Scale-space image processing

- Corresponding image features can appear at different scales

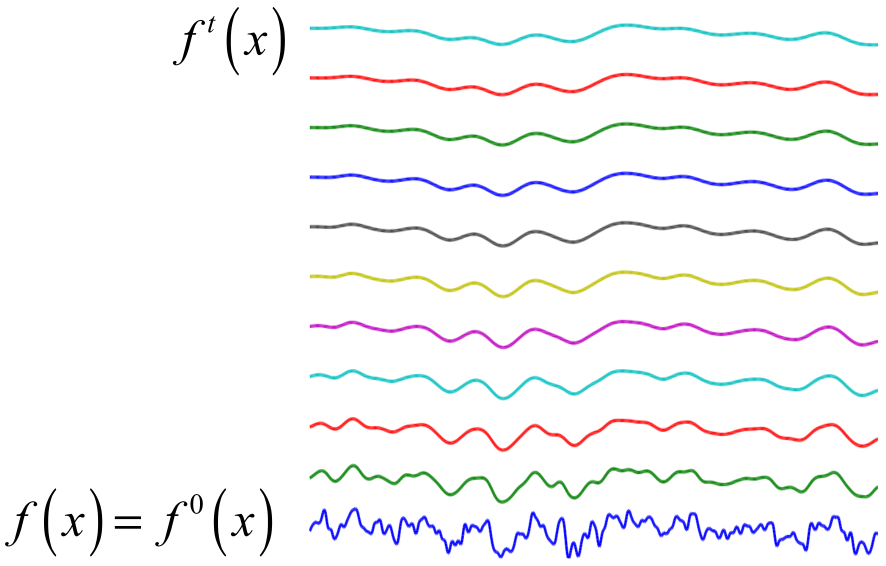
- Like shift-invariance, ***scale-invariance*** of image processing algorithms is often desirable.
- Scale-space representation is useful to process an image in a manner that is both shift-invariant and scale-invariant

Scale-space image processing

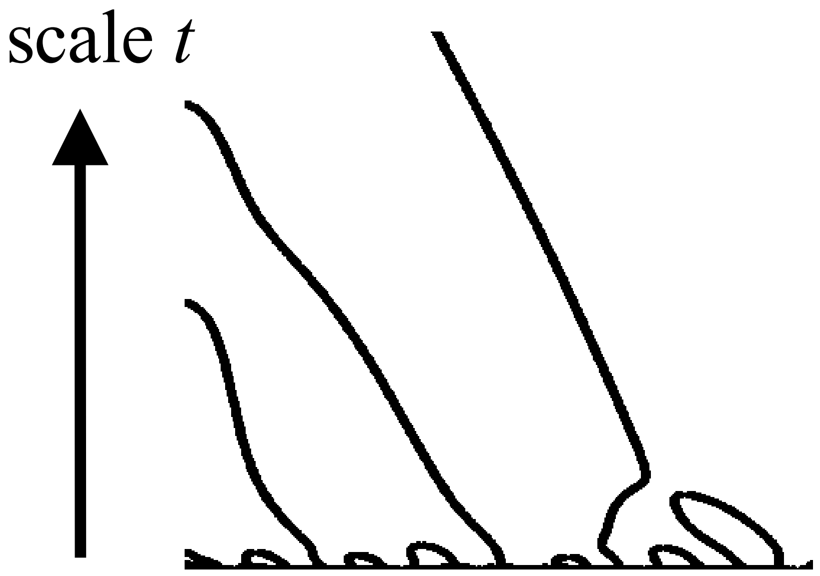
- Scale-space theory
- Laplacian of Gaussian (LoG) and Difference of Gaussian (DoG)
- Scale-space edge detection
- Scale-space keypoint detection
  - Harris-Laplacian
  - SIFT detector
  - SURF detector

Scale-space representation of a signal

Parametric family of signals $f^t(x)$ where fine-scale information is successively attenuated



Successive smoothing
with a Gaussian filter



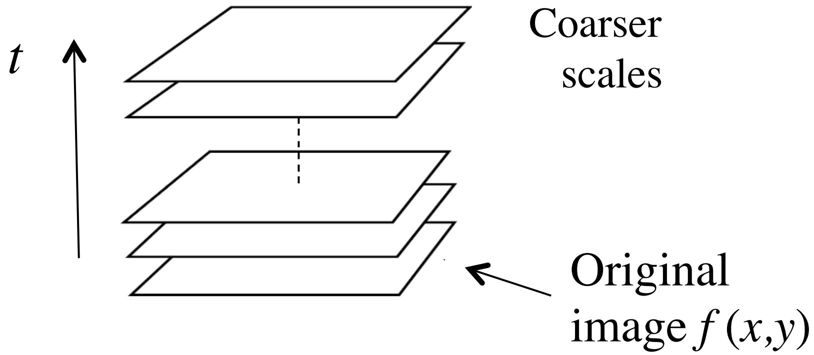
Zero-crossings of 2nd derivative $f^{t''}(x)$
Fewer edges at coarser scales

Scale-space representation of images

- Parametric family of images smoothed by Gaussian filter

$$f^t(x,y) = g^t(x,y) * f(x,y);\ \mathrm{t} \geq 0 \quad \text{with} \quad g^t(x,y) = \frac{1}{2\pi t}\exp\left(-\frac{x^2+y^2}{2t}\right)$$

$$F^t(\omega_x,\omega_y) = G^t(\omega_x,\omega_y)F(\omega_x,\omega_y) \quad \text{with} \quad G^t(\omega_x,\omega_y) = \exp\left(-\frac{t}{2}\left(\omega_x^2+\omega_y^2\right)\right)$$



- Shift-invariance

$$f^t(x-\Delta x, y-\Delta y) = g^t(x,y) * f(x-\Delta x, y-\Delta y)$$

- Rotation-invariance

$$f^t(x\cos\theta - y\sin\theta, x\sin\theta + y\cos\theta) = g^t(x,y) * f(x\cos\theta - y\sin\theta, x\sin\theta + y\cos\theta)$$

Scale-space representation of images (cont.)

- Commutative semigroup property

$$\begin{aligned} f^{t1+t2}(x,y) &= g^{t1}(x,y) * f^{t2}(x,y) = g^{t2}(x,y) * f^{t1}(x,y) \\ &= g^{t1}(x,y) * g^{t2}(x,y) * f(x,y) \end{aligned}$$

- Separability

$$g^{t}(x,y) = \frac{1}{2\pi t}\exp\left(-\frac{x^2+y^2}{2t}\right) = \frac{1}{\sqrt{2\pi t}}\exp\left(-\frac{x^2}{2t}\right)\cdot\frac{1}{\sqrt{2\pi t}}\exp\left(-\frac{y^2}{2t}\right)$$

$$G^{t}(\omega_x,\omega_y) = \exp\left(-\frac{t}{2}\left(\omega_x^2+\omega_y^2\right)\right) = \exp\left(-\frac{t}{2}\omega_x^2\right)\exp\left(-\frac{t}{2}\omega_y^2\right)$$

Scale-space representation of images (cont.)

- Non-creation of local extrema (for $f(x,y)$ and all of its partial derivatives) since $g^t(x,y) \geq 0$ and unimodal.
- Solution to diffusion equation (heat equation)

$$\boxed{\frac{\partial}{\partial t} f^t(x,y) = \frac{1}{2} \nabla^2 f^t(x,y)}$$

$$\begin{aligned}
\frac{\partial}{\partial t} F^t(\omega_x,\omega_y) &= \frac{\partial}{\partial t} G^t(\omega_x,\omega_y) F(\omega_x,\omega_y) \\
&= \frac{\partial}{\partial t} \exp\left(-\frac{t}{2}\left(\omega_x^2+\omega_y^2\right)\right) F(\omega_x,\omega_y) \\
&= -\frac{1}{2}\left(\omega_x^2+\omega_y^2\right) \exp\left(-\frac{t}{2}\left(\omega_x^2+\omega_y^2\right)\right) F(\omega_x,\omega_y) \\
&= -\frac{1}{2}\left(\omega_x^2+\omega_y^2\right) F^t(\omega_x,\omega_y)
\end{aligned}$$

