COLLEGE OF ENGINEERING
ELECTRICAL ENGINEERING & COMPUTER SCIENCE
UNIVERSITY OF MICHIGAN

Scale Invariance and Automatic Scale Selection

EECS 442 Computer Vision

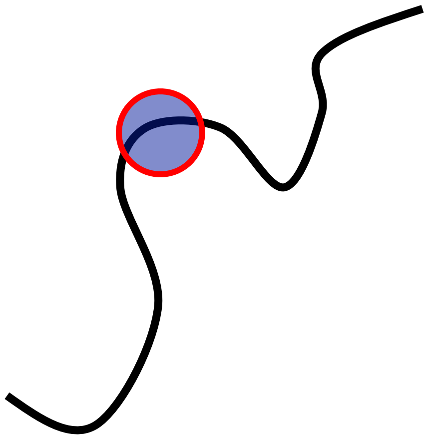
Instructor: Jason Corso (jjcorso)
web.eecs.umich.edu/~jjcorso/t/

Materials on these slides have come from many sources in addition to myself; specific sources are cited on each slide.


www.EgyptSons.com

Scale invariant detection

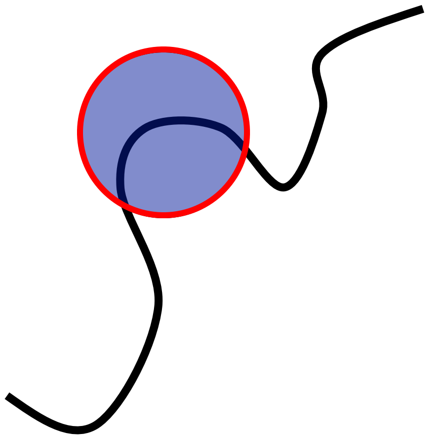
Suppose you are looking for corners



Source: S. Seitz.

Scale invariant detection

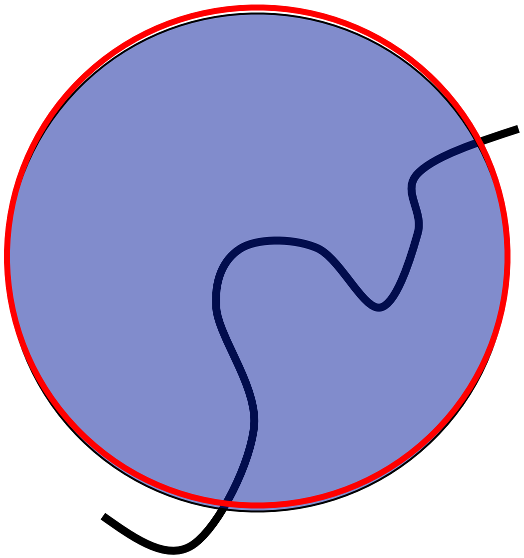
Suppose you are looking for corners



Source: S. Seitz.

Scale invariant detection

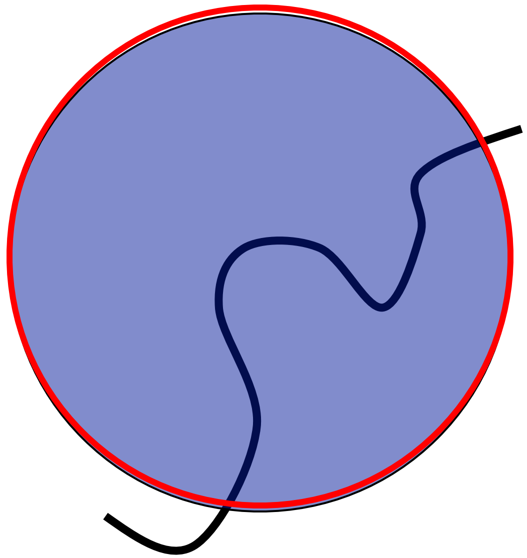
Suppose you are looking for corners



Source: S. Seitz.

Scale invariant detection

Suppose you are looking for corners



Key idea: find scale that gives local max/min of operator

- Want the operator to be a local maximum/minimum in both position and scale
- Common definition: Laplacian
  (or difference between two Gaussian filtered images with different sigmas)

Source: S. Seitz.

Scale-Invariant Feature Detection Example

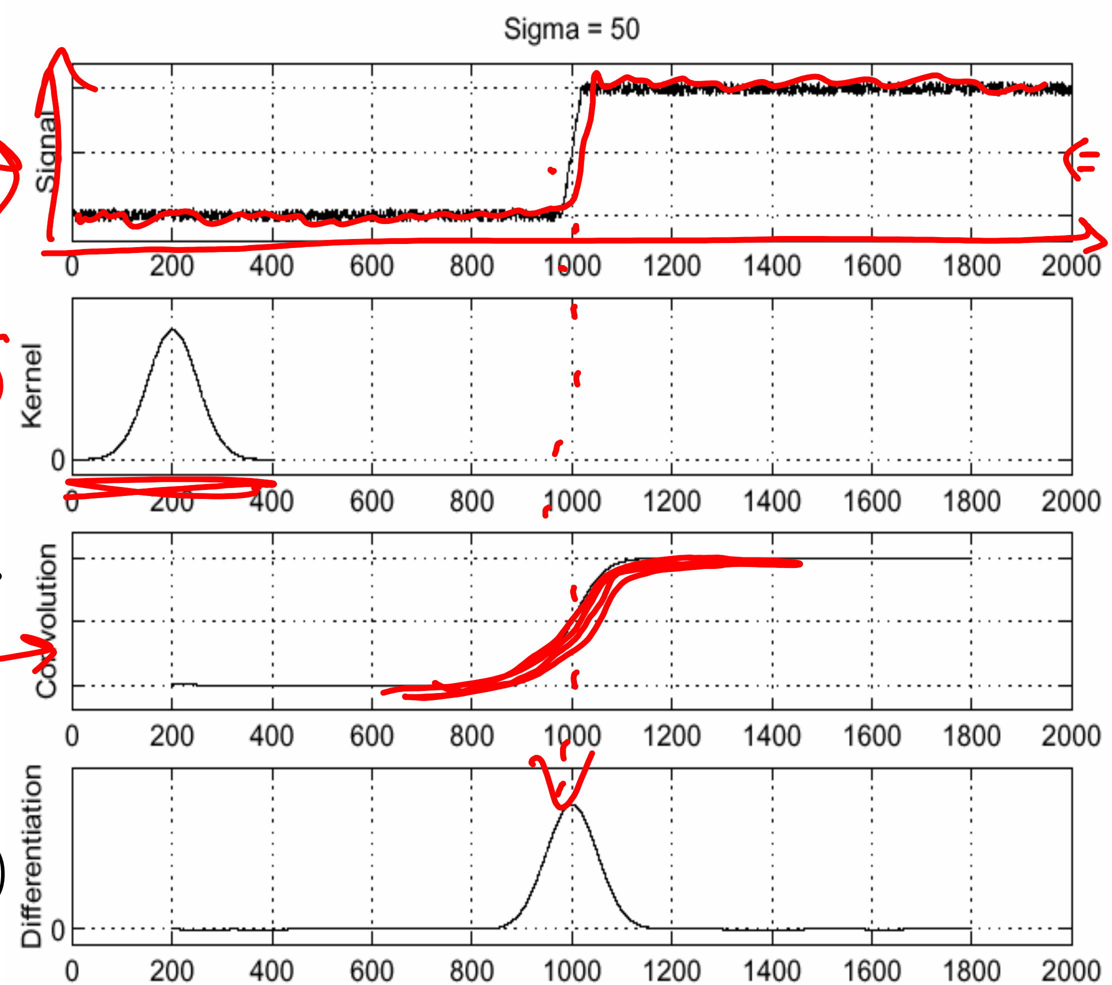
- Edges…

f

Gaussian Kernel g

$g \otimes f$

$\frac{d}{dx}(g \otimes f)$

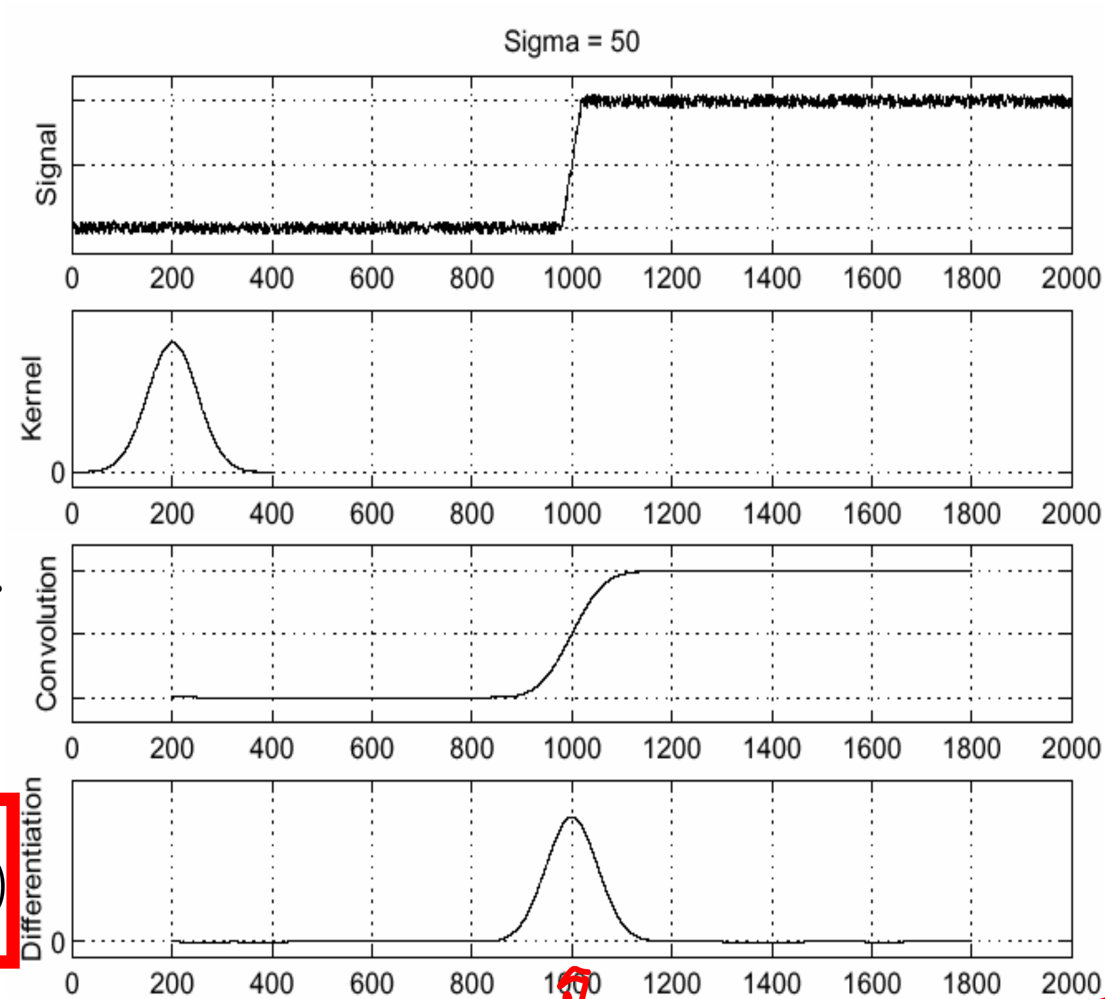


$\Leftarrow I * G(\sigma)$

Source: S. Seitz.

Scale-Invariant Feature Detection Example

- Edges…



f

$\frac{d}{dx}$ g

$g \otimes f$

$$\boxed{\frac{d}{dx}(g \otimes f)}$$

$$= \frac{dg}{dx} \otimes f$$ Derivative of a Gaussian filter

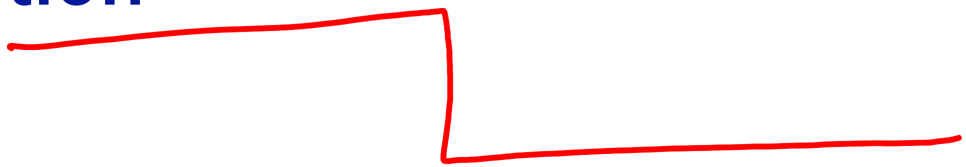
Difference

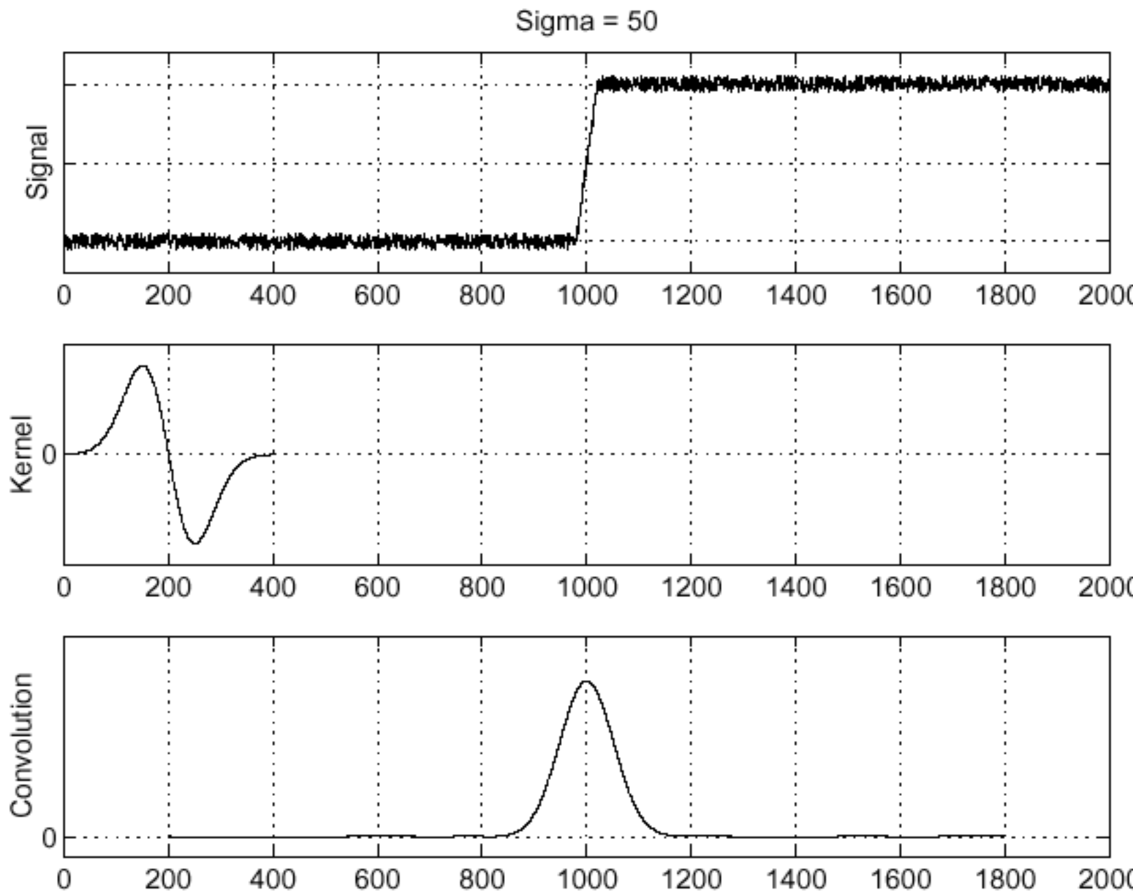
DoG

dog

Source: S. Seitz.

Edge detection





f

Edge

dog filter $\frac{d}{dx}g$

Derivative of Gaussian

$\frac{d}{dx}g \otimes f$

Edge = maximum of derivative

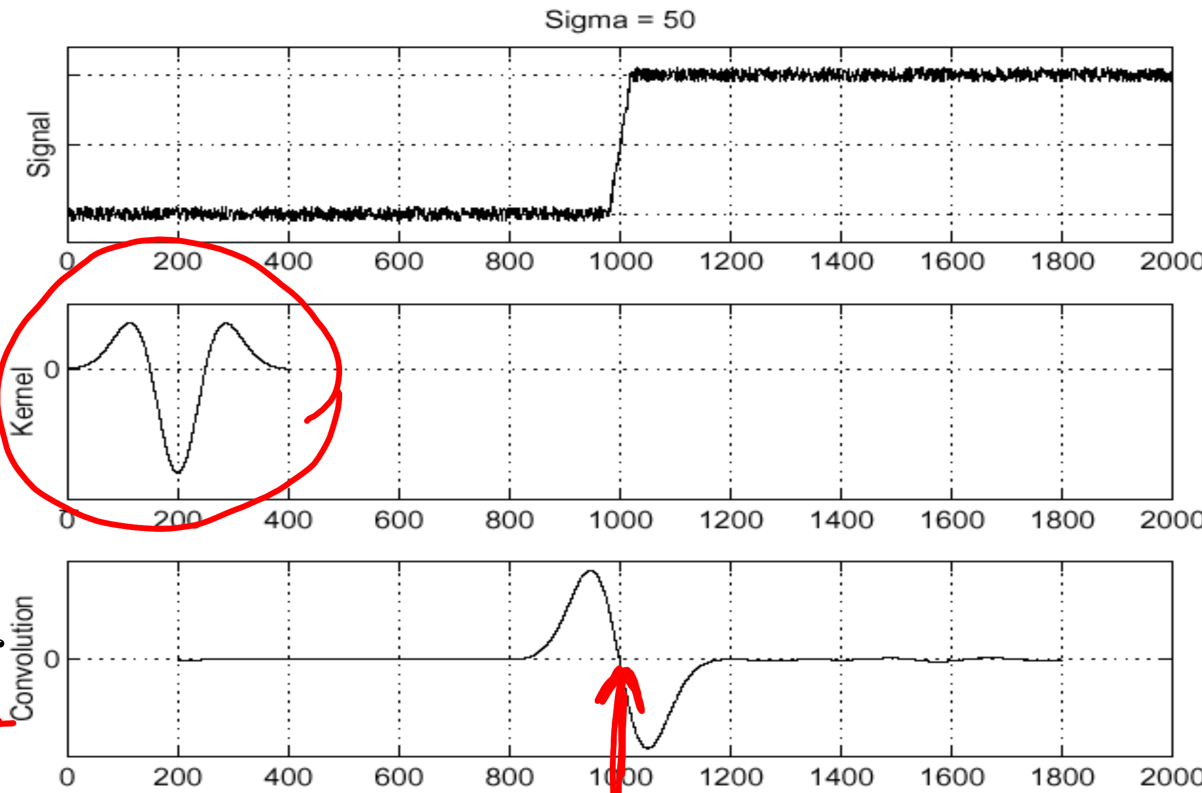
Source: S. Seitz.

Edge detection as zero crossing

f

$\frac{d^2}{dx^2} g$

$\frac{d^2}{dx^2} g \otimes f$



Edge

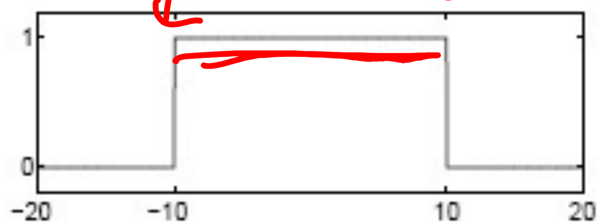
Second derivative of Gaussian

Edge = zero crossing of second derivative

Source: S. Seitz.

