

Figura 3.5: Segmentazione di una struttura anatomica da un'immagine ecocardiografica: segmentazione dell'atrio destro.

grafica. In entrambi i casi, come funzione iniziale  $\phi_0(x, y)$  si assume la distanza dei punti dell'immagine dal punto di riferimento,  $(x_0, y_0)$ , scelto all'interno della parte che si desidera segmentare [?] (Fig.3.5, Fig.3.6 in alto a sinistra). Tale funzione è data dalla seguente equazione:

$$\phi_d(x, y) = \sqrt{(x - x_0)^2 + (y - y_0)^2} + R, \quad (3.40)$$

dove  $R$  è il raggio (in tutti gli esperimenti è stato scelto  $R = 20$ ).

Il valore del parametro  $\mu$  deve essere elevato sia per le dimensioni della zona da segmentare, sia per garantire la stabilità della soluzione ( $\mu = 2$ ). Inoltre, poiché siamo interessati a segmentare una particolare parte dell'immagine, è necessario applicare, in alcuni passi dell'algoritmo, una particolare procedura. Durante tutta l'evoluzione del fronte, non si devono sviluppare regioni nè troppo piatte nè troppo ripide. In altre parole deve essere sempre soddisfatta la condizione:

$$|\nabla\phi| = 1. \quad (3.41)$$

La tecnica, che garantisce il soddisfacimento di tale proprietà, è comunemente detta *procedura di reinizializzazione*. Sussman, Smereka, e Osher [?] hanno proposto la seguente equazione differenziale:

$$\begin{cases} \psi_\tau = \text{sign}(\phi)(1 - |\nabla\psi|) \\ \psi(x, 0) = \phi(x, t) \end{cases} \quad (3.42)$$

dove  $\psi$  è una funzione ausiliaria che serve per rettificare la funzione level set in particolari istanti durante l'evoluzione, e  $\text{sign}(\phi)$  è la funzione segno. Tale

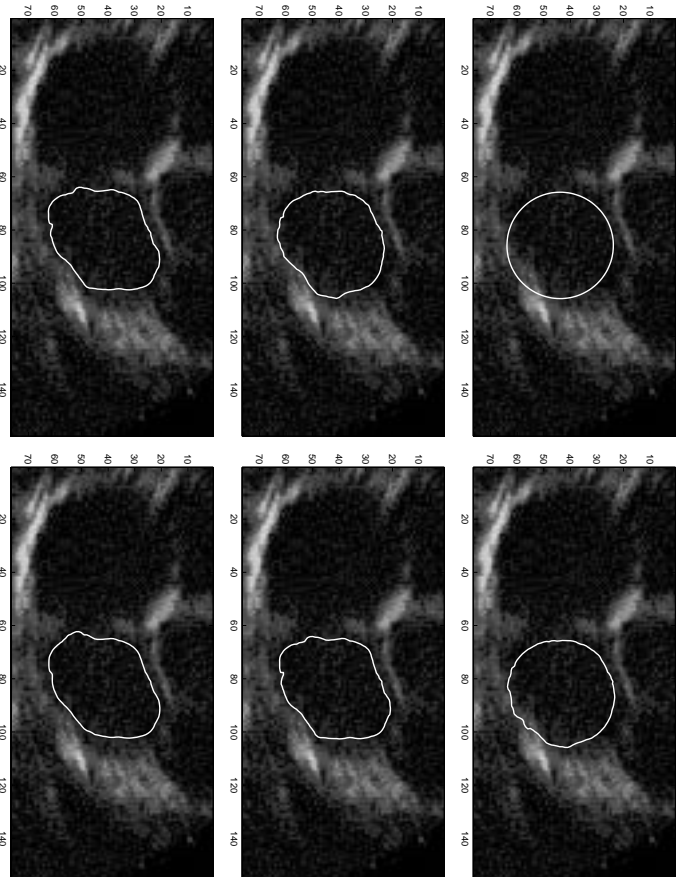


Figura 3.6: Segmentazione di una struttura anatomica da un'immagine ecocardiografica: segmentazione dell'atrio sinistro.

funzione controlla il fluire delle informazioni: se  $\phi$  è negativa, il fronte si muove in una direzione, se è positiva, nell'altra. Riassumendo: assegnata una qualunque funzione iniziale  $\phi$ , risolvendo la precedente equazione, si ottiene una soluzione  $\psi(x, t)$  che gode della proprietà  $|\nabla\psi| = 1$ .

È importante sottolineare che la soluzione asintotica  $\psi(x, t)$  dell'Eq.(3.42) ha lo stesso livello di zero della  $\phi(x, t)$ . Si evita così la formazione di ulteriori picchi. L'approssimazione numerica è ottenuta applicando le stesse tecniche utilizzate per l'equazione del moto [?]. In questo modo la soluzione dell'Eq.(3.30) è sicuramente una soluzione di minimo relativo.

```
% Codice Matlab della procedura di reinizializzazione
function f=rein(Mat)
[ m n ]=size(Mat);
f=sign(Mat).*(1-(G(Mat)));
```

Presentiamo una serie di immagini che sottolineano la modalità con la quale avviene l'evoluzione della curva. La soluzione dell'algoritmo, cioè quella con la massima probabilità, è rappresentata in basso a destra. La condizione di stop si ha quando il funzionale  $I$  (Eq.(3.12)), somma del logaritmo della probabilità dell'insieme dei punti all'interno della curva e del logaritmo della probabilità dell'insieme dei punti all'esterno, mostra un plateau.

Presentiamo inoltre un esempio di una sezione 2D ottenuta attraverso il dispositivo Volumetrics (Fig.3.7). L'obiettivo è la segmentazione della camera ventricolare. È importante sottolineare come le dimensioni della curva iniziale (3.40)

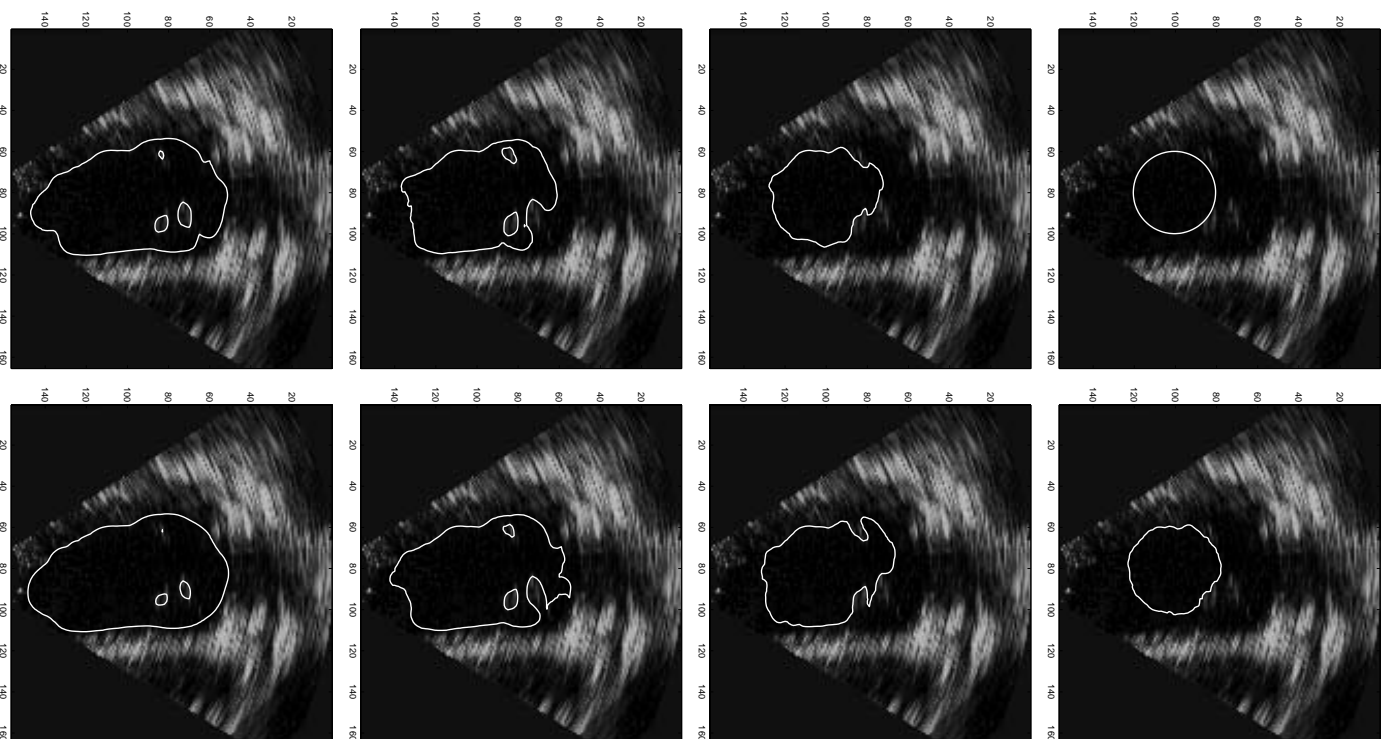


Figura 3.7: Segmentazione di una struttura anatomica da una sezione Volumetric: segmentazione della camera ventricolare.

non costituiscono una limitazione per l'area che si vuole segmentare. In questo caso infatti la curva iniziale è molto più piccola della zona di interesse. Inoltre, se all'interno dell'area vengono riconosciute strutture estranee, la curva finale può non essere unica: è possibile cioè segmentare l'oggetto al suo interno. Anche in questo caso, per le stesse ragioni delle due immagini precedenti,  $\mu$  deve assumere un valore elevato ( $\mu = 2$ ).

