



Elaborazione di immagini mediche

Laurea Magistrale in Ingegneria Biomedica

A.A. 2021-2022

## Vessel Wall Segmentation Challenge

Rigazio Sofia (282247), Roccaro Lucia (289423),  
Romano Anastasio (289707), Ruzzante Elena (292194)

# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	Descrizione del problema . . . . .	1
1.2	Descrizione generale della soluzione proposta . . . . .	1
<b>2</b>	<b>Descrizione del Dataset</b>	<b>1</b>
2.1	Correzione dei <i>label</i> del dataset originale . . . . .	2
2.2	Creazione delle maschere e riorganizzazione del dataset . . . . .	2
<b>3</b>	<b>Implementazione della soluzione proposta</b>	<b>3</b>
3.1	Preprocessing . . . . .	3
	Adattamento delle maschere alla costruzione di una rete multi-classe . . . . .	3
	<i>Centering</i> dell'immagine . . . . .	4
	Controllo sulle dimensioni delle immagini e <i>resize</i> . . . . .	4
	<i>Flip left-right</i> delle ICAL/ECAL e <i>crop</i> dell'immagine . . . . .	4
	Rimozione delle biforcazioni . . . . .	5
	Definizione della <i>Region Of Interest</i> . . . . .	5
	Preparazione delle immagini da fornire alle reti . . . . .	6
3.2	Training . . . . .	6
3.2.1	Allenamento ICA ed ECA . . . . .	6
3.2.2	Parametri della rete . . . . .	6
3.2.3	Data augmentation . . . . .	7
3.2.4	Architettura della rete . . . . .	7
3.3	Post-processing . . . . .	7
	Estrazione delle maschere predette dalla rete . . . . .	8
	Rimozione dello <i>zero padding</i> . . . . .	8
	Riposizionamento della ROI nell'immagine intera, alle coordinate opportune . . . . .	8
	<i>Flip left-right</i> delle maschere ICAL/ECAL . . . . .	8
	<i>Resize</i> per immagini di dimensioni diverse da quelle standard . . . . .	8
	<i>De-Centering</i> . . . . .	8
	Rimozione piccoli oggetti, piccoli buchi e segmentazioni non complete . . . . .	8
<b>4</b>	<b>Risultati</b>	<b>9</b>
4.1	Valutazione delle prestazioni . . . . .	9
<b>5</b>	<b>Conclusioni</b>	<b>10</b>
5.1	Analisi critica delle limitazioni . . . . .	10
5.2	Sviluppi futuri . . . . .	10
	<b>Bibliografia e riferimenti</b>	<b>11</b>

# 1 Introduzione

## 1.1 Descrizione del problema

L'arterosclerosi è globalmente una delle principali cause di mortalità. È caratterizzata dalla presenza di placche parietali che invadono progressivamente il lume delle arterie di medio e grosso calibro, originate si per diverse concause ed aventi una composizione disomogenea (generalmente contengono lipidi, cellule infiammatorie, colesterolo, cellule muscolari lisce e tessuto connettivo). Le placche possono divenire un problema protrudendo nel lume vasale, ostacolando il flusso sanguigno. Talvolta possono lacerarsi, portando a trombosi e/o ad un'obliterazione completa del lume del vaso, interrompendone il flusso ematico. Uno dei siti clinicamente più comuni e significativi di arterosclerosi è l'arteria carotide [2], [9].

La valutazione della morfologia e dello spessore della parete dei vasi carotidi è uno step fondamentale per il monitoraggio dei pazienti: anomalie nella morfologia dei vasi sono infatti precoci segnali di alterazioni collegate a questa patologia. Nelle grandi arterie, specialmente nelle arterie carotidi, la risonanza magnetica (MRI) si è dimostrata particolarmente adatta a descrivere le caratteristiche morfologiche (volume, area, spessore), strutturali e di composizione della placca aterosclerotica *in vivo* [10]. In particolare, la risonanza magnetica ad alta risoluzione delle pareti dei vasi (High Resolution VW-MRI) è il metodo ad oggi più utilizzato e studiato per le arterie carotidi.

Il controllo manuale delle immagini di risonanza magnetica risulta oneroso ed ha una bassa riproducibilità inter- e intra-operatore. Soluzioni a segmentazione automatica (o semi-automatica), sia in modalità 2D (quindi slice per slice) sia gestendo volumi 3D, sono ampiamente utilizzate in VM-MRI e sono tuttora oggetto di numerosi studi per migliorarne le prestazioni [1], [10], [13], [14]: la segmentazione automatica risulta infatti ancora difficoltosa soprattutto in presenza di geometrie particolarmente complesse.

La *challenge* di riferimento per questo studio [6] ha lo scopo di cercare una modalità di segmentazione automatica della parete dell'arteria carotide con un'elevata precisione e robustezza, suddividendola in lume e parete esterna. La segmentazione di lume e parete della carotide è infatti il primo - ma fondamentale - step per poter misurare lo spessore della parete esterna del vaso (o altri parametri ad esso connessi), al fine di individuare eventuali anomalie.

Le immagini forniteci sono state acquisite con la tecnica MRI 3D-MERGE (Multiple Echo Recombined Gradient Echo): essa sfrutta la combinazione di più segnali per generare immagini ad alta risoluzione, mantenendo tempistiche di scansione molto brevi (dell'ordine di appena un paio di minuti) [6], [8].

## 1.2 Descrizione generale della soluzione proposta

La soluzione da noi proposta comprende una prima fase di preprocessing, descritta nella sottosezione 3.1, nella quale vengono estratte due ROI centrate rispettivamente in corrispondenza di carotide interna e esterna, entrambe sul lato destro del soggetto. Sfruttando infatti la simmetria rispetto all'asse verticale dell'immagine, tutte le carotidi sinistre vengono specchiate verticalmente (mediante un *flipping*), permettendoci di localizzare unilateralmente la ROI.

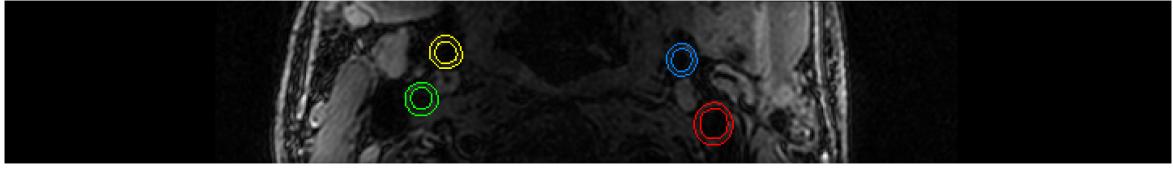
Successivamente, come dettagliato nella sottosezione 3.2, le ROI delle carotidi interne (ICAx) vengono utilizzate per allenare una rete che le segmenti; una seconda rete, con la stessa architettura della precedente, viene invece usata per la segmentazioni delle carotidi esterne (ECAx). Per entrambe le reti i pesi vengono inizializzati a partire da un pre-allenamento di una rete sulle carotidi interne per 5 epoche. Dopo la predizione, le maschere vengono riportate alle dimensioni originali ed, eventualmente, ri-specchiate, permettendo di localizzare correttamente la segmentazione a destra o a sinistra del soggetto. Il post-processing effettuato è descritto in dettaglio nella sottosezione 3.3.