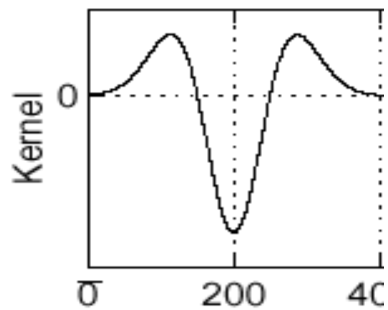
Edge detection as zero crossing

$f =$



- has two step edges
- ridge

$*$ - convolution

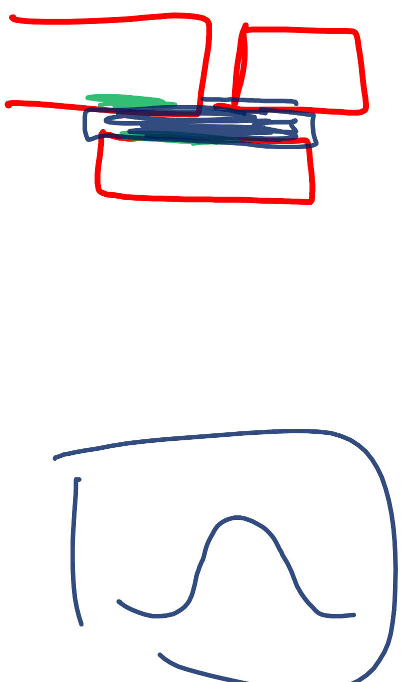


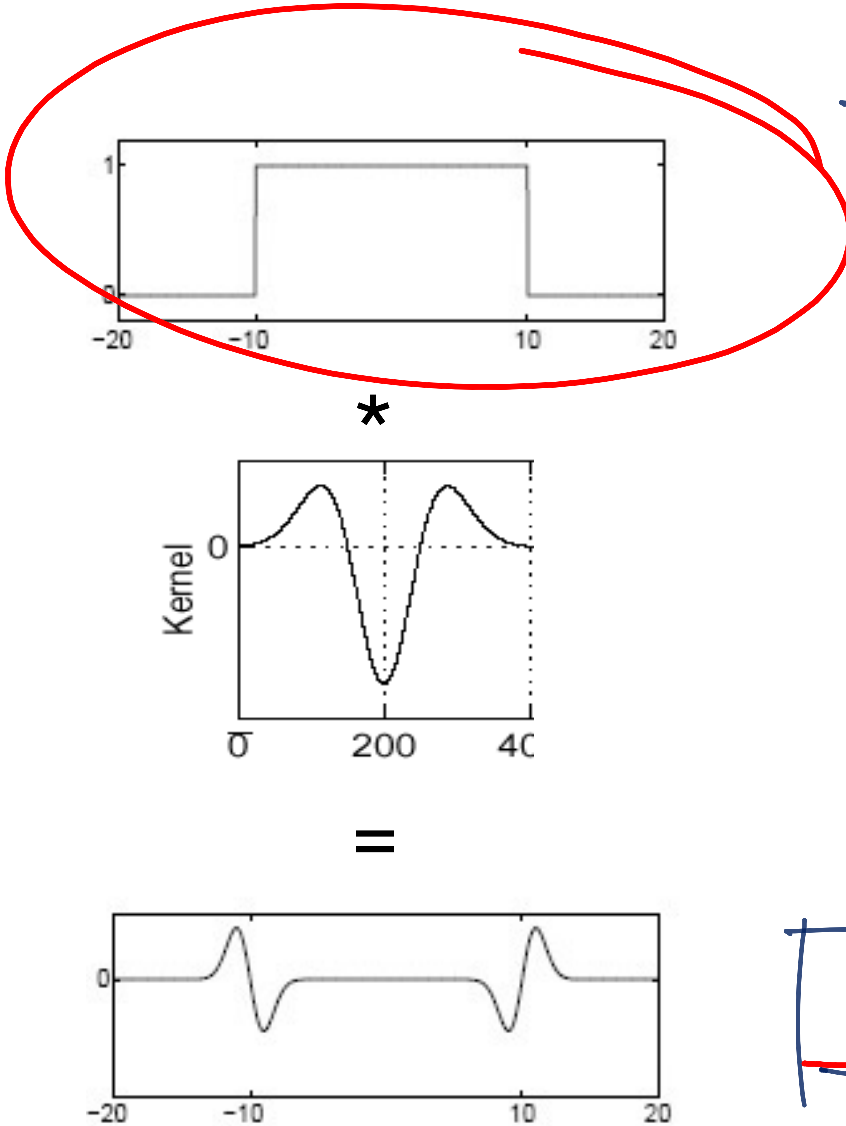
$\frac{d^2}{dx^2} g$

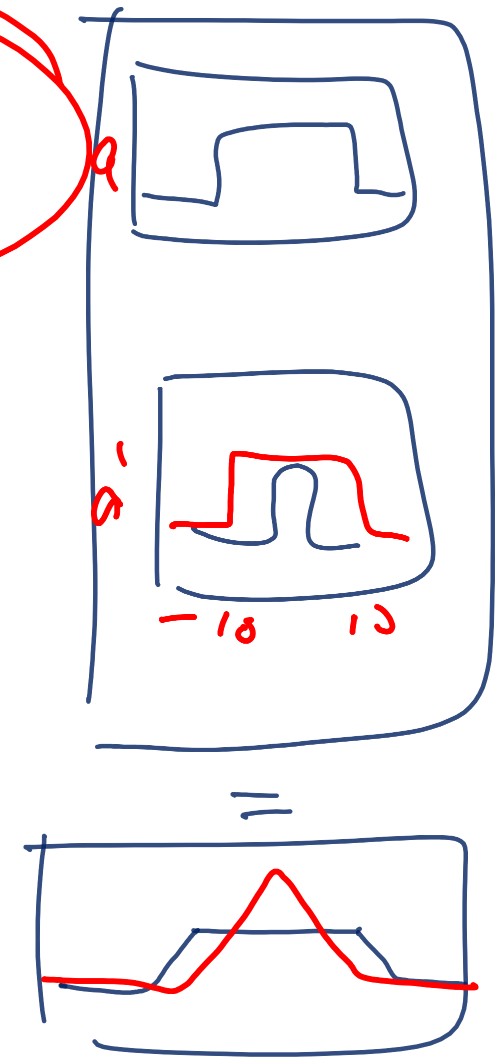
=

0

Edge detection as zero crossing

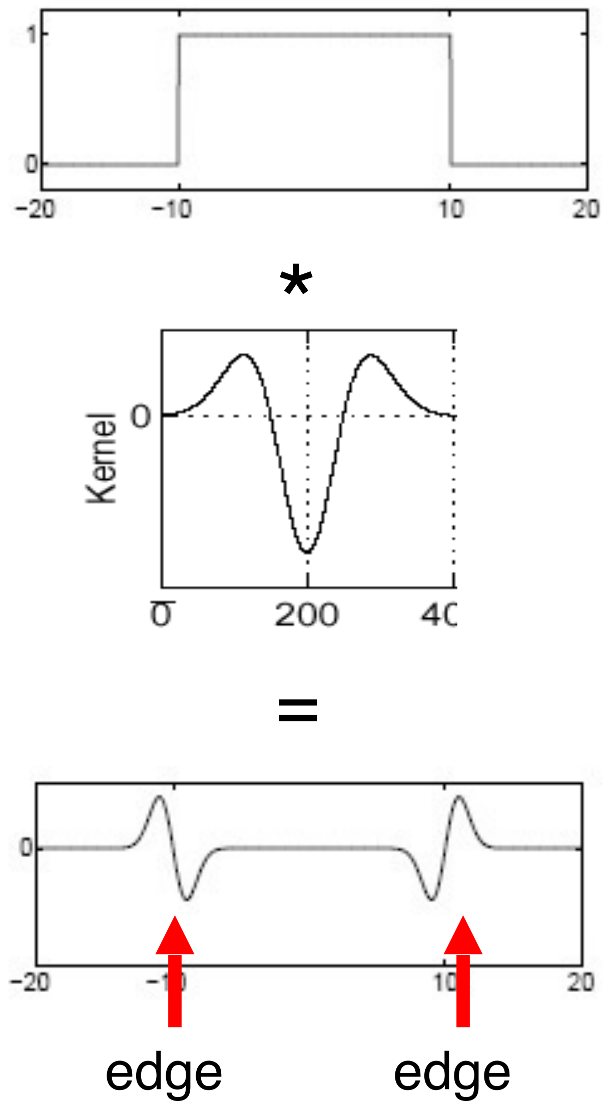






Source: S. Seitz.

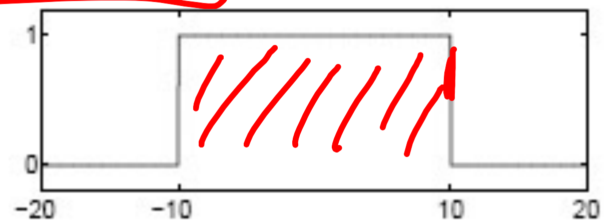
Edge detection as zero crossing



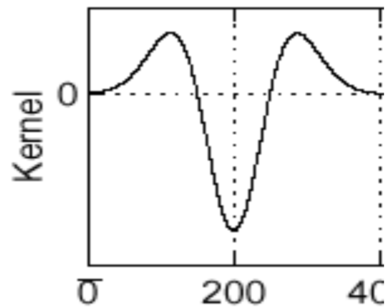
Source: S. Seitz.

Edge detection as zero crossing

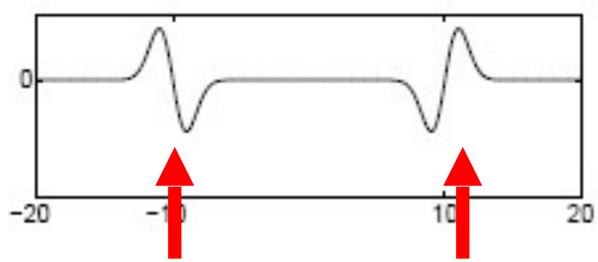
Ridge? Blob? Two Step Edges?



*



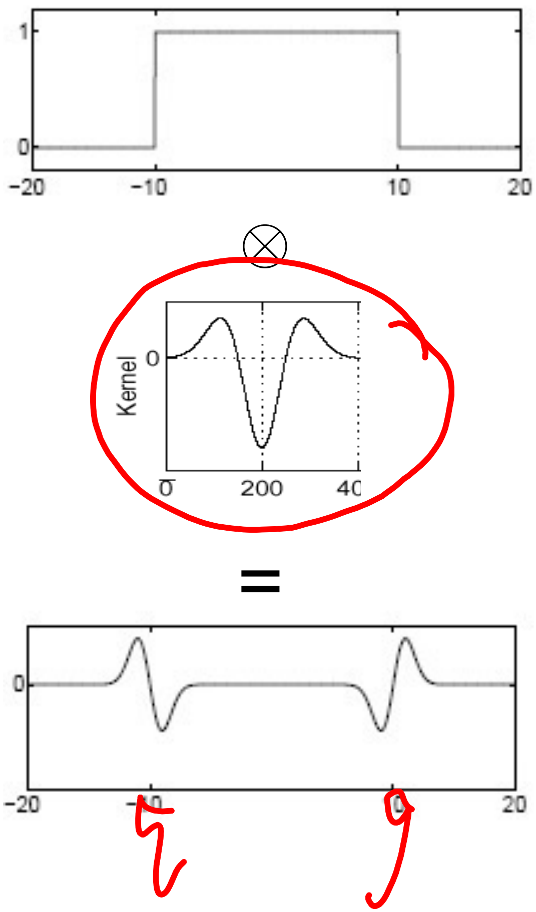
=



Source: S. Seitz.

From edges to blobs

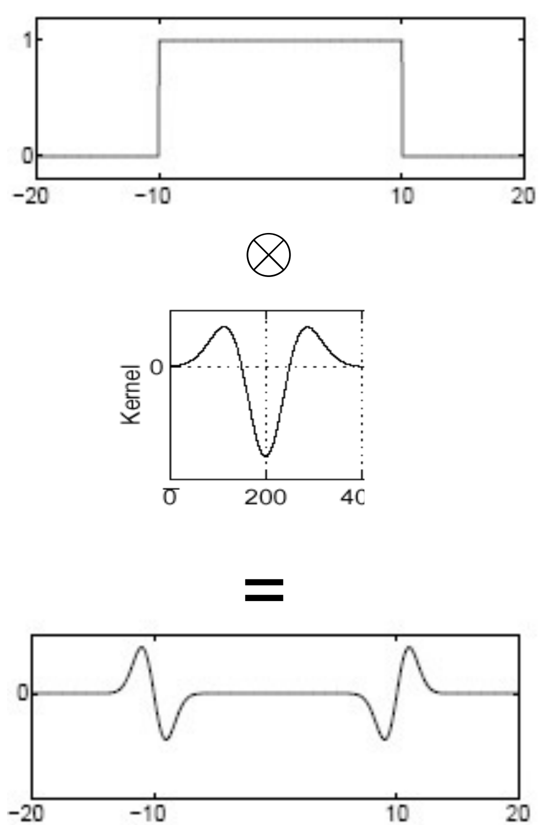
- Blob = superposition of nearby edges

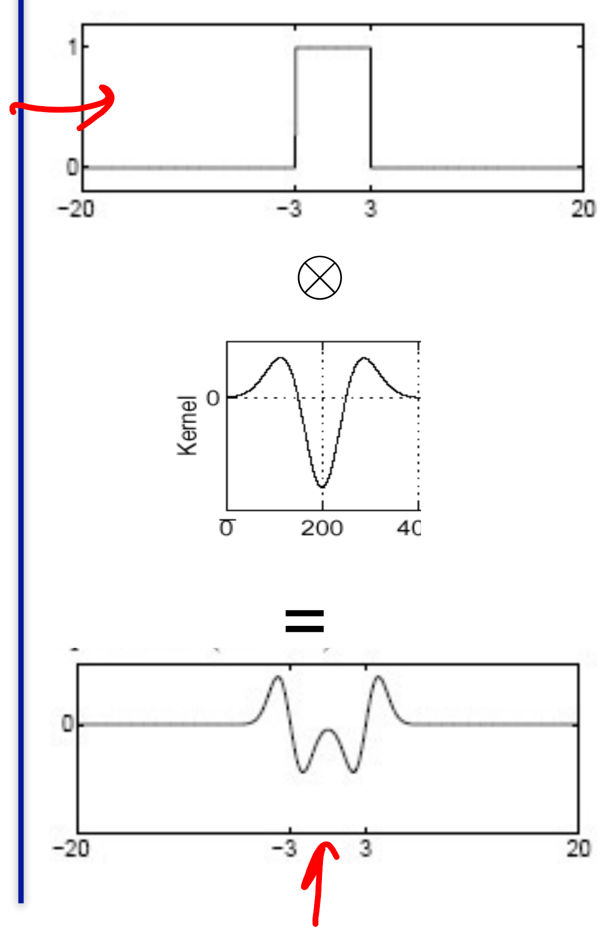


Source: S. Seitz.

From edges to blobs

- Blob = superposition of nearby edges

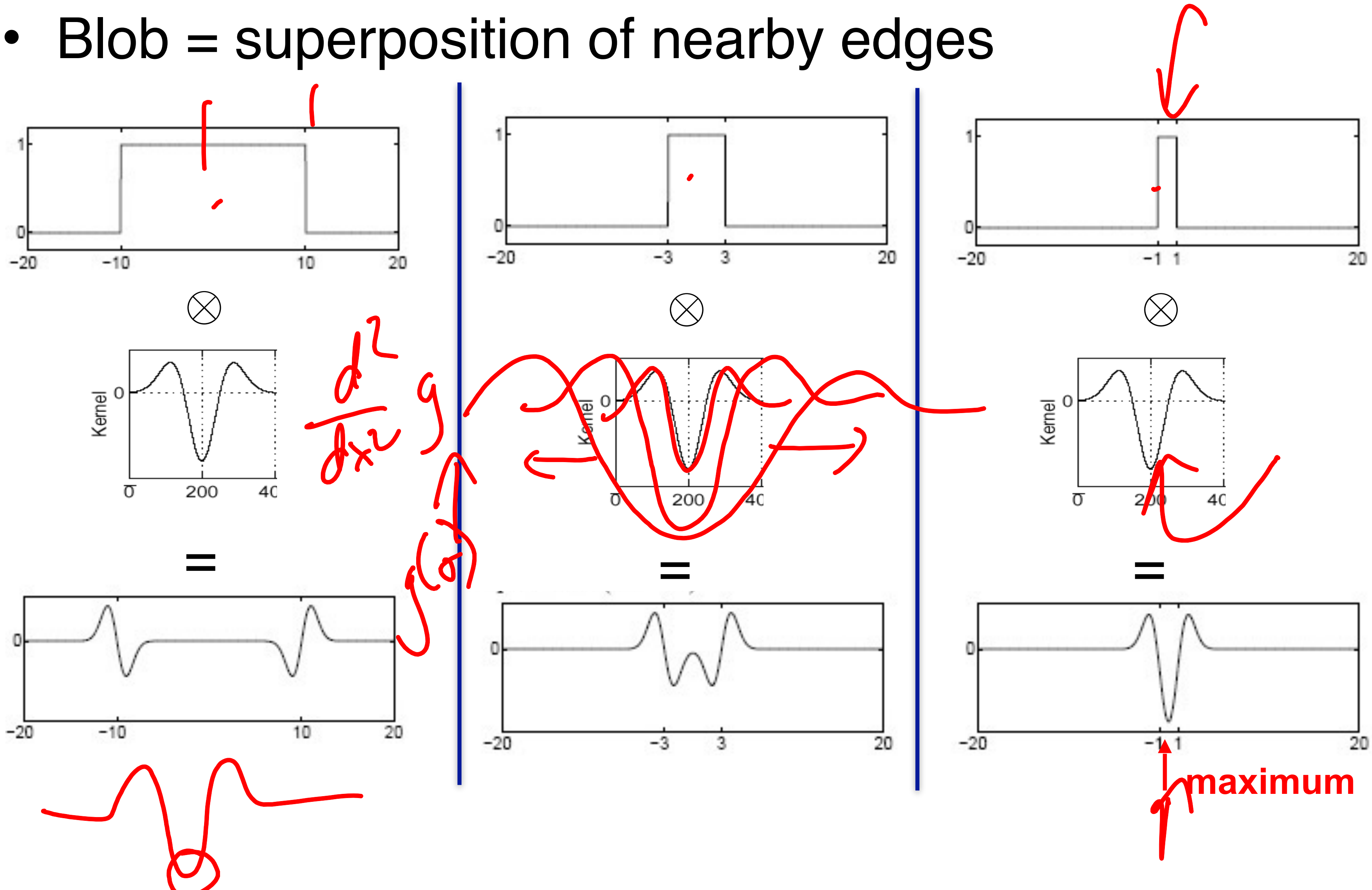




Source: S. Seitz.

From edges to blobs

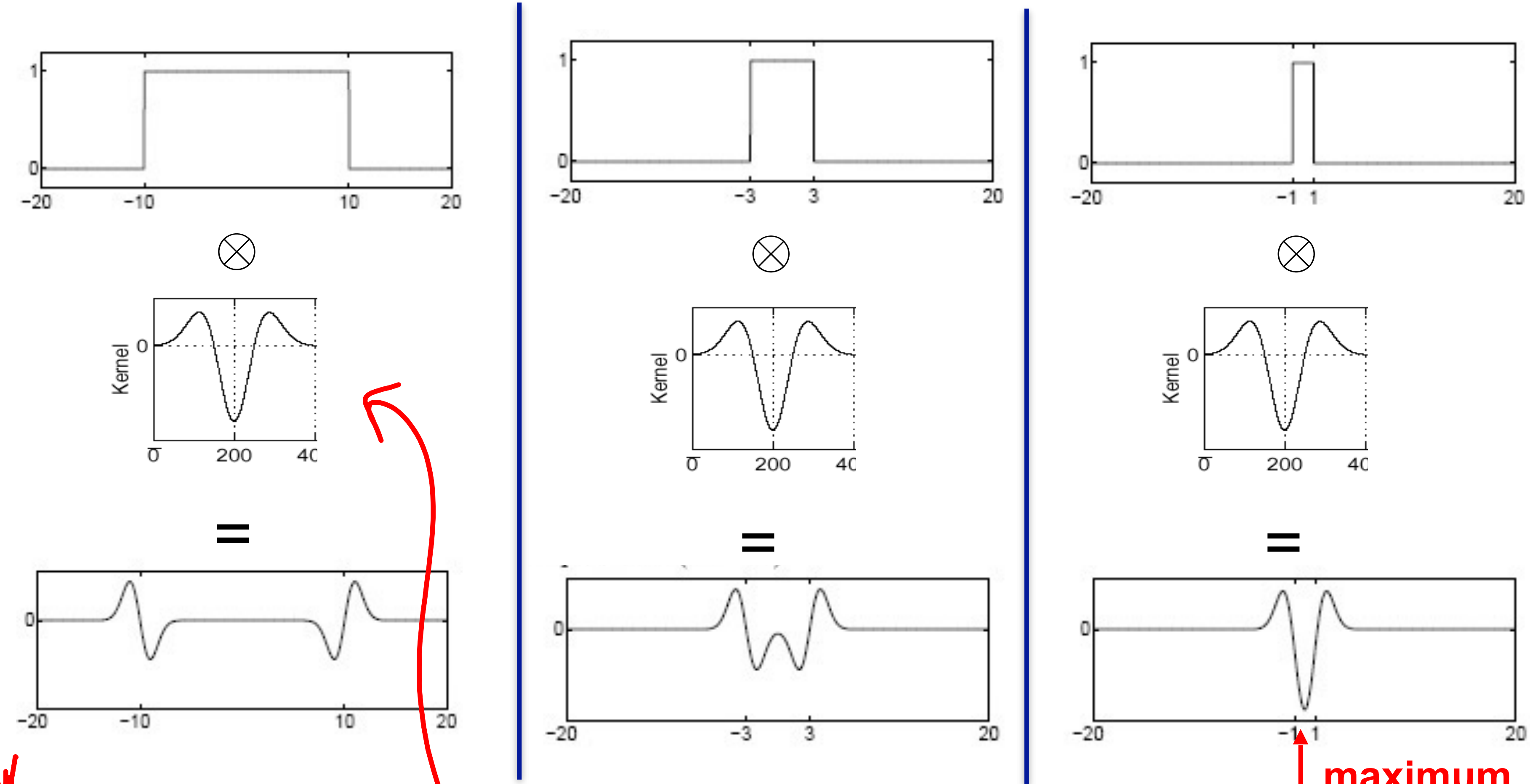
- Blob = superposition of nearby edges



Source: S. Seitz.

