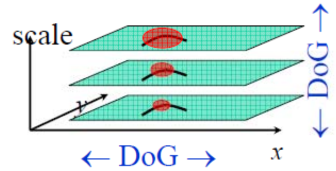
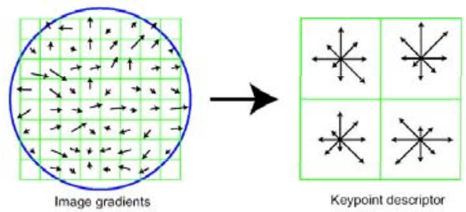
Scale Invariant Feature Transform

SIFT [Scale Invariant Feature Transform]

David Lowe, 1999



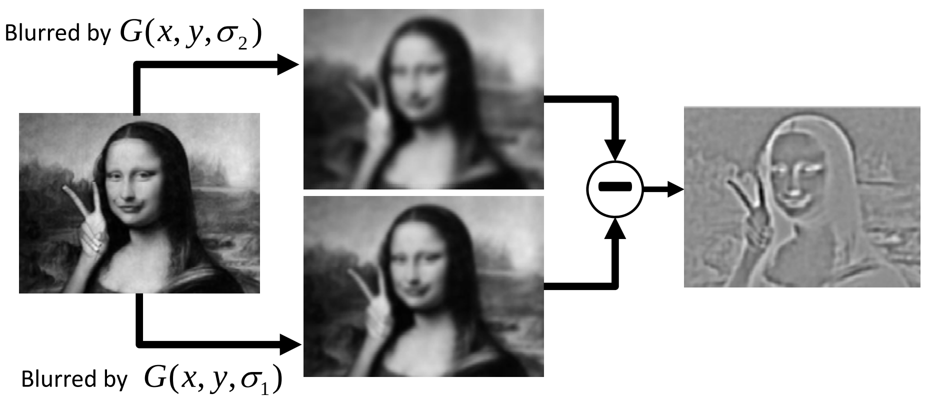
- **SIFT Keypoint Extraction**
  - Difference of Gaussian (DoG)
  - Scale space



- **SIFT Descriptors**
  - Histogram of gradient orientation

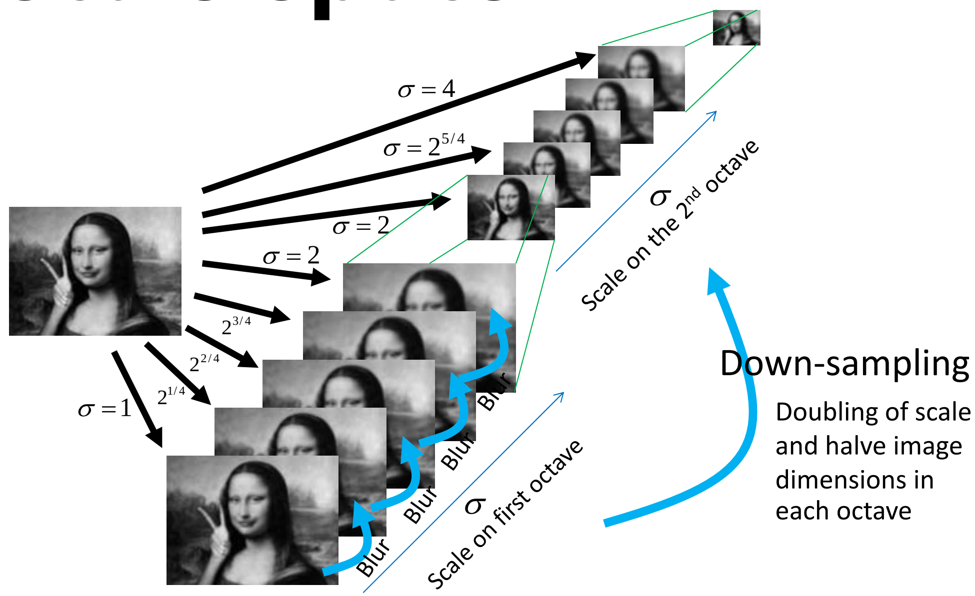
Adapted from David Lowe and Dan Huttenlocher

DoG [Difference of Gaussian]

