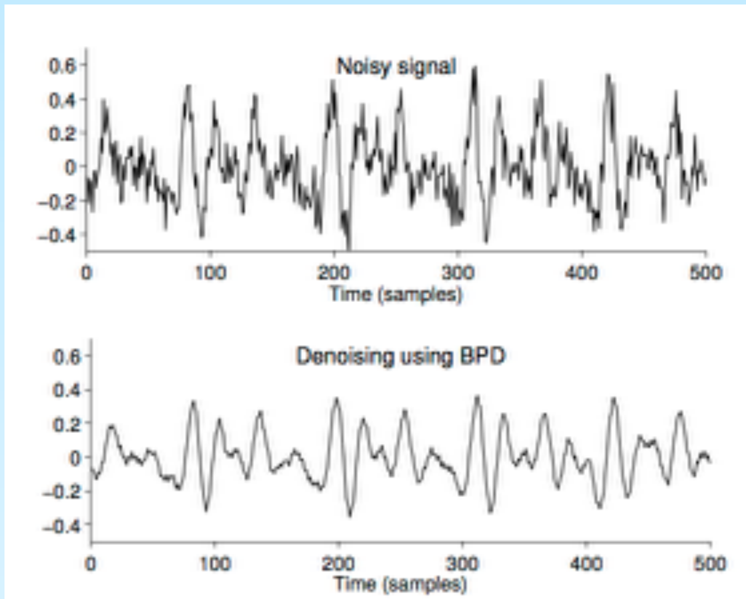
Image Processing

by Dominic Waithe
Weatherall Institute of Molecular Medicine

- Introduction to Image Processing
- Recap image formation
- Linear and non-linear filters
- Edge filters
- Filter banks
- Applied examples
- Deconvolution
- Conclusion

What is image processing



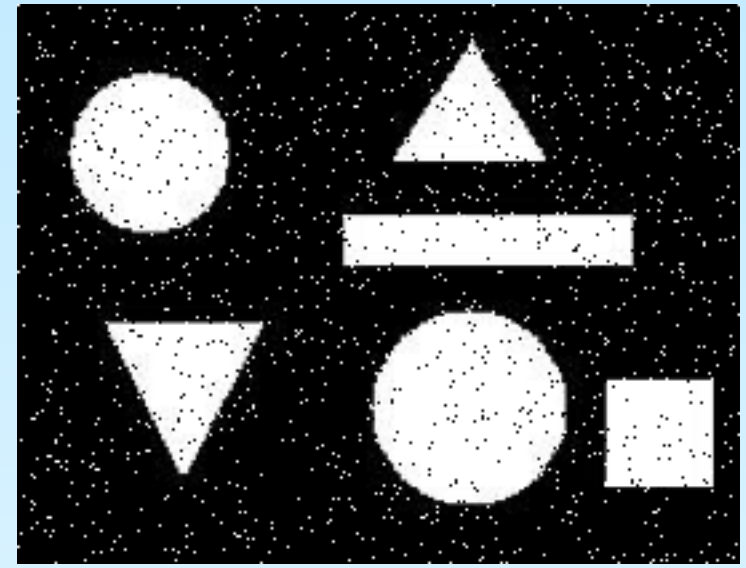


Image Processing usually involves a 2-D or 3-D images but has its roots in Signal Processing.

Signal processing is the science of transferring information contained in many different physical, symbolic, or abstract formats.

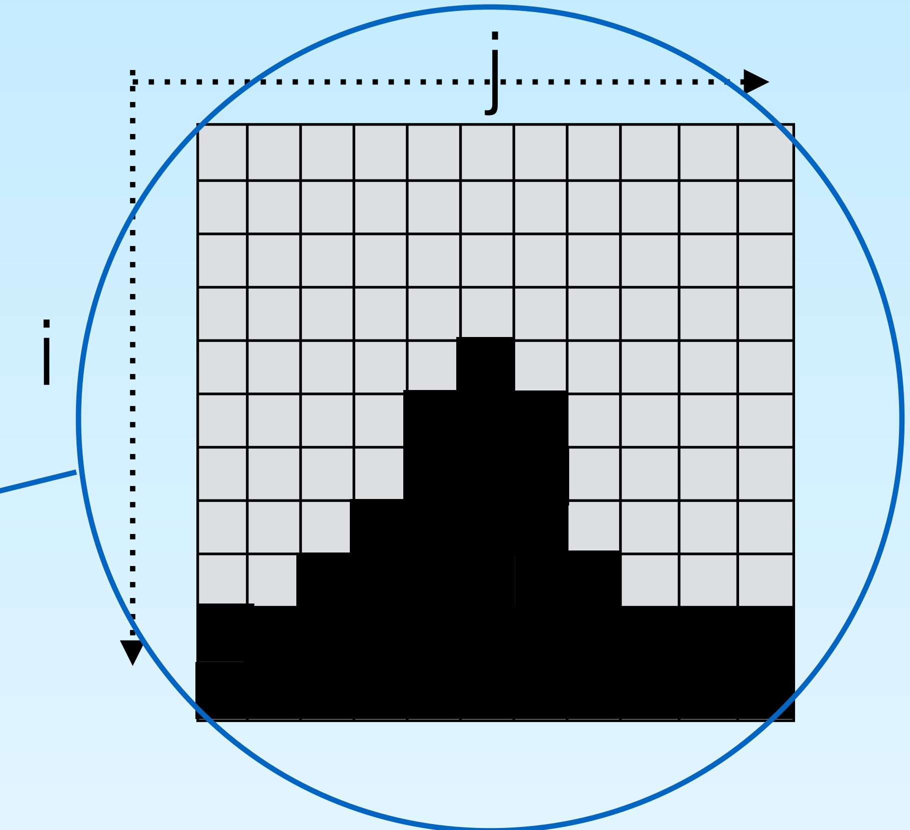
Usually we are trying to extract some type of meaning from our data, we process it to declutter our signal or focus on a particular aspect.

Source: http://eeweb.poly.edu/iselesni/lecture_notes/sparsity_intro/bpd.png
http://photos1.blogger.com/blogger/82/1614/1600/fig1.2.jpg

Image representation recap: Binary Image

Computer images are 2D arrays of numbers:

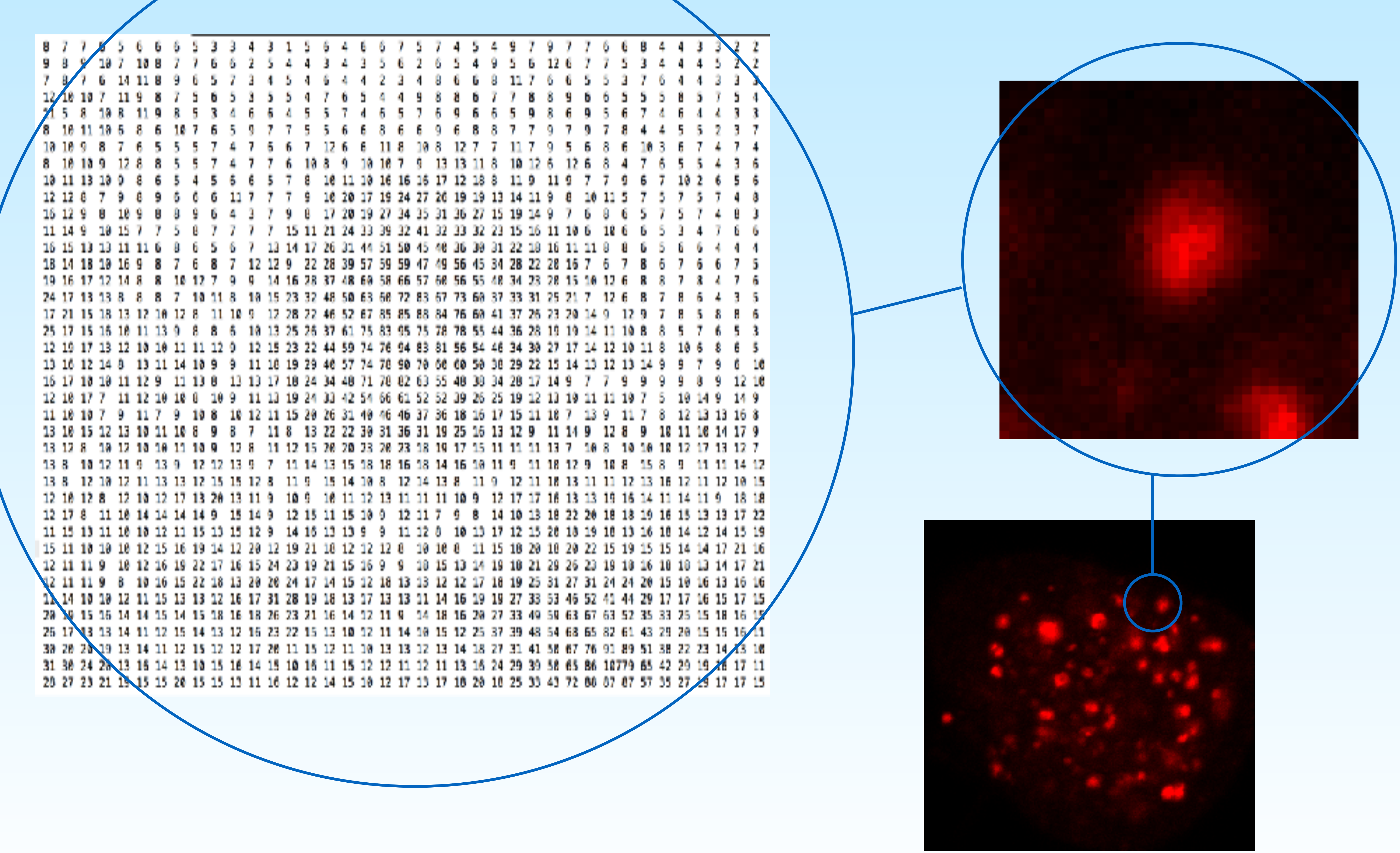
```
0,0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,1,0,0,0,0,0
0,0,0,0,1,1,1,0,0,0,0
0,0,0,0,1,1,1,0,0,0,0
0,0,0,0,1,1,1,0,0,0,0
0,0,0,1,1,1,1,0,0,0,0
0,0,1,1,1,1,1,1,0,0,0
1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1
```



Source: http://www.uff.br/cdme/matrix/matrix-html/matrix_boolean/matrix_boolean_en.html

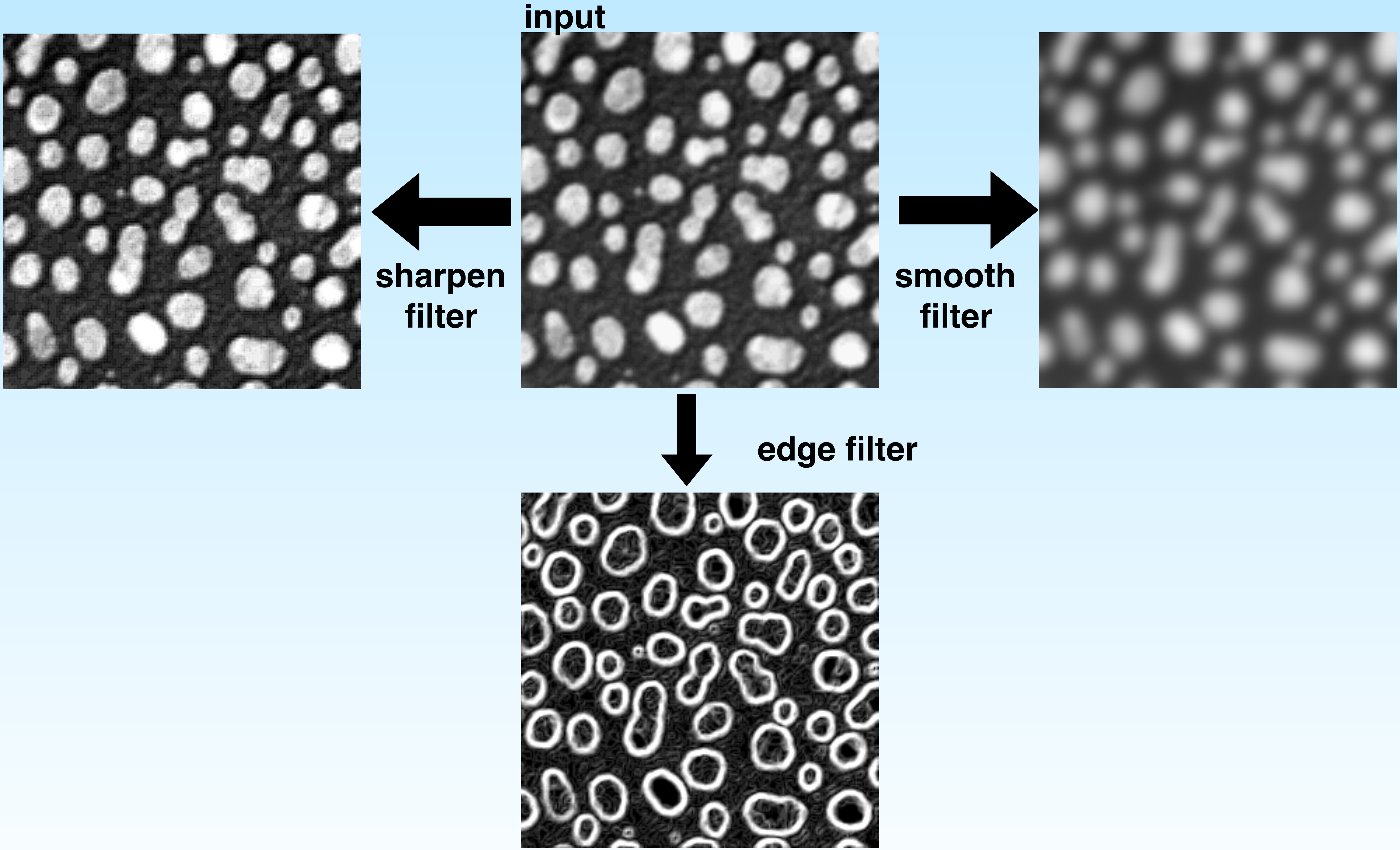
Image representation recap: Grayscale Image

Computer images are 2D arrays of numbers:



Source: http://www.uff.br/cdme/matrix/matrix-html/matrix_boolean/matrix_boolean_en.html

Image processing: Image Filtering



Linear and non-linear filtering.

Linear Filters

- mean filter
- Gaussian filter
- Sobel filter
- Prewitt filter
- Haar-like filter
- Gabor filter
- Difference of Gaussians

non-linear filters

- median filter
- percentile filter
- maximum filter
- minimum filter

(addition) $\bar{L}(f+g)=\bar{L}f+\bar{L}g$

(scalar multiplication) $\bar{L}(tf)=t\bar{L}f.$

Definition of a linear operator:
pair of functions **f** and **g**
(for example two images)
and scalar **s** and operator **L**

Convolution - Is a linear filter.

continuous convolution operator

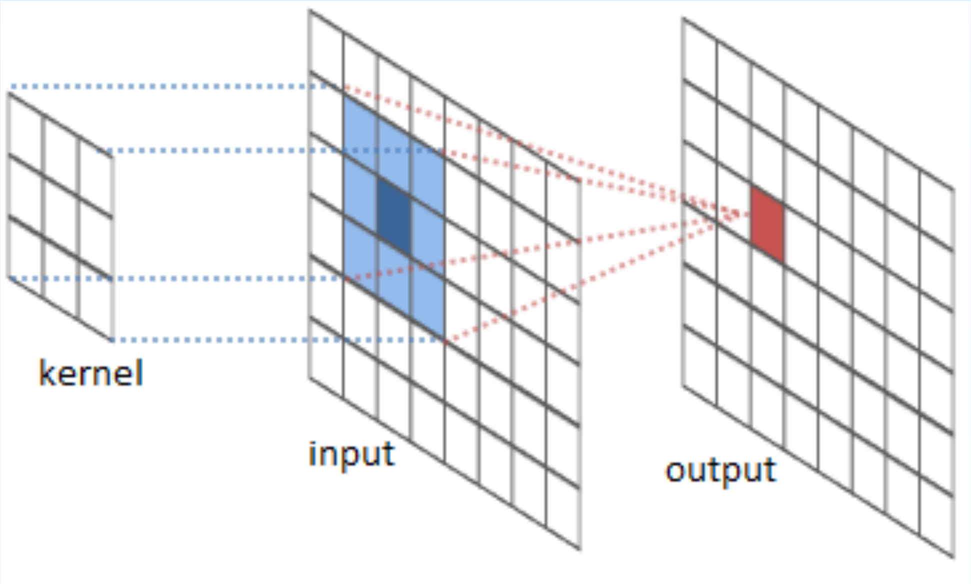
$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$$

discrete convolution operator

$$w(x, y) \star f(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t) f(x+s, y+t)$$

$$1/9 \times \boxed{\begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}}$$

kernel: 3x3 mean filter

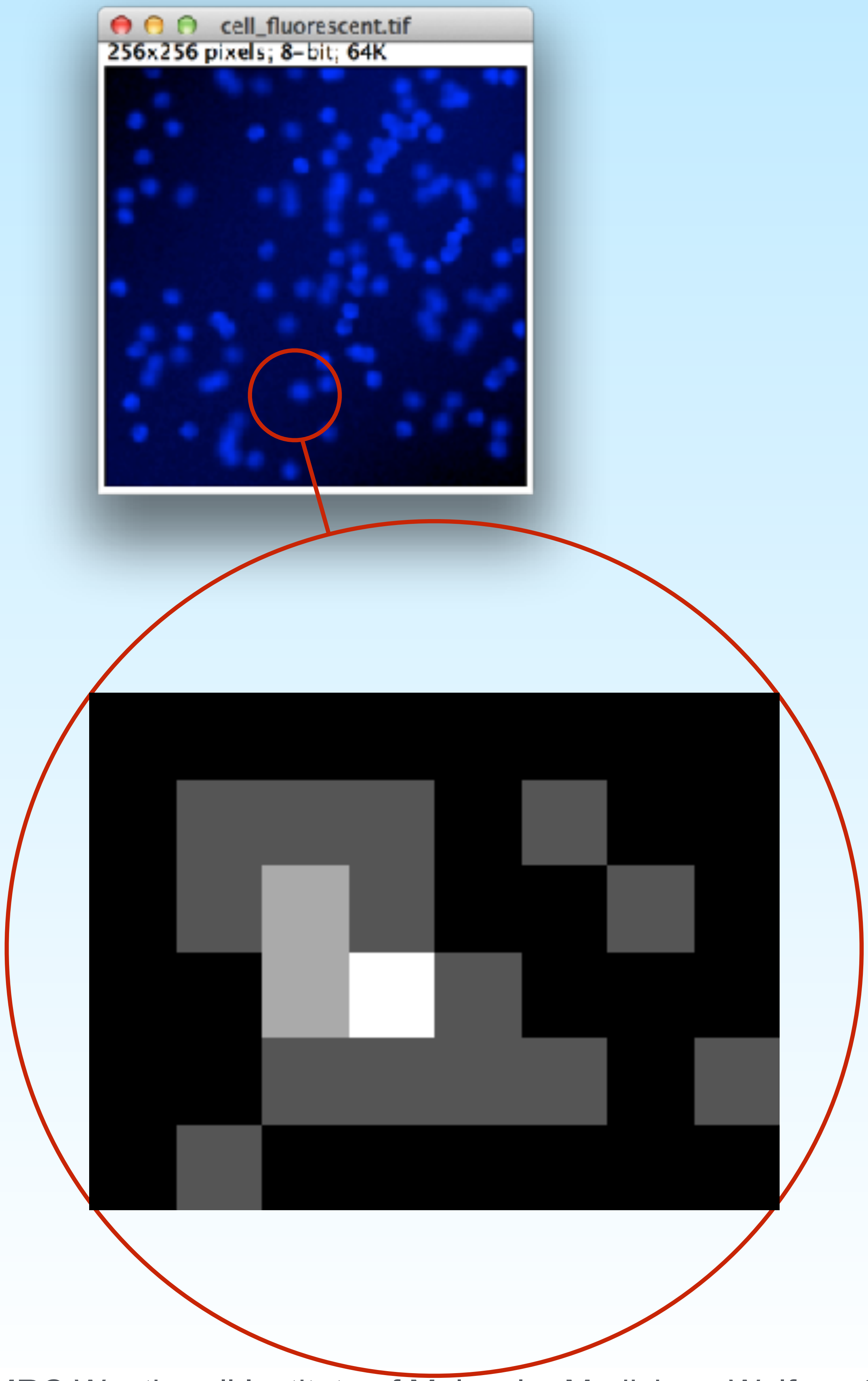


$$\left.\begin{matrix} (a_{-1\,-1} * b_{-1\,-1}) + \\ (a_{-1\,0} * b_{-1\,0}) + \\ (a_{-1\,1} * b_{-1\,1}) + \\ (a_{0\,-1} * b_{0\,-1}) + \\ (a_{0\,0} * b_{0\,0}) + \end{matrix}\right. = \text{output}$$

etc, etc

Convolution is the weighted sum of the pixel neighbourhood.
The kernel defines the weights.

Linear filter: Mean filter



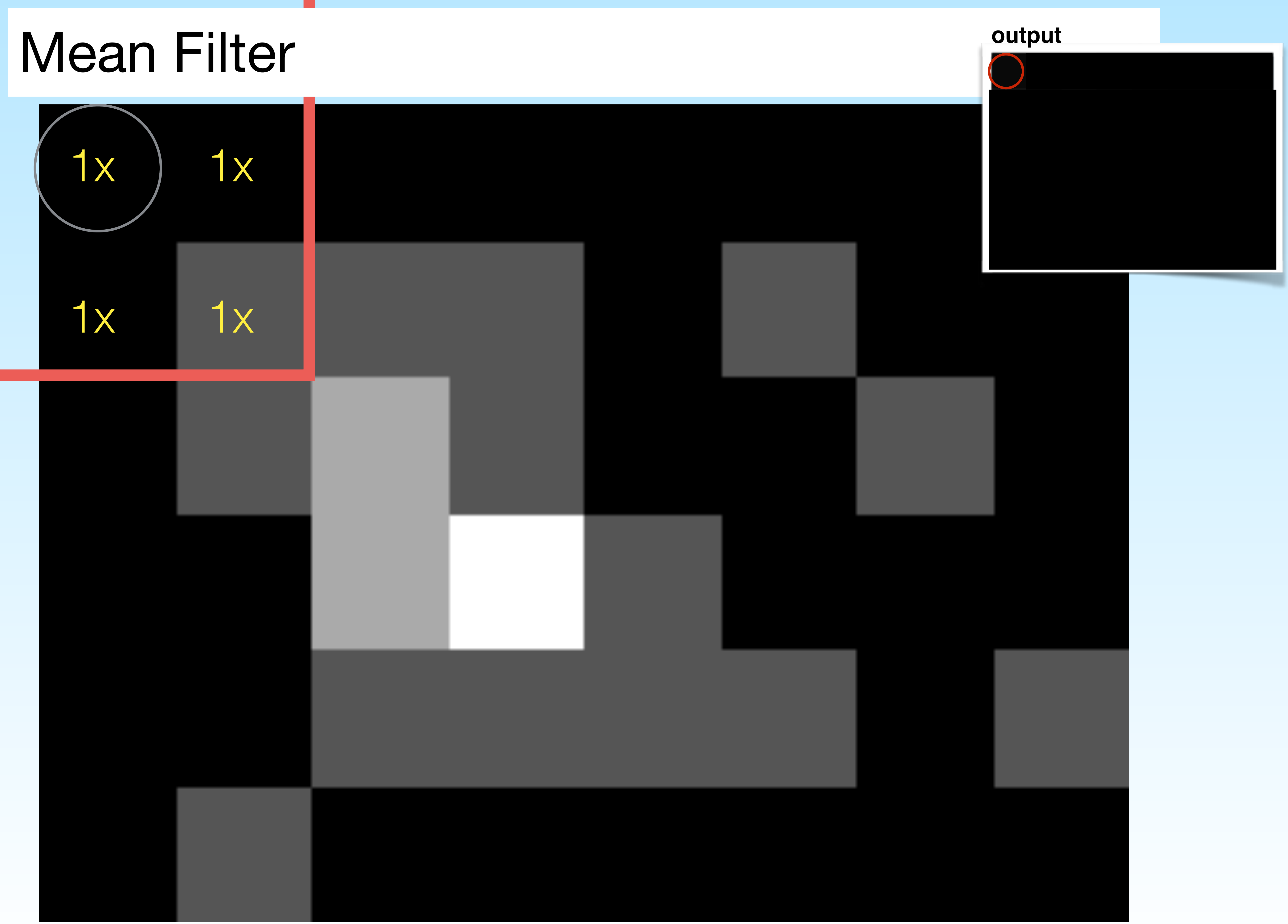
$$\begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}$$

kernel: 3x3 mean filter

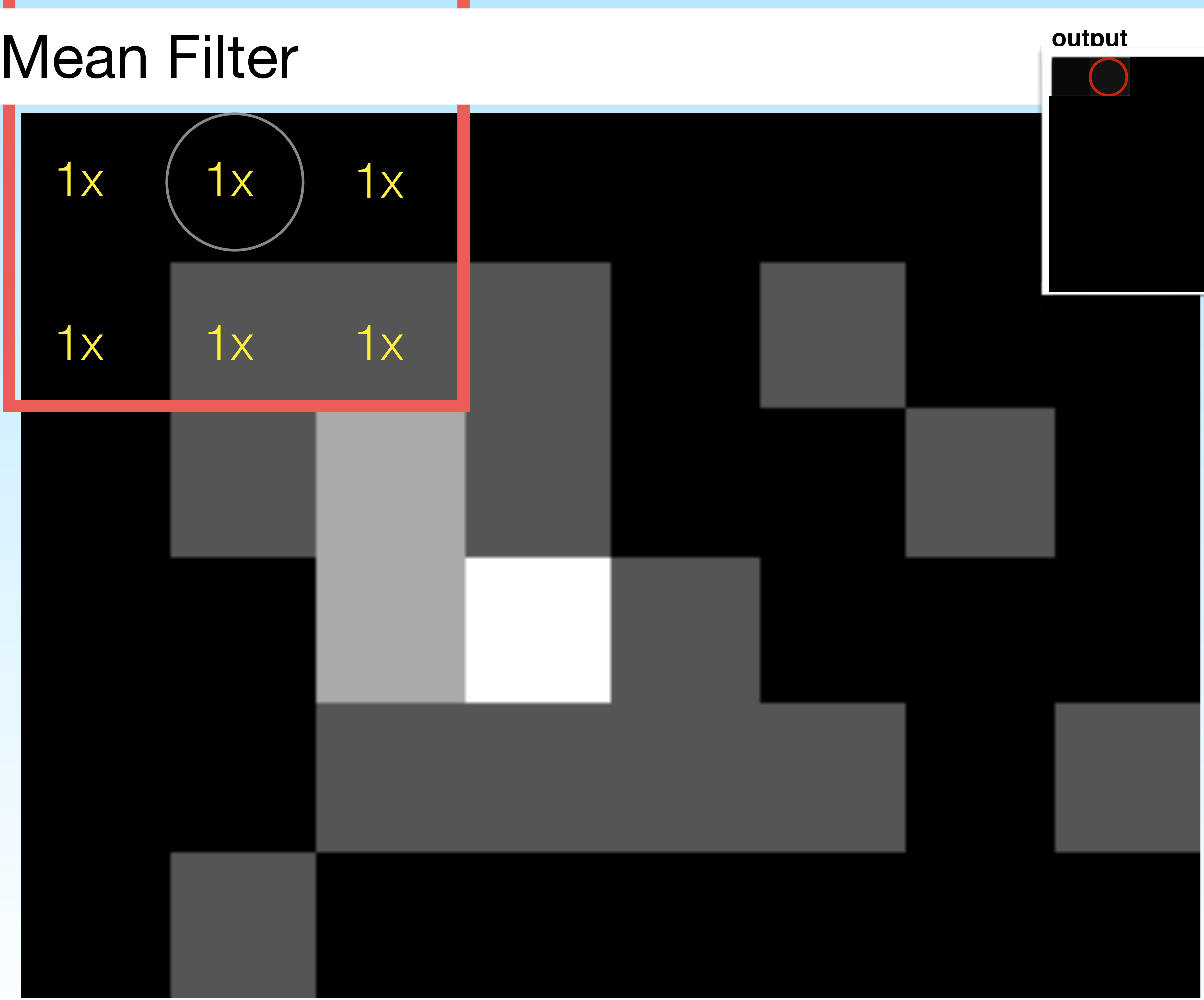
[[1, 1, 1, 1, 1, 1, 1, 1],
[1, 2, 2, 2, 1, 2, 1, 1],
[1, 2, 3, 2, 1, 1, 2, 1],
[1, 1, 3, 4, 2, 1, 1, 1],
[1, 1, 2, 2, 2, 2, 1, 2],
[1, 2, 1, 1, 1, 1, 1, 1]]

