

# Scale-space image processing

- Corresponding image features can appear at different scales



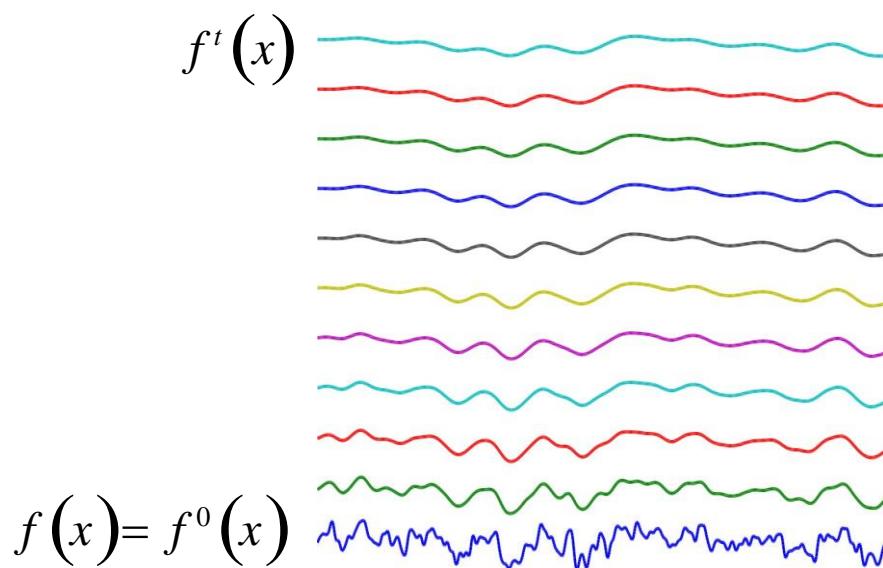
- Like shift-invariance, **scale-invariance** of image processing algorithms is often desirable.
- Scale-space representation is useful to process an image in a manner that is both shift-invariant and scale-invariant

# Scale-space image processing

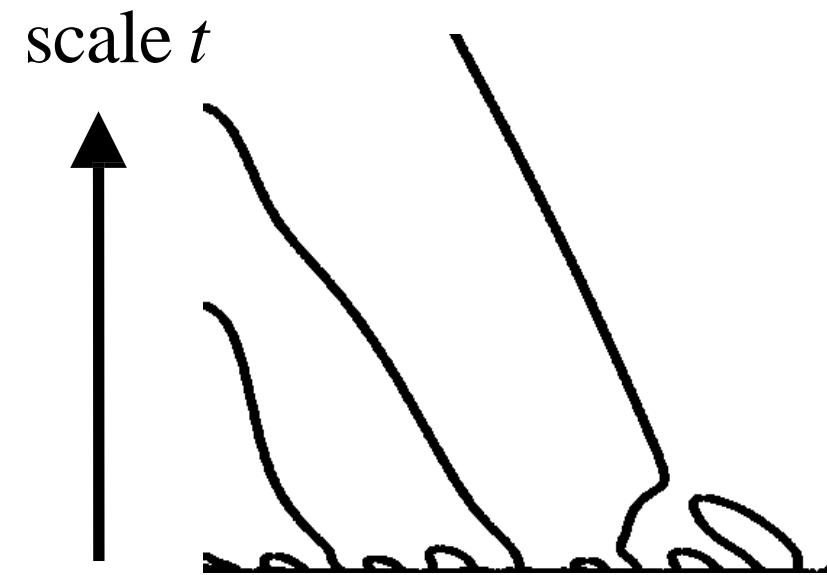
- Scale-space theory
- Laplacian of Gaussian (LoG) and Difference of Gaussian (DoG)
- Scale-space edge detection
- Scale-space keypoint detection
  - Harris-Laplacian
  - SIFT detector
  - SURF detector

# 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



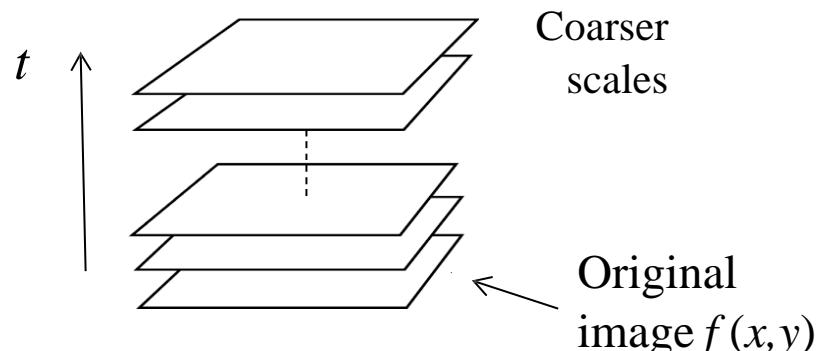
Zero-crossings of 2<sup>nd</sup> derivative  $f''(x)$   
Fewer edges at coarser scales



# Scale-space representation of images

- Parametric family of images smoothed by Gaussian filter

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



- 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

$$\begin{aligned} f^{t_1+t_2}(x, y) &= g^{t_1}(x, y) * f^{t_2}(x, y) = g^{t_2}(x, y) * f^{t_1}(x, y) \\ &= g^{t_1}(x, y) * g^{t_2}(x, y) * f(x, y) \end{aligned}$$

- Separability

$$\begin{aligned} g^t(x, y) &= \frac{1}{2\pi t} \exp\left(-\frac{x^2 + y^2}{2t}\right) = \frac{1}{\sqrt{2\pi t}} \exp\left(-\frac{x^2}{2t}\right) \cdot \frac{1}{\sqrt{2\pi t}} \exp\left(-\frac{y^2}{2t}\right) \\ G^t(\omega_x, \omega_y) &= \exp\left[-\frac{t}{2}(\omega_x^2 + \omega_y^2)\right] = \exp\left[-\frac{t}{2}\omega_x^2\right] \exp\left[-\frac{t}{2}\omega_y^2\right] \end{aligned}$$