From edges to blobs

- Blob = superposition of nearby edges

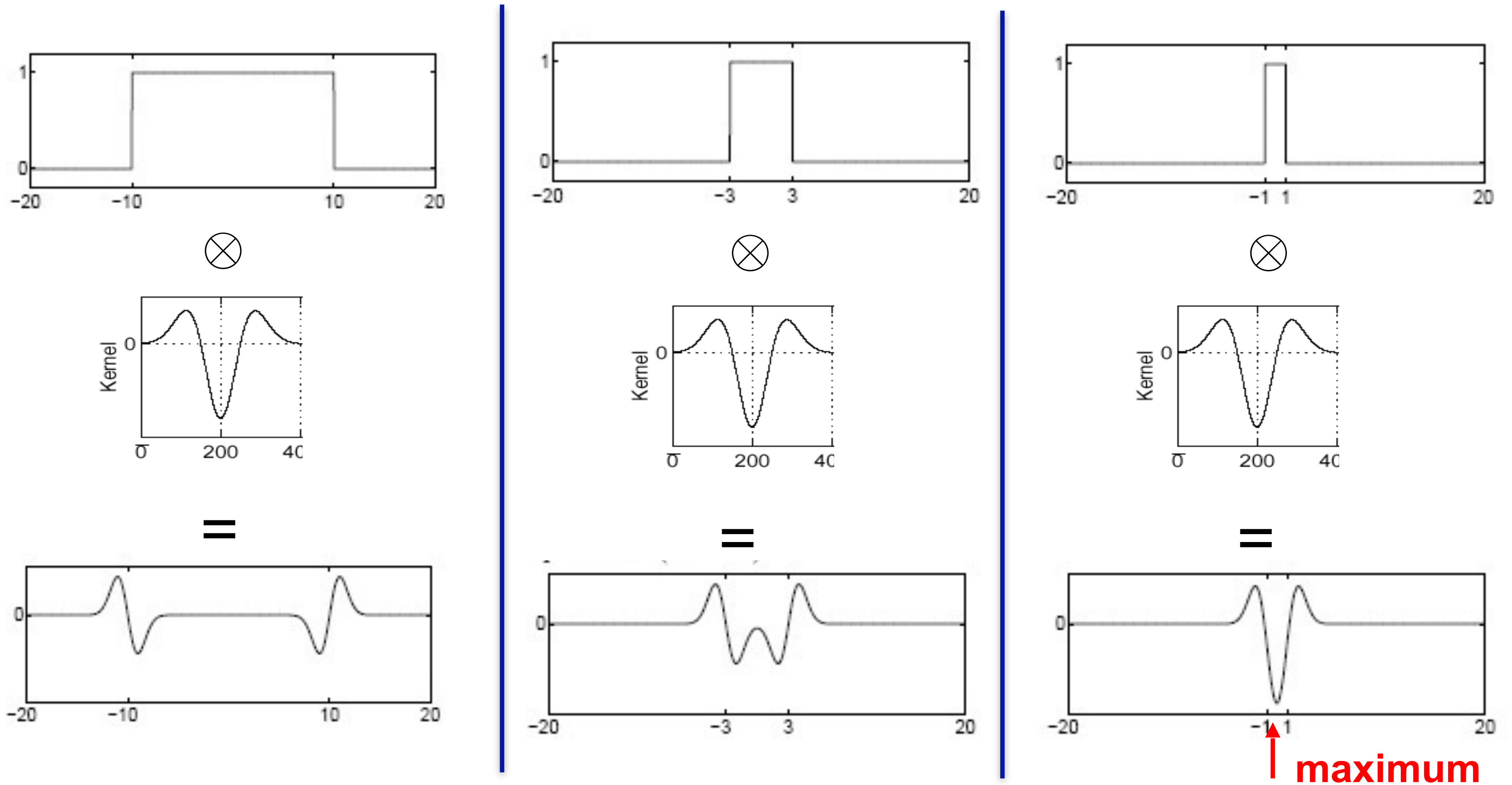


Magnitude of the Laplacian response achieves a maximum at the center of the blob, provided **the scale of the Laplacian is "matched" to the scale of the blob**

Source: S. Seitz.

From edges to blobs

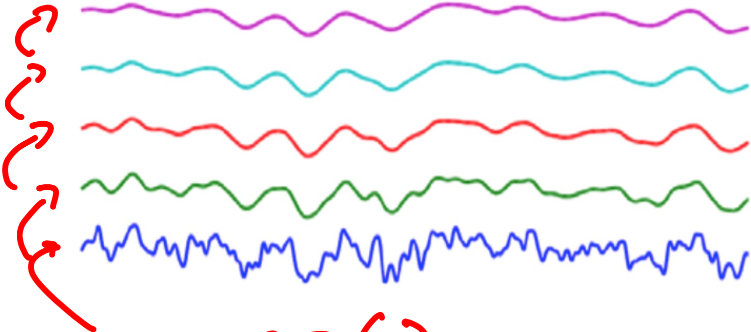
- Blob = superposition of nearby edges



What if the blob is slightly thicker or slimmer?

Source: S. Seitz.

Scale-Space Embedding



$G_s[\sigma](I)$

Source of image?

Toward Gaussian Scale-Space

Multi-scale representation comprising a continuous scale parameter and preservation of the same spatial sampling at all scales.
Following Witkin(1983) and Lindeberg(1994):

> *Embedding of original signal into a one-parameter family of derived signals constructed by convolution with a one-parameter family of Gaussian kernels of increasing width. Gaussian kernel is a unique choice.*

→ pyramids

Toward Gaussian Scale-Space

NOT A LAPLACIAN

Let $f : \mathbb{R}^N \to \mathbb{R}$ represent any given signal, e.g., our image.
Then, the scale-space representation $L : \mathbb{R}^N \times \mathbb{R}_+ \to \mathbb{R}$ is defined by:

$\mathbb{R}^1 \to \mathbb{R}$

$$L(\,\cdot\,; 0) = f$$
$$L(\,\cdot\,; t) = \underline{g(\,\cdot\,; t)} \otimes f$$

where $t \in \mathbb{R}_+$ is the scale parameter and $g : \mathbb{R}^N \times \mathbb{R}_+ \setminus \{0\} \to \mathbb{R}$ is the Gaussian kernel of dimension N:

$$g(\mathbf{x}; t) = \frac{1}{(2\pi t)^{\frac{N}{2}}} \exp\left[-\frac{\mathbf{x}^T\mathbf{x}}{2t}\right],$$

where $\mathbf{x} \in \mathbb{R}^N$ and $\sigma = \sqrt{t}$, standard deviation of the Gaussian.
As t increases, the signal becomes gradually smoother. (See earlier figure)

Toward Gaussian Scale-Space

Multiscale Spatial Derivatives

Multiscale spatial derivatives can be defined by:

$$L_{x^n}(\,\cdot\,;t) = \partial_{x^n} L(\,\cdot\,;t) = g_{x^n}(\,\cdot\,;t) \otimes f \qquad (*)$$

When g_{x^n} denotes a (possibly mixed) Gaussian derivative of some order $n = (n_1, \ldots, n_N)^T \in \mathbb{Z}^N,\ each\ n_i \in \mathbb{Z}\ and\ \partial_{x^n} = \partial_{x_1^n}\partial_{x_2^n}\ldots\partial_{x_N^n}\ where\ x = (x_1, \ldots, x_N)^T \in \mathbb{R}^N\ and\ x_i \in \mathbb{R}.$

Note that (*) creates a way of computing derivatives of f to any order even though f may not be differentiable of any order.

Toward Gaussian Scale-Space

Discussion

as t increases

Scale-space leads to suppressions at fine scale structures. Intuitively, when convolving a signal by a Gaussian kernel with standard deviation $\sigma = \sqrt{t}$, this leads to suppressing most of the structures in the signal with characteristic length less than σ. (not an exact property)

cannot

1. New local extrema ~~const~~ be created with increasing scale in 1D
2. The property that new local extrema can be created with increasing scale is inherit in two and higher dimensions.
3. Gaussian kernel is the unique kernel for generating the scale space (Koendeink 1994).
   (SSR must satisfy $\partial_t L = \frac{1}{2}\nabla^2 L$ and, since, Gaussian is the Green's function of the diffusion equation at an infinite domain.) *Beyond scope for this course; you are not responsible for this point.*
4. Have semigroup property: $h(\,\cdot\,;t_1) \otimes h(\,\cdot\,;t_2) = (\,\cdot\,;t_1 + t_2)$

Normalization in Gaussian Scale-Space

Amplitude of scale-space spatial derivatives decreases with scale due to non-enhancement property of local extrema.

Normalize by multiplying by $\sqrt{t}$ or σ of Gaussian:

$$\hat{\partial} = \sqrt{t}\partial$$

See other slides for discussion on this scale normalization.

Lindeberg's Scale Selection Principle

In the absence of other evidence, assume that a scale level, at which some (possibly non-linear) combination of normalized derivatives, assumes a local maximum over scales, can be treated as reflecting a characteristic length of a corresponding structure in the data.

Not Resp. in 442

Scale Invariance of Local Scale-Space Maxima

Let $f(x) = f'(sx)$ and $L(\,\cdot\,;t) = g(\,\cdot\,;t) \otimes f,\ L'(\,\cdot\,;t) = g(\,\cdot\,;t') \otimes f'$

$$x' = sx$$

$$t' = s^2 t$$

$\Rightarrow$ if input image is rescaled by factors, then scale of local maxima assumed to be multiplied by same factor. (measured in units of $\sigma = \sqrt{t}$).

Then

$$L(x;t) = L'(x';t')$$

and for m^{th} order derivatives:

$$\partial_{x^m} L(x;t) = s^m \partial_{x'^m} L'(x';t')$$

For $\gamma - normalized$ derivatives:

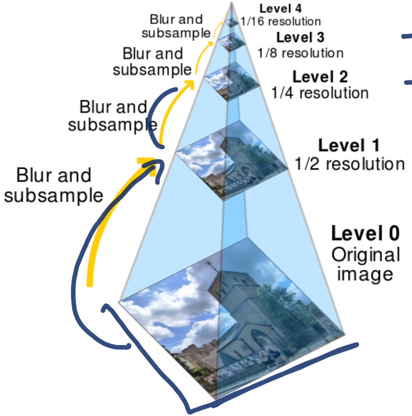
$$\partial_\zeta = t^{\gamma/2}\partial_x, \partial_{\zeta'} = t'^{\gamma/2}\partial_{x'}$$

We have:

$$\partial_{\zeta^m} L(x;t) = s^{m(1-\gamma)}\partial_{\zeta'^m} L'(x';t')$$

When $\gamma = 1$, we have perfect scale invariance

Early Scale-Spaces: Pyramids



Source of image?

- Gaussian Pyramids
- Laplacian Pyramids

Advantages

- lead to rapidly decreasing images size
- memory requirements are small

Disadvantages

- coarse quantization of scale
- pyramids are not translationally invariant

More examples on pyramids later.

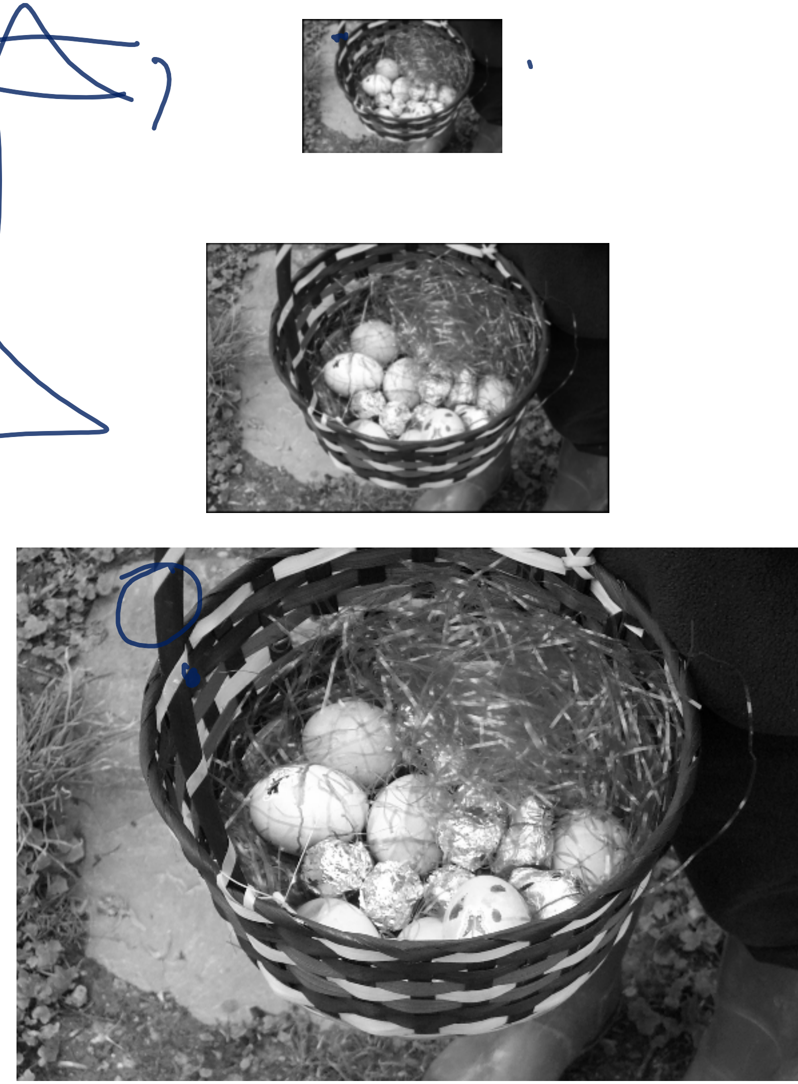
Filter Pyramids

- Recall we can always filter with $g(\sigma)$ for any σ
- As a result, we can think of a continuum of filtered images as σ grows.
  - This is referred to as the “scale space” of the images. We will see this show up several times.
- As a related note, suppose I want to subsample images
  - Subsampling reduces the highest frequencies
  - Averaging reduces noise
  - Pyramids are a way of doing both

Source: G Hager Slides

Gaussian Pyramid

- Algorithm:
  - 1. Filter with $\mathcal{G}(\sigma = 1)$
  - 2. Resample at every other pixel
  - 3. Repeat



Source: G Hager Slides

Laplacian Pyramid Algorithm

- Create a Gaussian pyramid by successive smoothing with a Gaussian and down sampling

- Set the coarsest layer of the Laplacian pyramid to be the coarsest layer of the Gaussian pyramid

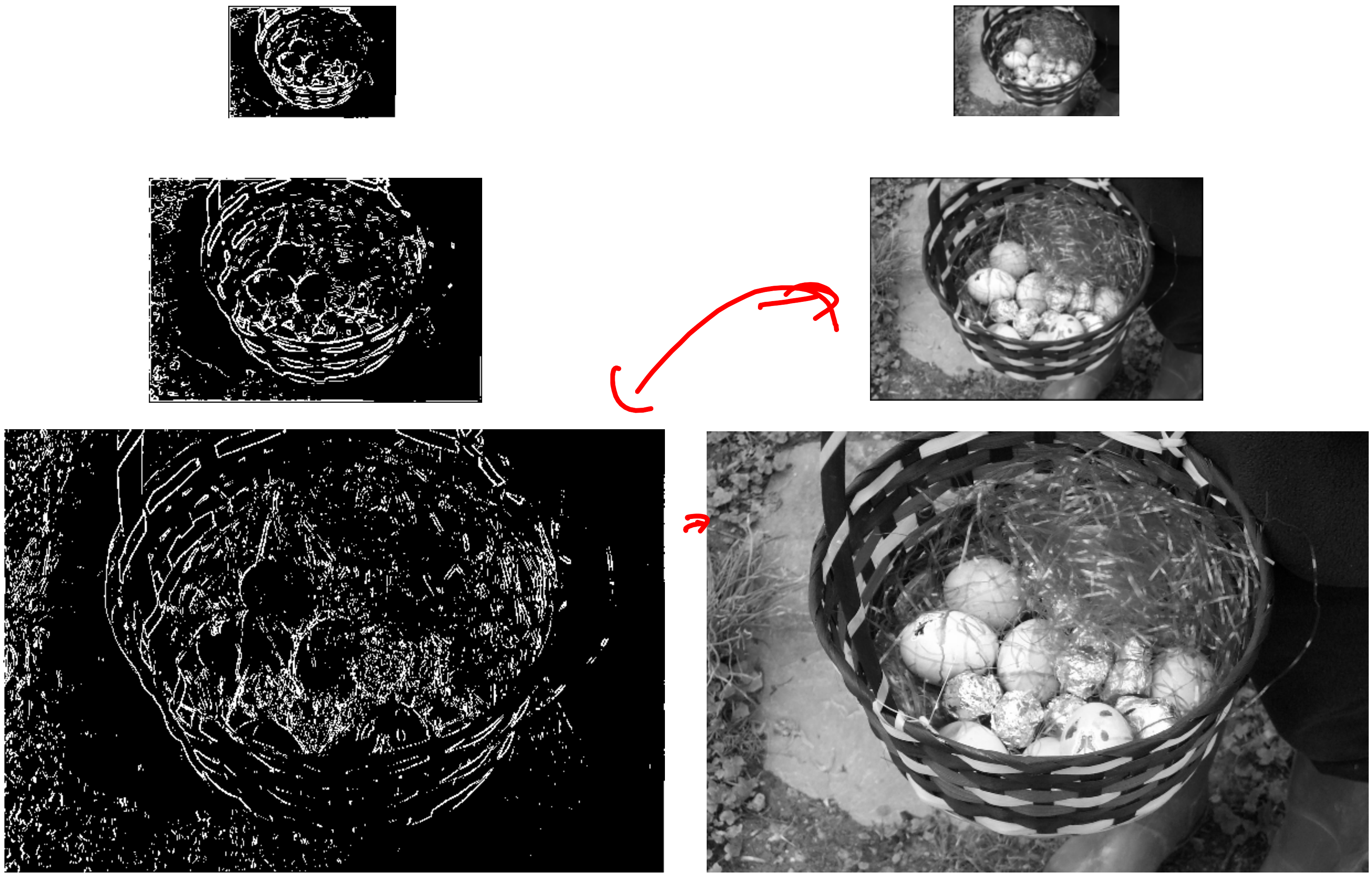
- For each subsequent layer n+1, compute

minus

$$L(n+1) = \underline{G(n+1)} \overset{-}{\cancel{=}} \underline{\text{Upsample}(G(n))}$$

