



POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Biomedica

Implementazione di un algoritmo automatico per il tracking 3D di strutture vascolari ramificate

Relatori:

Prof. Filippo Molinari

Prof.ssa Kristen M. Meiburger

Candidato:

Raffaella d'Arienzo



Luglio 2019

Indice

Introduzione	1
1 Tecniche di estrazione degli alberi vascolari	3
1.1 Tecniche di riconoscimento dei pattern	4
1.1.1 Approcci basati sullo skeleton	4
Skeleton topologici	5
Skeleton geometrici	5
1.1.2 Approcci ridge-based	6
1.1.3 Approcci basati sul region growing	7
1.2 Approcci basati sui modelli deformabili	7
1.2.1 Edge-based	7
Modelli parametrici	8
Modelli geometrici	8
1.2.2 Region-based	8
1.3 Approcci basati sul tracking	9
1.3.1 Tracking basato su modelli	10
1.3.2 Tracking basato sul percorso di costo minimo	11
1.4 Approcci basati sul machine learning	12

1.4.1	Apprendimento non supervisionato	12
1.4.2	Apprendimento supervisionato	13
2	Materiali e Metodi	17
2.1	Modelli simulati	17
2.1.1	Matlab	18
2.1.2	Solidworks	19
2.2	Immagini in vivo	25
2.3	Descrizione dell'algoritmo	27
2.3.1	Inizializzazione automatica del piano	32
2.3.2	Tracking	34
	Identificazione del piano perpendicolare	35
	Gestione delle biforcazioni e inizio-fine vaso	49
2.4	Metriche di validazione	75
2.4.1	Distanza di Hausdorff	75
2.4.2	Caratterizzazione numerica e parametri di tortuosità	78
2.4.3	Valutazione dei punti di biforcazione e punti di fine vaso	82
3	Risultati	85
3.1	Centerline	85
3.2	Distanza di Hausdorff	88
3.3	Caratterizzazione numerica e parametri di tortuosità	97
3.4	Valutazione dei punti di biforcazione e punti di fine vaso	99
	Conclusioni e sviluppi futuri	105
	Ringraziamenti	111

Elenco delle figure

1.1	Esempio di estrazione di centerline in un modello vascolare . . .	4
1.2	Esempio di maschera usata nello skeleton topologico	5
1.3	Skeleton geometrico: calcolo distance transform	6
1.4	Applicazione modelli deformabili geometrici	9
1.5	Tracking: model-based	11
1.6	Tracking: percorso di costo minimo	12
1.7	Apprendimento non supervisionato: schema riassuntivo	13
1.8	Apprendimento supervisionato: schema riassuntivo	14
1.9	Apprendimento supervisionato: reti neurali convoluzionali . .	15
2.1	Modelli simulati in Matlab e Solidworks	20
2.2	Schematizzazione della creazione dei modelli in Matlab	21
2.3	Volumi binari relativi ai modelli simulati	24
2.4	Creazione volumi binari reali: OCT-A	26
2.5	Volumi OCT-A usati per testare l'algoritmo	27
2.6	Tracking basato sulla rotazione del piano	28
2.7	Esempio di identificazione degli oggetti connessi in un volume reale	29

2.8	Scelta oggetto connesso per le due ROI OCT-A	30
2.9	Step generali dell'algoritmo	31
2.10	Inizializzazione piano	32
2.11	Flow chat per l'inizializzazione del piano	33
2.12	Identificazione direzione di tracking al primo step di tracking .	34
2.13	Schematizzazione della costruzione della sfera	35
2.14	Rotazione del piano nel processo di ricerca del piano perpen- dicolare	36
2.15	Scelta direzione di tracking	36
2.16	Step seguiti per identificare il piano localmente ortogonale all'asse del vaso	37
2.17	Traslazione del sistema di riferimento nel centro della sezione e rotazione del piano	38
2.18	Analisi della sezione identificata sulla base del numero di oggetti.	40
2.19	Visualizzazione intersezione piano-volume: controllo numero oggetti	41
2.20	Analisi della sezione identificata sulla base della percentuale di pixel bianchi	42
2.21	Visualizzazione intersezione piano-volume: controllo percen- tuale di pixel bianchi	42
2.22	Analisi della sezione identificata sulla base della posizione del centro	43
2.23	Variazione compactness	44
2.24	Esempio reale di analisi sulle sezioni	45

2.25 Esempio reale di analisi sulle sezioni: numero oggetti e percentuale pixel bianchi	45
2.26 Esempio reale di analisi sulle sezioni: compactness	46
2.27 Esempio di valutazione della compactness	47
2.28 Esempio di valutazione della varianza del raggio	47
2.29 Andamento rapporto tra raggio massimo e raggio minimo . . .	48
2.30 Calcolo raggio della sfera	49
2.31 Costruzione sfera e calcolo intensità media lungo i raggi	51
2.32 Esempio di gamma correction	51
2.33 Gamma correction applicata	52
2.34 Esempio pratico di applicazione di gamma correction	53
2.35 Step applicazione imadjust	54
2.36 Step per l'identificazione dei rami persi	55
2.37 Esempio pratico di perdita di rami	57
2.38 Step totali seguiti per l'identificazione dei rami persi	58
2.39 Step seguiti per calcolare i versori	59
2.40 Classificazione dei punti nell'algoritmo del DBSCAN	60
2.41 Costruzione del grafico KNN	63
2.42 Calcolo versori, costruzione grafico k-distance, clusterizzazione	64
2.43 Calcolo centroide per ogni cluster e costruzione dei piani . . .	65
2.44 Unione cluster	66
2.45 Unione cluster multipli	66
2.46 $Cluster \geq 3$: identificazione di una biforcazione	67
2.47 $Cluster = 2$: vaso singolo	68
2.48 $Cluster = 1$: inizio o fine vaso	69

2.49	Vaso già percorso	70
2.50	Calcolo della distanza tra il punto e i segmenti della centerline	72
2.51	Valutazione della posizione della proiezione del punto	72
2.52	Unione dei vasi: direzione di tracking associata ad ogni end- points	73
2.53	Unione dei vasi: schematizzazione connessione	74
2.54	Calcolo distanza di Hausdorff	76
2.55	Skeleton e Tracking a confronto	77
2.56	Calcolo distanza di Hausdorff per ogni ramo	77
2.57	Calcolo dei parametri di tortuosità	81
3.1	Centerline per i modelli simulati	86
3.2	Centerline per i volumi in vivo	87
3.3	Skeleton-Tracking modello curvo	88
3.4	Skeleton-Tracking modello elica	89
3.5	Skeleton-Tracking modello biforcazioneY	89
3.6	Skeleton-Tracking modello biforcazione	90
3.7	Skeleton-Tracking modello albero semplice	90
3.8	Skeleton-Tracking modello albero complesso	91
3.9	Skeleton-Tracking modello albero profondo	91
3.10	Skeleton-Tracking in-vivo 1	92
3.11	Skeleton-Tracking in-vivo 2	92
3.12	Confronto distanza di Hausdorff col il raggio medio per mo- dello curvo ed elica	93
3.13	Confronto distanza di Hausdorff col il raggio medio per il modello biforcazioneY	93

3.14 Confronto distanza di Hausdorff col il raggio medio per il modello biforcazione	94
3.15 Confronto distanza di Hausdorff col il raggio medio per il modello albero semplice	94
3.16 Confronto distanza di Hausdorff col il raggio medio per il modello albero complesso	95
3.17 Confronto distanza di Hausdorff col il raggio medio per il modello albero profondo	95
3.18 Confronto distanza di Hausdorff col il raggio medio per il modello in-vivo 1	96
3.19 Confronto distanza di Hausdorff col il raggio medio per il modello in-vivo 2	96
3.20 Confronto parametri di tortuosità modello biforcazioneY, cur- vo, biforcazione, elica	98
3.21 Confronto parametri di tortuosità modello albero semplice, albero complesso, albero profondo	98
3.22 Confronto parametri di tortuosità modello in-vivo 1, in-vivo 2	99
3.23 Valutazione branch-points e end-points	100
3.24 Valutazione branch-points e end-points modello biforcazione	101
3.25 Valutazione branch-points e end-points modello albero semplice	101
3.26 Valutazione branch-points e end-points modello albero com- plesso	102
3.27 Valutazione branch-points e end-points modello albero profondo	102
3.28 Valutazione branch-points e end-points mvolume reale 1	103

Elenco delle tabelle

2.1	Dimensioni modello 1 in Solidworks e in Matlab	23
2.2	Dimensioni modello 2 in Solidworks e in Matlab	23
2.3	Dimensioni modello 3 in Solidworks e in Matlab	23
2.4	Scelta della costante moltiplicativa per definire il raggio della sfera	50
3.1	Tempo impiegato per tracciare ciascun volume	87

Introduzione

In questo lavoro di tesi viene presentato un metodo di tracking 3D dei vasi sanguigni con lo scopo di estrarre la centerline da alberi vascolari ramificati. L'algoritmo è stato implementato servendosi di volumi simulati in Matlab e Solidworks e poi testato su immagini in vivo OCTA relative alla vascolarizzazione cutanea. Il tracking è uno degli approcci usati per la segmentazione dei vasi in immagini mediche. L'angiografia a raggi X (XRA), l'angiografia con risonanza magnetica (MRA), la tomografia computerizzata (CT), e altre modalità di imaging sono fortemente utilizzate nella pratica clinica infatti con i progressi della tecnologia, l'imaging diagnostico è diventato uno strumento indispensabile in medicina. L'estrazione dei vasi sanguigni da immagini mediche è di grande interesse per la diagnosi, la pianificazione, l'esecuzione dei trattamenti e la valutazione dei risultati clinici in diversi campi, tra cui la neurochirurgia, oftalmologia e oncologia. La segmentazione automatica o semiautomatica dei vasi può supportare i medici nell'esecuzione di questi compiti.

L'estrazione automatica di queste strutture da immagini 3D costituisce ancora un problema per la complessità degli schemi di ramificazione e per la diversità geometrica, caratteristiche che vengono accentuate in caso di patolo-

gia. Alcune delle principali ipotesi formulate per i vasi sani, come la linearità e la sezione trasversale circolare, non sono rispettate nei tessuti patologici che richiedono nuove formulazioni di modelli vascolari. Inoltre, c'è la crescente necessità di avere un gran numero di set di dati di immagini 3D segmentati per la formazione di sistemi di apprendimento automatico e per la validazione dei metodi ma vi è una scarsità di alberi 3D segmentati da esperti. La segmentazione manuale dei vasi sanguigni è una procedura costosa in termini di tempo e priva di ripetibilità e riproducibilità intra e inter-operatore. I metodi di estrazione dei vasi sanguigni possono essere classificati in diverse categorie a seconda dell'approccio di segmentazione usato: region growing, contorni attivi, tracking, machine learning ed altri [1]. Tra tutte queste tecniche, si è scelto quindi di implementare un algoritmo di tracking 3D da applicare a strutture vascolari ramificate di media complessità.

Capitolo 1

Tecniche di estrazione degli alberi vascolari

I metodi di segmentazione variano in base alla modalità di imaging, al dominio dell'applicazione, al metodo automatico o semi-automatico e ad altri fattori specifici [1]. È possibile suddividere gli algoritmi di segmentazione dei vasi in quattro categorie principali:

- tecniche di riconoscimento dei pattern;
- approcci basati sui modelli deformabili;
- approcci basati sul tracking;
- approcci basati sul machine learning.

1.1 Tecniche di riconoscimento dei pattern

Le tecniche di riconoscimento dei pattern riguardano il rilevamento o la classificazione automatica di oggetti o caratteristiche. Nel campo dell'estrazione dei vasi, queste tecniche rilevano automaticamente la struttura e le caratteristiche del vaso.

1.1.1 Approcci basati sullo skeleton

Lo skeleton è una tecnica che si occupa di ridurre al minimo la rappresentazione di un oggetto mantenendo intatta la coerenza morfologica, ovvero iterativamente si assottiglia l'oggetto in modo da ridurre al minimo la rappresentazione preservandone la forma. Ad esempio per i vasi sanguigni, la rappresentazione minima è data dal proprio asse (Figura 1.1). Ci sono diverse metodiche per ottenere lo skeleton di una rappresentazione purché si ottenga una rappresentazione minimale e si evitino le disconnessioni. Due tipologie di skeleton si adattano meglio ai problemi di tipo vascolare e sono quello topologico e geometrico.

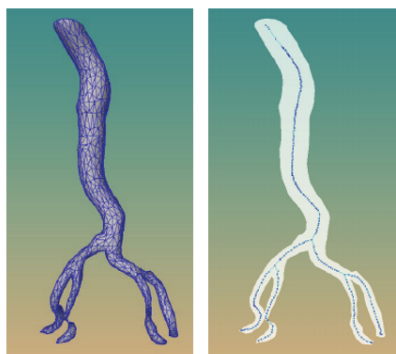


Figura 1.1: Estrazione centerline da un modello vascolare di aorta [2].

Skeleton topologici

Gli skeleton topologici considerano un voxel sulla base del proprio vicinato ovvero viene definita una maschera centrata nel voxel in questione e se il vicinato contiene voxel non essenziali per la rappresentazione, il voxel centrale viene eliminato. La maschera viene poi traslata sul voxel successivo e si eseguono le stesse valutazioni. Una tipologia di maschera può essere costituita ad esempio da un insieme 26-connesso (Figura 1.2). Tuttavia questa tecnica ha un alto costo computazionale essendo un processo iterativo pur avendo il vantaggio di ottenere sempre una rappresentazione minimale.

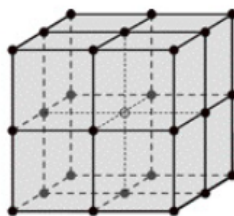


Figura 1.2: Insieme 26-connesso [3].

Skeleton geometrici

Gli skeleton geometrici si basano sul calcolo della distance transform (DT), nota anche come distance map la quale è una rappresentazione dell'immagine digitale tale per cui ogni pixel appartenente all'immagine viene etichettato con la distanza con il suo vicino più prossimo (Figura 1.3). La DT di un punto appartenente ad un oggetto è definita come la minima distanza dal bordo dell'oggetto stesso dunque i punti più vicini al centro dell'oggetto hanno un valore corrispondente alla DT molto alto. Quindi per poter calcolare lo skeleton di un oggetto contenuto in un'immagine si calcola la DT, succes-

sivamente si effettua global thresholding impostando una soglia opportuna. Questa tecnica permette di ottenere uno skeleton in tempi ridotti ma la scelta della soglia è cruciale: imponendo un valore troppo alto, si rischia di creare disconnessioni, per contro imponendo un valore troppo basso, non si ottiene lo skeleton minimale. Nell'elaborazione di immagini vascolari si usano entrambe le tecniche in cascata, skeleton geometrico con una soglia non molto alta seguito da skeleton topologico.

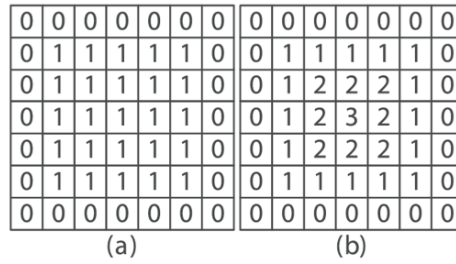


Figura 1.3: (a) Immagine binaria. (b) Distance transform. [4]

1.1.2 Approcci ridge-based

Sono metodi che trattano immagini in scala di grigi come una mappa 3D dove la terza dimensione è l'intensità dell'immagine. Partendo da un pixel arbitrario nella mappa di intensità e tracciando il percorso lungo la direzione di salita più ripida, si raggiunge un punto di massimo locale che costituisce una cresta. Poiché questi approcci rilevano lo scheletro centrale degli oggetti, possono essere pensati come approcci di skeletonizzazione specializzati. L'applicazione principale di questi metodi è nella registrazione di immagini vascolari ottenute dallo stesso paziente in tempi diversi di modo che l'esperto possa osservare i cambiamenti e la posizione di questi ultimi nel tempo [1].

1.1.3 Approcci basati sul region growing

Il region growing è una tecnica che consente di segmentare e riconoscere un oggetto nell'immagine valutando l'intensità del pixel e la prossimità spaziale: due pixel possono essere raggruppati se sono caratterizzati da un simile valore di intensità e se sono vicini l'un l'altro. Il principale svantaggio di questo metodo è che richiede un *seme* di partenza indicato dall'utente, inoltre a causa della presenza di rumore, il region growing può causare errori nella segmentazione. Questa metodica può essere usata per estrarre strutture vascolari che vengono poi sottoposte a tecniche di skeletonizzazione per poterne identificare la centerline [1].

1.2 Approcci basati sui modelli deformabili

I modelli deformabili prendono in considerazione le curve o le superfici (S), definite all'interno del dominio dell'immagine, che possono muoversi e deformarsi sotto l'influenza delle forze interne (F_{int}) ed esterne (F_{ext}). Le prime servono per conferire un movimento alla curva e per determinarne o regolarizzarne il comportamento, le seconde la spingono verso le caratteristiche dell'immagine desiderate come ad esempio linee o bordi. Tali approcci possono essere suddivisi in edge-based e region-based.

1.2.1 Edge-based

In base alla rappresentazione di S , vengono classificati in modelli parametrici e modelli geometrici.

Modelli parametrici

Sono chiamati anche contorni attivi o *snake* proprio perchè sono un insieme di punti connessi tra loro, chiamati *snaels* e ad ognuno di essi è associato un valore di energia. Quest'energia aumenta o diminuisce a seconda delle forze che agiscono su quel punto. Rispetto ai modelli geometrici, la principale limitazione di questa classe di algoritmi è la loro difficoltà ad adattarsi ai cambiamenti della topologia del vaso, a causa della parametrizzazione di S . Questi modelli possono essere usati ad esempio per la ricostruzione 3D dei percorsi cateteri in immagini di angiografia ma anche per tracciare l'aorta in immagini di risonanza magnetica cardiovascolare [1].

Modelli geometrici

I modelli deformabili geometrici, chiamati anche *Level Set*, sono introdotti per risolvere alcune problematiche che caratterizzano i modelli deformabili parametrici come la dipendenza di questi dalle *condizioni iniziali*, dalle energie *interna* ed *esterna*, la difficoltà di estensione alle *grandi strutture* e al *3D* e in ultimo la difficoltà nel gestire *cuspidi* e *concavità* [1]. Sono approcci che possono essere applicati per l'estrazione della struttura vascolare in immagini XRA come mostrato nella Figura 1.4.

1.2.2 Region-based

I modelli deformabili region-based considerano la curva deformabile S in movimento sotto vincoli del primo piano e dello sfondo dell'immagine. Sono metodi che si basano sulla definizione delle statistiche regionali. Sono sta-

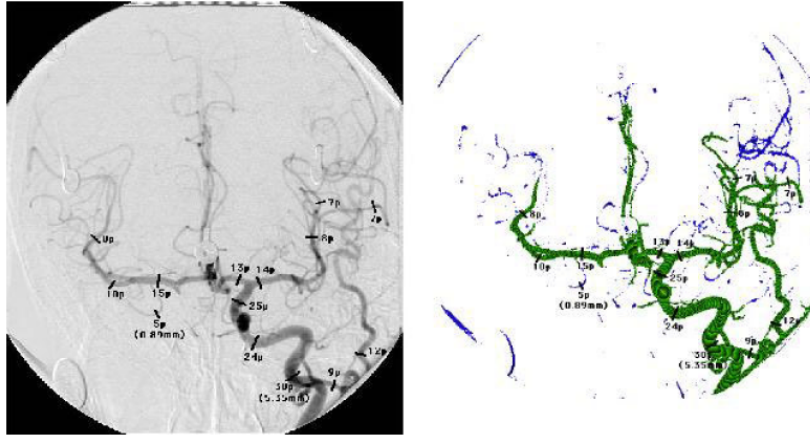


Figura 1.4: A sinistra immagine originale, a destra estrazione della struttura vascolare [1].

ti introdotti per superare alcune problematiche che caratterizzano i modelli edge-based come ad esempio la perdita dei bordi che si verifica soprattutto quando si analizzano immagini rumorose o di intensità non uniforme. Tuttavia, sono caratterizzati da costi computazionali più elevati rispetto ai modelli deformabili basati sul bordo [5].

1.3 Approcci basati sul tracking

Sono algoritmi che di solito consistono nella definizione di punti *seme* seguiti da un processo di crescita guidato da vincoli derivanti dall'immagine. I punti di *seme* possono essere definiti manualmente o ottenuti in modo automatico. Gli algoritmi di tracking sono particolarmente utili per segmentare alberi vascolari connessi per i quali la segmentazione può essere ottenuta usando un numero limitato di punti seme. Una classificazione dei diversi metodi di tracking può essere effettuata considerando approcci basati su *modelli*, che

tracciano i vasi secondo un modello vascolare e approcci di percorso di *costo minimo*, i quali trovano il percorso minimo tra due punti seme [5].

1.3.1 Tracking basato su modelli

Gli approcci di tracking basati su modelli, di solito usati per la segmentazione di vasi 3D, affrontano il problema della segmentazione come il tracciamento di un modello predefinito in volumi 3D. Il modello può assumere forme diverse, anche se il più comunemente adottato è un cilindro 3D con sezione ellittica o circolare, l'ultimo dei quali è solitamente preferito quando si tratta di piccoli vasi. La posizione iniziale e l'orientamento del modello sono definiti posizionando un seme iniziale e stimando la direzione del vaso manualmente o automaticamente. In ogni fase del tracking, la posizione e l'orientamento del modello vengono calcolati localmente come la migliore corrispondenza tra il modello e i dati dell'immagine nel vicinato del modello stesso.

Per identificare la possibile traiettoria del vaso possono essere usati sia approcci deterministici che statistici mentre per effettuare la corrispondenza tra modello e immagine possono essere sfruttate diverse features calcolate sull'immagine come ad esempio l'intensità. Questa tipologia di algoritmi non riesce a tracciare il vaso nel migliore dei modi in presenza di disomogeneità nell'intensità, con immagini rumorose e quando il modello non è in grado di replicare l'architettura del vaso come accade in presenza di patologie. Un approccio di tipo deterministico è proposto in [6] per segmentare le coronarie in immagini di tomografia computerizzata, sfruttando il *Multiple Hypothesis Template Tracking (MHTT)*: ad ogni fase di tracking si hanno una serie di possibili previsioni del modello cilindrico e al posto di preservare solo

la migliore corrispondenza assoluta, che non sempre rappresenta il percorso corretto, vengono considerate molteplici traiettorie costruendo un albero di ricerca. La foglia dell'albero che garantisce la corrispondenza globale migliore, fornisce la segmentazione finale. In questo modo è possibile gestire il tracking di vasi di piccolo diametro e identificare le biforcazioni anche in presenza di immagini rumorose.

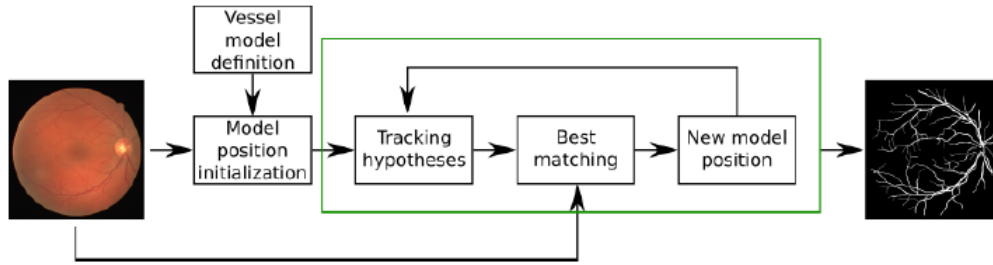


Figura 1.5: Gli approcci di tracking model-based richiedono la definizione iniziale del modello e una successiva inizializzazione. Ad ogni step di tracking, la nuova posizione del modello è ottenuta trovando la migliore corrispondenza tra immagine e modello [5].

1.3.2 Tracking basato sul percorso di costo minimo

L'algoritmo di costo minimo trova il percorso più breve tra due punti p_1 e p_2 , riducendo al minimo un funzionale di energia che dipende da un tensore metrico M , il quale è definito in base alle caratteristiche dell'immagine come intensità, gradiente o derivate di ordine superiore. La minimizzazione dell'energia viene raggiunta calcolando la mappa della minima azione e una volta fatto questo, viene identificato il percorso minimo tra i due punti (Figura 1.6).

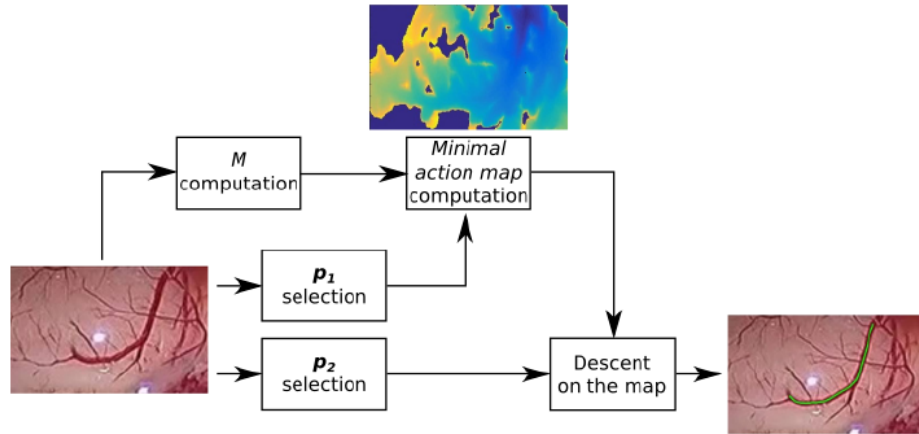


Figura 1.6: Passi fondamentali per il tracking usando un approccio basato sul percorso di costo minimo [5].

1.4 Approcci basati sul machine learning

Esistono due principali approcci di apprendimento automatico e sono il supervisionato e il non supervisionato. Il primo apprende un modello di dati da un insieme di caratteristiche già etichettate mentre il secondo trova un modello in grado di descrivere l'immagine di input senza alcuna conoscenza su di essa, quindi è utile nei casi in cui non si dispone di un set di dati segmentati da un esperto.

1.4.1 Apprendimento non supervisionato

Gli approcci di apprendimento non supervisionato costruiscono un modello di segmentazione basato su features (come intensità, gradiente, colore) di immagini delle quali non si conosce la classe di appartenenza. Durante la fase di allenamento viene valutata la bontà del modello secondo una funzione di

minimizzazione che mira a trovare la migliore separazione tra le classi (Figura 1.7).

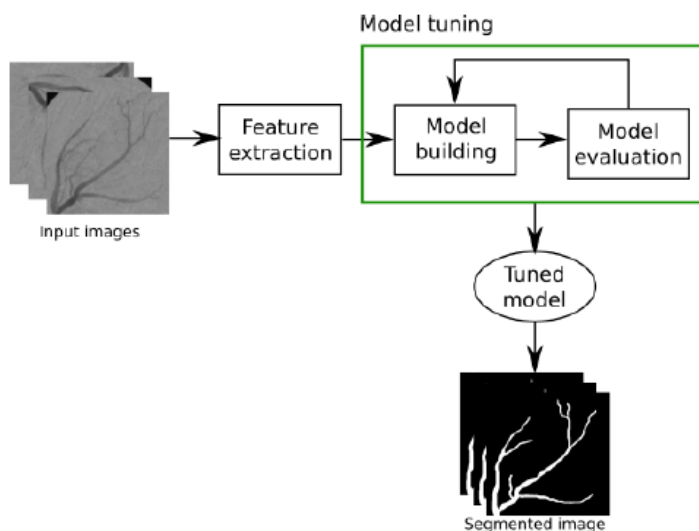


Figura 1.7: Approccio non supervisionato [5].

1.4.2 Apprendimento supervisionato

Durante la fase di training, le features dell'immagine vengono estratte dalle immagini appartenenti al set di dati di training. Un modello di machine learning viene allenato con queste features e la classe di appartenenza di ciascuna immagine. Terminata la fase di allenamento, il modello è usato per effettuare predizioni su immagini nuove, appartenenti al data set di test, ottenendo così la segmentazione dei vasi.

Un esempio di approcci supervisionati sono le reti neurali usate per segmentare i vasi della retina [5].

Inoltre, stanno acquistando sempre più popolarità le reti neurali convoluzionali (CNNs), le quali sono reti neurali *feed-forward* in cui la connessione

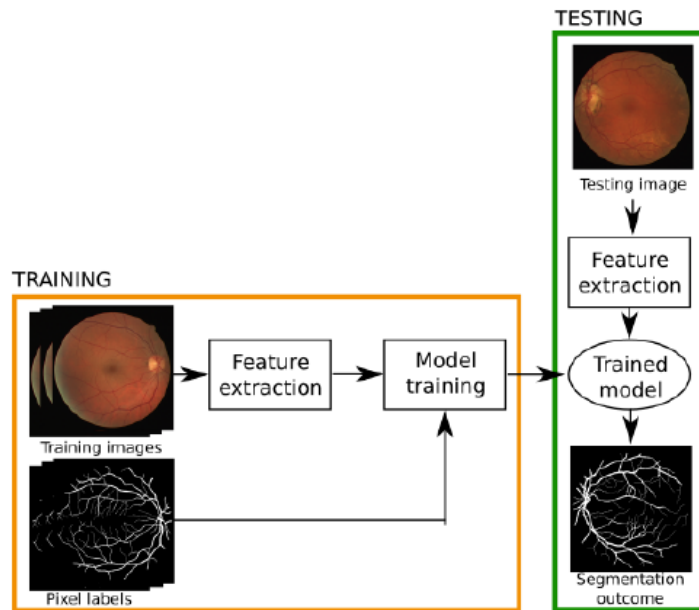


Figura 1.8: Approccio supervisionato [5].

tra i neuroni è ispirata all'organizzazione della corteccia visiva umana. Gli elementi costitutivi di una CNN sono gli strati convoluzionali (adibiti all'estrazione delle features), i layers non lineari, i layers di pooling e infine il fully connected layer il quale effettua la classificazione. I parametri dei layers convoluzionali consistono in una serie di filtri, i cui valori vengono appresi durante l'allenamento della rete. Le CNNs possono essere usate in due modi per la segmentazione dei vasi (Figura 1.9):

- i layers convoluzionali sono usati per estrarre le features dall'immagine che sono successivamente classificate da un approccio standard di apprendimento supervisionato;
- la CNN è usata per ottenere direttamente la segmentazione vascolare usando i fully connected layers.

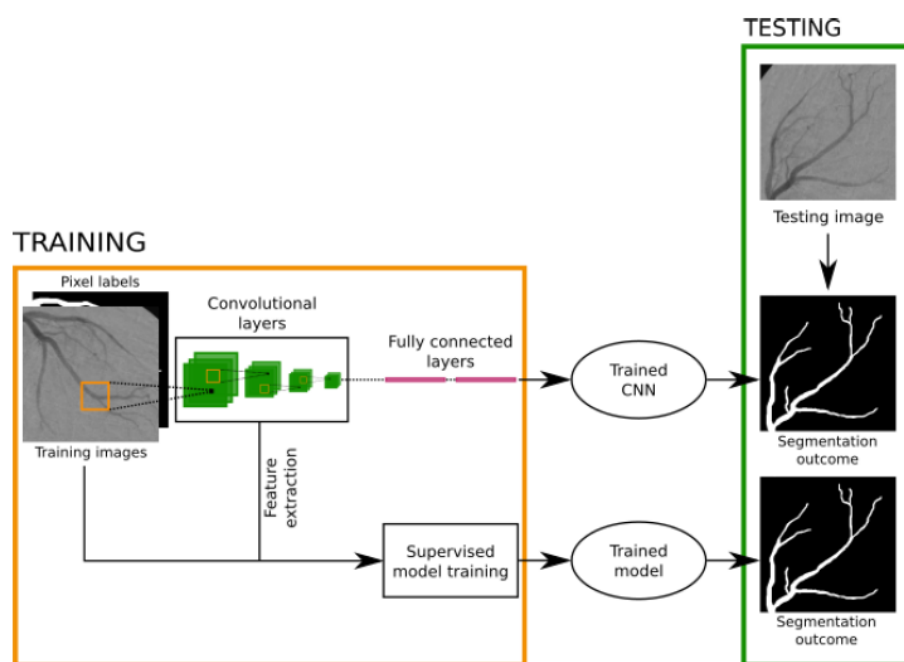


Figura 1.9: Approccio supervisionato: reti neurali convoluzionali [5].