La tecnica ecografica viene frequentemente applicata anche per il controllo e il monitoraggio dello stato di crescita del feto. La Fig.3.8 è un esempio di una ecografia fetale. In questo caso, l'approccio è differente: la funzione iniziale non è una singola curva ma una famiglia di curve (Fig.3.8 in alto a destra). L'equazione che le rappresenta è la seguente:

$$\phi_{nd}(x, y) = \sum_i \sum_j \phi_d(x_0 + i\Delta x, y_0 + j\Delta y), \quad (3.43)$$

dove  $\phi_d$  è la funzione distanza (3.40). La soluzione che si ottiene rappresenta un minimo globale; non è quindi necessario applicare la tecnica di reinizializzazione. Il parametro  $\mu$  deve assumere un valore più piccolo rispetto alle altre immagini ecocardiografiche poiché maggiori sono i dettagli che si vogliono riconoscere ( $\mu = 0.1$ ). Si può osservare che dopo poche iterazioni l'algoritmo riesce a riconoscere correttamente l'area da segmentare (Fig.3.8 quarta immagine), e molto rapidamente, determina il contorno finale.

Il vantaggio principale, che deriva dall'aver considerato l'Eq.(3.43) come funzione iniziale, consiste appunto nella possibilità di riconoscere chiaramente la forma della struttura da segmentare dopo pochissime iterazioni (al massimo 10-20 iterazioni). È importante inoltre sottolineare che il risultato della segmentazione risulta essere indipendente dal punto di riferimento ( $x_0, y_0$ ), fondamentale invece per segmentare una particolare struttura nel caso in cui, nell'immagine, siano presenti altre strutture con lo stesso livello di grigio. L'unico svantaggio di tale procedura, è il maggior tempo impiegato dall'algoritmo per costruire tutte le curve iniziali. Tale svantaggio però è solamente iniziale, poiché successivamente l'evoluzione procede molto rapidamente.

### 3.2.5 Immagini con distribuzione di probabilità di Poisson

#### Immagini angiografiche

Presentiamo due esempi di immagini angiografiche: la visualizzazione di un tratto di albero arterioso (Fig.3.9) e la visualizzazione di un tratto dell'albero carotideo (Fig.3.10). Tali figure sono state ottenute ritagliando una porzione di una DSA (Digital Subtraction Angiography).

I pixel sono considerati variabili random con distribuzione di probabilità di Poisson [?]. La funzione iniziale è una semplice funzione distanza  $\phi_d(x, y)$  (3.40): il punto di riferimento può essere scelto ovunque all'interno dell'immagine. Anche in questo caso, poiché vogliamo segmentare una determinata parte dell'immagine, è necessario considerare, in alcuni passi dell'algoritmo, insieme all'Eq.(3.36), l'equazione di reinizializzazione (3.42). Poiché nelle immagini sono presenti numerose ramificazioni, è necessario che il valore del parametro  $\mu$  sia piccolo ( $\mu = 0.002$ ). In figura sono presentati diversi frame dell'evoluzione: l'algoritmo è in grado di riconoscere la struttura complessa dei vasi sanguinei ricostruendo perfettamente in un caso l'albero arterioso e nell'altro l'albero

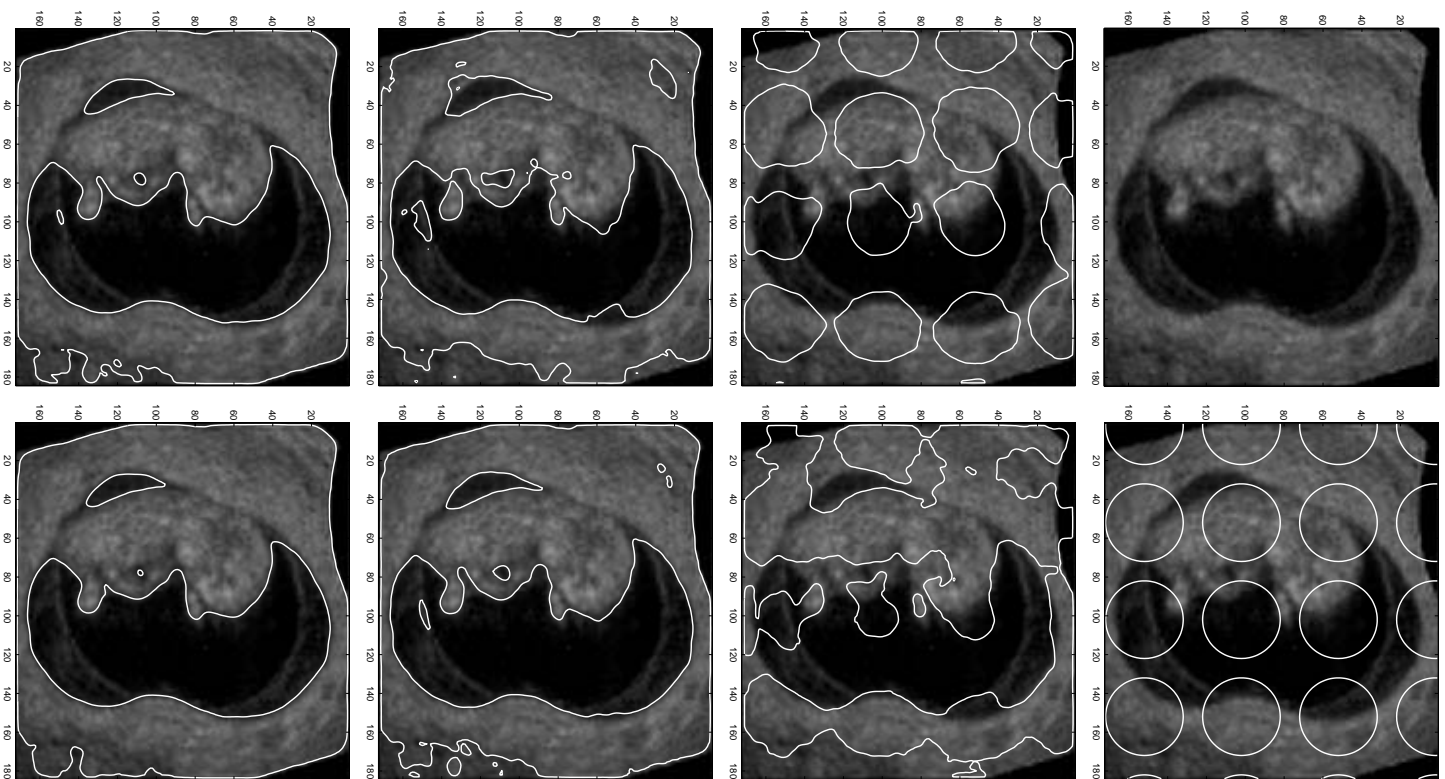


Figura 3.8. Ecografia fetale.

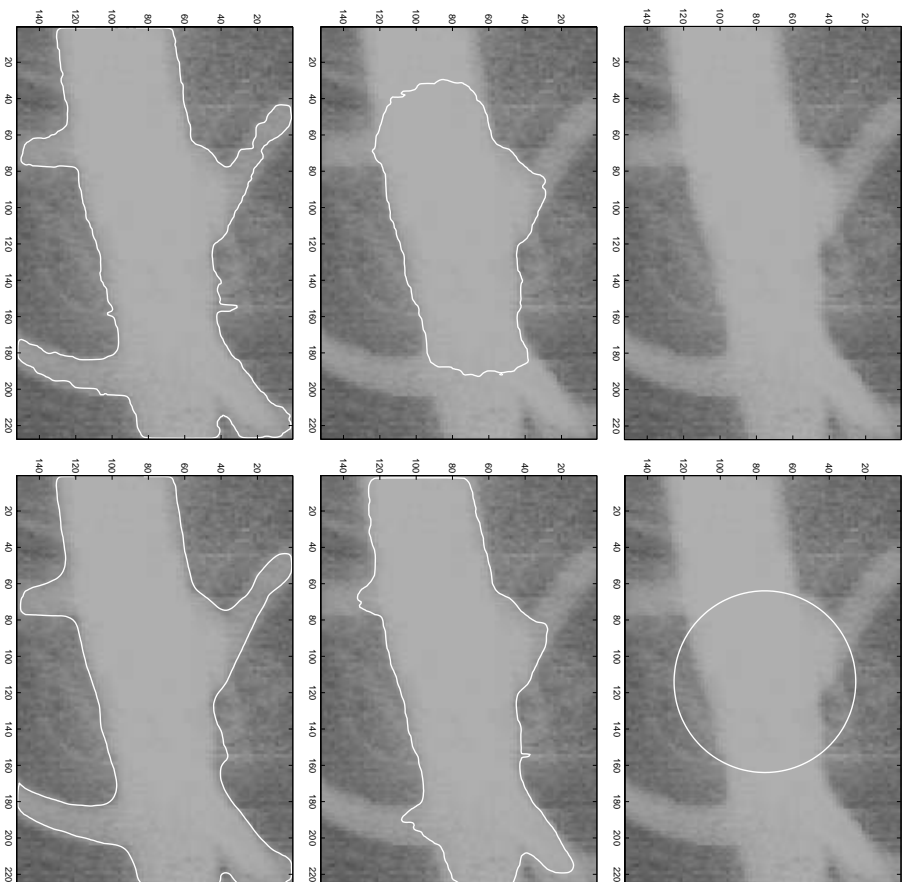


Figura 3.9: Segmentazione di un tratto di albero arterioso da un'immagine angiografica.