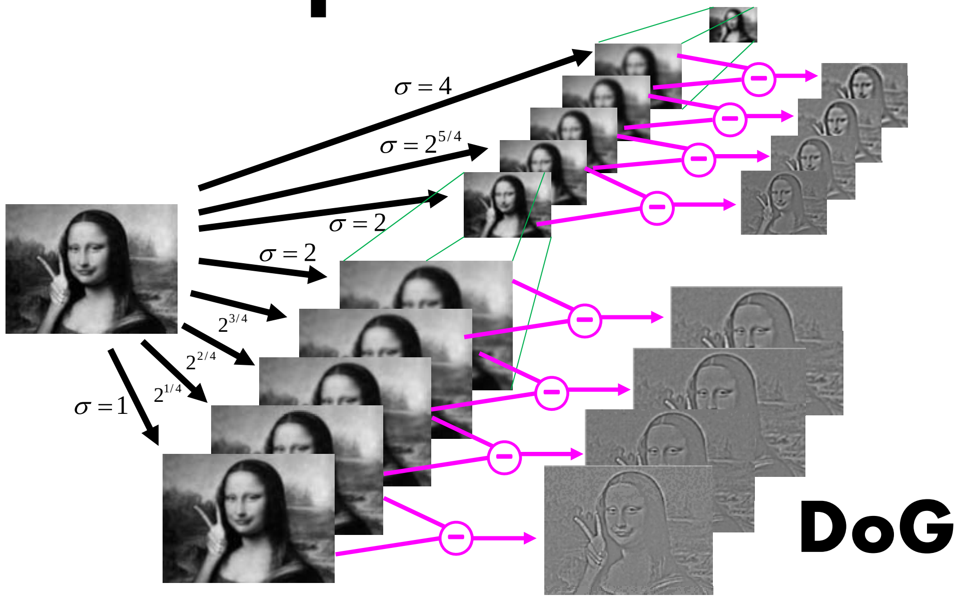


Gaussian filter kernel $G(x,y,\sigma)=\frac{1}{\sqrt{2\pi\sigma^2}}e^{\frac{-(x^2+y^2)}{2\sigma^2}}$

Scale Space

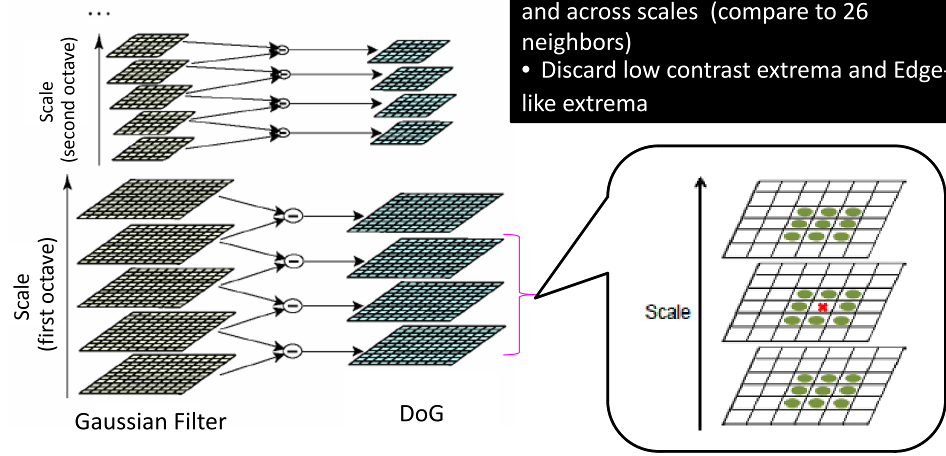


Scale Space

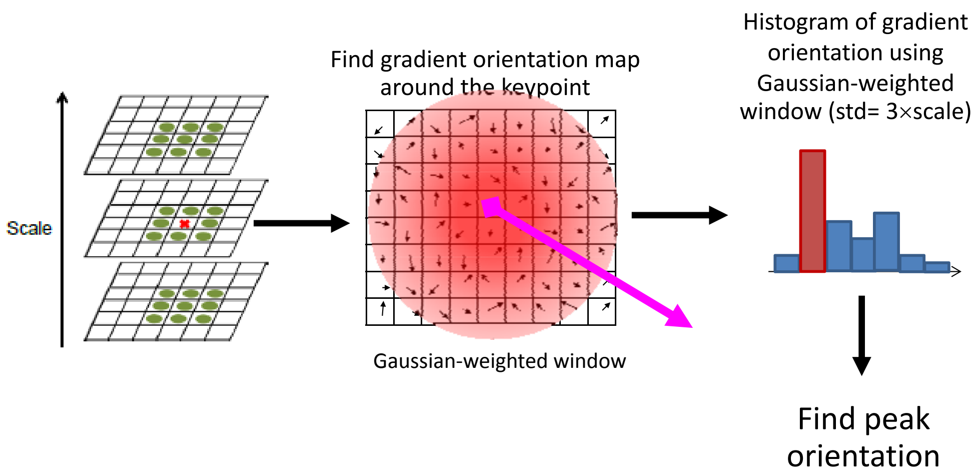


SIFT Keypoints

- Detect minima and maxima both spatially and across scales (compare to 26 neighbors)
- Discard low contrast extrema and Edge-like extrema



