



Politecnico di Torino

LAUREA MAGISTRALE IN INGEGNERIA BIOMEDICA

ELABORAZIONE DI IMMAGINE MEDICHE

Vessel Wall Segmentation Challenge

Titolare del corso:
Filippo Molinari

Tutor:
Kristen Mariko Meiburger

Autori:
Benedetta Boccardi - s292175
Martina Benvenuto - s291327
Domenico Ficili - s279931
Antonino Bellina - s283552

ANNO ACCADEMICO 2021/2022

Indice

1	Introduzione	1
1.1	Motivazione Clinica e Obiettivo	1
2	Metodi	1
2.1	Dataset	1
2.1.1	Eliminazione dell'area di biforcazione ed estrazione delle maschere	2
2.1.2	Creazione del dataset finale	3
2.2	Pre Processing	4
2.3	Data Augmentation	4
2.4	Architettura Rete	5
2.5	Training della rete	5
2.6	Post Processing	6
3	Risultati	7
4	Conclusione	8
A	Guida scripts	10
A.1	ReadCASCADEcontours_Final.ipynb	10
A.2	Create_Final_Masks.ipynb	10
A.3	Script_Training.ipynb	10
A.4	SavingMasks_TrainVal.ipynb	10
A.5	SavingMasks_Test.ipynb	10
A.6	Metrics.ipynb	10
B	Guida files	11
B.1	Cartelle	11
B.2	File	11
	Bibliografia	12

1 Introduzione

1.1 Motivazione Clinica e Obiettivo

L'*aterosclerosi*, una delle principali cause di morte in tutto il mondo, è una malattia sistemica che porta alla formazione di placche o al restringimento luminale in più letti vascolari, comprese le arterie carotidi; essa si sviluppa nelle pareti dell'arteria e quindi è importante misurare lo spessore della parete del vaso per differenziare casi normali da patologici. L'imaging della parete vascolare con risonanza magnetica dell'arteria carotide è in grado di identificare lesioni aterosclerotiche che possono causare eventi clinici, ciò nonostante le sequenze MRI di acquisizione assiale tradizionali richiedono una lunga scansione. Per garantire la conformità del paziente e la qualità dell'immagine diagnostica, è stata sviluppata una tipologia di MRI chiamata *3D-MERGE* (3D Motion Sensitized Driven Equilibrium prepared Rapid Gradient Echo), adoperata per diagnosticare patologie legate alla conformazione dei vasi, nel tentativo di superare il gold standard attuale rappresentato dalla tecnica ecografica che ha il limite di essere un'acquisizione 2D. Questa sequenza rapida, che può completare una scansione carotidea in 2 minuti, ha una potenziale applicazione clinica nell'identificazione di pazienti con lesioni avanzate, ma la sua applicazione è limitata a causa della complessità della revisione delle immagini 3D, del gran numero di immagini disponibili e della mancanza di radiologi qualificati con una vasta esperienza nella valutazione dello spessore della parete del vaso carotideo. L'obiettivo della challenge è la segmentazione automatica delle pareti carotidee (interna, esterna e comune), tramite l'elaborazione di immagini acquisite attraverso la tecnica 3D-MERGE usando algoritmi di Deep Learning basati sull'allenamento di reti neurali.

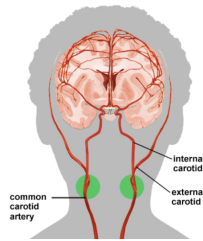


Figura 1: Area anatomica di interesse

2 Metodi

2.1 Dataset

Il dataset fornito è composto da immagini MRI in formato DICOM (ognuna rappresentante una slice del volume) e le segmentazioni manuali (ottenute tramite programma Cascade) in formato QVS, che vanno a comporre un totale di 50 volumi. Tali volumi, che dunque rappresentano ciascuno un paziente, risultano essere così suddivisi:

- *Training Set*: 35 volumi
- *Validation Set*: 5 volumi
- *Test Set*: 10 volumi

Per il Training e Validation Set sono stati estratti a partire dalle segmentazioni manuali i profili di parete ('wall') e lume ('lumen') per ogni tipo di carotide, intese rispettivamente come:

- *ECAL*: carotide esterna sinistra
- *ECAR*: carotide esterna destra
- *ICAL*: carotide interna sinistra
- *ICAR*: carotide interna destra

Le dimensioni delle immagini risultano essere 100x720, tranne in 5 pazienti del Training Set (8,13,31,32,49) che mostrano invece dimensioni pari a 160x640. Per evitare problemi di sovraccarico della memoria

durante la fase di training, dovendo ridurre il Dataset si è deciso di eliminare proprio tali pazienti, poichè un eventuale *resize* delle dimensioni creava problemi nel settaggio successivo delle label sulle maschere a causa dello smoothing.

