Feature detection and description (Week 06: Lectures 24-28)

Jayanta Mukhopadhyay
Dept. of Computer Science and Engg.

Can you match the scene points?

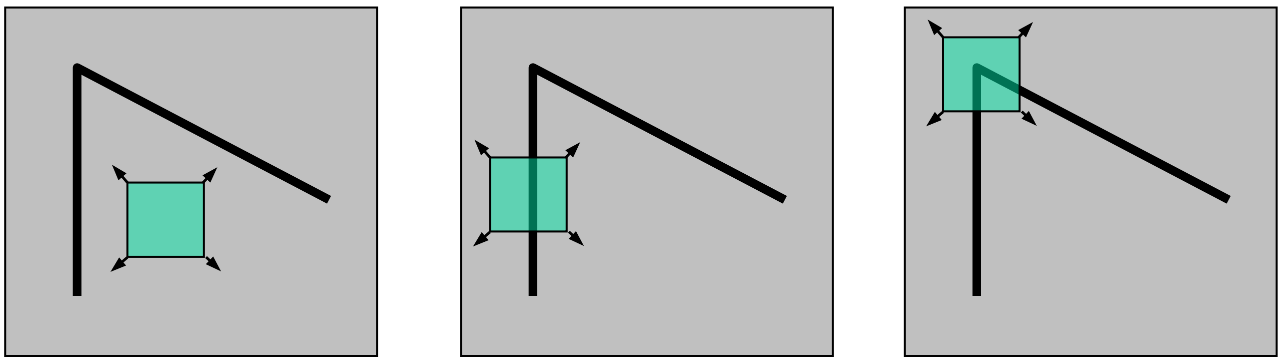
- Detection
- Description
- Matching

Translation, rotation, scale ..

Feature detection

A local measure for uniquely identifying the feature.

- How does the measure change on shifting windows at different points?



"flat" region: no change in all directions

"edge": no change along the edge direction

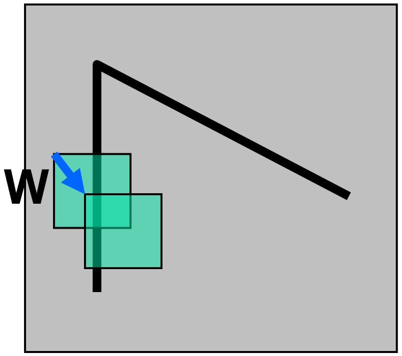
"corner": significant change in all directions

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Feature detection

Consider shifting the window **W** by (u,v).

- how do the pixels in **W** change?
- compare each pixel before and after by summing up the squared differences (SSD).
- this defines an SSD "error" of $E(u,v)$.



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

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Small motion assumption

$$I(x+u, y+v) = I(x,y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \text{higher order terms}$$

For small u and v

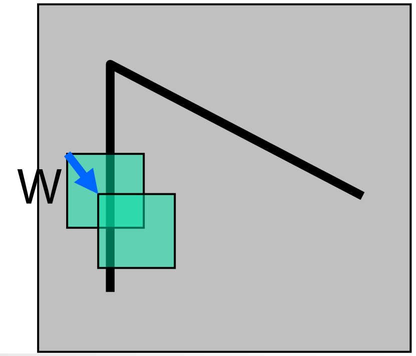
$$I(x+u, y+v) \approx I(x,y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v$$

$$\approx I(x,y) + [I_x \ I_y] \begin{bmatrix} u \\ v \end{bmatrix}$$

$$\text{shorthand: } I_x = \frac{\partial I}{\partial x}$$

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Feature detection



$$I(x+u, y+v) \approx I(x,y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v \approx I(x,y) + [I_x \ I_y]\begin{bmatrix} u \\ v \end{bmatrix}$$

$$\begin{aligned} E(u,v) &= \sum_{(x,y)\in W} [I(x+u,y+v) - I(x,y)]^2 \\ &\approx \sum_{(x,y)\in W} \left[I(x,y) + [I_x \ I_y]\begin{bmatrix} u \\ v \end{bmatrix} - I(x,y)\right]^2 \\ &\approx \sum_{(x,y)\in W} \left[[I_x \ I_y]\begin{bmatrix} u \\ v \end{bmatrix}\right]^2 \end{aligned}$$

$$\left(\begin{bmatrix} I_x & I_y \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix}\right)^T \left(\begin{bmatrix} I_x & I_y \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix}\right)$$

$$= \begin{bmatrix} u & v \end{bmatrix}\begin{bmatrix} I_x \\ I_y \end{bmatrix}\begin{bmatrix} I_x & I_y \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix}$$

$$= \begin{bmatrix} u & v \end{bmatrix}\begin{bmatrix} I_x^2 & I_xI_y \\ I_yI_x & I_y^2 \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix}$$

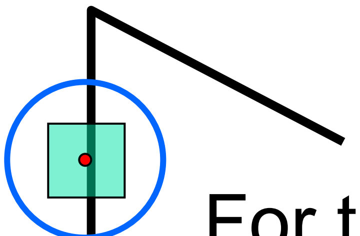
$$E(u,v) = \sum_{(x,y)\in W} [u \ v]\begin{bmatrix} I_x^2 & I_xI_y \\ I_yI_x & I_y^2 \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix}$$

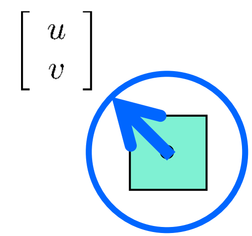
Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

$$E(u,v) = \sum_{(x,y)\in W} [u\ v] \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

Feature detection

$$E(u,v) = \sum_{(x,y)\in W} [u\ v] \underbrace{\begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix}}_{H} \begin{bmatrix} u \\ v \end{bmatrix}$$





For the example above

- Suppose the center of the green window moved to anywhere on the blue unit circle.
- Directions for the largest and the smallest E values?
  - Eigenvectors of H.

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Quick eigenvalue/eigenvector review

$$Ax = \lambda x$$

The **eigenvectors** of a matrix A are the vectors x that satisfy:

$$det(A - \lambda I) = 0 \quad det\begin{bmatrix} h_{11} - \lambda & h_{12} \\ h_{21} & h_{22} - \lambda \end{bmatrix} = 0$$

The scalar λ is the **eigenvalue** corresponding to x

- The eigenvalues are found by solving:

$$\lambda_{\pm} = \frac{1}{2}\left[(h_{11} + h_{22}) \pm \sqrt{4h_{12}h_{21} + (h_{11} - h_{22})^2}\right]$$

- For eigen vector solve the following:

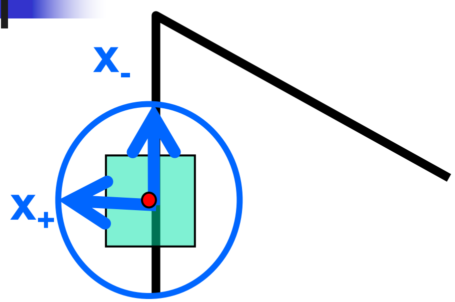
$$\begin{bmatrix} h_{11} - \lambda & h_{12} \\ h_{21} & h_{22} - \lambda \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = 0$$

Once you know λ, you find eigen vector $[x\ y]^T$ by solving the above.

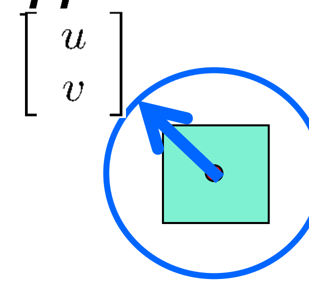
Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

$$E(u,v) = \sum_{(x,y)\in W} [u\ v] \underbrace{\begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix}}_{H} \begin{bmatrix} u \\ v \end{bmatrix}$$

Feature detection



$$\begin{aligned} Hx_+ &= \lambda_+ x_+ \\ Hx_- &= \lambda_- x_- \end{aligned}$$



Eigenvalues and eigenvectors of H

- Define shifts with the smallest and largest change (E value)
- x_+ = direction of **largest** increase in E.
- λ_+ = amount of increase in direction x_+
- x_- = direction of **smallest** increase in E.
- λ- = amount of increase in direction x_-

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

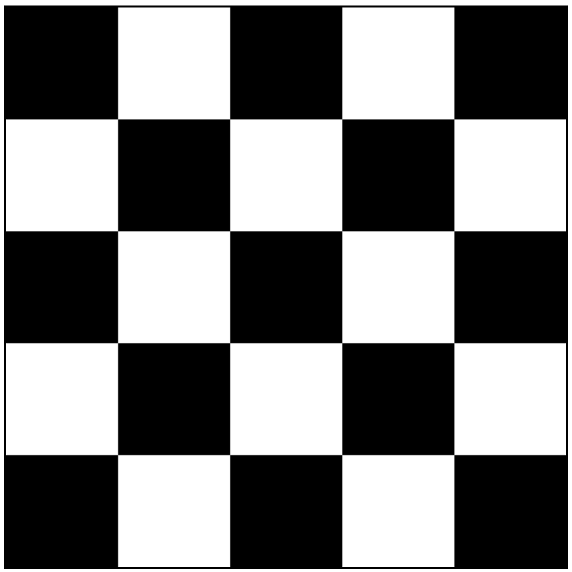
Feature scoring function

$E(u,v)$ to be ***large*** for small shifts in **all** directions

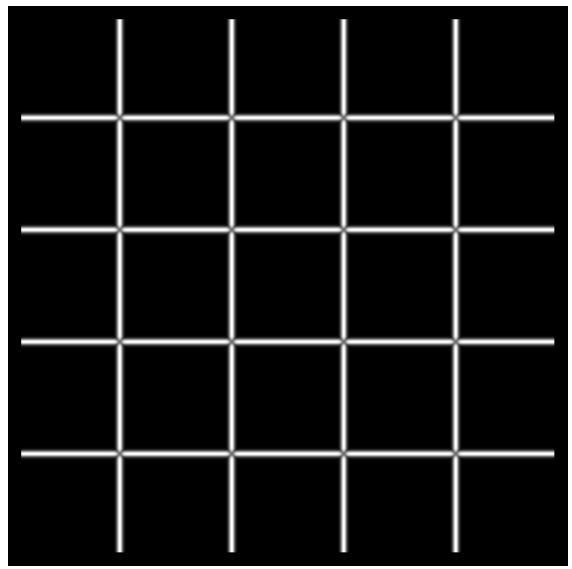
- the *minimum* of $E(u,v)$ should be large, over all unit vectors $[u\ v]$.
- this minimum is given by the smaller eigenvalue (λ_{-}) of H

