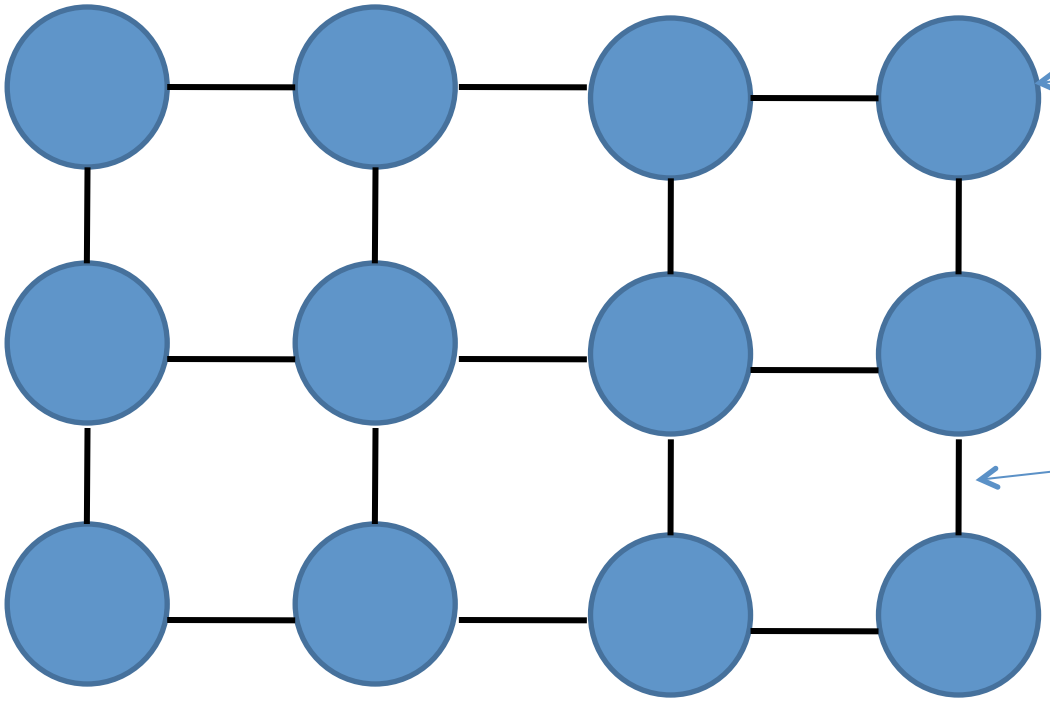
BIL 717
Image Processing

Erkut Erdem
Dept. of Computer Engineering
Hacettepe University

Semantic Segmentation

Review - Markov Random Fields



- 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

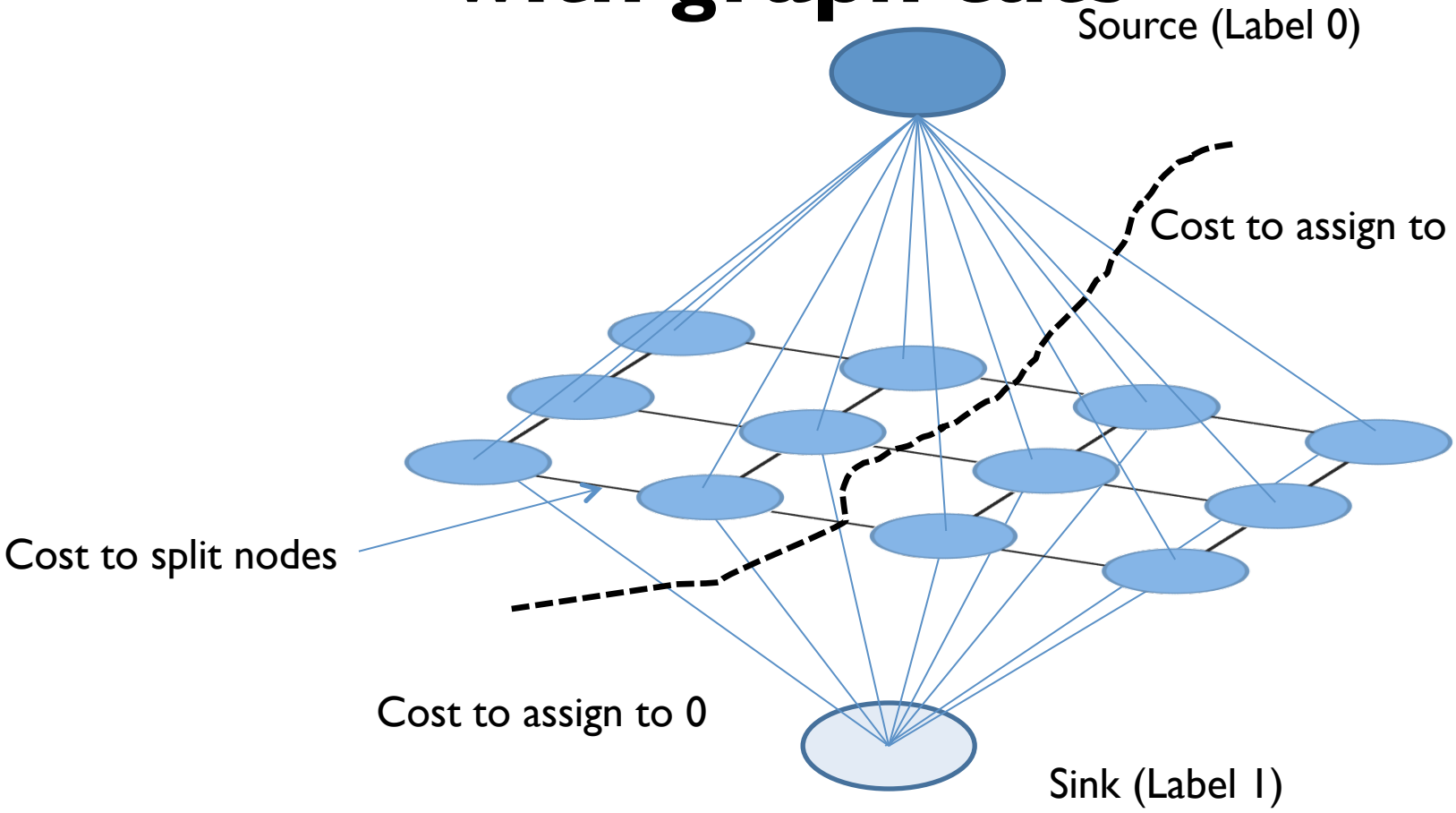
Review - Solving MRFs with graph cuts

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, $\mathbf{y}^\star = \text{argmin}_{\mathbf{y}}\ E(\mathbf{y}; \text{data})$.

* Requires non-negative energies

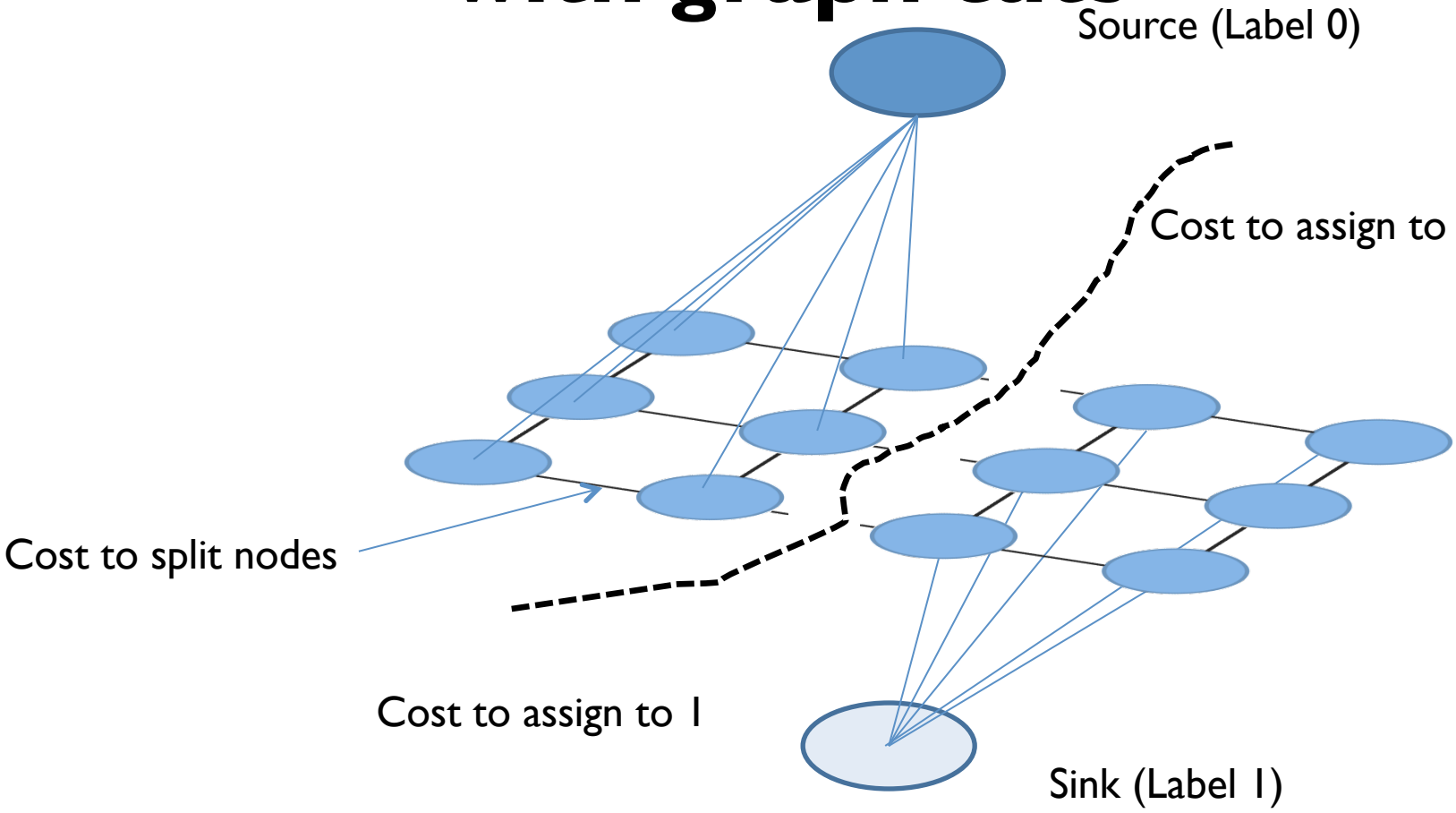
Review - Solving MRFs with graph cuts



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

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)?

Review - How does the code look like?

