

Ground Truth

Input

5x5 blur kernel

MRF: 80 connectivity - illustration
Binary Image Deconvolution

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

Ground Truth

Input

QPBO 80% unlab. (0.1sec)

QPBO 80% unlab. (0.9sec)

ICM E=6 (0.03sec)

GC E=999 (0sec)

BP E=71 (0.9sec)

QPBO+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”

[Image of various labeled images and diagrams]

- Stereo
- Panoramic stitching
- Image Segmentation;
  de-noising;
  in-painting

[Szelsiki et al ’06,08]
Stereo

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:

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

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 Linear Program (LP relaxation)
- Solve LP (polynomial time algorithms)
- Round LP to get best IP solution (no guarantees)
MAP Inference as an IP

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

s.t. \[
\begin{align*}
\sum_{a \in L} x_{p,a} &= 1 \\
\sum_{a \in L} x_{pq,ab} &= x_{q,b} \\
\sum_{b \in L} x_{pq,ab} &= x_{p,a} \\
x_{p,a}, x_{pq,ab} &\in \{0, 1\}
\end{align*}
\]
Relax to LP

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

s.t. $\sum_{a \in L} x_{p,a} = 1$

$\sum_{a \in L} x_{pq,ab} = x_{q,b}$

$\sum_{b \in L} x_{pq,ab} = x_{p,a}$

$x_{p,a} \geq 0, x_{pq,ab} \geq 0$

**Linear Program**

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