Lecture 14: Detecting Objects by Parts

Deformable parts model

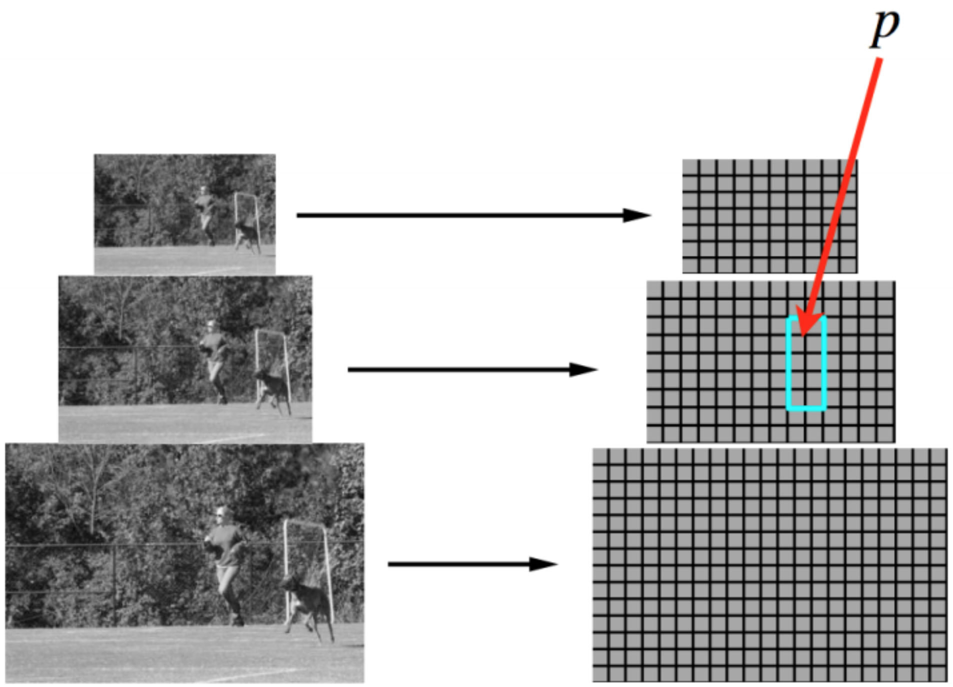
Juan Carlos Niebles and Jiajun Wu

CS131 Computer Vision: Foundations and Applications

What will we learn today?

- Deformable parts model
  - Overview
  - Method
  - Pipeline
  - Results and analysis

Recap: Dalal-Triggs Detector



HOG pyramid H

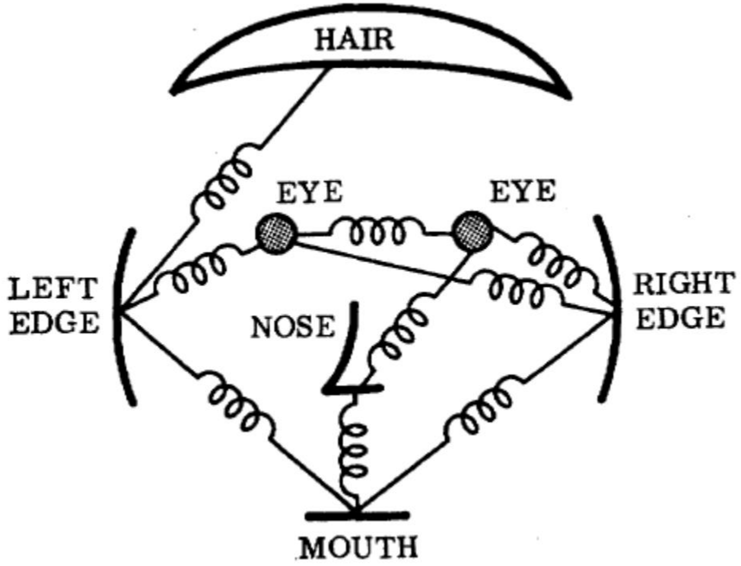
Filter F

Object Filter/Template:

- HOG features.
- Global for the entire object: no explicit information about the "parts" that make up the object.
- Rigid: no explicit handling of object deformation/change of pose.

Deformable parts model

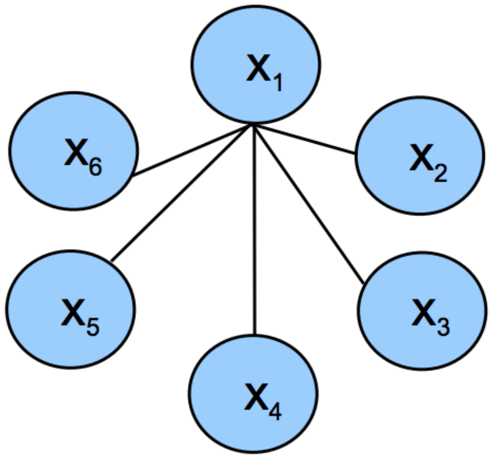
- Represents an object as a collection of parts arranged in a deformable configuration
- Each part represents local appearances
- Spring-like connections between certain pairs of parts

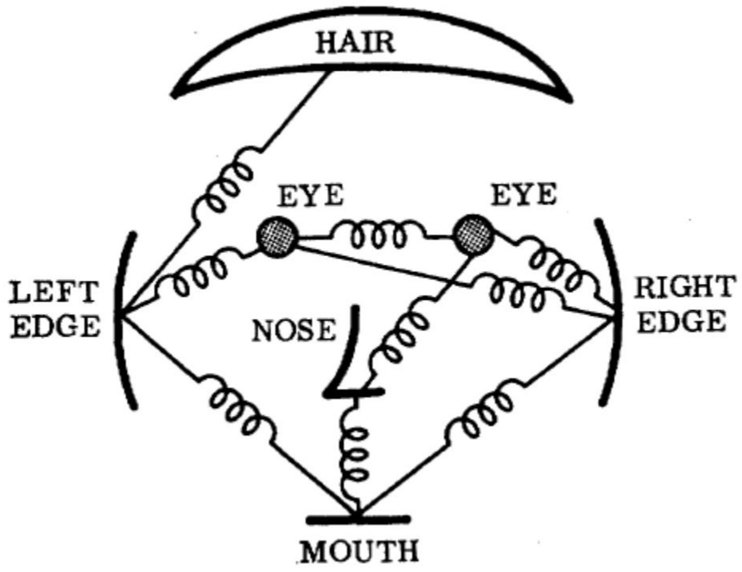


Fischler and Elschlager,
Pictoral Structures,
1973

Deformable parts model

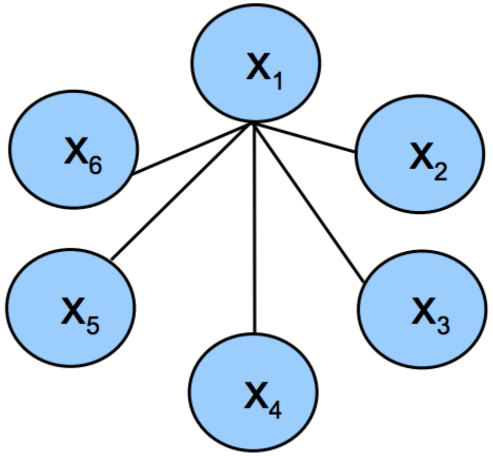
- The parts of an object form pairwise relationships.
- We can model this using a "star model"
  - where every part is defined relative to a root.





Detecting a person with their parts

- For example, a person can be modelled as having a head, left arm, right arm, etc.
- All parts can be modelled relative to the global person detector, which acts as the root.

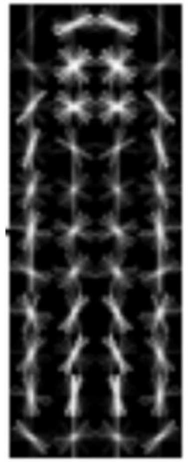


What will we learn today?