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

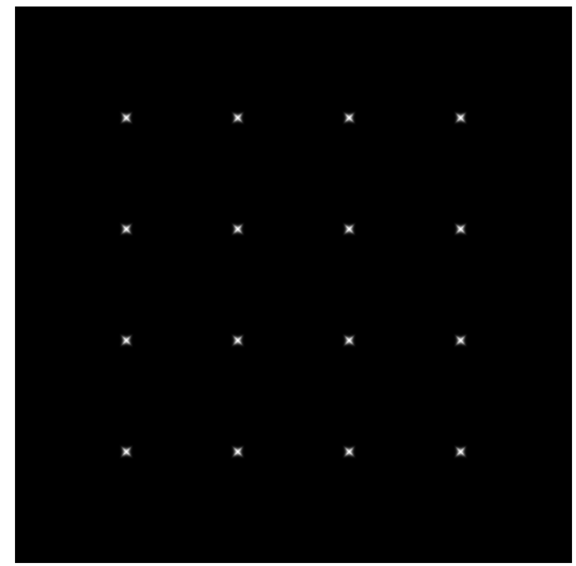
Feature detection



I



λ_+



λ_-

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Feature detection: Algorithm

- Compute the gradient at each point in the image.
- Obtain the H matrix from the entries in the gradient.

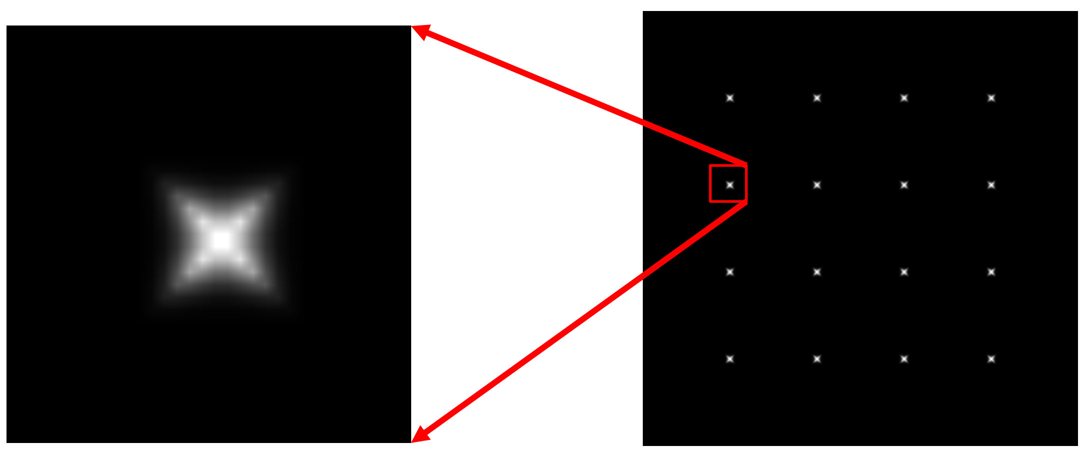
$$H = \begin{bmatrix} \overline{I_x^2} & \overline{I_x}\,\overline{I_y} \\ \overline{I_y}\,\overline{I_x} & \overline{I_y^2} \end{bmatrix}$$

- Compute eigenvalues of H.
- Locate points with large response (λ_- > threshold)
- Select points where λ_- is a local maximum as features.

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Local maximum: illustration

- Points with λ_- as a local maximum.



λ_-

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

The Harris operator

λ_- is a variant of the “Harris operator” for feature detection.

$$f = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2}$$

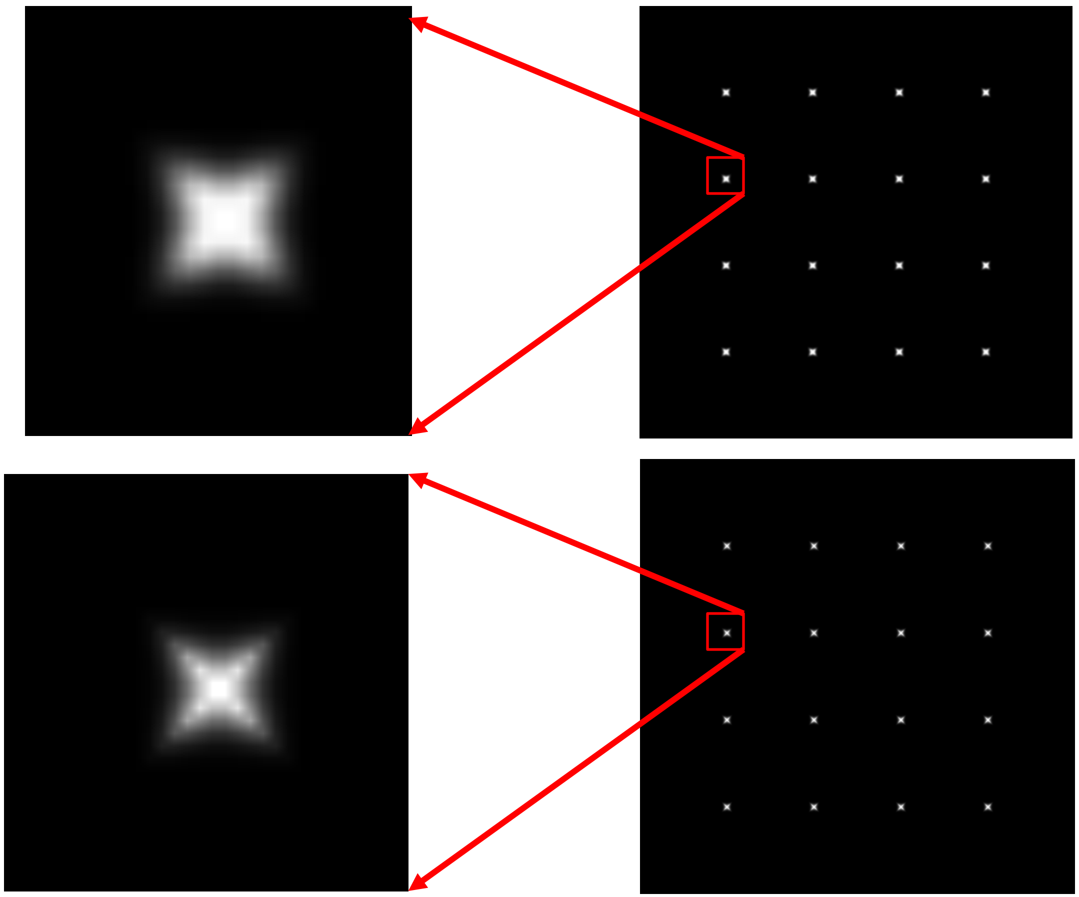
$$= \frac{determinant(H)}{trace(H)}$$

- The *trace* is the sum of the diagonals, i.e.,

  $trace(H) = h_{11} + h_{22}$
- Very similar to λ_- but less expensive (no square root)
- Called the “Harris Corner Detector” or “Harris Operator”
- Lots of other detectors, this is one of the most popular.

