Image Processing I

Computer Vision: CS 600.461/661

Image Processing I

Transform image to new one that is easier to manipulate.

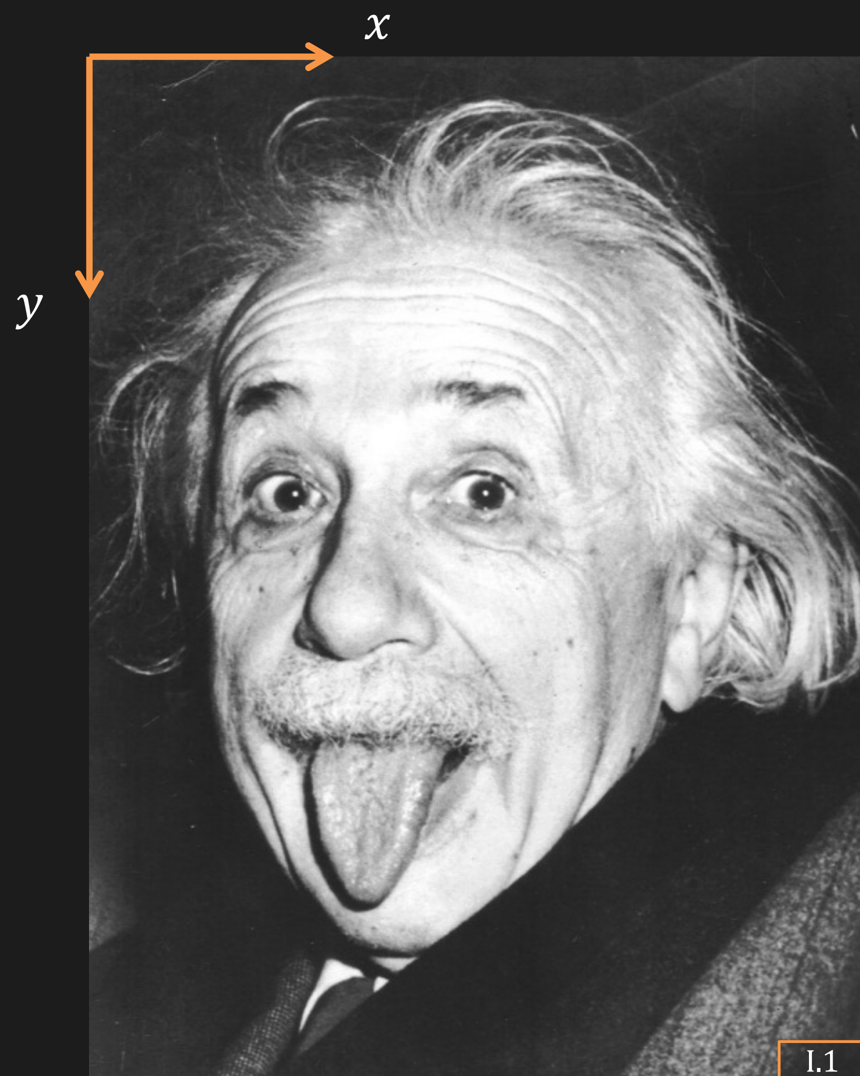
Topics:

(1) Pixel Processing

(2) Convolution Introduction

Computer Vision: Algorithms and Applications (Chapter 3.2)
Szelinski, 2011 (available online)

Image as a Function



$f(x, y)$ is the image intensity at position (x, y)

Image Processing

Transformation t of one image f to another image g

$$g(x,y) = t(f(x,y))$$

Point (Pixel) Processing

Darken $(f - 128)$

Original (f)

Lighten $(f + 128)$

Invert $(255 - f)$

Point (Pixel) Processing

Low Contrast $(f/2)$

Original (f)

High Contrast $(f * 2)$

Gray

$(0.3f_R + 0.6f_G + 0.1f_B)$

Linear Shift Invariant System

$$f(x) \longrightarrow \boxed{\text{LSIS}} \longrightarrow g(x)$$

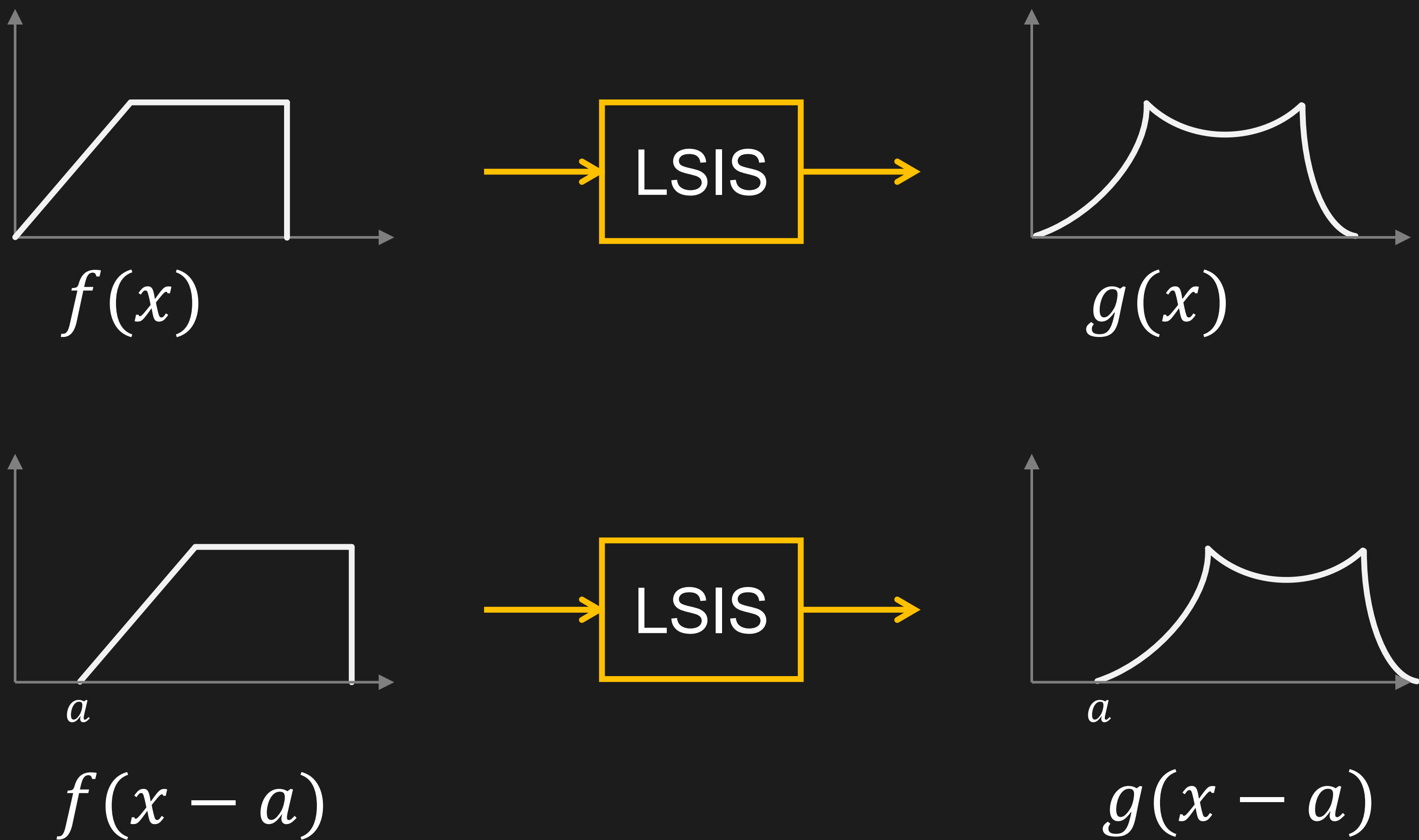
Study of Linear Shift Invariant Systems (LSIS) leads to useful image processing algorithms.

LSIS: Linearity

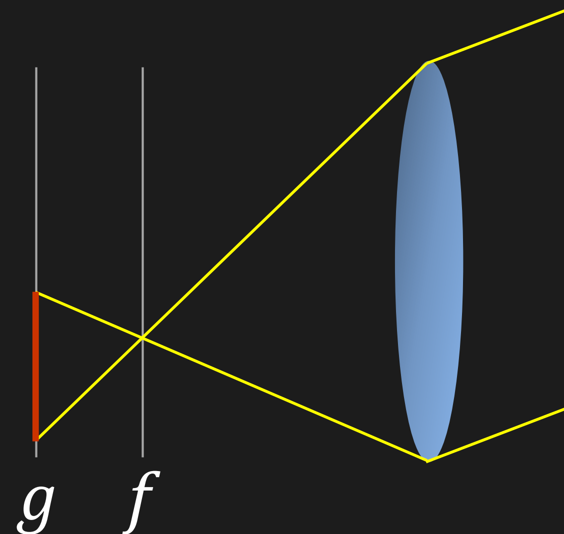
$$f_1 \longrightarrow \boxed{\text{LSIS}} \longrightarrow g_1 \qquad\qquad f_2 \longrightarrow \boxed{\text{LSIS}} \longrightarrow g_2$$

$$\alpha f_1 + \beta f_2 \longrightarrow \boxed{\text{LSIS}} \longrightarrow \alpha g_1 + \beta g_2$$

LSIS: Shift Invariance



Ideal Lens is an LSIS



$$f(x) \longrightarrow \boxed{\text{LSIS}} \longrightarrow g(x)$$

Defocused Image (g) is a Processed version of Focused Image (f)

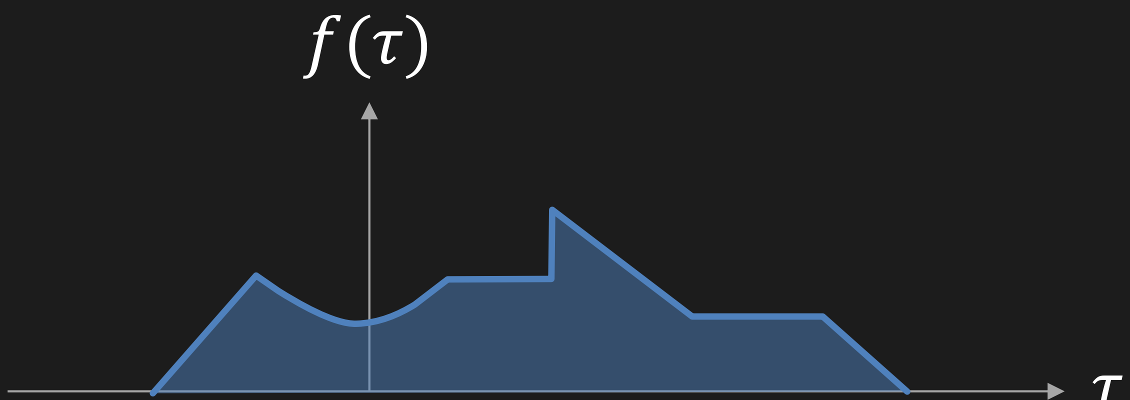
Linearity: Brightness variation

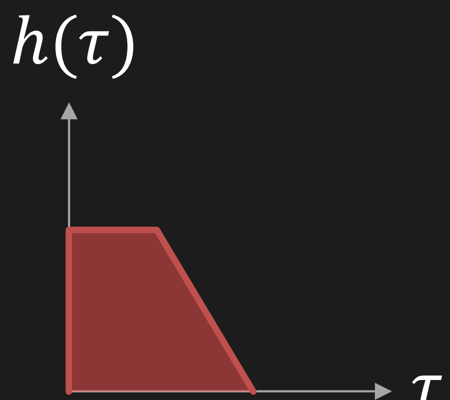
Shift invariance: Scene movement

Convolution (Important Concept)

