Image fundamentals

Francesca Odone francesca.odone@unige.it

UniGe | MaLGa

A proloque on visual perception

Computer vision in the core of AI

- Visual perception of the surrounding environment is a key "input" of an intelligent system
- **computer vision** studies and develops algorithms for automatic scene understanding
- These algorithms allow us to extract **descriptions** of the world from (mainly) **images and image sequences**

geometry (shapes, distances,...),
dynamic (movements, actions, ...),
semantics (classes, properties,..)

It sounds easy

In 1966 Marvin Minsky, one of the fathers of AI, underestimates the visual perception complexity and assigns a project to a summer intern requiring to “solve the vision of a computer”

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
PROJECT MAC

Artificial Intelligence Group
Vision Memo. No. 100.

July 7, 1966

THE SUMMER VISION PROJECT

Seymour Papert

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition".

!

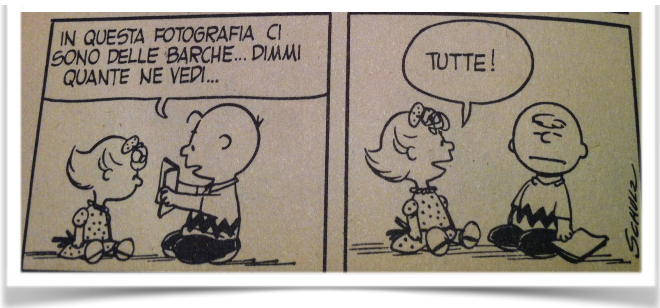
It sounds easy but it is crucial

Intelligence and Perception

In the evolution the perception of surrounding environment played a crucial role

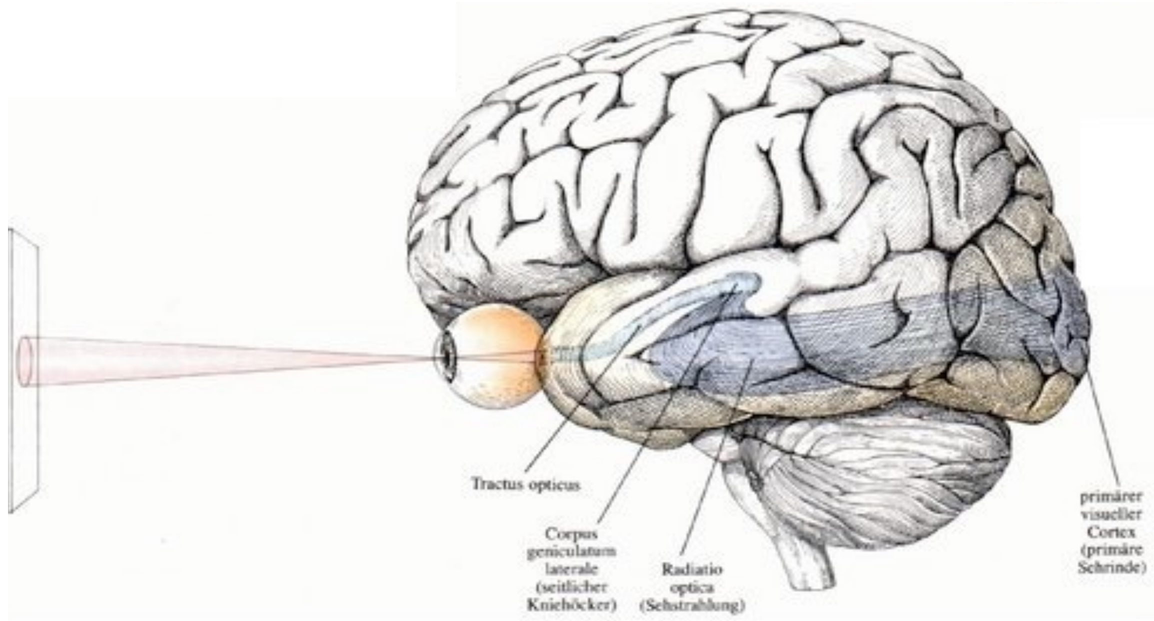
In this photo there are some boats...
Tell me how many do you see

All of them!

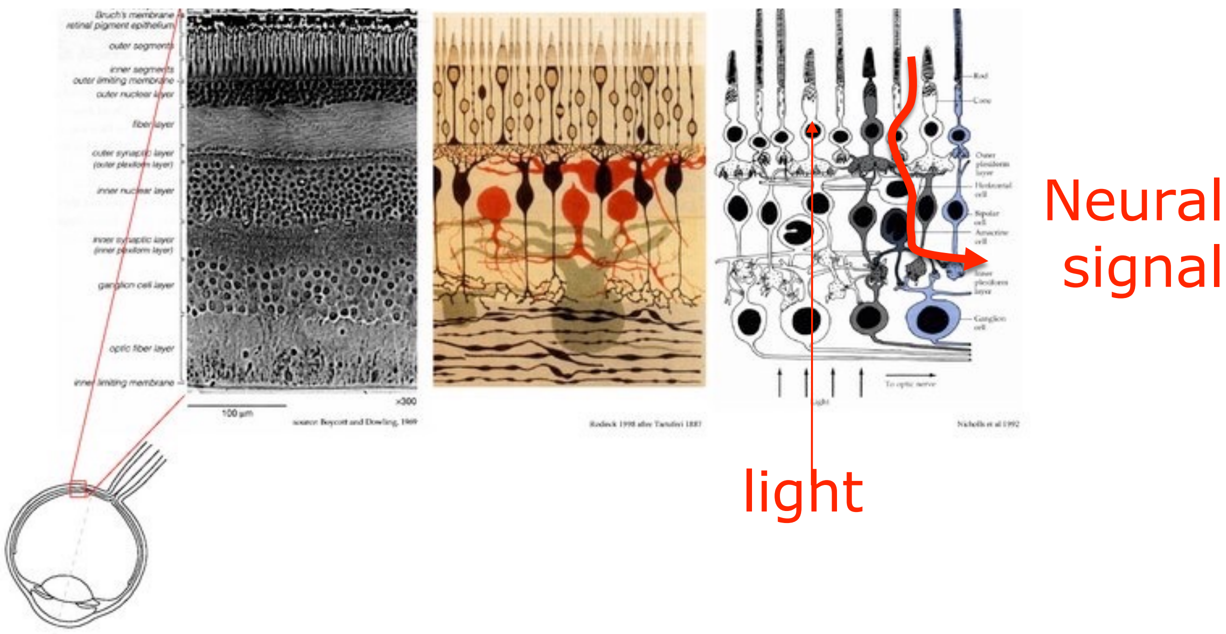


Vision is an “innate” ability of humans and, as such, it is often given for granted

