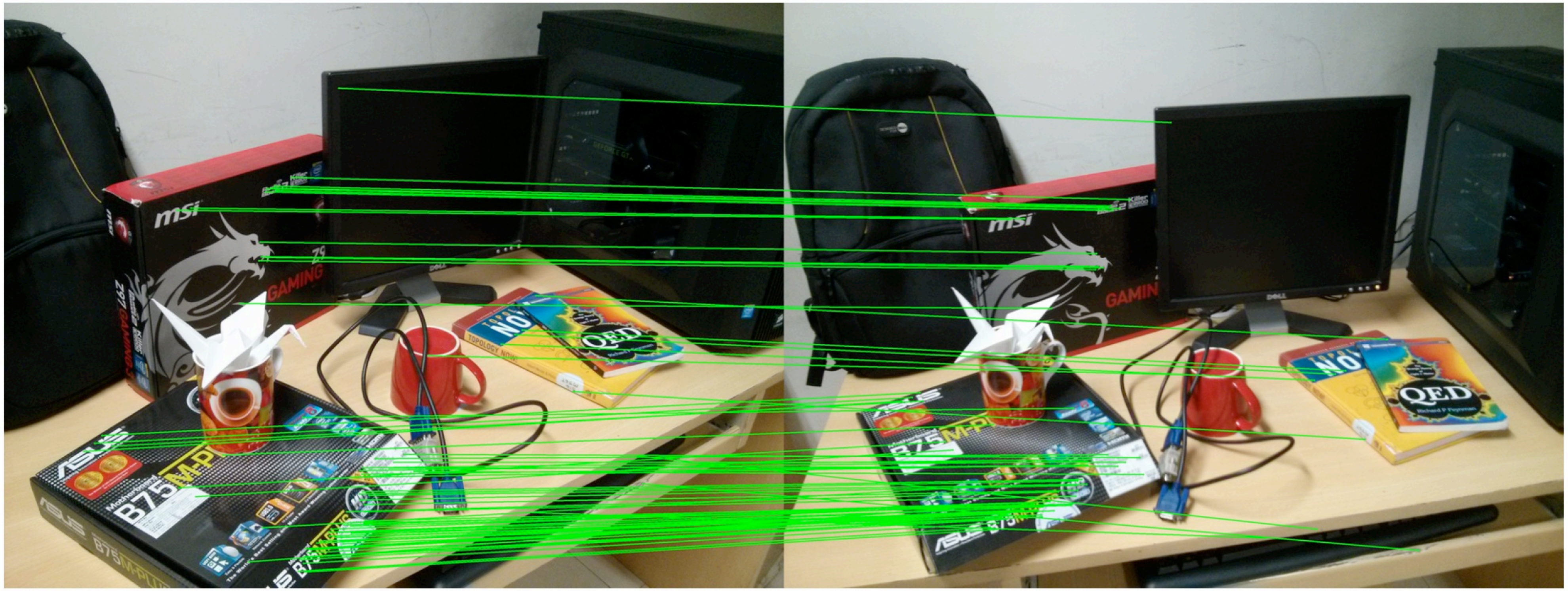
Feature Detection and Matching: Detectors and Descriptors II

CS 6384 Computer Vision
Professor Yapeng Tian
Department of Computer Science

Slides borrowed from Professor Yu Xiang.

Feature Detection and Matching

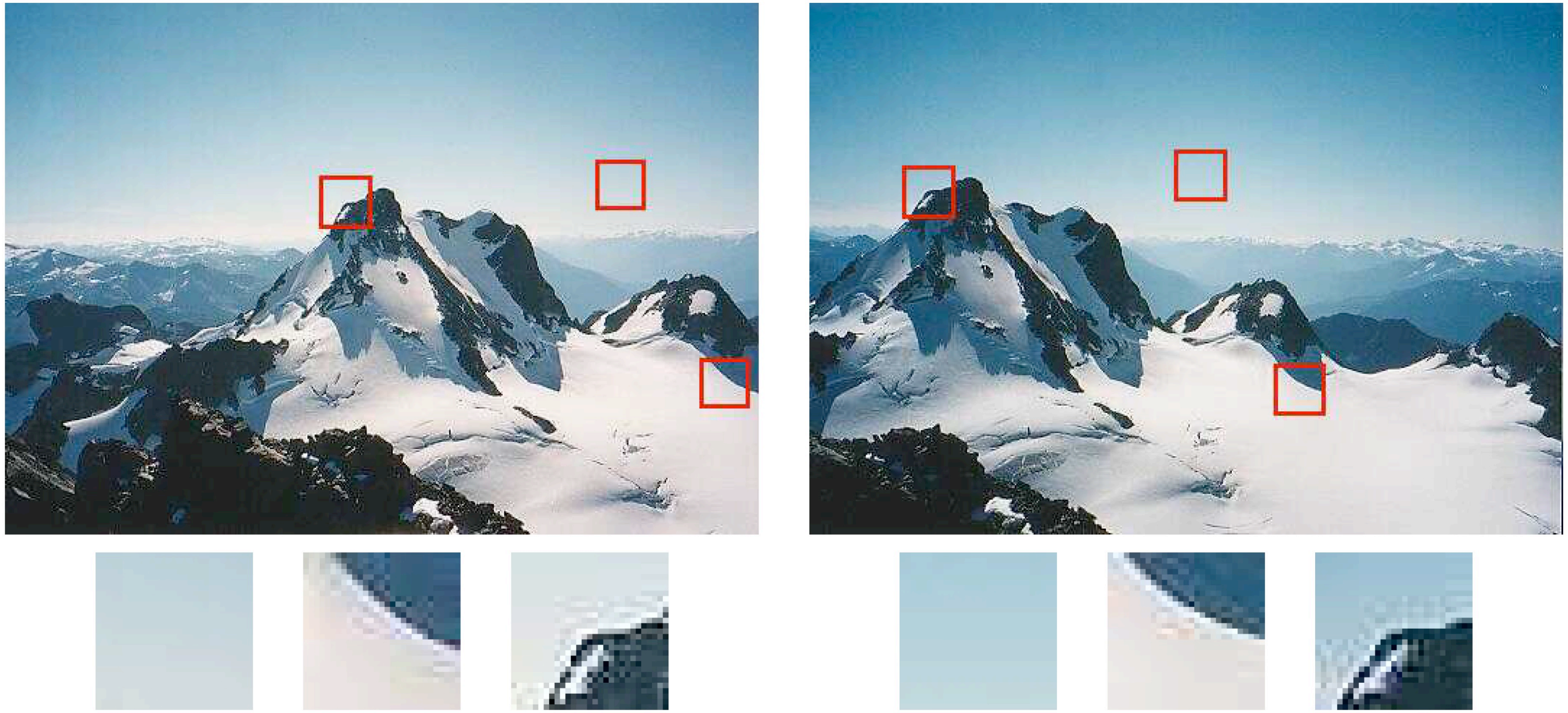


Geometry-aware Feature Matching for Structure from Motion Applications. Shah et al, WACV'15

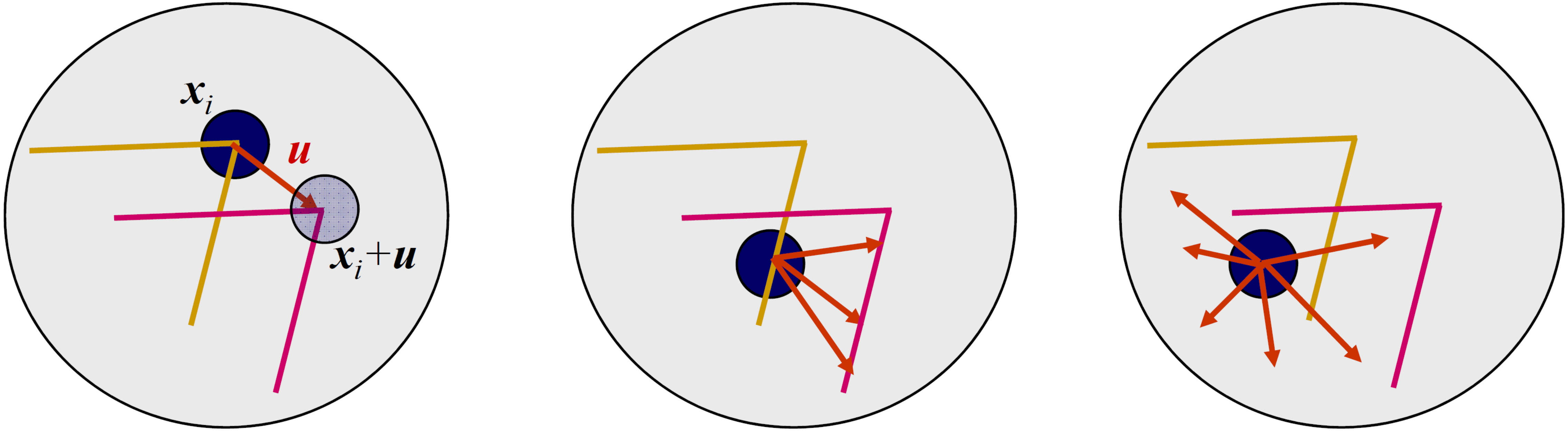
Applications: stereo matching, image stitching, 3D reconstruction, camera pose estimation, object recognition

Feature Detectors

How to find image locations that can be reliably matched with images?



Feature Detectors



(a) Corner

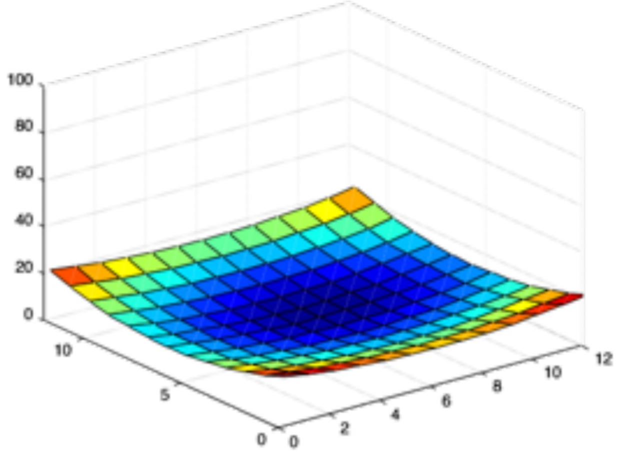
(b) Edge

(c) Textureless region

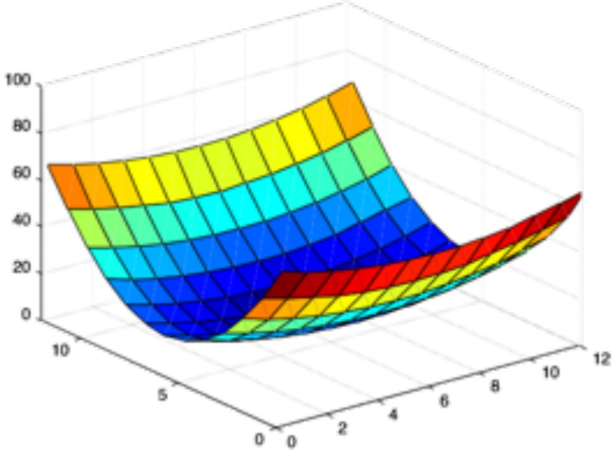
Harris Corner Detector

$$f(\Delta x, \Delta y) \approx \sum_{x,y} w(x,y)(I_x(x,y)\Delta x + I_y(x,y)\Delta y)^2$$