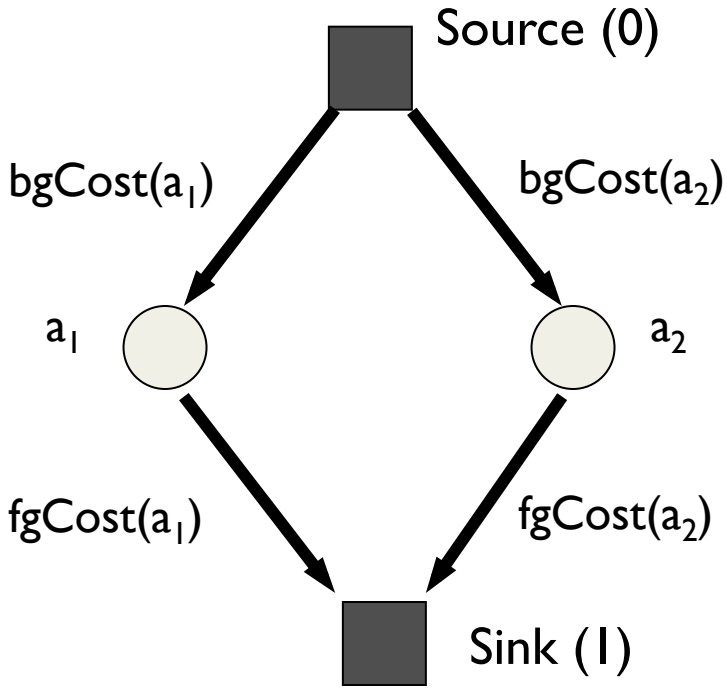
```
Graph *g;

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)
```

Source (0)

Sink (1)

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)
```

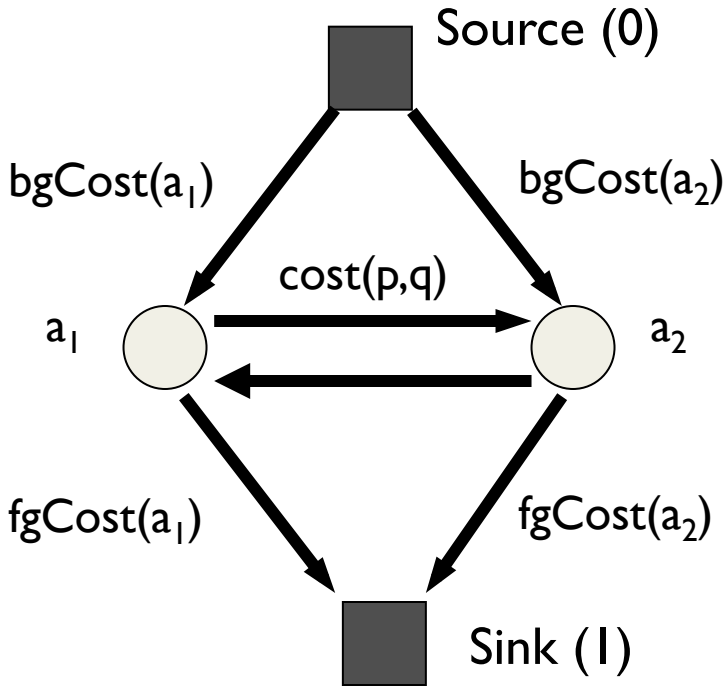


Review - How does the code look like?

```
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
```

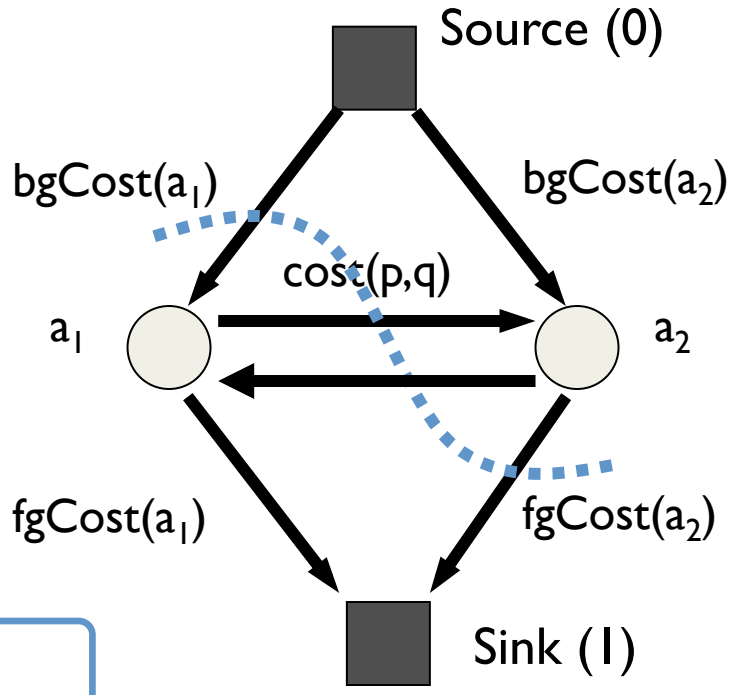
```
g->compute_maxflow();

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



Review - How does the code look like?

