



Università degli Studi di Padova

Facoltà di Ingegneria

Corso di Laurea Magistrale in Bioingegneria

Tesi di laurea

**Formulazione di un metodo a insieme di livelli multipli
con singola funzione implicita per la segmentazione di
immagini**

Relatore: Dott. Enrico Grisan

Correlatore: Dott. Elisa Veronese

Laureando: Andrea Campion

17 Aprile 2012

Sommario

In questa tesi viene proposta una variante della formulazione level set per la segmentazione automatica di immagini. Verranno illustrati lo stato dell'arte e la nuova metodologia proposta e infine verranno presentati i risultati sperimentali ottenuti applicando l'algoritmo sia ad immagini simulate che ad immagini reali, in particolare a immagini ottenute da risonanza magnetica. Verranno riportati ed analizzati i risultati qualitativi e quantitativi.

Indice

Sommario

1	Segmentazione di immagini	1
1.1	Riconoscimento dei bordi	1
1.1.1	Collegamento dei contorni interrotti	1
1.2	Segmentazione dell'immagine	2
1.2.1	Sogliatura	2
1.2.2	Metodi basati sulle regioni	3
1.3	Metodi basati su PDE	3
2	Level Set	5
2.1	Metodo Chan-Vese	6
2.2	Metodo Multiphase Level Set	11
3	Multi Level Set	17
3.1	Trattazione teorica	17
3.2	Aspetti implementativi	21
4	Risultati	27
4.1	Analisi qualitativa	27
4.2	Analisi quantitativa	38

4.3 Considerazioni	39
Bibliografia	40
Elenco delle figure	43

Capitolo 1

Segmentazione di immagini

1.1 Riconoscimento dei bordi

Al fine di riconoscere i bordi presenti in un'immagine si possono utilizzare operatori locali basati sulla stima del gradiente di luminosità. Come punti di massima variazione di intensità si possono considerare il minimo o il massimo della derivata prima o lo zero della derivata seconda. In uno spazio bidimensionale si possono quindi usare:

- Gradiente: $G(x, y) = \nabla f(x, y) = (f'_x, f'_y)$
- Laplaciano: $L(x, y) = \nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = f''_{xx} + f''_{yy}$

definendo un'opportuna formulazione discreta della derivata dell'immagine. Vanno poi costruiti degli appositi filtri come i filtri di Roberts, Prewitt, Sobel e il filtro LoG.

1.1.1 Collegamento dei contorni interrotti

Dopo l'iniziale estrazione del contorno accade spesso che siano presenti delle interruzioni dovute alla cattiva identificazione dei bordi. Bisogna quindi

eeguire una procedura per ricostruire il contorno esatto o la versione che meglio lo approssima.

Si considera un punto (x,y) che appartiene al contorno e un punto $(x_0, y_0) \in Im(x, y)$. Se (x_0, y_0) soddisfa le condizioni:

$$\begin{cases} |\Delta f(x, y) - \Delta f(x_0, y_0)| \leq R, & E \text{ soglia non negativa} \\ |\alpha(x, y) - \alpha(x_0, y_0)| \leq A, & A \text{ soglia angolare non negativa} \end{cases}$$

allora il pixel analizzato verrà unito al bordo e assumerà un dato valore di grigio. Questo processo viene ripetuto per tutti i pixel dell'intorno di ogni pixel del bordo.

1.2 Segmentazione dell'immagine

1.2.1 Sogliatura

La sogliatura è un metodo basato sull'istogramma dell'immagine. Ogni pixel di un'immagine a toni di grigio è di solito caratterizzato da un valore di luminanza e algoritmo fissa una soglia di luminanza, allo scopo di distinguere tra due regioni, ad esempio lo sfondo e gli oggetti dell'immagine. La soglia può essere fissata arbitrariamente, o in maniera automatica in base ad un certo criterio statistico. Una volta fissata la soglia il metodo produce un'immagine binaria composta dai pixel sotto la soglia (sfondo) e i pixel sopra la soglia (oggetto) che permette di evidenziare la forma degli oggetti dell'immagine e la loro posizione. E' inoltre possibile estendere il metodo a più livelli scegliendo soglie differenti in modo da segmentare l'immagine in un numero differente di regioni.

1.2.2 Metodi basati sulle regioni

Sono tecniche basate non più sulla ricerca delle discontinuità dei livelli di grigio, ma sulla ricerca delle regioni omogenee all'interno dell'immagine secondo criteri di omogeneità fra pixel.

Region growing

Si basa sull'adiacenza tra pixel. Si inizia prendendo un insieme di pixel detti semi ed ogni regione inizialmente è costituita dal solo insieme di tali semi. In seguito l'algoritmo analizza i pixel adiacenti ai pixel iniziali e secondo criteri di somiglianza cerca di ingrandire ognuna di queste regioni, lasciandole libere di aggregarsi, dividersi da esse oppure di diventare esse stesse un nuovo seme. Il processo è iterato finché l'intera immagine non viene segmentata.

Split and Merge

Si basa sul criterio di omogeneità. L'immagine inizialmente analizzata e se si riscontrano delle regioni non omogenee al suo interno essa viene divisa in un numero arbitrario di regioni disgiunte dette figlie che a loro volta vengono nuovamente divise se il criterio di omogeneità non è soddisfatto. Se qualcuna di queste regioni adiacenti presenta caratteristiche simili esse vengono unite, altrimenti il processo continua fino a che non siano più possibili divisioni o unioni.

1.3 Metodi basati su PDE

Sono metodi che si basano sulla soluzione di un'equazione differenziale parziale opportunamente definita attraverso schemi numerici di integrazione.

La propagazione di curve è una tecnica comune di questa categoria, con numerose applicazioni nell'estrazione di oggetti. L'idea base è quella di evolvere una curva fino al raggiungimento del minimo di un funzionale di costo definito in base al compito che deve essere svolto. Come tutti i problemi inversi la minimizzazione di un funzionale di costo è un problema non banale che richiede l'imposizione di alcuni vincoli di regolarità sulla soluzione che possono essere visti come vincoli geometrici nell'evoluzione della curva. Esempi di metodi basati su PDE sono i contorni attivi e i level set.

Capitolo 2

Level Set

Nei problemi di segmentazione di immagini è molto diffuso l'impiego di metodi basati sui level set, che si basano sull'evoluzione di curve nel dominio dell'immagine. In particolare il metodo basato sull'evoluzione guidata dalla curvatura media presentato da Osher e Sethian [1] è stato impiegato in modo estensivo poiché permette il cambiamento automatico della topologia della curva in presenza di angoli e cuspidi nell'immagine e nel passaggio dall'approccio continuo al discreto il problema della discretizzazione avviene su una griglia rettangolare di dimensioni prefissate. La curva è rappresentata implicitamente mediante una funzione di Lipschitz ϕ da $C = \{(x, y) | \phi(x, y) = 0\}$ e la sua evoluzione è data dall'evoluzione della curva a livello zero al tempo t descritta dalla funzione $\phi(t, x, y)$. Evolvere la curva C in direzione normale con velocità F corrisponde a risolvere l'equazione differenziale

$$\frac{\partial \phi}{\partial t} = |\nabla \phi| F, \phi(0, x, y) = \phi_0(x, y) \quad (2.1)$$

dove $\{(x, y) | \phi_0(x, y) = 0\}$ definisce il contorno iniziale. Nel caso particolare in cui il movimento sia guidato dalla curvatura media, dove $F =$

$\operatorname{div}\left(\frac{\nabla\phi(x,y)}{|\nabla\phi(x,y)|}\right)$ è la curvatura della curva di livello di ϕ passante per (x, y) , l'equazione diventa

$$\begin{cases} \frac{\partial\phi}{\partial t} = |\nabla\phi|\operatorname{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) & t \in (0, \infty), x \in \mathbb{R}^2 \\ \phi(0, x, y) = \phi_0(x, y) & x \in \mathbb{R}^2. \end{cases} \quad (2.2)$$

2.1 Metodo Chan-Vese

Come caso particolare analizziamo il modello di evoluzione proposto da Chan e Vese [2].

Sia Ω un sottoinsieme aperto di \mathbb{R}^2 e sia $\partial\Omega$ il suo contorno. Sia $u_0 : \bar{\Omega} \rightarrow \mathbb{R}$ una data immagine. Definiamo la curva C in Ω come il contorno di un sottoinsieme aperto ω di Ω .

Per semplificare la notazione definiamo come $\operatorname{in}(C)$ la regione ω e $\operatorname{out}(C)$ la regione $\Omega \setminus \bar{\omega}$.

Ipotizziamo che l'immagine u_0 sia composta da due regioni omogenee di intensità costante di valori distinti u_0^i e u_0^o e denotiamo il suo contorno con C_0 . Allora $u_0 \approx u_0^i$ nella regione interna l'oggetto e $u_0 \approx u_0^o$ fuori l'oggetto. Definendo come energia la somma dei termini

$$\begin{aligned} F_1(C) + F_2(C) &= \int_{\operatorname{in}(C)} |u_0(x, y) - c_1|^2 dx dy \\ &+ \int_{\operatorname{out}(C)} |u_0(x, y) - c_2|^2 dx dy \end{aligned} \quad (2.3)$$

dove C è una qualsiasi curva e le costanti c_1 e c_2 dipendenti da C sono i valori medi di u_0 rispettivamente dentro e fuori C . In questo semplice caso è ovvio che C_0 , il contorno dell'oggetto, sia la curva che minimizza

$$\inf_c \{F_1(C) + F_2(C)\} \approx 0 \approx F_1(C_0) + F_2(C_0). \quad (2.4)$$

Questo può essere dimostrato facilmente, infatti se la curva C si trova fuori l'oggetto allora $F_1(C) > 0$ e $F_2(C) \approx 0$. Se la curva è dentro l'oggetto allora $F_1(C) \approx 0$ e $F_2(C) > 0$. Se la curva si trova in parte all'interno e in parte all'esterno dell'oggetto allora $F_1(C) > 0$ e $F_2(C) > 0$. Infine se l'energia è minima, cioè risulta $C = C_0$, la curva si trova sul bordo dell'oggetto.

