

# POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Biomedica



**Politecnico  
di Torino**

Tesi di Laurea Magistrale

## Segmentazione e caratterizzazione di volumi MRI di arto inferiore tramite Deep Learning e Texture Analysis per il riconoscimento di distrofie muscolari

Relatore  
Prof.ssa KRISTEN MARIKO MEIBURGER  
Co-relatore  
Ing. FRANCESCO MARZOLA

Candidata  
EVELIN OMOBONO  
Matricola 282783

A.A 2022/2023  
Sessione di Laurea Marzo 2023



# Abstract

Le distrofie muscolari rappresentano una famiglia molto eterogenea di malattie ereditarie. Sono differenti per modalità di trasmissione, progressione, coinvolgimento sistemico e muscolare. Sebbene l'espressione fenotipica possa variare considerevolmente tra i pazienti, per alcune di esse sono individuabili pattern precisi da cui prendono il nome; è il caso della distrofia facio-scapolo-omerale (FSHD), che viene descritta in questo lavoro insieme alla distrofia miotonica di tipo 1 (MD1).

Le alterazioni del tessuto muscolare sono visibili dalle immagini MRI, e possono essere quantificate. Prima di calcolare tali parametri, legati alla progressione o al tipo di distrofia, bisogna segmentare i muscoli, per circoscrivere la regione di interesse. La segmentazione manuale è il gold standard ma impiega molte risorse temporali; dunque, sono stati valutati diversi approcci di Deep Learning come possibile alternativa.

Il Dataset iniziale comprendeva 216 volumi MRI raffiguranti gli arti inferiori di 59 soggetti, caratterizzati da tre condizioni muscolari diverse (25 soggetti FSHD, 24 soggetti MD1, 10 Healthy Volunteer). Sono stati ricavati due dataset, trattati separatamente: uno per le cosce (Training Set (TR) = 86 volumi; Validation Set (VL) = 16 volumi; Test Set (TS) = 5 volumi) e l'altro per le gambe (TR = 88 volumi; VL = 16 volumi; TS = 5 volumi).

Poiché distrofie diverse colpiscono muscoli diversi, lo scopo di questo lavoro è di segmentare individualmente i 12 muscoli della coscia e i 9 della gamba. Per questo task, sono state valutate 5 Deep Learning Neural Network differenti per: tipologia di segmentazione (2D, 3D), tipologia di encoder (ResNet, SwinTransformer) e tipologia di architettura (U-Net, K-Net). Nel caso della coscia, la rete con le migliori performance è stata una U-Net (2D) con encoder ResNet50 (Dice Coefficient =  $81.75 \pm 6.6$  %; Hausdorff Distance (95° percentile) =  $4.28 \pm 1.94$  voxel) ; mentre per la gamba, è risultata la K-Net con encoder SwinTransformer (Dice Coefficient =  $79.55 \pm 12.14$  %; Hausdorff Distance (95° percentile) =  $3.41 \pm 1.41$  voxel).

Nella seconda parte del lavoro, è stato dimostrato che la Texture Analysis è utile per discriminare le condizioni muscolari presenti nel dataset. Dopo aver estratto 86 texture feature, tra quelle del primo e del secondo ordine, è stata eseguita la feature selection attraverso il supporto di due test statistici: il t-test e il Mann-Whitney U test. Sono state mantenute le feature che evidenziassero una differenza statisticamente significativa tra le distrofie MD1 e FSHD. Attraverso la Principal Component Analysis (PCA), è stata ridotta la dimensionalità del set di feature, in modo da visualizzarne la distribuzione per ogni muscolo.

I risultati ottenuti sono promettenti. Tra le reti neurali implementate, quelle più recenti hanno mostrato performance competitive rispetto alle architetture più note, arrivando a superarle in alcuni casi. La Texture Analysis, si è rivelata utile per differenziare le distrofie muscolari. In fine, si possono individuare diversi sviluppi futuri: dal supporto ai clinici con la segmentazione automatica, che risulta valida anche su dati complessi come in questo caso, alla classificazione delle distrofie muscolari attraverso le texture feature.



# INDICE

Abstract.....	3
1. Le Distrofie Muscolari .....	8
1.1 Distrofia Miotonica di tipo 1 .....	9
1.2 Distrofia facio-scapolo-omeroale .....	10
2. La Risonanza Magnetica.....	13
2.1 Principio fisico.....	13
2.2 Tecniche di acquisizione .....	15
2.3 Risonanza Magnetica e distrofie muscolari.....	15
3. Deep Learning.....	19
3.1 Artificial Neural Network (ANN) .....	19
3.2 Deep Learning Neural Network.....	20
4. Stato dell'Arte .....	23
4.1 Convolutional Neural Network (CNN) .....	23
4.1.1 Convolutional Layer .....	23
4.1.2 Funzione di attivazione .....	24
4.1.3 Pooling Layer.....	25
4.1.4 Fully Connected Layer .....	25
4.2 ResNet.....	25
4.3 Vision Transformer.....	26
4.3.1 Multihead Self-Attention .....	27
4.3.2 Swin Transformer .....	28
4.4 Tipologie di Architetture .....	28
4.4.1 U-Net.....	28
4.4.2 K-Net .....	30
5. Dataset.....	32
5.1 Stato iniziale del Dataset.....	34
5.2 Riassegnazione manuale delle label.....	35
5.2.1 Label della Coscia.....	35
5.2.2 Label della Gamba .....	35
5.3 Pre-processing Dataset 3D.....	36
5.3.1 Flip: da "L" a "R" .....	36
5.3.2 Histogram Matching.....	37
5.3.3 Ricampionamento.....	40
5.4 Creazione del Dataset 2D .....	40
6. Segmentazione multi-classe .....	43
6.1 Training 3D.....	44

6.1.1 Training Set Transform .....	44
6.1.2 Inference Transform .....	45
6.1.3 Dettagli di implementazione .....	45
6.1.4 Descrizione dei modelli.....	46
6.2 Training 2D.....	46
6.2.1 Training Transform .....	46
6.2.2 Inference Transform .....	47
6.2.3 Dettagli di implementazione .....	47
6.2.4 Descrizione dei modelli.....	48
6.3 Inference.....	48
6.3.1 Inference 3D .....	48
6.3.2 Inference 2D .....	49
6.4 Post-processing.....	49
6.5 Metriche .....	51
<b>7. Texture Analysis.....</b>	<b>54</b>
7.1 Analisi Preliminare e Feature Extraction .....	56
7.2 Analisi sui singoli muscoli: Vasto Laterale .....	58
7.2.1 t-test .....	59
7.2.2 Test U di Mann-Whitney .....	61
7.2.3 Dimensionality reduction: PCA.....	61
7.2.4 Visualizzazione .....	63
<b>8. Risultati.....</b>	<b>65</b>
8.1 Confronto Deep Learning Neural Network .....	65
8.1.1 Confronto parziale (2D).....	65
8.1.2 Confronto completo (3D) .....	66
8.1.3 Considerazioni sui singoli muscoli .....	68
8.2 Texture Analysis .....	69
8.2.1 L'importanza del pre-processing.....	71
<b>9. Discussione .....</b>	<b>73</b>
<b>10. Conclusioni e sviluppi futuri .....</b>	<b>76</b>
<b>Appendice A.....</b>	<b>78</b>
<b>Risultati della Texture Analysis su tutti i muscoli .....</b>	<b>78</b>
Coscia t-test .....	78
Coscia test U di Mann-Whitney .....	81
Gamba t-test.....	84
Gamba test U di Mann-Whitney .....	87
<b>BIBLIOGRAFIA .....</b>	<b>91</b>



# Capitolo 1

## Le Distrofie Muscolari

Le Distrofie muscolari sono malattie genetiche, nella maggior parte dei casi ereditarie, che portano ad un progressivo indebolimento dei muscoli sfociando in una degenerazione sistemica che compromette la durata e la qualità di vita dei pazienti.

Possono essere trasmesse con modello di ereditarietà autosomico<sup>1</sup> dominante, autosomico recessivo o legato al cromosoma X :

- Per trasmissione autosomica dominante si intende che basta uno dei due alleli mutato (indifferentemente proveniente dal padre o dalla madre) affinché il figlio esprima la malattia. Le distrofie **miotonica di tipo 1** e **facio-scapolo-omerale** sono malattie a trasmissione autosomica dominante;
- Affinché una malattia autosomica recessiva venga espressa dal figlio, è necessario che sia l'allele trasmesso dal padre che quello trasmesso dalla madre siano mutati. La distrofia muscolare dei cingoli rientra nel modello a trasmissione autosomica recessiva;
- In una malattia ereditaria recessiva legata ad una mutazione del cromosoma X, gli individui di sesso maschile (cromosomi sessuali XY) saranno sempre malati, mentre gli individui di sesso femminile (cromosomi sessuali XX) esprimeranno la malattia solo se entrambi gli alleli sono mutati, negli altri casi invece saranno "portatori sani". La distrofia muscolare di Duchenne rientra in questa tipologia.

Le mutazioni genetiche dovute dalle distrofie causano, di solito, un'anomalia nella produzione di alcune proteine della fibra muscolare alterandone il funzionamento e la composizione.

La famiglia delle distrofie muscolari è piuttosto ampia e comprende patologie molto diverse tra loro, anche all'interno dello stesso sottogruppo. Alcune si manifestano in modo abbastanza caratteristico, mentre, per altre, non si riesce a individuare fin da subito un quadro sintomatico ben preciso, rendendo difficile la diagnosi [1]. Le distrofie muscolari possono insorgere a qualsiasi età, e avere un decorso più o meno lento. Sebbene il test genetico sia il gold standard per la diagnosi, esso è spesso affiancato da un'anamnesi familiare approfondita, tecniche di imaging come la risonanza magnetica e dalla biopsia muscolare[1]. Con la risonanza magnetica, ad esempio, si può restringere il campo dei test genetici, oppure, è possibile trovare il sito migliore per eseguire la biopsia muscolare [2].

Le distrofie muscolari possono essere classificate in base ai muscoli principalmente interessati (come nel caso della distrofia facio-scapolo-omerale), al decorso della malattia o alla mutazione genetica.

Tra gli adulti, la distrofia miotonica di tipo 1 è la più diffusa, con un'insorgenza di 1 su 8000 nati sani [1], seguita dalla distrofia facio-scapolo-omerale con una prevalenza di 1 su 20000 soggetti [3].

---

<sup>1</sup> Con il termine "autosomico" si fa riferimento ai cromosomi non sessuali.



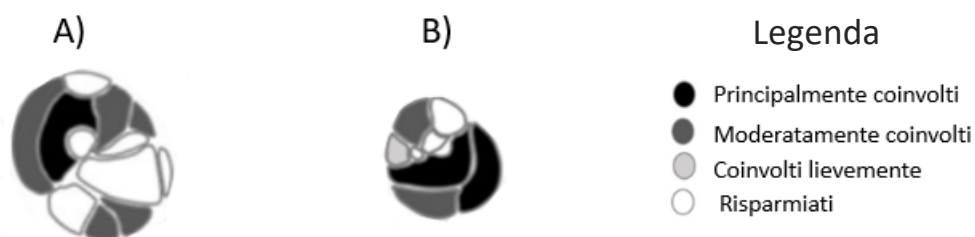
## 1.1 Distrofia Miotonica di tipo 1

La distrofia miotonica di tipo 1 (anche detta malattia di Steinert) è dovuta alla mutazione del gene DMPK (Myotonic Dystrophy Protein Kinase). Esso codifica per una proteina-chinasi della miosina ed è situato nel cromosoma 19q<sup>2</sup>. La mutazione consiste in un'amplificazione della tripletta nucleotidica CTG: se in un soggetto sano essa è ripetuta fino a 49 volte [1], in un soggetto affetto da MD1, può essere ripetuta circa 50 - 1000 volte. La gravità del quadro clinico del paziente, in termini di coinvolgimento sistemico, dipende dal numero di ripetizioni della sequenza CTG. La patologia presenta il fenomeno dell'anticipazione genica, ovvero, di generazione in generazione, la malattia insorge prima e in una forma più grave. La distrofia miotonica di tipo 1 può manifestarsi a qualsiasi età, e in base a questo è possibile individuare 4 sottogruppi [4]:

- Congenita: si presenta alla nascita;
- ad esordio infantile: si presenta tra 1 e 20 anni;
- ad esordio adulto o classica: si presenta tra 21 e 40 anni;
- ad esordio tardivo o asintomatica: si presenta ad un'età superiore ai 40 anni.

Per miotonia si intende un difetto nella normale funzionalità del muscolo, che gli impedisce di rilassarsi subito dopo una contrazione, mantenendolo in uno stato di rigidità.

La forma più critica della malattia di Steinert è quella congenita, che comporta anche problemi cardiaci, respiratori e deficit cognitivi. La malattia ad esordio adulto è anche detta "classica" perché è la forma più comune. Progredisce lentamente e tra i sintomi principali presenta insufficienza respiratoria, difetti di conduzione (aritmia), cataratta, disfunzioni endocrine (ipogonadismo o diabete). Tipicamente la malattia colpisce i muscoli distali. Possono essere interessati i muscoli facciali e i muscoli sternocleidomastoidei. I muscoli più colpiti degli arti superiori sono i flessori delle dita e del polso, ma anche i tricipiti, e i trapezi [5]. Per quanto riguarda i muscoli degli arti inferiori, nelle prime fasi della malattia vengono interessati il Vasto laterale, mediale e intermedio, nella coscia; il soleo e il gastrocnemio mediale, nella gamba (figura 1.1). Essendo interessati per primi, risulteranno più compromessi con il passare del tempo. Anche il tibiale anteriore e il semimembranoso vengono coinvolti [5].



**Figura 1.1:** Raffigurazione del pattern di coinvolgimento muscolare della distrofia miotonica di tipo 1 [6]. A) Coscia: vasto laterale; vasto mediale e vasto intermedio, semimembranoso e semitendinoso vengono coinvolti maggiormente. Grande adduttore, adduttore breve, bicipite femorale – capo lungo, gracile e retto femorale vengono risparmiati. B) Gamba: soleo, gastrocnemio mediale e gastrocnemio laterale vengono colpiti maggiormente.

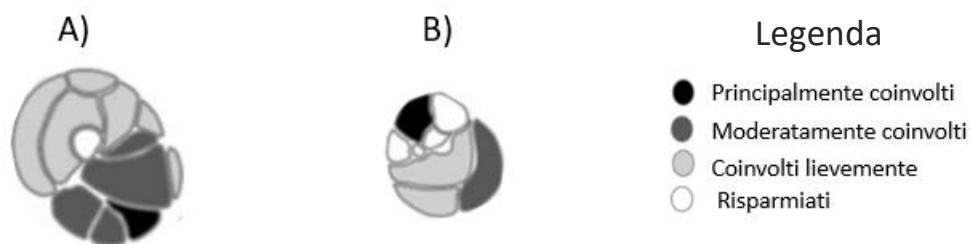
<sup>2</sup> La mutazione si trova nella porzione più lunga del cromosoma 19, indicata con "q" che sta per "queue".

## 1.2 Distrofia facio-scapolo-omerale

La distrofia facio-scapolo-omerale (FSHD) è causata dalla mutazione del gene D4Z4, situato nel cromosoma 4q. In un soggetto sano, la presenza di più di 10 unità ripetitive D4Z4 inibisce la sintesi della proteina DUX4, tossica per il muscolo. In un soggetto affetto da FSHD, invece, il numero ridotto di D4Z4 (meno di 10 ripetizioni) ne permette l'espressione. Minore è il numero di unità di D4Z4 residue, maggiore è la gravità della patologia, in termini di quadro clinico ed entità della debolezza muscolare [7].

La malattia può presentarsi a qualsiasi età e in modi differenti, anche tra i membri della stessa famiglia; prima si manifesta la malattia, maggiore sarà la gravità. Il pattern dei muscoli coinvolti è abbastanza riconoscibile, tanto da dare il nome alla patologia, tuttavia il modo in cui viene espresso varia da individuo a individuo [2].

Tipicamente, i pazienti presentano debolezza ai muscoli del volto e del cingolo scapolare. Questo rende loro difficile esprimere emozioni attraverso la mimica facciale, tenere gli occhi chiusi durante il sonno o alzare le braccia. Tra i muscoli del cingolo scapolare, la FSHD colpisce inizialmente il trapezio e il dentato anteriore [4], causando la così detta "scapola alata" che spesso porta a interventi chirurgici di fissaggio. Anche il tricipite e il bicipite vengono colpiti in maniera predominante nelle prime fasi. La malattia è caratterizzata da una debolezza muscolare asimmetrica, ovvero possono essere colpiti i muscoli solo da un lato del corpo, e la degenerazione progressiva procede in maniera discendente. Sono interessati anche i muscoli spinali, addominali e del pavimento pelvico [6]. Per quanto riguarda i muscoli degli arti inferiori, vengono coinvolti per primi e in modo preferenziale il tibiale anteriore e il gastrocnemio mediale; negli stadi più avanzati vengono coinvolti anche i muscoli posteriori della coscia ( bicipite femorale, semimembranoso e semitendinoso) e gli adduttori, come mostrato in figura 1.2.



**Figura 1.2:** Raffigurazione del pattern di coinvolgimento muscolare della distrofia facio-scapolo-omerale [6]. A) Coscia: grande adduttore, adduttore breve, bicipite femorale – capo lungo, semimembranoso e semitendinoso vengono coinvolti maggiormente. B) Gamba: gastrocnemio mediale e tibiale anteriore vengono colpiti maggiormente.

Quando la malattia si presenta nella forma più grave, i pazienti potrebbero aver bisogno della sedia a rotelle entro i primi dieci anni di vita; negli altri casi, la probabilità aumenta con l'età. La malattia ha una progressione lenta e raramente vengono interessati altri distretti oltre la muscolatura. Solo nei casi più gravi, che presentano pochissimi D4Z4 residui (da 1 a 3), si può manifestare un coinvolgimento sistemico che riguarda problemi respiratori, perdita dell'udito, vasculopatie retiniche. Le aritmie, laddove presenti, sono raramente sintomatiche. Per diagnosticare la malattia

ci si affida al test genetico, anche se una ridotta percentuale di popolazione sana (3-5%) presenta meno di 10 ripetizioni di D4Z4 [8]. Per questo motivo, è sempre più importante ricorrere ad altri esami di supporto. Ad esempio, poiché il pattern di coinvolgimento muscolare è piuttosto riconoscibile, risulta interessante il ruolo della risonanza magnetica come esame investigativo sull'efficacia delle terapie o per il monitoraggio della malattia [6],[8].

Tabella 1.1 : Confronto tra la distrofia miotonica di tipo 1 (MD1) e la distrofia facio-scapolo-omerale (FSHD).

CONFRONTO DISTROFIE		
	MD1	FSHD
<b>Mutazione genetica</b>	Amplificazione unità ripetitive di DMPK	Delezione unità ripetitive di D4Z4
<b>Sede della mutazione</b>	Cromosoma 19	Cromosoma 4
<b>Anticipazione genetica</b>	Sì	Non è stata documentata
<b>Età di insorgenza</b>	Qualsiasi (0-80 anni)	Qualsiasi (più probabile entro la prima o la seconda decade di vita)
<b>Progressione</b>	Lenta (1-3% l'anno [5])	Lenta (1-4% l'anno [9])
<b>Coinvolgimento sistemico</b>	Sì, è tipico della malattia	Raro, solo nei casi più gravi (colpisce retina e udito)
<b>Problemi cardiorespiratori</b>	Presenti nella maggior parte dei soggetti. Problemi respiratori e difetti di conduzione (aritmie)	Presenti solo nei casi più gravi. Problemi respiratori nei soggetti in sedia a rotelle, aritmie raramente sintomatiche
<b>Caratteristiche della progressione</b>	Tipicamente distale-prossimale	Tipicamente discendente , marcatamente asimmetrica
<b>Muscolatura coinvolta in maniera preferenziale (arti inferiori)</b>	<i>COSCIA</i> : Muscoli anteriori della coscia, Vasti (Laterale,Mediale,Intermedio) <i>GAMBA</i> : Soleo, Gastrocnemio Mediale	<i>COSCIA</i> : Muscoli posteriori della coscia (Semimembranoso, Semitendinoso e Bicipite femorale), Adduttori <i>GAMBA</i> : Tibiale Anteriore , Gastrocnemio Mediale



## Capitolo 2

### La Risonanza Magnetica

#### 2.1 Principio fisico

La risonanza magnetica nucleare è un fenomeno fisico che riguarda gli elementi con spin non nullo che, in presenza di un campo magnetico esterno, vengono eccitati e liberano energia.

Lo spin ( $I$ ) è una proprietà delle particelle e descrive le caratteristiche del moto che compiono attorno al proprio asse. È definito come segue:

$$I = (\text{protoni spaiati}) \frac{1}{2} + \text{neutroni spaiati} \quad (2.1)$$

Gli elementi che non hanno protoni e neutroni spaiati, hanno spin nullo e non risentono del fenomeno della risonanza magnetica.

Per i nuclei dotati di spin è possibile definire anche un momento angolare  $I$ , il cui modulo è definito dall'equazione (2.2):

$$|I| = \frac{h}{2\pi} \sqrt{I(I+1)} \quad (2.2)$$

Con  $h$  = costante di Planck.

Tra tutti gli elementi con spin non nullo, quello che interessa maggiormente nell'imaging medico è l'idrogeno ( $I = \frac{1}{2}$ ), molto abbondante nel nostro corpo, soprattutto nei tessuti molli.

Poiché il nucleo è dotato di carica e ruota attorno al proprio asse, è anche dotato di momento magnetico  $\mu$ , definito dall'equazione (2.3):

$$\mu = \gamma I \quad (2.3)$$

Con  $\gamma$  = rapporto giromagnetico.

Il vettore magnetizzazione  $M$  è definito come la somma dei momenti magnetici di tutti i protoni idrogeno (equazione (2.4)):

$$M = \sum_i \mu_i \quad (2.4)$$

Se non c'è alcun campo magnetico esterno applicato, il vettore di magnetizzazione è nullo, perché i momenti magnetici hanno un'orientazione casuale e si semplificano a vicenda. Appena è presente

un campo magnetico esterno statico ( $\mathbf{B}_0 = \hat{z}B_0$ ), però, i momenti magnetici si orienteranno lungo la sua stessa direzione, con due possibili orientazioni (2 livelli energetici<sup>3</sup>), dette:

- Spin-up, se sono orientati come  $\mathbf{B}_0$ ;
- Spin-down, se sono orientati in maniera opposta a  $\mathbf{B}_0$ .

A causa della presenza di una forzante esterna e del fatto che ruotano attorno al proprio asse, i protoni seguono il moto di precessione. La velocità angolare con cui descrivono questa traiettoria attorno alla direzione del campo  $\mathbf{B}_0$  è detta frequenza di Larmor ed è espressa dall'equazione (2.5):

$$\omega_0 = \gamma B_0 \quad (2.5)$$

Il vettore magnetizzazione, è diverso da zero adesso, e sarà pari alla somma vettoriale di tutti i momenti magnetici. L'intensità, dipende dalla prevalenza di un'orientazione rispetto all'altra. Tuttavia, la prevalenza naturale di spin-up rispetto a spin-down è piuttosto bassa (equazione (2.6)):

$$\frac{n(up) - n(down)}{n(up) + n(down)} = 3,3 * 10^{-6} \quad (2.6)$$

Ovvero, su un milione di protoni, c'è una prevalenza di circa 3 protoni spin-up.

Per amplificare la risposta del tessuto, allora, si manda il sistema in risonanza (da qui il nome risonanza magnetica). Insieme al campo magnetico statico  $\mathbf{B}_0$ , viene applicato un campo magnetico variabile  $\mathbf{B}_1$ , che oscilla nel piano perpendicolare alla direzione di applicazione di  $\mathbf{B}_0$ , alla frequenza di risonanza del sistema, ovvero alla frequenza di Larmor, che appartiene al range delle radiofrequenze<sup>4</sup>.

Considerando un sistema di assi cartesiani, il campo magnetico totale che agisce sui nuclei è dato dall'equazione (2.7) :

$$\mathbf{B} = \hat{z}B_0 + B_1(\hat{x}\cos\omega t + \hat{y}\sin\omega t) \quad (2.7)$$

Fornendo energia al sistema alcuni protoni vengono promossi allo stato energetico superiore (orientamento spin-down). Eccitando abbastanza a lungo il sistema, vedremo che la prevalenza sarà annullata (vettore magnetizzazione ruota di 90°) e poi invertita (angolo di flip pari a 180°).

A questo punto, se il tessuto non viene più eccitato, i protoni ritorneranno nelle condizioni di partenza, rilasciando una quantità di energia sotto forma di radiazione elettromagnetica, pari alla differenza tra i livelli energetici. Il segnale misurato è detto Free Induction Decay (FID) [10], la cui ampiezza si riduce come se fosse un'esponenziale decrescente. Il tempo impiegato dal sistema per tornare nelle condizioni iniziali, che viene estrapolato dal segnale FID, è detto tempo di rilassamento. È proprio quest'ultimo che differenzia i tessuti e viene utilizzato per generare l'immagine MRI.

<sup>3</sup> Livelli energetici =  $2I + 1$ ; quindi l'idrogeno ( $I = \frac{1}{2}$ ), ha 2 livelli energetici.

<sup>4</sup> Considerando:  $B_0 = 1 \text{ T}$ ;  $\gamma = 42 \frac{\text{MHz}}{\text{T}}$ ; allora la frequenza dell'eccitazione sarà pari a  $\omega_0 = 42 \text{ MHz}$ .

## 2.2 Tecniche di acquisizione

Per creare il contrasto nell'immagine, la MRI sfrutta le caratteristiche intrinseche del tessuto , ovvero i tempi di rilassamento  $T1^5$  e  $T2^6$  e la densità protonica (DP). A seconda di quale proprietà viene sfruttata, si utilizza un'appropriata sequenza di acquisizione. Le principali tipologie di immagini RMI sono:

- T1-pesate. Utili per studiare il tessuto adiposo [11], che appare iperintenso, o per rivelare infiltrazioni di grasso nei muscoli, tratto distintivo della degenerazione causata dalla distrofia;
- T2-pesate. Utili per evidenziare fluidi o la presenza di edema nei tessuti, che si manifesta in alcune miopatie infiammatorie [12];
- DP-pesate. Questo tipo di immagini ha un contrasto inferiore rispetto a quelle pesate in T1 o in T2, può essere utile nell'imaging di cosce e gambe [13] ed è prevalentemente utilizzato nello studio delle articolazioni [14].

Oltre alle principali citate sopra, altre sequenze di acquisizione mettono in risalto ulteriori aspetti delle immagini RM. Ad esempio, una delle tecniche più comuni per la saturazione del grasso è la Short Tau Inversion Recovery (STIR), che è in grado di evidenziare una compromissione muscolare prima che si presentino infiltrazioni di grasso visibili in soggetti affetti da MD1 [4]. Questo rende la risonanza magnetica sensibile anche alle manifestazioni precoci della distrofia. Un altro studio condotto su pazienti con FSHD ha mostrato come la sequenza TIRM (Turbo Inversion Recovery Magnitude) può essere utile nella scelta del sito più adatto per eseguire la biopsia muscolare [15].

In fine, una delle tecniche più promettenti, per le sue applicazioni in ambito clinico, è la Dixon, che sfrutta la differenza nella frequenza di risonanza dei protoni che si trovano nell'acqua e nel grasso dei tessuti biologici. Attraverso due sequenze spin-echo che differiscono per il tempo di eco (TE), si acquisiscono due immagini iniziali: una per cui i protoni nell'acqua e nel grasso sono in fase (in phase – IP) e una per cui si trovano in opposizione di fase (OP). Combinando opportunamente queste due immagini, se ne ottengono altre due: una che evidenzia l'acqua e l'altra che evidenzia il grasso nei tessuti [16]. Pertanto, con questa tecnica di acquisizione si producono 4 immagini ad ogni scansione. L'uso della soppressione e della quantificazione del grasso mediante la tecnica Dixon è un campo di ricerca vivo, soprattutto nell'ambito delle distrofie muscolari [17]. Rispetto ad altri approcci, la tecnica Dixon ha il vantaggio di essere più robusta alla disomogeneità del campo magnetico e permette di sopprimere il grasso dalle immagini MRI anche in fase di post-elaborazione [18].

## 2.3 Risonanza Magnetica e distrofie muscolari

La Risonanza Magnetica Muscoloscheletrica, riguarda le articolazioni e tutte le strutture ad esse annessi e offre un grande supporto per la diagnosi, lo studio, il monitoraggio delle distrofie

---

<sup>5</sup> Tempo di rilassamento spin-reticolo.

<sup>6</sup> Tempo di rilassamento spin-spin.

muscolari [19]; inoltre, non è una tecnica invasiva (a differenza di una biopsia muscolare) ed è sicura per i pazienti perché non utilizza radiazioni ionizzanti.

Grazie all'elevata risoluzione spaziale che permette di riconoscere dettagli anatomici meglio di altre tecniche di imaging [19], e alle acquisizioni multi planari, è possibile avere una visione completa della progressione della malattia e della modalità di coinvolgimento dei singoli muscoli [4]. Distrofie come MD1 e FSHD, colpiscono selettivamente i muscoli, mostrando pattern diversi. Individuarli e riconoscerli, dà un supporto importantissimo alla diagnosi, e può aiutare a restringere il campo dei test genetici utili [19].

Le immagini MRI sono un ottimo punto di partenza per successive analisi. Ad esempio, si possono estrarre dei parametri quantitativi correlabili alla gravità della distrofia [20],[15] o feature utili a distinguerne gli effetti attraverso la Texture Analysis come verrà approfondito più avanti in questo lavoro.

Per farlo, è necessario segmentare i muscoli, ovvero circoscrivere la regione di spazio che occupano all'interno dell'immagine. La segmentazione manuale è il gold standard attualmente, tuttavia, soffre dei limiti comuni a tutte le tecniche manuali ovvero:

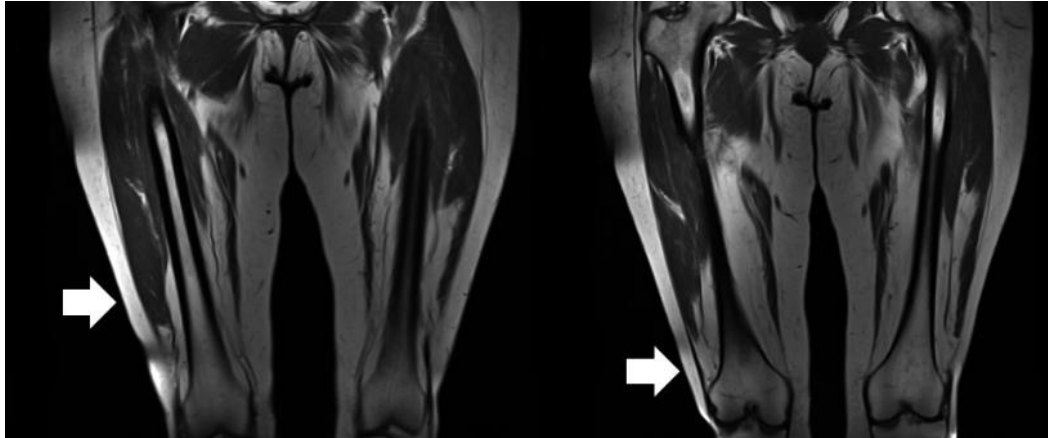
- È operatore dipendente. Il risultato dipende dall'esperienza del clinico;
- Variabilità inter e intra-operatore. Alla stessa immagine saranno associati ground truth differenti;
- Risorse temporali richieste.

Il limite più grande è sicuramente il tempo. Basti pensare che le immagini RM sono dei volumi composti da un numero variabile di slice, in base all'estensione della regione anatomica, e bisognerebbe segmentare ciascun muscolo su ognuna di esse, ripetendo poi l'operazione per più pazienti.

Per aggirare il problema, si sceglie una sezione considerata più rappresentativa, e si segmentano solo i muscoli più colpiti di quella slice [15], oppure si prendono solo alcune slices disseminate lungo il decorso del muscolo [20]

È chiaro come questa procedura, sebbene riduca i tempi, causi anche una perdita di informazioni non indifferente. Infatti, essa è legata ad una decisione da parte del clinico che deve essere specifica per ogni paziente, vista l'elevata variabilità fenotipica che contraddistingue le distrofie muscolari. Infatti, le manifestazioni di queste malattie, non sono solo diverse da soggetto a soggetto, ma anche lungo lo stesso muscolo: come mostrato in figura 2.1, la MD1 mostra un coinvolgimento distale-proximale e porzioni diverse dei muscoli, sono compromesse in maniera diversa.





**Figura 2.1:** Risonanza magnetica T1-pesata che mostra il coinvolgimento distale-proximale del Quadricipite (Vasti) in un paziente affetto da distrofia miotonica di tipo 1 [4].

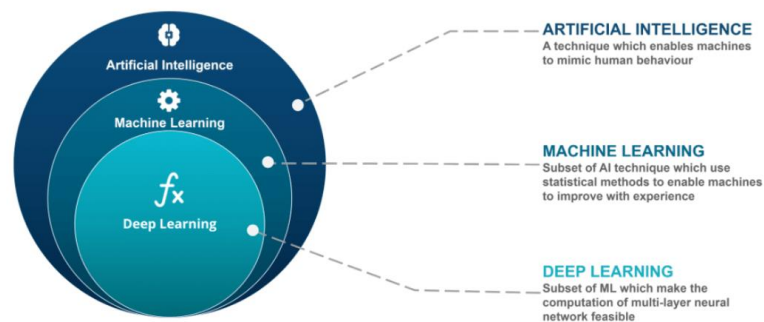
Per venire in contro alle problematiche esposte sopra, in questo lavoro sono stati proposti diversi approcci di segmentazione automatica basati sul Deep Learning, di cui parleremo nel prossimo capitolo. Verranno messi a confronto approcci più consolidati con modelli più recenti, che dimostrano di essere molto promettenti.



## Capitolo 3

### Deep Learning

Il Deep Learning (DL) fa parte del concetto più ampio di intelligenza artificiale (AI), insieme al Machine Learning (ML). Potremmo definirlo come un sottoinsieme del ML che è a sua volta un sottoinsieme dell'AI.

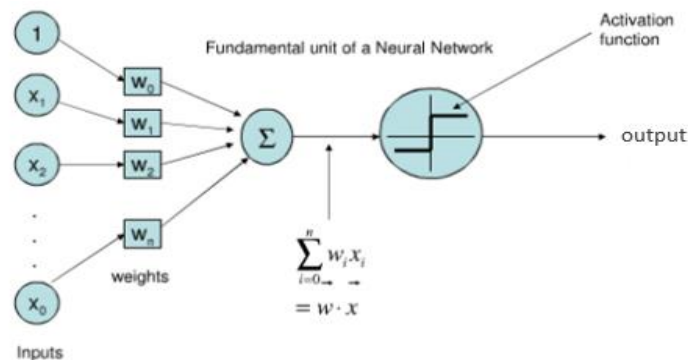


**Figura 3.1:** Rappresentazione schematica della differenza tra Artificial Intelligence; Machine Learning e Deep learning [21].

A differenza del machine learning, dove viene eseguita un'attenta operazione di feature extraction da parte dell'operatore, il deep learning si avvicina maggiormente al meccanismo di apprendimento del cervello umano, estraendo direttamente le feature dai dati.

