

# POLITECNICO DI TORINO

Corso di Laurea Magistrale in INGEGNERIA BIOMEDICA



Tesi di Laurea Magistrale

## SVILUPPO DI UN METODO AUTOMATICO BASATO SUL DEEP LEARNING PER LA VALUTAZIONE AUTOMATICA DEL NERVO OTTICO IN IMMAGINI ULTRASONOGRAFICHE TRANSORBITALI

Relatore

Prof. KRISTEN MARIKO MEIBURGER

Correlatori

Prof. FILIPPO MOLINARI

Ing. FRANCESCO MARZOLA

Candidata

**DENISE GENTILE**

Dicembre 2021



# Abstract

L'ultrasonografia transorbitale è una tecnica di imaging molto promettente nella caratterizzazione delle strutture del nervo ottico (ON) e le sue eventuali alterazioni, dovute a un aumento della pressione intracranica (ICP) o alla presenza di altre patologie, ad esempio Sclerosi Multipla e idrocefalo. È un metodo di indagine diagnostica non invasiva, rapida, ad alta accessibilità, adatta a contesti ospedalieri e ambulatoriali, e sicura per i pazienti. La sua applicazione più diffusa prevede la misurazione del diametro della guaina del nervo ottico (ONSD), direttamente dalle immagini ecografiche acquisite, con lo scopo di stimare un eventuale aumento dell'ICP. Numerosi studi in letteratura confermano una relazione tra l'innalzamento dell'ICP e l'aumento dell'ONSD, offrendo un'alternativa di indagine diagnostica non invasiva e sicura. Attualmente il "gold standard" per la valutazione dell'ICP prevede l'inserimento di una sonda intracranica, tecnica invasiva che potrebbe causare complicazioni (infezioni, emorragie). Non solo l'ONSD, ma anche il diametro del nervo ottico (OND) può essere d'interesse: studi in merito dimostrano il legame tra lo spessore dell'ON e il livello di atrofia del nervo, in pazienti affetti da Sclerosi Multipla. In particolare, nei soggetti patologici si ha una diminuzione dell'OND rispetto ai soggetti sani.

Il presente lavoro di tesi vede al suo centro lo sviluppo di un sistema completamente automatico in grado di segmentare le strutture del nervo ottico in immagini ecografiche transorbitali e di calcolarne i diametri: OND e ONSD.

Il processo di segmentazione si basa su modelli pre-addestrati di reti neurali completamente convoluzionali (FCN). Nello specifico è stato allenato un modello U-net, con encoder ResNet50; utilizzando nella fase di training le maschere binarie ottenute da una precedente segmentazione manuale. A partire dalle segmentazioni delle strutture del nervo ottico, ottenute come output della rete, si ricavano i valori di OND e ONSD in modo completamente automatico. Viene, innanzitutto, costruita la centerline dell'ON e individuato il punto che dista 3 mm dal bulbo oculare. Da qui vengono calcolate le distanze tra i bordi del nervo ottico e tra quelli della sua guaina, ricavati dalla precedente segmentazione. Tali distanze corrisponderanno rispettivamente all'OND e all'ONSD. Il sistema, così sviluppato, è stato applicato su un data set di 464 immagini eterogenee, acquisite da 110

soggetti diversi e utilizzando 4 macchinari differenti.

Le misure automatiche ottenute sono state confrontate con quelle di un operatore manuale. L'errore medio comparato con l'operatore è pari a  $-0.12 \pm 0.32$  mm e  $0.14 \pm 0.58$  mm, rispettivamente per l'OND e L'ONSD. Si individua una correlazione positiva tra le due misurazioni: il coefficiente di correlazione di Pearson ha un valore di 0.71 per l'OND e di 0.64 per l'ONSD.

Lo scopo finale è quello di ridurre la dipendenza che il calcolo dell'OND e dell'ONSD ha dall'operatore manuale, mitigando l'errore sistematico nella misurazione manuale e abbattendo la variabilità tra le misure, in modo da favorirne l'applicazione diagnostica in campo medico. L'algoritmo sviluppato risulta essere completamente automatico, e l'errore medio ottenuto per la misura dell'ONSD risulta essere coerente con i range della variabilità inter-operatore, individuati in letteratura.



# Indice

<b>Elenco delle tabelle</b>	VIII
<b>Elenco delle figure</b>	IX
<b>Acronimi</b>	XIII
<b>1 Il nervo ottico</b>	1
1.1 Cos'è il nervo ottico . . . . .	1
1.2 Patologie e alterazioni del nervo ottico . . . . .	3
1.2.1 Relazione tra OND, ONSD e pressione intracranica . . . . .	4
1.3 Imaging del nervo ottico . . . . .	5
<b>2 Ultrasonografia</b>	7
2.1 Cos'è l'ultrasonografia . . . . .	7
2.1.1 Formazione dell'immagine . . . . .	8
2.1.2 Tipologie di sonde . . . . .	9
2.1.3 Strategie di visualizzazione . . . . .	10
2.2 Ultrasonografia Transorbitale . . . . .	11
2.3 Calcolo di OND e ONSD da un'immagine di ultrasonografia transorbitale . . . . .	13
<b>3 Proposte di sistemi automatici per il calcolo dell'OND e dell'ONSD</b>	15
3.1 Primo modello per il calcolo automatico del diametro della guaina del nervo ottico (ONSD) . . . . .	16
3.2 AUTONoMA - AUTomatic Optic Nerve MeAsurement . . . . .	18
3.3 Algoritmo per la misurazione automatica dell'ONSD basato sul Clustering . . . . .	21
<b>4 Deep Learning per l'<i>image segmentation</i></b>	23
4.1 Reti Neurali Convolutionali (CNN) . . . . .	25

4.1.1	Evoluzione dei modelli CNN . . . . .	29
4.1.2	ResNet . . . . .	31
4.1.3	EfficientNet . . . . .	33
4.2	Segmentazione Semantica . . . . .	36
4.2.1	U-Net . . . . .	37
4.2.2	UNet++ . . . . .	38
4.2.3	MA-Net . . . . .	41
4.3	Overfitting . . . . .	42
<b>5</b>	<b>Segmentazione delle strutture del nervo ottico e calcolo di OND e ONSD</b>	<b>45</b>
5.1	Dataset . . . . .	45
5.2	Architettura dell'algoritmo . . . . .	46
5.3	Cropping dell'immagine . . . . .	47
5.4	Segmentazione manuale delle strutture del nervo ottico . . . . .	52
5.5	Segmentazione automatica delle strutture del nervo ottico . . . . .	53
5.5.1	Allenamento delle reti . . . . .	54
5.5.2	K-Fold Cross-Validation sul modello più performante . . . . .	57
5.6	Post-Processing sull'output della rete . . . . .	59
5.7	Calcolo automatico di OND e ONSD . . . . .	62
5.8	Metriche di validazione . . . . .	65
5.8.1	Metriche di validazione per la segmentazione . . . . .	65
5.8.2	Metriche di validazione per i diametri . . . . .	68
<b>6</b>	<b>Risultati</b>	<b>71</b>
6.1	Validazione delle segmentazioni . . . . .	71
6.1.1	Risultati degli allenamenti di U-Net, UNet++ e MA-Net . . . . .	71
6.1.2	Risultati dell'allenamento con cross-validation . . . . .	72
6.2	Validazione dei valori di OND e ONSD . . . . .	73
6.3	Validazione dei risultati per ogni macchinario . . . . .	74
6.4	Confronto dell'algoritmo proposto con AUTONoMA . . . . .	77
6.5	Discussione dei risultati . . . . .	77
6.5.1	Algoritmo proposto . . . . .	77
6.5.2	Confronto con AUTONoMA . . . . .	78
<b>7</b>	<b>Conclusioni e sviluppi futuri</b>	<b>81</b>
	<b>Bibliografia</b>	<b>83</b>

# Elenco delle tabelle

4.1	Architettura ResNet50. . . . .	33
4.2	Architettura EfficientNet-B0 [51]. . . . .	35
5.1	Percentuali immagini acquisite per ogni macchinario. . . . .	46
5.2	Modelli allenati. . . . .	54
5.3	Parametri di input per l'allenamento delle reti. . . . .	55
5.4	Suddivisione del dataset in 5 fold per eseguire Cross-Validation. . . . .	57
6.1	Metriche di validazione pixel-based. Si riportano IoU, Dice coefficient, BFscore, Precision e Recall. Si indicano i valori medi e le rispettive deviazioni standard. . . . .	72
6.2	Valori medi e rispettive deviazioni standard del coefficiente Dice calcolato per ogni Fold di Test. . . . .	72
6.3	Metriche di validazione pixel-based. Si riportano IoU, Dice coefficient, BFscore, Precision e Recall. Si indicano i valori medi e le rispettive deviazioni standard. . . . .	73
6.4	Risultati dell'algoritmo per l'OND e l'ONSD confrontati con le misurazioni manuali. Sono riportati i valori medi e le loro deviazioni standard. . . . .	73
6.5	Coefficiente di correlazione di Person valutato per OND e ONSD. . . . .	74
6.6	Intraclass Correlation Coefficient per OND e ONSD. . . . .	74
6.7	Coefficiente Dice valutato per ogni macchinario. Si riportano i valori medi e le corrispondenti deviazioni standard. . . . .	75
6.8	Valori medi e deviazioni standard di OND e ONSD per ognuno dei macchinari utilizzati. I valori sono espressi in mm. . . . .	75
6.9	Errori delle misurazioni di OND e ONSD suddivisi per macchinario. Dove MAE indica l'errore medio assoluto e FoM il Figure of Merit. . . . .	76
6.10	Valori dei coefficienti di correlazione di Pearson e di intraclass, riportati per ogni macchinario sia per OND che per ONSD. . . . .	76
6.11	Confronto tra l'algoritmo proposto e AUTONoMA. . . . .	77

# Elenco delle figure

1.1	Anatomia dell'occhio e del nervo ottico [2]. . . . .	1
1.2	Anatomia del nervo ottico - Meningi [3]. . . . .	2
2.1	Esempi di immagini ecografiche. Da sinistra verso destra: ecografia di un feto [17], ecografia tiroidea [18], ecografia transorbitale di un nervo ottico (da dataset). . . . .	7
2.2	Sonda Lineare [21]. . . . .	10
2.3	Sonda Convessa [21]. . . . .	10
2.4	Sonda Phased Array [21]. . . . .	10
2.5	Strategie di scansione. Da sinistra a destra: A-Mode, B-Mode, C-Mode [3]. . . . .	11
2.6	Ecografie Transorbitali acquisite con frequenze diverse. (A 3.5 MHz, B 5 MHz, C 6.6 MHz, D 10 MHz) [22]. . . . .	12
2.7	Da sinistra: esempio postazione per l'acquisizione dell'ecografia transorbitale, esecuzione di un'ecografia transorbitale [23][24]. . . . .	12
2.8	Immagine di ecografia transorbitale (da dataset). . . . .	13
2.9	Immagine di ecografia transorbitale in cui sono stati evidenziati OND (2-2) e ONSD (3-3) [22]. . . . .	14
3.1	Step intermedi della localizzazione del bulbo oculare. (A) Immagine di Input, (B) FPB Gaussiano e maschera binaria, (C) Distance Transform, (D) Ellisse corrispondente ai bordi del bulbo oculare [27].	16
3.2	Step intermedi per l'individuazione della guaina del nervo. (A) ROI, (B) FPB Gaussiano, (C) Distance Transform, (D) Barre verticali corrispondenti alla guaina [27]. . . . .	17
3.3	Immagine di Output [27]. . . . .	18
3.4	Flow-chart dell'algoritmo [29]. . . . .	18
3.5	Profilo del Bulbo [29]. . . . .	19
3.6	Centerline del Nervo [29]. . . . .	19
3.7	Segmentazione finale dell'ON e dell'ONS [29]. . . . .	20

4.1	Struttura base di una rete neurale [33]. . . . .	23
4.2	Esempio struttura degli strati di una Rete Neurale Convolutionale [36]. . . . .	25
4.3	Architettura generica di una CNN [37]. . . . .	25
4.4	Esempio di un'operazione di convoluzione. Da sinistra a destra: Input $I$ di dimensione $7 \times 7$ (evidenziato in rosso il campo ricettivo), filtro $K$ di dimensione $3 \times 3$ , mappa di attivazione di output di dimensione $5 \times 5$ . [39]. . . . .	27
4.5	Esempio di <i>max pooling</i> [36]. . . . .	27
4.6	Grafici delle funzioni di attivazione: (a) Sigmoidale, (b) Tangente iperbolica, (c) ReLU, (d) Leaky ReLU [40]. . . . .	28
4.7	Evoluzione temporale delle performance delle architetture CNN sul dataset ImageNet [42]. . . . .	29
4.8	Blocco residuale [47]. . . . .	31
4.9	Confronto delle mappe di attivazione estratte dalla VGG-19 e dalla ResNet50. Da sinistra a destra: immagine originale, feature map della VGG-19, feature map della ResNet50. [50]. . . . .	32
4.10	Ridimensionamento del modello. (a) rete di base generica, (b)-(d) ridimensionamento di una sola delle grandezze, (e) ridimensionamento composto [51]. . . . .	34
4.11	Blocco residuale invertito a collo di bottiglia [54]. . . . .	36
4.12	Generica struttura di una rete per segmentazione semantica [57]. . . . .	37
4.13	Architettura U-Net [58]. . . . .	38
4.14	Architettura UNet++ [59]. . . . .	39
4.15	<i>Skip pathway</i> che collega i nodi $X^{0,0}$ e $X^{0,4}$ [59]. . . . .	40
4.16	Deep supervision [59]. . . . .	41
4.17	Architettura MA-Net [60]. . . . .	42
5.1	Schema generale dell'algoritmo proposto. . . . .	47
5.2	Esempio immagine originale di input (da dataset - Mac_2_Sub_39_Im_3). 48	
5.3	Confronto di due immagini di input aventi valori di Fattore di Calibrazione differenti. Da sinistra verso destra: $CF = 0.11$ , $CF = 0.05$ . . . . .	49
5.4	Operazione di cropping. (a) Immagine originale. (b) Risultato del primo taglio. (c) Ricerca delle circonferenze basate sulla CHT. (d) Individuazione della circonferenza che meglio approssima il bulbo oculare, in rosso il punto di partenza per la costruzione della patch. (e) Output finale del cropping, immagine $256 \times 256$ . . . . .	51
5.5	Visualizzazione immagine tramite MATLAB R2021a. In verde i punti selezionati manualmente dall'utente. . . . .	52

5.6	(a) Immagine ecografica TOS originale. (b) Immagine segmentata manualmente. (c) Maschera binaria della guaina del nervo, ottenuta dai bordi dalla segmentazione manuale. . . . .	53
5.7	suddivisione effettuata per Cross-validation con k=5. In arancione i fold di <i>Test</i> , in giallo i fold di <i>Validation</i> e in verde quelli di <i>Training</i> .	58
5.8	Da sinistra verso destra: Immagine originale, Maschera binaria predetta dalla rete, Sovrapposizione dell'output sull'immagine originale.	59
5.9	Correzione di bordi non conformi all'anatomia del nervo ottico. Step1. Individuazioni bordi sinistri (rosso) e bordi destri (azzurro); Step2. Individuazione bordo da correggere ed estremo superiore della porzione di bordo da eliminare; Step3. Individuazione estremo inferiore della porzione di bordo da eliminare; Step4. Costruzione di una curva parabolica (magenta) che corregga il bordo originale; Step5. Correzione del bordo (verde). . . . .	61
5.10	Risultato del post-processing. Da sinistra verso destra: Output della rete, Output della rete sovrapposta all'immagine originale e alla segmentazione manuale, Risultato del post-processing, Risultato del post-processing sovrapposto all'immagine originale e alla segmentazione manuale. In verde l'intersezione della segmentazione automatica e manuale (TP), in rosso la segmentazione manuale (FN), in blu la segmentazione automatica (FP). . . . .	62
5.11	Step necessari per il calcolo di OND e ONSD. (a) Nervo ottico ricostruito. (b) Maschera binaria di input e primo profilo della centerline del nervo. (c) Fondo del bulbo oculare. (d) Maschera binaria sovrapposta all'immagine originale, si evidenziano la retta che approssima la centerline (giallo) e le rette perpendicolari, in rosso gli estremi dell'OND e in verde quelli dell'ONSD. . . . .	64
5.12	(a) Tentativo di calcolo dei diametri a 3 mm, nervo non identificato. (b) Tentativo di calcolo a 6 mm, nervo identificato. . . . .	65



# Acronimi

**CF** Calibration Factor

**CHT** Circular Hough Transform

**CNN** Convolutional Neural Network

**DL** Deep Learning

**EDSS** Expanded Disability Status Scale

**FCN** Fully Convolutional Network

**FoM** Figure of Merit

**FPB** Filtro Passa Basso

**GT** Ground Truth

**ICC** Intraclass Correlation Coefficient

**ICP** Intracranial Pressure

**ILSVRC** ImageNet Large Scale Visual Recognition Challenge

**IoU** Intersection over Union

**ML** Machine Learning

**MRI** Magnetic Resonance Imaging

**ON** Optic Nerve

**OND** Optic Nerve Diameter

**ONS** Optic Nerve Sheath

**ONSD** Optic Nerve Sheath Diameter

**ReLU** Rectified Linear Units

**ROI** Region of Interest

**SNC** Sistema Nervoso Centrale

**TC** Tomografia Computerizzata

**TOS** Transorbital Sonography

**US** Ultrasuoni

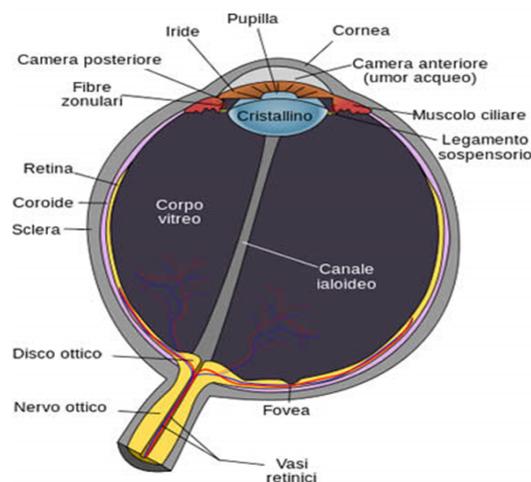
**VGG** Visual Geometry Group

# Capitolo 1

## Il nervo ottico

### 1.1 Cos'è il nervo ottico

Il nervo ottico rappresenta il secondo delle 12 paia di nervi cranici, anche se tecnicamente viene ritenuto una continuazione del Sistema Nervoso Centrale (SNC). Infatti, a differenza degli altri nervi cranici rivestiti da neurilemma<sup>1</sup>, le fibre del nervo ottico sono avvolte dalla guaina mielinica prodotta dagli oligodendrociti (cellule che rivestono i neuroni nel SNC) [1].



**Figura 1.1:** Anatomia dell'occhio e del nervo ottico [2].

---

<sup>1</sup>Il neurilemma è una sottile guaina che ricopre le fibre nervose ed è costituita dalle cellule di Schwann. Cellule che rivestono gli assoni dei neuroni del Sistema Nervoso Periferico.

Fuoriuscendo dall'occhio a livello del **disco ottico** (figura 1.1), noto anche come papilla, il nervo ottico costituisce l'inizio delle vie ottiche; ovvero quelle strutture che collegano il bulbo oculare al cervello, partendo dalla retina. Infatti, il suo compito è proprio quello di trasmettere al cervello gli impulsi elettrici generati in corrispondenza della retina, permettendo di conseguenza la percezione visiva. Il nervo ottico si estende approssimativamente per una lunghezza di 50 mm e può essere suddiviso in 4 segmenti:

- Intraoculare (1mm): porzione che inizia in corrispondenza del disco ottico per poi fuoriuscire dall'occhio.
- Intraorbitario (25 mm): tratto più lungo che prosegue dal polo posteriore dell'occhio fino al canale ottico<sup>2</sup>.
- Intracanalicolare (9 mm): segmento compreso nel canale ottico.
- Perichiasmatico (16 mm): porzione intracranica che si estende fino al chiasma ottico<sup>3</sup> [1].

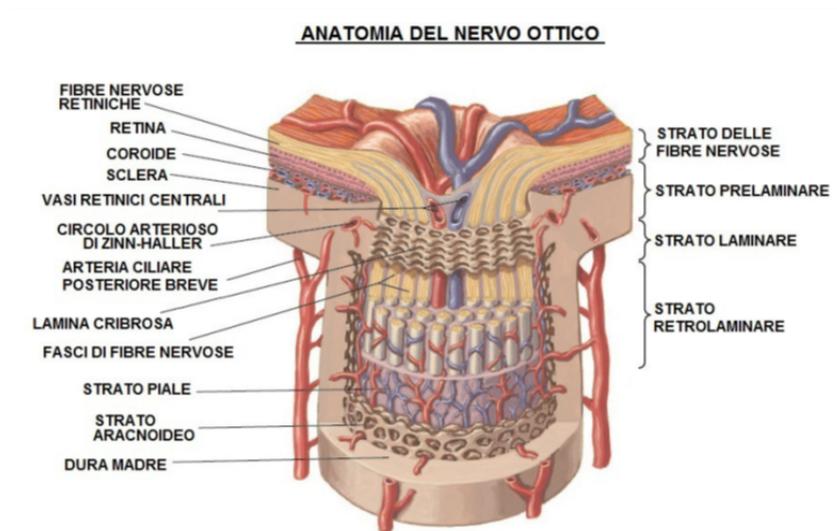


Figura 1.2: Anatomia del nervo ottico - Meningi [3].

<sup>2</sup>Il canale ottico è un foro attraverso il quale passa il nervo ottico, situato nell'osso sfenoide. Osso che partecipa alla formazione della base cranica e contribuisce alla formazione delle cavità orbitarie e della cavità nasale.

<sup>3</sup>Il chiasma ottico è la zona inferiore del cervello, sotto l'ipotalamo, in cui si verifica un parziale incrocio tra le fibre nervose costituenti i nervi ottici.

Osservando la figura 1.2 è possibile notare come il nervo ottico sia rivestito dalle **meningi**. Tre membrane laminari concentriche, che hanno scopo protettivo:

- **Dura Madre:** meninge più esterna costituita da un doppio strato connettivo.
- **Aracnoide:** membrana intermedia a contatto diretto con la dura madre, e salda all'ultimo strato grazie ai suoi filamenti trabecolari.
- **Pia Madre:** membrana più interna e più sottile, costituita da tessuto connettivo.

La membrana aracnoide e la pia madre, come accennato, sono fissate tra loro solo tramite le strutture trabecolari dello strato aracnoide. Quindi non è presente contatto diretto e tra loro si crea ciò che viene chiamato **spazio subaracnoideo**. Questo contiene il liquido cerebrospinale, la cui funzione è quella di aumentare l'azione protettiva esercitata dalle meningi [4]. Inoltre, queste due membrane (pia madre e aracnoide), comprensive naturalmente di spazio subaracnoideo, costituiscono ciò che viene indicato come **guaina del nervo ottico**. Questa rivestirà un ruolo importante nei capitoli successivi e per il lavoro illustrato di seguito.

Le fibre nervose del nervo ottico, a causa del fatto che non sono rivestite da neurilemma ma da mielina, hanno una bassissima capacità di rigenerazione. Di conseguenza, è facile intuire che eventuali danni subiti dal nervo ottico saranno irreversibili e potranno condurre alla cecità del soggetto.

## 1.2 Patologie e alterazioni del nervo ottico

Le patologie che affliggono il nervo ottico possono essere numerose, così come le cause: infiammatorie, ischemiche, tossiche, autoimmuni, infettive, neoplastiche, degenerative, demielinizzanti (Sclerosi Multipla), congenite o, ancora, traumatiche. Tra queste, le forme infiammatorie e degenerative sono tra quelle più pericolose, in termini di compromissione della capacità visiva.

Le prime, meglio note come **neuriti ottiche**, possono essere provocate da un'iperattività del sistema immunitario, dalla presenza di un batterio o di un virus, e la loro patogenesi è spesso strettamente correlata all'insorgenza della Sclerosi Multipla [5]. In effetti, insieme alla comparsa di neuriti ottiche, si assiste frequentemente alla perdita di una o più placche di mielina, ovvero a una demielinizzazione. La sintomatologia comprende un rapido calo della vista, dolore in seguito al movimento dell'occhio, un'alterazione del campo visivo e della percezione dei colori. Tra gli esami diagnostici vengono utilizzati i Potenziali evocati visivi, l'esame del campo visivo e la Risonanza magnetica; dal punto di vista terapeutico, invece, dà buoni risultati la somministrazione di cortisone.

D'altra parte, tra le patologie degenerative del nervo ottico spicca il **glaucoma**: malattia che provoca un progressivo schiacciamento della testa del nervo ottico e una conseguente graduale diminuzione del campo visivo. Questo, per altro, risulta essere l'unico sintomo percepito dal paziente, ritardando così la diagnosi fino alla comparsa di danni irreversibili. Anche in questo caso l'esame diagnostico preferibile è l'esame del campo visivo, e a livello terapeutico vengono somministrati specifici colliri per abbassare la pressione intraoculare [6].

Ciò che, invece, è di maggiore interesse per il lavoro presentato è un'alterazione specifica del nervo ottico: la variazione dello spessore, quindi del diametro, del nervo stesso e della sua guaina. Questa può essere causata indirettamente da altre patologie che non interessano esclusivamente il nervo ottico. Ad esempio, l'**OND** (Optic Nerve Diameter) può essere utilizzato come biomarker dell'atrofia del nervo stesso e della disabilità nei soggetti affetti da Sclerosi Multipla. È stata dimostrata l'esistenza di una correlazione tra la diminuzione dell'OND e l'EDSS (Expanded Disability Status Scale)<sup>4</sup> e la durata della malattia [7]. Un'ulteriore causa che può provocare la variazione del diametro del nervo ottico e della sua guaina, quella più ampiamente discussa, riguarda l'aumento della **pressione intracranica** (ICP). Un incremento dell'ICP può essere trasmesso allo spazio subaracnoideo che circonda il nervo ottico, inducendo un'espansione della sua guaina e un conseguente aumento dell'**ONSD** (Optic Nerve Sheath Diameter) [8]. Un'elevata pressione intracranica (ICP > 20 mmHg) può essere provocata da un trauma cranico, da un disturbo cerebrovascolare, quale ad esempio l'ictus cerebrale, o da ipertensione intracranica idiopatica; e può comportare un alto rischio di ischemia o addirittura decesso del paziente [9][10][11].

### 1.2.1 Relazione tra OND, ONSD e pressione intracranica

Nell'articolo di Maissan et al. [9] si illustra come la variazione dello spessore del nervo e il calcolo dell'ONSD possano essere indicatori accurati per il rilevamento di un'elevata ICP e delle sue variazioni in tempo reale, in soggetti che presentano trauma cranico. Le stesse conclusioni sono state tratte anche per pazienti con ictus [10] e per bambini con idrocefalo<sup>5</sup>, a cui sono stati impiantati degli shunt [12]. Un ulteriore studio mostra come sia possibile stabilire una correlazione tra un aumento dell'ONSD e un'alta ICP, con elevata specificità (93%) e sensibilità (88%) [13].

Attualmente il "gold standard" per la misurazione dell'ICP prevede l'utilizzo di una sonda intracranica. Quindi la procedura risulta estremamente invasiva

---

<sup>4</sup>L'EDSS è una scala utilizzata per misurare e valutare le caratteristiche cliniche e lo stato di invalidità dei pazienti affetti da Sclerosi Multipla.

<sup>5</sup>L'idrocefalo è una condizione che comporta l'accumulo di un'eccessiva quantità di liquido cerebrospinale in corrispondenza delle cavità del cervello.

e rischiosa: potrebbero insorgere complicazioni quali, ad esempio, infezioni ed emorragie. Inoltre, per un inserimento sicuro della sonda sono necessari un'ottima coagulazione del sangue, condizioni sterili e, naturalmente, la presenza di un neurochirurgo. Tutto ciò potrebbe risultare non replicabile sul luogo di un eventuale trauma cranico o disturbo cerebrovascolare, ritardando la diagnosi di elevata pressione intracranica e il conseguente intervento medico, che invece richiederebbero maggiore tempestività [9], [10]. Visti gli svantaggi di questa procedura e la relazione tra OND, ONSD e la pressione intracranica, si vuole indagare la possibilità di riconoscere un possibile aumento dell'ICP tramite la valutazione dei diametri del nervo ottico e della sua guaina. Adoperando, in questo senso, una tecnica non invasiva, come l'imaging del nervo ottico.

