12/05/2011

# Computer Vision: Summary and Discussion

Computer Vision

CS 143, Brown

James Hays

Many slides from Derek Hoiem

# Computer Vision Builds On…

- Image Processing
  - to extract low-level information from images.
- Machine Learning
  - to make decisions based on data.

# Fundamentals of Computer Vision

- Geometry
  - How to relate world coordinates and image coordinates
- Matching
  - How to measure the similarity of two regions
- Alignment
  - How to align points/patches
  - How to recover transformation parameters based on matched points
- Grouping
  - What points/regions/lines belong together?
- Categorization / Recognition
  - What similarities are important?

# Geometry

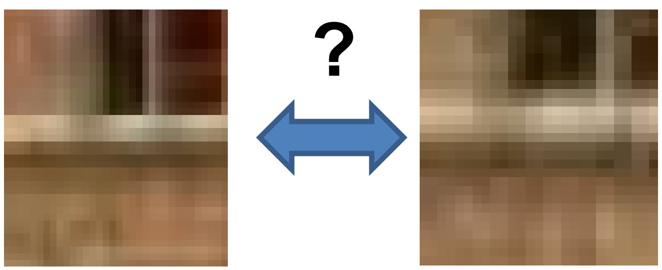
- $\mathbf{x} = \mathbf{K}\,[\mathbf{R}\ \mathbf{t}]\,\mathbf{X}$
  - Maps 3d point $\mathbf{X}$ to 2d point $\mathbf{x}$
  - Rotation $\mathbf{R}$ and translation $\mathbf{t}$ map into 3D camera coordinates
  - Intrinsic matrix $\mathbf{K}$ projects from 3D to 2D

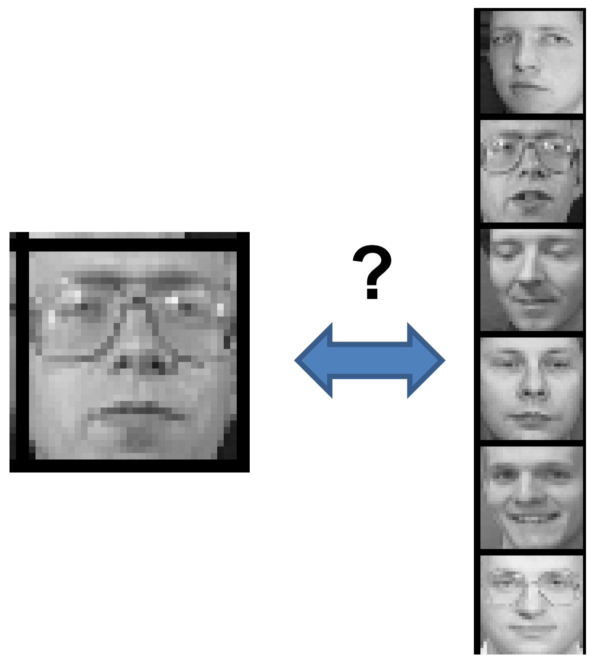
- Parallel lines in 3D converge at the **vanishing point** in the image
  - A 3D plane has a vanishing line in the image

- $\mathbf{x}'^{\mathrm{T}}\,\mathbf{F}\,\mathbf{x} = 0$
  - Points in two views that correspond to the same 3D point are related by the fundamental matrix $\mathbf{F}$

# Matching

- Does this patch match that patch?
  - In two simultaneous views? (stereo)
  - In two successive frames? (tracking, flow, SFM)
  - In two pictures of the same object? (recognition)

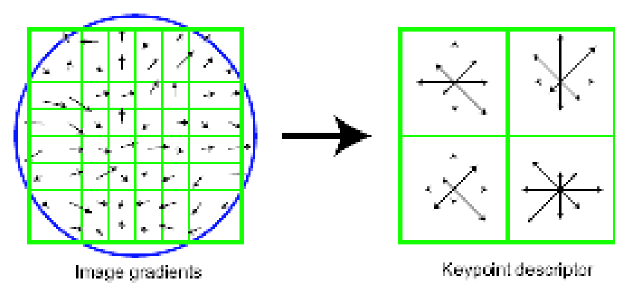


# Matching

**Representation**: be invariant/robust to expected deformations but nothing else

- Often assume that shape is constant
  - Key cue: local differences in shading (e.g., gradients)
- Change in viewpoint
  - Rotation invariance: rotate and/or affine warp patch according to dominant orientations
- Change in lighting or camera gain
  - Average intensity invariance: oriented gradient-based matching
  - Contrast invariance: normalize gradients by magnitude
- Small translations
  - Translation robustness: histograms over small regions




But can one representation do all of this?

