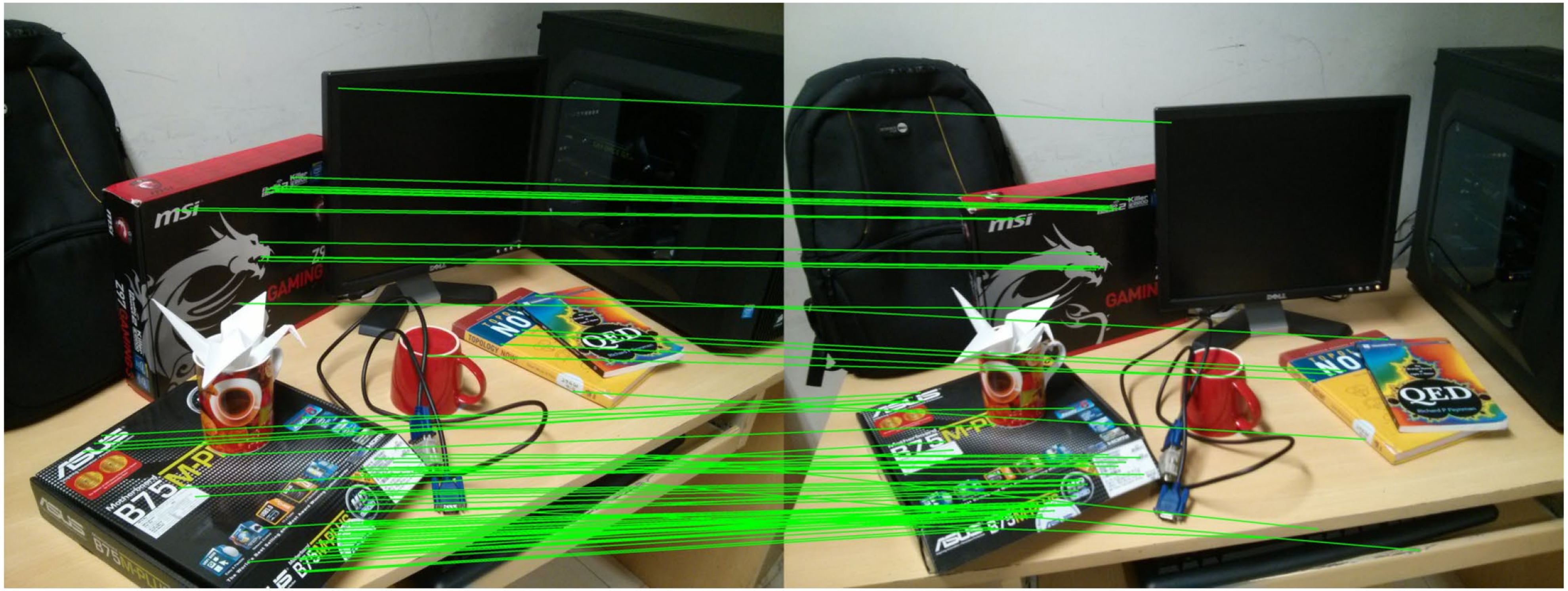
Scale Invariance Feature Transform

CS 4391 Introduction Computer Vision

Professor Yu Xiang

The University of Texas at Dallas

Feature Detection and Matching

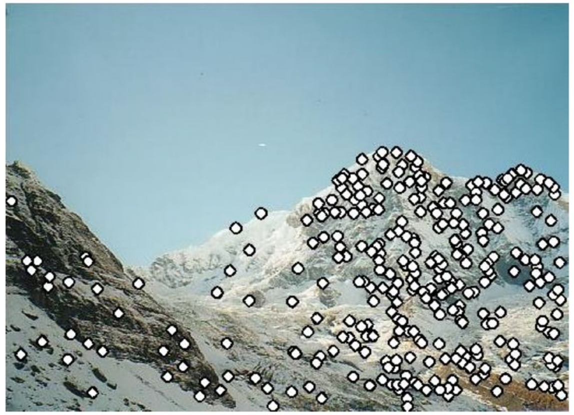


Geometry-aware Feature Matching for Structure from Motion Applications. Shah et al, WACV'15

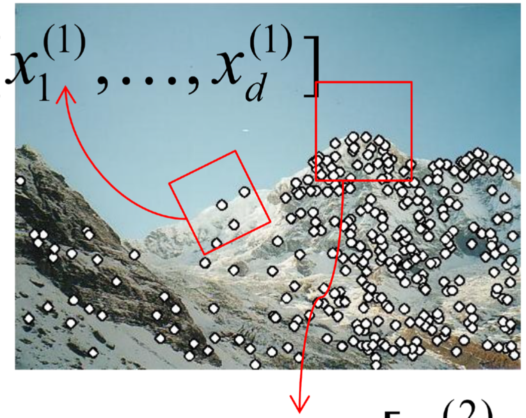
Applications: stereo matching, image stitching, 3D reconstruction, camera pose estimation, object recognition

