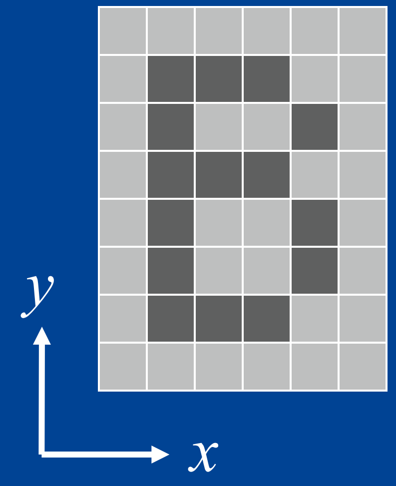
A Gentle Introduction to Bilateral Filtering and its Applications

# Naïve Image Smoothing: Gaussian Blur

*Sylvain Paris – Adobe*

# Notation and Definitions

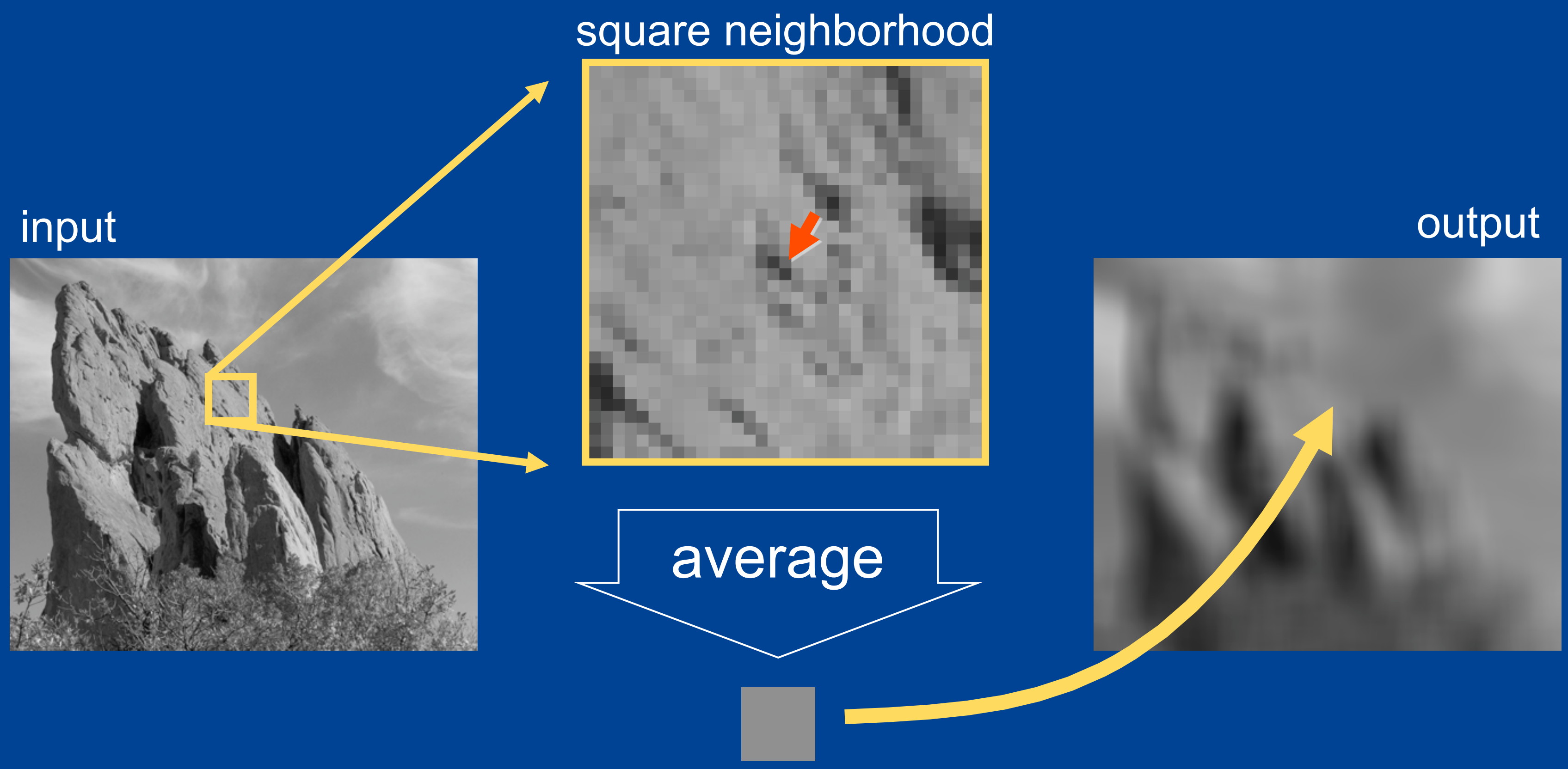



- Image = 2D array of pixels
- Pixel = intensity (scalar) or color (3D vector)
- $I_{\mathbf{p}}$ = value of image $I$ at position: $\mathbf{p} = ( p_x , p_y )$
- $F [ I ]$ = output of filter $F$ applied to image $I$

# Strategy for Smoothing Images

- Images are not smooth because adjacent pixels are different.
- Smoothing = making adjacent pixels look more similar.
- Smoothing strategy
  pixel → average of its neighbors

# Box Average



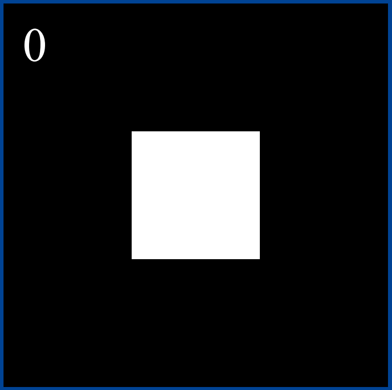
# Equation of Box Average

$$BA[I]_{\mathbf{p}} = \sum_{\mathbf{q}\in S} B_{\sigma}(\mathbf{p}-\mathbf{q})\, I_{\mathbf{q}}$$