$$\frac{\partial}{\partial t} f^t(x,y) = \frac{1}{2}\nabla^2 f^t(x,y)$$

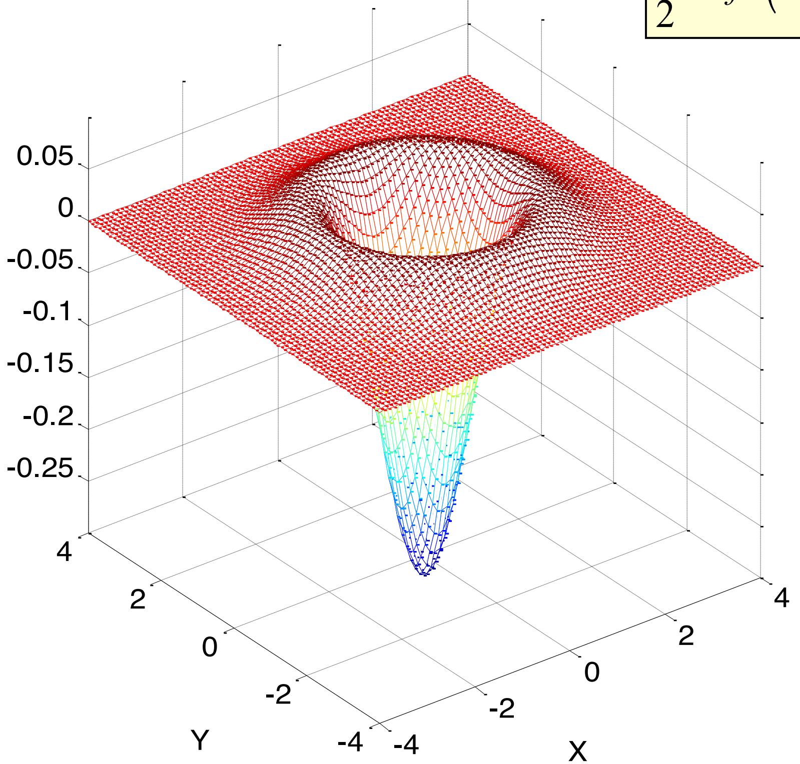
t = 0.07 sec

LoG vs. DoG

Laplacian of Gaussian

$t = \sigma^2 = 1$

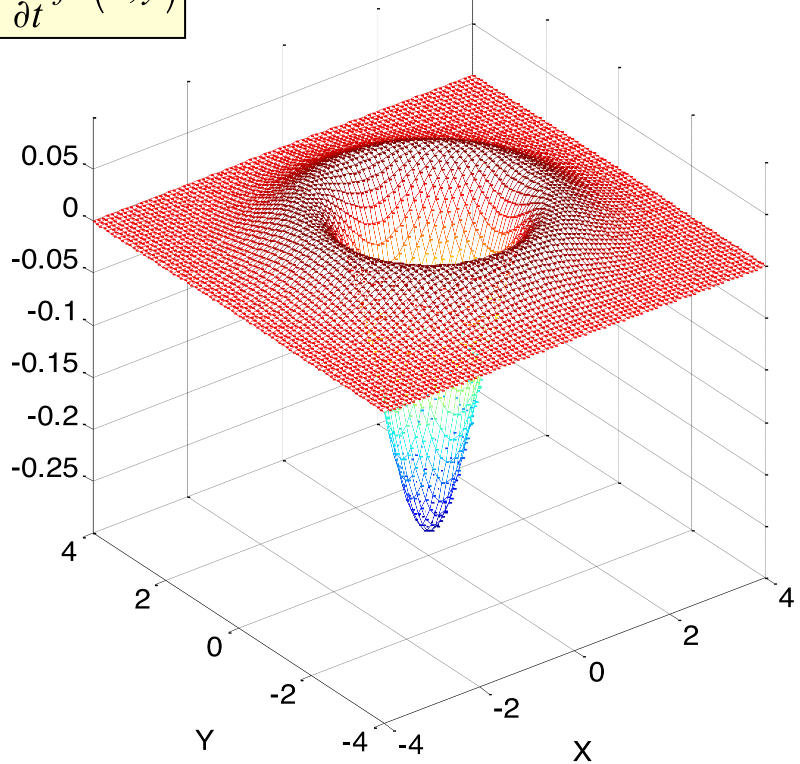
$$\frac{1}{2}\nabla^2 f^t(x,y) = \frac{\partial}{\partial t} f^t(x,y)$$



$$LoG(x,y) = -\frac{1}{\pi t^2}\left(1 - \frac{x^2+y^2}{2t}\right)e^{-\frac{x^2+y^2}{2t}}$$

Difference of Gaussians

$t = \sigma^2 = 1, \; k = 1.1$

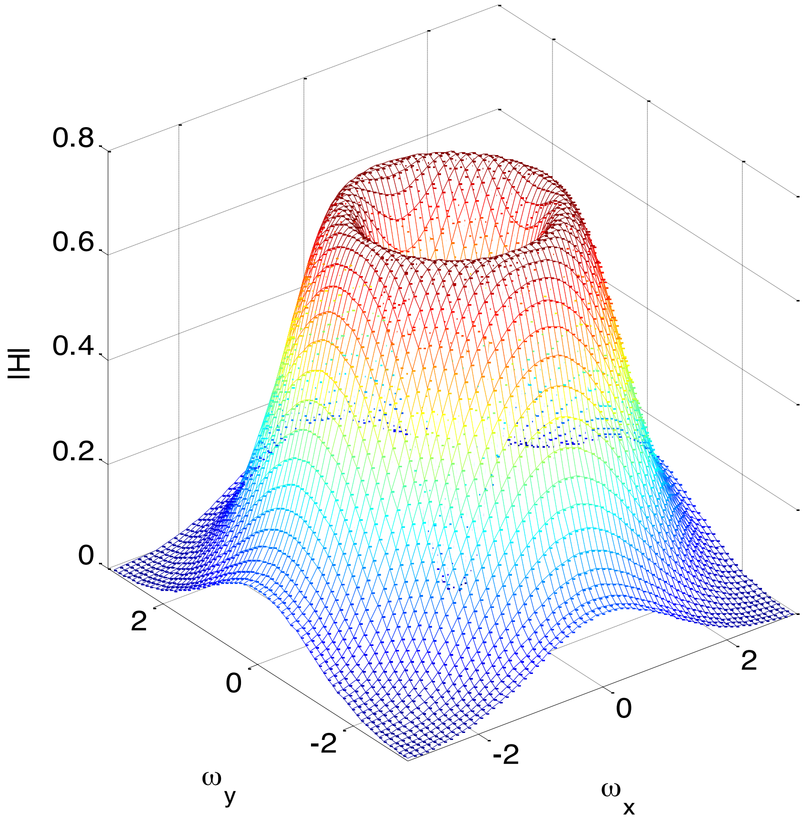


$$DoG(x,y) = \frac{1}{(k-1)t}\left(g^{k^2 t}(x,y) - g^t(x,y)\right)$$

LoG vs. DoG (cont.)