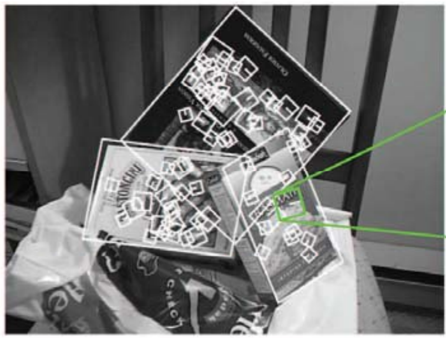
Orientation Assignment

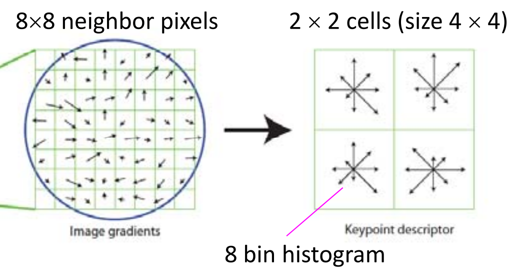


SIFT Descriptors

- Histogram of gradient
  - Weighted by magnitude
  - and Gaussian from center

For example:
- Use 16×16 neighbor pixels
- Divide into 4 × 4 cells (size of 4 × 4)
- Compute 8-bin histogram for each cell
- Concatenate histogram from all cells
- Descriptor length = $\underbrace{16}_{\text{cells}} \times \underbrace{8}_{\text{bins}} = 128$





Visual Object Recognition, Kristen Grauman and Bastian Leibe

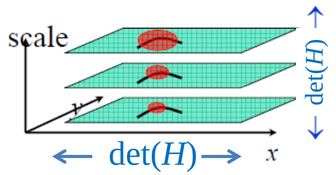
Properties of SIFT

- Extraordinarily robust matching technique
  - Can handle changes in viewpoint $\Rightarrow$ Up to about 60 degree out of plane rotation
  - Can handle significant changes in illumination $\Rightarrow$ Sometimes even day vs. night (below)
  - Fast and efficient—can run in real time
  - Lots of code available


Speed Up Robust Features

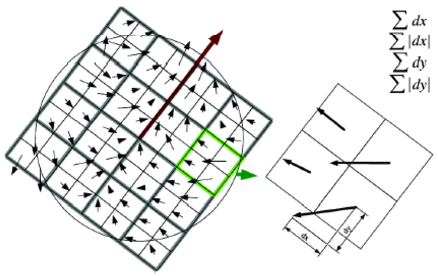
SURF [Speed Up Robust Features]

Herbert Bay et al., 2008



Approximated determinant of Hessian matrix

- **SURF Keypoint Extraction**
  - Fast Hessian Detector
  - Integral Image
  - Scale space (Gaussian derivatives)



- **SURF Descriptors**
  - Haar wavelet responses

Hessian Detector

Image

I ➡

Hessian Matrix

$$H = \begin{bmatrix} I_{XX} & I_{XY} \\ I_{XY} & I_{YY} \end{bmatrix}$$

(for each pixel)

➡

$$\det(H)$$

$$I_{XX} I_{YY} - (I_{XY})^2$$

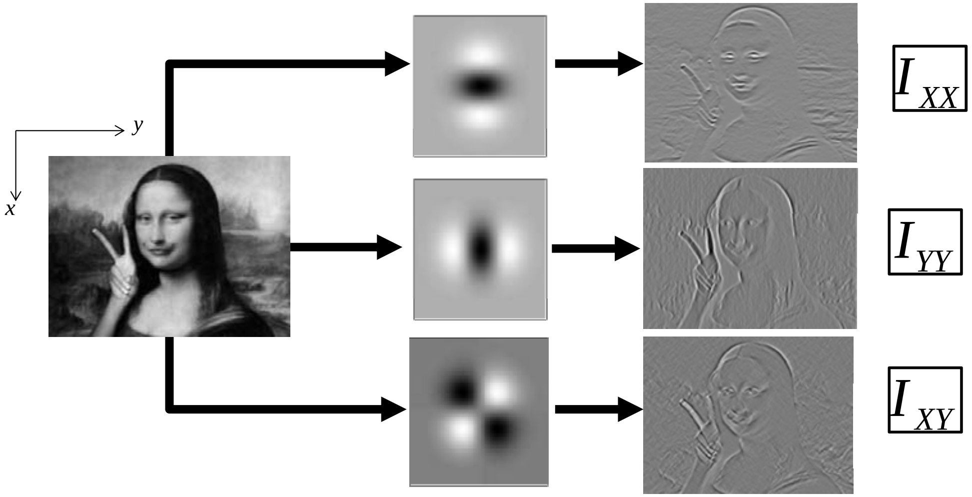
(for each pixel)

