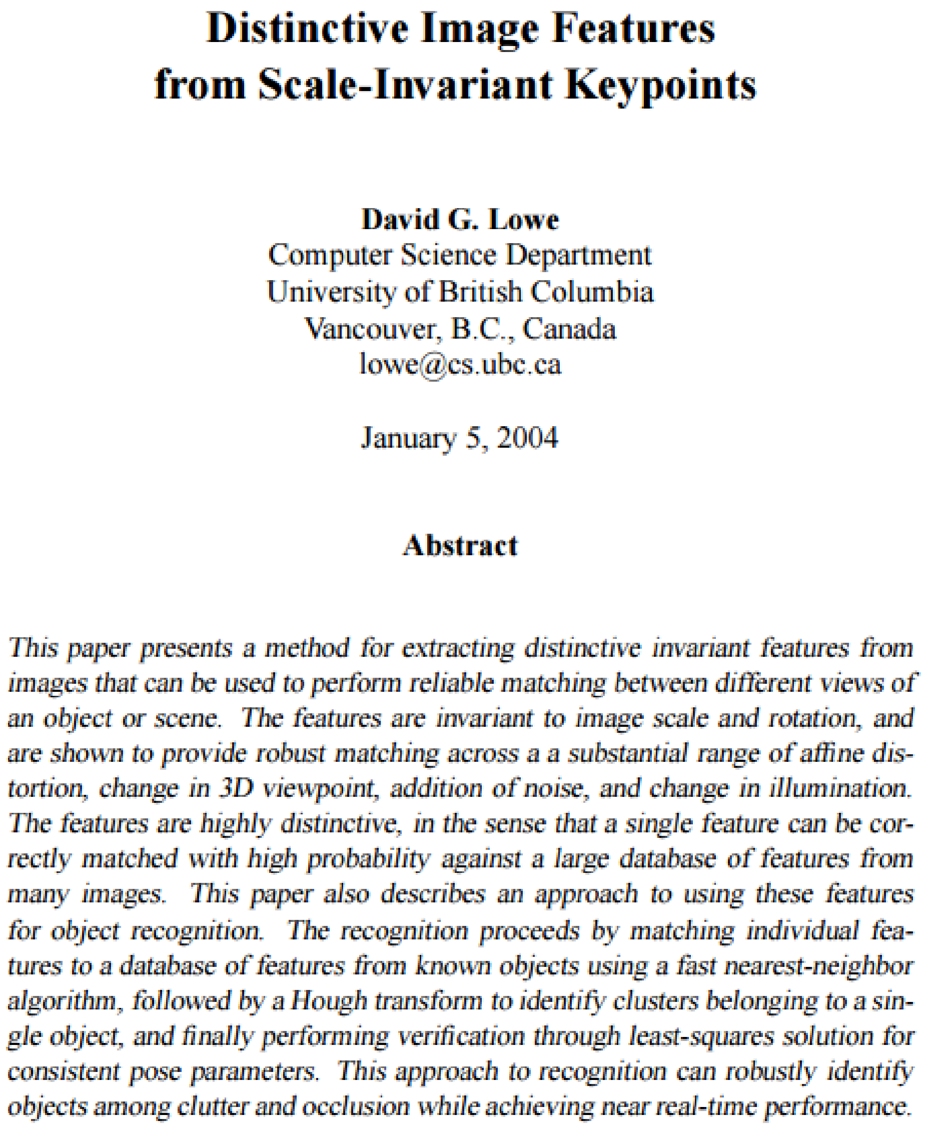
VL 10: SIFT

Scale Invariant Feature Transform

Tim Landgraf, WS 2015/16, Computer Vision



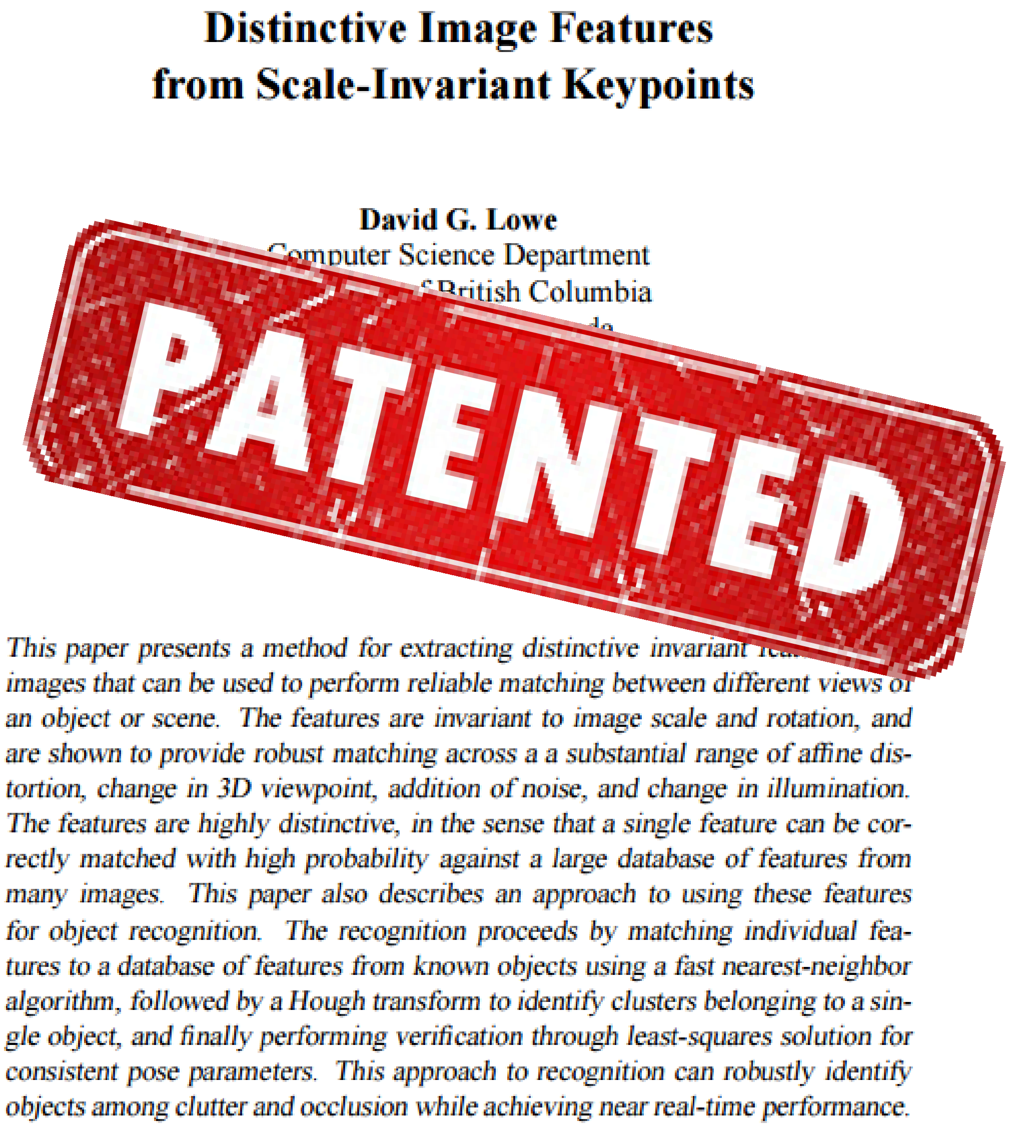
Distinctive Image Features from Scale-Invariant Keypoints

David G. Lowe
Computer Science Department
University of British Columbia
Vancouver, B.C., Canada
lowe@cs.ubc.ca

January 5, 2004

Abstract

This paper presents a method for extracting distinctive invariant features from images that can be used to perform reliable matching between different views of an object or scene. The features are invariant to image scale and rotation, and are shown to provide robust matching across a a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. The features are highly distinctive, in the sense that a single feature can be correctly matched with high probability against a large database of features from many images. This paper also describes an approach to using these features for object recognition. The recognition proceeds by matching individual features to a database of features from known objects using a fast nearest-neighbor algorithm, followed by a Hough transform to identify clusters belonging to a single object, and finally performing verification through least-squares solution for consistent pose parameters. This approach to recognition can robustly identify objects among clutter and occlusion while achieving near real-time performance.



