Human Detection

Greg Mori
CMPT888
Outline

• Human detection in images
  – Histograms of Oriented Gradients (HOG)
    • Dalal and Triggs CVPR 2005
  – Latent SVM (L-SVM)
    • Part-based model
    • Felzenszwalb et al. CVPR 2008

• Human detection in videos
  – Cascade of boosted classifiers
    • Viola et al. ICCV 2003
  – Motion HOG
    • Dalal et al. ECCV 2006
HISTOGRAMS OF ORIENTED GRADIENTS FOR HUMAN DETECTION

Slides from Navneet Dalal
Goals & Applications

Goal: Detect and localise people in images and videos

Applications:
- Images, films & multi-media analysis
- Pedestrian detection for smart cars
- Visual surveillance, behavior analysis
Difficulties

Wide variety of articulated poses
Variable appearance and clothing
Complex backgrounds
Unconstrained illumination
Occlusions, different scales

Videos sequences involves motion of the subject, the camera and the objects in the background

Main assumption: upright fully visible people
Static Feature Extraction

- **Input image**
  - Detection window

- **Normalise gamma**

- **Compute gradients**

- **Weighted vote in spatial & orientation cells**

- **Contrast normalise over overlapping spatial cells**

- **Collect HOGs over detection window**

- **Linear SVM**

- **Feature vector** $f = \left[ \ldots, \ldots, \ldots \right]$
Overview of Learning Phase

Learning phase

Input: Annotations on training images

Create fixed-resolution normalised training image data set

Encode images into feature spaces

Learn binary classifier

Resample negative training images to create hard examples

Encode images into feature spaces

Learn binary classifier

Object/Non-object decision

Retraining reduces false positives by an order of magnitude!
HOG Descriptors

Parameters
- Gradient scale
- Orientation bins
- Percentage of block overlap

Schemes
- RGB or Lab, colour/gray-space
- Block normalisation
  - \( L_2 \)-norm,
    \[ v \leftarrow v / \sqrt{\|v\|^2 + \epsilon} \]
  - \( L_1 \)-norm,
    \[ v \leftarrow \sqrt{v / (\|v\|_1 + \epsilon)} \]

R-HOG/SIFT

C-HOG
## Evaluation Data Sets

<table>
<thead>
<tr>
<th>MIT pedestrian database</th>
<th>INRIA person database</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.jpg" alt="Image" /></td>
<td><img src="image2.jpg" alt="Image" /></td>
</tr>
<tr>
<td><img src="image3.jpg" alt="Image" /></td>
<td><img src="image4.jpg" alt="Image" /></td>
</tr>
<tr>
<td><img src="image5.jpg" alt="Image" /></td>
<td><img src="image6.jpg" alt="Image" /></td>
</tr>
<tr>
<td><img src="image7.jpg" alt="Image" /></td>
<td><img src="image8.jpg" alt="Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Test</th>
<th>Train</th>
</tr>
</thead>
<tbody>
<tr>
<td>507 positive windows</td>
<td>1208 positive windows</td>
</tr>
<tr>
<td>Negative data unavailable</td>
<td>1218 negative images</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Test</th>
<th>Train</th>
</tr>
</thead>
<tbody>
<tr>
<td>200 positive windows</td>
<td>566 positive windows</td>
</tr>
<tr>
<td>Negative data unavailable</td>
<td>453 negative images</td>
</tr>
</tbody>
</table>

| Overall 709 annotations+ reflections | Overall 1774 annotations+ reflections |
Overall Performance

MIT pedestrian database

INRIA person database

R/C-HOG give near perfect separation on MIT database
Have 1-2 order lower false positives than other descriptors
Performance on INRIA Database

![Graph showing DET curve for different descriptors on INRIA database](image-url)
Effect of Parameters

Gradient smoothing, $\sigma$

Reducing gradient scale from 3 to 0 decreases false positives by 10 times

Orientation bins, $\beta$

Increasing orientation bins from 4 to 9 decreases false positives by 10 times
Normalisation Method & Block Overlap