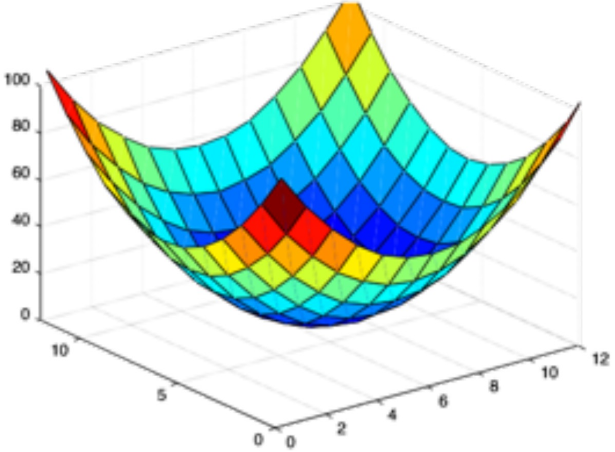
$$f(\Delta x, \Delta y) \approx \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} M \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \qquad M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = \begin{bmatrix} \sum_{x,y} w(x,y) I_x^2 & \sum_{x,y} w(x,y) I_x I_y \\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



Flat



Edge



Corner

Invariance

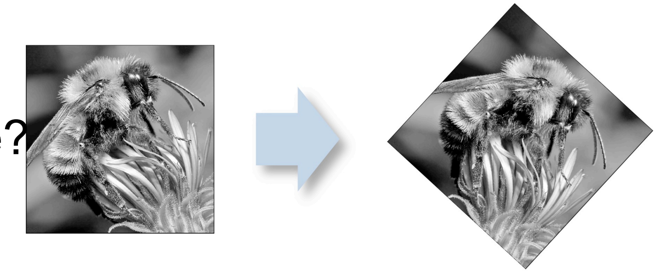
Can the same feature point be detected after some transformation?

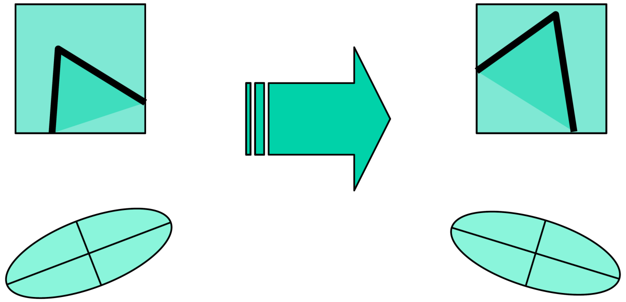
- Translation invariance

  Are Harris corners translation invariance?

- 2D rotation invariance

  Are Harris corners rotation invariance?

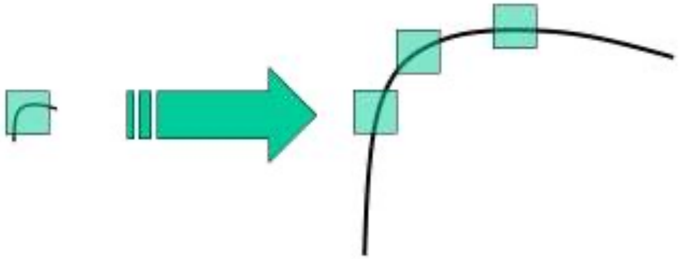


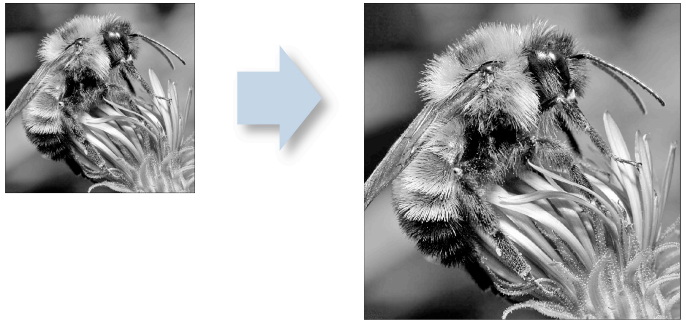


- Scale invariance

  Are Harris corners scale invariance?

No





Scale Invariance

Solution 1: detection features in all scales, matching features in corresponding scale (for small scale change)

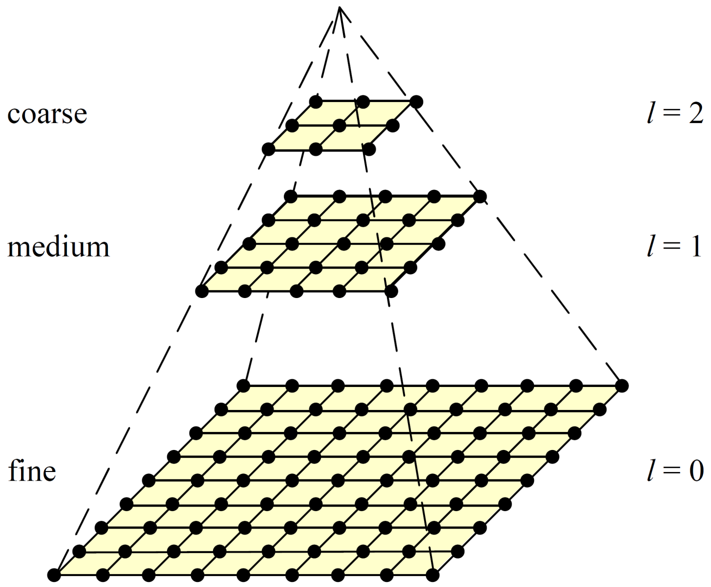
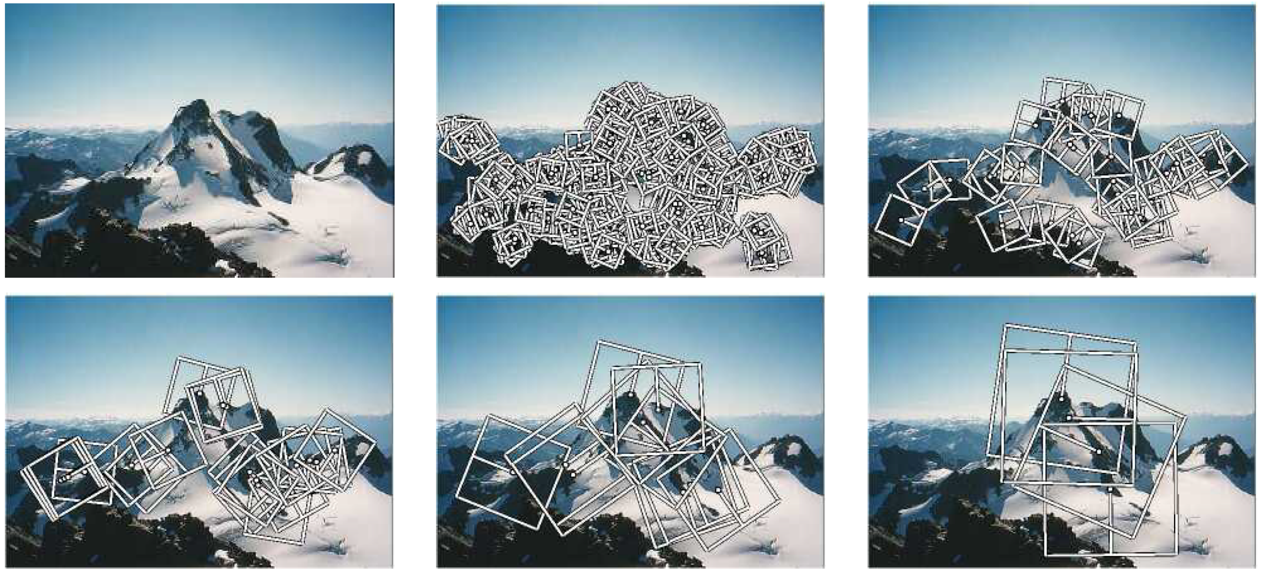


Image pyramid



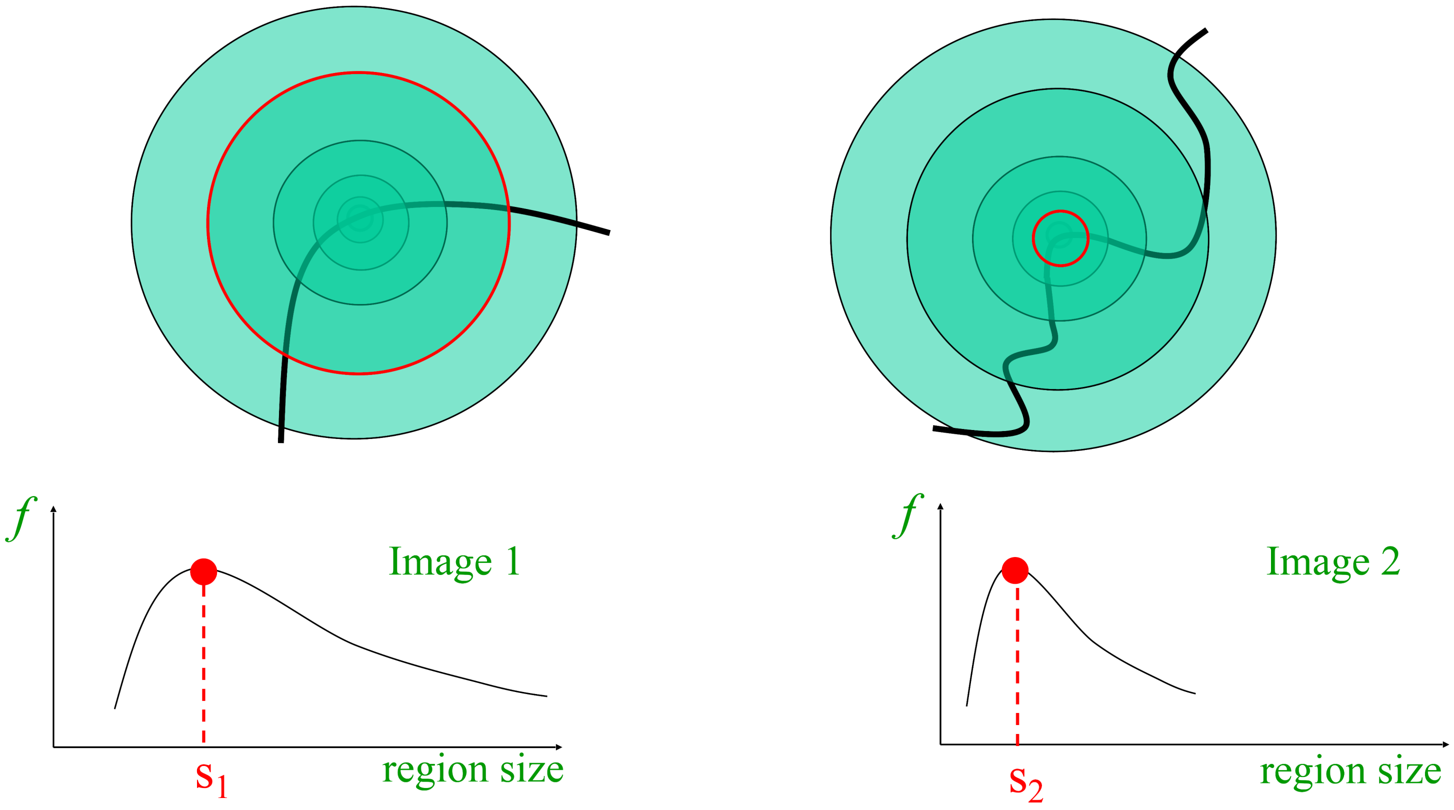
Multi-scale oriented patches (MOPS) extracted at five pyramid levels (Brown, Szeliski, and Winder 2005)

Scale Invariance

Solution 2: detect features that are stable in both location and scale

Intuition: Find local maxima in both position and scale

What filter can we use for scale selection?