Convolution of two functions $f(x)$ and $h(x)$

$$\boxed{g(x) = f(x) * h(x) = \int_{-\infty}^{\infty} f(\tau)h(x-\tau)\,d\tau}$$

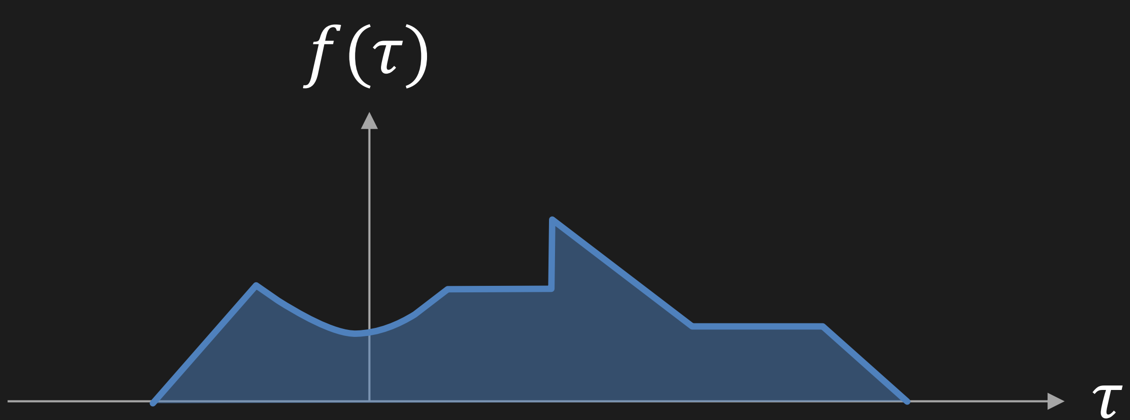


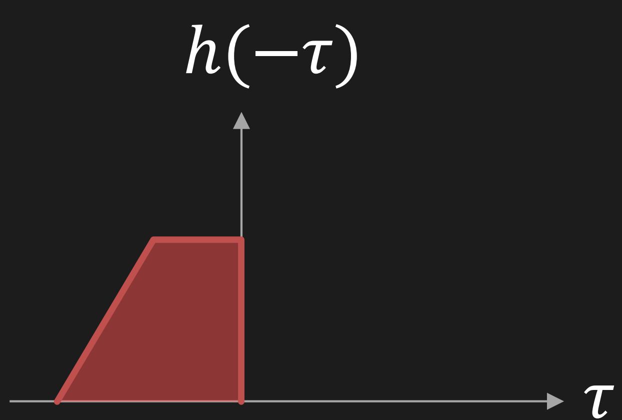


Convolution (Important Concept)

Convolution of two functions $f(x)$ and $h(x)$

$$\boxed{g(x) = f(x) * h(x) = \int_{-\infty}^{\infty} f(\tau)h(x-\tau)\,d\tau}$$

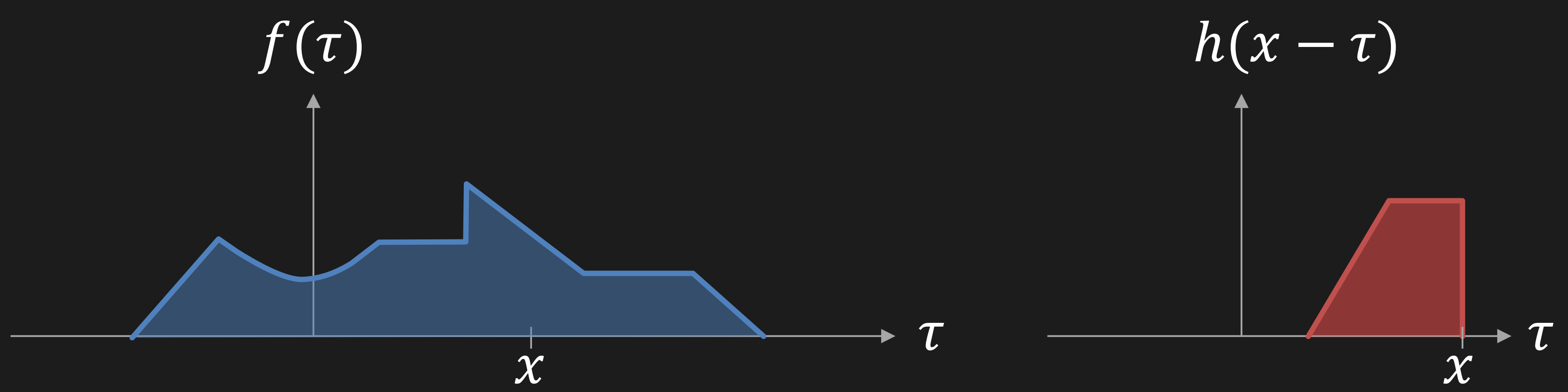




Convolution (Important Concept)

Convolution of two functions $f(x)$ and $h(x)$

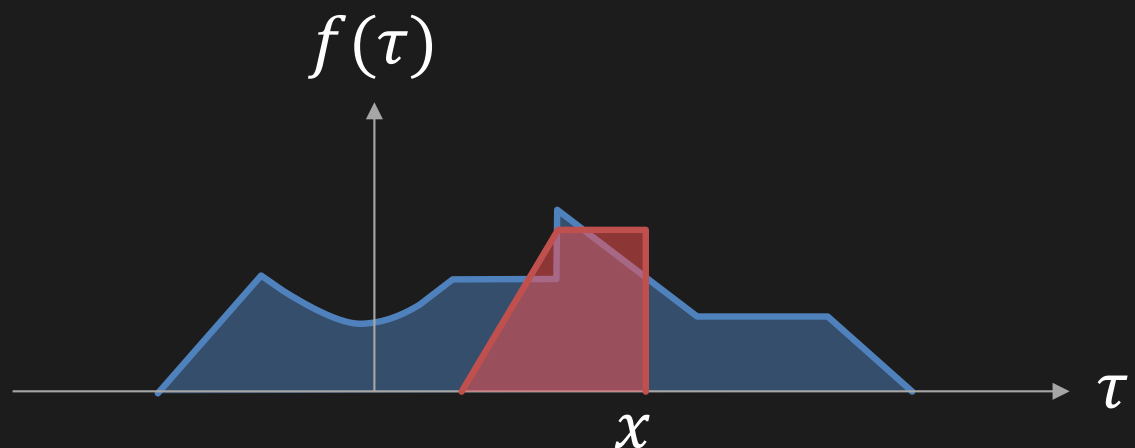
$$\boxed{g(x) = f(x) * h(x) = \int_{-\infty}^{\infty} f(\tau)h(x-\tau)\,d\tau}$$

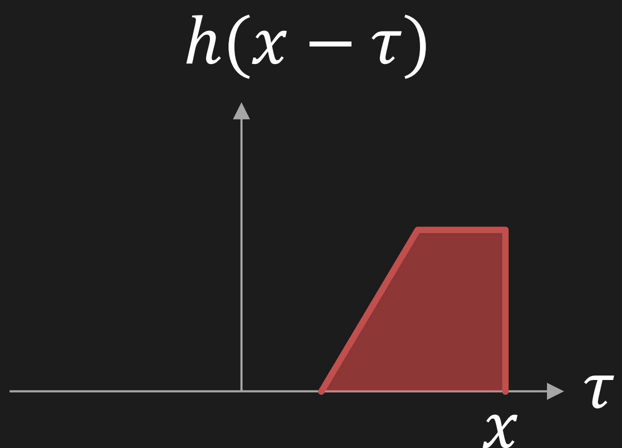


Convolution (Important Concept)

Convolution of two functions $f(x)$ and $h(x)$

$$\boxed{g(x) = f(x) * h(x) = \int_{-\infty}^{\infty} f(\tau)h(x-\tau)\, d\tau}$$

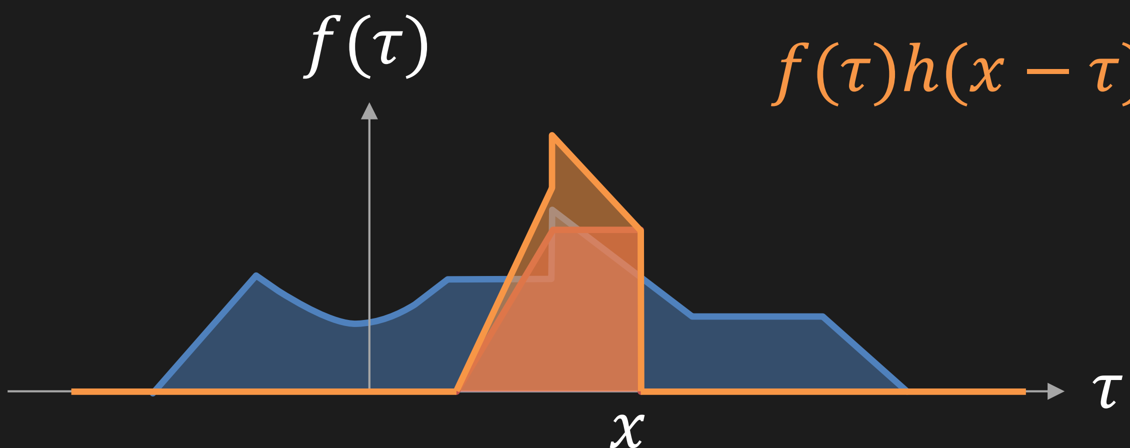


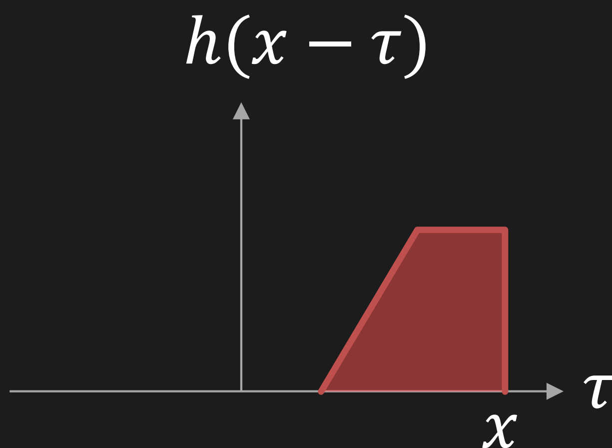


Convolution (Important Concept)

Convolution of two functions $f(x)$ and $h(x)$

$$\boxed{g(x) = f(x) * h(x) = \int_{-\infty}^{\infty} f(\tau)h(x-\tau)\, d\tau}$$

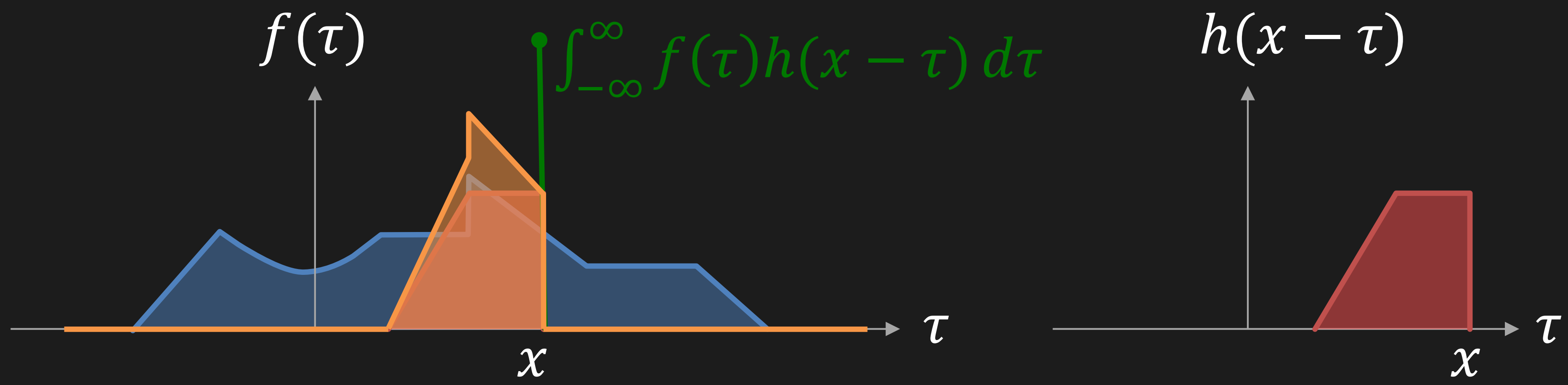




Convolution (Important Concept)

Convolution of two functions $f(x)$ and $h(x)$

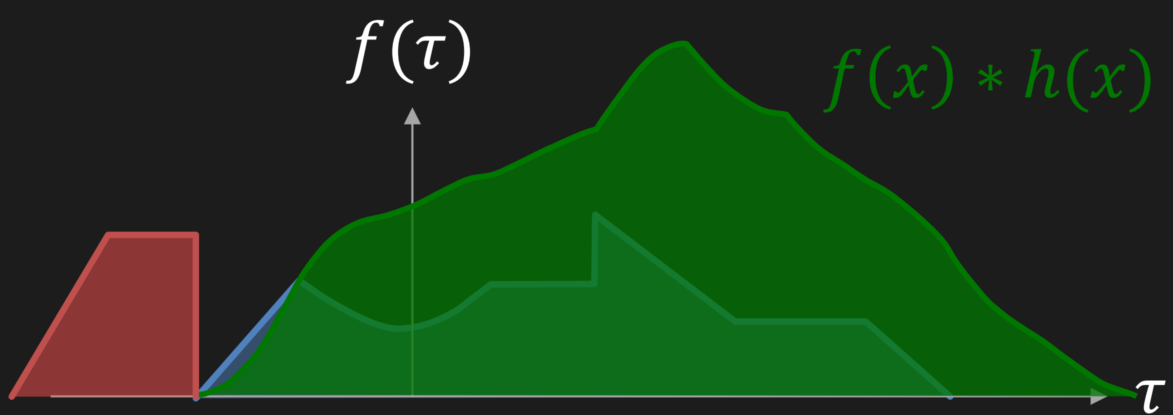
$$\boxed{g(x) = f(x) * h(x) = \int_{-\infty}^{\infty} f(\tau)h(x-\tau)\,d\tau}$$

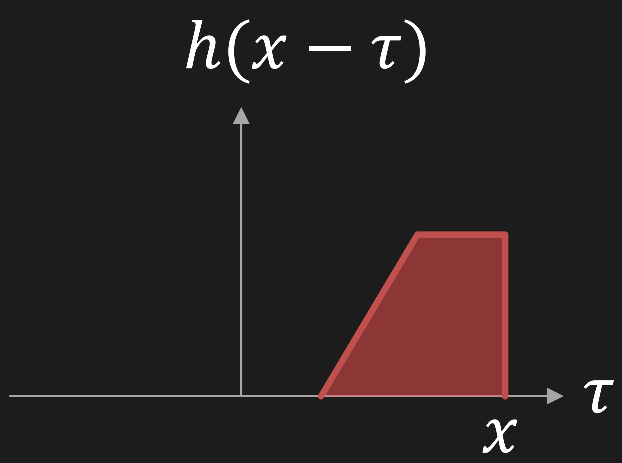


Convolution (Important Concept)

Convolution of two functions $f(x)$ and $h(x)$

$$\boxed{g(x) = f(x) * h(x) = \int_{-\infty}^{\infty} f(\tau)h(x-\tau)\,d\tau}$$

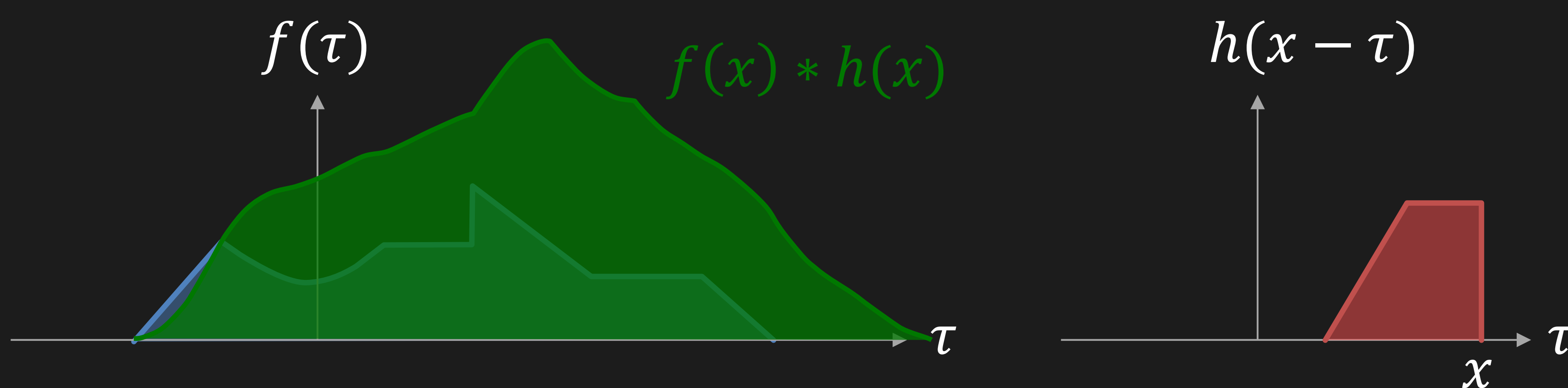




Convolution (Important Concept)

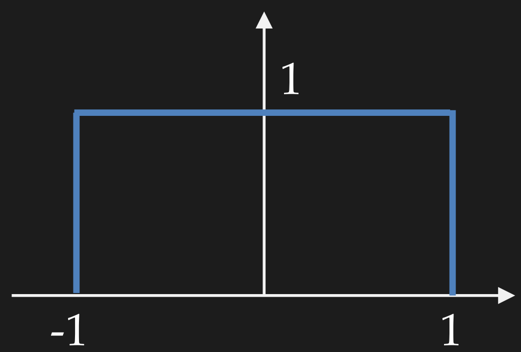
Convolution of two functions $f(x)$ and $h(x)$

$$\boxed{g(x) = f(x) * h(x) = \int_{-\infty}^{\infty} f(\tau)h(x-\tau)\,d\tau}$$

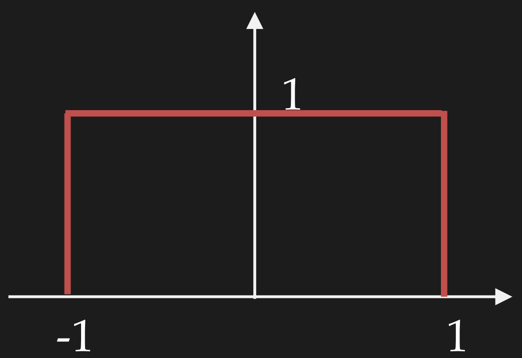


LSIS implies Convolution and Convolution implies LSIS

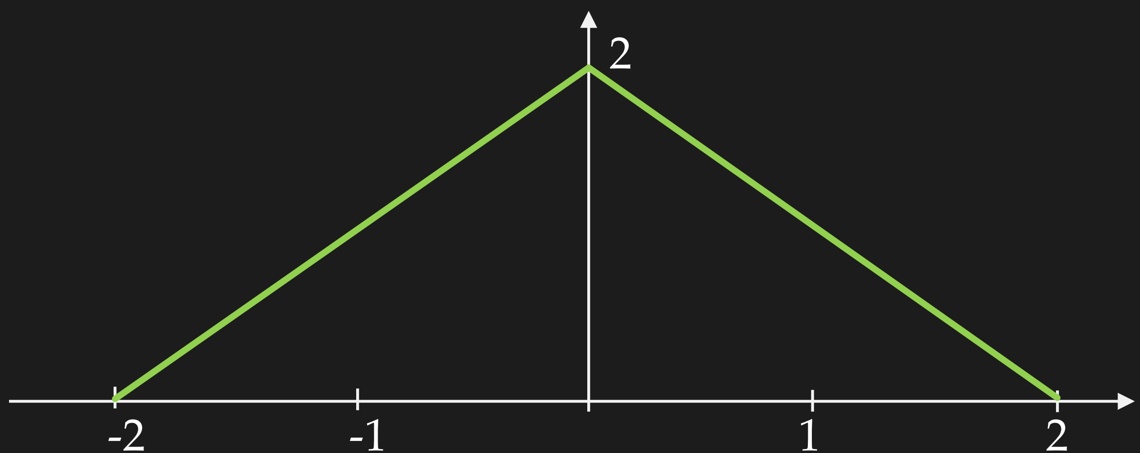
Convolution: Example



$f(x)$

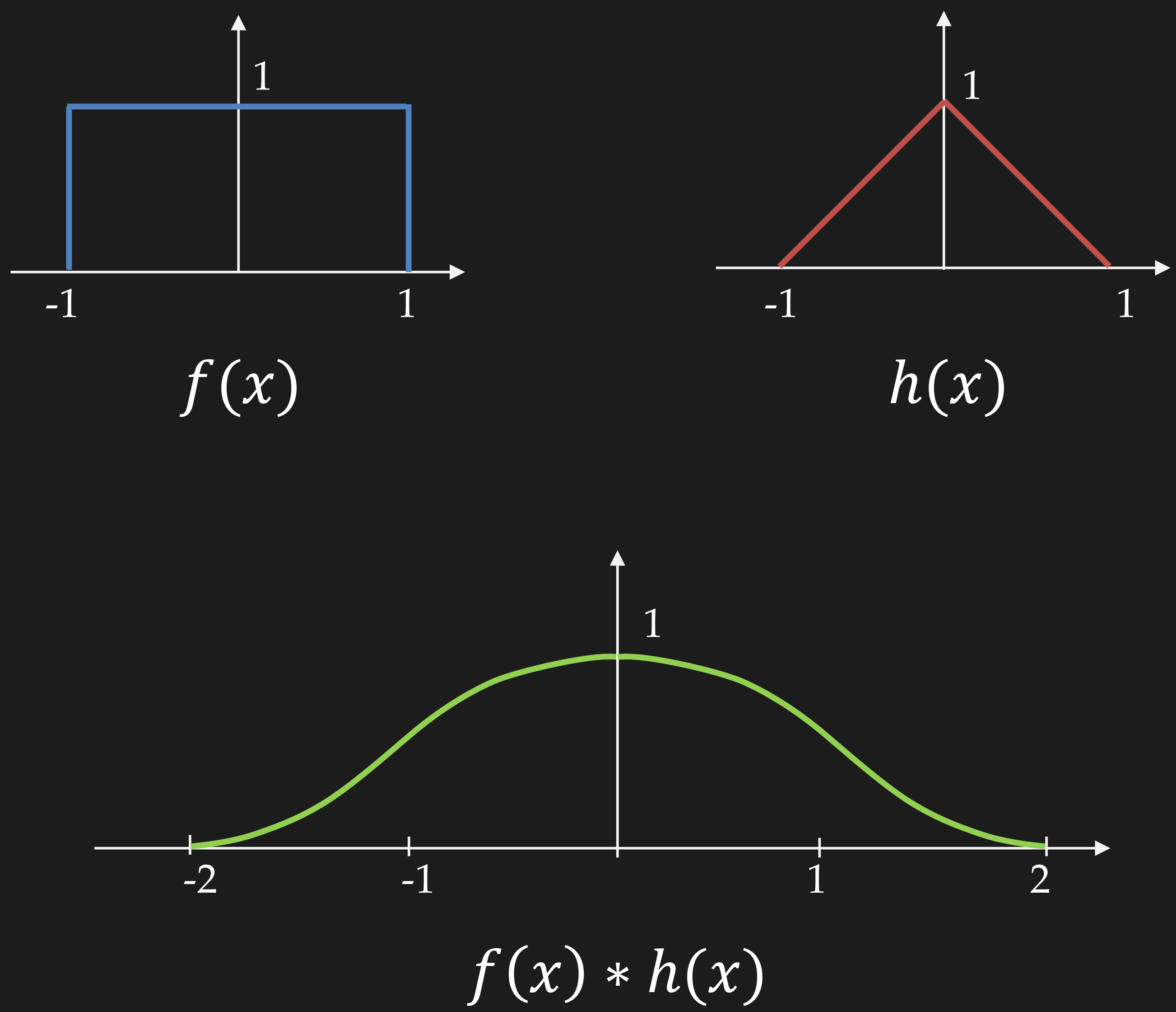


$h(x)$



$f(x) * h(x)$

Convolution: Example



Can we find h?

$$f \longrightarrow \boxed{h} \longrightarrow g$$

$$g(x) = \int_{-\infty}^{\infty} f(\tau)h(x-\tau)\,d\tau$$

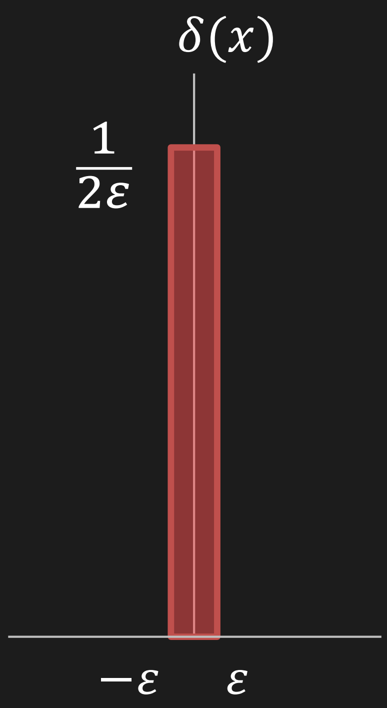
What input f will produce output $g = h$?

$$h(x) = \int_{-\infty}^{\infty} ?(\tau)h(x-\tau)\,d\tau$$

Unit Impulse Function

$$\delta(x) = \begin{cases} 1/2\varepsilon\,, & |x| \le \varepsilon \\ 0, & |x| > \varepsilon \end{cases}$$

$$\varepsilon \to 0$$



$$\int_{-\infty}^{\infty} \delta(\tau)\, d\tau = \frac{1}{2\varepsilon} . 2\varepsilon = 1$$

$$\boxed{\int_{-\infty}^{\infty} \delta(\tau) b(x-\tau)\, d\tau = b(x)}$$

Shifting Property

Impulse Response

$$f \longrightarrow \boxed{h} \longrightarrow g$$

$$\delta \longrightarrow \boxed{h} \longrightarrow h$$

Unit Impulse

Impulse Response

$$g(x) = f(x) * h(x)$$

$$h(x) = \delta(x) * h(x)$$

$$g(x) = \int_{-\infty}^{\infty} f(\tau) h(x - \tau)\, d\tau$$

$$h(x) = \int_{-\infty}^{\infty} \delta(\tau) h(x - \tau)\, d\tau$$

Properties of Convolution

Commutative $$a * b = b * a$$

Associative $$(a * b) * c = a * (b * c)$$

Cascaded System

$$f \rightarrow \boxed{h_1} \rightarrow \boxed{h_2} \rightarrow g$$

$$\equiv \quad f \rightarrow \boxed{h_1 * h_2} \rightarrow g$$

$$\equiv \quad f \rightarrow \boxed{h_2 * h_1} \rightarrow g$$

Summary

Convolution: A linear operator that allows us to transform one image to another

Essential concepts in this lecture:

The definition of convolution and the basic properties of convolution as an operator.

Image Processing I

Computer Vision: CS 600.461/661

Image Processing I

Transform image to new one that is easier to manipulate.

Topics:

(1) 2D Convolution

(2) Linear Filtering

(3) Non-Linear Filtering

(4) Correlation

Computer Vision: Algorithms and Applications (Chapter 3.2)
Szelinski, 2011 (available online)

2D Convolution

LSIS:

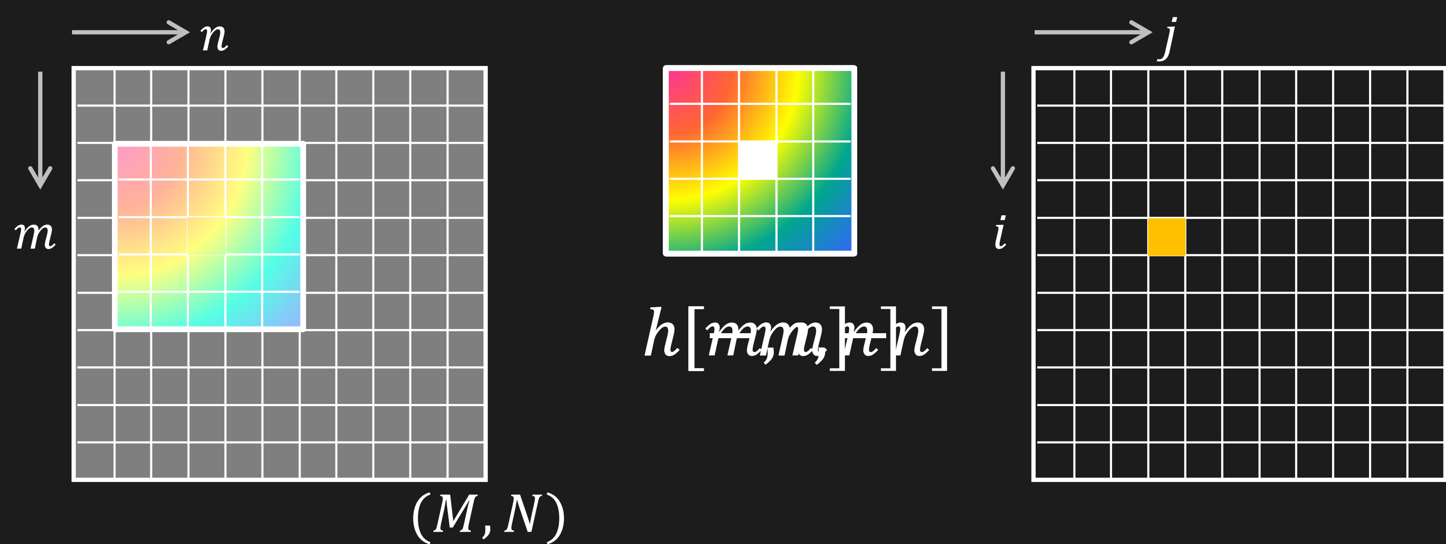
$$f(x,y) \longrightarrow \boxed{h(x,y)} \longrightarrow g(x,y)$$

Convolution:

$$g(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\tau,\mu)h(x-\tau,y-\mu)\,d\tau d\mu$$

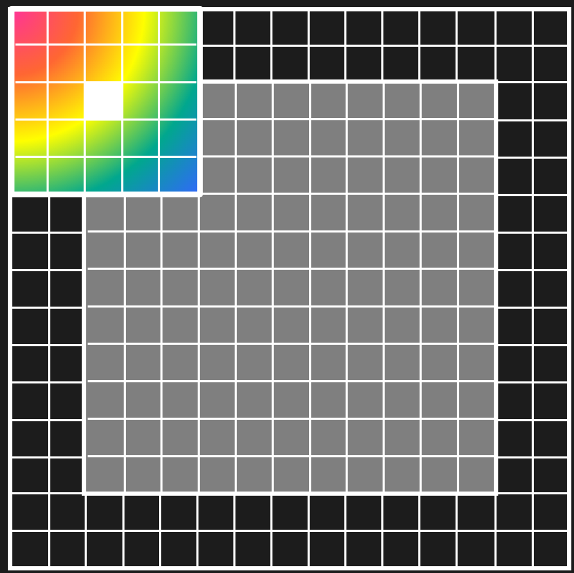
Convolution with Discrete Images

$$f[m,n] \longrightarrow \boxed{h[m,n]} \longrightarrow g[i,j]$$



$$g[i,j] = \sum_{m=1}^{M} \sum_{n=1}^{N} f[m,n]h[i-m,j-n]$$

Border Problem

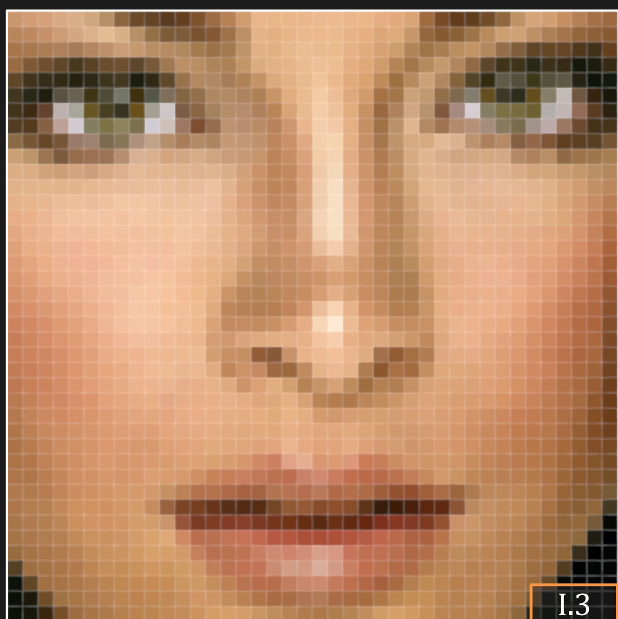


Solution:

- Ignore Border
- Pad with Constant Value
- Pad with Reflection

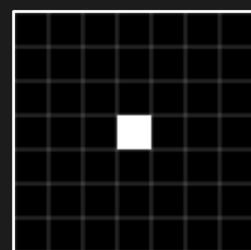
Example: Impulse Filter

Input



$f(x,y)$

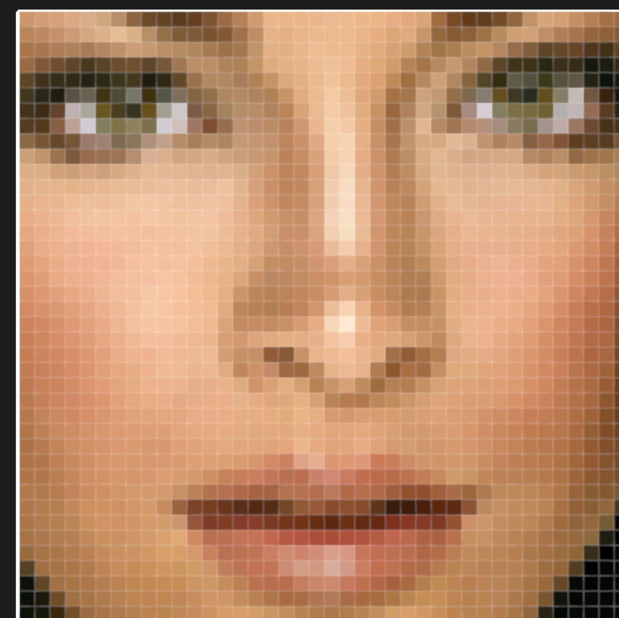
$*$



$\delta(x,y)$

$=$

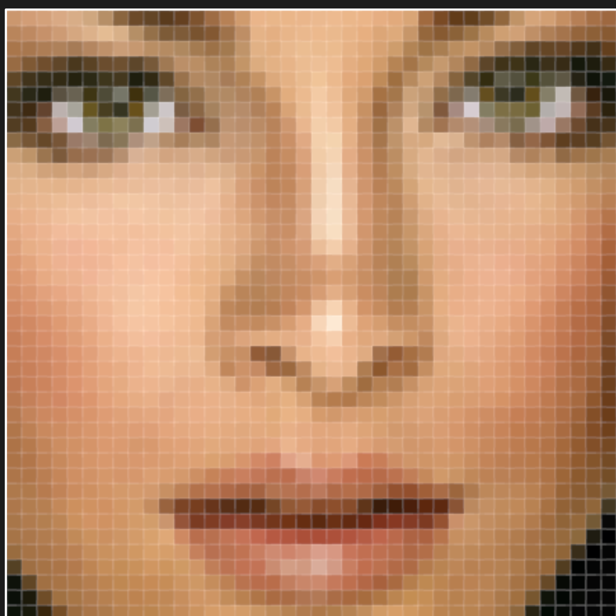
Output



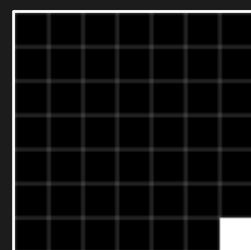
$f(x,y)$

Example: Image Shift

Input

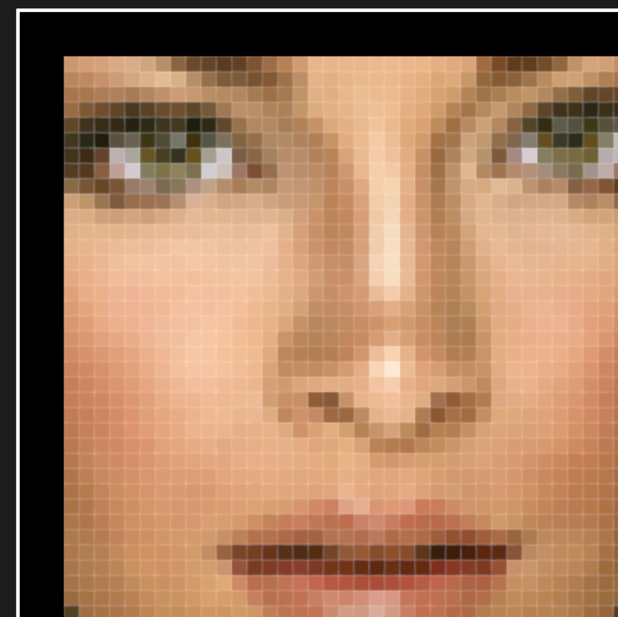


$*$



$=$

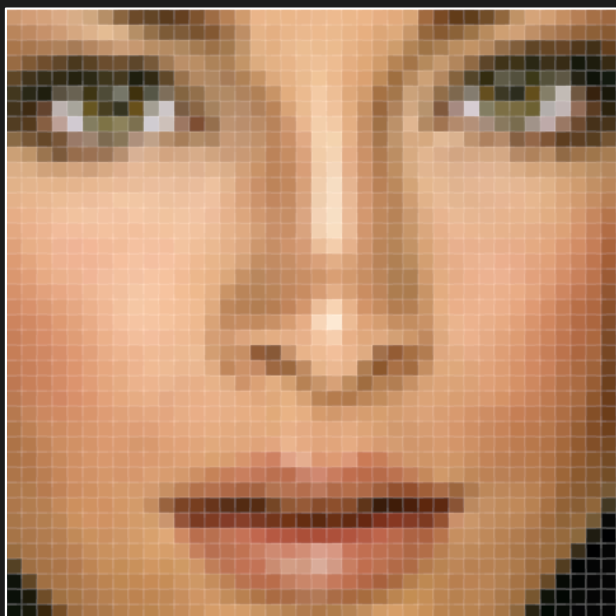
Output



$f(x, y)$ $\qquad$ $\delta(x-u, y-v)$ $\qquad$ $f(x-u, y-v)$

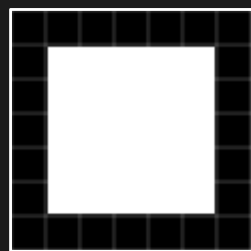
Example: Averaging

Input



$f(x,y)$

$*$



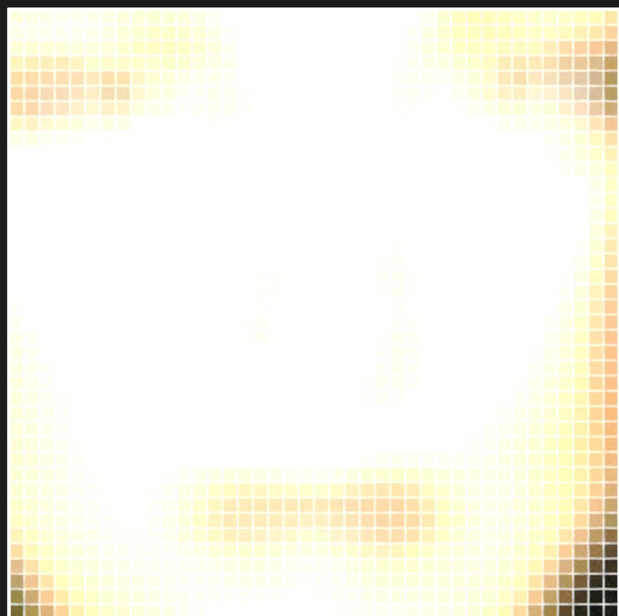
"Box Filter"

5 x 5

$a(x,y)$

$=$

Output

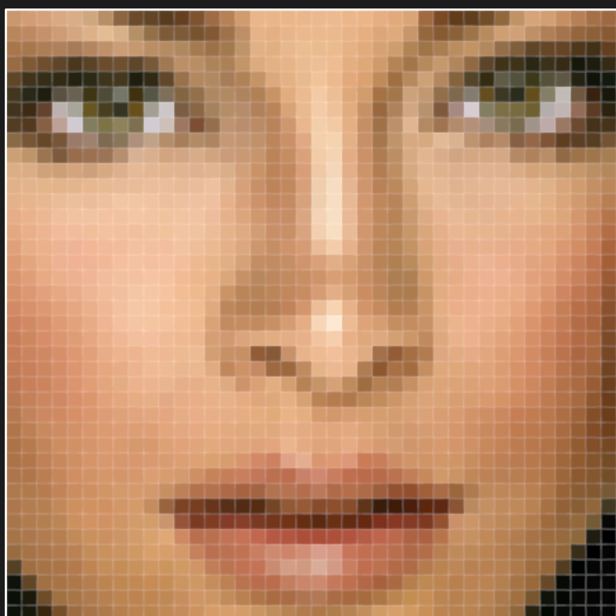


$g(x,y)$

Result Image is Saturated. Why?

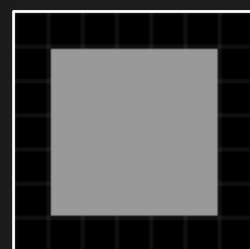
Example: Averaging

Input



$f(x, y)$

$*$



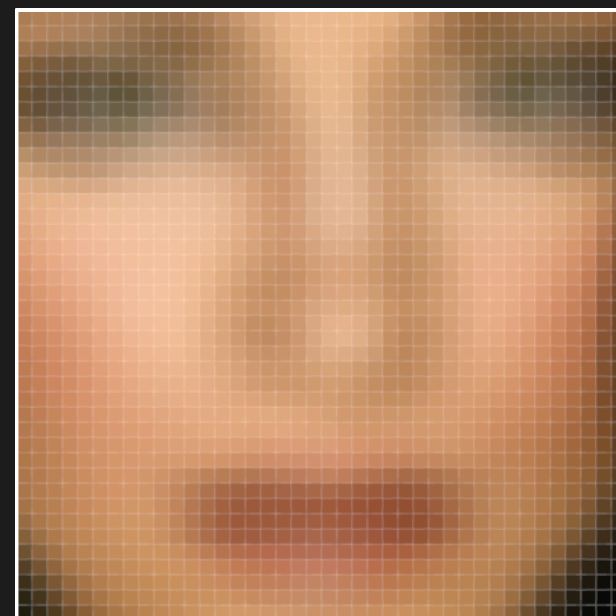
“Box Filter”

5 x 5

$a(x, y)$

$=$

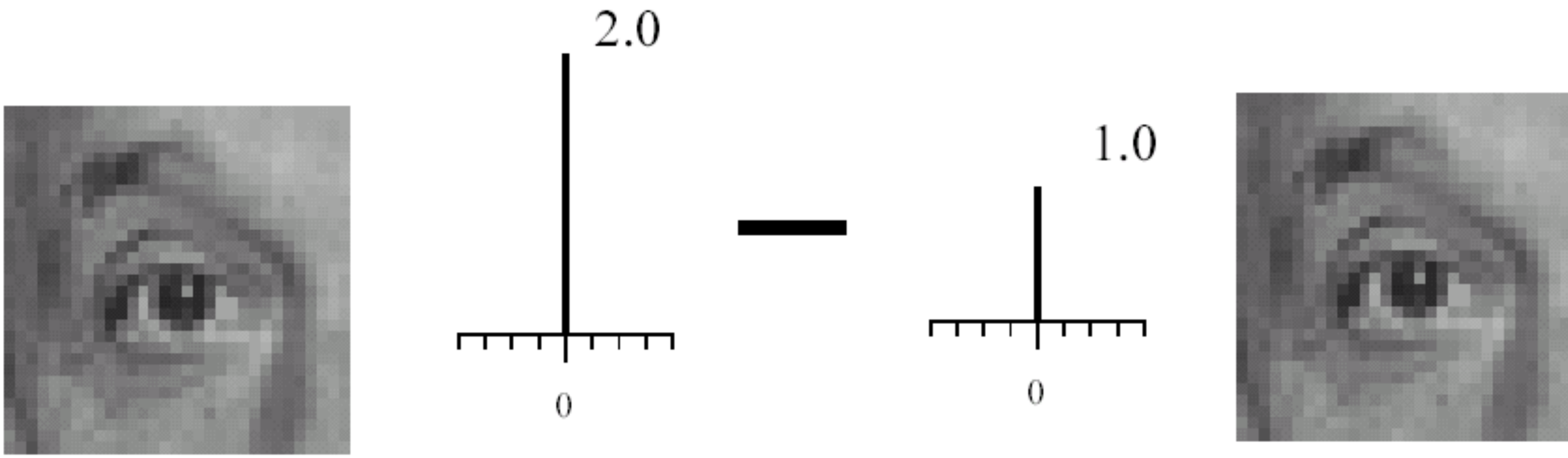
Output



$g(x, y)$

Sum of all the Filter (Kernel) Weights should be 1.

Linear filtering (no change)

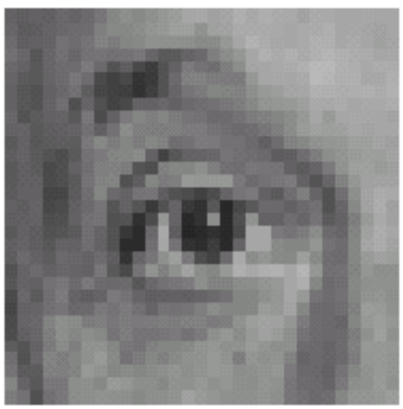


original

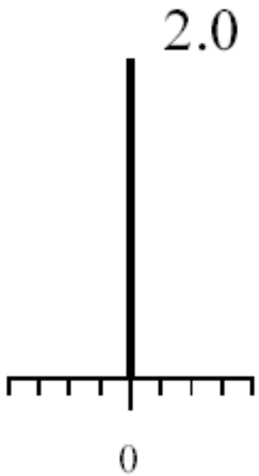
Filtered
(no change)

(Swiped from Bill Freeman)

Linear filtering



original

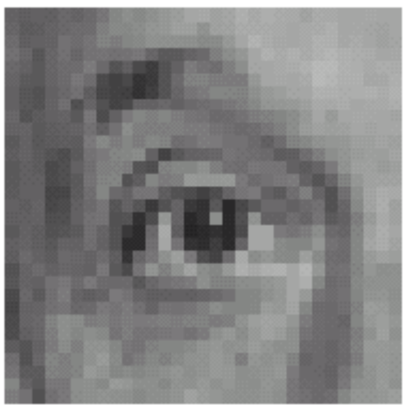


−

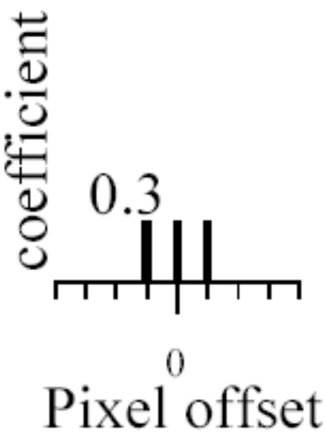
?

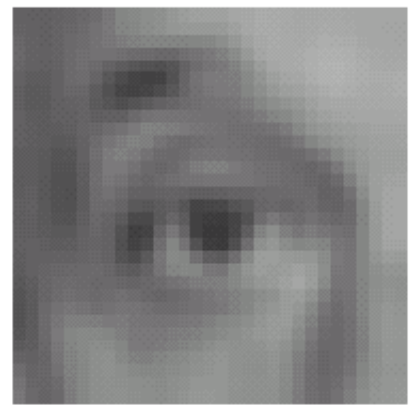
(Swiped from Bill Freeman)

(remember blurring)



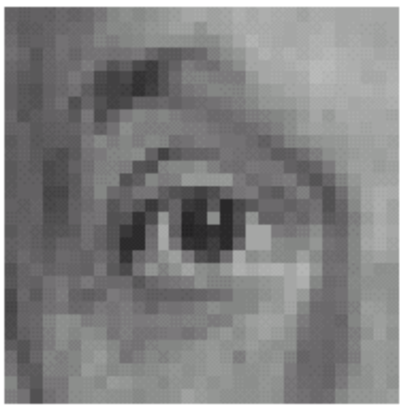
original



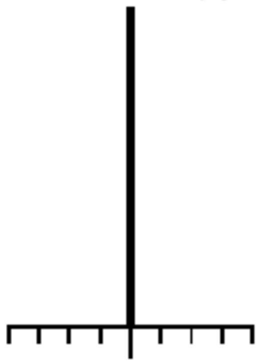


Blurred (filter applied in both dimensions).