Normalisation method

![Graph: DET – effect of normalization methods]

- L2-Hys
- L2-norm
- L1-Sqrt
- L1-norm
- No norm
- Window norm

Strong local normalisation is essential

Block overlap

![Graph: DET – effect of overlap (cell size=8, num cell = 2x2, wt=0)]

- overlap = 3/4, stride = 4
- overlap = 1/2, stride = 8
- overlap = 0, stride = 16

Overlapping blocks improve performance, but descriptor size increases
Effect of Block and Cell Size

Trade off between need for local spatial invariance and need for finer spatial resolution
Descriptor Cues

Most important cues are head, shoulder, leg silhouettes
Vertical gradients inside a person are counted as negative
Overlapping blocks just outside the contour are most important
Overview of Methodology

Detection Phase

- Scan image(s) at all scales and locations
- Extract features over windows
- Run linear SVM classifier on all locations
- Fuse multiple detections in 3-D position & scale space
- Object detections with bounding boxes

Focus on building robust feature sets (static & motion)
Multi-Scale Object Localisation

Multi-scale dense scan of detection window

Bias

Clip Detection Score

Threshold

\[ H_i = [\exp(s_i)\sigma_x, \exp(s_i)\sigma_y, \sigma_s] \]

\[ f(x) = \sum_i w_i \exp\left(-\frac{\|x - x_i\|}{H_{i}^{-1}}^2 / 2\right) \]

Apply robust mode detection, like mean shift
Effect of Spatial Smoothing

Spatial smoothing aspect ratio as per window shape, smallest sigma approx. equal to stride/cell size
Relatively independent of scale smoothing, sigma equal to 0.4 to 0.7 octaves gives good results
Effect of Other Parameters

**Different mappings**

Recall–Precision -- effect of function $t(w)$

Hard clipping of SVM scores gives the best results than simple probabilistic mapping of these scores

**Effect of scale-ratio**

Recall–Precision -- effect of scale-ratio

Fine scale sampling helps improve recall
DETECTING HUMANS USING A PART-BASED MODEL

Felzenszwalb et al., A Discriminatively Trained, Multiscale, Deformable Part Model, CVPR 2008

Slides from Pedro Felzenszwalb
PASCAL Challenge

- ~10,000 images, with ~25,000 target objects
  - Objects from 20 categories (person, car, bicycle, cow, table...)
  - Objects are annotated with labeled bounding boxes
Why is it hard?

• Objects in rich categories exhibit significant variability
  – Photometric variation
  – Viewpoint variation
  – Intra-class variability
    – Cars come in a variety of shapes (sedan, minivan, etc)
    – People wear different clothes and take different poses

We need rich object models
But this leads to difficult matching and training problems
Starting point: sliding window classifiers

Feature vector

\[ x = [..., ..., ..., ...] \]

- Detect objects by testing each subwindow
  - Reduces object detection to binary classification
  - Dalal & Triggs: HOG features + linear SVM classifier
  - Previous state of the art for detecting people
Histogram of Gradient (HOG) features

• Image is partitioned into 8x8 pixel blocks
• In each block we compute a histogram of gradient orientations
  – Invariant to changes in lighting, small deformations, etc.
• Compute features at different resolutions (pyramid)
HOG Filters

- Array of weights for features in subwindow of HOG pyramid
- Score is dot product of filter and feature vector

\[
\text{Score of } F \text{ at position } p \text{ is } F \cdot \phi(p, H)
\]

\[
\phi(p, H) = \text{concatenation of HOG features from subwindow specified by } p
\]
Dalal & Triggs: HOG + linear SVMs

There is much more background than objects

Start with random negatives and repeat:

1) Train a model
2) Harvest false positives to define “hard negatives”
Overview of our models

- Mixture of deformable part models
- Each component has global template + deformable parts
- Fully trained from bounding boxes alone
2 component bicycle model

Each component has a root filter $F_0$ and $n$ part models $(F_i, v_i, d_i)$
Object hypothesis

