Scale-Invariant Feature Transform

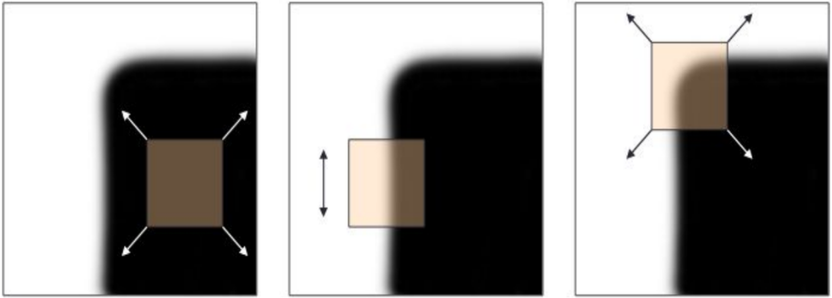
Dr. V Masilamani

masila@iiitdm.ac.in

Department of Computer Science and Engineering
IIITDM Kancheepuram
Chennai-127

Overview

- In the region around a corner, image gradient has two or more dominant directions.



Recap: Corner Detection: Basic Idea (cont.)

Change in appearance for the shift [u,v]

$$E(u,v) = \sum_{x,y} w(x,y)\ [\ I(x+u, y+v)\ -\ I(x,y)\ ]^2$$

Window function - $w(x,y)$

Shifted Intensity - $I(x+u, y+v)$

Intensity - $I(x,y)$

We're looking for windows that produce a large E value.

From taylor series we get,

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} \begin{bmatrix} \sum_{x,y} 2w(x,y)I_x^2(x,y) & \sum_{x,y} w(x,y)I_x(x,y)I_y(x,y) \\ \sum_{x,y} w(x,y)I_x(x,y)I_y(x,y) & \sum_{x,y} 2w(x,y)I_y^2(x,y) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

The quadratic expression simplifies to

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix}$$

Where M is the second moment matrix, given computed by image derivatives

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

Consider the axis aligned case where gradients are either horizontal or vertical

$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = Q^T A Q \approx \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}, where\ A = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$

- Sub M in E(u,v), we get

$$E(u,v) \approx \begin{bmatrix} u & v \end{bmatrix} Q^T A Q \begin{bmatrix} u \\ v \end{bmatrix}$$

-

$$E(u,v) \approx (Q \begin{bmatrix} u & v \end{bmatrix}^T)^T A Q \begin{bmatrix} u \\ v \end{bmatrix}$$

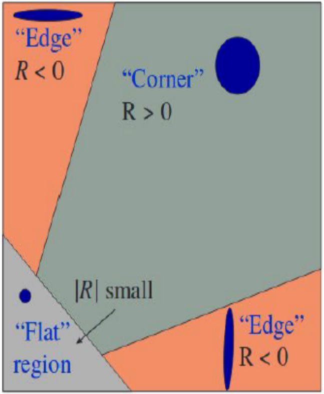
Interpretation of $Q(u, v)^T$

- $(u, v)^T$ is transformed into a new coordinate system with eigen vectors as axes, say $(u', v')^T$
- Hence

$$E(u, v) \approx (\begin{bmatrix} u' & v' \end{bmatrix} A \begin{bmatrix} u' \\ v' \end{bmatrix}$$

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

- R is large for a corner
- R is negative with large magnitude for an edge
- $|R|$ is small for a flat region

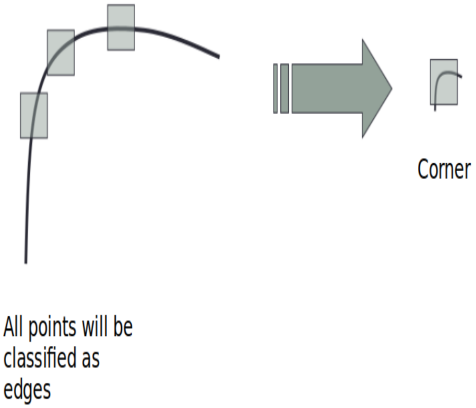


Properties of Harris corner points

- Harris corner points are rotational invariant
  - In $E(u, v)$, (u', v') provides direction information, and A is providing the magnitude
  - Let f be an image and f_r be a rotated image
  - The diagonal matrix A for both f and f_r (for a given point (x,y)) will be the same as rotation changes only the direction, not the magnitude

Properties of Harris corner points (cont.)