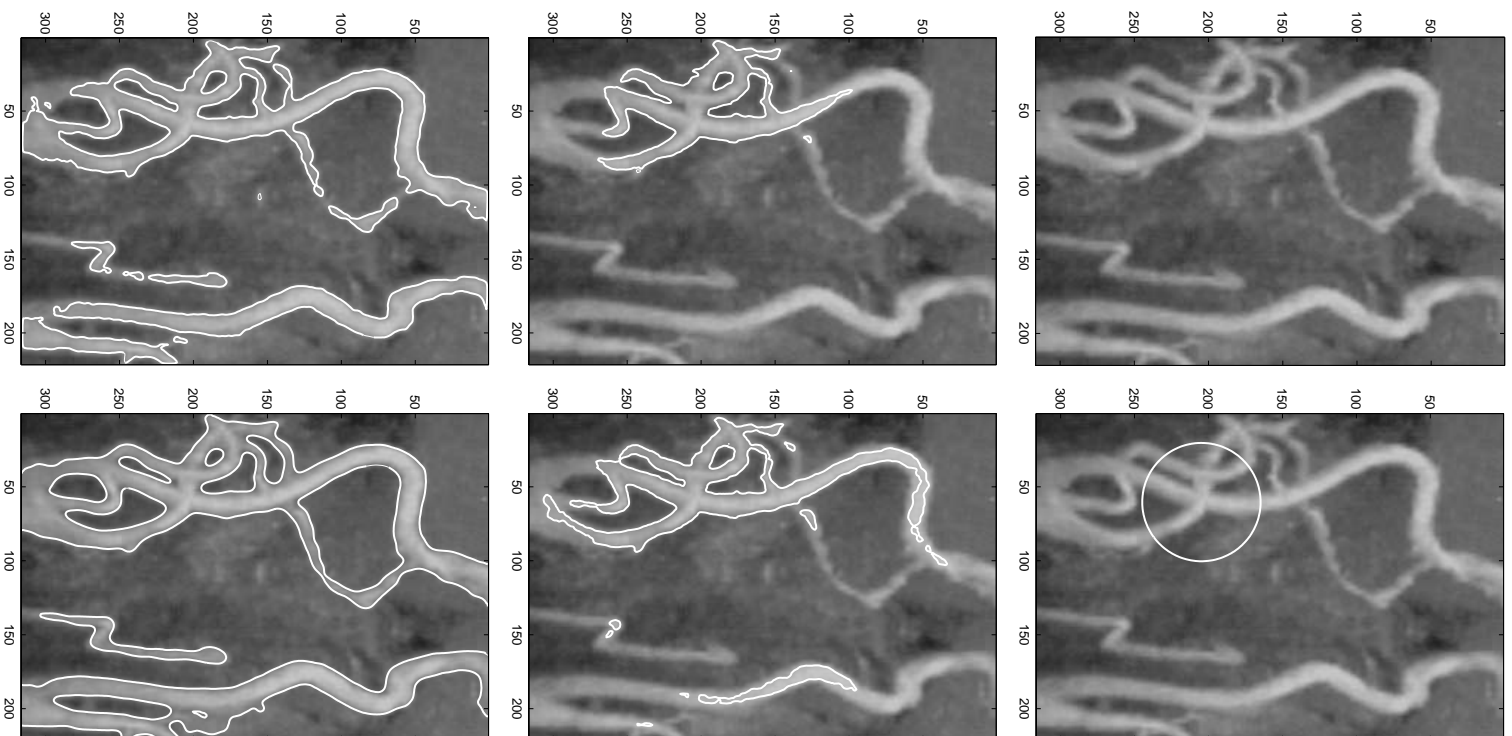


Figura 3.10: Segmentazione di un tratto dell'albero carotideo da un'immagine angiografica.

carotideo (Fig.3.9, Fig.3.10 in basso a destra). È importante sottolineare che la segmentazione avviene correttamente anche se la curva iniziale ha dimensioni molto minori rispetto all'area da segmentare; inoltre il punto di riferimento può essere scelto ovunque all'interno di tale struttura.

### Immagini CT

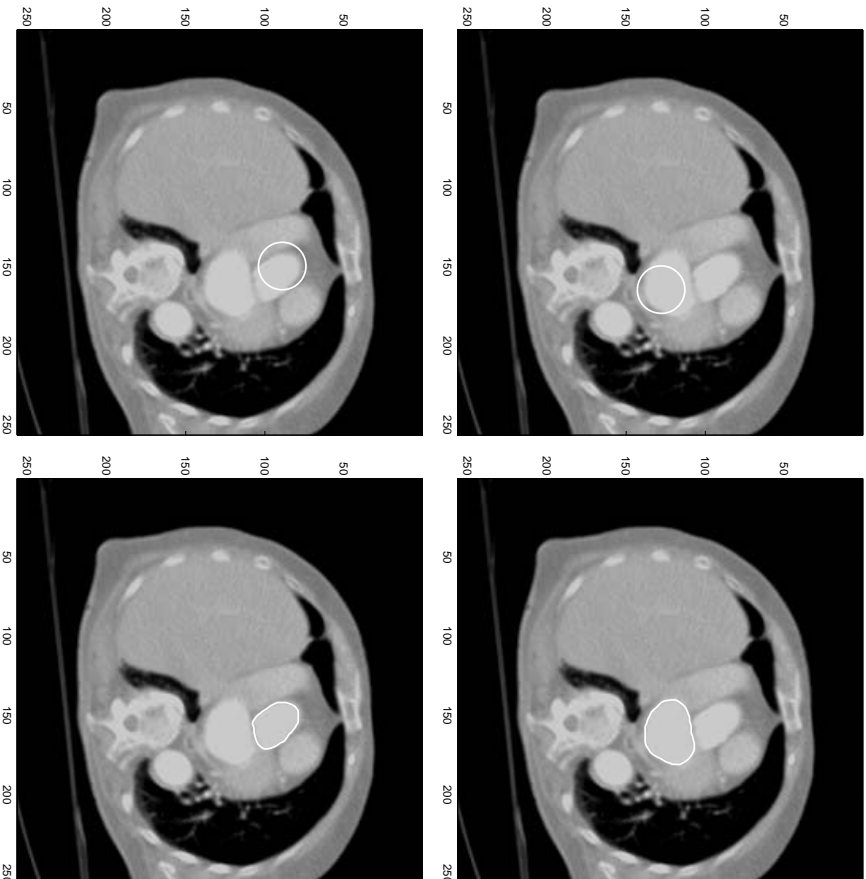


Figura 3.11: Esempio di minimi locali: segmentazione del ventricolo sinistro e del ventricolo destro da un'immagine CT.

Presentiamo un esempio di un'immagine CT nella quale i pixel sono considerati variabili random con distribuzione di probabilità di Poisson [?]. In Fig.3.11 e in Fig.3.12, mostriamo la segmentazione di diverse strutture anatomiche. Nel primo caso, (Fig.3.11), il modello si propone l'obiettivo di segmentare una particolare struttura: il ventricolo sinistro (Fig.3.11 in alto), e il ventricolo destro (Fig.3.11 in basso). Il parametro  $\mu$  deve assumere un valore abbastanza piccolo viste le dimensioni di tali strutture (in questo caso abbiamo scelto  $\mu = 0.02$ ). La funzione iniziale è la funzione distanza (3.40): il punto di riferimento,  $(x_0, y_0)$ , dovrà essere scelto all'interno della parte da segmentare. Come negli altri esperimenti fin'ora eseguiti, poiché cerchiamo un minimo relativo, cioè vogliamo