Thresholding
+
Non-maximum suppression

Keypoints

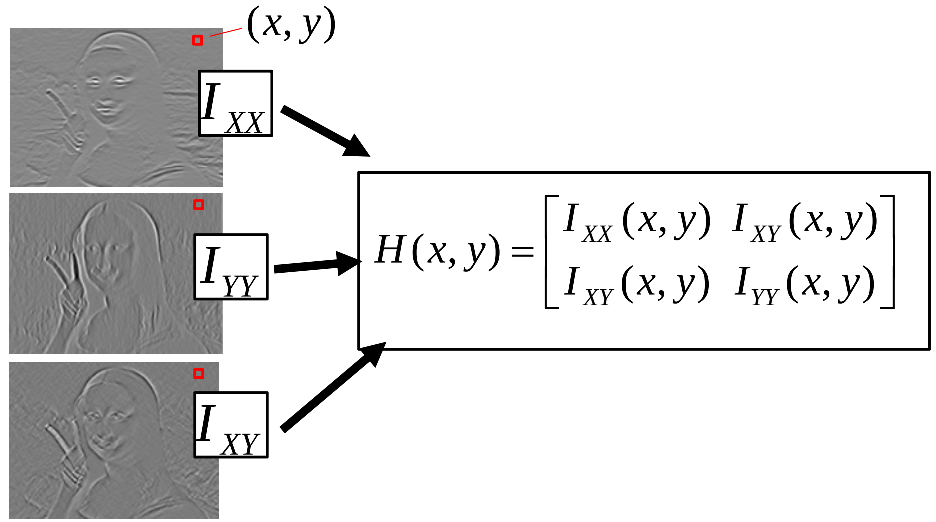
Locations which have strong change in gradient along both the orthogonal direction.

Hessian Matrix

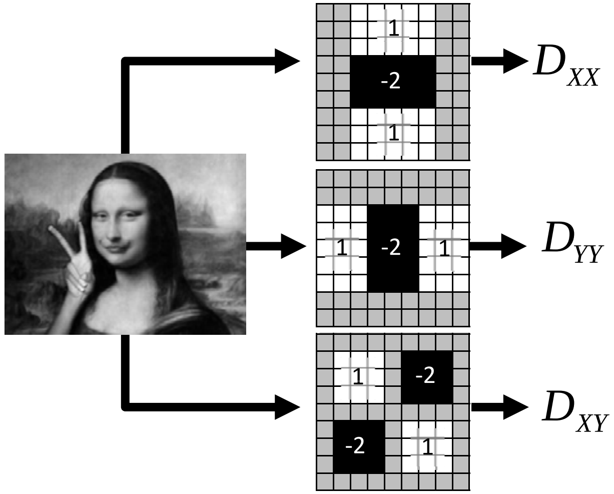


The 2nd-order Gaussian smoothed derivative filters

Hessian Matrix



Fast Hessian Detector



Approximated box filters

$$\tilde{H} = \begin{bmatrix} D_{XX} & D_{XY} \\ D_{XY} & D_{YY} \end{bmatrix}$$

$$\det(H) \approx D_{XX} D_{YY} - (0.9 D_{XY})^2$$

Integral Image

1	2	3	4	3	0	6	7	9
4	5	6	2	1	5	4	5	2
1	4	8	6	3	2	9	0	0
2	1	4	5	7	0	2	0	2
1	1	1	0	1	4	4	2	5
1	2	4	0	0	0	2	5	1
9	8	2	5	7	8	9	1	0
1	2	7	5	0	2	1	0	0

Original Image

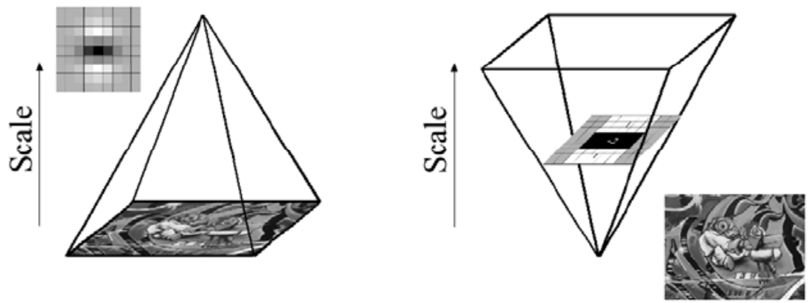
1	3	6	10	13	13	19	26	35
5	12	21	27	31	36	46	58	69
6	17	34	46	53	60	79	91	102
8	20	41	58	72	79	100	112	125
9	22	44	61	76	87	112	126	144
10	25	51	68	83	94	121	140	159
19	42	70	92	114	133	169	189	208
20	45	80	107	129	150	187	207	226