Nel metodo proposto in [2] vengono aggiunti dei termini di regolarizzazione oltre al termine energetico, quali la lunghezza della curva C e l'area della regione interna a C che impongono dei vincoli di regolarità sull'evoluzione della curva. Il funzionale finale risulta quindi essere

$$\begin{aligned}
 F(c_1, c_2, C) &= \mu \cdot \text{Lunghezza}(C) + \nu \cdot \text{Area}(\text{in}(C)) \\
 &+ \lambda_1 \int_{\text{in}(C)} |u_0(x, y) - c_1|^2 dx dy \\
 &+ \lambda_2 \int_{\text{out}(C)} |u_0(x, y) - c_2|^2 dx dy
 \end{aligned} \tag{2.5}$$

dove $\mu \geq 0$, $\nu \geq 0$, $\lambda_1, \lambda_2 > 0$ sono parametri fissati. Il problema dell'evoluzione della curva conduce alla minimizzazione

$$\inf_{c_1, c_2, C} F(c_1, c_2, C). \tag{2.6}$$

Nella formulazione level set [1], $C \subset \Omega$ è rappresentato dalla curva a livello zero di una funzione di Lipschitz $\phi : \Omega \rightarrow \mathbb{R}$ tale che

$$\begin{cases}
 C = \partial\omega = \{(x, y) \in \Omega : \phi(x, y) = 0\} \\
 \text{in}(C) = \omega = \{(x, y) \in \Omega : \phi(x, y) > 0\} \\
 \text{out}(C) = \Omega \setminus \bar{\omega} = \{(x, y) \in \Omega : \phi(x, y) < 0\}
 \end{cases} \tag{2.7}$$

Per la formulazione del modello viene sostituita la variabile incognita C con la variabile incognita ϕ . Utilizzando la funzione di Heaviside H e la funzione di Dirac monodimensionale δ_0 definite come

$$H(x) = \begin{cases} 1, & \text{se } x \geq 0 \\ 0, & \text{se } x < 0 \end{cases} \quad \delta_0(x) = \frac{d}{dx}H(x) \quad (2.8)$$

i termini che compaiono all'interno dell'energia di F possono essere espressi come

$$\begin{aligned} \text{Lunghezza}\{\phi = 0\} &= \int_{\Omega} |\nabla H(\phi(x, y))| dx dy \\ &= \int_{\Omega} \delta_0(\phi(x, y)) |\nabla \phi(x, y)| dx dy \end{aligned} \quad (2.9)$$

$$\text{Area}\{\phi \geq 0\} = \int_{\Omega} H(\phi(x, y)) dx dy \quad (2.10)$$

$$\int_{\phi > 0} |u_0(x, y) - c_1|^2 dx dy = \int_{\Omega} |u_0(x, y) - c_1|^2 H(\phi(x, y)) dx dy \quad (2.11)$$

$$\int_{\phi < 0} |u_0(x, y) - c_2|^2 dx dy = \int_{\Omega} |u_0(x, y) - c_2|^2 (1 - H(\phi(x, y))) dx dy \quad (2.12)$$

quindi l'energia $F(c_1, c_2, \phi)$ può essere riscritta come

$$\begin{aligned} F(c_1, c_2, C) &= \mu \int_{\Omega} |\nabla H(\phi(x, y))| dx dy \\ &+ \nu \int_{\Omega} H(\phi(x, y)) dx dy \\ &+ \lambda_1 \int_{\Omega} |u_0(x, y) - c_1|^2 H(\phi(x, y)) dx dy \\ &+ \lambda_2 \int_{\Omega} |u_0(x, y) - c_2|^2 (1 - H(\phi(x, y))) dx dy. \end{aligned} \quad (2.13)$$

Mantenendo fisso ϕ e minimizzando l'energia $F(c_1, c_2, \phi)$ rispetto alle costanti c_1 e c_2 queste funzioni costanti di ϕ si possono esprimere come

$$c_1(\phi) = \frac{\int_{\Omega} u_0(x, y) H(\phi(x, y)) dx dy}{\int_{\Omega} H(\phi(x, y)) dx dy} \quad (2.14)$$

sotto la condizione $\int_{\Omega} H(\phi(x, y)) dx dy > 0$ e

$$c_2(\phi) = \frac{\int_{\Omega} u_0(x, y) (1 - H(\phi(x, y))) dx dy}{\int_{\Omega} (1 - H(\phi(x, y))) dx dy} \quad (2.15)$$

se $\int_{\Omega} (1 - H(\phi(x, y))) dx dy > 0$. In tal caso c_1 e c_2 possono essere visti come i livelli medi dell'immagine dentro la curva

$$\begin{cases} c_1(\phi) = \text{media}(u_0) & \text{dove } \{\phi \geq 0\} \\ c_2(\phi) = \text{media}(u_0) & \text{dove } \{\phi \leq 0\} \end{cases} \quad (2.16)$$

e considerando c_1 e c_2 fissati, minimizzando F rispetto a ϕ si deduce l'equazione di Eulero-Lagrange associata per ϕ . Parametrizzando in direzione discendente con un tempo artificiale $t \geq 0$ l'equazione in $\phi(t, x, y)$ risulta

$$\begin{cases} \frac{\partial \phi}{\partial t} = \delta(\phi) \left[\mu \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) - \nu - \lambda_1 (u_0 - c_1)^2 + \lambda_2 (u_0 - c_2)^2 \right] = 0 & \text{in } (0, \infty) \times \Omega \\ \phi(0, x, y) = \phi_0(x, y) & \text{in } \Omega \\ \frac{\delta(\phi)}{|\nabla \phi|} \frac{\partial \phi}{\partial \vec{n}} = 0 & \text{in } \partial \Omega \end{cases} \quad (2.17)$$

dove \vec{n} denota la normale uscente al contorno $\partial\Omega$ e $\frac{\partial\phi}{\partial\vec{n}}$ indica la derivata della normale di ϕ sul bordo. La funzione di Heaviside con la sua derivata 2.8 sono state sostituite dalle versioni regolarizzate 2.1

$$H_\epsilon(x) = \frac{1}{2} \left(1 + \frac{2}{\pi} \arctan \left(\frac{x}{\epsilon} \right) \right)$$

$$\delta_\epsilon(x) = \frac{\epsilon}{\pi} \frac{1}{(\epsilon^2 + x^2)}$$

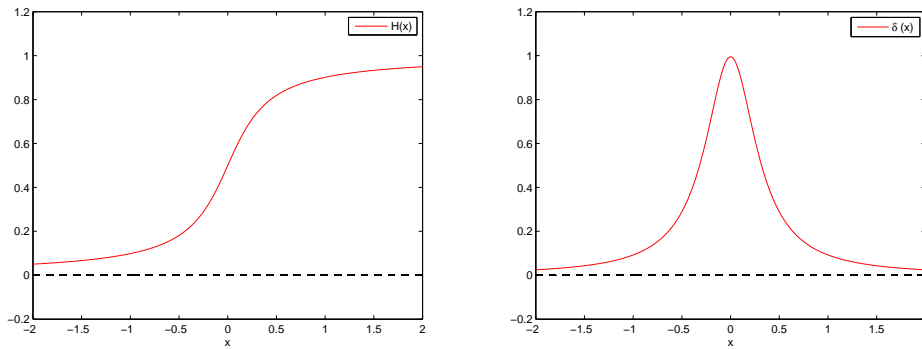


Figura 2.1: $H(x)$ e $\delta(x)$ regolarizzate

Si riporta in figura 2.2 un esempio dell'applicazione del metodo appena illustrato.

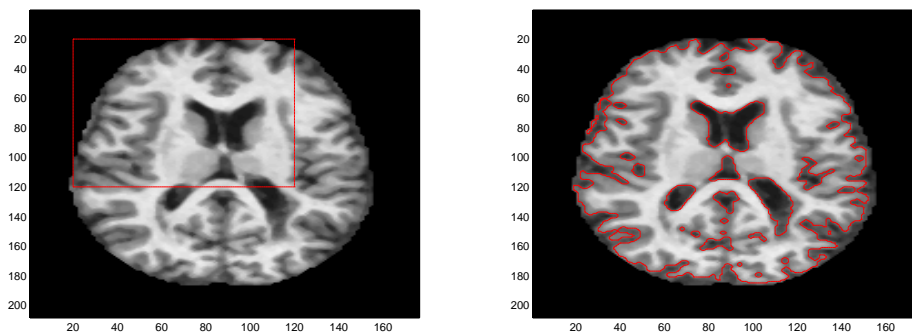


Figura 2.2: Metodo Chan-Vese

Come si può vedere da figura 2.2 il metodo è in grado di partizionare l'immagine in due regioni o fasi, ma da un'analisi qualitativa dell'immagine è evidente che le fasi non sono solamente due. In particolare il metodo riesce a separare l'encefalo dallo sfondo, senza però distinguere i vari tessuti che lo compongono.

2.2 Metodo Multiphase Level Set

Per risolvere i problemi di segmentazione in cui l'immagine presenta diverse regioni omogenee al proprio interno Chan e Vese hanno proposto un metodo [3] basato sul loro algoritmo per la segmentazione multifase. Sotto l'assunzione che le immagini da analizzare siano costanti a tratti vengono utilizzate N funzioni implicite ϕ_i , $i = 1, \dots, N$ sufficienti a produrre 2^N fasi differenti.

Con $N = 2$ ad esempio, i segni delle due funzioni implicite ϕ_1 e ϕ_2 sono utilizzati per determinare fino a quattro differenti fasi.

Consideriamo ora il caso più generico in cui si usino $m = \log_2 n$ funzioni $\phi_i : \Omega \rightarrow \mathbb{R}$, $i = 1, \dots, m$. L'unione dei punti in cui $\phi_i = 0$, $i = 1, \dots, m$ rappresenta i bordi dell'immagine segmentata.

Senza perdita di generalità si possono introdurre la funzione vettoriale $\Phi = (\phi_1, \dots, \phi_m)$ e la funzione di Heaviside vettoriale $H(\Phi) = (H(\phi_1), \dots, H(\phi_m))$ le cui componenti sono solo 0 o 1.

E' quindi possibile definire le fasi nel dominio Ω nel seguente modo: due pixel $(x_1, y_1), (x_2, y_2) \in \Omega$ appartengono allo stessa fase se e solo se $H(\Phi(x_1, y_1)) = H(\Phi(x_2, y_2))$. Ci sono fino a $n = 2^m$ valori possibili nel dominio di $H(\Phi)$.

In questo modo è possibile definire fino a 2^m fasi o classi del dominio di definizione Ω . Le classi così definite formano una decomposizione disgiunta e coprono l'intero dominio Ω e ogni pixel $(x, y) \in \Omega$ apparterrà ad uno solo dei livelli e mai a vuoti o sovrapposizioni tra diverse fasi.