Distinctive Image Features from Scale-Invariant Keypoints

David G. Lowe
Computer Science Department
f British Columbia
da

PATENTED

This paper presents a method for extracting distinctive invariant fea images that can be used to perform reliable matching between different views of an object or scene. The features are invariant to image scale and rotation, and are shown to provide robust matching across a a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. The features are highly distinctive, in the sense that a single feature can be correctly matched with high probability against a large database of features from many images. This paper also describes an approach to using these features for object recognition. The recognition proceeds by matching individual features to a database of features from known objects using a fast nearest-neighbor algorithm, followed by a Hough transform to identify clusters belonging to a single object, and finally performing verification through least-squares solution for consistent pose parameters. This approach to recognition can robustly identify objects among clutter and occlusion while achieving near real-time performance.

Applications

- Object recognition
- Tracking
- Stereo Vision / 3D Reconstruction
- Localization and Mapping
- Stitching

SIFT: Gliederung

- Scale Space Extrema Detection
- Keypoint Localization
- Keypoint Orientation Assignment
- Keypoint Descriptor

- Keypoint and Object Matching

Keypoint Requirements

- Scale-invariant
- Repeatable under different views
- Locate keypoints in „Scale Space“

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

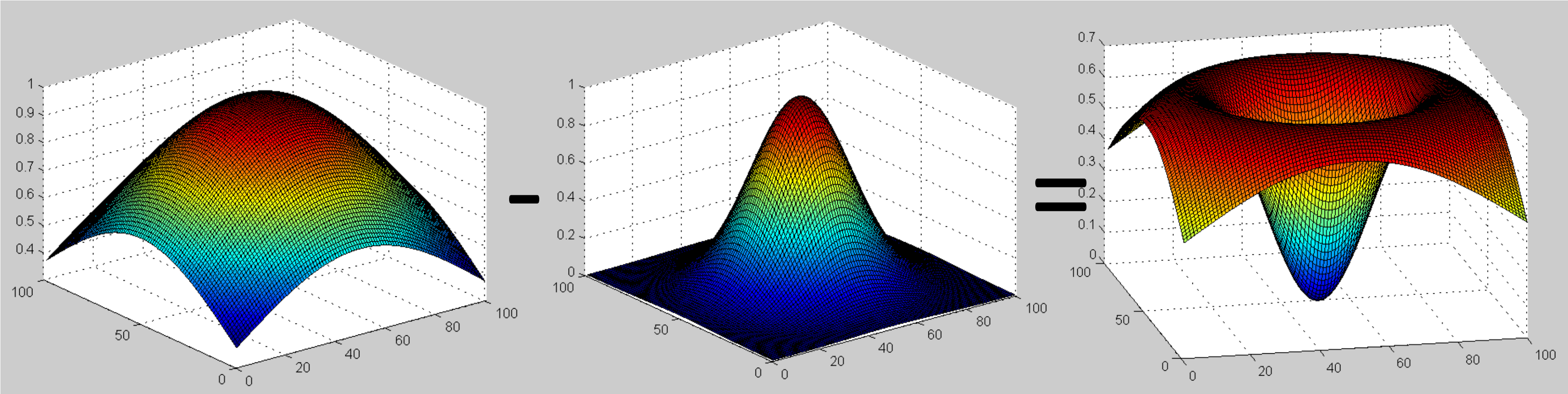
L(x,y,s) = (G(s)*I)(x,y)

