Image Convolution

Introducción a métodos de Machine Learning para ciencias

2023

What is a digital image?

$$f: \quad \mathbb{N}^d \to \mathbb{R}^B$$
$$m, n, \ldots \mapsto f(m, n, \ldots) = \{i_1, \ldots, i_B\}.$$

- For example, a grayscale image corresponds to $d = 2$ (the image has two dimensions) and $B = 1$ (there is only one value per location (m, n): the grayscale intensity).
- A common color image corresponds to $d = 2$ and $B = 3$ bands, typically red, green, and blue.
- An MRI image corresponds to $d = 3$ (the image is three-dimensional) and $B = 1$.

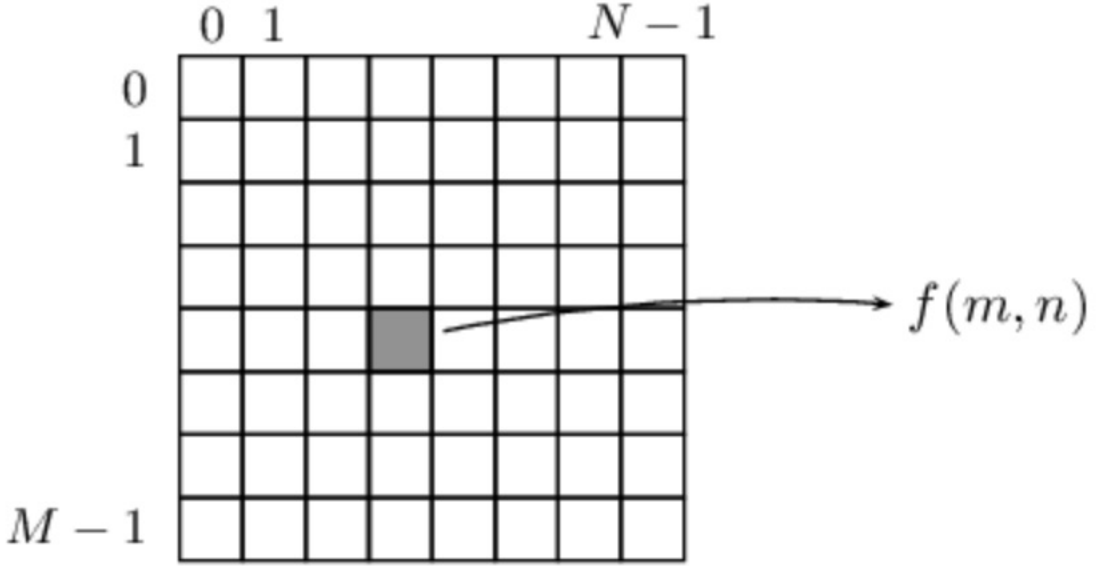
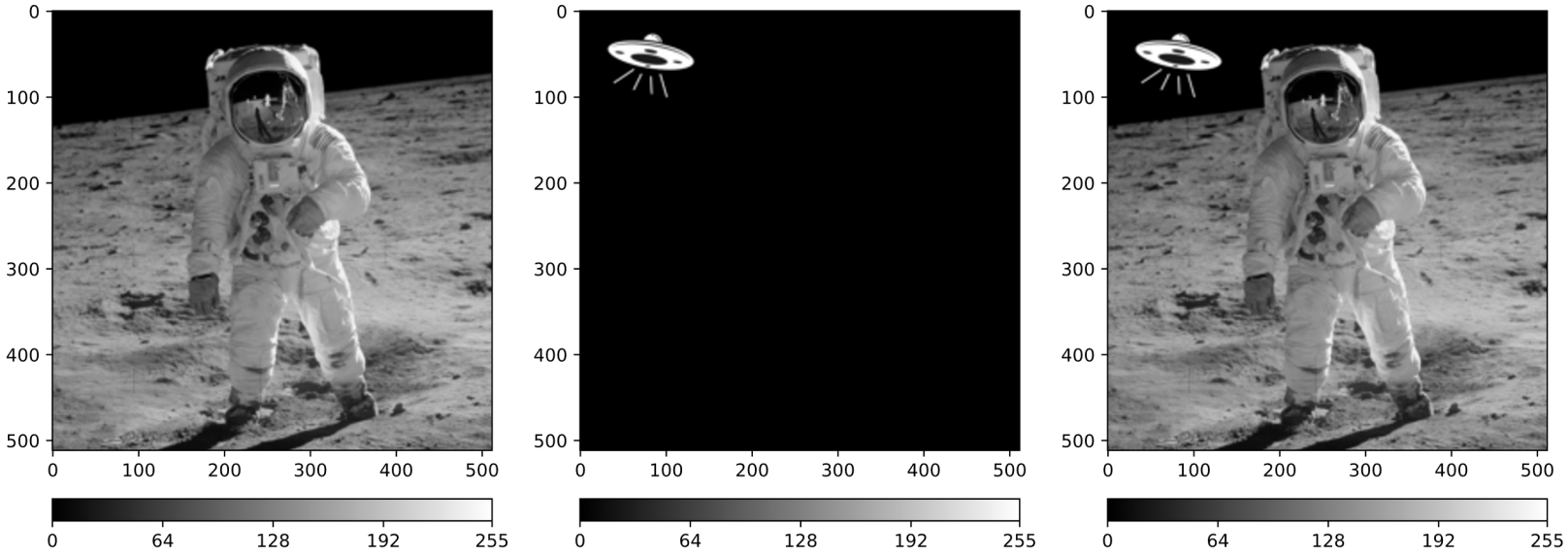


Fig. 1 Coordinate system generally used in image processing.

