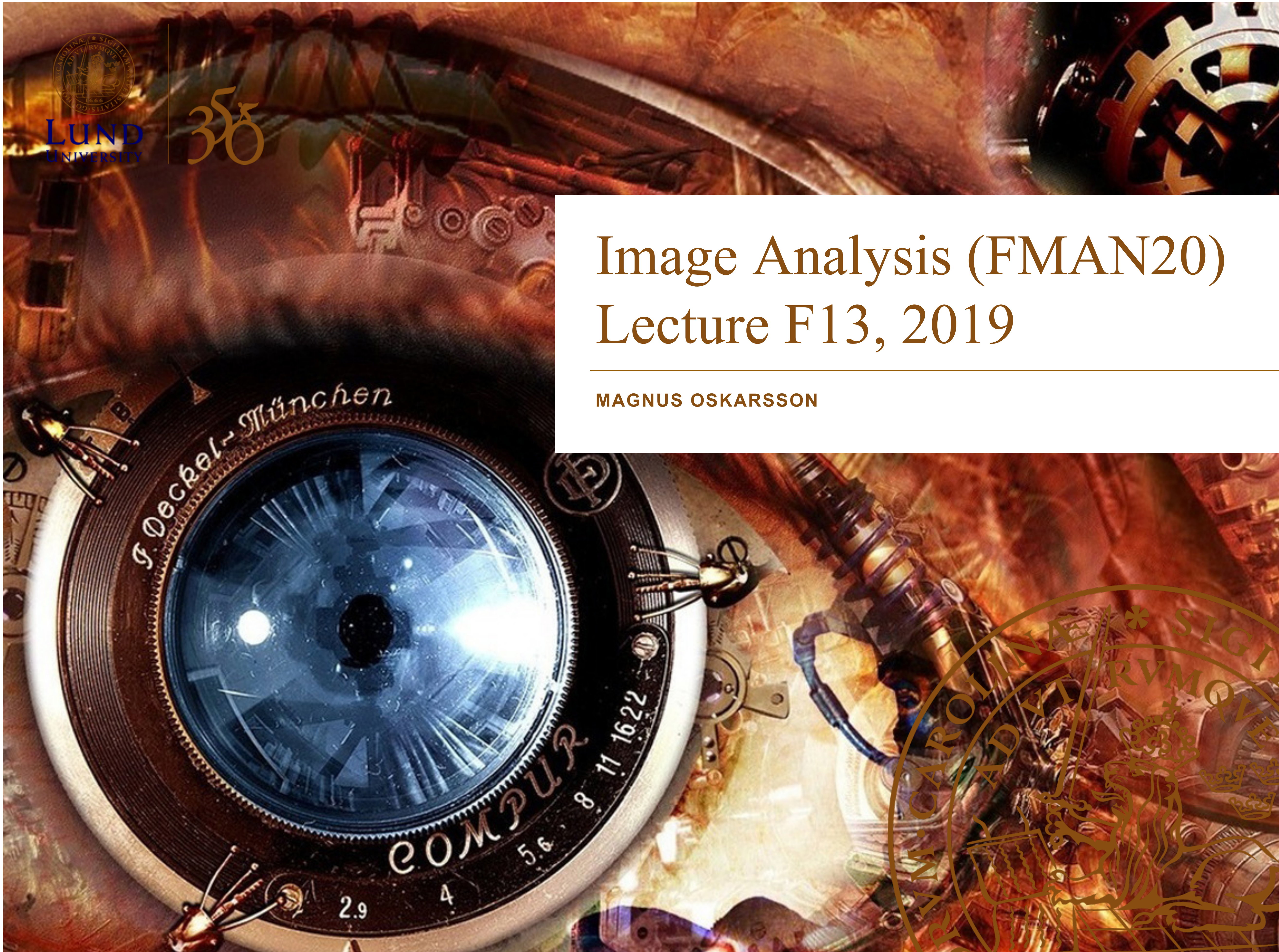


Image Analysis (FMAN20) Lecture F13, 2019

MAGNUS OSKARSSON

Overview

1. **Decision Trees**
2. Probabilistic segmentation and graph cuts
3. Semantic Segmentation

Decision trees advantages

- Simple to understand and interpret
- Requires little preparation
- Can handle both continuous and discrete data
- ’white box’ model. You can easily explain a decision afterwards
- Robust
- Performs well with large datasets

Regression trees learning

- Try each variable
- Try each threshold
- Calculate score e g
  - Entropy

$$I_E(f) = -\sum_{i=1}^{m} f_i \log_2 f_i$$

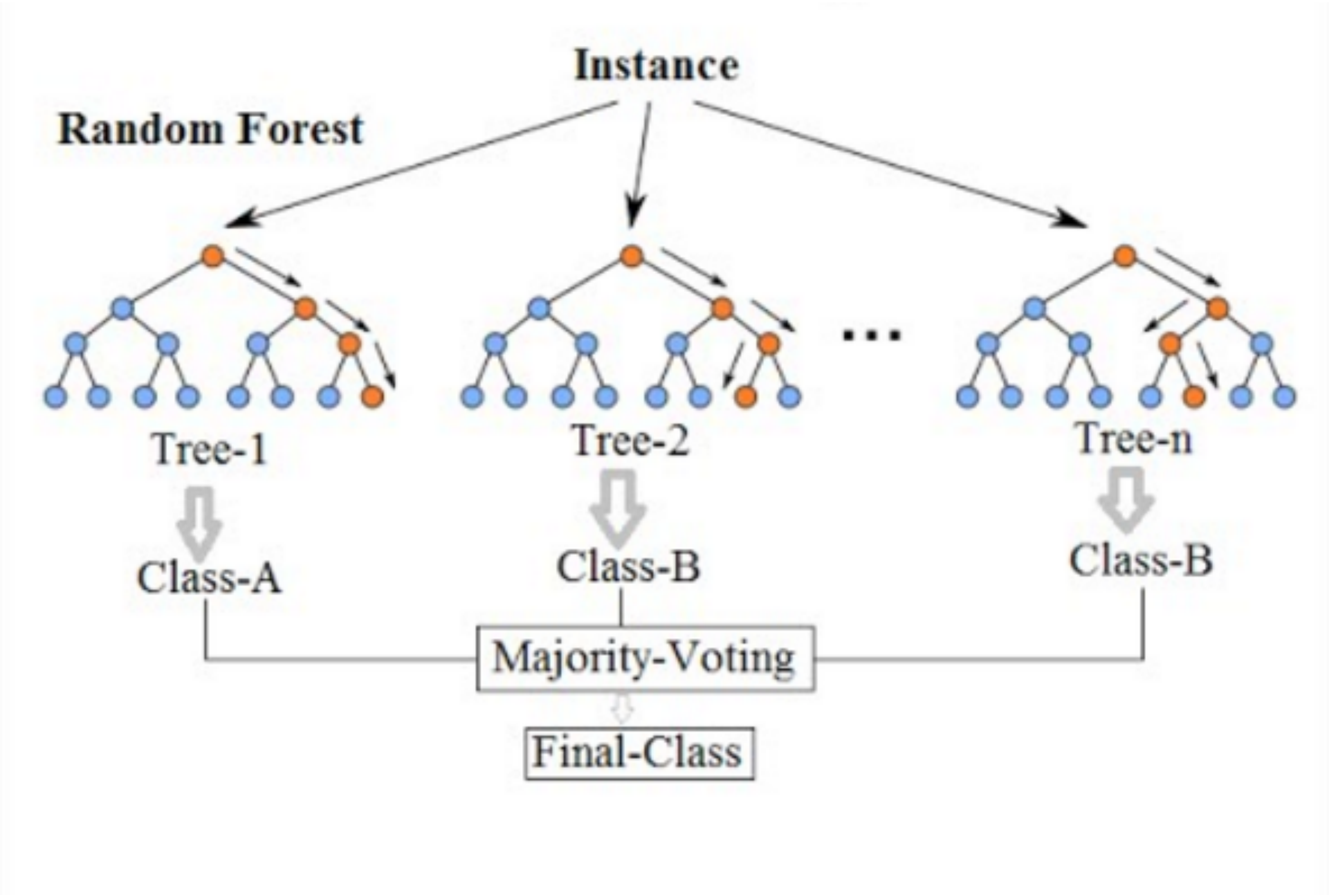
  - Gini Impurity

$$I_G(f) = \sum_{i=1}^{m} f_i(1 - f_i) = \sum_{i=1}^{m} (f_i - {f_i}^2) = \sum_{i=1}^{m} f_i - \sum_{i=1}^{m} {f_i}^2 = 1 - \sum_{i=1}^{m} {f_i}^2$$

Decision tree limitations

- Optimal learning is NP-complete (use heuristics)
- Problems with over fitting

Random Forest



Overview

1. Decision Trees
2. **Probabilistic segmentation and graph cuts**
3. Semantic Segmentation

Segmentation - Graph Cuts

- Idea:

1. See the segmentation problem as a classification problem
2. Finding the highest a posteriori classification (segmentation) is an optimization problem
3. Construct a graph so that the min-cut problem is equivalent to the optimization problem in step 2.
4. Compute a minimum cut that gives the optimal solution.

Note: Min-cut of a graph can be efficiently computed (polynomial time) via max flow algorithms.

A priori probabilities of segmentations

Idea:

We are segmenting pixels g_i as foreground (1) and some as background (0).

The probability of having a foreground pixel or a background pixel might be different

$P(g_i=0)=p_0$

$P(g_i=1)=p_1$

Note: Min-cut of a graph can be efficiently computed (polynomial time) via max flow algorithms.

Statistical interpretation

Notation:

f – observed image

g – sought, unknown image

$P(g|f)$ - posterior distribution

Using the *Maximum A Posteriori* (*MAP*) principle, we should maximize the posterior distribution.

Bayes rule: $$P(g|f) = \frac{P(f|g)P(g)}{P(f)}$$

Negative logs give:

$$-\log(P(g|f)) = -\log(P(f|g)) - \log(P(g)) + \text{const}$$

Energy:

$$E(f,g) = E_{data}(f,g) + E_{prior}(g)$$

Statistical two-phase Mumford-Shah

Energy: $E(f,g) = E_{data}(f,g) + E_{prior}(g)$

Recall:

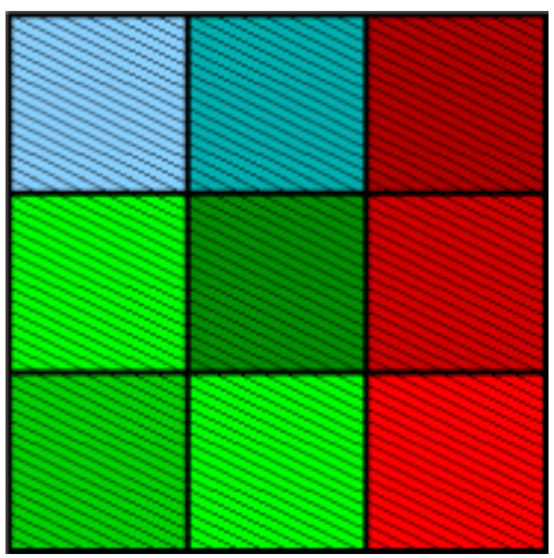
$$E_0(a_1, a_2, \Gamma) = \int_{R_1} (a_1 - f)^2 dxdy + \int_{R_2} (a_2 - f)^2 dxdy + \nu|\Gamma|$$

First two data terms: ”reconstructed *g* should be close to data (image) *f*”.
Third term: ”prior knowledge says that shorter curves *g* are preferred”.

