Introduction to image processing and image segmentation

Elena Faggiano

MOX

Outline

Cos'è un'immagine?

In questa lezione parliamo esclusivamente di immagini digitali.
Un'immagine digitale I è (nel caso 2D) una matrice 2x2 definita da:

- risoluzione (numero di pixels o anche numero di righe e colonne della matrice)
- profondità o risoluzione in ampiezza (numero di potenziali valori che può assumere ciascun pixel). Se parilamo di immagini a livelli di grigio (NON colorate), un valore standard è 256 (8 bit) ma dipende molto dall'applicazione. L'occhio umano è in grado di percepire solo 32 livelli di grigio (!).
- palette (scala di colori, look up table)

Risoluzione

512×512 pixels

256×256 pixels

64×64 pixels


N=64


N=32


N=16


N=8


N=4


N=2

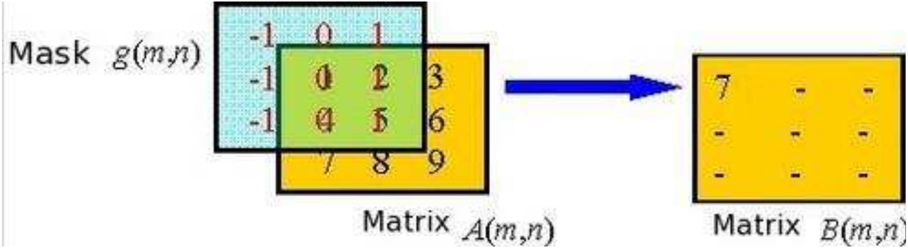
Profondità

Outline

Filtraggio lineare

- Obiettivi svariati tra cui : ridurre il rumore o gli artefatti dalle immagini
- filtrare una immagine corrisponde (nel dominio dell'immagine) ad effettuare una convoluzione dell'immagine I per il filtro h
- $f(n,m) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} h(k,l) I(n-k, m-l)$
- equivale a spostare una finestra di pochi pixel che rappresenta il filtro sulla mia immagine e per ogni pixel rimpiazzare il pixel con la somma dei pixel sotto la maschera moltiplicati per il valore della maschera.

Filtraggio lineare



In `Matlab, dopo aver creato il mio filtro, posso usare il comando conv2(I,h).`

Filtraggio lineare

- Filtro medio:

$$\texttt{hm} = \frac{1}{9}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

- Filtro Gaussiano:
  - il rumore nelle immagini è caratterizzato dalla alte frequenze, un filtro Gaussiano è un filtro di tipo passa-basso.
  - i parametri caratteristici sono la dimensione e σ

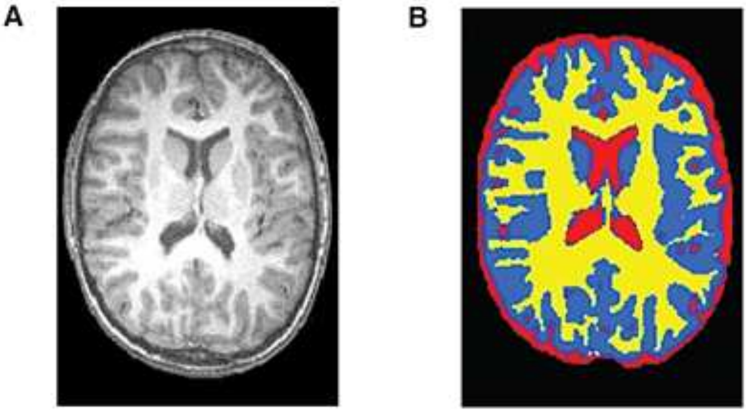
`Implementazione in Matlab`

Outline

Introduzione

SEGMENTAZIONE di immagini:

