

POLITECNICO DI MILANO

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

Corso di Laurea Magistrale in

Ingegneria Matematica



Un approccio probabilistico per la segmentazione di immagini mediche tramite contorni attivi parametrici

Relatore: Prof.ssa Alessandra GUGLIELMI

Correlatore: Ing. Elena FAGGIANO

Tesi di laurea di:
Francesca PIZZI
Matricola 781883

Anno Accademico 2012/2013

Ringraziamenti

Desidero ringraziare la Professoressa Alessandra Guglielmi per la disponibilità con cui mi ha seguito nella stesura di questo lavoro.

Rivolgo un particolare ringraziamento all'Ing. Elena Faggiano per avermi incoraggiato e seguito con pazienza e competenza in questi mesi.

Questo lavoro non sarebbe stato inoltre possibile senza il Professor Davide Ambrosi che, oltre ad avermi proposto questa tesi, mi ha trasmesso l'interesse per l'applicazione della matematica in ambito medico.

Ringrazio la mia famiglia per avermi sempre sostenuto e aver creduto in me anche nei momenti più difficili.

Un grazie di cuore va infine a tutti gli amici, la loro compagnia è stata un dono prezioso.

Indice

Elenco delle figure	3
Elenco delle tabelle	5
Abstract	6
Introduzione	7
1 Segmentazione delle immagini	10
1.1 L'immagine digitale	10
1.2 Definizione di segmentazione	11
1.3 Metodi classici di segmentazione	12
1.4 Snakes	15
1.5 Energia interna	16
1.6 Energia esterna	17
1.7 Minimizzazione dell'energia	17
1.8 Scelta della forza esterna	22
1.8.1 Forza deterministica	22
1.8.2 Forza probabilistica	24
1.8.3 Forza combinata	27
1.9 Filtraggio delle immagini	27
2 Modelli mistura	30
2.1 Modello parametrico	30
2.1.1 Metodo della Massima Verosimiglianza	31
2.1.2 Applicazione del modello ai dati	32
2.2 Modello bayesiano non parametrico	34
2.2.1 Processo di Dirichlet	35
2.2.2 Dirichlet Process Mixture models	37
2.2.3 Campi aleatori di Markov	38
2.2.4 DPM vincolato da MRF	39

2.2.5	Simulazione dalla posterior del modello	41
2.2.6	Applicazione alle immagini	42
2.2.7	Estensione del modello: informazione ai bordi	45
2.2.8	Convergenza del modello	46
3	Risultati sperimentali	49
3.1	Immagine test	49
3.2	Valvola cardiaca	55
3.3	Cervello	62
3.3.1	Ventricolo destro	64
3.3.2	Ventricolo sinistro	66
3.3.3	Materia grigia intorno ai ventricoli	68
3.3.4	Materia grigia	72
3.3.5	Materia bianca	74
3.4	Ventricolo	77
3.4.1	Ventricolo sinistro	77
3.4.2	Miocardio	81
4	Conclusioni	87
	Bibliografia	89

Elenco delle figure

1.1	Esempi di metodi classici di segmentazione.	14
1.2	Esempio di curva parametrica che rappresenta uno snake. . . .	16
1.3	Comportamento dello snake sull'immagine test, inizializzando la curva all'esterno dell'oggetto.	21
1.4	Comportamento dello snake sull'immagine test, inizializzando la curva all'interno dell'oggetto.	21
1.5	Applicazione della forza deterministica all'immagine test. . . .	23
1.6	Segmentazione errata della materia bianca mediante la forza deterministica.	23
1.7	Applicazione della forza probabilistica all'immagine test. . . .	24
1.8	Confronto tra forza deterministica e forza probabilistica sul- l'immagine di un cervello.	26
1.9	Funzione gaussiana bidimensionale a media nulla e varianza 4.	28
1.10	Esempio di convoluzione dell'immagine con un filtro gaussiano	29
2.1	Esempio di immagine in scala di grigi con relativo istogramma	33
2.2	Equalizzazione di un'immagine medica.	34
2.3	Confronto della clusterizzazione di un'immagine medica me- diante il metodo DPM/MRF con e senza l'informazione ai bordi.	46
2.4	Potenziale non parametrico $P(x, y)$ dell'immagine test, otte- nuto al variare del numero di iterazioni.	47
3.1	Immagine test - confronto di $P(x, y)$ per ogni modello.	50
3.2	Immagine test - campi di forze.	51
3.3	Zoom $F_{parametrica}$	52
3.4	Snake applicato all'immagine test - inizializzazione interna. . .	53
3.5	Snake applicato all'immagine test - inizializzazione esterna. . .	54
3.6	Valvola cardiaca - confronto di $P(x, y)$ per ogni modello. . . .	55
3.7	Valvola cardiaca - confronto dei campi di forze.	56
3.8	Snake applicato alla valvola cardiaca - inizializzazione interna all'orifizio valvolare.	58

3.9	Snake applicato alla valvola cardiaca - inizializzazione esterna all'orifizio valvolare.	59
3.10	Zoom sulla segmentazione di una zona dell'orifizio valvolare. . .	61
3.11	Cervello - confronto di $P(x, y)$ per ogni modello.	62
3.12	Cervello - confronto dei campi di forze.	63
3.13	Snake applicato al cervello - inizializzazione esterna al ventricolo destro.	65
3.14	Snake applicato al cervello - inizializzazione esterna al ventricolo sinistro.	67
3.15	Ventricoli del cervello - confronto di $P(x, y)$ per ogni modello. . .	68
3.16	Ventricoli del cervello - confronto dei campi di forze.	69
3.17	Snake applicato al cervello - segmentazione della materia grigia intorno al ventricolo destro.	70
3.18	Snake applicato al cervello - segmentazione della materia grigia intorno al ventricolo sinistro.	71
3.19	Snake applicato al cervello per la segmentazione della materia grigia.	73
3.20	Potenziale $P(x, y)$ del cervello ottenuto mediante il modello parametrico al variare del numero di cluster.	74
3.21	Snake applicato al cervello per la segmentazione della materia bianca.	76
3.22	Ventricolo cardiaco - confronto di $P(x, y)$ per ogni modello. . .	77
3.23	Ventricolo cardiaco - confronto dei campi di forze.	78
3.24	Snake applicato al ventricolo cardiaco sinistro.	79
3.25	Snake applicato al ventricolo cardiaco sinistro - segmentazione comprendendo i muscoli capillari.	81
3.26	Equalizzazione dell'immagine di un ventricolo	82
3.27	Ventricolo cardiaco - confronto di $P(x, y)$ dell'immagine equalizzata per ogni modello.	83
3.28	Ventricolo cardiaco - confronto dei campi di forze ricavati dall'immagine equalizzata.	84
3.29	Snake applicato al miocardio.	86

Elenco delle tabelle

3.1	Parametri dello snake applicato all'immagine test - inizializzazione interna.	51
3.2	Parametri dello snake applicato all'immagine test - inizializzazione esterna.	52
3.3	Parametri dello snake applicato alla valvola cardiaca - inizializzazione interna.	57
3.4	Parametri dello snake applicato alla valvola cardiaca - inizializzazione esterna.	57
3.5	Parametri dello snake applicato al ventricolo destro del cervello.	64
3.6	Parametri dello snake applicato al ventricolo sinistro del cervello.	66
3.7	Parametri dello snake applicato alla materia grigia intorno al ventricolo destro del cervello.	70
3.8	Parametri dello snake applicato alla materia grigia intorno al ventricolo sinistro del cervello.	71
3.9	Parametri dello snake applicato alla materia grigia.	72
3.10	Parametri dello snake applicato alla materia bianca.	75
3.11	Parametri dello snake applicato al ventricolo cardiaco sinistro.	79
3.12	Parametri dello snake applicato al ventricolo cardiaco sinistro comprendendo il muscolo papillare.	80
3.13	Parametri dello snake applicato al miocardio.	85

Abstract

In questo lavoro di tesi si considera il problema della segmentazione delle immagini mediche, ovvero il partizionamento delle immagini in regioni omogenee e successivamente il riconoscimento del bordo di una particolare regione d'interesse. In questa tesi si propone di incorporare un approccio probabilistico nel classico metodo ai contorni attivi parametrici, in modo da ottenere una tecnica più robusta ed efficiente rispetto ai metodi tradizionalmente utilizzati. In particolare, si propongono due tipi di modelli probabilistici per la classificazione: uno parametrico, costruito mediante il Gaussian Mixture Model, ed uno bayesiano non parametrico, costruito mediante il Dirichlet Process Mixture model vincolato dai campi aleatori di Markov. Un'ulteriore novità è fornita dalla combinazione di un termine deterministico, classicamente utilizzato, con il termine probabilistico costruito mediante il modello parametrico. Attraverso alcuni esempi applicativi, si confrontano i risultati ottenuti attraverso questi approcci con quelli prodotti utilizzando il solo termine deterministico. Si mostra come i metodi probabilistici portino a risultati di segmentazione più precisi rispetto agli altri modelli.

Introduzione

Negli ultimi anni l'utilizzo di immagini ha giocato un ruolo importante nell'ambito medico. La diagnostica attraverso le immagini permette infatti di raccogliere informazioni utili fornite da tecniche come la tomografia computerizzata o la risonanza magnetica, senza dover ricorrere a metodi invasivi. L'analisi delle immagini svolge un ruolo importante nella diagnosi e nel trattamento delle malattie; essa è diventata nel tempo uno strumento di pianificazione e simulazione chirurgica, che consente al tempo stesso di accertare la presenza di patologie e di pianificare un approccio operativo ottimale. In questo contesto gioca un ruolo fondamentale la segmentazione delle immagini.

Con il termine segmentazione si intende il partizionamento dell'immagine in regioni omogenee e la successiva identificazione del contorno di una particolare regione. Applicazioni che coinvolgono la segmentazione sono ad esempio il rilevamento del profilo delle coronarie (Kass, Witkin e Terzopoulos, 1988; Schaar et al., 2011), l'identificazione di stenosi nelle arterie (Anders et al., 2012), l'individuazione e misurazione di tumori (Rakesh e Ravi, 2012), ecc. Inoltre, il riconoscimento di contorni effettuato su una serie di immagini come quella fornita da una risonanza magnetica (Cheng e Mei, 2008) o da una tomografia computerizzata (Hohne e Bernstein, 1986) permette di ricostruire la forma tridimensionale degli oggetti. Una procedura computazionale che automaticamente identifichi, isoli e metta in evidenza determinate regioni d'interesse si rivela sicuramente d'aiuto in questi ambiti.

A causa della varietà di applicazioni e della difficoltà del problema, nel tempo sono stati sviluppati numerosi algoritmi per la segmentazione delle immagini e la ricerca è tuttora attiva. Un grosso problema deriva dalle caratteristiche molto diverse che possono assumere le immagini. In alcuni casi, infatti, i contorni sono ben definiti, mentre in altri casi sono difficilmente riconoscibili; per questo motivo non esiste un approccio universale, ma è necessario valutare caso per caso il metodo più adatto.

I modelli deformabili hanno suscitato nell'analisi di immagini cliniche grande interesse ed attenzione (Tsechpenakis, 2011). Essi furono introdotti

per la prima volta da Kass, Witkin e Terzopoulos (1988) con il nome di *snakes*.

In questo lavoro si prende in considerazione un particolare tipo di modelli deformabili, i contorni attivi parametrici; essi consistono nell'inizializzare una curva (snake) e nel deformarla sotto l'azione di forze che minimizzano una particolare funzione obiettivo che rappresenta l'energia dello snake. Affinché la segmentazione mediante questo tipo di metodi sia efficace, è necessario scegliere accuratamente tali forze. Classicamente sono state utilizzate delle forze di tipo deterministico (Tsechpenakis, 2011), basate cioè su proprietà dell'immagine; si è visto però che in molti casi la segmentazione fallisce, a causa dell'eccessivo rumore presente nelle immagini o della mancanza di contrasto. Si è allora pensato di introdurre delle forze di tipo probabilistico che includano le probabilità a posteriori (Yang, Tannenbaum e Giddens, 2004). In questo modo si previene il risultato indesiderato che la curva si blocchi su dei minimi locali, e inoltre si è visto che la segmentazione ottenuta mediante queste forze è più precisa. Questo lavoro propone un metodo di identificazione dei contorni basato su forze ricavate mediante due tipi di modelli probabilistici.

Il primo modello è di tipo parametrico e prevede il calcolo di probabilità condizionate ai dati dell'immagine, ricavate classificando l'immagine mediante un modello a mistura gaussiana. La novità introdotta in questa tesi rispetto al lavoro preso in considerazione (Yang et al., 2007) è il tentativo di combinare insieme la forza probabilistica con quella deterministica. Questo approccio si è dimostrato efficace e ha reso la segmentazione più veloce e stabile. Rimangono però degli svantaggi: essendo il modello parametrico, è necessario specificare a priori un numero di cluster in cui classificare l'immagine, e inoltre l'immagine deve comunque essere filtrata prima per ridurre il rumore. Tale novità si rivela quindi utile nei casi in cui le forze probabilistica e deterministica riescano ad identificare bene i contorni, velocizzando il processo di segmentazione. Nelle immagini a basso contrasto, dove i contorni non sono cioè ben definiti, anche la combinazione delle forze non si rivela efficace. Per questo motivo si è pensato di utilizzare un secondo modello, di tipo bayesiano non parametrico, basato sui Dirichlet Process Mixture models combinati con i campi aleatori di Markov. Il vantaggio del modello non parametrico è che non è necessario specificare a priori il numero di classi, ma solo un "livello di risoluzione" della segmentazione desiderata. Rispetto al lavoro già esistente (Orbanz e Buhmann, 2008), in cui l'approccio non parametrico è utilizzato solo per determinare il numero di classi in cui suddividere l'immagine, in questa tesi si utilizza tale metodo per costruire la forza esterna che guida lo snake, applicazione innovativa in quanto questo particolare modello non parametrico non è mai stato utilizzato per tale scopo. Tale metodo, al

contrario di quello parametrico, riesce ad individuare anche i contorni di immagini a basso contrasto; spesso inoltre la segmentazione è più precisa anche nei casi in cui il modello parametrico fornisca già dei risultati soddisfacenti.

Nel Capitolo 1, dopo una breve introduzione sulla struttura delle immagini digitali, si descrive il processo di segmentazione ed i metodi classicamente utilizzati. Successivamente viene presentato il modello utilizzato in questo lavoro, cioè il metodo per i contorni attivi o snake. Ci si sofferma in particolare sulla scelta di un'opportuna forza che muoverà lo snake: si descrivono una forza deterministica, tradizionalmente utilizzata, e una probabilistica. La prima novità che si introduce in questo lavoro è una combinazione pesata di queste due forze, in modo da velocizzare ed irrobustire il metodo di identificazione dei contorni. Si descrive infine il processo di filtraggio delle immagini per ridurre il rumore.

Nel Capitolo 2 si presentano i due modelli per la classificazione utilizzati per il calcolo della forza probabilistica. Il primo modello è di tipo parametrico ed è costruito mediante Gaussian Mixture Model. Il secondo è un modello bayesiano non parametrico, costruito mediante il Dirichlet Process Mixture model vincolato dai campi aleatori markoviani. La novità di questo approccio, mai utilizzato finora per i contorni attivi, porta a miglioramenti nella segmentazione delle immagini mediche, con una minore interazione da parte dell'utente rispetto al modello parametrico.

Nel Capitolo 3 si riportano tutti i risultati sperimentali con i nuovi metodi proposti, confrontandoli con i risultati ottenuti mediante un modello deterministico tradizionalmente utilizzato.

Nel Capitolo 4 si riassumono le novità introdotte in questa tesi e i risultati ottenuti, con possibili sviluppi futuri.

Capitolo 1

Segmentazione delle immagini

1.1 L'immagine digitale

Un'immagine è un metodo di rappresentazione di uno o più oggetti reali secondo coordinate spaziali indipendenti. Essa è una distribuzione di tipo bidimensionale o tridimensionale di un'entità fisica. In questa tesi verranno trattate solo immagini bidimensionali.

L'immagine è generata dalla combinazione di una sorgente di energia e dall'assorbimento dell'energia da parte dell'oggetto. Perché possa essere elaborata al computer, deve essere trasformata in un rappresentazione numerica attraverso un processo chiamato *digitalizzazione*.

Un'immagine digitale è descritta da una matrice di numeri che rappresenta la distribuzione di intensità continua di un segnale spaziale. Il segnale spaziale è campionato a intervalli regolari e l'intensità è quantizzata su un numero finito di livelli, detti livelli di grigio (Rangayyan, 2008). L'immagine digitale bidimensionale può essere pensata come una funzione

$$f(x, y) : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R} \quad \forall (x, y) \in \Omega$$

dove f è l'intensità del pixel e x e y definiscono la posizione del pixel. L'immagine ha quindi M righe e N colonne e assume valori di intensità da 0 a $P - 1$.

Le due tecniche con cui sono state acquisite le immagini prese in considerazione in questa tesi sono la *tomografia computerizzata* (TC) e la *risonanza magnetica* (RM). La tomografia computerizzata consiste nell'emissione di fasci di raggi X che, attraversando il corpo del paziente, vengono intercettati da dei sensori che ne misurano l'intensità, diversa a seconda della densità del tessuto incontrato, ed inviano i dati a un computer, dove vengono raccolti ed

elaborati. Il computer è in grado così di ricostruire e visualizzare l'immagine virtuale degli organi interni. Per migliorare la qualità delle immagini, spesso si inietta nel paziente un liquido di contrasto.

La risonanza magnetica permette di ottenere immagini sfruttando una particolare proprietà fisica degli elettroni detta appunto risonanza. Essa consente di visualizzare parti interne del corpo attraverso l'applicazione di un campo magnetico esterno, a cui i diversi organi e tessuti rispondono emettendo diversi tipi di onde. I segnali emessi dal corpo vengono captati dalle bobine del tomografo ed elaborati sotto forma di immagini. Questa tecnica mostra una maggiore risoluzione nel differenziare i tessuti; inoltre, al contrario della TC, non utilizza radiazioni ionizzanti ma campi magnetici con energie di per sé non dannose per l'organismo.

1.2 Definizione di segmentazione

La segmentazione è definita come il processo che partiziona l'immagine in regioni omogenee, dove tutti i pixel che corrispondono ad un oggetto sono raggruppati insieme; queste informazioni possono essere poi utilizzate per identificare il contorno di una particolare regione d'interesse. Lo scopo della segmentazione è semplificare l'immagine in modo da ottenere una rappresentazione più facile da analizzare.

Il raggruppamento dei pixel è basato su criteri di omogeneità specifici, per esempio valori di similarità di attributi come il colore e l'intensità, oppure valori di prossimità spaziale (distanza euclidea, etc.).

Il processo di segmentazione di un'immagine $f(x, y)$ in regioni R_1, \dots, R_n deve soddisfare le seguenti condizioni (Zhang, 2006):

1. $\bigcup_{k=1}^n R_k = f(x, y)$, ossia l'intera immagine deve essere partizionata.
2. Ogni regione R_i soddisfa un criterio di omogeneità $P(R_i)$, cioè $P(R_i) = \text{vero} \forall i = 1, \dots, n$
3. Ogni regione R_i è spazialmente connessa.
4. Pixel appartenenti a regioni confinanti R_i e R_j , quando considerati congiuntamente, non soddisfano il criterio di omogeneità:

$$P(R_i \cup R_j) = \text{falso} \quad \forall i, j \text{ delle } n \text{ regioni}$$

5. R_i è una partizione esclusiva:

$$R_i \cap R_j = \emptyset \text{ con } i \neq j$$

Il predicato di omogeneità $P(\bullet)$ indica la conformità di tutti i pixel di una regione R_i ad un particolare modello della regione stessa.

1.3 Metodi classici di segmentazione

La segmentazione rimane tutt'oggi un processo complicato a causa della grande variabilità delle forme degli oggetti e della variazione della qualità delle immagini; non c'è quindi un singolo metodo che possa essere utilizzato per ogni tipologia di immagine. Inoltre la difficoltà di utilizzo delle varie tecniche può aumentare a causa della presenza di rumore. Per questo motivo alcuni metodi necessitano di un'elaborazione dell'immagine prima o dopo l'algoritmo di segmentazione.

Nell'ambito dei metodi presenti in letteratura, esistono diversi approcci tradizionalmente utilizzati (Dass, Priyanka e Devi, 2012). Si fornisce di seguito una panoramica di quelli più comuni.

Thresholding

Ogni pixel è confrontato con un valore di soglia scelto: se è inferiore, il pixel appartiene allo sfondo, altrimenti appartiene alla regione cercata. Il metodo Thresholding crea quindi un'immagine binaria in cui tutti i pixel sotto la soglia assumono valore 0 e tutti gli altri assumono valore 1. I risultati ottenuti sono molto approssimativi e la limitazione di questo metodo è data dal fatto che solo due classi possono essere generate; inoltre il metodo non prende in considerazione le caratteristiche spaziali dell'immagine.

Edge detection

Questo metodo si basa sul riconoscimento dei contorni; il risultato è un'immagine binaria in cui i contorni assumono valore 1 e tutto lo sfondo assume valore 0. Basandosi sul fatto che un contorno corrisponde ad un cambiamento di intensità dell'immagine, le tecniche usate per l'estrazione dei contorni di solito si avvalgono di filtri derivativi: stima del gradiente nel pixel corrispondente. Gli operatori più comuni utilizzati sono Sobel, Canny, Prewitt, Roberts e l'operatore di Laplace (Gonzalez e Woods, 2008). La sola individuazione dei bordi non permette tuttavia di riconoscere le regioni significative di un'immagine, infatti molti bordi possono non essere chiusi o formare complesse intersezioni. Si tratta però di un buon punto da cui partire per costruire altri metodi di segmentazione.

Region based

Questi algoritmi partizionano l'immagine in regioni simili secondo dei criteri predefiniti. Un metodo di questo tipo è il cosiddetto *region growing*, che richiede in ingresso una serie di regioni, ognuna corrispondente ad un oggetto da segmentare; in seguito le regioni vengono allargate aggiungendo via via il pixel, tra quelli confinanti, con l'intensità più simile al valore medio della regione, finché tutti i pixel non sono allocati. Un altro metodo è il *region splitting and merging*, in cui l'ipotesi iniziale è che l'immagine sia una regione omogenea; se tale ipotesi non è verificata, l'immagine viene suddivisa in regioni più piccole. Questa procedura è poi ripetuta su ognuna delle sottoimmagini fino a quando non sono più necessari ulteriori partizionamenti. Per unire le regioni adiacenti che soddisfano i criteri predefiniti e che risultano distinte, ad ogni iterazione è seguito un merging: date due regioni, se la loro unione soddisfa tali criteri, esse vengono fuse.

Clustering

Il clustering consiste nella suddivisione del dominio iniziale in sottodomini attraverso una tecnica iterativa; il metodo più diffuso è il *K-means*: inizialmente si partiziona l'immagine in k regioni o cluster e si trovano i loro centri; ogni pixel viene poi associato al cluster con centro più vicino in termini di colore, intensità, posizione o una media pesata di questi fattori. In seguito si ricalcola il valore centrale dei cluster e si ripete il passo precedente fino a convergenza. La qualità del risultato dipende dalla misura di similarità utilizzata, dall'implementazione del metodo e dalla scelta del numero di cluster che deve essere fatta a priori.

Reti neurali

Questo tipo di segmentazione è totalmente differente dagli algoritmi convenzionali. L'immagine viene rappresentata come un grafo pesato, in cui i nodi sono associati ad uno o più pixel e i pesi dei lati indicano la somiglianza tra i pixel adiacenti. Per partizionare in modo efficace i nodi si possono sfruttare vari algoritmi (Cheng, Lin e Mao, 1996; Reddick et al., 1997).

In Figura 1.1 sono riportati esempi di alcune tra le tecniche descritte. La Figura 1.1b mostra l'utilizzo del metodo Thresholding, in cui la soglia è stata scelta utilizzando il metodo Otsu: si suppone che siano presenti solo due classi e si calcola la soglia ottima per separarle in modo da minimizzare la varianza intra-classe. In Figura 1.1c è stato utilizzato l'operatore Canny per

individuare i contorni, mentre in Figura 1.1d si mostra l'applicazione del metodo region growing, in cui è stata individuata una particolare regione di interesse.

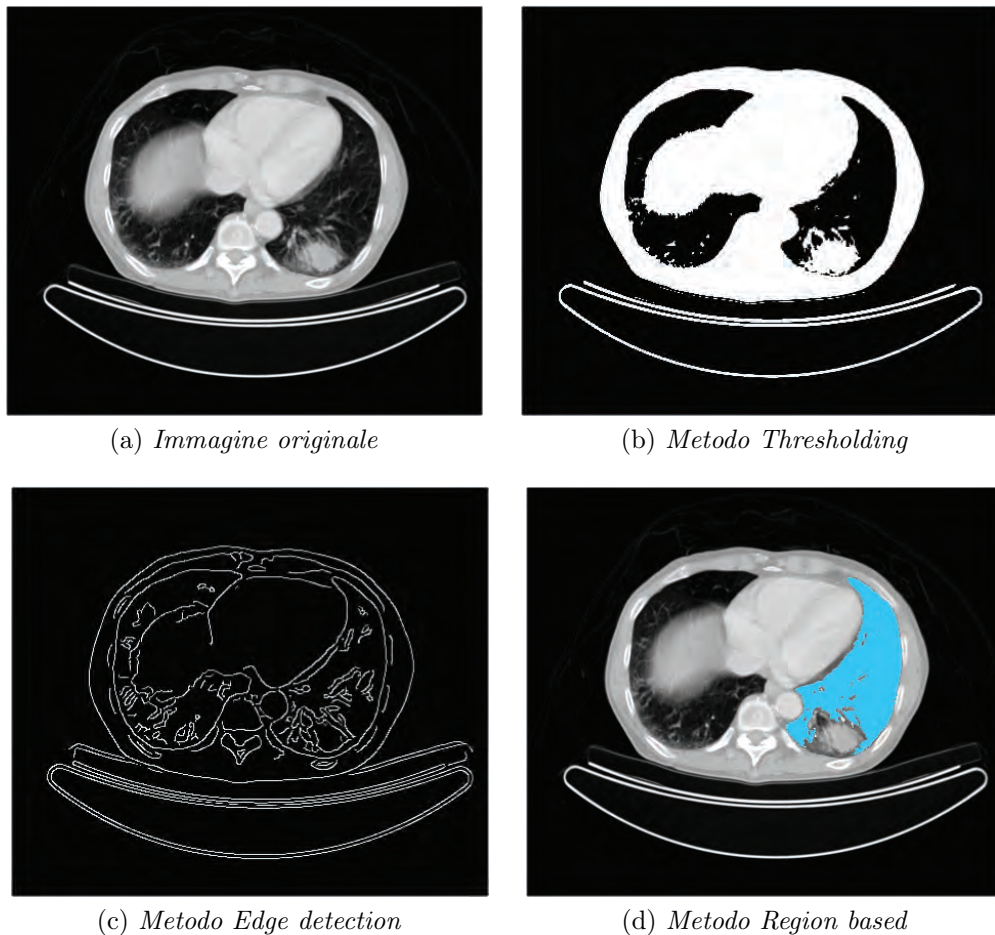


Figura 1.1: Esempi di metodi classici di segmentazione.