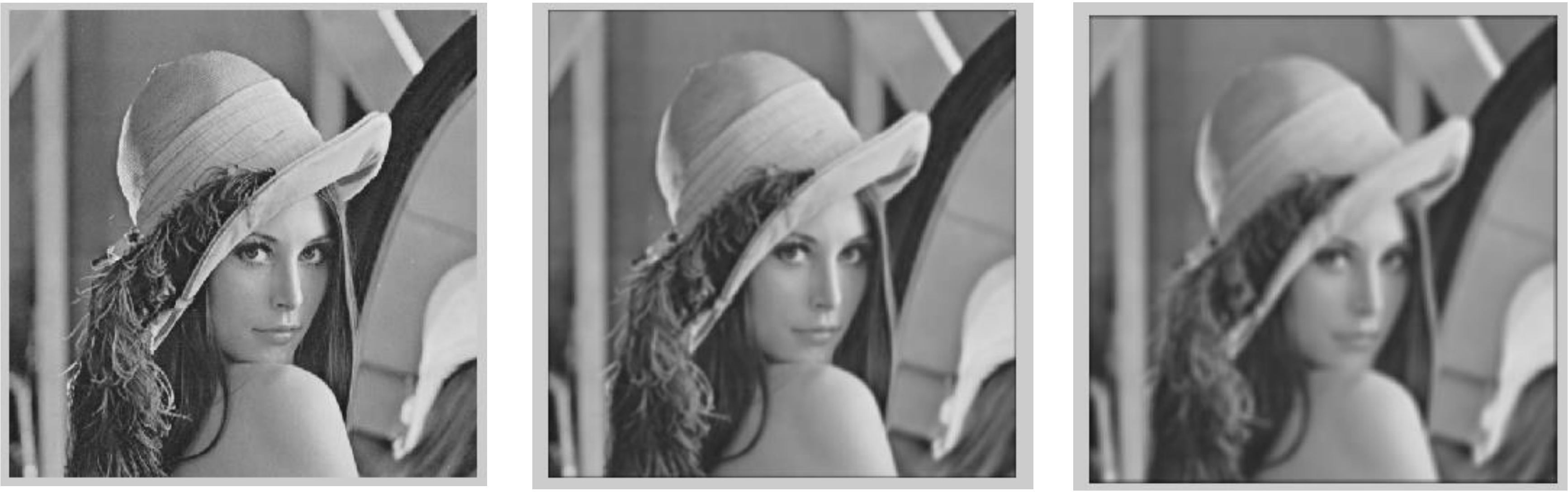
G(s) Gaussian with std s

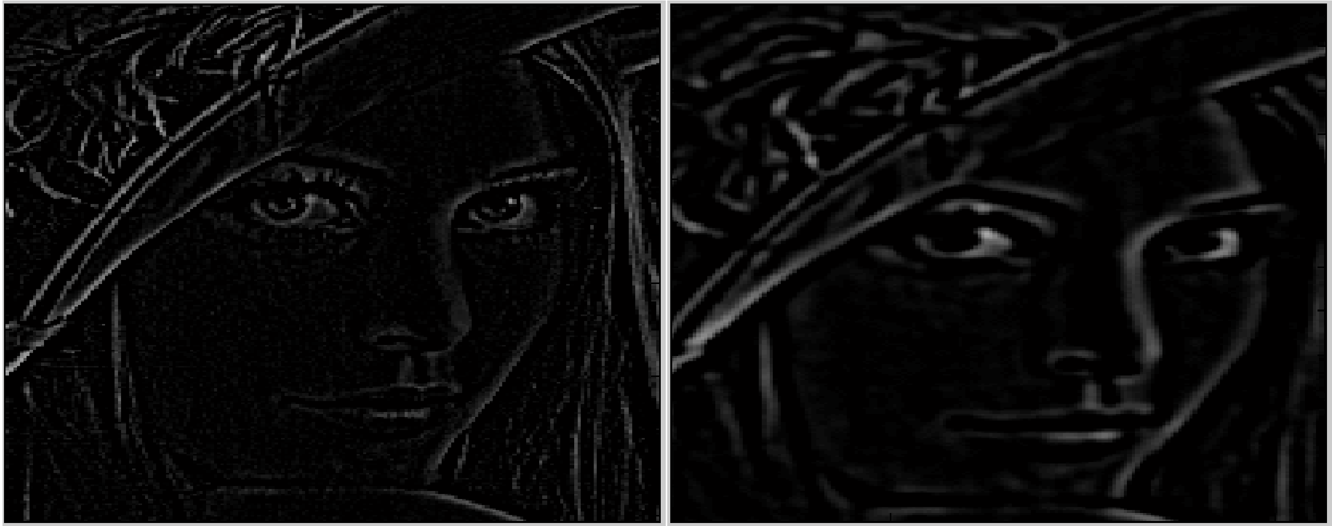
Keypoint Requirements

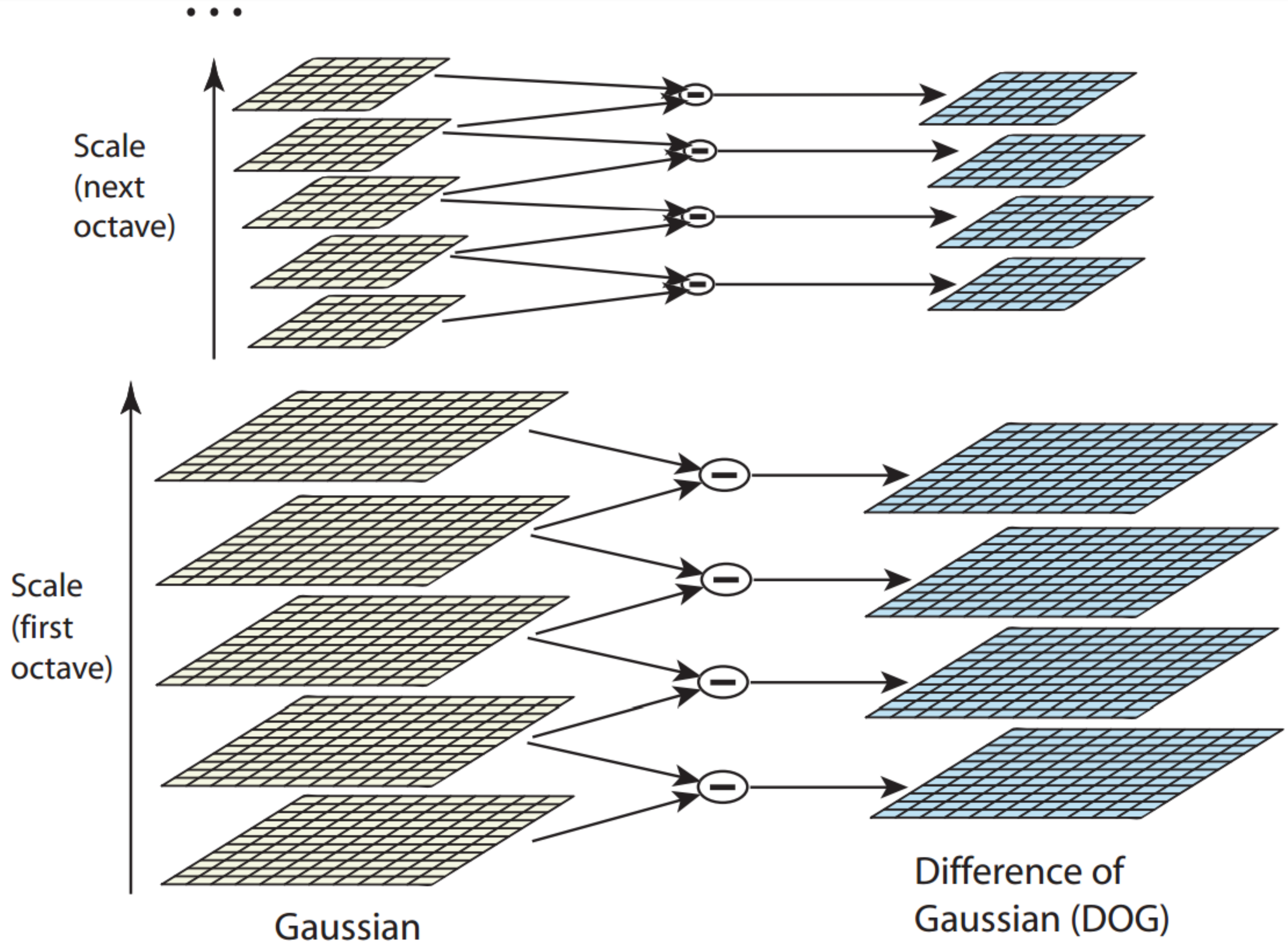
- Rotation-invariant

$$D(x,y,\sigma) \quad = \quad \underbrace{(G(x,y,k\sigma) - G(x,y,\sigma))}_{} * I(x,y)$$







