Review - Solving MRFs with graph cuts

Main idea:
- Construct a graph such that every st-cut corresponds to a joint assignment to the variables $y$
- The cost of the cut should be equal to the energy of the assignment, $E(y; data)^\ast$.
- The minimum-cut then corresponds to the minimum energy assignment, $y^\ast = \text{argmin}_y E(y; data)$.

\* Requires non-negative energies

S. Gould
Review - Solving MRFs with graph cuts

\[ \text{Energy}(y; \theta, \text{data}) = \sum_{i} \psi_{1}(y_i; \theta, \text{data}) + \sum_{i,j \text{ edge}} \psi_{2}(y_i, y_j; \theta, \text{data}) \]

P. Kohli

**Code for Image Segmentation**

\[ E(x) = \sum_{i} c_{i} x_{i} + \sum_{i,j} d_{ij} |x_{i} - x_{j}| \]

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

0 \rightarrow fg
1 \rightarrow bg

\[ n = \text{number of pixels} \]

\[ x = \arg \min_{x} E(x) \]

Global Minimum \((x')\)

P. Kohli

Review - How does the code look like?

```c
graph *g;

for all pixels p
    /* Add a node to the graph */
    nodeID(p) = g->add_node();
    /* Set cost of terminal edges */
    set_weights(nodeID(p), fgCost(p), bgCost(p));
end

for all adjacent pixels p, q
    add_weights(nodeID(p), nodeID(q), cost(p, q));
end

g->compute_maxflow();

label_p = g->is_connected_to_source(nodeID(p)); // is the label of pixel p (0 or 1)
```

P. Kohli
Graph *g;

For all pixels p
/* Add a node to the graph */
nodeID(p) = g->add_node();
/* Set cost of terminal edges */
set_weights(nodeID(p), fgCost(p), bgCost(p));
end

for all adjacent pixels p,q
add_weights(nodeID(p), nodeID(q), cost(p,q));
end
g->compute_maxflow();

label_p = g->is_connected_to_source(nodeID(p));
// is the label of pixel p (0 or 1)

Review - Random Fields in Vision

4-connected; pairwise MRF
E(x) = \sum_{i \in N_4} \theta_i (x_i, x_j)
Order 2

higher(8)-connected; pairwise MRF
E(x) = \sum_{i \in N_8} \theta_i (x_i, x_j)
Order 2

MRF with global variables
E(x) = \sum_{i \in N_4} \theta_i (x_i, x_j) + \theta(x_1, x_2)
Order 2

Higher-order MRF
E(x) = \sum_{i \in N_8} \theta_i (x_i, x_j) + \theta(x_1, \ldots, x_n)
Order n

C. Rother

Review - 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) + \sum_{i \in N_4} |x_i-x_j|

F_i = -\log Pr(z_i|\theta^F) 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]

C. Rother
**Review - GrabCut: Iterated Graph Cuts**

[Rother et al. Siggraph '04]

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

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]

---

**Review - Random Fields in Vision**

4-connected; pairwise MRF

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

Order 2

MRF with global variables

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

Order 2

\[ E(x) = \sum_{i \in N_8} \theta_{ij}(x_i, x_j) + \theta(x_1, \ldots, x_n) \]

Order n

---

**Review - Why Higher-order Functions?**

In general \( \theta(x_i, x_j, x_k) \neq \theta(x_i, x_j) + \theta(x_i, x_k) + \theta(x_j, x_k) \)

Reasons for higher-order RFs:

1. Even better image(texture) models:
   - Field-of Expert [FoE, Roth et al. '05]
   - Curvature [Woodford et al. '06]

2. Use global Priors:
   - Connectivity [Vicente et al. '08, Nowozin et al. '09]
   - Better encoding label statistics [Woodford et al. '09]
   - Convert global variables to global factors [Vicente et al. '09]

---

**Semantic Segmentation**

- Joint recognition & segmentation
  - segmenting all the objects in a given image and identifying their visual categories
- aka scene parsing or image parsing

- Early studies aim at segmenting out a single object of a known category
  - Borenstein & Ullman, 2002, Liebe & Schiele, 2003,
Early Studies of Semantic Segmentation

- Given an image and object category, to segment the object.