Una segmentazione effettuata con queste tecniche richiede però un considerevole carico di esperienza e conoscenza da parte dell'utilizzatore. Inoltre, se i contorni non sono ben definiti o l'intensità non è omogenea nelle regioni di interesse, tali metodi falliscono.

Si osservi che i metodi di segmentazione possono essere suddivisi in metodi globali e locali. I metodi globali, come per esempio il K-means, dividono l'immagine in N regioni; i metodi locali, invece, dividono l'immagine in oggetto di interesse e sfondo. In ambito medico spesso c'è la necessità di estrarre il contorno di un solo oggetto. Con un metodo globale esso appartiene ad

un certo cluster, però può appartenervi anche un'altra zona lontana da esso che ha lo stesso livello di grigio. Una suddivisione di questo tipo in ambito medico serve quindi a poco. Per superare queste difficoltà, sono stati introdotti i *modelli deformabili* (Terzopoulos e Fleischer, 1988). Queste tecniche, di tipo locale, forniscono una descrizione analitica compatta della forma dell'oggetto di interesse e consentono di individuarne ed estrarne i contorni con poche interazioni da parte dell'utente. Essi sono muniti di meccanismi di interazione intuitivi che permettono di guidare il modello, ove necessario, verso una corretta modellazione. I modelli deformabili permettono inoltre di ottenere una curva senza spigoli che delimita esattamente l'oggetto, anche se l'immagine presenta una risoluzione bassa. Poiché gli oggetti di interesse medico presentano dei contorni morbidi, questi metodi rappresentano quindi un ulteriore vantaggio rispetto a quelli globali.

In questo lavoro si è preso in considerazione un particolare tipo di modelli deformabili, i *contorni attivi parametrici* o *snakes*.

1.4 Snakes

Gli snakes vengono comunemente usati nel campo delle immagini 2D. Uno snake è una curva parametrica che evolve in base alle caratteristiche dell'immagine. Il modello che ha attirato maggiore attenzione fino ad oggi è quello introdotto da Kass (Kass, Witkin e Terzopoulos, 1988). Quest'ultimo introdusse per la prima volta l'algoritmo tradizionale, che punta a minimizzare il livello energetico totale dello snake. In generale, il funzionamento dei contorni attivi è dato da una prima fase in cui si stima la posizione dell'oggetto in esame e si inizializza lo snake. Successivamente si avvia il processo iterativo per far convergere lo snake verso bordi dell'oggetto. Uno dei maggiori vantaggi rispetto ai classici metodi di rilevamento dei bordi è la rappresentazione continua dei contorni, infatti ad ogni iterazione dell'algoritmo si ha una rappresentazione istantanea del probabile bordo permettendo così la visualizzazione delle evoluzioni temporali del processo.

Lo snake è rappresentato da una curva parametrica $\mathbf{C}(s) = [X(s), Y(s)] : U \rightarrow \mathbb{R}^2$, con $U \subset \mathbb{R}$ intervallo chiuso e $s \in [0, 1]$ lunghezza d'arco. La sua equazione di evoluzione può essere ottenuta dalla minimizzazione di un funzionale energetico che lavora sotto l'influenza di forze interne e forze esterne. Le forze interne servono a minimizzare il perimetro e regolarizzare il più possibile i contorni; quelle esterne invece dirigono lo snake verso i bordi dell'immagine. L'equazione di evoluzione può essere quindi ottenuta dalla minimizzazione di

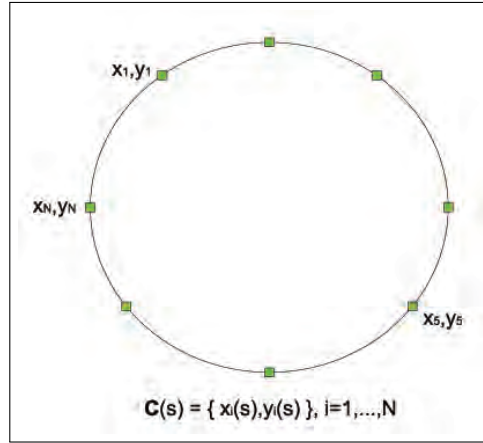


Figura 1.2: Esempio di curva parametrica che rappresenta uno snake.

$$E_{snake}(\mathbf{C}) = E_{interna}(\mathbf{C}) + E_{esterna}(\mathbf{C}), \quad (1.1)$$

dove $E_{interna}$ è un'energia di deformazione, mentre $E_{esterna}$ è un'energia di trasporto.

1.5 Energia interna

L'energia interna rappresenta l'energia intrinseca della curva e può essere scritta come somma pesata di due contributi:

$$E_{interna}(\mathbf{C}) = \int_0^1 \alpha \left\| \frac{\partial \mathbf{C}(s)}{\partial s} \right\|^2 + \beta \left\| \frac{\partial^2 \mathbf{C}(s)}{\partial s^2} \right\|^2 ds$$

Il termine di primo ordine è detto energia elastica e controlla la lunghezza dello snake; il suo contributo è pesato dal parametro α . L'energia elastica è dipendente dalla distanza dei punti che costituiscono la curva parametrica: aumentando la distanza tra due punti aumenta l'energia elastica.

Il termine di secondo ordine è detto energia di curvatura e controlla la rigidità regolarizzando la curva. Esso rappresenta la tendenza dello snake ad opporsi alle modifiche della sua curvatura. Aumentando il relativo parametro di controllo β , si massimizzano gli angoli interni dello snake, eliminando gli spigoli.

L'energia deformabile interna, quindi, è principalmente responsabile della flessibilità e dell'estensione del contorno; essa rappresenta l'energia intrinseca della curva.

1.6 Energia esterna

L'energia esterna deriva dalle caratteristiche dell'oggetto che si vuole descrivere ed è calcolata integrando una funzione lungo il contorno $\mathbf{C}(s)$:

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

L'idea è quella di definire una funzione potenziale $P(x, y)$ calcolata dai dati dell'immagine, che assume valori più piccoli sui contorni. Lo snake evolverà quindi fino a raggiungere un minimo di energia che corrisponderà a uno stato stabile, il quale dovrebbe essere proprio il contorno ricercato.

1.7 Minimizzazione dell'energia

Il problema di trovare una curva $\mathbf{C}(s)$ che minimizzi l'energia dell'equazione (1.1) è conosciuto come problema variazionale. L'equazione dello snake da minimizzare è data da

$$E_{snake}(\mathbf{C}(s)) = \int_0^1 \alpha \left\| \frac{\partial \mathbf{C}(s)}{\partial s} \right\|^2 + \beta \left\| \frac{\partial^2 \mathbf{C}(s)}{\partial s^2} \right\|^2 + k_1 \int_0^1 P(\mathbf{C}(s)) ds,$$

in cui il parametro k_1 pesa il contributo dell'energia esterna. La curva che minimizza E_{snake} deve soddisfare l'equazione di Eulero-Lagrange associata:

$$\frac{\partial}{\partial s} \left(\alpha \frac{\partial \mathbf{C}(s)}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left(\beta \frac{\partial^2 \mathbf{C}(s)}{\partial s^2} \right) - k_1 \nabla P(\mathbf{C}(s)) = 0 \quad (1.2)$$

Lo snake è di solito inizializzato come una linea chiusa che circonda l'oggetto di interesse. Esso poi si muove contraendosi fino a che l'energia esterna non lo fa avvicinare e fermare sui bordi desiderati. Infatti, se il termine di energia esterna fosse nullo, una curva che evolve per il solo effetto della curvatura e della lunghezza si contrarrebbe fino a ridursi in un punto.

A volte in ambito medico è più utile poter inizializzare il contorno dentro l'oggetto e farlo espandere fino ai bordi. Se però la curva non è già dall'inizio abbastanza vicina al bordo, l'energia esterna non è in grado di attirarla e di conseguenza la curva si riduce in un punto. Per superare questo problema, considerando la curva come un palloncino, si aggiunge un termine di espansione lungo la direzione normale alla curva per "gonfiarla" ed evitare che collassi in un punto, fino a raggiungere i contorni dell'immagine dove influisce la funzione di energia esterna (Cohen e Cohen, 1993). Bisogna però

stare attenti perché se il bordo è troppo debole rispetto a questa energia di espansione, la curva passa sopra di esso senza esserne attratta. Inoltre il termine di espansione evita che la curva venga bloccata da falsi bordi isolati e rende il risultato finale meno sensibile all'inizializzazione dell'utente.

Aggiungendo questo termine, la (1.2) diventa

$$\frac{\partial}{\partial s} \left(\alpha \frac{\partial \mathbf{C}(s)}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left(\beta \frac{\partial^2 \mathbf{C}(s)}{\partial s^2} \right) - k_1 \nabla P(\mathbf{C}(s)) - k_2 \mathbf{N} = 0, \quad (1.3)$$

dove N è la normale esterna alla curva. Il termine $k_2 N$ è anche chiamato "pressione" interna.

Discretizzando l'equazione, è possibile calcolarne una soluzione numerica (Kass, Witkin e Terzopoulos, 1988; Quarteroni, Sacco e Saleri, 2008). Lo snake viene campionato in N punti $\mathbf{C} = (\mathbf{C}_1, \dots, \mathbf{C}_N)$ distanti tra loro h .

Approssimando alle differenze finite centrate:

$$\begin{aligned} \frac{\partial^2 \mathbf{C}_i(s)}{\partial s^2} &= \frac{\mathbf{C}_{i+1} - 2\mathbf{C}_i + \mathbf{C}_{i-1}}{h^2} \\ \frac{\partial^4 \mathbf{C}_i(s)}{\partial s^4} &= \frac{\mathbf{C}_{i-2} - 4\mathbf{C}_{i-1} + 6\mathbf{C}_i - 4\mathbf{C}_{i+1} + \mathbf{C}_{i+2}}{h^4} \end{aligned}$$

e ipotizzando α e β costanti, la (1.3) diventa

$$\begin{aligned} \alpha \frac{\mathbf{C}_{i-1} - 2\mathbf{C}_i + \mathbf{C}_{i+1}}{h^2} - \beta \frac{\mathbf{C}_{i-2} - 4\mathbf{C}_{i-1} + 6\mathbf{C}_i - 4\mathbf{C}_{i+1} + \mathbf{C}_{i+2}}{h^4} &= \\ &= k_1 \nabla P(\mathbf{C}_i) + k_2 \mathbf{N}(\mathbf{C}_i). \end{aligned}$$

Introducendo quindi la matrice

$$A = \frac{1}{h^2} \begin{bmatrix} -2\alpha - 6\frac{\beta}{h^2} & \alpha + 4\frac{\beta}{h^2} & -\frac{\beta}{h^2} & 0 & \dots & \dots \\ \alpha + 4\frac{\beta}{h^2} & -2\alpha - 6\frac{\beta}{h^2} & \alpha + 4\frac{\beta}{h^2} & -\frac{\beta}{h^2} & 0 & \dots \\ -\frac{\beta}{h^2} & \alpha + 4\frac{\beta}{h^2} & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix}$$

è possibile riscrivere la (1.3) nella forma:

$$A\mathbf{C} = k_1 \nabla P(\mathbf{C}) + k_2 \mathbf{N},$$

equivalente al sistema di equazioni indipendenti

$$\begin{cases} A\mathbf{X} - k_1 \frac{\partial P(\mathbf{X}, \mathbf{Y})}{\partial x} - k_2 N_x = 0 \\ A\mathbf{Y} - k_1 \frac{\partial P(\mathbf{X}, \mathbf{Y})}{\partial y} - k_2 N_y = 0 \end{cases},$$

dove $\mathbf{N} = (N_x, N_y)$. Per risolvere il sistema, si tratta il contorno come funzione dipendente anche dal tempo t , considerando quindi il sistema come dinamico e cercando la soluzione della (1.3) come stato stazionario dell'equazione

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial s} \left(\alpha \frac{\partial C(s)}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left(\beta \frac{\partial^2 C(s)}{\partial s^2} \right) - k_1 \nabla P(C(s)) - k_2 \mathbf{N}. \quad (1.4)$$

Discretizzando rispetto al tempo, si ottiene il seguente sistema

$$\begin{cases} -\gamma(\mathbf{X}_{t+1} - \mathbf{X}_t) = A\mathbf{X}_{t+1} - k_1 \frac{\partial P(\mathbf{X}_t, \mathbf{Y}_t)}{\partial x} - k_2 N_x(\mathbf{X}_t, \mathbf{Y}_t) \\ -\gamma(\mathbf{Y}_{t+1} - \mathbf{Y}_t) = A\mathbf{Y}_{t+1} - k_1 \frac{\partial P(\mathbf{X}_t, \mathbf{Y}_t)}{\partial y} - k_2 N_y(\mathbf{X}_t, \mathbf{Y}_t) \end{cases},$$

dove γ è il passo temporale. Si trova infine la soluzione

$$\begin{cases} \mathbf{X}_{t+1} = (A + \gamma I)^{-1} \left(\gamma \mathbf{X}_t + k_1 \frac{\partial P(\mathbf{X}_t, \mathbf{Y}_t)}{\partial x} + k_2 N_x(\mathbf{X}_t, \mathbf{Y}_t) \right) \\ \mathbf{Y}_{t+1} = (A + \gamma I)^{-1} \left(\gamma \mathbf{Y}_t + k_1 \frac{\partial P(\mathbf{X}_t, \mathbf{Y}_t)}{\partial y} + k_2 N_y(\mathbf{X}_t, \mathbf{Y}_t) \right) \end{cases}.$$

In questa fase i pesi ed i valori delle forze interne ed esterne hanno un ruolo fondamentale; questi devono essere tali da attrarre il modello verso le caratteristiche salienti dell'immagine, individuando man mano i punti migliori su cui muovere il contorno. Inoltre, i coefficienti nella matrice A devono avere ordini di grandezza simili. Si è dimostrato che si ottengono risultati migliori quando i parametri sono dell'ordine di h^2 per α e h^4 per β (Cohen e Cohen, 1993).

Per dare risultati più stabili, servirebbe un passo temporale γ uniforme che vada bene per tutti i punti del contorno. Infatti, se il passo temporale γ è troppo grande, alcuni punti sulla curva possono muoversi più velocemente di altri, passare oltre il minimo cercato e non tornare più indietro; se invece è

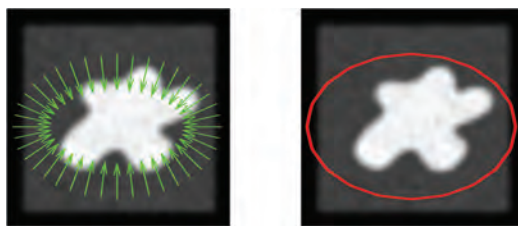
troppo piccolo, solo pochi punti ad alto gradiente attireranno a curva. Quindi, invece di modificare γ , si normalizza la forza esterna sostituendo $\frac{\nabla P}{\|\nabla P\|}$ al posto di ∇P (Cohen e Cohen, 1993). Questo simula un passo temporale locale che fa evolvere la curva alla stessa velocità in tutti i punti. Definendo quindi $F_{esterna} = \frac{\nabla P}{\|\nabla P\|}$, si riscrive la (1.4):

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial s} \left(\alpha \frac{\partial C(s)}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left(\beta \frac{\partial^2 C(s)}{\partial s^2} \right) - k_1 F_{esterna} - k_2 \mathbf{N}.$$

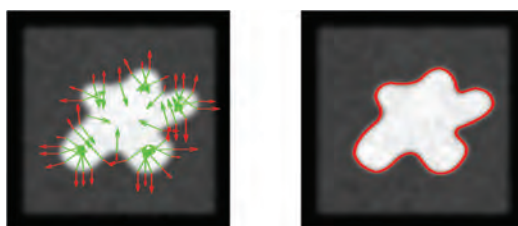
PSEUDOCODICE

- 1 Inizializzazione contorno C^0
- 2 Calcolo della funzione P
- 3 Calcolo di ∇P
- 4 for i=0:N
- 5 Calcolo di $F_{esterna}(C^i)$ e $N(C^i)$:
interpolazione di ∇P su C^i e calcolo delle normali alla curva C^i
- 6 $A_2 = (A + \gamma I)$
- 7 Risoluzione di $A_2 C^{i+1} = \gamma C^i + k_1 F_{esterna}(C^i) + k_2 N(C^i)$
- 8 Interpolazione del contorno C^{i+1} per avere sempre i punti a
distanza h
- 9 end

Si noti che non si è stabilito un criterio di arresto, il numero di iterazioni N va perciò scelto a priori dall'utente. In Figura 1.3 si riporta un esempio relativo al comportamento di uno snake tradizionale sull'immagine test considerata. Le frecce verdi rappresentano la pressione che sgonfia la curva ($k_2 < 0$), mentre quelle rosse rappresentano la forza esterna che blocca la curva sul contorno dell'immagine. In Figura 1.4 si riporta lo stesso esempio, in cui però lo snake è inizializzato all'interno dell'oggetto. Le frecce verdi ora rappresentano la forza che fa gonfiare la curva ($k_2 > 0$) quindi hanno verso opposto rispetto a prima, mentre quelle rosse bloccano la curva sui bordi spingendola verso l'interno.



(a) *Inizializzazione del contorno intorno all'oggetto*

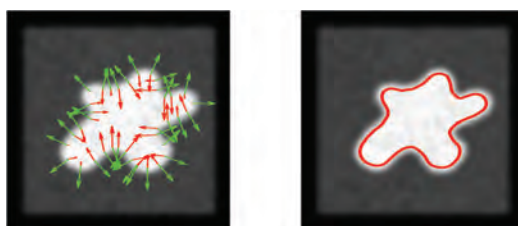


(b) *Risultato finale dopo 200 iterazioni*

Figura 1.3: Comportamento dello snake sull'immagine test, inizializzando la curva all'esterno dell'oggetto. Le frecce verdi rappresentano la pressione che tende a restringere lo snake, mentre il comportamento della forza esterna che si oppone alla pressione è descritto dalle frecce rosse.



(a) *Inizializzazione all'interno dell'oggetto*



(b) *Risultato finale dopo 200 iterazioni*

Figura 1.4: Comportamento dello snake sull'immagine test, inizializzando la curva all'interno dell'oggetto. Le frecce verdi mostrano il comportamento della pressione che fa gonfiare lo snake, mentre le frecce rosse descrivono l'effetto della forza esterna.

1.8 Scelta della forza esterna

Perché il modello sia efficiente, bisogna prestare particolare attenzione alla scelta della funzione potenziale $P(x, y)$ introdotta nel Paragrafo 1.6, che determina la forza esterna ($F_{esterna} = \frac{\nabla P(x, y)}{\|\nabla P(x, y)\|}$).

I contorni di un'immagine sono i punti in cui l'intensità luminosa cambia bruscamente. Un modo per identificarli è dato dalla ricerca della direzione in cui si ha il massimo gradiente locale dell'immagine. Poiché l'obiettivo è minimizzare l'energia, la funzione P deve essere tale da assumere valori minimi, cioè vicini a zero, in prossimità dei contorni.

Di seguito vengono descritte due scelte di $P(x, y)$ attraverso cui si calcolerà la forza esterna: una di tipo deterministico, basata sulle proprietà dell'immagine, e una di tipo probabilistico, in cui si considera la probabilità con cui un pixel appartiene al contorno.

1.8.1 Forza deterministica

Data un'immagine di intensità $I(x, y)$, ci si aspetta che la funzione potenziale P fermi la curva sui bordi degli oggetti. Classicamente si utilizzano funzioni di tipo deterministico, basate sul gradiente dell'immagine. Come visto anche nei metodi Edge based, infatti, esso assume valori massimi proprio sui bordi. Tradizionalmente si utilizzano vari tipi di funzioni, tra cui $P(x, y) = -|\nabla I(x, y)|$ oppure $P(x, y) = -|\nabla I(x, y)|^2$. In molti casi, quando nell'immagine è presente troppo rumore, prima di calcolare il gradiente di $I(x, y)$ si modifica l'immagine mediante un filtro gaussiano che rende i contorni più sfocati ma allo stesso tempo aumenta la capacità dello snake di catturare i contorni: la funzione utilizzata è in questo caso $P(x, y) = -|\nabla(G_\sigma(x, y) * I(x, y))|$ oppure $P(x, y) = -|\nabla(G_\sigma(x, y) * I(x, y))|^2$, dove G_σ è il kernel di un filtro gaussiano di deviazione standard σ (Kass, Witkin e Terzopoulos, 1988; Tsechpenakis, 2011). La formulazione che si utilizza in questo lavoro è (Caselles, Kimmel e Sapiro, 1995)

$$P(x, y) = \frac{1}{1 + |\nabla(G_\sigma(x, y) * I(x, y))|^2}. \quad (1.5)$$

Il gradiente infatti, come visto nei metodi Edge detection, assume valori massimi sui contorni dell'immagine; prendendo $P(x, y)$ in questo modo si ottengono valori minimi sui bordi dell'immagine e un valore di $P = 1$ dove l'immagine è uniforme ($\nabla I(x, y) = 0$).

La forza costruita mediante $P(x, y)$ è quindi di tipo deterministico, cioè si basa esclusivamente sui dati forniti dall'immagine. In Figura 1.5 viene

presentato un esempio di applicazione su un'immagine test, formata da due classi, in cui si è utilizzata la (1.5). La segmentazione con questo tipo di forza esterna funziona bene se i contorni sono molto definiti e non eccessivamente frastagliati, altrimenti il metodo ha difficoltà ad identificare con precisione i bordi. Nel caso della segmentazione della materia bianca del cervello, per esempio, il metodo fallisce (Figura 1.6).

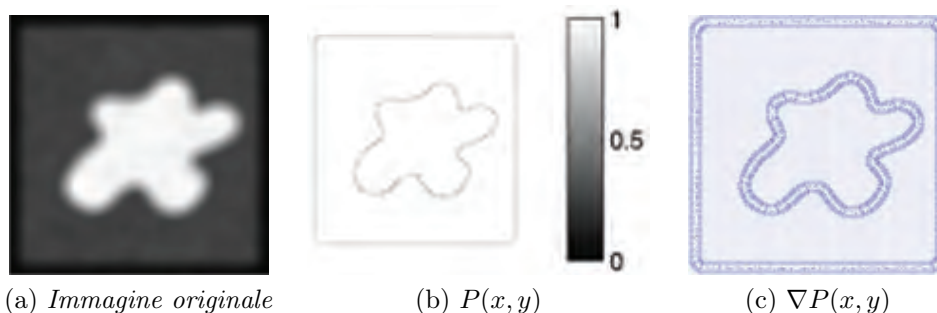


Figura 1.5: Applicazione della forza deterministica all'immagine test, utilizzando $P(x, y) = \frac{1}{1+|\nabla(G_\sigma(x,y)*I(x,y))|^2}$.

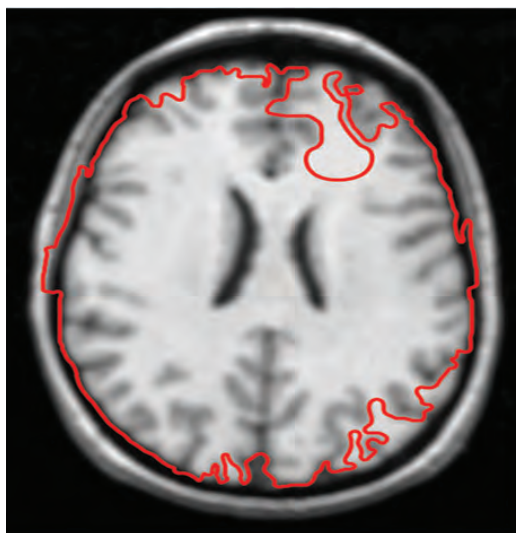


Figura 1.6: Segmentazione errata della materia bianca mediante la forza deterministica.

1.8.2 Forza probabilistica

Un'alternativa di tipo probabilistico è quella basata sulle probabilità, come proposto nell'articolo di Yang et al. (2007).

Nel caso semplice di un'immagine bimodale, dove c_{in} rappresenta l'oggetto da segmentare e c_{out} rappresenta lo sfondo, si costruisce un potenziale $P(x, y)$ del tipo (Yang et al., 2007):

$$P(x, y) = (Pr^*((x, y) \in c_{in}) - Pr^*((x, y) \in c_{out}))^2,$$

dove Pr^* è la probabilità che il pixel (x, y) appartenga rispettivamente a c_{in} o c_{out} . Per il calcolo di Pr^* si vedranno in seguito i due metodi che sono stati utilizzati in questa tesi.

La nuova funzione $P(x, y)$, da cui successivamente si calcola la forza esterna, agisce come quella deterministica: per i pixel appartenenti allo sfondo, cioè gli $(x, y) \in c_{out}$, si ha $Pr^*((x, y) \in c_{out}) = 1$ e $Pr^*((x, y) \in c_{in}) = 0$ poiché essi appartengono con certezza ad una determinata classe; in modo analogo, per i pixel appartenenti all'oggetto, cioè gli $(x, y) \in c_{in}$, si ha $Pr^*((x, y) \in c_{out}) = 0$ e $Pr^*((x, y) \in c_{in}) = 1$; il potenziale $P(x, y)$ assume in entrambi i casi valore 1. Per quanto riguarda i pixel che costituiscono i bordi, la probabilità condizionata di appartenere a c_{in} o c_{out} non è più 1 o 0: il pixel ha circa la stessa probabilità di appartenere allo sfondo o all'oggetto da segmentare, di conseguenza $P(x, y)$ assume valori molto vicini a zero. Nel

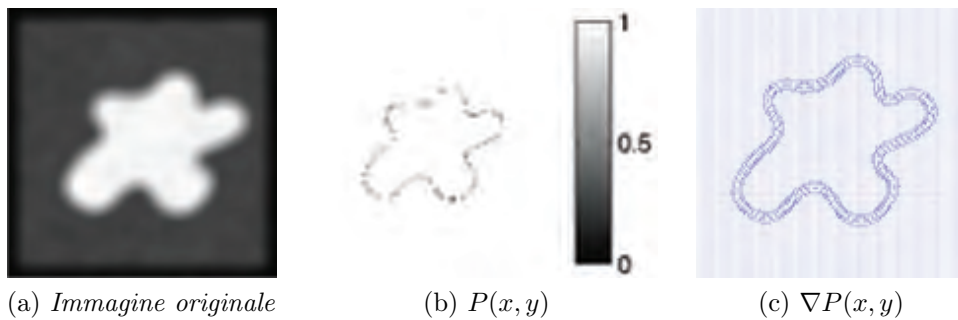


Figura 1.7: Applicazione della forza probabilistica all'immagine test.

caso più generale di un'immagine con n classi, si calcola per ogni pixel la differenza al quadrato delle due probabilità condizionate più grandi.

Come mostra la Figura 1.7, dove $P(x, y)$ è stata calcolata mediante il modello parametrico che sarà descritto più avanti, se i contorni dell'oggetto sono ben definiti l'utilizzo della forza probabilistica porta a un risultato simile a quello della forza deterministica (Figura 1.5).

In Figura 1.8 si osserva che i contorni del cervello sono identificati in modo più accurato mediante la forza probabilistica. In particolare si nota come il ventricolo destro sia segmentato in modo più preciso, tenendo conto non solo dei ventricoli ma anche della zona di materia grigia accanto ad essi, che altrimenti non sarebbe identificata.

Questo approccio chiaramente dipende da come si calcola la probabilità Pr^* .