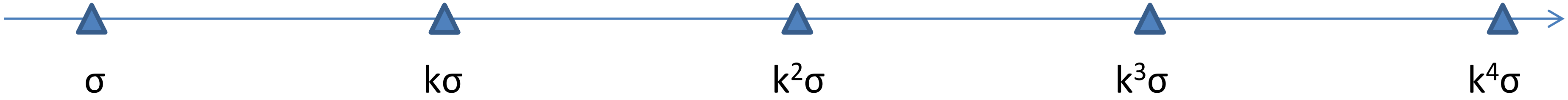


Scale Space

DoG pyramid

s := number of scale intervals

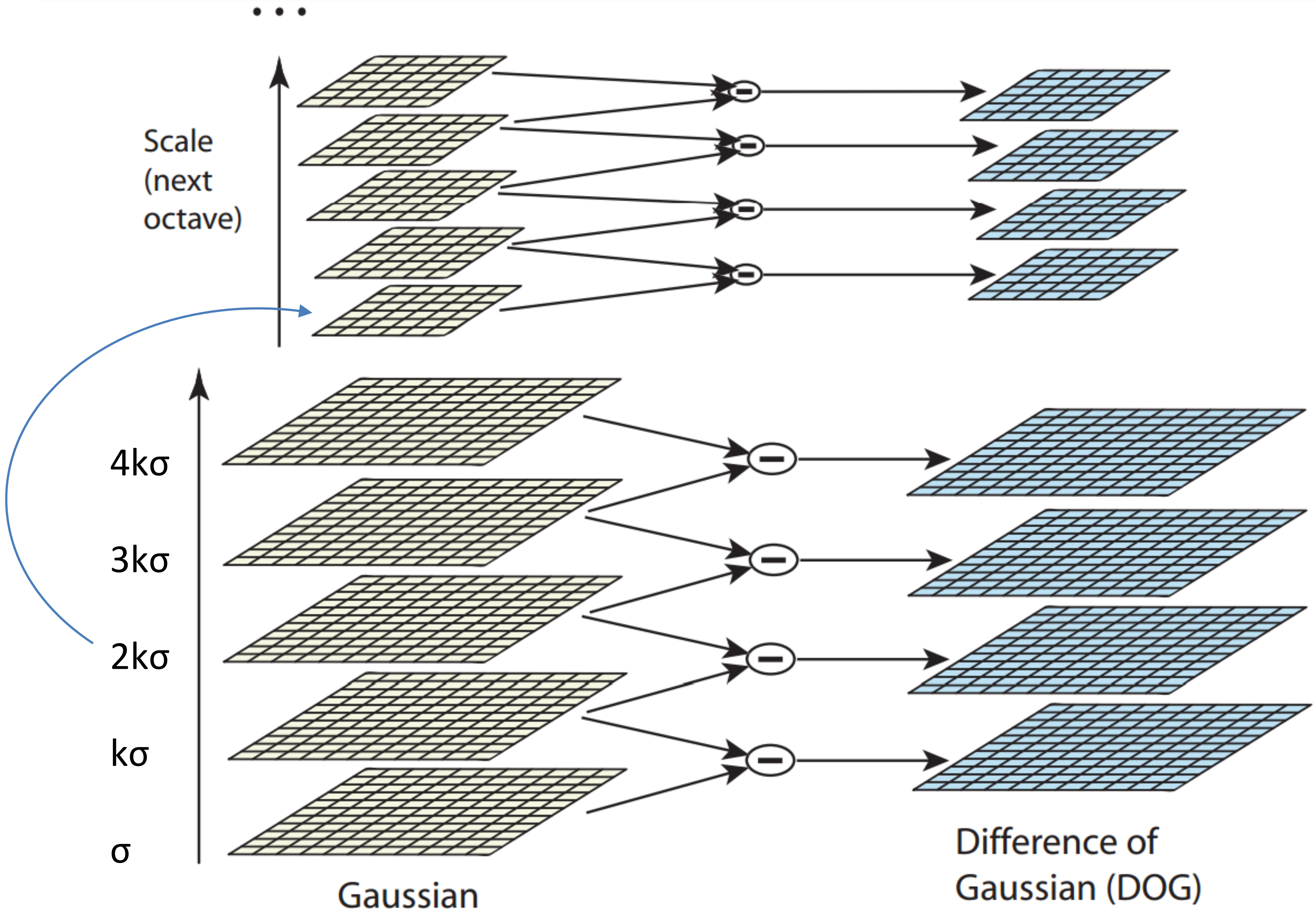
e.g. for s = 2 :



σ $k\sigma$ $k^2\sigma$ $k^3\sigma$ $k^4\sigma$

We need **s + 3 images per octave** for scale space extrema detection

$$\sigma(s,o) = \sigma_0 2^{o+s/S}, \qquad o \in \mathbb{Z}, \quad s = 0, ..., S-1$$



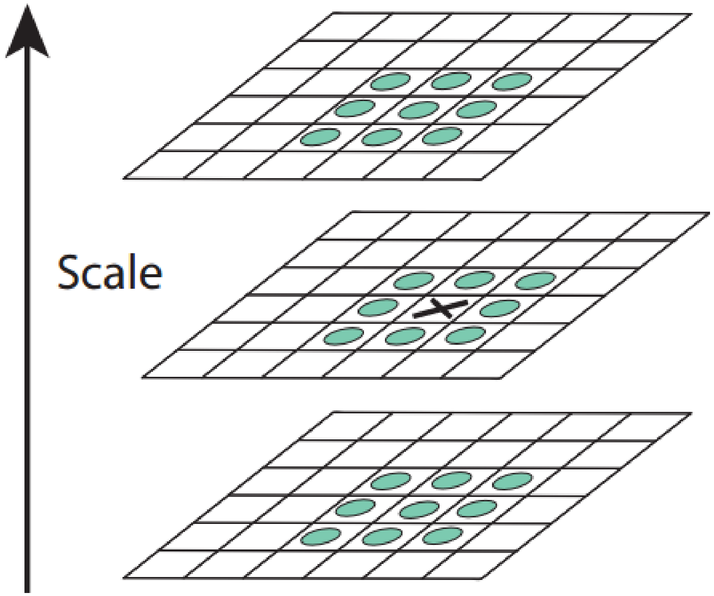
k = √2

Scale Space

DoG pyramid

Detection of Scale Space Extrema

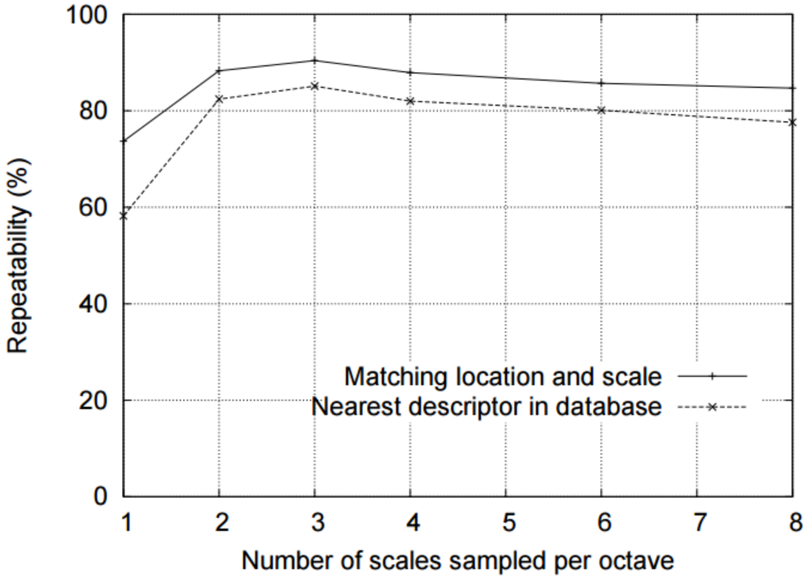
Detection of keypoint candidates in DoG pyramid