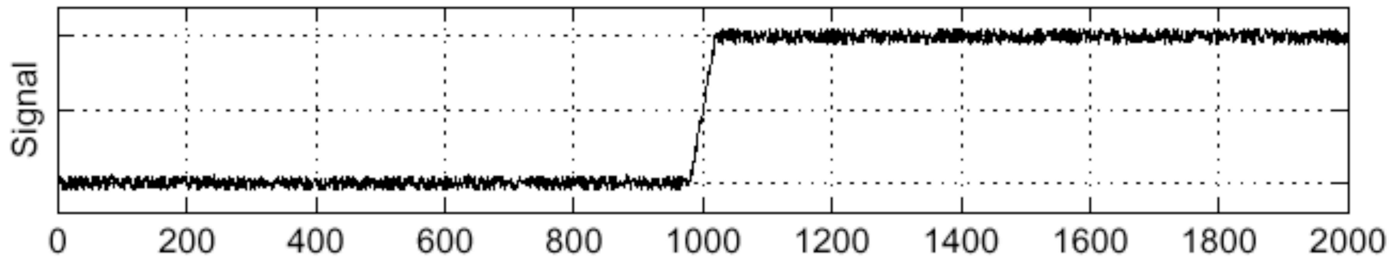
Recall Derivative Filter

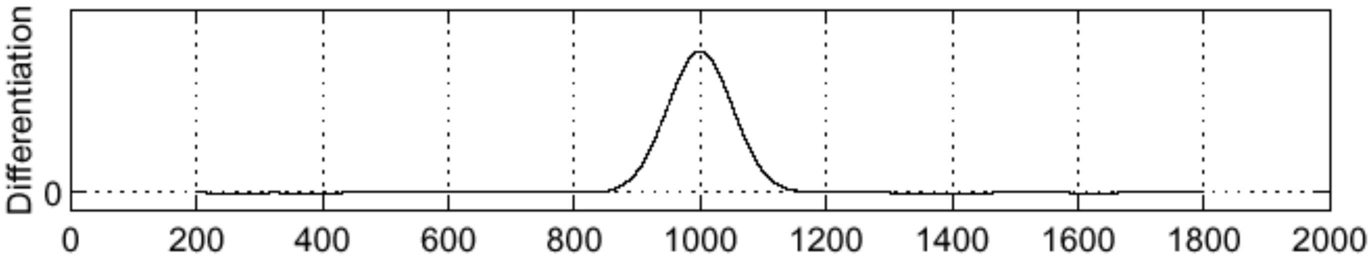
Central difference

$$f'(x) = \lim_{h \to 0} \frac{f(x + 0.5h) - f(x - 0.5h)}{h}$$

-1	0	1

X derivative

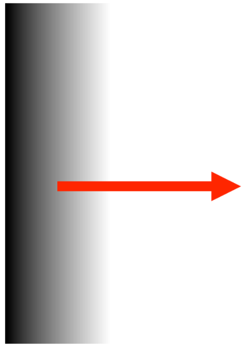




Find edge

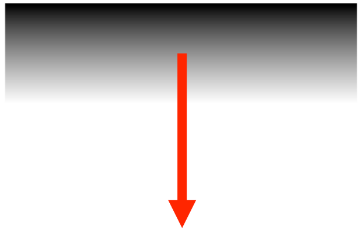
Image Gradient

Gradient in x only



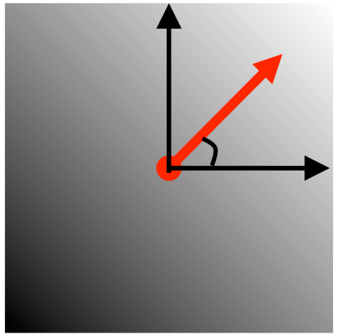
$$\nabla f = \left[\frac{\partial f}{\partial x}, 0\right]$$

Gradient in y only



$$\nabla f = \left[0, \frac{\partial f}{\partial y}\right]$$

Gradient in both x and y



$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

Gradient direction

$$\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} \Big/ \frac{\partial f}{\partial x}\right)$$

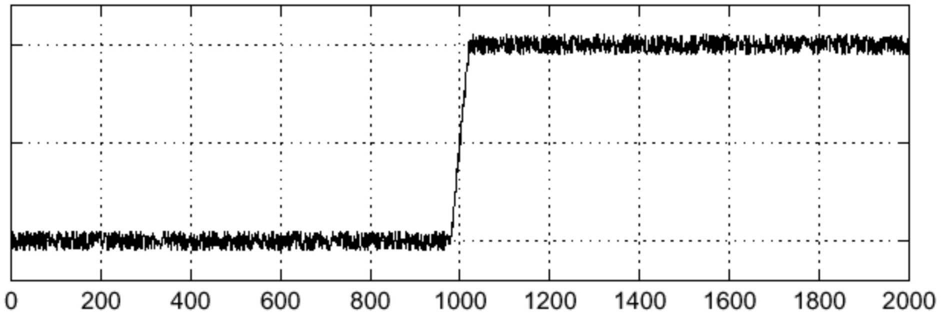
Gradient magnitude

$$||\nabla f|| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Signal Noises

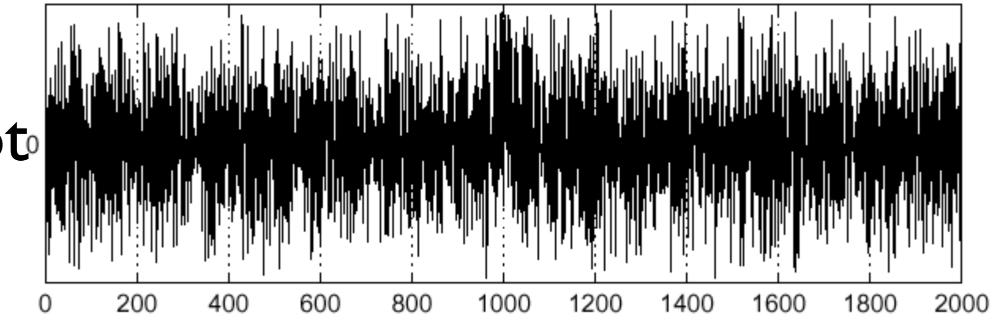
Derivative filters are sensitive to noises

Intensity plot



How to deal with noises?

Derivative plot



Gaussian Filter

Smoothing

1D $$g(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

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

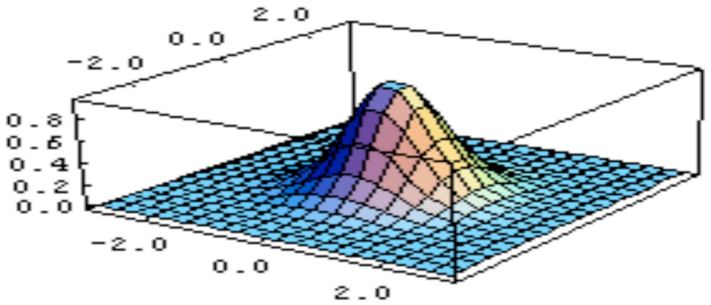
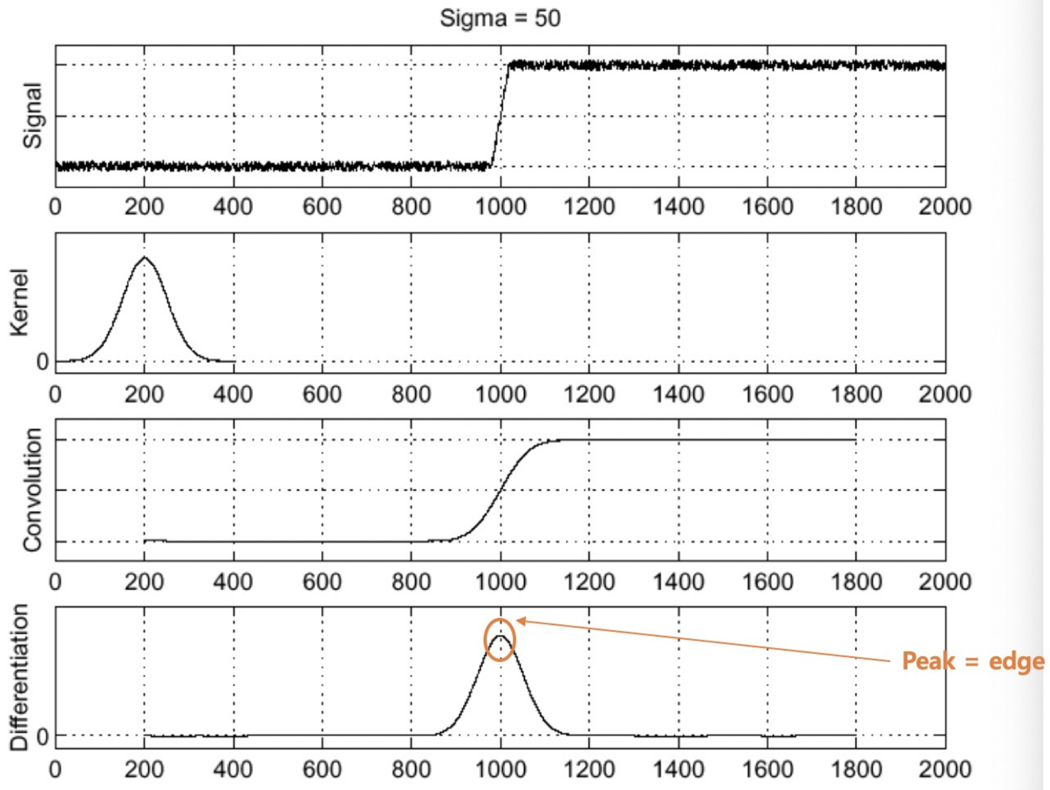