result at pixel $\mathbf{p}$

sum over all pixels $\mathbf{q}$

normalized box function




intensity at pixel $\mathbf{q}$

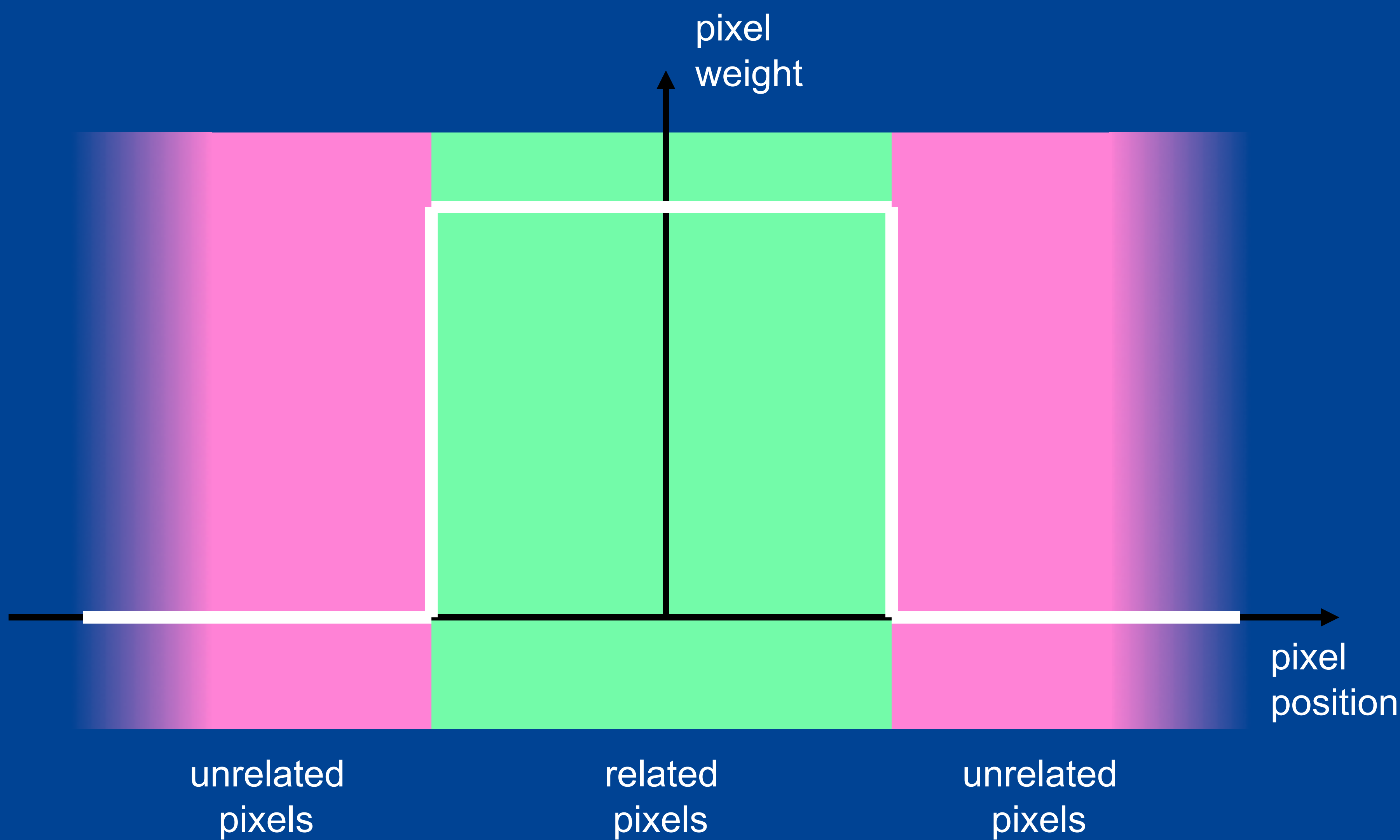
# Square Box Generates Defects

- Axis-aligned streaks
- Blocky results

output

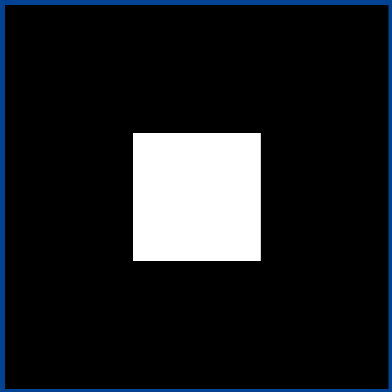
input

# Box Profile

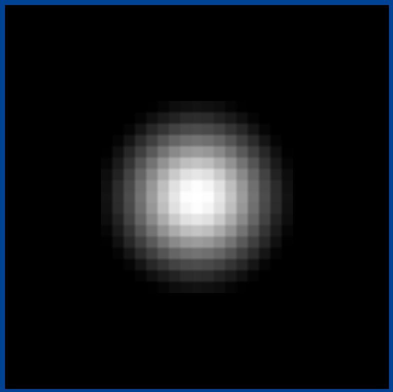

pixel
weight
pixel
position
unrelated
pixels
related
pixels
unrelated
pixels

# Strategy to Solve these Problems

- Use an isotropic (*i.e.* circular) window.
- Use a window with a smooth falloff.