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

The Harris operator



Harris operator

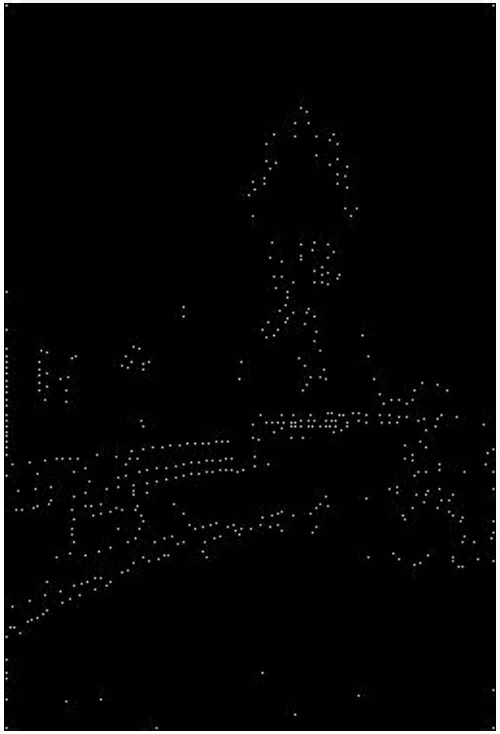
λ_-

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Harris detector example

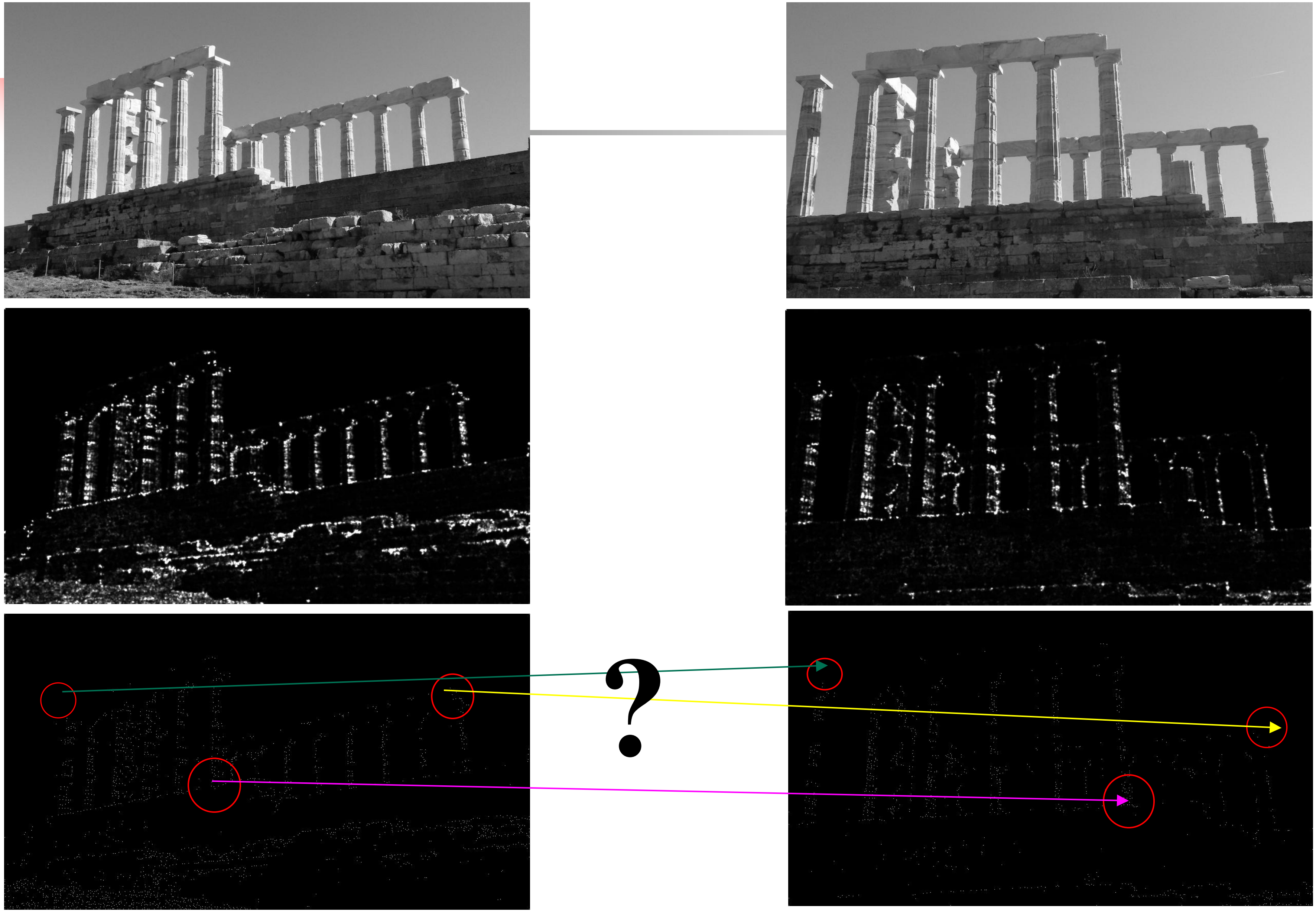
Input image

f-Value



Local maxima

Feature matching



Matching with Features

- Detect feature points in both images.
- Describe them by local statistics.
- Find corresponding pairs (Matching).

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Invariance

- Is it possible to select same features under various transformations?
  - Rotation
  - Change of illumination
  - Scale
  - ..

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Scale invariant detection

Key idea for detecting corners

Find scale that gives local maximum of f.

- f is a local maximum in both position and scale
- Common definition of f: Laplacian (or difference between two Gaussian filtered images with different s.d.'s).

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Invariance

Consider two images I_1 and I_2 so that I_2 is a transformed version of I_1

- For example,
  - Translation
  - Rotation
  - Scale
  - Reflection
  - Non-uniform scaling
  - Illumination
  - View …

Tranformational Invariance: Detection of the same features regardless of the transformation.

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Detection and description: to be invariant

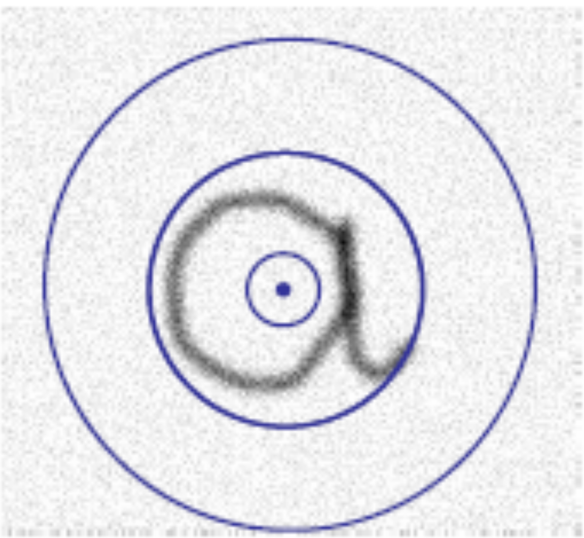
Both should be ensured.

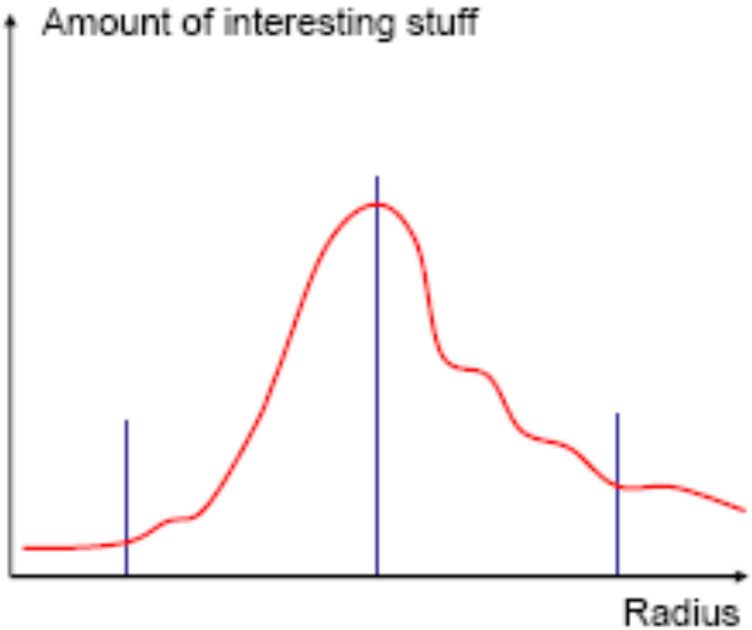
1. Detector to be transformational invariant.
   - Harris measure is invariant to translation and rotation.
   - Scale requires handling of multi-resolution representation
     - Usual approach: Computing features at multiple scales using a Gaussian pyramid.
     - More precise computation locates features at "the best scale" (e.g., SIFT)
2. Feature descriptor to be transformational invariant.
   - captures the information in a region around the detected feature point.
     - e.g. histogram of gradient directions in a square window centering a feature point.

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Finding Keypoints – Scale, Location

- How do we choose scale?





Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Scale Invariant Detection

- Functions for determining scale

Convolve with image and observe extrema in 3-D.

Kernels:

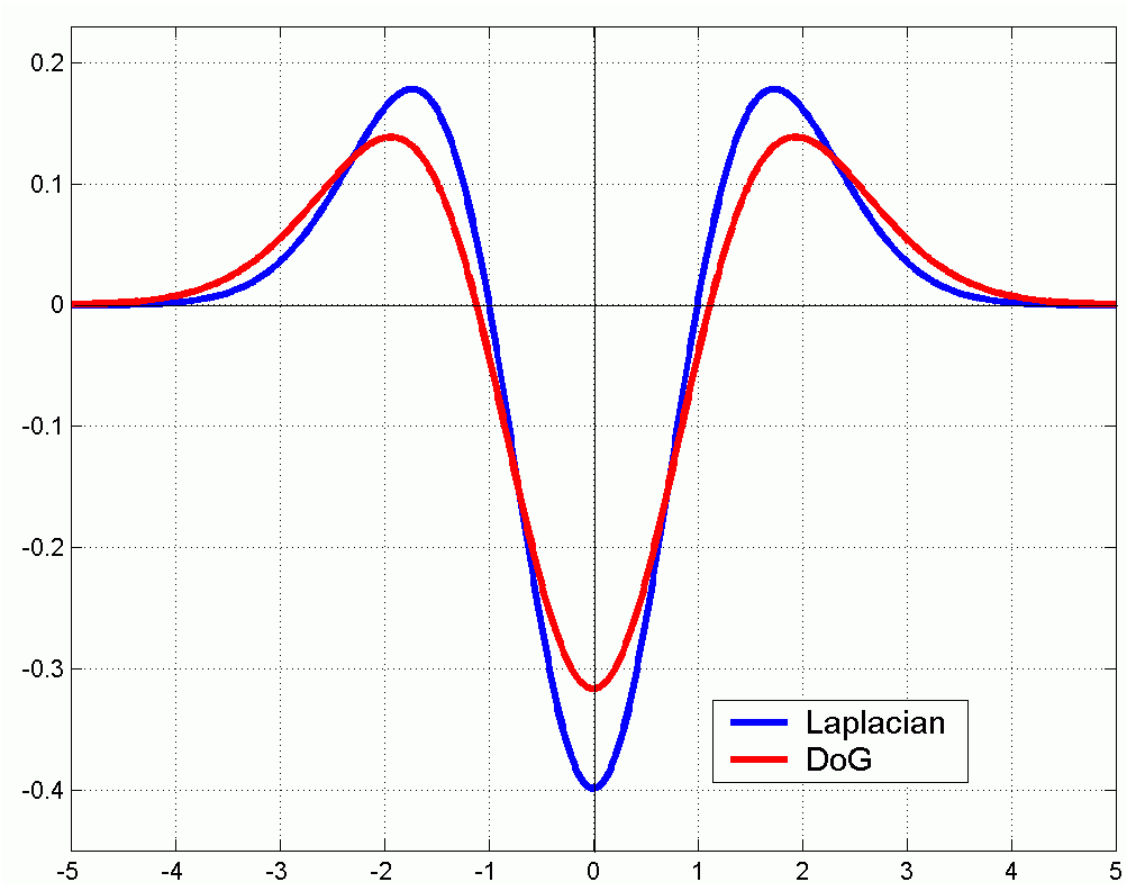
Laplacian $$L = \sigma^2\left(G_{xx}(x,y,\sigma) + G_{yy}(x,y,\sigma)\right)$$