Laplacian of Gaussian

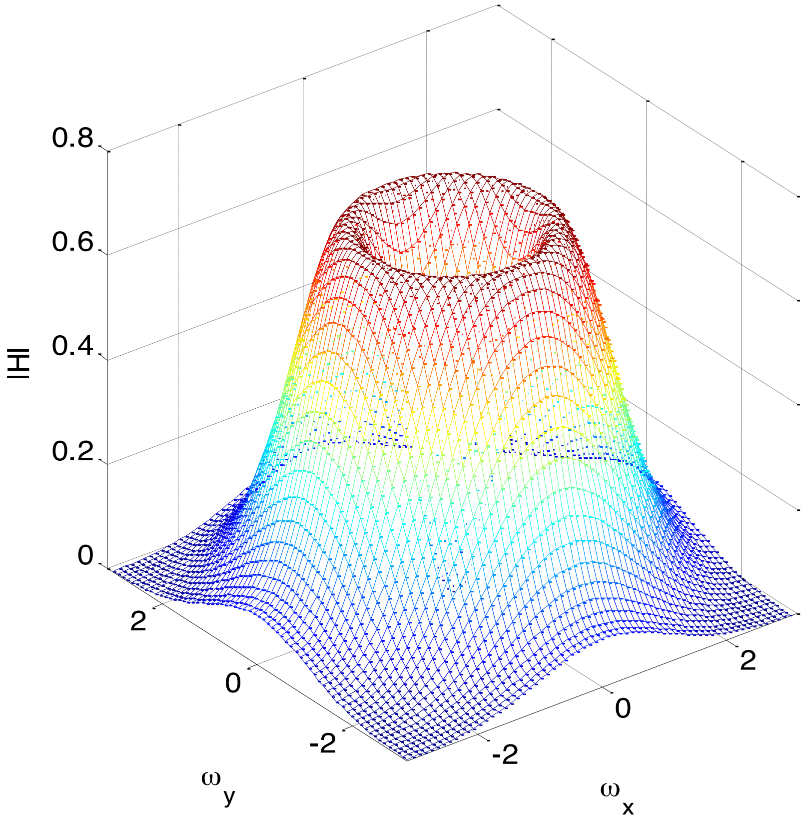
$t = \sigma^2 = 1$



$$H(\omega_x,\omega_y) = -(\omega_x^2+\omega_y^2)G^t(\omega_x,\omega_y)$$

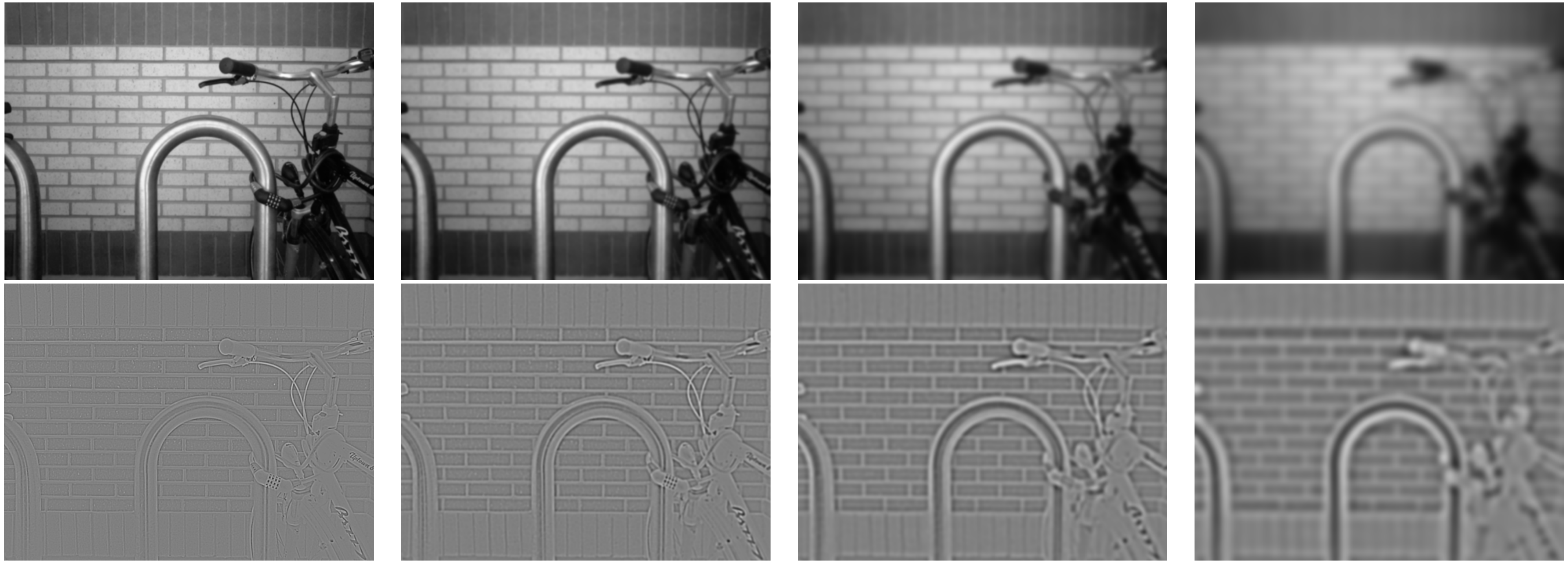
Difference of Gaussians

$t = \sigma^2 = 1, \; k = 1.1$



$$H(\omega_x,\omega_y) = \frac{1}{(k-1)t}\left[G^{k^2 t}(\omega_x,\omega_y) - G^t(\omega_x,\omega_y)\right]$$

