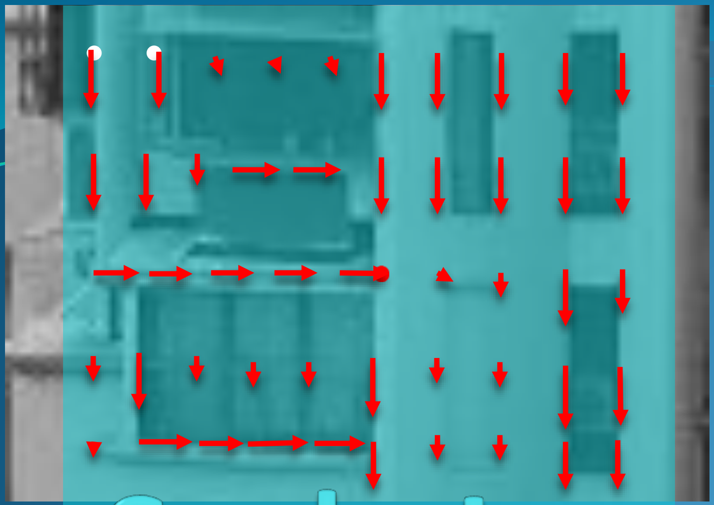
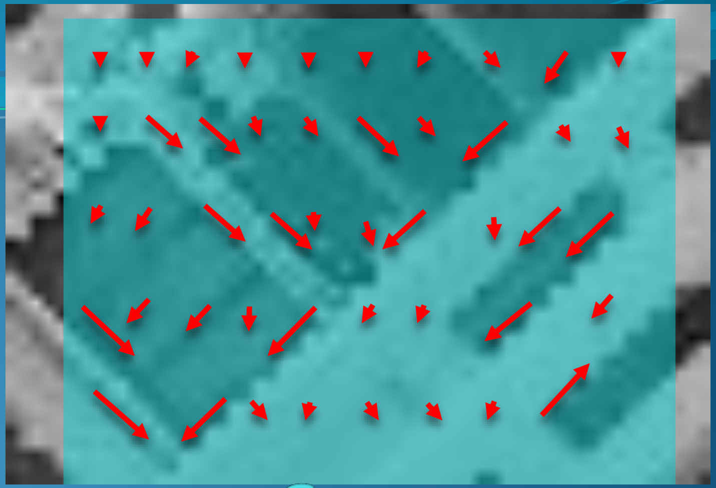
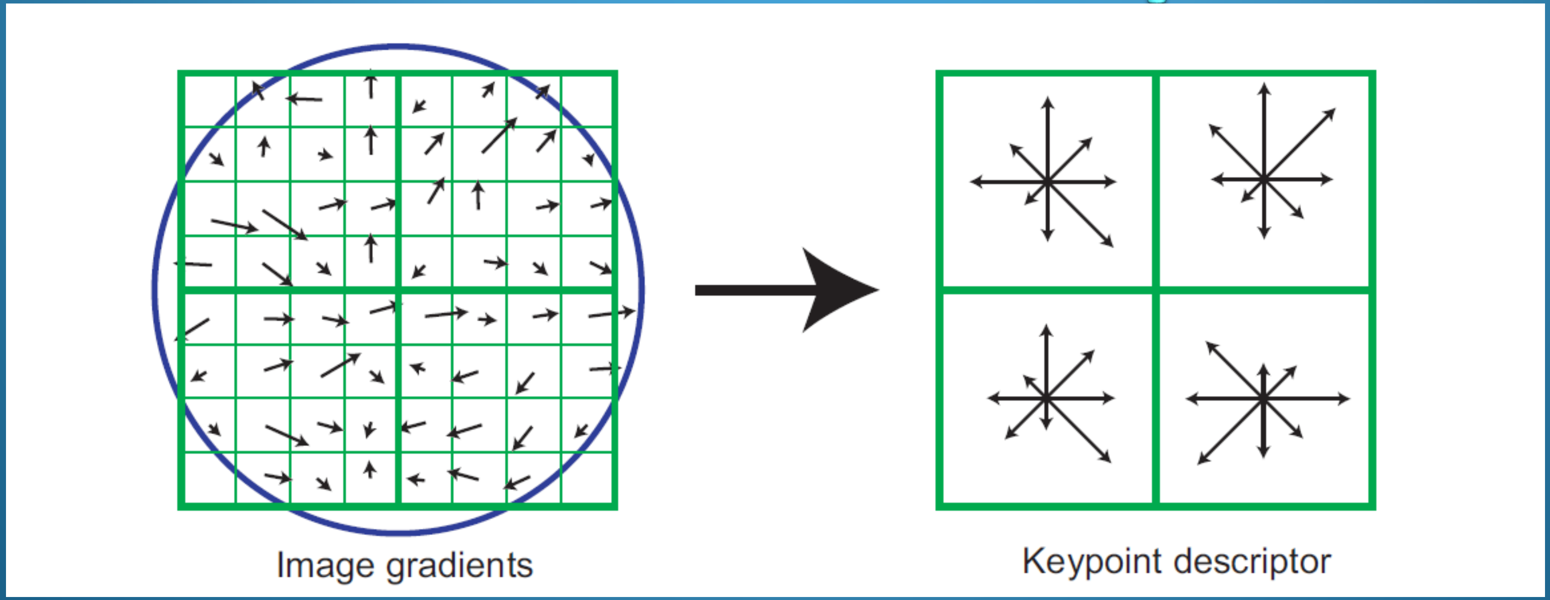


