Generic object detection with deformable part-based models

Many slides based on P. Felzenszwalb
Challenge: Generic object detection
Histograms of oriented gradients (HOG)

- Partition image into blocks at multiple scales and compute histogram of gradient orientations in each block

N. Dalal and B. Triggs, Histograms of Oriented Gradients for Human Detection, CVPR 2005

Image credit: N. Snavely
Histograms of oriented gradients (HOG)

- Partition image into blocks at multiple scales and compute histogram of gradient orientations in each block

Pedestrian detection with HOG

- Train a pedestrian template using a linear support vector machine

positive training examples

![Positive Training Examples](image1.png)

negative training examples

![Negative Training Examples](image2.png)

Pedestrian detection with HOG

• Train a pedestrian template using a linear support vector machine
• At test time, convolve feature map with template
• Find local maxima of response
• For multi-scale detection, repeat over multiple levels of a HOG pyramid

N. Dalal and B. Triggs, Histograms of Oriented Gradients for Human Detection, CVPR 2005
Example detections

[Dalal and Triggs, CVPR 2005]
Are we done?

• Single rigid template usually not enough to represent a category
  • Many objects (e.g. humans) are articulated, or have parts that can vary in configuration

• Many object categories look very different from different viewpoints, or from instance to instance
Discriminative part-based models

P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan, Object Detection with Discriminatively Trained Part Based Models, PAMI 32(9), 2010
Discriminative part-based models

Multiple components

Discriminative part-based models

Object hypothesis

- Multiscale model: the resolution of part filters is twice the resolution of the root

\[ z = (p_0, \ldots, p_n) \]

- \( p_0 \): location of root
- \( p_1, \ldots, p_n \): location of parts

Score is sum of filter scores minus deformation costs
Scoring an object hypothesis

- The score of a hypothesis is the sum of filter scores minus the sum of deformation costs

\[ \text{score}(p_0, \ldots, p_n) = \sum_{i=0}^{n} F_i \cdot H(p_i) - \sum_{i=1}^{n} D_i \cdot (dx_i, dy_i, dx_i^2, dy_i^2) \]
Scoring an object hypothesis

- The score of a hypothesis is the sum of filter scores minus the sum of deformation costs

\[
\text{score}(\mathbf{p}_0, \ldots, \mathbf{p}_n) = \sum_{i=0}^{n} F_i \cdot H(p_i) - \sum_{i=1}^{n} D_i \cdot (dx_i, dy_i, dx_i^2, dy_i^2)
\]
Detection

- Define the score of each root filter location as the score given the best part placements:

\[
\text{score}(p_0) = \max_{p_1, \ldots, p_n} \text{score}(p_0, \ldots, p_n)
\]
Detection

• Define the score of each root filter location as the score given the best part placements:

\[
\text{score}(p_0) = \max_{p_1, \ldots, p_n} \text{score}(p_0, \ldots, p_n)
\]

• Efficient computation: *generalized distance transforms*

• For each “default” part location, find the score of the “best” displacement

\[
R_i(x, y) = \max_{dx, dy} \left( F_i \cdot H(x + dx, y + dy) - D_i \cdot (dx, dy, dx^2, dy^2) \right)
\]

Head filter

Deformation cost
Detection

• Define the score of each root filter location as the score given the best part placements:

\[
score(p_0) = \max_{p_1, \ldots, p_n} score(p_0, \ldots, p_n)
\]

• Efficient computation: generalized distance transforms

• For each “default” part location, find the score of the “best” displacement

\[
R_i(x, y) = \max_{dx, dy} \left( F_i \cdot H(x + dx, y + dy) - D_i \cdot (dx, dy, dx^2, dy^2) \right)
\]
Detection

feature map

feature map at twice the resolution

model

response of root filter

response of part filters

transformed responses

color encoding of filter response values

combined score of root locations
Detection result
Training

- Training data consists of images with labeled bounding boxes
- Need to learn the filters and deformation parameters
Training

• Our classifier has the form

\[ f(x) = \max_z w \cdot H(x, z) \]

• \( w \) are model parameters, \( z \) are latent hypotheses

• **Latent SVM** training:
  • Initialize \( w \) and iterate:
    • Fix \( w \) and find the best \( z \) for each training example (detection)
    • Fix \( z \) and solve for \( w \) (standard SVM training)

• Issue: too many negative examples
  • Do “data mining” to find “hard” negatives
Car model

Component 1

Component 2
Car detections

high scoring true positives

high scoring false positives
Person model
Person detections

high scoring true positives

high scoring false positives
(not enough overlap)
Cat model
Cat detections

high scoring true positives

high scoring false positives (not enough overlap)
Bottle model
More detections

horse

sofa

bottle
Quantitative results (PASCAL 2008)

- 7 systems competed in the 2008 challenge
- Out of 20 classes, first place in 7 classes and second place in 8 classes
R-CNN: Regions with CNN features

Object detection system overview. Our system (1) takes an input image, (2) extracts around 2000 bottom-up region proposals, (3) computes features for each proposal using a large convolutional neural network (CNN), and then (4) classifies each region using class-specific linear SVMs. R-CNN achieves a mean average precision (mAP) of 53.7% on PASCAL VOC 2010. For comparison, Uijlings et al. (2013) report 35.1% mAP using the same region proposals, but with a spatial pyramid and bag-of-visual-words approach. The popular deformable part models perform at 33.4%.