# From 1D signals to images

## Lecture 4

Course of:

**Signal and imaging acquisition and modelling in environment**

14/03/2024

*Federico De Guio - Matteo Fossati*

# Intro to digital images

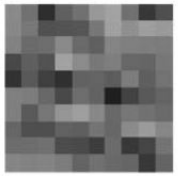
## What is a digital image?

- 2-dimensional matrix of intensity (gray or color) values

Set of Intensity values

Image coordinates are integers

$$I(u, v) \in \mathbb{P} \quad \text{and} \quad u, v \in \mathbb{N}.$$



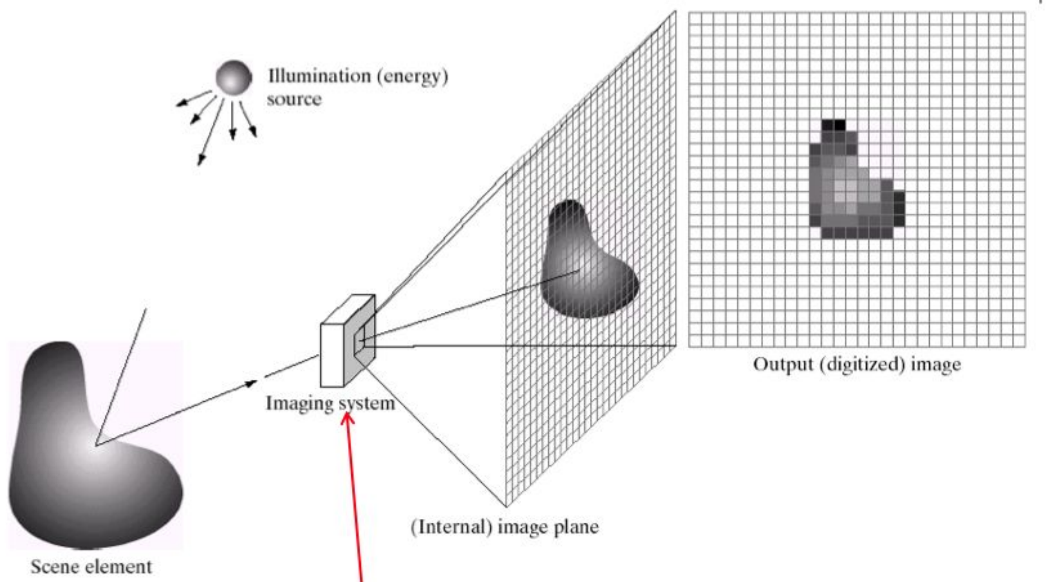
$F(x, y)$

$\rightarrow$

$$\begin{array}{|ccccccccccc|}
148 & 123 & 52 & 107 & 123 & 162 & 172 & 123 & 64 & 89 & \cdots \\
147 & 130 & 92 & 95 & 98 & 130 & 171 & 155 & 169 & 163 & \cdots \\
141 & 118 & 121 & 148 & 117 & 107 & 144 & 137 & 136 & 134 & \cdots \\
82 & 106 & 93 & 172 & 149 & 131 & 138 & 114 & 113 & 129 & \cdots \\
57 & 101 & 72 & 54 & 109 & 111 & 104 & 135 & 106 & 125 & \cdots \\
138 & 135 & 114 & 82 & 121 & 110 & 34 & 76 & 101 & 111 & \cdots \\
138 & 102 & 128 & 159 & 168 & 147 & 116 & 129 & 124 & 117 & \cdots \\
113 & 89 & 89 & 109 & 106 & 126 & 114 & 150 & 164 & 145 & \cdots \\
120 & 121 & 123 & 87 & 85 & 70 & 119 & 64 & 79 & 127 & \cdots \\
145 & 141 & 143 & 134 & 111 & 124 & 117 & 113 & 64 & 112 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots &
\end{array}$$

$I(u, v)$

## Imaging system

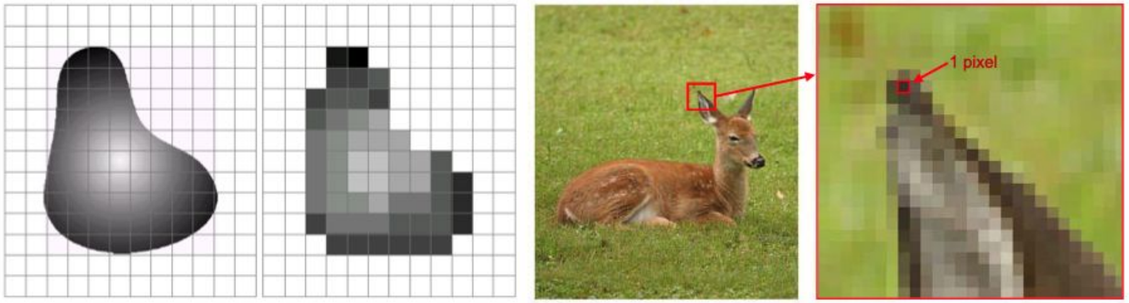



Example: a camera
Converts light to image

Credits: Gonzales and Woods

# Image digitization

- **Digitization** causes a digital image to become an **approximation** of a real scene




Real image | Digital Image (an approximation) | Real image | Digital Image (an approximation)

- 1 value per pixel: B&W or Grayscale
- 3 values per pixel: Color image (RGB)

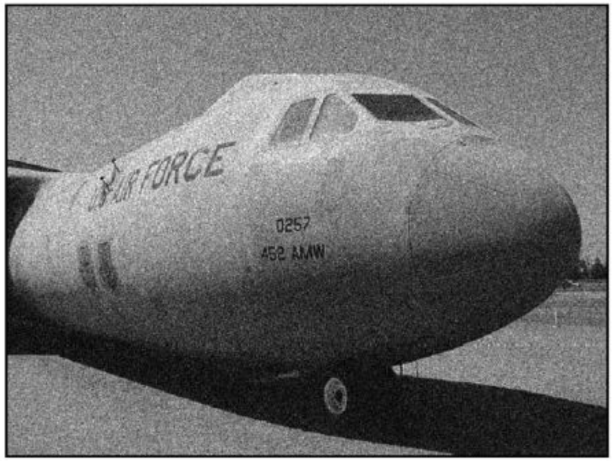
# What is image processing?

