Today

- Some more Pset 2 results
- Pset 2 returned, pick up solutions
- Pset 3 is posted, due 11/11

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

Sung Ju Hwang

Austin McGookay

Jose Luis Dominguez
To compute the homography, we needed pairs of corresponding points in the images.
Local features and alignment

- Detect feature points in both images
- Find corresponding pairs
- Use these pairs to align images

**Problem 1:**
- Detect the *same* point *independently* in both images

*We need a repeatable detector*

**Problem 2:**
- For each point correctly recognize the corresponding one

*We need a reliable and distinctive descriptor*
Similarly, the first step in our stereo pipeline using weak calibration was to find interest points,...

... and we let the surrounding pixels in a neighborhood patch serve as the local descriptor, which we can compare with correlation.

We want a sparse set of reliably detectable interest points.

Local features and stereo matching

Putative matches

Putative matches

• Patches of intensity have limited robustness for matching across different views
• Consider the case where we have a wide baseline separating the two views


• What would we like our local features to be invariant to?
Geometric transformations

And other nuisances…
• Noise
• Blur
• Compression artifacts
• Appearance variation for a category

Invariant local features
Subset of local feature types designed to be invariant to common geometric and photometric transformations.
Basic steps:
1) Detect distinctive interest points
2) Extract invariant descriptors

Main questions
• Where will the interest points come from?
  – What are salient features that we’ll detect in multiple views?
• How to describe a local region?
• How to establish correspondences, i.e., compute matches?

Finding Corners
Key property: in the region around a corner, image gradient has two or more dominant directions
Corners are repeatable and distinctive

Source: Lana Lazebnik
Corners as distinctive interest points

We should easily recognize the point by looking through a small window. Shifting a window in any direction should give a large change in intensity.

“flat” region: no change in all directions
“edge”: no change along the edge direction
“corner”: significant change in all directions

Harris Detector formulation

This measure of change can be approximated by:

\[ E(u, v) \approx [u \ v] M [u \ v]^T \]

where \( M \) is a 2x2 matrix computed from image derivatives:

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

Sum over image region – area we are checking for corner

\[ \cdot M = \begin{bmatrix} \sum I_x I_x \sum I_x I_y \\ \sum I_x I_y \sum I_y I_y \end{bmatrix} = \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \begin{bmatrix} I_x \\ I_y \end{bmatrix} \]

What does this matrix reveal?

First, consider an axis-aligned corner:

This means dominant gradient directions align with x or y axis.

If either \( \lambda \) is close to 0, then this is not a corner, so look for locations where both are large.

What if we have a corner that is not aligned with the image axes?
General Case

Since $M$ is symmetric, we have $M = R \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R^T$.

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 $M$:

- **“Corner”** $\lambda_1$ and $\lambda_2$ are large, $\lambda_1 - \lambda_2$; $E$ increases in all directions
- **“Edge”** $\lambda_1 > \lambda_2$; $E$ is almost constant in all directions
- **“Flat”** region $\lambda_1$ and $\lambda_2$ are small; $E$ increases in all directions

Corner response function

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

$\alpha$: constant (0.04 to 0.06)

Harris Corner Detector

- **Algorithm steps:**
  - Compute $M$ matrix within all image windows to get their R scores
  - Find points with large corner response ($R >$ threshold)
  - Take the points of local maxima of $R$
Harris Detector: Workflow

Find points with large corner response: \( R > \) threshold

Harris Detector: Workflow

Take only the points of local maxima of \( R \)

Harris Detector: Workflow

1) Find interest points

Harris Detector: Properties

• Rotation invariance

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

Corner response \( R \) is invariant to image rotation

Harris Detector: Properties

• Not invariant to image scale

All points will be classified as edges

Corner!
• How can we detect *scale invariant* interest points?

