CS 369: Introduction to Robotics

Prof. Thao Nguyen

Spring 2026

HAVERFORD
COLLEGE

Outline for today

- Object recognition

- Convolutional neural networks

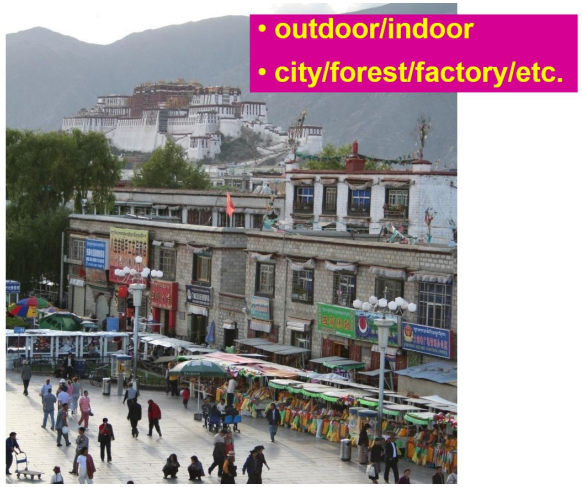
Outline for today

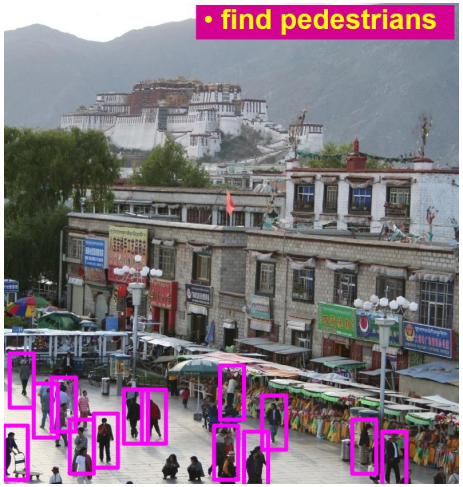
- Object recognition

- Convolutional neural networks

Recognition tasks

- Scene classification
- Image parsing / annotation
- Object detection

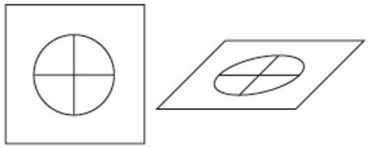




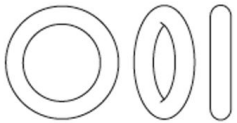
Recognition - why is it hard?

Primary challenge:

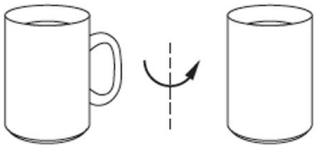
- Objects of the same class look different
- The same object looks different under different conditions



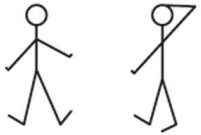
Foreshortening



Aspect



Occlusion



Deformation

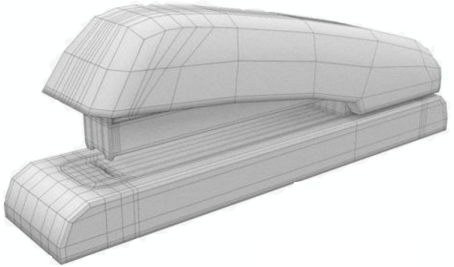
Recognition - why is it hard?

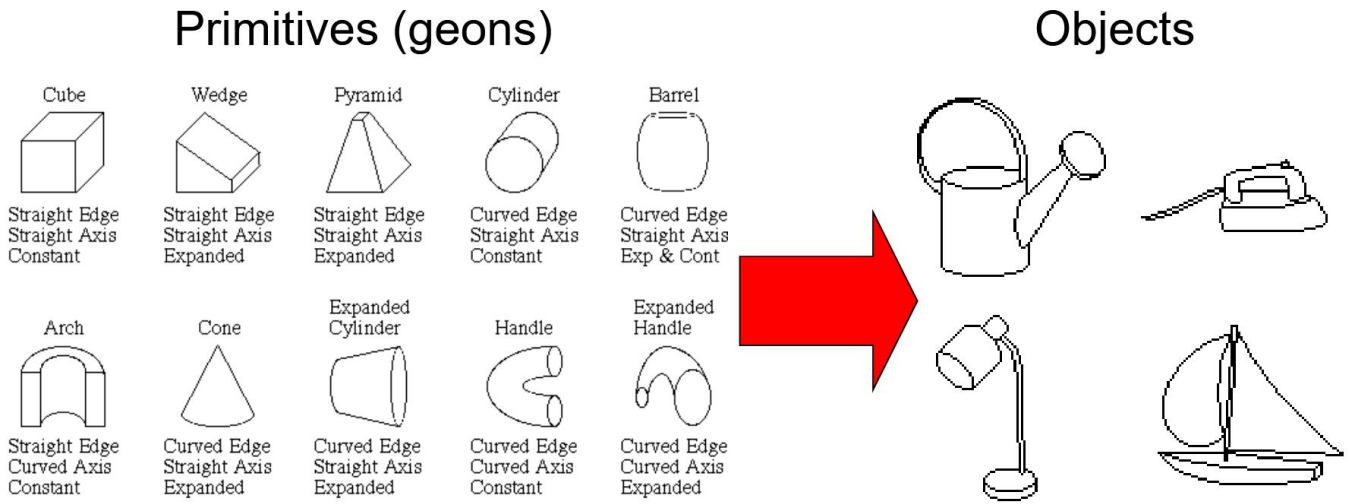
Objects of different classes may look similar