- Deformable parts model
  - Overview
  - Method
  - Pipeline
  - Results and analysis

Deformable parts model

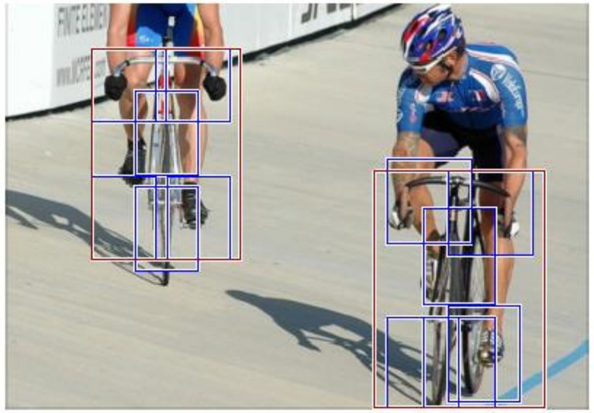
- Each model will have a global filter and a set of part filters.
- Part filters will generally be specified at higher resolution than the global filter. This helps capturing more detail.
- Here is an example of a global person filter with its 'head' part filter:

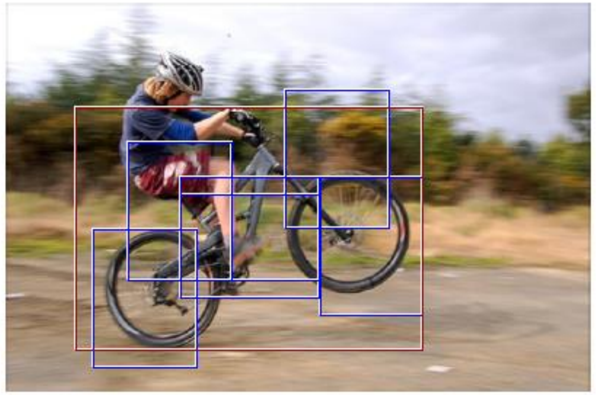


Global/root filter

Part filter

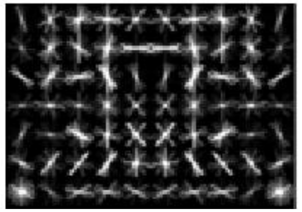
Two-component bicycle model



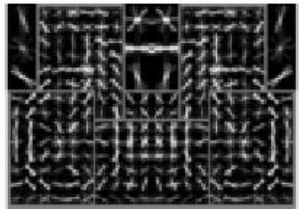


"side view" bike
model component

Root filter



Part filters

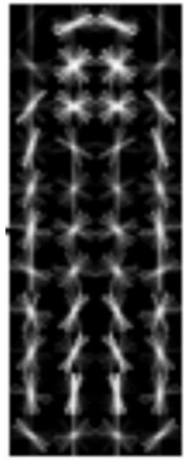


Deformable parts model

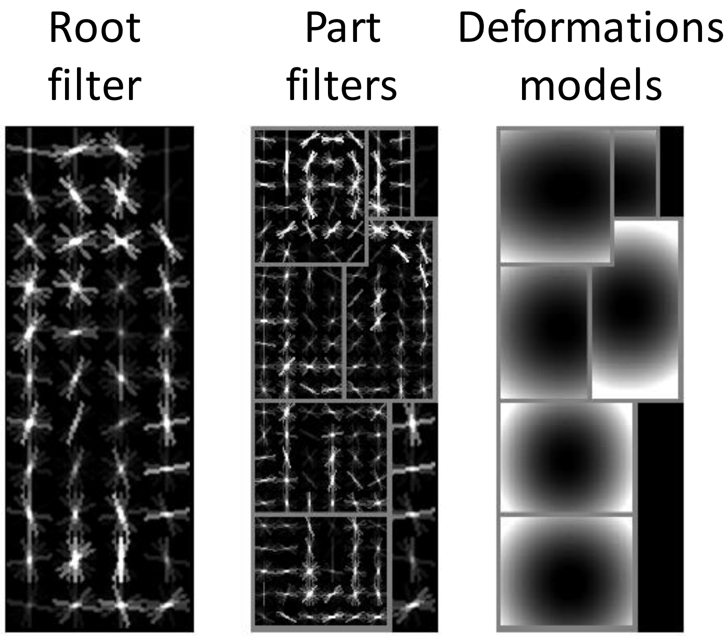
- Mixture of deformable part models (one component for each 'view-point' that we want to model)

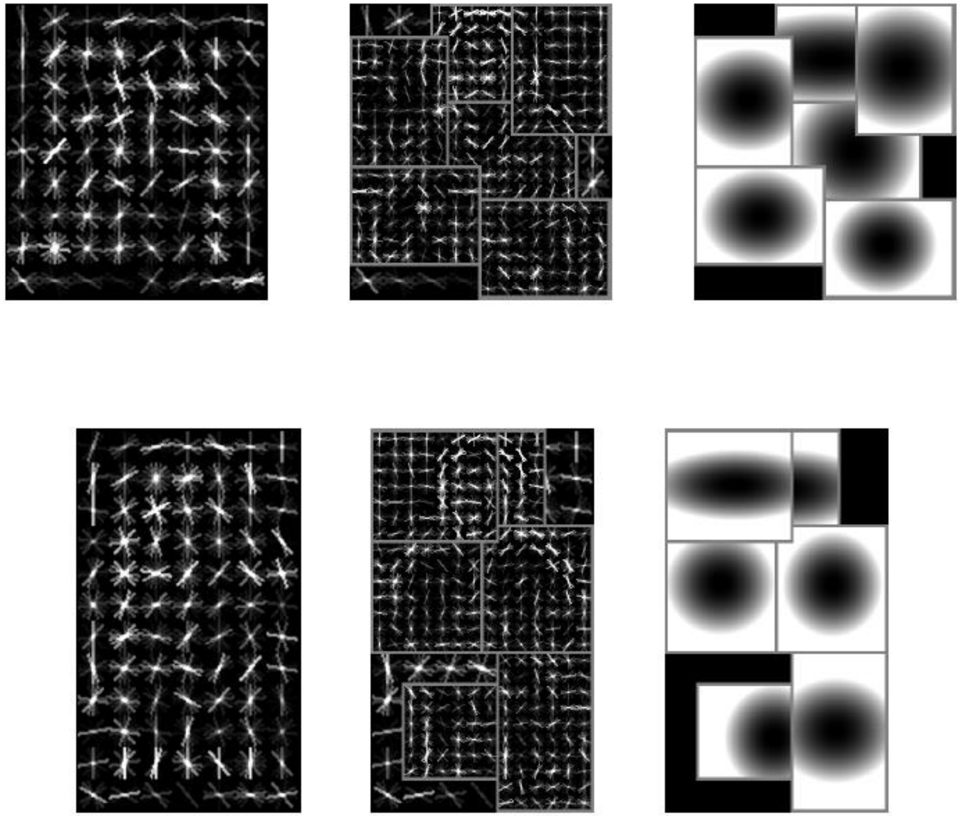
- Each component has global filter + deformable part filters