### 3.1 Artificial Neural Network (ANN)

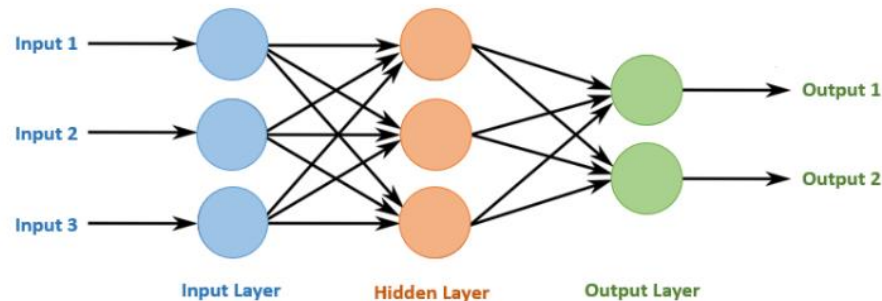
Nelle reti neurali artificiali, l'apprendimento è permesso da unità fondamentali chiamate neuroni (figura 3.2). Essi prendono degli input, attribuiscono loro dei pesi e li sommano, li elaborano attraverso una funzione di attivazione non necessariamente lineare, e restituiscono un risultato.



**Figura 3.2:** Rappresentazione dell'unità fondamentale delle Neural Network, chiamate neuroni artificiali o Perceptron [22].

I neuroni sono collegati tra loro e organizzati in layer (figura 3.3):

- Il primo layer è detto di input;
- Poi abbiamo (eventualmente) i layer nascosti (hidden layer). Il numero di layer nascosti determina la profondità della rete neurale;
- L'ultimo layer è detto layer di output, e restituisce il risultato del task.



**Figura 3.3:** Schema strutturale di una Artificial Neural Network [23].

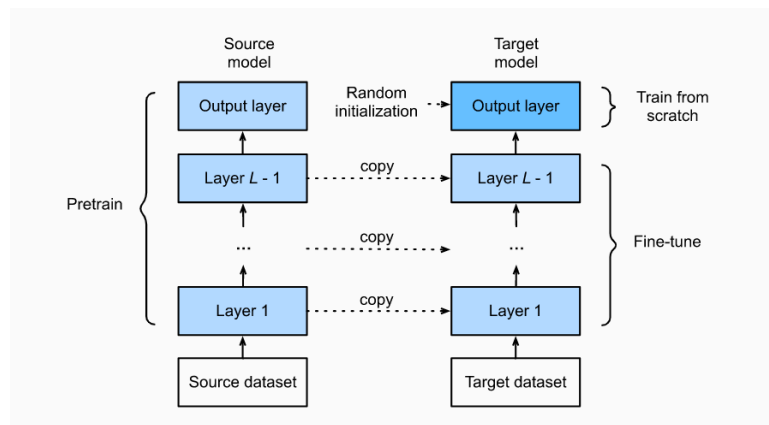
Una funzione di perdita (loss function) calcola l'errore tra l'output ottenuto e quello atteso e i pesi vengono modificati per minimizzarlo. Più la rete è profonda, maggiore è il numero dei parametri allenabili e delle connessioni; dunque aumenta anche la sua capacità di estrarre informazioni a diversi livelli di astrazione.

## 3.2 Deep Learning Neural Network

Il "Deep Learning", non è legato alla profondità della rete, piuttosto riguarda la sua capacità di estrarre direttamente le feature dai dati di input. Questo effetto "black box" [24], è un'arma a doppio taglio, poiché rende le reti performanti nell'individuare trend in dati molto complessi, ma i risultati sono difficilmente controllabili. Nonostante questo, il deep learning continua la sua ascesa perché ottiene buone performance anche dove i metodi classici falliscono [24]. Inoltre, la sua diffusione è favorita dall'avanzamento tecnologico con elementi che riducono i tempi di calcolo (Graphic Processor Unit - GPU) e Dataset più grandi. Una condizione necessaria per ottenere risultati robusti, infatti, è l'utilizzo di un set di dati abbastanza rappresentativo e numeroso, altrimenti si rischia di ottenere un modello incapace di generalizzare; in poche parole, si dice che va in overfitting. Per aggirare questo problema, è possibile ricorrere a due strategie: Data Augmentation e Transfer Learning.

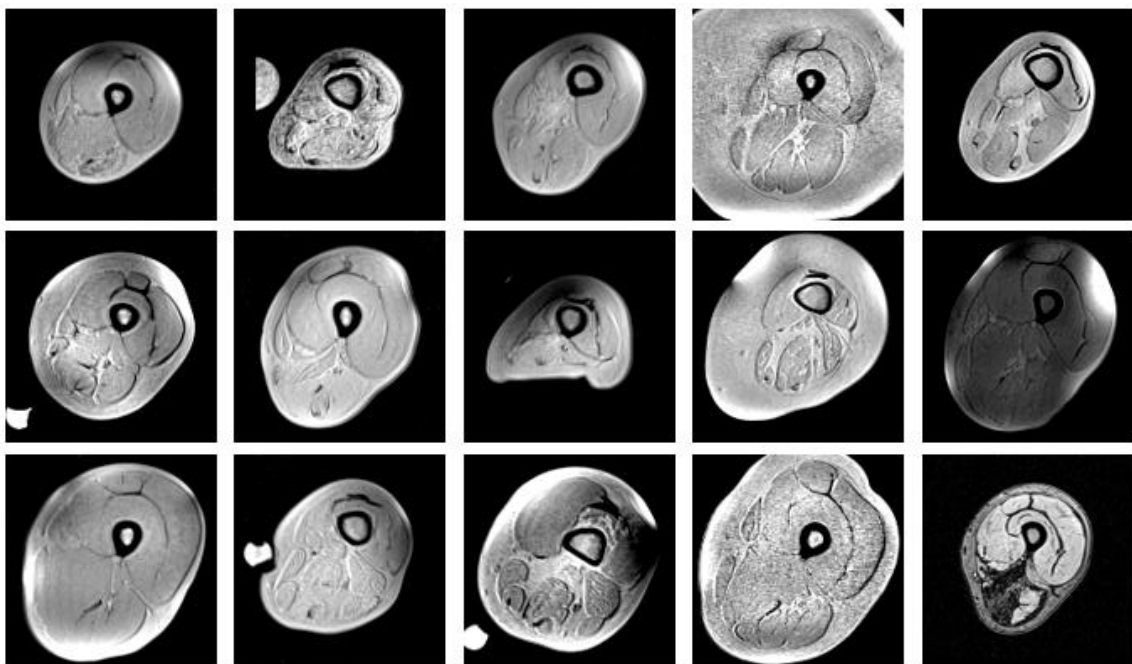
- Il Data Augmentation consiste nell'apportare piccole modifiche, come rotazioni, flip, intensity shift, crop, per ottenere più immagini partendo da una sola.
- Il Transfer Learning, invece, consiste nell'utilizzare reti pre-allenate.

Una tipologia di Transfer Learning è chiamata fine tuning (figura 3.4). In questo caso, si utilizzano reti allenate su grandi dataset (il più famoso è ImageNet), si copiano i pesi di tutti i layer tranne dell'ultimo in un'altra backbone, e poi si allena solo il layer di output sul proprio dataset. In questo modo, i parametri da allenare crollano drasticamente e anche un dataset meno numeroso può essere adeguato.



**Figura 3.4:** rappresentazione del concetto di fine-tuning [25].

Nel campo dell'healthcare e dell'imaging medico in particolare, dove la variabilità tra i pazienti è molto elevata (figura 3.5), le reti neurali basate su deep learning si rivelano un metodo sempre più promettente.



**Figura 3.5:** Esempio di variabilità tra diversi soggetti. Immagini estratte dal nostro dataset, raffiguranti la coscia destra. Ogni immagine rappresenta la dodicesima slice dei volumi considerati, appartenenti a soggetti malati (distrofia facio-scapolo-omerale, distrofia miotonica di tipo 1) e a soggetti sani.



## Capitolo 4

### Stato dell'Arte

Nel presente capitolo, verranno descritte le Deep Learning Neural Network confrontate in questo lavoro, fornendo una panoramica che comprende approcci consolidati e ampiamente utilizzati, basati sulle convoluzioni, e approcci emergenti, basati sui Transformer.

#### 4.1 Convolutional Neural Network (CNN)

Nel campo dell'immagine processing, le reti più conosciute e utilizzate sono le Convolutional Neural Network (CNN) chiamate così perché utilizzano le convoluzioni per estrarre le feature. Negli anni passati, le CNN hanno mostrato ottimi risultati nei task principali di Computer Vision [26], ovvero:

- Classificazione;
- Detection;
- Segmentazione.

Queste reti sono contraddistinte da:

- Convoluzioni;
- Funzioni di attivazione;
- Pooling;
- Fully-connected.

Ogni operazione è svolta dai rispettivi layer che, combinati in maniera opportuna, costituiscono l'architettura della CNN.

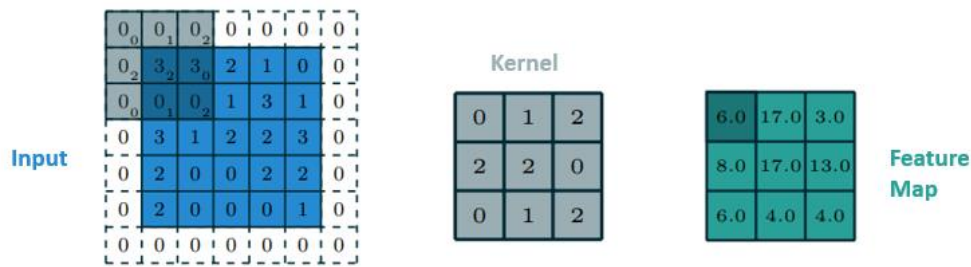
Si possono individuare dei così detti inductive bias, ovvero, delle caratteristiche che dipendono dal tipo di rete, come la spatial locality. Essa implica che pixel (o voxel) vicini nell'immagine siano generalmente simili tra loro e quindi correlati [27], [28].

##### 4.1.1 Convolutional Layer

La convoluzione è un'operazione di filtraggio tra un kernel, e l'immagine di input, lungo la quale viene fatto scorrere (figura 4.1). Esso è anche chiamato feature detector, dato che le convoluzioni permettono alla rete di estrarre le feature dall'immagine.

Bisogna definire tre parametri nella convoluzione:

- Dimensione del kernel;
- Stride (passo): di quanto scorre il kernel lungo l'immagine;
- Zero-padding: si sceglie se aggiungere o meno una cornice di zeri lungo le dimensioni dell'immagine.

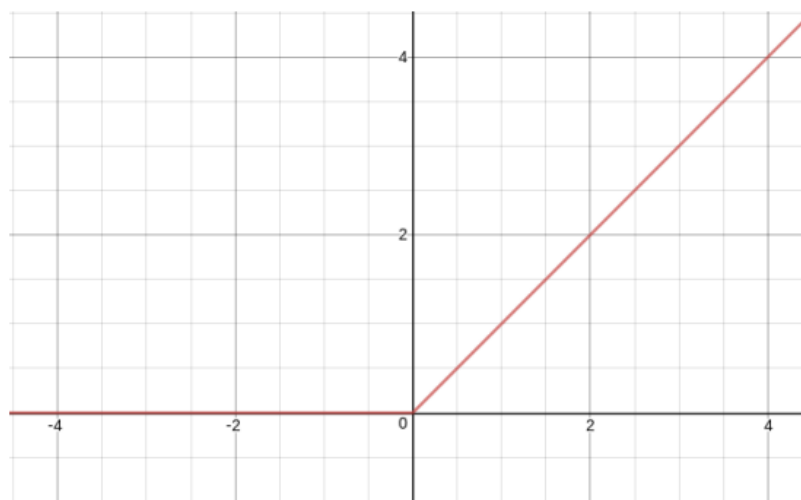


**Figura 4.1:** Esempio dell'operazione di Convoluzione. Consideriamo un'immagine di input  $5 \times 5$ ; un kernel di dimensione  $3 \times 3$ ; Zeropadding pari a 1 e stride pari a 2 [29].

All'immagine di input possono essere applicati più filtri: il numero dei kernel determina la profondità della nuova matrice, anche detta feature map.

#### 4.1.2 Funzione di attivazione

Le funzioni di attivazione introducono non linearità, ovvero modificano l'output della convoluzione. Questo è eseguito in modi diversi, a seconda della funzione di attivazione scelta. Una delle più utilizzate negli hidden layer delle CNN è la funzione ReLU, che lascia invariati i valori positivi della matrice, e porta a zero i valori negativi (figura 4.2).



**Figura 4.2:** Rappresentazione della funzione di attivazione ReLU [30].

Tra le altre funzioni di attivazione più comuni abbiamo Sigmoid e Softmax. La Sigmoid è utilizzata nei task di segmentazione binaria ed è definita dall'equazione (4.1).

$$\sigma(z_i) = \frac{e^{z_i}}{1 + e^{z_i}} \quad (4.1)$$

La Softmax, invece, rappresenta un'estensione della Sigmoid, applicabile in task di segmentazione multi-classe. È definita dall'equazione (4.2):



$$Softmax(z_i) = \frac{e^{z_i}}{\sum_{j=0}^{k-1} e^{z_j}} \quad (4.2)$$

Con  $k$  = numero di elementi che costituiscono  $z$ .

Entrambe le funzioni di attivazione, associano ad ogni elemento dell'output del precedente layer ( $z_i$ ) la probabilità di appartenere ad una classe, ottenendo valori compresi tra 0 e 1. Nel caso della Softmax, poiché al denominatore tiene conto di tutti gli elementi, la somma delle probabilità è sempre pari a 1. Questo non è necessariamente vero per la Sigmoidale.

#### 4.1.3 Pooling Layer

In fine, il layer di pooling esegue la feature selection, quindi riduce le dimensioni della feature map. Esistono diverse strategie di pooling, come mostrato in figura 4.3.

					Max Pooling		Average Pooling		Sum Pooling	
	6	6	6	6	6	6	5.25	5.25	21	21
	4	5	5	4	4	4	3	3	12	12
	2	4	4	2						
	2	4	4	2						

**Figura 4.3:** Esempi di diverse strategie di Pooling. Il Pooling è una strategia di feature selection, infatti passiamo da 16 a 4 feature. Max Pooling seleziona il massimo del subset di valori considerati; Average Pooling restituisce la loro media; Sum Pooling restituisce la loro somma [31].

#### 4.1.4 Fully Connected Layer

L'output dell'ultimo pooling layer viene, in fine, vettorizzato, e rappresenta l'input di una classica Artificial Neural Network. Il termine "Fully Connected" indica che ogni neurone del layer precedente è connesso a quelli del layer successivo. L'output del Fully-Connected layer (FC) viene poi sottoposto ad una funzione di attivazione, di solito una Softmax, che restituisce come output una classe.

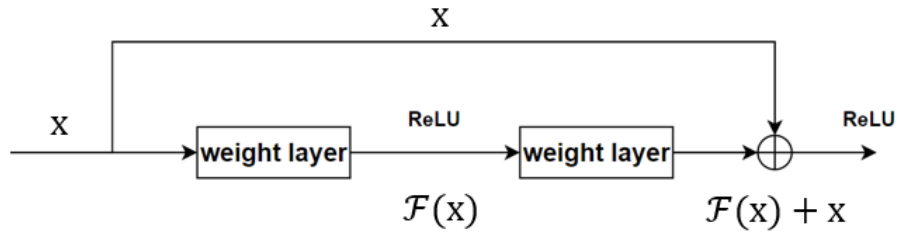
Nel task di segmentazione, il Fully-Connected layer, viene sostituito da un convolutional layer, poiché la presenza del FC causa la perdita delle relazioni spaziali tra gli elementi dell'immagine.

### 4.2 ResNet

L'accuratezza della rete migliora all'aumentare della sua profondità [32]. Tuttavia, con alcuni esempi di CNN, come AlexNet<sup>7</sup> o VGGNet<sup>8</sup>, si è visto che, oltre un certo numero di layer, le performance non miglioravano più a causa del vanishing gradient [33]. Per questo motivo si è passati a nuovi modelli in cui, anche le feature a basso/medio livello, venivano riportate nei layer più profondi. Questa differente strategia di apprendimento, si basa sulle skip connection (figura 4.4), in cui l'output del layer precedente avanza insieme a quello del layer corrente.

<sup>7</sup> Alex Krizhevsky et al., (2012).

<sup>8</sup> Simonyan and Zisserman, (2014).



**Figura 4.4:** Rappresentazione di una skip-connection [33].

Le skip connections sono anche dette connessioni o blocchi Residui. Da qui, il nome delle architetture che le implementano: Residual Net, note anche come ResNet (figura 4.5). A seconda del numero di layer convoluzionali, distinguiamo diversi tipi di ResNet (ad esempio, ResNet34, ResNet50).



**Figura 4.5:** Rappresentazione schematica della ResNet50 [34].

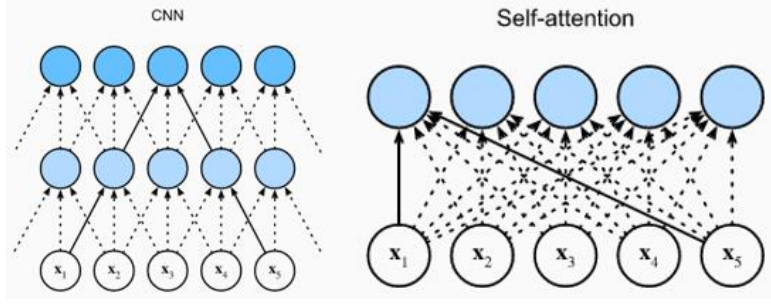
### 4.3 Vision Transformer

Le CNN estraggono molto bene le informazioni locali ma peccano nel riconoscere dipendenze “globali” (long-range dependency), a causa delle dimensioni limitate dei kernel. In questo contesto, si inserisce un nuovo modello, chiamato Transformer, che inizialmente, ha ottenuto eccellenti risultati nel Natural Language Processing (NLP).

I primi “Vision Transformer”, apparsi nel 2021 [35], hanno mostrato performance competitive sui benchmark più popolari, e sono stati poi declinati in altre varianti.

Le caratteristiche più importanti sono:

- Gli elementi di input vengono analizzati in parallelo. Nel caso delle immagini, esse vengono divise in patch di dimensioni fisse e dotate di una codifica di posizione (Position embedding). Le embedded patch, quindi, vanno in ingresso al blocco di encoder del Transformer [32]. Gli elementi della sequenza di input vengono anche chiamati token;
- Il meccanismo di Multihead Self-Attention [36], permette di riconoscere dipendenze ad ampio raggio, e informazioni relative al “contesto” in cui l’elemento della sequenza è inserito.



**Figura 4.6:** CNN coglie meglio le relazioni locali, transformers basati su Self attention colgono meglio le long range dependency. In questo caso,gni token viene messo in relazione con tutti gli altri [37].

#### 4.3.1 Multihead Self-Attention

Ogni elemento della sequenza di input viene convertito in una rappresentazione embedded comprensibile alla rete.

Ogni embedded input ( $I$ ), viene moltiplicato per dei pesi ( $W$ ):

- $W^Q$  matrice dei pesi delle Query;
- $W^K$  matrice dei pesi delle Key;
- $W^V$  matrice dei pesi dei Value.

Come descritto dall'equazione (4.3), si calcolano Query ( $Q$ ), Key ( $K$ ) e Value ( $V$ ) [38]:

$$Q = IW^Q, \quad K = IW^K, \quad V = IW^V \quad (4.3)$$

Con:

- $I \in \mathbb{R}^{n_i \times d_i}$  dove  $n$  e  $d$  sono la lunghezza e la dimensione dell'input, rispettivamente [39];
- $W^Q \in \mathbb{R}^{d_i \times d_k}$ ,  $W^K \in \mathbb{R}^{d_i \times d_k}$  dove  $d_k$  è la dimensione delle Key scelta uguale alla dimensione delle Query;
- $W^V \in \mathbb{R}^{d_i \times d_v}$  dove  $d_v$  è la dimensione dei Value

Ottenuti  $Q, K$  e  $V$ , il meccanismo di Self-Attention è descritto dall'equazione(4.4) [36] :

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (4.4)$$

La Multi-Head Self-Attention, è un'estensione della Self-Attention: anziché avere un solo layer che esegue la Self-Attention, se ne hanno  $h$  (da "head"), indipendenti tra loro, che operano in parallelo. Successivamente, gli output di ciascuna "testa" vengono concatenati (equazione (4.5)).

$$MultiHead(Q, K, V) = Concat(Z_1, Z_2, \dots, Z_h)W^O \quad (4.5)$$

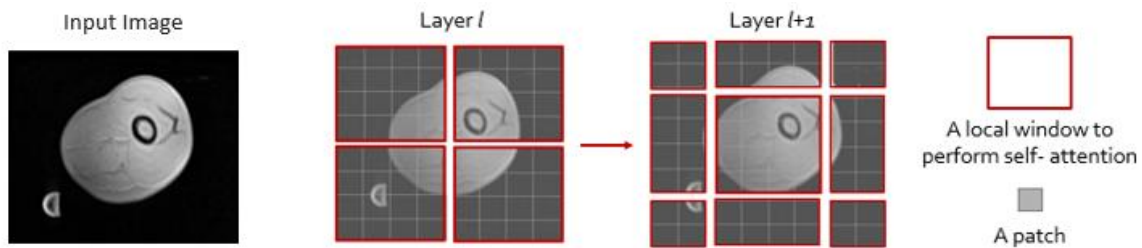
Con:

- $Z_i = Attention(Q_i K_i, V_i)$ ,  $i = 1, \dots, h$
- $W^O \in \mathbb{R}^{hd_v \times d_{model}}$ , matrice dei pesi degli output.

È stato proposto anche un approccio di Local Attention, ovvero il meccanismo di Self-Attention viene applicato localmente, all'interno di una finestra, riducendo la complessità computazionale [40]. Un modello che utilizza la Local Attention è lo Swin Transformer.

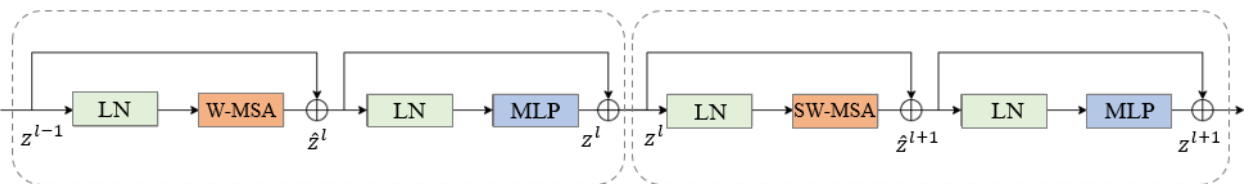
### 4.3.2 Swin Transformer

Gli Shifted Window (Swin) Transformers utilizzano un approccio gerarchico per estrarre le feature dall'immagine. Come mostrato in figura 4.7, prima dividono l'immagine in piccole patch e poi le raggruppano all'interno di finestre più grandi, non sovrapposte (tracciate in rosso). È all'interno di queste finestre che viene eseguita la Multi-Head Self Attention. L'elemento fondamentale degli Swin Transformers sono le Shifted Window: tra un layer e il successivo, la finestra viene shiftata, incorporando pixel diversi rispetto a prima, calcolando così relazioni diverse tra i token.



**Figura 4.7:** Rappresentazione delle Shifted Window, ispirata a [40].

In figura 4.8 vengono mostrati due SwinTransformer Block consecutivi. Vediamo che, nel layer  $l$ , il meccanismo di Local Attention è applicato alla finestra  $W$ , mentre nel layer successivo ( $l + 1$ ), è applicato alla Shifted Window (SW).



**Figura 4.8:** Rappresentazione di due layer consecutivi nell'architettura SwinTransformer[40]. MLP: Multilayer Perceptron; LN: Layer di Normalizzazione; W-SMA e SW-SMA: finestra (window) e la shifted window, rispettivamente, all'interno della quale viene calcolata la Multihead Self-Attention.

## 4.4 Tipologie di Architetture

### 4.4.1 U-Net

Nella segmentazione di immagini mediche, sia 2D che 3D, la U-Net (figura 4.9) è la più diffusa. È costituita da un encoder, che riduce le dimensioni dell'immagine e ne estrae le feature, e un decoder che combina: layer convoluzionali, di attivazione e upsampling layer, che sostituiscono i layer di

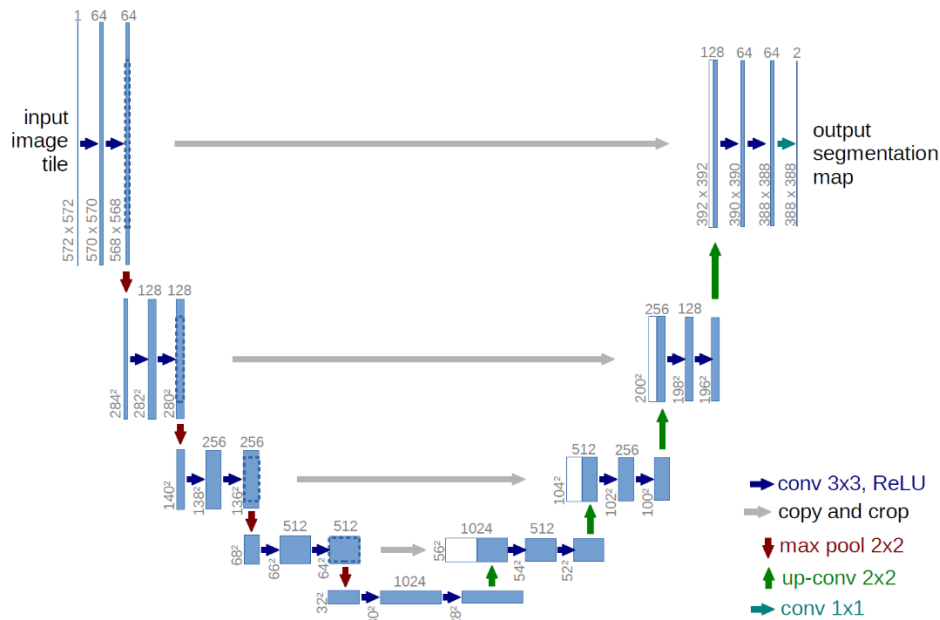
pooling. In questo modo, il decoder, sovracampiona la feature map fino ad ottenere una matrice delle stesse dimensioni dell'immagine di input. L'ultimo layer è costituito da una funzione di attivazione che permette di ottenere tante mappe quanti sono i canali di output, in cui ogni valore rappresenta la probabilità del pixel (o voxel) di appartenere ad una classe. Nei task di segmentazione multi-classe, viene spesso utilizzata la funzione Softmax; in questo caso, l'ultimo layer prende il nome di Softmax layer.

Per ottenere l'output definitivo della rete, ogni elemento deve essere univocamente associato ad una classe; per farlo si può applicare la funzione Argmax , che semplicemente assegna la classe con la probabilità massima. La classe predetta sarà  $\hat{y}$  (equazione (4.6)) [30]:

$$\hat{y} = \arg \max p_n \quad n \in 0, \dots, N - 1 \quad (4.6)$$

Dove :

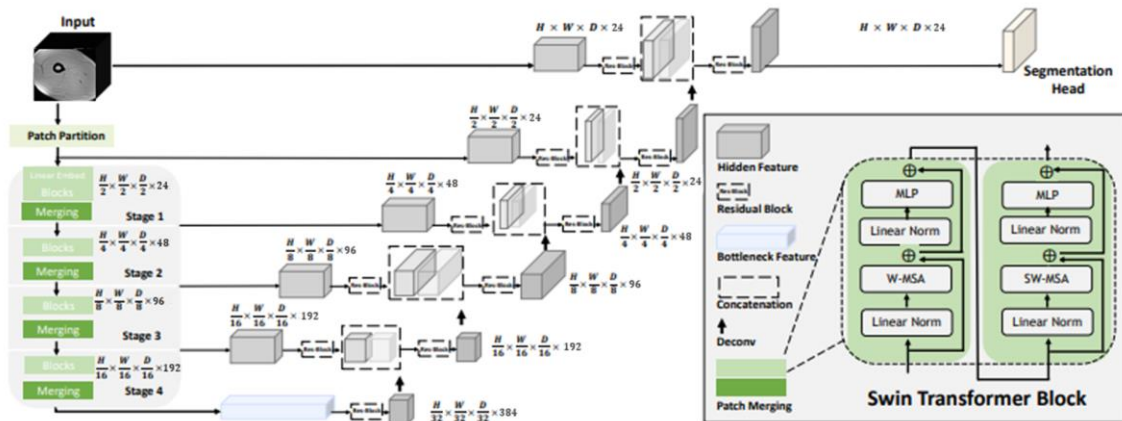
- $p_n$  è la probabilità dell'elemento di appartenere a una certa classe;
- $N$  è il numero di classi.



**Figura 4.9:** Rappresentazione generica della la caratteristica forma a U che la contraddistingue, da cui il nome U-Net. Encoder e decoder sono collegati dalle skip connections, a diverse risoluzioni spaziali[41].

La fully convolutional neural network con architettura U-Net rappresenta il modello più affermato, tuttavia, è possibile utilizzare anche altri tipi di encoder. Ad esempio, Lo Swin **UNETR** **Transformer** [42], (abbreviato SwinUNETR, mostrato in figura 4.10) , è un modello con architettura U-Net implementato per la segmentazione 3D e 2D [43], che presenta:

- per encoder uno Swin Transformer;
- per decoder una CNN.



**Figura 4.10:** Overview del modello SwinUNETR, usato in questo lavoro, per la segmentazione 3D. Immagine modificata da: [44].

#### 4.4.2 K-Net

Le reti K-Net, hanno l'obiettivo di utilizzare un unico framework per i tre task di segmentazione:

- *Segmentazione semantica.* Raggruppa gli elementi dell'immagine in base alle caratteristiche delle categorie. Di una categoria semantica, possono far parte diverse istanze. Se, ad esempio, la categoria è "animali" essa conterrà al suo interno altri tipi di oggetti, come: "gatto", "cane", "coniglio", che però, non verranno distinte;
- *Segmentazione di istanza.* Individua e discrimina le istanze come oggetti separati. Ritornando all'esempio precedente, saranno segmentati tutti i diversi tipi di animali ("cane", "gatto", "coniglio") ;
- *Segmentazione panottica.* Combina le informazioni della segmentazione semantica e di istanza per fare in modo che ogni oggetto appartenga ad una classe e tutte le istanze vengano identificate. Dunque riconosce che un cane e un gatto sono due istanze distinte, ma appartengono entrambi alla classe "animali".

Il nome K-Net deriva da Kernel, infatti, l'elemento chiave sono proprio le caratteristiche dei kernel e il loro meccanismo d'azione.

Nelle K-Net, i kernel sono dinamici, nel senso che si adattano alle caratteristiche dell'immagine di input, e la loro tipologia (semantic kernel, instance kernel o panoptic kernel), viene definita durante l'allenamento. Inoltre, c'è un rapporto uno-a-uno tra kernel e classe. Pertanto, ogni kernel è specifico solo per una classe semantica o un'istanza, e ricerca nell'immagine i pixel che ne potrebbero far parte, con una certa probabilità.

Utilizzare un framework unico per i tre task di segmentazione, migliora le capacità della rete di comprendere il contenuto della scena e di discriminare i diversi oggetti. Anche nel caso delle K-Net, è possibile utilizzare diverse backbone come, ad esempio, ResNet o SwinTransformer. Comparata con le altre architetture ha rivelato ottime performance nei tre task di segmentazione, superando in alcuni casi gli altri state-of-the-art model [45].



# Capitolo 5

## Dataset

Il Dataset è composto da immagini MRI acquisite con la tecnica Dixon, dal Radboud University Medical Center.

Il Dataset iniziale comprendeva 216 volumi MRI raffiguranti gli arti inferiori di 59 soggetti, caratterizzati da tre condizioni muscolari diverse (25 soggetti FSHD, 24 soggetti MD1, 10 Healthy Volunteer). Per i pazienti FSHD, sono presenti entrambi gli arti inferiori (destro e sinistro), data la progressione marcatamente asimmetrica della patologia, mentre, per i pazienti MD1, è stato fornito più di un volume a testa (da uno a tre), che può rappresentare l'arto destro o sinistro. Nel caso degli Healthy Volunteer, disponiamo di un solo volume per soggetto.

Sono stati trattati separatamente i volumi di coscia e gamba, quindi sono stati ottenuti due dataset differenti. I file sono distribuiti come segue:

- Coscia:
  - 50 volumi relativi a pazienti con distrofia facio-scapolo-omeroale (FSHD);
  - 47 volumi relativi a pazienti con distrofia miotonica di tipo 1 (MD1);
  - 10 volumi relativi a soggetti di controllo o Healthy Volunteer (HV).
- Gamba:
  - 50 volumi FSHD;
  - 49 volumi MD1;
  - 10 volumi HV.

A parte alcune eccezioni per la coscia (tabella 5.1), le dimensioni dei volumi sono:

- $160 \times 200 \times 72$  ;
- $256 \times 192 \times 32$ .

Lo spacing è compreso tra  $1.0 \times 1.0 \times 5.0$  mm e  $1.3594 \times 1.3594 \times 5.0000$  mm.

La suddivisione in Training, Validation e Test set , è riportata in tabella 5.1 per la coscia e in tabella 5.2 per la gamba. Il Dataset coscia comprende:

- Training Set: 86 volumi;
  - 48.84% FSHD; 45.35% MD1; 5.81% HV.
- Validation Set: 16 volumi;
  - 37.5% FSHD; 37.5% MD1; 25% HV.
- Test Set: 5 volumi.
  - 40% FSHD; 40% MD1; 20% HV.

Il Dataset gamba comprende:

- Training Set: 88 volumi;
  - 47.73% FSHD; 46.59% MD1; 5.68% HV;



- Validation Set: 16 volumi;
  - 37.5% FSHD; 37.5% MD1; 25% HV.
- Test Set: 5 volumi.
  - 40% FSHD; 40% MD1; 20% HV.

Tabella 5.1: Organizzazione del Dataset coscia.

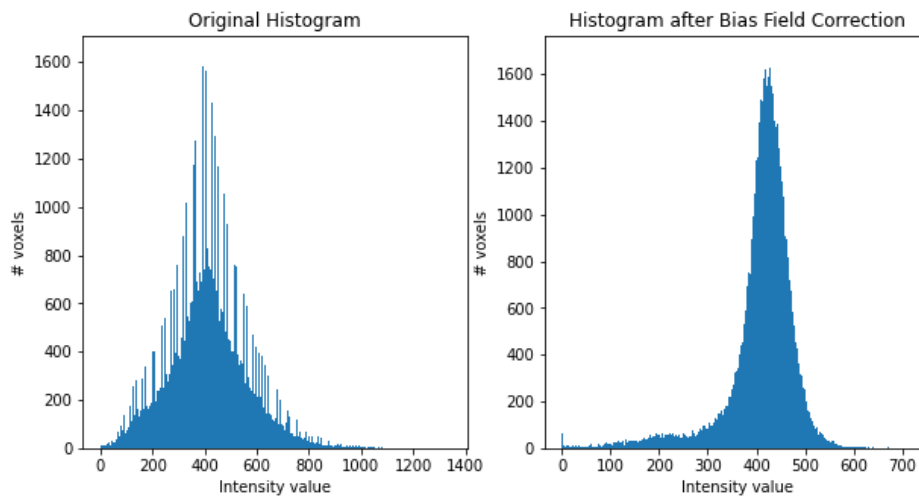
COSCIA		
TRAINING SET		
Number	Image Dimension	Voxel Spacing (mm)
42	160 × 200 × 72	1.3594 × 1.3594 × 5.0
29	256 × 192 × 32	1.0 × 1.0 × 5.0
6	256 × 192 × 32	1.0938 × 1.0938 × 5.0
3	256 × 192 × 32	1.0430 × 1.0430 × 5.0
2	256 × 224 × 32	1.0 × 1.0 × 5.0
1	256 × 192 × 32	1.1992 × 1.1992 × 5.0
1	256 × 192 × 32	1.1445 × 1.1445 × 5.0
1	256 × 232 × 32	1.0 × 1.0 × 5.0
1	256 × 256 × 32	1.0 × 1.0 × 5.0
VALIDATION SET		
Number	Image Dimension	Voxel Spacing (mm)
6	160 × 200 × 72	1.3594 × 1.3594 × 5.0
6	256 × 192 × 32	1.0 × 1.0 × 5.0
3	256 × 224 × 32	1.0 × 1.0 × 5.0
1	256 × 192 × 32	1.0938 × 1.0938 × 5.0
TEST SET		
Number	Image Dimension	Voxel Spacing (mm)
2	160 × 200 × 72	1.3594 × 1.3594 × 5.0
3	256 × 192 × 32	1.0 × 1.0 × 5.0

Tabella 5.2: Organizzazione del Dataset gamba.

GAMBA		
TRAINING SET		
Number	Image Dimension	Voxel Spacing (mm)
42	160 × 200 × 72	1.3594 × 1.3594 × 5.0
46	256 × 192 × 32	1.0 × 1.0 × 5.0
VALIDATION SET		
Number	Image Dimension	Voxel Spacing (mm)
6	160 × 200 × 72	1.3594 × 1.3594 × 5.0
10	256 × 192 × 32	1.0 × 1.0 × 5.0
TEST SET		
Number	Image Dimension	Voxel Spacing (mm)
2	160 × 200 × 72	1.3594 × 1.3594 × 5.0
3	256 × 192 × 32	1.0 × 1.0 × 5.0

## 5.1 Stato iniziale del Dataset

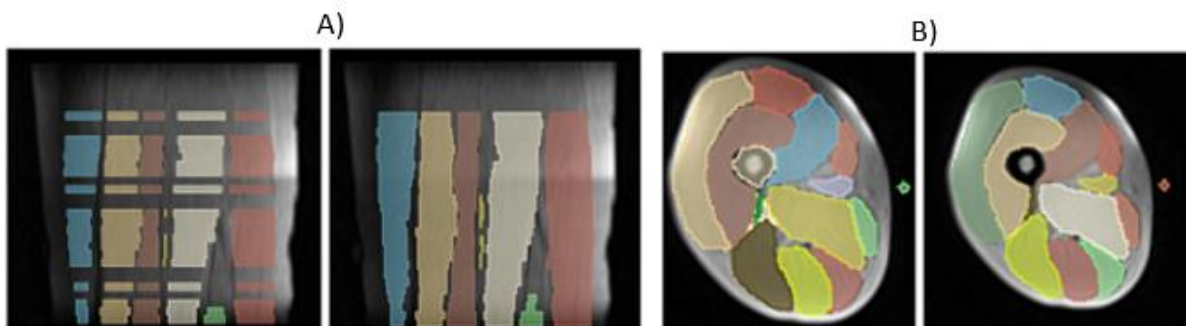
Tutti i volumi sono stati filtrati per correggere l'effetto della polarizzazione del campo (Bias Field Effect). Si tratta di un rumore a bassa frequenza che corrompe le immagini MRI, dovuto alla disomogeneità del campo magnetico degli strumenti di acquisizione [46]. Questo segnale deve essere rimosso in quanto potrebbe penalizzare in maniera significativa successivi step di elaborazione, come segmentazione o Texture Analysis [46]; in questo caso è stato utilizzato il filtro N4ITK [47]. In figura 5.1, è mostrato come cambia l'istogramma delle luminosità dopo la correzione della disomogeneità del campo magnetico.



**Figura 5.1:** Rappresentazione di come varia l'istogramma delle luminosità prima e dopo il filtraggio con N4ITK. In questo caso, i voxel dello sfondo sono stati esclusi dalla raffigurazione.

Dopo il filtraggio, l'intensità delle immagini è stata normalizzata.

Solo alcune slice disponevano della segmentazione manuale, quindi è stata eseguita un'interpolazione per ripristinare le label su tutto il volume. Le segmentazioni manuali utilizzate per il training, dunque, sono in parte state tracciate manualmente dai clinici, in parte interpolate (figura 5.2.A). In fine, alcuni volumi presentavano anche la label relativa all'osso, che è stata rimossa (figura 5.2.B).



**Figura 5.2:** Esempi di operazioni preliminari sulle segmentazioni. A) Interpolazione delle label lungo tutto il volume; B) Rimozione della label dell'osso.

## 5.2 Riassegnazione manuale delle label

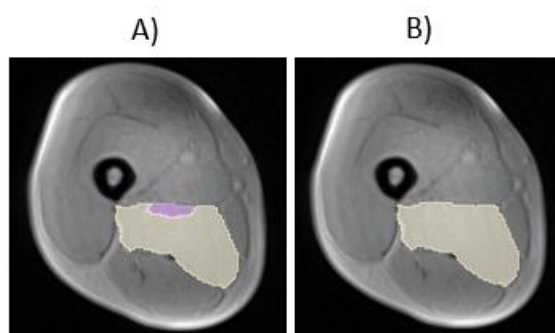
Le segmentazioni manuali hanno subito ulteriori modifiche, ovvero, è stata ripristinata la corrispondenza biunivoca tra label e muscolo, e le label mancanti sono state interamente aggiunte.

Le operazioni descritte in questo paragrafo sono state eseguite manualmente attraverso il software 3D Slicer<sup>9</sup>.

Si è cercato di uniformare il più possibile il dataset anche da questo punto di vista, in modo tale che i muscoli fossero rappresentati allo stesso modo in tutti i volumi.

### 5.2.1 Label della Coscia

I volumi relativi ai soggetti MD1 e HV, non presentavano la label del muscolo adduttore breve. Comparando diverse segmentazioni tra loro, sembrava che, laddove mancasse, fosse stata accorpata con quella del grande adduttore. I due muscoli, sono generalmente colpiti allo stesso modo dalle due distrofie [9] e fanno parte dello stesso complesso funzionale. Per queste ragioni, è stato deciso di unire le due label nei volumi in cui erano presenti entrambe (figura 5.3.A), ottenendo così la label 11: “Grande Adduttore + Adduttore breve” (figura 5.3.B).



**Figura 5.3:** Esempio della modifica effettuata per ottenere la label 11 sul volume 1\_001\_1\_1\_R.nii (Training Set). A) Prima della modifica; in bianco la label grande adduttore, in viola la label dell’adduttore breve. B) Dopo la modifica; label 11: “Grande Adduttore + Adduttore Breve”.

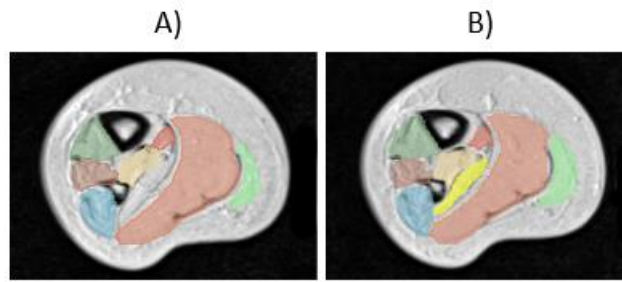
Nei volumi degli Healthy Volunteers mancavano molte label, che sono state aggiunte manualmente, ripristinando la corretta corrispondenza tra label e muscolo.

### 5.2.2 Label della Gamba

In tutti i volumi dei soggetti FSHD, erano presenti le label dei muscoli flessore lungo dell’alluce e popliteo, che invece mancavano negli altri. Il flessore lungo dell’alluce è stato contrassegnato manualmente, cercando di non modificare eccessivamente la label adiacente che individua il Soleo (figura 5.4).

---

<sup>9</sup> <https://slicer.org/>



**Figura 5.4:** File 2\_004\_02.nii (Training Set). A) Segmentazione manuale originale. B) Segmentazione dopo l'aggiunta della label 9: "Flessore lungo dell'Alluce" (in giallo).

Anche il Popliteo è stato aggiunto manualmente nei restanti volumi. Tuttavia, essendo una label molto piccola e presente solo in poche slice (poiché si trova nella porzione più prossimale della gamba), si è deciso di eliminarla vedendo che le reti non riuscivano a segmentarla. È risultato un buon compromesso per aumentare le performance medie delle reti, senza perdere molta informazione.

Anche per la gamba, è stato necessario riassegnare correttamente la corrispondenza label-muscolo, in modo tale che risultasse biunivoca.

### 5.3 Pre-processing Dataset 3D

Avere un dataset eterogeneo può compromettere le prestazioni di successive operazioni [48]. Per uniformare i Dataset, sono stati eseguiti i seguenti tre step di pre-processing:

- Flip;
- Histogram matching;
- Ricampionamento.

Per il Flip e il Ricampionamento, sono state utilizzate le funzioni di MONAI<sup>10</sup> pensate per le immagini mediche, con le quali è possibile lavorare in 3D ma anche in 2D. Inoltre, fornisce metodi standardizzati per calcolare metriche per validare i modelli, che supportano sia la segmentazione binaria, ma anche quella multi-classe.

#### 5.3.1 Flip: da "L" a "R"

Per permettere alle reti di riconoscere meglio i diversi muscoli, tutti gli arti inferiori di sinistra sono stati flippati rispetto al loro asse verticale, in modo da ottenere la stessa conformazione dell'arto destro. Questa scelta è stata compiuta arbitrariamente.

Confrontando le performance dei primi allenamenti con quelle ottenute dopo il flip, i risultati miglioravano notevolmente.

<sup>10</sup> <https://docs.monai.io/en/stable/transforms.html>

### 5.3.2 Histogram Matching

L'histogram matching (HM) consiste nel modificare l'istogramma di un'immagine in base a quello di un'altra immagine di riferimento, quindi, è utile per rendere più simili i dati dal punto di vista del contrasto e della luminosità. Si tratta di un'operazione consigliata nel pre-processing di volumi MRI, che risentono particolarmente dei diversi parametri di acquisizione [49].

Per eseguire l'histogram matching è necessario che, sia l'immagine da modificare, sia l'immagine di riferimento, abbiano le stesse dimensioni. Le dimensioni dei dataset di coscia e gamba si dividono in:

- $160 \times 200 \times 72$  per i soggetti FSHD;
- $256 \times 192 \times 32$  per i soggetti MD1 e HV.

All'interno del dataset coscia, le dimensioni di 8 volumi, si discostavano da quelle citate, come mostrato nella tabella 5.3. Quindi, sono state ripristinate correttamente, prima di procedere.

Tabella 5.3: Volumi del Dataset Coscia, con dimensioni da adattare al resto del sottogruppo ( $256 \times 192 \times 32$ ).

NOME VOLUME	DIMENSIONI ORIGINALI
2_002_1_1.nii	$256 \times 232 \times 32$
2_004_1_1.nii	$256 \times 224 \times 32$
2_005_1_1.nii	$256 \times 256 \times 32$
2_007_1_1.nii	$256 \times 224 \times 32$
2_008_1_1.nii	$256 \times 224 \times 32$
2_008_1_2.nii	$256 \times 224 \times 32$
2_008_1_3.nii	$256 \times 224 \times 32$

Successivamente, per uniformare le dimensioni dei dati, è stato eseguito un resize per ottenere volumi  $256 \times 192 \times 32$  da quelli  $160 \times 200 \times 72$ .

A questo punto, non resta che scegliere un istogramma di riferimento a cui adattare tutti gli altri. In alcuni approcci è stata scelta arbitrariamente un'immagine dal dataset [50], [51]; nel nostro caso, invece, per ottenere dei riferimenti più rappresentativi, tutti i volumi sono stati mediati, ricavando un istogramma di riferimento medio per i volumi della coscia e uno per quelli della gamba.

Per l'histogram matching, è stata seguita l'implementazione di Nyul et al. [52]. Nel codice utilizzato<sup>11</sup> sono stati impostati i seguenti parametri:

- `histogram_levels = 256`;
- `match_points = 95`;
- `ThresholdAtMeanIntensity = "True"` per escludere i pixel del background e ottimizzare l'algoritmo [53].

Dopo aver eseguito l'HM, le dimensioni originali dei volumi sono state ripristinate. Successivamente, è stato calcolato l'istogramma medio di quest'ultimo sottogruppo ottenuto (ovvero i volumi

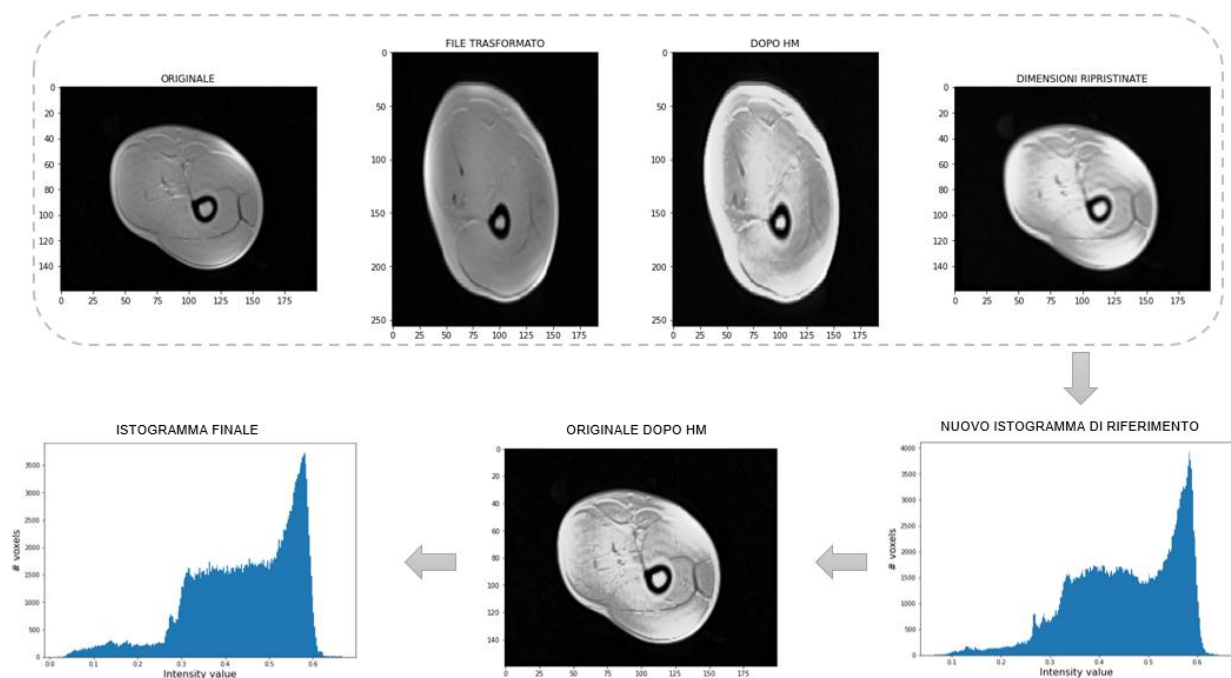
<sup>11</sup> [https://github.com/serqivalverde/mri\\_utils](https://github.com/serqivalverde/mri_utils)

ritornati alle dimensioni  $160 \times 200 \times 72$  dopo l'HM), per poter eseguire l'histogram matching con i volumi originali.

In questo modo, le variazioni introdotte dal resize non vengono riportate nelle successive elaborazioni, e tutti i volumi, indipendentemente dalle loro dimensioni, vengono uniformati rispetto lo stesso istogramma di riferimento.

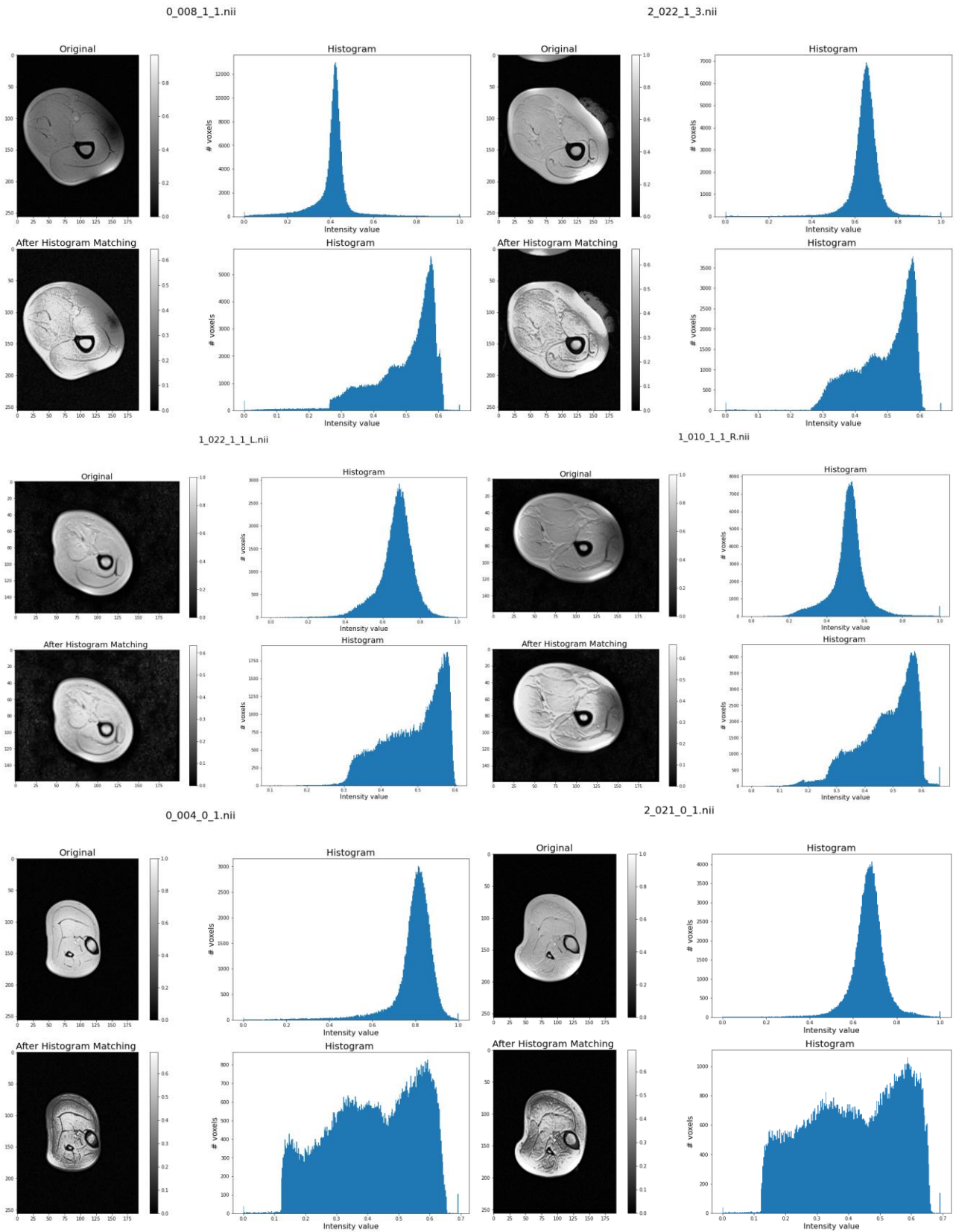
Riassumendo:

- Se le dimensioni del volume MRI si discostano da quelle di riferimento (figura 5.5):
  - Eseguire il resize rispetto le dimensioni del volume di riferimento;
  - Eseguire l'histogram matching;
  - Ripristinare le dimensioni originali;
  - Ottenere il nuovo istogramma di riferimento (HM\_new);
  - Eseguire l'histogram matching tra il volume originale e HM\_new.
- Se il volume MRI ha le stesse dimensioni del riferimento:
  - Eseguire direttamente l'histogram matching.



**Figura 5.5:** Illustrazione degli step per eseguire l'histogram matching su volumi che si discostano dalle dimensioni dall'istogramma di riferimento, facendo in modo che mantengano le proprie caratteristiche (ovvero che non vengano alterati dal resize) e, allo stesso tempo, siano omogenei rispetto agli altri volumi del dataset. Si nota che il volume originale dopo l'HM sia più nitido rispetto a quello ottenuto dopo il ripristino delle dimensioni originali.

In figura 5.6, vediamo come i volumi dei nostri dataset risultino più omogenei dal punto di vista dell'istogramma delle luminosità.



**Figura 5.6:** Panoramica dell'effetto dell'Histogram Matching su diversi volumi di Coscia e Gamba, con dimensioni differenti. Immagini che prima avevano istogrammi di luminosità molto differenti tra loro, sono stati uniformati.



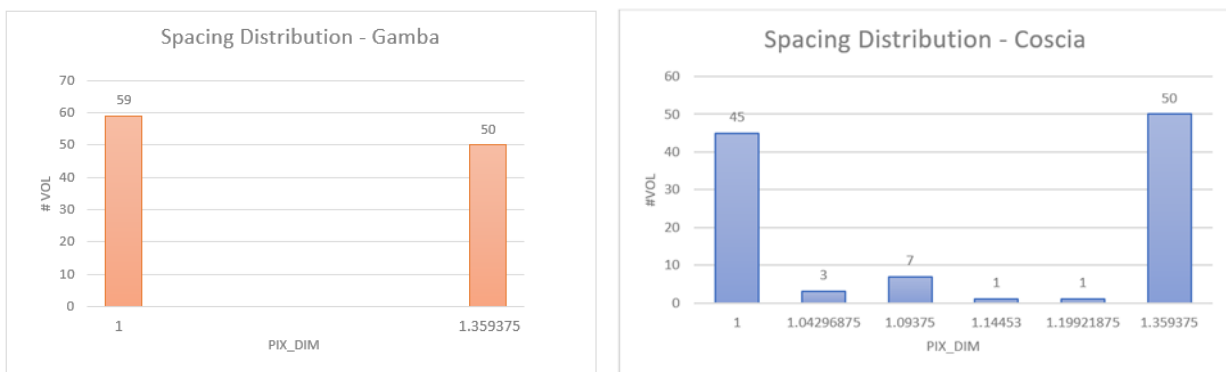
### 5.3.3 Ricampionamento

In questo paragrafo, vedremo come è stato scelto il Voxel Spacing utilizzato per il ricampionamento.

Come riportato nelle tabelle 5.1 e 5.2, il nostro dataset è abbastanza eterogeneo in dimensioni e spacing. Scegliere opportunamente la dimensione dei voxel in funzione di un'operazione di ricampionamento è cruciale, soprattutto nelle immagini mediche, in cui ogni elemento contiene informazioni importanti dal punto di vista clinico, che rischiano di essere mascherate o stravolte.

Considerando tutti gli elementi dei dataset, è stato scelto il valore mediano tra i Voxel Spacing, per uniformare i volumi:

- Voxel Spacing gamba:  $1.0 \times 1.0 \times 5.0$  mm;
- Voxel Spacing coscia:  $1.0938 \times 1.0938 \times 5.0000$  mm.



**Fig.5.7:** Distribuzione delle dimensioni dei voxel per i due dataset. Sull'asse delle ascisse, vediamo riportata solo la dimensione che varia tra i diversi spacing.

## 5.4 Creazione del Dataset 2D

Dai volumi pre-processati come descritto nei paragrafi precedenti, sono state estratte le patch usate per il training dei modelli 2D.

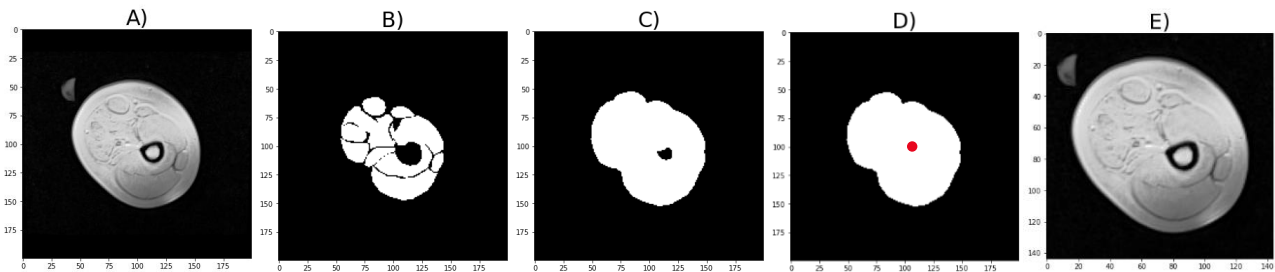
Per ottimizzare l'allenamento, è importante che le patch contengano meno background possibile, e racchiudano completamente la gamba o la coscia. Per questo motivo, è stato eseguito il crop rispetto al centroide della regione di interesse.

Di seguito vengono elencati gli step eseguiti per l'estrazione delle patch 2D :

1. Caricare i volumi e le rispettive maschere di segmentazione manuali;
2. Ricampionamento;
3. Zeropadding lungo altezza e larghezza del volume (la dimensione di profondità è stata lasciata intatta) (figura 5.8.A);
4. Considerare ciascuna slice alla volta;
5. Applicare un thresholding alla maschera di segmentazione manuale per renderla binaria (figura 5.8.B);
6. Eseguire un'operazione di dilatazione, per chiudere le fessure (figura 5.8.C);



7. Eseguire un'operazione di chiusura per riempire i buchi più grandi ancora rimasti e trovare il centroide (in rosso) di questa regione (figura 5.8.D) ;
8. Croppare le patch di immagine e segmentazione rispetto alle coordinate del centroide (figura 5.8.E).



**Figura 5.8:** Rappresentazione del framework per l'estrazione delle patch che compongono il dataset 2D.

Le prime e le ultime slice di ogni volume non hanno le segmentazioni manuali. In questo caso, per eseguire il crop delle slice iniziali, si è considerato semplicemente il centro dell'immagine; mentre per quelle finali, si sono mantenute le ultime coordinate dei centroidi calcolate.

Per sfruttare al meglio la dinamica delle immagini, per passare da 0 a 255 si è considerato il range compreso tra il 1° e al 99° percentile della regione contenente i muscoli.

In fine, è stata mantenuta una patch ogni 3 per mantenere il dataset rappresentativo evitando ridondanze e sono state scartate le patch che non disponevano di una segmentazione manuale pur raffigurando dei muscoli.

In questo modo otteniamo il dataset 2D, formato da:

- 1103 patch  $144 \times 144$  per le cosce così suddivise:
  - 897 nel Training Set;
  - 156 nel Validation Set;
  - 50 nel Test Set.
- 1086 patch  $112 \times 112$  Per le gambe così suddivise:
  - 875 nel Training Set;
  - 163 nel Validation Set;
  - 48 nel Test Set.



# Capitolo 6

## Segmentazione multi-classe

Per una panoramica completa sul task di segmentazione multi-classe della muscolatura degli arti inferiori, sono state allenate 5 diverse Deep Learning Neural Network:

- 2D:
  - U-Net con encoder ResNet50;
  - K-Net + UPerNet con encoder ResNet50;
  - K-Net + UPerNet con encoder Swin Transformer;
- 3D:
  - Residual U-Net;
  - SwinUNETR.

Di ciascuna rete sono state ottenute due versioni: una per la coscia e una per la gamba. La struttura delle reti è rimasta invariata in entrambi i casi, ad eccezione del numero di canali di output che rappresentano il numero di muscoli da segmentare.

Per la coscia sono presenti 13 classi:

- 0 : Background;
- 1 : Vasto Laterale;
- 2 : Vasto Intermedio;
- 3 : Vasto Mediale;
- 4 : Retto Femorale;
- 5 : Sartorio;
- 6 : Gracile;
- 7 : Semimembranoso;
- 8 : Semitendinoso;
- 9 : Bicipite Femorale – Capo Lungo;
- 10 : Bicipite Femorale – Capo Breve;
- 11 : Grande Adduttore + Adduttore Breve;
- 12 : Adduttore Lungo.

Per la gamba sono presenti 10 classi:

- 0 : Background;
- 1 : Tibiale Anteriore;
- 2 : Tibiale Posteriore;
- 3 : Estensore Lungo;
- 4 : Peroneo;

- 5 : Flessore Lungo delle Dita;
- 6 : Soleo;
- 7 : Gastrocnemio Mediale;
- 8 : Gastrocnemio Laterale;
- 9 : Flessore Lungo dell'Alluce.

## 6.1 Training 3D

La pipeline, dal pre-processing al training, fino al calcolo delle metriche, è stata implementata utilizzando MONAI che fornisce dei framework, basati su Pytorch, ottimizzati per le immagini mediche.

Nel confronto delle architetture 3D, gli iper-parametri e le trasformazioni di Training e Validation Set, sono rimaste uguali, in modo tale che le differenze nelle performance dipendessero principalmente dai modelli.

### 6.1.1 Training Set Transform

Dopo aver caricato i volumi di immagini e rispettive segmentazioni manuali, è stato aggiunto un canale lungo la prima dimensione per trasformare i volumi nella rappresentazione channel-first; è stata applicata *CropForegroundd* per eliminare, quando possibile, i voxel dello sfondo e mantenere l'oggetto di interesse (foreground = voxel > 0); si è normalizzato l'orientamento delle immagini e delle label secondo la convenzione "RAS" con *Orientationd* e si sono ricampionati tutti i volumi con la trasformazione *Spacingd*, impostando `pix_dim = 1.0938, 1.0938, 5.0` per la coscia e `pix_dim = 1.0, 1.0, 5.0` per la gamba.

Successivamente, è stato eseguito il Data Augmentation attraverso delle trasformazioni Random :

- *RandCropByPosNegLabeld*: Random crop di immagini e corrispettive segmentazioni manuali. In seguito a questa operazione, da ogni volume, sono stati ottenuti altri 4 volumetti di dimensione  $96 \times 96 \times 32$ , tagliati in maniera random lungo il volume originale, rispetto il rapporto pos/neg, che è stato impostato pari a 1. Poiché le dimensioni del nostro dataset sono eterogenee, utilizzare questo step ha permesso anche di uniformarle. Sia per la gamba che per la coscia sono state scelte le stesse dimensioni per eseguire il crop, sebbene la coscia occupi una regione maggiore nell'immagine. Infatti, si è visto che aumentare le dimensioni di altezza e larghezza per la coscia non ha migliorato le performance ma ha aumentato la complessità computazionale. Al contrario, utilizzare dimensioni inferiori per la gamba, non ha diminuito considerevolmente i tempi di allenamento, ma ha peggiorato le performance. Per quanto riguarda la dimensione di profondità, è stata scelta la minima tra quelle di tutti i volumi per evitare il Resize che avrebbe alterato le segmentazioni manuali. Si ritiene che questo non abbia portato ad una notevole perdita di informazioni per i file con 72 slice, grazie alla casualità del taglio lungo il volume.

- *RandFlipd*: Esegue il flip lungo tutti e 3 gli assi, con una probabilità di 0.1 (ovvero c'è la probabilità del 10% di ottenere l'immagine flippata lungo l'asse specificato). Questo introduce ancora più casualità all'interno delle trasformazioni.
- *RandRotate90d*: Rotazioni random di 90° con una probabilità del 10 % (valore di default);
- *RandShiftIntensityd* : shift dell'intensità dell'immagine (non applicata alla label) con un offset di 0.10 e una probabilità del 50 %.

### 6.1.2 Inference Transform

Per i volumi del Validation Set, il pre-processing è composto dalle stesse trasformazioni del Training Set, fino allo Spacing (quindi escludendo le trasformazioni random). Durante l'allenamento, il Validation Set è utilizzato per evitare che ci sia over-fitting nel modello. È stato scelto di valutarlo dopo ogni epoca per ottenere risultati più precisi. Per farlo, è stata eseguita la sliding window inference con batch pari a 2, che applica il modello, di volta in volta, ad una finestra dell'immagine di partenza, per poi scorrere su di essa e ripetere l'operazione. In questo caso, anche le dimensioni della sliding window sono state impostate pari a  $96 \times 96 \times 32$ , come i volumetti ricavati dal Training Set dopo il Random Crop. Una volta ottenuto l'output, vengono applicate le seguenti trasformazioni:

- Funzione di attivazione Softmax;
- Argmax per assegnare univocamente una sola classe ad ogni pixel;
- Conversione in formato one-hot.

Anche la segmentazione manuale viene convertita in formato one-hot (specificando il numero opportuno di classi) e viene calcolato il Mean Dice come metrica di validazione. In fine, se il valore risultante supera quello dell'epoca precedente, si salva il modello.

### 6.1.3 Dettagli di implementazione

È stato impostato un training ripetibile (`set_determinism()`) poiché sono presenti delle trasformazioni random.

Il training è stato eseguito su Kaggle utilizzando una GPU T4  $\times 2$ , che è appropriata per l'immagine processing e le reti neurali [54].

Il training è stato eseguito con batch size pari a 2 per:

- 350 epoche nel caso delle Residual U-Net di coscia e gamba;
- 300 epoche nel caso della SwinUNETR di coscia e gamba. Con 350 epoche, il tempo di allenamento superava le 12 ore, che è il limite massimo di attività continua supportato da Kaggle. Dopo il primo allenamento di 300 epoche, i modelli sono stati ricaricati e riallenati per un numero inferiore di epoche, ma non sono stati ottenuti risultati migliori.

La SwinUNETR ha impiegato 11 ore e 59 minuti per un allenamento di 300 epoche , mentre la Residual U-Net 6 ore e 37 minuti per un allenamento di 350 epoche.

Come loss function è stata utilizzata DiceCELoss [55], [56] ,che restituisce la somma pesata di dice loss e cross entropy loss.

Come ottimizzatore è stato scelto AdamW, con learning rate pari a  $1e-4$  e weight decay pari a  $1e-5$  [44].

#### 6.1.4 Descrizione dei modelli

Sono state allenate una Residual U-Net e una SwinUNETR , implementate con MONAI.

Per eseguire la segmentazione 3D con la Residual UNet basta specificare `spatial_dim = 3`. Nel nostro caso, avremo un solo input channel, mentre l'output varia in base al numero delle classi (10 per la gamba e 13 per la coscia). La rete presenta 5 layer con 16,32,64,128 e 256 canali rispettivamente. Ogni layer ha uno stride pari a 2 e sono presenti 2 convolution residual unit (`num_res_units = 2`). È stata impostata la batch normalization poiché limita l'over-fitting del modello [57].

Nel caso dell'architettura SwinUNETR, invece, sono state impostate la dimensione delle immagini di input pari a  $96 \times 96 \times 32$  , e feature size pari a 24. Per quanto riguarda i parametri relativi ai canali di input e output, valgono le stesse considerazioni della Residual U-Net. Per ottimizzare il training e renderlo più efficiente in termini di utilizzo della memoria è stato impostato `use_checkpoint = True` [44].

## 6.2 Training 2D

Oltre che lavorare con file NiftI, MONAI permette anche di eseguire operazioni su formati 2D come DICOM o PNG, come nel nostro caso. Quindi, anche per il training sulle patch, sono state utilizzate le funzioni di MONAI.

### 6.2.1 Training Transform

Poiché le patch avevano già subito negli step precedenti il crop e il ricampionamento, nella pipeline del pre-processing 2D sono state omesse queste operazioni. Dopo aver caricato immagine e rispettiva maschera di segmentazione manuale, è stato aggiunto un canale lungo la prima dimensione per passare alla rappresentazione channel-first. Anche in questo caso è stato fatto Data Augmentation con le seguenti funzioni:

- U-Net
  - *RandFlipd* per flippare le immagini lungo entrambi gli assi con una probabilità del 10%;

- *RandRotate90d* per ruotare l'immagine di 90° in maniera random rispetto entrambi gli assi, con una probabilità del 10%;
- *RandShiftIntensityd* applicata solo all'immagine e non alla segmentazione, con un offset di 0.10 e una probabilità del 50 %.
- K-Net:
  - *Random flip* lungo l'asse y;
  - *Photometric Distortion* con i seguenti parametri: `brightness_delta = 6`, `contrast_range = (0.85, 1.15)`, `saturation_range = (0.85, 1.15)`, `hue_delta = 4`.