### 1.3 Imaging del nervo ottico

Lo strumento ideale per la ricerca non invasiva delle variazioni della pressione intracranica potrebbe essere la **Risonanza Magnetica** (MRI). Grazie a quest'ultima, infatti, è possibile ottenere immagini del nervo ottico con un'elevata risoluzione spaziale e una chiara delineazione delle strutture orbitali. Tuttavia, alcuni aspetti relativi all'esame di risonanza magnetica non lo rendono il più adatto per questa indagine medica. La limitata disponibilità, l'alto costo delle apparecchiature e i lunghi tempi d'attesa potrebbero non favorire, ad esempio, il suo utilizzo durante le fasi di follow-up [14]. D'altro lato, utilizzando come tecnica di imaging l'**Ultrasonografia**, si riescono ad ottenere risultati correlati positivamente con quelli della risonanza magnetica e con un'elevata accuratezza [14][15].

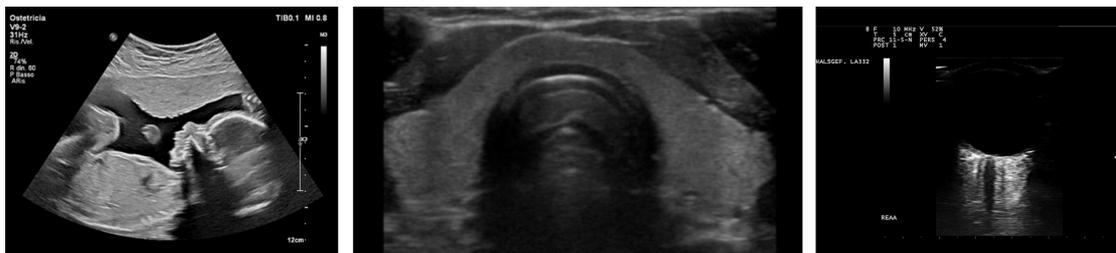
Un altro strumento utilizzato per l'imaging del nervo ottico, che consente di calcolare OND e ONSD è la **Tomografia Computerizzata** (TC). Anche in questo caso si riporta, però, come l'ultrasonografia riesca ad ottenere dei risultati con una buona accuratezza diagnostica se comparati con quelli della TC [16]. In generale, quindi, l'ultrasonografia risulta essere una accurata, rapida ed economica tecnica di imaging, per l'indagine medica considerata, sia se paragonata alla risonanza magnetica che alla tomografia computerizzata. Permette di evitare gli elevati costi della MRI, l'esposizione ad energia ionizzante della TC e di diagnosticare la presenza di elevata ICP anche in circostanze non ospedaliere grazie alla strumentazione portatile. Tutto ciò rende l'ultrasonografia l'imaging del nervo ottico più promettente per la misurazione dell'OND e dell'ONSD.



## Capitolo 2

# Ultrasonografia

### 2.1 Cos'è l'ultrasonografia



**Figura 2.1:** Esempi di immagini ecografiche. Da sinistra verso destra: ecografia di un feto [17], ecografia tiroidea [18], ecografia transorbitale di un nervo ottico (da dataset).

L'ultrasonografia, più comunemente nota come ecografia, è una tecnica di indagine medica mirata allo studio degli organi interni e in generale dei tessuti molli. Questa tecnica utilizza come vettore gli **ultrasuoni** (US) e in particolare la loro riflessione (eco). Ecco perché il nome ecografia [19]. Visto il mancato impiego di energie ionizzanti, l'ultrasonografia risulta essere meno pericolosa, in quanto riduce di molto il rischio di danni biologici. Questo la rende estremamente adatta ai casi più critici e ad un utilizzo prolungato, ovvero a procedure di monitoraggio (follow-up). Inoltre, i device attualmente in commercio riescono ad ottenere un'elevata risoluzione temporale: è possibile produrre 300 immagini al secondo; potendo monitorare anche stadi in evoluzione e fornendo immagini in tempo reale.

Gli ultrasuoni sono onde meccaniche che presentano frequenze molto alte, che variano tra 1-2 MHz e 20 MHz, al di sopra dell'udibile umano. Queste viaggiano all'interno del tessuto, mettendone in moto le molecole, con una velocità di propagazione che varia in base al tessuto e alla sua elasticità. Nello specifico, la velocità

dell'onda dipende anche dalla sua frequenza di emissione, tramite la relazione:

$$v = \lambda f \quad (2.1)$$

dove  $v$  è la velocità di propagazione,  $\lambda$  la lunghezza d'onda e  $f$  la frequenza. La lunghezza d'onda dà una misura della risoluzione spaziale che si potrà ottenere nell'immagine ecografica. Dalla relazione 2.1 è chiaro come l' $\lambda$  sia indirettamente proporzionale al valore della frequenza dell'onda ( $\lambda = \frac{v}{f}$ ). Di conseguenza, maggiore sarà la frequenza utilizzata, minore sarà la lunghezza d'onda e, quindi, migliore sarà la risoluzione spaziale.

Gli ultrasuoni durante la loro propagazione attraverso i tessuti subiscono una progressiva attenuazione causata dall'assorbimento da parte dei tessuti stessi. Questo limita il livello di profondità raggiungibile dalle onde e, di conseguenza, di indagine medica. L'ampiezza dell'onda diminuisce con andamento esponenziale:

$$A(\zeta) = A_0 e^{-\alpha \zeta} \quad (2.2)$$

dove  $A(\zeta)$  è l'ampiezza dell'onda alla profondità  $\zeta$ ,  $A_0$  è l'ampiezza iniziale e  $\alpha$  è il **coefficiente di assorbimento**. Quest'ultimo è specifico per ogni tessuto e dipende dalla frequenza di emissione dell'onda:  $\alpha$  aumenta con l'aumentare di  $f$ . Sarebbe, quindi, possibile contenere l'assorbimento degli ultrasuoni limitando la frequenza con cui questi vengono emessi. Ciò entrerebbe in conflitto con la necessità di frequenze elevate per ottenere una buona risoluzione spaziale. È, quindi, indispensabile trovare un compromesso tra risoluzione e assorbimento. In genere, un'indagine mirata ad organi più superficiali permette l'uso di frequenze più elevate, favorendo la risoluzione spaziale. Diversamente l'acquisizione di immagini di organi più profondi richiede frequenze più basse, in modo da limitare l'assorbimento [20].

### 2.1.1 Formazione dell'immagine

I dispositivi ad ultrasuoni generano le onde meccaniche grazie a cristalli piezoelettrici<sup>1</sup> inseriti all'interno di una sonda, posta a contatto con la cute del soggetto durante l'esame ecografico. L'energia viene quindi trasferita per contatto.

Gli ultrasuoni, propagandosi all'interno del corpo, incontrano tessuti e organi diversi dei quali riconoscono un'unica grandezza fisica: l'**impedenza acustica**. Questa è specifica per ogni tessuto e dipende dalla densità di quest'ultimo:

$$z = \rho v \quad (2.3)$$

---

<sup>1</sup>I materiali piezoelettrici emettono energia in modo anisotropo e sono in grado di trasformare energia elettrica in energia meccanica e viceversa.

dove  $z$  è l'impedenza acustica,  $v$  la velocità di propagazione e  $\rho$  la densità del tessuto. È proprio la differenza di impedenza acustica tra i vari tessuti a generare la riflessione degli ultrasuoni, gli echi. Quando l'onda si imbatte in mezzi con valori di  $z$  differenti, parte dell'energia viene riflessa e parte si propaga nel mezzo; maggiore è la differenza, maggiore è la quantità di energia riflessa. Come descrive la seguente relazione:

$$R = \left( \frac{Z_1 - Z_2}{Z_1 + Z_2} \right)^2 \quad (2.4)$$

in cui  $R$  è il **coefficiente di riflessione** e  $Z_1$  e  $Z_2$  sono i valori di impedenza acustica di due tessuti diversi attraverso i quali l'onda si sta propagando [20]. Ad esempio, l'osso ha un valore di impedenza acustica molto elevato e molto diverso da quello degli altri tessuti. Quindi, quando gli ultrasuoni lo incontrano vengono quasi completamente riflessi, impedendo l'indagine medica dei tessuti oltre l'osso. Le onde meccaniche riflesse tornano verso la sonda e la componente elettronica dell'ecografo le misura trasformando queste informazioni in immagini. Più nel dettaglio, l'immagine così acquisita corrisponde alla rappresentazione spaziale di tutti i punti in cui gli ultrasuoni hanno incontrato una discontinuità, riproducendo in questo modo il profilo di organi e tessuti.

Nell'immagine finale potremmo individuare tessuti diversi in base alla loro capacità di generare echi:

- **Anecogeno** se non genera echi di ritorno. Apparirà di colore nero sull'immagine.
- **Ipoecogeno** quando genera una poca quantità di echi. Apparirà di colore grigio scuro.
- **Ipercogeno** se viene generata un'elevata quantità di echi. Apparirà di colore bianco sull'immagine.

### 2.1.2 Tipologie di sonde

Le sonde moderne presentano schiere di sensori (array), ognuno dei quali è responsabile di una linea di scansione. Gli array possono essere configurati in modo che le linee di scansione formino fasci rettangoli o a ventaglio. Le principali tipologie di sonde utilizzate sono:

- **Sonda Lineare:** ha un array di tipo lineare, utilizzata principalmente per applicazioni superficiali (3-4 cm sotto la cute). Impiega frequenze elevate (7-15 MHz) e quindi alta risoluzione.

- **Sonda Convex:** ha un array curvo e permette di vedere organi con dimensioni maggiori (grazie al campo acustico a ventaglio) e più in profondità, come la cavità addominale. Lavora con frequenza più basse (2-7 MHz).
- **Sonda Phased Array:** presenta un appoggio molto piccolo e consente di modificare la direzione del fascio durante la scansione. Utilizzata per ecografie transtoraciche e transcraniche.



**Figura 2.2:** Sonda Lineare [21].



**Figura 2.3:** Sonda Convessa [21].



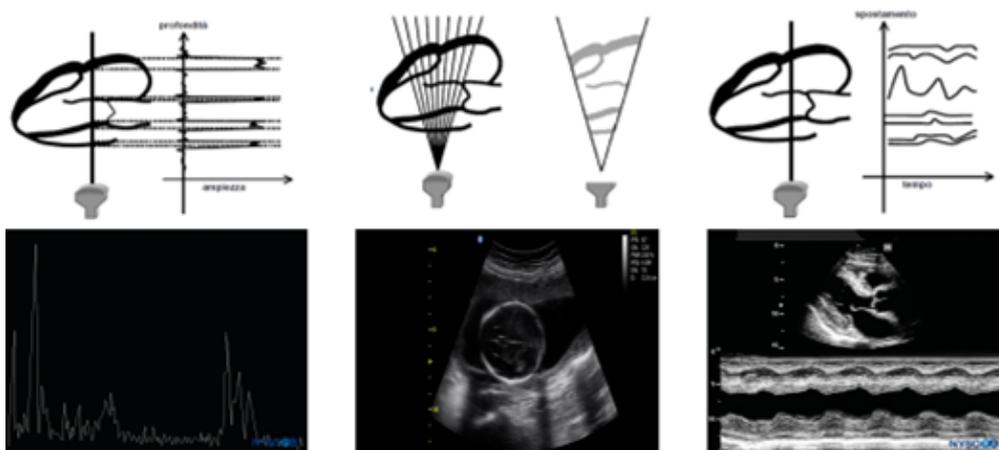
**Figura 2.4:** Sonda Phased Array [21].

### 2.1.3 Strategie di visualizzazione

Le strategie di visualizzazione di un dispositivo ad ultrasuoni sono:

- **A-Mode (Amplitude):** questa modalità prevede l'utilizzo di una sola linea di scansione; a video verrà mostrato un segnale monodimensionale e non una vera e propria immagine. Per le applicazioni mediche questa tecnica non viene più utilizzata.

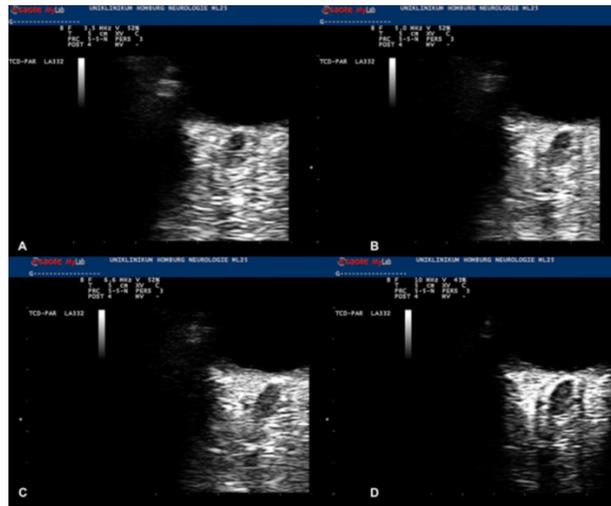
- **B-Mode (Brightness)**: vengono utilizzate più linee di scansione e la formazione dell'immagine coincide con quella descritta nei paragrafi precedenti. È la strategia di visualizzazione più comunemente usata.
- **M-Mode (Motion)**: anche questa tecnica prevede l'uso di una sola linea di scansione; ma in questo caso si valuta come variano le discontinuità incontrate nel tempo.



**Figura 2.5:** Strategie di scansione. Da sinistra a destra: A-Mode, B-Mode, C-Mode [3].

## 2.2 Ultrasonografia Transorbitale

L'ultrasonografia transorbitale (**TOS**) è l'esame di imaging che consente di valutare l'OND, l'ONSD e indirettamente la pressione intracranica (ICP); utilizzando una modalità non invasiva, altamente accessibile e user-friendly. L'ecografia transorbitale viene acquisita in modalità B-Mode, utilizzando una sonda lineare ad alta frequenza ( $> 7.5$  MHz); requisito necessario per ottenere una risoluzione spaziale laterale adeguata ( $\leq 0.4$  mm). L'uso di frequenze inferiori fornirebbe risoluzioni più basse e potrebbe compromettere la distinzione delle strutture del nervo ottico e della sua guaina, viste le loro ridotte dimensioni anatomiche. Nella figura 2.6 sono mostrate 4 diverse acquisizioni di ecografie transorbitali, ottenute con frequenza differenti. Come si può osservare utilizzando frequenze troppo basse ( $< 7.5$  MHz), le strutture del nervo ottico e in particolare la sua guaina non sono chiaramente riconoscibili [22].



**Figura 2.6:** Ecografie Transorbitali acquisite con frequenze diverse. (A 3.5 MHz, B 5 MHz, C 6.6 MHz, D 10 MHz) [22].

L'esame in questione viene eseguito su soggetti in posizione supina, con la parte superiore del corpo e la testa sollevati di un angolo di 20 – 30° rispetto all'orizzontale, nell'ottica di ridurre eventuali variazioni di pressioni sull'occhio. Inoltre, viene chiesto ai soggetti di mantenere l'occhio in una posizione intermedia e neutra con lo scopo di limitare il più possibile qualunque movimento oculare. La sonda viene, poi, appoggiata sulla zona temporale della palpebra superiore, tenuta chiusa, usando un sottile strato di gel ultrasonico. In questo modo verrà acquisita un'immagine raffigurante la parte anteriore del nervo ottico nel suo decorso longitudinale [11]. Nella figura 2.7 sono riportate due immagini esplicative dell'esecuzione di un esame di ultrasonografia transorbitale.



**Figura 2.7:** Da sinistra: esempio postazione per l'acquisizione dell'ecografia transorbitale, esecuzione di un'ecografia transorbitale [23][24].

Per ultimo, ma non meno importante, un esame di sonografia transorbitale richiede un operatore esperto. Come tutte le tecniche di imaging ecografiche, questa procedura risulta essere operatore-dipendente: la preparazione di chi esegue l'esame può influenzare e compromettere, sia positivamente che negativamente, l'acquisizione dell'immagine. È necessaria un'ottima conoscenza della strumentazione, delle strutture oggetto dell'esame e una sviluppata capacità di osservazione [19]. Nonostante questo, se l'ecografia transorbitale viene eseguita correttamente, rimane un'ottima metodica per misurare il diametro del nervo ottico e quello della sua guaina presentando un'elevata affidabilità intra e inter-operatore [25].

## 2.3 Calcolo di OND e ONSD da un'immagine di ultrasonografia transorbitale



**Figura 2.8:** Immagine di ecografia transorbitale (da dataset).

La figura 2.8 mostra come appare un'immagine di tipo TOS. Si possono osservare nella metà superiore il bulbo oculare, ipoecogeno, e nella metà inferiore le strutture del nervo ottico. Più nel dettaglio, partendo dal suo centro e spostandosi verso l'esterno, il nervo ottico è così rappresentato:

- Fibre nervose che costituiscono la zona ipoecogena del nervo.
- Pia madre che avvolge le fibre nervose e appare iperecogena.
- Spazio subaracnoideo anch'esso iperecogeno grazie alle sue strutture trabecolari.

- Dura madre e grasso periorbitale iperecogeni.

Avendo chiaro come identificare le strutture anatomiche del nervo ottico è possibile calcolare il suo diametro (OND) e quello della guaina (ONSD). Entrambi vengono valutati lungo l'asse centrale del nervo a una distanza di 3 mm dalla papilla, ovvero dal bordo inferiore del bulbo oculare (distanza 1 indicata nella figura 2.9). In particolare, il diametro del nervo ottico (OND) viene misurato in modo che corrisponda alla distanza interna alla pia madre (segmento 2-2 della figura 2.9). Diversamente il diametro della guaina (ONSD) viene misurato tra i bordi esterni iperecogeni dello spazio subaracnoideo, coincidendo quindi con la distanza interna alla dura madre (segmento 3-3 della figura 2.9) [22].



**Figura 2.9:** Immagine di ecografia transorbitale in cui sono stati evidenziati OND (2-2) e ONSD (3-3) [22].

Come accennato nel paragrafo precedente, l'acquisizione di immagini ecografiche transorbitali risulta essere dipendente dall'operatore. Allo stesso modo anche le misurazioni dell'OND e dell'ONSD possono risentire dell'esperienza di chi la esegue, oltre che da eventuali artefatti dell'immagine. In tal senso si indaga lo sviluppo di sistemi automatici per la segmentazione delle strutture del nervo ottico. L'impiego di tali sistemi potrebbe ridurre la dipendenza dall'operatore e, di conseguenza, la variabilità inter-osservatore [26].

## Capitolo 3

# Proposte di sistemi automatici per il calcolo dell'OND e dell'ONSD

Come è stato anticipato la misurazione manuale dei diametri del nervo ottico (OND) e della guaina (ONSD) risente eccessivamente dell'operatore e dell'errore umano. La problematicità non riguarda solo un'eventuale elevata variabilità inter-osservatore, ma anche le elevate tempistiche richieste da una misurazione manuale. L'impiego di sistemi completamente automatici consentirebbe di sviluppare un unico modello di misurazione e di standardizzare il metodo di elaborazione delle misure. Questo ridurrebbe i tempi di calcolo dei diametri.

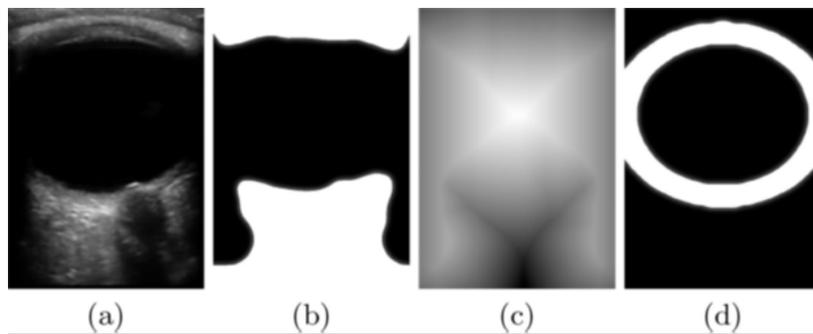
Di seguito sono riportate alcune proposte di algoritmi pensati per rendere automatiche le misurazioni del diametro del nervo ottico e della guaina (principalmente di quest'ultima). Questi sono risultati tutti dei modelli affidabili. Però, come verrà specificato, gli studi raccolti presentano tutti la stessa problematica: l'utilizzo di un dataset limitato a poche immagini ecografiche transorbitali. Inoltre, si potrà notare come tutte le proposte sfruttino tradizionali tecniche di *image processing*, e nessuna indaghi approcci che sfruttino il deep learning. Scelta sicuramente legata al ridotto numero di immagini a disposizione. Proprio per questo, come verrà ampiamente discusso nei capitoli successivi, il lavoro di tesi qui presentato si avvarrà di un dataset più ampio e di tecniche di deep learning.

### 3.1 Primo modello per il calcolo automatico del diametro della guaina del nervo ottico (ONSD)

Il lavoro di Gerber et al. del 2017 [27] risulta essere la prima proposta di un algoritmo automatico in grado di misurare il diametro della guaina del nervo ottico (ONSD). L'obiettivo prefissato è quello di valutare lesioni cerebrali traumatiche sul campo, senza la necessità di personale esperto e riducendo i tempi di intervento. L'algoritmo presenta un approccio a due fasi:

**Fase1.** Definizione della posizione del bulbo oculare e stima delle sue dimensioni attraverso alcuni passaggi elencati di seguito:

- Elaborazione delle immagini mediante un filtro passa basso (FPB) Gaussiano e creazione di una maschera binaria utilizzando un valore soglia (Step *b* della figura 3.1).
- Calcolo della *distance transform*<sup>1</sup>, la cui distanza massima (punto più chiaro) coincide con il centro del bulbo oculare (Step *c* della figura 3.1).
- Seconda applicazione della *distance transform* che fornisce gli assi (minore e maggiore) dell'ellisse corrispondente ai bordi del bulbo oculare; e conseguente creazione della maschera binaria dell'ellisse (Step *d* della figura 3.1)

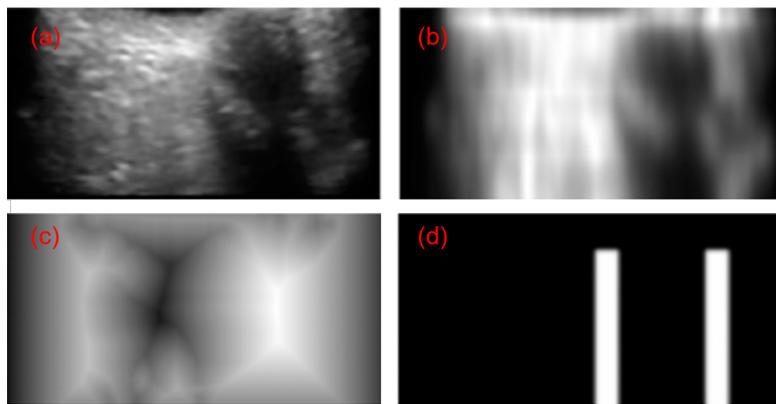


**Figura 3.1:** Step intermedi della localizzazione del bulbo oculare. (A) Immagine di Input, (B) FPB Gaussiano e maschera binaria, (C) Distance Transform, (D) Ellisse corrispondente ai bordi del bulbo oculare [27].

<sup>1</sup>La distance transform è un'operazione applicata a immagini binarie e dà come risultato un'immagine, il cui valore dei pixel corrisponde alla distanza euclidea tra il pixel stesso e il pixel più vicino che ha valore diverso da zero [28]

**Fase2.** Definizione, dall'ellisse, della posizione approssimativa del nervo ottico e di due barre corrispondenti alle sue pareti:

- Estrazione della ROI (nervo ottico) utilizzando l'ellisse ottenuto nella fase 1 (Step *a* della figura 3.2).
- Applicazione di un FPB Gaussiano (Step *b* della figura 3.2) e creazione di una maschera binaria utilizzando un valore soglia.
- Applicazione della *distance transform* alla maschera binaria. La distanza massima individuata fornisce il diametro del nervo ottico e il suo centro (Step *c* della figura 3.2)
- Con le informazioni ottenute viene creata una maschera binaria con due barre verticali che corrispondono alla guaina del nervo (Step *d* della figura 3.2).



**Figura 3.2:** Step intermedi per l'individuazione della guaina del nervo. (A) ROI, (B) FPB Gaussiano, (C) Distance Transform, (D) Barre verticali corrispondenti alla guaina [27].

Ultimato il processing, i risultati ottenuti dalle due fasi vengono sovrapposti all'immagine originale (figura 5.8) e viene calcolato il diametro della guaina del nervo ottico e una distanza di 3 mm dalla papilla (fondo del bulbo oculare).

La tecnica di automazione del calcolo dell'ONSD, appena illustrata, è stata testata su 23 immagini di sonografia transorbitale, i cui diametri sono stati misurati da 13 volontari, sia principianti che esperti. Il coefficiente di correlazione, ottenuto dal confronto tra i valori automatici e quelli di un medico esperto, conferma il corretto funzionamento dell'algoritmo proposto ( $R=0.85$ ) [27].

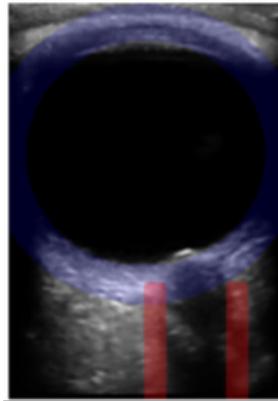


Figura 3.3: Immagine di Output [27].

### 3.2 AUTONoMA - AUTomatic Optic Nerve MeAsurement

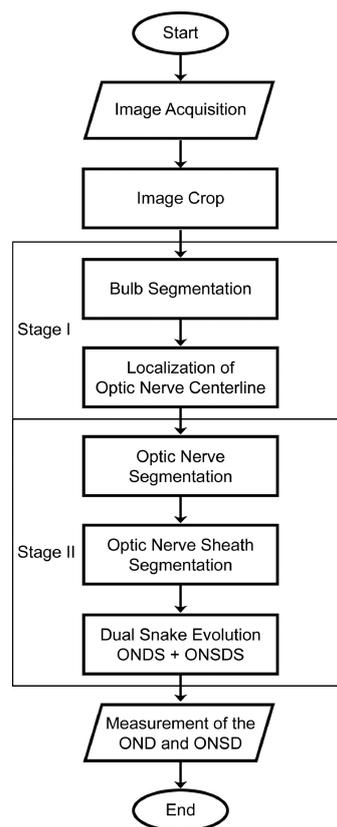


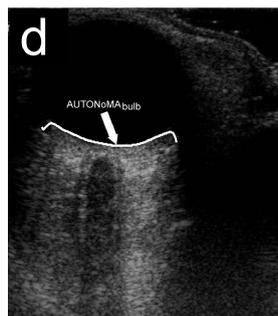
Figura 3.4: Flow-chart dell'algoritmo [29].

In base alla letteratura, lo studio di Meiburger et al. del 2020 [29], AUTONoMA, si rivela essere il primo a presentare un sistema completamente automatico per la misurazione del diametro sia del nervo ottico (OND) che della sua guaina (ONSD) in immagini ecografiche.

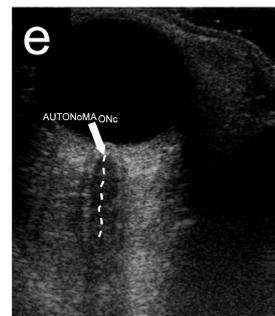
L’algoritmo proposto si basa su modelli deformabili ed è strutturato su due fasi automatiche principali. Nella figura 3.4 ne è rappresentato un flow-chart.

**Fase1.** Localizzazione grossolana della ROI attraverso il profilo del bulbo oculare e il tracciamento della centerline dell’ON (Optic Nerve).