# 2 Descrizione del Dataset

Il dataset originale è composto in totale da 50 volumi (35 per Training set, 5 per Validation set e 10 per Test Set), ciascuno contenente un numero variabile di slice. Per ogni soggetto, sono inoltre disponibili alcune segmentazioni manuali della parete esterna e del lume di carotide interna ed esterna sinistra (ICAL/ECAL) e destra (ICAR/ECAR).

In totale, le slice a nostra disposizione sono rispettivamente 2138 ICAR, 1796 ICAL, 204 ECAR, 276 ECAL per il Training Set; 292 ICAR, 300 ICAL, 21 ECAR, 117 ECAL per il Validation Set. Lo sbilanciamento del numero di segmentazioni manuali in carotidi esterne e interne è in parte dovuto al fatto che la sezione comune delle carotidi, caudale alla biforcazione, è generalmente segmentata come carotide interna.

Un esempio di visualizzazione di una slice con corrispondenti maschere manuali è riportato in Figura 1.



**Figura 1:** Visualizzazione di una slice (Soggetto P20, slice EP20S101I317) a cui sono stati sovrapposti i contorni delle segmentazioni manuali di tutte le carotidi.  
Codifica colore: rosso ICAL, blu ECAL, verde ICAR, giallo ECAR.

## 2.1 Correzione dei *label* del dataset originale

Visualizzando la sovrapposizione delle maschere manuali alle immagini, sono emersi degli errori di assegnazione dei *label* per alcune segmentazioni. Nella maggior parte dei soggetti in cui si riscontrano le incongruenze, la segmentazione della carotide esterna veniva erroneamente salvata nella cartella della corrispondente carotide interna (solo in rari casi accadeva anche il contrario).

Per non incorrere in un peggioramento delle prestazioni dovuto ad un errato *labeling* delle maschere manuali, è stato ritenuto opportuno riassegnare le maschere al loro *label* corretto. Questa operazione viene effettuata nello script `A_ReadCASCADEcontours` mediante la funzione `cleaning`. Le modifiche hanno coinvolto i soggetti P15, P33 e P42 del Training set e i soggetti P4, P19, P27 e P36 del Validation set e sono riportate in Tabella 1. Inoltre, il soggetto P16 è stato rimosso in quanto, pur facendo parte del Training Set, non presenta segmentazione manuale di alcuna carotide.

TRAINING SET		
Soggetto	Incongruenza riscontrata	Modifica al dataset
P15	Le maschere ICAR fino alla slice EP15S101I363 (inclusa) dovrebbero essere nella cartella ECAR	riassegnazione alla cartella ECAR
P16	Nessuna maschera manuale presente	rimosso dal dataset
P33	Le slice EP33S101I: 331, 332, 336 e 337 sono rumore bianco ma presentano segmentazione manuale	rimozione di immagini e maschere
P42	La maschera ICAR di EP42S101I318 dovrebbe essere nella cartella ECAR	riassegnazione alla cartella ECAR
	La maschera ICAL di EP42S101I308 dovrebbe essere nella cartella ECAL	riassegnazione alla cartella ECAL

(a)

VALIDATION SET		
Soggetto	Incongruenza riscontrata	Modifica al dataset
P4	Le maschere ECAL fino alla slice EP4S101I235 (inclusa) dovrebbero essere nella cartella ICAL	riassegnazione alla cartella ICAL
P27	La maschera ICAR di EP27S101I259 dovrebbe essere nella cartella ECAR	riassegnazione alla cartella ECAR
	La maschera ICAL di EP36S101I314 dovrebbe essere nella cartella ECAL	riassegnazione alla cartella ECAL
P36	Le maschere ICAR di EP36S101I: 313, 316, 321, 326, 331, 336, 341, 346, 351, 356 dovrebbero essere nella cartella ECAR	riassegnazione alla cartella ECAR
P19	Le maschere ICAL di EP19S101I: 368, 360, 357, 352, 351, 347, 342, 337, 332, 330 dovrebbero essere nella cartella ECAL	riassegnazione alla cartella ECAL

(b)

**Tabella 1:** Tabella modifiche apportate sul Training Set (a) e Validation Set (b)

## 2.2 Creazione delle maschere e riorganizzazione del dataset

Nel dataset originale è presente un Qlikview Script file (.qvs) per ogni tipologia di carotide segmentata dall'operatore, al cui interno sono riportate le coordinate dei punti che costituiscono i bordi contornanti l'area riconosciuta come *"Lumen"* o *"Outer Wall"*. Le maschere vengono create a partire da queste coordinate nello script `A_ReadCASCADEcontours`. Per ogni file .qvs generiamo quindi due matrici di maschere, corrispondenti a *lumen* e *wall*.

Per una più agevole lettura dei dati del Training set e del Validation set, è stato creato un Dataset parallelo, contenuto nella cartella `3D_dataset`, in cui vengono salvate in modo ordinato soltanto le informazioni d'interesse per la nostra elaborazione. La divisione in sottocartelle di `3D_dataset` è analoga a quella del DATASET originale con la differenza che all'interno delle cartelle `xCAx` sono salvate 3 matrici 3D in formato `.tiff`, contenenti rispettivamente le maschere manuali (`lumen_mask.tiff` e `wall_mask.tiff`) e il volume di slice a cui sono associate le segmentazioni (`image.tiff`).

Le slice che presentavano la segmentazione di solo uno dei due bordi (quindi o solo *lumen* o solo *wall*) sono state escluse dal dataset.

All'interno delle matrici in `3D_dataset`, per ogni soggetto, sono state aggiunte anche delle slice in cui la carotide non è visibile e che sicuramente non necessitano di segmentazione: questa scelta è stata presa nell'ottica di allenare la rete anche a riconoscere situazioni in cui la carotide non è presente. Il numero

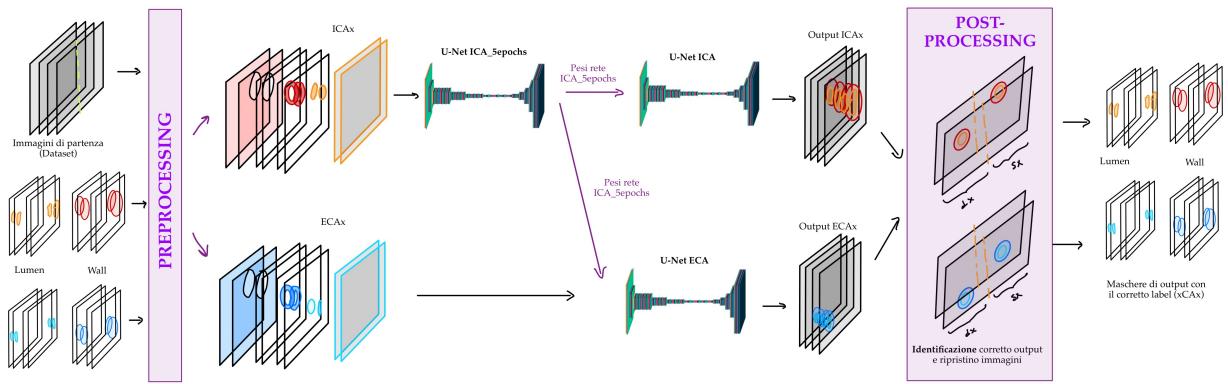
di slice da aggiungere è stato determinato sulla base dello sbilanciamento nel numero di maschere manuali presenti per le carotidi interne ed esterne.