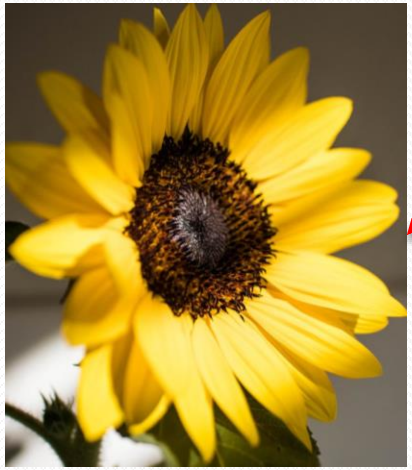


Scale Invariant Feature Transform

Feature Descriptor



Pratik Jain



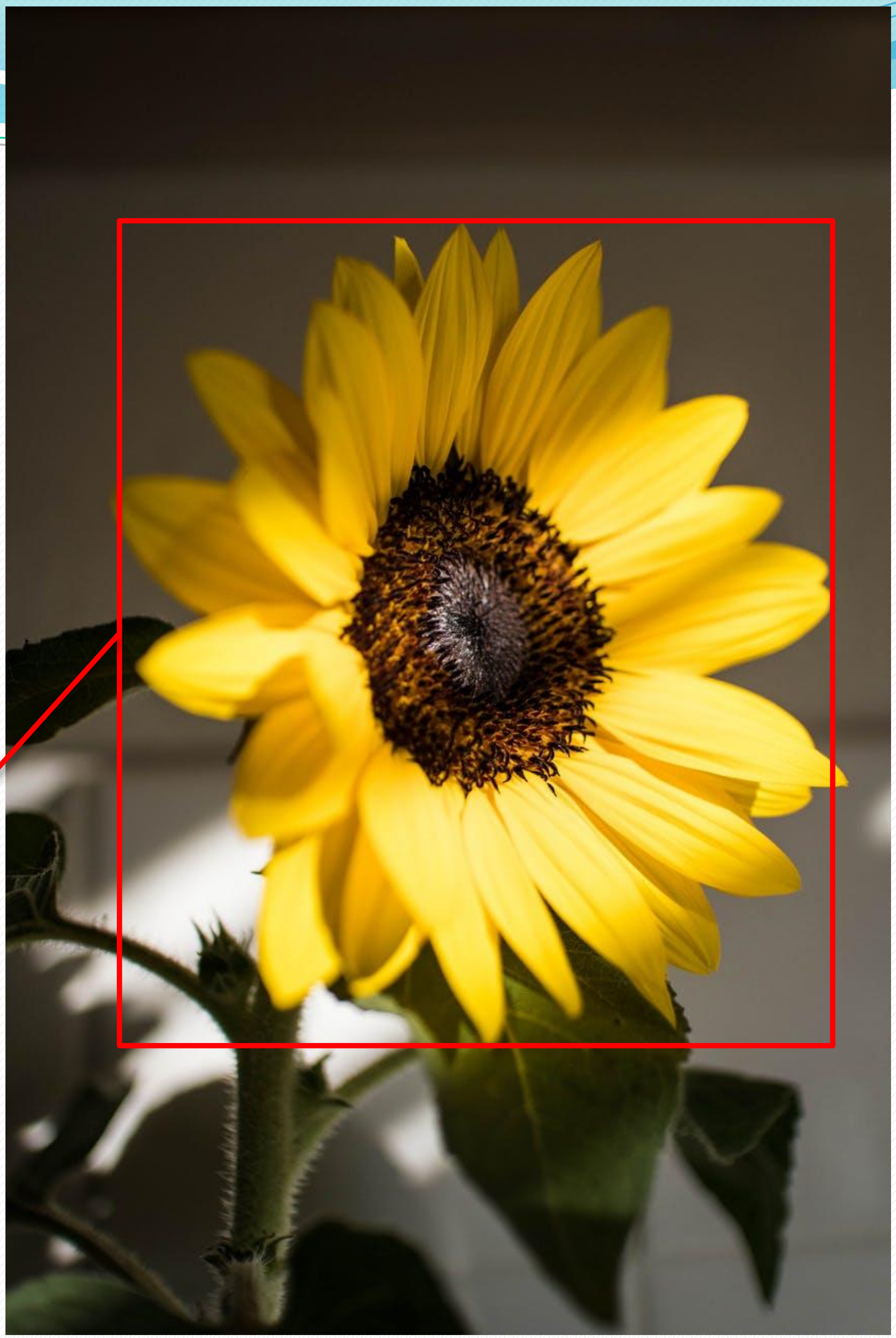


Photo by fotografierende from Pexels

Photo by Ana Arantes from Pexels

Photo by fotografierende from Pexels

Photo by Ana Arantes from Pexels

Blob Detection

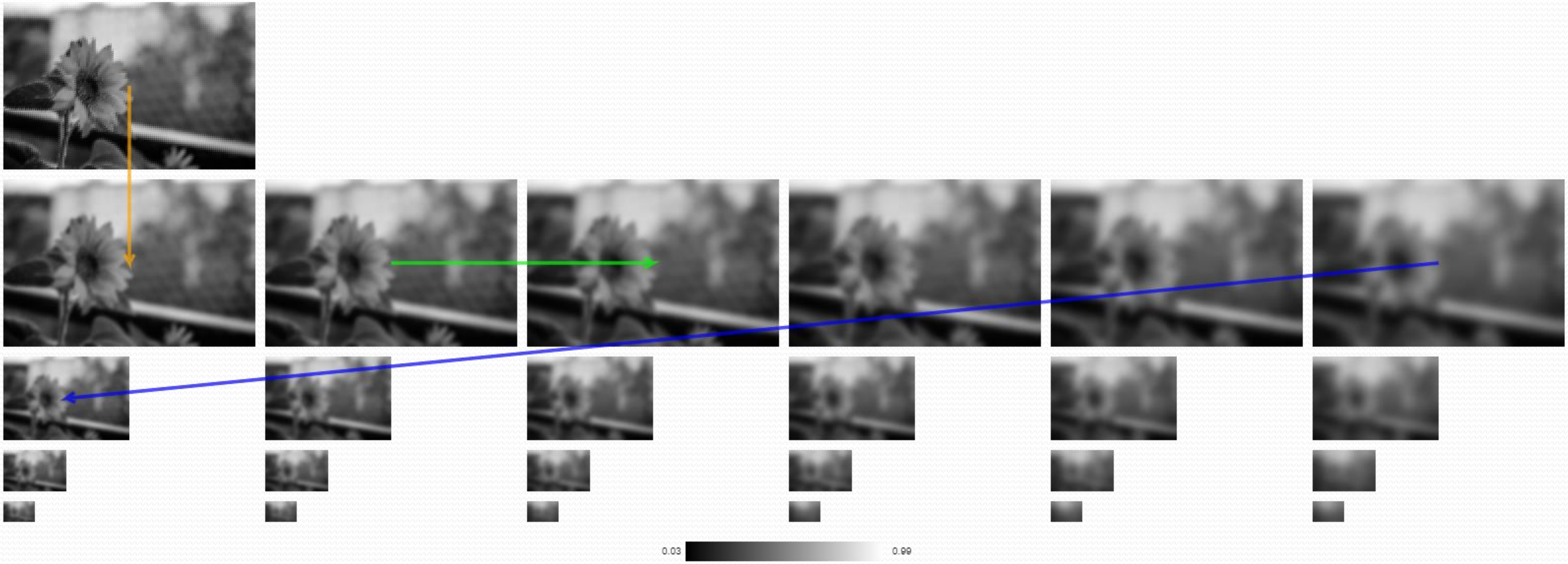


Image generated from : http://weitz.de/sift/index.html

Blob Detection

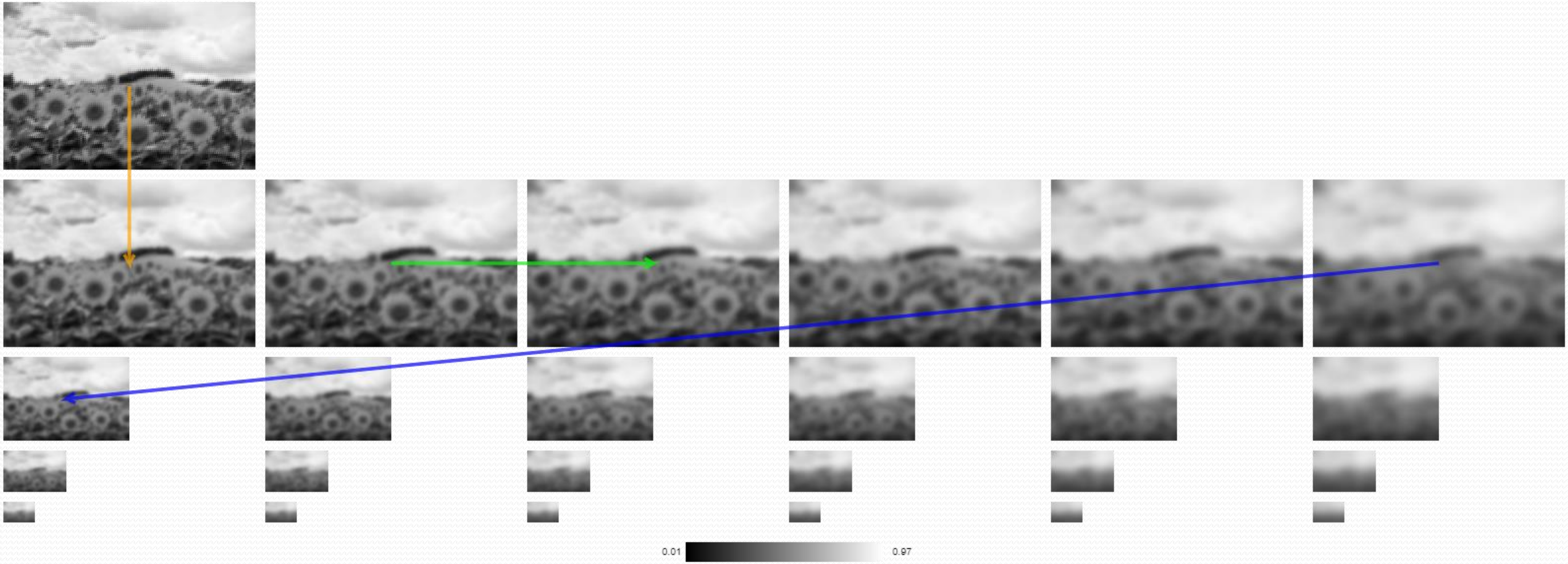


Image generated from : http://weitz.de/sift/index.html

Edge Detection

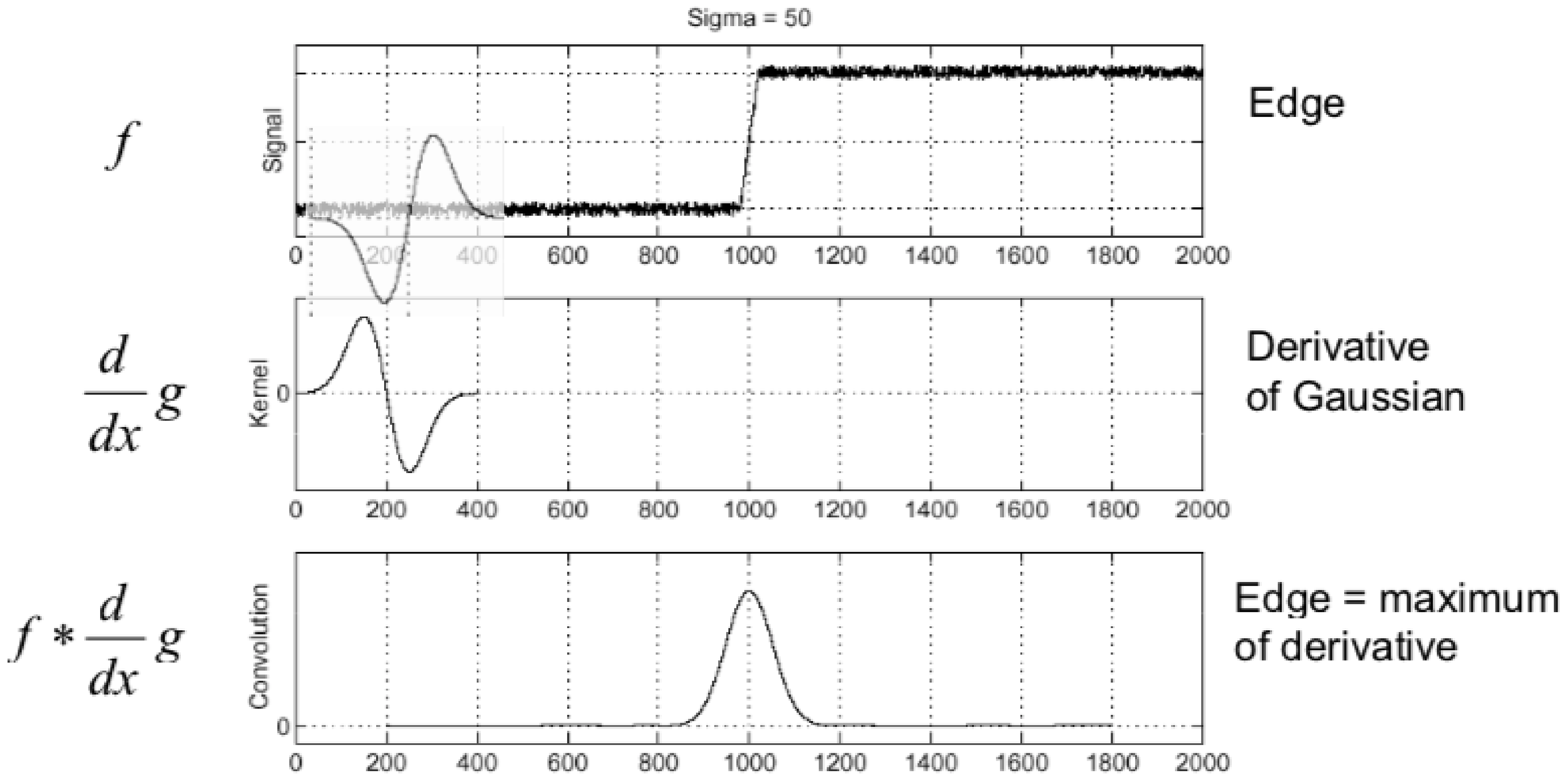


Image taken from https://towardsdatascience.com/sift-scale-invariant-feature-transform-c7233dc60f37

Edge Detection Take 2 (LoG)

