MAP-MRF approach with graph-cuts

Uma Mudanagudi

Dept. of Computer Science and Engineering,
Indian Institute of Technology Delhi
Outline

- Motivation of selecting MAP-MRF framework
- MAP-MRF approach
- Mapping of MAP-MRF to graph-cut problem
- What Energy functions can be minimized with Graph-cuts?
- Some examples of MAP-MRF framework using graph-cut
- MAP-MRF framework for Super Resolution
- Summary
Motivation of selecting MAP-MRF framework

- Bayesian framework suitable for problems in Computer Vision
- MAP-MRF with Gibbs gives easy implementation and formulation.
- Problems: High computational cost or Standard methods used are very slow.
- Boykov et.al proposed methods to solve MAP-MRF using graph-cut algorithms - MAP-MRF estimation is equivalent to min-cut problem on a graph
- Applied to many vision problems
Outline

- Motivation of selecting MAP-MRF framework
- MAP-MRF approach
- Mapping of MAP-MRF to graph-cut problem
- What Energy functions can be minimized with Graph-cuts?
- Some examples of MAP-MRF framework using graph-cut
- MAP-MRF framework for Super Resolution
- Summary
MAP-MRF framework

- MRF framework: Given set of pixels $S = \{s_1 \ldots s_m\}$ and set of labels $\Lambda = \{l_1 \ldots l_L\}$ and neighborhood system $N$, Find mapping of $S$ to $\Lambda$.

- Let $F$ be the configuration for labels
  $F = \{f_i \ldots f_N\}, f_i \in \Lambda$ is the label for $s_i$

- $F$ is MRF with respect to $N$ iff
  - Positivity: $P(F = f) \geq 0 \ \forall \ f \in F$
  - Markovianity:
    $$P(F_s = f_s/F_r = f_r, \forall r \neq s) = P(F_s = f_s/F_r = f_r, \forall r \in N_s)$$

- Easy Implementability
Let $G$ be the observed image

$G = \phi(H(F)) + N$

where $H =$ Camera Transfer Function and $\phi =$ Recorder distortion

$H$ is assumed to be LSI and $\phi$ is invertible nonlinear function and $N$ is additive noise assumed to be iid

In the framework of Restoration: Given $G$ What is $F$?

- $P(F = f | G = g)$, Maximum likelihood of $F = f$ given $G = g$

- From Bayes rule
  
  $P(F = f | G = g) \propto P(G = g | F = f)P(F = f)$

  where $P(G = g | F = f)$ = Data model, $P(F = f)$ = Prior and $P(F = f | G = g)$ = A posteriori distribution

- Need to maximize a posteriori (MAP) distribution
Gibbs Distribution

- Geman and Geman proved equivalence between MRF and Gibbs distribution

\[ P(f) = \frac{1}{Z} \exp\left(-\frac{U(f)}{T}\right) \]

where \( U(f) = \sum_{c \in N} V_c(f) \) = Energy function, \( V_c \) = Clique potential, \( Z = \sum_{f} \exp\left(-\frac{U(f)}{T}\right) \) = Partition function and \( T \) = Temperature

- Hammersley Clifford Theorem:

  \( F \) is MRF on \( S \) with respect to \( N \)
  
  if and only if
  \( F \) is Gibbs random field on \( S \) with respect to \( N \)

- Relates the conditional distribution(local characteristics) and joint distribution(Gibbs measure)

- Easy Formulation
MAP-MRF

\[ \hat{f} = \arg\max_f P(f|g) \]

From Bayes Rule

\[ \hat{f} = \arg\max_f P(g|f) \hat{P}(f) \]

I-term: Likelihood function and II-term: Prior Model

\[ \hat{f} = \arg\max_f \exp\left\{ \sum_p \ln(p(g|f_p)) - \sum_{p,q \in N} V_{p,q}(f_p, f_q) \right\} \]

