Conditional and Markov Random Fields

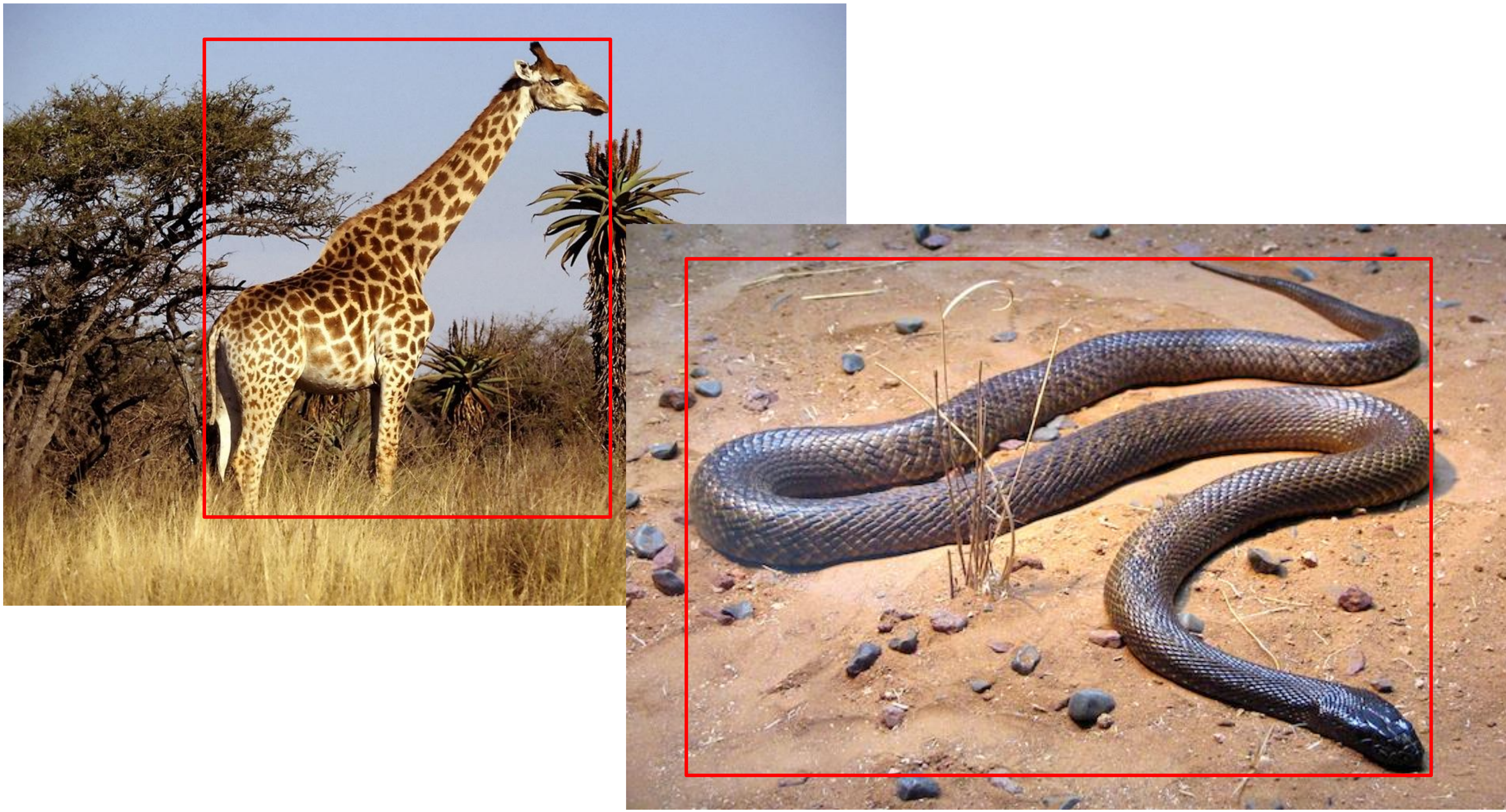
Julian Kooij
Intelligent Vehicles group, 3ME

IN4393 - Computer Vision
2017-05-24

TU Delft

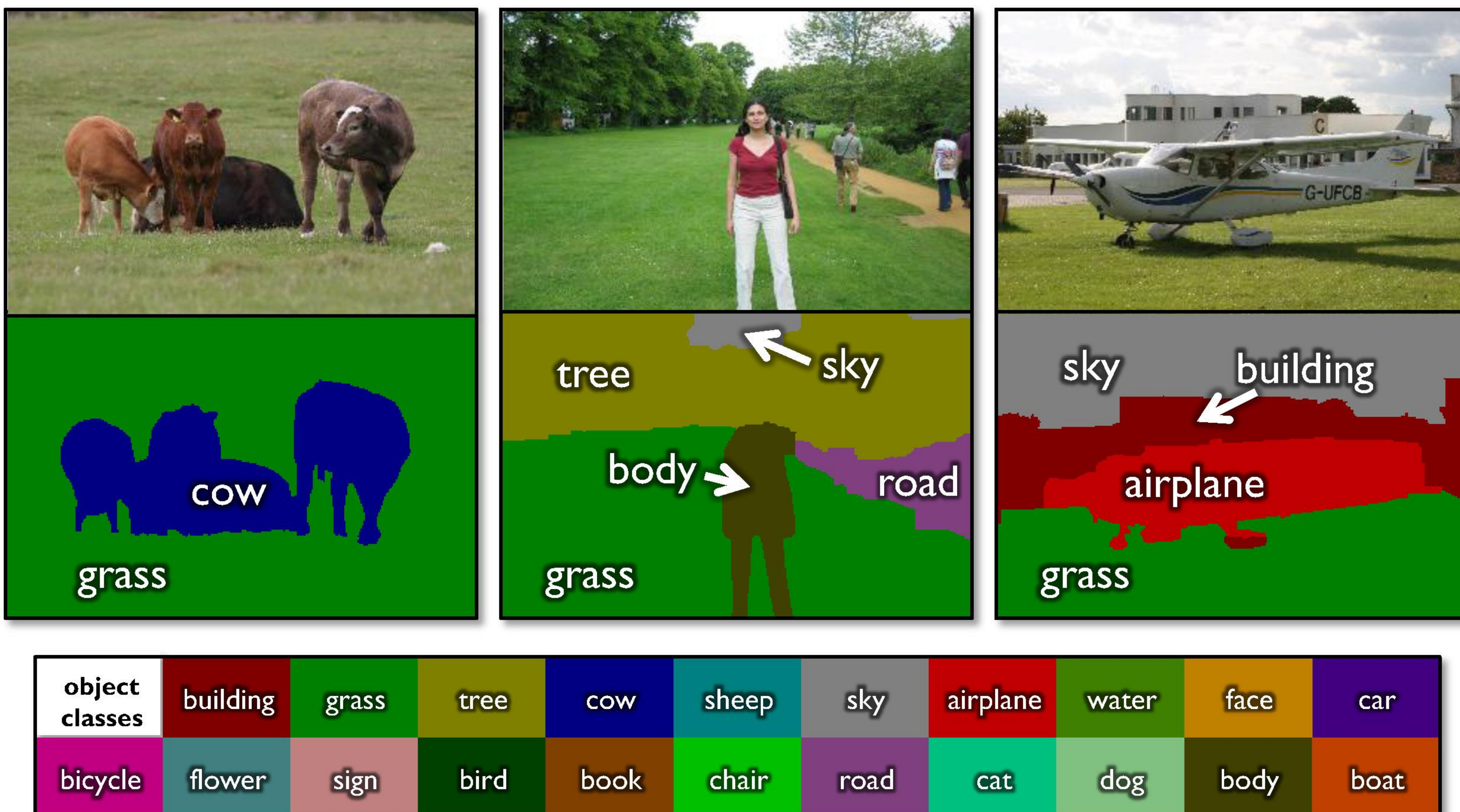
Supervised image segmentation

- Bounding-box detection is problematic for *articulated objects*:



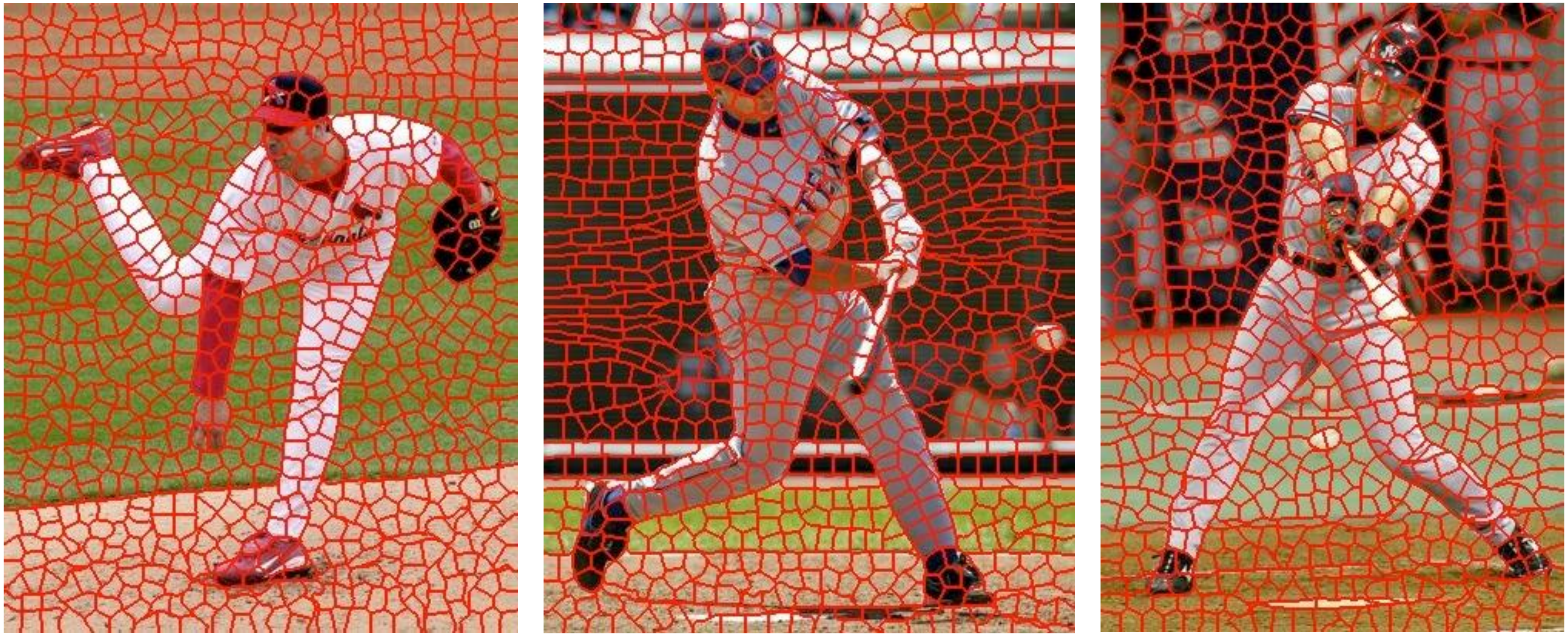
Supervised image segmentation

- Bounding-box detection is problematic for *articulated objects*
- To resolve this issue, we could try to assign a class to each pixel in the image:



Supervised image segmentation

- Because images are very large, one often first constructs *superpixels*:



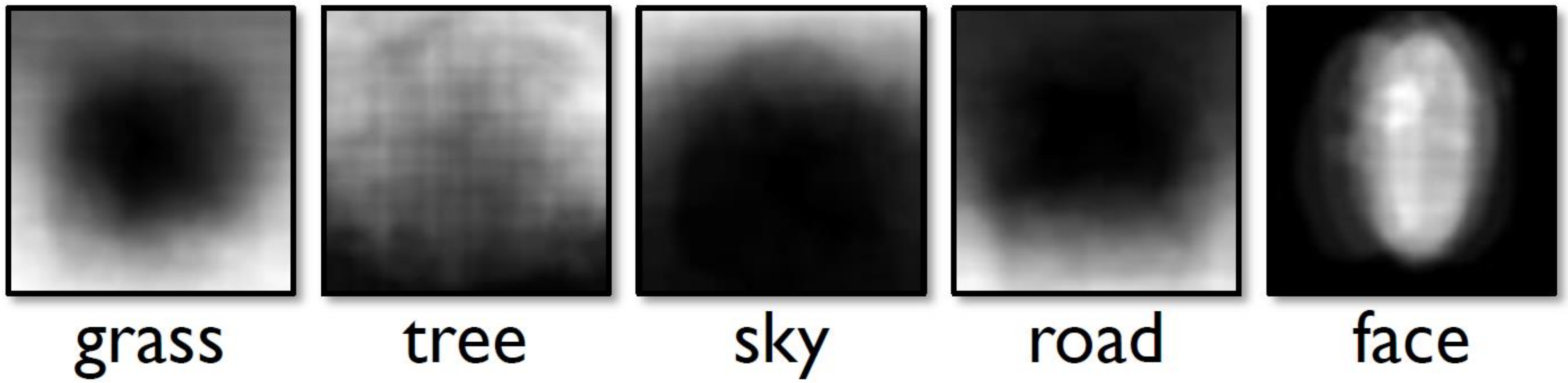
- Simple approach to finding superpixels: Cluster per-pixel R,G,B,X,Y-features

Supervised image segmentation

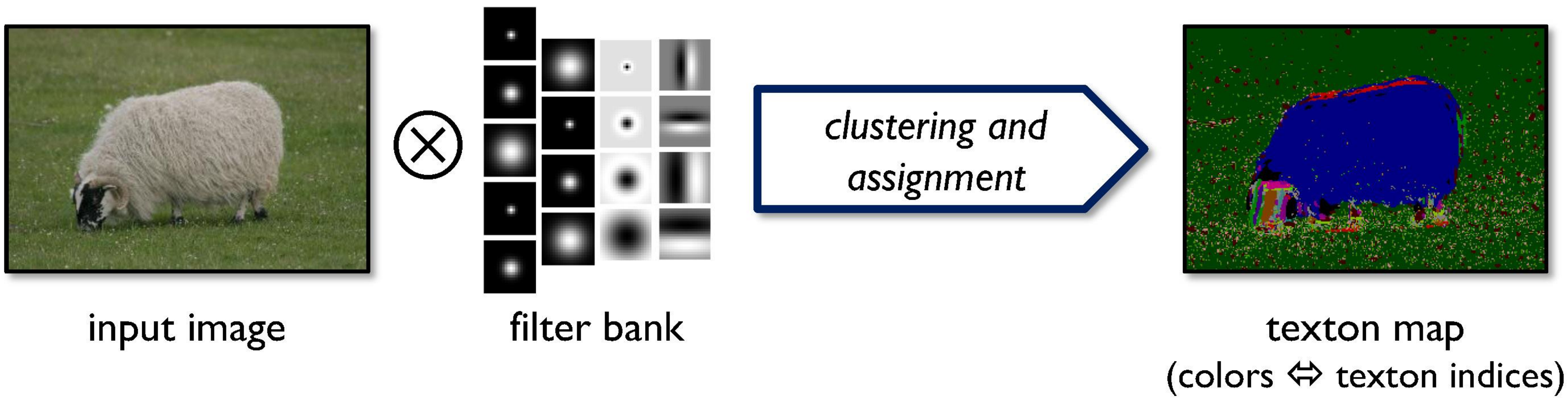
- Commonly used *features* to represent superpixels:
  - Texture layout (textons), color, edge presence, superpixel location, *etc.*
- Commonly used *classifiers* to assign superpixels to a class:
  - Linear classifiers such as *logistic regression* and *support vector machines*
  - Ensemble approaches such as *AdaBoost*
  - Classifiers that exploit *structure* in the label field (conditional random fields)
- Often, we also incorporate a *location prior* in the segmentation algorithm

Example: TextonBoost

- Location:



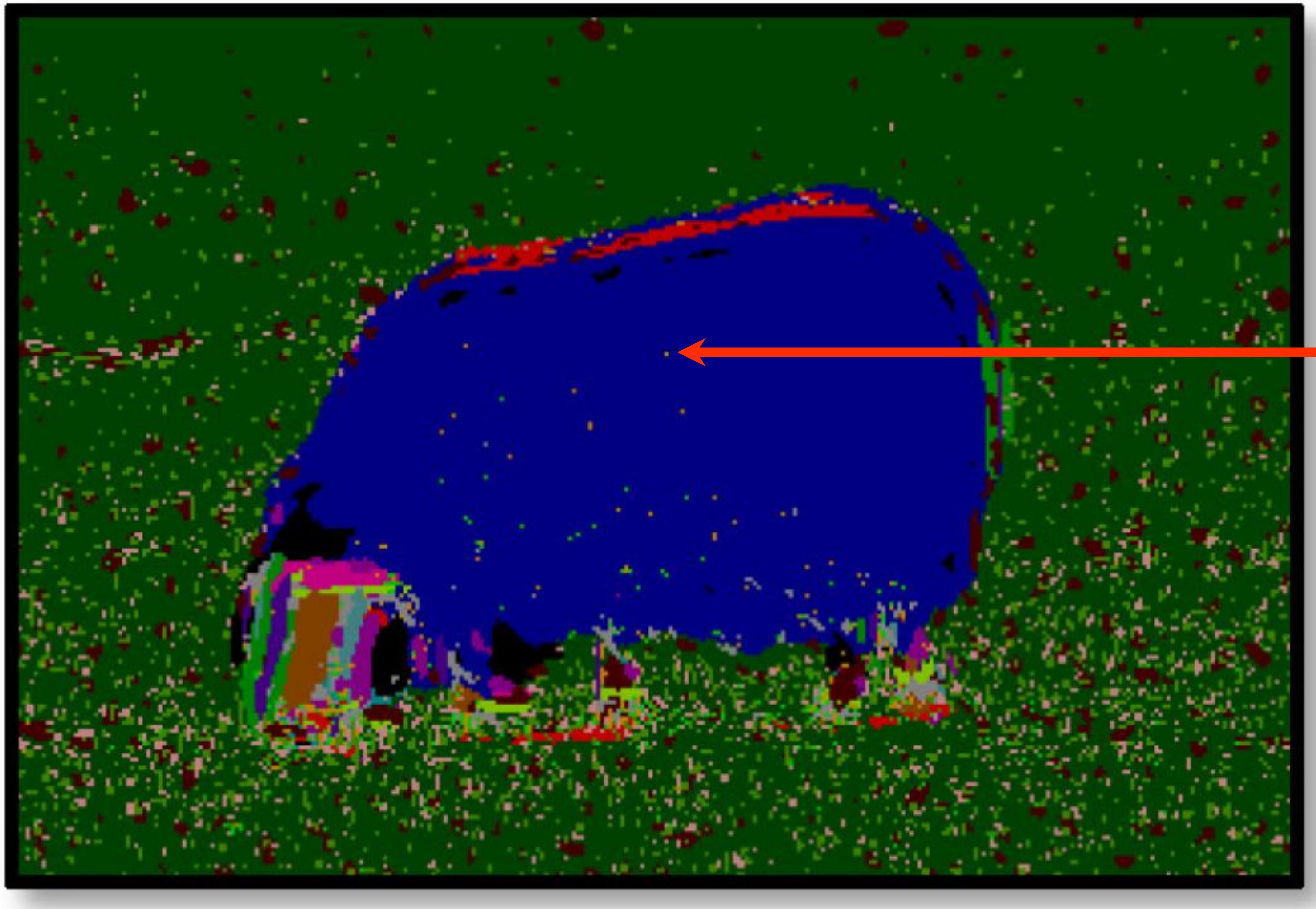
- Texture:



Per pixel class posteriors from *texton map* using boosted weak classifiers

Example: TextonBoost

- The resulting label image looks quite noisy:



Is this pixel a sheep or not?

