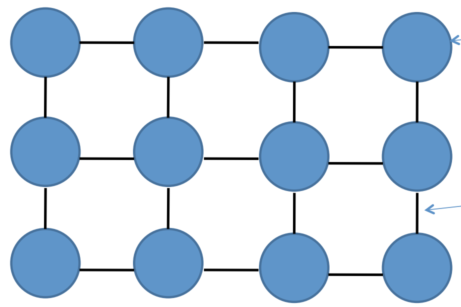
BIL 717
Image Processing

Semantic Segmentation

Erkut Erdem
Hacettepe University
Computer Vision Lab (HUCVL)

Review - Markov Random Fields



Unary potential
0: $-\log P(y_i = 0 ; data)$
1: $-\log P(y_i = 1 ; data)$

Pairwise Potential

	0	1
0	0	K
1	K	0

- Example: "label smoothing" grid

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

D. Hoiem

Review - Solving MRFs with graph cuts

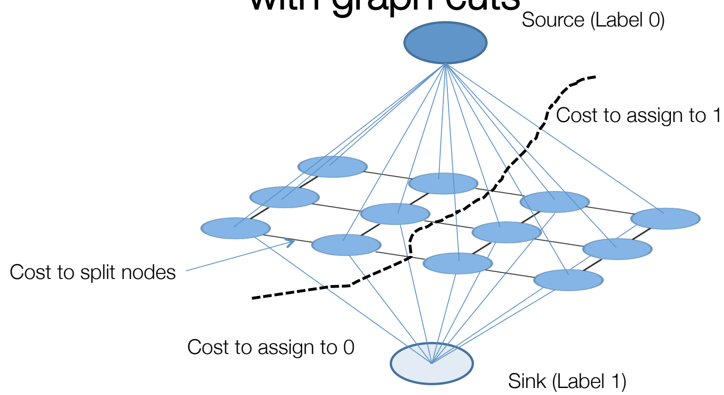
Main idea:

- 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)*.
- The minimum-cut then corresponds to the minimum energy assignment, **y*** = $\text{argmin}_{\mathbf{y}}$ E(**y**; data).

* Requires non-negative energies

S. Gould

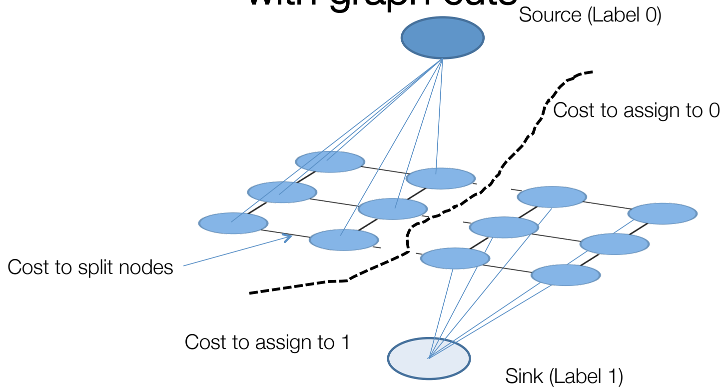
Review - Solving MRFs with graph cuts



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

D. Hoiem

Review - Solving MRFs with graph cuts



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

D. Hoiem

Code for Image Segmentation

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

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

n = number of pixels

Global Minimum (x^*)

$$x^* = \arg\min_x E(x)$$

How to minimize E(x)?

P. Kohli

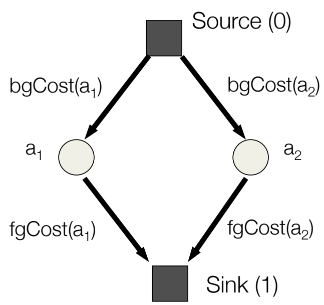
Review - How does the code look like?

```
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

Review - How does the code look like?

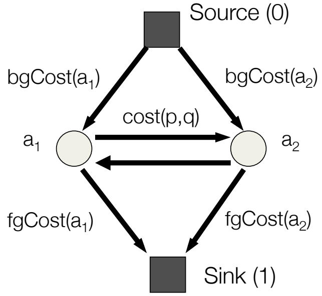
```
Graph *g;
For all pixels p

    /* 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

Review - How does the code look like?