### 6.2.2 Inference Transform

Le immagini del Validation Test e le rispettive segmentazioni manuali sono state caricate e trasformate nella rappresentazione channel-first. Anche in questo caso, il Validation Set è stato utilizzato per evitare l'over-fitting del modello. Per la valutazione epoca dopo epoca del modello si è optato per un Simple Inferer (anziché per uno Sliding Window Inferer, come nel caso 3D). Infatti, una volta che sono state utilizzate le patch intere (non sono stati praticati random crop), ha dato performance migliori questa tipologia di inferer che prende in input direttamente tutta l'immagine. Le metriche di validazione utilizzate sono: Mean Dice per la U-Net con encoder ResNet50 e Intersection over Union (IoU) per le reti K-Net.

Prima di calcolare il Mean Dice, l'output dell'inference viene elaborato con Softmax, Argmax e trasformato in formato one-hot, e anche la maschera di segmentazione manuale viene convertita in formato one-hot, specificando il numero corretto di classi. In fine, viene salvato il modello con la metrica migliore.

### 6.2.3 Dettagli di implementazione

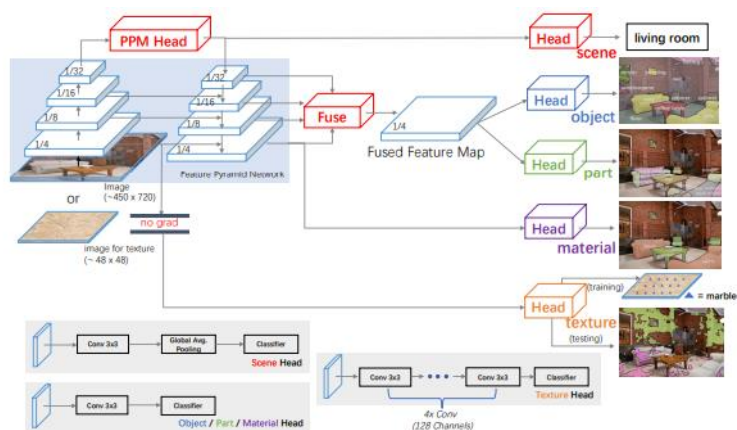
Il training della rete U-Net con encoder ResNet50, è stato eseguito su Google Colab con una GPU Tesla T4 ed è stato condotto per 50 epoche, con batch size pari a 2. Riallenare il modello per altre 50 epoche non ha portato a migliori risultati. Come funzione di perdita è stata utilizzata Dice Loss, mentre come ottimizzatore è stato scelto Adam con learning rate pari a  $1e-4$  [58].

Per le reti K-Net, il training è stato condotto per 150 epoche con batch size pari a 4. È stata utilizzata Cross Entropy Loss come funzione di perdita e AdamW come optimizer, con weight decay pari a  $5e-4$ . Il learning rate, è stato impostato pari a  $1e-04$  per la K-Net+UPerNet con encoder ResNet50 [59], mentre per la K-Net+UPerNet con encoder SwinTransformer è stato utilizzato learning rate pari a  $6e-05$  [60]. Per il training è stato abilitato Online Hard Example Mining (OHEM) come pixel sampler, sia sull'encoder che sul decoder, in questo modo solo i pixel di interesse vengono selezionati per il training, aumentando l'efficienza computazionale.

## 6.2.4 Descrizione dei modelli

La rete U-Net con encoder ResNet50, fa parte della libreria `segmentation_models_pytorch`<sup>12</sup>, che fornisce architetture preallenate sui più importanti benchmark. In questo caso, i pesi sono stati inizializzati su ImageNet. Poiché le patch avevano dimensione divisibile per 16, sono stati impostati 4 layers, con 16,32,64 e 128 canali, rispettivamente. È stata scelta `encoder_depth` pari a 4. Ciò significa che i tensori, generati ad ogni stadio dall'encoder, avranno risoluzione:  $[(H,W),(H/2,W/2),(H/4,W/4),(H/8,W/8),(H/16,W/16)]$ . Ciò che è stato modificato, tra la versione della gamba e quella della coscia è il numero di classi (*classes*, ce corrisponde a `out_channel`).

Per quanto riguarda le reti K-Net, l'encoder (o backbone) è costituito da una ResNet50 (`depth = 8`) preallenate su ImageNet [61] in un caso, e da uno SwinTransformer, nell'altro caso. Il numero di input e output channel dipende dalle dimensioni delle immagini, e non dalle classi che intendiamo individuare a priori. Questo aspetto riguarda la natura dinamica dei kernel nelle K-Net. In questo lavoro, è stata utilizzata una loro particolare configurazione: K-Net+UPerNet. La UPerNet (figura 6.1) si sposa bene con l'obiettivo della segmentazione unificata perseguito dalle K-Net, in quanto nasce per il task di *Unified Perceptual Parsing*, ovvero estrarre quante più informazioni possibili da un'immagine, a diversi livelli di dettaglio (la scena che viene raffigurata, gli oggetti presenti, i diversi materiali da cui sono composti o le loro texture) [62].



**Figura 6.1:** Rappresentazione della UPerNet [62].

## 6.3 Inference

Attraverso l'inference testiamo i modelli su dati completamente nuovi. In questa fase, entrano in gioco le immagini del Test Set, per valutare la capacità delle reti di generalizzare.

### 6.3.1 Inference 3D

Per testare il modello sui volumi, è stata implementata la Sliding Window Inference, che consiste nell'applicare il modello, di volta in volta, in una porzione dell'input, delineata da una "finestra" che scorre (sliding window) lungo di esso. Le dimensioni della sliding window sono scelte dall'utente; in

<sup>12</sup> <https://segmentation-models-pytorch.readthedocs.io/en/latest/>



questo caso si è optato per una finestra tridimensionale  $96 \times 96 \times 32$ , per riprodurre le condizioni in cui il modello è stato allenato<sup>13</sup>. Un altro parametro da impostare all'interno dell'inferer è il batch (non necessariamente uguale al batch utilizzato nell'allenamento), che è stato scelto pari a 2. Prima di eseguire l'inference, i volumi del Test Set (immagini e label) subiscono un pre-processing che consiste nell'aggiunta di un canale lungo la prima dimensione (rappresentazione channel-first), normalizzazione dell'orientamento secondo la convezione 'RAS' e ricampionamento, impostando l'opportuno Voxel Spacing.

Dopo l'inference, gli output subiscono un'ulteriore trasformazione, che è la composizione dei seguenti tre step:

- Ripristino dei i metadati originali del volume di input (original spacing, original affine);
- Funzione di attivazione (Softmax);
- Assegnazione di ogni pixel ad una classe , in maniera univoca (Argmax).

In fine, le segmentazioni automatiche così ricavate possono essere salvate e sono pronte per il post-processing.

### 6.3.2 Inference 2D

Per testare il modello sulle patch, è stato usato Simple Inferer , che considera interamente la patch in input. Come pre-processing, in questo caso , sono solamente state caricate le patch e trasformate nella rappresentazione channel-first. All'output vengono applicate le funzioni Softmax e Argmax e poi viene salvato.

## 6.4 Post-processing

Per rifinire le segmentazioni automatiche 2D e 3D ottenute dopo l'inference, è stato eseguito il post processing composto da due step: eliminazione di piccoli oggetti e riempimento dei buchi (chiusura).

Sono state provate diverse combinazioni, prima di scegliere il miglior compromesso:

- Nessun post-processing (p-p 0);
- Rimuovere oggetti con area inferiore a 30 pixel, chiusura buchi con area inferiore a 20 pixel (p-p 1);
- Rimuovere oggetti con area inferiore a 60 pixel, chiusura buchi con area inferiore a 40 pixel (p-p 2);
- Rimuovere oggetti con area inferiore a 80 pixel, chiusura buchi con area inferiore a 100 pixel (p-p 3).

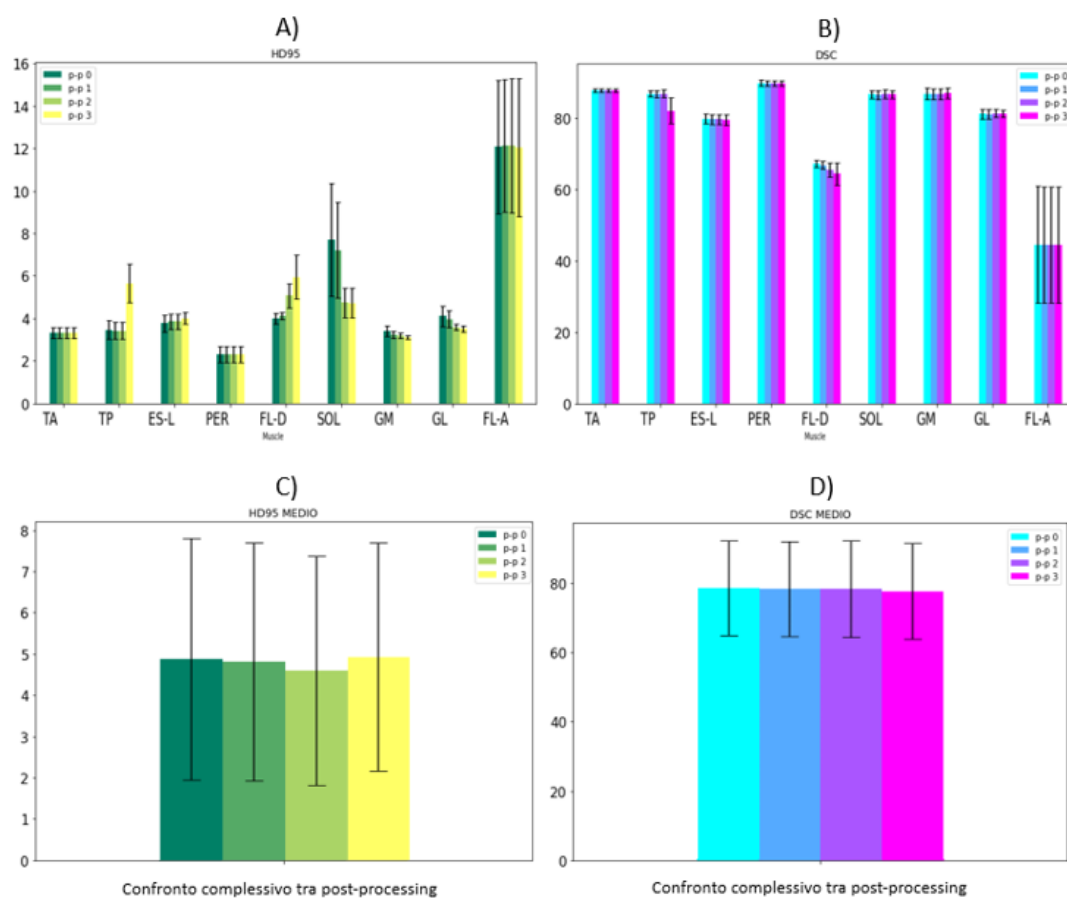
Vengono riportati gli effetti dei diversi post processing solo per la U-Net 3D della gamba, poiché

---

<sup>13</sup> Ricordiamo, infatti, che nel pre-processing del Training Set 3D, venivano ricavati dal volume originale 4 volumetti  $96 \times 96 \times 32$ .

mostra meglio come influisca sulle singole label e sulle metriche complessive. I risultati sono stati calcolati sui soggetti del Test Set. Considerazioni analoghe possono essere fatte per le altre reti. Per la coscia è stato utilizzato lo stesso post-processing della gamba.

Nella gamba, label più piccole come il flessore lungo delle dita, hanno risentito negativamente del post processing, come è stato sottolineato anche nel paragrafo 8.1.3 . In figura 6.1.A, vediamo che la distanza di Hausdorff al 95° percentile per il flessore lungo delle dita aumenta, mentre per il soleo diminuisce, infatti vengono eliminati i blob che ne compromettevano la precisione. Il post-processing influisce maggiormente sulla distanza di Hausdorff al 95° percentile, mentre il Dice Coefficient non viene modificato visibilmente (figura 6.1.B). Nonostante influisca negativamente sulle piccole label, il post-processing è stato mantenuto perché migliora complessivamente le performance (figura 6.1.C). In fine, il miglior compromesso si è rivelato p-p 2, ed è stato applicato a tutte le segmentazioni automatiche.



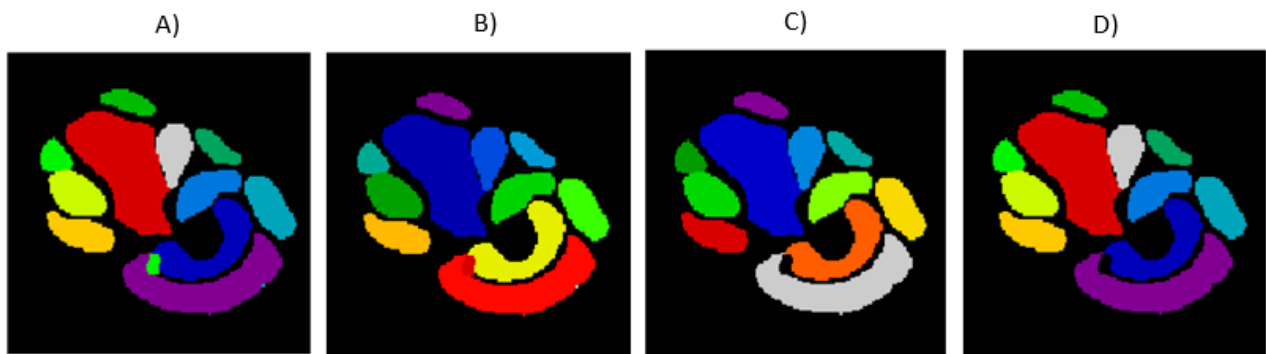
**Figura 6.2:** Effetto del post processing, mostrato sulle segmentazioni automatiche ottenute con la U-Net 3D della gamba. A) HD95 sui singoli muscoli; B) DSC sui singoli muscoli; C) HD95 mediato rispetto tutte le label; D) DSC mediato rispetto tutte le label.

Il post processing è stato eseguito slice per slice in tutti i casi, con le funzioni di `skimage.morphology`<sup>14</sup> ed è composto dai seguenti step (figura 6.3):

1. Separare i piccoli blob dalle label di interesse riassegnando un'etichetta a tutti i gruppi di pixel presenti nella maschera;

<sup>14</sup> <https://scikit-image.org/docs/stable/api/skimage.morphology.html>

2. Rimuovere i piccoli oggetti con area inferiore a 60 pixel;
3. Chiudere i buchi minori di 40 pixel;
4. Ripristinare il valore originale delle label.



**Figura 6.3:** Rappresentazione del framework del post-processing. A) Segmentazione automatica prima del post-processing. B) Riassegnazione delle label: ogni blob o gruppo di pixel ha una propria label. In questo modo possiamo rimuovere i piccoli oggetti che prima non venivano eliminati perché rientravano nell'area complessiva della label di interesse. C) Rimozione di blob inferiori a 60 pixel e riempimento dei buchi inferiori a 40 pixel. D) Ripristino della mappa con le label originali.

Dopo aver post-processato le segmentazioni automatiche 2D, sono state inizialmente aumentate le loro dimensioni attraverso lo zeropadding; successivamente, è stato eseguito un crop tenendo conto delle coordinate dello stesso centroide utilizzato per ottenere le patch. In questo modo, sono state ripristinate le dimensioni delle slice originali, per poi essere riassemblate in file NIfTI.

Poiché solo le slice più centrali erano dotate delle segmentazioni manuali, tutti i volumi (sia segmentazioni automatiche che manuali) sono stati tagliati lungo la dimensione di profondità, come segue:

- FSHD
  - GAMBA: dalla slice 23 alla slice 55, per un totale di 33 slice;
  - COSCIA: fino alla slice 38;
- MD1 e HV
  - COSCIA e GAMBA: dalla slice 1 alla slice 25.

sono stati considerati questi ultimi volumi per il calcolo delle metriche.

## 6.5 Metriche

Per calcolare le performance delle reti, sono state combinate due metriche:

- Dice Sørensen Coefficient (DSC);
- Hausdorff Distance al 95° percentile (HD95).

Il DSC è un indicatore della sovrapposizione tra la predizione del modello ( $P$ ) e il Ground Truth ( $G$ ). La definizione generale è descritta dall'equazione (6.1).

$$DSC = \frac{2|P \cap G|}{|P| + |G|} \quad (6.1)$$

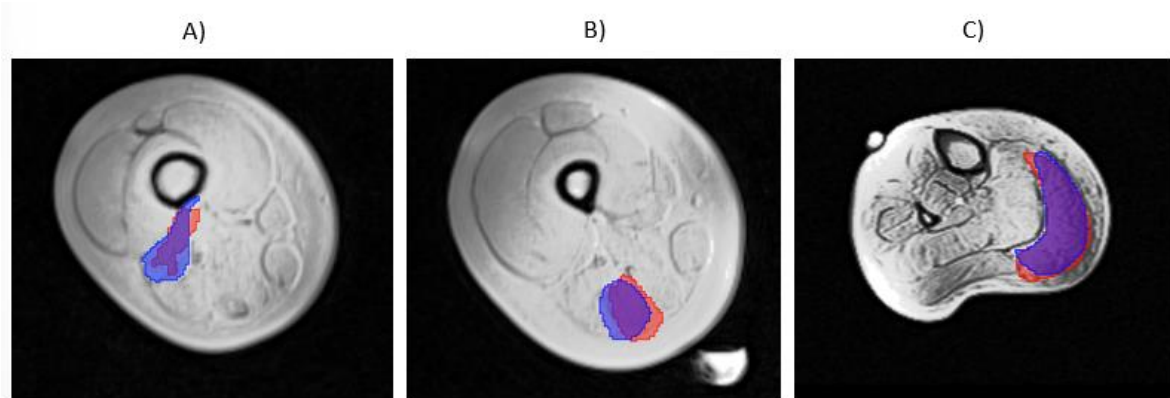
L'Hausdorff Distance (HD) è un indicatore di somiglianza tra la predizione del modello e il Ground Truth. Poiché calcola il massimo della distanza tra le superfici delle due segmentazioni, questa metrica potrebbe essere penalizzata da outlier (come ad esempio dei blob segmentati erroneamente e non eliminati del tutto dal post processing). Per limitarne l'effetto e rendere la metrica più robusta si è considerato il 95° percentile della distanza massima (equazione 6.2):

$$HD95(G, P) = \max \left\{ P_{95} \min_{p \in P} \|g - p\|, P_{95} \min_{g \in G} \|p - g\| \right\} \quad (6.2)$$

Dove:

- $g$  e  $p$  indicano dei punti appartenenti alla superficie di  $G$  e  $P$  [55];
- $\|g - p\|$  (e  $\|p - g\|$ ) è la distanza euclidea tra due punti, nel nostro caso.

Quindi, una HD alta, indica una importante differenza tra le due label (figura 6.3.A). Questa metrica quantifica la qualità della segmentazione automatica e quanto fedelmente si adatta al contorno di quella manuale. In questo modo si possono trarre delle conclusioni più dettagliate, infatti, non sempre DSC elevati corrispondono a segmentazioni precise (figura 6.3.B). Intuitivamente, un Dice elevato e una Hausdorff Distance bassa, descrivono una buona segmentazione automatica (figura 6.3.C).



**Figura 6.4:** Sovrapposizione della segmentazione automatica (rosso) al ground truth (blu). A) Esempio di una cattiva segmentazione del Bicipite Femorale – capo breve: DSC basso e HD95 elevata. B) Esempio che mostra l'utilità del confronto tra le due metriche, per la label del Semitendinoso: il DSC risulta elevato perché un'ampia porzione delle due segmentazioni si sovrappone, però anche HD95 risulta elevata poiché i contorni non aderiscono in maniera precisa. C) Esempio di una buona segmentazione del Gastrocnemio Mediale: DSC elevato e HD95 bassa.

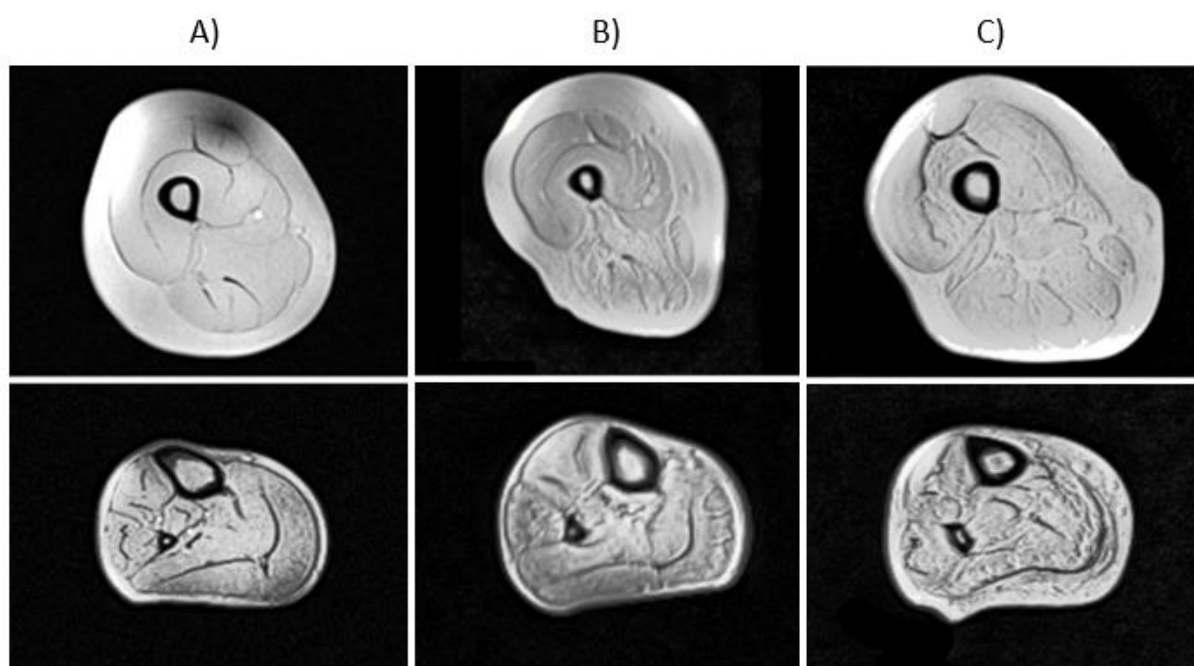
In questo lavoro sia DSC che HD95 sono state calcolate con MONAI Metrics API, che ha permesso di gestire comodamente l'approccio multi-classe grazie alla modalità di riduzione "*mean\_batch*".



## Capitolo 7

### Texture Analysis

La Texture Analysis nasce per quantificare certe caratteristiche dei tessuti, come omogeneità, anomalie o irregolarità dovute a condizioni patologiche. Nel campo dell'imaging medico si è rivelata uno strumento molto prezioso poiché i tessuti biologici sono quasi sempre caratterizzati da uno specifico pattern che può essere modificato da alcuni tipi di malattie. È il caso, ad esempio, delle distrofie muscolari (figura 7.1).



**Figura 7.1:** Alterazioni del tessuto muscolare dovute alle distrofie: i muscoli sani appaiono “levigati” e “omogeni”; quelli colpiti sembrano “rugosi” e “irregolari”. A) Volumi che raffigurano una sezione della coscia (0\_004\_1\_1.nii) e della gamba (0\_005\_0\_1.nii) di soggetti sani. B) Volumi che raffigurano una sezione della coscia (1\_018\_1\_1\_L.nii) e della gamba (1\_15\_0\_1\_L.nii) di pazienti FSHD. C) Volumi che raffigurano una sezione della coscia (2\_008\_1\_3.nii) e della gamba (2\_008\_0\_1.nii) di pazienti MD1.

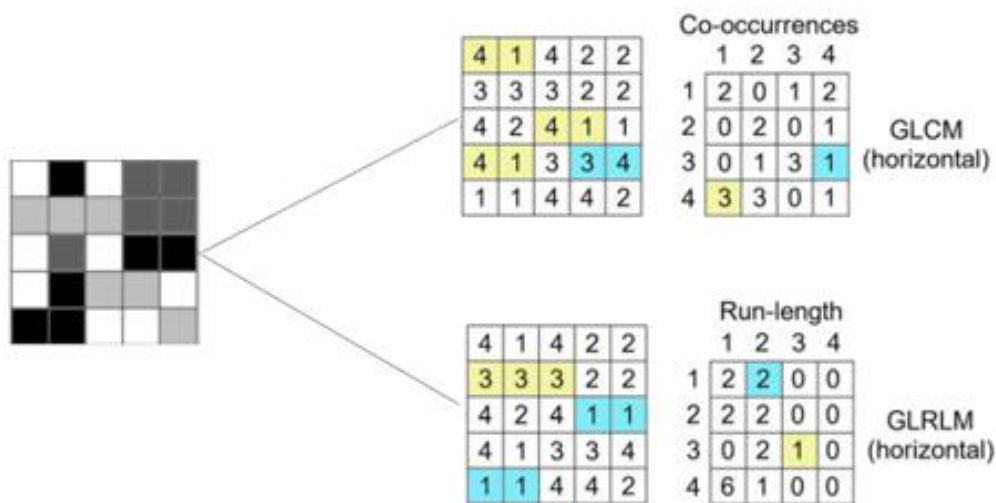
Per esprimere numericamente certe percezioni visive relative al tessuto, vengono estratte una serie di feature dalle regioni di interesse dell'immagine, dette, appunto, texture feature. Dividiamo le feature di tessitura in quelle del primo e quelle del secondo ordine. In entrambi i casi, per estrarre i parametri quantitativi non utilizzano direttamente l'immagine ma un suo surrogato.

Per le feature del primo ordine, si ricavano i valori dall'istogramma delle luminosità della regione di interesse, contrassegnata dalla label. In questo caso, però, si perdono le informazioni spaziali, e due regioni molto diverse tra loro con lo stesso istogramma delle luminosità, non verrebbero distinte.

Per le feature del secondo ordine, si utilizza una matrice ricavata dall'immagine con criteri diversi a seconda della tipologia. Ad esempio, i descrittori di Haralick, si basano sulla Gray Level Co-occurrence Matrix (GLCM) in cui si rappresenta la probabilità che elementi dell'immagine con un determinato livello di grigio possano trovarsi accanto, lungo una direzione.

Oppure, i descrittori di Galloway, si basano sulla Gray Level Run-Length Matrix (GLRLM) che tiene conto di informazioni spaziali meno circoscritte rispetto alla GLCM. In questo caso, la matrice si costruisce in base alla lunghezza dei percorsi (run) rappresentati da un certo numero di elementi adiacenti con la stessa intensità, lungo una direzione (figura 7.2).

Come si può notare, le feature danno informazioni complementari tra loro, dunque per ottenere una descrizione completa è importante combinarle.



**Figura 7.2:** Rappresentazione della GLCM e GLRLM, calcolate rispetto la direzione orizzontale (0°) [71].

Per l'analisi presentata in questo lavoro, è stata utilizzata la libreria Pyradiomics<sup>15</sup> e sono state estratte inizialmente 86 texture features del primo e del secondo ordine:

- First Order Statistics, 19 feature;
- Second Order Features:
  - Gray Level Co-occurrence Matrix (GLCM), 22<sup>16</sup> feature;
  - Gray Level Run Length Matrix (GLRLM), 16 feature;
  - Gray Level Dependence Matrix (GLDM), 14 feature;
  - Gray Level Size Zone Matrix (GLSZM), 16 feature.

<sup>15</sup> <https://pyradiomics.readthedocs.io/en/latest/>

<sup>16</sup> In totale, le feature della categoria GLCM sono 24. Ne sono state escluse 2: "Sum Average" perché dà le stesse informazioni di "Joint Average" a causa della simmetria della GLCM; e "Maximal Correlation Coefficient" poiché non veniva riportato tra i settings.

Per il setting dei parametri iniziali, si è fatto riferimento all'apposita repository su GitHub<sup>17</sup> [63], dedicata a immagini MRI con spessore pari a 5 mm (come quelle che compongono i dataset utilizzati).

## 7.1 Analisi Preliminare e Feature Extraction

Per comprendere meglio le potenzialità dei dataset, si è partiti da un'analisi preliminare. Come prima cosa, sono stati divisi tutti i file NIfTI in FSHD, MD1 e HV come segue:

- 50 volumi FSHD, per coscia e gamba;
- 47 volumi MD1 per coscia e 49 per gamba;
- 10 volumi HV, per coscia e gamba.

L'analisi è stata condotta muscolo per muscolo poiché:

- La degenerazione è tendenzialmente selettiva (alcuni muscoli vengono risparmiati a differenza di altri);
- Il pattern di coinvolgimento muscolare varia a seconda della patologia.

Lo scopo dell'analisi era vedere se le texture feature fossero in grado di differenziare lo stesso muscolo colpito da patologie diverse. La feature extraction è stata eseguita utilizzando le label manuali.

Per visualizzare i risultati, sono state valutate tre tecniche di dimensionality reduction ,tra cui:

- PCA;
- t-SNE;
- UMAP.

La Principal Component Analysis ha mostrato la migliore separazione tra le tre classi (figura 7.3.A), pertanto si è scelto di continuare solo con quest'ultima.

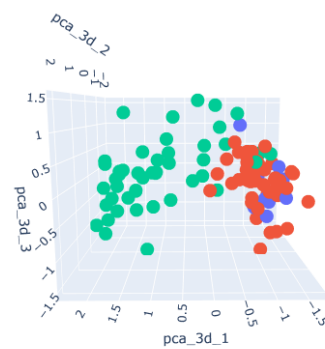
---

<sup>17</sup> [https://github.com/AIM-Harvard/pyradiomics/blob/master/examples/exampleSettings/exampleMR\\_5mm.yaml](https://github.com/AIM-Harvard/pyradiomics/blob/master/examples/exampleSettings/exampleMR_5mm.yaml)



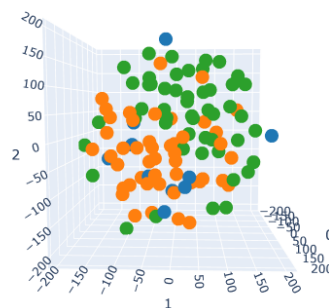
Bicipite femorale - capo breve

Class  
 ● HV  
 ● MD1  
 ● FSHD



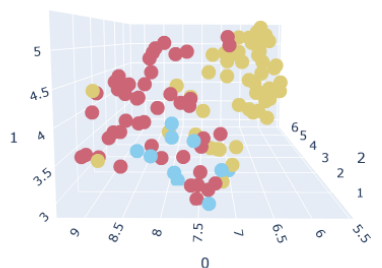
Bicipite femorale - capo breve

Class  
 ● HV  
 ● MD1  
 ● FSHD



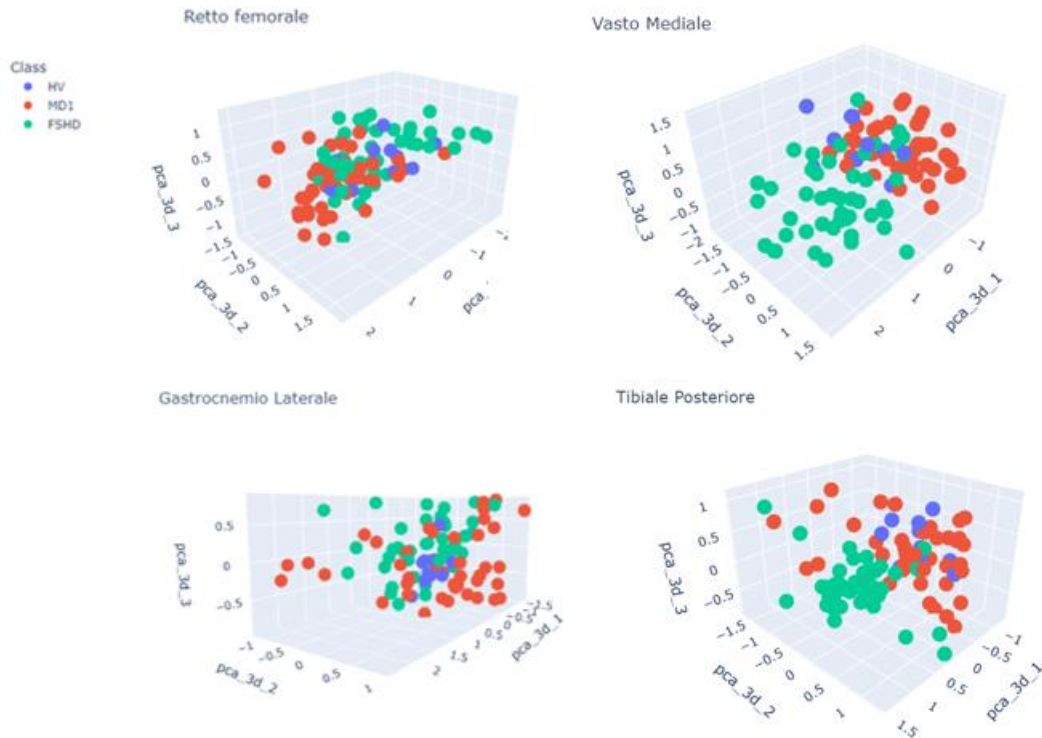
Bicipite femorale - capo breve

Class  
 ● HV  
 ● MD1  
 ● FSHD



**Figura 7.3:** Confronto metodi di dimensionality reduction, sulle feature relative alla label “Bicipite femorale – capo breve” a titolo di esempio . A) Principal Component Analysis (PCA); B) t-Distributed Stochastic Neighbor Embedding (t-SNE) ; C) Uniform Manifold Approximation and Projection (UMAP).

Confrontando questi primi risultati su tutti i muscoli, alcuni meglio di altri evidenziavano le differenze tra le classi FSHD e MD1. Tipicamente, si tratta di quelli che vengono coinvolti da una distrofia e risparmiati dall'altra. Invece, i soggetti sani (HV) non erano distinti, anche a causa dell'inferiorità numerica (figura 7.4).



**Figura 7.4:** Rappresentazione della separazione delle tre classi considerando tutte le feature, per alcuni muscoli più rappresentativi. Il Retto Femorale (coscia) e il Gastrocnemio Laterale (gamba) mostrano una separazione minima rispetto ai muscoli Vasto Mediale (coscia) e Tibiale Posteriore (gamba).

A causa della numerosità del dataset e delle feature, ci saranno sicuramente delle ridondanze che non permettono di raggruppare al meglio le classi.

Per selezionare le feature che evidenziassero meglio la differenza tra le due distrofie, sono stati utilizzati due test statistici:

- t-test;
- test di Wilcoxon-Mann-Whitney (test U di Mann-Whitney).

Nel prossimo paragrafo vengono approfonditi gli step di questa fase dell'analisi, prendendo come esempio il muscolo Vasto Laterale.

## 7.2 Analisi sui singoli muscoli: Vasto Laterale

La seconda parte dell'analisi consiste nella feature selection. Gli step dettagliati per il Vasto Laterale (VL) in questo paragrafo, sono stati ripetuti iterativamente per tutti gli altri muscoli.

Poiché lo scopo è aumentare la separazione tra le classi, si sono utilizzati dei test statistici per individuare quali feature mostrassero differenze statisticamente significative. I gruppi di pazienti sono tre, ma i soggetti HV sono molto meno rappresentati rispetto alle altre due tipologie, quindi i test saranno condotti considerando solo i pazienti FSHD e MD1.