- Part filters have finer details



Deformable parts person model

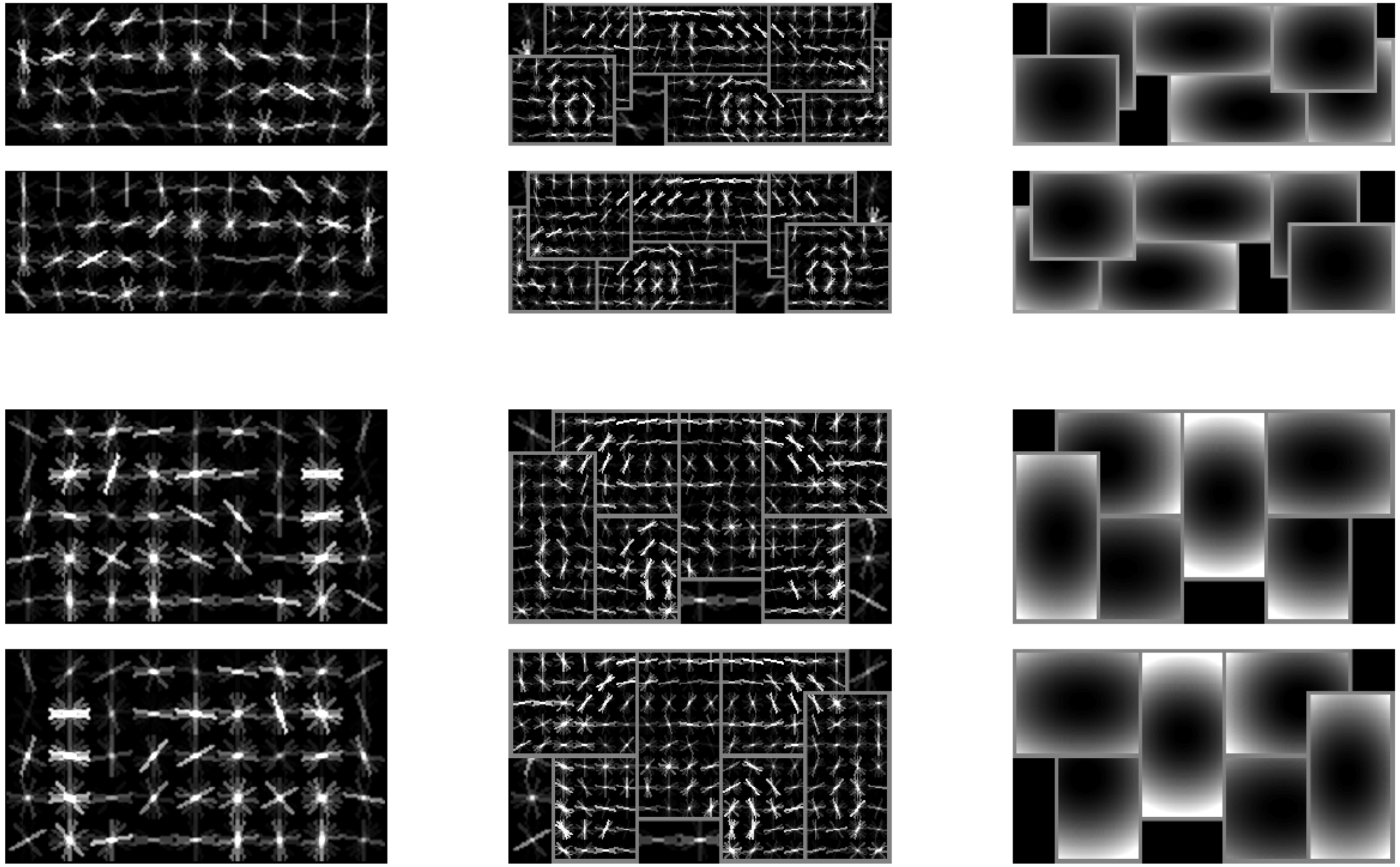




Deformable parts car model

side view

frontal view

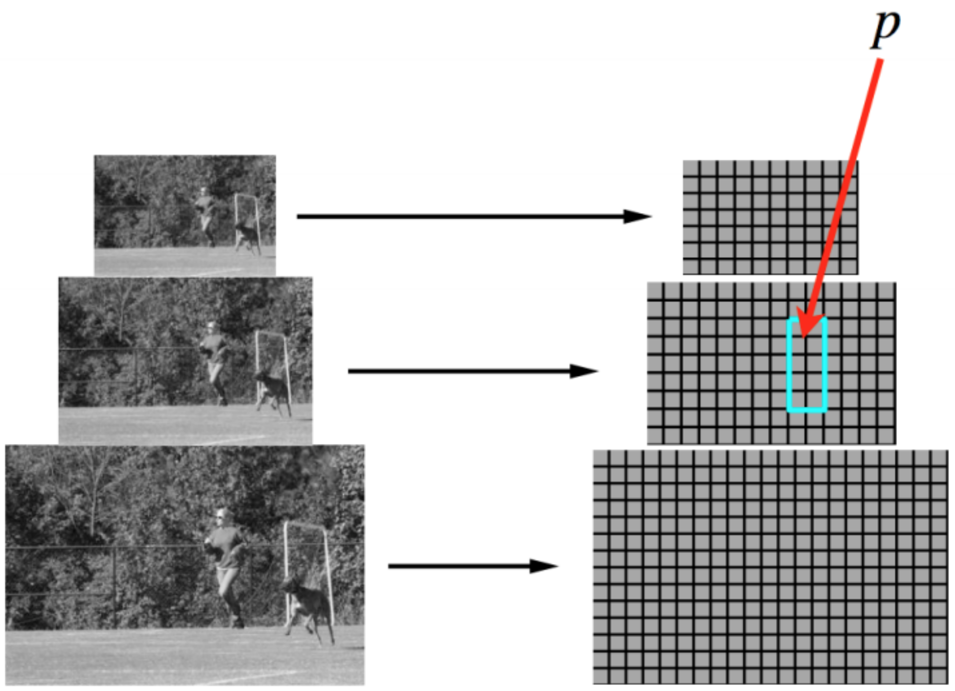


root filters (coarse)

part filters (fine)

deformation models

Remember from Dalal and Triggs



HOG pyramid H

Filter F 

Score of F at position p is
$F \cdot \phi(p, H)$

$\phi(p, H)$ = concatenation of
HOG features from
subwindow specified by p

Deformable parts model

- A model for an object with n parts is a $(n+2)$ tuple:

$$(F_0, P_1, \cdots, P_n, b)$$

Root filter Model for 1st part Bias term

- Each part-based model defined as:

$$(F_i, v_i, d_i)$$

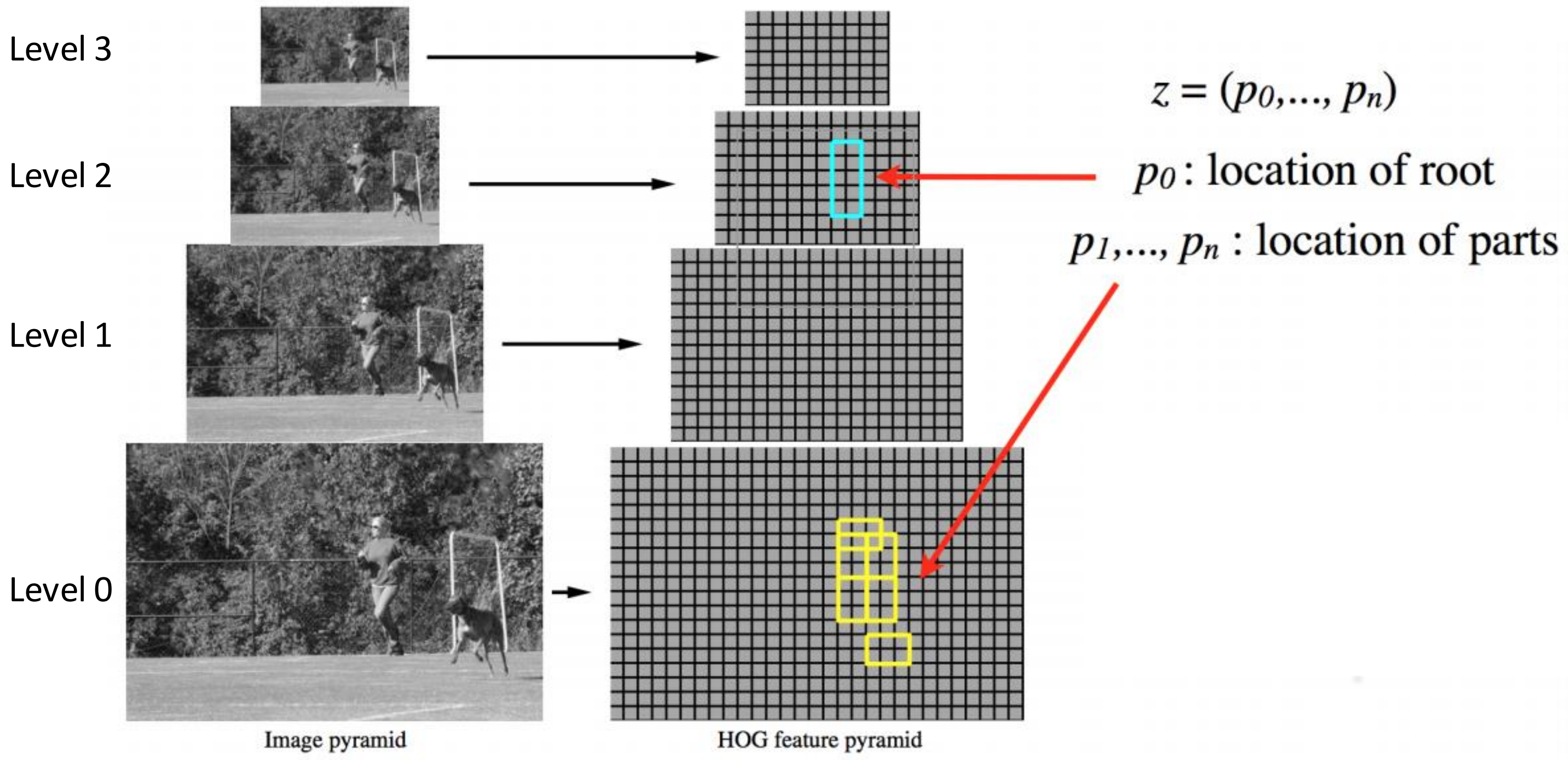
F_i filter for the *i*-th part

v_i “anchor” position for part *i* relative to the root position

d_i defines a deformation cost for each possible placement of the part relative to the anchor position

Specifying the location of a detection

$p_i = (x_i, y_i, l_i)$ specifies the level and position of the *i*-th filter



Calculating the score for a detection

- The score for a detection is defined as the sum of scores for the global and part detectors minus the sum of deformation costs for each part.

$$\text{detection score} = \sum_{i=0}^{n} F_i \, \phi(p_i, H) - \sum_{i=1}^{n} d_i\left(\Delta x_i, \Delta y_i, \Delta x_i^2, \Delta y_i^2\right)$$

- This means that if a detection's parts are really far away from where they should be, it's probably a false positive.

Calculating the score for a detection

$$\text{detection score} = \sum_{i=0}^{n} F_i\, \phi(p_i, H) - \sum_{i=1}^{n} d_i\left(\Delta x_i, \Delta y_i, \Delta x_i^2, \Delta y_i^2\right)$$

- Scores for each part filter + global filter (appearance information only).

