IMAGE PROCESSING

CS 3630 Introduction to Robotics and Perception

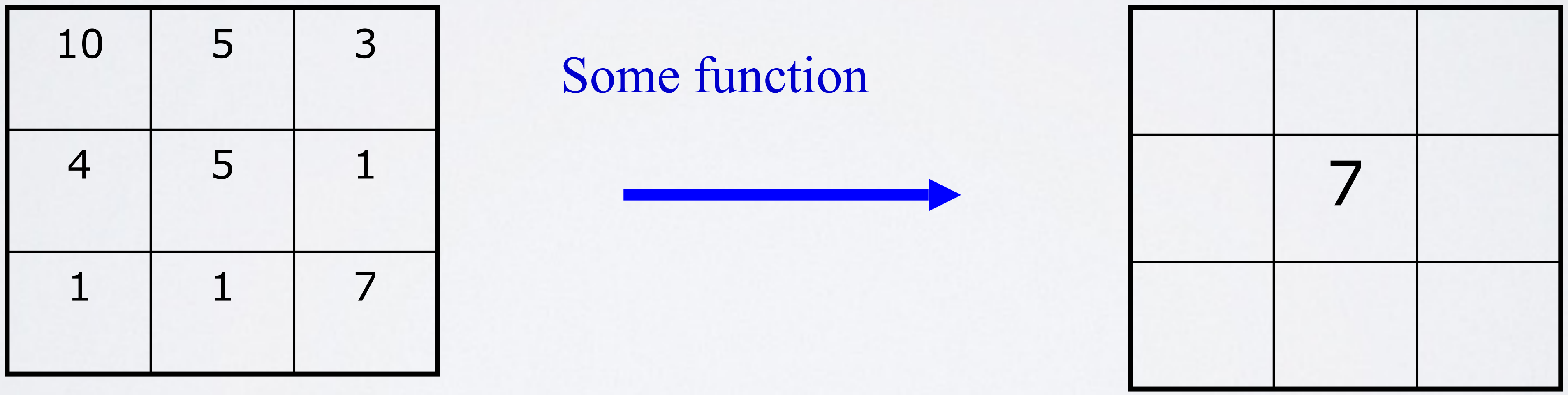
Frank Dellaert

FILTERING

- Noise removal
- Edge detection
- Texture description
- Multi-scale algorithms
- Feature detection
- Matched filters

WHAT IS IMAGE FILTERING?

Modify the pixels in an image based on some function of a local neighborhood of the pixels



LINEAR FILTERING

- Linear case is simplest and most useful
  - Replace each pixel with a linear combination of its neighbors.
- Prescription for linear combination is called the convolution kernel.

10	5	3
4	5	1
1	1	7

$*$

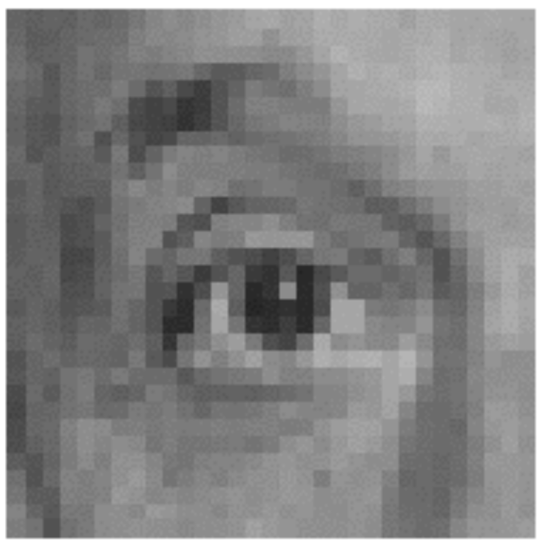
0	0	0
0	0.5	0
0	1.0	0.5

kernel

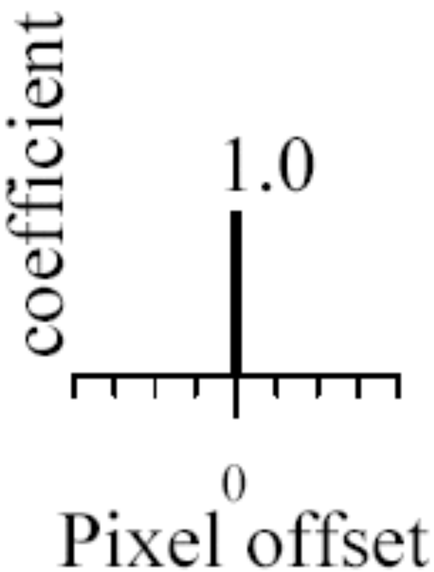
$=$

	7	

FILTERING EXAMPLES

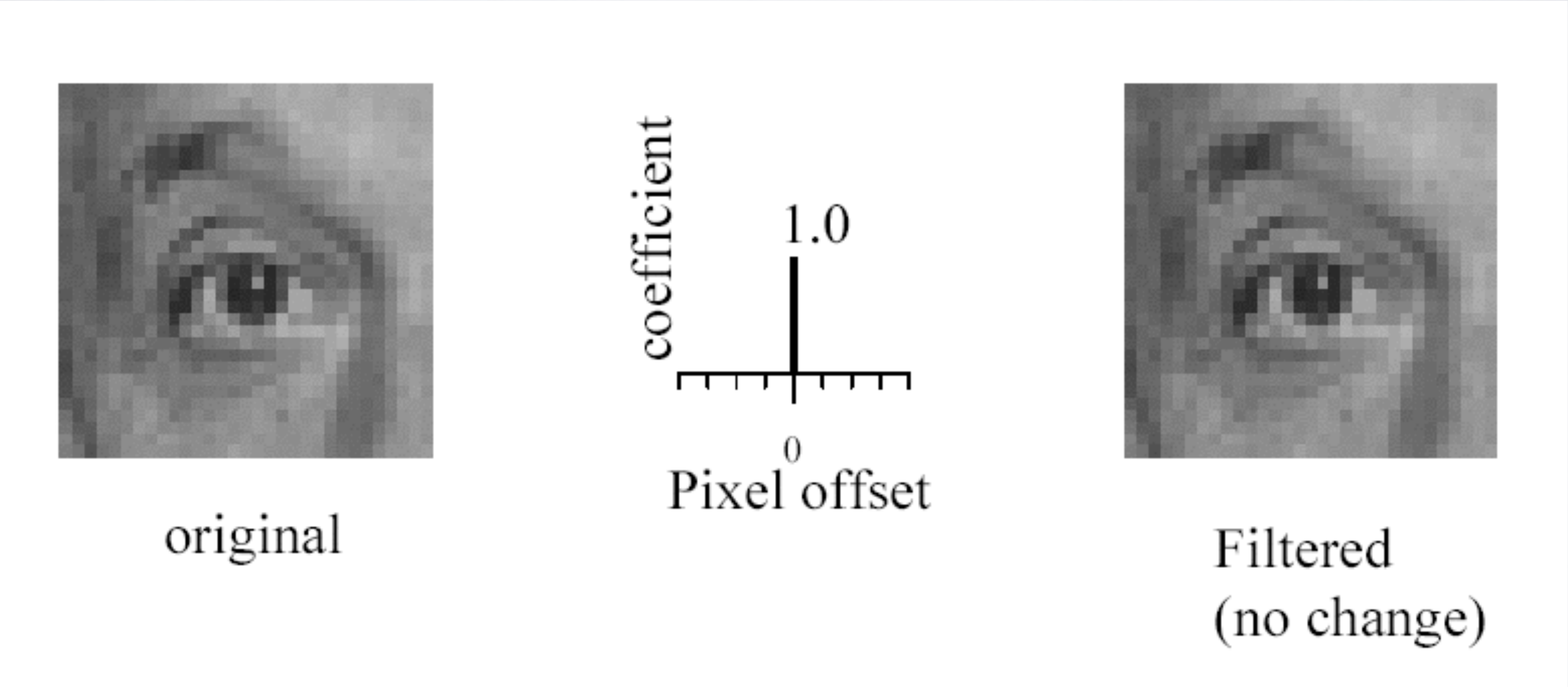


original

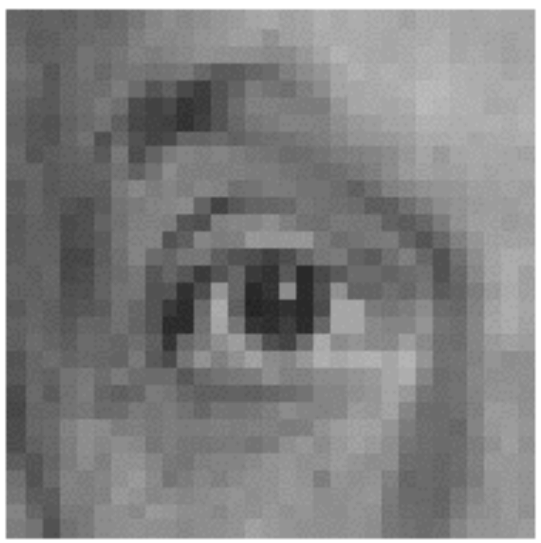


?

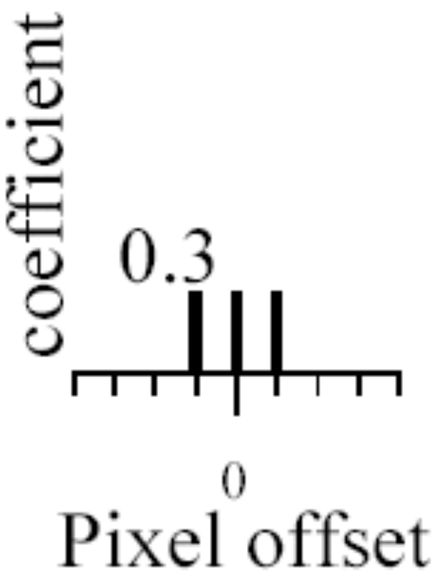
FILTERING EXAMPLES: IDENTITY



FILTERING EXAMPLES

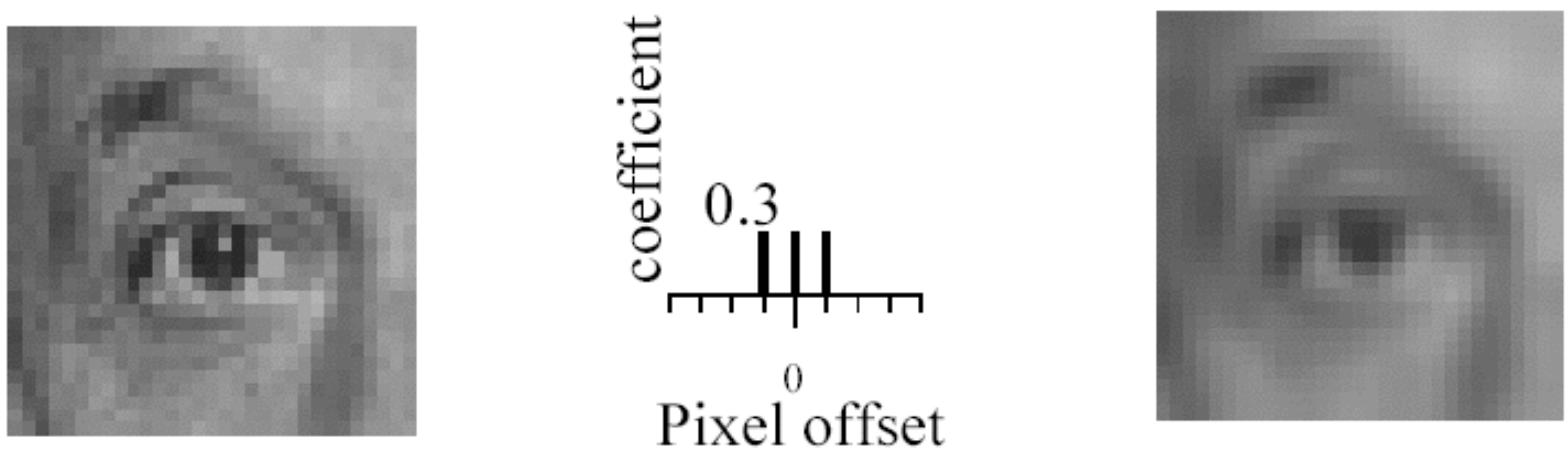


original



?

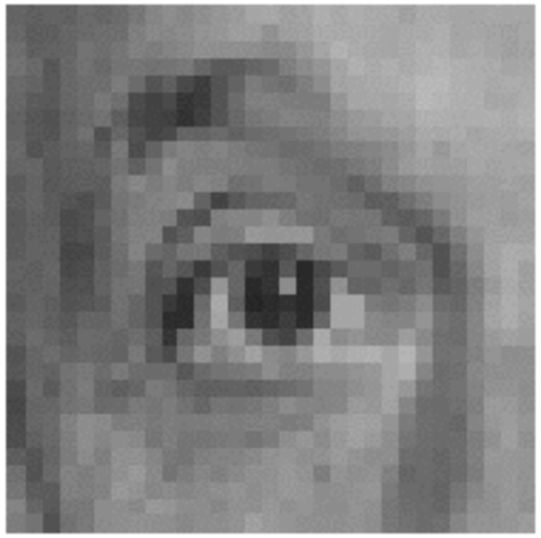
FILTERING EXAMPLES: BLUR



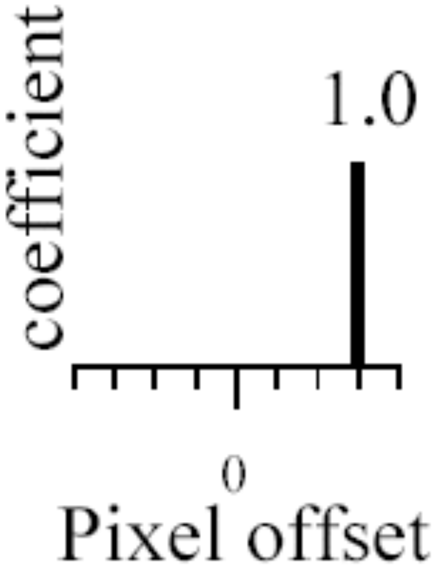
original

Blurred (filter applied in both dimensions).

FILTERING EXAMPLES

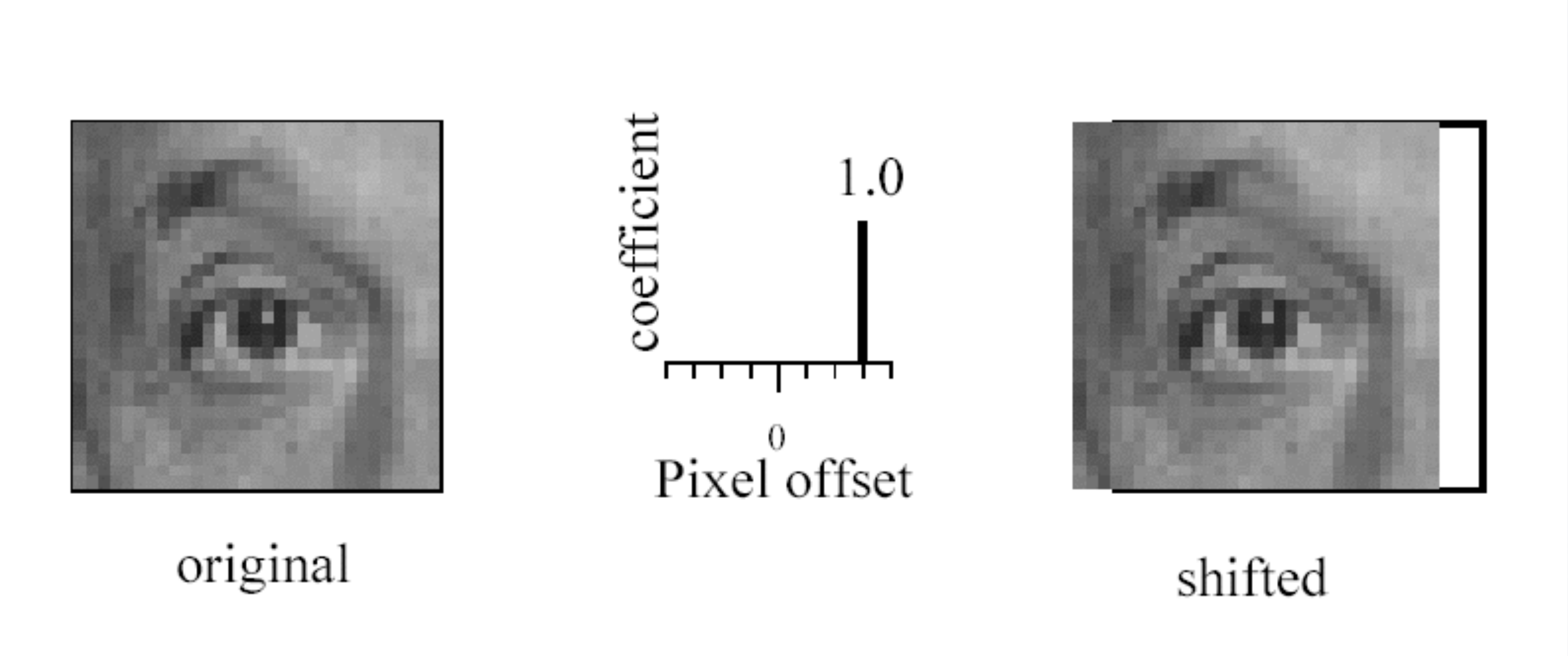


original



?

FILTERING EXAMPLES: SHIFT

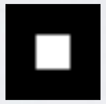


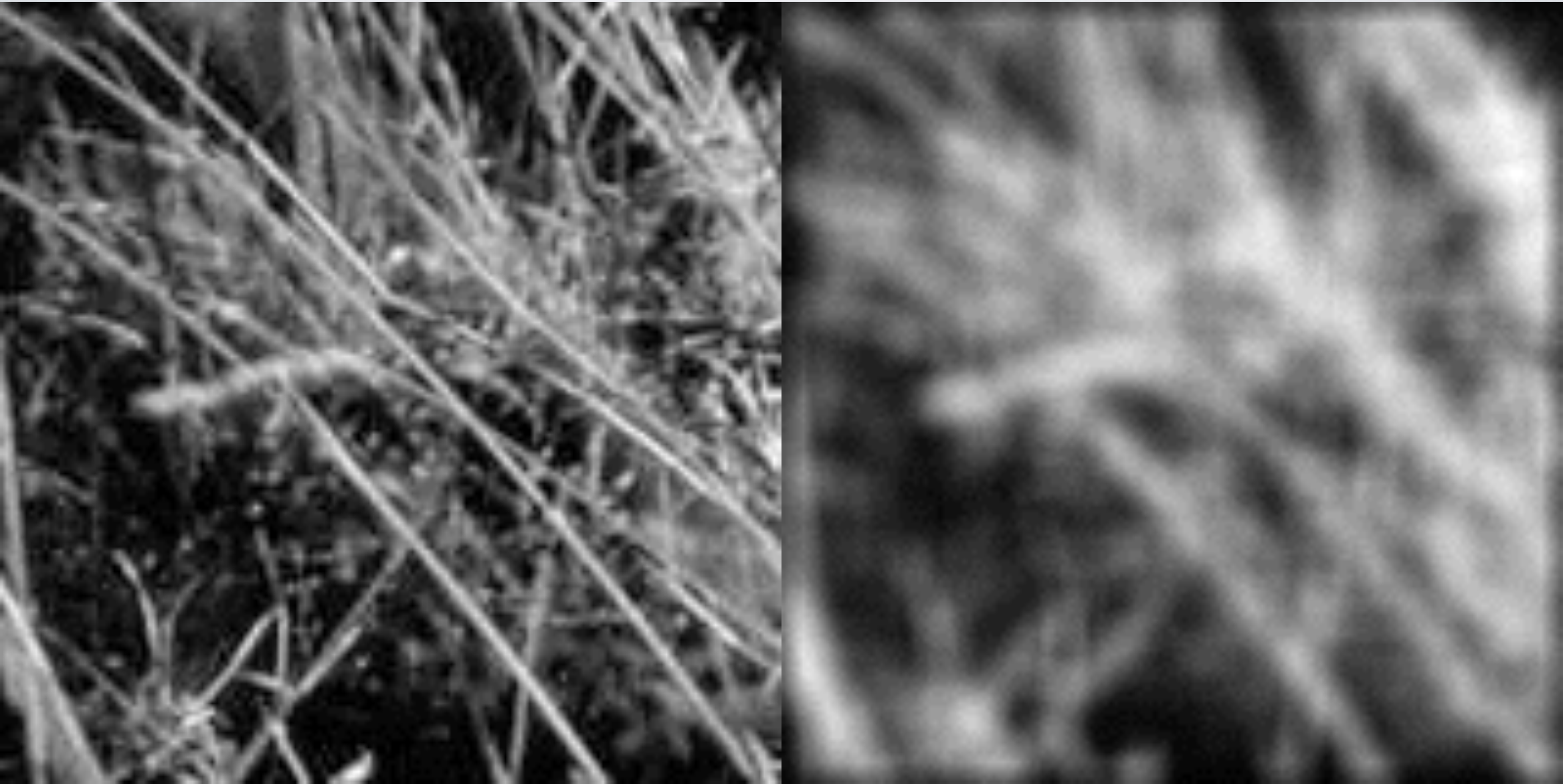
CONVOLUTION

$$R_{ij} = \sum_{u,v} H_{i-u,j-v} F_{uv}$$

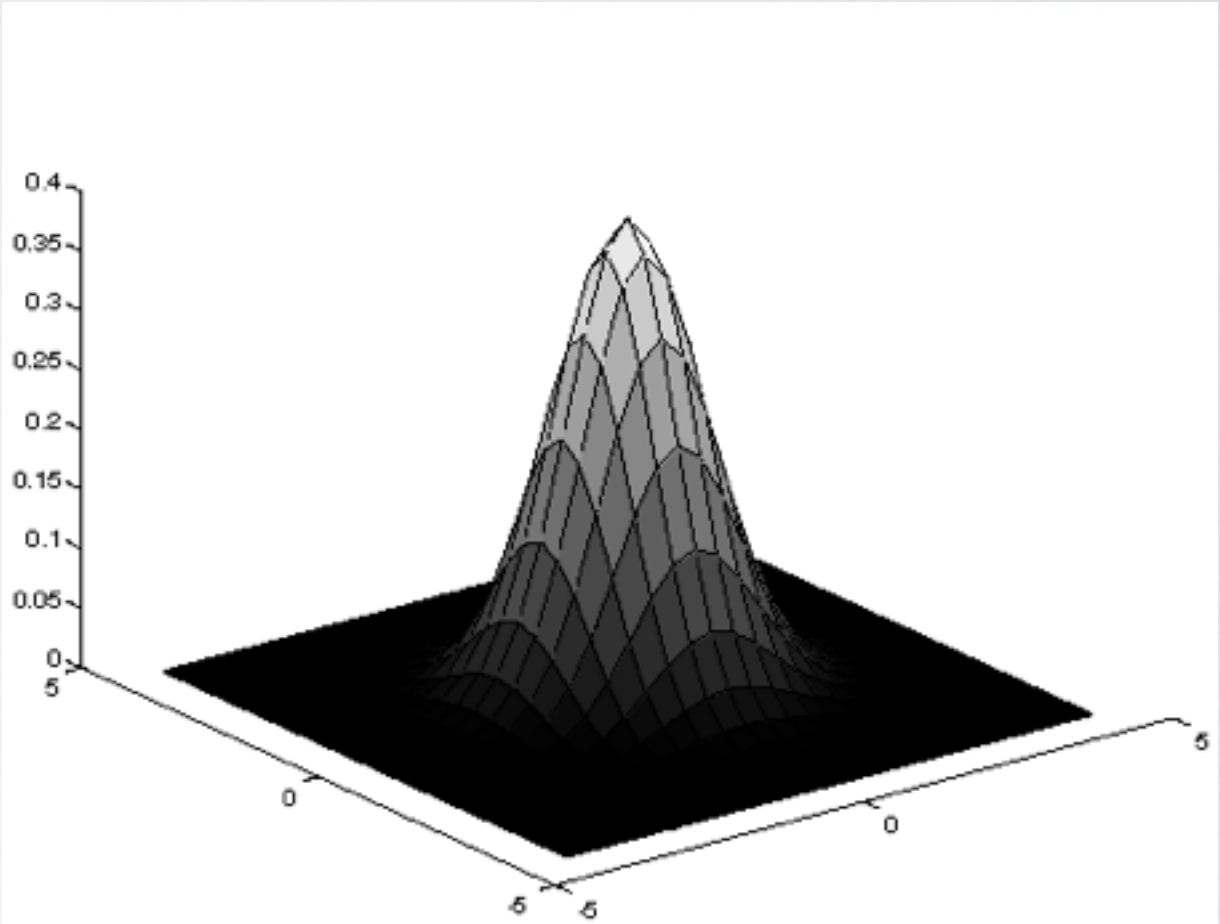
- Represent these weights as an image, H
- H is usually called the **kernel**
- Operation is called **convolution**

EXAMPLE: SMOOTHING BY AVERAGING





SMOOTHING WITH A GAUSSIAN



- Averaging does not model defocussed lens well
- impulse response should be fuzzy blob

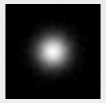
AN ISOTROPIC GAUSSIAN

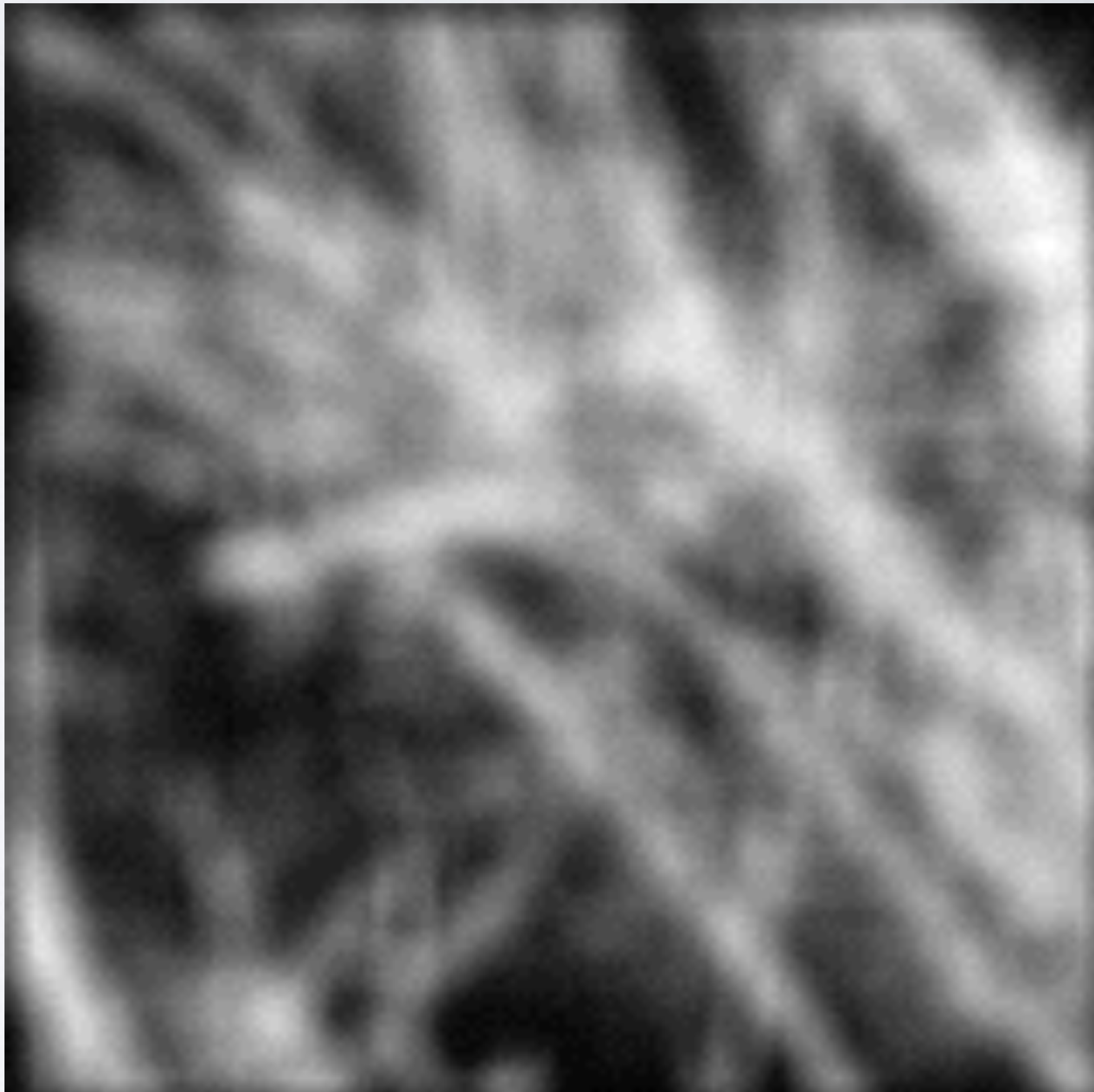
- The picture shows a smoothing kernel proportional to

$$\exp\left(-\left(\frac{x^2+y^2}{2\sigma^2}\right)\right)$$

- reasonable model of a circularly symmetric blob

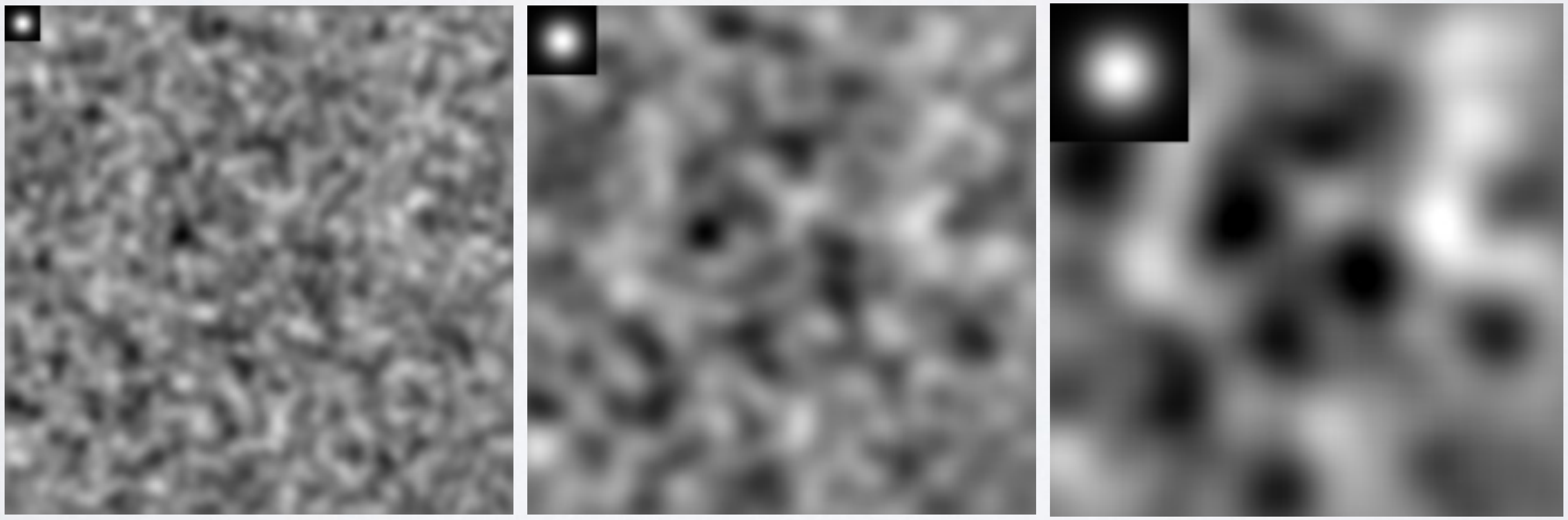
SMOOTHING WITH A GAUSSIAN



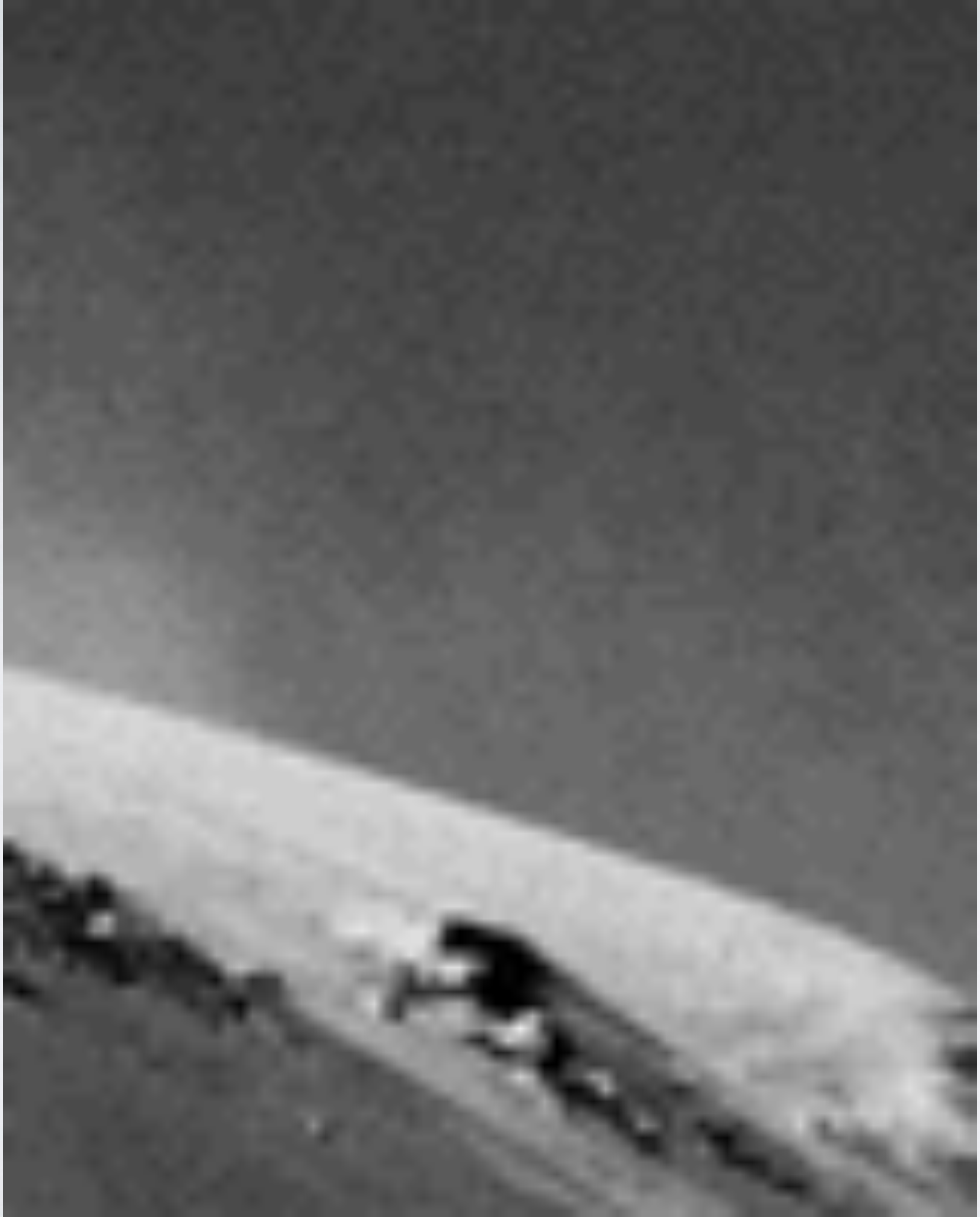


FILTER RESPONSES ARE CORRELATED

- Correlated over scales similar to scale of filter
- Filtered noise is sometimes useful
  - looks like some natural textures, can be used to simulate fire, etc.