# 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)

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

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

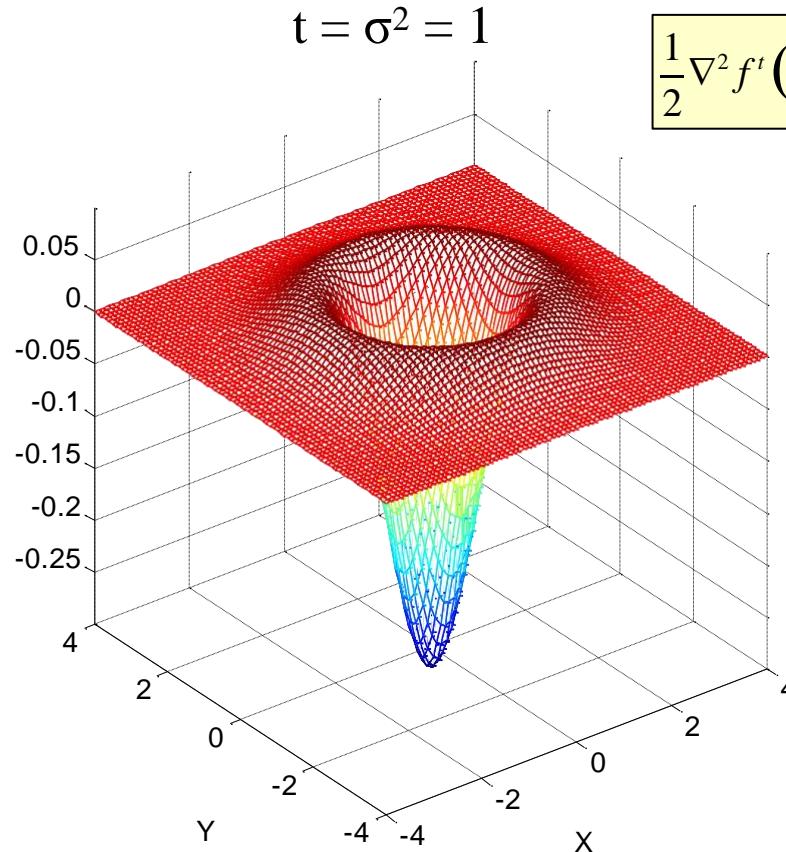
$t = 0.07 \text{ sec}$

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

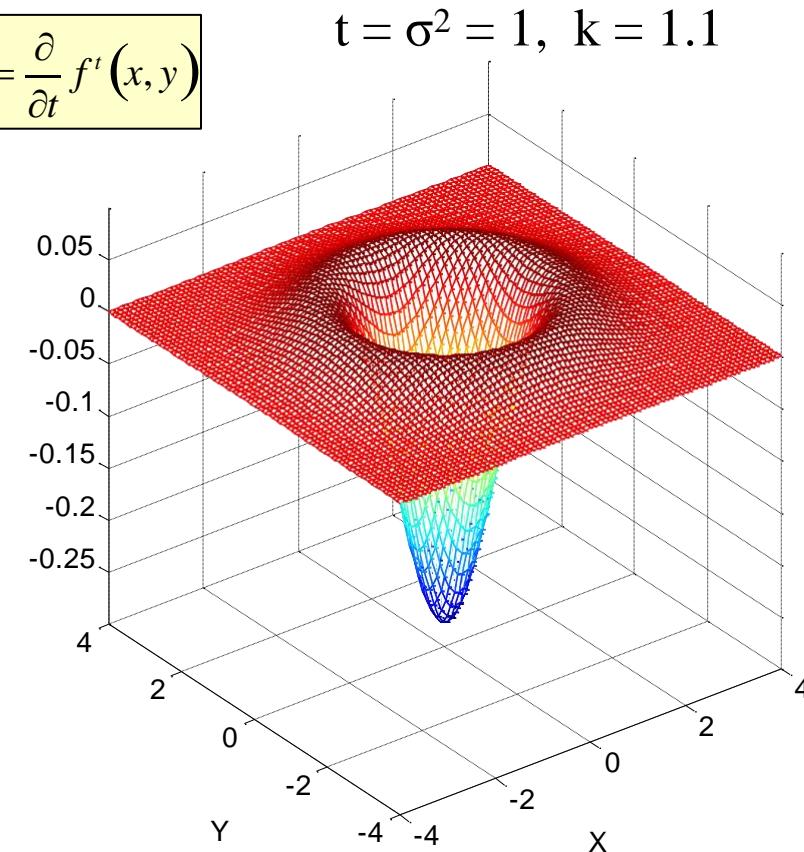


# LoG vs. DoG

Laplacian of Gaussian



Difference of Gaussians



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

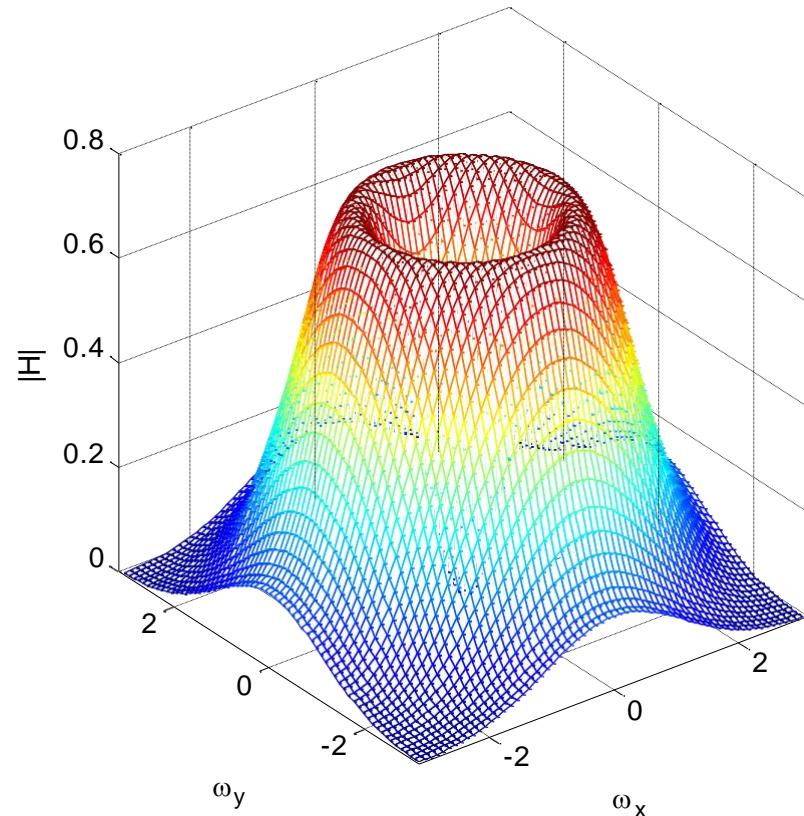
$$DoG(x, y) = \frac{1}{(k-1)t} \left( g^{k^2 t}(x, y) - g^t(x, y) \right)$$