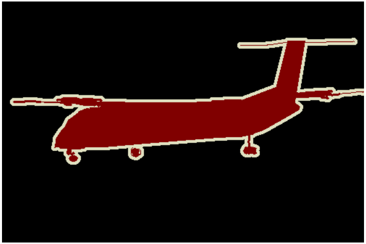
- dividere l'immagine in diverse regioni (N)



Introduzione

SEGMENTAZIONE di immagini:

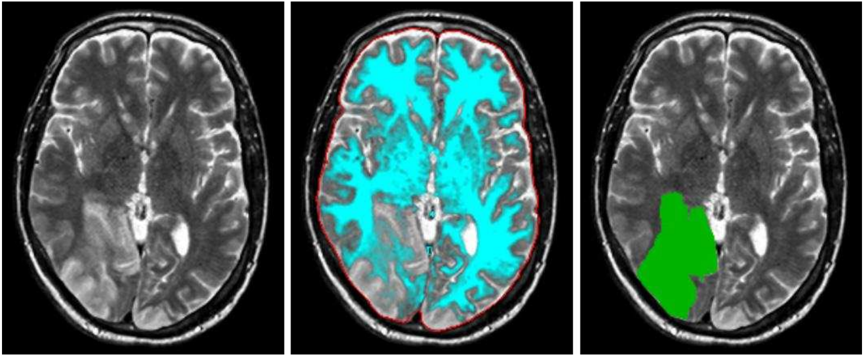
- in particolare dividere l'immagine in una regione di interesse e lo sfondo (tutto quello che non è la mia regione di interesse)



Introduzione

Applicazioni:

- Effettuare delle misure dell'oggetto selezionato (ex. in ambito medico i radiologo misura dimensioni di organi o tumori o altro)



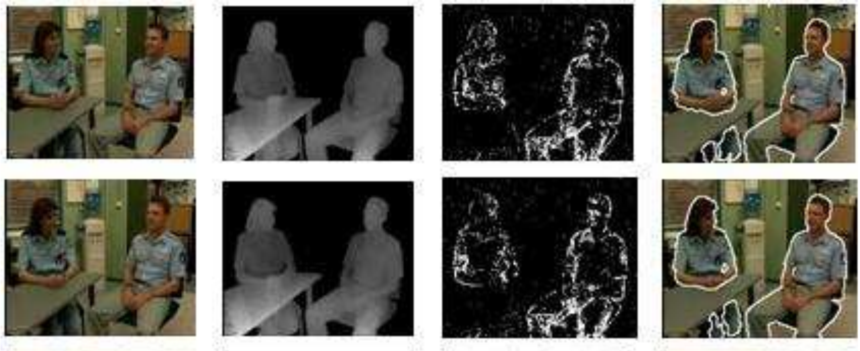
R

magnetica del cervello. In azzurro la materia bianca. In verde la zona tumorale.

Introduzione

Applicazioni:

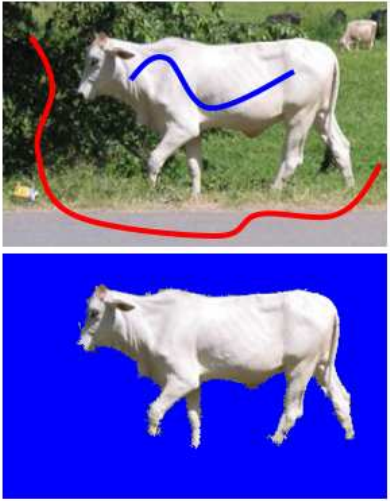
- Riconoscere degli oggetti (ex. videosorveglianza)



Introduzione

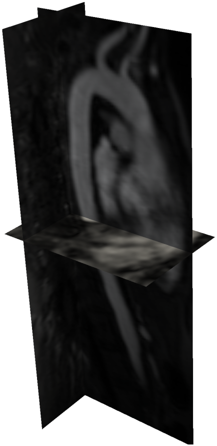
Applicazioni:

- Applicazioni grafiche (ex. modificare lo sfondo in una fotografia)