Etichettando le classi come I con $1 \leq I \leq 2^m = n$ vengono introdotti il vettore dei valori medi $c = (c_1, \dots, c_n)$ dove $c_I = \text{media}(u_0)$ nella classe I -esima e la funzione caratteristica χ_I per ogni classe I . Il funzionale da minimizzare che ne risulta è

$$F_n^{MS}(c, \Phi) = \sum_{1 \leq I \leq n=2^m} \int_{\Omega} (u_0(x, y) - c_I)^2 \chi_I dx dy + \nu \frac{1}{2} \sum_{1 \leq I \leq n=2^m} \int_{\Omega} |\nabla \chi_I|. \quad (2.18)$$

Per semplificare il modello il termine associato alla lunghezza del contorno può essere sostituito con $\sum_i \int_{\Omega} |\nabla H(\phi_i)|$ risultando

$$F_n(c, \Phi) = \sum_{1 \leq I \leq n=2^m} \int_{\Omega} (u_0 - c_I)^2 \chi_I dx dy + \nu \frac{1}{2} \sum_{1 \leq I \leq n=2^m} \int_{\Omega} |\nabla H(\phi_i)|. \quad (2.19)$$

Ovviamente quando $n = 2$ si ricade nel metodo proposto nel capitolo precedente. Riportiamo ora il caso di $n = 4$ fasi che può essere risolto con $m = 2$ funzioni implicite.

Il funzionale da minimizzare è

$$\begin{aligned}
F_4(c, \Phi) &= \int_{\Omega} (u_0 - c_{11})^2 H(\phi_1) H(\phi_2) dx dy \\
&+ \int_{\Omega} (u_0 - c_{10})^2 H(\phi_1) (1 - H(\phi_2)) dx dy \\
&+ \int_{\Omega} (u_0 - c_{01})^2 (1 - H(\phi_1)) H(\phi_2) dx dy \\
&+ \int_{\Omega} (u_0 - c_{00})^2 (1 - H(\phi_1)) (1 - H(\phi_2)) dx dy \\
&+ \nu \int_{\Omega} |\nabla H(\phi_1)| + \nu \int_{\Omega} |\nabla H(\phi_2)|
\end{aligned} \tag{2.20}$$

dove $c = (c_{11}, c_{10}, c_{01}, c_{00})$ è un vettore costante e $\Phi = (\phi_1, \phi_2)$. Con tali notazioni possiamo descrivere l'immagine u come

$$\begin{aligned}
u &= c_{11} H(\phi_1) H(\phi_2) \\
&+ c_{10} H(\phi_1) (1 - H(\phi_2)) \\
&+ c_{01} (1 - H(\phi_1)) H(\phi_2) \\
&+ c_{00} (1 - H(\phi_1)) (1 - H(\phi_2)).
\end{aligned}$$

e le equazioni di Eulero-Larange che si ottengono minimizzando 2.20 rispetto a c e Φ con condizioni iniziali date $\phi_1(0, x, y) = \phi_{1,0}(x, y)$, $\phi_2(0, x, y) = \phi_{2,0}(x, y)$,

$$c_{11}(\Phi) = \text{media}(u_0) \text{ in } \{(x, y) : \phi_1(t, x, y) > 0, \phi_2(t, x, y) > 0\}$$

$$c_{10}(\Phi) = \text{media}(u_0) \text{ in } \{(x, y) : \phi_1(t, x, y) > 0, \phi_2(t, x, y) < 0\}$$

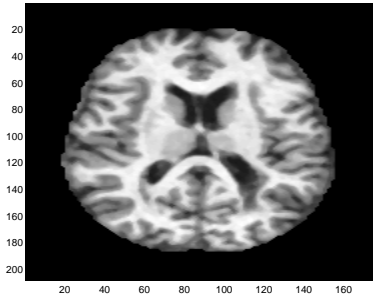
$$c_{01}(\Phi) = \text{media}(u_0) \text{ in } \{(x, y) : \phi_1(t, x, y) < 0, \phi_2(t, x, y) > 0\}$$

$$c_{00}(\Phi) = \text{media}(u_0) \text{ in } \{(x, y) : \phi_1(t, x, y) < 0, \phi_2(t, x, y) < 0\}$$

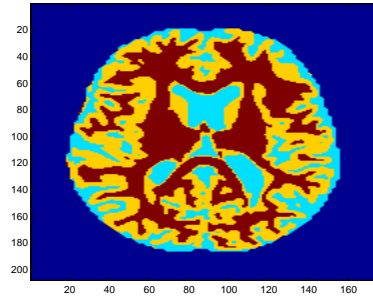
dove si è impiegato il tempo artificiale t risultano essere

$$\begin{aligned} \frac{\partial \phi_1}{\partial t} = & \delta_\epsilon(\phi_1) \left\{ \nu \operatorname{div} \left(\frac{\nabla \phi_1}{|\nabla \phi_1|} \right) \right. \\ & + [((u_0 - c_{11})^2 - (u_0 - c_{01})^2) H(\phi_2) \\ & \left. + ((u_0 - c_{10})^2 - (u_0 - c_{00})^2) (1 - H(\phi_2)) \right\} \end{aligned}$$

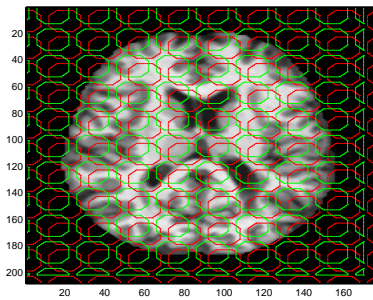
$$\begin{aligned} \frac{\partial \phi_2}{\partial t} = & \delta_\epsilon(\phi_2) \left\{ \nu \operatorname{div} \left(\frac{\nabla \phi_2}{|\nabla \phi_2|} \right) \right. \\ & + [((u_0 - c_{11})^2 - (u_0 - c_{01})^2) H(\phi_1) \\ & \left. + ((u_0 - c_{10})^2 - (u_0 - c_{00})^2) (1 - H(\phi_1)) \right\}. \end{aligned}$$



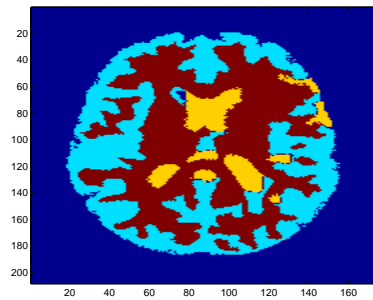
(a) Immagine originale.



(b) Segmentazione vera.



(c) Inizializzazione.



(d) Segmentazione algoritmo.

Figura 2.3: Multiphase

La limitazione maggiore di questo approccio è che può essere applicato solo a immagini con valori vettoriali (RGB o multispettrali) e non a immagini scalari. Inoltre il numero di fasi è sempre una potenza di 2, senza tener conto dell'effettivo numero di fasi presenti nell'immagine. Come osservato da Osher e Fedkiw [4] questo fatto si riflette in maniera negativa sui termini di minimizzazione basati sulla lunghezza dei contorni che sono usati come termini di regolarizzazione. Al posto di utilizzare funzioni di livello multiple Lie et al. [5] hanno proposto un differente approccio che impiega una singola funzione implicita per identificare più di una fase. In particolare un insieme di funzioni base ψ_i sono associate alla funzione implicita ϕ . Ogni funzione base è definita come

$$\psi_i = \frac{1}{\alpha_i} \prod_{j=1, j \neq i}^n (\phi - j) \quad (2.21)$$

con α_i definito come

$$\alpha_i = \prod_{k=1, k \neq i}^n (i - k). \quad (2.22)$$

L'evoluzione di ϕ è vincolata a risultare una funzione costante a tratti, inoltre per assicurare che non ci siano vuoti o sovrapposizioni tra le differenti fasi devono essere introdotti dei vincoli espliciti nella formulazione. Questi termini, assieme alla bontà della scelta delle funzioni base che saranno impiegate, rendono difficile la convergenza del metodo.

Capitolo 3

Multi Level Set

3.1 Trattazione teorica

Contrariamente a quanto proposto in [3], nel nostro approccio vogliamo trovare più di un livello con una singola funzione implicita ϕ . Assumiamo che il numero di fasi in cui vogliamo segmentare l'immagine sia noto e pari a L . Definiamo $L - 1$ soglie costanti $\theta_i, i = 1, \dots, L - 1$ che, senza perdita di generalità, possono essere scelte in \mathbb{N} come $\theta_i \in \{1, 2, \dots, L - 1\}$. Tali soglie corrispondono a diversi livelli che intersecano ϕ , ognuno dei quali descrive un contorno $C_i, i = 1, \dots, L - 1$ dove $C_i = \{(x, y) | \phi(x, y) = \theta_i\}$. In questo modo l'immagine u_0 può essere divisa in L sottoinsiemi Ω_j risultando $\bigcup_{j=1}^L \Omega_j = u_0$ e $\Omega_j \cap \Omega_k = \emptyset \forall j \neq k$. Utilizzando questa formulazione possiamo calcolare la lunghezza di $C_i = \partial\Omega_i$ e l'area dentro Ω_j . In particolare la lunghezza totale e l'area totale risultano essere rispettivamente

$$\mathcal{L}(\phi) = \frac{1}{2} \sum_{i=1}^{L-1} \int_{\Omega} \delta_i(\phi(\Omega) - \theta_i) |\nabla(\phi(\Omega) - \theta_i)| d\Omega \quad (3.1)$$

$$\mathcal{A}(\phi) = \sum_{j=1}^{L-1} \int_{\Omega} H_j(\phi(\Omega)) d\Omega \quad (3.2)$$

dove $H_j(x)$ è la funzione caratteristica di Ω_j calcolata come la combinazione di funzioni di Heaviside multiple e $\delta_i(x)$ è la sua derivata. Sono state definite H_j differenti in corrispondenza dei vari livelli come

$$\begin{cases} H_1(x, y) = 1, & \phi < \theta_1 \\ H_j(x, y) = 1, & \theta_{j-1} < \phi < \theta_j, \forall j \neq 1, L \\ H_L(x, y) = 1, & \phi > \theta_{L-1} \end{cases} \quad (3.3)$$