Calculating the score for a detection

$$\text{detection score} = \sum_{i=0}^{n} F_i\, \phi(p_i, H) - \sum_{i=1}^{n} d_i\left(\Delta x_i, \Delta y_i, \Delta x_i^2, \Delta y_i^2\right)$$

- Scores for each part filter + global filter (appearance information only).
- The deformation costs for each part (captures part location information).
  - Δx_i measures the distance in the x-direction from where part i should be.
  - Δy_i measures the same in the y-axis direction.
  - d_i is the weight associated for part i that penalizes the part for being away.

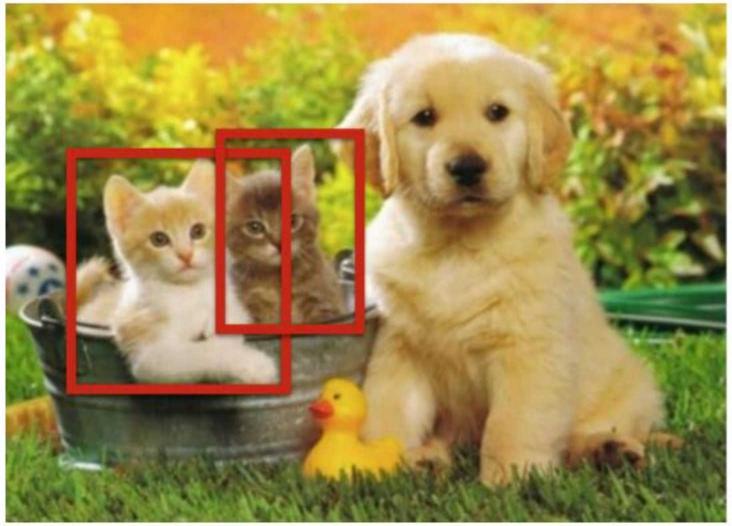
What will we learn today?

- Deformable parts model
  - Overview
  - Method
  - Pipeline
  - Results and analysis

Detection pipeline

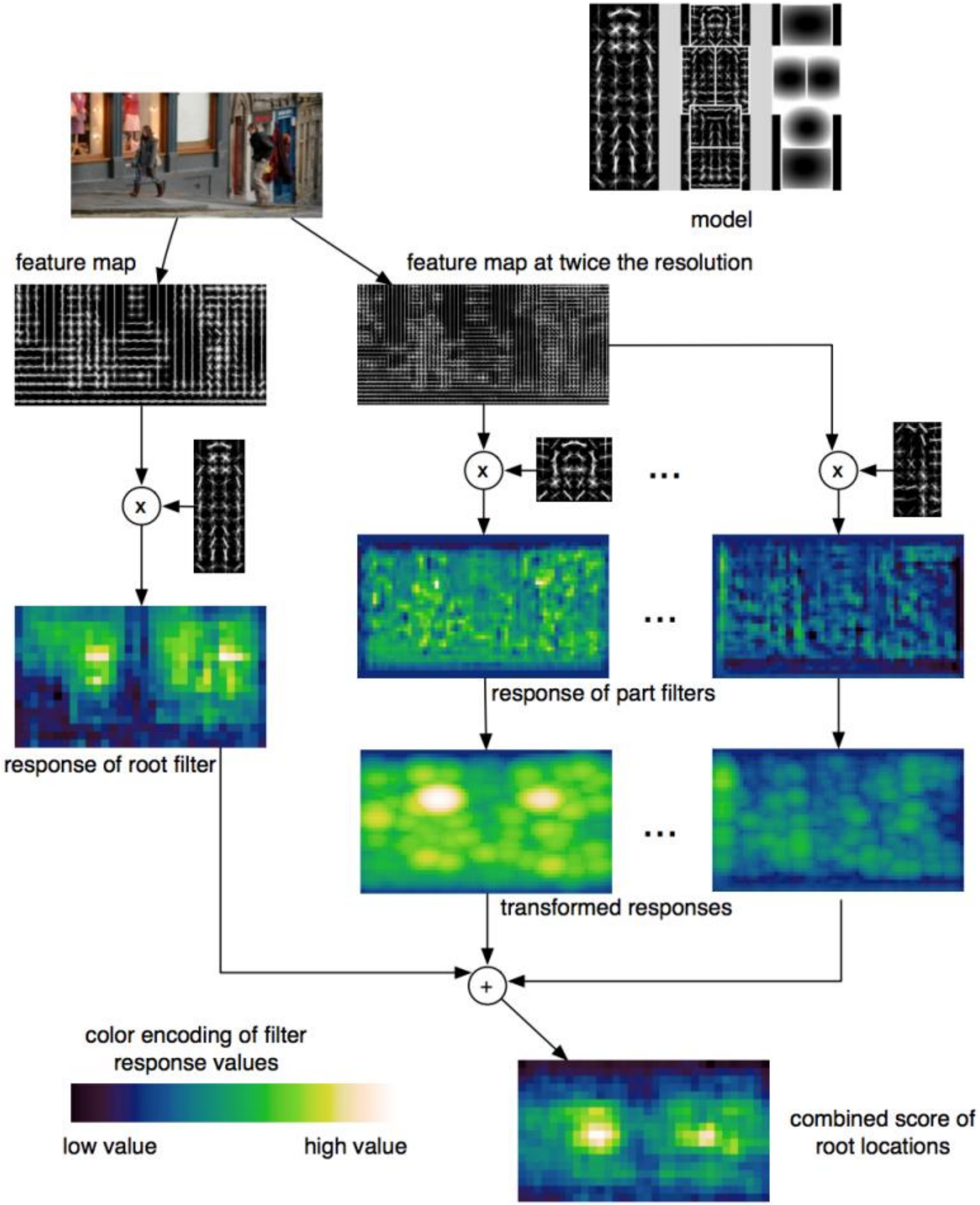
- So, to make a detection, we use the sliding window technique and with the global and part filters.

- To score a detection, we accumulate the global and part scores and penalize the deformation of the parts.