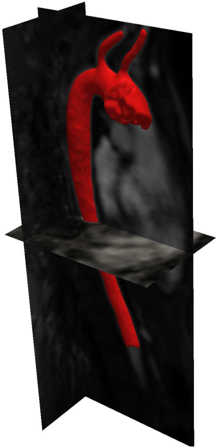


Introduzione

Problema 2D ma sempre più spesso, soprattutto in ambito medico 3D.



risonanza magnetica dell'aorta



estrazione della superficie dell'aorta

Region-based vs. Edge-based methods

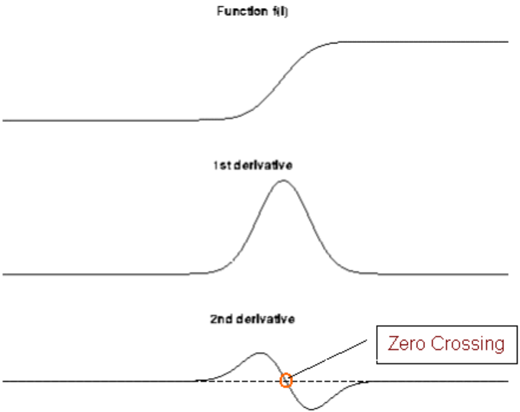
A way to divide segmentation methods:

- Edge-based - the object to be segmented should have its boundary visible in the image, as some sort of prominent edge.
- Region-based - the region of the object in the image should have a different statistic in some feature space, compared to its surroundings.

These functionals can be combined with information on the shape of the region being extracted.

Outline

Edges of an image



edges = maximum of the image gradient in the gradient direction

or

edges = zero crossing of the laplacian

Edges of an image

From a mathematical standpoint, a 2D image can be identified with a domain $\Omega \subset \mathbb{R}^2$ (*i.e.* the image domain) and a function $I(x, y)$ standing for the image intensity distribution over Ω. The image

gradient:

$$\mathbf{G} = \nabla I(x, y) = [\frac{\partial I(x, y)}{\partial x}, \frac{\partial I(x, y)}{\partial y}]$$

Example of edge detector:

$$P = |\nabla I(x, y)| = \sqrt{(\frac{\partial I(x, y)}{\partial x})^2 + (\frac{\partial I(x, y)}{\partial y})^2}$$

or:

$$P = |\nabla G_\sigma(x, y) * I(x, y)|$$

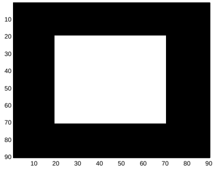
Metodi basati sul gradiente

Numericamente si usano delle maschere di convoluzione:

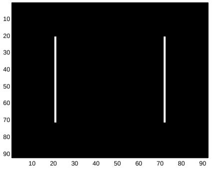
1. Gradiente:

$$\mathtt{gradx} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{pmatrix} \qquad \mathtt{grady} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \end{pmatrix}$$

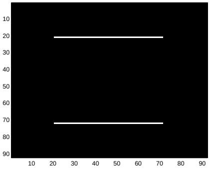
`rect`



`conv2(rect,gradx);`



`conv2(rect,grady);`



Metodi basati sul gradiente

Maschere più elaborate:

- Roberts:

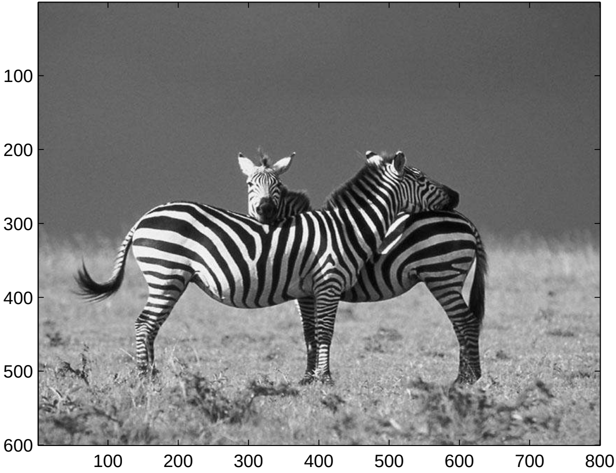
$$\texttt{rob} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