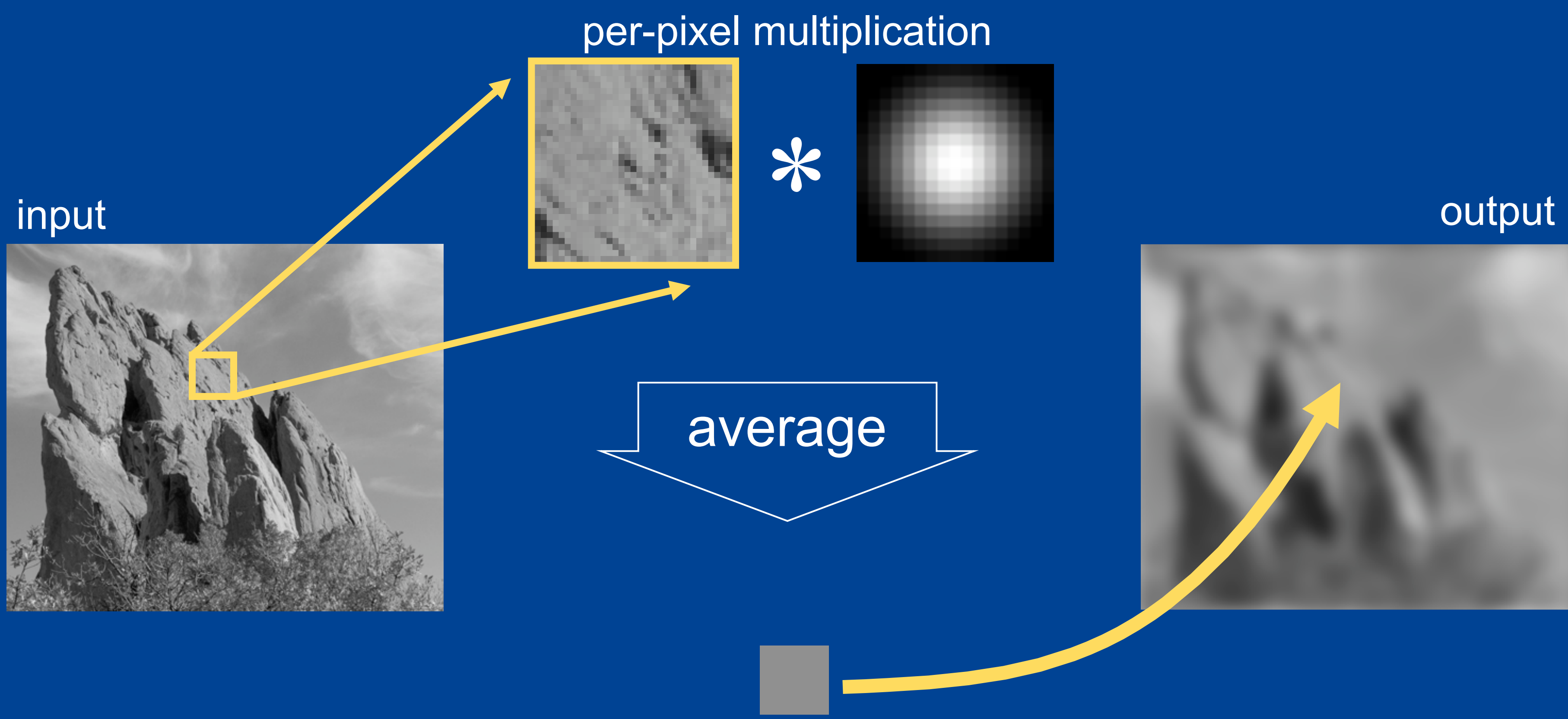


box window



Gaussian window

# Gaussian Blur




input

# box average

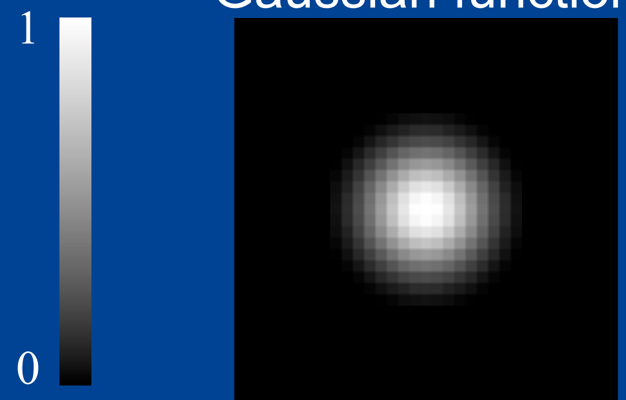

Gaussian blur

# Equation of Gaussian Blur

Same idea: **weighted average of pixels**.

