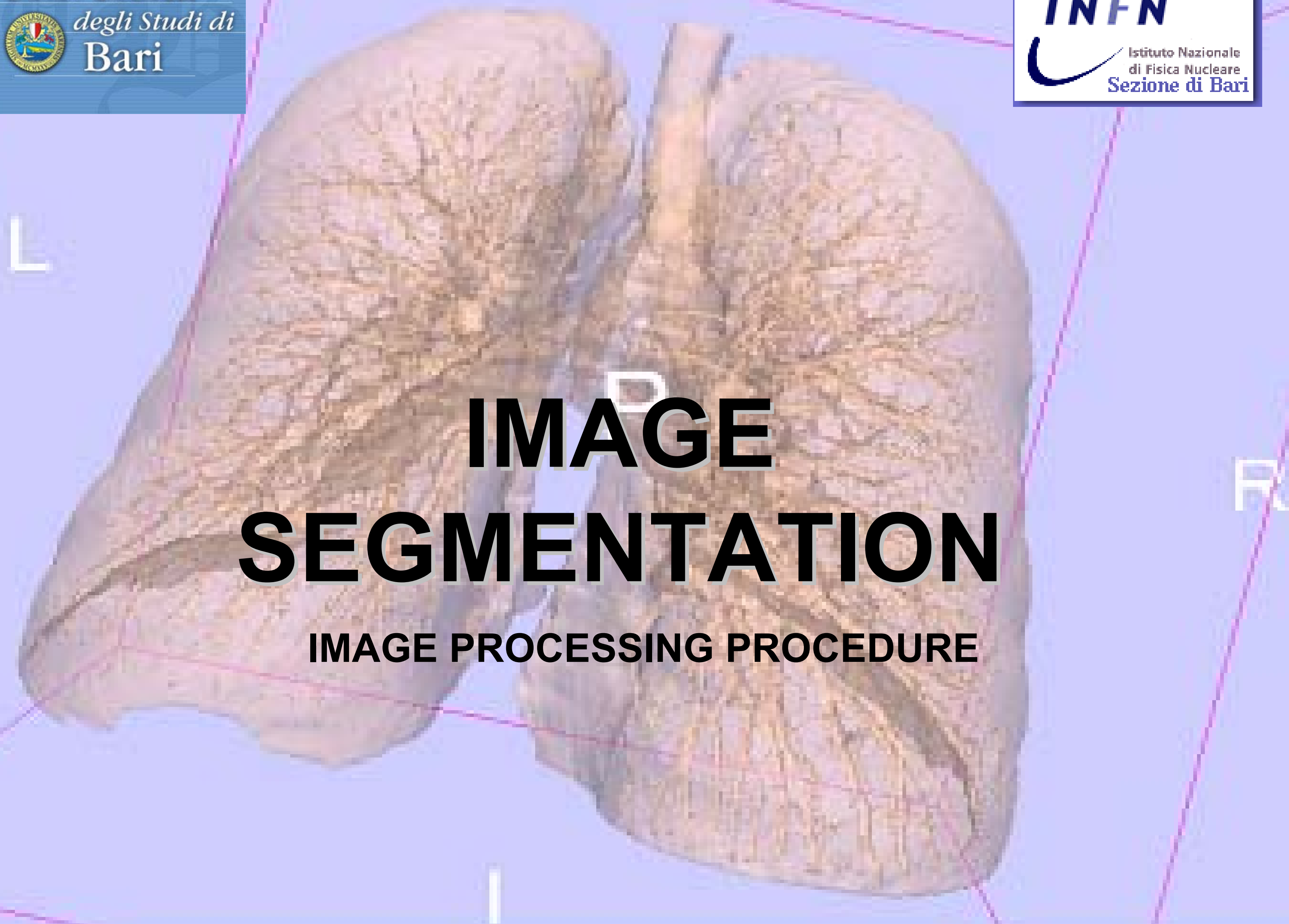


IMAGE SEGMENTATION

IMAGE PROCESSING PROCEDURE

Index

- Basic definitions
- Application fields
- Medical applications
- Different techniques
- Lung segmentation in MAGIC5-CAD
- …

Note
The following exposition regards grey level images, but it could be easily extended to colour ones.

Segmentation

Image processing method that subdivides an image into its constituent regions or objects

- **Objective**: to isolate the regions/objects of interest
- **Input**: a digital image
- **Output**: attributes extracted from the input image
- **Importance**:

  - segmentation accuracy determines the eventual   success or failure of computerized analysis procedures;

  - segmentation can be useful in many fields

Application fields

- **Electronics**:
  in automated inspection of electronic assemblies to determine the presence or absence of specific anomalies (such as missing components or broken connection paths);

- **Industrial inspection**:
  for some measures of control over the environment;

- **Military field**:
  in infrared image analysis to detect objects with strong heat signatures (such as equipment and troop in motion);

- **Astronomy**: to detect heavenly bodies;

•**Biology**:
for the control of biological and bacteriological culture;

•**Laboratory analysis:**
for the counting of the number of red corpuscles, blood platelets and different cellular masses;

•**Medicine**:
for anatomical structure and morphologic pathologies extraction from biomedical images;

•**Territory investigations**

•**Object and face recognition**

…

Standard approaches

Image segmentation algorithms are generally based on the analysis of one of two based ***proprieties of intensity values***:

- discontinuity → Partition of an image based on abrupt **changes in intensity** (identification of gray-level discontinuities such as points, lines, edges…)

- similarity → Partition of an image into regions that are **similar** according to a set of predefined criteria (region growing, thresholding, region splitting, merging…)

Gray-level discontinuity detection

- Basic types of gray-level discontinuities in a digital image:
  - points
  - lines
  - edges
- Main approach for their identification: to run a *mask* through the images, computing the sum of products of the coefficients with the gray levels contained in the region encompassed by the mask.

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

a general 3x3 mask

$$R = w_1 z_1 + w_2 z_2 + \cdots + w_9 z_9 = \sum_{i=1}^{9} w_i z_i$$

(*)Response of the mask at any point in the image,
defined with respect to its center location.

z_i= gray level of the pixel
ω_i =associated mask coefficient

Isolate points identification

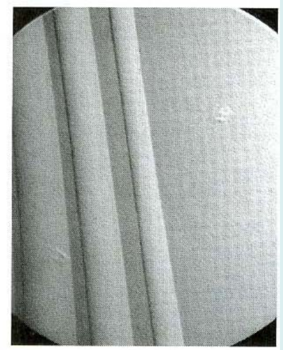
isolate point = point whose gray level is significantly different from its background and which is located in an homogeneous or nearly homogeneous area

Its identification is based on the measure of weighted differences between the center point and its neighbours.

If it is quite different from its surroundings, it will be easily detectable by an appropriate mask.

There are approaches based on single-pixel discontinuities in homogeneous background.
When this condition is not satisfied, other procedures are more suitable for the same objectives.

Es



X-ray image of a jet-engine turbine blade with a porosity in the upper
There is a single black pixel embedded within the porosity

Result of point detection

Result of the application of equation (*)

Line detection

For detecting all the lines of an image *<u>in a specified direction</u>*, the main approach is using a mask associated with the direction and thresholding its outputs

Moving the mask around the image, ***it would respond more strongly to lines (one pixel thick) oriented in a definite direction.***
For example, in an image with a quite constant background, with the mask.1 the maximum response would result when the line passed through the middle row of the mask (lines oriented horizontally).

−1	−1	−1
2	2	2
−1	−1	−1

mask.1

Similarly the mask.2 responds best to lines oriented at +45°
the mask.3 responds best to vertical lines
the mask.2 responds best to lines in -45° direction

−1	−1	2
−1	2	−1
2	−1	−1

mask.2

−1	2	−1
−1	2	−1
−1	2	−1

mask.3

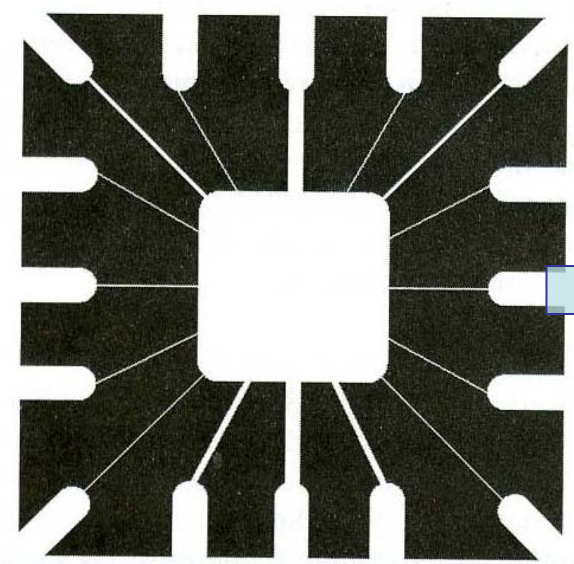
2	−1	−1
−1	2	−1
−1	−1	2

mask.4

The desire direction can be established choosing a mask where the preferred direction is weighted with a larger coefficient than other possible directions.

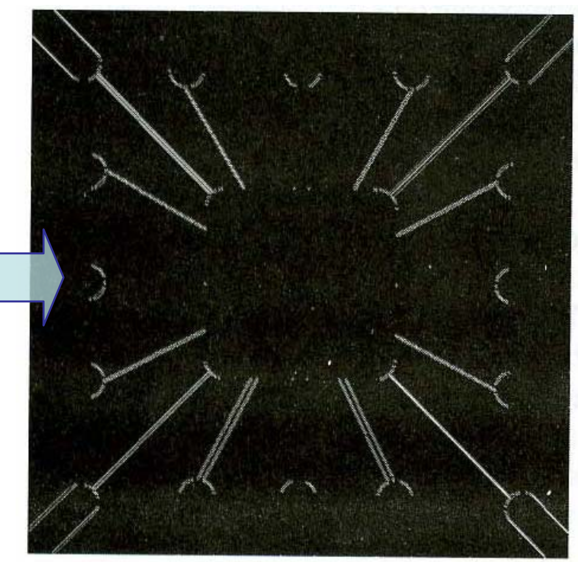
Note: the coefficients in each mask sum to zero: a zero response from the masks in areas of constant gray level.

Line detection: an example



2	−1	−1
−1	2	−1
−1	−1	2

−45°



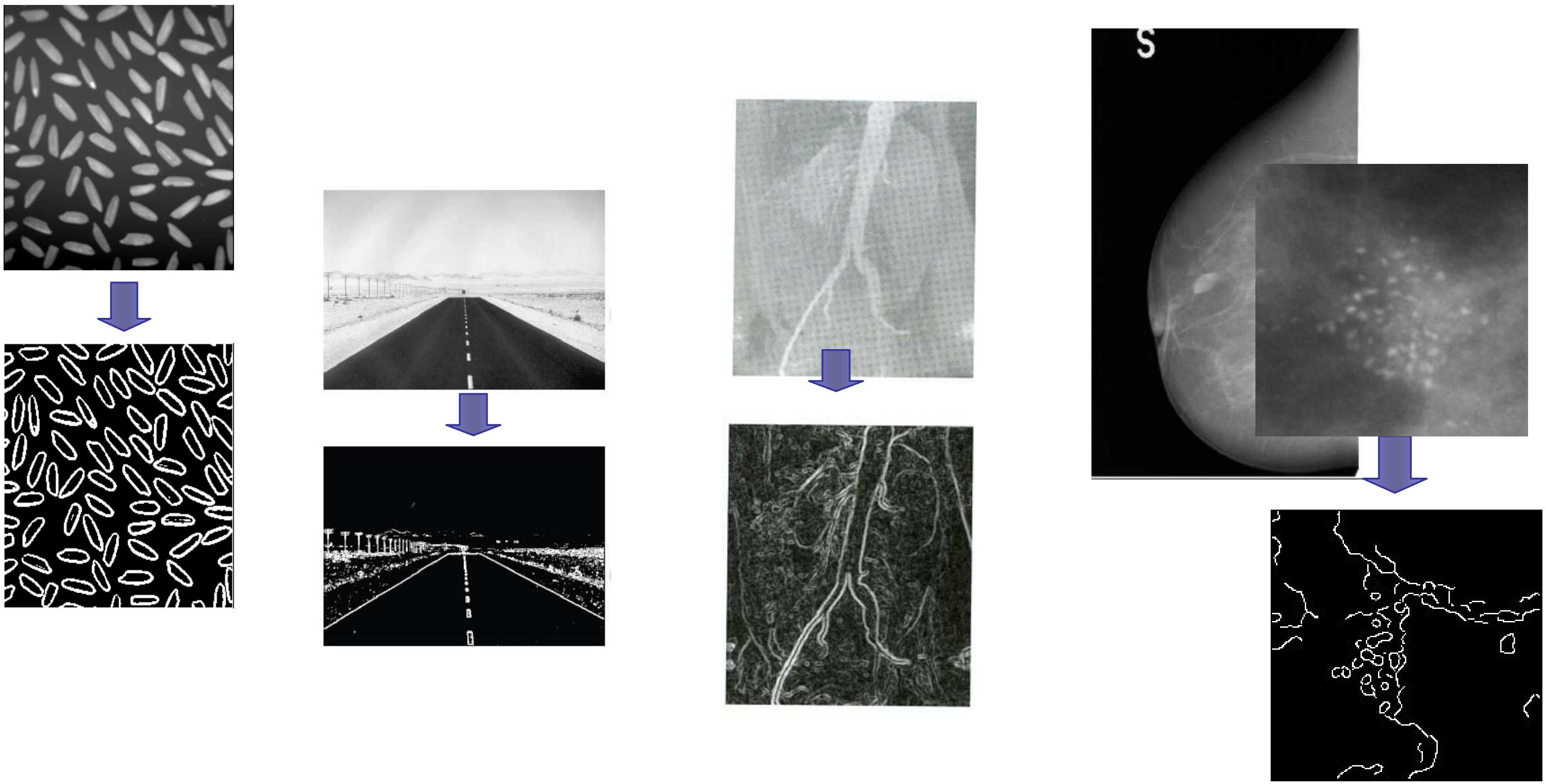
A digitized (binary) portion of a wire-bond mask for an electronic circuit

The absolute value of the result:
all the vertical and horizontal components are eliminated;
the components that toward a -45° direction produced the strongest responses.

If we are interested in finding all the lines that are one pixel thick and are oriented at -45°, we can use the mask.4

Edge detection

Edge detection has many applications in different field: it is a very important procedure of the image processing.