# LoG vs. DoG (cont.)

Laplacian of Gaussian

$$t = \sigma^2 = 1$$

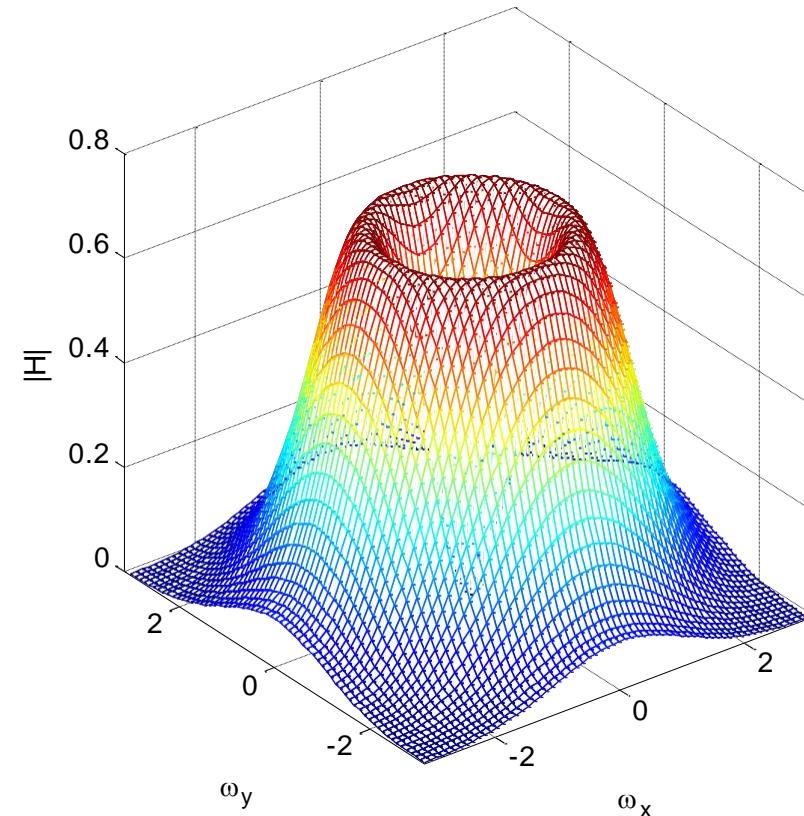


$$H(\omega_x, \omega_y) = -(\omega_x^2 + \omega_y^2) G^t(\omega_x, \omega_y)$$



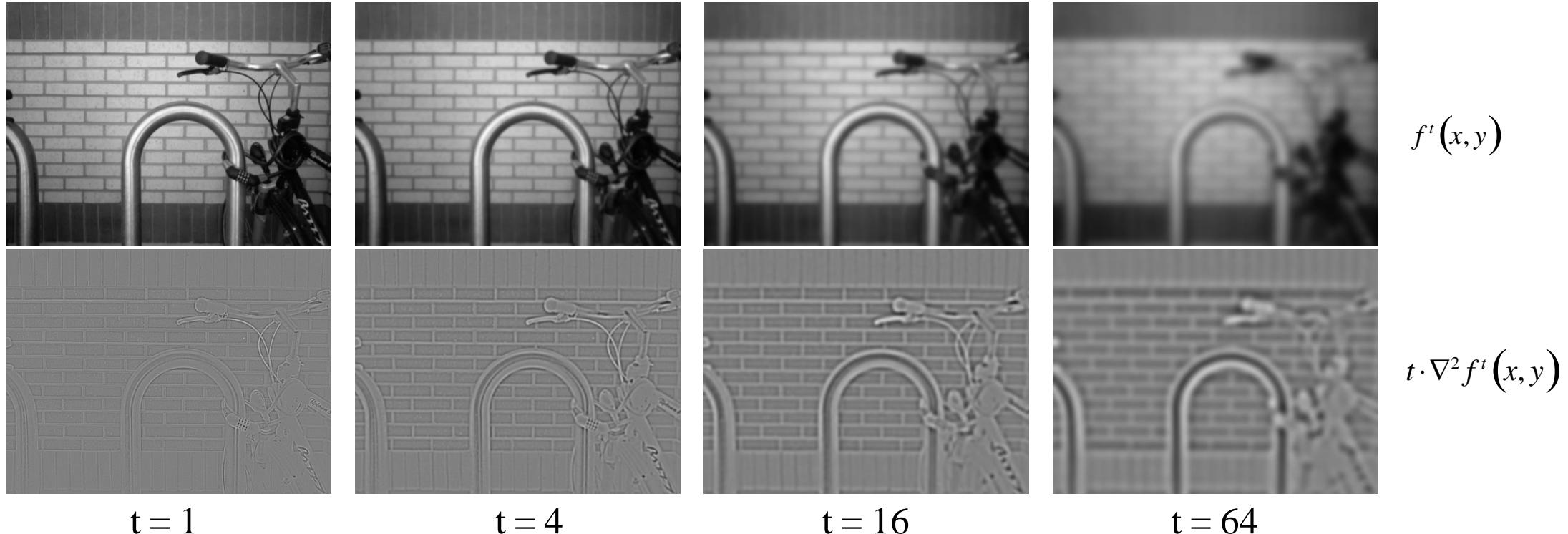
Difference of Gaussians

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

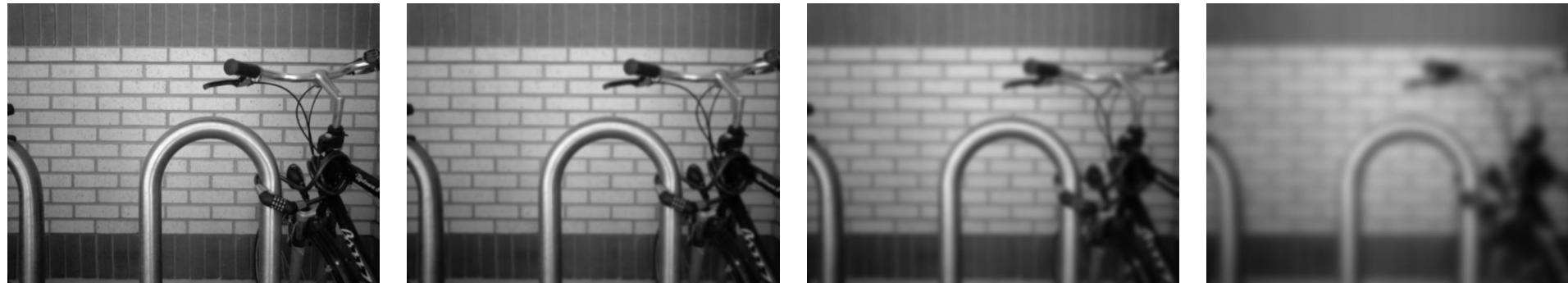


$$H(\omega_x, \omega_y) = \frac{1}{(k-1)t} [G^{k^2 t}(\omega_x, \omega_y) - G^t(\omega_x, \omega_y)]$$