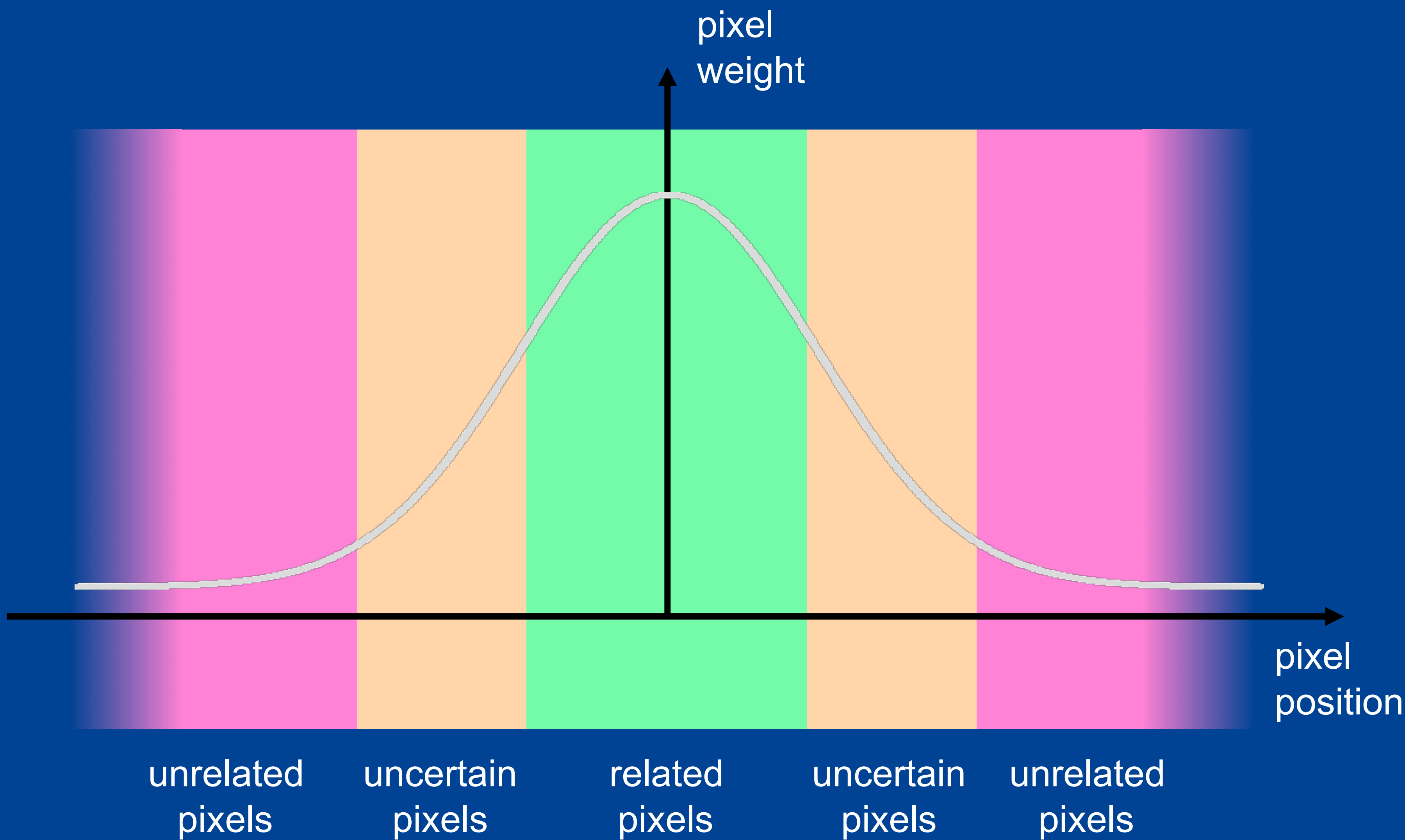
$$GB[I]_{\mathbf{p}} = \sum_{\mathbf{q} \in S} G_{\sigma}(\| \mathbf{p} - \mathbf{q} \|) I_{\mathbf{q}}$$

normalized
Gaussian function



# Gaussian Profile

$$G_\sigma(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

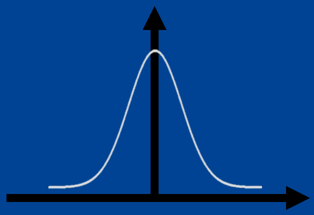


# Spatial Parameter

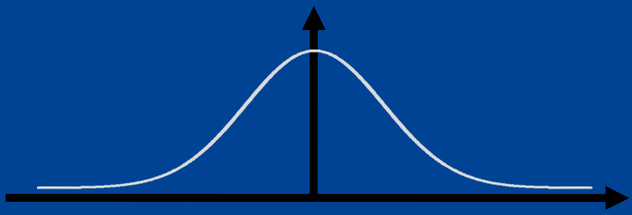
input

$$GB[I]_{\mathbf{p}} = \sum_{\mathbf{q} \in S} G_{\sigma}(\| \mathbf{p} - \mathbf{q} \|) I_{\mathbf{q}}$$

$\sigma$ : size of the window



small $\sigma$



large $\sigma$

limited smoothing

strong smoothing

# How to set $\sigma$

- Depends on the application.

- Common strategy: proportional to image size
  - e.g. 2% of the image diagonal
  - property: independent of image resolution

# Properties of Gaussian Blur

- Weights independent of spatial location
  - linear convolution
  - well-known operation
  - efficient computation (recursive algorithm, FFT…)

# Properties of Gaussian Blur

- Does smooth images
- But smoothes too much: **edges are blurred**.
  - Only spatial distance matters
  - No edge term

$$GB[I]_{\mathbf{p}} = \sum_{\mathbf{q}\in S} \underbrace{G_{\sigma}\left(\|\mathbf{p}-\mathbf{q}\|\right)}_{\text{space}} I_{\mathbf{q}}$$

input

output