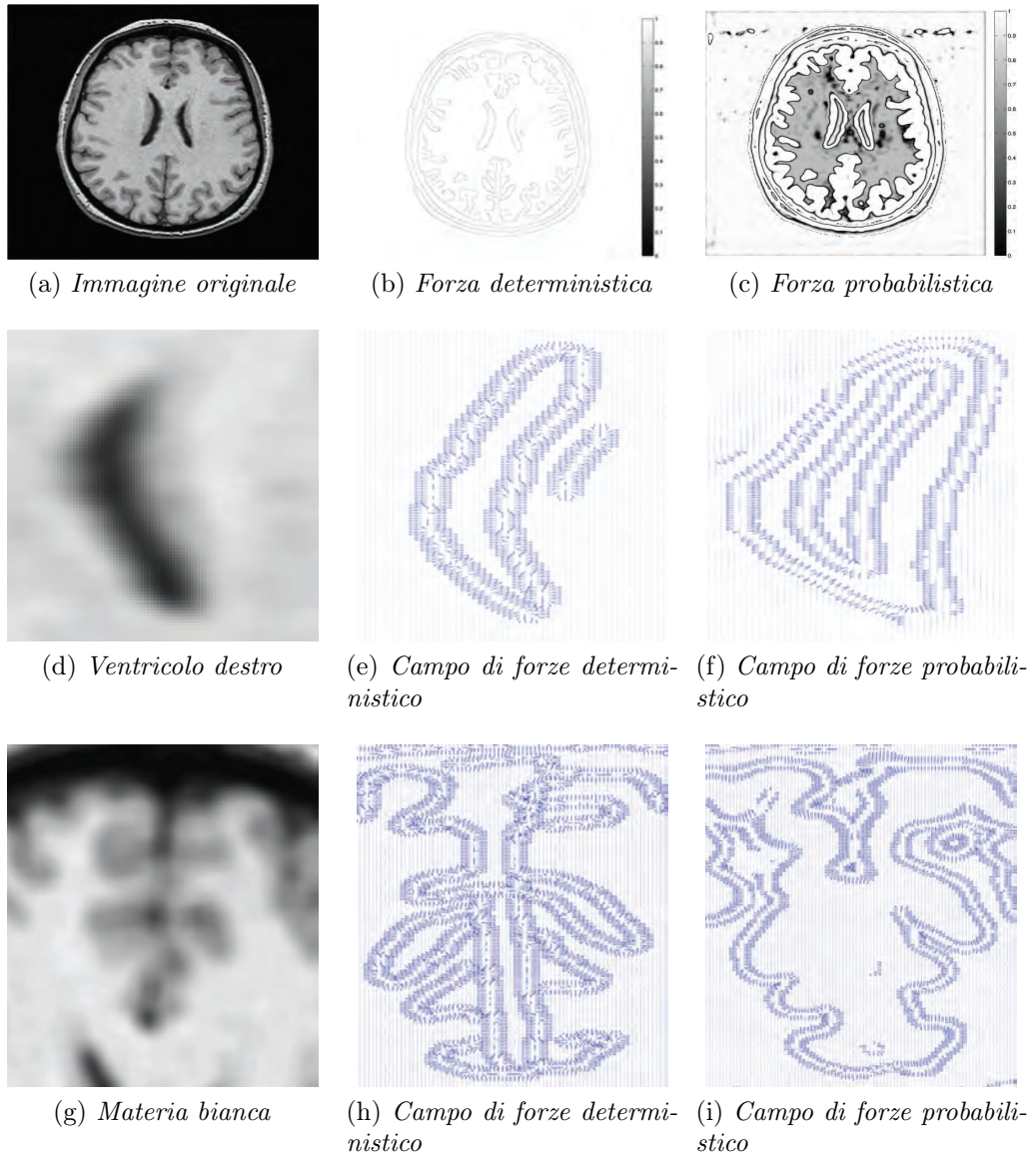


Figura 1.8: Confronto tra forza deterministica e forza probabilistica sull'immagine di un cervello, con particolare attenzione al ventricolo destro e alla materia grigia.

1.8.3 Forza combinata

Si propone in questa tesi una nuova forza esterna, costruita combinando insieme la forza probabilistica e la forza deterministica viste precedentemente, in modo da rendere più robusta la segmentazione. Per costruire questa forza si calcolano le due funzioni $P_{deterministica}(x, y)$ e $P_{probabilistica}(x, y)$ introdotte precedentemente, ottenendo:

$$F_{deterministica} = \frac{\nabla P_{deterministica}}{\|\nabla P_{deterministica}\|}$$

e

$$F_{probabilistica} = \frac{\nabla P_{probabilistica}}{\|\nabla P_{probabilistica}\|}$$

Poiché potrebbe essere utile pesare in modo diverso il contributo deterministico e quello probabilistico, si introducono dei coefficienti che moltiplicano le due forze, i quali dovranno essere scelti opportunamente in base all'immagine considerata e all'oggetto da segmentare:

$$\nabla P_{combinata} = w_1 F_{probabilistica} + w_2 F_{deterministica}, \quad (1.6)$$

da cui, normalizzando, si ottiene la forza combinata:

$$F_{combinata} = \frac{\nabla P_{combinata}}{\|\nabla P_{combinata}\|}.$$

In pratica questo tipo di forza è una generalizzazione di quelle viste prima: ponendo w_1 o w_2 pari a zero, si riottiene rispettivamente la sola $F_{probabilistica}$ o $F_{deterministica}$.

Come verrà mostrato nel Capitolo 3, nei casi in cui la segmentazione effettuata porta già a risultati soddisfacenti usando le forze singolarmente, questo approccio non genera grosse differenze ma per alcune immagini velocizza il processo di identificazione dei contorni; in altri casi, dove i contorni sono più difficili da identificare, questo metodo può portare a dei miglioramenti nei risultati.

1.9 Filtraggio delle immagini

Quando l'immagine da analizzare presenta troppo rumore, cioè variazioni casuali di intensità, la si processa mediante opportuni metodi di filtraggio. Il filtraggio può essere visto come un sistema lineare

$$f(x, y) \longrightarrow \boxed{g(x, y)} \longrightarrow h(x, y)$$

dove $f(x, y)$ e $h(x, y)$ rappresentano rispettivamente l'immagine di input e l'immagine di output. $h(x, y)$ è data dalla convoluzione di $f(x, y)$ con una *maschera di convoluzione* $g(x, y)$:

$$h(x, y) = f(x, y) * g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x', y') g(x - x', y - y') dx' dy' .$$

Per le funzioni discrete l'equazione diventa:

$$h[i, j] = f[i, j] * g[i, j] = \sum_{k=1}^n \sum_{l=1}^m f[k, l] g[i - k, j - l] .$$

Esistono molte tipologie di filtri, tra cui il più adatto per le immagini mediche è quello gaussiano (Jain, Kasturi e Schunck, 1995). La funzione gaussiana bidimensionale discreta a media zero è usata come filtro:

$$g[i, j] = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{i^2+j^2}{2\sigma^2}}$$

Un grafico di questa funzione è mostrato in Figura 1.9.

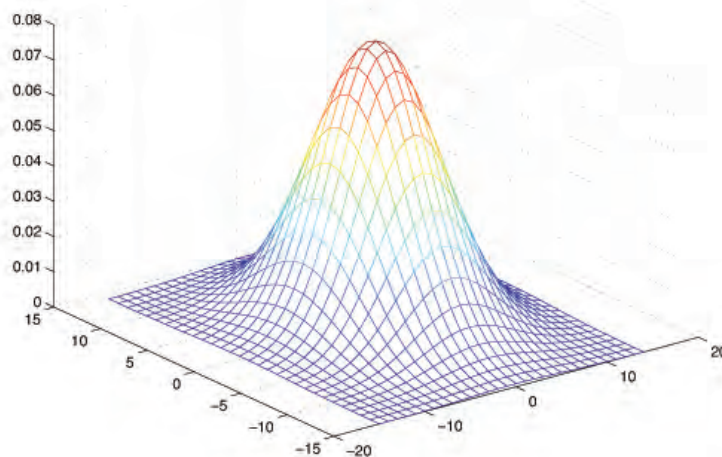


Figura 1.9: Funzione gaussiana bidimensionale a media nulla e varianza 4.

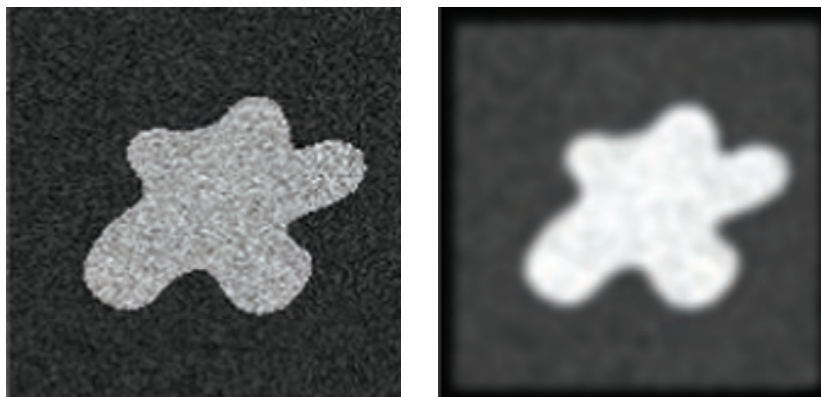


Figura 1.10: Esempio di convoluzione dell'immagine con un filtro gaussiano a media zero e varianza $\sigma^2 = 4$.

Le funzioni gaussiane hanno alcune proprietà che le rendono particolarmente utili:

1. In due dimensioni, sono simmetriche in modo rotazionale; questo significa che lo spianamento fatto dal filtro è lo stesso in tutte le direzioni. I contorni in un'immagine non sono orientati in una direzione particolare, quindi non c'è ragione di filtrare di più in una direzione rispetto ad un'altra.
2. Un filtro gaussiano sostituisce ogni pixel dell'immagine con una media pesata dei pixel adiacenti in modo che il peso dato ai vicini decresca monotonamente con la distanza dal pixel centrale. Questa proprietà è importante perché un bordo è una caratteristica locale dell'immagine e un'operazione di filtraggio che dia più importanza a pixel più lontani ne distorcerebbe le caratteristiche.
3. Le immagini sono spesso corrotte da segnali ad alte frequenze indesiderati. I bordi delle immagini hanno però componenti sia ad alte che a basse frequenze. Il filtro gaussiano permette che l'immagine non sia corrotta dai contributi dei segnali ad alte frequenze, mentre la maggior parte degli altri segnali vengono mantenuti; in pratica è un filtro passa-basso.

In Figura 1.10 si mostra il filtraggio di un'immagine in cui è presente molto rumore. Si nota come i contorni siano più sfuocati, per questo bisogna sempre prestare attenzione a non filtrare eccessivamente le immagini, soprattutto nel caso di immagini mediche.

Capitolo 2

Modelli mistura

Per ottenere una corretta segmentazione, è necessario innanzitutto scegliere un metodo di classificazione adatto alla tipologia di immagini prese in considerazione, che associ ogni pixel alla classe opportuna. Per costruire la forza esterna probabilistica descritta nel Sottoparagrafo 1.8.2 a pagina 24, sono stati presi in considerazione due modelli statistici per la classificazione, uno parametrico e l'altro bayesiano non parametrico. Entrambi i modelli sono di tipo globale, cioè forniscono una clusterizzazione dell'immagine in N classi. Per poter estrarre quindi l'oggetto di interesse si è deciso di inserirli all'interno di un modello di tipo snake.

2.1 Modello parametrico

Come è stato detto nel Paragrafo 1.1, l'immagine è una matrice X in cui ogni elemento è un pixel, il cui valore rappresenta l'intensità dell'immagine. Si assume che i dati siano descritti da una mistura di distribuzioni: ogni classe ha generato i dati in accordo con una specifica distribuzione, ma al termine della generazione essi appaiono come prodotti da un'unica distribuzione.

Il modello parametrico preso in considerazione è il *Gaussian Mixture Model* (GMM) (Farnoosh e Zarpak, 2008), cioè una distribuzione di probabilità parametrica costituita da una somma pesata di s densità gaussiane:

$$p(x) = \sum_{i=1}^s w_i N(x|\mu_i, \sigma_i^2),$$

dove w_i sono i pesi delle componenti e devono soddisfare il vincolo $\sum_{i=1}^s w_i = 1$, e $N(x|\mu_i, \sigma_i^2)$ sono le componenti gaussiane. Ogni componente $N(x|\mu_i, \sigma_i^2)$

ha densità

$$f(x) = \frac{1}{\sigma_i \sqrt{2\pi}} \exp \left\{ -\frac{(x - \mu_i)^2}{2\sigma_i^2} \right\}$$

dove μ_i, σ_i^2 sono la media e la varianza di ogni classe i . E' necessario stimare l'insieme dei parametri $\lambda = \{w_i, \mu_i, \sigma_i^2\}$ in modo da ricostruire al meglio la distribuzione dei livelli di grigio nell'immagine. Esistono molte tecniche per effettuare questa stima, tra cui la più diffusa è sicuramente il metodo della *Massima Verosimiglianza* (ML) (Reynolds, 2008).

2.1.1 Metodo della Massima Verosimiglianza

L'obiettivo di questo approccio è trovare i parametri che massimizzano la verosimiglianza del GMM avendo come input l'immagine $X = \{x_1, \dots, x_n\}$. Si assume che gli x_i siano indipendenti e identicamente distribuiti con distribuzione $p(\cdot)$ controllata dall'insieme di parametri λ . Allora la densità congiunta di X , sotto queste assunzioni, è:

$$p(X|\lambda) = \prod_{i=1}^n p(x_i|\lambda) = L(\lambda|X).$$

La funzione $L(\lambda|X)$ è detta *verosimiglianza* dei parametri noti i dati. L'obiettivo del metodo è quindi quello di trovare λ^* che massimizza la verosimiglianza:

$$\lambda^* = \arg \max_{\lambda} L(\lambda|X).$$

La stima ML di λ^* può essere ottenuta iterativamente usando l'algoritmo *Expectation-maximization* (EM). L'idea base è, partendo da un modello iniziale λ , cercare un nuovo $\bar{\lambda}$ tale che $p(X|\bar{\lambda}) \geq p(X|\lambda)$. Il nuovo modello diventa poi quello iniziale per la prossima iterazione e si ripete il processo finché non si raggiunge la convergenza. Ad ogni iterazione si effettuano due passi, l'*Expectation-step* (E-step) e il *Maximization-step* (M-step).

Nell'E-step si calcola il valore atteso della verosimiglianza, dati X e una stima di parametri λ^i all'iterazione corrente:

$$Q(\lambda|\lambda^i) = E(p(X|\lambda)|X, \lambda^i).$$

Nel M-step si determina il valore dei parametri che massimizzano il valore

atteso calcolato all'E-step:

$$\lambda^{i+1} = \arg \max_{\lambda} Q(\lambda, \lambda^i) \quad (2.1)$$

Ad ogni iterazione vengono usate le seguenti formule di stima dei parametri, che garantiscono una crescita monotona nel valore della verosimiglianza:

$$\begin{aligned} \bar{w}_l &= \frac{1}{n} \sum_{i=1}^n p(l|x_i, \lambda) \\ \bar{\mu}_l &= \frac{\sum_{i=1}^n x_i p(l|x_i, \lambda)}{\sum_{i=1}^n p(l|x_i, \lambda)} \\ \bar{\sigma}_l^2 &= \frac{\sum_{i=1}^n x_i^2 p(l|x_i, \lambda)}{\sum_{i=1}^n p(l|x_i, \lambda)} - \bar{\mu}_l^2 \end{aligned}$$

La probabilità condizionata per la componente l è data da

$$p(l|x_i, \lambda) = \frac{w_l p(x_i|\mu_l, \sigma_l^2)}{\sum_{k=1}^s w_k p(x_i|\mu_k, \sigma_k^2)},$$

calcolata attraverso il Teorema di Bayes. In pratica l'implementazione dell'algoritmo EM può essere schematizzata come:

- 1 Inizializzazione $s, \lambda_i = \{w_i, \mu_i, \sigma_i^2\}, i = 1, \dots, s$
- 2 **do**
- 3 Expectation: calcolo di $p(l|x_i, \lambda)$ per $l = 1, \dots, s$ e $i = 1, \dots, n$
- 4 Maximization: calcolo di $\bar{w}_l, \bar{\mu}_l, \bar{\sigma}_l^2$ per $l = 1, \dots, s$
- 5 **while** (i parametri w_i o $\{\mu_i, \sigma_i^2\}$ sono variati sostanzialmente)

La convergenza è sempre dimostrata, anche se per alcune soluzioni iniziali non si arriva alla soluzione ottima ma ad un minimo locale.

2.1.2 Applicazione del modello ai dati

Per applicare il GMM ai dati si utilizza la funzione Matlab *gmdistribution.fit*, che richiede come input solo l'immagine X e il numero s di cluster scelti a priori. Il vantaggio di questa funzione rispetto alla funzione simile *gmdistribution* è che non è necessario fornire dei valori iniziali ai parametri w, μ e σ^2 , ma semplicemente decidere il numero di classi s in cui clusterizzare l'immagine. Per scegliere un valore s adatto, si osserva l'istogramma

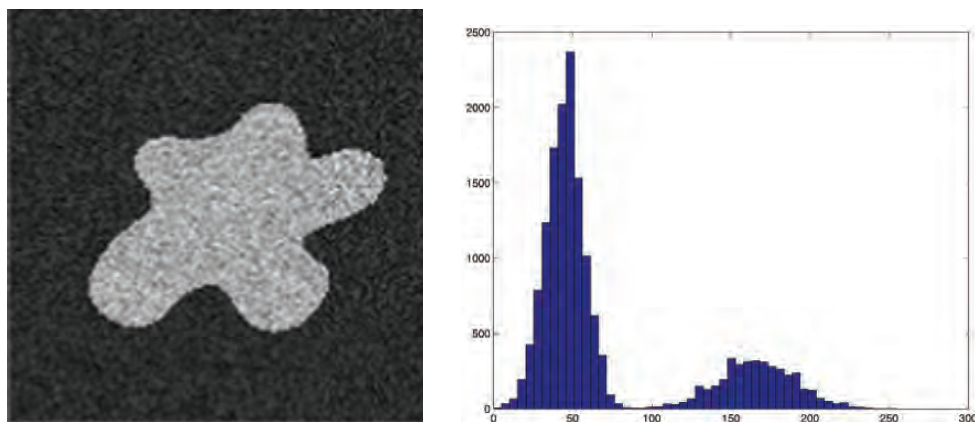


Figura 2.1: Esempio di immagine in scala di grigi con relativo istogramma

dell'immagine, che rappresenta la distribuzione dei livelli di grigio dell'immagine tracciando il numero di pixel per ogni livello. Esso è pertanto una stima delle probabilità di occorrenza dei livelli di grigio dell'immagine, quindi osservandolo è possibile stimare il numero di gaussiane che costituiscono la distribuzione totale dell'immagine. In Figura 2.1 si riporta un esempio in cui dall'istogramma dell'immagine si vede che l'immagine test è composta da due classi. Spesso, come anche in questo caso, non è necessario stimare s dall'istogramma, ma basta osservare direttamente l'immagine per capire in quanti cluster suddividerla. Dopo aver scelto il numero di cluster s , si costruisce quindi il GMM da cui, mediante la funzione Matlab *posterior*, si ricavano per ogni pixel le probabilità condizionate Pr^* di appartenere alle varie classi, da utilizzare per calcolare la $P(x, y)$ descritta nel Sottoparagrafo 1.8.2.

A volte nelle immagini è presente molto rumore che causa grosse difficoltà nella segmentazione; spesso, inoltre il poco contrasto nelle immagini non permette di identificare bene i contorni dell'oggetto di interesse. Il primo problema viene risolto filtrando l'immagine, come si è visto nel Sottoparagrafo 1.9. Per superare la seconda difficoltà si equalizza l'istogramma dell'immagine, cioè si migliora la distribuzione delle intensità in modo da aumentare il contrasto dell'immagine, come mostrato in Figura 2.2.

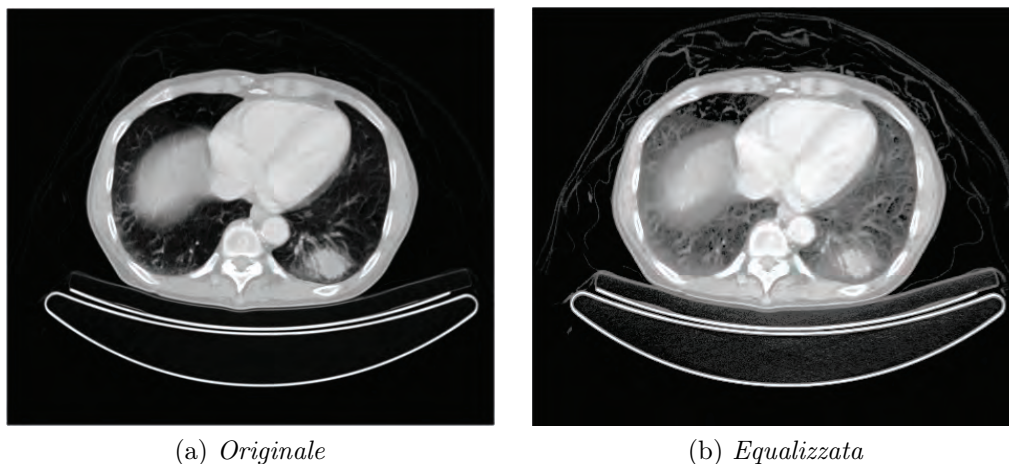
(a) *Originale*(b) *Equalizzata*

Figura 2.2: Equalizzazione di un'immagine medica.

2.2 Modello bayesiano non parametrico

Come si è visto, il modello parametrico richiede di specificare a priori il numero di classi. Esistono invece dei modelli più complessi che includono una selezione automatica di questo numero, stimato dai dati in input. In letteratura sono presenti molti approcci non parametrici utilizzati per il riconoscimento dei contorni (Abd-Almageed, Smith e Ramadan, 2003; Kim et al., 2002; Mory, Ardon e Thiran, 2007; Ozertem e Erdogmus, 2007), anche applicati a immagini mediche (Cramer et al., 2008).

In questo lavoro si applica un modello già proposto in letteratura che si è rivelato molto efficiente per la clusterizzazione delle immagini (Orbanz e Buhmann, 2008; Nguyen e Wu, 2011); la novità che si introduce sta nell'applicare tale metodo al passo successivo della segmentazione, cioè utilizzarlo per costruire la forza esterna probabilistica descritta nel Sottoparagrafo 1.8.2. Il metodo considerato è un modello bayesiano non parametrico basato sui *Dirichlet Process Mixture models* (DPM), dove il numero di cluster è sostituito da una variabile casuale con un parametro di controllo. In generale, una possibile debolezza dei metodi di classificazione è data dal fatto che trascurano lo smoothness spaziale, cioè non tengono conto del fatto che gli oggetti presenti nell'immagine sono costituiti da superfici continue. Le immagini molto rumorose possono avere contorni molto frastagliati e cluster incoerenti. Questo inconveniente può essere superato introducendo un accoppiamento spaziale tra caratteristiche dell'immagine adiacenti.

Il modello proposto (Orbanz e Buhmann, 2008) combina il DPM con i campi aleatori markoviani (MRF), che modellano i vincoli di smoothness

spaziale. Il vantaggio di questa combinazione è che le immagini non hanno bisogno di essere processate mediante dei metodi di filtraggio come nel caso del GMM, proprio grazie all'introduzione del "sottostante" campo di Markov. Prima di procedere con la descrizione del modello proposto e vedere la sua applicazione alle immagini, si fornisce una breve introduzione al processo di Dirichlet, ai modelli DPM e ai campi aleatori markoviani. Tutti i contenuti descritti in seguito sono tratti dall'articolo di Orbanz e Buhmann (2008).

2.2.1 Processo di Dirichlet

Si considera inizialmente il modello parametrico, dove la distribuzione a posteriori è della forma

$$p(\theta|\mathbf{x}) \propto F(\mathbf{x}|\theta)G(\theta),$$

grazie al teorema di Bayes. La generazione dei dati secondo questa probabilità può essere vista in due passi: si sceglie una distribuzione F generando il parametro θ dalla distribuzione a priori e successivamente si campionano i valori \mathbf{x} da F . Il modello parametrico restringe F a uno spazio finito-dimensionale. L'approccio non parametrico generalizza questo modello costruendo F come un oggetto con infiniti gradi di libertà, riuscendo così a raggiungere un alto livello di flessibilità e adattamento ai dati.

Il processo di Dirichlet è parametrizzato da uno scalare $m \in \mathbb{R}_+$ e una misura di probabilità G_0 , chiamata misura base del processo, ed è descritto come $DP(mG_0)$. Poiché, se $F \sim DP(mG_0)$, $F(\omega)$ è una distribuzione di probabilità, la distribuzione del processo di Dirichlet è una distribuzione sulle distribuzioni.

In generale, il modello parametrico più semplice che possiamo adottare in ambito bayesiano è assumere che i dati X_1, \dots, X_n siano un campione da $F(\cdot|\theta)$, dove $F(\cdot|\theta)$ è una distribuzione che dipende da un parametro θ , con densità $f(\cdot|\theta)$, e θ è esso stesso aleatorio con una certa densità π , detta prior. Le inferenze su θ sono basate sulla densità a posteriori di θ , cioè

$$\pi(\theta|x_1, \dots, x_n) = \frac{\prod_{i=1}^n f(x_i|\theta)\pi(\theta)}{\int_{\Omega} \prod_{i=1}^n f(x_i|\theta)\pi(\theta)}.$$

In questo caso la densità a posteriori di θ , cioè la densità condizionale di θ dato il campione osservato (x_1, \dots, x_n) , è stata calcolata con il Teorema di Bayes.

Secondo l'approccio bayesiano non parametrico, invece,

$$\begin{aligned} X_1, \dots, X_n | F &\stackrel{iid}{\sim} F \\ F &\sim DP(mG_0), \end{aligned} \tag{2.2}$$

cioè è la stessa distribuzione dell' i -esimo dato che è una realizzazione da un processo di Dirichlet.

Il processo di Dirichlet deve la sua applicabilità alle seguenti proprietà:

1. *Esistenza*: Il processo $DP(mG_0)$ esiste per uno spazio misurabile arbitrario (Ω, A) e una misura di probabilità additiva G_0 su Ω .
2. *Stima a posteriori*: Si ipotizza che le osservazioni x_1, \dots, x_n siano generate da una distribuzione F secondo la (2.2). Sia \hat{F}_n la distribuzione empirica dei dati osservati,

$$\hat{F}_n(x) := \frac{1}{n} \sum_{i=1}^n \delta_{x_i}(x)$$

dove $\delta_{x_i}(x)$ è la misura di Dirac centrata in x_i . Allora la probabilità a posteriori della distribuzione F condizionata ai dati è ancora un processo di Dirichlet:

$$F | x_1, \dots, x_n \sim (mG_0 + n\hat{F}_n).$$

Si dice in questo caso che la distribuzione a priori è coniugata alla distribuzione a posteriori.