Supervised image segmentation

- We know that the *label field* is generally *smooth*: changes are uncommon

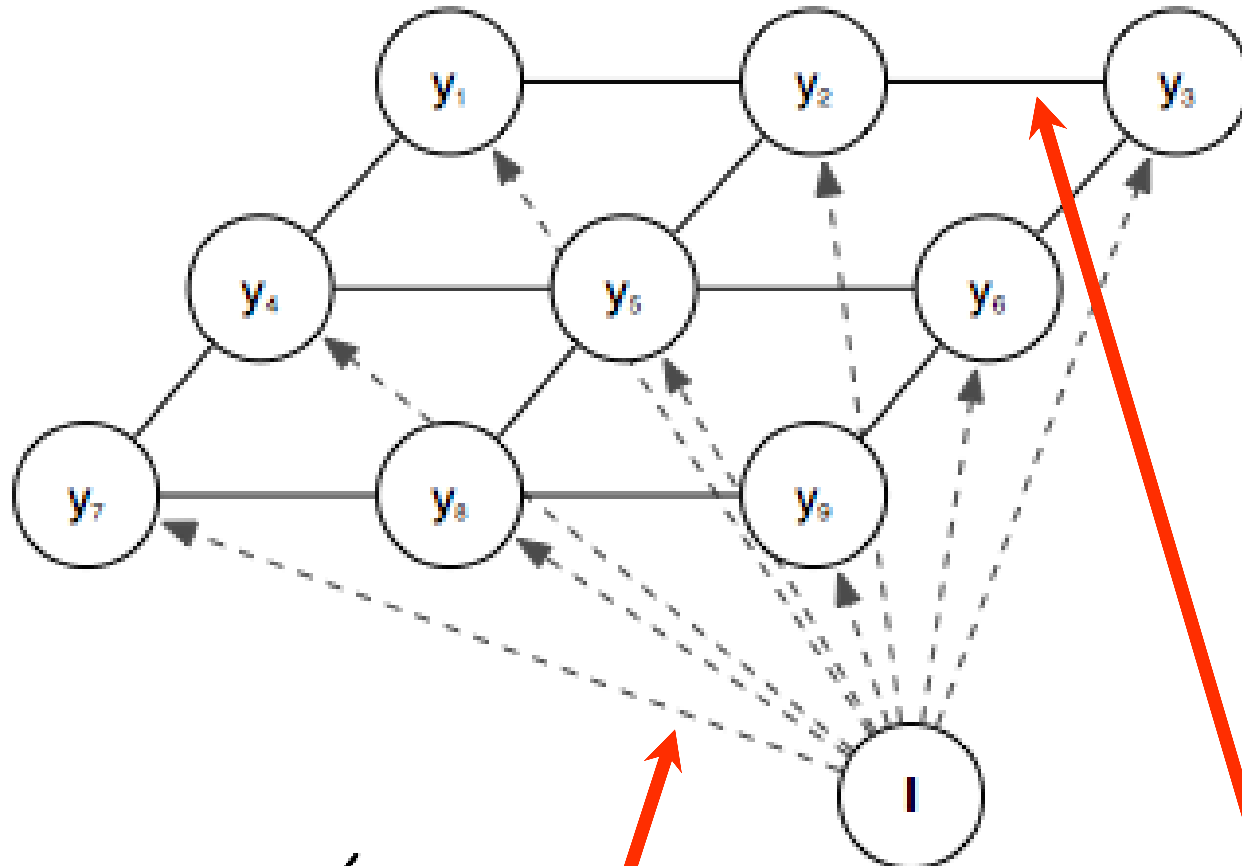
- We know that some labels are *incompatible*: “people do not walk on water”

- Conditional and Markov Random Fields allow us to incorporate such things, *e.g.*, to penalize different neighboring labels *except* when there is an edge:

original image

edge potentials

Conditional Random Field



$$p(\mathbf{y}|\mathbf{I}) = \frac{1}{Z(\mathbf{I})} \exp\left(\sum_{i\in V}\Phi(y_i;\mathbf{I}) + \sum_{(i,j)\in E}\Psi(y_i,y_j;\mathbf{I})\right)$$

label field

normalization

exponentiate

score of “regular” classifier

label compatibility function

Edge potential

- Example of an edge potential*: $\Psi(y_i, y_j; \mathbf{I}) = \lambda y_i y_j$

Ising model
(encourages similar labeling)

- When is an Ising model inappropriate?
  - At locations where an image edge is present!

- Alternative edge potential: $\Psi(y_i, y_j; \mathbf{I}) = \lambda \exp\left(-\frac{1}{2\sigma^2}(\mathbf{I}_i - \mathbf{I}_j)^2\right) y_i y_j$
  - If two pixels are similar, this gives a high penalty for different labels

* Assuming label y is {-1, +1}

Inference

- Given the CRF model, we need to find the *most likely* labeling (MAP assignment)

- We can do this by maximizing the logarithm of the likelihood:

$$\max_{\mathbf{y}} \left[\sum_{i \in V} \Phi(y_i; \mathbf{I}) + \sum_{(i,j) \in E} \Psi(y_i, y_j; \mathbf{I}) - \cancel{\log Z(\mathbf{I})} \right]$$

- How many possible labelings are we maximizing over?

  - For a binary classification problem, there are already two to the power of the number of (super)pixels possible label fields

Inference

- *Iterated conditional modes* (ICM) iteratively maximizes over one of the labels:

$$\max_{y_k} \left[\sum_{i \in V} \Phi(y_i; \mathbf{I}) + \sum_{(i,j) \in E} \Psi(y_i, y_j; \mathbf{I}) - \log Z \right] =$$

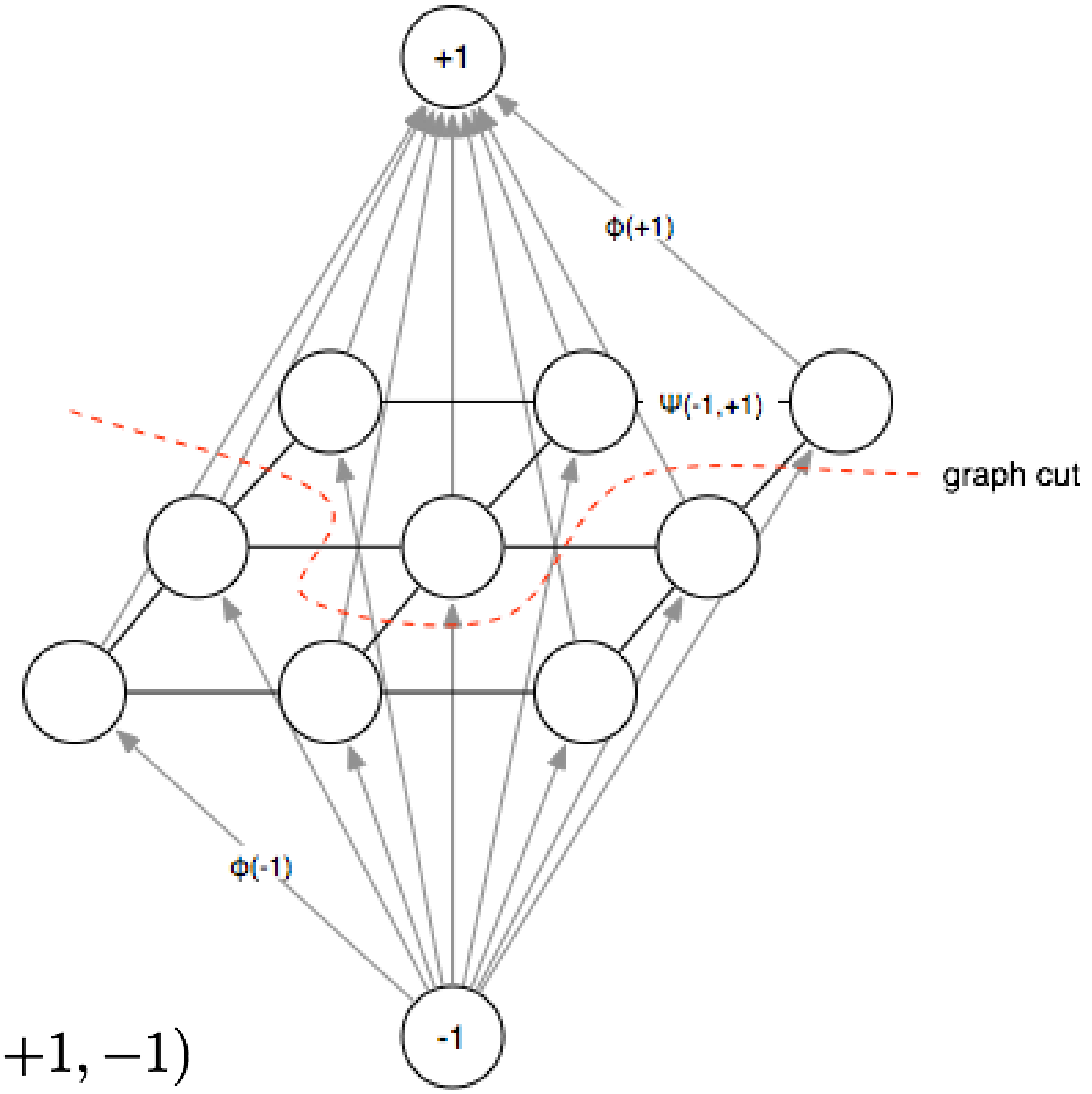
for a lattice, only need to compute 5 x #-of-labels terms

- Label field can be initialized to labels that maximize the unary potentials

- This procedure converges to a local maximum of the log-likelihood

Inference

- MAP solution for *binary pairwise MRF*: $\min_{\mathbf{y}} \sum_{i \in V} \phi(y_i; \mathbf{I}) + \sum_{(i,j) \in E} \psi(y_i, y_j; \mathbf{I})$
- Identical to finding *minimal graph cut* that separates *source* 0 from *sink* 1:



* This assumes that $\psi(-1,+1) = \psi(+1,-1)$

Example: segmenting occluded people

Model estimates person locations and appearances

Post-processing with CRF to segment objects

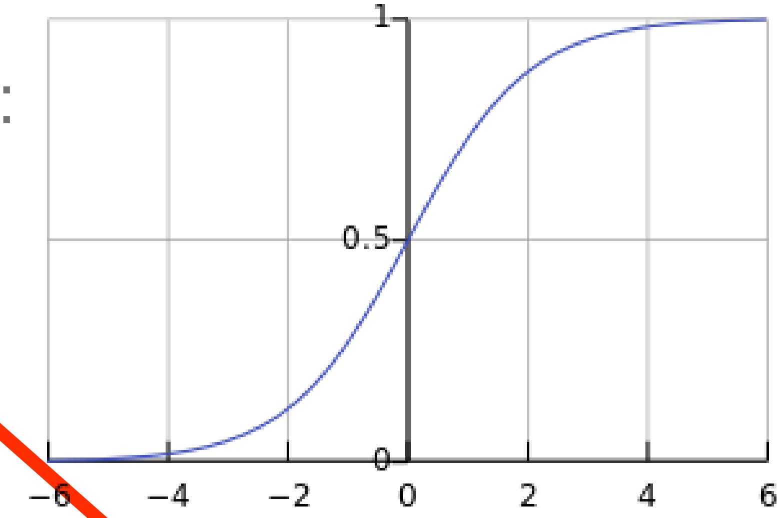
Per pixel classes distributions

(comparison tracker)

Kooij, Julian FP, Gwenn Englebienne, and Dariu M. Gavrila. "Identifying multiple objects from their appearance in inaccurate detections." *Computer Vision and Image Understanding* 136 (2015): 103-116.