Nell'implementazione dell'algoritmo sono state impiegate delle versioni regolarizzate della funzione $H(x, y)$ e di $\delta(x, y)$. In particolare abbiamo introdotto una regolarizzazione $C^\infty(\bar{\Omega})$ di H modellata come la funzione errore di Gauss

$$\text{erf}(\phi) = \frac{2}{\pi} \int_0^\phi \exp(-t^2) dt. \quad (3.4)$$

Ridefinendo ogni funzione caratteristica H_j come

$$\begin{cases} H_1(x, y) = 1 - \text{erf}\left(\frac{\phi - \theta_1}{\sigma}\right), & \phi < \theta_1 \\ H_j(x, y) = \text{erf}\left(\frac{\phi - \theta_j}{\sigma}\right) - \text{erf}\left(\frac{\phi - \theta_{j-1}}{\sigma}\right), & \theta_{j-1} < \phi < \theta_j, \forall j \neq 1, L \\ H_L(x, y) = 1 + \text{erf}\left(\frac{\phi - \theta_{L-1}}{\sigma}\right), & \phi > \theta_{L-1} \end{cases} \quad (3.5)$$

con $0 < \sigma < 1$ costante, si ottiene per derivazione

$$\begin{cases} \delta_1(x, y) = \frac{d}{d\phi} \operatorname{erf} \left(\frac{\phi - \theta_1}{\sigma} \right), & \phi < \theta_1 \\ \delta_j(x, y) = \frac{d}{d\phi} \operatorname{erf} \left(\frac{\phi - \theta_j}{\sigma} \right) - \frac{d}{d\phi} \operatorname{erf} \left(\frac{\phi - \theta_{j-1}}{\sigma} \right), & \theta_{j-1} < \phi < \theta_j, \forall j \neq 1, L \\ \delta_L(x, y) = \frac{d}{d\phi} \operatorname{erf} \left(\frac{\phi - \theta_{L-1}}{\sigma} \right), & \phi > \theta_{L-1} \end{cases} \quad (3.6)$$

dove

$$\frac{d}{d\phi} \operatorname{erf} \left(\frac{\phi - \theta_j}{\sigma} \right) = \frac{2}{\sigma\sqrt{\pi}} \exp \left(-\frac{(\phi - \theta_j)^2}{\sigma^2} \right).$$

La scelta della funzione errore di Gauss fornisce quindi una versione regolarizzata sia di $H(x, y)$ 3.1 che della sua derivata $\delta(x, y)$ 3.2.

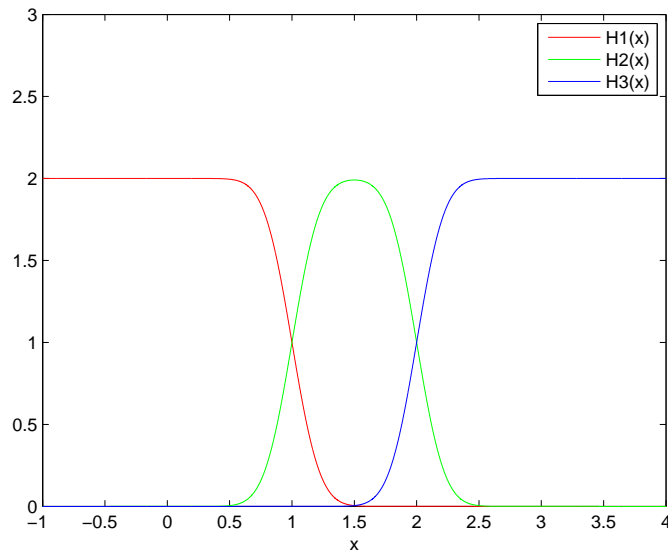


Figura 3.1: $H_j(x)$ regolarizzate

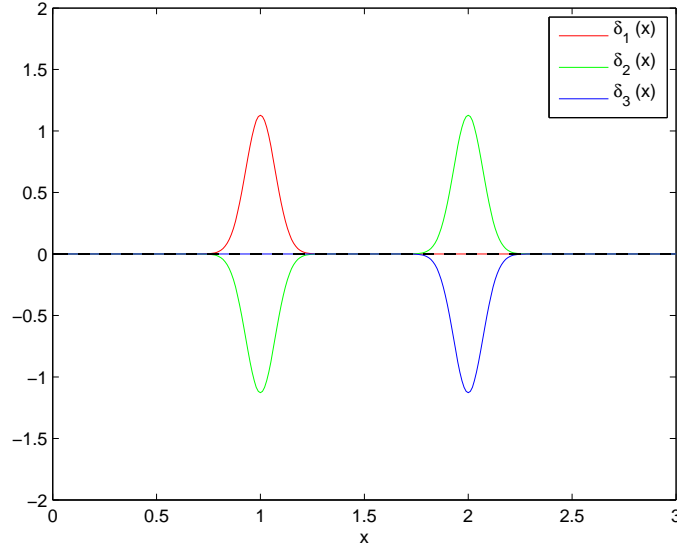


Figura 3.2: $\delta_j(x)$ regolarizzate

Segmentare u_0 in L classi conduce alla definizione di un funzionale di energia che dovrà essere poi minimizzato. Per il caso proposto definiamo il funzionale come

$$F(c_j, \phi) = \sum_{j=1}^L \int_{\Omega} (u_0(\Omega) - c_j)^2 \text{sgn}(u_0(\Omega) - c_j) H_j(\phi) d\Omega + \mu \mathcal{L}(\phi) + \nu \mathcal{A}(\phi) \quad (3.7)$$

dove $\mu \geq 0, \nu \geq 0$ e $\lambda_j > 0$ sono parametri fissati che permettono di controllare il peso di lunghezza, area e distanza da ogni valore c_j .

Nell'equazione 3.7 L differenti $c_j, j = 1, 2, \dots, L$ rappresentano il valor medio di u_0 in ciascuna classe che compone u_0 . In maniera simile a quanto proposto da Chan e Vese [2] tali valori possono essere calcolati come

$$c_j(\phi) = \frac{\int_{\Omega} u_0(\Omega) H_j(\phi(\Omega)) d\Omega}{\int_{\Omega} H_j(\phi(\Omega)) d\Omega}. \quad (3.8)$$

Mantenendo c_j fissato e minimizzando l'equazione 3.7 rispetto a ϕ , parametrizzando in direzione discendente con un tempo artificiale t possiamo formalmente scrivere l'equazione di Eulero-Lagrange associata al funzionale con condizioni iniziali $\phi(0, x, y) = \phi_0(x, y)$ come

$$\begin{aligned} \frac{\partial \phi}{\partial t} = & \sum_{i=1}^{L-1} \delta_i(\phi) \left[\nu + \mu \operatorname{div} \left(\frac{\nabla(\phi - th_i)}{|\nabla(\phi - th_i)|} \right) \right] \\ & + \sum_{j=1}^L \delta_j(\phi) [\lambda_j (u_0 - c_j)^2 \operatorname{sgn}(u_0 - c_j)]. \end{aligned} \quad (3.9)$$

dove abbiamo posto $\nu = 0$. Osserviamo che per $L = 2$ il metodo coincide con la segmentazione proposta nei metodi classici che utilizzano i level set.

3.2 Aspetti implementativi

Nell'implementazione numerica i parametri sono stati fissati a $\lambda_j = \lambda = 1 \forall j \in 1, \dots, L, \nu = 0, \Delta t = 1$. Inoltre è stato posto $\sigma = 0.25$.

Il parametro di regolarizzazione μ può essere modificato per controllare la dimensione degli oggetti che devono essere riconosciuti: se vogliamo segmentare oggetti di grandi dimensioni trascurando quelli di piccole dimensioni il parametro dovrà essere grande e viceversa. Infine il numero delle classi L deve essere fissato a priori per poter segmentare l'immagine in un numero prestabilito di classi.

Le immagini in seguito riportate provengono dal database OASIS [6] (<http://www.oasis-brain.org/>) che fornisce sia le immagini sia la segmentazione per il confronto eseguita da personale qualificato. In figura 3.3 viene riportato un esempio di immagine e la segmentazione associata.

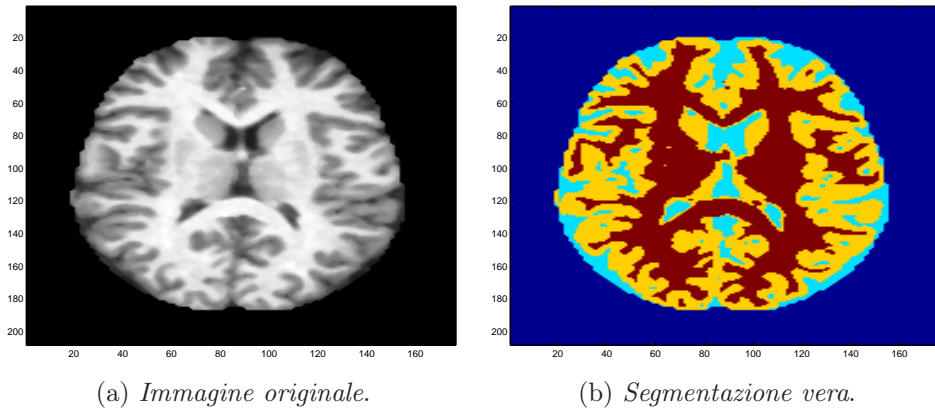


Figura 3.3: Immagini estratte dal database OASIS

Come prima cosa vengono caricate le immagini e filtrate con un filtro mediano bidimensionale deputato alla riduzione del rumore presente. Si procede quindi alla costruzione della funzione ϕ_0 e per costruire tale funzione possiamo procedere in vari modi. Il primo è l'impiego di un classificatore K-NN che a partire dall'immagine iniziale e dal numero di livelli prefissato, segmenta automaticamente l'immagine nel numero di classi desiderato. Il metodo è stato impiegato in una fase preliminare e poi scartato in quanto l'inizializzazione poteva condurre a delle condizioni iniziali troppo specifiche che conducevano l'algoritmo ad un'errata convergenza in quanto si raggiungeva un minimo locale del funzionale e non il minimo globale che corrisponde alla corretta segmentazione. Si è pensato allora di impiegare un'inizializzazione più generica, ottenuta mediante una funzione costante a tratti che assume valori $L/2$ all'interno di regioni circolari tra loro intersecate 3.4.