Image f

Gaussian Filter h

Convolution $h \star f$

Derivative $\frac{\partial}{\partial x}(h \star f)$



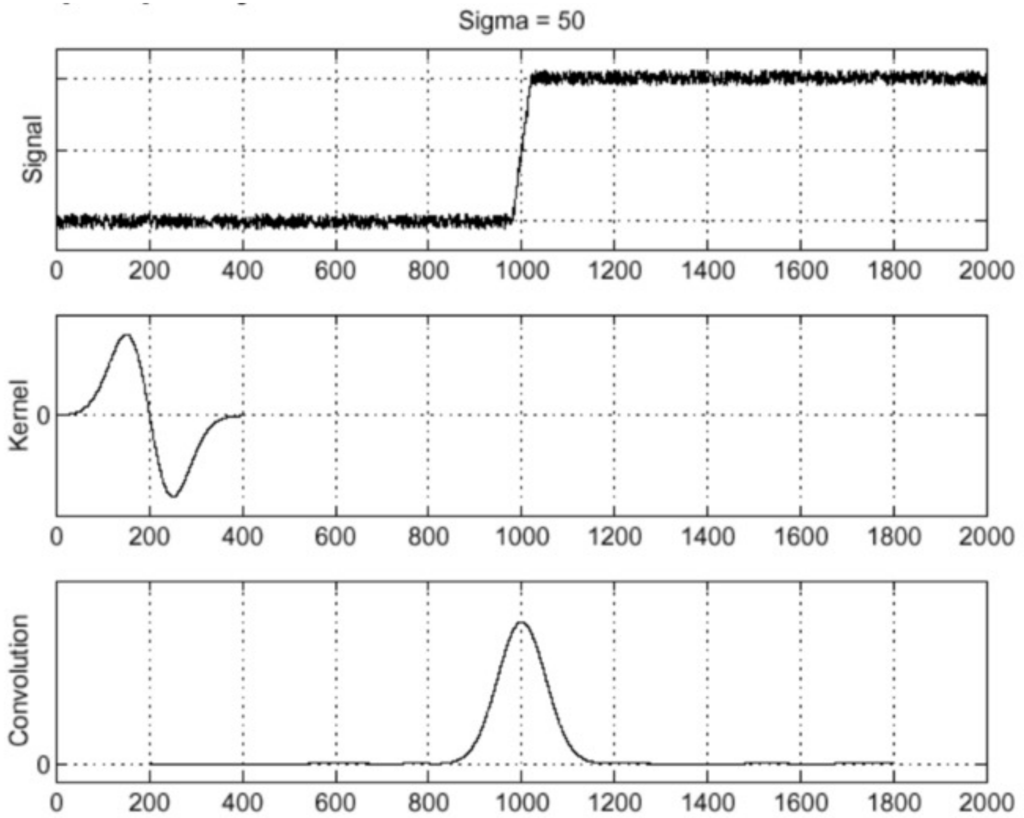
Derivative of Gaussian Filter

- Convolution is associative $\frac{\partial}{\partial x}(h \star f) = (\frac{\partial}{\partial x}h) \star f$

f

Smoothing and derivative $\frac{\partial}{\partial x}h$

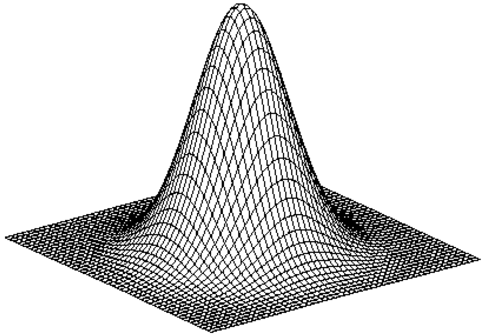
$(\frac{\partial}{\partial x}h) \star f$



Derivative of Gaussian Filter

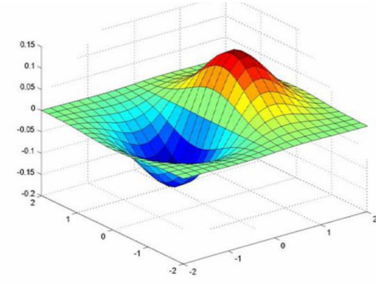
Convolution is associative $$\frac{\partial}{\partial x}(h \star f) = (\frac{\partial}{\partial x}h) \star f$$

$$g_x(x,y) = \frac{\partial g(x,y)}{\partial x} = \frac{-x}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad g_y(x,y) = \frac{\partial g(x,y)}{\partial y} = \frac{-y}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

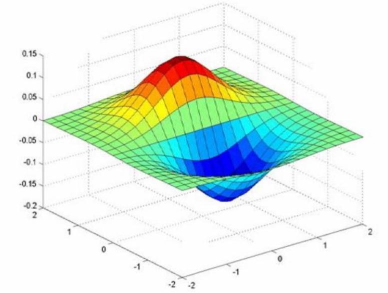


Gaussian

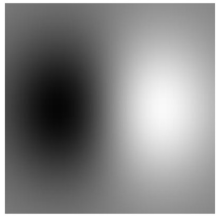
$$g(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

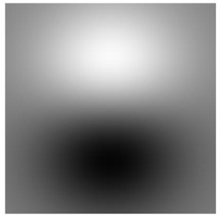


x-direction



y-direction





Laplace Filter

first-order
finite difference

$$f'(x) = \lim_{h\to 0} \frac{f(x+0.5h) - f(x-0.5h)}{h}$$

Derivative filter

-1	0	1

second-order
finite difference

$$f''(x) \approx \frac{\delta_h^2[f](x)}{h^2} = \frac{\frac{f(x+h)-f(x)}{h} - \frac{f(x)-f(x-h)}{h}}{h} = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

Laplace filter

1	-2	1

Laplace Filter

2D $$\nabla^2 \mathbf{I} = \frac{\partial^2 \mathbf{I}}{\partial x^2} + \frac{\partial^2 \mathbf{I}}{\partial y^2}$$

1	-2	1

1D Laplace filter

0	1	0
1	-4	1
0	1	0

2D Laplace filter

Laplacian of Gaussian Filter

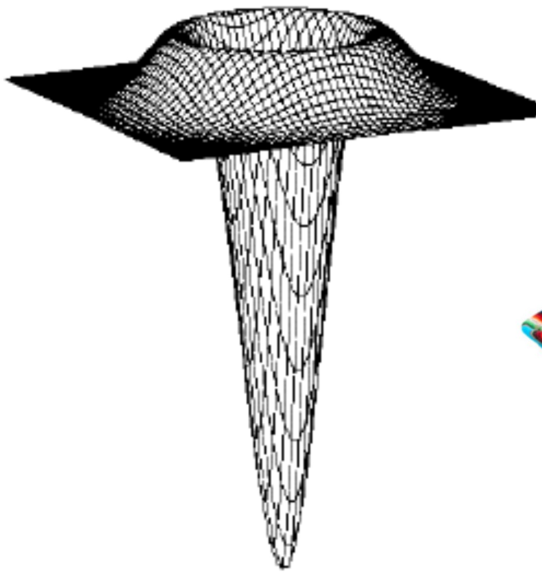
$$\nabla^2 \mathbf{I} = \frac{\partial^2 \mathbf{I}}{\partial x^2} + \frac{\partial^2 \mathbf{I}}{\partial y^2}$$