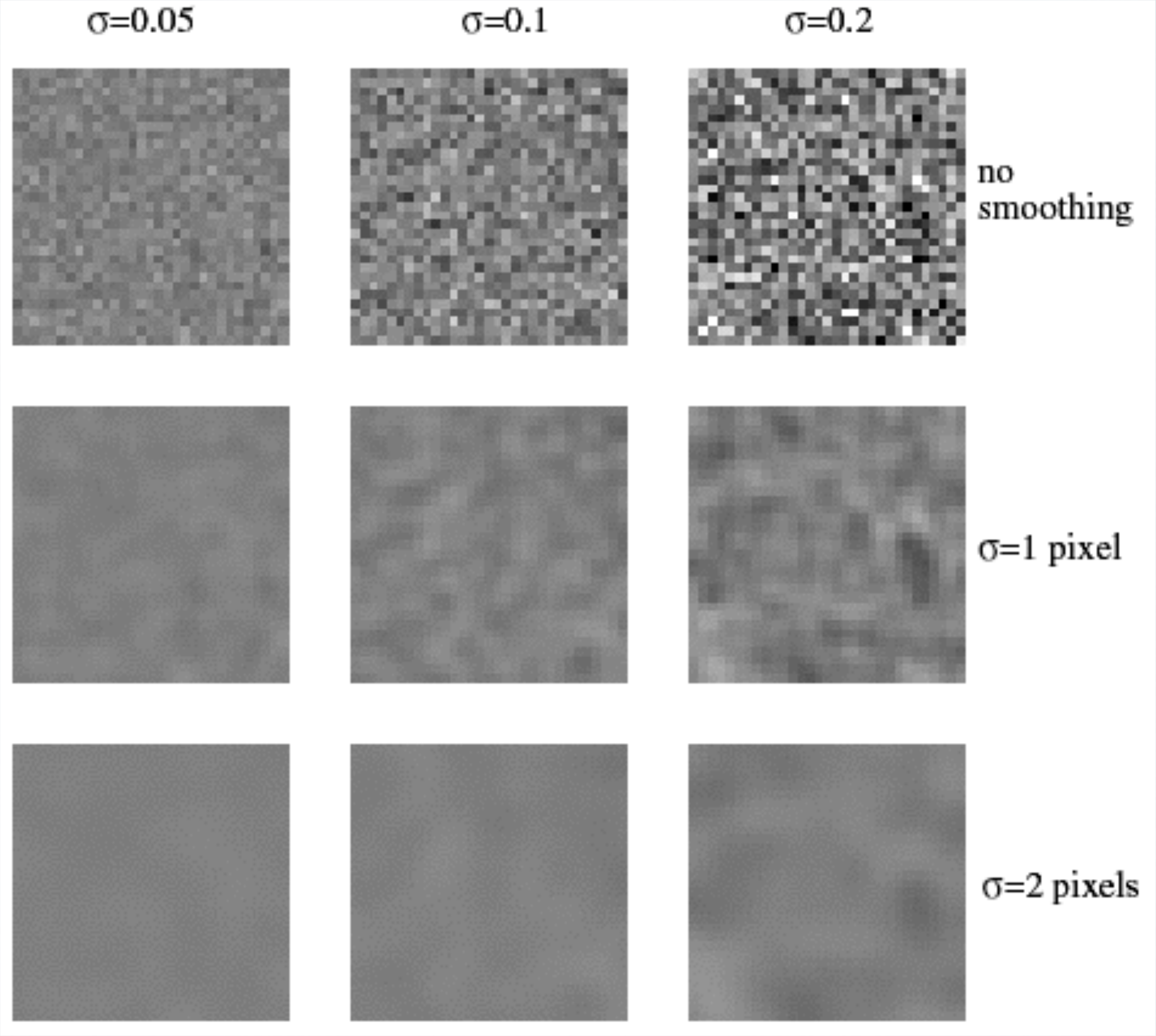


sigma=1



sigma=16

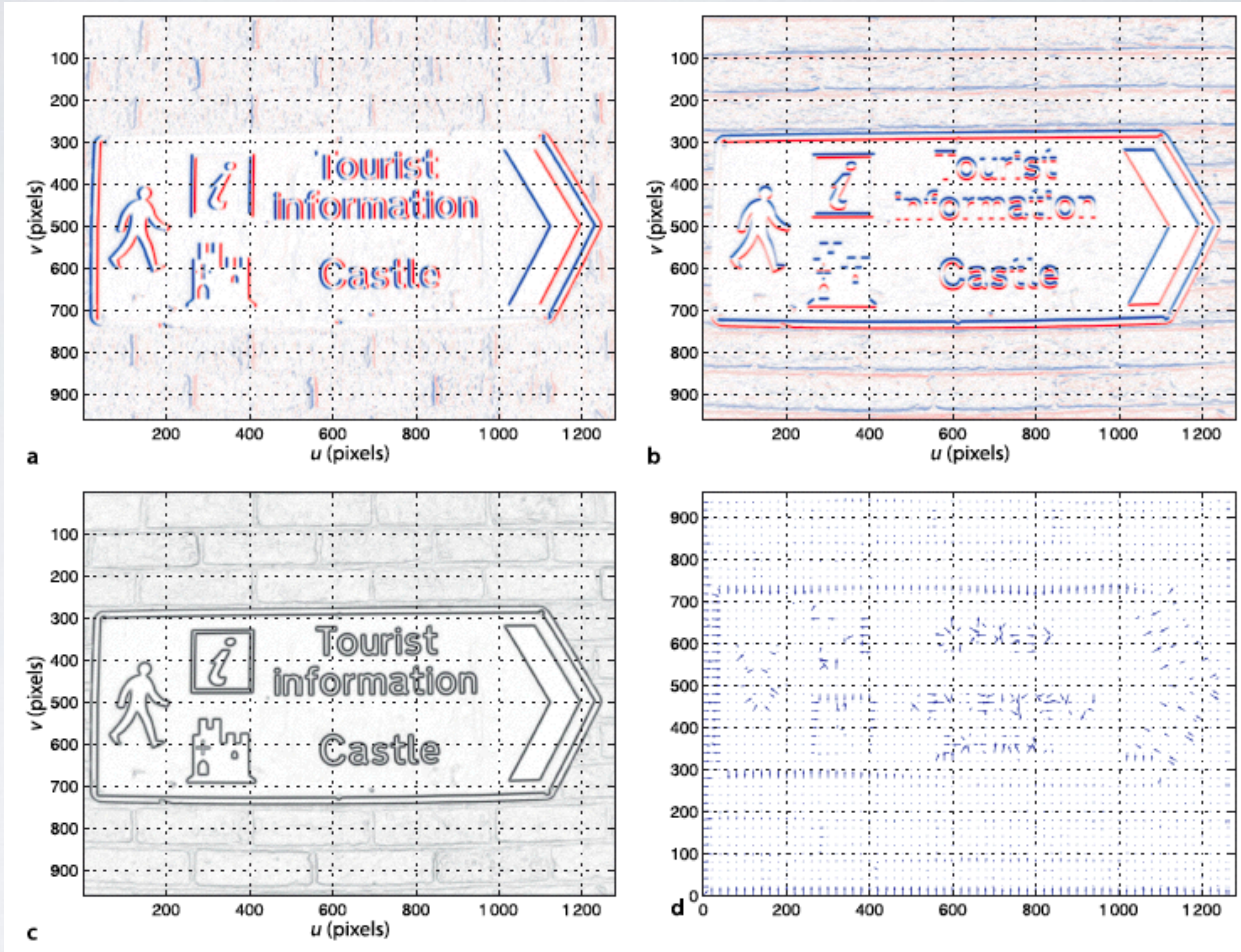
The effects of smoothing



EDGE DETECTION

```
>> Du = ksobel
Du =
    -1     0     1
    -2     0     2
    -1     0     1
```

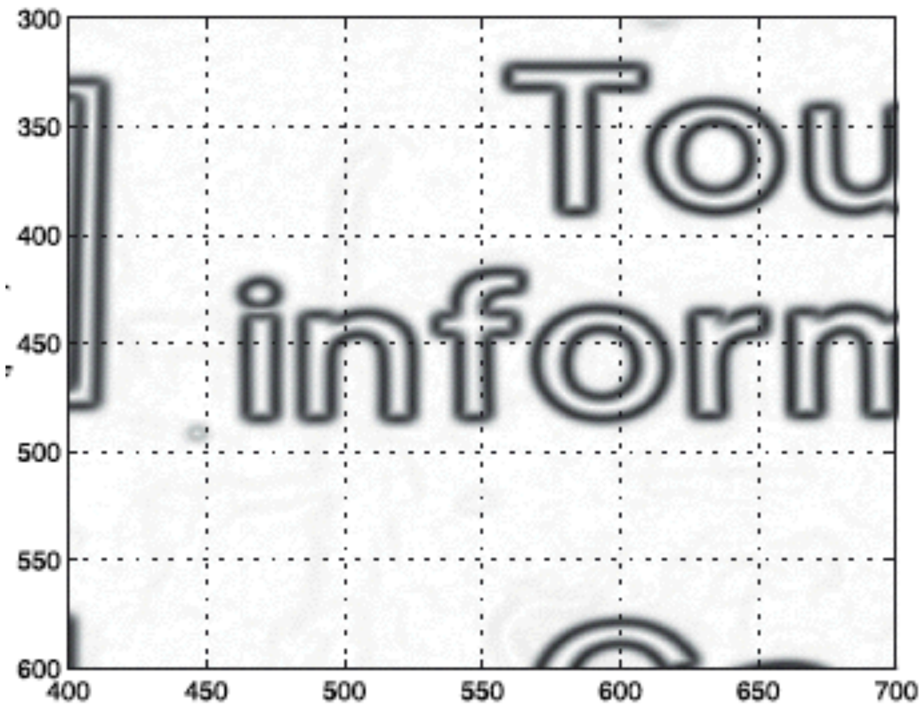
Magnitude & direction



- Sobel Kernel (Corke Chapter 12)

|DOG| VS. CANNY

- Derivative of Gaussian operator, then take magnitude
- Canny: smart post-processing of edge magnitude



$$\nabla \mathbf{I} = \mathbf{D} \otimes (\mathbf{G}(\sigma) \otimes I) = \underbrace{(\mathbf{D} \otimes \mathbf{G}(\sigma))}_{\text{DoG}} \otimes \mathbf{I}$$

$$\mathbf{G}_u(u, v) = -\frac{u}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}}$$

```
>> Iu = iconv( castle, -kdgauss(2) );
>> Iv = iconv( castle, -kdgauss(2)' );
>> m = sqrt( Iu.^2 + Iv.^2 );
```