Image pyramid

HOG feature pyramid

Multiscale model captures features at two-resolutions

Score is sum of filter scores minus deformation costs

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

\[ p_0 : \text{location of root} \]

\[ p_1, \ldots, p_n : \text{location of parts} \]

Multiscale model captures features at two-resolutions
Score of a hypothesis

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

concatenation filters and deformation parameters

concatenation of HOG features and part displacement features

\[
\text{score}(z) = \beta \cdot \Psi(H, z)
\]
Matching

• Define an overall score for each root location
  - Based on best placement of parts

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

• High scoring root locations define detections
  - “sliding window approach”

• Efficient computation: dynamic programming + generalized distance transforms (max-convolution)
head filter

Response of filter in l-th pyramid level

\[ R_l(x, y) = F \cdot \phi(H, (x, y, l)) \]

cross-correlation

Transformed response

\[ D_l(x, y) = \max_{d_x, d_y} \left( R_l(x + d_x, y + d_y) - d_i \cdot (d_x^2, d_y^2) \right) \]

max-convolution, computed in linear time
(spreading, local max, etc)
feature map → response of root filter → transformed responses

model

feature map at twice the resolution → response of part filters → transformed responses

combined score of root locations

color encoding of filter response values
Matching results

(after non-maximum suppression)

~1 second to search all scales
Training

• Training data consists of images with labeled bounding boxes.
• Need to learn the model structure, filters and deformation costs.
Latent SVM (MI-SVM)

Classifiers that score an example \( x \) using

\[
f_\beta(x) = \max_{z \in Z(x)} \beta \cdot \Phi(x, z)
\]

\( \beta \) are model parameters
\( z \) are latent values

Training data \( D = (\langle x_1, y_1 \rangle, \ldots, \langle x_n, y_n \rangle) \quad y_i \in \{-1, 1\} \)

We would like to find \( \beta \) such that: \( y_i f_\beta(x_i) > 0 \)

Minimize

\[
L_D(\beta) = \frac{1}{2} \|\beta\|^2 + C \sum_{i=1}^{n} \max(0, 1 - y_i f_\beta(x_i))
\]
Semi-convexity

- Maximum of convex functions is convex
- \( f_\beta(x) = \max_{z \in Z(x)} \beta \cdot \Phi(x, z) \) is convex in \( \beta \)
- \( \max(0, 1 - y_i f_\beta(x_i)) \) is convex for negative examples

\[
L_D(\beta) = \frac{1}{2} ||\beta||^2 + C \sum_{i=1}^{n} \max(0, 1 - y_i f_\beta(x_i))
\]

Convex if latent values for positive examples are fixed
Latent SVM training

\[ L_D(\beta) = \frac{1}{2} \|\beta\|^2 + C \sum_{i=1}^{n} \max(0, 1 - y_i f_{\beta}(x_i)) \]

- Convex if we fix \( z \) for **positive** examples

- Optimization:
  - Initialize \( \beta \) and iterate:
    - Pick best \( z \) for each positive example
    - Optimize \( \beta \) via gradient descent with data-mining
Training Models

- Reduce to Latent SVM training problem
- Positive example specifies some $z$ should have high score
- Bounding box defines range of root locations
  - Parts can be anywhere
  - This defines $Z(x)$
Background

- Negative example specifies no $z$ should have high score
- One negative example per root location in a background image
  - Huge number of negative examples
  - Consistent with requiring low false-positive rate
Training algorithm, nested iterations

- Fix “best” positive latent values for positives
- Harvest high scoring (x,z) pairs from background images
- Update model using gradient descent
- Trow away (x,z) pairs with low score

- Sequence of training rounds
  - Train root filters
  - Initialize parts from root
  - Train final model
Person model

root filters
coarse resolution
part filters
finer resolution
deformation models
Person detections

high scoring true positives

high scoring false positives (not enough overlap)
Quantitative results

• 7 systems competed in the 2008 challenge

• Out of 20 classes we got:
  - First place in 7 classes
  - Second place in 8 classes

• Some statistics:
  - It takes ~2 seconds to evaluate a model in one image
  - It takes ~4 hours to train a model
  - MUCH faster than most systems.
HUMAN DETECTION IN VIDEO
Motion is Helpful!