Capitolo 2

Materiali e Metodi

In questo capitolo vengono descritti i volumi usati per implementare l'algoritmo partendo dai modelli simulati in Matlab e con il software CAD Solidworks fino a giungere ai volumi reali relativi alla vascolarizzazione cutanea. Si entra poi nella descrizione delle metodiche applicate per effettuare il tracking 3D di strutture vascolari ramificate più o meno complesse. Infine vengono analizzate le metriche per la validazione dei risultati ottenuti.

2.1 Modelli simulati

Per l'implementazione dell'algoritmo si è fatto uso di quattro modelli creati in ambiente Matlab e tre creati con il software CAD Solidworks in modo da simulare sia semplici segmenti vascolari con diversa curvatura e tortuosità sia alberi vascolari complessi per numero di biforcazioni e presenza di rami con diametro variabile.

2.1.1 Matlab

Nella Figura 2.1 (a-d) sono mostrati i quattro modelli usati per gestire i casi di variazione di curvatura lieve (Figura 2.1a), torsione nello spazio (Figura 2.1b), gestione di una semplice biforcazione (Figura 2.1c) e infine gestione di biforcazioni miste a rami curvi (Figura 2.1d). Dopo aver definito analiticamente ciascun modello e ricavate le coordinate, si è creata una maschera binaria volumetrica avente 1 solo in corrispondenza delle coordinate del vaso. Gli step seguiti per la creazione di ciascun volume sono schematizzati nella Figura 2.2.

Il primo modello rappresenta un segmento di vaso che cambia la sua curvatura nello spazio seguendo la funzione periodica coseno:

$$y = R\cos(2\pi t) \quad (2.1)$$

con $R = 7$.

Nella Figura 2.1b è modellizzata un'elica cilindrica con raggio variabile. L'equazione parametrica di un'elica avvolta attorno ad un cilindro di raggio R , con asse verticale passante per l'origine è data dalla 2.2. Per la creazione del modello è stata introdotta un moltiplicazione con una funzione esponenziale in modo da ottenere un'elica con raggio variabile (2.3).

$$\begin{cases} x = R\cos(2\pi t) \\ y = R\sin(2\pi t) \\ z = ct \end{cases} \quad (2.2)$$

dove $|c|$ indica il passo dell'elica.

$$\begin{cases} x = Re^{\alpha t}\cos(2\pi t) \\ y = Re^{\alpha t}\sin(2\pi t) \\ z = ct \end{cases} \quad (2.3)$$

con $\alpha = 1,1$.

Per quel che riguarda il modello a forma di *Y rovesciata* (Figura 2.1c), è stato creato un segmento di lunghezza 25 pixel e poi ruotato nello spazio in modo da formare una biforcazione con tre rami e un angolo di 140° tra loro. L'ultimo modello è il risultato di una composizione di rami, diversi per curvatura e lunghezza che si incontrano in due punti in modo da simulare una doppia biforcazione.

2.1.2 Solidworks

In Solidworks sono stati creati tre modelli: il primo simula un'architettura vascolare semplice con pochi punti di biforcazione e rami con diametro costante; il secondo è più complesso, caratterizzato da un elevato numero di rami; il terzo modella una struttura profonda caratterizzata da rami con diametro variabile (Figura 2.1e-g). In questo lavoro di tesi non ci si è occupa-

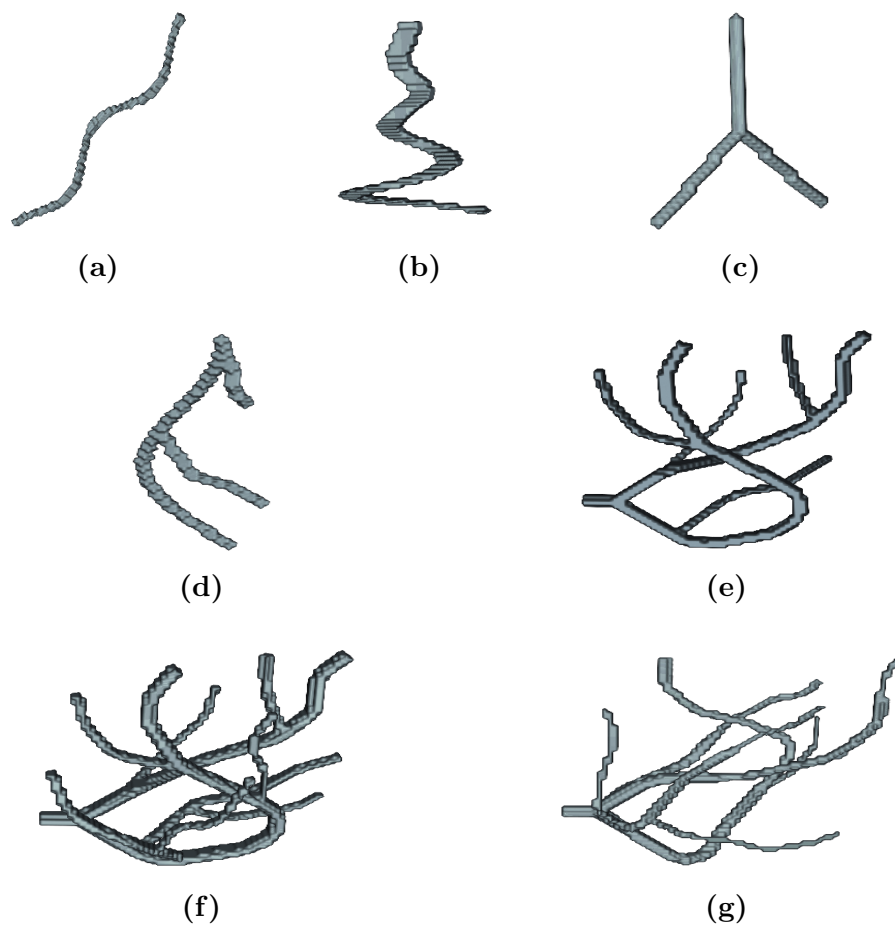


Figura 2.1: Modelli simulati in Matlab e Soliworks. (a) Curvatura lieve. (b) Elica cilindrica con raggio variabile. (c) Biforcazione ad Y. (d) Biforcazione doppia. (e) Struttura semplice. (f) Struttura complessa. (g) Struttura profonda.

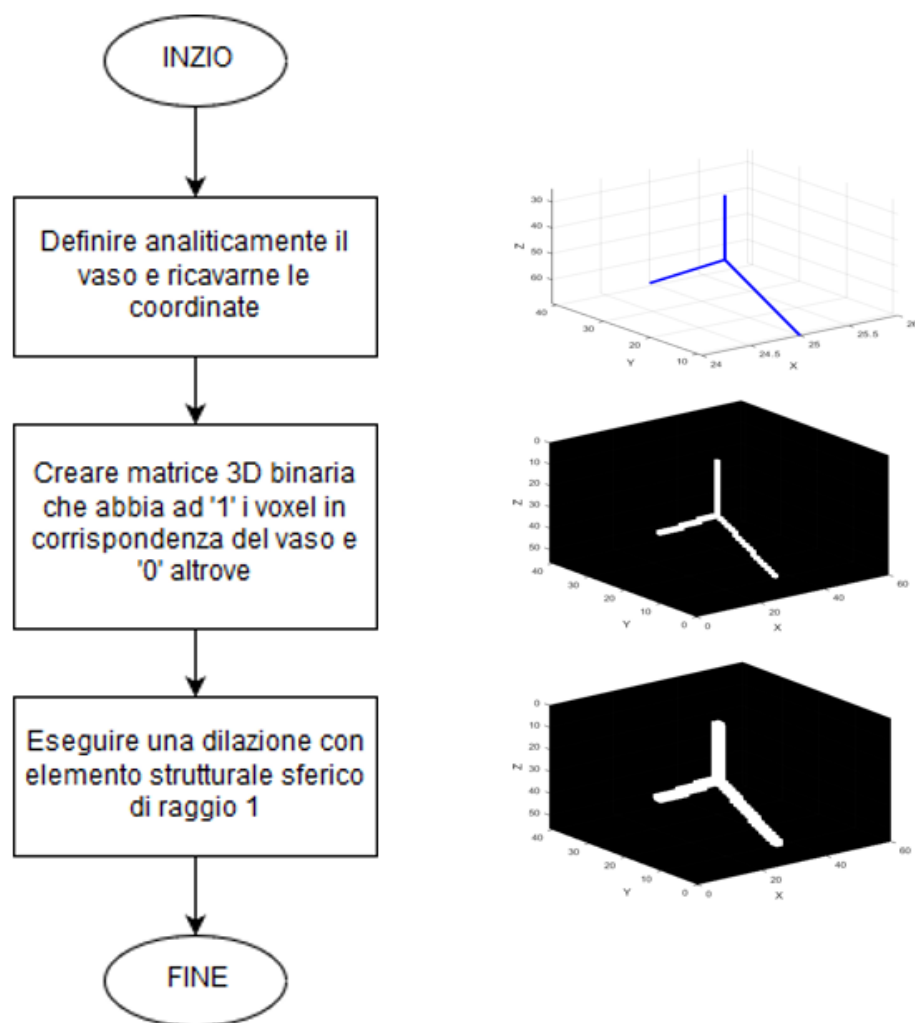


Figura 2.2: Schematizzazione della creazione dei modelli in Matlab.

ti della creazione in Solidworks dei tre modelli sopracitati ma sono stati usati come strumento per rendere l'algoritmo più facilmente adattabile alla diversa complessità delle architetture vascolari reali. Poichè i modelli creati in Solidworks sono serviti in un altro lavoro di tesi per la creazione di strutture reali mediante stampante 3D, sono stati precedentemente convertiti in formato *.stl*. Tuttavia per essere utilizzati in Matlab, è stata necessaria un'ulteriore conversione in formato *.mat* tramite un insieme di funzioni *Mesh voxelisation* disponibili su FileExchange di MathWorks [7] le quali implementano un algoritmo di *Ray tracing* [8].

Nelle tabelle 2.1, 2.2 e 2.3 sono riportate le misure reali dei modelli espresse in *mm* e quelle dei volumi corrispondenti usati in Matlab. Per ottenere le matrici 3D si deve scegliere un fattore di conversione dato dall'equazione 2.4:

$$FC = 1/Risoluzione \quad (2.4)$$

dove

$$Risoluzione = \frac{D_{MAX}}{D_{MATR}} \quad (2.5)$$

con D_{MAX} pari alla dimensione massima dell'oggetto e D_{MATR} pari alla dimensione della matrice, scelta dall'utente prima della procedura di conversione. In particolare, in questo lavoro di tesi è stato scelto un FC diverso per ogni direzione (x, y e z) e per ogni modello, in modo da ottenere architetture vascolari con diametro dei vasi pari al massimo a 5 pixel. È stata fatta questa

scelta perché si vuole implementare un algoritmo di tracking rivolto a strutture vascolari con piccolo diametro. In particolare il modello 1 fa riferimento alla Figura 2.1e, il modello 2 alla 2.1f e il modello 3 alla 2.1g.

	Direzione	Dim modello(mm)	Dim modello(pixel)	Risoluzione
Modello 1	x	6,9184	16	0,4324
	y	33,2243	56	0,5932
	z	37,7824	67	0,5639

Tabella 2.1: Dimensioni modello 1 in Solidworks e in Matlab.

	Direzione	Dim modello(mm)	Dim modello(pixel)	Risoluzione
Modello 2	x	10,7786	50	0,2155
	y	33,2243	59	0,5631
	z	37,7824	63	0,5997

Tabella 2.2: Dimensioni modello 2 in Solidworks e in Matlab.

	Direzione	Dim modello(mm)	Dim modello(pixel)	Risoluzione
Modello 3	x	30,1146	55	0,5475
	y	37,6454	62	0,6071
	z	39,6774	68	0,5834

Tabella 2.3: Dimensioni modello 3 in Solidworks e in Matlab.

Poichè l'algoritmo è stato sviluppato su volumi binari, nella Figura 2.3 sono mostrate le maschere binarie volumetriche relative ai modelli descritti precedentemente.

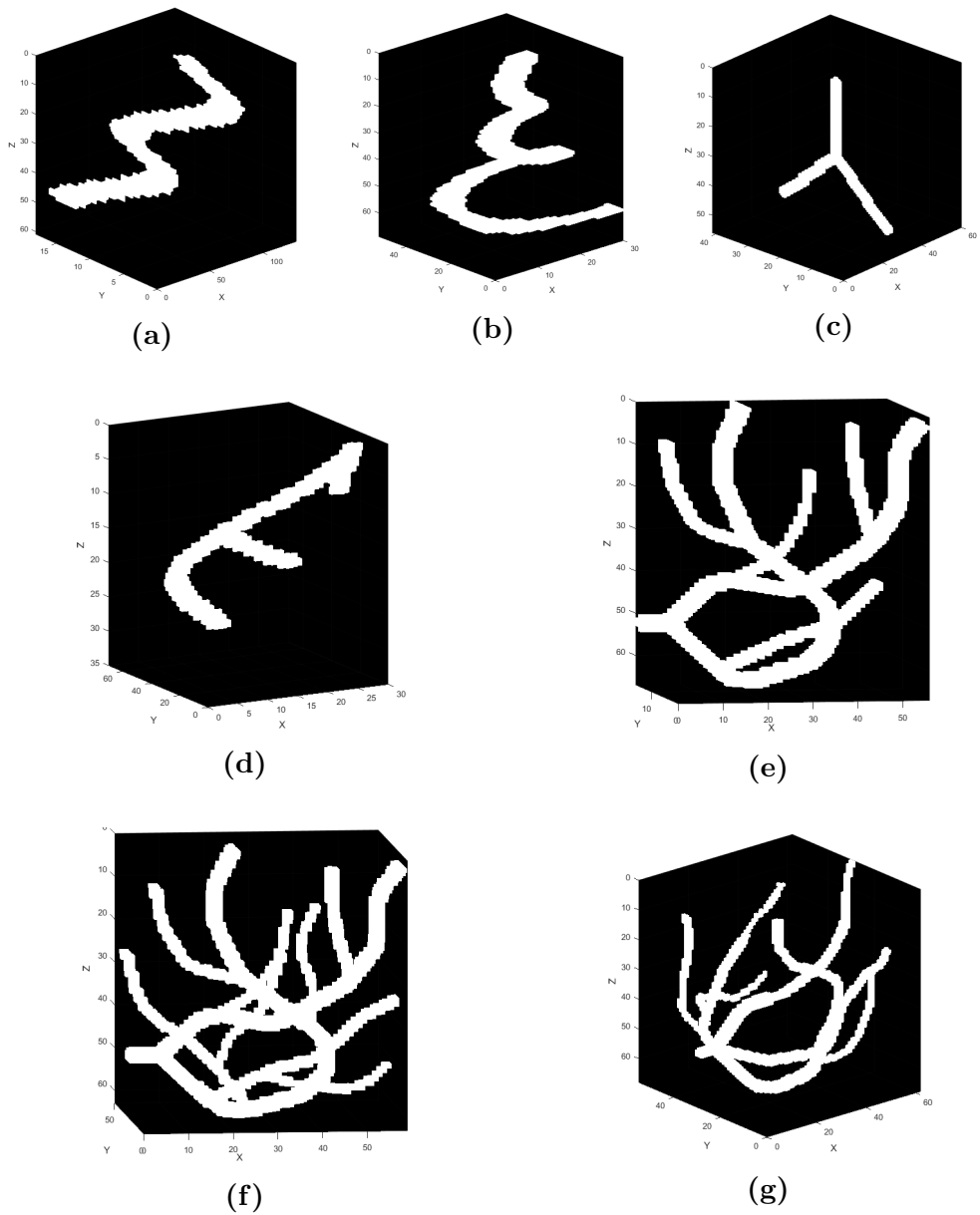


Figura 2.3: Volumi binari. (a) Curvatura lieve. (b) Elica cilindrica con raggio variabile. (c) Biforcazione ad Y. (d) Biforcazione doppia. (e) Struttura semplice. (f) Struttura complessa. (g) Struttura profonda.

2.2 Immagini in vivo

L'algoritmo sviluppato sui modelli simulati è stato testato su immagini in vivo relative alla vascolarizzazione cutanea. Di seguito viene introdotta sinteticamente la tecnica di imaging OCT-A e descritti i volumi utilizzati.

L'angiografia OCT è emersa come tecnica non invasiva, in vivo e ad alta risoluzione per l'imaging della microvascolarizzazione della retina. Tuttavia, un campo promettente di applicazione è senza dubbio la dermatologia: mentre la dermatoscopia offre una vista in alta risoluzione della superficie della pelle, l'OCT-A offre una visualizzazione trasversale della morfologia tissutale fino ad una profondità da 1 a 2 millimetri. La tecnologia OCT-A utilizza la riflettanza della luce laser della superficie dei globuli rossi in movimento per rappresentare accuratamente i vasi. Inoltre, la stessa area del tessuto può essere ripetutamente esposta e le differenze possono essere analizzate nel tempo, in modo da monitorare la patogenesi e la terapia senza la necessità di prelevare campioni di tessuto. I principali vantaggi sono senza dubbio i tempi di acquisizione brevi e il fatto che si tratta di una tecnica di imaging non invasiva [9].

Sapendo che la visualizzazione in vivo dei vasi sanguigni cutanei può aiutare nella diagnosi e nel trattamento dei disturbi dermatologici, negli ultimi anni l'OCT-A è stata applicata attivamente nella ricerca dermatologica per una varietà di malattie della pelle [10].

Nella Figura 2.4 si osserva un esempio di creazione di un volume binario a partire dalle immagini OCT-A: si ha l'applicazione del filtro mediano 3D (con kernel $3 \times 3 \times 3$) per ridurre il rumore presente; poi viene applicato un

filtro di vascolarità Frangi poiché gli oggetti di interesse nel volume sono vasi sanguigni, infatti questo filtro ha la duplice funzione di ridurre il rumore nel volume che non corrisponde ai vasi sanguigni e migliorare i vasi stessi. Infine per ottenere il volume binario in cui il bianco corrisponde ai vasi, è stata utilizzata una tecnica di thresholding adattiva [10].

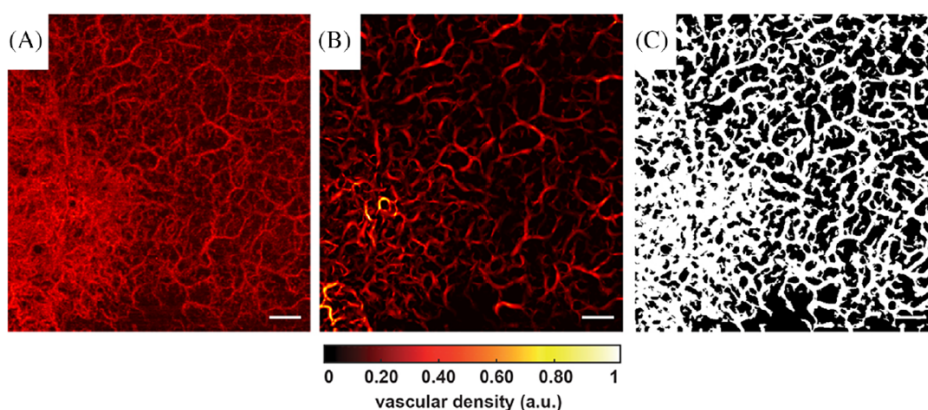


Figura 2.4: Le immagini riportate sono MIP (maximum intensity projection) del volume 3D. (a) Volume originale OCT-A. (b) Volume processato. (c) Volume segmentato. [10]

In questo lavoro di tesi è stato preso in esame un volume $490 \times 512 \times 100$ da cui sono state estratte due ROI di dimensione e posizione diversa in modo da testare l'algoritmo su due strutture vascolari di diversa complessità. Nella Figura 2.5 sono mostrate le MIP relative alle ROI considerate: una fa riferimento ad una struttura semplice, con un'evidente biforcazione; la seconda ROI è relativa ad una struttura più complessa.

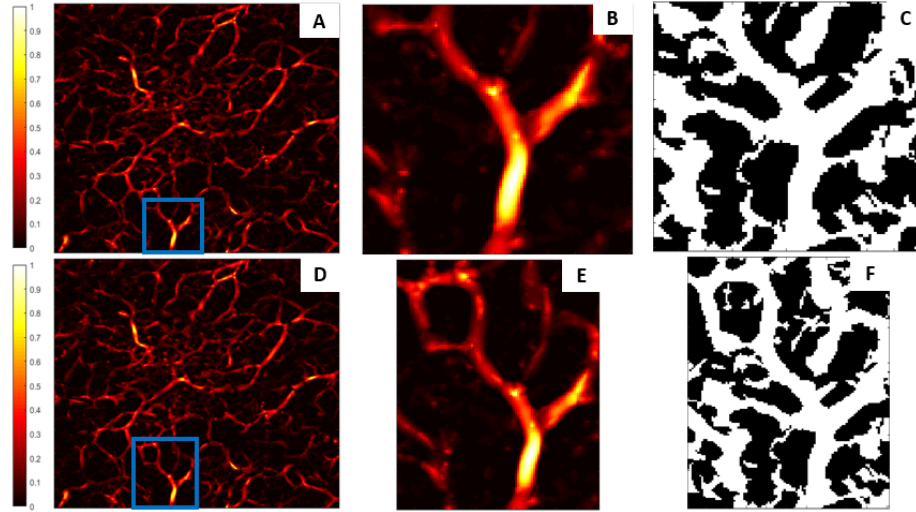


Figura 2.5: Le immagini riportate sono MIP (maximum intensity projection) del volume 3D. (a)(d) Volume OCT-A che ha subito il processing con ROI indicata in blu. (b)(e) Visualizzazione ROI. (c)(f) ROI segmentata.

2.3 Descrizione dell'algoritmo

L'algoritmo di tracking 3D è stato implementato in Matlab facendo riferimento all'articolo [11], nel quale viene proposto un metodo che traccia il vaso in modo attivo riformando il piano perpendicolare all'asse del vaso stesso e segmentando la sua sezione trasversale assumendo che il vaso possa essere assimilato ad un cilindro con sezione ellittica. La scelta di voler identificare ad ogni step di tracking il piano localmente perpendicolare all'asse del vaso è dovuto ad una semplice constatazione osservabile nella Figura 2.6: seguendo un approccio basato su un piano che non varia il suo orientamento nello spazio, si ha una corretta segmentazione solo quando l'asse del vaso è perpendicolare al piano; negli altri casi la sezione del vaso non è ellittica, di conseguenza il punto centrale della sezione viene identificato erroneamente.

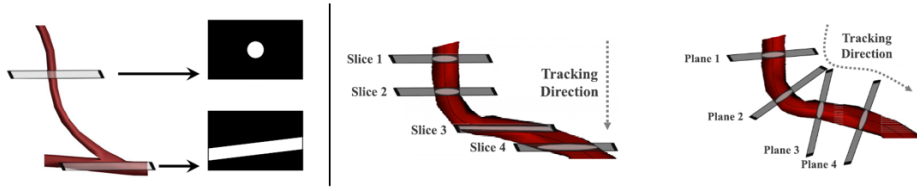


Figura 2.6: Identificazione piano perpendicolare alla sezione del vaso. A sinistra si osserva la sezione trasversale del vaso quando il suo asse è perpendicolare al piano e quando il suo asse è parallelo al piano [11].

L'algoritmo può essere schematizzato nel seguente modo:

1. Dal volume binario di partenza si identificano gli oggetti connessi usando una connettività 18-connessa. La Figura 2.7 mostra una ROI del volume OCT-A descritto precedentemente, nella quale si identificano 110 oggetti connessi. Si sceglie di intraprendere il processo di tracking solo sugli oggetti con un numero di voxel bianchi elevato: in particolare per le due ROI considerate, si sceglie l'oggetto con numero di voxel bianchi massimo poichè si è osservato che gli altri oggetti sono molto piccoli (Figura 2.8). Per quel che riguarda i volumi simulati, essendoci un solo albero vascolare, non viene effettuata nessuna scelta.
2. Per ogni oggetto identificato, si ha l'inizializzazione del piano (processo che verrà descritto dettagliatamente in seguito). In questo modo per ciascun oggetto viene intrapreso il processo di tracking. Una volta tracciati tutti gli oggetti, le centerline risultanti vengono accorpate assieme.
3. Viene eseguito il processo di tracking.

Il tutto viene effettuato in ciclo fino a che tutti gli oggetti scelti nel volume sono stati tracciati (Figura 2.9).

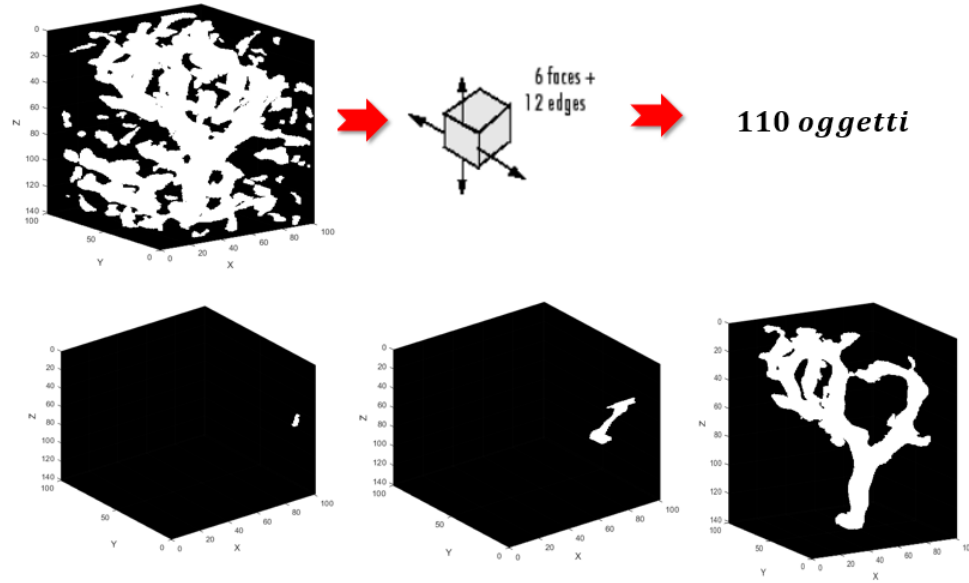
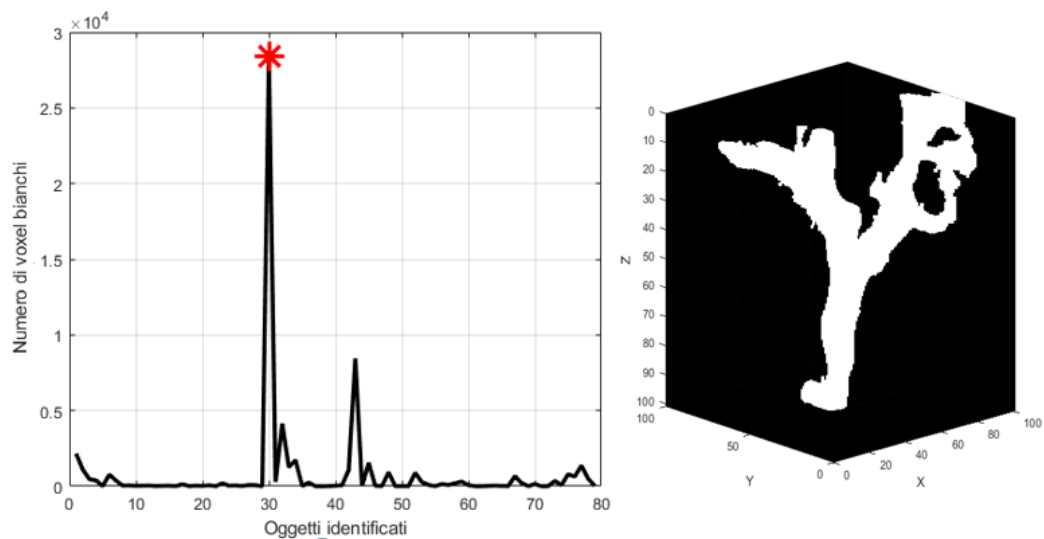
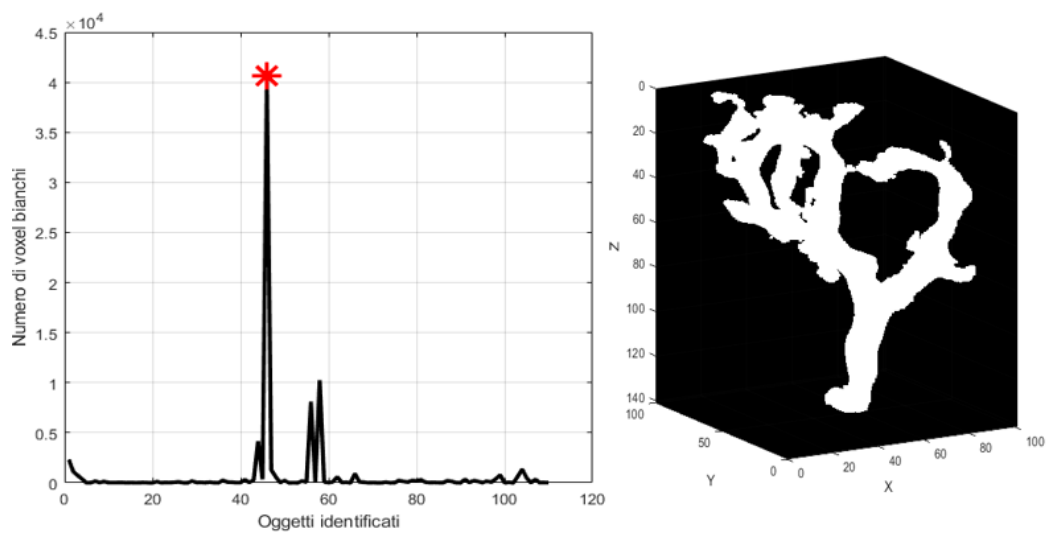


Figura 2.7: Esempio di identificazione degli oggetti connessi in un volume reale: dei 110 oggetti identificati ne sono mostrati solo 3.

In seguito si descrivono i processi di inizializzazione automatica del piano e quello di tracking. Quest'ultimo è caratterizzato da due macro processi, quello di ricerca del piano ortogonale all'asse del vaso e quello della gestione delle biforcazioni e inizio-fine vaso. L'algoritmo è strutturato in modo tale da eseguire una sola volta l'inizializzazione del piano e poi in maniera ciclica i due macro processi del tracking fino a che si è tracciato l'intero volume.



(a)



(b)

Figura 2.8: Scelta dell'oggetto connesso per le due ROI: i due grafici riportano il numero di voxel bianchi per ciascun oggetto connesso per un totale di 79 oggetti nell'immagine (a) e 110 per la (b). Di lato è mostrato il volume binario relativo solo all'oggetto scelto.