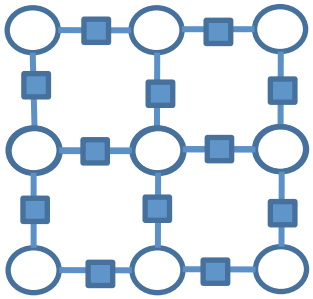
```
Graph *g;

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

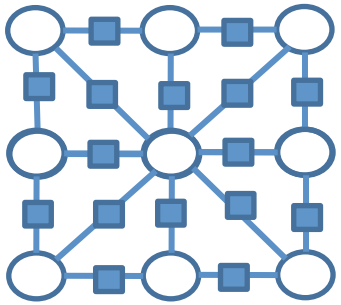
Review - Random Fields in Vision



4-connected;
pairwise MRF

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

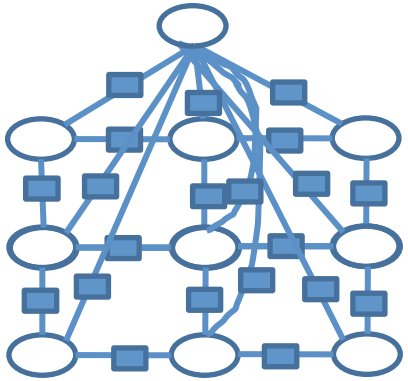
Order 2



higher(8)-connected;
pairwise MRF

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

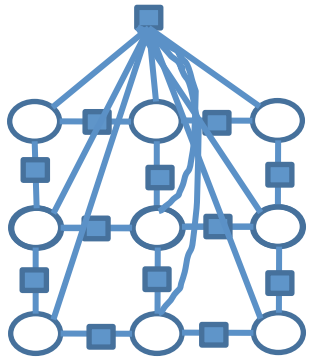
Order 2



MRF with
global variables

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

Order 2



Higher-order MRF

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

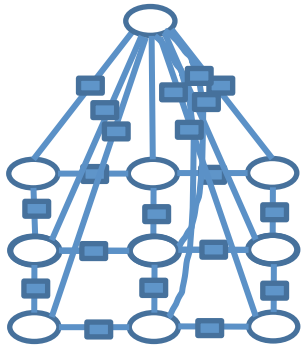
Order n

C. Rother

Review - MRF with global potential

GrabCut model [Rother et. al. '04]

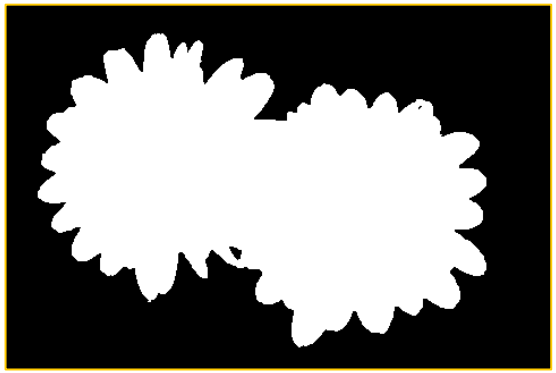
$\theta^{F/B}$



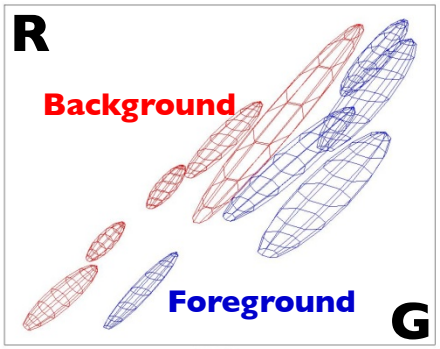
$$E(x, \theta^F, \theta^B) = \sum_i F_i(\theta^F)x_i + B_i(\theta^B)(1-x_i) + \sum_{i,j \in N} |x_i - x_j|$$

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

Image z



Output x



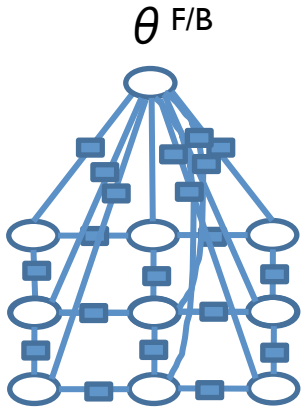
$\theta^{F/B}$ Gaussian Mixture models

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]



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

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

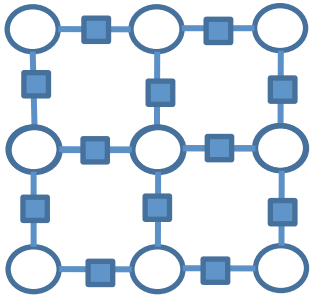
Learning of the colour distributions

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

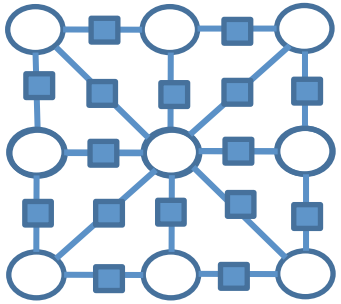
Review - Random Fields in Vision



4-connected;
pairwise MRF

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

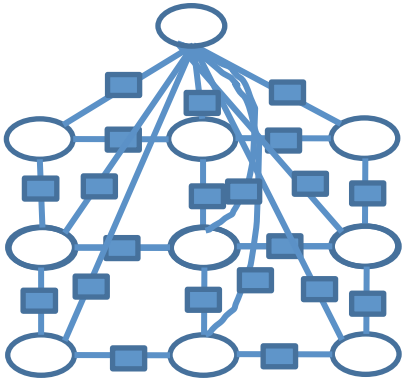
Order 2



higher(8)-connected;
pairwise MRF