MAP estimate of \( f \) given \( g \) is equivalent to minimizing energy function with prior and data model

\[ \hat{f} = \arg\min_f \left\{ -\sum_p \ln(p(g|f_p)) + \sum_{p,q \in N} V_{p,q}(f_p, f_q) \right\} \]
MAP-MRF

The Energy Function has data term and regularization term

\[ U(f) = \left\{ \sum_i (g_i - \phi(H(f_i))) + \sum_{i,j \in N} V_{i,j}(f_i, f_j) \right\} \]

Different ways of defining Clique potential which defines the regularization term or smoothness term in the energy function and describes the prior probability of a particular realization of the elements of the clique.

Data model should capture the cost of assigning the label

MAP-MRF is usually solved using SA which is very slow but guarantees the global minima for any arbitrary energy function

Boykov et.al suggested max-flow/min-cut graph algorithms to solve some class of energy functions with MAP-MRF framework within a known factor of global minimum
Outline

- Motivation of selecting MAP-MRF framework
- MAP-MRF approach
- Mapping of MAP-MRF to graph-cut problem
- What Energy functions can be minimized with Graph-cuts?
- Some examples of MAP-MRF framework using graph-cut
- MAP-MRF framework for Super Resolution
- Summary
Mapping of MAP-MRF to graph-cut

- Vision problems as image labeling: Depth (stereo), Object Index (Segmentation), Original Intensity (Restoration)

- Labeling problem can be cast in terms of energy minimization
  - Labeling of pixels
  - Penalty for pixel labeling
  - Interaction between neighboring pixels: Smoothness term

- All pixels and labels are considered as vertices, edge and edge weights are calculated dynamically

- Min-cut on G has unique binary segmentation

- Segmentation associated with min-cut that satisfies user defined constraints minimizes the energy function
Energy Minimization

- Global minimum can be found in polynomial time if the energy function
  - is convex,
  - or has only two labels, eg. Icing model.

- Discontinuity-preserving energy function is not convex, eg. Potts model. Thus global minimization is NP-hard, takes exponential time,

- Thus global minimization Approximation algorithm to find local minimum
  - EM, Belief Propagation, Graph-Cuts

- What is Graph-Cuts?
  - Minimize an energy function with non binary variables by repeatedly minimizing an energy function with binary variables using Max-flow/ min-cut method
**s-t graph-cuts for Binary Energy Minimization**

- Posterior energy (MRF)
- Complete characterization of binary energies that can be minimized with s-t graph cuts.

\[ U(f) = \sum_p (D_p(f_p)) + \sum_{pq \in N} V(f_p, f_q) \]

- \( U(f) \) can be minimized by graph-cuts
  \[ \Leftrightarrow V(s, s) + V(t, t) \leq V(s, t) + V(t, s) \]

Example: binary image restoration
s-t graph-cuts for multi label problems

- Class of Energy that can be minimized exactly: Energies with convex interactions
  - excludes robust discontinuity-preserving interactions

- Guaranteed quality approximation algorithms for multi-label energies with discontinuity-preserving interactions like Potts model of interactions and Metric interactions

Graph \( G(C) = \langle V, E-C \rangle \)
Different types of Pixel Interactions

Discontinuity preserving interactions:

Convex interactions:
Linear Models
Convex vs. Discontinuity-preserving
Multi way Graph-cut

\[ E(L) = \sum_p -D_p(L_p) + \sum_{pq} w_{pq} \cdot \delta_{L_p \neq L_q} \]
Multi way Graph-cut algorithms by Boykov et.al

- Equivalent to Potts energy minimization
- NP-hard problem (3 or more labels)
  - Two labels can be solved via s-t cuts (Greig et. al., 1989)
- Two approximation algorithms (Boykov et.al 1998, 2001)
  Basic Idea: break multi-way cut computation into a sequence of binary s-t cuts.
  - $\alpha$- Expansion
    - Each label competes with the other labels for space in the image
  - $\alpha - \beta$ Swap: Define a move which allows to change pixels from $\alpha$ to $\beta$ and $\beta$ to $\alpha$
\(\alpha\)-Expansion approximation algorithm