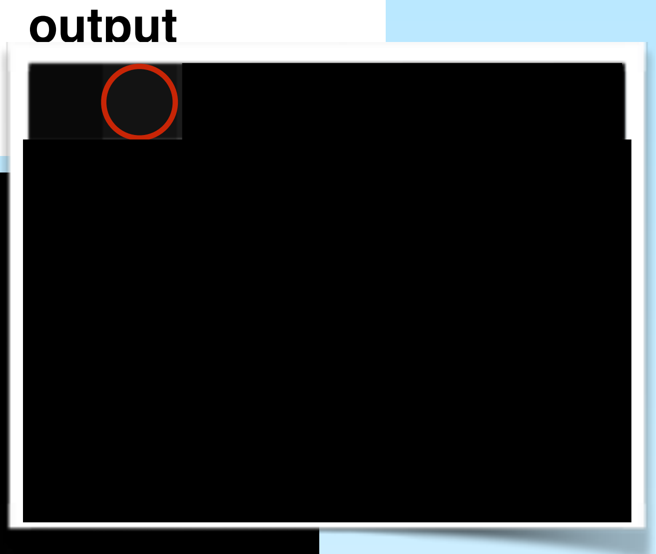
Image region

Mean Filter

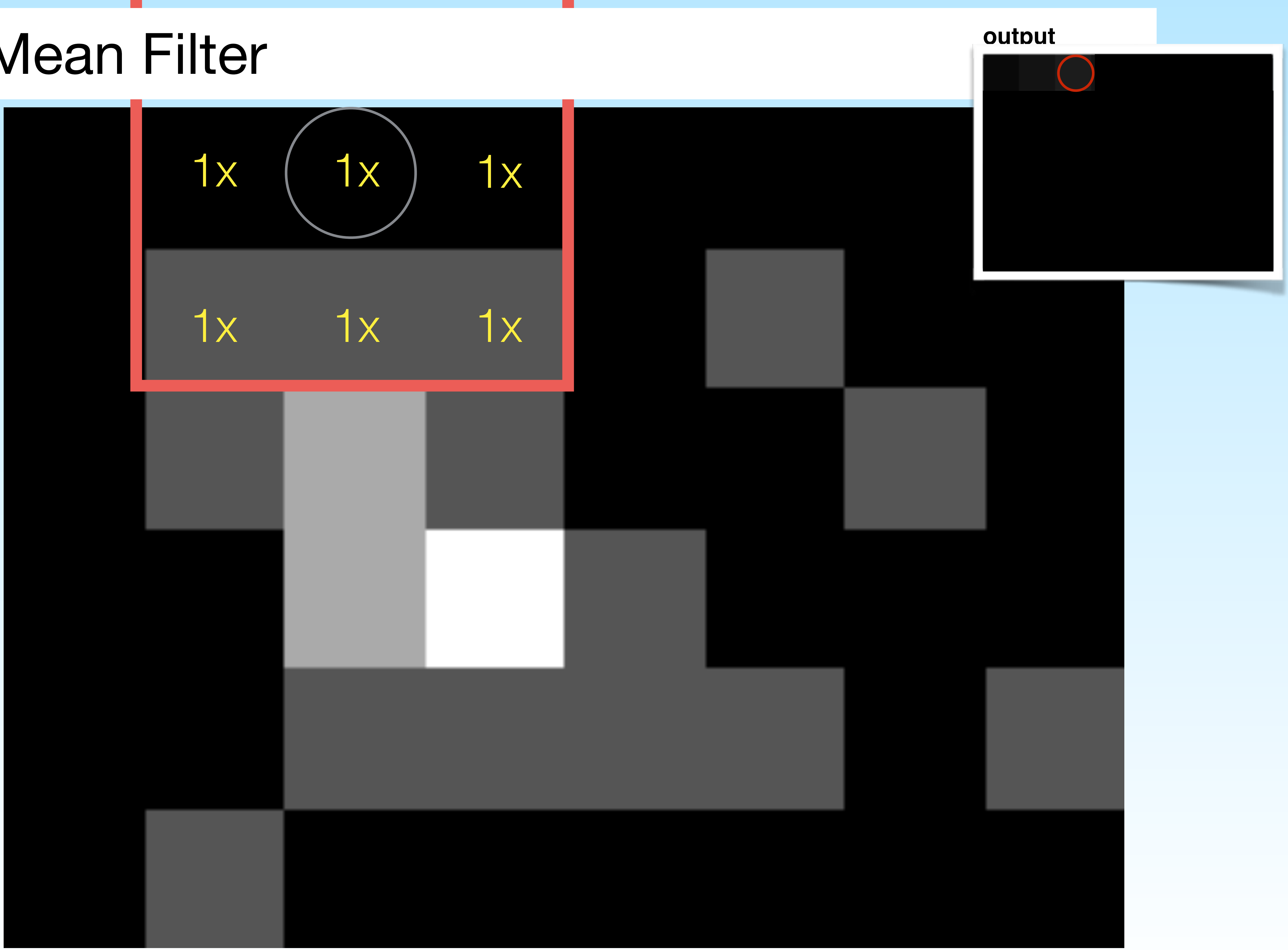


Mean Filter

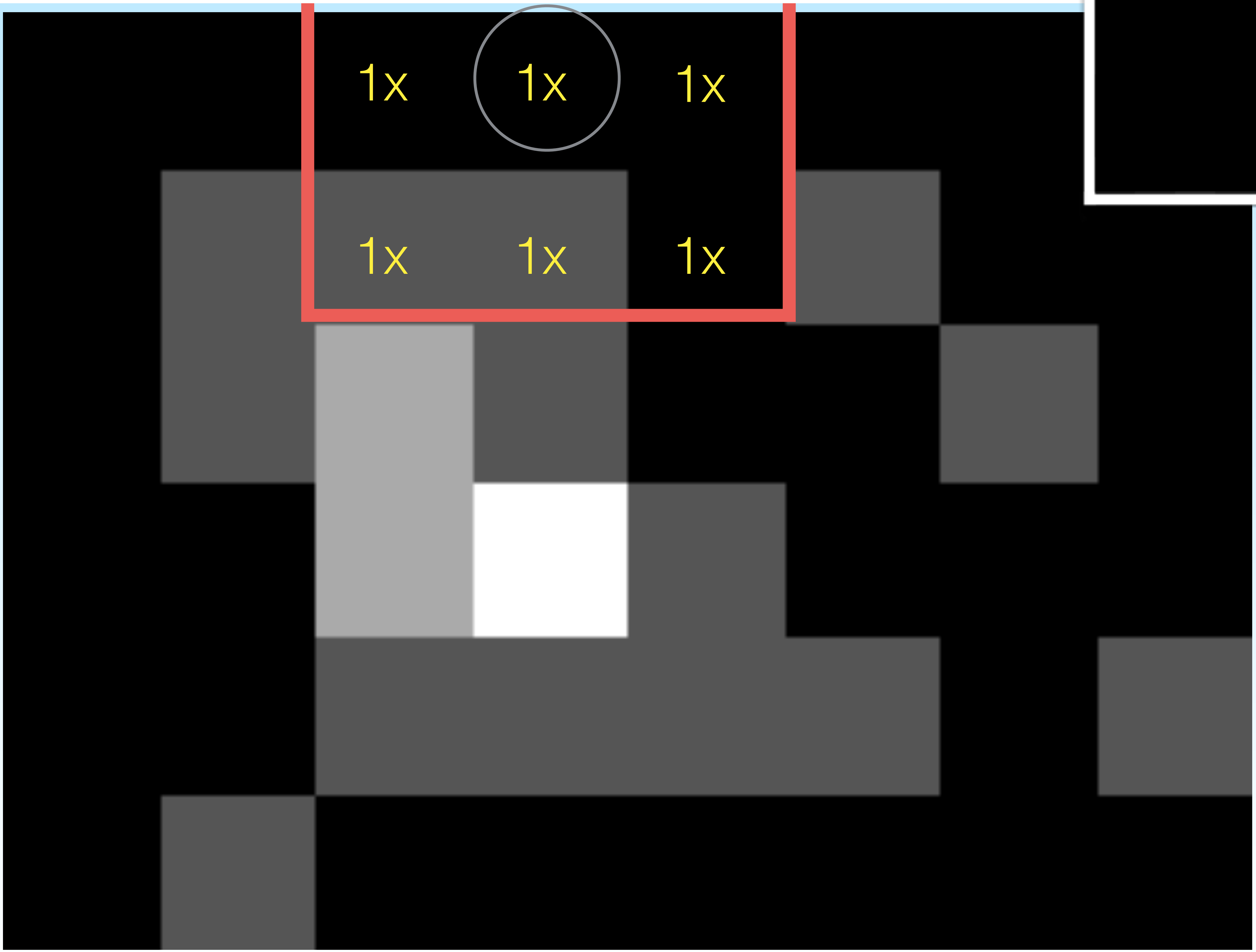


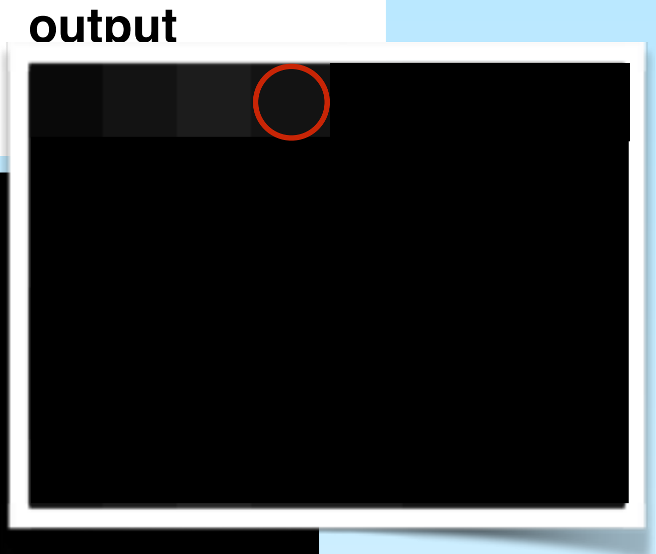


Mean Filter

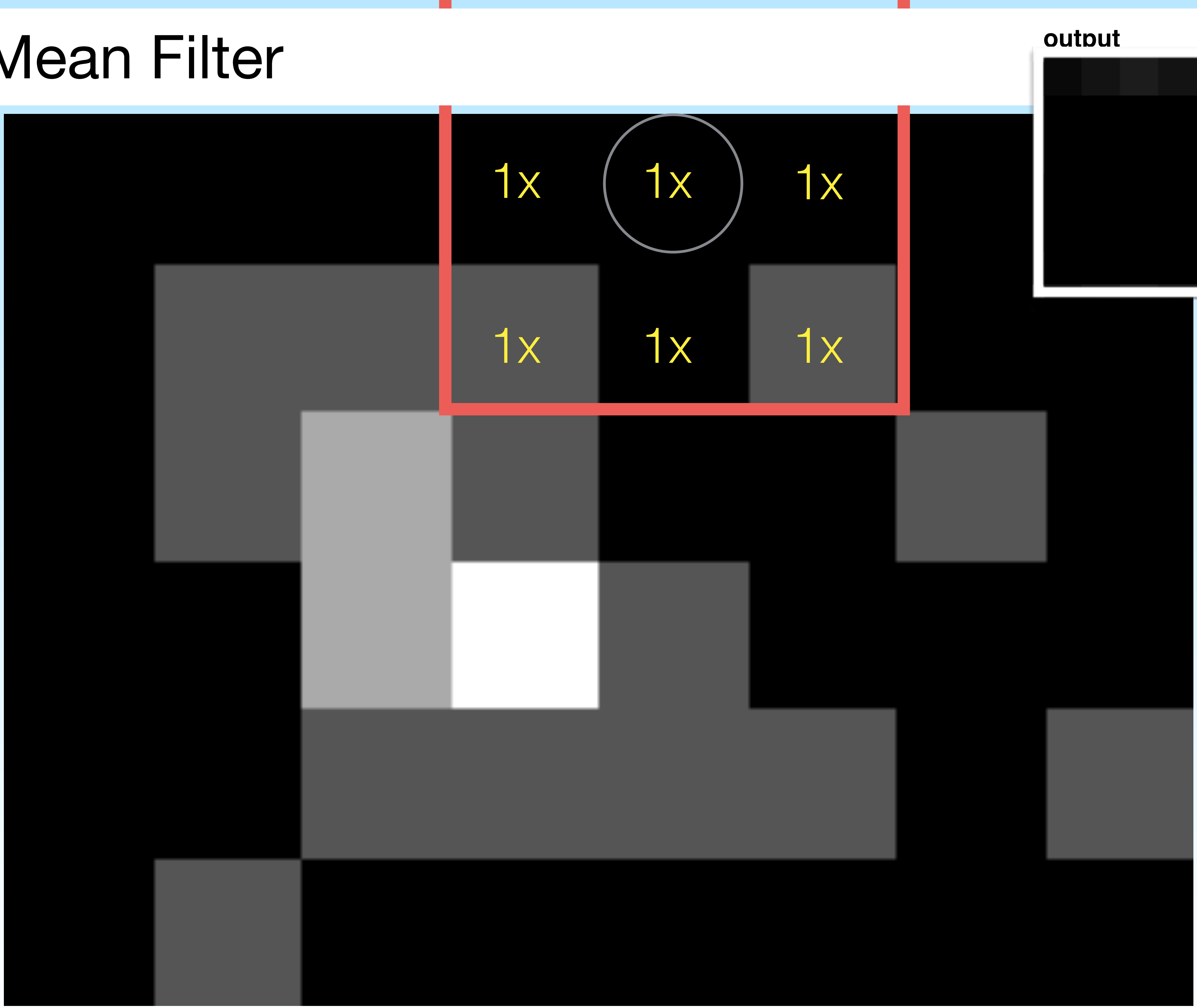


Mean Filter

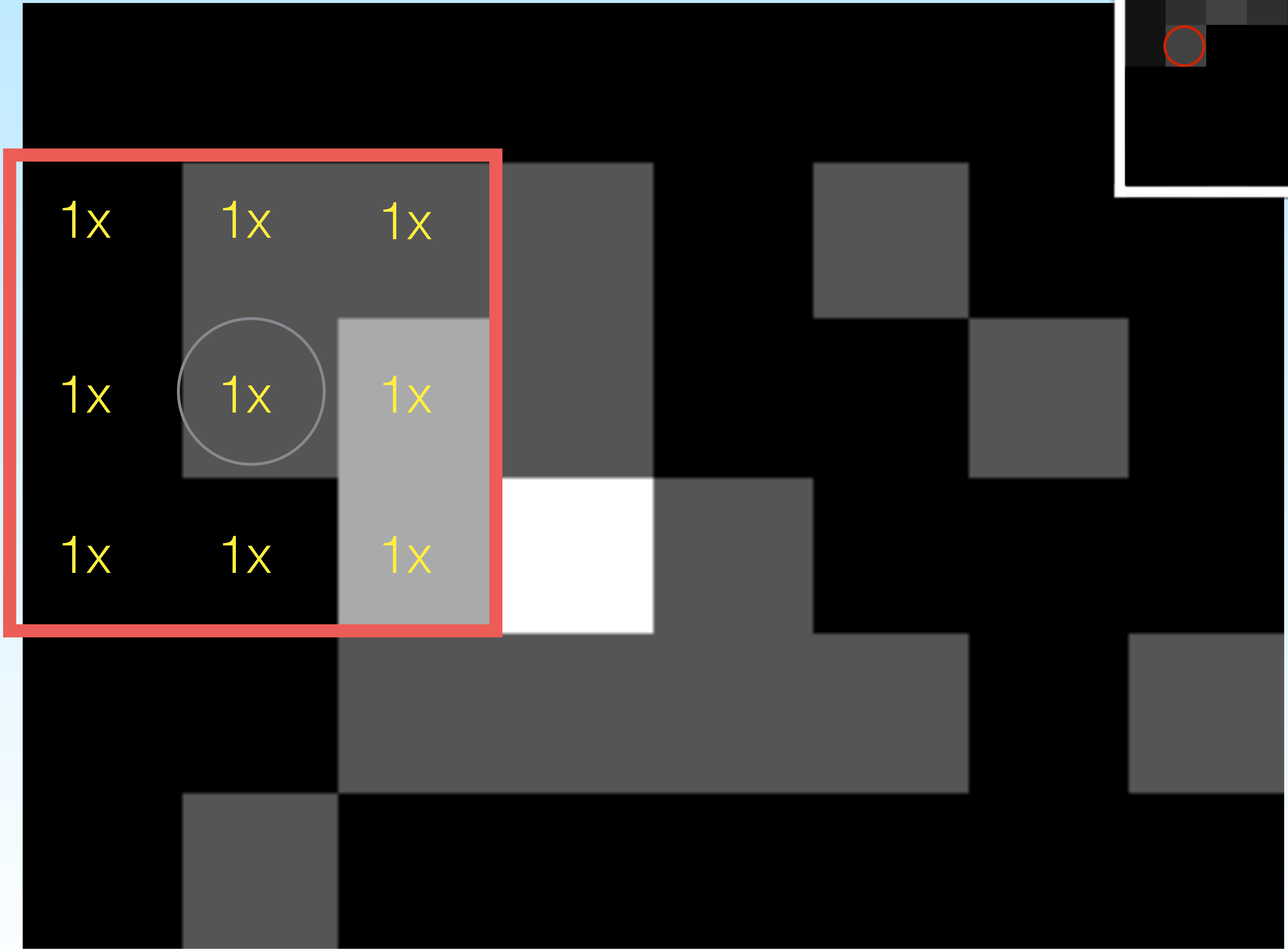


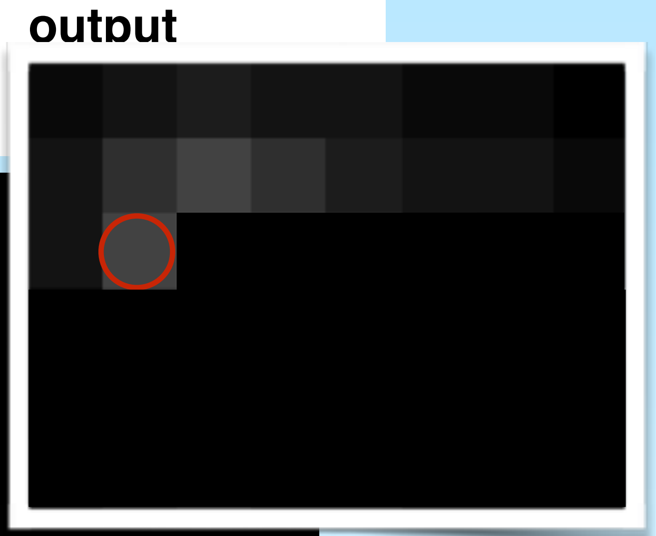


Mean Filter

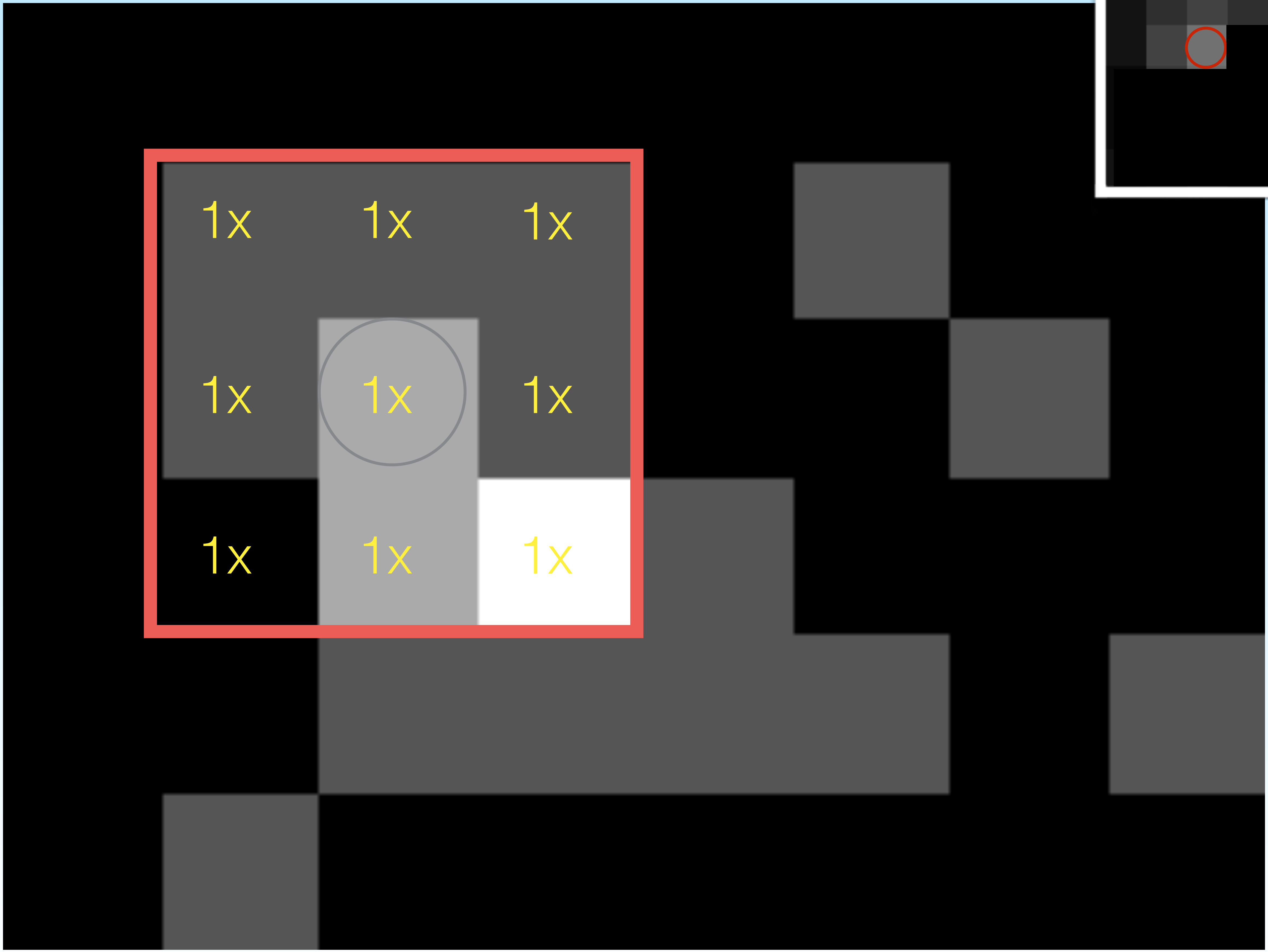


Mean Filter

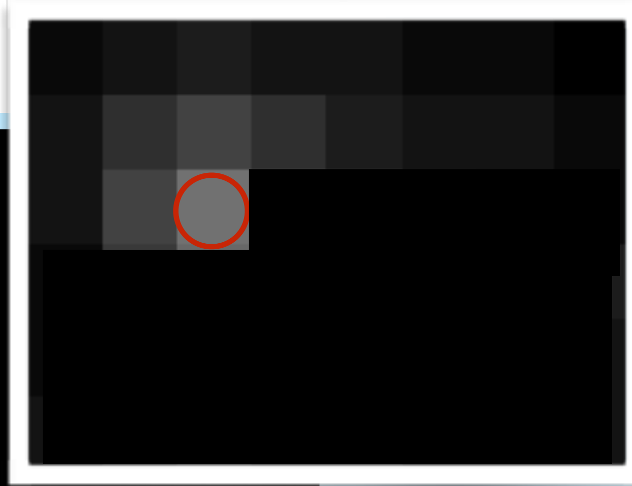




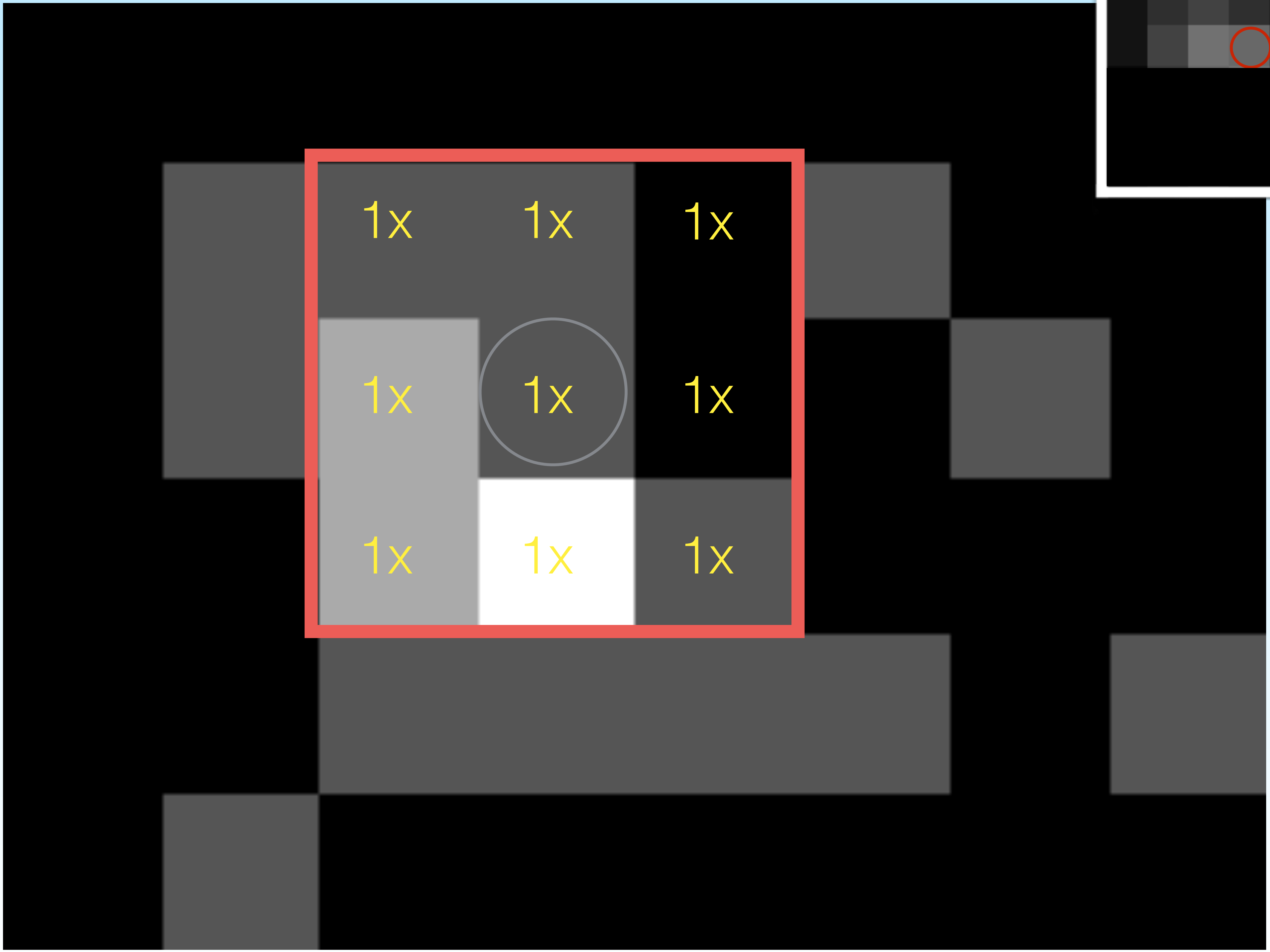
Mean Filter

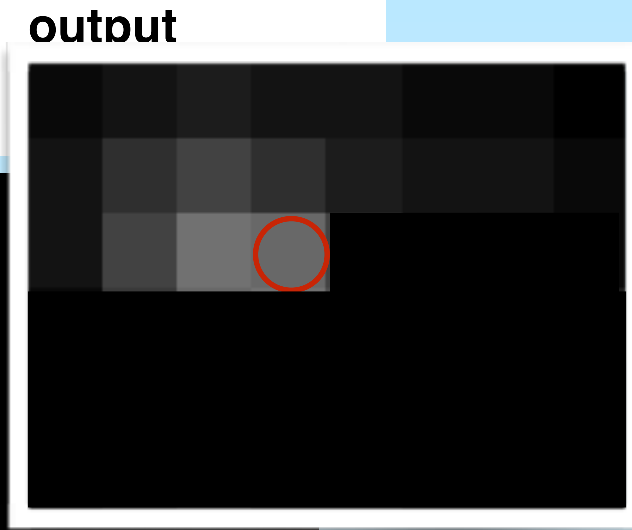


output

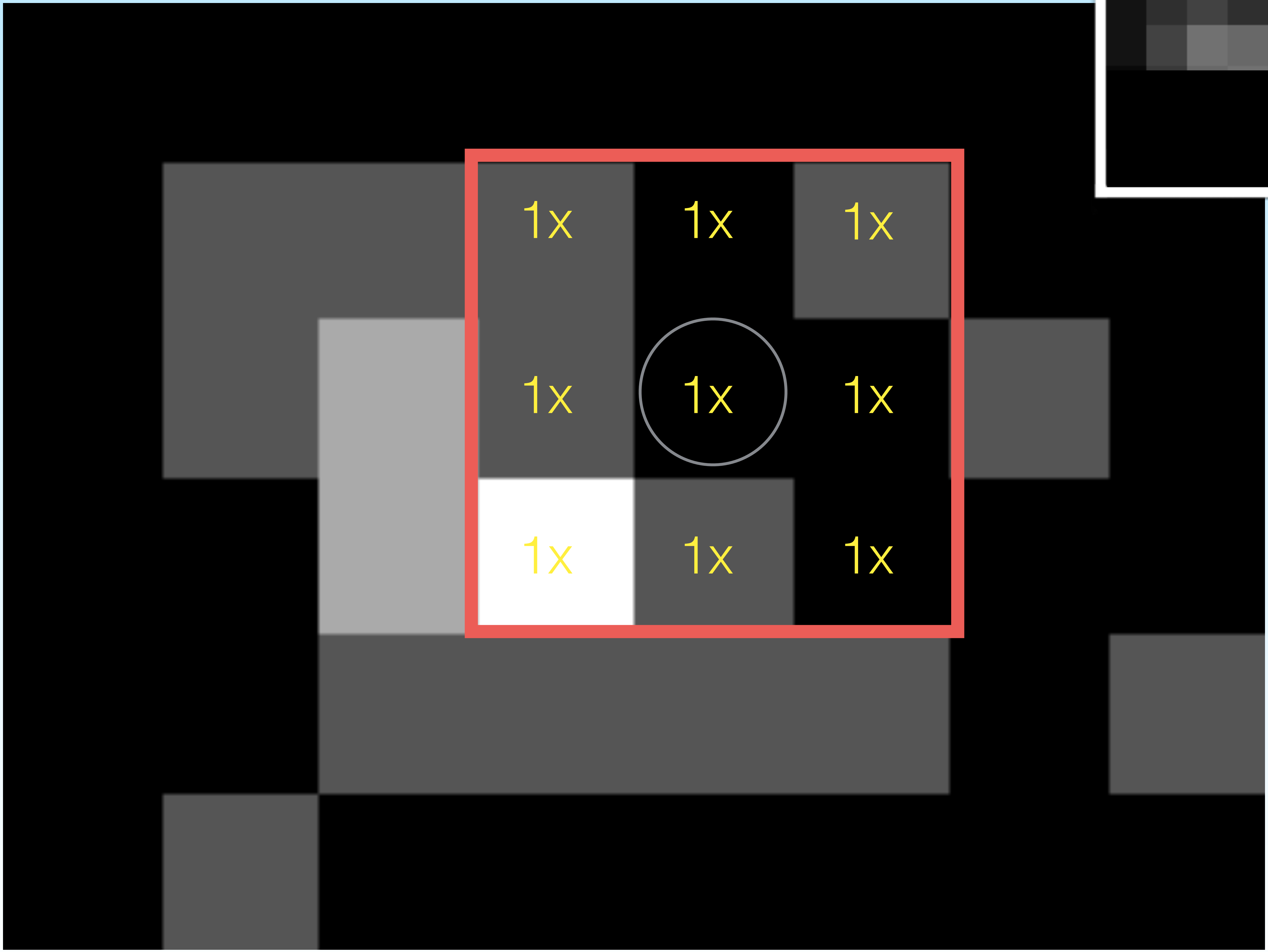