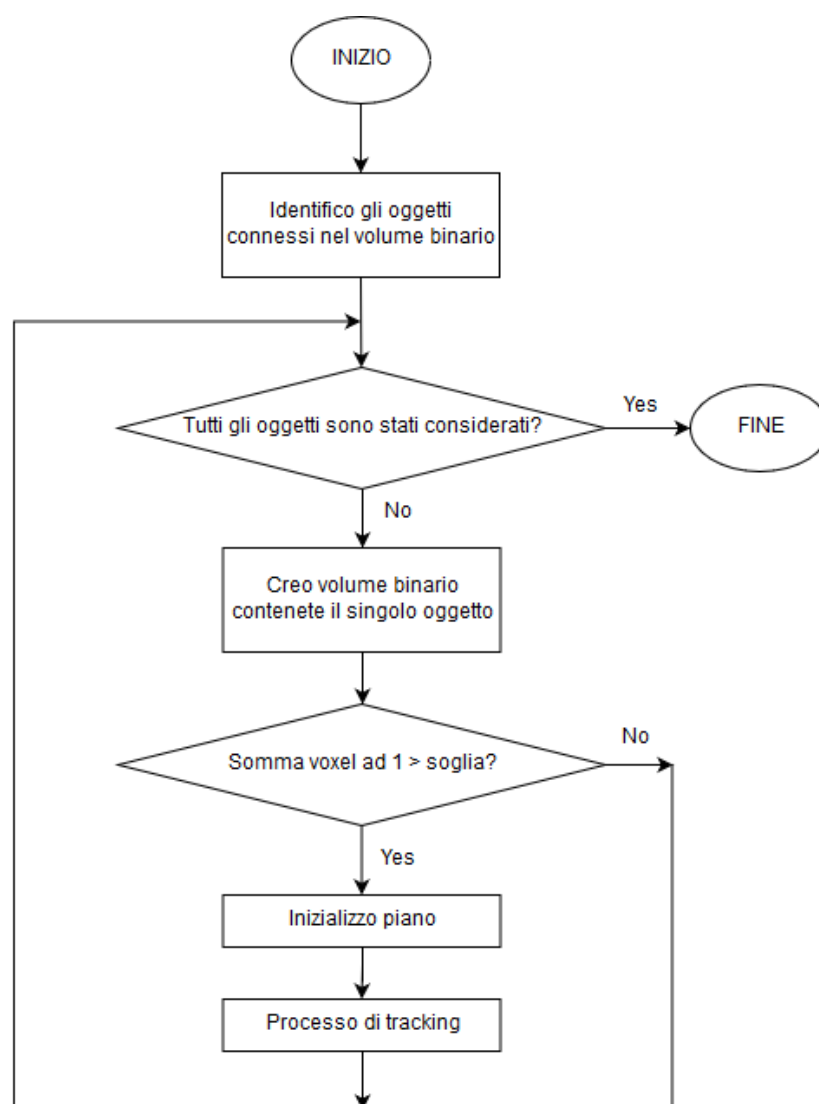


Figura 2.9: Step generali dell'algoritmo.

2.3.1 Inizializzazione automatica del piano

Dato $[x, y, z]$ il sistema di riferimento del volume, vengono definite le coordinate del piano nel sistema di riferimento $[u, v, w]$ che all'inizio del processo di tracking coincide con quello del volume, quindi il piano è orientato orizzontalmente avendo la normale concorde in direzione e verso con l'asse z (Figura 2.10). Per rendere automatica l'inizializzazione del piano, si è scel-

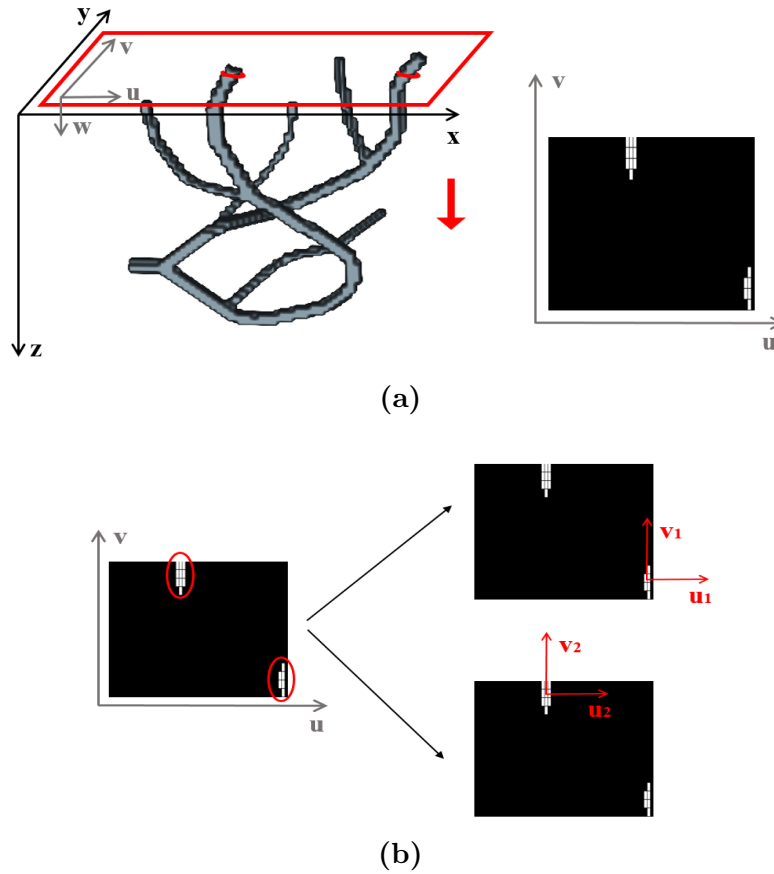


Figura 2.10: Inizializzazione piano. (a) Traslazione del piano iniziale lungo l'asse z . (b) Il sistema di coordinate del piano è centrato nel pixel centrale di ciascuna zona bianca identificata.

to di traslarlo lungo l'asse z e valutare la presenza di pixel bianchi ad ogni intersezione volume-piano: una volta identificato un numero di pixel bianchi diverso da zero, la ricerca del piano termina e vengono salvate le coordinate del piano centrato nel pixel centrale della zona bianca identificata; se si identifica più di una zona bianca, vengono calcolate le coordinate del piano centrato nel pixel centrale di ciascuna di esse e salvate. In quest'ultimo caso, il processo di tracking inizia dal primo piano salvato e una volta terminato, si effettua il tracking partendo dagli altri.

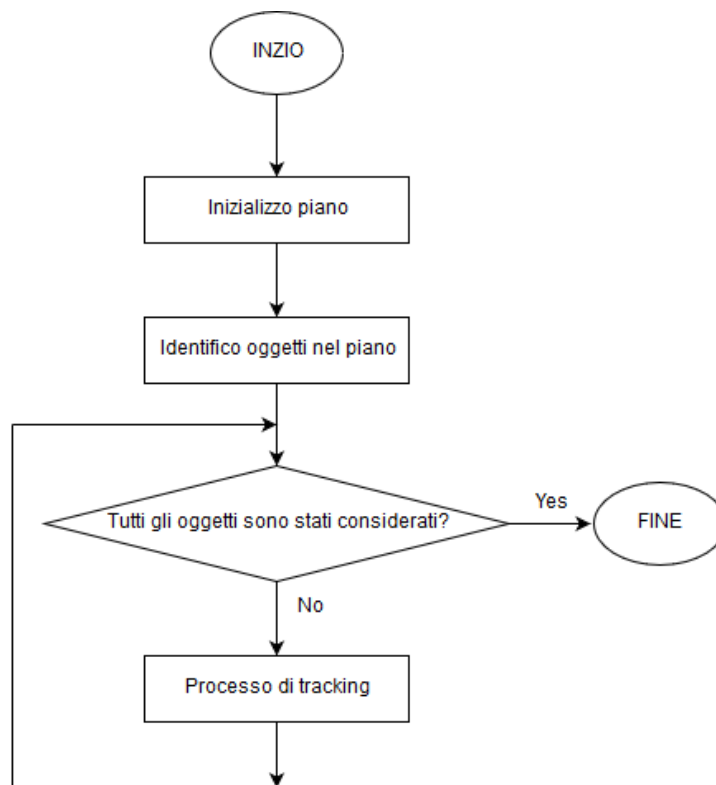


Figura 2.11: Inizializzazione piano.

2.3.2 Tracking

Il tracking può essere scomposto in due sottoprocessi:

1. Ricerca del piano ortogonale alla direzione locale del vaso;
2. Gestione delle biforcazioni e caso di inizio e fine vaso.

A partire dalle coordinate del piano iniziale, si effettua la ricerca del piano ortogonale all'asse del vaso e una volta trovato, viene calcolata la direzione di tracking locale. Vengono calcolate due direzioni come le normali al piano, uguali in direzione ma di verso opposto. Se si è alla prima ricerca del piano ortogonale, viene scelta una direzione come quella di inizio e l'altra viene salvata come direzione B e utilizzata dopo aver terminato il tracking in direzione A (Figura 2.12). Se si è a ricerche successive del piano, viene scelta come direzione di tracking locale quella che forma l'angolo più piccolo con quella dello step di tracking precedente.

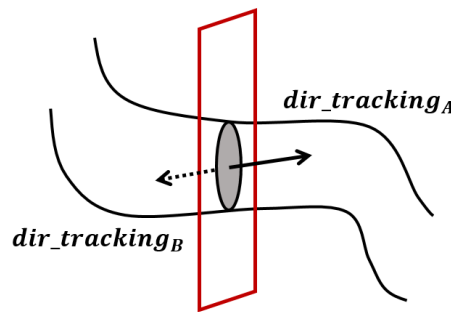


Figura 2.12: Identificazione direzione di tracking al primo step di tracking.

Segue il processo della gestione delle biforcazioni e inizio-fine vaso: si costruisce una sfera centrata nel centro della sezione del vaso identificata. La gestione delle biforcazioni e l'identificazione della fine del vaso si basa

sull'algoritmo di clustering DBSCAN che restituisce le direzioni dei rami. Si possono avere diverse situazioni:

1. $Cluster \geq 3$, si ha una biforcazione;
2. $Cluster = 2$, non si sono identificate diramazioni, quindi il tracking può procedere nella direzione corrente;
3. $Cluster = 1$, si è alla fine o all'inizio del vaso.

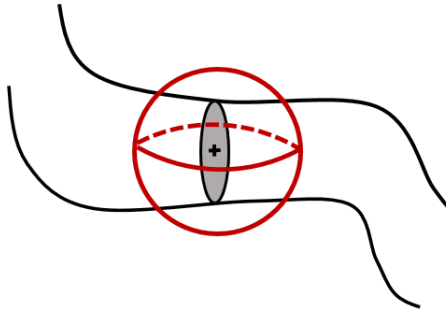


Figura 2.13: Costruzione sfera.

Identificazione del piano perpendicolare

A partire dalle coordinate del piano iniziale, sotto l'assunzione che il vaso possa essere assimilato ad un cilindro con sezione ellittica, si cerca il piano la cui intersezione con il vaso restituisce un profilo il più possibile circolare. Nella Figura 2.14 si osserva il processo di ricerca del piano effettuato ad ogni step di tracking: l'idea alla base è quella di ruotare il piano attorno all'asse minore dell'ellisse e valutare il rapporto tra asse maggiore e minore per ogni angolo di rotazione. La rotazione in teoria non varia la lunghezza dell'asse

minore ma solo del maggiore: più il piano diventa perpendicolare all'asse del vaso, tanto più il rapporto tende ad uno. Si sceglie il piano a cui è associato il più basso valore per questo rapporto e una volta identificato, la normale al piano diventa la direzione locale del vaso.

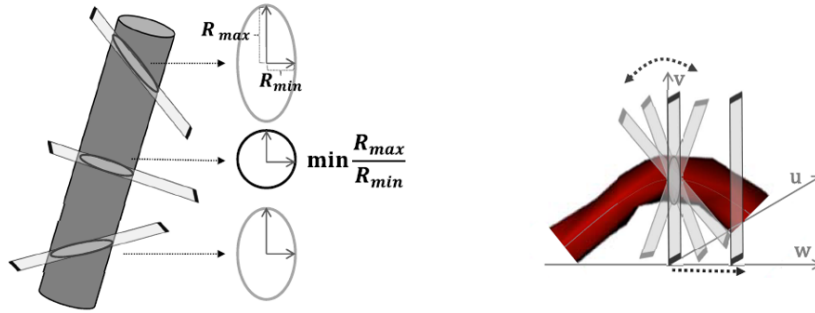


Figura 2.14: Rotazione piano [11].

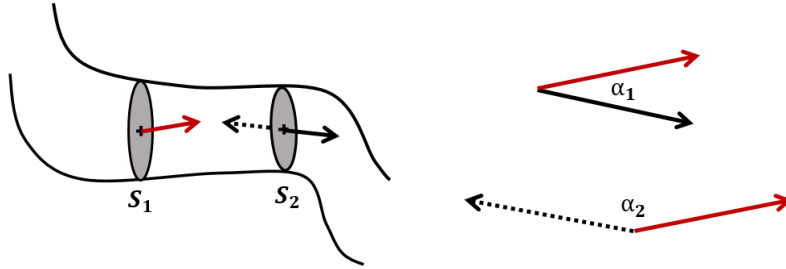


Figura 2.15: Scelta direzione di tracking per S_2 : $\alpha_1 < \alpha_2$, viene scelta come direzione di tracking quella indicata con la linea continua nera.

In questo lavoro di tesi si è scelto di effettuare la rotazione a step di 10° da 0° a 180° . In particolare, gli step seguiti per la ricerca del piano ortogonale alla direzione del vaso sono sintetizzati nella Figura 2.16.

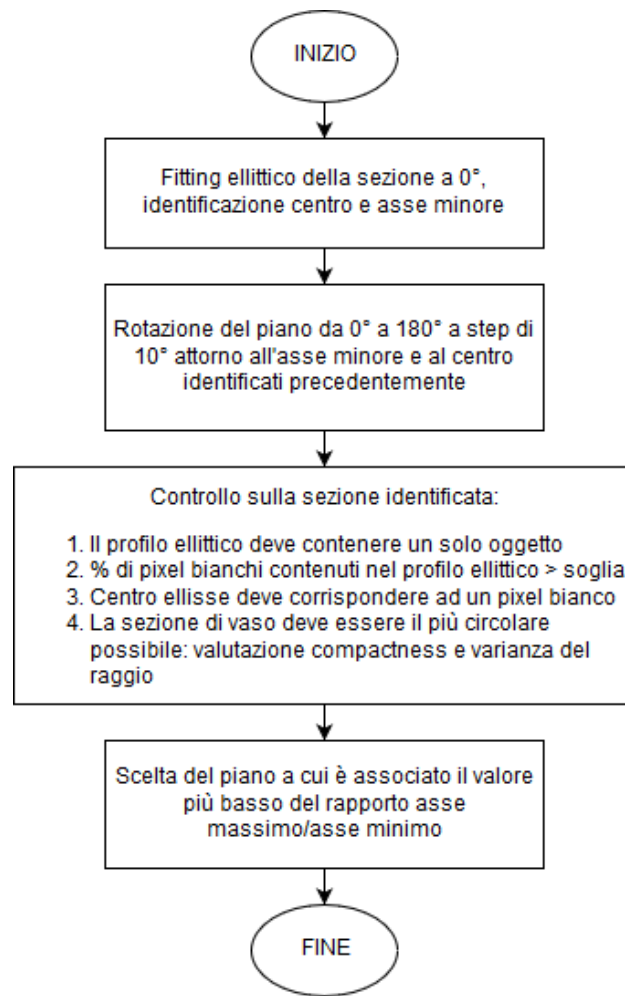


Figura 2.16: Step seguiti per identificare il piano localmente ortogonale all'asse del vaso.

Rotazione piano

É necessario identificare le coordinate del punto e la direzione attorno ai quali effettuare le rotazioni del piano: si interseca il volume con il piano ottenuto dal precedente step di tracking senza applicargli una rotazione.

Successivamente, si ha una traslazione del sistema di riferimento del piano in modo da far coincidere il centro del piano con il pixel centrale della sezione ellittica identificata (Figura 2.17). Ad ogni rotazione successiva la sezione di vaso viene scelta come quella il cui centro ha la minima distanza dal centro del piano. Per effettuare le rotazioni del piano nello spazio viene applicata la formula di rotazione di Rodrigues [12] la quale fornisce un metodo efficiente per calcolare la matrice di rotazione $R_{\hat{\omega}}(\theta)$ corrispondente ad una rotazione di un angolo θ attorno ad un asse fisso specificato dal vettore unitario $\hat{\omega} = (\omega_x, \omega_y, \omega_z) \in \mathbf{R}^3$. Le coordinate del punto e la direzione attorno ai quali effettuare la rotazione restano costanti per tutti gli angoli di rotazione.

$$R_{\hat{\omega}}(\theta) = \begin{bmatrix} \cos\theta + \omega_x^2(1 - \cos\theta) & \omega_x\omega_y(1 - \cos\theta) - \omega_z\sin\theta & \omega_y\sin\theta + \omega_x\omega_z(1 - \cos\theta) \\ \omega_z\sin\theta + \omega_x\omega_y(1 - \cos\theta) & \cos\theta + \omega_y^2(1 - \cos\theta) & -\omega_x\sin\theta + \omega_y\omega_z(1 - \cos\theta) \\ -\omega_y\sin\theta + \omega_x\omega_z(1 - \cos\theta) & \omega_x\sin\theta + \omega_y\omega_z(1 - \cos\theta) & \cos\theta + \omega_z^2(1 - \cos\theta) \end{bmatrix} \quad (2.6)$$

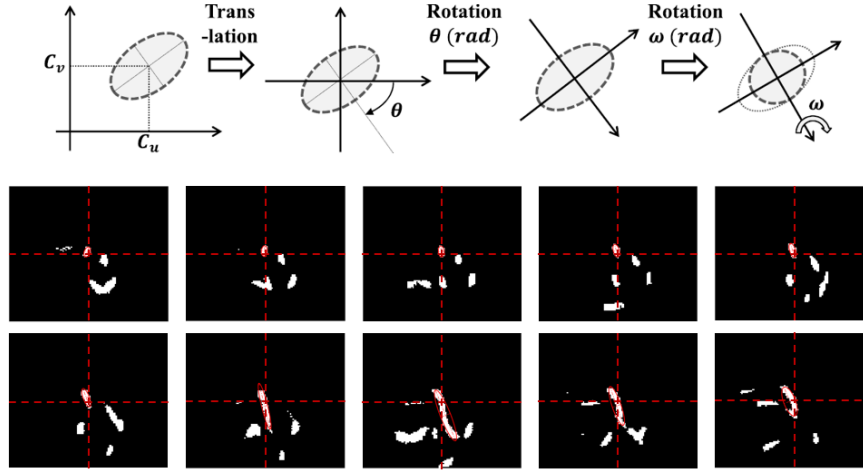
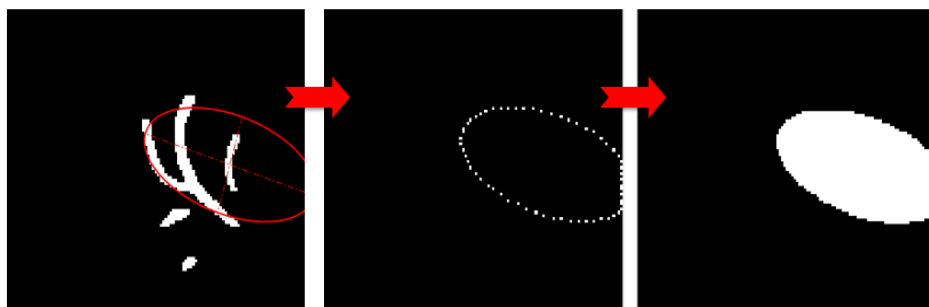


Figura 2.17: Traslazione del sistema di riferimento nel centro dell'ellisse identificata a 0° e rotazione del piano di un angolo θ attorno all'asse $\hat{\omega}$ [11]. In basso sono mostrati 10 piani ruotati nel range $0^\circ - 90^\circ$.

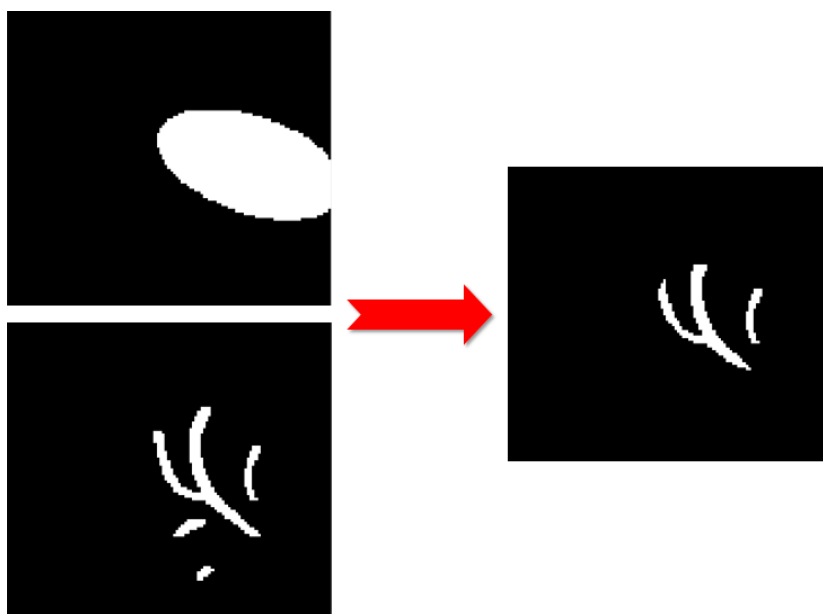
Analisi della sezione di vaso identificata

Basarsi solo sul valore del rapporto tra asse maggiore e minore per la scelta del piano migliore non costituisce sempre la scelta ottima per cui prima di valutare il valore del rapporto, viene effettuata un'analisi sulla sezione del vaso. In particolare si effettuano i seguenti controlli:

- Il profilo ellittico della sezione deve contenere al suo interno un solo **oggetto**: quando la sezione non ha forma circolare-ellittica, l'ellisse ingloba al suo interno sezioni di vaso contigue a quella di interesse; questo succede soprattutto quando il piano è lontano dall'essere perpendicolare alla direzione del vaso. Nella Figura 2.18 è possibile osservare gli step seguiti per analizzare il numero degli oggetti: dal profilo ellittico si ottiene una maschera binaria attraverso la funzione Matlab *poly2mask* che a partire da un poligono definito dalle coordinate dell'ellisse, genera la maschera corrispondente (Figura 2.18a); in secondo luogo, si effettua l'intersezione tra la maschera così ottenuta e il risultato dell'intersezione piano-volume, ottenendo gli oggetti contenuti nel profilo ellittico (Figura 2.18b). Se il numero degli oggetti è superiore ad uno, questa sezione non viene presa in considerazione per la scelta finale.
- Se la sezione in questione supera il controllo precedente, viene sottoposta a quello sulla **percentuale** di pixel bianchi contenuti nel profilo ellittico sul totale di pixel interni all'ellisse. Se il numero di pixel bianchi è minore dell'80%, la sezione viene scartata. Nella Figura 2.20 è



(a)



(b)

Figura 2.18: Analisi della sezione identificata sulla base del numero di oggetti. (a) A partire dal profilo ellittico di ottiene la maschera binaria corrispondente con la funzione Matlab *poly2mask*. (b) Si effettua l'intersezione tra la maschera binaria ottenuta in (a) e quella ottenuta dall'intersezione piano-volume.

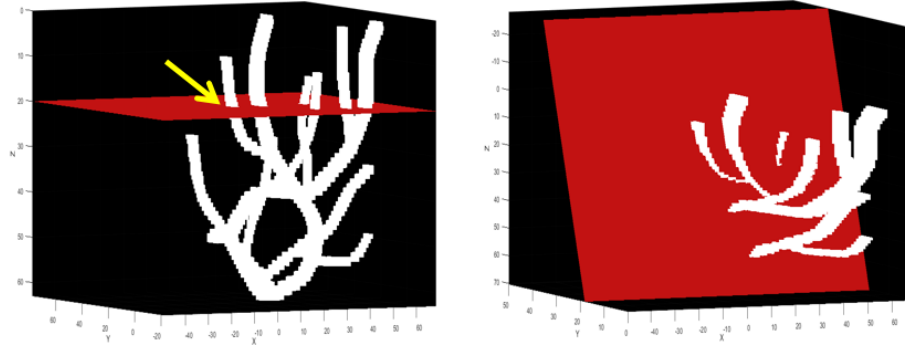


Figura 2.19: Visualizzazione dell'intersezione tra il piano e il volume relativo al modello complesso della Figura 2.1f. A sinistra è mostrato il piano con angolo di rotazione pari a 0° e la freccia gialla indica il vaso attorno al quale viene effettuata la rotazione. A destra si ha il piano ruotato di 100° , il quale porta ad individuare due oggetti separati, infatti non è il piano ortogonale alla direzione del vaso.

mostrato un caso di sezione che riporta un numero di pixel bianchi pari al 43%, per cui non viene considerata per la scelta finale. Anche in questo caso, questa problematica si presenta quando il piano non è perpendicolare alla direzione del vaso (Figura 2.21).

- Superando i controlli precedenti, la sezione viene sottoposta alla verifica della posizione del **centro** dell'ellisse: in corrispondenza di esso deve esserci un pixel bianco altrimenti la sezione viene scartata. Questo controllo è introdotto per i profili leggermente concavi che superando le due verifiche precedenti porterebbero a scegliere come vincitrice una sezione errata. La Figura 2.22 mostra un caso in cui all'interno del profilo ellittico si identifica un solo oggetto, la percentuale di pixel bianchi è maggiore dell'80% ma il centro dell'ellisse è esterno alla sezione di vaso: questo accade con i piani non perpendicolari all'asse del vaso.

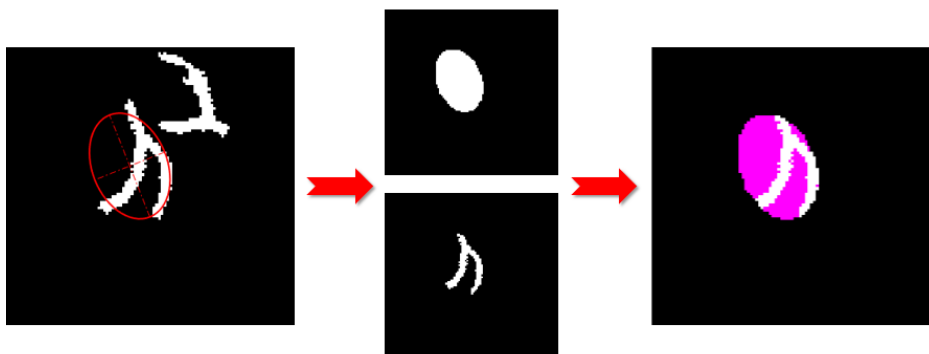


Figura 2.20: Analisi della sezione identificata sulla base della percentuale di pixel bianchi: a sinistra si osserva la sezione ottenuta dall'intersezione tra il piano e il volume, in alto centralmente si ha la maschera binaria relativa al profilo ellittico, in basso la maschera sulla quale si effettua il conteggio degli oggetti, a destra la sovrapposizione tra le due aree mettendo in risalto le differenze tra i due profili.

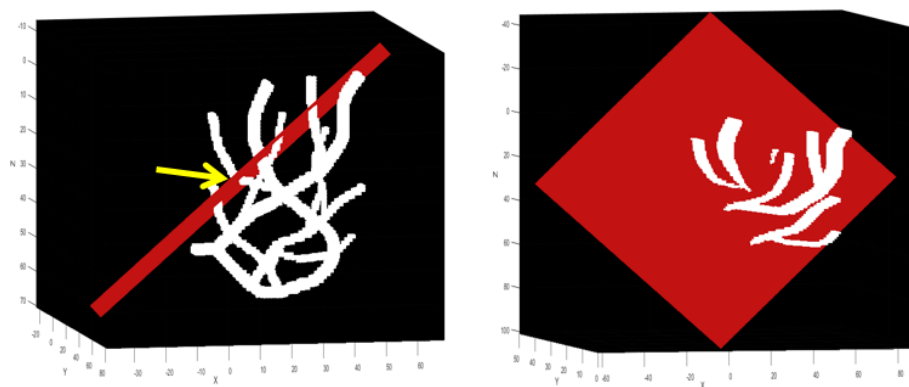


Figura 2.21: Visualizzazione dell'intersezione tra il piano e il volume relativo al modello complesso della Figura 2.1f. A sinistra è mostrato il piano con angolo di rotazione pari a 0° e la freccia gialla indica il vaso attorno al quale viene ruotato il piano. A destra si ha il piano ruotato di 60° , il quale porta ad individuare una sezione non ellittica, infatti non è il piano ortogonale alla direzione del vaso.

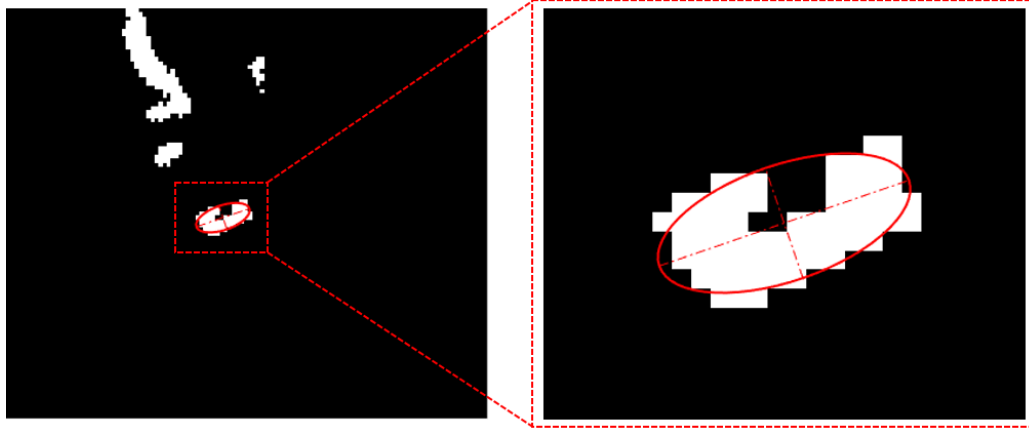


Figura 2.22: Analisi della sezione identificata sulla base della posizione del centro: a destra si ha l'ingrandimento della sezione di vaso all'interno della quale si osserva che in corrispondenza del centro del profilo non si ha un pixel bianco.

- In ultimo, per le sezioni che superano i controlli precedenti, viene valutata la **compactness** e la **varianza** del raggio del profilo ellittico. La prima è definita dall'equazione 2.7 mentre la seconda dalla 2.8.

$$C = \frac{P^2}{A} \quad (2.7)$$

$$V = \frac{1}{N-1} \sum_{i=1}^N |R_i - \mu|^2 \quad (2.8)$$

dove μ è il valor medio di R :

$$\mu = \frac{1}{N} \sum_{i=1}^N R_i \quad (2.9)$$

La compactness indica quanto un contorno è circolare: nella Figura 2.23 è mostrato come il valore della compactness cresce più il profilo diventa simile ad un cerchio. La varianza del raggio è calcolata come la varianza della distanza tra il centro e i punti del contorno della sezione. Si sono valutati entrambi i parametri per tutte le sezioni rimanenti e sono stati identificati gli outliers. Se la sezione in questione è outlier per almeno uno dei due indicatori, viene eliminata.

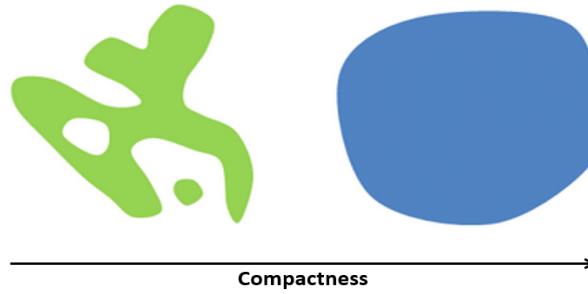


Figura 2.23: Variazione compactness [13].