3. *Campionamento*: Se una sequenza x_1, \dots, x_n sia costruita da $F \sim DP(mG_0)$, il primo dato è generato secondo

$$x_1 \sim G_0$$

Tutti i dati successivi sono generati secondo

$$x_{n+1} | x_1, \dots, x_n \sim \frac{n}{m+n} \hat{F}_n(x_{n+1}) + \frac{m}{m+n} G_0(x_{n+1}). \tag{2.3}$$

2.2.2 Dirichlet Process Mixture models

Il processo di Dirichlet è l'ingrediente fondamentale nei modelli DPM. Una distribuzione costruita dal DP è discreta con probabilità 1 anche se la misura base G_0 è continua. Il DPM evita questa restrizione costruendo una prior $G \sim DP(mG_0)$ e combinandola con la verosimiglianza parametrica $F(\cdot|\theta)$:

$$\begin{aligned} x_i &\stackrel{iid}{\sim} F(\cdot|\theta_i) & i = 1, \dots, n \\ \theta_i &\stackrel{iid}{\sim} G & i = 1, \dots, n \\ G &\sim DP(mG_0) \end{aligned}$$

Per un set di valori $\theta_1, \dots, \theta_n$ generati da G , si ottiene la probabilità condizionata a priori:

$$p(\theta_{n+1}|\theta_1, \dots, \theta_n) = \frac{1}{n+m} \sum_{i=1}^n \delta_{\theta_i}(\theta_{n+1}) + \frac{m}{n+m} G_0(\theta_{n+1}) \quad (2.4)$$

Questa equazione è analoga alla formula (2.3).

I θ_i vengono raggruppati in $N_c \leq n$ gruppi di valori identici. Ogni gruppo è rappresentato da un parametro θ_k^* , $k \in \{1, \dots, N_c\}$ (in pratica $\theta_i = \theta_k^*$ per tutti i parametri θ_i nella classe k).

Poiché la prima sommatoria può essere scritta come:

$$\sum_{i=1}^n \delta_{\theta_i}(\theta_{n+1}) = \sum_{k=1}^{N_c} n_k \delta_{\theta_k^*}(\theta_{n+1}),$$

dove n_k denota il numero di valori appartenenti al gruppo k , la (2.4) può essere riscritta come:

$$p(\theta_{n+1}|\theta_1, \dots, \theta_n) = \sum_{k=1}^{N_c} \frac{n_k}{n+m} \delta_{\theta_k^*}(\theta_{n+1}) + \frac{m}{n+m} G_0(\theta_{n+1}).$$

La combinazione con la verosimiglianza parametrica F dà come risultato la funzione $F(\cdot|\theta_k^*)$ per ogni classe k . Quindi il modello può essere visto come un modello mistura costituito da N_c componenti parametriche $F(\cdot|\theta_k^*)$ e una componente “zero”, responsabile della creazione di nuove classi.

2.2.3 Campi aleatori di Markov

Per modellare i vincoli spaziali, si combina il modello DPM con i campi aleatori di Markov (MRF), che fanno in modo che i punti adiacenti siano assegnati alla stessa classe.

Un MRF è una collezione di variabili casuali definite su un grafo pesato non direzionato $\mathcal{N} = (V_{\mathcal{N}}, E_{\mathcal{N}}, W_{\mathcal{N}})$, dove i $V_{\mathcal{N}}$ rappresentano le posizioni, $E_{\mathcal{N}}$ gli archi e $W_{\mathcal{N}}$ i pesi sugli archi. Poiché il grafo non è direzionato, i pesi sono simmetrici ($w_{ij} = w_{ji}$). Ogni posizione v_i è associata ad un'osservazione x_i e una variabile casuale θ_i . Una distribuzione Π è chiamato campo aleatorio markoviano rispetto a \mathcal{N} se soddisfa la *proprietà di Markov*:

$$\Pi(\theta_i | \theta_{-i}) = \Pi(\theta_i | \theta_{\partial(i)}) \quad \forall v_i \in V_{\mathcal{N}} \quad (2.5)$$

dove $\theta_{-i} := \{\theta_1, \dots, \theta_{i-1}, \theta_{i+1}, \dots, \theta_n\}$ è il set di parametri senza θ_i , mentre $\partial(i) := \{j | (i, j) \in E_{\mathcal{N}}\}$ rappresenta l'insieme di tutti i vicini di v_i in \mathcal{N} . Questa proprietà indica che le variabili θ_i sono dipendenti, ma le dipendenze sono locali, cioè solo tra variabili adiacenti. La distribuzione $\Pi(\theta_1, \dots, \theta_n)$ ha il ruolo di distribuzione a priori. Il parametro θ_i descrive la generazione dell'osservazione x_i mediante la verosimiglianza F .

La definizione di una distribuzione MRF per modellare un certo problema richiede la verifica della proprietà di Markov (2.5). Questa verifica però è di solito molto difficile da effettuare, per cui si ricorre a una proprietà equivalente, più facile da verificare: il teorema di Hammersely-Clifford.

Teorema di Hammersely-Clifford

Una funzione costo H definisce un campo aleatorio di Markov se e solo se

$$H(\theta_1, \dots, \theta_n) = \sum_{C \subseteq \mathcal{C}} H_C(\theta_C),$$

dove \mathcal{C} denota tutti i sottoinsiemi completamente connessi di $V_{\mathcal{N}}$. In pratica la funzione H è un MRF se è data dalla somma di tutte le funzioni di costo locali sui sottoinsiemi di nodi completamente connessi.

La funzione costo H definisce una distribuzione per mezzo di

$$\Pi(\theta_1, \dots, \theta_n) := \frac{1}{Z_H} \exp(-H(\theta_1, \dots, \theta_n)),$$

con un termine di normalizzazione Z_H .

Nel caso di problemi di classificazione i vincoli sono modellati sullo spazio discreto di indicatori $\{S_1, \dots, S_n\}$. Per questo tipo di problemi si utilizza una funzione costo del tipo:

$$H(\theta_i|\theta_{-i}) = -\lambda \sum_{l \in \partial(i)} w_{il} \delta_{S_i, S_l}, \quad (2.6)$$

dove δ è il delta di Kronecker e λ è una costante positiva. Gli indicatori S_i, S_l specificano le classi definite dai parametri θ_i e θ_l . Quindi, se θ_i definisce una classe differente dalle classi di tutti i nodi vicini, $\exp(-H) = 1$. In generale si considerano funzioni costo che soddisfano:

$$\begin{aligned} H(\theta_i|\theta_{-i}) &= 0 \quad \text{se } S_i \notin S_{\partial(i)}, \\ H(\theta_i|\theta_{-i}) &< 0 \quad \text{se } S_i \in S_{\partial(i)}. \end{aligned}$$

Quanti più nodi vicini sono assegnati alla stessa classe di θ_i , tanto più la funzione costo assume valore negativo.

2.2.4 DPM vincolato da MRF

La distribuzione MRF Π introdotta precedentemente può essere decomposta in due termini:

$$\Pi(\theta_1, \dots, \theta_n) \propto P(\theta_1, \dots, \theta_n) M(\theta_1, \dots, \theta_n), \quad (2.7)$$

con

$$\begin{aligned} P(\theta_1, \dots, \theta_n) &:= \frac{1}{Z_P} \exp \left(- \sum_i H_i(\theta_i) \right), \\ M(\theta_1, \dots, \theta_n) &:= \frac{1}{Z_M} \exp \left(- \sum_{C \in C_2} H_C(\theta_C) \right). \end{aligned}$$

P è una distribuzione parametrica a priori di θ , usata per modellare le classi composte da un solo elemento, mentre M è un termine di interazione per le classi di dimensione maggiore. Z_P e Z_M sono le rispettive funzioni di ripartizione, mentre H è la funzione costo data dall'equazione (2.6).

La rappresentazione del modello risultante è:

$$(\theta_1, \dots, \theta_n) \sim M(\theta_1, \dots, \theta_n) \prod_{i=1}^n G(\theta_i),$$

$$G \sim DP(mG_0).$$

La componente P in (2.7) è sostituita da $G \sim DP(mG_0)$. Per giustificare questa sostituzione, si assume una G data e si definisce una funzione costo per posizioni individuali in termini di G :

$$H_i(\theta_i) := -\log(G(\theta_i))$$

$$Z_G := \int \prod_{i=1}^n \exp(-\log G(\theta_i)) d\theta_1, \dots, d\theta_n$$

Poichè il termine agisce solo su variabili casuali individuali, la sostituzione nel MRF non viola il teorema di Hammersley-Clifford.

Quando i parametri $(\theta_1, \dots, \theta_n)$ sono campionati da $G \sim DP(mG_0)$, i θ_i sono condizionatamente indipendenti data G e la loro distribuzione congiunta assume la forma:

$$P(\theta_1, \dots, \theta_n | G) = \prod_{i=1}^n G(\theta_i).$$

Questa indipendenza giustifica la sostituzione fatta.

Il modello è combinato con una verosimiglianza parametrica $F(\cdot | \theta)$ assumendo che i dati osservati siano generati da:

$$(x_1, \dots, x_n) \sim \prod_{i=1}^n F(x_i | \theta_i)$$

$$(\theta_1, \dots, \theta_n) \sim M(\theta_1, \dots, \theta_n) \prod_{i=1}^n G(\theta_i)$$

$$G \sim DP(mG_0).$$

La prior condizionata risultante è:

$$P(\theta_i | \theta_{-i}) = \sum_{k=1}^{N_C} \frac{n_k^{-i}}{n-1+m} \delta_{\theta_k^*}(\theta_i) + \frac{m}{n-1+m} G_0(\theta_i),$$

dove n_k^{-i} denota sempre il numero di campioni nel gruppo k , in cui è indicata l'esclusione di θ_i .

2.2.5 Simulazione dalla posterior del modello

L'applicazione di questo approccio richiede un metodo per stimare lo stato del modello dai dati. L'inferenza è di solito fatta mediante un algoritmo *Markov Chain Monte Carlo* (MCMC), in particolare si utilizza l'algoritmo *Gibbs sampling*.

Il campionamento del modello DPM/MRF è una versione modificata dell'algoritmo di MacEachern's (MacEachern e Muller, 1998). L'algoritmo originale è composto da due passi, uno di assegnamento e uno di aggiornamento dei parametri. Per ogni valore x_i , il passo di assegnamento calcola un vettore di probabilità q_{ik} che x_i sia assegnato alla classe k , secondo lo stato corrente del modello. Si stima anche una probabilità addizionale q_{i0} per la creazione di una nuova classe per spiegare x_i . Gli indicatori S_i sono campionati da queste probabilità di assegnamento. Dato un set completo di S_i , si calcola una probabilità a posteriori per ogni classe, basata sui dati che contiene, e i parametri θ_k^* del modello sono aggiornati mediante il campionamento dalla distribuzione a posteriori.

Questo algoritmo, simile all'algoritmo EM per modelli a mistura finita visto nel Sottoparagrafo 2.1.1, viene adattato per ottenere un campionario del DPM/MRF:

Inizializzazione:

- 1 Si genera un singolo cluster contenente tutti i punti:

$$\theta_1^* \sim G_0(\theta_1^*) \prod_{i=1}^n F(x_i|\theta_1^*)$$

Iterazione:

- 2 Si genera una permutazione σ di tutti gli indici dei dati.
 3 *Passo di assegnamento.* Per $i = \sigma(1), \dots, \sigma(n)$:
 a) Se x_i è l'unica osservazione assegnata al cluster $k = S_i$, il cluster viene rimosso.
 b) Calcolo delle probabilità dei cluster

$$q_{i0} \propto m \int_{\Omega_\theta} F(x_i|\theta) G_0(\theta) d\theta$$

$$q_{ik} \propto n_k^{-i} \exp(-H(\theta_k^*|\theta_{-i})) F(x_i|\theta_k^*)$$

- c) Costruzione di un indice casuale k secondo la distribuzione $(q_{i0}, \dots, q_{iN_C})$
 d) Assegnamento:
 - Se $k \in \{1, \dots, N_C\}$, si assegna x_i al cluster k .
 - Se $k = 0$, si crea un nuovo cluster per x_i .
 4 *Aggiornamento dei parametri.* Per ogni cluster $k = 1, \dots, N_C$ si aggiornano i θ_k^* dati gli assegnamenti S_1, \dots, S_n campionando

$$\theta_k^* \sim G_0(\theta_k^*) \prod_{i|S_i=k} F(x_i|\theta_k^*)$$

Modo di assegnamento:

- 5 Per ogni punto si sceglie il cluster assegnato più frequentemente durante un numero di iterazioni scelto.

2.2.6 Applicazione alle immagini

Il metodo descritto viene utilizzato per la segmentazione delle immagini, scegliendo delle distribuzioni appropriate per G_0 , F e la funzione costo H . Innanzitutto dall'immagine vengono estratti gli istogrammi locali. Questo processo viene effettuato posizionando una griglia equidistanziata sull'immagine di input. I pixel coincidenti con i nodi della griglia vengono identificati con le posizioni, indicizzate da $i = 1, \dots, n$. Una finestra è collocata su ogni posizione e viene costruito un istogramma $\mathbf{h}_i = (h_{i1}, \dots, h_{iN_{bins}})$. La dimensione della finestra, e quindi il numero N_{counts} di dati registrati in ogni

istogramma, viene tenuta costante per l'intera immagine, così come il numero N_{bins} di "contenitori" dell'istogramma. Gli istogrammi $\mathbf{h}_1, \dots, \mathbf{h}_n$ sono i dati di input per l'algoritmo di classificazione, sostituiscono le osservazioni x_1, \dots, x_n .

I parametri θ_i campionati dal processo di Dirichlet nel modello DPM sono in questo caso le probabilità dei rettangoli degli istogrammi: θ_{ij} è la probabilità per un valore di capitare nel rettangolo j di un istogramma nel punto i . Date queste probabilità, gli istogrammi sono distribuiti multinomialmente e la verosimiglianza è data da:

$$F(\mathbf{h}_i|\theta_i) = N_{counts}! \prod_{j=1}^{N_{bins}} \frac{\theta_{ij}^{h_{ij}}}{h_{ij}!} = \frac{1}{Z_M(\mathbf{h}_i)} \exp \left(\sum_{j=1}^{N_{bins}} h_{ij} \log(\theta_{ij}) \right).$$

Per i parametri θ_i si assume a priori una distribuzione coniugata, quindi una Dirichlet di dimensione N_{bins} . Questa distribuzione ha due parametri β, π , dove β è uno scalare positivo e π è un vettore di probabilità N_{bins} -dimensionale:

$$\begin{aligned} G_0(\theta_i|\beta\pi) &= \frac{\Gamma(\beta)}{\prod_{j=1}^{N_{bins}} \theta_{ij}^{\beta\pi_j - 1}} \\ &= \frac{1}{Z_D(\beta\pi)} \exp \left(\sum_{j=1}^{N_{bins}} (\beta\pi_j - 1) \log(\theta_{ij}) \right). \end{aligned}$$

Come funzione costo, si sceglie una funzione del tipo

$$H(\theta_i|\theta_{-i}) = -\lambda \sum_{l \in \partial(i)} \delta_{\theta_i, \theta_l}. \quad (2.8)$$

Per il campionamento di questo particolare modello, si utilizza l'algoritmo visto precedentemente. Poiché F e G_0 sono coniugate, il loro prodotto è una distribuzione di Dirichlet:

$$\begin{aligned} F(\mathbf{h}_i|\theta_i)G_0(\theta_i) &\propto \exp \left(\sum_j (h_{ij} + \beta\pi_j - 1) \log(\theta_{ij}) \right) \\ &= G_0(\theta_i|\mathbf{h}_i + \beta\pi). \end{aligned}$$

q_{i0} è calcolato come:

$$\int_{\Omega_\theta} F(\mathbf{h}_i|\theta_i)G_0(\theta_i) d\theta_i = \frac{Z_D(\mathbf{h}_i + \beta\pi)}{Z_M(\mathbf{h}_i)Z_D(\beta\pi)},$$

mentre

$$\begin{aligned} q_{ik} &\propto n_k^{-i} \exp(-H(\theta_k^*|\theta_i))F(\mathbf{h}_i|\theta_k^*) \\ &= \frac{n_k^{-i}}{Z_M(\mathbf{h}_i)} \exp\left(\lambda \sum_{l \in \partial(i)} \delta_{\theta_i, \theta_l} + \sum_j h_{ij} \log(\theta_{kj}^*)\right). \end{aligned}$$

Poiché la funzione di ripartizione $Z_M(\mathbf{h}_i)$ compare in tutte le equazioni, le probabilità delle classi possono essere ricavate per ogni i calcolando i valori iniziali:

$$\begin{aligned} \tilde{q}_{i0} &:= \frac{Z_D(\mathbf{h}_i + \beta\pi)}{Z_D(\beta\pi)} \\ \tilde{q}_{ik} &:= n_k^{-i} \exp\left(\lambda \sum_{l \in \partial(i)} \delta_{\theta_i, \theta_l} + \sum_j h_{ij} \log(\theta_{kj}^*)\right). \end{aligned}$$

Da questi valori iniziali, le probabilità delle classi possono essere calcolate come

$$q_{ik} := \frac{\tilde{q}_{ik}}{\sum_{l=0}^{N_C} \tilde{q}_{il}}.$$

La probabilità a posteriori da cui campionare θ_k^* è ancora una distribuzione di Dirichlet:

$$\begin{aligned} &G_0(\theta_k^*|\beta\pi) \prod_{i|S_i=k} F(\mathbf{h}_i|\theta_k^*) \\ &\propto \exp\left(\sum_j \left(\beta\pi_j + \sum_{i|S_i=k} h_{ij} - 1\right) \log(\theta_{kj}^*)\right) \\ &\propto G_0\left(\theta_k^*|\beta\pi - \sum_{i|S_i=k} \mathbf{h}_i\right). \end{aligned}$$

Il modello di classificazione è di facile comprensione: ogni istogramma \mathbf{h}_i è generato dalla verosimiglianza F , che è determinata in ogni punto dal parametro θ_i . La sua distribuzione $G_0(\theta_i|\beta\pi)$ è parametrizzata da π , che definisce il valore atteso di G_0 , e lo scalare β , che controlla la dispersione della distribuzione: più è grande il valore di β , più la distribuzione sarà concentrata intorno a π e la probabilità a priori influenzerà il modello. Definendo

$$\tilde{\beta}_k := \left\| \beta\pi + \sum_{i|S_i=k} \mathbf{h}_i \right\|_1,$$

$$\tilde{\pi}_k := \frac{1}{\tilde{\beta}_k} \left(\beta\pi + \sum_{i|S_i=k} \mathbf{h}_i \right),$$

la probabilità a posteriori assume la forma $G_0(\cdot|\tilde{\beta}_k\tilde{\pi}_k)$. Per ogni classe k , il valore atteso della distribuzione a posteriori è $\tilde{\pi}_k$, cioè la media normalizzata degli istogrammi assegnati alla classe, e la dispersione è determinata da $\tilde{\beta}_k$.

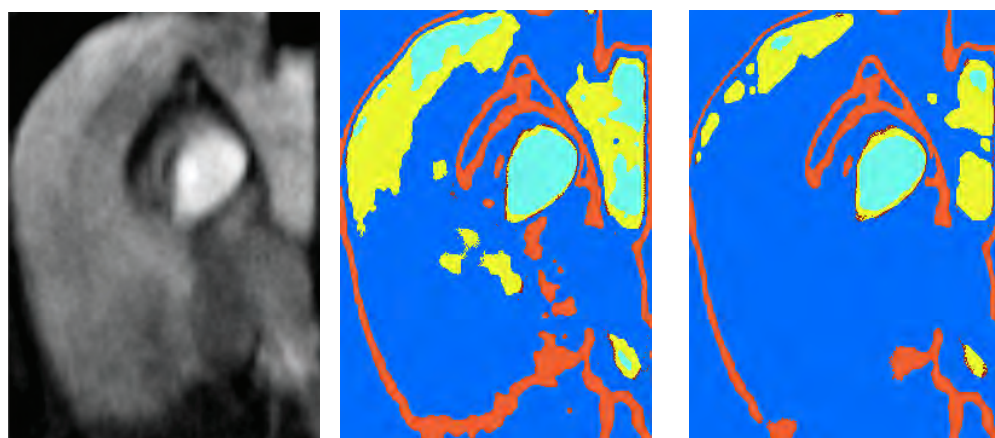
Gli assegnamenti alle classi sono determinati campionando dalle distribuzioni $(q_{i0}, \dots, q_{iN_C})$, che sono basate sulla verosimiglianza multinomiale F .

Per ottenere i risultati che verranno mostrati nel Capitolo 3, è stato utilizzato il codice Matlab proposto da Orbanz e Buhmann (Orbanz e Buhmann, 2008), da cui sono state successivamente estratte le probabilità a posteriori per ogni pixel di appartenere alle classi costruite dall'algoritmo. Esse sono infatti le probabilità Pr^* necessarie per costruire il potenziale $P(x, y)$ introdotto nel Sottoparagrafo 1.8.2 che serve a muovere lo snake verso i contorni dell'oggetto di interesse.

2.2.7 Estensione del modello: informazione ai bordi

I vincoli di smoothing possono portare ad effetti di accoppiamento indesiderati vicino ai contorni dell'immagine: due pixel che appartengono a classi differenti, a causa dello smoothing possono essere assegnati alla stessa classe.

L'informazione sui bordi dell'immagine, in aggiunta ai dati di input dell'algoritmo, può essere usata per migliorare la risoluzione dei contorni. Assumiamo a priori che due pixel non debbano essere accoppiati da un vincolo di smoothing se sono separati da un bordo dell'immagine. Quindi l'infor-



(a) Immagine originale (b) Clusterizzazione con informazione ai bordi (c) Clusterizzazione senza informazione ai bordi

Figura 2.3: Confronto della clusterizzazione di un'immagine medica mediante il metodo DPM/MRF con e senza l'informazione ai bordi.

mazione ai bordi può essere presa in considerazione nel modello DPM/MRF modificando il grafo \mathcal{N} del campo aleatorio markoviano:

1. Generazione di una mappa dei bordi utilizzando la funzione $P(x, y)$ descritta nel Sottoparagrafo 1.8.1.
2. Se due pixel i e j sono separati da un bordo, si impone $w_{ij} = w_{ji} = 0$.

In pratica, poiché il vincolo MRF agisce lungo i bordi di grafi, vengono eliminati gli accoppiamenti tra i due istogrammi corrispondenti \mathbf{h}_i e \mathbf{h}_j . Data una mappa dei bordi, l'algoritmo determina dove due pixel sono separati da un bordo.

In Figura 2.3 si osserva come vengono identificati anche i bordi con un contrasto più basso, grazie all'aggiunta dell'informazione ai bordi.

2.2.8 Convergenza del modello

Gli algoritmi Gibbs sampling sono notoriamente molto lenti e spesso è difficile determinare dopo quante iterazioni raggiungono la distribuzione di interesse; solitamente sono necessarie migliaia di iterazioni prima di arrivare a risultati soddisfacenti.

Il vantaggio del metodo proposto è che l'obiettivo non è la stima dei parametri della distribuzione di interesse, scopo per cui viene in genere utilizzato il modello DPM; l'intento non è nemmeno quello di Orbanz e Buhmann, cioè

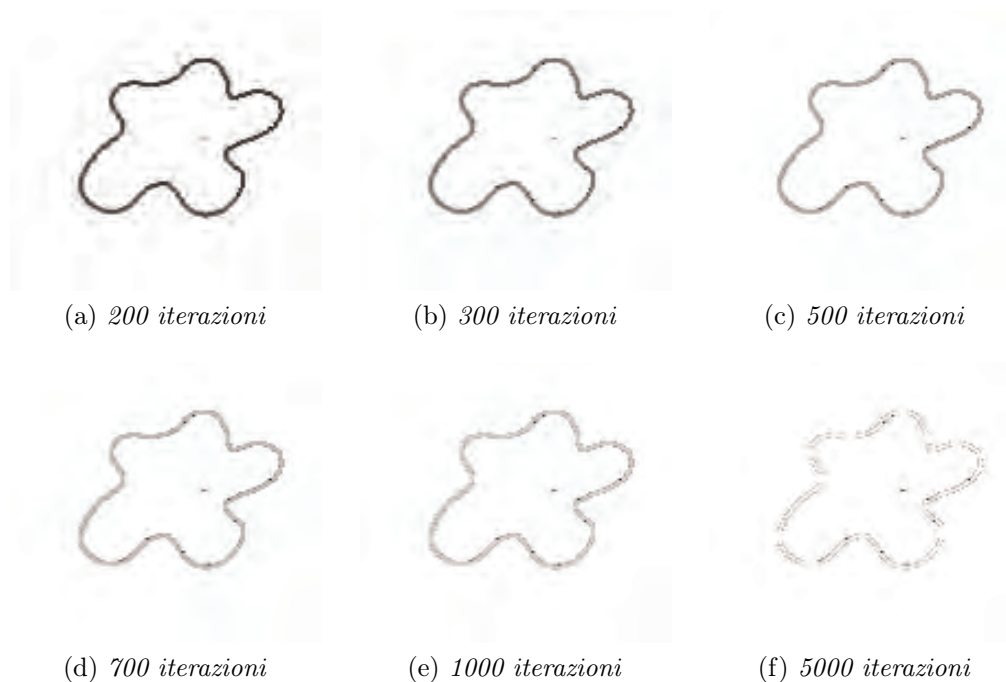


Figura 2.4: Potenziale non parametrico $P(x, y)$ dell'immagine test, ottenuto al variare del numero di iterazioni.

ottenere una classificazione perfetta dell'immagine medica in cui a posteriori la probabilità che un pixel appartenga alla classe esatta sia 1. Quello che si vuole ottenere, infatti, è un riconoscimento preciso dei contorni. Per questo motivo, mentre per un'ottima classificazione sono necessarie almeno 500 iterazioni prima di raggiungere un numero stabile di classi, per l'obiettivo proposto in questa tesi ne bastano meno.

Se si effettuassero migliaia di iterazioni, i pixel sarebbero assegnati alle rispettive classi con maggiore probabilità, ma questo porterebbe ad un valore della forza probabilistica descritta nel Sottoparagrafo 1.8.2 pari a 1, cioè lo stesso valore che assumono i pixel non appartenenti ai bordi. Ciò significa che non si riuscirebbero più ad ottenere i contorni delle immagini mediante la forza introdotta, di conseguenza lo snake non sarebbe guidato dalla forza esterna.

In Figura 2.4 si mostra un confronto effettuato sull'immagine test, facendo variare il numero di iterazioni. Come si nota, effettuando migliaia di iterazioni il contorno inizia ad essere meno visibile in alcuni punti. Si osserva inoltre che già da 200 iterazioni (Figura 2.4a) il contorno della figura è visibile perfettamente.

Tali prove sono state effettuate anche sulle immagini mediche, giungendo alle stesse conclusioni. Per questo motivo si è deciso di considerare in tutti i test un numero di iterazioni compreso tra 200 e 300.

Inoltre, c'è un ulteriore vantaggio, rappresentato dal fatto che non è necessario stimare in modo preciso il parametro di precisione m . Questo parametro, infatti, definisce in pratica il livello di risoluzione dell'immagine e quindi il numero di classi: più il valore è piccolo, più diminuisce la risoluzione della segmentazione. Poiché non serve ottenere una stima precisa di questo numero, si vedrà nel capitolo successivo che è sufficiente attribuire ad m un valore compreso tra 10^{-1} e 10^{-4} ; diminuendo ulteriormente tale valore, si è notata una difficoltà nell'individuare i bordi più deboli.