Scale Invariance Feature Transform (SIFT)

- Keypoint detection



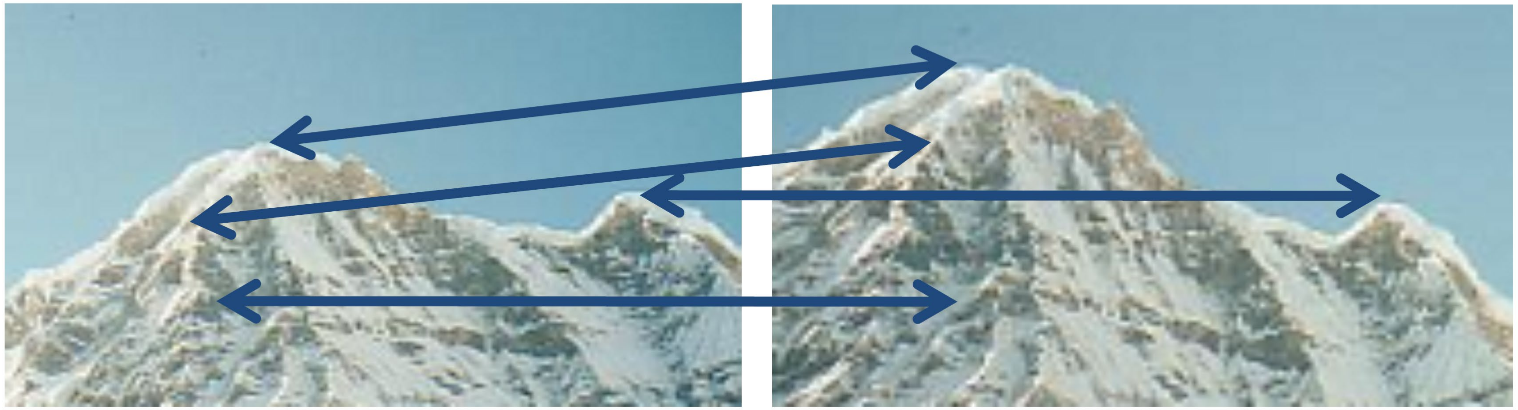
- Compute descriptors



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

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

- Matching descriptors



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

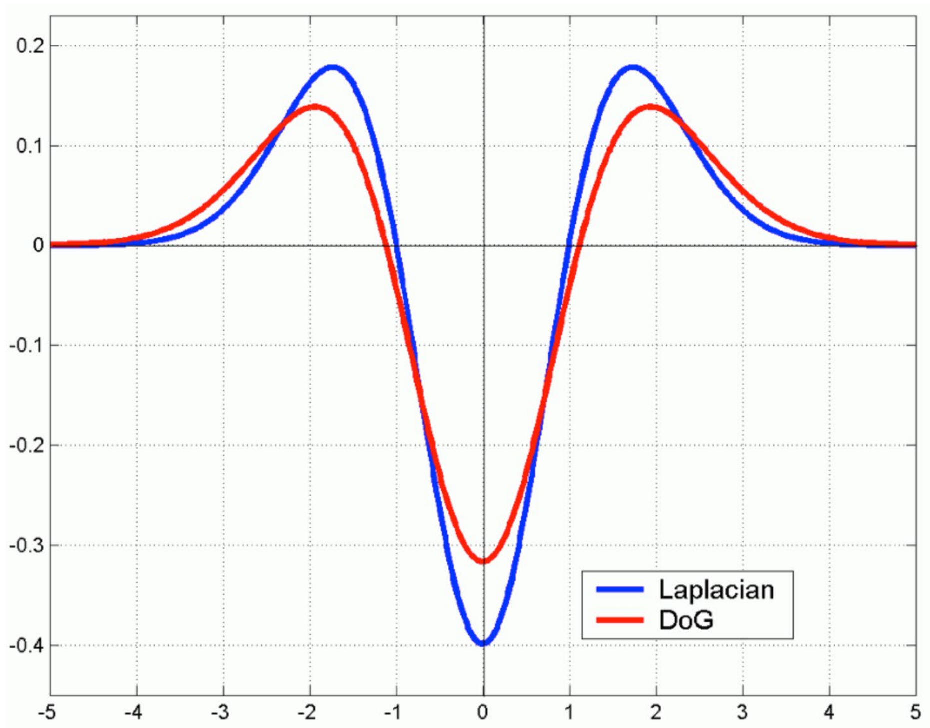
SIFT: Scale-space Extrema Detection

- Difference of Gaussian (DoG)