Guaranteed quality approximation

- within a factor of 2 from Global minimum (Potts Model)
- applies to a wide class of energies with robust interactions
- Potts model (BVZ 1989), Metric interactions (BVZ 2001), Sub modular interactions (KZ 2002, 2004)

Algorithm

1. Start with any arbitrary labeling \(f\)
2. Set \(\text{success} = 0\)
3. For each label \(\alpha \in L\) (random order)
   (a) find \(\hat{f} = \text{argmin} \ U(f^{1})\) among \(f^{1}\) within one \(\alpha\)-expansion \(f\)
   (b) If \(U(\hat{f}) < U(f)\), set \(f = \hat{f}\) and \(\text{success} = 1\)
4. If \(\text{success} = 1\) go to step 2
5. return \(f\)
\( \alpha - \beta \) Swap approximation algorithm

Handles more general energy functions

- Experimentally proved results

**Algorithm**

1. Start with any arbitrary labeling \( f \)
2. Set \( \text{success} = 0 \)
3. For each pair of labels \( \{\alpha, \beta\} \in L \) (random order)
   a. find \( \hat{f} = \arg\min_U U(f^1) \) among \( f^1 \) within one \( \alpha - \beta \) swap of \( f \)
   b. If \( U(\hat{f}) < U(f) \), set \( f = \hat{f} \) and \( \text{success} = 1 \)
4. If \( \text{success} = 1 \) go to step 2
5. return \( f \)
Moves

- Initial labeling
- $\alpha$-$\beta$-swap
- $\alpha$-expansion
Finding optimal expansion move

3. A step in algo. The structure of the graph is dynamically determined by the current position \( P \) and label \( \alpha \).

Figure 6: An example of \( G_\alpha \) for a 1D image. The set of pixels in the image is \( \mathcal{P} = \{p, q, r, s\} \) and the current partition is \( \mathcal{P} = \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_\alpha\} \) where \( \mathcal{P}_1 = \{p\} \), \( \mathcal{P}_2 = \{q, r\} \), and \( \mathcal{P}_\alpha = \{s\} \). Two auxiliary nodes \( a = a_{(p,q)} \), \( b = a_{(r,s)} \) are introduced between neighboring pixels separated in the current partition. Auxiliary nodes are added at the boundary of sets \( \mathcal{P}_i \).
## Expansion move-assignment of weights

<table>
<thead>
<tr>
<th>edge</th>
<th>weight</th>
<th>for</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_{p}^{\alpha}$</td>
<td>$\infty$</td>
<td>$p \in \mathcal{P}_{\alpha}$</td>
</tr>
<tr>
<td>$t_{p}^{\alpha}$</td>
<td>$D_{p}(f_{p})$</td>
<td>$p \notin \mathcal{P}_{\alpha}$</td>
</tr>
<tr>
<td>$t_{p}^{\alpha}$</td>
<td>$D_{p}(\alpha)$</td>
<td>$p \in \mathcal{P}$</td>
</tr>
<tr>
<td>$e_{{p,a}}$</td>
<td>$V(f_{p}, \alpha)$</td>
<td>${p, q} \in \mathcal{N}$, $f_{p} \neq f_{q}$</td>
</tr>
<tr>
<td>$e_{{a,q}}$</td>
<td>$V(\alpha, f_{q})$</td>
<td>${p, q} \in \mathcal{N}$, $f_{p} = f_{q}$</td>
</tr>
<tr>
<td>$t_{\alpha}$</td>
<td>$V(f_{p}, f_{q})$</td>
<td></td>
</tr>
<tr>
<td>$e_{{p,q}}$</td>
<td>$V(f_{p}, \alpha)$</td>
<td>${p, q} \in \mathcal{N}$, $f_{p} = f_{q}$</td>
</tr>
</tbody>
</table>
Finding optimal swap move

3.1 step in algo. The structure of the graph is dynamically determined by the current position $P$ and labels $\alpha, \beta$.