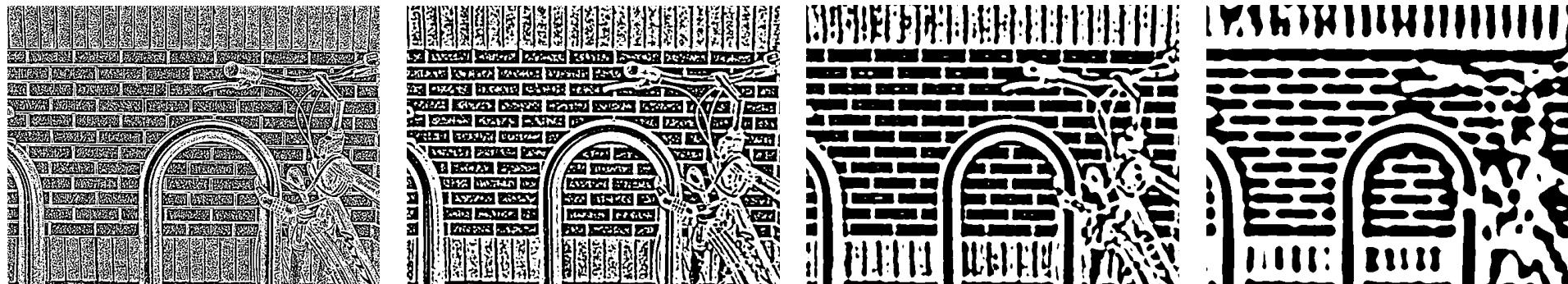
# Scale space: Laplacian images



# Scale space: binarized Laplacian images



$$f^t(x, y)$$



$$\text{sign}[t \cdot \nabla^2 f^t(x, y)]$$

$t = 1$

$t = 4$

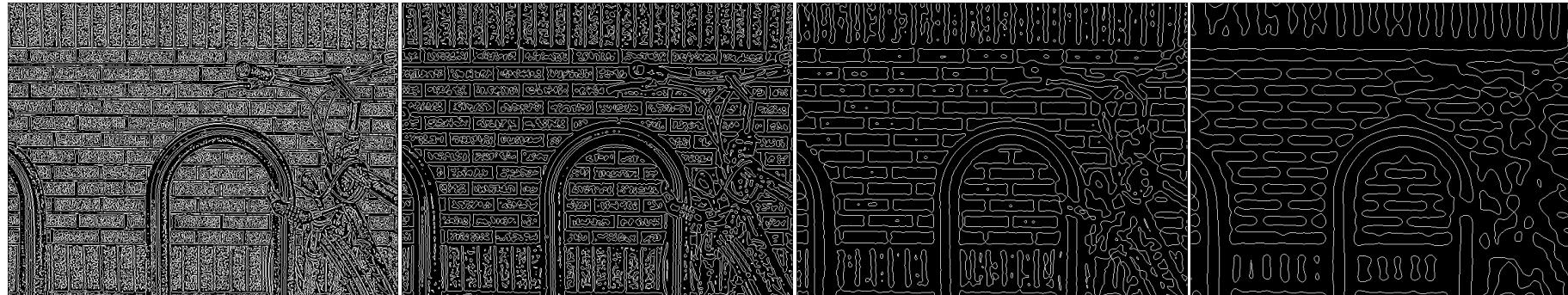
$t = 16$

$t = 64$



# Scale space: edge detection

Zero crossings of Laplacian images



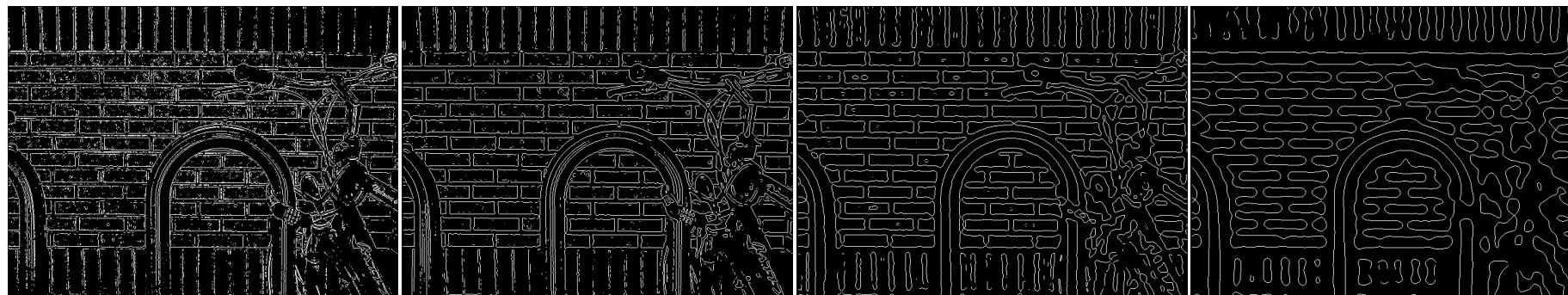
$t = 1$

$t = 4$

$t = 16$

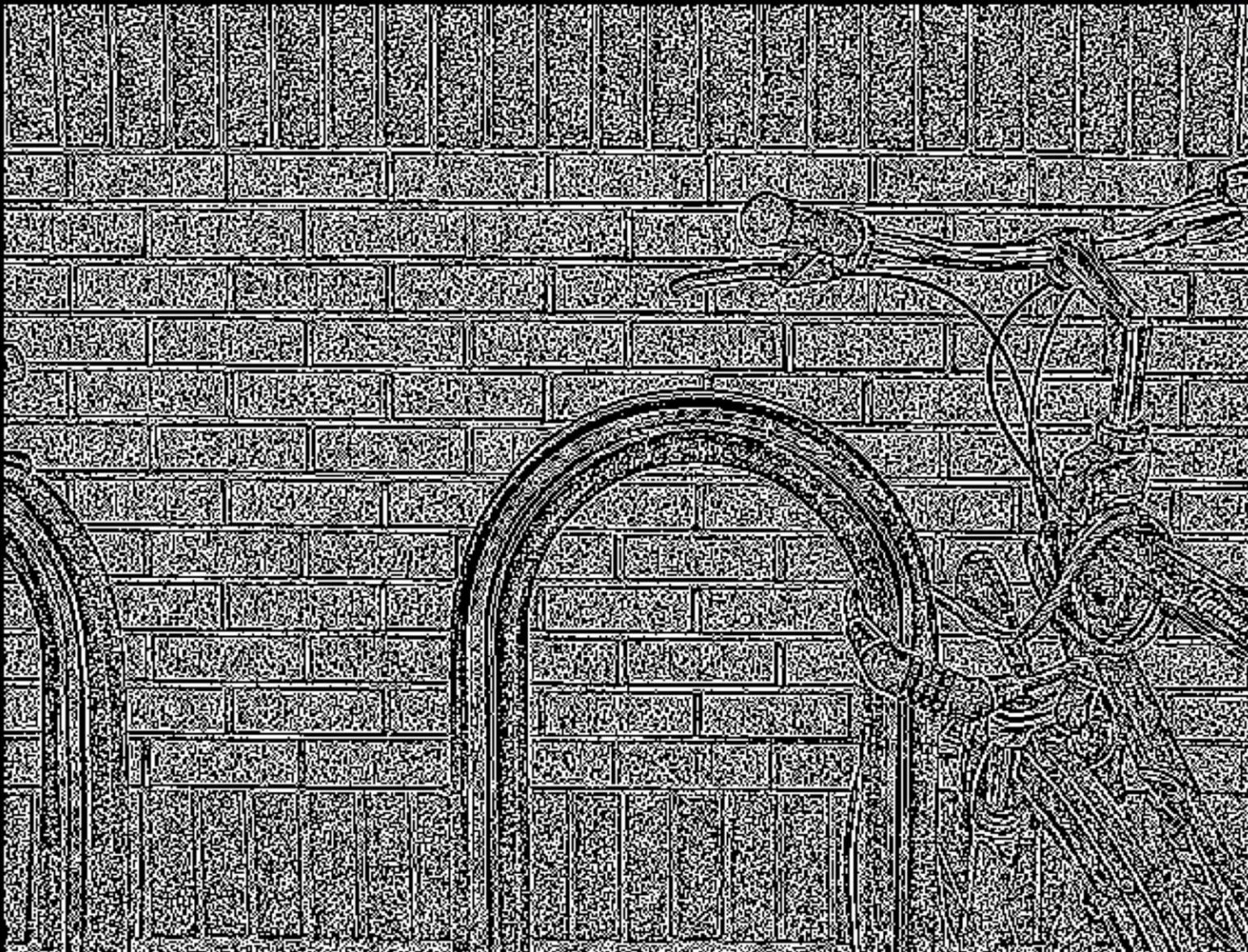
$t = 64$

Low-gradient-magnitude edges removed



# Laplacian zero-crossings

t = 0.07 sec



# Keypoint detection with automatic scale selection

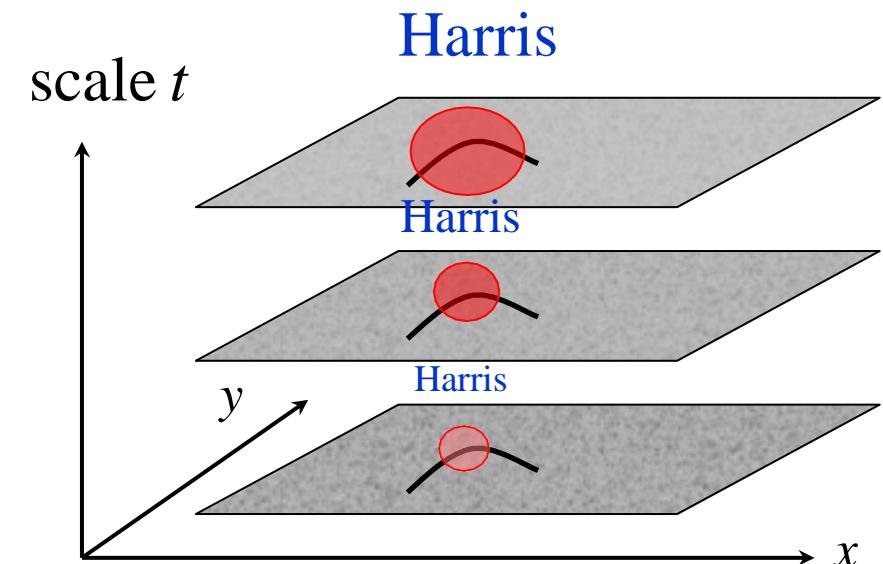
- Scale-space representation provides all scales; which scale is best for keypoint detection?

- *Harris-Laplacian*

1. Detect Harris corners at some initial scale
2. For each Harris corner  $x_h, y_h$  detect characteristic scale

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

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)