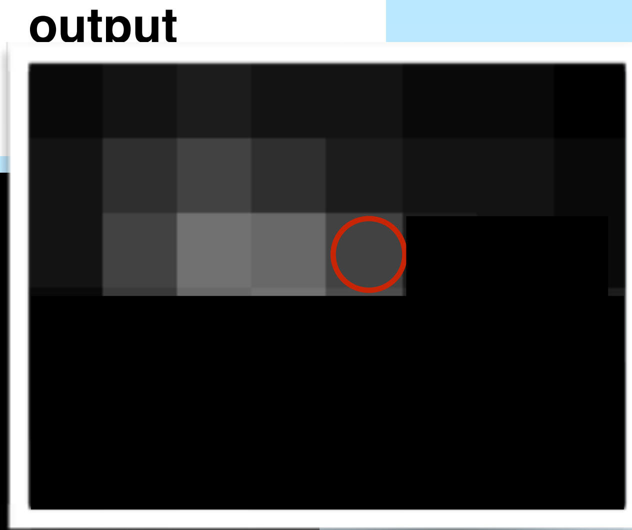
Mean Filter



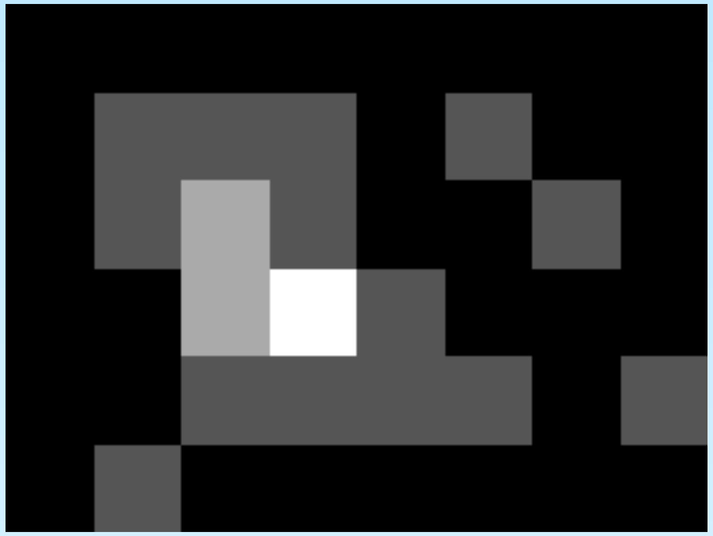


Mean Filter

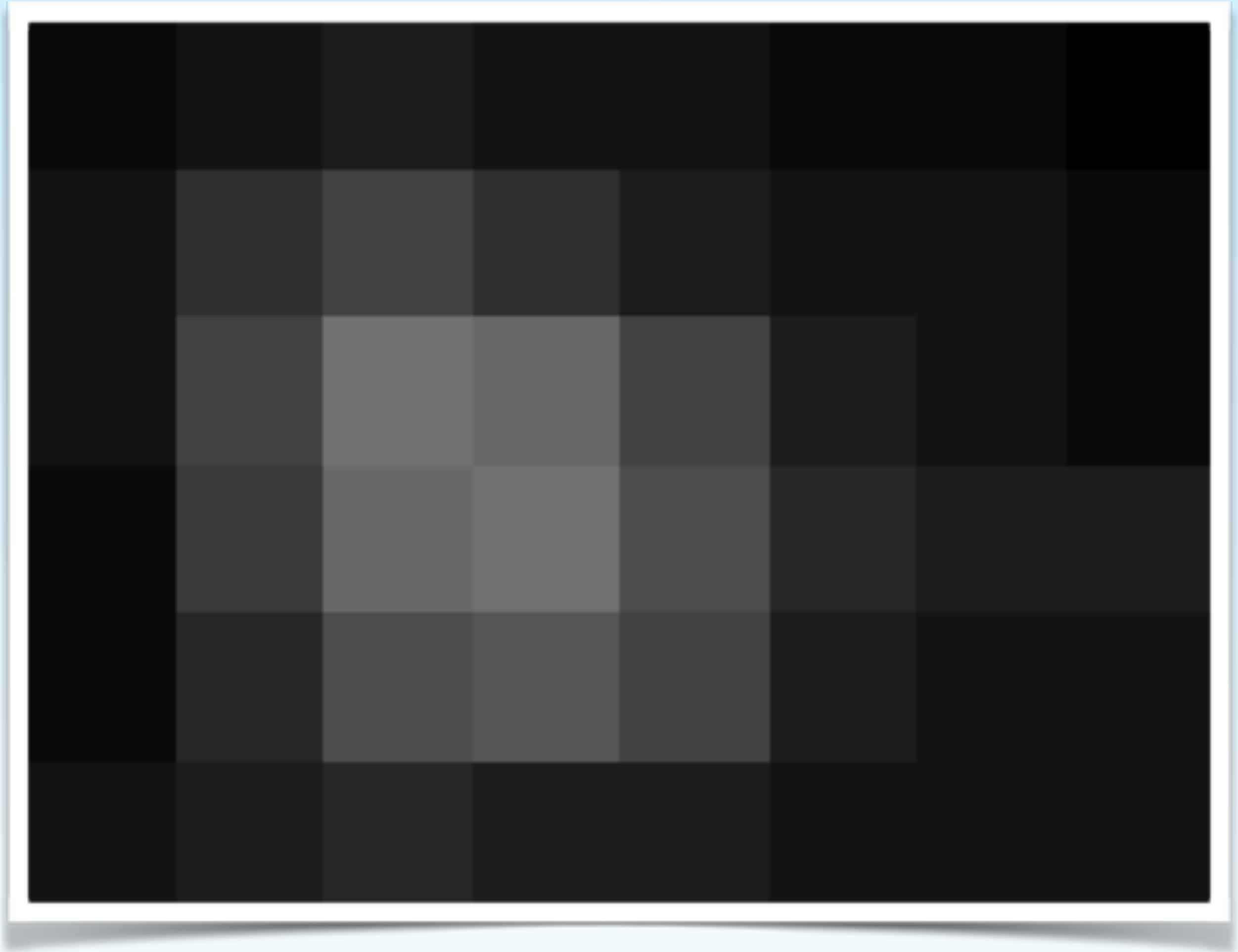




Result:

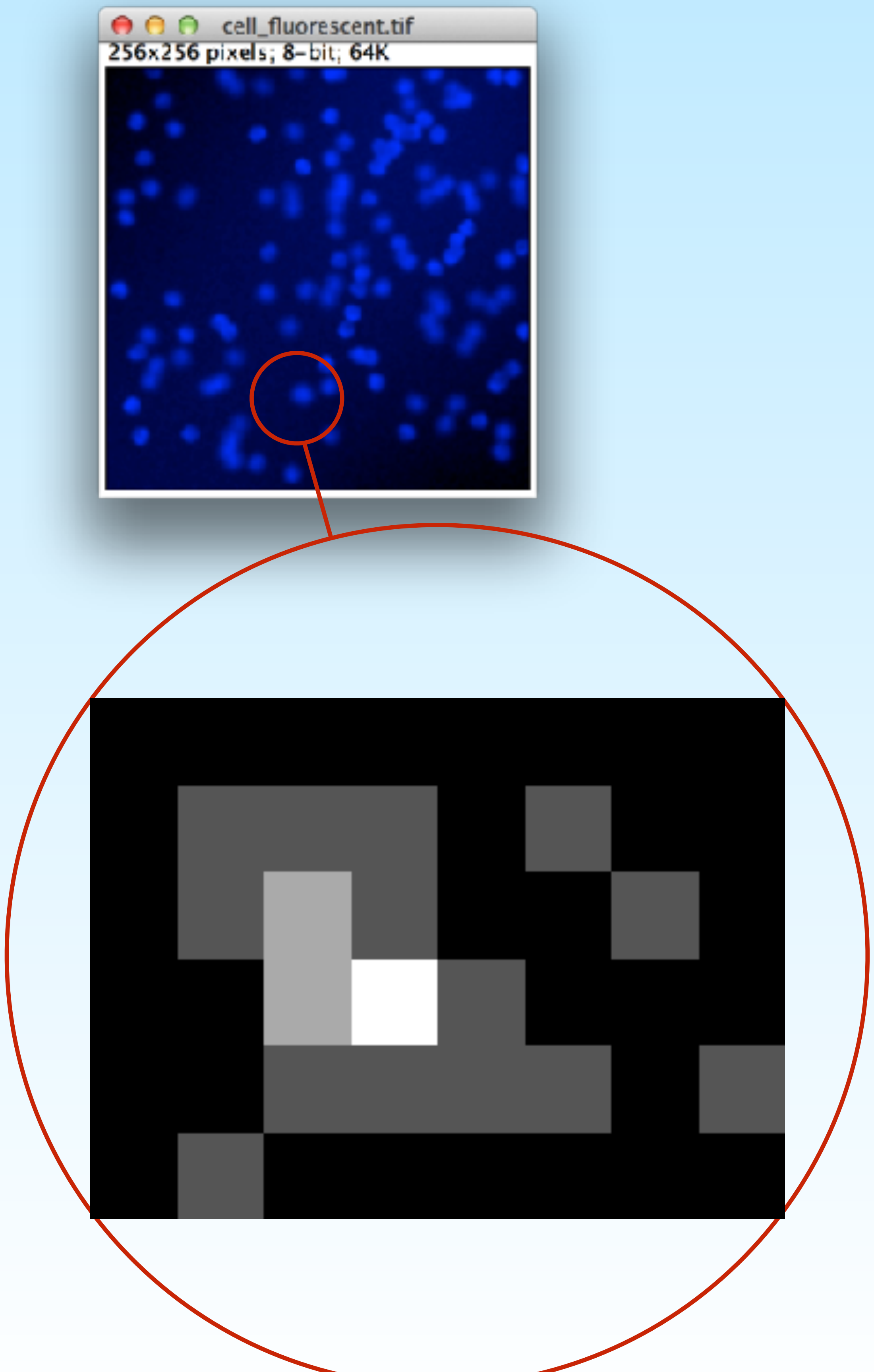


The output is a smoothed representation of the input.



Source:

Non-linear filter: Maximum filter

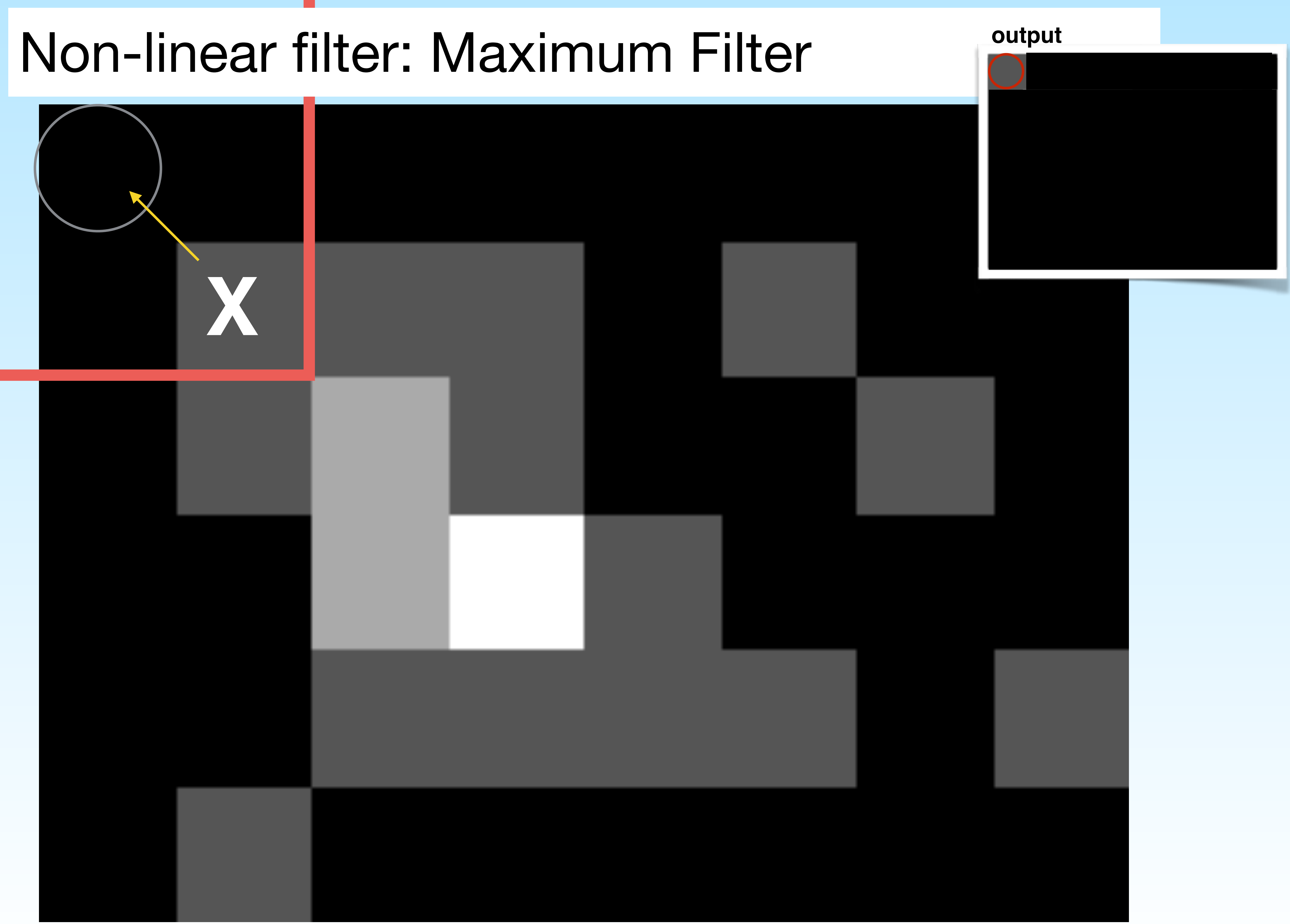


?

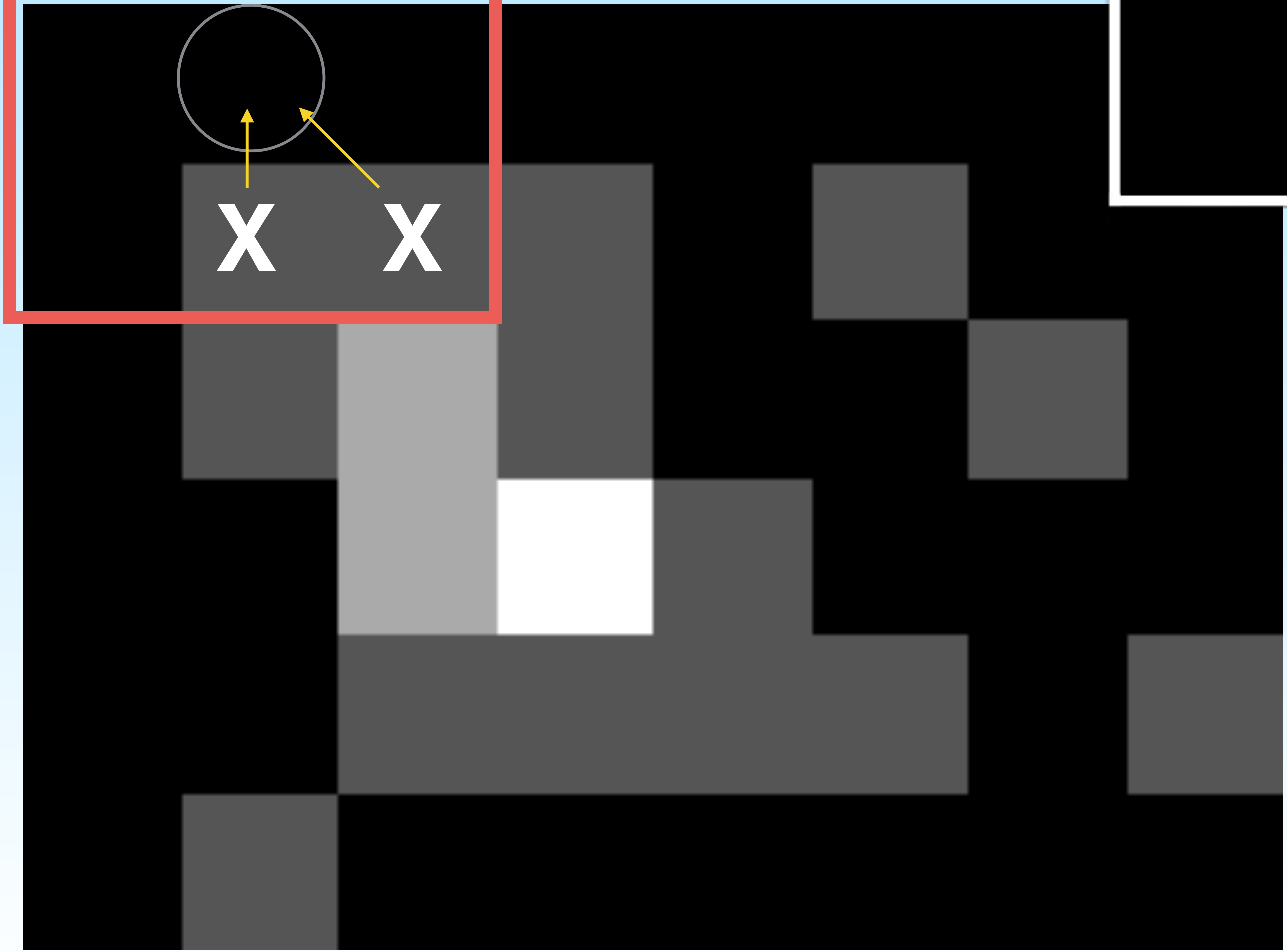
kernel can not be expressed as is non-linear

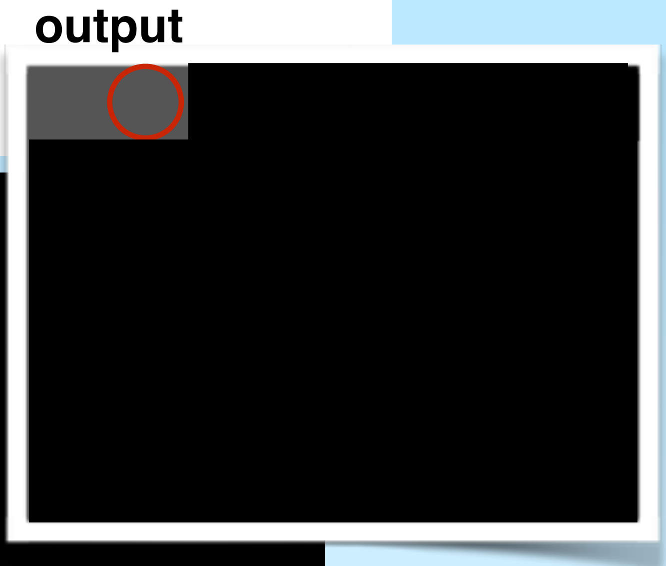
$$
[[1, 1, 1, 1, 1, 1, 1, 1],\\
[1, 2, 2, 2, 1, 2, 1, 1],\\
[1, 2, 3, 2, 1, 1, 2, 1],\\
[1, 1, 3, 4, 2, 1, 1, 1],\\
[1, 1, 2, 2, 2, 2, 1, 2],\\
[1, 2, 1, 1, 1, 1, 1, 1]]
$$