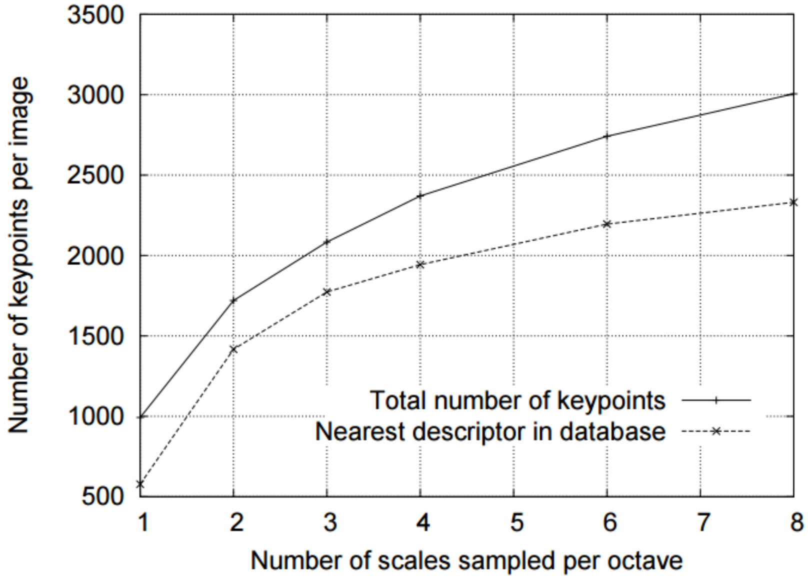


2*9 + 8 = 26
comparisons
(at max)

Implementation Details

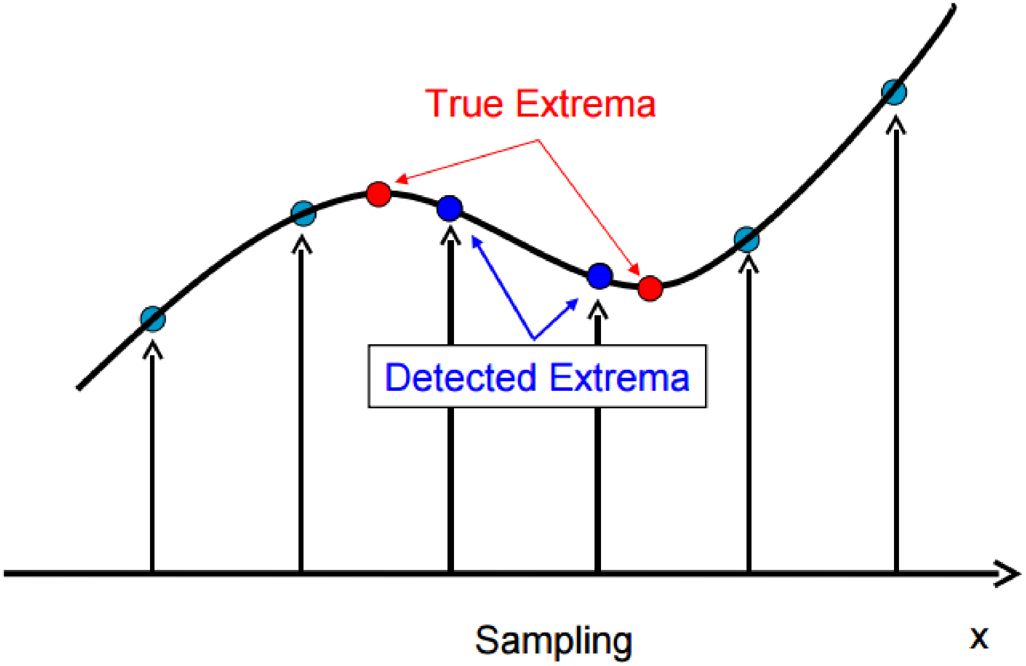
- Expand original image to make full use of high frequency features
- Smooth every first image per octave
- Number of scales per octave = 3 is good compromise between number of keypoints, repeatability and performance





Keypoint Localization

- Coarse sampling in the DoG pyramid for speed
- We need to localize the maxima



Taken from : http://www.inf.fu-berlin.de/lehre/SS09/CV/uebungen/uebung09/SIFT.pdf

Keypoint Localization

$$D(\mathbf{x}) = D + \frac{\partial D}{\partial \mathbf{x}}^T \mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathbf{T}} \frac{\partial^2 D}{\partial \mathbf{x}^2}\mathbf{x}$$

Taylor expansion +

set to 0

$$\hat{\mathbf{x}} = -\frac{\partial^2 D}{\partial \mathbf{x}^2}^{-1} \frac{\partial D}{\partial \mathbf{x}}.$$

x : offset from sample point

$\hat{\mathbf{x}}$: offset of extremum

- If |x^| > 0.5 (in x or y) re-do computation at next sample point

Rejection of Keypoints

- Low curvature (mostly edges)

1. $\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$ Hessian matrix (estimated by subtracting neighbors in D) ratio of eigenvalues proportional to curvature

2. $\mathrm{Tr}(\mathbf{H}) = D_{xx} + D_{yy} = \alpha + \beta,$ following the approach by Harris (see lecture #8)

   $\mathrm{Det}(\mathbf{H}) = D_{xx}D_{yy} - (D_{xy})^2 = \alpha\beta$

3. $\frac{\mathrm{Tr}(\mathbf{H})^2}{\mathrm{Det}(\mathbf{H})} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(r\beta+\beta)^2}{r\beta^2} = \frac{(r+1)^2}{r}$ for $\alpha = r\beta$

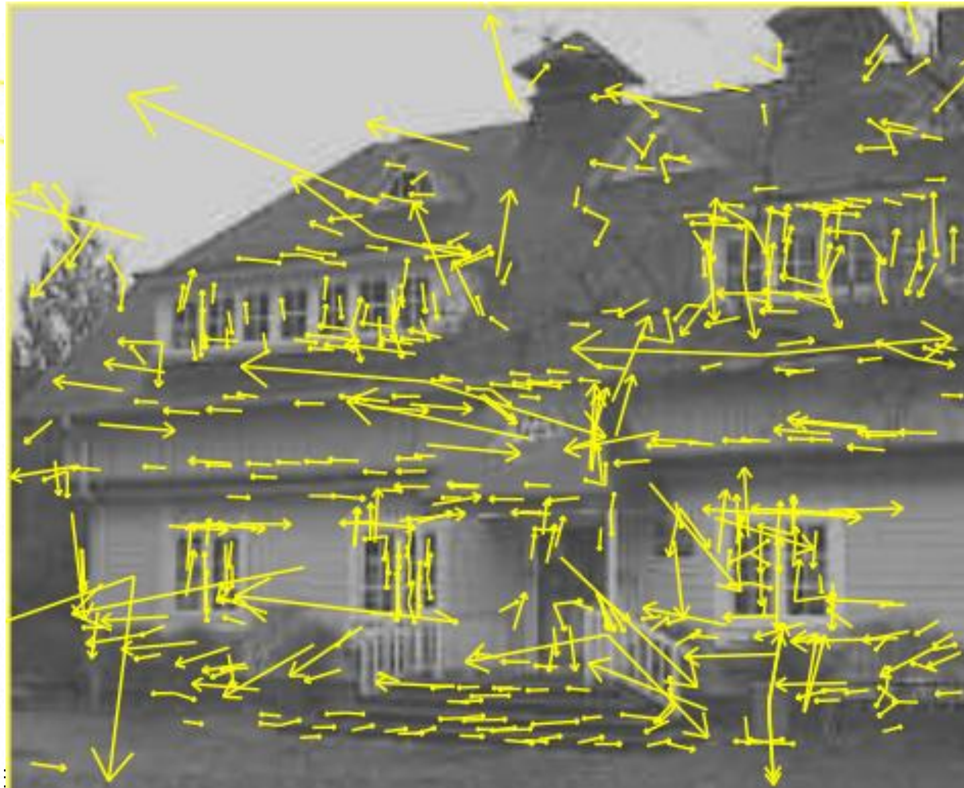
4. $\frac{\mathrm{Tr}(\mathbf{H})^2}{\mathrm{Det}(\mathbf{H})} < \frac{(r+1)^2}{r}$ for r = 10