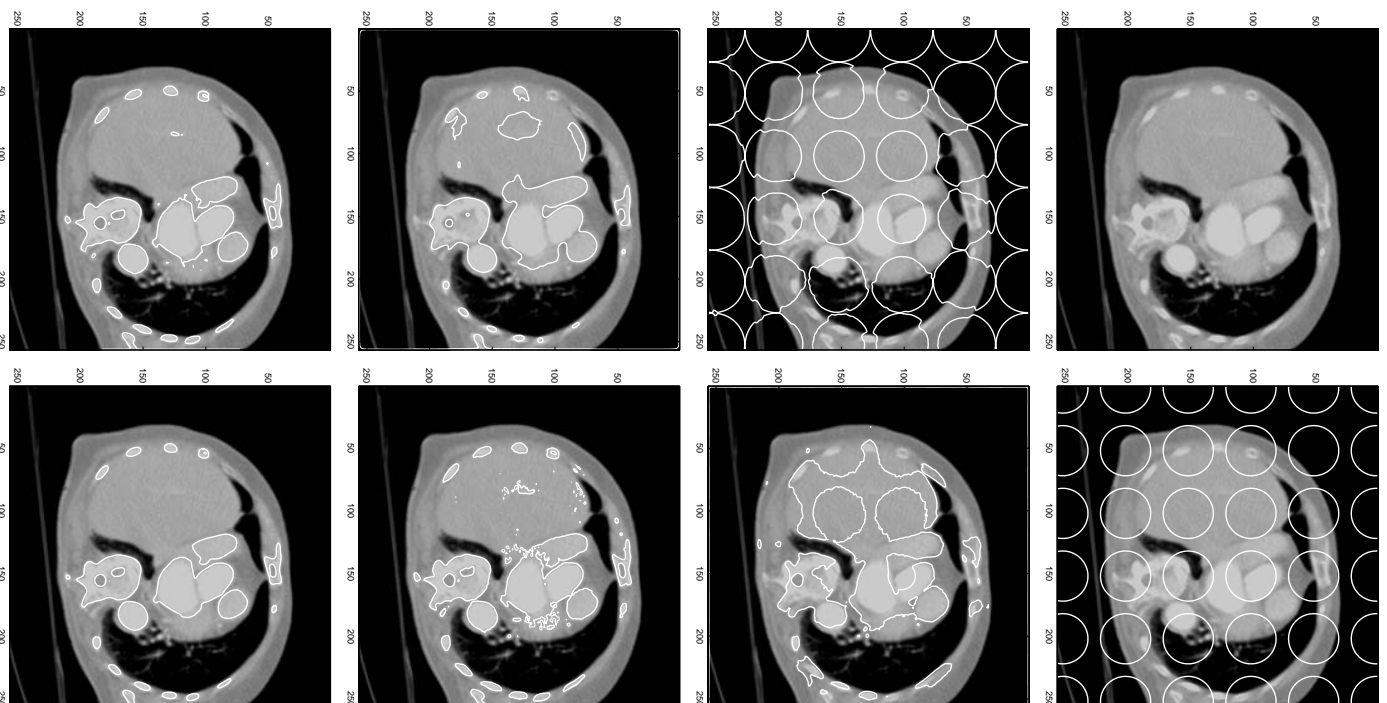


Figura 3.12: Esempio di minimo globale: segmentazione di strutture anatomiche da un immagine CT.

segmentare una particolare struttura dell'immagine, è necessario considerare insieme all'Eq.(3.36), l'equazione di reinizializzazione (3.42). Il risultato è visualizzato in Fig.3.11, rispettivamente, in alto a destra per il ventricolo sinistro e in basso a destra per il ventricolo destro. Nel secondo caso, (Fig.3.12), il modello si pone l'obiettivo di segmentare diverse strutture dell'immagine. La funzione iniziale, poiché si vuole ottenere un minimo globale, è la funzione distanza multipla (3.43). È necessario quindi, attribuire al parametro  $\mu$  un valore minore rispetto al primo caso; in particolare si assume  $\mu = 0.002$ . In questo caso non è necessario applicare la procedura di reinizializzazione. L'algoritmo è in grado di riconoscere, dopo poche iterazioni, le strutture di interesse (Fig.3.12 quarta immagine) e di fornire l'esatta soluzione del problema. Tale soluzione è visualizzata in Fig.3.12 in basso a destra.

### Immagini MR

Presentiamo due esempi di immagini ottenute applicando la Risonanza Magnetica. In entrambe le immagini, i pixel possono essere considerati variabili random con distribuzione di probabilità di Poisson [?].

La Fig.3.13 è un esempio di applicazione di questa tecnica al campo neurologico. L'immagine rappresenta una sezione del cervello: l'obiettivo dell'algoritmo (3.36) è la segmentazione della materia grigia. La funzione iniziale è la funzione distanza multipla (3.43). Dopo poche iterazioni l'algoritmo riconosce correttamente l'area desiderata; per visualizzare tutte le singole strutture (solchi) è necessario che il parametro  $\mu$  assuma un valore piccolo (in questo caso  $\mu = 0.001$ ).

Con questa tecnica è anche possibile visualizzare, con un grado di accuratezza elevato, strutture tumorali situate anche a notevole profondità. La Fig.3.14 rappresenta uno stadio avanzato di un tumore gastrico: presenta una struttura molto frastagliata con numerose ramificazioni. L'immagine è stata ottenuta attraverso l'iniezione di un liquido di contrasto. In questo caso sarebbe stato indifferente considerare come funzione iniziale la funzione distanza (3.40) o la funzione distanza multipla (3.43), poiché non vi sono altre strutture che possono essere erroneamente segmentate. Per ragioni di efficienza computazionale, cioè per ottenere una soluzione nel minor tempo possibile, abbiamo adottato la funzione distanza multipla che permette di riconoscere, già dopo pochissime iterazioni, l'intera struttura. Al fine di visualizzare correttamente tutte le ramificazioni, è necessario assegnare al parametro  $\mu$  un valore piccolo ( $\mu = 0.001$ ). La soluzione corrispondente alla segmentazione con la massima probabilità è presentata in Fig.3.14 in basso a destra.

### Immagini mammografiche

In Fig.3.15 e Fig.A.13, sono visualizzati rispettivamente, un carcinoma e una cisti dalla mammella. I pixel sono considerati variabili random con distribuzione di probabilità di Poisson [?]. L'obiettivo del modello è la segmentazione di tali strutture al fine di derivare importanti informazioni mediche, quali la loro forma e dimensione. Come funzione iniziale si sceglie la funzione distanza (3.40) e si applica l'Eq.(3.36) insieme all'equazione di reinizializzazione (3.42). Quest'ultima equazione permette la segmentazione solamente della zona di interesse in pedendo la rilevazione di ulteriori minimi (picchi della funzione level set) diversi da

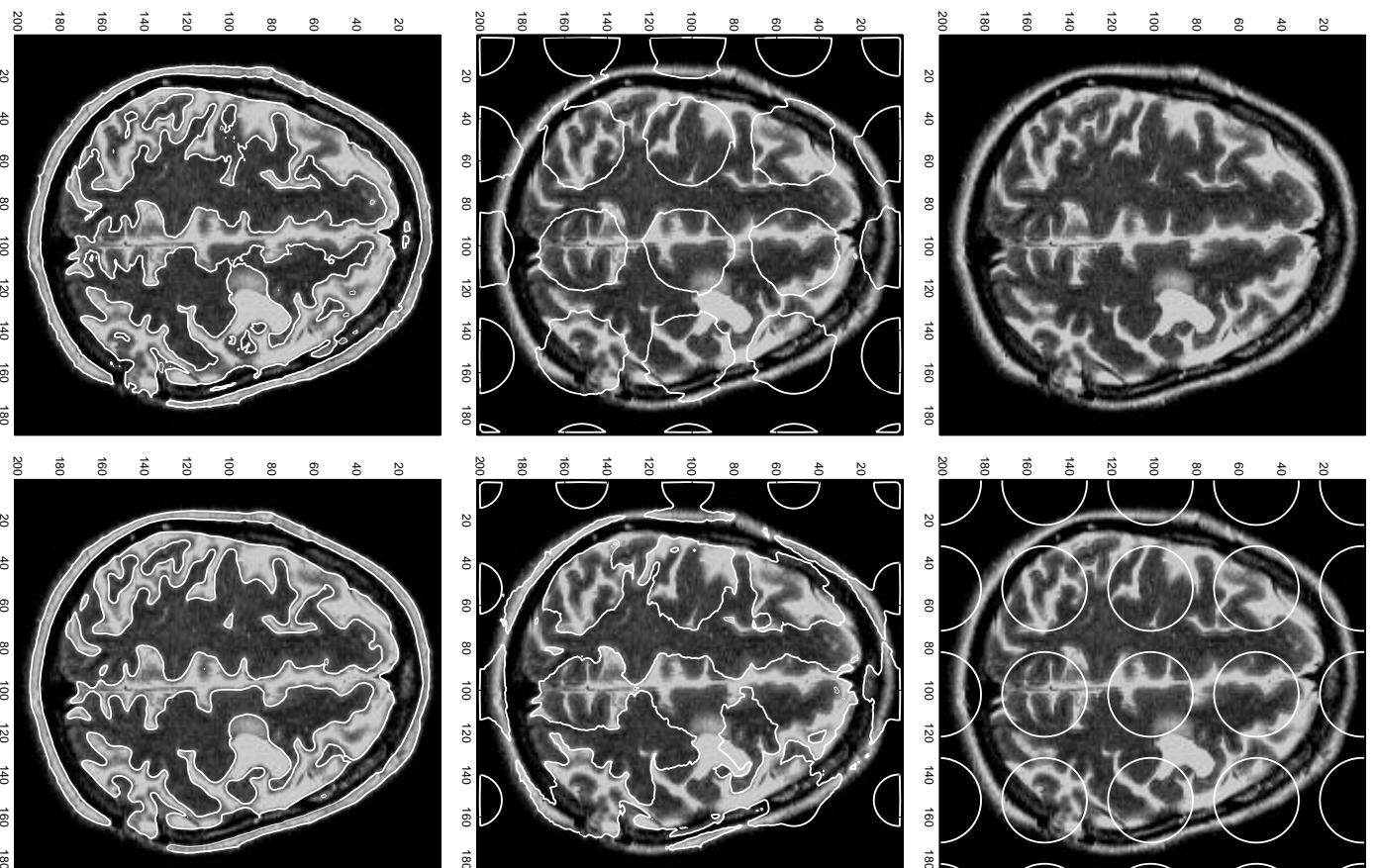


Figura 3.13: Segmentazione di strutture anatomiche del cervello da un'immagine MR: segmentazione della materia grigia.

