Interest Point Detection II

Dr Wenjia Bai

Department of Computing & Brain Sciences

Interest point detection

- How to handle scales in detection?
  - Scale adapted Harris detector
- Other interest point detectors
  - Laplacian of Gaussian
  - Difference of Gaussian

Scale

- Harris detector is not invariant to scale.

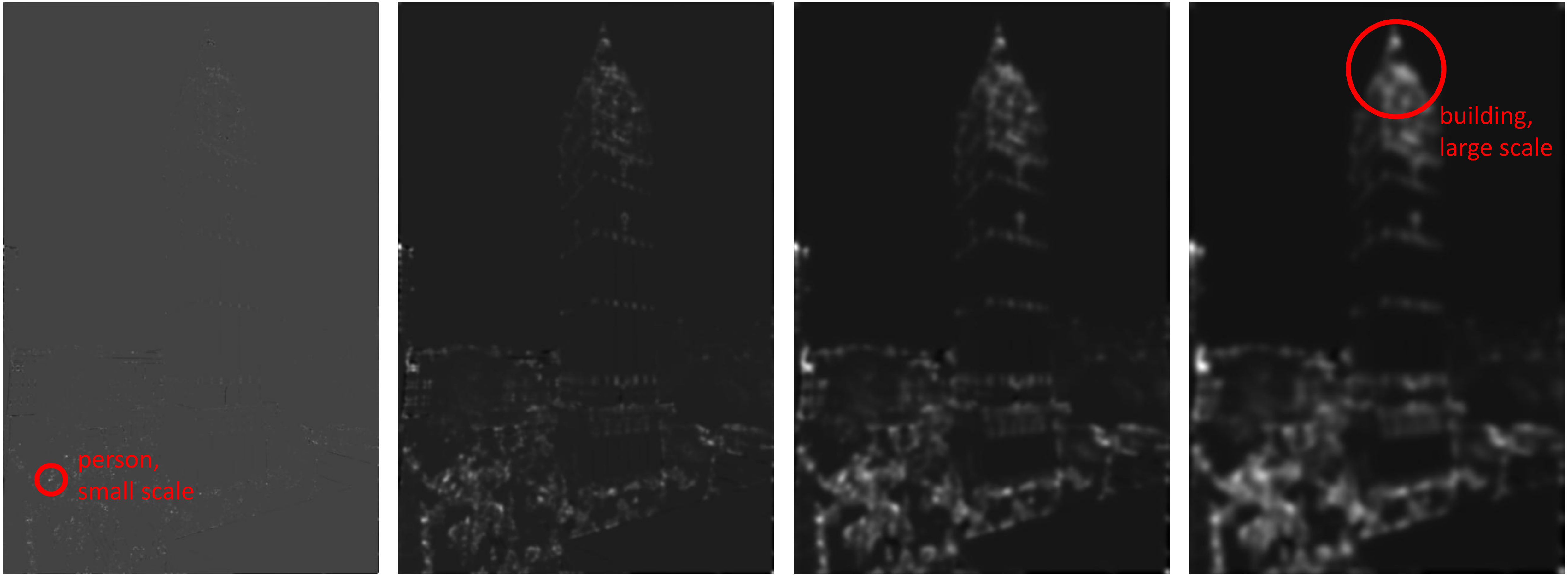
$\sigma = 1$

$\sigma = 3$

$\sigma = 5$

$\sigma = 7$

Image convolved with Gaussian kernels of different σ provide information at different scale.

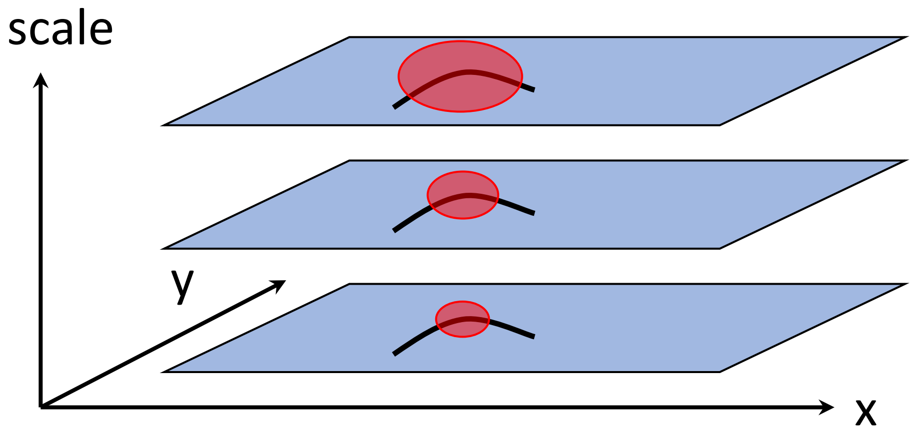


$\sigma = 1$ $\sigma = 3$ $\sigma = 5$ $\sigma = 7$

Harris detector response when different Gaussian kernels are used for calculating image derivatives.

Scale

- How do we determine the scale we use at each pixel?
  - Are we looking at a big building or a small cat?
- Intuitive idea
  - We check whether the Harris detector gives the highest response at this scale.



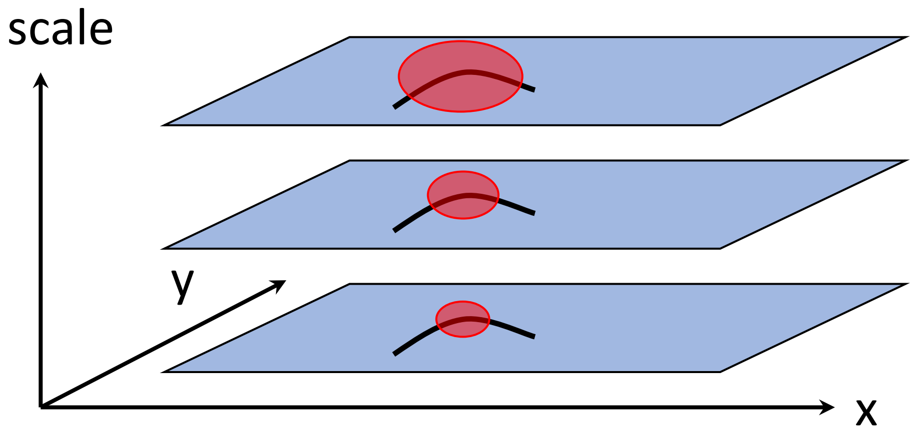
Harris detector response at different scales

If this region looks most like a corner at this scale σ, Harris detector response should be high.

If this region does not look like a corner at this scale, Harris detector response will be low.

Scale

- There is only one problem.
- A direct comparison of the Harris detector response between scales may not be fair.



Harris detector response at different scales

If this region looks most like a corner at this scale σ, Harris detector response should be high.

If this region does not look like a corner at this scale, Harris detector response will be low.

Response at different scales

- The Harris detector response is calculated using the eigenvalues of M,

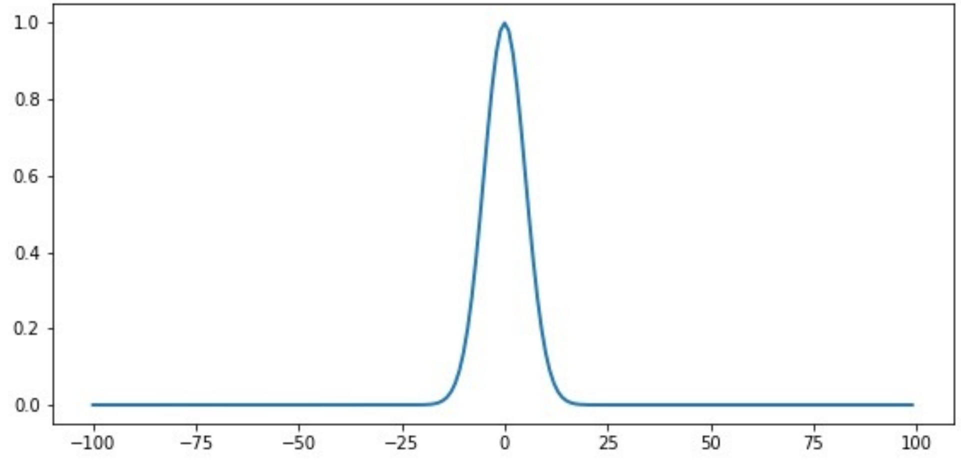
$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2(\sigma) & I_x(\sigma)I_y(\sigma) \\ I_x(\sigma)I_y(\sigma) & I_y^2(\sigma) \end{bmatrix}$$

image gradient at scale σ
(Gaussian kernel σ + derivative)

