# DD2423: Feature Detection II

## Interest point detection, Image descriptors, Image-based matching

Tony Lindeberg

Division of Computational Science and Technology
KTH Royal Institute of Technology
Stockholm, Sweden

November 19, 2021

# Feature detection

Previous lecture:

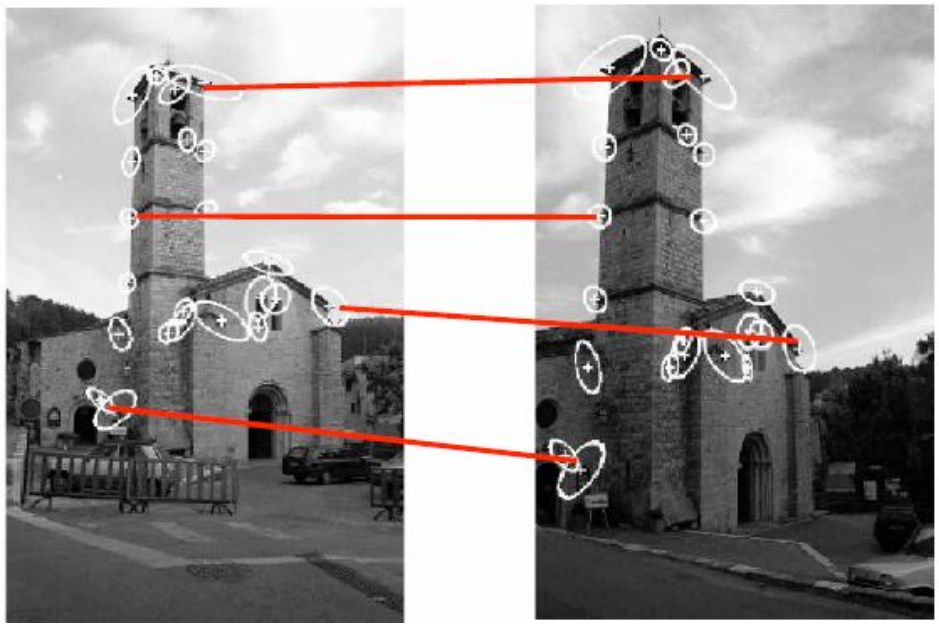
- General framework for feature detection based on: receptive fields, scale space and Gaussian derivatives
- Applied to the task of edge detection

This lecture:

- Detection of point-like image features: “interest point detection”’ (alternative terminology: “corner detection” and “blob detection”)
- Image descriptors at interest points for image-based matching and object recognition

# Motivation

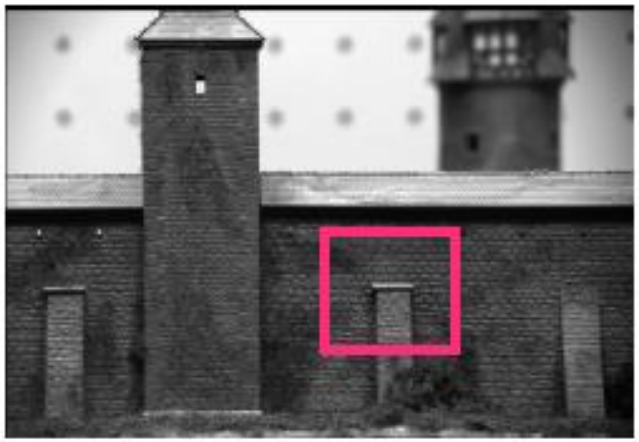
Feature-based methods for stereo and motion estimation typically require determination of correspondences between features in several images.

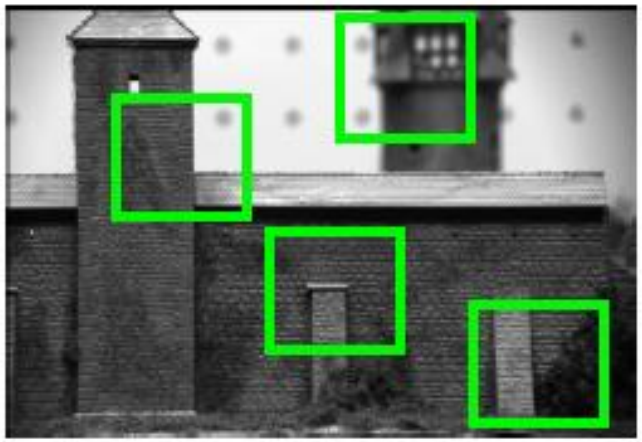


Feature-based methods for object recognition are based on correspondences between features in objects models and features in image data.

# Matching of image patches

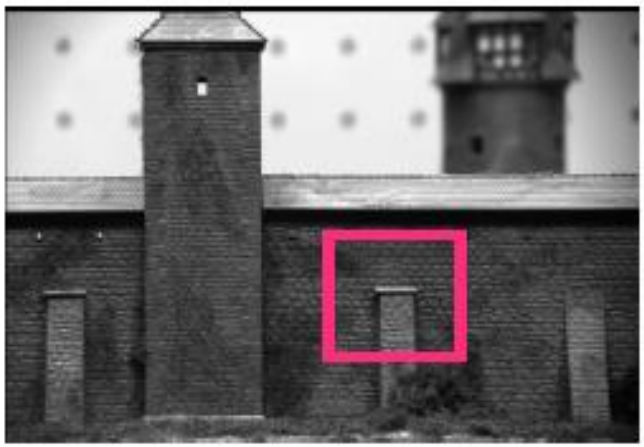
Elements to be matched are image patches of fixed size

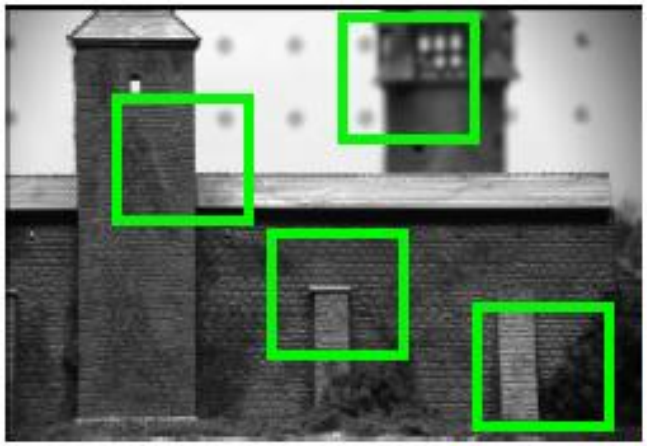




Task: find the best (most similar) patch in a second image

# Matching of image patches

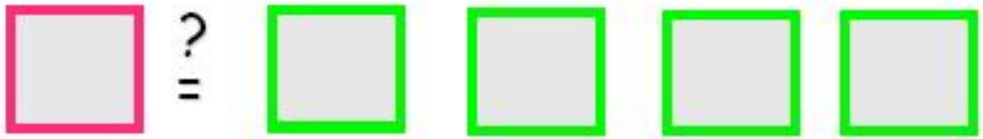




Inituition: this would be a good patch for matching, since it is very distinctive (there is only one patch in the second frame that looks similar).

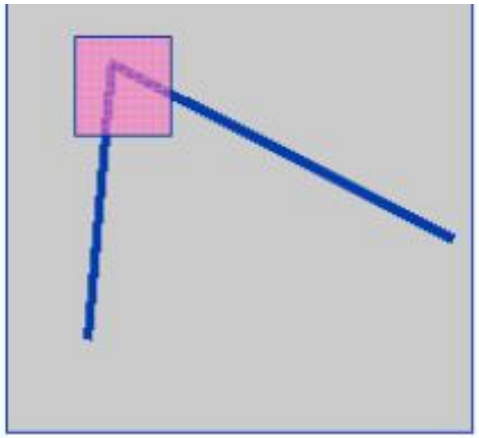
# Matching of image patches

Inituition: this would be a BAD patch for matching, since it is not very distinctive (there are many similar patches in the second frame)



# Interest points: Intuitive ideas

- We should easily recognize the point by looking at the intensity values within a small window.
- Shifting the window in *any direction* should yield a *large change* in appearence.



# How to determine good features to match

Basic requirements of interest points:

- clear preferably mathematically well-founded definition
- have well-defined *position* in image space
- *be rich in information content* in a local neighbourhood such that the interest points can be reliably matched
- be *stable* under natural image transformations (viewing direction, viewing distance, illumination variations)
- be sufficiently *distinct* such that interest points corresponding to different physical points can be kept separate
- preferably have an attribute of *scale* to handle scale variations in the image domain and determine a locally adapted region of interest around each interest point

# Interest point detection

Common approaches to interest point detection:

- Harris corner detection (Harris and Stephens 1988)

  Complemented by Laplacian scale selection in Harris-Laplace operator (Mikolajczyk and Schmid 2004)

- Laplacian blob detection with automatic scale selection (Lindeberg 1998)

  Approximated by difference-of-Gaussians in SIFT (Lowe 2004)

- Determinant of Hessian blob detection with automatic scale selection (Lindeberg 1998)

  Approximated by Haar wavelets in SURF (Bay et al 2008)

# Second-moment matrix

Idea: Accumulate statistics of local directions in regional neighbourhood around every image point

- With vector notation:
  $\mu(x;\ t, s) = \int_{\mathbb{R}^2} (\nabla L(\cdot;\ t))\, (\nabla L(\cdot;\ t))^T\, g(x - q;\ s)\, dx$
- In terms of explicit components:
  $\mu(x, y;\ t, s) =$
  $$\int_{(\xi,\eta)\in\mathbb{R}^2} \begin{pmatrix} L_x^2(\xi, \eta;\ t) & L_x(\xi, \eta;\ t)\, L_y(\xi, \eta;\ t) \\ L_x(\xi, \eta;\ t)\, L_y(\xi, \eta;\ t) & L_y^2(\xi, \eta;\ t) \end{pmatrix} g(x-\xi, y-\eta;\ s)\, d\xi\, d\eta$$

where

- $t$ is a local scale for computing derivatives
- $s$ is an integration scale for computing statistics of gradient directions
- $g(\cdot;\ s)$ is a regional window function over which statistics is accumulated

# The Harris operator

- Given the second-moment matrix $\mu$ with eigenvalues $\lambda_1, \lambda_2 \geq 0$, compute the Harris measure

$$H = \det \mu - k \, (\mathrm{trace} \, \mu)^2 = \lambda_1 \lambda_2 - k \, (\lambda_1 + \lambda_2)^2$$

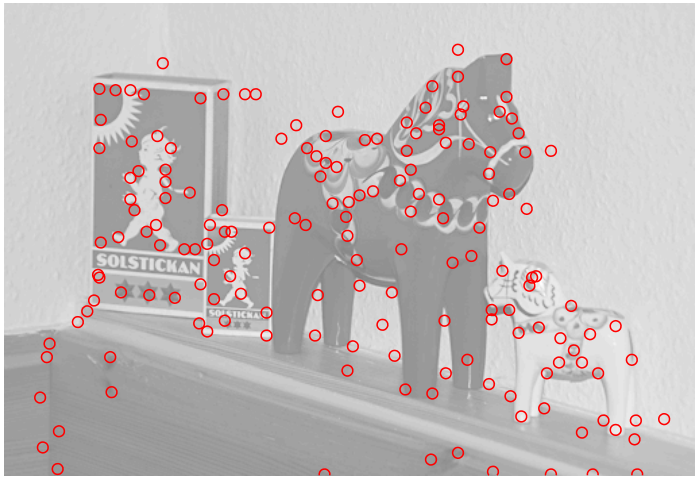
  where

  - $k$ is a constant in range $k \in [0, 1/4]$ with preferred value $k \in [0.04, 0.06]$

- Then, detect thresholded local maxima of $H$ with $H \geq H_0 > 0$

(Harris and Stephens 1988)

# The Harris operator


Figures from Lindeberg (2015) “Image matching using generalized scale-space interest points”, Journal of Mathematical Imaging and Vision 52(1): 3-36.