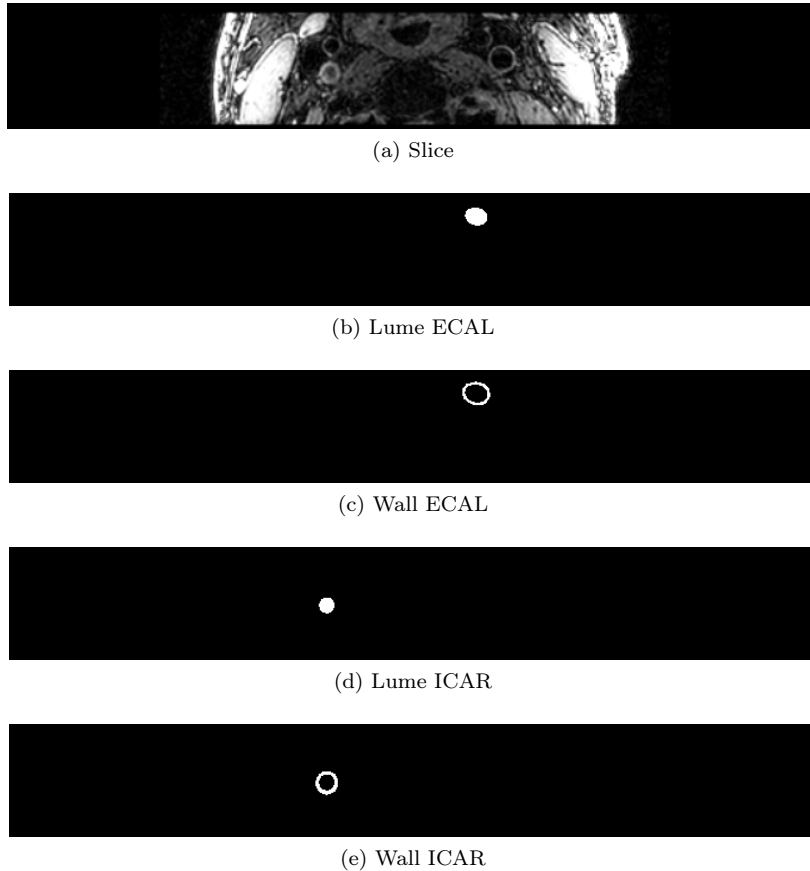


Figura 2: Esempio di slice e maschere estratte

2.1.1 Eliminazione dell'area di biforcazione ed estrazione delle maschere

Come primo passo nella gestione del dataset, per ogni paziente (volume) sono state lette le immagini e i relativi contorni delle maschere disponibili (tramite la funzione fornitaci *GetContours*); sono stati poi calcolati i diametri di tutti i contorni, individuando successivamente il diametro massimo (calcolato come distanza massima tra i pixel del contorno). Avendo la zona di biforcazione area maggiore rispetto alle carotidi esterne e interne, è stato scelto il diametro massimo come criterio per l'individuazione della zona di biforcazione. Si è dunque proceduto all'eliminazione di circa 10 slice in totale: quella con diametro massimo (considerata come slice di riferimento), 3 slice precedenti e circa 6 slice seguenti.

Tale scelta è stata dettata da 2 fattori:

- Il contorno con diametro massimo non sempre coincideva con l'inizio della zona di biforcazione; dunque si è preferito eliminare dal dataset 3 slice prima del riferimento per mitigare il problema.
- Considerando che la zona di biforcazione si estende, in termini di profondità (asse z), per 1 cm (29 slice), sono state eliminate circa 6 slice successive al riferimento (Figura 3).

Terminata la procedura di eliminazione delle slice nella zona di biforcazione, sono state salvate per ogni paziente, l'immagine MRI e le tipologie di maschere disponibili, rispettivamente per wall e lume.

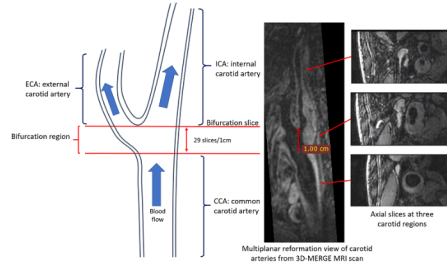


Figura 3: Morfologia carotidi

2.1.2 Creazione del dataset finale

Data la grandezza di ciascuna slice, a partire dal dataset intermedio, le immagini e relative maschere sono state tagliate in due parti, rispettivamente destra e sinistra, in modo da allenare due reti differenti, una per le carotidi di destra e un'altra per le carotidi di sinistra. Questa scelta è stata dettata da due ragioni:

- Osservando il dataset, si è notato che alcune maschere manuali erano presenti solo per un lato (dx o sx) dell'immagine nonostante l'altra parte avesse anch'essa un vaso di cui però non si possedeva la manuale; in questo senso la rete poteva facilmente confondersi e andare incontro ad una cattiva segmentazione.
- Inoltre, aver tagliato l'immagine e di conseguenza averne diminuito le dimensioni beneficia l'efficacia dell'allenamento della rete.