Non-linear filter: Maximum Filter

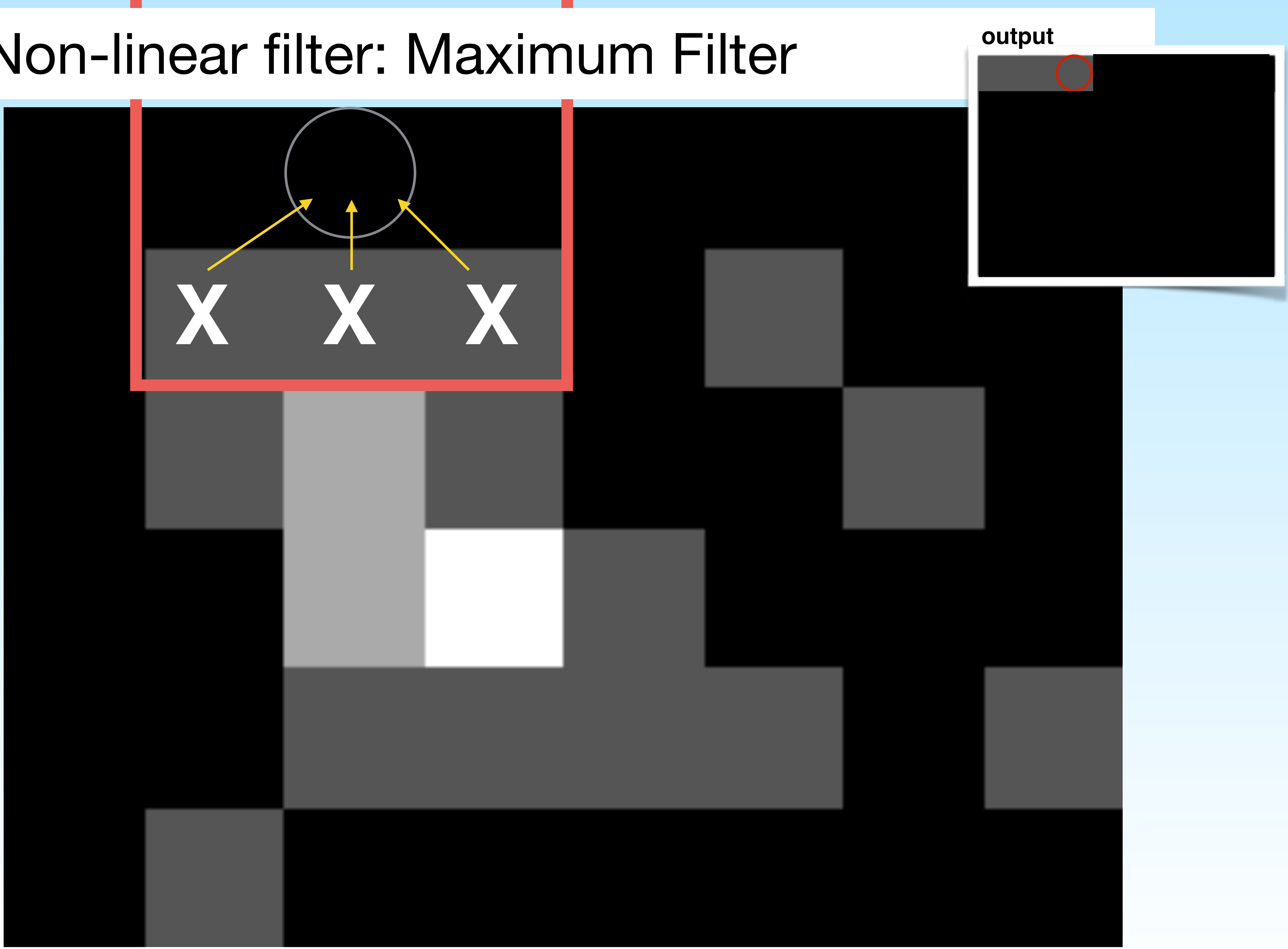


Non-linear filter: Maximum Filter

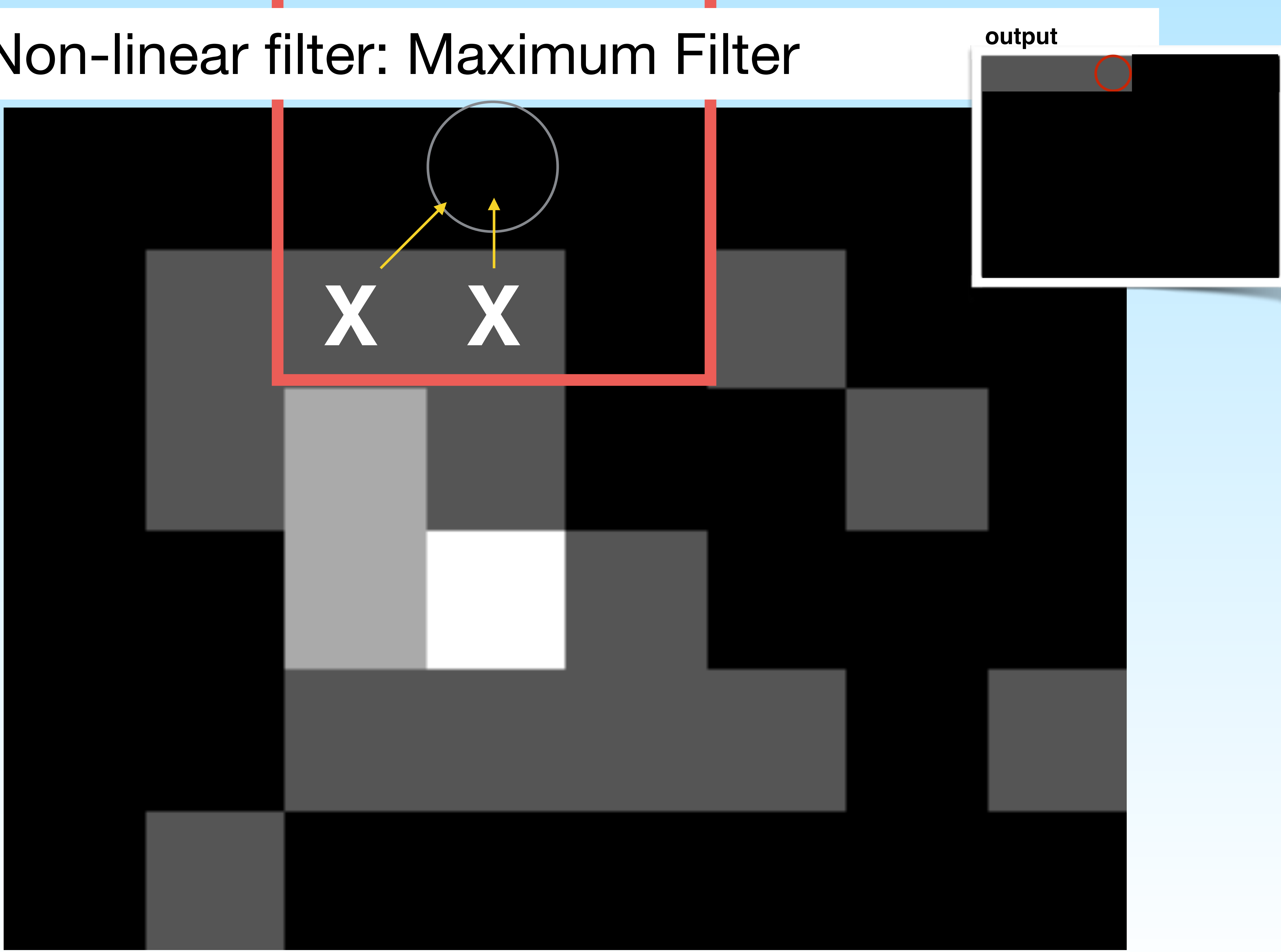




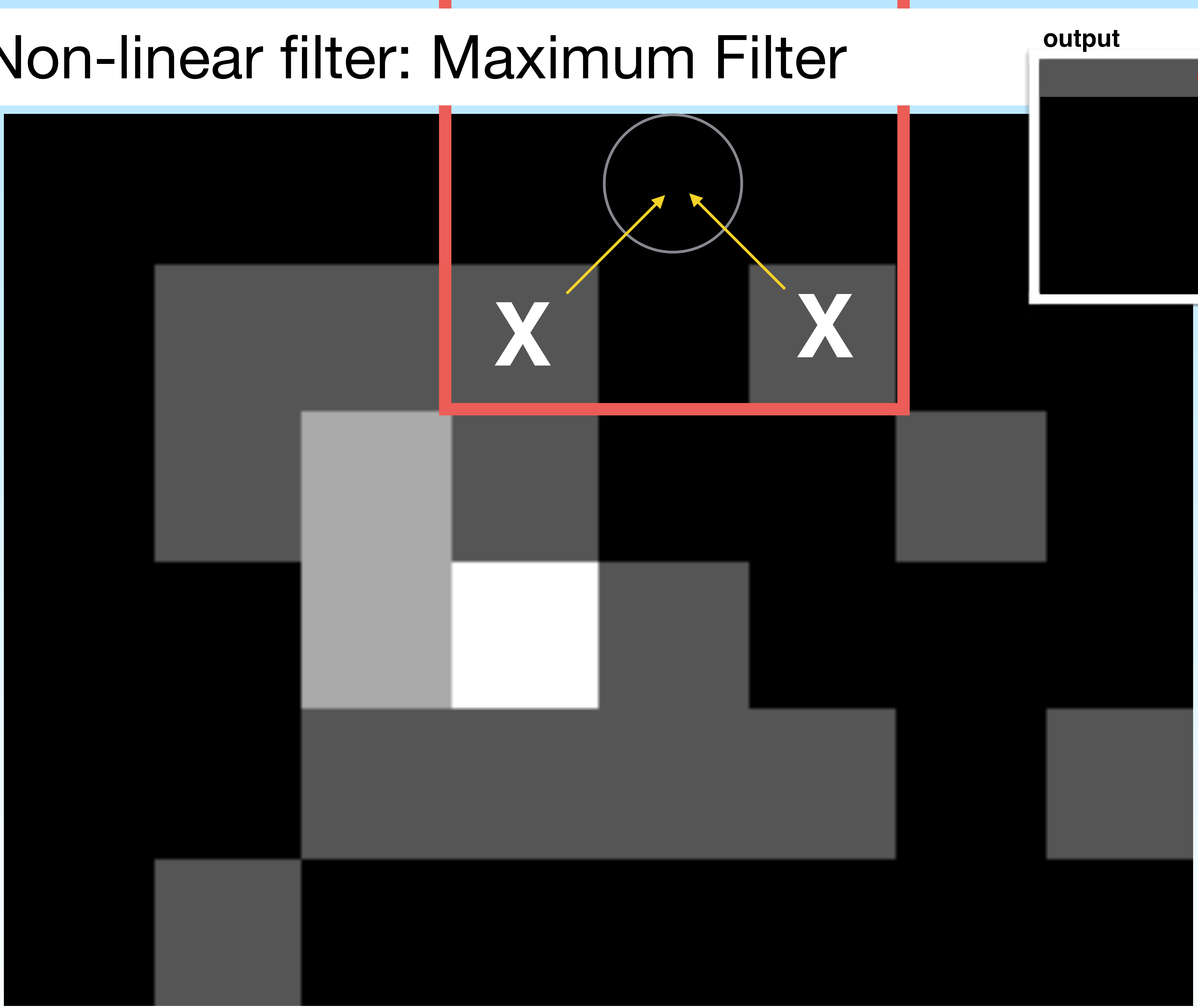
Non-linear filter: Maximum Filter

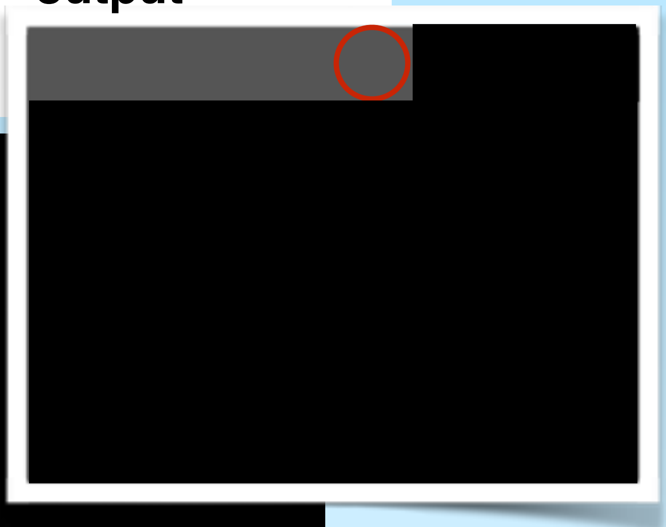


Non-linear filter: Maximum Filter

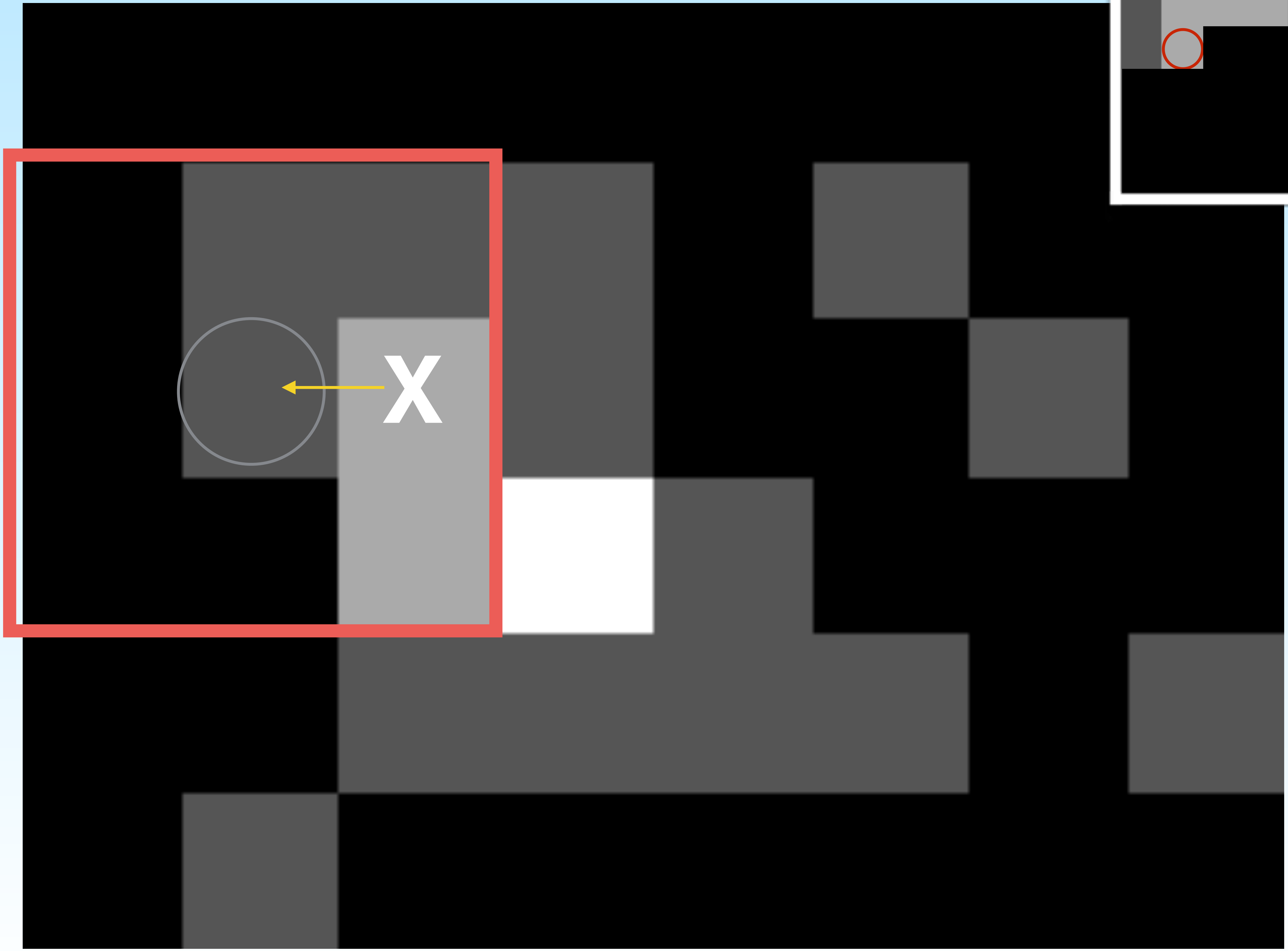


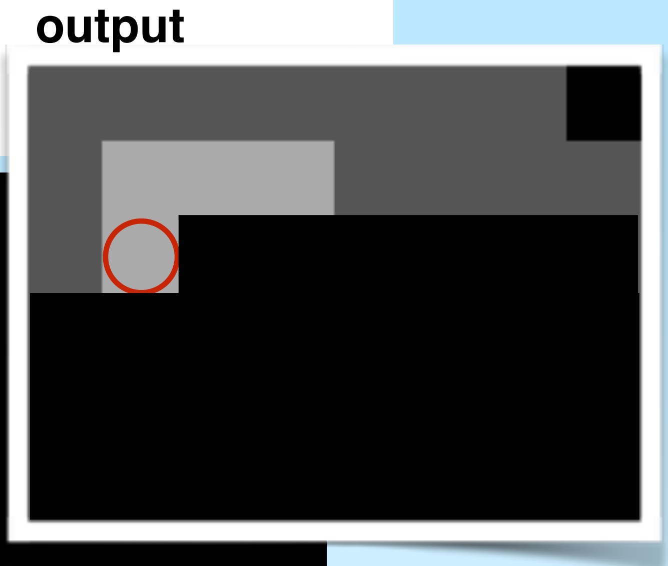
Non-linear filter: Maximum Filter



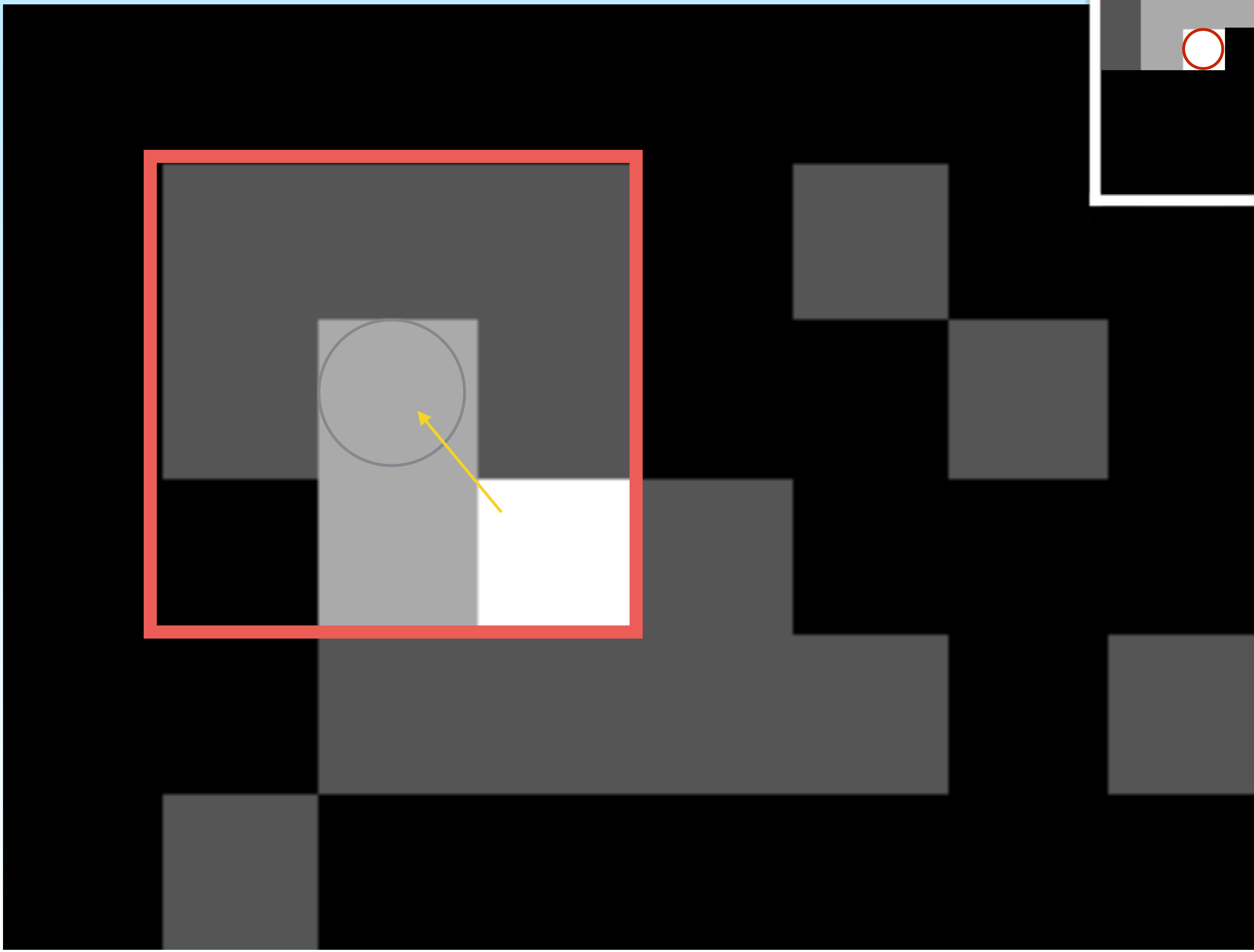


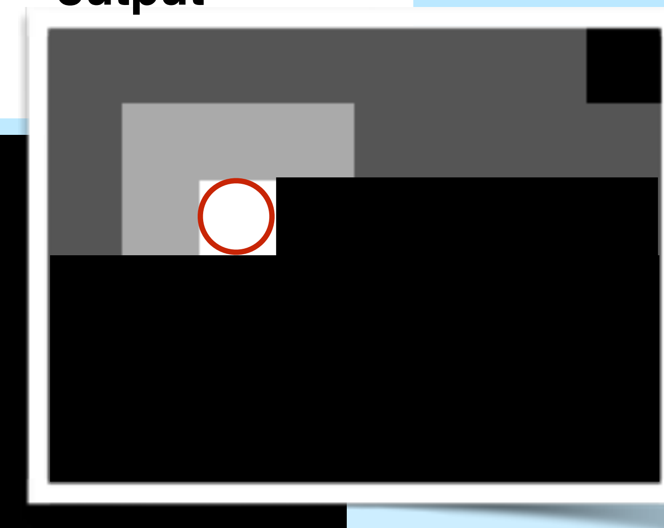
Non-linear filter: Maximum Filter



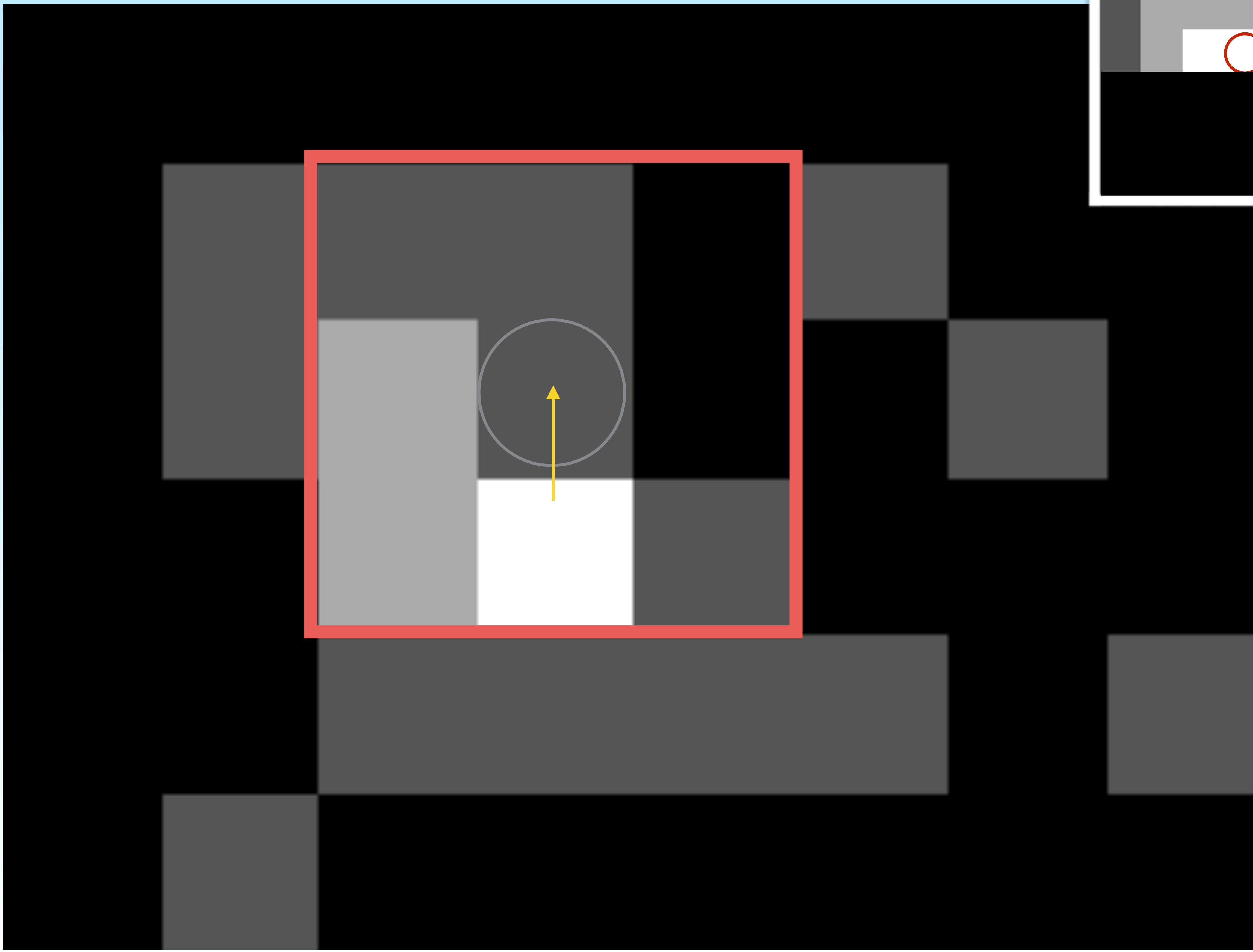


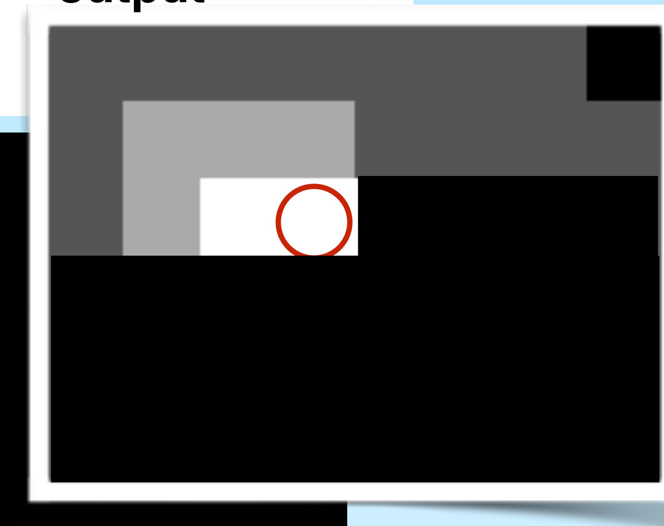
Non-linear filter: Maximum Filter



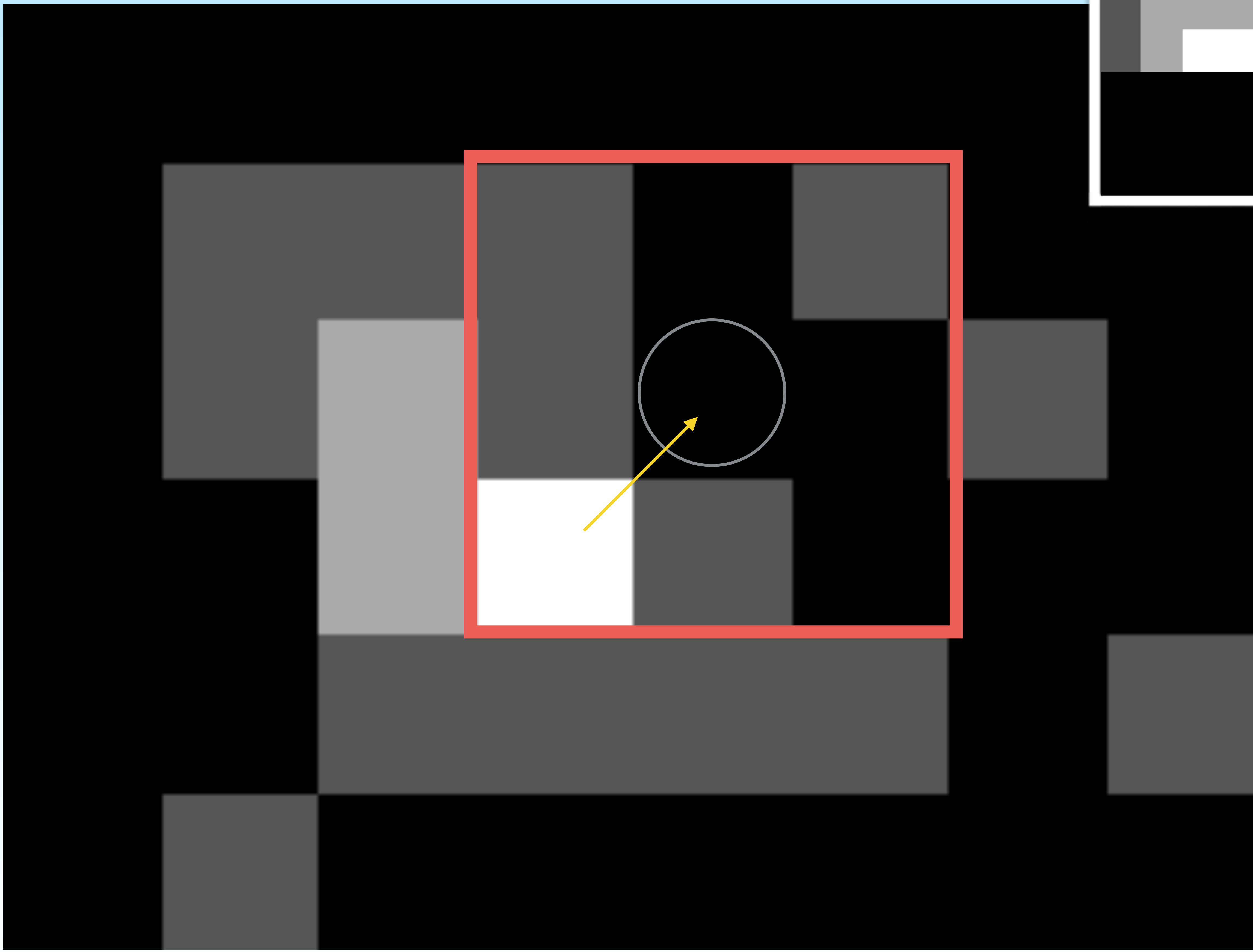


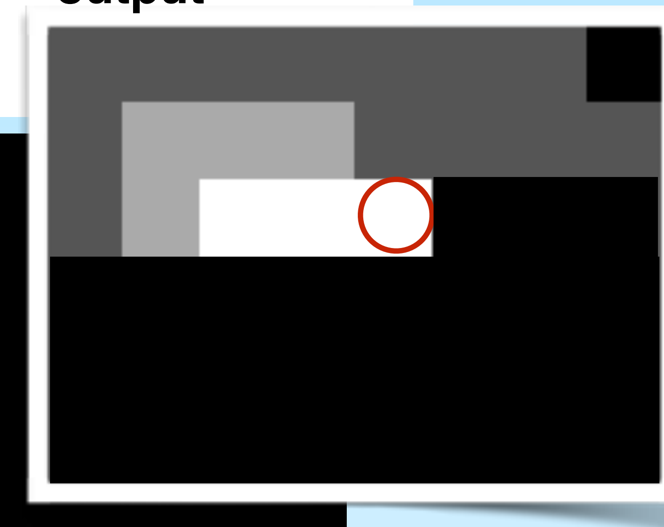
Non-linear filter: Maximum Filter



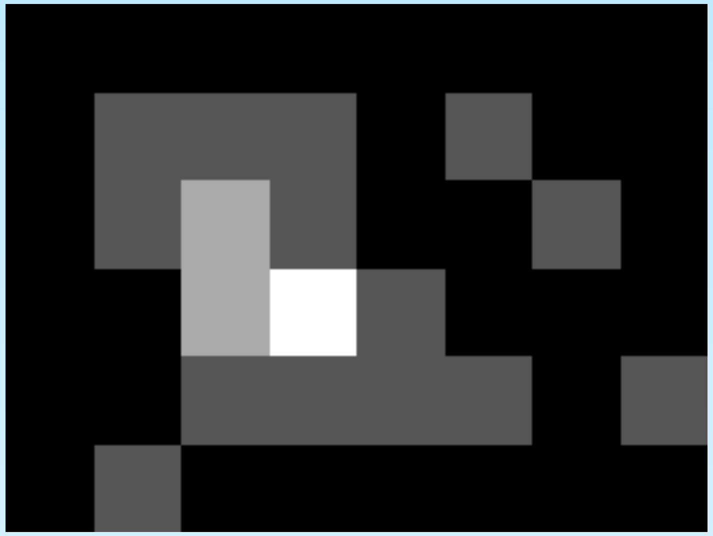


Non-linear filter: Maximum Filter

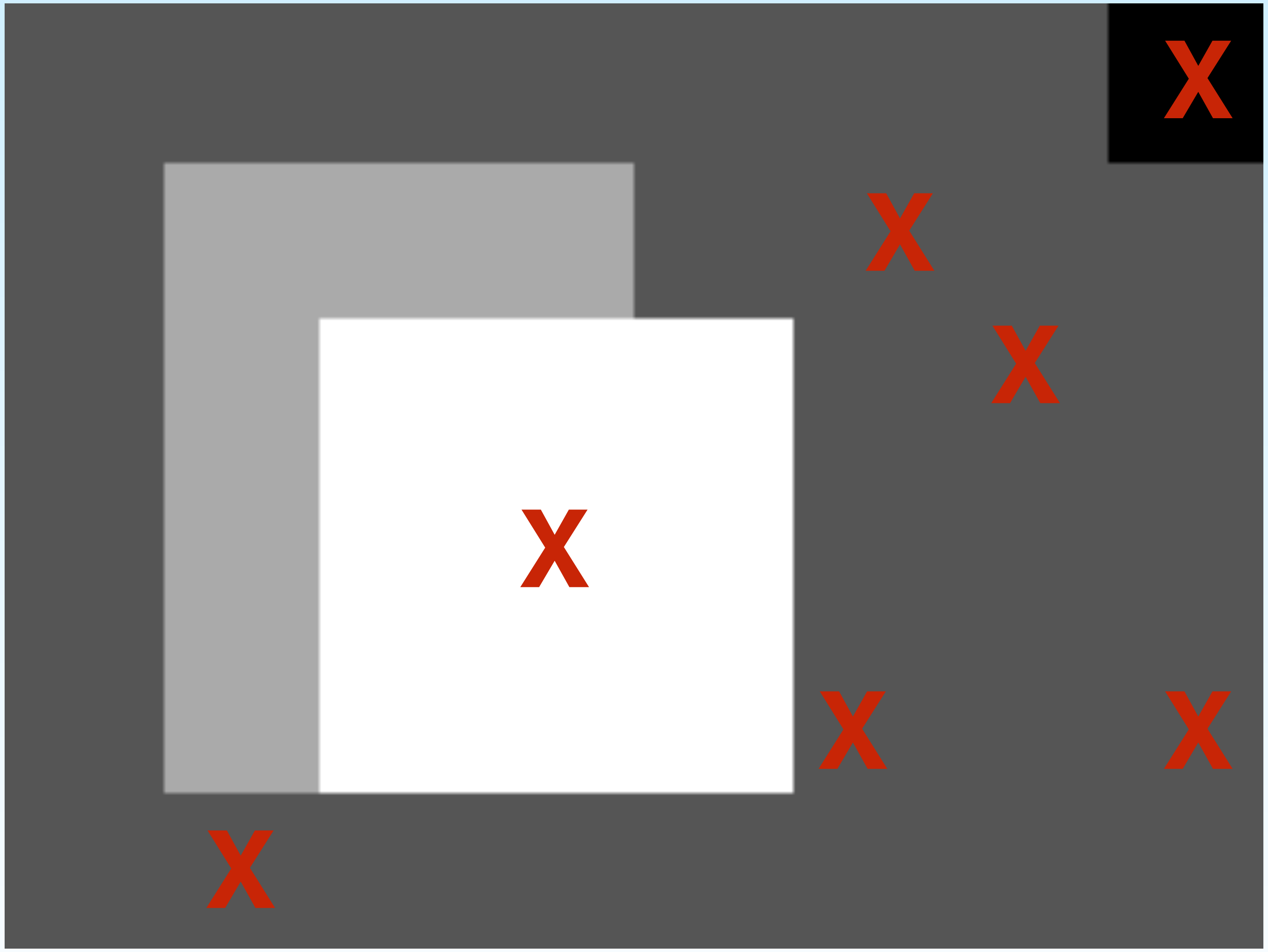




Result:



Where the pixel is the same in the input as the maximum filtered image you find your local maxima



Source:

Convolution - edge filters

1st derivative filter

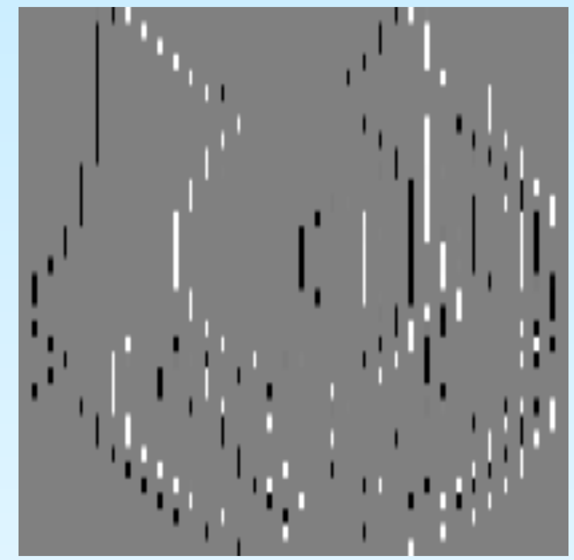
kernel

$$\begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

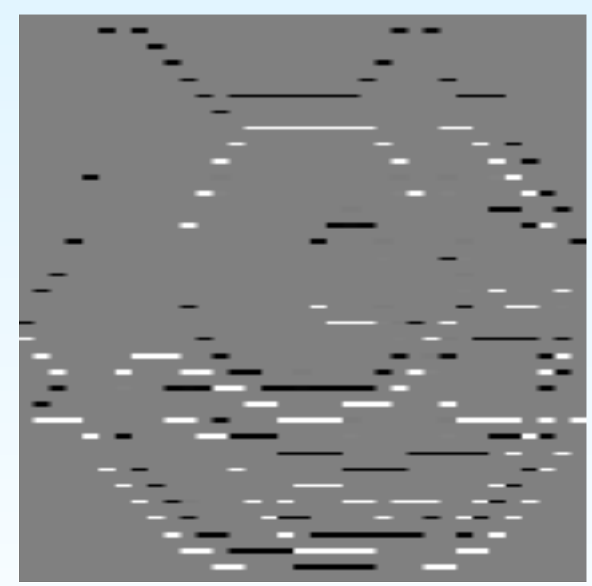
input

output (convolved)



other direction

$$\begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$$



Extracts edges in a particular direction

black is negative
white is positive

Source: http://www.uff.br/cdme/matrix/matrix-html/matrix_boolean/matrix_boolean_en.html

Convolution - edge filters

1st derivative filter 2d filter

kernel

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

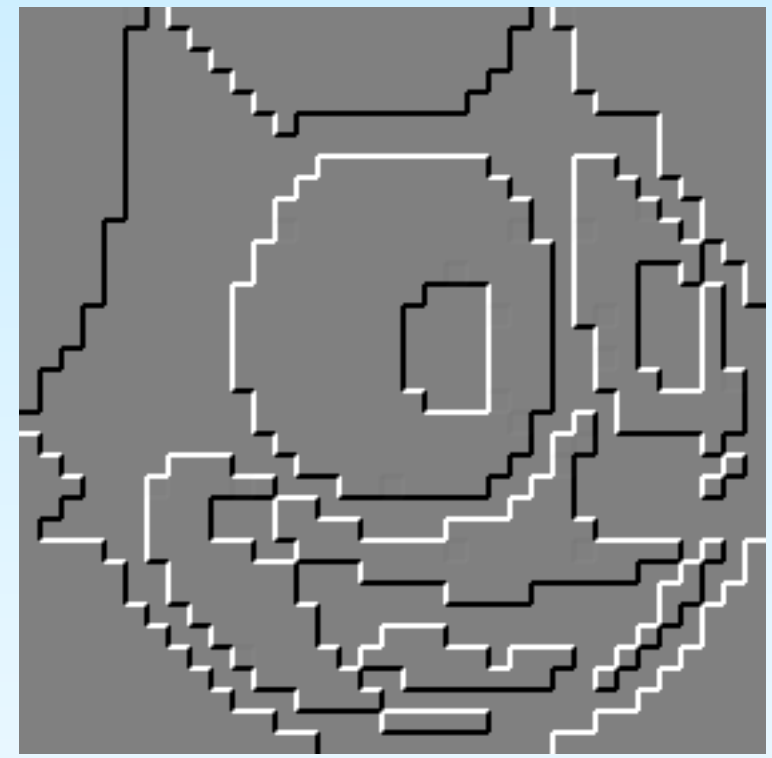
$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

2D filter

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

*

output (convolved)

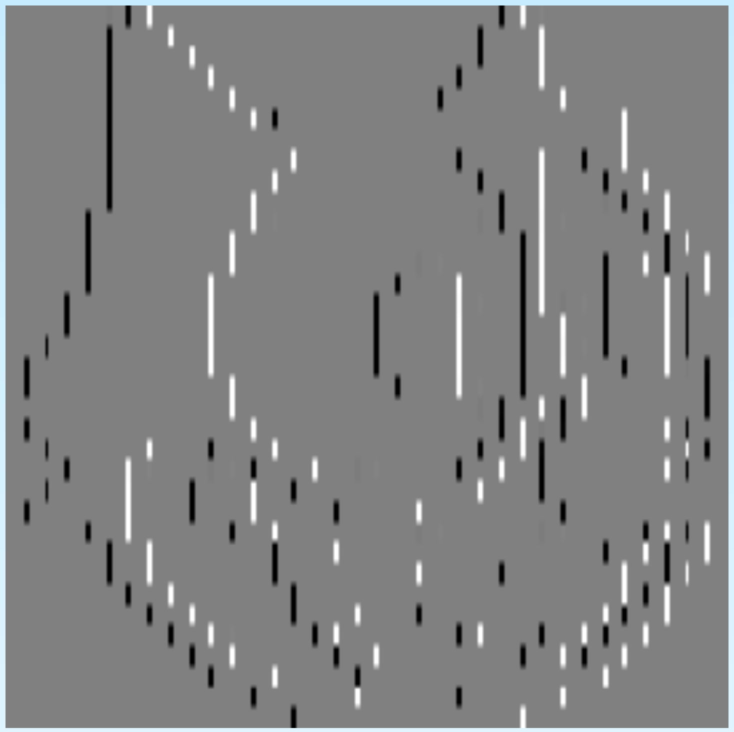


black is negative
white is positive

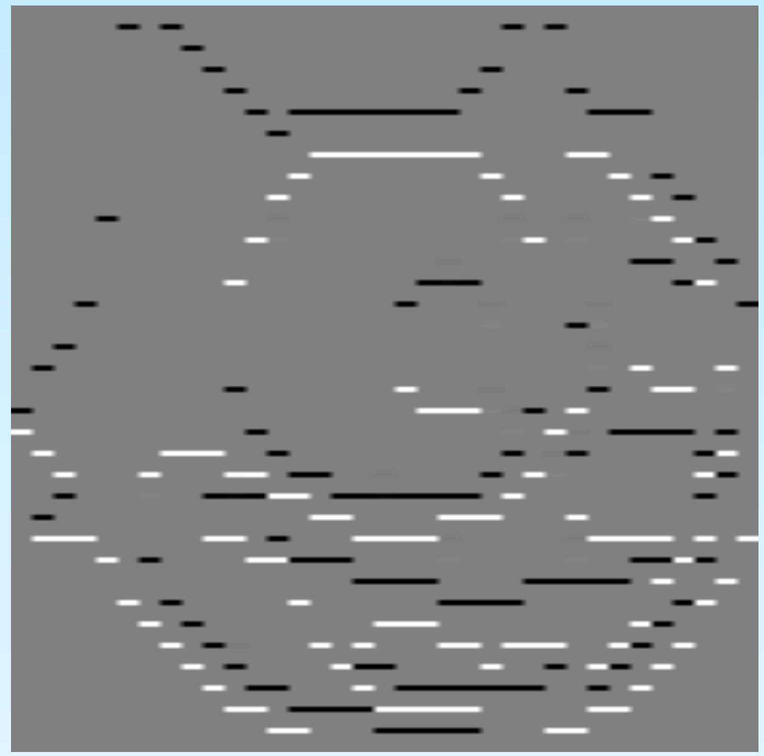
Source: http://www.uff.br/cdme/matrix/matrix-html/matrix_boolean/matrix_boolean_en.html

Convolution - Gradient magnitude gives edges

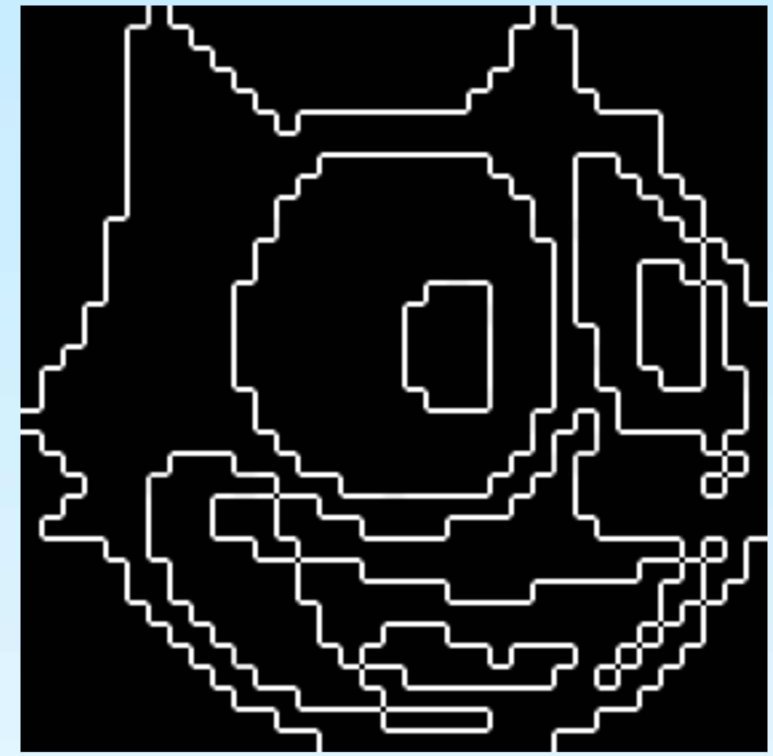
image**2

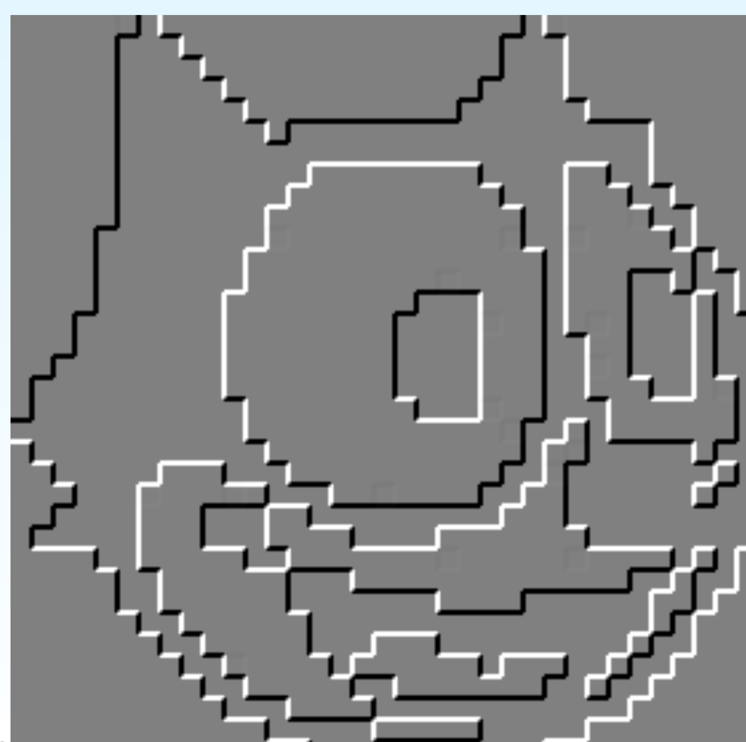


+



=

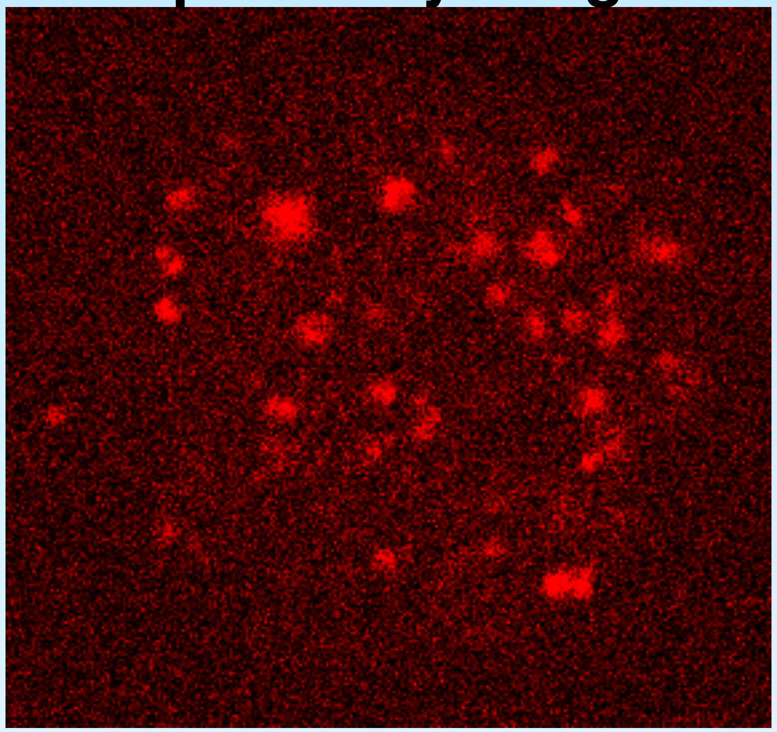




Source:

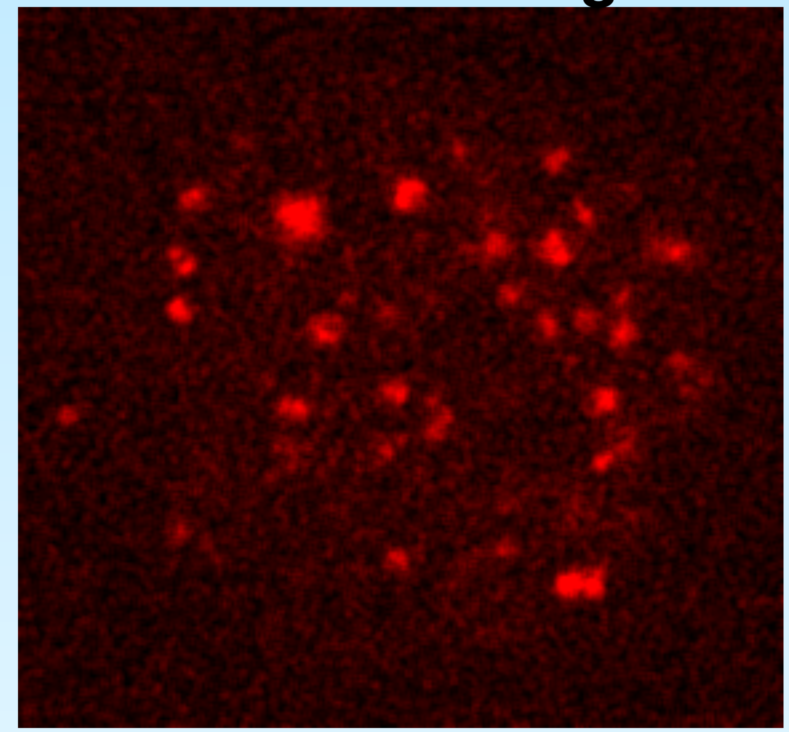
Mean Filter

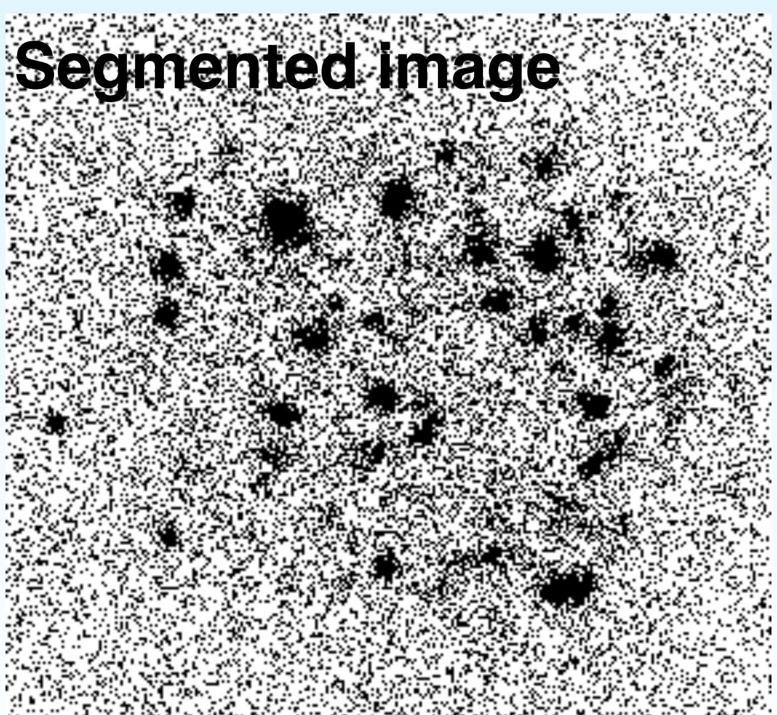
input noisy image

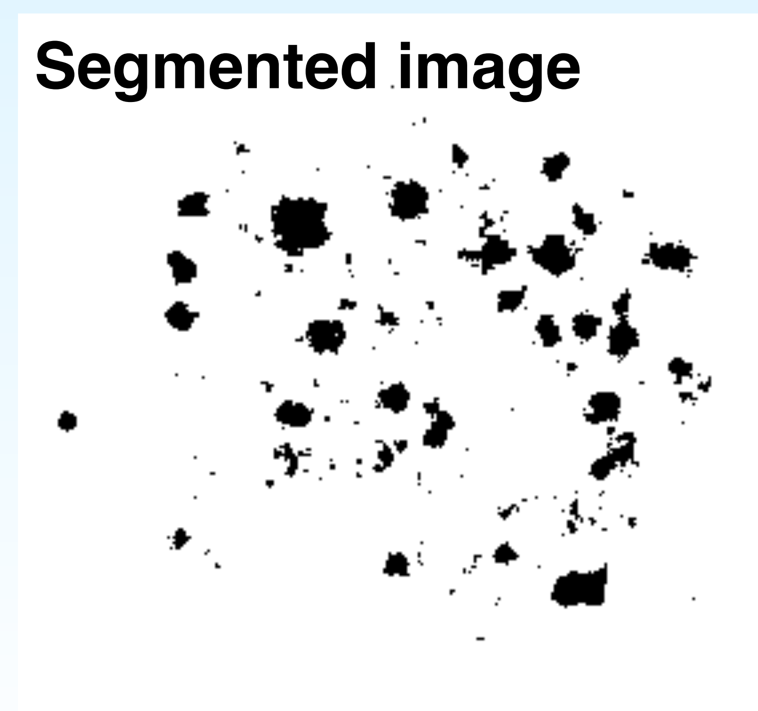


$$* \quad \begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix} \quad =$$

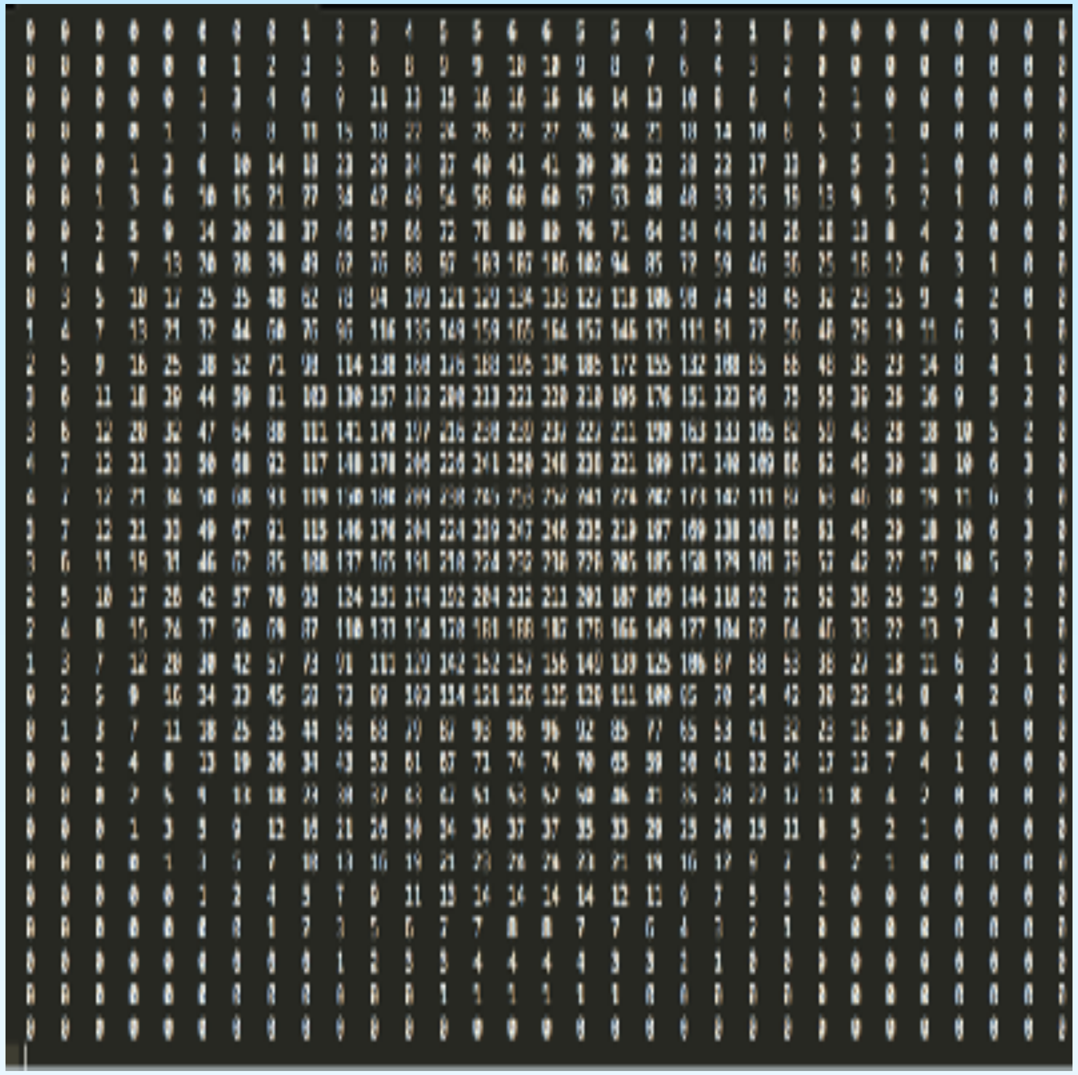
smoothed image

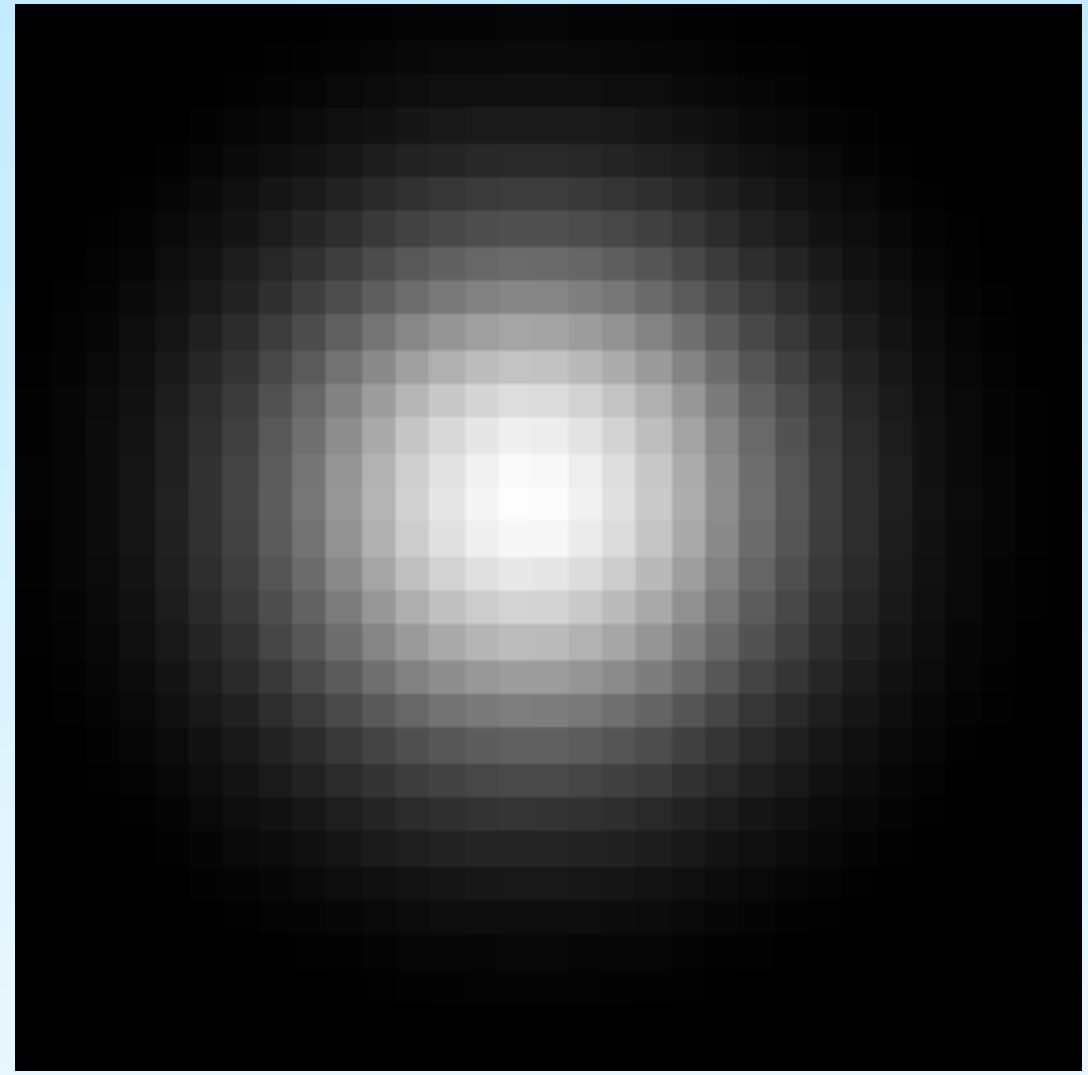






Gaussian filter.