- SIFT: local normalized histograms of oriented gradients provides robustness in-plane orientation, lighting, contrast, translation
- HOG: like SIFT but does not rotate to dominant orientation

# Alignment of points

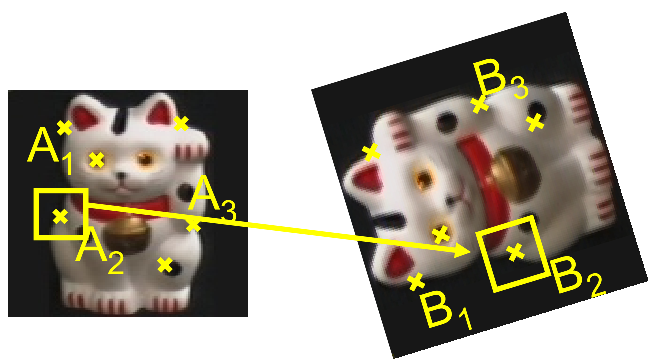
**Search**: efficiently align matching patches

- Interest points: find repeatable, distinctive points
  - Long-range matching: e.g., wide baseline stereo, panoramas, object instance recognition
  - Harris: points with strong gradients in orthogonal directions (e.g., corners) are precisely repeatable in x-y
- Local search
  - Short range matching: e.g., tracking, optical flow
  - Gradient descent on patch SSD, often with image pyramid
- Windowed search
  - Long-range matching: e.g., recognition, stereo w/ scanline

# Alignment of sets

Find transformation to align matching sets of points

- Geometric transformation (e.g., affine)
  - Least squares fit (SVD), if all matches can be trusted
  - Hough transform: each potential match votes for a range of parameters
    - Works well if there are very few parameters (3-4)
  - RANSAC: repeatedly sample potential matches, compute parameters, and check for inliers
    - Works well if fraction of inliers is high and few parameters (4-8)



# Grouping

- Clustering: group items (patches, pixels, lines, etc.) that have similar appearance
  - Discretize continuous values; typically, represent points within cluster by center
  - Improve efficiency: e.g., cluster interest points before recognition
  - Summarize data

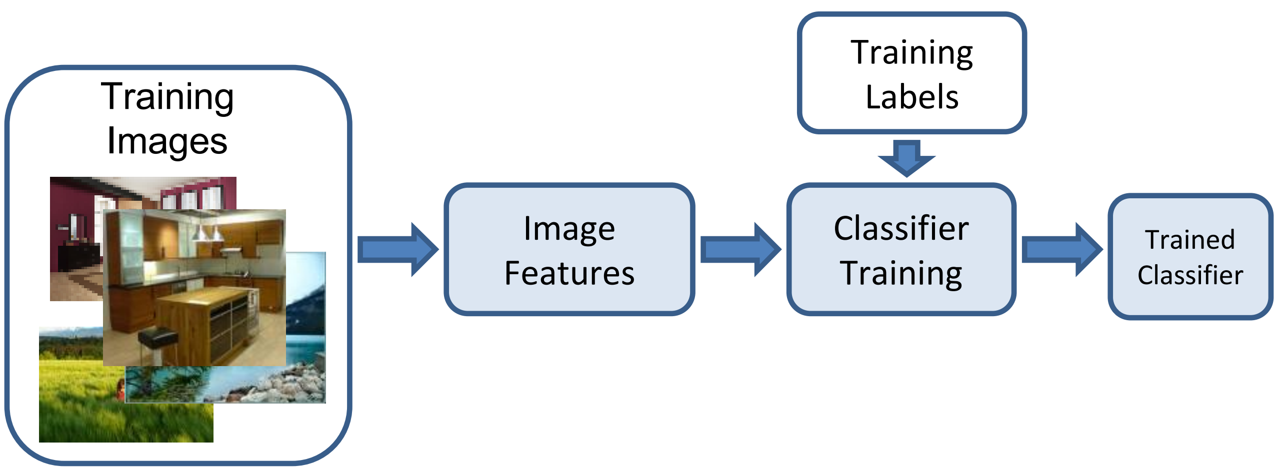
- Segmentation: group pixels into regions of coherent color, texture, motion, and/or label
  - Mean-shift clustering
  - Graph-based segmentation: e.g., MRF and graph cuts

- EM, mixture models: probabilistically group items that are likely to be drawn from the same distribution, while estimating the distributions' parameters