- Ritaglio dell’immagine utilizzando operatori morfologici e gradienti.
- Applicazione del filtro derivato gaussiano ( $\sigma = 7$ ).
- Localizzazione automatica del bordo inferiore del bulbo oculare, attraverso una ricerca euristica lungo le colonne. L’inizio di questa ricerca corrisponde al primo pixel (a partire dall’immagine in alto) il cui valore si trova al di sopra di una certa intensità e che presenta, sopra di esso, una zona ipoecogena ragionevolmente grande. Il primo punto di discontinuità da una zona ipoecogena a una zona iperecogena, viene classificato come punto appartenente al confine del bulbo ottico. Ogni punto, ciascuno ottenuto da una colonna diversa, viene interpolato per dare la segmentazione del bulbo oculare.
- Applicazione del filtro derivato gaussiano.
- Localizzazione della centerline dell’ON attraverso una ricerca euristica, al di sotto del profilo del bulbo oculare, simile a quella precedentemente descritta; ma effettuata lungo le righe.



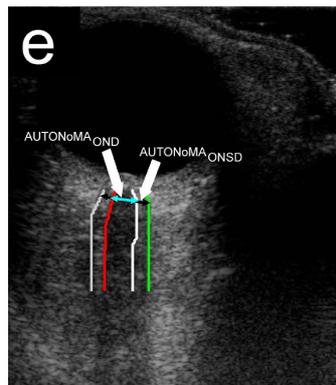
**Figura 3.5:** Profilo del Bulbo [29].



**Figura 3.6:** Centerline del Nervo [29].

**Fase2.** Segmentazione del nervo ottico e della sua guaina attraverso doppio snake e misurazione automatica dell’OND e dell’ONSD.

- Partendo dalla centerline, il modello a doppio snake viene inizializzato individuando un primo grezzo bordo del nervo. Nel dettaglio, questo viene trovato grazie a una ricerca euristica lungo le righe dell'immagine originale filtrata con un filtro derivativo gaussiano in due direzioni. Partendo dall'inizio della centerline, il primo pixel il cui valore è superiore a una soglia specifica viene candidato come punto del bordo dell'ON.
- Partendo dal bordo del nervo, viene inizializzato un primo grezzo bordo della guaina. L'inizializzazione avviene utilizzando una tecnica simile a quella precedentemente descritta.
- Evoluzione del modello a doppio snake grazie a tre energie:
  1. ENERGIA INTERNA necessaria per vincolare la forma del contorno ed evitare un'eccessiva curvatura.
  2. ENERGIA ESTERNA necessaria per attirare il modello verso le discontinuità dell'immagine.
  3. ENERGIA DI INTERAZIONE RECIPROCA, un secondo termine di energia esterna, necessario per evitare che il modello collassi o converga.



**Figura 3.7:** Segmentazione finale dell'ON e dell'ONS [29].

Una volta segmentati ON e ONS (Optic Nerve Sheath), i diametri di entrambi vengono misurati automaticamente a una distanza di 3 mm dal bordo inferiore del bulbo oculare.

L'algoritmo proposto è stato applicato su un database comprensivo di un totale di 75 immagini, acquisite sia su soggetti patologici che sani. I valori di OND e ONSD ottenuti dal sistema automatico sono stati confrontati con le misure manuali di due operatori (uno esperto e uno no). Per il confronto sono stati calcolati diversi parametri: errore medio, errore assoluto medio, errore quadratico medio, il FoM

(Figure of Merit che caratterizza le prestazioni complessive dell'algoritmo) e il coefficiente di correlazione.

Il sistema proposto è completamente automatico, indipendente dall'utente ed è stato in grado di elaborare il 95% delle immagini nel dataset. In base ai risultati ottenuti, nell'uso medico il sistema AUTONoMA può essere considerato un valido strumento per ridurre l'ampia variabilità dei valori di ONSD e OND e standardizzare le misure. Tuttavia, l'algoritmo AUTONoMA richiede una differenza sostanziale tra le strutture iperecogene e ipoecogene del nervo, per poter segmentare correttamente le immagini. Inoltre, nel set di dati utilizzato per lo studio non c'erano immagini ottenute da diverse macchine ad ultrasuoni. E anche in questo caso si sottolinea come sia necessario analizzare un numero maggiore di immagini per convalidare il sistema proposto [29].

### 3.3 Algoritmo per la misurazione automatica dell'ONSD basato sul Clustering

Anche lo studio di Rajajee et al. del 2021 [30] si prefissa lo sviluppo di un algoritmo che calcoli solo il diametro della guaina del nervo ottico (ONSD). La procedura automatica proposta si basa sul Clustering<sup>2</sup> e implementa diverse tecniche di elaborazione delle immagini:

- Identificazione della ROI contenente solo la guaina del nervo, analizzando l'immagine integrale.
- Separazione della guaina sia dal tessuto circostante che dal nervo ottico stesso, consentendo di valutare quantitativamente l'ONSD.
- Esecuzione del clustering per combinare i pixel in base alla loro intensità.
- Utilizzando i risultati del clustering, individuazione della riga dell'immagine in corrispondenza dei 3 mm di distanza dal bulbo oculare e calcolo della sua derivata.
- Localizzazione dei bordi e calcolo dell'ONSD, utilizzando i valori della derivata.

Tutti questi passaggi, che costituiscono l'algoritmo, sono stati applicati in modo automatico su 44 immagini acquisite su soggetti patologici. I risultati sono stati confrontati con le misurazioni dell'ONSD di due operatori esperti.

---

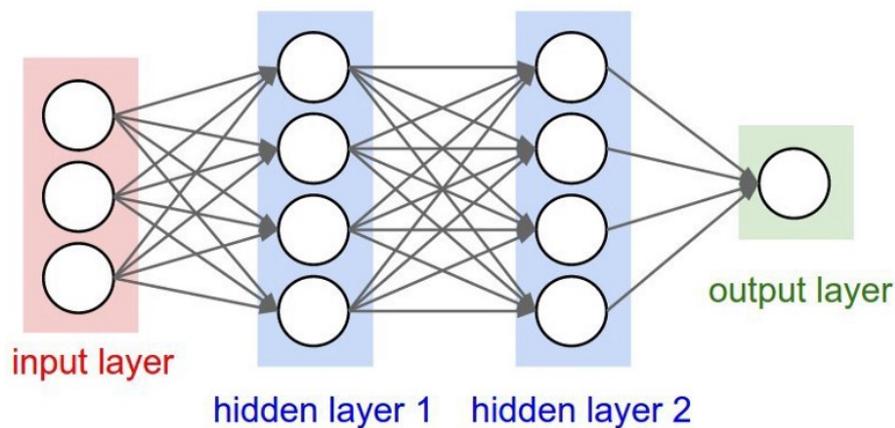
<sup>2</sup>Il clustering è una tecnica volta al raggruppando di elementi omogenei in insiemi di dati, chiamati *cluster*. La suddivisione avviene sulla base dei valori dei loro attributi, in modo che in un cluster siano raccolti gli oggetti che presentano similarità tra loro e dissimilarità con gli elementi degli altri cluster.

Per la validazione dei risultati sono stati calcolati l'ICC (Interclass correlation coefficient), il coefficiente di correlazione di Pearson, la differenza degli errori percentuali (operatori VS algoritmo). Dai valori dei coefficienti di correlazione ottenuti ( $R=0.73$  e  $ICC=0.8$ ) l'algoritmo proposto risulta essere un metodo affidabile per elaborare immagini ad ultrasuoni e misurare ONSD con alta precisione. In ogni caso, gli autori sottolineano come sia indispensabile ingrandire il dataset di immagini per una migliore validazione [30].

## Capitolo 4

# Deep Learning per l'*image segmentation*

Il Deep Learning (DL) è una tecnica di *Machine Learning* (ML) ispirata al modo in cui il cervello umano risponde a un input e prende decisioni, simulandone il processo di apprendimento [31]. Gli algoritmi di Deep Learning insegnano al computer come filtrare input di varia natura con lo scopo di imparare a predire e classificare le informazioni: dato un input  $X$  lo utilizzano per predire l'output  $Y$  [32], [33]. Il Deep Learning si avvale dell'uso delle Reti Neurali. Queste vengono allenate con un ampio dataset di dati di ingresso con l'obiettivo di minimizzare le differenze tra le predizioni del modello e gli output attesi, considerati veritieri.



**Figura 4.1:** Struttura base di una rete neurale [33].

La struttura base di una generica rete neurale è costituita da tre *layer* principali:

- **Input Layer** che riceve in ingresso i dati numerici.

- **Hidden Layers** o livelli *nascosti* necessari per l'apprendimento.
- **Output Layer** che fornisce in uscita i risultati predetti dal modello.

Come si può osservare dalla figura 4.1 ogni layer è composto da più *nodi*, meglio noti come **neuroni**. I neuroni di ogni strato sono collegati con quelli dei livelli successivi tramite opportune funzioni di attivazione pesate. L'algoritmo di apprendimento si basa proprio sull'aggiornamento di tali pesi ( $W$ ) e bias ( $b$ ), con il fine di minimizzare l'errore tra output predetto e output atteso [33]:

$$J(W, b) = \frac{1}{m} \sum_{i=1}^m L(\hat{y}_i, y_i) \quad (4.1)$$

dove  $L$  è la funzione di perdita,  $\hat{y}_i$  è l'output generato dalla rete e  $y_i$  quello atteso.

Negli ultimi anni il Deep Learning è diventato approccio dominante nelle attività di **Computer Vision** (o visione artificiale), favorendone un rapido avanzamento. La Computer Vision è un campo di studi interdisciplinare che indaga come i computer possano riprodurre funzioni e processi visivi umani. L'obiettivo non è limitarsi al solo riconoscimento di oggetti all'interno di immagini e video, ma quello di ottenere una comprensione e una capacità di estrazione di informazioni (necessarie per l'elaborazione) di alto livello [34]. Gli algoritmi di Computer Vision sono, quindi, in grado di eseguire indagini a diversi livelli di astrazione, in base al tipo di compito che deve essere effettuato. Tra i più comuni [34][35]:

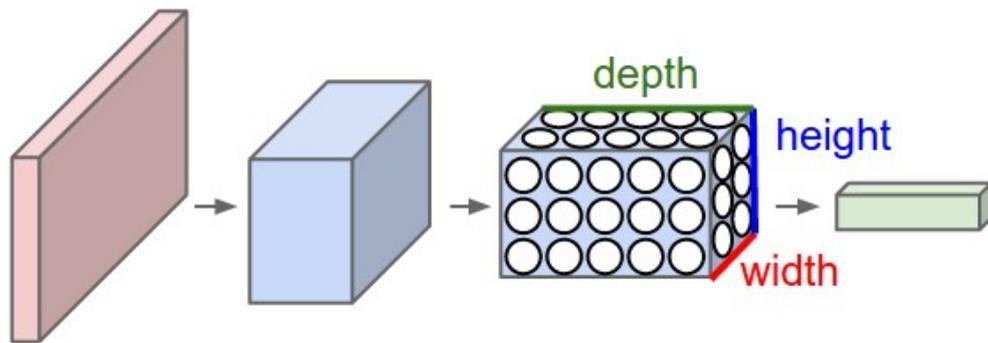
- **Image Classification:** classificazione del contenuto dell'immagine con assegnazione di un'*etichetta* (Ad es. cane, gatto).
- **Object Detection:** individuazione e classificazione di uno o più oggetti all'interno dell'immagine.
- **Semantic Segmentation:** suddivisione dell'immagine in gruppi di pixel che verranno classificati ed etichettati.
- **Pattern Recognition:** riconoscimento di pattern presenti all'interno dell'immagine (volti, posizioni del corpo, forme).
- **Object Tracking:** individuazione di uno o più oggetti di interesse e tracciamento della loro evoluzione nel tempo e nello spazio.

In tutti questi ambiti, per estrarre informazioni dalle immagini, gli algoritmi di Computer Vision hanno bisogno di essere addestrati con un gran numero di immagini. L'architettura principale utilizzata è costituita dalle **Reti Neurali Convolutionali** (CNN).

Tra i *task* elencati sarà di particolare interesse per il seguente lavoro di tesi la Segmentazione Semantica.

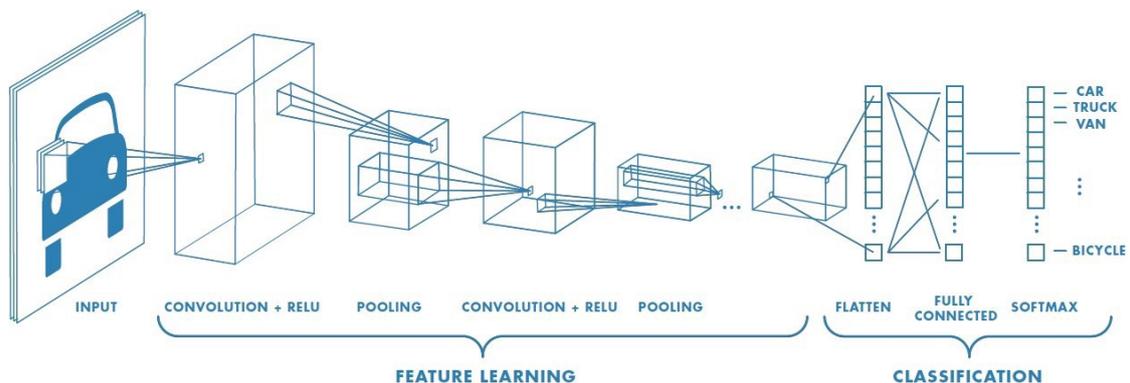
## 4.1 Reti Neurali Convoluzionali (CNN)

Le Reti Neurali Convoluzionali sono Reti Neurali *feed-forward* ispirate al funzionamento della corteccia visiva. A differenza delle tradizionali Reti Artificiali la loro architettura è strutturata specificatamente per assumere come input delle immagini. Questo si traduce in layer aventi neuroni a 3 dimensioni: larghezza, altezza e profondità. I neuroni di un livello saranno connessi solo a una piccola regione dello strato precedente. Diversamente, nelle Reti Neurali ordinarie i neuroni di ogni strato risultavano connessi a tutti i neuroni del layer precedente [36]. In figura 4.2 è possibile vedere un esempio degli strati di una CNN. Se paragonata alla figura 4.1, risulta evidente la differenza con le Reti Neurali classiche.



**Figura 4.2:** Esempio struttura degli strati di una Rete Neurale Convoluzionale [36].

L'architettura di una CNN tipicamente vede due blocchi principali: il **blocco convoluzionale** e il **blocco classificatore**.



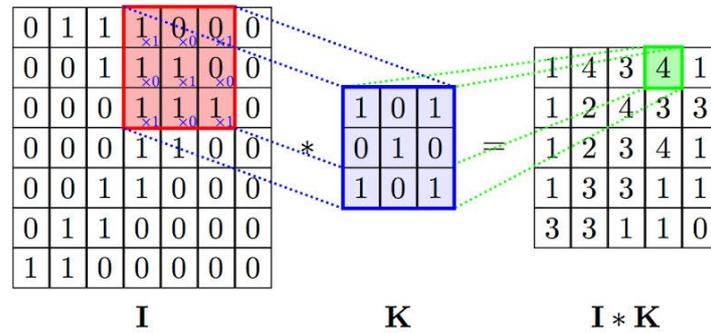
**Figura 4.3:** Architettura generica di una CNN [37].

Il primo estrae dalle immagini *features* e schemi, aumentando strato dopo strato il livello di astrazione e di specificità. Il secondo, di fatto, esegue la classificazione delle immagini, come ben esplicitato dal nome.

Il blocco convoluzionale è formato dal susseguirsi ripetuto dei seguenti layer [36][38]:

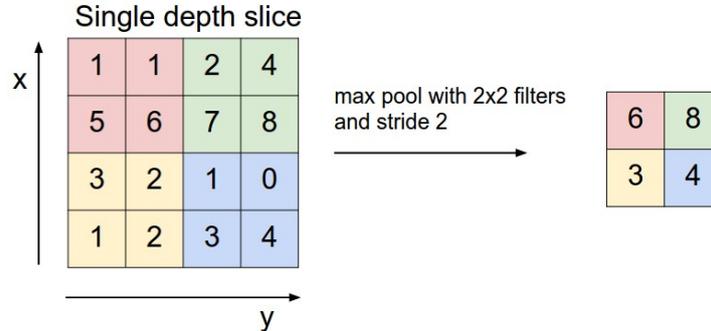
- **Convolutional Layer:** è il livello principale della rete e può essere ripetuto più volte all'interno della sua architettura. Ha come obiettivo individuare schemi all'interno dell'immagine; ad esempio curve, angoli, contorni, circonferenze e così via. Maggiore è il numero di strati convoluzionali presenti, maggiore sarà la complessità delle caratteristiche individuabili. I parametri di questo livello consistono in una serie di filtri digitali, o *kernel*. In genere le loro dimensioni sono minori rispetto alle corrispondenti dimensioni dell'immagine di input. Il filtro viene fatto scorrere, eseguendo un'operazione di convoluzione, lungo l'altezza e la larghezza dell'input calcolando il prodotto scalare tra i valori del filtro e quelli del **campo ricettivo**. Il campo ricettivo corrisponde alla porzione di input su cui il kernel si va a posizione ad ogni spostamento. L'output dell'operazione descritta è una matrice 2D chiamata **mappa di attivazione**. Questa acquisisce la terza dimensione, la profondità, in base al numero di filtri. Nella costruzione del livello convoluzionale è necessario definire i seguenti **iperparametri**:

1. *Numero di filtri:* influenza la profondità del volume di output. Si otterranno un numero di mappe di attivazione pari alla quantità di filtri applicati.
2. *Dimensione del filtro:* da questa dipendono l'altezza e la larghezza delle mappe di attivazione. In genere 3x3 o 5x5.
3. *Stride o passo:* indica di quanto il kernel viene fatto scorrere sull'immagine. Più alto sarà il passo, minore saranno le dimensioni della mappa di attivazione.
4. *Padding:* prevede l'applicazione di una "cornice" di zeri intorno alla matrice di input. Questo consentirebbe di ottenere un volume di output più ampio e, di conseguenza, di evitare di perdere informazioni utili durante il passaggio da un livello convoluzionale a un altro.



**Figura 4.4:** Esempio di un'operazione di convoluzione. Da sinistra a destra: Input  $I$  di dimensione  $7 \times 7$  (evidenziato in rosso il campo ricettivo), filtro  $K$  di dimensione  $3 \times 3$ , mappa di attivazione di output di dimensione  $5 \times 5$ . [39].

- **Pooling Layer:** è uno livello di raggruppamento periodicamente inserito tra strati successivi di convoluzione. Lo scopo è quello di ridurre progressivamente le dimensioni e quindi i parametri della rete. Anche il *pooling* viene eseguito facendo scorrere sulla matrice di input un filtro, spesso di dimensioni  $2 \times 2$  e con passo 2. Le operazioni possono essere di massimo, **max pooling**, oppure di media, **average pooling**. La prima è la più comune e ne è riportato un esempio nella figura 4.5



**Figura 4.5:** Esempio di *max pooling* [36].

- **Activation Layer:** livello che prende come input l'output dello strato convoluzionale precedente e ne applica una funzione di attivazione. Tra le funzioni più comuni:
  - **Sigmoide:** funzione adatta alle classificazioni binarie che limita l'uscita tra 0 e 1.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (4.2)$$

- **Tangente iperbolica:** funzione molto simile alla sigmoide ma centrata sullo zero. Fornisce output compresi tra -1 e 1.

$$\tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})} \quad (4.3)$$

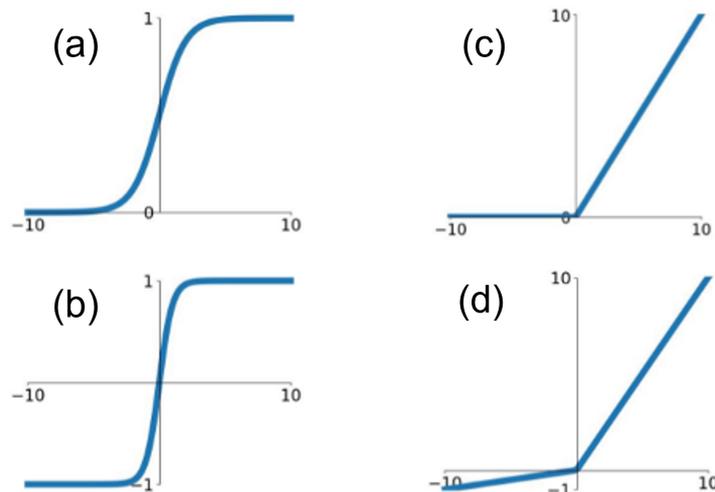
- **ReLU** (Rectified Linear Units): funzione che restituisce in output l'input stesso quando è positivo, oppure 0 nel caso in cui l'input sia negativo.

$$f(x) = \max(0, x). \quad (4.4)$$

- **Leaky ReLU:** versione modificata della funzione ReLU. I valori di input positivi rimangono invariati, come previsto dalla ReLU. Ma in questo caso è possibile avere come output dei valori negativi, quando l'input è negativo.

$$f(x) = \max(0.01x, x). \quad (4.5)$$

Nell'immagine 4.6 sono rappresentati i grafici delle funzioni citate.

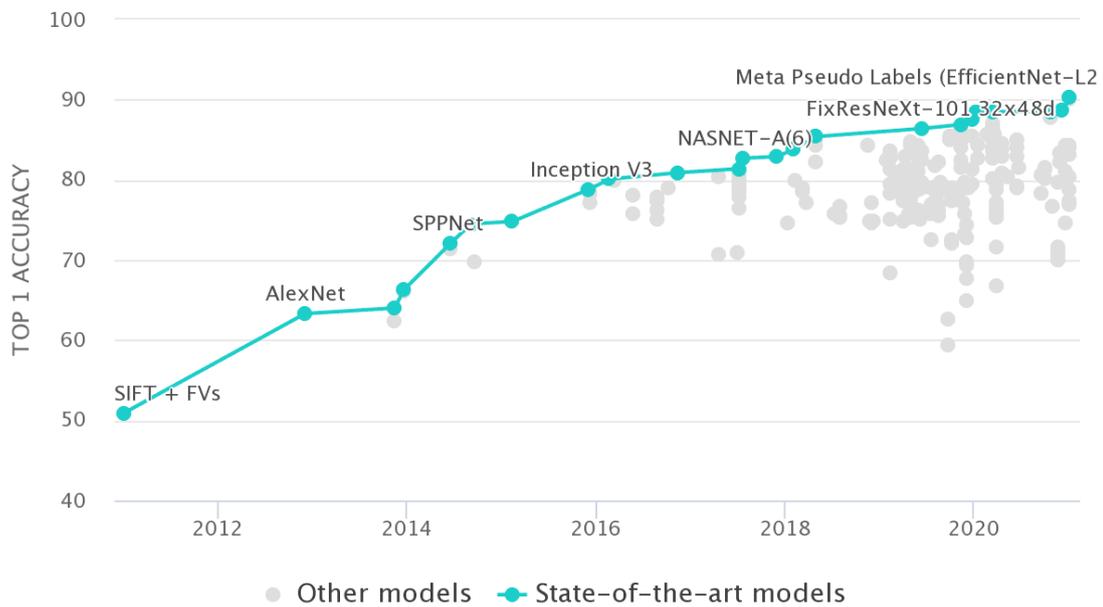


**Figura 4.6:** Grafici delle funzioni di attivazione: (a) Sigmoide, (b) Tangente iperbolica, (c) ReLU, (d) Leaky ReLU [40].

Terminato il blocco convoluzionale, l'architettura di una CNN prevede un blocco di classificazione. Questo è costituito dal **Fully Connected layer**, nel quale i neuroni sono completamente connessi allo strato precedente. Questo ultimo layer determina la classe a cui le caratteristiche, individuate dal blocco convoluzionale, sono maggiormente correlate; ovvero classifica l'immagine.

Le CNN, così strutturate, presentano un ingente numero di parametri da addestrare e richiedono un elevato tempo computazionale. Questo è aggirabile attraverso l'uso di **modelli pre-addestrati**, modelli precedentemente allenati su un set di dati di grandi dimensioni per risolvere un *task* simile a quello che ci si propone di risolvere. Si importano, quindi, modelli di CNN pre-addestrati e i rispettivi pesi, allenando nuovamente il modello sul proprio dataset. Lo scopo non è solo quello di ridurre i costi computazionali ma anche ovviare a un'eventuale carenza di dati di input. Questo approccio è noto come *transfer learning* [41].

### 4.1.1 Evoluzione dei modelli CNN



**Figura 4.7:** Evoluzione temporale delle performance delle architetture CNN sul dataset ImageNet [42].

Dal 2012 ad oggi si è assistito ad rapido progresso delle architetture e delle performance delle Reti Neurali Convolutionali. Si è partiti da un'accuratezza del 63.3%, ottenuta dalla rete AlexNet sul dataset ImageNet, fino a superare oggi il 90% con le reti EfficientNet.

**AlexNet**, proposta nel 2012 da A. Krizhevsky et al. [43], fu la prima CNN ad aver avuto una diffusa applicazione nel campo della Computer Vision. Costituita da 8 strati, di cui 5 convoluzionali e 3 completamente connessi, fu la prima ad implementare il *max-pooling* e la funzione ReLU nell'*Activation Layer*. Prevede più

di 60 milioni di parametri da addestrare ed è in grado di classificare le immagini in 1000 classi diverse [42][44].

Ispirandosi all'AlexNet, nel 2014 i membri del *Visual Geometry Group* (VGG), Karen Simonyan e Andrew Zisserman, svilupparono la rete **VGGNet** [45]. Fu il primo studio a dimostrare che l'aumento di strati consente di raggiungere prestazioni migliori. Quindi, si è pensato di arricchire l'architettura di AlexNet aggiungendo livelli convoluzionali. Nello specifico sono stati sostituiti gli strati 7x7 presenti in AlexNet con 3 layer convoluzionali 3x3. L'idea è che la successione di 3 strati 3x3 restituisca una mappa di attivazione paragonabile a quella di un'unico layer 7x7, ma riducendo di circa l'80% i parametri addestrabili. Gli sviluppatori della VGGNet hanno creato versioni della rete con diverse profondità: vanno da 11 a 19 strati. Le più comuni sono la VGG-16 e la VGG-19, in grado di raggiungere, rispettivamente, un'accuratezza del 74.4% e del 74.5% sul dataset ImageNet [42][44].

Sempre nel 2014, C. Szegedy et al. con il loro studio [46] si resero conto che è possibile ingrandire un modello aumentandone non solo la profondità, ma anche la larghezza. Proposero la rete **GoggleNet** o **InceptionNet**, introducendo per la prima volta i *moduli inception*. L'architettura prevede di impilare gli uni su gli altri moduli o blocchi, che a loro volta contengono all'interno strati convoluzionali. In questo modo realizzarono un modello più profondo e più largo, riuscendo a contenere il numero di parametri da addestrare. La famiglia delle reti Inception è riuscita a raggiungere nel 2015, con Inception-V3, un'accuratezza del 78.8% su ImageNet [42][44].

Tuttavia, superato un certo livello di profondità l'accuratezza del modello subisce una rapida degradazione. Con lo scopo di proporre un rimedio, nel 2015 nacquero le reti **ResNet**. La novità principale riguarda l'implementazione di blocchi strutturali detti *skip connection* o *blocchi residuali*. Le reti ResNet possono essere molto profonde, fino a 152 strati (ResNet-152), mantenendo una complessità inferiore rispetto ai modelli precedentemente sviluppati e raggiungendo un'accuratezza del 78.57% sul dataset ImageNet [42].

Per raggiungere prestazioni sempre migliori, nel 2020 vennero sviluppate le **EfficientNets**. La novità introdotta si basa sull'idea di aumentare le dimensioni del modello, ma questa volta in modo uniforme. Invece di puntare solo sulla profondità, si vogliono aumentare in modo proporzionato profondità, larghezza e risoluzione. Con la rete EfficientNet-B7 è stato possibile raggiungere un'accuratezza del 84.3% sul dataset ImageNet.