- Segmentation should (ideally) be:
  - shaped like the object e.g. cow-like
  - obtained efficiently in an unsupervised manner
  - able to handle self-occlusion

Using Normalized Cuts, Shi & Malik, 1997
Early Studies of Semantic Segmentation

Using Normalized Cuts, Shi & Malik, 1997

Borenstein and Ullman, ECCV 2002

Jigsaw approach: Borenstein and Ullman, 2002

Implicit Shape Model - Liebe and Schiele, 2003

Random Fields for segmentation

\[ I = \text{Image pixels (observed)} \]
\[ h = \text{foreground/background labels (hidden) – one label per pixel} \]
\[ \theta = \text{Parameters} \]

\[ p(h \mid I, \theta) \]

Posterior
Random Fields for segmentation

\[ I = \text{Image pixels (observed)} \]
\[ h = \text{foreground/background labels (hidden) – one label per pixel} \]
\[ \theta = \text{Parameters} \]

\[ p(h | I, \theta) \propto p(I, h | \theta) = p(I | h, \theta)p(h | \theta) \]

1. Generative approach models joint
   → Markov random field (MRF)

2. Discriminative approach models posterior directly
   → Conditional random field (CRF)

Generative Markov Random Field

\[ p(h, I | \theta) = \frac{p(I | h, \theta)p(h | \theta)}{Z(\theta)} \]
\[ = \frac{1}{Z(\theta)} \prod_i \phi_i(I | h_i, \theta) \prod_i \psi_{ij}(h_i, h_j | \theta_{ij}) \]

- Prior has no dependency on \( I \)

- Dependency on \( I \) allows introduction of pairwise terms that make use of image.

- For example, neighboring labels should be similar only if pixel colors are similar → Contrast term

Conditional Random Field

\[ p(h | I, \theta) = \frac{1}{Z(I, \theta)} \left[ \prod_i \phi_i(h_i | I, \theta_i) \prod_{ij} \psi_{ij}(h_i, h_j | I, \theta_{ij}) \right] \]

- Discriminative approach

  - Lafferty, McCallum and Pereira 2001

  - Dependency on \( I \)

  - For example, neighboring labels should be similar only if pixel colors are similar → Contrast term

  - e.g. Kumar and Hebert 2003

Levin & Weiss [ECCV 2006]

\[ E(h; I) = \sum_i \lambda_i |h_i - h_{F,i}| + \sum_{ij} \psi(i,j) |h_i - h_j| \]

- Consistency with fragments segmentation

- Segmentation alignment with image edges

- Resulting min-cut segmentation
Semantic Segmentation
Joint Object recognition & segmentation

Goal: Detect and segment test image:

Large set of example segmentation:

\[ E(x,w) = \sum |T(w) - x_i| + \sum \theta_{ij}(x_i, x_j) \]

\[ i,j \in N \]

Up to 2.000.000 shape templates

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

\[ E(x,w) = \sum |T(w) - x_i| + \sum \theta_j(x_i, x_j) \]

"Hamming distance"

[ Lempitsky et al. ECCV '08 ]

C. Rother

Semantic Segmentation
Joint Object recognition & segmentation

\[ E(x,w) = \sum \theta_i(\omega, x_i) + \sum \theta_i(x_i) + \sum \theta_j(x_i, x_j) \]

\[ i,j \in N \]

For \( K \) object classes

\[ x_i \in \{1, \ldots, K\} \]

Location

Class (boosted textons)

\[ \text{sky, grass} \]

[ TextonBoost; Shotton et al., '06 ]

Semantic Segmentation
Joint Object recognition & segmentation

UIUC dataset; 98.8% accuracy

Class+
location + edges + color

TextonBoost; Shotton et al., '06

C. Rother
Nonparametric Scene Parsing via Label Transfer (Liu et al. TPAMI’12)

A non-parametric formulation

• Framework consists of three main modules:
  1. Scene retrieval: finding nearest neighbors (k-NN approach)
  2. Dense scene alignment: dense scene matching (SIFT Flow)
Dense Scene Alignment via SIFT Flow