Examples of edge detection in different field

What is an edge?

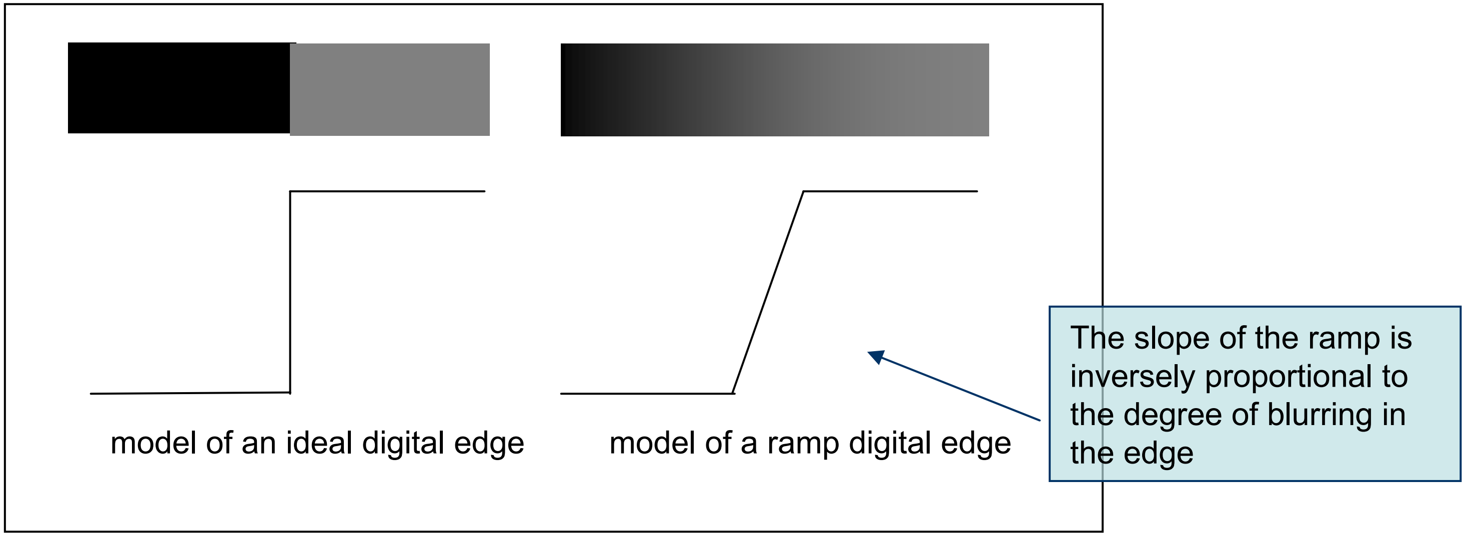
- ***Intuitively***, **an edge is a set of connected pixels that lie on the boundary between two regions.**
- An edge (“local” concept) is different from a boundary (“global” idea)
- **An *ideal edge* is a set of connected pixels (ex. in the vertical direction) each of which is located at an orthogonal step transition in gray level (ex. horizontal profile in figure)**
- ***Real edges* have a “ramplike” profile, due of different possible factors:**

  - quality of the image acquisition system;

  - the sampling rate;

  - illumination conditions under which the image is acquired…

Gray-level profile of a horizontal line through the image



•**Edge** as *a set of the connected points contained in the ramp*
(the entire transition from black to white is a single edge)

•**Its "thickness"** as *the length of the ramp*, as its transitions from an initial to a final gray- level.

It is determined by the slope, which, in turn, is determined by the degree of theblurring, so that:
•blurred edges tend to be thick,
•sharp edges tend to be thin.

Implementing first- and second- derivatives for edge detection

The study of the profile derivatives is a fundamental instrument for edge detection:

First derivative →

its magnitude can be used to detect the presence of an edge at a point of the image (i.e. to determine if a point is on a ramp)

A point in an image is an *edge point* if its two-dimensional first-order derivative is greater than a specified threshold. An *edge* is a set of these points satisfied a predefined criterion of connectedness

Second derivative →

its sign can be used to determine whether an edge pixel lies on the dark or light side of an edge.

Additional proprieties of the second derivative around an edge can be used for different purposes.

For example we note that :
- the 2-nd derivative produces two values for every edge.
- an imaginary straight line joining the extreme positive and negative values of the second derivative would cross zero near the midpoint of the edge.

This *zero-crossing* properly is useful for locating the centers of thick edges

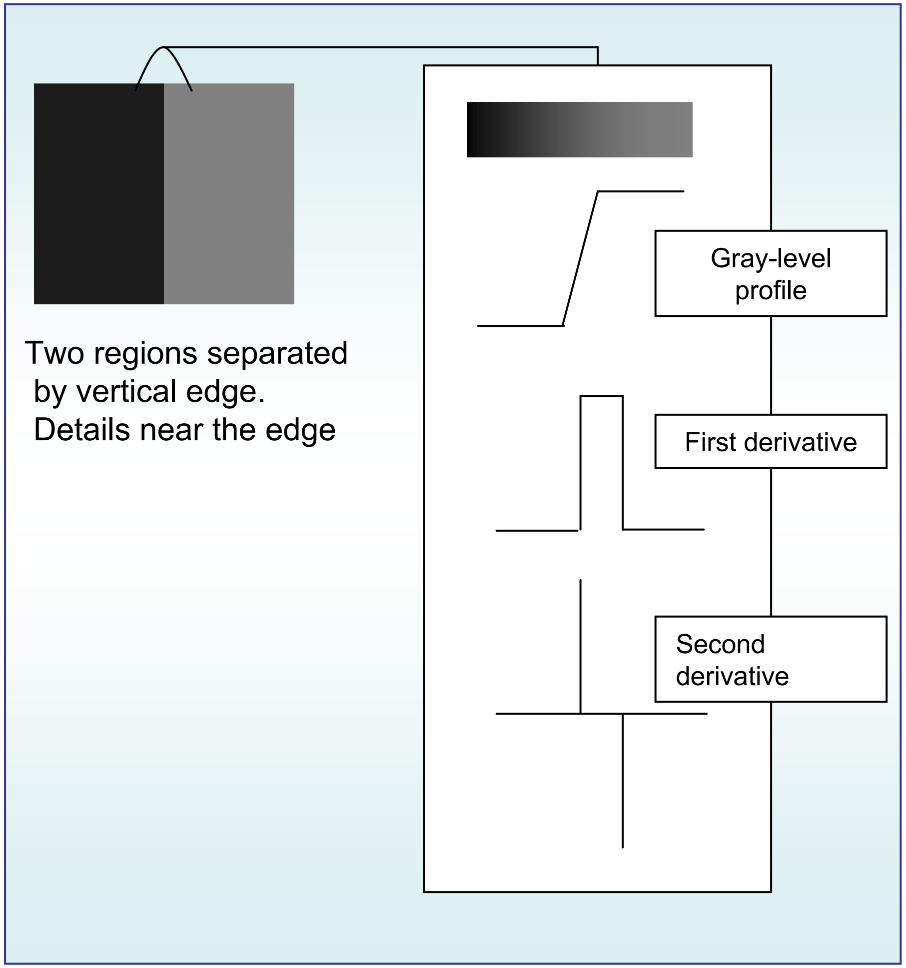
Example

The figure shows:

- **a horizontal gray-level profile of an edge beetween two regions**

- **the first derivative of the profile**
  It is:
  - *positive* at the points of transition into and out of the ramp as we move from left to right along the profile
  - *constant* for points in the ramp
  - *zero* in areas of constant gray-level

- **the second derivative of the profile**
  It is:
  - positive at the transition associated with the dark side of the edge
  - negative at the transition associated with the light side of the edge
  - zero long the ramp and in areas of constant gray level



The signs of derivatives would be reversed for an edge that transitions from light to dark

ELABORAZIONI LOCALI SPAZIALI

- Estrazione dei contorni

Elaborazioni locali spaziali

Sommario

- Introduzione
- Differenza dei livelli di grigio
- Operatori locali spaziali di estrazione dei contorni
- Operatori locali basati su maschere

Introduzione

Un'altra operazione locale spaziale è rappresentata dalla determinazione delle differenze fra i livelli di grigio adiacenti

Introduzione

Questa operazione di tipo passa-alto (esaltazione delle frequenze spaziali piu' alte) è utile per esaltare le *variazioni* e i *contrasti*, rinforzando i bordi e i contorni

Differenza dei livelli di grigio

Un primo procedimento corrisponde a spostare l'immagine originaria di una riga e ad effettuare la differenza fra i pixel corrispondenti dell'immagine originaria e di quella spostata

Differenza dei livelli di grigio

Indicando con d_i la differenza, possono essere poi adottate differenti decisioni.

Con: $d_i = 0$

sostituzione di un livello di grigio intermedio (livello a metà scala) al pixel corrispondente nell'immagine originaria

Differenza dei livelli di grigio

Con: $d_i > 0$

sostituzione di un livello di grigio elevato (caso limite fondo scala, ad esempio bianco) al pixel corrispondente nell'immagine originaria

Differenza dei livelli di grigio

Con : $d_i < 0$

sostituzione di un livello di grigio piccolo (caso limite inizio scala, ad esempio nero) al pixel corrispondente nell'immagine originaria

Differenza dei livelli di grigio

E' evidente che con la decisione precedente si ottiene un'immagine in cui sono rinforzate le transizioni da un livello ad un altro di grigio (ad esempio con un fondo grigio e linee bianche e nere intorno ai contorni degli oggetti)

Differenza dei livelli di grigio

Le operazioni precedenti di differenza corrispondono come già osservato a filtraggi di *tipo passa-alto* nel campo delle frequenze spaziali, con il risultato di esaltazione dei bordi e contorni e loro estrazione piu' o meno netta dal contesto dell'immagine

Elaborazioni locali spaziali

Sommario

- Introduzione
- Differenza dei livelli di grigio
- Operatori locali spaziali di estrazione dei contorni
- Operatori locali basati su maschere

Operatori locali spaziali di estrazione dei contorni

Si è già più volte indicato l'interesse notevole per l'estrazione dei contorni o bordi dall'immagine in esame, al fine di ottenere gli elementi strutturali fondamentali dell'immagine stessa (figure, oggetti)

Operatori locali spaziali di estrazione dei contorni

Diverse elaborazioni già indicate permettono di ottenere esaltazione dei contrasti e dei contorni

Operatori locali spaziali di estrazione dei contorni

In particolare si è indicato come le differenze fra i livelli di grigio forniscano indicazioni utili sui contorni. In effetti tali differenze rappresentano misure o stime del *gradiente* dell'immagine (*gradiente* del livello di grigio) nei vari punti analizzati

Operatori locali spaziali di estrazione dei contorni

La maggior parte degli operatori locali spaziali per l'estrazione dei contorni si basa appunto sulla determinazione o stima del *gradiente*

Operatori locali spaziali di estrazione dei contorni

Una volta che sia valutato il gradiente in ampiezza (*modulo*) e direzione (*angolo*) in un punto, viene confrontata la sua ampiezza con un certo valore di soglia

Operatori locali spaziali di estrazione dei contorni

Se essa è maggiore del valore di soglia, si assume in genere che il punto considerato fa parte di un contorno, la cui direzione è ortogonale o normale alla direzione del gradiente

Operatori locali spaziali di estrazione dei contorni

Un primo metodo per stimare il valore del gradiente consiste nel valutare le due componenti ortogonali del gradiente nel punto in esame, D_x secondo x e D_y secondo y

Operatori locali spaziali di estrazione dei contorni

Si ottiene quindi il suo modulo:

$$D = \sqrt{D_x^2 + D_y^2}$$

e la sua direzione:

$$\varphi = \operatorname{arctg} \frac{D_y}{D_x}$$

Operatori locali spaziali di estrazione dei contorni

Le due componenti D_x e D_y possono essere stimate in molti modi. Se si considera un punto (pixel) di coordinate (n_1,n_2) e livello di grigio pari a $f(n_1,n_2)$, il modo più semplice per ottenere le componenti D_x e D_y è quello delle differenze con i livelli dei punti vicini

Operatori locali spaziali di estrazione dei contorni

$$D_x = f(n_1, n_2 + 1) - f(n_1, n_2)$$

$$D_y = f(n_1, n_2) - f(n_1 + 1, n_2)$$

dove $f(n_1,n_2+1)$ è il livello del punto a destra di quello in esame e $f(n_1+1,n_2)$ è il livello del punto al di sotto di quello in esame

Operatori locali spaziali di estrazione dei contorni

Le operazioni precedenti corrispondono a usare le cosiddette *maschere* indicate di seguito (la maschera indica in sostanza, con i suoi numeri, il *peso* o moltiplicatore da applicare al corrispondente livello di grigio, con il relativo segno di somma o sottrazione)

Operatori locali spaziali di estrazione dei contorni

Così abbiamo:

$$D_x = \begin{vmatrix} -1 & 1 \\ 0 & 0 \end{vmatrix} \qquad D_y = \begin{vmatrix} 1 & 0 \\ -1 & 0 \end{vmatrix}$$

Operatori locali spaziali di estrazione dei contorni

Un modo equivalente per stimare le due componenti del gradiente è il metodo di Roberts, le cui relazioni sono:

$$D_1 = f(n_1, n_2 + 1) - f(n_1 + 1, n_2)$$

$$D_2 = f(n_1, n_2) - f(n_1 + 1, n_2 + 1)$$

Operatori locali spaziali di estrazione dei contorni

Le corrispondenti maschere risultano:

$$D_1 = \begin{vmatrix} 0 & 1 \\ -1 & 0 \end{vmatrix} \qquad D_2 = \begin{vmatrix} 1 & 0 \\ 0 & -1 \end{vmatrix}$$

che fornisce due componenti ortogonali, ruotate di un angolo pari a $\pi/4$ rispetto agli assi (x,y) dell'immagine

Operatori locali spaziali di estrazione dei contorni

Si può osservare che, per accentuare i contorni, si possono effettuare anche le differenze delle differenze; ad esempio, per quanto riguarda contorni orizzontali, si ha:

$$
\begin{aligned}
D_x^{'} &= f(n_1, n_2) - f(n_1, n_2 - 1) - \\
&- \left[f(n_1, n_2 + 1) - f(n_1, n_2) \right] = \\
&= 2 f(n_1, n_2) - f(n_1, n_2 - 1) - f(n_1, n_2 + 1)
\end{aligned}
$$