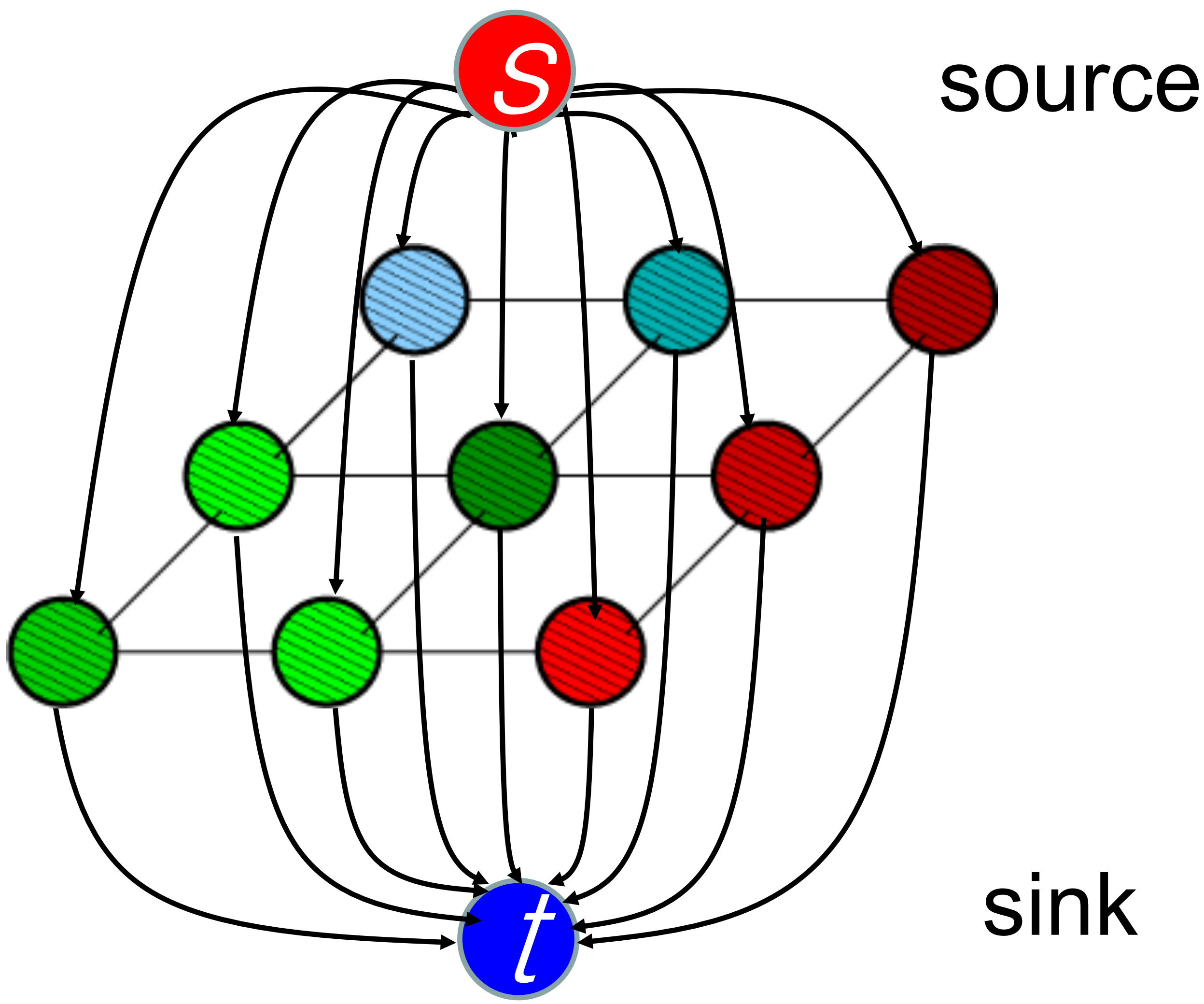
More general formulation:

$$E_0(\Gamma) = \int_{R_1} -\log(P(f(x,y)|\text{class1})dxdy + \int_{R_2} -\log(P(f(x,y)|\text{class2})dxdy + \nu|\Gamma|$$

Graph representation of images

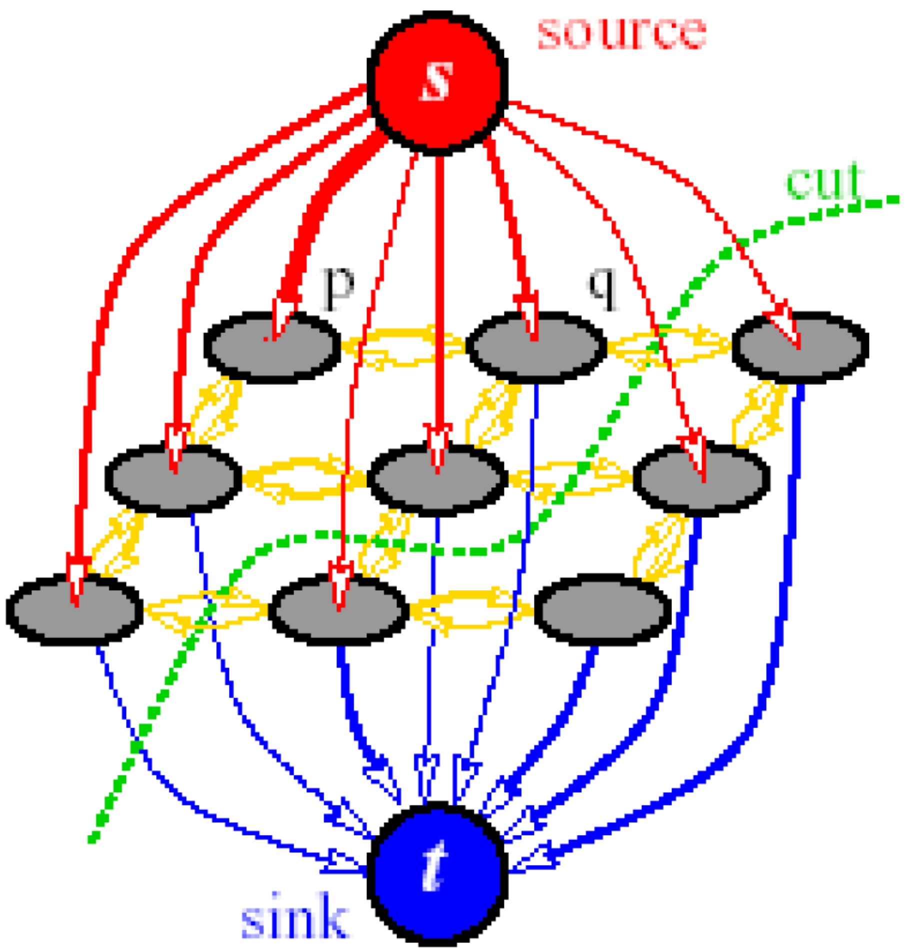


3x3 image



Directed, weighted graph, one node for every pixel + source and sink nodes

Minimum Cuts

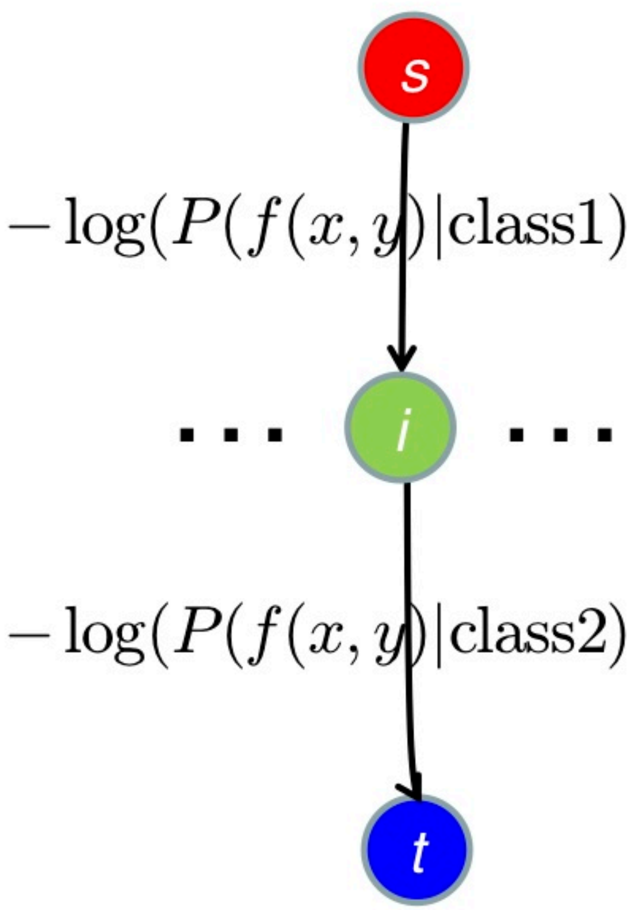


Definition: A *minimum cut* is a cut with minimum cost.

Note: A cut separates all nodes in two sets: (i) nodes that are connected to the source nodes, and (ii) those that are not.

Edge weights for statistical model

Set edge weights such that a cut corresponds to a solution of the optimization problem



Consider pixel *i*. A cut must contain either:

1. the edge (*s*,*i*), or
2. the edge (*i*,*t*)

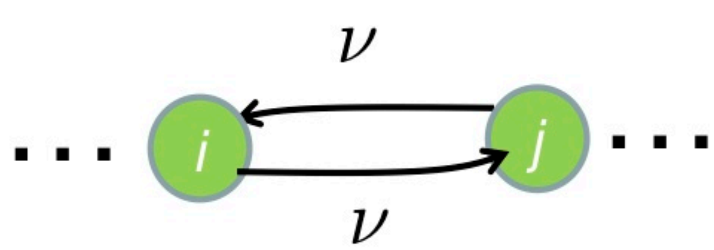
Set edge weights accordingly:

1. $-\log(P(f(x,y)|\text{class1})$ for edge (*s*,*i*),
2. $-\log(P(f(x,y)|\text{class2})$ for edge (*i*,*t*)

$$E_0(\Gamma) = \int_{R_1} -\log(P(f(x,y)|\text{class1})dxdy + \int_{R_2} -\log(P(f(x,y)|\text{class2})dxdy + \nu|\Gamma|$$

Edge weights – regularization term

Set edge weights such that a cut corresponds to a solution of the optimization problem



Consider two neighbouring pixels *i* and *j*. If they are in different classes and hence a boundary is passing between them, then a cut must contain either:

1. the edge (*i,j*), or
2. the edge (*j,i*)

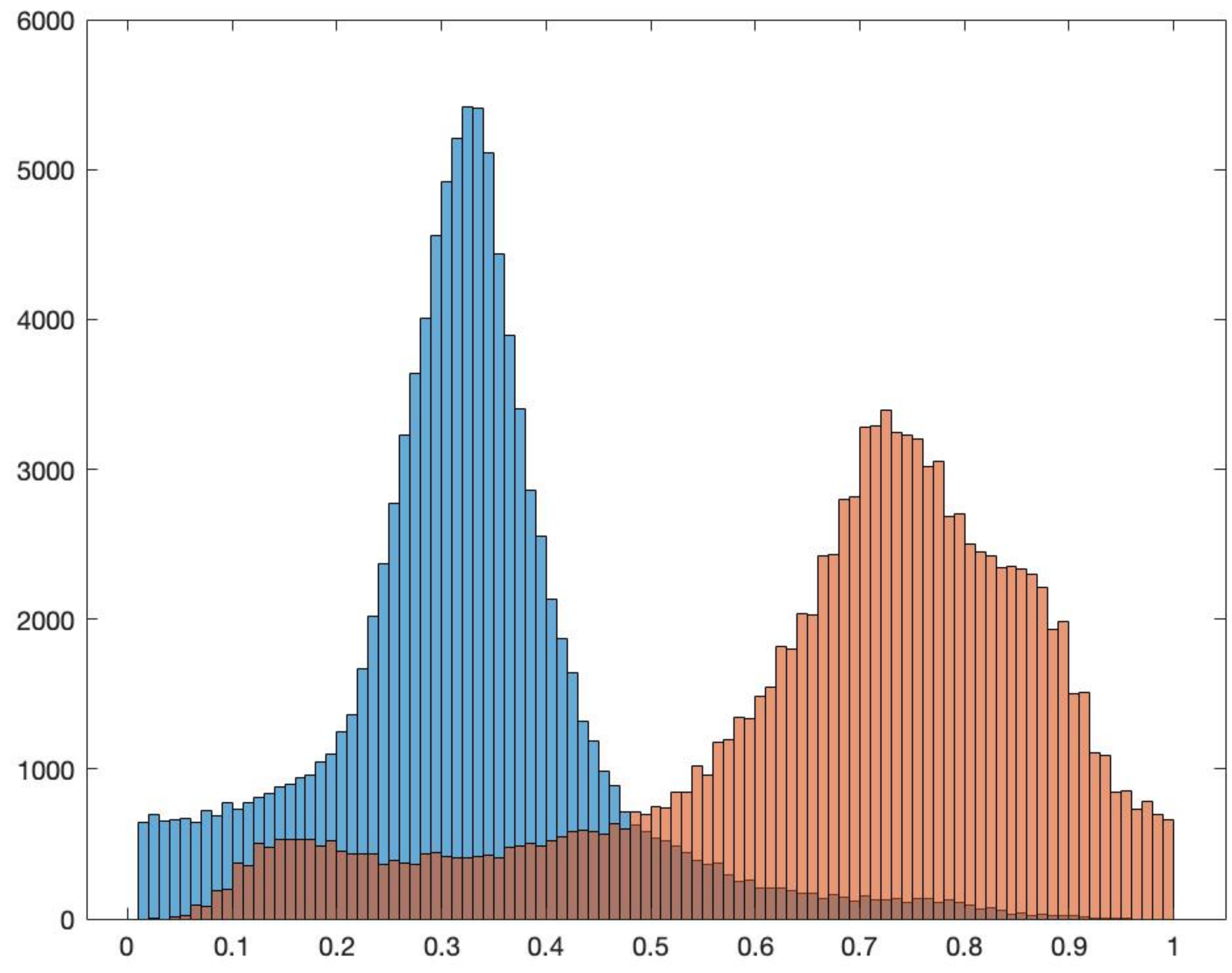
Set edge weights accordingly:

1. ν for edge (*i,j*),
2. ν for edge (*j,i*)

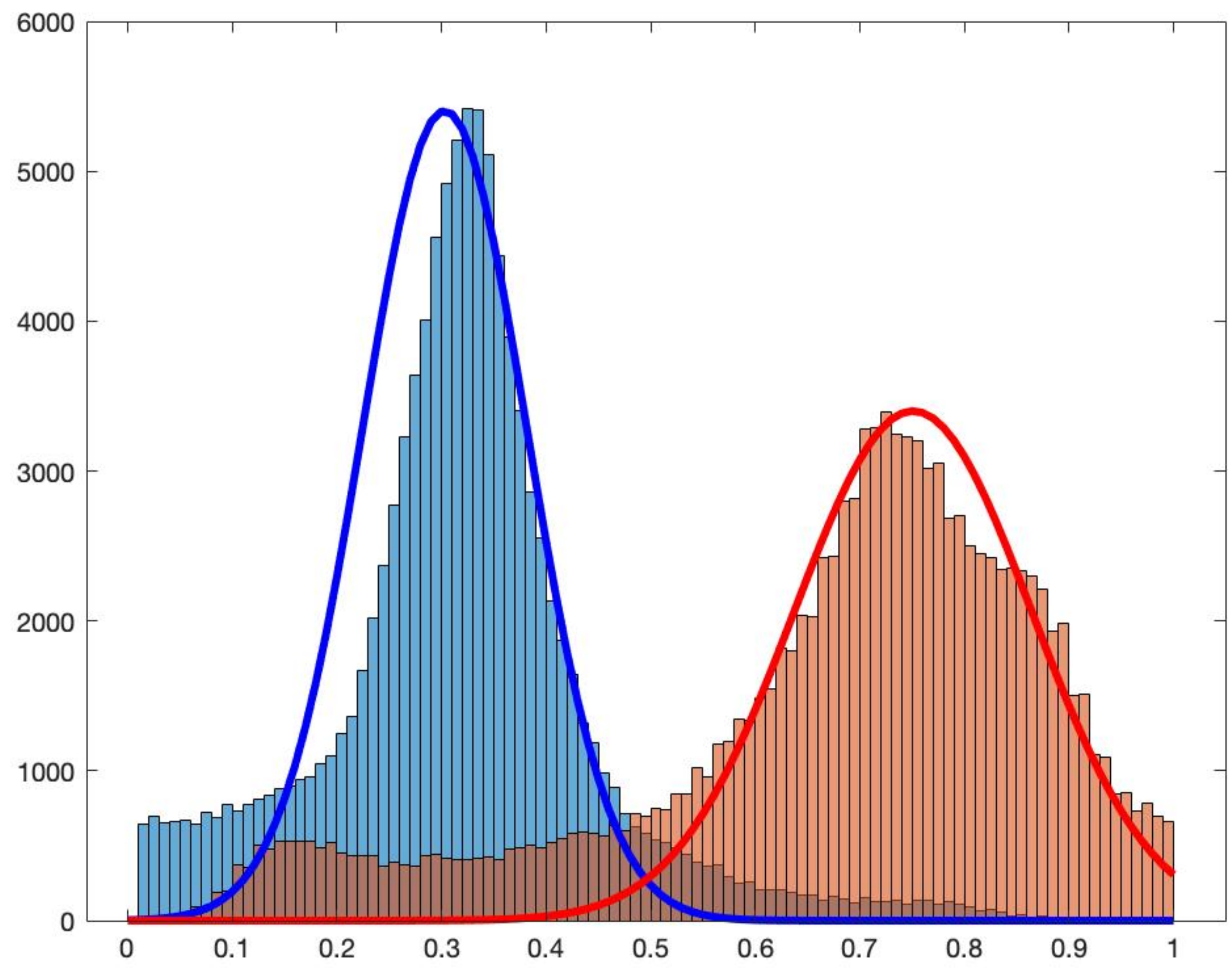
reg. term

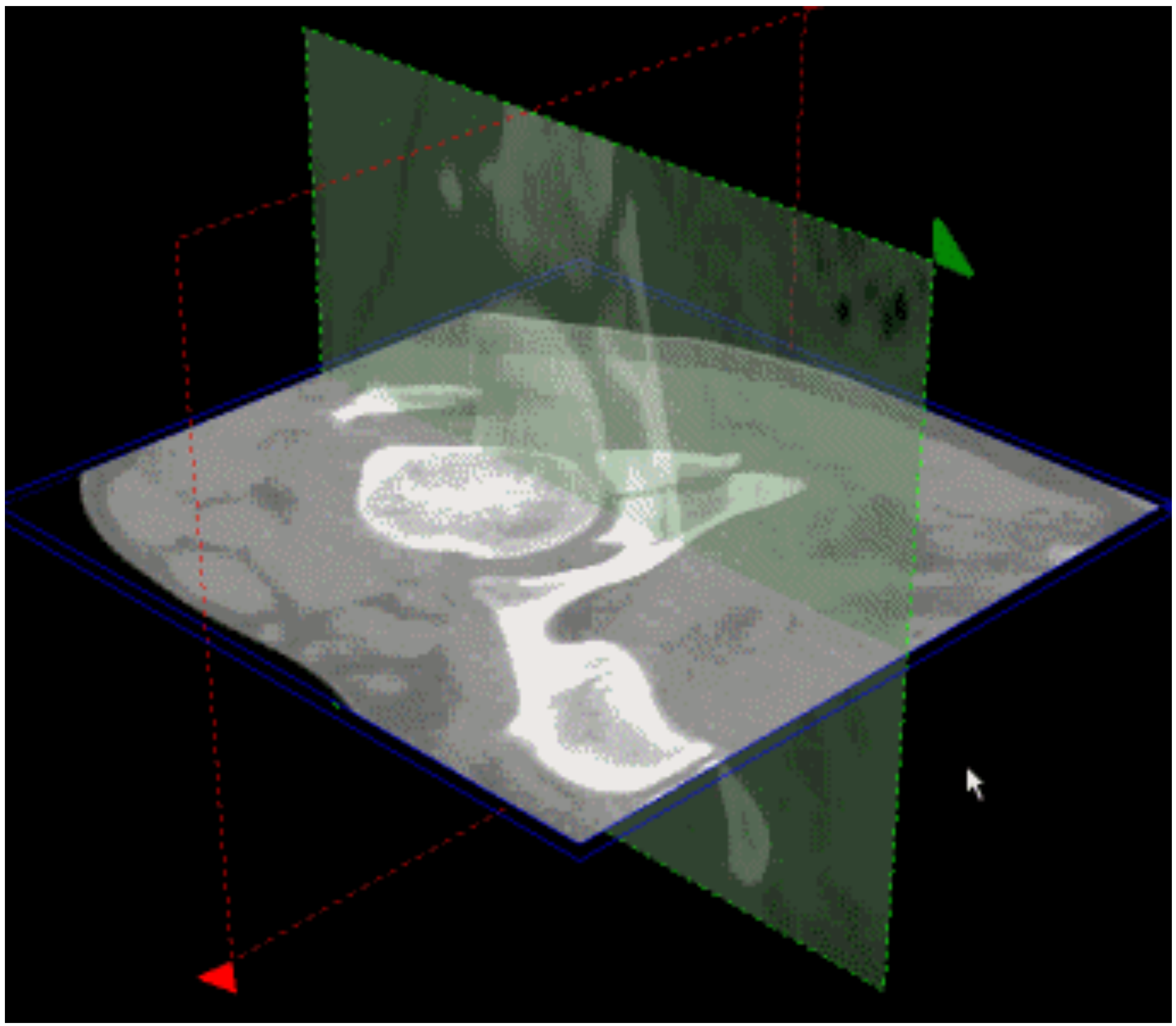
$$E_0(\Gamma) = \int_{R_1} -\log(P(f(x,y)|\text{class1})dxdy + \int_{R_2} -\log(P(f(x,y)|\text{class2})dxdy + \nu|\Gamma|$$

Example: model background and foreground as normal distributions



Example: model background and foreground as normal distributions





Overview

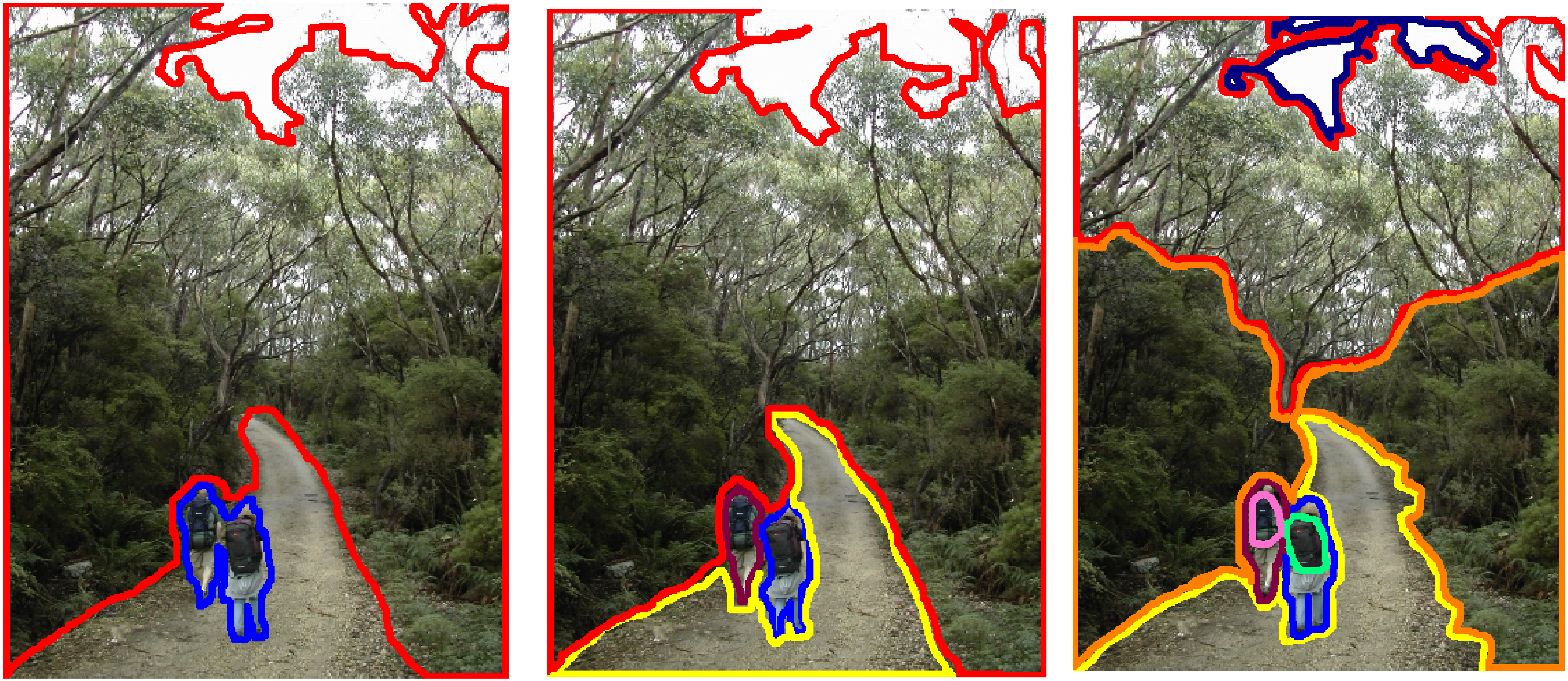
1. Decision Trees
2. Probabilistic segmentation and graph cuts
3. **Semantic Segmentation**

Segmentation

- **Image segmentation**: breaking the pixels or tokens of an image into regions (groups) that share some property
- **Semantic segmentation**: attach category labels to groups

Segmentation - ill-posed
What is the right segmentation?