Nelle Figure 2.24-2.29 viene mostrato un caso reale di ricerca del piano ortogonale alla direzione del vaso: è stata presa in considerazione una ROI dai volumi in vivo analizzati in questa tesi.

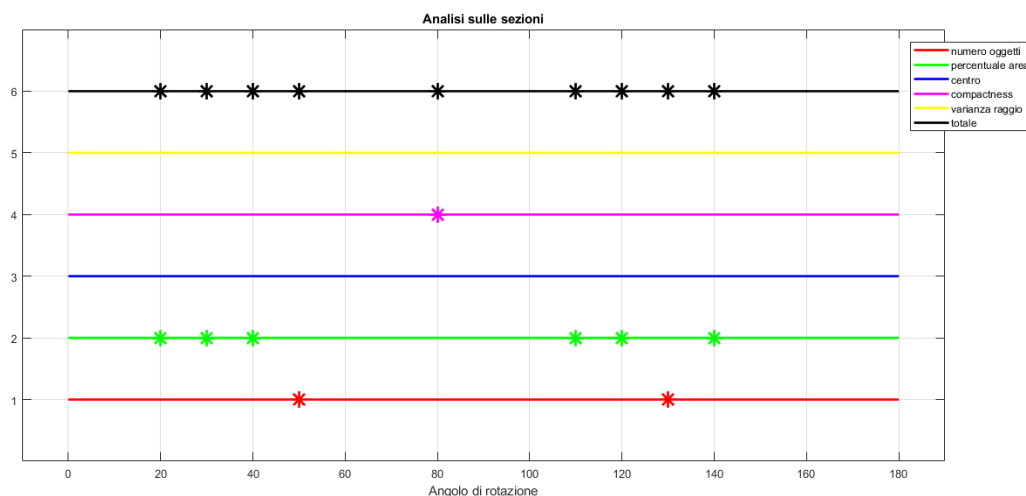


Figura 2.24: Esempio reale di analisi sulle sezioni: con l'asterisco sono indicate le sezioni eliminate per il numero di oggetti in rosso, per la percentuale di pixel bianchi in verde, per la posizione del centro in blu, per la compactness in magenta, per la varianza del raggio in giallo.

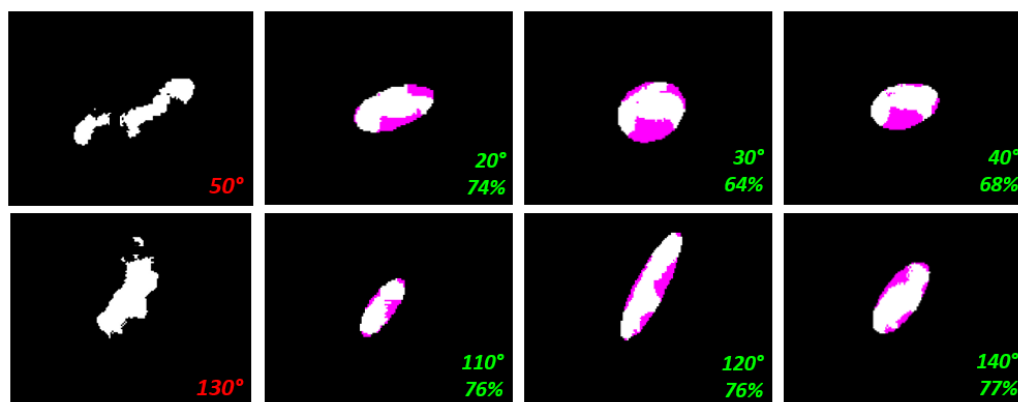


Figura 2.25: Nella prima colonna a sinistra sono mostrate le sezioni eliminate per il numero di oggetti e il relativo angolo di rotazione. Le restanti colonne indicano quelle eliminate per la percentuale dei pixel bianchi. Entrambe fanno riferimento alla Figura 2.24.

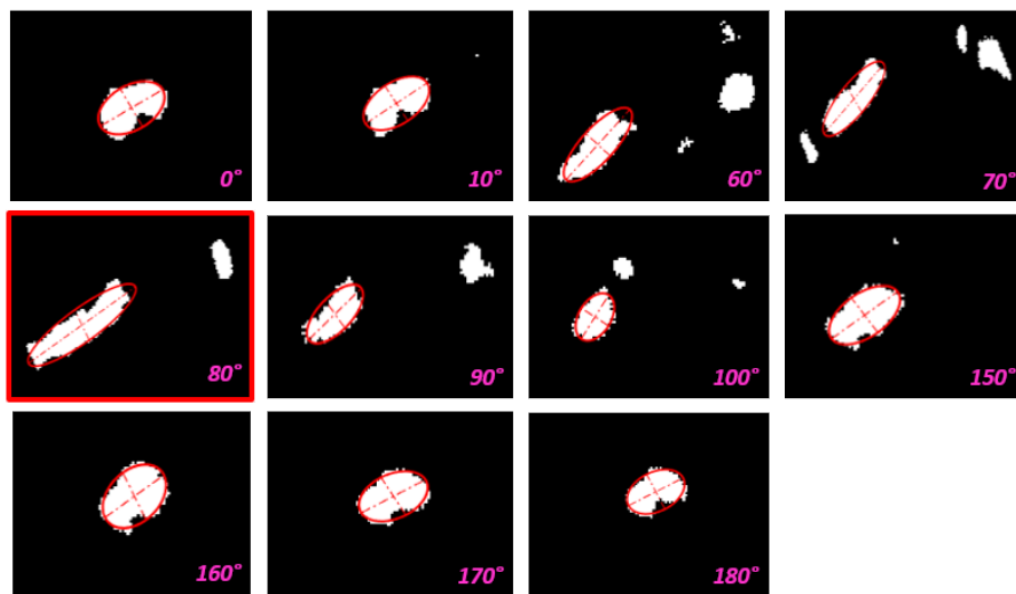


Figura 2.26: Facendo riferimento alla Figura 2.24, sono mostrate le sezioni che hanno superato le verifiche sul numero di oggetti, sulla percentuale di pixel bianchi e sulla posizione del centro. É possibile riscontrare un accordo tra queste figure e l'outlier identificato nella Figura 2.27: la sezione corrispondente ad un angolo di rotazione di 80° non risulta essere compatta ed è molto allungata rispetto alle altre.

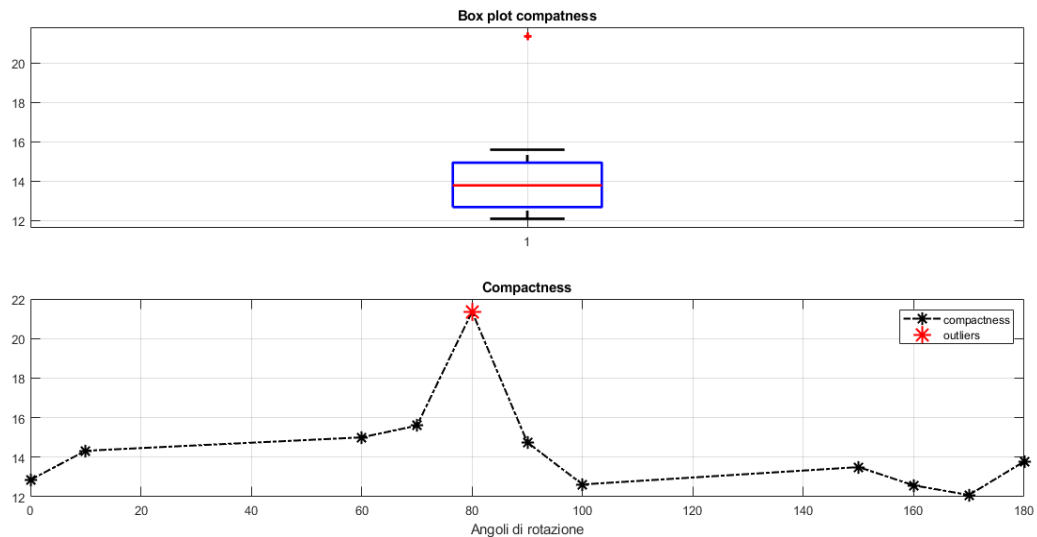


Figura 2.27: Esempio di valutazione della compactness: viene identificato un outlier, dunque la sezione corrispondente all'angolo di rotazione 80 viene eliminata. In nero con l'asterisco sono indicati i valori di compactness delle sezioni non scartate nei controlli precedenti.

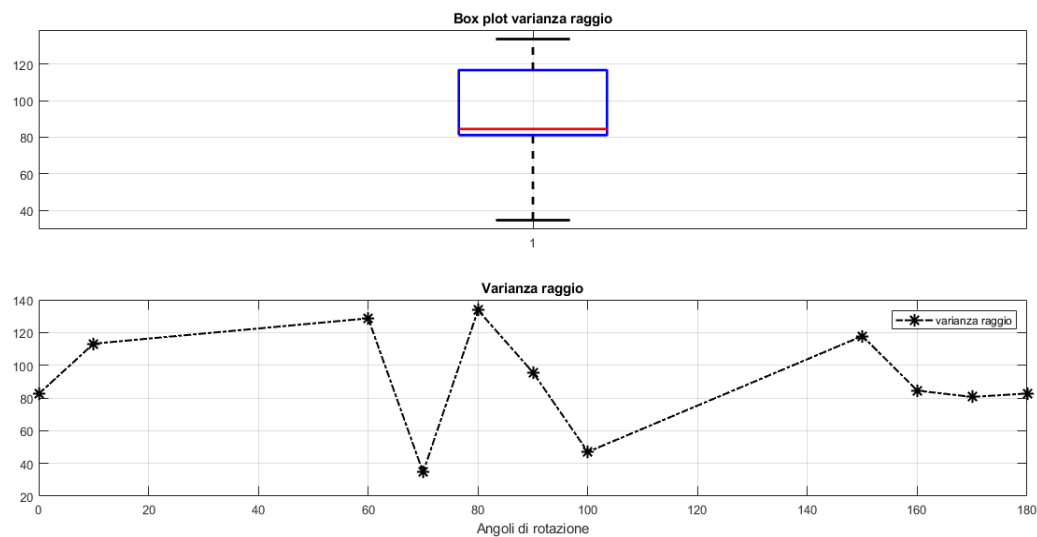
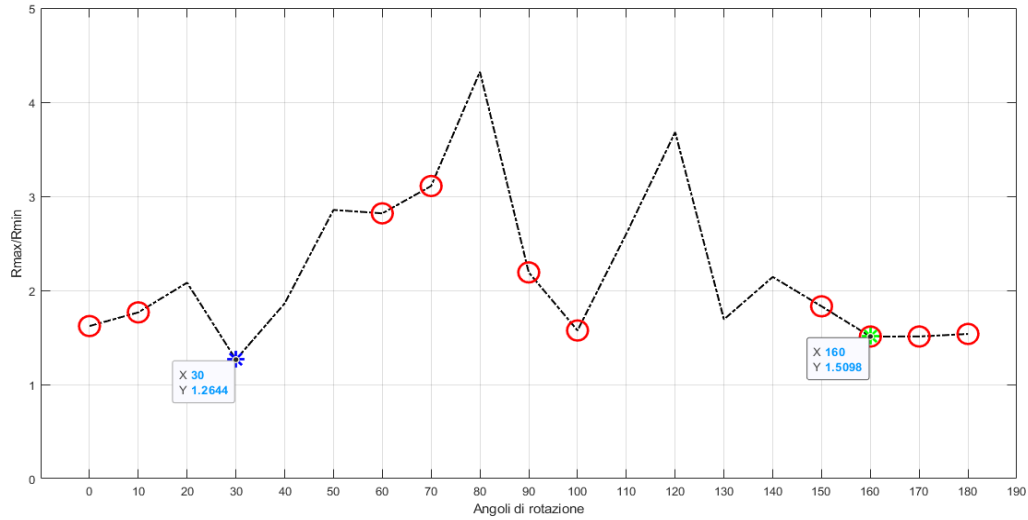
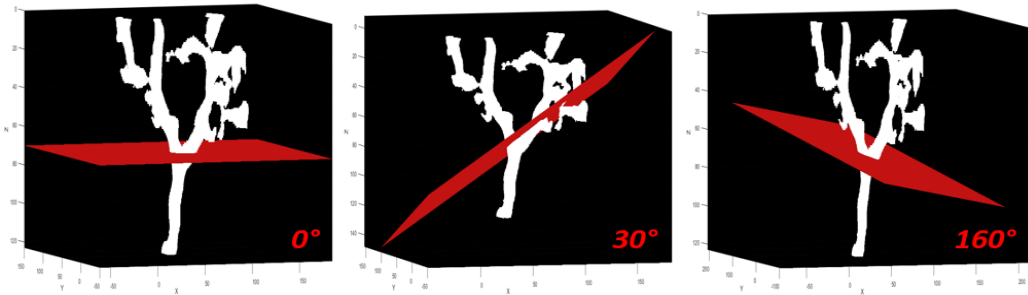


Figura 2.28: Esempio di valutazione della varianza del raggio: in questo caso non sono stati identificati outliers. In nero con l'asterisco sono indicati i valori della varianza del raggio delle sezioni non scartate nei controlli precedenti.



(a)



(b)

Figura 2.29: (a) É possibile visualizzare il valore del rapporto tra il raggio massimo e il raggio minimo per ogni angolo di rotazione: i cerchi rossi indicano le sezioni che hanno superato positivamente *l'analisi sulla sezione* e l'asterisco verde indica la sezione vincitrice. In blu è indicata quella che avrebbe vinto basandosi solo sul rapporto tra raggio massimo e minimo ma quella sezione è stata eliminata per la percentuale di pixel bianchi (Figura 2.25). (b) Mostra tre situazioni 3D: a sinistra il piano di partenza con angolo di rotazione pari a 0° , al centro a 30° e a destra a 160° . Si osserva che la situazione centrale sarebbe stata la scelta sbagliata quindi l'analisi sulle sezioni è utile per la scelta corretta del piano ortogonale alla direzione locale del vaso.

Gestione delle biforcazioni e inizio-fine vaso

Scelto il piano ortogonale, viene costruita una sfera in corrispondenza della sezione del vaso, centrata nel centro dell'ellisse. Il raggio della sfera è definito nel seguente modo:

$$R_{SFERA} = c\sqrt{R} \quad (2.10)$$

dove R è definito da:

$$R = \text{mediana}(R_{medio_i}, R_{medio_{i-1}}, R_{medio_{i-2}}) \quad (2.11)$$

$$R_{medio_i} = \frac{R_{max_i} + R_{min_i}}{2} \quad (2.12)$$

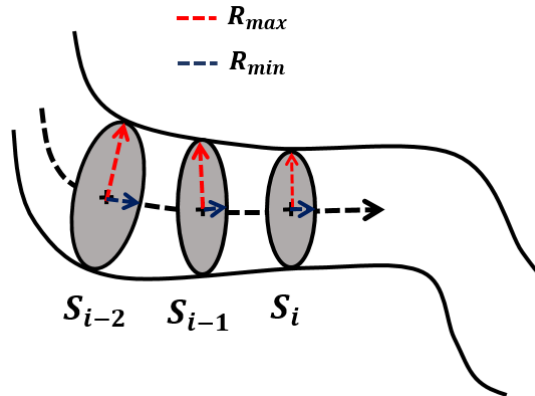


Figura 2.30: Calcolo raggio della sfera

Si calcola il valor medio tra il raggio massimo e minimo associato alla sezione corrente e alle prime due precedenti. Poi viene calcolata la mediana tra questi tre valori: si effettua questa scelta poichè se si hanno brusche va-

riazione del raggio, la mediana ne risulta influenzata in minor modo rispetto al valor medio. Inoltre, viene considerata la radice quadrata di questo valore per attenuare ulteriormente queste variazioni ed evitare di costruire una sfera eccessivamente grande dove non necessario. La costante moltiplicativa c è necessaria per ottenere un sfera che inglobi il vaso ad ogni step di tracking ovvero il raggio della sfera deve adattarsi alla variazione di diametro del vaso. Gli alberi vascolari nei modelli simulati sono caratterizzati da vasi con diametro nel range 2-8 pixel mentre nei modelli reali va da 8 a 14 pixel, dunque si è deciso di scegliere valori variabili per c in base al valore del diametro. In questo modo si è creata una sfera che si adatta in modo attivo al raggio locale del vaso. La sfera in questione è definita da un numero di punti pari a 120 e si è scelto questo valore perchè dopo diverse prove si è osservato essere il migliore.

diámetro	c
$2 \leq d < 4$	3.7
$4 \leq d < 7$	3.8
$7 \leq d < 11$	4.3
$11 \leq d \leq 14$	5.3

Tabella 2.4: Scelta della costante moltiplicativa per definire il raggio della sfera.

Una volta definita la sfera, si valuta l'intensità media del volume lungo ogni raggio, ottenendo una mappa in scala di grigi dove le zone più chiare si hanno in corrispondenza dei vasi (Figura 2.31). Su quest'immagine è stata poi applicata gamma correction per modificarne il contrasto attraverso la funzione *imadjust* di Matlab. La gamma correction è definita come

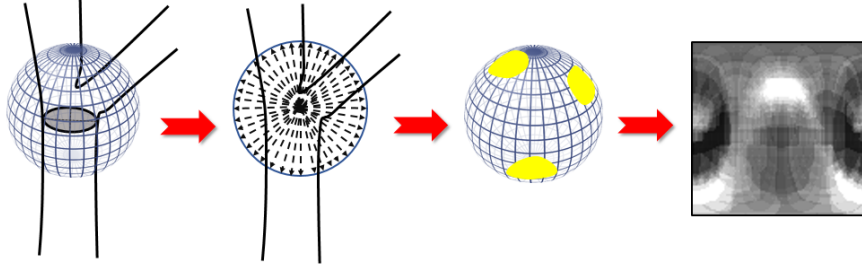


Figura 2.31: Sono mostrati gli step seguiti per il calcolo dell'intensità media: costruzione sfera con raggio e numero di punti scelti, costruzione raggi della sfera (vista 2D), visualizzazione 3D delle zone più chiare indicate in giallo, immagine grayscale che riporta l'intensità media del volume lungo ogni raggio in 2D.

l'applicazione di γ come esponenziale dell'immagine di input:

$$I_{output} = (I_{input})^{\gamma} \quad (2.13)$$

Ponendo $\gamma > 1$, si aumenta il contrasto e più alto è il valore di γ più il contrasto viene migliorato. Se $\gamma < 1$, il contrasto viene diminuito. Nella Figura 2.32 è possibile osservare l'effetto su un'immagine grayscale usando diversi valori di γ .

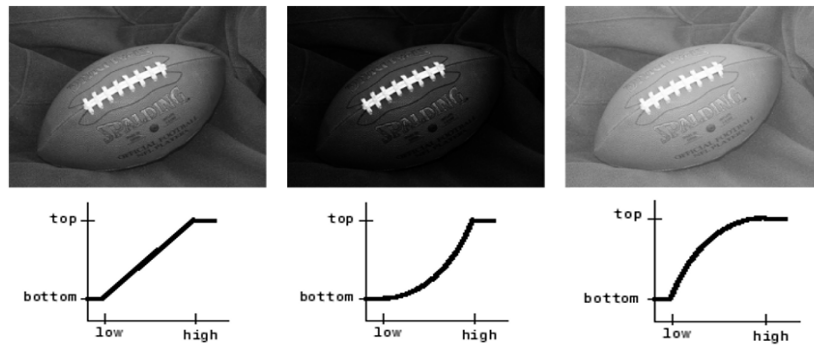


Figura 2.32: La prima colonna fa riferimento ad una $\gamma = 1$, la seconda $\gamma = 2.2$, la terza $\gamma = 0.5$ [14].

In questo lavoro di tesi é stato scelto un valore di $\gamma = 4$ poiché si vuole aumentare il contrasto, mettendo in risalto il più possibile le zone chiare in corrispondenza dei vasi. Nella Figura 2.33 è mostrata la relazione che c'è tra l'immagine di input e quella di output: in particolare è stato scelto un valore di h calcolato secondo l'equazione 2.14:

$$h = v + 0.6(M - v) \quad (2.14)$$

dove v indica il valor medio dell'intensità dei pixel diversi da zero mentre M il valore massimo.

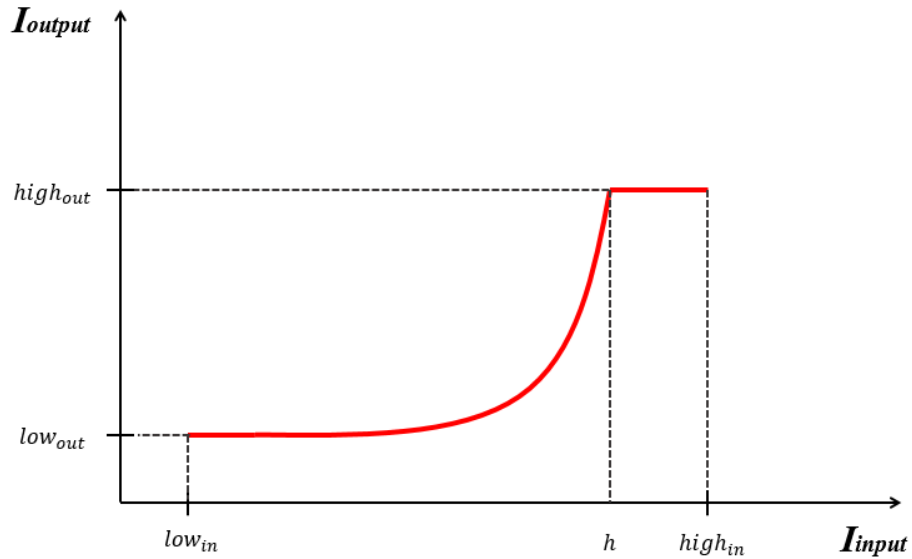


Figura 2.33: É stata applicata gamma correction in modo da portare al valore indicato da $high_{out}$ tutti i pixel con intensità superiore ad h , data dall'equazione 2.14.

Si è scelto di modificare il contrasto in questo modo per sopperire ad alcune problematiche che incorrono nel momento in cui la sfera traccia vasi

curvi. In questo caso non è possibile identificare in modo netto zone con alta intensità tali da applicare un successivo thresholding per segmentarle. Questo è il caso mostrato nella Figura 2.34 dove c'è un evidente caso di vaso curvo, la relativa immagine che riporta l'intensità media lungo i raggi della sfera e quella che ha subito gamma correction.

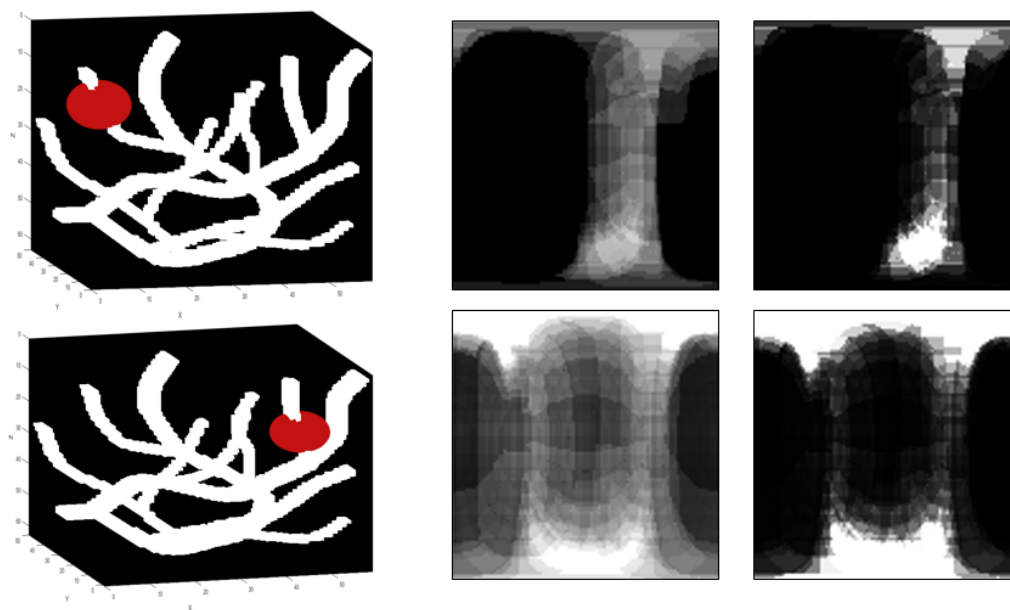


Figura 2.34: In alto è presente un esempio di vaso curvo: nell'immagine che riporta l'intensità media (prima riga, seconda colonna), non si riescono ad identificare zone con alta intensità. In basso è presente un caso di vaso rettilineo: la mappa dell'intensità media permette di identificare zone molto chiare. L'ultima colonna riporta l'immagine che ha subito gamma correction.

Sull'immagine che ha subito gamma correction, viene applicato un thresholding con soglia fissa pari a 0.97 in modo da ottenere una maschera binaria che ha 1 solo in corrispondenza del vaso. Viene applicata anche apertura con elemento strutturale quadrato di lato 3 pixel. Nella Figura 2.35 sono mo-

strati gli step che si eseguono a partire dall'immagine che riporta l'intensità media fino alla maschera ottenuta dopo il thresholding.

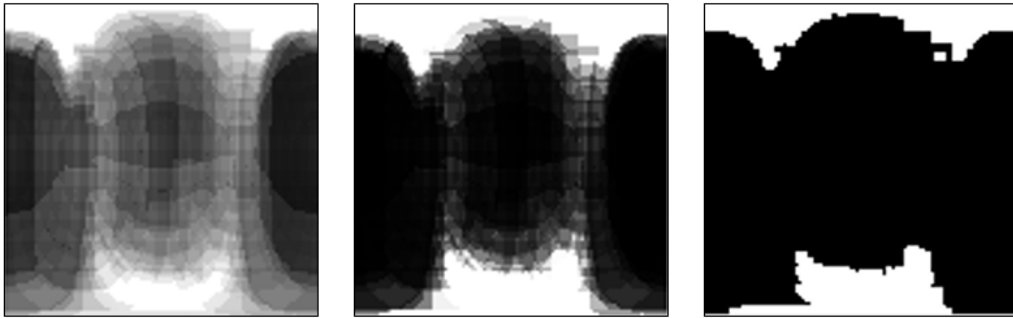


Figura 2.35: Da sinistra a destra: intensità media lungo i raggi della sfera, gamma correction, thresholding.

Tuttavia, l'approccio seguito porta a perdere in alcuni casi i vasi di piccolo diametro e quelli caratterizzati da alta curvatura. Si é quindi calcolata una maschera binaria che riporta l'intersezione tra la superficie della sfera e il volume. Su questa maschera si è applicata apertura con elemento strutturale di forma quadrata e lato pari a 2 pixel. Nella Figura 2.36 sono sintetizzati gli step per la corretta identificazione dei rami persi. In particolare, tra gli elementi identificati nella maschera ottenuta dall'intersezione tra il volume e la superficie della sfera, bisogna considera solo quelli che non sono contenuti nella maschera ottenuta dal thresholding sulla maschera di intensità media che ha subito gamma correction. Se c'è corrispondenza di almeno un pixel bianco tra le due maschere, quell'oggetto non viene considerato. Inoltre è necessario che in corrispondenza di quest'oggetto ci sia un massimo locale. In quel caso l'oggetto viene salvato come oggetto perso.

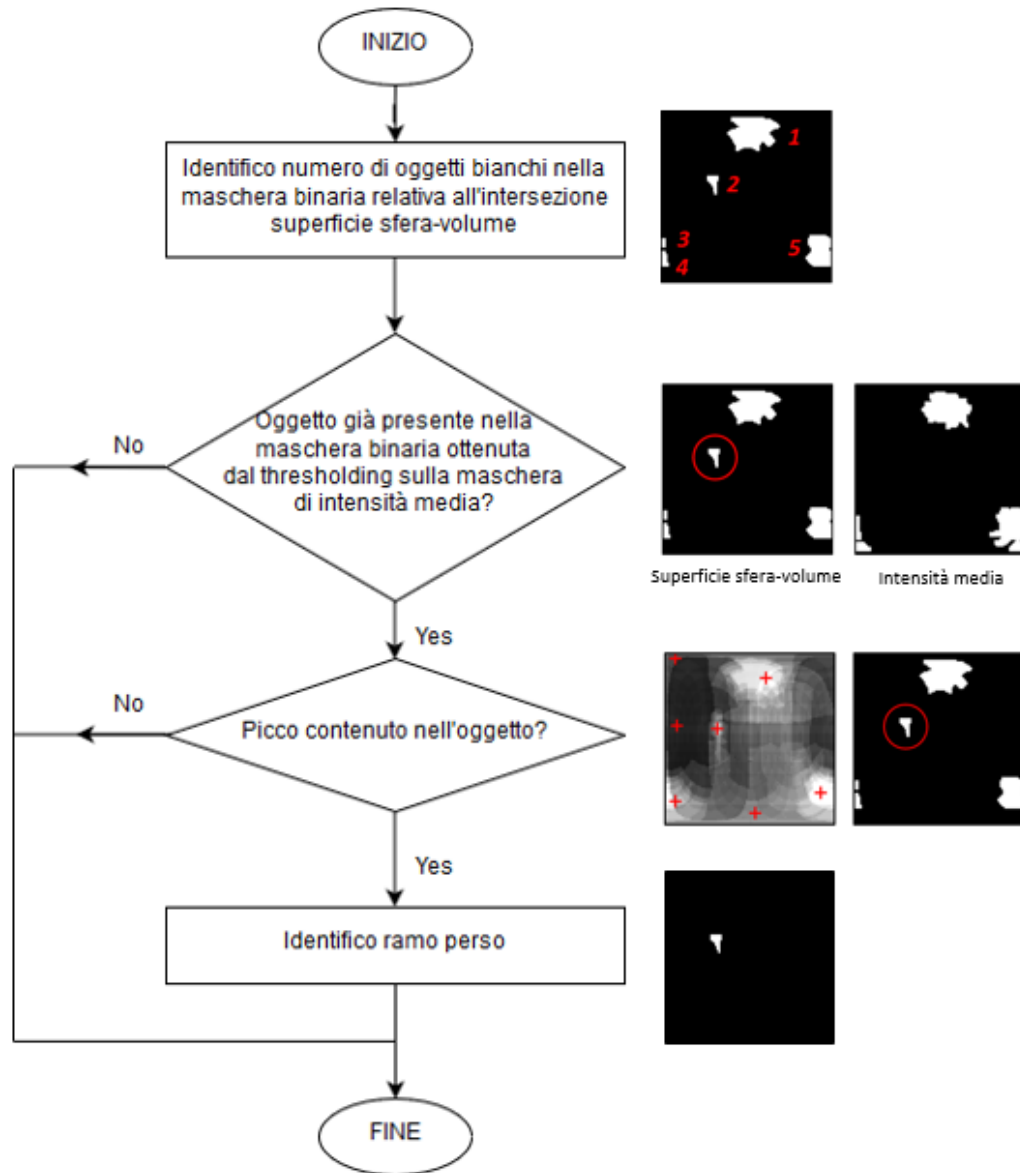


Figura 2.36: Step per identificare correttamente i rami persi: si identificano gli oggetti e li si confrontano con quelli presenti nella maschera di intensità media che ha subito gamma correction e thresholding. Se ci sono oggetti non contenuti nella seconda maschera e se si ha una corrispondenza tra questi oggetti e i massimi locali, viene identificato il ramo perso.