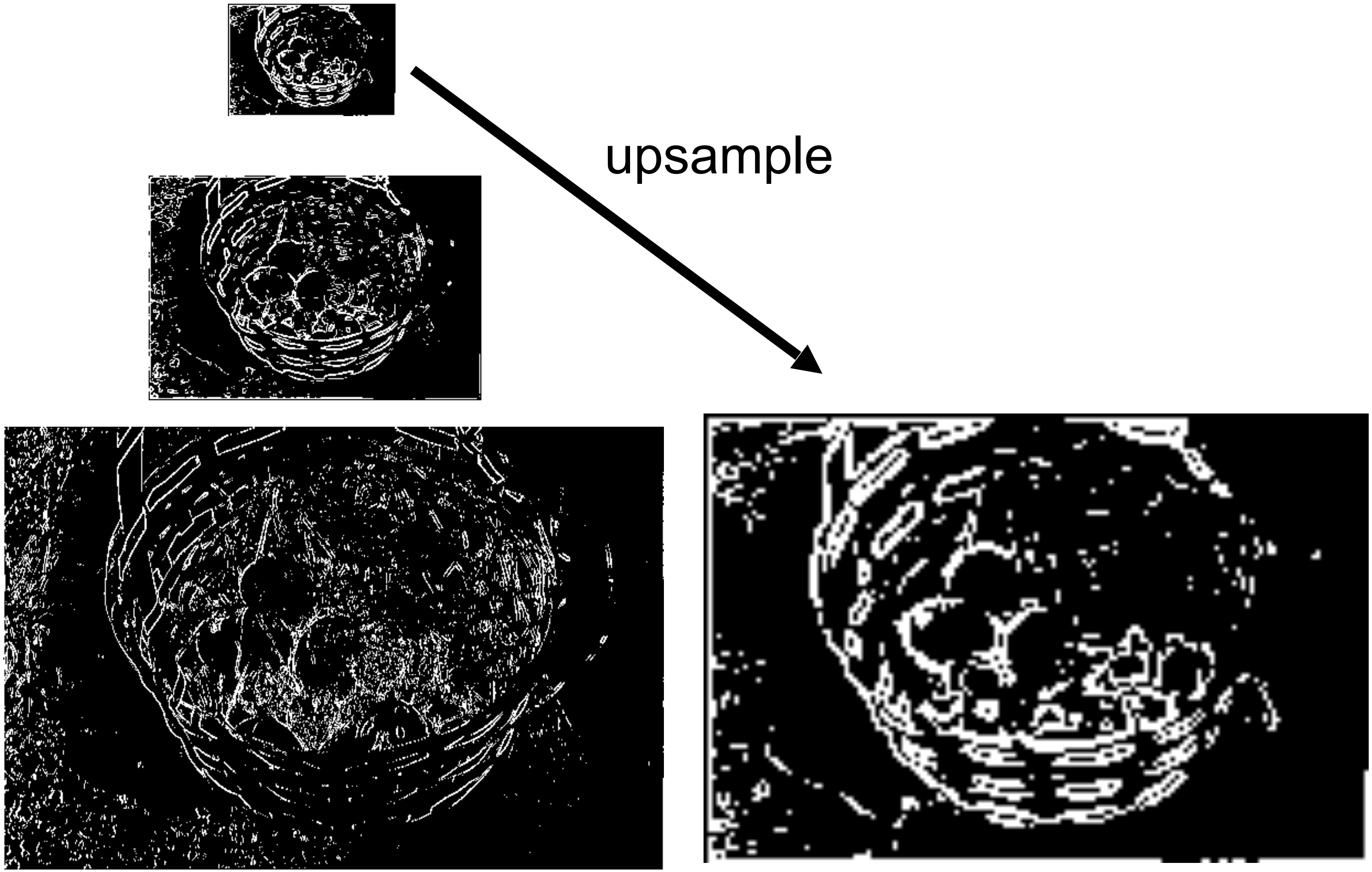
Source: G Hager Slides

Laplacian of Gaussian Pyramid



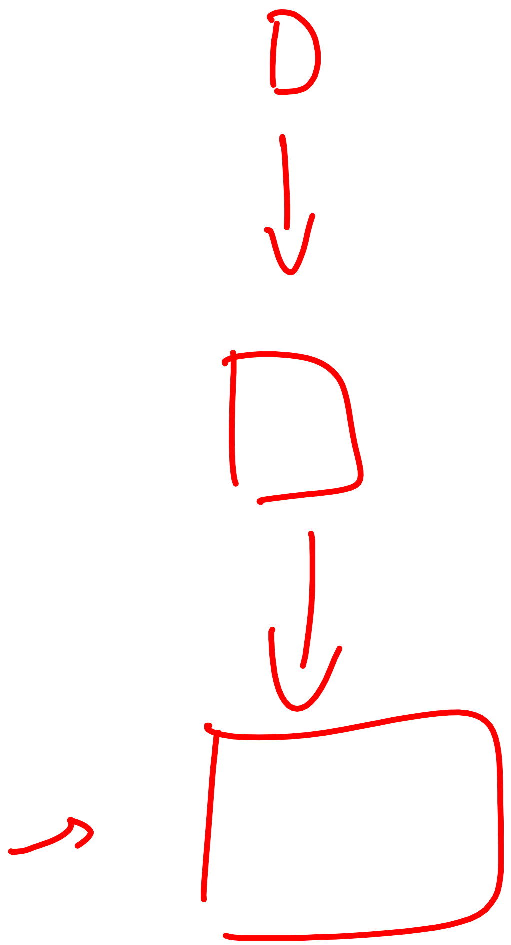
Source: G Hager Slides

Laplacian of Gaussian Pyramid



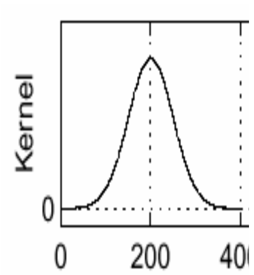
Source: G Hager Slides

Stop Slides



Scale selection

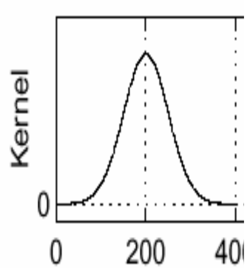
- We want to find the characteristic scale of the blob by convolving it with Laplacians at several scales and looking for the maximum response



Source: S. Seitz.

Scale selection

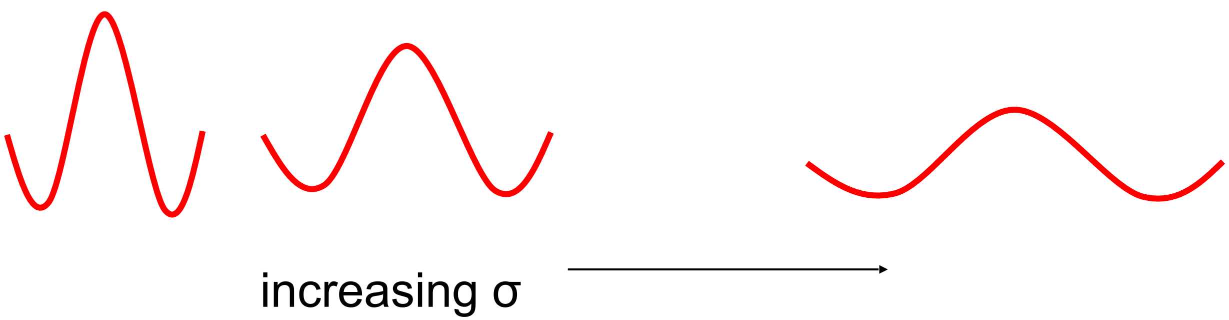
- We want to find the characteristic scale of the blob by convolving it with Laplacians at several scales and looking for the maximum response

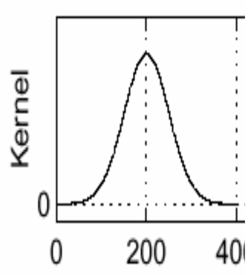


Source: S. Seitz.

Scale selection

- We want to find the characteristic scale of the blob by convolving it with Laplacians at several scales and looking for the maximum response

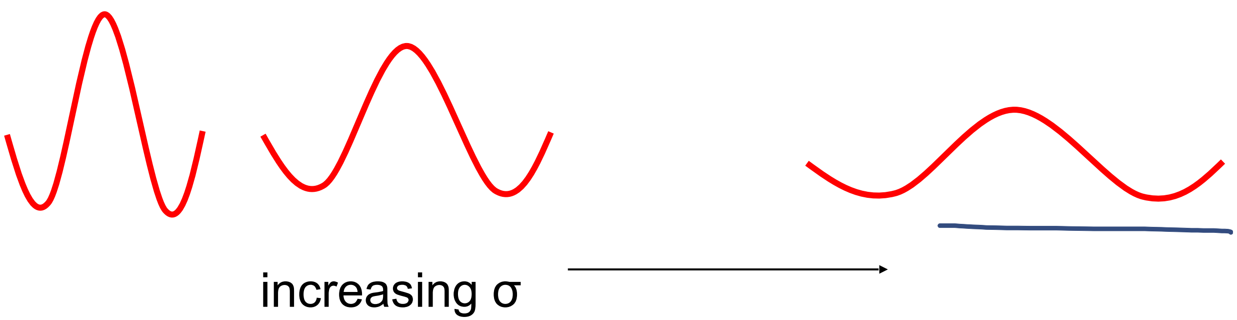


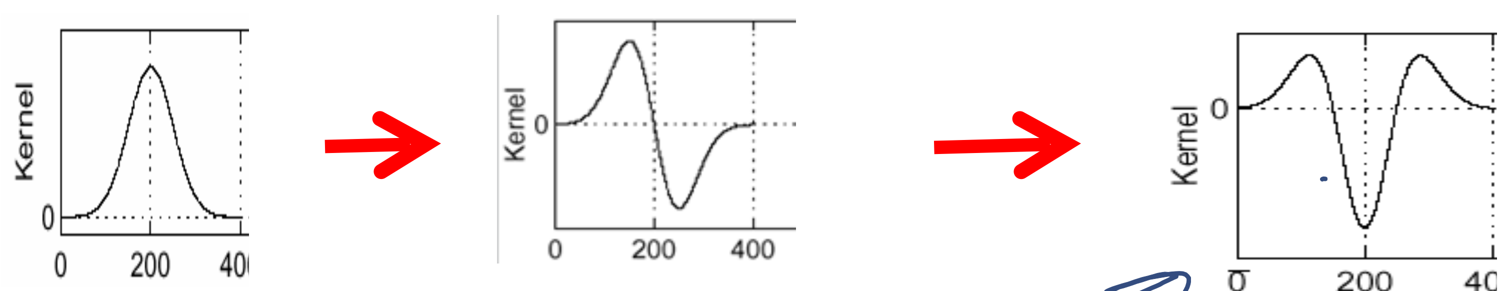


Source: S. Seitz.

Scale selection

- We want to find the characteristic scale of the blob by convolving it with Laplacians at several scales and looking for the maximum response





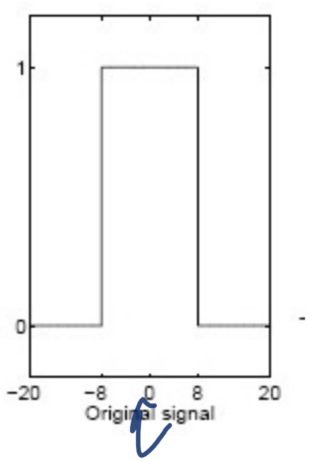
g (σ) $\qquad \frac{d}{dx}g \qquad \frac{d^2}{dx^2}g$

Source: S. Seitz.

Scale selection

- We want to find the characteristic scale of the blob by convolving it with Laplacians at several scales and looking for the maximum response

•However, Laplacian response decays as scale increases:



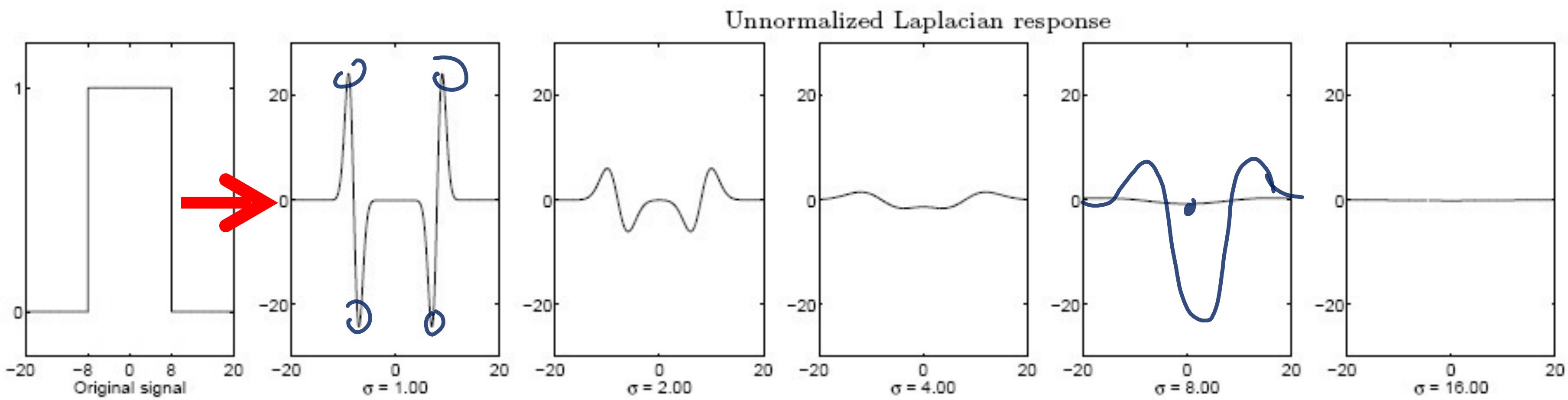
original signal (radius=8)

Source: S. Seitz.

Scale selection

- We want to find the characteristic scale of the blob by convolving it with Laplacians at several scales and looking for the maximum response

•However, Laplacian response decays as scale increases:



original signal
(radius=8)

$\sigma = 1$ 2 4 8 16

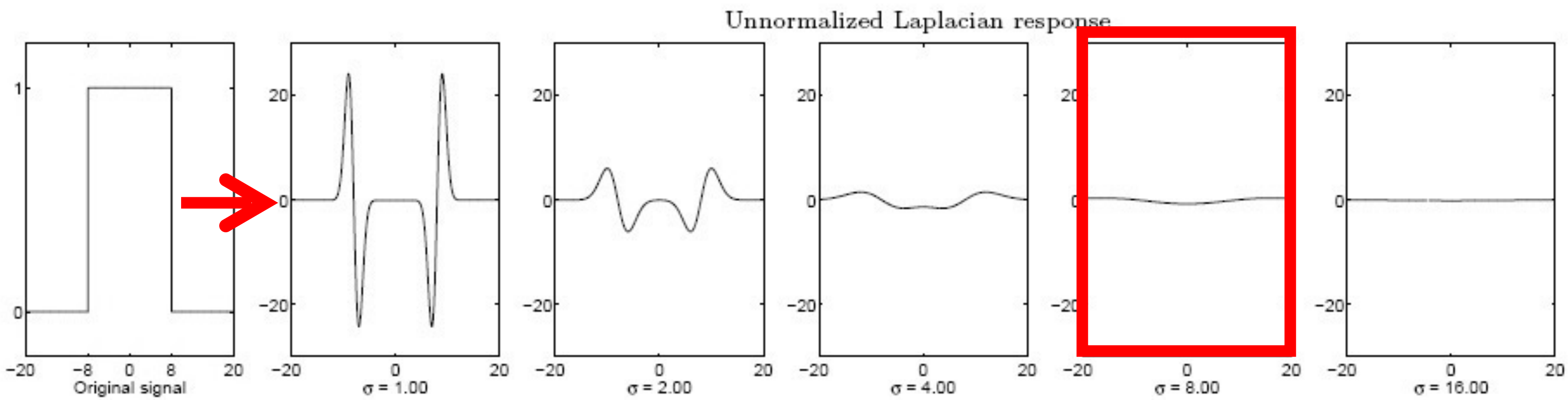
increasing σ

Source: S. Seitz.

Scale selection

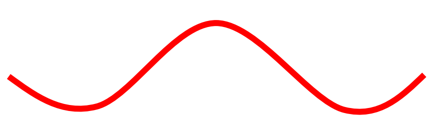
- We want to find the characteristic scale of the blob by convolving it with Laplacians at several scales and looking for the maximum response

•However, Laplacian response decays as scale increases:



original signal (radius=8)

This should give the max response ☹

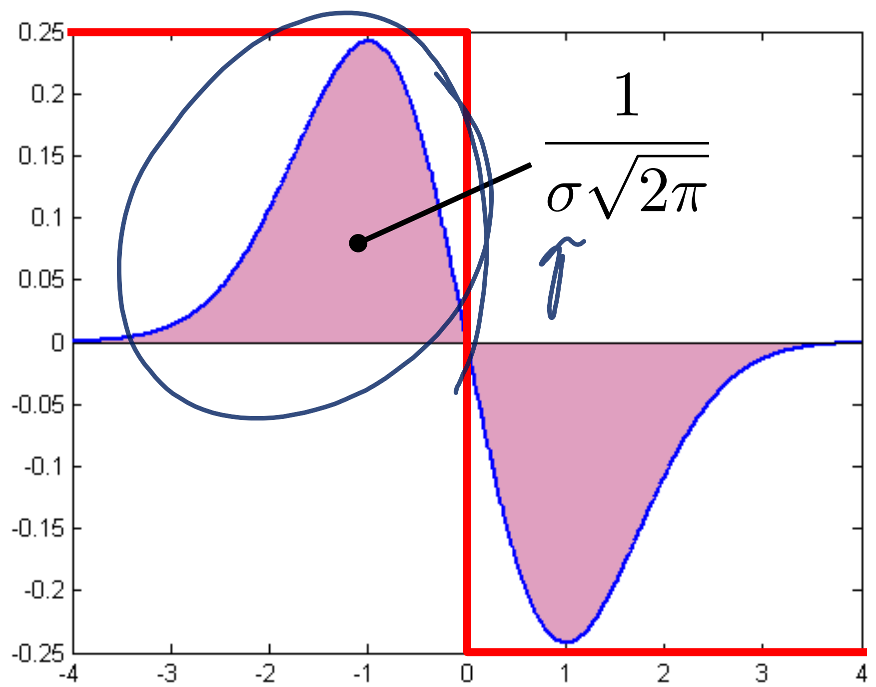


increasing σ

Source: S. Seitz.

Scale normalization

- The response of a derivative of Gaussian filter to a perfect step edge decreases as σ increases



Source: S. Seitz.

Scale normalization

- To keep response the same (scale-invariant), must multiply Gaussian derivative by σ

- Laplacian is the second Gaussian derivative, so it must be multiplied by σ^2

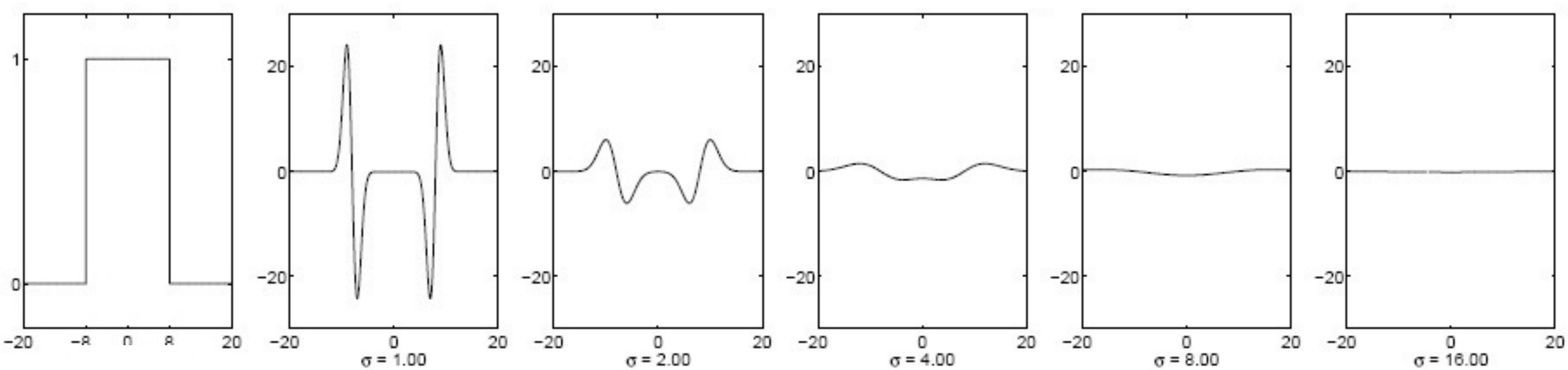
Derivation in notes.

Source: S. Seitz, S. Savarese.