$$\nabla^2 \mathbf{I} \circ g = \nabla^2 g \circ \mathbf{I}$$

$$\nabla^2 g = \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} g(x, y)$$

Smoothing and second derivative

$\nabla^2 h_\sigma(u, v)$



Laplacian of Gaussian

Mexican Hat Function

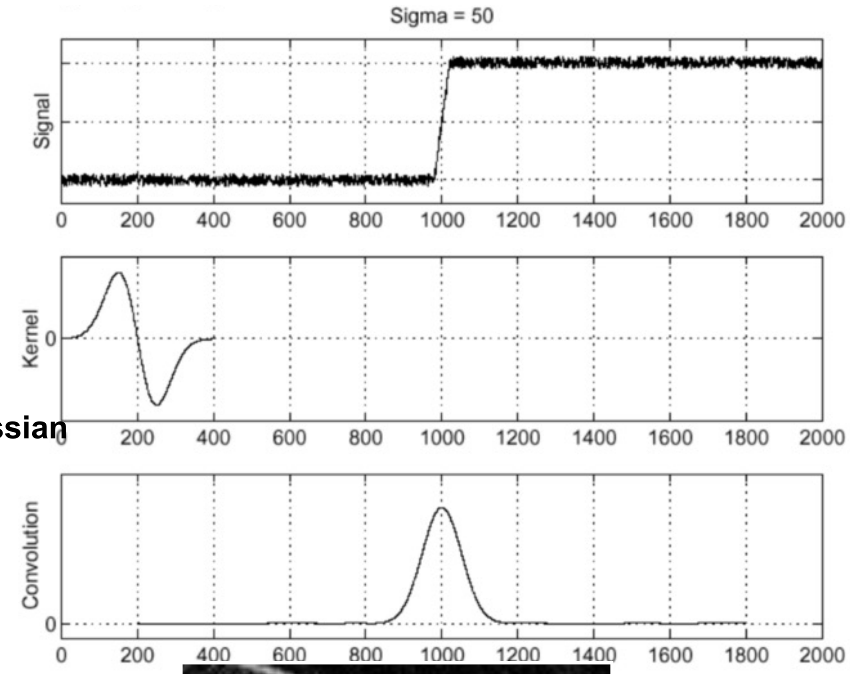
Laplacian of Gaussian Filter

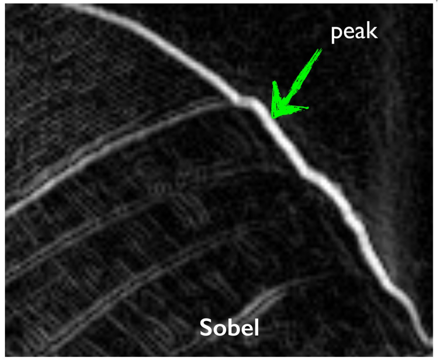
f

$\frac{\partial}{\partial x} h$

Derivative of Gaussian

$(\frac{\partial}{\partial x} h) \star f$



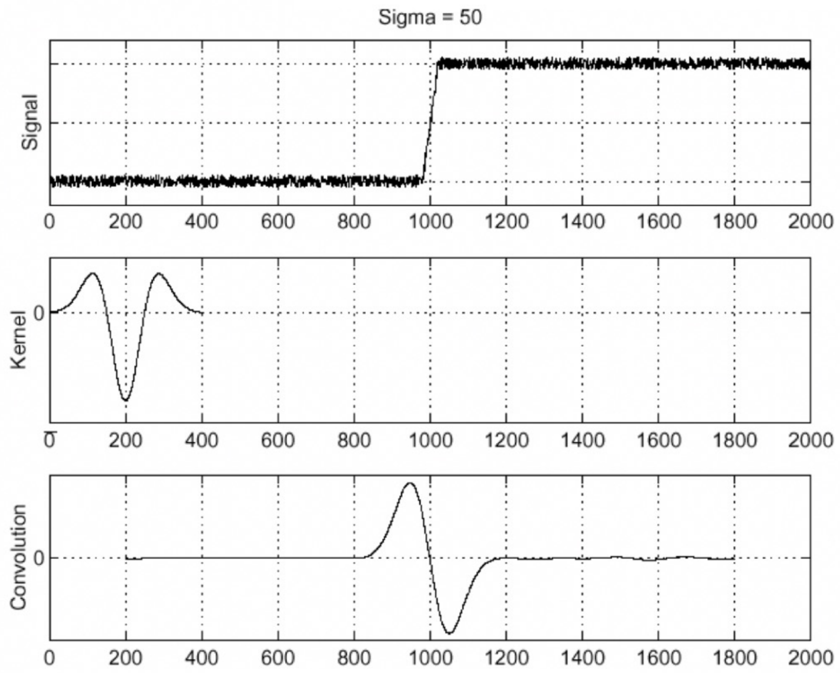


f

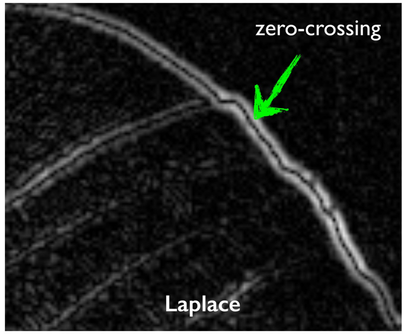
$\frac{\partial^2}{\partial x^2} h$

Laplacian of Gaussian

$(\frac{\partial^2}{\partial x^2} h) \star f$

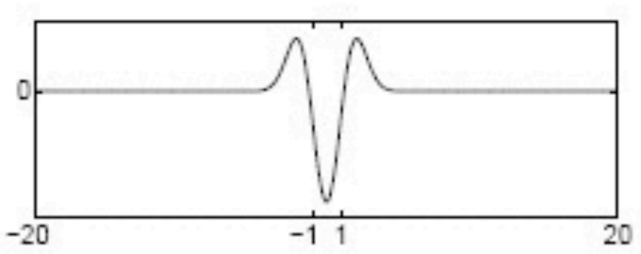


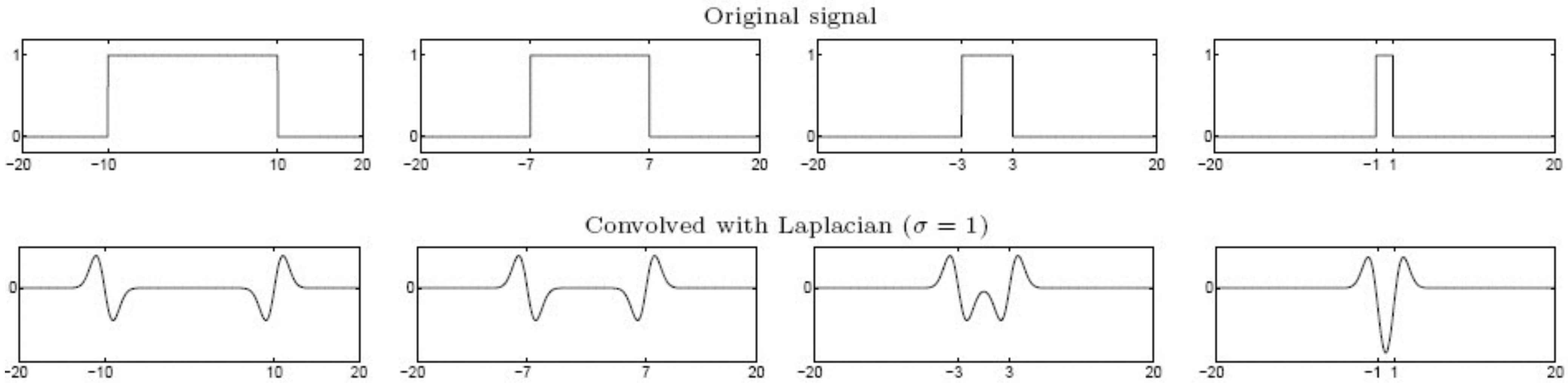
Zero crossings



Laplacian of Gaussian for Scale Selection

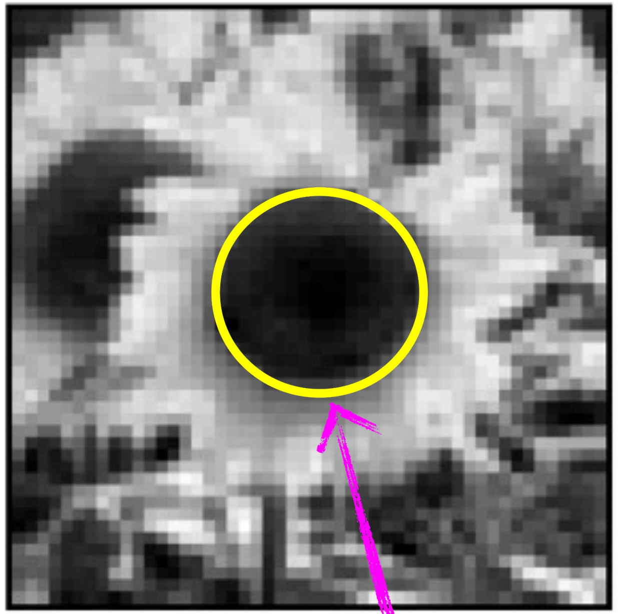
Laplacian filter

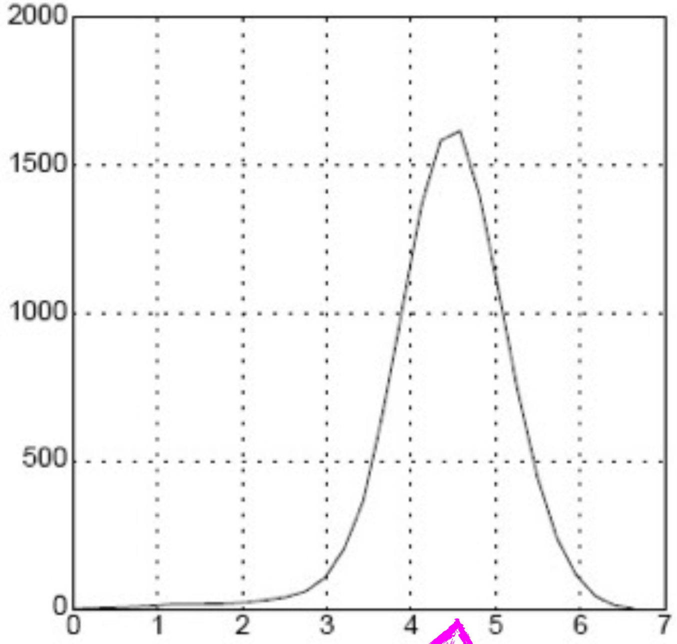




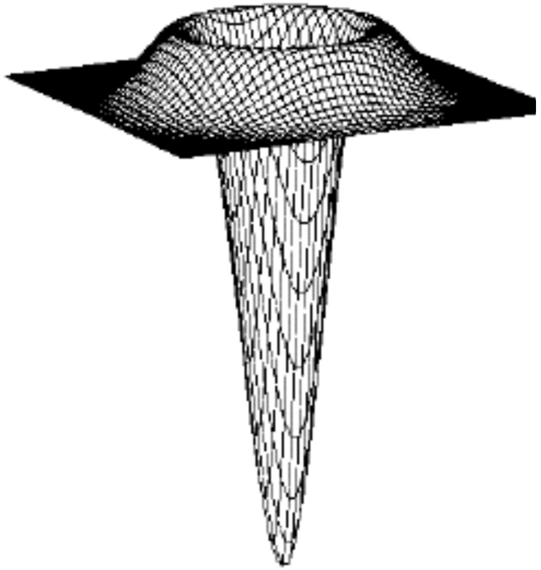
Highest response when the signal has the same **characteristic scale** as the filter

Laplacian of Gaussian for Scale Selection





$\nabla^2 h_\sigma(u, v)$

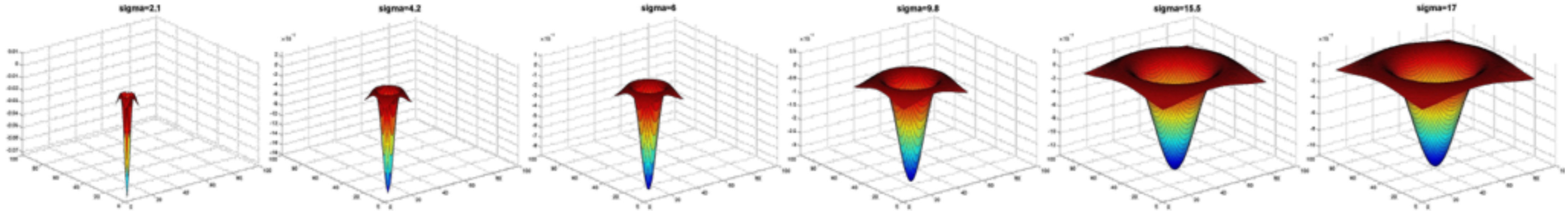


Laplacian of Gaussian

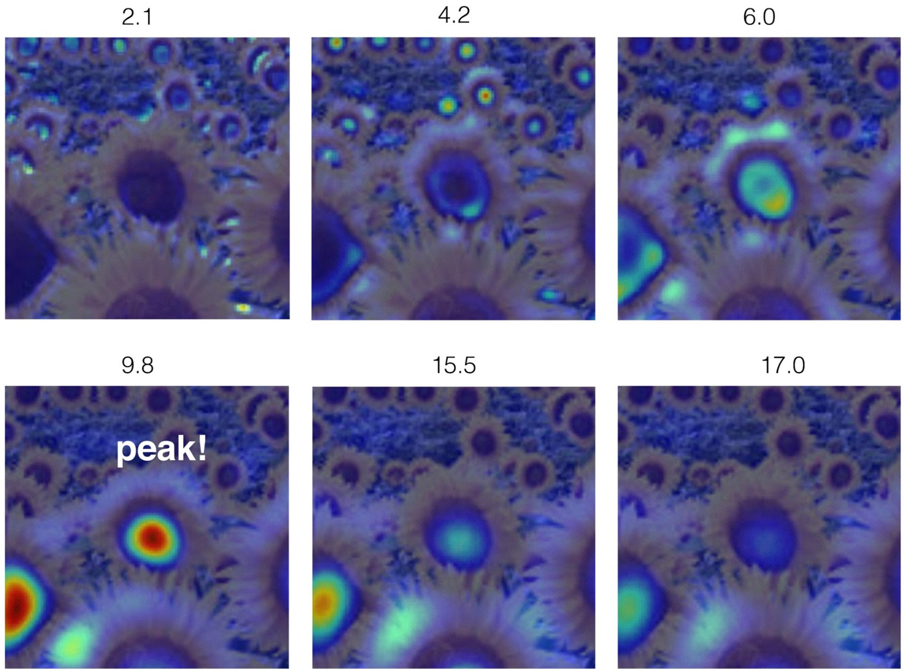
characteristic scale

Search over different scales σ

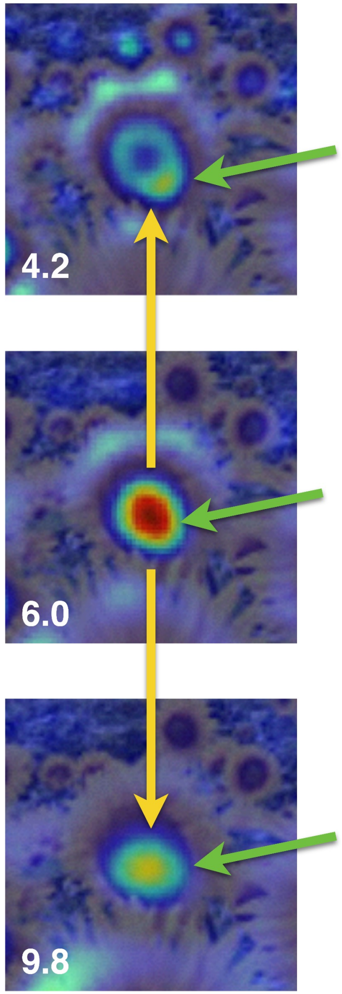
Laplacian of Gaussian for Scale Selection



Multi-scale
2D Blob
detection



Laplacian of Gaussian for Scale Selection



local maximum

cross-scale maximum

local maximum

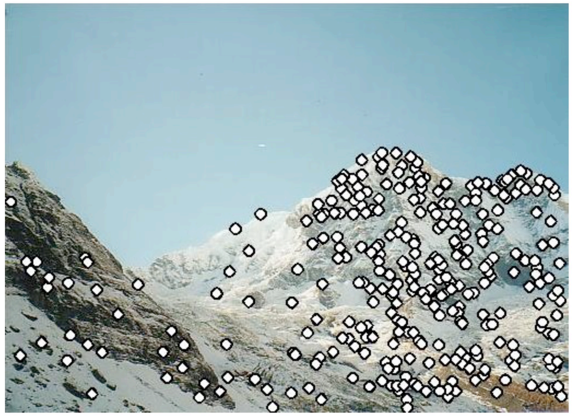
local maximum

Scale Invariance Feature Transform (SIFT)