Tutti i modelli qui introdotti, più in generale le CNN, possono essere utilizzati come *encoder* in reti per segmentazione semantica (verrà spiegato meglio nei successivi paragrafi cosa si intende). Nel lavoro di tesi presentato vengono impiegate le reti **ResNet50** ed **EfficientNet-B4** proprio come encoder. Per questo motivo verranno approfondite separatamente nei paragrafi successivi.

### 4.1.2 ResNet

ResNet, proposta da K. He et al. [47], è la rete CNN classificata al primo posto al concorso ILSVRC (*ImageNet Large Scale Visual Recognition Challenge*) del 2015. Contest mirato alla classificazione di più di un milione di immagini del dataset ImageNet in 1000 classi differenti.

La principale novità introdotta dagli autori della rete è la presenza di blocchi noti come *skip connection* o **blocchi residuali**, da cui il nome *Residual Network* (ResNet). L'obiettivo è quello di risolvere il problema della degradazione dell'accuratezza della rete: con l'aumentare della profondità del modello, l'accuratezza tende a saturare e poi a degradare rapidamente. In diversi studi è stato dimostrato che l'aggiunta di strati all'architettura di una CNN, raggiungendo elevate profondità, causa un aumento dell'errore di training [47][48][49]. Gli sviluppatori di ResNet hanno ricondotto questo problema all'ipotesi che le mappature dirette siano difficili da allenare. In alternativa hanno ritenuto fosse più semplice apprendere la differenza tra un input  $x$  e la mappatura desiderata  $H(x)$ , ovvero il *residuale*. Un esempio del blocco residuale presente nell'architettura della ResNet è mostrato in figura 4.8.

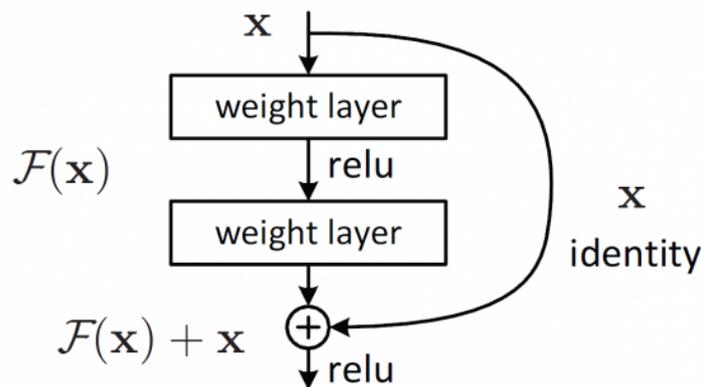


Figura 4.8: Blocco residuale [47].

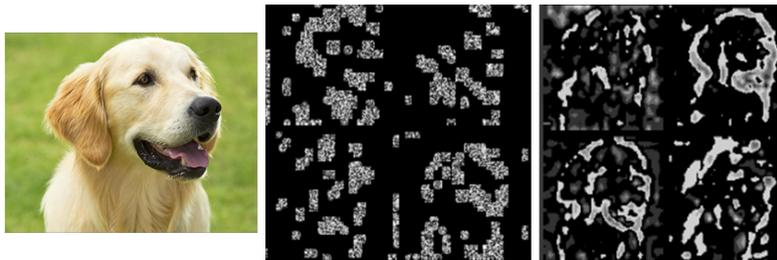
Con  $x$  si indica l'input del blocco, con  $F(x)$  le trasformazioni non lineari (convoluzione più funzione di attivazione) effettuate su  $x$ . Nello specifico vale  $F(x) = H(x) - x$  dove, come già accennato,  $H(x)$  corrisponde alla mappa di attivazione desiderata. Questa differenza corrisponde proprio al **residuo** che viene appreso dal singolo blocco. Per ricostruire  $H(x)$  la *skip connection* prevede la seguente operazione di somma:

$$H(x) = F(x) + x \quad (4.6)$$

Per eseguirla è necessario che l'input  $x$  e  $F(x)$  abbiano le stesse dimensioni. In caso contrario, la differenza verrà compensata tramite un'operazione di *padding*.

Successivamente all'output così ottenuto viene applicata la funzione di attivazione ReLU.

Un blocco di questo tipo permette di apprendere nuove informazioni sull'immagine senza perdere però quelle dell'input  $x$ , ricavate dagli strati precedenti [47]. In una rete convoluzionale i layer più superficiali permettono di estrarre *features spaziali* e quelli più profondi di estrarre *features semantiche*. Senza preservare le informazioni ottenute dagli strati precedenti sarà difficile portare fino al fondo della rete le caratteristiche spaziali estratte, perdendo informazioni importanti. Il blocco residuale consente di evitare questa perdita, come conferma il confronto (figura 4.9) delle mappe di attivazione ricavate, allo stesso livello di profondità, dalle reti VGG-19 e ResNet50 [50]. Inoltre l'utilizzo delle *skip connections* o blocchi residuali ha consentito ai creatori della rete ResNet di realizzare architetture molto profonde, fino a 152 strati (8 volte la profondità della VGG-19), mantenendo però una complessità inferiore rispetto alla VGG. Ad esempio, la ResNet a 152 strati produce 58 milioni di parametri da addestrare, meno della metà dei parametri previsti dalla VGG (138 milioni) [47].



**Figura 4.9:** Confronto delle mappe di attivazione estratte dalla VGG-19 e dalla ResNet50. Da sinistra a destra: immagine originale, feature map della VGG-19, feature map della ResNet50. [50].

Anche nel caso della CNN di tipo ResNet esistono diverse versioni che differiscono tra loro in base al numero di strati presenti. La rete utilizzata nel lavoro di tesi qui presentato è la ResNet50, avente 50 layer. La sua architettura è sintetizzata nella tabella 4.1. La rete prende in ingresso immagini di dimensioni  $224 \times 224 \times 3$ , che attraverseranno 5 blocchi convoluzionali. Il primo prevede un solo layer convoluzionale con 64 filtri di dimensioni  $7 \times 7$  e stride pari a 2, al quale segue lo strato di *max pooling* con kernel  $3 \times 3$  e stride 2. I successivi 4 blocchi convoluzionali sono costituiti da più blocchi residuali impilati, raggiungendo rispettivamente un totale di 9, 12, 18 e 9 layer. Nello specifico ogni blocco residuale prevede 3 strati di tipo convolutivo: il primo e il terzo con un kernel  $1 \times 1$ , e il secondo con un kernel  $3 \times 3$ . La quantità di filtri utilizzati per ogni strato è ben indicata nella tabella 4.1. Ultimato l'ultimo blocco convoluzionale (Conv5), segue uno strato di *average pooling* e un layer completamente connesso con 1000 neuroni. Infine, l'output della

rete sarà una *softmax*, ovvero una funzione di attivazione che converte un vettore di input di  $N$  elementi ( $N =$  numero delle classi) in un vettore probabilità delle stesse dimensioni. Nello specifico ogni elemento del vettore di output corrisponde alla probabilità che l'immagine appartenga ad una delle  $N$  classi. Dall'architettura, così descritta, si ottengono quindi 50 layer e 23 milioni di parametri da addestrare [47].

Layer Name	50-Layer	Output Size
Conv1	7x7, 64, stride 2 3x3 max pool, stride 2	112x112
Conv2	$\begin{bmatrix} 1x1, 64 \\ 3x3, 64 \\ 1x1, 256 \end{bmatrix} \times 3$	56x56
Conv3	$\begin{bmatrix} 1x1, 128 \\ 3x3, 128 \\ 1x1, 512 \end{bmatrix} \times 4$	28x28
Conv4	$\begin{bmatrix} 1x1, 256 \\ 3x3, 256 \\ 1x1, 1024 \end{bmatrix} \times 6$	14x14
Conv5	$\begin{bmatrix} 1x1, 512 \\ 3x3, 512 \\ 1x1, 2048 \end{bmatrix} \times 3$	7x7
	Average pool 1000 fc Softmax	1x1

**Tabella 4.1:** Architettura ResNet50.

### 4.1.3 EfficientNet

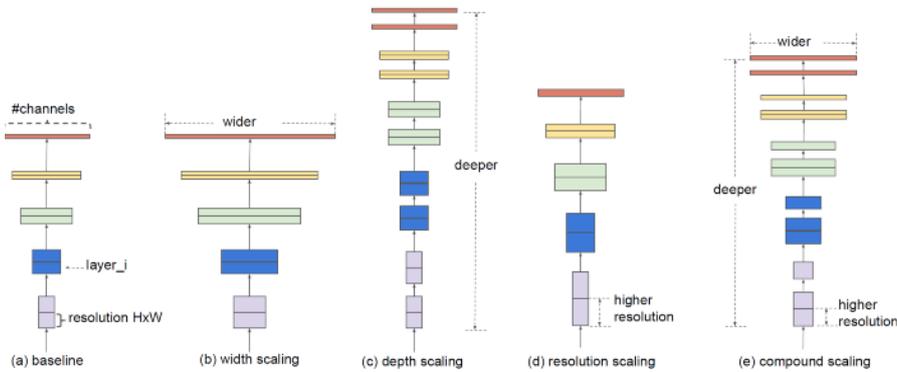
Le reti convoluzionali EfficientNet sono state proposte nel 2019 da M. Tan et al. [51], partendo dallo sviluppo di una rete di base, EfficientNet-B0, fino alla creazione di una famiglia di reti (dalla EfficientNet-B1 alla EfficientNet-B7) molto efficienti, come suggerisce il nome stesso. Queste sono state realizzate applicando alla rete di base diversi ridimensionamenti, tutti basati sull'innovativo metodo proposto dagli stessi sviluppatori di EfficientNet. L'idea iniziale dello studio riguardava proprio una revisione del ridimensionamento delle CNN con l'obiettivo di migliorarne le prestazioni, proponendo una nuova strategia.

Il ridimensionamento di una rete può dipendere da tre grandezze:

- **Profondità:** corrispondente al numero di strati che costituiscono la rete.
- **Larghezza:** indicata dal numero di canali presenti in ogni layer convoluzionale, ovvero il numero di filtri.

- **Risoluzione:** legata alla dimensione delle immagini di input, nello specifico altezza e larghezza.

In tutti gli studi precedenti il ridimensionamento della rete è legato alla modifica di una sola delle 3 grandezze elencate: viene aumentata solo la profondità, oppure solo la larghezza, o solo la risoluzione. Tuttavia, se le dimensione dell'immagine di input sono maggiori (aumento della risoluzione), c'è bisogno di più livelli e di più canali per poter estrarre anche i *pattern* a grana più fine. Il metodo suggerito da M. Tan et al. [51] si propone proprio di ridimensionare le reti modificando uniformemente profondità, larghezza e risoluzione, e non una sola di queste. Questa tecnica prende il nome di **compound scaling** (ridimensionamento composto).



**Figura 4.10:** Ridimensionamento del modello. (a) rete di base generica, (b)-(d) ridimensionamento di una sola delle grandezze, (e) ridimensionamento composto [51].

Il ridimensionamento composto prevede l'uso di un **compound coefficient**,  $\Phi$ . Tramite  $\Phi$  vengono definiti i 3 coefficienti  $d$ ,  $w$ ,  $r$ , necessari per modificare rispettivamente profondità, larghezza e risoluzione della rete:

$$\begin{aligned} d &= \alpha^\Phi \\ w &= \beta^\Phi \\ r &= \gamma^\Phi \end{aligned} \quad (4.7)$$

dove  $\alpha$ ,  $\beta$ ,  $\gamma$  sono costanti che devono rispettare le seguenti condizioni:

$$\begin{aligned} \alpha \cdot \beta^2 \cdot \gamma^2 &\approx 2 \\ \alpha \geq 1, \beta \geq 1, \gamma \geq 1 \end{aligned} \quad (4.8)$$

Gli autori hanno testato il *compound scaling*, così definito, sui modelli ResNet ottenendo ottimi miglioramenti. In generale, hanno osservato che può essere utilizzato su qualsiasi architettura CNN, così hanno sviluppato un nuovo modello base, EfficientNet-B0. La cui architettura è semplificata nella tabella 4.2.

Stage i	Operazione $F_i$	Risoluzione $H_i \times W_i$	Canali $C_i$	Layer $L_i$
1	Conv3x3	224x224	32	1
2	MBCConv1, k3x3	112x112	16	1
3	MBCConv6, k3x3	112x112	24	2
4	MBCConv6, k5x5	56x56	40	2
5	MBCConv6, k3x3	28x28	80	3
6	MBCConv6, k5x5	14x14	112	3
7	MBCConv6, k5x5	14x14	192	4
8	MBCConv6, k3x3	7x7	320	1
9	Conv1x1, Pooling, FC	7x7	1280	1

**Tabella 4.2:** Architettura EfficientNet-B0 [51].

Il blocco principale è il **MBCConv**, ovvero un blocco residuale invertito a collo di bottiglia, introdotto per la prima volta nel 2019 da M. Sandler et al. [52]. MBCConv è costituito da tre successive convoluzioni. Queste sono rappresentate, insieme alle relative mappe di input e output, con uno schema esplicativo in figura 4.11. Viene preso in input un tensore avente pochi canali e viene applicata la prima convoluzione, nello specifico una convoluzione puntuale 1x1. Lo scopo di questo primo strato è quello di espandere le dimensioni dell’input in termini di profondità, come è possibile notare osservando la seconda mappa di attivazione in figura 4.11. A questa viene applicato un livello convoluzionale 3x3, avente un numero di canali pari a quelli della mappa di input. Per finire viene eseguita l’ultima operazione di convoluzione, nuovamente tramite un layer puntuale 1x1. Quest’ultimo livello convolutivo riporta i dati in un tensore di output avente un numero di canali ridotto (ultima mappa della figura 4.11). Gli ultimi due blocchi convolutivi descritti rappresentano una **convoluzione separabile in profondità**. Questa divide un tradizionale livello convolutivo 3x3 in due layer convoluzionali diversi, con l’obiettivo di ridurre il numero complessivo di parametri. Quindi, riassumendo, l’input e l’output del blocco MBCConv sono tensori a bassa dimensione (pochi canali), invece gli strati all’interno del blocco vengono applicati su tensori ad alta dimensione. Questo consente di ridurre notevolmente i costi computazionali della rete [53].

Al blocco appena illustrato viene aggiunto un blocco di **squeeze and excitation**. Questo consente di assegnare a ciascun canale di una mappa di attivazione un peso diverso, che potrà essere allenato dalla rete insieme agli altri parametri.

Il modello base EfficientNet-B0, così strutturato, è stato ridimensionato 7 volte per ottenere le 7 reti della famiglia EfficientNet. Per il ridimensionamento sono stati fissati i valori  $\alpha$ ,  $\beta$ ,  $\gamma$  ed è stata applicata l’equazione 4.7 modificando di volta in volta il valore di  $\Phi$ . Nel lavoro di tesi presentato è stata utilizzata la rete EfficientNet-B4, che presenta 19 milioni di parametri da addestrare.

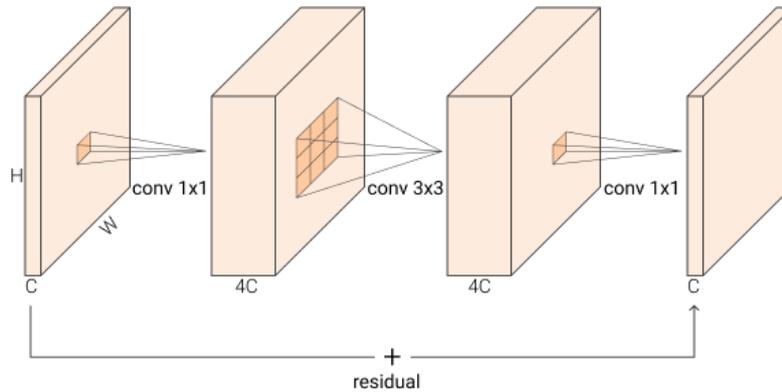


Figura 4.11: Blocco residuale invertito a collo di bottiglia [54].

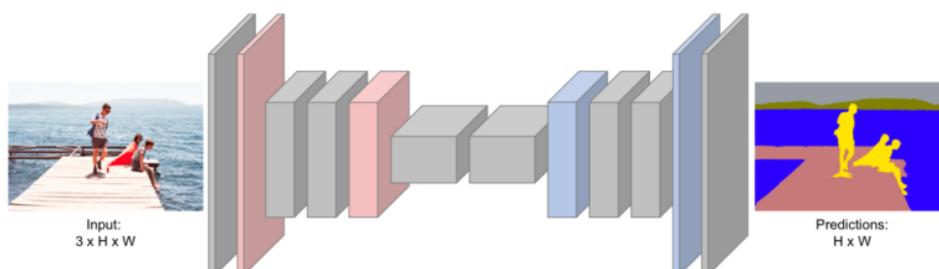
## 4.2 Segmentazione Semantica

La **segmentazione semantica**, come già accennato, è uno dei *task* che la Computer Vision si propone di risolvere. L'obiettivo è quello di assegnare ad ogni pixel dell'immagine di input la classe corrispondente. Viene, quindi, effettuata una previsione per ogni pixel dell'immagine. L'output desiderato sarà un'immagine ad alta risoluzione, normalmente avente le stesse dimensioni dell'immagine di input, in cui ogni pixel è associato a una specifica classe. Per ottenere questo vengono utilizzate le *Fully Convolutional Networks* (FCN). Queste prevedono due grandi percorsi [55]:

1. **Encoder.** Chiamato anche *contraction path*, è la sezione in cui viene effettuato **down-sampling**. La sua struttura prevede il susseguirsi di strati convoluzioni e di *max-pooling*, ovvero corrisponde all'architettura di una CNN. Come encoder, infatti, è possibile utilizzare un modello CNN pre-addestrato. In questa fase le dimensioni dell'immagine diminuiscono progressivamente e aumenta la profondità, con l'obiettivo di comprendere "cosa" è presente nell'immagine. Perdendo, però, qualunque informazione di tipo spaziale.
2. **Decoder.** Questo è il percorso di *decodifica* in cui si verifica un'espansione delle mappe di attivazione, **up-sampling**. Preso come input l'output dell'encoder, ne aumenta gradualmente le dimensioni fino a raggiungere quelle dell'immagine di input della rete. In questo modo è possibile estrarre le informazioni spaziali necessarie. Una volta unite alle informazioni ricavate dall'encoder, la rete sarà in grado di comprendere *quali* oggetti sono presenti nell'immagine e *dove*.

Il processo di *up-sampling* viene effettuato implementando un'operazione di *Convolutione Trasposta*. Operazione inversa a quella di convoluzione che, diversamente da quest'ultima, permette di ampliare le dimensioni di input. Nello specifico,

supponiamo di avere un input e un kernel, rispettivamente di dimensioni  $n_h \times n_w$  e  $k_h \times k_w$ . Supponiamo anche che tra questi venga eseguita l'operazione di convoluzione trasposta, si otterrà un output di dimensioni  $(n_h + k_h - 1) \times (n_w + k_w - 1)$  [56].



**Figura 4.12:** Generica struttura di una rete per segmentazione semantica [57].

Tra le reti per segmentazione semantica disponibili, ne spiccano alcune nel campo dell'*imaging* medico. Di queste ne viene fatto un focus nei paragrafi successivi.

### 4.2.1 U-Net

La rete U-Net è una Fully Convolution Network (FCN) proposta nel 2015 da O. Ronneberger et al. [58]. Nasce proprio per *tasks* di segmentazione semantica. Nello specifico è in grado di processare e segmentare immagini biomediche, anche avendo a disposizione un dataset di input relativamente piccolo. In qualità di rete FCN, la U-Net presenta la struttura descritta nel paragrafo precedente. È quindi presente una prima fase di *down-sampling*, seguita da *up-sampling*. In questo caso il percorso di *espansione* è stato realizzato simmetrico al percorso di *contrazione*. Questo fa sì che l'architettura della rete sia a forma di *U*, da cui il nome U-Net.

Una rappresentazione grafica dell'architettura è riportata in figura 4.13. Per una sua maggiore comprensione, si noti che ogni box blu rappresenta una mappa di attivazione a più canali, le cui dimensioni sono riportate intorno al blocco. Il numero di canali si trova al di sopra, la larghezza e l'altezza sono indicate in basso a sinistra. I box bianchi rappresentano le mappe di attivazione copiate e, infine, le frecce sono specifiche per ogni tipo di operazione effettuata. Il ramo di sinistra corrisponde al percorso di *contrazione* e segue la classica architettura di una CNN. Si ripete più volte un blocco, così costituito: 2 layer convoluzionali 3x3, ognuno seguito da uno strato di attivazione ReLU, e un livello di *max-pooling* 2x2 con stride pari a 2. Ad ogni step di *down-sampling* i canali delle mappe di attivazione vengono raddoppiati. Il ramo di destra rappresenta, invece, il percorso di *espansione*. Anche in questo caso è prevista la ripetizione di un blocco comprendente operazioni diverse: *up-convoluzione* 2x2, che dimezza il numero di



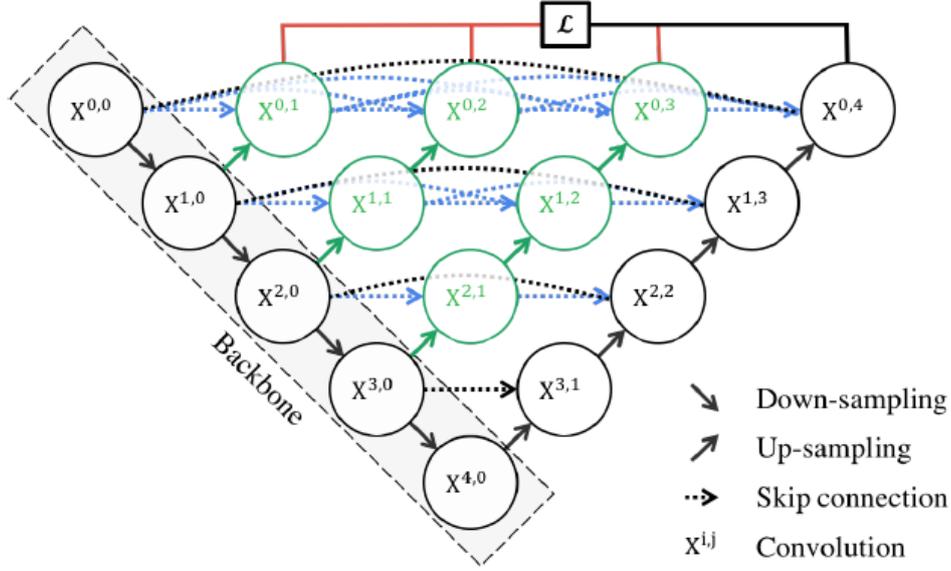
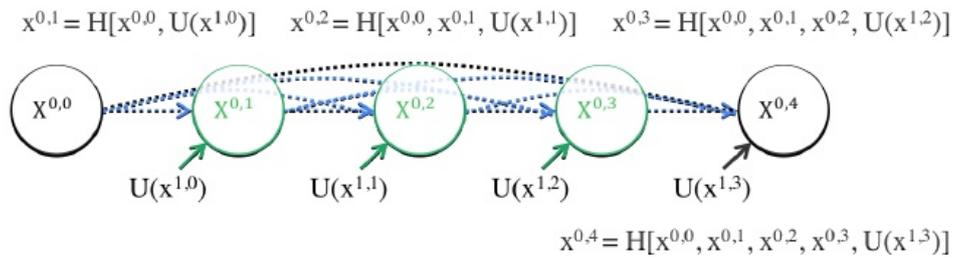


Figura 4.14: Architettura UNet++ [59].

In figura 4.14 è possibile osservare come queste dense e nidificate *skip connections* sono state strutturate. Sono indicati in nero i blocchi U-Net originali: la sotto-rete di encoder, indicata come **backbone**, seguita dalla sotto-rete di decoder. In blu e in verde, invece, sono rappresentati i nuovi *percorsi di salto* (o *skip pathway*), che collegano le corrispondenti feature map dei due rami. Questi sono costituiti da blocchi convoluzionali densi, il cui numero di strati convoluzionali dipende dal livello della piramide. Ad esempio, il collegamento tra i nodi  $X^{0,0}$  e  $X^{0,4}$  prevede tre livelli convoluzionali. Ognuno di questi è preceduto da una concatenazione che fonde l'output del precedente layer convoluzionale dello stesso blocco denso con il corrispondente output del blocco denso sottostante, sul quale è stato eseguito up-sampling. È possibile formulare gli *skip pathway* come segue:

$$x^{i,j} = \begin{cases} H(x^{i-1,j}), & j = 0 \\ H\left(\left[\left[x^{i,k}\right]_{k=0}^{j-1}, U(x^{i+1,j-1})\right]\right), & j > 0 \end{cases} \quad (4.9)$$

$x^{i,j}$  corrisponde all'output del generico nodo  $X^{i,j}$ , dove  $i$  indica il livello di down-sampling lungo l'encoder e  $j$  lo strato convoluzionale lungo il percorso di salto stesso. La funzione  $H(\cdot)$  indica l'operazione di convoluzione seguita dalla funzione di attivazione,  $U(\cdot)$  corrisponde al livello di up-sampling e infine  $[\ ]$  indica l'operazione di concatenazione. In generale un nodo al livello  $j$ -esimo riceve  $j + 1$  input:  $j$  input sono gli output dei precedenti nodi dello stesso *skip pathway* e l'ultimo input è l'output del percorso di salto sottostante, sul quale è stato eseguito up-sampling. In figura 4.15 è mostrato nel dettaglio il primo *skip pathway*.



**Figura 4.15:** *Skip pathway* che collega i nodi  $X^{0,0}$  e  $X^{0,4}$  [59].

Infine i collegamenti evidenziati in rosso in figura 4.14 rappresentano l'uso della **deep supervision**. L'introduzione della deep supervision permette alla rete UNet++ di operare in due modalità diverse:

1. **Accurate Mode.** L'output finale è dato dalla media degli output derivanti da tutti i rami di segmentazione.
2. **Fast Mode.** La segmentazione finale può essere selezionata da uno dei rami di segmentazione. Ovvero è possibile selezionare le feature map ad alta risoluzione  $\{x^{0,j}, j \in \{1,2,3,4\}\}$ .

In figura 4.16 sono rappresentate le diverse architetture derivanti dalla scelta di uno dei rami di segmentazione. Riassumendo la rete UNet++ differisce dalla rete U-Net in 3 modi: presenza di layer convoluzionali nei percorsi di salto, presenza di dense *skip connections* e presenza di deep supervision.

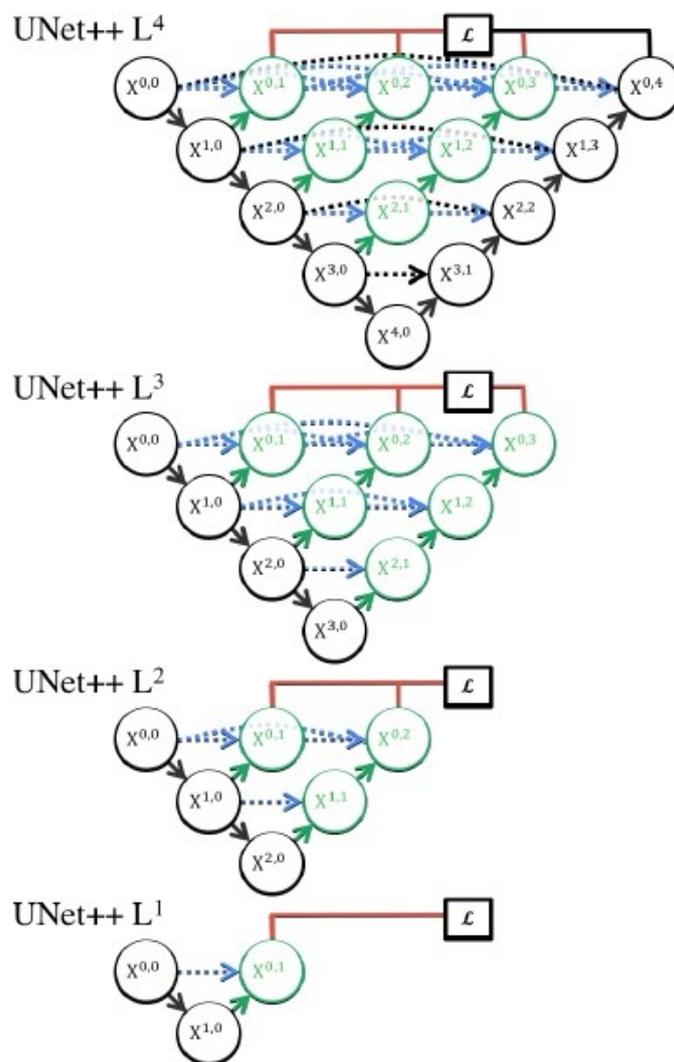


Figura 4.16: Deep supervision [59].

### 4.2.3 MA-Net

MA-Net (*Multi-scale Attention Net*) è stata sviluppata nel 2020 da T. Fan et al. [60]. Anche in questo caso, la rete è stata pensata per applicazioni nel campo della segmentazione di immagini mediche. Viene, quindi, riproposta un'architettura di tipo encoder-decoder, di cui è rappresentato uno schema in figura 4.17.

La struttura presenta 3 blocchi principali. Il primo è una rivisitazione del blocco residuale della rete ResNet: il **Res-block**, costituito da 3 blocchi convoluzioni 3x3 e

da una connessione residuale. Gli altri due sono due blocchi basati sul meccanismo *self-attention*<sup>1</sup> e progettati dagli stessi autori della MA-Net.

- **Position-wise Attention Block (PAB)**. Consente di ottenere la dipendenza spaziale tra i pixel nelle mappe di attivazione.
- **Multi-scale Fusion Attention Block (MFAB)**. Viene utilizzato per ottenere la dipendenza delle mappe di attivazione dai canali, combinando le feature map di alto e basso livello. Le feature map di alto livello contengono le informazioni semantiche, quelle di basso livello le informazioni legate alla forma. Il blocco MFAB seleziona automaticamente le informazioni più importanti per la segmentazione, promuovendo le mappe di attivazione che le contengono e sopprimendo quelle che risultano essere meno utili.

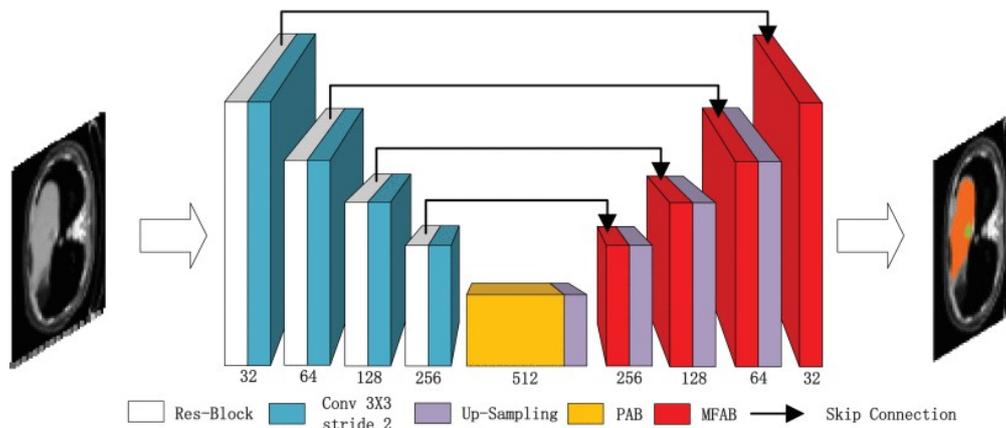


Figura 4.17: Architettura MA-Net [60].

### 4.3 Overfitting

L'*overfitting* è un problema molto comune dell'allenamento delle Reti Neurali. Si verifica quando il modello si adatta eccessivamente alle caratteristiche dei dati su cui viene allenato, non riuscendo più a generalizzare. Fornendo nuovi dati alla rete, questa non sarebbe in grado di fare previsioni attendibili. È possibile attuare alcune tecniche per ridurre e prevenire la comparsa di *overfitting* [61].

<sup>1</sup>Il meccanismo *self-attention* è in grado di simulare il sistema visivo umano, concentrandosi su una specifica area di interesse. Lo scopo è selezionare le informazioni importanti per il problema in esame, estraendole da una grande varietà di informazioni.

- *Data Augmentation*. Un dataset molto ampio è in grado di generalizzare meglio un modello, fornendo una maggiore varietà di dati in fase di allenamento. Nel caso in cui le dimensioni del dataset a disposizione siano più contenute, è possibile aumentare il *training set* deformando le immagini di input. Ad esempio, possono essere ridimensionate, traslate, ruotate o, ancora, corrotte dall'aggiunta di rumore.
- *Regolarizzazione*. Consiste nell'aggiunta di un termine di penalità alla funzione di perdita, usata per aggiornare i pesi della rete. La più diffusa è la regolarizzazione L2 che permette ai pesi di decadere verso lo zero, senza però raggiungerlo.
- *Dropout*. É una forma di regolarizzazione che prevede lo spegnimento casuale di alcuni neuroni. Questo permette di evitare che la rete sia più sensibile al peso di alcuni neuroni a discapito di altri.
- *Arresto anticipato*. Questo approccio risulta necessario quando il modello si adatta eccessivamente ai dati di *training*. Normalmente, quando si verificano un certo numero di iterazioni (parametro settato), durante le quali non si hanno miglioramenti sul *validation set*, l'allenamento viene interrotto anticipatamente.



## Capitolo 5

# Segmentazione delle strutture del nervo ottico e calcolo di OND e ONSD

In questo capitolo viene presentato il dataset utilizzato per lo sviluppo e la convalida dell'algoritmo. Di quest'ultimo ne viene descritta la pipeline, che prevede l'uso di reti per segmentazione semantica. Le modalità e i parametri di allenamento delle reti verranno approfondite in questo capitolo. Si descrivono anche gli step necessari al calcolo del diametro del nervo ottico (OND) e dalla guaina (ONSD). Infine si discutono le metriche di validazione e statistiche utilizzate per quantificare la corretta segmentazione delle strutture e la corretta misurazione dei diametri.

### 5.1 Dataset

Per la creazione del dataset le immagini di ecografia transorbitale (TOS) sono state rese disponibili dai Dipartimenti di Neurologia di tre strutture ospedaliere differenti: l'Ospedale Universitario del Saarland di Homburg (Germania), l'Azienda Ospedaliero-Universitaria Città della Salute e della Scienza di Torino (Italia) e l'Ospedale San Giovanni Bosco di Torino (Italia). Tutte le immagini sono state acquisite in modalità B-Mode rispettando il protocollo descritto nel paragrafo 2.2. Provengono da **110 soggetti** diversi, sia sani che patologici, dei quali non sono stati forniti dati inerenti a genere, età e stato di salute. Le immagini a disposizione sono state acquisite da 5 macchinari ecografici diversi; si riportano in tabella 5.1 gli ecografi utilizzati e le corrispondenti percentuali di immagini. Poiché le immagini sono state acquisite da macchinari diversi non hanno tutte le stesse dimensioni. Sono state, però, fornite tutte con lo stesso formato *bitmap* (.bmp).

Macchinario	N. Immagini	Percentuale su totale
1 - MyLab, Esaote - Homburg (Germania)	207	45%
2 - MyLab, Esaote - Torino (Italia)	113	24%
3 - Aplio300, Toshiba	93	20%
4 - Vivid 7, GE Healthcare	51	11%

**Tabella 5.1:** Percentuali immagini acquisite per ogni macchinario.

Le immagini sono state selezionate in modo da averne 6 per ogni soggetto, quando disponibili, di cui 3 dell'occhio sinistro e 3 dell'occhio destro. Raggiungendo, così, un totale di **464 immagini**.

Viste le diverse fonti di provenienza delle immagini, queste sono state tutte rinominate in modo da rendere uniforme la notazione. In questo modo risulta più semplice il riconoscimento dell'immagine, nello specifico risulta immediata l'associazione con il macchinario, utilizzato in fase di acquisizione, e la suddivisione per soggetto. Il nome di ogni immagine rispetta la seguente notazione:

$$Mac\_x\_Sub\_yy\_Im\_zz$$

Dove  $x$  è un numero che va da 1 a 4 e indica il macchinario. Ad ogni ecografo, riportato in tabella 5.1, è stato associato un numero in modo che possa essere facilmente identificato. Dopo di che viene indicato il soggetto da cui proviene l'immagine tramite il numero  $yy$ . Essendo i soggetti totali 110,  $yy$  assume tutti i valori interi compresi tra 1 e 110. Infine con  $zz$  si indica il numero di immagine selezionata per un soggetto specifico. Ad esempio, si supponga di aver selezionato 3 immagini provenienti dal soggetto numero 2 e acquisite con l'ecografo *MyLab* dall'Ospedale Universitario del Saarland di Homburg (macchinario numero 1). Queste vengono rinominate come segue:  $Mac\_1\_Sub\_02\_Im\_1$ ,  $Mac\_1\_Sub\_02\_Im\_2$ ,  $Mac\_1\_Sub\_02\_Im\_3$ .

## 5.2 Architettura dell'algoritmo

La struttura generale dell'algoritmo proposto è rappresentata nella figura 5.1. Presa come input un'immagine dal dataset ne viene effettuato un *cropping*. Avendo ogni immagine dimensioni diverse, viene ricavata in modo automatico una patch, contenente il nervo ottico, di dimensioni 256x256. In questo modo, durante l'allenamento delle reti verranno fornite in ingresso immagini aventi tutte le stesse dimensioni. Ultimato il *cropping*, nelle immagini ottenute vengono segmentate manualmente le strutture del nervo ottico e create le corrispettive maschere binarie. Queste rappresentano il *Ground Truth* (GT), che verrà utilizzato in fase di allenamento delle reti e di validazione dei risultati ottenuti dall'algoritmo. Segue il *training*

di 3 reti per segmentazione semantica: U-Net, UNet++, MA-Net. Ognuna delle quali viene allenata utilizzando due differenti CNN come encoder: ResNet50 ed EfficientNet-B4. In totale, quindi, vengono allenati 6 diversi modelli. Dal confronto dei rispettivi output viene identificato il modello più performante. Questo viene allenato nuovamente eseguendo una ***K-fold Cross-Validation***. Dopo di ch , si esegue il *post-processing* sull'output della rete, ovvero sulle maschere binarie predette corrispondenti alla segmentazione della guaina del nervo ottico. Infine, partendo dalle maschere post-processate, si calcolano i diametri del nervo ottico (OND) e dalla guaina (ONSD).

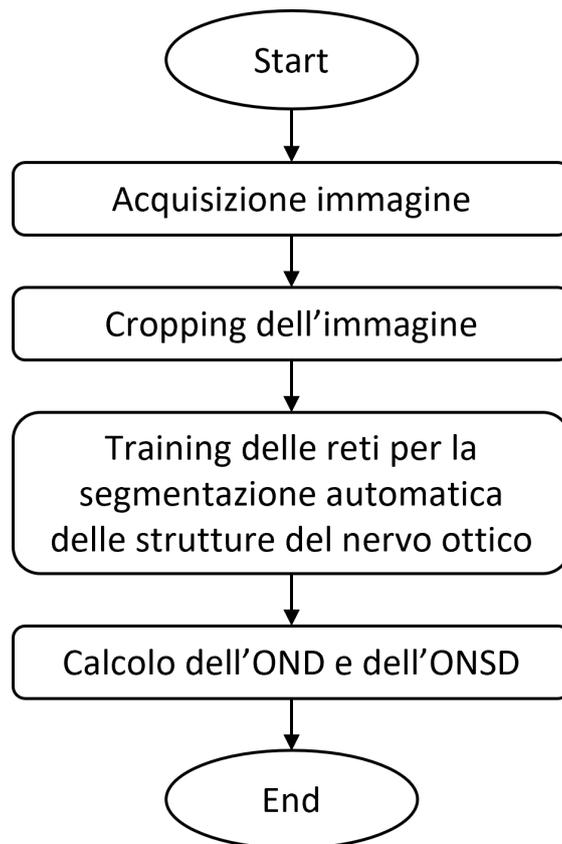


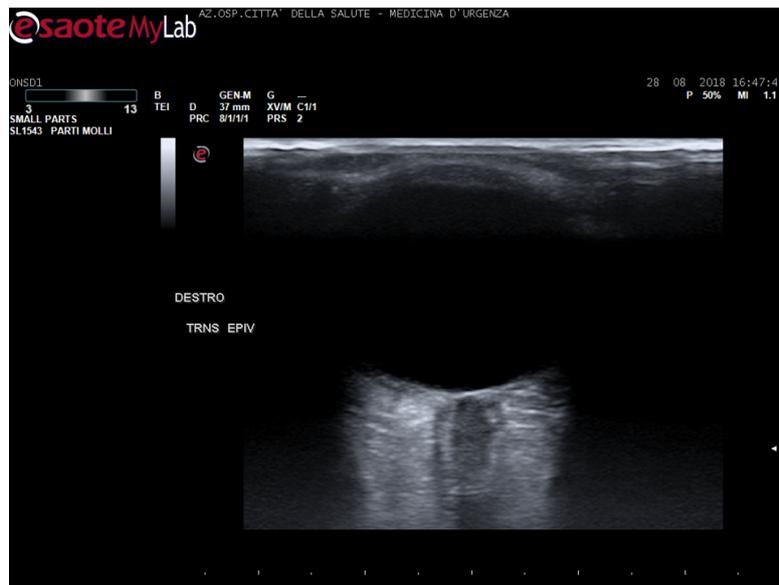
Figura 5.1: Schema generale dell'algoritmo proposto.

### 5.3 Cropping dell'immagine

L'allenamento delle reti per segmentazione semantica, cos  come quello delle Reti Neurali in generale, risulta essere ampiamente influenzato dalle informazioni provenienti dall'input. Diventa importante, quindi, fornire in ingresso immagini

che contengano solo le informazioni di interesse, omettendo il più possibile quelle superflue. Nel lavoro di tesi presentato le informazioni rilevanti si trovano, senza dubbio, nella zona in cui è rappresentato il nervo ottico. Per questo motivo si ritiene necessario eseguire un'operazione di *cropping* sulle immagini originali, estraendo per ognuna una patch, di dimensioni 256x256, che sia più o meno centrata sul nervo ottico. Inoltre, le immagini originali hanno dimensioni differenti in base al macchinario utilizzato per l'acquisizione. In questo modo si ottengono, invece, immagini aventi tutte le stesse dimensioni pari a 256x256. Queste verranno fornite come input alle reti per segmentazione semantica.

Per prima cosa viene calcolato il valore del **Fattore di Calibrazione (CF)**, che indica quanti *mm* reali sono rappresentati in un singolo pixel dell'immagine (*mm/pixel*). Il valore di CF può essere ricavato direttamente dall'immagine: tutte le immagini selezionate presentano una scala spaziale laterale dalla quale è possibile estrarre le informazioni necessarie per il calcolo del fattore di calibrazione (figura 5.2).



**Figura 5.2:** Esempio immagine originale di input (da dataset - Mac\_2\_Sub\_39\_Im\_3).

La scala laterale ha delle tacche numeriche equidistanti, la cui distanza corrisponde a 10 mm reali. Misurandone la distanza in pixel è possibile ottenere il valore di CF applicando questa semplice formula:

$$CF = \frac{10}{dist\_pixel} \quad (5.1)$$

48

dove  $dist\_pixel$  indica proprio la distanza in pixel tra le tacche. I valori dei fattori di calibrazione, così ottenuti, vengono salvati su un file di testo in modo che possano essere utilizzati per il successivo calcolo dei diametri.

Il ritaglio viene effettuato sulla base dei valori di intensità dei pixel, ma anche del valore di CF. Infatti, il valore del fattore di calibrazione può fornire informazioni relative a un eventuale zoom presente nell'immagine. È stata identificata un'immagine di riferimento, considerata priva di zoom, avente  $CF = 0,1$ . Questo valore viene assunto come valore base di CF ( $CF_{base}$ ). Quindi tutte le immagini aventi Fattori di Calibrazione inferiori o superiori risulteranno più o meno zoomate rispetto all'immagine di riferimento. In figura 5.3 si riportano due immagini da dataset che esemplificano il concetto appena espresso. La prima ha un  $CF = 0,11$ , paragonabile al  $CF_{base}$ , la seconda un  $CF = 0,05$ . La differenza di zoom risulta ben visibile già a un primo sguardo.



**Figura 5.3:** Confronto di due immagini di input aventi valori di Fattore di Calibrazione differenti. Da sinistra verso destra:  $CF = 0,11$ ,  $CF = 0,05$ .

È possibile calcolare la percentuale di zoom come segue:

$$zoom = \frac{CF_{base} - CF}{CF_{base}} \cdot 100 \quad (5.2)$$

dove  $CF_{base}$  corrisponde al valore di riferimento pari a 0.1 e  $CF$  al valore del fattore di calibrazione caratteristico di una specifica immagine.

Viene poi effettuato un primo crop. L'entità del taglio orizzontale dipende dalle dimensioni dell'immagine originale: si taglia all'altezza delle righe corrispondenti al 17% e al 90% di quelle totali. Diversamente, il taglio verticale dipende dai valori di intensità dei pixel dell'immagine. Eseguendo una somma per colonne, viene ricercata la più larga fascia verticale complessivamente più luminosa. La ricerca si basa sul fatto che, come si può osservare in figura 5.2, la zona in cui è rappresentato il nervo ottico è quella contenente la maggior percentuale di pixel con i valori di intensità maggiori. Si supponga che la fascia verticale individuata

vada dalla colonna  $c_1$  alla colonna  $c_2$ , in corrispondenza di queste viene effettuato il taglio. In figura 5.4 (b) è possibile osservare il risultato del primo crop, così ottenuto.

Lo step successivo si basa sulle caratteristiche delle strutture rappresentate in un'immagine di ecografia transorbitale. In particolare sull'ipoecogenicità del bulbo oculare e, quindi, sulla presenza di una struttura circolare nera, posizionata proprio al di sopra del nervo ottico. L'individuazione del bulbo risulta, per cui, funzionale per l'individuazione del nervo. Per identificare e localizzare il bulbo oculare si cerca la circonferenza che meglio lo approssima. Questa fase prevede la ricerca di circonferenze a partire da una maschera binaria ottenuta tramite *thresholding*. Il valore soglia è posto uguale a 20 (i pixel assumono valori di intensità compresi tra 0 e 255). A video, quindi, risulterà nera la regione di spazio corrispondente al nervo ottico e bianca la zona circolare del bulbo oculare (figura 5.4 (c)). La ricerca delle circonferenze si basa sulla *Circular Hough Transform*<sup>1</sup> (CHT). Vengono riconosciute sulla maschera binaria le circonferenze aventi raggio compreso in un range definito. Si forniscono range diversi in base al valore del fattore di calibrazione dell'immagine. Se questa ha un valore di CF inferiore a una certa soglia (risulta zoomata), il raggio minimo e il raggio massimo saranno costruiti partendo da un valore scelto a priori e diminuito di una quantità proporzionale alla metà della percentuale di zoom (equazione 5.3). Diversamente se il valore di CF risulta sopra soglia, il raggio minimo e il raggio massimo vengono definiti indipendentemente dal fattore di calibrazione.

$$R_{min} = \begin{cases} 80 - \left(80 \cdot \frac{zoom}{100} \cdot 0.5\right) & CF < 0.07 \\ 80 & CF \geq 0.07 \end{cases} \quad (5.3)$$

$$R_{max} = \begin{cases} 150 - \left(150 \cdot \frac{zoom}{100} \cdot 0.5\right) & CF < 0.07 \\ 120 & CF \geq 0.07 \end{cases}$$

$R_{min}$  ed  $R_{max}$  indicano, rispettivamente, il raggio minimo e il raggio massimo e  $zoom$  è la percentuale di zoom dell'immagine calcolata secondo la formula 5.2.

Tutte le circonferenze individuate vengono sovrapposte alla maschera binaria e si identifica la circonferenza contenente il maggior numero di pixel con intensità pari a 1. Questa sarà l'oggetto circolare che meglio approssima il bulbo oculare. Si individua il suo punto più basso (figura 5.4 (d) in rosso), si assume che si trovi in corrispondenza del fondo del bulbo oculare e che, quindi, i pixel sottostanti rappresentino il nervo ottico. Questo punto costituisce il pixel da cui partirà la

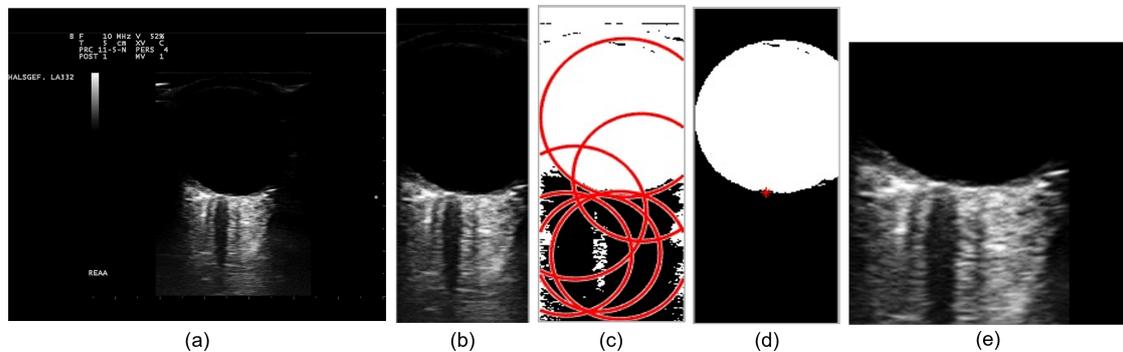
---

<sup>1</sup>la Trasformata di Hough è una tecnica di estrazione usata per l'elaborazione digitale delle immagini. Permette di riconoscere particolari configurazioni presenti nell'immagine, come segmenti o curve.

costruzione della patch 256x256, che si vuole estrarre. Indichiamo le coordinate di questo pixel  $(X_c, Y_c)$ . A questo punto viene effettuato un secondo crop sull'immagine originale. La regione di crop viene definita in modo che il vertice superiore sinistro abbia le seguenti coordinate:  $(X_c - 128, Y_c - 128)$ . Partendo dal pixel così definito si taglia orizzontalmente scendendo di 256 righe e verticalmente spostandosi verso destra di 256 colonne. La regione di taglio, così costruita, corrisponde proprio alla patch di dimensioni 256x256, obiettivo finale di tutta l'operazione di *cropping*.

Nel caso in cui l'immagine originale presenti uno zoom, effettuare il taglio in corrispondenza della regione appena definita, potrebbe escludere alcune informazioni importanti per il successivo allenamento delle reti. Se l'immagine è molto zoomata, tagliando al di sotto di sole 128 righe dal fondo del bulbo oculare, si rischia di troncare il nervo ottico e di non includerlo completamente all'interno della patch. Per questo motivo, nel caso in cui l'immagine abbia un valore di  $CF < 0.09$ , la regione di taglio viene definita come sopra e poi traslata verticalmente verso il basso. L'entità della traslazione è proporzionale al valore del fattore di calibrazione, in particolare la patch viene abbassata di un numero di righe pari al 70% della percentuale di *zoom* (equazione 5.2) dell'immagine. Un esempio del risultato finale del taglio dell'immagine è mostrato in figura 5.4 (e).

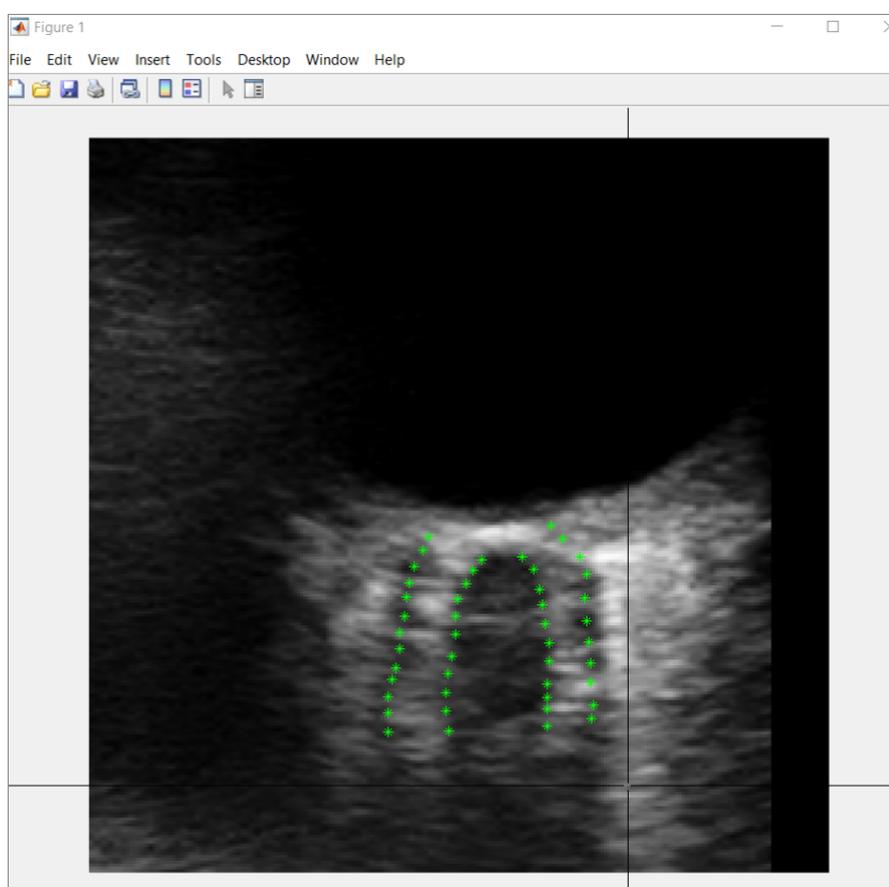
Concluso il *cropping*, la patch 256x256 viene salvata in formato *.png* mantenendo lo stesso nome dell'immagine originale da cui è stata estratta. Tutti gli step descritti sono stati eseguiti in ambiente MATLAB R2021a.



**Figura 5.4:** Operazione di cropping. (a) Immagine originale. (b) Risultato del primo taglio. (c) Ricerca delle circonferenze basate sulla CHT. (d) Individuazione della circonferenza che meglio approssima il bulbo oculare, in rosso il punto di partenza per la costruzione della patch. (e) Output finale del cropping, immagine 256x256.

## 5.4 Segmentazione manuale delle strutture del nervo ottico

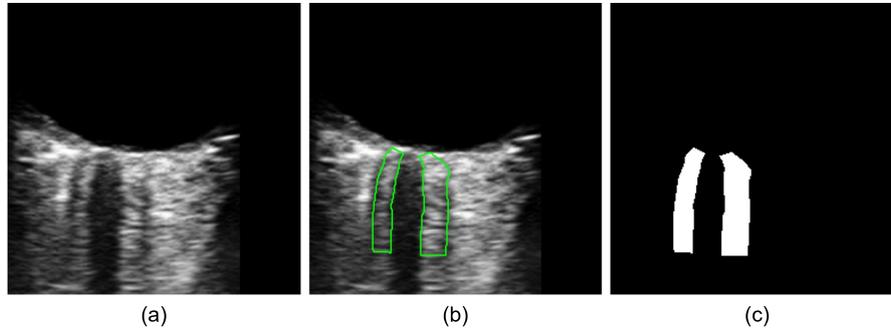
Ognuna delle immagini ritagliate nello step di *cropping* precedente viene segmentata manualmente. Si identificano le strutture del nervo ottico, individuando i bordi interni del nervo e i bordi esterni della guaina. Anche questa operazione viene effettuata utilizzando la piattaforma di calcolo MATLAB R2021a, all'interno della quale viene caricata l'immagine di dimensioni 256x256. Questa viene visualizzata a video insieme a un puntatore, in modo da consentire all'utente di selezionare manualmente i punti che costituiscono i bordi del nervo e della guaina (in verde in figura 5.5).



**Figura 5.5:** Visualizzazione immagine tramite MATLAB R2021a. In verde i punti selezionati manualmente dall'utente.

I punti inseriti vengono interpolati in modo da ricostruire i bordi delle strutture del nervo ottico. A partire da questi si crea la maschera binaria, nella quale

viene evidenziata la struttura corrispondente alla guaina del nervo ottico. Infine si procede con il salvataggio della maschera binaria. È possibile osservare un esempio di segmentazione manuale in figura 5.6.