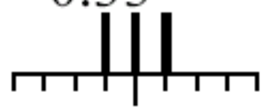
Sharpening

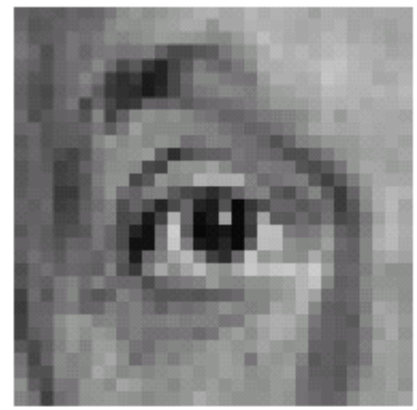


original



−

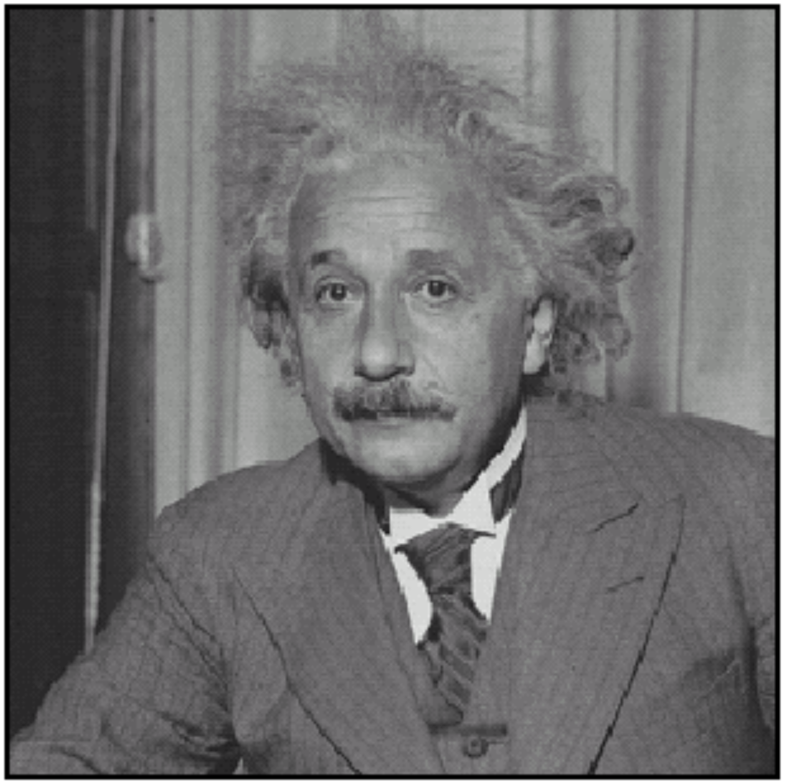




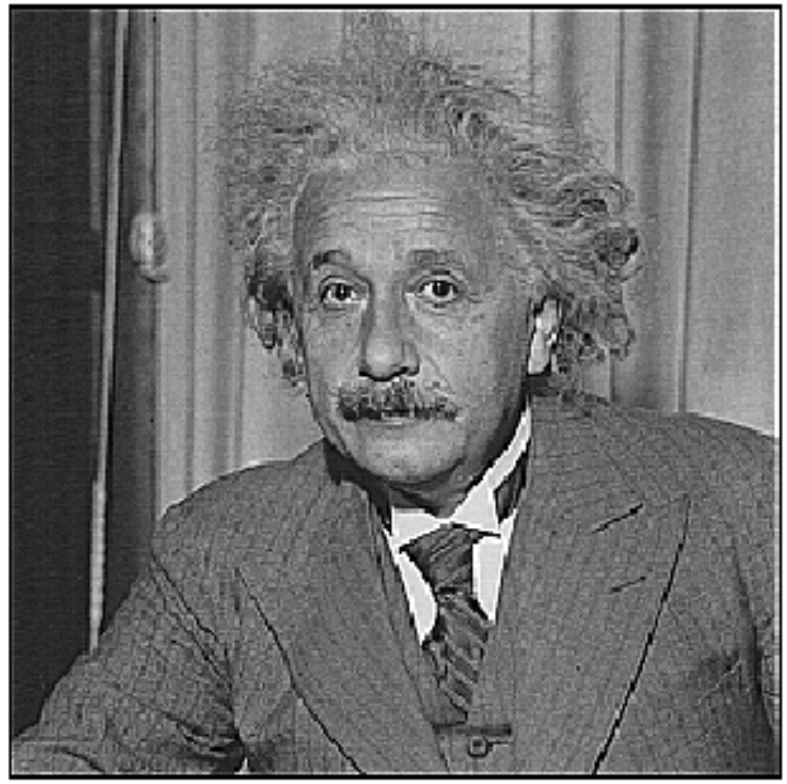
Sharpened original

borrowed from D. Kriegman

Sharpening



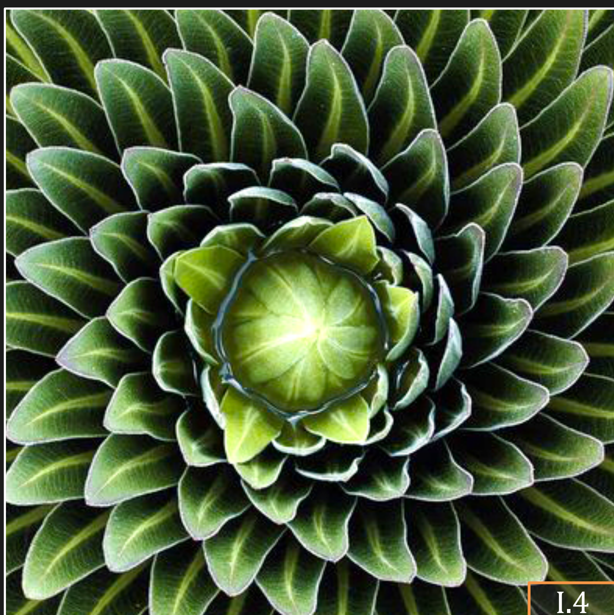
before



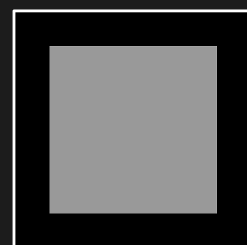
after

Smoothing With Box Filter

Input



$*$



"Box Filter"
21 x 21

$=$

Output

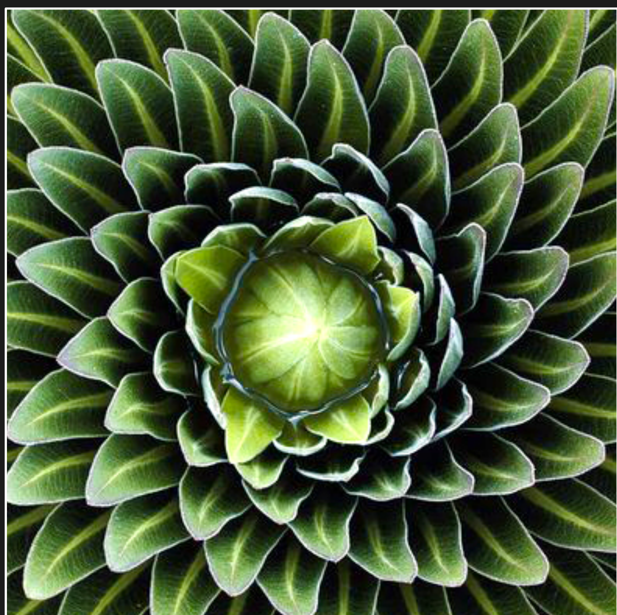
$f(x, y)$ $a(x, y)$ $g(x, y)$

Image smoothed with a box filter does not look "natural."
Has blocky artifacts.

Smoothing With "Fuzzy" Filter

Input

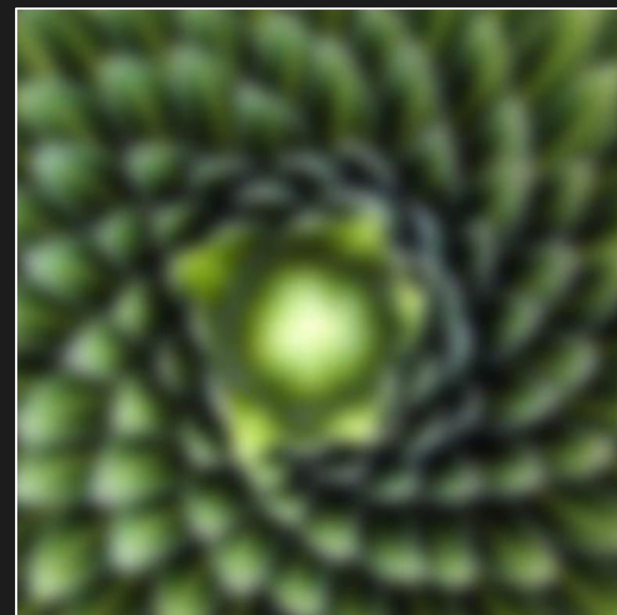
Output



$*$

"Fuzzy Filter"
21 x 21

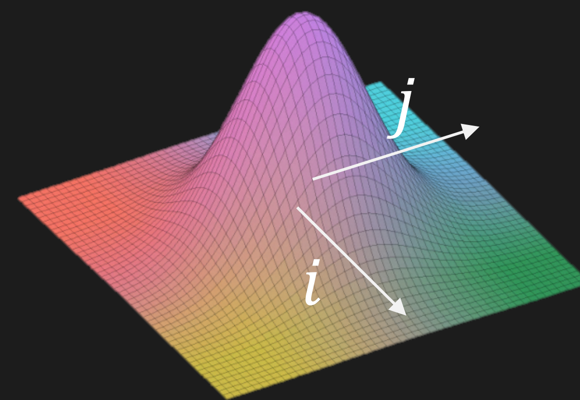
$=$



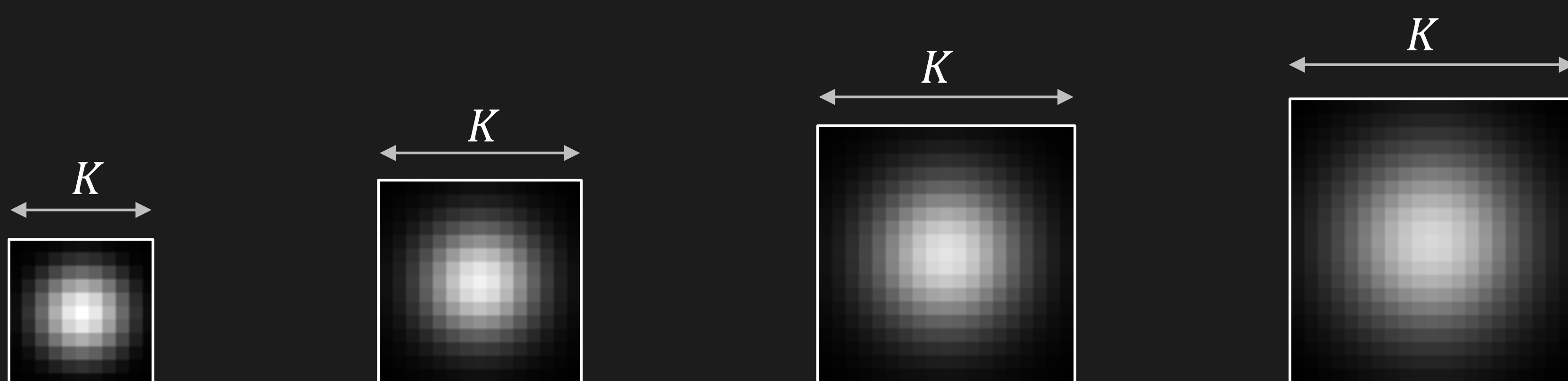
$f(x, y)$ $b(x, y)$ $g(x, y)$

Gaussian Kernel: A Fuzzy Filter

$$n_\sigma[i,j] = \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2}\left(\frac{i^2+j^2}{\sigma^2}\right)}$$



σ^2: Variance



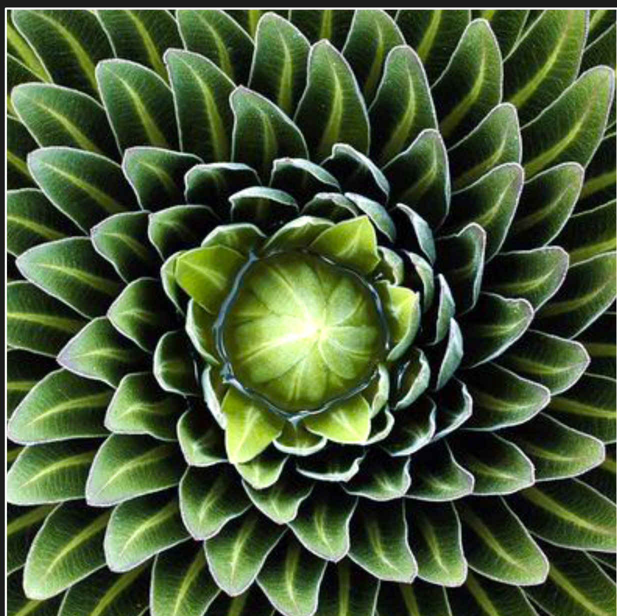
$\sigma = 2$ $\sigma = 3$ $\sigma = 4$ $\sigma = 5$

