CS4670 / 5670: Computer Vision

Noah Snavely

Lecture 31: Modern recognition

Visual Object Classes Challenge 2009 (VOC2009)

PASCAL2

Pattern Analysis, Statistical Modelling and Computational Learning

[click on an image to see the annotation]

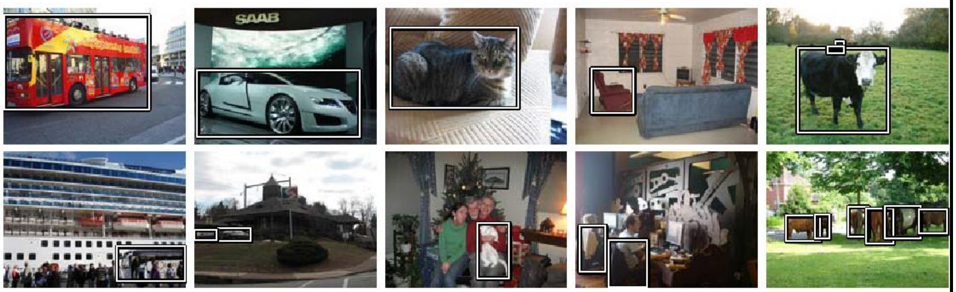
Object detection: where are we?

Credit: Flickr user [neilalderney123]

- Incredible progress in the last ten years
- Better features, better models, better learning methods, better datasets
- Combination of science and hacks

Vision Contests

- PASCAL VOC Challenge

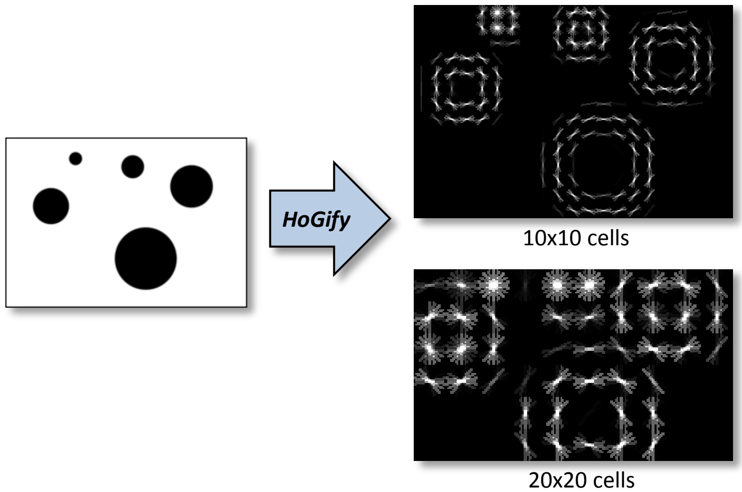


- 20 categories
- Annual classification, detection, segmentation, … challenges

Machine learning for object detection

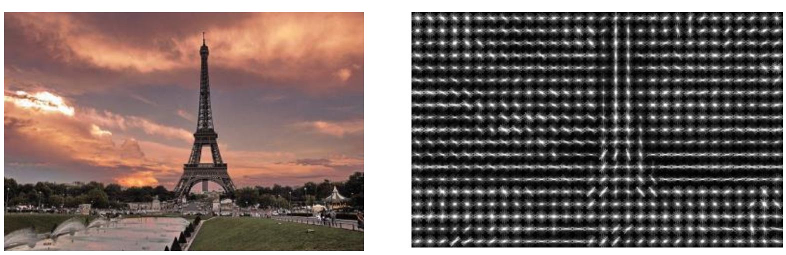
- What features do we use?
  - intensity, color, gradient information, …
- Which machine learning methods?
  - generative vs. discriminative
  - k-nearest neighbors, boosting, SVMs, …
- What hacks do we need to get things working?

Histogram of Oriented Gradients (HoG)

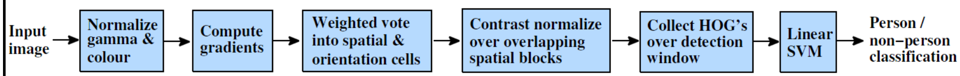


[Dalal and Triggs, CVPR 2005]

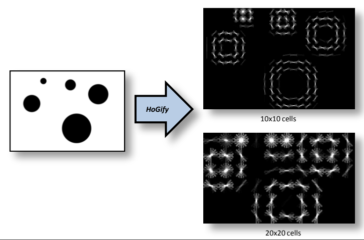
Histogram of Oriented Gradients (HoG)



Histogram of Oriented Gradients (HoG)



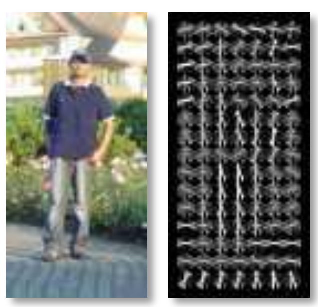
- Like SIFT (Scale Invariant Feature Transform), but…
  - Sampled on a dense, regular grid
  - Gradients are contrast normalized in overlapping blocks