Difference of Gaussians $$DoG = G(x,y,k\sigma) - G(x,y,\sigma)$$

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

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Kernel plots in 1-D



Both kernels are invariant to *scale* and *rotation.*

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Relationship between LoG and DoG operator

$$\frac{\partial G}{\partial \sigma} = \sigma \nabla^2 G$$

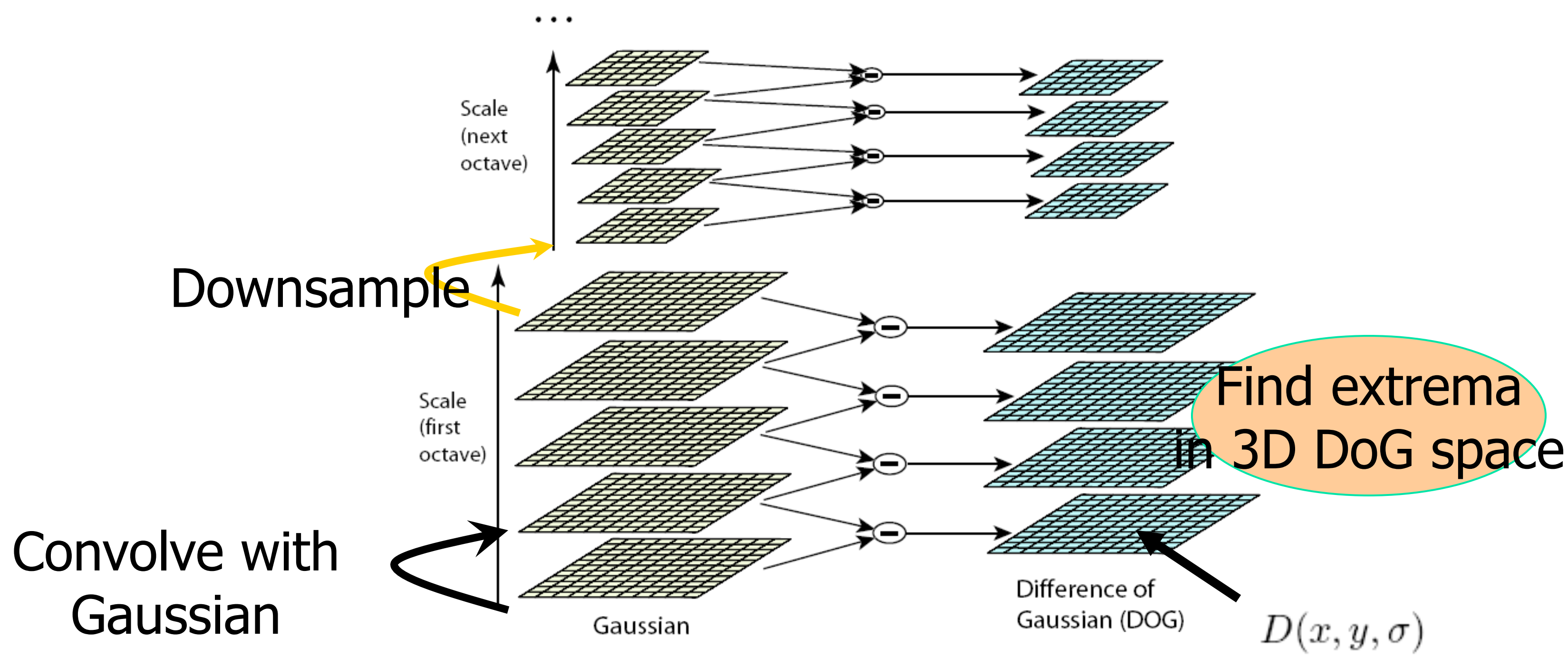
$$\sigma \nabla^2 G = \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma}$$

$$G(x, y, k\sigma) - G(x, y, \sigma) = (k - 1)\sigma^2 \nabla^2 G$$

The factor (k-1) is kept constant across scales.
➔does not influence extrema locations.

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Finding Keypoints – Scale, Location



Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Scale Invariant Detectors

- Harris-Laplacian (Mikolajczyk & Schmid)
  - *Apply Laplacian operation with varying scale.*
  - *Get local maxima of Harris corner response in space and scale.*

- SIFT (Lowe)
  - *Find local maximum Difference of Gaussians in space and scale*

K. Mikolajczyk, C.Schmid. “Indexing Based on Scale Invariant Interest Points”. ICCV 2001.

D.Lowe. “Distinctive Image Features from Scale-Invariant Keypoints”. IJCV 2004

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).

Methods:

1. Harris-Laplacian [Mikolajczyk, Schmid]: maximize Laplacian over scale, Harris' measure of corner response over the image.
2. SIFT [Lowe]: maximize Difference of Gaussians over scale and space.

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Keypoint localization

- There are still a lot of points, some of them are not good enough.
- The locations of keypoints may be not accurate.
- Eliminating edge points.

Eliminating edge points

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

$D(.)$=DoG of the function.

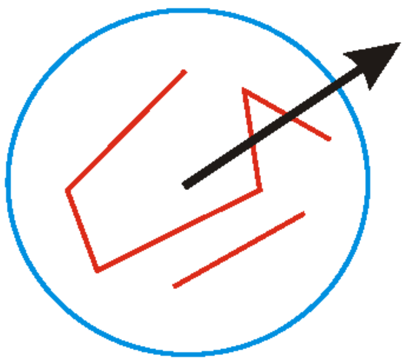
- Edge point: large principal curvature across the edge but a small one in the perpendicular direction.
- The eigenvalues of H are proportional to the principal curvatures, so two eigenvalues shouldn't differ too much.

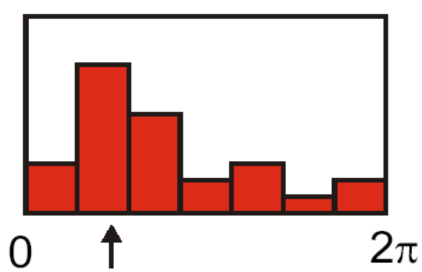
Eliminate keypoint if the ratio greater than the threshold.

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r}$$ for high value of r (say 10)

Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Orientation assignment





- Create histogram of local gradient directions at selected scale.
- Assign canonical orientation at peak of smoothed histogram.
- Each key specifies stable 2D coordinates (x, y, scale, orientation).

If there are two major orientations, use both.

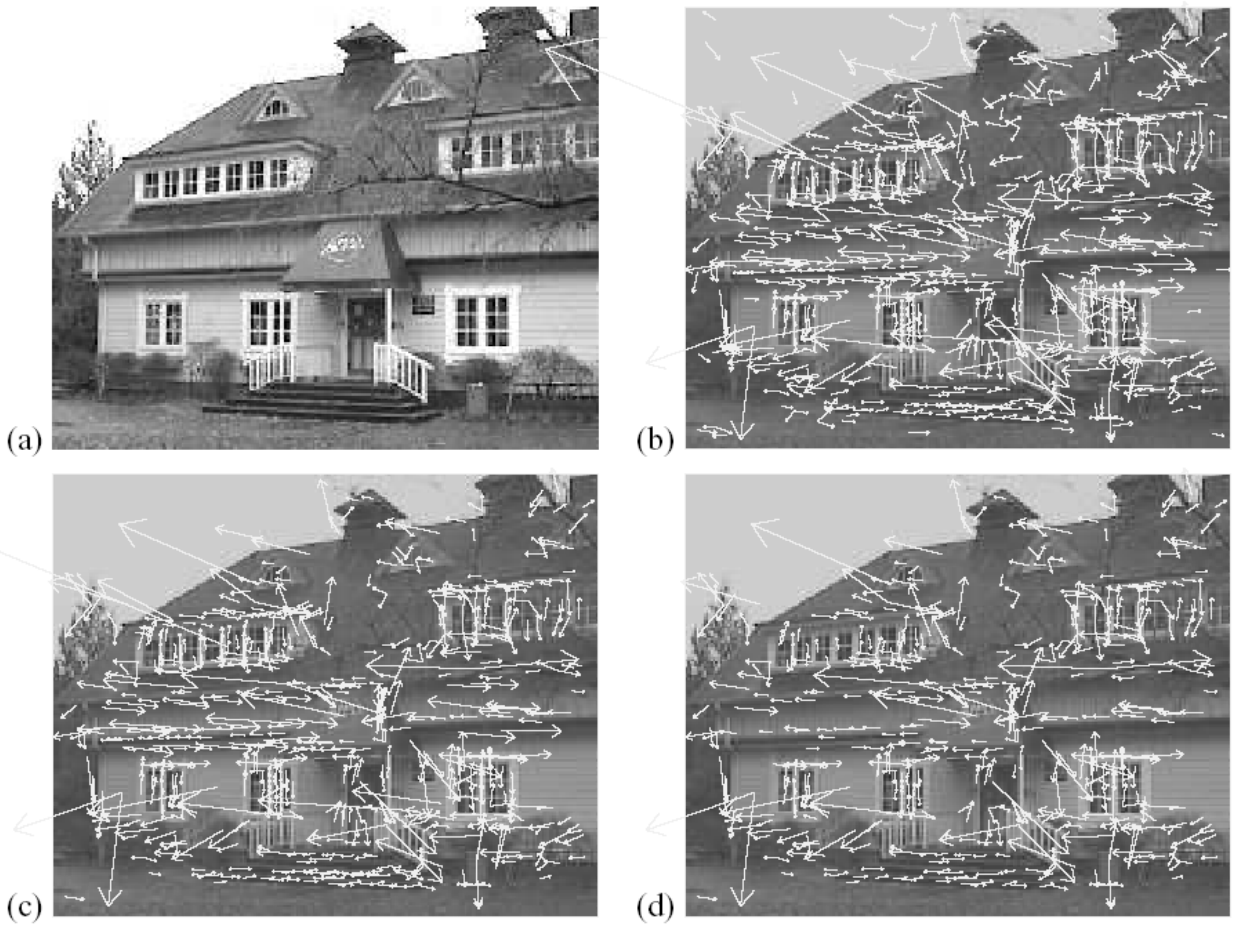
Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

