Announcements

• PS1 grades up on SmartSite
• PS1 stats:
  – Mean: 83.26
  – Standard Dev: 28.51

• PS2 deadline extended to *Saturday, 11:59 am*
due to SmartSite maintenance
Last time

- Image warping based on homography
- Detecting corner-like points in an image

Today

- Local invariant features
  - Detection of interest points
    - (Harris corner detection)
    - Scale invariant blob detection: LoG
  - Description of local patches
    - SIFT: Histograms of oriented gradients

Local features: main components

1) Detection: Identify the interest points

2) Description: Extract vector feature descriptor surrounding each interest point.

3) Matching: Determine correspondence between descriptors in two views
Goal: interest operator repeatability

- We want to detect (at least some of) the same points in both images.
- Yet we have to be able to run the detection procedure independently per image.
- No chance to find true matches!

Goal: descriptor distinctiveness

- We want to be able to reliably determine which point goes with which.
- Must provide some invariance to geometric and photometric differences between the two views.

Local features: main components

1) Detection: Identify the interest points
2) Description: Extract vector feature descriptor surrounding each interest point.
3) Matching: Determine correspondence between descriptors in two views
Recall: Corners as distinctive interest points

\[ M = \sum w(x,y) \begin{bmatrix} I_x I_x & I_x I_y \\ I_x I_y & I_y I_y \end{bmatrix} \]

2 x 2 matrix of image derivatives (averaged in neighborhood of a point).

Notation:
\[ I_x \Leftrightarrow \frac{\partial I}{\partial x}, \quad I_y \Leftrightarrow \frac{\partial I}{\partial y}, \quad I_x I_y \Leftrightarrow \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \]

Recall: Corners as distinctive interest points

Since \( M \) is symmetric, we have
\[ M = \lambda \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \]

(Eigenvalue decomposition)

Notation:
\[ I_x \Leftrightarrow \frac{\partial I}{\partial x}, \quad I_y \Leftrightarrow \frac{\partial I}{\partial y}, \quad I_x I_y \Leftrightarrow \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \]

The eigenvalues of \( M \) reveal the amount of intensity change in the two principal orthogonal gradient directions in the window.

Recall: Corners as distinctive interest points

"edge": \( \lambda_1 \gg \lambda_2 \) \( \lambda_2 \gg \lambda_1 \)

"corner": \( \lambda_1 \) and \( \lambda_2 \) are large, \( \lambda_1 \approx \lambda_2 \)

"flat" region \( \lambda_1 \) and \( \lambda_2 \) are small.

One way to score the cornerness:
\[ f = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2} \]
Harris corner detector

1) Compute $M$ matrix for image window surrounding each pixel to get its cornerness score.
2) Find points with large corner response ($f >$ threshold)
3) Take the points of local maxima, i.e., perform non-maximum suppression

Harris Detector: Steps

Compute corner response $f$
Harris Detector: Steps

1. Find points with large corner response: $f > \text{threshold}$

2. Take only the points of local maxima of $f$
Properties of the Harris corner detector

Rotation invariant? Yes
Translation invariant? Yes
Scale invariant? No

All points will be classified as edges

Corner!
Scale invariant interest points

How can we independently select interest points in each image, such that the detections are repeatable across different scales?

Automatic scale selection

Intuition:
• Find scale that gives local maxima of some function $f$ in both position and scale.

What can be the “signature” function?
Recall: Edge detection

**Source:** S. Seitz

**Recall: Edge detection**

From edges to blobs

- Edge = ripple
- Blob = superposition of two ripples

**Spatial selection:** the magnitude of the Laplacian response will achieve a maximum at the center of the blob, provided the scale of the Laplacian is "matched" to the scale of the blob

**Source:** S. Seitz

Slide credit: Lana Lazebnik
Blob detection in 2D

Laplacian of Gaussian: Circularly symmetric operator for blob detection in 2D

\[ \nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \]

Blob detection in 2D: scale selection

Laplacian-of-Gaussian = "blob" detector

\[ \nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \]

Blob detection in 2D

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

characteristic scale
Scale invariant interest points

Interest points are local maxima in both position and scale.

