Computer Vision

António J. R. Neves / Paulo Dias

Departamento de Electrónica, Telecomunicações e Informática

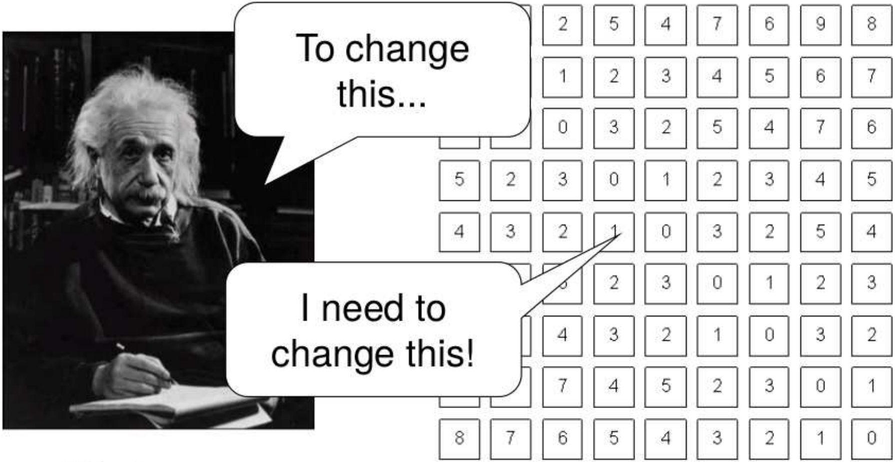
Universidade de Aveiro

an@ua.pt / paulo.dias@ua.pt

http://elearning.ua.pt/

Contents

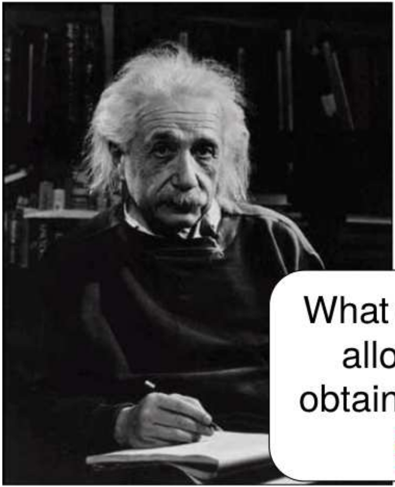
Contents



What we see

What a computer sees

Low level image processing (2)



What I see

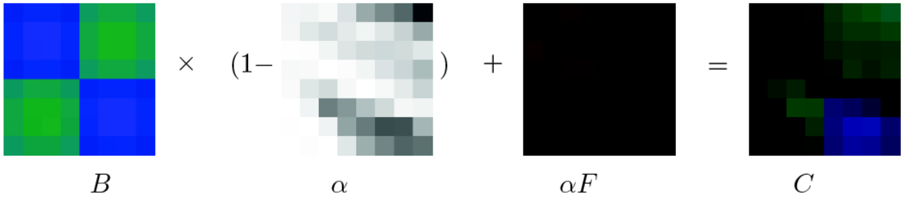
What I want to see

What operation T allows me to obtain this result?

g=T(f)

Image arithmetics

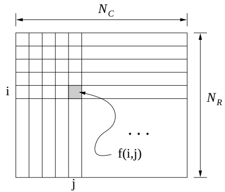
- It is possible to apply the common arithmetic operations on images:
  - Addition
  - Subtraction
  - Multiplication
  - Division
- And also logic operations on binary images (AND, OR, NOT) . . .

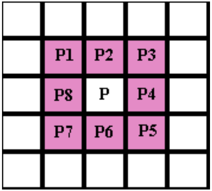


Contents

Pixel Neighbours

- Many image processing operations make use of spatial relationships between pixels.
- A number of methods have been devised to specify pixel neighbors and calculate distance.
- The 4-neighbors of a pixel (x,y) are the closest pixels in horizontal and vertical directions (D4).
- The 8-neighbors are the 4-neighbors plus the four closest pixels in diagonal direction (D8).
- Diagonal only (DN).