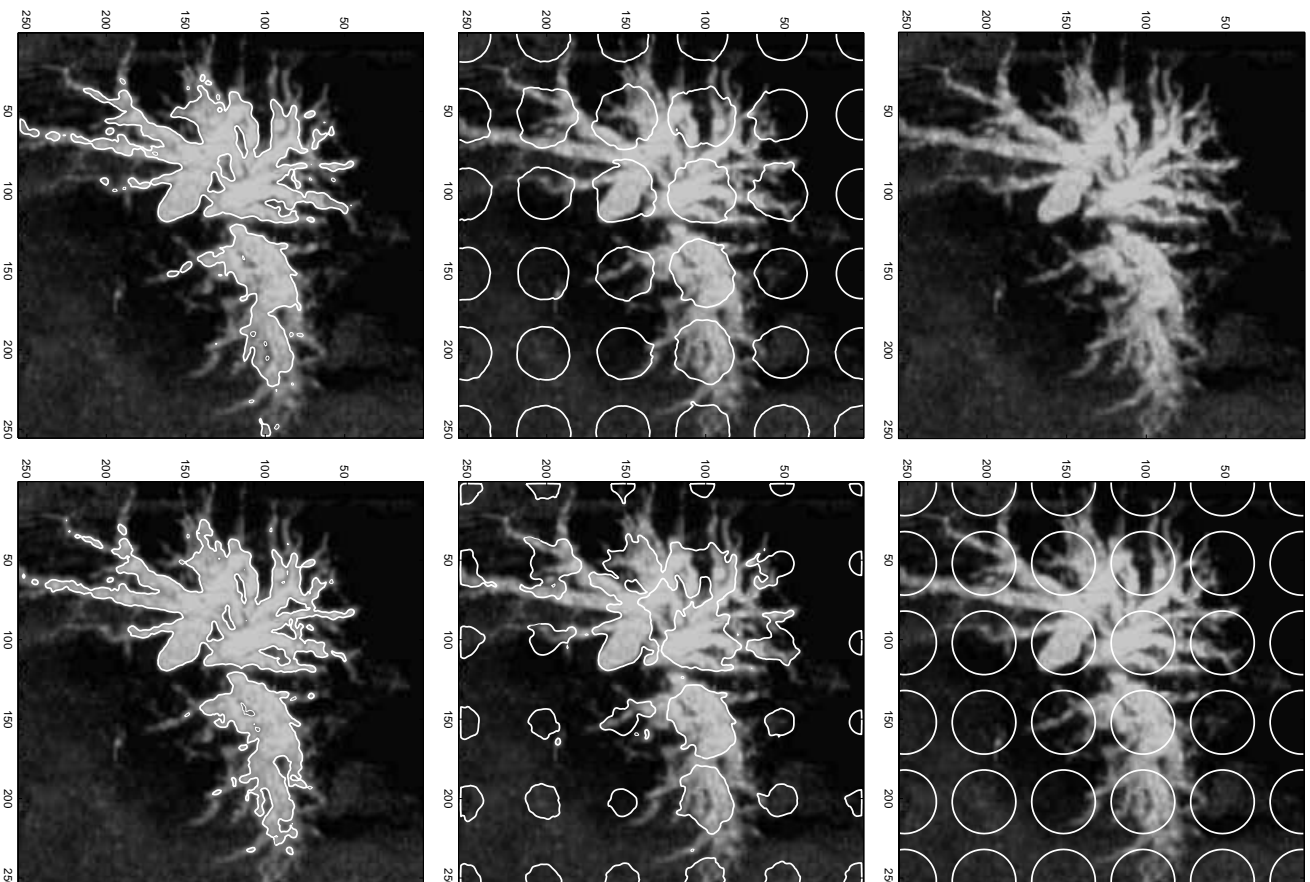


Figura 3.14: Segmentazione di un tumore gastrico da un'immagine MR con mezzo di contrasto.

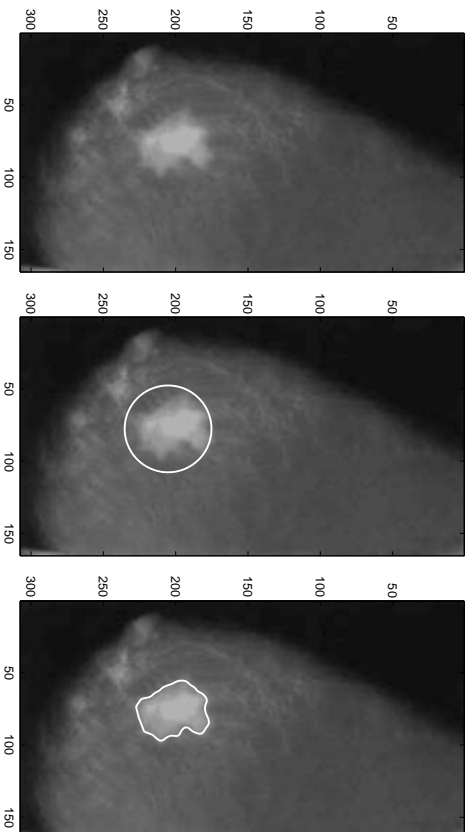


Figura 3.15: Segmentazione di un tumore al seno da un'immagine mammografica.

quello cercato. Per le dimensioni delle strutture, è necessario che il parametro  $\mu$  assuma un valore abbastanza piccolo ( $\mu = 0.05$ ).

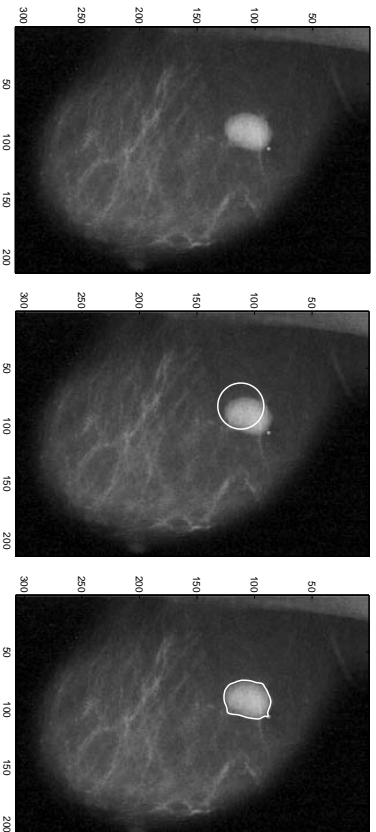


Figura 3.16: Segmentazione di un nodulo al seno da un'immagine mammografica.

### 3.2.6 Immagini con distribuzione di probabilità di Gauss

La Fig.3.17 appartiene alle oltre 8000 immagini digitali a disposizione di tale progetto. In questo caso il nostro metodo si pone l'obiettivo di individuare, nel modo più preciso possibile, il contorno del melanoma in modo da poter osservare cambiamenti, di forma e dimensioni rispetto ad un'immagine precedente, che possano fare pensare ad una neoplasia maligna.

I pixel sono considerati variabili random con distribuzione di probabilità di Gauss [?]. In questo caso, poiché si cerca un minimo relativo, si sceglie

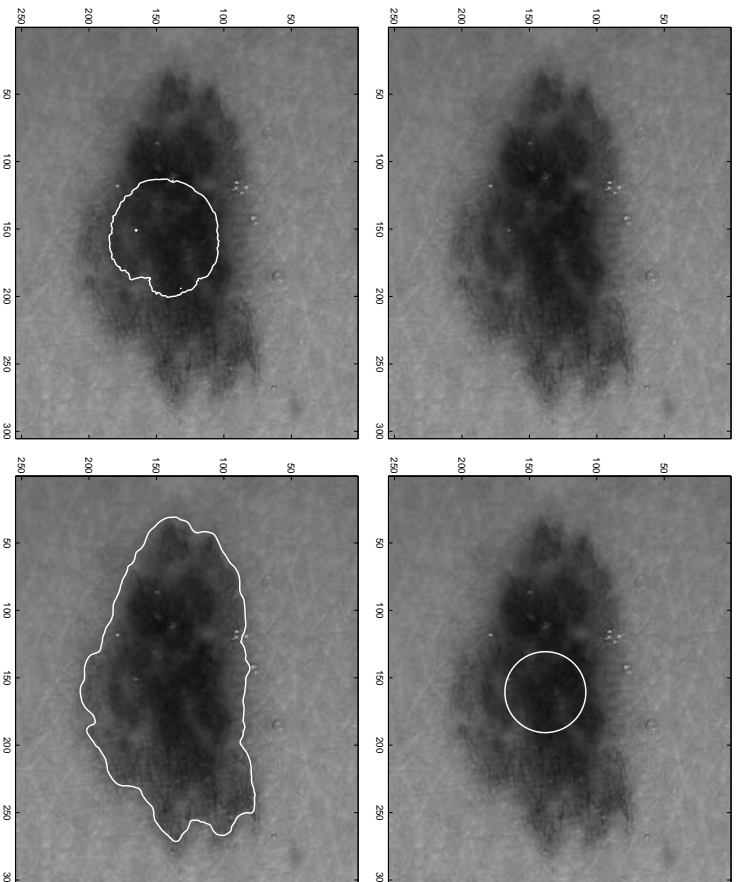


Figura 3.17: Segmentazione di un melanoma da un'immagine fotografica.

come funzione iniziale la funzione distanza (3.40) (in alto a destra). Inoltre il parametro  $\mu$  non deve essere nè troppo grande, essendo i contorni dell'area frastagliati, nè troppo piccolo viste le dimensioni di tale area ( $\mu = 0.1$ ). Per evitare il riconoscimento di altri minimi (picchi nell'evoluzione della funzione level set), si applica oltre all'Eq. (3.33) l'equazione di reinizializzazione (3.42). La soluzione finale è visualizzata in Fig. 3.17 in basso a destra: l'algoritmo riesce a determinare nel miglior modo possibile i contorni della struttura pur essendo questi sfumati e non definiti.

### 3.2.7 Significato del parametro $\mu$

Negli esperimenti numerici eseguiti, abbiamo posto, per semplicità,  $\Delta x = \Delta y = 1$  e  $\Delta t = 0.1$ . L'unico parametro risulta essere  $\mu$ : tale parametro è un fattore di scala. Se si vogliono riconoscere oggetti di piccole dimensioni,  $\mu$  deve essere piccolo, altrimenti può assumere valori più elevati. Inoltre è un parametro di regolarizzazione: evita che la curva si polverizzi (frantumi) durante il moto. Ovviamente, senza un parametro di regolarizzazione, predominerebbe la segmentazione, nella quale, ad ogni pixel, è associata una curva.

È possibile trovare una relazione che lega il valore di tale parametro al raggio di curvatura e quindi alla curvatura stessa. In Fig. 3.18 sono rappresentati 12 cerchi, di diverso raggio: 0.5 pixels per il primo e 6 pixels per l'ultimo (tra un cerchio e il successivo il raggio viene incrementato di 0.5 pixels). A seconda del

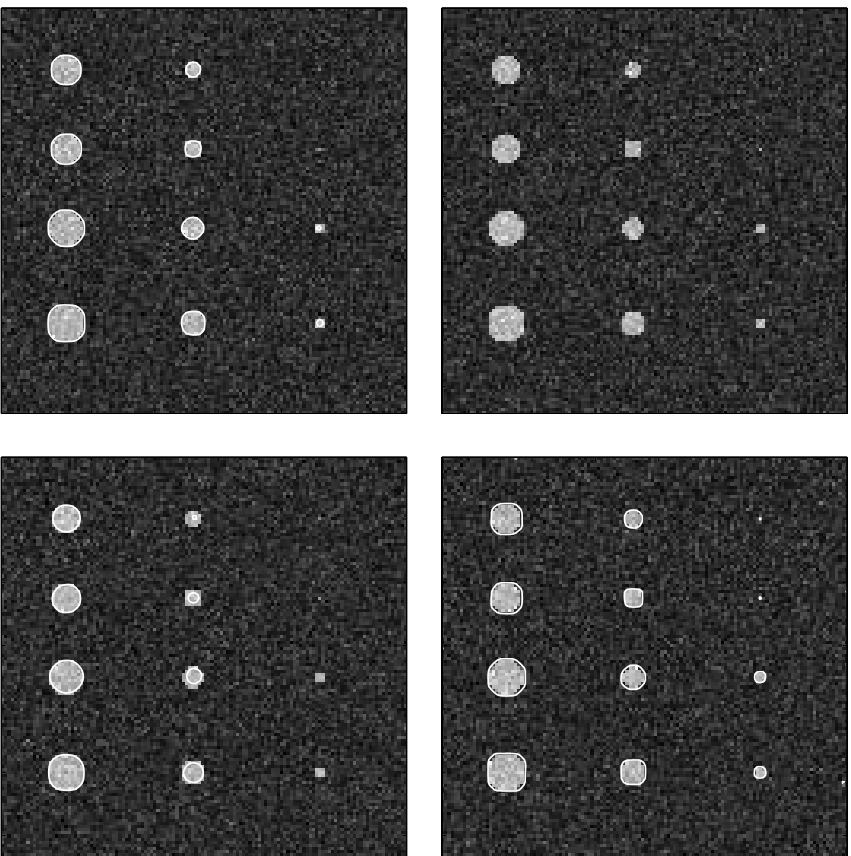


Figura 3.18: Risultato della segmentazione di un'immagine costituita da 12 cerchi con diversi valori del parametro  $\mu$ .

valore di  $\mu$  il risultato della segmentazione è differente: se  $\mu = 0.001$  vengono segmentati correttamente anche i cerchi di minore dimensione (Fig.3.18 in alto a destra), se  $\mu = 0.1$  la segmentazione avviene correttamente solo per i cerchi con raggio maggiore di 2 pixels (Fig.3.18 in basso a sinistra), ed infine, se  $\mu = 2$  solo per i cerchi con raggio maggiore di 4 pixels (Fig.3.18 in basso a destra). Applicando quindi il modello per diversi valori di  $\mu$ , è possibile ricavare una relazione tra il parametro e il raggio di curvatura. Tale relazione è monotonamente crescente e facilmente rappresentabile in scala logaritmica. In Fig.3.19, a partire dai dati ottenuti dalle simulazioni (gli asterischi), si rappresenta la curva che ne esegue il miglior fitting. Attraverso questa curva, a partire dal raggio di curvatura delle parti che si vogliono segmentare, si ricava il valore del parametro più opportuno. Chiaramente tale curva si può estendere anche a valori minori del parametro che permettono quindi la segmentazione di strutture il cui raggio è inferiore a 0.5 pixels. La visualizzazione ad occhio nudo del risultato della segmentazione (linea bianca) di cerchi di così piccole dimensioni, risulta però impossibile; per tale ragione nell'effettuare il test, siamo partiti da un cerchio

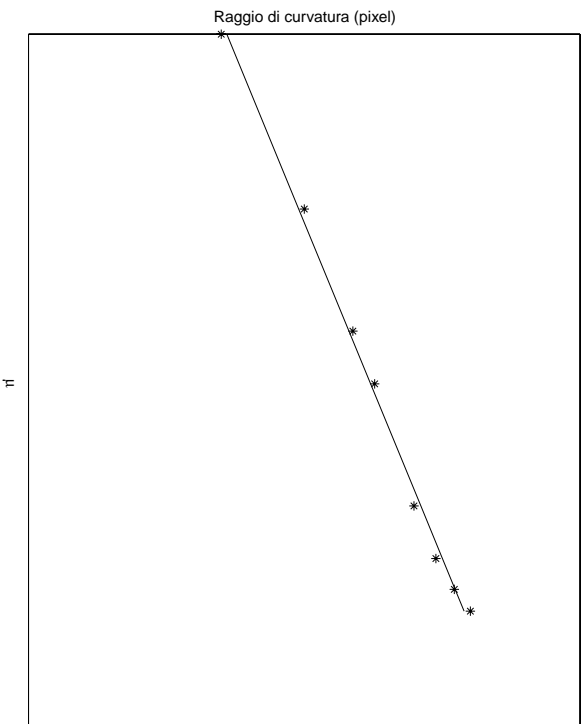


Figura 3.19: Relazione tra il parametro  $\mu$  e il raggio di curvatura in scala logaritmica.

con raggio di curvatura pari a 0.5 pixels.

Riassumendo quindi, è possibile, in base al grado di risoluzione che si desidera ottenere, scegliere, per ogni immagine, il valore di  $\mu$  più opportuno. Tale valore deve essere comunque compatibile con la condizione di stabilità della soluzione. Per chiarire e visualizzare l'effetto di tale parametro nella segmentazione di una struttura di interesse, presentiamo un'immagine medica (Fig.3.20). In questo caso l'obiettivo è la segmentazione delle arterie cerebrali. Minore è il valore del parametro, maggiore è il grado di risoluzione che si ottiene dalla applicazione del modello: è possibile cioè segmentare anche i vasi di piccolissime dimensioni. Al contrario, aumentando il valore del parametro, si ha il riconoscimento solamente dei vasi maggiori. In figura è rappresentato il risultato della segmentazione ottenuto applicando il modello con tre valori del parametro, rispettivamente:  $\mu = 0.01$  (Fig.3.20 in alto a destra),  $\mu = 0.001$  (Fig.3.20 in basso a sinistra), e  $\mu = 0.0001$  (Fig.3.20 in basso a destra). In questo caso chiaramente, per il gran numero di parti da segmentare e per le loro dimensioni, il valore che è necessario attribuire al parametro per avere la segmentazione di tutti i vasi, deve essere molto piccolo.