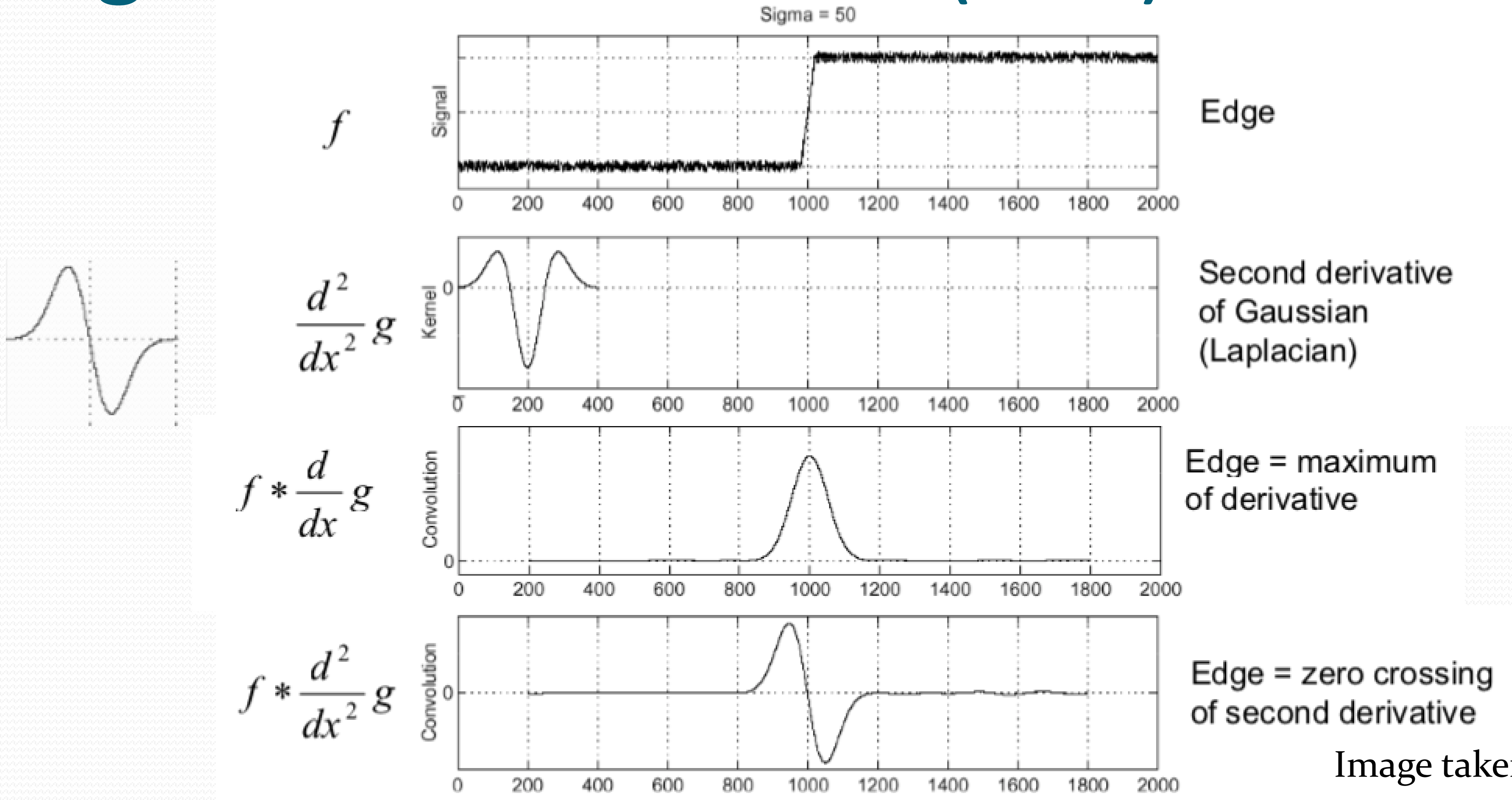
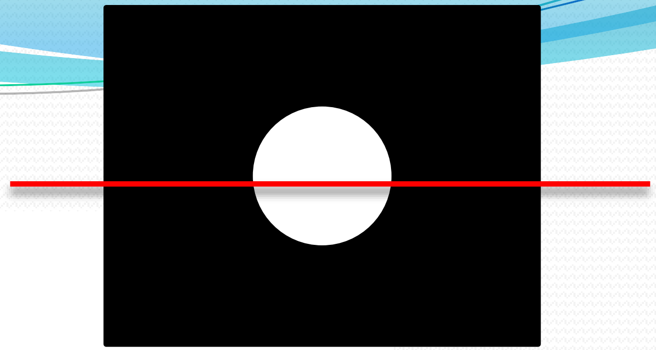
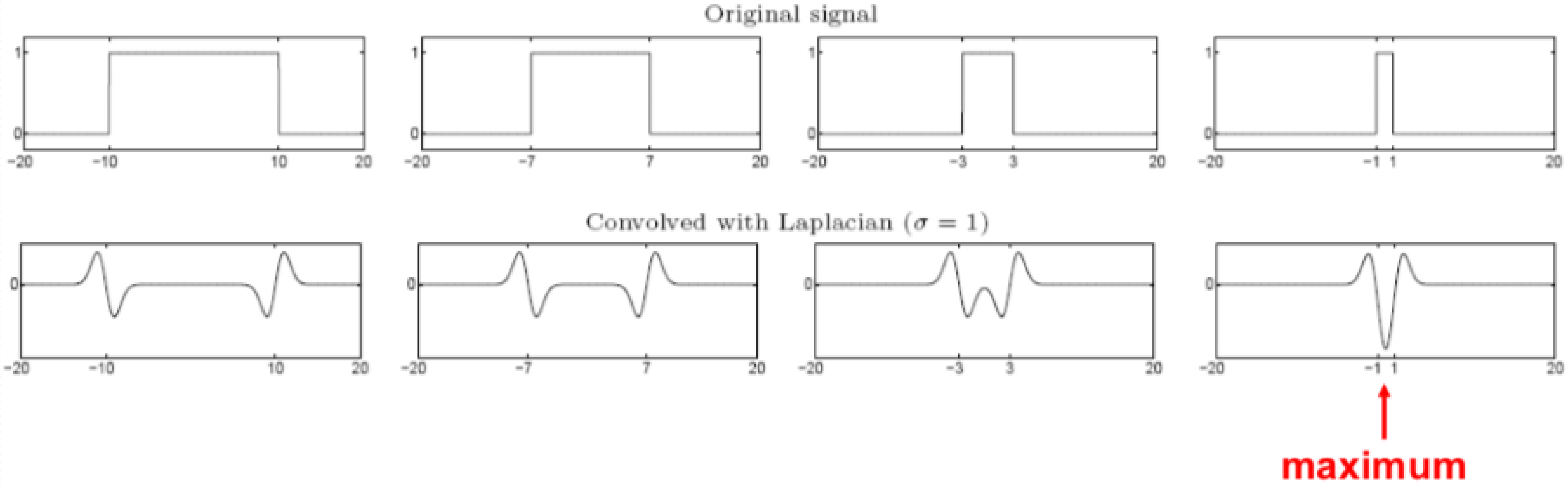


Image taken from here

Coming to the point

- Edge = ripple
- Blob = superposition of two ripples

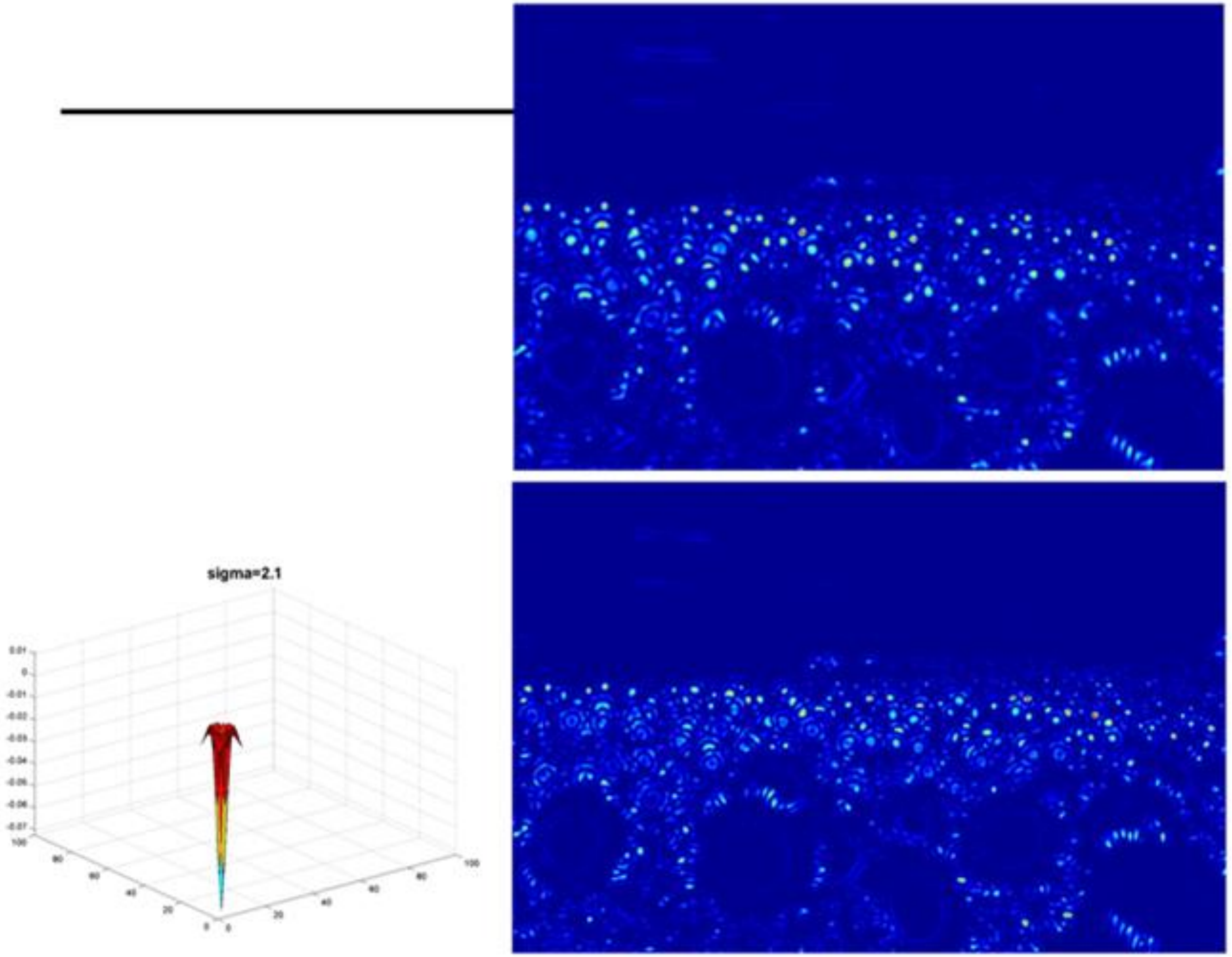




Spatial selection: the magnitude of the Laplacian response will achieve a maximum at the center of the blob, provided the scale of the Laplacian is "matched" to the scale of the blob