Integral Image

= 140-91-51+34 = 32

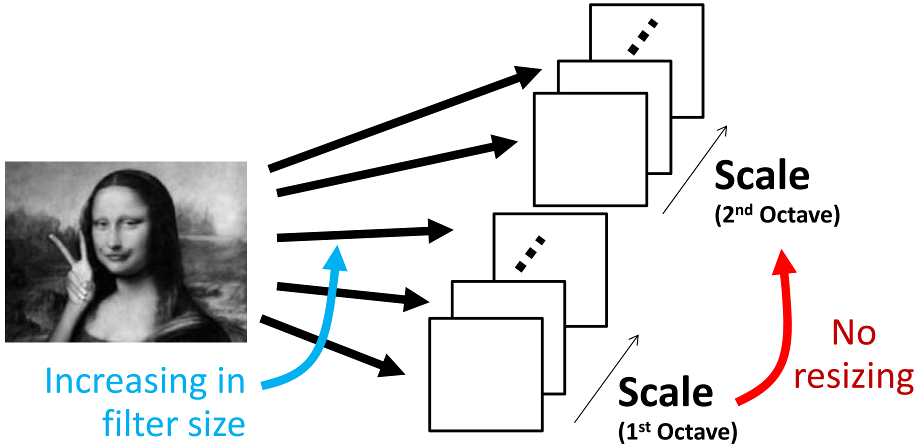
Can be compute using 3 operators

Scale Space



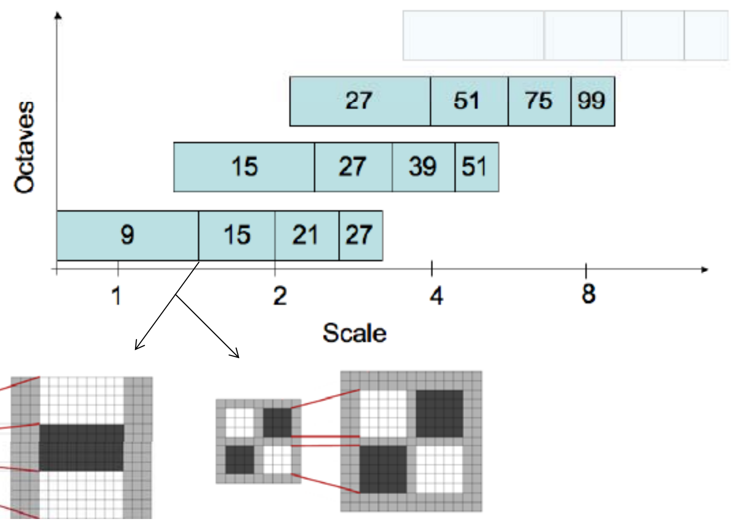
Instead of iteratively reducing the image size (left), the use of integral images allows the up-scaling of the filter at constant cost (right).

Scale Space



Approximated determinant of Hessian matrix from various scale

Scale Space



Time Comparison

detector	threshold	nb of points	comp. time (ms)
FH-15	60000	1813	160
FH-9	50000	1411	70
Hessian-Laplace*	1000	1979	700
Harris-Laplace	2500	1664	2100
DoG	default	1520	400

FH-15 = Fast Hessian detector with filter kernel of size 15×15

FH-9 = Fast Hessian detector with filter kernel of size 9×9

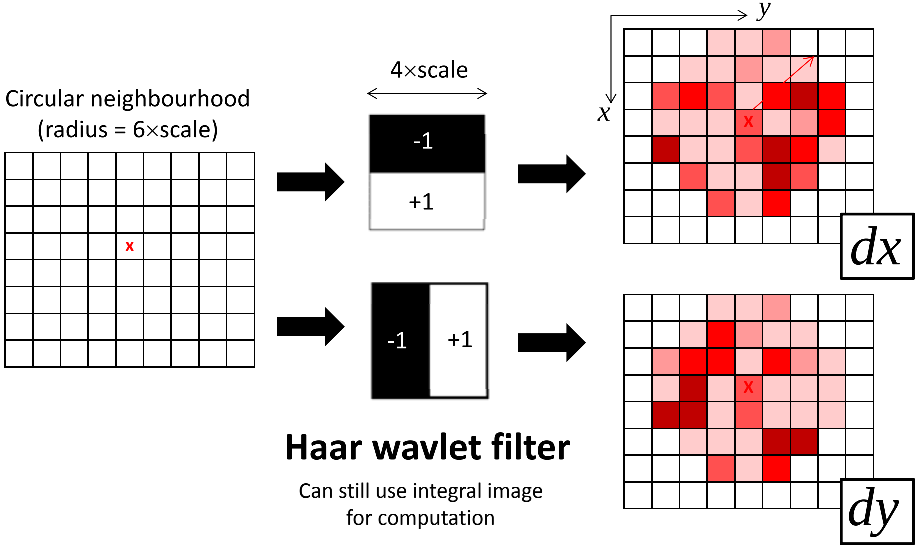
Hessian-Laplace = Hessian corner detector + Laplacian-of-Gaussian (LoG) scale space