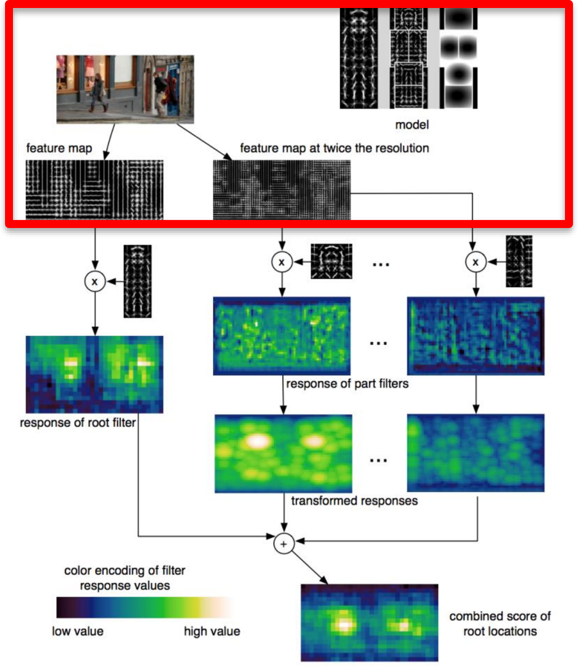
Overall detection pipeline

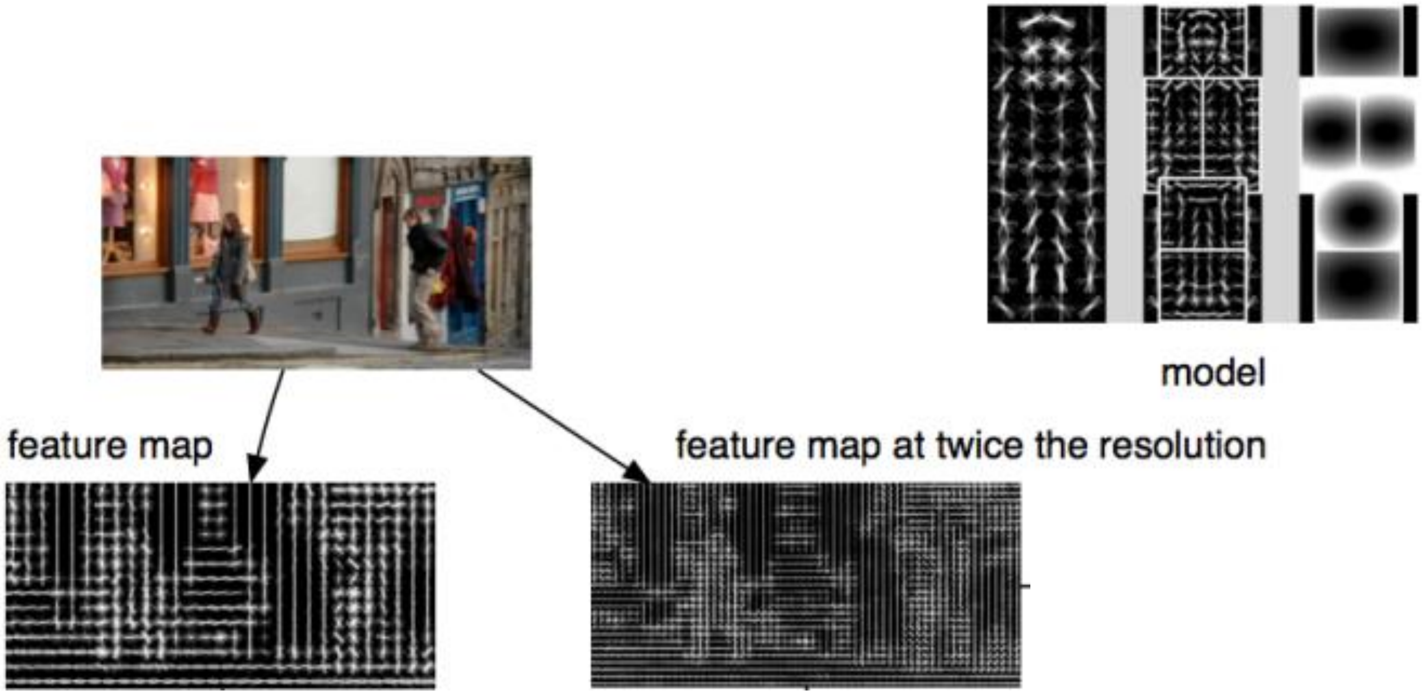
Let's break this down



Detection pipeline

1. Make sure you have filters for the global and the parts: F_i
2. Compute HOG feature maps from the input image

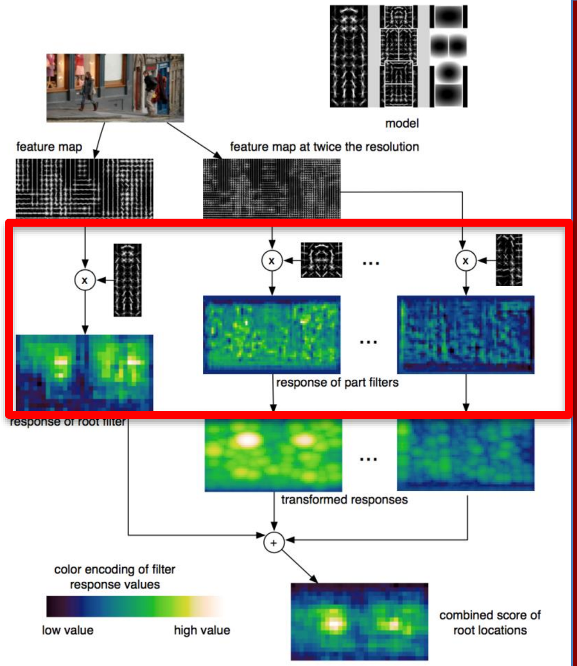


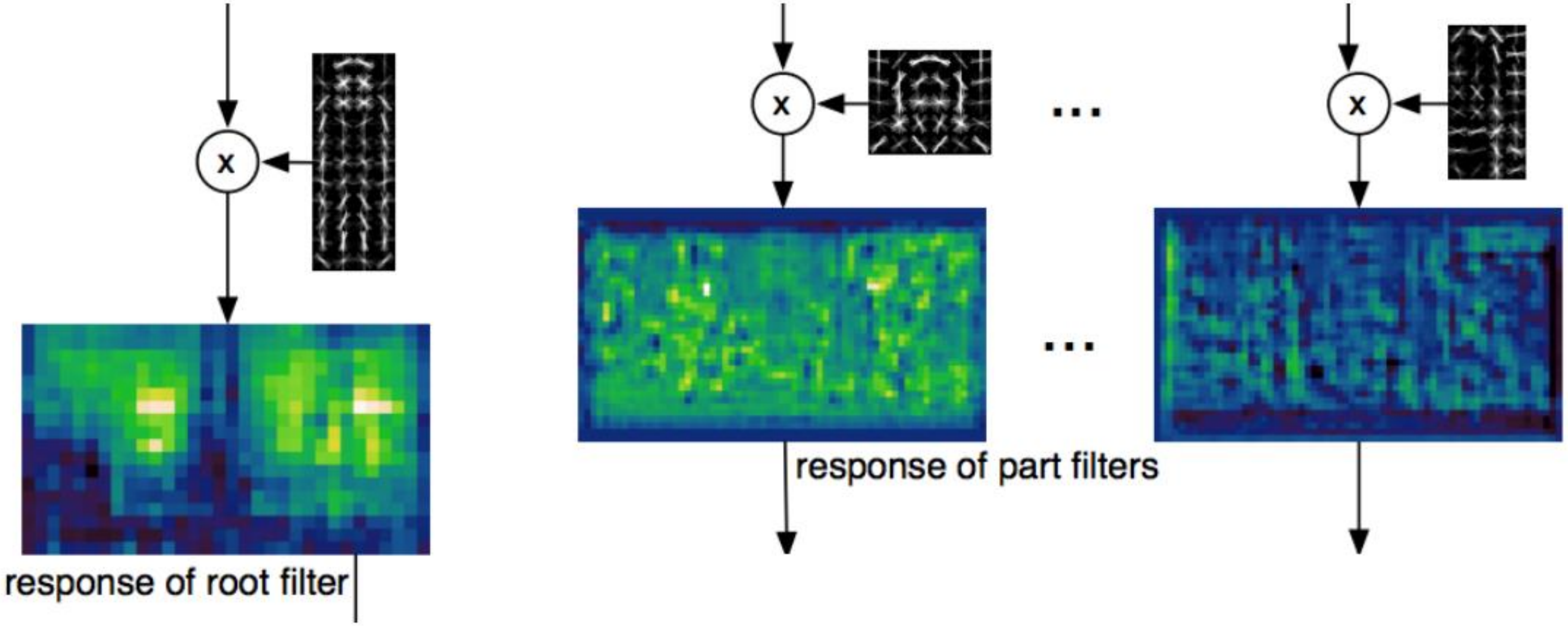


Detection pipeline

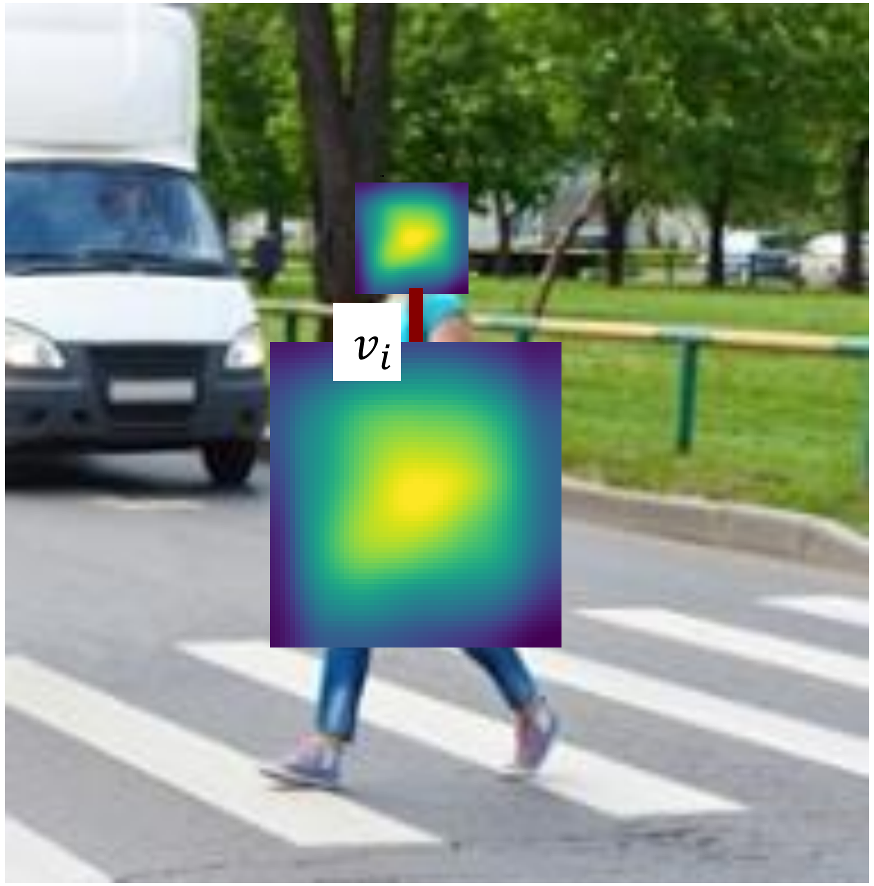
Apply the filters:

$$F_i\, \phi(p_i, H)\ ,\ i = 1, \dots, n$$





Accounting for Spatial cost with a Transformation

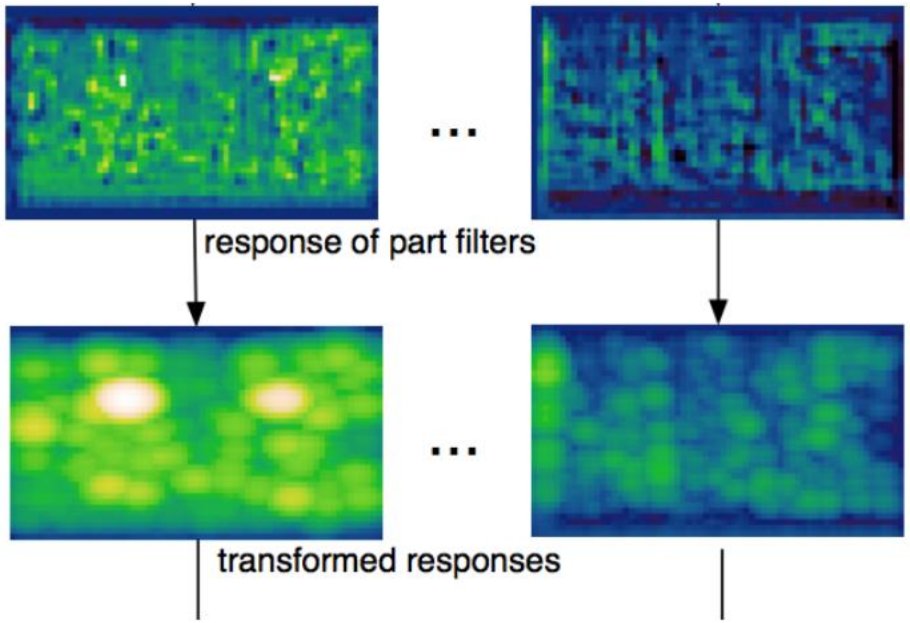


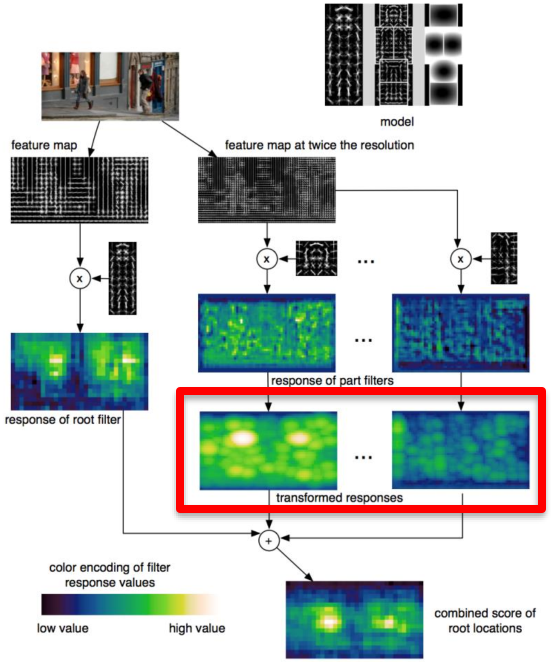
- Given the location for the detected head, we can guess where the body (root filter) should be.
- The body should be in the direction calculated from the root person filter: v_i
- But we allow for some deformation or spatial shift on the location of the head with respect to the body: d_i
- We should 'spread' the head detection when calculating potential locations of the root!

Detection pipeline

Now apply the spatial costs for each part:

$$detection\ score = Fi\ \phi(p_i, H) - d_i(\Delta x_i, \Delta y_i, \Delta x_i^2, \Delta y_i^2)$$



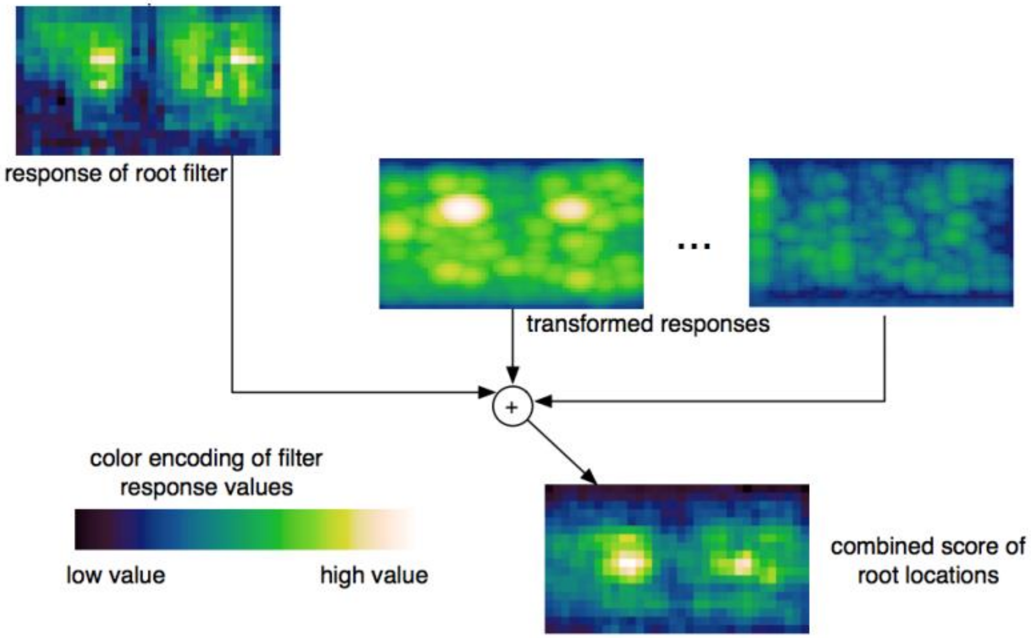


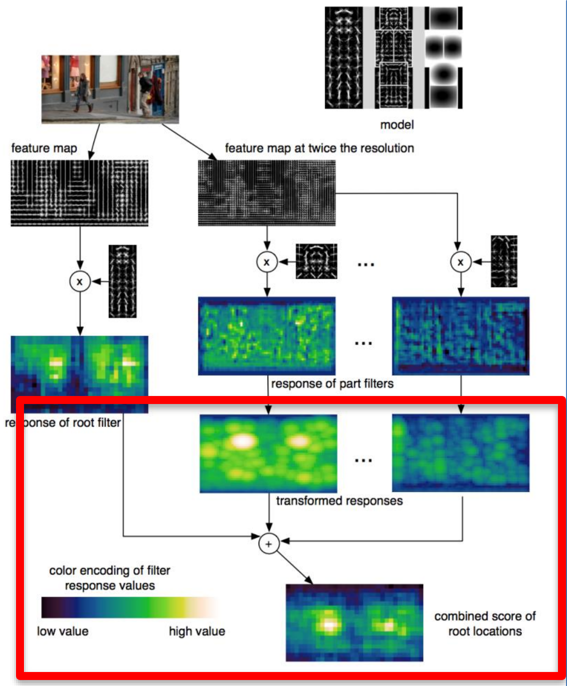
Detection pipeline

Now add the global filter:

$$detection\ score$$

$$= F_0 \phi(p_i, H) + \sum_{i=1}^{n} F_i\, \phi(p_i, H) - \sum_{i=1}^{n} d_i\left(\Delta x_i, \Delta y_i, \Delta x_i^2, \Delta y_i^2\right)$$

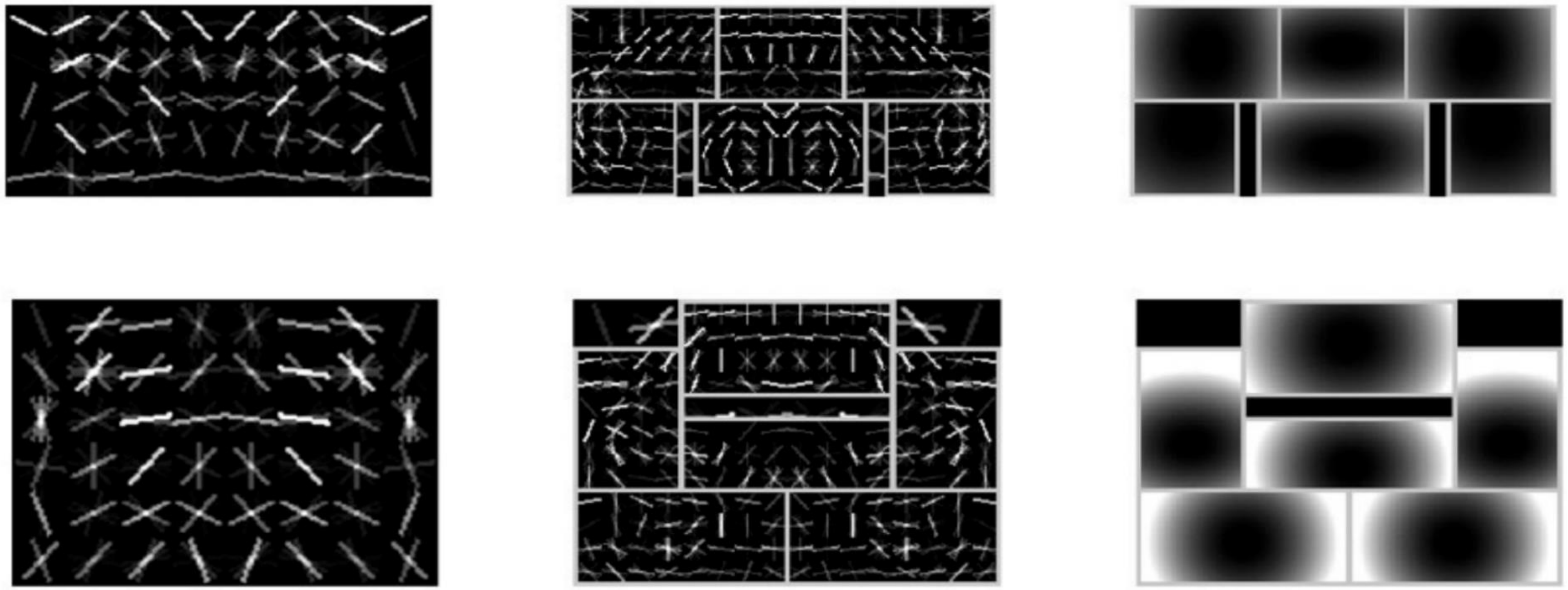




What will we learn today?