Come si vedrà nel capitolo successivo, questo nuovo approccio proposto porta a riconoscimenti dei contorni molto più precisi, anche nei casi in cui il poco contrasto delle immagini non consente di individuare con facilità alcuni bordi. Inoltre, come già detto prima, non c'è più bisogno di applicare opportuni filtri alle immagini rischiando di sfocarne troppo i contorni.

Capitolo 3

Risultati sperimentali

In questo capitolo si riportano tutti i risultati sperimentali ottenuti utilizzando i metodi descritti precedentemente. Tali risultati si confrontano con quelli conseguiti mediante un metodo deterministico classico di segmentazione, in cui la forza esterna utilizzata è costruita mediante la funzione (1.5). Per poter visualizzare meglio le differenze dei vari metodi, si presentano ogni volta i risultati ottenuti per ogni immagine.

3.1 Immagine test

Inizialmente tutte le prove sono state effettuate sull'immagine test (Figura 3.1a). Più che per notare le differenze, sostanzialmente molto piccole, si mostrano i risultati per confermare l'efficacia dei vari metodi. Prima di procedere con le analisi sulle immagini mediche è stato necessario infatti testare su un'immagine semplice la validità di tutti i modelli considerati.

Si indicheranno d'ora in poi $P_{deterministica}$, $P_{parametrica}$ e $P_{nonparametrica}$ le funzioni $P(x, y)$ calcolate rispettivamente con il metodo deterministico (Sottoparagrafo 1.8.1), con il metodo probabilistico parametrico e con il metodo probabilistico non parametrico (Capitolo 2).

Per il metodo parametrico sono state fissate a priori 2 classi, mentre per il metodo bayesiano non parametrico è stato assegnato un valore $m = 10^{-4}$ (parametro del processo di Dirichlet), $\lambda = 3$ (costante della funzione costo (2.8)) e 200 iterazioni. Questo ha portato all'identificazione di 3 classi.

In Figura 3.1 si mostrano le funzioni probabilistiche e quella deterministica, che identificano i contorni dell'immagine. Da esse sono successivamente calcolate le rispettive forze esterne (Figura 3.2). La forza combinata, ottenuta mettendo come $F_{probabilistica}$ dell'Equazione (1.6) la forza parametrica, è calcolata in modo da dare lo stesso contributo alle due forze poiché entrambi

i metodi riconoscono in modo preciso i bordi dell'immagine; sono stati scelti quindi i valori $w_1 = 0.5$ e $w_2 = 0.5$.

$P_{deterministica}$ e $P_{parametrica}$ sono ottenute dall'immagine filtrata (si veda il Sottoparagrafo 1.9), utilizzando come maschera di convoluzione una funzione gaussiana a media nulla e varianza $\sigma^2 = 4$, mentre la funzione non parametrica è calcolata direttamente dall'immagine originale.

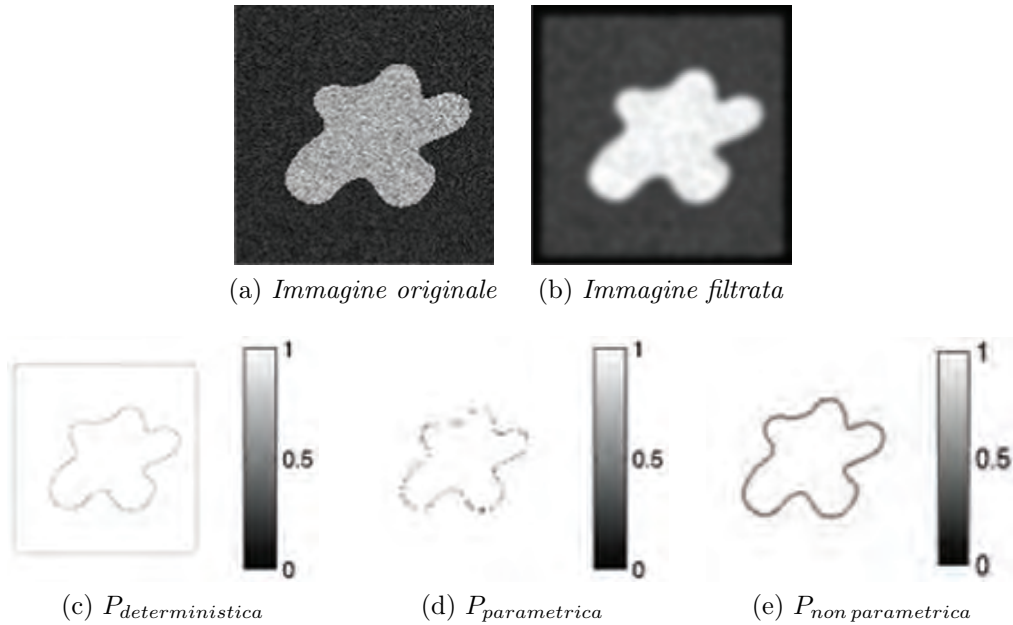


Figura 3.1: Immagine test - confronto di $P(x, y)$ per ogni modello.

Per questa immagine in tutti i modelli sono stati scelti i parametri, relativi all'energia interna descritta nel Sottoparagrafo 1.5, $\alpha = 0.5$ (lunghezza dello snake) e $\beta = 0.5$ (curvatura dello snake); si è scelto un numero di iterazioni $N = 200$ per tutti i modelli tranne che per l'inizializzazione esterna di quello combinato, in cui ne sono bastate 150. I pesi k_1 e k_2 , relativi rispettivamente alla forza esterna ed al termine di espansione, variano a seconda del modello utilizzato. La loro scelta dipende infatti dal tipo di inizializzazione della curva e da quanto forza esterna e pressione sono in grado di controbilanciarsi. Nel caso di inizializzazione interna, per il modello deterministico si sono scelti $k_1 = 1$ e $k_2 = 0.5$, per il parametrico $k_1 = 0.4$ e $k_2 = 0.2$, per il combinato $k_1 = 0.4$ e $k_2 = 0.25$, mentre per il non parametrico $k_1 = 0.5$ e $k_2 = 0.25$. I parametri sono riassunti in Tabella 3.1. Nel caso di inizializzazione esterna all'oggetto, si sono scelti $k_1 = 0.5$ e $k_2 = -0.3$ per il deterministico, $k_1 = 0.4$ e $k_2 = -0.2$ per il parametrico, $k_1 = 0.4$ e $k_2 = -0.3$ per il combinato e $k_1 = 0.5$ e $k_2 = -0.2$ per il non parametrico (Tabella 3.2).

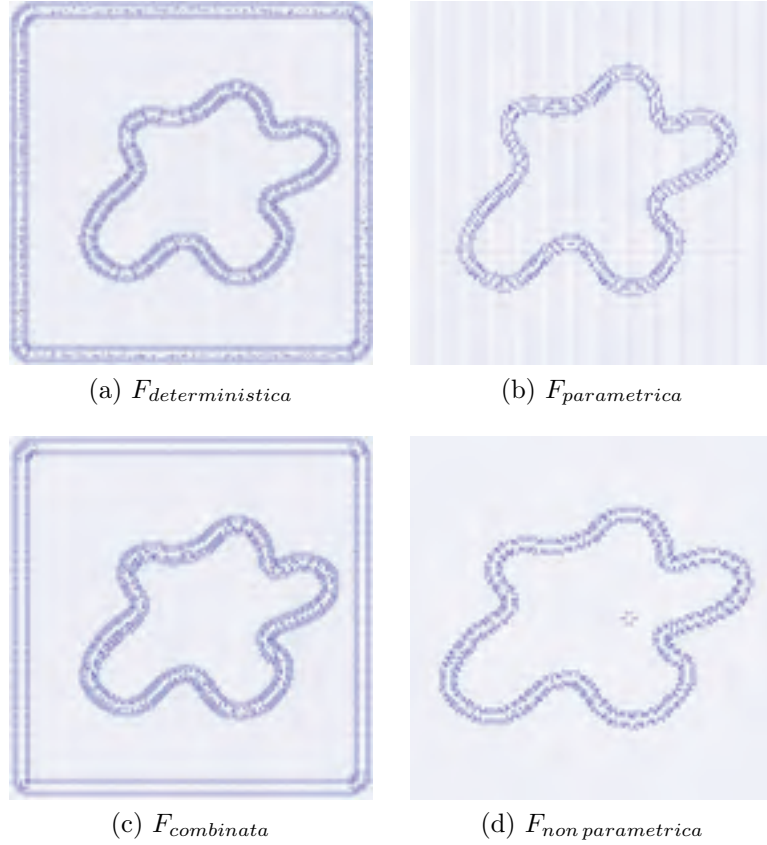


Figura 3.2: Immagine test - campi di forze.

Modello	α	β	k_1	k_2	N
deterministico	0.5	0.5	1	0.5	200
parametrico	0.5	0.5	0.4	0.2	200
parametrico combinato	0.5	0.5	0.4	0.25	200
non parametrico	0.5	0.5	0.5	0.25	200

Tabella 3.1: Parametri dello snake applicato all'immagine test - inizializzazione interna.

Modello	α	β	k_1	k_2	N
deterministico	0.5	0.5	0.5	-0.3	200
parametrico	0.5	0.5	0.4	-0.2	200
combinato	0.5	0.5	0.4	-0.3	150
non parametrico	0.5	0.5	0.5	-0.2	200

Tabella 3.2: Parametri dello snake applicato all'immagine test - inizializzazione esterna.

I contorni sono identificati bene con tutti i metodi; una volta superato il problema del rumore presente nell'immagine originale, l'immagine test non presenta mancanze di contrasto o bordi frastagliati che ne impediscono una buona segmentazione.

In Figura 3.3 si mostra uno zoom sul campo di forze creato per mezzo del modello parametrico.

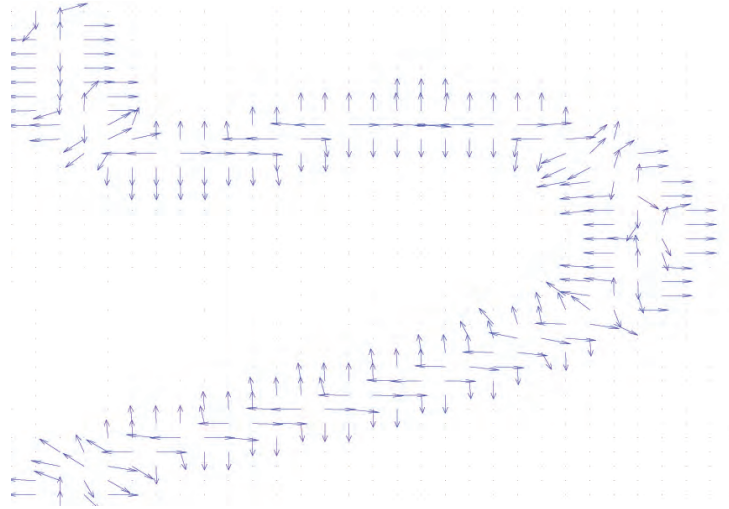


Figura 3.3: Zoom $F_{parametrica}$

In Figura 3.4 e Figura 3.5 si mostrano i risultati dello snake deformato mediante forza interna, pressione e forza esterna. Come ci si aspettava, i risultati ottenuti sono molto simili tra loro, sia nel caso in cui l'inizializzazione sia effettuata internamente all'oggetto, sia nel caso di inizializzazione esterna.

I metodi proposti funzionano molto bene sull'immagine test, quindi si procede a verificarne l'efficacia su alcune immagini mediche.

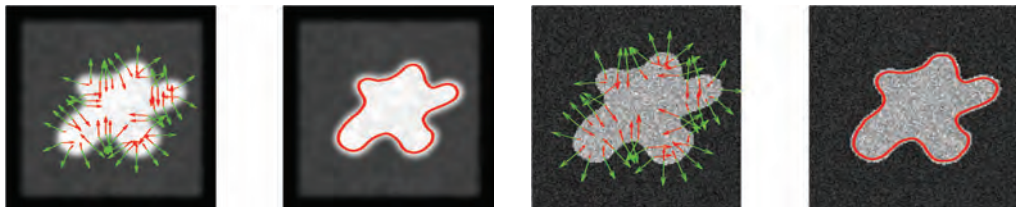


(a) *Inizializzazione*



(b) *Modello deterministico*

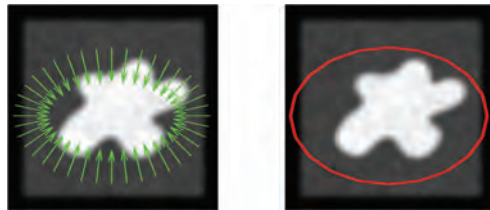
(c) *Modello parametrico*



(d) *Modello parametrico combinato*

(e) *Modello non parametrico*

Figura 3.4: Snake applicato all'immagine test - inizializzazione interna.

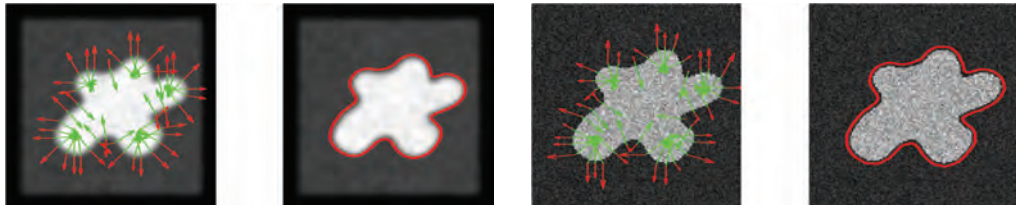


(a) *Inizializzazione*



(b) *Modello deterministico*

(c) *Modello parametrico*



(d) *Modello parametrico combinato*

(e) *Modello non parametrico*

Figura 3.5: Snake applicato all'immagine test - inizializzazione esterna.

3.2 Valvola cardiaca

La prima immagine medica analizzata è quella di una valvola cardiaca. L'obiettivo è riuscire a identificare l'orifizio valvolare, cioè la parte bianca al centro della Figura 3.6a. L'immagine filtrata (Figura 3.6b) è ottenuta mediante un filtro con varianza $\sigma^2 = 49$.

Il modello parametrico è costruito dando a priori un numero di classi pari a 4. Per il modello non parametrico sono stati dati i valori $m = 10^{-2}$ e $\lambda = 5$; il numero di iterazioni effettuate è pari a 300. Per la forza combinata sono stati scelti i pesi $w_1 = 0.8$ (forza probabilistica) e $w_2 = 0.2$ (forza deterministica). Sia in Figura 3.6 che in Figura 3.7 si può notare che con il modello deterministico la segmentazione non è molto precisa, mentre con gli approcci probabilistici i contorni sono più dettagliati, in particolare con il modello non parametrico.

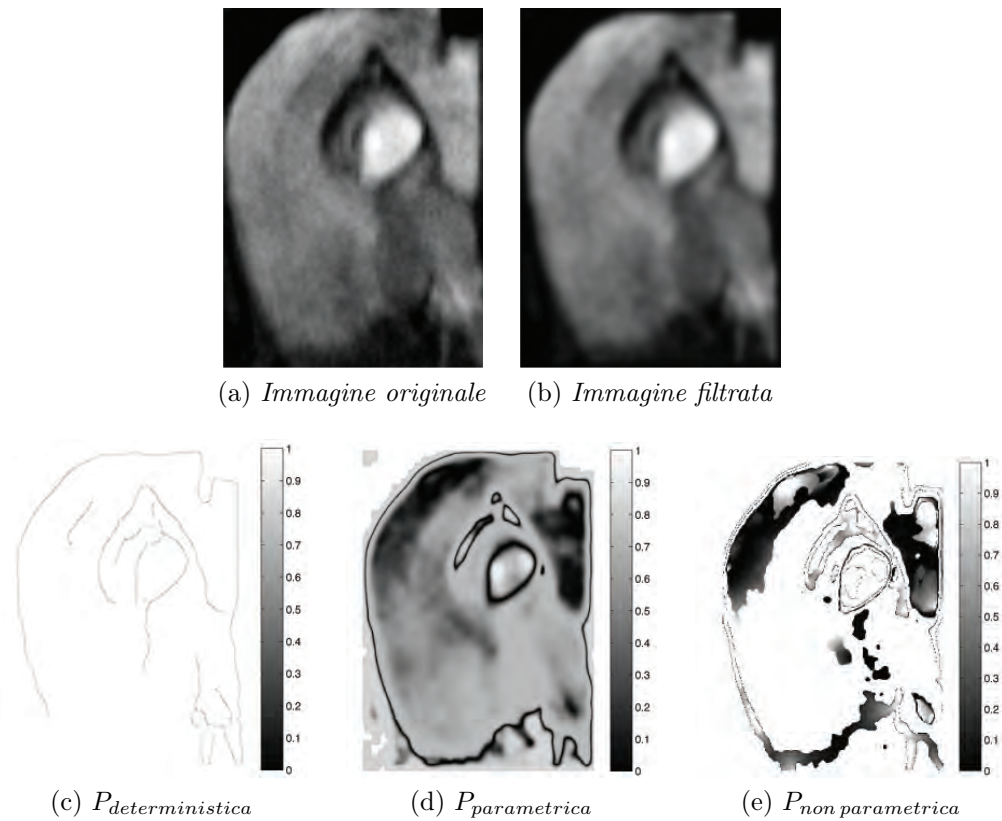


Figura 3.6: Valvola cardiaca - confronto di $P(x, y)$ per ogni modello.

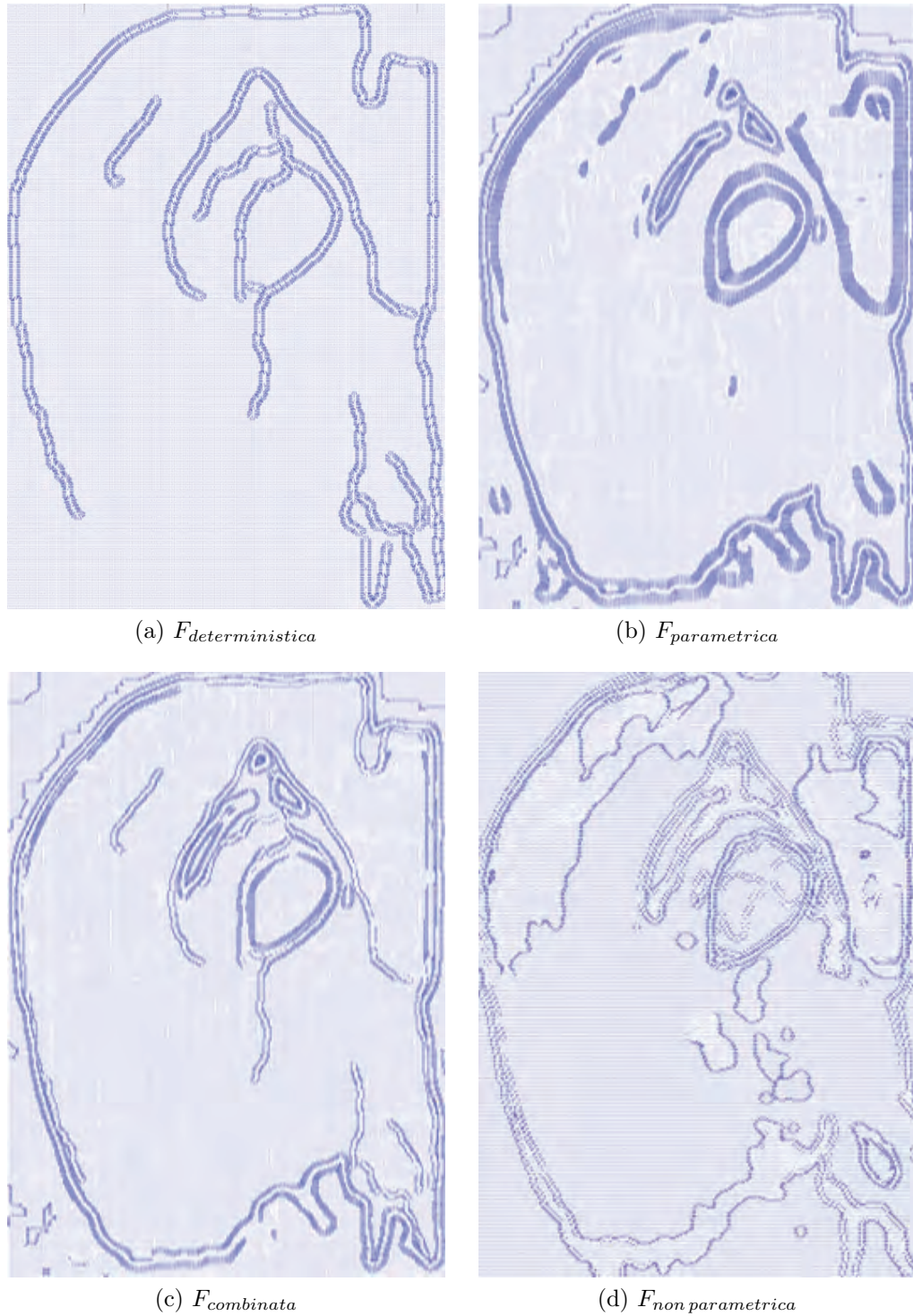


Figura 3.7: Valvola cardiaca - confronto dei campi di forze.

Modello	α	β	k_1	k_2	N
deterministico	0.5	0.3	0.3	0.15	300
parametrico	0.5	0.3	0.3	0.3	150
parametrico combinato	0.5	0.3	0.3	0.3	100
non parametrico	0.5	0.3	0.5	0.2	500

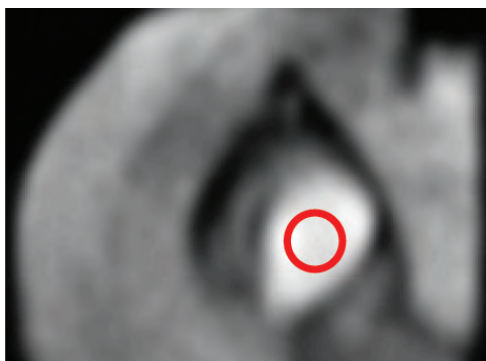
Tabella 3.3: Parametri dello snake applicato alla valvola cardiaca - inizializzazione interna.

Modello	α	β	k_1	k_2	N
deterministico	0.5	0.3	0.4	-0.15	200
parametrico	0.5	0.3	0.3	-0.2	150
parametrico combinato	0.5	0.3	0.3	-0.2	150
non parametrico	0.5	0.3	0.45	-0.2	600

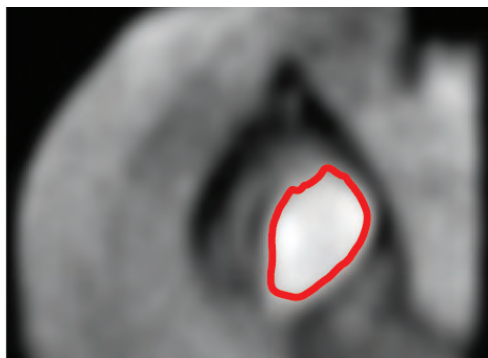
Tabella 3.4: Parametri dello snake applicato alla valvola cardiaca - inizializzazione esterna.

In tutti i casi sono stati scelti i parametri $\alpha = 0.5$ e $\beta = 0.3$, mentre il numero di iterazioni N in questo caso varia in base al modello e all'inizializzazione dello snake. Nel caso di inizializzazione interna, per il modello deterministico si sono scelti $k_1 = 0.3$, $k_2 = 0.15$ e $N = 300$, per il parametrico $k_1 = 0.3$, $k_2 = 0.3$ e $N = 150$, per il combinato $k_1 = 0.3$, $k_2 = 0.3$ e $N = 100$, mentre per il non parametrico $k_1 = 0.5$, $k_2 = 0.2$ e $N = 500$. Tutti i parametri sono riportati in Tabella 3.3. Per l'inizializzazione esterna, come riassunto in Tabella 3.4, si sono scelti $k_1 = 0.4$, $k_2 = -0.15$ e $N = 200$ per il deterministico, $k_1 = 0.3$, $k_2 = -0.2$ e $N = 150$ per il parametrico, $k_1 = 0.3$, $k_2 = -0.2$, e $N = 150$ per il combinato, $k_1 = 0.45$, $k_2 = -0.2$ e $N = 600$ per il non parametrico.

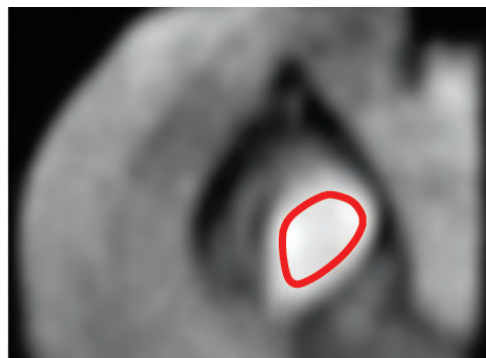
Il difetto del modello parametrico è che, se lo snake è inizializzato all'interno, non riesce ad arrivare fino al bordo perché è bloccato dalla forza esterna che in questo caso è troppo "conservativa" (Figura 3.8c); questo problema si ripercuote di conseguenza anche sul modello combinato (Figura 3.8d). Per quanto riguarda l'inizializzazione esterna (Figura 3.9), il modello deterministico dà un risultato molto simile al precedente, mentre si hanno dei miglioramenti per il modello parametrico e quello combinato.



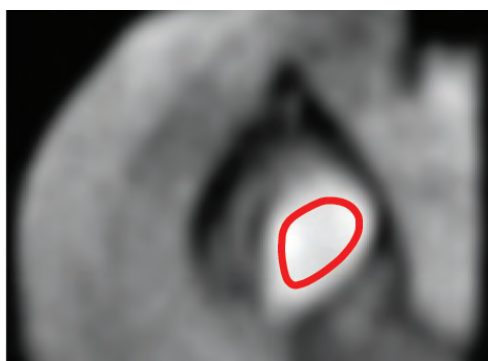
(a) *Inizializzazione*



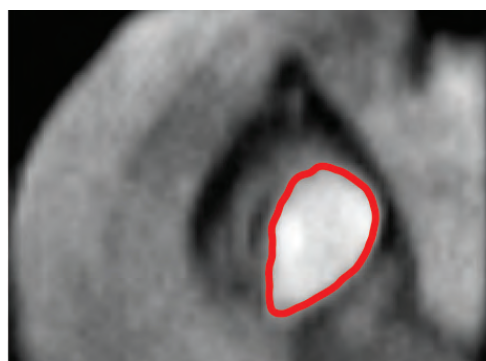
(b) *Modello deterministico*



(c) *Modello parametrico*



(d) *Modello parametrico combinato*

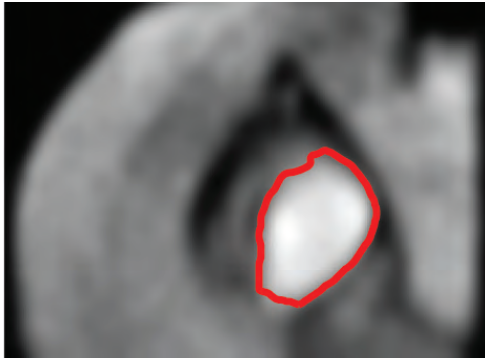


(e) *Modello non parametrico*

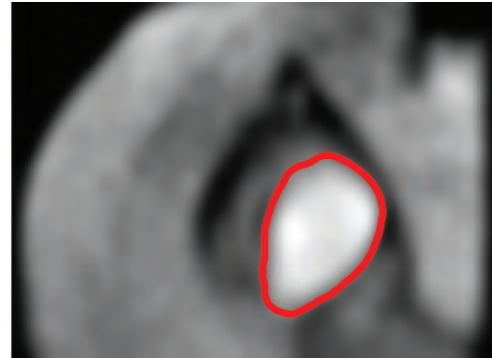
Figura 3.8: Snake applicato alla valvola cardiaca - inizializzazione interna all'orifizio valvolare.



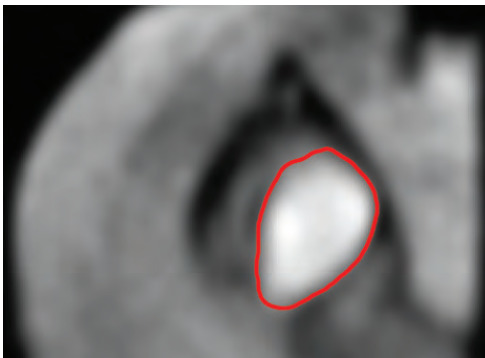
(a) *Inizializzazione*



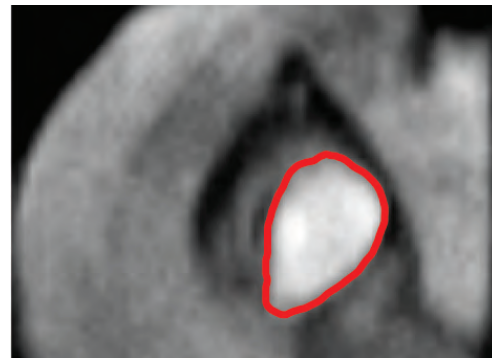
(b) *Modello deterministico*



(c) *Modello parametrico*



(d) *Modello parametrico combinato*



(e) *Modello non parametrico*

Figura 3.9: Snake applicato alla valvola cardiaca - inizializzazione esterna all'orifizio valvolare.

Si nota quindi già da questa prima immagine come l'approccio non parametrico sia più "stabile" rispetto agli altri, nel senso che il risultato che si ottiene è più simile, rispetto agli altri metodi, sia inizializzando lo snake dall'interno, sia inizializzandolo dall'esterno, come si può osservare anche nello zoom in Figura 3.10.

La differenza dei campi di forze in questo punto dell'orifizio porta a segmentazioni diverse: il campo di forze deterministico e quello parametrico sono molto forti e creano in pratica due bordi più spessi rispetto a quello non parametrico, di conseguenza l'inizializzazione dello snake internamente o esternamente all'oggetto porta a risultati differenti. La segmentazione è inoltre più imprecisa poiché in quel punto il contorno dell'immagine filtrata non è ben definito. La maggior precisione dell'approccio non parametrico è infatti data anche al fatto che l'immagine analizzata è quella originale e non la filtrata, perciò il contorno è individuato in maniera più dettagliata.

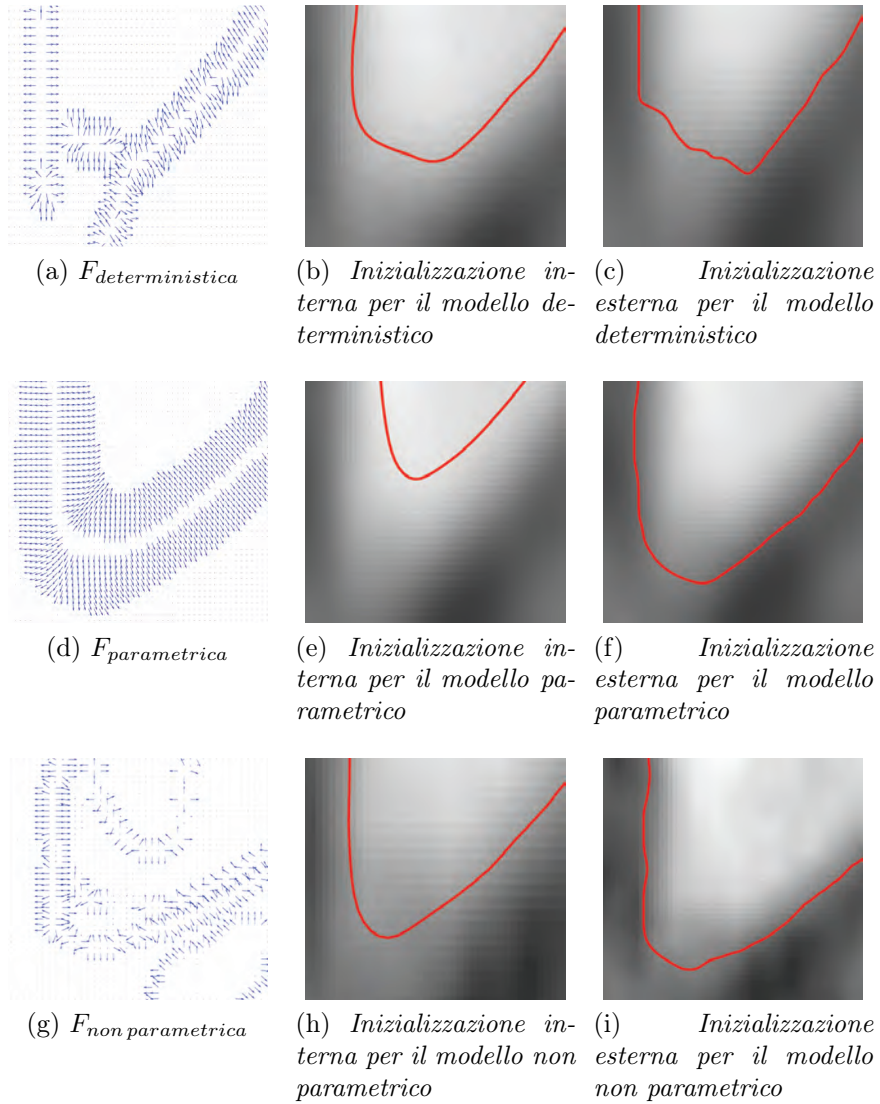


Figura 3.10: Zoom sulla segmentazione di una zona dell'orifizio valvolare.

3.3 Cervello

La seconda immagine analizzata è quella di un cervello, in cui l'obiettivo è segmentare i due ventricoli, la materia grigia e la materia bianca.

In Figura 3.11 si mostrano l'immagine originale, la filtrata ottenuta con una maschera di convoluzione gaussiana di varianza $\sigma^2 = 36$, e i valori del potenziale P ottenuti mediante i metodi deterministico, parametrico e non parametrico.

Per il modello parametrico è stato scelto inizialmente un numero di cluster pari a 4. Per il modello bayesiano non parametrico sono stati scelti i valori $m = 10^{-2}$, $\lambda = 7$ e un numero pari a 200 iterazioni, ottenendo una suddivisione dell'immagine in 7 classi. Il valore di λ più alto rispetto a quello impostato per le altre immagini è giustificato dal fatto che questo tipo di scansione presenta più rumore rispetto alle altre, per cui è necessario un parametro di smoothing più elevato.

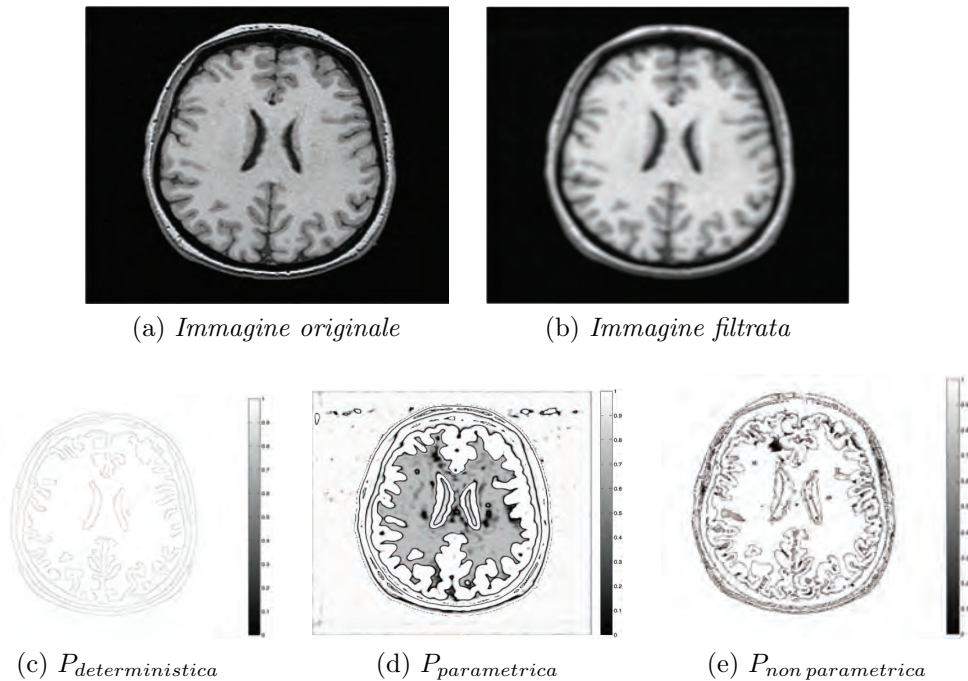


Figura 3.11: Cervello - confronto di $P(x, y)$ per ogni modello.

In Figura 3.12 si osservano i campi di forze ottenuti. Per il campo di forze combinato si sono scelti $w_1 = 0.3$, $w_2 = 0.7$. Si osservi come i ventricoli vengano identificati da tutti i metodi, mentre materia grigia e materia bianca sono più dettagliate in Figura 3.12b.



(a) $F_{deterministica}$



(b) $F_{parametrica}$



(c) $F_{combinata}$



(d) $F_{non\ parametrica}$

Figura 3.12: Cervello - confronto dei campi di forze.

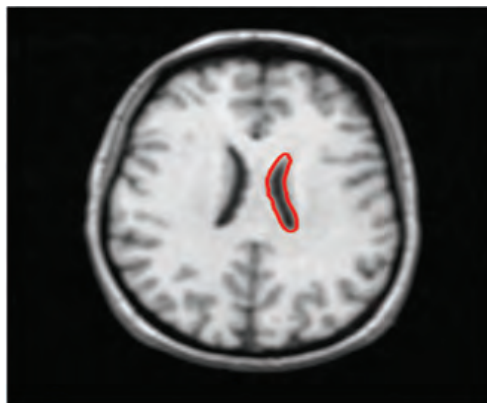
Modello	α	β	k_1	k_2	N
deterministico	0.5	0.5	0.5	-0.2	700
parametrico	0.5	0.5	0.6	-0.1	2500
parametrico combinato	0.5	0.5	0.5	-0.2	1000
non parametrico	0.5	0.5	0.5	-0.13	3000

Tabella 3.5: Parametri dello snake applicato al ventricolo destro del cervello.

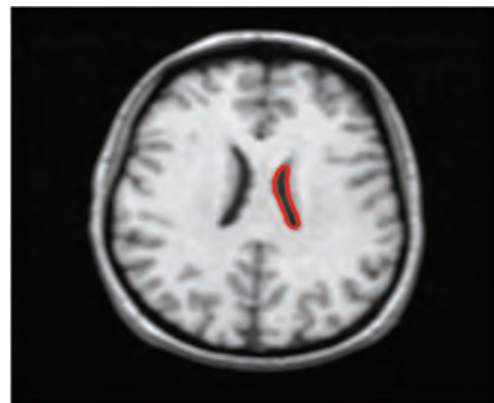
3.3.1 Ventricolo destro

In tutti i modelli sono stati scelti i parametri $\alpha = 0.5$ e $\beta = 0.5$. Per il modello deterministico si sono scelti, come riassunto in Tabella 3.5, $k_1 = 0.5$, $k_2 = -0.2$ e $N = 700$, per il parametrico $k_1 = 0.6$, $k_2 = -0.1$ e $N = 2500$, per il combinato $k_1 = 0.5$, $k_2 = -0.2$, e $N = 1000$, mentre per il non parametrico $k_1 = 0.5$, $k_2 = -0.13$ e $N = 3000$. Si noti in questo caso che il numero di iterazioni fornito per gli approcci probabilistici è più alto rispetto al caso deterministico. La motivazione è data dal fatto che, come si è detto precedentemente, mediante i metodi probabilistici si riesce ad individuare anche la materia grigia intorno al ventricolo; di conseguenza sono necessarie più iterazioni affinché lo snake riesca a superarla e fermarsi sul contorno del ventricolo.

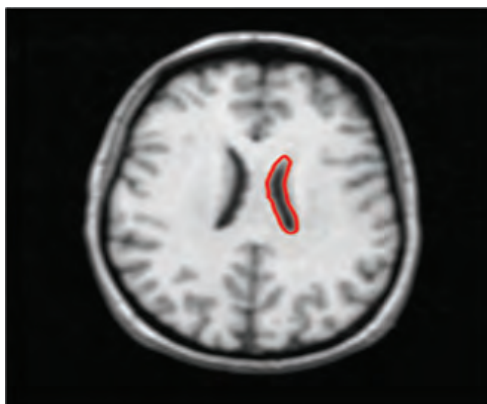
Il ventricolo destro è individuato in modo abbastanza preciso sia tramite l'approccio deterministico (Figura 3.13a) sia tramite quello non parametrico (Figura 3.13d). Il metodo parametrico (Figura 3.13b) presenta dei problemi nella parte in alto del ventricolo, a causa della materia grigia che lo circonda. Infatti in questo punto i contorni dell'immagine sono molto sfumati e quindi un peso troppo grande dato alla forza esterna, necessario per poter superare la materia grigia, porta ad un errore di segmentazione. Il modello combinato (Figura 3.13c) supera questo problema grazie al contributo della forza deterministica che, non "vedendo" la materia grigia, è molto più forte in questo punto.



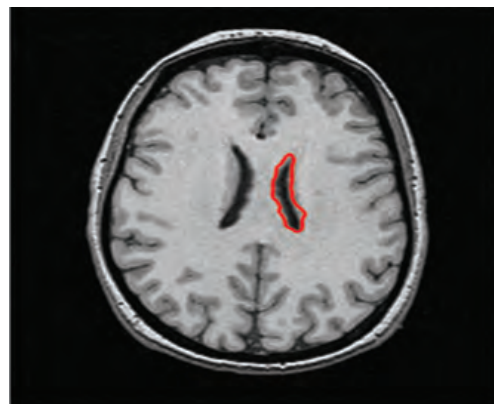
(a) *Modello deterministico*



(b) *Modello parametrico*



(c) *Modello parametrico combinato*



(d) *Modello non parametrico*

Figura 3.13: Snake applicato al cervello - inizializzazione esterna al ventricolo destro.

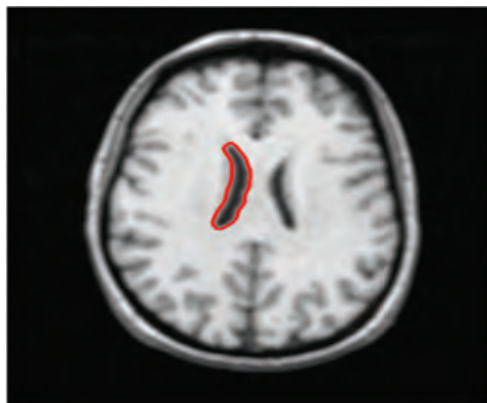
Modello	α	β	k_1	k_2	N
deterministico	0.5	0.5	0.5	-0.2	600
parametrico	0.5	0.5	0.4	-0.1	2000
parametrico combinato	0.5	0.5	0.4	-0.2	550
non parametrico	0.5	0.5	0.3	-0.12	3000

Tabella 3.6: Parametri dello snake applicato al ventricolo sinistro del cervello.

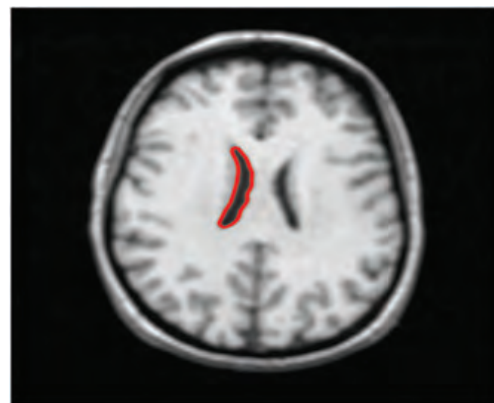
3.3.2 Ventricolo sinistro

Per il ventricolo sinistro sono stati scelti i valori per $\alpha = 0.5$ e $\beta = 0.5$. Per il modello deterministico si sono scelti $k_1 = 0.5$, $k_2 = -0.2$ e $N = 600$, per il parametrico $k_1 = 0.4$, $k_2 = -0.1$ e $N = 2000$, per il combinato $k_1 = 0.4$, $k_2 = -0.2$ e $N = 550$, per il non parametrico $k_1 = 0.3$, $k_2 = -0.12$ e $N = 3000$. I valori sono riassunti in Tabella 3.6

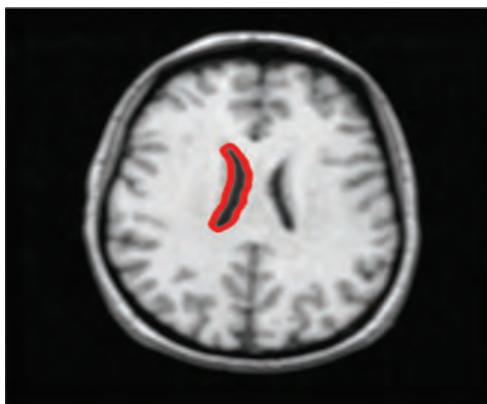
Il ventricolo sinistro presenta una maggior facilità nella segmentazione rispetto al ventricolo destro. In Figura 3.14 si può notare come tutti i metodi portino a risultati simili.



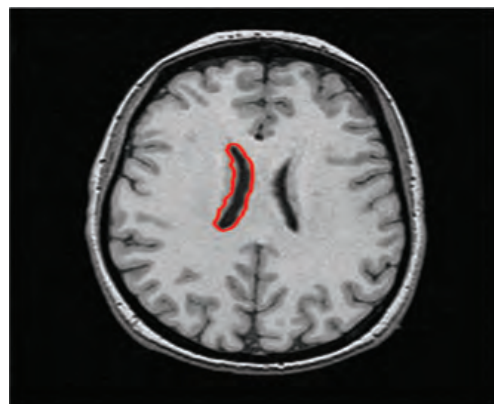
(a) *Modello deterministico*



(b) *Modello parametrico*



(c) *Modello parametrico combinato*



(d) *Modello non parametrico*

Figura 3.14: Snake applicato al cervello - inizializzazione esterna al ventricolo sinistro.

3.3.3 Materia grigia intorno ai ventricoli

Si è effettuato un ulteriore test, in cui si è provato a segmentare i due ventricoli comprendendo anche la materia grigia intorno ad essi. Per i modelli probabilistici sono stati forniti gli stessi parametri scelti per l'intera immagine del cervello. Con il modello non parametrico sono stati ottenuti 4 cluster.

In Figura 3.15 si vede come i potenziali probabilistici, al contrario della $P_{deterministica}$, riescano a cogliere in modo preciso tutta la materia grigia.

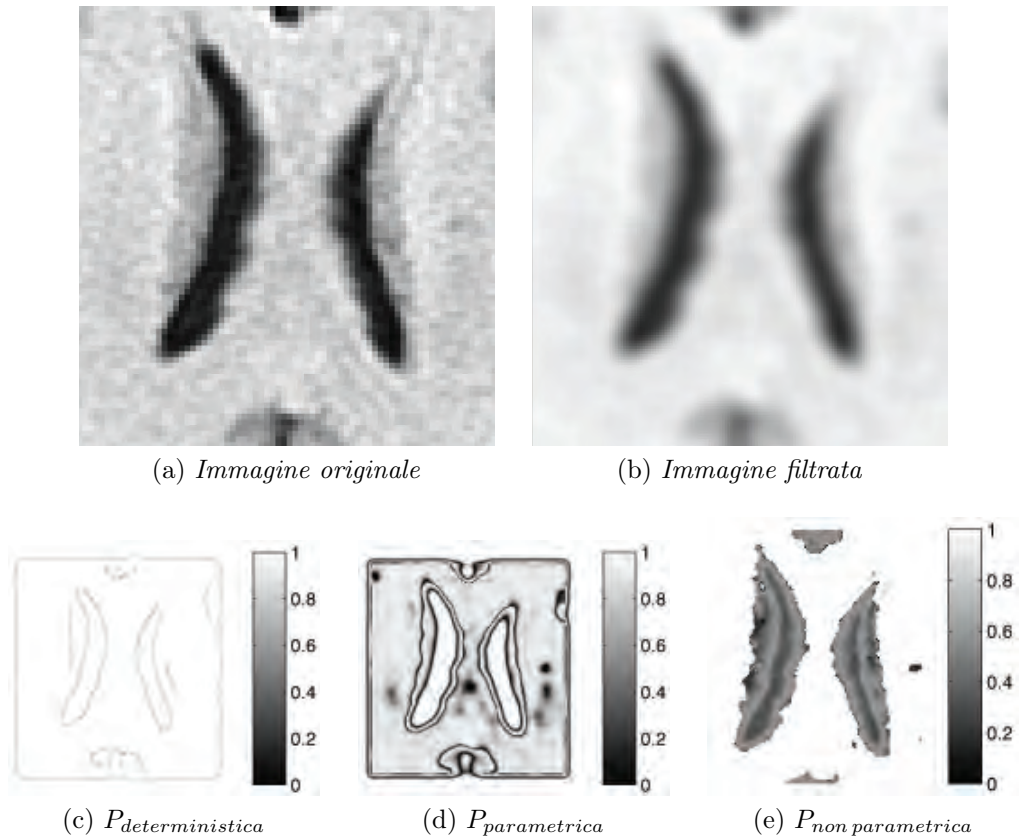


Figura 3.15: Ventricoli del cervello - confronto di $P(x, y)$ per ogni modello.

La Figura 3.16 mostra il confronto tra i campi di forze. Il campo di forze non parametrico (Figura 3.16d) sembra più confuso rispetto a quello parametrico, ma come si vedrà porta ad una segmentazione più dettagliata perché è stato calcolato dall'immagine originale. La forza combinata in Figura 3.16c è stata calcolata scegliendo i pesi $w_1 = 0.7$ per la $F_{probabilistica}$ e $w_2 = 0.3$ per la $F_{deterministica}$. Tale campo di forze è molto forte sui contorni dei ventricoli, ma non sulla materia grigia circostante, in quanto il contributo della

forza deterministica ne peggiora i bordi; per questo motivo si decide di non utilizzare la forza combinata.

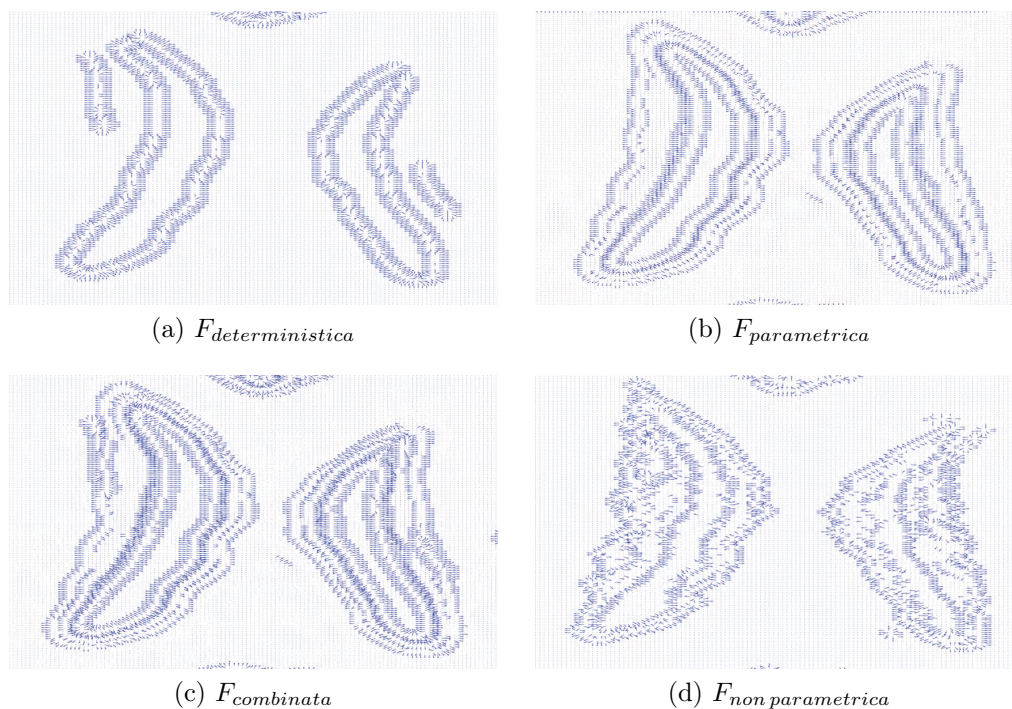


Figura 3.16: Ventricoli del cervello - confronto dei campi di forze.

Materia grigia intorno al ventricolo destro

L'inizializzazione è stata fatta, come per il ventricolo, esternamente all'oggetto. Sono stati tenuti i valori per α e β uguali a prima; per il modello parametrico si sono scelti $k_1 = 0.5$, $k_2 = -0.1$ e $N = 400$ e per il non parametrico $k_1 = 0.5$, $k_2 = -0.1$ e $N = 230$. Questi valori sono riportati in Tabella 3.7. Confrontando tali parametri con quelli utilizzati per individuare i contorni del ventricolo, si nota che per entrambi i metodi è stato aumentato il peso relativo alla forza esterna e diminuito quello della pressione interna. Questa scelta è giustificata dal fatto che il campo di forze in oggetto è più debole rispetto a quello del ventricolo, quindi se si utilizzassero i parametri di prima si segmenterebbe di nuovo il ventricolo senza la materia grigia intorno.

In Figura 3.17 si presentano i risultati della segmentazione. Si vede molto bene in questo caso come il modello non parametrico porti a una segmentazione molto precisa, anche se già l'approccio parametrico dà una buona approssimazione dei contorni.

Modello	α	β	k_1	k_2	N
parametrico	0.5	0.5	0.5	-0.1	400
non parametrico	0.5	0.5	0.5	-0.1	230

Tabella 3.7: Parametri dello snake applicato alla materia grigia intorno al ventricolo destro del cervello.



(a) *Modello parametrico*



(b) *Modello non parametrico*

Figura 3.17: Snake applicato al cervello - segmentazione della materia grigia intorno al ventricolo destro.

Materia grigia intorno al ventricolo sinistro

Anche in questo caso sono stati tenuti i valori per α e β uguali a prima; seguendo lo stesso ragionamento fatto per il ventricolo destro, è stato aumentato il peso della forza esterna e diminuito quello per la pressione interna: per il modello parametrico si sono scelti $k_1 = 0.5$, $k_2 = -0.1$ e $N = 360$, per il non parametrico $k_1 = 0.5$, $k_2 = -0.1$ e $N = 250$. Si riassumono i parametri in Tabella 3.8.

