Alma Mater Studiorum · Università di Bologna

SCUOLA DI SCIENZE
Corso di Laurea in Informatica per il Management

Metodi Variazionali e Reti Neurali per la Ricostruzione di Immagini 3D da Tomosintesi: Un Approccio Ibrido

Relatore:
Chiar.mo Prof.
Elena Loli Piccolomini

Presentata da:
Enjun Hu

Correlatore:
Chiar.mo Prof.
Davide Evangelista

IV Sessione
Anno Accademico 2023/2024

Every story has a value.
It is a story that is interesting
to some people and salvation for others.

Omniscent Reader Viewpoint

Abstract

In questa tesi viene analizzato un possibile utilizzo di una rete convoluzionale nel campo della diagnostica per immagini, con l'obiettivo di ricostruire volumi tridimensionali di tomosintesi, in particolare di una Tomosintesi Mammaria Digitale (DBT), in cui la rete viene addestrata per ricostruire immagini 3D di alta qualità partendo da input incompleti o degradati.

Nell'approccio ibrido proposto, si vuole inserire l'utilizzo di una rete convoluzionale a metà del processo di ricostruzione di una DBT iniziata dal metodo iterativo, in quanto consente di velocizzare il processo di generazione dei volumi, garantendo una diagnostica più rapida rispetto alla ricostruzione iterativa tradizionale, che richiede numerose iterazioni. Questo avviene senza compromettere al contempo la qualità e la risoluzione dei dettagli contenuti nel volume generato, poiché la rete è in grado di ricostruire immagini 3D dettagliate e prive di artefatti, anche partendo da volumi a bassa risoluzione o molto rumorosi.

Si propone quindi l'addestramento supervisionato di un modello basato su un'architettura di una UNet 3D, che riceve in input volumi generati tramite poche iterazioni di un metodo iterativo. L'obiettivo è ottenere volumi che siano il più visibilmente simili a quelli generati dopo numerose iterazioni del metodo iterativo, preservando al contempo la qualità dei dettagli anatomici.

Indice

Elenco delle figure

Introduzione

L'intelligenza artificiale negli ultimi anni è diventata sempre più presente nel campo della medicina, in particolar modo nel campo della diagnostica medica. È stato analizzato come le reti neurali profonde (Deep Network), in particolar modo i metodi basati sulle reti convoluzionali (Convolutional Neural Network) siano stati valutati e testati positivamente per la ricostruzione di Tomografie Computerizzate Sparse.

L'utilizzo delle reti neurali nel campo della diagnostica per immagini e della ricostruzione medica è dovuto alla necessità di cercare di ridurre l'esposizione totale del paziente alle radiazioni, permettendo al contempo di ottenere risultati con qualità e affidabilità elevata, così da riuscire a garantire una diagnosi veloce e al contempo accurata.

Per ridurre la quantità di radiazioni totali assorbite dal paziente durante una Tomografia tradizionale, c'è la possibilità di ridurre la corrente nel tubo a raggi X ottenendo così una Tomografia Computerizzata a basso dosaggio (low-dose TC), la quale comporta un aumento del rumore nei dati acquisiti, oppure esiste anche la possibilità di diminuire il numero di proiezioni che vengono acquisite, ottenendo così una Tomografia a poche viste (few-view TC).[1] [2] Alcune ricerche in merito hanno provato a testare l'efficacia delle reti convoluzionali nella ricostruzione delle immagini tomografiche a poche viste, in particolar modo nel campo delle immagini bidimensionali.