Nella Figura 2.37 sono riportati due esempi relativi al modello *profondo* (Figura 2.1g):

- In *(a)* è riportato lo stesso caso preso in esame nella Figura 2.36, viene mostrata l'immagine volumetrica e poi sei immagini che riportano a partire dalla prima riga, l'intensità media del volume lungo i raggi della sfera, gamma correction, thresholding e nella seconda riga, l'intensità media del volume con i massimi locali, la maschera binaria data dall'intersezione tra la superficie della sfera e il volume e la maschera che riporta i rami persi.
- Nel caso *(b)* è preso in considerazione un caso di assenza di biforcazione infatti l'immagine che riporta l'intensità media del volume identifica correttamente i rami, tuttavia quella data dall'intersezione della superficie e il volume riporta due oggetti che non sono classificati come rami persi perchè non si ha un massimo locale in corrispondenza di essi.

Quindi, sebbene la maschera ottenuta come intersezione tra la superficie della sfera e il volume aiuta ad affrontare il problema della perdita di vasi con piccolo diametro o particolarmente curvi, è necessario considerare anche l'identificazione dei massimi locali nella maschera dell'intensità media poichè se non c'è un massimo locale, gli oggetti presenti nella seconda maschera sono solo rumore dovuto a come la sfera è posizionata e non alla reale presenza di un ramo.

La Figura 2.38 riporta tutti gli step che sono stati descritti nel dettaglio precedentemente.

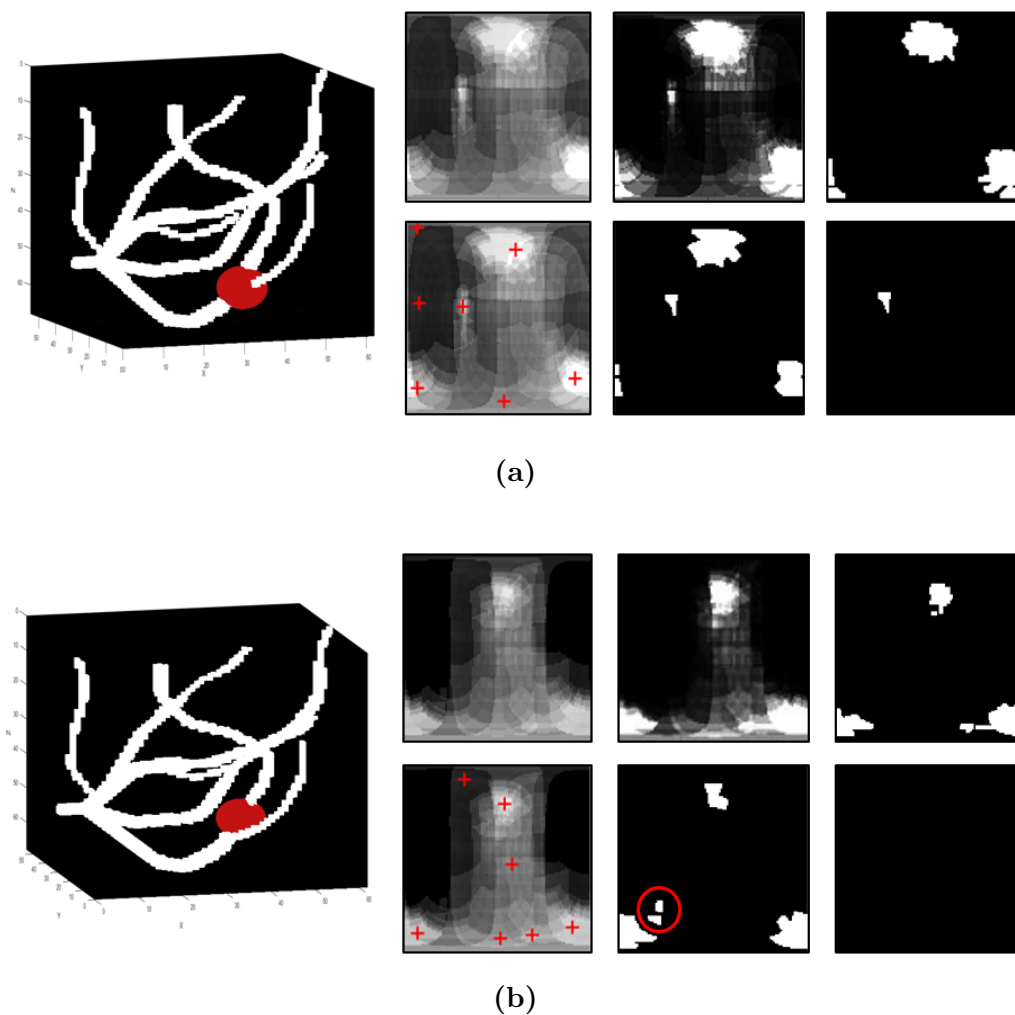


Figura 2.37: (a) La sfera è in corrispondenza di una biforcazione composta da tre rami: viene identificato correttamente il ramo perso. (b) La sfera ha superato la biforcazione: in rosso sono cerchiati due oggetti che non sono presenti nella maschera dell'intensità media che ha subito gamma correction e thresholding. Non c'è corrispondenza con nessun punto di massimo: non vengono considerati come rami persi.

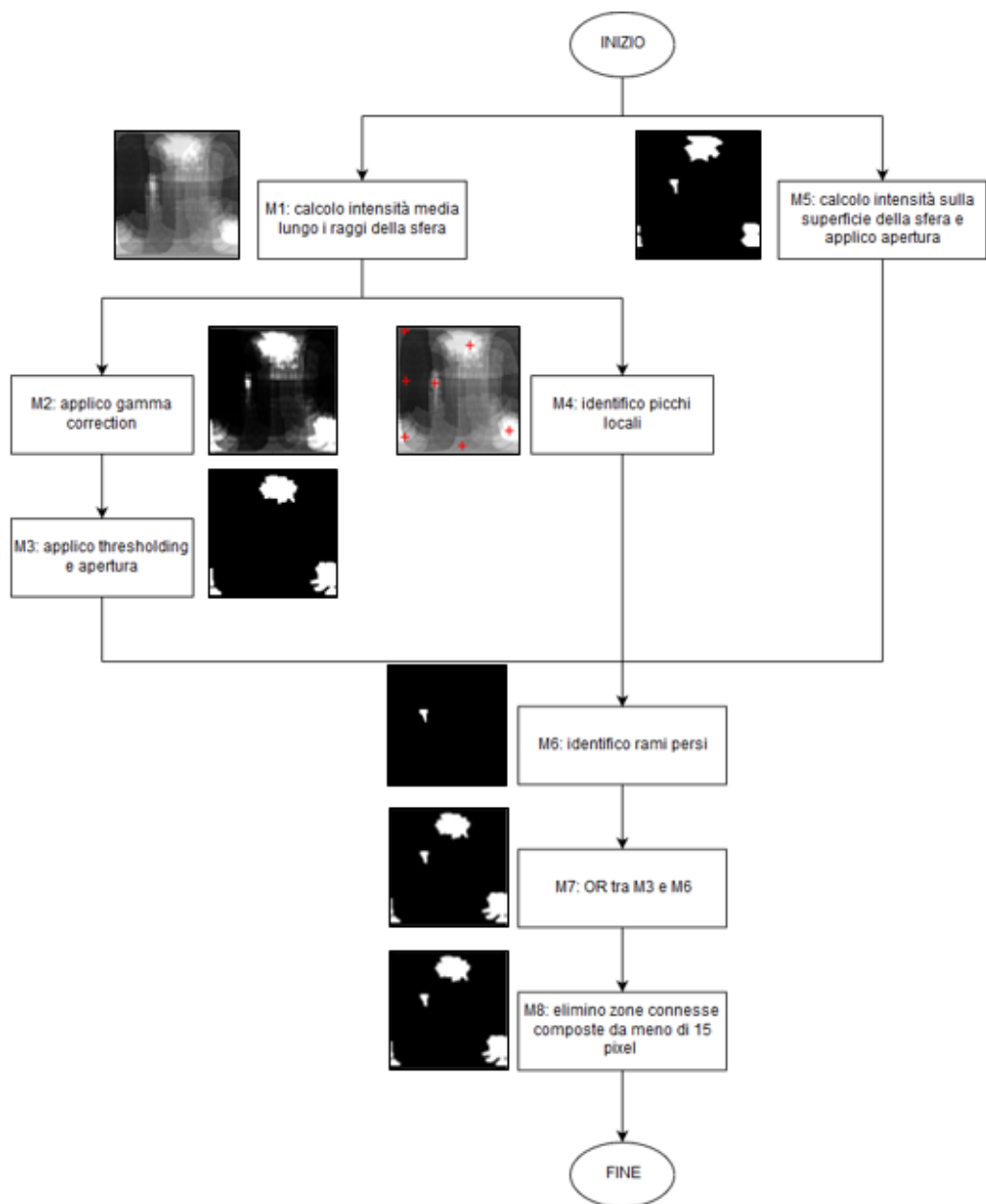


Figura 2.38: Step totali seguiti a partire dalla costruzione della sfera fino ad ottenere la maschera binaria finale.

La maschera binaria ottenuta alla fine di questo processo non è altro che una visualizzazione 2D di una mappa sferica quindi le coordinate di ciascun pixel bianco possono essere ricondotte alle coordinate della superficie sferica come può essere osservato nella Figura 2.39. Sono calcolati i vettori che congiungono il centro della sfera con tutti i punti sulla superficie in corrispondenza delle zone bianche e se ne calcola il versore in modo da avere solo un'informazione sulla direzione e verso.

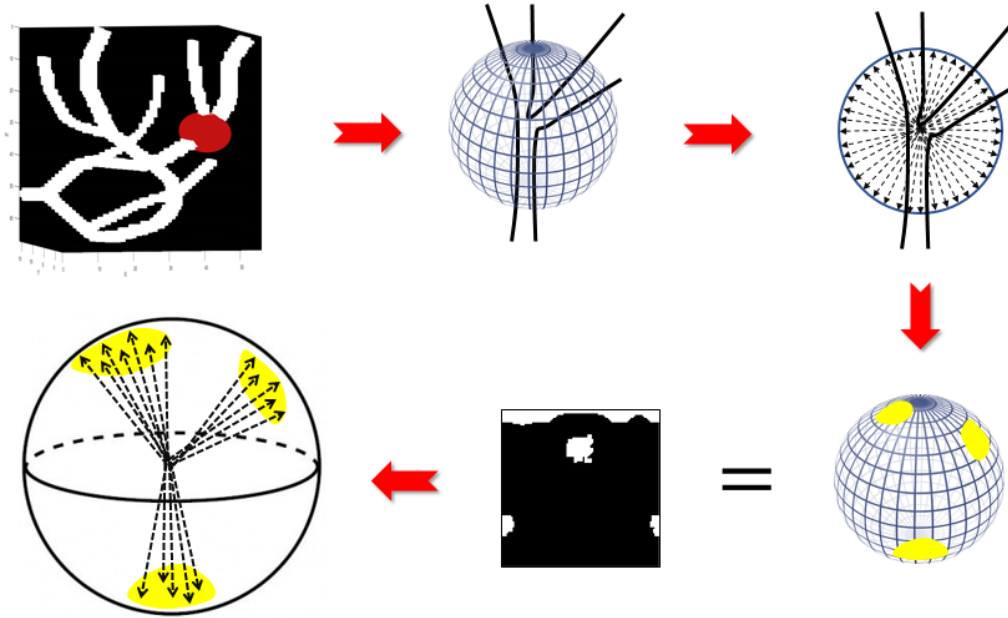


Figura 2.39: Step seguiti per ottenere la maschera binaria finale e calcolo dei versori in corrispondenza delle zone bianche.

Questi versori sono sottoposti al metodo di clustering DBSCAN per individuare il corretto numero di cluster e quindi direzioni possibili da seguire nel processo di tracking. Il DBSCAN (Density-Based Spatial Clustering of Ap-

plications with Noise) è un metodo di clustering basato sulla densità poiché connette regioni di punti con densità sufficientemente alta. È necessario settare due parametri:

- **Epsilon**: è la misura della distanza in base alla quale viene definita la somiglianza tra i punti. Due punti appartengono allo stesso cluster se si trovano a una distanza massima pari a *Epsilon*;
- **Minpts**: indica il numero minimo di punti richiesti per formare un cluster.

Nel DBSCAN, un punto può essere di tre tipi:

- **Core point**, che ha almeno *Minpts* punti nel suo vicinato;
- **Border point**, che ha un core point nel suo vicinato;
- **Noise point**, non è né un core né un border point ed è considerato un outlier nel set di dati.

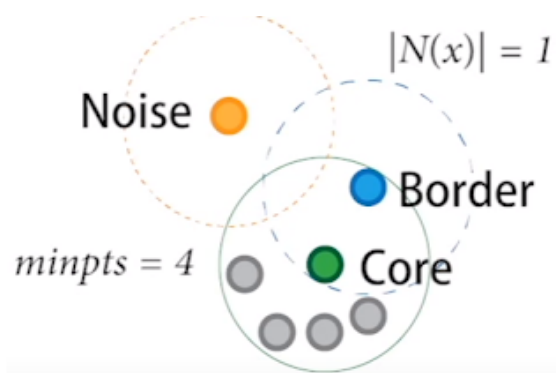


Figura 2.40: Classificazione dei punti nell'algoritmo del DBSCAN [15].

L'algoritmo è strutturato nel seguente modo:

1. Dal set di dati di input, si seleziona un punto in modo random tra quelli senza etichetta e si inizializza la prima etichetta dei cluster C ad 1 (cluster 1).
2. Si esplora il vicinato del punto corrente: se il numero di vicini è inferiore a $Minpts$, si etichetta il punto corrente come rumore e si va al punto 4. Altrimenti, si etichetta il punto corrente come **core point** del cluster C .
3. Si itera su ciascun punto del vicinato (nuovi punti correnti) e si ripete il punto 2 fino a quando non vengono trovati nuovi vicini che possono essere etichettati come appartenenti al cluster corrente C .
4. Si seleziona un altro punto non etichettato e si aumenta il numero di cluster di 1.
5. Si ripetono gli step 2-4 fino a quando tutti i punti sono stati etichettati.

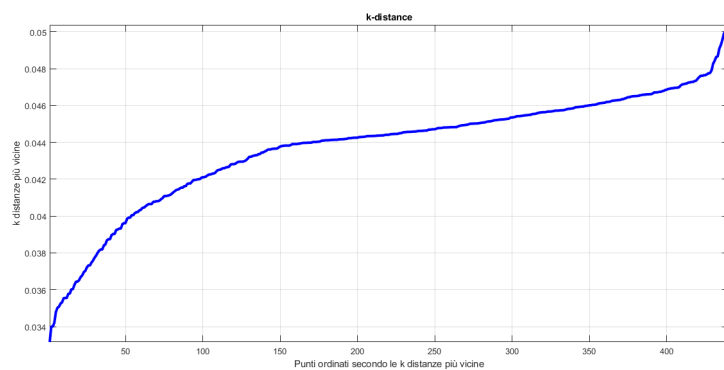
Diversamente dal k-means [16], l'algoritmo DBSCAN non richiede una conoscenza preliminare sul numero di cluster e i cluster non devono essere necessariamente di forma sferoidale. Ciò è molto utile in questo lavoro di tesi poiché non si ha la conoscenza di quanti siano i rami da identificare ad ogni step di tracking e inoltre nelle situazioni più complesse ci si può trovare di fronte ad alcuni casi di cluster di forma non perfettamente circolare. Tra gli svantaggi di questo metodo c'è senza dubbio la scelta di epsilon: un valore

troppo piccolo può portare a far aumentare il numero di outliers mentre un valore troppo grande porta a fondere insieme i cluster.

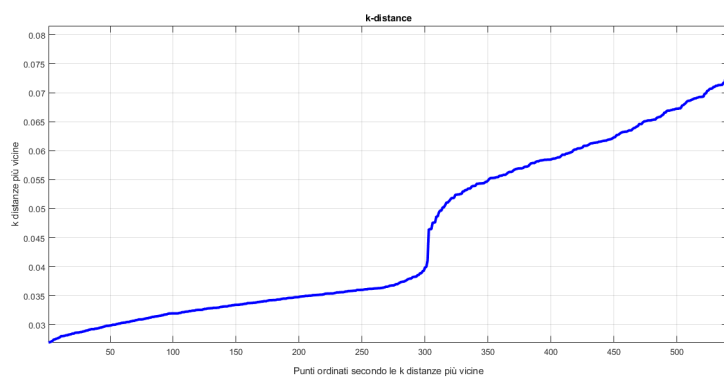
Non esiste un modo generale per scegliere i parametri *Minpts* ed *Epsilon*. In genere *Minpts* viene settato pari ad un valore maggiore o uguale della dimensione dei dati di input più uno: se ad esempio i dati di input sono tridimensionali quindi ogni punto è caratterizzato da tre variabili, *Minpts* può essere settato ad un valore maggiore o uguale a quattro. In questo lavoro di tesi, applicando il DBSCAN ai versori (di cui si sono descritte sopra le modalità di calcolo), si è scelto di settare $Minpts = 4$. Per quanto riguarda la scelta di *Epsilon*, una strategia per stimarlo consiste nel generare il grafico della *k-distance* per i dati di input. Per ogni punto si trova la distanza dai k punti più vicini e si traccia il grafico con questi punti ordinati rispetto alla distanza. Il grafico contiene un ginocchio e la distanza in corrispondenza di questo generalmente è una buona scelta per *Epsilon* perché da quella distanza in poi i punti iniziano a diventare outliers.

La distanza viene calcolata usando come metrica la distanza euclidea, cercando le k distanze più piccole dove k è pari a *Minpts*. Poiché non è sempre stato possibile identificare il ginocchio o uno solo, si è scelto di settare $Epsilon = 0.1$, valore che ha portato a buoni risultati di clusterizzazione.

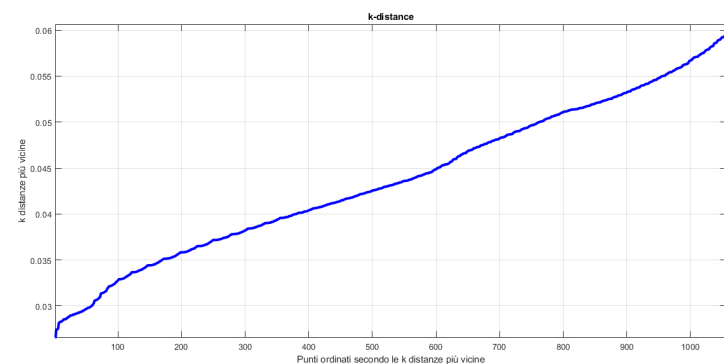
Nella Figura 2.41 sono mostrati tre casi reali del grafico della *k-distance* che sull'asse delle ordinate riporta le k distanze più vicine e su quello delle ascisse i punti ordinati rispetto a questa distanza.



(a)



(b)



(c)

Figura 2.41: (a) Il ginocchio è ben visibile in corrispondenza di una distanza pari a 0.048. (b) Si possono osservare due ginocchi a 0.04 e 0.073. (c) Non si riesce a identificare un ginocchio.

Settati i parametri per l'algoritmo di clusterizzazione, si ottengono i cluster: nella Figura 2.42 è possibile osservare in modo schematico il processo di calcolo dei vettori, la costruzione del grafico della k -distance e una volta settati i parametri, l'applicazione del DBSCAN ottenendo i cluster.

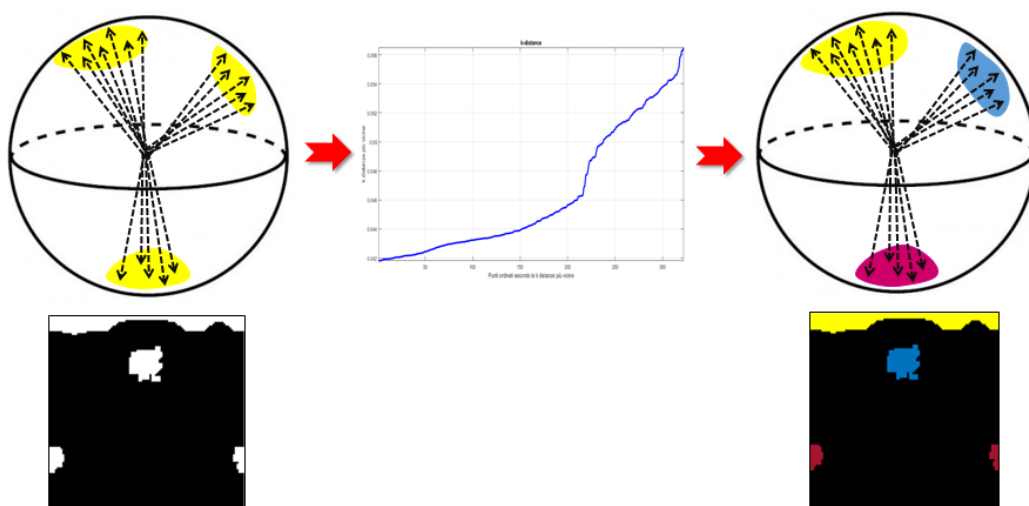


Figura 2.42: Schema per l'identificazione dei cluster.

Tuttavia per avere una direzione unica di tracking, tra quelle appartenenti a ciascun cluster se ne calcola il centroide come il valor medio dei vettori. Per ogni centroide si costruisce il piano che ha come normale il centroide stesso ed è tangente alla superficie della sfera stessa. I piani in questione servono per portare avanti il tracking su ciascun ramo.

Si può incorrere in over-clustering ottenendo un numero di cluster superiore al numero corretto. Per far fronte a questa problematica, si calcola l'angolo tra ciascun centroide e se questo risulta essere inferiore a 30° , si ha

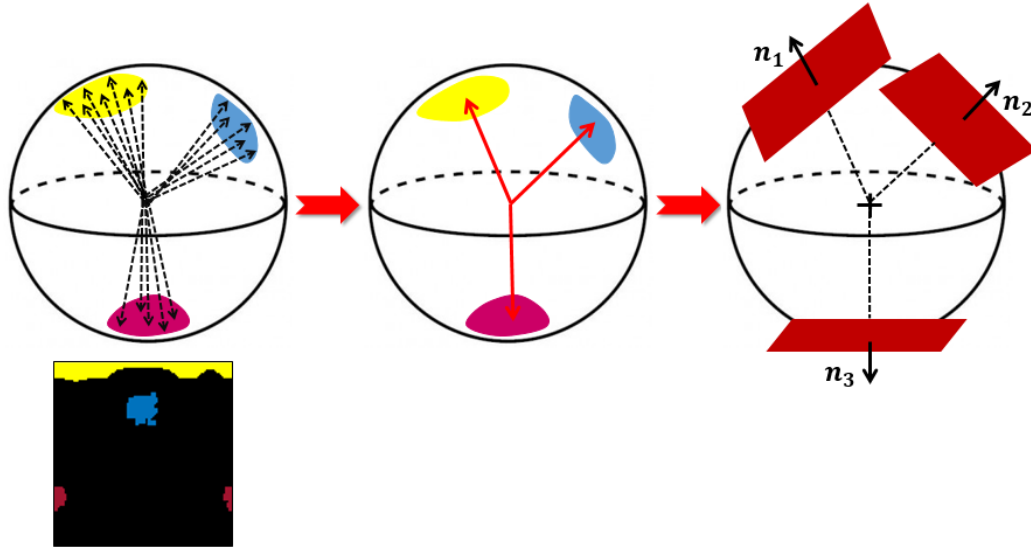


Figura 2.43: Direzioni clusterizzate, calcolo centroidi, costruzione piani.

l'unione dei cluster e conseguente ricalcolo del centroide (Figura 2.44). Per l'unione dei cluster si è fatta la seguente scelta: si confrontano i centroidi dei cluster considerati a coppie e se c'è almeno un cluster in comune, l'unione avviene tra tutti gli elementi delle coppie. Nella Figura 2.45 si può comprendere facilmente il processo di unione: α_4 e α_5 sono minori di 30° ma $\alpha_4 + \alpha_5$ è maggiore; in questo caso, vengono uniti tra loro il cluster arancio, celeste e verde poichè le coppie arancio-celeste e celeste-verde hanno in comune il cluster celeste, anche se tra i centroidi del cluster arancio e verde non si ha un angolo più piccolo della soglia posta. Si avrà anche l'unione dei cluster magenta e giallo essendo caratterizzati da un angolo α_2 minore di 30° .

Uniti i cluster, si calcolano nuovamente i centroidi e si ricalcolano i piani tangenti alla superficie.

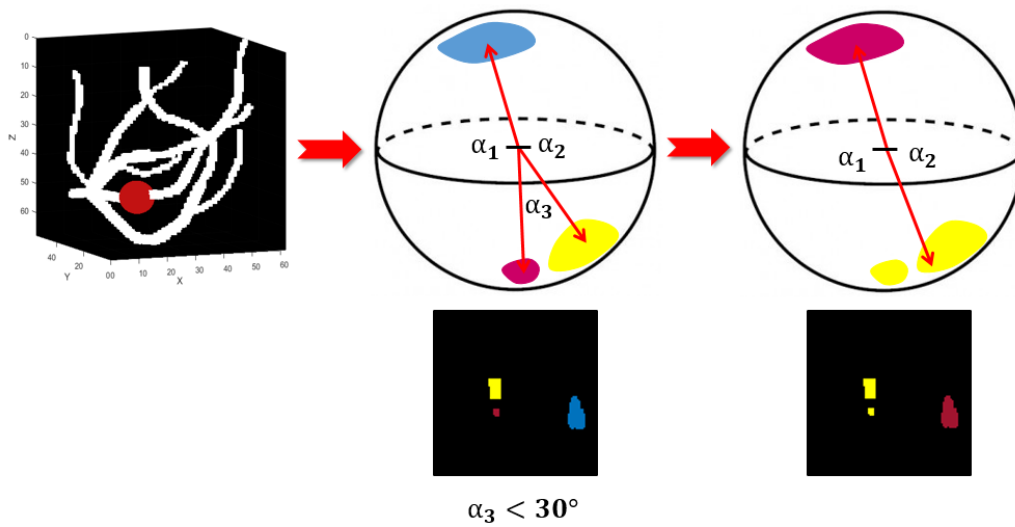


Figura 2.44: Unione cluster e ricalcolo centroidi

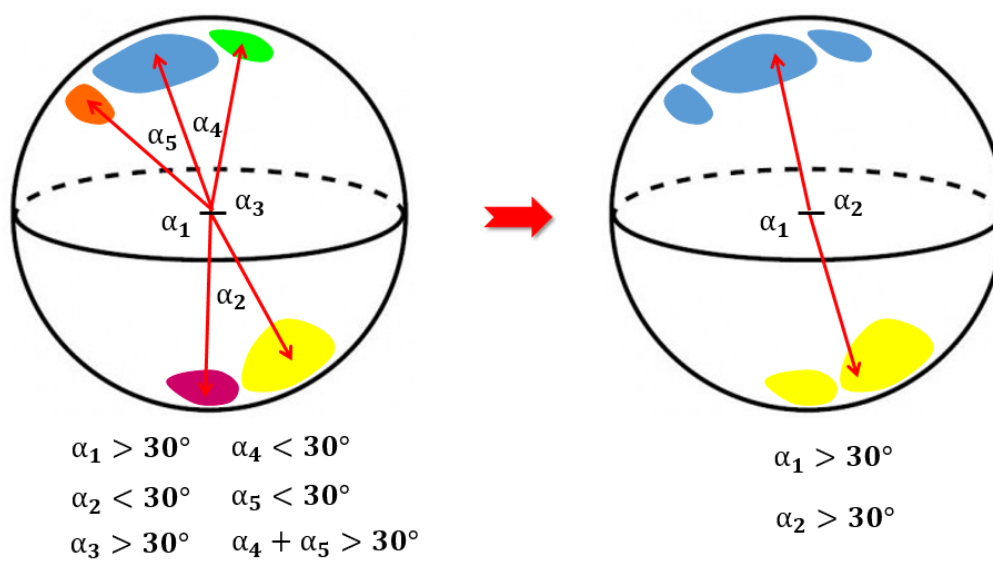


Figura 2.45: Unione cluster multipli.

Gestione dei cluster

Dopo il processo di clustering ed eventuale unione dei cluster, si possono presentare tre diverse situazioni:

- $Cluster \geq 3$, si ha l'identificazione di una biforcazione con un numero di rami pari al numero di cluster. Delle direzioni identificate va eliminata quella opposta alla direzione di tracking: ad esempio nella Figura 2.46, la direzione concorde con la normale n_3 non deve essere considerata. In seguito, viene descritto come determinare se una direzione è già stata tracciata.

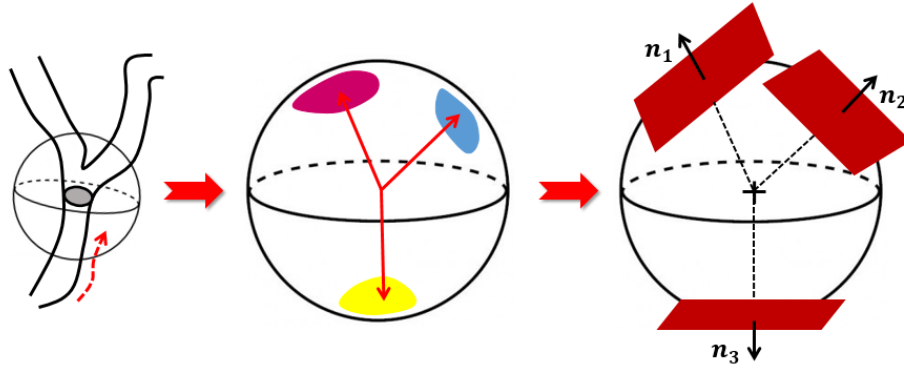


Figura 2.46: $Cluster \geq 3$: identificazione di una biforcazione che in questo caso ha 3 rami.

- $Cluster = 2$, non si sono identificate biforcazioni ma solo il cluster relativo alla direzione di tracking e quella opposta. La scelta tra le due viene effettuata calcolando l'angolo tra ogni centroide e la direzione di tracking dello step precedente e si sceglie quella che forma l'angolo minore. Tuttavia, in presenza di vasi con alta curvatura questa scelta

non è sempre la migliore: dunque viene calcolata la differenza tra i due angoli e se questa è più piccola di 30° la scelta della direzione come quella a cui è associata l'angolo più piccolo non è più quella migliore. In quel caso allora la scelta è effettuata considerando entrambe le direzioni come direzioni utili, di fatto considerando quel punto come uno di biforcazione e poi affidarsi al controllo del vaso già percorso.

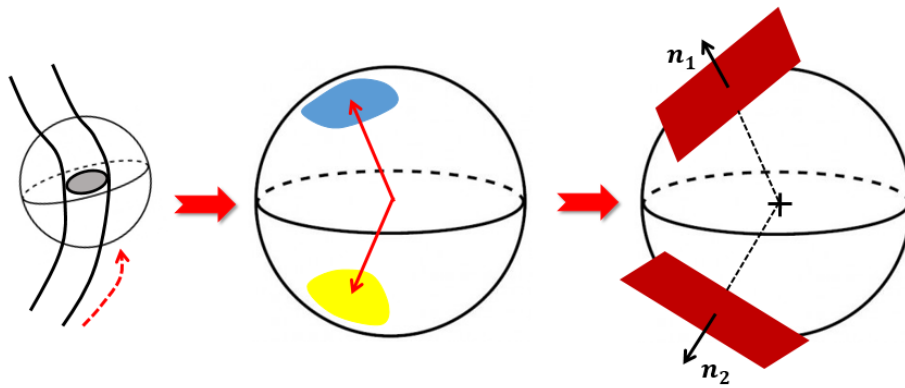


Figura 2.47: *Cluster* = 2: vaso singolo.