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

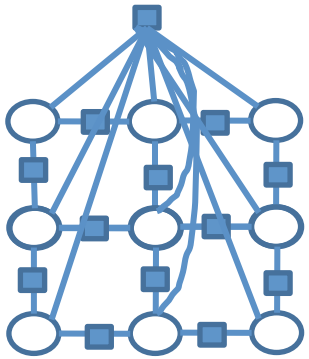
Order 2



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]

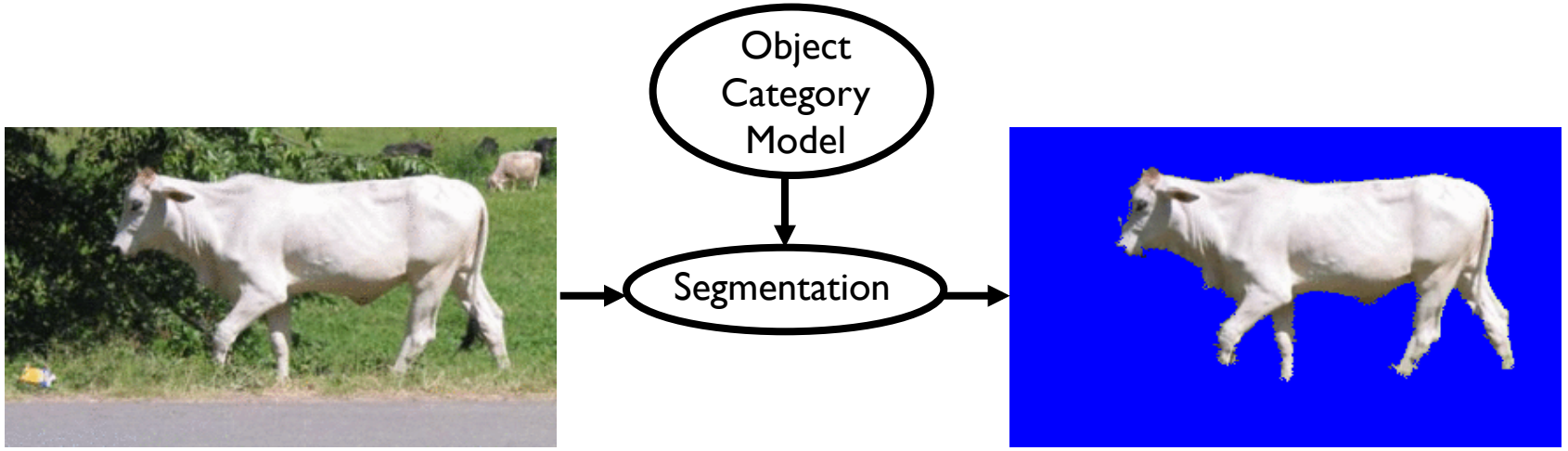
Semantic Segmentation

- 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

Early Studies of Semantic Segmentation

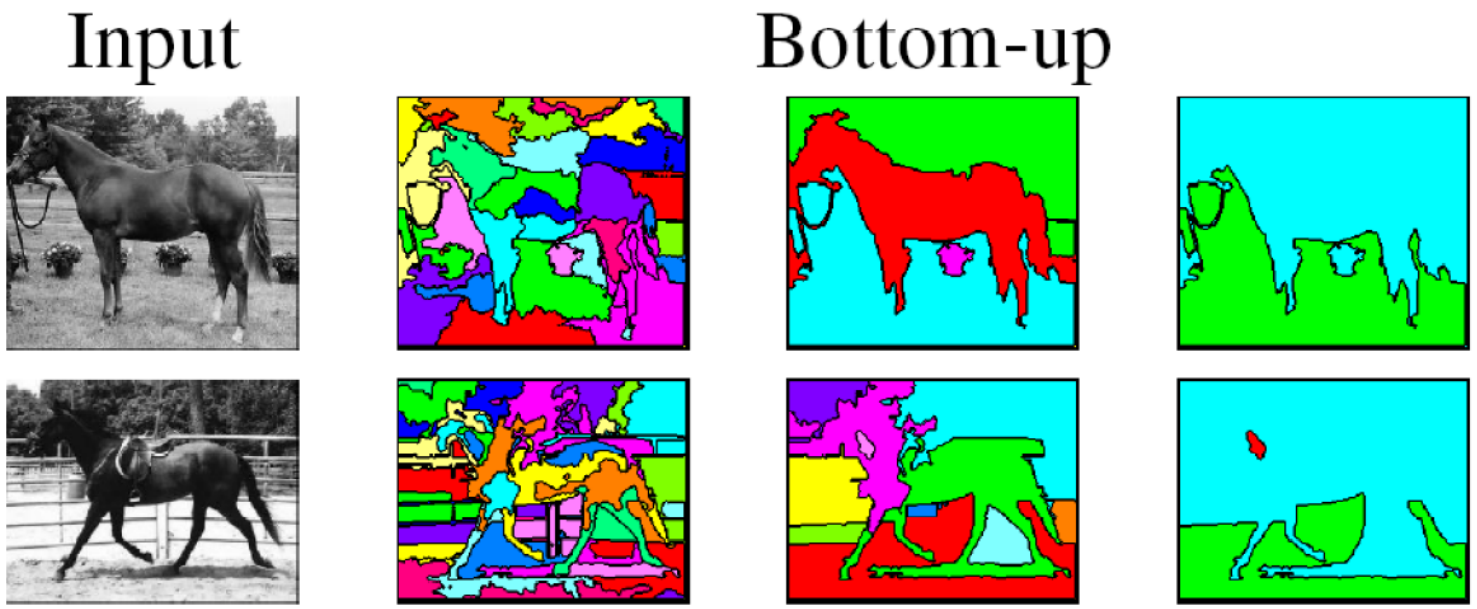
R. Fergus

Early Studies of Semantic Segmentation

R. Fergus

Early Studies of Semantic Segmentation

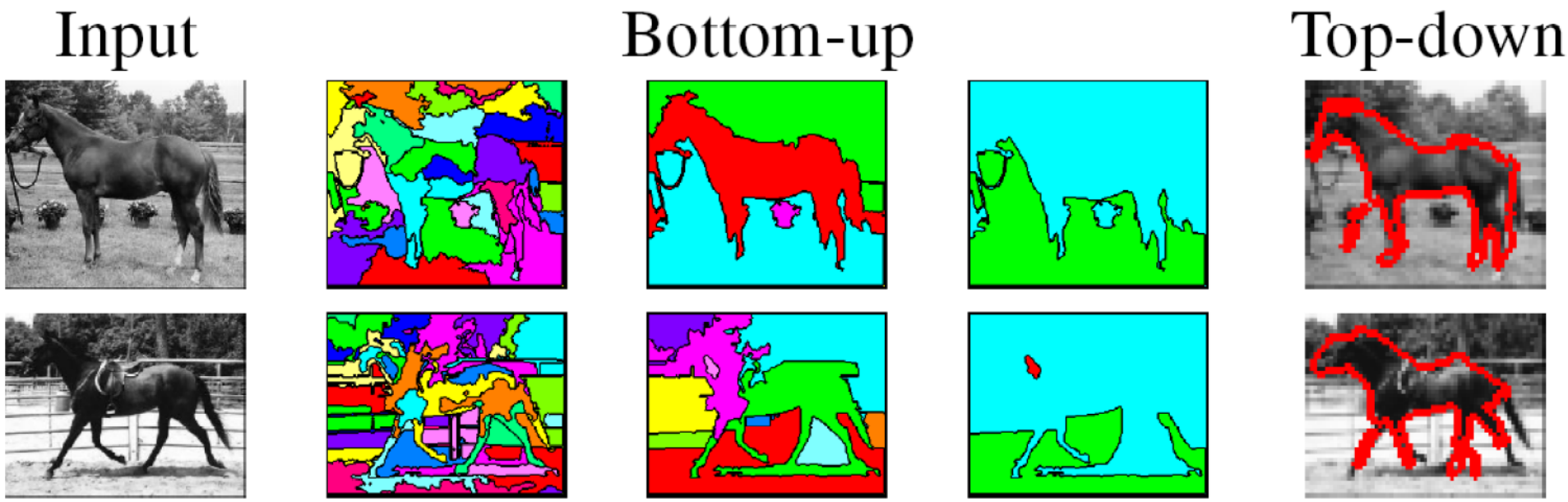
Using Normalized Cuts, Shi & Malik, 1997



R. Fergus

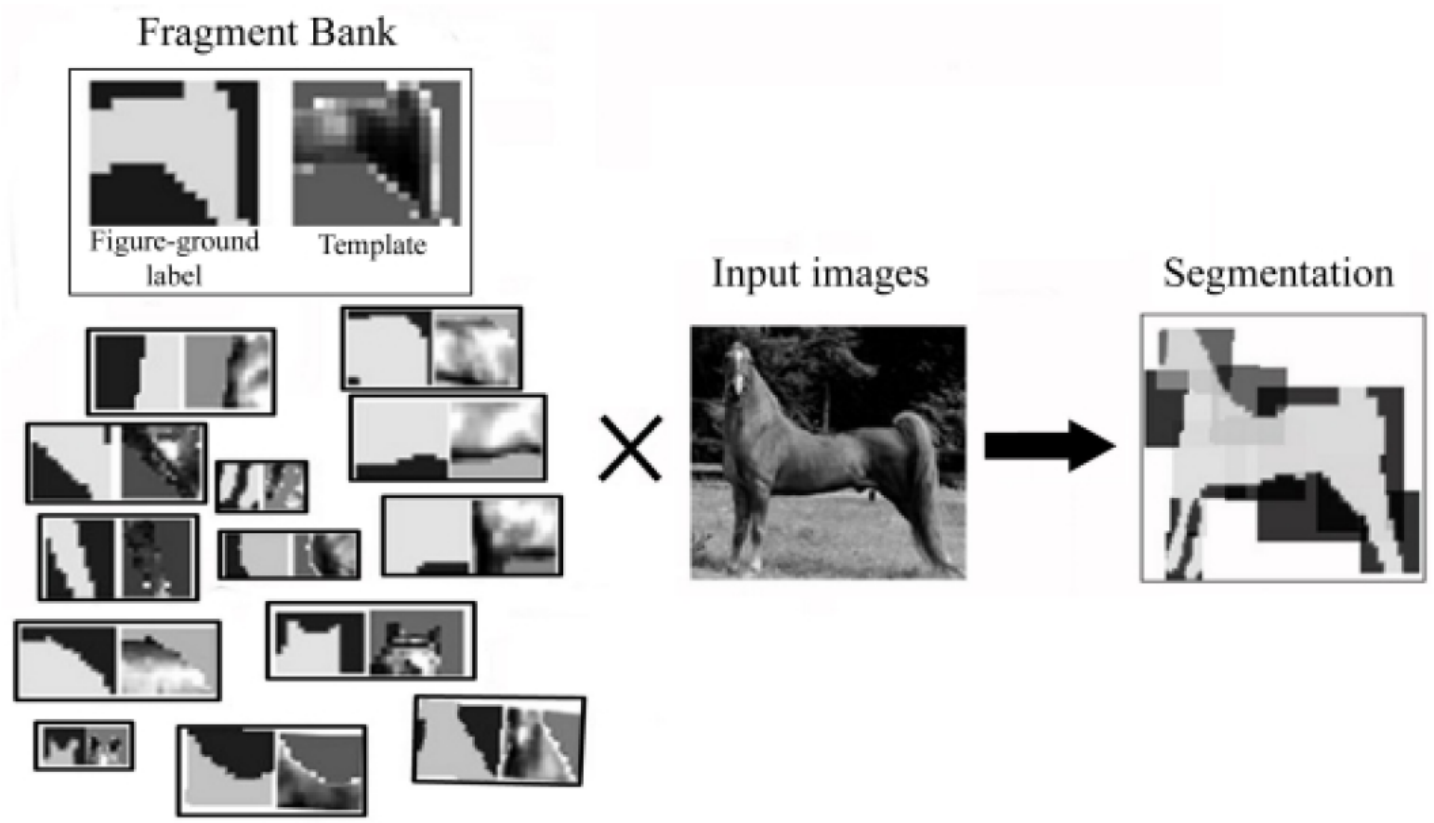
Early Studies of Semantic Segmentation

Using Normalized Cuts, Shi & Malik, 1997



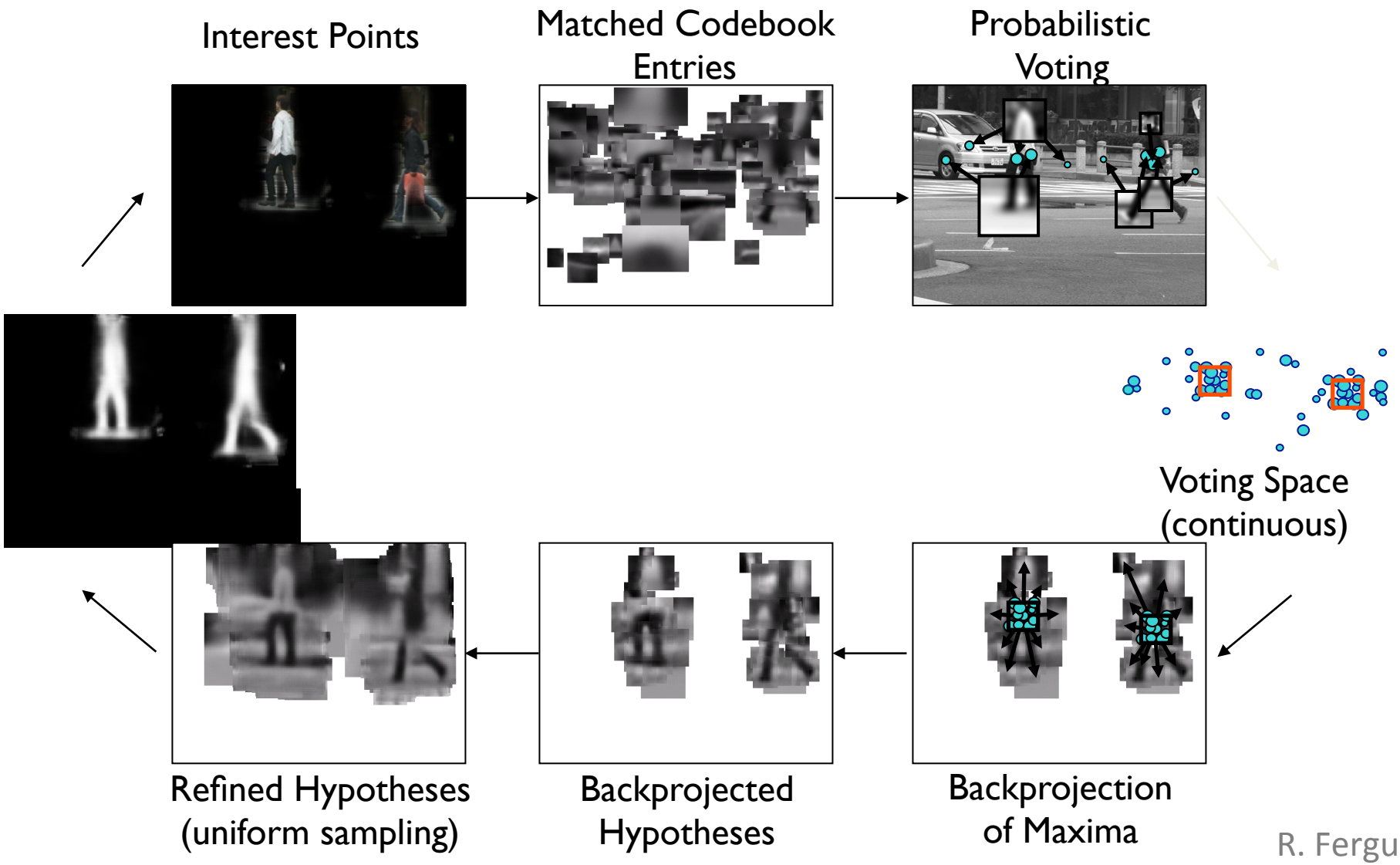
Borenstein and Ullman, ECCV 2002

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

$$\underbrace{p(h \mid I, \theta)}$$

Posterior

Random Fields for segmentation

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

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

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

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

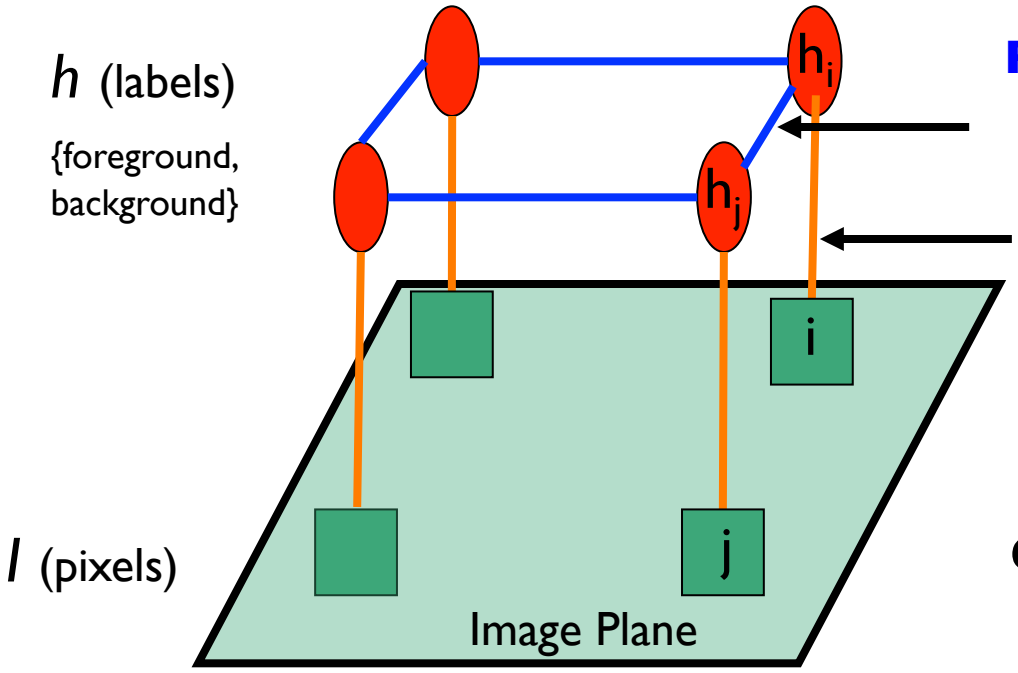
R. Fergus

Generative Markov Random Field

$$p(h, I \mid \theta) = p(I \mid h, \theta)\, p(h \mid \theta)$$

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

Likelihood **MRF Prior**



Pairwise Potential (MRF)

$\psi_{ij}(h_i, h_j \mid \theta_{ij})$

Unary Potential

$\phi_i(I \mid h_i, \theta_i)$

Prior has no dependency on I

R. Fergus

Conditional Random Field

Discriminative approach

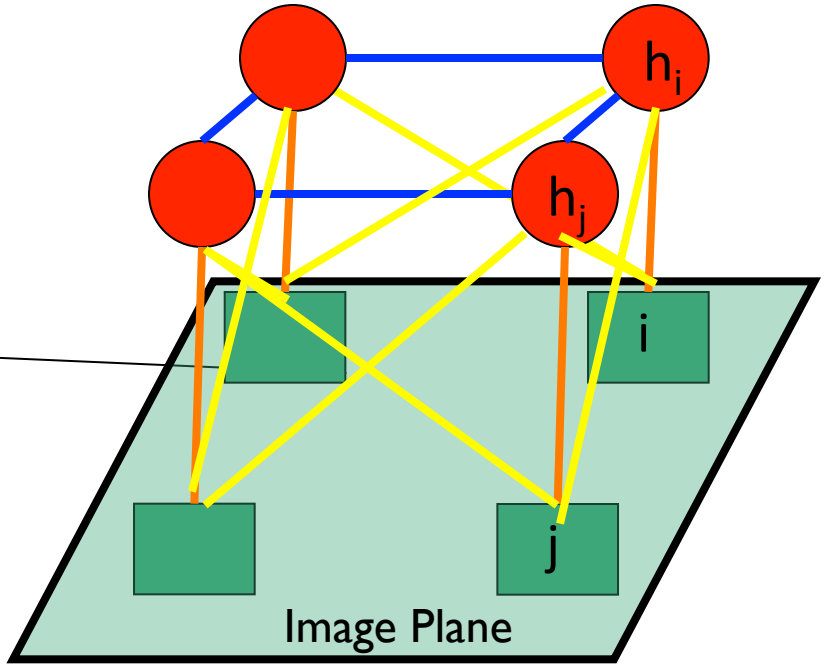
Lafferty, McCallum and Pereira 2001

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