In questa tesi viene estesa l'analisi alla ricostruzione tridimensionale, considerando tutte e tre le dimensioni spaziali, poiché la tomografia computerizzata moderna acquisisce informazioni lungo tutti e tre gli assi. A tal fine, vengono applicati metodi di deep learning basati su reti convoluzionali per la ricostruzione di volumi tomografici di Tomosintesi Mammaria Digitali (DBT) pre-elaborati. Questi volumi sono generati dai sinogrammi attraverso metodi iterativi di ricostruzione basati su modelli, i quali verranno sostituiti nella parte intermedia di ricostruzione da un approccio basato su reti neurali, accelerando il processo, in quanto nonostante gli algoritmi di ricostruzione iterativi tradizionali siano ampiamente riconosciuti e utilizzati in ambito medico, per ottenere risultati di alta qualità dopo molte iterazioni richiede un elevato costo computazionale e temporale.
L'obiettivo è ottimizzare la qualità dei volumi tridimensionali ricostruiti dopo sole 2 ite-

razioni dell'algoritmo iterativo, prendendo come riferimento i volumi risultanti dopo 30 iterazioni, questi ultimi considerati di qualità nettamente superiore.

Nei risultati generati verrà data più attenzione alla qualità visiva di essi in quanto, nel processo della diagnostica per immagini, è fondamentale che le immagini ricostruite siano chiare e non contengano artefatti che potrebbero portare a una diagnosi incorretta.

Il dataset utilizzato viene fornito da IMS Giotto S.p.A. la quale si ringrazia per aver concesso l'utilizzo di tali dati allo sviluppo di questa tesi.

Nel primo capitolo vengono introdotti i concetti fondamentali della diagnostica per immagini, per poi approfondire le diverse tecniche di *medical imaging*. Si parte dall'argomento della Tomografia Computerizzata (TC), trattata nel capitolo 1.2 per arrivare al caso specifico della Tomosintesi Mammaria Digitale (DBT), analizzata nel capitolo 1.4.

Il secondo capitolo è dedicato alla descrizione del framework proposto, con un focus sulle reti convoluzionali, che costituiscono la base del modello utilizzato in questa tesi. Nel capitolo 2.1 viene affrontato il problema caratterizzante della Tomografia Sparsa (SpCT) e il relativo algoritmo iterativo dello Scaled Gradient Projection (SGP), usato per risolvere tale problema.

Successivamente, nel capitolo 2.4 viene introdotto il dataset utilizzato, descrivendo più nel dettaglio i dati utilizzati. A seguire, nel capitolo 2.5,viene presentata l'analisi degli iperparametri impiegati, con un focus particolare sulle funzioni di loss scelte per l'addestramento della rete, approfondite nel capitolo 2.6.

Nel terzo capitolo vengono analizzate le varianti del modello e i risultati ottenuti da esse, tramite un confronto tra input, target e sotto-volume predetto. Viene inoltre esaminata l'influenza di diverse funzioni di loss sulle prestazioni delle varie varianti del modello.

Infine, nel capitolo conclusivo 4 vengono riassunti i principali risultati ottenuti, evidenziando possibili sviluppi futuri per migliorare la qualità dei volumi generati. In particolare, si propone di sperimentare con altre funzioni di loss e testare architetture alternative, simili a quella adottata in questa tesi al fine di ottimizzare le prestazioni del modello.

Capitolo 1

Diagnostica per Immagini: La Tomosintesi

1.1 Diagnostica per Immagini

La Diagnostica per Immagini, detta anche Medical Imaging, è una tecnica e un processo con l'obiettivo di rappresentare e visualizzare l'interno del corpo umano in modo da diagnosticare e monitorare le condizioni mediche di un paziente. Si usano tecniche di Imaging, ovvero produzione di immagini, per riuscire a ottenere delle rappresentazioni visive di organi, tessuti e strutture anatomiche, aiutando così i medici a identificare malattie, lesioni o anomalie.[3]

A partire dalla scoperta delle radiazioni X-ray nel 1895 da parte di Wilhelm Konrad Röntgen, il campo della diagnostica per immagini si è evoluto in una grande disciplina scientifica, grazie alle possibilità di vedere l'interno del corpo umano tramite metodi non invasivi.
La Diagnostica per Immagini si è espansa negli ultimi 20 anni, includendo varie tipologie di tecniche come la Mammografia in 2D e la Tomografia Computerizzata (TC) che si basano sull'utilizzo di radiazioni X-ray; mentre sono state sviluppate altre tecnologie come la Risonanza Magnetica (RM) che usa i campi magnetici e le onde radio per produrre immagini di organi e tessuti molli, oppure l'Ecografia (Ultrasuoni) che si basa sulle onde sonore, con l'obiettivo di trovare nuovi metodi non invasivi per visualizzare tessuti molli senza esporre il paziente ai rischi delle radiazioni.[4] [5]