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

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

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

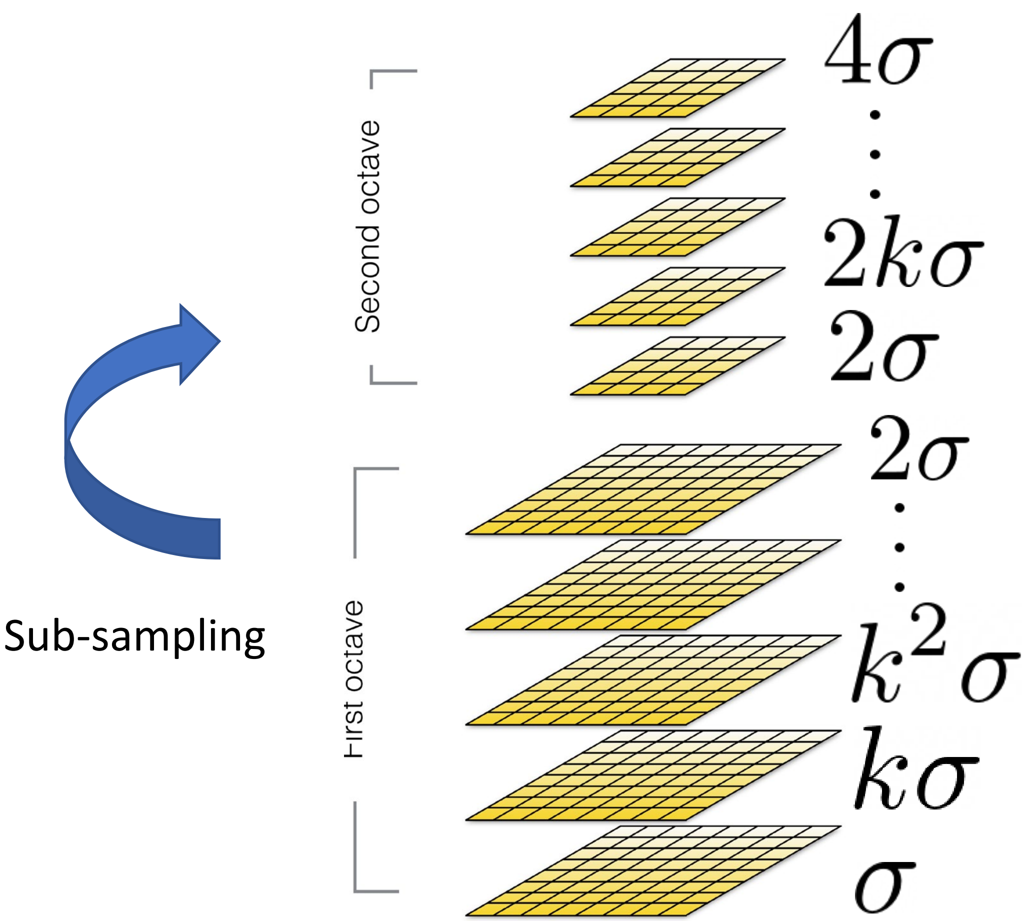


Approximate of Laplacian of Gaussian
(efficient to compute)