- *Cluster* = 1, si è nel caso di inizio o fine vaso. Se si è già tracciato parte del vaso, nel momento in cui si identifica un unico cluster, si è sicuramente alla fine del vaso. Se invece si è al primo step di tracking, quindi alla prima ricerca del piano ortogonale e alla prima costruzione della sfera, a seconda della direzione di tracking, si può essere alla fine del vaso o all'inizio. In questo caso, il piano viene traslato nella direzione di tracking: se l'intersezione tra questo e il vaso è nullo, si è effettivamente a fine vaso e si può procedere con il tracking nella direzione opposta se non è già stato fatto; se l'intersezione non è nulla, si procede con il tracking nella direzione corrente (Figura 2.48).

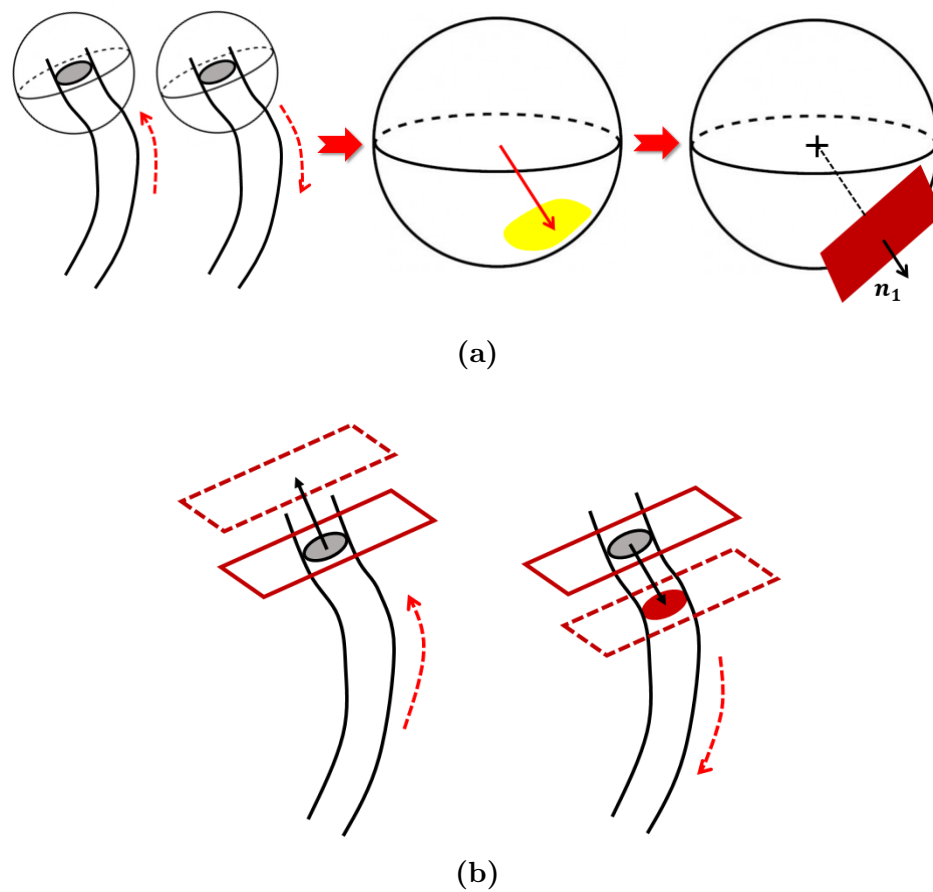


Figura 2.48: *Cluster* = 1: inizio o fine vaso. (a) Costruzione piano in corrispondenza del centroide identificato. (b) Traslazione piano per capire se si è ad inizio o fine vaso: a sinistra si è a fine vaso mentre a destra si ha un'intersezione non nulla tra piano traslato e volume, quindi si è ad inizio vaso.

Nella Figura 2.49 è mostrata schematicamente una biforcazione: in questo caso, ciascun ramo viene tracciato singolarmente ma è necessario verificare che il vaso che si sta seguendo, non sia stato già tracciato. Per ogni direzione viene identificato il primo punto che viene sottoposto alla verifica del vaso già percorso.

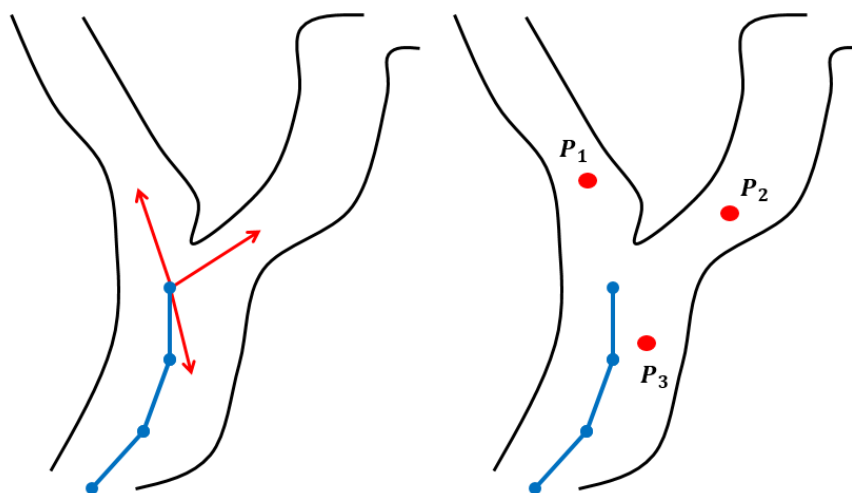


Figura 2.49: In blu è indicata la centerline già tracciata mentre in rosso i punti presi in esame.

Per poter verificare se si sta tracciando un percorso già seguito, si calcola la distanza tra il punto e ciascun segmento che congiunge i punti della centerline. La distanza è definita come la minima delle distanze tra il punto in questione e ciascun punto del segmento. Infatti dato un punto P e una generica retta r , si definisce la distanza punto-retta $d(P, r)$ come la minima distanza tra il punto in questione e tutti i punti che giacciono sulla retta.

$$d(P, r) = \min(d(P, Q)) \quad (2.15)$$

con $Q \in r$.

Queste distanze sono poi confrontate con il raggio medio locale associato al punto, calcolato come il valor medio tra il raggio massimo e minimo: se almeno una distanza è inferiore al raggio medio, il punto viene classificato come già tracciato, quindi il processo di tracking in quella direzione si arresta. Tuttavia se la scelta si basa unicamente sul valore della distanza si può incorrere in errore quando la proiezione del punto non giace sul segmento stesso: in quel caso può capitare che un punto sia classificato come già tracciato solo perchè la distanza è inferiore al raggio anche se il punto è lontano dal segmento in questione (Figura 2.50).

Dunque, si calcola la proiezione del punto sul segmento e si verifica che sia contenuta all'interno di questo: si calcola l'angolo formato tra i vettori che partono dalla proiezione del punto e puntano ognuno ad un estremo del segmento. Se l'angolo è nullo, la proiezione del punto è esterna quindi anche se la distanza risultasse essere inferiore al raggio, il punto non sarebbe classificato come già tracciato. Se invece la proiezione del punto è interna al segmento poichè l'angolo è pari a 180° , qualora la distanza dovesse essere inferiore al raggio, il punto verrebbe classificato come già tracciato (Figura 2.51).

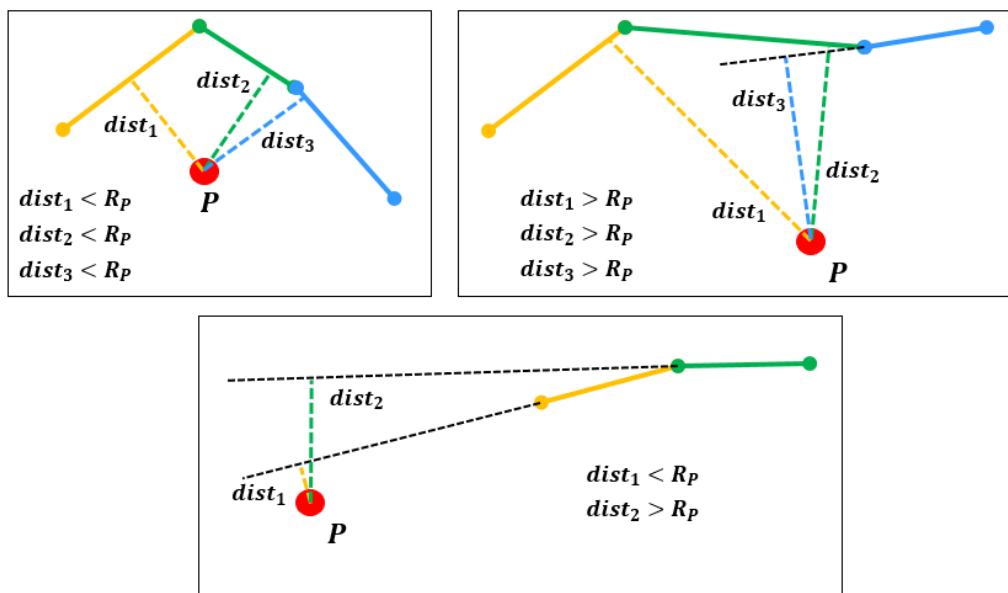


Figura 2.50: Calcolo della distanza tra il punto e i segmenti della centerline: in alto a sinistra si ha un caso di punto già tracciato, a destra un caso di punto non tracciato, in basso un caso di punto non tracciato ma che viene classificato erroneamente come tracciato poichè $dist_1 < R_p$.

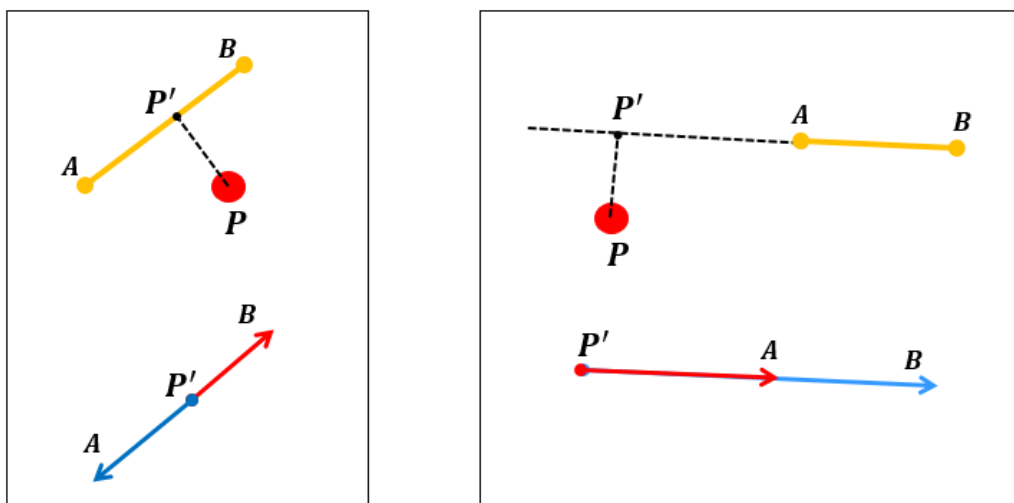


Figura 2.51: Valutazione della posizione della proiezione del punto: a sinistra è mostrato un caso di proiezione del punto interna al segmento, a destra esterna.

Gestione dell'unione dei vasi

Terminato il processo di tracking e ottenuta la centerline, si può incorrere in errori come non identificare correttamente una biforcazione oppure tracciare un singolo vaso come due vasi separati. In questo caso si procede con l'unione delle centerline ottenute qualora fosse rispettata una condizione sulla distanza e una sull'intensità del volume. In particolare, per ogni end-points delle centerline ottenute si considera la direzione di tracking locale (Figura 2.52).

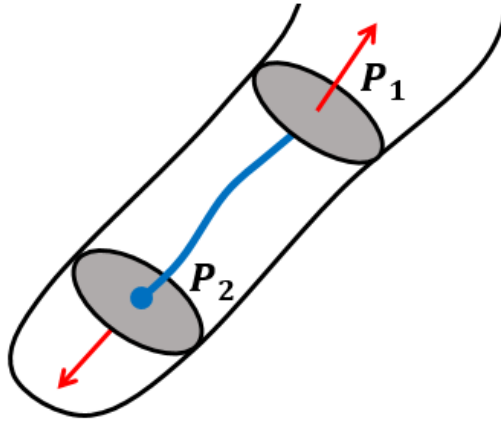


Figura 2.52: P_1 e P_2 sono due end-points e con la freccia rossa è indicata la direzione di tracking locale.

Per ogni end-point viene costruita una zona di ricerca semisferica attorno alla direzione di tracking di raggio pari a due volte il raggio massimo associato alla sezione ellittica di quell'end-point. Per ogni punto contenuto nella semisfera, viene valutata l'intensità del volume lungo il segmento che

congiunge ogni punto e l'end-point. In questo lavoro di tesi, ci si è occupati di volumi binari quindi si è posta una soglia di 0.9 sapendo che i vasi bianchi hanno un valore di 1. Tra tutti i punti che superano questa soglia, viene scelto come vincitore quello al quale è associata la distanza più piccola. Qualora non ci fossero punti che superano la soglia sull'intensità non si procederà con l'unione. Si possono presentare diverse situazioni: unione tra due end-points e unione tra end-point e un punto della centerline formando un punto di biforcazione (branch-point). Entrambe le situazioni sono schematizzate nella Figura 2.53.

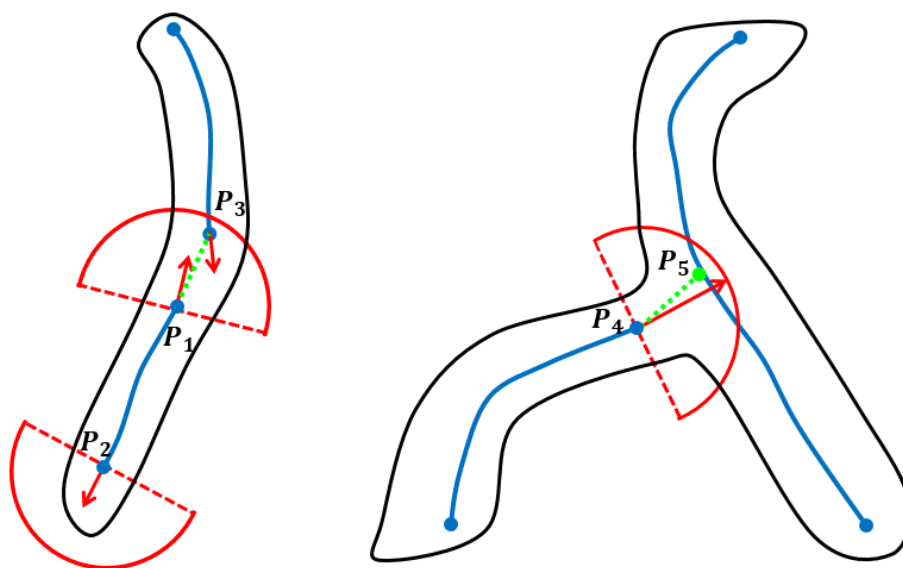


Figura 2.53: Schematizzazione 2D dell'unione delle centerline. A sinistra si ha l'unione tra due punti di fine vaso P_1 e P_3 , mentre nella semisfera associata a P_2 non si sono trovati punti utili per effettuare l'unione. A destra si ha l'unione tra P_4 e un punto interno alla centerline con la creazione di un punto di biforcazione (P_5).

2.4 Metriche di validazione

Applicando l'algoritmo di tracking 3D descritto in precedenza, si ottiene la centerline delle diverse strutture vascolari prese in esame. La validazione è stata effettuata confrontando la centerline ottenuta dall'algoritmo sviluppato, con lo skeleton calcolato tramite medial axis extraction [17]. In particolare è stata calcolata la distanza di Hausdorff, i parametri di tortuosità (DM, ICM e SOAM) e parametri caratteristici quali densità vascolare (VD), numero di alberi (NT) e il raggio medio (AR). É stata valutata anche la corretta identificazione degli end-points e branch-points in numero e posizione.

2.4.1 Distanza di Hausdorff

La distanza di Hausdorff è una metrica che può essere utilizzata per valutare la similarità tra due curve, infatti misura la vicinanza di ciascun punto su una curva rispetto ad ogni punto dell'altra. É data dall'equazione 2.16:

$$H(A, B) = \max(h(A, B), h(B, A)) \quad (2.16)$$

dove

$$h(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\| \quad (2.17)$$

dove a e b rappresentano i punti appartenenti rispettivamente alla curva A e B .

La Figura 2.54 mostra le quattro distanze tra a_3 ed ogni punto appartenente a B . La distanza tra a_3 e b_1 è la più piccola quindi viene identificata

come la minima corrispondente al punto a_3 , successivamente vengono calcolate le restanti distanze minime tra il profilo A e B . La massima distanza tra le minime identificate precedentemente corrisponde ad $h(A, B)$ nell'equazione 2.17. Analogamente è possibile calcolare $h(B, A)$ e il massimo tra $h(A, B)$ e $h(B, A)$ coincide con la distanza di Hausdorff tra le due curve date.

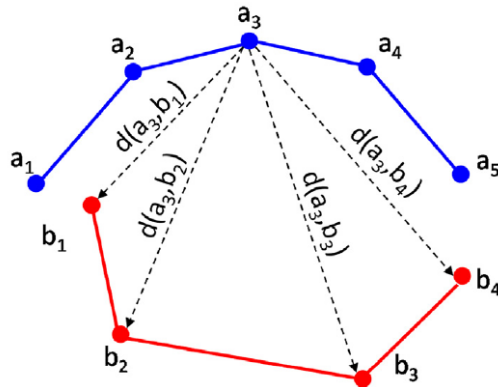


Figura 2.54: Esempio di calcolo della distanza di Hausdorff [18].

Per ciascun modello si è scelto di calcolare una distanza di Hausdorff relativa all'intera centerline in modo da avere una valutazione complessiva. Si è voluta valutare anche la distanza di Hausdorff relativa a ciascun ramo per avere una valutazione locale. Nel primo caso, la distanza calcolata è stata confrontata con il raggio medio totale, nel secondo caso con il raggio medio del vaso preso in considerazione. Si ritiene accettabile un valore della distanza inferiore al raggio.

Nella Figura 2.55 è mostrato un esempio di confronto tra skeleton e tracking: si calcola la distanza di Hausdorff tra le coordinate ottenute dall'algoritmo automatico e quelle dello skeleton.

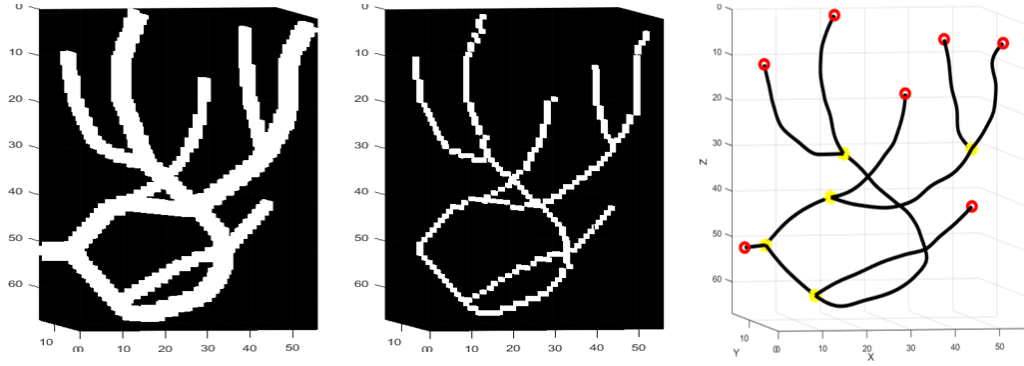


Figura 2.55: Esempio di confronto tra Skeleton e Tracking relativo al modello della Figura 2.1e. A sinistra volume binario originale, al centro lo skeleton, a destra la centerline ottenuta nel processo di tracking.

Per calcolare la distanza di Hausdorff relativa a ciascun ramo è necessario trovare una corrispondenza tra i singoli rami della centerline e quelli dello skeleton. Si è deciso di calcolare la matrice binaria corrispondente a ciascun ramo, applicare a questa una dilatazione con elemento strutturale sferico di raggio 3 e successivamente effettuare un'intersezione tra quest'ultima e la maschera totale dello skeleton in modo da isolare il ramo voluto (Figura 2.56).

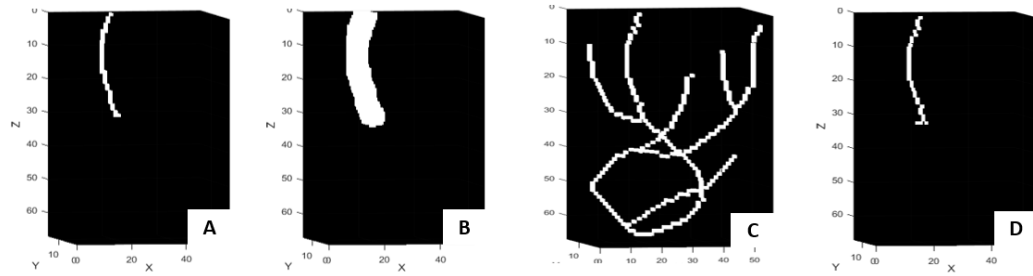


Figura 2.56: (a) Matrice binaria relativa alle coordinate di un ramo del centerline ottenuta. (b) Dilatazione. (c) Maschera binaria relativa allo skeleton. (d) Intersezione tra (b) e (c).

Quindi la distanza di Hausdorff locale per ciascun ramo è stata calcolata tra le coordinate del ramo ottenute dal tracking e quelle del ramo isolato nello skelton.

2.4.2 Caratterizzazione numerica e parametri di tortuosità

Quasi tutte le patologie influiscono sulle caratteristiche dei vasi sanguigni come il numero di alberi vascolari, il raggio, la tortuosità o il numero di biforcazioni. La misurazione di questi parametri fornisce un importante strumento per diagnosticare le malattie. Per caratterizzare numericamente una rete vascolare, si possono analizzare diverse caratteristiche:

- **NT**: indica il numero di alberi vascolari indipendenti ovvero il numero di alberi con un proprio specifico numero di ramificazioni e che non intersecano mai un altro albero vascolare. In presenza di lesioni tumorali si ha un NT che è sempre maggiore di quello relativo alle lesioni sane. Inoltre questo valore aumenta man mano che la lesione progredisce nel tempo.
- **VD**: indica la densità vascolare dato un certo volume ovvero presa in esame una ROI binaria, la VD è data dal numero di voxel bianchi diviso il numero totale dei voxel della ROI. A reti più complesse caratteristiche di zone tumorali, è associato un alto valore di questo parametro.
- **NB**: indica il numero di diramazioni ed assume valore più alto in caso di lesione sospetta rispetto ad una non sospetta e aumenta man mano

che la lesione progredisce nel tempo.

- **AR:** indica il raggio medio del vaso che risulta essere più piccolo in caso di tumore in cui i vasi sono privi di endotelio in quanto devono occupare poco spazio ma quasi tutto il volume.

Per analizzare un'architettura sospetta da una non sospetta bisogna valutare la complessità ossia la tortuosità servendosi dei seguenti descrittori:

- **DM:** *distance metric*, indica la lunghezza reale del vaso rapportata alla distanza euclidea tra il punto di inizio e di fine del percorso vascolare (equazione 2.18). Più il vaso è complesso e lungo, tanto più la lunghezza che connette realmente i punti è maggiore della distanza euclidea. Questo parametro sarà maggiore in caso di tumore in quanto la rete vascolare è più complessa di una lesione benigna.

$$DM = \frac{L}{DE} \quad (2.18)$$

- **ICM:** *inflections count metric*, è definita a partire da punti connessi tra loro e tiene conto delle variazioni della curva con elevata ampiezza, infatti è maggiore per curve che hanno delle oscillazioni molto grandi a frequenza abbastanza elevata. Viene introdotto perchè anche se *DM* calcola quanto un percorso devia rispetto ad una linea retta, la sua utilità è limitata in quanto potrebbe assegnare lo stesso valore di tortuosità ad una curva a forma di C ampia e ad una più tortuosa e che effettua brusche variazioni di direzione. L'*ICM* è dato dal prodotto

tra la DM e il numero di punti di flesso più uno (equazione 2.19). Nel caso in cui non si identificano cambi di direzione, l' ICM restituisce lo stesso valore di DM .

$$ICM = DM \times (num.flessi + 1) \quad (2.19)$$

- **SOAM:** *sum of angles metric*, è definito come la somma di tutti gli angoli che si hanno percorrendo il vaso, diviso la lunghezza totale del percorso e viene introdotto perchè uno svantaggio sia del DM che dell' ICM è che nessuno dei due gestisce bene il caso di vasi con avvolgimento a spirale.

Tuttavia, in questo lavoro di tesi non si è effettuato un confronto tra zone tumorali e sane ma i parametri di tortuosità e parametri caratteristici sono stati usati per effettuare un confronto tra la centerline estratta e lo skeleton. In particolare, il raggio medio (AR) è usato nel confronto con la distanza di Hausdorff come descritto precedentemente, il numero di biforcazioni (NB) viene preso in considerazione nel paragrafo successivo mentre il numero di alberi (NT) non è informativo perchè tutti i volumi analizzati sono caratterizzati da un solo albero vascolare. Inoltre sono stati presi in esame i parametri DM, ICM, SOAM e VD. Sono stati calcolati questi quattro parametri per ogni modello e poi rapportati a quello meno tortuoso in modo da osservare la variazione della complessità tra i diversi volumi e successivamente valutare se si ha lo stesso trend sia per l'algoritmo proposto che per lo skeleton.

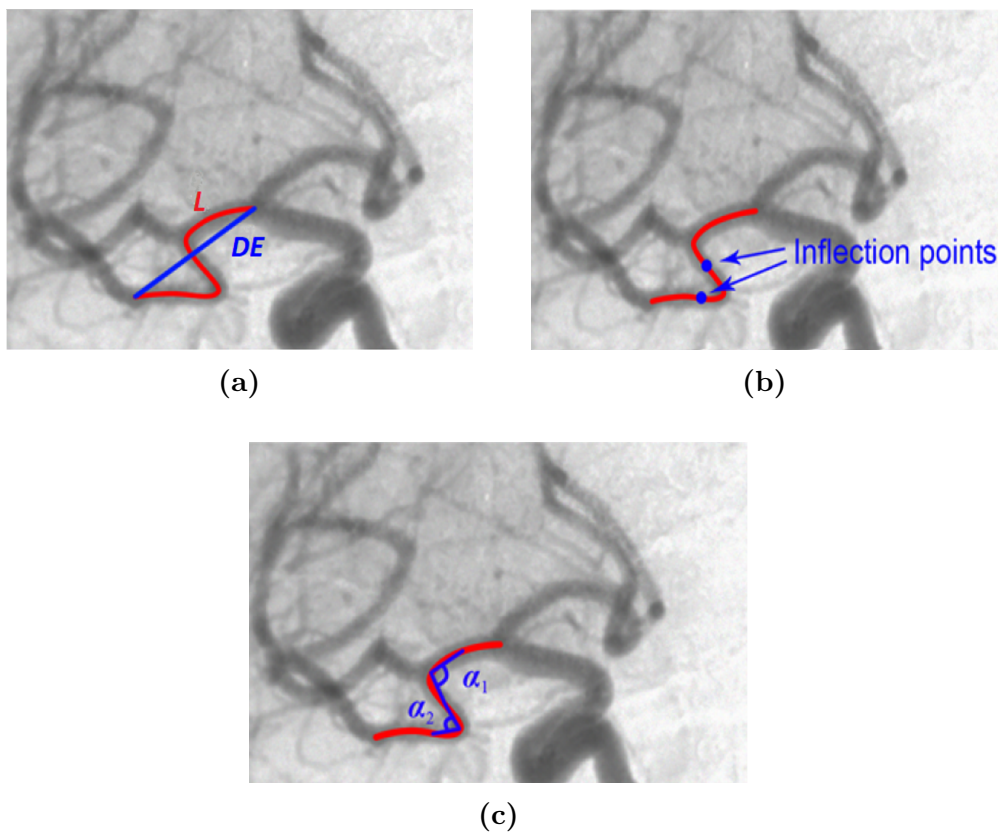


Figura 2.57: Esempio di calcolo dei parametri di tortuosità: (a) Distance metric data dal rapporto tra la lunghezza totale del vaso L e la distanza euclidea DE . (b) Sono indicati due punti di flesso utili per il calcolo di ICM. (c) α_1 e α_2 sono gli angoli formati tra ogni trio di punti della curva in rosso [19].

2.4.3 Valutazione dei punti di biforcazione e punti di fine vaso

Si è poi valutata la corretta identificazione dei punti di fine vaso (end-points) e di biforcazione (branch-points) in numero e posizione calcolando i seguenti parametri:

- *Precision*, detto anche valor predittivo positivo (PPV): è il rapporto tra veri positivi e tutti i valori che sono stati assegnati come positivi.

$$PPV = \frac{TP}{TP + FP} \quad (2.20)$$

- *Recall*, detta anche sensibilità o true positive rate (TPR): è il rapporto tra i veri positivi e tutti i campioni che sono davvero positivi.

$$TPR = \frac{TP}{TP + FN} \quad (2.21)$$

- *F-score*: è definito come la media armonica tra la precisione e la sensibilità ed è definita tra 0 e 1.

$$F - score = \frac{2TP}{2TP + FP + FN} \quad (2.22)$$

dove TP indica il numero di veri positivi, FP il numero di falsi positivi e FN il numero di falsi negativi.

Per definire il numero di TP per gli end-points, si calcola la distanza tra ciascun end-points ottenuto dall'algoritmo automatico e ognuno di quelli ottenuti tramite la funzione *bwmorph* di Matlab applicata allo skeleton. La

distanza massima è posta pari al raggio medio associato a quel punto: se la soglia è rispettata, quel punto è classificato come TP. Una volta terminato il conteggio dei TP, i rimanenti end-points relativi all'algoritmo automatico costituiscono i FP mentre gli end-points rimanenti relativi allo skeleton sono i FN. Lo stesso procedimento è stato effettuato per il calcolo dei TP, FP e FN dei branch-points.

Capitolo 3

Risultati

In questo capitolo sono mostrati i risultati ottenuti: una prima sezione riporta la centerline relativa ad ogni volume considerato, una seconda sezione riporta i risultati relativi all'analisi sulla distanza di Hausdorff, una terza inerente al confronto sui parametri di tortuosità e infine l'analisi sull'identificazione dei branch-points e end-points.

3.1 Centerline

Di seguito sono riportate le centerline ottenute con l'applicazione dell'algoritmo di tracking sui modelli simulati e sulle immagini reali. In particolare, nella Figura 3.1 sono mostrate le centerline relative ai modelli simulati ed è mostrato un caso di gestione dell'unione dei vasi nella Figura 3.1g. Nella Figura 3.2 sono mostrate invece le centerline relative alle ROI prese in considerazione sulle immagini OCT-A.