Scale space: Laplacian images

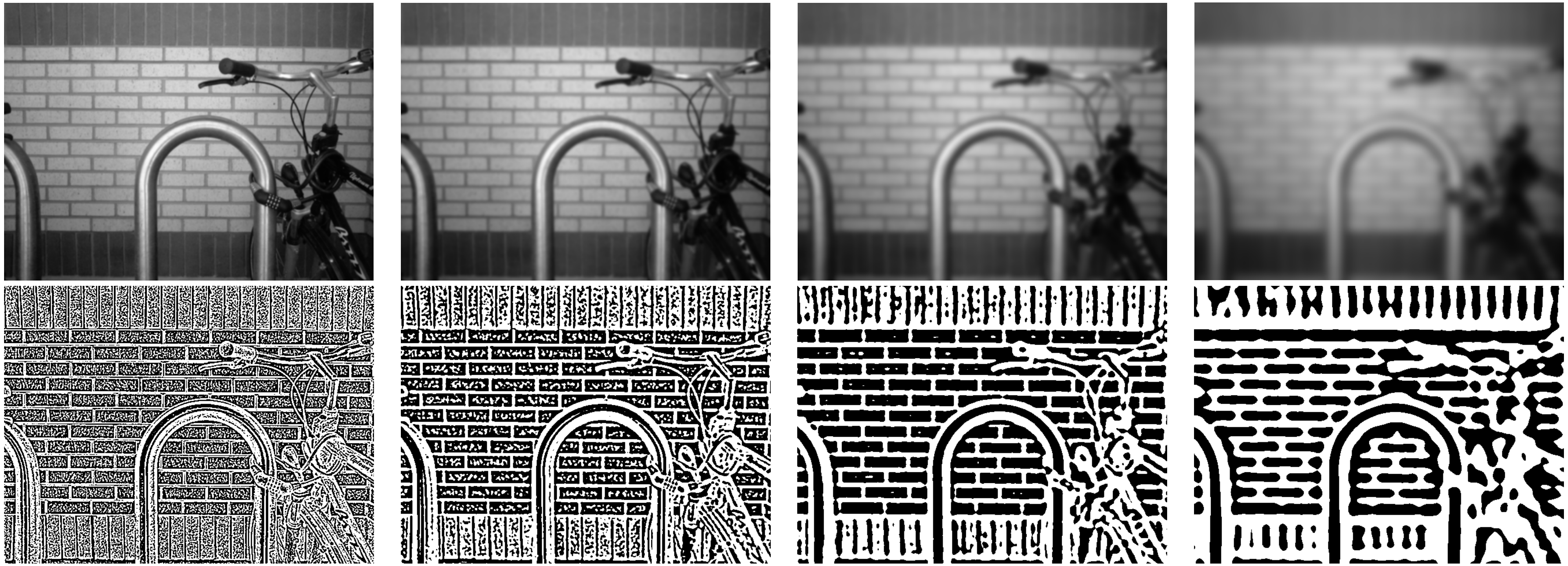


$f^t(x,y)$

$t \cdot \nabla^2 f^t(x,y)$

t = 1 t = 4 t = 16 t = 64

Scale space: Binarized Laplacian images



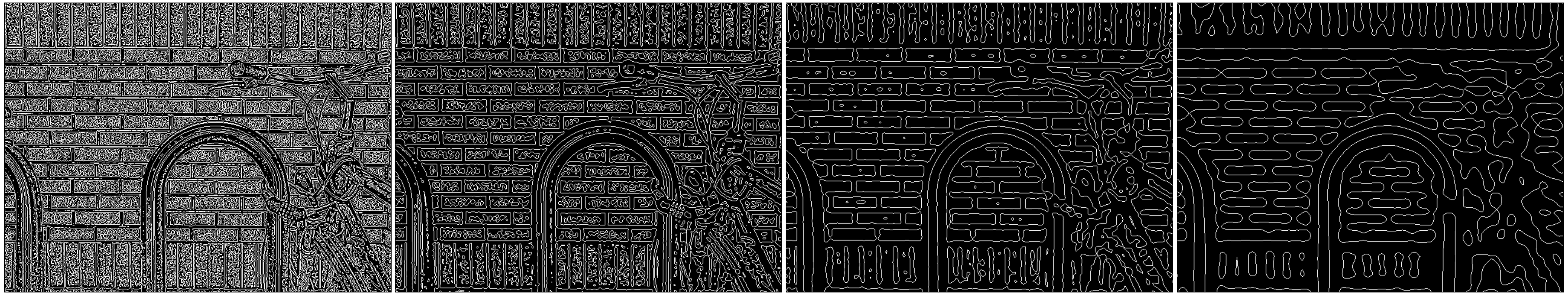
$f^t(x,y)$

$sign\left[t \cdot \nabla^2 f^t(x,y)\right]$

t = 1   t = 4   t = 16   t = 64

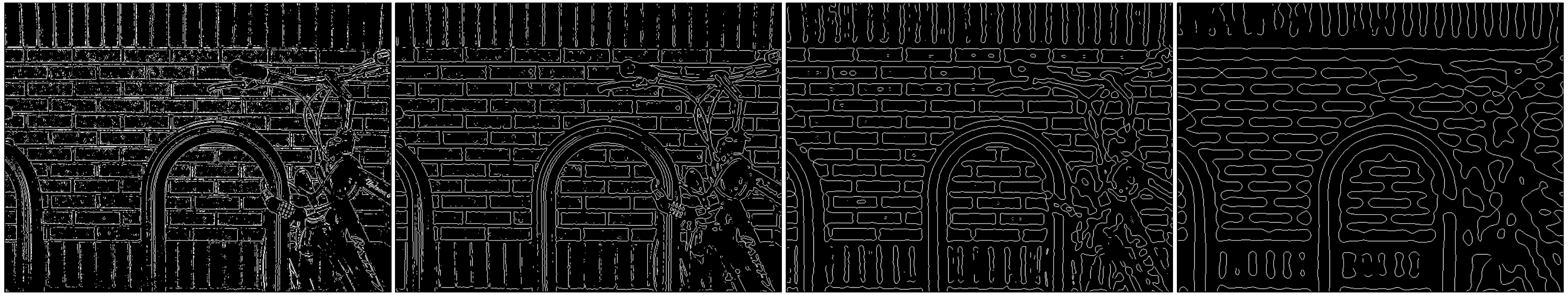
Scale space: edge detection

Zero crossings of Laplacian images

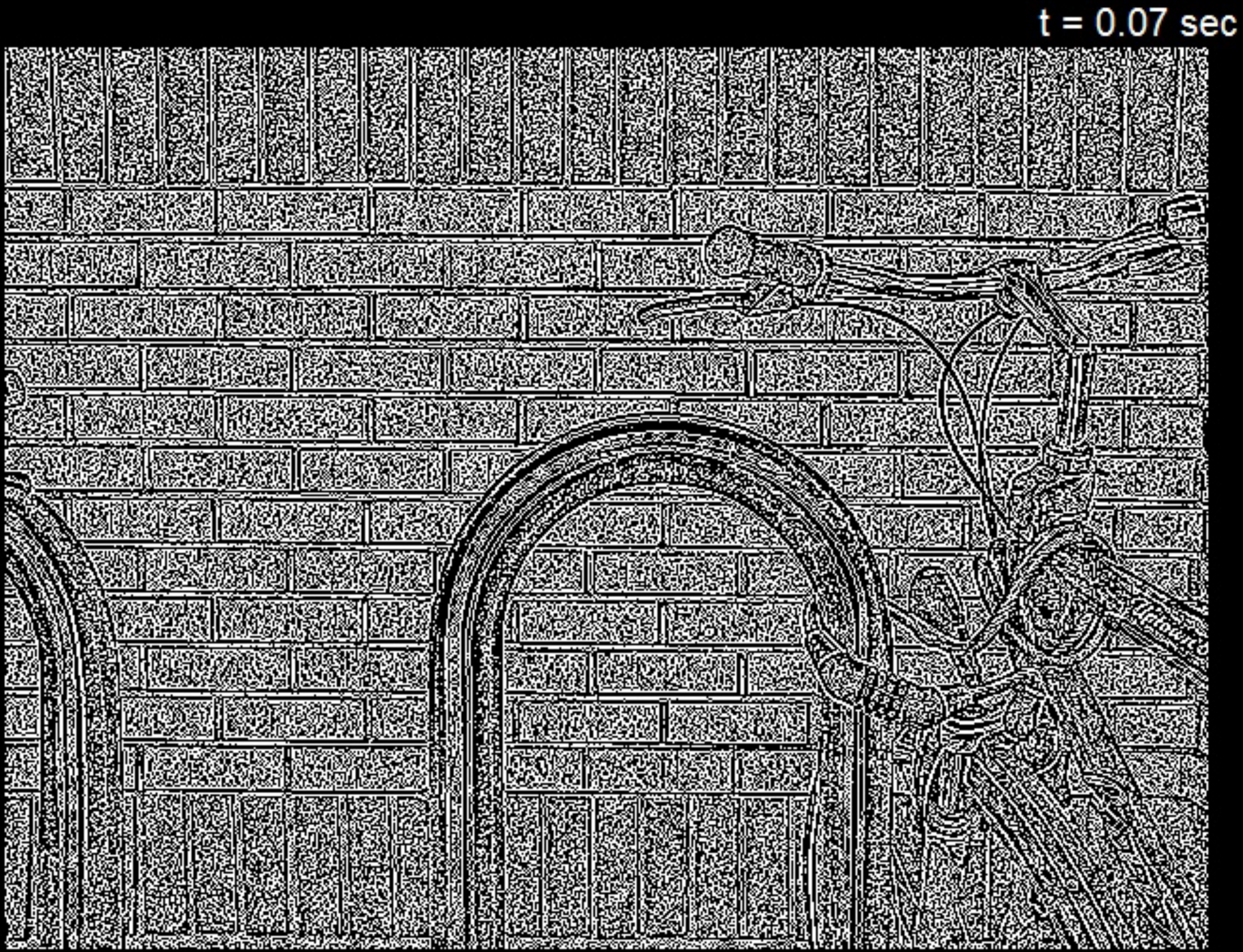


t = 1 t = 4 t = 16 t = 64

Low-gradient-magnitude edges removed



Laplacian zero-crossings

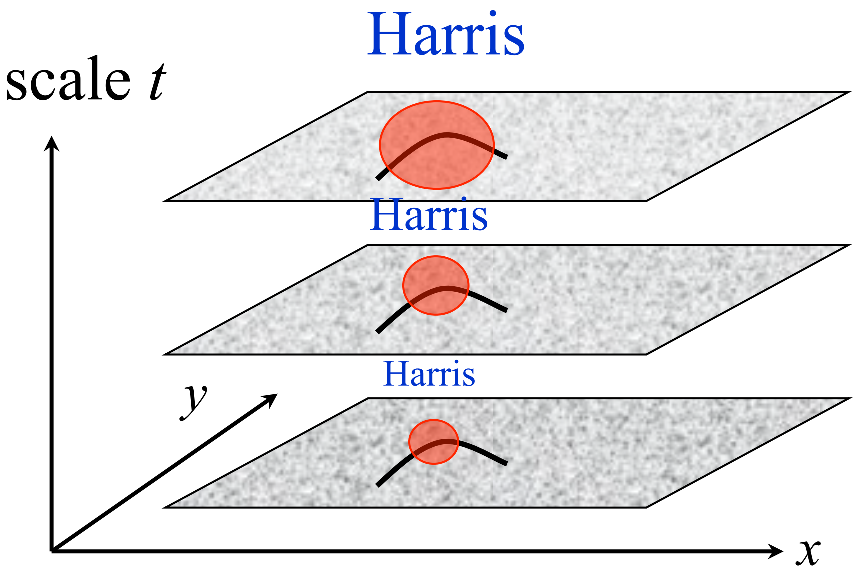


Keypoint detection with automatic scale selection

- Scale-space representation provides all scales; which scale is best for keypoint detection?
- *Harris-Laplacian*
  1. Detect Harris corners at some initial scale
  2. For each Harris corner x_h, y_h detect characteristic scale

$$t_h = \arg\max_t \left| t \cdot \nabla^2 f^t \left(x_h, y_h \right) \right|$$

  3. Apply Harris detector in a spatial neighborhood at scale t_h to refine keypoint location x_h, y_h
  4. Repeat 2. and 3. until convergence