**Figura 5.6:** (a) Immagine ecografica TOS originale. (b) Immagine segmentata manualmente. (c) Maschera binaria della guaina del nervo, ottenuta dai bordi dalla segmentazione manuale.

Le maschere, così ricavate, rappresentano il *Ground Truth*, che verrà utilizzato in fase di allenamento delle reti e di validazione dei risultati ottenuti dall’algoritmo. Ognuna delle 464 maschere binarie ottenute è stata validata da un operatore esperto.

## 5.5 Segmentazione automatica delle strutture del nervo ottico

La segmentazione automatica delle strutture del nervo ottico è stata eseguita tramite implementazione di modelli per segmentazione semantica. Per ottenere dei risultati il più affidabili possibile e individuare il modello più performante sul dataset a disposizione, sono state allenate 3 diverse reti: U-Net, UNet++, MA-Net. Tutte e 3 necessitano di un encoder, avendo un’architettura di tipo *encoder-decoder*. Per ognuna sono stati utilizzati due modelli CNN pre-addestrati: ResNet50 e EfficientNet-B4. In totale, quindi, sono stati eseguiti 6 allenamenti diversi, riassunti in tabella 5.2.

Ultimati i 6 allenamenti, gli output predetti dalla rete si confrontano tra loro, con lo scopo di individuare quale tra i 6 modelli sia quello con performance più elevate. Il modello identificato come più performante è stato sottoposto a un’ulteriore fase di training, utilizzando la tecnica della *k-fold cross-validation*.

Tutte le reti sono state implementate in Python utilizzando i modelli e i *backbones* (encoder) forniti dalla *repository Segmentation Models* di Pytorch [62]. Pytorch è una libreria del linguaggio Python contenente molte funzioni dedicate al Machine

Learning e al Deep Learning. L'allenamento è stato eseguito su **Kaggle**, una piattaforma online per competizioni di *data science* che consente di sviluppare e lanciare codici usufruendo delle GPU gratuite messe a disposizione.

Architettura	Encoder
U-Net	ResNet50
U-Net	EfficientNet-B4
UNet++	ResNet50
UNet++	EfficientNet-B4
MA-Net	ResNet50
MA-Net	EfficientNet-B4

**Tabella 5.2:** Modelli allenati.

### 5.5.1 Allenamento delle reti

Il dataset a disposizione per la segmentazione del nervo ottico conta 464 immagini di dimensioni pari a 256x256. Con lo scopo di ottenere modelli robusti con buone capacità di generalizzazione, il dataset è stato suddiviso in 3 set differenti: **Training Set**, **Validation Set** e **Test Set**. In questo modo la rete non viene allenata sull'intero campione di dati a disposizione, ma solo su quelli del *training set*. Vengono, poi, forniti nuovi dati per la validazione, ovvero il *validation set*. E infine il modello viene testato sul *test set*, ovvero su un ulteriore gruppo di dati mai visti prima dalla rete. La divisione è stata effettuata in modo che il set di training sia quello più numeroso, nello specifico l'80% dei dati è stato destinato al *training set* e il restante 20% è stato equamente diviso tra il *validation set* e il *test set*. Il dataset creato per l'allenamento delle reti è, quindi, così strutturato:

- **Training Set** (80%): 373 immagini.
- **Validation Set** (10%): 46 immagini.
- **Test Set** (10%): 45 immagini.

Le immagini sono state divise in modo che in ogni set tutti i macchinari di acquisizione fossero rappresentati, e in modo che le immagini appartenenti a uno stesso soggetto non venissero divise.

È stato eseguito un allenamento **supervisionato**, quindi in input si forniscono le immagini e le rispettive maschere binarie, che indicano la classe a cui ogni pixel appartiene:

- *classe 0*: pixel del *background*.

- *classe 1*: pixel appartenenti alla guaina del nervo ottico.

Le immagini (a) e (c) della figura 5.6 rappresentano, rispettivamente, un esempio di immagine e maschera binaria di input.

Sui dati di training è stata effettuata *Data Augmentation* applicando le seguenti trasformazioni:

- Capovolgimento orizzontale dell'immagine.
- Traslazione sia orizzontale che verticale pari al 10% delle dimensioni dell'immagine.
- Rotazione in range compreso tra  $[-10^\circ, +10^\circ]$ .
- Ridimensionamento dell'immagine in un range compreso tra  $[-10\%, +10\%]$ .
- Corruzione dell'immagine con rumore Gaussiano.
- *Sharpening* dell'immagine.
- *Blurring*, sfocatura dell'immagine ottenuta tramite il passaggio di un kernel.

L'applicazione di Data Augmentation consente di ridurre il problema, tipico delle applicazioni biomediche, di avere un dataset con una numerosità ridotta. In questo modo verrà allenato un algoritmo più robusto, in grado di apprendere da tali deformazioni senza la necessità che queste siano contenute nei dati a disposizione [58].

Nella tabella 5.3 sono riportati i parametri settati in fase di allenamento di ciascuno dei 6 modelli elencati in tabella 5.2.

Pesi Encoder	ImagNet
Algoritmo di ottimizzazione	Adam (learning rate = $1e^{-4}$ , $\beta_1 = 0.9$ , $\beta_2 = 0.999$ )
Batch size	8
Funzione di perdita	Dice Loss
Metrica	IoU score
Numero di epoche	40
learning rate diminuito	$1e^{-5}$
Epoca diminuzione del learning rate	25
Shuffle dei dati	Ogni epoca

**Tabella 5.3:** Parametri di input per l'allenamento delle reti.

Per una migliore e più rapida convergenza del modello sono stati utilizzati i pesi pre-allenati degli encoder (ResNet50 o EfficientNet-B4). Nello specifico sono stati importati i pesi ottenuti dall'allenamento delle CNN sul dataset ImageNet. È stato utilizzato come algoritmo di ottimizzazione **Adam**, metodo stocastico della discesa

del gradiente, basato sulla stima adattiva dei momenti del primo e del secondo ordine (media e varianza) [63]. Il *learning rate* viene adattato per ogni peso della rete, rendendo il modello più adattivo al dataset preso in esame. I pesi vengono aggiornati come segue [64]:

$$w_t = w_{t-1} - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \quad (5.4)$$

dove  $w_t$  rappresenta il peso dell'iterazione corrente,  $w_{t-1}$  il peso dell'iterazione precedente,  $\eta$  il *learning rate*,  $\epsilon$  un termine di smoothing aggiunto con fini di stabilità numerica,  $\hat{m}_t$  e  $\hat{v}_t$  gli stimatori, rispettivamente, del primo e del secondo momento. Questi dipendono da  $\beta_1$  e  $\beta_2$ , che controllano il decadimento esponenziale.

È stata impostata una dimensione di Batch pari a 8. La Batch-size indica il numero di campioni di training che vengono osservati durante un'iterazione di un'epoca. Un'epoca, infatti, consta di più iterazioni durante le quali i pesi vengono aggiornati in seguito al passaggio di un *mini-batch* nella rete. L'epoca, invece, prevede il passaggio di tutti i dati di training. Si è scelto un numero di epoche pari a 40. Per ognuna delle quali è previsto lo *shuffle* dei dati, in modo tale che se il numero di elementi del training non fosse divisibile per la dimensione del mini-batch scelta, non verrebbero scartate sempre le stesse immagini. Inoltre in corrispondenza della 25-esima epoca il valore del *learning rate* viene reinizializzato e posto pari a  $1e^{-5}$ .

La funzione di perdita scelta è la **Dice Loss**, definita dalla formula qui riportata [65]:

$$DL(y, \hat{p}) = 1 - \frac{2y\hat{p} + 1}{y + \hat{p} + 1} \quad (5.5)$$

dove  $y$  è l'output atteso, ovvero il Ground Truth, e  $\hat{p}$  è l'output predetto dalla rete.

Inoltre, i risultati ottenuti dal modello vengono di volta in volta confrontati con il Ground Truth preso in ingrosso, per monitorare le prestazioni della rete durante l'allenamento. La metrica utilizzata è il punteggio **IoU** (*Intersection over Union*), che fornisce un'indicazione in merito alla sovrapposizione tra la segmentazione predetta e quella attesa:

$$IoU = \frac{X \cap Y}{X \cup Y} \quad (5.6)$$

dove  $X$  indica la segmentazione del Ground Truth e  $Y$  quella predetta dalla rete. Ad ogni epoca si confronta il valore di IoU ottenuto con quelli delle epoche precedenti, con lo scopo di individuarne il punteggio massimo. Il modello ottenuto nell'epoca con IoU massimo viene salvato e utilizzato sul *test set*.

Gli output, ottenuti da ognuno dei 6 modelli allenati, sono stati salvati come maschere binarie e sono stati confrontati tra loro tramite l'impiego di metriche di validazione che verranno discusse nel corso del capitolo. Si vuole individuare la

rete più performante sul dataset fornito in ingresso. Il modello, così individuato, è il primo riportato in tabella 5.2: Architettura U-Net con ResNet50 come encoder.

### 5.5.2 K-Fold Cross-Validation sul modello più performante

La suddivisione del dataset effettuata per il precedente allenamento dei modelli si basa su una tecnica statica. La divisione, infatti, rimane rigida e costante. Questo potrebbe risultare problematico nel caso in cui, ad esempio, nel set di test ci siano dei casi limite oppure nel training set siano presenti dei dati dai quali non è possibile apprendere bene. È possibile ovviare a questo problema optando per una tecnica di suddivisione più dinamica, la ***K-Fold Cross-validation***, che consente di alternare i dati di input tra i 3 set [66].

Questo approccio prevede una divisione equa del dataset in  $k$  gruppi, noti come ***fold***. Uno di questi fold costituirà il *Test Set*, un altro il *Validation Set*, e i rimanenti  $k-1$  fold verranno utilizzati come *Training Set*. Questo processo viene ripetuto  $k$  volte, in modo che ogni gruppo venga utilizzato come set di test dalla rete. In questo modo l'allenamento del modello dovrà essere eseguito  $k$  volte. Di conseguenza si verificherà un aumento dei tempi computazionali [66].

Nel lavoro di tesi presentato è stato scelto  $k = 5$ , quindi l'intero set di dati (464 immagini 256x256) è stato suddiviso in 5 fold, ciascuno con un numerosità pari a circa il 20% della totalità. In tabella 5.4 è schematizzata la divisione del dataset nei 5 fold.

Fold1	Fold2	Fold3	Fold4	Fold5
94	93	92	92	93

**Tabella 5.4:** Suddivisione del dataset in 5 fold per eseguire Cross-Validation.

Anche in questo caso i dati sono stati suddivisi in modo tale che in ogni fold fossero presenti immagini acquisite da ognuno dei macchinari e in modo che immagini provenienti dallo stesso soggetto non venissero divise su più fold.

Come descritto sopra, iterativamente, uno dei 5 fold è stato destinato al Test Set, uno al validation Set e i 3 rimanenti al set di Training. Sono state ricavate in questo modo 5 diverse divisioni del dataset, ognuna delle quali comprensiva di:

- *Training Set*. Sempre costituito da 3 fold, quindi corrispondente al 60% del dataset.
- *Validation Set*. Costruito a partire da un solo fold e quindi corrispondente al 20% del dataset.
- *Test Set*. Comprensivo di un solo fold e quindi, anche questo, corrispondente al 20% del dataset.

In figura 5.7 è mostrato un schema esplicativo delle suddivisioni effettuate.

Iterazione 1	Fold1	Fold2	Fold3	Fold4	Fold5
Iterazione 2	Fold1	Fold2	Fold3	Fold4	Fold5
Iterazione 3	Fold1	Fold2	Fold3	Fold4	Fold5
Iterazione 4	Fold1	Fold2	Fold3	Fold4	Fold5
Iterazione 5	Fold1	Fold2	Fold3	Fold4	Fold5

**Figura 5.7:** suddivisione effettuata per Cross-validation con  $k=5$ . In arancione i fold di *Test*, in giallo i fold di *Validation* e in verde quelli di *Training*.

Ognuno dei 5 dataset per Cross-Validation, così definiti, è stato utilizzato come input per l'allenamento del modello per segmentazione semantica U-Net con encoder la CNN ResNet50. I parametri scelti per l'allenamento sono i medesimi riporti nella tabella 5.3. In questo modo è stato possibile testare la rete su tutte le 464 immagini del dataset e avere una stima più robusta della capacità di generalizzazione della rete.

## 5.6 Post-Processing sull'output della rete

Ultimato l'allenamento della rete per segmentazione semantica U-Net, utilizzando come encoder la CNN ResNet50, segue una fase di post-processing sull'output. L'output del modello corrisponde alle maschere binarie di tutte le 464 immagini del dataset. Avendo eseguito l'allenamento sfruttando la tecnica della cross-validation è stato possibile allenare iterativamente il modello 5 volte e testarlo su 5 diversi set di test. Unendo gli output di tutti i Test set, si ottiene un output complessivo comprendente le maschere predette di tutte le 464 immagini.

Il post-processing proposto viene implementato in ambiente MATLAB R2021a. Prevede innanzitutto di prendere in ingresso il risultato della segmentazione predetta dalla rete. Ovvero viene caricata una matrice di dimensioni 256x256, corrispondente a una maschera binaria. I pixel con valore di intensità pari a 1 sono quelli individuati dal modello come appartenenti alla guaina del nervo ottico. Diversamente, i pixel con valore di intensità pari a 0 indicano il background dell'immagine, ovvero tutte le informazioni non di interesse. In figura 5.8 è mostrato un esempio di output ottenuto dalla rete e la rispettiva sovrapposizione sull'immagine originale.



**Figura 5.8:** Da sinistra verso destra: Immagine originale, Maschera binaria predetta dalla rete, Sovrapposizione dell'output sull'immagine originale.

La prima operazione di post-processing implementata corrisponde a un riempimento di eventuali buchi presenti sulla maschera. Dopo di che viene eseguito un controllo sulle aree degli oggetti, individuati dalla segmentazione, per eliminare eventuali regioni in eccesso. Tramite *Regionprops* si analizza il numero degli oggetti e la loro estensione in termini di numerosità dei pixel contenuti in ciascun oggetto. Visto il significato anatomico della segmentazione ci si aspetta siano riconosciute due regioni diverse all'interno della maschera: una relativa alla porzione sinistra della guaina del nervo, e una relativa a quella destra. Nel caso in cui il conteggio degli oggetti restituisca un valore maggiore di 2, sarà evidente che in fase di segmentazione il modello ha classificato erroneamente alcune regioni come appartenenti alla guaina. In questo caso si procede con un controllo sui valori delle aree, con lo scopo di

individuare quelle da rimuovere. Nello specifico, se si individua un numero di oggetti uguale o superiore a 3 e se uno di questi presenta un'area inferiore al 60% dell'area massima individuata, viene rimosso. Il controllo viene effettuato in modo iterativo, verificando di volta in volta il numero di strutture rimaste. Il ciclo si interrompe quando ne rimangono due.

Sulle regioni rimaste viene applicata l'operazione più corposa del post-processing proposto, che riguarda la correzione di eventuali bordi mal segmentati dal modello. Sono previsti alcuni passaggi illustrati di seguito:

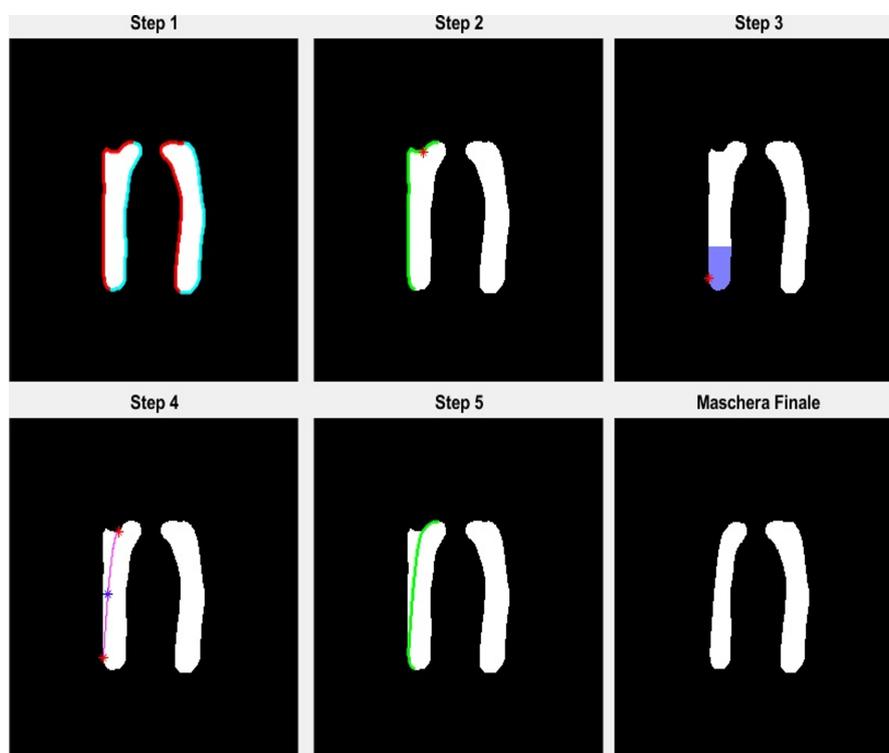
- **Step 1.** Si ricostruiscono i bordi di ciascuna delle regioni presenti nell'immagine binaria, separando i bordi sinistri da quelli destri. Nella figura 5.9 - Step1 è mostrato un esempio di come i bordi vengono identificati e separati: sono riportati in rosso i bordi sinistri e in azzurro quelli destri.
- **Step 2.** Si ricercano i bordi non conformi alla fisiologica anatomia del nervo ottico. Ogni bordo sarà costituito da un certo numero di punti (pixel), definiti da una coppia di coordinate (x,y). Ognuno dei bordi viene analizzato a partire dal pixel più alto. Ci si aspetta che tutti i punti successivi assumano progressivamente coordinate y sempre maggiori. Ovvero ci si aspetta che il bordo della guaina prosegua sviluppandosi verso il basso dell'immagine. Per questo motivo un bordo viene identificato come non conforme se presenta un gruppo di pixel successivi, nei quali si verifica una diminuzione della coordinata y (il bordo risale verso l'alto) o per i quali la coordinata y rimane costante (il bordo prosegue e assume andamento costante orizzontale). Dopo aver individuato il bordo (o i bordi) da correggere (figura 5.9 - Step2 in verde) si ricerca il pixel (figura 5.9 - Step2 in rosso) a partire dal quale l'andamento non risulta più fisiologico.
- **Step 3.** Si ricerca l'estremo basso laterale dell'oggetto di cui si vuole correggere il bordo (figura 5.9 - Step3 in rosso). Nel caso in cui il bordo in esame sia sinistro si ricerca l'estremo sinistro, diversamente si ricerca quello destro. Per avere la certezza che l'estremo individuato si trovi alla fine dell'oggetto, la ricerca viene effettuata solo nella porzione finale (figura 5.9 - Step3 in blu) corrispondente al 30% della lunghezza dell'oggetto stesso.
- **Step 4.** I punti individuati nei due step precedenti rappresentano gli estremi della porzione di bordo che verrà rimossa e corretta. Si vuole, quindi, ricostruire tale porzione del bordo approssimandola a una curva parabolica di equazione:

$$y(x) = ax^2 + bx + c \tag{5.7}$$

I parametri  $a$ ,  $b$  e  $c$  vengono individuati in modo che la tale curva passi proprio per i due pixel identificati negli step 2 e 3 (figura 5.9 - Step4 in rosso) e un

terzo punto aggiunto. Le sue coordinate vengono individuate calcolando il punto medio del segmento che ha per estremi i due punti segnati in rosso. Viene, poi, traslato in orizzontale di una quantità pari a 3 pixel, in modo che la curva ricostruita sia effettivamente parabolica e non rettilinea. Nella figura 5.9 - Step4 è possibile osservare il terzo punto in blu e la curva parabolica ricostruita in magenta.

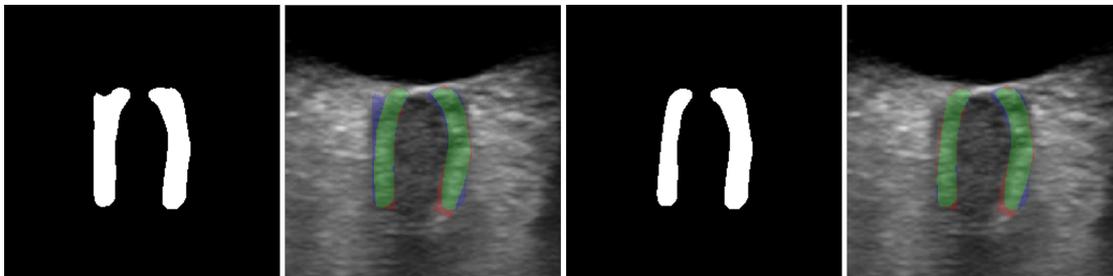
- **Step 5.** La curva parabolica ottenuta nello step precedente viene aggiunta al bordo di partenza, in sostituzione alla porzione originale da correggere. In questo modo si ottiene il nuovo bordo, evidenziato in verde nella figura 5.9 - Step5.



**Figura 5.9:** Correzione di bordi non conformi all'anatomia del nervo ottico. Step1. Individuazioni bordi sinistri (rosso) e bordi destri (azzurro); Step2. Individuazione bordo da correggere ed estremo superiore della porzione di bordo da eliminare; Step3. Individuazione estremo inferiore della porzione di bordo da eliminare; Step4. Costruzione di una curva parabolica (magenta) che corregga il bordo originale; Step5. Correzione del bordo (verde).

Nel caso in cui, in un'immagine siano presenti più bordi con le stesse problematiche, gli step dal 2 al 5 vengono ripetuti per ognuno. Si riporta in figura 5.10 il

risultato ottenuto da questa fase di post-processing, confrontato con la maschera di output pre post-processing e con la segmentazione manuale. È possibile notare come la maschera ottenuta dopo il post-processing sia molto più simile alla segmentazione manuale.



**Figura 5.10:** Risultato del post-processing. Da sinistra verso destra: Output della rete, Output della rete sovrapposta all'immagine originale e alla segmentazione manuale, Risultato del post-processing, Risultato del post-processing sovrapposto all'immagine originale e alla segmentazione manuale. In verde l'intersezione della segmentazione automatica e manuale (TP), in rosso la segmentazione manuale (FN), in blu la segmentazione automatica (FP).

## 5.7 Calcolo automatico di OND e ONSD

A partire dalle maschere binarie, ottenute in seguito alla segmentazione delle strutture del nervo, vengono calcolati in modo automatico i diametri del nervo ottico (OND) e della guaina (ONSD). La misurazione automatica viene effettuata sia a partire dalle maschere predette, dopo aver ultimato il post-processing, che a partire dalle maschere manuali. Da queste vengono ricavati i valori di OND e ONSD di riferimento, considerati *Ground Truth*.

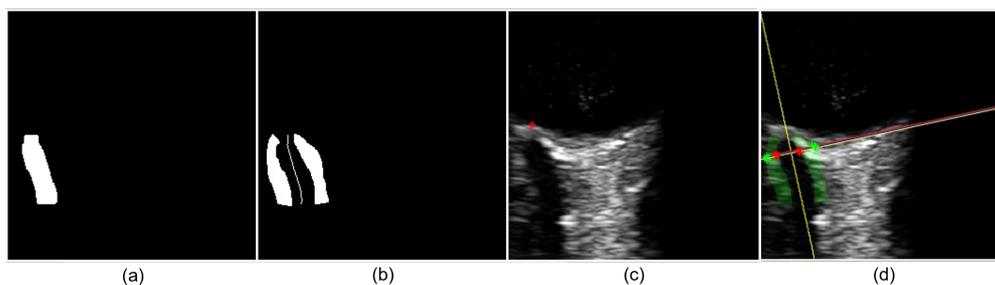
L'algoritmo per il calcolo dei diametri necessita in ingresso le maschere binarie 256x256 e le immagini originali, sempre di dimensioni 256x256. Rispettando le indicazioni cliniche, il calcolo dei diametri viene effettuato a una distanza pari a 3 mm dal fondo del bulbo oculare. Quindi, per ricavare i pixel situati a tale distanza è necessario acquisire in ingresso anche i valori dei Fattori di Calibrazione misurati precedentemente per ogni immagine. In modo da poter convertire le distanza in pixel in distanze in mm. Definiti gli input si descrivono i 4 macro-step in cui è possibile suddividere l'algoritmo.

1. **Ricostruzione nervo ottico.** Si applica *Regionprops* alla maschera binaria di input per poter analizzare i due oggetti presenti. Ovvero le due regioni corrispondenti alla guaina sinistra e destra del nervo ottico. Si ricercano i

loro punti estremi superiori e inferiori: quelli destri nel caso della regione sinistra e quelli sinistri nel caso della regione destra. Questi corrispondono a 4 vertici, che vengono utilizzati per ricostruire un poligono. Questo si troverà in corrispondenza della regione compresa tra i due oggetti della maschera binaria (la guaina), ovvero il nervo ottico. Il risultato di questo primo step è mostrato in figura 5.11a.

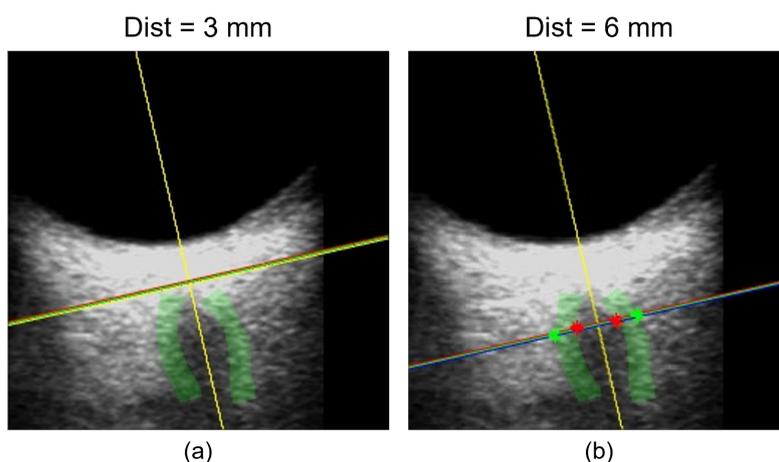
2. **Costruzione centerline.** Si analizza riga per riga l'oggetto nervo ottico ricostruito nello step precedente. Per ogni riga si salva il primo pixel della regione e l'ultimo. Le loro coordinate vengono poi utilizzate per calcolarne il punto medio, che corrisponderà ad uno dei punti della centerline. Ripetendo questa operazione per ogni riga e mettendo insieme le coordinate di ogni punto medio individuato, si otterrà un primo profilo della centerline del nervo ottico. È possibile visualizzarla in figura 5.11b. Nei passaggi successivi sarà necessario avere un prolungamento della centerline oltre l'estensione del nervo ottico. Per questo motivo, il primo profilo ottenuto viene approssimato a una linea retta ( $y(x) = mx + q$ ) passante per i punti che lo definiscono.
3. **Individuazione fondo del bulbo oculare.** Si applica *Regionprops* alla maschera binaria di input per poter analizzare le regioni corrispondenti alla guaina sinistra e destra del nervo ottico. Si ricercano i loro punti estremi superiori: l'estremo sinistro nel caso della regione sinistra e quello destro nel caso della regione destra. Le loro coordinate vengono utilizzate per definire i vertici inferiori (sinistro e destro) di una box alta 60 pixel. Questa viene evidenziata nell'immagine originale, ponendo i pixel al di fuori della box pari ad 1. In base a come è stata costruita la box, al suo interno viene rappresentata la zona finale del bulbo oculare e quella immediatamente superiore al nervo ottico. Da qui viene creata una maschera binaria tramite sogliatura: tutti i pixel con intensità  $< 0.1$  (i pixel hanno intensità comprese tra 0 e 1) vengono posti pari a 1 e i rimanenti pari a 0. Visualizzando a video la maschera binaria, così ottenuta, la zona all'interno della box corrispondente al bulbo oculare apparirà bianca e tutto il resto nero. A questo punto si procede con la ricerca vera e propria del punto che si trova sul fondo del bulbo oculare e dal quale verranno calcolati i 3 mm. È importante che questo si trovi in corrispondenza dell'asse centrale del nervo ottico. Quindi vengono presi in esame solo i pixel appartenenti alla retta che approssima la centerline. Partendo dal pixel più in alto, l'ultimo che avrà intensità pari a 1 sarà proprio il pixel che si trova in corrispondenza del fondo del bulbo oculare. Evidenziato in rosso nella figura 5.11c.
4. **Calcolo di OND e ONSD.** A partire dal pixel del bulbo oculare individuato nello step precedente, si calcola la sua distanza con gli altri punti sottostanti

appartenenti alla centerline. Le distanze in pixel vengono misurate utilizzando la *Geodesic Distance Transform*, e poi convertite in mm moltiplicandole per il Fattore di Calibrazione dell'immagine. Per ridurre la variabilità delle misurazioni finali di OND e ONSD, i diametri non vengono calcolati solo in corrispondenza dei 3 mm al di sotto del bulbo oculare. Ma i loro valori finali vengono ottenuti come valore medio di più misurazioni, effettuate tutte in un intorno dei 3 mm, di ampiezza 0.3 mm. Quindi si ricercano tutti i pixel appartenenti alla centerline posti a una distanza dal bulbo compresa tra 2.85 mm e 3.15 mm ( $2.85 < d < 3.15$ ). In corrispondenza di ognuno di questi pixel vengono calcolati OND e ONSD, ripetendo per ciascuno i passaggi descritti di seguito. Si costruisce la retta perpendicolare alla centerline passante per il punto di interesse. Successivamente si valuta l'intersezione tra la retta perpendicolare e gli oggetti (guaina del nervo ottico) della maschera binaria di input. L'intersezione si tradurrà in due segmenti, uno sovrapposto alla regione rappresentate la guaina sinistra del nervo e l'altro sovrapposto alla regione rappresentante la guaina destra (figura 5.11d). I pixel corrispondenti agli estremi di tali segmenti saranno i punti tra cui verranno misurati i diametri. Il diametro del nervo ottico viene calcolato come la distanza tra i due estremi interni (punti segnati in rosso nella figura 5.11d). Diversamente, il diametro della guaina del nervo viene calcolato come la distanza tra i due estremi più esterni (punti segnati in verde nella figura 5.11d). Entrambe le distanze vengono prima calcolate in pixel e poi convertite tramite il Fattore di Calibrazione in modo da ricavare i valori espressi in mm. Infine, dopo aver ripetuto queste operazioni per ciascuno dei punti posti in un intorno di 3 mm dal bulbo, si ricavano i valori finali di OND e ONSD effettuando la media delle singole misurazioni.