Keypoint localization with orientation

233x189

initial keypoints

832



729

keypoints after gradient threshold

536

keypoints after ratio threshold

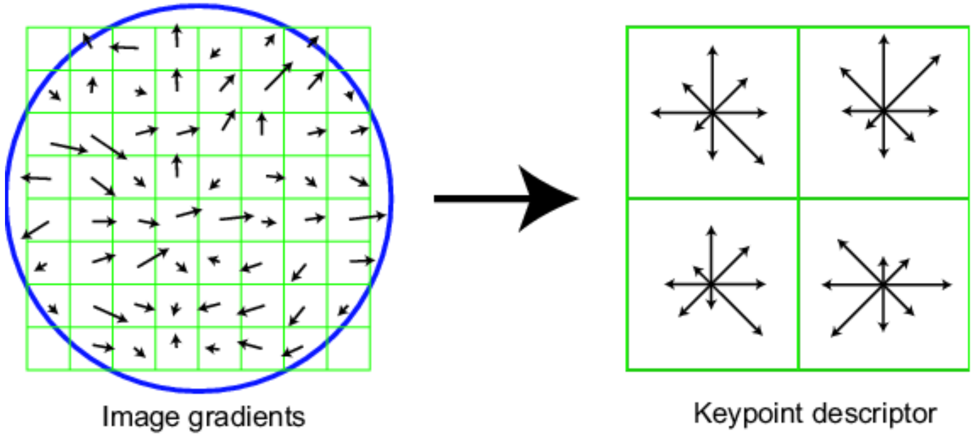
Slide adapted from Darya Frolova, Denis Simakov, Weizmann Institute.

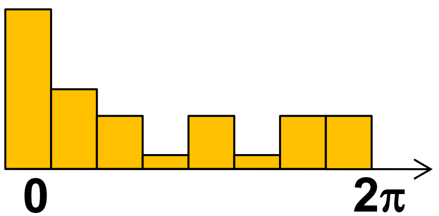
Keypoint Descriptors

- 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

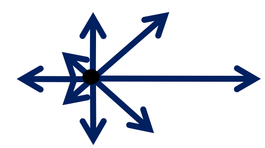
Scale Invariant Feature Transform (Lowe'99, ICCV)

- Take 16x16 square window (orientation corrected) around detected feature
- Compute edge orientation (angle of the gradient - 90°) for each pixel
- Throw out weak edges (threshold gradient magnitude)
- Create histogram of surviving edge orientations





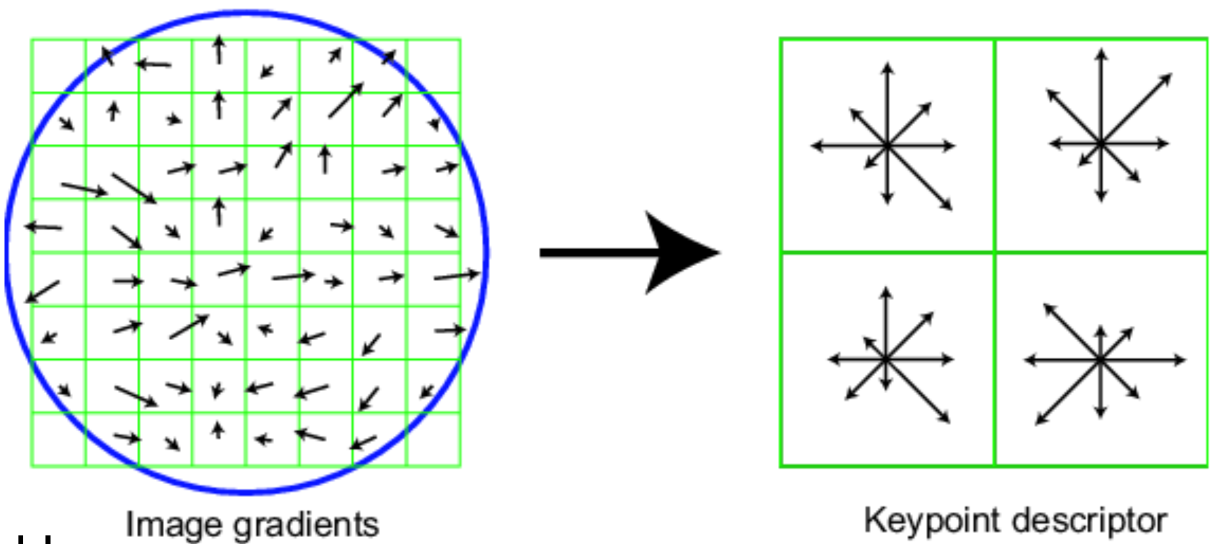
angle histogram



Adapted from slide by David Lowe

SIFT descriptor

- 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.



Adapted from slide by David Lowe

Robustness

Capable of handling

- changes in viewpoint
  - Up to about 60 degree out of plane rotation
- significant changes in illumination
  - Sometimes even day vs. night

Speeded-Up Robust Features (SURF): Another descriptor

- Speeded-Up Robust Features (SURF)
  - (Bay et al. ECCV, 2006)
  - Box-type convolution filters and use of integral images to speed up the computation.
  - Use of Hessian operator for key point detection→ Local maxima of det(H).
  - Accumulate orientation corrected Haar wavelet responses.

Hessian Operator

Convolution with the Gaussian second order derivative with image.

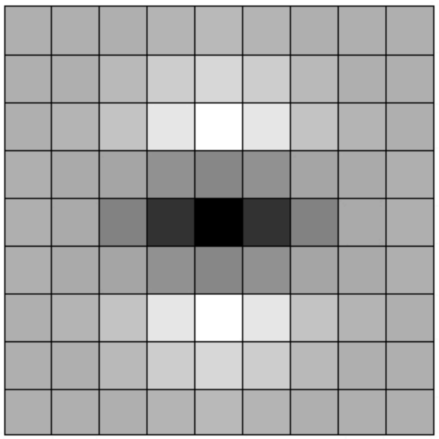
$$\mathcal{H}(\mathbf{x},\sigma) = \begin{bmatrix} D_{xx}(\mathbf{x},\sigma) & D_{xy}(\mathbf{x},\sigma) \\ D_{yx}(\mathbf{x},\sigma) & D_{yy}(\mathbf{x},\sigma) \end{bmatrix}$$

Keypoint: Maximum of det($\mathcal{H}$(.)) over space and scale.

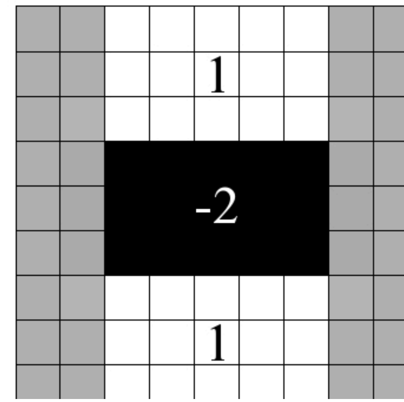
Bay et al, Speeded up robust features (SURF), CVIU, 2008

Approximation of Gaussian by Box filters

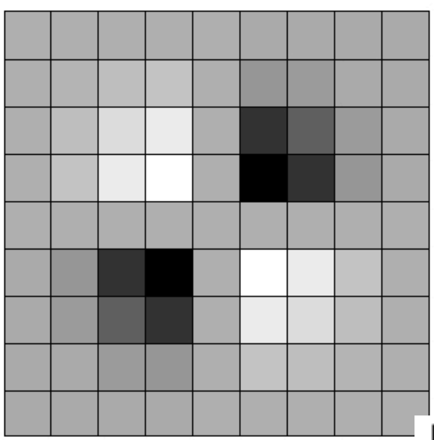
Bay et al, Speeded up robust features (SURF), CVIU, 2008



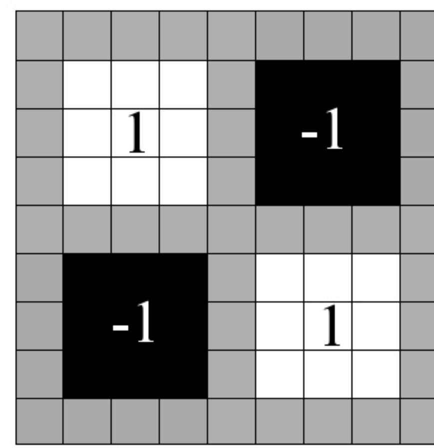
D_{yy}



9x9 Box-filters are approximation of Gaussian width 1.2.