Rejection of Keypoints

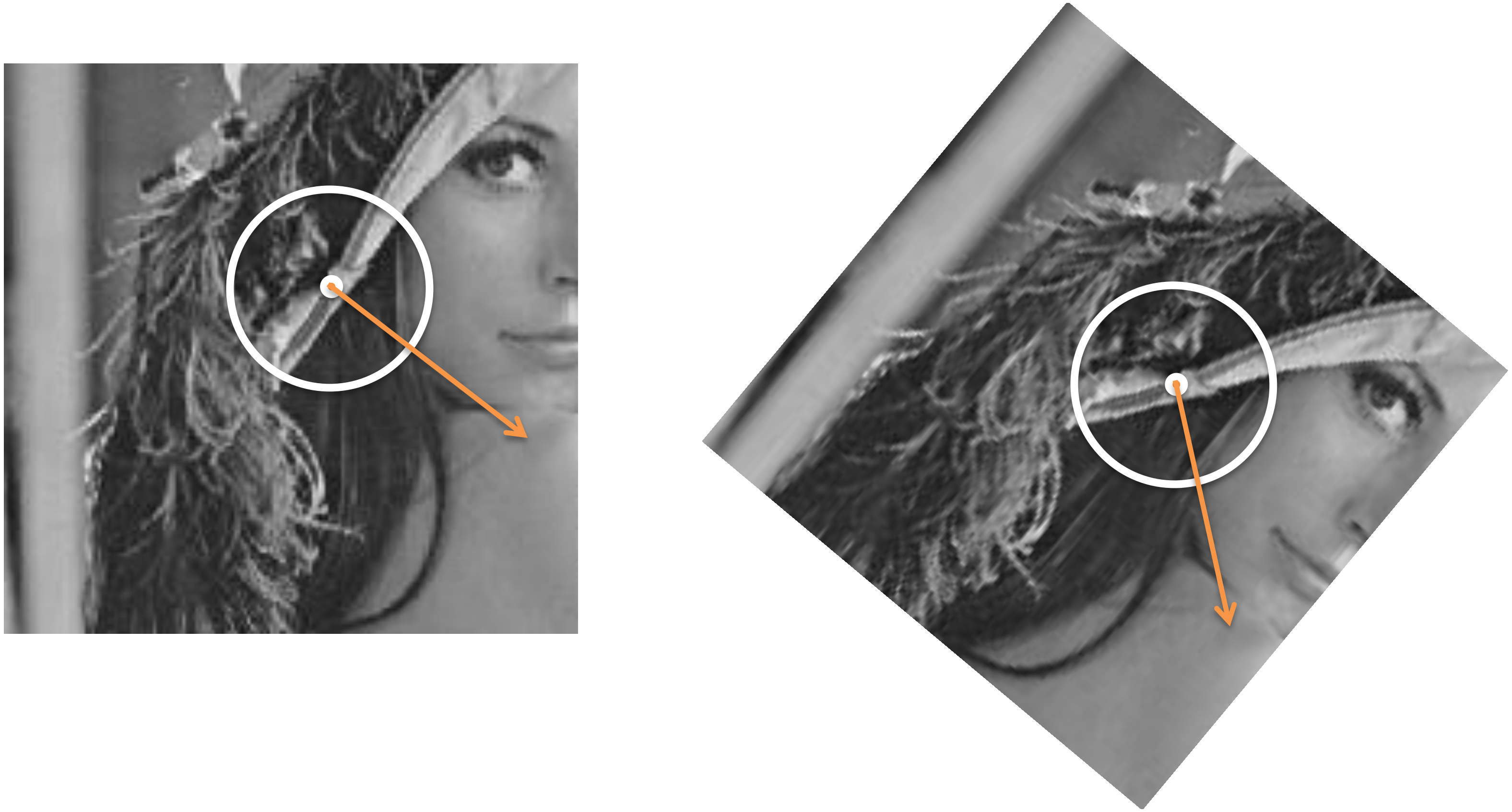
- Low contrast

$$D(\hat{\mathbf{x}}) = D + \frac{1}{2}\frac{\partial D^T}{\partial \mathbf{x}}\hat{\mathbf{x}}$$

(in the paper: |D(x^) | < 0.03)



Keypoint Orientation Assignment

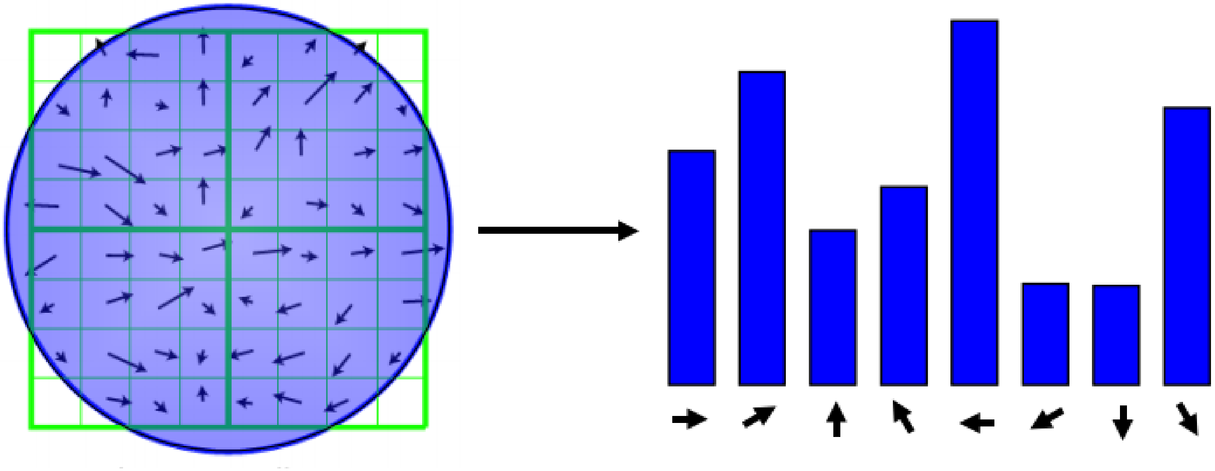


$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2}$$

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

Orientation Assignment

- Create gradient histogram (36 bins)
  - Weighted by magnitude and Gaussian window (σ is 1.5 times that of the scale of a keypoint)



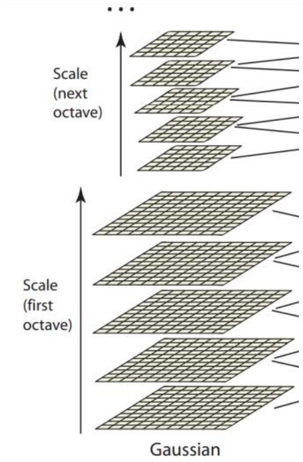
Taken from : http://www.inf.fu-berlin.de/lehre/SS09/CV/uebungen/uebung09/SIFT.pdf