Pixel connections

- A group of pixels is said to be 4-connected if every pixel is 4-connected to the group.
- A group of pixels is said to be 8-connected every pixel is 8-connected to the group.

Distances

- The distance between pixels (x,y) and (u,v) can be calculated in several ways:
  - Euclidean (L2): $D = [(x-u)^2 + (y-v)^2]^{1/2}$
  - City-block (L1): $D = |x-u| + |y-v|$
  - Chessboard (Linf): $D = max(|x-u|, |y-v|)$
- Although Euclidean distance is more accurate, the sqrt makes it expensive to calculate.

Contents

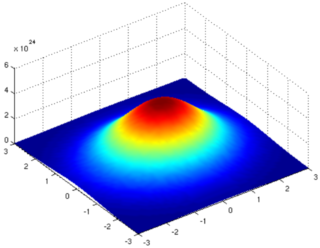
Spacial filtering

- Spatial filters make use of a fixed sized neighborhood in an input image to calculate output intensities.
- Linear filters use a weighted sum of pixels in the input image $f(i,j)$ to calculate the output pixel $g(i,j)$. In most cases, the sum of weights is one, so the output brightness = input brightness.
- Nonlinear filters can not be calculated using just a weighted sum (sqrt, log, sorting, selection).
- We can formalize the phrase “weighted sum of pixels” using correlation and convolution.
- The mathematical model is the discrete convolution operator based on the kernel h:

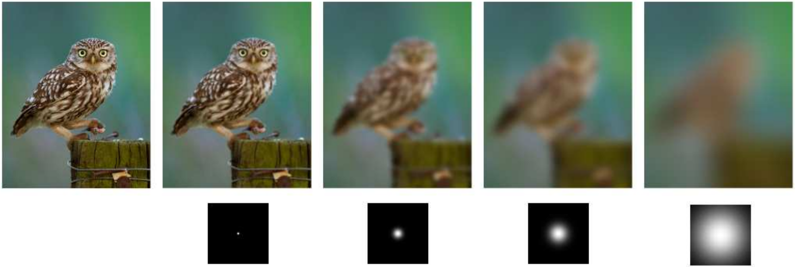
$$g(i,j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} h(i-m, j-n) f(i,j)$$

Examples of filters (1)

- Average - the easiest spatial filter to implement. The kernel is a matrix with all the values equals to one (the pixel is replaced by an average of the $N \times M$ neighbors). This filter smooths an image and removes noise and small details.
- Binomial - uses Binomial coefficients as weights to give more emphasis to pixels near the center of the $N \times M$ neighborhood.
- Gaussian - uses the Gaussian function to define the neighborhood weights.

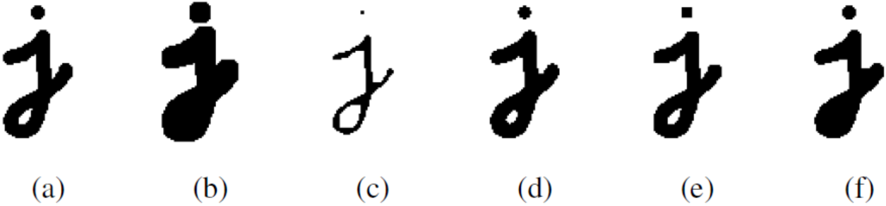


Example of filters (2)



σ = 1 pix σ = 5 pix σ = 10 pix σ = 30 pix

- The most common binary image operations are called morphological operations.
- The operation is a convolution of the binary image with a binary structuring element.
- The standard operations used in binary morphology include:



(a) original image; (b) dilation; (c) erosion; (d) majority; (e) opening; (f) closing. The structuring element for all examples is a 5×5 square.

Contents

Histograms: definition

- In statistics, a histogram is a graphical display of tabulated frequencies.
- Typically represented as a bar chart.

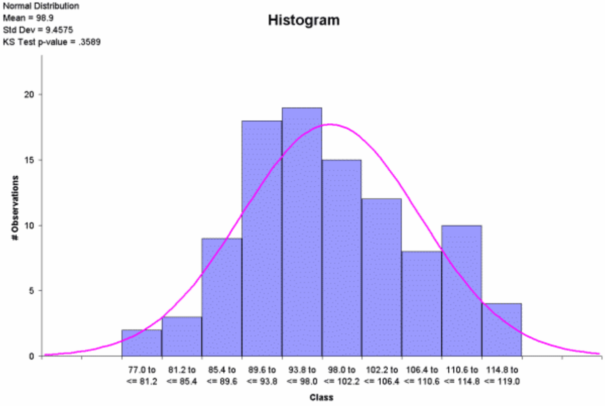
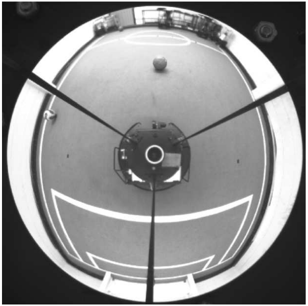


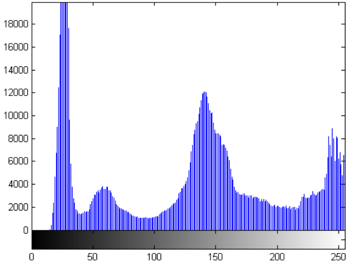
Image histograms

- In images, allow us to see the color or intensity distribution.
- The collected counts of data can be organized into a set of predefined bins.
- It is also possible to count image features that we want to measure (i.e. gradients, directions, etc).
- Some important parts of an histogram:
  - dims: The number of parameters you want to collect data.
  - bins: The number of subdivisions in each dim.
  - range: The limits for the values to be measured.
- If we want to count two features, the resulting histogram would be a 3D plot (in which x and y would be bin_x and bin_y for each feature and z would be the number of counts for each combination of (bin_x, bin_y)).

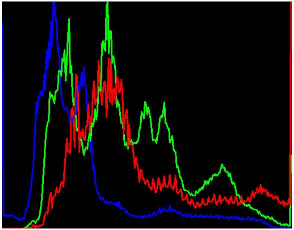
Histograms: example (1)

- Example of an histogram obtained from a grayscale image.
- Each bin shows the number of times each one of the gray values are present in the image.





- Example of an histogram showing the distribution of the colors on an image.



Histograms: operations

- Histogram operations are designed to enhance the visibility of objects of interest in an image.
- Histogram Equalization - improves the contrast in an image, in order to stretch out the intensity range.
- Local Histogram Equalization - increase the amount of enhancement by looking at local intensity properties (dividing an image into regions and perform histogram equalization on each sub-image or using local statistics).
- Histogram Comparison - get a numerical parameter that expresses how well two histograms match each other (ex. Correlation, Chi-Square, Intersection, . . .).
- Sum, subtract, . . .

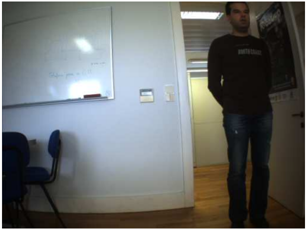
- Goal of histogram equalization is to reshape the image histogram to make it flat and wide.
- One of the solutions is to use the cumulative histogram (integral of intensity histogram) as the intensity mapping function.

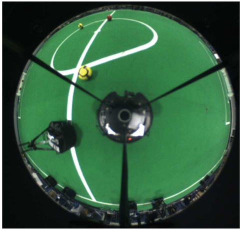
Contents

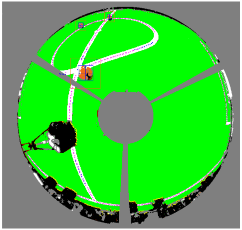
Segmentation: concept

- Intermediate processing towards object recognition.
- Localize regions with common properties.
- Make a partition over the pixel ensemble.
- Usual grouping properties (Gray level, Color, Texture).
- Often requires preprocessing.
- Segmentation of non-trivial images is a difficult task.
- Segmentation accuracy determines the eventual success/failure of computerized image analysis.

Applications of segmentation

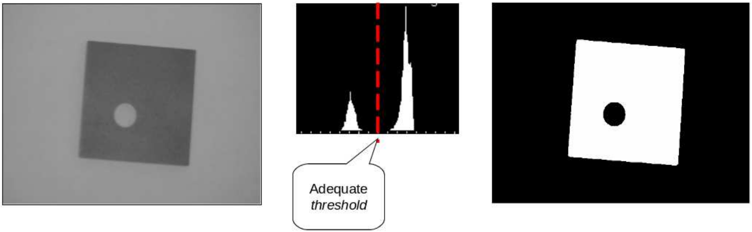




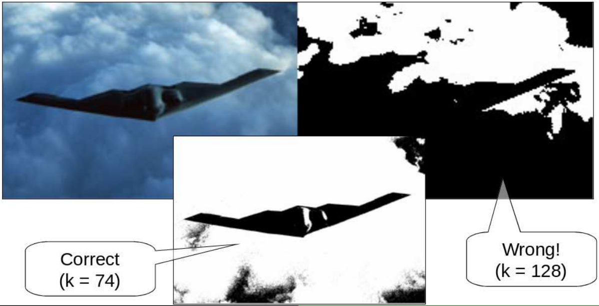


Thresholding (1)

- The basis of many region based segmentation algorithms.
- The most immediate and computationally appealing step.
- Direct image partition based on intensity properties.
  - $0 iff(x, y) \leq K$
  - $1 iff(x, y) > K$
- Not easy to find the ideal k magic number.



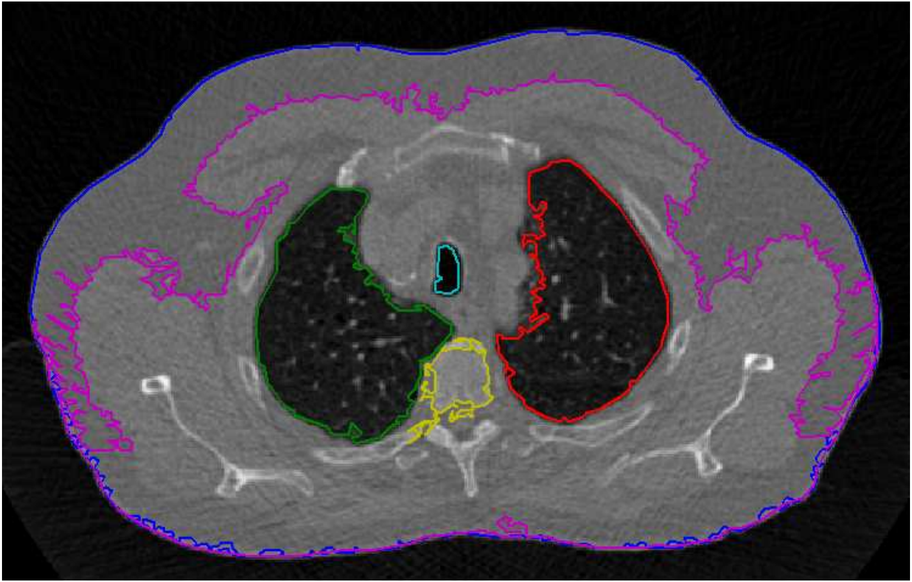
- Several approaches:
  - Global Thresholding
  - Variable Thresholding
    - Local - $T(x, y)$ depends on properties of the neighborhood of (x, y).
    - Adaptive - $T(x, y)$ depends on the spatial coordinates, x and y.
    - The Otsu's method - Optimal global thresholding based on probabilistic estimates obtained from the histogram.



Region Growing (1)

- Region growing is a procedure that groups pixels or subregions into larger regions based on a predefined criteria.
  - Start with a set of "seed" points and from these, grow regions by appending to each seed those neighboring pixels that have properties similar to the seed (intensity, color, . . .).
- Selection of seeds
  - Often interactive
  - Automated
- Centroids of pixel clusters
- Additional criteria: size and shape of region grown so far
- Stopping rules
  - Ideally, growing a region should stop when no more pixels satisfy the criteria for inclusion in that region.

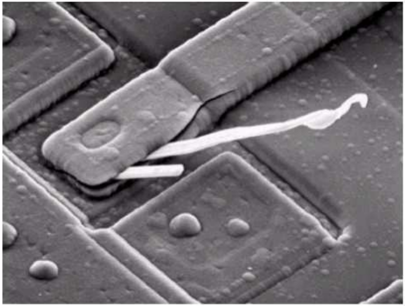
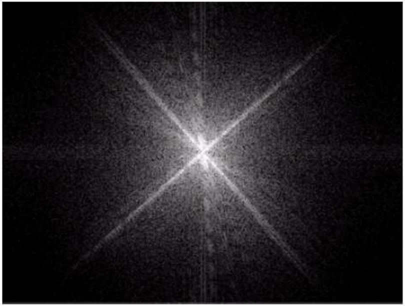
Region Growing (2)



Contents

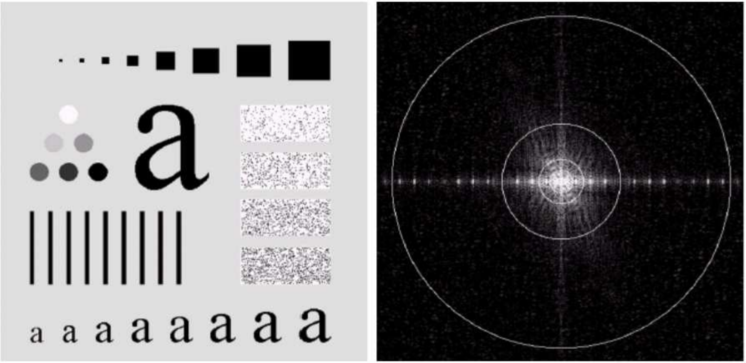
Why frequency?

- Great for filtering.
- Great for compression.
- In some situations: Much faster than operating in the spatial domain.
- Convolutions are simple multiplications in Frequency space!
- ...

On the left an image in its normal representation: $f(x, y)$ - more intuitive. On the right, the same image represented in the frequency space: $F(u, v)$.

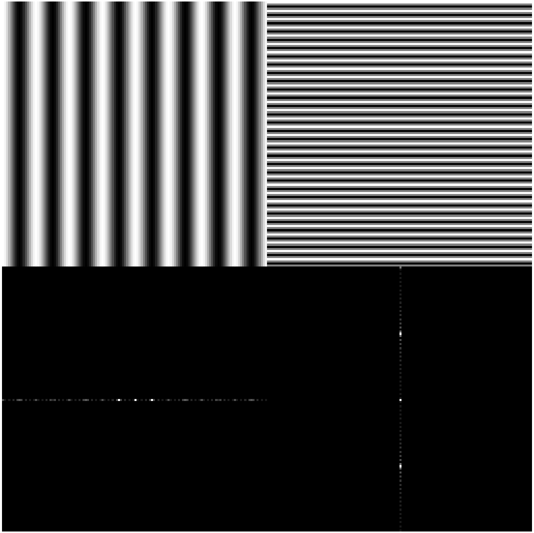
Frequency space (1)



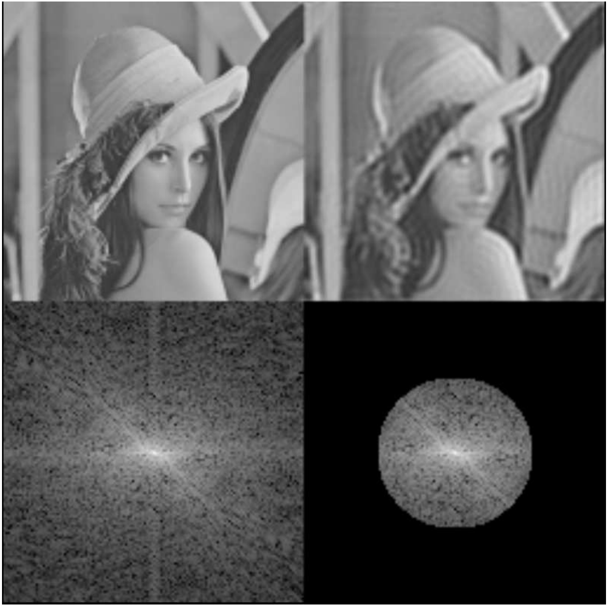
An image (500x500 pixels) and its Fourier spectrum. The super-imposed circles have radii values of 5, 15, 30, 80, and 230, which respectively enclose 92.0, 94.6, 96.4, 98.0, and 99.5% of the image power.

Horizontal and vertical frequency

- Horizontal frequencies correspond to horizontal gradients.
- Vertical frequencies correspond to vertical gradients.

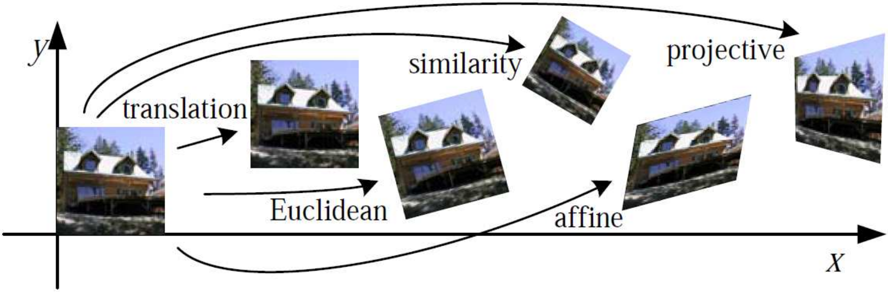


Removing frequencies

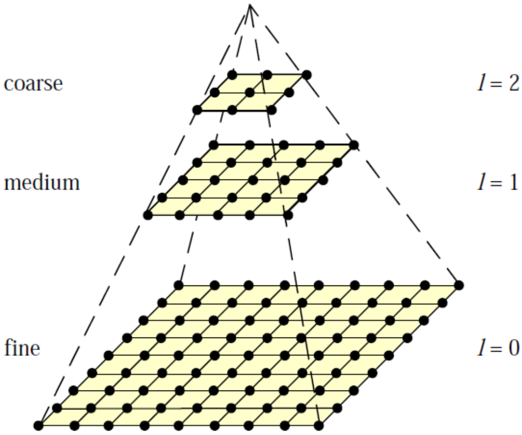


Contents

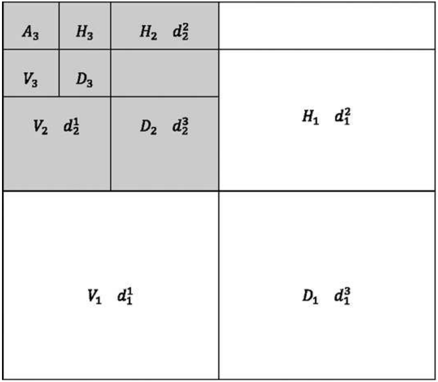
2D geometric image transformations

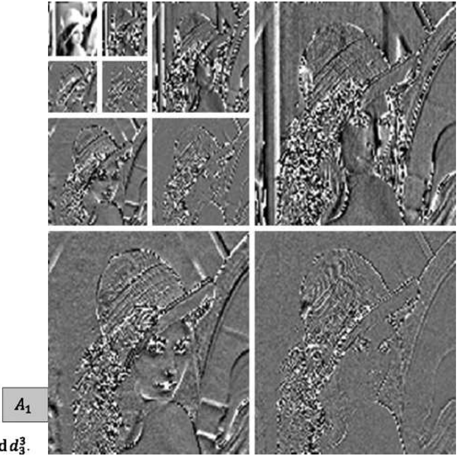


Multi-resolution representations - Pyramids



Multi-resolution representations - Wavelets





Decomposition of approximation A_1 is represented in gray:
Approximation A_2 is decomposed as :
A_3 denoted a_3, H_3 denoted d_3^2 , V_3 denoted d_3^1 and D_3 denoted d_3^3.