D_{xy}

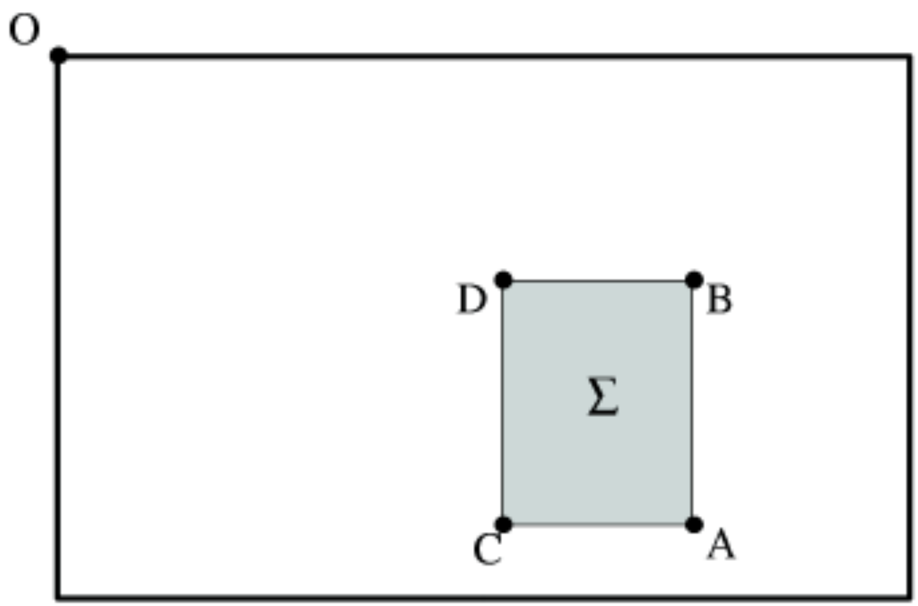


$$\boxed{\det(\mathcal{H}_{\text{approx}}) = D_{xx}D_{yy} - (wD_{xy})^2.}$$

w~0.9

Fast computation using the integral image

$$I_{\Sigma}(\mathbf{x}) = \sum_{i=0}^{i \le x} \sum_{j=0}^{j \le y} I(i, j)$$



$\boxed{\Sigma = A - B - C + D}$

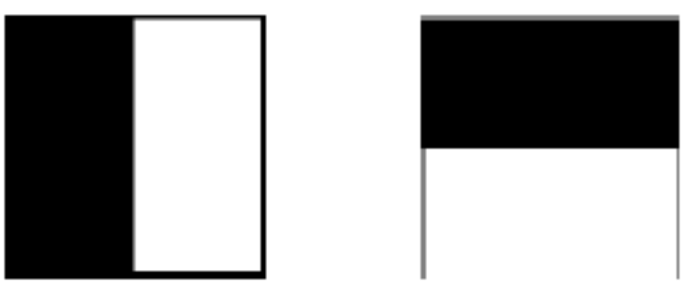
Only 3 additions and four memory access.

Bay et al, Speeded up robust features (SURF), CVIU, 2008

Haar Filter Responses

- Dominant orientation by accumulating Haar horizontal and vertical responses in a rotated sliding window (of width 60^0) at the scale of key point.
- The longest vector provides the dominant direction.

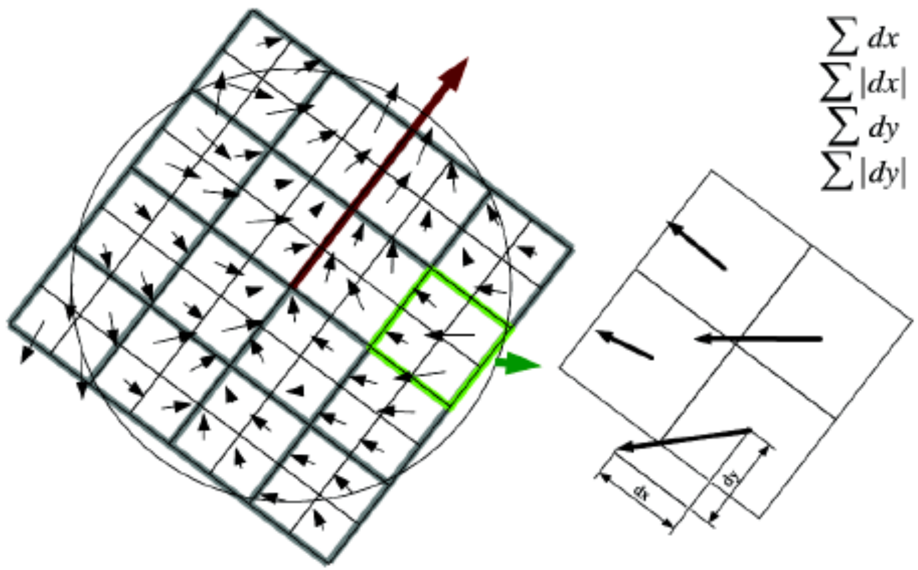
6 operations needed for computing each filter response using integral image.



Haar Filters
Box filter implementation.

SURF: Sum of Haar Wavelet responses

- Partitioned into 4x4 square sub-regions.
- Haar wavelet responses at regularly spaced 5x5 sample patches in each sub-region.
- Each sub-region has 4D vector.
- Concatenate them to 64 D vector.



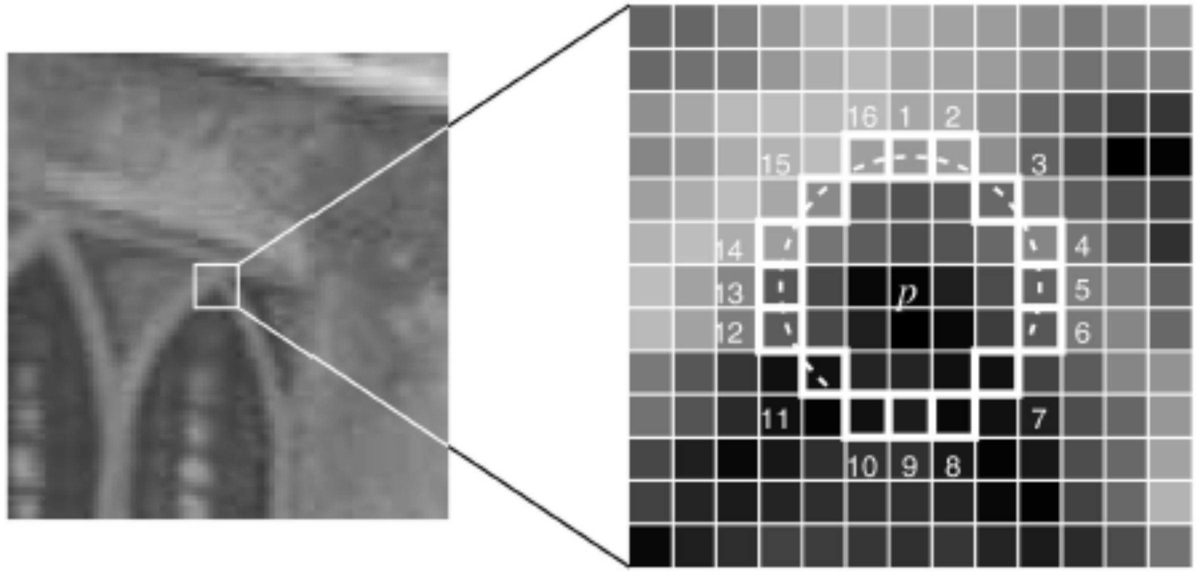
Size of the window=20 x scale

Bay et al, Speeded up robust features (SURF), CVIU, 2008

Other types of detectors and descriptors

- FAST: Features from Accelerated Segment Test (Rosten et al, PAMI, 2010).
- BRIEF: Binary Robust Independent Elementary Features (Colonder et al, ECCV, 2010).
- ORB: Oriented FAST and Rotated BRIEF (Rublee et al, ICCV, 2011)

FAST: Principle



- 12 point test: If there exists 12 consecutive points in the set of 16 points brighter than the central pixel (Rosten et al ICCV'05).
- Modified strategy: Train decision tree on the boolean conditions given labelled data.

Rosten et al, Faster and Better: A Machine Learning Approach to Corner Detection, PAMI, 2010

FAST: Partitioning of points on the circle and Training a DT

$$S_{p\to x} = \begin{cases} d, & I_{p\to x} \le I_p - t & (darker) \\ s, & I_p - t < I_{p\to x} < I_p + t & (similar) \\ b, & I_p + t \le I_{p\to x} & (brighter) \end{cases}$$

- Classification of points in three classes and create three partitions for a point.
- Train a decision tree.