- Prewitt

$$\texttt{prewx} = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \qquad \texttt{prewy} \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

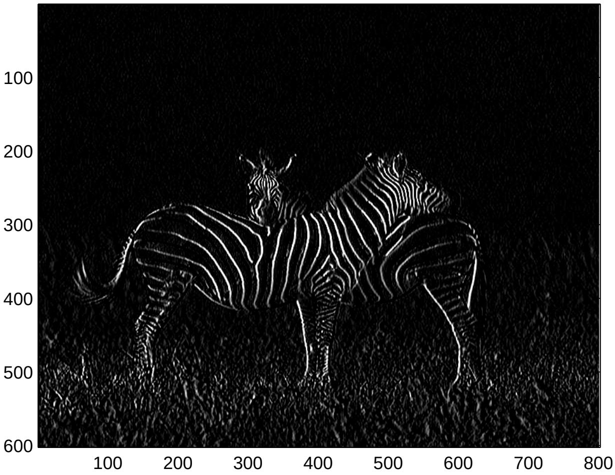
Metodi basati sul gradiente: Prewitt

```
image(zebra);
```



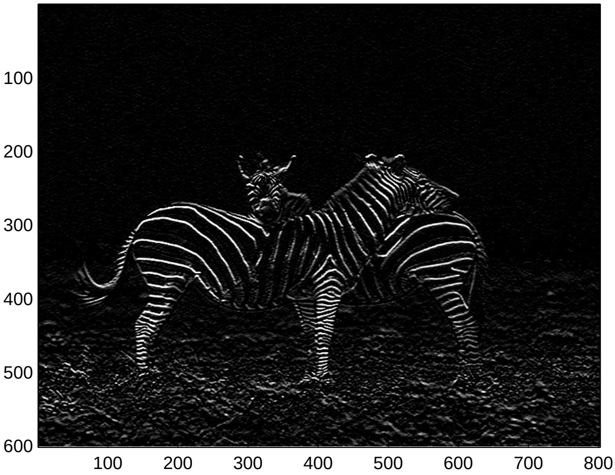
Metodi basati sul gradiente

```
prewxf=conv2(zebra,prewx); image(prewxf);
```



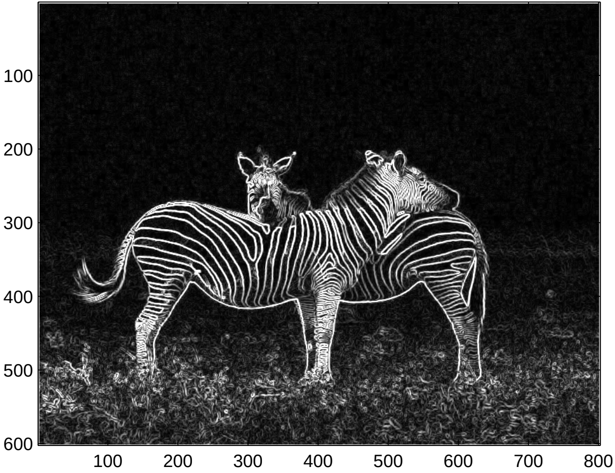
Metodi basati sul gradiente

```
prewyf=conv2(zebra,prewy); image(prewyf);
```



Metodi basati sul gradiente

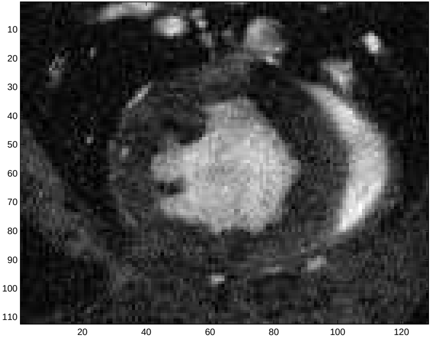
```
prewf=max(abs(prewxf),abs(prewyf)); image(prewf);
```