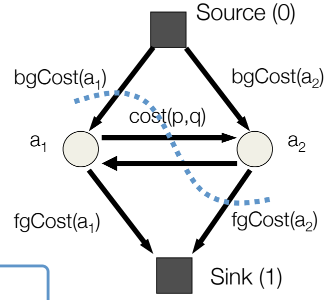
```
Graph *g;
For all pixels p

    /* 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

Review - How does the code look like?

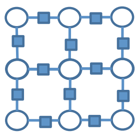
```
Graph *g;
For all pixels p

    /* 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)
```



a_1 = bg a_2 = fg

P. Kohli

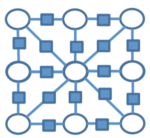
Review - Random Fields in Vision



4-connected; pairwise MRF

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

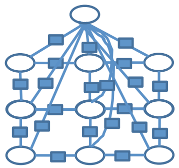
Order 2



higher(8)-connected; pairwise MRF

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

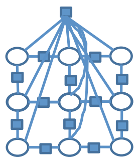
Order 2



MRF with global variables

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

Order 2



Higher-order MRF

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

Order n

C. Rother

Review - MRF with global potential

GrabCut model [Rother et. al. '04]

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

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

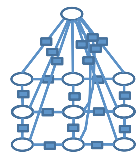
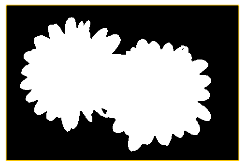
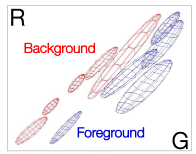


Image z



Output x



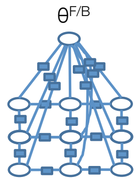
$\theta^{F/B}$ Gaussian Mixture models

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

C. Rother

Review - GrabCut: Iterated Graph Cuts

[Rother et al. Siggraph '04]



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

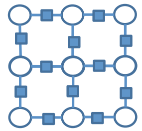
Learning of the colour distributions

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]

C. Rother

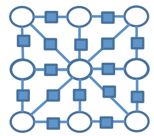
Review - Random Fields in Vision



4-connected; pairwise MRF

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