- Not invariant to image scale



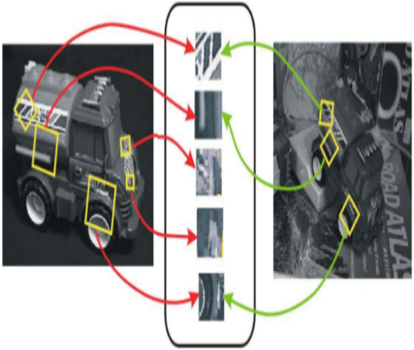
Scale Invariant Feature Transform(SIFT)

Goal: Find features of image that are

- Invariant to image scale and rotation
- Robust to
  - Distortion,
  - Change in 3D viewpoint,
  - Addition of noise,
  - Change in illumination.

Idea of SIFT

- Find Feature points(locations in image) that are invariant to scaling and rotation and robust to other changes
- Find a descriptor for each feature point, considering patch around the point



SIFT Features

Overall Procedure of SIFT

- Scale-space extrema detection
  - Search over multiple scales and image locations
- Keypoint localization
  - Select keypoints based on a measure of stability.
- Orientation assignment
  - Compute best orientation(s) for each keypoint region.
- Keypoint description
  - Use local image gradients at selected scale and rotation to describe each keypoint region.

Find LoG for each image which is equivalent to find difference of gaussians(DoG) for two different blurred image (Computationally effective)

Laplacian

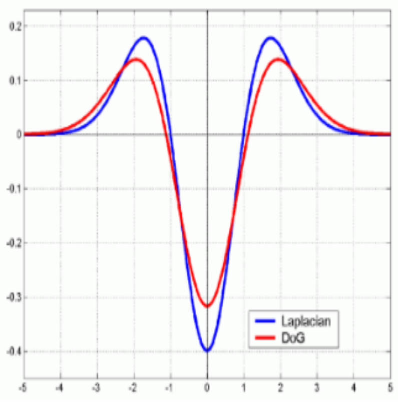
$$L = \sigma^2 (G_{xx}(x, y, \sigma) + G_{yy}(x, y, \sigma))$$

Difference of Gaussians

$$DOG = G(x, y, k\sigma) - G(x, y, \sigma)$$

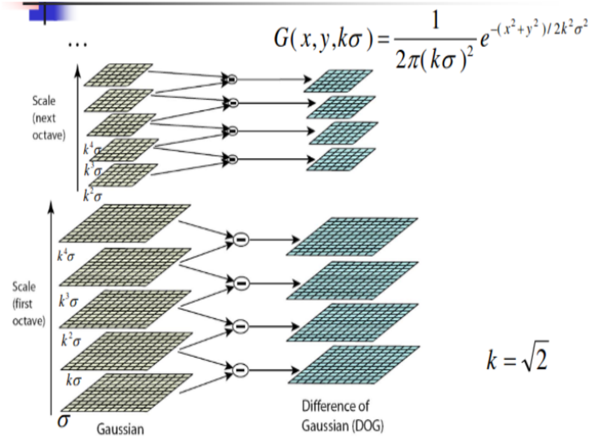
where

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



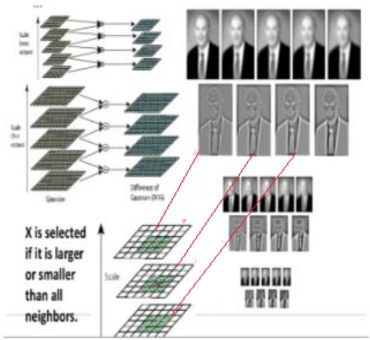
Note: LoG is invariant to scale and rotation