# Properties of the second-moment matrix

Eigenvalues $\lambda_1 \geq \lambda_2 \geq 0$ of $\mu$:

- Smooth image region without stronger edge structures:
  - Both eigenvalues $\lambda_1$ and $\lambda_2$ small
- Along a straight edge
  - $\lambda_1 >> \lambda_2$
- At image corner with two or more dominant directions
  - Both eigenvalues $\lambda_1$ and $\lambda_2$ large

The definition of the Harris measure

$$H = \det \mu - k\,(\mathrm{trace}\,\mu)^2 = \lambda_1 \lambda_2 - k\,(\lambda_1 + \lambda_2)^2$$

in combination with a low threshold on $H \geq H_0$ implies that
both eigenvalues must be large for the Harris operator to respond

# Influence of the parameter $k$

Given the condition

$$H = \lambda_1 \lambda_2 - k\,(\lambda_1 + \lambda_2)^2 \geq 0$$

divide by $\lambda_1^2$

$$H = \frac{\lambda_2}{\lambda_1} - k\,\left(1 + \frac{\lambda_2}{\lambda_1}\right)^2 \geq 0$$

and solve for the ratio $\frac{\lambda_2}{\lambda_1}$ with $\lambda_1 > \lambda_2 > 0$ in terms of $k \Rightarrow$

$$\frac{2k}{1-2k+\sqrt{1-4k}} \leq \frac{\lambda_2}{\lambda_1} \leq 1$$

The parameter determines how different the eigenvalues $\lambda_1$ and $\lambda_2$ are allowed to be for the Harris operator to respond:
a higher value of $k$ implies a more restrictive corner detector

# Algorithmic steps: Harris corner detection

1. Compute partial derivatives $L_x$ and $L_y$ at some scale $t$
2. Compute products $L_x^2$, $L_x L_y$ and $L_y^2$ at every image point
3. Compute weighed sums of products $E(L_x^2)$, $E(L_x L_y)$ and $E(L_y^2)$ using window function at scale $s$

   $\Rightarrow$ second-moment matrix $\mu = \begin{pmatrix} E(L_x^2) & E(L_x L_y) \\ E(L_x L_y) & E(L_y^2) \end{pmatrix}$
4. Compute

   $H = \det \mu - k\,(\mathrm{trace}\,\mu)^2 = E(L_x^2)E(L_y^2) - (E(L_x L_y))^2 - k\left(E(L_x^2) + E(L_y^2)\right)^2$

   at every image point
5. Detect local extrema of $H$ that satisfy the thresholding criterion $H \geq H_0 > 0$

# Laplacian blob detection

Given a scale-space representation $L$ of an image $f$ obtained by Gaussian smoothing

$$L(\cdot;\ t) = g(\cdot;\ t) * f$$

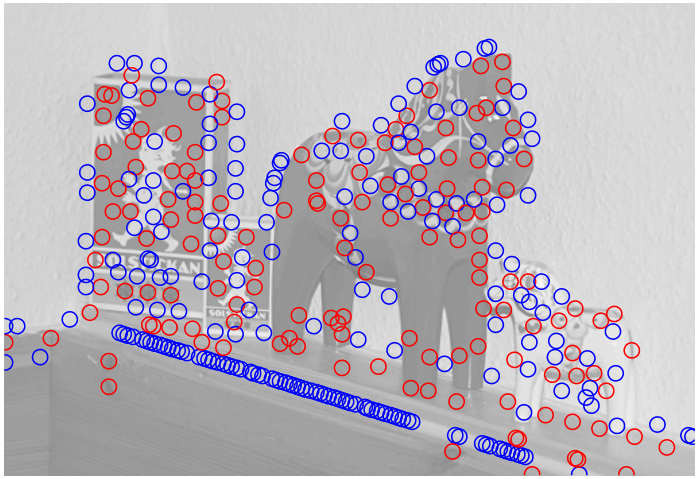
compute the *Laplacian operator*

$$\nabla^2 L = L_{xx} + L_{yy}$$

Spatial extrema of $\nabla^2 L$ regarded as blob responses with

$\nabla^2 L < 0 \Rightarrow$ “bright blob”
$\nabla^2 L > 0 \Rightarrow$ “dark blob”

# The Laplacian operator

Figures from Lindeberg (2015) “Image matching using generalized scale-space interest points”,

Journal of Mathematical Imaging and Vision 52(1): 3-36.

# Determinant of the Hessian blob detection

Given a scale-space representation $L$ of an image $f$ obtained by Gaussian smoothing

$$L(\cdot;\ t) = g(\cdot;\ t) * f$$

compute the *determinant of the Hessian matrix*

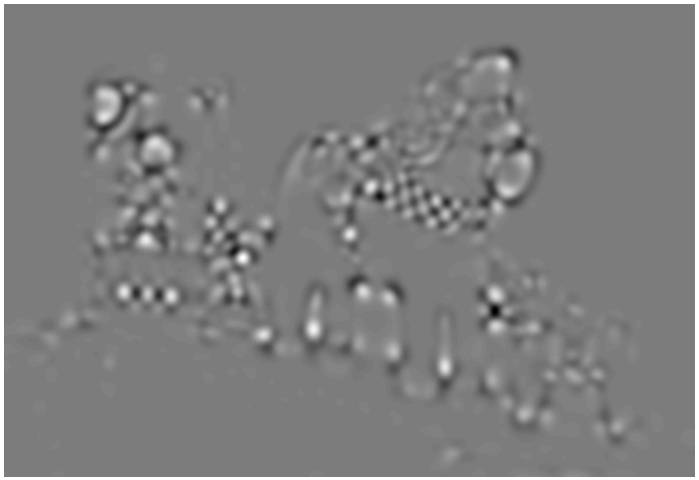
$$\det \mathcal{H}L = L_{xx} L_{yy} - L_{xy}^2$$

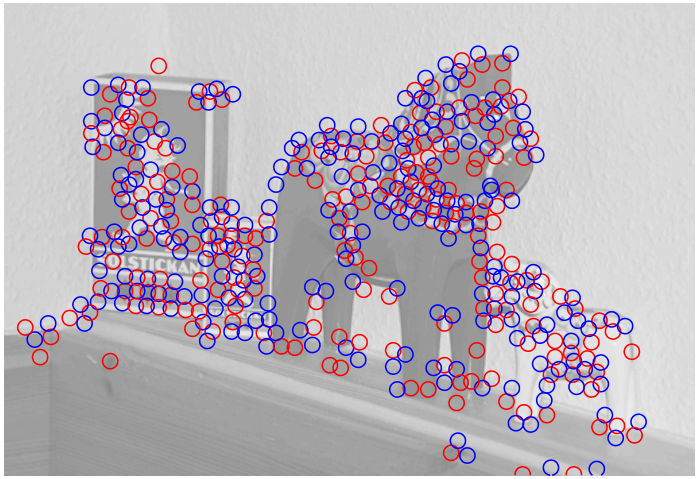
Positive local maxima of $\det \mathcal{H}L$ regarded as blob responses with

$\nabla^2 L < 0 \Rightarrow$ "bright blob"
$\nabla^2 L > 0 \Rightarrow$ "dark blob"

Negative local minima of $\det \mathcal{H}L$ regarded as saddle-like features

# The determinant of the Hessian operator



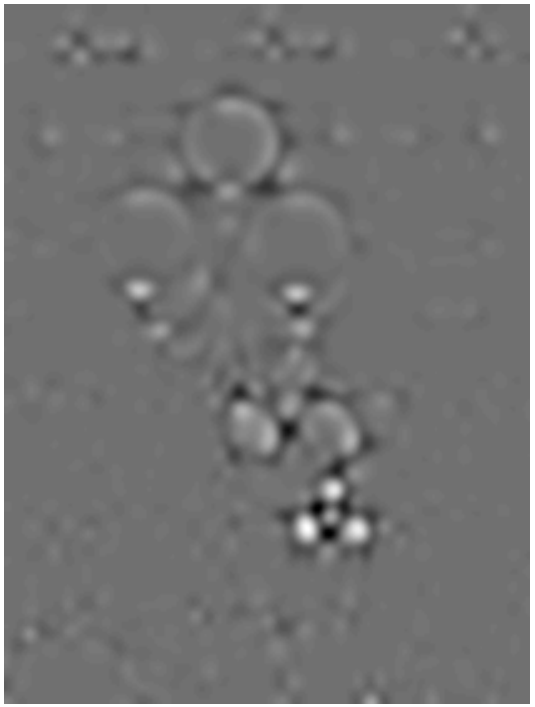
Figures from Lindeberg (2015) “Image matching using generalized scale-space interest points”, Journal of Mathematical Imaging and Vision 52(1): 3-36.

# Blob detection from differential invariants

original image

$\nabla^2 L$ at $t = 16$

$\det \mathcal{H} L$ at $t = 16$



Strong blob responses can be obtained provided that the scale level is adapted to the size of the image structures in the image domain

Figures from Lindeberg (2009) “Scale-space”, Encyclopedia of Computer Science and Engineering, IV:2495–2504.

# Laplacian *vs.* determinant of the Hessian interest points

In a coordinate frame aligned to the eigendirections of the Hessian matrix

$$\mathcal{H}L = \begin{pmatrix} L_{xx} & L_{xy} \\ L_{xy} & L_{yy} \end{pmatrix} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

The Laplacian and determinant of the Hessian operators correspond to

$$\nabla^2 L = L_{xx} + L_{yy} = \lambda_1 + \lambda_2$$
$$\det \mathcal{H}L = L_{xx} L_{yy} - L_{xy}^2 = \lambda_1 \lambda_2$$

Thereby,

- for the Laplacian operator to give a strong response it is sufficient that there is a strong response in one of the coordinate directions (either $\lambda_1$ or $\lambda_2$, implying risk for spurious responses near edges),
- for the determinant of the Hessian operator to give a strong response it is necessary that there are strong responses in both coordinate directions (both $\lambda_1$ and $\lambda_2$).

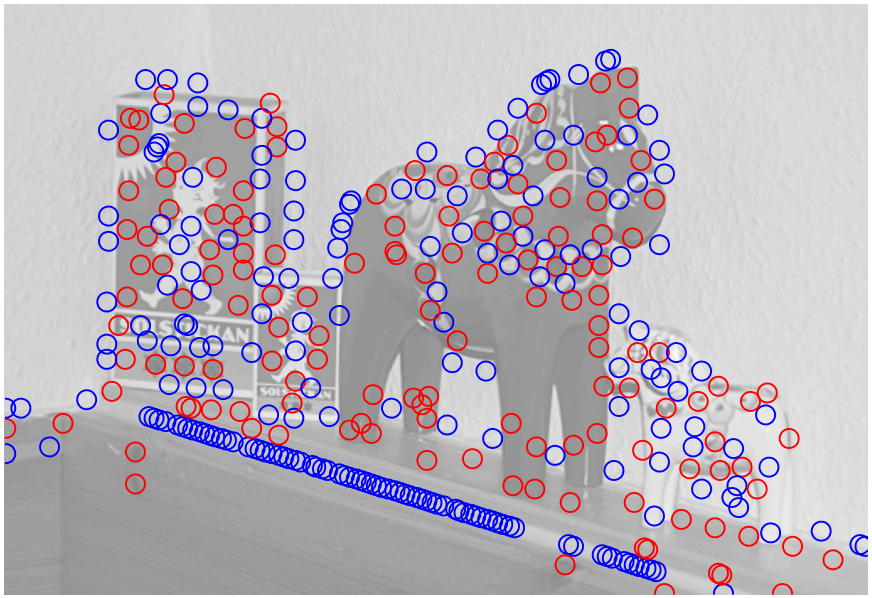
# Suppression of spurious edge responses

Perform *complementary thresholding* on the differential entity

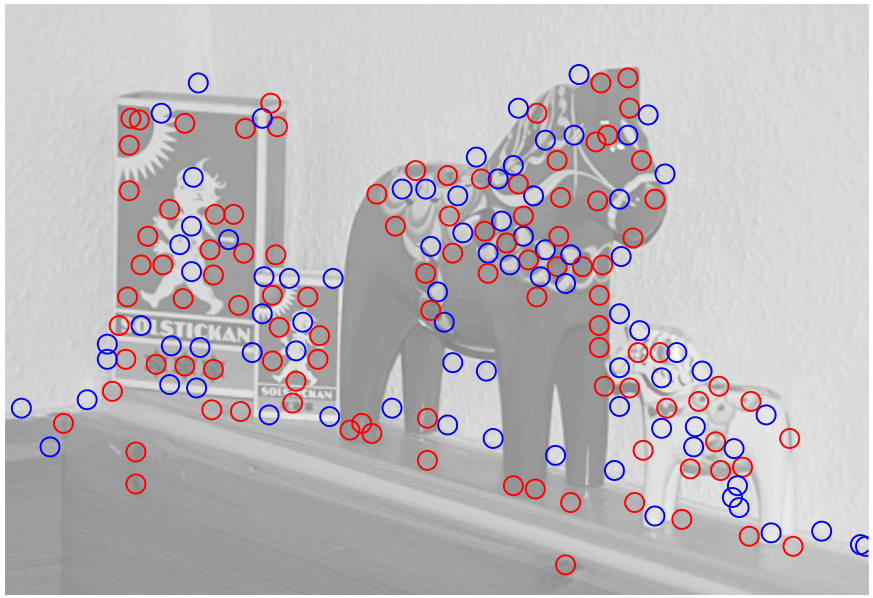
$$\mathcal{D}_1 L = L_{xx} L_{yy} - L_{xy}^2 - k\,(L_{xx} + L_{yy})^2 \geq 0$$

for some $k \in [0, 1/4[$ where suitable values may be taken in the range $k \in [0.04, 0.08[$.

$\nabla^2 L$ features



with thresholding on $\mathcal{D}_1 L > 0$



Figures from Lindeberg (2015) “Image matching using generalized scale-space interest points”, Journal of Mathematical Imaging and Vision 52(1): 3-36.

# Complementary thresholding: Explanation

In coordinate frame aligned to the eigendirections of the Hessian matrix

$$\mathcal{D}_1 L = L_{xx} L_{yy} - L_{xy}^2 - k\,(L_{xx} + L_{yy})^2 = \lambda_1 \lambda_2 - k(\lambda_1 + \lambda_2)^2 \geq 0$$

does for $\lambda_2 \geq \lambda_1 > 0$ correspond to the condition

$$\frac{\lambda_1}{\lambda_2} - k \left(1 + \frac{\lambda_1}{\lambda_2}\right)^2 > 0$$

Since the left hand side in becomes negative if $\lambda_1/\lambda_2$ is close to zero, this inequality cannot be satisfied if the eigenvalues differ too much in magnitude.

Thus, the criterion $\mathcal{D}_1 L > 0$ can only be satisfied if the ratio of the eigenvalues $\lambda_1/\lambda_2$ of $\mathcal{H}L$ is sufficiently close to one, in other words only if the image structures have strong information in two different coordinate directions.

Further details in Lindeberg (2015) “Image matching using generalized scale-space interest points”,

Journal of Mathematical Imaging and Vision 52(1): 3-36.

# Need for scale selection

Major problems when applying interest point detectors at a single fixed scale:

- How choose appropriate scale level(s)?