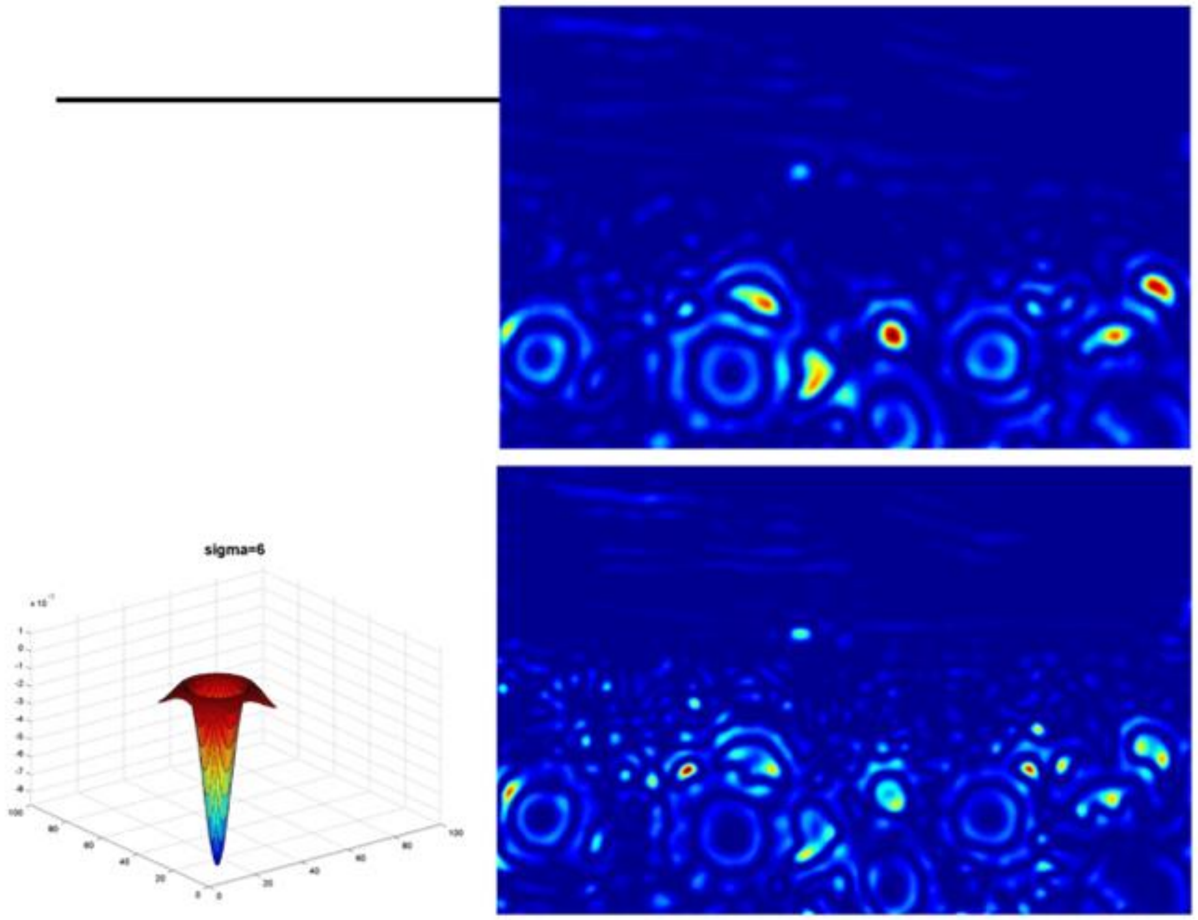
Image taken from here

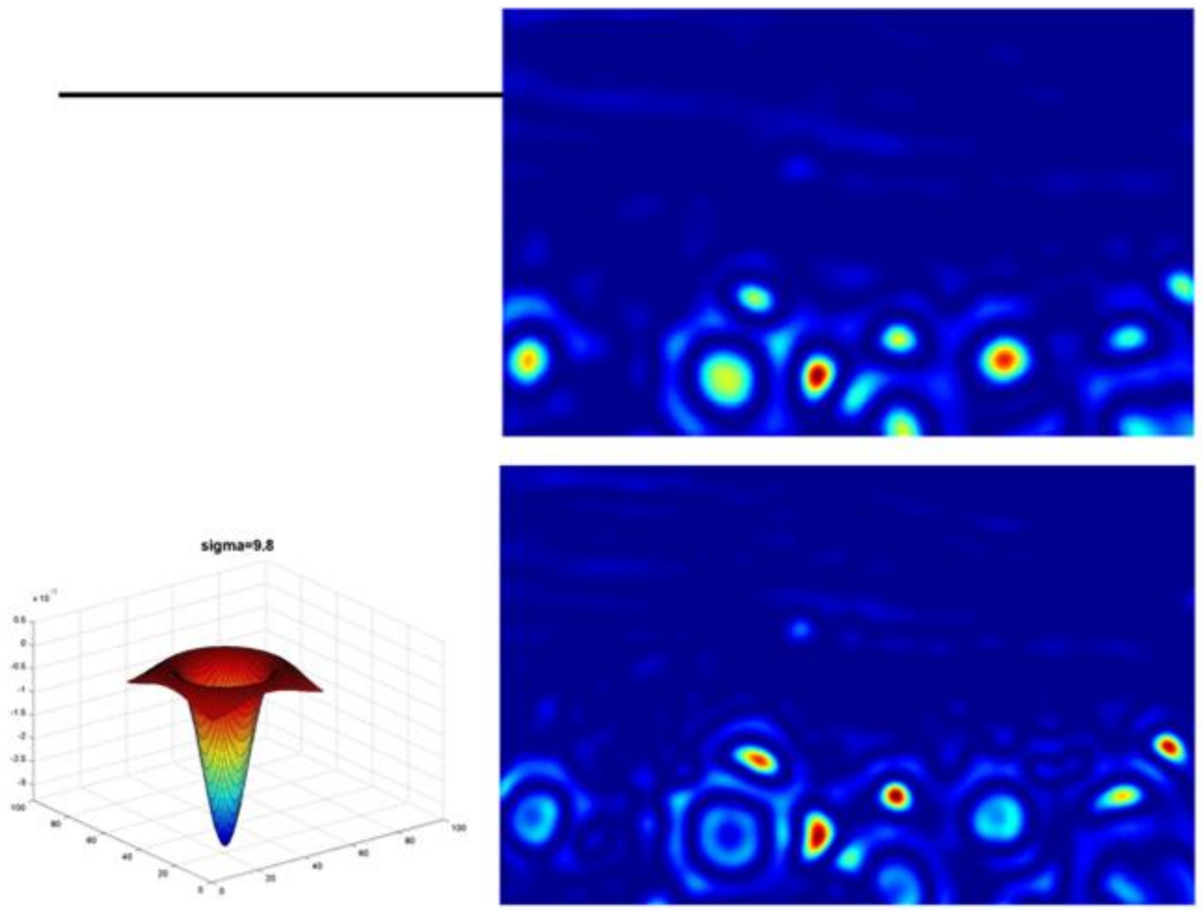
Blob Detection



Images from Kristen Grauman's slides

Blob Detection





Images from Kristen Grauman's slides

From LoG to DoG

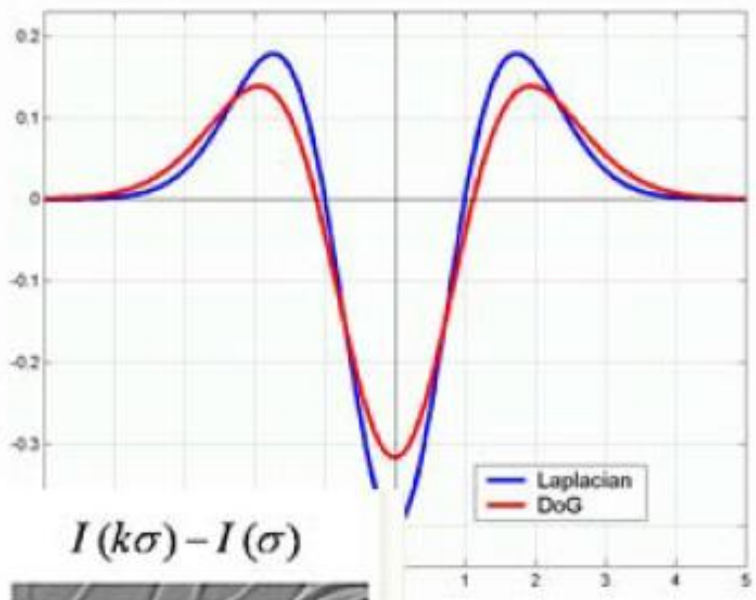
We can approximate the Laplacian with a difference of Gaussians; more efficient to implement.