1.2 Tomografia Computerizzata

Una grande svolta nel campo della Diagnostica per Immagini fu lo sviluppo della Tomografia Assiale Computerizzata (TAC), che fu sviluppata negli anni '70 dal fisico britannico Godfrey Hounsfield e dal radiologo Allan Cormack, i quali ricevettero il Premio Nobel per la Medicina nel 1979.

Le prime macchine erano dedicate allo studio del cranio, le quali impiegavano 4-5 minuti a effettuare una scansione completa, mentre la prima ricostruzione di una sezione bidimensionale del cranio impiegava circa 9 giorni. [6]

Inizialmente il software di elaborazione permetteva di elaborare solo il piano assiale o trasversale, perpendicolare all'asse lungo del corpo, da qua il nome di Tomografia Assiale Computerizzata (TAC). Con le attuali macchine moderne, le proiezioni non sono più su piani distinti ma a spirale, facendo scorrere il lettino su cui si trova il paziente, permettendo di ottenere ricostruzioni tridimensionali del segmento corporeo esaminato, facendo proiezioni su 360 gradi, migliorando così anche le capacità diagnostiche. [7]

Grazie anche all'evoluzione tecnologica si sono sviluppati nuovi dispositivi che circondano completamente il paziente con migliaia di unità di rilevamento, facendo ruotare solamente la sorgente di raggi X e riuscendo così a ottenere immagini di tutto il corpo in pochi secondi.

La Tomografia Computerizzata (TC) ormai fa parte delle tecniche di diagnostica per imaging consolidate, che utilizzano i raggi X e algoritmi computazionali per generare immagini tridimensionali dettagliate dei diversi tipi di tessuto analizzati, come i tessuti molli e gli organi interni del paziente.

1.3 Tomosintesi

La TC impiega l'uso delle radiazioni X, le cui esposizioni aumentano il rischio di tumori e leucemie in relazione alla dose, in particolar modo nei bambini e nei giovani. [8] Proprio per tale motivo, negli anni si sono cercati nuovi modi per sviluppare una tecnica di imaging che utilizzasse una dose inferiore di radiazioni. Nella TC classica, le proiezioni vengono acquisite a 360 gradi, con l'angolo tra una proiezione e la successiva tipicamente meno di un grado.

Un modo per ridurre le radiazioni totali sul corpo del paziente è quello di aumentare questo angolo (tra una proiezione e quella successiva), ottenendo così un insieme di proiezioni più ridotte; questa tecnica prende poi il nome di Tomografia Sparsa (SpCT).

Per ridurre ulteriormente le dosi di radiazioni, è possibile diminuire anche il campo angolare di scansione, invece di acquisire proiezioni su 360 gradi come nella TC classica,

si può decidere di acquisire solo proiezioni a 180 gradi. Questo è utile in particolar modo quando non si riesce a far girare la sorgente dei raggi X a 360 gradi rispetto al corpo del paziente. L'insieme delle due tecniche per ridurre le dosi di radiazioni totali viene denominato *Tomosintesi Digitale*, la quale fa parte della classe delle Tomografie Sparse in quanto mancano delle proiezioni rispetto alla TC classica.

Nel corso degli anni, la Tomosintesi Digitale è stata adottata in diversi settori della diagnostica per immagini. Tra le sue applicazioni vi sono la Tomosintesi Toracica, utilizzata per individuare noduli polmonari e tubercolosi, e la Tomosintesi Ortopedica di mano e polso, impiegata nella diagnosi di microfratture e artrite reumatoide. [9]

Forse una delle Tomosintesi Digitali più famose è quella Mammaria, ovvero la Digital Breast Tomosynthesis (DBT), che viene denominata anche come una Mammografia 3D.