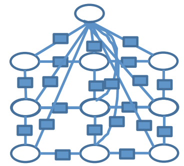
Order 2



higher(8)-connected; pairwise MRF

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

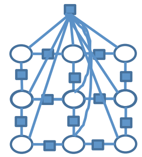
Order 2



MRF with global variables

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

Order 2



Higher-order MRF

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

Order n

C. Rother

Review - 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 RFs:</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, Nowozin et al. '09]
   - Better encoding label statistics [Woodford et al. '09]
   - Convert global variables to global factors [Vicente et al. '09]

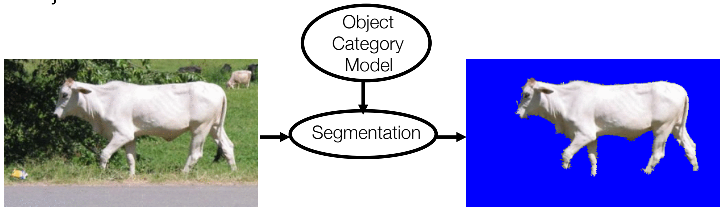
C. Rother

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



Cow Image

Segmented Cow

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

M. P. Kumar

Early Studies of Semantic Segmentation

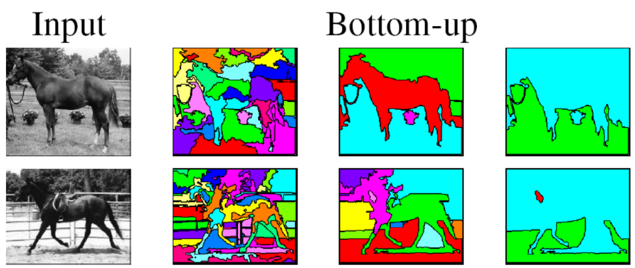
R. Fergus

Early Studies of Semantic Segmentation

R. Fergus

Early Studies of Semantic Segmentation

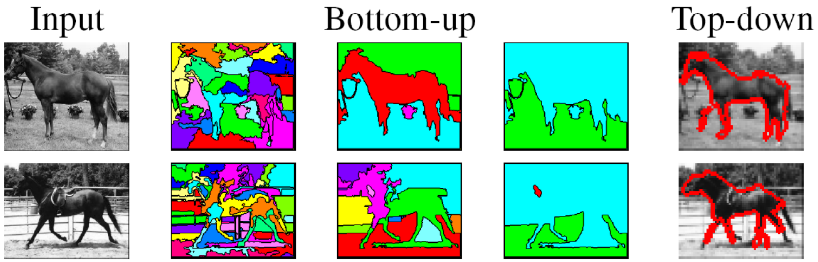
Using Normalized Cuts, Shi & Malik, 1997



R. Fergus

Early Studies of Semantic Segmentation

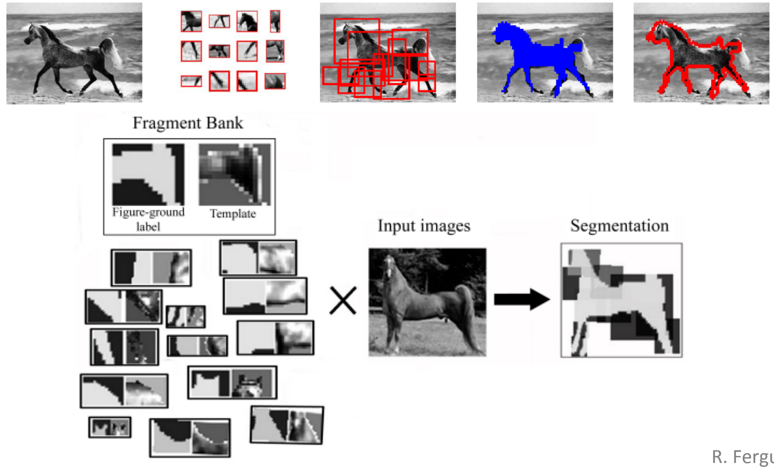
Using Normalized Cuts, Shi & Malik, 1997



Borenstein and Ullman, ECCV 2002

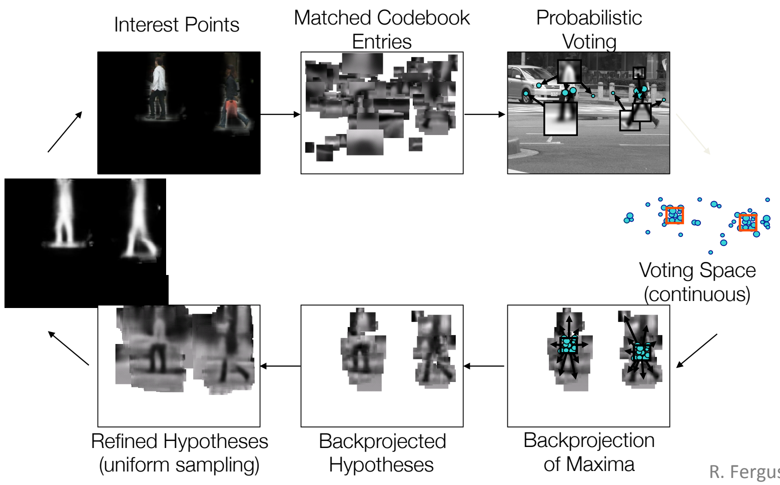
R. Fergus

Jigsaw approach: Borenstein and Ullman, 2002



R. Fergus

Implicit Shape Model - Liebe and Schiele, 2003



R. Fergus

Random Fields for segmentation

I = Image pixels (observed)
h = foreground/background labels (hidden) – one label per pixel
θ = Parameters

$$\underbrace{p(h|I,\theta)}_{\text{Posterior}}$$

R. Fergus

Random Fields for segmentation

I = Image pixels (observed)
h = foreground/background labels (hidden) – one label per pixel