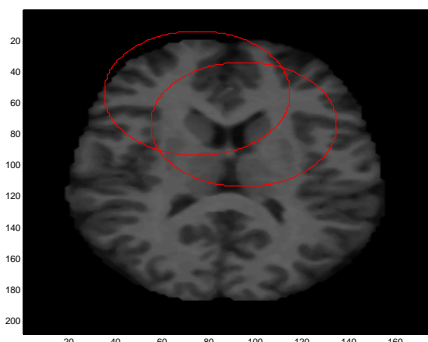


Figura 3.4: Inizializzazione: $\phi_0(x, y)$

Si passa ora all'evoluzione vera e propria. Per ogni livello vengono calcolate la funzione H_i , il livello medio di grigio c_i e i termini di regolarizzazione. A partire da questi valori è possibile calcolare il successivo passo di evoluzione di ϕ come

$$\phi_{t+1} = \phi_t + \frac{\partial \phi}{\partial t} \Delta t \quad (3.10)$$

e procedere iterativamente con tale metodo fino alla convergenza dell'algoritmo che può essere verificata quando la variazione tra i livelli di grigio all'interno delle regioni raggiunge un valore stabile o minore di una certa variazione percentuale tra iterazioni successive. In figura 3.5 la segmentazione ottenuta.

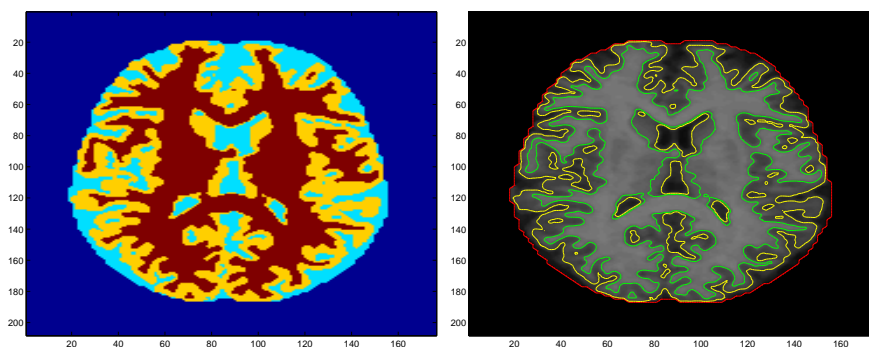


Figura 3.5: Fine evoluzione: $\phi(x, y)$

Presentiamo ora i risultato dell'applicazione del metodo a un'immagine costante a tratti in scala di grigi, composta da quattro aree caratterizzate da un livello di grigio differente 3.6 ed al fantoccio originale di Shepp-Logan 3.7. Entrambe le simulazioni sono state effettuate impostando $L = 4$ livelli.

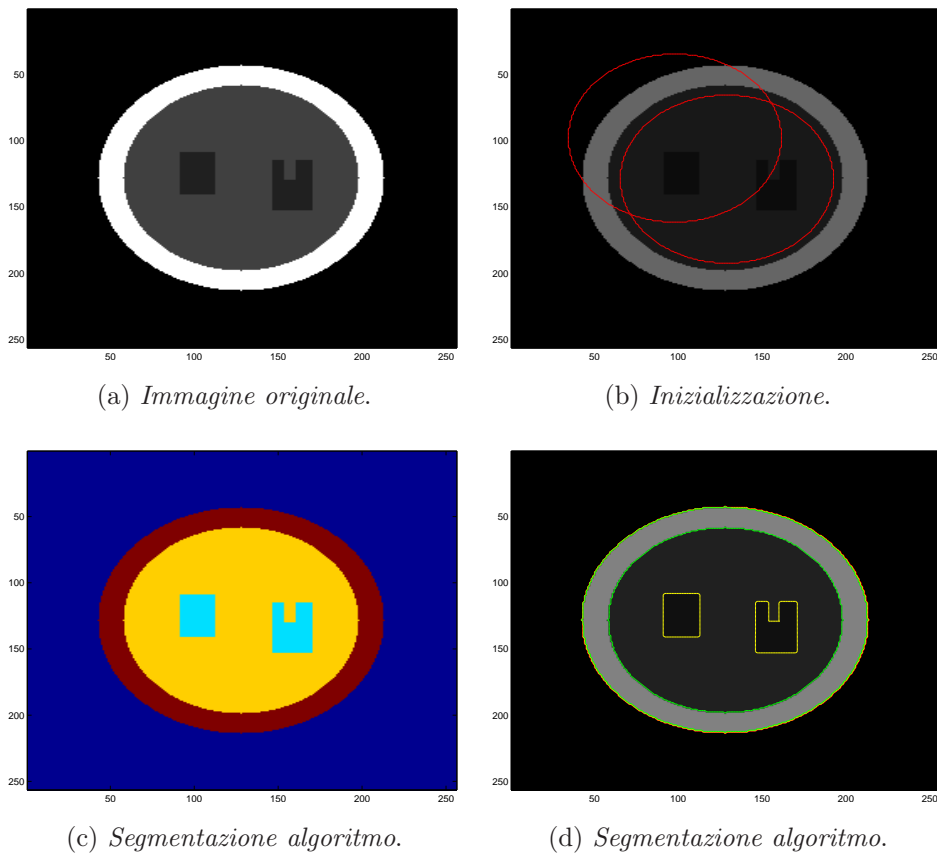


Figura 3.6: Simulazione

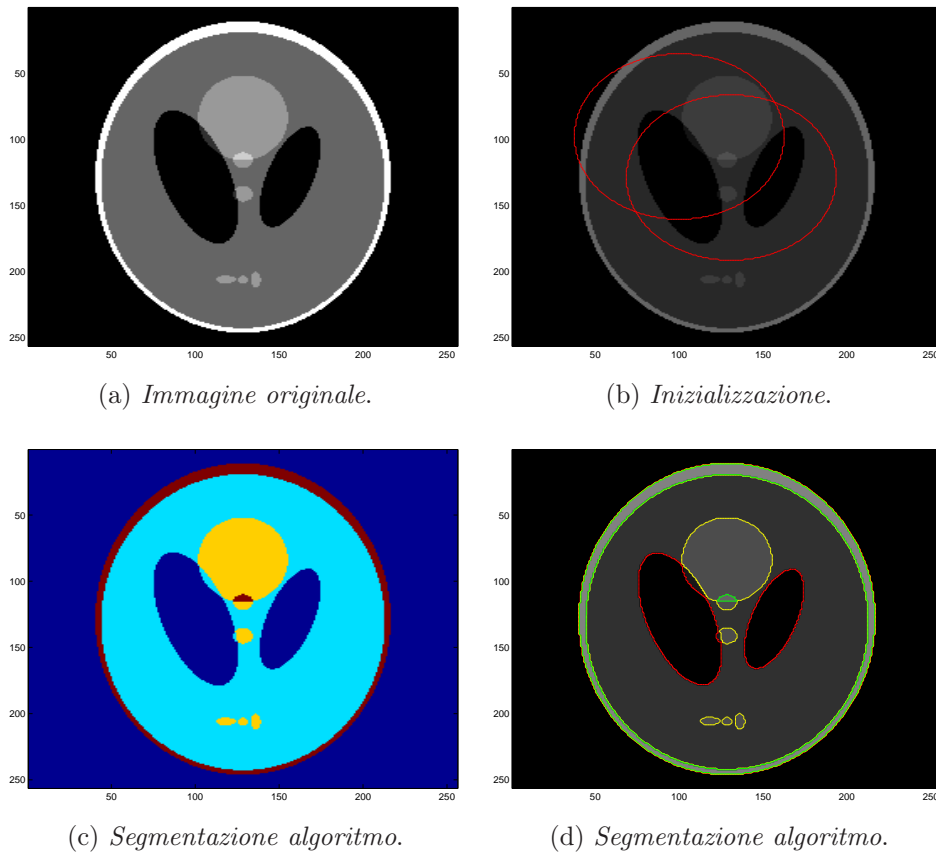


Figura 3.7: Shepp-Logan phantom

Capitolo 4

Risultati

Vengono ora presentati i risultati ottenuti dall'applicazione del metodo proposto a varie immagini sia reali che simulate, riportando sia risultati qualitativi che quantitativi, seguiti da una breve discussione sui risultati ottenuti.

4.1 Analisi qualitativa

Nelle seguenti figure vengono presentati i risultati delle segmentazioni su immagini campione provenienti dal database OASIS [6] (<http://www.oasis-brain.org/>). Le immagini sono state acquisite con il dispositivo Siemens 1.5T con sequenziamento MP-RAGE, $TR/TE/TI/TD = 9.7\text{ms} / 4.0\text{ms} / 20\text{ms} / 200\text{ms}$, flip angle=10°. Le dimensioni di ciascuna immagine sono 208×176 pixel. La segmentazione fornita per il confronto è stata fatta basandosi su un atlante anatomico e validata da esperti prima della pubblicazione. Al fine di ottenere una corretta segmentazione dei tessuti sono stati impostati 4 livelli ($L=4$) e la funzione implicita ϕ è stata inizializzata con una funzione costante a tratti ϕ_0 che assume i valori 0.5, 1.5, 2.5, 3.5 in

quattro regioni circolari tra loro intersecate. La scelta di $L = 4$ è motivata dal voler distinguere i livelli associati allo sfondo, fluido cerebro spinale, materia grigia e materia bianca all'interno di ogni immagine.

Vengono di seguito riportate l'immagine originale, la segmentazione fornita e quella ottenuta dall'applicazione del metodo proposto per ognuna delle immagini campione.