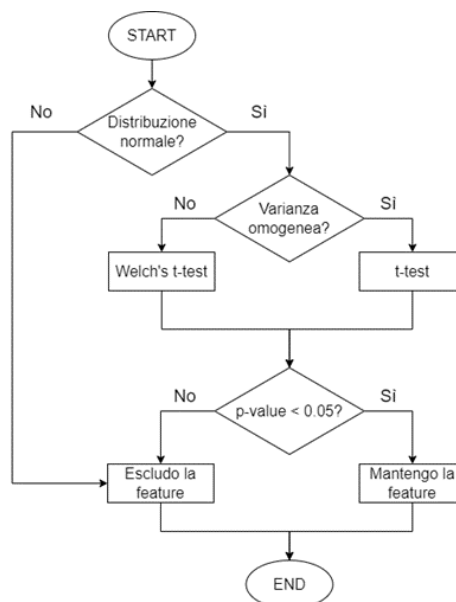
Si considera una feature alla volta, raggruppando i valori ottenuti per i soggetti FSHD e MD1 in due campioni separati. Tra questi ultimi, verrà calcolato il test statistico.

Sono stati utilizzati separatamente due test statistici, come accennato nel paragrafo precedente: il t-test e il test U di Mann-Whitney, che può essere applicato anche quando i campioni non hanno una distribuzione normale.

### 7.2.1 t-test

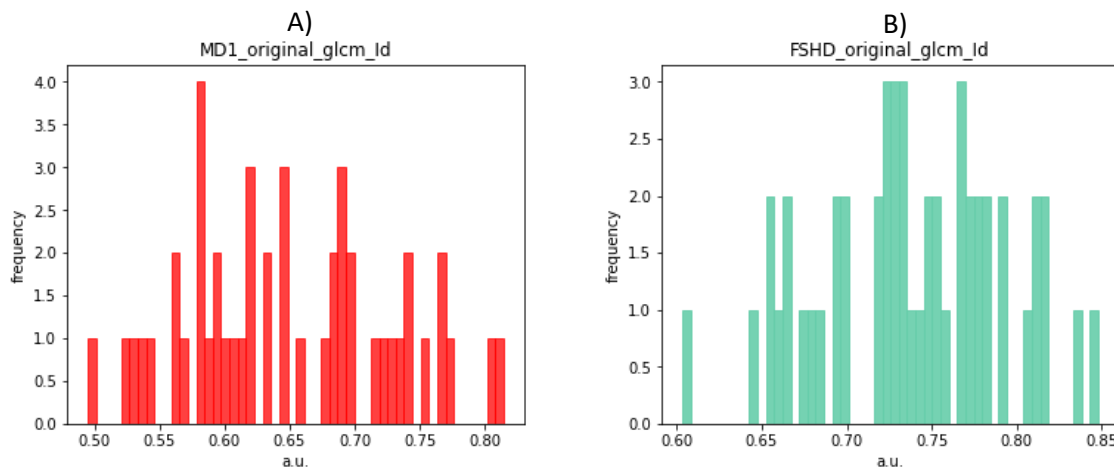
Prima di procedere, bisogna verificare che i dati soddisfino i criteri di applicabilità del test:

- La distribuzione deve essere normale;
- La varianza deve essere omogenea.



**Figura 7.5:** Flow-chart della feature selection attraverso il t-test.

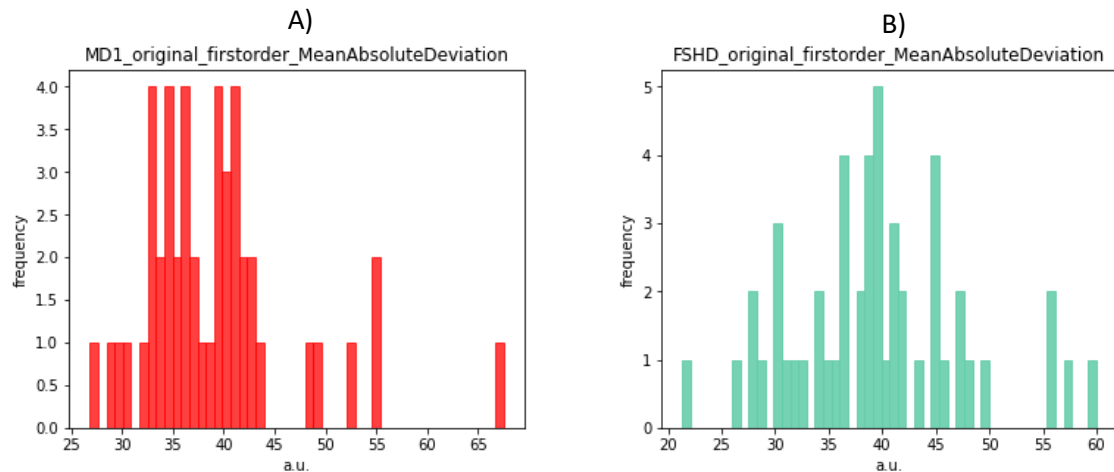
Per verificare che le distribuzioni delle feature fossero normali è stato condotto il test di Shapiro-Wilk (figura 7.6).



**Figura 7.6:** Esempio di una feature con distribuzione normale per entrambi i gruppi indipendenti. A) Feature “Inverse Difference (Id)” del secondo ordine (GLCM), soggetti MD1; distribuzione normale

(p-value = 0.4543). B) Feature “Inverse Difference (Id)” del secondo ordine (GLCM), soggetti FSHD; distribuzione normale (p-value = 0.8331).

Le feature che non avevano distribuzione normale per entrambe le classi, sono state scartate (figura 7.7).



**Figura 7.7:** Esempio di una feature con distribuzione normale per FSHD e non normale per MD1. A) Feature “Mean Absolute Deviation” del primo ordine, soggetti MD1; distribuzione non normale (p-value = 0.0001); B) Feature “Mean Absolute Deviation” del primo ordine, soggetti FSHD; distribuzione normale (pvalue = 0.2958).

Dopo questo step siamo passati da 86 feature a 26.

Dopo aver verificato l’ipotesi sulla varianza delle distribuzioni con il test di Levene, è stato eseguito il t-test per le feature con varianza omogenea e il Welch’s t-test nell’altro caso.

Alla fine, sono state mantenute solo quelle che mostravano una differenza statisticamente significativa (p-value < 0.05), cioè che rigettavano l’ipotesi nulla per cui le differenze tra le medie delle due distribuzioni fossero solo dovute al caso.



**Figura 7.8:** Esempio di una texture feature che evidenzia una differenza significativa tra le due classi (p-value = 4.99e-12), e verrà, dunque, mantenuta per caratterizzare il muscolo.

Dopo questa fase, sono rimaste le 16 texture features elencate in tabella 7.1.

Tabella 7.1: Subset di feature selezionate attraverso il t-test per il muscolo Vasto Laterale.

Categoria	Feature name
First Order	Energy Entropy TotalEnergy
GLCM	DifferenceAverage DifferenceEntropy Id Idm Idn JointEntropy SumEntropy
GLRLM	RunLengthNonUniformityNormalized RunPercentage ShortRunEmphasis ShortRunHighGrayLevelEmphasis
GLSZM	SizeZoneNonUniformityNormalized SmallAreaEmphasis

### 7.2.2 Test U di Mann-Whitney

In questo caso, si procede direttamente con il test statistico, senza confermare alcuna ipotesi.

Non dover escludere a priori feature per via dei criteri di applicabilità, permette di selezionarne di più per ogni muscolo. Per il muscolo Vasto Laterale, delle 86 features iniziali, 55 hanno mostrato una differenza significativa con questa procedura.

### 7.2.3 Dimensionality reduction: PCA

Dopo aver selezionato il subset di feature, il prossimo passo è visualizzare i risultati. Il numero delle dimensioni corrisponde al numero di feature, quindi , per poterle rappresentare dobbiamo prima ridurre il numero delle dimensioni almeno a 3.

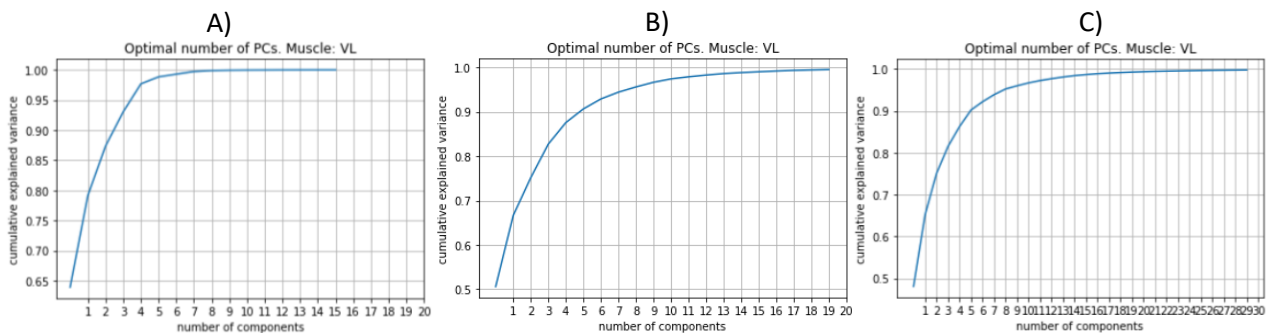
La Principal Component Analysis (PCA), è una tecnica che permette di ridurre le dimensioni di un set di dati mantenendo la maggior parte delle sue informazioni, ovvero massimizzando la varianza. Quello che tenta di fare è sintetizzare il contenuto informativo di tutte le feature in un sottogruppo, più ristretto e facilmente gestibile, di nuove feature.

Prima di tutto è necessario standardizzare il set di dati. Le texture feature, infatti, possono assumere valori molto differenti tra loro, e quelle con i valori più alti finirebbero per pesare di più sulle altre. Per la standardizzazione è stato utilizzato il metodo MinMaxScaling della libreria scikit-learn<sup>18</sup>.

Dopo la standardizzazione, bisogna chiedersi quale sia il numero più opportuno di principal component per mantenere quanta più informazione possibile. Per stabilire questo, è importante

<sup>18</sup> <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html>

comprendere il concetto di explained variance. L'explained variance è la varianza che può essere contenuta da una componente principale. Per ottenere una buona rappresentazione dei dati, è necessario che la cumulata dell'explained variance delle componenti principali sia pari ad almeno 0.6-0.8: questo significa che contengono almeno il 60-80% della varianza del set di dati, e quindi sono abbastanza rappresentative. Più è numeroso il dataset o il subset di feature, maggiore è il numero di componenti principali richieste (figura 7.9).



**Figura 7.9:** Cumulative explained variance in funzione del numero di componenti principali. All'aumentare del numero di feature considerate, sono necessarie più componenti principali per rappresentare correttamente il dataset. A) Considerando 16 feature; B) Considerando 55 feature; C) Considerando 86 feature.

Nel caso del muscolo Vasto Laterale, due o tre principal component risultano abbastanza rappresentative: per il subset estratto con il t-test, contenente 16 feature, 2 principal component risultano più che sufficienti, mentre per il set iniziale (86 feature), ne basteranno 3.

Gli step per ottenere le principal components sono elencati di seguito [64] :

1. Si ricava la matrice della covarianza del set di feature standardizzato. La matrice della covarianza è una matrice quadrata e simmetrica perché la covarianza gode della proprietà commutativa;
2. Si calcolano gli autovalori della matrice di covarianza;
3. Si trovano gli autovettori associati ad ogni autovalore;
4. Si ricavano i vettori di Eigen (Eigenvector) e si ordinano in maniera decrescente;
5. Se N è il numero delle principal component che si vogliono ricavare, si prendono i primi N Eigenvector;
6. Si calcolano le principal component moltiplicando la matrice delle texture feature standardizzata con la matrice contenente gli Eigenvector selezionati.

Le componenti principali, dunque, sono ricavate dalla combinazione lineare delle feature iniziali moltiplicate per dei "pesi". La prima principal component (PC1) racchiude la maggior parte della varianza, che poi va a diminuire a mano a mano che il numero aumenta. Inoltre, le feature moltiplicate per i coefficienti più alti (in valore assoluto) sono quelle più rilevanti.

Siano:

- $a$  = Peso Massimo;
- $b = a - \xi$ , con  $\xi$  = quantità piccola scelta arbitrariamente.

Considerando le feature moltiplicate per i coefficienti compresi tra  $a$  e  $b$  troviamo quelle più rappresentative.

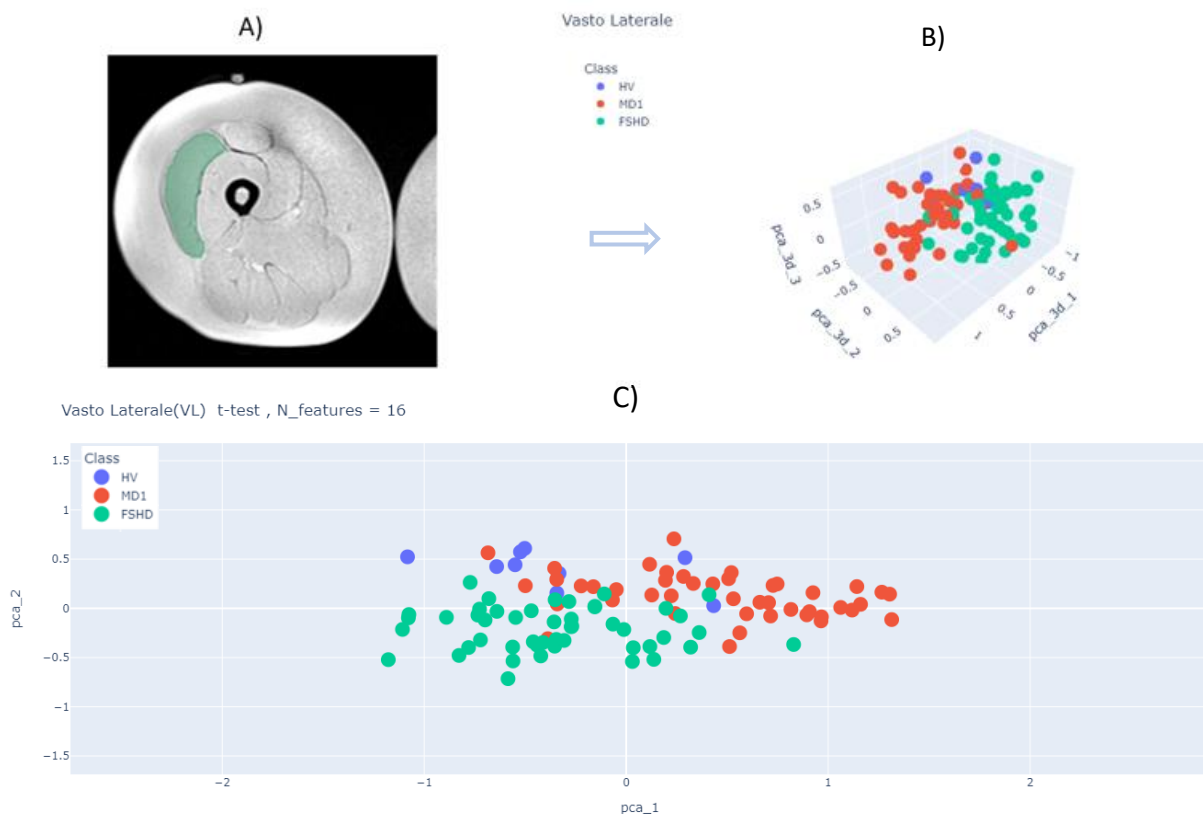
Nel caso del Vasto Laterale, scegliendo  $\xi = 0.025$ , le features più rappresentative tra quelle selezionate con il t-test, sono elencate in tabella 7.2.

Tabella 7.2: Feature più rappresentative nel caso del subset estratto con il t-test.

PC1		PC2	
feature name	loading	feature name	loading
glrlm_RunPercentage	0.361499	glszm_SizeZoneNonUniformityNormalized	0.309773
glrlm_RunLengthNonUniformityNormalized	0.336704	glszm_SmallAreaEmphasis	0.298357

#### 7.2.4 Visualizzazione

Una volta calcolate le componenti principali ed eliminati gli outlier, è possibile visualizzare i risultati.



**Figura 7.10:** Visualizzazione delle texture feature selezionate con il t-test. A) Partiamo dall'immagine MRI, e dalla label manuale che identifica il muscolo VL. B) Per racchiudere la maggior parte delle informazioni del set di feature ricavato, scegliamo inizialmente 3 principal component. C) Per una visualizzazione più comoda, le proiettiamo in 2D (scegliendo 2 principal component).





## Capitolo 8

### Risultati

Nel seguente capitolo vengono descritti i risultati ottenuti in questo lavoro. In primo luogo, saranno confrontate le diverse reti utilizzate; successivamente, si discuteranno le considerazioni riguardo la Texture Analysis.

Vengono messe a confronto 5 Deep Learning Neural Network, diverse per:

- Tipologia di segmentazione: 2D, 3D;
- Tipologia di modello: U-Net, K-Net;
- Tipologia di encoder: ResNet, SwinTransformer.

La Texture Analysis è stata condotta con le segmentazioni manuali sui volumi di gamba e coscia divisi per condizione muscolare:

- Distrofia Facio-Scapolo-Omerale (FSHD);
- Distrofia Miotonica di tipo 1 (MD1);
- Healthy Volunteers (HV).

### 8.1 Confronto Deep Learning Neural Network

#### 8.1.1 Confronto parziale (2D)

Per un confronto preliminare in 2D , è stata effettuata l'inference estraendo dai volumi del Test Set le patch corrispondenti a quelle utilizzate per l'allenamento. Per un confronto equilibrato, anche dalle segmentazioni automatiche 3D è stato estratto lo stesso subset di slice. Dopo questa selezione la numerosità delle patch ammontava a:

- 50 per la coscia;
- 48 per la gamba.

Quindi, i risultati sono stati calcolati per ogni patch e poi mediati rispetto ai muscoli. In fine, è stata ottenuta una media complessiva per confrontare in maniera più generica tutti i modelli. I risultati sono mostrati nelle tabelle 8.1 e 8.2.

Tabella 8.1: DSC e HD95 (2D) per i muscoli della Coscia.

DSC (%)						HD 95 (pixel)					
	2D			3D			2D			3D	
Muscle	U-Net-ResNet	K-Net-Swin	K-Net-ResNet	U-Net	SwinUNETR	Muscle	U-Net-ResNet	K-Net-Swin	K-Net-ResNet	U-Net	SwinUNETR
VL	88.41 ± 0.89	88.17 ± 0.84	87.44 ± 1.18	91.07 ± 0.46	86.97 ± 0.93	VL	3.91 ± 0.73	5.1 ± 0.95	4.6 ± 0.75	3.32 ± 0.29	4.78 ± 0.78
VI	88.04 ± 0.98	87.68 ± 0.6	88.15 ± 0.47	88.87 ± 0.82	88.22 ± 1.1	VI	3.1 ± 0.21	3.61 ± 0.24	3.33 ± 0.3	3.32 ± 0.22	3.66 ± 0.26
VM	90.22 ± 1.15	89.54 ± 1.41	89.28 ± 1.39	87.14 ± 1.76	88.78 ± 0.95	VM	3.01 ± 0.27	3.42 ± 0.42	3.12 ± 0.24	5.28 ± 1.1	3.8 ± 0.41
RF	82.57 ± 1.67	72.69 ± 4.53	84.09 ± 1.93	76.26 ± 4.75	82.33 ± 3.25	RF	3.0 ± 0.16	5.01 ± 0.65	2.72 ± 0.38	3.73 ± 0.44	2.7 ± 0.41
SAR	84.29 ± 1.46	82.63 ± 2.2	84.06 ± 1.5	79.26 ± 3.35	82.53 ± 3.29	SAR	2.77 ± 0.41	3.53 ± 0.61	2.63 ± 0.33	3.92 ± 0.71	3.11 ± 0.6
GRA	86.63 ± 2.73	82.47 ± 1.43	84.03 ± 2.68	82.58 ± 1.72	85.82 ± 1.62	GRA	2.19 ± 0.3	2.84 ± 0.16	2.42 ± 0.3	3.8 ± 0.73	2.23 ± 0.12
SEMI-MB	64.64 ± 4.58	64.81 ± 6.29	69.18 ± 5.86	60.68 ± 6.43	64.35 ± 6.17	SEMI-MB	8.25 ± 1.72	7.35 ± 1.87	7.25 ± 1.96	10.35 ± 2.68	7.9 ± 2.35
SEMI-TN	79.87 ± 4.79	74.78 ± 6.62	77.17 ± 6.14	80.33 ± 4.77	77.14 ± 5.92	SEMI-TN	4.46 ± 0.85	5.18 ± 1.17	5.49 ± 1.47	4.28 ± 1.0	4.5 ± 0.97
BF-CL	80.44 ± 3.3	82.59 ± 3.9	81.4 ± 3.21	83.65 ± 3.39	82.06 ± 4.48	BF-CL	4.45 ± 0.47	4.32 ± 0.9	4.17 ± 0.66	3.77 ± 0.68	4.63 ± 1.18
BF-CB	67.52 ± 8.25	73.63 ± 3.37	75.53 ± 3.0	66.93 ± 6.31	71.16 ± 4.49	BF-CB	5.62 ± 2.18	4.08 ± 0.86	4.5 ± 0.91	5.42 ± 1.14	5.15 ± 0.68
ADD	74.3 ± 4.97	73.44 ± 4.38	75.59 ± 3.74	72.94 ± 4.21	75.34 ± 4.06	ADD	7.21 ± 1.09	8.26 ± 1.51	8.03 ± 1.36	8.86 ± 1.28	6.97 ± 1.23
AL	75.6 ± 2.4	73.7 ± 10.76	76.98 ± 3.06	79.83 ± 11.86	75.87 ± 5.03	AL	3.26 ± 0.43	3.86 ± 0.73	3.59 ± 0.47	3.14 ± 0.53	4.45 ± 1.05
Muscle Avg	80.21 ± 8.27	78.84 ± 7.72	<b>81.08 ± 6.17</b>	79.13 ± 8.90	80.05 ± 7.47	Muscle Avg	<b>4.27 ± 1.87</b>	4.71 ± 1.63	4.32 ± 1.80	4.93 ± 2.32	4.49 ± 1.64

Tabella 8.2: DSC e HD95 (2D) per i muscoli della Gamba.

DSC (%)						HD 95 (pixel)					
	2D			3D			2D			3D	
Muscle	U-Net-ResNet	K-Net-Swin	K-Net-ResNet	U-Net	SwinUNETR	Muscle	U-Net-ResNet	K-Net-Swin	K-Net-ResNet	U-Net	SwinUNETR
TA	88.8 ± 1.66	85.96 ± 1.63	87.45 ± 1.65	87.43 ± 1.57	88.02 ± 1.82	TA	2.63 ± 0.49	3.27 ± 0.67	2.46 ± 0.25	2.9 ± 0.37	2.58 ± 0.34
TP	86.28 ± 0.9	83.48 ± 1.21	85.03 ± 0.62	87.01 ± 1.69	86.76 ± 1.9	TP	3.8 ± 0.56	3.68 ± 0.36	4.06 ± 0.51	2.96 ± 0.28	3.2 ± 0.4
ES-L	81.21 ± 1.57	82.28 ± 0.88	78.97 ± 1.54	80.24 ± 2.4	80.97 ± 2.48	ES-L	2.93 ± 0.37	2.71 ± 0.16	3.08 ± 0.25	3.03 ± 0.41	2.96 ± 0.37
PER	90.63 ± 0.62	89.24 ± 0.88	88.37 ± 1.09	88.99 ± 1.18	89.57 ± 1.97	PER	1.97 ± 0.17	2.37 ± 0.25	2.62 ± 0.23	2.22 ± 0.3	2.1 ± 0.35
FL-D	70.84 ± 3.36	66.53 ± 3.27	68.73 ± 4.0	66.47 ± 4.2	69.62 ± 4.0	FL-D	3.34 ± 0.31	3.77 ± 0.23	3.64 ± 0.34	3.52 ± 0.26	3.46 ± 0.21
SOL	85.57 ± 1.34	85.47 ± 1.84	85.42 ± 2.59	84.93 ± 1.84	83.96 ± 1.99	SOL	6.49 ± 1.13	5.17 ± 0.49	5.48 ± 0.53	6.17 ± 1.45	6.92 ± 0.96
GM	67.75 ± 9.38	74.48 ± 5.16	68.72 ± 6.8	80.7 ± 5.37	78.62 ± 7.4	GM	3.07 ± 0.35	4.02 ± 0.45	3.22 ± 0.28	3.25 ± 0.29	3.3 ± 0.63
GL	59.0 ± 9.5	60.82 ± 8.28	65.7 ± 7.75	73.52 ± 12.67	63.55 ± 8.93	GL	3.23 ± 0.28	3.69 ± 0.26	3.56 ± 0.29	3.52 ± 0.58	4.82 ± 0.72
FL-A	46.11 ± 12.46	61.85 ± 4.29	60.23 ± 3.59	54.36 ± 21.22	49.05 ± 16.34	FL-A	8.65 ± 2.15	7.17 ± 0.6	7.08 ± 0.96	6.2 ± 2.37	9.64 ± 3.75
Muscle Avg	75.13 ± 15.30	76.68 ± 11.06	76.51 ± 10.73	<b>78.18 ± 11.53</b>	76.68 ± 13.50	Muscle Avg	4.01 ± 2.15	3.98 ± 1.44	3.91 ± 1.49	<b>3.75 ± 1.43</b>	4.33 ± 2.45

### 8.1.2 Confronto completo (3D)

Per la valutazione in 3D, è stata eseguita l'inferenza su tutte le patch e i volumi del Test Set. Dopo aver ripristinato i volumi slice per slice e averli croppati, come precedentemente descritto nel paragrafo 6.4, sono stati calcolati il DSC e l'HD95. I risultati riportati, rappresentano la media calcolata per ciascuna label rispetto tutti i soggetti del Test Set. Il confronto per coscia e gamba è riportato nelle Tabelle 8.3 e 8.4, rispettivamente, in cui vengono anche evidenziate le criticità comuni ai diversi modelli.

Tabella 8.3: Confronto performance delle diverse reti sul Test Set sui muscoli della coscia. **VL**: Vasto Laterale, **VI**: Vasto Intermedio, **VM**: Vasto Mediale, **RF**: Retto femorale, **SAR**: Sartorio, **GRA**: Gracile, **SEMI-MB**: Semimembranoso, **SEMI-TN**: Semitendinoso, **BF-CL**: Bicipite femorale - capo lungo, **BF-CB**: Bicipite femorale - capo breve, **ADD**: Grande Adduttore + adduttore corto, **AL**: Adduttore Lungo.

DSC (%)						HD 95 (voxel)					
	2D			3D			2D			3D	
Muscle	U-Net-ResNet	K-Net-Swin	K-Net-ResNet	U-Net	SwinUNETR	Muscle	U-Net-ResNet	K-Net-Swin	K-Net-ResNet	U-Net	SwinUNETR
VL	89.19 ± 0.83	88.82 ± 0.9	88.98 ± 0.68	90.79 ± 0.44	87.94 ± 0.67	VL	3.14 ± 0.62	3.45 ± 0.43	3.87 ± 0.92	2.8 ± 0.15	4.15 ± 0.27
VI	88.45 ± 1.22	88.37 ± 0.35	88.44 ± 0.62	88.42 ± 0.92	88.8 ± 0.91	VI	2.27 ± 0.07	2.28 ± 0.17	2.44 ± 0.21	2.99 ± 0.03	3.05 ± 0.18
VM	90.82 ± 0.9	90.43 ± 1.46	90.33 ± 1.52	89.34 ± 0.87	90.62 ± 0.6	VM	2.47 ± 0.17	2.32 ± 0.13	2.37 ± 0.27	3.35 ± 0.36	2.92 ± 0.16
RF	83.57 ± 0.78	78.83 ± 1.32	82.61 ± 1.65	78.01 ± 2.47	83.02 ± 2.61	RF	3.07 ± 0.17	5.4 ± 0.69	5.19 ± 1.84	<b>7.72 ± 1.24</b>	5.22 ± 1.38
SAR	83.77 ± 0.55	82.93 ± 1.5	84.07 ± 0.86	79.54 ± 3.15	83.85 ± 2.15	SAR	3.82 ± 0.73	2.89 ± 0.39	2.41 ± 0.1	3.39 ± 0.59	3.2 ± 0.33
GRA	84.95 ± 2.49	84.06 ± 1.24	84.56 ± 2.05	81.26 ± 1.52	84.77 ± 1.3	GRA	2.38 ± 0.49	2.81 ± 0.55	2.34 ± 0.33	4.9 ± 1.15	3.15 ± 0.26
SEMI-MB	<b>70.51 ± 4.06</b>	<b>71.01 ± 5.36</b>	<b>73.16 ± 4.88</b>	<b>64.44 ± 4.81</b>	<b>69.88 ± 6.81</b>	SEMI-MB	<b>8.23 ± 2.24</b>	<b>7.26 ± 2.31</b>	<b>7.42 ± 2.43</b>	<b>9.06 ± 1.86</b>	<b>7.08 ± 1.77</b>
SEMI-TN	81.31 ± 2.79	80.0 ± 4.19	81.52 ± 4.14	81.04 ± 3.82	79.05 ± 5.57	SEMI-TN	<b>6.47 ± 1.86</b>	<b>5.89 ± 1.71</b>	<b>11.04 ± 5.69</b>	5.9 ± 1.5	6.06 ± 1.5
BF-CL	83.15 ± 3.13	84.58 ± 2.77	83.89 ± 2.34	84.35 ± 3.85	83.11 ± 4.59	BF-CL	3.05 ± 0.45	3.25 ± 0.56	3.05 ± 0.44	4.01 ± 0.87	4.77 ± 1.38
BF-CB	<b>71.78 ± 2.77</b>	<b>73.44 ± 2.75</b>	<b>75.7 ± 2.09</b>	<b>73.18 ± 2.11</b>	<b>69.04 ± 5.27</b>	BF-CB	5.12 ± 1.01	4.28 ± 0.84	4.61 ± 0.52	5.94 ± 0.81	<b>6.97 ± 1.2</b>
ADD	76.68 ± 4.61	<b>74.06 ± 5.94</b>	<b>76.05 ± 5.05</b>	<b>74.07 ± 5.47</b>	<b>73.78 ± 6.52</b>	ADD	<b>6.34 ± 1.37</b>	<b>8.21 ± 2.42</b>	<b>7.69 ± 2.21</b>	<b>8.71 ± 2.24</b>	<b>9.62 ± 3.3</b>
AL	<b>76.77 ± 0.6</b>	74.6 ± 1.21	80.02 ± 1.71	79.19 ± 1.23	81.7 ± 1.23	AL	4.95 ± 0.74	5.12 ± 0.98	3.14 ± 0.47	4.23 ± 0.62	4.19 ± 0.39
Muscle Avg	<b>81.75 ± 6.6</b>	80.93 ± 6.62	82.44 ± 5.47	80.3 ± 7.53	81.3 ± 7.1	Muscle Avg	<b>4.28 ± 1.94</b>	4.43 ± 1.96	4.63 ± 2.75	5.25 ± 2.22	5.03 ± 2.07

Tabella 8.4: Confronto performance delle diverse reti sul Test Set sui muscoli della gamba. **TA**: Tibiale Anteriore , **TP**: Tibiale Posteriore , **ES-L**: Estensore Lungo , **PER**: Peroneo , **FL-D**: Flessore Lungo delle dita, **SOL**: Soleo , **GM**: Gastrocnemio Mediale , **GL**: Gastrocnemio Laterale , **FL-A**: Flessore Lungo Dell'alluce.

Muscle	DSC (%)				
	2D			3D	
	U-Net-ResNet	K-Net-Swin	K-Net-ResNet	U-Net	SwinUNETR
TA	87.99 ± 0.47	87.67 ± 1.16	87.57 ± 0.76	87.75 ± 0.53	87.92 ± 0.73
TP	80.81 ± 3.31	84.68 ± 0.35	85.65 ± 0.54	86.73 ± 1.11	86.87 ± 1.24
ES-L	81.64 ± 0.6	81.89 ± 0.4	78.77 ± 0.7	79.63 ± 1.37	80.16 ± 1.54
PER	90.35 ± 0.44	89.75 ± 0.49	88.79 ± 0.46	89.75 ± 0.76	90.54 ± 1.08
FL-D	<b>69.4 ± 2.12</b>	<b>54.73 ± 4.53</b>	<b>55.37 ± 9.58</b>	<b>65.56 ± 2.0</b>	<b>67.56 ± 1.95</b>
SOL	86.95 ± 0.9	87.78 ± 1.34	87.37 ± 1.76	86.68 ± 1.19	86.23 ± 1.57
GM	87.7 ± 0.93	86.53 ± 0.81	87.36 ± 0.51	86.82 ± 1.51	85.43 ± 2.61
GL	80.62 ± 1.08	79.05 ± 2.03	81.36 ± 1.75	81.3 ± 1.17	78.77 ± 3.31
FL-A	<b>47.6 ± 10.14</b>	<b>63.9 ± 4.23</b>	<b>56.96 ± 2.81</b>	<b>44.44 ± 16.35</b>	<b>46.08 ± 15.55</b>
Muscle Avg	79.23 ± 13.41	<b>79.55 ± 12.14</b>	78.8 ± 13.24	78.74 ± 14.8	78.84 ± 14.07

Muscle	HD 95 (voxel)				
	2D			3D	
	U-Net-ResNet	K-Net-Swin	K-Net-ResNet	U-Net	SwinUNETR
TA	3.19 ± 0.67	1.98 ± 0.17	2.35 ± 0.18	3.32 ± 0.26	2.7 ± 0.14
TP	5.95 ± 0.96	2.73 ± 0.3	2.59 ± 0.27	3.42 ± 0.39	3.36 ± 0.21
ES-L	2.77 ± 0.35	2.68 ± 0.14	3.91 ± 1.21	3.86 ± 0.35	3.25 ± 0.17
PER	1.64 ± 0.22	1.85 ± 0.2	1.98 ± 0.16	2.31 ± 0.37	1.89 ± 0.27
FL-D	3.13 ± 0.58	5.85 ± 0.78	5.91 ± 1.29	5.08 ± 0.56	5.56 ± 0.36
SOL	4.7 ± 0.79	3.02 ± 0.41	3.31 ± 0.57	4.74 ± 0.68	4.89 ± 0.64
GM	4.83 ± 1.35	3.47 ± 0.26	3.2 ± 0.49	3.18 ± 0.13	3.58 ± 0.37
GL	4.85 ± 1.45	3.65 ± 0.53	2.54 ± 0.16	3.59 ± 0.15	5.33 ± 0.4
FL-A	<b>8.73 ± 1.22</b>	<b>5.48 ± 1.11</b>	<b>6.85 ± 0.99</b>	<b>12.12 ± 3.16</b>	<b>12.13 ± 3.39</b>
Muscle Avg	4.42 ± 2.09	<b>3.41 ± 1.41</b>	3.63 ± 1.68	4.62 ± 2.93	4.74 ± 3.03

Nel caso della coscia, risultano migliori le reti con encoder ResNet e, in particolare, K-Net ottiene il miglior DSC, mentre U-Net la miglior HD95:

- Il DSC della K-Net supera quello della U-Net dello 0.79%;
- La HD della U-Net è più bassa dell'8.55%.

Pertanto, il compromesso migliore risulta la U-Net con encoder ResNet50 (figura 8.1), poiché richiede anche meno risorse computazionali (50 epoche contro le 150 epoche utilizzate per il training della K-Net).