Human vision

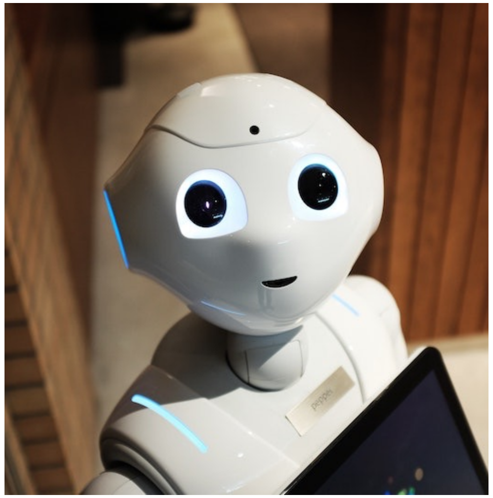


Hubel 1985



Computer vision

A camera can play the role of the computer "eye"



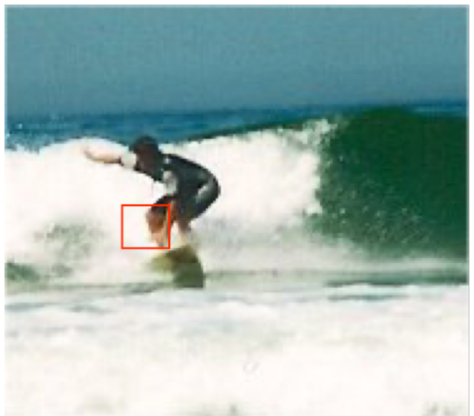
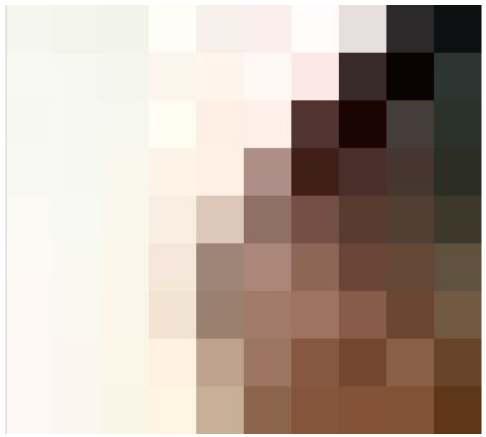
- The acquired *images* are ***meaningless***, until they are processed by computer vision algorithms

98	103	102	110	118	118	119	119	118	118	109	88
98	105	101	110	118	118	119	118	116	113	105	84
92	98	96	109	116	121	130	130	142	141	151	145
95	98	98	104	110	112	124	127	148	147	157	159
95	98	98	104	110	112	124	127	148	147	157	159
103	104	107	111	116	121	128	128	137	135	146	169
101	106	106	110	116	119	128	128	134	133	145	166
99	109	106	118	127	131	143	145	154	153	155	168
102	110	110	121	131	136	148	148	157	157	160	169
102	110	111	124	136	140	153	154	164	165	167	174
105	113	112	124	130	135	147	147	159	159	167	175
104	113	112	125	134	137	144	147	161	161	169	177
102	110	108	122	131	131	140	140	149	150	157	168
103	109	109	121	128	131	139	140	149	148	156	167
101	106	103	116	127	133	144	143	148	148	149	159
84	94	91	103	113	118	132	134	145	146	146	149
85	92	91	103	114	119	134	135	146	145	146	149
70	82	81	91	97	100	112	115	131	130	139	142
70	82	81	91	97	100	114	115	131	132	139	142
77	76	76	82	89	89	100	101	115	113	127	135
111	85	84	79	81	81	90	90	102	100	111	125
107	86	88	79	79	79	88	88	100	101	110	126

- Similarly, the image formed on the retina needs to be interpreted by the brain

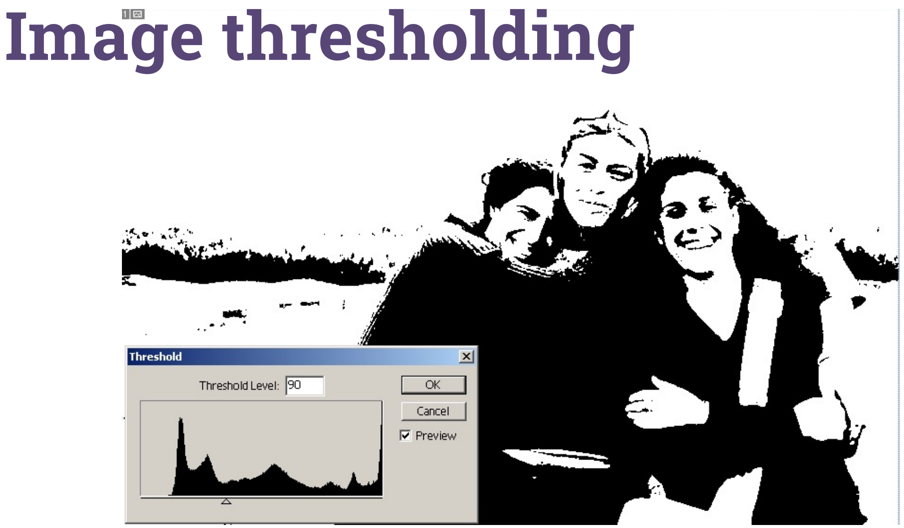
Digital images

- A digital image is a matrix of elements called **pixels** (*pic*ture *el*ements) each one associated with an numeric encoding of a “concept” (intensity, color, depth, temperature, ...)
- The **grid structure** and **spatial coherence** of neighbouring pixels are very important properties of this data type

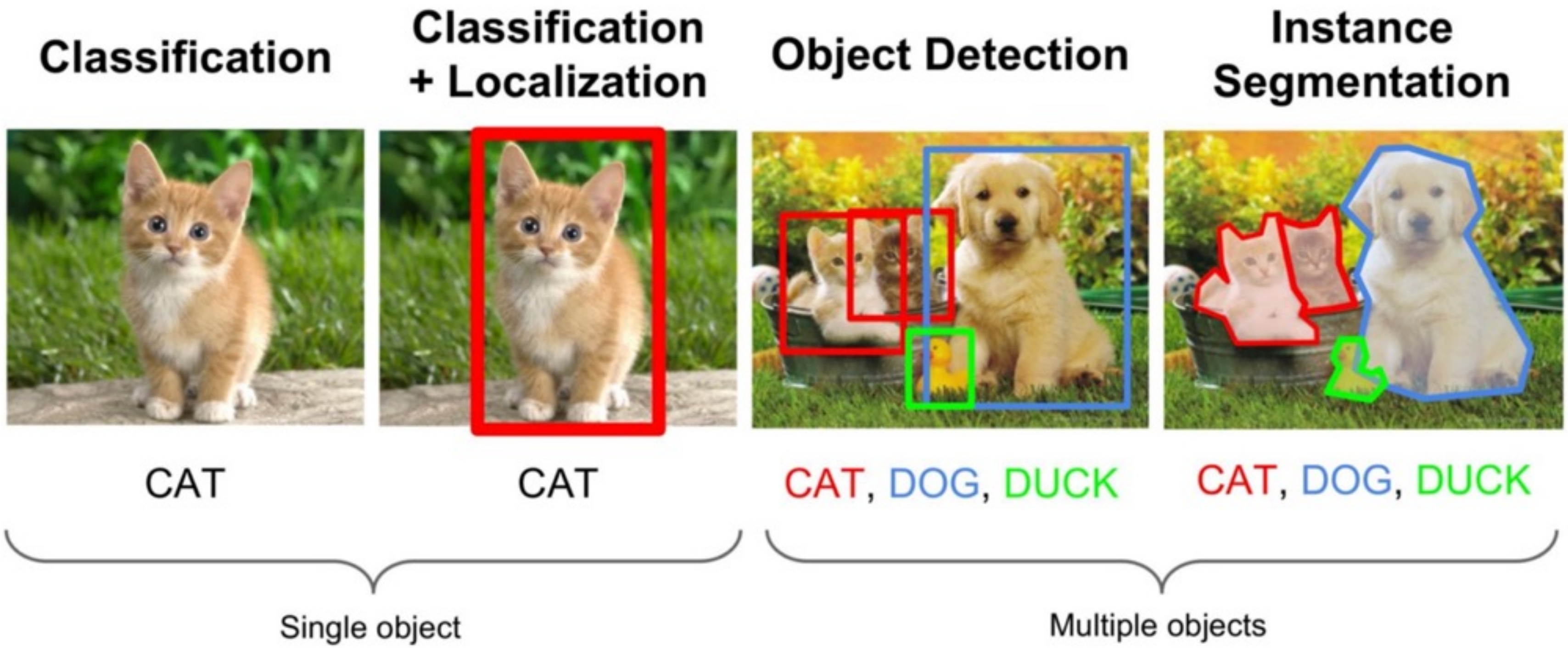



An example of a low-level algorithm

The structure of the obtained output clearly shows the spatial coherence property



Examples of high-level algorithms



UniGe | MaLGa

Image processing basics

- What is a digital image
- Spatial coherence and adjacency
- Histograms
- Operations on images

Digital Images

Size: number of pixels composing the image
Convention: rows x columns

Matrices of numbers

COLUMN

ROW

98	103	102	110	118	118	119	119	118	118	109	88
98	105	101	110	118	118	119	118	116	113	105	84
92	98	96	109	116	121	130	130	142	141	151	145
95	98	98	104	110	112	124	127	148	147	157	159
95	98	98	104	110	112	124	127	148	147	157	159
103	104	107	111	116	121	128	128	137	135	146	169
101	106	106	110	116	119	128	128	134	133	145	166
99	109	106	118	127	131	143	145	154	153	155	168
102	110	110	121	131	136	148	148	157	157	160	169
102	110	111	124	136	140	153	154	164	165	167	174
105	113	112	124	130	135	147	147	159	159	167	175
104	113	112	125	134	137	144	147	161	161	169	177
102	110	108	122	131	131	140	140	149	150	157	168
103	109	109	121	128	131	139	140	149	148	156	167
101	106	103	116	127	133	144	143	148	148	149	159
84	94	91	103	113	118	132	134	145	146	146	149
85	92	91	103	114	119	134	135	146	145	146	149
70	82	81	91	97	100	112	115	131	130	139	142
70	82	81	91	97	100	114	115	131	132	139	142
77	76	76	82	89	89	100	101	115	113	127	135
111	85	84	79	81	81	90	90	102	100	111	125
107	86	88	79	79	79	88	88	100	101	110	126

Digital images

Pixel Content

It depends on the image type

- Pictorial digital images (photos): intensity, color
- Range images: Depth information
- Medical images: Radiations absorbance level
- Termal images: Heat
- ....

Digital Images

Two important observations

- The exact relationship of a *digital image* to the *physical world* is determined by the **acquisition process**

- Any information obtained from images (shape, measurements, objects identities) is estimated starting from 2D numerical arrays.

Acquisition process

Digitalization (informally)

Let us imagine an image to be digitalized is overlaid with a regular grid.

This grid is referred to as **sampling grid.**

Each element of the grid will contain a portion (region) of the image. The whole portion will be approximated by a unique (average) value.

A coarse sampling grid produces an image with fewer details.

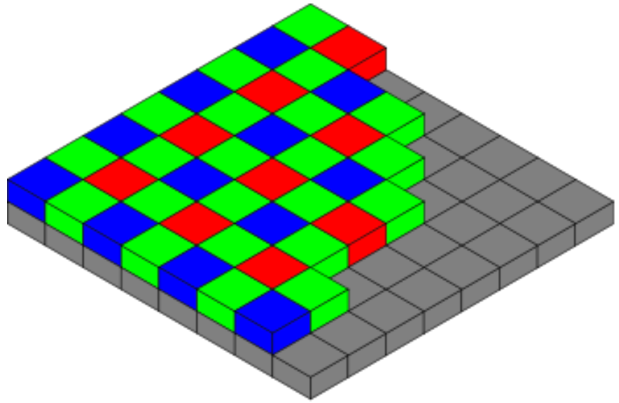
Size and resolution

- The intrinsic **size of the image** is given by the number of pixels composing it (rows x colums)
- An image with a given size (in pixels) can be visualized at different sizes (in mm) on various supports (paper, monitor, …)

- **The visualization size** is influenced by the resolution of the visualization support
- The resolution tells us how dense can be the elements on the support.
- It is measured in dots/cm or, more frequently, dots/inches (dpi).

Digital Images

Color



From the sensor we receive a stream of data which is then organized in ordered data structures.

A color image is usually acquired by using 3 filters sensitive to red green and blue. we obtain 3 monochrome images which are then combined in a single image

Here we refer to **color RGB images** where each pixel is described by a triplet (R,G,B)

a standard 24-bit image (also called *full color)* associates 1 byte per pixel to each color field (overall 3 bytes per pixel)

***Gray level** images are obtained by a (weigthed) average of color intensities*

Color spaces

RGB – Red Green Blue

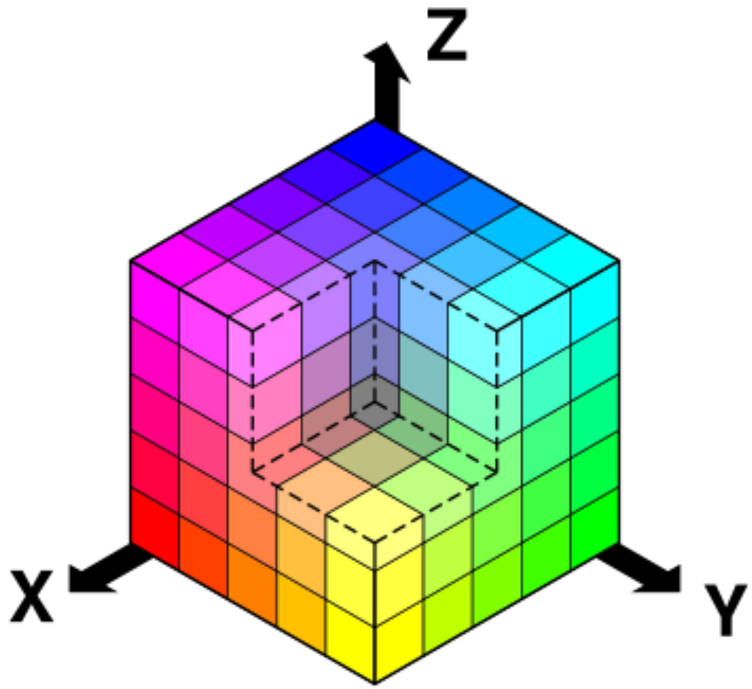
It is based on 3 primary monochrome colors:

- R red(wavelength λ=700nm)
- G green(λ=546.1nm)
- B blue(λ=435.8nm)

Most visible spectrum may be represented by blending red, green and blue lights in different proportions and intensities.

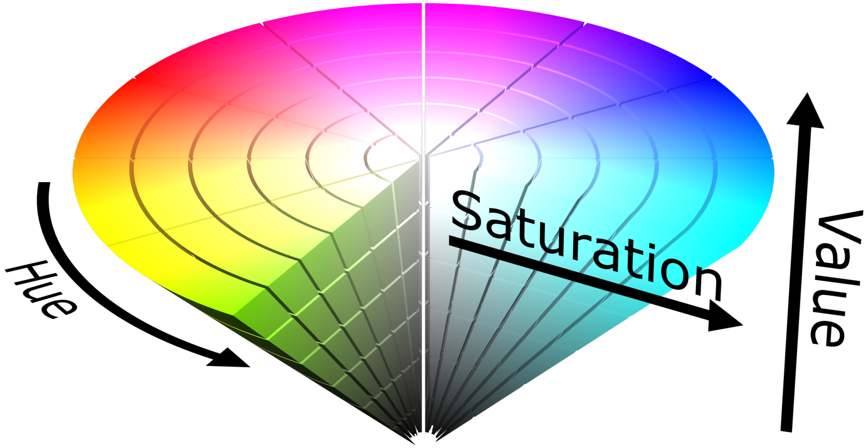
If a maximum quantity of primary colors is summed up we obtain the white

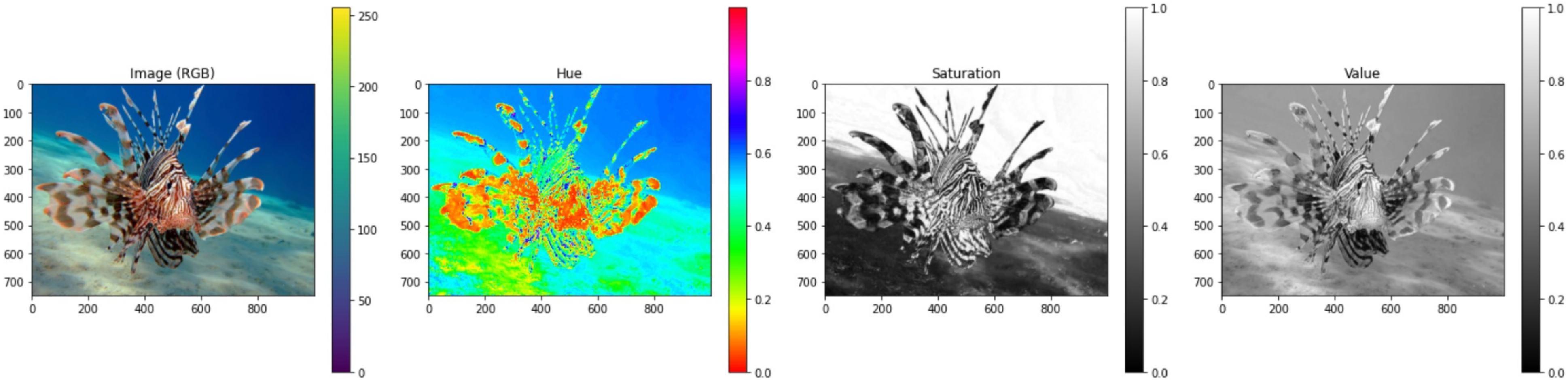
- RGB is an additive model



Perceptive color spaces

- RGB has been designed to model color for specific devices
- It does not consider our perception of color
- Other spaces are the result of studies in human perception
- An example is the HSV color space
  - H = hue
  - S = saturation
  - V = value





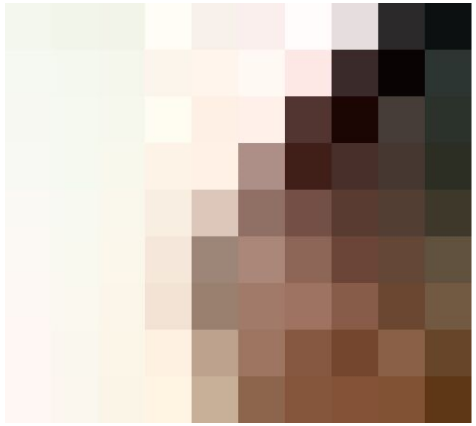
Dynamic Range

- Total number of distinctive values occurring in the image

- it is limited by the number of bit per pixel we may want to use
- it is also limited by the physical dynamic range of the sensor

- to represent black-white images 1 bit is sufficient
- gray level images: usually associate a byte to a pixel 2^8=256 gray levels
- color images: usually 1 byte per channel ("millions of color")

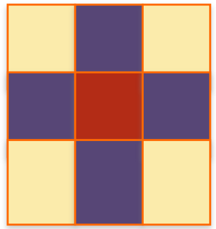
Spatial coherence and relationships among adjacent pixels



Neighbouring pixels usually carry similar information

A pixel *p* at coordinates *(i,j)* has four horizontal and vertical neighbors at coordinates *(i-1,j) (i+1,j) (i,j-1) (i,j+1)*

This set is called **4-neighborhood $N_4(p)$**



The pixel also has four diagonal neighbors: *(i-1,j-1) (i+1,j-1) (i+1,j-1) (i+1,j+1)*

The 8 points together form a **8-neighborhood $N_8(p)$**

Another way to look at the image content

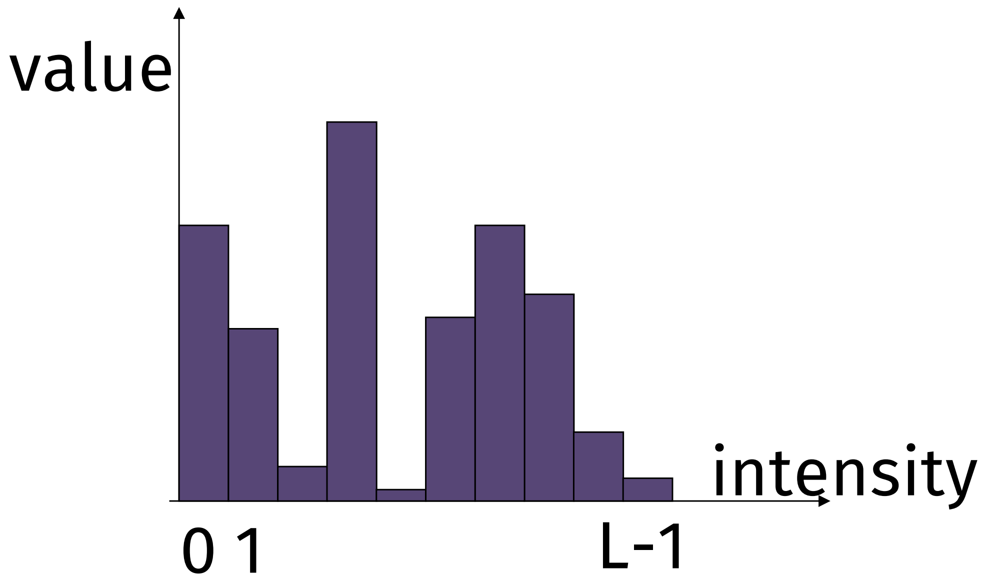
Histograms

An image histogram is a representation of the values distribution of an image

The histogram of a digital image with intensity values in the range [0,L-1] is a discrete function $h(r_k)=n_k$

where

- r_k is the k*th* intensity value of the range
- n_k is the number of pixels in the image with intensity r_k

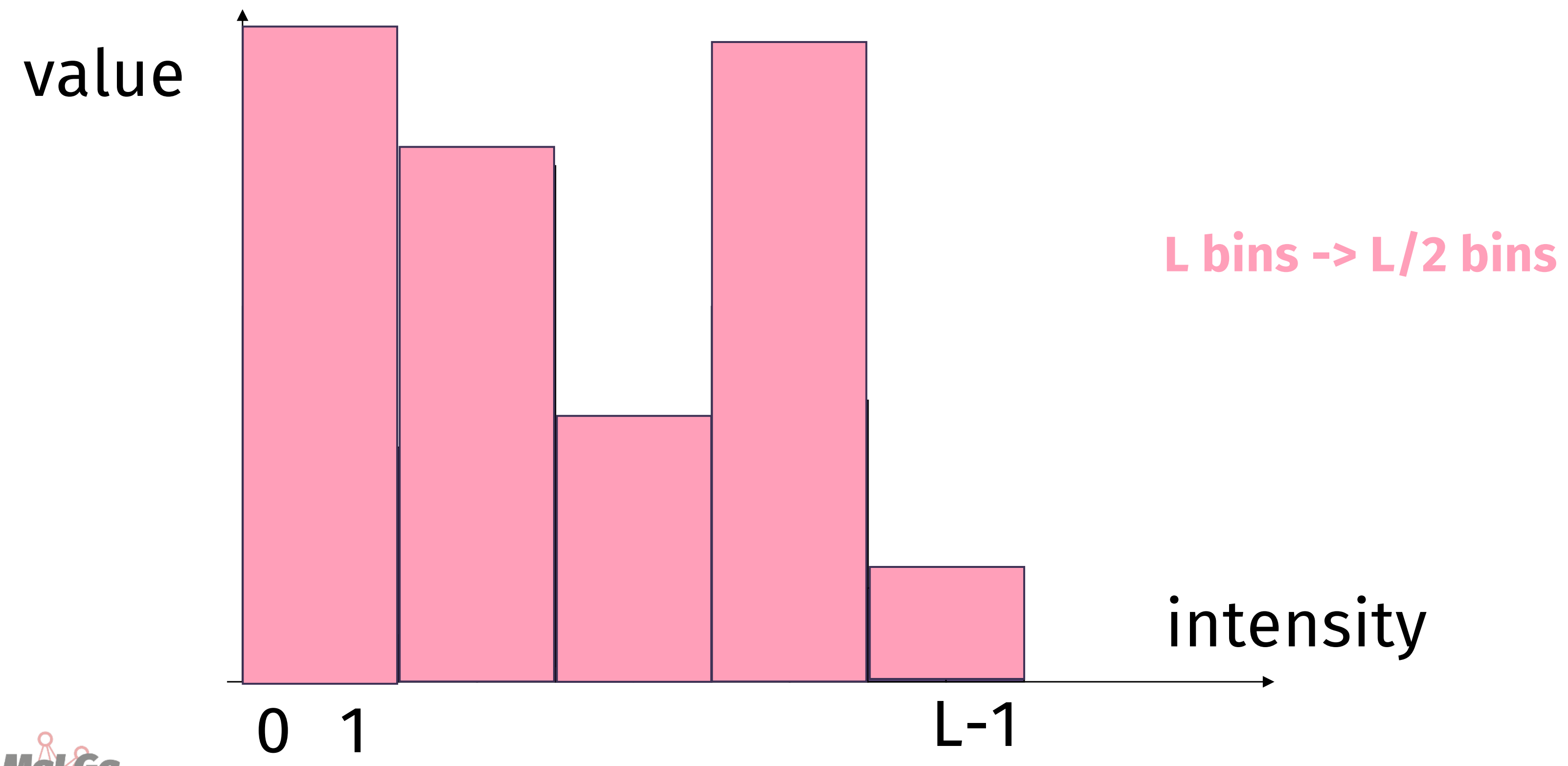


UniGe | MaLGa

HISTOGRAMS: BINS

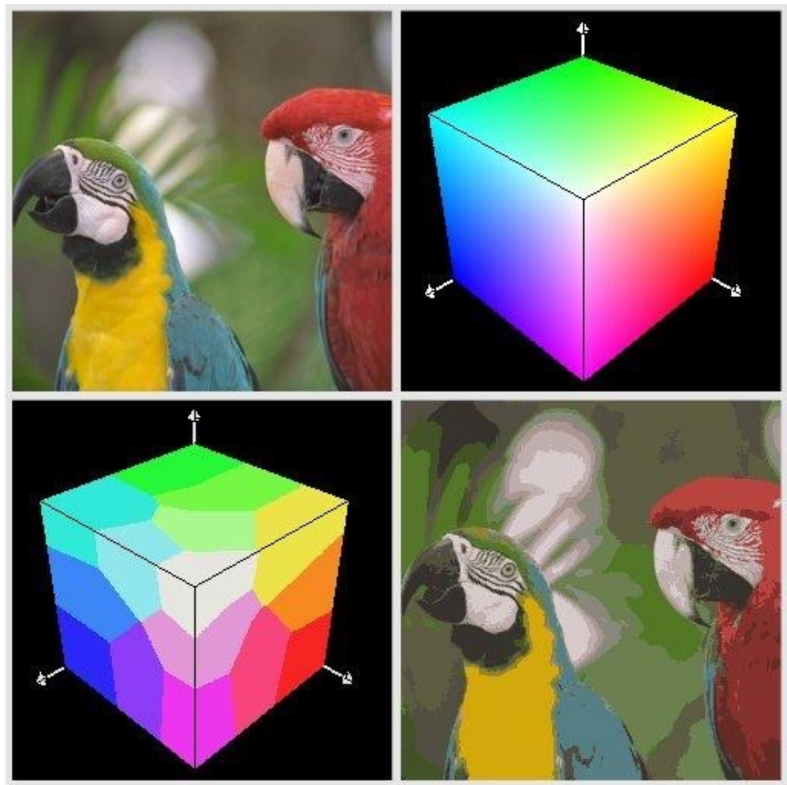
It is common practice to group similar values while computing the histogram

The range of possible values [0,L-1] can be quantized in *bins* each of which will group pixels of the image with similar values.



UniGe | MaLGa

Histogram bins & image quantization



Histogram: information loss

The corresponding histograms are identical!

Operations on images

Operations on pixels

- They produce changes in pixels content
- Alter corresponding histograms

Geometrical transformations:

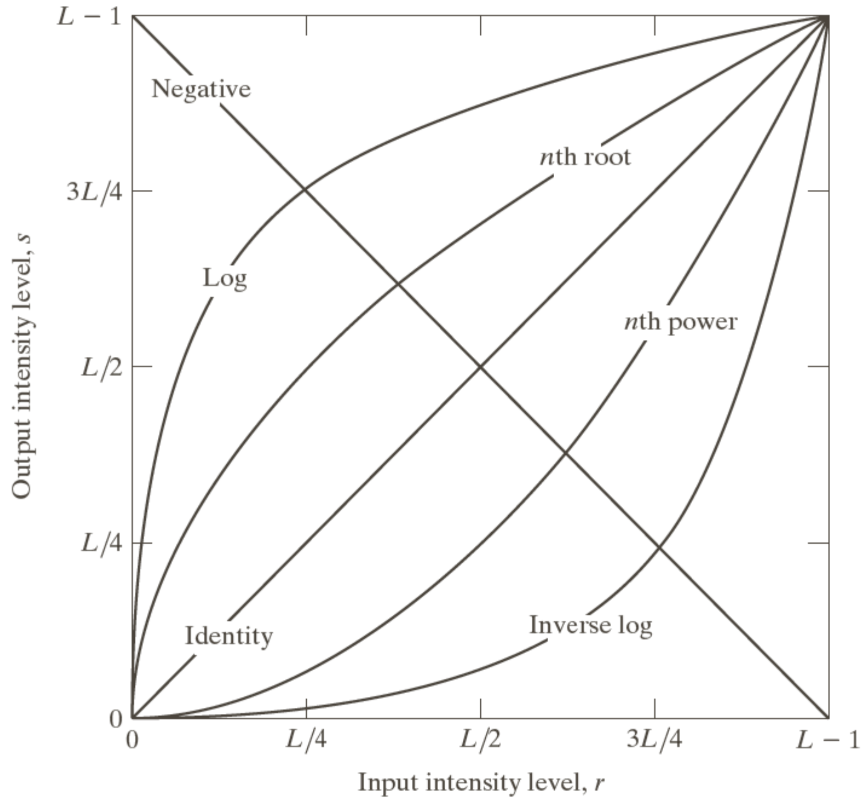
- They modify the position of pixels instead than their value
- Do not alter histograms

Intensity level transformations

Transformed output intensity level

Input pixel intensity level

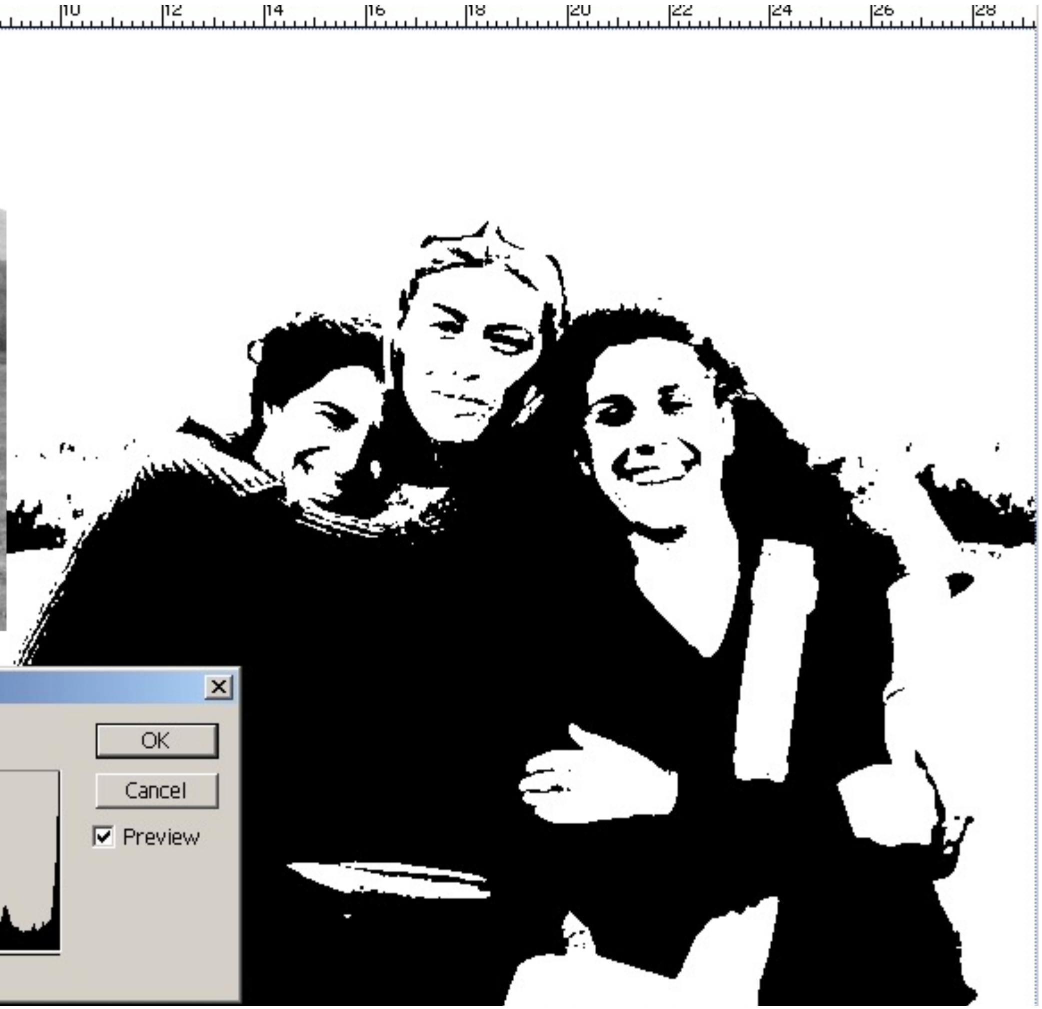
$$s = T(r)$$

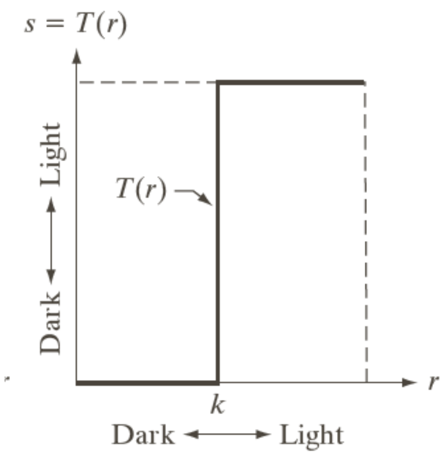


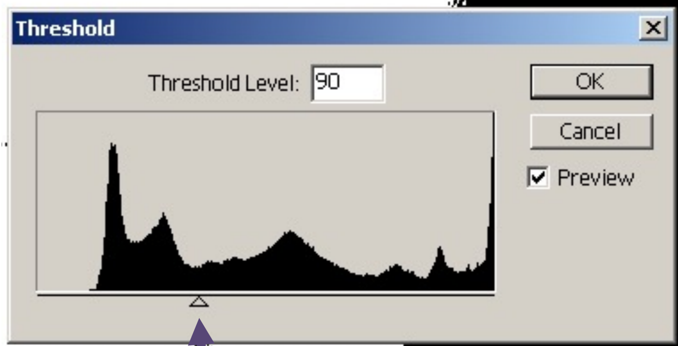
Operations on pixel values (with an effect on histograms)