• Humans can perceive human figure presence and action in videos
  – Even from solely from body joint positions
  – Even in clutter

• Moving light displays
  – Ideas used by Song et al. CVIU 2000
CASCADE OF BOOSTED FEATURES FOR DETECTING PEDESTRIANS

Viola, Jones, and Snow, Detecting pedestrians using patterns of motion and appearance, ICCV 2003
Viola-Jones

• Viola-Jones face detector
  – Viola and Jones CVPR 2001
  – Window-scanning approach

• Two nice ideas
  – Define many, efficient-to-compute features
    • AdaBoost to select good ones from them
  – Cascade architecture to quickly eliminate non-face sub-windows
Adaboost Algorithm

• Given a set of “weak learners”
  \[ h_i(x) \in \{+1, -1\} \]

• Build “strong learner”
  \[ h(x) = \sum_{t=1}^{T} \alpha_t h_t(x) \]
  – Greedy selection of weak learners
  – Each iteration, choose best weak learner
AdaBoost Algorithm
Face Features

• Features – Haar-like rectangle features
• Each weak learner examines a single feature

\[ h_j(x) = \begin{cases} 
1 & \text{if } p_j f_j(x) < p_j \theta_j \\
0 & \text{otherwise} 
\end{cases} \]
Integral Images

- Fast computation of features possible using Integral Images
Cascade of Classifiers

- Most image sub-windows don’t contain a face
Learned Classifier

- First two weak learners chosen:
And People?

- Same algorithm, slightly different features
- Diagonal to capture legs
- Frame differencing for motion
MOTION HOG

Dalal, Triggs, and Schmid, Human Detection Using Oriented Histograms of Flow and Appearance, ECCV 2006
Slides from Navneet Dalal
Motion HOG Processing Chain

Detection windows →

Normalise gamma & colour

Compute optical flow

Compute differential flow

Accumulate votes for differential flow orientation over spatial cells

Normalise contrast within overlapping blocks of cells

Collect HOGs for all blocks over detection window

Input image →

Consecutive image

Flow field

Magnitude of flow

Differential flow X

Differential flow Y

Block

Overlap of Blocks

Cell
Overview of Feature Extraction

- **Input image**
  - Appearance Channel
    - Static HOG Encoding
    - Consecutive image(s)
      - Motion HOG Encoding

- **Collect HOGs over detection window**
- **Linear SVM**
- **Object/Non-object decision**

**Data Set**

<table>
<thead>
<tr>
<th></th>
<th>Train</th>
<th>Test 1</th>
<th>Test 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5 DVDs, 182 shots 5562 positive windows</td>
<td>Same 5 DVDs, 50 shots 1704 positive windows</td>
<td>6 new DVDs, 128 shots 2700 positive windows</td>
</tr>
</tbody>
</table>
Coding Motion Boundaries

Treat $x$, $y$-flow components as independent images.
Take their local gradients separately, and compute HOGs as in static images.

Motion Boundary Histograms (MBH) encode depth and motion boundaries.
Coding Internal Dynamics

Ideally compute relative displacements of different limbs
  Requires reliable part detectors
Parts are relatively localised in our detection windows
Allows different coding schemes based on fixed spatial differences

Internal Motion Histograms (IMH) encode relative dynamics of different regions
Simple difference
Take $x, y$ differentials of flow vector images $[I_x, I_y]$
Variants may use larger spatial displacements while differencing, e.g. $[1 0 0 0 -1]$

Center cell difference

Wavelet-style cell differences
Summary

• Large literature on human detection
  – These are a few, widely used, examples
    • Code is available
    – Ask me for reading list of others

• Encode shape and motion
  – Gradient filters
  – Motion histograms

• Encode spatial variability
  – Part-based models