Effect of scale normalization

Original signal

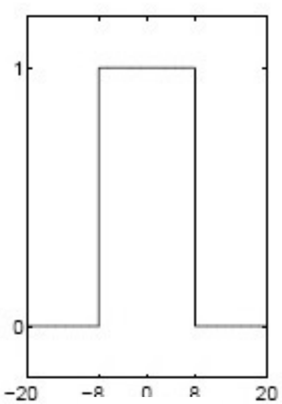
Unnormalized Laplacian response



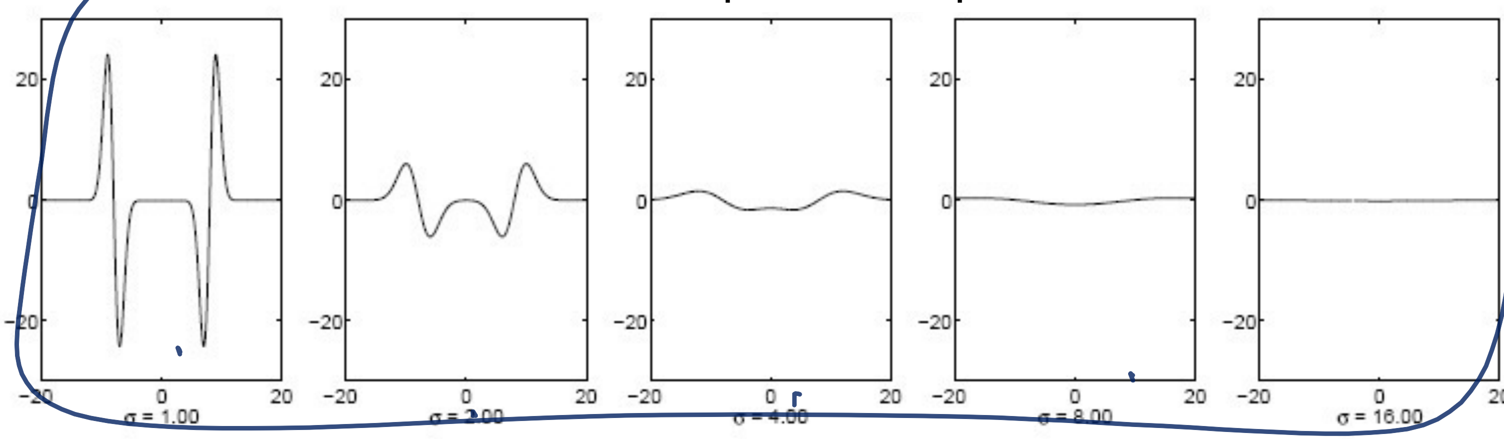
Source: S. Seitz, S. Savarese.

Effect of scale normalization

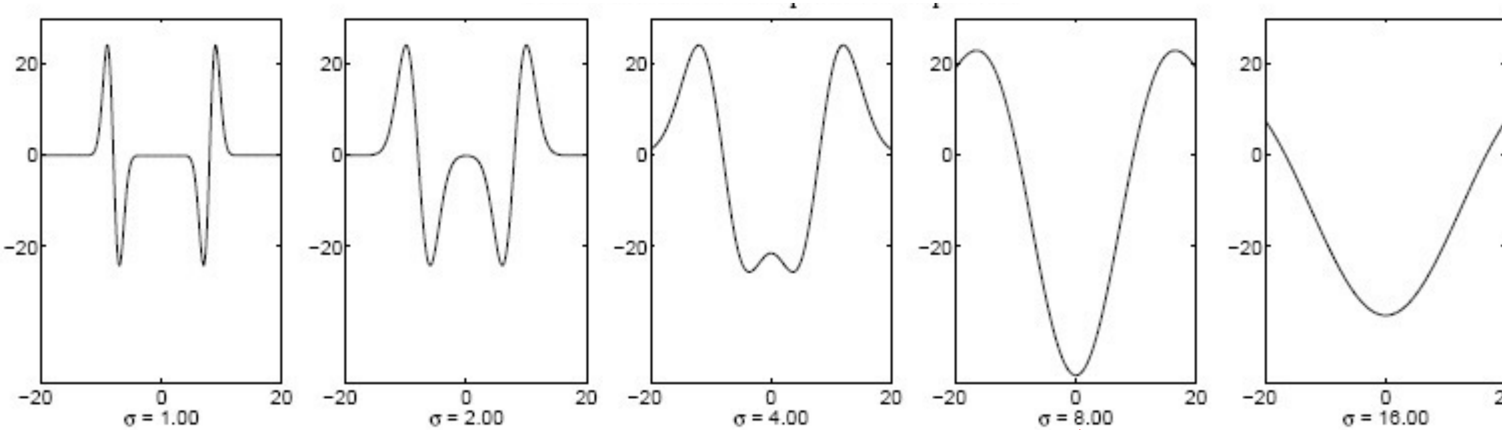
Original signal



Unnormalized Laplacian response



Scale-normalized Laplacian response

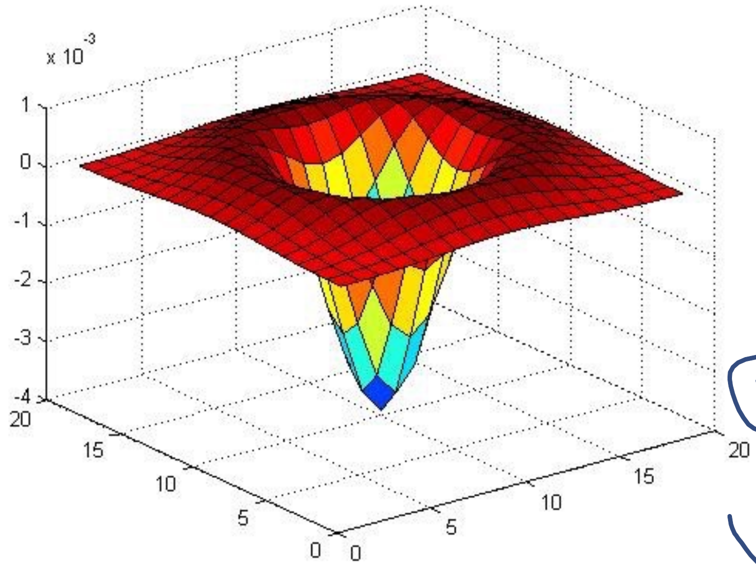


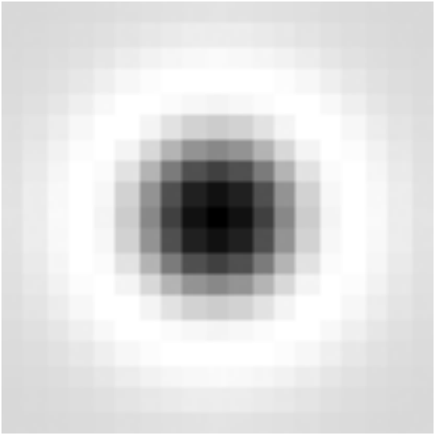
Maximum ☺️

Source: S. Seitz, S. Savarese.

Example: Blob detection in 2D

- Laplacian of Gaussian: Circularly symmetric operator for blob detection in 2D



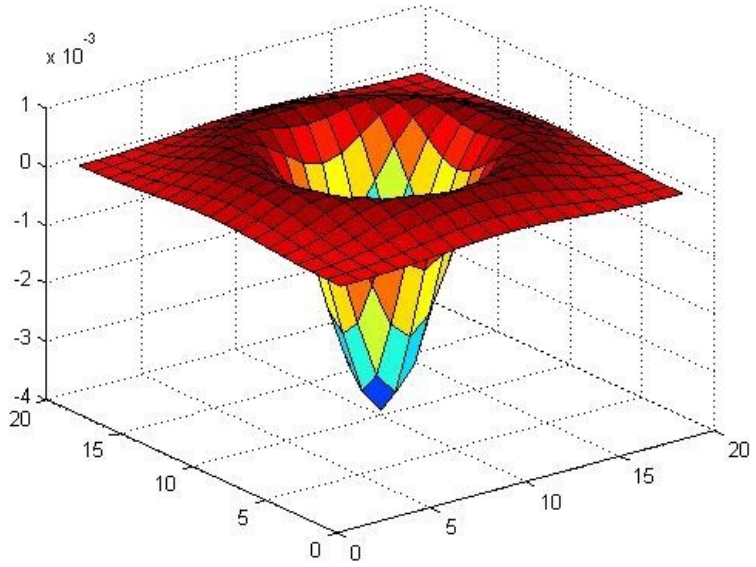


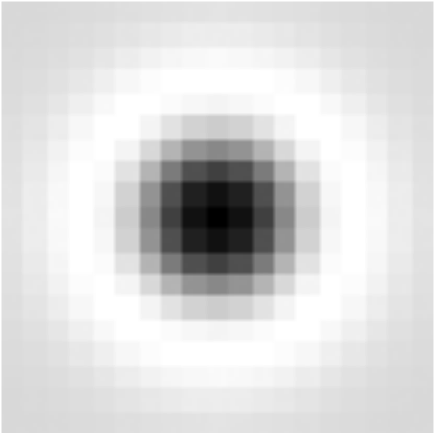
$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

Source: S. Seitz, S. Savarese.

Example: Blob detection in 2D

- Laplacian of Gaussian: Circularly symmetric operator for blob detection in 2D



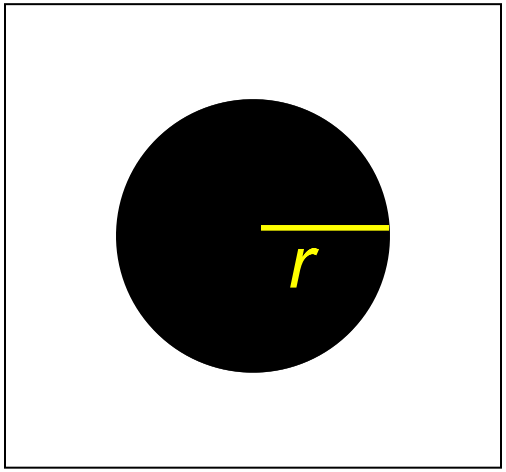


Scale-normalized: $$\nabla^2_{\text{norm}} g = \sigma^2 \left(\frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \right)$$

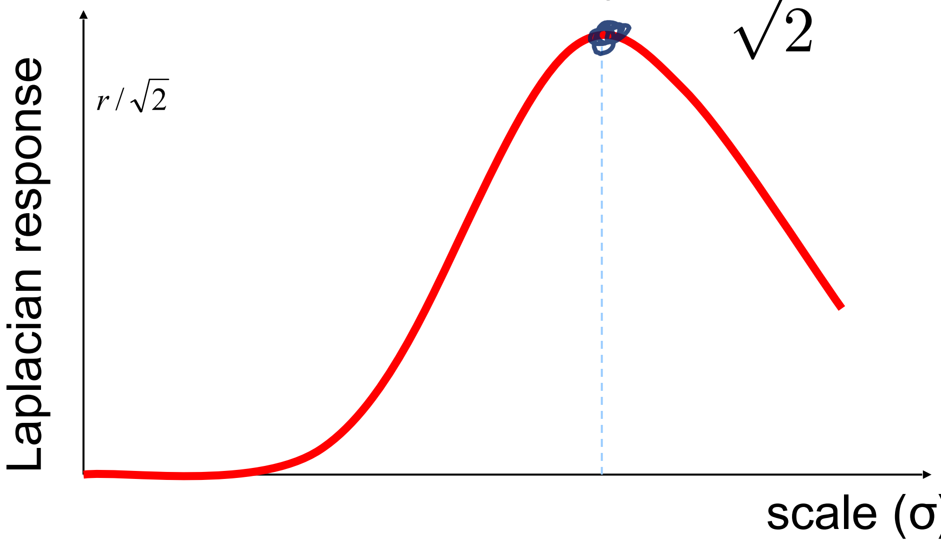
Source: S. Seitz, S. Savarese.

Scale selection

- For a binary circle of radius r, the Laplacian achieves a maximum at



image



$$\sigma = \frac{r}{\sqrt{2}}$$

Source: S. Seitz, S. Savarese.

Characteristic scale

- We define the characteristic scale as the scale that produces peak of Laplacian response

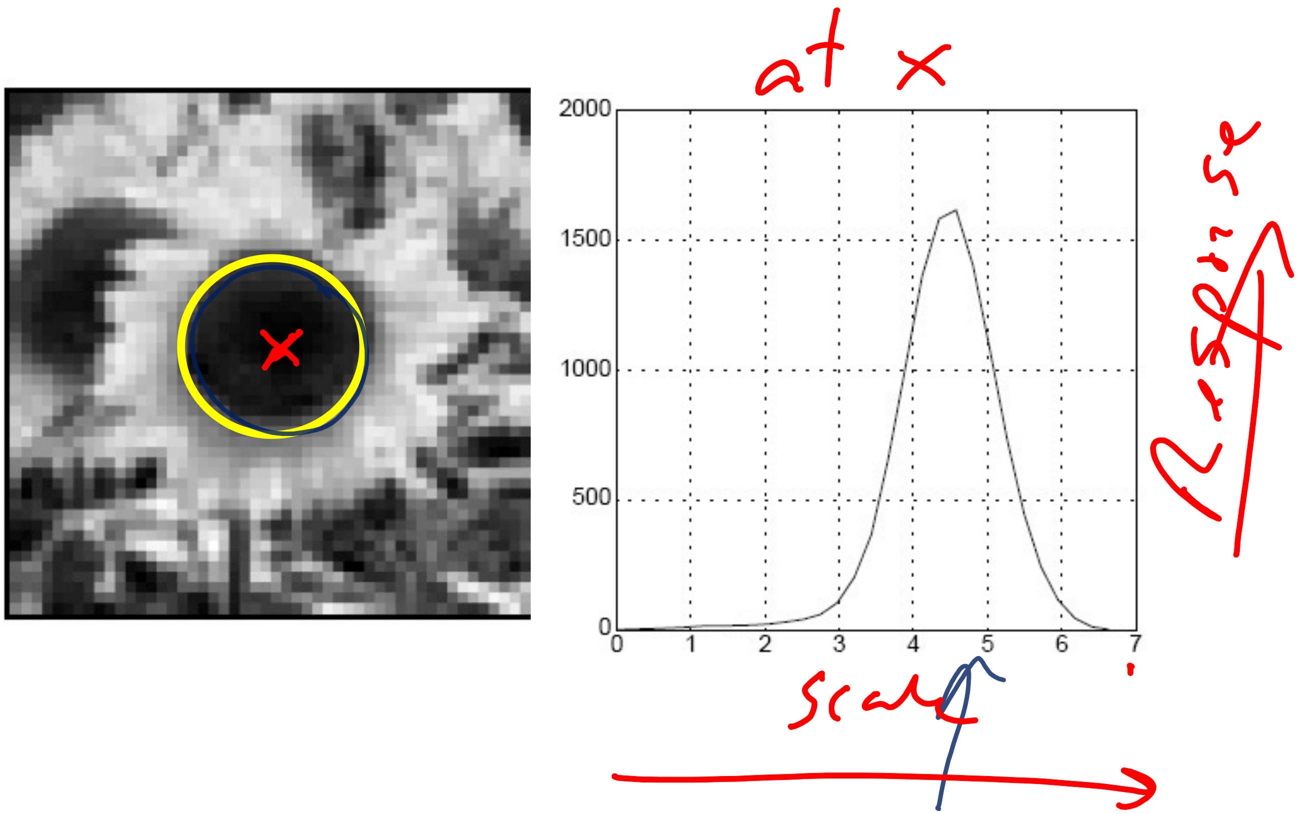
σ

T. Lindeberg (1998). "Feature detection with automatic scale selection." *International Journal of Computer Vision* **30** (2): pp 77--116.

Source: S. Seitz, S. Savarese.

Characteristic scale

- We define the characteristic scale as the scale that produces peak of Laplacian response

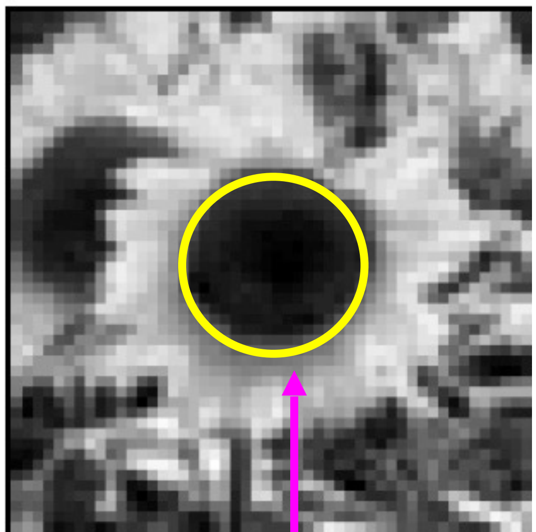


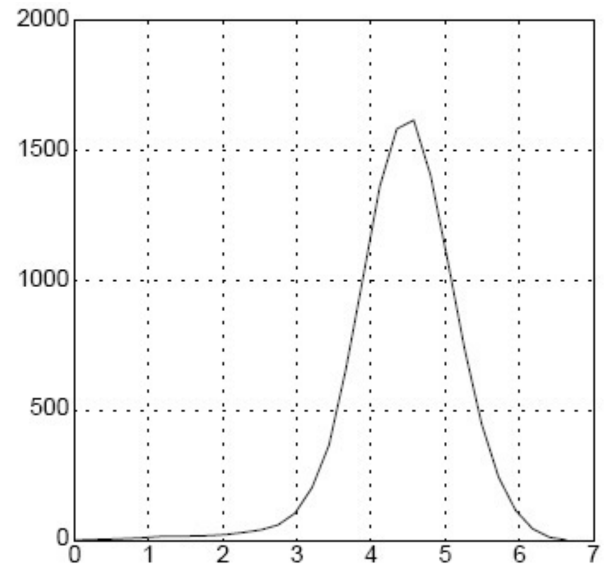
T. Lindeberg (1998). ["Feature detection with automatic scale selection."]() *International Journal of Computer Vision* **30** (2): pp 77--116.

Source: S. Seitz, S. Savarese.

Characteristic scale

- We define the characteristic scale as the scale that produces peak of Laplacian response





characteristic scale

T. Lindeberg (1998). "Feature detection with automatic scale selection." *International Journal of Computer Vision* **30** (2): pp 77--116.

Source: S. Seitz, S. Savarese.

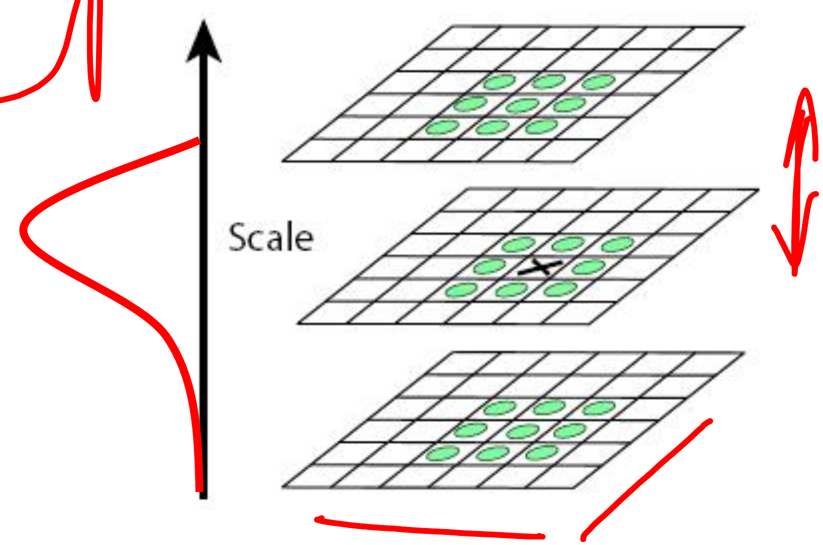
Scale-space blob detector

1. Convolve image with scale-normalized Laplacian at several scales

2. Find maxima of squared Laplacian response in scale-space (position and scale)

3. This indicates if a blob has been detected

4. And what is its intrinsic scale



Source: S. Seitz, S. Savarese.