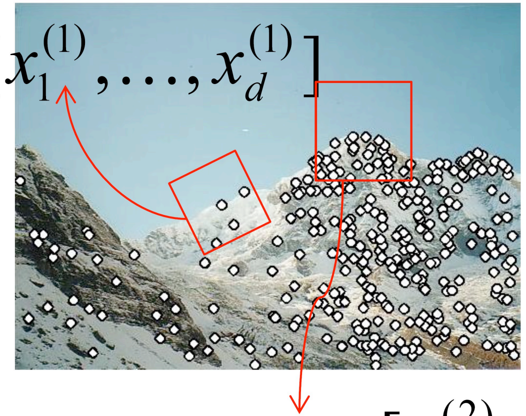
Keypoint detection

Compute descriptors

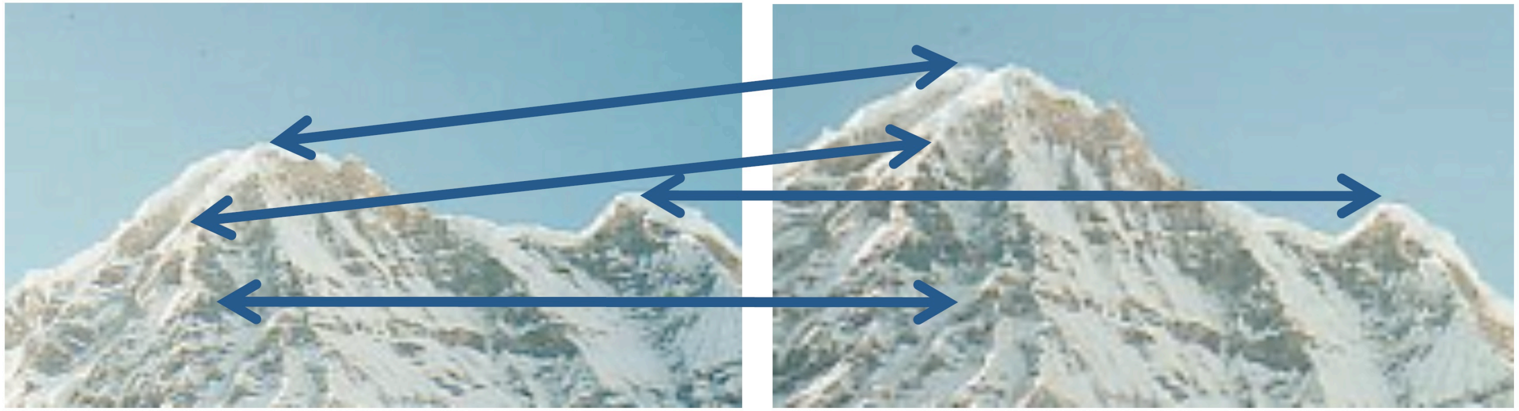
Matching descriptors



$$\mathbf{x}_1 = [x_1^{(1)}, \ldots, x_d^{(1)}]$$



$$\mathbf{x}_2 = [x_1^{(2)}, \ldots, x_d^{(2)}]$$



David Lowe, Distinctive Image Features from Scale-Invariant Keypoints. IJCV, 2004

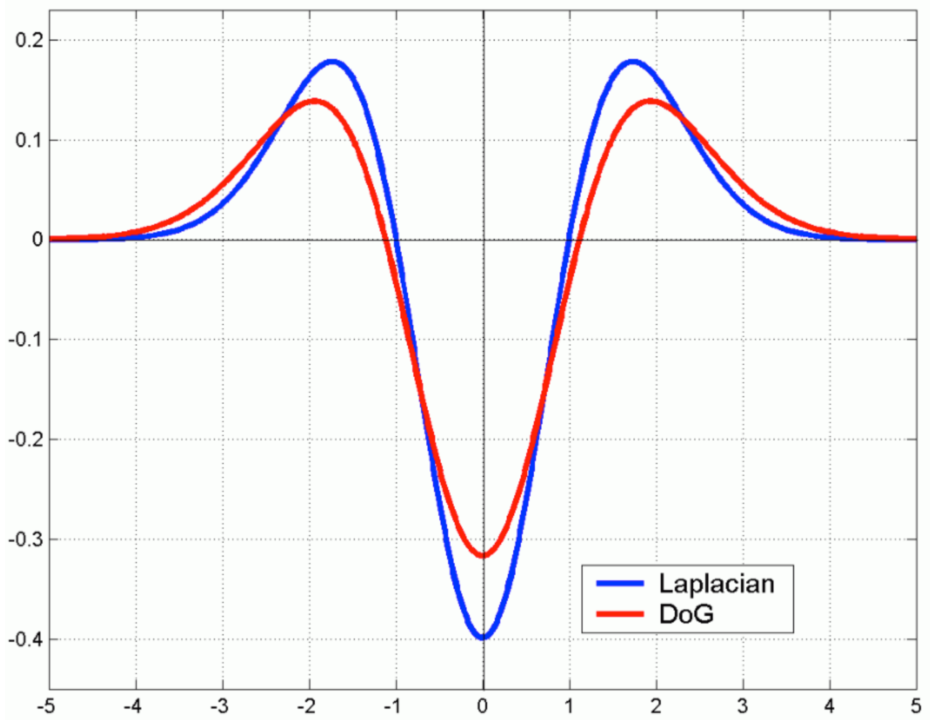
SIFT: Scale-space Extrema Detection

Difference of Gaussian (DoG)

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

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

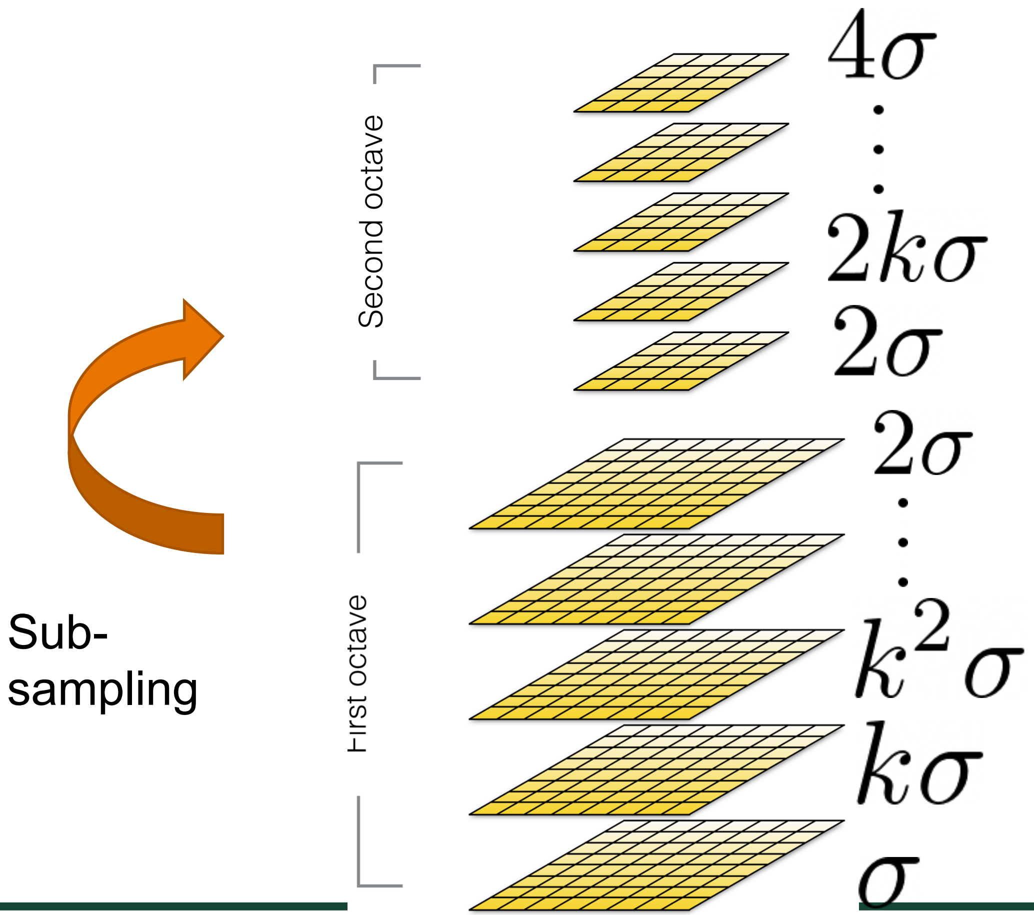
$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma). \end{aligned}$$



Approximate of Laplacian of Gaussian
(efficient to compute)