Più in dettaglio, sono state inserite nel Training Set rispettivamente 3 slices (le numero 5, 35, 70 del file DICOM di ogni soggetto) per le carotidi interne (ICA) e 2 slices (le numero 35, 70) per le carotidi esterne (ECA). Per quanto concerne il Validation Set, è stata invece aggiunta solo una slice (la 70) indipendentemente dal fatto che fosse presente carotide interna o esterna. Alle immagini aggiunte sono state associate delle maschere completamente nere. In totale, nel Training set sono state aggiunte 234 immagini in cui non è presente alcuna carotide e 234 corrispondenti maschere nere per *lumen* e *wall*. Nel Validation set ne sono invece state aggiunte 13.

In seguito alle osservazioni emerse dalla visualizzazione del dataset che hanno portato alle modifiche appena descritte, la struttura del dataset è cambiata come segue:

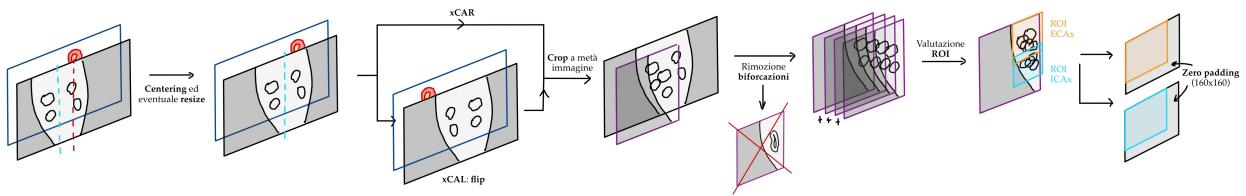
- il numero di soggetti del Training set è pari di 34 soggetti (e non più 35, dopo l'esclusione di P16);
- il numero di slice nel Training set per ogni carotide è diventato rispettivamente 2209 ICAR, 1889 ICAL, 246 ECAR e 295 ECAL;
- il numero di slice nel Validation set per ogni carotide è invece pari a 286 ICAR, 300 ICAL, 34 ECAR e 124 ECAL.

### 3 Implementazione della soluzione proposta



**Figura 2:** Schematizzazione pipeline dell'intero progetto

#### 3.1 Preprocessing



**Figura 3:** Schematizzazione pipeline processi della fase di preprocessing

**Adattamento delle maschere alla costruzione di una rete multi-classe** Al fine di creare una rete multi-classe che riconosca contemporaneamente sfondo, *lumen* e *wall* vengono realizzate tre maschere, rispettivamente:

- Una prima maschera per il lume
- Una seconda maschera contenente solo la segmentazione della parete, ottenuta per differenza tra le maschere *wall* e *lumen*
- Una terza maschera per il background, che esclude sia l'area del lume sia quella della parete

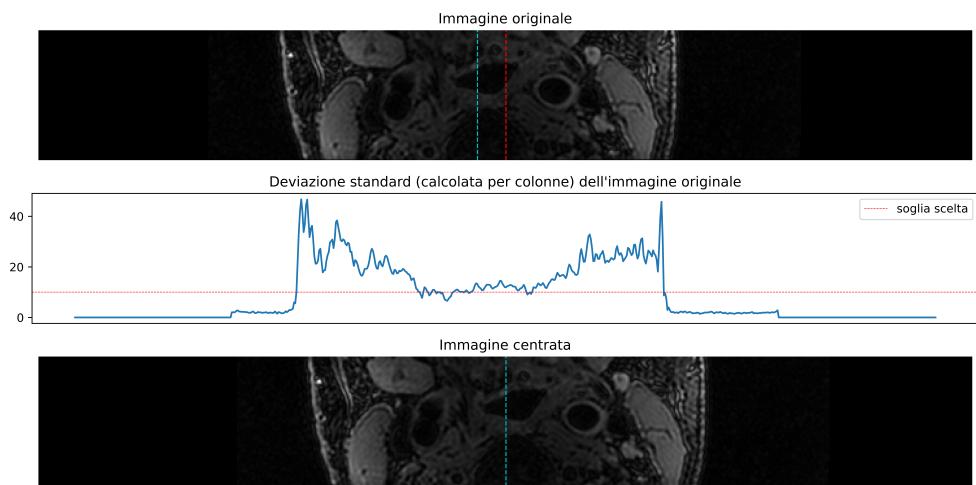
**Centering dell’immagine** È stato osservato che, in alcuni soggetti, le sezioni del collo acquisite non sono perfettamente allineate con il centro dell’immagine. Per semplificare i successivi passaggi di identificazione delle ROI, è nostro interesse sfruttare la simmetria sul piano sagittale dell’area di studio. È stato quindi implementato un algoritmo che effettua uno *shift* dell’immagine in modo da far corrispondere l’asse di simmetria della sezione del collo con l’asse di simmetria verticale della slice.

L’analisi è stata eseguita per ogni soggetto a partire da un’immagine prelevata al centro del volume di slice in esame e considerata come rappresentativa per il soggetto. In particolare, per ogni colonna della slice in esame, è stata valutata la deviazione standard dei valori dei pixel in essa contenuti: la deviazione standard infatti, a differenza della media o della somma, fornisce un’informazione legata alla variazione relativa tra pixel della stessa colonna.

Poiché tutte le immagini presentano delle colonne nere sia a destra che a sinistra, ci aspettiamo che esternamente la deviazione standard sia molto piccola in quanto il colore è piuttosto omogeneo e varia molto poco attorno allo zero (sfondo nero). La discontinuità tra tessuto e sfondo, invece, fa in modo che la deviazione standard abbia valori maggiori in corrispondenza dell’inizio del collo, in cui in una colonna sono presenti sia valori molto elevati appartenenti al tessuto sia valori circa nulli appartenenti allo sfondo.

L’inizio e la fine del tessuto biologico sono stati quindi identificati come il primo e l’ultimo valore significativo di deviazione standard sulle colonne (imponendo una soglia di  $\sigma > 10$ ): è possibile quindi valutare lo scostamento della sezione d’interesse rispetto al centro dell’immagine e conseguentemente spostare alcune colonne marginali di sfondo nel lato opposto. Questa operazione non comporta problemi perché le colonne spostate sono quasi completamente nere e verranno in seguito escluse dalla ROI.