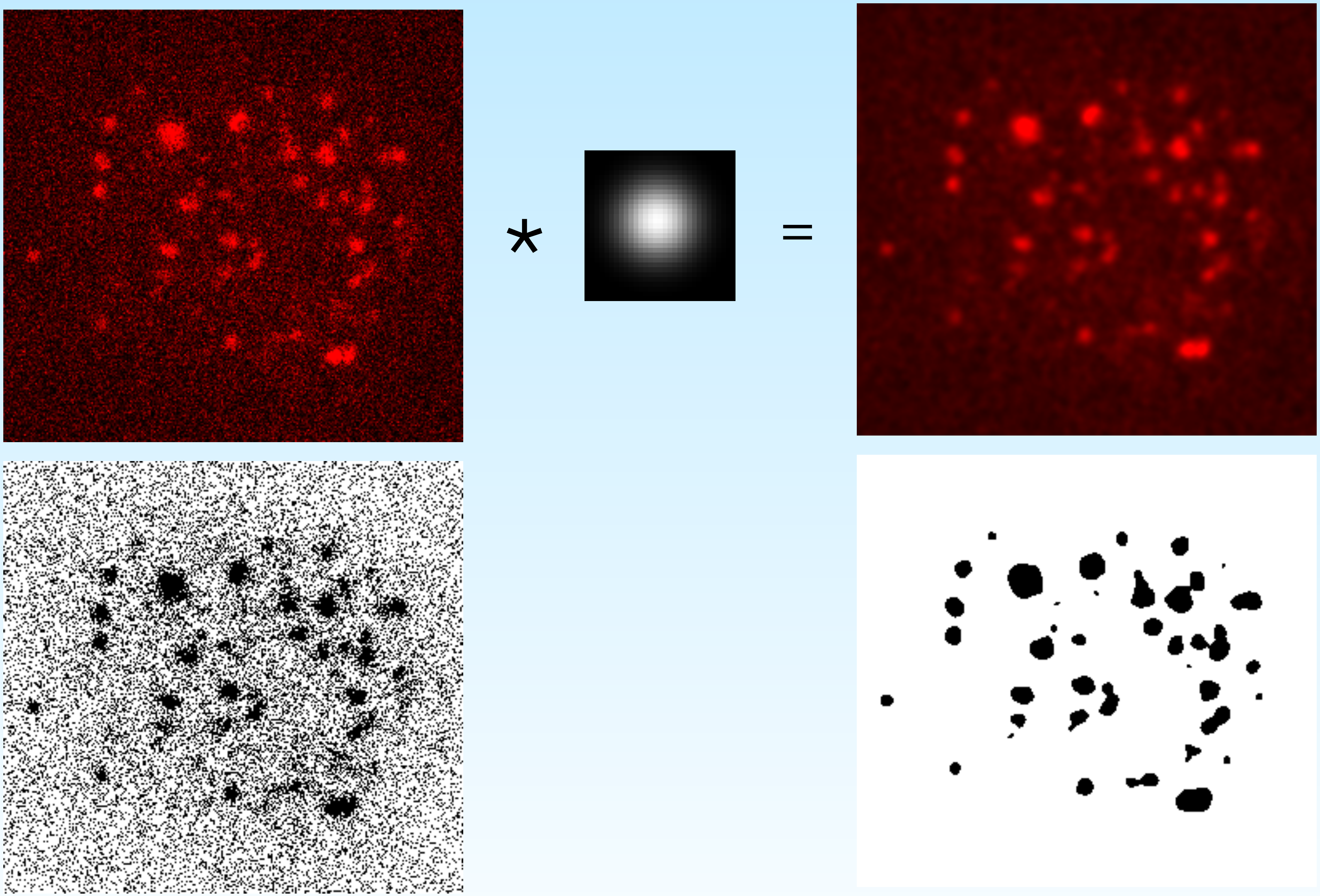


$$f(x,y) = A\exp\left(-\left(\frac{(x-x_o)^2}{2\sigma_x^2} + \frac{(y-y_o)^2}{2\sigma_y^2}\right)\right)$$

x, y (pixel coordinates)
sigma of kernel
x0,y0 center of Gaussian

We can use this to remove the image background.

Denoising through Gaussian deconvolution



Weights pixels nearer centre more

Convolution - Laplacian filter

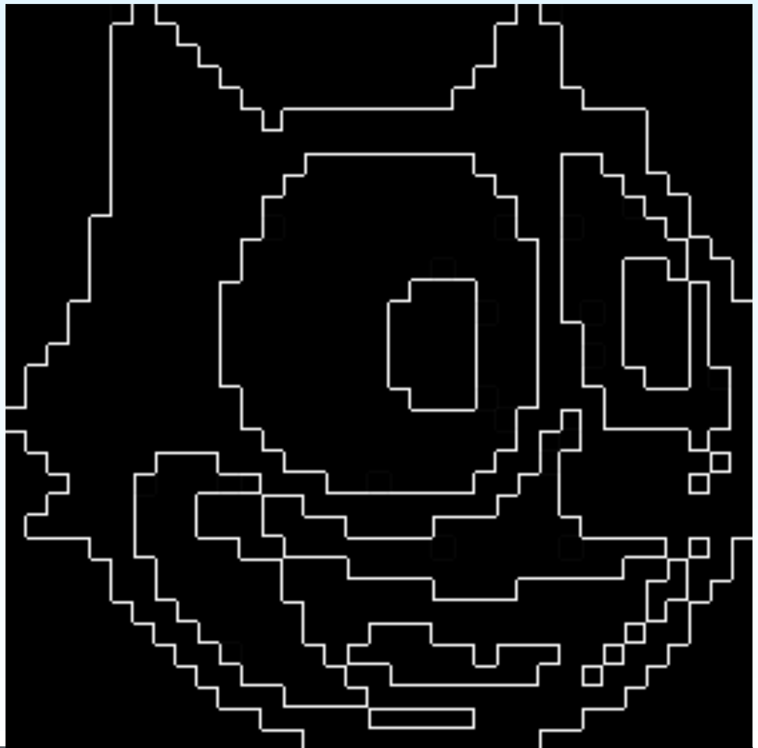
2nd derivate filter

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

1D filter: $\vec{D}_x^2 = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix},$

2D filter: $\mathbf{D}_{xy}^2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$

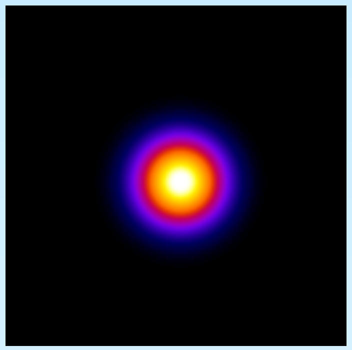
2D filter: $\mathbf{D}_{xy}^2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$



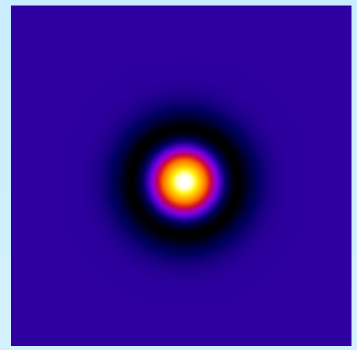
2-D laplacian filter detects change no matter in which direction. Wherever the rate of change is large.

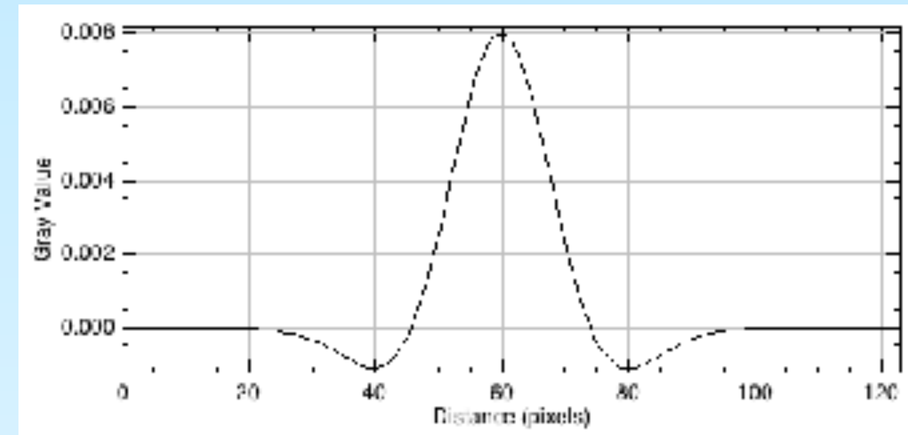
Gaussian of laplacian filters

In practice discrete filters are noisy.

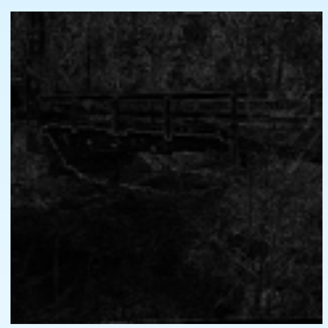


$$* \begin{matrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{matrix} =$$

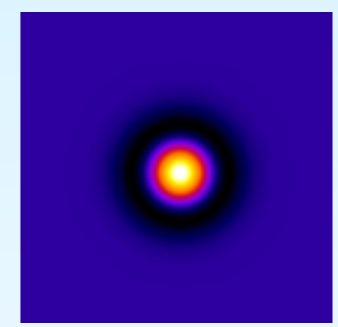




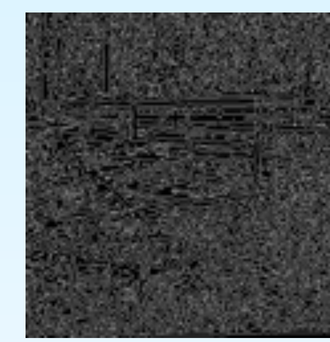
$$* \begin{matrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{matrix} =$$



$*$



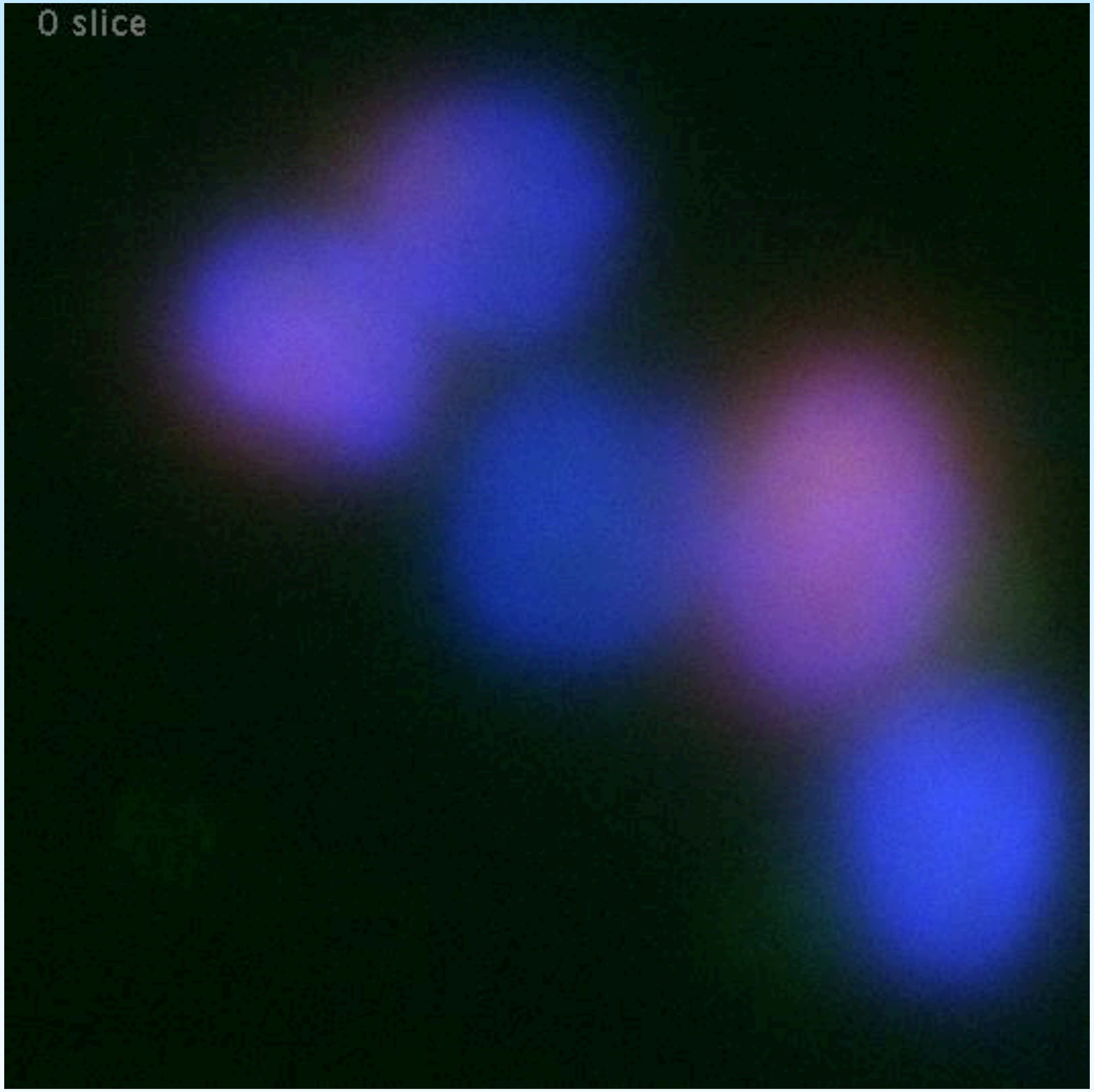
$=$



zero crossing

We can prevent the effects of this by convolving our filter (or our image first with a Gaussian filter).

Motivation for Analysis



Acquired on a Deltavision wide-field microscope.

Multiple z-slices.

The data is very rich, 3-D.

We want to find the colocalisation of the ATRX protein (Green) and the YH2Ax protein (Red)

Source: David Clynes

Motivation for Analysis

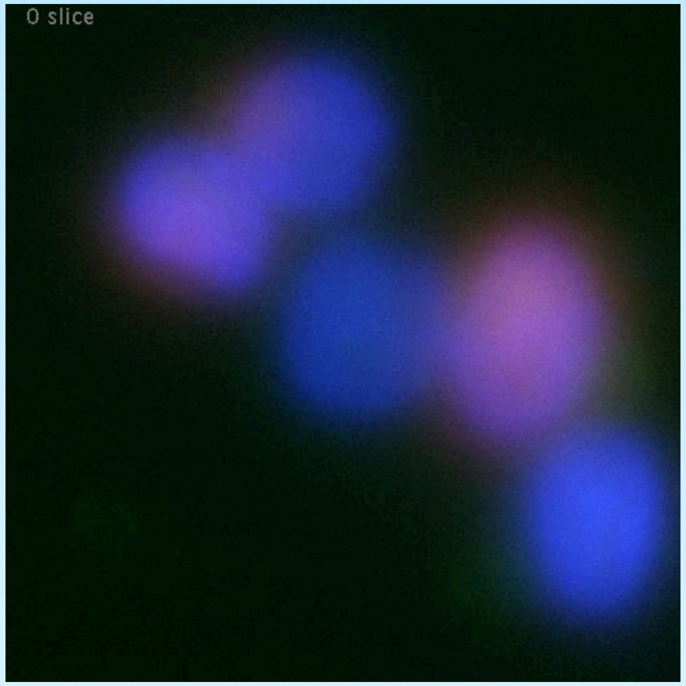
Early in your experimental pipeline you need to decide what the role of analysis is in your work.

- Is your research focus about developing novel analysis to solve a problem.
- Is your research about using analysis to investigate a biological question.

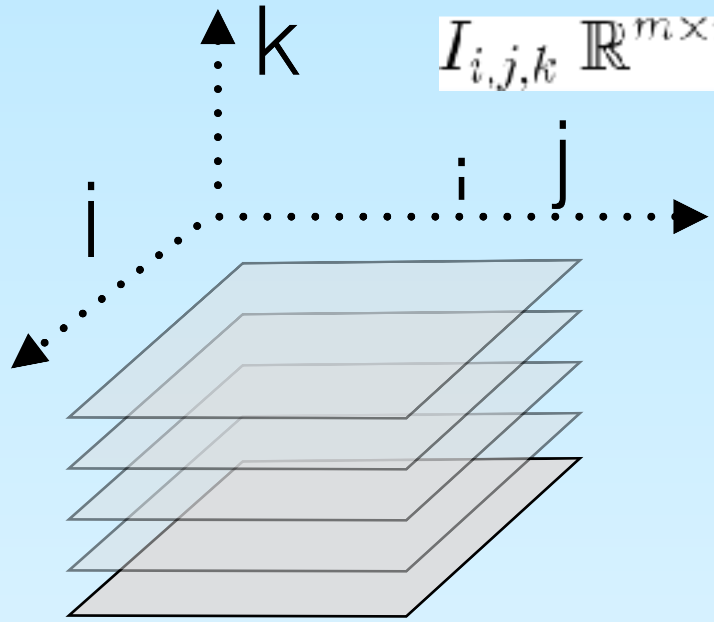
In this case we are just interested in creating a read-out for the biological experiment which is efficient and robust.

Source:

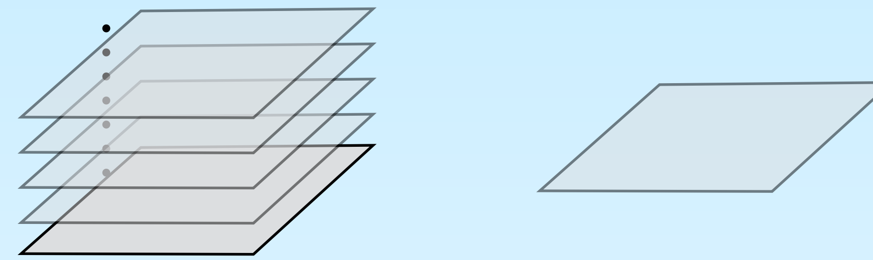
Typical imaging experiment

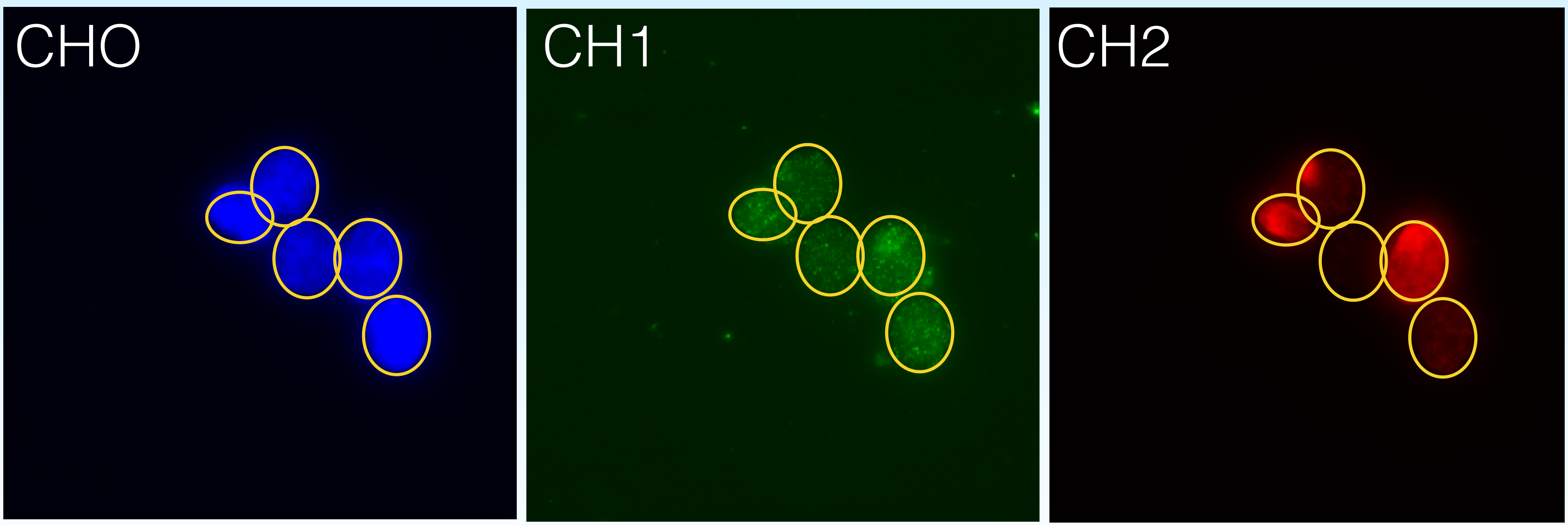


$I_{i,j,k} \; \mathbb{R}^{m \times n \times p}$



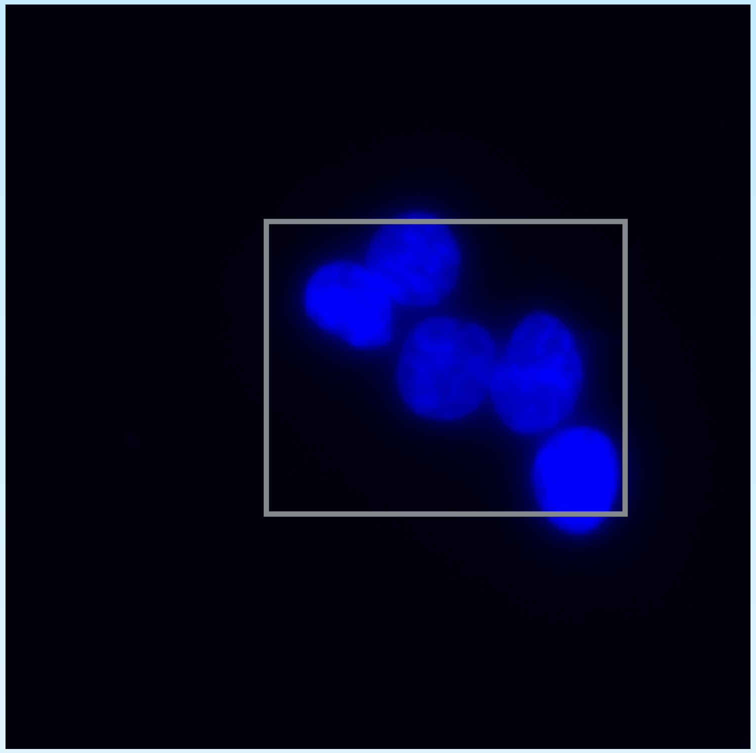
$$I_{i,j} = \max_{k=1}^{P}(I_{i,j,k})$$

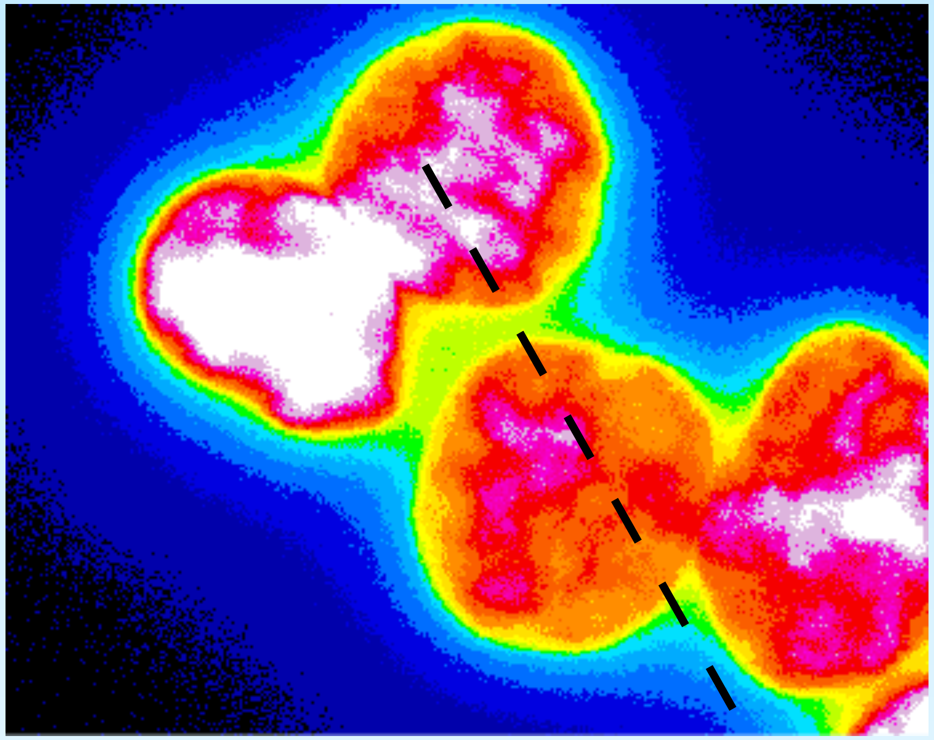


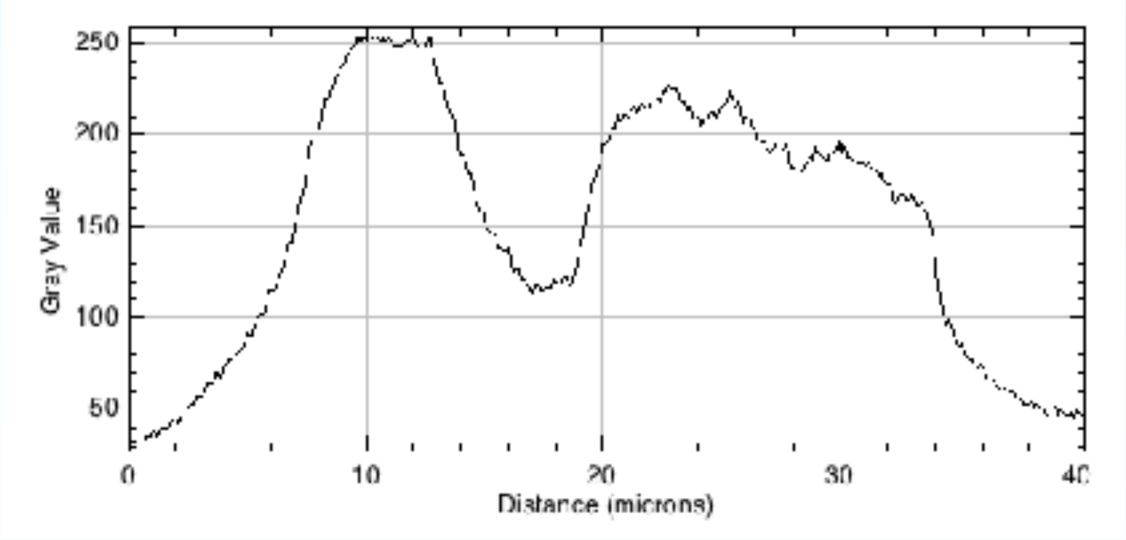


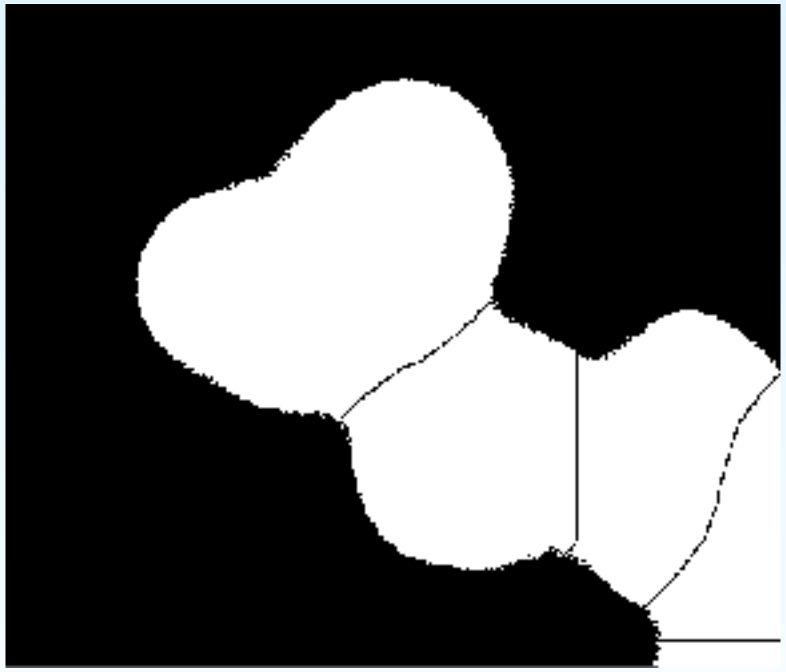
We want to isolate image regions into individual cells and then make measurements.