Scale-space blob detector: example

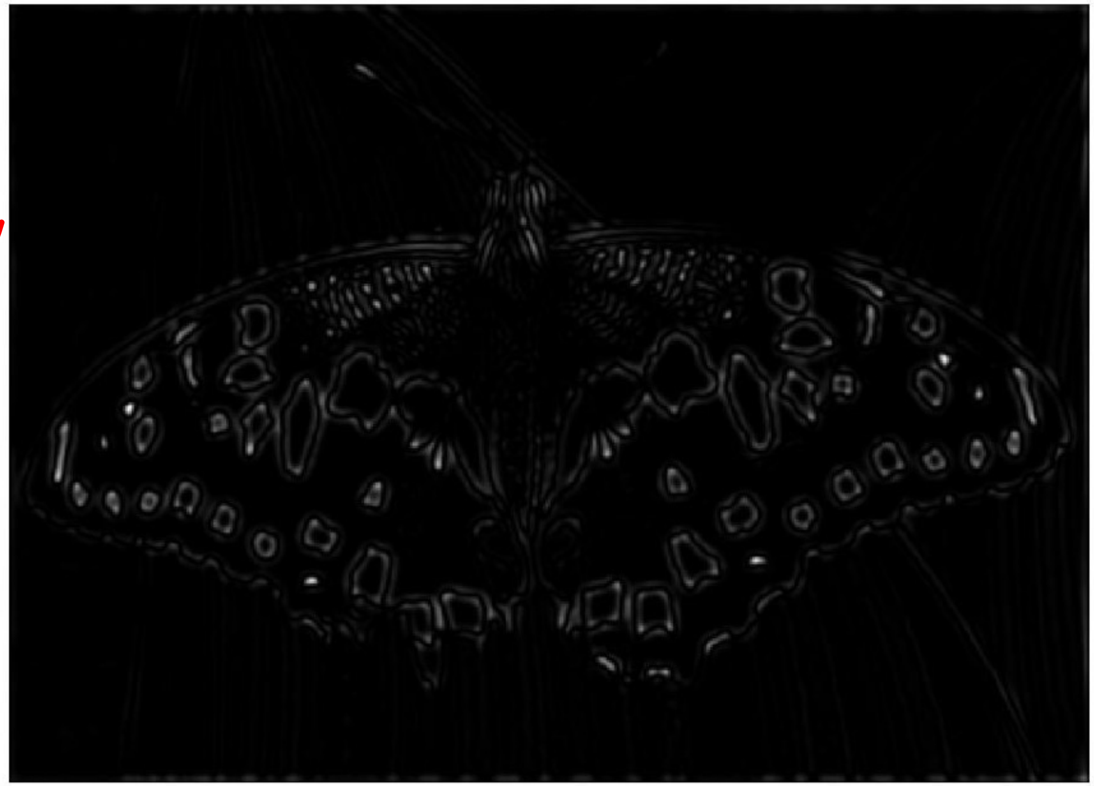
Source: S. Seitz, S. Savarese.

Scale-space blob detector: example

Source: S. Seitz, S. Savarese.

Scale-space blob detector: example

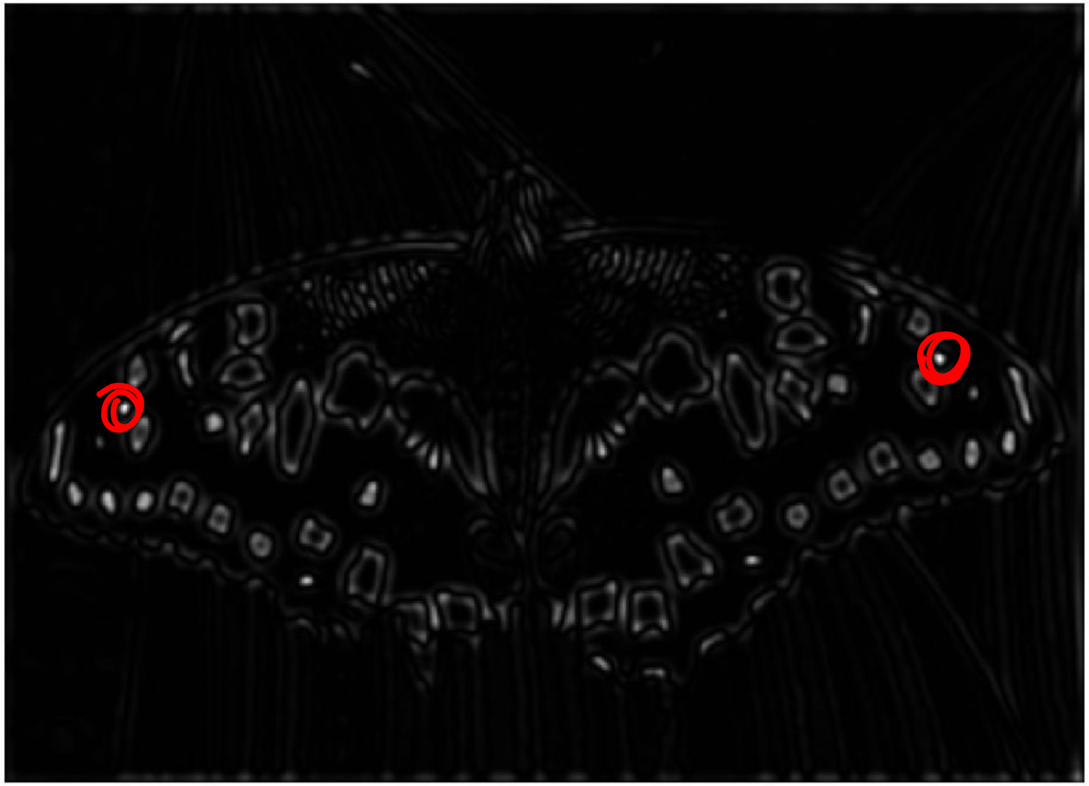
output of Laplacian at sigma = 2



sigma = 2

Source: S. Seitz, S. Savarese.

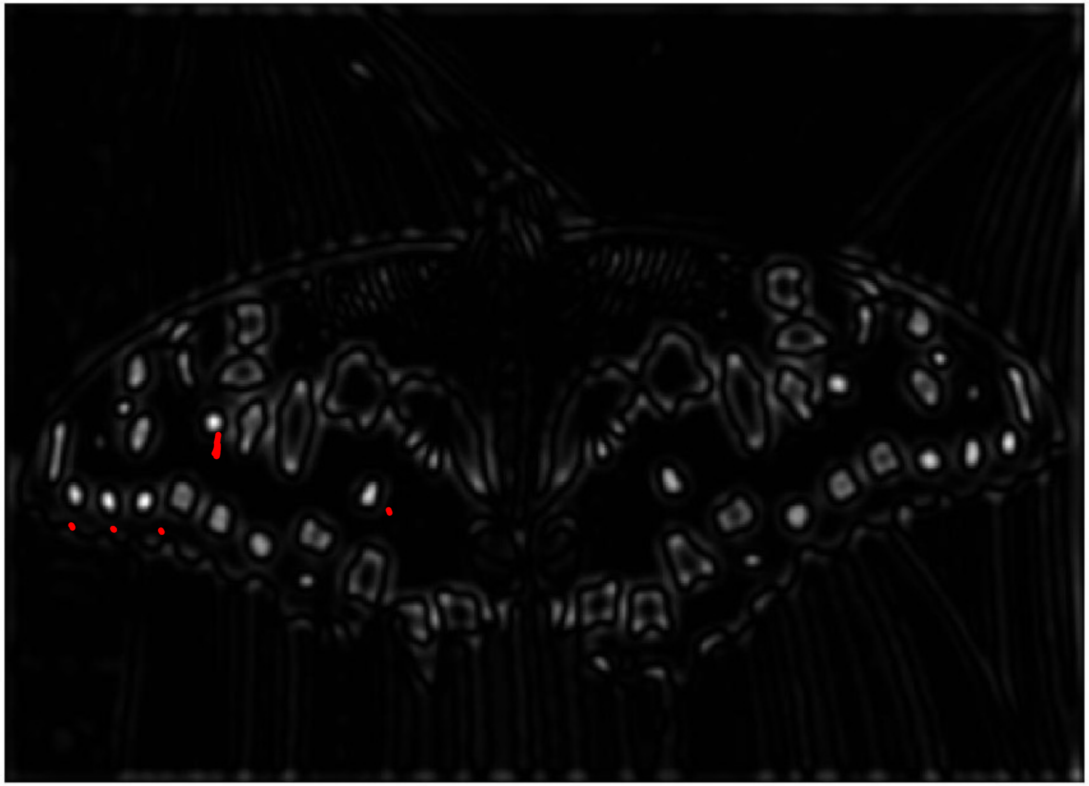
Scale-space blob detector: example



sigma = 2.5018

Source: S. Seitz, S. Savarese.

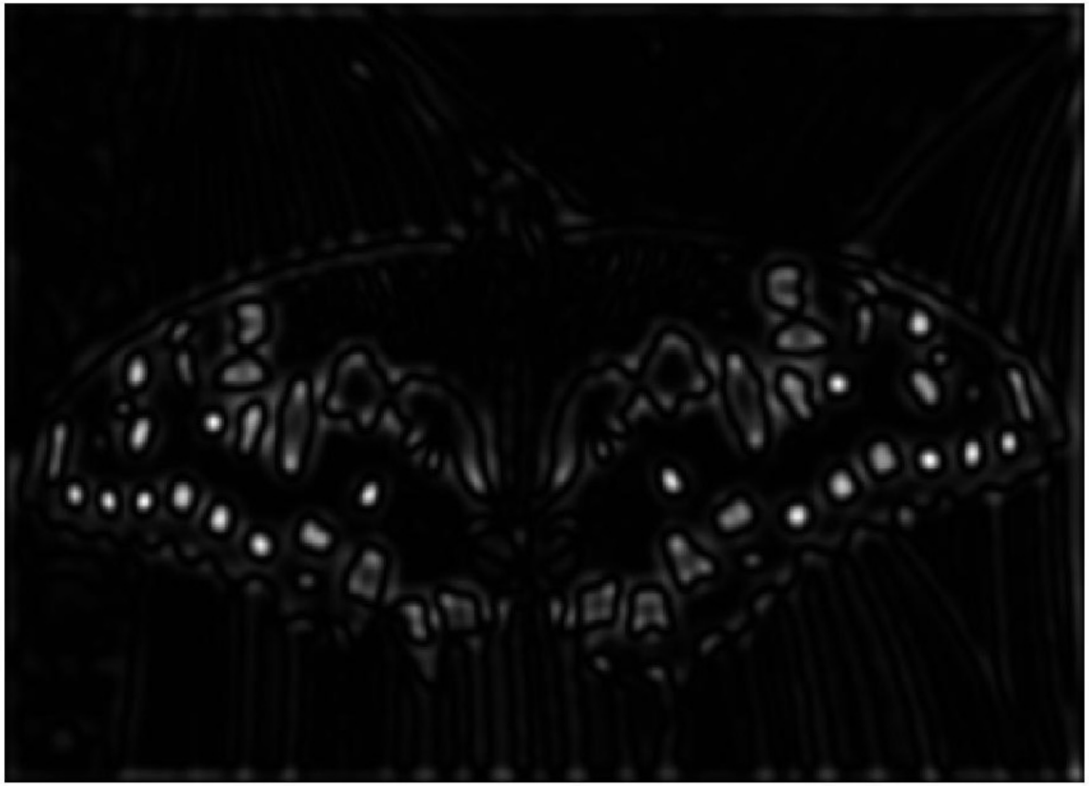
Scale-space blob detector: example



sigma = 3.1296

Source: S. Seitz, S. Savarese.

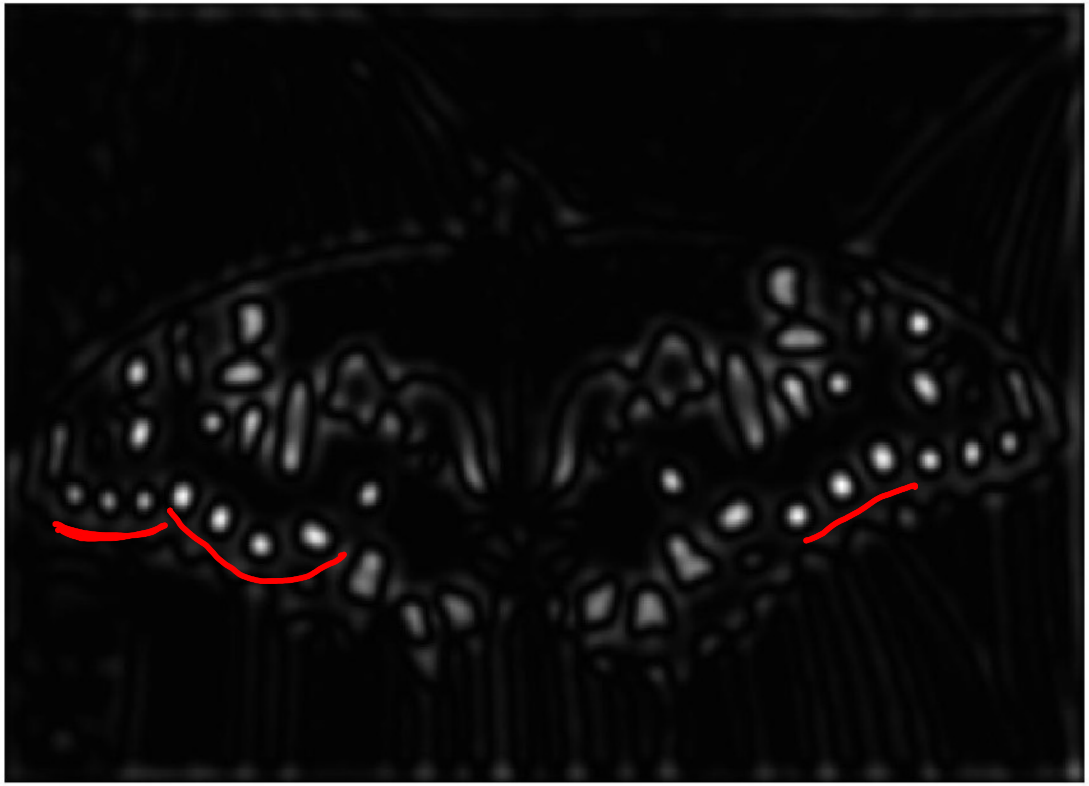
Scale-space blob detector: example



sigma = 3.9149

Source: S. Seitz, S. Savarese.

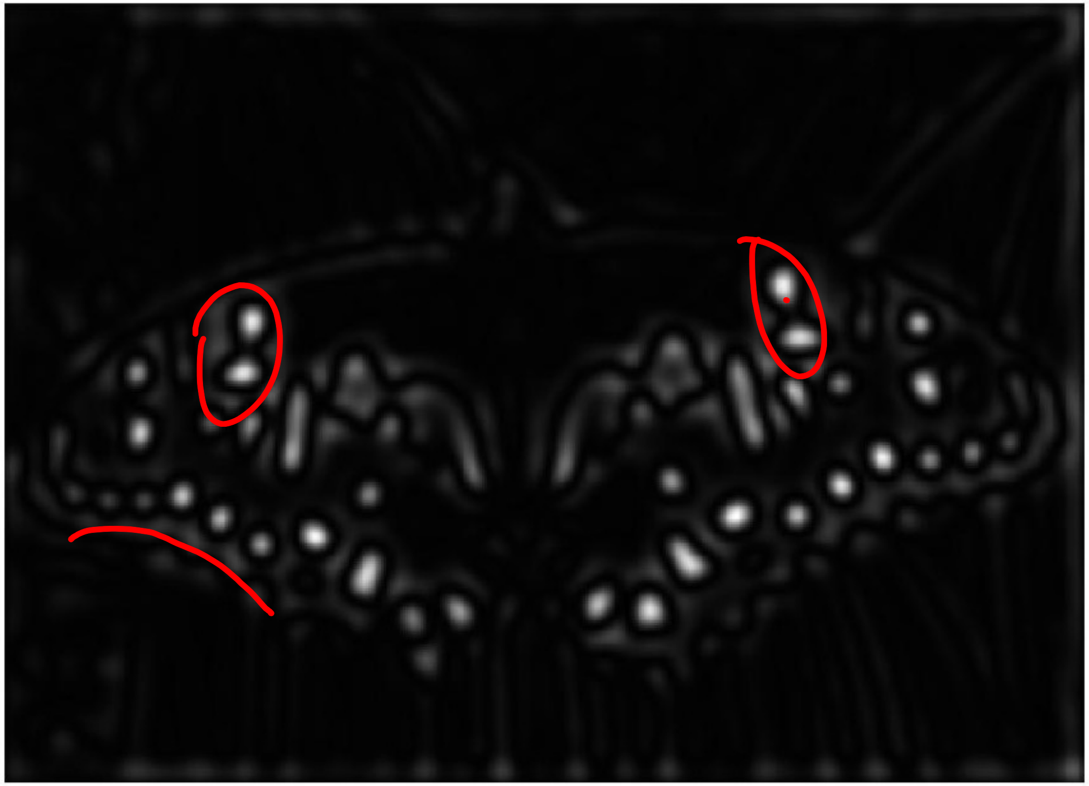
Scale-space blob detector: example



sigma = 4.8972

Source: S. Seitz, S. Savarese.

Scale-space blob detector: example



sigma = 6.126

Source: S. Seitz, S. Savarese.

Scale-space blob detector: example

sigma = 7.6631

Source: S. Seitz, S. Savarese.

Scale-space blob detector: example

sigma = 9.5859

Source: S. Seitz, S. Savarese.

Scale-space blob detector: example

sigma = 11.9912

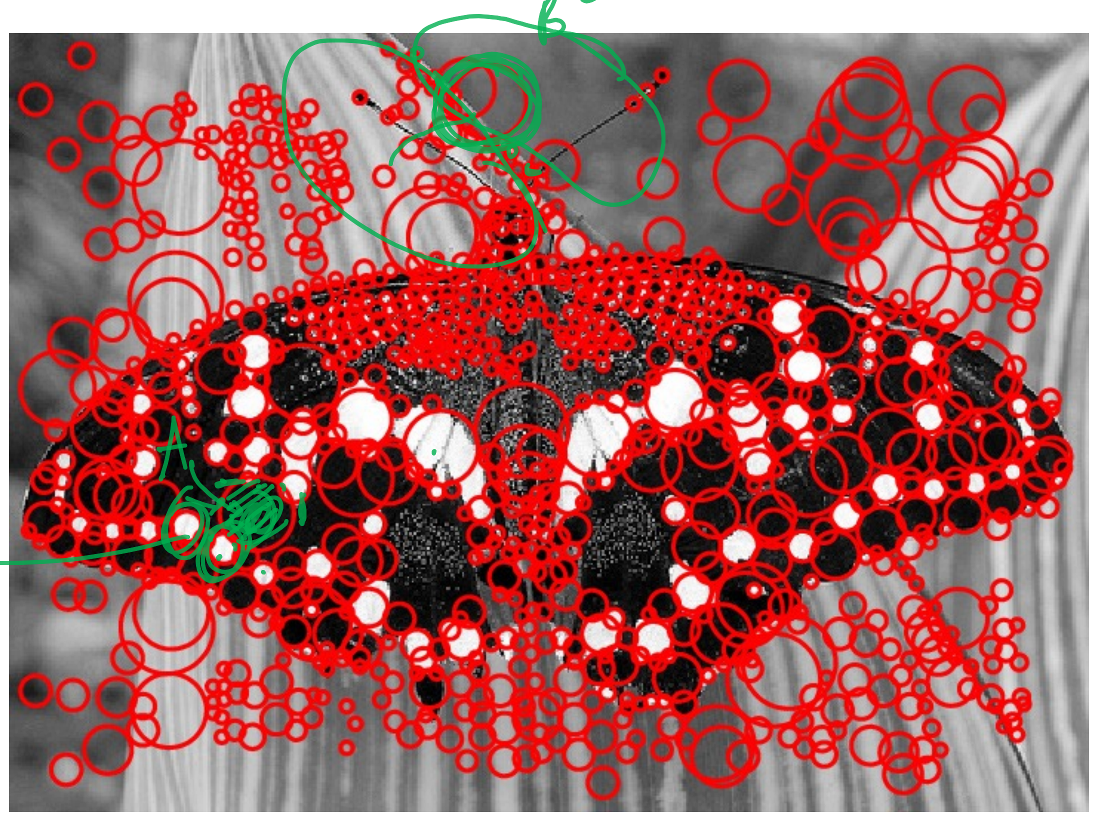
Source: S. Seitz, S. Savarese.