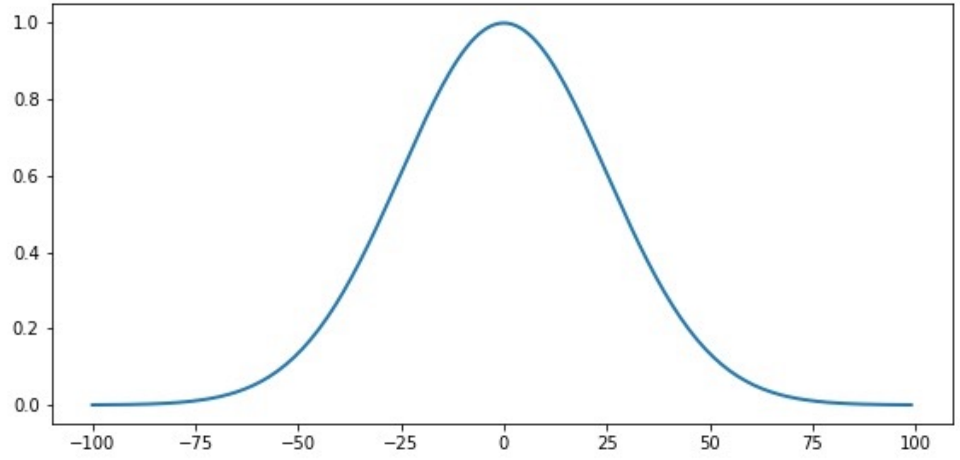
- The response is determined by the eigenvalues of M, which are in turn determined by the derivatives $I_x(\sigma)$ and $I_y(\sigma)$.
- As you will see, the derivatives $I_x(\sigma)$ and $I_y(\sigma)$ are inversely proportional to scale σ.
  - The larger the scale σ, the smaller the derivative magnitude.

Derivatives at different scales

- Consider two signals $f(x)$ and $g(x)$, which are related by
$$f(x) = g(sx)$$
i.e. they only differ by scale s and they have same peak magnitude.
- This can happen, for example, when we take a picture of the same object (human, mountain etc) with different zoom factors.



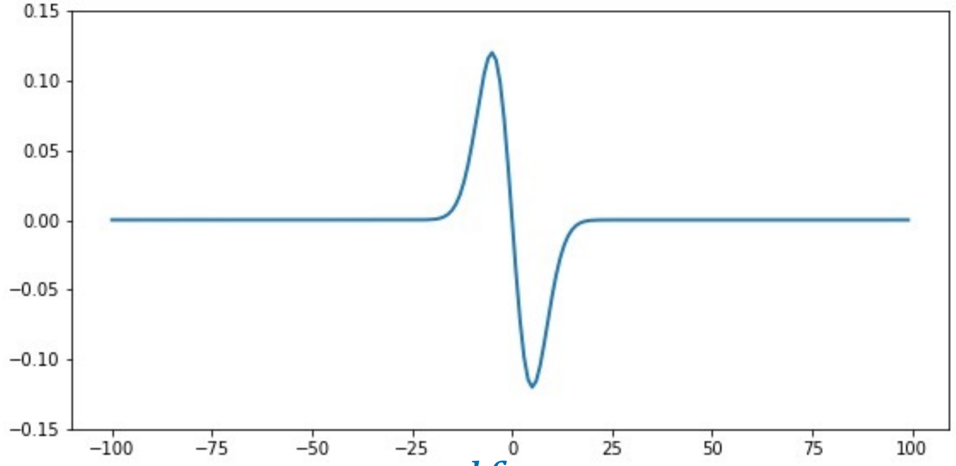
$f(x)$



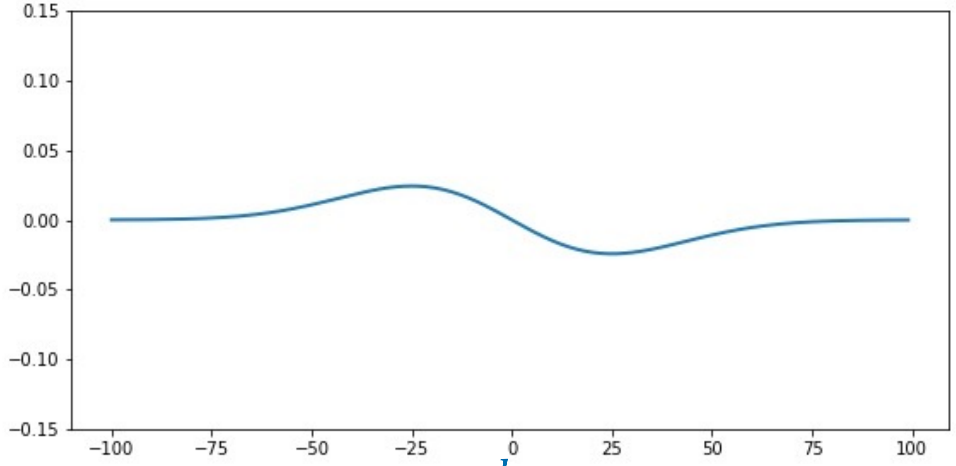
$g(x)$

Derivatives at different scales

- However, when we calculate the derivatives, their peak magnitudes differ.
- How much do they differ?



$$\frac{df}{dx}$$



$$\frac{dg}{dx}$$

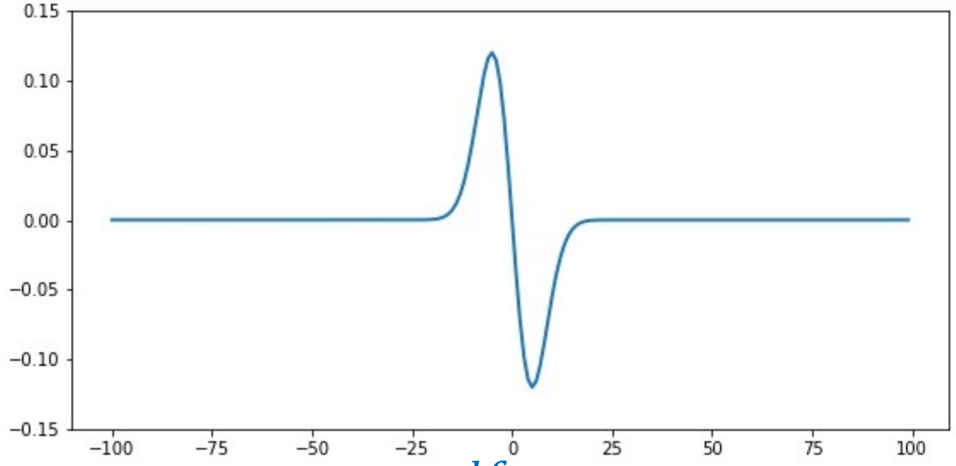
Derivatives at different scales

- Since $f(x) = g(sx)$, we have
$$f(x + \Delta x) - f(x) = g(sx + s\Delta x) - g(sx)$$
- Let us write down the Taylor expansion for $f(x + \Delta x)$ and $g(sx + s\Delta x)$,
$$f(x + \Delta x) = f(x) + \Delta x \cdot f'(x) + \cdots$$
$$g(x + \Delta x) = g(x) + \Delta x \cdot g'(x) + \cdots$$
$$g(sx + s\Delta x) = g(sx) + s\Delta x \cdot g'(sx) + \cdots$$
- Substitute into the first equation, we have,
$$\Delta x \cdot f'(x) = s\Delta x \cdot g'(sx)$$
$$f'(x) = sg'(sx)$$
- In other words,
$$\frac{df}{dx} = \boldsymbol{s} \cdot \frac{dg}{dx}\Big|_{sx}$$

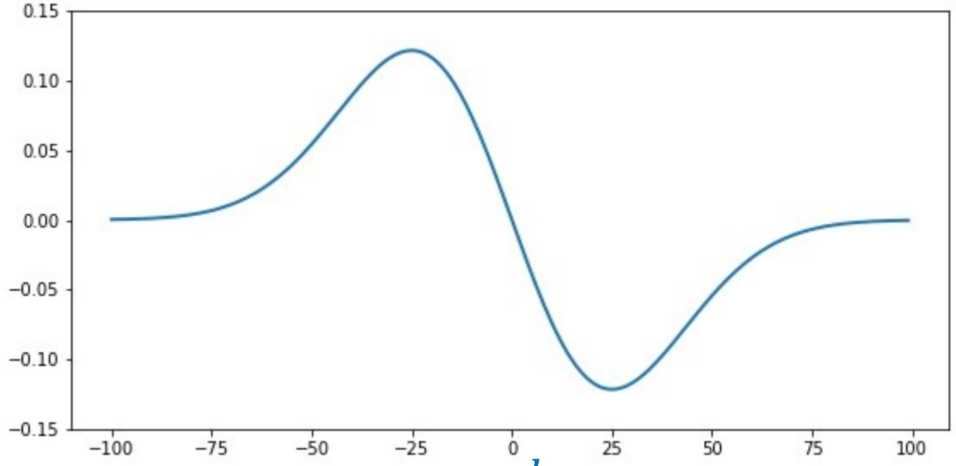
T. Lindeberg. Feature detection with automatic scale selection. IJCV 1998.