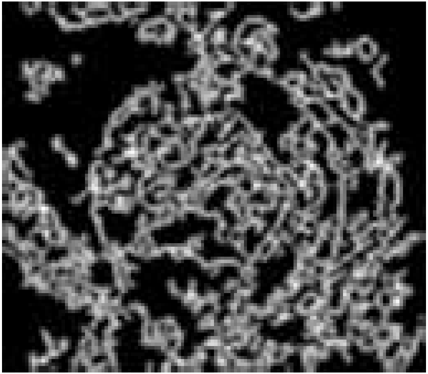
Edges of an image

The gradient of $I(x, y)$ gives an information about the edges

in the medical field...



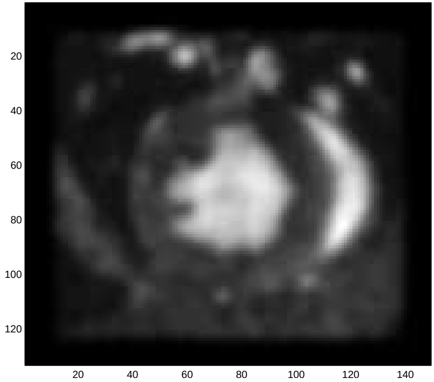
image



gradient

Edges of an image

with smoothing



filtered image

gradient

Outline

Edge-based methods

can be classified in two families:

- Parametric deformable models (Snakes and Balloons):
  - explicit representation of the contour
  - real-time applications
  - no topological changes

Edge-based methods

can be classified in two families:

- Parametric deformable models (Snakes and Balloons):
  - explicit representation of the contour
  - real-time applications
  - no topological changes
- Implicit models (Geodesic Active Contours):
  - implicit representation of the contour as the iso-level 0 of a scalar function in dimension (D+1)
  - no need of contour parametrization
  - topological changes are admitted
  - the problem is formulated in a space of higher dimension

Outline

Parametric deformable models - Snakes

- The contour is represented as a parametric curve $\mathbf{C}(s) = [X(s), Y(s)] : U \mapsto \mathbb{R}^2$, with $U \subset \mathbb{R}$, $s \in [0, 1]$ the arc length.
- the idea is to search for $\mathbf{C}(s)$ which minimize the energy functional E_{total}:

$$E_{total} = E_{internal}(\mathbf{C}(s)) + E_{external}(\mathbf{C}(s))$$

Parametric deformable models - Snakes

Internal energy $E_{internal}(\mathbf{C}(s))$:

Aim: obtain a regular contour

- penalization of curve dimension (curve length);
- penalization of curve irregularities (curvature).

$$E_{internal}(\mathbf{C}(s)) = \int_0^1 \underbrace{\alpha|\mathbf{C}_s(s)|^2}_{\text{curve length}} + \underbrace{\beta|\mathbf{C}_{ss}(s)|^2}_{\text{curvature}} ds$$

Parametric deformable models - Snakes

External energy $E_{external}(\mathbf{C}(s))$:

Aim: stop the curve on the image contours:

$$E_{external}(\mathbf{C}(s)) = \int_0^1 P(\mathbf{C}(s))ds$$

where:

$$P(x, y) = -w|\nabla I(x, y)|^2$$

or

$$P(x, y) = -w|\nabla G_\sigma(x, y) * I(x, y)|^2$$

$P(x, y)$ define an edge map

Parametric deformable models - Snakes

In order to minimize the functional $E(\mathbf{C})$, the active contour $\mathbf{C}$ must satisfy the Euler-Lagrange equation:

$$\alpha \frac{\partial^2 \mathbf{C}(s)}{\partial s^2} - \beta \frac{\partial^4 \mathbf{C}(s)}{\partial s^4} - \nabla P(\mathbf{C}) = 0$$

i.e. two separate equations:

$$\alpha \frac{\partial^2 X(s)}{\partial s^2} - \beta \frac{\partial^4 X(s)}{\partial s^4} - \frac{\partial P(\mathbf{C})}{\partial x} = 0$$

$$\alpha \frac{\partial^2 Y(s)}{\partial s^2} - \beta \frac{\partial^4 Y(s)}{\partial s^4} - \frac{\partial P(\mathbf{C})}{\partial y} = 0$$