Segmentation - ill-posed
What is the right segmentation?



Segmentation - ill-posed
What is the right segmentation?

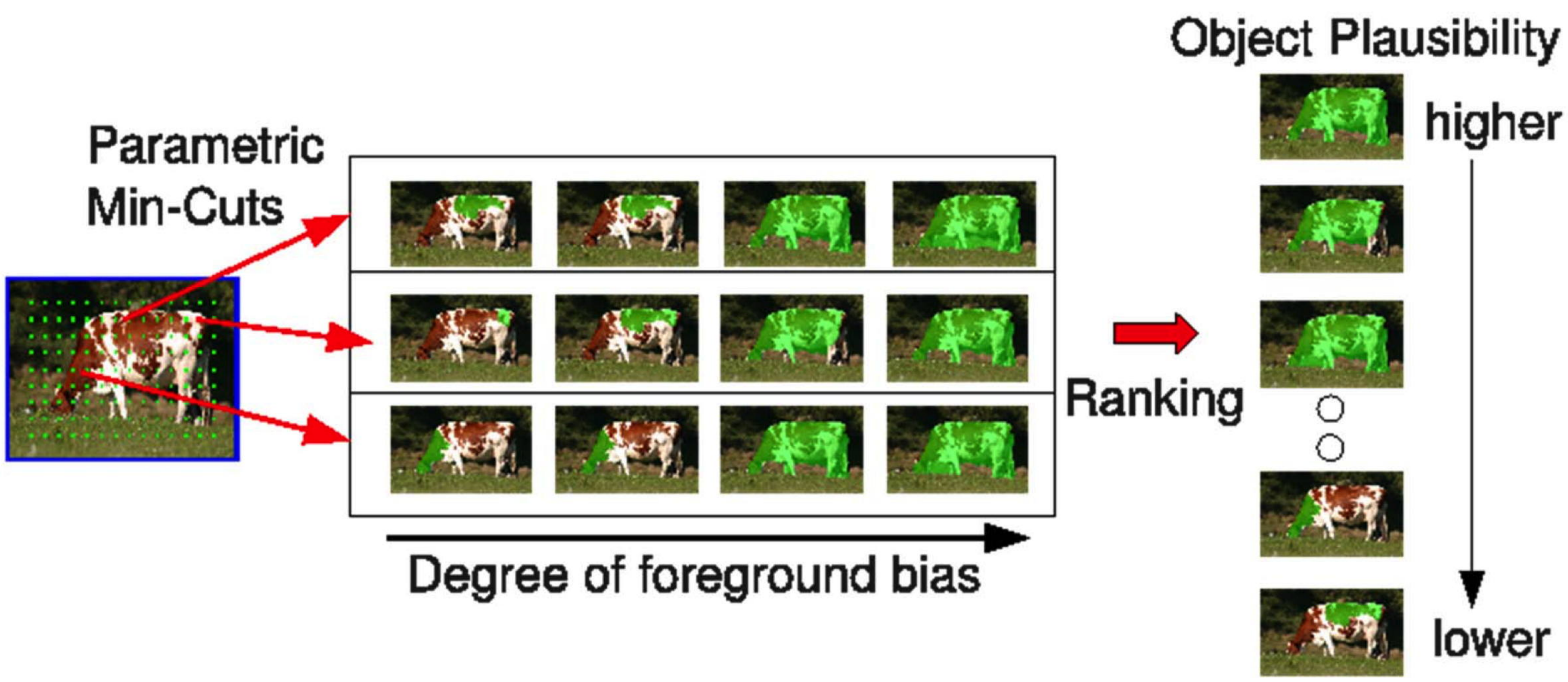
"A woman with a backpack and a man, also wearing a backpack, are walking on a road. On the sides of the road high trees as well as lower vegetation can be seen. Above, a white sky is peeking through the treetops."

LUND UNIVERSITY

Semantic Segmentation

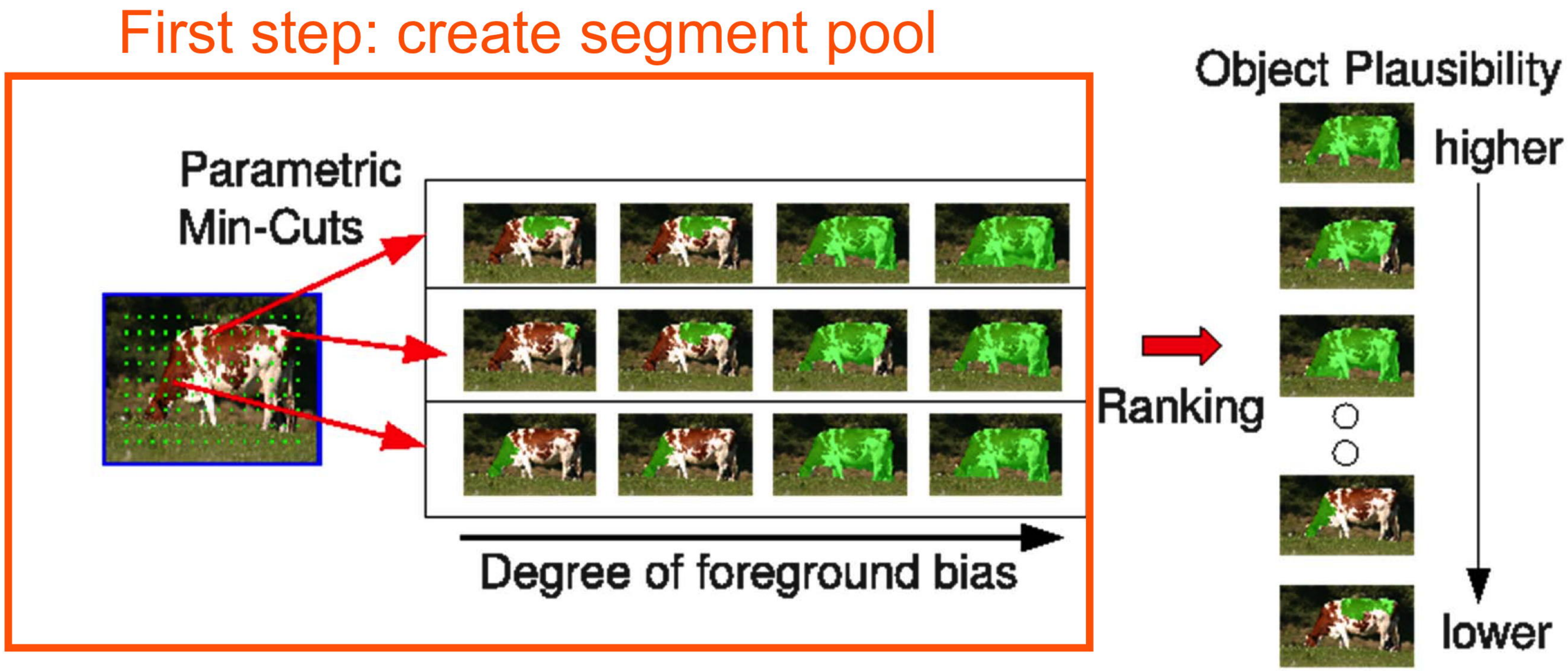
- **Multiple figure-ground segmentations** generated by searching for breakpoints of constrained min-cut energies, at multiple locations and spatial scales on image grid (CPMC)
- **Plausible object segments** are selected after ranking and diversifying the low-level segmentations based on mid-level, class-independent, visual cues
- **Recognition stage** detects objects from the multiple categories and sequentially resolves inconsistencies

CPMC: Constrained Parametric Min-Cuts for Automatic Object Segmentation



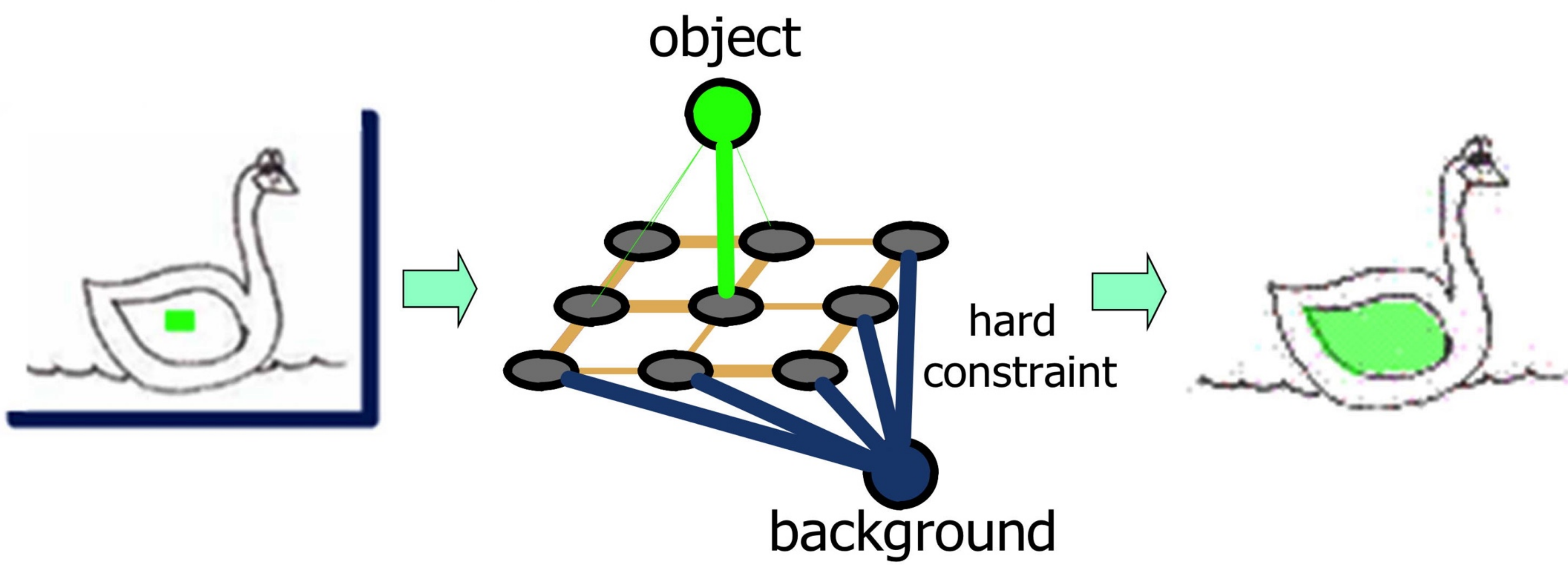
Carreira and Sminchisescu. CVPR 10, PAMI 11

CPMC: Constrained Parametric Min-Cuts for Automatic Object Segmentation



Generating a segment pool: constrained min-cut

$$E_{\lambda}(x) = \sum_{u \in V} D(x_u, \lambda) + \sum_{(u,v) \in E} V_{uv}(x_u, x_v)$$