Discriminative Random Fields

- A Conditional Random Field that involves a unary factor:



$$\phi(y_i; \mathbf{I}) = -\log(1 + \exp(-y_i \mathbf{w}^{\mathrm{T}} f_i(\mathbf{I})))$$

model parameters

logistic regressor

image features near site *i*

- And a pairwise factor (interaction potential) that is modeled as follows:

$$\psi(y_i, y_j; \mathbf{I}) = K y_i y_j + (1-K)\left(2\left(\frac{1}{1+\exp(-y_i y_j \mathbf{v}^{\mathrm{T}} g_{ij}(\mathbf{I}))}\right) - 1\right)$$

logistic regressor
(scaled between -1 and +1)

model parameters

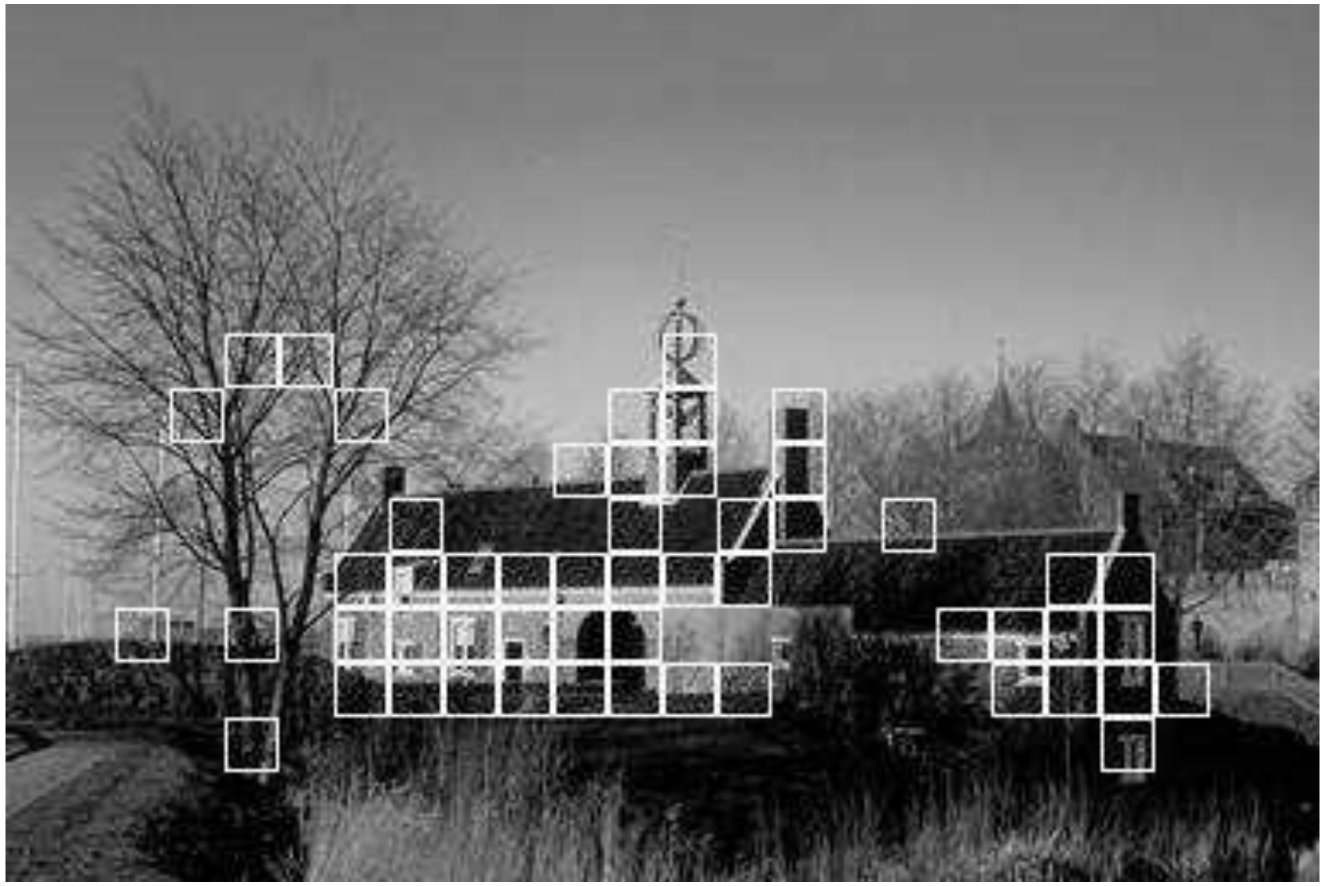
data-independent term

same label?

image features for pair *(i, j)*

Example: Discriminative Random Fields

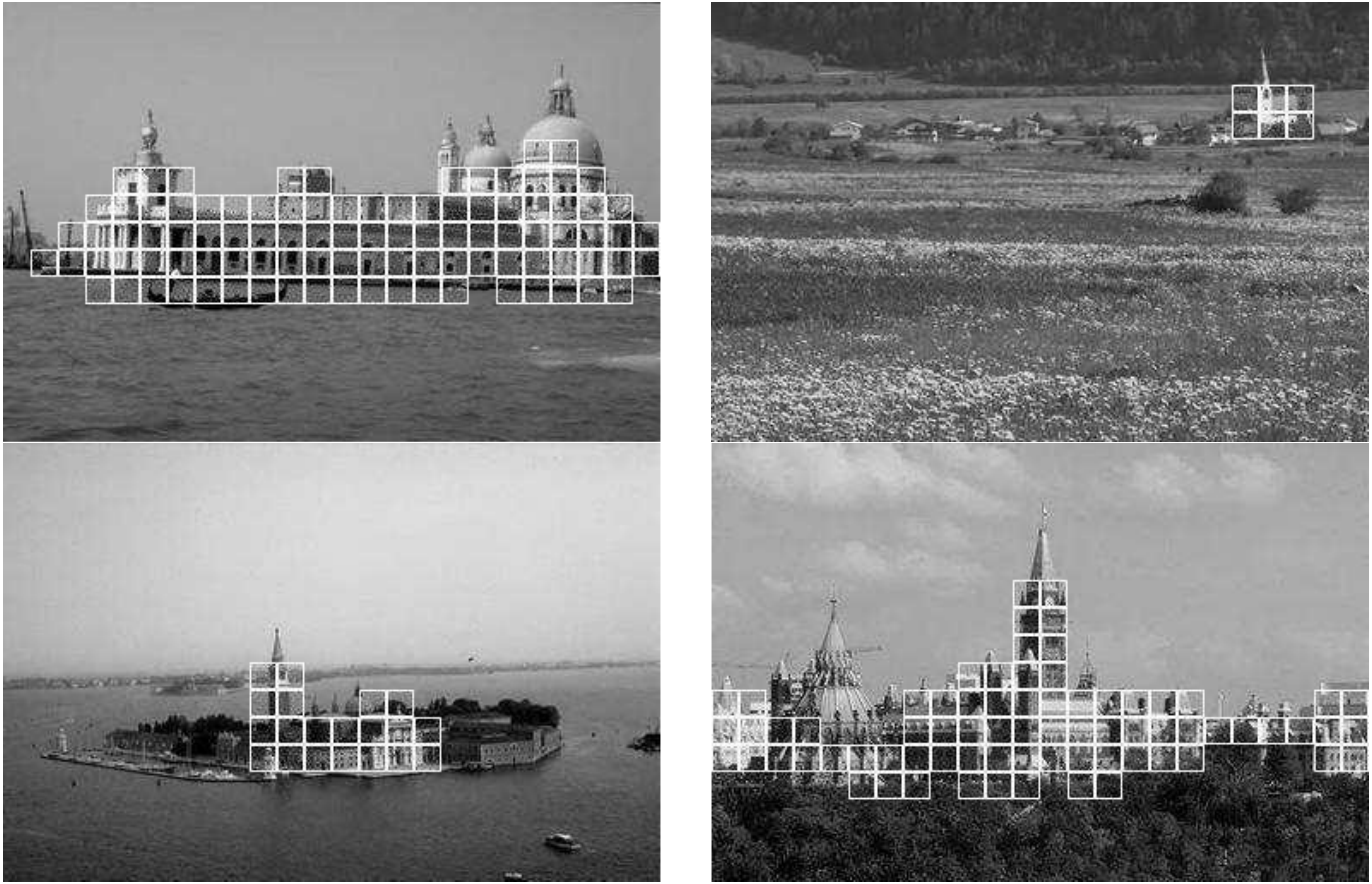
- The DRF graph is a lattice over neighboring mage patches
- Recognition of “man-made” structures, with and without pairwise factors:



logistic regression

DCRF

Example: Discriminative Random Fields



Markov Random Fields

Markov Random Fields

- In conditional random fields, we defined a distribution of label fields
- In some problems, we want to define a distribution over images $p(\tilde{\mathbf{I}})$:

Markov Random Fields

- In conditional random fields, we defined a distribution of label fields
- In some problems, we want to define a distribution over images $p(\tilde{\mathbf{I}})$:
  - Assume our image is corrupted by Gaussian noise
  - We can then try to infer the non-corrupted image by maximizing:

$$p(\tilde{\mathbf{I}}|\mathbf{I}) \propto p(\mathbf{I}|\tilde{\mathbf{I}})p(\tilde{\mathbf{I}})$$

non-corrupted image

observed, corrupted image

corruption model
(Gaussian noise)

prior over images

  - Markov Random Fields are an appropriate model for $p(\tilde{\mathbf{I}})$

Markov Random Fields

- An example of a Markov Random Field is the following model:

$$p(\tilde{\mathbf{I}}) = \frac{1}{Z} \exp\left(\sum_{i \in V} \Phi(\tilde{\mathbf{I}}_i) + \sum_{(i,j) \in E} \Psi(\tilde{\mathbf{I}}_i, \tilde{\mathbf{I}}_j) \right)$$