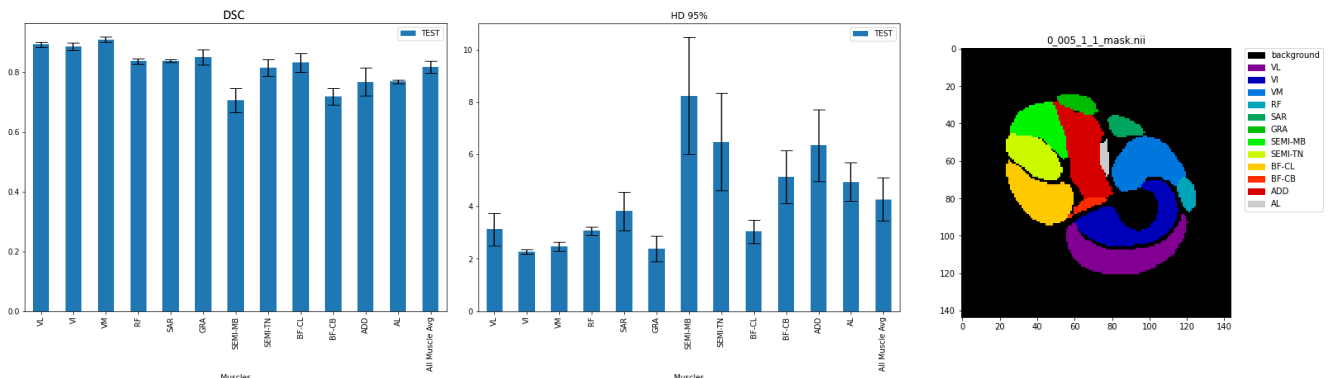


Figura 8.1: Confronto grafico delle performance della rete U-Net con encoder ResNet50 per i muscoli della coscia.

Per la gamba, invece, ha ottenuto i migliori risultati in entrambe le metriche la rete K-Net con encoder SwinTransformer (figura 8.2).

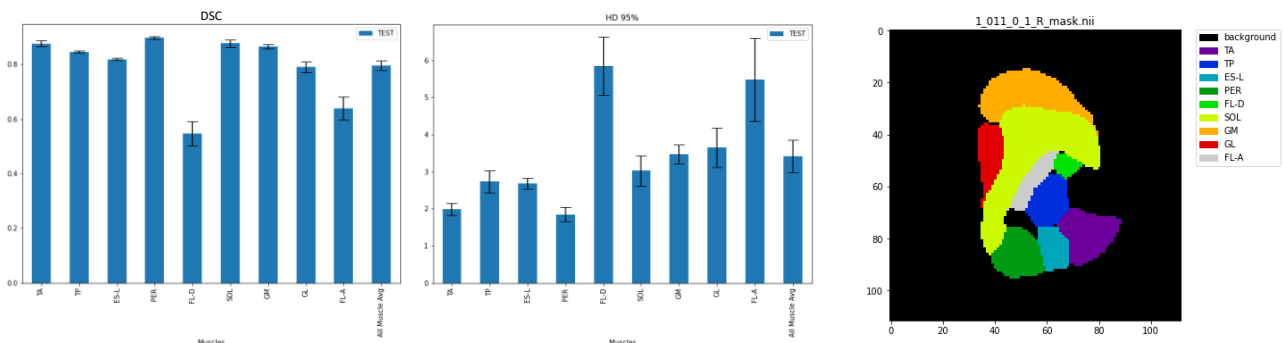


Figura 8.2: Confronto grafico delle performance della rete K-Net con encoder SwinTransformer per i muscoli della gamba.

### 8.1.3 Considerazioni sui singoli muscoli

Dopo aver confrontato le performance complessive delle reti, in questo paragrafo verranno discusse le osservazioni sulle label considerate individualmente, osservando i risultati ricavati dalle segmentazioni 3D.

Alcune di esse, vengono segmentate molto bene, altre non in maniera ottimale.

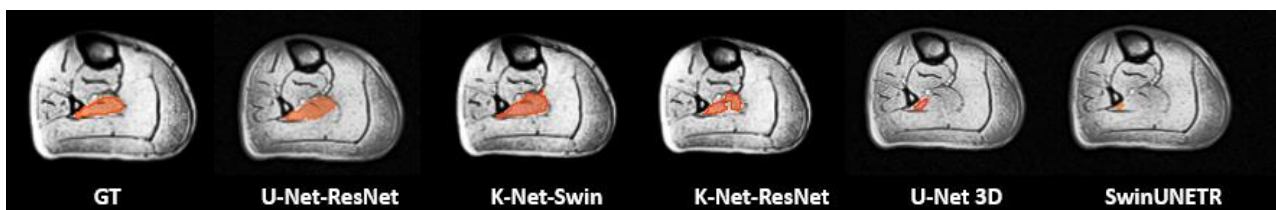
Sulla base dei risultati ottenuti e di un'indagine visiva, le metriche saranno definite come "alte" o "basse" rispetto alle seguenti soglie:

- DSC = 75%;
- HD95 = 5.5 voxel.

Notiamo che, per la maggior parte dei casi, le reti ottengono buoni valori per quanto riguarda la sovrapposizione e la qualità della segmentazione. Esse mostrano dei limiti nel segmentare i seguenti muscoli:

- GAMBA
  - Flessore Lungo dell'Alluce.
  - Flessore Lungo delle Dita.
- COSCIA
  - Grande Adduttore + Adduttore Breve;
  - Semimembranoso;
  - Semitendinoso.

Il Flessore Lungo dell'Alluce è una delle label aggiunte manualmente ad alcuni dei volumi presenti nel dataset, per uniformarne la rappresentazione (paragrafo 5.2.1). Probabilmente, la segmentazione non rigorosa della regione del muscolo e la differenza rispetto alle label contrassegnate dai clinici, non ha permesso a nessuna rete di individuarlo correttamente. La rete che si è dimostrata più robusta alle differenze è la K-Net con encoder SwinTransformer (DSC =  $63.9 \pm 4.23$  %; HD95 =  $5.48 \pm 1.11$  voxel).

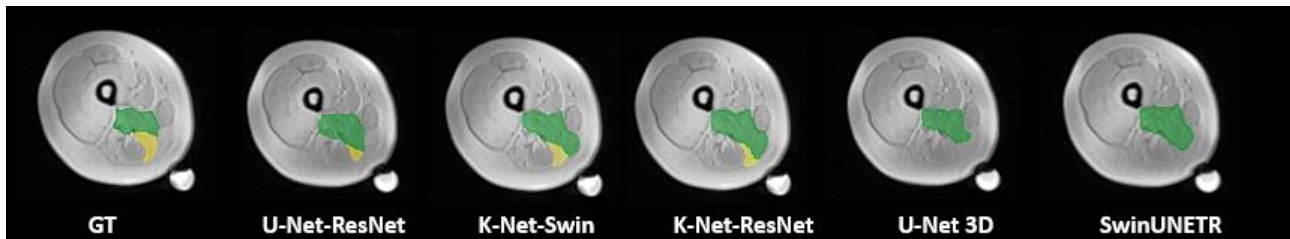


**Figura 8.3:** Confronto tra le segmentazioni della label Flessore Lungo dell'Alluce sulla slice 32 del volume 0\_005\_0\_1.nii

Il Flessore Lungo delle Dita, è una label piccola, quindi ha risentito negativamente dell'effetto del post-processing, che però è stato mantenuto comunque perché migliorava le performance complessive delle reti.

Anche il Grande Adduttore (ADD) fa parte delle label rifinite manualmente. Quello che risulta dall'indagine visiva (figura 8.4) è che questa label si espande verso il basso, "invadendo" la regione del muscolo Semimembranoso; pertanto, anche le metriche riguardo quest'ultimo risultano

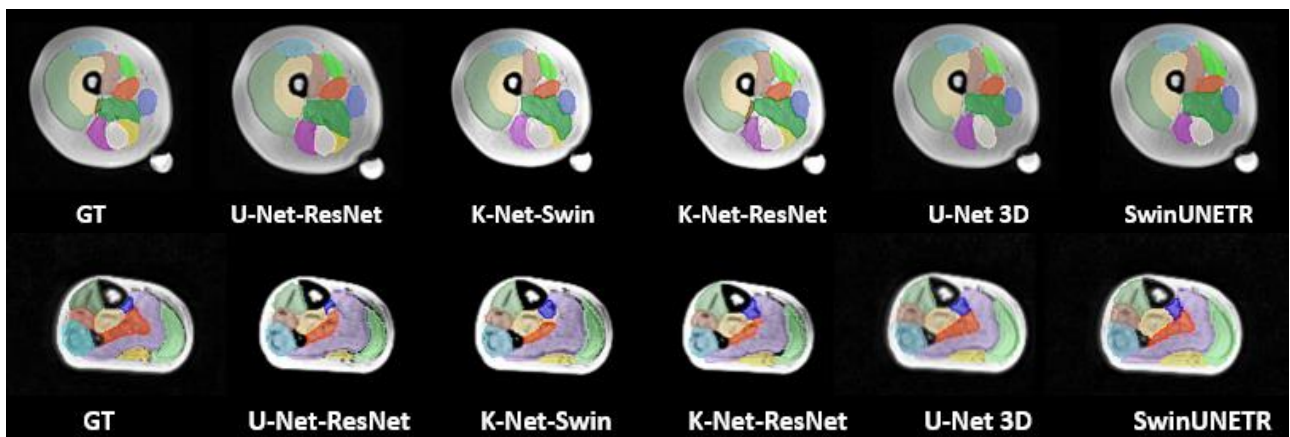
compromesse. In alcune slice, le reti 3D non riescono neanche a segmentarlo, risultando inevitabilmente meno precise rispetto a quelle bidimensionali.



**Figura 8.4:** Confronto tra le segmentazioni delle label Grande Adduttore + Adduttore Breve (in verde) e Semimembranoso (in giallo) sulla slice 33 del volume 1\_011\_1\_1\_L.nii

Il Bicipite Femorale – capo breve, invece, presenta un DSC basso. Probabilmente, hanno influito le piccole dimensioni della label e la presenza ridotta all’interno del volume.

Tra i muscoli che vengono segmentati particolarmente bene abbiamo i Vasti (Vasto Laterale, Vasto Mediale, Vasto Intermedio), il Sartorio e il Gracile per la coscia; mentre nella gamba notiamo il Tibiale Anteriore, il Tibiale Posteriore e il Peroneo.

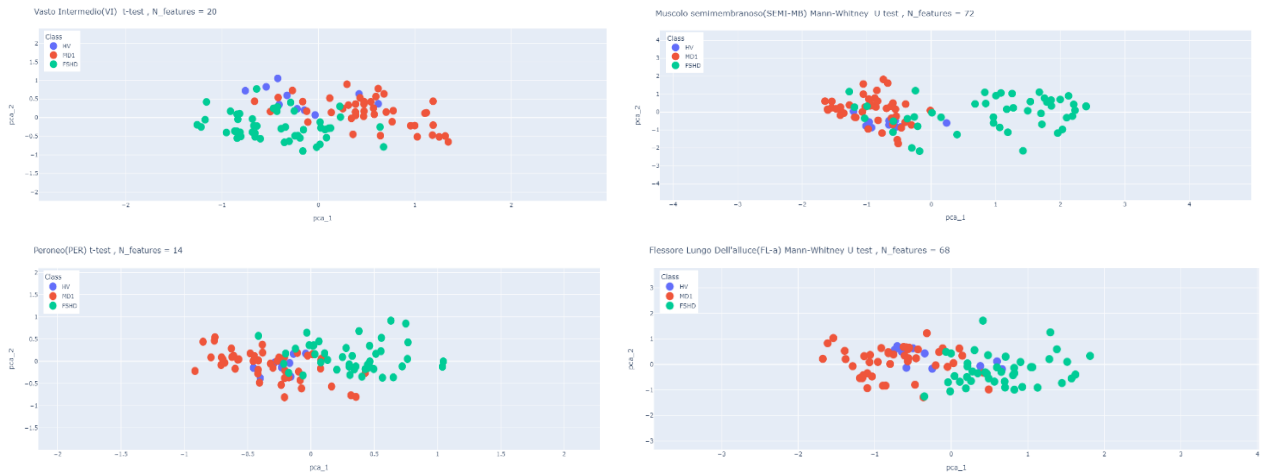


**Figura 8.5:** confronto complessivo su tutte le label. Riga superiore (coscia) con il volume 1\_011\_1\_1\_L.nii; Riga inferiore (gamba) con il volume 1\_011\_0\_1\_R.nii .

## 8.2 Texture Analysis

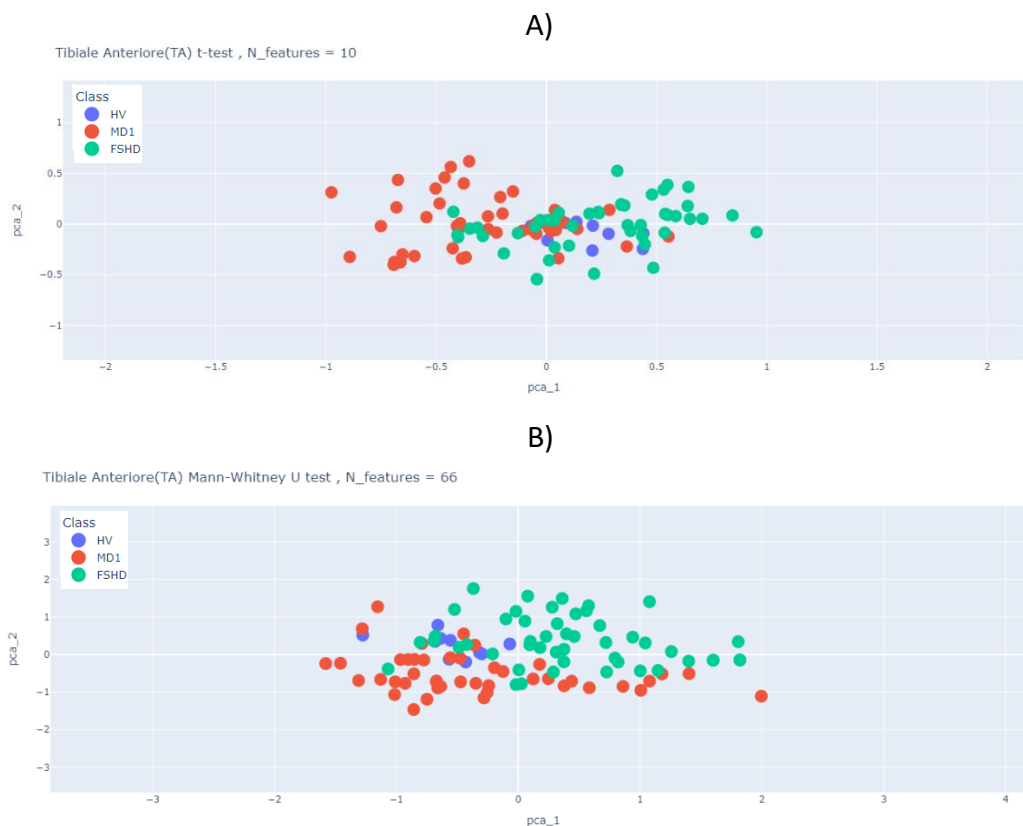
In questo paragrafo verranno descritti i risultati ottenuti per la Texture Analysis, discutendo i casi più emblematici per coscia e gamba. Alla fine, mostreremo l’utilità dell’analisi per differenziare le diverse condizioni muscolari. Per una visione completa su tutti i muscoli, si rimanda all’appendice A.

La distinzione tra muscoli colpiti da MD1 ed FSHD, è apparsa già dalle prime fasi dell’analisi in cui sono state considerate tutte le texture feature, per poi migliorare dopo la feature selection (figura 8.6).



**Figura 8.6:** Alcuni dei migliori esempi di distinzione del muscolo nelle tre condizioni (si nota in particolare la distinzione tra le classi MD1 ed FSHD).

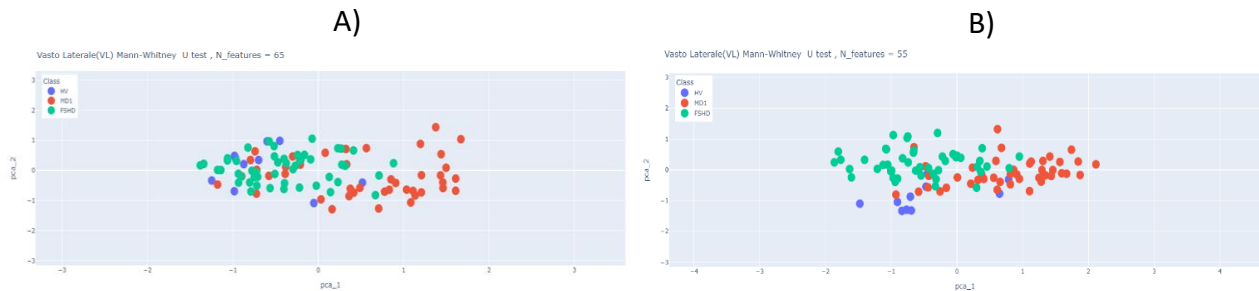
In alcuni casi, il subset di feature estratto utilizzando il t-test risultava troppo poco rappresentativo. Molte feature venivano scartate perché non seguivano una distribuzione normale, e il sottogruppo rimanente non riusciva a distinguere le classi in maniera appropriata, come nel Tibiale Anteriore (figura 8.7.A). Dunque, è stato utilizzato un altro test statistico, che non ha bisogno di confermare l'ipotesi di normalità: il test U di Mann-Witney, che riduce meno drasticamente il numero di feature per ciascun muscolo, risultando più adeguato in alcuni casi (figura 8.7.B).



**Figura 8.7:** Esempio di come influisca il corretto subset di features sulla separazione delle classi. A) il t-test ha selezionato un set di feature troppo ristretto. B) Il test di Mann-Whitney ha estratto un subset più rappresentativo che ha permesso di distinguere meglio le classi.

### 8.2.1 L'importanza del pre-processing

Quando si conduce un'analisi della tessitura, il pre-processing è fondamentale per uniformare i dati e fare in modo che le differenze evidenziate riguardino solo le grandezze di interesse, e non dipendano da fattori esterni, come i differenti strumenti di acquisizione. Ricoprono grande importanza, in particolare, gli step di filtraggio per eliminare la polarizzazione del campo magnetico (Bias Field correction con N4ITK), e l'histogram matching. In figura, viene mostrato l'impatto dell'histogram matching sulla separazione tra le distrofie.



**Figura 8.8:** Effetto dell'histogram matching. A) Senza histogram matching, le classi non sono raggruppate al meglio. B) La combinazione dell'histogram matching e della feature selection ha permesso una separazione tra le distrofie.





## Capitolo 9

### Discussione

Le distrofie muscolari sono malattie rare, pertanto non sono presenti molti studi riguardanti l'imaging muscolare di pazienti affetti da queste patologie [4]. Per lo studio delle distrofie muscolari in questo lavoro, sono stati considerati i volumi MRI degli arti inferiori, in cui la compromissione dei muscoli era abbastanza riconoscibile. Oltre la variabilità dettata dalle caratteristiche cliniche dei soggetti che compongono il dataset, anche le caratteristiche tecniche dei volumi erano abbastanza eterogenee, a livello di dimensioni, spacing, istogrammi delle luminosità ed effetti indesiderati tipici delle immagini MRI come la polarizzazione del campo magnetico, anche detta Bias Field Effect. Si tratta di un rumore a bassa frequenza dovuto alla disomogeneità del campo magnetico, presente soprattutto negli strumenti più datati. Nell'utilizzo dell'immagine in clinica, non è un problema, tuttavia, costituisce un ostacolo per le operazioni che tengono conto dell'intensità dei voxel come la segmentazione o la texture analysis. Per eliminarlo, i volumi sono stati elaborati con il filtro N4ITK. Il pre-processing era finalizzato a uniformare il dataset, dunque, è stato eseguito un flip in modo tale da ottenere tutti volumi raffiguranti (apparentemente) l'arto inferiore destro. È stato eseguito un ricampionamento e un crop per uniformare lo spacing e le dimensioni, rispettivamente. Queste operazioni sono state eseguite con MONAI, un insieme di framework specializzati nell'imaging medico. Per uniformare i volumi dal punto di vista dell'istogramma delle luminosità, invece, è stato utilizzato un algoritmo di histogram matching, implementato con ITK.

Nel caso delle distrofie muscolari, è importante riuscire a segmentare i muscoli individualmente poiché il coinvolgimento è diverso sia in base al tipo di distrofia che in base al paziente. Nel caso della coscia i muscoli da segmentare sono 12 mentre nel caso della gamba sono 9. Le immagini relative a cosce e gambe sono state trattate separatamente e sono state implementate due versioni per ciascuna rete. Il confronto tra diverse Deep Learning Neural Network, ha avuto un duplice scopo: fornire una panoramica completa attraverso differenti tipi di approcci, valutando i più recenti rispetto quelli più consolidati, e comprendere quali fossero i più adatti per la segmentazione della muscolatura di pazienti distrofici.

Con le reti 3D è stato possibile utilizzare direttamente i volumi, mentre per le reti 2D, sono state ricavate delle patch. Sebbene i modelli 2D abbiano ottenuto risultati migliori in termini di sovrapposizione (DSC) e precisione (HD95), quelli 3D hanno mostrato performance confrontabili utilizzando un pipeline più veloce.

Le reti neurali convoluzionali riescono bene a riconoscere le relazioni locali. Si basano sul concetto che pixel (o voxel) vicini hanno una più alta probabilità di essere correlati tra loro. A causa delle dimensioni limitate dei kernel, però, hanno dei limiti nel cogliere le dipendenze a lungo raggio. Le architetture basate sui transformer, invece, riescono a cogliere meglio le dipendenze globali. Avendo delle caratteristiche complementari e molto importanti nella segmentazione di immagini mediche, questi due modelli sono stati combinati in alcuni approcci più recenti. I risultati ottenuti,

hanno mostrato che queste soluzioni “ibride” sono migliori in certi casi: nella segmentazione 3D, la rete SwinUNETR ha superato complessivamente le performance della Residual Unet. Bisogna comunque considerare che le reti convoluzionali rimangono più efficienti dal punto di vista computazionale, infatti la SwinUNETR ha richiesto circa il doppio del training time della Unet.

Nel caso delle architetture 2D, per la coscia, dove i muscoli sono più grandi e hanno una forma più riconoscibile (come il Vasto Intermedio), hanno avuto la meglio le reti con encoder ResNet. Nello specifico, la rete K-Net ha mostrato il miglior Dice Coefficient complessivo, ma il compromesso migliore tra le due metriche utilizzate si è rivelata la U-Net con encoder ResNet50. Nel caso della gamba, invece, in cui i muscoli sono più piccoli e più simili tra loro, ha avuto la meglio la K-Net+UPerNet con encoder SwinTransformer, dimostrandosi la più robusta di fronte la label del Flessore Lungo dell’Alluce.

Dopo aver ottenuto le segmentazioni automatiche, è stato eseguito il post-processing per eliminare qualche imprecisione, come piccoli blob o buchi. Questo ha migliorato i risultati complessivi sia per la coscia che per la gamba, anche se le label più piccole, come il Flessore Lungo delle Dita, sono state penalizzate.

Le distrofie danneggiano i muscoli attraverso infiltrazioni di tessuto adiposo. Tali variazioni strutturali sono visibili dalle immagini MRI e possono essere quantificate attraverso la Texture Analysis. Pertanto, il secondo scopo di questo lavoro era comprendere le potenzialità della Texture Analysis nel distinguere lo stesso muscolo in base alla condizione del soggetto: affetto da MD1, FSHD o sano.

In questa fase, sono state utilizzate le segmentazioni manuali poiché circoscrivono più precisamente il muscolo, e non inglobano altri tessuti. Per ottenere dei risultati affidabili, è stato fondamentale il ruolo del pre-processing per mantenere solo quelle differenze che dipendessero dalle condizioni del muscolo. Poiché le texture feature utilizzano l’intensità dei voxel, la rimozione del bias field effect e l’histogram matching si sono rivelati fondamentali per discriminare al meglio le differenti classi. Sono state estratte 86 feature tra quelle del primo e del secondo ordine, utilizzando Pyradiomics, in modo tale da implementare un approccio standardizzato (librerie diverse calcolano i descrittori in maniera differente). Per eliminare le ridondanze che peggiorano la classificazione e aumentano la complessità, è stata eseguita la feature selection con dei test statistici: il t-test e il test di Wilcoxon-Mann-Whitney. Si tratta dell’alternativa non parametrica del t-test, ed è chiamato così poiché sia il test della somma dei ranghi di Wilcoxon, sia il test U di Mann-Whitney portano allo stesso risultato [65]. In questo lavoro è stato utilizzato il test U di Mann-Whitney, che ha permesso di selezionare un subset di feature più rappresentativo, laddove ne venivano scartate troppe per il t-test perché non seguivano una distribuzione normale. Attraverso la Principal Component Analysis, la dimensionalità del set di feature è stata ridotta, mantenendo quanta più informazione possibile, potendo così rappresentare i risultati. Si è vista una buona separazione tra le classi, ottenuta dalla maggior parte dei muscoli, anche quelli che venivano coinvolti da entrambe le distrofie.



## Capitolo 10

### Conclusioni e sviluppi futuri

I due scopi di questo lavoro erano segmentare individualmente i muscoli di coscia e gamba tramite metodi basati su Deep Learning, fornendo una panoramica su diversi tipi di approcci e valutare se la Texture Analysis fosse utile per riconoscere gli effetti di distrofie diverse sui singoli muscoli.

Entrambi i task hanno dato risultati promettenti. Nel primo, tutte le reti confrontate per la coscia hanno ottenuto performance intorno l'80%, mentre per la gamba intorno al 78% , in termini di Dice Coefficient. Sono emerse delle criticità che però possono essere facilmente superate, iniziando dall'utilizzo di segmentazioni manuali più precise.

Il secondo task ha mostrato che le texture feature possono differenziare lo stesso muscolo colpito da distrofie diverse. Questo risultato, potrebbe essere utilizzato come punto di partenza per implementare un classificatore in grado di riconoscere diversi tipi di distrofie, fornendo un importante supporto decisionale ai clinici. Di solito, la texture analysis viene utilizzata nelle diagnosi differenziali, dunque aver ottenuto una distinzione tra i due tipi di distrofie presenti nel dataset è un risultato interessante. Al contrario, le differenze tra muscoli sani e distrofici, non è stata rilevata, sebbene emergesse dall'ispezione visiva. A causa della presenza ridotta dei soggetti sani nel dataset, non si può affermare con certezza se si tratti di un limite della tecnica oppure sia dovuto all'inferiorità numerica del gruppo degli Healthy Volunteer. Sarebbe utile ripetere l'analisi considerando un gruppo di soggetti sani più ampio.

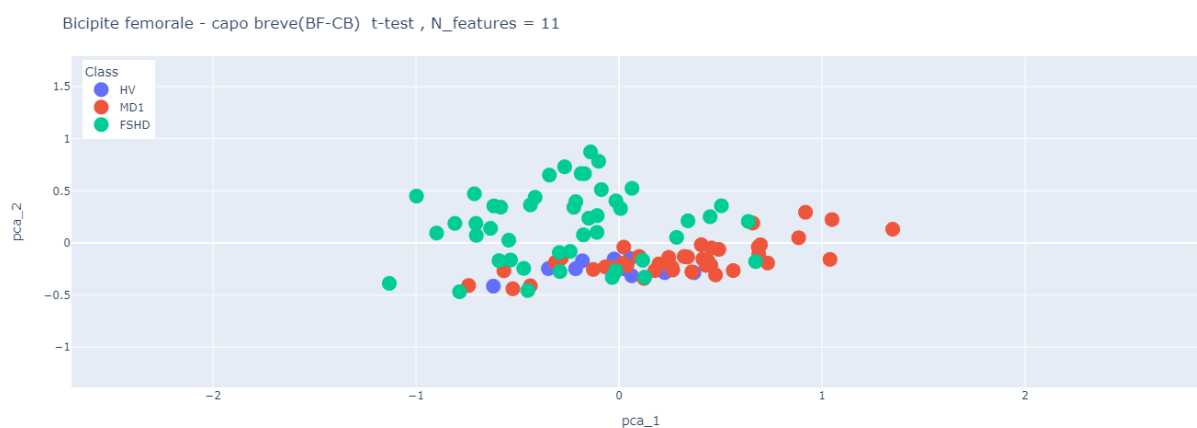
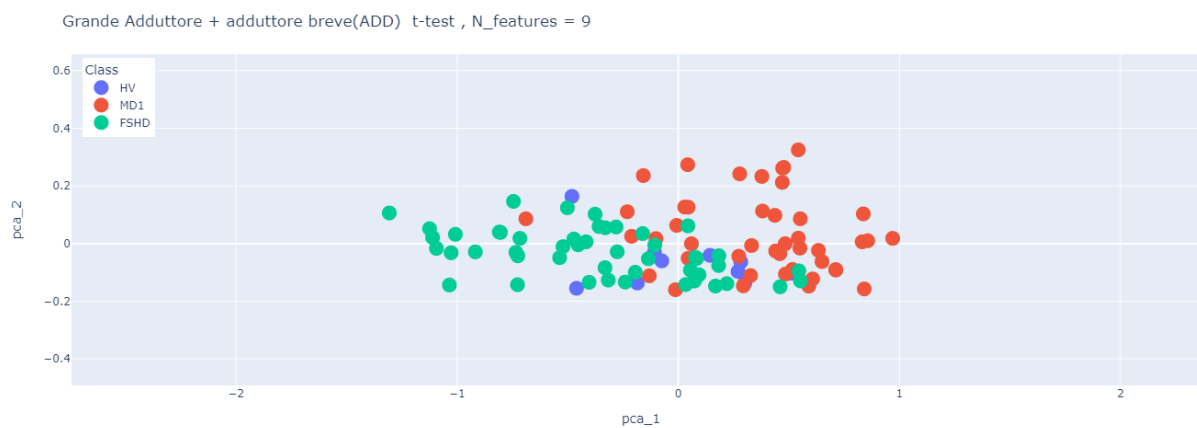
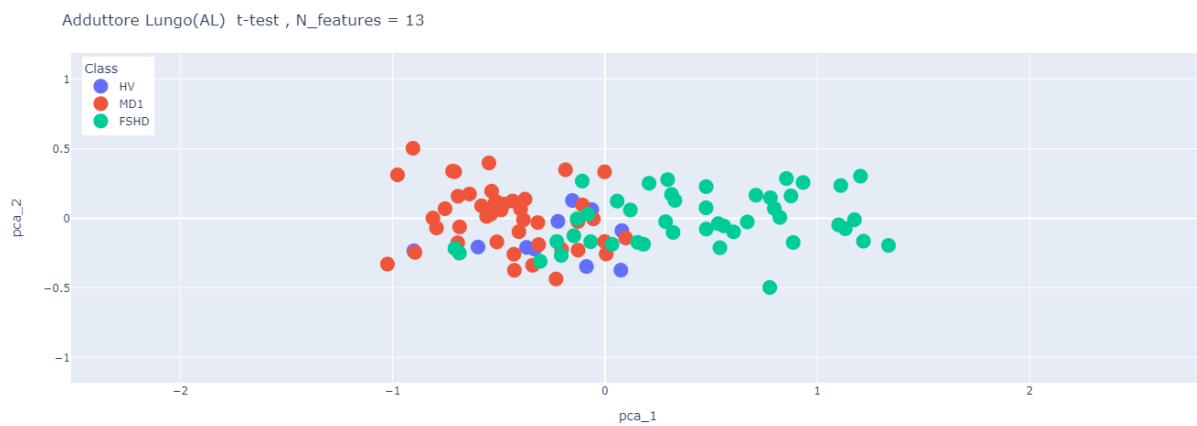
Dalle immagini MRI, in fine, possono essere estratti altri indicatori utili a quantificare la gravità della distrofia o monitorarne la progressione. Essi sono detti biomarcatori [15] , e non si tratta solamente di feature statistiche , come le texture feature, ma di parametri clinici. Uno dei più utilizzati è la frazione della percentuale di grasso [20],[15], che esprime quanto tessuto muscolare è stato sostituito da infiltrazioni di tessuto adiposo. Per calcolare questi parametri, è prima necessario circoscrivere la regione di interesse individuata dal muscolo, ovvero segmentarlo. Piuttosto che utilizzare le segmentazioni manuali, si potrebbero utilizzare le segmentazioni automatiche e poi calcolare l'errore della stima rispetto all'utilizzo dei ground truth, per testarne le potenzialità.



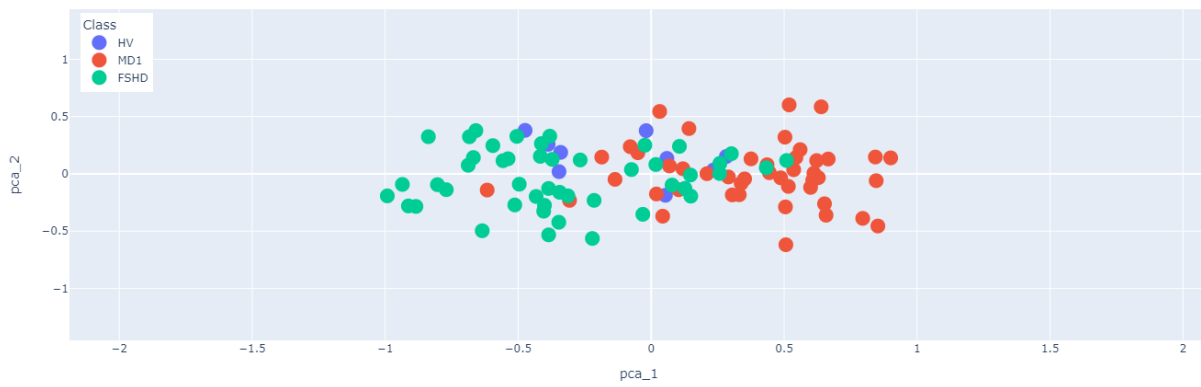
## Appendice A.