L’operazione di *centering* qui descritta viene effettuata nello script `A_ReadCASCADEcontours` per tutte le immagini del dataset (Training, Validation e Test set), ed un esempio è visibile in Figura 4.



**Figura 4:** Esempio di valutazione dell’asse di simmetria del collo (in azzurro) rispetto al centro dell’immagine (asse in rosso) e conseguente *centering* dell’immagine. Soggetto P18 (Training Set) - slice EP18S101I389

**Controllo sulle dimensioni delle immagini e `resize`** Alcuni soggetti (rispettivamente P8, P13, P31, P32, P49 per il Training Set, e P11, P29, P45 per il Test Set) presentano immagini di dimensioni pari a  $160 \times 640$  pixel, diversamente da tutti gli altri che invece hanno dimensioni di  $100 \times 720$  pixel. Per questi soggetti è stato effettuato per un `resize` delle slice e delle maschere ad esse associate, in modo da ottenere matrici di uguali dimensioni per tutti i soggetti ( $100 \times 720$  pixel). Per non modificare l’*aspect ratio*, è stato utilizzato il comando `resize_with_pad` di `tensorflow.image` che, prima di effettuare il `resize`, effettua uno *zero padding* in modo da uguagliare le proporzioni dell’immagine a quella desiderata.

**Flip left-right delle ICAL/ECAL e crop dell’immagine** Sfruttando la simmetria sul piano sagittale dell’area in analisi, è stata attuata una strategia che permettesse di lavorare solo con metà immagine [1], tagliata in corrispondenza dell’asse centrale verticale. Questo è stato realizzato mediante due step successivi:

1. Per tutte le immagini e le maschere presenti nelle cartelle `xCAL` (che presentano segmentazione della carotide sinistra, la quale si trova nella metà destra dell’immagine) viene effettuato un *flip* sull’asse verticale. In questo modo, in ognuno dei due casi (`xCAL` e `xCAR`), la zona di interesse si trova nel lato sinistro dell’immagine.

2. Tutte le immagini e le maschere vengono tagliate verticalmente a metà, in modo da mantenerne solamente il lato sinistro.

Grazie a questa strategia potremo allenare un'unica rete per le carotidi interne e una seconda rete per le carotidi esterne, senza fare distinzione tra lato destro e sinistro. Questo ci consente di aumentare il numero di immagini utili per l'allenamento della rete.

Le maschere automatiche ottenute in output dalle due reti vengono automaticamente assegnate al corrispondente *label*: rispettivamente ICA se segmentate dalla rete allenata sulle carotidi interne ed ECA se segmentate dalla rete allenata sulle carotidi esterne.

Da qui in avanti considereremo quindi le xCAx indipendentemente dal lato. L'assegnazione della segmentazione automatica a xCAL o xCAR verrà effettuata in fase di post-processing.

**Rimozione delle biforazioni** Per la rimozione delle slice contenenti biforazioni, sono stati utilizzati due parametri calcolati sulle maschere manuali: un coefficiente di forma (*roundness*, si veda Equazione 1) e un coefficiente legato alla dimensione (area del lumen) della carotide.

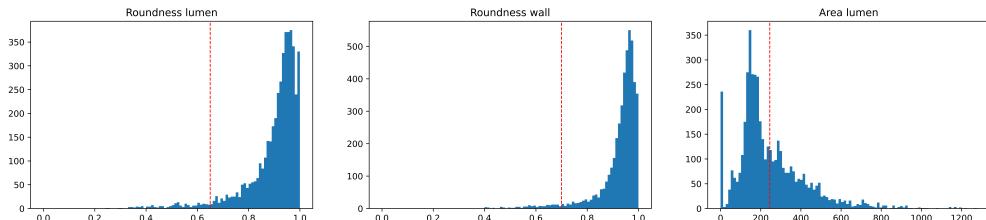
La *roundness* sarà circa pari a 1 per strutture perfettamente circolari, mentre è sempre più prossima a 0 man mano che la struttura diviene allungata.

$$roundness = 4 * \frac{Area}{\pi * MajorAxis^2} \quad (1)$$

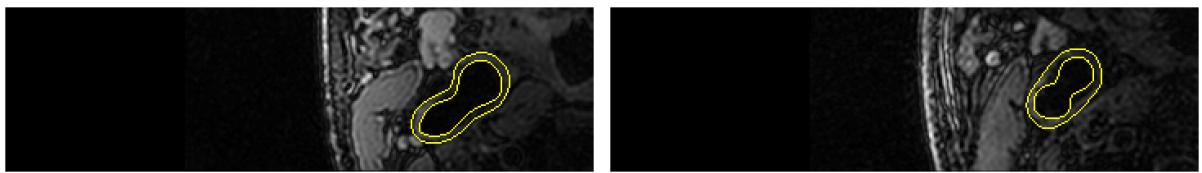
Il parametro di area è stato considerato a complementazione di quello di roundness. Le biforazioni, comprendendo due vasi, presentano oltre alla forma allungata un'area del lume nettamente maggiore.

La rimozione dalle slice contenenti biforazioni dal dataset è stata effettuata quando vengono soddisfatte entrambe le seguenti condizioni, scelte a partire dalla visualizzazione degli istogrammi mostrati in Figura 5:

- Il coefficiente di *roundness* di *lumen* e *wall* è inferiore ad una certa soglia: si è scelto di soddisfare le condizioni rispettivamente  $\text{coeff\_roundness} < 0.7$  per la parete, e  $\text{coeff\_roundness} < 0.65$  per il lume.
- L'area del *lumen* deve invece risultare maggiore di una soglia, posta pari a 245 pixel. Questo valore è stato scelto per escludere il picco visibile in corrispondenza di un area di circa 300 pixel.



**Figura 5:** Istogramma dei coefficienti di *roundness* e del valore d'area del *lumen* per il Training set. Da qui sono state scelte le soglie per la rimozione delle biforazioni (segnalate in rosso).



(a) Soggetto P18 (L-side), slice EP18S101I363

(b) Soggetto P12 (R-side), slice EP12S101I367

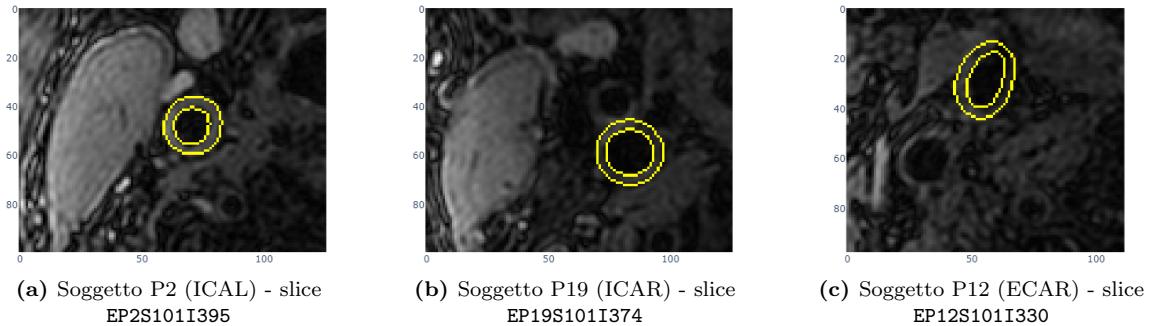
**Figura 6:** Esempi di biforazioni individuate e rimosse dall'algoritmo

**Definizione della *Region Of Interest*** Partendo dalle immagini e dalle maschere rielaborate come precedentemente descritto, per ogni tipologia di carotide (ICA ed ECA) viene definita una Region of Interest (ROI). La ROI ricercata nelle immagini è identificabile come il minimo rettangolo contenente tutte le segmentazioni manuali appartenenti al Training Set, aumentato del 10% per lato. La maggiorazione delle dimensioni viene effettuata per essere certi di comprendere anche tutti i casi degli altri Set.