Efficient DoG Computation using Gaussian Scale Pyramid



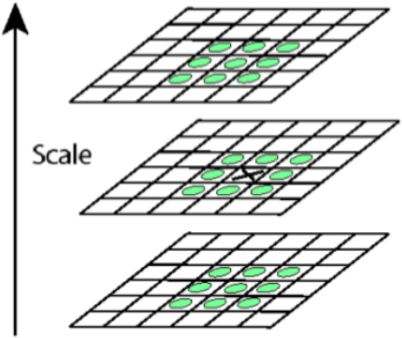
- Scale: Standard Deviation used in Gaussian filer
- Octave: Set of Images with the same resolution

DOG detector : Flowchart



Local Extrema in DoG Images

- Minima
- Maxima
- 26 neighbours for a candidate key point
- A point is an extreme Point if it is less than or equal to all 26 neighbours or grater than or equal to all 26 neighbours



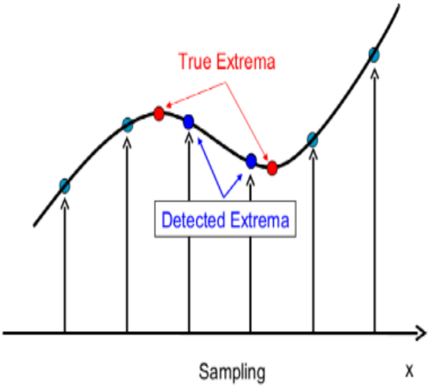
Key Point Localization

Candidates are chosen from extrema detection

original image

extrema locations

- Poorly localized candidates along an edge can be removed
  - Use Taylor series expansion of DOG
  - Find min or max points in DOG



$$D(X) = D(0) + \frac{\partial D(0)}{\partial X} X + \frac{1}{2} X^T \frac{\partial^2 D(0)}{\partial X^2} X$$

To maximize $D(X)$, set $\frac{\partial D(X)}{\partial X} = 0$

$$\frac{\partial D(X)}{\partial X} = 0 + \frac{\partial D(0)}{\partial X} + \frac{c}{2} \frac{\partial}{\partial X} (X^T X)$$

where $c = \frac{\partial^2 D(0)}{\partial X^2}$

$$\frac{\partial D(X)}{\partial X} = \frac{\partial D(0)}{\partial X} + \frac{c}{2} \frac{\partial}{\partial X} \|X\|^2$$

Initial Outlier Rejection (cont.)

$$\frac{\partial D(X)}{\partial X} = \frac{\partial D(0)}{\partial X} + cX$$

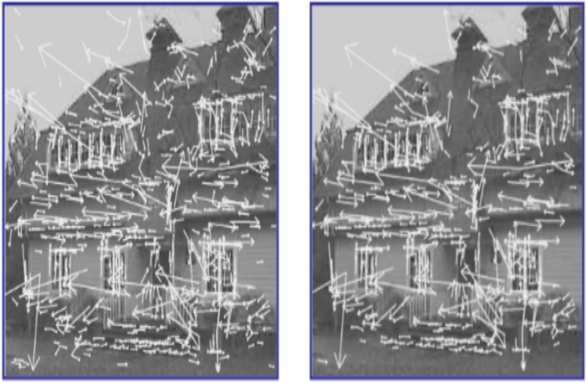
By setting, $\frac{\partial D(X)}{\partial X} = 0$

$$X = \frac{1}{c}\left(-\frac{\partial D(0)}{\partial X}\right)$$

Substitute c in above equation

$$X = -\left(\frac{\partial^2 D(0)}{\partial X^2}\right)^{-1} \frac{\partial D(0)}{\partial X}$$

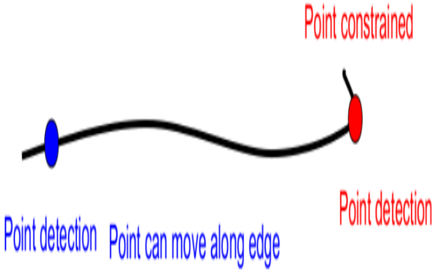
- Minima or maxima is located at X
- Value of $D(X)$ at minima/maxima must be large, $|D(X)| > th$
- Reject x as key point if $|D(X)| < th$



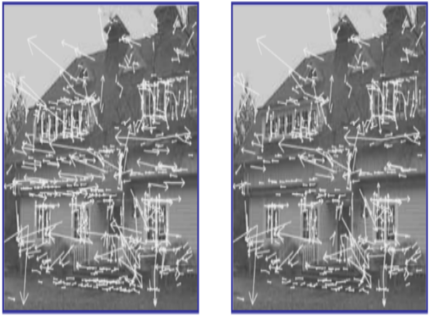
from 832 key points to 729 key points, th=0.03.