Exhaustive search
A multi-scale approach

Exhaustive search
A multi-scale approach

Exhaustive search
A multi-scale approach

Automatic scale selection
We want to extract the patches from each image *independently.*
Automatic scale selection

- **Solution:**
  - Design a function on the region, which is "scale invariant" (the same for corresponding regions, even if they are at different scales)
    - Example: average intensity. For corresponding regions (even of different sizes) it will be the same.
  - For a point in one image, we can consider it as a function of region size (patch width)

\[ f \] Image 1
\[ f \] Image 2

\[ \text{region size} \]

- **Common approach:**
  - Take a local maximum of this function
  - Observation: region size, for which the maximum is achieved, should be invariant to image scale.

\[ s_1 \] Image 1
\[ s_2 \] Image 2

\[ \text{region size} \]

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

Automatic Scale Selection

- Function responses for increasing scale (scale signature)

\[ f(U_{(x',\sigma')}) = f(U_{(x',\sigma')}) \]

Same operator responses if the patch contains the same image up to scale factor.
Automatic Scale Selection

• Function responses for increasing scale (scale signature)

Scale selection

• Use the scale determined by detector to compute descriptor in a normalized frame

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

Laplaceian-of-Gaussian (LoG)

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

\[ L_\sigma(x,y) = L_\sigma(x,y) \]

\[ \Rightarrow \text{List of } (x,y,\sigma) \]

We can efficiently approximate the Laplacian with a difference of Gaussians:

\[ L = \sigma^2 (G_x(x,y,\sigma) + G_y(x,y,\sigma)) \]

(Lapacian)

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

(Difference of Gaussians)

This is used in Lowe’s SIFT (Scale Invariant Feature Transform) pipeline for keypoint detection.
Key point localization with DoG
- Detect maxima of difference-of-Gaussian (DoG) in scale space
- Then reject points with low contrast (threshold)
- Eliminate edge responses

Candidate keypoints: list of \((x, y, \sigma)\)

Example of keypoint detection

Scale Invariant Detection: Summary
- **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)

Main questions
- **Where** will the interest points come from?
  - What are salient features that we’ll detect in multiple views?
- **How** to describe a local region?
- **How** to establish *correspondences*, i.e., compute matches?

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

Point descriptor should be:
1. Invariant
2. Distinctive

Local descriptors
- Simplest descriptor: list of intensities within a patch.
- What is this going to be invariant to?
Feature descriptors

Disadvantage of patches as descriptors:

- Small shifts can affect matching score a lot

Solution: histograms

Feature descriptors: SIFT

Scale Invariant Feature Transform

Descriptor computation:

- Divide patch into 4x4 sub-patches: 16 cells
- Compute histogram of gradient orientations (8 reference angles) for all pixels inside each sub-patch
- Resulting descriptor: 4x4x8 = 128 dimensions

Rotation Invariant Descriptors

- Harris corner response measure: depends only on the eigenvalues of the matrix $M$

Rotation Invariant Descriptors

- Find local orientation
  Dominant direction of gradient for the image patch

- Rotate patch according to this angle
  This puts the patches into a canonical orientation.

Rotation Invariant Descriptors

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

Feature descriptors: SIFT

Stein Seitz
Working with SIFT descriptors

- One image yields:
  - \(n\) 128-dimensional descriptors: each one is a histogram of the gradient orientations within a patch
    - \([n \times 128\) matrix]
  - \(n\) scale parameters specifying the size of each patch
    - \([n \times 1\) vector]
  - \(n\) orientation parameters specifying the angle of the patch
    - \([n \times 1\) vector]
  - \(n\) 2d points giving positions of the patches
    - \([n \times 2\) matrix]

Main questions

- Where will the interest points come from?
  - What are salient features that we’ll detect in multiple views?
- How to describe a local region?
- How to establish correspondences, i.e., compute matches?

Summary

- Interest point detection
  - Harris corner detector
  - Laplacian of Gaussian: scale selection
- Invariant descriptors
  - Rotation according to dominant gradient direction
  - Histograms for robustness to small shifts and translations
    - SIFT

Next

- Recognition & image retrieval
- Thursday:
  - Bag of words models and inverted file indexing for images
  - Read
    - Video Google paper by Sivic & Zisserman
    - Excerpt on vector models posted on Blackboard
- Pset 3 is posted, due Tuesday 11/11.