Harris-Laplace = Harris corner detector + Laplacian-of-Gaussian (LoG) scale space

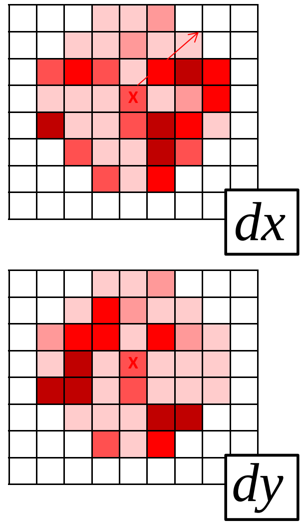
DoG= Difference of Gaussian detector used in SIFT

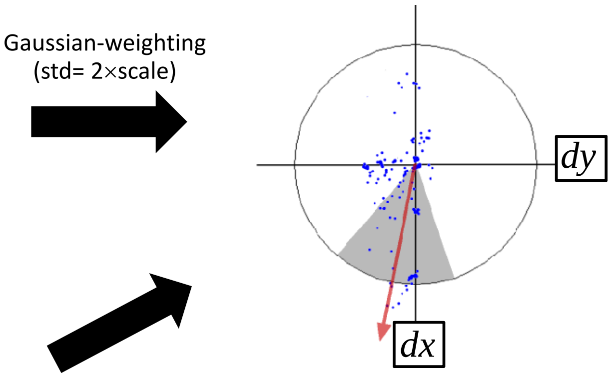
* K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. IJCV, 60(1):63-86, 2004.

Orientation Assignment



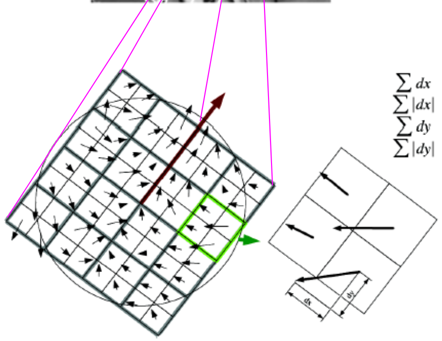
Orientation Assignment





- Calculating sum of horizontal and vertical responses within sliding orientation bin.
- The two summed responses yield a orientation vector
- The longest such vector over all orientation bins defines the orientation of the keypoint.

SURF Descriptors



- Construct a square region centred around the keypoint and oriented along the orientation (size = 20×scale).
- Split up the regions into smaller 4×4 square sub-regions.
  - For each sub-region, compute Haar wavelet response.

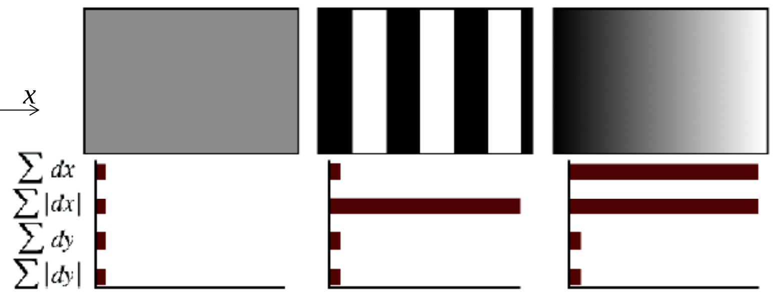
$$dx, dy$$

  - Apply Gaussian weight to wavelet response.
  - Compute 4 features for each sub-region.