- 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



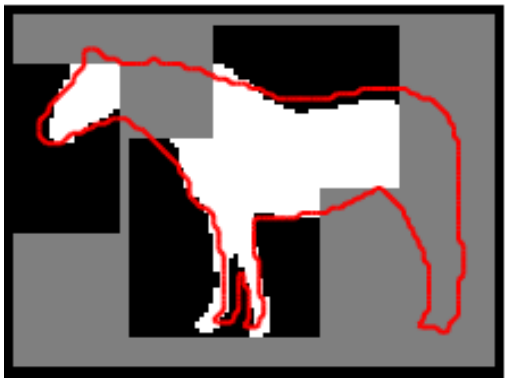
R. Fergus

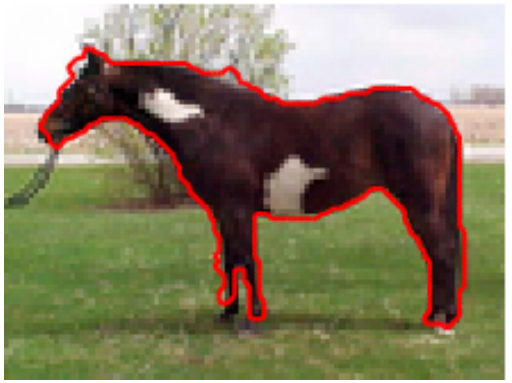
Levin & Weiss [ECCV 2006]

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

Consistency with fragments segmentation

Segmentation alignment with image edges

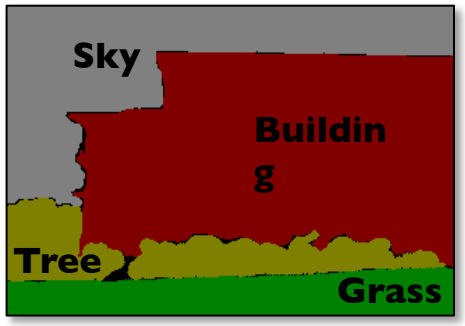




Resulting min-cut segmentation

R. Fergus

Semantic Segmentation
Joint Object recognition & segmentation

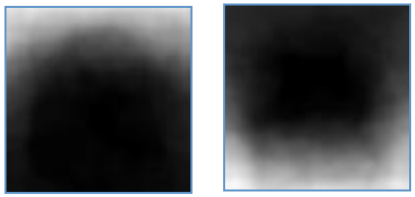


$$E(x,\omega) = \sum_{i} \theta_i(\omega, x_i) + \sum_{i} \theta_i(x_i) + \sum_{i} \theta_i(x_i) + \sum_{i,j} \theta_{ij}(x_i,x_j)$$

$\theta_i(\omega, x_i)$: (color) $\theta_i(x_i)$: (location) $\theta_i(x_i)$: (class) $\theta_{ij}(x_i,x_j)$: (edge aware Ising prior)

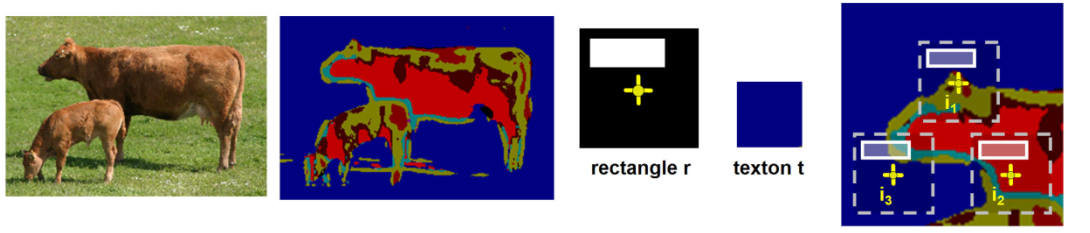
$x_i \in \{1,\ldots,K\}$ for K object classes

Location



sky grass

Class (boosted textons)

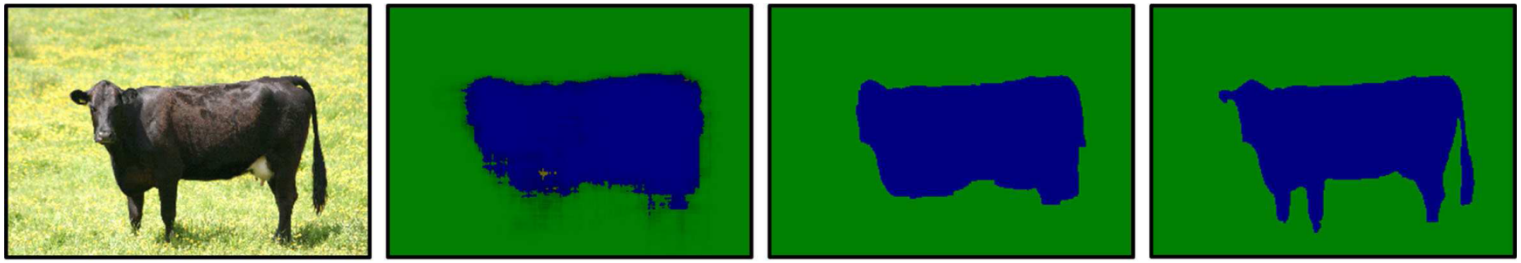


(a) Input image (b) Texton map (c) Feature pair = (r,t) (d) Superimposed rectangles

[TextonBoost; Shotton et al, '06]

C. Rother

Semantic Segmentation
Joint Object recognition & segmentation



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

Class+ location

+ edges

+ color

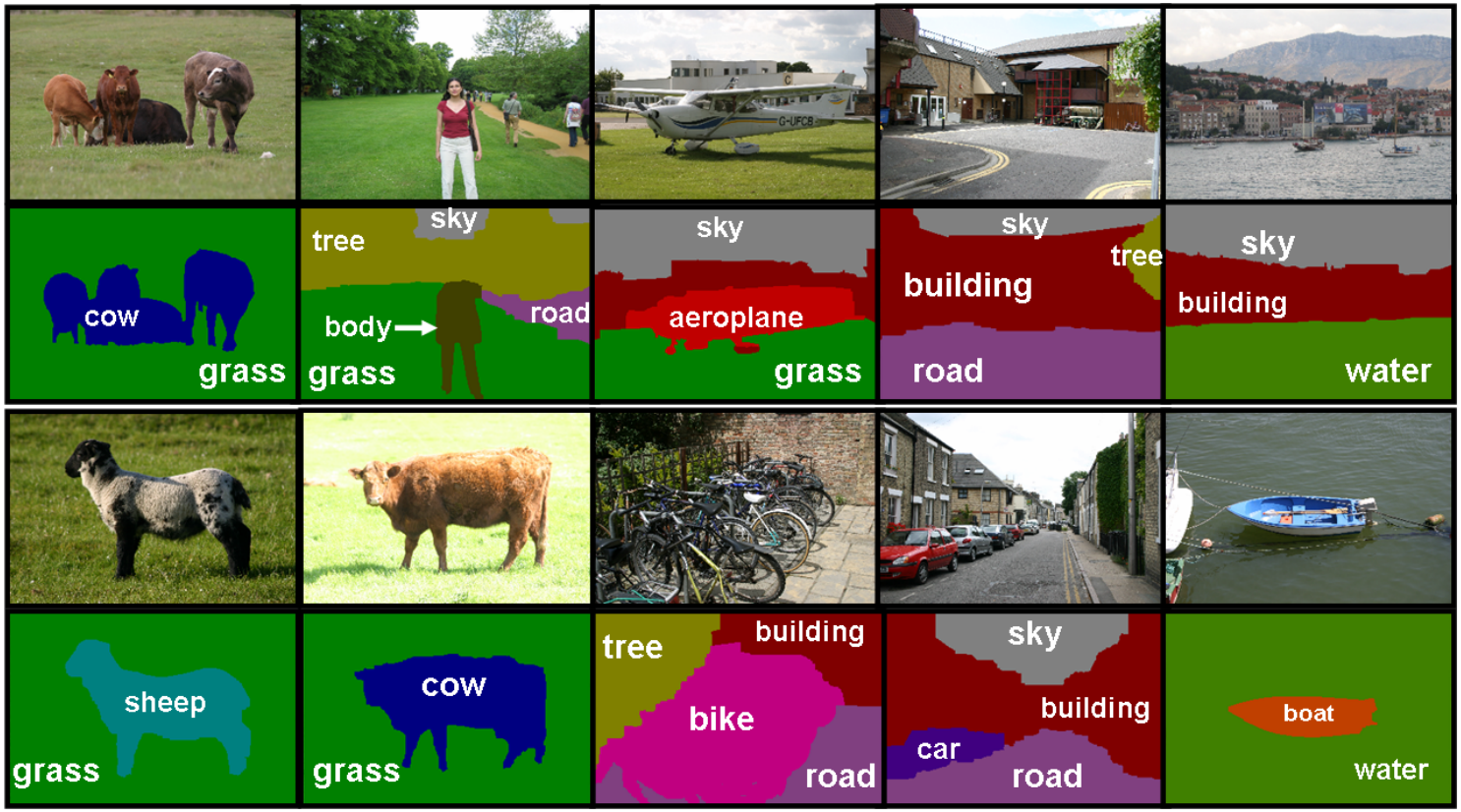
[TextonBoost; Shotton et al, '06]

C. Rother

Semantic Segmentation

Joint Object recognition & segmentation

Good results …



Object classes	Building	Grass	Tree	Cow	Sheep	Sky	Aeroplane	Water	Face	Car
Bike	Flower	Sign	Bird	Book	Chair	Road	Cat	Dog	Body	Boat

[TextonBoost; Shotton et al, '06]

C. Rother

Semantic Segmentation
Joint Object recognition & segmentation

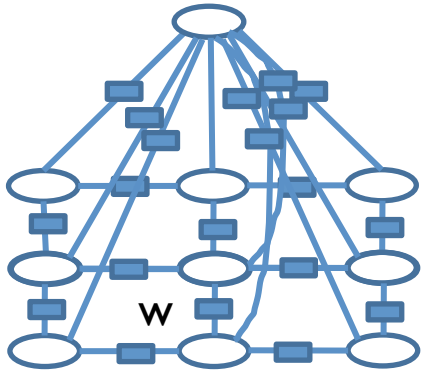
Failure cases…

C. Rother