Rule of Thumb: Set Kernel Size $K \approx 2\pi\sigma$

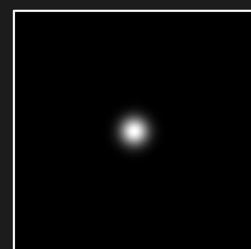
Gaussian Smoothing

Input

Output



$*$



$=$

$\sigma = 4$

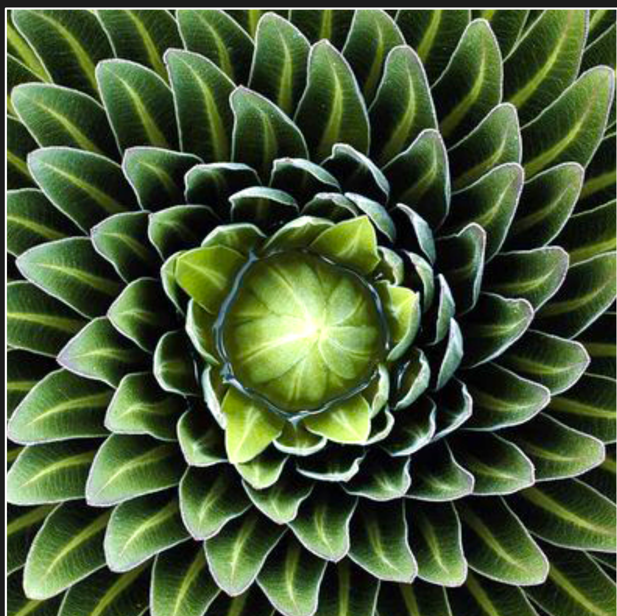
$f(x,y)$ $n_4(x,y)$ $g(x,y)$

Larger the Kernel (or σ), More the Blurring

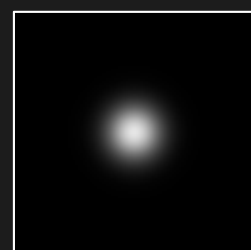
Gaussian Smoothing

Input

Output

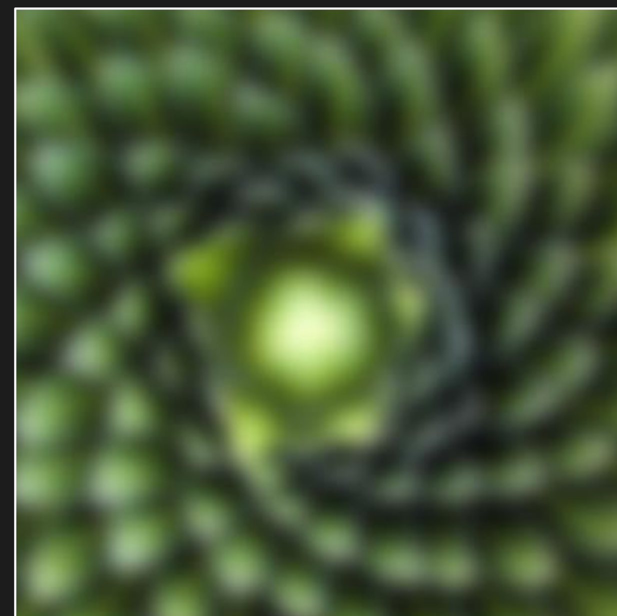


$*$



$\sigma = 8$

$=$



$f(x,y)$

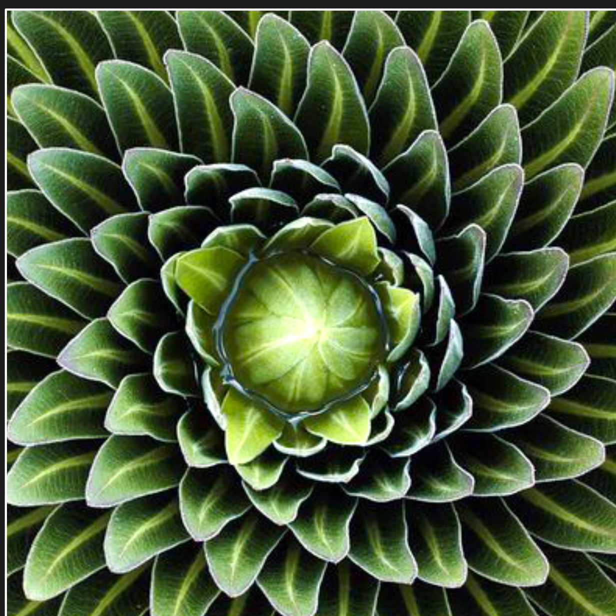
$n_8(x,y)$

$g(x,y)$

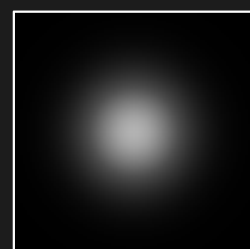
Larger the Kernel (or σ), More the Blurring

Gaussian Smoothing

Input | Output

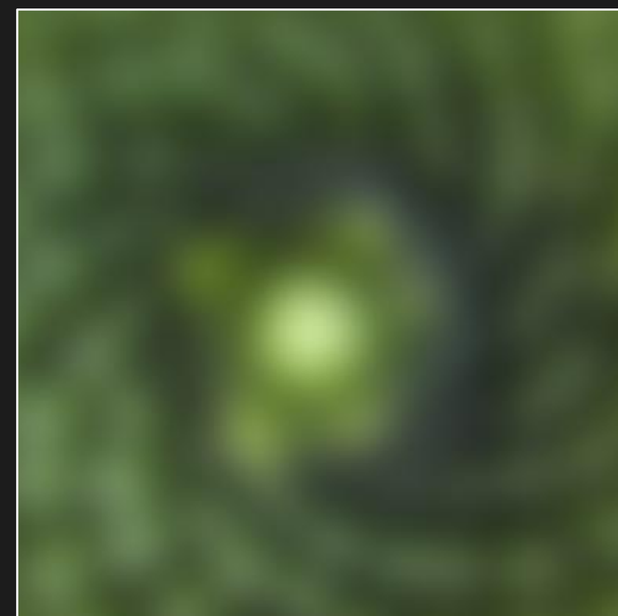


$*$



$\sigma = 16$

$=$



$f(x, y)$ $\quad n_{16}(x, y)$ $\quad g(x, y)$

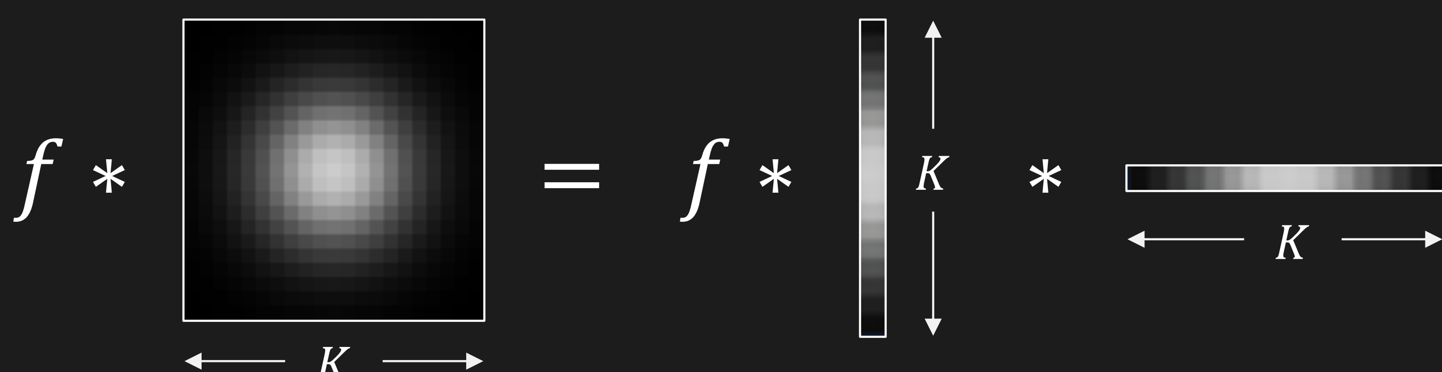
Larger the Kernel (or σ), More the Blurring

Gaussian Smoothing is Separable

$$g[i,j] = \frac{1}{2\pi\sigma^2} \sum_{m=-K/2}^{K/2} \sum_{n=-K/2}^{K/2} e^{-\frac{1}{2}\left(\frac{m^2+n^2}{\sigma^2}\right)} f[i-m, j-n]$$

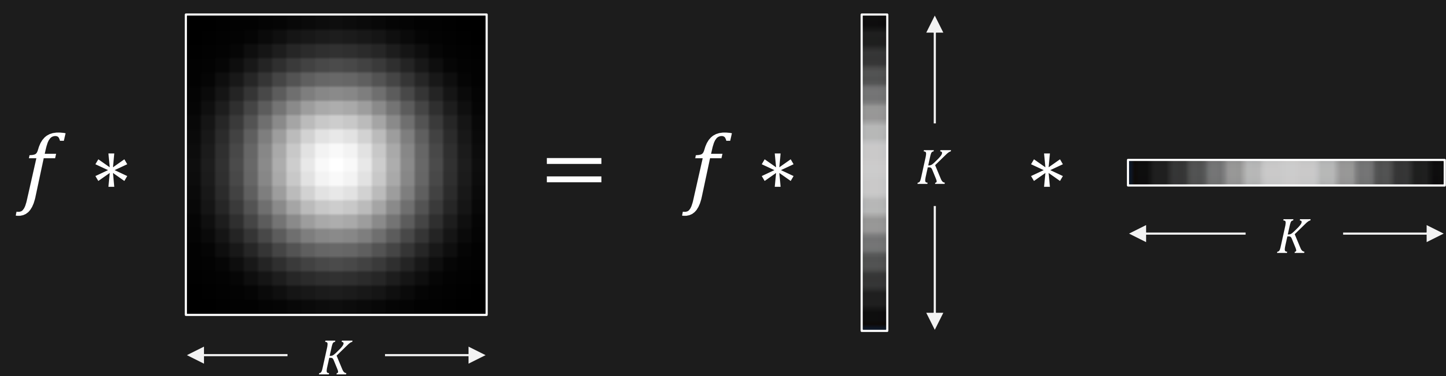
$$g[i,j] = \frac{1}{2\pi\sigma^2} \sum_{m=-K/2}^{K/2} e^{-\frac{1}{2}\left(\frac{m^2}{\sigma^2}\right)} \cdot \sum_{n=-K/2}^{K/2} e^{-\frac{1}{2}\left(\frac{n^2}{\sigma^2}\right)} f[i-m, j-n]$$

Using One 2D Gaussian Filter ≡ Using Two 1D Gaussian Filters



Gaussian Smoothing is Separable

Using One 2D Gaussian Filter ≡ Using Two 1D Gaussian Filters



Which one is faster? Why?

K^2 Multiplications	$2K$ Multiplications
$K^2 - 1$ Additions	$2(K-1)$ Additions

Summary

Convolution: A linear operator that allows us to transform one image to another

Essential concepts in this lecture:

2D convolution in the discrete domain, linear filtering, Gaussian smoothing, and properties of the Gaussian filter.