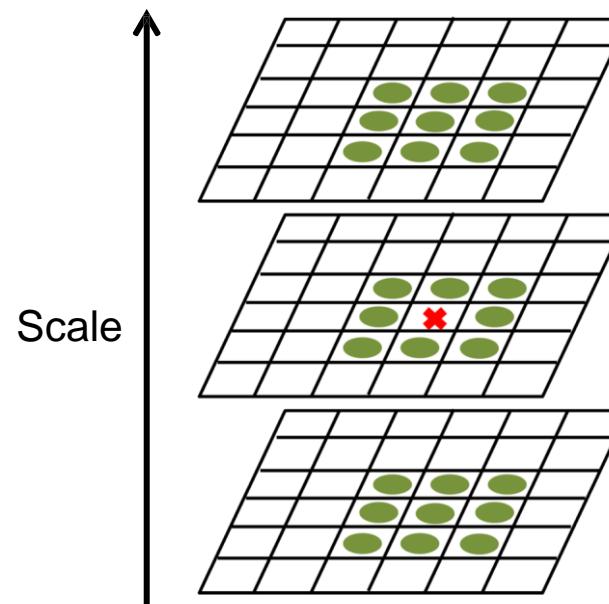
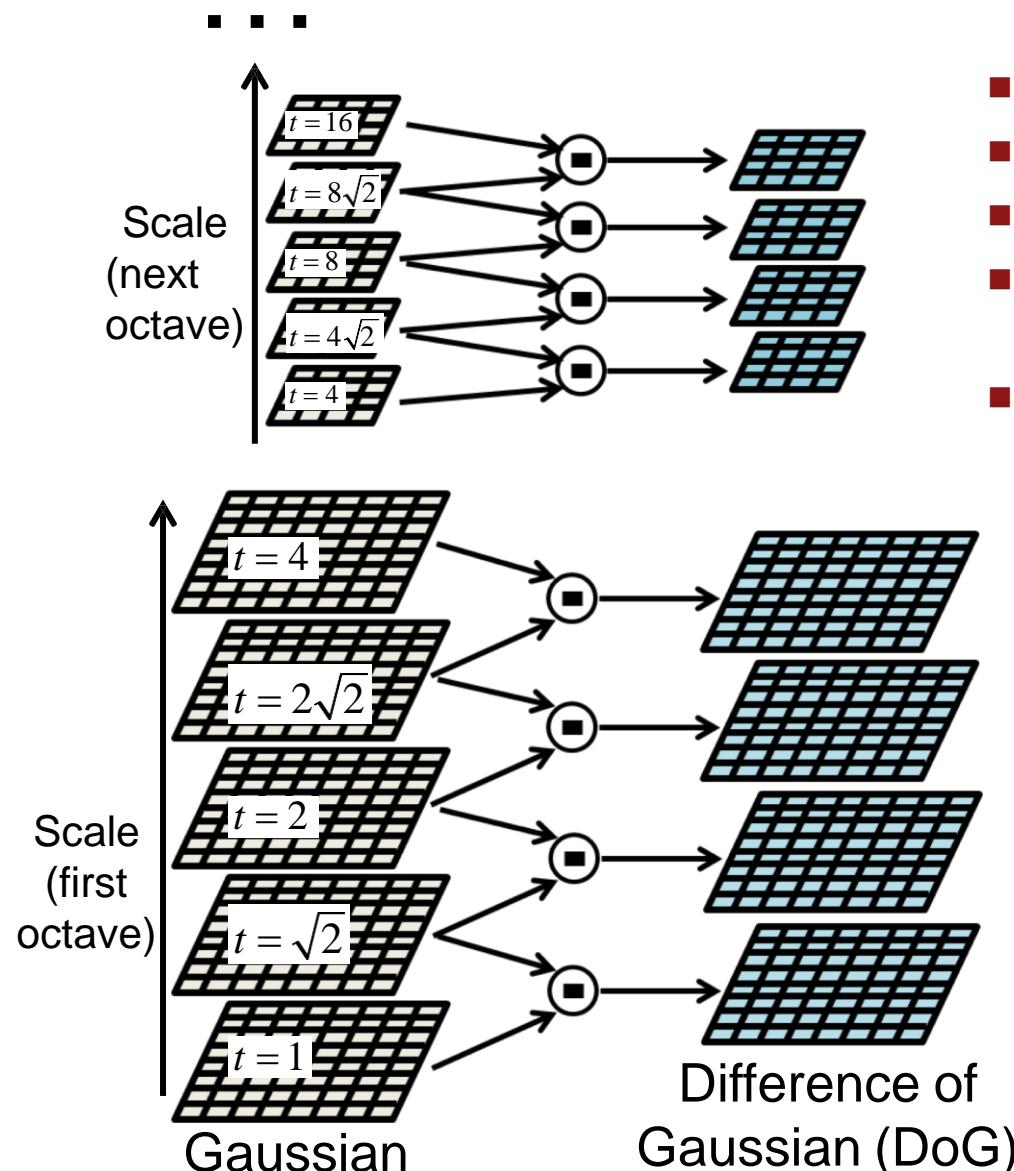
# Keypoint detection with automatic scale selection

Harris-Laplacian example (200 strongest peaks)