### Risultati della Texture Analysis su tutti i muscoli

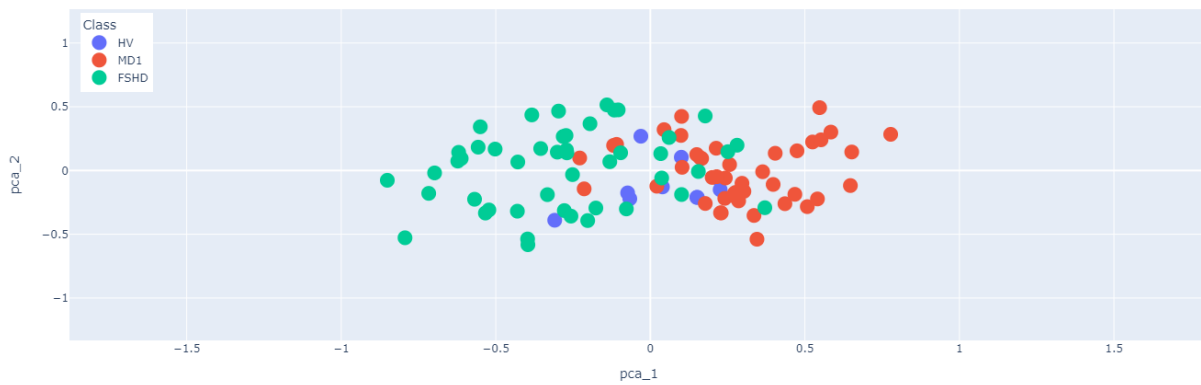
#### Coscia t-test



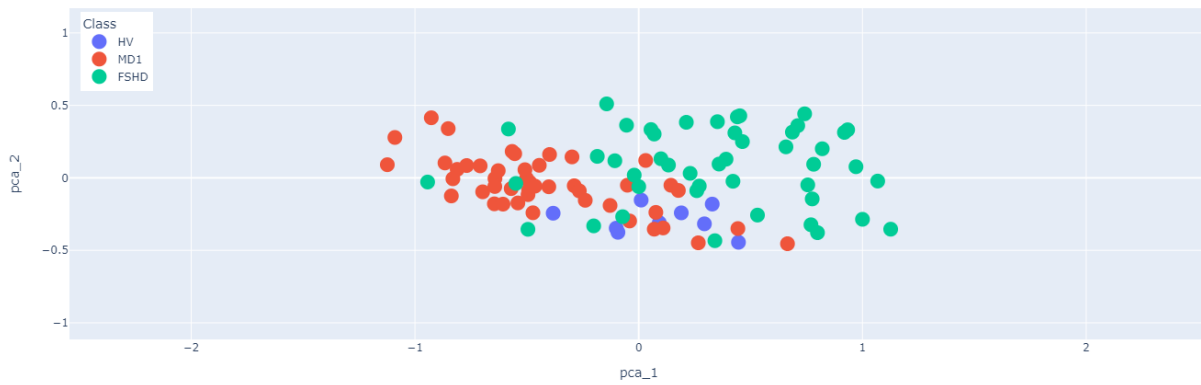
Bicipite femorale - capo lungo(BF-CL) t-test , N\_features = 9



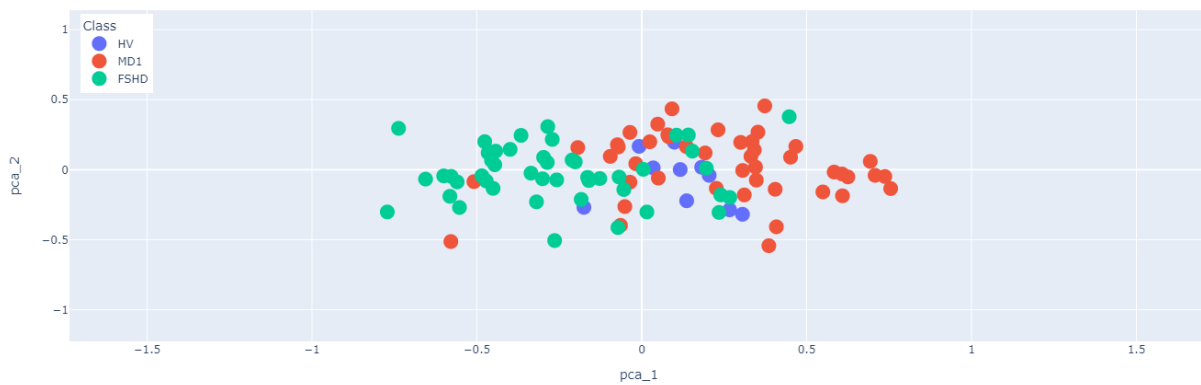
Muscolo semitendinoso(SEMI-TN) t-test , N\_features = 8



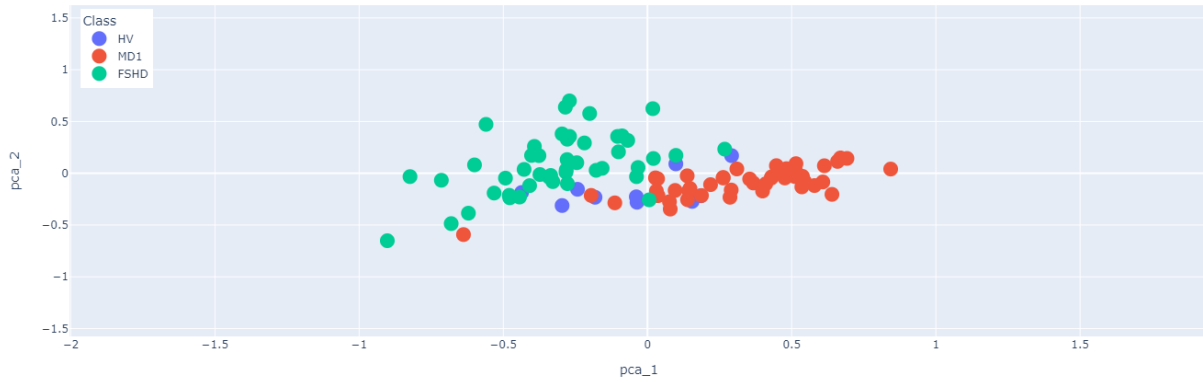
Muscolo semimembranoso(SEMI-MB) t-test , N\_features = 10



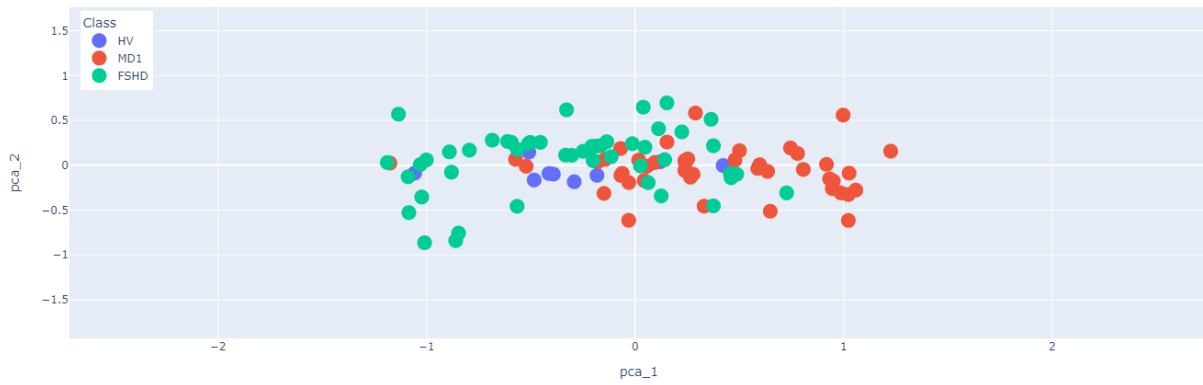
Muscolo gracile(GRA) t-test , N\_features = 6



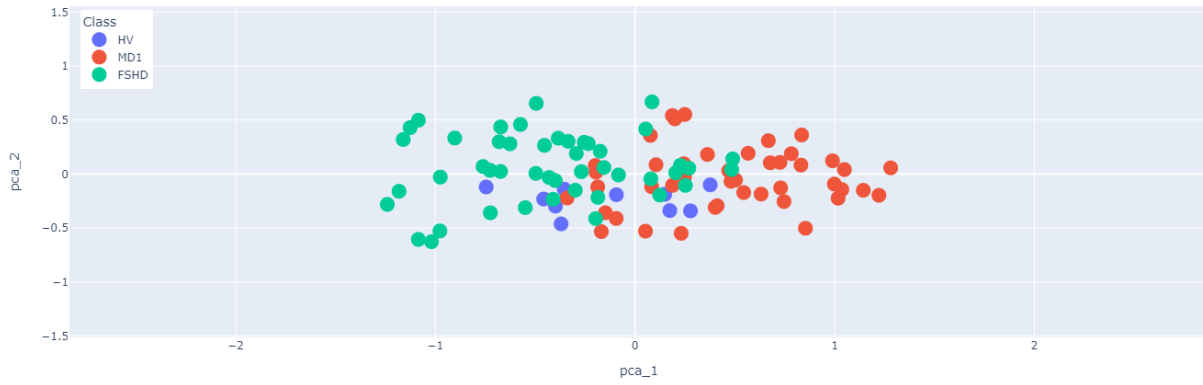
Muscolo sartorio(SAR) t-test , N\_features = 10



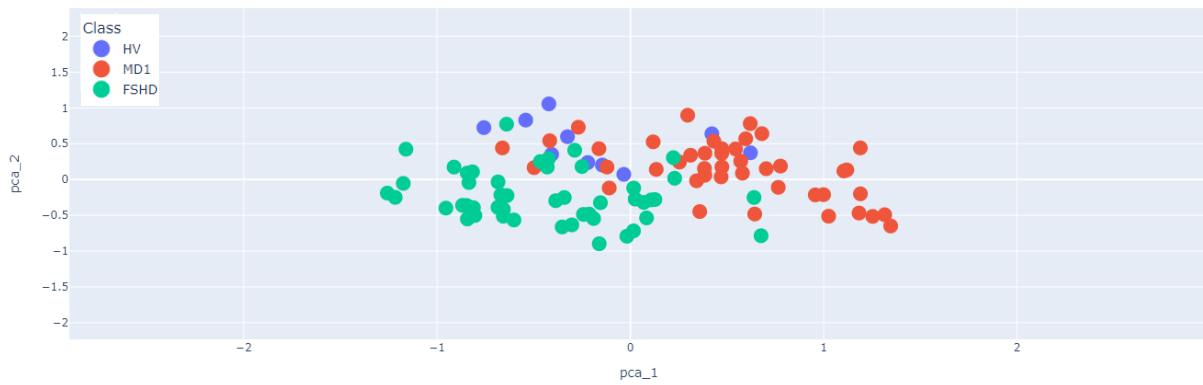
Retto femorale(RF) t-test , N\_features = 16



Vasto Mediale(VM) t-test , N\_features = 14

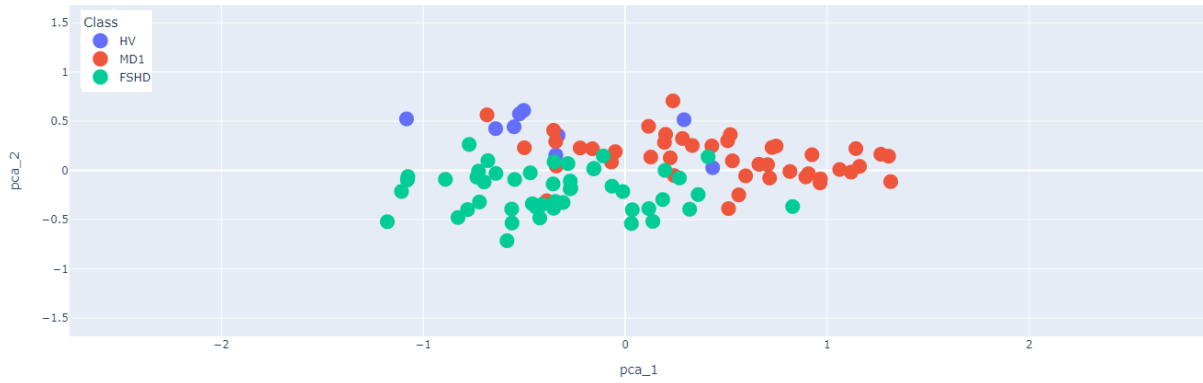


Vasto Intermedio(VI) t-test , N\_features = 20



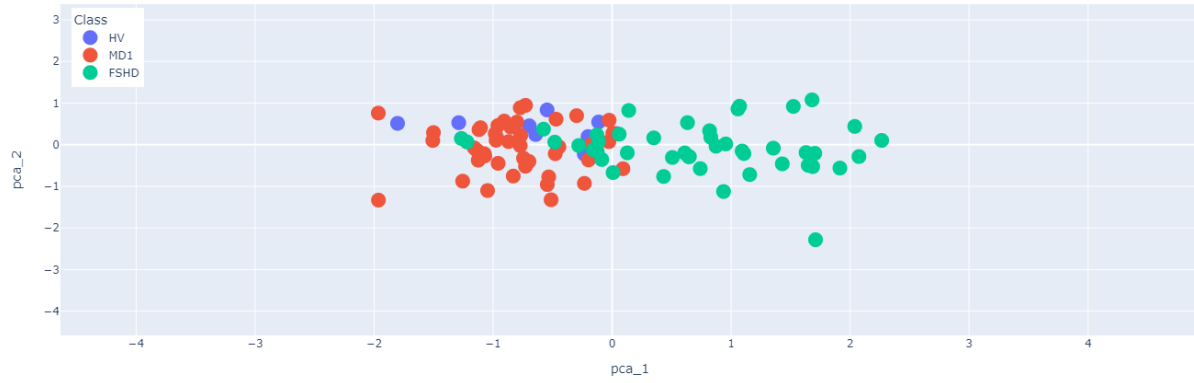


Vasto Laterale(VL) t-test , N\_features = 16

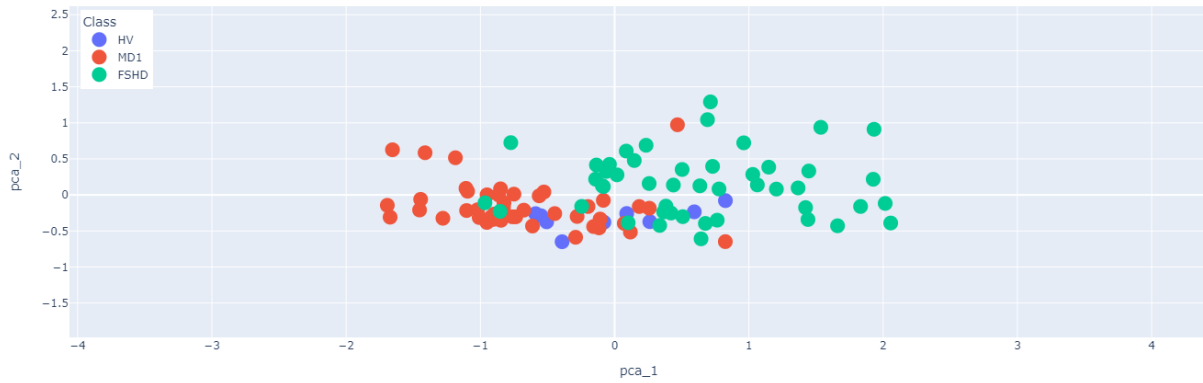


## Coscia test U di Mann-Whitney

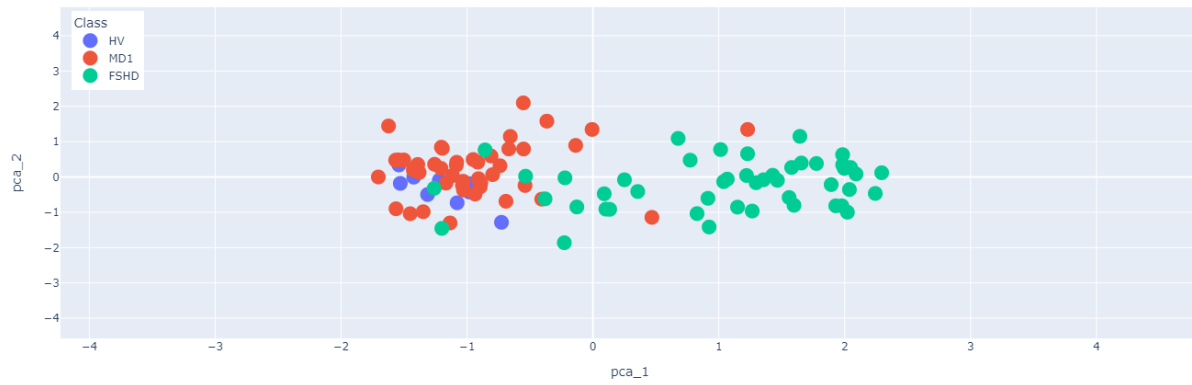
Adduttore Lungo(AL) Mann-Whitney U test , N\_features = 69



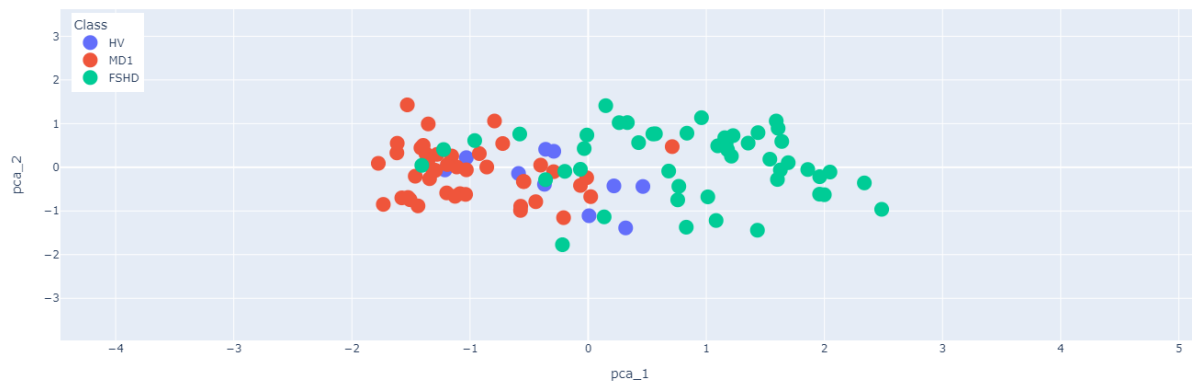
Grande Adduttore + adduttore breve(ADD) Mann-Whitney U test , N\_features = 54



Bicipite femorale - capo breve(BF-CB) Mann-Whitney U test , N\_features = 72



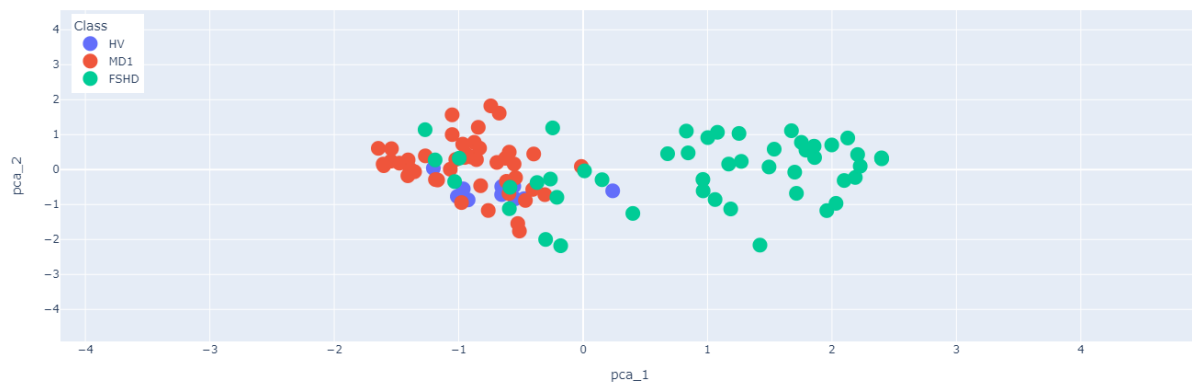
Bicipite femorale - capo lungo(BF-CL) Mann-Whitney U test , N\_features = 70



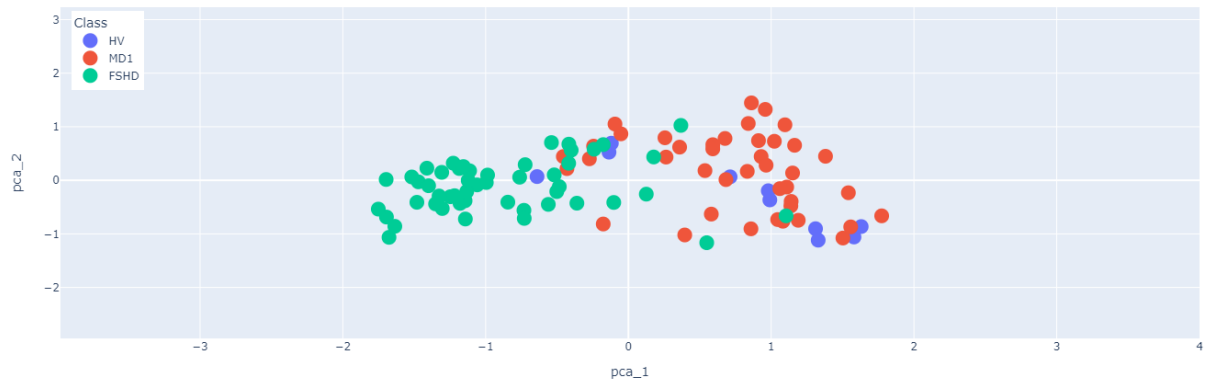
Muscolo semitendinoso(SEMI-TN) Mann-Whitney U test , N\_features = 69



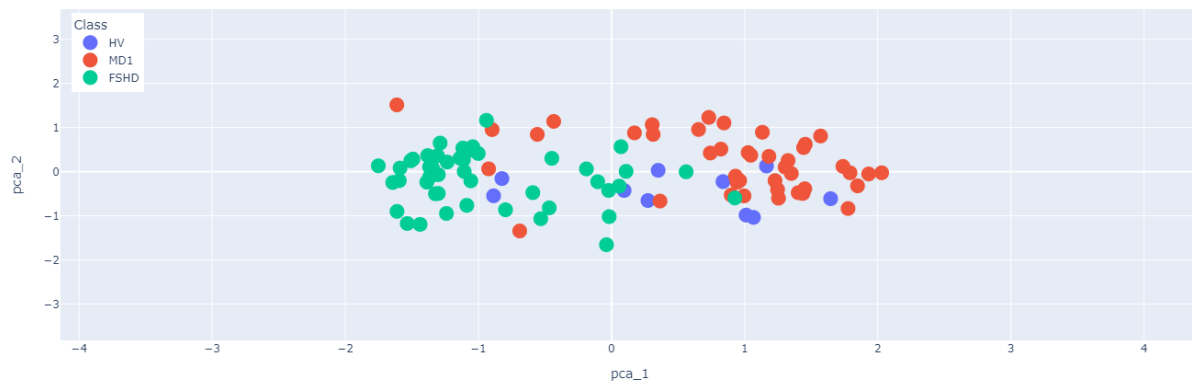
Muscolo semimembranoso(SEMI-MB) Mann-Whitney U test , N\_features = 72



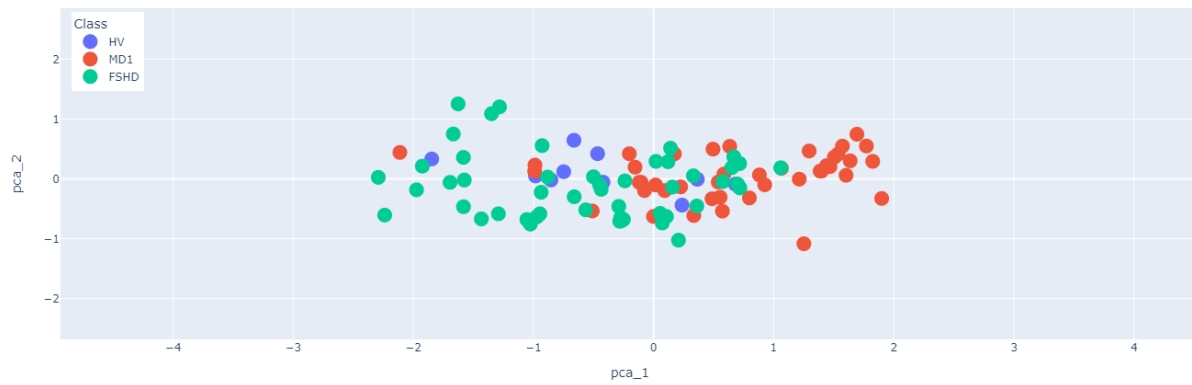
Muscolo gracile(GRA) Mann-Whitney U test , N\_features = 74



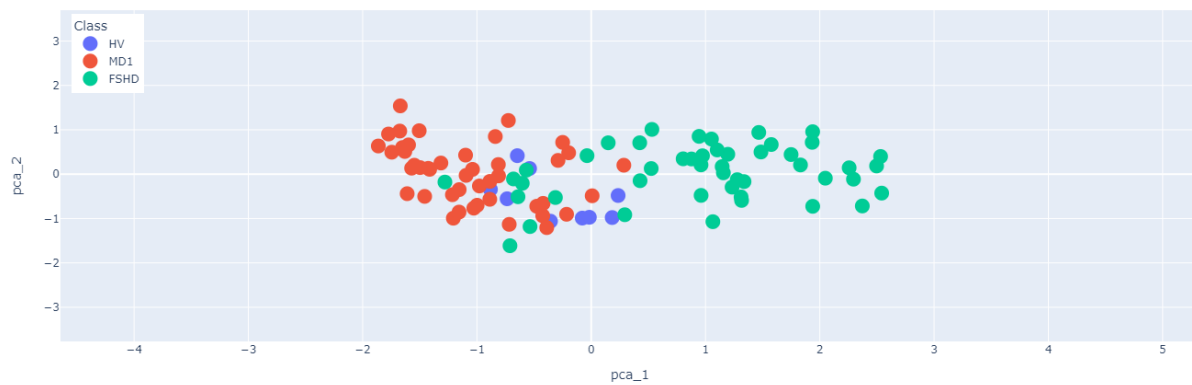
Muscolo sartorio(SAR) Mann-Whitney U test , N\_features = 71



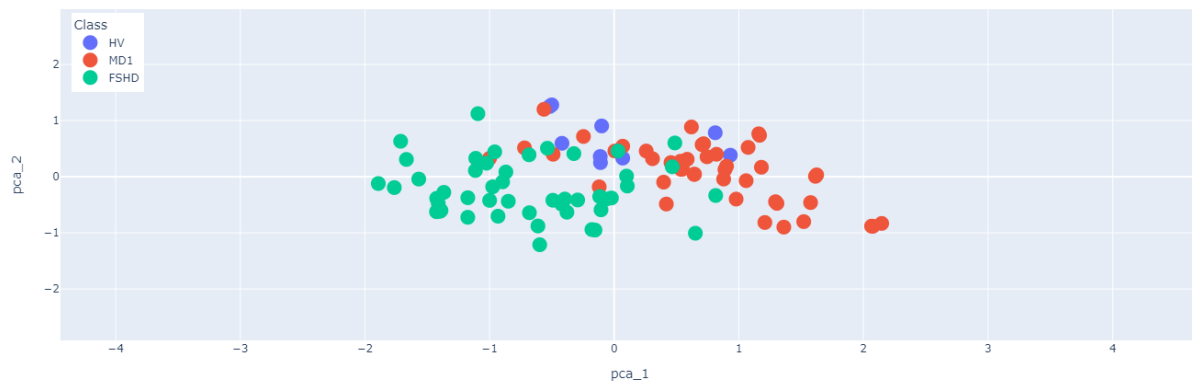
Retto femorale(RF) Mann-Whitney U test , N\_features = 58



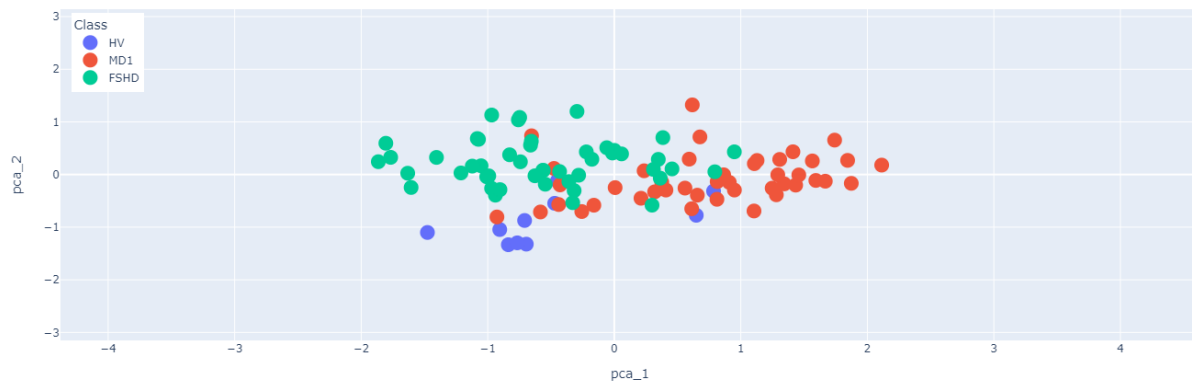
Vasto Mediale(VM) Mann-Whitney U test , N\_features = 70



Vasto Intermedio(VI) Mann-Whitney U test , N\_features = 60

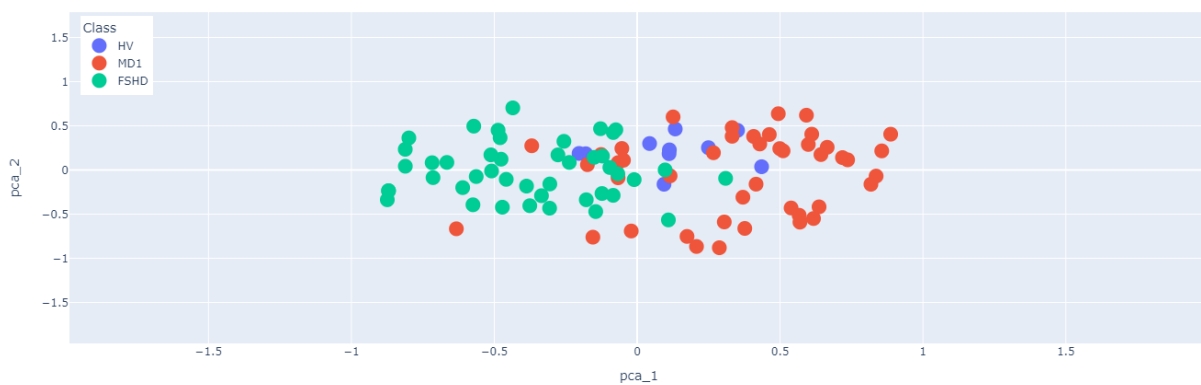


Vasto Laterale(VL) Mann-Whitney U test , N\_features = 55

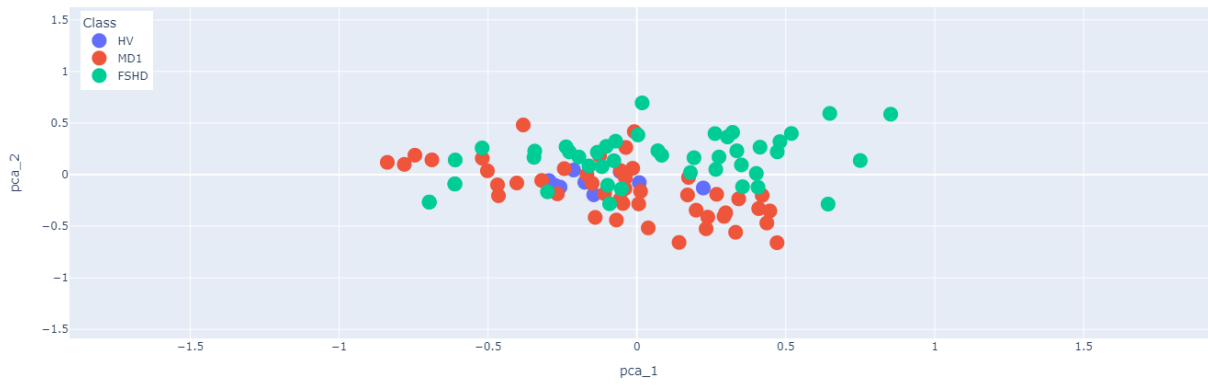


## Gamba t-test

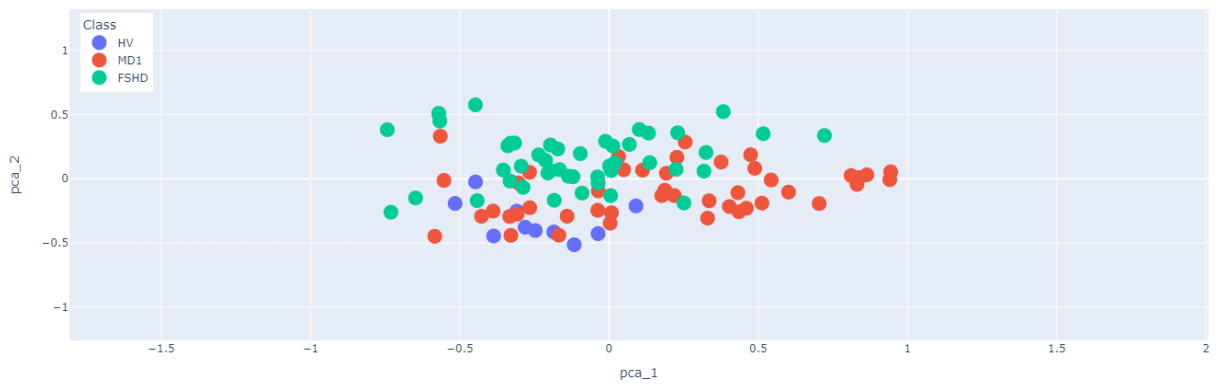
Flessore Lungo Dell'alluce(FL-a) t-test , N\_features = 13



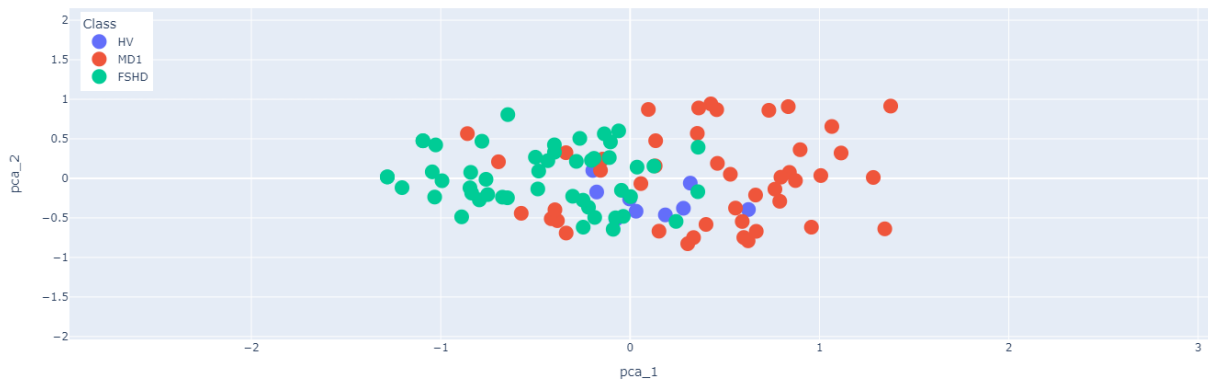
Gastrocnemio Laterale(GL) t-test , N\_features = 6



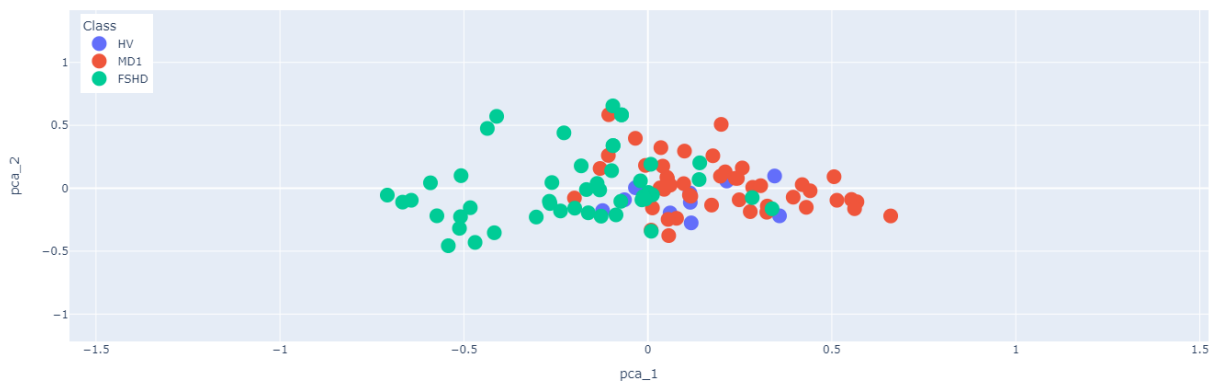
Gastrocnemio Mediale(GM) t-test , N\_features = 9