Come nel caso del ventricolo destro, non è stato utilizzato il modello combinato. Anche in questo caso il modello non parametrico fornisce una segmentazione migliore del parametrico (Figura 3.18).

Modello	α	β	k_1	k_2	N
parametrico	0.5	0.5	0.5	-0.1	360
non parametrico	0.5	0.5	0.5	-0.1	250

Tabella 3.8: Parametri dello snake applicato alla materia grigia intorno al ventricolo sinistro del cervello.

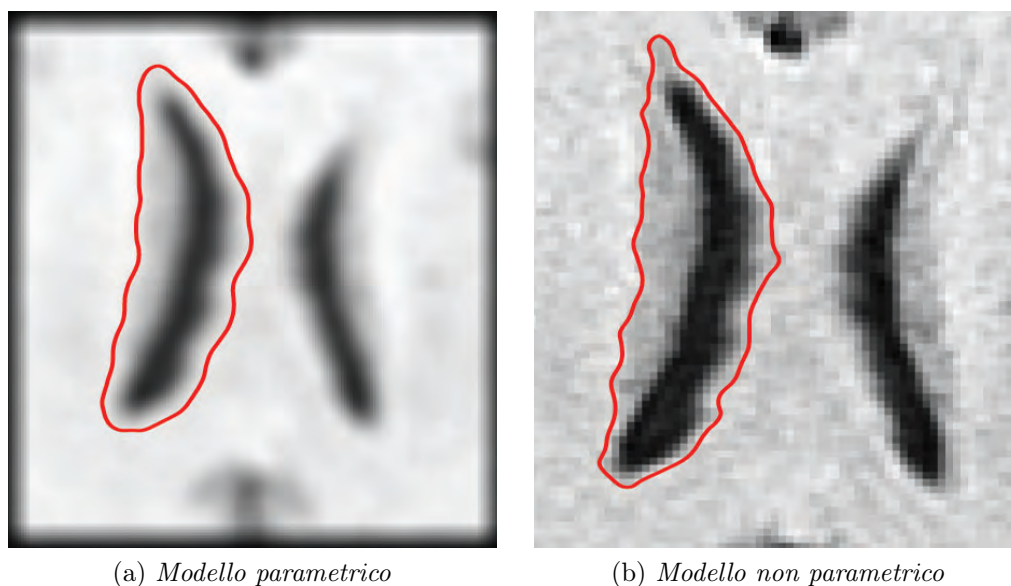


Figura 3.18: Snake applicato al cervello - segmentazione della materia grigia intorno al ventricolo sinistro.

3.3.4 Materia grigia

Per la segmentazione della materia grigia, si sono scelti $\alpha = 0.3$ e $\beta = 0.7$. Per muovere lo snake sono stati assegnati $k_1 = 0.5$, $k_2 = -0.2$ e $N = 1500$ per il modello deterministico, $k_1 = 0.3$, $k_2 = -0.2$ e $N = 1500$ per il parametrico, $k_1 = 0.3$, $k_2 = -0.2$ e $N = 1300$ per il modello combinato, $k_1 = 0.6$, $k_2 = -0.15$ e $N = 2500$ per il non parametrico. I valori sono riassunti in Tabella 3.9.

Si nota in Figura 3.19 come i risultati del modello parametrico e del modello combinato siano molto più precisi rispetto a quello deterministico. Per quanto riguarda il modello non parametrico, le forze rimangono bloccate sulla corona più esterna del cervello, quindi la segmentazione è imprecisa.

Modello	α	β	k_1	k_2	N
deterministico	0.3	0.7	0.5	-0.2	1500
parametrico	0.3	0.7	0.3	-0.2	1500
parametrico combinato	0.3	0.7	0.3	-0.2	1300
non parametrico	0.3	0.7	0.6	-0.15	2500

Tabella 3.9: Parametri dello snake applicato alla materia grigia.

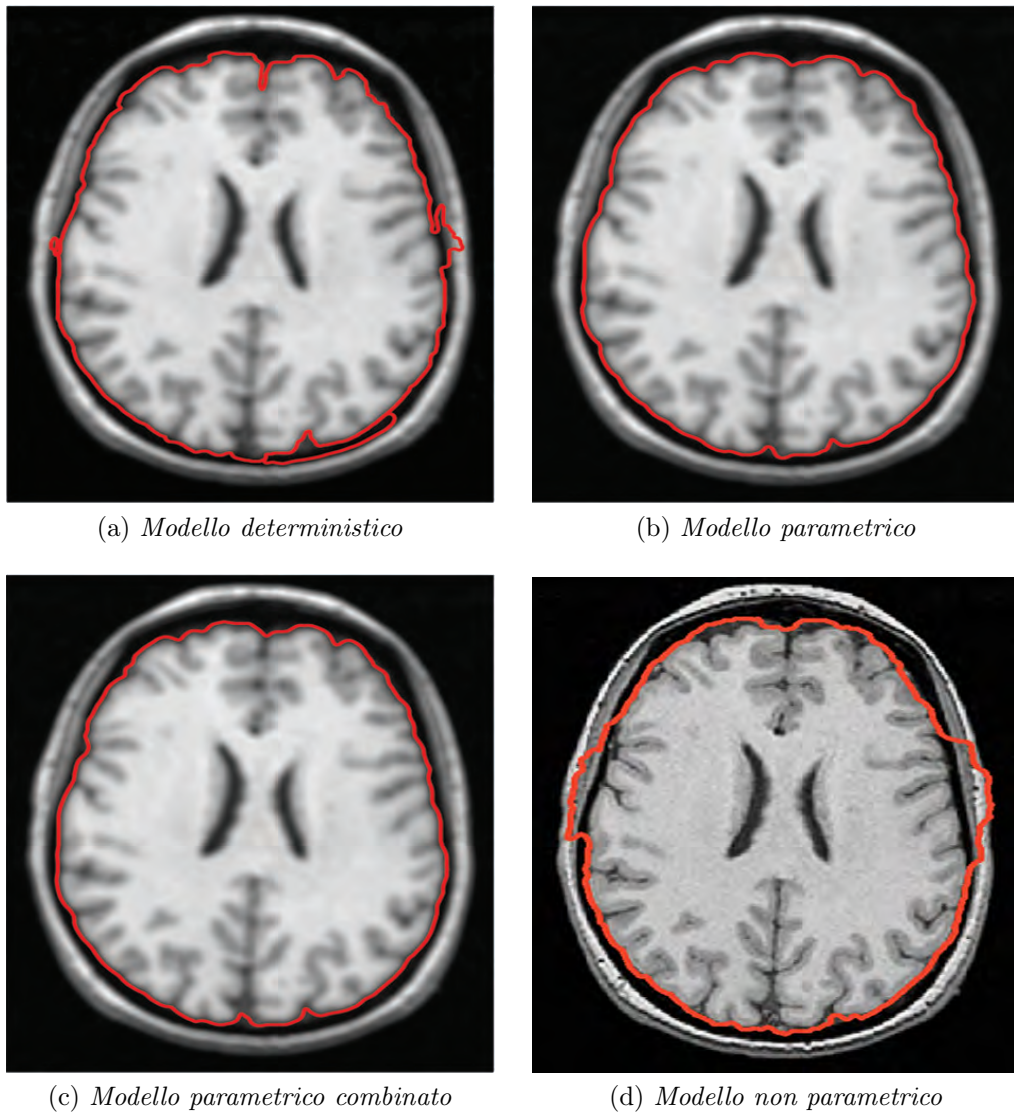


Figura 3.19: Snake applicato al cervello per la segmentazione della materia grigia.

3.3.5 Materia bianca

Per quanto riguarda la materia bianca, si osserva già dalla Figura 3.12 che viene identificata meglio mediante il modello parametrico. Si vedrà infatti che lo snake, utilizzando gli altri metodi, non riuscirà a bloccarsi sui contorni della materia bianca.

I primi test sono stati effettuati inizializzando lo snake all'interno dell'oggetto d'interesse. Tale decisione ha però portato a dei risultati insoddisfacenti per tutti i metodi. Per superare il problema si è allora scelto di modificare i parametri dei modelli probabilistici per fare in modo che il contorno della materia grigia non venisse identificato. Per il modello parametrico si sono forniti a priori 3 cluster e non più 4 come nei casi precedenti. Come si vede dalla Figura 3.20, infatti, con 3 classi è possibile posizionare lo snake all'esterno ed evitare che venga bloccato dalla materia grigia. Per muovere lo

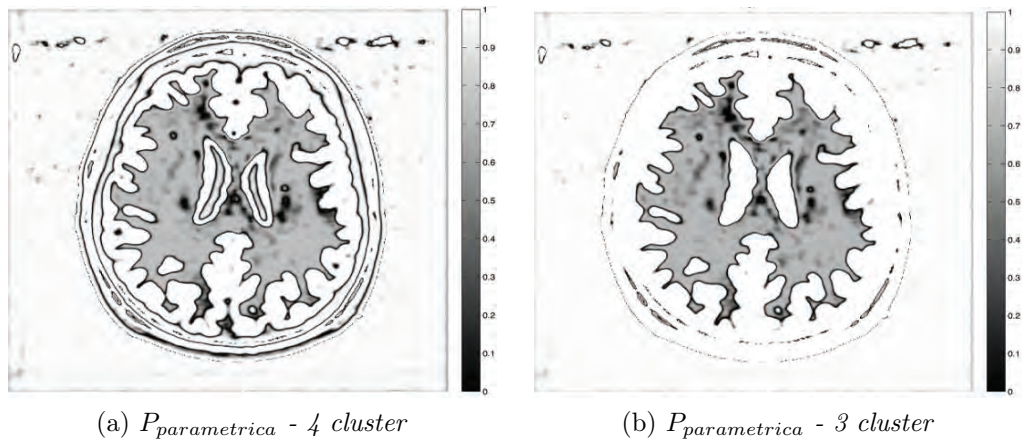


Figura 3.20: Potenziale $P(x,y)$ del cervello ottenuto mediante il modello parametrico al variare del numero di cluster.

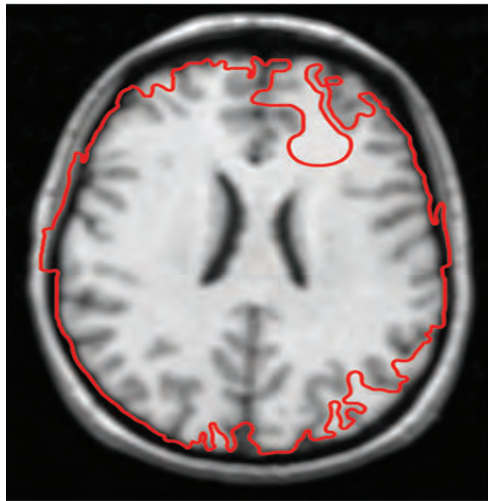
snake si sono scelti i valori $k_1 = 0.6$ $k_2 = -0.1$ $N = 2000$ per il modello deterministico, $k_1 = 0.6$ $k_2 = -0.1$ $N = 2500$ per il parametrico, $k_1 = 0.6$ $k_2 = -0.1$ $N = 2500$ per il combinato, $k_1 = 0.8$ $k_2 = -0.1$ $N = 5000$ per il non parametrico. I valori α e β sono stati fissati rispettivamente pari a 0.3 e 0.7, come nel caso della materia grigia. Tali valori sono riassunti in Tabella 3.10.

Come si era previsto, il modello deterministico fallisce completamente nella segmentazione della materia bianca (Figura 3.21a), mentre l'approccio parametrico ne individua i bordi in modo quasi perfetto, come si può osservare in Figura 3.21b. Il modello combinato è troppo influenzato da quello deterministico sebbene sia stato dato più peso alla forza probabilistica, di

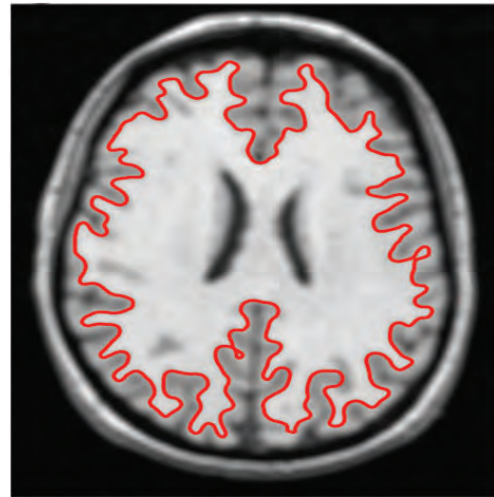
Modello	α	β	k_1	k_2	N
deterministico	0.3	0.7	0.6	-0.1	2000
parametrico	0.3	0.7	0.6	-0.1	2500
parametrico combinato	0.3	0.7	0.6	-0.1	2500
non parametrico	0.3	0.7	0.8	-0.1	5000

Tabella 3.10: Parametri dello snake applicato alla materia bianca.

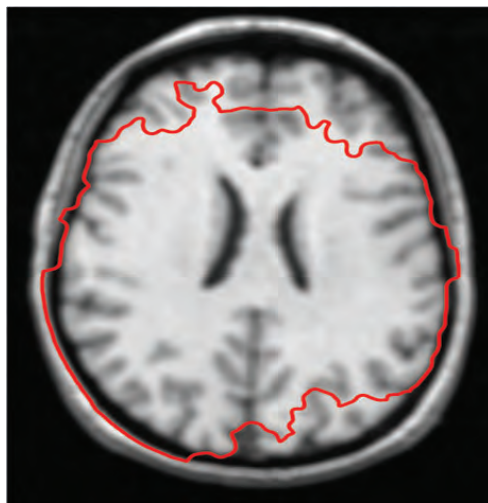
conseguenza anch'esso fallisce (Figura 3.21c). Per quanto riguarda il modello non parametrico, lo snake si ferma sul contorno esterno della materia grigia e non riesce a raggiungere i bordi della materia bianca.



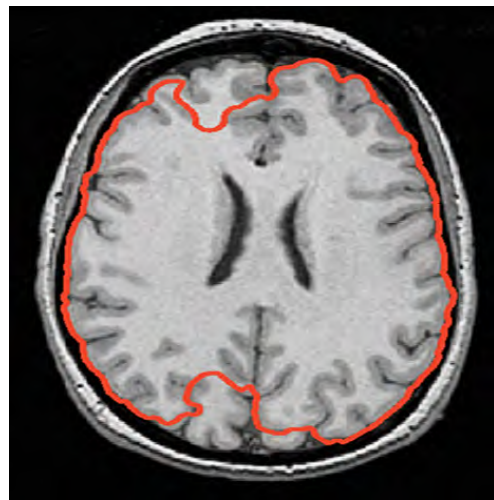
(a) *Modello deterministico*



(b) *Modello parametrico*



(c) *Modello parametrico combinato*



(d) *Modello non parametrico*

Figura 3.21: Snake applicato al cervello per la segmentazione della materia bianca.

3.4 Ventricolo

L'ultima immagine analizzata è una scansione in risonanza magnetica cardiaca. L'obiettivo è riuscire a segmentare il ventricolo sinistro e il miocardio. Per quanto riguarda il ventricolo sinistro, la segmentazione viene effettuata senza problemi. La maggiore difficoltà si riscontra nell'individuazione del contorno esterno del muscolo cardiaco.

3.4.1 Ventricolo sinistro

L'immagine filtrata (Figura 3.22b) è ricavata applicando un filtro gaussiano di varianza $\sigma^2 = 100$. Per il modello parametrico sono state date a priori 3 classi, mentre per quello non parametrico sono stati forniti i valori $m = 10^{-1}$, $\lambda = 4$ e sono state effettuate 200 iterazioni, ottenendo 6 classi.

Come si vede già dalla Figura 3.22, tutti i metodi individuano il contorno del ventricolo sinistro.

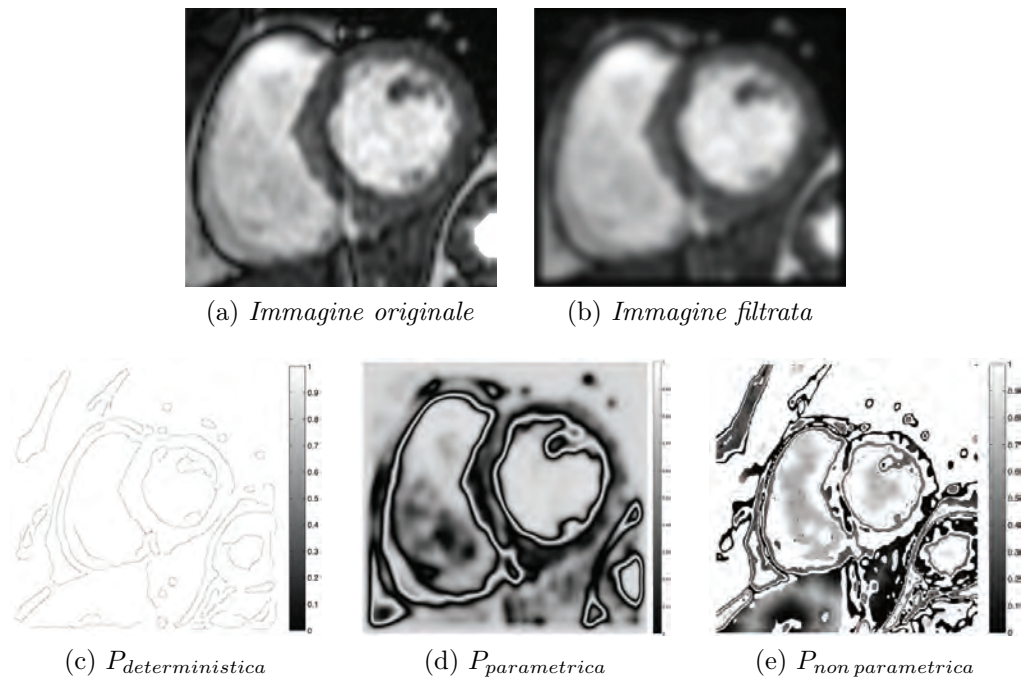


Figura 3.22: Ventricolo cardiaco - confronto di $P(x, y)$ per ogni modello.

Si nota in Figura 3.23 che con il modello non parametrico i bordi sono più dettagliati rispetto agli altri metodi. Per il modello combinato si è scelto di dare più peso al modello parametrico, in quanto è più preciso ($w_1 = 0.8$ e $w_2 = 0.2$).

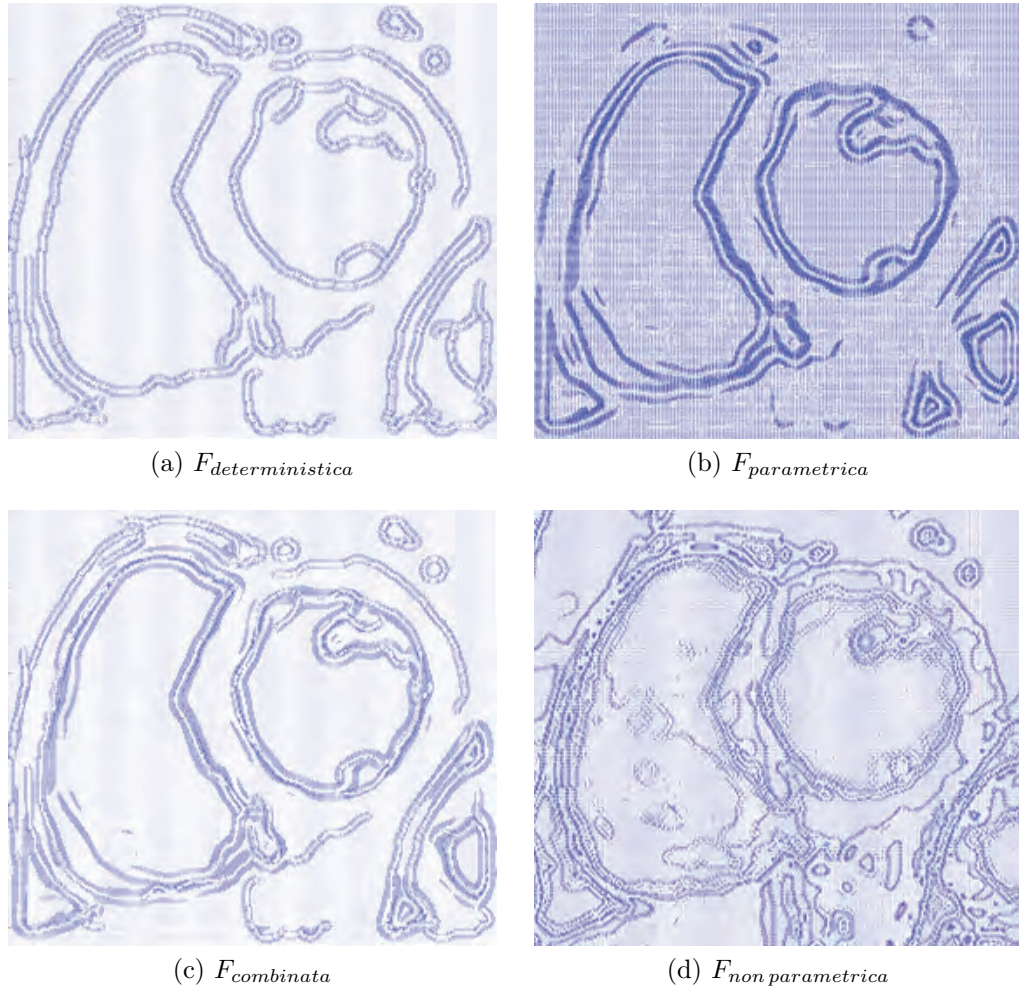


Figura 3.23: Ventricolo cardiaco - confronto dei campi di forze.

Per muovere lo snake sono stati scelti i valori $\alpha = 0.5$ e $\beta = 0.5$. Per il modello deterministico si sono fissati $k_1 = 0.3$, $k_2 = 0.3$ e $N = 1500$, per il parametrico $k_1 = 0.3$, $k_2 = 0.2$ e $N = 1500$, per il combinato $k_1 = 0.4$, $k_2 = 0.2$, e $N = 1500$, per il non parametrico $k_1 = 0.3$, $k_2 = 0.15$ e $N = 2000$. I valori dei parametri si riassumono in Tabella 3.11.

I modelli probabilistici, con i valori dei pesi k_1 e k_2 detti sopra, riescono ad inseguire perfettamente il contorno escludendo il muscolo papillare, in modo particolare l'approccio non parametrico permette una maggior precisione (Figura 3.24d), sempre grazie al fatto che la funzione $P(x, y)$ è calcolata dall'immagine originale e non dalla filtrata. L'approccio deterministico invece con questa scelta dei parametri supera il contorno del ventricolo; se si abbas-

Modello	α	β	k_1	k_2	N
deterministico	0.5	0.5	0.3	0.3	1500
parametrico	0.5	0.5	0.3	0.2	1500
parametrico combinato	0.5	0.5	0.4	0.2	1500
non parametrico	0.5	0.5	0.3	0.15	2000

Tabella 3.11: Parametri dello snake applicato al ventricolo cardiaco sinistro.

sasse il peso della pressione interna non riuscirebbe comunque a segmentare con precisione il contorno dell'oggetto perché verrebbe bloccato dai bordi del muscolo papillare, come vedremo successivamente.

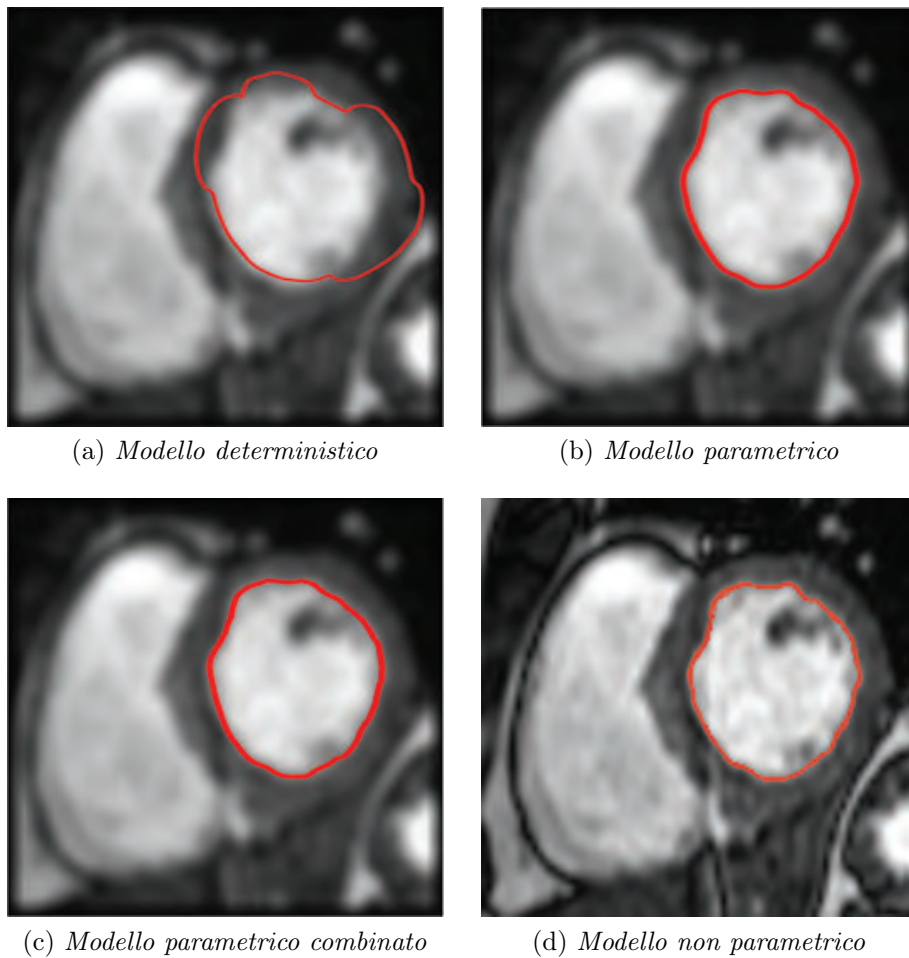


Figura 3.24: Snake applicato al ventricolo cardiaco sinistro.

Modello	α	β	k_1	k_2	N
deterministico	0.5	0.5	0.3	0.2	1500
parametrico	0.5	0.5	0.5	0.1	1500
parametrico combinato	0.5	0.5	0.5	0.12	1500
non parametrico	0.5	0.5	0.5	0.1	2000

Tabella 3.12: Parametri dello snake applicato al ventricolo cardiaco sinistro comprendendo il muscolo papillare.

Il vantaggio dei metodi probabilistici sta nel fatto che se si volesse segmentare anche il muscolo papillare, basta fornire un peso più forte per la forza esterna e uno minore per la pressione interna. Per il modello deterministico si sono assegnati $k_1 = 0.3$, $k_2 = 0.2$ e $N = 1500$, per il parametrico $k_1 = 0.5$, $k_2 = 0.1$ e $N = 1500$, per il combinato $k_1 = 0.5$, $k_2 = 0.12$ e $N = 1500$, per il modello non parametrico $k_1 = 0.5$, $k_2 = 0.1$ e $N = 2000$. In Tabella 3.12 si riassumono i valori dei parametri. I risultati della segmentazione sono mostrati in Figura 3.25.