Signal at different scales

- To make the derivative magnitude comparable across scales, we need to multiply the derivative by its scale s.
  - Then $\frac{df}{dx}$ can be compared with $s\frac{dg}{dx}$.
  - The same object will give same magnitude of response, regardless of the zoom factor.



$\frac{df}{dx}$



$s\frac{dg}{dx}$

Scale adapted Harris detector

normaliser

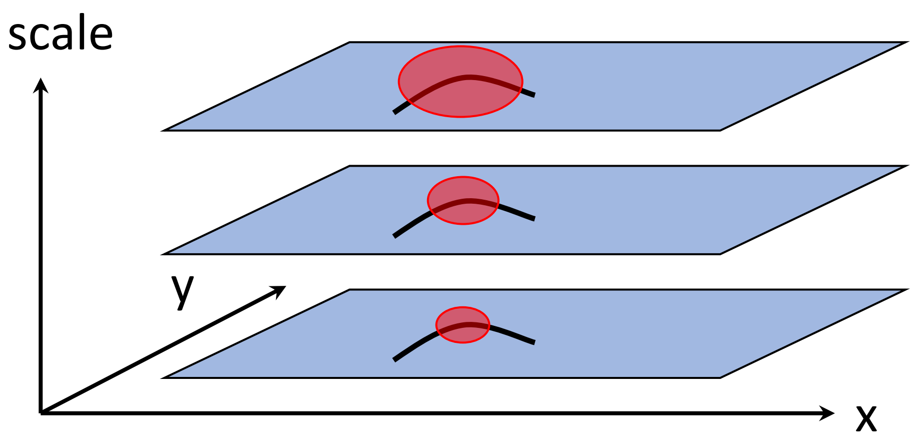
- This is the scale adapted Harris detector,

$$
\begin{aligned}
M &= \sum_{x,y} w(x,y) \begin{bmatrix} \sigma^2 I_x^2(\sigma) & \sigma^2 I_x(\sigma) I_y(\sigma) \\ \sigma^2 I_x(\sigma) I_y(\sigma) & \sigma^2 I_y^2(\sigma) \end{bmatrix} \\
&= \sum_{x,y} w(x,y) \sigma^2 \begin{bmatrix} I_x^2(\sigma) & I_x(\sigma) I_y(\sigma) \\ I_x(\sigma) I_y(\sigma) & I_y^2(\sigma) \end{bmatrix}
\end{aligned}
$$

- Apply scale adapted Harris detector at multiple scales.
- At each pixel, determine the scale which gives us the largest detector response (e.g. at this scale, it looks most like a corner).

Scale adapted Harris detector

- We calculate the scale adapted detector response for a series of σ, from small-scale to large-scale.
- When we perform interest point detection, we look for local maxima both across space and across scale.



Harris detector response at different scales

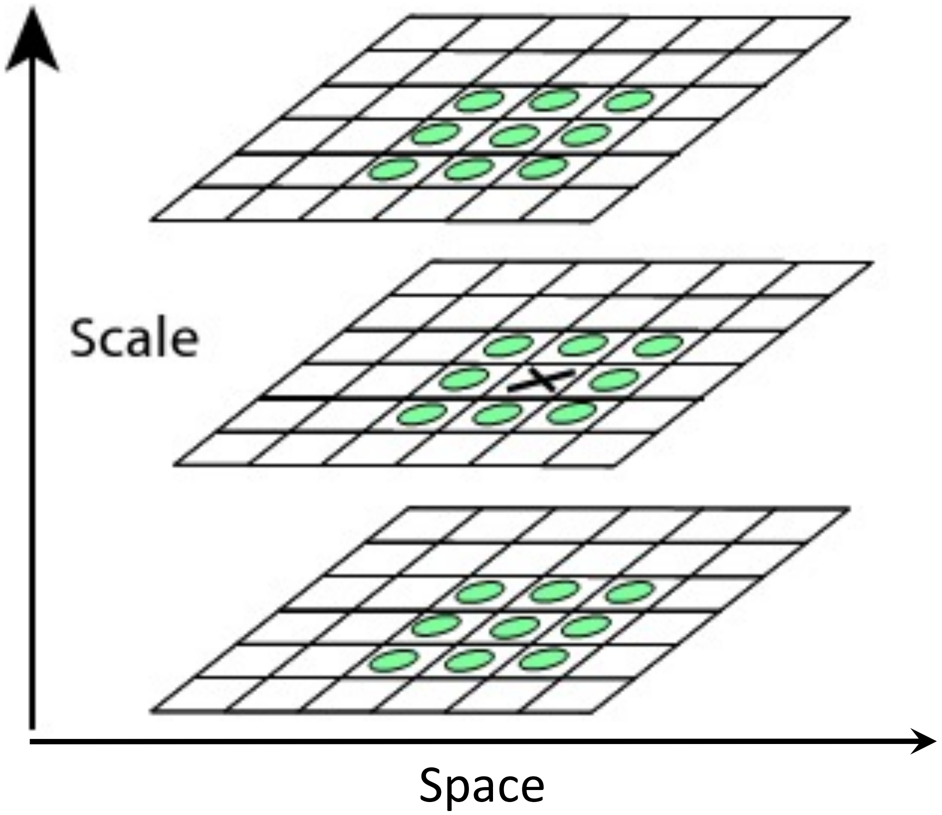
For example,

$\sigma_3 = 4\, pixel$

$\sigma_2 = 2\, pixel$

$\sigma_1 = 1\, pixel$

Scale-space extrema



X is detected as an interest point if it is a local maximum both along scale dimension (most appropriate scale) and across space. A threshold may also be applied.

Scale adapted Harris detector

Algorithm

For each scale σ

Perform Gaussian smoothing with σ

Calculate the x and y derivatives of the smoothed image $I_x(\sigma)$ and $I_y(\sigma)$

At each pixel, compute the matrix M,

$$M = \sum_{x,y} w(x,y)\sigma^2 \begin{bmatrix} I_x^2(\sigma) & I_x(\sigma)I_y(\sigma) \\ I_x(\sigma)I_y(\sigma) & I_y^2(\sigma) \end{bmatrix}$$

Calculate the detector response $R = \lambda_1\lambda_2 - k(\lambda_1 + \lambda_2)^2$

Detect interest points which are local maxima across both scale and space and whose response R are above a threshold.

Interest point detectors

- Harris detector calculates the response

$$R = \lambda_1\lambda_2 - k(\lambda_1 + \lambda_2)^2$$

for detecting interest points.

- There are also other mathematical operators being used, such as the Laplacian of Gaussian (LoG).
    - It means performing Gaussian smoothing first, followed by Laplacian.
    - But what is Laplacian?