High background signal



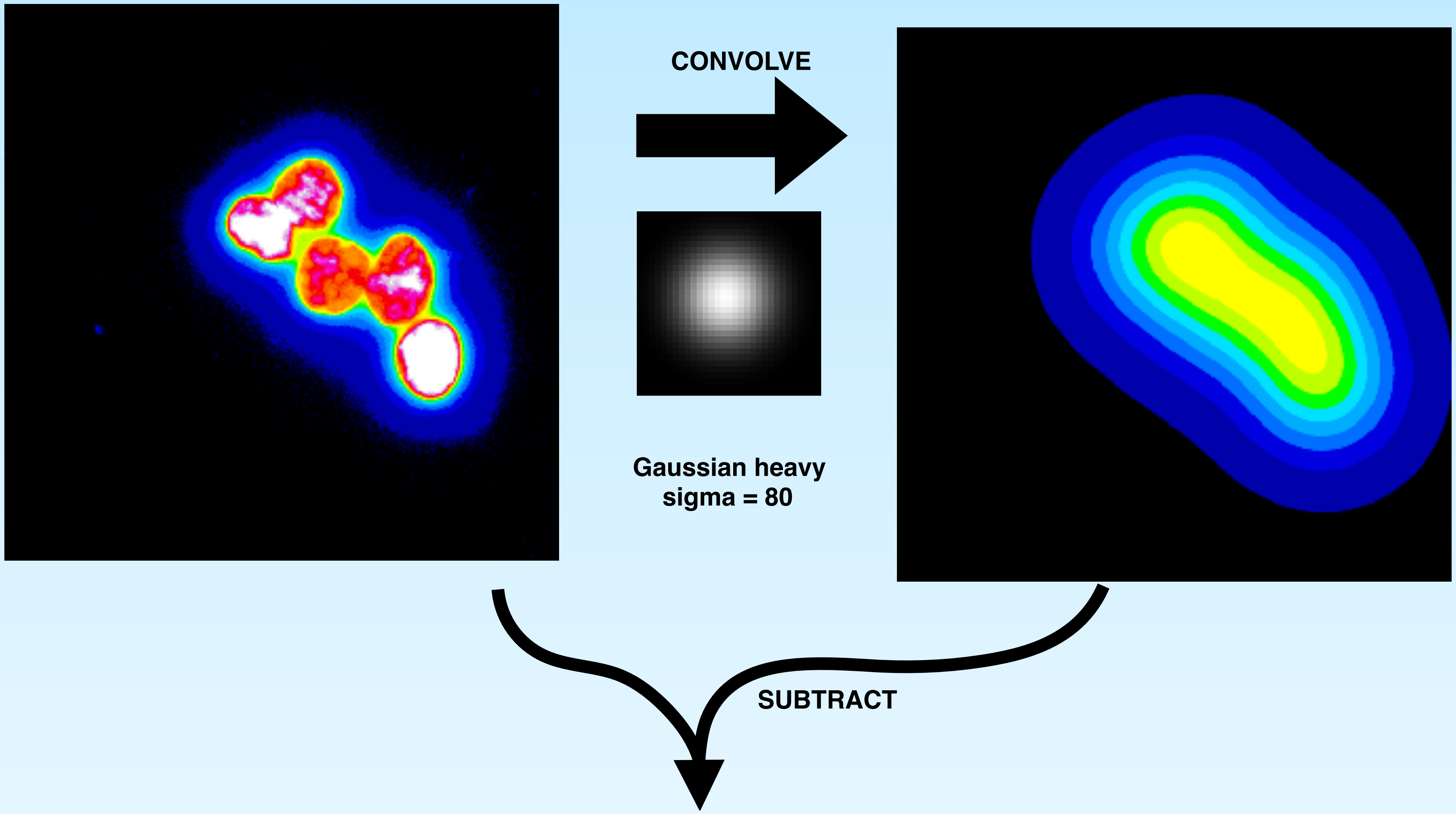






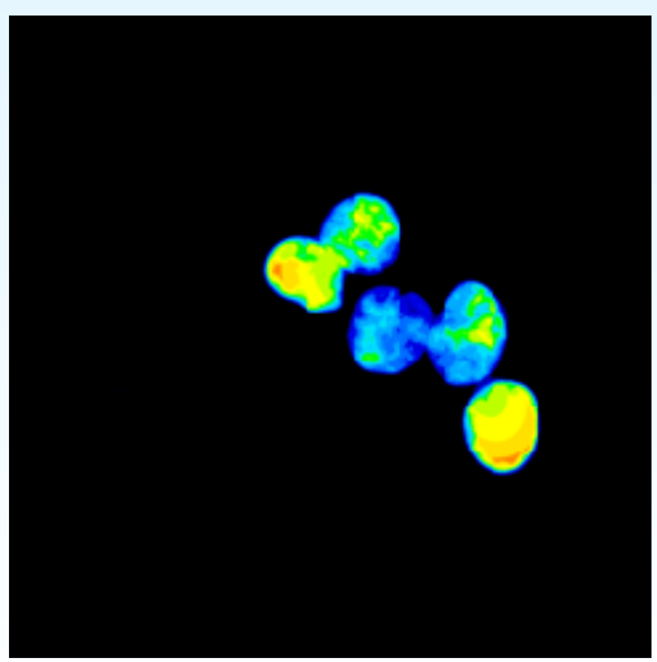
Cells will be hard to separate. We need to remove this smooth background signal. This background signal is low-frequency image signal.

Background removal, using filtering



Low frequencies of image.

equivalent to a low pass filter



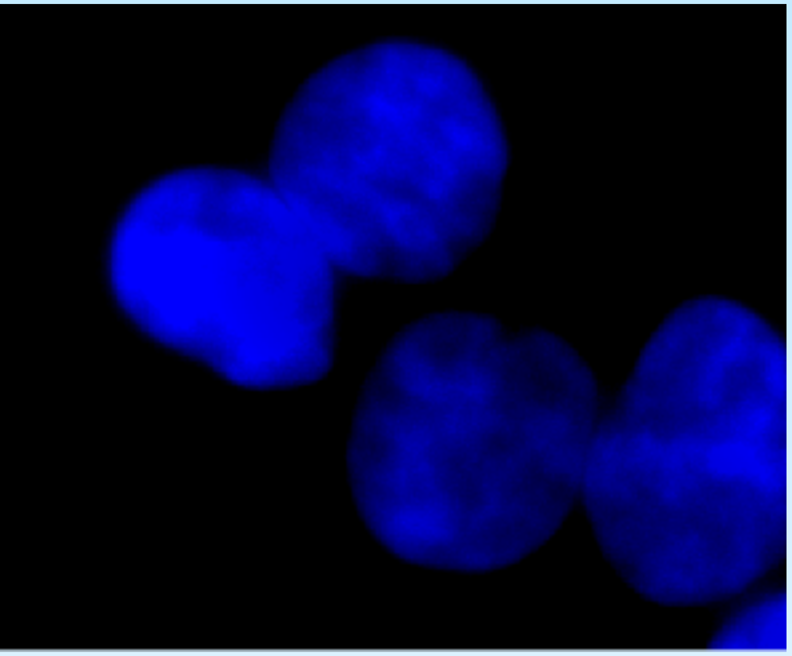
This is the resulting image.

The low frequencies have been removed leaving all the other frequencies.

Equivalent to a high pass filter.

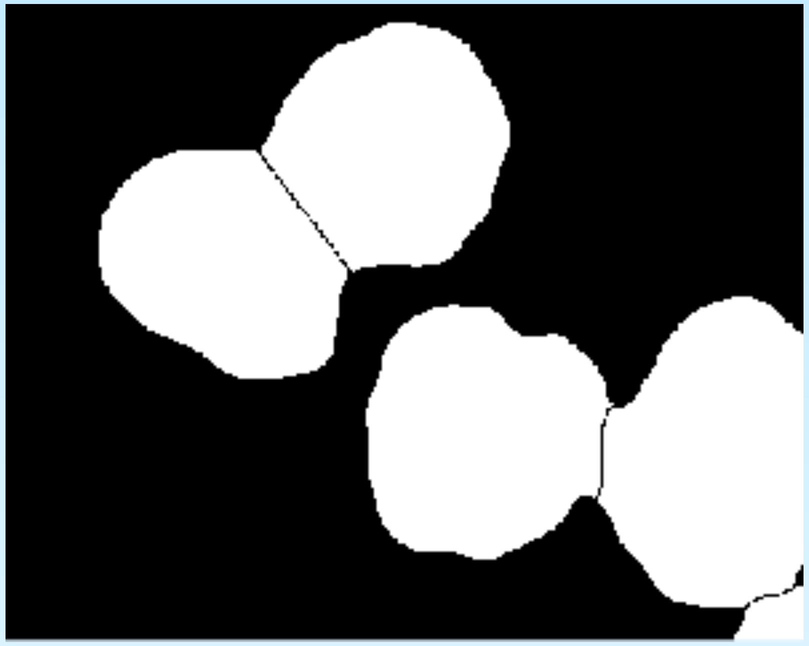
Source:

Segmentation much improved.



Filtered image

Segmented image

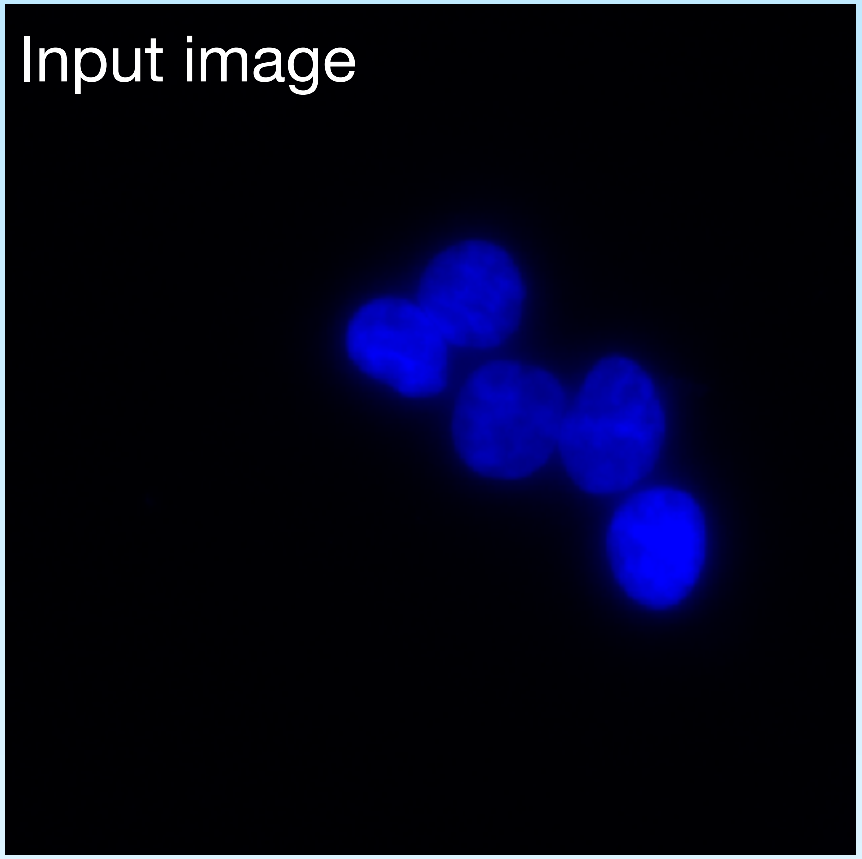


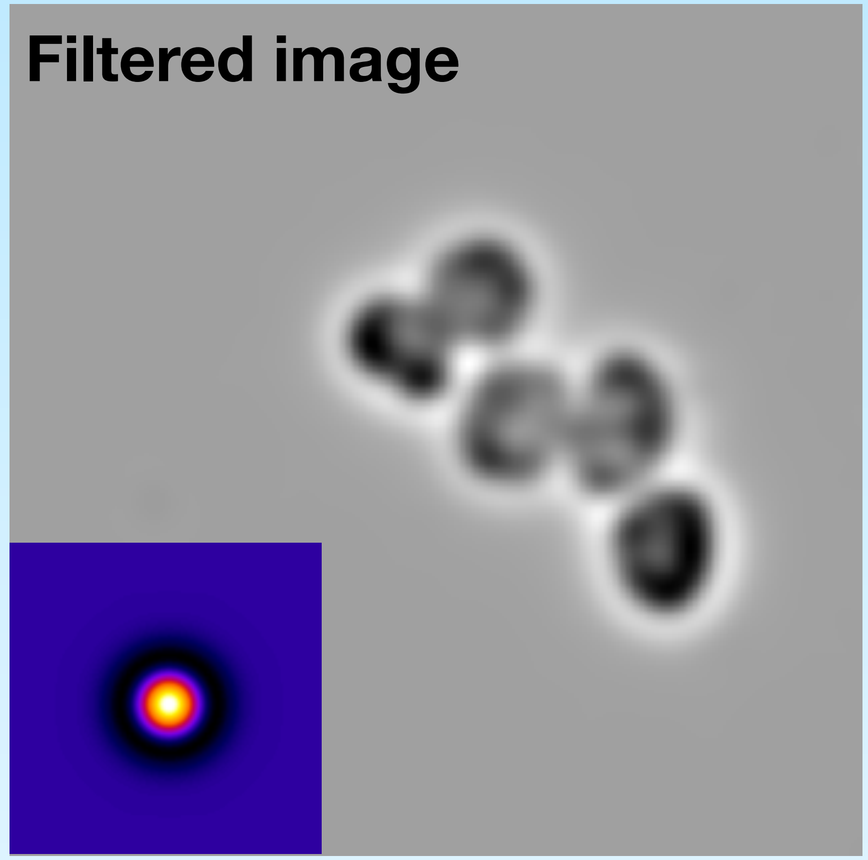
Watershedded image

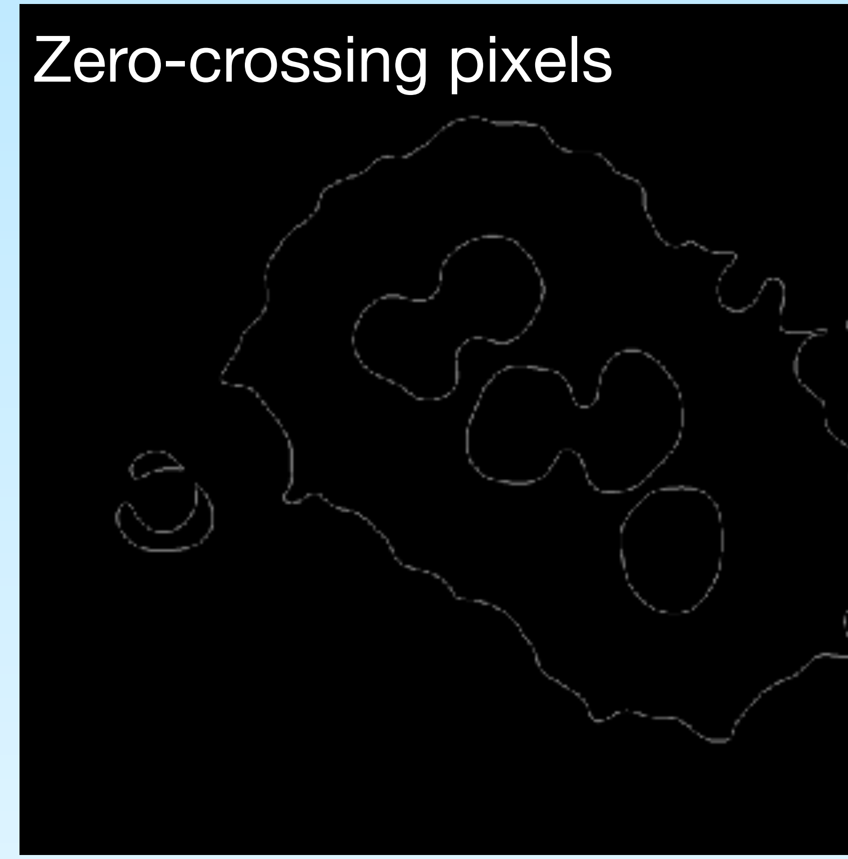
Segmented cells

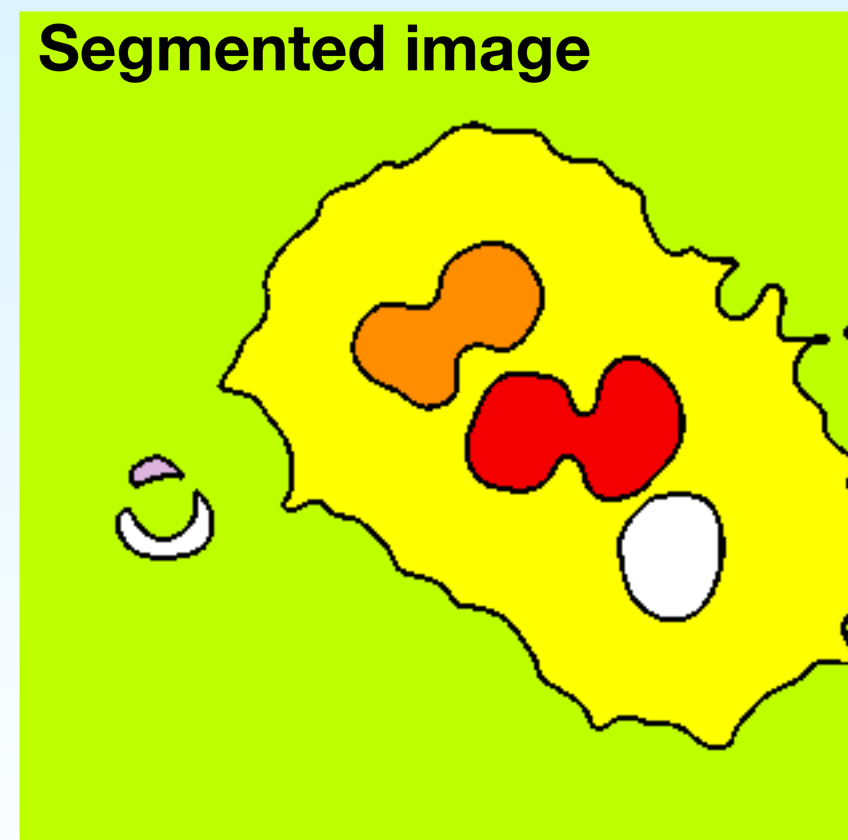
Source:

Smoothed laplacian filter







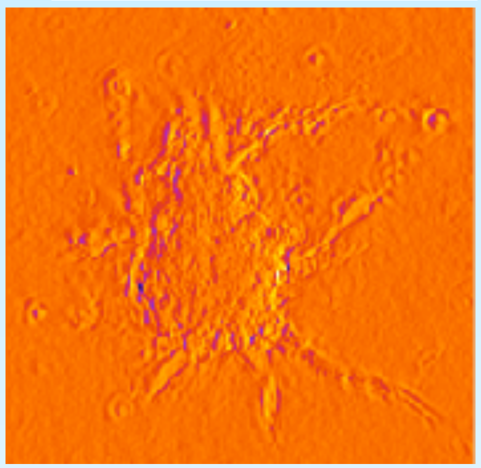


Source:

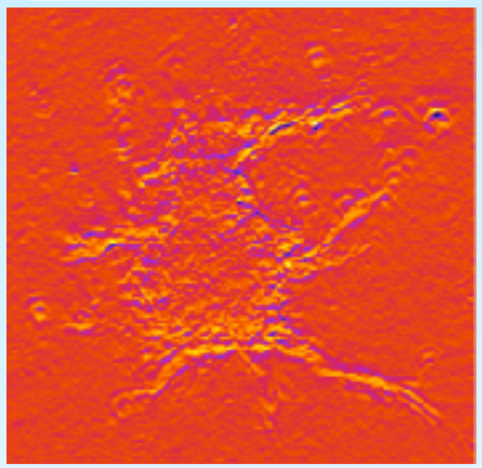
Feature calculation

If you filter and image with

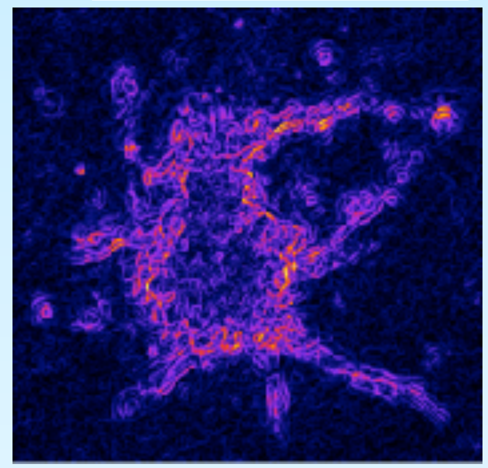
derivative-X



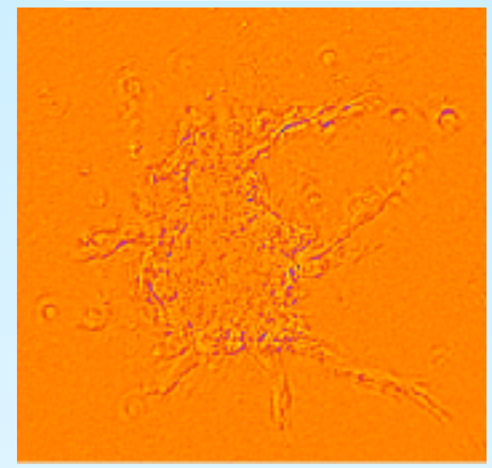
derivative-Y



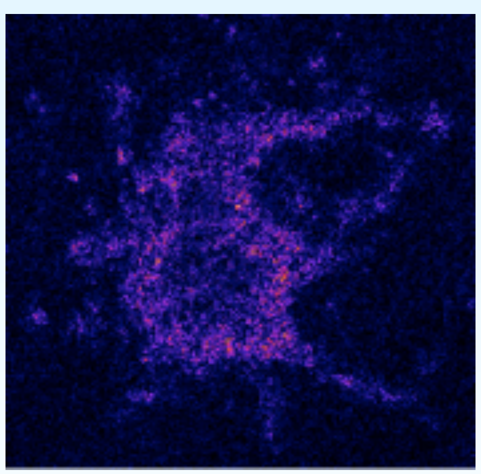
Edge mag.



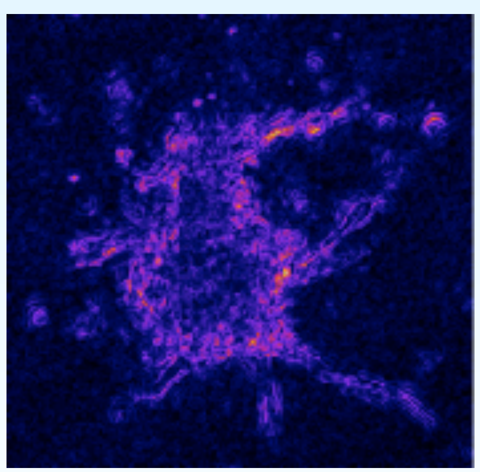
Laplacian



Hessian min

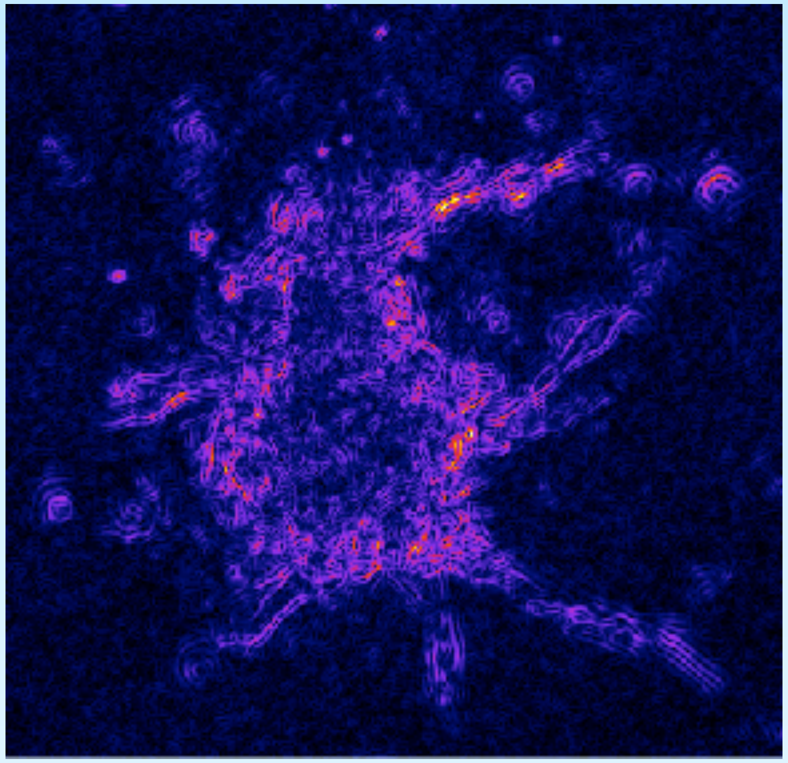


Hessian max

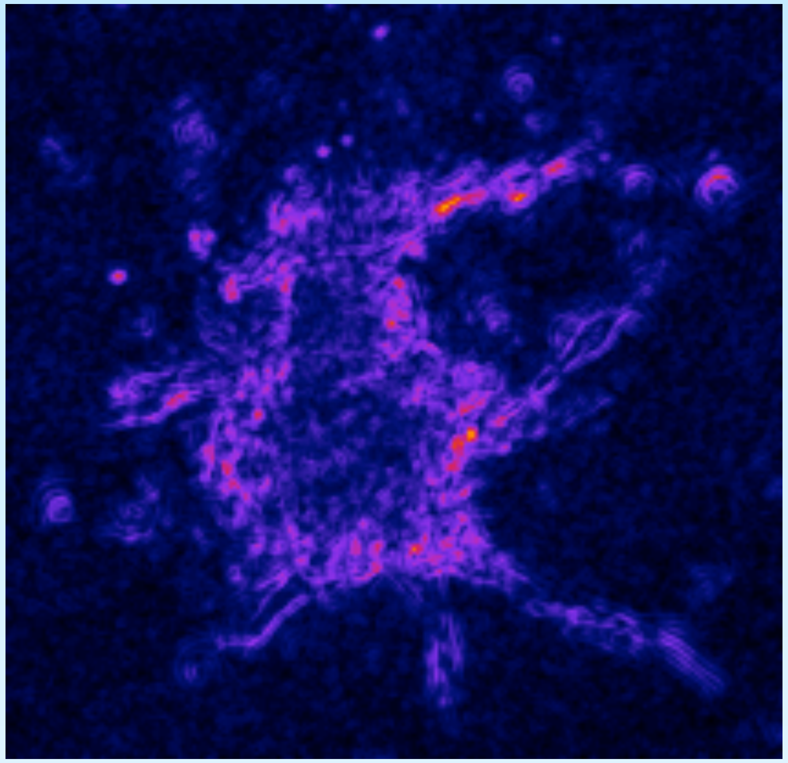


Source:FeatureJ

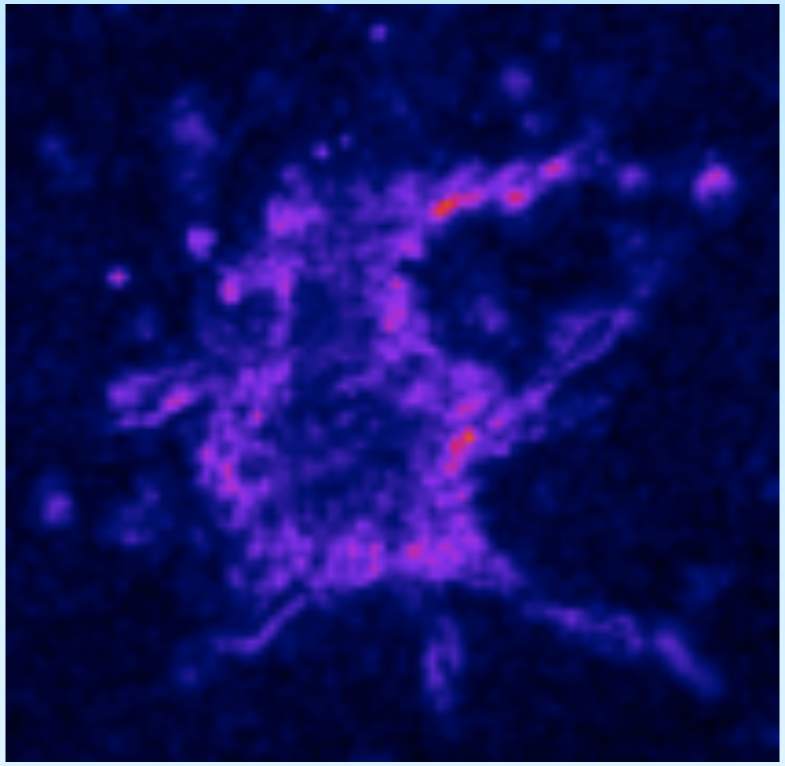
Feature Calculation different scales



sigma = 1.0

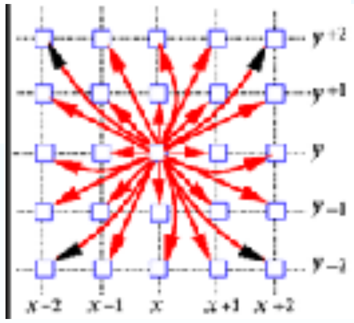


sigma = 4.0



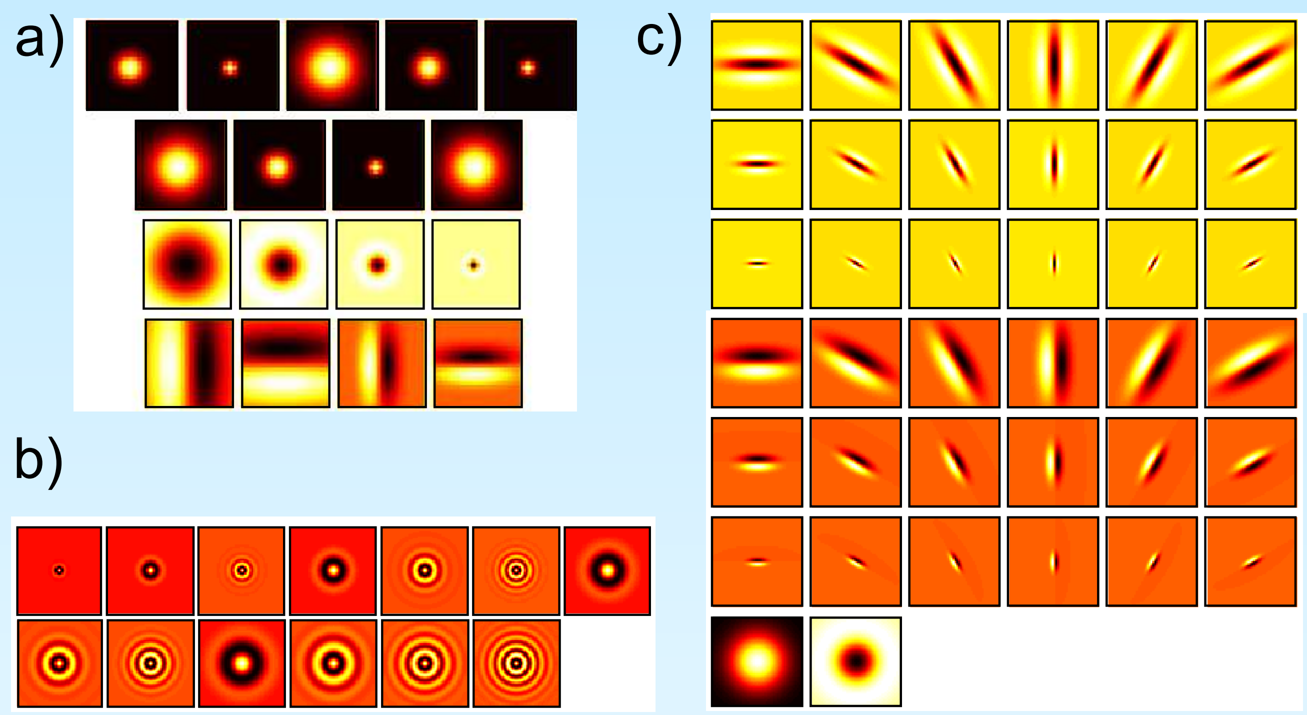
sigma = 8.0

By blurring the image and applying the different filters you get a sense of the greater neighbourhood. Many scales can be used in combination.