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

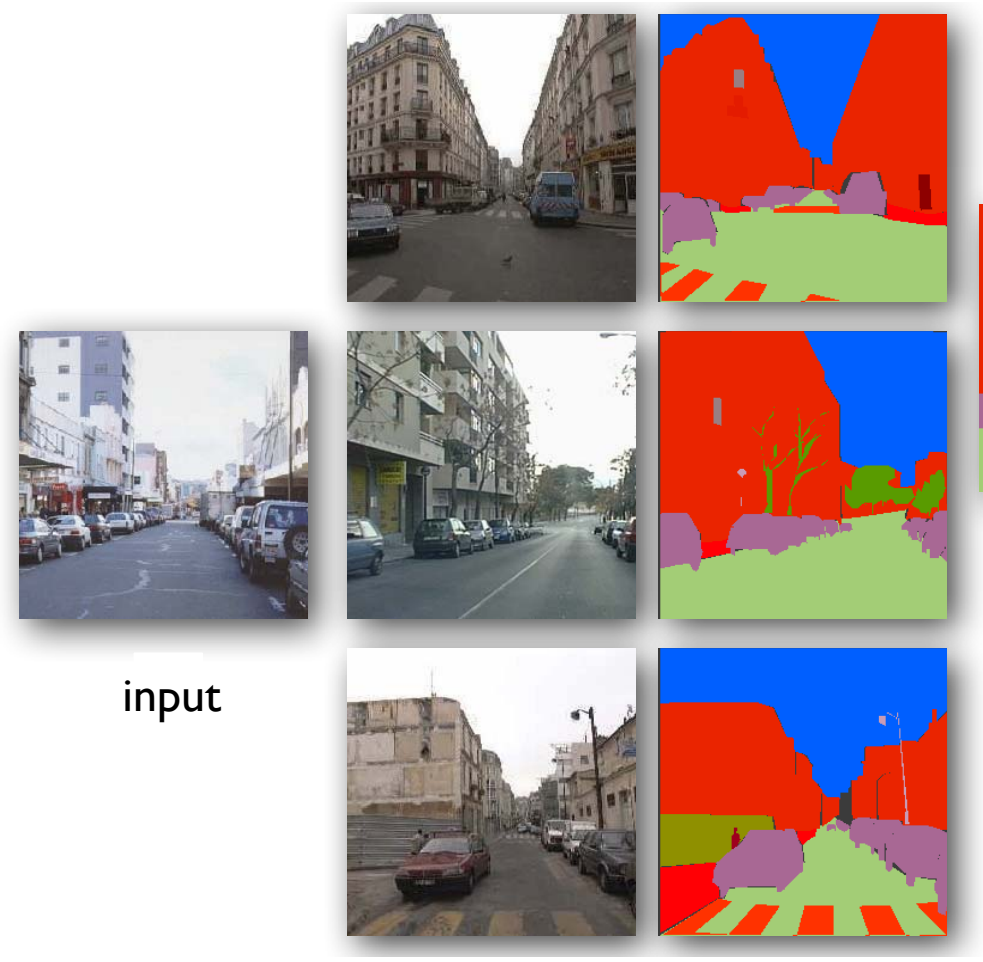
Joint Object recognition & segmentation

UIUC dataset; 98.8% accuracy

[Lempitsky et al. ECCV '08]

C. Rother

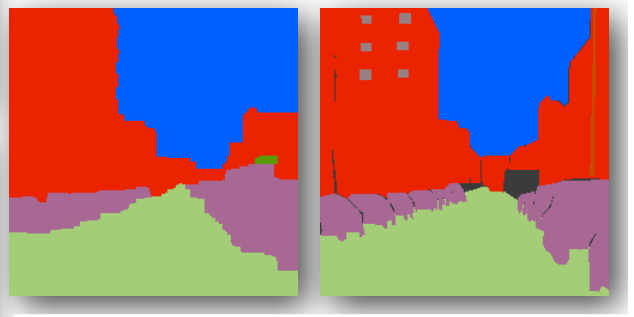
Nonparametric Scene Parsing via Label Transfer (Liu et al. TPAMI'12)



input

retrieved images and their annotations

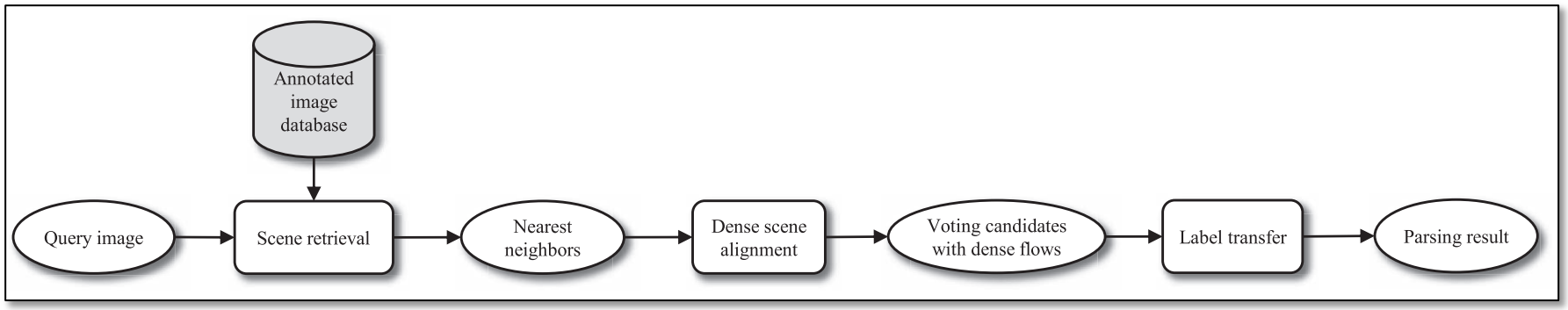
A non-parametric formulation



result groundtruth

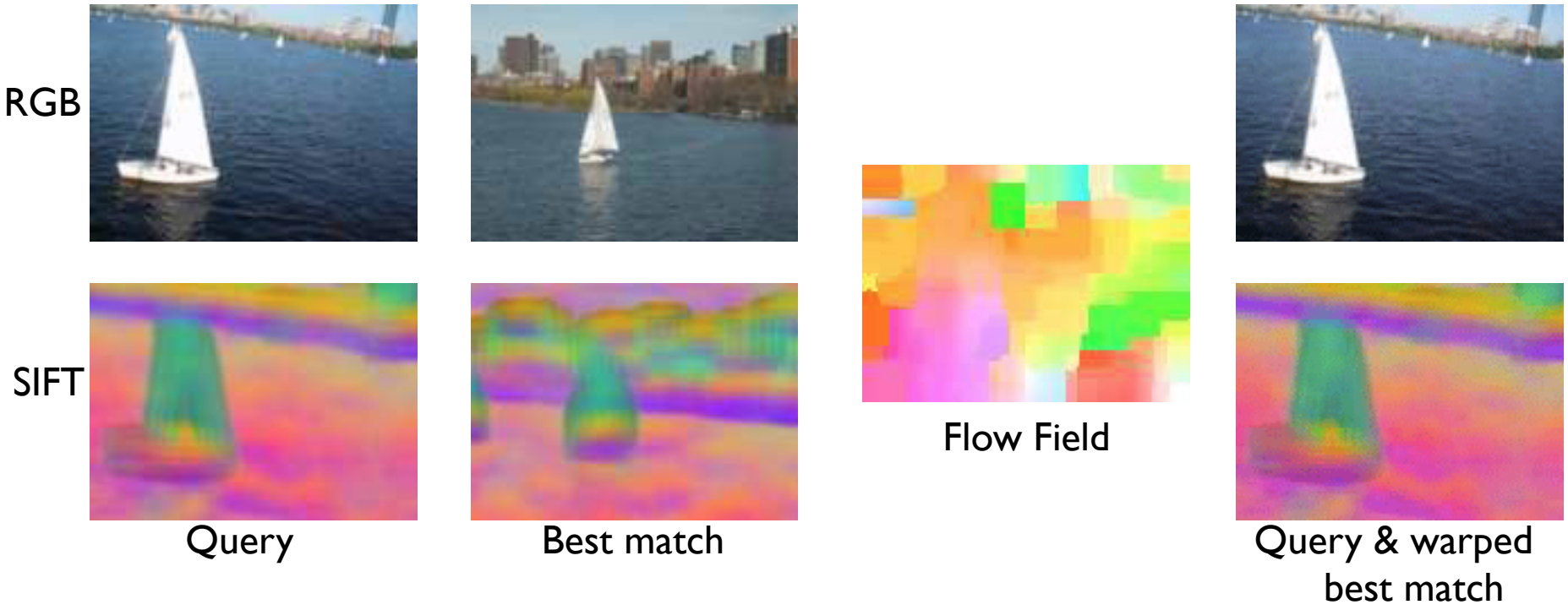
Nonparametric Scene Parsing via Label Transfer

- Framework consists of three main modules:
  1. Scene retrieval: finding nearest neighbors (k-NN approach)
  2. Dense scene alignment: dense scene matching (SIFT Flow)
  3. Label transfer: using a MRF model to label input image



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 ith 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

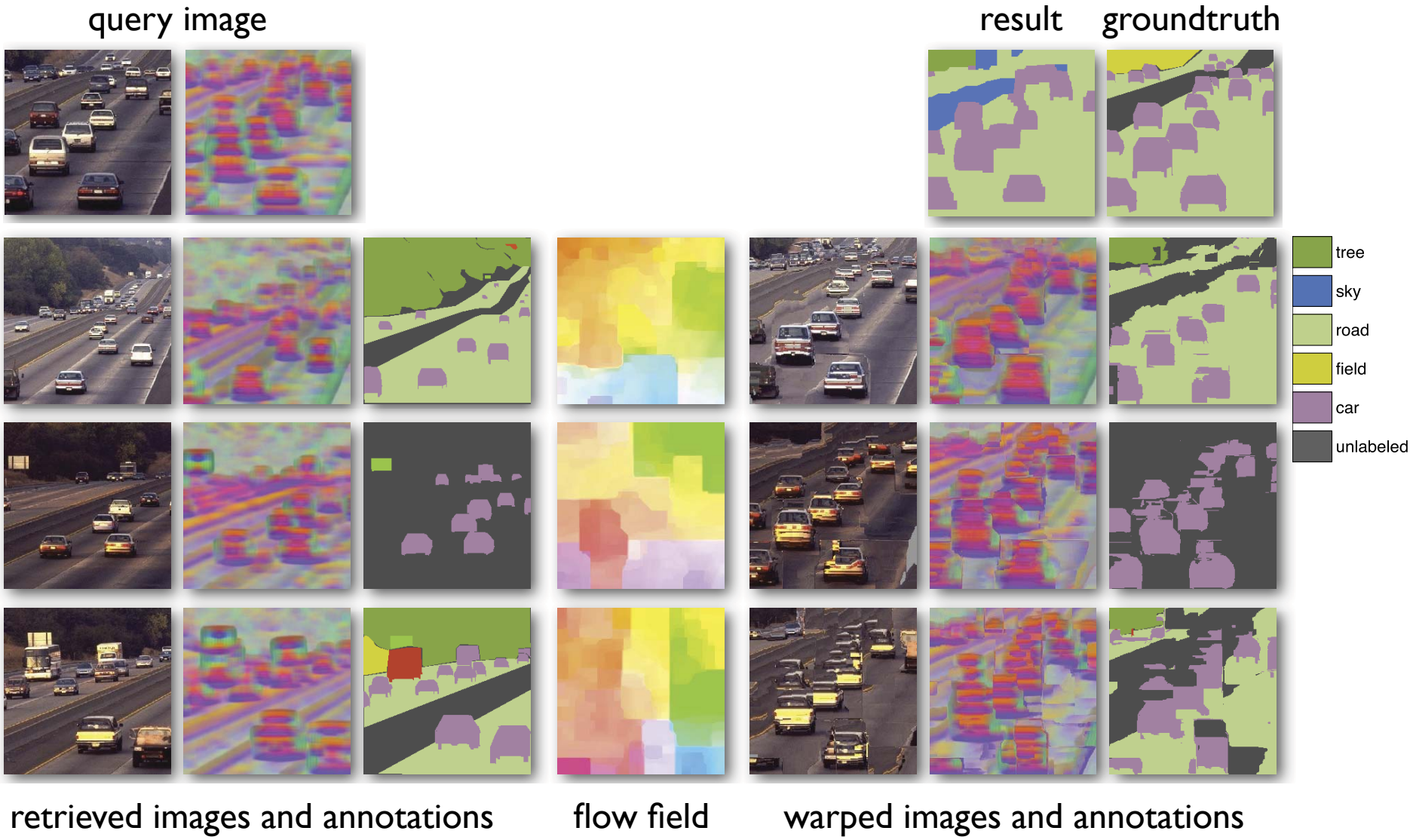
Label Transfer

- Spatial smoothness term:

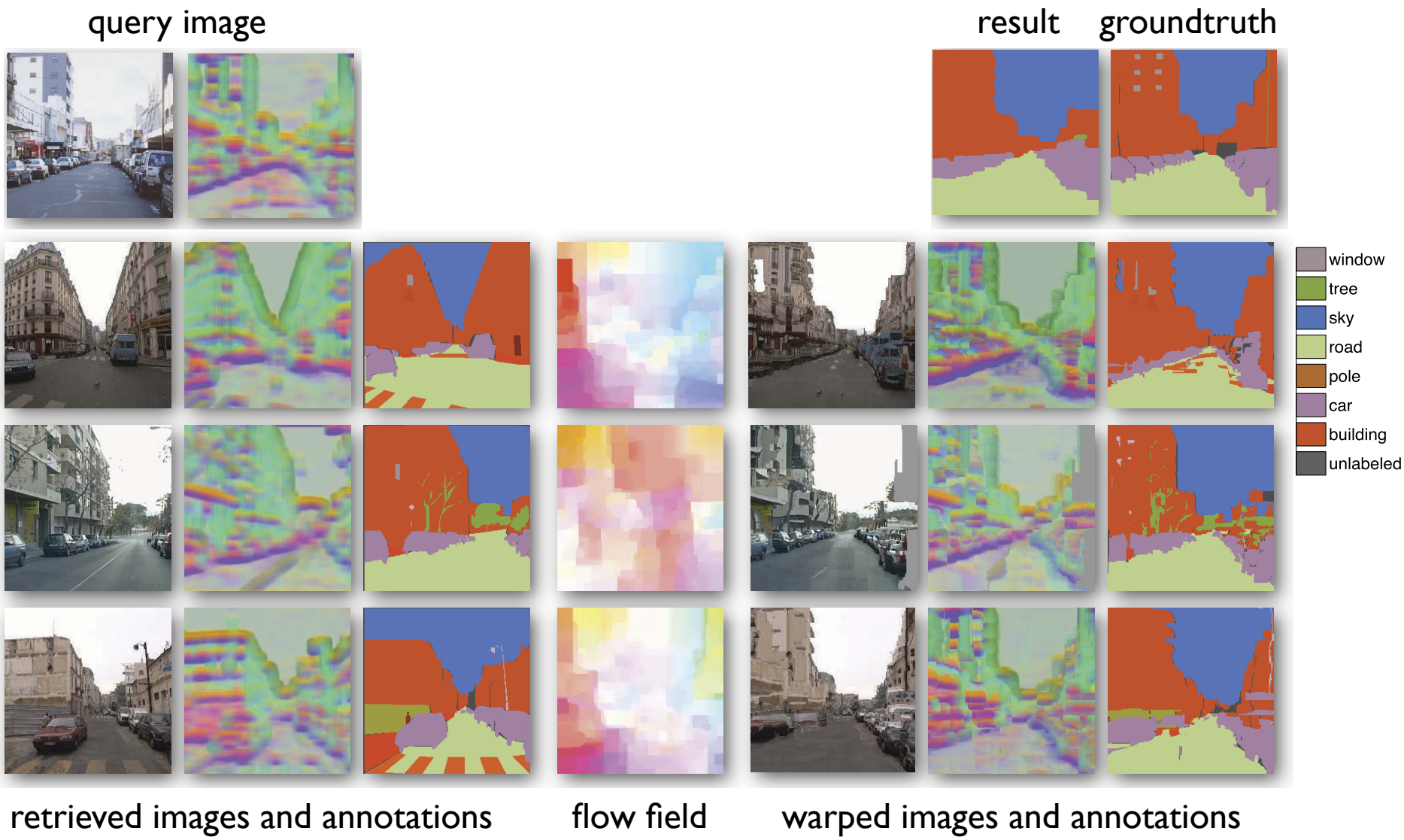
$$\phi\big(c(\mathbf{p}), c(\mathbf{q})\big) = \delta[c(\mathbf{p}) \neq c(\mathbf{q})]\left(\frac{\xi + e^{-\gamma\|I(\mathbf{p})-I(\mathbf{q})\|^2}}{\xi + 1}\right)$$

- 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.

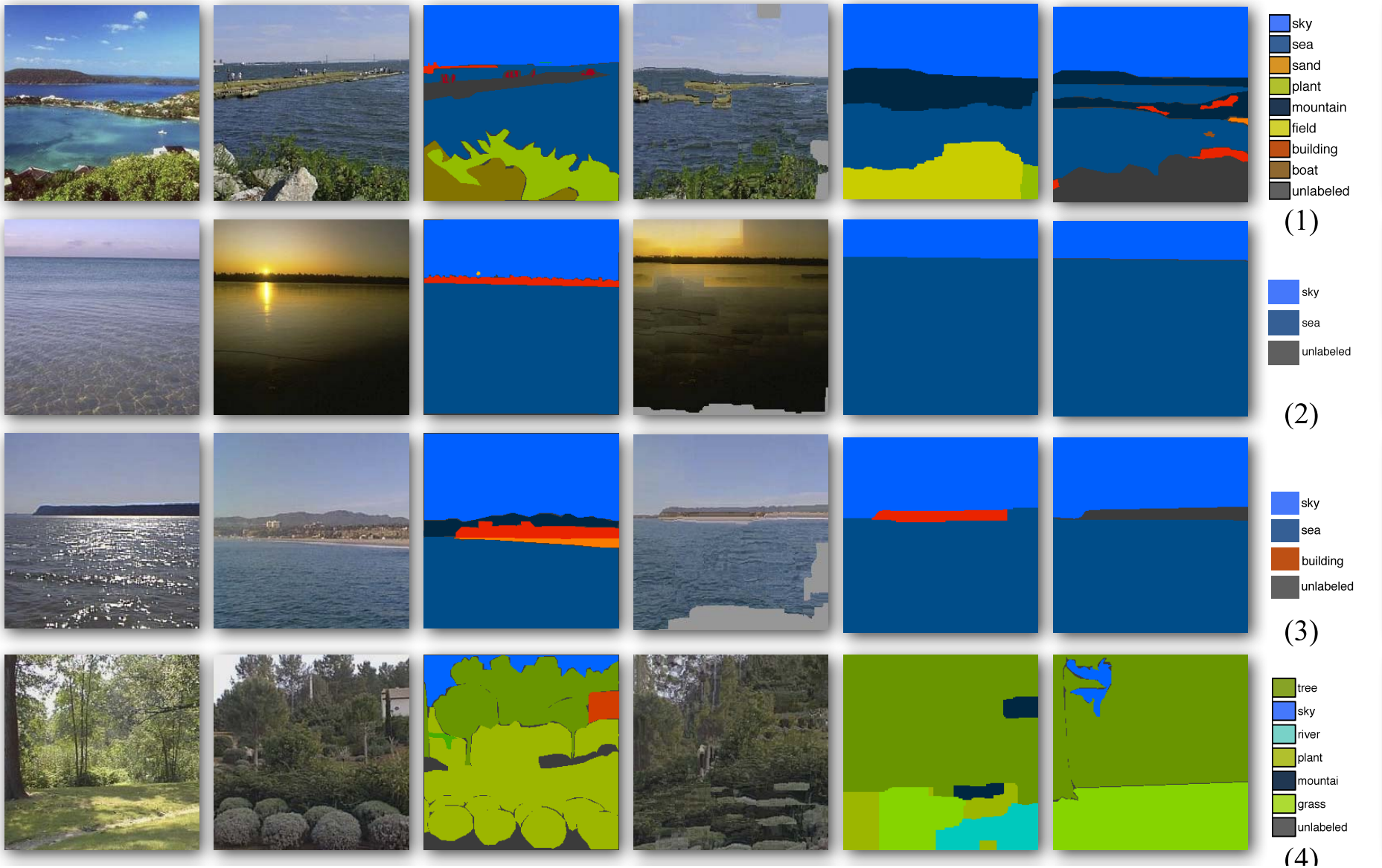
Parsing Results



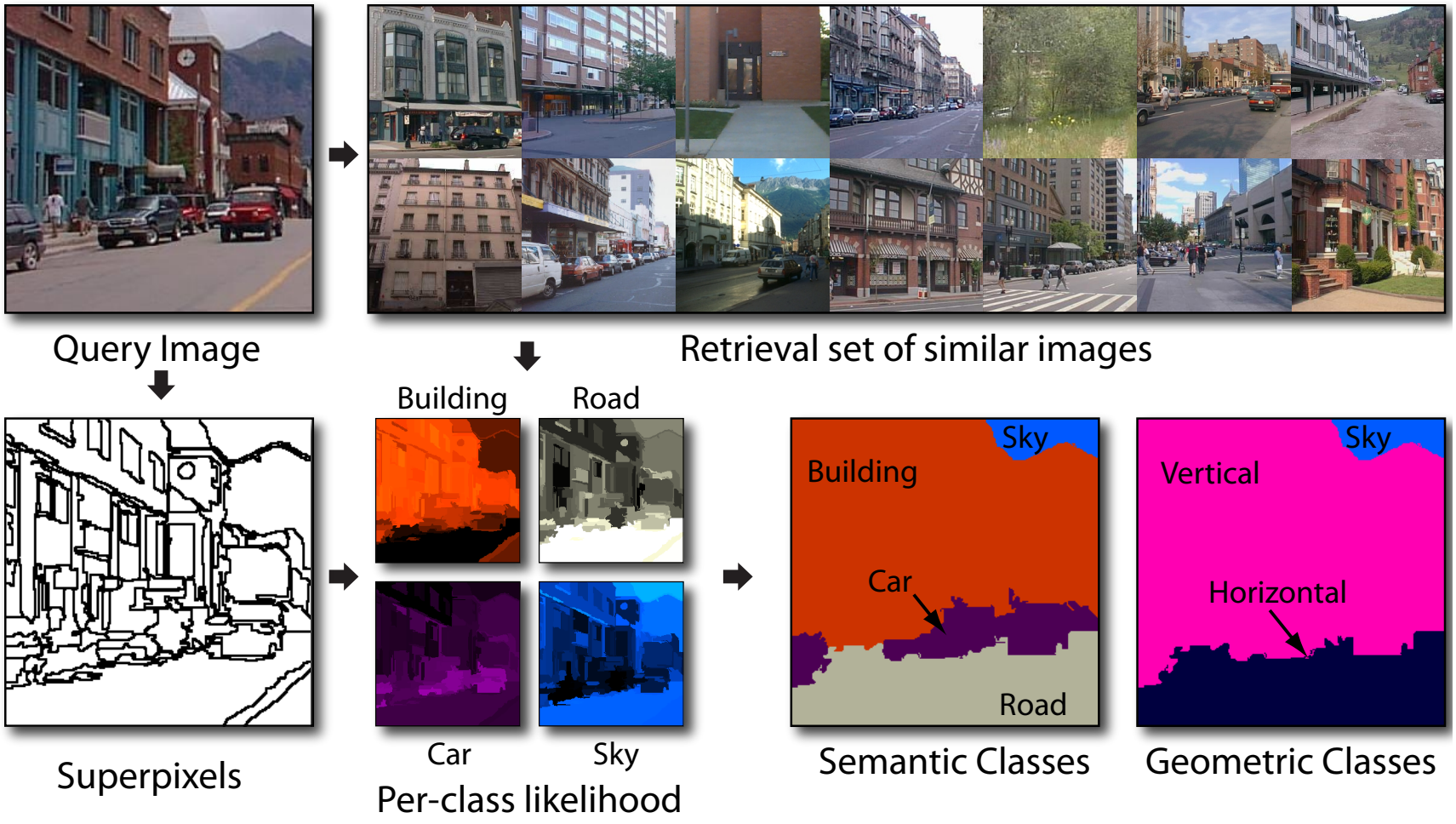
Parsing Results



Parsing Results



Superparsing: Scalable Nonparametric Image Parsing with Superpixels (Tighe & Lazebnik, ECCV 2010)



Superparsing: Scalable Nonparametric Image Parsing with Superpixels

- Framework consists of three main modules:
  1. Scene retrieval: finding nearest neighbor images (k-NN approach)