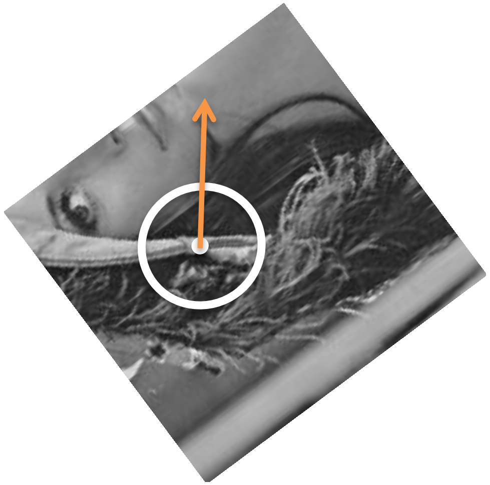
Orientation Assigment

- Look for high peaks in the histogram
- If more than one peak (within 80% magnitude of the maximum peak), create a keypoint for each of them
- Fit parabola to histogram values around peaks to refine peak location

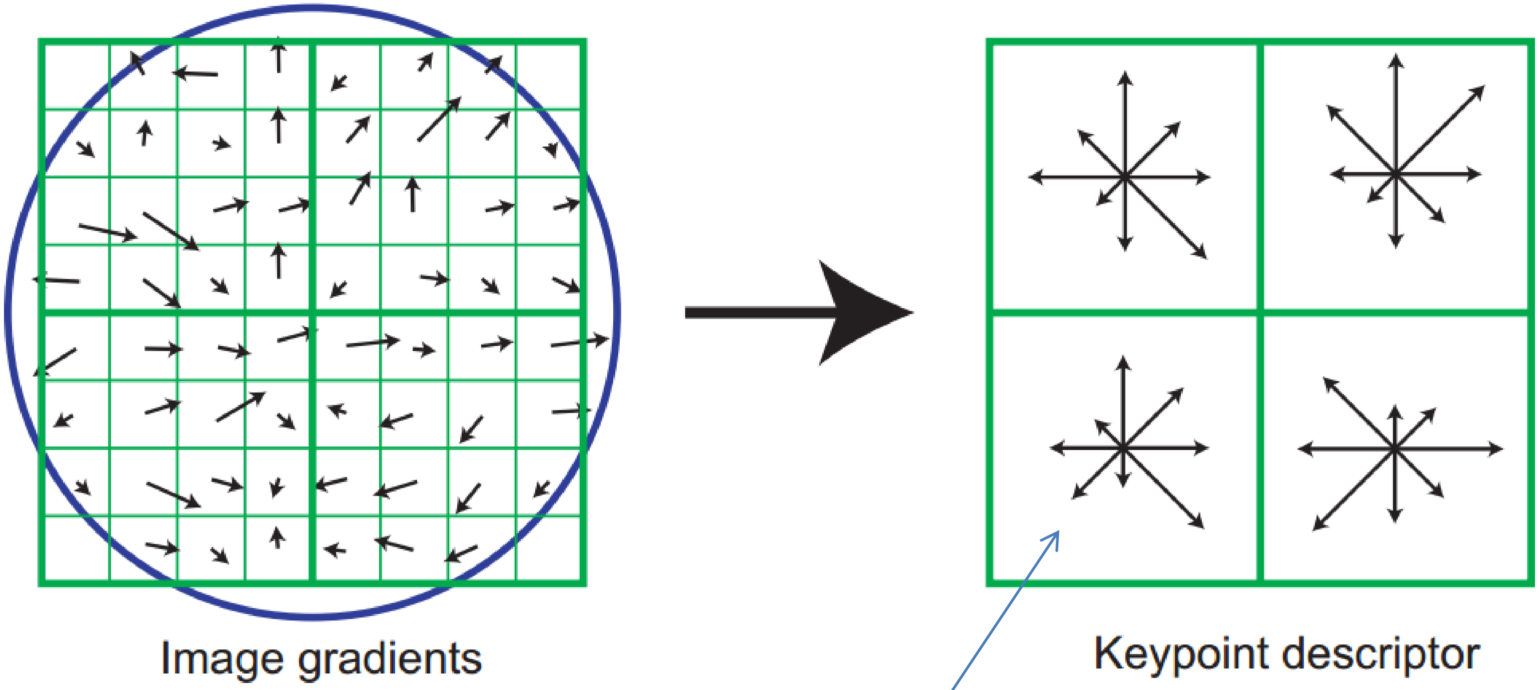
Keypoint Descriptor

- Goal: extract features of an environment around the keypoint that are stable under most transformations (scale and orientation are handled by previous steps)





Keypoint Descriptor



orientation histograms

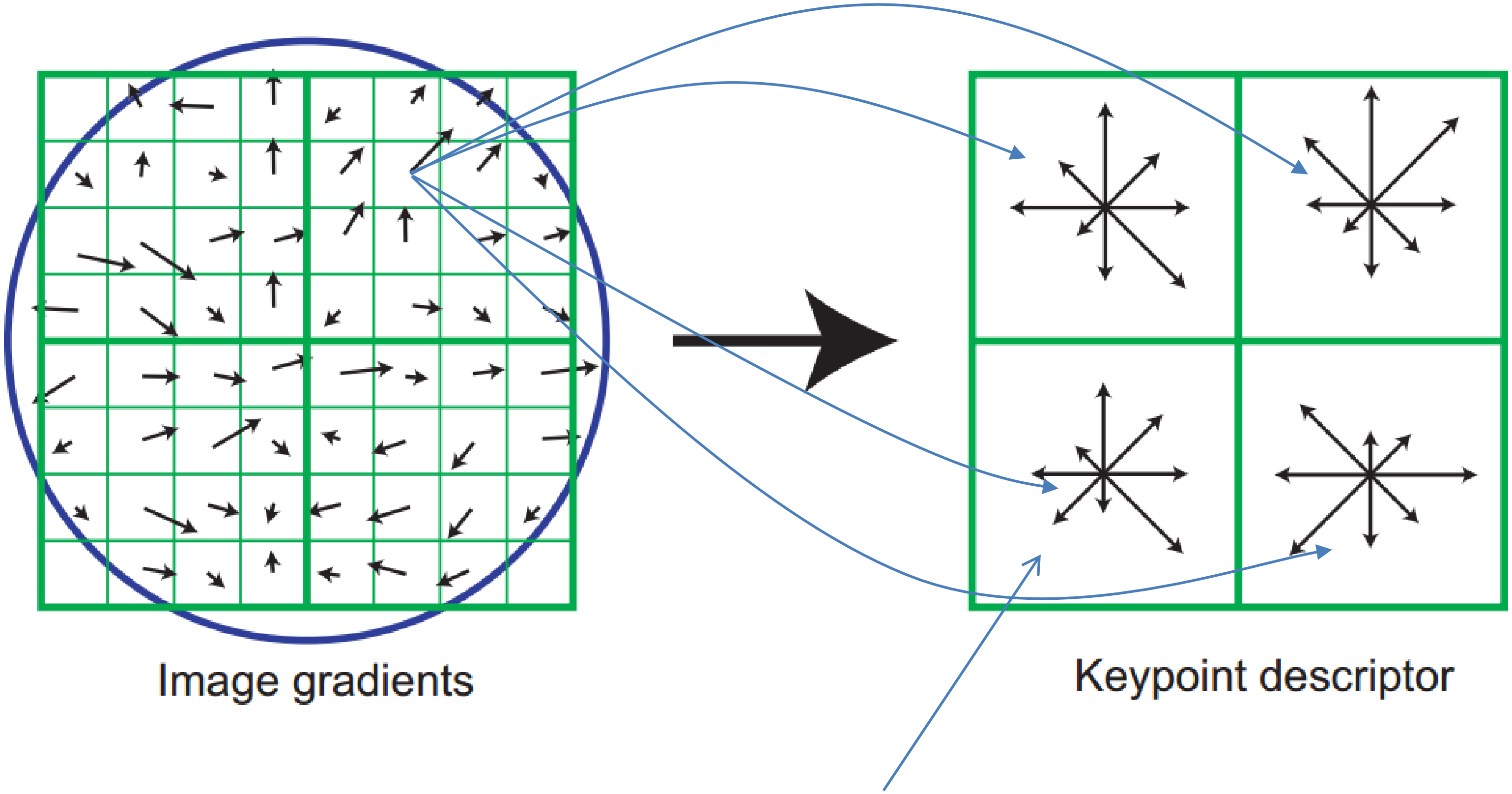
In the paper:

16 x 16 environment

4 x 4 descriptor

Keypoint Descriptor

interpolation, weighted with distance to histogram centers



orientation histograms

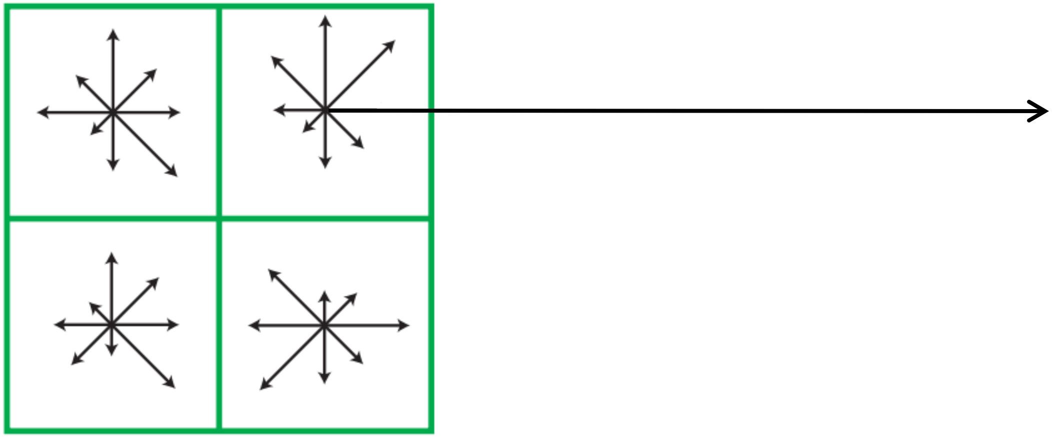
In the paper:

16 x 16 environment

4 x 4 descriptor

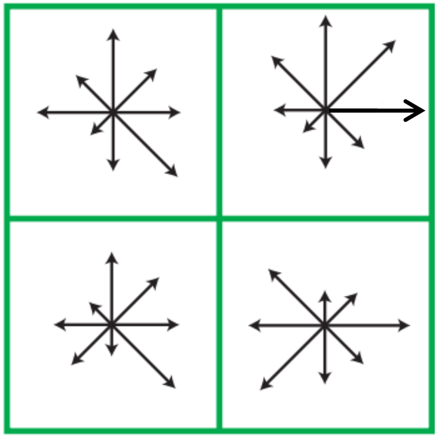
Keypoint Descriptor

- Orientation histograms are 8 bins wide, 4 x 4 histograms → 8 * 16 = 128 dimensional feature vector
- Vector is normalized to unit length, large values are truncated, re-normalize



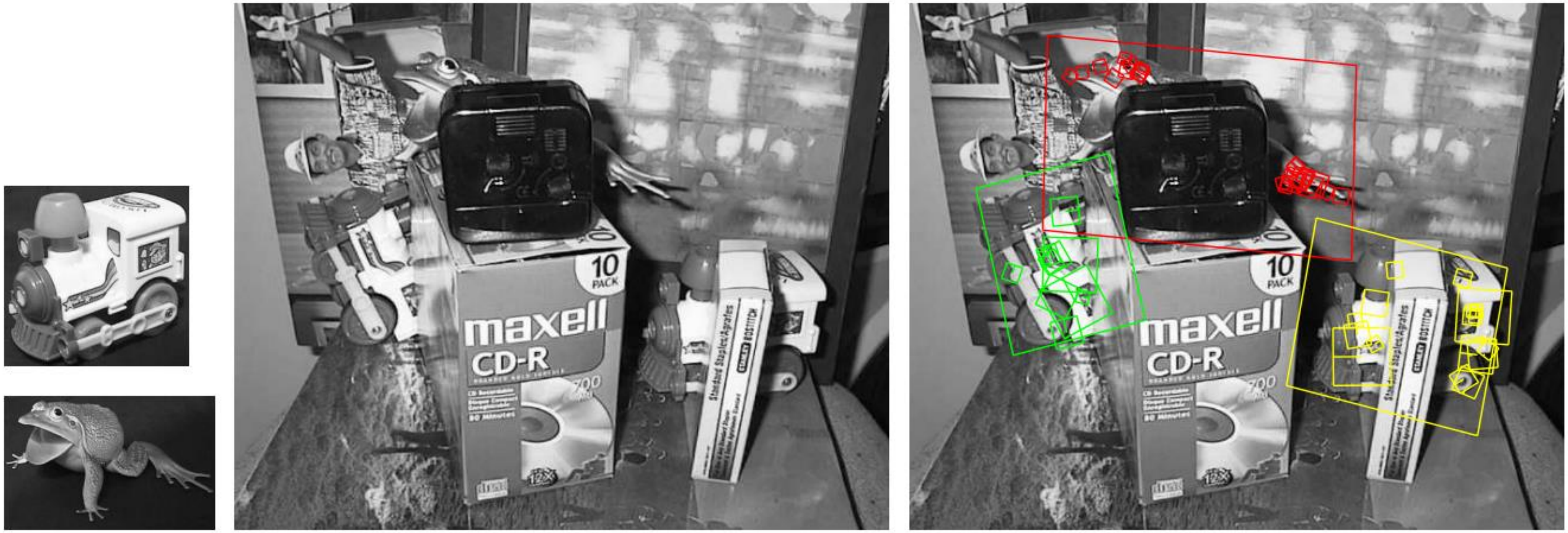
Keypoint Descriptor

- Orientation histograms are 8 bins wide, 4 x 4 histograms → 8 * 16 = 128 dimensional feature vector
- Vector is normalized to unit length, large values are truncated, re-normalize



Read yourself about

- Matching keypoints to a database of objects
- Find transformations that at least three keypoints undergo



Conclusions: SIFT …

- … is still one of the best feature descriptors!
- … uses many known data structures and algorithms, …, and got patented!
  - convolution, pyramids, Hessian matrix, Harris‘ estimate of the eigenvalue proportion, gradient histograms, k-d trees, Hough transform
- … has inspired similar approaches: PCA-SIFT, SURF, GLOH