  A single scale level may not be sufficient to capture
  the relevant image structures in a given image

- Resulting image features may be strongly dependent on the imaging conditions, e.g., if the same object is seen from two different distances

$\Rightarrow$ Desirable to include mechanism for:

  - selecting scale levels automatically and
  - computing scale-invariant image features

# Scale selection from normalized derivatives

General methodology:

1. Define $\gamma$-*normalized derivatives* according to

$$\partial_\xi = t^{\gamma/2}\, \partial_x \qquad \partial_\eta = t^{\gamma/2}\, \partial_y$$

2. Detect *local extrema over scale* of normalized differential invariants $D_{\gamma-norm}L$ obtained by replacing the regular Gaussian derivatives $L_{x^\alpha y^\beta}$ by corresponding scale-normalized derivatives $L_{\xi^\alpha \eta^\beta}$

It can be shown that:

- local extrema over scale are *preserved under scaling transformations* $f'(x', y') = f(x, y)$ with $(x', y') = (sx, sy)$
- scale estimates $\hat{t} = \operatorname{argmax} \mathcal{D}_{\gamma-norm}L$ are *transformed in a scale covariant way*:

$$\hat{t}' = s^2\, \hat{t}$$

(Lindeberg 1998)

# Scale invariance by local scale selection

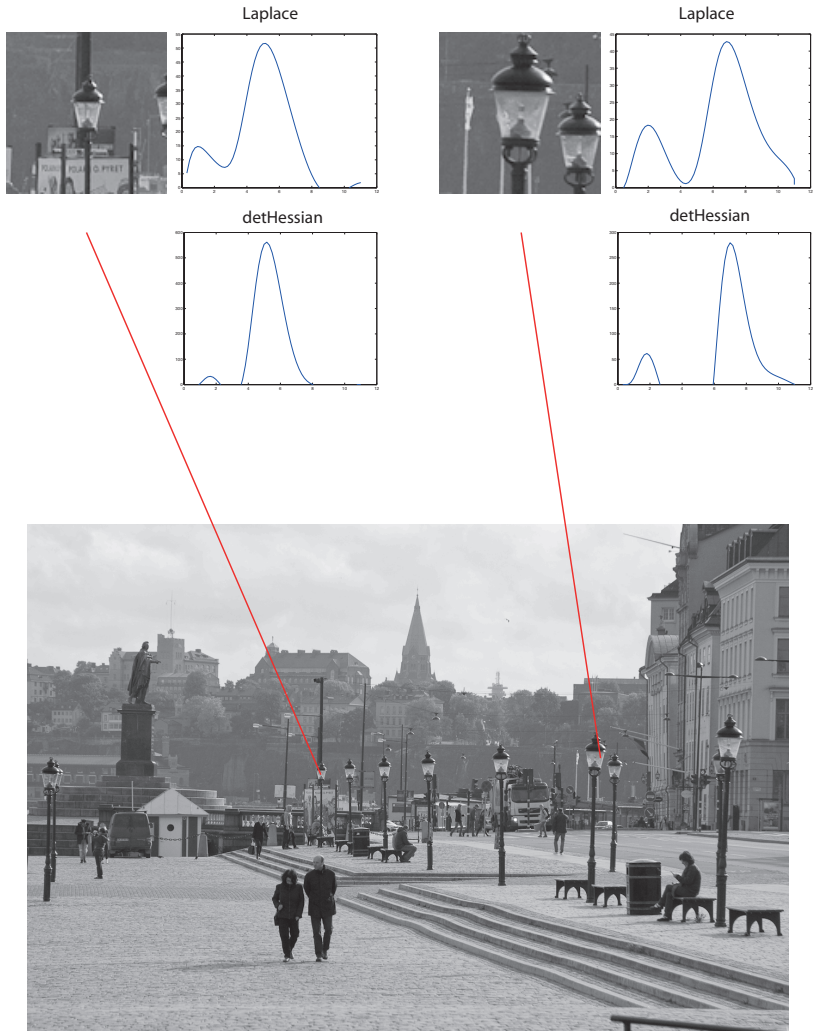



Figure from Lindeberg (2013) “Invariance of visual operations at the level of receptive fields”, PLOS ONE 8(7): e66990:1–33.

# Scale invariance by local scale selection

Local scale selection makes it possible to estimate the size of local image structures from the behaviour of image structures over scale in scale-space

Scale invariance carries over to:

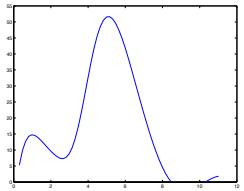
- features computed at a scale proportional to $\hat{t}$
- image descriptors computed with window size proportional to $\hat{t}$.

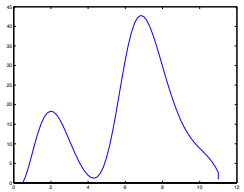
Can be used for defining a *scale invariant reference frame*, such that any computations relative to such a reference frame will also be scale invariant.

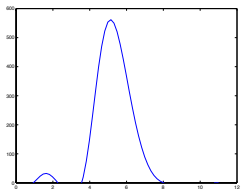
Applications: feature detection, image descriptors (SIFT, SURF, GLOH, etc)

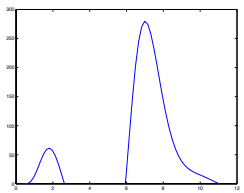
# Scale invariant reference frame

*original window* *scale estimate* *scale normalized*


Figures from Lindeberg (2013) “Invariance of visual operations at the level of receptive fields”, PLOS ONE 8(7): e66990:1–33.

# Scale selection applied to Gaussian blob

Input image = Gaussian blob:

$$f(x, y) = g(x, y;\ t_0) = \frac{1}{2\pi t_0} e^{-\frac{x^2+y^2}{2t_0}}$$

Scale-space representation by semi-group property:

$$L(x, y;\ t) = g(x, y;\ t) * g(x, y;\ t_0) = g(x, y;\ t_0 + t) = \frac{1}{2\pi(t_0+t)} e^{-\frac{x^2+y^2}{2(t_0+t)}}$$

Scale-normalized Laplacian at origin when $\gamma = 1$:

$$\nabla^2_{norm} L = t\,(L_{xx} + L_{yy}) = t\left(\left(\frac{x^2-t_0+t}{(t_0+t)^2}\right) L + \left(\frac{y^2-t_0+t}{(t_0+t)^2}\right) L\right) = -\frac{t}{\pi(t_0+t)^2}$$

Differentiate with respect to scale:

$$\partial_t(\nabla^2_{norm} L) = 0 \quad \Rightarrow \quad \hat{t} = t_0 \qquad \textit{“measures size of Gaussian blob”}$$

# Scale selection applied to sine wave

Input signal = 1-D sine wave:

$$f(x) = \sin \omega_0 x$$

Scale-space representation from Fourier transform:

$$L(x;\ t) = e^{-\omega_0^2 t/2} \sin \omega_0 x$$

Amplitude of $m$:th order spatial derivative

$$L_{\xi^m,max} = t^{m/2}\, \omega_0^m\, e^{-\omega_0^2 t/2}$$

Differentiate with respect to scale and express in terms of $\sigma = \sqrt{t}$ and wavelength $\lambda_0 = \frac{2\pi}{\omega_0}$:

$$\hat{\sigma} = \frac{\sqrt{m}}{2\pi} \lambda_0 \quad \Rightarrow \quad \textit{“scale estimate proportional to wavelength”}$$

# Interest points from scale-space extrema

General methodology:

- Detect points $(\hat{x}, \hat{y};\ \hat{t})$ in scale-space that are *simultaneously* local extrema with respect to both space and scale
- Such points are referred to as *scale-space extrema*
- Interest point detection with integrated scale selection mechanism

Differential entities for blob detection:

- Scale-normalized Laplacian:

$$\nabla^2_{norm} L = t\,(L_{xx} + L_{yy})$$

- Scale-normalized determinant of the Hessian:

$$\nabla^2_{norm} L = t^2\,(L_{xx}\,L_{yy} - L^2_{xy})$$

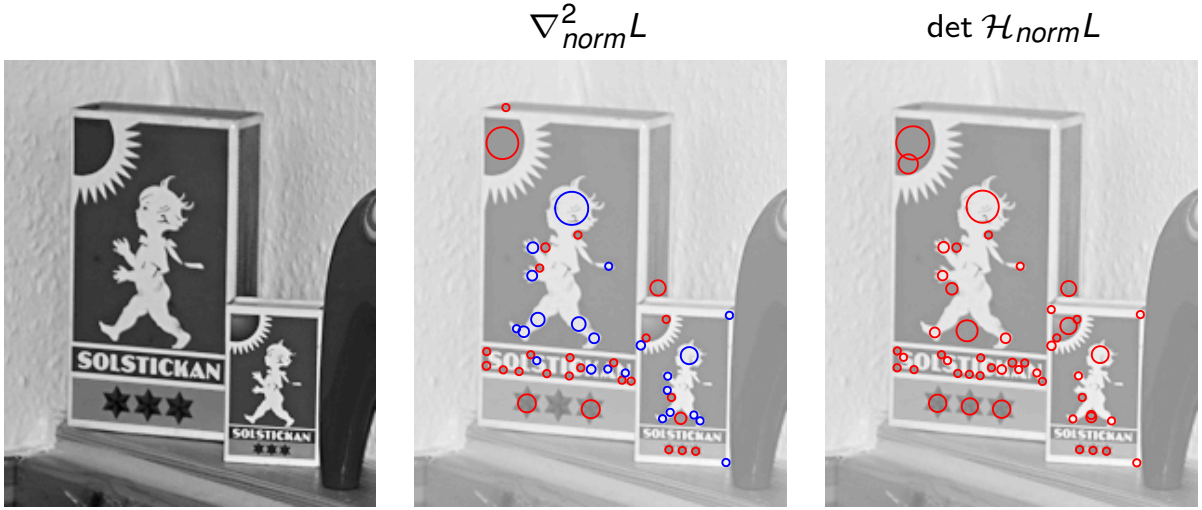
(Lindeberg 1998)

# Scale invariance

**General scaling property:** If the original image pattern $f$ is rescaled by uniform scaling factor $f'(x', y') = f(s\,x, s\,y)$, then

- a scale-space extremum in $f$ at $(x_0, y_0;\ t_0)$ is transformed to a scale-space extremum in $f'$ at $(x'_0, y'_0;\ t'_0) = (s\,x_0, s\,y_0;\ s^2 t_0)$

$\Rightarrow$ selected scale levels automatically adapt to scaling variations




Figures from Lindeberg (2014) “Scale selection”, Computer Vision: A Reference Guide, pages 701-713, Springer.

# Interest point detection from scale-space extrema

*Original grey-level image*

# Scale-invariant Laplacian interest points

*1000 strongest scale-space extrema of the Laplacian* $\nabla^2_{norm} L$

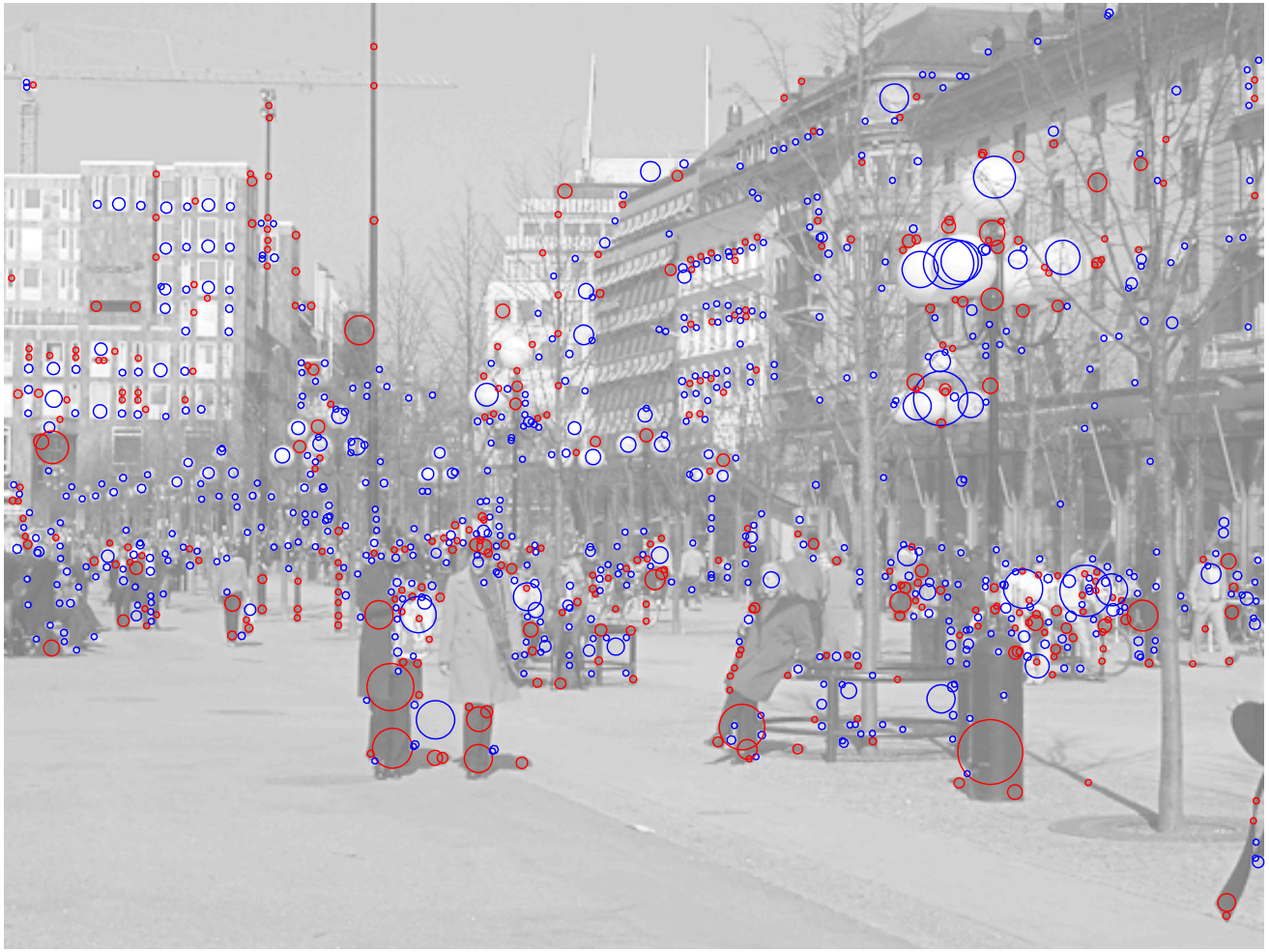


Figure from Lindeberg (2014) “Scale selection”, Computer Vision: A Reference Guide, pages 701-713, Springer.