θ = Parameters

$$\underbrace{p(h\,|\,I,\theta)}_{\text{Posterior}} \propto \underbrace{p(I,h\,|\,\theta)}_{\text{Joint}} = \underbrace{p(I\,|\,h,\theta)}_{\text{Likelihood}}\underbrace{p(h\,|\,\theta)}_{\text{Prior}}$$

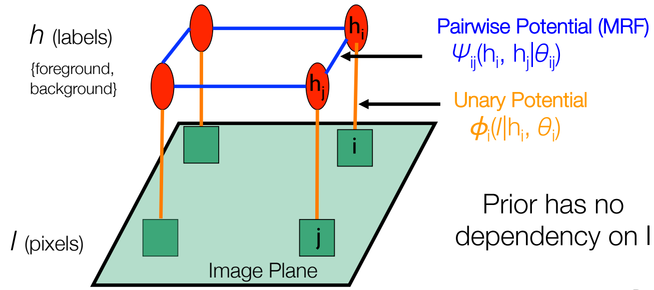
1. Generative approach models joint
   → Markov random field (MRF)

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

R. Fergus

Generative Markov Random Field

$$p(h,I\,|\,\theta) = \boxed{p(I\,|\,h,\theta)}\boxed{p(h\,|\,\theta)}$$
$$= \frac{1}{Z(\theta)}\left[\underbrace{\prod_i \phi_i(I\,|\,h_i,\theta_i)}_{\text{Likelihood}}\underbrace{\prod_{ij}\psi_{ij}(h_i,h_j\,|\,\theta_{ij})}_{\text{MRF Prior}}\right]$$



Prior has no dependency on I

R. Fergus

Conditional Random Field

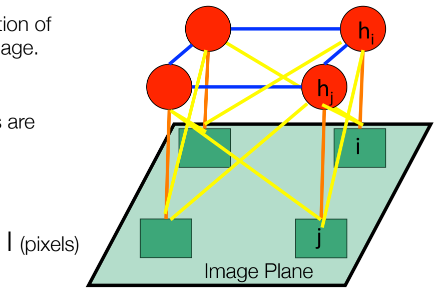
Discriminative approach

Lafferty, McCallum and Pereira 2001

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

- 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

e.g Kumar and Hebert 2003



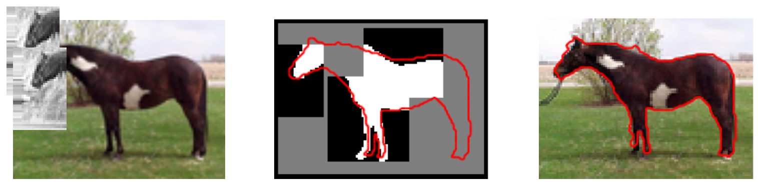
R. Fergus

Levin & Weiss [ECCV 2006]

$$E(h;I) = \sum_i \lambda_i \left|h - h_{F_i,I}\right| + \sum_{ij} w(i,j)\left|h_i - h_j\right|$$

Consistency with fragments segmentation

Segmentation alignment with image edges

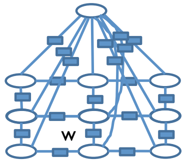


Resulting min-cut segmentation

R. Fergus

Semantic Segmentation
Joint Object recognition & segmentation

Goal: Detect and segment test image:



Large set of example segmentation:

T(1) T(2) T(3)

Up to 2.000.000 shape templates

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

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

"Hamming distance"

[Lempitsky et al. ECCV '08]

C. Rother

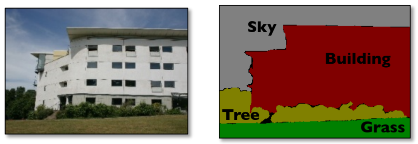
Semantic Segmentation