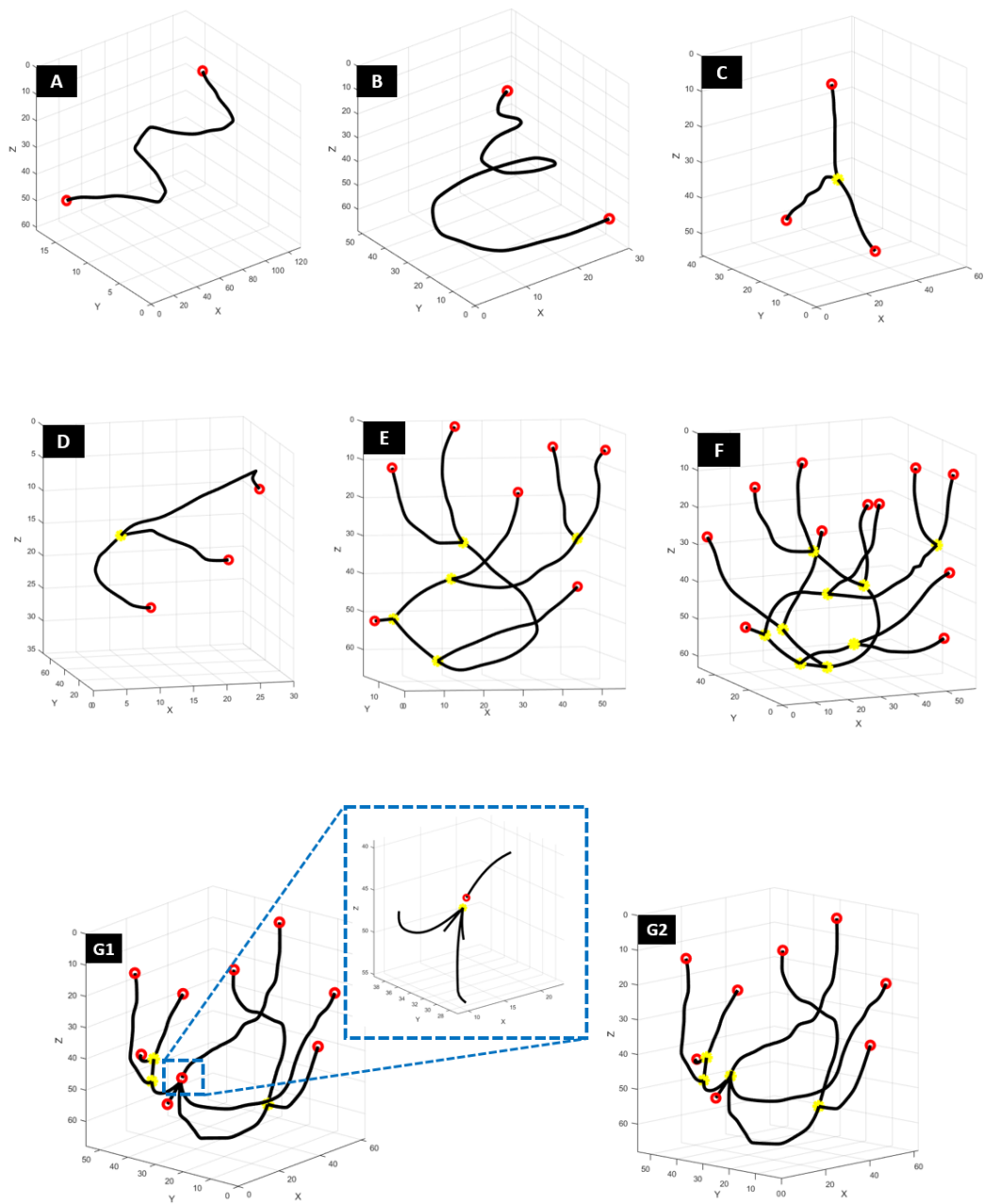


Figura 3.1: Le centerline mostrate sono state ottenute applicando l'algoritmo sulle maschere binarie della Figura 2.3. In particolare la G1 è la centerline ottenuta sul modello 2.3g e si osserva una disconnessione che è stata unita nella figura G2.

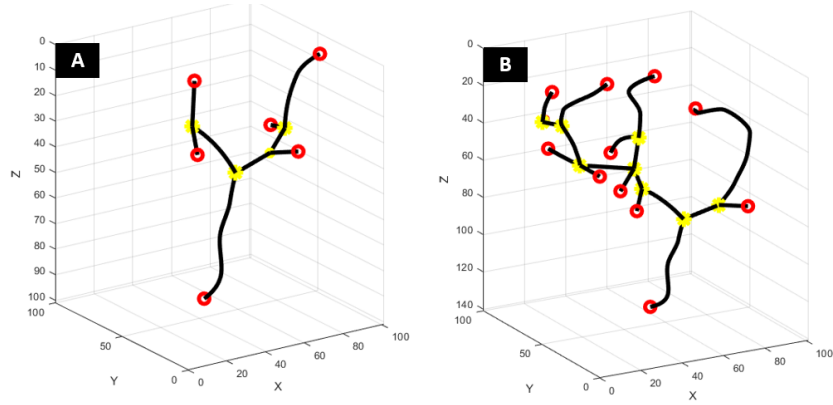


Figura 3.2: Centerline per i volumi reali. (a) Centerline del volume binario 2.8a. (b) Centerline del volume binario 2.8b.

Nella Tabella 3.1 è riportato il tempo impiegato per effettuare il tracking di ciascun volume.

Volume	Dimensione	Tempo(s)
Curvo	18 x 131 x 61	5.0273
Elica	52 x 30 x 69	4.7236
Biforcazione Y	41 x 60 x 56	3.0569
Biforcazione	70 x 30 x 35	4.8272
Semplice	16 x 56 x 67	11.3716
Complesso	50 x 59 x 63	16.2327
Profondo	55 x 62 x 68	13.0423
In-vivo 1	100 x 101 x 100	4.8626
In-vivo 2	138 x 101 x 100	11.5442

Tabella 3.1: Tempo impiegato per tracciare ciascun volume.

3.2 Distanza di Hausdorff

Nelle Figure 3.3-3.11 è mostrato per ogni modello la maschera binaria del volume iniziale, lo skeleton e il risultato del tracking. Nelle Figure 3.12-3.19 è mostrato invece il confronto tra la distanza di Hausdorff e il valore del raggio medio: è riportato sia il confronto tra la distanza di Hausdorff relativa a ciascun ramo che quella totale. Per i modelli che non hanno ramificazioni, è riportato un unico valore di distanza. È facile osservare come la distanza di Hausdorff risulti essere accettabile poichè sempre inferiore al raggio medio per ciascun volume considerato.

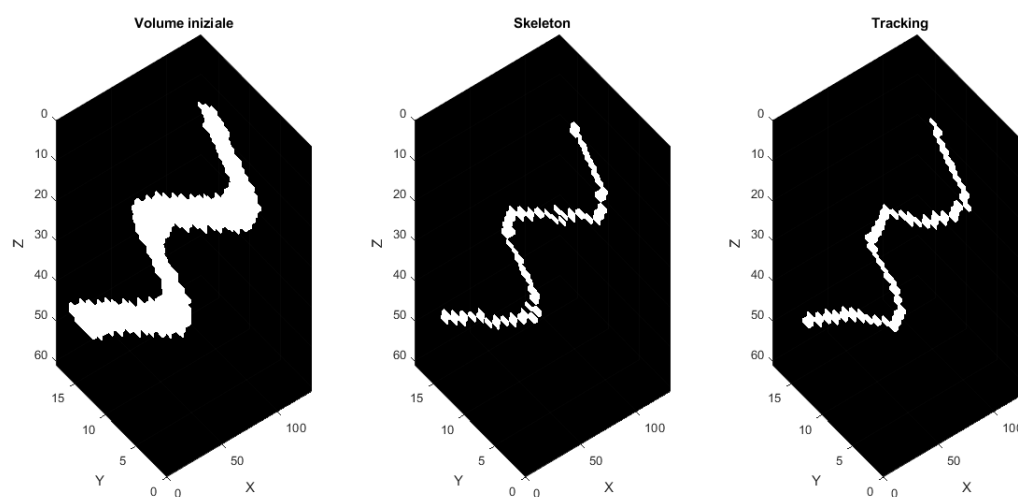


Figura 3.3: Skeleton-Tracking modello curvo.

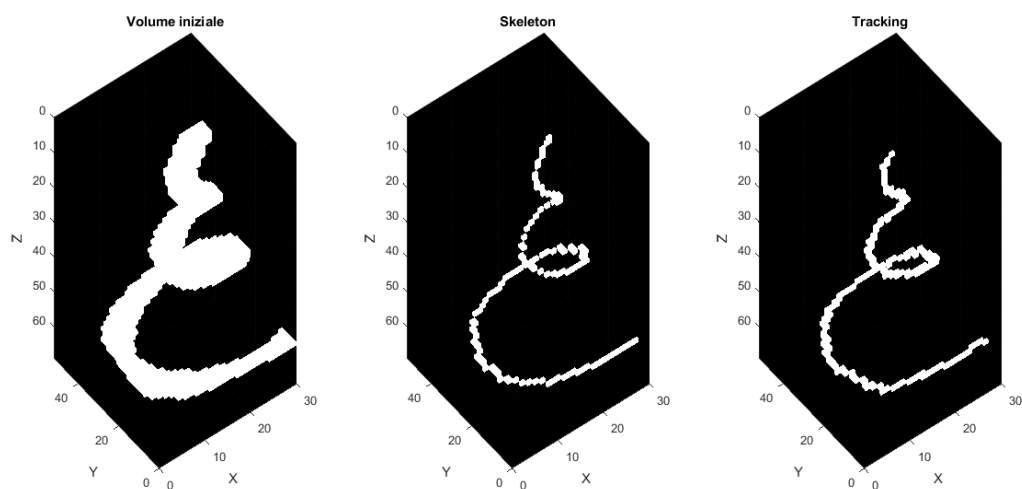


Figura 3.4: Skeleton-Tracking modello elica.

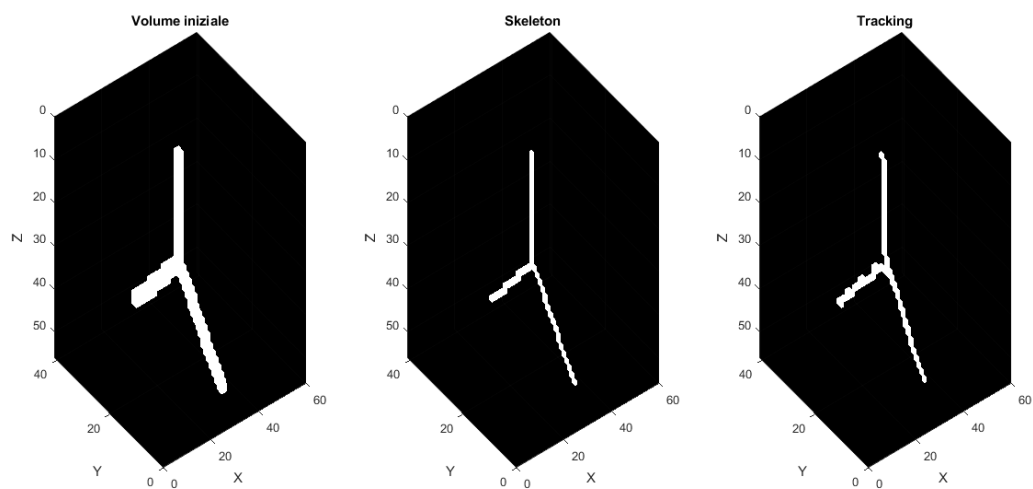


Figura 3.5: Skeleton-Tracking modello biforcazione Y.

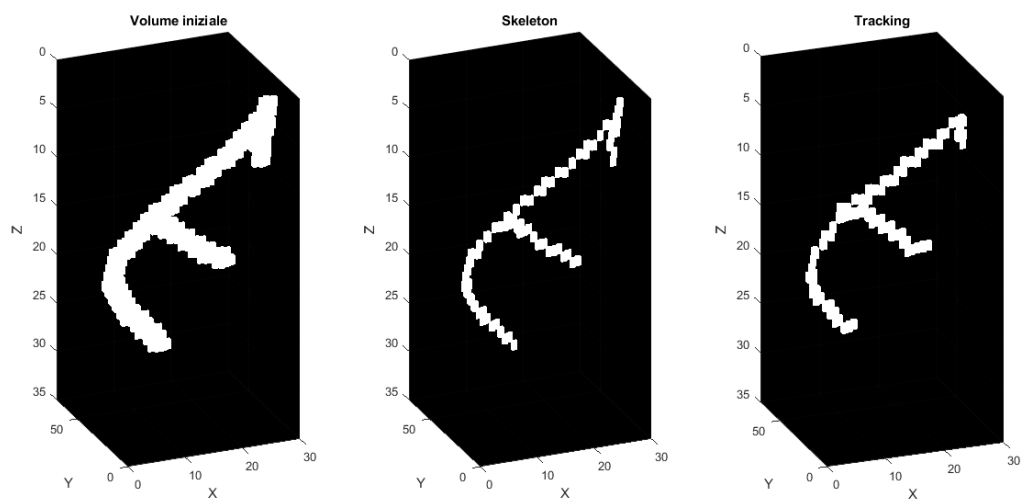


Figura 3.6: Skeleton-Tracking modello biforcazione.

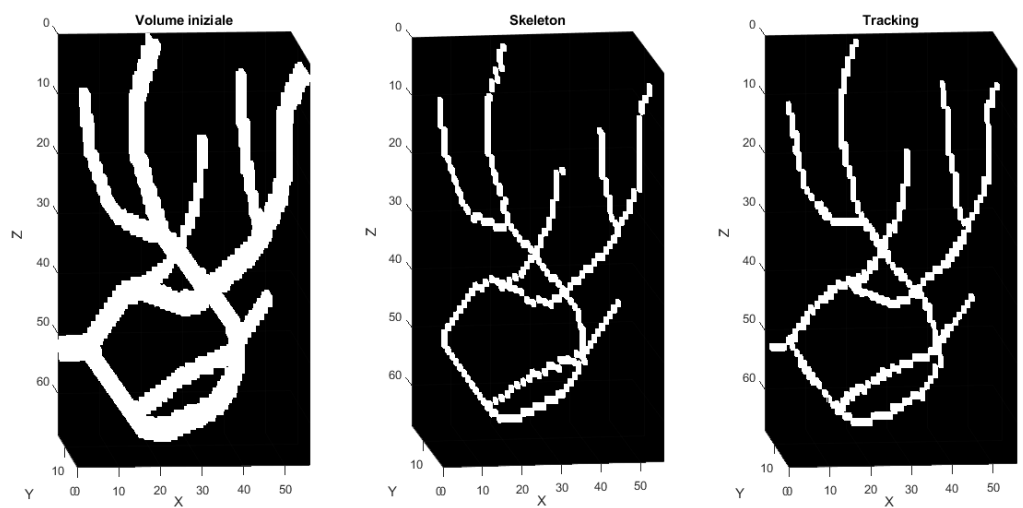


Figura 3.7: Skeleton-Tracking modello albero semplice.

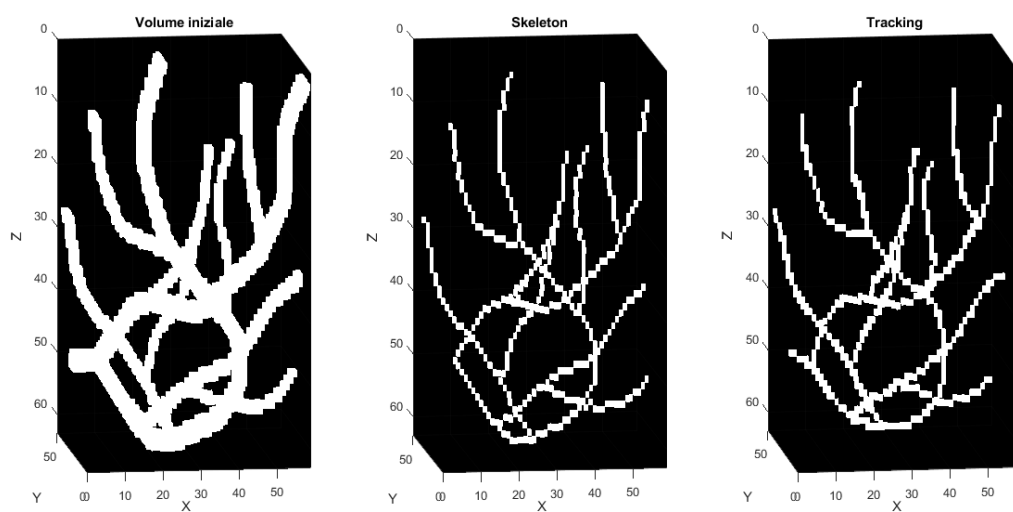


Figura 3.8: Skeleton-Tracking modello albero complesso.

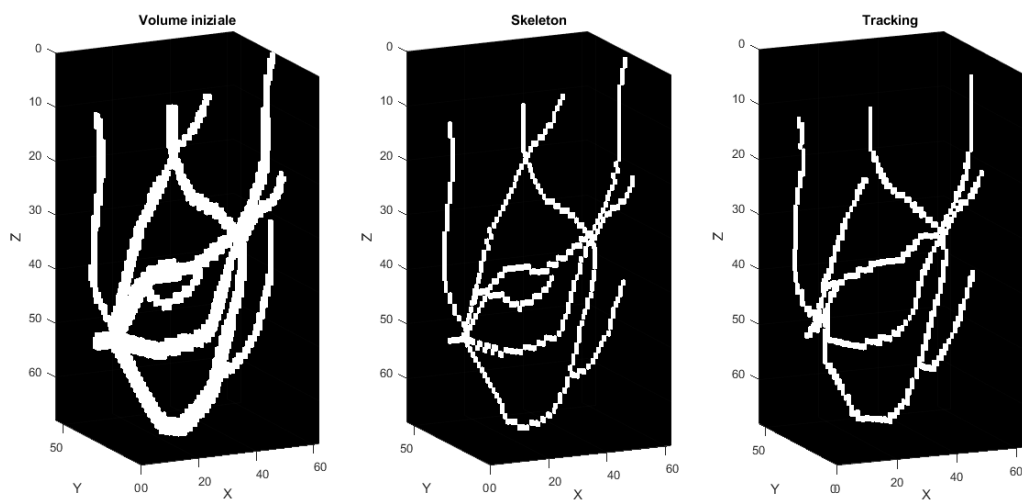


Figura 3.9: Skeleton-Tracking modello albero profondo.

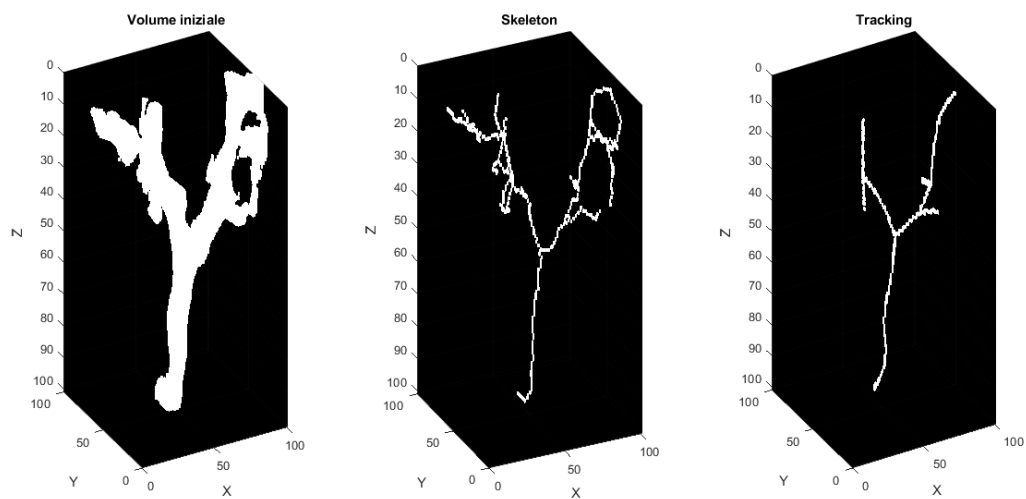


Figura 3.10: Skeleton-Tracking in-vivo 1.

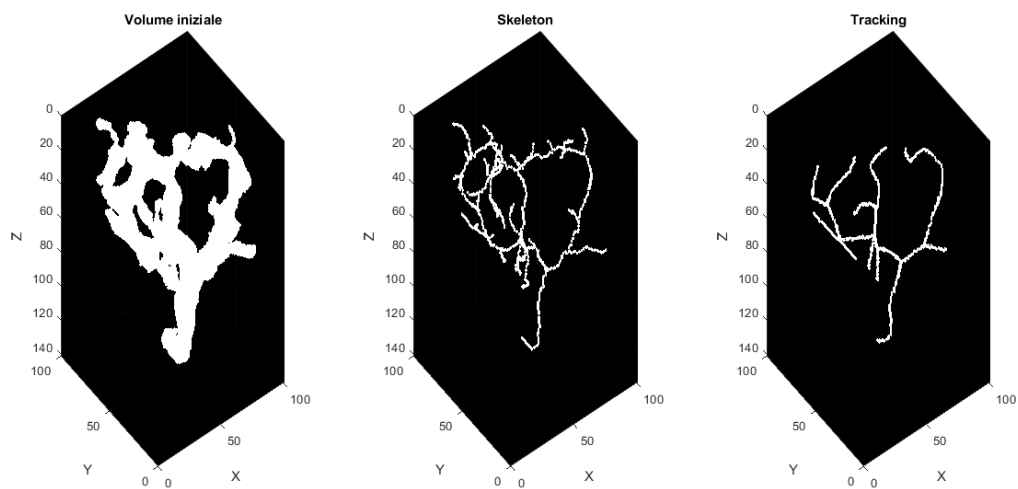


Figura 3.11: Skeleton-Tracking in-vivo 2.

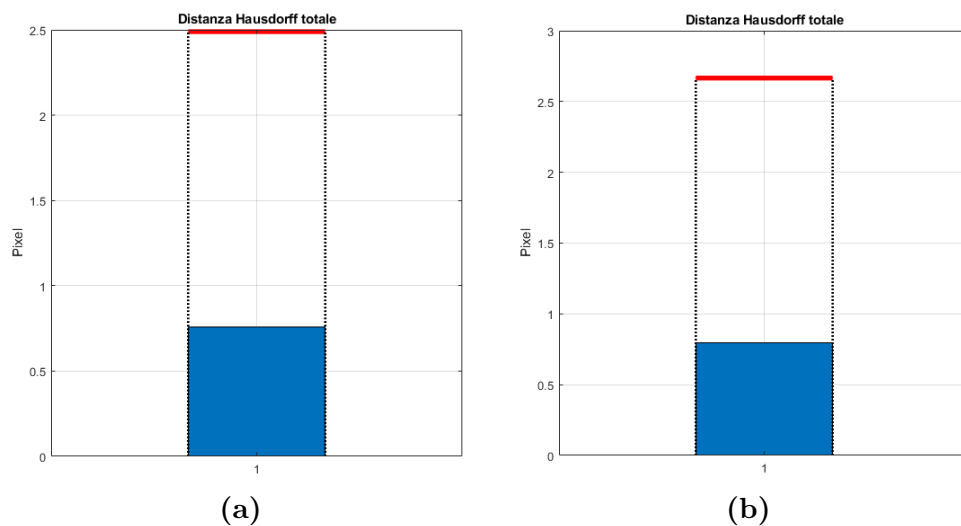


Figura 3.12: Confronto distanza di Hausdorff col il raggio medio per modello curvo ed elica: la soglia rossa indica il valore del raggio medio. (a) La distanza di Hausdorff relativa al modello curvo. (b) La distanza di Hausdorff relativa all'elica.

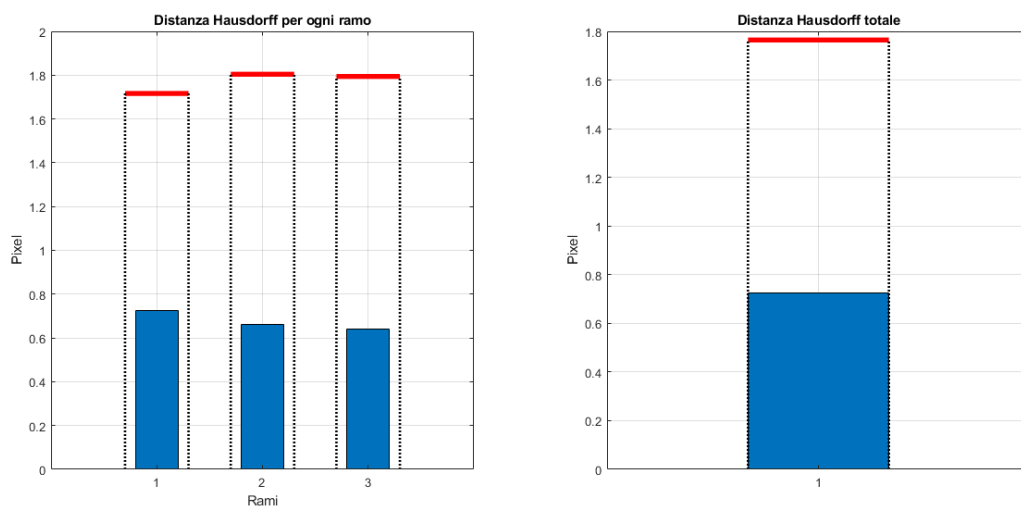


Figura 3.13: Confronto distanza di Hausdorff col il raggio medio per il modello biforcazioneY: la soglia rossa indica il valore del raggio medio, a sinistra è mostrato il confronto per ogni ramo, a destra quello totale.

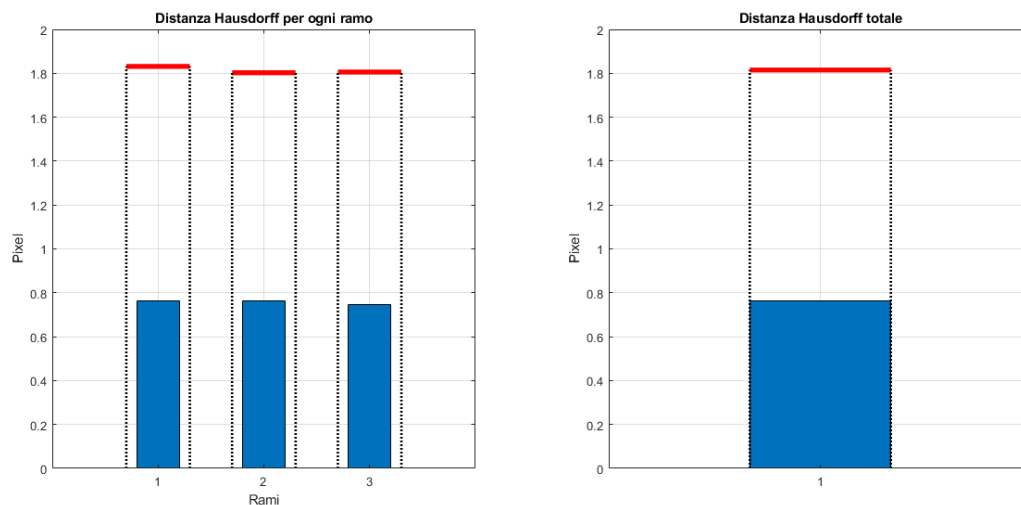


Figura 3.14: Confronto distanza di Hausdorff col il raggio medio per il modello biforcazione: la soglia rossa indica il valore del raggio medio, a sinistra è mostrato il confronto per ogni ramo, a destra quello totale.

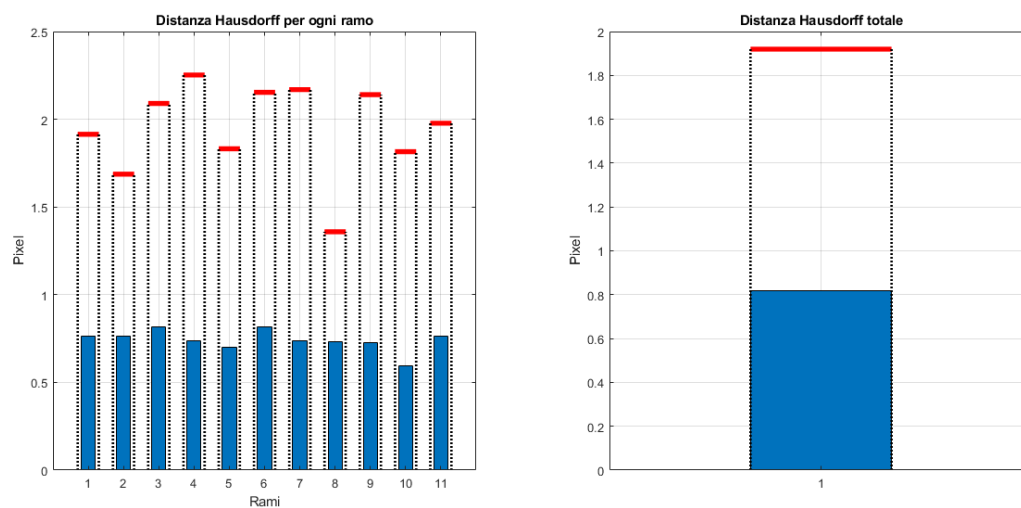


Figura 3.15: Confronto distanza di Hausdorff col il raggio medio per il modello albero semplice: la soglia rossa indica il valore del raggio medio, a sinistra è mostrato il confronto per ogni ramo, a destra quello totale.

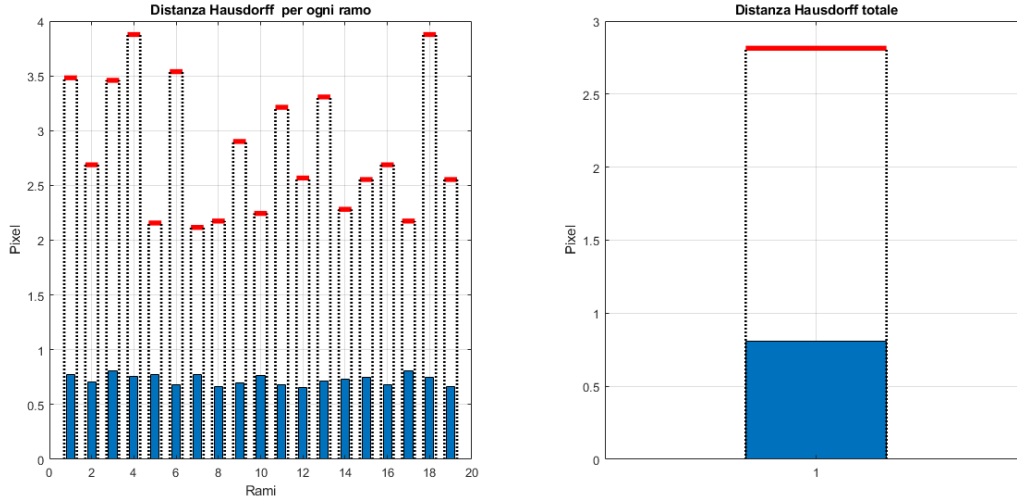


Figura 3.16: Confronto distanza di Hausdorff col il raggio medio per il modello albero complesso: la soglia rossa indica il valore del raggio medio, a sinistra è mostrato il confronto per ogni ramo, a destra quello totale.

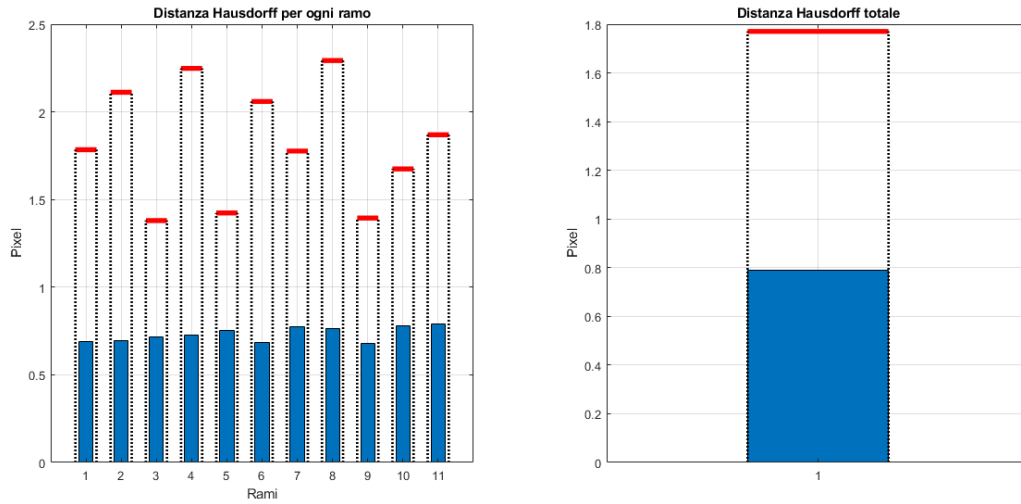


Figura 3.17: Confronto distanza di Hausdorff col il raggio medio per il modello albero profondo: la soglia rossa indica il valore del raggio medio, a sinistra è mostrato il confronto per ogni ramo, a destra quello totale.

