Combined Object Categorization and Segmentation with an Implicit Shape Model

Bastian Leibe
Ales Leonardis
Bernt Schiele

Slides/images sourced off:
http://crcv.ucf.edu/courses/CAP5415/Fall2012/Lecture-4-Harris.pdf
Leibe et al. Combined Object Categorization and Segmentation with an Implicit Shape Model
Leibe et al. Interleaved Object Categorization and Segmentation
http://www.cse.psu.edu/~rtc12/CSE598G/introMeanShift_6pp.pdf

Presented by:
Mukul Sati
2/18/2016

Combined Object Categorization and Segmentation with an Implicit Shape Model

- Traditionally (back then), no object information used in segmentation.
- Segmentation done primarily on low level features in an unsupervised setting.
- Human vision - object recognition is intertwined with segmentation.

Overview

- Goal – object categorization and segmentation in the wild.
- Steps:
  - Training:
    - Learn a code-book of local appearance for each object class.
    - Learn implicit shape model for the object classes using the local code-books.
  - Object Detection:
    - Extract test image patches and match them to code-book entries for each object.
    - Each “activated” code-book entry votes for the object and the object’s center.
  - Segmentation:
    - Per object segmentation masks + per-pixel confidence estimate for segmentation.

Overview

Input

Training

Training images

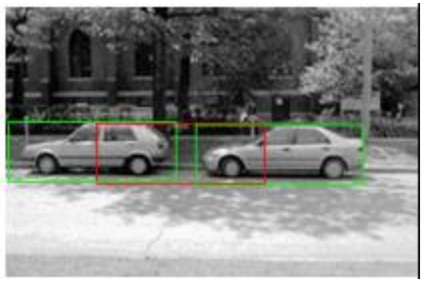
Segmentations

Car

Object Labels

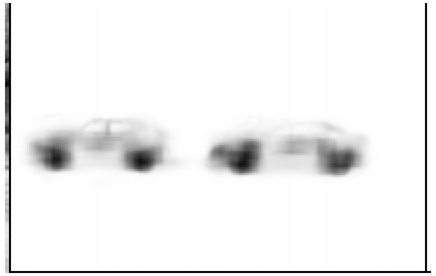
+ Perhaps object centers

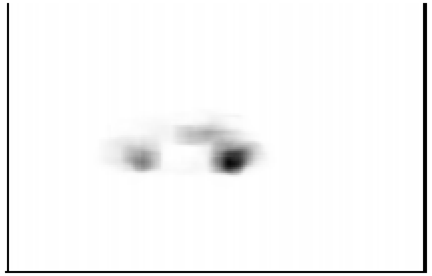
Testing



Novel Image (ignore bounding boxes)

Output

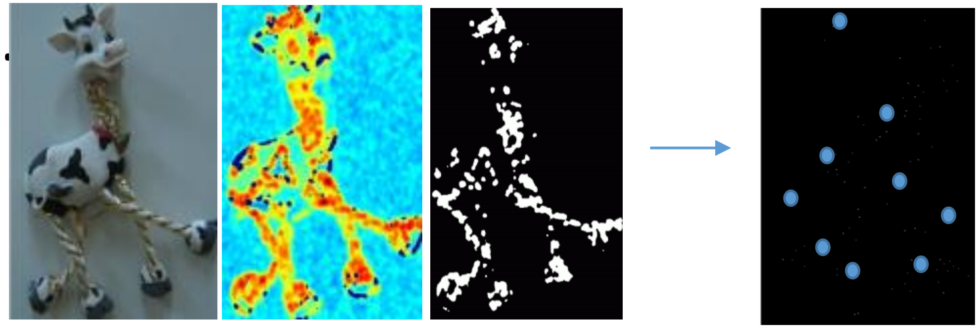




Object Hypotheses + Per pixel background / foreground estimate

Local code-book creation – Interest Point Detection

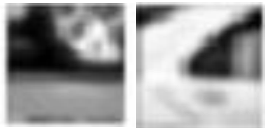
- First, detect interest points using Harris detectors
  - Compute sample covariance matrix M for (2D) intensity gradient at each pixel.
  - Compute corner response (C = f(EigenVals(M))).
  - Interest points – maxima of thresholded C.
  - Take 25X25 pixel patches.