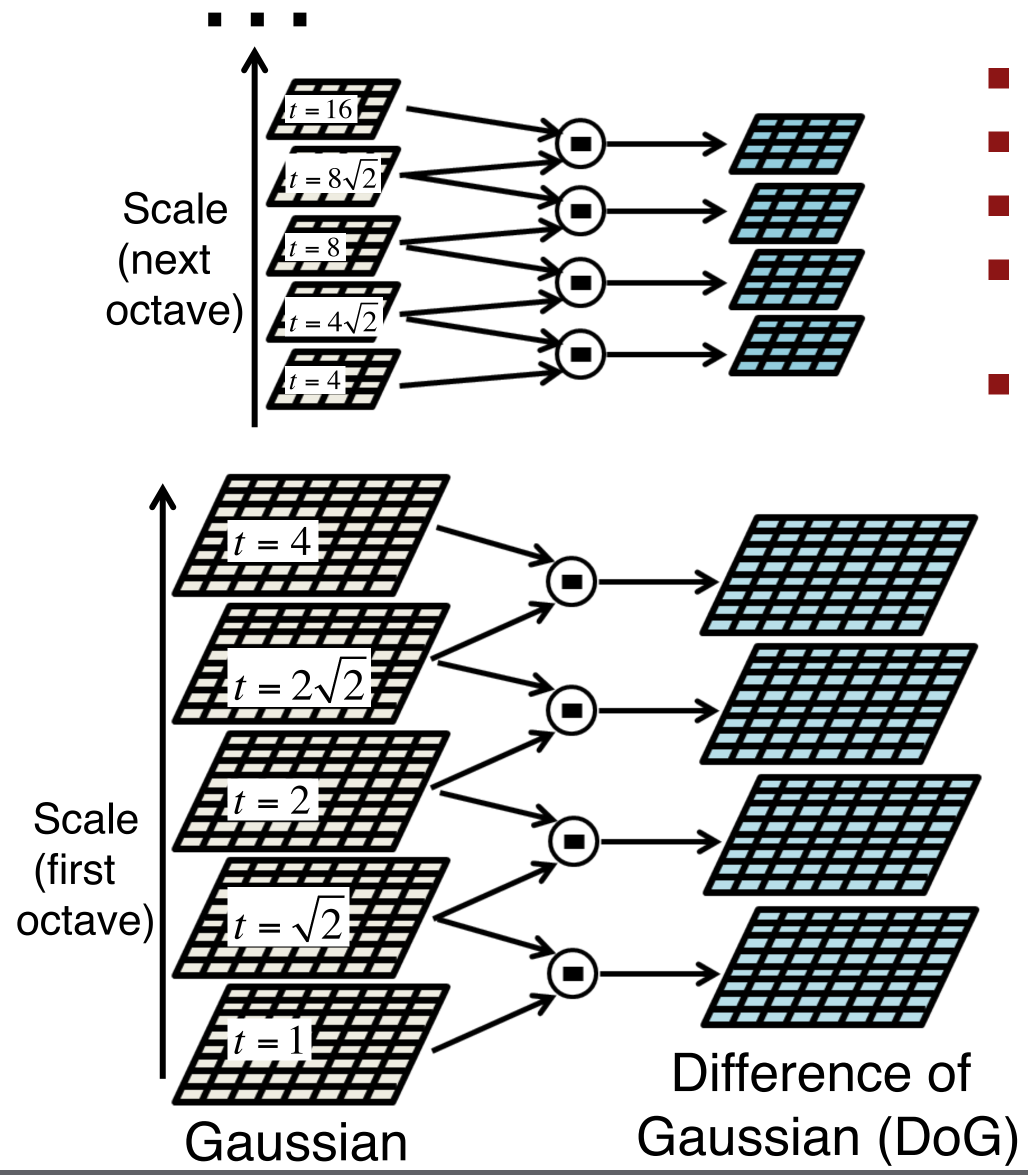
Keypoint detection with automatic scale selection

Harris-Laplacian example (150 strongest peaks)

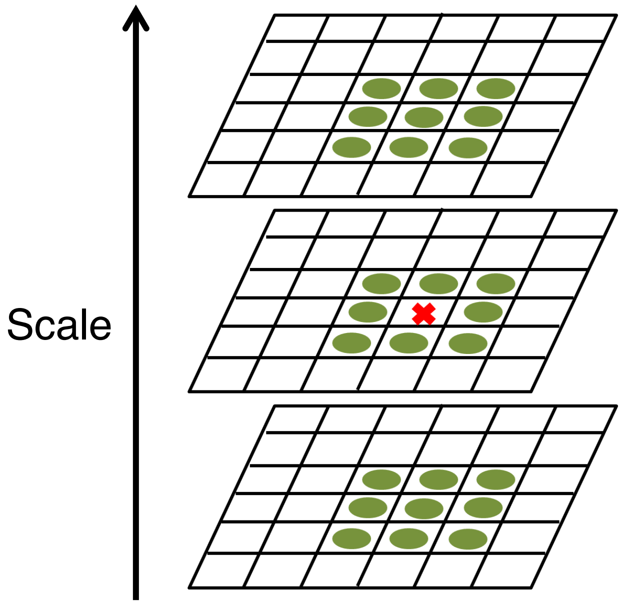
Keypoint detection with automatic scale selection

Harris-Laplacian example (200 strongest peaks)

SIFT keypoint detection

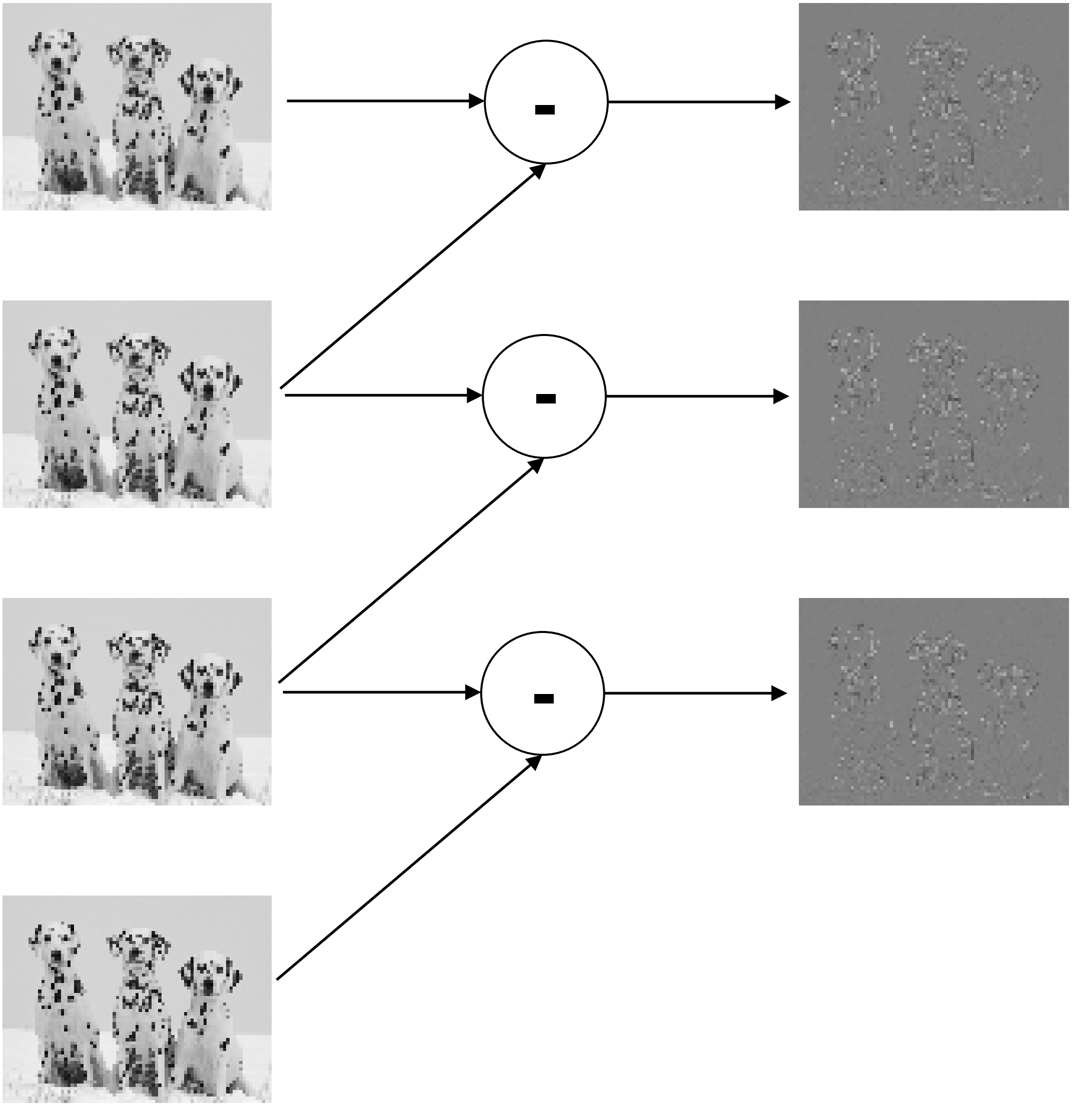


- SIFT - Scale-Invariant Feature Transform
- Decompose image into DoG scale-space representation
- Detect minima and maxima locally and across scales
- Fit 3-d quadratic function to localize extrema with sub-pixel/sub-scale accuracy *[Brown, Lowe, 2002]*
- Eliminate edge responses based on Hessian