SIFT: Scale-space Extrema Detection

Gaussian pyramid



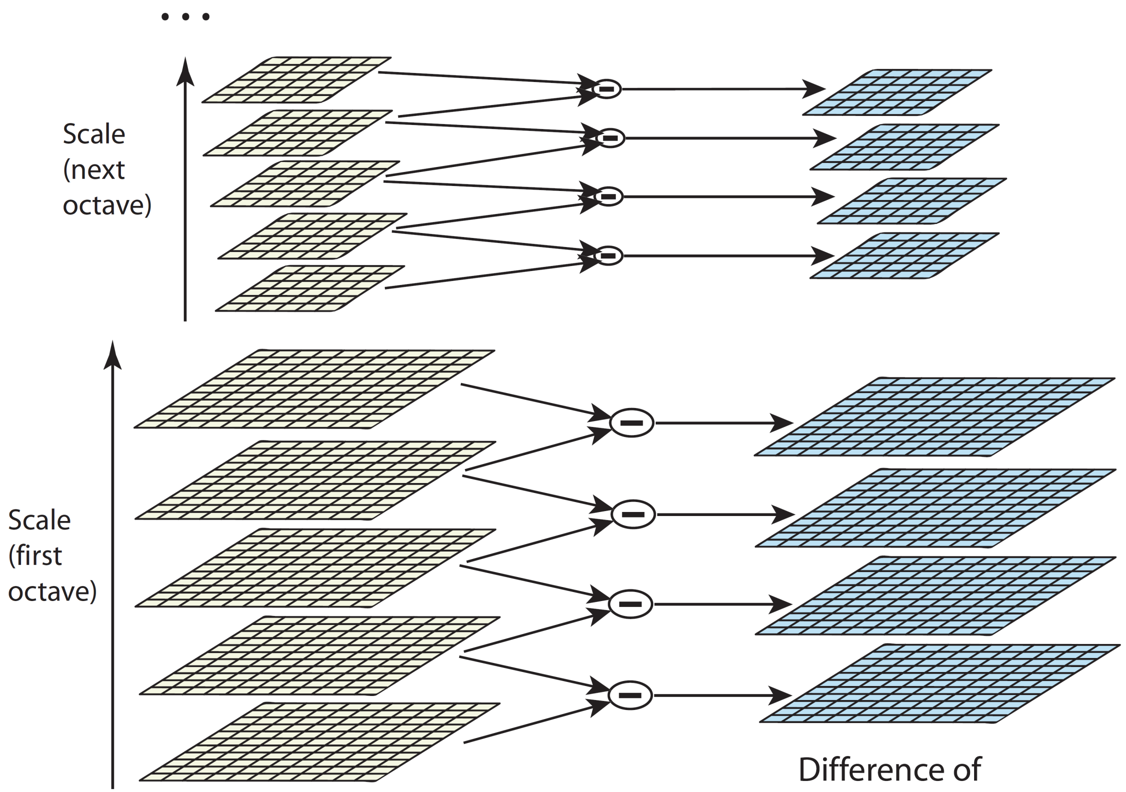
- Gaussian filters

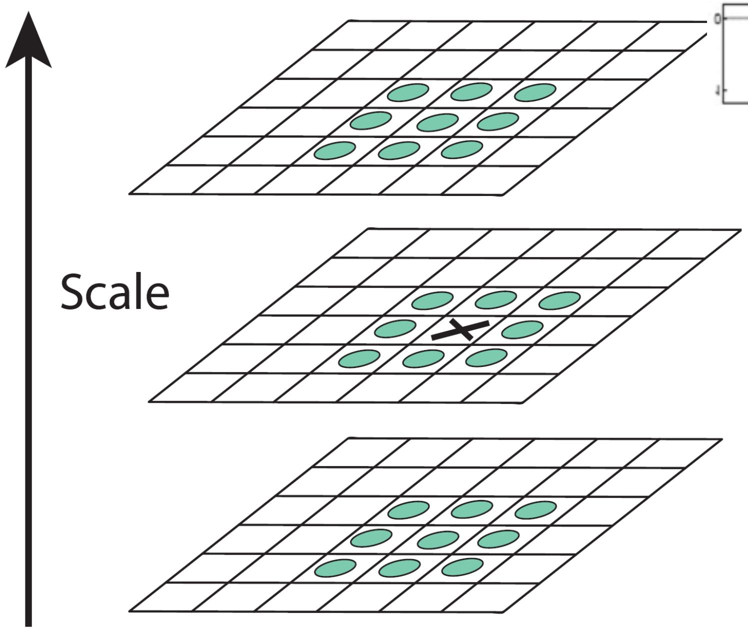
$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$$

$$G_{\sigma_1} * G_{\sigma_2} = G_{\sigma} \quad \sigma^2 = \sigma_1^2 + \sigma_2^2$$

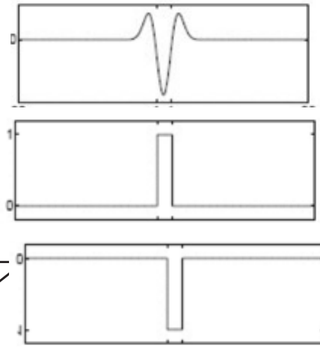
- Sub-sampling by a factor of 2
  - Multiple the Gaussian kernel deviation by

SIFT: Scale-space Extrema Detection





Maxima and minima of DoG images



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

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

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma). \end{aligned}$$

SIFT Descriptor

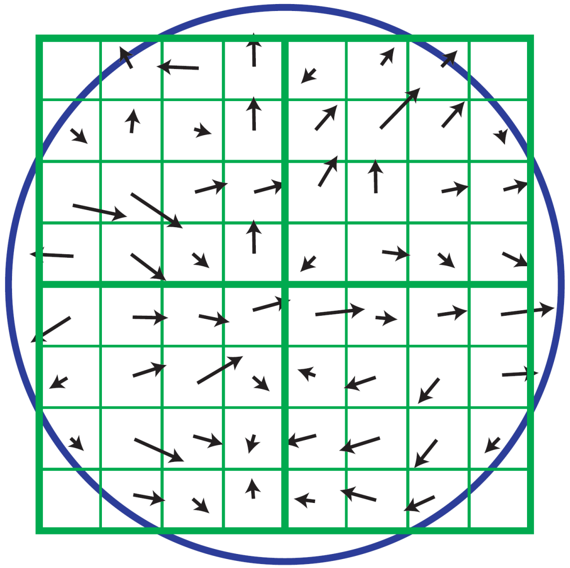


Image gradients

Image gradient magnitude and orientation

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

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

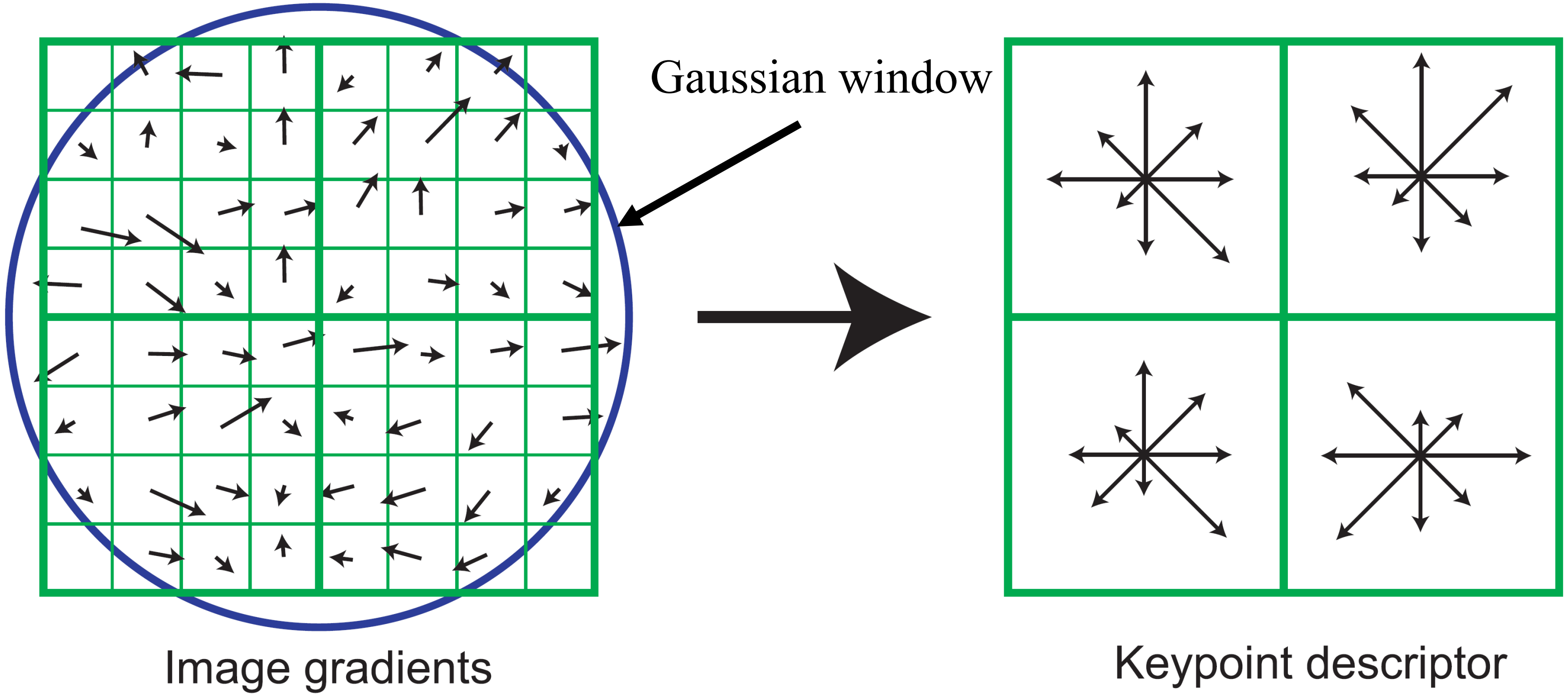
X-derivative Y-derivative

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

SIFT Descriptor

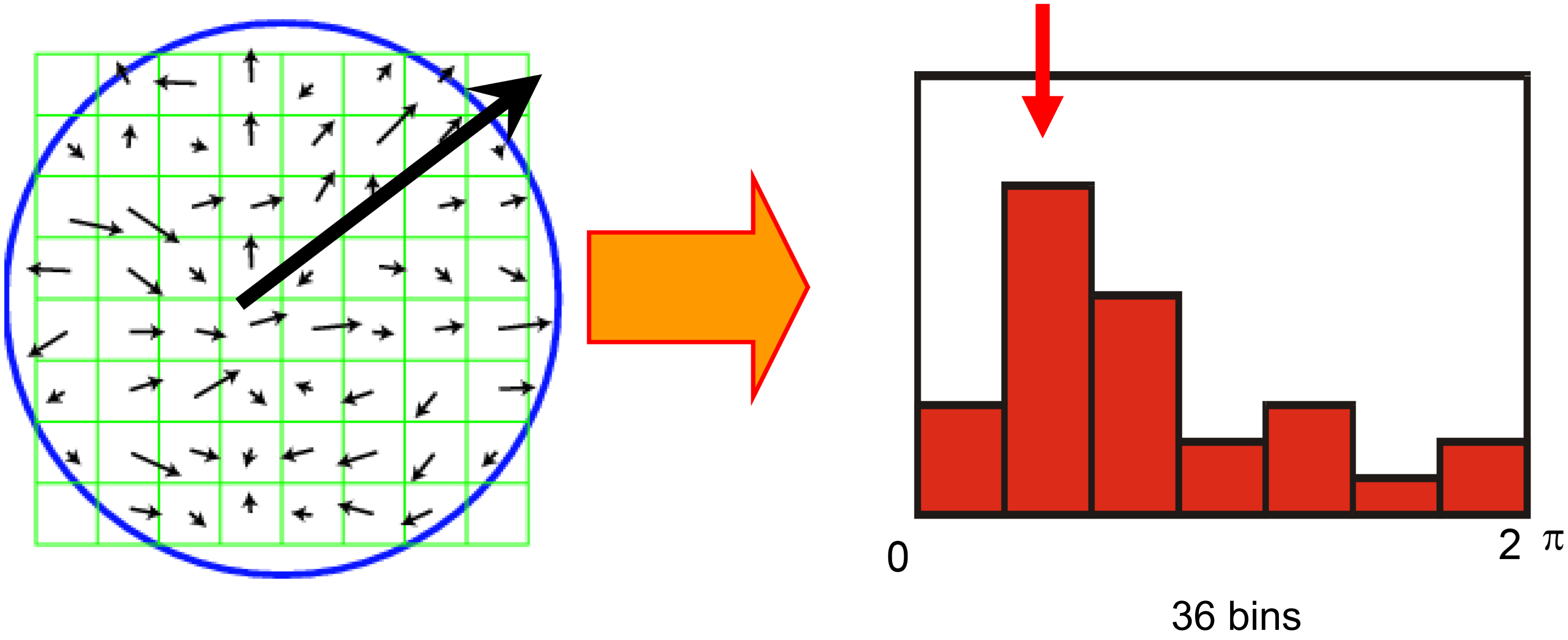
- Divide the 16x16 window into a 4x4 grid of cells (2x2 case shown below)
- Compute an orientation histogram for each cell
- 16 cells * 8 orientations = 128 dimensional descriptor

Using the scale of the keypoint to select the level of Gaussian blur for the image