Thresholding

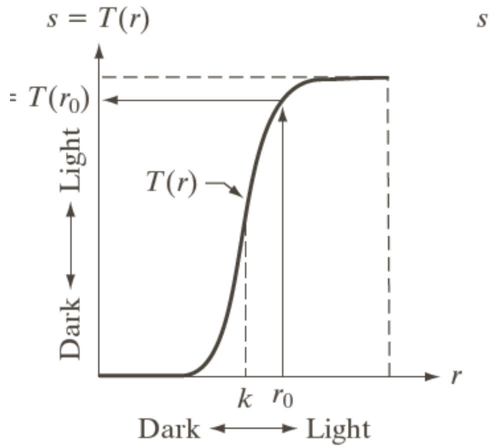
$$J(i,j) = \begin{cases} 0 & \text{if } \; I(i,j) < T \\ 1 & \text{otherwise} \end{cases}$$







Operations on pixel values (with an effect on histograms)

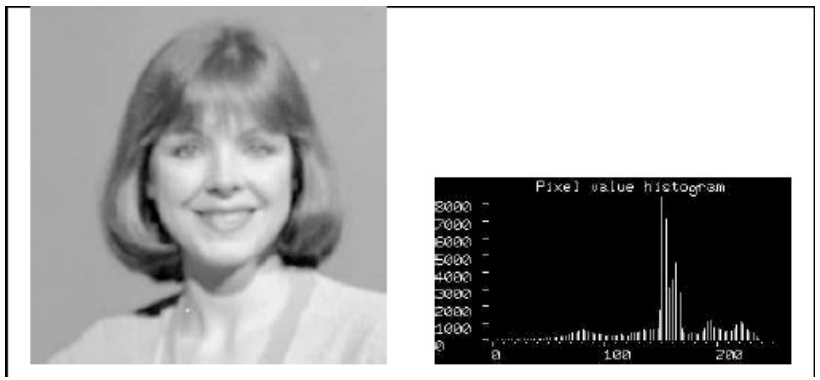


Contrast Stretch

Contrast stretch is a simple linear operation that produces an expansion of the image histogram

We choose the range of values of output image J, for instance [0,L-1].

$$J(i,j) = \frac{I(i,j) - min_I}{max_I - min_I}(L-1)$$



Geometric spatial transformations

Each point $\mathbf{p} = (x, y)$ of an input image is mapped to a point $\mathbf{q} = (u, v)$ of the output image, through the effect of a geometric transformation *T*

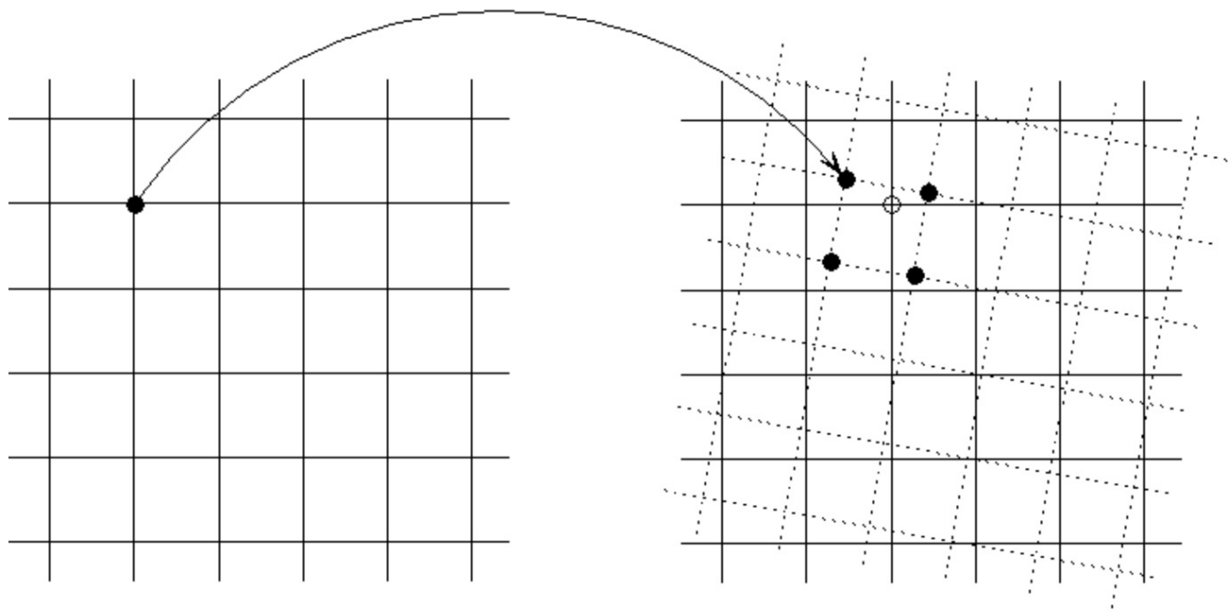
$$\mathbf{q} = T(\mathbf{p})$$

$$I_{out}(\mathbf{q}) = I_{in}(\mathbf{p})$$

Geometric spatial transformations

In **digital** image processing they consist of two steps

- A spatial transformation of coordinates from **p** to **q** according to T
- *Intensity interpolation* to assign intensity values to the spatially transformed pixels



Geometric transformation examples

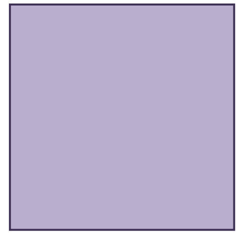


Image in-plane rotation

$$u = x\cos\theta + y\sin\theta$$
$$v = -x\sin\theta + y\cos\theta$$

$$T = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

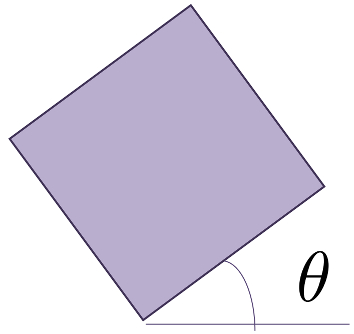
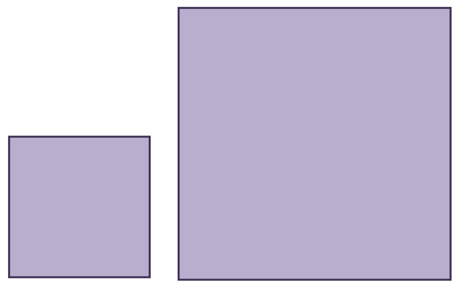


Image scaling

$$u = s_1 x$$
$$v = s_2 y$$

$$T = \begin{bmatrix} s_1 & 0 \\ 0 & s_2 \end{bmatrix}$$



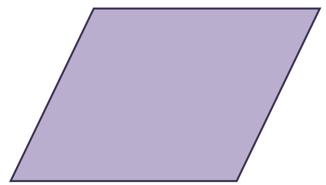
$s_1 = s_2$

$s_1 \neq s_2$

Affine transformation

$$u = ax + by + c$$
$$v = dy + ey + f$$

$$\mathbf{q} = A\mathbf{p} + \mathbf{t}$$



Direct mapping

Notice the holes

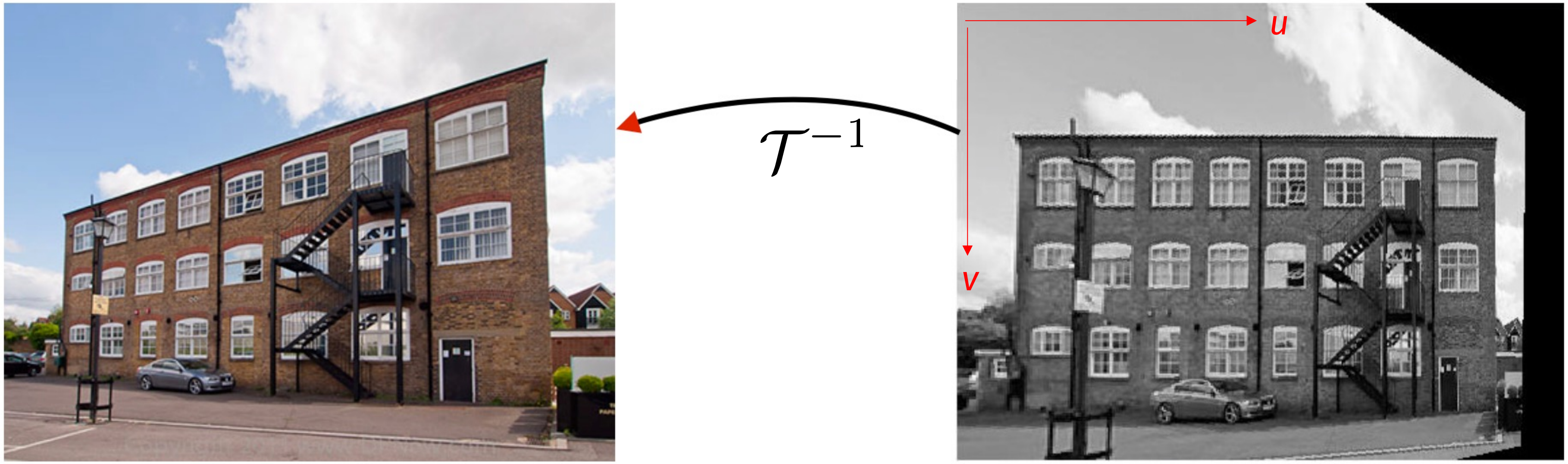
$$(u, v) = \mathcal{T}(x, y)$$

$\mathcal{T}$

$$I_{in}(x, y) \rightarrow I_{out}(u, v)$$

Inverse mapping

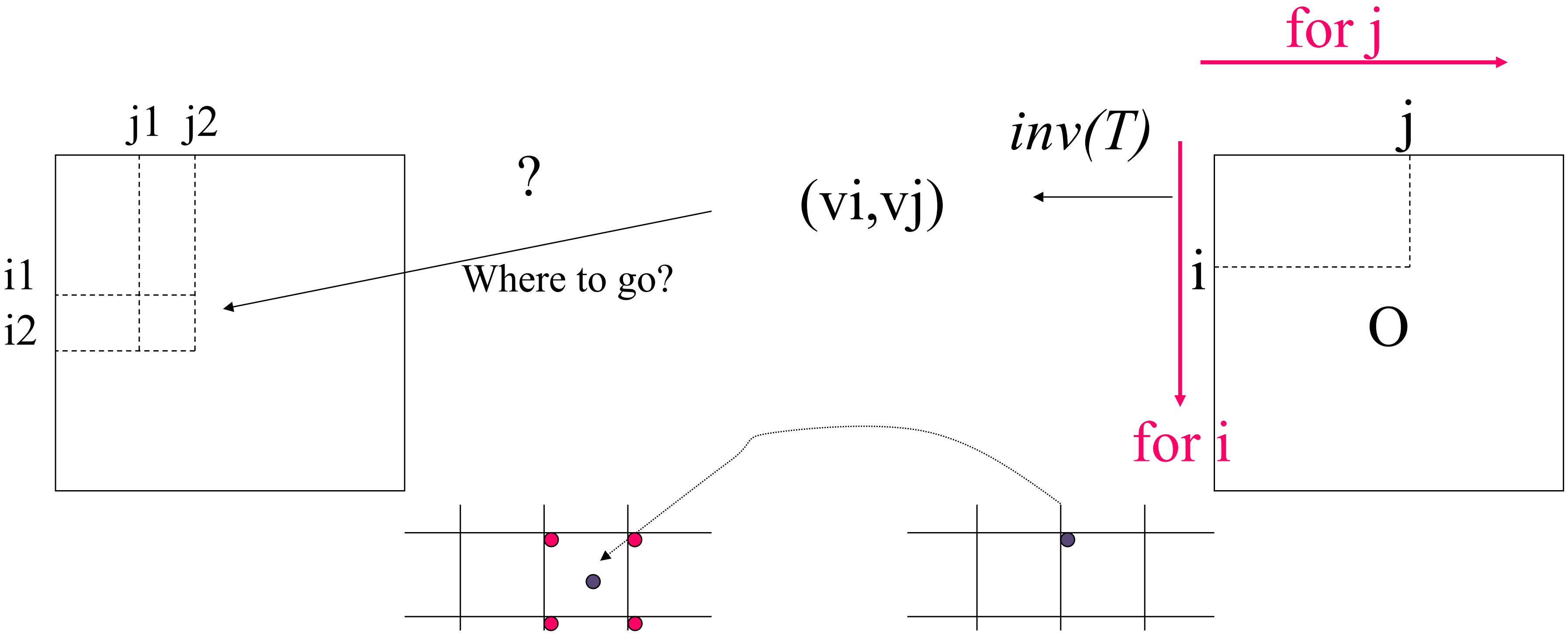
$$\mathcal{T}^{-1}(u, v) = (x, y)$$



$$I_{in}(x, y) \leftarrow I_{out}(u, v)$$

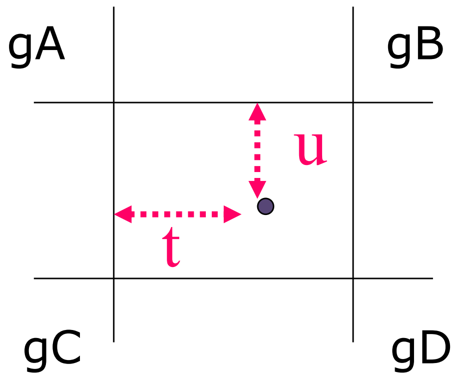
Interpolation with an inverse mapping

We start from the output image to have the guarantee that all pixels are filled



Interpolation with an inverse mapping

There are different interpolation methods, here we see the *bilinear interpolation*



g_new = (1-t)(1-u)gA + (1-t)u gB+
+ u (1-t) gC + u t gD

$t = vi - i1$

$u = vj - j1$

$$0 \leq u, t \leq 1$$

UniGe

MaLGa