- Deformable parts model
  - Overview
  - Method
  - Pipeline
  - Results and analysis

Deformable Parts Model (DPM) - bicycle

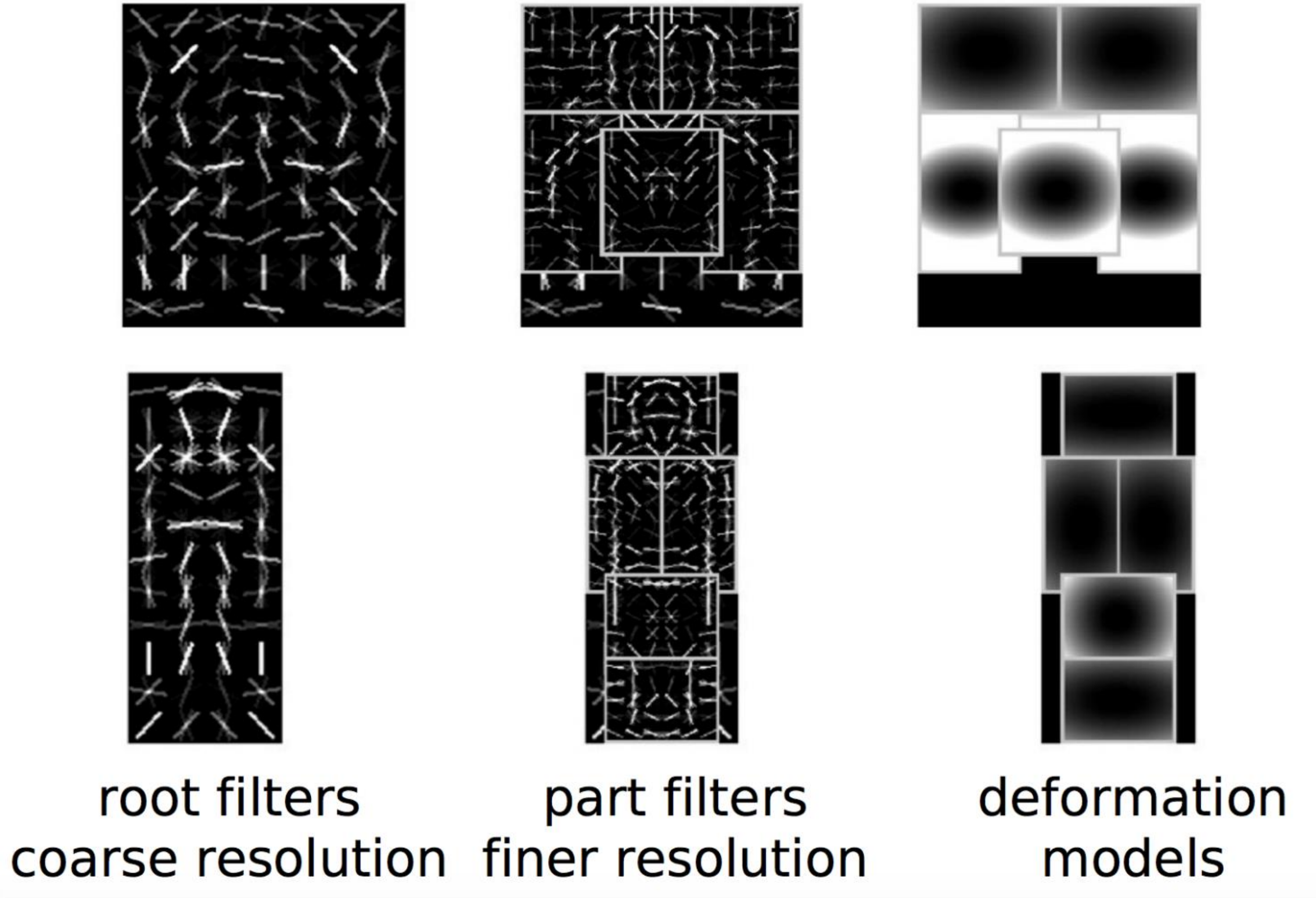


root filters
coarse resolution

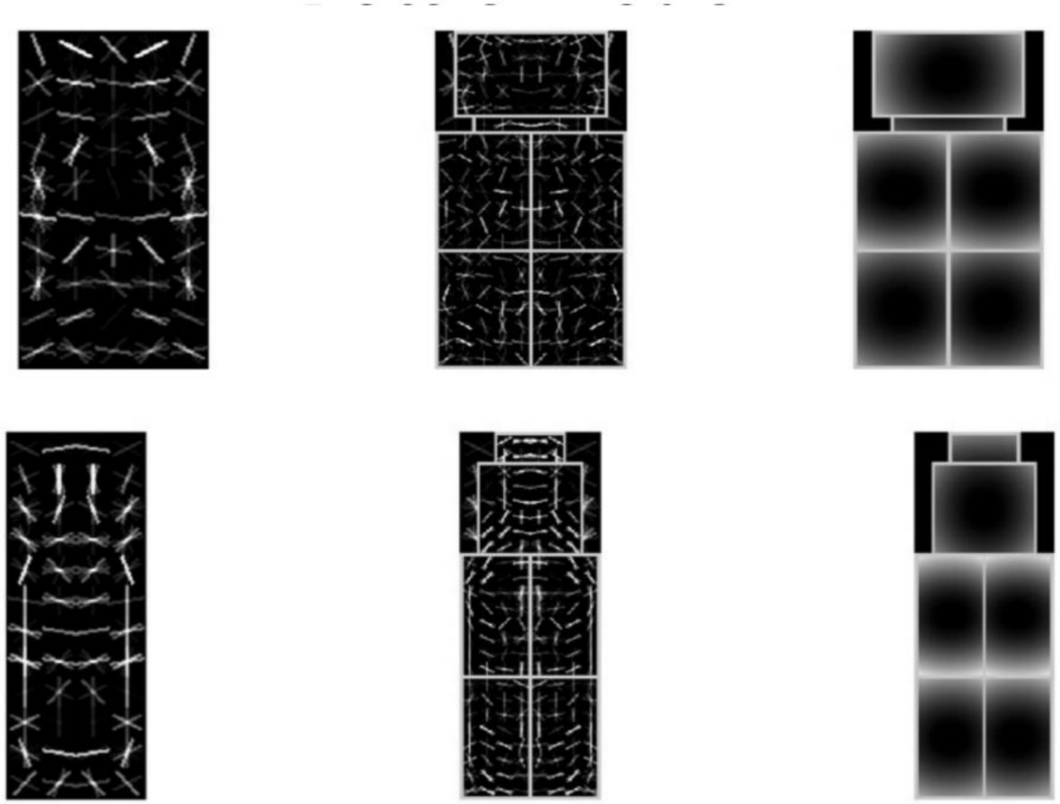
part filters
finer resolution

deformation
models

DPM - person


root filters
coarse resolution
part filters
finer resolution
deformation
models

DPM - bottle



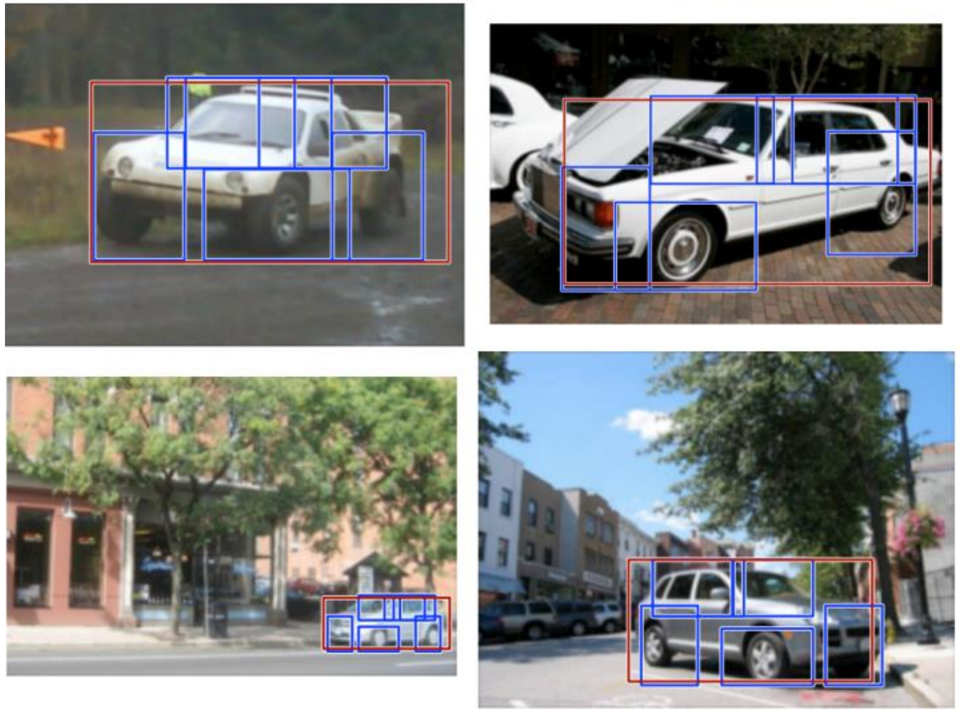
root filters
coarse resolution

part filters
finer resolution

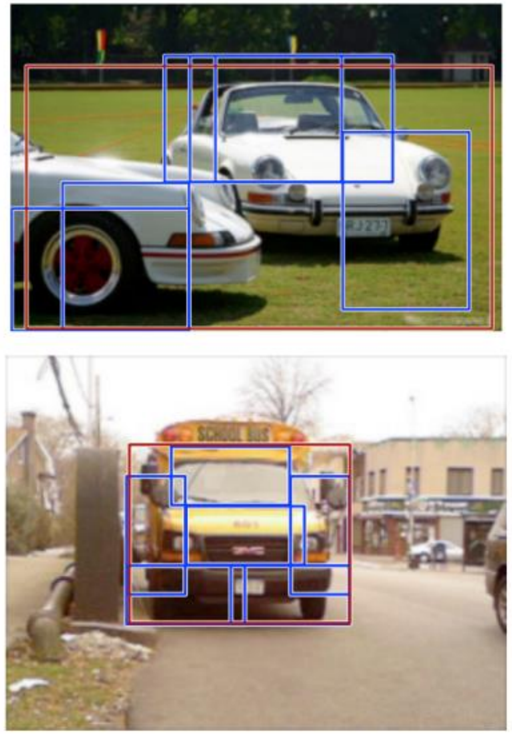
deformation
models

Results – car detection

high scoring true positives

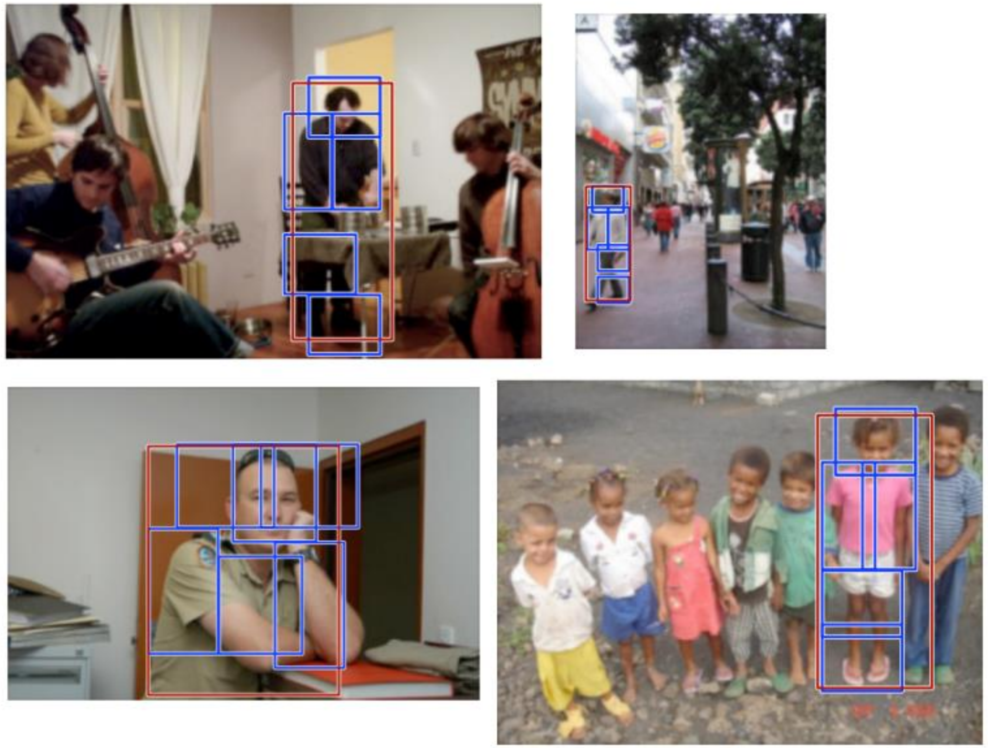


high scoring false positives

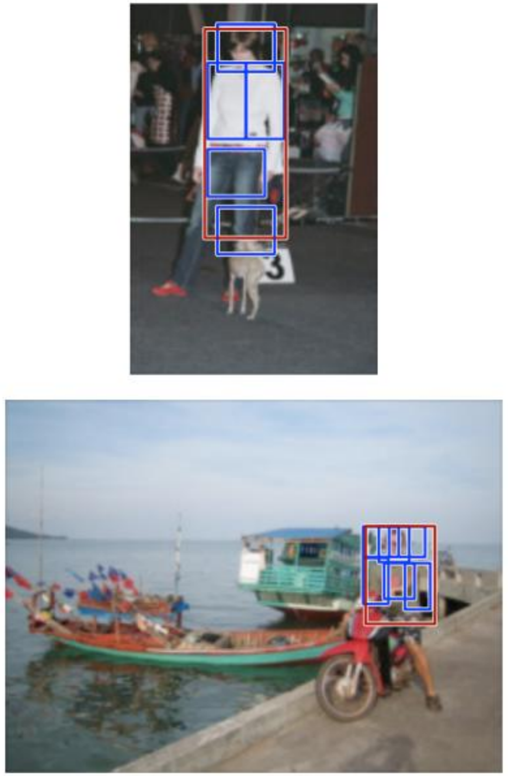


Results – Person detection

high scoring true positives



high scoring false positives
(not enough overlap)



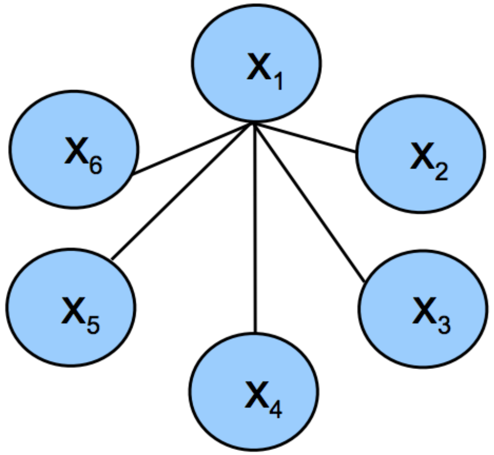