Joint Object recognition & segmentation

UIUC dataset; 98.8% accuracy

[Lempitsky et al. ECCV '08]

C. Rother

Semantic Segmentation
Joint Object recognition & segmentation

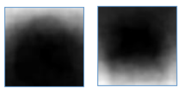


$$E(x,\omega) = \sum_i \theta_i(\omega, x_i) + \sum_i \theta_i(x_i) + \sum_i \theta_i(x_i) + \sum_{i,j} \theta_{ij}(x_i,x_j)$$

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

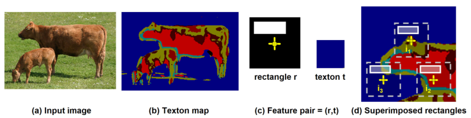
$x_i \in \{1,\ldots,K\}$ for K object classes

Location



sky grass

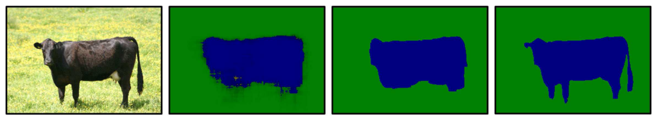
Class (boosted textons)



[TextonBoost; Shotton et al, '06]

C. Rother

Semantic Segmentation
Joint Object recognition & segmentation



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

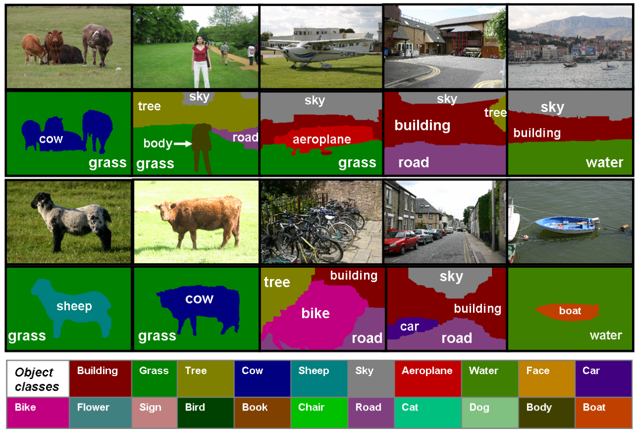
Class+ location | + edges | + color

[TextonBoost; Shotton et al, '06]

C. Rother

Semantic Segmentation
Joint Object recognition & segmentation

Good results …



[TextonBoost; Shotton et al, '06]

C. Rother

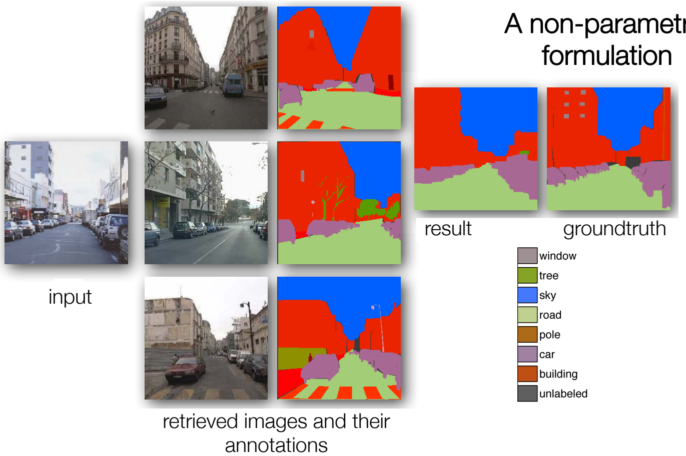
Semantic Segmentation
Joint Object recognition & segmentation

Failure cases…

C. Rother

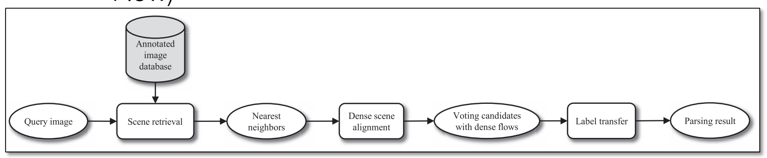
Nonparametric Scene Parsing via Label Transfer (Liu et al. TPAMI'12)

A non-parametric formulation



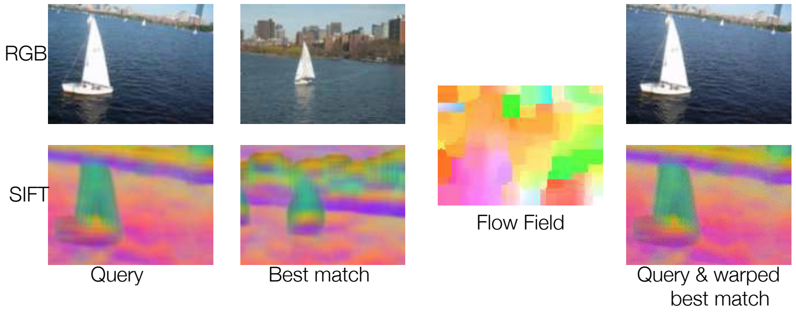
Nonparametric Scene Parsing via Label Transfer

- Framework consists of three main modules:
  1. <u>Scene retrieval:</u> finding nearest neighbors (k-NN approach)
  2. <u>Dense scene alignment:</u> 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



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

$$E(\mathbf{w}) = \sum_{\mathbf{p}} \min(\|s_1(\mathbf{p}) - s_2(\mathbf{p}+\mathbf{w}(\mathbf{p}))\|_1, t) +$$

data term

$$\sum_{\mathbf{p}} \eta(|u(\mathbf{p})| + |v(\mathbf{p})|) +$$