Codice Matlab dei metodi di segmentazione di immagini, in particolare del metodo Chan-Vese e del metodo probabilistico.

```
% Metodi di segmentazione di immagini
close all, clear all

I00=caricamentoimmagine;
global raggio noiser noiseg mu dt;
global dtcv curv pesoint Mr Np Ng pd;
```



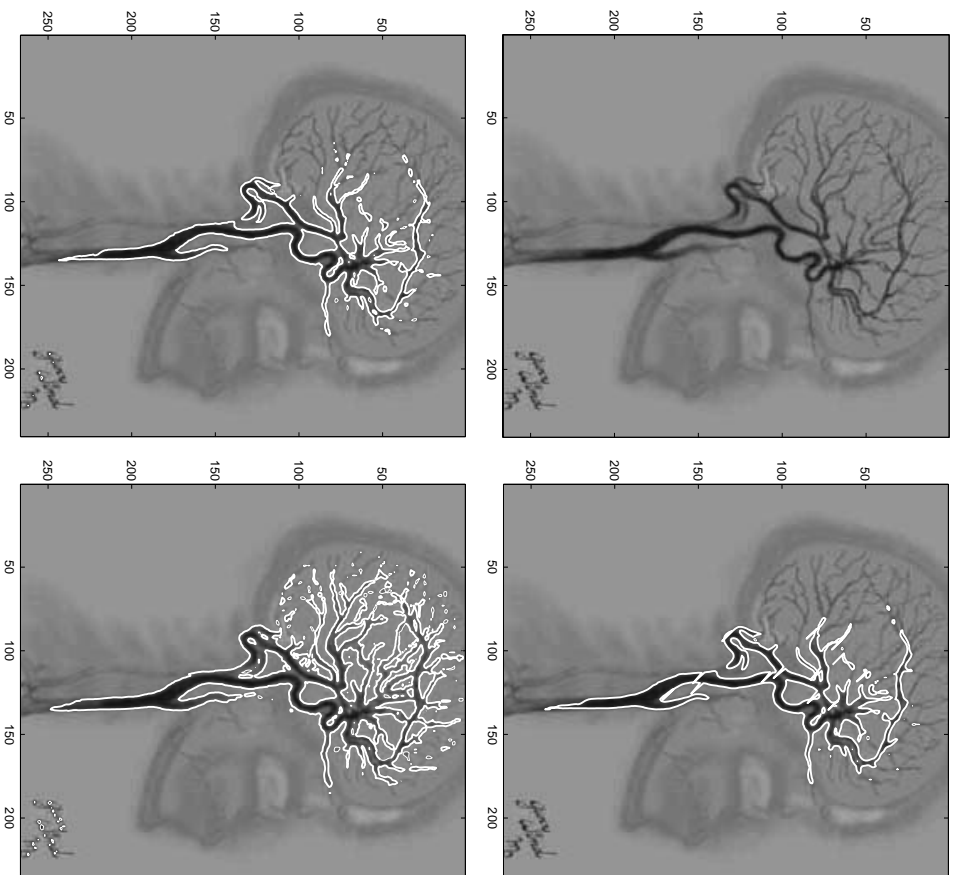


Figura 3.20: Segmentazione delle arterie cerebrali per diversi valori del parametro  $\mu$ , rispettivamente:  $\mu = 0.01$  nell'immagine in alto a destra,  $\mu = 0.001$  nell'immagine in basso a sinistra, e  $\mu = 0.0001$  nell'immagine in basso a destra.

```

[rx ny]=size(I00);
I00=double(I00)/255.;
% Rumore di Rayleigh
Nr=raylrnd(0.1,nx,ny);
% Rumore di Poisson
Np=poissrnd(0.5,nx,ny);
% Rumore di Gauss
Ng=normrnd(1,1,nx,ny);
global I0 im;
I0=I00;
% visualizzazione dell'oggetto
figure(1)
colormap(gray);
image(I0.*70,'Erasemode','None');
axis image; drawnow; hold on;
global I;
% superficiali di riferimento
if im==4 | im==10
    [X Y]=meshgrid(0:ny-1,0:nx-1);
    I=ones(nx,ny).*10000000;
    for x=1:50:450,
        for y=1:50:450,
            Itemp=sqrt((X-x).(X-x)+(Y-y).(Y-y));
            I=(I>=Itemp).*Itemp+(I<Itemp).*I;
        end
    end
end
I=I-raggio;
contour(I,[0,0],'r');
else
    [X Y]=meshgrid(0:ny-1,0:nx-1);
    [x,y]=ginput;
    I=sqrt((X-x(1)).*(X-x(1))+(Y-y(1)).*(Y-y(1)));
    I=I-raggio;
    contour(I,[0,0],'r');
end

mo=input('?Choose the model ','s');
global Iter;
Iter=3000;
% Metodo Chan-Vese
if mo=='cv'
    metodochanvese;
else
    % Metodo probabilistico
    elseif mo=='pr'
        metodoprobabilistico;
    else
        fprintf('ERROR');
        break
end

```

```

end

% Il seguente file esegue il caricamento di un'immagine
% scelta tra le 20 immagini considerate
function f=caricamentoimmagine

global im;
i=input('Choose the image ','s');
im=str2num(i);
global raggio noisec noiser noiseq mu dt;

global dtcv dtr curv pesoint M nx ny pd;
% Oggetto a forma di 8
if im==1
    [X0 Y0]=meshgrid(1:128,1:128);
    R1=sqrt((X0-64).^2 + (Y0-36).^2);
    R2=sqrt((X0-64).^2 + (Y0-90).^2);
    f=150.*(((R1<30)+(R2<30))>0);
    raggio=20;
    % Metodo Chan-Vese
    dtcv=1;
    curv=0.1; pesoint=50;
    % Metodo probabilistico
    mu=0.01;
    dt=5; dtr=0;
    % rumore di Rayleigh
    noiser=1.5;
    % rumore di Poisson
    noiseq=0.2; M=1;
    % rumore di Gauss
    noisec=0.1;

% Ecocardiografia camere atriali1
elseif im==2
    f=imread('normalheart4c.tif','tif');
    [nx ny]=size(f);
    f=f([nx-200:2:nx-50],[150:2:ny-150]+30);
    noiser=0;
    raggio=20;
    mu=2;
    dt=0.1;
    pd='ra';

% Volumetrics camera ventricolare
elseif im==3
    f=imread('laxis.tif','tif');
    [nx ny nz]=size(f);
    f=f(:,:,1);
    f=double(f([1:4:nx],[1:4:ny]));
    f=(f<=0).*25+(f>0).*f;

```

```
noiser=0;
raggio=20;
mu=2;
dt=0.1;
pd='ra';

% Ecografia fetale
elseif im==4
    f=imread('54.0.tif','tiff');
    [nx ny nz]=size(f);
    f=f(:,:,1);
    noiser=0;
    raggio=20;
    mu=0.1;
    dt=1;
    pd='ra';

% Ecocardiografia camere cardiache
elseif im==5
    f=imread('apicalecho.jpg','jpg');
    [nx ny nz]=size(f);
    f=f(:,:,1);
    f=double(f([1:2:nx],[1:2:ny]));
    noiser=0;
    raggio=20;
    mu=1;
    dt=0.4;
    pd='ra';

% Angiografia
elseif im==6
    f=imread('angio.tif','tiff');
    [nx ny]=size(f);
    f=double(f([85:1:nx-20],[20:1:ny-10]));
    f=(f<=0).*125+(f>0).*f;
    noiser=0;
    curv=0.02; pesoint=10;
    raggio=50;
    mu=0.0002;
    dt=500; dtr=0;
    M=1;
    pd='po';

% MRI carotidi
elseif im==7
    f=imread('neck1.tif','tiff');
    [nx ny]=size(f);
    noiser=0;
    raggio=40;
    mu=0.001;
```

```
dt=50; dtr=0;
M=1;
pd='po';

% Angiografia cervello
elseif im==8
    f=imread('angiocerv1.tif','tif');
    [nx ny nz]=size(f);
    f=f(:,:,1);
    noise=0;
    raggio=20;
    mu=0.0001;
    dt=10; dtr=0;
    M=1;
    pd='po';

% Angiografia aorta
elseif im==9
    f=imread('aorta.jpg','jpg');
    [nx ny nz]=size(f);
    f=f(:,:,1);
    noise=0;
    raggio=50;
    mu=0.001;
    dt=50; dtr=0;
    M=1;
    pd='po';

% MRI cervello
elseif im==10
    f=imread('mri.tif','tif');
    [nx ny]=size(f);
    f=double(f([1:2:nx],[1:2:ny]));
    noise=0;
    raggio=20;
    mu=0.01;
    dt=1; dtr=0;
    M=(f/255.>0);
    pd='po';

% MRI meningioma
elseif im==11
    f=imread('meningiomamri.jpg','jpg');
    [nx ny nz]=size(f);
    f=f(:,:,1);
    noise=0;
    raggio=30;
    mu=0.1;
    dt=5; dtr=0.1;
    M=((double(f)/255.)>0.4);
```

```
pd='po';

% MRI cervello
elseif im==12
    f=imread('normalbrain.tif','tif');
    [nx ny]=size(f);
    noisep=0;
    raggio=20;
    mu=0.1;
    dt=1; dtr=0.1;
    M=((double(f)/255.)>0.2);
    pd='po';

% MRI carcinoma broncogenico
elseif im==13
    f=imread('metastaticcarcinomabroncogenico.tif','tif');
    [nx ny]=size(f);
    noisep=0;
    raggio=20;
    mu=0.2;
    dt=1; dtr=0.3;
    M=((double(f)/255.)>0.4);
    pd='po';

% MRI tumore al fegato
elseif im==14
    f=imread('tumore.tif','tif');
    [nx ny]=size(f);
    noisep=0;
    raggio=20;
    mu=0.001;
    dt=10; dtr=0;
    M=1;
    pd='po';

% CT
elseif im==15
    f=imread('heartCTImage.256.256.tif','tif');
    [nx ny]=size(f);
    noisep=0;
    raggio=20;
    mu=0.02;
    dt=5; dtr=0.2;
    M=((double(f)/255.)>0.4);
    pd='po';

% CT addominale
elseif im==16
    f=imread('stomacoct.jpg','jpg');
    [nx ny nz]=size(f);
```

```

f=f(:,:,1);
noisep=0;
raggio=30;
mu=0.05;
dt=10; dtr=0.1;
M=((double(f)/255.)>0.4);
pd='po';

% Mammografia tumore
elseif im==17
    f=imread('mammogr.tif','tif');
    [nx ny]=size(f);
    noisep=0;
    raggio=20;
    mu=0.01;
    dt=0.5; dtr=0.1;
    M=1;
    pd='po';

% Mammografia carcinoma
elseif im==18
    f=imread('carcinoma1.tif','tif');
    [nx ny]=size(f);
    noisep=0;
    raggio=20;
    mu=0.05;
    dt=5; dtr=0.1;
    M=1;
    pd='po';

% Mammografia
elseif im==19
    f=imread('mammografia.jpg','jpg');
    [nx ny nz]=size(f);
    f=f(:,:,1);
    noisep=0;
    raggio=20;
    mu=0.05;
    dt=10; dtr=0.1;
    M=((double(f)/255.)>0.3);
    pd='po';

% Neo
elseif im==20
    f=imread('neo.tif','tif');
    [nx ny]=size(f);
    f=f([50:2:nx-20],[100:2:ny-60]);
    noiseg=0;
    raggio=30;
    mu=0.1;