Qian Zhang, Brown University

Geometric models

- Use an object model to match an object in a scene
- Recognition by components



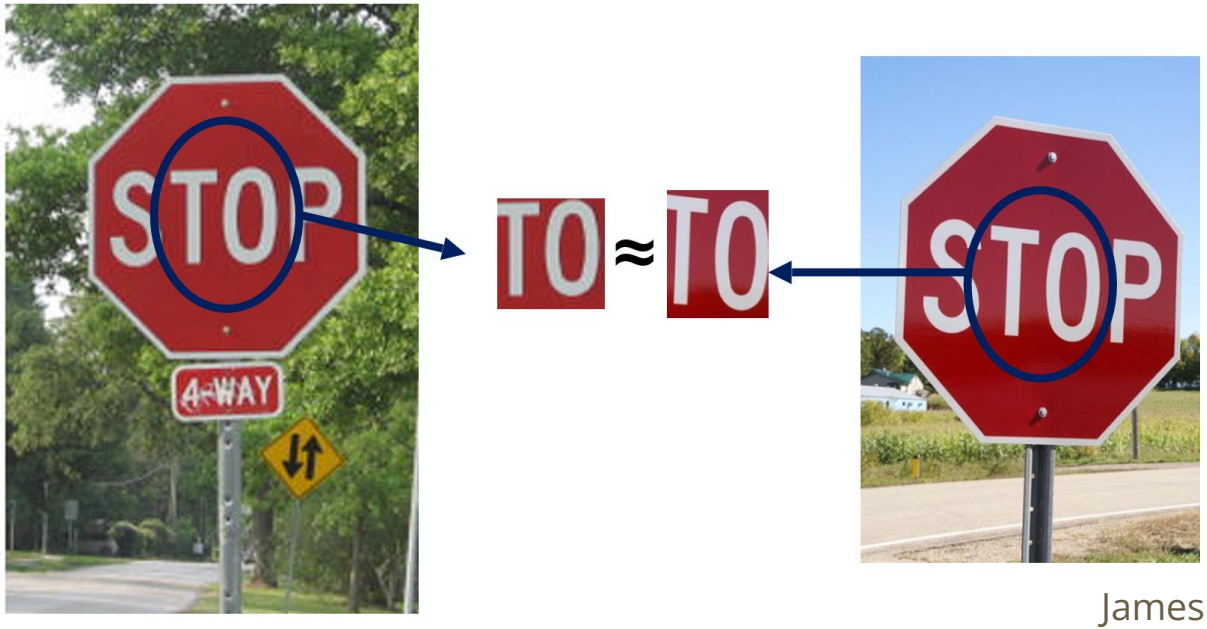


Svetlana Lazebnik, UIUC

George Konidaris,
Brown University

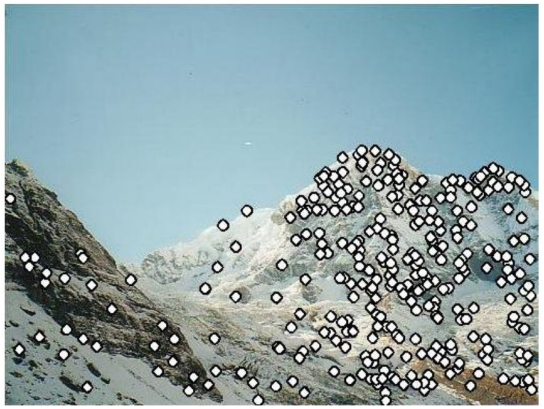
Correspondence

- Matching points, patches, edges, or regions across images
- Sparse or local correspondence vs. dense (at every pixel)