**Figura 5.11:** Step necessari per il calcolo di OND e ONSD. (a) Nervo ottico ricostruito. (b) Maschera binaria di input e primo profilo della centerline del nervo. (c) Fondo del bulbo oculare. (d) Maschera binaria sovrapposta all'immagine originale, si evidenziano la retta che approssima la centerline (giallo) e le rette perpendicolari, in rosso gli estremi dell'OND e in verde quelli dell'ONSD.

Non in tutte le 464 immagini di dataset è stato possibile calcolare i diametri a una distanza di 3 mm dal bulbo oculare. In alcuni casi, infatti, il nervo ottico inizia a svilupparsi al di sotto di tale distanza. Nelle immagini in cui il primo tentativo di calcolo a 3 mm non è andato a buon fine, i diametri sono stati calcolati a 6 mm dal bulbo. Nella figura 5.12 ne è riportato un esempio: è possibile notare come le rette perpendicolari alla centerline a una distanza di 3 mm non riescono ad intersecare la maschera del nervo ottico (in verde). Effettuando un secondo tentativo a 6 mm, il nervo risulta ben formato ed è possibile calcolarne i diametri (OND e ONSD).



**Figura 5.12:** (a) Tentativo di calcolo dei diametri a 3 mm, nervo non identificato. (b) Tentativo di calcolo a 6 mm, nervo identificato.

## 5.8 Metriche di validazione

Per validare le prestazioni dell’algoritmo e confrontarle con il *Ground Truth* ricavato da un operatore manuale, sono state utilizzate alcune metriche di validazione. È stata valutata la bontà dell’algoritmo di segmentazione confrontando ciascuna maschera binaria predetta con la corrispondente manuale, secondo diverse metriche definite come *Pixel-based*. Anche le misurazioni di OND e ONSD ottenute dalle maschere automatiche sono state confrontate con quelle ricavate dalle maschere manuali di *Ground Truth*. In questo caso sono state utilizzate metriche di valutazione statistiche.

### 5.8.1 Metriche di validazione per la segmentazione

Le prestazioni di tutti i modelli allenati sono state validate su tutte le immagini utilizzate in fase di Test della rete. Le maschere predette, ottenute come output

dalle architetture elencate in tabella 5.2, sono state confrontate con le corrispondenti maschere di *Ground Truth* del Test Set (10% del dataset). Da questi primi confronti è stato individuato il modello più performante, riallenato su un dataset per cross-validation. La stessa operazione è stata ripetuta per valutare la bontà dei risultati ottenuti allenando il modello U-Net-ResNet50 utilizzando k-fold cross-validation. In questo caso, avendo usato iterativamente tutte le immagini del dataset come dati di test, è possibile confrontare le maschere automatiche di tutte le 464 immagini con tutte le maschere manuali. Complessivamente il calcolo delle metriche di validazione è stato ripetuto 7 volte. Le metriche utilizzate sono *pixel-based*, ovvero la validazione viene effettuata a livello dei singolo pixel dell'immagine. I singoli pixel avranno valore 1 nel caso in cui ci sia la guaina del nervo, e 0 su tutto il resto dell'immagine. Si definiscono i seguenti termini, necessari per il calcolo delle metriche:

- **Veri Positivi (TP)**: somma dei pixel dell'intersezione tra la maschera manuale e quella automatica, ovvero i pixel correttamente identificati come appartenenti alla guaina.
- **Falsi Positivi (FP)**: somma dei pixel dell'intersezione tra la maschera automatica e la maschera manuale negata. Corrisponde al numero di pixel riconosciuti erroneamente come appartenenti alla guaina.
- **Falsi Negativi (FN)**: somma dei pixel dell'intersezione tra la maschera automatica negata e la maschera manuale. Pixel appartenenti alla guaina del nervo, ma non classificati come tali dall'algoritmo.
- **Veri Negativi (TN)**: somma dei pixel dell'intersezione tra la maschera automatica negata e la maschera manuale negata. Pixel correttamente classificati come non appartenenti alla guaina, ma allo sfondo.

A partire dai parametri appena definiti, sono state calcolate le seguenti metriche:

- **Precision** (Valore predittivo positivo): indica il rapporto tra le previsioni positive corrette e il totale degli elementi classificati come positivi.

$$Precision = \frac{TP}{TP + FP} \quad (5.8)$$

- **Recall**: rappresenta il rapporto tra le previsioni positive corrette e il totale dei pixel che l'algoritmo dovrebbe classificare come appartenenti alla guaina se funzionasse perfettamente.

$$Recall = \frac{TP}{TP + FN} \quad (5.9)$$

- **Dice Coefficient:** corrisponde al doppio del rapporto tra l'intersezione tra la maschera automatica e manuale e il totale dei pixel presenti in entrambe.

$$Dice = 2 \frac{TP}{2TP + FP + FN} \quad (5.10)$$

- **IoU score** (*Jaccard index*): come suggerisce il nome, *Intersection over Union*, indica il rapporto tra l'intersezione delle maschere (manuale e predetta) e la loro unione.

$$IoU = \frac{TP}{TP + FP + FN} \quad (5.11)$$

- **BF score coefficient:** rappresenta la media armonica di Precision e Recall e dà un'indicazione di quanto simili siano i bordi degli oggetti predetti e quelli di Ground Truth.

$$BF = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (5.12)$$

dove *Precision* e *Recall* vengono calcolati secondo le equazioni 5.8 e 5.9

Ognuna delle metriche di validazione presentate può assumere valori compresi nel range  $[0,1]$ . Dove 1 rappresenta una similarità perfetta tra la predizione dell'algoritmo e il Ground Truth. Quindi, maggiori saranno i valori e migliori saranno le performance dell'algoritmo.

Le metriche sono state calcolate su due porzioni diverse dei pixel della maschera manuale e di quella predetta:

1. **Supporto Comune.** Si misurano i valori delle metriche solo in corrispondenza delle righe dell'immagine in cui esistono sia l'oggetto predetto che quello segmentato manualmente. Ovvero si considerano solo le righe in cui si ha intersezione tra maschera automatica e Ground Truth.
2. **Zona 3 mm.** Si calcolano le metriche solo in corrispondenza delle righe che si trovano in un intorno (ampiezza 1 mm) del punto distante 3 mm dal bulbo oculare, in cui verranno calcolati i diametri (OND e ONSD). Nel lavoro di tesi presentato la segmentazione delle strutture del nervo ottico viene effettuata con il fine di calcolarne i diametri, a una distanza pari a 3 mm dal fondo del bulbo oculare. Per definire meglio la bontà della segmentazione automatica, in modo specifico per il *task* che si vuole risolvere, risulta interessante valutare la maschera predetta proprio in prossimità del punto in cui verranno calcolati OND e ONSD.

### 5.8.2 Metriche di validazione per i diametri

Nell'ottica di valutare l'affidabilità dell'algoritmo proposto, si rende necessaria anche una validazione delle misurazioni dell'OND e dell'ONSD. Sono stati, quindi, confrontati i valori dei diametri ricavati dalle maschere automatiche con quelli di Ground Truth calcolati a partire dalle maschere manuali. Nonostante tutti i diametri siano stati ricavati in modo automatico, a partire dalle segmentazioni (predette e manuali), per semplicità, di seguito si parlerà di **misurazioni manuali o di Ground Truth** in riferimento ai diametri calcolati a partire dalle maschere manuali e si indicheranno come **misurazioni automatiche** quelle ricavate dalle maschere predette.

In questo caso sono stati calcolati i seguenti indicatori:

- **Errore Medio:** definito come la differenza media tra la misura di Ground Truth e quella automatica.

$$ME = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i) \quad (5.13)$$

dove  $N$  corrisponde al numero totale di misurazioni,  $y_i$  indica il valore di Ground Truth del diametro e  $\hat{y}_i$  il valore della misurazione automatica. Visto che i valori dei diametri sono stati calcolati per ogni immagine,  $N$  risulta pari a 464.

- **Errore Medio Assoluto:** matematicamente rappresenta la distanza tra il valore predetto e quello effettivo.

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (5.14)$$

dove le grandezze indicate assumono gli stessi significati descritti per l'errore medio (Eq. 5.13).

- **Figure of Merit (FoM):** parametro che caratterizza le prestazioni complessive dell'algoritmo. Definito come segue:

$$FoM = 100 - \left| \frac{mean(\hat{y}) - mean(y)}{mean(y)} \right| \cdot 100 \quad (5.15)$$

dove  $mean(\hat{y})$  e  $mean(y)$  corrispondono rispettivamente alla media dei valori di diametro misurati automaticamente, e alla media dei valori delle misurazioni di Ground Truth [29].

- **Coefficiente di Correlazione di Pearson:** tecnica statistica per valutare la correlazione lineare tra due variabili, in questo caso misurazione manuale e automatica dei diametri. Viene definito come segue:

$$r(A, B) = \frac{1}{N-1} \sum_{i=1}^N \left( \frac{A_i - \mu_A}{\sigma_A} \right) \left( \frac{B_i - \mu_B}{\sigma_B} \right) \quad (5.16)$$

dove A e B indicano due grandezze scalari, nel lavoro di tesi corrispondono al valore del diametro di Ground Truth e a quello misurato automaticamente,  $\mu_A$  e  $\mu_B$  indicano, rispettivamente, la media dei valori di A e la media dei valori di B,  $\sigma_A$   $\sigma_B$  invece rappresentano le loro deviazioni standard. Infine  $N$  corrisponde, nuovamente, al numero totale di valori pari a 464 [67].

- **Intraclass Correlation Coefficient (ICC):** è una misura statistica che fornisce un'indicazione dell'affidabilità delle misurazioni organizzate in gruppi. Descrive quanto le unità di uno stesso gruppo si somiglino. Nel caso in esame i dati appartenenti a uno stesso gruppo sono i diametri misurati dalla stessa immagine.
- **Test signed rank di Wilcoxon:** fornisce indicazioni in merito alla presenza o meno di una significativa differenza statistica tra le misurazioni effettuate. Forniti i valori di diametro di Ground Truth e quelli misurati in modo automatico, indicati rispettivamente con  $x$  e  $y$ , si valuta che la mediana di  $x - y$  sia diversa da zero. Il risultato viene fornito sotto forma di *p-value*: se  $p < 0.05$  è presente una significativa differenza statistica tra le due misurazioni.

Per ogni immagine sono state effettuate due misurazioni differenti, una per il diametro del nervo ottico (OND) e una per il diametro della guaina (ONSD). Entrambe le misure sono state confrontate con le corrispondenti misurazioni manuali, quindi tutte le metriche descritte in questo paragrafo sono state applicate due volte. Una per la validazione dei valori dell'OND e una per la validazione dei valori di dell'ONSD.



# Capitolo 6

## Risultati

In questo capitolo si riportano i risultati ottenuti dal sistema automatico proposto nel capitolo precedente. Vengono descritte le validazioni quantitative effettuate sull'output delle reti. Per valutare la bontà dei modelli e la loro capacità di generalizzazione sono state confrontate le segmentazioni predette con quelle automatiche, utilizzando le metriche precedentemente presentate. Si riportano, anche, i risultati ottenuti dal confronto dei valori dei diametri del nervo ottico e della guaina.

### 6.1 Validazione delle segmentazioni

Le prestazioni delle reti allenate e la bontà delle segmentazioni, ottenute come output, sono state valutate sulle immagini di test. Le metriche di validazione *pixel-based* sono state applicate agli output di ogni architettura allenata. Si presentano prima i risultati ottenuti dal training delle reti U-Net, UNet++ e MA-Net con gli encoder ResNet50 ed EfficientNet-B4. Successivamente vengono descritti i risultati ottenuti in seguito all'allenamento dell'architettura U-Net - ResNet50 con cross-validation.

#### 6.1.1 Risultati degli allenamenti di U-Net, UNet++ e MA-Net

Si riportano le metriche di segmentazione, descritte nel paragrafo 5.8, per ognuno dei primi 6 modelli allenati. I risultati delle metriche sono riportati in tabella 6.1 con lo scopo di effettuarne un confronto. Così da poter individuare il modello più performante sul dataset utilizzato.

Architetture	IoU	Dice	BFscore	Precision	Recall
U-Net - ResNet50	$0.76 \pm 0.14$	$0.85 \pm 0.12$	$0.83 \pm 0.16$	$0.82 \pm 0.13$	$0.89 \pm 0.13$
U-Net - EfficientNet-B4	$0.74 \pm 0.12$	$0.84 \pm 0.10$	$0.81 \pm 0.15$	$0.81 \pm 0.13$	$0.89 \pm 0.09$
UNet++ - ResNet50	$0.74 \pm 0.15$	$0.84 \pm 0.12$	$0.81 \pm 0.17$	$0.81 \pm 0.14$	$0.89 \pm 0.11$
UNet++ - EfficientNet-B4	$0.73 \pm 0.14$	$0.84 \pm 0.11$	$0.80 \pm 0.17$	$0.80 \pm 0.13$	$0.89 \pm 0.10$
MA-Net - ResNet50	$0.75 \pm 0.13$	$0.85 \pm 0.10$	$0.83 \pm 0.16$	$0.82 \pm 0.13$	$0.90 \pm 0.09$
MA-Net - EfficientNet-B4	$0.72 \pm 0.15$	$0.82 \pm 0.12$	$0.77 \pm 0.17$	$0.77 \pm 0.13$	$0.91 \pm 0.12$

**Tabella 6.1:** Metriche di validazione pixel-based. Si riportano IoU, Dice coefficient, BFscore, Precision e Recall. Si indicano i valori medi e le rispettive deviazioni standard.

I valori ottenuti dal confronto delle maschere predette con quelle manuali per ognuno dei modelli sono molto simili tra loro. Non è possibile individuare un'architettura che spicchi sulle altre. Volendo comunque indicare le reti di cui sono stati ottenuti i valori più alti, si possono nominare le reti U-Net e MA-Net, entrambe allenate con ResNet50 come encoder. Nella scelta del modello da utilizzare per eseguire cross-validation, a parità di risultati, si predilige la semplicità del modello. Sulla base dei dati riportati, quindi, si decide di riallenare la rete U-Net - ResNet50 sfruttando la tecnica della cross-validation.

### 6.1.2 Risultati dell'allenamento con cross-validation

In questo paragrafo si riportano i risultati dell'allenamento per Cross-Validation. Il training in questo caso viene ripetuto iterativamente 5 volte e viene testato su 5 set di Test differenti. Ognuna delle immagini dei 5 fold di Test è stata processata con successo dal modello. Ricordando che l'unione dei 5 fold di Test costituisce la totalità delle immagini, si deduce che l'algoritmo è in grado di processare il 100% delle immagini di dataset. Si valuta poi la correttezza del *processing* sulla base dei valori delle metriche di validazione utilizzate.

Innanzitutto si valuta la segmentazione ottenuta per ogni singolo fold utilizzato come set di Test. Nella tabella 6.2 si riportano i valori del Coefficiente Dice ricavati confrontando, per ogni Test set, le maschere predette con le corrispondenti automatiche.

	Fold1	Fold2	Fold3	Fold4	Fold5
Dice	$0.83 \pm 0.14$	$0.87 \pm 0.07$	$0.85 \pm 0.11$	$0.86 \pm 0.09$	$0.87 \pm 0.08$

**Tabella 6.2:** Valori medi e rispettive deviazioni standard del coefficiente Dice calcolato per ogni Fold di Test.

Dopo di ché, si effettua la media dei 5 training diversi. Si uniscono tutti i fold di Test Set per ottenere un set di Test risultante sul quale vengono valutate le

prestazioni complessive del modello. Osservando la tabella 6.3 è possibile valutare le performance della rete tramite i valori delle metriche.

<b>IoU</b>	<b>Dice</b>	<b>BFscore</b>	<b>Precision</b>	<b>Recall</b>
$0.76 \pm 0.13$	$0.86 \pm 0.10$	$0.87 \pm 0.13$	$0.83 \pm 0.13$	$0.90 \pm 0.11$

**Tabella 6.3:** Metriche di validazione pixel-based. Si riportano IoU, Dice coefficient, BFscore, Precision e Recall. Si indicano i valori medi e le rispettive deviazioni standard.

I risultati risultano buoni, in particolare Dice e BFscore con percentuali pari, rispettivamente all’86% e all’87%. Nella valutazione complessiva delle prestazioni del modello, i singoli valori ottenuti per il Coefficiente Dice vengono utilizzati come discriminante per valutare la correttezza della segmentazione. Si definisce la presenza di un errore di *processing* nel caso in cui Dice assuma un valore inferiore a 0.2. Tra la 464 immagini di dataset, per 2 di queste si ottiene  $Dice < 0.2$ . Inoltre si riporta il 1°-percentile dei valori del coefficiente Dice, pari a 0.41. Quindi l’algoritmo risulta essere in grado di processare correttamente 2 su 464 immagini, ovvero il 99,6% del dataset.

## 6.2 Validazione dei valori di OND e ONSD

Per il lavoro di tesi proposto, più che la segmentazione delle strutture, è rilevante che i valori dei diametri misurati siano affidabili. Anche le misurazioni di OND e ONSD sono state validate su tutte le 464 immagini di dataset. I valori medi e gli errori di misurazione del diametro del nervo ottico e della guaina sono riportati nella tabella 6.4.

	<b>Algoritmo</b>		<b>Ground Truth</b>	
	OND	ONSD	OND	ONSD
Mean Value (mm)	$2.6 \pm 0.39$	$5.7 \pm 0.58$	$2.7 \pm 0.44$	$5.6 \pm 0.44$
Mean Error (mm)	$-0.12 \pm 0.32$	$0.14 \pm 0.58$		
Mean Absolute Error (mm)	$0.25 \pm 0.24$	$0.43 \pm 0.41$		
Figure of Merit (FoM)	95.6%	97.5%		

**Tabella 6.4:** Risultati dell’algoritmo per l’OND e l’ONSD confrontati con le misurazioni manuali. Sono riportati i valori medi e le loro deviazioni standard.

Confrontando i valori si può notare subito come il calcolo automatico commetta un errore dell’ordine di 0,1 mm. Nello specifico si hanno errori medi di  $-0.12 \pm 0.32$  e  $0.14 \pm 0.58$ , rispettivamente per l’OND e per l’ONSD. Inoltre si possono

apprezzare valori piuttosto elevati di *Figure of Merit*, 95.6% per l'OND e 97.5% per l'ONSD.

L'esecuzione del test statistico di Wilcoxon ha evidenziato la presenza di una significativa differenza statistica tra le misurazioni automatiche dell'algoritmo proposto e le misurazioni ricavate dalle maschere manuali ( $p < 0.05$ ), sia per i valori di OND che per quelli di ONSD.

In tabella 6.5 sono riportati i valori del coefficiente di correlazione di Pearson ottenuti sia per il diametro del nervo che per quello della guaina. Si riportano anche l'intervallo di confidenza al 95% e i *p-values* ottenuti.

	Correlation coefficient	Confidence Interval		<i>p value</i>
		Lower limit	Upper Limit	
<b>OND</b>	0.71	0.66	0.75	$6.31 \cdot 10^{-72}$
<b>ONSD</b>	0.64	0.59	0.69	$8.24 \cdot 10^{-56}$

**Tabella 6.5:** Coefficiente di correlazione di Person valutato per OND e ONSD.

In questo caso i *p-values* ottenuti ( $p \ll 0.05$ ) mostrano una correlazione statisticamente significativa tra i valori ricavati dalle maschere predette e quelli ricavati dalle maschere manuali. Questo vale sia per l'OND che per l'ONSD.

Nella tabella 6.6 sono riassunti i valori del coefficiente di intraclassa ottenuti per il diametro nel nervo e per la guaina.

	ICC	Confidence Interval	
		Lower limit	Upper Limit
<b>OND</b>	0.80	0.76	0.84
<b>ONSD</b>	0.76	0.71	0.80

**Tabella 6.6:** Intraclass Correlation Coefficient per OND e ONSD.

Complessivamente i risultati ottenuti con la valutazione dei coefficienti di correlazione risultano buoni. Tutti i coefficienti hanno valori al di sopra del 60%, raggiungendo un massimo pari all'80% per le misurazioni del diametro del nervo ottico.

### 6.3 Validazione dei risultati per ogni macchinario

Si vogliono indagare le performance dell'algoritmo sulla base del macchinario ecografico utilizzato in fase di acquisizione. Lo scopo è valutare se il modello proposto performa diversamente in base alla tipologie di ecografo da cui deriva l'immagine.

In caso di risposta affermativa, risulta di interesse individuare quale macchinario dia risultati migliori e quale risultati peggiori. Per effettuare questo controllo tutte le maschere automatiche, così come quelle manuali, sono state divise in 4 gruppi differenti in base al macchinario con il quale le immagini da cui derivano sono state acquisite. Per ogni gruppo le maschere predette sono state confrontate con quelle di Groun Truth ed è stato calcolato il coefficiente Dice per valutare la bontà delle segmentazioni. In tabella 6.7 sono riportati i valori di Dice ottenuti. I macchinari qui sono indicati come Mac1, Mac2, Mac3 e Mac4; è possibile risalire all'ecografo corrispondente facendo riferimento alla tabella 5.1.

	<b>Mac1</b>	<b>Mac2</b>	<b>Mac3</b>	<b>Mac4</b>
Dice	$0.86 \pm 0.11$	$0.87 \pm 0.07$	$0.81 \pm 0.11$	$0.88 \pm 0.06$

**Tabella 6.7:** Coefficiente Dice valutato per ogni macchinario. Si riportano i valori medi e le corrispondenti deviazioni standard.

I valori ottenuti sono coerenti con gli quelli riportati nelle tabelle 6.2 e 6.3.

Per ogni macchinario si confrontano anche i valori dei diametri (OND e ONSD) misurati. Si riportano quindi i valori medi di misurazione divisi per ecografo utilizzato (tabella 6.8).

	<b>Algoritmo</b>		<b>Ground Truth</b>	
	OND	ONSD	OND	ONSD
Mac1	$2.5 \pm 0.35$	$5.8 \pm 0.55$	$2.6 \pm 0.39$	$5.7 \pm 0.39$
Mac2	$2.8 \pm 0.36$	$5.8 \pm 0.61$	$2.9 \pm 0.42$	$5.6 \pm 0.42$
Mac3	$2.4 \pm 0.39$	$5.5 \pm 0.59$	$2.5 \pm 0.46$	$5.3 \pm 0.46$
Mac4	$2.5 \pm 0.28$	$5.7 \pm 0.48$	$2.6 \pm 0.39$	$5.6 \pm 0.39$

**Tabella 6.8:** Valori medi e deviazioni standard di OND e ONSD per ognuno dei macchinari utilizzati. I valori sono espressi in mm.

Nella tabella 6.9, invece, sono riassunti gli errori di misurazione commessi dall'algoritmo durante il calcolo dei diametri. Come riscontrato in precedenza l'errore medio commesso in fase misurazione è di media pari a 0.1 mm, fatta eccezione per i calcoli dei diametri effettuati sulle immagini acquisite con il terzo macchinario. In questo caso i valori dell'errore medio risultano più alti, pari a  $-0.18 \pm 0.44$  e  $0.19 \pm 0.71$ , rispettivamente per l'OND e l'ONSD.

	Measure	Mean Error (mm)	MAE (mm)	FoM
Mac1	OND	$-0.10 \pm 0.29$	$0.22 \pm 0.21$	96.00%
	ONSD	$0.10 \pm 0.60$	$0.43 \pm 0.42$	98.30%
Mac2	OND	$-0.10 \pm 0.28$	$0.22 \pm 0.20$	96.60%
	ONSD	$0.20 \pm 0.42$	$0.34 \pm 0.31$	96.40%
Mac3	OND	$-0.18 \pm 0.44$	$0.35 \pm 0.32$	93.00%
	ONSD	$0.19 \pm 0.71$	$0.58 \pm 0.45$	96.40%
Mac4	OND	$-0.10 \pm 0.25$	$0.21 \pm 0.17$	96.10%
	ONSD	$0.09 \pm 0.49$	$0.35 \pm 0.36$	98.40%

**Tabella 6.9:** Errori delle misurazioni di OND e ONSD suddivisi per macchinario. Dove MAE indica l'errore medio assoluto e FoM il Figure of Merit.

Infine si riportano i valori dei coefficienti di correlazione calcolati. In tabella 6.10 sono indicati i valori del coefficiente di correlazione di Pearson e dei coefficienti di intraclassa (ICC). Per entrambi i calcoli sono stati effettuati per ognuno degli ecografi utilizzati durante l'acquisizione delle immagini.

	Measure	Pearson Coefficient	ICC
Mac1	OND	0.70	0.80
	ONSD	0.60	0.73
Mac2	OND	0.75	0.84
	ONSD	0.78	0.85
Mac3	OND	0.46	0.57
	ONSD	0.44	0.57
Mac4	OND	0.78	0.82
	ONSD	0.82	0.83

**Tabella 6.10:** Valori dei coefficienti di correlazione di Pearson e di intraclassa, riportati per ogni macchinario sia per OND che per ONSD.

Anche in questo caso i valori ottenuti risultano coerenti con i valori ottenuti sulla validazione di tutto il dataset, fatta nuovamente eccezione per il macchinario che non raggiunge valori di correlazione pari al 50% nel caso del coefficiente di Pearson.