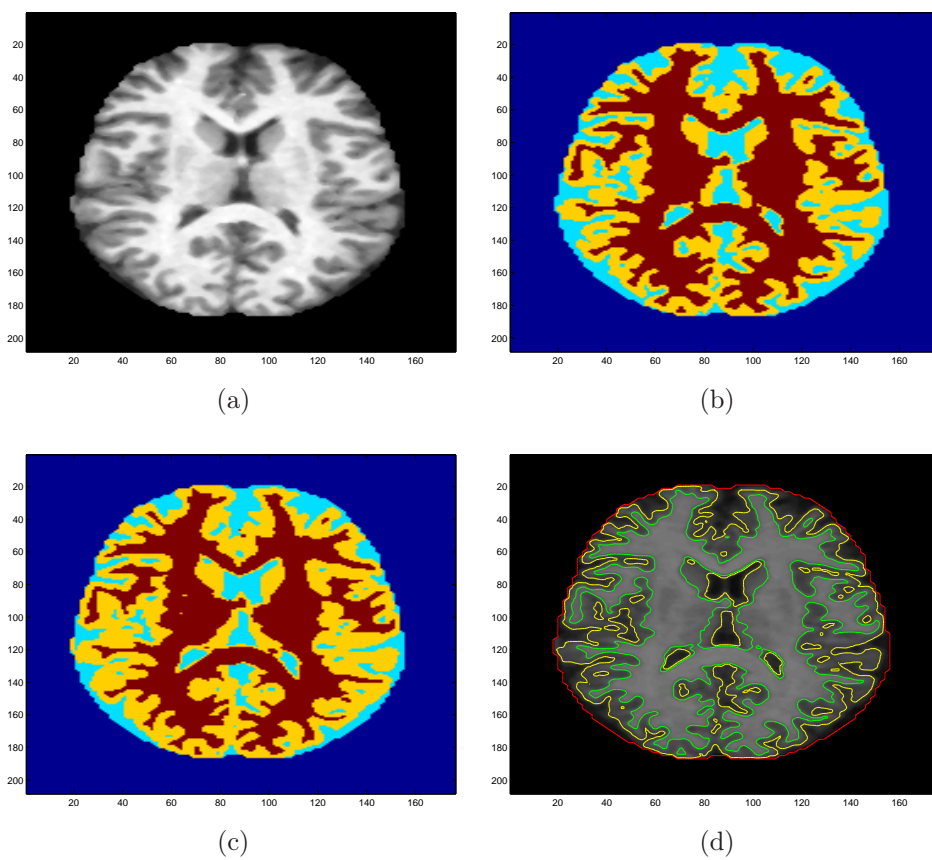


Figura 4.1: (a) Immagine originale, (b) Segmentazione algoritmo, (c) Segmentazione vera, (d) Estrazione dei bordi.

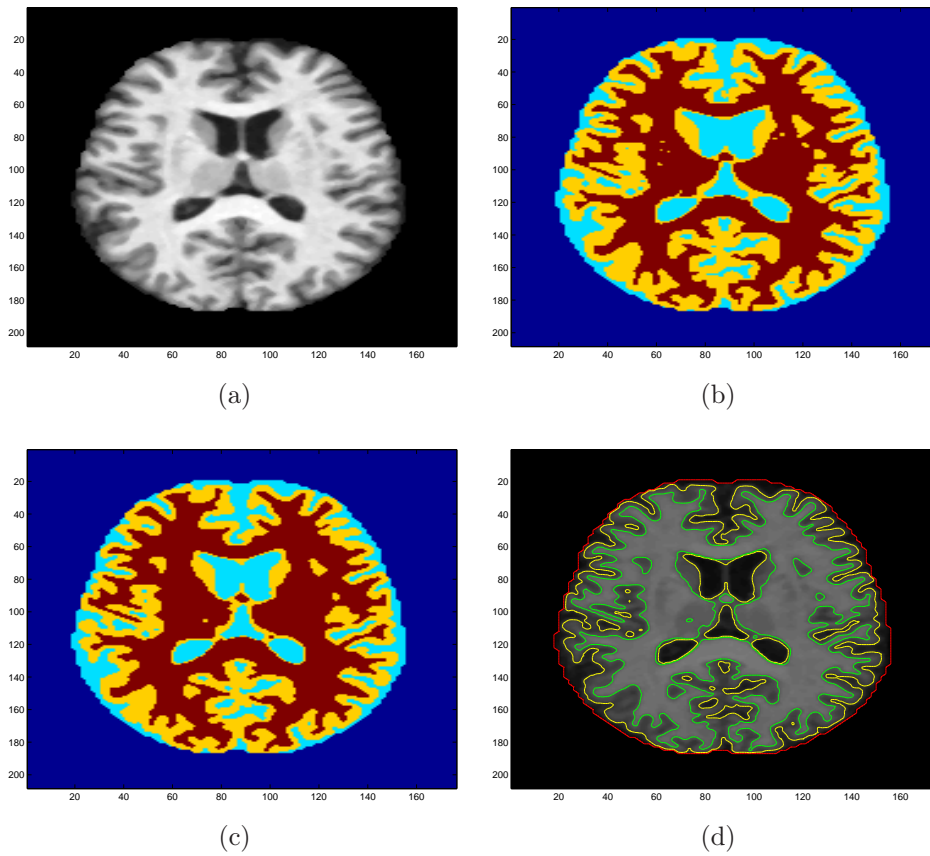


Figura 4.2: (a) Immagine originale, (b) Segmentazione algoritmo, (c) Segmentazione vera, (d) Estrazione dei bordi.

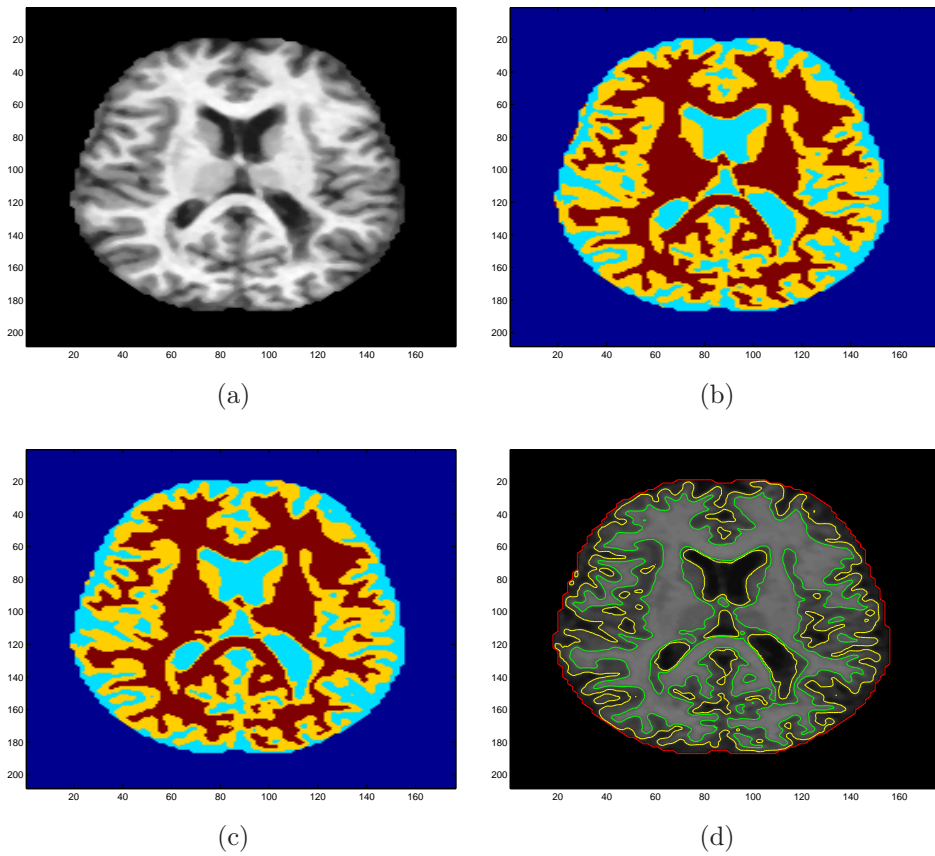


Figura 4.3: (a) Immagine originale, (b) Segmentazione algoritmo, (c) Segmentazione vera, (d) Estrazione dei bordi.

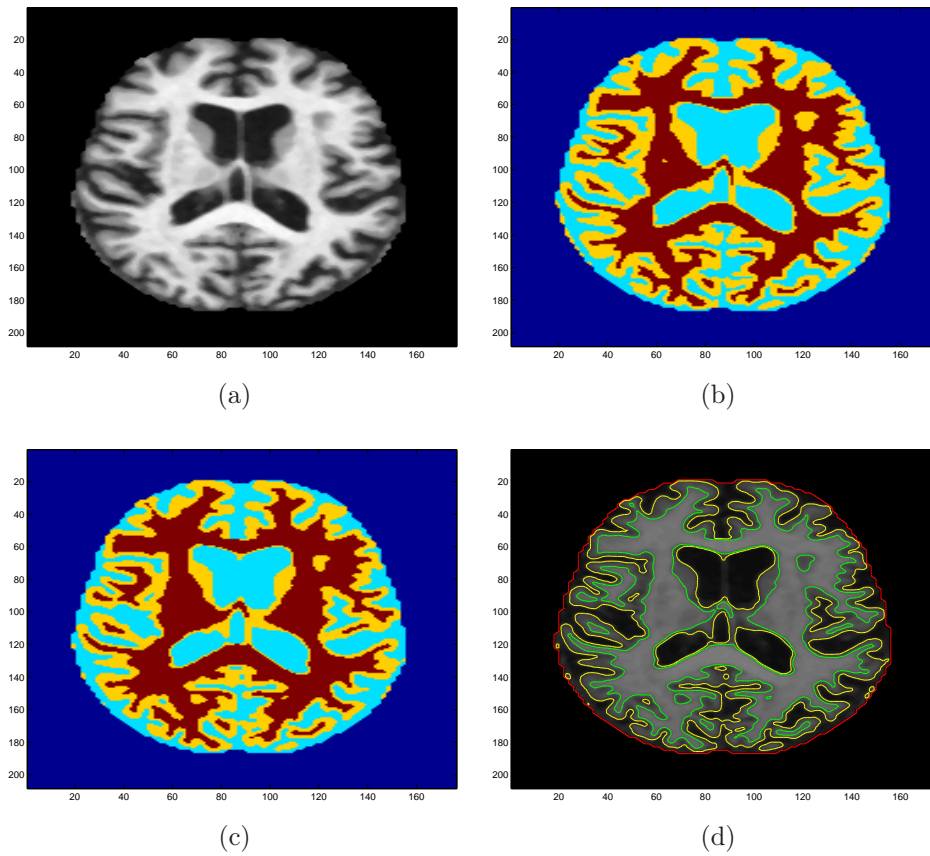


Figura 4.4: (a) Immagine originale, (b) Segmentazione algoritmo, (c) Segmentazione vera, (d) Estrazione dei bordi.

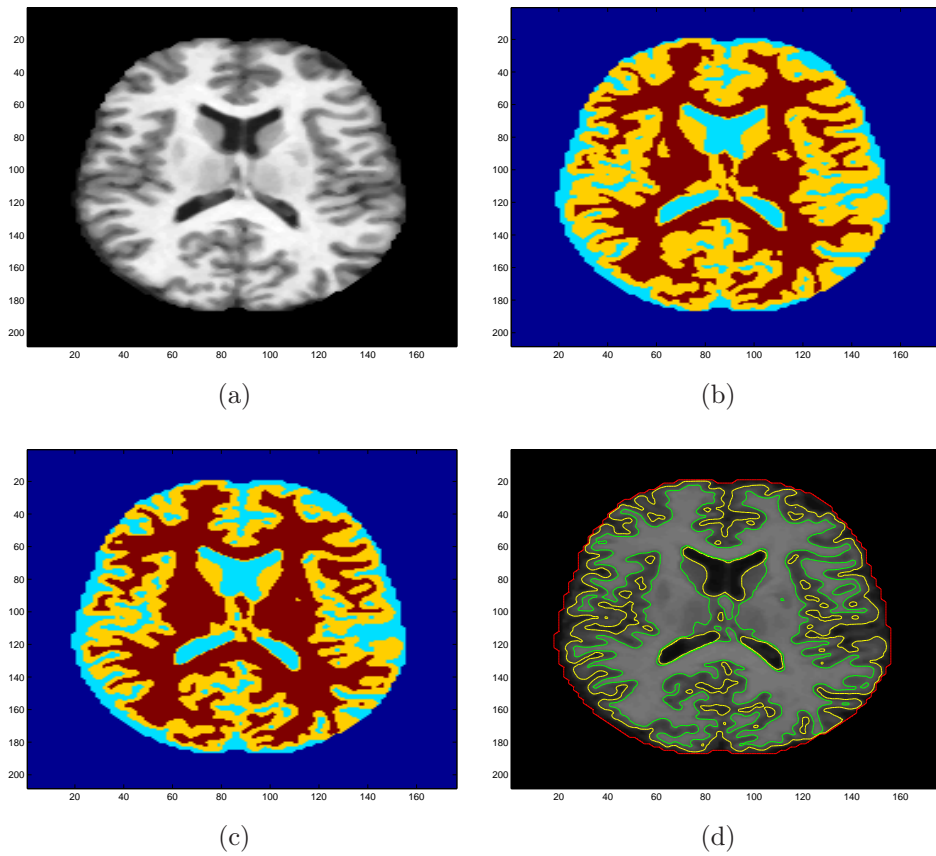


Figura 4.5: (a) Immagine originale, (b) Segmentazione algoritmo, (c) Segmentazione vera, (d) Estrazione dei bordi.

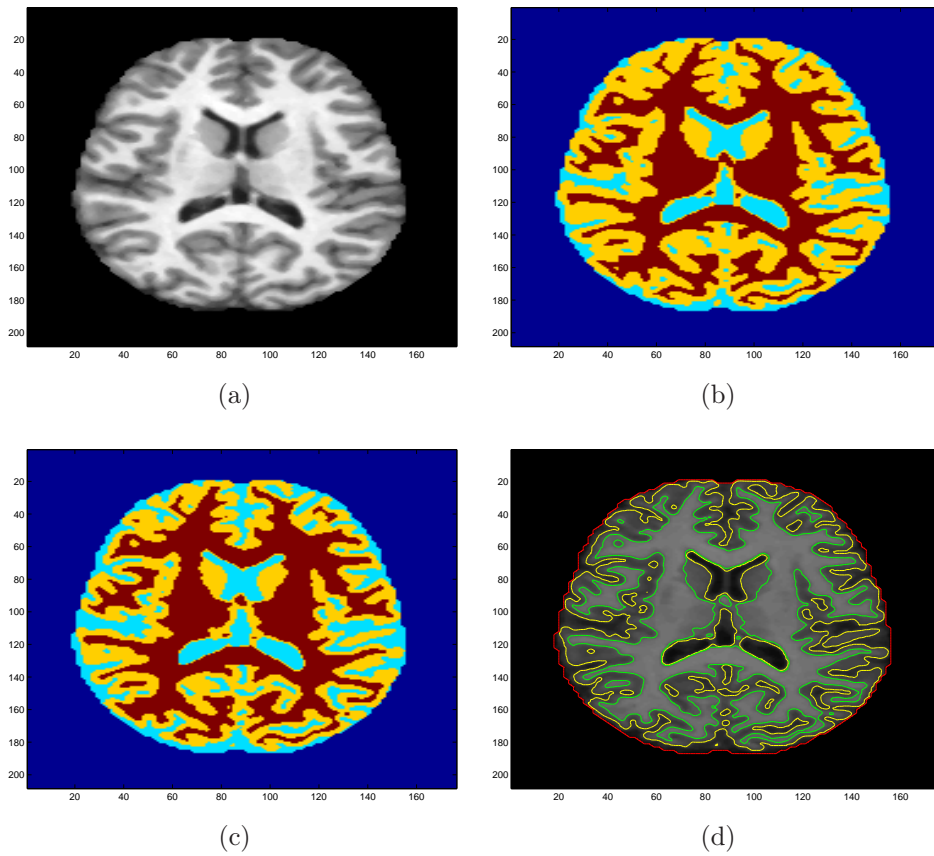


Figura 4.6: (a) Immagine originale, (b) Segmentazione algoritmo, (c) Segmentazione vera, (d) Estrazione dei bordi.

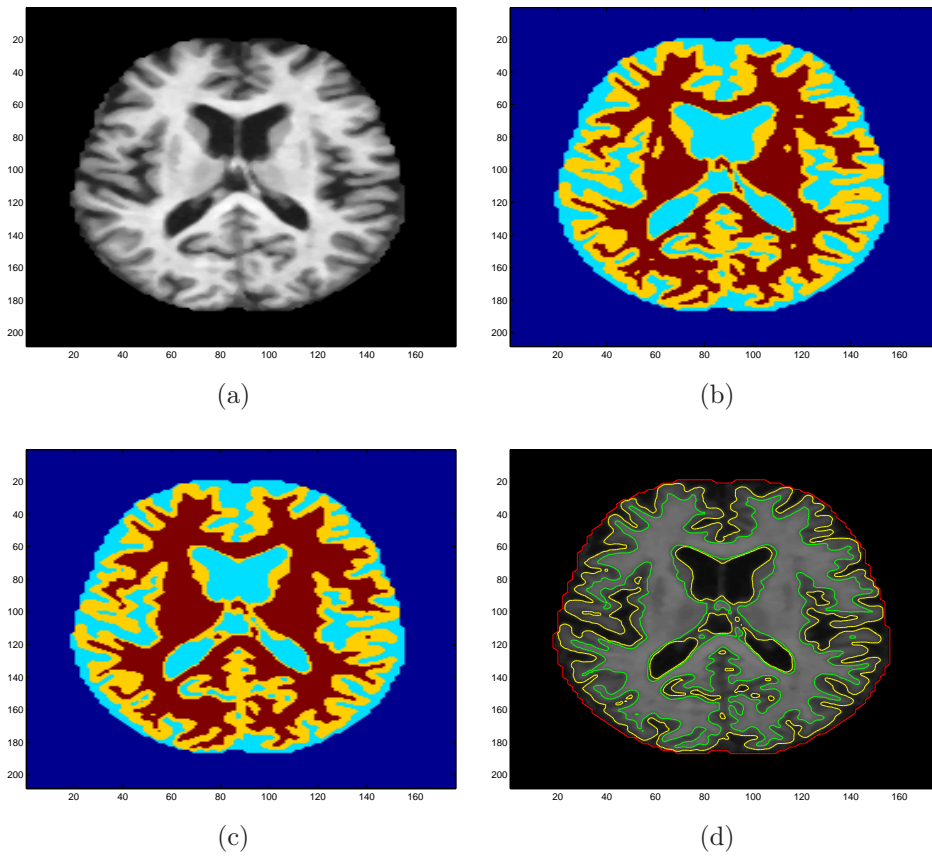


Figura 4.7: (a) Immagine originale, (b) Segmentazione algoritmo, (c) Segmentazione vera, (d) Estrazione dei bordi.

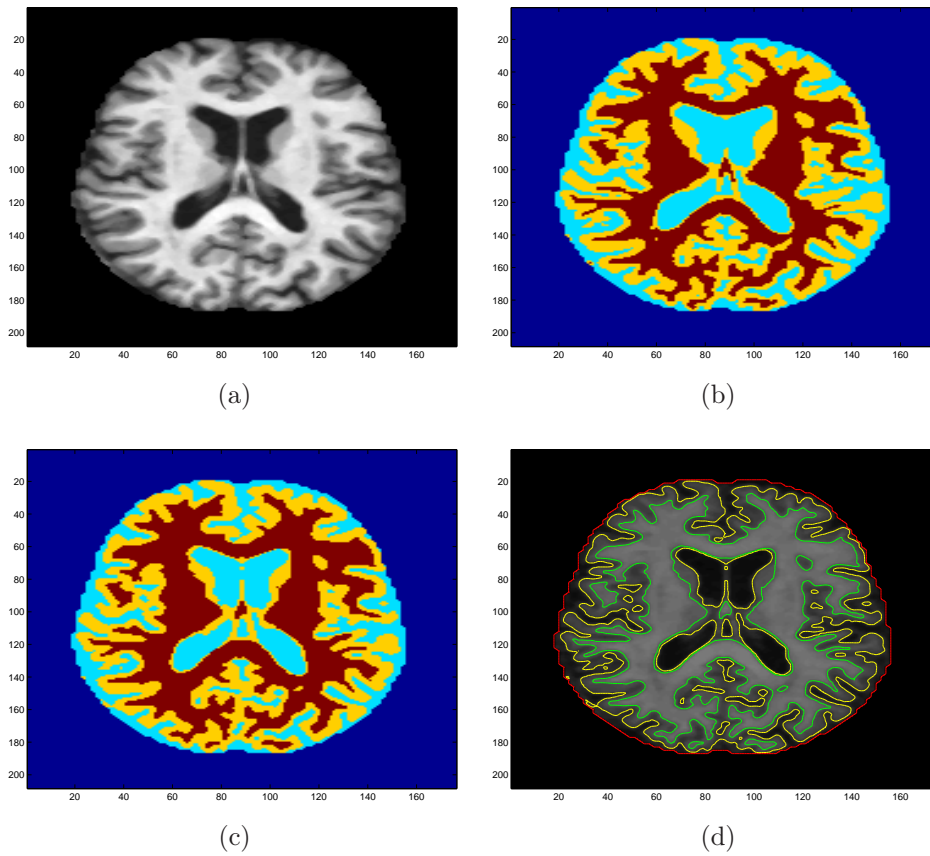


Figura 4.8: (a) Immagine originale, (b) Segmentazione algoritmo, (c) Segmentazione vera, (d) Estrazione dei bordi.