  2. Superpixel segmentation: segmenting query image into superpixels
  3. Superpixel retrieval: finding nearest neighbors superpixels
  4. Labeling: using a MRF model to label input image

Scene Retrieval

- Find a relatively small retrieval set of training images that will serve as the source of candidate superpixel-level matches.

(a) Global features for retrieval set computation (Section 2.1)		
Type	Name	Dimension
Global	Spatial pyramid (3 levels, SIFT dictionary of size 200)	4200
	Gist (3-channel RGB, 3 scales with 8, 8, & 4 orientations)	960
	Color histogram (3-channel RGB, 8 bins per channel)	24

Superpixel Segmentation

- superpixels using the fast graph- based segmentation algorithm of Felzenszwalb and Hut- tenlocher

(b) Superpixel features (Section 2.2)		
Shape	Mask of superpixel shape over its bounding box (8×8) Bounding box width/height relative to image width/height Superpixel area relative to the area of the image	64 2 1
Location	Mask of superpixel shape over the image Top height of bounding box relative to image height	64 1
Texture/SIFT	Texton histogram, dilated by 10 pix texton histogram Quantized SIFT histogram, dilated by 10 pix quantized SIFT histogram Left/right/top/bottom boundary quantized SIFT histogram	100×2 100×2 100×4
Color	RGB color mean and std. dev. Color histogram (RGB, 11 bins per channel), dilated by 10 pix color histogram	3×2 33×2
Appearance	Color thumbnail (8×8) Masked color thumbnail Grayscale gist over superpixel bounding box	192 192 320

Labeling

- A probabilistic MRF model is built to integrate
  - multiple category labels,
  - contextual relations between category labels

$$J(\mathbf{c}) = \sum_{s_i \in SP} E_{\text{data}}(s_i, c_i) + \lambda \sum_{(s_i, s_j) \in A} E_{\text{smooth}}(c_i, c_j)$$

