Recap: Sliding-Window Object Detection

- If object may be in a cluttered scene, slide a window around looking for it.

- Essentially, this is a brute-force approach with many local decisions.

Recap: HOG Descriptor Processing Chain

- Train a pedestrian template using a linear SVM
- At test time, convolve feature map with learned template \( w \)

Recap: Pedestrian Detection with HOG

Classify object as pedestrian or non-pedestrian.

\[
y(x) = w^\top x + b
\]
Recap: AdaBoost

Final classifier is combination of the weak classifiers

Example: Face Detection

- Frontal faces are a good example of a class where global appearance models + a sliding window detection approach fit well:
  - Regular 2D structure
  - Center of face almost shaped like a "patch"/window

- Now we’ll take AdaBoost and see how the Viola-Jones face detector works

Feature extraction

"Rectangular" filters

Value at (x,y) is sum of pixels above and to the left of (x,y)

Large Library of Features

Considering all possible filter parameters: position, scale, and type:

180,000+ possible features associated with each 24 x 24 window

Use AdaBoost both to select the informative features and to form the classifier

Weak classifier: feature output > \theta?
AdaBoost for Feature+Classifier Selection

Want to select the single rectangle feature and threshold that best separates positive (faces) and negative (non-faces) training examples, in terms of weighted error.

Resulting weak classifier:
$$h_t(x) = \begin{cases} +1 & \text{if } f_t(x) > \theta_t \\ -1 & \text{otherwise} \end{cases}$$

For next round, reweight the examples according to errors, choose another filter/threshold combo.

Cascading Classifiers for Detection

- Even if the filters are fast to compute, each new image has a lot of possible windows to search.
- For efficiency, apply less accurate but faster classifiers first to immediately discard windows that clearly appear to be negative; e.g.,
  - Filter for promising regions with an initial inexpensive classifier
  - Build a chain of classifiers, choosing cheap ones with low false negative rates early in the chain

Recap: Viola-Jones Face Detector

- Train with 5K positives, 350M negatives
- Real-time detector using 38 layer cascade
- 6061 features in final layer
- [Implementation available in OpenCV: http://sourceforge.net/projects/opencvlibrary/]

You Can Try It At Home…

- The Viola & Jones detector was a huge success
  - First real-time face detector available
  - Many derivative works and improvements

- C++ implementation available in OpenCV [Lienhart, 2002]
  - http://sourceforge.net/projects/opencvlibrary/
- Matlab wrappers for OpenCV code available, e.g. here

P. Viola, M. Jones, Robust Real-Time Face Detection, IJCV, Vol. 57(2), 2004
Limitations: Changing Aspect Ratios

- Sliding window requires fixed window size
  - Basis for learning efficient cascade classifier
- How to deal with changing aspect ratios?
  - Fixed window size
    - Wastes training dimensions
  - Adapted window size
    - Difficult to share features
    - “Squashed” views [Dalal & Triggs]
      - Need to squash test image, too

Limitations (continued)

- Not all objects are “box” shaped

Limitations (continued)

- Non-rigid, deformable objects not captured well with representations assuming a fixed 2D structure; or must assume fixed viewpoint
- Objects with less-regular textures not captured well with holistic appearance-based descriptions

Limitations (continued)

- If considering windows in isolation, context is lost

Topics of This Lecture

- Local Invariant Features
  - Motivation
  - Requirements, Invariances
- Keypoint Localization
  - Harris detector
  - Hessian detector
- Scale Invariant Region Selection
  - Automatic scale selection
  - Laplacian-of-Gaussian detector
  - Difference-of-Gaussian detector
  - Combinations
- Local Descriptors
  - Orientation normalization
  - SIFT

Motivation

- Global representations have major limitations
- Instead, describe and match only local regions
- Increased robustness to
  - Occlusions
  - Articulation
  - Intra-category variations
Application: Image Matching

- Harder Case
  - NASA Mars Rover images

- Harder Still?
  - NASA Mars Rover images with SIFT feature matches

Application: Image Stitching

- Procedure:
  - Detect feature points in both images

Answer Below (Look for Tiny Colored Squares)
Application: Image Stitching

- Procedure:
  - Detect feature points in both images
  - Find corresponding pairs

General Approach

1. Find a set of distinctive keypoints
2. Define a region around each keypoint
3. Extract and normalize the region content
4. Compute a local descriptor from the normalized region
5. Match local descriptors

Common Requirements

- Problem 1:
  - Detect the same point independently in both images
- Problem 2:
  - For each point correctly recognize the corresponding one

We need a reliable and distinctive descriptor!

Invariance: Geometric Transformations

- No chance to match!
- We need a repeatable detector!
Levels of Geometric Invariance

Requirements

- Region extraction needs to be repeatable and accurate
  - Invariant to translation, rotation, scale changes
  - Robust or covariant to out-of-plane (=affine) transformations
  - Robust to lighting variations, noise, blur, quantization
- Locality: Features are local, therefore robust to occlusion and clutter.
- Quantity: We need a sufficient number of regions to cover the object.
- Distinctiveness: The regions should contain “interesting” structure.
- Efficiency: Close to real-time performance.