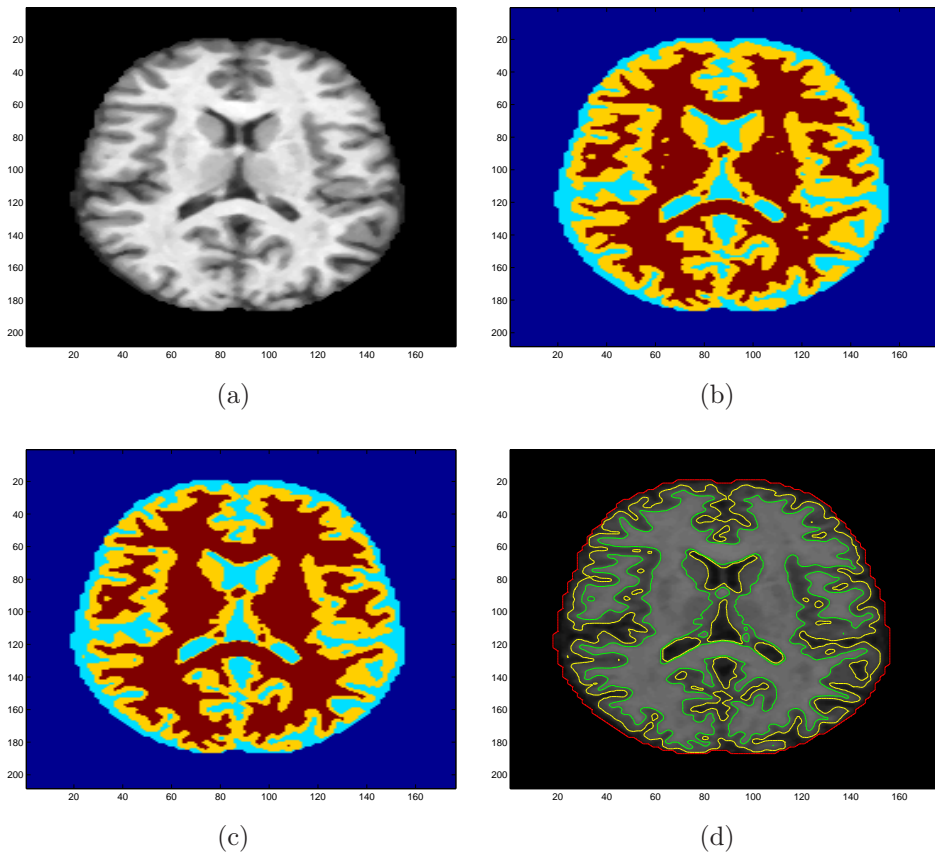


Figura 4.9: (a) Immagine originale, (b) Segmentazione algoritmo, (c) Segmentazione vera, (d) Estrazione dei bordi.

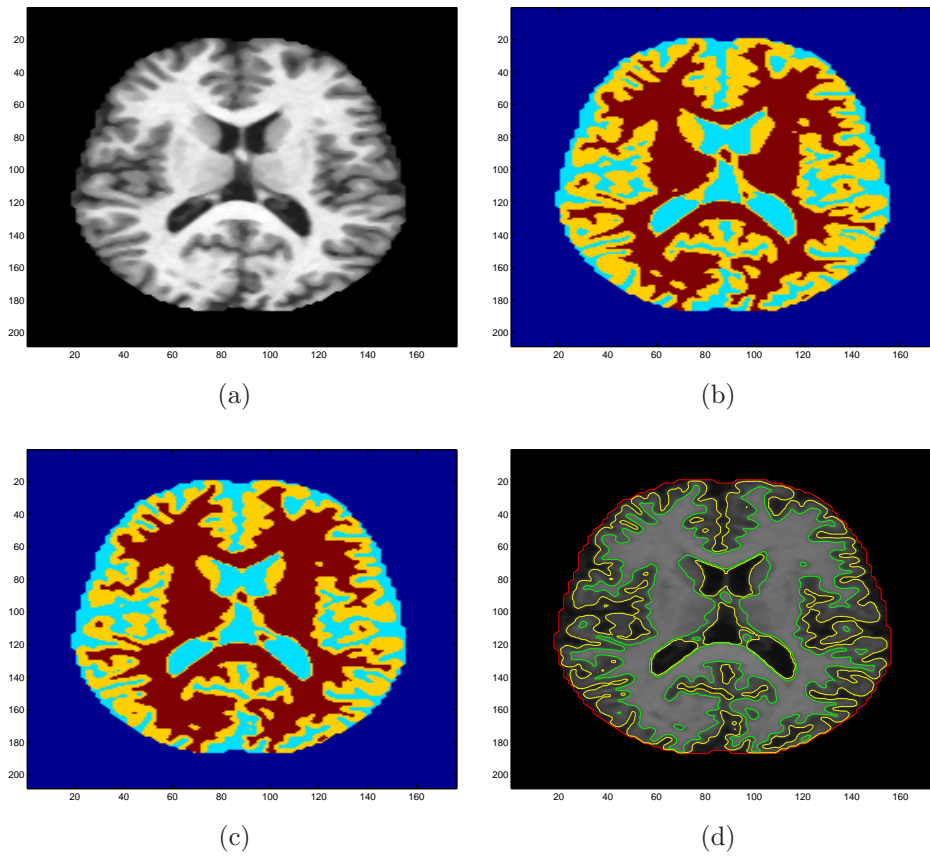


Figura 4.10: (a) Immagine originale, (b) Segmentazione algoritmo, (c) Segmentazione vera, (d) Estrazione dei bordi.

4.2 Analisi quantitativa

Presentiamo ora i risultati quantitativi dell'applicazione dell'algoritmo alle 10 immagini estratte a caso dal database OASIS [6]. Vengono riportati in tabella 4.1 gli indici Dice e sensitività ottenuti definiti come

$$\text{Dice} = \frac{2TP}{2TP + FP + FN} \quad (4.1)$$

e

$$\text{Sensitività} = \frac{A \cap B}{B}. \quad (4.2)$$

Entrambi gli indici sono stati calcolati per ognuno dei livelli in cui è stata segmentata l'immagine. In particolare l'indice Dice 4.1 è calcolato comparando l'area segmentata correttamente (TP) con l'area segmentata in eccesso (FP) e quella in difetto (FN), mentre per la sensitività 4.2 vengono confrontate l'intersezione tra l'area identificata dall'algoritmo (A) e quella fornita dalla segmentazione manuale (B).

	MB		MG		FCS	
	Dice	Sens	Dice	Sens	Dice	Sens
media	0.95	0.99	0.90	0.85	0.92	0.96
varianza	0.01	0.01	0.02	0.04	0.05	0.03
max	0.96	1.00	0.92	0.90	0.96	0.99
min	0.93	0.96	0.88	0.79	0.80	0.91

Tabella 4.1: Risultati numerici: **MB** = Materia bianca, **MG** = Materia Grigia, **FCS** = Fluido cerebro-spinale

4.3 Considerazioni

Nello sviluppo di questa tesi si è cercato di superare le limitazioni imposte dai metodi level set fino ad ora proposti che permettono la segmentazione in due sole fasi con una singola funzione implicita. Introducendo diverse soglie ognuna rappresentativa di una singola fase e riformulando il funzionale che guida l'evoluzione siamo riusciti ad ottenere lo scopo che ci eravamo prefissati. I risultati sperimentali confermano sia qualitativamente che quantitativamente la bontà dell'approccio utilizzato. Inoltre il metodo è robusto a differenti inizializzazioni e non ha problemi di convergenza, aspetti che rimarcano la buona qualità dell'algorithmo stesso.

Bibliografia

- [1] S. Osher and J. A. Sethian, “Fronts propagating with curvature dependent speed: Algorithms based on hamilton-jacobi formulations,” *Journal of Computational Physics*, vol. 79, no. 1, pp. 12–49, 1988.
- [2] T. F. Chan and L. A. Vese, “Active contours without edges,” *IEEE Transactions on image processing*, vol. 10, pp. 266–277, February 2001.
- [3] T. F. Chan and L. A. Vese, “A multiphase level set framework for image segmentation using the mumford and shah model,” *International Journal of Computer Vision*, vol. 50, no. 3, pp. 271–293, 2002.
- [4] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, vol. 153 of *Applied Mathematical Sciences*. Springer-Verlag.
- [5] J. Lie, M. Lysaker, and X.-C. Tai, “A variant of the level set method and applications to image segmentation,” *Mathematics of Computation*, vol. 75, pp. 1155–1174, July 2006.
- [6] D. S. Marcus, T. H. Wang, J. Parker, J. G. Csernansky, J. C. Morris, and R. L. Buckner, “Open access series of imaging studies (oasis): Cross-sectional mri data in young, middle aged, nondemented, and demented older adults,” *Journal of Cognitive Neuroscience*, vol. 19, no. 9, pp. 1498–1507, 2007.

- [7] C. Li, C.-Y. Kao, J. C. Gore, and Z. Ding, “Implicit active contours driven by local binary fitting energy,” in *Computer Vision and Pattern Recognition*, 2007.

- [8] X.-C. Tai and H. Li, “A piecewise constant level set method for elliptic inverse problems,” *Applied Numerical Mathematics*, vol. 57, May 2007.

Elenco delle figure

2.1	$H(x)$ e $\delta(x)$ regolarizzate	10
2.2	Metodo Chan-Vese	10
2.3	Multiphase	14
3.1	$H_j(x)$ regolarizzate	19
3.2	$\delta_j(x)$ regolarizzate	20
3.3	Immagini estratte dal database OASIS	22
3.4	Inizializzazione: $\phi_0(x, y)$	23
3.5	Fine evoluzione: $\phi(x, y)$	23
3.6	Simulazione	24
3.7	Shepp-Logan phantom	25
4.1	Immagine 1	28
4.2	Immagine 2	29
4.3	Immagine 3	30
4.4	Immagine 4	31
4.5	Immagine 5	32
4.6	Immagine 6	33
4.7	Immagine 7	34
4.8	Immagine 8	35
4.9	Immagine 9	36

4.10 Immagine 10	37
----------------------------	----