Further Outlier Rejection

- Reject points with strong edge response in one direction only
- Use Harris - using Trace and Determinant of Hessian



Further Outlier Rejection (cont.)



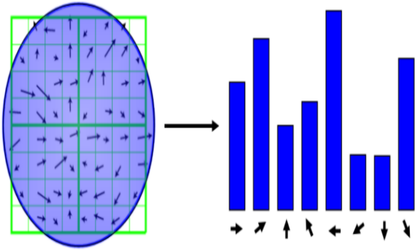
from 729 key points to 536 key points.

Orientation Assignment

- Aim : Assign constant orientation to each keypoint based on local image property to obtain rotational invariance.

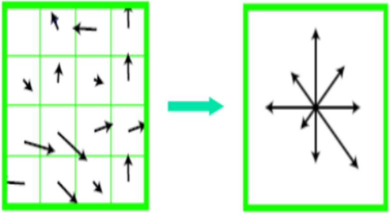
Orientation Assignment (cont.)

- Create a weighted direction histogram in a neighborhood of a key point (36 bins)
- To assign weights, use Gaussian kernel



Orientation Assignment (cont.)

- Select the peak direction as direction of the key point
- Keep all directions with 80% of max peak of the histogram



Keypoint Descriptors

- At this point, each keypoint has
  - Location
  - Scale
  - Orientation
- Next is to compute a descriptor for the local image region about each keypoint that is
  - highly distinctive
  - invariant as possible to variations such as changes in viewpoint and illumination

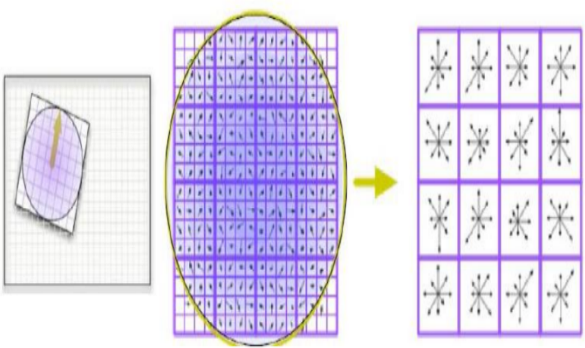
Keypoint Descriptors (cont.)

- Rotate the window to standard orientation
- Scale the window size based on the scale at which the point was found.
- Compute relative orientation and magnitude in a 16x16 neighborhood at key point
- Form weighted histogram (8 bin) for 4x4 regions
  - Weight by magnitude and Gaussian
  - Concatenate 16 histograms in one long vector of 128 dimensions

Dimension of keypoint Descriptor

[allowframebreak]

- 4x4 array of gradient orientation histograms over 4x4 pixels
- 8 orientations x 4x4 array = 128 dimensions
- 128-dim vector normalized to unit length to reduce the effect of illumination



Why is SIFT keypoint descriptor invariant to scale and rotation

- **Scale Invariant:**
  - Suppose the key point is found at (x, y) at scale s and octave o, the descriptor is computed after resizing the window in the octave to a standard size
  - Hence, when test image and its corresponding data base image are in different sizes, their corresponding descriptors will match
- **Rotation Invariant:**
  - The peak of weighed directional histogram for a key point is aligned to a standard direction by rotating the window centered at the key point
  - Hence, if the test image is a rotated version of its corresponding database image, then the descriptors of the corresponding key points will match

Why is SIFT Key point descriptor robust to distortion

- Since the difference between the first and the second peak is atleast 20 %, the the peaks for the windows of the corresponding key points in distorted and original images will be the same
- Hence their descriptors will match

Key point matching

- Match the key points against a database of that obtained from training images.
- Find the nearest neighbor i.e. a key point with minimum Euclidean distance
- An improved Nearest Neighbor matching
  - Looks at ratio of distance between best and 2nd best match (.8)