Arithmetic operations of images

Addition

$$\forall m, n, \quad h(m,n) = f(m,n) + g(m,n).$$



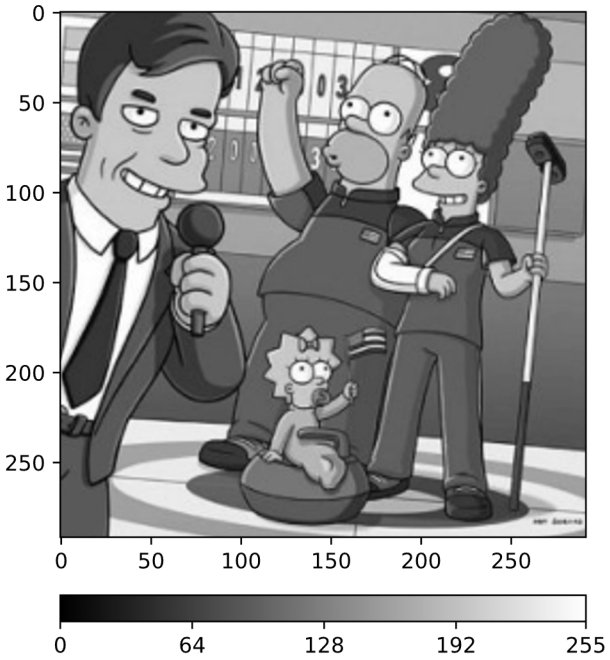
Arithmetic operations of images

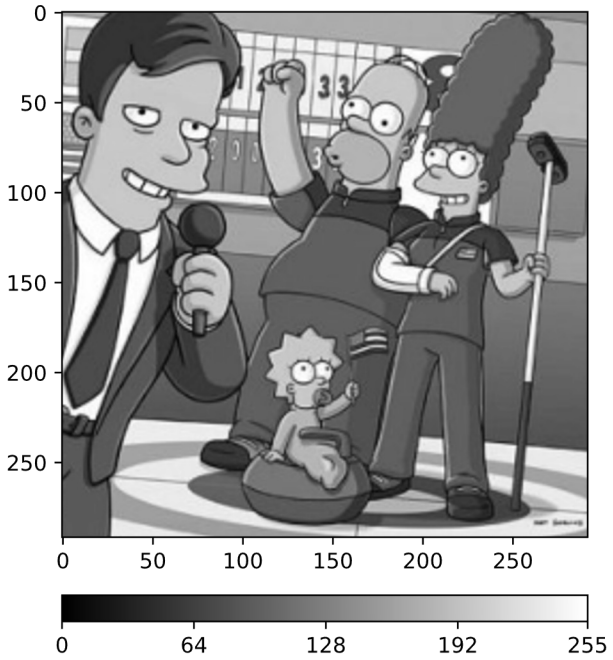
substraction

$$\forall m, n, \quad h(m,n) = f(m,n) - g(m,n)$$

or

$$\forall m, n, \quad h(m,n) = |f(m,n) - g(m,n)|$$





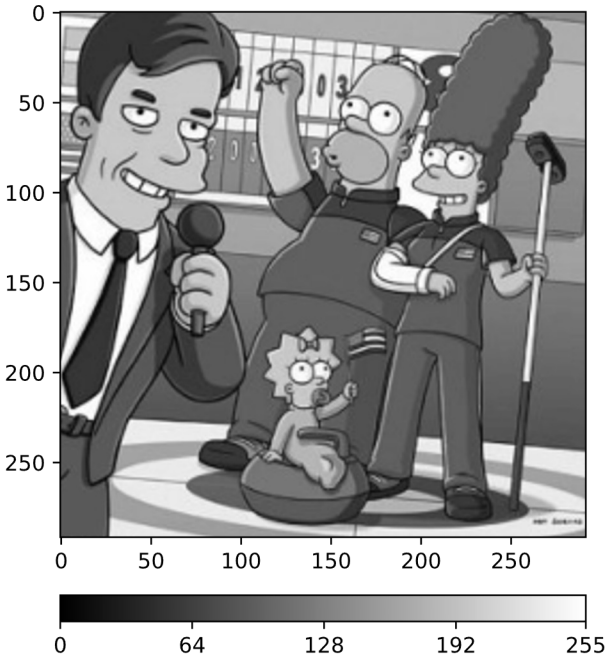
Arithmetic operations of images

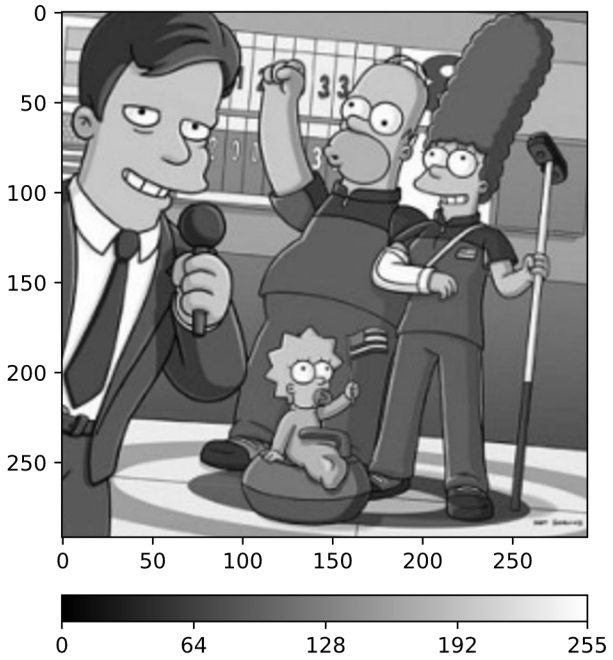
substraction

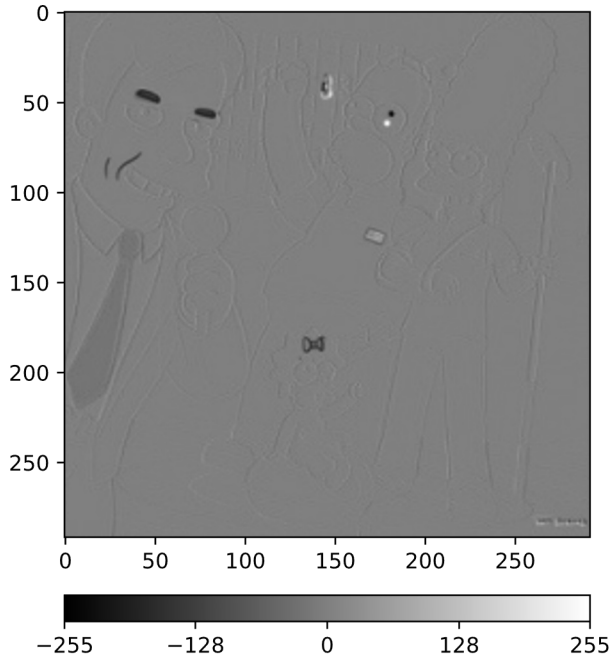
$$\forall m, n, \quad h(m,n) = f(m,n) - g(m,n)$$

or

$$\forall m, n, \quad h(m,n) = |f(m,n) - g(m,n)|$$







Arithmetic operations of images

Division

$$\forall m, n, \quad h(m,n) = \frac{f(m,n)}{g(m,n)}.$$

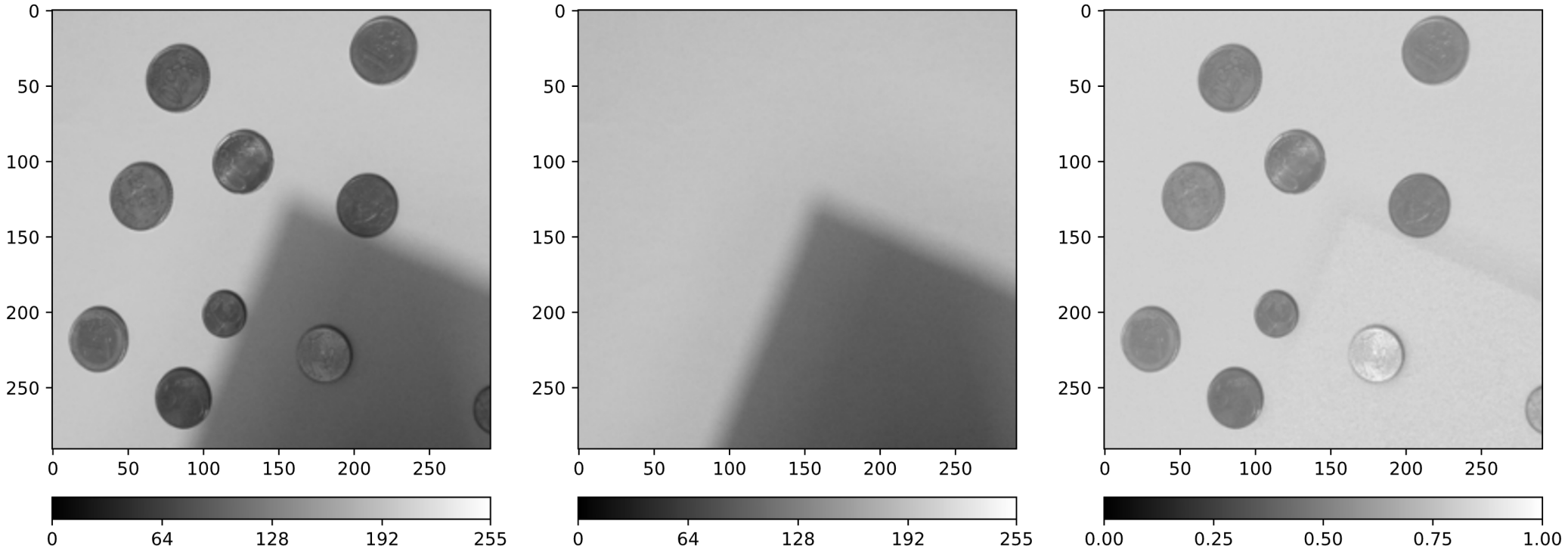
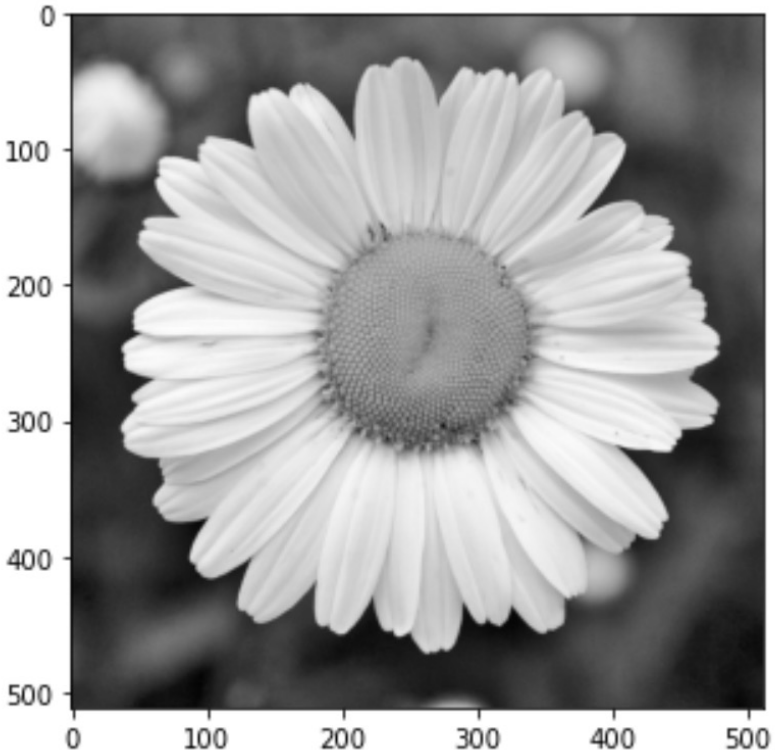
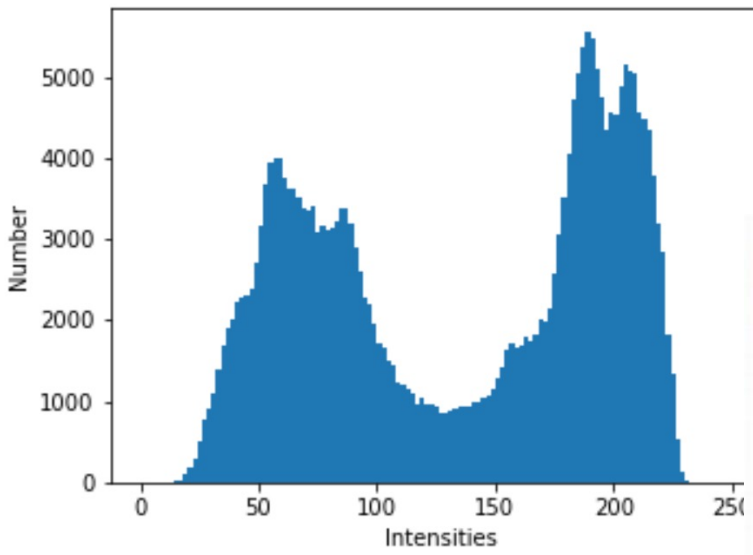
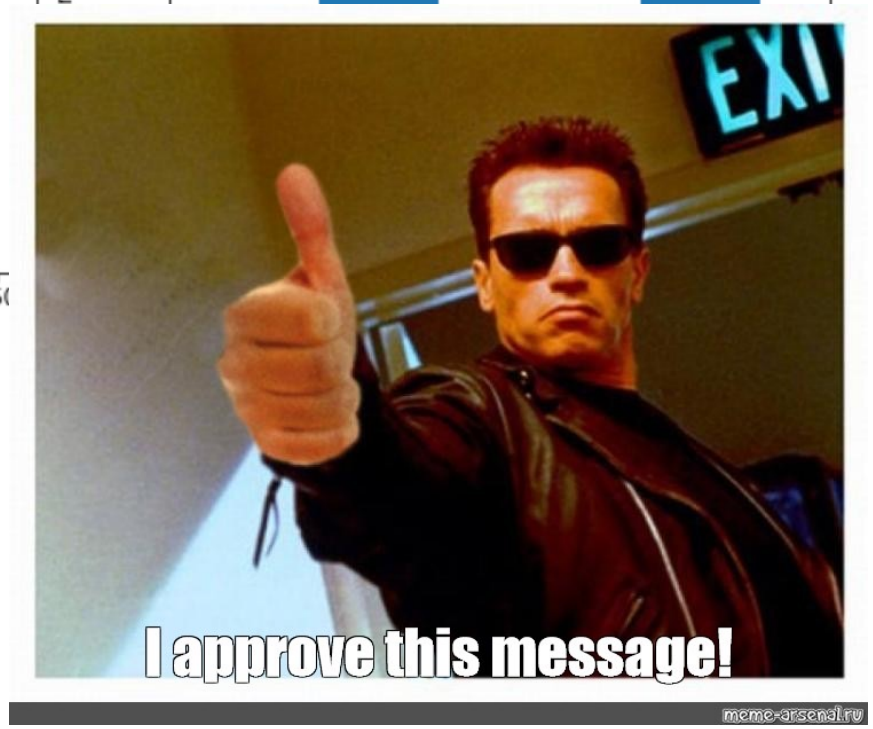


Image histogram







Histogram Transfomation

An histogram transformation consists in applying a mathematical function to the intensity distribution.

Generally, the transformations are useful to improve the visual quality of an image, but are rarely needed inside an automatic processing.

$$j = T(i)$$

we suppose the pixel intensities to lie in [0, 1]

Some commun transformations:

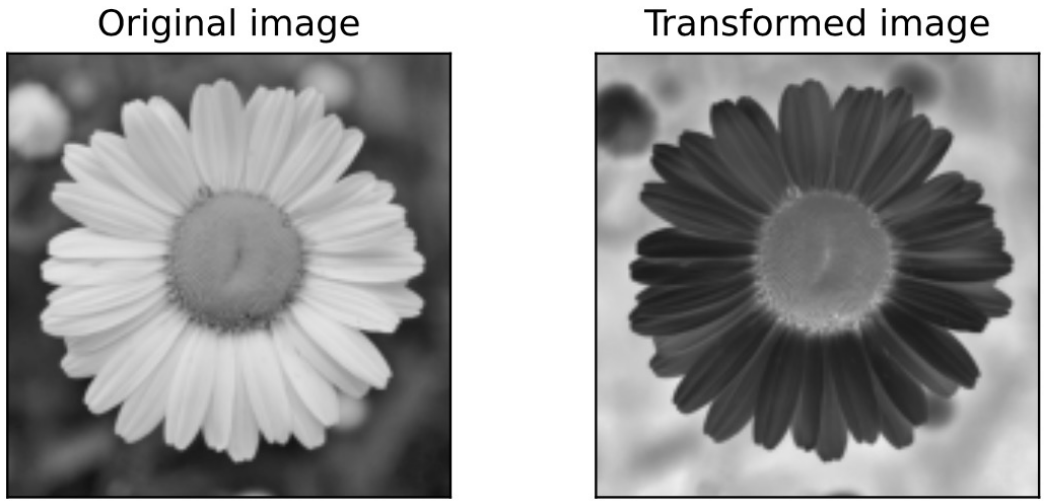
- Negative image
- Gamma correction
- Histogram spreading
- Histogram equalization

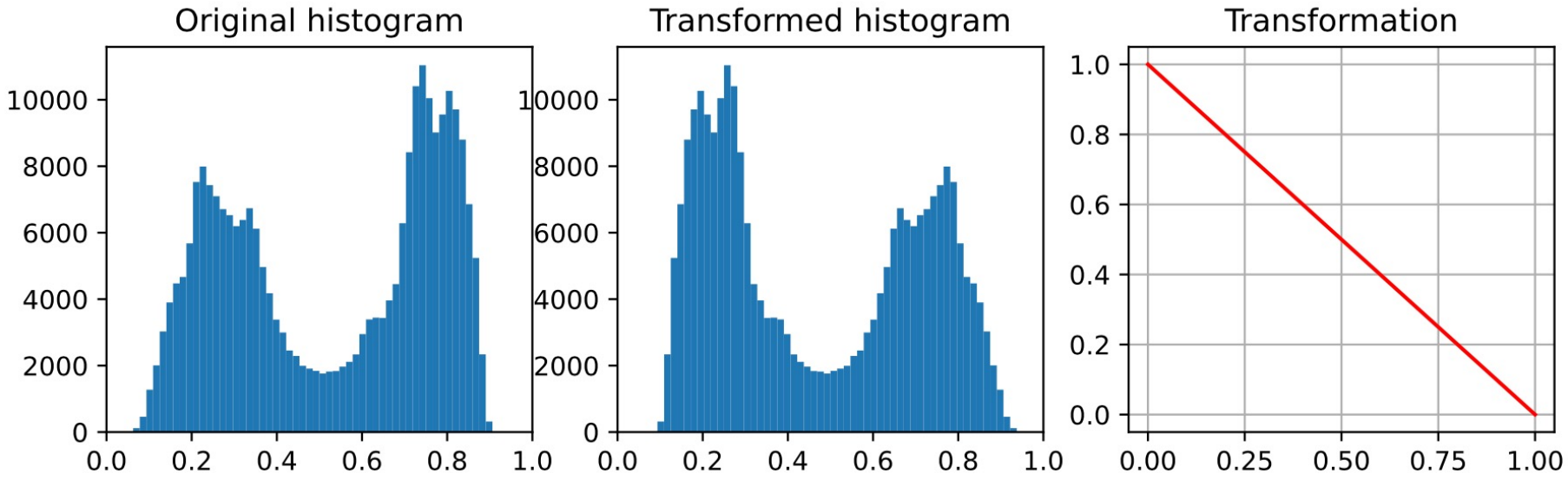
Negative image

we suppose the pixel intensities to lie in [0, 1]

$$T(i) = 1 - i$$

Negative image: the gray levels are reversed





Gamma correction

Original image

Transformed image

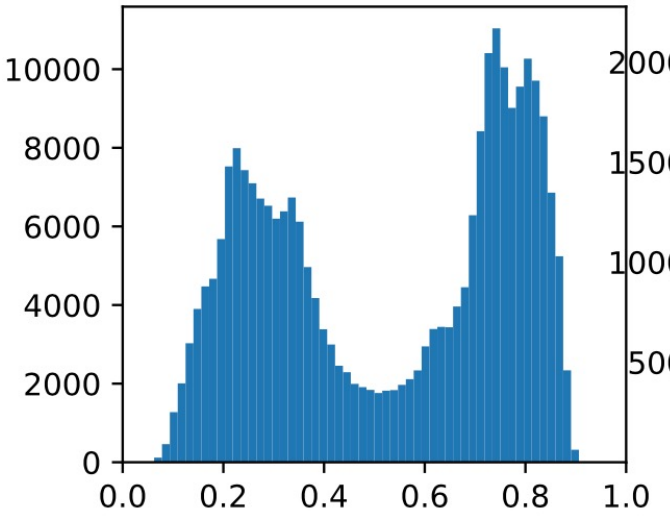
we suppose the pixel intensities to lie in [0, 1]

$$T(i) = i^{\gamma}$$

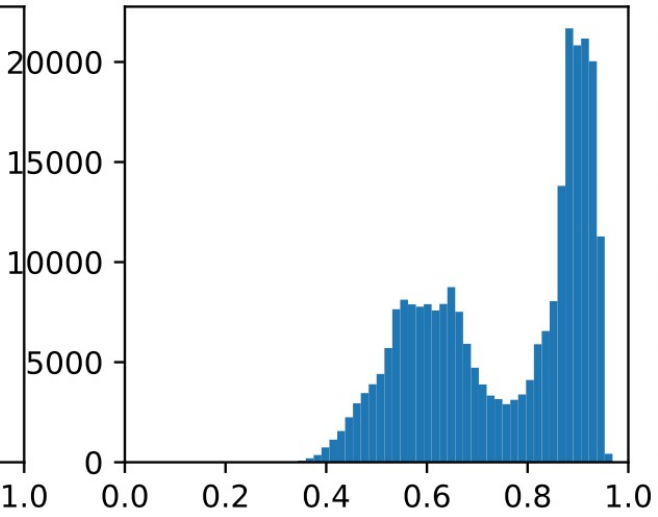
Gamma correction modifies the colors of an image acquired by an electronic system.

It is used to consider the non-linear sensibility of human eyes to the light. Here, $\boldsymbol{\gamma}$=0.4

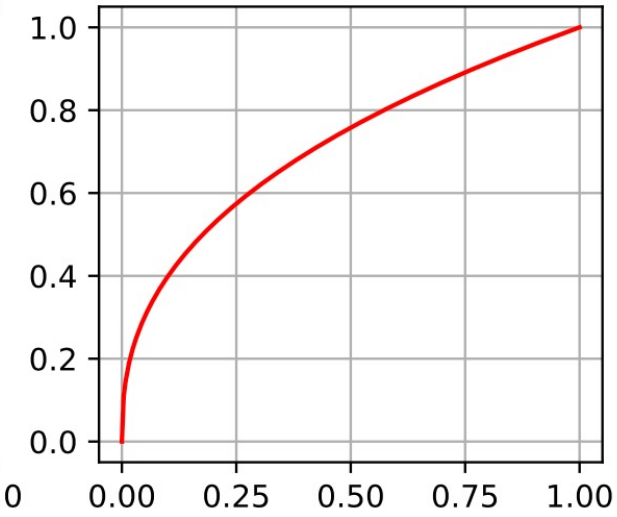
Original histogram



Transformed histogram



Transformation



Histogram spreading

Original image

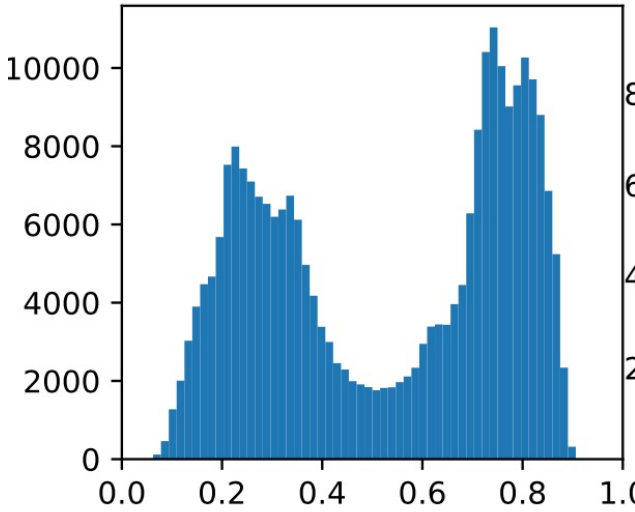
Transformed image

we suppose the pixel intensities to lie in [0, 1]

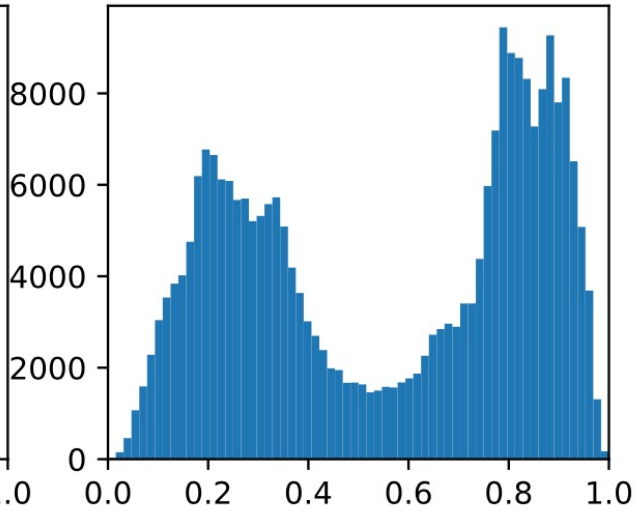
$$T(i) = \frac{i - i_{\min}}{i_{\max} - i_{\min}}$$

Histogram spreading enhances the contrast by “dilating” the histogram to the whole intensity interval.

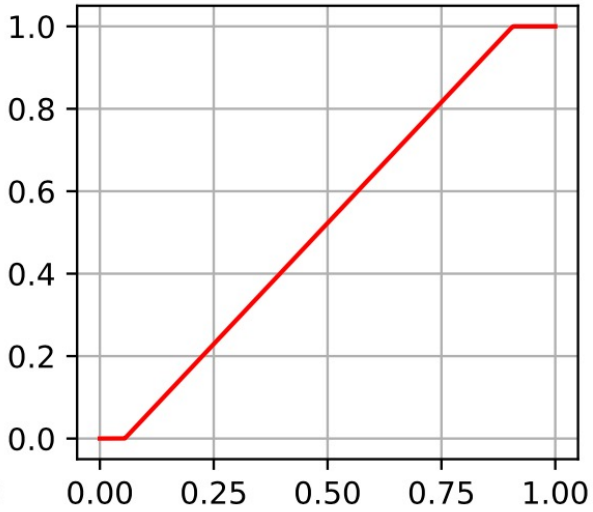
Original histogram



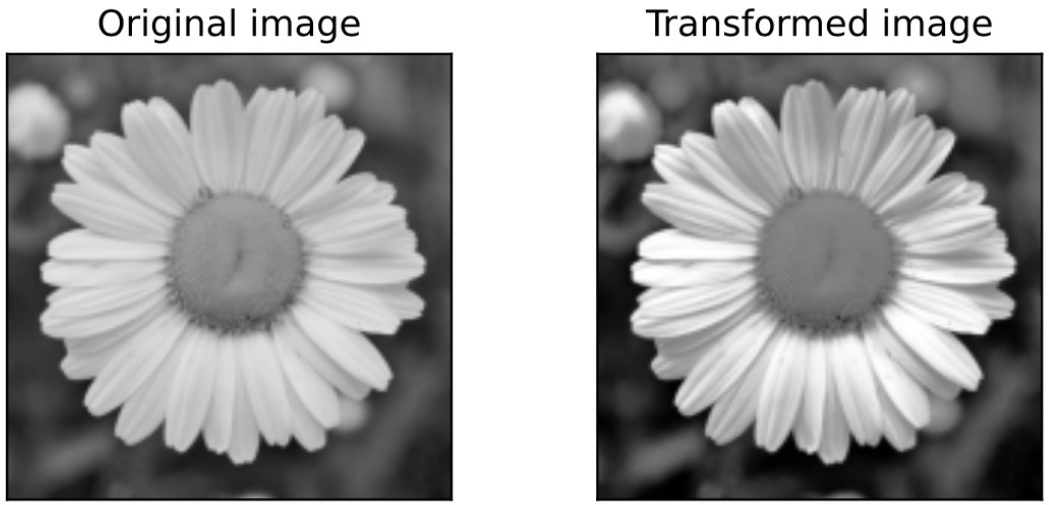
Transformed histogram



Transformation



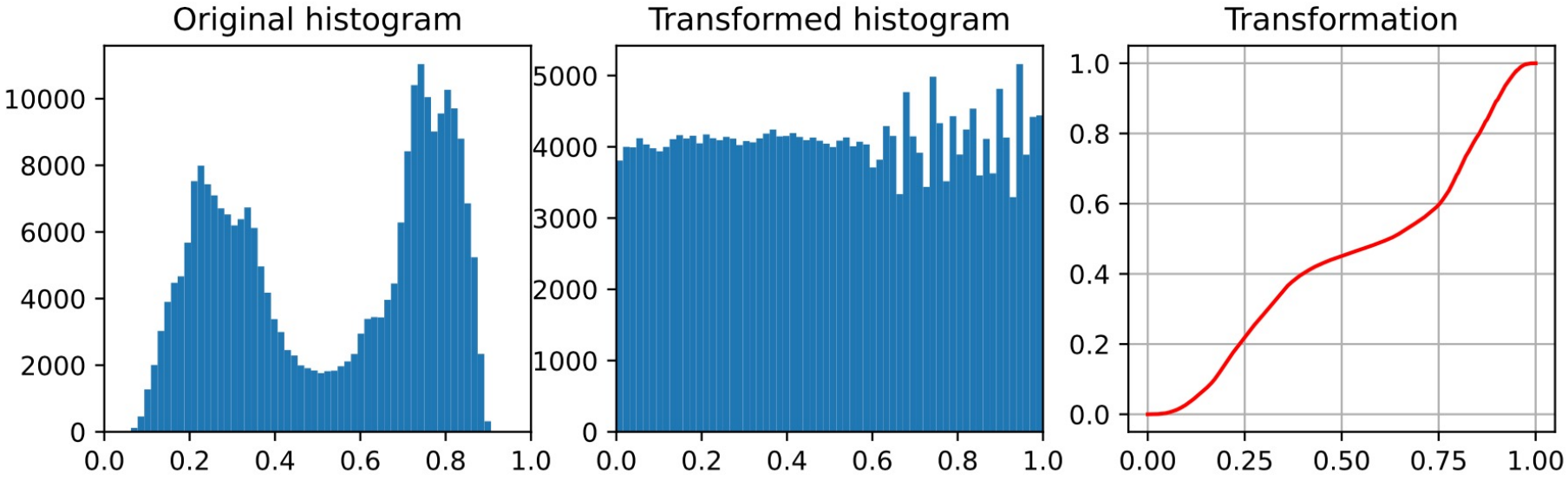
Equalization Histogram



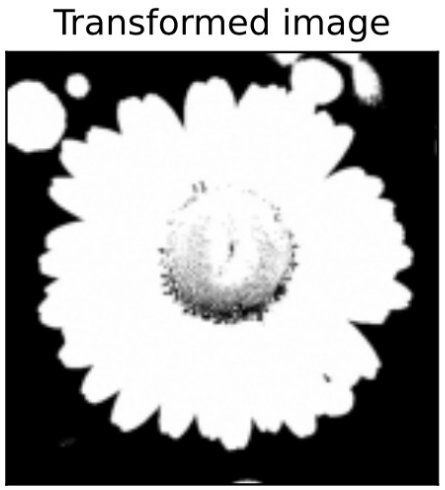
we suppose the pixel intensities to lie in [0, 1]

$$T(i) = \frac{1}{MN} \sum_{k=0}^{i} n_k$$

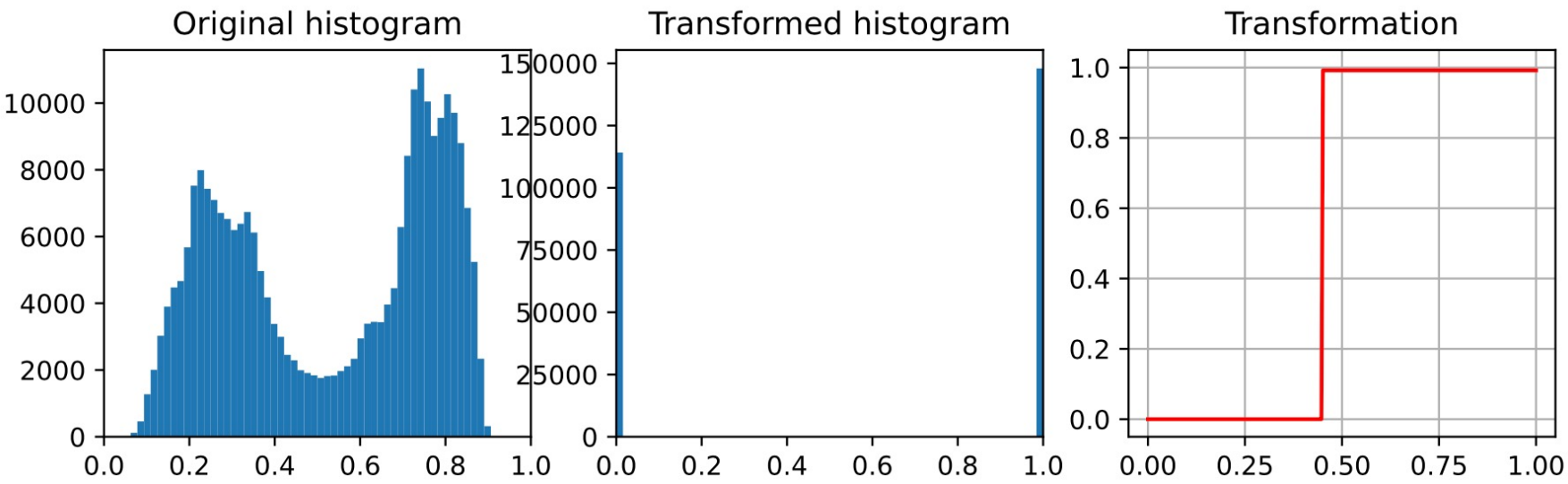
- where M and N are the image size and n_k is the number of pixels with intensity i_k.
- This transformation aims to spread the histogram over the entire intensity range, and to make the histogram as flat as possible.
- A consequence is an increasing of the image contrast



Thresholding



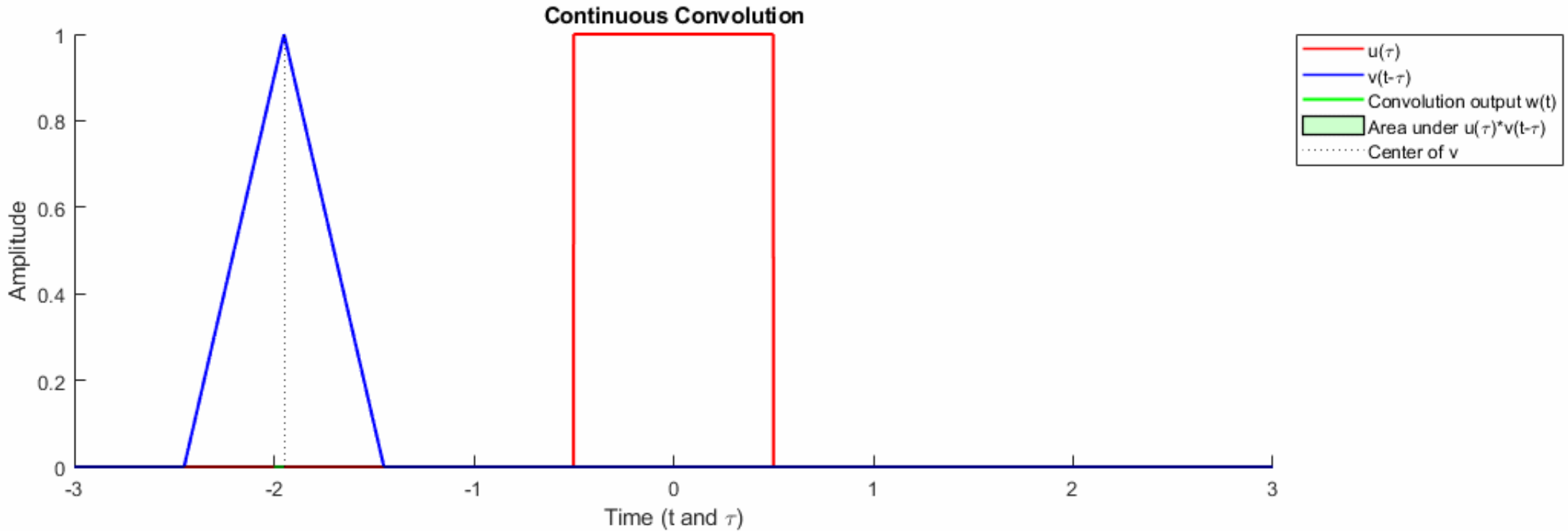
Threshold with a threshold set to 0.45

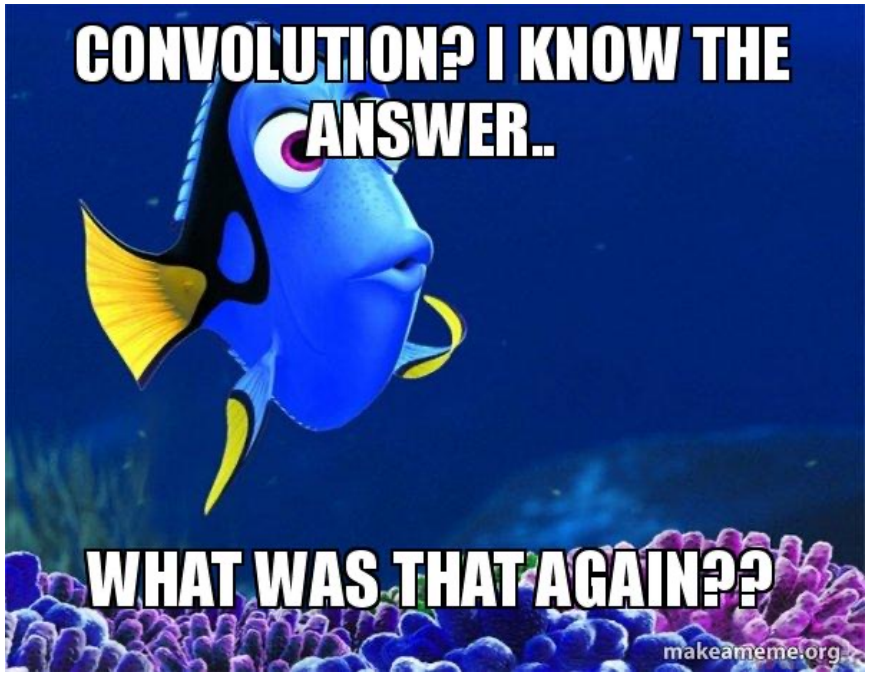


Convolution

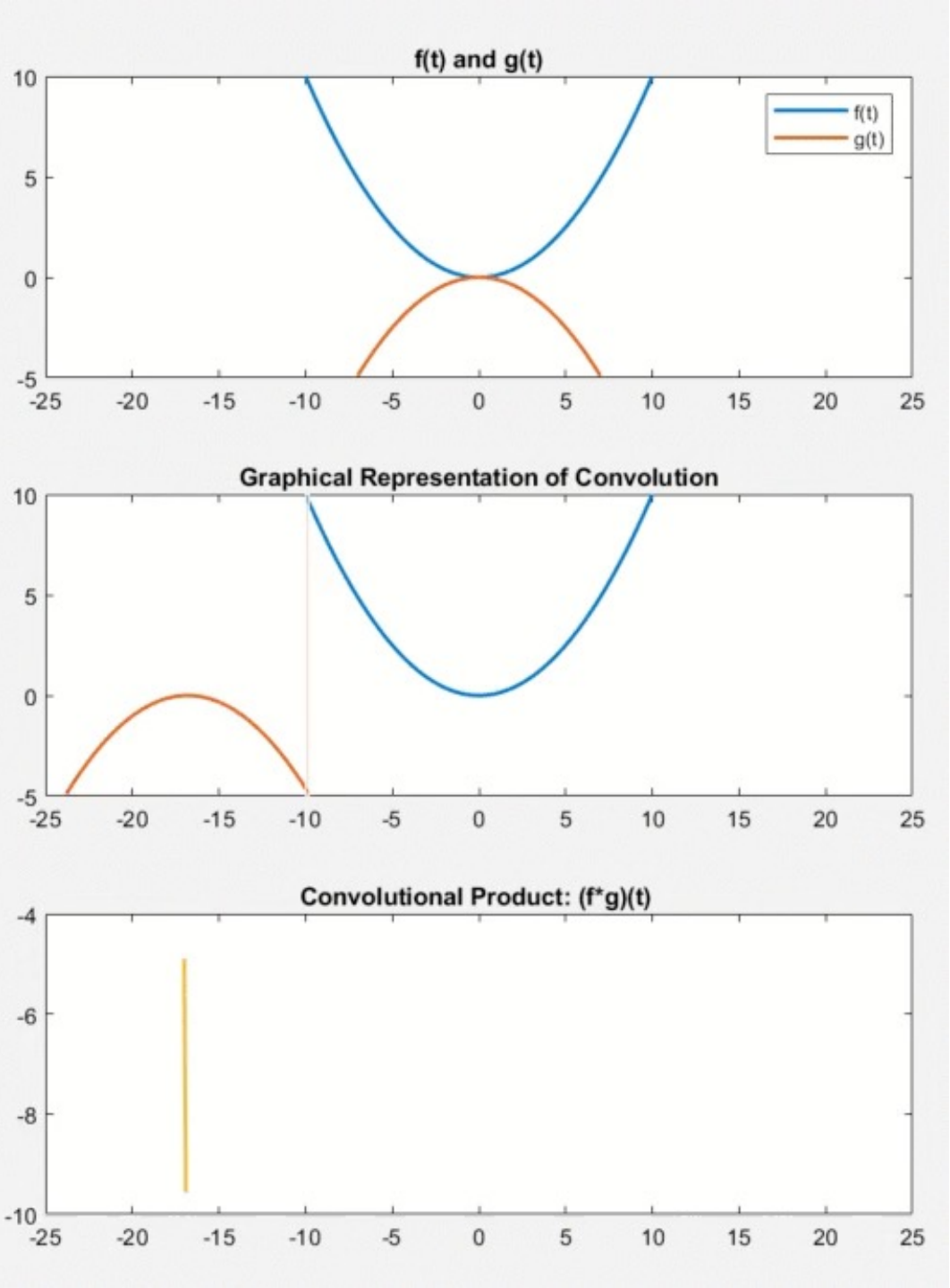
Formally, convolution is an integral that expresses the amount of overlap of one function, $f(t)$, as it is shifted over function $g(t)$, expressed as:

$$(f * g)(t) \approx^{def} \int_{-\infty}^{\infty} f(\tau)g(t-\tau)dr$$





- Convolution is a mathematical operation that combines two functions to describe the overlap between them.
- Convolution takes two functions and “slides” one of them over the other, multiplying the function values at each point where they overlap, and adding up the products to create a new function.
- This process creates a new function that represents how the two original functions interact with each other

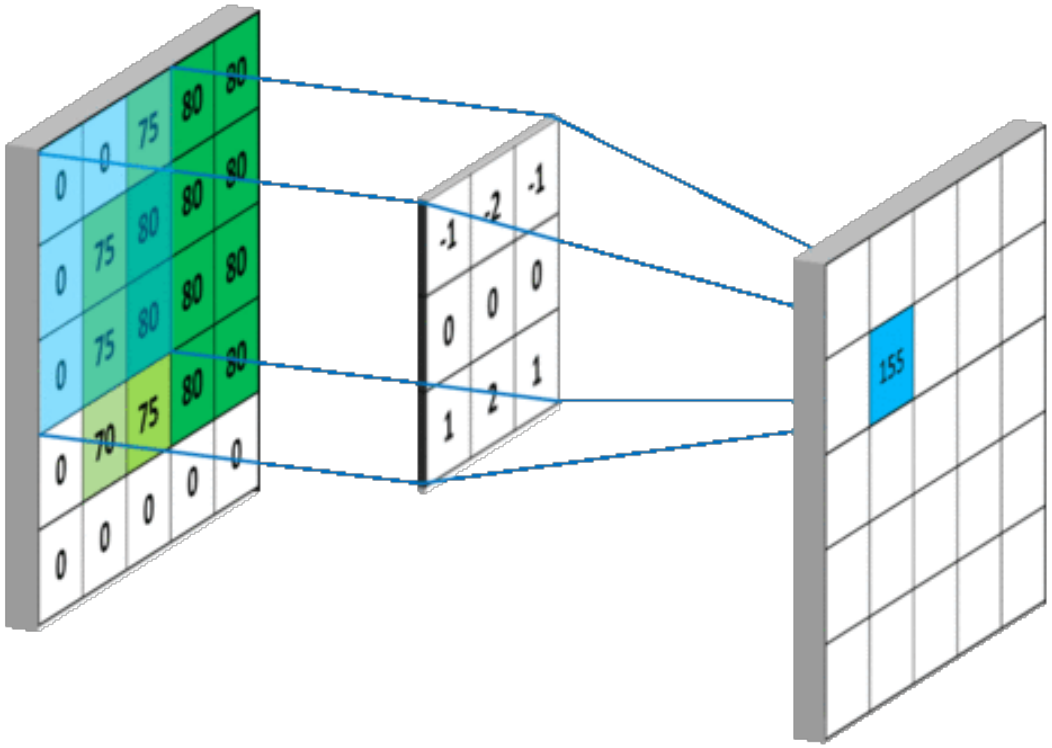


Convolution in Image Processing

In image processing, convolutional filtering can be used to implement algorithms such as:

1. edge detection,
2. image sharpening, and
3. image blurring.

This is done by selecting the appropriate kernel (convolution matrix).



Sharpening Filter Kernel

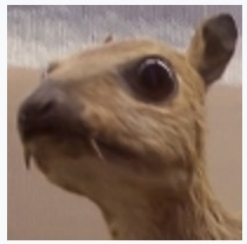
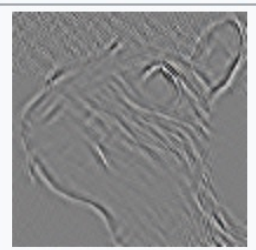
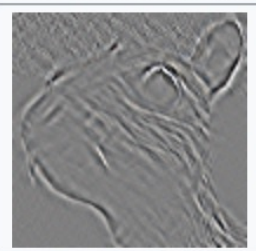
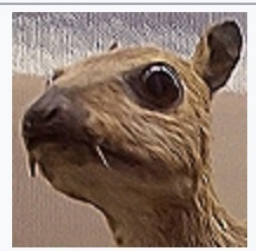
0	-1	0
-1	5	-1
0	-1	0

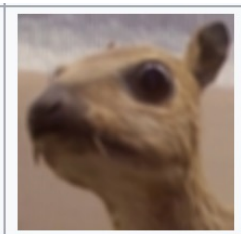
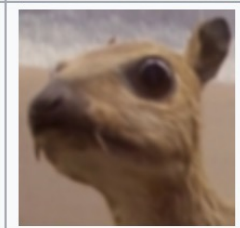
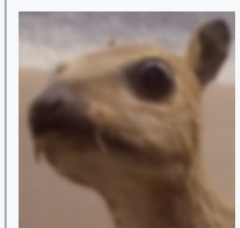
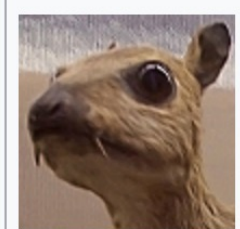
Input

0	0	0	0	0	0	0
0	1	2	2	1	0	0
0	1	2	1	2	1	0
0	0	2	3	2	0	0
0	0	5	2	2	0	0
0	0	7	2	1	1	0
0	0	0	0	0	0	0

Sharpened Output

2	5	6	1	-2
2	4	-4	5	3
-3	0	8	3	-3
-5	14	-2	5	-3
-7	28	0	0	4

Operation	Kernel ω	Image result g(x,y)
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Ridge or **edge detection**	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	

Box blur (normalized)	$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur 3 × 3 (approximation)	$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	
Gaussian blur 5 × 5 (approximation)	$\frac{1}{256}\begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$	
Unsharp masking 5 × 5 Based on Gaussian blur with amount as 1 and threshold as 0 (with no image mask)	$\frac{-1}{256}\begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & -476 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$	


```
for each image row in input image:
    for each pixel in image row:

        set accumulator to zero

        for each kernel row in kernel:
            for each element in kernel row:

                if element position  corresponding* to pixel position then
                    multiply element value  corresponding* to pixel value
                    add result to accumulator
                endif

         set output image pixel to accumulator
```

Tarea:

Parte 1:

- A una imagen, aplicar todos los métodos de transformación de histogramas vistos en clase.

Parte 2:

- Implemente el algoritmo de convolución de una imagen mostrado en clase.