Laplacian filter

- The Laplacian is the sum of second derivatives, for 2D image, it is

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

$$\begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Laplacian filter

Second derivative

- First derivative

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x + \Delta x) - f(x)}{\Delta x}$$

- Second derivative

$$f''(x) = \lim_{\Delta x \to 0} \frac{f'(x) - f'(x - \Delta x)}{\Delta x}$$

$$= \lim_{\Delta x \to 0} \frac{f(x + \Delta x) - 2f(x) + f(x - \Delta x)}{\Delta x^2}$$

1	-2	1

Laplacian filter

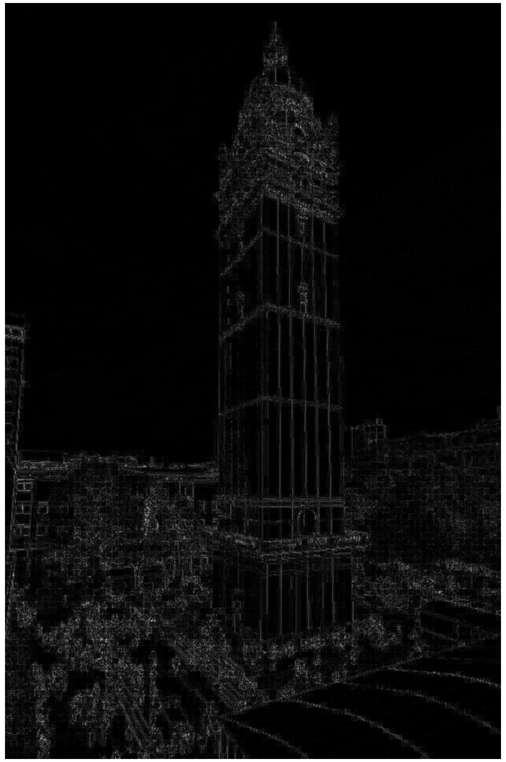
- The Laplacian is the sum of second derivatives, for 2D image, it is

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

$$\underbrace{\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 1 & -2 & 1 \\ \hline 0 & 0 & 0 \\ \hline \end{array}}_{\frac{\partial^2 f}{\partial x^2}} + \underbrace{\begin{array}{|c|c|c|}\hline 0 & 1 & 0 \\ \hline 0 & -2 & 0 \\ \hline 0 & 1 & 0 \\ \hline \end{array}}_{\frac{\partial^2 f}{\partial y^2}} = \underbrace{\begin{array}{|c|c|c|}\hline 0 & 1 & 0 \\ \hline 1 & -4 & 1 \\ \hline 0 & 1 & 0 \\ \hline \end{array}}_{\text{Laplacian filter}}$$

Laplacian filter

Input image



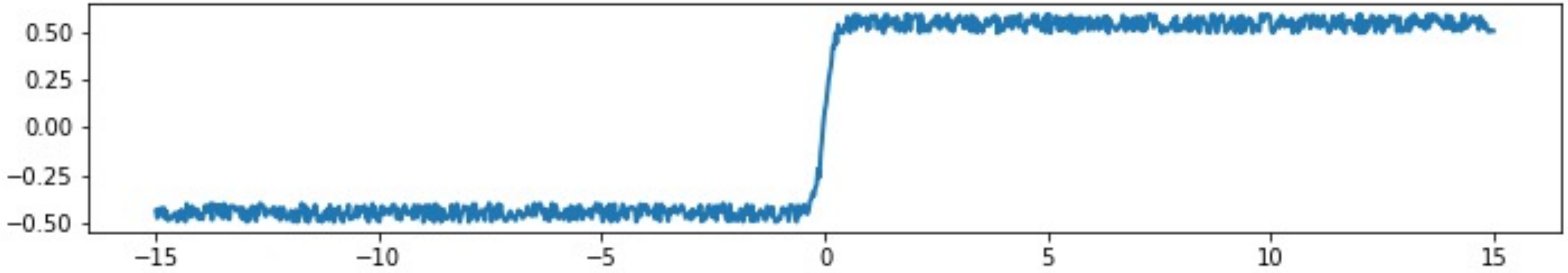
Laplacian

Second derivative

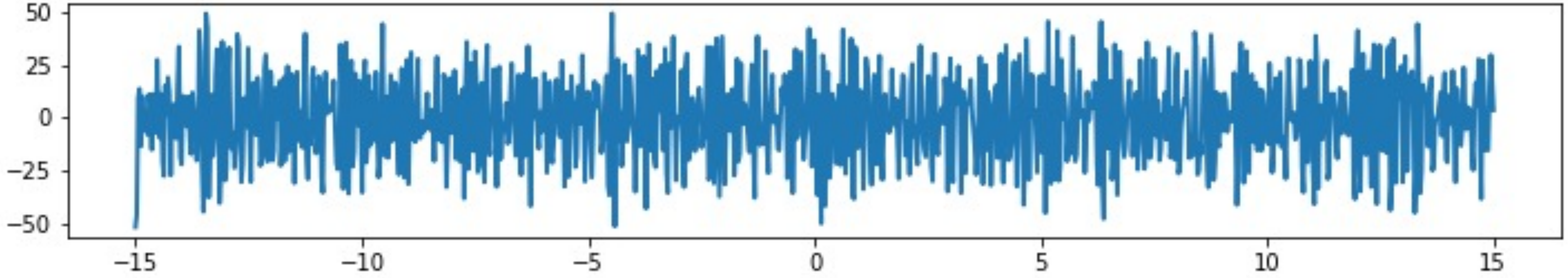
- The second derivative is even more sensitive to noise than the first derivative.