Constrained Parametric Min-Cuts

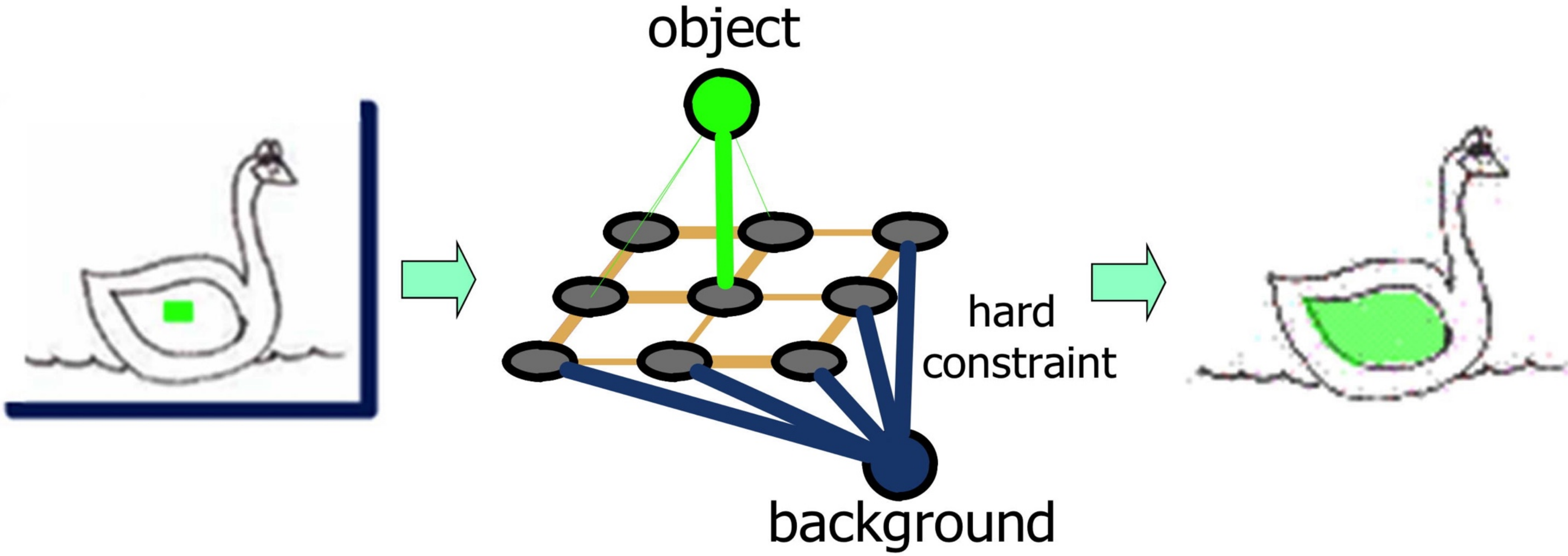
$$E_\lambda(x) = \sum_{u \in V} D(x_u, \lambda) + \sum_{(u,v) \in E} V_{uv}(x_u, x_v)$$

$$D(x_u, \lambda) = \begin{cases} 0, & \text{if } x_u = 1, x_u \notin X_b \\ \infty, & \text{if } x_u = 1, x_u \in X_b \\ \infty, & \text{if } x_u = 0, x_u \in X_f \\ \ln \dfrac{p_f(x_u)}{p_b(x_u)} + \lambda, & \text{if } x_u = 0, x_u \notin X_f \end{cases}$$

$$V(x_u, x_v) = \begin{cases} 0, & \text{if } x_u = x_v \\ \exp\left[-\dfrac{\max(gPb(u), gPb(v))}{\sigma^2}\right], & \text{if } x_u \neq x_v \end{cases}$$

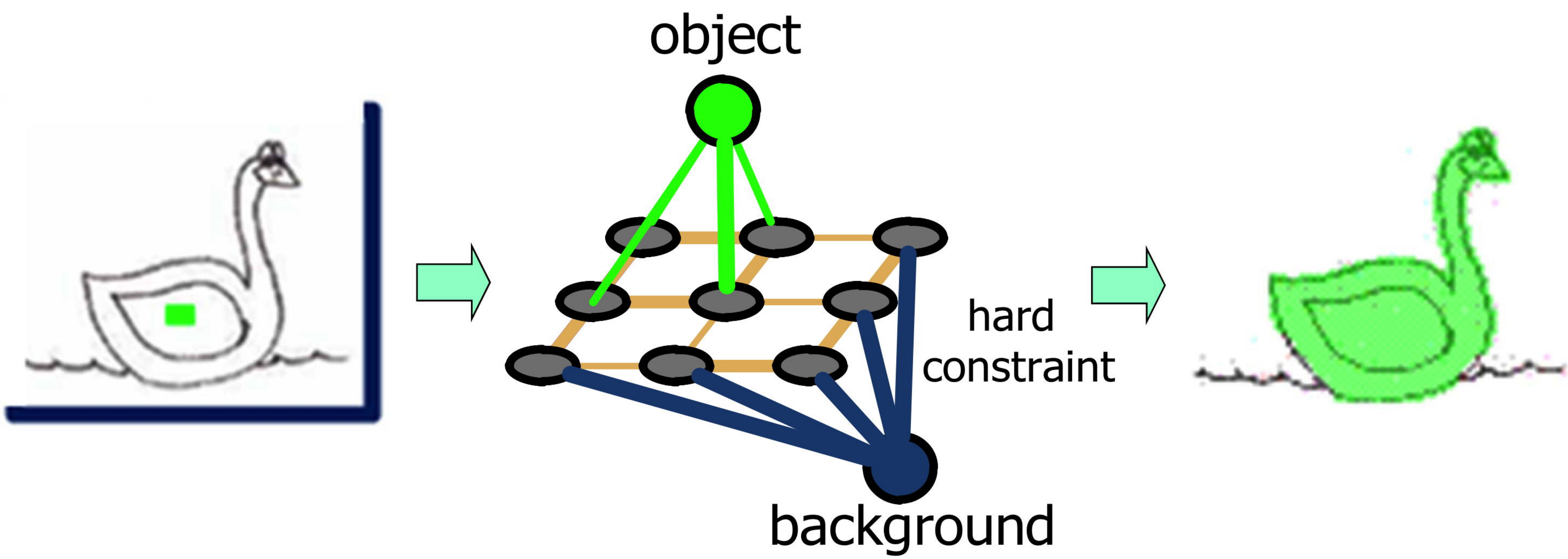
Generating a segment pool: constrained min-cut

$$E_\lambda(x) = \sum_{u \in V} D(x_u, \lambda) + \sum_{(u,v) \in E} V_{uv}(x_u, x_v)$$

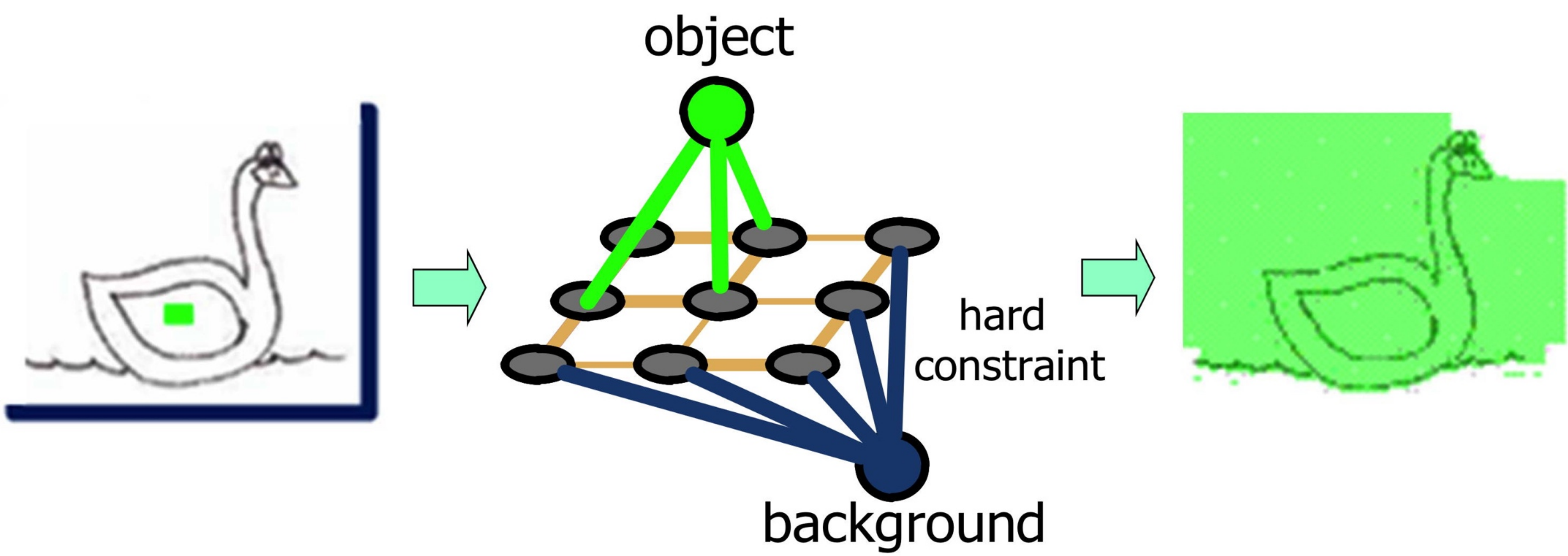


Generating a segment pool: constrained *parametric* min-cuts

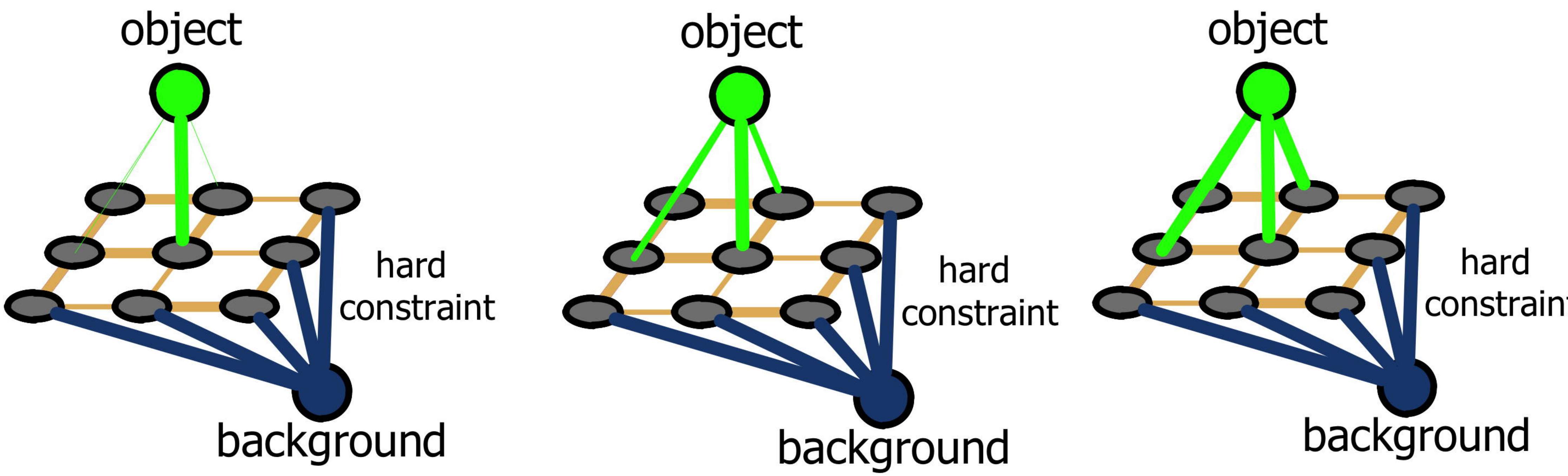
$$E_\lambda(x) = \sum_{u \in V} D(x_u, \lambda) + \sum_{(u,v) \in E} V_{uv}(x_u, x_v)$$