# Determinant of the Hessian interest points

*1000 strongest scale-space extrema of the determinant of the Hessian* $\det \mathcal{H}_{norm} L$

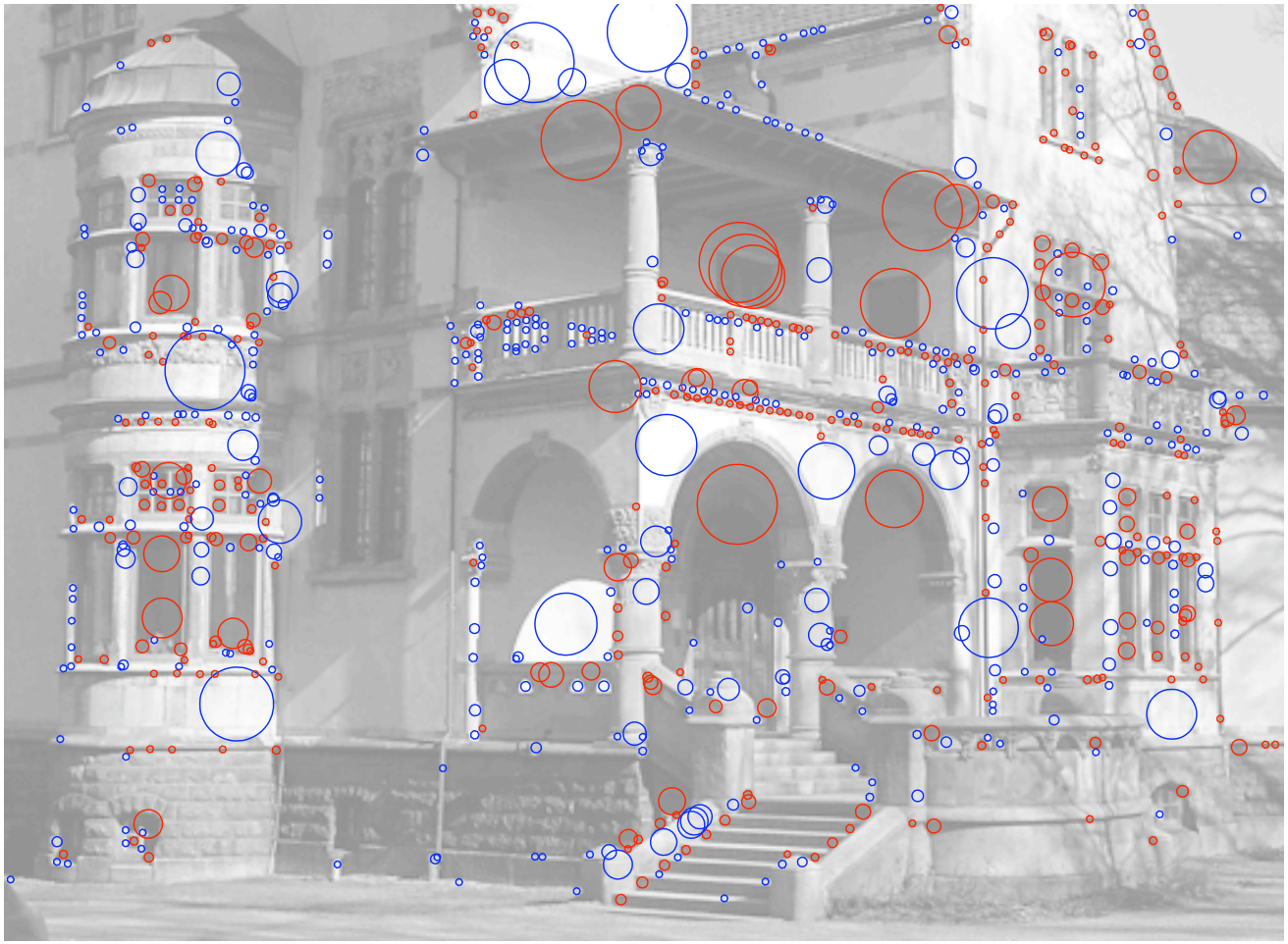
Figure from Lindeberg (2014) “Scale selection”, Computer Vision: A Reference Guide, pages 701-713, Springer.

# Interest point detection from scale-space extrema

*Original grey-level image*

# Scale-invariant Laplacian interest points

*800 strongest scale-space extrema of the Laplacian* $\nabla^2_{norm} L$



# Determinant of the Hessian interest points

*800 strongest scale-space extrema of the determinant of the Hessian* $\det \mathcal{H}_{norm} L$

# Algorithmic steps

Scale-space extrema detection algorithm:

1. Given a scale-range $[t_{min}, t_{max}]$ distribute a set of scale levels uniformly in terms of effective scale $\tau = \log t$
   (or preferably some better discrete approximation)
2. Convolve image by Gaussian kernel to each scale level
   (discrete analogue to Gaussian exists with better numerical properties)
3. For every image point, compute discrete approximation of the necessary derivatives and combine these into the desired differential invariant
4. Detect local extrema over scale and space by comparisons with the nearest neighbours in a $3 \times 3 \times 3$ neighbourhood complemented by threshold on the magnitude of the response
5. Optionally, sort the interest points in decreasing order with respect to their scale-normalized magnitude values

# Real-time implementation in pyramids

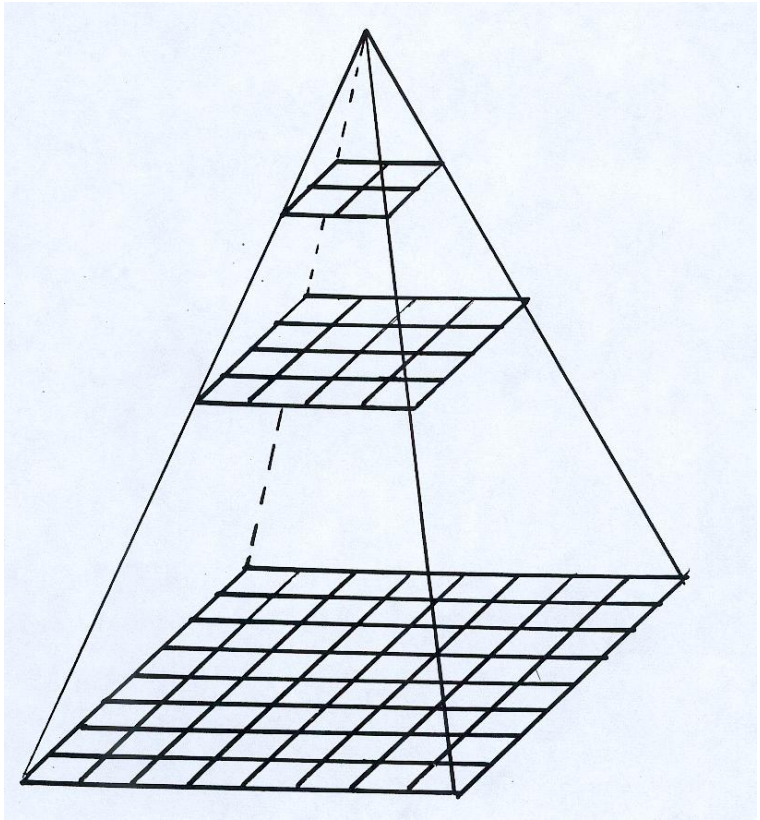
Motivations:

- A regular scale-space representation makes use of the same resolution (spatial sampling density) at all scales
- With increasing scales, smaller size image structures will on the other hand be suppressed

$\Rightarrow$ It should be possible to subsample coarser scale representations to improve the computational efficiency at coarser scales

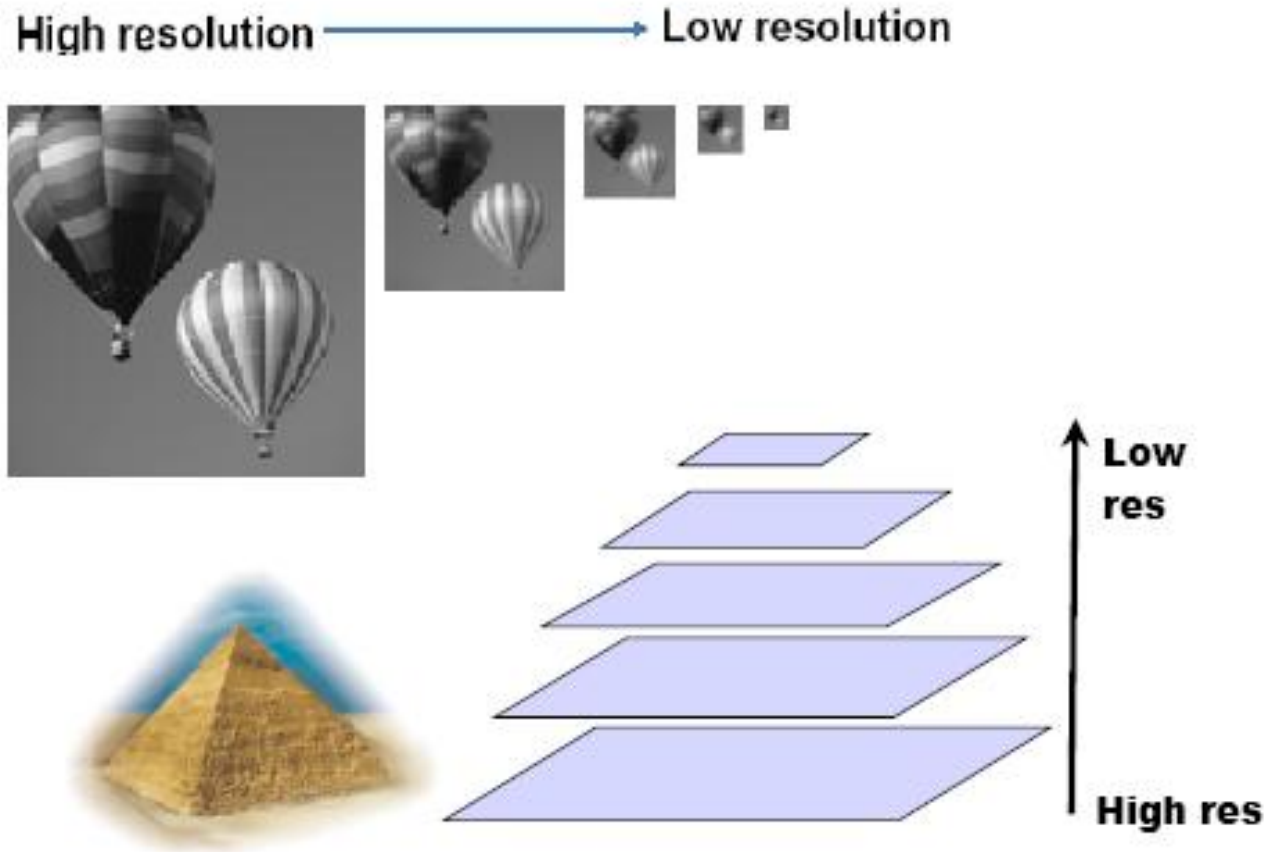
# Pyramid representation

Basic idea: Combine successive smoothing and subsampling.

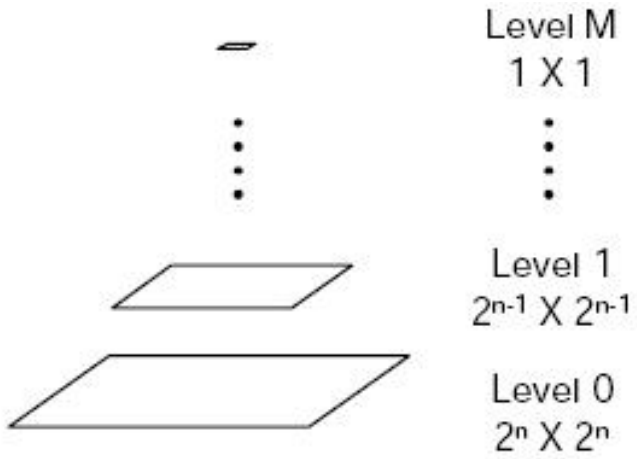


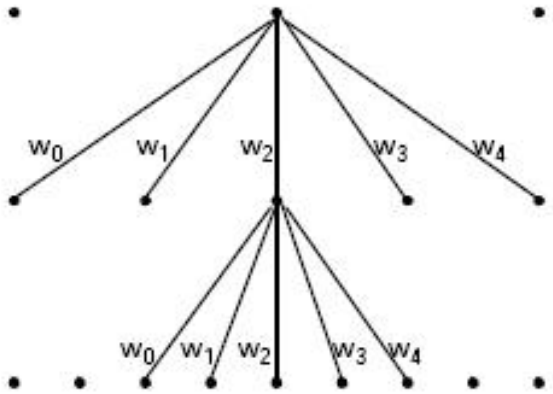
Usually: From $2^n \times 2^n$ image compute $2^{n-1} \times 2^{n-1}$ image.