k is a constant

SIFT: Scale-space Extrema Detection

- Gaussian pyramid



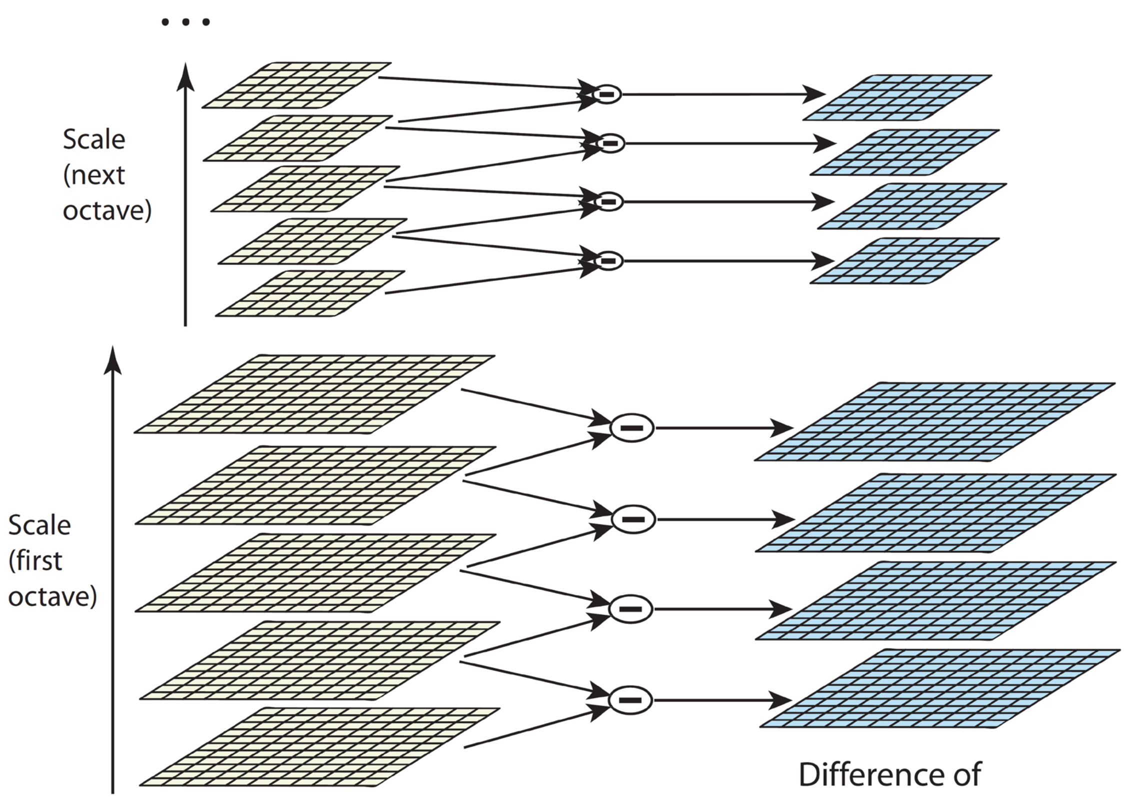
- Gaussian filters

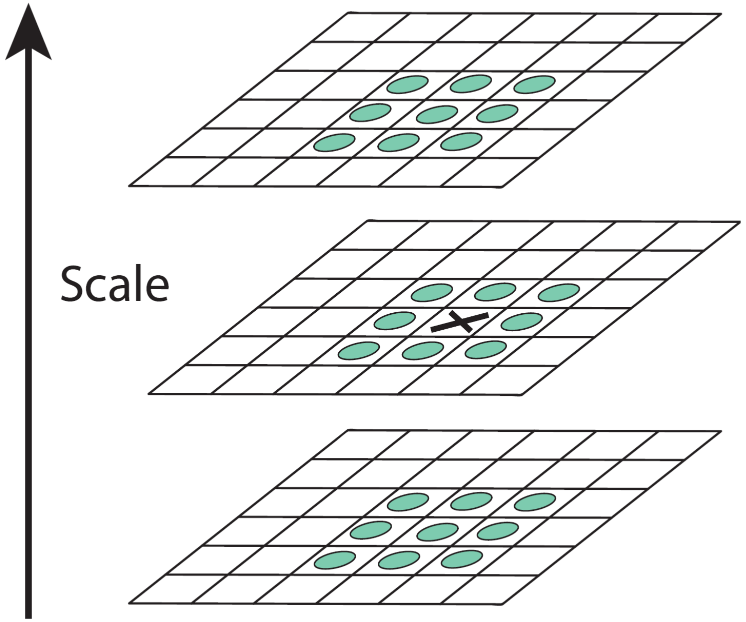
$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$$

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

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

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 gradient magnitude and orientation