# Categorization

Match objects, parts, or scenes that may vary in appearance

- Categories are typically defined by human and may be related by function, location, or other non-visual attributes
- Key problem: what are important similarities?
  - Can be learned from training examples



# Categorization

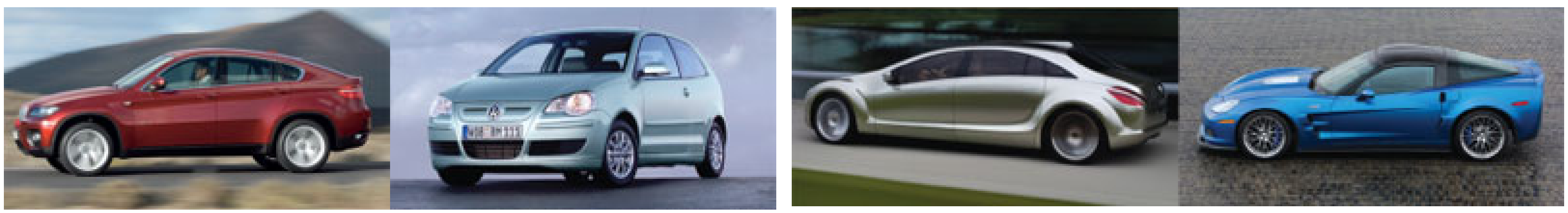
**Representation**: ideally should be compact, comprehensive, direct

- Histograms of quantized local descriptors (SIFT, HOG), color, texture
  - Typical for image or region categorization
  - Degree of spatial encoding is controllable by using spatial pyramids

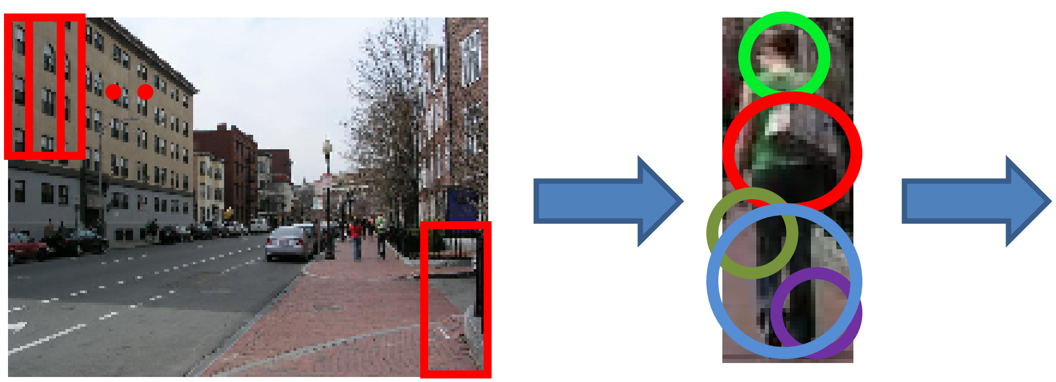
# Object Categorization

**Search** by Sliding Window Detector

- May work well for rigid objects



- Key idea: simple alignment for simple deformations



Object or Background?

# Object Categorization

**Search** by Parts-based model

- Key idea: more flexible alignment for articulated objects
- Defined by models of **part appearance**, **geometry** or spatial layout, and **search** algorithm

# Vision as part of an intelligent system

3D Scene

| | | | | | |
|---|---|---|---|---|---|
| Feature Extraction | Texture | Color | Optical Flow | Stereo Disparity | |
| Grouping | Surfaces | Bits of objects | Sense of depth | Motion patterns | |
| Interpretation | Objects | Agents and goals | Shapes and properties | Open paths | Words |
| Action | Walk, touch, contemplate, smile, evade, read on, pick up, … | | | | |

# Important open problems

Computer vision is potentially worth major $$$, but there are major challenges to overcome first.

- Driver assistance
- Entertainment (Kinect, movies, etc.)
- Security
- Robot workers
- Many more

# If you want to learn more…

- Read lots of papers: IJCV, PAMI, CVPR, ICCV, ECCV, NIPS, ICCP
- Related Classes
  - CS 2951B – Data-driven Vision and Graphics (Spring '12)
  - CS 1950F – Intro. Machine Learning (Spring '12)
  - ENGN 2520 – Pattern Rec. and Machine Learning (Spring '12)
  - ENGN 2502 – 3d Photography (Spring '12)
  - CS 123 – Computational Photography (Fall '12)
- Just implement stuff, try demos, see what works