Operatori locali spaziali di estrazione dei contorni

La valutazione del gradiente è più accurata, se si considerano matrici di 3x3 elementi nell'intorno del punto in cui si valuta il gradiente. Sono noti molti metodi, che assegnano pesi diversi ai valori vicini al punto centrale

Operatori locali spaziali di estrazione dei contorni

Alcune delle maschere utilizzate sono indicate di seguito:

Gradiente "*smoothed*"

$$D_x = \begin{vmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{vmatrix} \qquad D_y = \begin{vmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{vmatrix}$$

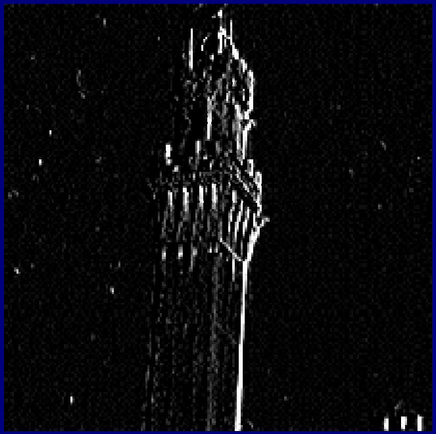
Operatori locali spaziali di estrazione dei contorni

Gradiente di Sobel

$$D_x = \begin{vmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{vmatrix} \quad D_y = \begin{vmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{vmatrix}$$

Operatori locali spaziali di estrazione dei contorni

Applicazione del gradiente di Sobel sulla Torre del Mangia, Siena



Operatori locali spaziali di estrazione dei contorni

Applicazione del gradiente di Sobel
sul volto della Venere

Sandro Filipepi, detto il Botticelli, *La nascita di Venere,* Firenze, Galleria degli Uffizi, particolare

Operatori locali spaziali di estrazione dei contorni

Applicazione del gradiente di Sobel
sull' Abbazia di S. Galgano

Image corrupted by random Gaussian noise

Sensitivity of derivatives to noise: an example

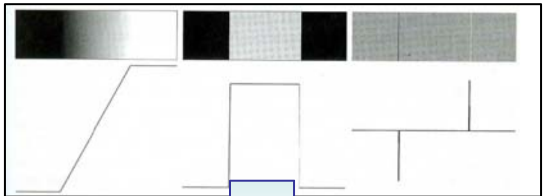


Image free of noise

Introducing additive Gaussian noise into the original image

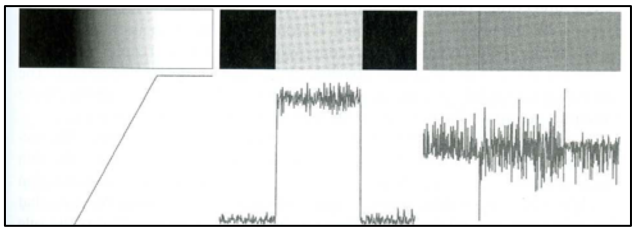


Image corrupted by Gaussian noise with mean σ = 0.1

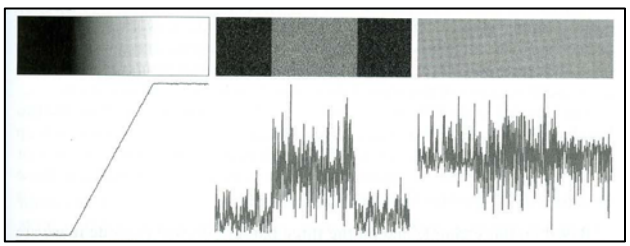


Image corrupted by Gaussian noise with mean σ = 1.0

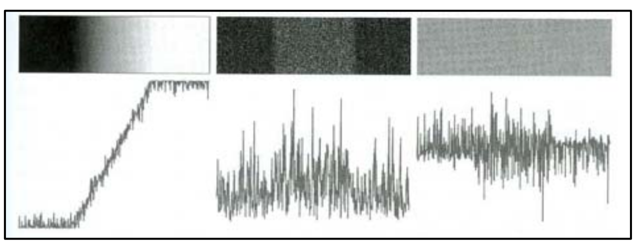


Image corrupted by Gaussian noise with mean σ = 10.0

As evident in figure, the derivatives become increasingly different from the noiseless case

Sensitivity of derivatives to noise

High sensitivity of derivatives to noise

The profile derivatives become increasingly different from the noiseless case. In particular, the second one is more sensitive to noise: it could be difficult indeed to detect their positive and negative components, which are the truly useful features of the second derivative in terms of edge detection.

Important consequences in edge detection

Fairly little noise can have a significant impact on the two key derivatives used for edge detection in image.

Needs of smoothing step

Image smoothing should be a serious consideration prior to the use of derivatives in applications with this kind of noise

Gradient operators

First- order derivatives in a digital image are computing using various approximations of the 2-D gradient.

- The gradient of an image f(x,y) at location (x,y) is defined as the vector:

$$\nabla \mathbf{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}.$$

- Gradient magnitude gives the maximum rate of increase of f(x,y) per unit distance in the direction of ∇f :

$$\nabla f = \mathrm{mag}(\nabla \mathbf{f}) = \left[G_x^2 + G_y^2\right]^{1/2}.$$

- Gradient direction represent the direction angle of the vector ∇f at (x,y):

$$\alpha(x, y) = \tan^{-1}\left(\frac{G_y}{G_x}\right)$$

where the angle is misured with respect to the x-axis.

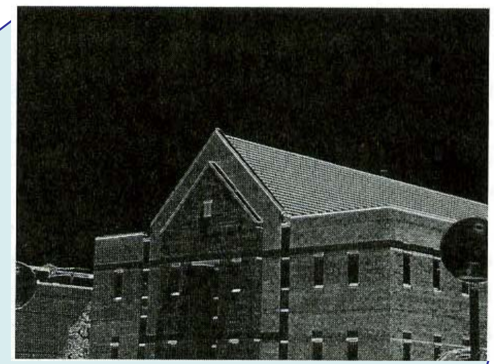
The direction of an edge at (x,y) is perpendicular to the direction of the gradient vector at that point

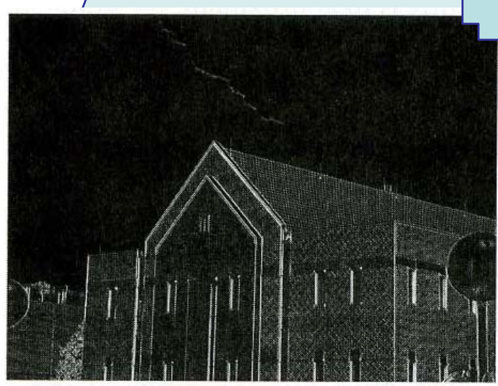
Example:

response of the two gradient components and of the sum of these

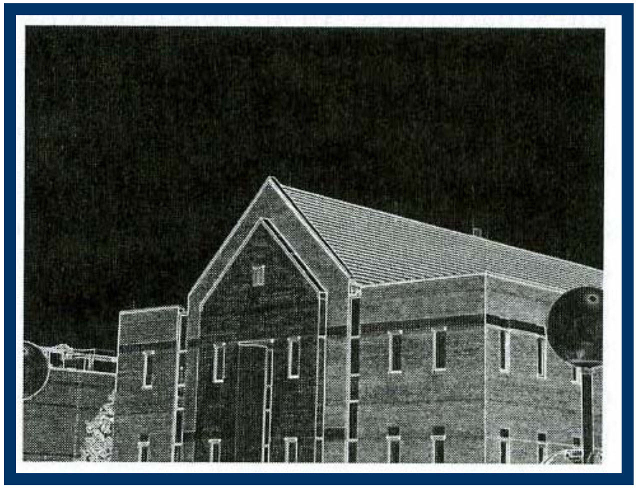
original image

$|G_y|$ **component of the gradient in y-direction**





$|G_x|$ **component of the gradient in x-direction**



Gradient image: $|G_x| + |G_y|$

Laplacian

- **Second-order derivatives in an image f(x,y) are computing using the 2D-Laplacian:**

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}.$$

- **For a 3x3 region (zi are the gray-level values)**

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

the forms frequently used (in practice) for digital approximation are:

$$\nabla^2 f = 4z_5 - (z_2 + z_4 + z_6 + z_8)$$

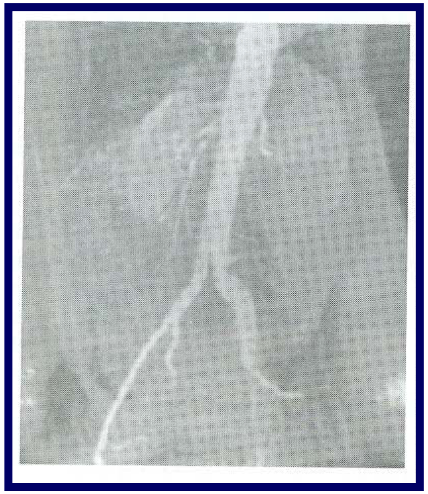
$$\nabla^2 f = 8z_5 - (z_1 + z_2 + z_3 + z_4 + z_6 + z_7 + z_8 + z_9).$$

and the laplacian masks use to implement these two equations are respectively:

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

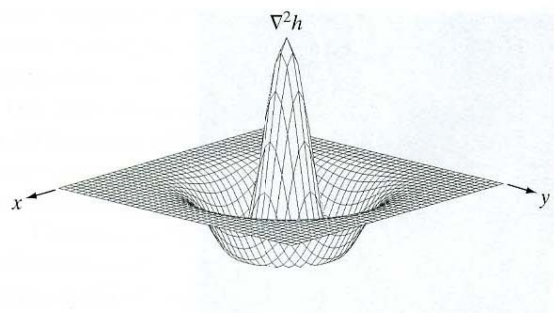
example



angiogram image

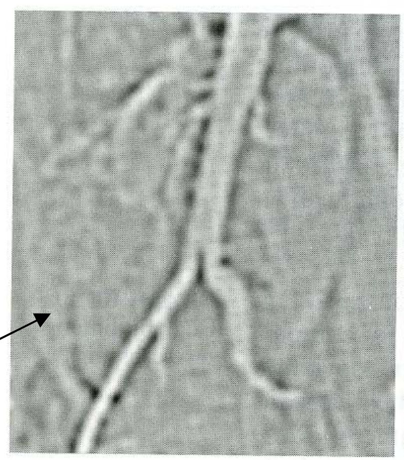
−1	−1	−1
−1	8	−1
−1	−1	−1

laplacian mask



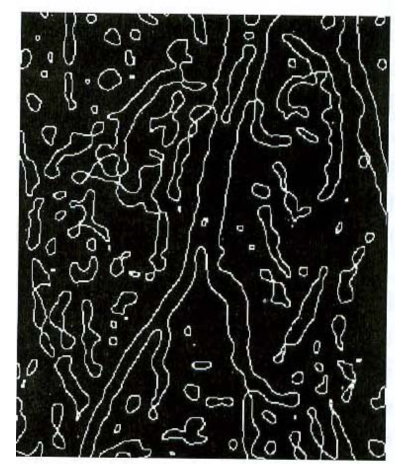
laplacian of a gaussian: 3d-plot

It is obtained by smoothing the original image with the Gaussian smoothing mask, followed by application of the Laplacian mask



LoG image
(Laplacian of a gaussian)

thresholded LoG



zero crossing

Edge linking and boundary detection

Noise and other effects introduce spurious intensity discontinuities in the image

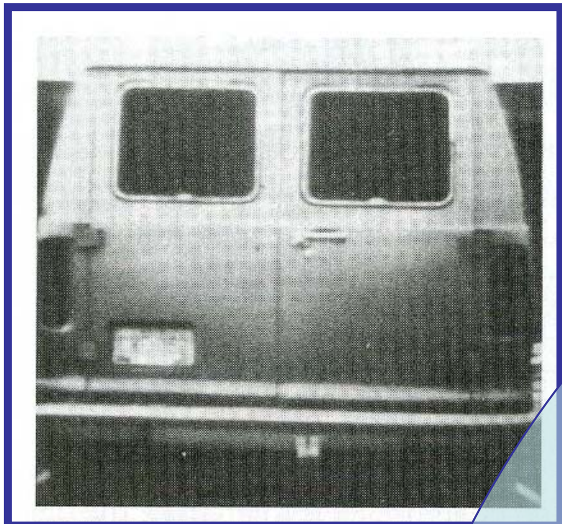
Edge detection algorithm should be followed by linking procedures to assemble edge pixels into meaningful edges

Different approaches for linking edge points are possible:
one of the simplest (local process) is to analyze the characteristics of pixels in small neighbours about every point in an image that has been labelled an edge point.
All points that are similar according to a set of predefined criteria are linked, forming an edge of pixels that share those criteria.