Figure 4: An example of the graph $G_{\alpha \beta}$ for a 1D image. The set of pixels in the image is $\mathcal{P}_{\alpha \beta} = \mathcal{P}_\alpha \cup \mathcal{P}_\beta$ where $\mathcal{P}_\alpha = \{p, r, s\}$ and $\mathcal{P}_\beta = \{q, \ldots, w\}$.
### Optimal swap move-assignment of weights

<table>
<thead>
<tr>
<th>edge</th>
<th>weight</th>
<th>for</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t^\alpha_p$</td>
<td>$D_p(\alpha) + \sum_{\substack{q \in N_p \ q \notin \mathcal{P}_{\alpha \beta}}} V(\alpha, f_q)$</td>
<td>$p \in \mathcal{P}_{\alpha \beta}$</td>
</tr>
<tr>
<td>$t^\beta_p$</td>
<td>$D_p(\beta) + \sum_{\substack{q \in N_p \ q \notin \mathcal{P}_{\alpha \beta}}} V(\beta, f_q)$</td>
<td>$p \in \mathcal{P}_{\alpha \beta}$</td>
</tr>
<tr>
<td>$\epsilon_{{p,q}}$</td>
<td>$V(\alpha, \beta)$</td>
<td>${p,q} \in \mathcal{N} \ \forall \ p,q \in \mathcal{P}_{\alpha \beta}$</td>
</tr>
</tbody>
</table>
α-Expansion move
Example for Metric Interactions
Comparison

single “one-pixel” move (simulated annealing, ICM,...)

• Only one pixel can change its label at a time
• Finding an optimal move is computationally trivial

single a-expansion move

• Large number of pixels can change their labels simultaneously
• Finding an optimal move is computationally intensive \(O(2^n)\) (s-t cuts)
Comparisons contd..

**simulated annealing**
- Finds local minimum of energy with respect to small one-pixel moves
- Initialization is important
- Solution could be arbitrarily far from the global minima
- May not know when to stop. Practical complexity may be worse than exhaustive search
- Can be applied to anything

**\(\alpha\)-Expansion**
- Finds local minimum of energy with respect to very strong moves
- In practice, results do not depend on initialization
- Solution is within the factor of 2 from the global minima
- In practice, one cycle through all labels gives sufficiently good results
- Applies to a restricted class of energies
Outline

- Motivation of selecting MAP-MRF framework
- MAP-MRF approach
- Mapping of MAP-MRF to graph-cut problem
- What Energy functions can be minimized with Graph-cuts?
- Some examples of MAP-MRF framework using graph-cut
- MAP-MRF framework for Super Resolution
- Summary
What Energy functions can be minimized with Graph-cuts?

- $\alpha$-expansion algorithm can be applied to pairwise interactions that are metric on the space of labels
  
  - $V(a, a) = 0$
  
  - $V(a, b) \geq 0$
  
  - $V(a, b) \leq V(a, c) + V(c, b)$

- Any truncated metric is also a metric (includes robust interactions)

- $\alpha$-expansion algorithm further generalizes to submodular pair-wise interactions

\[ V(c, c) + V(a, b) \leq V(a, c) + V(c, b) \]

- $\alpha - \beta$ swap can be applied to pairwise interactions which are semi-metric on the space of labels

- Let $E$ be a function of binary variables. If $E$ is not regular, then $E$ is not graph-representable.
Regular functions

- All functions of one variable are regular.
- A function $V$ of two variables is called regular if
  \[ V(0, 0) + V(1, 1) \leq V(0, 1) + V(1, 0) \]
- A function $V$ of more than two variables is called regular if all projections of $V$ of two variables are regular.
- Let $V$ be a function of $n$ binary variables from $F^3$, ie.
  \[
  V(x_1, \ldots x_n) = \\
  \sum_i V^i(x_i) + \sum_{i<j} V^{i,j}(x_i, x_j) + \sum_{i<j<k} V^{i,j,k}(x_i, x_j, x_k).
  \]
  Then, $V$ is graph-representable if and only if $V$ is regular.
- Any projection of a graph-representable function is graph-representable.
Moves