Se, in seguito alla maggiorazione di dimensioni, la ROI sfiora i bordi dell'immagine, il limite viene settato al bordo dell'immagine stessa.

Il risultato di questo algoritmo è una ROI avente dimensioni  $100 \times 126$  pixel.

Per le ECA è stata ripetuta la stessa procedura: in questo caso la ROI ottenuta ha dimensioni pari a  $98 \times 112$  pixel.



**Figura 7:** Esempi di ROI individuate per Training (a) e Validation Set (b,c) ed utilizzate nelle fasi successive

**Preparazione delle immagini da fornire alle reti** Volendo utilizzare un modello di rete fornito da Keras [7], è necessario dare in input delle immagini le cui dimensioni siano multiple di 32 [16], preferibilmente quadrate. Nel nostro caso è stata scelta una dimensione di 128 pixel per lato, in quanto esso è il multiplo di 32 immediatamente superiore alle larghezze delle ROI (126 pixel per ICA e 112 pixel per ECA). Per portare le immagini alle dimensioni desiderate si è scelto di effettuare uno *zero padding* piuttosto che un *resize*, in modo da non distorcere le immagini né creare valori non conformi per le maschere (che, al seguito dell'applicazione di questa tecnica, avrebbero potuto differire da 0 e 1).

## 3.2 Training

### 3.2.1 Allenamento ICA ed ECA

Per la segmentazione automatica di carotidi interne ed esterne sono state allenate due reti distinte. In entrambi i casi, per l'allenamento è stato scelto di partire da un modello preallenato sulle ICA.

In primo luogo, infatti, è stata allenata una rete sulle immagini delle carotidi interne per 5 epoche e senza imporre alcuna condizione di stop. I pesi di output di questa rete sono stati utilizzati come pesi di partenza per allenare due reti successive, una per la segmentazione delle ICA e una per le ECA. Si è osservato che un maggior numero di epoche di preallenamento sulle ICA induceva la rete ECA a segmentare anche le carotidi interne, anzichè specializzarsi.

### 3.2.2 Parametri della rete

Per tutte le reti, come funzione di *loss* è stata utilizzata la `categorical_crossentropy`, ovvero la funzione prediletta nel caso di classificazioni multi-classe. La funzione di attivazione utilizzata è la `softmax`, la quale viene generalmente accoppiata alla funzione di *loss* scelta ed è anch'essa appositamente progettata per attività di classificazione multi-classe. Come metrica per la valutazione delle performance della rete è stata scelta la `categorical_accuracy`. Come *optimizer*, per la rete è stato scelto l'algoritmo `Adam`, noto per la sua rapidità di convergenza e le buone prestazioni in numerose applicazioni [4], [5], [12].

Il massimo numero di epoche di allenamento per le due reti (ICA ed ECA) è stato settato pari a 40, ma le condizioni di stop imposte per evitare *overfitting* sono risultate essere più stringenti ad ogni allenamento. In particolare, il parametro da monitorare impostato in `EarlyStopping` è la *categorical accuracy* sul Validation set (`val_categorical_accuracy`): si è imposto alla rete di interrompere l'allenamento qualora non si verificasse un miglioramento di questo parametro per un numero prestabilito di epoche successive (valore di `patience`). Il minimo incremento di *categorical accuracy* identificato come miglioramento è stato impostato pari allo 0.5% (`min_delta = 0.005`). Sono state effettuate delle prove anche con valori di `min_delta` pari a 0.01, senza ottenere significative variazioni delle performance.

Sono stati testati valori di `patience` compresi tra 4 e 16. Valori di `patience` inferiori a 8 sono però risultati essere troppo restrittivi per l'allenamento: le reti infatti si allenavano poco e non raggiungevano prestazioni soddisfacenti. Al contrario, valori di `patience` superiori a 12 portavano ad *overfitting*, e quindi a prestazioni visibilmente scarse sul Test set.

La soluzione ottimale per la rete adibita alla segmentazioni delle ICA è risultata essere quella avente `patience` pari a 8. Per la rete ECA, dato l'esiguo numero di dati in input, è stata scelta una `patience` leggermente superiore, pari a 12, per permetterle di allenarsi più a lungo.

Per una visione schematica di tutti i parametri impostati, si faccia riferimento alla Tabella 2.

La dimensione del *batch*, ovvero il numero di immagini e corrispondenti maschere che la rete prende in input ad ogni iterazione, è stato scelto considerando che, generalmente, il `batch_size` utilizzato per circa 1000 immagini in input è pari a 32. Inoltre, *batch* di dimensioni maggiori determinano progressi più rapidi nell'allenamento, ma non sempre convergono altrettanto velocemente: pertanto, si prestano bene in dataset di grandi dimensioni. *Batch* più piccoli si allenano invece più lentamente, ma possono convergere più velocemente: sono quindi preferibili per dataset di dimensioni minori [11].

Dato che per la rete ICA si ha un gran numero di dati in input (circa 4000 immagini), è stato impostato per essa un `batch_size` pari a 64. Per la rete ECA, poiché il numero di immagini a disposizione è pari a circa 500, il *batch* considerato viene invece posto pari a 16.

	Immagini in input	Max num di epoche effettuabili	Early Stopping conditions			Epoche effettuate	Dimensione batch	Altri parametri comuni a tutte le reti
			Parametro monitorato	min delta	Patience			
Rete Preallenamento	ICA	5	-	-	-	5	64	<ul style="list-style-type: none"> <li>- Adam optimizer</li> <li>- Softmax activation function</li> <li>- Categorical cross entropy loss function</li> </ul>
Rete ICA	ICA	40	Categorical Accuracy sul Validation Set	+0.005	8 epoche	15	64	
Rete ECA	ECA	40	Categorical Accuracy sul Validation set	+0.005	12 epoche	17	16	

**Tabella 2:** Parametri delle reti

### 3.2.3 Data augmentation

Poiché le carotidi possiedono una disposizione spaziale ben precisa si è ritenuto inopportuno effettuare un significativo *data augmentation*. L'unica operazione effettuata in questo senso è stata una rotazione delle immagini, dato che alcune carotidi possono risultare leggermente ovali con asse maggiore orientato diversamente nelle varie immagini. Il range scelto per le rotazioni randomiche è pari a  $\pm 15^\circ$ , e permette di aggiungere variabilità ed evitare *overfitting*.