Pseudocode for 2D balloon:

- `initialize of contour` $\mathbf{C}^0$
- `compute the external terms (edge map)`
- `for i = 0:N iterations`
  - `compute` $F_{img}(\mathbf{C}^i)$ `:  interpolate the edge map in` $\mathbf{C}^i$
  - `compute the normals on` $\mathbf{C}^i$
  - `set` $F_{ext}(\mathbf{C}^i) = F_{img} + F_{balloon}$
  - `matrix assembly` A`:` $A_2 = (A + \gamma I_d)$
  - `solve` $A_2 \mathbf{C}^{i+1} = \gamma \mathbf{C}^i + F_{ex}(\mathbf{C}^i)$
  - `contour interpolation` $\mathbf{C}^{i+1}$ `to obtain the new points at distance` h

Remarks:

- the new formulation needs the normal at each iteration see Matlab example...

Outline

Level Sets

- 2D case: $\mathbf{C}(t) : \mathbb{R} \times \mathbb{R}^+ \to \mathbb{R}^2 \Leftrightarrow$ zero-level iso-contour $\mathbf{C}(t) = \{\mathbf{x} \in \mathbb{R}^2 : \phi(\mathbf{x}, t) = 0\}$ of a scalar function $\phi(\mathbf{x}, t) : \mathbb{R}^2 \times \mathbb{R}^+ \to \mathbb{R}$;
- 3D case: $\mathbf{S}(t) : \mathbb{R}^2 \times \mathbb{R}^+ \to \mathbb{R}^3 \Leftrightarrow$ zero-level iso-surface $\mathbf{S}(t) = \{\mathbf{x} \in \mathbb{R}^3 : \phi(\mathbf{x}, t) = 0\}$ of a scalar function $\phi(\mathbf{x}, t) : \mathbb{R}^3 \times \mathbb{R}^+ \to \mathbb{R}$;

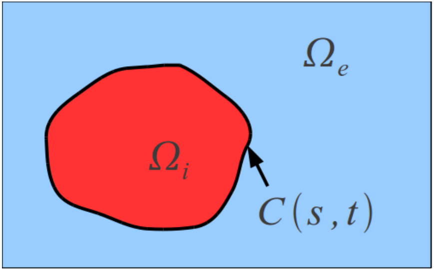
Level Sets

- 2D case: $\mathbf{C}(t) : \mathbb{R} \times \mathbb{R}^+ \rightarrow \mathbb{R}^2 \Leftrightarrow$ zero-level iso-contour $\mathbf{C}(t) = \{\mathbf{x} \in \mathbb{R}^2 : \phi(\mathbf{x}, t) = 0\}$ of a scalar function $\phi(\mathbf{x}, t) : \mathbb{R}^2 \times \mathbb{R}^+ \rightarrow \mathbb{R}$;
- 3D case: $\mathbf{S}(t) : \mathbb{R}^2 \times \mathbb{R}^+ \rightarrow \mathbb{R}^3 \Leftrightarrow$ zero-level iso-surface $\mathbf{S}(t) = \{\mathbf{x} \in \mathbb{R}^3 : \phi(\mathbf{x}, t) = 0\}$ of a scalar function $\phi(\mathbf{x}, t) : \mathbb{R}^3 \times \mathbb{R}^+ \rightarrow \mathbb{R}$;