# Pyramids



# Pyramid construction



# Pyramid construction

In the 1-D case (extended to 2-D by separable filtering) we have

$$L^{(k-1)} = \textsc{Reduce}(L^{(k)})$$
$$L^{(k-1)}(x) = \sum_{n=-\infty}^{\infty} c(n)\, L^{(k)}(2x - n)$$

where the filter coefficients should satisfy

- positivity: $c(n) \geq 0$
- symmetry: $c(-n) = c(n)$
- unimodality: $c(n) \geq c(n+1)$ (for $n \geq 0$)
- normalization: $\sum_{n=-\infty}^{\infty} c(n) = 1$
- equal contribution: $\sum_{n=-\infty}^{\infty} c(2n) = \sum_{n=-\infty}^{\infty} c(2n+1)$

# Relation to diffusion equation

Discretize the diffusion equation

$$\partial_t L = \tfrac{1}{2}\nabla^2 L = \{\textit{in 1-D}\} = \tfrac{1}{2} L_{xx}$$

using difference operators

$$\partial_t L^{(k)} = \frac{L^{(k+1)} - L^{(k)}}{\Delta t}$$

$$L_{xx}(x) = \frac{L(x-h) - 2L(x) + L(x+h)}{h^2} = \{h = 1\} = L(x-1) - 2L(x) + L(x+1)$$

gives

$$L^{(k+1)}(x) \approx L^{(k)}(x) + \Delta t\, \partial_t L^{(k)} = L^{(k)}(x) + \frac{\Delta t}{2} L_{xx}^{(k)}$$

One iteration corresponds to filtering with the discrete kernel

$$\left(\frac{\Delta t}{2}, 1 - \Delta t, \frac{\Delta t}{2}\right)$$

# Binomial kernels

For one iteration, the equal contribution condition implies $\Delta t = \frac{1}{2}$ and the kernel

$$\left(\tfrac{1}{4}, \tfrac{1}{2}, \tfrac{1}{4}\right) = \left(\tfrac{1}{2}, \tfrac{1}{2}\right)^2$$

For kernels of size 5 with coefficients $(c, b, a, b, c)$

- the normalization condition implies: $a + 2b + 2c = 1$
- the equal contribution condition implies: $a + 2c = 2b$

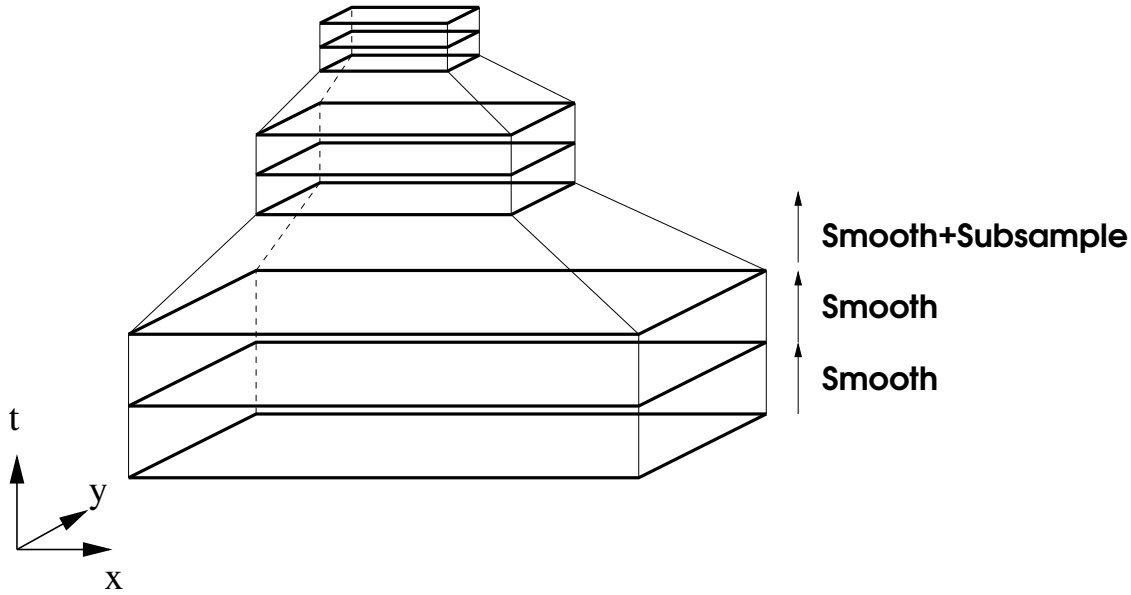
$$\Rightarrow \begin{cases} a \geq 1/4 \quad \text{(by unimodality } a \geq b\text{)} \\ b = 1/4 \\ c = 1/2 - a/4 \end{cases}$$

Empirically, Burt and Adelson (1983) suggested $a \approx 0.4$.
The value $a = 3/8 = 0.375$ corresponds to the binomial kernel

$$\left(\tfrac{1}{16}, \tfrac{4}{16}, \tfrac{6}{16}, \tfrac{4}{16}, \tfrac{1}{16}\right) = \left(\tfrac{1}{4}, \tfrac{2}{4}, \tfrac{1}{4}\right)^2$$

# Hybrid pyramids

Allow for several intermediate scale levels between each subsampling operation




Allow for different trade-offs between computational accuracy and computational efficiency

Figure from Lindeberg and Bretzner (2003) “Real-time scale selection in hybrid multi-scale representations”, Proc. Scale-Space Methods in Computer Vision, Springer LNCS 2695: 148–163.

# Hybrid pyramids

Degree of subsampling $h$ at any scale $\sigma = \sqrt{t}$ parameterized by *subsampling rate* $\rho$

$$h \leq \rho\, \sigma = \rho \sqrt{t}$$

In practice, $h$ is chosen as the maximum power of two that does not violate this condition

# Hybrid pyramids

| Pyramid type | $\rho$ | 500 blobs | | 1000 blobs | |
|---|---|---|---|---|---|
| | | det | det+loc | det | det+loc |
| BIN5PYRAMID | 1.73 | 16 | 32 | 17 | 45 |
| BIN5(2)PYRAMID | 1.22 | 23 | 51 | 25 | 79 |
| BIN5(3)PYRAMID | 1.00 | 39 | 66 | 43 | 97 |
| BIN5(4)PYRAMID | 0.87 | 55 | 89 | 63 | 127 |
| BIN5(5)PYRAMID | 0.77 | 72 | 105 | 81 | 153 |
| BIN5(6)PYRAMID | 0.71 | 88 | 121 | 101 | 173 |

*Computation time* (in ms) for blob detection in different hybrid pyramids with and without the additional post-processing stage for scale localization. The timings have been performed on a 2.4 GHz DELL PC with a Pentium 4 processor. (Anno 2002)

Table from Lindeberg and Bretzner (2003) “Real-time scale selection in hybrid multi-scale representations”, Proc. Scale-Space Methods in Computer Vision, Springer LNCS 2695: 148–163.

# Hybrid pyramids

| Pyramid type | $\rho$ | $\delta$ (pixels) | $r_{spread}$ |
|---|---|---|---|
| BIN5PYRAMID | 1.73 | 1.72 | 1.250 |
| BIN5(2)PYRAMID | 1.22 | 0.52 | 1.050 |
| BIN5(3)PYRAMID | 1.00 | 0.29 | 1.032 |
| BIN5(4)PYRAMID | 0.87 | 0.18 | 1.022 |
| BIN5(5)PYRAMID | 0.77 | 0.12 | 1.022 |
| BIN5(6)PYRAMID | 0.71 | 0.11 | 1.019 |

*Spatial and scale localization errors* for different subsampling factors $\rho$ using $l_p$-normalization. The experiments were performed on 1000 Gaussian blobs with random position and random variances between 10 and 100.

$\delta = E\left((\hat{x} - x_0)^2 + (\hat{y} - y_0)^2\right)$ = spatial error

$r_{spread} = \sqrt{2^{\sqrt{E\left(\log_2 \frac{\hat{t}}{t_0}\right)^2}}}$ = scale localization error in units of $\sigma$

$E()$ = expectation operator = averaging operator over the data

Table from Lindeberg and Bretzner (2003) “Real-time scale selection in hybrid multi-scale representations”, Proc. Scale-Space Methods in Computer Vision, Springer LNCS 2695: 148–163.

# Trade-offs in hybrid pyramids

14

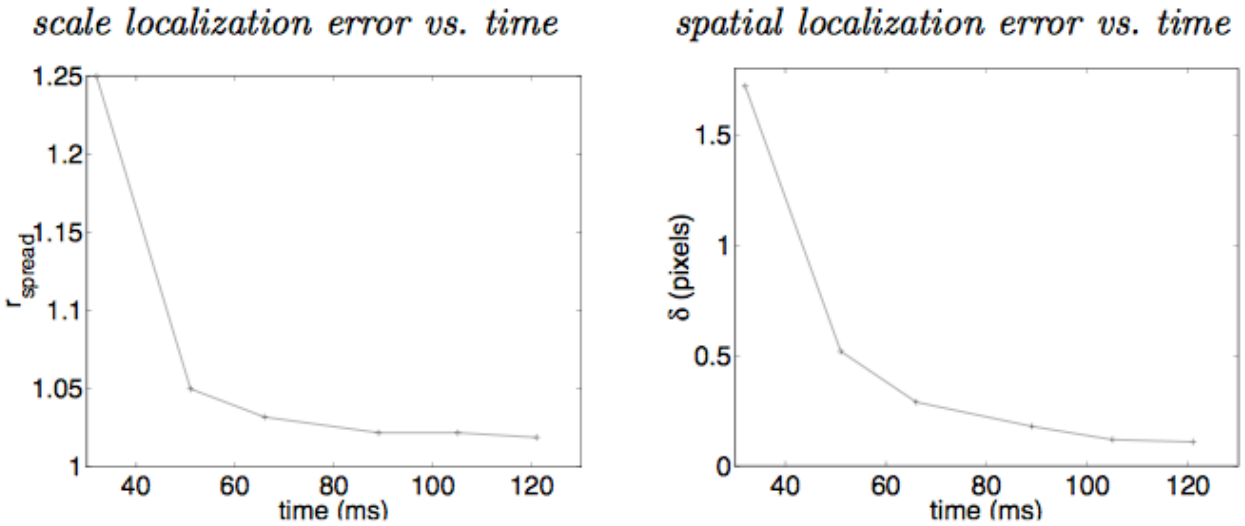
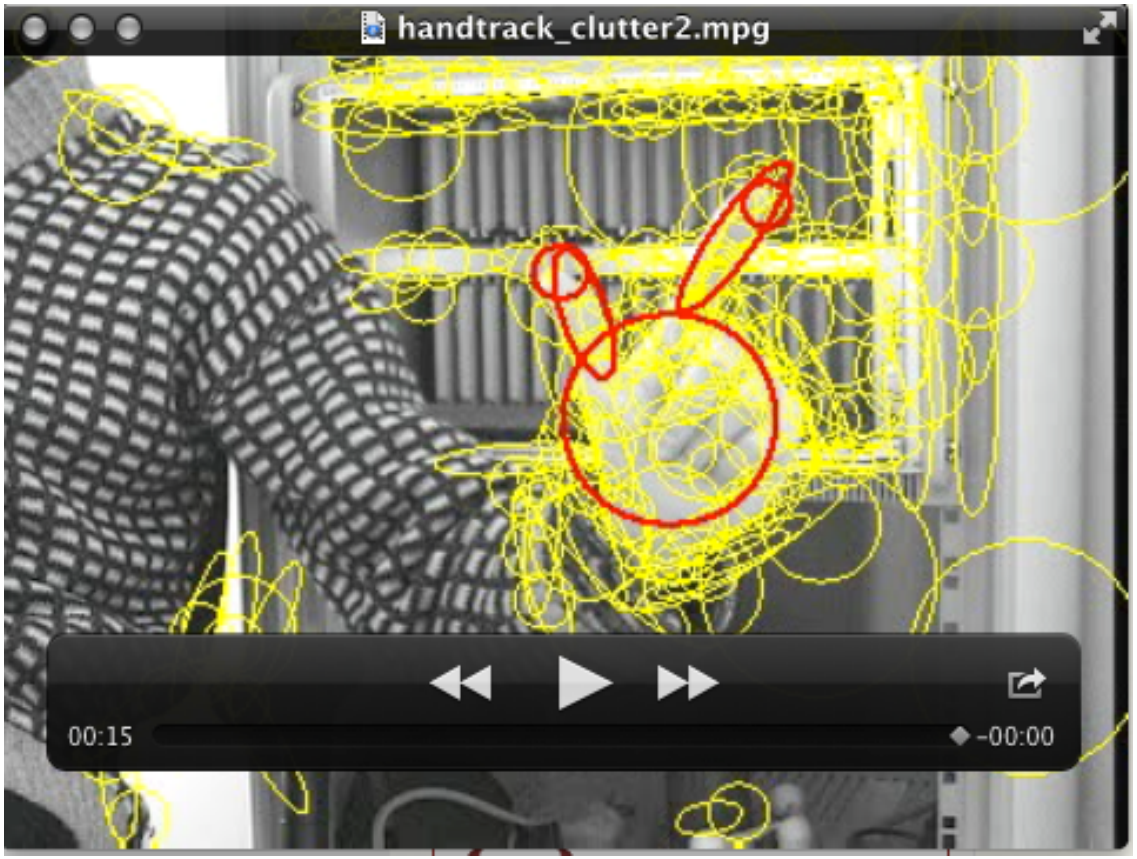



Fig. 4: Trade-offs between the localization error (vertical axis) and the computation time (horizontal axis) for hybrid pyramids with different values of $\rho$: (left) scale localization error, (right) spatial localization error.