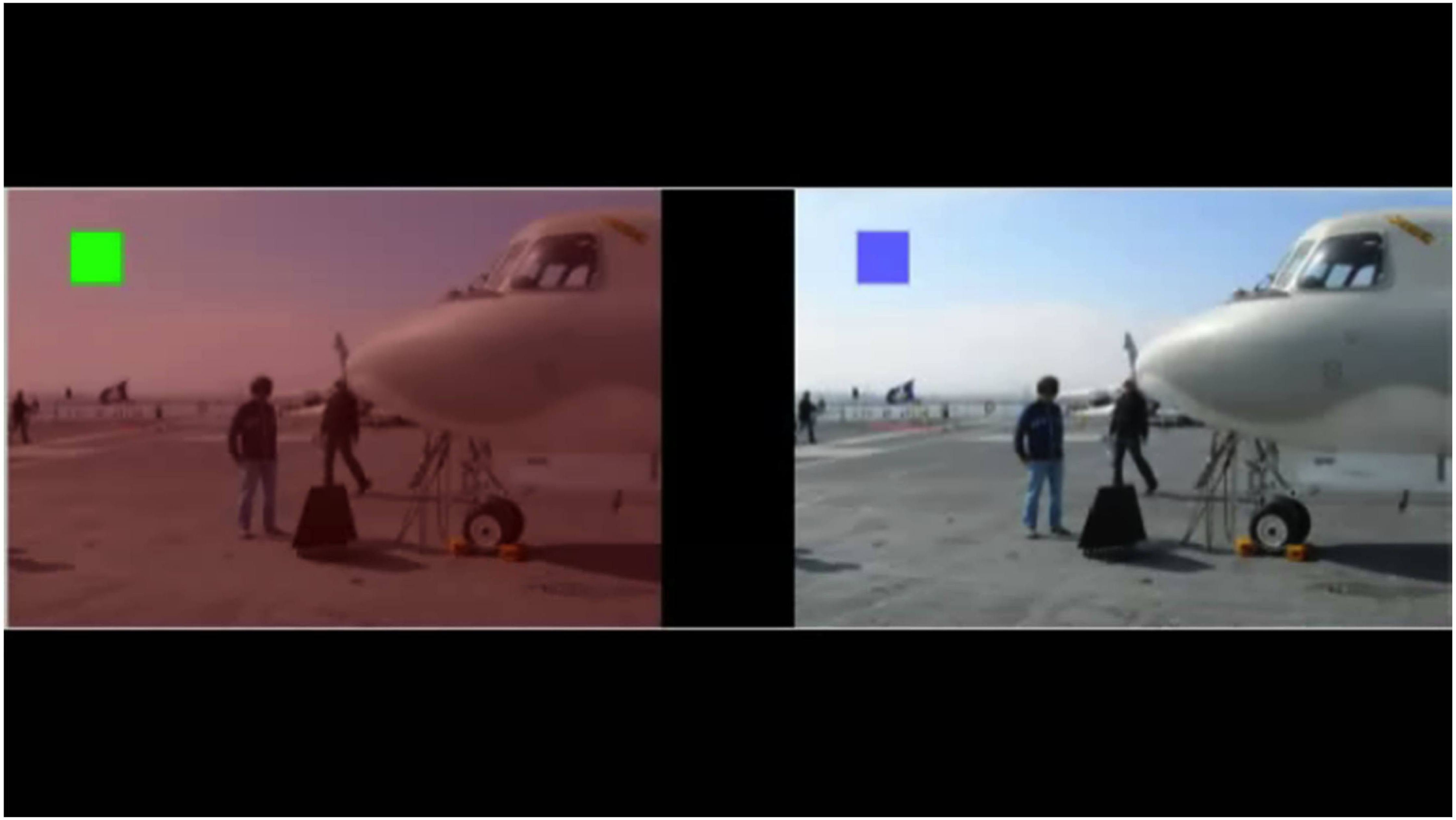
Generating a segment pool: constrained *parametric* min-cuts

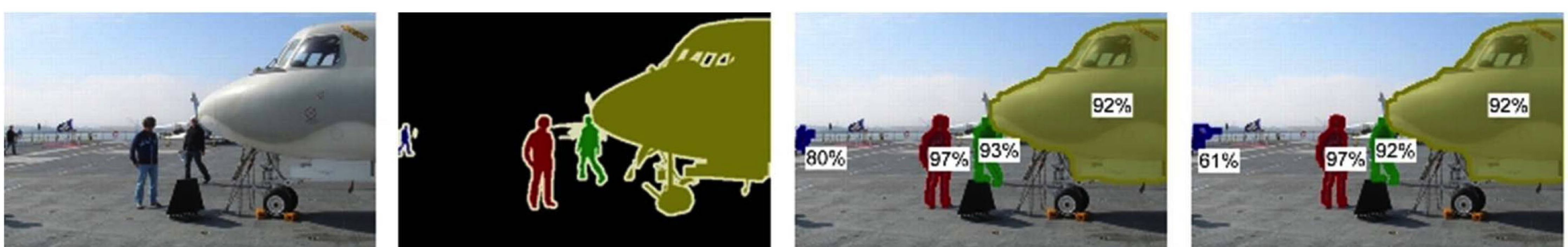


Generating a segment pool: constrained *parametric* min-cuts



Can solve for all values of object bias in the same time complexity of solving a single min-cut using a **parametric max-flow solver**



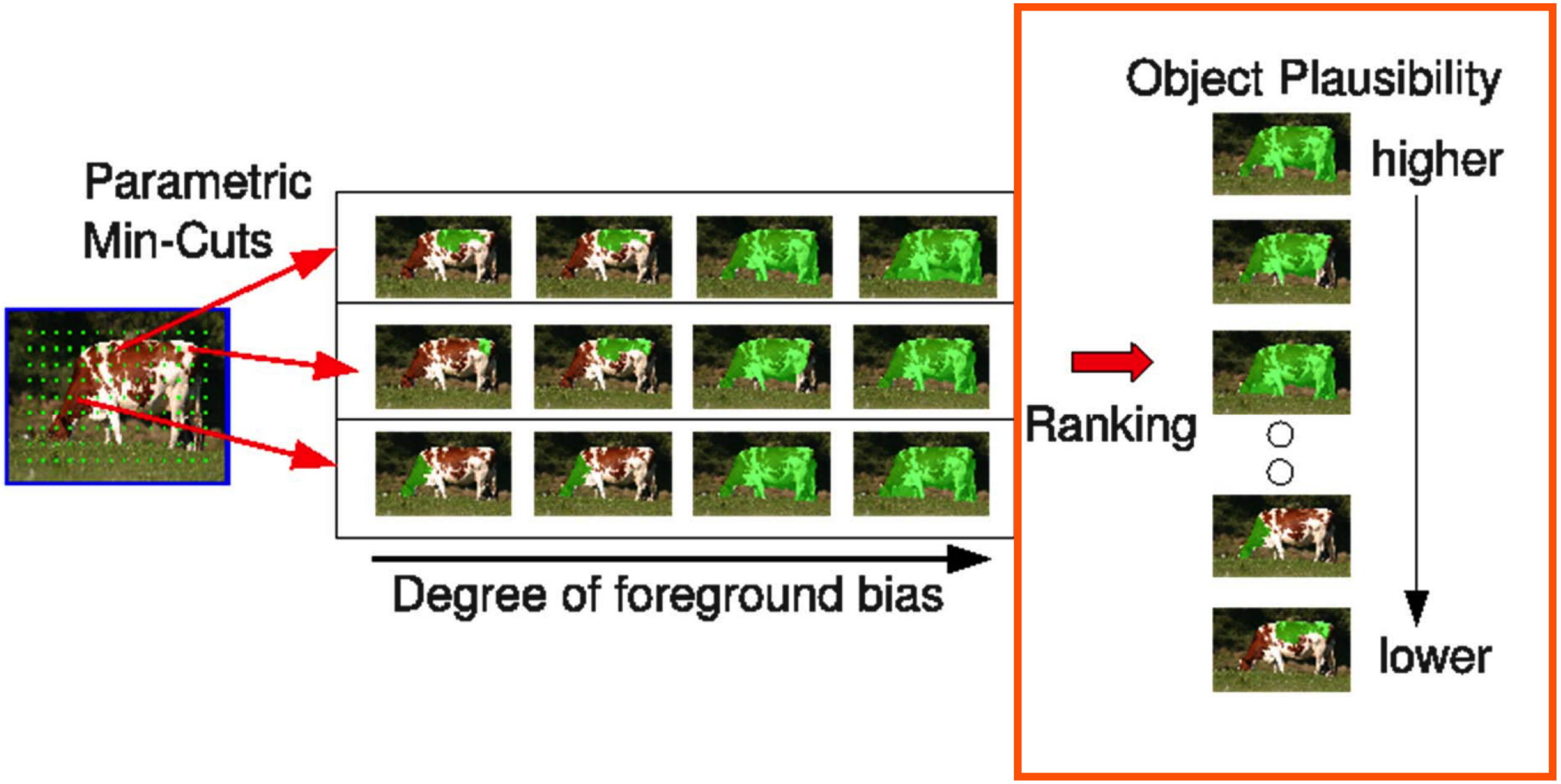

92%
80%
97%
93%
92%
61%
97%
92%

Overview – Semantic Segmentation

1. Edge detectors based on machine learning
2. Segmentation is an ill-posed problem
3. Generating a pool of possible segments (CPMC)
4. **Rating segments in the pool**
5. Visual and Semantic Processing
6. Second Order Pooling

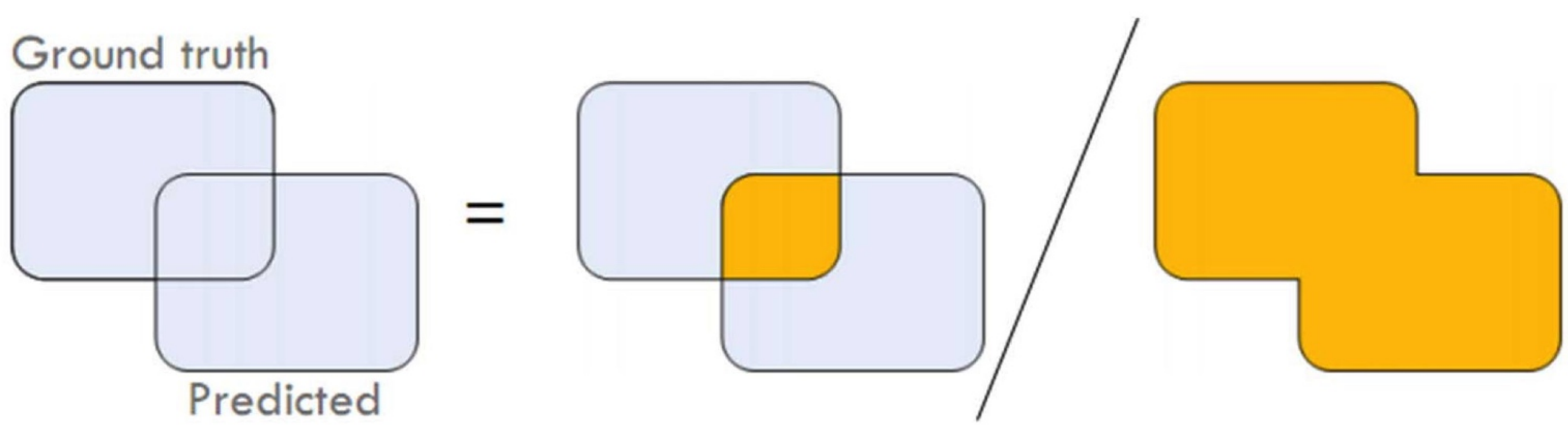
CPMC: Constrained Parametric Min-Cuts for Automatic Object Segmentation