Many Existing Detectors Available

- Hessian & Harris [Beaudet '78], [Harris '88]
- Laplacian, DoG [Lindeberg '88], [Lowe '99]
- Harris-/Hessian-Laplace [Mikolajczyk & Schmid '01]
- Harris-/Hessian-Affine [Mikolajczyk & Schmid '04]
- EBR and IBR [Tuytelaars & Van Gool '04]
- MSER [Matas '02]
- Salient Regions [Kadir & Brady '01]
- Others…
- Those detectors have become a basic building block for many applications in Computer Vision.

Keypoint Localization

Key property:

- In the region around a corner, image gradient has two or more dominant directions
- Corners are repeatable and distinctive

Findings:

- Repeatable detection
- Precise localization
- Interesting content

⇒ Look for two-dimensional signal changes

Corners as Distinctive Interest Points

- Design criteria
  - We should easily recognize the point by looking through a small window (locality)
  - Shifting the window in any direction should give a large change in intensity (good localization)

"Flat" region: no change in all directions
"edge": no change along the edge direction
"corner": significant change in all directions
**Harris Detector Formulation**

- Start from the second-moment matrix $M$:

$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

Sum over image region – the area we are checking for corner

$$M = \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \begin{bmatrix} I_x & I_y \\ I_y & I_y \end{bmatrix} \begin{bmatrix} I_x & I_y \\ I_y & I_y \end{bmatrix}$$

**What Does This Matrix Reveal?**

- Since $M$ is symmetric, we have $M = R^T \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$ (Eigenvalue decomposition)

- We can visualize $M$ as an ellipse with axis lengths determined by the eigenvalues and orientation determined by $R$

**Interpreting the Eigenvalues**

- Classification of image points using eigenvalues of $M$:

  - $\lambda_2$, $\lambda_2$ are small; $\alpha$ is almost constant in all directions
  - $\lambda_1$ and $\lambda_2$ are large; $\lambda_1 \sim \lambda_2$; $R$ increases in all directions

**Corner Response Function**

$$R = \det(M) - \alpha \text{trace}(M)^2 = \lambda_1 \lambda_2 - \alpha (\lambda_1 + \lambda_2)^2$$

- Fast approximation
  - Avoid computing the eigenvalues
  - $\alpha$: constant (0.04 to 0.06)

**Window Function $w(x,y)$**

$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

- Option 1: uniform window
  - Sum over square window
    $$M = \sum_{x,y} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$
  - Problem: not rotation invariant

- Option 2: Smooth with Gaussian
  - Gaussian already performs weighted sum
    $$M = g(\sigma) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$
  - Result is rotation invariant

**Summary: Harris Detector [Harris88]**

- Compute second moment matrix (autocorrelation matrix)
  $$M(\sigma_1,\sigma_2) = g(\sigma) \begin{bmatrix} I_x^2(\sigma) & I_x I_y(\sigma) \\ I_x I_y(\sigma) & I_y^2(\sigma) \end{bmatrix}$$

- 1. Image derivatives
- 2. Square of derivatives
- 3. Gaussian filter (g)

- 4. Cornerness function – two strong eigenvalues
  $$R = \det(M(\sigma_1,\sigma_2)) - \alpha \text{trace}(M(\sigma_1,\sigma_2))^2 = g(I_x^2(\sigma_1)) - [g(I_x I_y(\sigma_1))]^2 - \alpha [g(I_y^2(\sigma_1))]^2$$

- 5. Perform non-maximum suppression
Harris Detector: Workflow

• Compute corner responses $R$

• Take only the local maxima of $R$, where $R > \text{threshold}$.

Harris Detector – Responses [Harris88]

Effect: A very precise corner detector.

• Results are well suited for finding stereo correspondences

Slide credit: Kristen Grauman
Harris Detector: Properties

- Rotation invariance?

![Diagram of rotation invariance](image)

Ellipse rotates but its shape (i.e., eigenvalues) remains the same.

**Corner response \( K \) is invariant to image rotation**

Harris Detector: Properties

- Rotation invariance
- Scale invariance?

![Diagram of scale invariance](image)

Corner

**Not invariant to image scale!**

Hessian Detector [Beaudet78]

- Hessian determinant

\[
\text{Hessian}(I) = \begin{bmatrix}
I_{xx} & I_{xy} \\
I_{xy} & I_{yy}
\end{bmatrix}
\]

**Note:** these are 2nd derivatives!

Intuition: Search for strong derivatives in two orthogonal directions.

Hessian Detector – Responses [Beaudet78]

**Effect:** Responses mainly on corners and strongly textured areas.

Topics of This Lecture

- Local Invariant Features
  - Motivation
  - Requirements, Invariances
- Keypoint Localization
  - Harris detector
  - Hessian detector
- Scale Invariant Region Selection
  - Automatic scale selection
  - Laplacian-of-Gaussian detector
  - Difference-of-Gaussian detector
  - Combinations
- Local Descriptors
  - Orientation normalization
  - SIFT
From Points to Regions...