Figure from Lindeberg and Bretzner (2003) “Real-time scale selection in hybrid multi-scale representations”,

Proc. Scale-Space Methods in Computer Vision, Springer LNCS 2695: 148–163.

# Application to tracking




Video from Bretzner, Laptev and Lindeberg (2002) “Hand-gesture recognition using multi-scale colour features, hierarchical features and particle filtering”, Proc. Face and Gesture’02, 63–74.

# Image-based matching and recognition

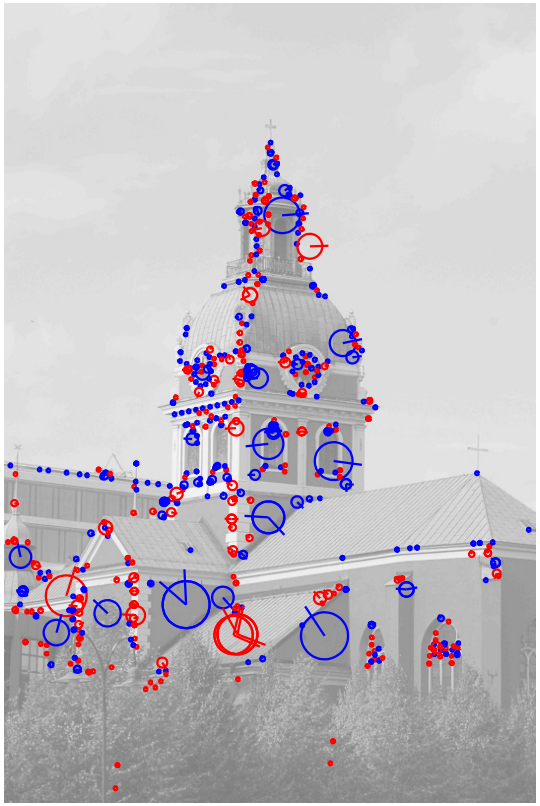
Common approach to image-based matching and recognition:

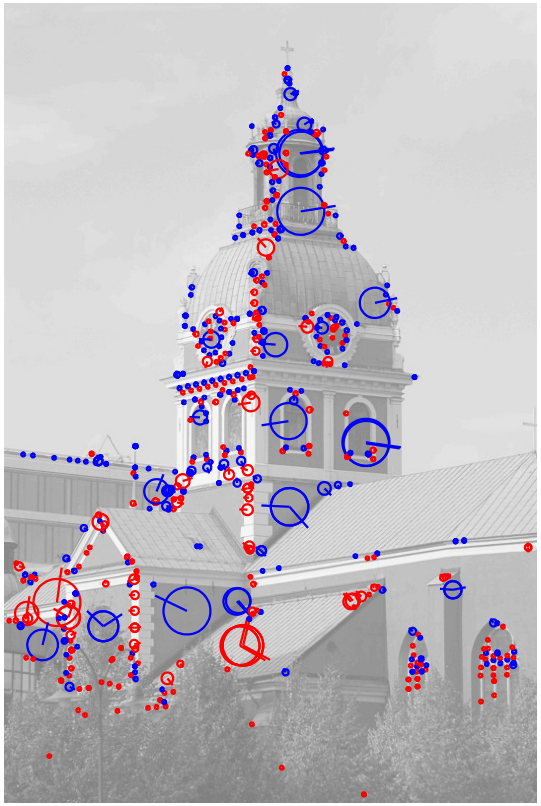
- Detect interest points.
- Compute image descriptors around the interest points.
- Match the image descriptors.

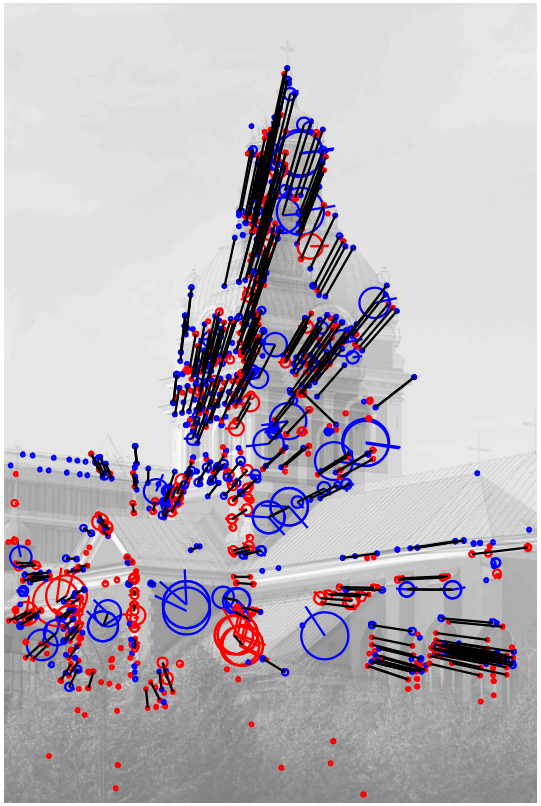
SIFT (Lowe 2004), SURF (Bay et al 2008) and related approaches.

Methodology in multi-view matching, object recognition,
3-D object and scene modelling, video tracking, gesture recognition,
panorama stitching, robot localization and matching.

# Image-based matching and recognition







Figures from Lindeberg (2012) “Scale-invariant feature transform”, Scholarpedia 7(5): 10491.

# The SIFT descriptor

- Compute gradients $\nabla L$ around $(x_0, y_0)$ at scale $\hat{t}$ of interest point
- Orientation normalization by computing histogram of local gradient directions over 36 bins and detecting peaks in the orientation histogram

  Strongest peak(s) $\Rightarrow$ orientation estimates
- Define $4 \times 4$ grid around interest point with spacing proportional to scale in orientation normalized image frame

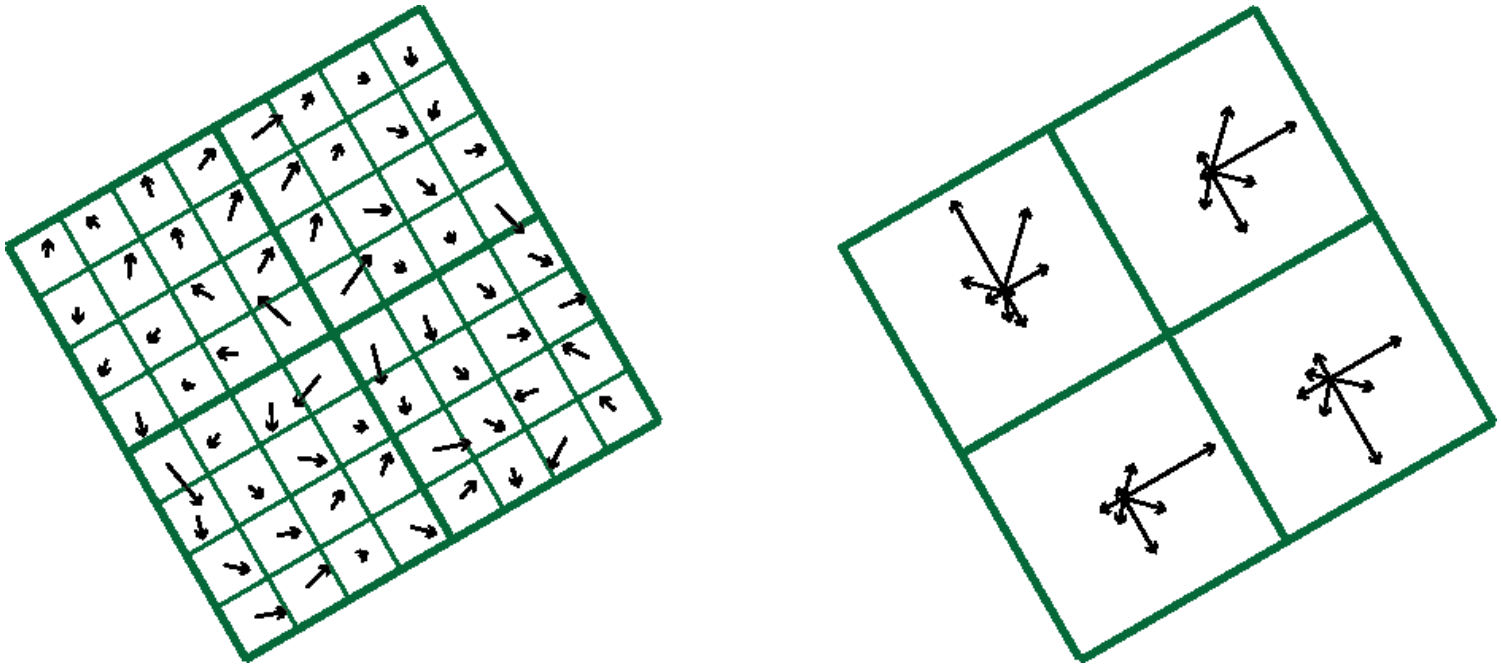
  Accumulate histogram of gradient directions quantized into 8 bins

  This histogram is computed in orientation normalized image frame

  Weighted accumulation using overlapping Gaussian window functions
- Trilinear interpolation for distributing weights between adjacent bins

# The SIFT descriptor

Here, schematically illustrated for a $2 \times 2$ instead of $4 \times 4$ grid:



$4 \times 4$ grid + 8 bins for gradient directions $\Rightarrow$ 128-D descriptor

Figure from Lindeberg (2012) “Scale-invariant feature transform”, Scholarpedia 7(5): 10491.

# Keypoint detection in original SIFT

- Detect scale-space extrema in difference-of-Gaussians pyramid
  Approximation of scale-space extrema of Laplacian in scale-space

$$\frac{1}{2}\nabla^2 L(x;\ t) = \partial_t L(x;\ t) \approx \frac{L(x;\ t+\Delta t) - L(x;\ t)}{\Delta t} = \frac{DOG(x;\ t, \Delta t)}{\Delta t}$$

  With self-similar scale sampling $\sigma_{i+1} = k\,\sigma_i$ we have $t_{i+1} = k^2\,t_i$

$$\Delta t\,\nabla^2 L = (k^2 - 1)\,t\,\nabla^2 L = (k^2 - 1)\,\nabla^2_{norm} L$$

  which implies

$$DOG(x, y; t) \approx \frac{(k^2 - 1)}{2}\,\nabla^2_{norm} L(x, y; t)$$

- Supress responses along edges

$$\frac{\det \mathcal{H} L}{\mathrm{trace}^2\,\mathcal{H} L} = \frac{L_{xx} L_{yy} - L_{xy}^2}{(L_{xx} + L_{yy})^2} \geq \frac{r}{(r+1)^2}$$

  where $r \geq 1$ denotes an upper limit on the permitted ratio between the larger and the smaller eigenvalues of the Hessian matrix

# Lowe's keypoint detection in SIFT

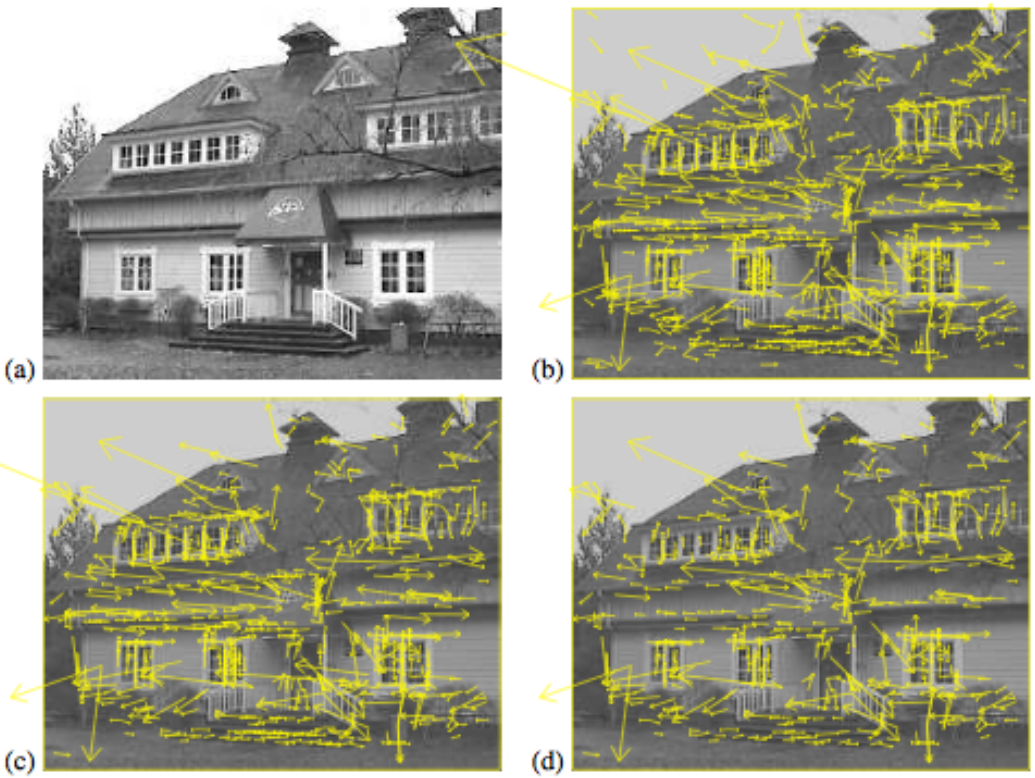



Figure 5: This figure shows the stages of keypoint selection. (a) The 233x189 pixel original image. (b) The initial 832 keypoints locations at maxima and minima of the difference-of-Gaussian function. Keypoints are displayed as vectors indicating scale, orientation, and location. (c) After applying a threshold on minimum contrast, 729 keypoints remain. (d) The final 536 keypoints that remain following an additional threshold on ratio of principal curvatures.