Rosten et al, Faster and Better: A Machine Learning Approach to Corner Detection, PAMI, 2010

BRIEF: Principle

- Generate randomly a set of n_d pairs of locations ($\mathbf{x_i},\mathbf{y_i}$) in a patch centered around a point.
- Perform the following boolean test.

$$\tau(\mathbf{p};\mathbf{x},\mathbf{y}) = \begin{cases} 1 & if\ \mathbf{p}(\mathbf{x}) < \mathbf{p}(\mathbf{y}) \\ 0 & otherwise \end{cases}$$

BRIEF descriptor: n_d *Dimensional binary string*

128, 256, 512

$$f_{n_d}(\mathbf{p}) = \sum_{1 \le i \le n_d} 2^{i-1}\tau(\mathbf{p};\mathbf{x}_i,\mathbf{y}_i)$$

Colonder et al, BRIEF: Binary Robust Independent Elementary Features, ECCV, 2010

ORB: Principle

- FAST does not operate across scale.
- Apply FAST detector on pyramid of smoothened images.
- Orientation by intensity centroid: The vector from the center of the patch and to a centroid considering the intensity distribution (intensity weighted center of patch).
- Rotate the patch by the angle and compute BRIEF called steered BRIEF.

Matching

- Representation of a key-point by a feature vector.
  - e.g. $[f_0 f_1 \ldots f_n]^T$
- Use distance functions / similarity measures.
  - L_1 norm
  - L_2 norm
  - L_p norm

$$L_1(\vec{f},\vec{g}) = \sum_{i=0}^{n} |f_i - g_i|$$

$$L_2(\vec{f},\vec{g}) = \left(\sum_{i=0}^{n} |f_i - g_i|^2\right)^{\frac{1}{2}}$$

$$L_p(\vec{f},\vec{g}) = \left(\sum_{i=0}^{n} |f_i - g_i|^p\right)^{\frac{1}{p}}$$

Region descriptors

- Patch descriptors
- Texture descriptors
- Image / Sub-Image global descriptors

Patch Descriptor: Histogram of Gradients (HoG)

- Compute centered horizontal and vertical gradients with no smoothing.
- Compute gradient orientation and magnitudes,
- For color image, pick the color channel with the highest gradient magnitude for each pixel.
- For a 64x128 image, divide the image into 16x16 blocks of 50% overlap. →7x15=105 blocks in total.

N.Dalal and B. Triggs, Histograms of oriented gradients for human detection, CVPR-2005

Histogram of Gradients (HoG)

- Each block: 2x2 cells with size 8x8.
- Quantize the gradient orientation into 9 bins.
- The vote is the gradient magnitude.
- Interpolate votes between neighboring bin center.
- The vote can also be weighted with Gaussian to down-weight the pixels near the edges of the block.
- Concatenate histograms.
  - Feature dimension: 105x4x9 = 3,780

N.Dalal and B. Triggs, Histograms of oriented gradients for human detection, CVPR-2005

Object detection with patch descriptors.

- Typical examples:
  - Pedestrian detection
  - Character recognition
- Detection as a classification task.
  - Generate labeled sample feature descriptors.
  - Train a classifier.
    - NN, SVM, Decision Tree, Random Forest …..
  - Label an unknown patch using its descriptor.

Non-maximal suppression

- Expected to get a high detection score with neighboring overlapping patches.
  - Select the patch with locally maximal score.
- A greedy approach:
  - Select the best scoring window
    - It is expected to cover the target object.
  - Suppress the windows that are too close to the selected window.
  - Search next top-scoring windows out of the rest.

Texture descriptor

- Texture: spatial arrangement of the colors or intensities in an image
  - A quantitative measure of the arrangement of intensities in the region.

Computer Vision by Shapiro and Stockman, Pearson, (2001)

Texture descriptors

- Edge density and direction
- Local Binary Pattern (LBP).
- Co-occurrence Matrix.
- Laws’ texture energy features.

Edge density and direction

- Compute gradient at each pixel.
- The descriptor: normalized histograms of magnitudes and directions of gradients over a region.
  - $(H_R(mag), H_R(dir))$

Normalized histogram of magnitudes.

Normalized histogram of directions.

- Numbers of bins in histograms kept small (e.g. 10).
  - Use L1 norm between the feature vectors as a distance measure.

Normalized histogram → Area =1;

Local Binary Pattern (LBP).

3	2	1
4	c	0
5	6	7

$$b(i) = \begin{cases} 1 & if\ (I(i) > I(c)) \\ 0 & Otherwise \end{cases}$$

$$LBP(c) = \sum_{i=0}^{7} b(i)2^i$$

You may have different ordering of neighbors.

- Values range from 0 to 255.
- Obtain normalized histogram over a region.
- Not rotational invariant.
- Invariant to illumination and contrast.

T. Ojala, M. Pietikainen, and D. Harwood, A Comparative Study of Texture Measures with Classification Based on Feature Distributions, Pattern Recognition, vol. 29, pp. 51-59, 1996.

Variations of LBP

- Making it rotational invariant.
  - A circular neighborhood of radius R, with P pixels at equal intervals of angles.
  - Use interpolation if does not belong to the discrete grid.

$$LBP_{P,R}(c) = \sum_{i=0}^{P-1} b(i)2^i \qquad LBP_{8,1} \leftarrow \rightarrow LBP$$

$$LBP_{P,R}^{ri}(c) = \min\{ROR(LBP_{P,R}(c), i) \mid i = 0,1,2, \ldots P-1\}$$

where $ROR(x, i)$ performs a circular bit-wise right shift on the P-bit number x, i times.

36 distinct values for $LBP^{ri}{}_{8,1}$.

T. Ojala, M. Pietikainen, and T.i Maenpaa , Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, TPAMI, VOL. 24, NO. 7, JULY 2002, pp. 971-987

Variations of LBP

- Uniform pattern
  - Not more than 2 spatial transitions in the bit sequence.
  - U(11111111)=0
  - U(11101111)=2
  - U(10001001)=4

Use rotation invariant value by computing minimum applying ROR operator.

$$LBP_{P,R}^{riu2}(c) = \begin{cases} \sum_{i=0}^{P-1} b(i)2^i & if\ U(LBP_{p.R}(c)) \le 2 \\ P+1 & Otherwise \end{cases}$$

Exactly P+1 uniform patterns, hence P+2 distinct values.

T. Ojala, M. Pietikainen, and T.i Maenpaa , Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, TPAMI, VOL. 24, NO. 7, JULY 2002, pp. 971-987

Nine uniform patterns of $LBP_{8,R}^{riu2}$

- The nine uniform patterns and the numbers inside them correspond to their unique codes.

T. Ojala, M. Pietikainen, and T.i Maenpaa , Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, TPAMI, VOL. 24, NO. 7, JULY 2002, pp. 971-987

Variance Measures of the Contrast

- Local variance of intensities of uniform pattern.
- Normalized histogram of local variances.

$$var_{P,R}(c) = \frac{1}{P}\sum_{p=0}^{P-1}(g_p - \mu)^2 \qquad \mu = \frac{1}{P}\sum_{p=0}^{P-1} g_p$$

Another robust representation: $\dfrac{LBP_{P,R}^{riu2}}{var_{P,R}}$

T. Ojala, M. Pietikainen, and T.i Maenpaa , Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, TPAMI, VOL. 24, NO. 7, JULY 2002, pp. 971-987

Co-occurrence Matrix (C_r)

- $C_r(x,y)$: How many times elements x and y occur at a pair of pixels related spatially (designated by r in the notation).
  - e.g. **p** r **q** denotes **q** is shifted from **p** by a translation of **t**=(a,b), i.e. **q**=**p**+**t**.
    - $C_{(a,b)}(x,y)$: Number of cases in an image where I(**p**)=x and I(**p**+**t**)=y.

Co-occurrence Matrix (C_r)

0	0	1	1
0	0	1	1
1	1	0	0
1	1	0	0

	0	1
0		
1		

$C_{(0,1)}$

	0	1
0		
1		

$C_{(1,0)}$

	0	1
0		
1		

$C_{(0,1)}$

Co-occurrence Matrix (C_r)

0	0	1	1
0	0	1	1
1	1	0	0
1	1	0	0

	0	1
0	4	2
1	2	4

$C_{(0,1)}$

	0	1
0	4	2
1	2	4

$C_{(1,0)}$

	0	1
0	2	2
1	2	3

$C_{(1,1)}$

Normalized Co-occurrence Matrix (N_r)

Divide by the sum of frequencies in a matrix.

0	0	1	1
0	0	1	1
1	1	0	0
1	1	0	0

	0	1
0	1/3	1/6
1	1/6	1/3

$C_{(0,1)}$

	0	1
0	1/3	1/6
1	1/6	1/3

$C_{(1,0)}$

	0	1
0	2/9	2/9
1	2/9	1/3

$C_{(1,1)}$

Symmetric Co-occurrence Matrix (S_r)

$$S_r(x,y)=C_r(x,y)+C_{-r}(x,y)$$

0	0	1	1
0	0	1	1
1	1	0	0
1	1	0	0

	0	1
0	4+4	2+2
1	2+2	4+4

$C_{(0,1)}+C_{(0,-1)}$

	0	1
0	4+4	2+2
1	4+4	2+2

$C_{(1,0)}+C_{(-1,0)}$

	0	1
0	2+2	2+2
1	2+2	3+3

$C_{(1,1)}+C_{(-1,-1)}$

Features from Normalized Co-occurrence Matrix

$$Energy = \sum_x \sum_y N_r^2(x,y)$$

$$Entropy = -\sum_x \sum_y N_r(x,y)\log_2 N_r(x,y)$$

$$Contrast = \sum_x \sum_y (x-y)^2 N_r(x,y)$$

$$Homogeneity = \sum_x \sum_y \frac{N_r(x,y)}{1+|x-y|}$$

Features from Normalized Co-occurrence Matrix

$$Correlation = \frac{\sum_x \sum_y (x - \mu_x)\left(y - \mu_y\right) N_r(x, y)}{\sigma_x \sigma_y}$$

Mean and s.d. of row sums

Mean and s.d. of column sums

$$f(x) = \sum_y N_r(x, y) \qquad g(y) = \sum_x N_r(x, y)$$

Laws' texture energy features

- A set of 9 5x5 masks used to compute texture energy.