James Hays, Georgia Tech

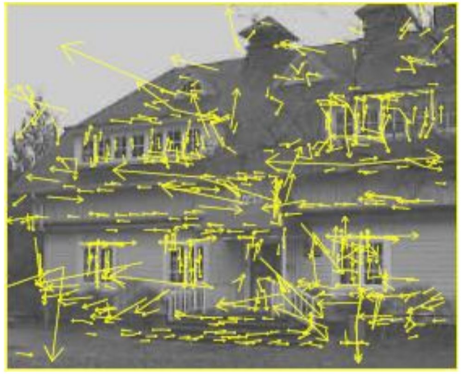
Local features

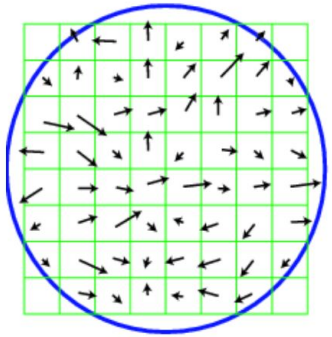


- Detection: Identify the interest points
- Description: Extract vector feature descriptor surrounding each interest point
- Matching: Determine correspondence between descriptors in two views
- Characteristics of good features: repeatability, saliency, compactness and efficiency, locality

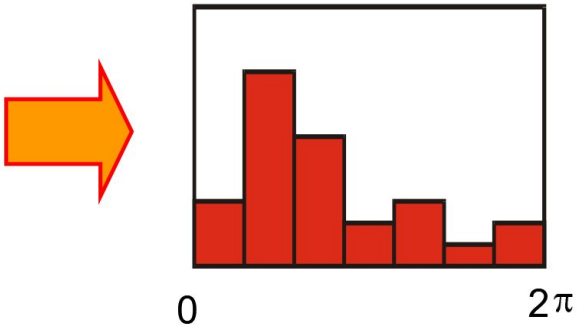
Scale-Invariant Feature Transform (SIFT) descriptors

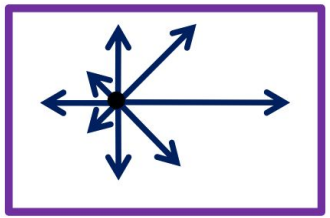
- Use histograms to bin pixels within sub-patches according to their orientation
- Descriptor normalized to account for illumination
- Rotate patch according to its dominant gradient orientation