1.4 Tomosintesi Mammaria Digitale

Secondo l'American Cencer Society, il cancro al seno è il cancro più diagnosticato e la causa più frequente di morte per cancro tra le donne, in quanto cresce lentamente o anche senza sintomi visibili. [10;11;12]

Per questo l'American College of Radiology raccomanda una mammografia annuale a tutte le donne a partire dai 40 anni, enfatizzando l'importanza di una diagnosi precoce tramite controlli di routine. Negli ultimi 10 anni, le morti dovute al cancro al seno si sono ridotte; questo miglioramento è dovuto anche in parte alle nuove tecnologie, ma in particolar modo grazie ai medici che riescono a rilevare il tumore al seno in fase precoce, prima che riesca a diffondersi ai linfonodi o ad altre parti del corpo. [13]

Una delle tecniche che hanno contribuito a questo è la Tomosintesi Mammaria Digitale (DBT), considerata un approccio innovativo alla Mammografia in 2D. Questa tecnologia consente di acquisire una serie di proiezioni lungo un arco e di ricostruire i dati tramite tecniche di post-processing, generando un volume quasi tridimensionale del seno.

Questa metodologia combina proiezioni bidimensionali da diverse angolazioni lungo un arco limitato per generare una rappresentazione tridimensionale del seno, permettendo inoltre una miglior visibilità strutturale, in particolare in pazienti che hanno del tessuto mammario denso, riducendo le sovrapposizioni di tessuti che potrebbero oscurare eventuali lesioni o micro-calcificazioni in una mammografia 2D convenzionale.

Secondo uno studio effettuato dalla RSNA del 2024, durato ben 10 anni, lo screening per il cancro al seno effettuato tramite la DBT ha aumentato i tassi di rilevamento e ridotto significativamente i tumori in fase avanzata, rispetto a usare la mammografia 2D convenzionale. [14] [15]

Capitolo 2

Framework Proposto

In questo capitolo viene analizzato il framework proposto per la ricostruzione dei volumi ottenuti dalla Tomografia Sparsa. In particolare, i volumi inizialmente generati dopo due iterazioni del metodo iterativo vengono successivamente elaborati da una rete neurale, che completa il processo di ricostruzione, restituendo volumi il più fedeli possibile a quelli ottenuti dopo 30 iterazioni del metodo iterativo.

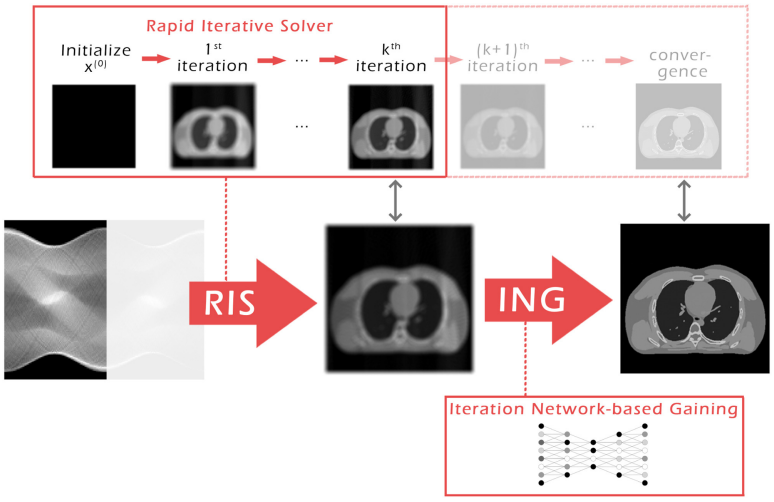

Figura 2.1: Schema di ricostruzione tramite approccio ibrido.

Descrizione: Schema Logico in cui la ricostruzione viene iniziata tramite un metodo iterativo, per poi passare l'ouput a una rete neurale il quale lo prende in input e lo porta alla convergenza.