E' stata portata avanti una segmentazione multiclasse, in particolare si è scelto un approccio a tre classi per i motivi precedentemente descritti e pertanto le maschere finali sono state create variando il valore di intensità dei pixel come di seguito:

- Pixel appartenenti allo sfondo: pari a 0
- Pixel appartenenti ai lumen: pari a 1
- Pixel appartenenti ai wall: pari a 2

In questo modo il valore del pixel viene usato come discriminante tra le tre classi. Essendo richiesto che la carotide comune venisse segmentata come ICA, tramite algoritmo sono state cambiate tutte le label delle slice che, a partire dalla numero 400 fino alla fine, avevano maschere manuali classificate come ECA; la scelta conservativa di fissare a 400 la soglia delle slice da cui cambiare la classe è derivata da un'osservazione dell'intero Dataset in cui si è notato che a quell'altezza del volume si era sempre nella zona della carotide comune. Infine, le immagini e relative maschere ottenute sono state salvate in 2 dataset differenti, pronti per essere dati in input alla rete.

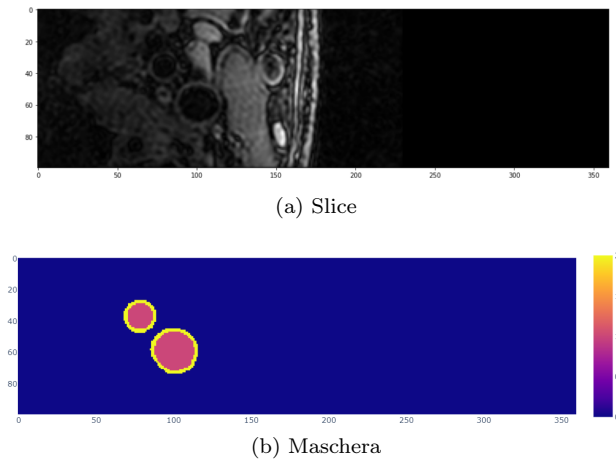


Figura 4: Esempio di slice e maschera finale ottenute e date in input alla rete

2.2 Pre Processing

I passi di preprocessing effettuati sono stati i seguenti:

1. **Crop** dell'immagine per portarla da una dimensione di 100x360 a una di 100x256. In questo modo è stata individuata una ROI eliminando da ogni slice la porzione relativa allo sfondo nero e lo stesso è stato fatto anche con le maschere per riportarle ad una dimensione pari alle immagini.
2. **Zero padding** dell'immagine, passando a una dimensione finale di 128x256.
3. **Percentile contrast stretching** per migliorare il contrasto delle immagini, allungando i valori di intensità di un'immagine per riempire l'intera gamma dinamica, dove l'immagine viene ridimensionata per includere tutte le intensità che rientrano nel 2° e 98° percentile.
4. **Filtro mediano**, con kernel 2x2, per ridurre la rumorosità.

Ciascuna slice (in formato grayscale), è stata rimappata su tre canali in modo da poterle fornire in input alla rete nel formato opportuno. In questo modo si ottengono matrici di dimensioni:

- *Training Set*: Left: (1387x128x256x3) - Right: (1723x128x256x3)
- *Validation Set*: Left: (327x128x256x3) - Right: (269x128x256x3)

Dove le dimensioni rappresentano: (numero di slice utilizzate, altezza, lunghezza, numero di canali). Per quanto riguarda le maschere manuali sono state rimappate in modo da ottenere label categoriche da dare in input alla rete per effettuare la segmentazione multiclasse:

- *Training Set*: Left: (1387x512x512x3) - Right: (1723x128x256x3)
- *Validation Set*: Left: (327x512x512x3) - Right: (269x128x256x3)

dove le dimensioni rappresentano: (numero di maschere utilizzate, altezza, lunghezza, numero di label).

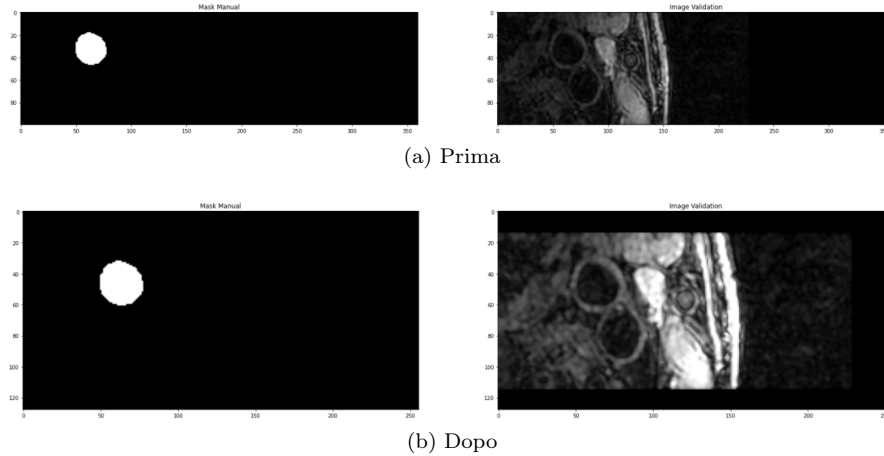


Figura 5: Esempio di slice e maschera prima e dopo il Preprocessing

2.3 Data Augmentation

Per l'allenamento delle rete è stata attuata una leggera Data Augmentation utilizzando un generatore che, durante la fase di allenamento, crea le immagini modificate iterazione per iterazione tramite i seguenti parametri:

- **Rotation Range** = 3: ruota rispetto agli assi di angoli compresi in un range tra 0 e 3°.
- **Width Shift Range** = 0.08: spostamento casuale dell'immagine verso sinistra o verso destra di una quantità pari al valore di una frazione della larghezza dell'immagine.
- **Height Shift Range** = 0.08: spostamento casuale dell'immagine verso l'alto o verso il basso di una quantità pari al valore di una frazione dell'altezza dell'immagine.

- **Zoom Range** = 0.07: ingrandimento dell'immagine.
- **Fill Mode** = *reflect*: riempie i valori vuoti dell'immagine in ordine inverso rispetto ai valori noti.
- **Elastic Deformation**: in immagini di tipo biomedicale è particolarmente significativa, in quanto simula realisticamente la variazione dei tessuti. Così facendo si è cercato di aumentare e diversificare leggermente le immagini fornite in input, rispettando comunque l'anatomia della zona trattata.

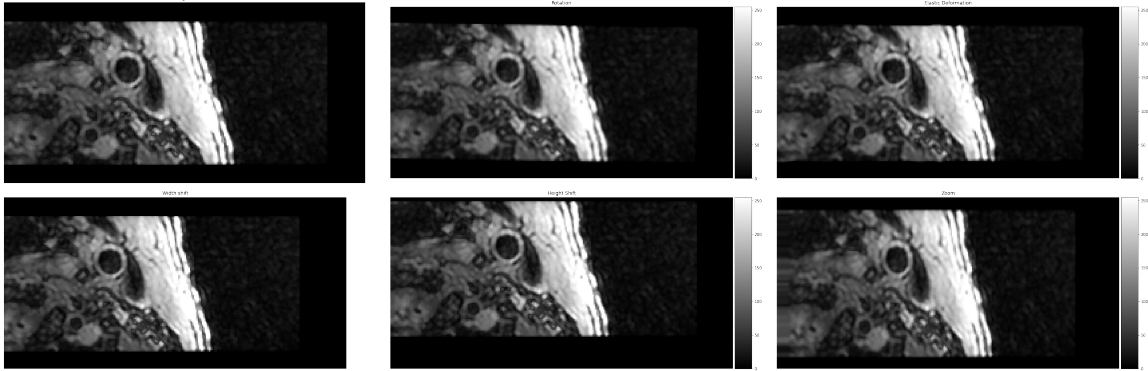


Figura 6: Data Augmentation

2.4 Architettura Rete

Per la segmentazione è stata utilizzata una rete **Unet** con architettura **ResNet34** preallenata sul dataset **Imagenet** con rispettivi pesi preinizializzati.

I parametri utilizzati durante la fase di training sono:

- Batch size: 16
- Numero di epoche: 40
- Funzione di ottimizzazione: **Adam**, con *learning rate* iniziale pari a 0.001. Il valore iniziale viene decrementato di un fattore 0.1 nel caso in cui per almeno 4 epoche il valore di loss non varia.
- Funzione di loss: Si è utilizzato come criterio il Dice e Focal loss, calcolando una Loss totale data dalla somma pesata dei due: $\text{Loss} = \text{Loss}_{\text{dice}} + \theta \cdot \text{Loss}_{\text{focal}}$. Dopo varie prove è stato trovato come valore ottimale $\theta = 0.5$
- Metriche: le metriche monitorate durante la fase di allenamento sono **IOU score** e **F1 score**, in quanto più informative ed appropriate nel caso di segmentazione multiclasse.

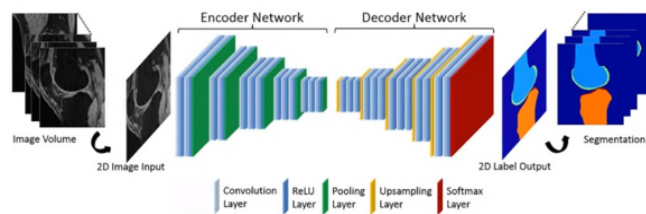


Figura 7: Architettura Unet

2.5 Training della rete

Nella fase di training è stata settata una condizione di *Early Stopping* monitorando la funzione di loss nel tentativo di minimizzarla; l'allenamento termina precocemente se il valore di loss non varia per almeno 7 epoche.

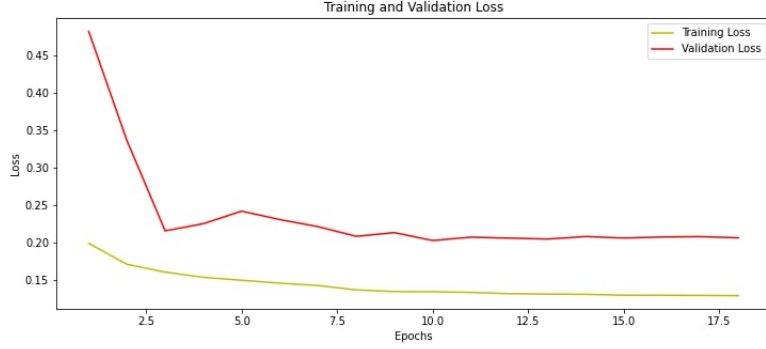


Figura 8: Andamento Loss per epoche

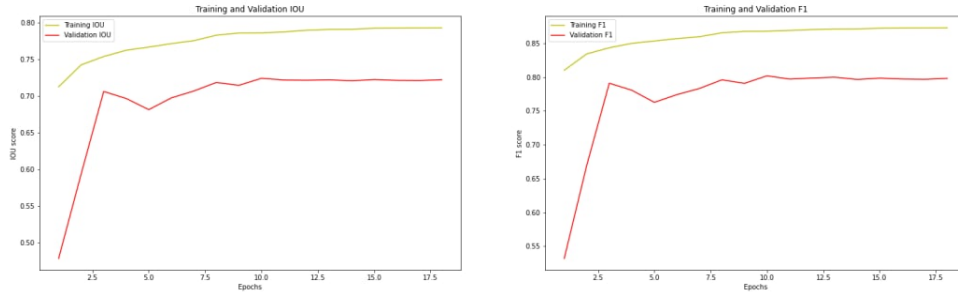


Figura 9: Andamento Metriche per epoche

2.6 Post Processing

Tutte le slice presenti nel Training e Validation Set sono state processate riempiendo i buchi lasciati dalle maschere predette e rimuovendo piccoli oggetti di dimensione minore rispetto ad una soglia settata. Per ricostruire l'informazione relativa a ICA ed ECA, persa durante la creazione delle maschere complete per via dell'allenamento della rete con tre label, si è optato per un metodo basato sui centroidi.

Per ogni paziente, partendo dal basso (ovvero dalle ultime slice) e ciclando sull'intero volume, si è andati a selezionare come riferimento la prima slice in cui la rete segmenta due vasi (corrispondenti ad ICA ed ECA); di questa slice (riferimento) sono stati calcolati i centroidi dei due vasi e assegnati a interna ed esterna $[(cxi, cyi), (cxe, cye)]$.

Per quanto concerne l'assegnazione, dopo la biforcazione, la carotide interna è stata classificata (nelle maschere manuali) come il vaso posizionato nelle maschere automatiche più in basso lungo l'asse delle y, mentre l'esterna come il vaso con posizione più alta rispetto all'asse delle y. Nel caso in cui invece le due segmentazioni avessero la medesima coordinata y del centroide, è stata classificata:

- dataset destro: come interna la segmentazione con valore minore lungo l'asse x e come esterna il restante vaso segmentato
- dataset sinistro: come interna la segmentazione con valore maggiore lungo l'asse x e come esterna il restante vaso segmentato

Per il calcolo dei centroidi di riferimento, si sono considerate esclusivamente le segmentazioni dei lumi, essendo le pareti del vaso concentriche a quest'ultime.

A partire dalla slice in cui è stato trovato il riferimento (tutte le precedenti vengono assegnate a ICAL o ICAR facendo parte della porzione di carotide comune) e proseguendo lungo l'asse trasversale, si sono calcolati i centroidi di ogni slice del paziente e si è ricavata la distanza euclidea tra i centroidi della slice i-esima corrispondente e i centroidi della slice di riferimento: le maschere vengono assegnate ad ICA o ECA scegliendo la classificazione che minimizza la distanza euclidea.

Nel caso in cui in un paziente non venisse trovata nessuna slice con entrambe ECA ed ICA (quindi il riferimento), si è deciso di assegnare tutte le segmentazioni predette per quel paziente come ICA.

Nella fase di assegnazione delle maschere alle opportune classi, sono state gestite numerose eccezioni derivanti dal diverso numero di contorni e/o morfologia delle maschere predette dal modello.