Second derivative of a noisy signal

input signal f

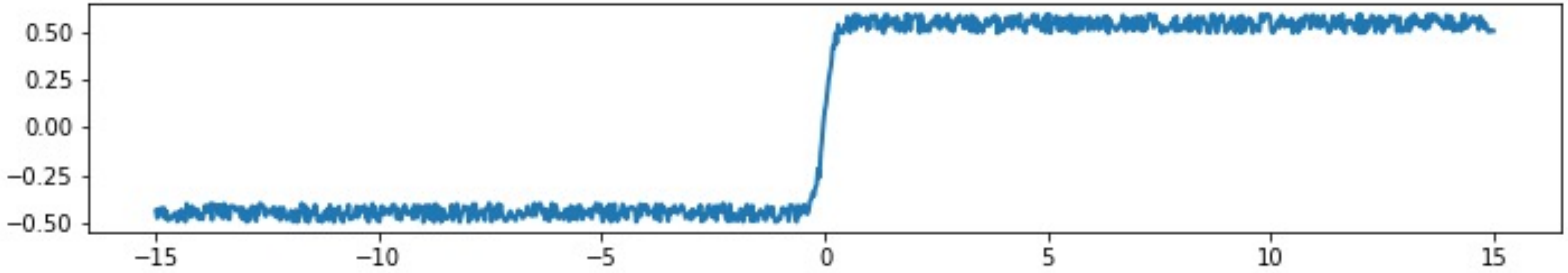


second derivative $\frac{d^2 f}{dx^2}$

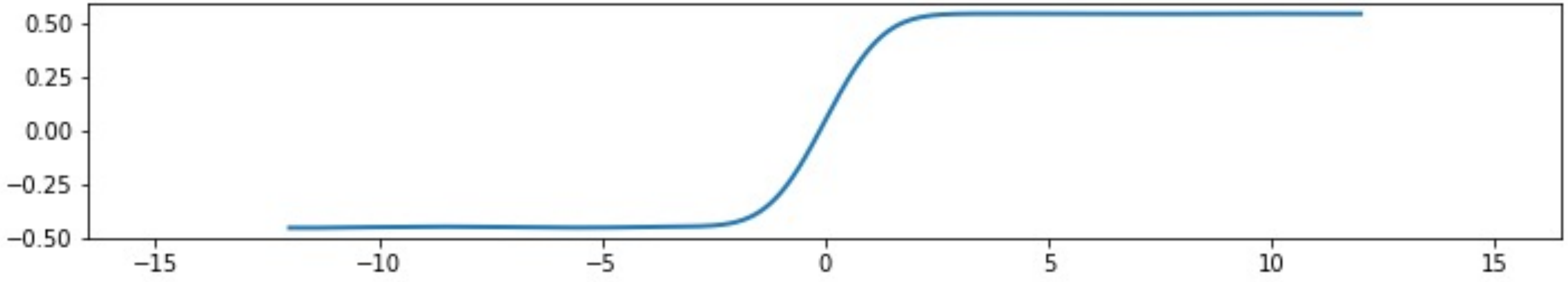


Second derivative of Gaussian smoothed signal

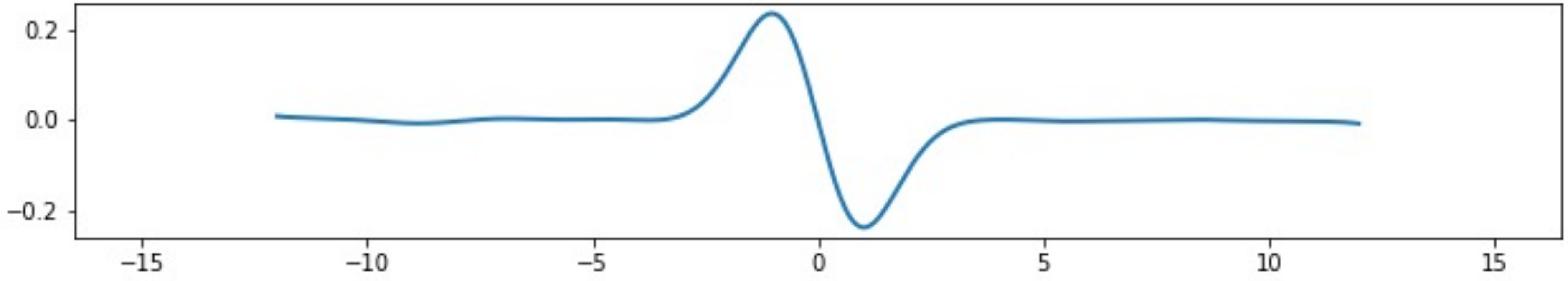
input signal f



Gaussian smoothed $f * h$



second derivative $\frac{d^2(f*h)}{dx^2}$



Laplacian of Gaussian (LoG)

- For 2D images, we can smooth the image using a Gaussian kernel before calculating the Laplacian.
- This is called the Laplacian of Gaussian filter (LoG).

$$\text{Laplacian: } \Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

$$\text{LoG: } \Delta(f * h) = \frac{\partial^2 (f * h)}{\partial x^2} + \frac{\partial^2 (f * h)}{\partial y^2}$$

Laplacian of Gaussian (LoG)

$$\frac{d}{dx}(f*h)=\frac{df}{dx}*h=f*\frac{dh}{dx}$$

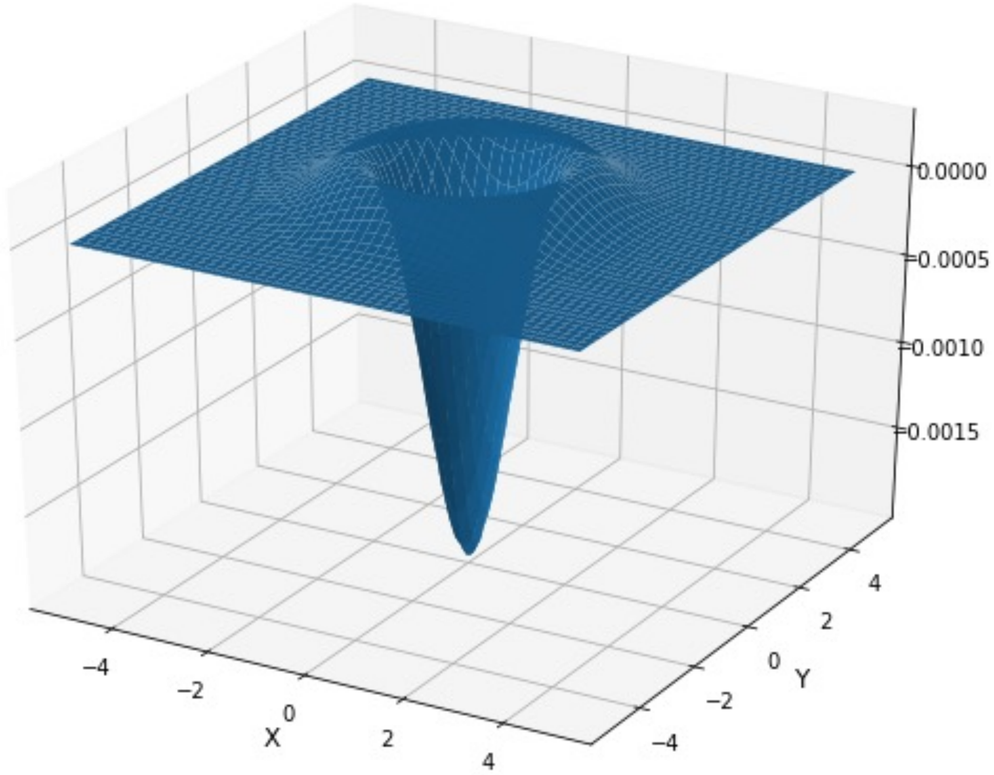
- Using the differentiation property of convolution, we can also get the analytical form for the LoG kernel

$$\text{LoG: } \Delta(f*h)=\frac{\partial^2(f*h)}{\partial x^2}+\frac{\partial^2(f*h)}{\partial y^2}=f*(\frac{\partial^2 h}{\partial x^2}+\frac{\partial^2 h}{\partial y^2})$$

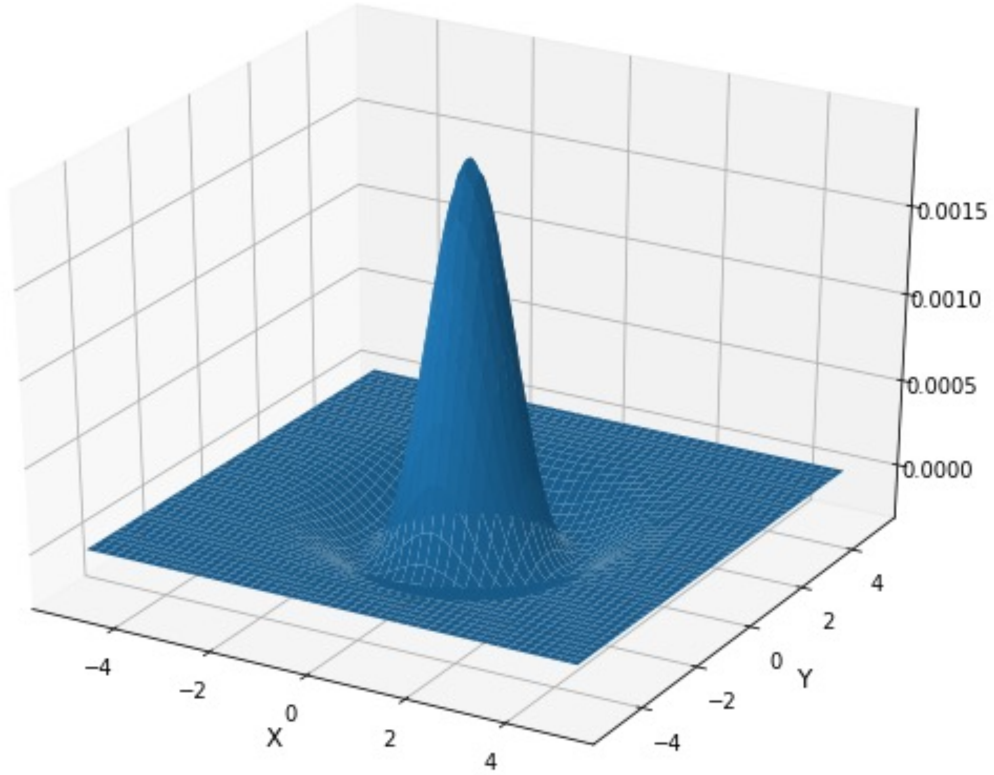
- Since 2D Gaussian is formulated as $h(x,y)=\frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}$, we can derive

$$\frac{\partial^2 h}{\partial x^2}+\frac{\partial^2 h}{\partial y^2}=-\frac{1}{\pi\sigma^4}(1-\frac{x^2+y^2}{2\sigma^2})e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Laplacian of Gaussian (LoG)



LoG filter

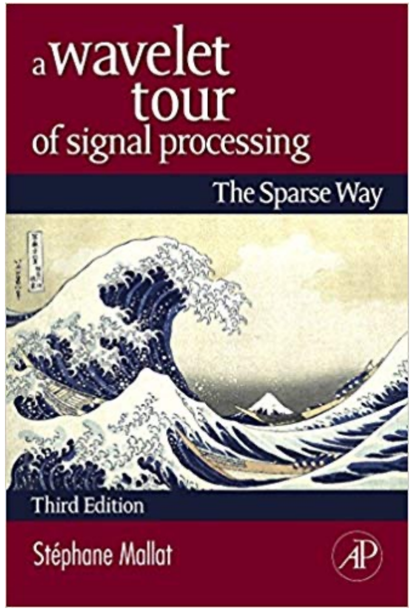


LoG filter (negative)

Another name: Mexican hat

- The negative LoG filter looks like a Mexican hat.
- Also called the Mexican hat wavelet, used in wavelet analysis to extract features from signals.
- Also called Marr's wavelet, named after David Marr.