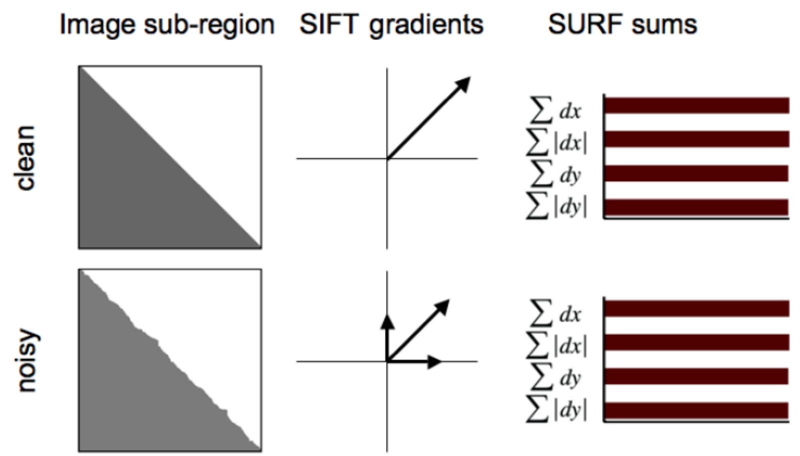
$$\sum dx, \sum |dx|, \sum dy, \sum |dy|$$

  - Concatenate features from all 4×4 sub-regions = descriptor vector of length 64. (do normalization after concatenating)

SURF Descriptors



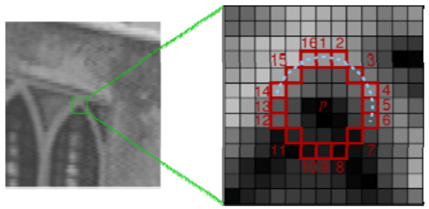
SURF Descriptors

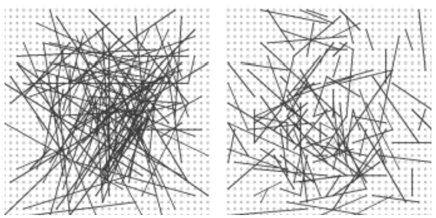


Oriented FAST and Rotated BRIEF

ORB [Oriented FAST and Rotated BRIEF]

E Rublee et al., 2011

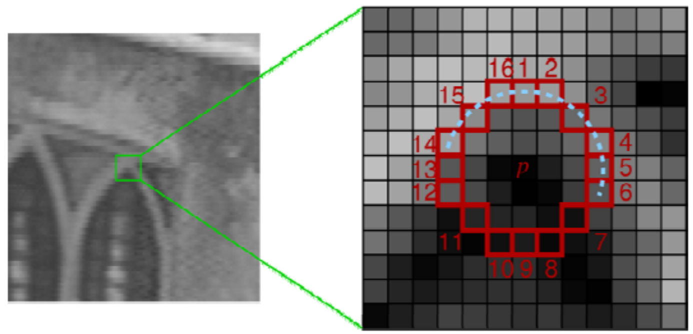




- **FAST Keypoint Extraction**
  - Segment-test detector
  - Decision Tree
  - Scale Pyramid
- **BRIEF Descriptors**
  - Bit string description
  - Intensity comparison of pixel pair
  - Low complexity and memory usage

FAST [Features from Accelerated Segment Test]

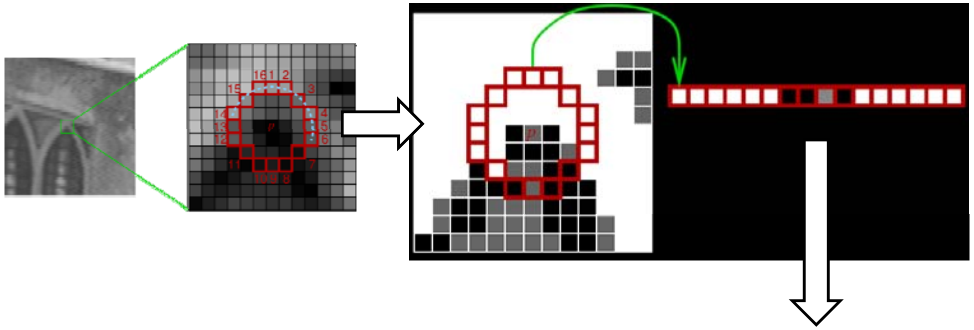
Edward Rosten et al., 2006



Classifies p as a corner if there exists a set of n contiguous pixels in the circle which all brighter than p (+threshold) or all darker than p (-threshold)

FAST [Features from Accelerated Segment Test]

Edward Rosten et al., 2006

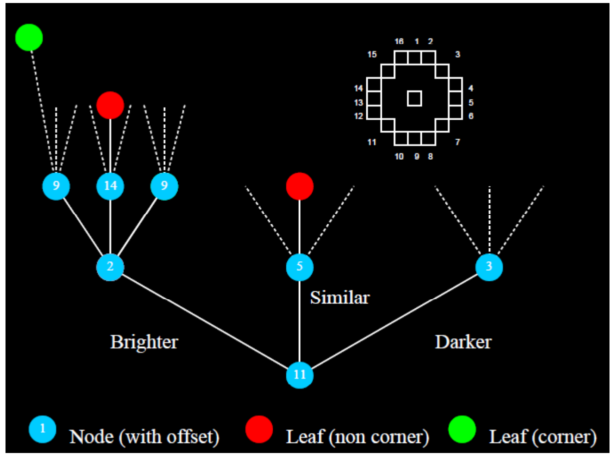


Classify by decision tree

FAST [Features from Accelerated Segment Test]