- Algorithms that alter an input image to create new image
  - **The input is an image, the output is an image**


- **Improve images for human or AI interpretation**
  - Image and features enhancement

## An example: noise removal

Noisy Image




Denoised Image

## An example: contrast adjustment

Low Contrast

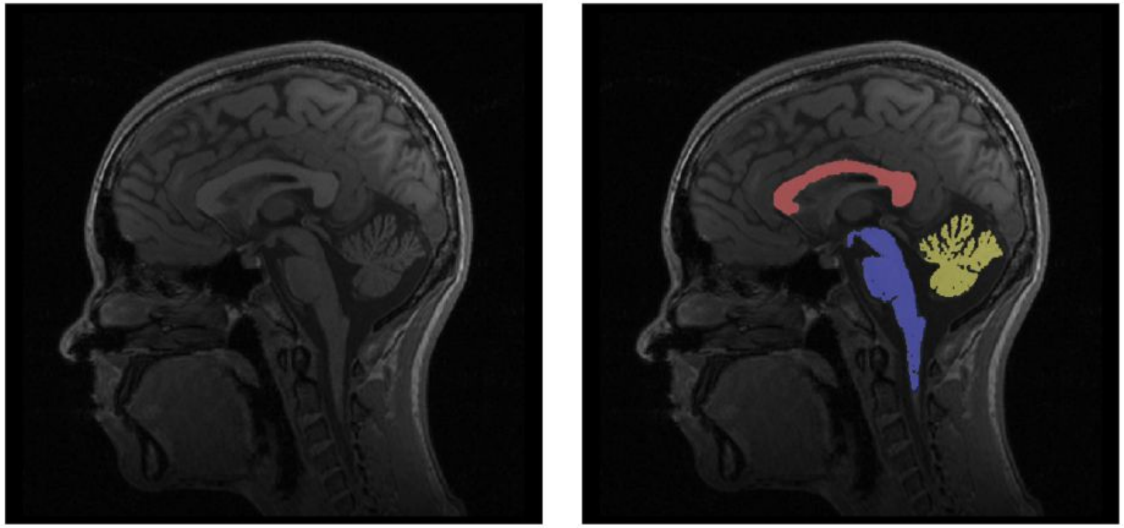
Original Contrast

High Contrast

# An example: edge detection

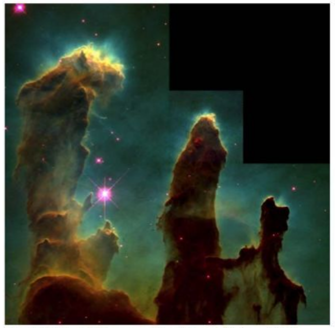
## An example: region detection and segmentation



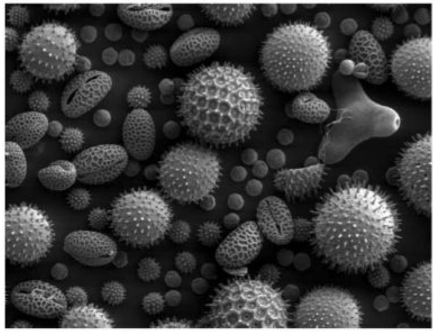
## Application of image processing

- **Applications are countless:**
  - Biology
  - Astronomy
  - Medicine
  - Security and biometric
  - Precision agriculture
  - Satellite imagery and terrain classification
  - Meteorology
  - Art
  - ...

Credit: NASA



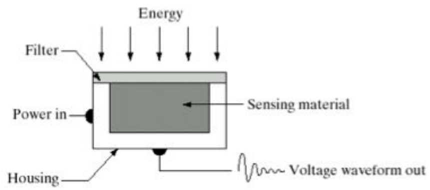
Credit: NASA, Jeff Hester, and Paul Scowen (Arizona State)
More info here



Credit: Dartmouth Electron Microscopy Facility

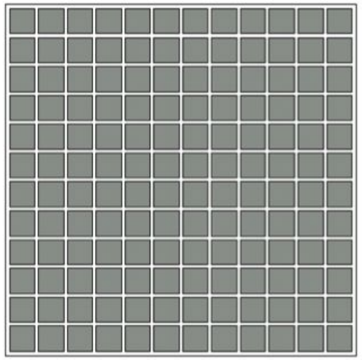
# Image acquisition

- The incoming energy (**light**) hits a sensor material which is **responsive to that type of energy**
- Collections of sensors are arranged to capture images
  - **Record image values at discrete x, y (e.g. 10 MegaPixel)**
  - **Discrete set of energy levels  (e.g. 8-bit grayscale)**




Imaging Sensor

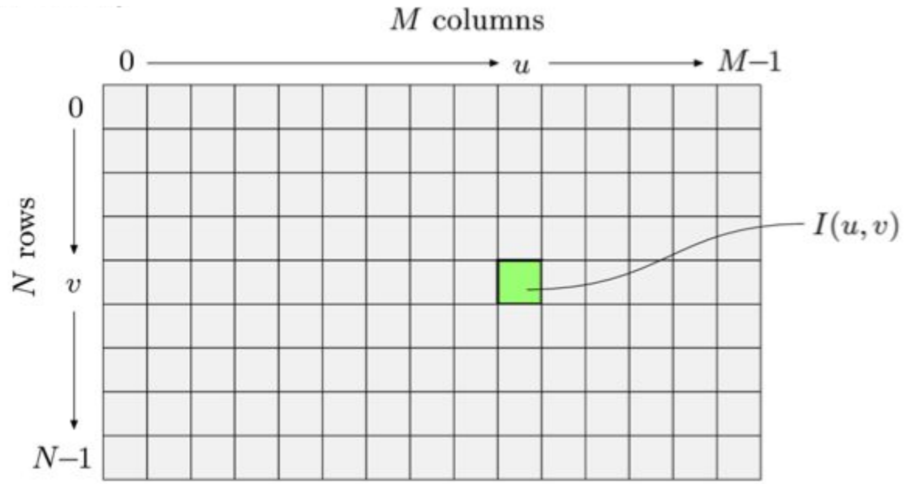
Line of Image Sensors



Array of Image Sensors

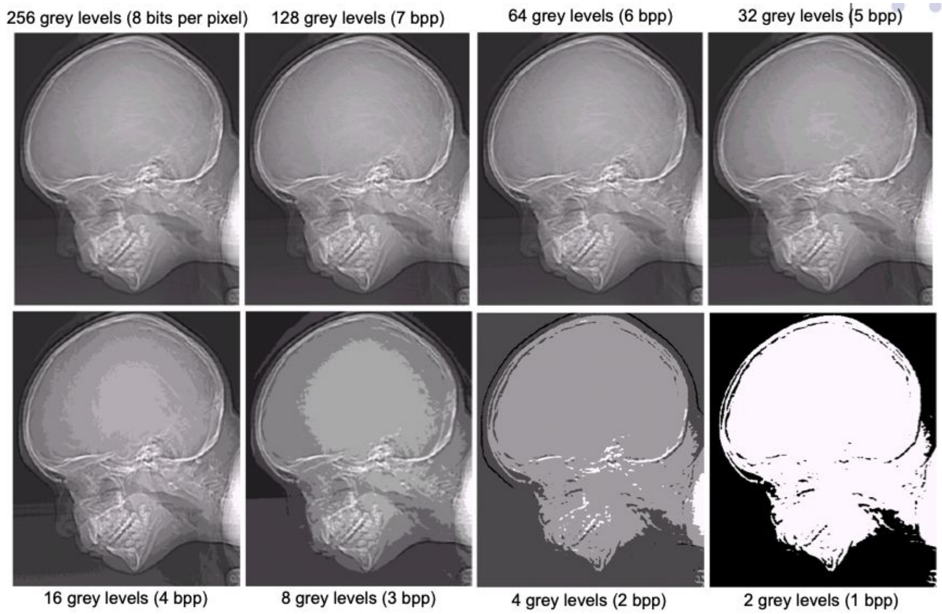
## Representing images

- The image data structure is a 2D array of pixel values
- Pixel values are gray levels in the range 0-255 (or RGB colors)



## Intensity level resolution

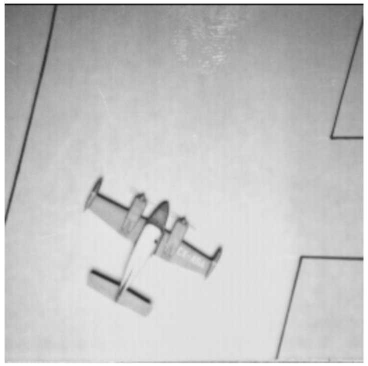
- Is the **number of intensity levels** used to represent the image

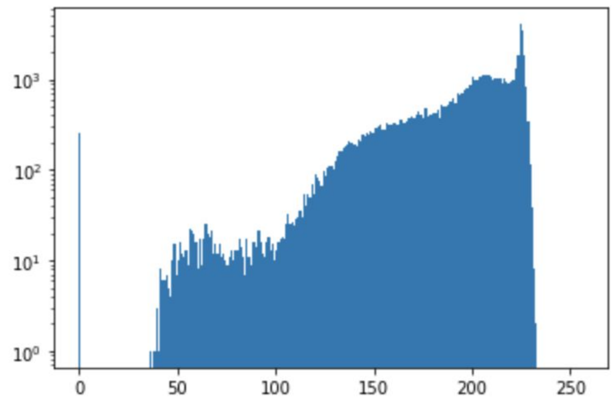


# Histograms

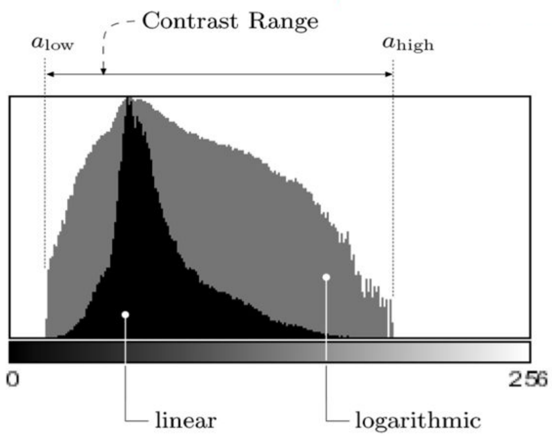
## Histograms

- They show **how many times each intensity value occurs in the picture**





# Histograms



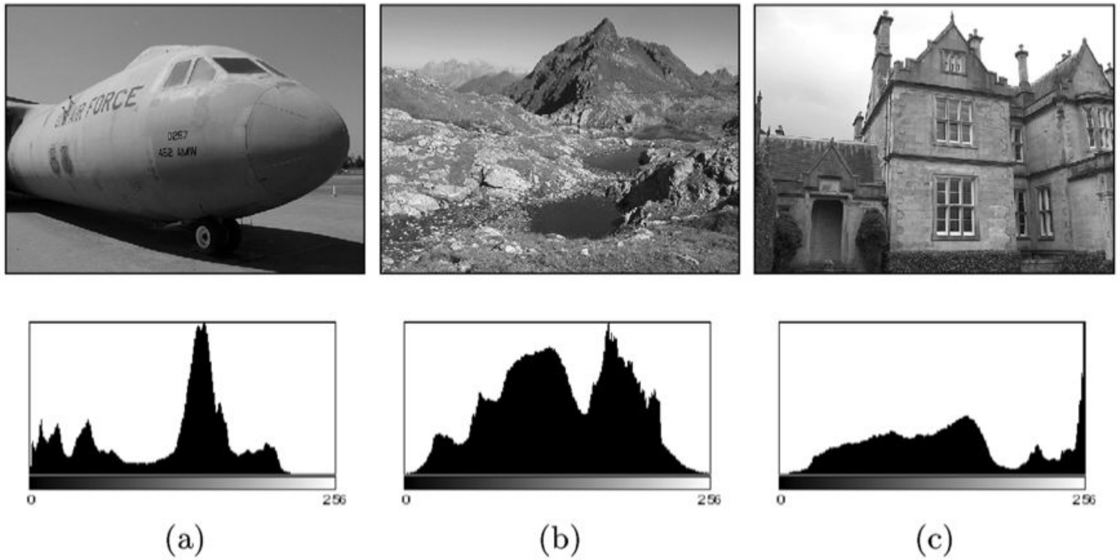

Brightness

$$B(I) = \frac{1}{wh} \sum_{v=1}^{h} \sum_{u=1}^{w} I(u, v)$$

2. Divide by total number of pixels

1. Sum up all pixel intensities

# Exposure




**Image**

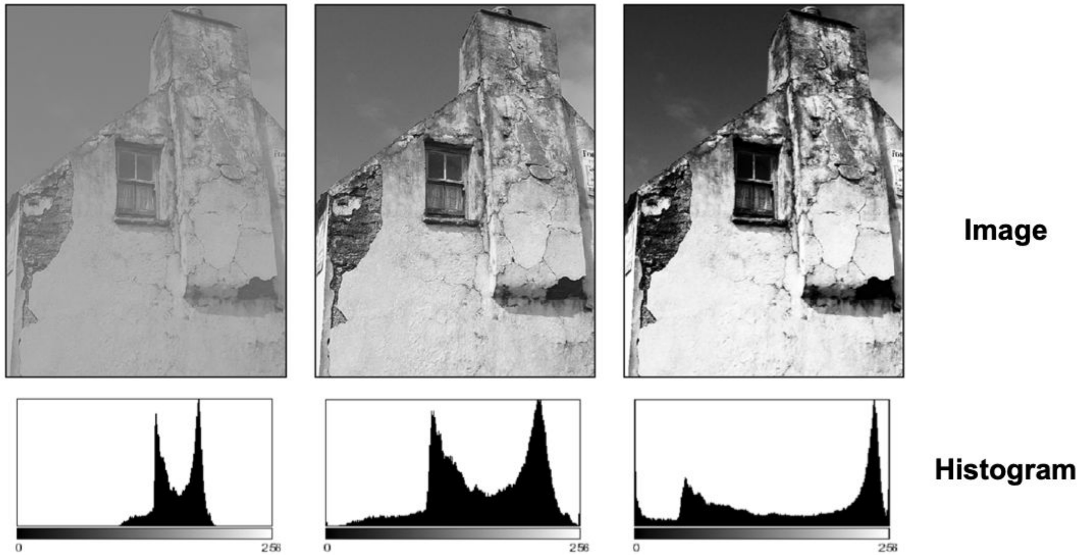
**Histogram**

**Underexposed**

**Properly Exposed**

**Overexposed**

# Contrast

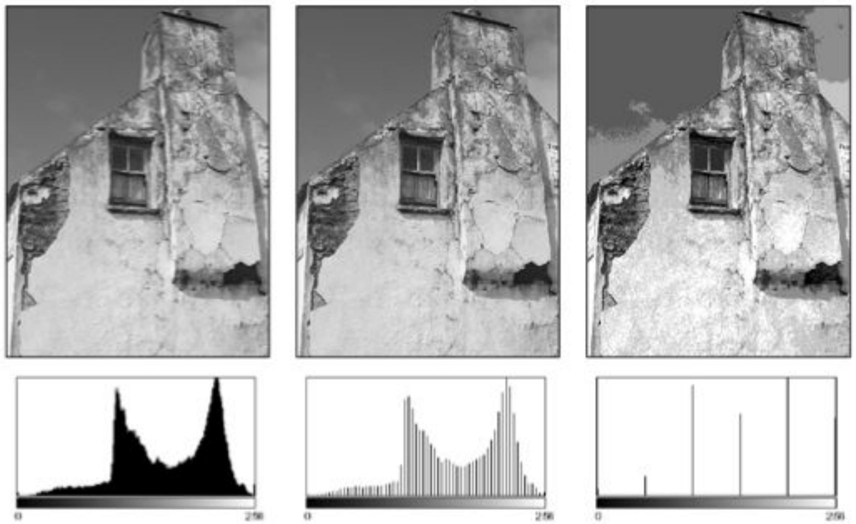



(a) Low contrast

(b) Normal contrast

(c) High contrast

# Dynamic range



(a) High Dynamic Range

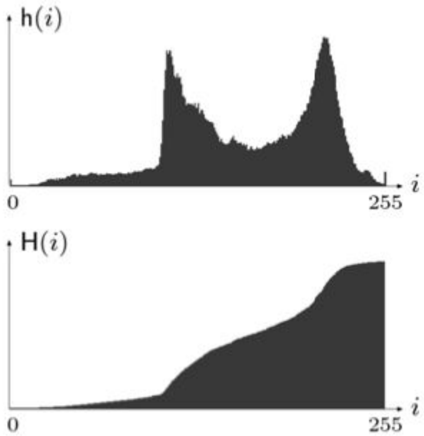
(b) Low Dynamic Range (64 intensities)

(c) Extremely low Dynamic Range (6 intensity values)

## Computing histograms

- Question for you: **How would you compute the histogram of an image?**
    - Range of the x axis?
    - Number of bins?
    - What about color images?

- Sometimes the **cumulative distribution** gives useful information
    - E.g. for histograms equalization



# Point operations

## What do they do?

- Change the pixel's intensity value according to some function
  - Homogeneous operations

- New pixel intensity **doesn't depend on:**
  - The pixel location (x, y)
  - The pixel neighbours

- New pixel intensity **depends on:**
  - The pixel previous intensity
  - The mapping function

$$a' \leftarrow f(a)$$

$$I'(u,v) \leftarrow f\big(I(u,v)\big)$$

## Example: clamping

- Deals with pixel values outside the displayable range
  - if (a > 255): a = 255;
  - if (a < 0): a = 0;

- This function will **clamp** (force) all values to fall within range [a, b]

$$f(p) = \begin{cases} a & \text{if } p < a \\ p & \text{if } a \leq p \leq b \\ b & \text{if } p > b \end{cases}$$

## Example: thresholding

- The input values below **threshold** $\mathbf{a_{th}}$ are set to $\mathbf{a_0}$
- The input values below **threshold** $\mathbf{a_{th}}$ are set to $\mathbf{a_1}$

$$f_{\text{threshold}}(a) = \begin{cases} a_0 & \text{for } a < a_{\text{th}} \\ a_1 & \text{for } a \geq a_{\text{th}} \end{cases}$$

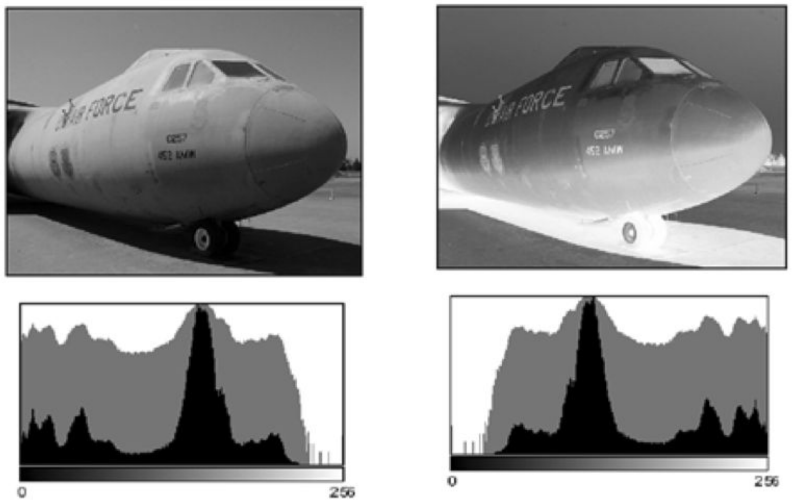
- Converts the grayscale image to binary image

Original Image

Thresholded Image

## Example: negative images

$$f_{\text{invert}}(a) = -a + a_{\max} \; = \; a_{\max} - a$$



# Example: grey levels transformation - the power law

- There are many options. A commonly used transformation is the **power law**
- Map a narrow range of dark input values into a wider range of output values or vice versa

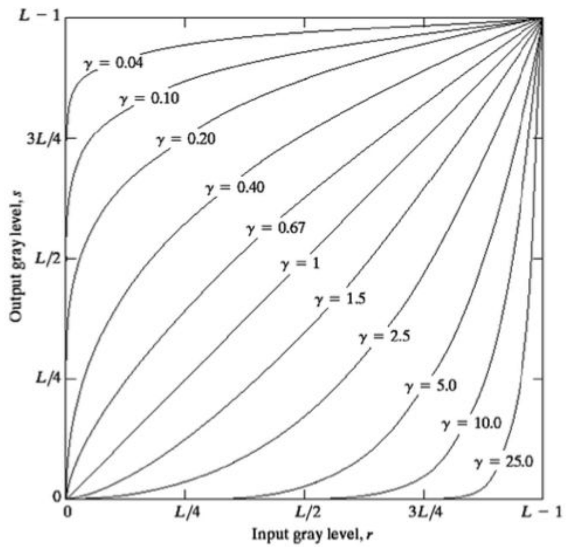
Power

$$s = c * r^{\gamma}$$

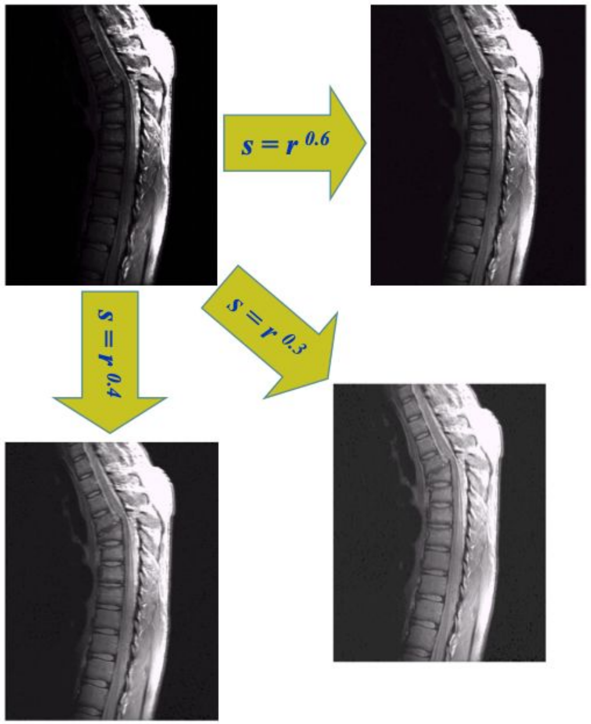
Old pixel value

New pixel value

Constant

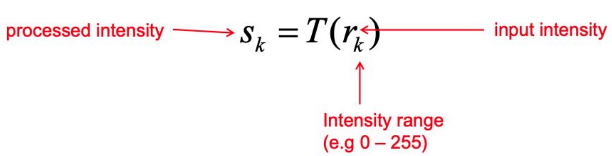


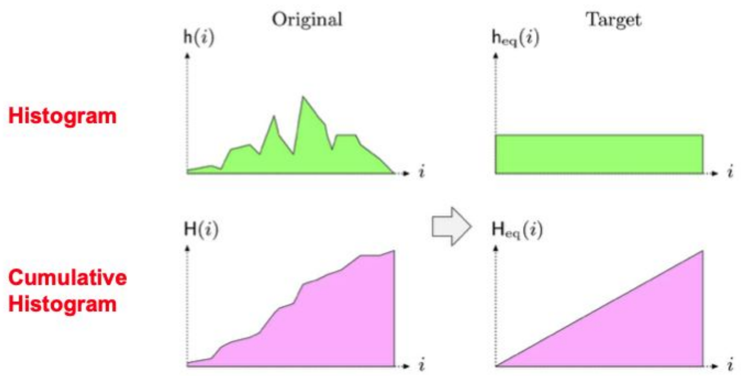
## Different  power values highlight different details



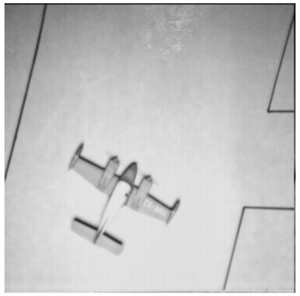
## Example: histograms equalization

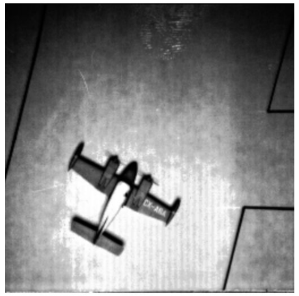
- Adjust two different images to make their histograms (intensity distributions) similar
- Apply a point operation that changes the histogram to a **uniform distribution** as much as possible

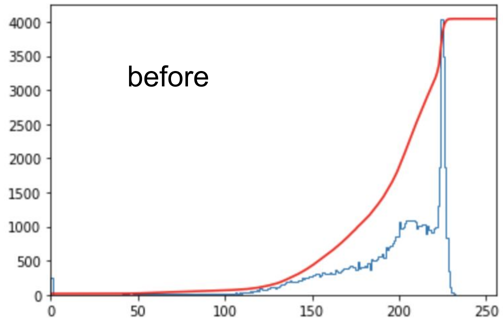


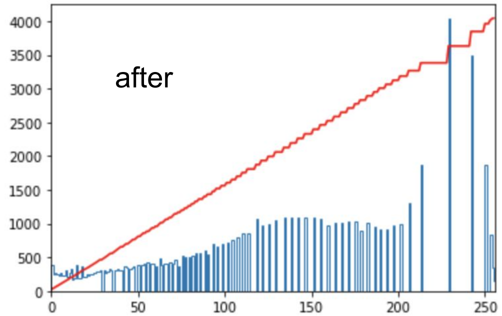



# Example: linear histograms equalization








Point operation that returns
Linear equalized value of *a*

$$f_{\text{eq}}(a) = \left\lfloor \mathsf{H}(a) \cdot \frac{K-1}{MN} \right\rfloor$$

Cumulative Histogram: Σ how many
times intensity a occurs

# Filters

## What is a filter?

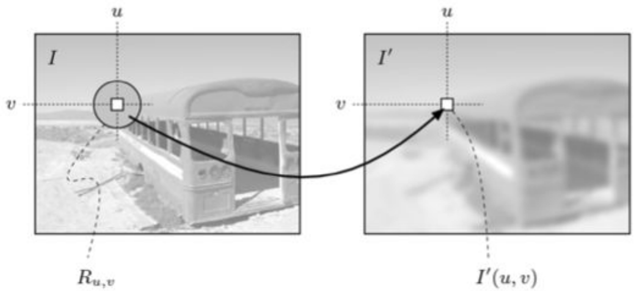
- Capabilities of point operations are limited
- **Filters: combine pixel's value + values of neighbors**

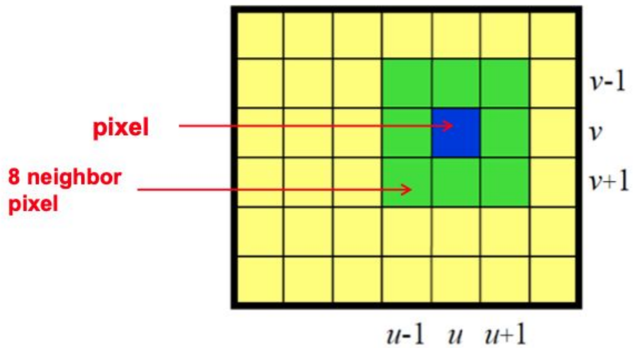
- **E.g. burring:** compute the average intensity of blocks of pixels

## Average of a 3x3 matrix

- **Blurring:** Replace each pixel with the average intensity of pixel+neighbors

$$I'(u,v) \leftarrow \frac{p_0 + p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8}{9}$$

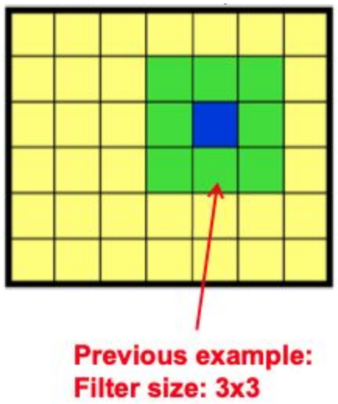


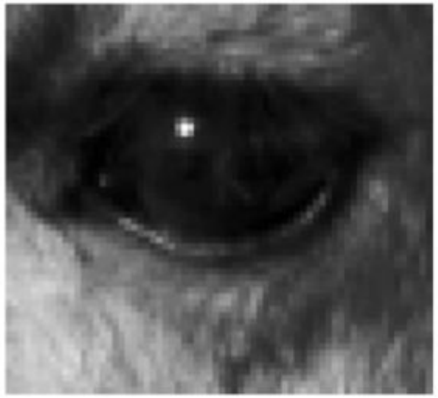

- **In general:** $I'(u,v) \leftarrow \frac{1}{9} \cdot \sum_{j=-1}^{1} \sum_{i=-1}^{1} I(u+i, v+j)$

## Filters parameters

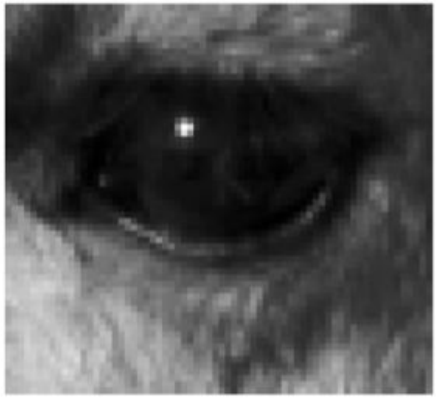
- **Filter size (size of neighborhood):** 3x3, 5x5, 7x7, ...,21x21,..
- **Filter shape:** not necessarily square. Can be rectangle, circle, etc
- **Filter weights:** May apply unequal weighting to different pixels
- **Filters function:** can be linear (a weighted summation) or nonlinear



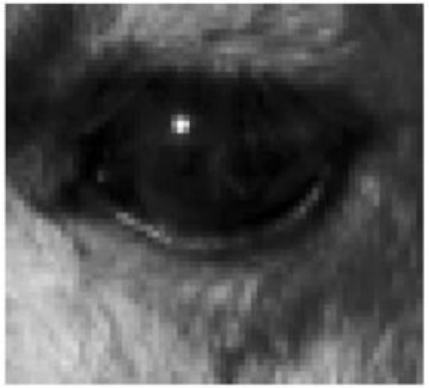
## What does this filter do?



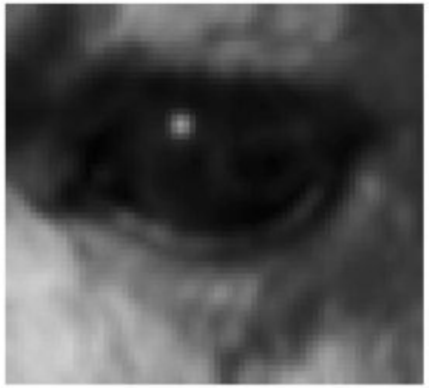
| | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |



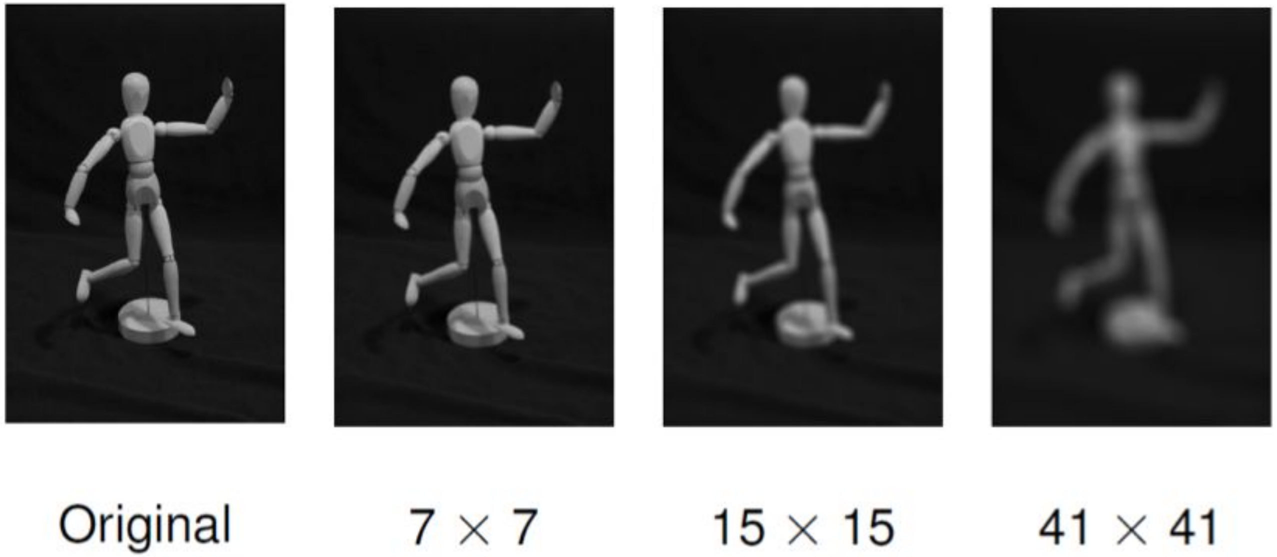
## What does this filter do?



$$\frac{1}{9}\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix}$$



## Effect of filter size



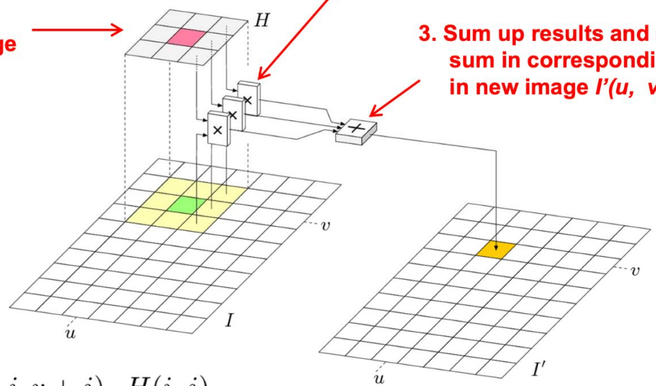
# Applying linear filters: convolution

**For each image position *I(u,v)*:**

**1. Move filter matrix *H* over image such that *H(0,0)* coincides with current image position *(u,v)***

**2. Multiply all filter coefficients *H(i,j)* with corresponding pixel *I(u + i, v + j)***

**3. Sum up results and store sum in corresponding position in new image *I'(u, v)***




**Stated formally:**

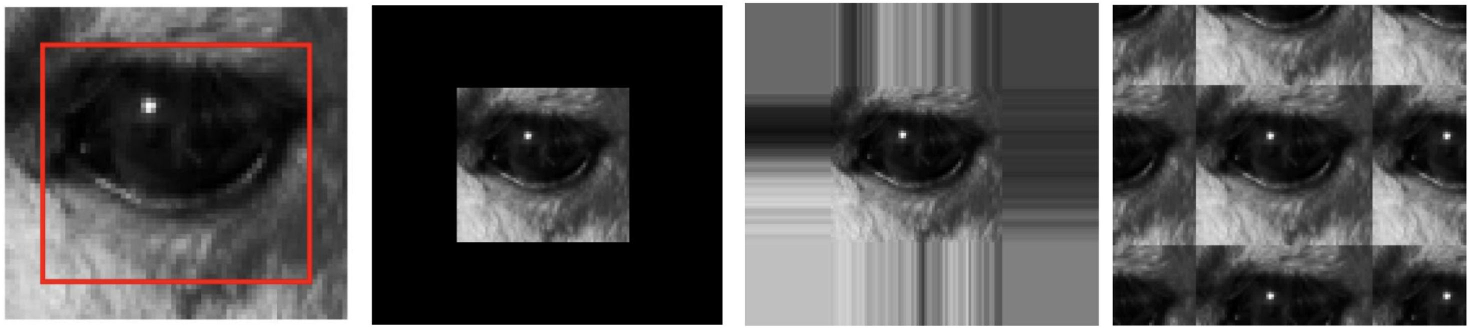
$$I'(u,v) \leftarrow \sum_{(i,j)\in R_H} I(u+i, v+j) \cdot H(i,j)$$

**$R_H$ is set of all pixels Covered by filter. For 3x3 filter, this is:**

$$I'(u,v) \leftarrow \sum_{i=-1}^{i=1} \sum_{j=-1}^{j=1} I(u+i, v+j) \cdot H(i,j)$$

## How to handle the boundaries?

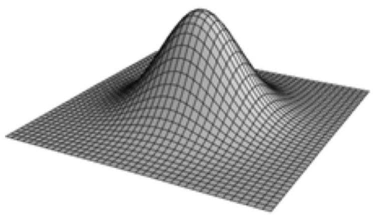
Crop | Pad | Extend | Wrap



## Smoothing filters

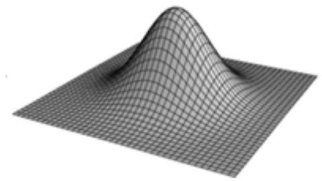
- More effective smoothing filters can be generated by associating different weights to the pixels in the neighbourhood
  - **Larger weight applied to the center**

$$H(i,j) = \begin{bmatrix} 0.075 & 0.125 & 0.075 \\ 0.125 & \mathbf{0.2} & 0.125 \\ 0.075 & 0.125 & 0.075 \end{bmatrix}$$

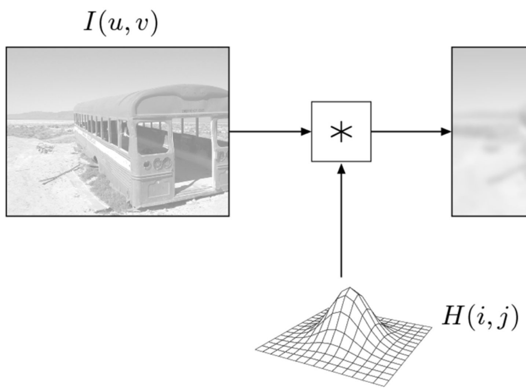


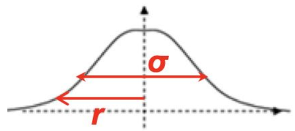
# An example: the gaussian filter

$$G_\sigma(r) = e^{-\frac{r^2}{2\sigma^2}} \qquad \text{or} \qquad G_\sigma(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$



- **σ is width** (standard deviation)
- **r is distance from center**



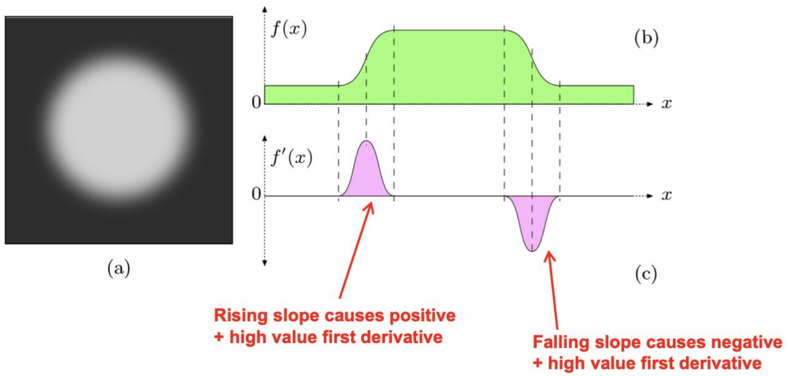

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 2 | 1 | 0 |
| 1 | 3 | 5 | 3 | 1 |
| 2 | 5 | 9 | 5 | 2 |
| 1 | 3 | 5 | 3 | 1 |
| 0 | 1 | 2 | 1 | 0 |

**Gaussian filter**

## Other types of filters: edge detection

- Filters can be used to highlight features in the image
- For example **detect the edges as sharp changes in brightness**
    - Can occur in boundaries between objects or within the same object (e.g. zebras)
- Ideal edges are step functions. In real life the **step** is smooth and **can be detected by looking at the first derivative**

$$f'(x) = \frac{df}{dx}(x)$$

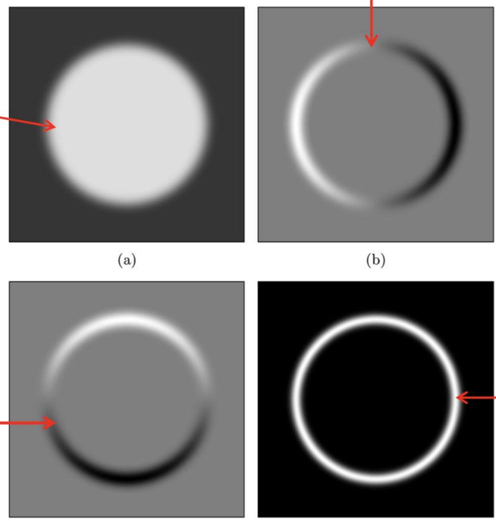


# Derivative filters

**Gradient slope in horizontal direction**

$$H_x^D = \begin{bmatrix} -0.5 & \mathbf{0} & 0.5 \end{bmatrix} = 0.5 \cdot \begin{bmatrix} -1 & \mathbf{0} & 1 \end{bmatrix}$$

**A synthetic image**



(a) (b) (c) (d)

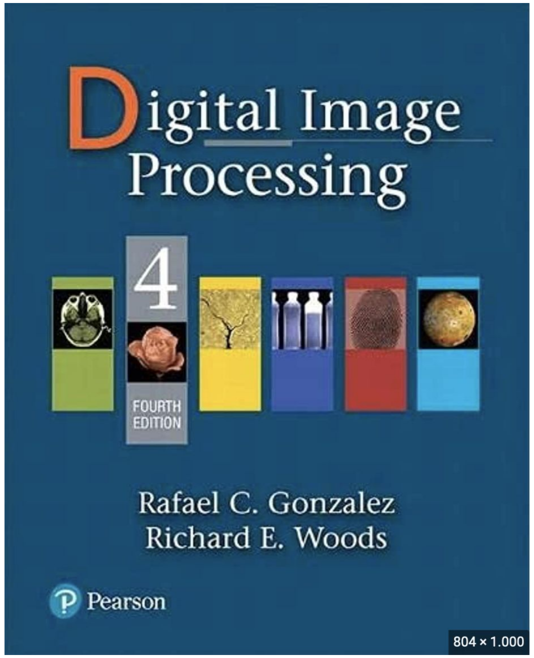
**Magnitude of gradient**

$$H_y^D = \begin{bmatrix} -0.5 \\ \mathbf{0} \\ 0.5 \end{bmatrix} = 0.5 \cdot \begin{bmatrix} -1 \\ \mathbf{0} \\ 1 \end{bmatrix}$$

**Gradient slope in vertical direction**

# There are many more filters

- You have seen them in other courses
- A good reference is the book
  - **Digital image processing-Pearson (2018)**
  - **Gonzalez, Rafael C. - Woods, Richard E.**



# Exercise

Let’s gain some familiarity with operations on a digital image

## Today's exercise with ***OpenCV***

- Pick an image among the ones here (suggestion: start with a grayscale image) and:
  - Show the original picture
  - Display its histogram and its cumulative distribution: comment the histogram
  - Transform the image from grayscale to B&W  and show its histogram: comment the histogram
  - Equalize the image so that the full dynamic range is used, and show its histogram
  - Show the negative of the image
  - Apply a filter that enhances the edges
  - What features are highlighted?
  - Mirror, zoom and rotate the image (relevant for data augmentation)

- A basic example on how to use **openCV** can be found here