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

(Laplacian)

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

(Difference of Gaussians)



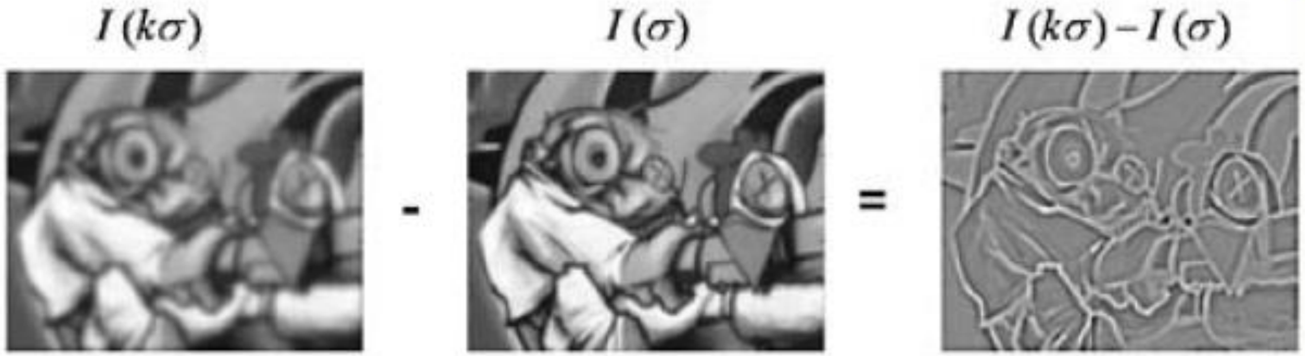
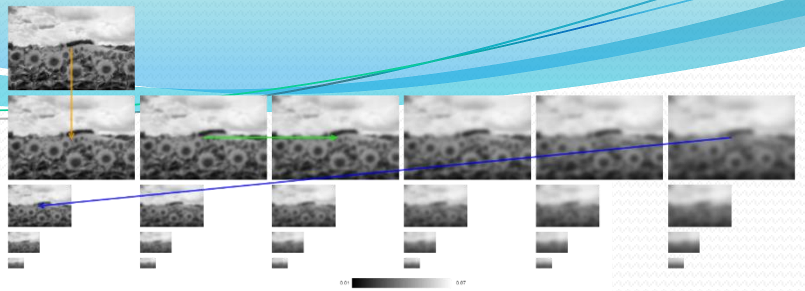
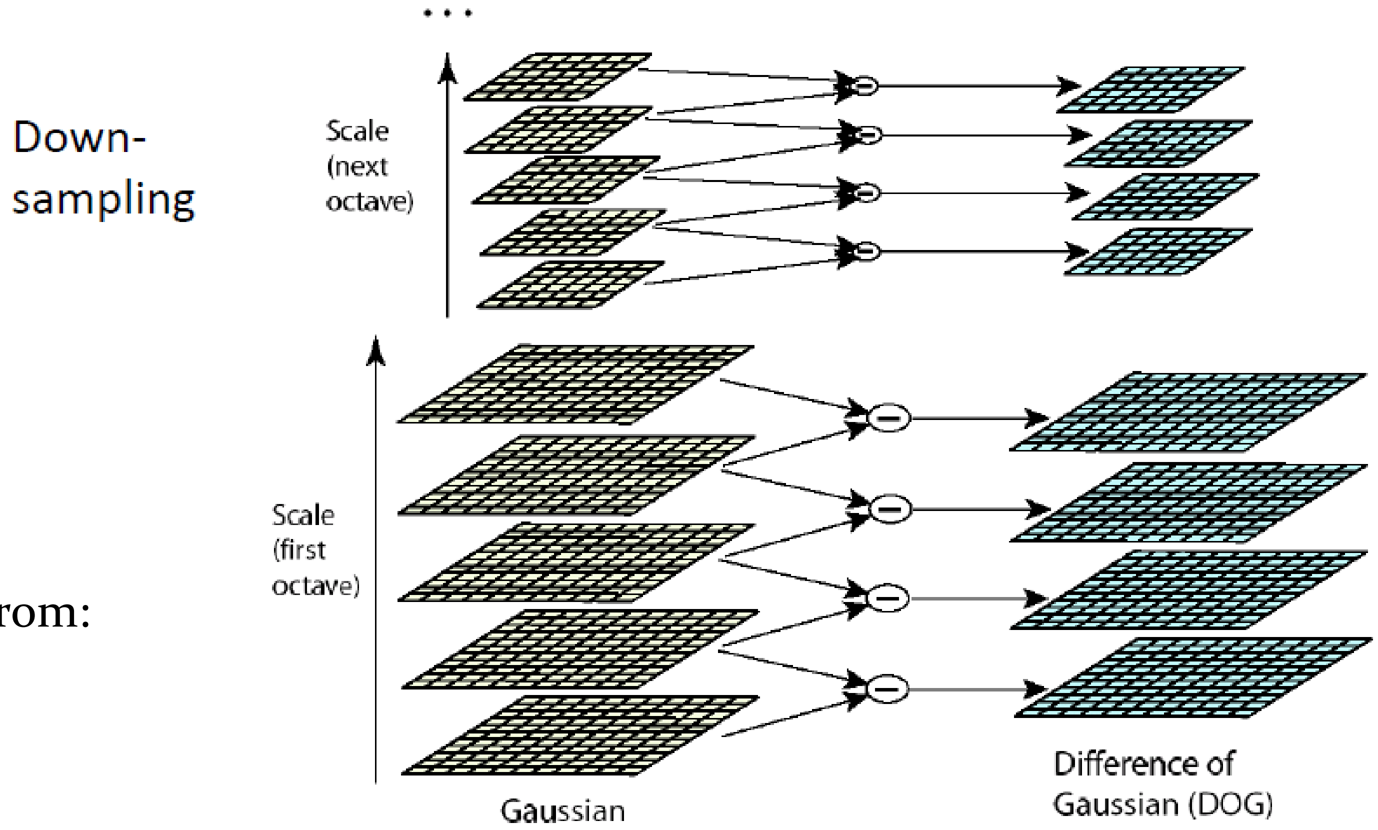


Image from Kristen Grauman's slides

The Scale Space





$$D(x,y,\sigma) = L(x,y,k\sigma) - L(x,y,\sigma)$$

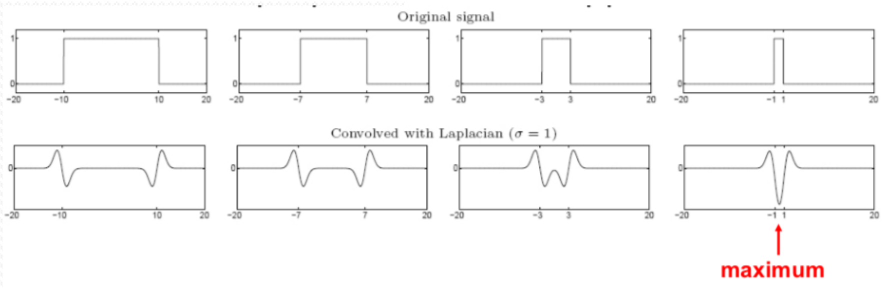
$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y)$$

Image taken from:
CS 763
Ajit Rajwade

Octave = doubling of σ_0. Within an octave, the adjacent scales differ by a constant factor k. If an octave contains s+1 images, then $k = 2^{(1/s)}$. The first image has scale σ_0, the second image has scale $k\sigma_0$, the third image has scale $k^2\sigma_0$, and the last image has scale $k^s\sigma_0$. Such a sequence of images convolved with Gaussians of increasing σ constitute a so-called **scale space**.

Scale Space Extrema detection

- Extract local extrema (i.e., minima or maxima) in DoG pyramid.
  - -Compare each point to its 8 neighbors at the same level, 9 neighbors in the level above, and 9 neighbors in the level below (i.e., 26 total).



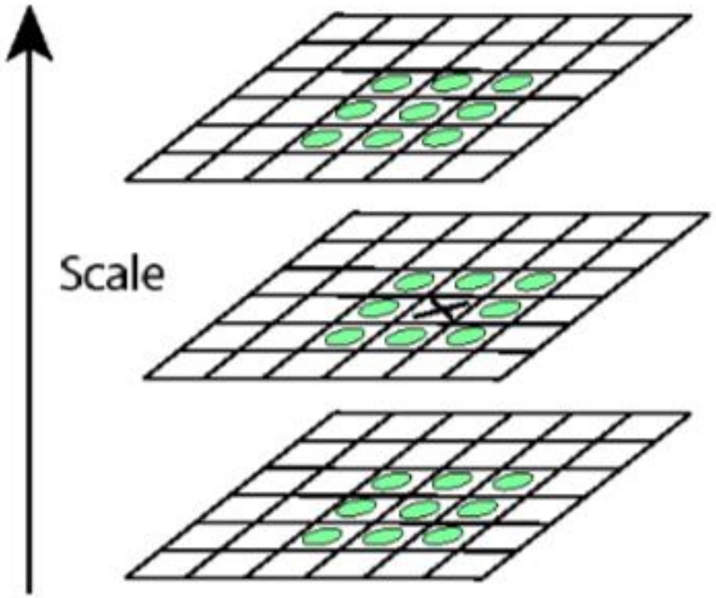


Image taken from:
[CS 763](CS 763)
Ajit Rajwade

Initial Points Detection

Image taken from [here](#) (Wikipedia)