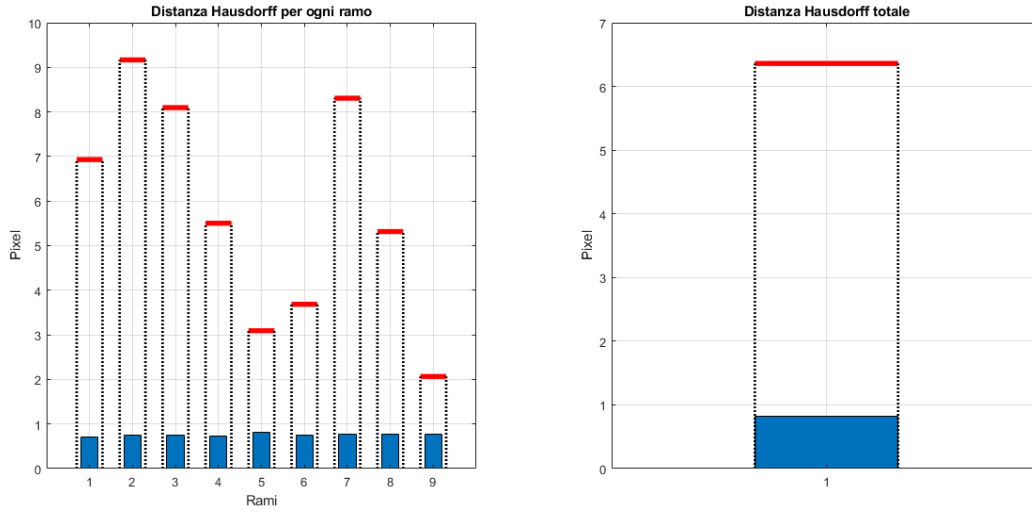


Figura 3.18: Confronto distanza di Hausdorff col il raggio medio per il modello in-vivo 1: la soglia rossa indica il valore del raggio medio, a sinistra è mostrato il confronto per ogni ramo, a destra quello totale.

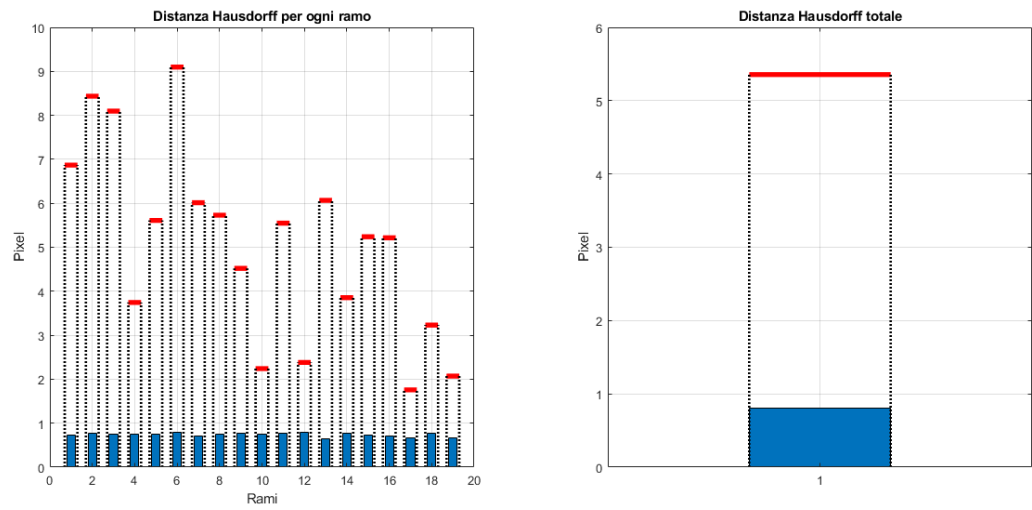


Figura 3.19: Confronto distanza di Hausdorff col il raggio medio per il modello in-vivo 2: la soglia rossa indica il valore del raggio medio, a sinistra è mostrato il confronto per ogni ramo, a destra quello totale.

3.3 Caratterizzazione numerica e parametri di tortuosità

In questa sezione sono mostrati i trend a confronto tra lo skeleton e il tracking. Si è scelto di confrontare i volumi formando tre sottogruppi: il primo composto dai modelli *biforcazioneY*, *curvo*, *biforcazione* ed *elica* in ordine crescente di complessità; il secondo che mette a confronto l'*albero semplice*, il *complesso* e il *profondo*, il terzo confronta i due modelli in-vivo.

I parametri calcolati sono stati rapportati a quelli relativi al meno complesso. Nelle Figure 3.20-3.22 è possibile osservare un andamento simile per tutti i parametri calcolati eccetto che per il parametro SOAM nella Figura 3.20 nel quale si riscontra una complessità più alta per il modello *biforcazione* rispetto all'*elica* per lo skeleton.

Questo è molto probabilmente dovuto al fatto che l'algoritmo automatico non identifica un branch-point e un end-point nel modello *biforcazione* (come verrà spiegato nel dettaglio nella sezione 3.4 Valutazione dei punti di biforcazione e punti di fine vaso). Quindi ad essa corrisponde una complessità inferiore per l'algoritmo automatico.

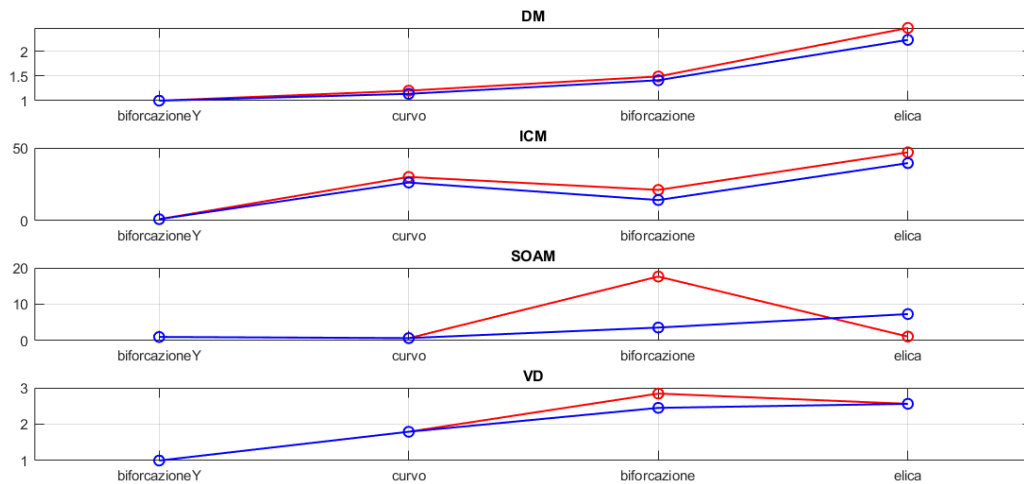


Figura 3.20: Confronto parametri di tortuosità modello biforcazioneY, curvo, biforcazione, elica. In rosso lo skeleton, in blu il tracking.

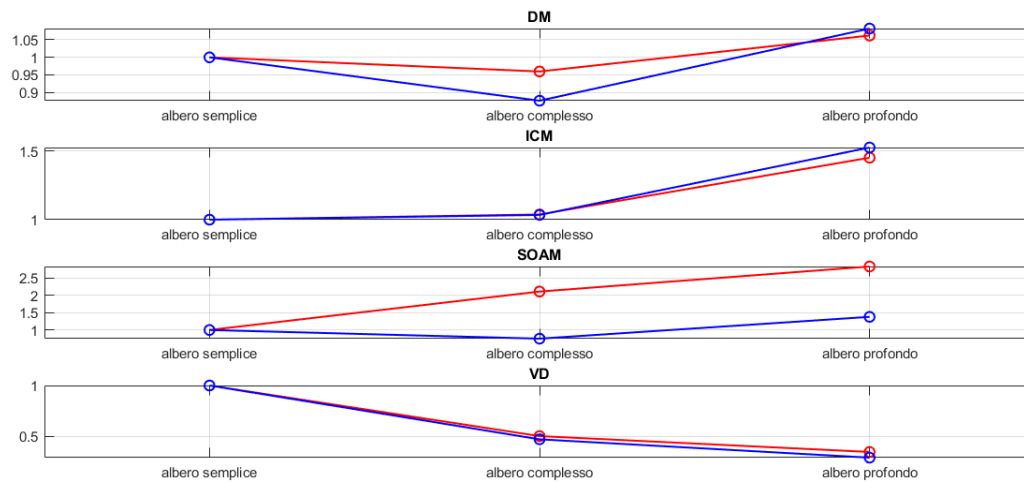


Figura 3.21: Confronto parametri di tortuosità modello albero semplice, albero complesso, albero profondo. In rosso lo skeleton, in blu il tracking.

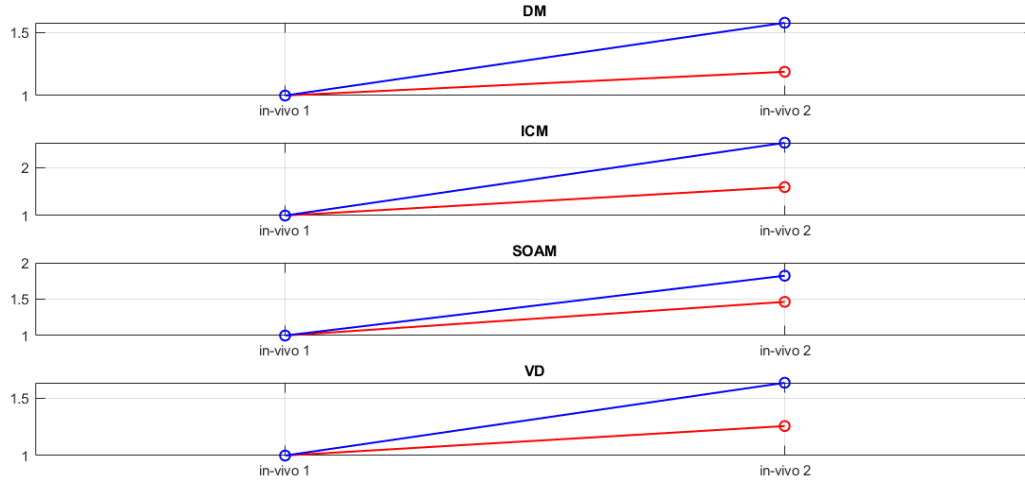


Figura 3.22: Confronto parametri di tortuosità modello in-vivo 1, in-vivo 2. In rosso lo skeleton, in blu il tracking.

3.4 Valutazione dei punti di biforcazione e punti di fine vaso

Nella Figura 3.23 è possibile avere un'idea complessiva sulla corretta identificazione degli end-points e branch-points per tutti i volumi presi in considerazione. Si osserva che per i modelli *curvo*, *elica* e *biforcazioneY* le prestazioni sono ottime essendo i modelli molto semplici. Per il modello *biforcazione* si ha la perdita di un branch-point e un end-point come si può osservare nella Figura 3.24. Per i modelli di albero vascolare *semplice* e *complesso* si presenta la stessa problematica facilmente osservabile nella Figura 3.25 e 3.26: il valore di TPR è pari al 100% in entrambi i modelli perchè tutti i branch-points e end-points sono stati correttamente identificati però lo skeleton perde un end-point e un branch-point che l'algoritmo automatico identifica correttamente. Questo fa abbassare il valore di PPV e quindi anche quello dell'F-score. Per

quel che riguarda il modello di albero vascolare *profondo* si ha la corretta identificazione dei punti in termini di numero ma si osservano dei problemi nell'identificazione degli end-points: nella Figura 3.27 è possibile osservare che due rami non sono stati tracciati completamente quindi gli end-points identificati sono classificati come FP poichè troppo distanti dai rispettivi nello skeleton. Per i volumi in-vivo si sono ottenute le peggiori prestazioni poichè questi sono caratterizzati da un'architettura molto complessa e reticolata. Tuttavia in entrambi i volumi si osservano valori per PPV superiori all'80% quindi gli end-points e branch-points identificati dall'algoritmo automatico sono corretti. Nella Figura 3.28 è possibile visualizzare graficamente la problematica riscontrata per i due volumi in-vivo.

		<i>Numero end-points</i>	<i>PPV</i>	<i>TPR</i>	<i>F-score</i>	<i>Numero branch-points</i>	<i>PPV</i>	<i>TPR</i>	<i>F-score</i>
Curvo	<i>Skeleton</i>	2	100%	100%	100%	0	/	/	/
	<i>Tracking</i>	2				0			
Elica	<i>Skeleton</i>	2	100%	100%	100%	0	/	/	/
	<i>Tracking</i>	2				0			
BiforcazioneY	<i>Skeleton</i>	3	100%	100%	100%	1	100%	100%	100%
	<i>Tracking</i>	3				1			
Biforcazione	<i>Skeleton</i>	4	100%	75%	86%	2	100%	50%	67%
	<i>Tracking</i>	3				1			
Semplice	<i>Skeleton</i>	6	86%	100%	92%	4	80%	100%	89%
	<i>Tracking</i>	7				5			
Complesso	<i>Skeleton</i>	10	91%	100%	95%	8	89%	100%	94%
	<i>Tracking</i>	11				9			
Profondo	<i>Skeleton</i>	8	75%	75%	75%	4	100%	100%	100%
	<i>Tracking</i>	8				4			
In-vivo1	<i>Skeleton</i>	20	83%	25%	38%	22	100%	18%	31%
	<i>Tracking</i>	6				4			
In-vivo2	<i>Skeleton</i>	35	100%	34%	51%	35	88%	20%	33%
	<i>Tracking</i>	12				8			

Figura 3.23: Valutazione branch-points e end-points.

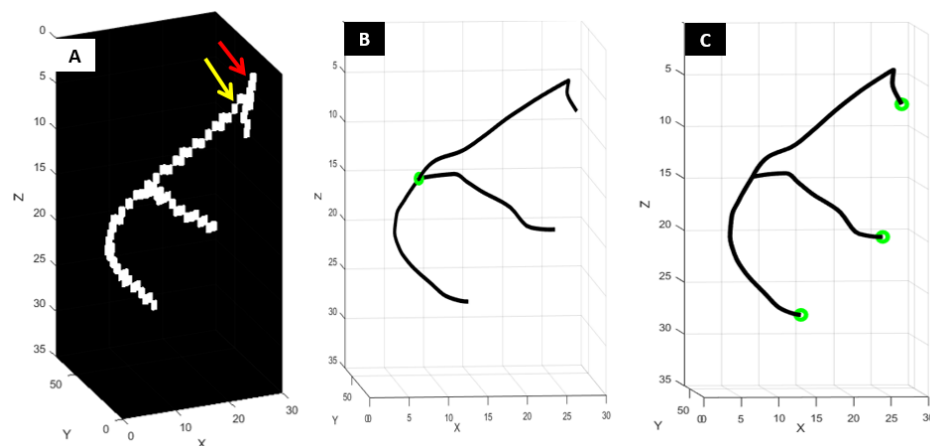


Figura 3.24: (a) Skeleton: la freccia gialla indica il branch-point perso, quella rossa l'end-point perso. (b) In verde sono indicati i branch-points TP. (c) In verde sono indicati gli end-points TP.

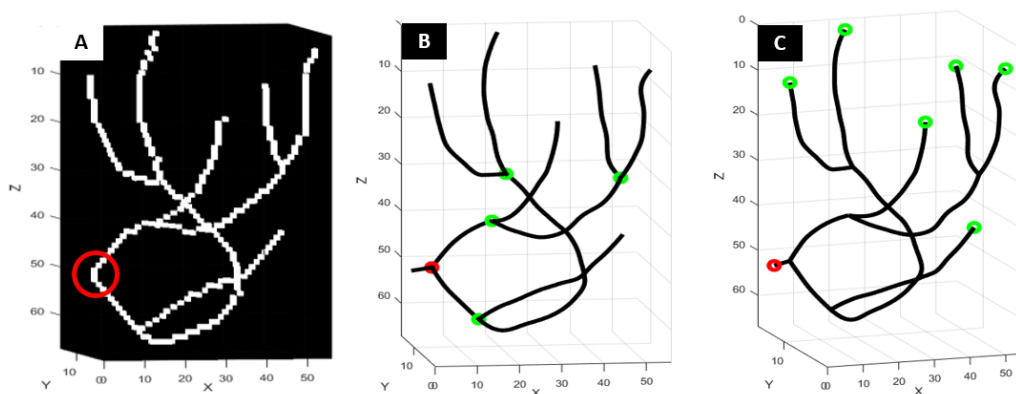


Figura 3.25: Skeleton: in rosso è cerchiato il segmento di vaso perso. (b) In verde sono indicati i branch-points TP e in rosso quelli classificati come FP. (c) In verde sono indicati i end-points TP e in rosso quelli classificati come FP.

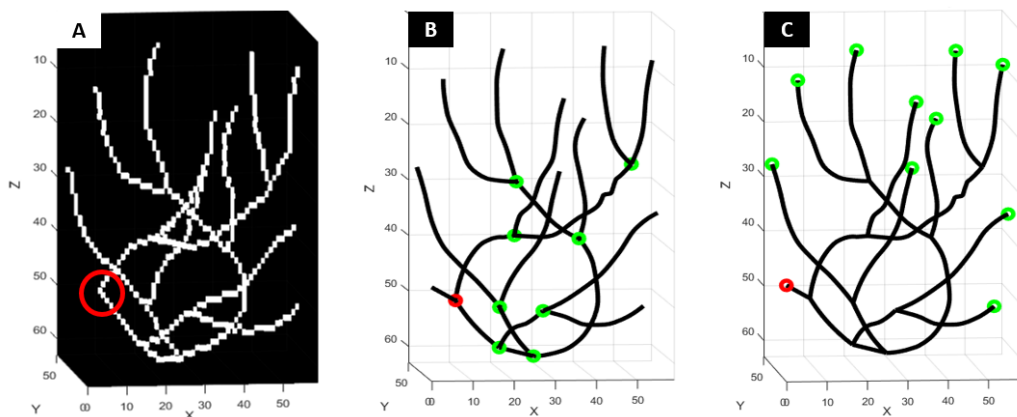


Figura 3.26: (a) Skeleton: in rosso è cerchiato il segmento di vaso perso. (b) In verde sono indicati i branch-points TP e in rosso quelli classificati come FP. (c) In verde sono indicati i end-points TP e in rosso quelli classificati come FP.

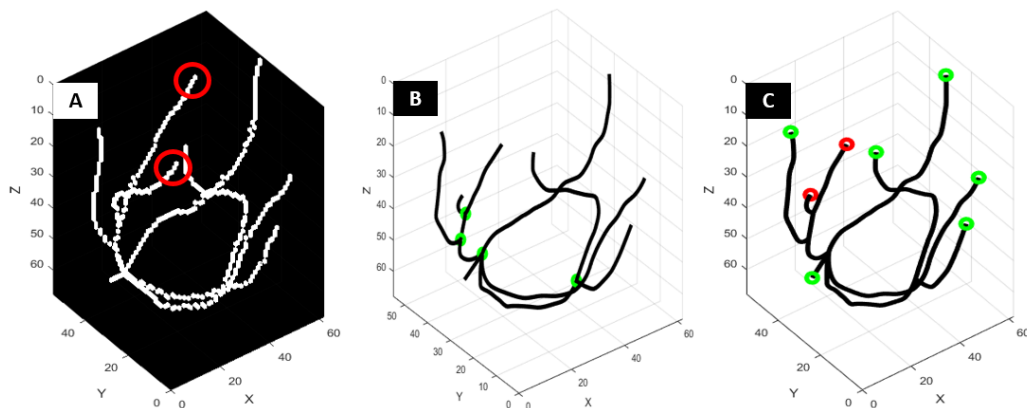


Figura 3.27: (a) Skeleton: in rosso sono cerchiati gli end-points corretti. (b) In verde sono indicati i branch-points TP. (c) In verde sono indicati i end-points TP e in rosso quelli classificati come FP.

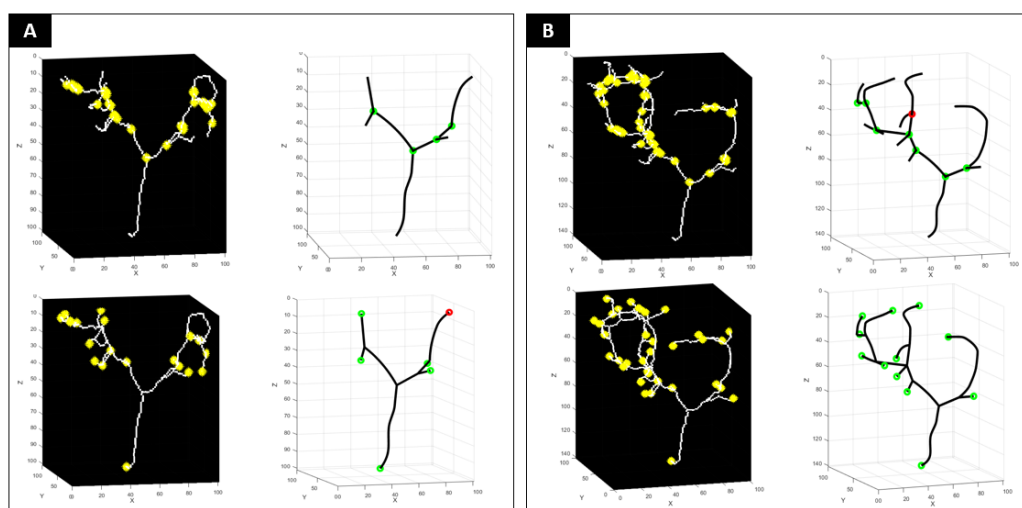


Figura 3.28: (a) Volume in-vivo 1. (b) Volume in-vivo 2.

Conclusioni e sviluppi futuri

In questo lavoro di tesi si è voluto sviluppare un algoritmo automatico per il tracking 3D di strutture vascolari ramificate di media complessità.

Dai risultati ottenuti sui modelli simulati si può concludere che l'algoritmo sviluppato è adatto per questa tipologia di architettura. Tra i modelli simulati, quello per il quale si sono riscontrati alcuni problemi è il modello *profondo* che è caratterizzato da variazioni di diametro che vanno da 6 a 1 pixel. Infatti i rami che l'algoritmo automatico non è riuscito a tracciare hanno un diametro di 1 pixel. Sebbene si sia cercato di tracciare il vaso in modo che la sfera costruita si adatti il più possibile alle variazioni brusche di diametro, è necessario migliorare quest'aspetto rendendo l'algoritmo più robusto da questo punto di vista.

Per quel che riguarda invece i volumi in-vivo è necessario sottolineare che sono modelli reali, quindi in quanto tali più complessi dei simulati. Inoltre sono caratterizzati da una struttura reticolata. Come sviluppo futuro, si potrebbe adattare l'algoritmo a strutture a rete in modo da renderlo più universale e applicabile a diverse architetture vascolari dalle più semplici alle più complesse.

Bibliografia

- [1] Jianhuang Wu et al. «A Review of Vessel Extraction Techniques and Algorithms». In: *ACM Computing Surveys* (dic. 2002). DOI: 10.1145/1031120.1031121.
- [2] Cemil Kirbas e Francis Quek. «A Preliminary Real-Time and Realistic Simulation Environment for Percutaneous Coronary Intervention». In: *BioMed Research International* (apr. 2015). ISSN: 2314-6141. DOI: 10.1155/2015/183157.
- [3] Molinari Filippo. «Elaborazione si immagini mediche, slide, Politecnico di Torino». In: (2017).
- [4] Runhuai Yang et al. «Cell segmentation and pipette identification for automated patch clamp recording». In: *Robotics and Biomimetics* (dic. 2014). DOI: 10.1186/s40638-014-0020-5.
- [5] Sara Moccia et al. «Blood vessel segmentation algorithms-Review of methods, datasets and evaluation metrics». In: *Computer Methods and Programs in Biomedicine* 158 (mag. 2018), pp. 71–91. DOI: 10.1016/j.cmpb.2018.02.001.

-
- [6] Ola Friman et al. «Multiple hypothesis template tracking of small 3D vessel structures». In: *Medical Image Analysis* 14.2 (apr. 2010), pp. 160–171. DOI: 10.1016/j.media.2009.12.003.
- [7] *Mesh voxelisation*. URL: <https://it.mathworks.com/matlabcentral/fileexchange/27390-mesh-voxelisation>.
- [8] *Introduction to Ray Tracing: a Simple Method for Creating 3D Images*. URL: <https://www.scratchapixel.com/lessons/3d-basic-rendering/introduction-to-ray-tracing/implementing-the-raytracing-algorithm>.
- [9] *Optical Coherence Tomography Angiography*. URL: https://eyewiki.aao.org/Optical_Coherence_Tomography_Angiography.
- [10] Kristen M. Meiburger et al. «Automatic skin lesion area determination of basal cell carcinoma using optical coherence tomography angiography and a skeletonization approach: Preliminary results». In: *Journal of Biophotonics* (mag. 2019). DOI: 10.1002/jbio.201900131.
- [11] Jiwoo Kang et al. «3D Active Vessel Tracking Using an Elliptical Prior». In: *IEEE Transactions on Image Processing* 27.12 (dic. 2018), pp. 5933–5946. DOI: 10.1109/TIP.2018.2862346.
- [12] *Rodrigues' Rotation Formula*. URL: <http://mathworld.wolfram.com/RodriguesRotationFormula.html>.
- [13] *Compactness*. URL: <http://37steps.com/738/compactness/>.
- [14] Motong Qiao e K. Michael Ng. «Tone mapping for high-dynamicrange images using localized gamma correction». In: *Journal of Electronic Imaging* 24.11 (gen. 2015).

-
- [15] *DBSCAN Clustering Algorithm*. URL: <https://practice2code.blogspot.com/2017/07/dbscan-clustering-algorithm.html>.
- [16] *A Tutorial on Clustering Algorithms*. URL: https://home.deib.polimi.it/matteucc/Clustering/tutorial_html/kmeans.html.
- [17] Ta-Chih Lee e Rangasami L. Kashyap. «Building Skeleton Models via 3-D Medial Surface/Axis Thinning Algorithms». In: *Graphical Models and Image Processing* 56.6 (ago. 1994).
- [18] Hoonyoung Jeong e Sanjay Srinivasan. «Fast selection of geologic models honoring CO2 plume monitoring data using Hausdorff distance and scaled connectivity analysis». In: *International Journal of Greenhouse Gas Control* 59 (apr. 2017), pp. 40–57. DOI: 10.1016/j.ijggc.2017.02.005.
- [19] Kornelia Kliś et al. «Computer-aided analysis of middle cerebral artery tortuosity: Association with aneurysm development». In: *Journal of Neurosurgery* 130.2 (mag. 2018), pp. 1–7. DOI: 10.3171/2017.12.JNS172114.

Ringraziamenti

Concludendo questo lavoro di tesi, sento la necessità di ringraziare non solo chi ha contribuito alla buona riuscita di questo elaborato ma anche chi mi ha accompagnata nel lungo percorso accademico, permettendomi di crescere e migliorarmi sotto ogni aspetto.

Vorrei ringraziare il Prof. Filippo Molinari per avermi permesso di svolgere questo lavoro di tesi. Ringrazio la Prof.ssa Kristen Mariko Meiburger per la costanza con la quale mi ha guidato attraverso le difficoltà. Grazie all'Ing. Nicola Michielli per i suoi innumerevoli consigli, fondamentali per la buona riuscita di questa tesi.

Ringrazio Noemi, Silvia, Dani, Franceschini, Ale, Elisa, Michelone, tutti i dottorandi e borsisti che hanno sostato anche solo per un minuto nel Biolab durante i miei mesi di tesi per aver portato leggerezza, tante pause caffè e spritz alle mie giornate ingrigite dallo sconforto che ogni tanto sopraggiungeva. Grazie a questo Bioclub per la spensieratezza con la quale ho portato avanti il lavoro di tesi. Spero di trovare in futuro un ambiente lavorativo accogliente come quello trovato con voi.

Un grande grazie va a Luca, mio grande amico, porto sicuro nei momenti difficili, un grande motivatore, sei stato per me fonte di ispirazione per vivere la vita un viaggio alla volta, riscoprendo il mondo con occhi nuovi. Grazie a te ho compreso il vero valore dell'amicizia.

Grazie a Monica per la continua presenza, per le tante sere passate in sella alle nostre Lelle, per il supporto che mi hai sempre dato su quel divano in via Mombasiglio 99, per l'amicizia vera che mi hai dimostrato in questi anni di Poli e anche ora che sei dall'altra parte del mondo.

Grazie a Tania e Agnese per essere state parte fondamentale della mia famiglia torinese assieme a Monica, siete state amore, gioia, presenza ma anche tanto disagio. Il disagio è qualcosa che ci ha sempre accompagnate in ogni nostra serata o avventura in quelle aule grigie del Poli. Porterò sempre con me un bellissimo ricordo di questi anni passati insieme.

Ringrazio Roberta per essere stata una compagna di sventure e di ansie al Poli, non pochi progetti abbiamo affrontato insieme ma siamo sempre uscite vincitrici in un modo o nell'altro.

Ringrazio la mia sorellina Maria Luisa per avermi supportato e sopportato per venticinque anni ma in questi ultimi sei ancor di più, per aver ascoltato ogni mio pianto, per aver gioito per ogni mia vittoria, per la costante presenza e per essere sempre stata per me un riferimento fondamentale.

Ringrazio Carmen e Antonio per l'affetto che mi avete sempre dimostrato, per il costante supporto in questi anni di studio anche quando con delle occhiaie profondissime mi obbligavate ad andare al mare perchè solo dopo un po' di riposo si può studiare meglio.

Grazie ai miei fratelli gemelli Gabriele e Michele, diversi sotto ogni aspetto ma a modo loro sempre presenti, uno super attivo e l'altro un po' meno. Anche se con voi ho concluso questi sei anni con un numero di capelli bianchi elevato e un livello di esaurimento altissimo, in fin dei conti non avrei portato con me i bellissimi ricordi di questi anni passati insieme da coinquilini.

Un profondo grazie va alla mia mamma e al mio papà per avermi insegnato tutto, per avermi guidato nella mia vita tracciando la via ma lasciandomi sempre libera di scegliere qualunque deviazione e percorso alternativo. Per non avermi mai fatto mancare il supporto economico e soprattutto emotivo necessario per affrontare questo lungo percorso accademico. Per avermi insegnato a rialzarmi sempre dopo una sconfitta e che le cadute non finiranno mai nella vita. Per avermi insegnato che il mondo non è tutto rose e fiori come nelle favole ma bisogna rimboccarsi le maniche sempre e comunque, e che gli sforzi prima o poi vengono ripagati con la più dolce delle ricompense.

Ringrazio tutti voi, perchè il panorama dalla cima è meraviglioso ma se in vetta ci arrivi con qualcun'altro a cui vuoi bene, lo spettacolo è ineguagliabile.