patch gradients

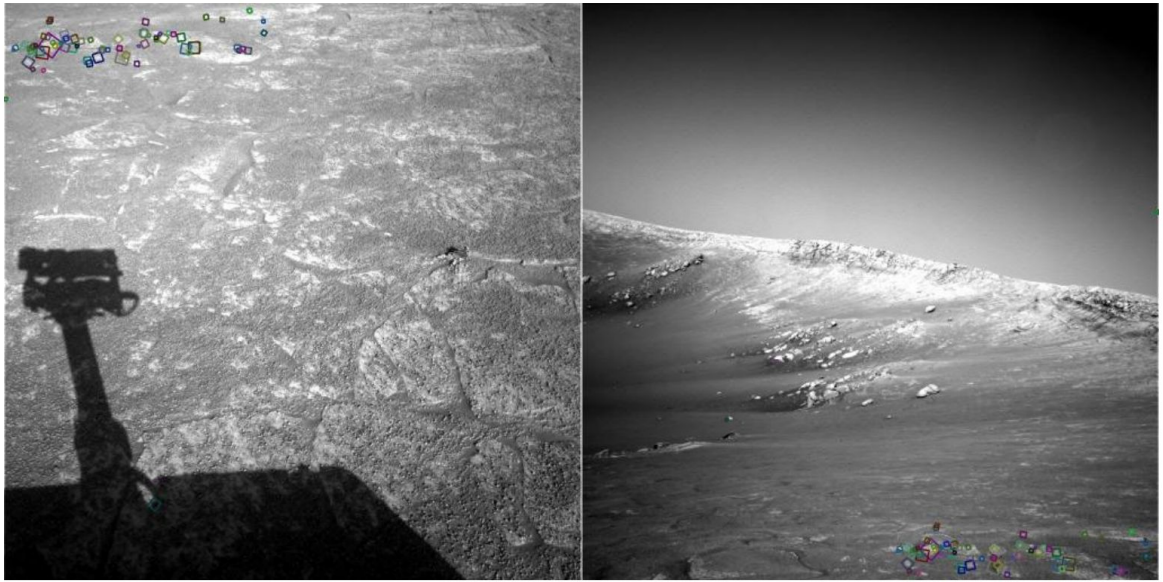




keypoint descriptor

SIFT properties

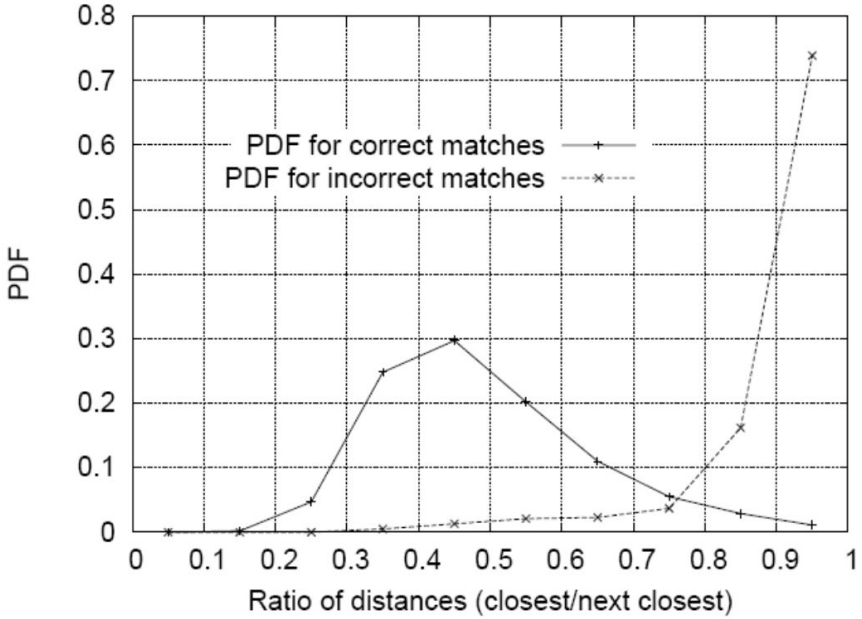
- Invariant to
  - Scale
  - Rotation
- Partially invariant to
  - Illumination changes
  - Camera viewpoint
  - Occlusion, clutter



NASA Mars Rover images
with SIFT feature matches
Figure by Noah Snavely

Matching SIFT descriptors

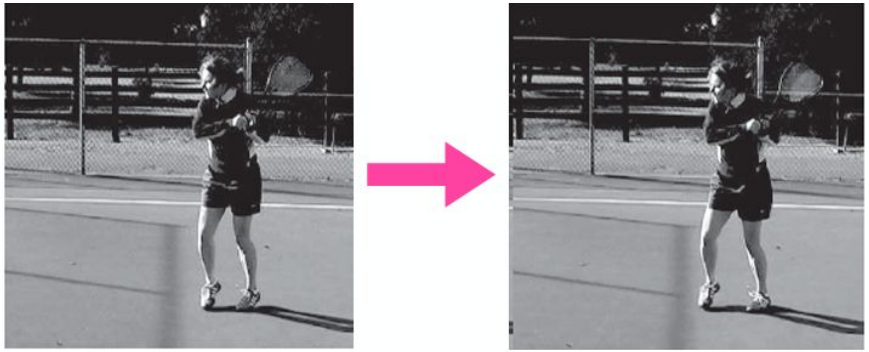
- Nearest neighbor (Euclidean distance)
- Threshold ratio of nearest to 2nd nearest descriptor



Optical flow

Given two images I_1 and I_2

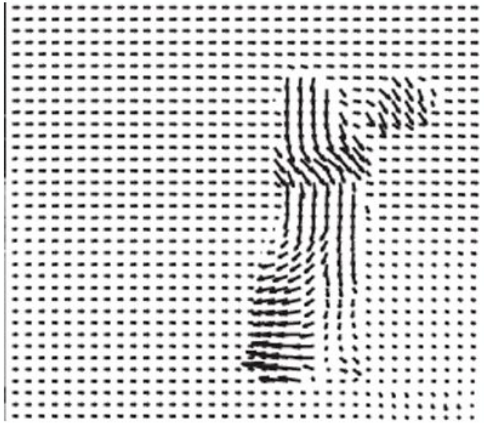
- Produce optical flow field F
  - $F(x, y) = (dx, dy)$
  - where pixel $I_1[x,y]$ moves to $I_2[x + dx, y + dy]$



This boils down to finding ***correspondences***.

One approach

- Find correspondences that minimize "patch" error
- Regularize for smaller movements



Russell & Norvig,
Artificial Intelligence: A Modern Approach

Outline for today

- Object recognition

- Convolutional neural networks

Convolutional neural networks (CNNs)

https://cs231n.github.io/convolutional-networks/