## 6.4 Confronto dell’algoritmo proposto con AUTONoMA

Il sistema automatico sviluppato e presentato nel lavoro di tesi qui esposto, nasce come evoluzione del sistema AUTONoMA, proposto nel 2020 da Meiburger et al. [29] e riportato nel paragrafo 3.2. Con l’algoritmo proposto si vuole colmare i limiti riscontrati nell’applicazione di AUTONoMA. Si ritiene, quindi, di interesse effettuare un confronto tra le prestazioni dei due sistemi, in termini di correttezza delle misurazioni dei diametri (OND e ONSD). Nello specifico vengono messi a confronto gli errori di misurazione e i valori del coefficiente di correlazione di Pearson, riportati in tabella 6.11. L’algoritmo proposto in questo lavoro di tesi viene indicato nella tabella come *Nuova Proposta*.

	AUTONoMA		Nuova Proposta	
	95.00%		100%	
Immagini Processate	OND	ONSD	OND	ONSD
Mean Error (mm)	$0.06 \pm 0.35$	$0.06 \pm 0.52$	$-0.12 \pm 0.32$	$0.14 \pm 0.58$
Mean Absolute Error (mm)	$0.28 \pm 0.22$	$0.41 \pm 0.32$	$0.25 \pm 0.24$	$0.43 \pm 0.41$
Figure of Merit (FoM)	98.2%	99.0%	95.6%	97.5%
Pearson Coefficient	0.47	0.64	0.71	0.64

**Tabella 6.11:** Confronto tra l’algoritmo proposto e AUTONoMA.

Nel 7% delle immagini processate da AUTONoMA (5 su 71) le segmentazioni sono state ritenute non accettabili secondo il parere di un operatore manuale esperto. Del 100% del dataset processato dal nuovo sistema proposto, il 0.4% delle segmentazioni (2 su 464) sono state considerate errate sulla base del valore di Dice ottenuto.

## 6.5 Discussione dei risultati

### 6.5.1 Algoritmo proposto

I risultati delle segmentazioni ottenuti forniscono una prima informazione in merito alle prestazioni dell’algoritmo. Con un valore di Dice pari a  $0.86 \pm 0.1$  e un valore di BFscore di  $0.87 \pm 0.13$ , i risultati relativi alle segmentazione delle strutture del nervo ottico si ritengono soddisfacenti.

In ambito clinico, l’analisi delle immagini di ultrasonografia transorbitale e delle strutture nel nervo ottico è funzionale alla misurazione di OND e ONSD. Ovvero piuttosto che la segmentazione del nervo, sono i valori dei diametri e la loro correttezza ad essere di interesse clinico e diagnostico. Pertanto risulta

fondamentale valutarne l'affidabilità. I valori medi di OND e ONSD, ottenuti dal sistema proposto, pari rispettivamente a  $2.6 \pm 0.39$  e  $5.7 \pm 0.58$ , risultano molto simili ai corrispondenti valori medi ricavati dalle maschere manuali ( $2.7 \pm 0.44$  e  $5.6 \pm 0.44$ ). Si discostano tra loro di una quantità pari a 0.1 mm. Valore che rientra pienamente nel range di tolleranza, pari a 0.2 mm, adottato in ambito clinico. Dai valori appena citati è possibile notare che l'algoritmo sovrastima le misurazioni nel caso del diametro della guaina e le sottostima nel caso del diametro del nervo.

L'analisi statistica effettuata tramite test di Wilcoxon evidenzia una significativa differenza tra i valori di OND e ONSD di Ground Truth e tra quelli misurati in modo automatico. Tuttavia questo dato fornisce solo informazioni parziali. Infatti, valutando i coefficienti di correlazione tra le misurazioni risulta esserci una correlazione positiva tra i risultati. I valori di ICC sono entrambi al di sopra del 70%: si raggiunge una correlazione pari all'80% per i valori del diametro del nervo, e una correlazione pari al 76% per quelli dell'ONSD. Complessivamente si può osservare come le prestazioni sulle misurazioni dell'ONSD siano inferiori. Questo è causato dalla presenza di oggettive difficoltà tecniche nel differenziare la guaina del nervo ottico iperocogena dal grasso periorbitale, anch'esso iperocogeno. Difficoltà riscontrate non solo dall'algoritmo ma anche dai clinici esperti.

Il confronto effettuato sui macchinari ha evidenziato la presenza di un ecografo problematico per il modello allenato. Effettivamente i risultati ottenuti per le immagini acquisite con il macchinario 3 sono al di sotto della media dei risultati raggiungibili con le altre immagini. Gli errori medi di misurazione della macchina 3 sono pari quasi al doppio di quelli ottenuti dagli altri macchinari. La stessa osservazione può essere estesa ai valori dei coefficienti di correlazione. Probabilmente escludendo dal dataset le immagini acquisite con il macchinario 3, si assisterebbe ad un complessivo aumento delle performance dell'algoritmo. Si nota, anche, la presenza di una macchina ecografica sulla quale il modello sviluppato risulta molto performante, ovvero il macchinario 4.

Infine, nella valutazione complessiva dei risultati ottenuti, è importante tenere in considerazione che la misurazione dei diametri dipende dalla distanza misurata in pixel. Questo vuol dire che è sufficiente commettere un errore di misurazione anche di un solo pixel nell'immagine, per ottenere valori in mm molto diversi tra loro.

## 6.5.2 Confronto con AUTONoMA

Analizzando i risultati ottenuti dal confronto tra il sistema proposto e AUTONoMA è possibile notare come gli errori medi di misurazione commessi dal primo siano maggiori rispetto a quelli commessi da AUTONoMA. A differenza, invece, paragonando i rispettivi coefficienti di correlazione, le prestazioni relative al calcolo del diametro del nervo ottico risultano essere maggiori per il nuovo sistema proposto rispetto

ad AUTONoMA. Per quest'ultimo la correlazione tra le misurazioni automatiche di OND e quelle manuali è pari al 47%, inferiore del 22% rispetto al valore di correlazione raggiunto dall'algoritmo presentato in questo lavoro di tesi. In merito all'ONSD i valori di correlazione, invece, risultano identici per i due sistemi discussi. Inoltre si nota come, diversamente da ciò che è stato riscontrato dall'analisi dei risultati ottenuti dal nuovo sistema, AUTONoMA risulta più performante nel calcolo del diametro della guaina piuttosto che nella misurazione del diametro del nervo.

Nel mettere a confronto i due algoritmo va sicuramente tenuto conto di come questi sono stati sviluppati. La valutazione delle strutture del nervo ottico implementata dal nuovo sistema si basa sul Deep Learning, AUTONoMA invece si basa sull'impiego di modelli deformabili. In questo caso l'algoritmo deve fare delle assunzioni a priori su come le specifiche strutture sono rappresentate nell'immagine, con il rischio che se la rappresentazione effettiva è troppo diversa dalla tipica immagine di ecografia transorbitale, l'algoritmo non sia in grado di elaborare correttamente l'immagine. AUTONoMA, infatti, necessita di una sostanziale differenza tra le strutture ipoecogene e iperecogene, per poter riconoscere correttamente i profili del nervo ottico e della guaina e calcolarne i diametri. Inoltre è stato costruito e validato su un dataset di 75 immagini acquisite da un unico macchinario ecografico. Questo potrebbe aver reso il riconoscimento delle strutture del nervo molto specifico per le immagini di dataset e poco generalizzabile. Diversamente per lo sviluppo di questo lavoro di tesi è stato possibile accedere a una maggiore quantità di immagini di ecografia transorbitale, anche molto diverse tra loro, provenienti da strutture ospedaliere differenti e, quindi, acquisite da macchinari diversi. Questo ha permesso di costruire un dataset eterogeneo da 464 immagini acquisite con 4 diverse macchine ecografiche. Unendo l'ampio numero di immagini all'impiego del Deep Learning è stato possibile sviluppare un sistema automatico con una buona capacità di generalizzazione. Da qui le differenti percentuali di immagini processate. AUTONoMA riesce a processare il 95% delle immagini (71 su 75) di cui il 93% in modo corretto, il nuovo sistema proposto invece è in grado di processare la totalità delle immagini di dataset di cui il 99,6% correttamente.

Quindi il nuovo algoritmo proposto riesce a superare i limiti evidenziati dagli sviluppatori stessi di AUTONoMA, ovvero l'accesso a un numero troppo ridotto di immagini per poter validare meglio il sistema, la provenienza della immagini da un solo macchinario e infine l'eccessiva dipendenza dall'ipoecogenicità o ipercogenicità delle strutture del nervo ottico.



## Capitolo 7

# Conclusioni e sviluppi futuri

Il lavoro di tesi presentato propone un algoritmo per la segmentazione delle strutture anatomiche del nervo ottico e il calcolo dei diametri (OND e ONSD), in immagini di ultrasonografia transorbitale acquisite in modalità B-Mode. Si vuole fornire uno strumento di supporto clinico per l'indagine diagnostica delle alterazioni, in termini di diametri, del nervo ottico. Il progetto proposto risulta essere il primo sistema completamente automatico, pensato per la misurazione di OND e ONSD, basato su algoritmi di Deep Learning. Inoltre il dataset a disposizione è 6-7 volte più ampio del set di dati utilizzati nei precedenti studi di automatizzazione del calcolo di OND e ONSD.

Il valore medio del diametro del nervo ottico ottenuto dall'algoritmo,  $2.6 \pm 0.39$  mm, è molto simile al valore ottenuto dalla misurazione manuale,  $2.7 \pm 0.44$  mm. Si può dire la stessa cosa per i valori medi di ONSD ricavati dall'algoritmo,  $5.7 \pm 0.58$  mm, e dall'operatore manuale,  $5.6 \pm 0.44$  mm. In entrambi i casi i risultati misurati in modo automatico si discostano da quelli di Ground Truth di solo 0.1 mm. Tenendo in considerazione che nella pratica clinica viene tollerata una differenza delle misurazioni di 0.2 mm, i valori misurati dall'algoritmo sono decisamente paragonabili alla valutazione di un operatore manuale. In generale, i risultati ottenuti sono molto robusti in quanto ricavati dalla media dei 5 output dei 5 allenamenti della rete effettuati con cross-validation.

Dunque le performance raggiunte con un sistema basato sul Deep Learning rappresentano una solida base per futuri miglioramenti e lo sviluppo di un sistema effettivamente impiegabile nell'indagine clinica delle strutture del nervo ottico.

Il progetto proposto potrebbe essere per prima cosa arricchito dal confronto con almeno un secondo operatore manuale. Nel lavoro di tesi non è stato possibile ricavare le maschere di segmentazione manuale da due diversi operatori. Questo ha impedito la possibilità di valutare la variabilità intra-operatore e di confrontarla con la variabilità operatore-algoritmo. Risulterebbe, quindi, interessante aggiungere le annotazioni manuali di un secondo operatore in modo da offrire una validazione più

robusta delle performance del sistema sviluppato. In questo modo sarà possibile stabilire se la variabilità introdotta dalle misurazioni automatiche rientra nel range della variabilità intra-operatore.

Inoltre le reti implementate sono state allenate su un dataset comprendente 464 immagini. Un possibile miglioramento futuro potrebbe essere legato proprio all'ampliamento del dataset. Avere a disposizione una maggiore numerosità di dati consentirebbe di ottenere prestazioni più elevate. Più immagini di input vengono fornite alle reti, più sarà possibile ottenere modelli con elevate capacità di generalizzazione e quindi più performanti.

# Bibliografia

- [1] Foram Gala. «Magnetic resonance imaging of optic nerve». In: *The Indian journal of radiology and imaging* 2015 (nov. 2015), pp. 421–38. DOI: 10.4103/0971-3026.169462 (cit. alle pp. 1, 2).
- [2] Amedeo Lucente. *Neurite ottica: l'infiammazione del nervo ottico*. URL: <https://www.amedeolucente.it/neurite-ottica.html> (cit. a p. 1).
- [3] Giacomo Viola, Filippo Molinari, Kristen Meiburger e Nicola Michielli. «Segmentazione e caratterizzazione automatica del nervo ottico in immagini ecografiche transorbitali per la valutazione di disturbi neurologici= Automated optic nerve segmentation and characterization in transorbital ultrasound images for neurological disorders assessment». Tesi di dott. Politecnico di Torino, 2019 (cit. alle pp. 2, 11).
- [4] Henry Gray. *Anatomy of Human Body*. Philadelphia e New York: LEA FABIGER, 1918, pp. 872–879 (cit. a p. 3).
- [5] Backlund Jye-yu C. Beck Roy W. Cleary Patricia A. «The course of visual recovery after optic neuritis. Experience of the Optic Neuritis Treatment Trial». In: *Ophthalmology* 101 (1994), pp. 1771–1778. DOI: 10.1016/s0161-6420(94)31103-1 (cit. a p. 3).
- [6] Humanitas Gradenigo Presidio Sanitario Torino (Italia). *Il nervo ottico e le sue patologie da tenere sotto stretto controllo*. URL: <https://www.gradenigo.it/news/nervo-ottico-patologie/> (cit. a p. 4).
- [7] A Candelieri Merlicco, L Gabaldón Torres, R Villaverde González, I Fernández Romero, E Aparicio Castro e MC Lastres Arias. «Transorbital ultrasonography for measuring optic nerve atrophy in multiple sclerosis». In: *Acta Neurologica Scandinavica* 138.5 (2018), pp. 388–393 (cit. a p. 4).
- [8] HC Hansen e K Helmke. «The subarachnoid space surrounding the optic nerves. An ultrasound study of the optic nerve sheath». In: *Surgical and Radiologic Anatomy* 18.4 (1996), pp. 323–328 (cit. a p. 4).

- [9] Iscander M Maissan, Perjan JAC Dirven, Iain K Haitisma, Sanne E Hoeks, Diederik Gommers e Robert Jan Stolker. «Ultrasonographic measured optic nerve sheath diameter as an accurate and quick monitor for changes in intracranial pressure». In: *Journal of neurosurgery* 123.3 (2015), pp. 743–747 (cit. alle pp. 4, 5).
- [10] YÜCEL YÜZBAŞIOĞLU, SEMA YÜZBAŞIOĞLU, Selçuk Coşkun, Ferhat Icme, Tolga Öz, Refik Kunt, Sinan Becel, Emine Akcay e HAVVA ŞAHİN KAVAKLI. «Bedside measurement of the optic nerve sheath diameter with ultrasound in cerebrovascular disorders». In: *Turkish journal of medical sciences* 48.1 (2018), pp. 93–99 (cit. alle pp. 4, 5).
- [11] Piergiorgio Lochner, Francesco Brigo, Maria Luisa Zedde, Sandro Sanguigni, Lorenzo Coppo, Raffaele Nardone, Andrea Naldi, Daniele Sola e Erwin Stolz. «Feasibility and usefulness of ultrasonography in idiopathic intracranial hypertension or secondary intracranial hypertension». In: *BMC neurology* 16.1 (2016), pp. 1–7 (cit. alle pp. 4, 12).
- [12] WD Newman, AS Hollman, GN Dutton e R Carachi. «Measurement of optic nerve sheath diameter by ultrasound: a means of detecting acute raised intracranial pressure in hydrocephalus». In: *British journal of ophthalmology* 86.10 (2002), pp. 1109–1113 (cit. a p. 4).
- [13] Heidi Harbison Kimberly, Sachita Shah, Keith Marill e Vicki Noble. «Correlation of optic nerve sheath diameter with direct measurement of intracranial pressure». In: *Academic Emergency Medicine* 15.2 (2008), pp. 201–204 (cit. a p. 4).
- [14] Jochen Bäuerle, Florian Schuchardt, Laure Schroeder, Karl Egger, Matthias Weigel e Andreas Harloff. «Reproducibility and accuracy of optic nerve sheath diameter assessment using ultrasound compared to magnetic resonance imaging». In: *BMC neurology* 13.1 (2013), pp. 1–6 (cit. a p. 5).
- [15] Chetan G Shirodkar, Kartik Munta, S Manimala Rao e M Uma Mahesh. «Correlation of measurement of optic nerve sheath diameter using ultrasound with magnetic resonance imaging». In: *Indian journal of critical care medicine: peer-reviewed, official publication of Indian Society of Critical Care Medicine* 19.8 (2015), p. 466 (cit. a p. 5).
- [16] Robert Ohle, Sarah M McIsaac, Michael Y Woo e Jeffrey J Perry. «Sonography of the optic nerve sheath diameter for detection of raised intracranial pressure compared to computed tomography: a systematic review and meta-analysis». In: *Journal of Ultrasound in Medicine* 34.7 (2015), pp. 1285–1294 (cit. a p. 5).
- [17] Reparto di Medicina Materno-Fetale Ospedale Buzzi Milano (Italia). *Ecografia per valutazione anatomia fetale*. URL: <https://www.terapiafetale.it/ecografia-su-indicazione/> (cit. a p. 7).

- [18] Massimiliano Andrioli. *Lecografia tiroidea*. URL: <https://www.endocrinologiaoggi.it/2011/06/ecografia-tiroidea/> (cit. a p. 7).
- [19] Alberto Tarozzi. *Cos'è l'ecografia*. 2017. URL: <https://www.ecografiabologna-tarozzi.it/cose-lecografia> (cit. alle pp. 7, 13).
- [20] Kristen M Meiburger. *Quantitative ultrasound and photoacoustic imaging for the assessment of vascular parameters*. Springer, 2016 (cit. alle pp. 8, 9).
- [21] Filippo Molinari. *Dispositivi ad Ultrasuoni*. Politecnico di Torino, 2016 (cit. a p. 10).
- [22] Piergiorgio Lochner, Stefanie Behnke, Klaus Fassbender, Anika Andrejewski, Sophia Knodel, Antonio Siniscalchi, Klaus-V Jenderka e Robert Lemor. «Simulation and experimental characterization of lateral imaging resolution of ultrasound systems and assessment of system suitability for acoustic optic nerve sheath diameter measurement». In: *Journal of Neuroimaging* 29.1 (2019), pp. 34–41 (cit. alle pp. 11, 12, 14).
- [23] GRAMMAMEDICA Roma (Italia). *Ecografia oculare del bulbo e dell'orbita*. URL: <https://www.gammamedica.it/ecografia-oculare/> (cit. a p. 12).
- [24] SonoSim. *Distaccamento di coroide, una rara causa di perdita della vista diagnosticata da ultrasuoni point-of-care*. URL: <https://sonosim.com/it/2018/11/choroid-detachment-a-rare-cause-of-vision-loss-diagnosed-by-point-of-care-ultrasound/> (cit. a p. 12).
- [25] Piergiorgio Lochner, Lorenzo Coppo, Roberto Cantello, Raffaele Nardone, Andrea Naldi, Maurizio A Leone e Francesco Brigo. «Intra-and interobserver reliability of transorbital sonographic assessment of the optic nerve sheath diameter and optic nerve diameter in healthy adults». In: *Journal of ultrasound* 19.1 (2016), pp. 41–45 (cit. a p. 13).
- [26] Kristen M Meiburger, U Rajendra Acharya e Filippo Molinari. «Automated localization and segmentation techniques for B-mode ultrasound images: A review». In: *Computers in biology and medicine* 92 (2018), pp. 210–235 (cit. a p. 14).
- [27] Samuel Gerber, Maeliss Jallais, Hastings Greer et al. «Automatic estimation of the optic nerve sheath diameter from ultrasound images». In: *Imaging for Patient-Customized Simulations and Systems for Point-of-Care Ultrasound*. Springer, 2017, pp. 113–120 (cit. alle pp. 16–18).
- [28] MathWorks Italia. *Distance Transform of binary image - MATLAB bwdist*. URL: <https://it.mathworks.com/help/images/ref/bwdist.html> (cit. a p. 16).

- 
- [29] Kristen M Meiburger, Andrea Naldi, Nicola Michielli, Lorenzo Coppo, Klaus Fassbender, Filippo Molinari e Piergiorgio Lochner. «Automatic optic nerve measurement: a new tool to standardize optic nerve assessment in ultrasound b-mode images». In: *Ultrasound in medicine & biology* 46.6 (2020), pp. 1533–1544 (cit. alle pp. 18–21, 68, 77).
- [30] Venkatakrishna Rajajee, Reza Soroushmehr, Craig A Williamson, Kayvan Najarian, Jonathan Gryak, Abdelrahman Awad, Kevin R Ward e Mohamad H Tiba. «Novel Algorithm for Automated Optic Nerve Sheath Diameter Measurement Using a Clustering Approach». In: *Military Medicine* 186.Supplement\_1 (2021), pp. 496–501 (cit. alle pp. 21, 22).
- [31] Mike Wolfe. *Deep Learning nella scienza dei dati*. 2021. URL: <https://towardsdatascience.com/deep-learning-in-data-science-f34b4b124580> (cit. a p. 23).
- [32] Anne Bonner. *The Complete Beginners Guide to Deep Learning*. 2019. URL: <https://towardsdatascience.com/intro-to-deep-learning-c025efd92535> (cit. a p. 23).
- [33] James Liang. *An Introduction to Deep Learning*. 2018. URL: <https://towardsdatascience.com/an-introduction-to-deep-learning-af63448c122c> (cit. alle pp. 23, 24).
- [34] REDAZIONE OSSERVATORI DIGITAL INNOVATION. *Computer Vision: definizione, funzionamento e applicazioni*. 2021. URL: [https://blog.osservatori.net/it\\_it/computer-vision-definizione-applicazioni](https://blog.osservatori.net/it_it/computer-vision-definizione-applicazioni) (cit. a p. 24).
- [35] Lorenzo Govoni. *Computer Vision: come una macchina riconosce gli oggetti*. 2021. URL: <https://www.lorenzogovoni.com/computer-vision/> (cit. a p. 24).
- [36] Stanford CS class CS231n. *Convolutional Neural Networks for Visual Recognition*. URL: <https://cs231n.github.io/convolutional-networks/#conv> (cit. alle pp. 25–27).
- [37] Sumit Saha. *A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way*. 2018. URL: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53> (cit. a p. 25).
- [38] Lorenzo Govoni. *Semplice architettura di rete neurale convoluzionale*. 2021. URL: <https://www.lorenzogovoni.com/architettura-di-rete-neurale-convoluzionale/> (cit. a p. 26).

- 
- [39] Vincent Tatan. *Understanding CNN (Convolutional Neural Network)*. 2019. URL: <https://towardsdatascience.com/understanding-cnn-convolutional-neural-network-69fd626ee7d4> (cit. a p. 27).
- [40] Shruti Jadon. *Introduction to Different Activation Functions for Deep Learning*. 2018. URL: <https://medium.com/@shrutijadon10104776/survey-on-activation-functions-for-deep-learning-9689331ba092> (cit. a p. 28).
- [41] Pedro Marcelino. *Transfer learning from pre-trained models*. 2018. URL: <https://towardsdatascience.com/transfer-learning-from-pre-trained-models-f2393f124751> (cit. a p. 29).
- [42] Nikolas Adaloglou. *Best deep CNN architectures and their principles: from AlexNet to EfficientNet*. 2021. URL: <https://theaisummer.com/cnn-architectures/> (cit. alle pp. 29, 30).
- [43] Alex Krizhevsky, Ilya Sutskever e Geoffrey E Hinton. «Imagenet classification with deep convolutional neural networks». In: *Advances in neural information processing systems* 25 (2012), pp. 1097–1105 (cit. a p. 29).
- [44] Raimi Karim. *Illustrated: 10 CNN Architectures*. 2019. URL: <https://towardsdatascience.com/illustrated-10-cnn-architectures-95d78ace614d#e4b1> (cit. a p. 30).
- [45] Karen Simonyan e Andrew Zisserman. «Very deep convolutional networks for large-scale image recognition». In: *arXiv preprint arXiv:1409.1556* (2014) (cit. a p. 30).
- [46] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke e Andrew Rabinovich. «Going deeper with convolutions». In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1–9 (cit. a p. 30).
- [47] Kaiming He, Xiangyu Zhang, Shaoqing Ren e Jian Sun. «Deep residual learning for image recognition». In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778 (cit. alle pp. 31–33).
- [48] Kaiming He e Jian Sun. «Convolutional Neural Networks at Constrained Time Cost». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Giu. 2015 (cit. a p. 31).
- [49] Rupesh Kumar Srivastava, Klaus Greff e Jürgen Schmidhuber. «Highway networks». In: *arXiv preprint arXiv:1505.00387* (2015) (cit. a p. 31).
- [50] Giuseppe Pio Cannata. *Additive skip connection*. 2020. URL: <https://cannydatascience.medium.com/additive-skip-connection-222b45a6a0e2> (cit. a p. 32).

- 
- [51] Mingxing Tan e Quoc Le. «Efficientnet: Rethinking model scaling for convolutional neural networks». In: *International Conference on Machine Learning*. PMLR. 2019, pp. 6105–6114 (cit. alle pp. 33–35).
- [52] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov e Liang-Chieh Chen. «Mobilenetv2: Inverted residuals and linear bottlenecks». In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 4510–4520 (cit. a p. 35).
- [53] Luis Gonzales. *A Look at MobileNetV2: Inverted Residuals and Linear Bottlenecks*. 2019. URL: [https://medium.com/@luis\\_gonzales/a-look-at-mobilenetv2-inverted-residuals-and-linear-bottlenecks-d49f85c12423](https://medium.com/@luis_gonzales/a-look-at-mobilenetv2-inverted-residuals-and-linear-bottlenecks-d49f85c12423) (cit. a p. 35).
- [54] Francesco Zuppichini. *Residual, BottleNeck, Inverted Residual, Linear BottleNeck, MBConv Explained*. 2021. URL: <https://towardsdatascience.com/residual-bottleneck-inverted-residual-linear-bottleneck-mbconv-explained-89d7b7e7c6bc> (cit. a p. 36).
- [55] Harshall Lamba. *Understanding Semantic Segmentation with UNET*. 2019. URL: <https://towardsdatascience.com/understanding-semantic-segmentation-with-unet-6be4f42d4b47> (cit. a p. 36).
- [56] Dive into Deep Learning. *Transposed Convolution*. URL: [https://d2l.ai/chapter\\_computer-vision/transposed-conv.html](https://d2l.ai/chapter_computer-vision/transposed-conv.html) (cit. a p. 37).
- [57] Navdeep Singh. *Machine Learning Algorithms for semantic segmentation*. 2018. URL: <https://codeallogic.org/2018/05/30/machine-learning-algorithms-for-semantic-segmentation/> (cit. a p. 37).
- [58] Olaf Ronneberger, Philipp Fischer e Thomas Brox. «U-net: Convolutional networks for biomedical image segmentation». In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241 (cit. alle pp. 37, 38, 55).
- [59] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh e Jianming Liang. «Unet++: A nested u-net architecture for medical image segmentation». In: *Deep learning in medical image analysis and multimodal learning for clinical decision support*. Springer, 2018, pp. 3–11 (cit. alle pp. 38–41).
- [60] Tongle Fan, Guanglei Wang, Yan Li e Hongrui Wang. «Ma-net: A multi-scale attention network for liver and tumor segmentation». In: *IEEE Access* 8 (2020), pp. 179656–179665 (cit. alle pp. 41, 42).
- [61] David Chuan-En Lin. *8 Simple Techniques to Prevent Overfitting*. 2020. URL: <https://towardsdatascience.com/8-simple-techniques-to-prevent-overfitting-4d443da2ef7d> (cit. a p. 42).

- [62] Pavel Yakubovskiy. *Segmentation Models Pytorch*. [https://github.com/qubvel/segmentation\\_models.pytorch](https://github.com/qubvel/segmentation_models.pytorch). 2020 (cit. a p. 53).
- [63] Diederik P. Kingma e Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG] (cit. a p. 56).
- [64] Vitaly Bushaev. *Adam — latest trends in deep learning optimization*. 2018. URL: <https://towardsdatascience.com/adam-latest-trends-in-deep-learning-optimization-6be9a291375c> (cit. a p. 56).
- [65] Shruti Jadon. «A survey of loss functions for semantic segmentation». In: *2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*. IEEE. 2020, pp. 1–7 (cit. a p. 56).
- [66] PulpLearning. *Cross Validation: cos'è e come usarla nel Machine Learning*. 2021. URL: <https://pulplearning.altervista.org/cross-validation-cose-e-come-usarla/> (cit. a p. 57).
- [67] MathWorks. *Help Center - corrcoeff*. URL: <https://www.mathworks.com/help/matlab/ref/corrcoef.html> (cit. a p. 69).