Source:

Filter banks



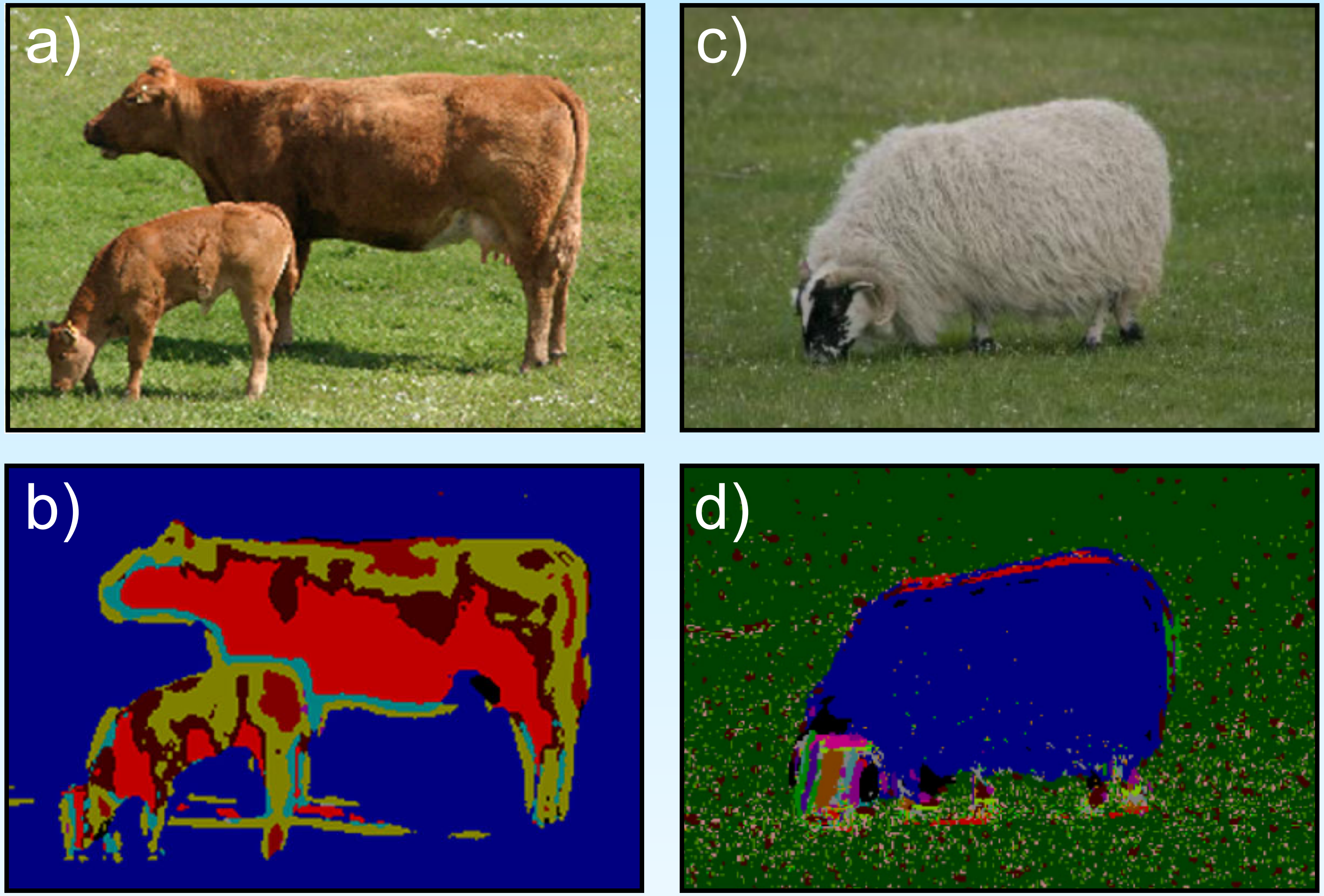
a) A combination of Gaussians, derivatives of Gaussians, Laplacians of Gaussians

b) Some rotationally invariant filters.

c) The maximum response MR8 filter bank

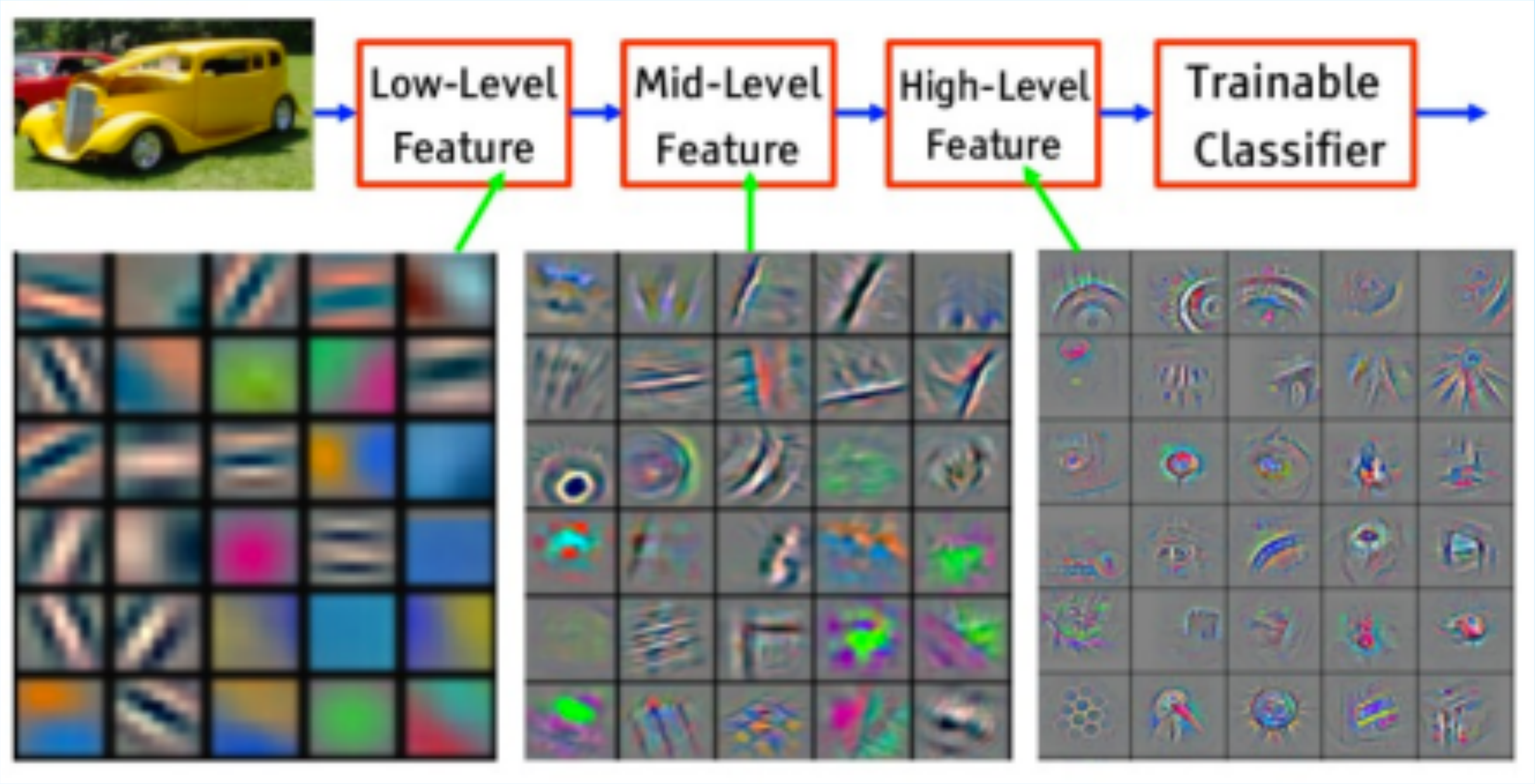
Source: Computer Vision, Simon Prince

Filters banks used for semantic segmentation



Source: Computer Vision, Simon Prince

These days filters banks are learnt using deep learning



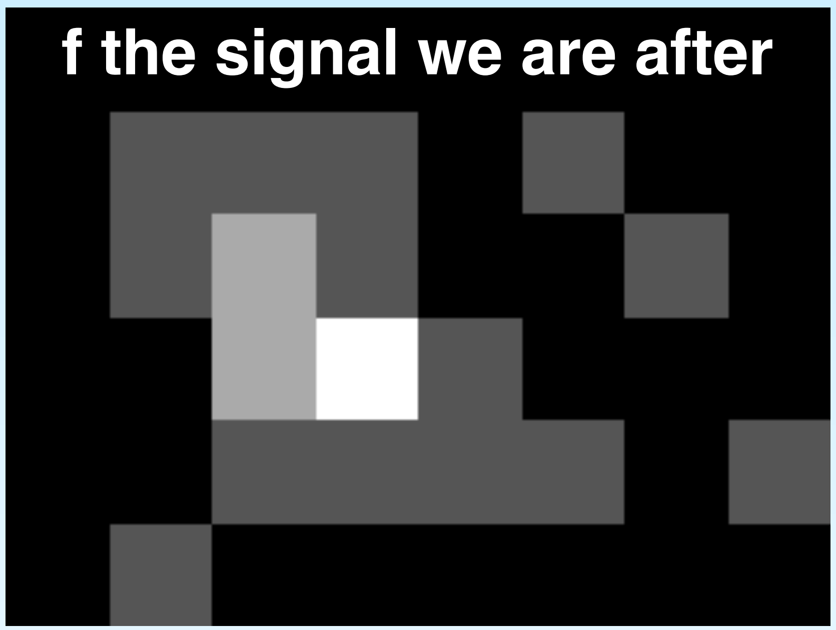
You will cover this in more detail later in the week.

Source: www.quora.com%2FWhat-is-a-convolutional-neural-network

Deconvolution

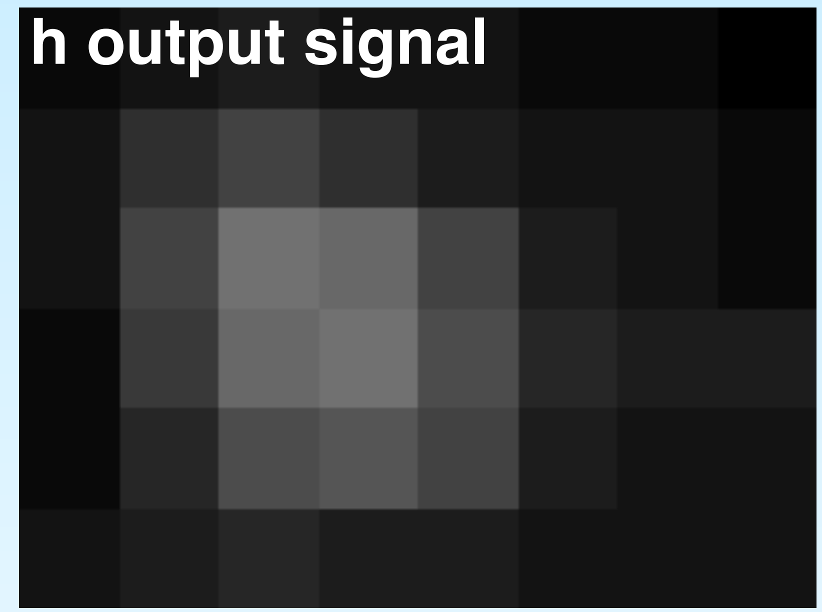
$$f * g = h$$

convolution



$$\begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}$$

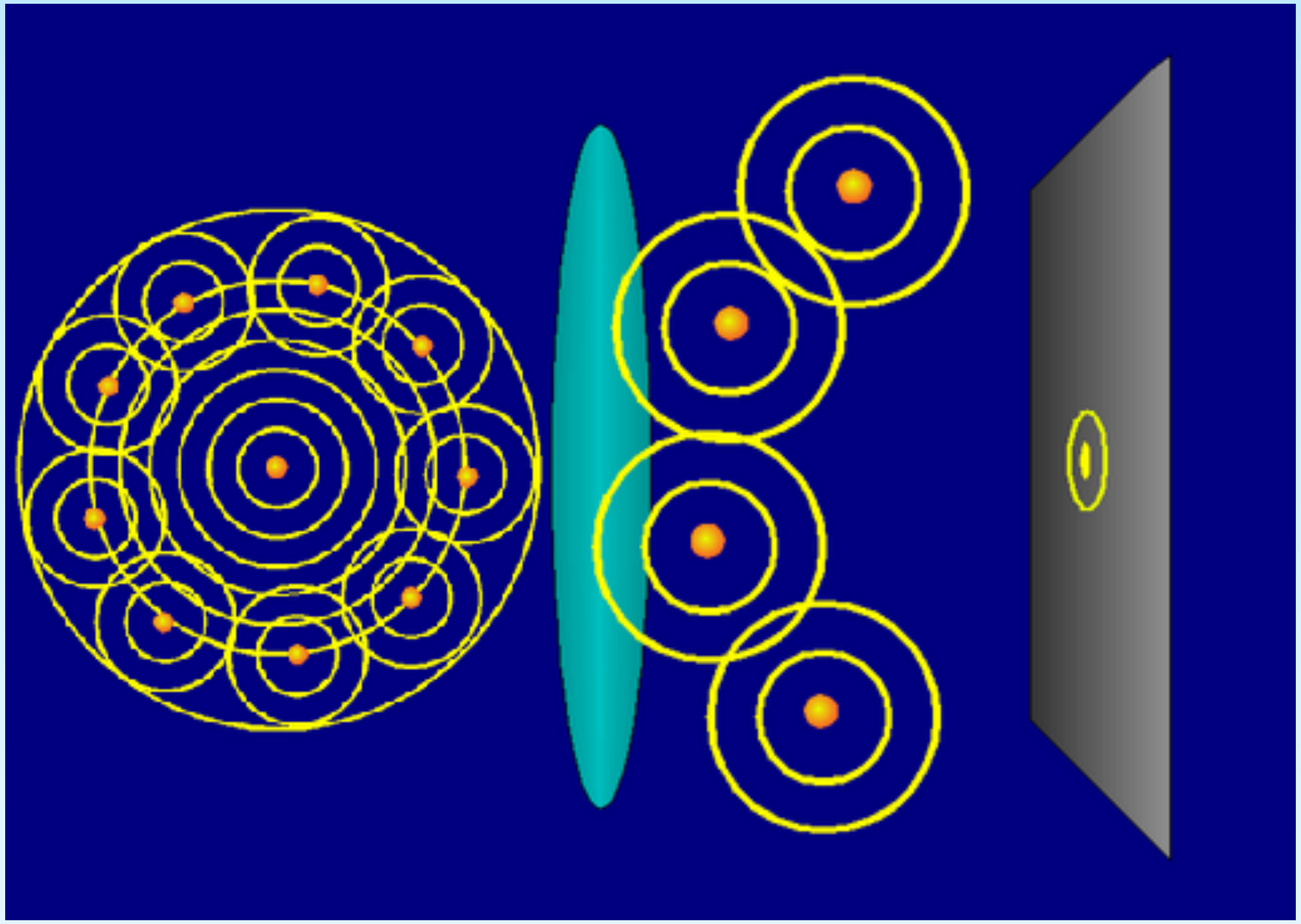
kernel: 3x3 mean filter

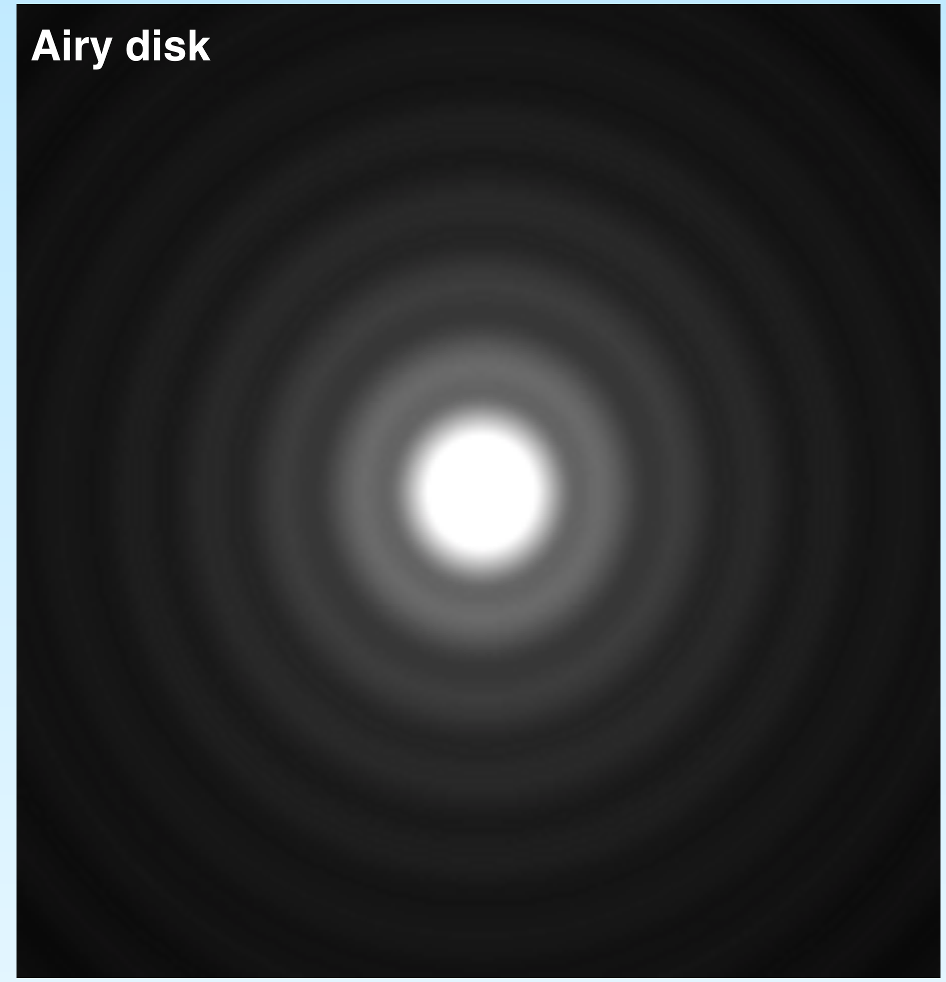


deconvolution

The goal of deconvolution is to find f given the kernel g and the output signal h

Deconvolution: Why we deconvolve?



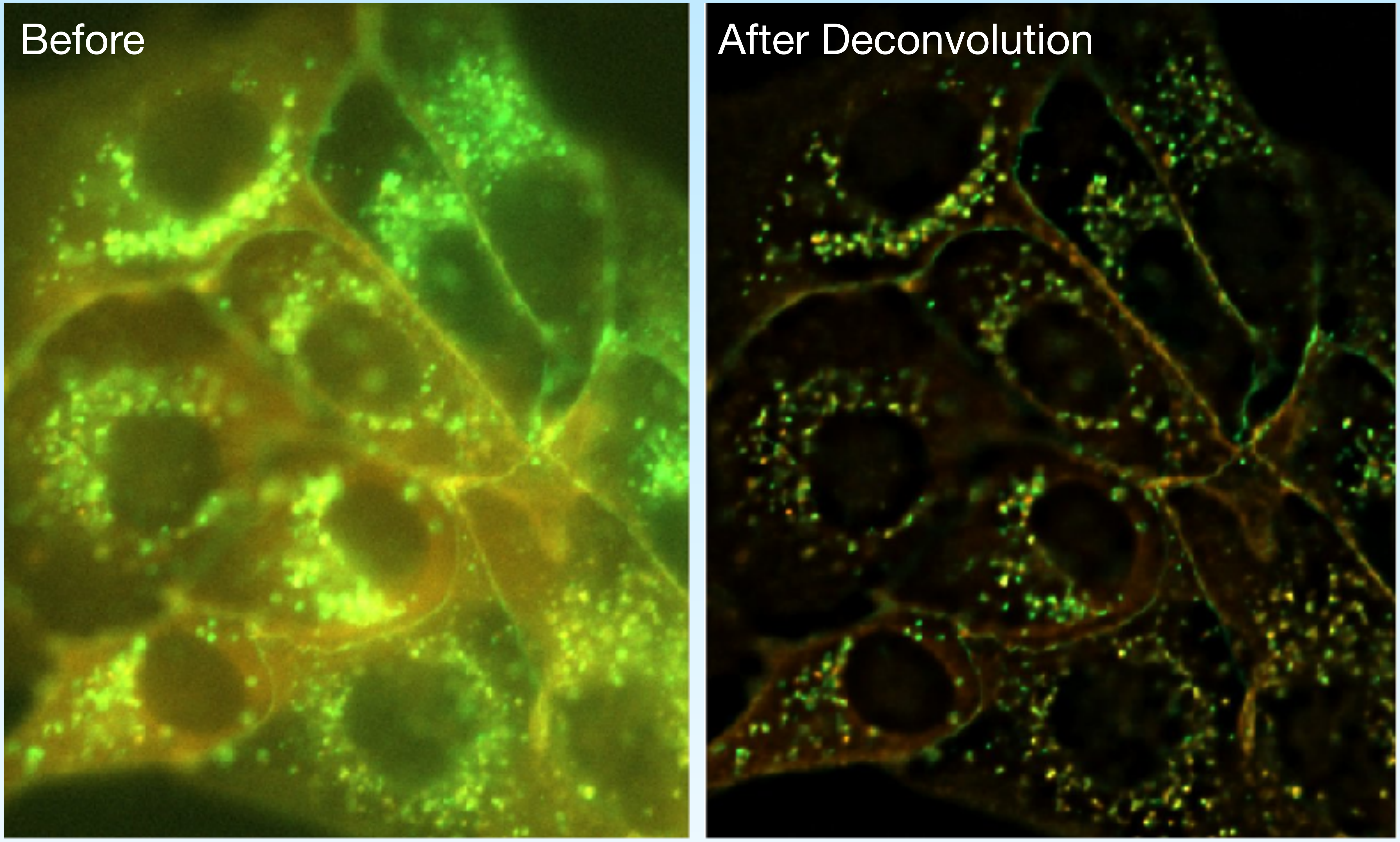


We have out-of-focus light contributing to our image plane, but we also have an artefact of the collection process influencing our final image.

Deconvolution seeks to remedy this issues.

Source: www.huygens.org

Huygens software

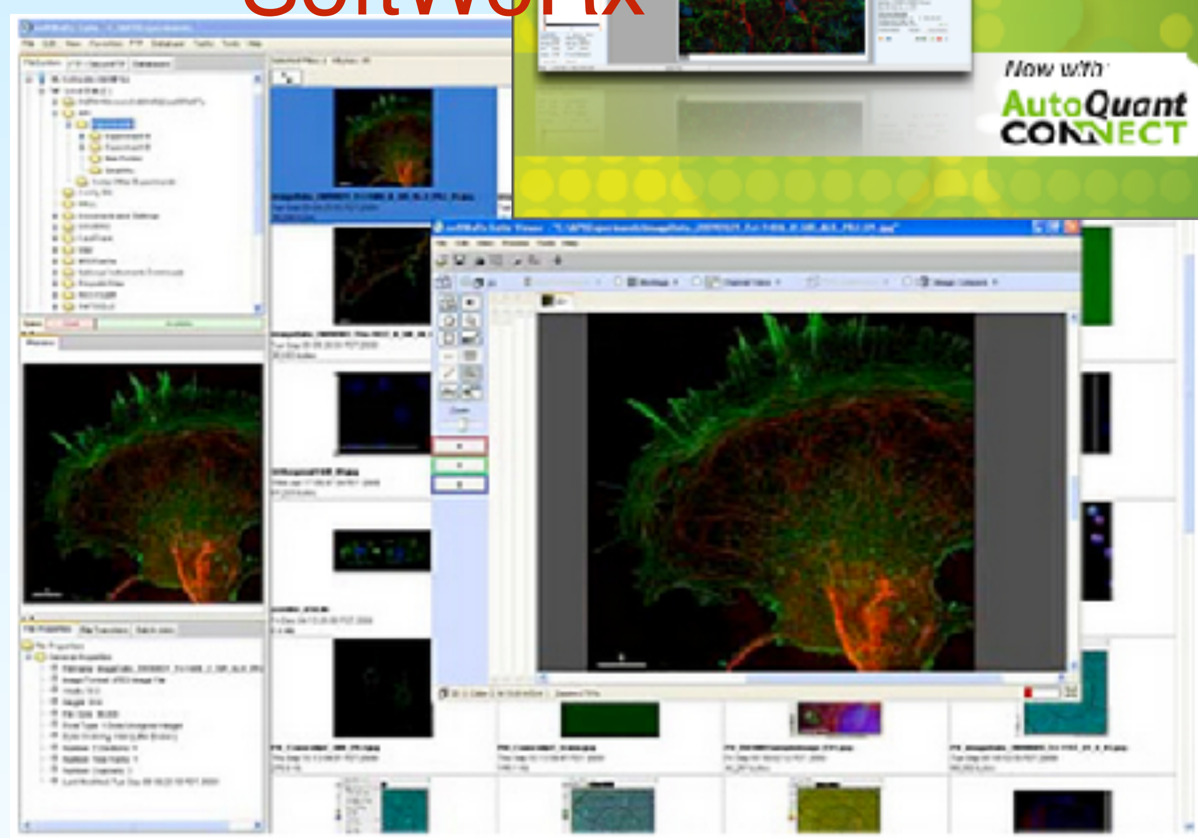


Deconvolution of cell-cell junctions of MDCK cells.
Nikon Ti widefield microscope (Objective 40x; 1.3 NA oil lens).

source: http://www.svi.nl/ImageGallery

Deconvolution software

SoftWoRx



Deconvolution

- Image restoration
- Volume Visualisation
- Some analysis

Expensive

WIMM CBRG has the 2nd fastest Huygens server in the world.

Source: http://www.svi.nl/HuygensSoftware, http://www.mediacy.com/index.aspx?page=AutoQuant, http://api.gehealthcare.com/api/softworx-suite.asp

Conclusions

- Image processing is fundamental to image analysis
- Filtering has many uses including noise removal
- Filtering can be linear or non-linear and plays a role in many different types of processing.
- Using many filters in coordination allows texture analysis of objects and pixels.

Thank you for time

https://twitter.com/dwaithe

https://github.com/dwaithe

dominic.waithe@imm.ox.ac.uk

BBSRC
bioscience for the future

Weatherall Institute of Molecular Medicine

NanO

Micron OXFORD

Nanoscopy Oxford

MRC Medical Research Council

EPSRC

BBSRC

UNIVERSITY OF OXFORD

Practical content and lectures

Doctoral Training Centre (MPLS) Resources /Modules/ 2017-18 / Michaelmas term / Week 9-10: Foundations of Image Analysis / day01_practicals.html

Summary of course: Timetable Template 2017-18.pdf

Practicals: day01_practicals.html

Source: