Descriptors

CSE 576
Ali Farhadi

Many slides from Larry Zitnick, Steve Seitz
How can we find corresponding points?
How can we find correspondences?
How do we describe an image patch?
How do we describe an image patch?

Patches with similar content should have similar descriptors.
Raw patches as local descriptors

The simplest way to describe the neighborhood around an interest point is to write down the list of intensities to form a feature vector.

But this is very sensitive to even small shifts, rotations.
What do human use?

Gabor filters...

... and many other things.
**SIFT descriptor**

**Full version**

- Divide the 16x16 window into a 4x4 grid of cells (2x2 case shown below)
- Compute an orientation histogram for each cell
- 16 cells * 8 orientations = 128 dimensional descriptor

![Image gradients](Image gradients)  
![Keypoint descriptor](Keypoint descriptor)

Adapted from slide by David Lowe
SIFT descriptor

Full version

• Divide the 16x16 window into a 4x4 grid of cells (2x2 case shown below)
• Compute an orientation histogram for each cell
• 16 cells * 8 orientations = 128 dimensional descriptor
• Threshold normalize the descriptor:

\[ \sum_i d_i^2 = 1 \]  

such that: \[ d_i < 0.2 \]

Adapted from slide by David Lowe
Properties of SIFT

Extraordinarily robust matching technique

- Can handle changes in viewpoint
  - Up to about 30 degree out of plane rotation
- Can handle significant changes in illumination
  - Sometimes even day vs. night (below)
- Fast and efficient—can run in real time
- Lots of code available
Example

NASA Mars Rover images with SIFT feature matches
Figure by Noah Snavely
Example: Object Recognition

SIFT is extremely powerful for object instance recognition, especially for well-textured objects

Lowe, IJCV04
Example: Google Goggles

Google Goggles in Action

Click the icons below to see the different ways Google Goggles can be used.
panorama?

- We need to match (align) images
Matching with Features

• Detect feature points in both images
Matching with Features

- Detect feature points in both images
- Find corresponding pairs
Matching with Features

• Detect feature points in both images
• Find corresponding pairs
• Use these matching pairs to align images - the required mapping is called a homography
Automatic mosaicing

http://www.cs.ubc.ca/~mbrown/autostitch/autostitch.html
Recognition of specific objects, scenes

Schmid and Mohr 1997

Sivic and Zisserman, 2003

Rothganger et al. 2003

Lowe 2002

Kristen Grauman
When does SIFT fail?

Patches SIFT thought were the same but aren’t:
Other methods: Daisy

Circular gradient binning

SIFT

Daisy

Picking the best DAISY, S. Winder, G. Hua, M. Brown, CVPR 09
Other methods: SURF

For computational efficiency only compute gradient histogram with 4 bins:

Fig. 3. The descriptor entries of a sub-region represent the nature of the underlying intensity pattern. Left: In case of a homogeneous region, all values are relatively low. Middle: In presence of frequencies in $x$ direction, the value of $\sum |d_x|$ is high, but all others remain low. If the intensity is gradually increasing in $x$ direction, both values $\sum d_x$ and $\sum |d_x|$ are high.

SURF: Speeded Up Robust Features
Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, ECCV 2006
Other methods: BRIEF

Randomly sample pair of pixels a and b. 1 if a > b, else 0. Store binary vector.

Fig. 2. Different approaches to choosing the test locations. All except the rightmost one are selected by random sampling. Showing 128 tests in every image.

BRIEF: binary robust independent elementary features, Calonder, V Lepetit, C Strecha, ECCV 2010
Feature distance

How to define the difference between two features $f_1$, $f_2$?

- Simple approach is $\text{SSD}(f_1, f_2)$
  - sum of square differences between entries of the two descriptors
  - can give good scores to very ambiguous (bad) matches
Feature distance

How to define the difference between two features $f_1$, $f_2$?

- Better approach: ratio distance = $\frac{\text{SSD}(f_1, f_2)}{\text{SSD}(f_1, f_2')}$
  - $f_2$ is best SSD match to $f_1$ in $I_2$
  - $f_2'$ is 2nd best SSD match to $f_1$ in $I_2$
  - gives large values (~1) for ambiguous matches
Eliminating bad matches

Throw out features with distance > threshold
  • How to choose the threshold?
True/false positives

The distance threshold affects performance

- True positives = # of detected matches that are correct
  - Suppose we want to maximize these—how to choose threshold?
- False positives = # of detected matches that are incorrect
  - Suppose we want to minimize these—how to choose threshold?
Local Descriptors: Shape Context

Count the number of points inside each bin, e.g.:

Count = 4

Count = 10

Log-polar binning: more precision for nearby points, more flexibility for farther points.

Belongie & Malik, ICCV 2001