- Key difference with CRFs: we do not *condition* on the image

- This makes inference in MRFs is even harder than in CRFs. Why?
  - MRFs need to normalize over *all possible images* instead of all possible labelings
  - However, similar inference algorithms as before are generally be applied

Example: Simple denoising MRF

- Example of using a simple MRF over binary (-1, +1) images for denoising:

$$p(\tilde{\mathbf{I}}|\mathbf{I}) \propto p(\mathbf{I}|\tilde{\mathbf{I}})p(\tilde{\mathbf{I}})$$

$$p(\tilde{\mathbf{I}}|\mathbf{I}) \propto \overbrace{\exp\left(\eta \sum_{i\in V} \tilde{\mathbf{I}}_i \mathbf{I}_i\right)}^{\text{Corruption model}} \overbrace{\exp\left(\alpha \sum_{i\in V}\tilde{\mathbf{I}}_i + \beta \sum_{(i,j)\in E} \tilde{\mathbf{I}}_i\tilde{\mathbf{I}}_j\right)}^{\text{MRF prior}}$$

"cost" for changing a pixel value

prior over individual pixel

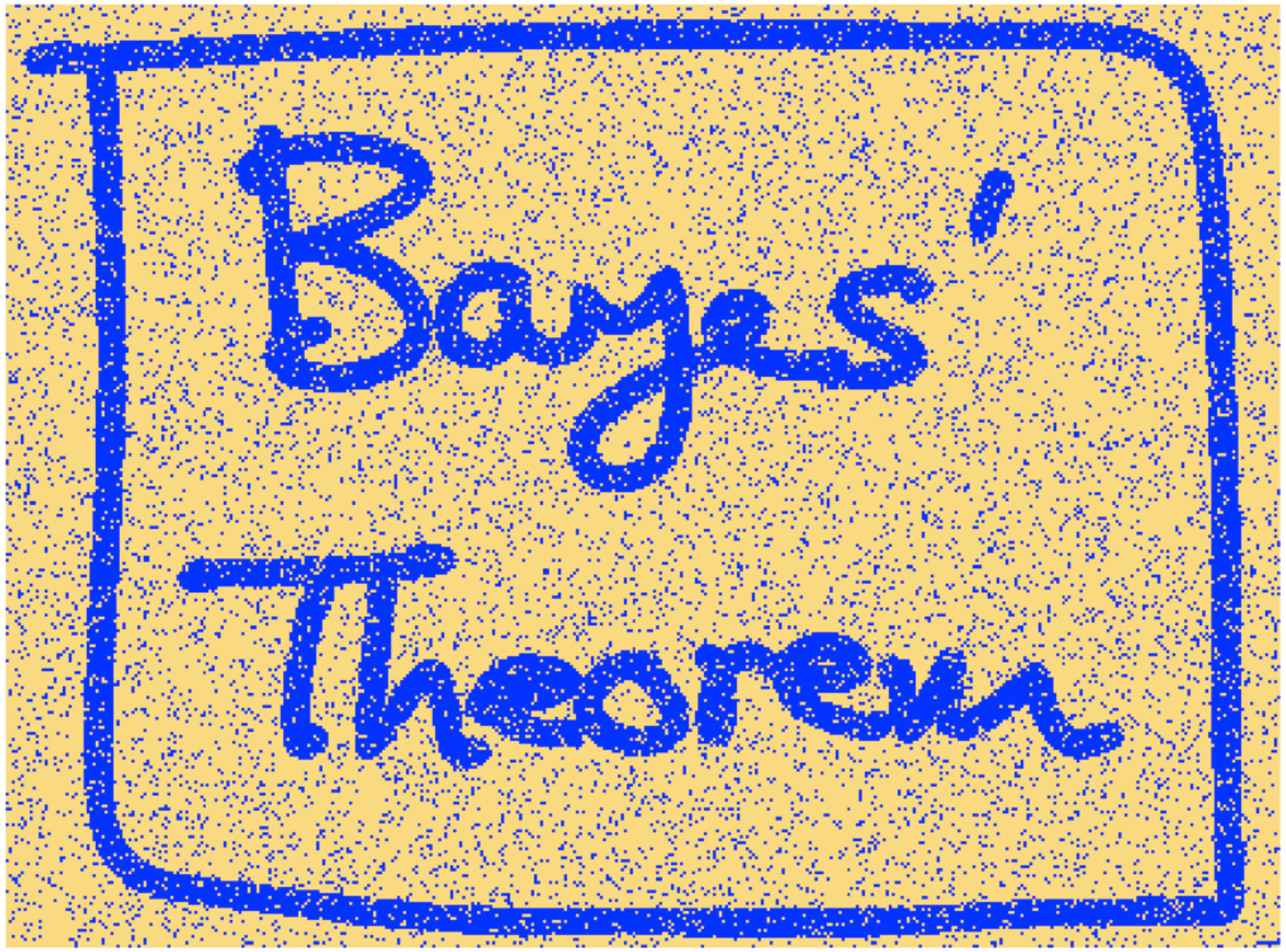
prior over neighboring pixels

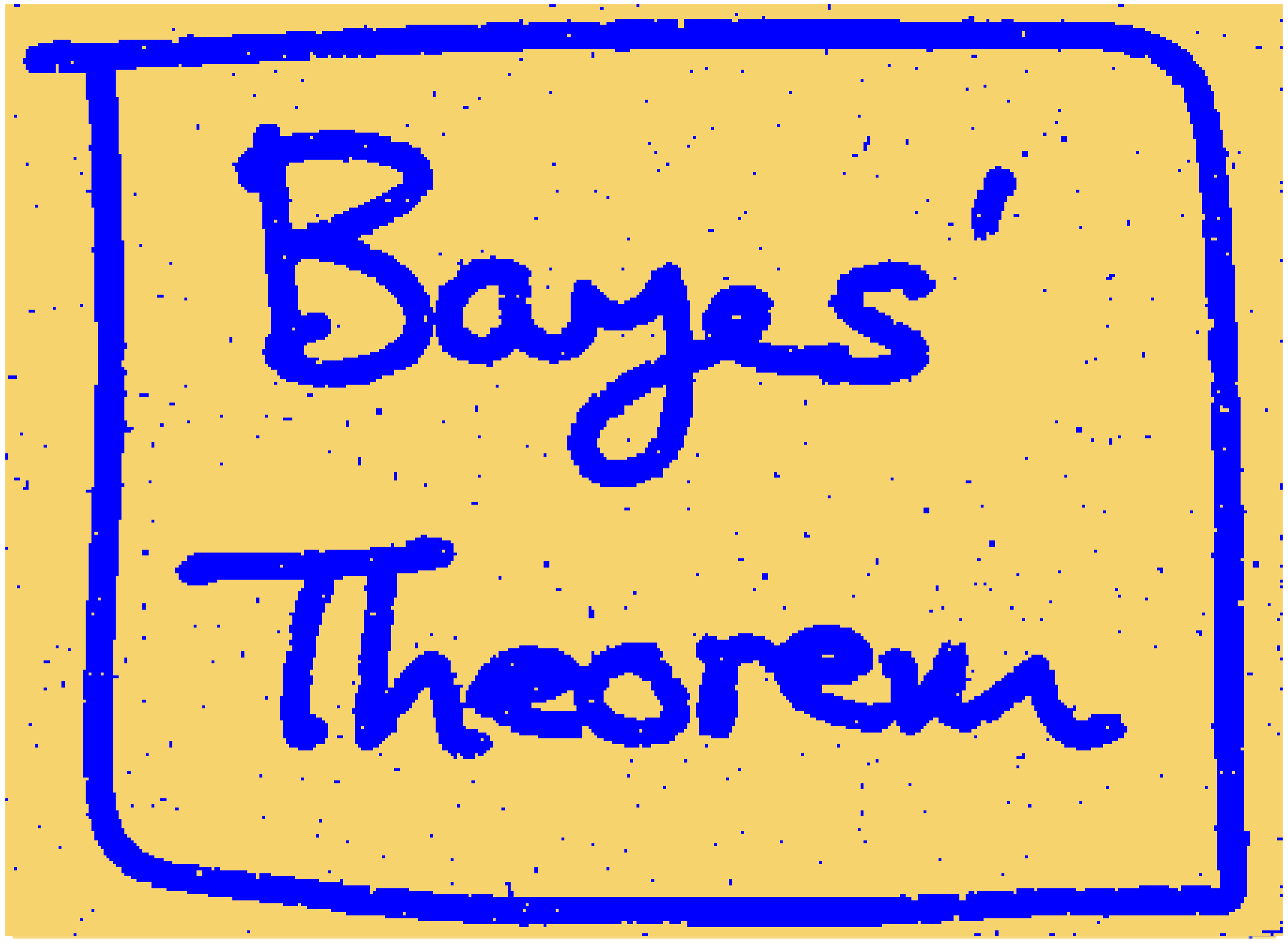
- Note: MAP inference for this simple MRF is similar to the simple CRF earlier

Example: Simple denoising MRF

- Example of using a simple MRF over binary (-1, +1) images for denoising:

$$p(\tilde{\mathbf{I}}|\mathbf{I}) \propto \exp\left(\eta \sum_{i \in V} \tilde{\mathbf{I}}_i \mathbf{I}_i\right) \exp\left(\alpha \sum_{i \in V} \tilde{\mathbf{I}}_i + \beta \sum_{(i,j) \in E} \tilde{\mathbf{I}}_i \tilde{\mathbf{I}}_j\right)$$





Graph Cut (MAP)

Example: Fields of Experts

- FoE models each potential using a product of Student-t distributions:

$$p(\mathbf{I};\Theta) = \frac{1}{Z(\Theta)} \exp\left(\sum_{k=1}^{K}\sum_{n=1}^{N} \log\left(1 + \frac{1}{2}(\mathbf{J}_n^{\mathrm{T}}\mathbf{I}_{(k)})^2\right)^{-\alpha_i}\right)$$

sum over multiple filters and patches

log of Student-t distribution
(heavy-tailed distribution)