Local code-book creation – Agglomerative Clustering

For a pair of patches (p,q): Normalized Grayscale correlation = $NGC(p,q) = \dfrac{\sum_i (p_i - \overline{p_i})(q_i - \overline{q_i})}{\sqrt{\sum_i (p_i - \overline{p_i})^2 \sum_i (q_i - \overline{q_i})^2}}$

NGC = 0.9



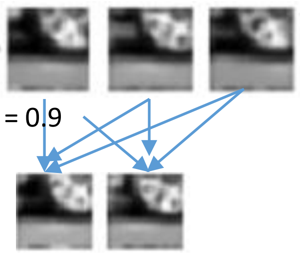
NGC = 0.4

Start with each patch as a separate cluster. Merge two clusters if they are similar. $similarity(C_1, C_2) = \dfrac{\sum_{p \in C_1, q \in C_2} NGC(p,q)}{|C_1| \times |C_2|} > t$

(Average of pair-wise NGCs of each pair in C1 X C2. t = 0.7 for their experiments)

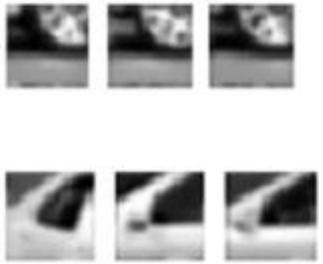
C1

$NGC_{0,0} = 0.9$



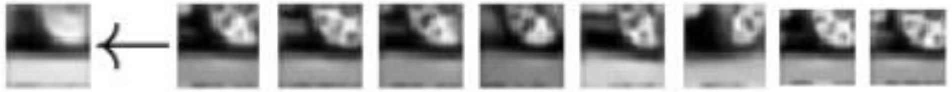
C2

Similarity(C1, C2) = 0.85
(C1 and C2 will be merged)



Similarity = 0.4

When no more clustering possible, average patches in each cluster to obtain representative patch for each cluster.



Local code-book creation: Input => Output

Car

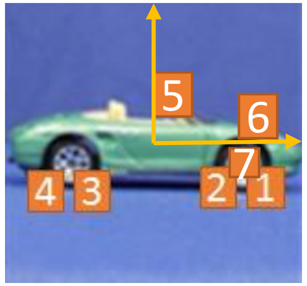
Training images +
Labels (object categories)
Object centers??

Interest Point Patches

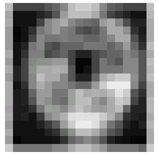
Per object class code-book

Implicit Shape Model creation from local code-book

- For all training images, match codebook entries to interest point image patches using the similarity measure.
- Activate all entries whose similarity is above threshold ‘t’.
- For each activated entry, store positions it was activated in wrt. object center.