L5 (Level): [1 4 6 4 1]
E5 (Edge): [-1 -2 0 2 1]
S5 (Spot): [-1 0 2 0 -1]
R5 (ripple): [1 -4 6 -4 1]

Computation with mask: Convolution

A mask: Outer product of any pair.
e.g. E5L5: E5.L5^T

$$\begin{bmatrix} -1 \\ -2 \\ 0 \\ 2 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

K. Laws, "Rapid Texture Identification", in SPIE Vol. 238: Image Processing for Missile Guidance, 1980, pp. 376-380.

Laws' texture energy features

- A set of 9 5x5 masks used to compute texture energy.

L5 (Level): [1 4 6 4 1]
E5 (Edge): [-1 -2 0 2 1]
S5 (Spot): [-1 0 2 0 -1]
R5 (ripple): [1 -4 6 -4 1]

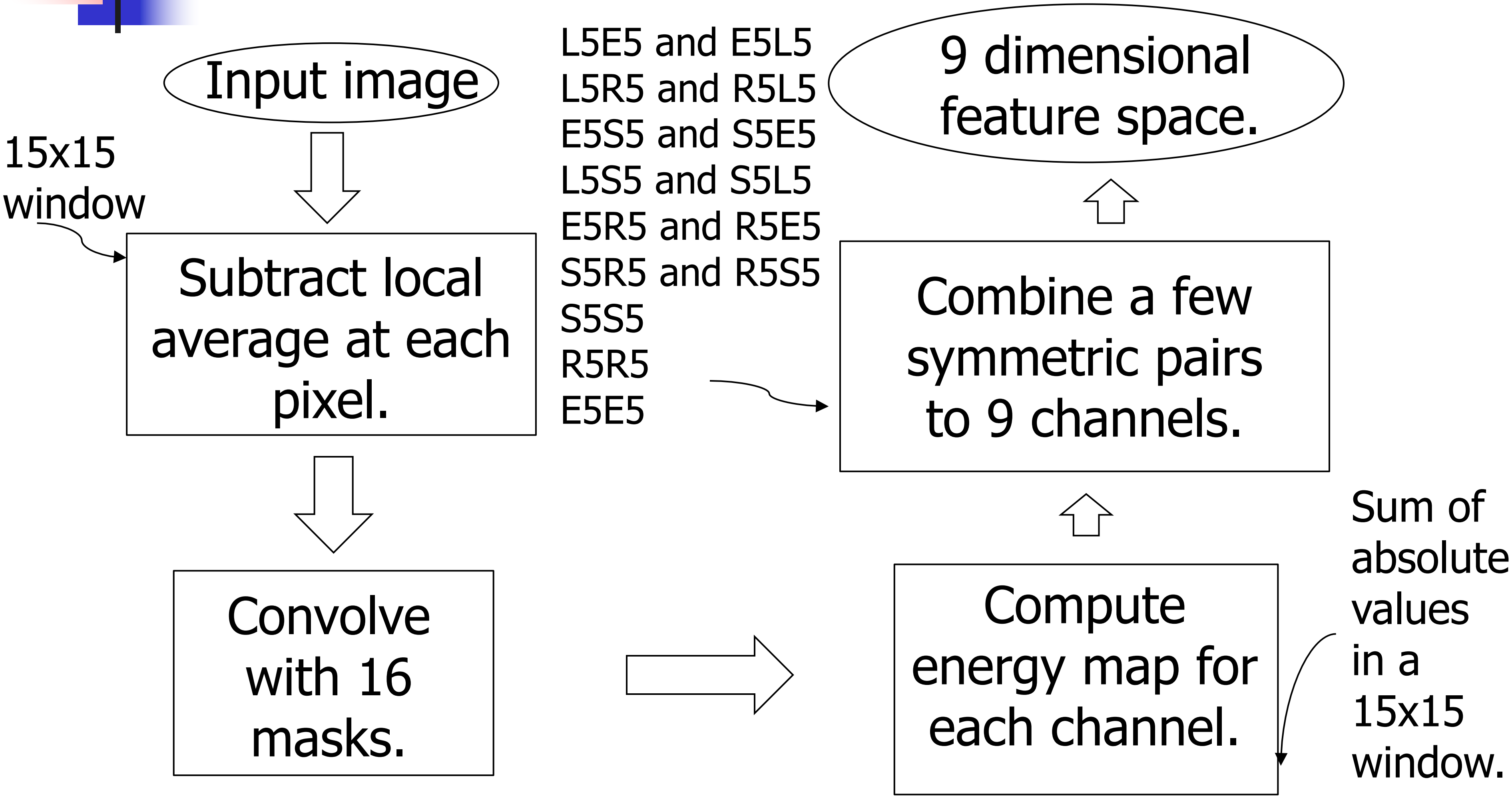
Take average of responses of two masks.

L5E5 and E5L5
L5R5 and R5L5
E5S5 and S5E5
L5S5 and S5L5
E5R5 and R5E5
S5R5 and R5S5
S5S5
R5R5
E5E5

16 such masks possible.
Combine a few pairs to make 9 masks.

K. Laws, "Rapid Texture Identification", in SPIE Vol. 238: Image Processing for Missile Guidance, 1980, pp. 376-380.

Laws' texture energy



Use of texture descriptors

- Detection of object patches represented by textured patterns.
- Segmentation of images.
- Classification / Matching
  - Generate a Library of labelled feature descriptors.
  - Detection of classes (class labels).
    - Matching to the nearest texture descriptor.

Image / Object Descriptor

- Bag of visual words
  - Compute key-point based feature descriptors over a library of images.
  - Quantize them (clustering) to form a finite set of representative descriptors (visual words).
  - For an image assign the nearest visual word corresponding to the feature descriptor of a key point.
  - Represent by each image by a histogram of visual words.

K-means clustering

Sivic et. al., Discovering objects and their location in images, ICCV' 2005.

Vector of locally aggregated descriptors (VLAD)

- Form the codebook of visual words as in BoVW representation.
  - $C_1, C_2, \ldots, C_k$ Cluster centers of dimension D (say).
- Each local descriptor x in an image is associated to one of these visual words.
- Accumulate the differences w.r.t. the corresponding cluster center.

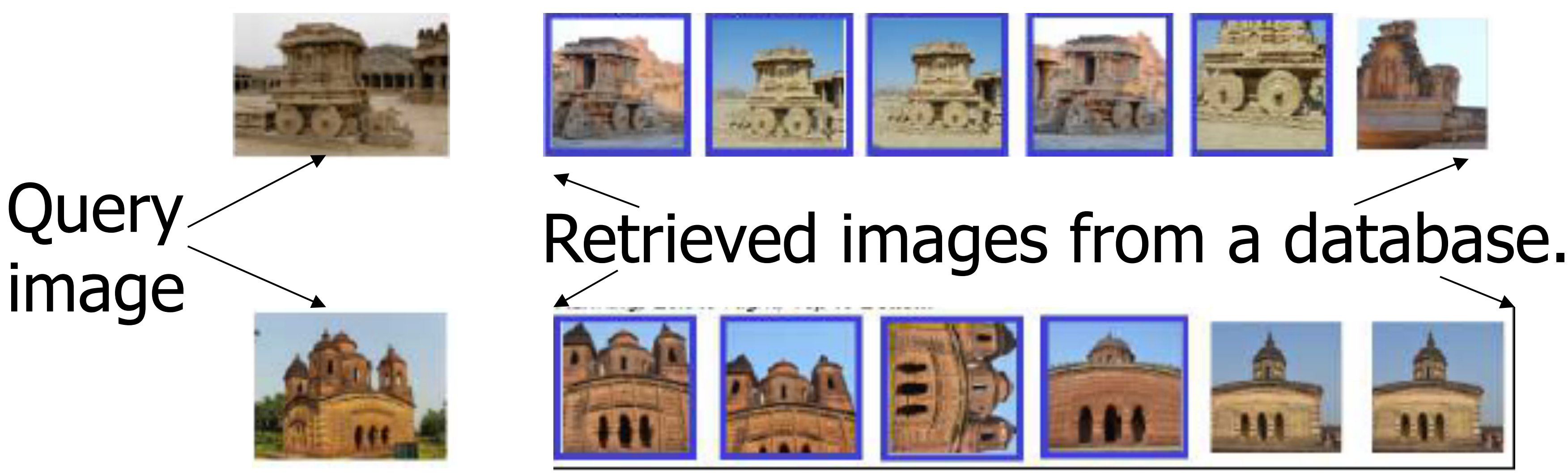
$$v_i = \sum_{x \text{ assigned to } C_i} (x - C_i)$$

  - Form $V=[v_1\ v_2\ \ldots.\ v_k]$
  - VLAD descriptor$=V/||V||$ Dimension: $k.D$

Jegou et al., Aggregating local descriptors into a compact image representation, CVPR, 2010.

Application of global image descriptor

- Content based image retrieval
  - Image search based on visual content



Summary of Techniques

- Scale and transformation invariant feature detection:
  - Harris corner-Laplacian Maximum.
  - DOG Maximum.
  - Intensity weighted FAST.
- Scale and transformation invariant Feature descriptor:
  - SIFT, SURF, ORB

Summary of Techniques

- Region and texture descriptors.
  - HoG
  - Edge density
  - LBP
  - Co-occurrence matrix
  - Laws' texture energy
- Image global descriptor
  - BoVW
  - VLAD

- A few applications
  - Key point descriptor: Matching corresponding points of a scene.
  - Region descriptor: Object detection
  - Global descriptor :Image retrieval.