small displacement term

$$\sum_{(\mathbf{p},\mathbf{q})\in\varepsilon} \min(\lambda|u(\mathbf{p}) - u(\mathbf{q})|, d) + \min(\lambda|v(\mathbf{p}) - v(\mathbf{q})|, d),$$

smoothness term

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

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

$$-\log P\big(c|I, s, \{s_i, c_i, \mathbf{w}_i\}\big) = \sum_{\mathbf{p}} \psi\big(c(\mathbf{p}); s, \{s_i'\}\big)$$

$$+\,\alpha \sum_{\mathbf{p}} \lambda\big(c(\mathbf{p})\big) + \beta \sum_{\{\mathbf{p},\mathbf{q}\}\in\varepsilon} \phi\big(c(\mathbf{p}), c(\mathbf{q}); I\big) + \log Z$$

Label Transfer

- Likelihood term:

$$\psi\big(c(\mathbf{p}) = l\big) = \begin{cases} \min\limits_{i\in\Omega_{\mathbf{p},l}} \|s(\mathbf{p}) - s_i(\mathbf{p}+\mathbf{w}(\mathbf{p}))\|, & \Omega_{\mathbf{p},l} \neq \emptyset, \\ \tau, & \Omega_{\mathbf{p},l} = \emptyset, \end{cases}$$

- $\Omega_{\mathbf{p},l} = \{i; c_i(\mathbf{p}+\mathbf{w}(\mathbf{p})) = l\}$ where *l=1,...,L* indicates the index set of the voting candidates whose label is *l* after being warped to pixel p.
- τ is set to be the value of the maximum difference of SIFT feature: $\tau = \max_{s_1,s_2,\mathbf{p}} \|s_1(\mathbf{p}) - s_2(\mathbf{p})\|$

Label Transfer

- Prior term :

$$\lambda\big(c(\mathbf{p})=l\big) = -\log \mathrm{hist}_l(\mathbf{p})$$

- 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

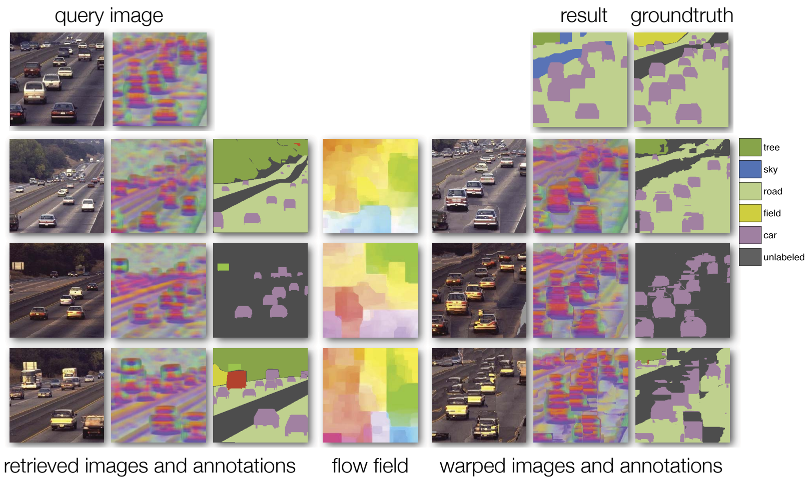
Label Transfer

- Spatial smoothness term:

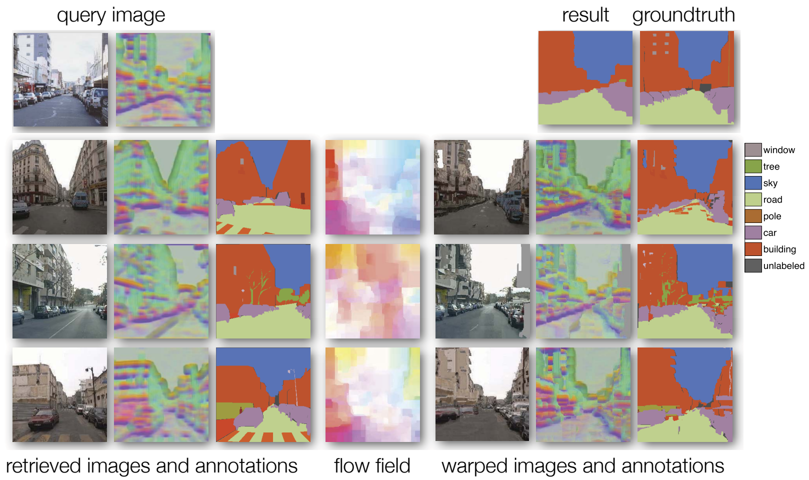
$$\phi\big(c(\mathbf{p}), c(\mathbf{q})\big) = \delta[c(\mathbf{p}) \neq c(\mathbf{q})] \left(\frac{\xi + e^{-\gamma \|I(\mathbf{p}) - I(\mathbf{q})\|^2}}{\xi + 1} \right)$$

- 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



Parsing Results



Parsing Results