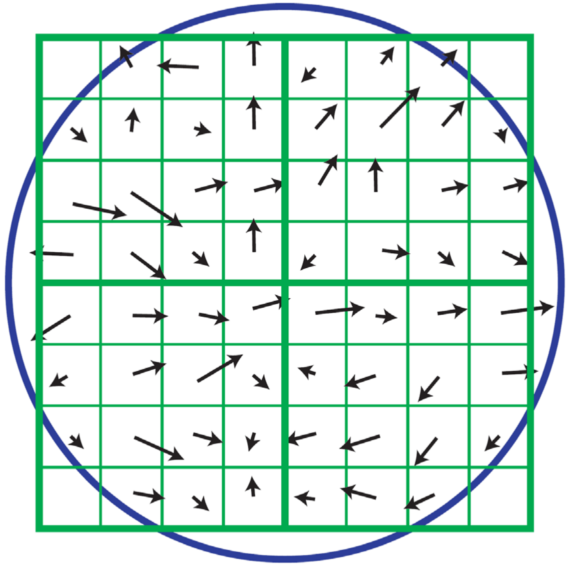


Image gradients

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

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

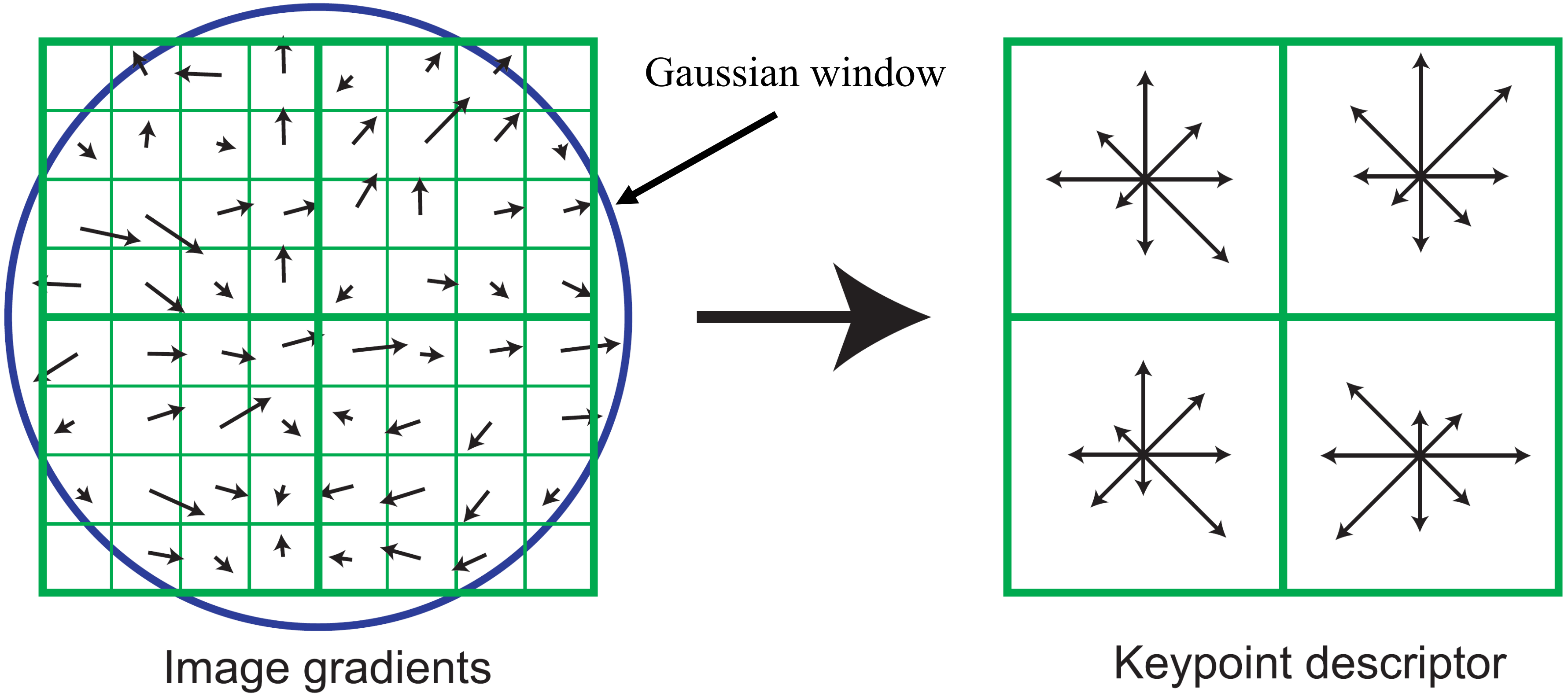
X-derivative Y-derivative

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

SIFT Descriptor

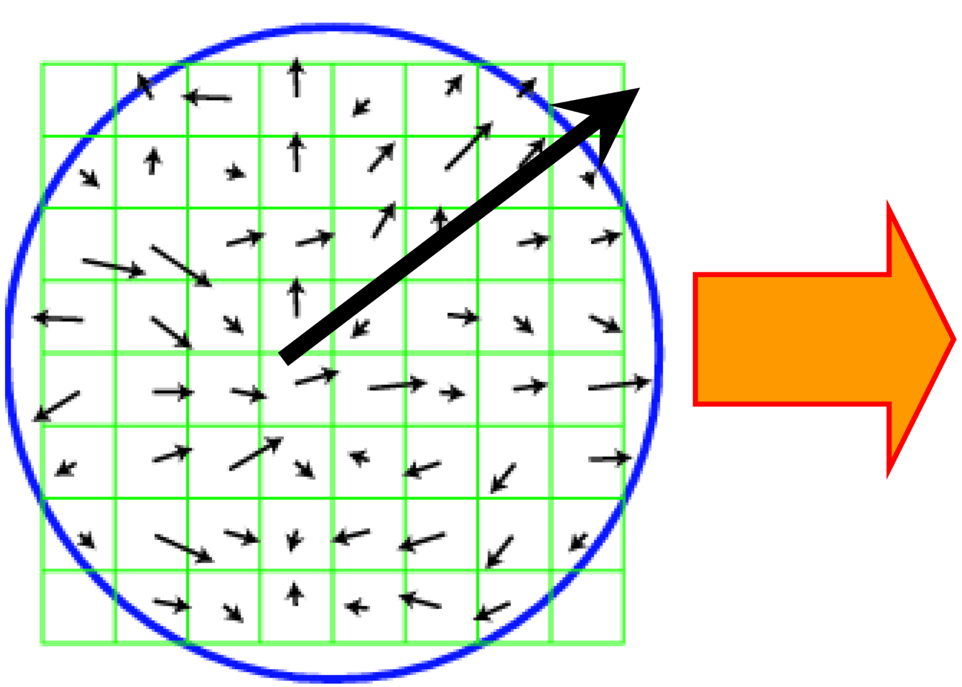
- Divide the 16x16 window into a 4x4 grid of cells (2x2 case shown below)
- Compute an orientation histogram for each cell
- 16 cells * 8 orientations = 128 dimensional descriptor

