Scale Invariant Feature Transform

Why do we care about matching features?

- Camera calibration
- Stereo
- Tracking/SFM
- Image mosaicing
- Object/activity Recognition
- ...

Objection representation and recognition

- Image content is transformed into local feature coordinates that are invariant to translation, rotation, scale, and other imaging parameters

Automatic Mosaicing

We want invariance!!!

- To illumination
- To scale
- To rotation
- To affine
- To perspective projection

Types of invariance

- Illumination

Types of invariance

- Illumination
- Scale
- Rotation
Types of invariance

- Illumination
- Scale
- Rotation
- Affine

How to achieve illumination invariance

- The easy way (normalized)
- Difference based metrics (random tree, Haar, and sift, gradient)

Types of invariance

- Illumination
- Scale
- Rotation
- Affine
- Full Perspective

How to achieve scale invariance

- Pyramids
  - Divide width and height by 2
  - Take average of 4 pixels for each pixel (or Gaussian blur with different $\sigma$)
  - Repeat until image is tiny
  - Run filter over each size image and hope its robust
- Scale Space (DOG method)
Pyramids

How to achieve scale invariance

• Scale Space: Difference of Gaussian (DOG)
  – Take features from differences of these images—producing the gradient image
  – If the feature is repeatedly present in between Difference of Gaussians, it is Scale Invariant and should be kept.

Differences Of Gaussians

Rotation Invariance

• Rotate all features to go the same way in a determined manner
• Take histogram of Gradient directions. Rotate to most dominant (maybe second if its good enough, sub-Bin accuracy)
Rotation Invariance

Affine Invariance

- Easy way: Warp your training and hope
- Fancy way: design your feature itself to be robust against affine transformations (SIFT method)

SIFT algorithm overview

- Scale-space extrema detection
  - Get tons of points from maxima+minima of DOGS
- Keypoint localization
  - Threshold on simple contrast (low contrast is generally less reliable than high for feature points)
  - Threshold based on principal curvatures
  - Orientation assignment
- Keypoint descriptor
  - Construct histograms of gradients (HOG)

Scale-space extrema detection

- Find the points, whose surrounding patches (with some scale) are distinctive
- An approximation to the scale-normalized Difference of Gaussian

\[
L(x, y, \sigma) = G(x, y, \sigma) \ast I(x, y)
\]

\[
G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}
\]

\[
D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) \ast I(x, y)
\]

\[
= L(x, y, k\sigma) - L(x, y, \sigma).
\]
**Eliminating edge points**

- Such a point has large principal curvature across the edge but a small one in the perpendicular direction.
- The principal curvatures can be calculated from a Hessian function or covariance matrix of gradient (Harris detector):
  \[
  H = \begin{bmatrix}
    D_{xx} & D_{xy} \\
    D_{xy} & D_{yy}
  \end{bmatrix}
  \]
  \[
  C = \begin{bmatrix}
    \Sigma L_r & \Sigma L_t \\
    \Sigma L_t & \Sigma L_r
  \end{bmatrix}
  \]
- The eigenvalues of \( H \) or \( C \) are proportional to the principal curvatures, so two eigenvalues shouldn’t diff too much.

**Finding Keypoints – Orientation**

- Create histogram of local gradient directions computed at selected scale.
- Assign canonical orientation at peak of smoothed histogram, achieving invariance to image rotation.
- Each key point specifies stable 2D coordinates \((x, y, \text{scale}, \text{orientation})\).
How to use these features?

- Distance could be L2 norm on histograms
- Match by \((\text{nearest neighbor distance})/(2^{\text{nd}} \text{ nearest neighbor distance})\) ratio
### Application: object recognition

- The SIFT features of training images are extracted and stored
- For a query image
  1. Extract SIFT feature
  2. Nearest neighbor matching

### Conclusion

- A novel method for detecting interest points. The most successful feature in computer vision
- Histogram of Oriented Gradients are becoming more popular
- SIFT may not be optimal for general object classification