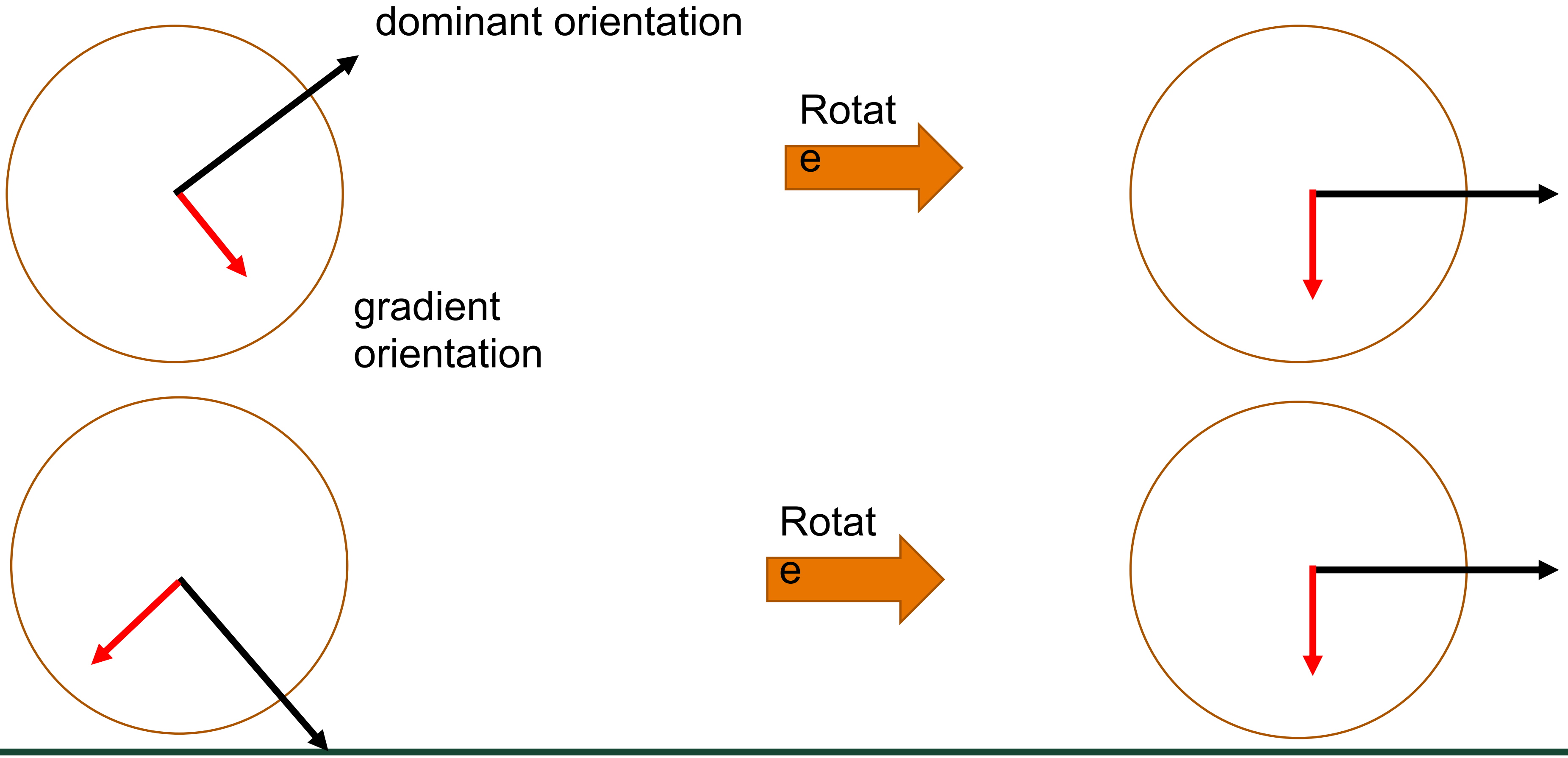
SIFT: Rotation Invariance

Rotate all orientations by the dominant orientation



SIFT: Rotation Invariance

Rotate all orientations by the dominant orientation



SIFT Properties

Can handle change in viewpoint (up to about 60 degree out of plane rotation)

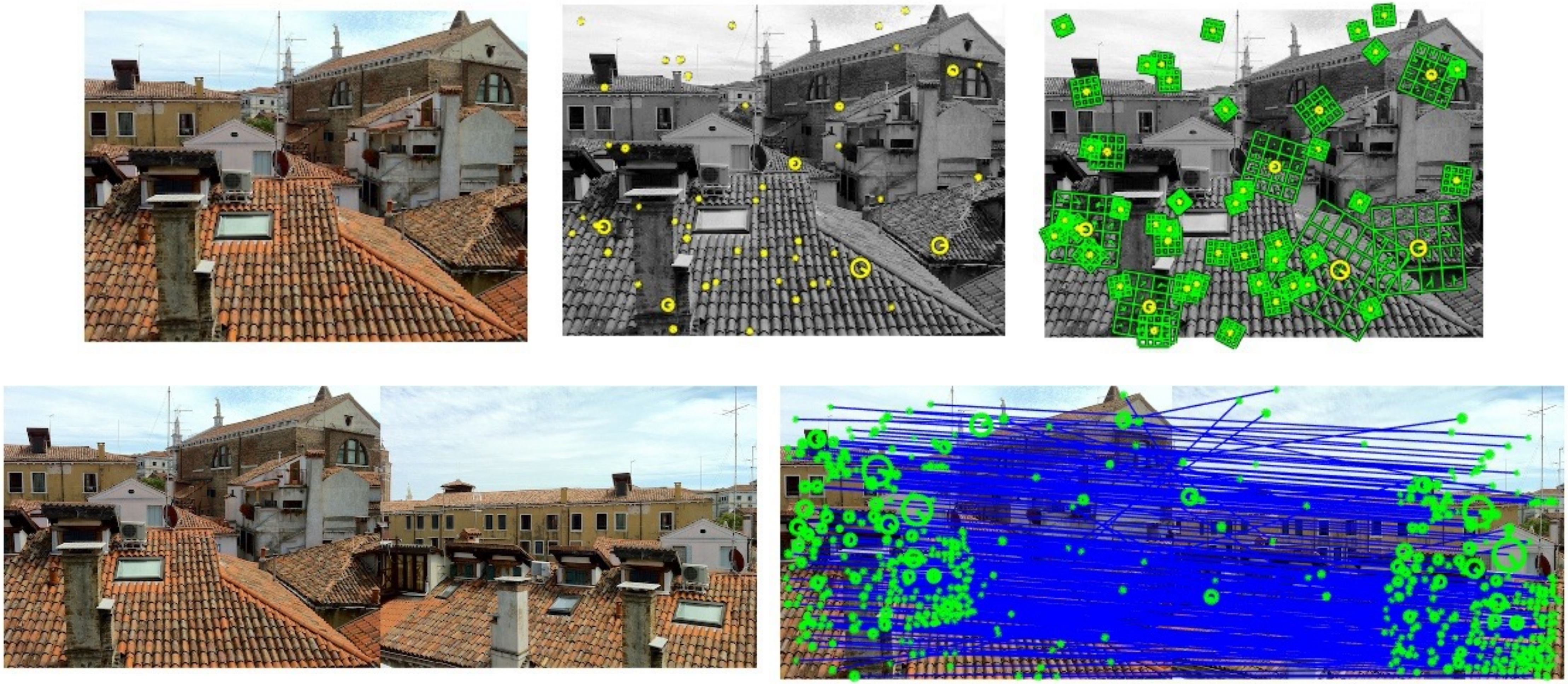
Can handle significant change in illumination

Relatively fast < 1s for moderate image sizes

Lots of code available

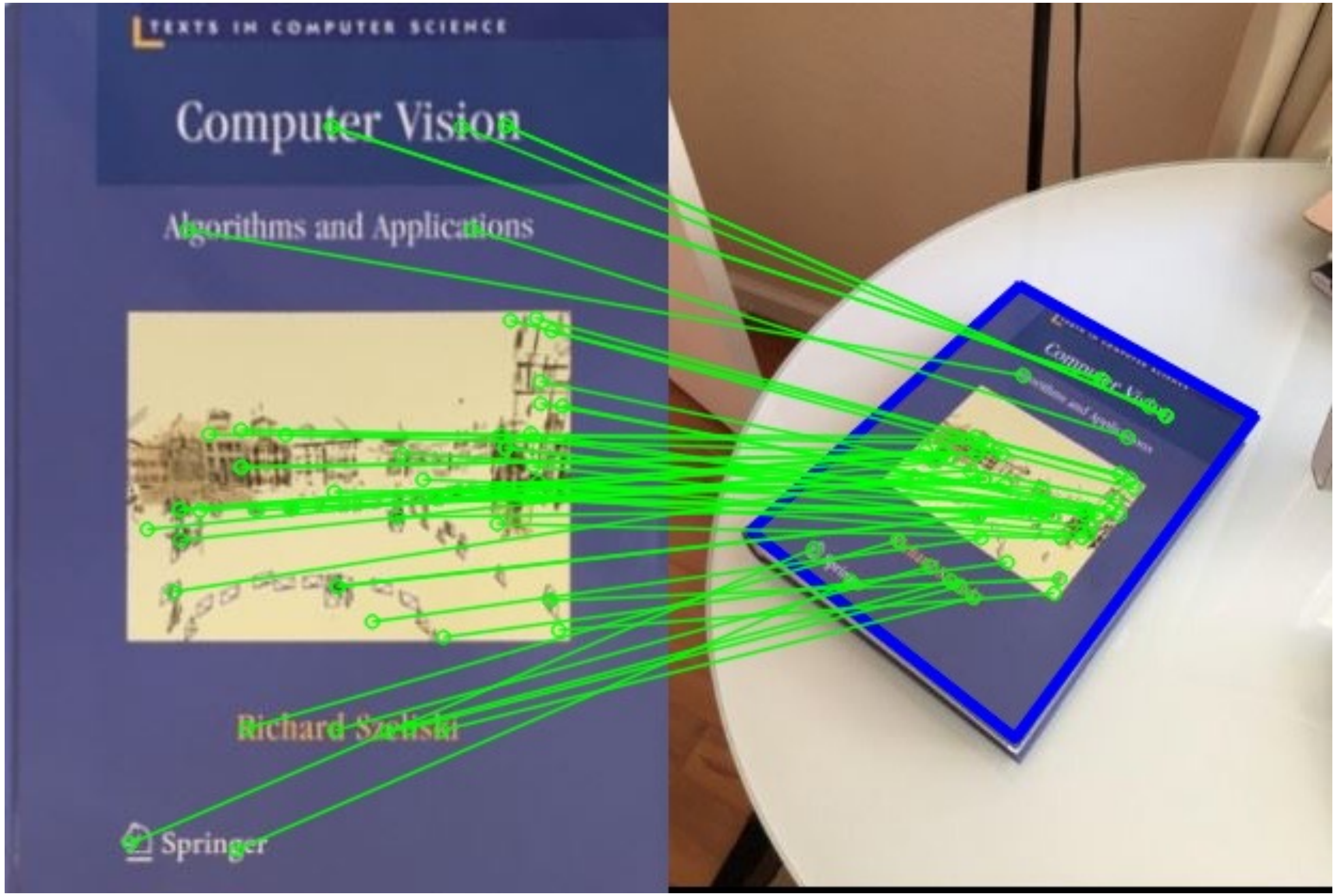
- E.g., https://www.vlfeat.org/overview/sift.html

SIFT Matching Example



https://www.vlfeat.org/overview/sift.html

SIFT Matching Example



Further Reading

Section 7.1, Computer Vision, Richard Szeliski

David Lowe, Distinctive Image Features from Scale-Invariant Keypoints. IJCV, 2004

ORB: An efficient alternative to SIFT or SURF. Rublee et al., ICCV, 2011