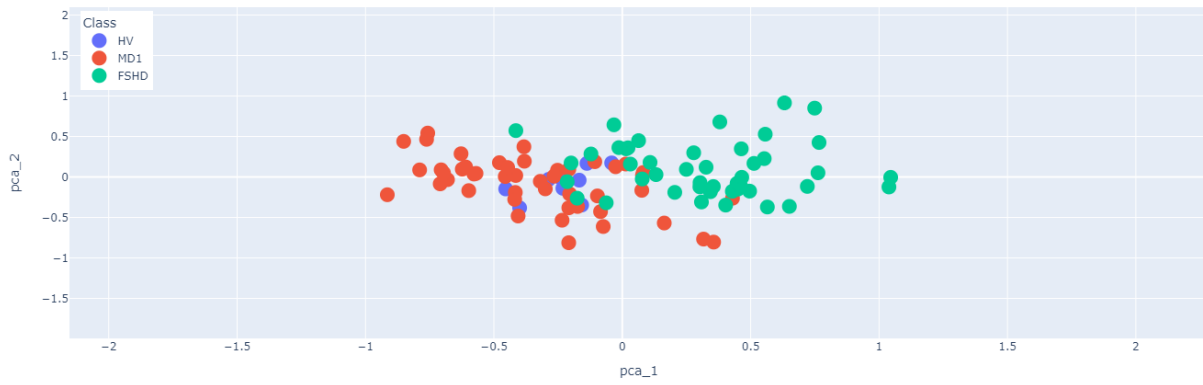
Soleo(SOL) t-test , N\_features = 21



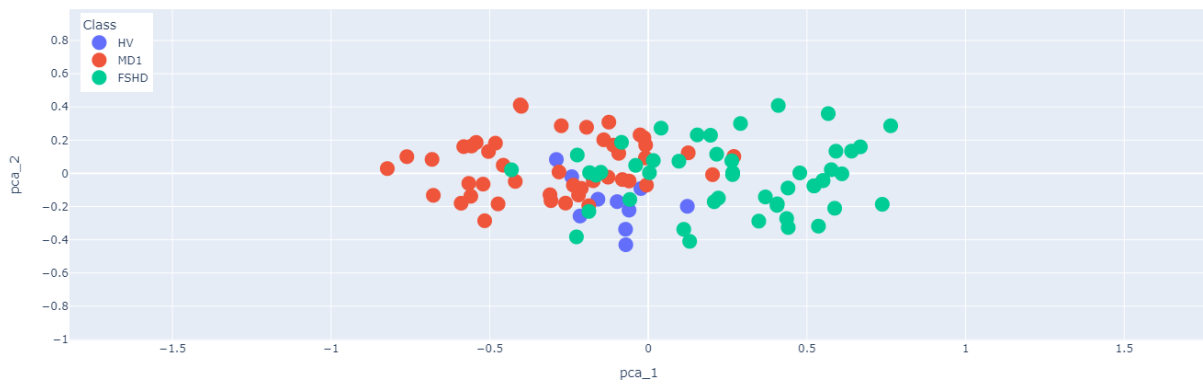
Flessore Lungo delle dita(FL-d) t-test , N\_features = 5



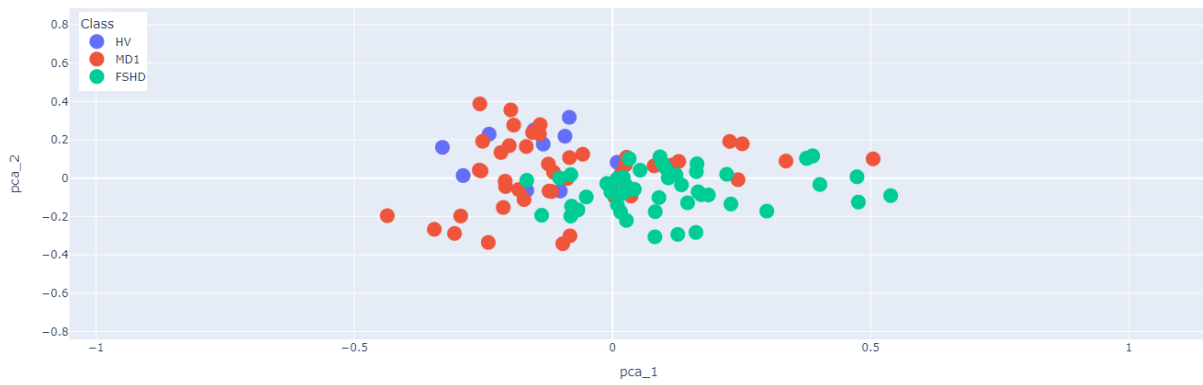
Peroneo(PER) t-test , N\_features = 14



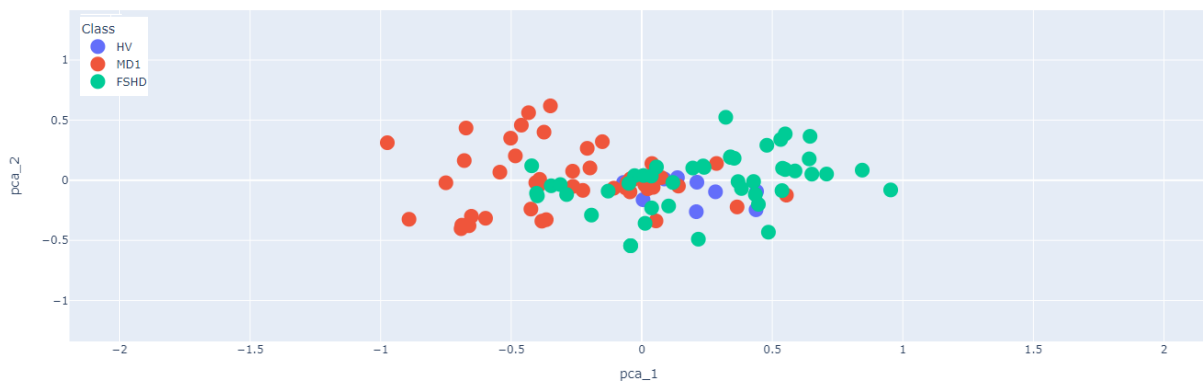
Estensore Lungo(EL) t-test , N\_features = 6



Tibiale Posteriore(TP) t-test , N\_features = 2

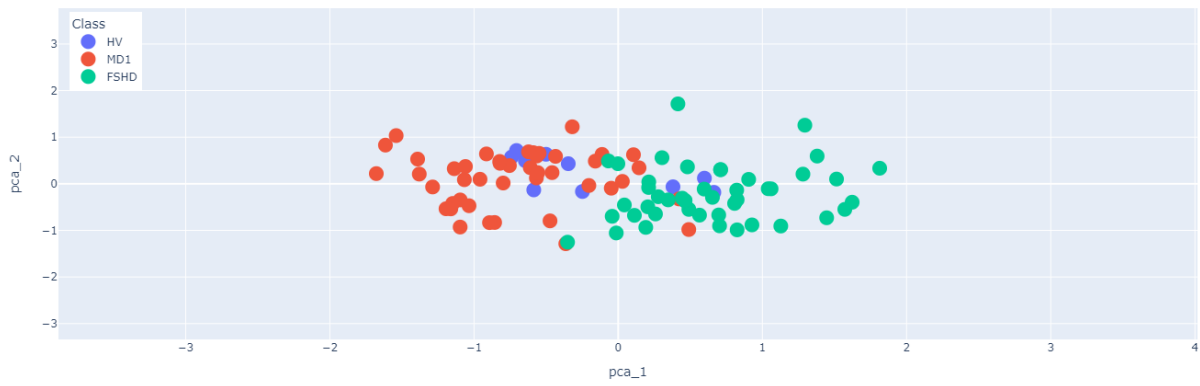


Tibiale Anteriore(TA) t-test , N\_features = 10

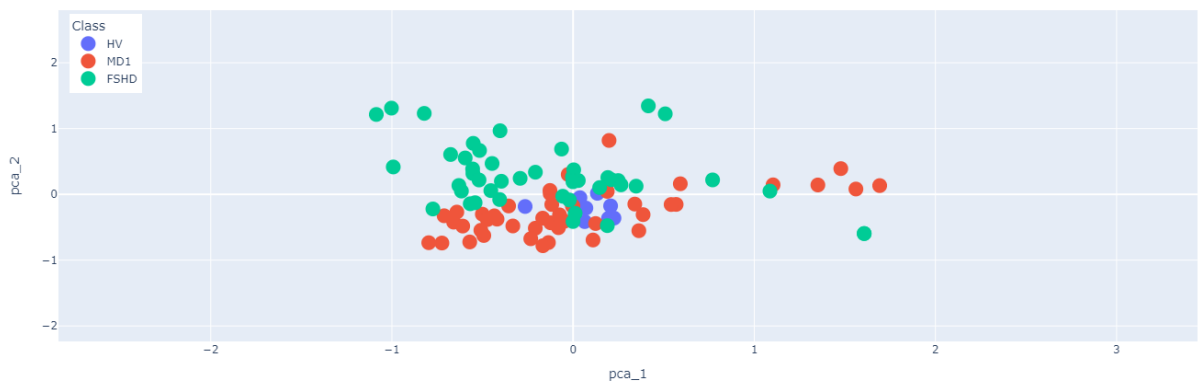


# Gamba test U di Mann-Whitney

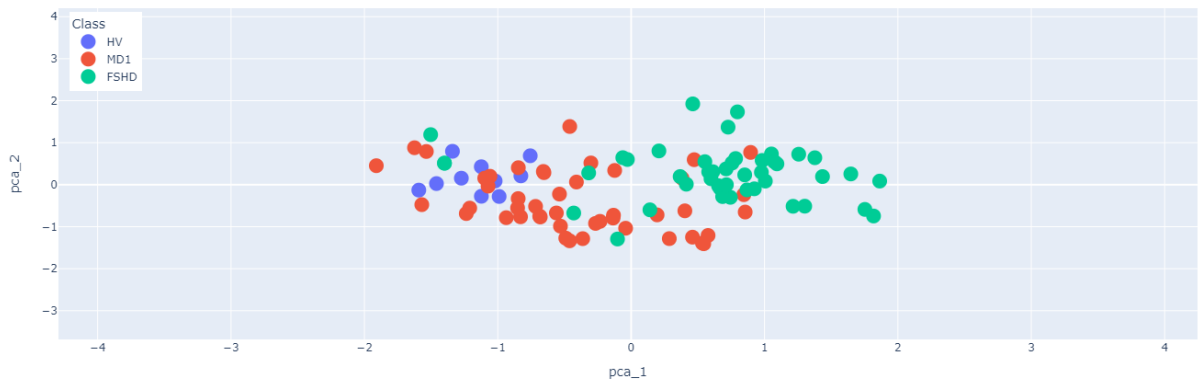
Flessore Lungo Dell'alluce(FL-a) Mann-Whitney U test , N\_features = 68



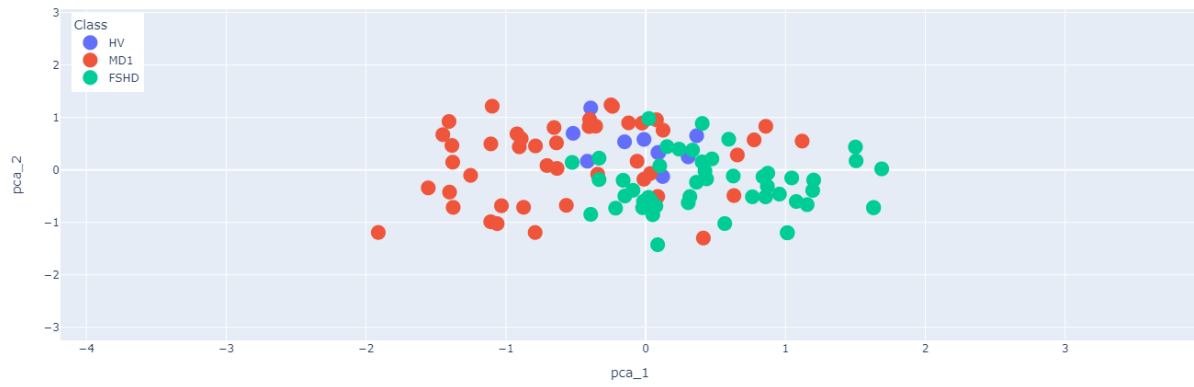
Gastrocnemio Laterale(GL) Mann-Whitney U test , N\_features = 33



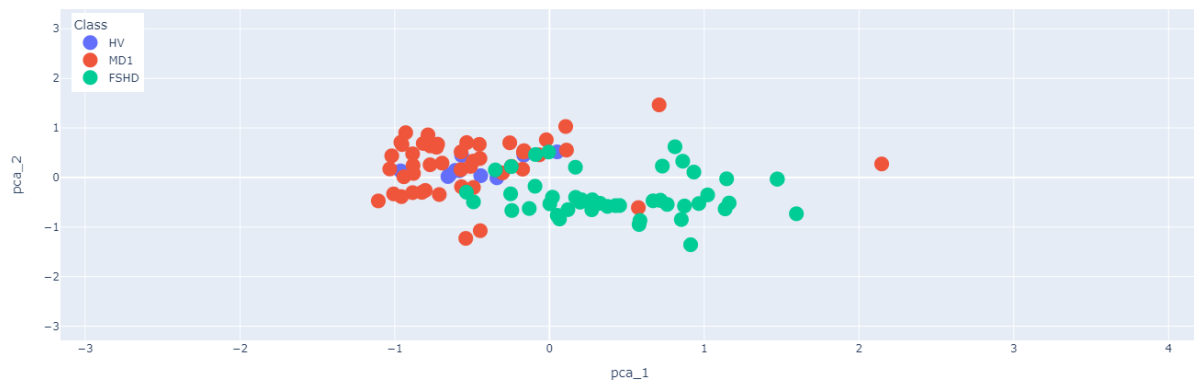
Gastrocnemio Mediale(GM) Mann-Whitney U test , N\_features = 62



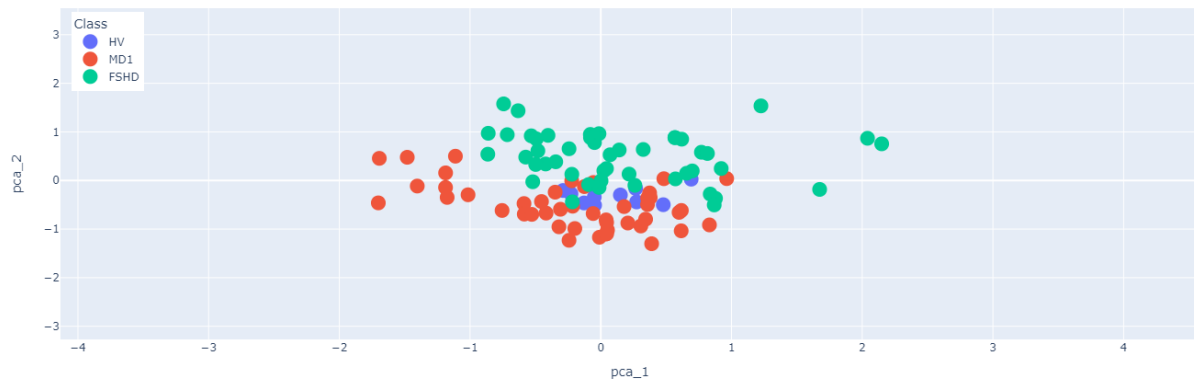
Soleo(SOL) Mann-Whitney U test , N\_features = 60



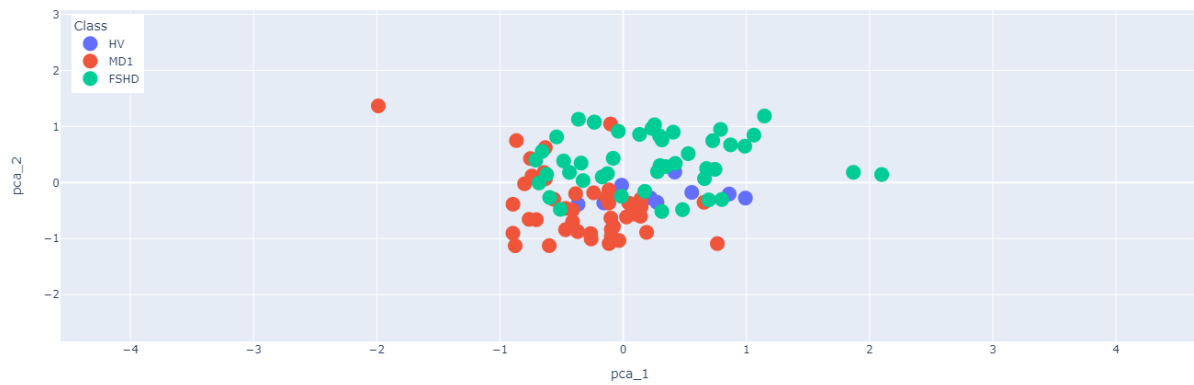
Flessore Lungo delle dita(FL-d) Mann-Whitney U test , N\_features = 58



Peroneo(PER) Mann-Whitney U test , N\_features = 59

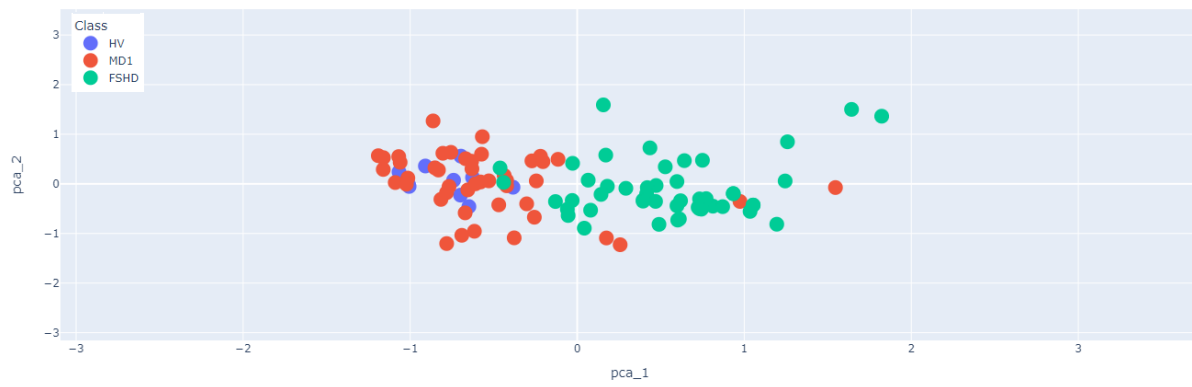


Estensore Lungo(EL) Mann-Whitney U test , N\_features = 49

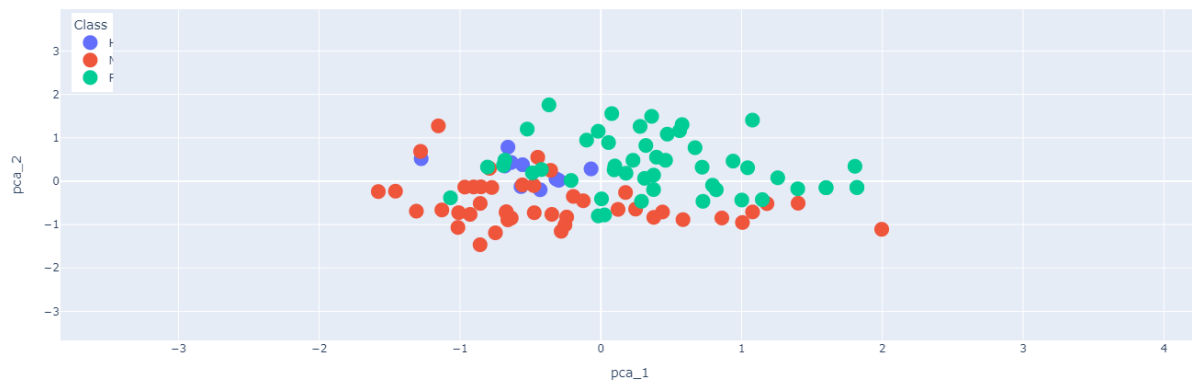




Tibiale Posteriore(TP) Mann-Whitney U test , N\_features = 60



Tibiale Anteriore(TA) Mann-Whitney U test , N\_features = 66





## BIBLIOGRAFIA

- [1] G. Meola and R. Cardani, "Myotonic dystrophies: An update on clinical aspects, genetic, pathology, and molecular pathomechanisms," *Biochimica et Biophysica Acta - Molecular Basis of Disease*, vol. 1852, no. 4. Elsevier B.V., pp. 594–606, Apr. 01, 2015.
- [2] D. G. Leung, J. A. Carrino, K. R. Wagner, and M. A. Jacobs, "Whole-body magnetic resonance imaging evaluation of facioscapulohumeral muscular dystrophy," *Muscle Nerve*, vol. 52, no. 4, pp. 512–520, Oct. 2015.
- [3] Napoli Floriana Maria. (2021). *Il Registro Nazionale Italiano per la Distrofia Muscolare Facioscapolo-omerale: uno strumento per la diagnosi e la prognosi avanzate nelle malattie neuromuscolari*. Uòtima visione: 27 Dicembre 2022. <https://iris.unimore.it/handle/11380/1256058>
- [4] M. Garibaldi *et al.*, "Muscle magnetic resonance imaging in myotonic dystrophy type 1 (DM1): Refining muscle involvement and implications for clinical trials," *Eur J Neurol*, vol. 29, no. 3, pp. 843–854, Mar. 2022.
- [5] G. Gutiérrez Gutiérrez *et al.*, "Clinical guide for the diagnosis and follow-up of myotonic dystrophy type 1, MD1 or Steinert's disease," *Med Clin (Barc)*, vol. 153, no. 2, pp. 82.e1-82.e17, Jul. 2019.
- [6] L. U. Aivazoglou *et al.*, "MR imaging of inherited myopathies: a review and proposal of imaging algorithms," *European Radiology*, vol. 31, no. 11. Springer Science and Business Media Deutschland GmbH, pp. 8498–8512, Nov. 01, 2021.
- [7] M. M. O. Tonini, M. R. Passos-Bueno, A. Cerqueira, S. R. Matioli, R. Pavanello, and M. Zatz, "Asymptomatic carriers and gender differences in facioscapulohumeral muscular dystrophy (FSHD)," *Neuromuscular Disorders*, vol. 14, no. 1, pp. 33–38, 2004.
- [8] G. Andersen, J. R. Dahlqvist, C. R. Vissing, K. Heje, C. Thomsen, and J. Vissing, "MRI as outcome measure in facioscapulohumeral muscular dystrophy: 1-year follow-up of 45 patients," *J Neurol*, vol. 264, no. 3, pp. 438–447, Mar. 2017.
- [9] Hamel J and Tawil R., "Facioscapulohumeral Muscular Dystrophy: Update on Pathogenesis and Future Treatments," *Neurotherapeutics*, vol. 15, no. 4, pp. 863–871, Oct. 2018.
- [10] Canese R. and Podo F."Introduzione alla risonanza magnetica ad uso clinic. Principi fisici e strumentazione", *Annali dell'Istituto Superiore di Sanità*, vol. 30, n.1 (1994), pp.7-29.
- [11] Kocak M. (2021). *Imaging a risonanza magnetica*. Iltima visione: 15 Gennaio 2023. <https://www.msmanuals.com/it-it/professionale/argomenti-speciali/principi-di-imaging-radiologico/imaging-a-risonanza-magnetica>
- [12] A. Pitecchio, C.Cini, M.G. Egitto , A. Berardinelli, E.Mercuri, C.Ugetti. "La Risonanza Magnetica muscolare nelle malattie neuromuscolari." *Rivista di Neuroradiologia*, vol 16, pp. 503-510, 2003
- [13] *Proton Density (PD)*. Ultima visione: 19 Gennaio 2023. <https://mrimaster.com/characterise%20image%20pd.html>
- [14] Yahya Baba. (2022). *Sequenze MRI (panoramica)*. Ultima visione 19 Gennaio 2023. <https://radiopaedia.org/articles/mri-sequences-overview>
- [15] A. van den Heuvel *et al.*, "Facioscapulohumeral dystrophy transcriptome signatures correlate with different stages of disease and are marked by different MRI biomarkers," *Sci Rep*, vol. 12, no. 1, Dec. 2022.
- [16] C. Curatolo. *Limiti fisico-chimici delle tecniche di saturazioni convenzionali: l'utilità del metodo Dixon per l'imaging in risonanza magnetica* . Ultima visione: 02 Marzo 2023. <https://jahc.eu/limiti-fisico-chimici-delle-tecniche-di-saturazioni-convenzionali-lutilita-del-metodo-dixon-per-limaging-in-risonanza-magnetica/>
- [17] A. Grimm *et al.*, "Evaluation of 2-point, 3-point, and 6-point Dixon magnetic resonance imaging with flexible echo timing for muscle fat quantification," *Eur J Radiol*, vol. 103, pp. 57–64, Jun. 2018.
- [18] C. F. Lins, C. E. G. Salmon, and M. H. Nogueira-Barbosa, "Applications of the dixon technique in the evaluation of the musculoskeletal system," *Radiol Bras*, vol. 54, no. 1, pp. 33–42, 2021.
- [19] E. Mercuri, A. Pichiechio, J. Allsop, S. Messina, M. Pane, and F. Muntoni, "Muscle MRI in inherited

- neuromuscular disorders: Past, present, and future," *Journal of Magnetic Resonance Imaging*, vol. 25, no. 2. pp. 433–440, Feb. 2007.
- [20] H. Reyngoudt *et al.*, "Global versus individual muscle segmentation to assess quantitative MRI-based fat fraction changes in neuromuscular diseases," *Eur Radiol*, vol. 31, no. 6, pp. 4264–4276, Jun. 2021.
- [21] Sogeti Labs. (2019). *Machine Learning: An Introduction*. Ultima visione: 15 Febbraio 2023. <https://labs.sogeti.com/machine-learning-an-introduction/>
- [22] K. Verma and P. Kumar Singh, "An Insight to Soft Computing based Defect Prediction Techniques in Software," *International Journal of Modern Education and Computer Science*, vol. 7, no. 9, pp. 52–58, Sep. 2015.
- [23] P. Singh.(2022). *Introduction to Neural Networks in Deep Learning* .Ultima visione: 15 Febbraio 2023. <https://www.naukri.com/learning/articles/neural-networks-in-deep-learning/>
- [24] I. Castiglioni *et al.*, "AI applications to medical images: From machine learning to deep learning," *Physica Medica*, vol. 83. Associazione Italiana di Fisica Medica, pp. 9–24, Mar. 01, 2021.
- [25] Zhang, Aston and Lipton, Zachary C. and Li, Mu and Smola, Alexander J. (2021). *Dive into Deep Learning*. Ultima visione: 17 Febbraio 2023. [https://d2l.ai/chapter\\_computer-vision/fine-tuning.html](https://d2l.ai/chapter_computer-vision/fine-tuning.html)
- [26] A. G. Schwing and R. Urtasun, "Fully Connected Deep Structured Networks." *arXiv preprint arXiv:1503.02351* (2015).
- [27] G. Kumichev. (2022). *The Inductive Bias of ML Models, and Why You Should Care About It*. Ultima visione: 20 Febbraio 2023. <https://towardsdatascience.com/the-inductive-bias-of-ml-models-and-why-you-should-care-about-it-979fe02a1a56>
- [28] S. J. Lee, G. H. Kwak, and T. H. Kim, "TORRES: A Resource-Efficient Inference Processor for Binary Convolutional Neural Networks Based on Locality-Aware Operation Skipping," *Electronics (Switzerland)*, vol. 11, no. 21, Nov. 2022.
- [29] V. Dumoulin, and F. Visin. "A guide to convolution arithmetic for deep learning." *arXiv preprint arXiv:1603.07285* (2016).
- [30] A. F. Agarap. "Deep learning using rectified linear units (relu)." *arXiv preprint arXiv:1803.08375* (2018).
- [31] F. Hussain. (2020). *Your Handbook to Convolutional Neural Networks* . Ultima visione: 08 Febbraio 2023. <https://medium.com/analytics-vidhya/your-handbook-to-convolutional-neural-networks-628782b68f7e>
- [32] S. Das. (2017). *CNN Architectures: LeNet, AlexNet, VGG, GoogLeNet, ResNet and more...* . Ultima visita: 08 Febbraio 2023. <https://medium.com/analytics-vidhya/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5>
- [33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
- [34] M. Talo, "Convolutional Neural Networks for Multi-class Histopathology Image Classification." *arXiv preprint arXiv 1903.10035*.
- [35] A. Dosovitskiy *et al.*, "AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE." *arXiv preprint arXiv 2010.11929*. (2021)
- [36] A. Vaswani *et al.*, "Attention Is All You Need." *arXiv preprint arXiv 1706.03762*. (2017)
- [37] Zhang, Aston and Lipton, Zachary C. and Li, Mu and Smola, Alexander J. (2021). *Dive into Deep Learning*. Ultima visione: 13 Febbraio 2023. [https://d2l.djl.ai/chapter\\_attention-mechanisms/self-attention-and-positional-encoding.html](https://d2l.djl.ai/chapter_attention-mechanisms/self-attention-and-positional-encoding.html)
- [38] R. Karim. (2019). *Illustred: Self-Attention. A step-by-step guide to self-attention with illustrations and code*. Ultima visione 20 Febbraio 2023. <https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a>
- [39] Y. Liu *et al.*, "A Survey of Visual Transformers." *arXiv preprint arXiv 2111.06091*. (2021)
- [40] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., ... & Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 10012-10022).
- [41] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III* 18 (pp. 234-241).

- [42] Hatamizadeh, A., Nath, V., Tang, Y., Yang, D., Roth, H. R., & Xu, D. (2022, July). Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 7th International Workshop, BrainLes 2021, Held in Conjunction with MICCAI 2021, Virtual Event, September 27, 2021, Revised Selected Papers, Part I* (pp. 272-284).
- [43] *MONAI Documentation*. Ultima visita: 21 Febbraio 2023.  
<https://docs.monai.io/en/stable/networks.html#swinunetr>
- [44] Tang, Y., Yang, D., Li, W., Roth, H. R., Landman, B., Xu, D., ... & Hatamizadeh, A. (2022). Self-supervised pre-training of swin transformers for 3d medical image analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* pp. 20730-20740.
- [45] Zhang, W., Pang, J., Chen, K., & Loy, C. C. (2021). "K-net: Towards unified image segmentation". *Advances in Neural Information Processing Systems*, 34, pp. 10326-10338.
- [46] M. Kurzyński., M. Woźniak, A. żołnerek, and E. Puchała, "Bias Field Correction for MRI Images.," *Computer Recognition Systems*, vol. 30, 2005.
- [47] N. J. Tustison *et al.*, "N4ITK: Improved N3 bias correction," *IEEE Trans Med Imaging*, vol. 29, no. 6, pp. 1310–1320, Jun. 2010.
- [48] X. Sun *et al.*, "Histogram-based normalization technique on human brain magnetic resonance images from different acquisitions," *Biomed Eng Online*, vol. 14, no. 1, Jul. 2015.
- [49] A. Carré *et al.*, "Standardization of brain MR images across machines and protocols: bridging the gap for MRI-based radiomics," *Sci Rep*, vol. 10, no. 1, Dec. 2020.
- [50] K. v. Hoebel *et al.*, "Radiomics repeatability pitfalls in a scan-rescan mri study of glioblastoma," *Radiol Artif Intell*, vol. 3, no. 1, Jan. 2021.
- [51] F. I. Tushar, B. Alyafi, M. K. Hasan and L. Dahal, "Brain Tissue Segmentation Using NeuroNet With Different Pre-processing Techniques," *2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, Spokane, WA, USA, 2019, pp. 223-227.
- [52] Nyul LG, Udupa JK, and Zhang X, "New variants of a method of MRI scale standardization ,," *IEEE Transactions on Medical Imaging* , vol. 19, no. 2, pp. 143–150, 2000.
- [53] *ITK. Isight Toolkit*. Ultima visita: 18 Febbraio 2023.  
[https://itk.org/Doxygen/html/classitk\\_1\\_1HistogramMatchingImageFilter.html](https://itk.org/Doxygen/html/classitk_1_1HistogramMatchingImageFilter.html)
- [54] R. Melis. (2022). Preprocessing in chunks with GPU (P100 vs T4 x2) Ultima visita: 21 Febbraio 2023.  
<https://www.kaggle.com/code/raimondomelis/preprocessing-in-chunks-with-gpu-p100-vs-t4-x2>
- [55] Hatamizadeh, A., Tang, Y., Nath, V., Yang, D., Myronenko, A., Landman, B., & Xu, D. (2022). Unetr: Transformers for 3d medical image segmentation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision* (pp. 574-584).
- [56] Siddiquee, M. M. R., Yang, D., He, Y., Xu, D., & Myronenko, A. (2022). Automated segmentation of intracranial hemorrhages from 3D CT. *arXiv preprint arXiv:2209.10648*.
- [57] P. Carnahan, J. Moore, D. Bainbridge, M. Eskandari, E. C. S. Chen, and T. M. Peters, "DeepMitral: Fully Automatic 3D Echocardiography Segmentation for Patient Specific Mitral Valve Modelling," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2021, vol. 12905 LNCS, pp. 459–468
- [58] P. Iakubovskii. (2019). "Segmentation Models Pytorch". *GitHub repository*. Ultima visita: 7 Gennaio 2023. [https://github.com/qubvel/segmentation\\_models.pytorch](https://github.com/qubvel/segmentation_models.pytorch)
- [59] *MMSegmentation : OpenMMLab Semantic Segmentation Toolbox and Benchmark*, MMSegmentation Contributors,  
[https://github.com/open-mmlab/mmssegmentation/blob/master/configs/knet/knet\\_s3\\_upernet\\_r50-d8\\_8x2\\_512x512\\_adamw\\_80k\\_ade20k.py](https://github.com/open-mmlab/mmssegmentation/blob/master/configs/knet/knet_s3_upernet_r50-d8_8x2_512x512_adamw_80k_ade20k.py) , (2020).
- [60] *MMSegmentation : OpenMMLab Semantic Segmentation Toolbox and Benchmark*, MMSegmentation Contributors, [https://github.com/open-mmlab/mmssegmentation/blob/master/configs/knet/knet\\_s3\\_upernet\\_swin-t\\_8x2\\_512x512\\_adamw\\_80k\\_ade20k.py](https://github.com/open-mmlab/mmssegmentation/blob/master/configs/knet/knet_s3_upernet_swin-t_8x2_512x512_adamw_80k_ade20k.py) . (2020)

- [61] GitHub.(2020) *TUTORIAL 1: LEARN ABOUT CONFIGS*. Ultima visione: 28 Febbraio 2023.  
<https://mmsegmentation.readthedocs.io/en/latest/tutorials/config.html>
- [62] Xiao, T., Liu, Y., Zhou, B., Jiang, Y., & Sun, J. (2018). Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 418-434).
- [63] van Griethuysen, J. J. M., Fedorov, A., Parmar, C., Hosny, A., Aucoin, N., Narayan, V., Beets-Tan, R. G. H., Fillion-Robin, J. C., Pieper, S., Aerts, H. J. W. L. (2017). Computational Radiomics System to Decode the Radiographic Phenotype. *Cancer Research*, 77(21), pp. 104–107.
- [64] *A Step-By-Step Complete Guide to Principal Component Analysis*. Ultima visita: 19 Febbraio 2023.  
<https://www.turing.com/kb/guide-to-principal-component-analysis>
- [65] P. Pozzolo. (2021). *Il Wilcoxon-Mann-Whitney test spiegato semplice*. Ultima visita: 16 Febbraio 2023.  
<https://paolapozzolo.it/wilcoxon-mann-whitney-test/>