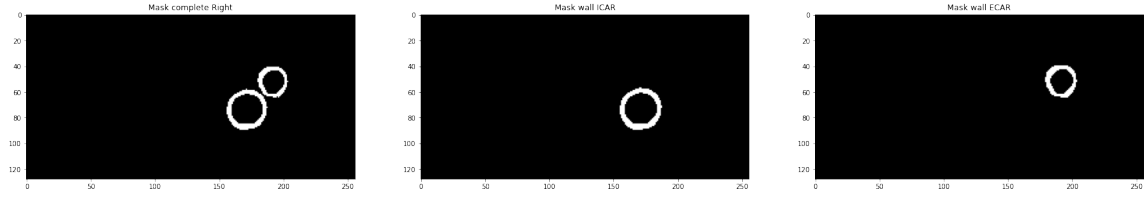


Figura 10: Divisione da maschera completa in ICA ed ECA nel Validation Set: paziente 12 - slice 354

3 Risultati

Per valutare le prestazioni del modello allenato, esso è stato applicato a Train e Validation Set per ottenere le maschere predette e tramite un confronto con le maschere manuali, sono state ricavate le seguenti metriche richieste per ogni slice del Training e Validation Set:

- Dice Similarity Coefficient (DSC): $\frac{2|X \cap Y|}{|X| + |Y|}$
- Relative Volume Difference (RVD): $\frac{|Y| - |X|}{|X|}$

dove X indica la segmentazione manuale e Y la segmentazione predetta ottenuta dal modello. Il DSC è calcolato solamente per le maschere relative alla parete del vaso, l'RVD sia per le maschere del lume che per l'unione delle maschere di parete e lume.

Le due metriche sono state calcolate sia per ogni slice del Training e Validation Set, che come valori espressi in termini di *Media* \pm *DeviazioneStandard* dell'intero Set:

	DSC	RVD
Training Set	0.721 ± 0.044	0.037 ± 0.027
Validation Set	0.696 ± 0.034	0.016 ± 0.043

Tabella 1: Metriche finali di Training e Validation Set.

Di seguito sono riportati le informazioni riassuntive relativamente al Validation Set e un esempio di confronto tra segmentazione manuale e predetta di un paziente del Validation-Set (Figura 13)

Paziente	DSC	RVD	Auto. non trovate	Man. non trovate	Totali	% Trovate
4	0.694	0.003	90	71	221	45.80
12	0.746	-0.023	337	306	846	39.88
19	0.692	-0.006	58	56	254	71.43
27	0.698	0.017	40	18	166	85.71
36	0.650	0.088	66	91	347	67.62

Tabella 2: Metriche riassuntive relative al Validation Set.

Le metriche per paziente del Validation-Set evidenziano un totale di maschere automatiche non corrisposte dalle maschere manuali piuttosto elevato: questo problema è dovuto ad una mancanza di segmentazioni delle carotidi esterne nel dataset di partenza anche se presenti nella rispettiva immagine, che tuttavia vengono comunque segmentate dalla rete. L'utilizzo dunque di un dataset di partenza senza errori nell'assegnazione della classe delle maschere manuali, soprattutto nella zona dove sono entrambi i vasi presenti, avrebbe portato ad una predizione più accurata delle maschere automatiche, essendo l'area maggiormente critica per l'allenamento del modello.

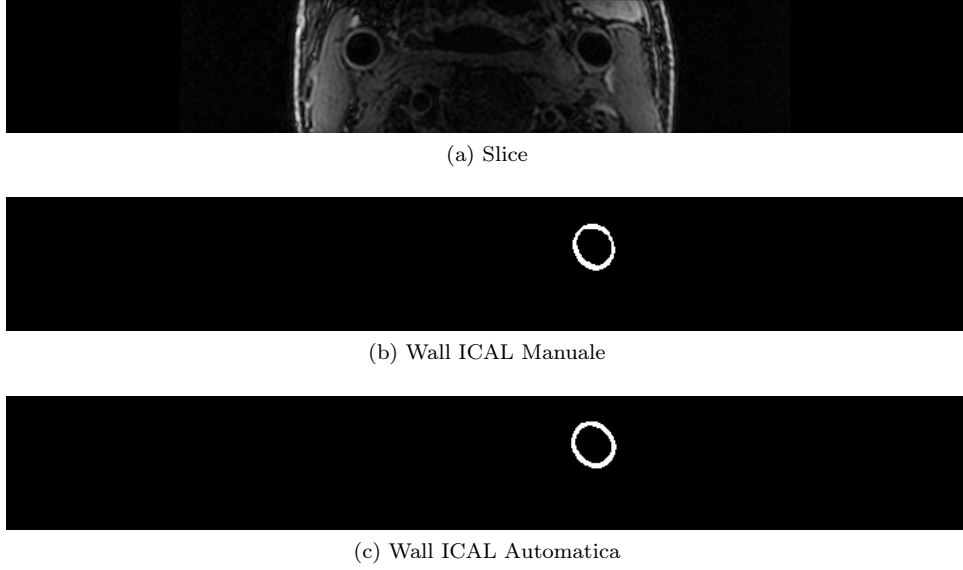


Figura 11: Confronto tra segmentazione manuale e predetta del paziente 36 - slice 403, del Validation Set

4 Conclusione

Prima di arrivare al modello finale, sono state effettuate varie prove per valutare come ciascun elemento della pipeline influisse sull'allenamento della rete per trovare il giusto compromesso tra efficacia e ottimizzazione computazionale.