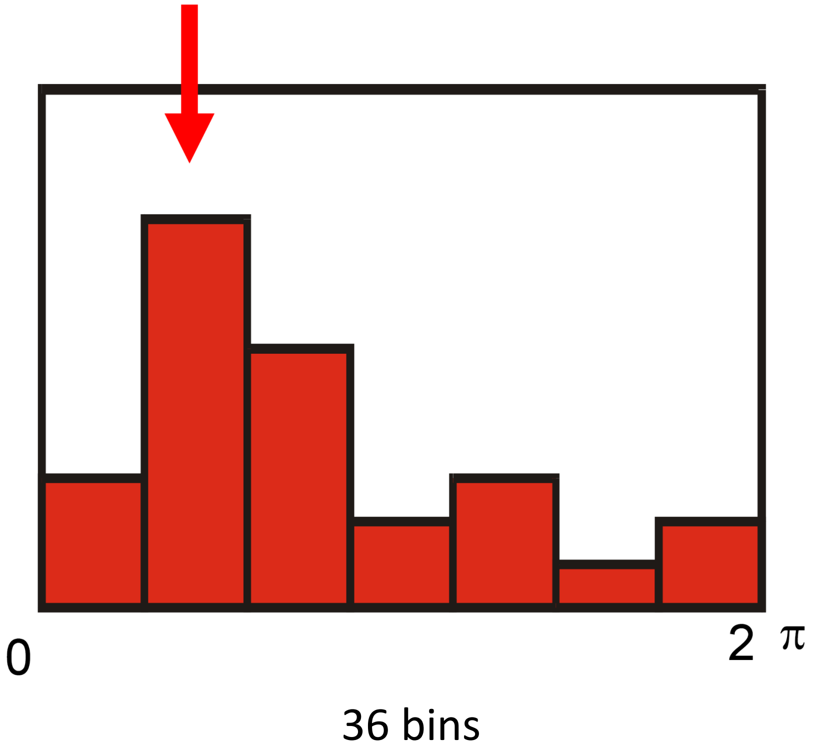
Using the scale of the keypoint to select the level of Gaussian blur for the image