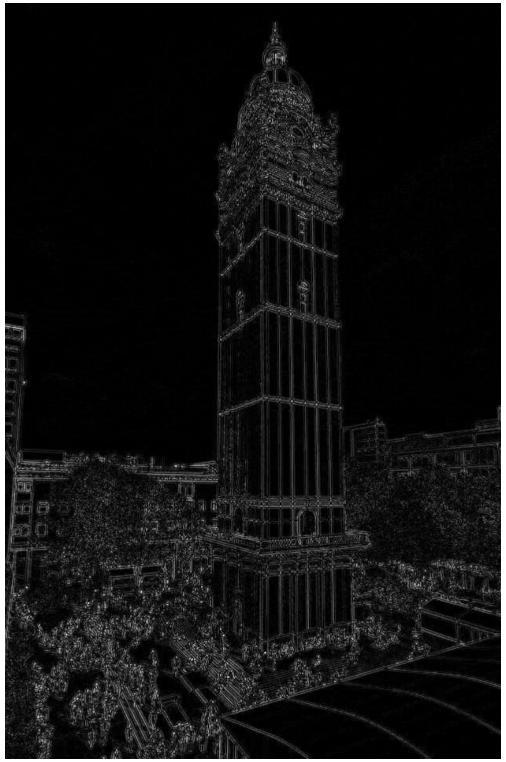
Mexican hat



Wavelet is a way to decompose and analyse signals.

Laplacian of Gaussian (LoG)

Input image



LoG

Laplacian of Gaussian (LoG)

- LoG is also a good interest point detector.
- Similarly to Harris detector, if we want to determine the optimal scale at each pixel, we need to make sure the LoG response is comparable between scales.
- The LoG response at scale σ is

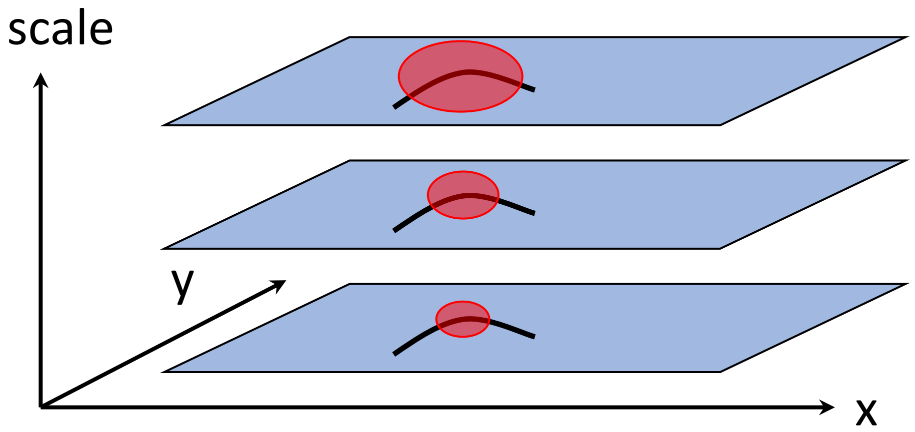
$$LoG(x, y, \sigma) = I_{xx}(x, y, \sigma) + I_{yy}(x, y, \sigma)$$

- The normalised LoG response at scale σ is

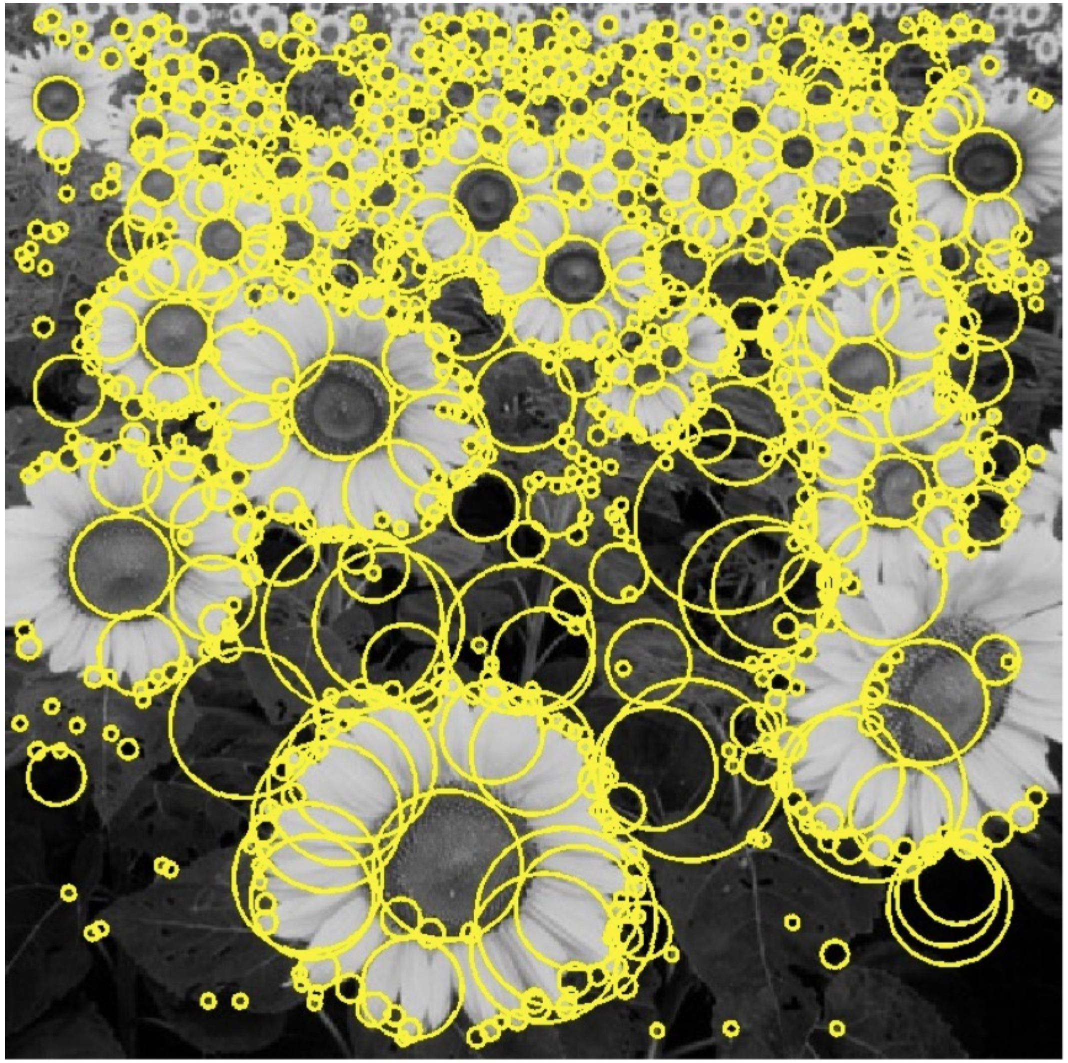
$$LoG_{norm}(x, y, \sigma) = \underbrace{\sigma^2}_{\text{normaliser}}(\underbrace{I_{xx}(x, y, \sigma) + I_{yy}(x, y, \sigma)}_{\text{Gaussian filter followed by Laplacian filter}})$$

Laplacian of Gaussian (LoG)

- The interest points are detected as local extrema across both scale and space and above a threshold.



Laplacian of Gaussian response at different scales



Detected interest points, each denoted using a circle representing its scale.