- Superpixel-based labeling as opposed to pixel-level one.

Data Term

- Likelihood term:

$$E_{\text{data}}(s_i, c_i) = -w_i \sigma(L(s_i, c_i))$$

with $\sigma(t) = \exp(\gamma t)/(1 + \exp(\gamma t))$ being a sigmoid function.

- constructed from the superpixel matches:

$$L(s_i, c) = \log \frac{P(s_i|c)}{P(s_i|\bar{c})} = \log \prod_k \frac{P(f_i^k|c)}{P(f_i^k|\bar{c})} = \sum_k \log \frac{P(f_i^k|c)}{P(f_i^k|\bar{c})},$$

$$\frac{P(f_i^k \mid c)}{P(f_i^k \mid \bar{c})} = \frac{(n(c, \mathcal{N}_i^k) + \epsilon)/n(c, \mathcal{D})}{(n(\bar{c}, \mathcal{N}_i^k) + \epsilon)/n(\bar{c}, \mathcal{D})} = \frac{n(c, \mathcal{N}_i^k) + \epsilon}{n(\bar{c}, \mathcal{N}_i^k) + \epsilon} \times \frac{n(\bar{c}, \mathcal{D})}{n(c, \mathcal{D})},$$

Nonparametric density estimation

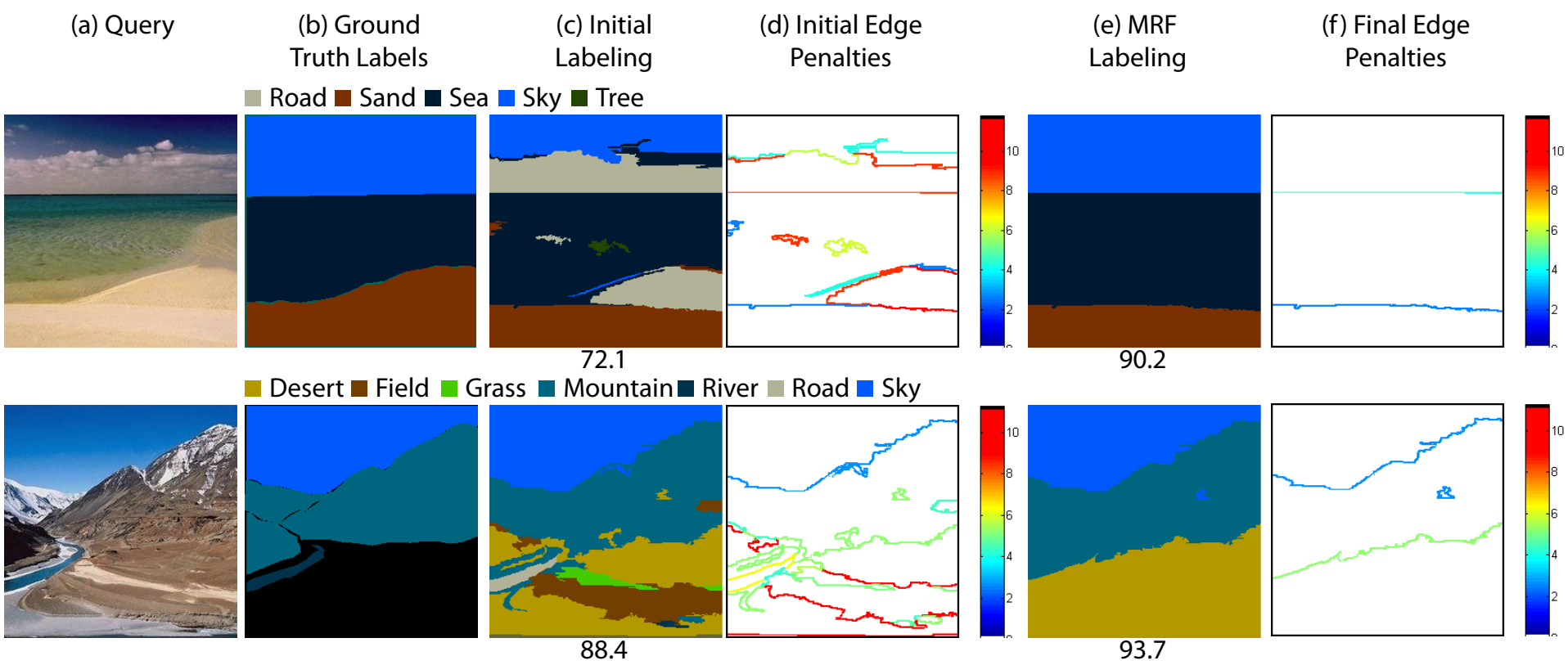
Smoothess Term

- Contextual inference:

$$E_{\text{smooth}}(c_i, c_j) = -\log[(P(c_i|c_j) + P(c_j|c_i))/2] \times \delta[c_i \neq c_j],$$

- enforce contextual constraints.
- defined based on probabilities of label co-occurrence.

Parsing Results



Including Geometric Priors

- Simultaneous labeling of regions into two types of classes: semantic and geometric

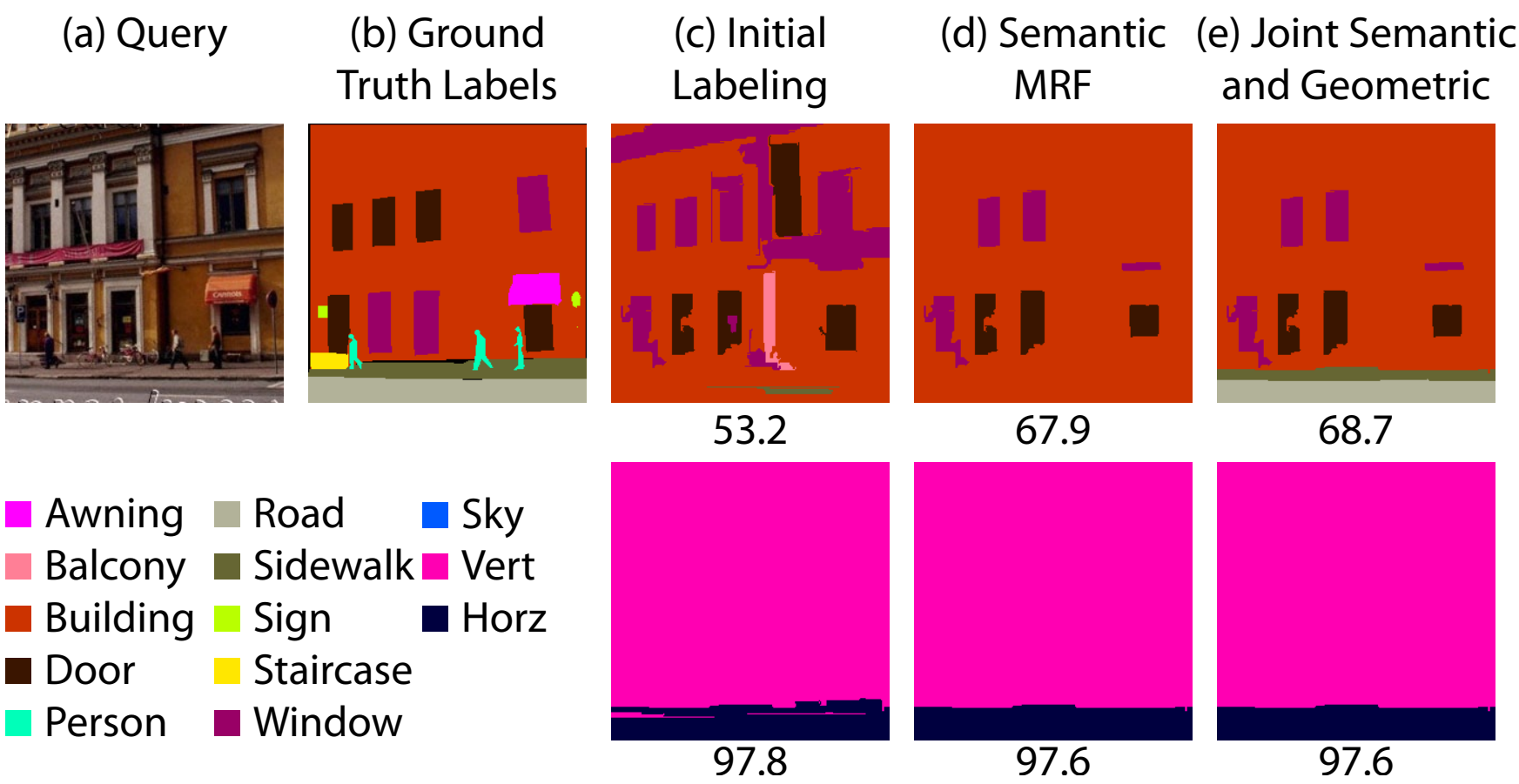
$$H(\mathbf{c}, \mathbf{g}) = J(\mathbf{c}) + J(\mathbf{g}) + \mu \sum_{s_i \in SP} \varphi(c_i, g_i)$$

- Three geometric labels:
  - sky, horizontal, and vertical

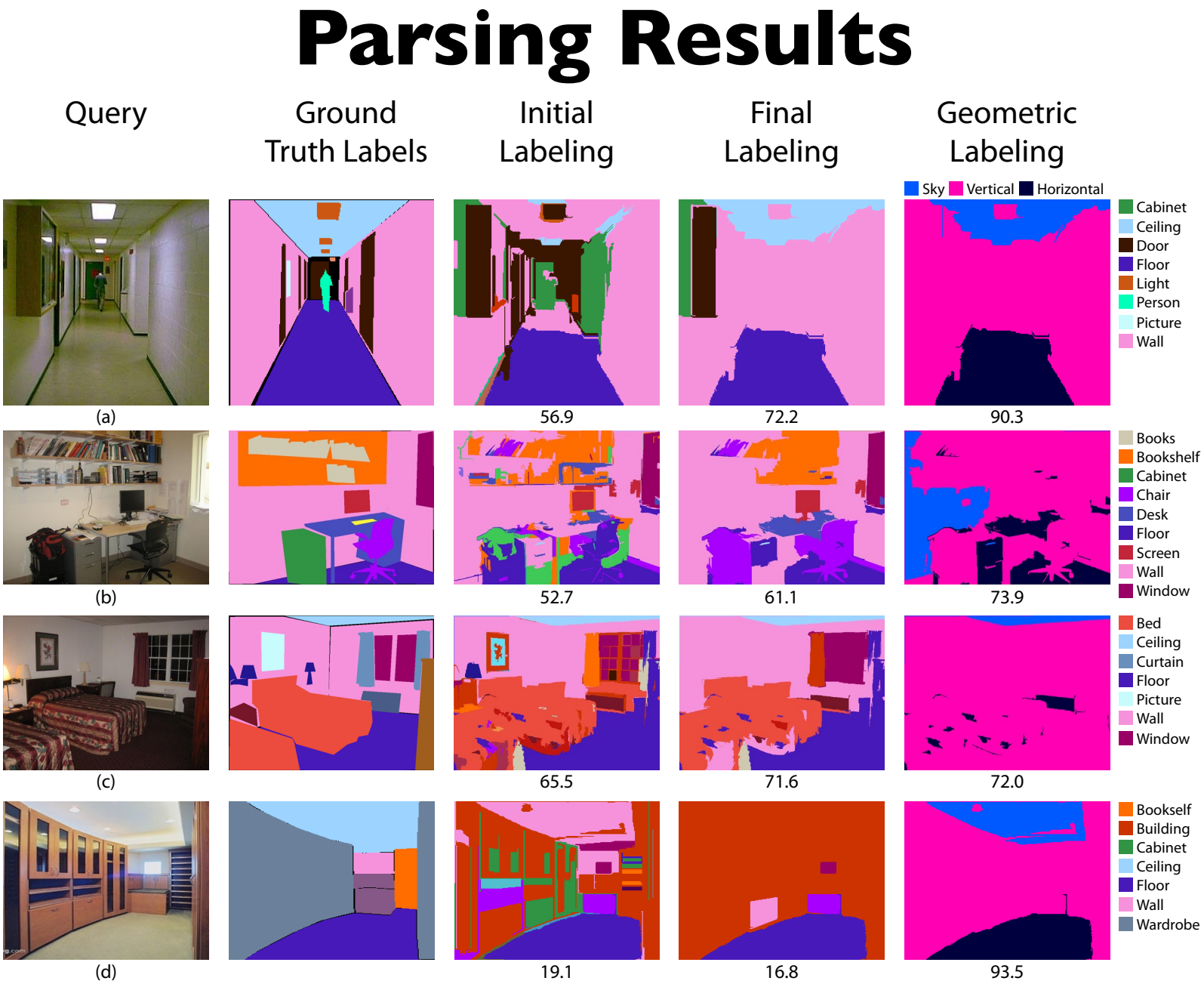
- enforces coherence between the geometric and semantic labels.

$\varphi(c_i, g_i)$: a coherence term between the semantic and geometric labels of the same superpixel.

Parsing Results



Parsing Results



Parsing Results