SIFT: Rotation Invariance

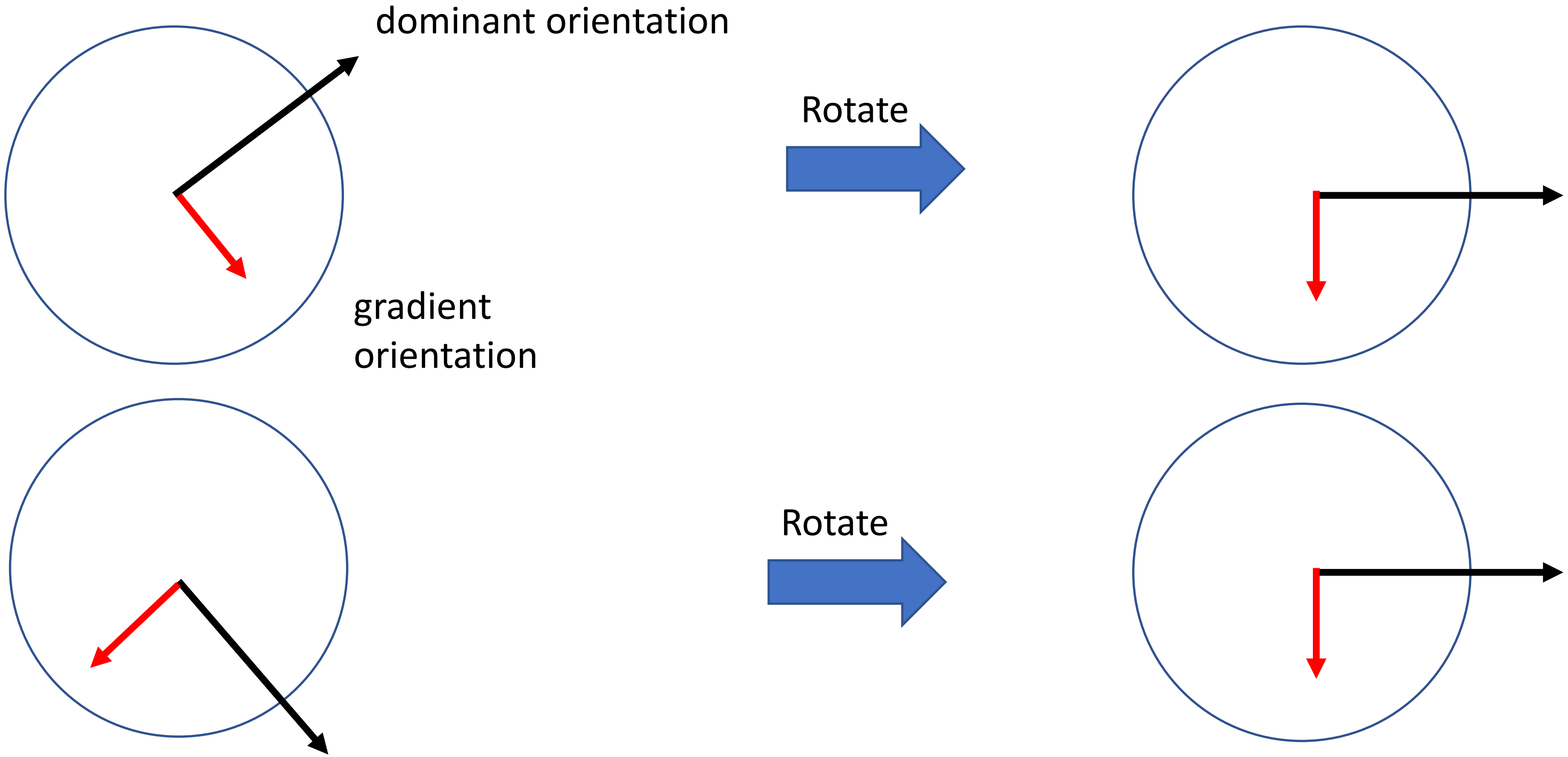
- Rotate all orientations by the dominant orientation





SIFT: Rotation Invariance

- Rotate all orientations by the dominant orientation



SIFT Properties

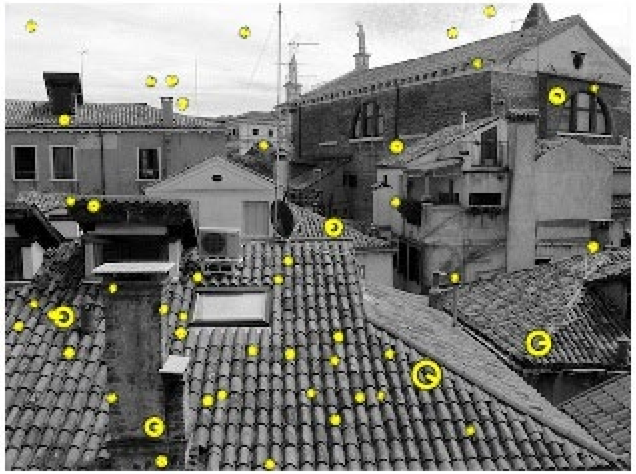
- Can handle change in viewpoint (up to about 60 degree out of plane rotation)

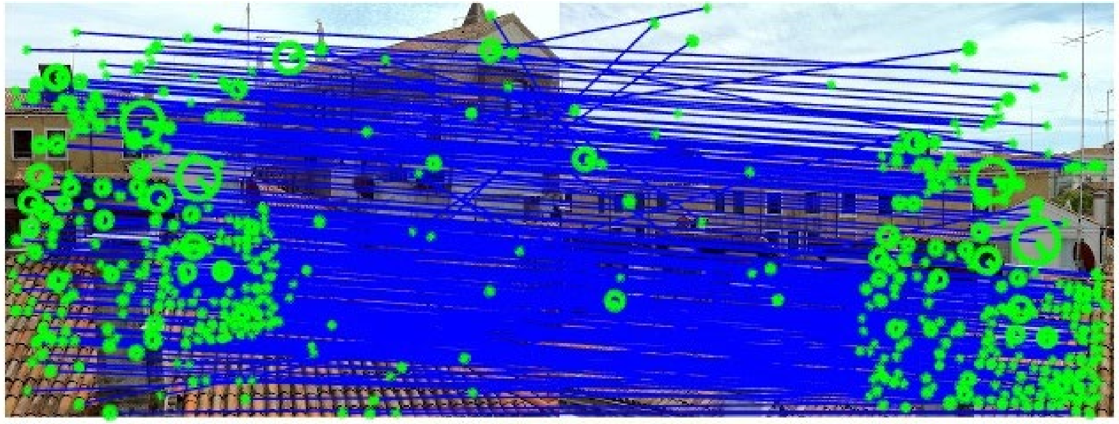
- Can handle significant change in illumination