CS 763
Ajit Rajwade

Feature Point Localization

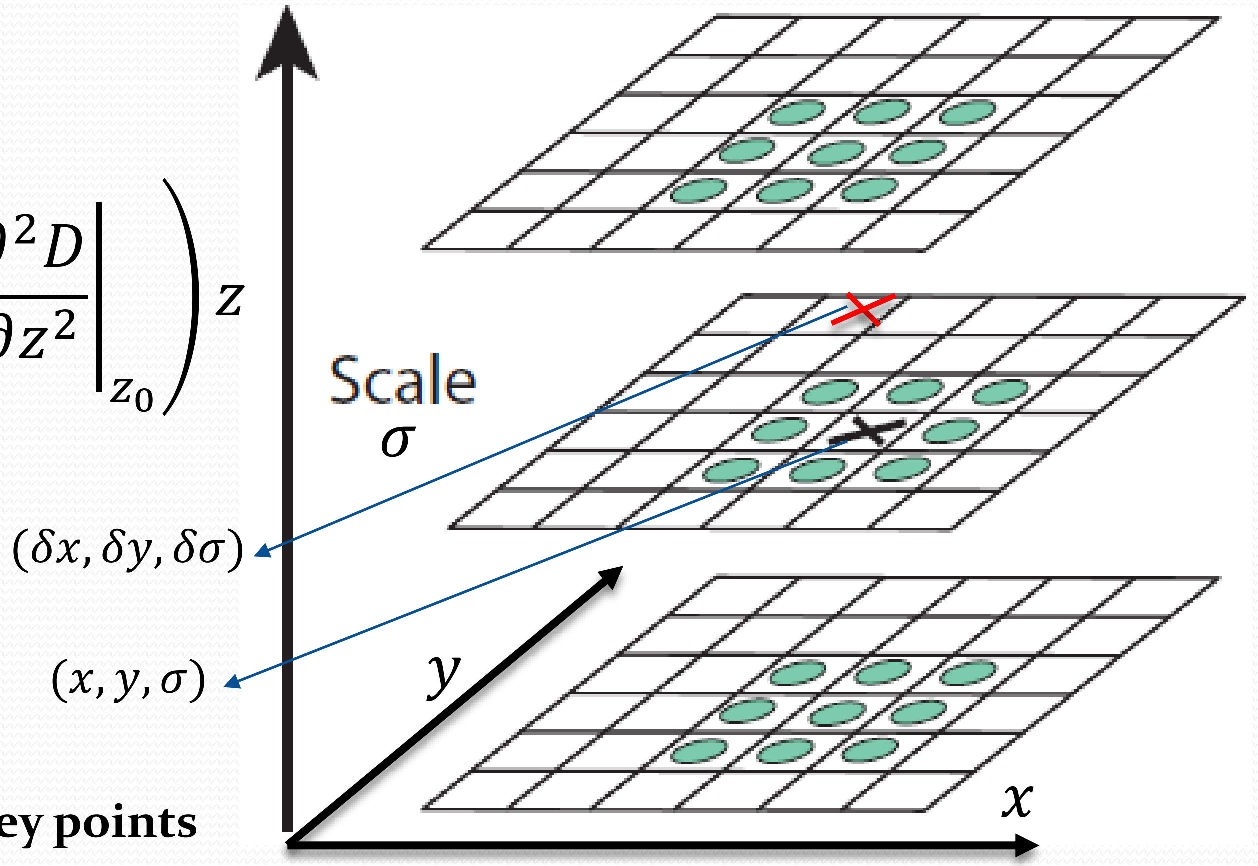
Interpolation using by fitting a Taylor expansion to fit a 3D quadratic surface (in x,y, and σ)

$$z_0 = [\, x, y, \sigma \,]^T \qquad z = [\, \delta x, \delta y, \delta\sigma \,]^T$$

$$D(z_0 + z) \approx D(Z_0) + \left(\frac{\partial D}{\partial z}\bigg|_{z_0} \right)^T z + \frac{1}{2} z^T \left(\frac{\partial^2 D}{\partial z^2}\bigg|_{z_0} \right) z$$

Finding the maxima or minima

$$\frac{\partial D(z_0 + z)}{\partial z} = 0$$



Distinctive Image Features from Scale-Invariant Key points
David G. Lowe

Feature Point Localization

$$D(z_0 + z) \approx D(Z_0) + \left(\left. \frac{\partial D}{\partial z} \right|_{z_0} \right)^T z + \frac{1}{2} z^T \left(\left. \frac{\partial^2 D}{\partial z^2} \right|_{z_0} \right) z$$

$$\frac{\partial D(z_0 + z)}{\partial z} = 0 + \left(\left. \frac{\partial D}{\partial z} \right|_{z_0} \right) + \frac{1}{2} \times 2 \times \left(\left. \frac{\partial^2 D}{\partial z^2} \right|_{z_0} \right) \hat{z} = 0$$

$$\left(\left. \frac{\partial^2 D}{\partial z^2} \right|_{z_0} \right) \hat{z} = - \left(\left. \frac{\partial D}{\partial z} \right|_{z_0} \right) \qquad \hat{z} = - \left(\left. \frac{\partial^2 D}{\partial z^2} \right|_{z_0} \right)^{-1} \left(\left. \frac{\partial D}{\partial z} \right|_{z_0} \right)$$

Feature Point Localization

$$D(z_0 + z) \approx D(Z_0) + \left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)^T z + \frac{1}{2} z^T \left(\frac{\partial^2 D}{\partial z^2}\bigg|_{z_0}\right) z$$

$$\hat{z} = -\left(\frac{\partial^2 D}{\partial z^2}\bigg|_{z_0}\right)^{-1} \left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)$$

$$D(z_0 + \hat{z}) \approx D(Z_0) + \left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)^T \hat{z} + \frac{1}{2}\left(-\left(\frac{\partial^2 D}{\partial z^2}\bigg|_{z_0}\right)^{-1}\left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)\right)^T \left(\frac{\partial^2 D}{\partial z^2}\bigg|_{z_0}\right)\hat{z}$$

$$D(z_0 + \hat{z}) \approx D(Z_0) + \left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)^T \hat{z} - \frac{1}{2}\left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)^T \left(\left(\frac{\partial^2 D}{\partial z^2}\bigg|_{z_0}\right)^{-1}\right)^T \left(\frac{\partial^2 D}{\partial z^2}\bigg|_{z_0}\right)\hat{z}$$

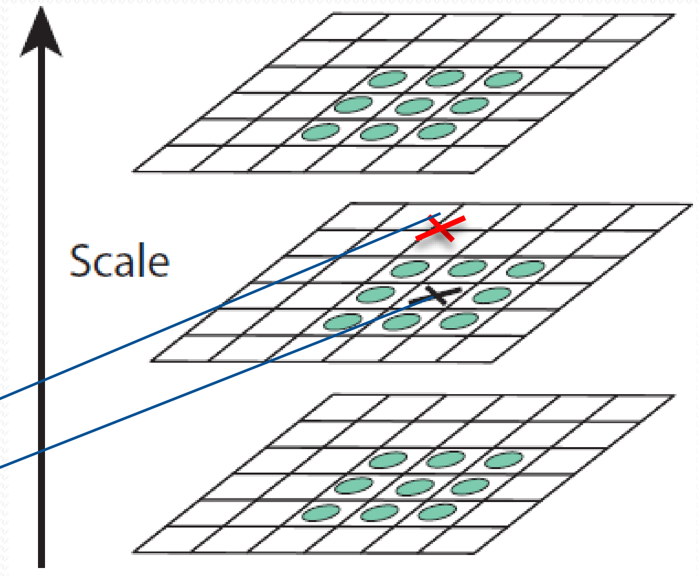
Hessian matrix

Feature Point Localization

$$D(z_0+\hat{z}) \approx D(Z_0) + \left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)^T \hat{z} - \frac{1}{2}\left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)^T \left(\underbrace{\left(\frac{\partial^2 D}{\partial z^2}\bigg|_{z_0}\right)^{-1}}_{\text{Hessian matrix}}\right)^T \left(\frac{\partial^2 D}{\partial z^2}\bigg|_{z_0}\right)\hat{z}$$

$$D(z_0+\hat{z}) \approx D(Z_0) + \left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)^T \hat{z} - \frac{1}{2}\left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)^T \hat{z}$$

$$D(z_0+\hat{z}) \approx D(Z_0) + \frac{1}{2}\left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)^T \hat{z}$$



$z = (\delta x, \delta y, \delta\sigma)$

$z_0 = (x, y, \sigma)$

Distinctive Image Features from Scale-Invariant Key points
David G. Lowe

Feature Point Localization (Summary)

For Every extrema point we calculate $\hat{z}$

$$A = \begin{bmatrix} \frac{\partial D}{\partial x} & \frac{\partial D}{\partial y} & \frac{\partial D}{\partial \sigma} \end{bmatrix}^T$$

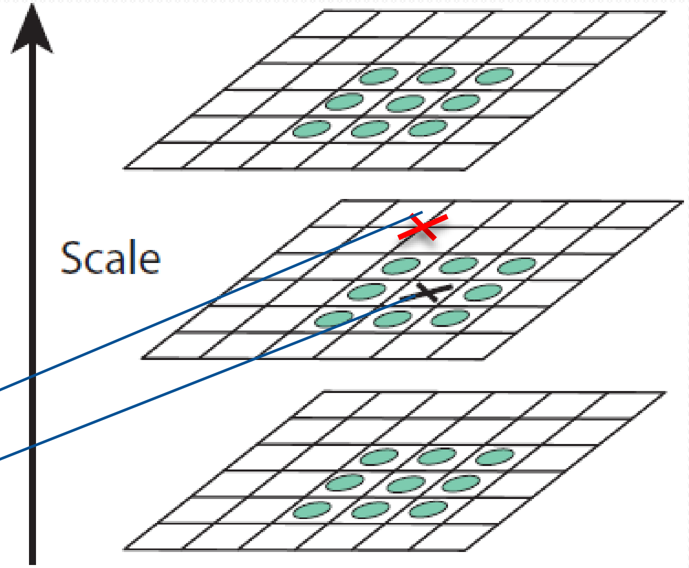
$$\hat{z} = [\delta x, \delta y, \delta\sigma]^T = -\left(\overbrace{\frac{\partial^2 D}{\partial z^2}\bigg|_{z_0}}^{AA^T}\right)^{-1} \left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)$$

Distinctive Image Features from Scale-Invariant Key points
David G. Lowe

If $\delta x, \delta y$ are greater than 0.5 then we change z_0. E.g. if $\delta x > 0.5$,$x = x + 1$

Again $\hat{z}$ will be calculated at that point.

If a pixel keeps on oscillating, we discard it saying its unstable.