Two proprieties used in similarity criteria are:

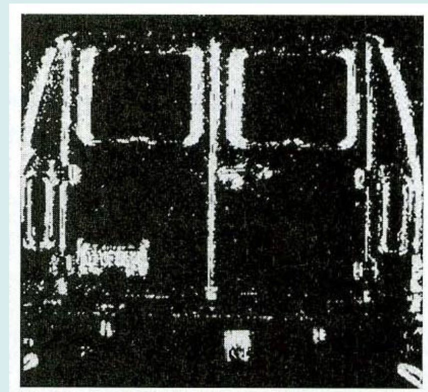
1. **the strength of the response of the gradient operator used to produce the edge pixel**
2. **the direction of the gradient vector**

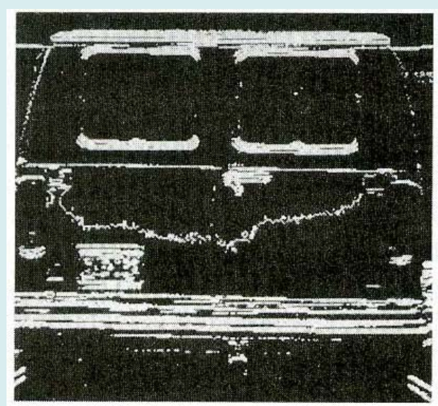
Example



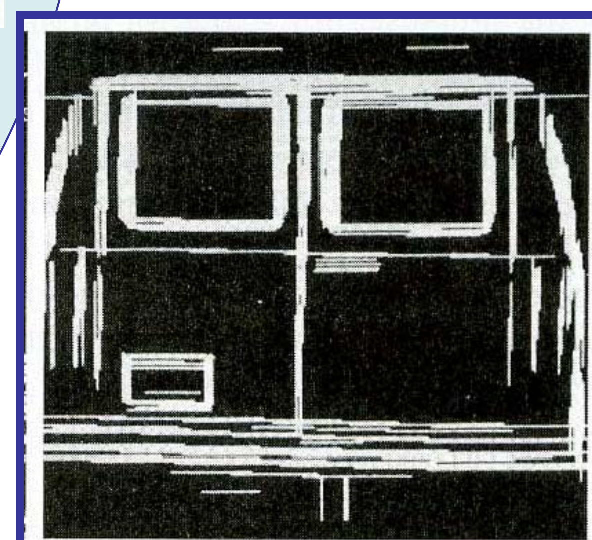
original image

G_y component of the gradient





G_x component of the gradient



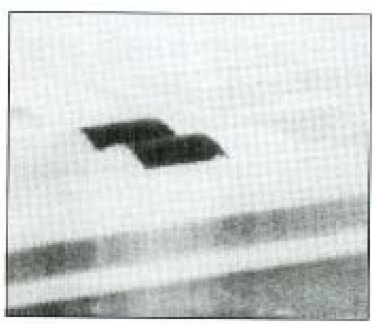
result of edge linking

Using Hough Transform
(global processing)

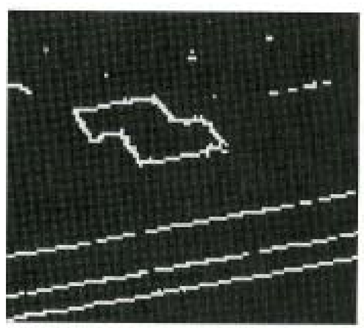
An alternative approach <u>for linking edge points</u> is offered from Hough transform. It allows the recognition of segments, curves and defined shapes.

It is based *on the transformation of all the points of an image into the points of a new space (parameter space)* and on the fact that each edge of the image offers a contribute to the recognition global objective.

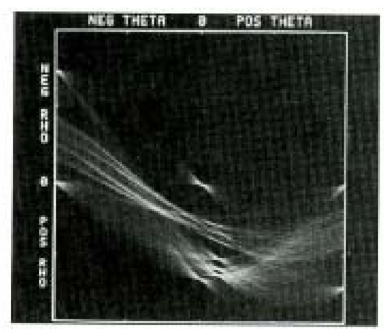
Defined the interest curve to find in the scene, for each point of the image the parameters of all the curves that passes through this point are calculated and the cell of an n-dimensional space are increased. The maximum of the obtained accumulation function are estimated, to find the curve with maximum probability to be present in the image.



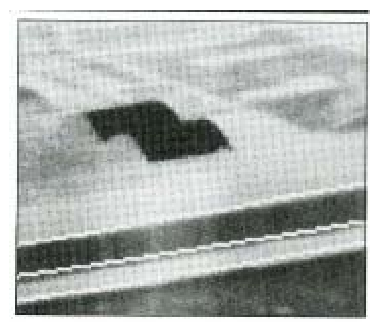
infrared image



Thresholded gradient image

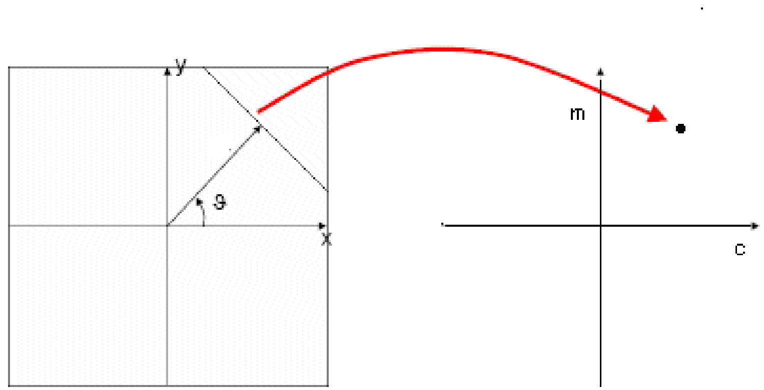


Hough transform



Linked pixels

Trasformata di Hough (1)

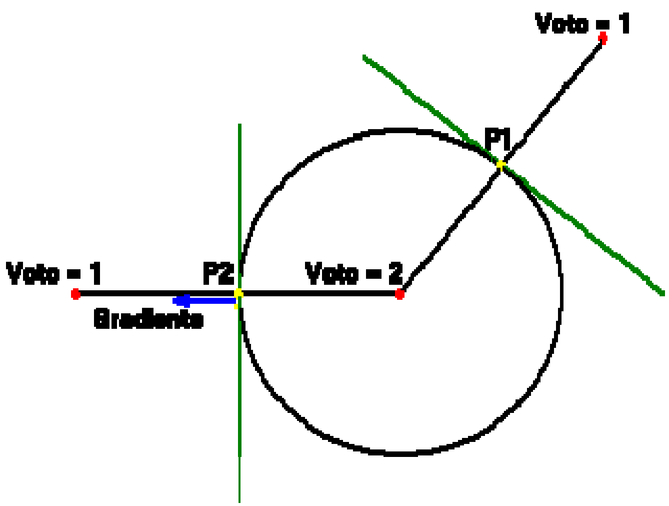


$y = mx + c$

$(x, y) \longrightarrow (m, c)$

Si consideri una discretizzazione del piano dei parametri (m, c).

Ciò permette di rappresentare tale piano su una matrice H(s,t) i cui indici di riga e colonna corrispondono ai valori quantizzati di (m, c).



$(x-a)^2+(y-b)^2= r^2$

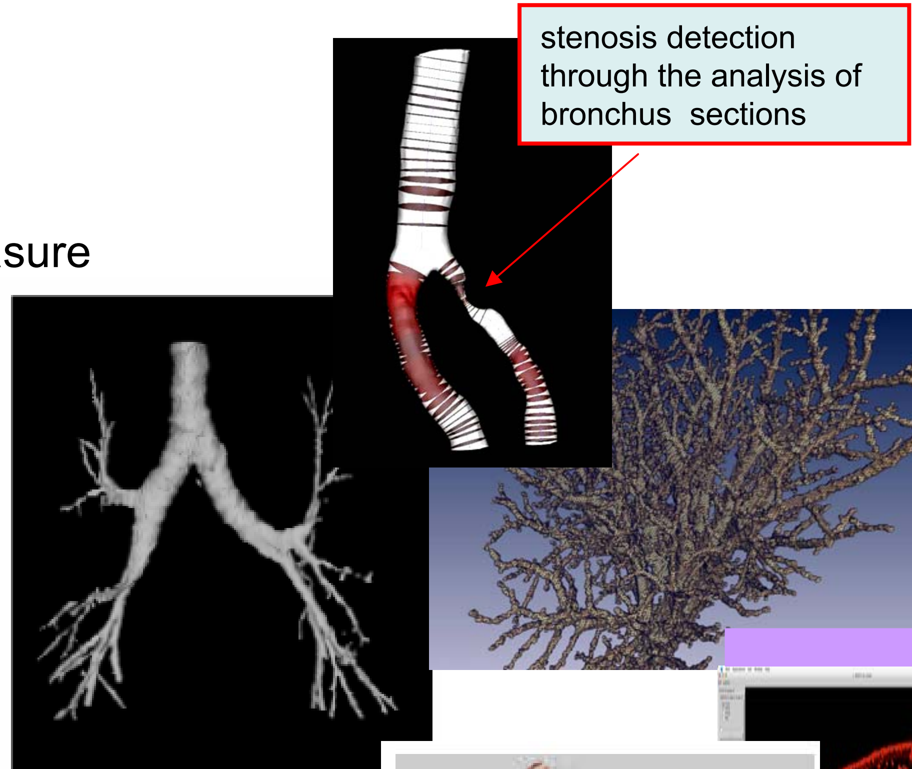
$(x, y) \longrightarrow (a, b, r)$

Il centro della figura deve essere la zona più “votata” nello spazio parametri.

Segmentation algorithms in medical field (1)

- **Airway segmentation** in CT scans

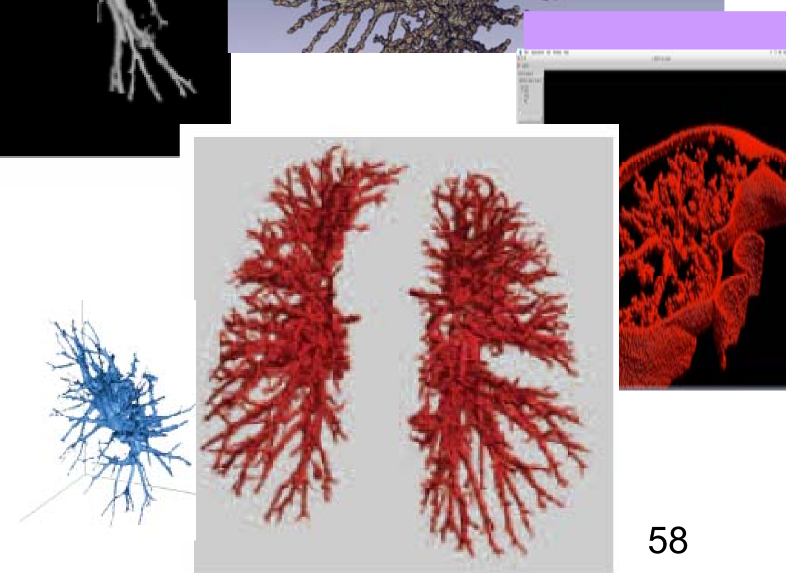
It could be used:
- to identify the airway lumen and to measure airway geometry
- to study airway reactivity
- to guide surgical interventions
- in virtual broncoscopy
-to determine the presence of stenosys
 …



- **Vascular tree extraction**

It could be used:
- to analyze potential nodules with a vascular contact
- to reduce the number of false positive in the pulmonary nodule searching algorithm based on chest CT analysis

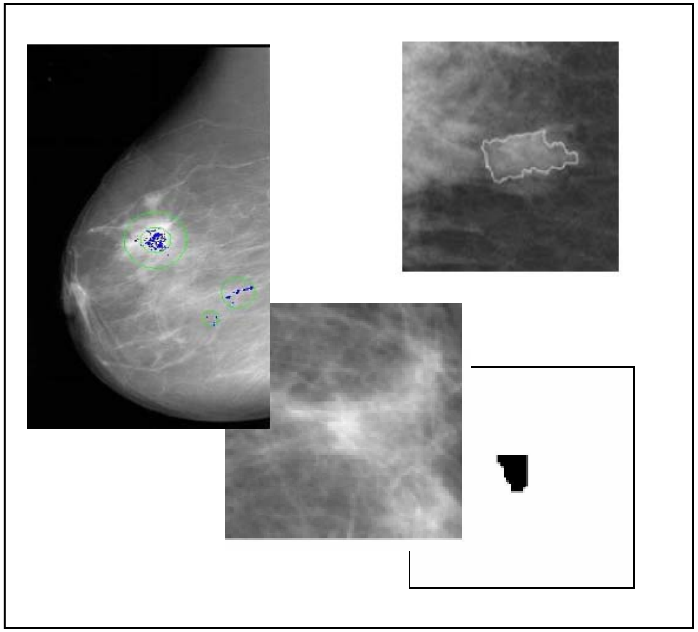
…

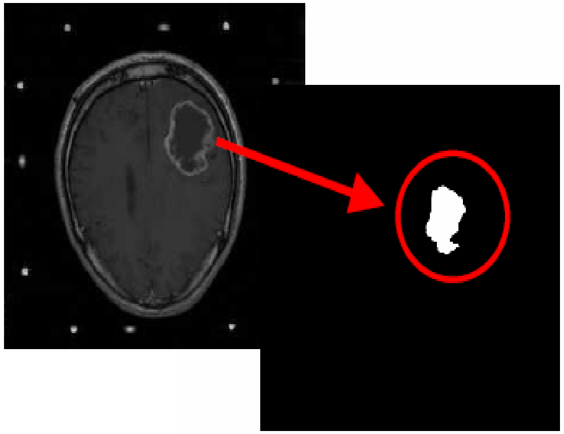


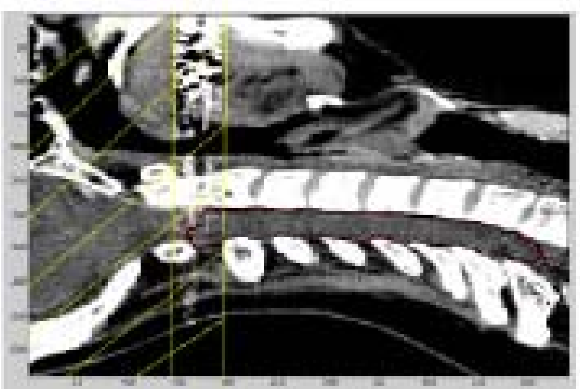
Segmentation algorithms in medical field (2)

- Micro-calcifications and massive lesions identification through mammographic image analysis
- bony marrow segmentation
- different kinds of pathologies identification

…



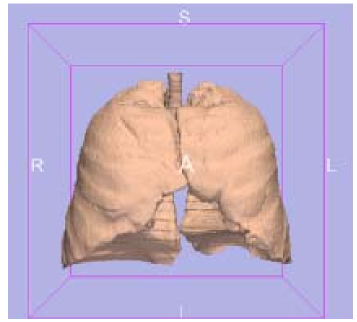




Segmentation algorithms in medical field (3)