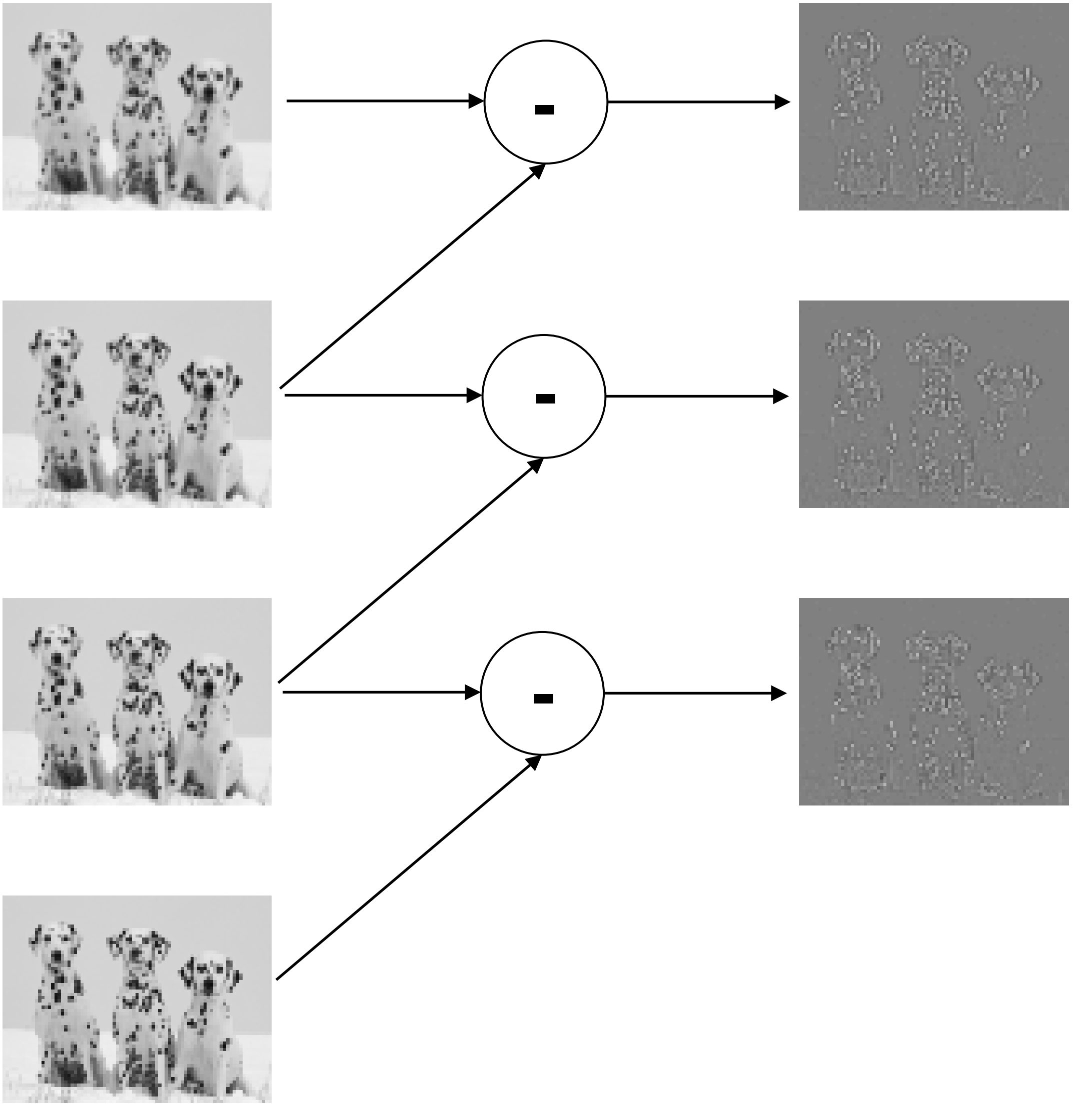


[Lowe, 1999, 2004]