- The Harris and Hessian operators define interest points.
  - Precise localization
  - High repeatability

- In order to compare those points, we need to compute a descriptor over a region.
  - How can we define such a region in a scale invariant manner?
  - I.e. how can we detect scale invariant interest regions?

Naïve Approach: Exhaustive Search

- Comparing descriptors while varying the patch size
  - Computationally inefficient
  - Inefficient but possible for matching
  - Prohibitive for retrieval in large databases
  - Prohibitive for recognition

Automatic Scale Selection

- Solution:
  - Design a signature function on the region that is "scale invariant" (the same for corresponding regions, even if they are at different scales)

  - For a point in one image, we can consider it as a function of region size (patch width)

  \[ f() \]

  \[ \text{Region size} \]

  \[ \text{Image 1} \]

  \[ \text{scale} = \frac{1}{2} \]

  \[ \text{Region size} \]

  \[ \text{Image 2} \]

  \[ f() \]

  \[ \text{Region size} \]

  \[ \text{Image 1} \]

  \[ \text{scale} = \frac{1}{2} \]

  \[ \text{Region size} \]

  \[ \text{Image 2} \]

Important: this scale invariant region size is found in each image independently!
Automatic Scale Selection

- Function responses for increasing scale (scale signature)

Slide credit: Krystian Mikolajczyk

Automatic Scale Selection

- Function responses for increasing scale (scale signature)

Slide credit: Krystian Mikolajczyk

Automatic Scale Selection

- Function responses for increasing scale (scale signature)

Slide credit: Krystian Mikolajczyk

Automatic Scale Selection

- Function responses for increasing scale (scale signature)

Slide credit: Krystian Mikolajczyk

Automatic Scale Selection

- Normalize: Rescale to fixed size

Slide credit: Tinne Tuytelaars

What Is A Useful Signature Function?

- Laplacian-of-Gaussian = "blob" detector
Characteristic Scale

- We define the characteristic scale as the scale that produces peak of Laplacian response


Laplacian-of-Gaussian (LoG)

- Interest points:
  - Local maxima in scale space of Laplacian-of-Gaussian

Laplacian-of-Gaussian (LoG)

- Interest points:
  - Local maxima in scale space of Laplacian-of-Gaussian

Laplacian-of-Gaussian (LoG)

- Interest points:
  - Local maxima in scale space of Laplacian-of-Gaussian

\[ L_x(\sigma) + L_y(\sigma) \]

⇒ List of \((x, y, \sigma)\)

LoG Detector: Workflow
**Technical Detail**

- We can efficiently approximate the Laplacian with a difference of Gaussians:
  \[
  L = \sigma^2 (G_x(x,y,\sigma) + G_y(x,y,\sigma))
  \]
  (Laplacian)
  \[
  \text{DoG} = G(x,y,k\sigma) - G(x,y,\sigma)
  \]
  (Difference of Gaussians)

**Difference-of-Gaussian (DoG)**

- Difference of Gaussians as approximation of the LoG
  - This is used e.g. in Lowe’s SIFT pipeline for feature detection.
- Advantages
  - No need to compute 2nd derivatives
  - Gaussians are computed anyway, e.g. in a Gaussian pyramid.

**DoG – Efficient Computation**

- Computation in Gaussian scale pyramid

**Results: Lowe's DoG**
Summary: Scale Invariant Detection

- **Given:** Two images of the same scene with a large *scale difference* between them.
- **Goal:** Find the same interest points *independently* in each image.
- **Solution:** Search for maxima of suitable functions in scale and in space (over the image).

- Two strategies
  - Laplacian-of-Gaussian (LoG)
  - Difference-of-Gaussian (DoG) as a fast approximation
  - These can be used either on their own, or in combinations with single-scale keypoint detectors (Harris, Hessian).

Local Descriptors

- We know how to detect points
- Next question: How to *describe* them for matching?

You Can Try It At Home…

- For most local feature detectors, executables are available online:
  - [http://robots.ox.ac.uk/~vgg/research/affine/](http://robots.ox.ac.uk/~vgg/research/affine/)
  - [http://www.vision.ee.ethz.ch/~surf](http://www.vision.ee.ethz.ch/~surf)

References and Further Reading

- Read David Lowe’s SIFT paper
  - D. Lowe, *Distinctive image features from scale-invariant keypoints*, IJCV 60(2), pp. 91-110, 2004
- Good survey paper on Int. Pt. detectors and descriptors
- Try the example code, binaries, and Matlab wrappers
  - Good starting point: Oxford interest point page [http://www.robots.ox.ac.uk/~vgg/research/affine/detectors.html#binaries](http://www.robots.ox.ac.uk/~vgg/research/affine/detectors.html#binaries)