Source: https://doi.org/10.1016/j.compmedimag.2022.102156

Questo approccio riduce significativamente il tempo e il costo computazionale necessari per generare i volumi, risultando particolarmente utile in situazioni reali dove i tempi di attesa per la convergenza dell'algoritmo iterativo sono limitati. In questo modo, si rende possibile una diagnosi più rapida, facilitando l'individuazione di eventuali problemi, come microcalcificazioni all'interno del seno.

L'obiettivo della rete proposta è quello di riuscire a completare e affinare il processo di ricostruzione iniziato dall'algoritmo iterativo descritto nel capitolo 2.1.1, migliorando il volume di input da due iterazioni affinché assomigli il più possibile al volume di target da 30 iterazioni.

Viene introdotto inoltre il concetto di reti convoluzionali nel capitolo 2.2, seguito dal capitolo 2.3 in cui viene spiegata l'architettura UNet utilizzata in questa tesi.

Infine, viene introdotto il dataset utilizzato per addestrare e testare il modello nel capitolo 2.4, per poi analizzare i vari iperparametri utilizzati con particolare focus sulla scelta della funzione di loss applicata e su come vadano a influire sui risultati.

2.1 Un approccio iterativo alla Tomografia Sparsa

I dati utilizzati nella tesi sono dei volumi generati da delle tomosintesi mammarie digitali, che rappresentano dei volumi quasi tridimensionali del seno di vari pazienti, ottenuti tramite l'elaborazione dei dati di proiezione a poche viste tramite un algoritmo iterativo. Quando non vi è la possibilità di acquisire un insieme completo di proiezioni, il sistema lineare che descrive la cosiddetta Tomografia Sparsa (SpCT) è sottodeterminato.

Il sistema lineare della SpCT è derivato dalla discretizzazione del modello CT continuo, trasformando un problema continuo in uno discreto, adatto a essere risolto tramite l'implementazione di algoritmi di ricostruzione iterativi (Iterative Image Reconstruction). Nel caso della SpCT, il sistema lineare risultante è sottodeterminato, il che rende necessario l'utilizzo di tecniche di regolarizzazione per ottenere una soluzione stabile e significativa. Il problema di ricostruzione del volume di questo tipo è considerato un problema mal condizionato, in cui esistono infinite soluzioni al sistema, quindi l'introduzione di informazioni a priori è necessaria per decidere una delle possibili infinite soluzioni. [16;17;18]

Possiamo porre questo come un problema di ottimizzazione, in cui si cerca di ricostruire un volume partendo da dati incompleti o sottocampionati. Più precisamente, il problema di ottimizzazione viene formulato come la minimizzazione di una funzione obiettivo

$$\arg\min_f F(x; M; g) + \lambda R(f)$$

dove:

- x rappresenta l'immagine da ricostruire
- M è la matrice di proiezione che modella il processo di acquisizione dei dati
- g sono i dati acquisiti.
- λ è un parametro di bilanciamento che controlla il peso relativo dei due termini nella funzione obiettivo.
- $R(f)$ rappresenta il termine di regolarizzazione, che include le informazioni a priori su f. Più precisamente, il termine impone qualche forma di regolarità alla soluzione.

Utilizzando un termine di regolarizzazione sul sistema CT, forza la ricostruzione iterativa (IIR) a cercare una precisa soluzione $\bar{f}$, tra tutte le infinite soluzioni possibili. Visto che molte immagini mediche sono quasi uniformi all'interno degli organi, gran parte del lavoro della ricostruzione iterativa (IIR) usa tecniche per la preservazione dei bordi, dove la funzione di regolarizzazione si concentra di più sulla riduzione del rumore. Una delle funzioni di regolarizzazione più ampiamente utilizzate per i problemi della tomografia sparsa è la funzione di Variazione Totale (VT), nota per preservare i bordi all'interno delle immagini.