[Dalal and Triggs, CVPR 2005]

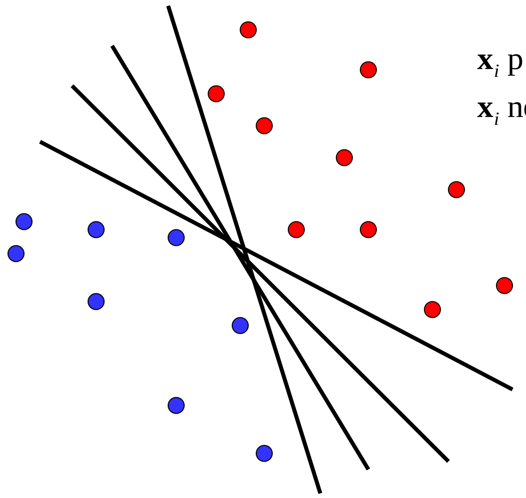
Histogram of Oriented Gradients (HoG)

- First used for application of person detection [Dalal and Triggs, CVPR 2005]
- Cited since in thousands of computer vision papers



Linear classifiers

- Find linear function to separate positive and negative examples



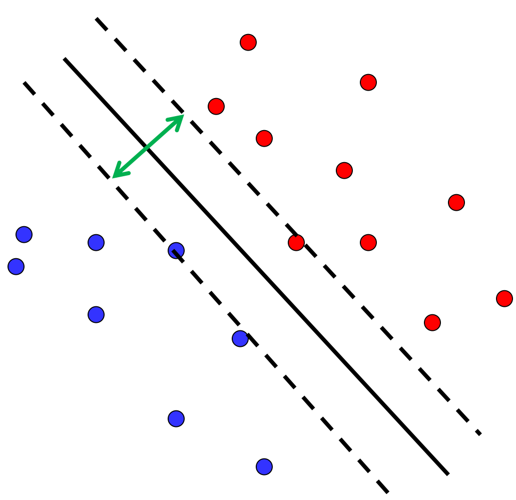
$\mathbf{x}_i$ positive: $\quad \mathbf{x}_i \cdot \mathbf{w} + b \geq 0$

$\mathbf{x}_i$ negative: $\quad \mathbf{x}_i \cdot \mathbf{w} + b < 0$

Which line is best?

[slide credit: Kristin Grauman]

Support Vector Machines (SVMs)

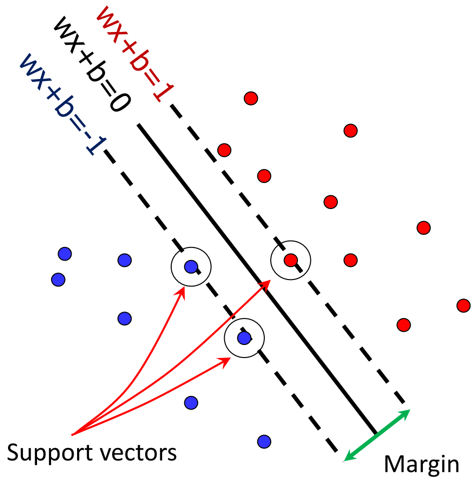


- Discriminative classifier based on *optimal separating line (for 2D case)*
- Maximize the *margin* between the positive and negative training examples

[slide credit: Kristin Grauman]

Support vector machines

- Want line that **maximizes the margin.**



$\mathbf{x}_i$ positive $(y_i = 1)$: $\quad \mathbf{x}_i \cdot \mathbf{w} + b \geq 1$

$\mathbf{x}_i$ negative $(y_i = -1)$: $\quad \mathbf{x}_i \cdot \mathbf{w} + b \leq -1$

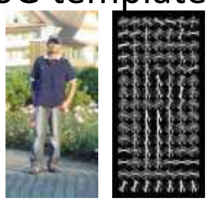
For support, vectors, $\quad \mathbf{x}_i \cdot \mathbf{w} + b = \pm 1$

C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998

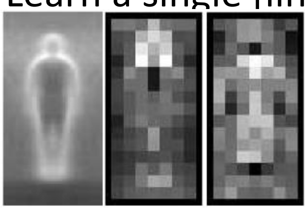
[slide credit: Kristin Grauman]

Person detection, ca. 2005

1. Represent each example with a single, fixed HoG template



2. Learn a single [linear] SVM as a detector



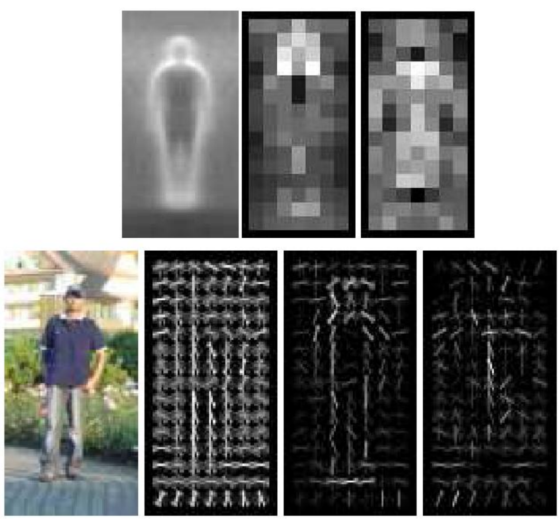
(average gradient image, max positive weight, max negative weight)