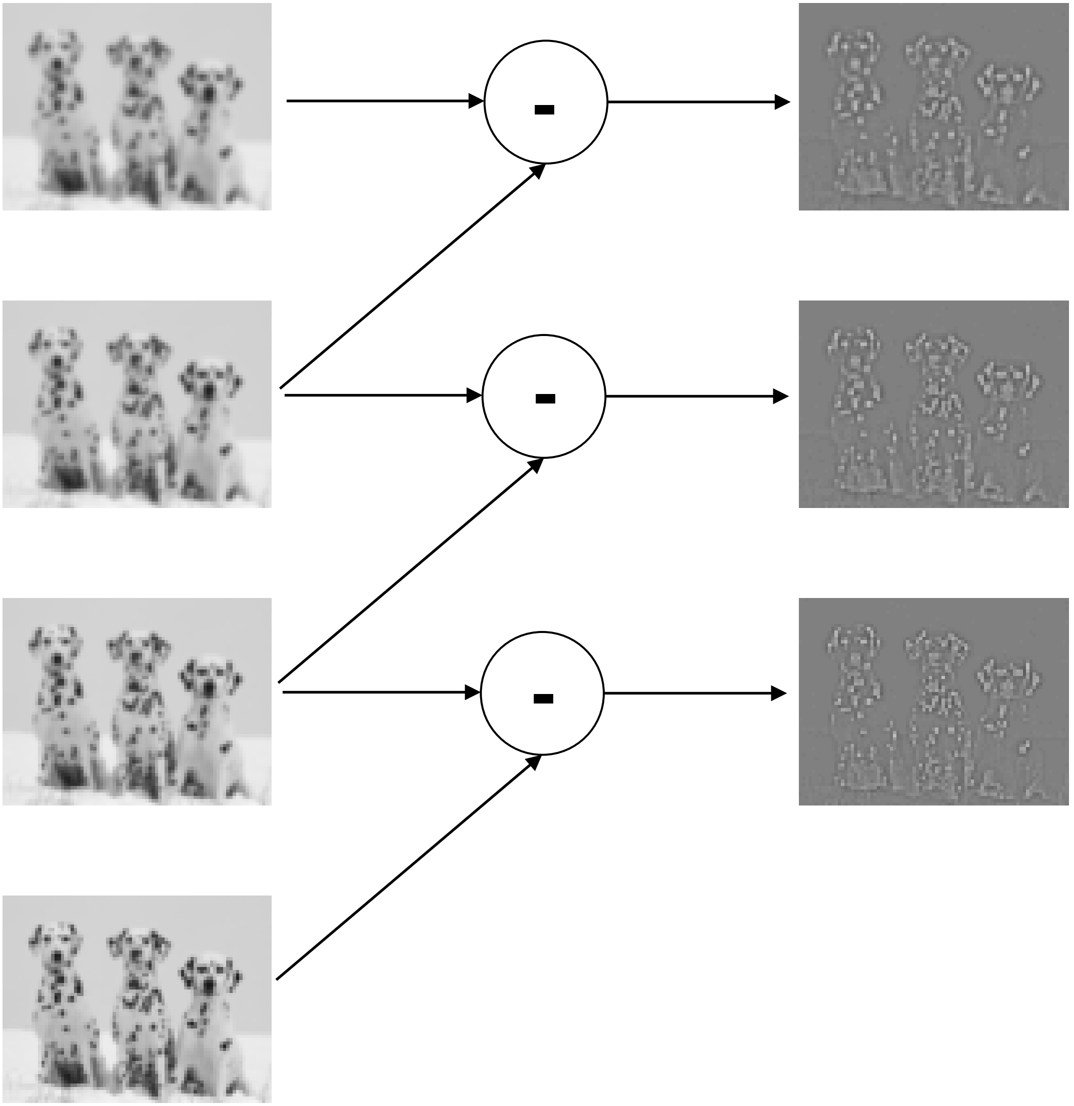
SIFT scale space pyramid: octave 1



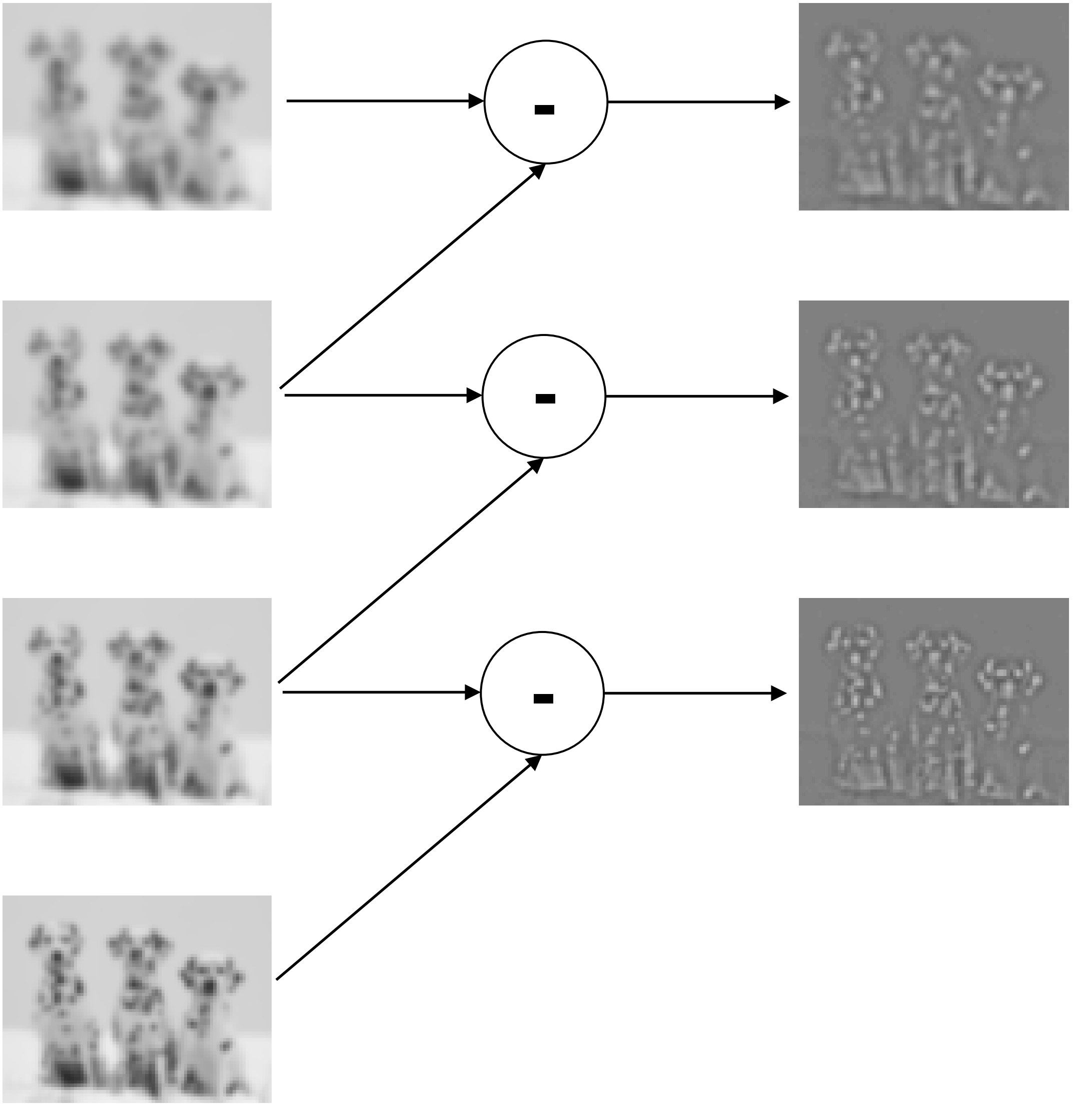
SIFT scale space pyramid: octave 2



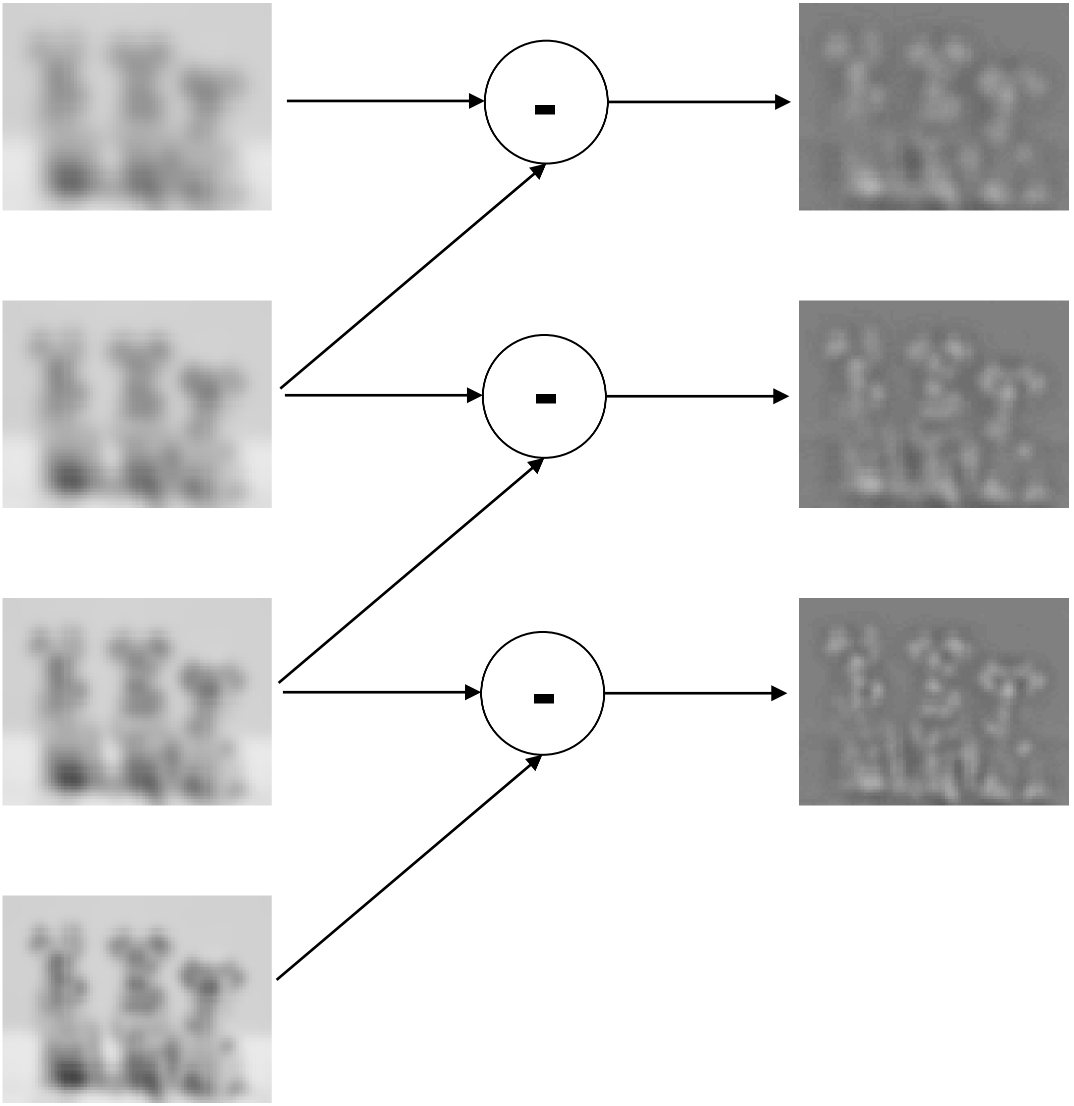
SIFT scale space pyramid: octave 3



SIFT scale space pyramid: octave 4



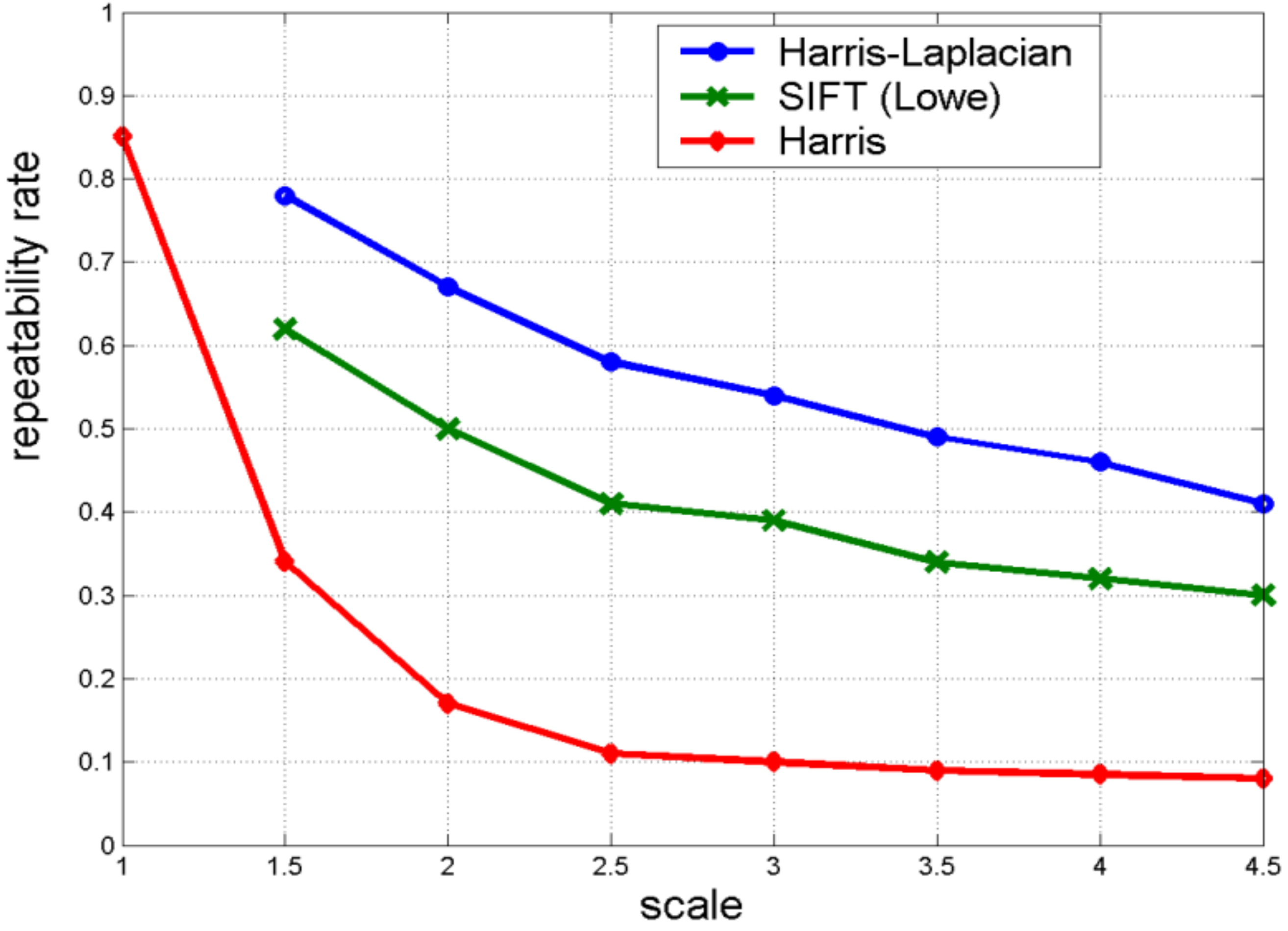
SIFT scale space pyramid: octave 5



SIFT keypoints

SIFT keypoints

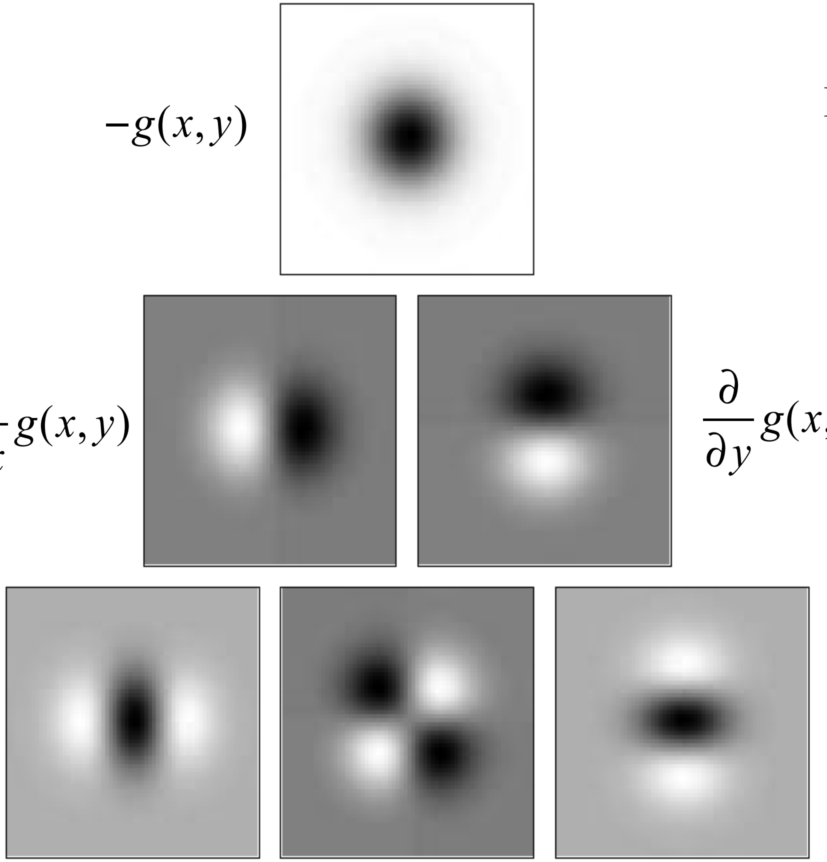
Robustness against scaling



[Mikolajczyk, Schmid, 2001]

Hessian keypoints in scale space

$$\mathbf{H}^t(x,y) = \begin{bmatrix} \frac{\partial^2}{\partial x^2} f^t(x,y) & \frac{\partial^2}{\partial x \partial y} f^t(x,y) \\ \frac{\partial^2}{\partial x \partial y} f^t(x,y) & \frac{\partial^2}{\partial y^2} f^t(x,y) \end{bmatrix}$$



$$-g(x,y)$$

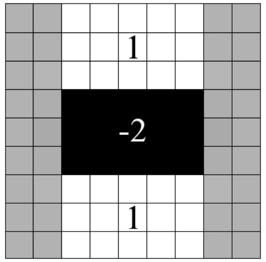