Difference of Gaussian (DoG)

- Difference of Gaussians (DoG) filter is defined as,

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

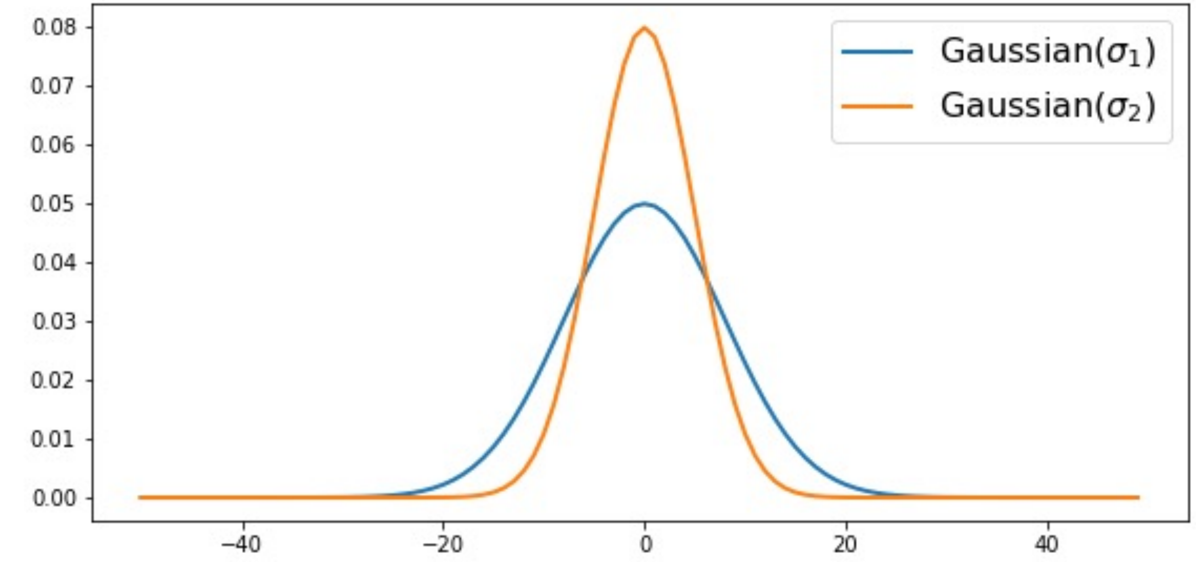
Gaussian filters with different scales

- In Lowe's paper, he suggested $k=\sqrt{2}$.

- DoG approximates the normalised Laplacian of Gaussian (LoG),

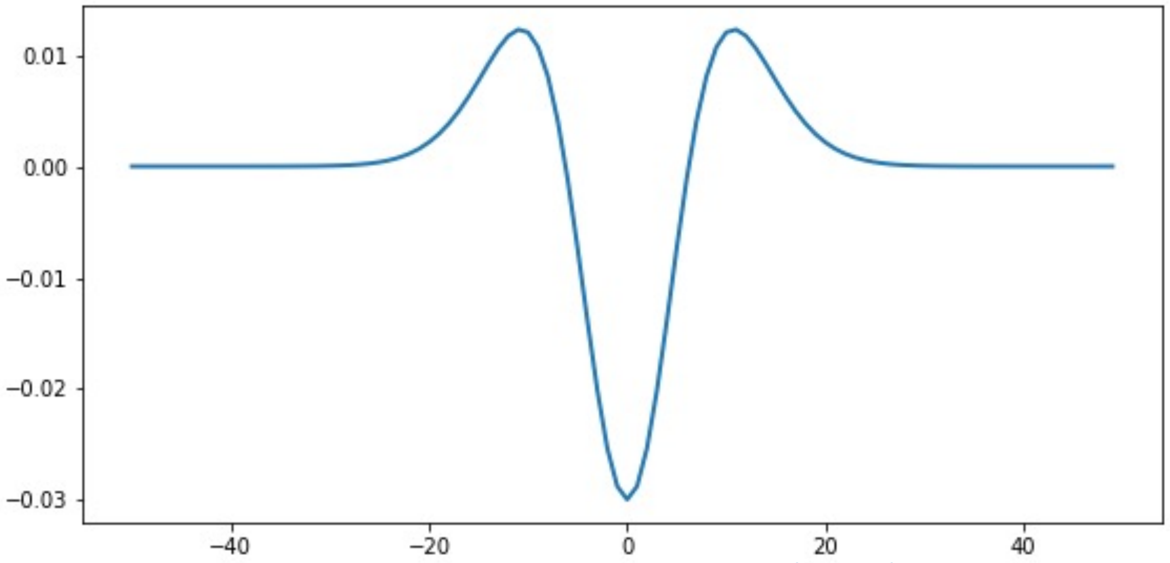
$$LoG_{norm}(x, y, \sigma) = \sigma^2 (I_{xx}(x, y, \sigma) + I_{yy}(x, y, \sigma))$$

D. Lowe. Distinctive image features from scale-invariant keypoints, IJCV 2004.

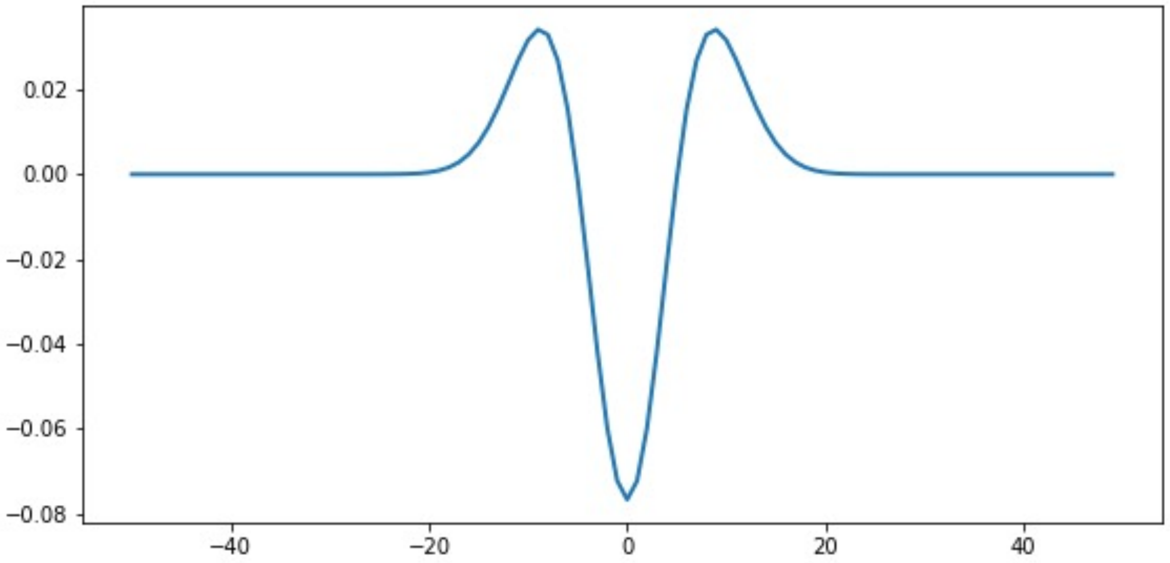


Two Gaussian filters with different scales

- DoG is a good approximation to LoG.
- $DoG(x, y, \sigma) \approx (k-1)\sigma^2\nabla^2 G(x, y, \sigma)$ as shown in Lowe's paper.



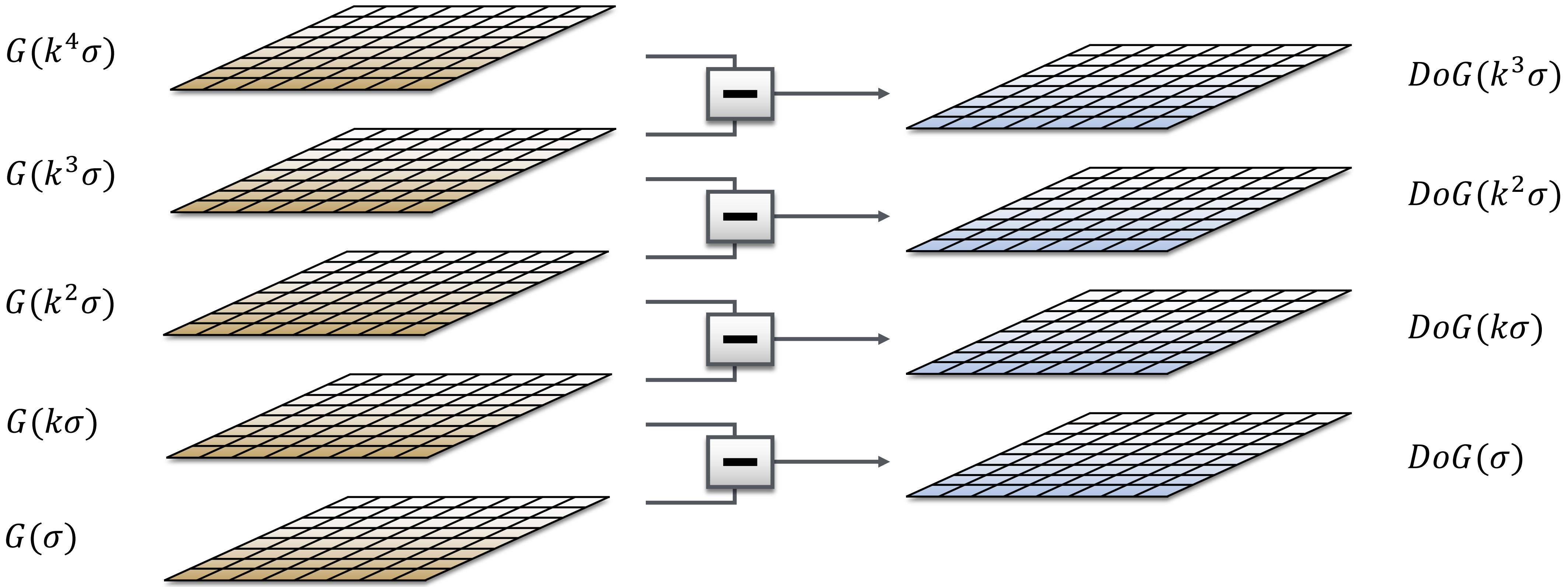
Difference of Gaussians (DoG)