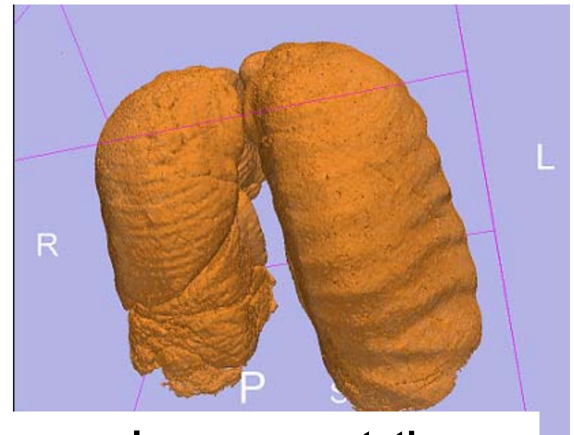
LUNG SEGMENTATION

input: CT scans

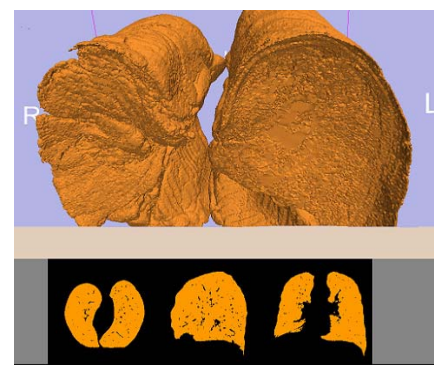


Respiratory apparatus segmentation

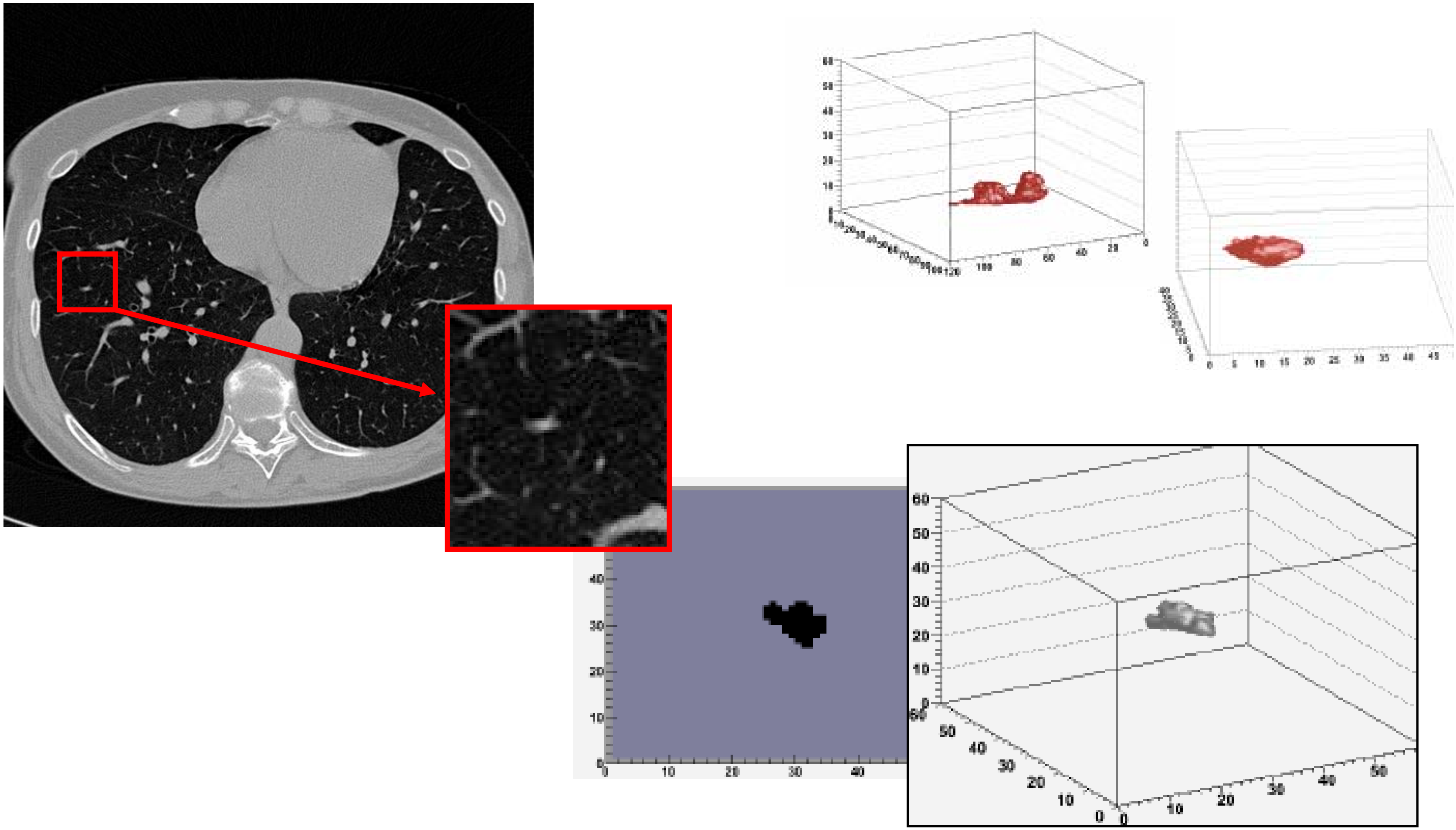
External airway extraction



lung segmentation



Segmentation algorithms in medical field (4)



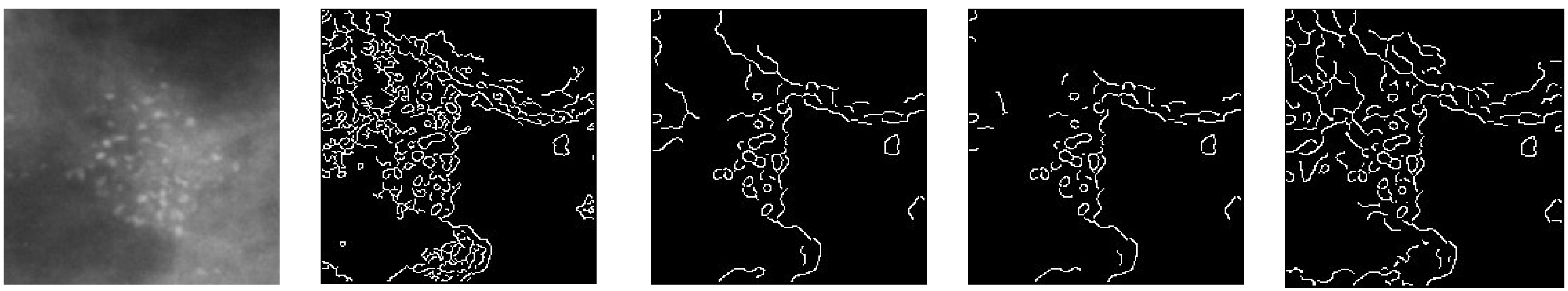
Pulmonary nodule segmentation in chest CT for their analysis

Algoritmo di Canny

- L'immagine è filtrata con filtro di convoluzione gaussiana
- All'immagine filtrata viene applicato un operatore di derivata prima bidimensionale (sono evidenziate le zone con alta derivata prima)
- Vengono mantenuti gli edge dell'immagine e messi a zero tutti i pixel che non sono massimi locali così da ottenere in output una curva

Parametri utilizzati per l' algoritmo di Canny

Il processo di tracciamento utilizza una isteresi controllata tra due soglie T_1 e T_2 ($T_1 > T_2$): il processo comincia solo dove il punto di edge è più alto di T_1 e continua fino a che non trova un punto più basso di T_2



Cluster di microcalcifi cazioni	Sigma= 1.0 T_2=0.2 T_1=0.8	Sigma= 2.0 T_2=0.2 T_1=0.8	Sigma= 2.0 T_2=0.7 T_1=0.8	Sigma= 2.0 T_2=0.2 T_1=0.6

Parametri utilizzati per l' algoritmo di Canny

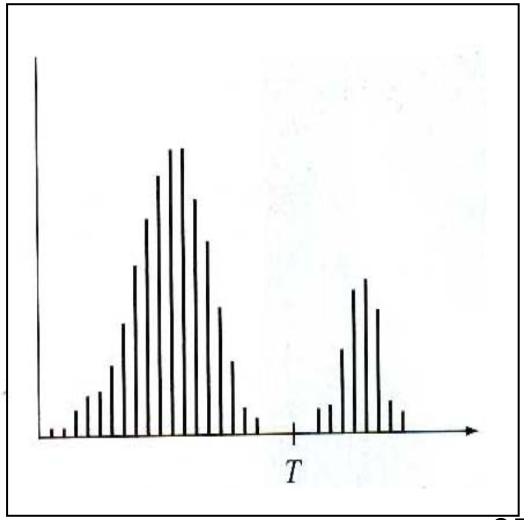
Il processo di traking utilizza una isteresi controllata tra due soglie T_1 e T_2 ($T_1 > T_2$): il processo comincia solo dove il punto di edge è più alto di T_1 e continua in entrambe le direzioni fino a che non trova un punto più basso di T_2

Cluster di microcalcificazioni	Sigma= 1.0 T_2=0.2 T_1=0.8	Sigma= 2.0 T_2=0.2 T_1=0.8	Sigma= 2.0 T_2=0.7 T_1=0.8	Sigma= 2.0 T_2=0.2 T_1=0.6

Thresholding

- This image processing procedure enjoys a central position in segmentation techniques, because of its simplicity and efficiently.
- It is based on the *selection of one or more gray-value thresholds appropriate to discriminate the object from the background.*
- It is very useful when the grey-level histogram is bimodal: object and background pixel grey levels are grouped into two dominant modes.

  In this case, image histogram can be partitioned by a single threshold value (in the valley).



- If f(x,y) denotes the gray level of point (x,y), the thresholded image g(x,y) is defined as:

$$g(x, y) = \begin{cases} 1 & \text{if } f(x, y) > T \\ 0 & \text{if } f(x, y) \leq T. \end{cases}$$

where pixel labelled 1→ object
pixel labelled 0→ background

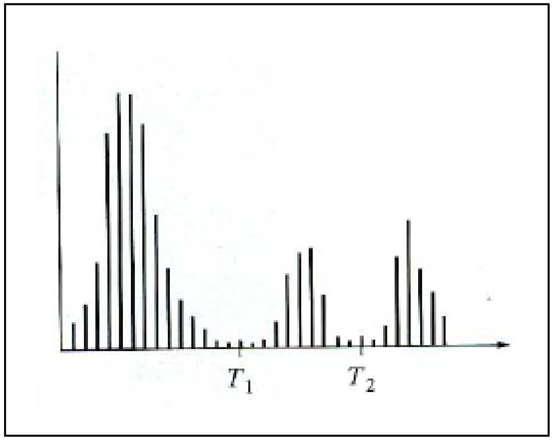
- A more general case of this approach is the ***multilevel thresholding***, when, for example, three dominant modes characterize the image histogram.

- In this case the thresholding procedure classifies a point (x,y) as belonging:

  - to one object class if $T1 \leq f(x,y) \leq T2$

  - to the other object class if $f(x,y) \geq T2$

  - to background if $f(x,y) \leq T1$



- In general, segmentation problems requiring multiple threshold are best resolved using region growing methods

- Thresholding may be viewed as an operation that involves tests against a function T of the form:

$$\boxed{\mathbf{T = T\,[\,x,\ y,\ p(x,y),\ f(x,y)\,]}}$$

where f(x,y) = the gray level of point (x,y)
p(x,y) = a local propriety of this point
(e.i. the average gray level of a neighboor centred on (x,y))

- Threshold is called:
global → if depends only on f(x,y)
local → if depends on both f(x,y) and p(x,y)
dynamic or adaptive → if depends also on the spatial coordinates x and y

The role of illumination

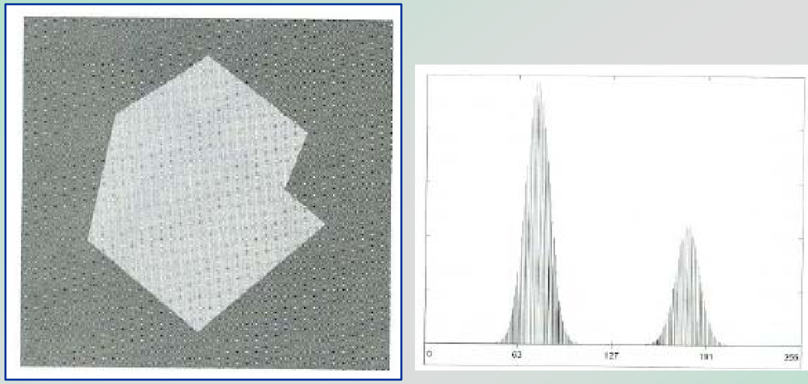
- Fundamental is the *effect of illumination on thresholding* (especially on global one): an image resulting from poor illumination could be quite difficult to segment, due to thedistortion of its histogram.

- For this reason, the type of global thresholding can be expected to be successful in high controlled environments, where the control of the illumination is feasible (i.e. industrial inspection applications).

- Then, when access to illumination is available, a solution frequently used in pratice to compensate for nonuniformity is to project the illumination pattern onto a costant, with riflective surface.

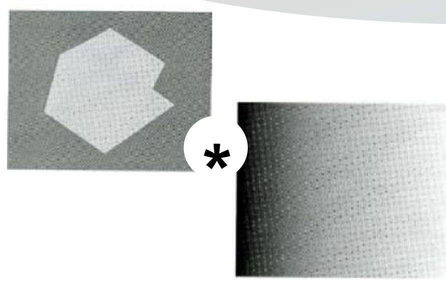
- Another approach to solve the problem is the use of an adaptive threshold, based on subdivision of the image into smaller subimages.

The role of illumination: an example

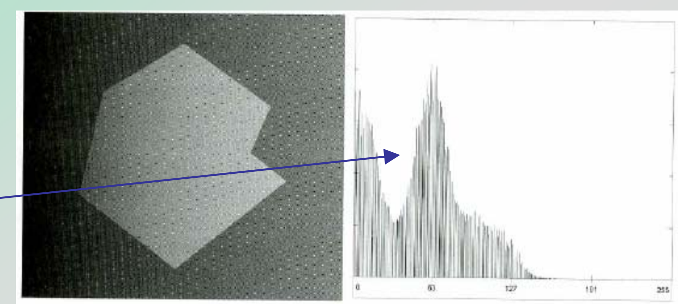


Original image and its histogram
The histogram of reflectance function is clearly bimodal and could be easily partitioned by placing a single threshold

make the product of the original image and computer generated illumination



The original valley is virtually eliminated , making segmentation by a single threshold an impossible task



Product image and its histogram

Basic global Thresholding

- It is based on partition of the image histogram by **using a single global threshold**. It is the simplest of all thresholding techniques.

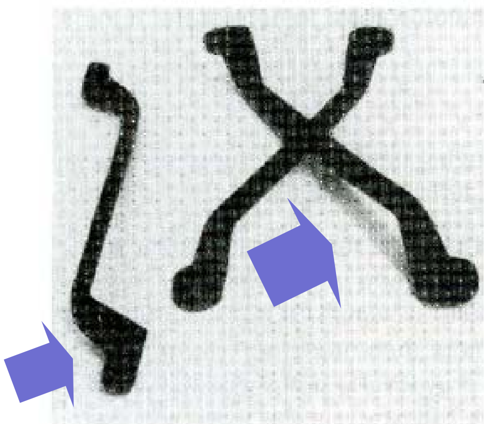
- *Scanning* of the image pixel by pixel and *labelling each pixel* as object or background is made.
  It depends on whether the gray level of that pixel is greater or less than the predefined value T.

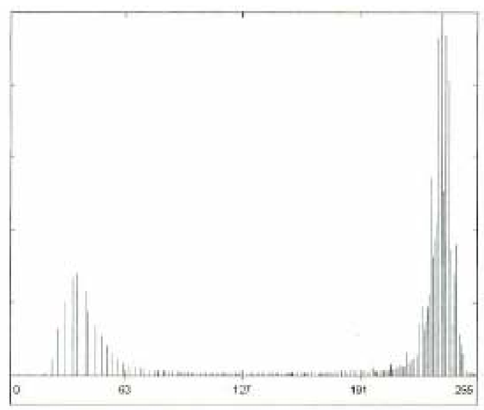
- In bimodal histogram the optimal value of threshold can be defined though the following iterative procedure (*Isodata algorithm*)

  1. The histogram is divided in two regions with equal number of bins and the mean values of the bins in the two regions are computed;
  2. The previously computed mean values are averaged and the bin having the intensity nearest to the new mean is selected as the threshold to divide the histogram;
  3. The routine is iterated until the threshold bin does not change any more.

using the midway value

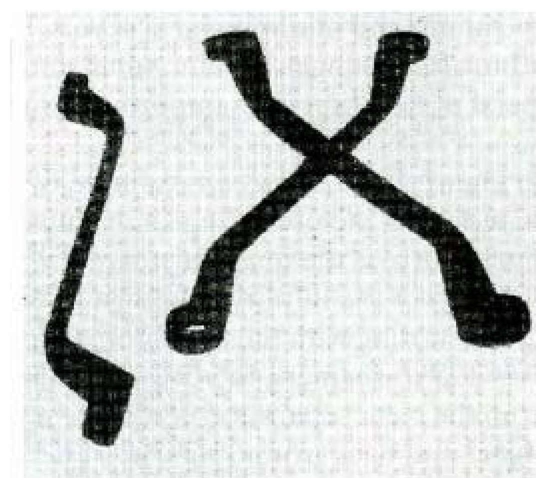
- Using as threshold the ***midway value between the maximum and minimum gray levels***, we can obtain a "clean" segmentation, by eliminating the shadows and leaving only the objects themselves.





original image and its gray-value histogram

Result of global thresholding with midway between the maximum and minimum grey levels.
The absence of the object shadows is evident.
The image appears "clean" from "extern object".



Using isodata algorithm

original image

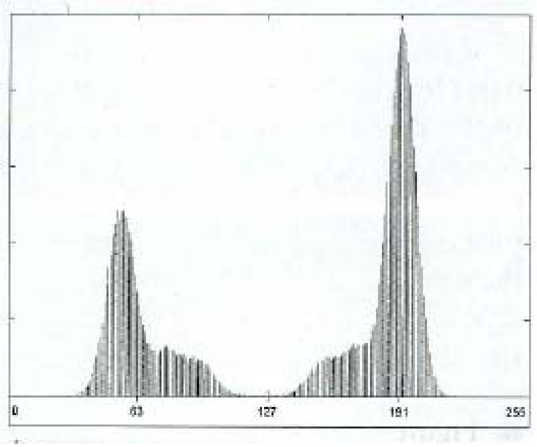


image histogram

iterative algorithm → T = 125.4

result of segmentation with threshold estimated by iteration

The segmentation between object and background is very effective

Image segmentation using an estimated global threshold

Basic Adaptive Thresholding

It is based on the subdivision of the image into smaller sub-images, then subdivided by more parts again.

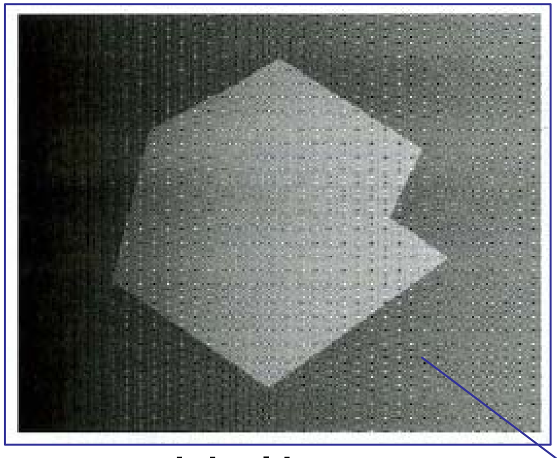
- This procedure is often useful to reduce the effect of nonuniform illumination: the partition of the scene into more regions is such that the illumination of each sub-image can be considered approximately uniform

- The key issue in this approach are:

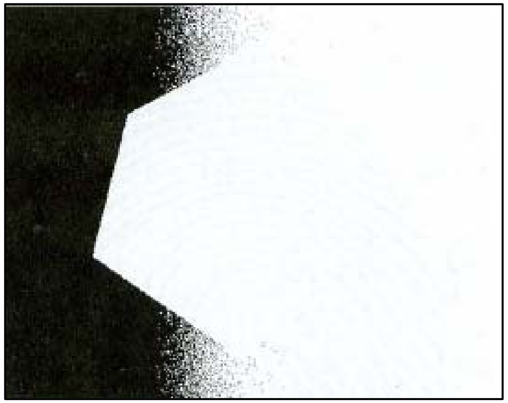
  1. **subdivion**: how to subdivide the image in its subparts?
  2. **threshold values**: how to estimate the optimal threshold for each sub-image?

- It is called “adaptive” because of the threshold used for each pixel depends on the location of the pixel in terms of the subimage

Example



original image



result of global thresholding

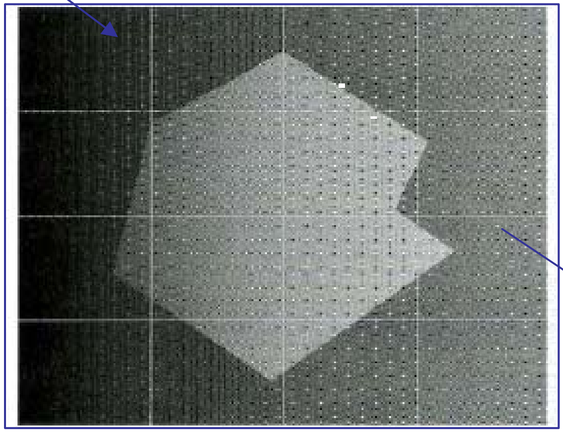
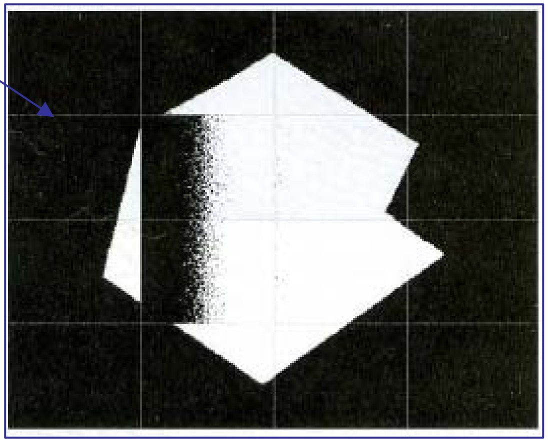


Image subdivided into
individual sub-images



result of adaptive thresholding

Boundary Characteristics for histogram improvement

The choose of an appropriate threshold is favourite if the image histogram peaks are tall, narrow, symmetric and separated by deep valleys.

An approach to improve the histogram shape is ***to consider only those pixels that lie on or near the edges between objects and the background.***

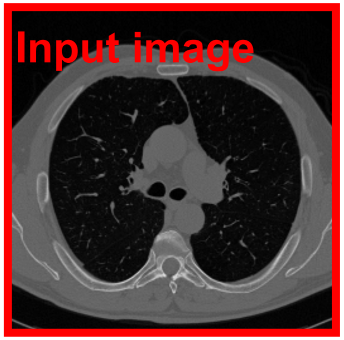
Immediate improvements:

1. **Histogram would be less dependent on the relative size of objects and background**.
   For instance the histogram of an image composed of a small object on a large background are (or the contrary) would be dominated by a large peak because of the high concentration of one type of pixels.

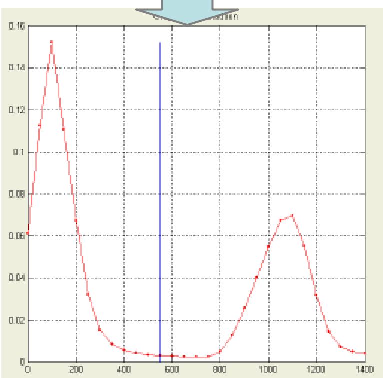
2. **Resulting histogram would have peaks of approximately the same height**

3. **The symmetry of the histogram peaks is improved:**
the probability that any of those given pixels lies on an object would be approximately equal to the probability that it lies on the background

Example of 2D-segmentation algorithm based on thresholding



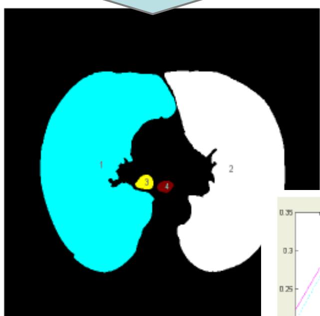
Typical input image: a CT *slice.*
CT resolution: 512x512
16 bit/pixel (12 really used)
2^{12} grey-levels describe the density interval from air (tissue absence), whose intensity pixels is near to zero, to bony tissue, corresponding to high intensity.



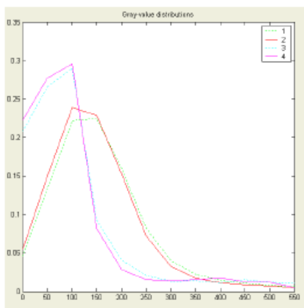
The grey tone distribution of the CT pixels typically shows two well distinct parts: one containing air, lung parenchyma, trachea and bronchial tree; the other one containing vascular tree, bones, muscles and fat.
A good threshold is set at the plateau between these two regions.

Through the application of the *Isodata method*, a binary mask is obtained.

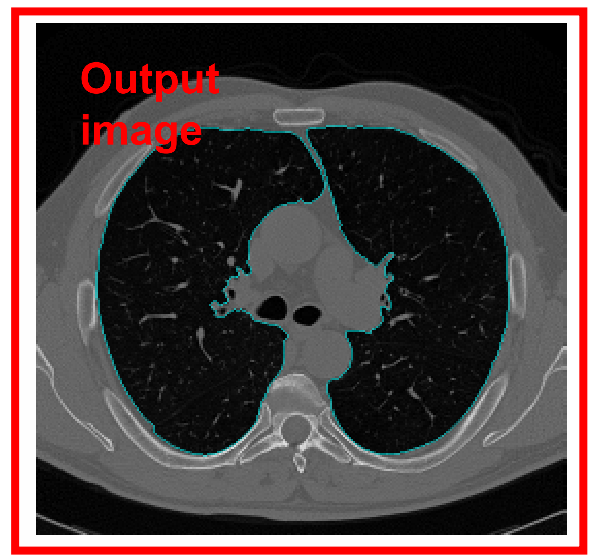
Border and noise removing



Identification and labelling of the connected objects.

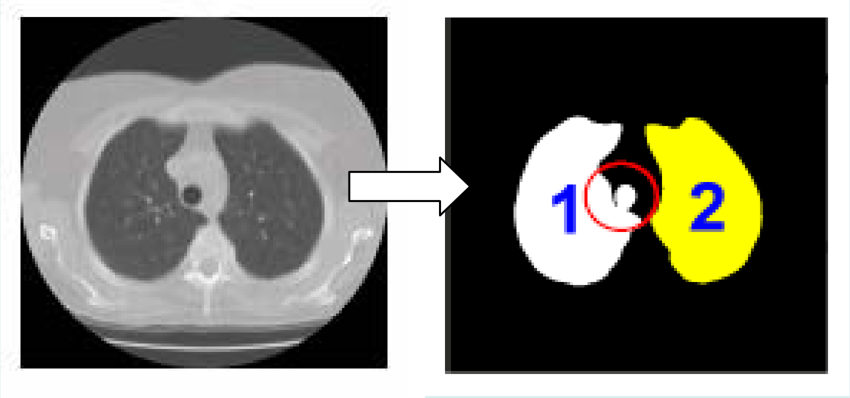


Grey distribution of lung regions (violet peak) are easily distinct from bronchial tree (red peak).
Bronchial tree composition (air) is different from lung parenchyma one. It is use to distinguee lung tissue from bronchial tree.

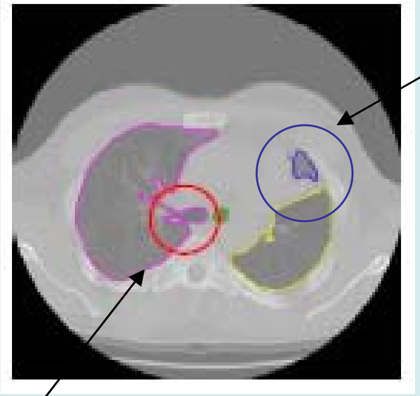


From logical AND between the original image and the mask, the segmented lung volume is obtained.
It can be studied, for example, for nodule searching.

Some cases of incorrect segmentation



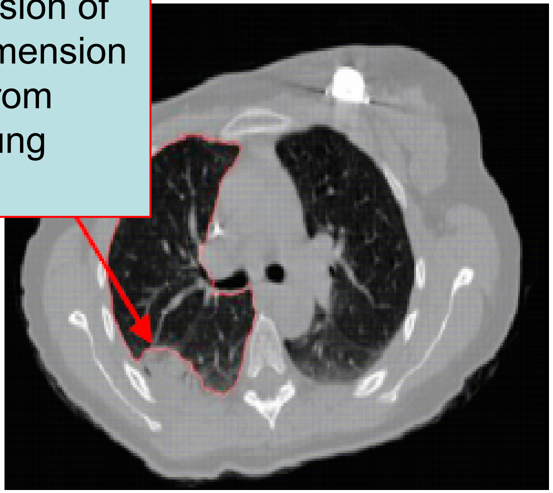
"Fusion" of trachea and right lung: they are incorrectly identified in a single structure

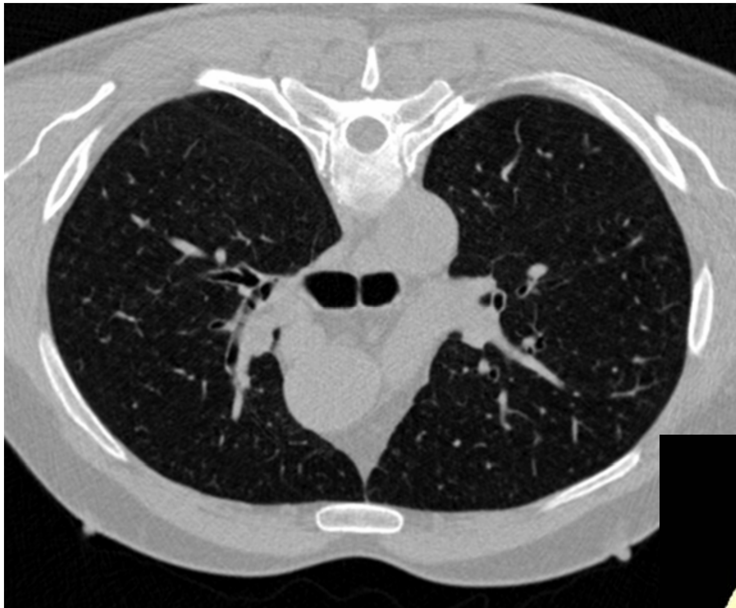


Two section of the same lung are identified as different structures

"Fusion" of bronchial tree and right lung: they are incorrectly identified in a single object

An internal lesion of significant dimension is excluded from segmented lung volume.

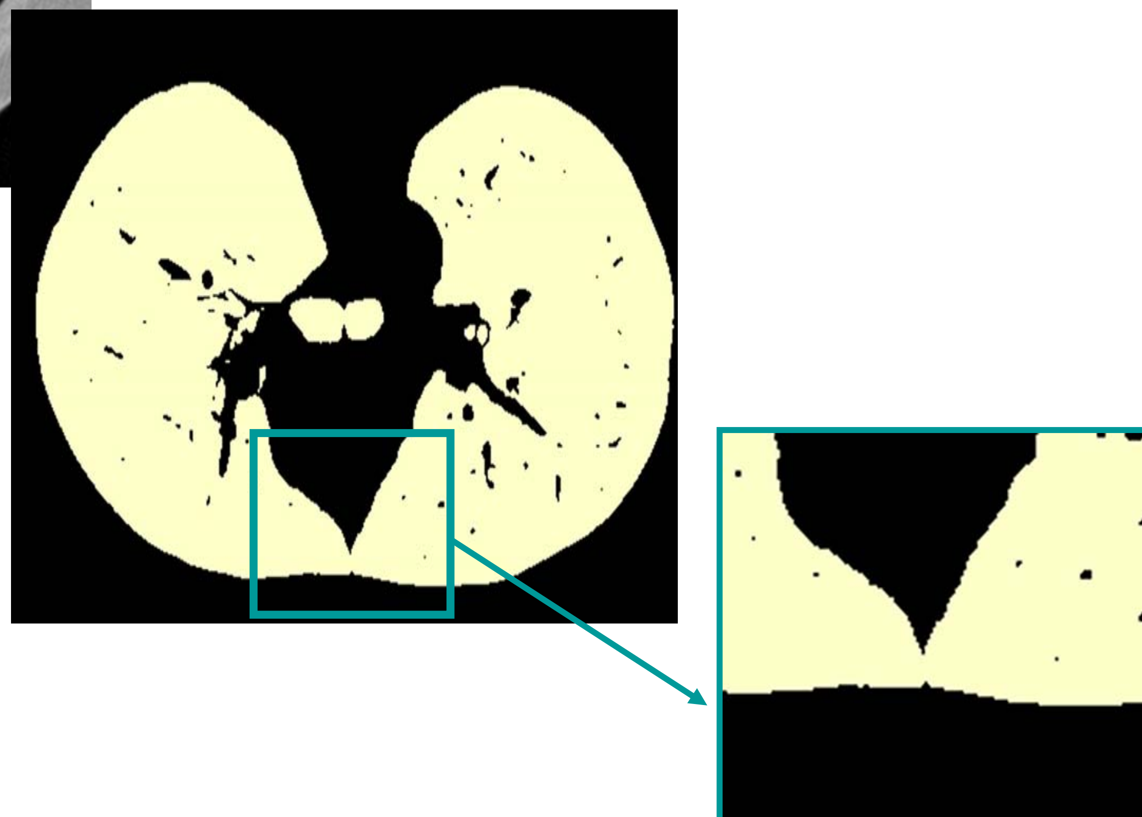




Problems of incorrect segmentation

The right and left lung appear fused in a single object.

It is due to volume partial effects.



Region Based Segmentation

While the discussed techniques are developed on the distribution of the pixel proprieties (such as gray-level), region based segmentation aims to find the regions directly.

If R represents the entire image region, segmentation may be view as the process that partitions R into n sub-regions R_1, R_2,…R_n such that some properties are satisfied:

(a) $\bigcup_{i=1}^{n} R_i = R$

(b) R_i is a connected region, $i = 1, 2, \ldots, n$

(c) $R_i \cap R_j = \varnothing$ for all i and j, $i \neq j$

(d) $P(R_i) = \text{TRUE}$ for $i = 1, 2, \ldots, n$

(e) $P(R_i \cup R_j) = \text{FALSE}$ for $i \neq j$

where P(Ri) is a logical predicate defined over the points in set R, and Ǿ is the null set

Condition meaning

- (a)→ **segmentation must be complete (every pixel must be in a region)**
- (b)→ **points in a region must be connected in a predefined sense**
  (there are different kinds of connectivity)
- (c)→ **the regions must be disjoint**
- (d)→ **deals the proprieties that must be satisfied by the pixels in a segmented region**
  (for example P(Ri) = TRUE if all the pixels in Ri have the same gray level)
- (e)→ **regions Ri and Rj are different in the sense of predicate P**

Region growing (RG)

An iterative image analysis technique employed *to identify connected regions of pixels/voxel satisfying a predefined criteria (inclusion rule)* and according to well-defined notion of discrete connectivity or contiguity.

The algorithm works as follows:

1. **a seed point (*original seed*) is chosen in the region of interest to be grown and its neighbours* are checked;**

2. **if the neighbours satisfy a specific inclusion rule, they are included into the growing region, otherwise they are ruled out;**

3. **all points included at a certain iteration become seed points (*actual seed)* for the following step;**

4. **the process is iterated until no more points satisfy the inclusion rule and can be added to the growing area/volume.**

* according to a predefined criterion of connectedness

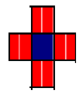
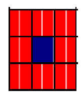
Examples of inclusion rules

The selection of similarly criteria depends not only on the problem under analysis but also on the type of image data available. They are often based on threshold values

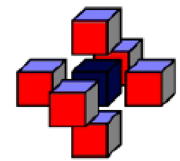
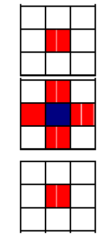
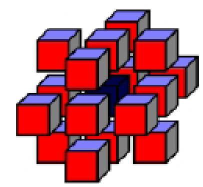
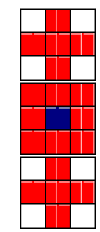
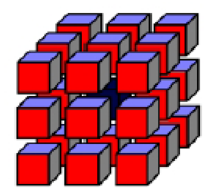
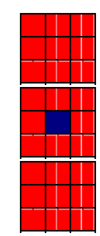
- ***Simple bottom/top Threshold***: if the pixel intensity is greater/lower that a certain threshold, the pixel is included into the growing region.
- ***Double Threshold:*** if the pixel intensity is enclosed in an defined interval, the pixel is included into the growing region
- ***Mean Simple bottom/top Threshold***: for every pixel, the average of its intensity and those of its neighbors is calculated. If it is greater that a certain threshold, the pixel is included into the segmented region.
- ***Mean Double Threshold***: the intensities of every pixel and its neighbors are averaged; if the average is enclosed in an defined interval, the pixel is included into the growing region.
- ***Average Homogeneity I***: The average intensity I_1 between the seed point and its neighbors is calculated. The average intensity I_2 between the considered pixel and its neighbors is valuated. If the difference between I_1 e I_2 is lower that a defined threshold, the pixel is included into the growing region.

The choice of the inclusion rule with the optimal threshold and the selection of a proper seed point are of great relevance for the best working of the algorithm

Different criterions of connectedness

Dominio	Connettività	Neighbors (vicini)
2D	4-connected	
	8-connected	

typologies of connectivity **in 2D**

6-connected		
18-connected		
26-connected		

typologies of connectivity **in 3D**

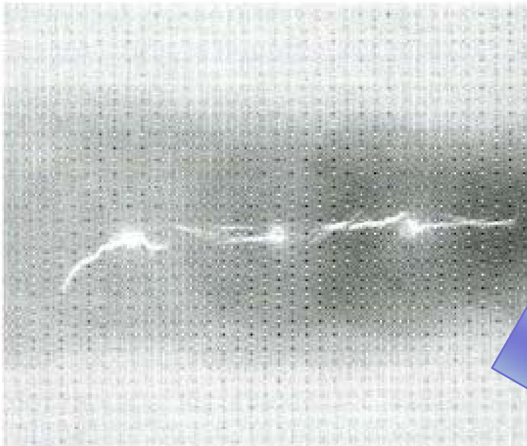
Stopping rule

• Generally a growing process should stop when no more pixels satisfy the criteria for inclusion in that region.

• One or more additive criteria that establish when the growing of a region must be stopped can be introduced. They should interrupt the growing process when the proposed objective is reached

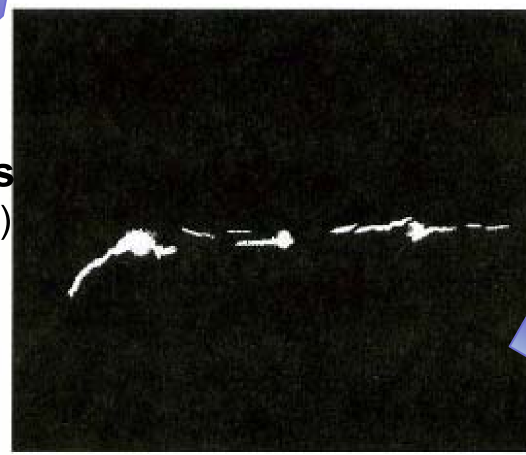
• For example, stopping rules can be based on:
- gray level
- texture
- colour
- size
- likeness between a candidate pixel and the pixels grown so far
- shape of the region being grown

Example:
application of RG in weld inspection

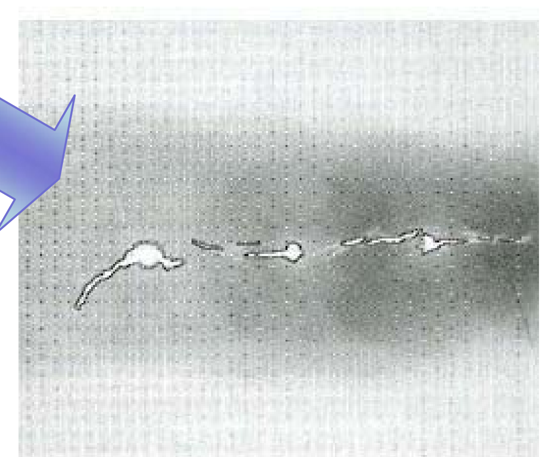


Input image:
X-ray image of a defective welds
(it contains several cracks and porosity)

using of region growing technique to segment the regions of the weld failure



result of region growing process



Boundaries of segmented defective welds

The choice of seed point is based on the concept that the pixels of defective welds tend to have the maximum allowable digital value
(in this case 255)

An useful application of RG:

THE NODULE HUNTER

1. A digital image (for example a scan of the 3D matrix) is carried out and the first point satisfying the inclusion rule is chosen to start the growth.

2. When the growth is finished, the segmented region is removed from the image;

3. The image scan restarts to search for a new seed point.

4. This routine is iterated until no more seeds are found.

In this way, a number of not connected regions satisfying the same inclusion rule is obtained.

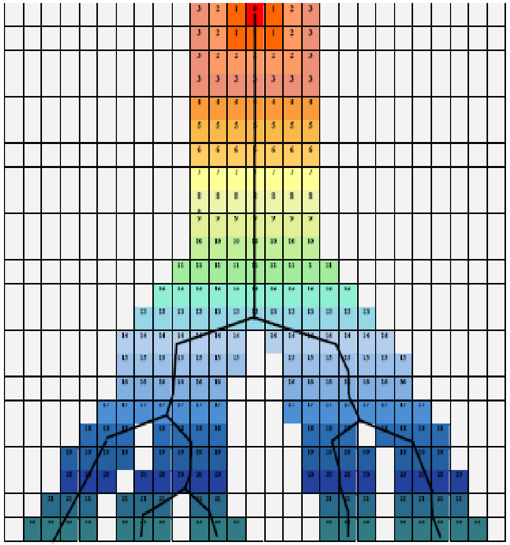
Wavefront algorithm: an "alternative" RG

- **An alternative version of RG can be used *for external bronchi extraction*. A wavefront algorithm is an iterative procedure that combines the principles of the traditional RG with a *wavefront simulation model*.**

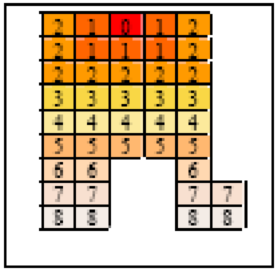
It assure a correct handling of hilus (region in which airways and vascular tree enter into the lung parenchyma), notoriously difficult to be segmented.

- The algorithm aims at the segmentation of the airways out, starting from the trachea down to the bronchi entrance into the lungs:
  - The initial seed is automatically chosen in the trachea;
  - The growth process starts, according to the predefined inclusion rule;
  - It is interrupted by appropriate stop conditions that do not allow flooding into the lung parenchyma

- **Unlike traditional RG techniques, this method *assigns label k to each voxel grown in iteration k*. The label is equal to the checkerboard distance of the voxel from the starting seed: in this way a distance map is built in the volume. Equally labeled voxels are grouped into connected components; the set of the CCs with the highest label *k* (the most recent iteration number) simulates a wavefront, propagating down the airway lumen.**



- When the wavefront finds a bifurcation it splits and grows simultaneously down each ramification. Bifurcations are detected by verifying the existence of disconnected components with the same *k* value.