### 3.2.4 Architettura della rete

Considerando che le reti CNN sono le più utilizzate nell'ambito dell'imaging biomedicale [15], è stato implementato un algoritmo in grado di verificare quale tra le architetture CNN disponibili nella libreria Keras sia la più adatta al problema proposto e al dataset ad esso associato [3].

A questo scopo, sono state selezionate alcune tra le reti più utilizzate fornite da Keras. Ciascuna rete è stata allenata dando in input il 30 % del dataset di Training e Validation set, impostando il *pool* di parametri scelti, descritti in sottosezione 3.2.2, ed effettuando sulle immagini il pre-processing descritto in sottosezione 3.1. L'unica significativa differenza è che, in questa fase, le reti sono state allenate per 3 epoche ciascuna senza imporre alcuna condizione di stop.

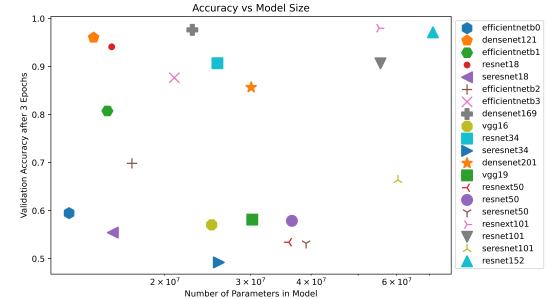
I criteri utilizzati per la scelta del modello sono:

- accuratezza più alta;
- maggiore velocità di addestramento, inversamente proporzionale al numero di parametri del modello.

Dai risultati, mostrati in Figura 8, le reti `densenet121` e `resnet18` sono risultate le migliori: sono state pertanto testate sull'intero Dataset.

## 3.3 Post-processing

Per il post-processing è stata ripercorsa la pipeline di preprocessing (Figura 3) in verso opposto, in modo da ripristinare le immagini nella loro forma e dimensione originale. In aggiunta, è stata effettuata una pulizia delle maschere automatiche ottenute in output dalle reti. Tutti gli step di post-processing effettuati sono descritti qui in seguito.



**Figura 8:** Accuratezza sul Validation set in funzione del numero di parametri dei modelli forniti da Keras

**Estrazione delle maschere predette dalla rete** Dopo la predizione viene effettuata una sogliatura (soglia pari a 0.5) sulle maschere di lume e parete, in modo da massimizzare la distanza tra i valori attesi (0 e 1). Per fare in modo che le maschere corrispondenti alla parete del vaso fossero cerchi pieni, esse sono state ottenute sommando la maschera restituita dalla rete per *wall* (corona circolare) e *lumen* (cerchio pieno).

**Rimozione dello zero padding** In questa fase sono state rimosse le righe e le colonne nere che erano state aggiunte in fase di preprocessing per rendere le immagini da fornire alla rete di dimensioni multiple di 32.

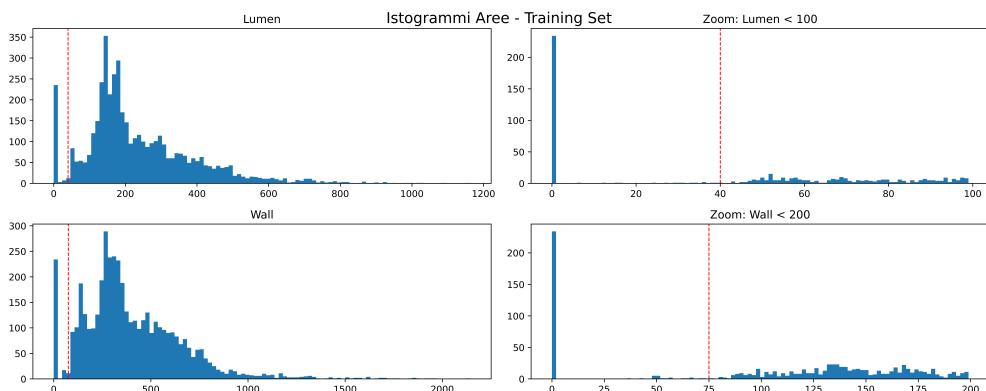
**Riposizionamento della ROI nell'immagine intera, alle coordinate opportune** Le maschere automatiche contenute nelle ROI sono state ricollocate alle coordinate originali rispetto alle dimensioni delle immagini di partenza.

**Flip left-right delle maschere ICAL/ECAL** Le maschere relative alle carotidi del lato sinistro (posizionate originariamente a destra dell'immagine) sono state *flippeate* nuovamente rispetto all'asse verticale, in modo da riportare la segmentazione sul lato corretto del collo.

**Resize per immagini di dimensioni diverse da quelle standard** Le maschere dei soggetti le cui dimensioni originali sono diverse da  $100 \times 720$  pixel sono state riportate alle dimensioni di partenza ( $160 \times 640$  pixel).

**De-Centering** È stato effettuato per ogni maschera uno shift in verso opposto rispetto a quello applicato in fase di preprocessing. In questo modo la posizione della maschera automatica torna ad allinearsi con la corrispondente immagine originale.

**Rimozione piccoli oggetti, piccoli buchi e segmentazioni non complete** In aggiunta alle fasi precedentemente descritte nella pipeline di preprocessing, ripercorse in verso opposto, è stato attuato un perfezionamento delle maschere automatiche rimuovendo piccole aree erroneamente segmentate dalla rete, e riempiendole nel caso in cui fossero presenti dei piccoli buchi. L'operazione è stata eseguita attraverso le apposite funzioni `remove_small_objects` e `remove_small_holes` tratte dalla libreria `skimage.morphology`.



**Figura 9:** Istogrammi utilizzati per la scelta delle soglie di `small_holes` e `small_objects`. In rosso sono evidenziati i due valori scelti

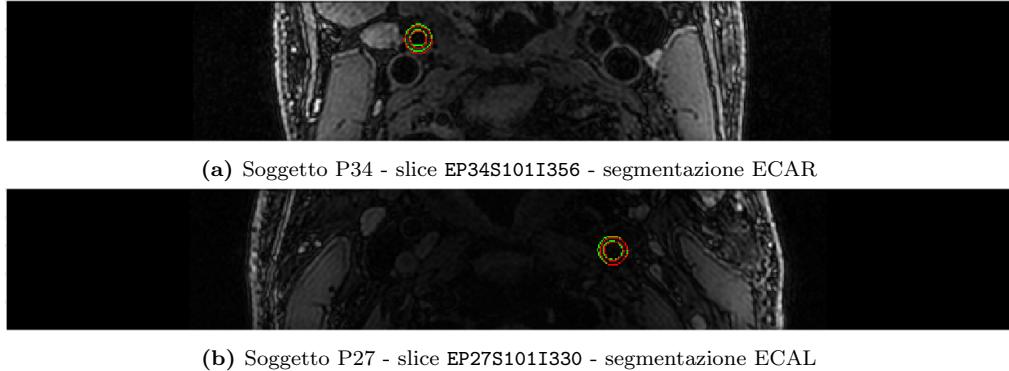
La soglia di area al di sotto della quale le segmentazioni sono considerate "piccoli oggetti" da rimuovere è determinata a partire dagli istogrammi delle aree di *lumen* e *wall* di tutte le segmentazioni manuali appartenenti al Training set (si veda Figura 9). Le soglie sono state posizionate al di sotto delle più piccole aree rilevate, escludendo poche eccezioni: rispettivamente 75 pixel per le pareti e 40 pixel per i lumi.