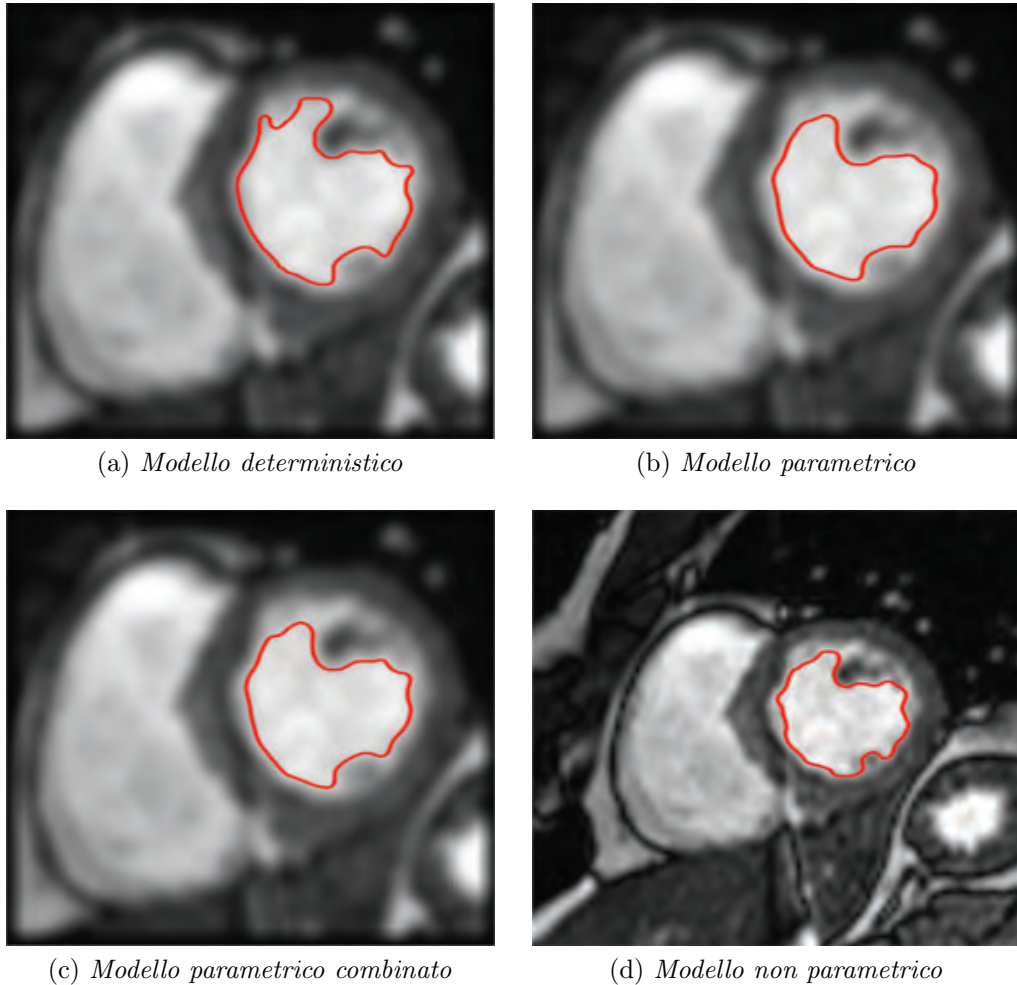


Figura 3.25: Snake applicato al ventricolo cardiaco sinistro - segmentazione comprendendo i muscoli capillari.

3.4.2 Miocardio

Come si vede dalla Figura 3.22a, il livello di grigi a destra del muscolo cardiaco è molto simile alla zona adiacente, di conseguenza non si riesce bene ad identificare il contorno dell'oggetto. Per questo motivo, prima di applicare qualsiasi metodo si equalizza l'immagine per cercare di aumentare il contrasto sul bordo (Figura 3.26). Successivamente per il modello parametrico l'immagine equalizzata viene modificata mediante il filtro gaussiano di varianza $\sigma^2 = 9$. La varianza è più bassa rispetto a prima, per non smussare eccessivamente i contorni.

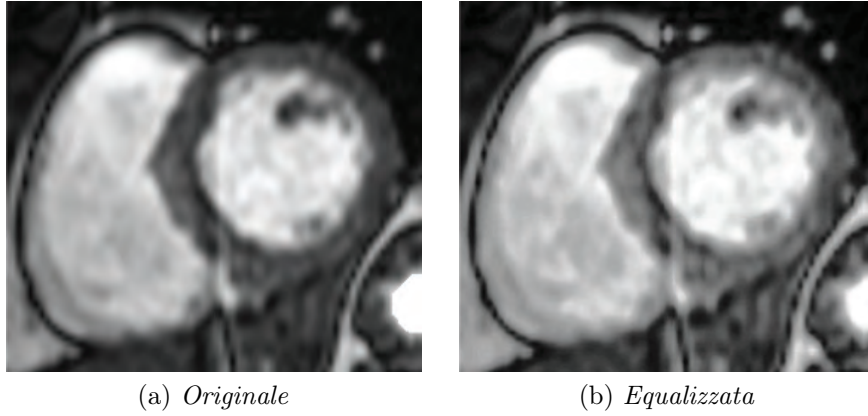


Figura 3.26: Equalizzazione dell'immagine di un ventricolo. Nell'immagine equalizzata i contorni del setto inter-ventricolare sono più visibili rispetto all'originale.

Il numero di classi a priori per il modello parametrico è pari a 6, mentre per il modello non parametrico si dà il valore $m = 10^{-3}$, mentre λ e il numero di iterazioni sono rispettivamente 4 e 200, come nel caso precedente; mediante il modello non parametrico si sono ottenute 4 classi. Per quanto riguarda il modello combinato, sono stati scelti rispettivamente per $F_{probabilistica}$ e $F_{deterministica}$ i pesi $w_1 = 0.6$ e $w_2 = 0.4$.

Dalla Figura 3.27 e dai campi di forze (Figura 3.28) si può osservare come i contorni del miocardio siano più visibili nel modello parametrico combinato e nel modello non parametrico, quindi si suppone che questi ultimi due approcci siano più efficienti.

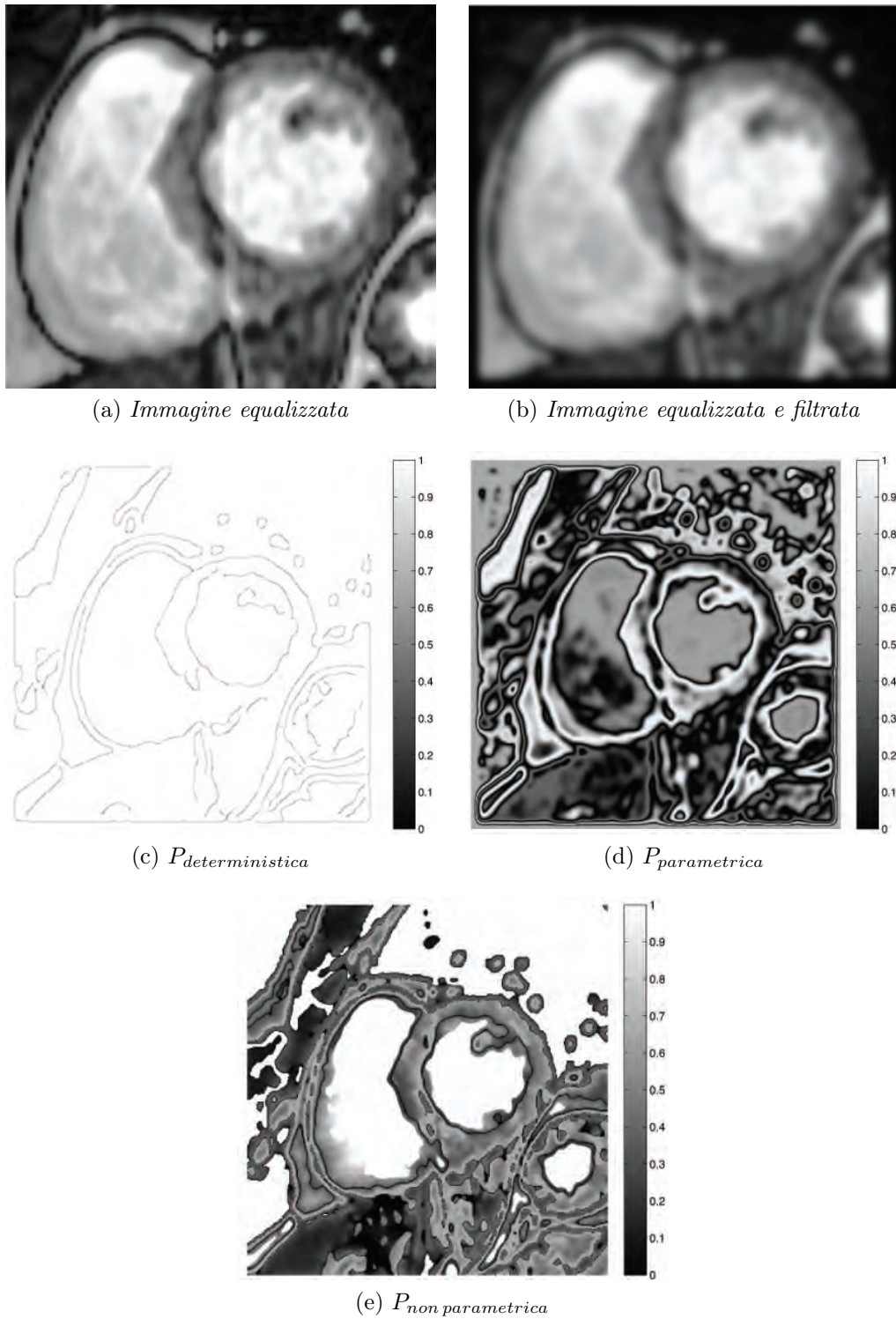


Figura 3.27: Ventricolo cardiaco - confronto di $P(x, y)$ dell'immagine equalizzata per ogni modello.

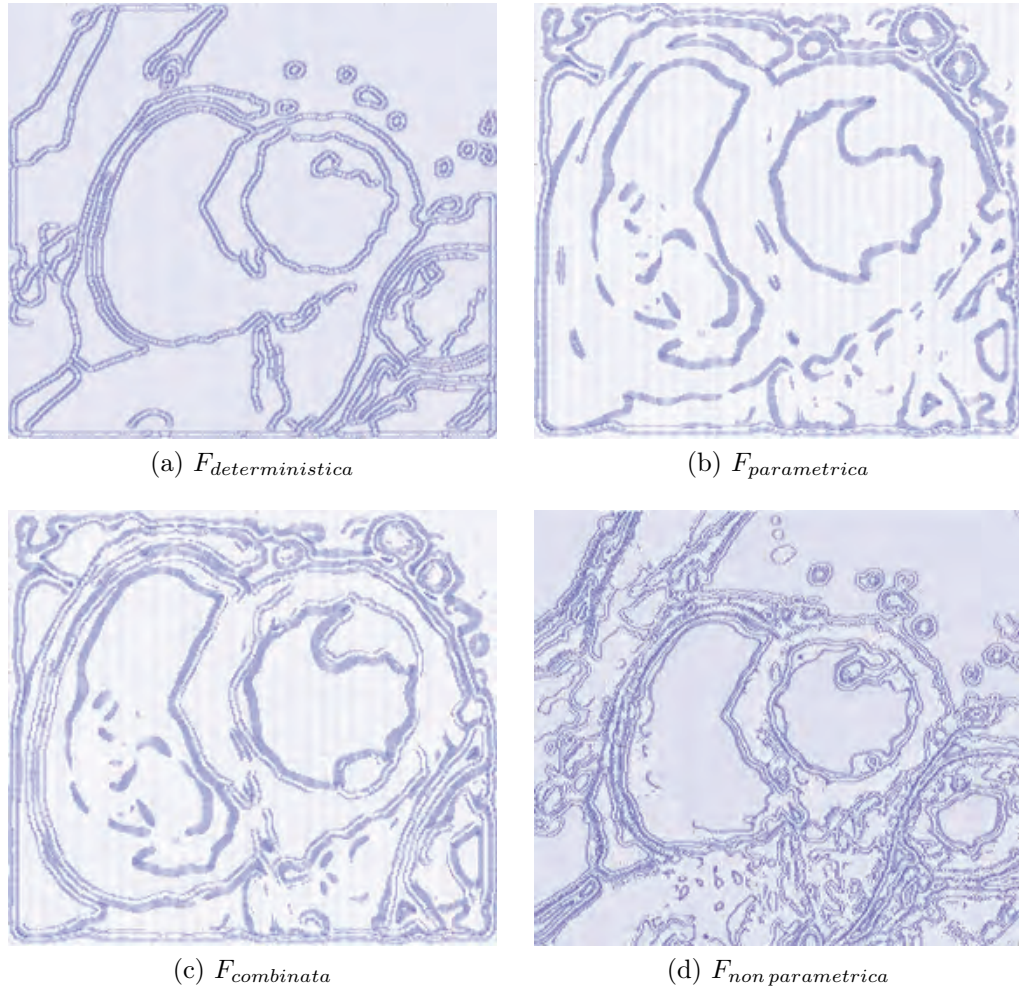


Figura 3.28: Ventricolo cardiaco - confronto dei campi di forze ricavati dall'immagine equalizzata.

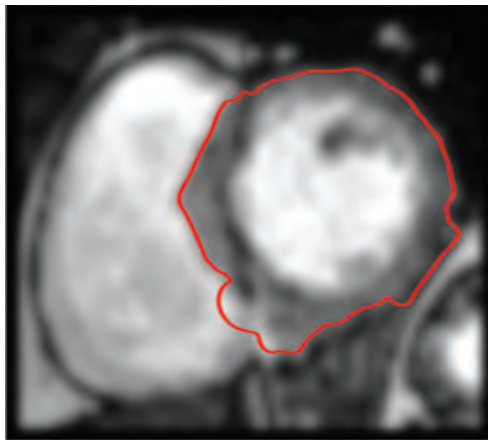
Per segmentare l'oggetto sono stati scelti i valori per $\alpha = 0.5$ e $\beta = 0.3$. Per il modello deterministico si sono forniti i parametri $k_1 = 0.4$, $k_2 = 0.1$ e $N = 1000$, per il parametrico $k_1 = 0.5$, $k_2 = 0.1$ e $N = 1500$, per il combinato $k_1 = 0.6$, $k_2 = 0.1$ e $N = 1500$, per il non parametrico $k_1 = 0.6$, $k_2 = 0.1$ e $N = 1500$. Tali valori sono sintetizzati in Tabella 3.13. Il peso k_2 è sempre molto basso perché il campo di forze è molto debole, quindi la forza esterna, a cui invece viene dato un peso maggiore, rischierebbe di non riuscire a controbilanciare la pressione interna.

In Figura 3.29 si nota come il modello parametrico da solo non riesca ad individuare il contorno del miocardio, mentre già con la forza combinata si ha un miglioramento. Anche in questo caso però non si riesce a segmentare

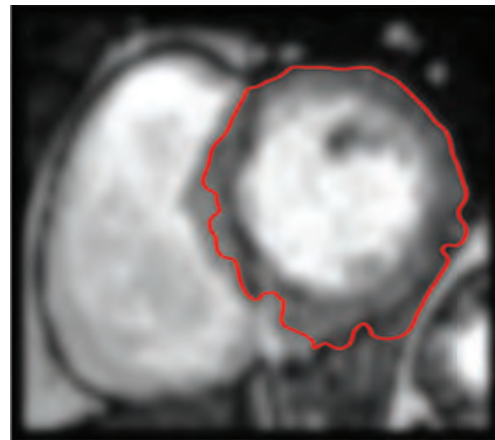
Modello	α	β	k_1	k_2	N
deterministico	0.5	0.3	0.4	0.1	1000
parametrico	0.5	0.3	0.5	0.1	1500
parametrico combinato	0.5	0.3	0.6	0.1	1500
non parametrico	0.5	0.3	0.6	0.1	1500

Tabella 3.13: Parametri dello snake applicato al miocardio.

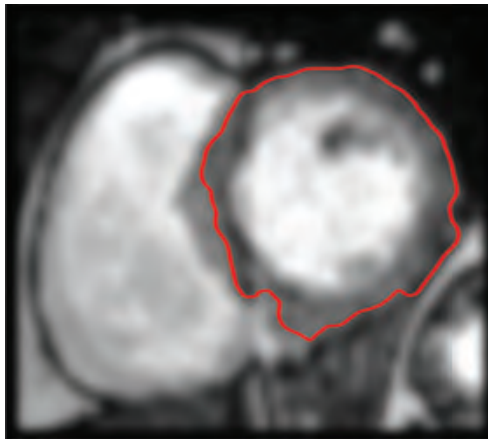
in modo precisissimo il bordo; inoltre, a causa dell'aumento del numero di classi fornito a priori, l'immagine non viene clusterizzata sempre allo stesso modo, di conseguenza i risultati sono molto variabili. Molte volte il contorno in basso non viene nemmeno identificato, perciò prima di utilizzare questo metodo si deve prestare molta attenzione al campo di forze che si crea. Si ricorre allora al metodo non parametrico, che porta ad un risultato molto più preciso rispetto agli altri.



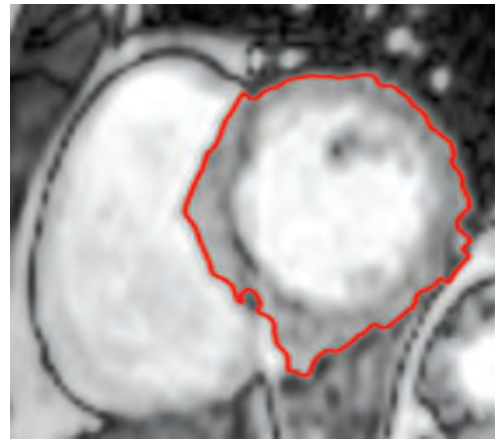
(a) *Modello deterministico*



(b) *Modello parametrico*



(c) *Modello parametrico combinato*



(d) *Modello non parametrico*

Figura 3.29: Snake applicato al miocardio.

Capitolo 4

Conclusioni

In questo lavoro di tesi si è utilizzato il metodo ai contorni attivi parametrici per la segmentazione di immagini mediche, incorporando delle forze di tipo probabilistico. I metodi deterministici classicamente utilizzati funzionano bene se il contorno è ben definito, cioè se i bordi delle immagini (e quindi i gradienti) sono forti. Nei casi in cui i bordi non sono invece ben evidenti, tali metodi non portano ad una segmentazione corretta degli oggetti, di conseguenza bisogna ricorrere ad altre tecniche. Come si è visto nei risultati mostrati nel Capitolo 3, l'inclusione di forze di tipo probabilistico nella segmentazione mediante i contorni attivi parametrici porta a notevoli miglioramenti nella segmentazione, in modo particolare nei casi in cui i contorni degli oggetti non sono ben definiti o le immagini presentano poco contrasto.

Si è notato che i modelli probabilistici permettono una maggiore flessibilità nel scegliere le zone da estrarre (si vedano per esempio i risultati sul miocardio presentati nel Sottoparagrafo 3.4.2). Inoltre essi riescono a identificare bene anche le zone che non sono caratterizzate da un forte gradiente sui bordi.

Il modello parametrico necessita di definire a priori un numero di classi in cui suddividere l'immagine. Ciò significa che prima di applicarlo è necessario osservare bene l'immagine per decidere un numero di cluster adatto. A volte questo può però rivelarsi un vantaggio, come ad esempio per l'immagine del cervello, dove ha permesso di ottenere il contorno della materia bianca che altrimenti non si sarebbe riuscito ad estrarre in alcun modo (si veda il Sottoparagrafo 3.3.5). Questo modello si è dimostrato efficace anche nell'identificare la materia grigia del cervello.

Il difetto di questo approccio rimane quello del filtraggio dell'immagine: senza un'opportuna maschera di convoluzione, infatti, la segmentazione fallisce completamente a causa del rumore presente nelle immagini mediche. Inoltre,

filtrando le immagini, i bordi vengono “smussati”; questo significa che vengono individuati tali contorni e non quelli originali. Un'altra difficoltà che si è presentata è stato il basso contrasto presente a volte in questo tipo di immagini. Provando ad aumentarne il contrasto e quindi risaltandone i bordi, si è ottenuto un notevole miglioramento dei risultati, ma la segmentazione rimane comunque imprecisa.

La novità introdotta dalla forza combinata migliora leggermente la segmentazione rispetto al modello parametrico, velocizzando e rendendo un po' più precisa l'individuazione dei contorni. Se però questi ultimi non sono identificati né dal modello deterministico né da quello parametrico, la forza combinata non porta ad alcun perfezionamento.

Il modello bayesiano non parametrico ha il vantaggio di non dover definire a priori un numero di classi. Inoltre, come si è detto prima, tutti gli altri metodi necessitano di pre-processare l'immagine mediante opportuni metodi di filtraggio, per cui bisogna decidere dei parametri adatti in modo da non smussare eccessivamente i bordi dell'immagine. Per il modello non parametrico, invece, questa operazione non è necessaria, in quanto esso lavora direttamente sui dati dell'immagine originale.

Tale modello riesce ad individuare meglio anche i contorni meno definiti, come per esempio quelli del miocardio, dove anche il modello parametrico si è rivelato poco efficiente. L'unica difficoltà riscontrata è stata nella segmentazione della materia bianca, dove il modello parametrico sembra funzionare meglio.

Si penserà di effettuare un'analisi più accurata su altri dataset, utilizzando anche altre tecniche di imaging, per indagare più a fondo sull'efficienza e robustezza di questi approcci proposti. Un ulteriore sviluppo futuro potrebbe essere quello di dare come informazione a priori per l'approccio non parametrico non tanto l'informazione ai bordi fornita dal modello deterministico, quanto quella fornita dal modello parametrico. In questo modo si potrebbero combinare insieme le potenzialità di questi due modelli probabilistici.

Bibliografia

- Abd-Almageed, Wael, Christopher E. Smith e Samah Ramadan (2003). «Kernel Snakes: Non-parametric Active Contour Models». In: *IEEE International Conference on Systems, Man and Cybernetics*, pp. 240–244.
- Anders, Katharina et al. (2012). «Accuracy of automated software-guided detection of significant coronary artery stenosis by CT angiography: comparison with invasive catheterisation». In: *European Radiology*, pp. 1218–1225.
- Caselles, Vicent, Ron Kimmel e Guillermo Sapiro (1995). «Geodesic active contours». In: *Proceedings of IEEE International Conference on Computer Vision*, pp. 61–79.
- Cheng, Kuo-Sheng, Jzau-Sheng Lin e Chi-Wu Mao (1996). «The application of competitive Hopfield neural network to medical image segmentation». In: *IEEE Transactions on Medical Imaging* 15.4, pp. 560–567.
- Cheng, Z. e X. Mei (2008). «MRI head space-based segmentation for object based volume visualization». In: *International Conference on Computer Science and Information Technology*.
- Cohen, Laurent D. e Isaac Cohen (1993). «Finite element methods for active contour models and balloons for 2D and 3D images». In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1131–1147.
- Cramer, Jayashree Kalpathy et al. (2008). «Semi-supervised Segmentation using Non-parametric Snakes for 3D-CT Applications in Radiation Oncology». In: *IEEE Workshop on Machine Learning for Signal Processing*, pp. 109–114.
- Dass, Rajeshwar, Priyanka e Swapna Devi (2012). «Image segmentation techniques». In: *IJECT Vol. 3, Issue 1*, pp. 66–70.
- Farnoosh, Rahman e Behnam Zarpak (2008). «Image segmentation using Gaussian Mixture Model». In: *International Journal of Engineering Science* 19, pp. 29–32.
- Gonzalez, Rafael C. e Richard E. Woods (2008). «Digital Image Processing». In: Prentice Hall.

- Hohne, Karl Heinz e Ralph Bernstein (1986). «Shading 3D-Images from CT Using Gray-Level Gradients». In: *IEE Transactions on Medical Imaging* 5.1, pp. 45–47.
- Jain, Ramesh, Rangachar Kasturi e Brian G. Schunck (1995). «Machine Vision». In: McGraw-Hill. Cap. Image Filtering.
- Kass, Michael, Andrew Witkin e Demetri Terzopoulos (1988). «Snakes: Active Contour Models». In: *International Journal of Computer Vision*, 321-331, pp. 321–331.
- Kim, Junmo et al. (2002). «Nonparametric Methods for Image Segmentation». In: *Proceedings of the 2002 IEE International Conference on Image Processing*.
- MacEachern, Steven N. e Peter Muller (1998). «Estimating Mixture of Dirichlet Process Models». In: *Journal of Computational and Graphical Statistics* 7.2, pp. 223–238.
- Mory, Benoit, Roberto Ardon e Jean-Philippe Thiran (2007). «Variational Segmentation using Fuzzy Region Competition and Local Non-Parametric Probability Density Functions». In: *IEEE 11th International Conference on Computer Vision*, pp. 1–8.
- Nguyen, Thanh Minh e Q. M. Jonathan Wu (2011). «Dirichlet Gaussian mixture model: Application to image segmentation». In: *Image and Vision Computing* 29, pp. 818–828.
- Orbanz, Peter e Joachim M. Buhmann (2008). «Nonparametric Bayesian Image Segmentation». In: *International Journal of Computer Vision* 77, pp. 25–45.
- Ozertem, Umut e Deniz Erdogmus (2007). «Nonparametric Snakes». In: *IEE Transactions on Image Processing* 16.9, pp. 2361–2368.
- Quarteroni, Alfio, Riccardo Sacco e Fausto Saleri (2008). *Matematica numerica*. Springer.
- Rakesh, M. e T. Ravi (2012). «Image segmentation and detection of tumor objects in MR brain images using Fuzzy C-means algorithm». In: *International Journal of Engineering Research and Applications* 2, pp. 2088–2094.
- Rangayyan, Rangaraj M. (2008). «Handbook of Medical Image Processing and Analysis». In: a cura di Isaac N. Bankman. Academic Press. Cap. 1, pp. 69–195.
- Reddick, W.E. et al. (1997). *Automated segmentation and classification of multispectral magnetic resonance images of brain using artificial neural networks*, pp. 911–918.
- Reynolds, Douglas (2008). «Encyclopedia of Biometric Recognition». In: Springer. Cap. Gaussian Mixture Models, pp. 659–663.

- Schaarp, Michiel et al. (2011). «Robust shape regression for supervised vessel segmentation and its application to coronary segmentation in CTA». In: *IEEE Transactions on Medical Imaging*, pp. 1974–1986.
- Terzopoulos, Demetri e Kurt Fleischer (1988). «Deformable models». In: *The Visual Computer* 4, pp. 205–214.
- Tsechpenakis, Gavriil (2011). «Multi Modality State-of-the-Art Medical Image Segmentation and Registration Methodologies». In: vol. 1. Springer. Cap. Deformable Model-Based Medical Image Segmentation, pp. 33–67.
- Yang, Yan, Allen Tannenbaum e Don Giddens (2004). «Knowledge-Based 3D Segmentation and Reconstruction of Coronary Arteries Using CT Images». In: *Proceedings of the 26th Annual International Conference of the IEEE EMBS - San Francisco, CA, USA*, pp. 1664–1666.
- Yang, Yan et al. (2007). «Automatic segmentation of coronary arteries using bayesian driven implicit surfaces». In: *International Symposium on Biomedical Imaging*, pp. 189–192.
- Zhang, Yu-Jin (2006). «Advances in image and video segmentation». In: IRM Press.