```

```

dt=1;
pd='ga';

else
    fprintf('ERROR');
    break
end

% Funzione Matlab
% Metodo Chan-Vese
function f=metodochanvесе
global I I0 Iiter dtcv curv pesoint;

for i=1:Iiter,
    i
    He=(I>=0);
    Hi=(I<0);
    I0i=I0.*Hi;
    I0e=I0.*He;
    c1=sum(sum(I0i))./(sum(sum(Hi)));
    c2=sum(sum(I0e))./(sum(sum(He)));
    I=I+dtcv.*(curv.*KG(I)-pesoint.*(-((I0-c1).^2)+((I0-c2).^2)));
    P1(i)=sum(sum((I0i-c1).^2));
    P2(i)=sum(sum((I0e-c2).^2));
    if (mod(i,10)==1)
        figure(2)
        colormap(gray);
        image(I0.*70,'EraseMode','None');
        axis image; drawnow;
        hold on;
        contour(I,[0,0],'r');
        drawnow;
        if (((I<0)-Hi)<=0.0001)
            contour(I,[0,0],'r');
            break;
        end
    end
end

% Funzione Matlab
% Metodo probabilistico
function f=metodoprobabilistico
global I I0 Iiter dt mu dtr nx ny;
global noiser noisep noiseg Nr Np Ng pd im;

if im==1
    pd=input('Choose the pixel distribution ','s');
end

% Distribuzione di probabilità di Rayleigh

```



```

if pd=='ra'
    I0=I0+noiser*Nr;
    for i=1:Iter,
        i
        He=(I>=0);
        Hi=(I<0);
        I0i=I0.*Hi;
        I0e=I0.*He;
        ni=sum(sum(Hi));
        ne=sum(sum(He));
        P1(i)=-ni.*log((sum(sum(I0i.^2))./ni));
        P2(i)=-ne.*log((sum(sum(I0e.^2))./ne));
        P(i)=P1(i)+P2(i);
        if(i>1)
            Pp=P(i-1);
        else
            Pp=100;
        end
        I=I+dt.*(mu.*Kg(I)+(log((1./ni)).*sum(sum(I0i.^2)))
            -log((1./ne).*sum(sum(I0e.^2)))+(ni.*(I0.^2)
            -sum(sum(I0i.^2))./(sum(sum(I0i.^2)))-(ne.*(I0.^2)
            -sum(sum(I0e.^2))./(sum(sum(I0e.^2))));
        vi(i)=i;
        figure(3);
        plot(vi,P1,'k-.',vi,P2,'k-.',vi,P1+P2,'k. '); drawnow;
        if (mod(i,10)==1)
            figure(2);
            colormap(gray);
            image(I0.*70,'EraseMode','None');
            axis image; drawnow; hold on;
            contour(I,[0,0],'r'); hold off;
            drawnow;
            if ((P(i)-Pp)<=0.000001)
                colormap(gray);
                image(I0.*70,'EraseMode','None');
                axis image; drawnow;
                hold on;
                contour(I,[0,0],'r');
                drawnow;
                break;
            end
        end
    end
end
% Distribuzione di probabilità di Poisson
elseif pd=='po'
    I0=I0+noiser*Np;
    global M;
    for i=1:Iter,
        i
        He=(I>=0).*M;

```

```

Hi=(I<0).*M;
I0i=I0.*Hi;
I0e=I0.*He;
ni=sum(sum(Hi));
ne=sum(sum(He));
P1(i)=(sum(sum(I0i)).*log((sum(sum(I0i)))./ni));
P2(i)=(sum(sum(I0e)).*log((sum(sum(I0e)))./ne));
P(i)=(P1(i)+P2(i));
if(i>1)
    Pp=P(i-1);
else
    Pp=100;
end
I=I+dt.*(mu.*KG(I)-(I0.*log((1./ni).*sum(sum(I0i)))
-I0.*log((1./ne).*sum(sum(I0e))))+(ni.*(I0)
-sum(sum(I0i)))./ni-(ne.*(I0)-sum(sum(I0e)))./ne));
I=I+dt.*rein(I);
vi(i)=i;
if(mod(i,10)==1)
    figure(2);
    colormap(gray);
    image(I0.*70,'EraseMode','None');
    axis image; drawnow; hold on;
    contour(I,[0,0],'r');
    hold off;
    drawnow;
    if(abs(P(i)-Pp)<=0.000001)
        colormap(gray);
        image(I0.*70,'EraseMode','None');
        axis image; drawnow;
        hold on;
        contour(I,[0,0],'r');
        drawnow;
        break;
    end
end
end
end
% Distribuzione di probabilità di Gauss
elseif pd=='ga'
    I0=I0+noiseg*Ng;
    for i=1:Iter,
        i
        He=(I>=0);
        Hi=(I<0);
        I0i=I0.*Hi;
        I0e=I0.*He;
        ni=sum(sum(Hi));
        ne=sum(sum(He));
        P1(i)=-ni.*log((ni.*(sum(sum(I0i.^2)))
        -(sum(sum(I0i))).^2))./ni.^2);

```

```

P2(i)=-ne.*log(ne.*(sum(sum(I0e.^2)))
      -((sum(sum(I0e)).^2))./ne.^2);
P(i)=abs(P1(i)+P2(i));
if(i>1)
    Pp=P(i-1);
else
    Pp=100;
end
I=I+dt.*(mu.*KG(I)+(log(ni.*(sum(sum(I0i.^2)))
      -(sum(sum(I0i)).^2)/(ni.^2))+((I0.^2).*(ni.^2)
      -2.*I0.*ni.*(sum(sum(I0i))+2.*(sum(sum(I0i))).^2)
      -ni.*(sum(sum(I0i).^2)))/(ni.*(sum(sum(I0i).^2)))
      -((sum(sum(I0i)).^2)-(log(ne.*(sum(sum(I0e.^2)))
      -(sum(sum(I0e)).^2)/(ne.^2))+((I0.^2).*(ne.^2)
      -2.*I0.*ne.*(sum(sum(I0e))+2.*(sum(sum(I0e)).^2)
      -ne.*(sum(sum(I0e).^2)))/(ne.*(sum(sum(I0e).^2)))
      -((sum(sum(I0e)).^2)))));
vi(i)=i;
if (mod(i,10)==1)
    figure(2);
    colormap(gray);
    image(I0.*70,'EraseMode','None');
    axis image; drawnow; hold on;
    contour(I,[0,0],'r');
    drawnow;
    if ((P(i)-Pp)<=0.000001)
        colormap(gray);
        image(I0.*70,'EraseMode','None');
        axis image; drawnow;
        hold on;
        contour(I,[0,0],'r');
        drawnow;
        break;
    end
end
end
else
    fprintf('ERROR');
    break
end
end

```