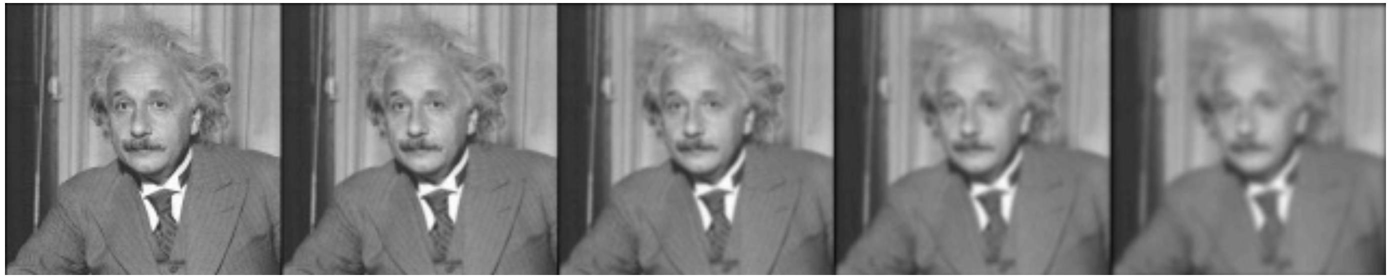
Laplacian of Gaussian (LoG)

Difference of Gaussian (DoG)

- DoG is a good approximation to the normalised Laplacian of Gaussian (LoG).
- It provides some convenience in calculating the response across different scales.



Gaussian | Difference of Gaussians (DoG)



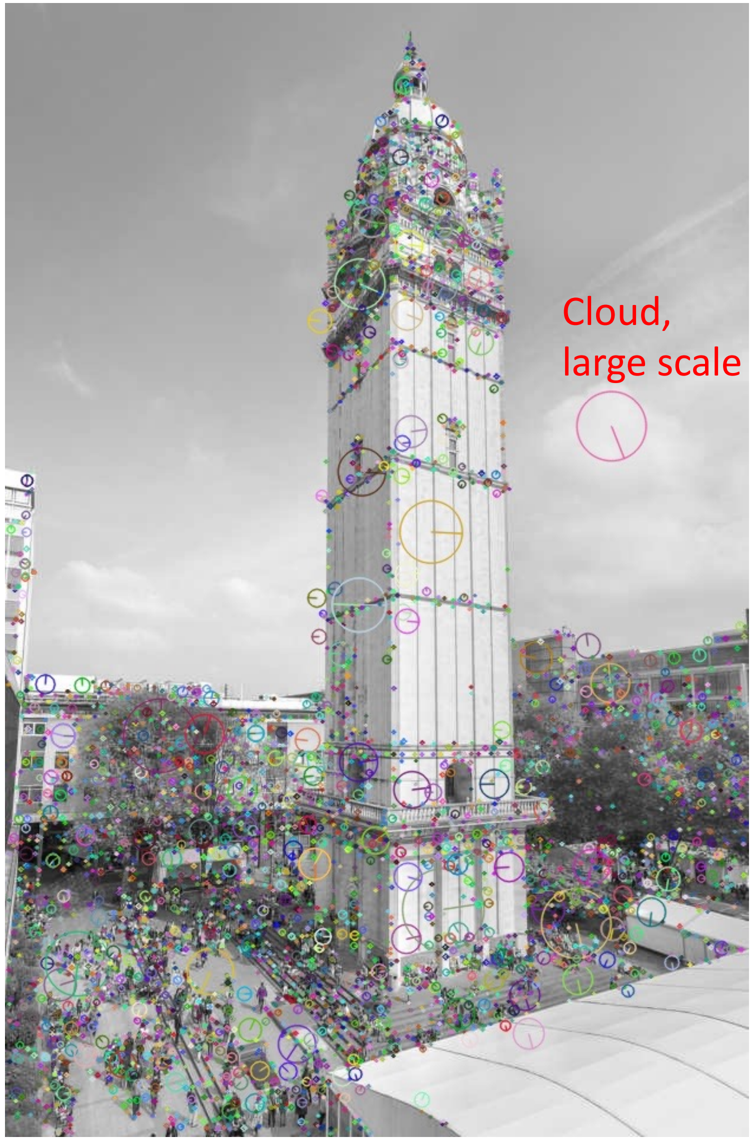
$G(\sigma)$ | $G(k\sigma)$ | $G(k^2\sigma)$ | $G(k^3\sigma)$ | $G(k^4\sigma)$

$DoG(\sigma)$ | $DoG(k\sigma)$ | $DoG(k^2\sigma)$ | $DoG(k^3\sigma)$

Difference of Gaussian (DoG)

- DoG filters are used in one of the most popular algorithms, called SIFT, which is a pipeline for detecting and describing interest points.

Image

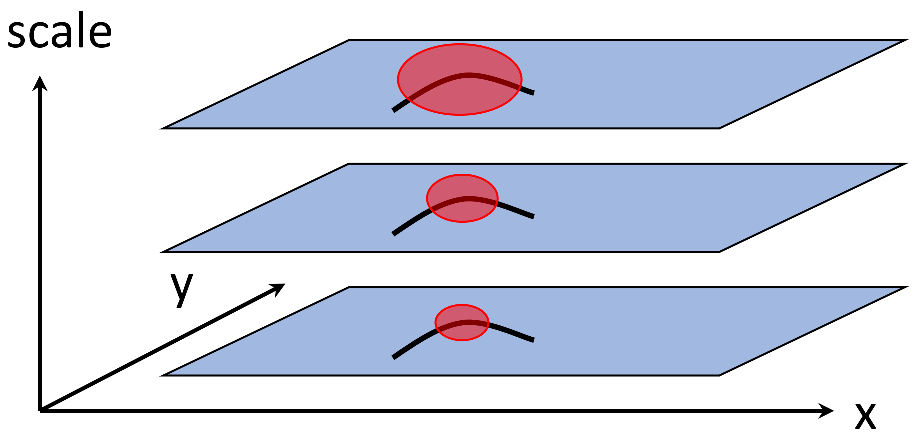


Interest points with associated scales

Interest point detectors

- Scale-variant
  - Harris detector
- Scale-invariant
  - Scale adapted Harris detector
  - Normalised Laplacian of Gaussian
  - Difference of Gaussian
- The scale-invariant detectors follow similar procedures.
  - Calculate the detector response across scales.
  - Find local extrema both across scale and across space.

Interest point detectors



Detector response at different scales

Detector	Response
Scale adapted Harris detector	$\lambda_1\lambda_2 - k(\lambda_1 + \lambda_2)^2$
Normalised Laplacian of Gaussian	$\sigma^2(I_{xx}(x, y, \sigma) + I_{yy}(x, y, \sigma))$
Difference of Gaussian	$I * G(k\sigma) - I * G(\sigma)$

Note: calculation of λ_1, λ_2 depends on σ.

Next

- Now we can detect interest points in images, with a scale associated with each interest point.
- Suppose the subsequent task is image matching.
- How do we know an interest point in one image correspond to another interest point in another image?
- We need to describe the features for the interest points.

References

- Sec. 4.1.1 Feature detectors. Richard Szeliski, Computer Vision: Algorithms and Applications (http://szeliski.org/Book).