Second step: ranking object hypotheses



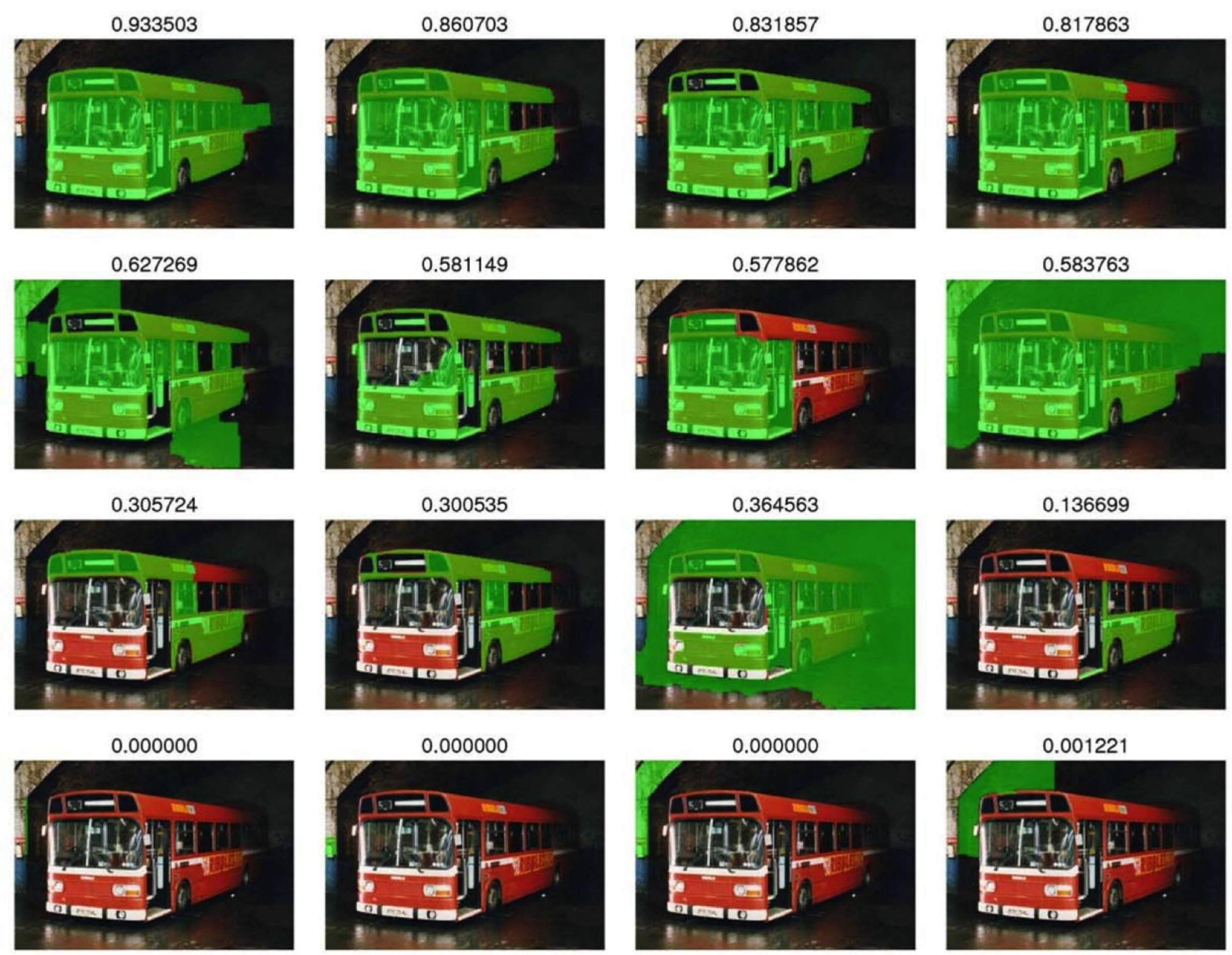
How to model segment quality ?

Best **overlap** with a ground truth object computed by intersection-over-union.



Ranking figure-ground hypotheses

- Supervised learning framework
- Hypothesized segments ranked using regression
- Ranking is class–independent (mid-level)



Learning to rank object hypotheses

Input object hypotheses

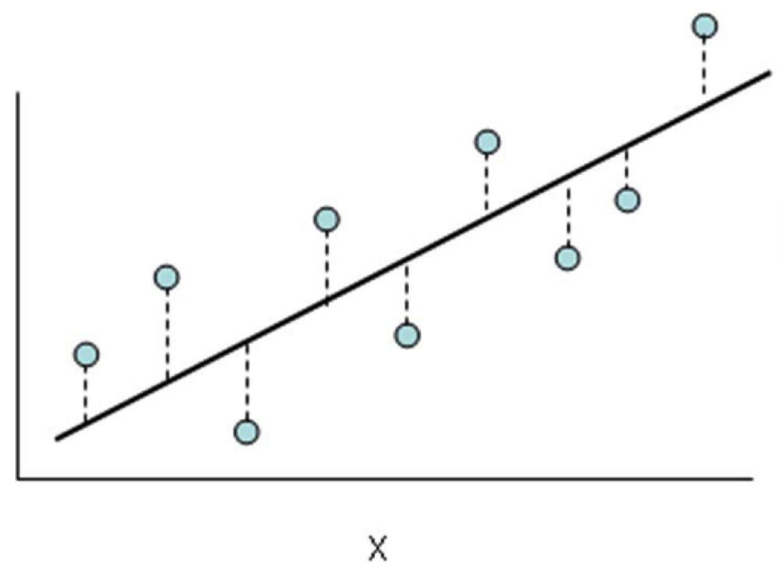
Feature Vectors

$$\begin{bmatrix} 0.1 \\ -0.3 \\ 0.5 \\ \vdots \\ 0.4 \end{bmatrix}$$

$$\begin{bmatrix} 0.5 \\ 0.3 \\ 0.9 \\ \vdots \\ 0.4 \end{bmatrix}$$

$$\begin{bmatrix} -0.3 \\ 0 \\ 0.1 \\ \vdots \\ 0.9 \end{bmatrix}$$

Parameter Optimization



Desired Prediction

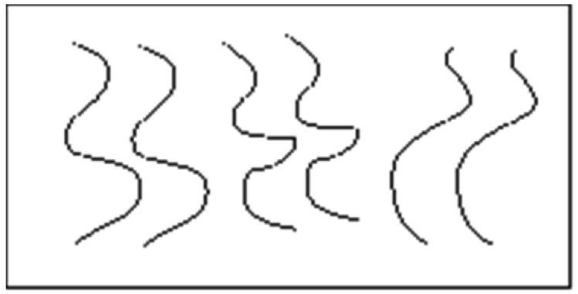
0.63

0.93

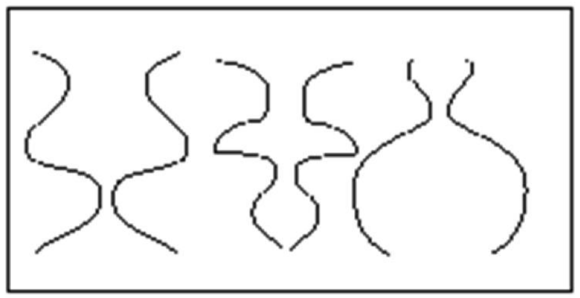
0

Gestalts

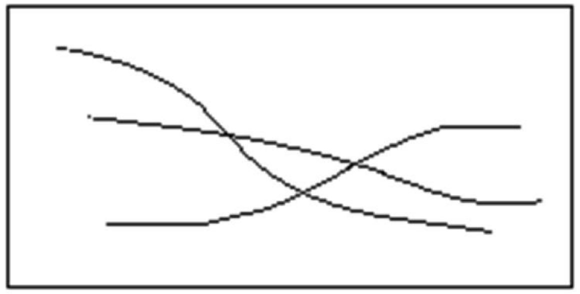
Gestalt psychology identifies several properties that result in grouping/segmentation:



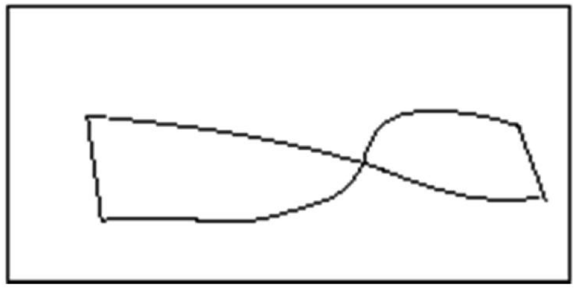
Parallelism



Symmetry

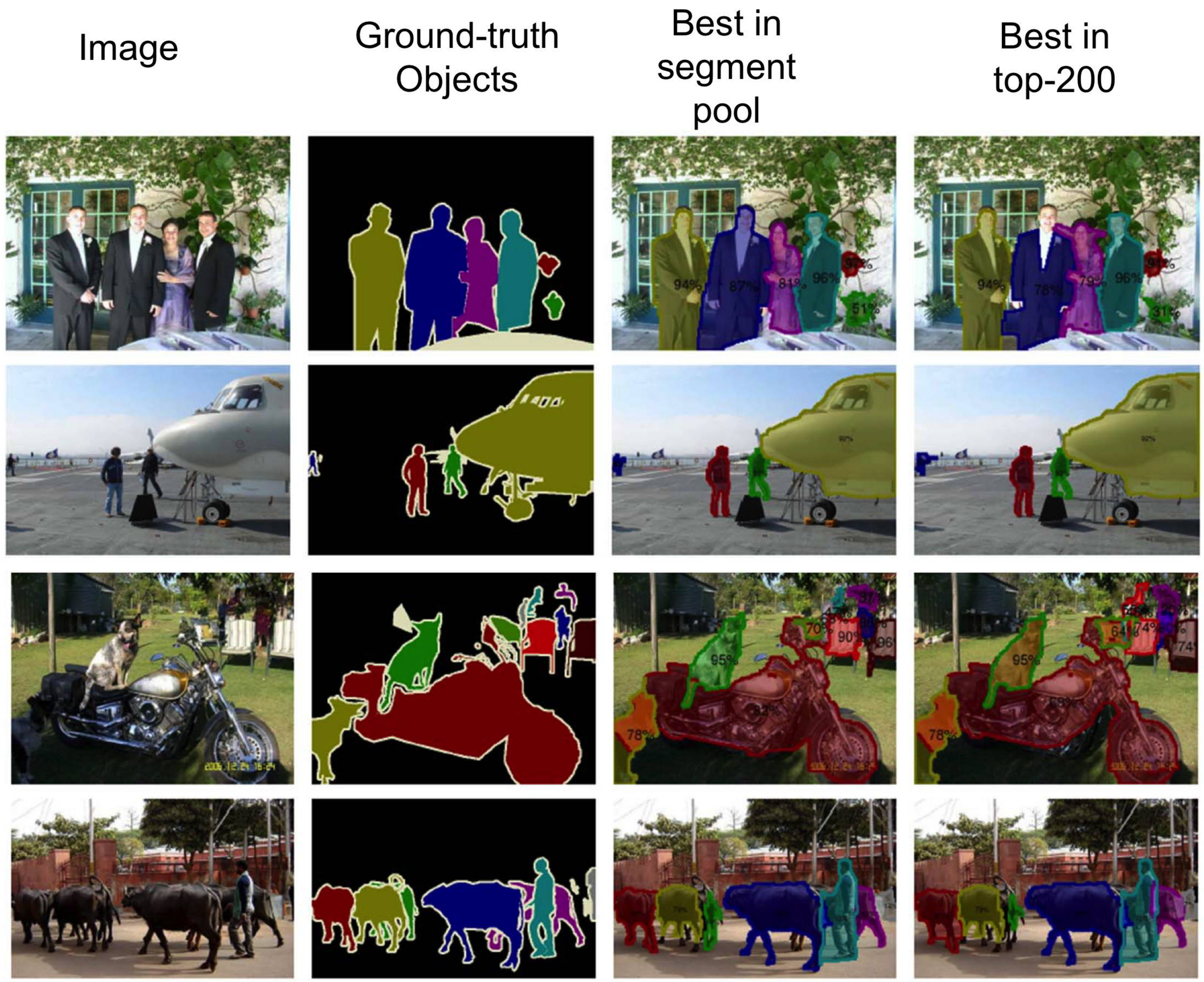


Continuity



Closure

Segmentation Examples



Semantic Segmentation
Second Order Pooling

https://www.youtube.com/watch?v=u5Ee0HFboLA

LUND UNIVERSITY

Visual and semantic processing

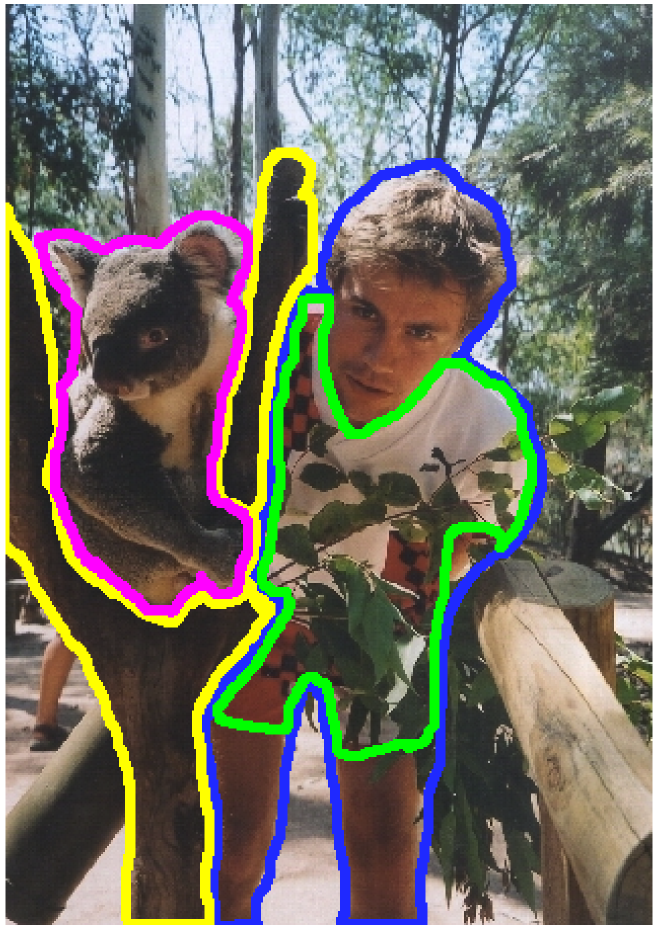
Input:

"a man with a white, black and red football uniform is standing behind a trunk with a koala on it"

LUND UNIVERSITY

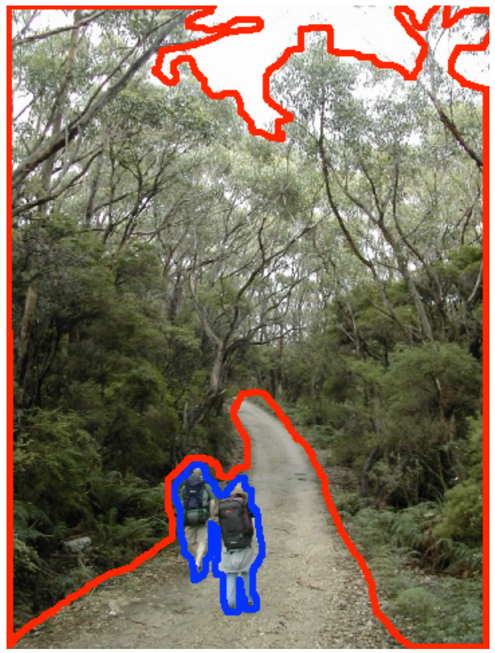
Visual and semantic processing

Output:



"a man with a white, black and red football uniform is standing behind a trunk with a koala on it"

Visual and Semantic Processing



"People walking in the woods”

“A woman with a backpack and a man, also wearing a backpack, are walking on a road. On the sides of the road high trees as well as lower vegetation can be seen. Above, a white sky is peeking through the treetops."

Problem Formulation

- We investigate the problem of segmenting images using the information in text annotations.
- In contrast to the general image understanding problem, this type of annotation guided segmentation is less ill-posed.
- We present a system based on a combined visual and semantic pipeline.

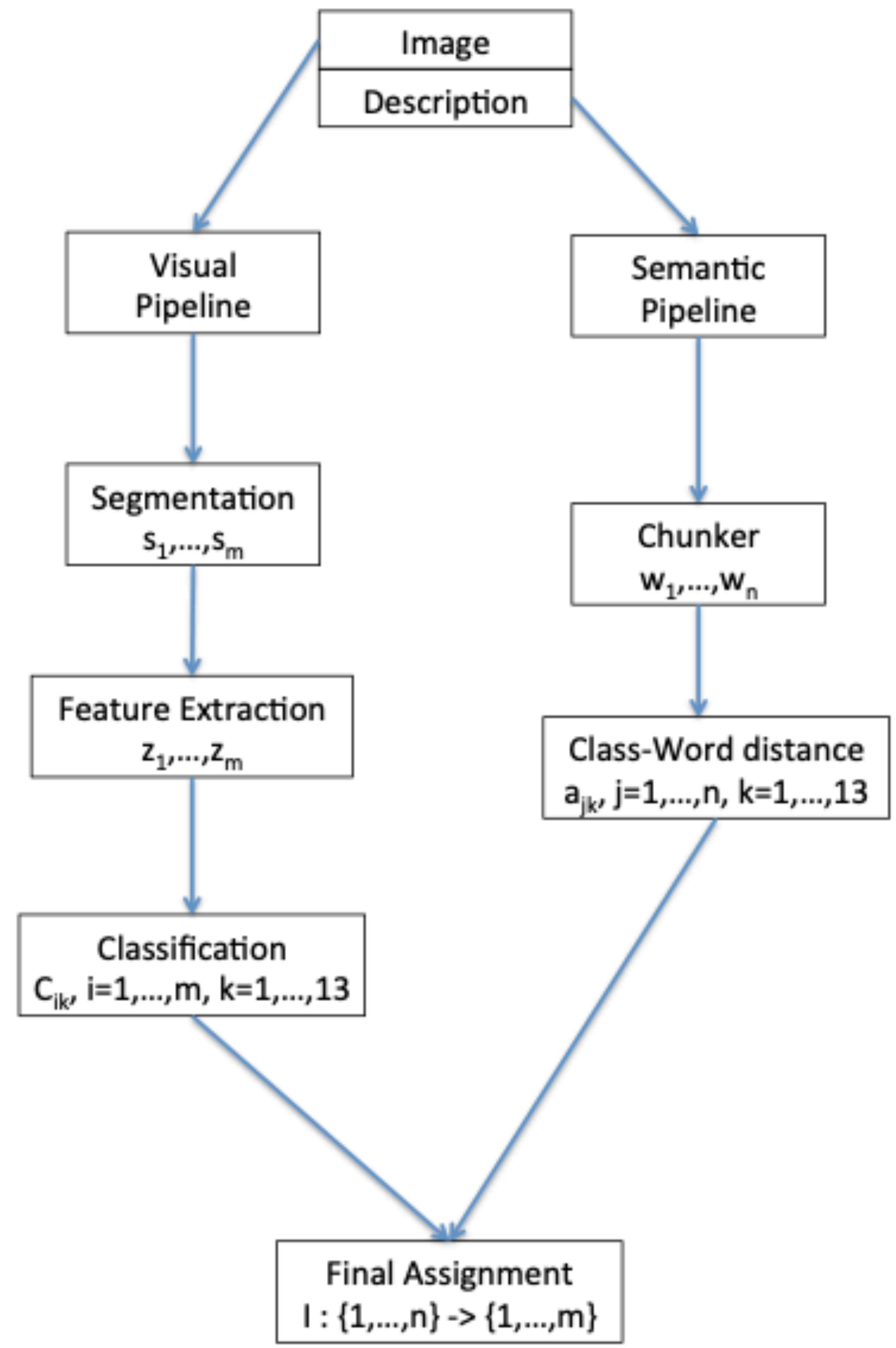
Visual and Semantic Processing

Visual Parsing

- Image segmented using CPMC. Usually 500-1000 segments.
- 27 features.
- Classification into 13 visual categories

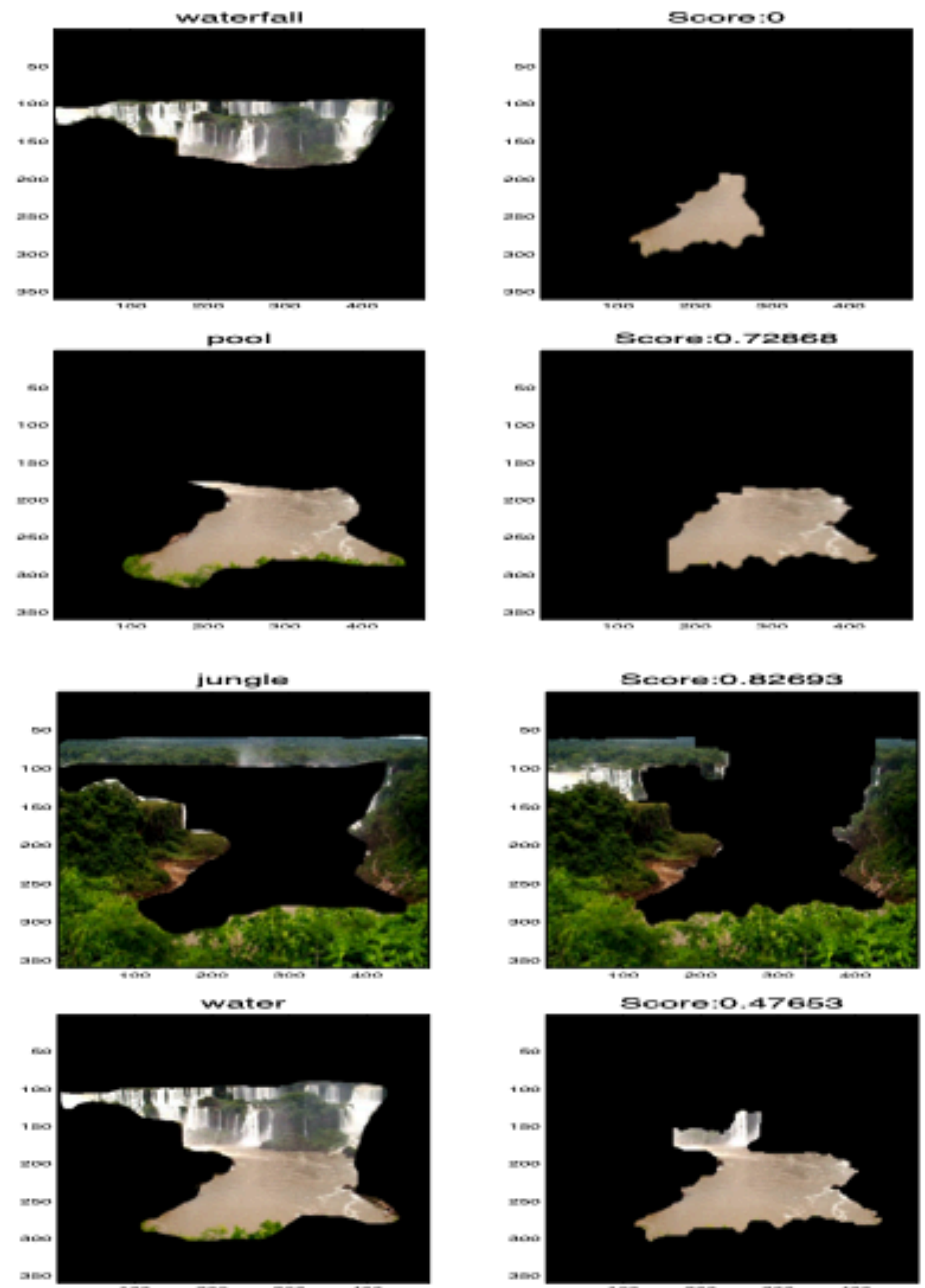
Semantic Parsing

- Chunking of text to produce key-nouns in text. Usually 3-10 key-nouns per annotation
- Calculation of semantic distance between each key-noun and each visual category.
- Final assignment using combinatorial optimization of segment for each key-noun



Visual and Semantic Processing

"A cascading waterfall in the middle of the jungle; front view with pool of dirty water in the foreground"



LUND UNIVERSITY

350