Positive and negative examples

+ thousands more…

+ millions more…

HoG templates for person detection



Detection

- Run detector as a sliding window over an image, at a range of different scales
- Non-maxima suppression

Person detection with HoG & linear SVM

[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

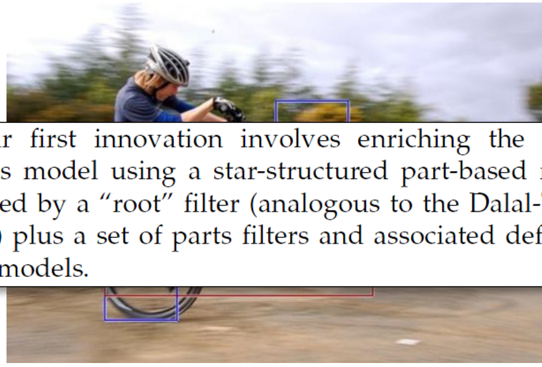
  - Many object categories look very different from different viewpoints, or from instance to instance

Difficulty of representing positive instances

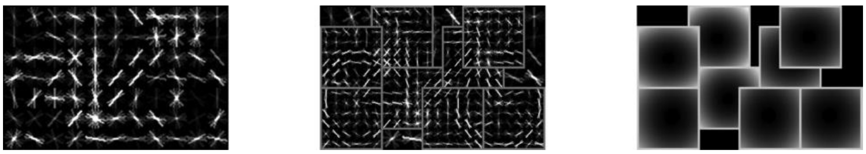
- Discriminative methods have proven very powerful
- But linear SVM on HoG templates not sufficient?

- Alternatives:
  - Parts-based models [Felzenszwalb et al. CVPR 2008]
  - Latent SVMs [Felzenszwalb et al. CVPR 2008]

Parts-based models



> Our first innovation involves enriching the Dalal-Triggs model using a star-structured part-based model defined by a "root" filter (analogous to the Dalal-Triggs filter) plus a set of parts filters and associated deformation models.

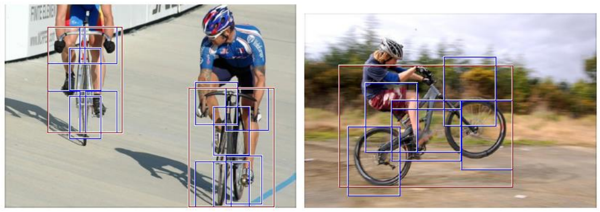


Felzenszwalb, et al., **Discriminatively Trained Deformable Part Models,**
http://people.cs.uchicago.edu/~pff/latent/

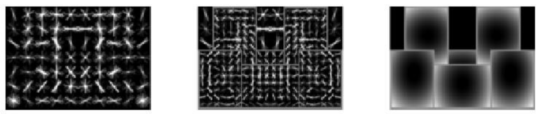
Latent SVMs

- Rather than training a single linear SVM separating positive examples…
- … cluster positive examples into "components" and train a classifier for each (using all negative examples)

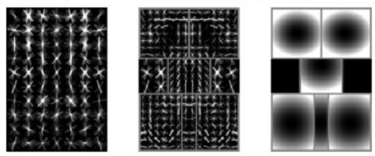
Two-component bicycle model



“side” component



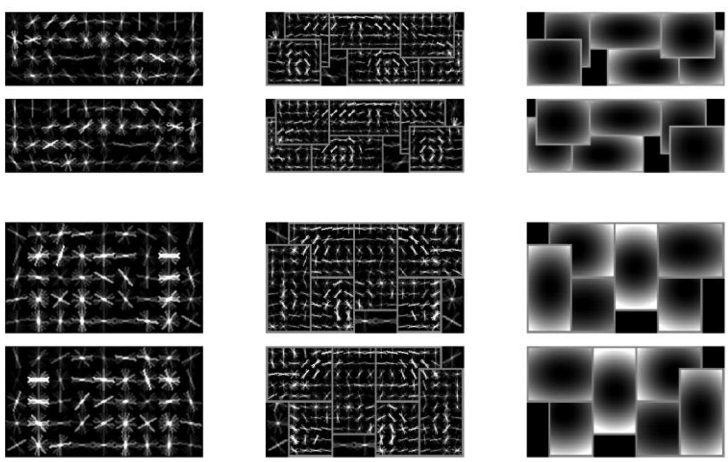
“frontal” component



Six-component car model

side view

frontal view



root filters (coarse) | part filters (fine) | deformation models

Six-component person model