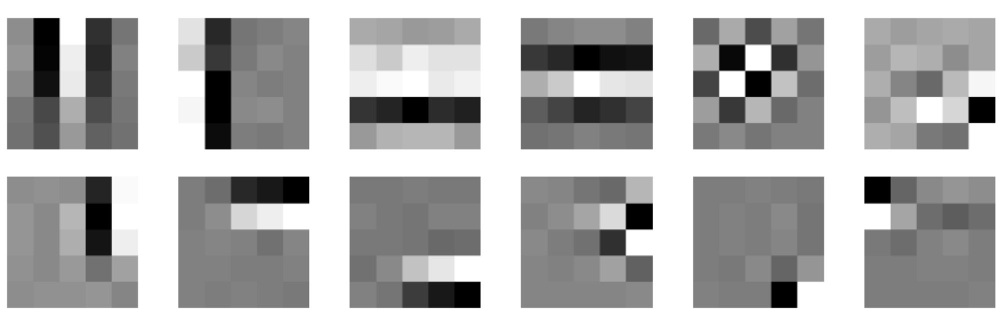
filtered *k*-th image patch $\mathbf{I}_{(k)}$

- Patches are overlapping pixel neighborhoods (unlike pair-wise MRF)
- Intuitively, the models assigns a probability to an image as follows:
  - Patch gets high probability if it does not resemble any filter (zero inner product)
  - Image gets high probability if many of the patches get high probability

Example: Fields of Experts

- Learning expert filters independently vs. within Markov Random Field
- Train experts on generic image database
- **Q: Why will we not learn trivial filters that are all zero?**

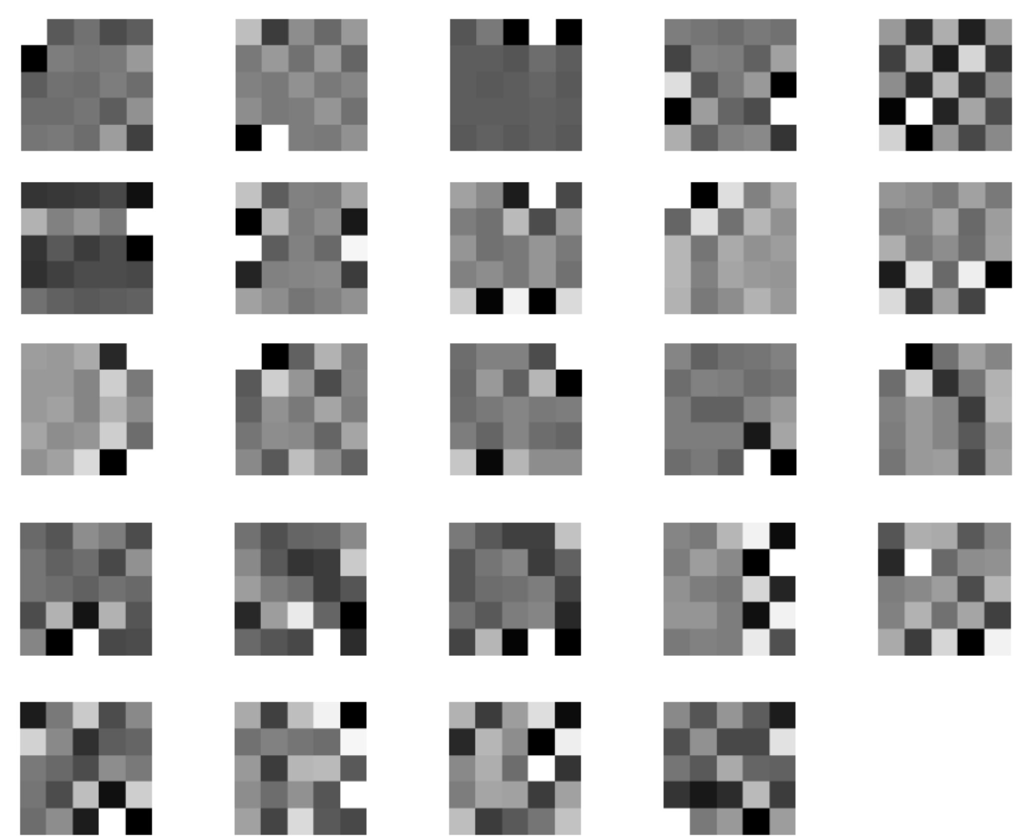
Using independent image patches



FoE has no clear structure?

Filters account for statistical dependency in overlapping patches?

Image patches as potentials in MRF



Denoising using FoE

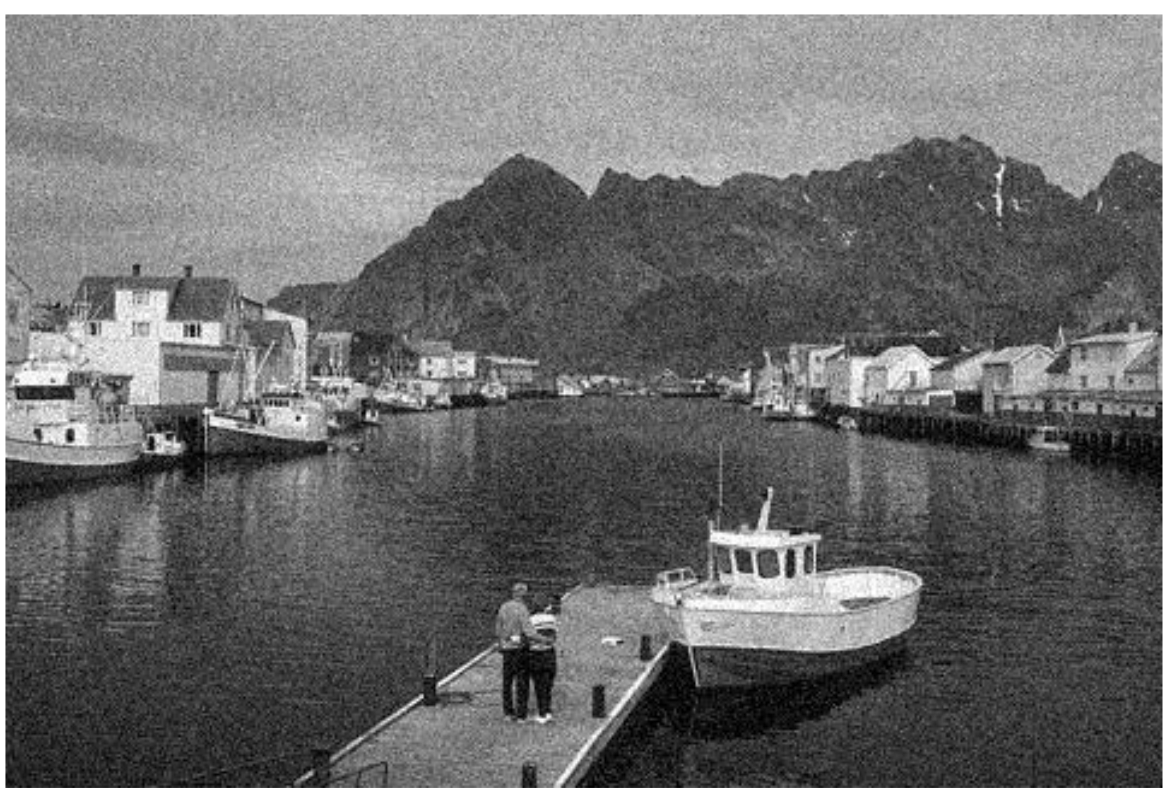
- Using the FoE as image prior, denoising can be phrased as a MAP-problem:

$$\max_{\tilde{\mathbf{I}}} \ \mathcal{N}(\mathbf{I}|\tilde{\mathbf{I}},\sigma^2)p(\tilde{\mathbf{I}})$$

corruption model
(Gaussian noise)

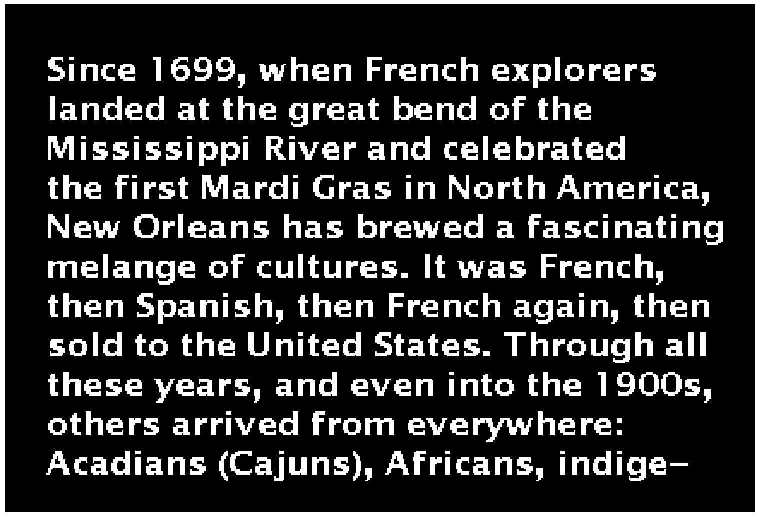
field of experts prior

- The result of the MAP-inference has removed Gaussian noise from the image:



Inpainting using FoE

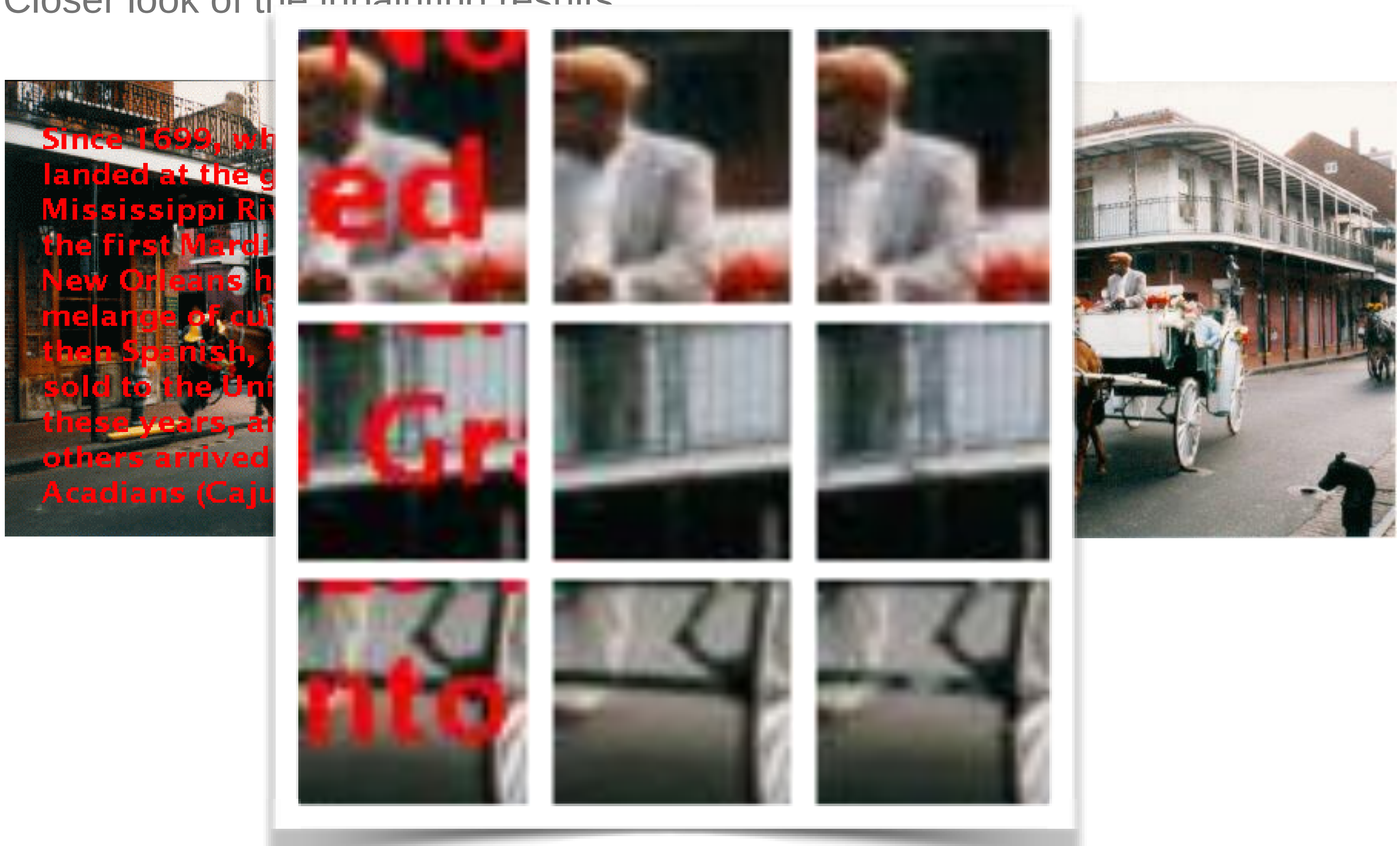
- Given a mask image, inpainting can also be phrased as a MAP-problem:



- Example of inpainting to remove text from an image:

Inpainting using FoE

- Closer look of the inpainting results:



Inpainting using FoE

- Closer look of the inpainting results:

- Can you give an intuition for what the FoE model has learned?
  - Hard to say, but for instance: Edges generally continue in same direction

Deep-learning for Semantic Segmentation

Semantic segmentation state-of-the-art

- Semantic Segmentation: Label each pixel with a semanticly meaningful class
- Various applications, and variations (class level, instance level, part level)
- Large datasets with accurate manually annotated data have been created

CITYSCAPES DATASET

www.cityscapes-dataset.com

5000 high-quality frames
20000 weak annotated frames

Semantic segmentation state-of-the-art

- Driving around TU Delft campus with our (almost) self-driving Prius ...

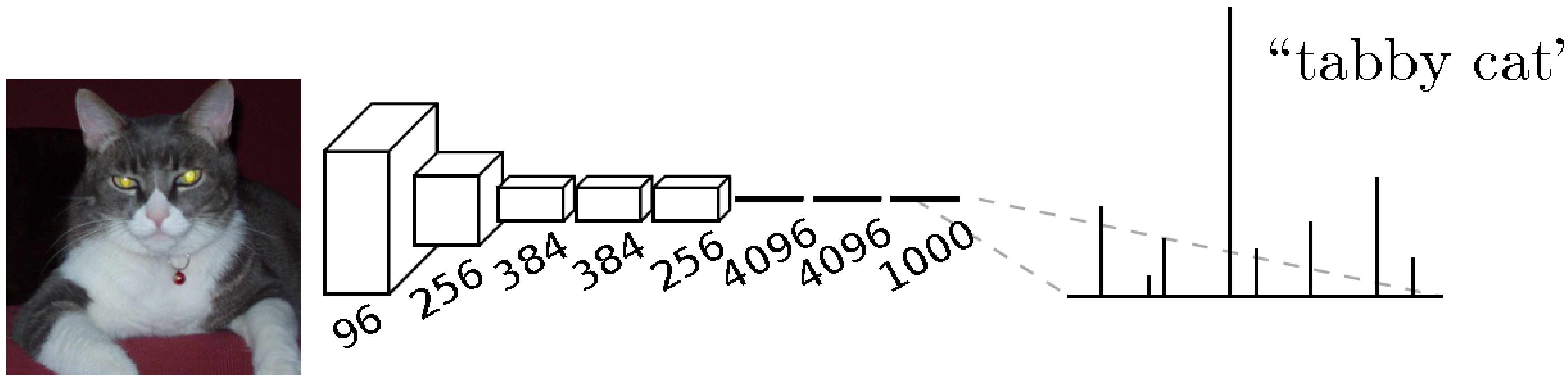
Semantic segmentation state-of-the-art

- Driving around TU Delft campus with our (almost) self-driving Prius ...

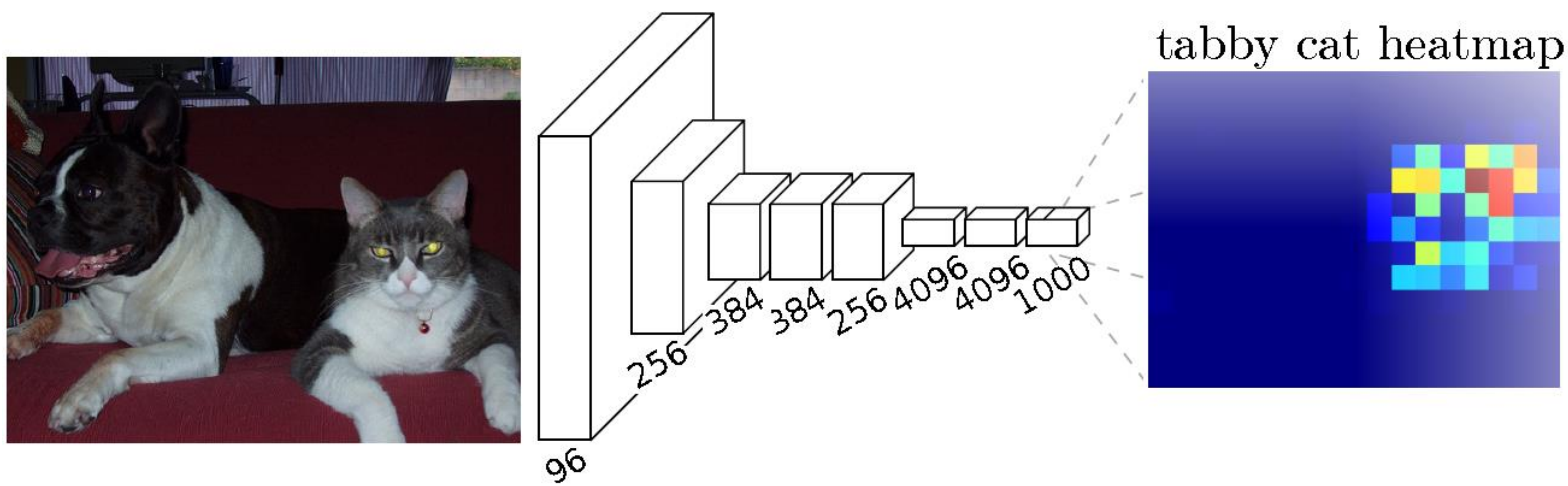
Created with “Pyramid Scene Parsing Network”, H. Zhao et al., CVPR’17. Trained on Cityscapes

CNNs for Semantic Segmentation

- Convolutional Neural Networks (CNNs) for image classification



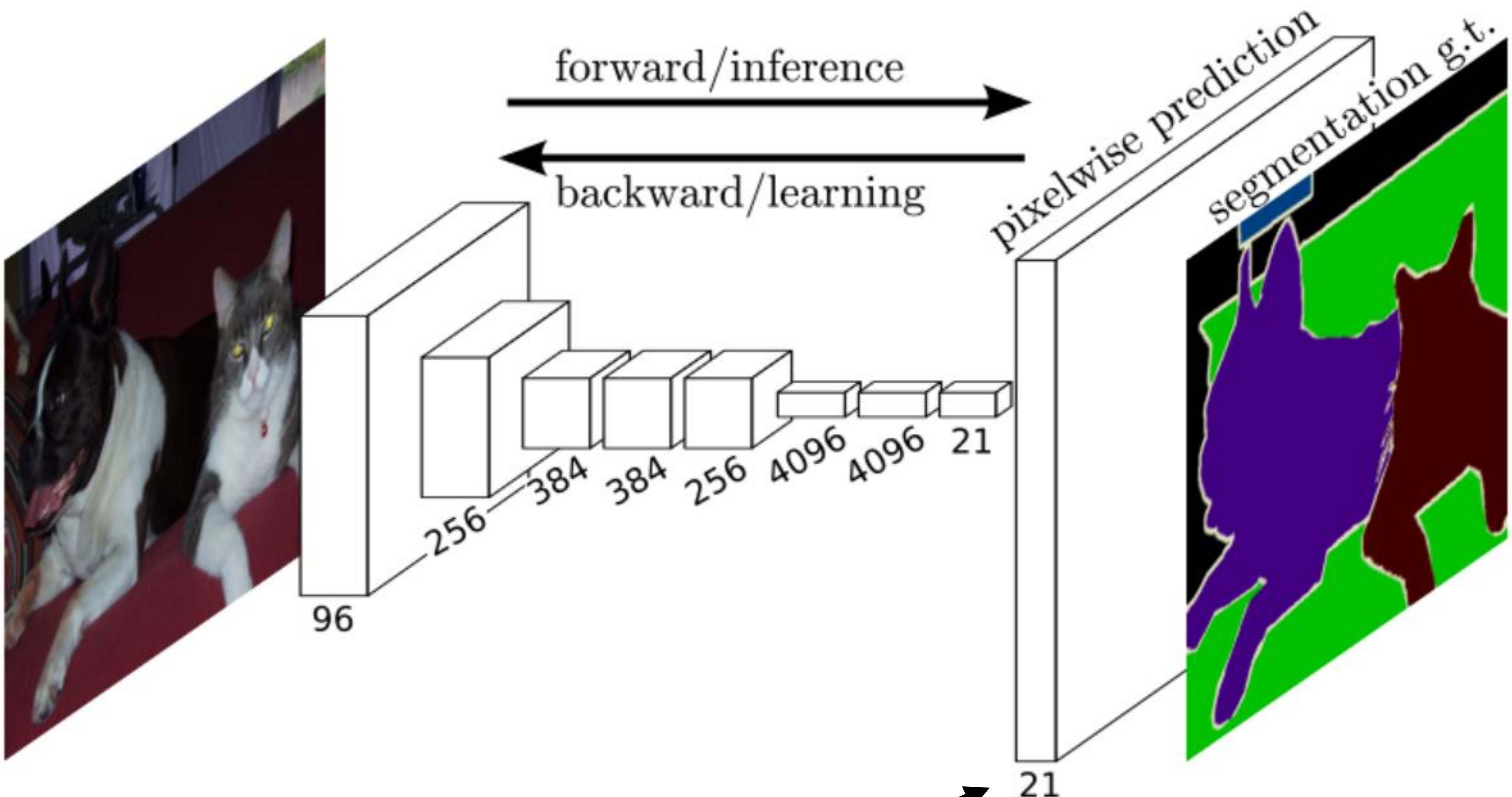
- **Fully Convolutional Network** (FCN) turns last layers in convolutions too