Pixel value at that point is calculated

$$D(z_0 + \hat{z}) \approx D(Z_0) + \frac{1}{2}\left(\frac{\partial D}{\partial z}\bigg|_{z_0}\right)^T \hat{z}$$

$z = (\delta x, \delta y, \delta\sigma)$

$z_0 = (x, y, \sigma)$

If this D is less than 0.03 it is discarded saying it's a low contrast point.(All pixels are normalized between [0,1])

After Feature Point Localization and thresholding

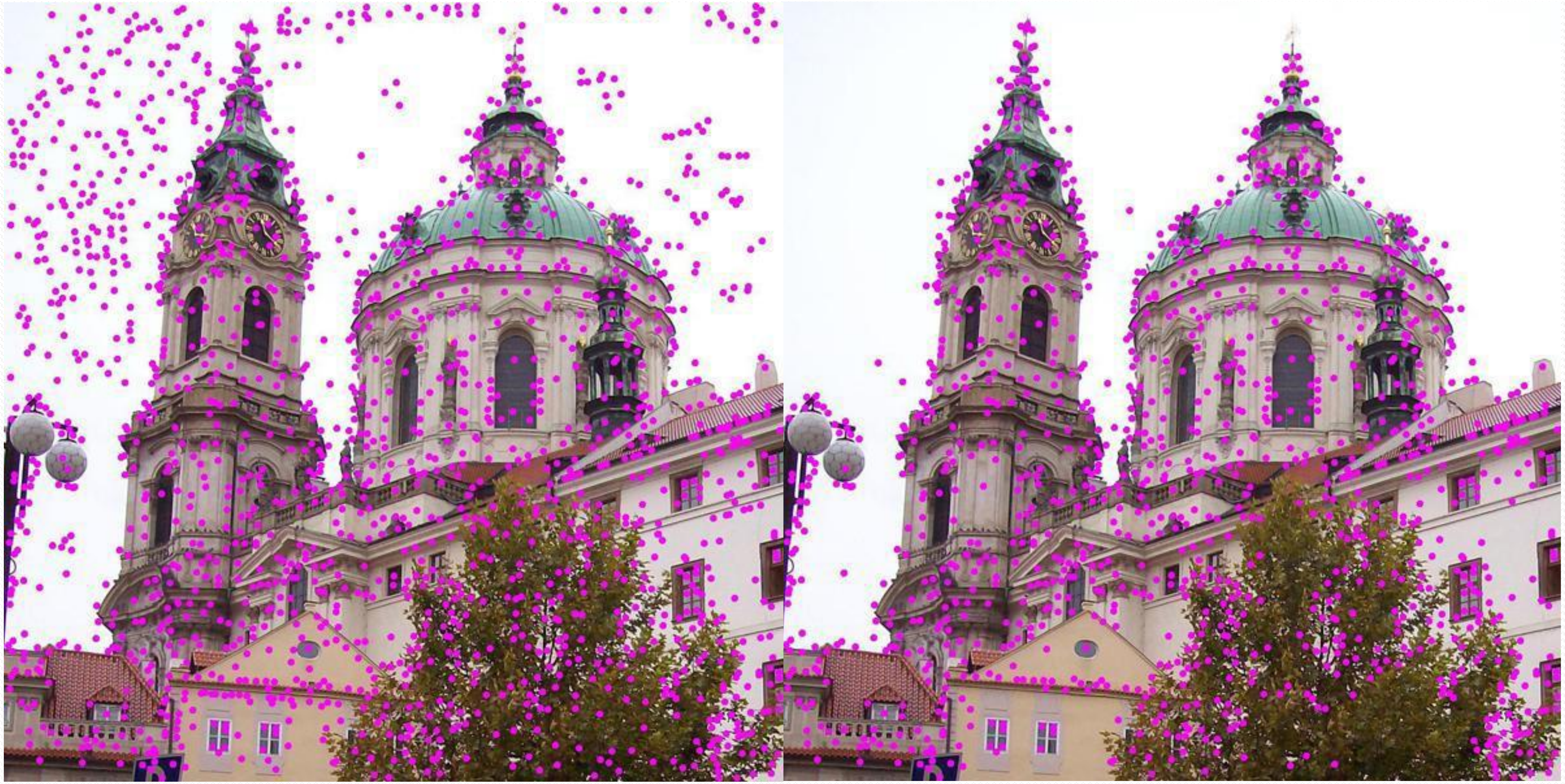


Image taken from here (Wikipedia)

Removing points on the edges

The principal curvatures can be computed from a 2x2 Hessian matrix, H, computed at the location and scale of the key point

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

α *and* β *are eigen values of the matrix*

$$\frac{\operatorname{Tr}(\mathbf{H})^2}{\operatorname{Det}(\mathbf{H})} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(r\beta+\beta)^2}{r\beta^2} = \frac{(r+1)^2}{r},$$

Keep only values that follow this condition

$$\frac{\operatorname{Tr}(\mathbf{H})^2}{\operatorname{Det}(\mathbf{H})} < \frac{(r+1)^2}{r}.$$

r=10

After removal of edges points

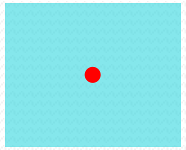
Image taken from here (Wikipedia)

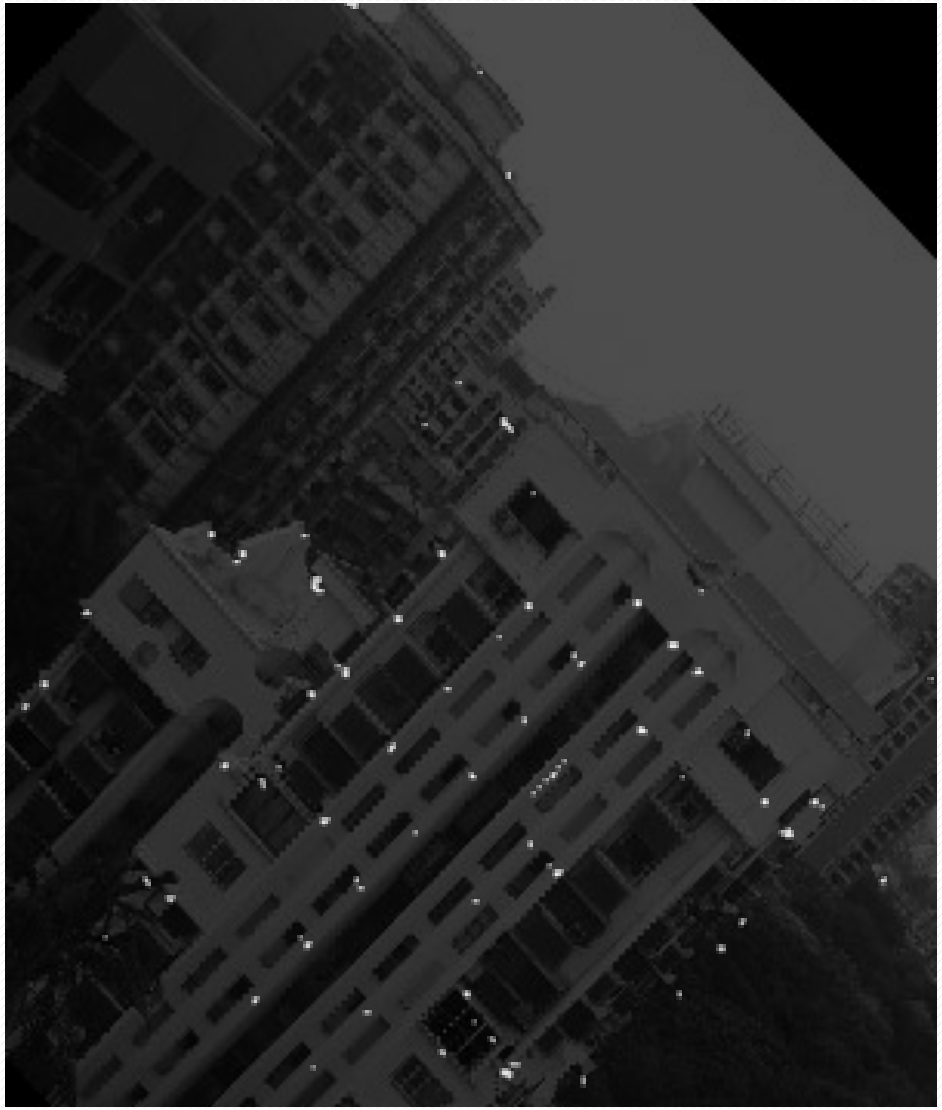
Rotational Invariance

Good Detector
But we also need good descriptor

Repeatability Rate

$$r = \frac{No.of\ corresponding\ points\ detected\ in\ common\ region}{Total\ no.of\ points\ detected\ in\ common\ region}$$