If $V$ is Metric, then each expansion move is regular

$$E(0, 0) + E(1, 1) = V(\beta, \gamma) + V(\alpha, \alpha) \leq V(\beta, \alpha) + V(\alpha, \gamma) = E(0, 1) + E(1, 0)$$

If $V$ is Semi-metric, then each swap move is regular

$$E(0, 0) + E(1, 1) = V(\beta, \beta) + V(\alpha, \alpha) \leq V(\beta, \alpha) + V(\alpha, \beta) = E(0, 1) + E(1, 0)$$
Outline

- Motivation of selecting MAP-MRF framework
- MAP-MRF approach
- Mapping of MAP-MRF to graph-cut problem
- What Energy functions can be minimized with Graph-cuts?
- Some examples of MAP-MRF framework using graph-cut
- MAP-MRF framework for Super Resolution
- Summary
Some examples of MAP-MRF using graph-cut

- Image segmentation 1: Jiangjian Xiao and Mubarak Shah
  CVPR 2005
  - Motion cue to segment using graph-cut
  - Refine the segmentation by alpha matte
  - Example 1

- Image segmentation/Object Extraction: Yuri Boykov and Vladimir Kolmogorov 2004
  - Combine both active contours and graph-cuts
  - Reduces the metrification error
  - Example 2
Examples contd..

- Texture Synthesis: Image quilting by Efros and Freeman, 2001
  Example

- Video Texture: 3D generalization of Image quilting by Kwatra, Schodl, Essa, Bobick 2003
  Process
  Source
  Synthesized

- Stereo: Boykov et.al 98, 2002
  Example

- Multiview reconstruction by Boykov et.al 2004
  Example

- Interactive Digital photomontage by microsoft research lab
  Example
Outline

- Motivation of selecting MAP-MRF framework
- MAP-MRF approach
- Mapping of MAP-MRF to graph-cut problem
- What Energy functions can be minimized with Graph-cuts?
- Some examples of MAP-MRF framework using graph-cut
- MAP-MRF framework for Super Resolution
- Summary
The Energy function is given by

\[ E(f) = \sum_p D_p(f_p) + \sum_{p,q \in N_p} V_{p,q}(f_p, f_q) + \sum_{(p,q)\downarrow \in N_p} V_{(p,q)\downarrow}(g_{p\downarrow}, g_{q\downarrow}) \]

where \( D = g - DHf \) is a data model term and next two terms are regularization terms. \( g \) = observed image, \( D \) = decimation function and \( H \) = Camera transfer function.

\( f \) is the Super Resolution Image - needs to estimate

Regularization terms are truncated linear models

\[ V_{p,q}(f_p, f_q) = \min(K, |f_p - f_q|) \]

\[ V_{(p,q)\downarrow}(g_{p\downarrow}, g_{q\downarrow}) = \min(K, |g_{p\downarrow} - g_{q\downarrow}|) \]

Once we have the MAP-MRF framework for SR, we can apply Graph-cuts to estimate \( f \).
Outline

- Motivation of selecting MAP-MRF framework
- MAP-MRF approach
- Mapping of MAP-MRF to graph-cut problem
- What Energy functions can be minimized with Graph-cuts?
- Some examples of MAP-MRF framework using graph-cut
- MAP-MRF framework for Super Resolution
- Summary
Summary

- MAP-MRF framework
- What energy functions can be minimized with Graph-cuts
- $\alpha$–Expansion and $\alpha – \beta$ Swap algorithms in Graph-cuts
- Examples using Graph-cuts
- Framework of SR
Thank you

THANK YOU