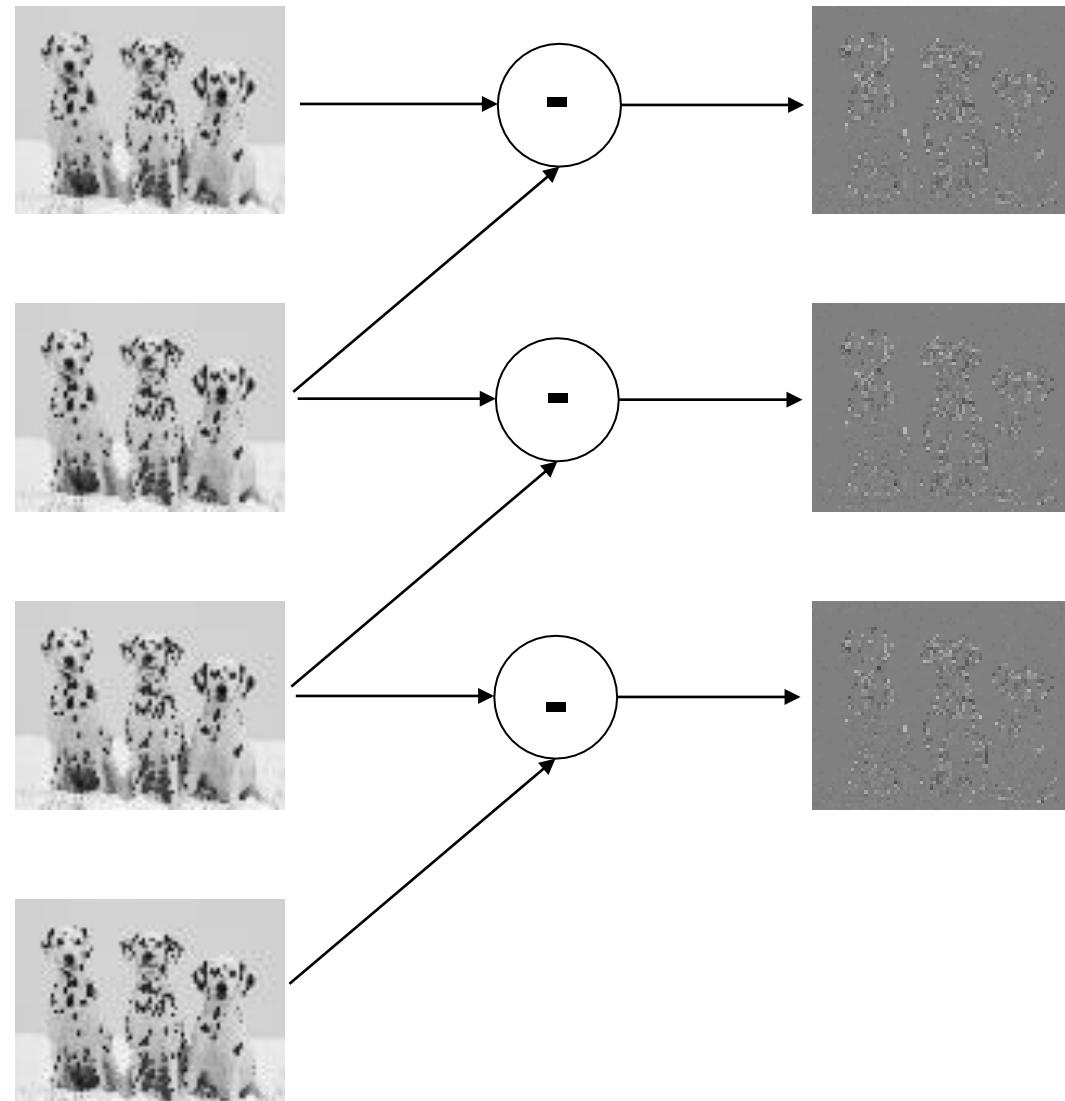
# SIFT keypoint detection

- 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

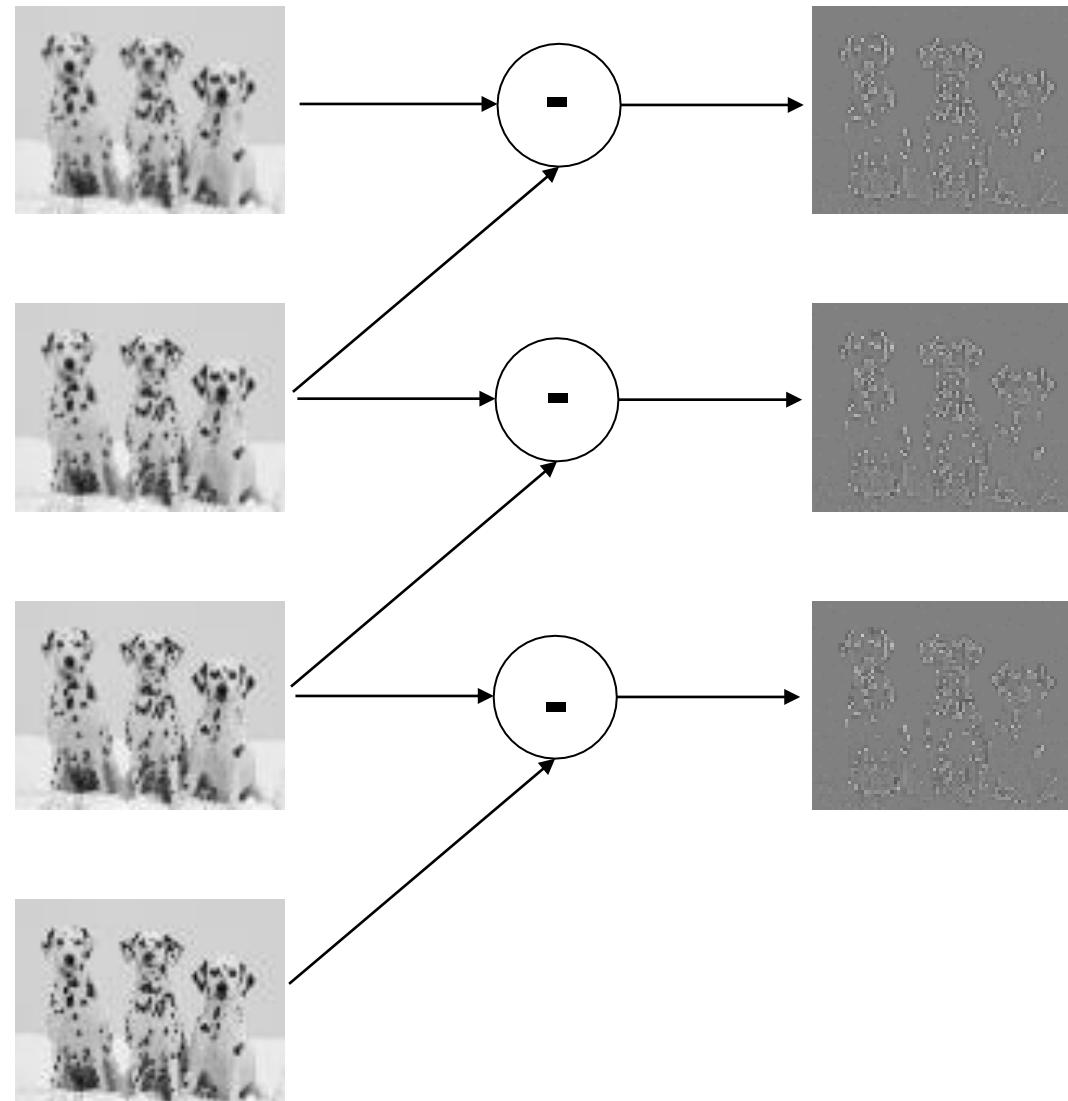


[Lowe, 1999, 2004]