“Fully Convolutional Networks for Semantic Segmentation”, Long et al, ICCV 2014

Fully Convolutional Networks

- Add “deconvolutional” layers: upscale feature maps to per-pixel classifiers

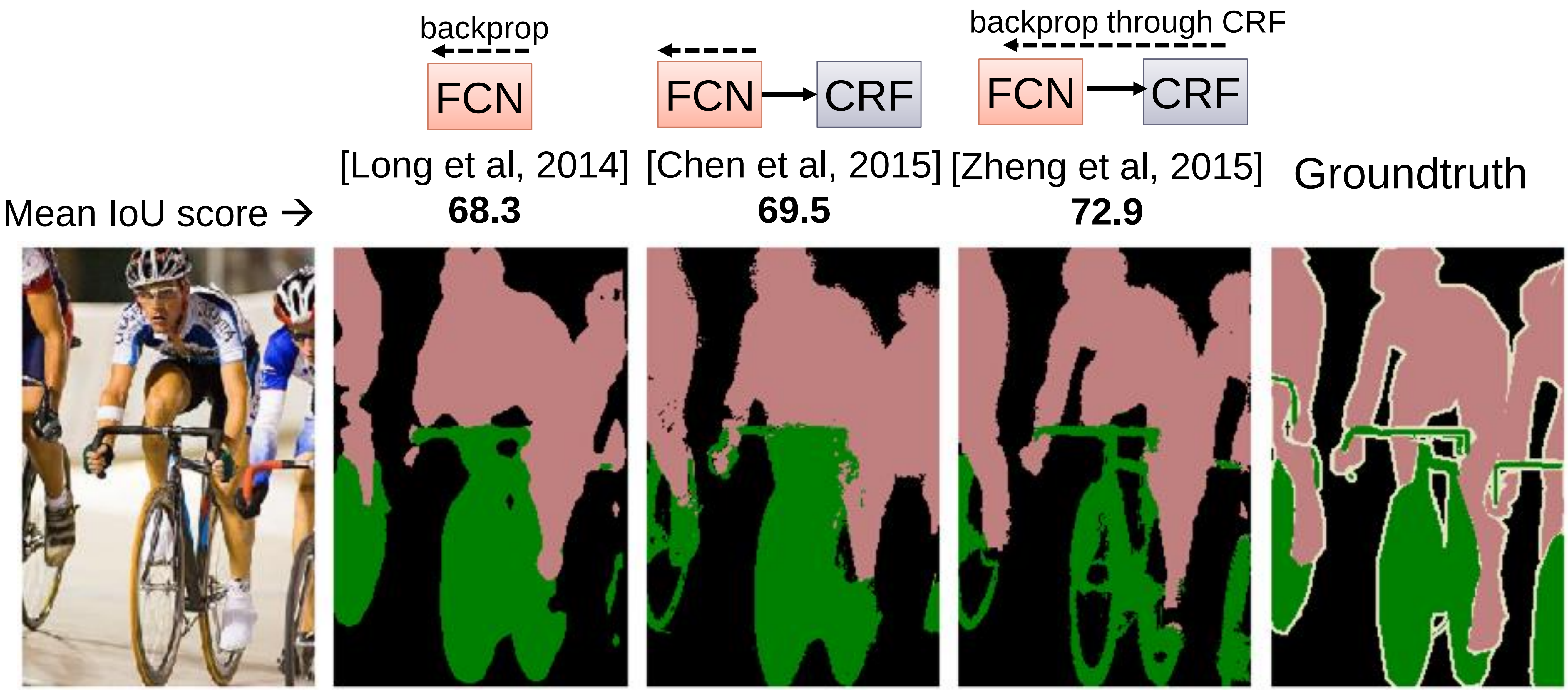


Per-pixel 21 class responses, FCN takes argmax

“Fully Convolutional Networks for Semantic Segmentation”, Long et al, ICCV 2014

Improving FCNs with CRF-as-RNN

- CRF inference as differentiable operations in a Recurrent Neural Network
- Perform backpropagation “through” a CRF, optimize FCN+CRF combination



“Conditional Random Fields as Recurrent Neural Networks”, Zheng et al., ICCV’15

Exploiting Superpixels in CRF-RNN

- CRF-RNN formulation can also benefit from other potentials, e.g. Superpixels
- Train and test CRF-RNN network, enforcing consistency over Superpixel region

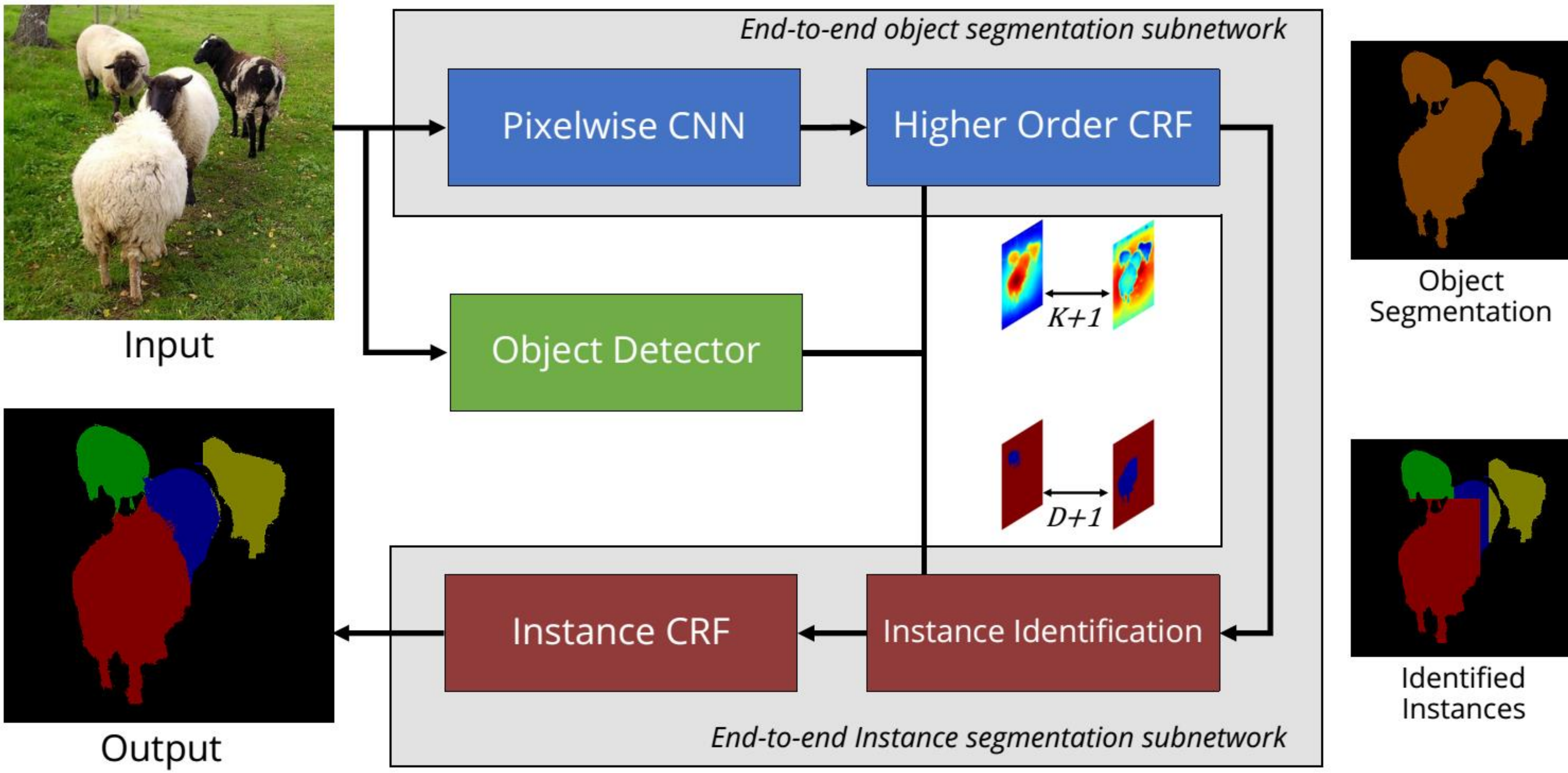
Exploiting Object Detections in CRF-RNN

- Object Detections Bounding Boxes can also define potentials
- Improves semantic class segmentation, but also instance-level segmentation

“Bottom-up Instance Segmentation using Deep Higher-Order CRFs”, Arnab et al., BMVC’16

Instance segmentation

- Revisiting the sheep ...



“Bottom-up Instance Segmentation using Deep Higher-Order CRFs”, Arnab et al., BMVC’16

Reading material:

- Section 3.7 and 10.5 and Appendix B
- Paper by Kumar and Hebert (2003)
- Paper by Roth and Black (2009)