Figure from Lowe (2004) "Distinctive image features from scale-invariant keypoints", Int. J. of Computer Vision 60(2): 91–110.

# Recognition based on SIFT descriptors

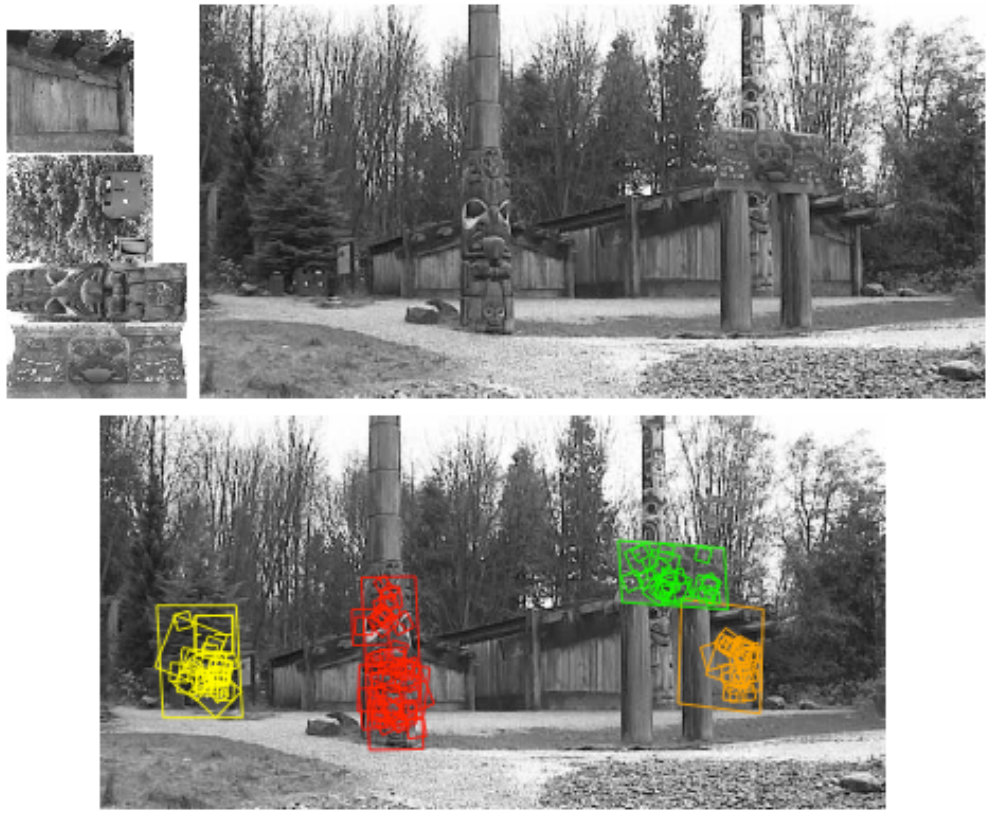


Figure 13: This example shows location recognition within a complex scene. The training images for locations are shown at the upper left and the 640x315 pixel test image taken from a different viewpoint is on the upper right. The recognized regions are shown on the lower image, with keypoints shown as squares and an outer parallelogram showing the boundaries of the training images under the affine transform used for recognition.

Figure from Lowe (2004) “Distinctive image features from scale-invariant keypoints”, Int. J. of Computer Vision 60(2): 91–110.

# The SURF descriptor

- Compute partial derivatives $L_x$ and $L_y$ around $(x_0, y_0)$ at scale $\hat{t}$ of interest point
- Orientation normalization similar to SIFT
- Sums of derivatives $\sum L_x$, $\sum |L_x|$, $\sum L_y$, $\sum |L_y|$ over $4 \times 4$ subwindows around the interest point

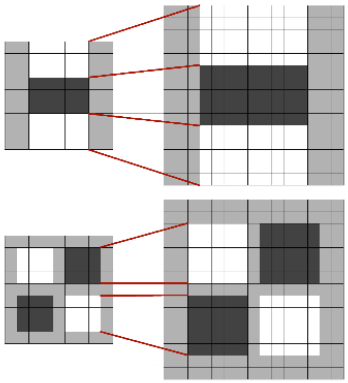
  "can be computed faster than SIFT"

# Bay et al's implementation in SURF

- Interest point detection by determinant of Hessian

$$\det \mathcal{H}_{norm} L = L_{xx} L_{yy} - L_{xy}^2$$

- Derivatives approximated by Haar wavelets (box filters)



to allow for fast computations by *integral images*

(however at the cost of sacrificing rotational invariance, ringing phenomena and scale-space properties over scale)

Figures from Bay, Ess, Tuytelaars and van Gool (2008) "Speeded-up robust features (SURF)", Computer Vision and Image Understanding 110(3): 346–359.

# Scene reconstruction based on SURF descriptors

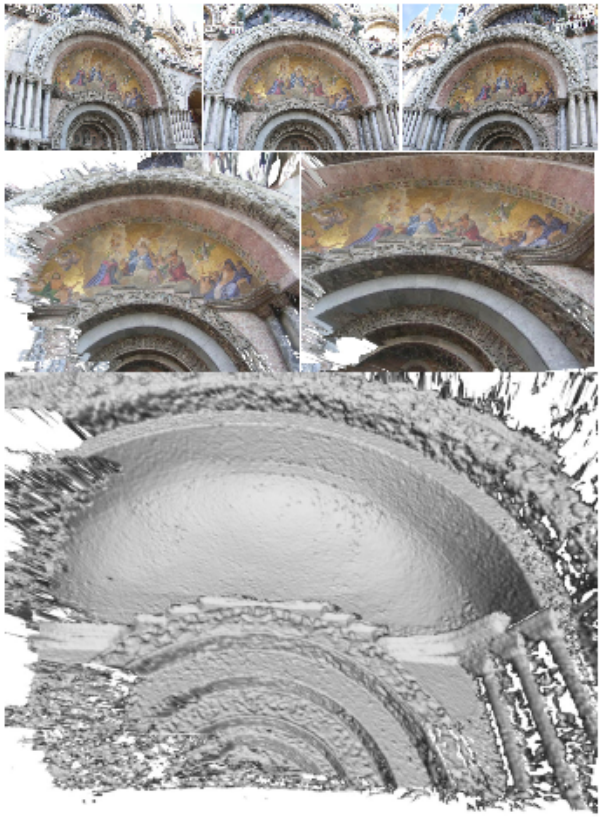


Fig. 23. 3D reconstruction with KU-Leuven's 3D webservice. Top row: The 3 input images of a detail of the San Marco Cathedral in Venice. Middle row: Samples of the textured dense reconstruction. Bottom row: un-textured dense reconstruction. The quality of the dense 3D model directly reflects the quality of the camera calibration. The images were taken by Maurizio Forte, CNR-ITABC, Rome).

Figure from Bay, Ess, Tuytelaars and van Gool (2008) "Speeded-up robust features (SURF)", Computer Vision and Image Understanding 110(3): 346–359.

# Comparison of features on poster image dataset

High resolution photos $4900 \times 3200$ pixels taken of 12 indoor and outdoor scenes in natural city/office environments, from which posters of size $100 \times 70$ cm were produced by professional laboratory.

*"representative selection of natural image structures"*

Each poster was photographed from 14 positions:

- 11 normal views with relative scale factors $s \approx 1.25$, 1.5, 1.75, 2.0, 2.5, 3.0, 3.5, 4.0, 5.0 and 6.0.
- 3 additional oblique views with slant angles $\approx 22.5°$, $30°$ and $45°$ relative to the frontal view with $s \approx 2.0$.

*"natural image transformations corresponding to variations in viewing distance and viewing direction"*

Planar posters $\Rightarrow$ Ground truth can be defined by homographies/calibration.

# Poster image dataset

*Distance variations*

$s \approx 1.25$

$s \approx 6.0$

*Viewing variations*

$\varphi \approx 0^\circ$

$\varphi \approx 45^\circ$

Figures from Lindeberg (2015) “Image matching using generalized scale-space interest points”, Journal of Mathematical Imaging and Vision 52(1): 3-36.

# Matching of image descriptors

- Given two images $f_A$ and $f_B$, compute sets of interest points $A = \{A_i\}$ and $B = \{B_j\}$ from each image
- Compare interest points in the two domains by computing the Euclidean difference between their image descriptors

  (with SIFT or SURF defined from Gaussian derivatives instead of pyramid or Haar wavelets: “Gauss-SIFT” and “Gauss-SURF”)
- Accept match between pair of interest points $(A_i, B_j)$ only if:
  - $A_i$ is the best match for $B_j$ in relation to all the other, points in $A$
  - $B_j$ is the best match for $A_i$ in relation to all the other points in $B$
- To suppress possibly ambigous matches, in addition require the ratio between the distances to the nearest and the next nearest image descriptor to be less than $r = 0.9$

# Experimental protocol

For each pair of images within domain of scale or viewing variations:

- Compute interest points from each image separately.
- Transform interests points to other domain using homography, and scale of interest points by local scale factor of homography.
- Represent each interest point by circle of size propertional to scale.
- Accept match if ratio between intersection and union of circles above threshold:

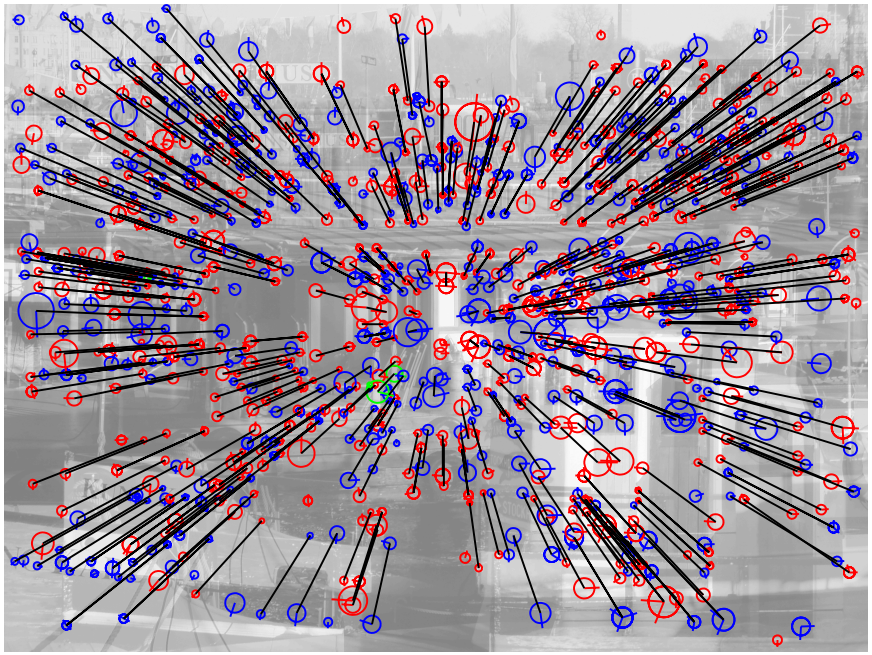
$$m(C_A, C_B) = \frac{|\bigcap(C_A, C_B)|}{|\bigcup(C_A, C_B)|} \geq m_0$$

- Measure performance of interest point detector by:

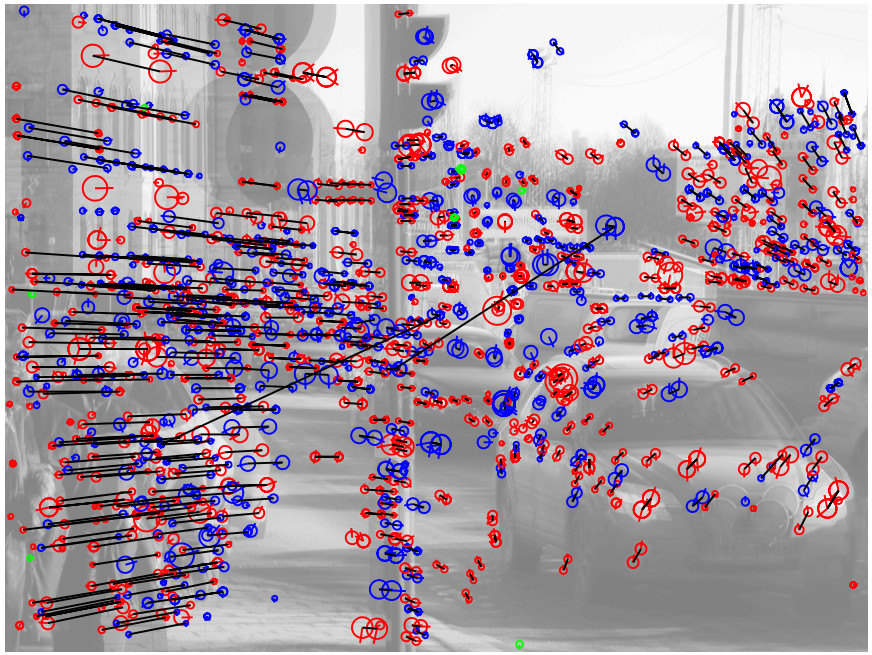
$$\text{efficiency} = \frac{\#(\text{interest points that lead to accepted matches})}{\#(\text{interest points})}$$

# Examples of matching results

*Scaling transformation*



*Foreshortening transformation*



(here improved interest point detector with Gauss-SIFT descriptors)

Figures from Lindeberg (2015) “Image matching using generalized scale-space interest points”,

Journal of Mathematical Imaging and Vision 52(1): 3-36.