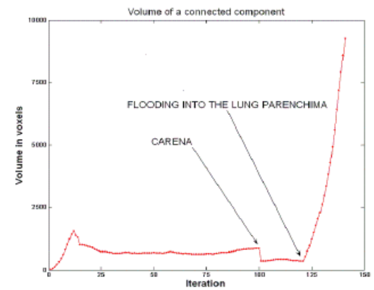
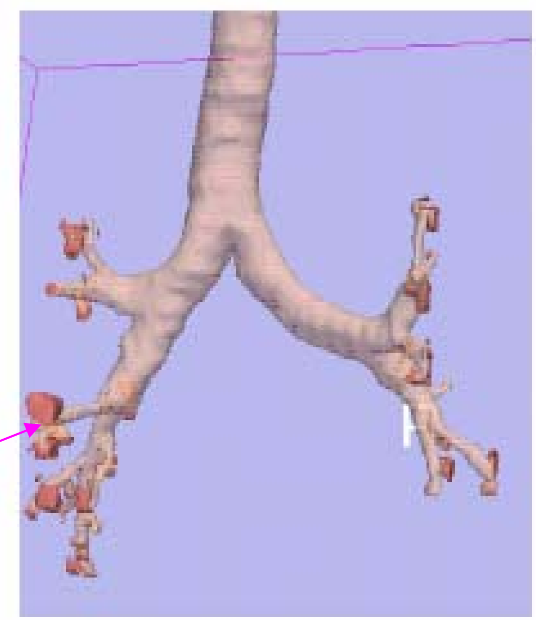


Figura 56 Analisi di un profilo di crescita

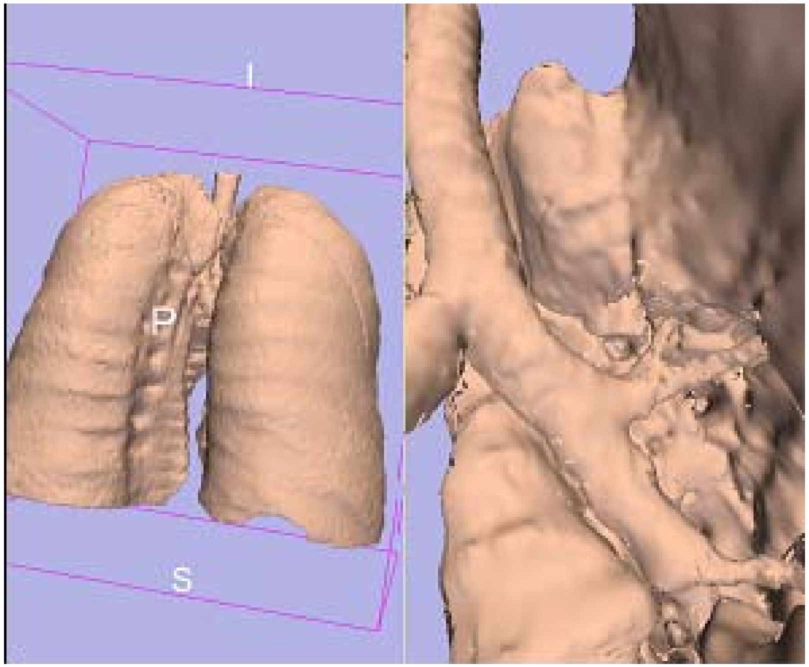
- To interrupt the grown process when it is entering the lung ‘boundaries’ (ideally while traversing the pleura), appropriate stop conditions are imposed.

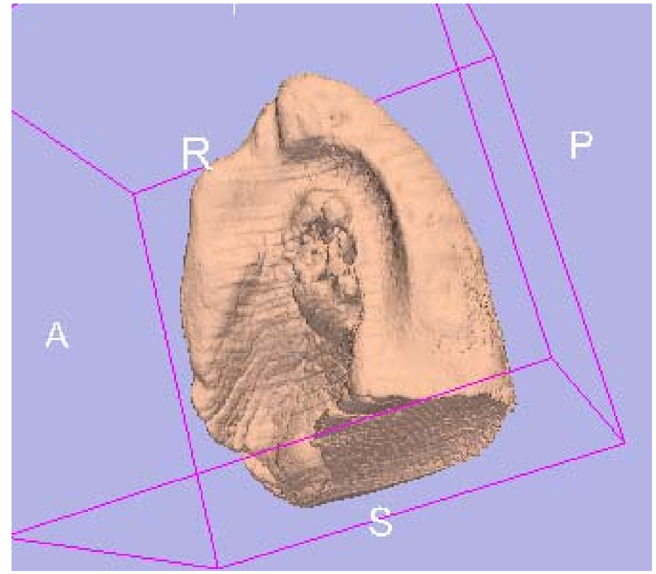
- In this way the RG process will not flood into the pulmonary parenchyma and the extra-pulmonary airways can be extracted. The hilar region is correctly handled.

The last connected components



The algorithm assure the correct handling of hilar region





Lung segmentation
as a preliminary step of a CAD

- **CAD (*Computer Aided Detection*):** software for the automatic detection of pathological structures (such as tumours) in medical images.

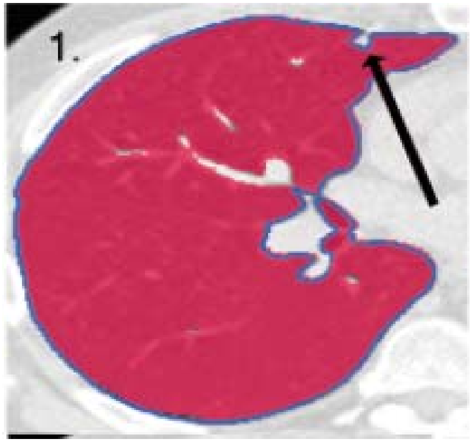
- In a CAD, segmentation step is necessary *to reduce the volume of the image under analysis*, fundamental:

  - **to reduce the computational time;**
  - **to limit the number of false positive FP**

  (FP: healthy subject erroneously identified like sick)

LUNG SEGMENTATION in MAGICV-CAD

In the MAGICV-CAD, lung segmentation is obtained as follows:

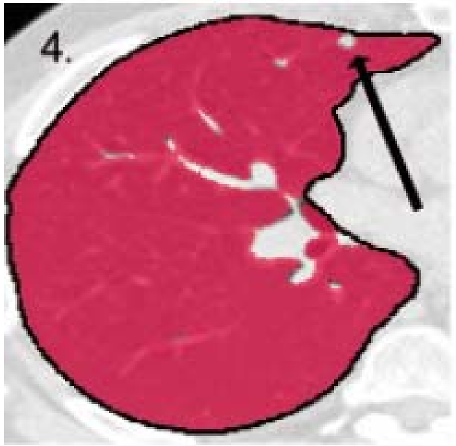


1) **Region growing** to define lung parenchyma

At this stage, anatomical structures external to the lung parenchyma are not included in the segmented volume. Also nodules and internal vascular tree are ruled out because they don't satisfy the inclusion rule.

2) **2D-Glued Elastic Band** to refine the anatomic lung contour, including nodules and internal vascular tree in the segmented volume.

- It is based on the simulation of the dynamics of an elastic band. The virtual spline, joining a number of nodes, is glued around the contour of the region of interest and it evolves, driven by some forces.
- It is applied slice by slice.



Concave regions of the lung contour **with small bending radius** are included in the segmented volume; those with **great bending radius** are ruled out .

Using Graph-Theoretic Techniques

(global processing)

An alternative global approach for edge detection and linking is based on *representing edge segments in the form of a graph and searching the graph for low-cost paths that correspond to significant edges*.

This representation provides a a rugged approach that performs well <u>in the presence of noise.</u>

But this procedure is considerably <u>more complicated</u> and required <u>more processing time</u> than the other methods.

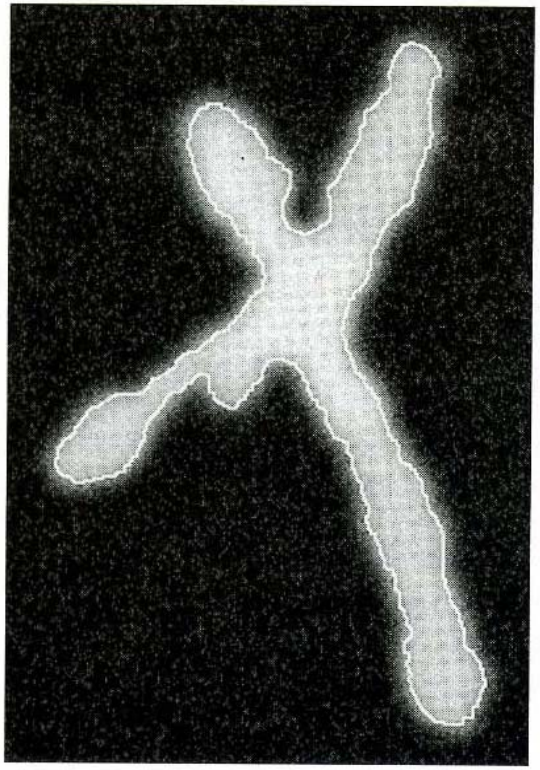


Image of noisy chromosome silhouette.
The edge boundary, determined by graph search, are indicated in white.

Region splitting and merging

It is based on the *initial subdivision of the image into a set of arbitrary, disjointed regions and than merge and/or split the regions in an attempt to satisfy the just introduced five conditions:*

(a) $\bigcup_{i=1}^{n} R_i = R$.

(b) R_i is a connected region, $i = 1, 2, \ldots, n$.

(c) $R_i \cap R_j = \varnothing$ for all i and j, $i \neq j$.

(d) $P(R_i) = \text{TRUE}$ for $i = 1, 2, \ldots, n$.

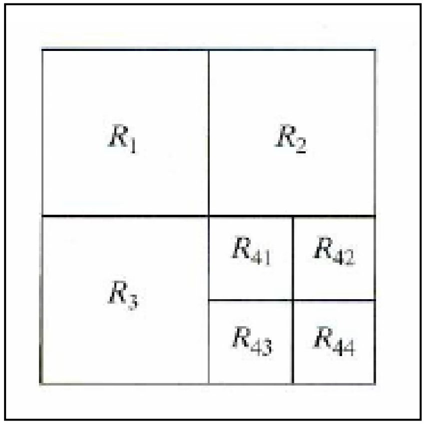
(e) $P(R_i \cup R_j) = \text{FALSE}$ for $i \neq j$.

It is an iterative procedure:

•Selected a predicate P, if R represent the entire image region an approach to segment R is to subdivide it successively into smaller and smaller quadrant regions, so that, for any region Ri, P (Ri) = TRUE.

•The algorithm starts with the entire region:
if P(R) =FALSE, the image is divided into quadrants.

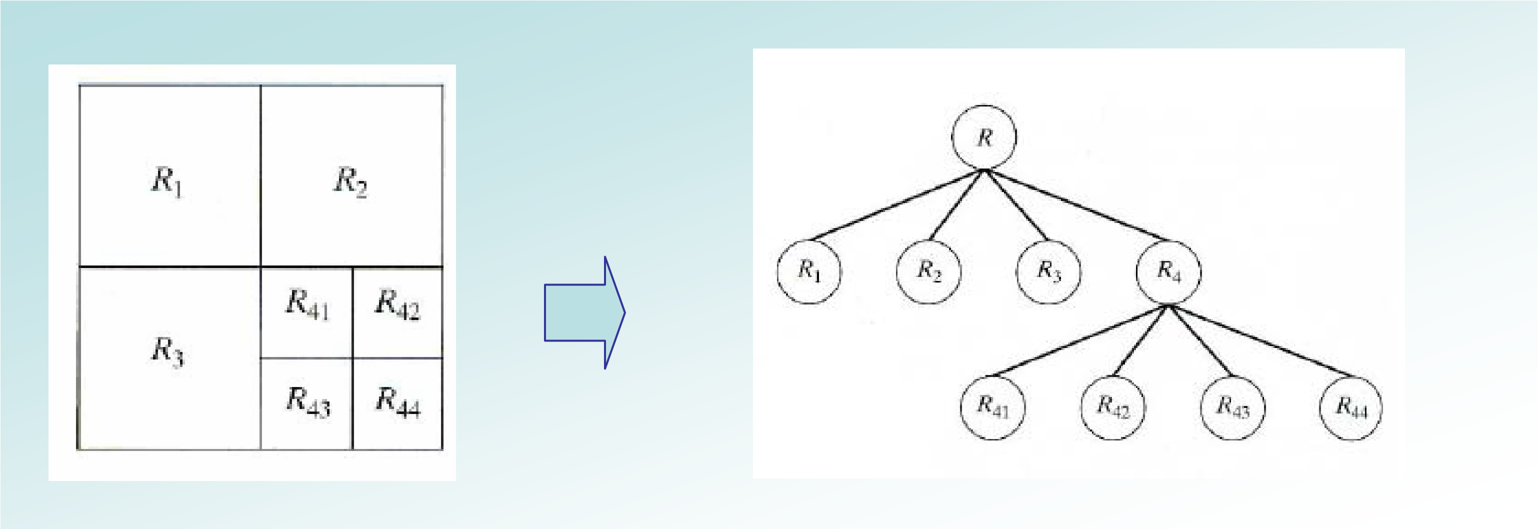
•if P is FALSE for any quadrant, that quadrant is subdivided into subquadrants and so on…



partitioned image

“Quatree” representation

- Splitting technique has a convenient representation in the form of a “quatree”, a tree in which nodes have exactly four descendants.
- The root of the tree corresponds to the entire image
- Each mode corresponds to subdivision



splitting and merging

splitting step → the final partition would contain adjacent regions with identical proprieties

merging step→ to merge only adjacent regions whose combined pixels satisfy the selected predicate P.
That is two adjacent regions Ri and Rj are merged only if P (RiURj) = TRUE.

<u>Procedure scheme</u>
At any step:

1. **splitting into four disjoint quadrants of any region Ri for which**
   P(Ri) = FALSE

2. **merging any adjacent regions Ri and Rj for which**
   P(RiURj) = TRUE

3. Stop when no further merging or splitting is possible