Edward Rosten et al., 2006

Decision Tree



FAST [Features from Accelerated Segment Test]

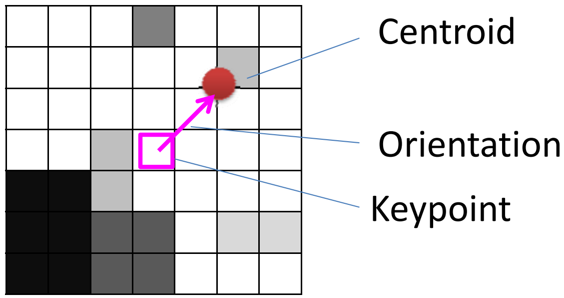
Edward Rosten et al., 2006

- FAST does not produce a measure of cornerness, and it has large responses along edges.
- Harris corner measure may be used to verify the FAST keypoints.
- To obtain ORB keypoints, FAST keypoints are ordered according to the Harris measure, then the top N points are selected as ORB keypoints.
- FAST does not produce multi-scale features. A scale pyramid of the image is used, FAST features (filtered by Harris) are produced at each level in the pyramid.

Orientation Assignment

Intensity centroid :

The intensity centroid assumes that a corner's intensity is offset from its center, and this vector may be used to impute an orientation.



$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right)$$

$$\theta = \tan^{-1}\left(\frac{m_{01}}{m_{10}} \right)$$

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y)$$

BRIEF

[Binary Robust Independent Elementary Features]

M. Calonder et al., 2010

BRIEF descriptor = a bit string description of an image patch constructed from a set of binary intensity tests.

A binary intensity tests τ

$$\tau(I,\mathbf{p},\mathbf{q}) = \begin{cases} 1; & I(\mathbf{p}) < I(\mathbf{q}) \\ 0; & I(\mathbf{p}) \geq I(\mathbf{q}) \end{cases}$$

$I = \text{Image patch}$

$\mathbf{p},\mathbf{q} = \text{Coordinate}\,(x,y)$

BRIEF

[Binary Robust Independent Elementary Features]

M. Calonder et al., 2010

BRIEF descriptor

0	1	0	0	1	0	...	1	1

$$f_n(I) = \sum_{1 \leq i \leq n} 2^{i-1} \tau(I,\mathbf{p}_i,\mathbf{q}_i) \Rightarrow$$

Convert to bit string

- There are many different types of distributions of test points
- ORB chooses a Gaussian distribution around the center of the patch and vector length n = 256.
- The image should be smoothed before performing the tests.
- Hamming Distance can be used to match the descriptor

To obtain rotation invariance descriptor, ORB steer BRIEF according to the orientation of keypoints.

BRIEF

[Binary Robust Independent Elementary Features]

M. Calonder et al., 2010

y (→), x (↓)

1	2	3	4	3	0
4	5	6	2	1	5
1	4	8	6	3	2
2	1	4	5	7	0
1	1	1	0	1	4
1	2	4	0	0	0

Image Patch

I

$(0,0),(0,1)$

$(4,3),(3,1)$

$(1,2),(3,2)$

$(0,4),(1,3)$

List of location pairs

$(\mathbf{p}_i,\mathbf{q}_i)$

$\Rightarrow$

1	**1**		

BRIEF Descriptor

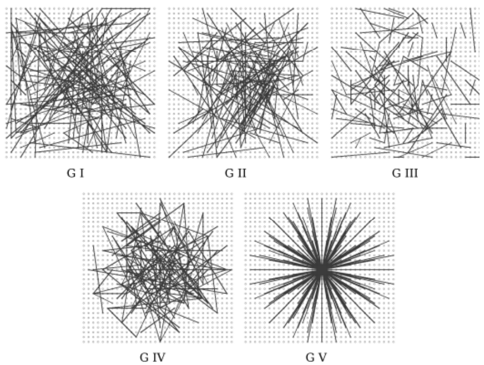
$f_n(I)$

BRIEF

[Binary Robust Independent Elementary Features]

M. Calonder et al., 2010

Spatial Arrangement of the Test Location



G I: Unifrom Distribution
G II: Gaussian Distribution, **p** and **q** are centered on origin
G III: Gaussian, **p** is centered on origin, **q** is centered on **p**
G IV: Random **p** and **q** on Coarse polar grid
G V: Random only **q** on coarse polar grid, **p** is set at origin