Per il riempimento dei piccoli buchi si è optato per una soglia ragionevolmente piccola, pari a 10 pixel.

In aggiunta, qualora la rete estragga per una slice soltanto una tra le due segmentazioni di lume e parete, la predizione è stata considerata errata e nessuna delle due maschere è stata salvata.

## 4 Risultati

In base alle prestazioni ottenute, tra le due architetture di rete identificate come migliori alla sottosottosezione 3.2.4 è stata scelta la `resnet18`. Alcuni esempi di risultati ottenuti utilizzando questo modello di rete sono visibili in Figura 10.



**Figura 10:** Esempi di risultati ottenuti per Training (a) e Validation (b) Set.

In verde sono contornate le segmentazioni manuali fornite dell'operatore, mentre in rosso è visibile la segmentazione manuale ottenuta come output della corrispondente rete neurale

### 4.1 Valutazione delle prestazioni

Le metriche utilizzate per la valutazione delle performances sono il Dice Similarity Coefficient (DSC), calcolato unicamente sulla parete del vaso (ottenuta come differenza tra le maschere *lumen* e *wall*), e il Relative Volume Difference (RVD) calcolato separatamente sul lume e sulla parete. In Equazione 2, "X" corrisponde alla maschera segmentata manualmente, mentre "Y" è la maschera automatica predetta dalla rete.

$$DSC(X, Y) = \frac{2|X \cap Y|}{|X| + |Y|} \quad RVD(X, Y) = \frac{|Y| - |X|}{|X|} \quad (2)$$

Il DSC è pari a 1 quando la segmentazione automatica della parete coincide con quella manuale, ed è nullo se nessun pixel viene correttamente segmentato. Il coefficiente RVD, al contrario, assume un valore pari a 0 quando la maschera automatica è perfettamente coincidente a quella manuale. Esso quindi tenderà ad assumere valori positivi qualora nelle maschere automatiche siano presenti segmentazioni errate, mentre avrà valori negativi qualora ci fossero segmentazioni mancanti.

Per ottenere un'indicazione quantitativa sulle prestazioni, tutti i coefficienti calcolati sulle segmentazioni ottenute da entrambe le reti neurali (ICA ed ECA) sono stati mediati, distinguendo i risultati tra carotide destra e sinistra. I risultati dalla nostra rete sono mostrati in Tabella 3.

	TRAINING SET			VALIDATION SET		
	DSC wall	RVD wall	RVD lumen	DSC wall	RVD wall	RVD lumen
Sinistra	0.8062±0.1199	-0.0016±0.1113	-0.0018±0.1532	0.6651±0.2307	-0.1160±0.2354	-0.0734±0.2491
Destra	0.7972±0.1295	-0.0126±0.1207	-0.002±0.1176	0.7710±0.1368	-0.0510±0.1286	-0.0122±0.1288

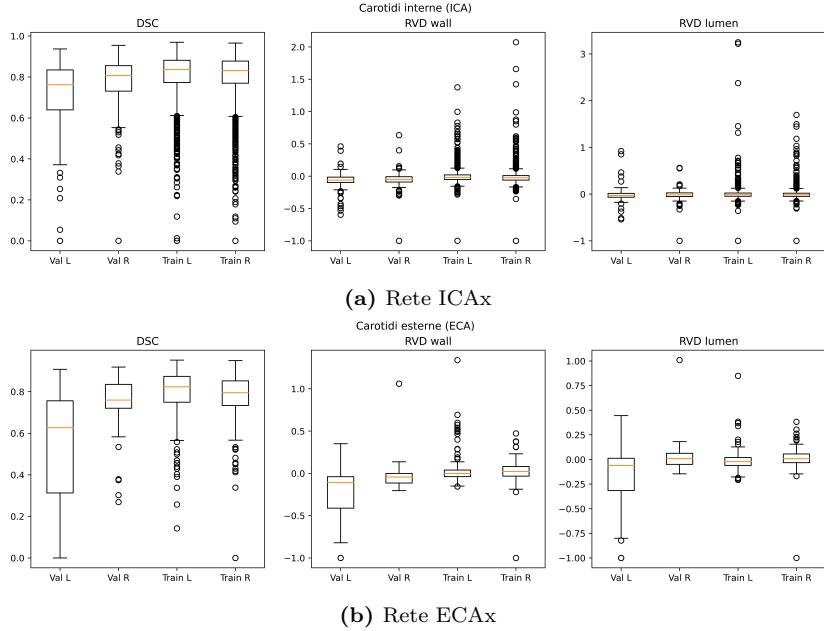
**Tabella 3:** Prestazioni calcolate su Training e Validation Set, espresse come valor medio±deviazione standard

Come parametro di valutazione aggiuntivo, le maschere automatiche sono state sovrapposte alle immagini corrispondenti e visualizzate (Figura 10). Questo ci ha permesso di evidenziare eventuali anomalie non direttamente percepibili osservando il valore dei coefficienti numerici, oltre che di effettuare un'analisi puramente qualitativa delle prestazioni della rete anche sul Test set.

Sul Training set e sul Validation Set la rete risulta sempre sufficientemente accurata: le segmentazioni sono molto simili a quelle manuali, anche se non sempre perfettamente sovrapposte, ed il vaso viene classificato correttamente nella maggior parte dei casi.

È tuttavia emerso, prevalentemente nel Test set, che la rete per la segmentazione delle ECA non sempre riesce a distinguere tra carotide esterna ed interna, ma anzi, a volte riconosce come "carotide" anche altri piccoli vasi visibili sull'immagine. La rete per la segmentazione delle ICA, invece, risulta essere molto accurata sia nell'individuare che nel segmentare la corretta carotide interna. Dai grafici mostrati in Figura 11 si può infatti osservare come per la rete ECA si ottengano prestazioni inferiori.

La carotide comune, quando presente, viene riconosciuta e correttamente segmentata da entrambe le reti.



**Figura 11:** Visualizzazione delle prestazioni delle reti tramite coefficienti DSC e RVD su Training e Validation Set

## 5 Conclusioni

### 5.1 Analisi critica delle limitazioni

Considerando il dataset a disposizione e la difficoltà del task, riteniamo che i risultati ottenuti siano soddisfacenti e coerenti con le nostre aspettative.