Scale-space blob detector: example

sigma = 11.9912

Source: S. Seitz, S. Savarese.

Scale-space blob detector: example



expert

A still an extremum in scale

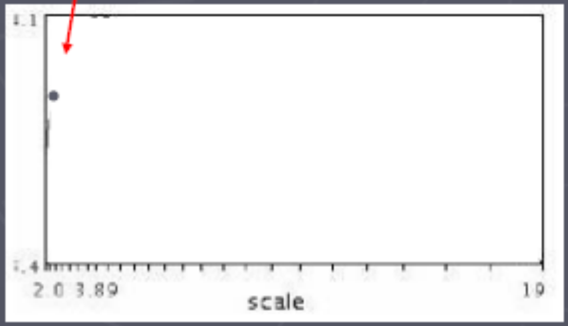
B creation of local extrema in 2D scale-space

THIS IS SIFT.

Source: S. Seitz, S. Savarese.

Automatic scale selection

Lindeberg et al., 1996



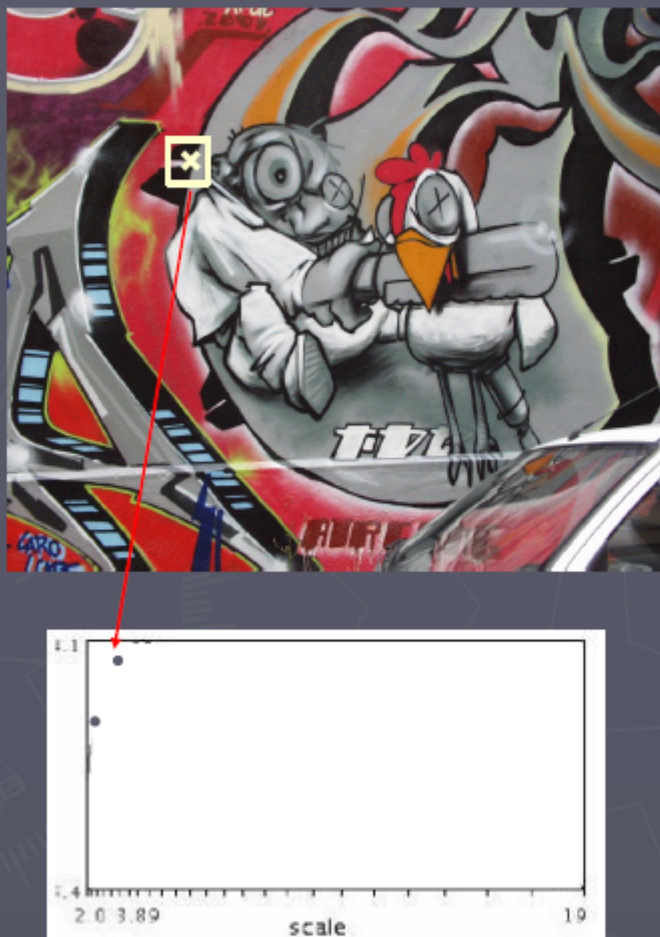
$f(I_{i_1 \ldots i_m}(x,\sigma))$

Slide from Tinne Tuytelaars

Automatic scale selection

Function responses for increasing scale

Scale trace (signature)

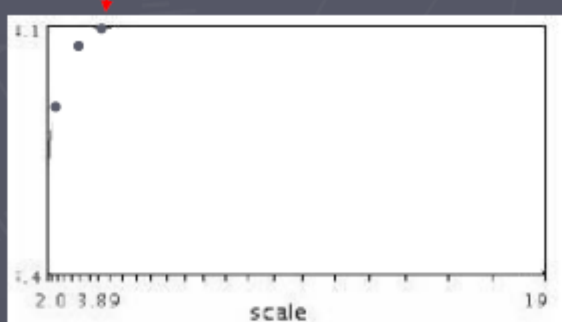


$f(I_{i_1 \ldots i_m}(x,\sigma))$

Automatic scale selection

Function responses for increasing scale

Scale trace (signature)



$f(I_{i_1 \ldots i_m}(x,\sigma))$

Automatic scale selection

Function responses for increasing scale

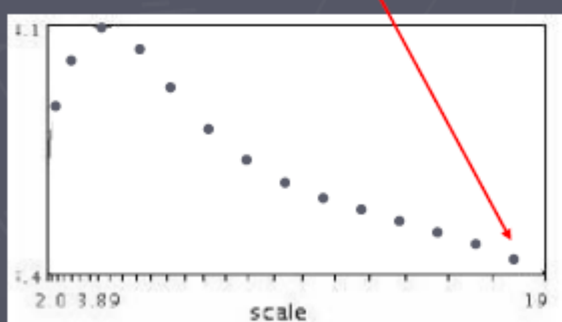
Scale trace (signature)

$$f(I_{i_1 \ldots i_m}(x,\sigma))$$

Automatic scale selection

Function responses for increasing scale

Scale trace (signature)

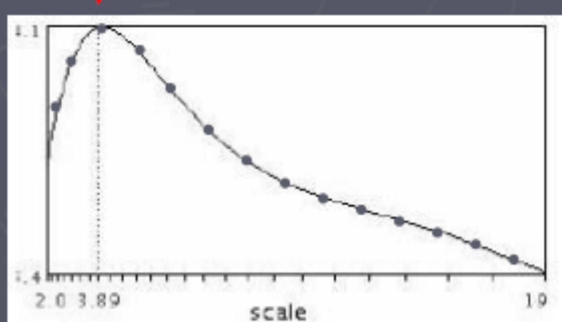


$f(I_{i_1 \ldots i_m}(x,\sigma))$

Automatic scale selection

Function responses for increasing scale

Scale trace (signature)

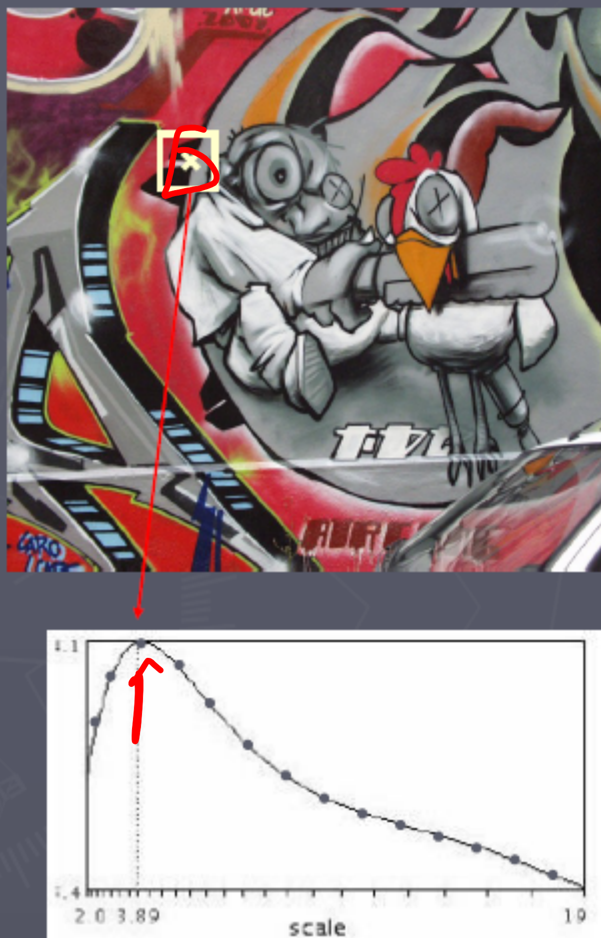


$$f(I_{i_1 \ldots i_m}(x,\sigma))$$

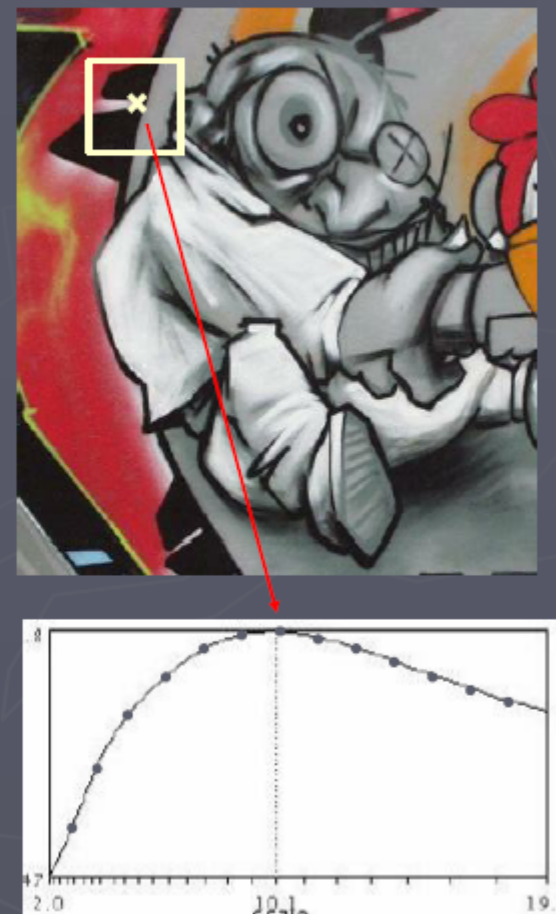
Automatic scale selection

Function responses for increasing scale

Scale trace (signature)



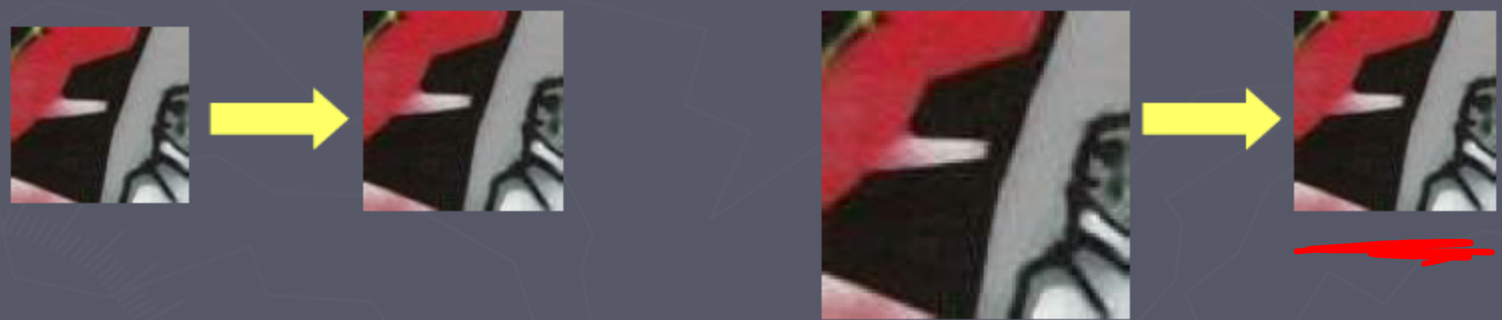
$f(I_{i_1 \ldots i_m}(x,\sigma))$



$f(I_{i_1 \ldots i_m}(x',\sigma'))$

Automatic scale selection

Normalize: rescale to fixed size



Difference of Gaussians Approximations to Laplacian

David G. Lowe. **"Distinctive image features from scale-invariant keypoints."** *IJCV* 60 (2), 04

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2 L$$

Source: S. Seitz, S. Savarese.

Difference of Gaussians Approximations to Laplacian

David G. Lowe. **"Distinctive image features from scale-invariant keypoints."** *IJCV* 60 (2), 04

- Approximating the Laplacian with a difference of Gaussians:

$$L = \sigma^2 \left(G_{xx}(x, y, \sigma) + G_{yy}(x, y, \sigma) \right)$$

Laplacian

$$DoG = G(x, y, k\sigma) - G(x, y, \sigma)$$

Difference of Gaussians

or

Difference of gaussian blurred images at scales k σ and σ

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2 L$$

Source: S. Seitz, S. Savarese.

Difference of Gaussians Approximations to Laplacian

David G. Lowe. **"Distinctive image features from scale-invariant keypoints."** *IJCV* 60 (2), 04

- Approximating the Laplacian with a difference of Gaussians:

$$L = \sigma^2 \left(G_{xx}(x, y, \sigma) + G_{yy}(x, y, \sigma) \right)$$

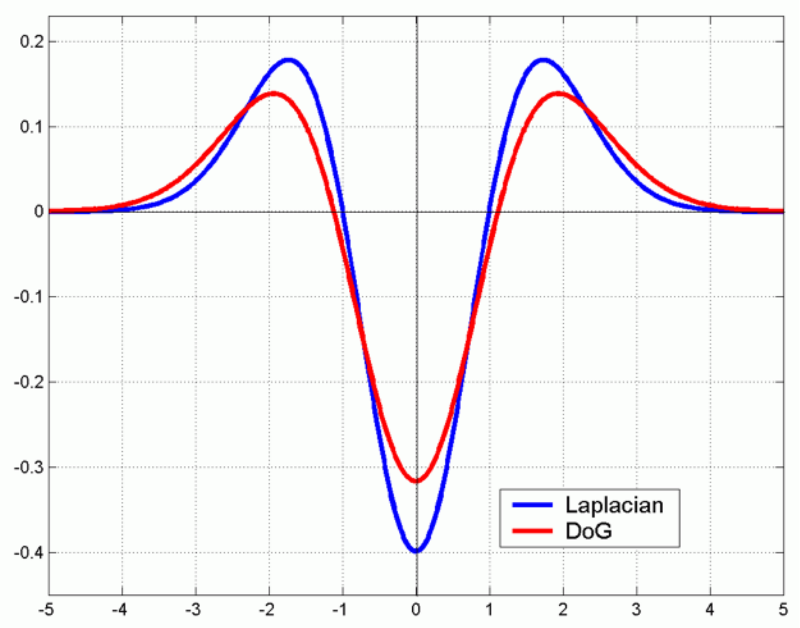
Laplacian

$$DoG = G(x, y, k\sigma) - G(x, y, \sigma)$$

Difference of Gaussians

or

Difference of gaussian blurred images at scales k σ and σ

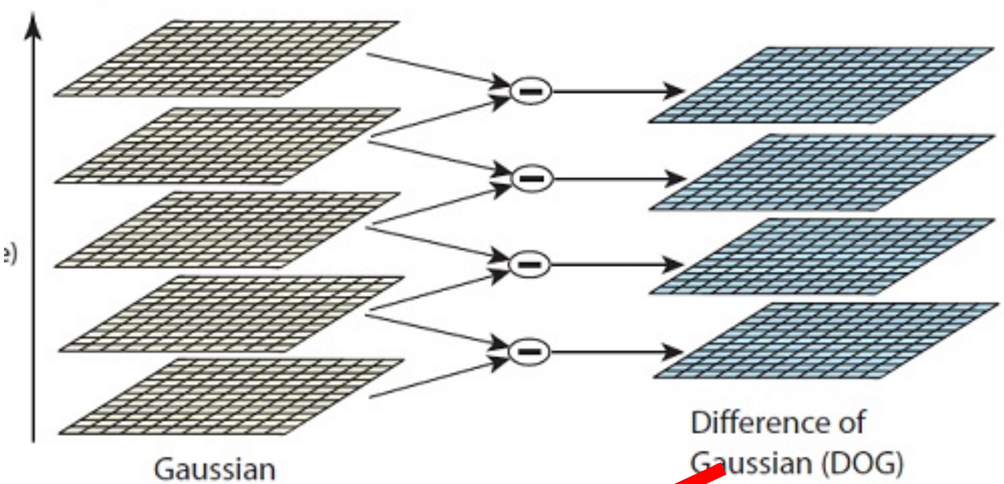


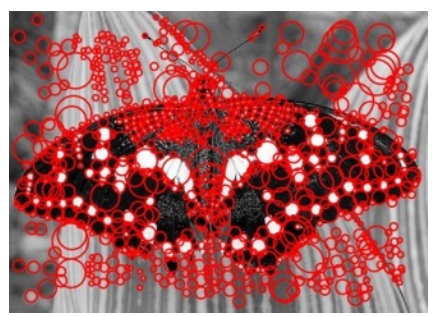
$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2 L$$

Source: S. Seitz, S. Savarese.

Difference of Gaussians (DoG)

k

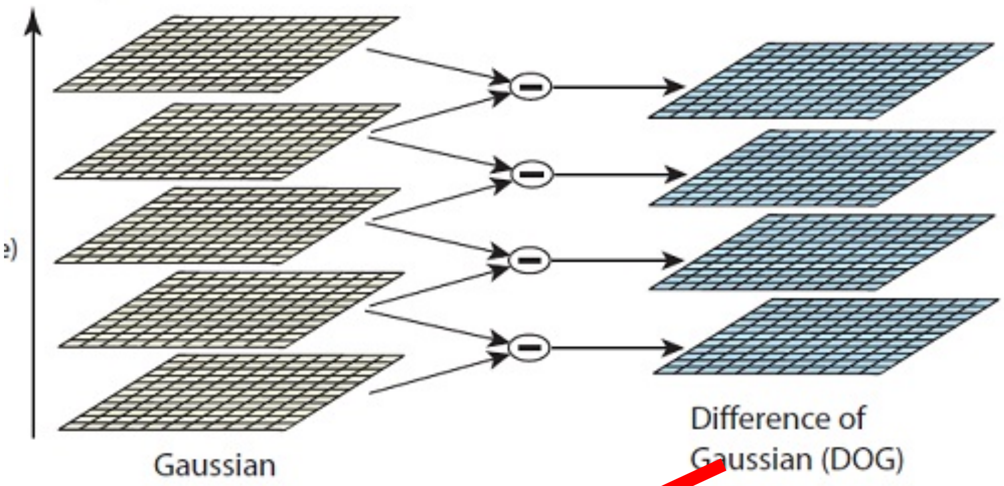


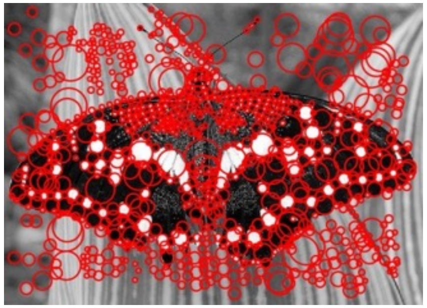


Source: S. Seitz, S. Savarese.

Difference of Gaussians (DoG)

k





Output: location, scale, orientation

Source: S. Seitz, S. Savarese.

End

Invariance

affine

Detector	Illumination	Rotation	Scale	View point
Harris corner	Yes partially	Yes	No	No
Lowe '99 (DoG)	Yes	Yes	Yes	No

affine

projective

Source: S. Savarese.

Harris-Laplace

[Mikolajczyk & Schmid '01]

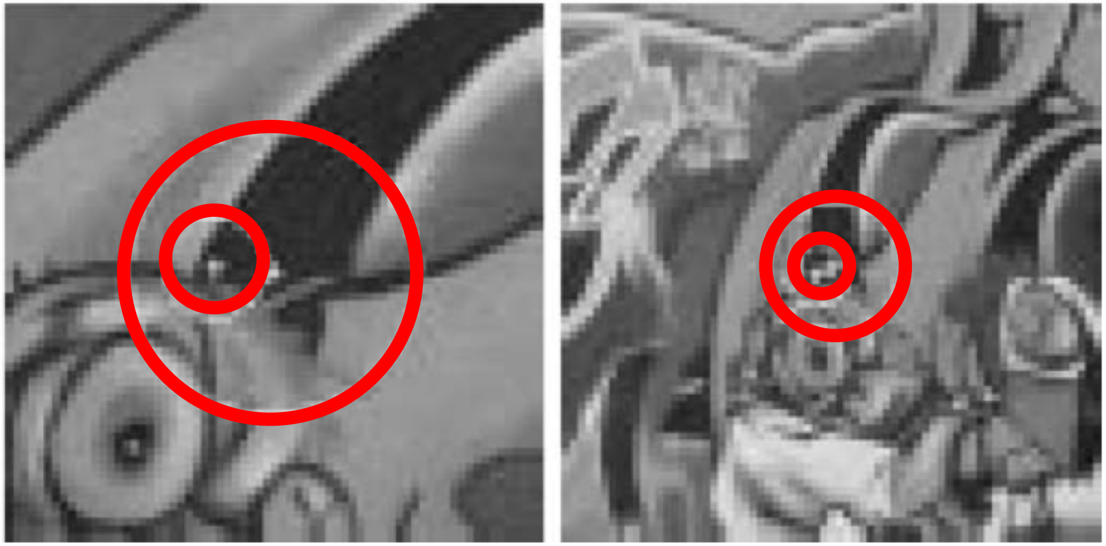
- Collect locations (x,y) of detected Harris features for $\sigma = \sigma_1 \dots \sigma_2$ (the sigma is here comes from g_x, g_y)
- For each detected location (x,y) and for each σ, reject detection if Laplacian(x,y, σ) is not a local maximum

Source: S. Savarese.