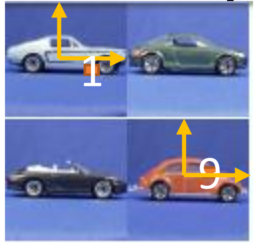


={<x1, y1>, <x2, y2>, <x3, y3>, <x4, y4>}



={<x1, y1>, <x7, y7>, …}

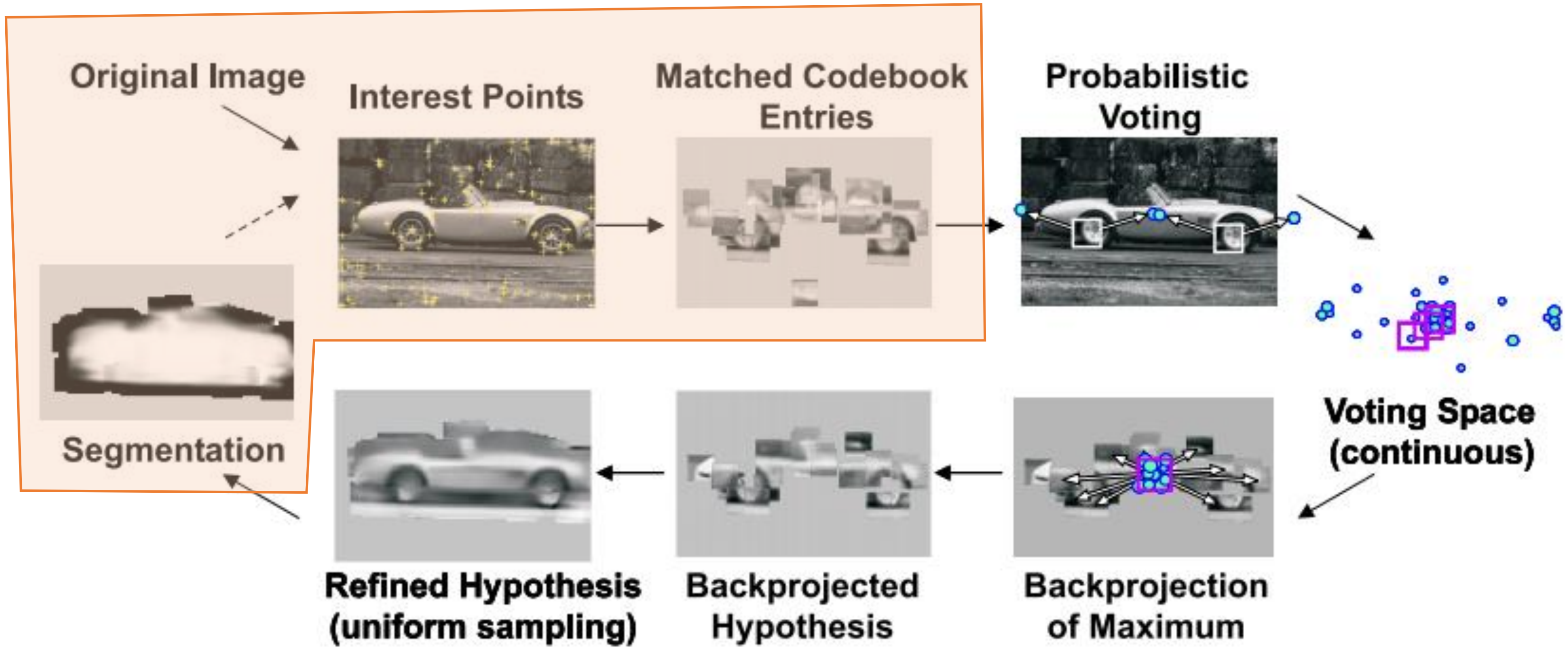
- **Output: one Implicit model per object class (encoding spatial locations of code-book entries)**



={<Car, x1, y1>,…}

={<Car, x9, y9>,…}

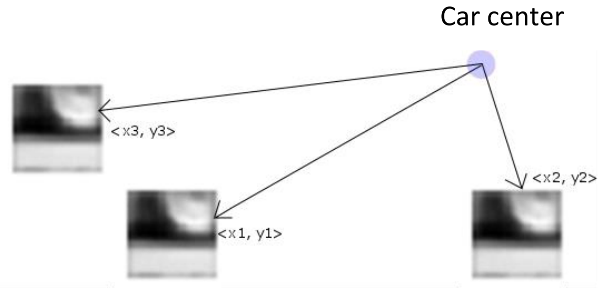
Image Recognition at a glance

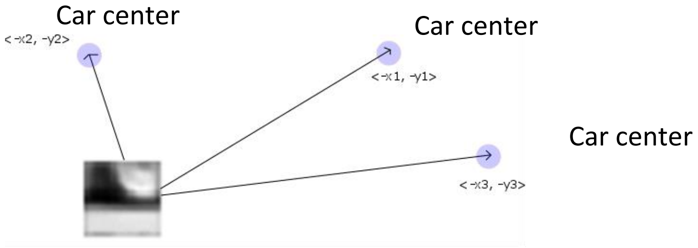


For a given image, find interest points and match against all code-book entries for all classes to find activated code-book entries. *Note – each image patch can match multiple code-book entries, even across object code-books.*