La maggiore difficoltà riscontrata in fase di *testing* è legata alla segmentazione automatica delle carotidi esterne. Tale problema non è tuttavia sorprendente, ed è a nostro avviso riconducibile all'esiguo numero di segmentazioni manuali ECA disponibili. Questa problematica è stata in parte ovviata inizializzando i pesi della rete ECA tramite i pesi di una rete allenata sulle carotidi interne per 5 epochhe (section 3.2.1).

I valori di DSC sono piuttosto bilanciati sul lato destro e sinistro nel Training set mentre risultano migliori (del 11 % circa) sul lato destro rispetto al sinistro nel Validation set. Si nota inoltre una buona omogeneità di risultati su Validation e Training Set con il *pool* dei parametri scelto: questa valutazione è importante in quanto indica che, probabilmente, la rete non incorre in *overfitting* dei dati.

I valori di RVD risultano globalmente bilanciati tra destra e sinistra per entrambi i set. Il loro valor medio è molto prossimo allo 0 e assume in tutti i casi valori negativi. Questo può essere indicativo del fatto che le reti tendono a non riconoscere il vaso e quindi a non segmentarlo correttamente.

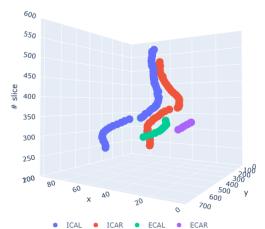
Le nostre prestazioni risultano essere in linea con quelle di altre soluzioni proposte in letteratura per la stessa Challenge ([1], [13]).

Come per ogni algoritmo di segmentazione automatico, i risultati ottenuti non possono essere ritenuti del tutto affidabili in ambito clinico: la segmentazione automatica necessita di un controllo da parte di un operatore esperto. Tuttavia, la soluzione proposta può rappresentare un punto di partenza per la segmentazione delle carotidi, velocizzando in questo modo i processi di prevenzione, diagnosi e monitoraggio dei pazienti.

### 5.2 Sviluppi futuri

Un aggiuntivo miglioramento alla soluzione proposta potrebbe essere attuato seguendo puntualmente il decorso dei due rami carotidei a partire dalla carotide comune, individuando così una ROI ancora più piccola e permettendo l'eliminazione delle segmentazioni di vasi che si discostano troppo dal percorso centrale.

L'implementazione di questo algoritmo richiederebbe tuttavia un numero elevato di maschere manuali ravvicinate (e non distanziate di circa 5 slices come quelle presenti nella maggior parte dei soggetti del Dataset), oltre che un'analisi più approfondita della traiettoria delle carotidi sul piano sagittale, risultata in prima analisi (Figura 12) anche molto diversa da soggetto a soggetto.



**Figura 12:** Traiettoria dei centroidi delle carotidi - soggetto P20

## Bibliografia e riferimenti

- [1] D. Alblas, C. Brune e J. M. Wolterink, “Deep Learning-Based Carotid Artery Vessel Wall Segmentation in Black-Blood MRI Using Anatomical Priors,” 2021. arXiv: 2112.01137 [eess.IV].
- [2] *Aterosclerosi - Disturbi dell'apparato cardiovascolare*, it-IT. indirizzo: <https://www.msdmanuals.com/it-it/professionale/disturbi-dell-apparato-cardiovascolare/arteriosclerosi/aterosclerosi> (visitato il 18/02/2022).
- [3] S. Bianco, R. Cadene, L. Celona e P. Napoletano, “Benchmark Analysis of Representative Deep Neural Network Architectures,” *IEEE Access*, vol. 6, pp. 64 270–64 277, 2018. DOI: 10.1109/ACCESS.2018.2877890.
- [4] J. Brownlee. “Gentle Introduction to the Adam Optimization Algorithm for Deep Learning.” (gen. 2021), indirizzo: <https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/>.
- [5] ——, “How to Choose an Activation Function for Deep Learning.” (gen. 2021), indirizzo: <https://machinelearningmastery.com/choose-an-activation-function-for-deep-learning/>.
- [6] *Carotid Artery Vessel Wall Segmentation Challenge*, Endorsed by SMRA 2021 and MICCAI 2021. indirizzo: <https://vessel-wall-segmentation.grand-challenge.org/> (visitato il 03/02/2022).
- [7] F. Chollet et al. “Keras.” (2015), indirizzo: <https://keras.io>.
- [8] GE Healthcare, *MERGE MRI*. indirizzo: <https://www.gehealthcare.it/products/magnetic-resonance-imaging/mr-applications/merge---spine>.
- [9] Humanitas Research Hospital, *Aterosclerosi*. indirizzo: <https://www.humanitas.it/malattie/aterosclerosi/> (visitato il 03/02/2022).
- [10] W. Kerwin, D. Xu, F. Liu et al., “Magnetic Resonance Imaging of Carotid Atherosclerosis: Plaque Analysis,” *Topics in Magnetic Resonance Imaging*, vol. 18, n. 5, 2007, ISSN: 1536-1004. DOI: 10.1097/rmr.0b013e3181598d9d.
- [11] K. Shen. “Effect of batch size on training dynamics.” (giu. 2018), indirizzo: <https://medium.com/minidistill/effect-of-batch-size-on-training-dynamics-21c14f7a716e>.
- [12] M. Singh, *Multiclass Classification Neural Network using Adam Optimizer*, lug. 2021. indirizzo: <https://towardsdatascience.com/multiclass-classification-neural-network-using-adam-optimizer-fb9a4d2f73f4>.
- [13] F. Thamm, F. Denzinger, L. Rist, C. M. Vicario, F. Kordon e A. Maier, *Segmentation of the Carotid Lumen and Vessel Wall using Deep Learning and Location Priors*, 2022. arXiv: 2201.06259 [eess.IV].
- [14] J. Wu, J. Xin, X. Yang et al., “Deep morphology aided diagnosis network for segmentation of carotid artery vessel wall and diagnosis of carotid atherosclerosis on black-blood vessel wall MRI,” *Medical Physics*, vol. 46, n. 12, pp. 5544–5561, 2019. DOI: <https://doi.org/10.1002/mp.13739>.
- [15] S. S. Yadav e S. M. Jadhav, “Deep convolutional neural network based medical image classification for disease diagnosis,” *Journal of Big Data*, vol. 6, n. 1, p. 113, dic. 2019, ISSN: 2196-1115. DOI: 10.1186/s40537-019-0276-2. indirizzo: <https://doi.org/10.1186/s40537-019-0276-2>.
- [16] P. Yakubovskiy, *Segmentation Models*, [https://github.com/qubvel/segmentation\\_models](https://github.com/qubvel/segmentation_models), 2019. indirizzo: [https://github.com/qubvel/segmentation\\_models/blob/master/segmentation\\_models/models/linknet.py#~:text=input\\_shape%5C%3A%5C%20shape%5C%20of,by%5C%20factor%5C%20%5C%60%5C%6032%5C%60%5C%60..](https://github.com/qubvel/segmentation_models/blob/master/segmentation_models/models/linknet.py#~:text=input_shape%5C%3A%5C%20shape%5C%20of,by%5C%20factor%5C%20%5C%60%5C%6032%5C%60%5C%60..)