# Experimental results

*Efficiency: SIFT-like image descriptor*

| | scaling | | foreshortening | | average | |
|---|---|---|---|---|---|---|
| Interest points | SIFT | SURF | SIFT | SURF | SIFT | SURF |
| $\nabla^2_{norm} L$ | 0.7484 | 0.7424 | 0.7512 | 0.7280 | 0.7498 | 0.7352 |
| $\det \mathcal{H}_{norm} L$ | 0.7721 | 0.7656 | 0.7635 | 0.7402 | 0.7678 | 0.7529 |
| Harris-Laplace | 0.7002 | 0.6948 | 0.7046 | 0.6724 | 0.7024 | 0.6836 |

Experimental results from Lindeberg (2015) “Image matching using generalized scale-space interest points”, Journal of Mathematical Imaging and Vision 52(1): 3-36.

# Relative ranking of interest point detectors

Interest points and image descriptors ranked on matching efficiency:

| *Interest points* | *Descriptor* | *Efficiency* |
|---|---|---|
| $\det \mathcal{H}_{norm} L$ | *SIFT* | *0.7678* |
| $\det \mathcal{H}_{norm} L$ | *SURF* | *0.7529* |
| $\nabla^2_{norm} L$ | *SIFT* | *0.7498* |
| $\nabla^2_{norm} L$ | *SURF* | *0.7352* |
| *Harris-Laplace* | *SIFT* | *0.7024* |
| *Harris-Laplace* | *SURF* | *0.6836* |

Conclusions (both image features and image descriptors from Gaussian derivatives):

- $\det \mathcal{H}_{norm} L$ is a better interest point detector than $\nabla^2_{norm} L$

  (it can also be theoretically shown that $\det \mathcal{H}_{norm} L$ has better properties under affine image deformations than $\nabla^2_{norm} L$)
- SIFT is a better image descriptor than SURF

  (SURF can on the other hand be computed more efficiently)
- both $\det \mathcal{H}_{norm} L$ and $\nabla^2_{norm} L$ are much better interest point detectors than Harris-Laplace

# Extensions of the SIFT descriptor: RootSIFT

Idea: Change the metric for comparing the SIFT histogram descriptors from Euclidean distance to Hellinger distance.

Can be implemented in practice in the following way:

1. $l_1$-normalize each image descriptor.
2. Take the element-wise square root.
3. $l_2$-normalize the result.
4. Compare the modified image descriptors by Euclidean distance.

Usually improves the performance of SIFT matching significantly. (Examples: “before 0.672, after 0.720”, “before 0.726, after 0.756”.)

Arandjelović and Zisserman (2012) “Three things everyone should know to improve object retrieval”, *Proc. Computer Vision and Pattern Recognition (CVPR 2012)*, pages 2911–2918.

# Further extensions of SIFT-based matching

Ideas:

- Perform further quantization from 128-D SIFT to 48-bytes PSIFT or 61-bytes BiSIFT reduces the memory footprint substantially.
- Hierarchical coarse-to-fine descriptor to speed up the descriptor matching process if the distance computations are heavy.
- Symmetric nearest-neighbour-ratio measure performs better than one-sided nearest-neighbour-ratio measure.

Conclusions:

- Deep descriptors achieve the best matching accuracy and can be computationally efficient on GPUs.
- SIFT-like descriptors, especially if properly quantized and matched, remain competitive still today in terms of balance between accuracy, storage, efficiency and hardware-software flexibility.

Bellavia and Colombo (2020) “Is there anything new to say about SIFT matching?”, *International Journal of Computer Vision* 128(7): 1847–1866.

# Learning-based interest points and image descrip

Several deep-learning-based approaches to interest point detection and image descriptors have been developed, for example

- LIFT (Yi et al 2016)
- L2-Net (Tian et al 2017)
- SuperPoint (DeTone et al 2018)
- Key.Net (Barroso-Laguna et al 2019)
- R2D2 (Revaud et al 2019)
- D2-Net (Dusmanu et al 2019)

Learning-based approaches in general report higher performance values compared to classical image descriptors, such as SIFT or SURF.

Problems have, however, been reported that training of interest point detectors on one dataset may not give good generalization properties to other datasets. Generalization of image descriptors appears to be more robust.

# Analogy to scale selection in deep learning

Deep networks are trained to detect hypotheses about bounding boxes, in which deep learning-based object detection is then performed:

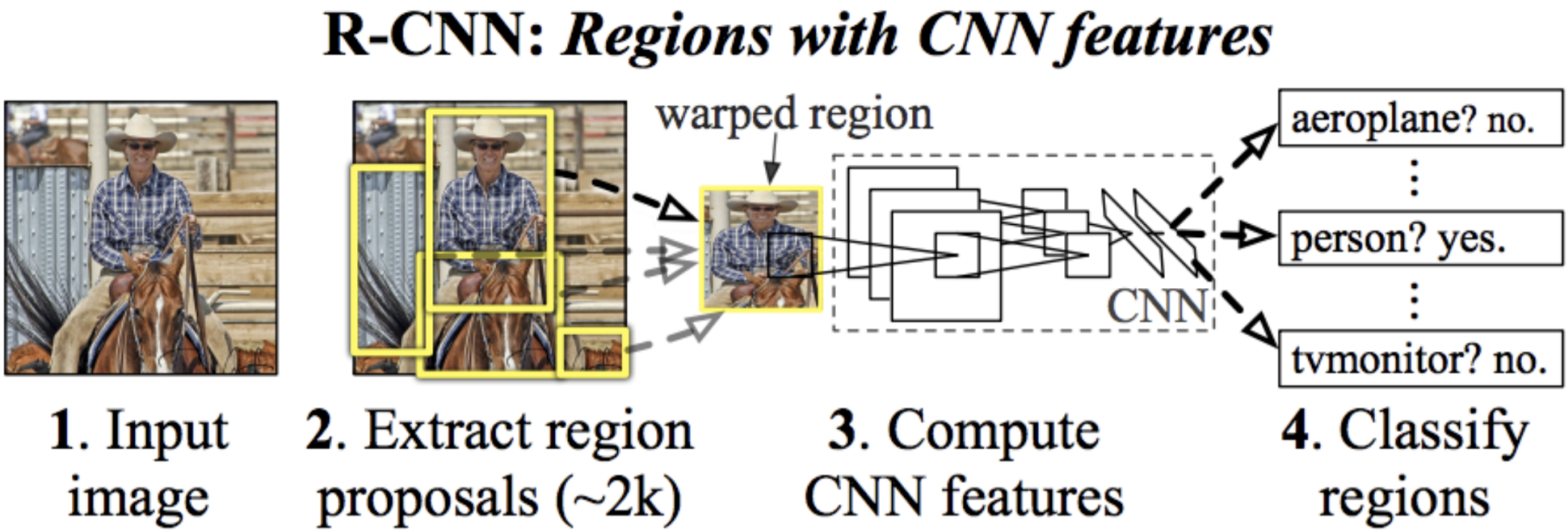
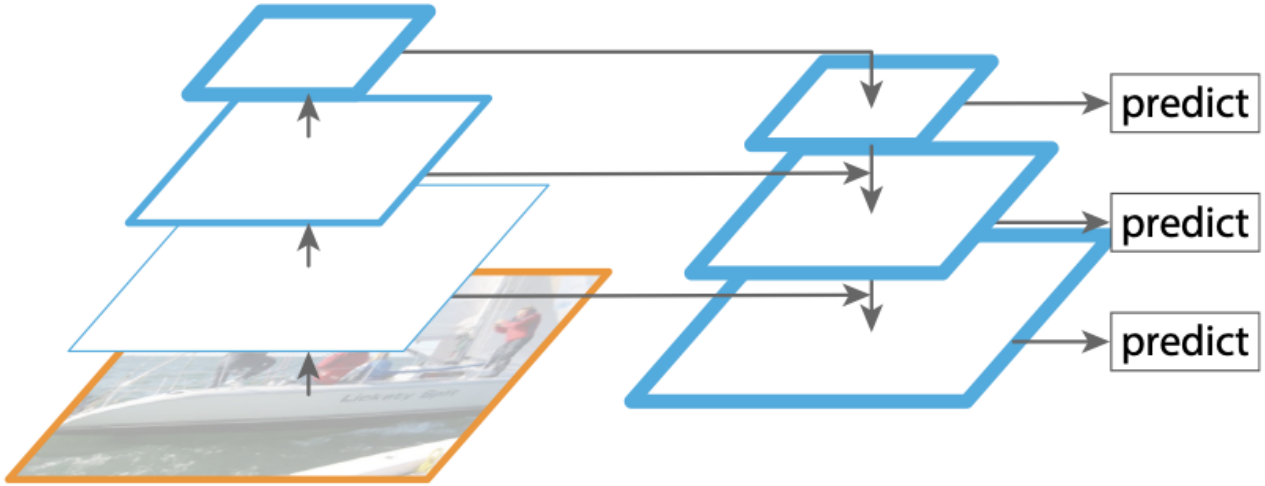



Figure from Girshick, Donahue, Darrell and Malik (2014) “Rich feature hierarchies for accurate object detection and semantic segmentation”, Proc. Computer Vision and Pattern Recognition (CVPR 2014), pages 580–587.

# Pyramids in deep learning

Several deep networks are formulated in terms of pyramids.




Deep networks are also formulated with parallel pathways between adjacent layers, with processing at multiple levels of resolution or scale.

Figure from Lin, Dollar, Girshick, He, Hariharan and Belongie (2017) “Feature pyramid networks for object detection”, Proc. Computer Vision and Pattern Recognition (CVPR 2017), pages 2117-2125.

# Summary of good questions I

- What are the motivations for computing interest points at an early stage in a vision system? What are interest points typically used for?
- Describe three common interest point detectors including their mathematical definitions and underlying motivations.
- Why is scale selection an important operation? Describe how scale selection can be performed in practice.
- What is meant by scale-space extrema? Describe an algorithm for computing scale-space extrema from a 2-D image.
- Show how scale selection applied to a Gaussian blob selects the scale of the blob.

# Summary of good questions II

- What is the motivation for using image pyramids in computer vision?
- How are image pyramids computed from image data? Describe basic conditions for choosing the filter coeffients in a pyramid scheme.
- Describe how pyramid weights can be related to the diffusion equation.
- Describe a basic trade-off issue that arises in hybrid pyramids.
- What is the purpose of computing image descriptors at interest points?
- How is the SIFT descriptor defined from image data?
- How is the SURF descriptor defined from image data?
- Outline the basic steps in an algorithm that matches interest points with associated image descriptors between two images of the same scene.

# Literature for further reading


- Lindeberg (2009) “Scale-space”, *Encyclopedia of Computer Science and Engineering*, John Wiley and Sons, IV: 2495–2504,
  http://dx.doi.org/10.1002/9780470050118.ecse609
  Preprint available from
  http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-40202
- Lindeberg (2012) “Scale-invariant feature transform”, *Scholarpedia* 7(5): 10491.
  http://dx.doi.org/10.4249/scholarpedia.10491
- Lindeberg (2014) “Scale selection”. In: K. Ikeuchi (ed.) *Computer Vision: A Reference Guide*, pages 701-713.
  http://dx.doi.org/10.1007/978-0-387-31439-6_242
  Preprint available from
  http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-267559

# Research papers on this topic


- Harris and Stephens (1988) “A combined corner and edge detector”, *Proc. 4th Alvey Vision Conference*, 147–151.
- Lindeberg (1998) “Feature detection with automatic scale selection”, *International Journal of Computer Vision* 30(2): 77–116.
- Lindeberg and Bretzner (2003) “Real-time scale selection in hybrid multi-scale representations”, *Proc. Scale-Space’03*, Springer LNCS 2695: 148–163.
- Lowe (2004) “Distinctive image features from scale-invariant keypoints”, *International Journal of Computer Vision* 60(2): 91–110.
- Bay, Ess, Tuytelaars and van Gool (2008) “Speeded-up robust features (SURF)”, *Computer Vision and Image Understanding* 110(3): 346–359.
- Lindeberg (2015) “Image matching using generalized scale-space interest points”, *Journal of Mathematical Imaging and Vision* 52(1): 3–36.
- Bellavia and Colombo (2020) “Is there anything new to say about SIFT matching?”, *International Journal of Computer Vision* 128(7): 1847–1866.

# Selected deep learning approaches

- Yi, Trulls, Lepetit and Fua (2016) “LIFT: Learned invariant feature transform”, *Proc. European Conference on Computer Vision (ECCV 2016)*, Springer LNCS 9910: 467–483.
- Tian, Fan and Wu (2017) “L2-Net: Deep learning of discriminative patch descriptor in Euclidean space”, *Proc. Computer Vision and Pattern Recognition (CVPR 2017)*, pages 661–669.
- DeTone, Malisiewicz and Rabinovich (2018) “SuperPoint: Self-supervised interest point detection and description”, *Proc. Computer Vision and Pattern Recognition Workshops*, pages 224–236.
- Barroso-Laguna, Riba, Ponsa and Mikolajczyk (2019) “Key.Net: Keypoint detection by handcrafted and learned CNN filters”, *Proc. International Conference on Computer Vision (ICCV 2019)*, pages 5836–5844.
- Revaud, Weinzaepfel, Souza, Pion, Csurka, Cabon and Humenberger (2019) “R2D2: Repeatable and reliable detector and descriptor”, *Proc. Neural Information Processing Systems (NeurIPS 2019)*.
- Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii and Sattler (2019) “D2-Net: A trainable CNN for joint description and detection of local features”, *Proc. Computer Vision and Pattern Recognition (CVPR 2019)*, pages 8092–8101.