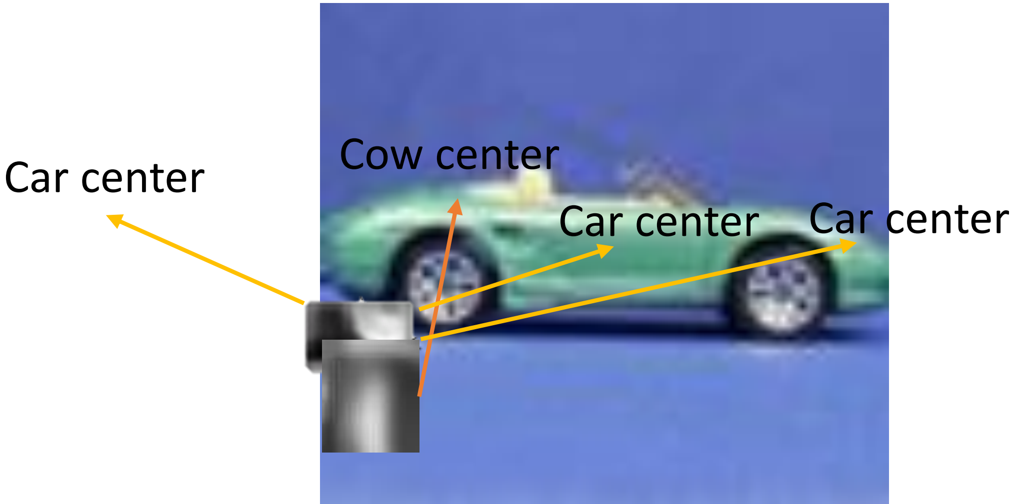
Generalized Hough Transform / patch voting

- Implicit Shape Model:  = {<Object, x1, y1>,…}. This is the same info as:

= 

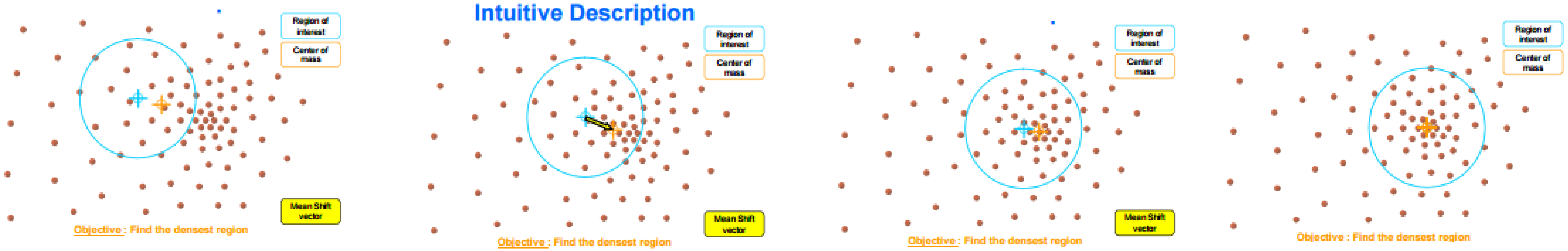
= 

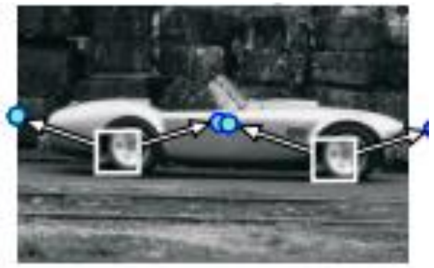
- Each matched code-book entry for each Harris detected image-patch makes votes for **objects** & their **centers**



Mean-shift Mode estimation for maxima in *continuous* voting space

- The object centers would concentrate at valid hypothesized object centers.
- Mean-shift mode estimation - non-parametric way of finding mode(s) of the probability density function from discrete samples.





Probabilistic Interpretation of the voting scheme

- Given an image patch e, at location ℓ, it activates a set $\{I_i\}$ of code-book entries. The contribution of each entry will be weighted by $p(I_i|e,\ell)$.
- An activated entry casts its vote for its object o_n at multiple positions x.

$$p(o_n, x|\mathbf{e}, \ell) = \sum_i p(o_n, x|\mathbf{e}, I_i, \ell) p(I_i|\mathbf{e}, \ell).$$