Harris-Laplace

[Mikolajczyk & Schmid '01]

- Collect locations (x,y) of detected Harris features for $\sigma = \sigma_1 \ldots \sigma_2$ (the sigma is here comes from $g_{x,}$ g_y)
- For each detected location (x,y) and for each σ, reject detection if Laplacian(x,y, σ) is not a local maximum

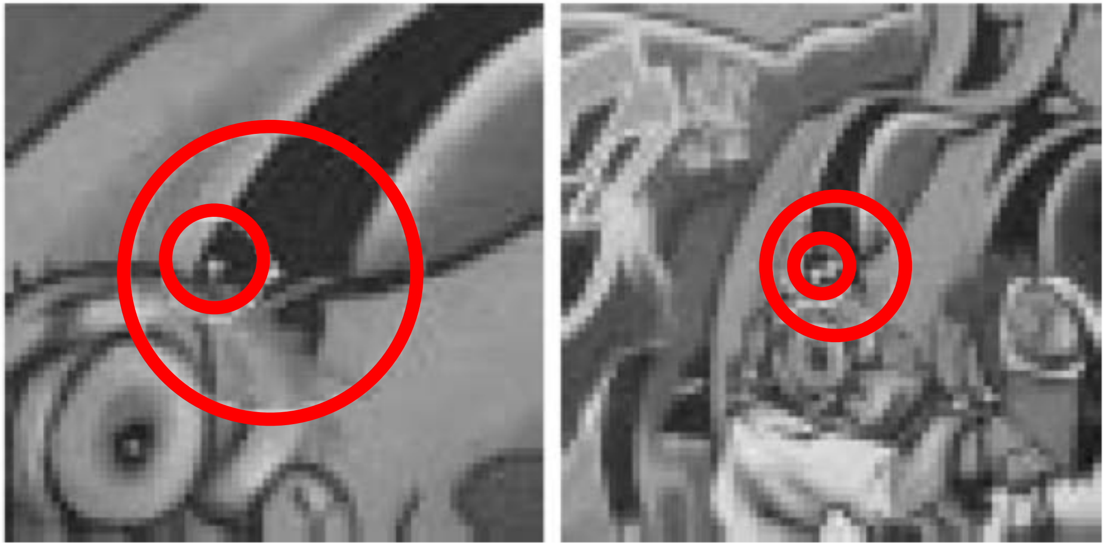


Source: S. Savarese.

Harris-Laplace

[Mikolajczyk & Schmid '01]

- Collect locations (x,y) of detected Harris features for $\sigma = \sigma_1 \ldots \sigma_2$ (the sigma is here comes from $g_{x,}$ g_y)
- For each detected location (x,y) and for each σ, reject detection if Laplacian(x,y, σ) is not a local maximum



Output: location, scale

Source: S. Savarese.

Invariance

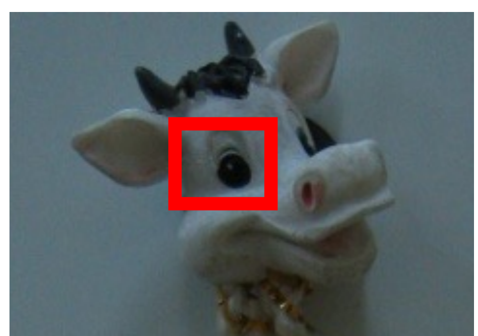
Detector	Illumination	Rotation	Scale	View point
Harris corner	Yes partial	Yes	No	No
Lowe '99 (DoG)	Yes	Yes	Yes	No
Mikolajczyk & Schmid '01 Harris-Laplace	Yes partial	Yes	Yes	No

Source: S. Savarese.

Repeatability

Illumination invariance

Scale invariance

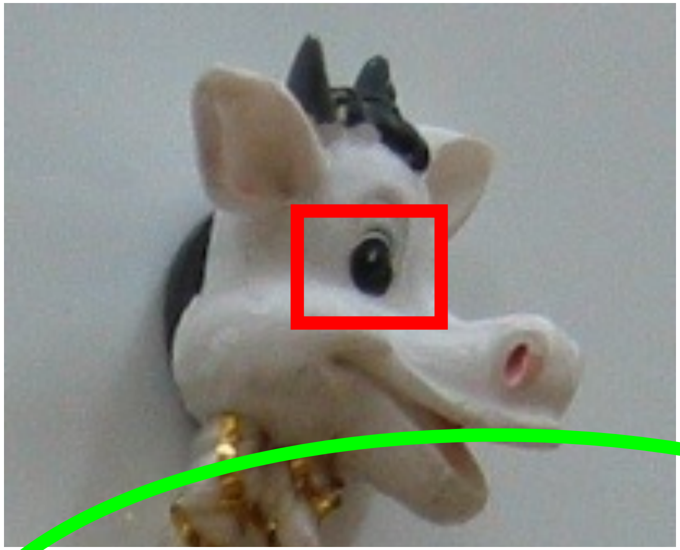


Pose invariance
- Rotation
- Affine

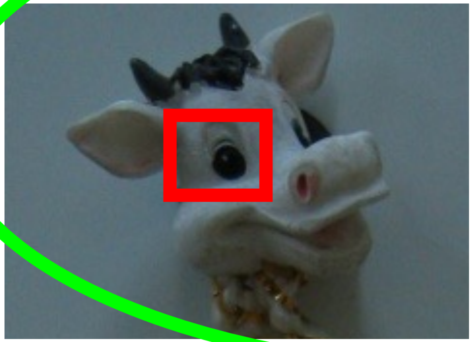
Source: S. Savarese.

Repeatability

Illumination invariance



Scale invariance



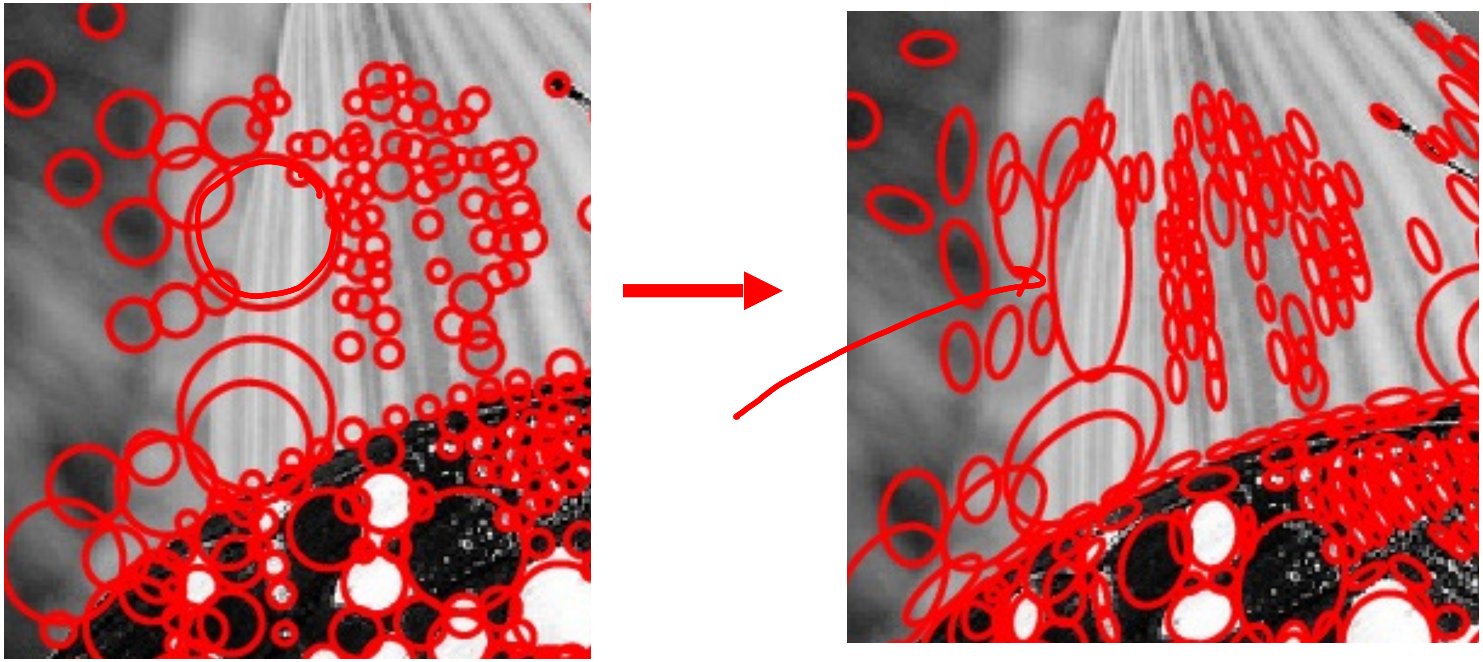
Pose invariance
- Rotation
- Affine

Source: S. Savarese.

Affine invariance

K. Mikolajczyk and C. Schmid, Scale and Affine invariant interest point detectors, IJCV 60(1):63-86, 2004.

Similarly to characteristic scale selection, detect the characteristic shape of the local feature



Source: S. Savarese.

Affine invariance

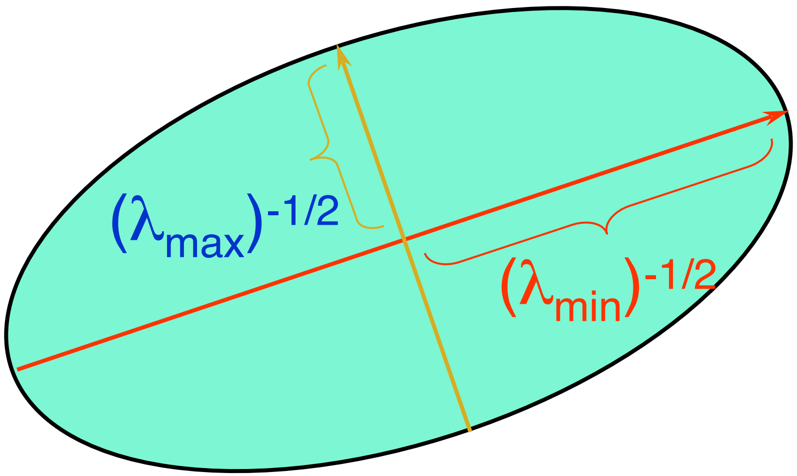
$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = R^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$$

Source: S. Savarese.

Affine invariance

$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = R^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$$

We can visualize *M* as an ellipse with axis lengths determined by the eigenvalues and orientation determined by *R*

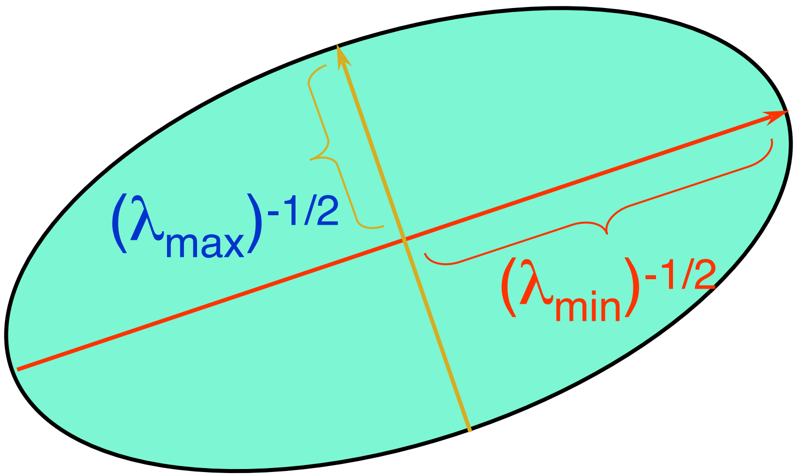


Source: S. Savarese.

Affine invariance

$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = R^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$$

We can visualize *M* as an ellipse with axis lengths determined by the eigenvalues and orientation determined by *R*



The second moment ellipse can be viewed as the "characteristic shape" of a region

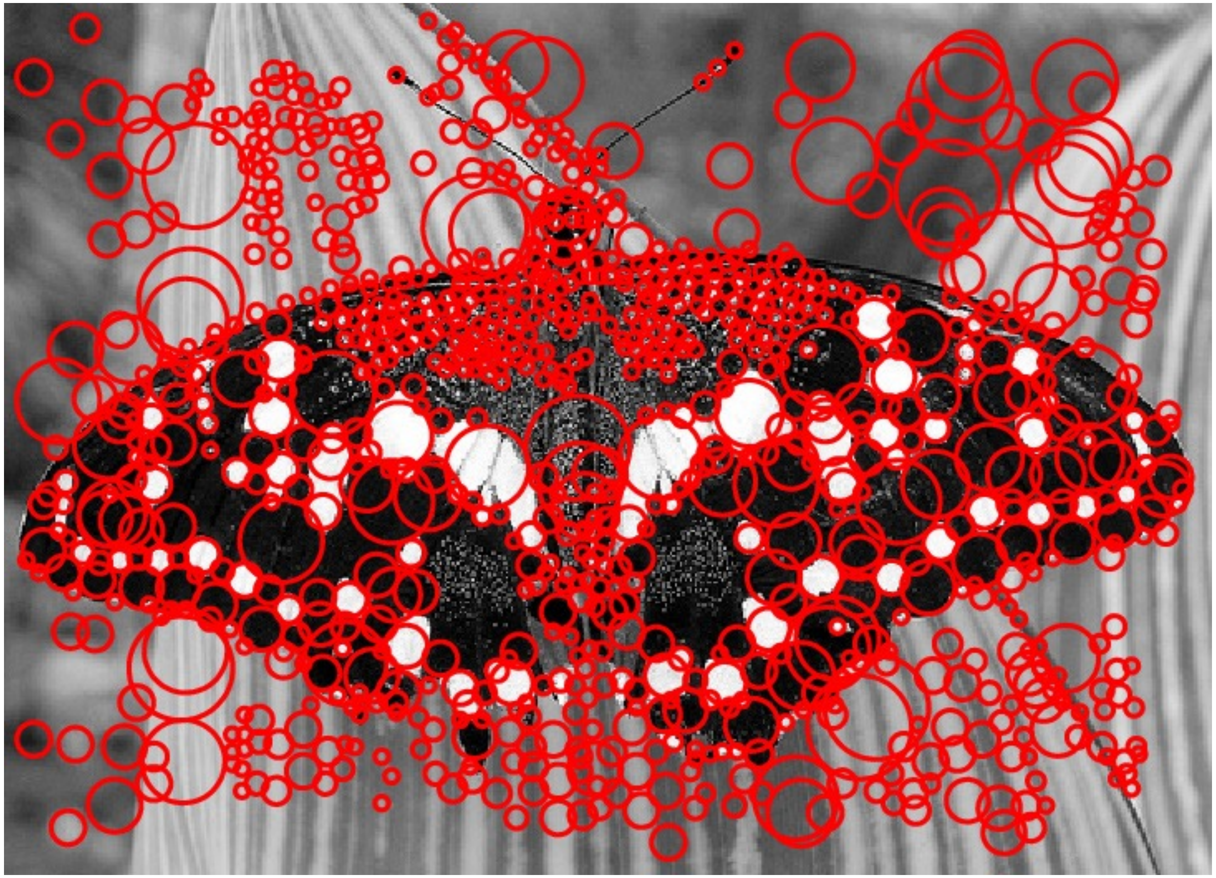
Source: S. Savarese.

Affine adaptation

1. Detect initial region with Harris Laplace
2. Estimate affine shape with M
3. Normalize the affine region to a circular one
4. Re-detect the new location and scale in the normalized image
5. Go to step 2 if the eigenvalues of the M for the new point are not equal [detector not yet adapted to the characteristic shape]

Source: S. Savarese.

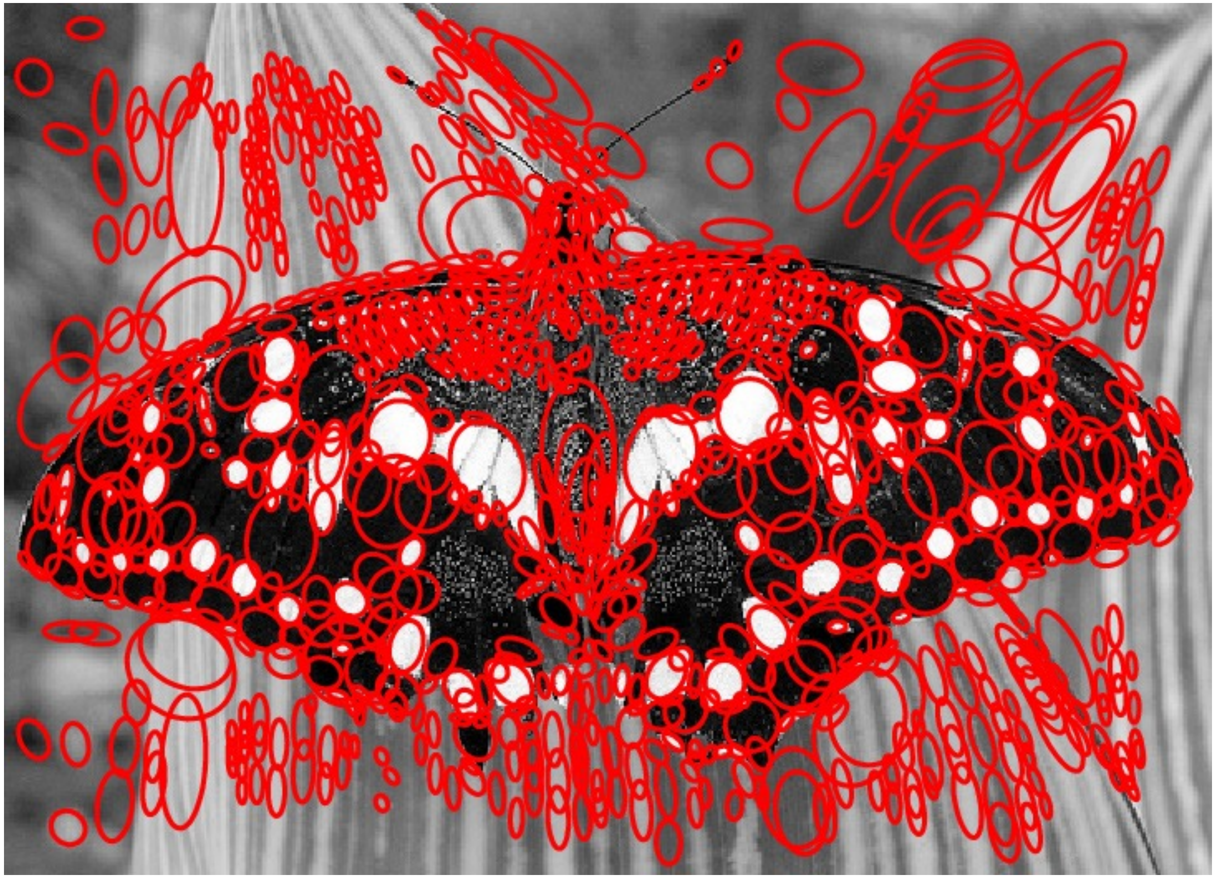
Without affine invariance



Scale-invariant regions (blobs)

Source: S. Savarese.

With affine invariance



Affine-adapted blobs

Invariance

Detector	Illumination	Rotation	Scale	~~View point~~ Affine
Harris corner	Yes	Yes	No	No
Lowe '99 (DoG)	Yes	Yes	Yes	No
Mikolajczyk & Schmid '01	Yes	Yes	Yes	No
Mikolajczyk & Schmid '0~~2~~4	Yes	Yes	Yes	Yes

Source: S. Savarese.

Detector	Illumination	Rotation	Scale	View point
Harris corner	Yes	Yes	No	No
Lowe '99 (DoG)	Yes	Yes	Yes	Yes
Mikolajczyk & Schmid '01, '02	Yes	Yes	Yes	Yes
Tuytelaars, '00	Yes	Yes	No (Yes '04)	Yes
Kadir & Brady, 01	Yes	Yes	Yes	no
Matas, '02	Yes	Yes	Yes	no

Source: S. Savarese.