Sono stati testati:

- varie tipologie di pre-processing tra cui N4 Bias Correction, rescaling, equalizzazione e filtraggio sia singolarmente che in combinazione
- modifiche nei parametri dell'architettura quali learning Rate, metriche e loss
- Unet con diversi Backbone (Figura 12)

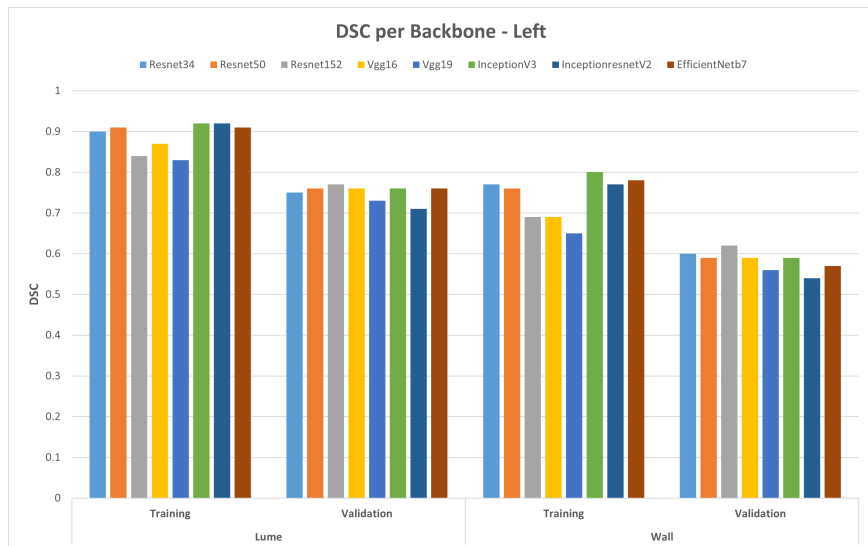
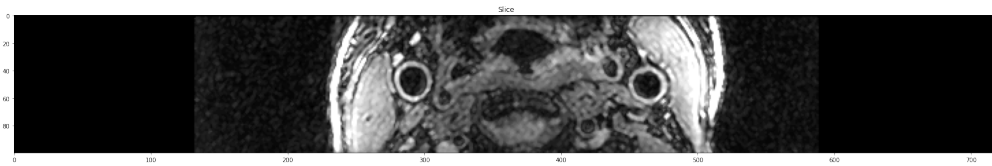


Figura 12: Performance Unet con diversi Backbone sul nostro dataset

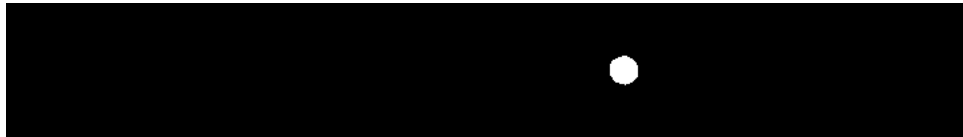
Il modello maggiormente performante, ottenuto con i parametri descritti in precedenza (in ogni sezione apposita), è stato applicato alle immagini del Test Set (a cui dapprima sono state applicate le stesse procedure di pre-processing applicate a Training e Validation Set) ottenendo le segmentazioni automatiche predette.

I risultati delle metriche di DSC e RVD ottenute sul Training e Validation Set, evidenziano prestazioni non molto soddisfacenti soprattutto per alcuni pazienti in cui vi è un elevato numero di maschere automatiche non segmentate di cui si possedeva invece la segmentazione manuale. Ciò è indice del fatto che il criterio dei centroidi, adottato per la divisione delle maschere predette dalla rete nelle rispettive ECA ed ICA presenta dei punti deboli in alcune parti. Lo stesso può essere notato per le segmentazioni predette nel caso del Test Set, dove si riscontrano problemi nell'individuazione della corretta slice di riferimento. Esperimenti futuri e possibili miglioramenti potrebbero ottenersi:

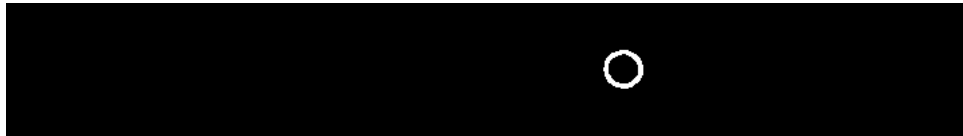
- nella ricostruzione dell'informazione relativa a ICA ed ECA, cercando di ottenere un algoritmo più robusto e fedele alla realtà anatomica delle carotidi.
- utilizzando una registrazione ad hoc per le immagini MRI con lo scopo di risolvere problemi di disallineamento [1].



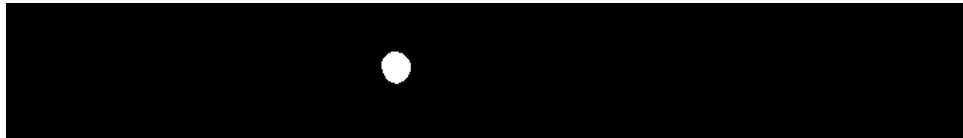
(a) Slice



(b) Lume ICAL Automatica



(c) Wall ICAL Automatica



(d) Lume ICAR Automatica



(e) Wall ICAR Automatica

Figura 13: Segmentazioni predette Test Set: paziente 35 - slice 404

A Guida scripts

L'intera pipeline richiede di eseguire gli script forniti nel seguente ordine.