- Relatively fast < 1s for moderate image sizes

- Lots of code available
  - E.g., https://www.vlfeat.org/overview/sift.html

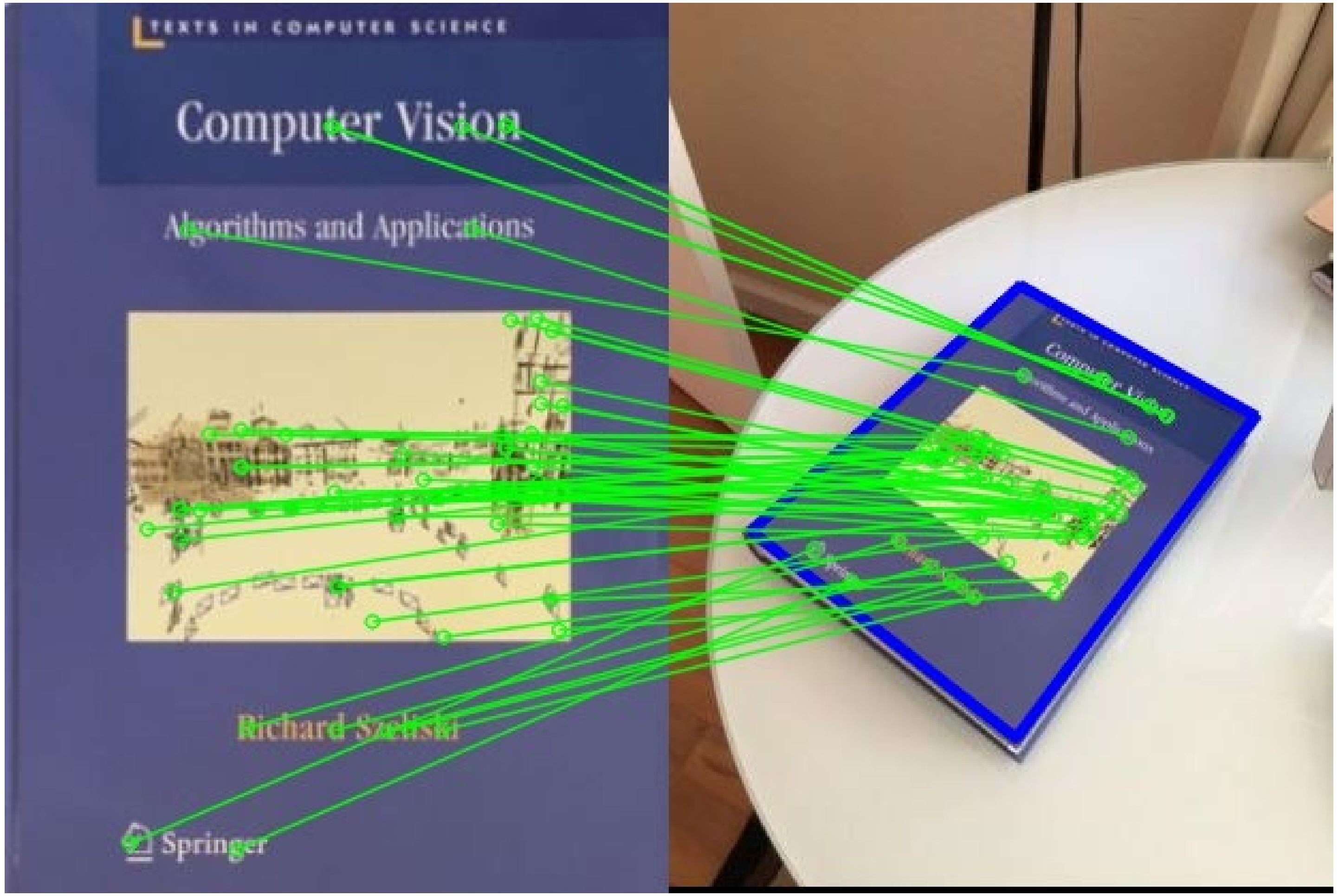
SIFT Matching Example

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 https://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf
- ORB: An efficient alternative to SIFT or SURF. Rublee et al., ICCV, 2011