Orientation Assignment

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2}$$

$$\theta(x,y) = \tan^{-1}((L(x,y+1) - L(x,y-1))/(L(x+1,y) - L(x-1,y)))$$

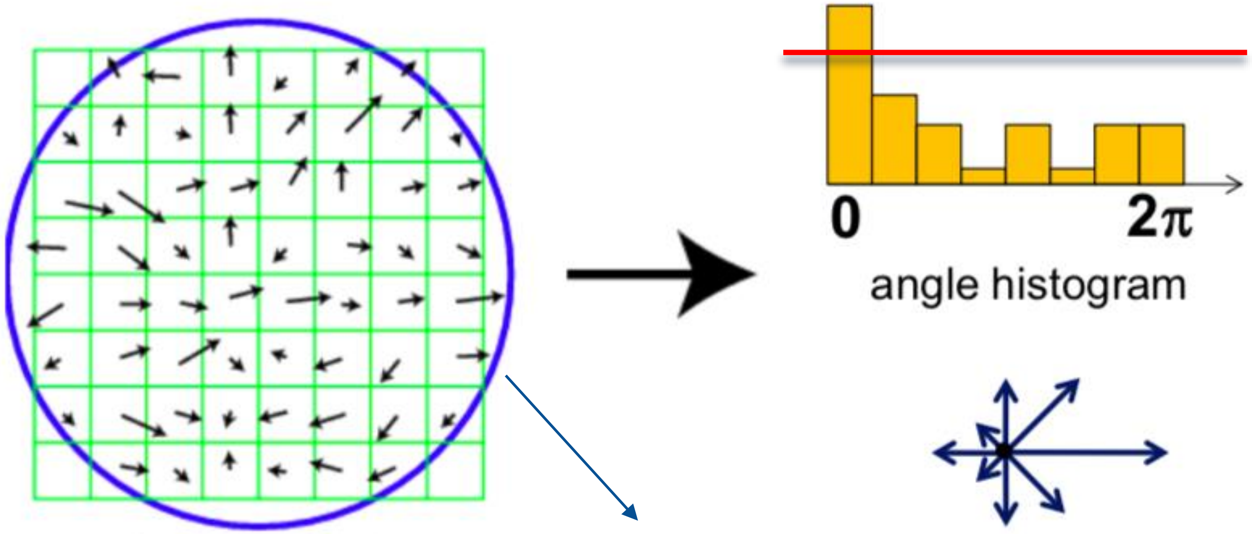


Image gradients

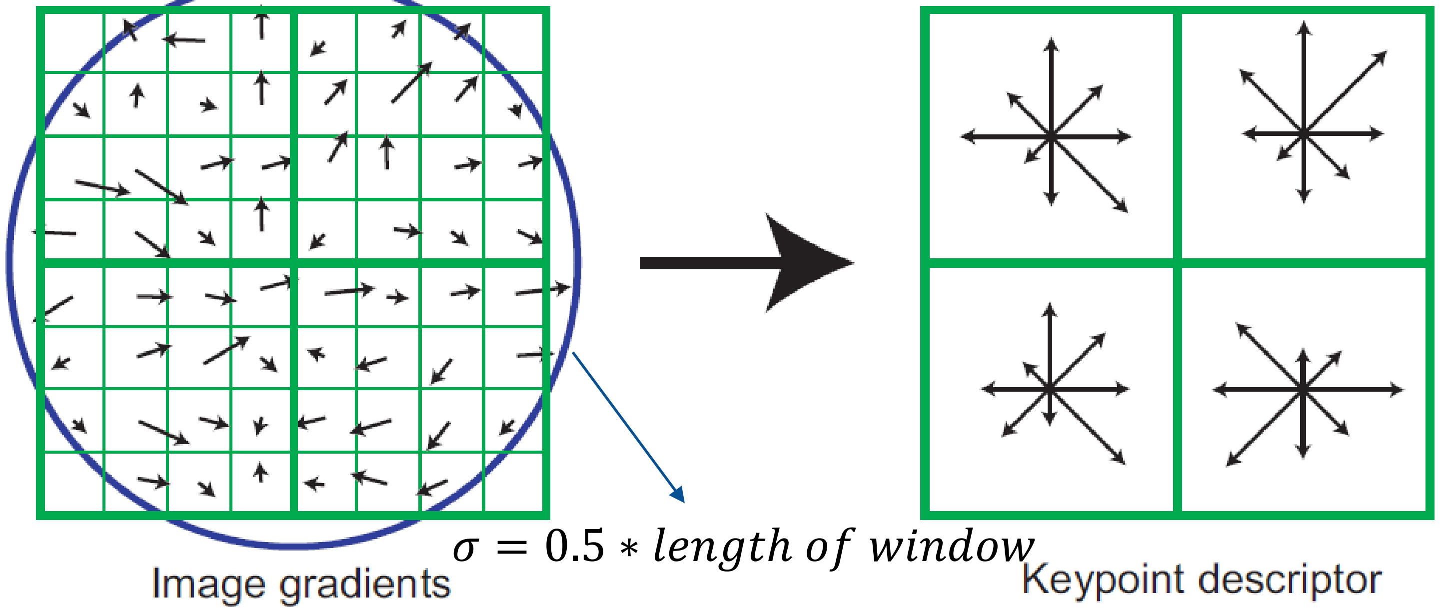
$\sigma = 1.5 * scale\ of\ image$

Histogram of gradient orientation – the bin-counts are weighted by gradient magnitudes and a Gaussian weighting function. Usually, 36 bins are chosen for the orientation.

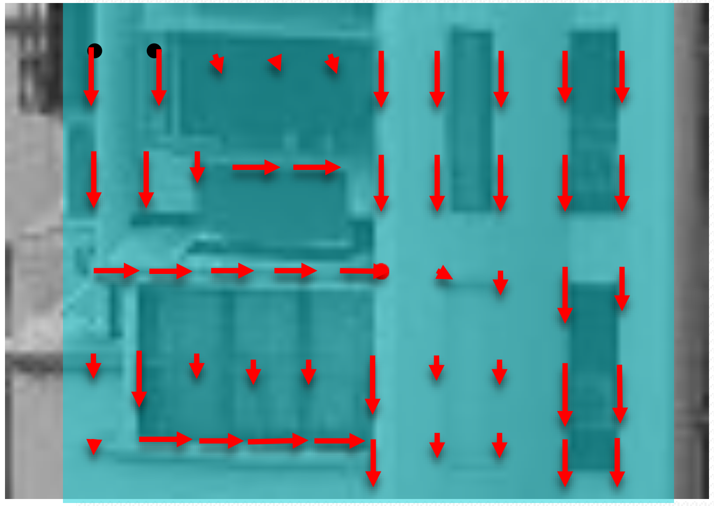
Image taken from:
[CS 763](CS 763)
Ajit Rajwade

Descriptor for Each feature

The next step is to compute a descriptor for the local image region that is highly distinctive yet is as invariant as possible to remaining variations, such as change in illumination or 3D viewpoint.
In order to achieve orientation invariance, the coordinates of the descriptor and the gradient orientations are rotated relative to the key point orientation.



$\sigma = 0.5 * length\ of\ window$



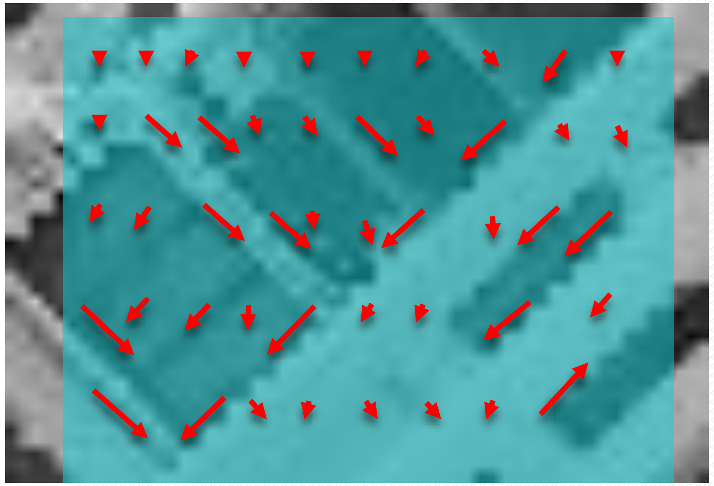
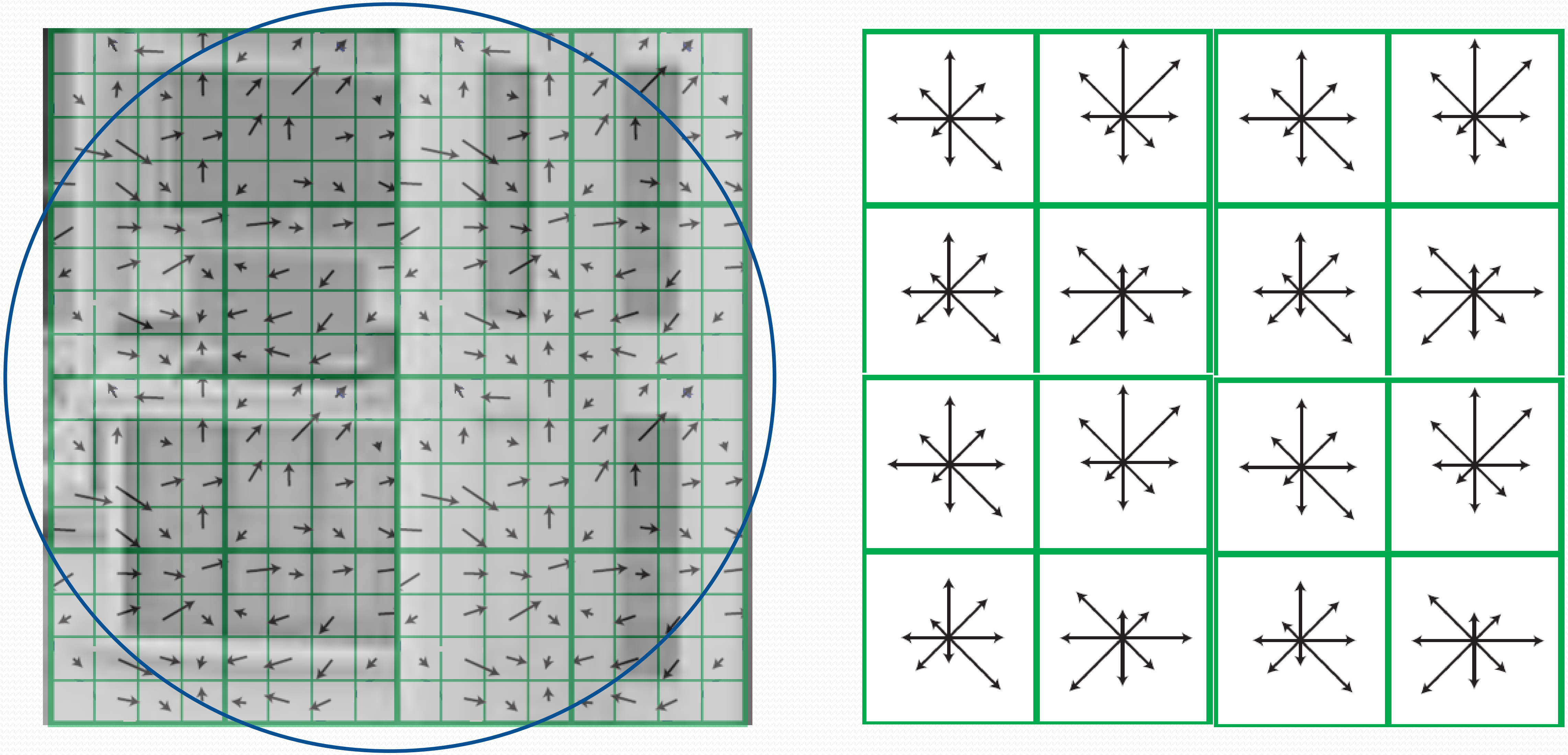


Figure 7: A keypoint descriptor is created by first computing the gradient magnitude and orientation at each image sample point in a region around the keypoint location, as shown on the left. These are weighted by a Gaussian window, indicated by the overlaid circle. These samples are then accumulated into orientation histograms summarizing the contents over 4x4 subregions, as shown on the right, with the length of each arrow corresponding to the sum of the gradient magnitudes near that direction within the region. This figure shows a 2x2 descriptor array computed from an 8x8 set of samples, whereas the experiments in this paper use 4x4 descriptors computed from a 16x16 sample array.

Distinctive Image Features from Scale-Invariant Key points: David G. Lowe

SIFT Descriptor



Object Detection

Figure 12: The training images for two objects are shown on the left. These can be recognized in a cluttered image with extensive occlusion, shown in the middle. The results of recognition are shown on the right. A parallelogram is drawn around each recognized object showing the boundaries of the original training image under the affine transformation solved for during recognition. Smaller squares indicate the keypoints that were used for recognition.

Distinctive Image Features from Scale-Invariant Key points: David G. Lowe