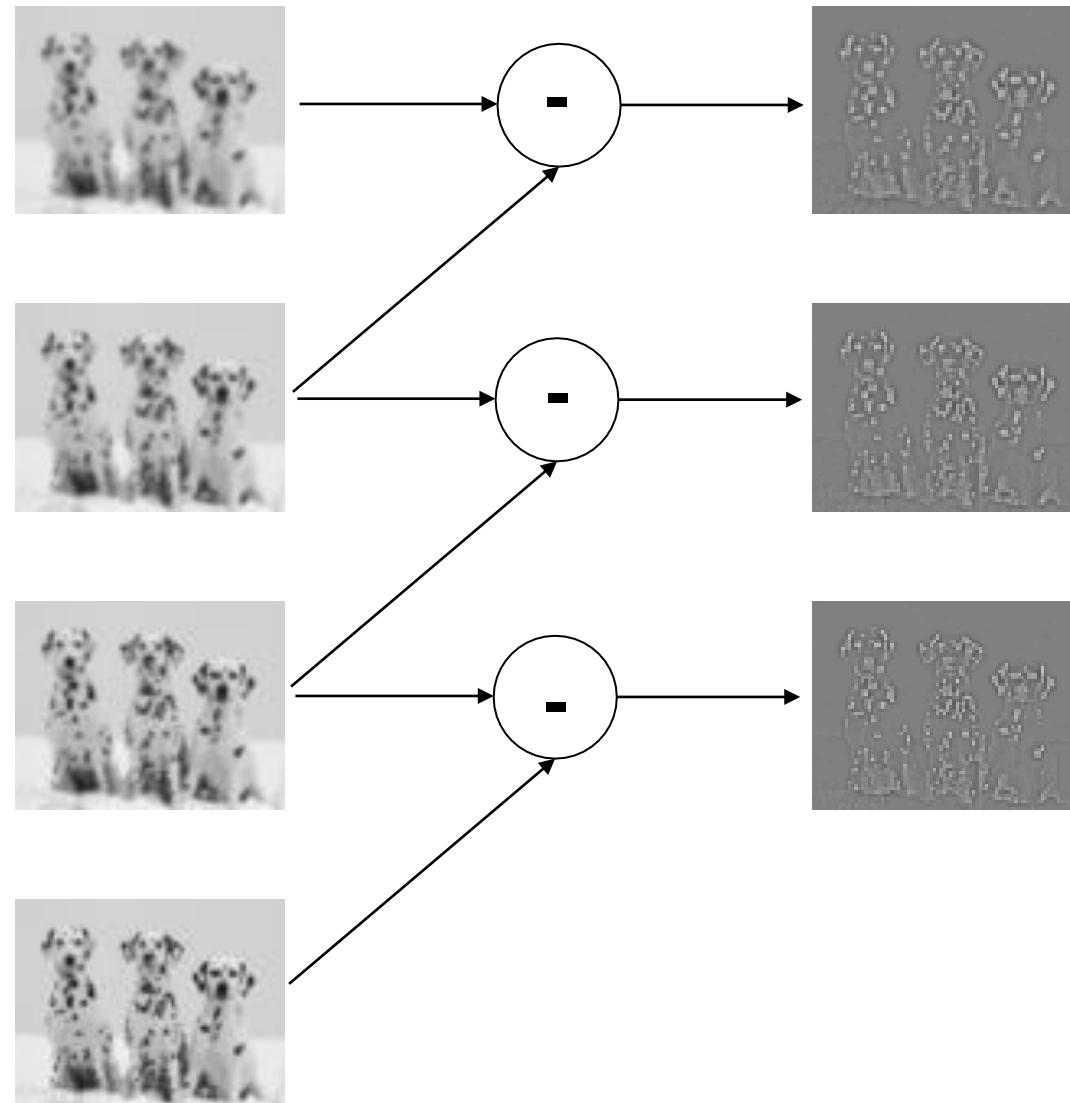
# SIFT scale space pyramid: octave 1



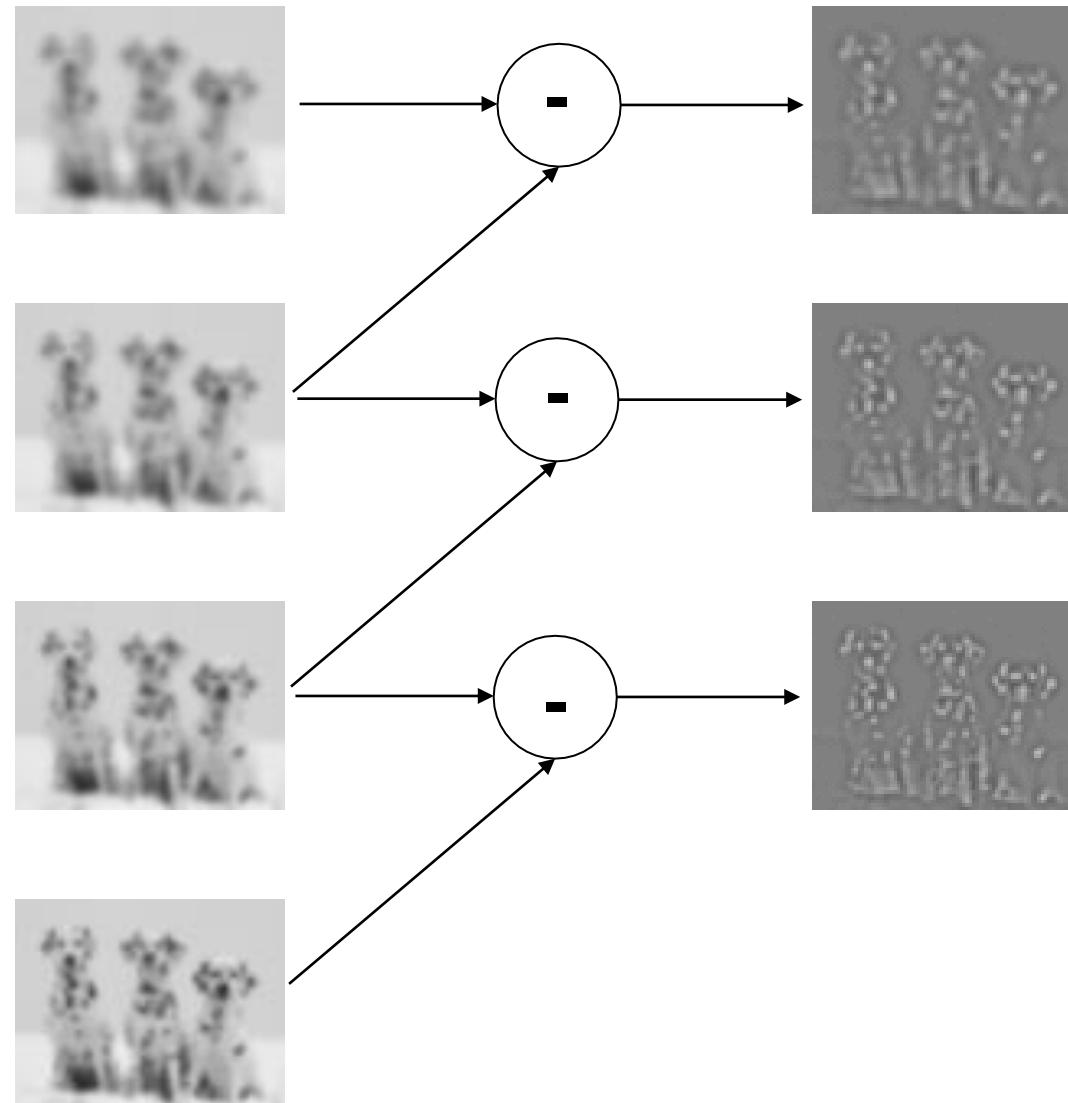
# SIFT scale space pyramid: octave 2



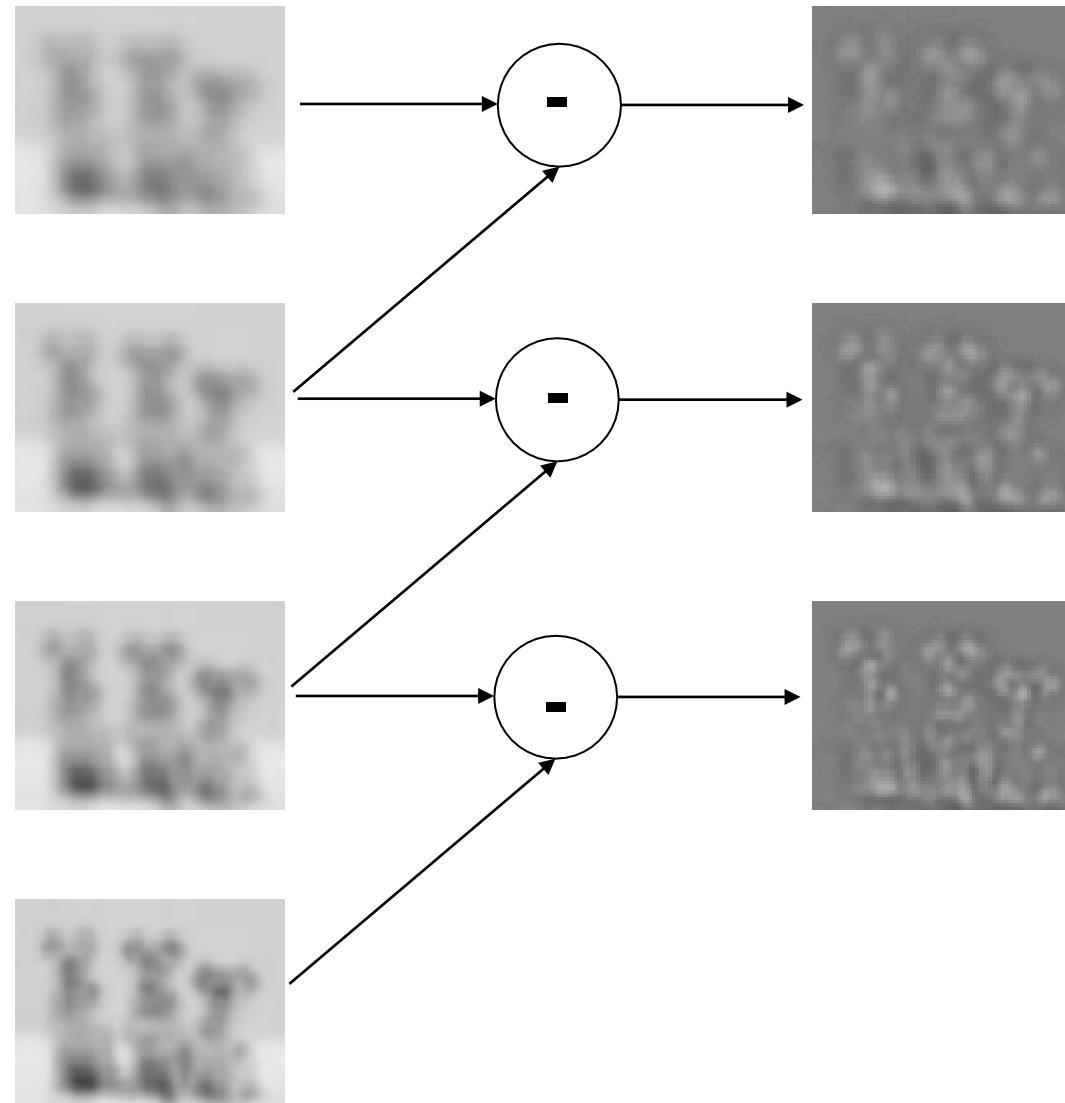
# SIFT scale space pyramid: octave 3



# SIFT scale space pyramid: octave 4



# SIFT scale space pyramid: octave 5



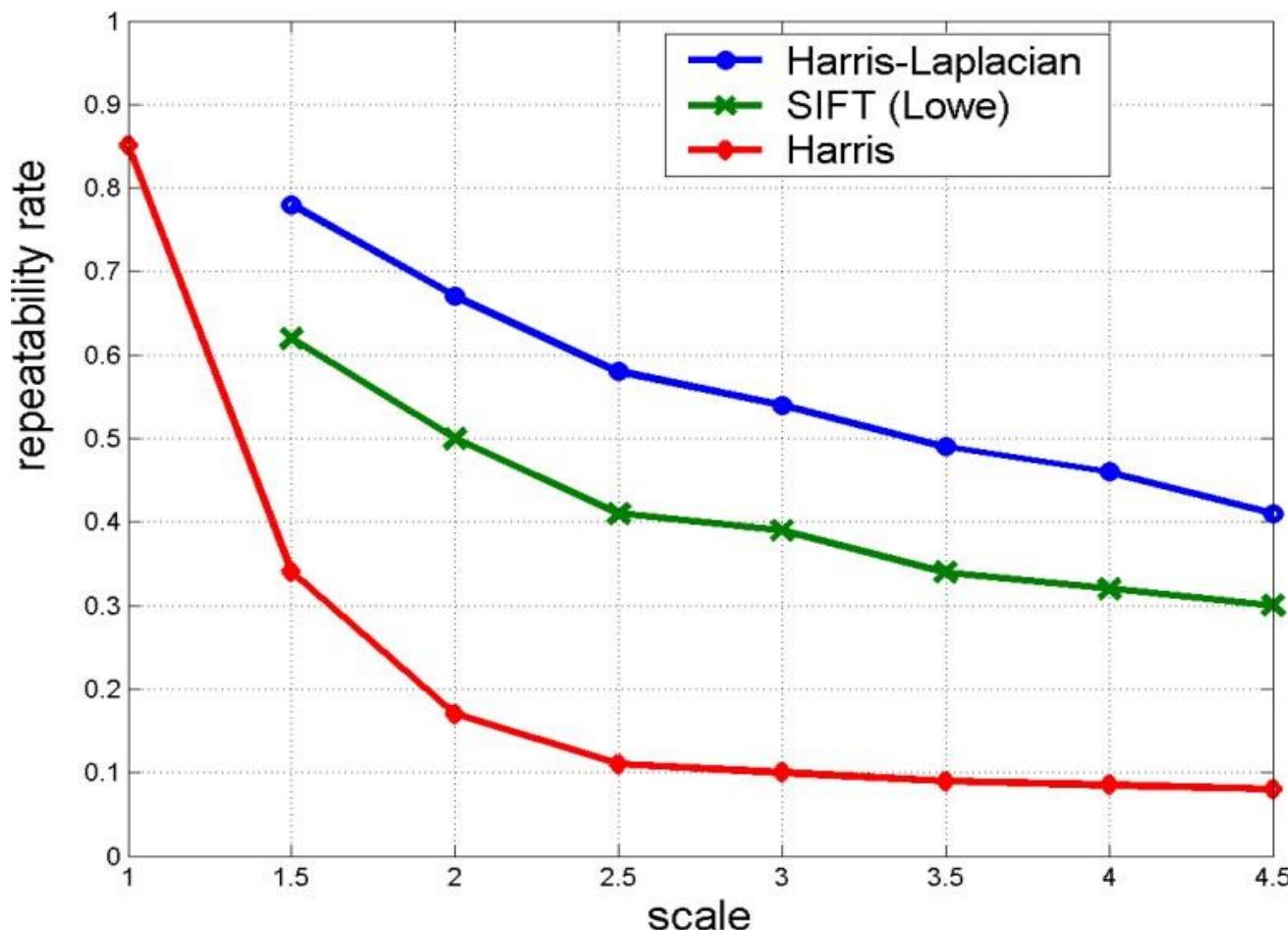
# SIFT keypoints



# SIFT keypoints

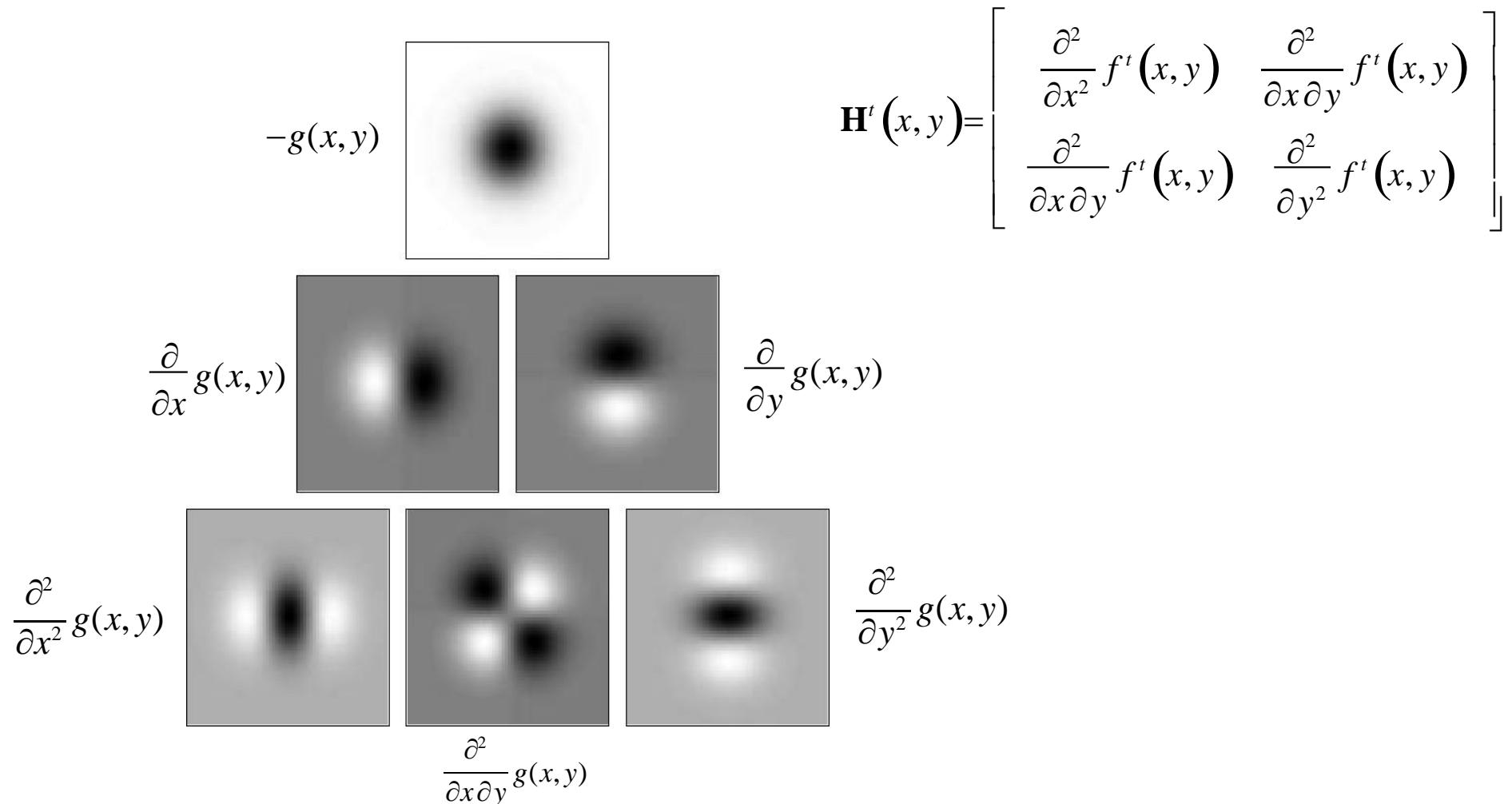


# Robustness against scaling



[Mikolajczyk, Schmid, 2001]

# Hessian keypoints in scale space



# 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

$$D_{yy}^t \quad \begin{matrix} & & 1 \\ & & -2 \\ & & 1 \end{matrix}$$
$$D_{xy}^t \quad \begin{matrix} 1 & -1 \\ -1 & 1 \end{matrix}$$

- Determinant of Hessian
- $$\det(\mathbf{H}^t) \approx D_{xx}^t D_{yy}^t - (0.9 D_{xy}^t)^2$$
- 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