Matching is location agnostic

Once I is known, independent of e

$$p(o_n, x|\mathbf{e}, \ell) = \sum_i p(o_n, x|I_i, \ell) p(I_i|\mathbf{e}).$$

Hough vote

$$= \sum_i p(x|o_n, I_i, \ell) p(o_n|I_i, \ell) p(I_i|\mathbf{e}).$$

Quality of patch match with code-book entry

Confidence code-book entry is a foreground patch

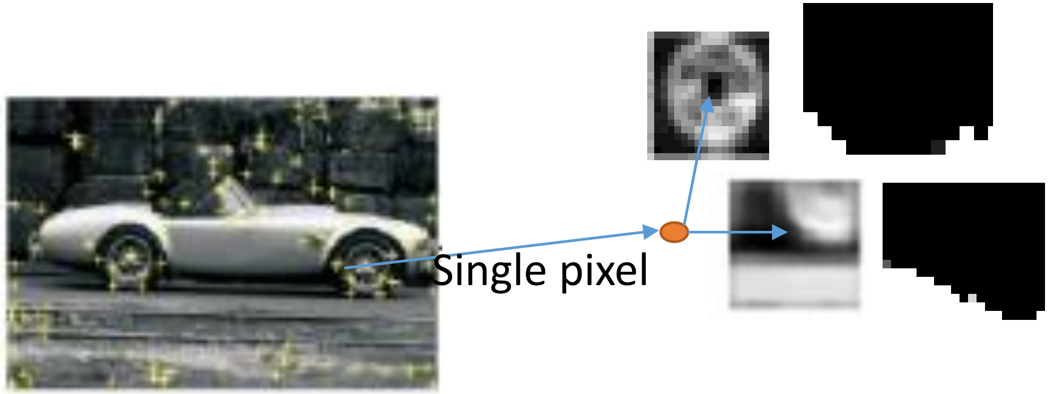
$$score(o_n, x) = \sum_k \sum_{x_j \in W(x)} p(o_n, x_j|\mathbf{e}_k, \ell_k).$$

- Mean-shift search corresponds to a Parzen window density estimate of the object center.

Category Specific object Segmentation – per pixel foreground / background estimates

- From the voting – Hypothesis - $p(o_n, x)$.
- Want $p(p = \text{figure} \mid o_n, x)$ for each pixel p, given a hypothesis – *category specific.*
- Each patch detected in training images has a p(figure) segmentation mask. Each code-book entry stores mask of matched patch as well along with its location in the object. Gives $p(p = \text{figure} \mid o_n, x, I, \ell)$

Category Specific object Segmentation – per pixel foreground / background estimates



Each pixel belongs to interest patches that are matched to multiple code-book entries that each have the segmentation mask assigned to them.

Probabilistic formulation

$$p(\mathbf{e}, \ell | o_n, x) = \frac{p(o_n, x|\mathbf{e}, \ell)p(\mathbf{e}, \ell)}{p(o_n, x)} = \frac{\sum_I p(o_n, x|I, \ell)p(I|\mathbf{e})p(\mathbf{e}, \ell)}{p(o_n, x)}$$

Influence of patch 'e' on a particular hypothesis

Bayes

Accumulate over constituent code-book entries

$$p(\mathbf{p} = \mathit{figure}|o_n, x) = \sum_{\mathbf{p}\in(\mathbf{e},\ell)} p(\mathbf{p} = \mathit{figure}|o_n, x, \mathbf{e}, \ell)p(\mathbf{e}, \ell|o_n, x)$$

Consider each image interest patch overlapping pixel 'p'

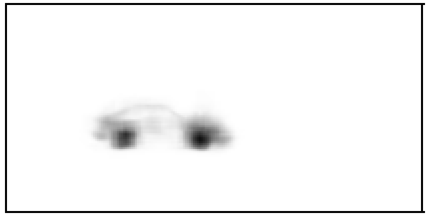
$$p(\mathbf{p} = \mathit{figure}|o_n, x) = \sum_{\mathbf{p}\in(\mathbf{e},\ell)} \sum_I p(\mathbf{p}=\mathit{fig.}|o_n, x, \mathbf{e}, I, \ell)p(\mathbf{e}, I, \ell|o_n, x)$$

Split each patch into contributions of code-book entries

$$= \sum_{\mathbf{p}\in(\mathbf{e},\ell)} \sum_I p(\mathbf{p}=\mathit{fig.}|o_n, x, I, \ell)\frac{p(o_n, x|I, \ell)p(I|\mathbf{e})p(\mathbf{e}, \ell)}{p(o_n, x)}$$