- SIFT Flow (Liu et al., ECCV 2008)
  - Finds semantically meaningful correspondences among two images by matching local SIFT descriptors

Label Transfer

- A set of voting candidates \(\{s_i,c_i,w_i\}_{i=1,M}\) is obtained from the retrieved images with \(s_i, c_i\), and \(w_i\) denoting the SIFT image, annotation, and SIFT flow field of the \(i\)th voting candidate.
- A probabilistic MRF model is built to integrate
  - multiple category labels,
  - prior object (category) information
  - spatial smoothness of category labels
- \(\log P(c|I, s, \{s_i, c_i, w_i\}) = \sum_p \psi(c(p); s, \{s'_i\}) + \alpha \sum_p \lambda(c(p)) + \beta \sum_{(p,q)\in\epsilon} \phi(c(p), c(q); I) + \log Z\)

Dense Scene Alignment via SIFT Flow

- SIFT Flow (Liu et al., ECCV 2008)
  - Finds semantically meaningful correspondences among two images by matching local SIFT descriptors

\[
E(w) = \sum_p \min\{s_1(p) - s_2(p + w(p))\}, \ell + \text{data term} \\
\sum_p \eta(|v(p)| + |v'(p)|) + \text{small displacement term} \\
\sum_{(p,q)\in\epsilon} \min(\lambda|u(p) - u(q)|, \delta) + \min(\lambda|v(p) - v(q)|, \delta), \text{smoothness term}
\]

\(w(p) = u(p), v(p)\) : flow vector at point \(p\)

Label Transfer

- Likelihood term:
  \[
  \psi(c(p) = l) = \begin{cases} 
  \min_{i\in\Omega_{p,l}} \|s(p) - s_i(p + w(p))\|, & \Omega_{p,l} \neq \emptyset, \\
  \tau, & \Omega_{p,l} = \emptyset,
  \end{cases}
  \]
- \(\Omega_{p,l} = \{i; c_i(p + w(p)) = l\}\) where \(l=1,...,L\) indicates the index set of the voting candidates whose label is \(l\) after being warped to pixel \(p\).
- \(\tau\) is set to be the value of the maximum difference of SIFT feature: \(\tau = \max_{s_1,s_2,p} \|s_1(p) - s_2(p)\|\)
Label Transfer

- Prior term:
  \[ \lambda(c(p) = l) = -\log \text{hist}_l(p) \]

- The prior probability that the object category \( l \) appears at pixel \( p \).
  - obtained by counting the occurrence of each object category at each location in the training set
  - Location prior

Label Transfer

- Spatial smoothness term:
  \[ \phi(c(p), c(q)) = \delta[c(p) \neq c(q)] \left( \frac{\xi + e^{-\gamma|f(p) - f(q)|^2}}{\xi + 1} \right) \]

- The neighboring pixels into having the same label with the probability depending on the image edges:
  - Stronger the contrast, the more likely it is that the neighboring pixels may have different labels.

Parsing Results

![Query Image, Result, Ground Truth, Retrieved Images and Annotations, Flow Field, Warped Images and Annotations]

![Query Image, Result, Ground Truth, Retrieved Images and Annotations, Flow Field, Warped Images and Annotations]
Because the regularity of the database is the key to the success, we remove the SIFT flow matching, i.e., set the flow vector to be zero for every pixel, and obtain an average recognition rate of 61.23 percent without MRF and 67.96 percent with MRF, shown in Figs. 12d and 12f, respectively. This result is significant because SIFT flow is the bottleneck of the system in terms of speed. A fast implementation of our system consists of removing the dense scene alignment module, and simply performing a grid-to-grid label transfer (the likelihood term in the label transfer module still comes from SIFT descriptor distance).

How would different scene retrieval techniques affect our system? Other than the GIST distance used for retrieving nearest neighbors for the results in Fig. 12, we also use the spatial pyramid histogram intersection of HOG visual words and of the ground-truth annotation, with the corresponding per-class recognition rate displayed in Figs. 12g and 12h, respectively. For this database, GIST performs slightly better than HOG visual words. We also explore an upper bound of the label transfer framework in the ideal scenario of having access to perfect scene matching. In particular, we retrieve the nearest neighbors for each image using their ground truth annotation.