Squared filter response maps

⇒ List of \((x, y, \sigma)\)

Scale-space blob detector: Example
Technical detail

We can approximate the Laplacian with a difference of Gaussians; more efficient to implement.

\[ L = \sigma^2 (G_{x,y} + G_{x,y}) \]

(Laplacian)

\[ \text{DoG} = G(x,y,k\sigma) - G(x,y,\sigma) \]

(Difference of Gaussians)

Local features: main components

1) Detection: Identify the interest points

2) Description: Extract vector feature descriptor surrounding each interest point.

3) Matching: Determine correspondence between descriptors in two views

Geometric transformations

e.g. scale, translation, rotation
Photometric transformations

Figure from T. Tuytelaars ECCV 2006 tutorial

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.

SIFT descriptor [Lowe 2004]

- Use histograms to bin pixels within sub-patches according to their orientation.

Why subpatches?
Why does SIFT have some illumination invariance?
Making descriptor rotation invariant

- Rotate patch according to its dominant gradient orientation
- This puts the patches into a canonical orientation

Image from Matthew Brown

SIFT descriptor [Lowe 2004]

- Extraordinarily robust matching technique
  - Can handle changes in viewpoint
    - Up to about 60 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
Example

[Image: NASA Mars Rover images with SIFT feature matches. Figure by Noah Snavely]

SIFT descriptor properties

• Invariant to
  – Scale
  – Rotation

• Partially invariant to
  – Illumination changes
  – Camera viewpoint
  – Occlusion, clutter

Local features: main components

1) Detection: Identify the interest points

2) Description: Extract vector feature descriptor surrounding each interest point.

3) Matching: Determine correspondence between descriptors in two views
Matching local features

To generate **candidate matches**, find patches that have the most similar appearance (e.g., lowest SSD).
Simplest approach: compare them all, take the closest (or closest k, or within a thresholded distance).

Ambiguous matches

To add robustness to matching, can consider ratio:
distance to best match / distance to second best match
If low, first match looks good.
If high, could be ambiguous match.
Matching SIFT Descriptors

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

Recap: robust feature-based alignment

- Extract features
Recap: robust feature-based alignment

• Extract features
• Compute putative matches

Source: L. Lazebnik

Recap: robust feature-based alignment

• Extract features
• Compute putative matches
• Loop:
  • Hypothesize transformation $T$ (small group of putative matches that are related by $T$)

Source: L. Lazebnik

Recap: robust feature-based alignment

• Extract features
• Compute putative matches
• Loop:
  • Hypothesize transformation $T$ (small group of putative matches that are related by $T$)
  • Verify transformation (search for other matches consistent with $T$)

Source: L. Lazebnik
Recap: robust feature-based alignment

- Extract features
- Compute putative matches
- Loop:
  - Hypothesize transformation $T$ (small group of putative matches that are related by $T$)
  - Verify transformation (search for other matches consistent with $T$)

Source: L. Lazebnik

Applications of local invariant features

- Wide baseline stereo
- Motion tracking
- Panoramas
- Mobile robot navigation
- 3D reconstruction
- Recognition
- …

Slide credit: Kristen Grauman

Automatic mosaicing

http://www.cs.ubc.ca/~mbrown/autostitch/autostitch.html

Slide credit: Kristen Grauman
Wide baseline stereo

Recognition of specific objects, scenes

Summary

• Interest point detection
  – Harris corner detector
  – Laplacian of Gaussian, automatic scale selection

• Invariant descriptors
  – Rotation according to dominant gradient direction
  – Histograms for robustness to small shifts and translations (SIFT descriptor)
Questions?

See you Thursday!