Results – horse detection

DPM - discussion

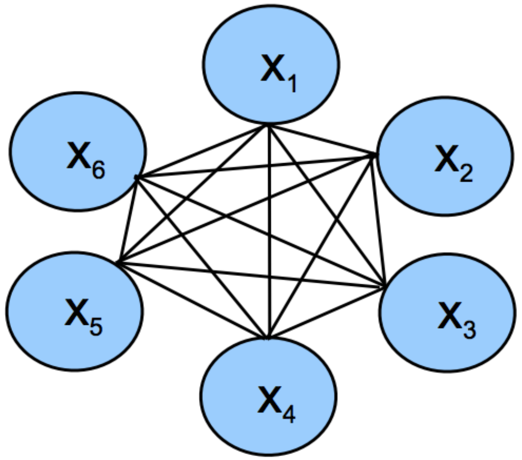
- Approach
  - Manually selected set of parts - Specific detector trained for each part
  - Spatial model trained on part activations
  - Evaluate joint likelihood of part activations
- Advantages
  - Parts have intuitive meaning.
  - Standard detection approaches can be used for each part.
  - Works well for specific categories.
- Disadvantages
  - Parts need to be selected manually
  - Semantically motivated parts sometimes don't have a simple appearance distribution
  - No guarantee that some important part hasn't been missed
- When switching to another category, the model has to be rebuilt from scratch.

Extensions - From star shaped model to constellation model

“Star” shape model



Fully connected shape model



Summary

- Deformable parts model
  - Overview
  - Method
  - Pipeline
  - Results and analysis