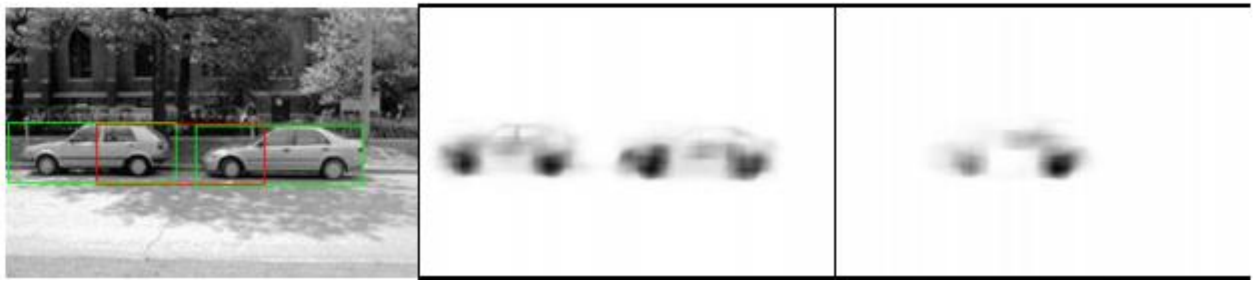
Resolve into code-book entries (like last time)

$$L = \frac{p(\mathbf{p} = \mathit{figure}|o_n, x)}{p(\mathbf{p} = \mathit{ground}|o_n, x)}.$$

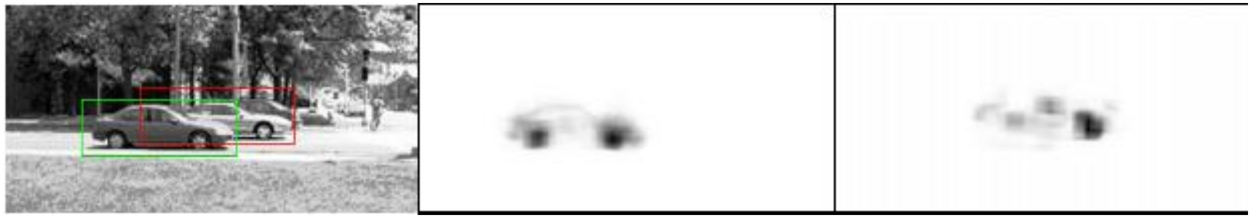


Reducing false positives / Handling multiple objects

- Matching is done via local patches, with global structure enforced through voting.
- Large number of false positives due to secondary hypothesis



- Bounding boxes based rejection (if two hypothesis bounding boxes intersect, keep the one which is stronger)
- solves above case. Bounding boxes may actually intersect due to occlusion.



- **So, when do we combine report the weaker hypothesis as well, and when not?**

Hypothesis set selection based on Min Descriptor Length

- Describe image: We can explain away a pixel as belonging to an object or we have to encode its grayscale value. We "save" on description length if we explain away S_{area} pixels due to an object. However, we subtract model complexity – prefer low number of objects, and penalize explaining away a pixel as object when segmentation says it is background.

$$S_h = K_0 S_{area} - K_1 S_{model} - K_2 S_{error}$$

$$\sum_{\mathbf{p} \in Seg(h)} (1 - p(\mathbf{p} = \mathit{figure} | h))$$

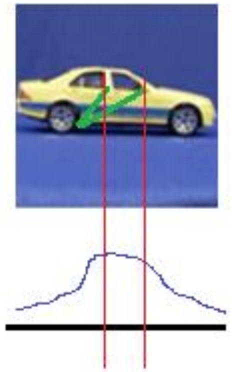
- For overlapping hypotheses h_1 and h_2 consider the "savings" made by combined hypotheses. Select overlapping hypothesis if +ve.

$$S_{h_1 \cup h_2} = S_{h_1} + S_{h_2} - S_{area}(h_1 \cap h_2) + S_{error}(h_1 \cap h_2)$$

Potential discussion topics

- Articulated objects – interpolation across different types of objects
  - Votes made by patches from different training images and continuous voting space. Disadvantages?

Training



Testing

- Invariances in the matching?
- Non-rigid articulations / soft body deformations.
- Statistical issues in the use of per frame snapshots of videos for analysis.
- Category independent code-books – bag of visual words type?