$$\frac{\partial}{\partial x} g(x,y) \qquad \frac{\partial}{\partial y} g(x,y)$$

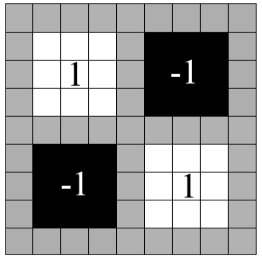
$$\frac{\partial^2}{\partial x^2} g(x,y) \qquad \frac{\partial^2}{\partial x \partial y} g(x,y) \qquad \frac{\partial^2}{\partial y^2} g(x,y)$$

SURF keypoint detection

- SURF – Speeded Up Robust Features *[Bay, Tuytelaars, Van Gool, ECCV 2006]*
- No subsampling – all resolution levels at full spatial resolution
- Simple approximation of scale space Gaussian derivatives using integral images

D^t_{yy}





D^t_{xy}

- Determinant of Hessian

$$\det\left(\mathbf{H}^t\right) \approx D^t_{xx} D^t_{yy} - \left(0.9\, D^t_{xy}\right)^2$$

- Non-maximum suppression in 3x3x3 $[x,y,t]$ neighborhood
- Interpolation of maximum of $\det(\mathbf{H})$ in image space x,y and scale t

SURF keypoints

SIFT keypoints

SURF keypoints

SIFT keypoints