ϕ can be modeled as a signed distance function:

$$\begin{cases} \phi(\mathbf{x}, t) = -D_{\mathbf{C}}(\mathbf{x}), \mathbf{x} \in \Omega_i \\ \phi(\mathbf{x}, t) = +D_{\mathbf{C}}(\mathbf{x}), \mathbf{x} \in \Omega_e \\ \phi(\mathbf{x}, t) = 0, \mathbf{x} \in \mathbf{C} \end{cases}$$

where $D_{\mathbf{C}}(\mathbf{x}) = min\{|\mathbf{x} - \mathbf{C}(\mathbf{x})|\}$



Geodesic Active Contours

- finally the evolution of $\phi(\mathbf{x}, t)$ is described by a PDE of the kind [Caselles,Kimmel,Sapiro, 1995]]

$$\phi_t = -w_1 G(\mathbf{x})|\nabla\phi| + w_2 H(\mathbf{x})|\nabla\phi| + w_3 \nabla P(\mathbf{x}) \cdot \nabla\phi$$

- $G(\mathbf{x})$ can be defined as $G(\mathbf{x}) = 1/(1 + |\nabla I(\mathbf{x})|)$ so that inflation speed is higher where image gradient is lower;
- $P(\mathbf{x})$ can be defined as $P(\mathbf{x}) = -|\nabla I(\mathbf{x})|$ which gives origin to the valleys of the attraction potential

Geodesic Active Contours

Considerations:

- The equation describes a deformable surface, such as the balloon, embedded as a level set of a scalar field evolving in time;
- The great advantage respect to the balloon is the lack of parametrization in level sets formulation;
- The model can therefore freely deform and change its topology without the need of any particular care;
- When dealing with 3D images, the level set equation can be directly solved on the image regular grid by one of classic numerical methods, such as finite differences;
- this method is implemented in vmtk (www.vmtk.it).

Outline

Region based methods: Chan and Vese

$$E(\mathbf{C}, k_1, k_2) = \underbrace{\nu L(\mathbf{C}) + \mu A(\mathbf{C})}_{\text{regolarizzazione}} + \underbrace{\lambda_1 \int_{\Omega_i} |I - k_1|^2 d\Omega + \lambda_2 \int_{\Omega_e} |I - k_2|^2 d\Omega}_{\text{vincolo di omogeneità}}$$

$k_1 =$ valore medio all'interno di $\mathbf{C}$
$k_2 =$ valore medio all'esterno di $\mathbf{C}$
$L(\mathbf{C}) =$ lunghezza del contorno $\mathbf{C}$
$A(\mathbf{C}) =$ area racchiusa in $\mathbf{C}$

- Non cerco solo $\mathbf{C}$, ma anche le due k_1, k_2
- Cerco di suddividere il dominio totale in due regioni il più possibili omogenee + solita regolarità della curva cercata

Region based methods: Chan and Vese and Level Sets

- The same equation can be described by a level set evolution equation.
- The resulting PDE for updating the level set ϕ is:

$$\phi_t = [\lambda \nabla^T \left(\frac{\nabla \phi}{|\nabla \phi|} \right) - ((I - \nu_1)^2 + (I - \nu_2)^2)] \delta_\varepsilon(\phi)$$

References

1. Kass M., Witkin A. and Terzopoulos D. *Snakes: Active contour models.* International journal of computer vision 1988, 1(4):321–331
2. Cohen L.D. and Cohen I. *Finite-element methods for active contour models and balloons for 2-D and 3-D images.* Pattern Analysis and Machine Intelligence, IEEE Transactions on 1993, 15(11):1131–1147
3. Malladi R., Sethian J.A. and Vemuri B.C. *Shape modeling with front propagation: A level set approach.* Pattern Analysis and Machine Intelligence, IEEE Transactions on 1995, 17(2):158–175
4. Chan T.F. and Vese L.A. *Active contours without edges.* Image Processing, IEEE Transactions on 2001, 10(2):266–277