A.1 ReadCASCADEcontours_Final.ipynb

1. Legge le immagini in formato DICOM e i contorni delle maschere in formato QVS
2. Seleziona all'interno di ogni volume solamente le slice di cui sono disponibili i contorni delle maschere
3. Calcola i diametri di tutte le slice presenti nel volume
4. Individua l'area di biforcazione attraverso il criterio dei diametri ed elimina le slice appartenenti a tale zona
5. Costruisce la maschera binaria di *lume* e *wall* a partire dai contorni
6. Salva in un Dataset intermedio le slice e le maschere

A.2 Create_Final_Masks.ipynb

1. A partire dal Dataset intermedio, legge le immagini e maschere di interesse e assegna ai pixel delle maschere i valori delle label
2. Crea due Dataset finali, rispettivamente per parte destra e sinistra, con le immagini e maschere salvate in opportune cartelle e pronte per essere caricate sul Drive

A.3 Script_Training.ipynb

1. Carica i file .zip dei 2 dataset finali che contengono le immagini e maschere da dare in input alla rete
2. Applica processi di Preprocessing e Data Augmentation
3. Effettua l'allenamento della rete
4. Salva il modello e tutte le informazioni ad esso legate

A.4 SavingMasks_TrainVal.ipynb

1. Applica azioni di Post processing quali operatori morfologici, rimozione di oggetti e riempimenti di aree bucate per sistemare alcune segmentazioni mal predette
2. Ricostruisce l'informazione di ICA ed ECA per poter dividere le maschere
3. Salva tutte le maschere ricostruite nelle opportune cartelle, pronte per calcolarne le metriche

A.5 SavingMasks_Test.ipynb

Effettua le stesse operazioni dello script precedente ma per il Test Set. Si è scelto di dividere i due scripts per un maggiore livello di comprensione e ordine, data la loro lunghezza.

A.6 Metrics.ipynb

1. Salva in memoria, in un dizionario, le matrici di slice e maschere manuali insieme all'informazione della slice e paziente
2. Itera nella cartella RESULTS per leggere e confrontare man mano le maschere predette con le manuali allocate in memoria
3. Calcola sia le metriche richieste che delle informazioni riassuntive per ogni paziente di Training e Validation Set, e successivamente salva tutto in file JSON.

B Guida files

Cartelle e files ricavati e forniti, eseguendo la pipeline degli scripts

B.1 Cartelle

1. DATASET: dataset originale estratto dal file zip fornito dalla challenge
2. DATASET_Intermedio: dataset intermedio creato dallo script A.1
3. DATASET_Final_Left: dataset finale con le maschere della parte sinistra creato dallo script A.2
4. DATASET_Finale_Right: dataset finale con le maschere della parte destra creato dallo script A.2

B.2 File

1. Metrics_train.json: file contenente le metriche per ogni singola slice del Training Set
2. Metrics_val.json: file contenente le metriche per ogni singola slice del Validation Set
3. Info_train.json: file riassuntivo contenente le metriche e alcune informazioni utili sulla bontà della predizione, per ogni paziente del Training Set; sono anche espresse in termini di media e deviazione standard le due metriche richieste (DSC, RVD) relative all'intero Set.
4. Info_val.json: file contenenti le stesse informazioni elencate nel punto precedente per il Validation Set.

Bibliografia

- [1] Samira Masoudi et al. “Quick guide on radiology image pre-processing for deep learning applications in prostate cancer research”. In: *Journal of Medical Imaging* 8.1 (2021), pp. 1–14. DOI: 10.1117/1.JMI.8.1.010901. URL: <https://doi.org/10.1117/1.JMI.8.1.010901>.
- [2] Tianshu Zhou et al. “Fully automatic deep learning trained on limited data for carotid artery segmentation from large image volumes”. In: *Quantitative Imaging in Medicine and Surgery* 11.1 (2020). ISSN: 2223-4306. URL: <https://qims.amegroups.com/article/view/50255>.
- [3] Dieuwertje Alblas, Christoph Brune e Jelmer M. Wolterink. *Deep Learning-Based Carotid Artery Vessel Wall Segmentation in Black-Blood MRI Using Anatomical Priors*. 2021. arXiv: 2112.01137 [eess.IV].
- [4] *Pre-trained Models for Image Classification*. <https://www.analyticsvidhya.com/blog/2020/08/top-4-pre-trained-models-for-image-classification-with-python-code/>.
- [5] *Learning Rate Schedules*. <https://towardsdatascience.com/learning-rate-schedules-and-adaptive-learning-rate-methods-for-deep-learning-2c8f433990d1>.