Per risolvere questo problema di ottimizzazione, viene considerato un approccio model-based, utilizzando l'algoritmo iterativo noto come Scaled Gradient Projection (SGP), con la quale vengono generati sia i volumi di input che i volumi di target utilizzati in questa tesi. Questo è dovuto al fatto che è particolarmente complesso ottenere dei volumi di ground truth, poiché i volumi reali sono sconosciuti e inaccessibili, in quanto anch'essi devono essere ricostruiti partendo da dati incompleti e imperfetti, ottenuti dai dispositivi medici di acquisizione. Per questo, come input al nostro modello viene dato dei volumi generati dopo soli 2 iterazioni dell'algoritmo iterativo, mentre per il target sono stati generati gli stessi volumi ma usando l'algoritmo iterativo per 30 iterazioni.

2.1.1 Scaled Gradient Projection

Lo Scaled Gradient Projection (SGP) è stato proposto come una soluzione alternativa alle tecniche tradizionali come la Filtered Back Projection (FBP) e la Algebraic Reconstruction Techniques (ART).[1;19] L'SGP è un metodo di proiezione del gradiente accelerato tramite matrice di scaling e una regola di scelta del passo, ottimizzata per il modello SpCT. L'obiettivo del SGP è quello di risolvere il problema di minimizzazione vincolata formulato per la SpCT, cercando di trovare un'immagine che sia coerente con i dati acquisiti e che soddisfi un certo criterio di regolarità, definito dal termine di regolarizzazione, come la Variazione Totale (VT). L'SGP utilizza una matrice di scaling e una regola di scelta del passo, ottimizzato per il modello SpCT; infatti, a ogni iterazione dell'algoritmo esegue:

- Calcolo del Gradiente: Calcola il gradiente della funzione obiettivo nel punto corrente. Il gradiente indica la direzione di massima crescita della funzione, quindi l'algoritmo si muove nella direzione opposta per minimizzare la funzione.
- Scaling del gradiente: Il gradiente viene moltiplicato per una matrice di scaling diagonale. Questa matrice ha lo scopo di migliorare il condizionamento del problema, accelerando la convergenza. I valori della matrice di scaling viene scelto in base alle proprietà del problema e all'iterazione corrente.
- Proiezione: Il punto ottenuto dopo lo scaling viene proiettato sull'insieme delle soluzioni che soddisfano i vincoli del problema. Nel caso della SpCT, un vincolo tipico è la non-negatività dei valori dei voxel. La proiezione assicura che la soluzione rimanga fisicamente realizzabile.
- Ricerca lineare: Viene eseguita una ricerca lineare per determinare la lunghezza del passo da compiere nella direzione del gradiente proiettato, con l'obiettivo di trovare un passo che riduca sufficientemente la funzione obiettivo.
- Aggiornamento: Viene aggiornato con il passo calcolato, ottenendo una nuova stima della soluzione.

L'esecuzione delle iterazioni continua fintanto che l'algoritmo non raggiunge un livello di convergenza, in cui la variazione tra due iterazioni successive è inferiore a una certa soglia prefissata oppure, nel nostro caso, l'algoritmo si arresta dopo un numero massimo di iterazioni definite.

Si è riusciti a dimostrare in varie ricerche come l'utilizzo del SGP porti a una convergenza più rapida rispetto ai metodi del gradiente tradizionali, in particolare nelle prime iterazioni.[20;21;22] Inoltre è noto anche per il suo ruolo nella riduzione degli artefatti e miglioramento della qualità delle immagini.[23;24]. Nonostante l'SGP offra vari vantaggi, bisogna scegliere attentamente gli iperparametri del sistema, in quanto la performance dell'algoritmo è influenzata da essi.

2.2 Reti Convoluzionali (CNN)

Una Rete Convoluzionale (CNN) è un'architettura particolare delle reti profonde, progettata per processare dati che hanno una struttura a matrice, come nel caso delle immagini, e si basa sul concetto dell'operazione matematica di convoluzione. Le CNN sono utilizzate particolarmente per trovare dei pattern nelle immagini, così da riconoscere oggetti, classi e categorie. Facendo parte delle reti neurali profonde, sono composte da molteplici layer, ciascuno dei quali impara a rilevare feature diverse da un'immagine.[25]

A differenza delle reti neurali standard, i neuroni presenti in una CNN non sono completamente connessi, ma ogni neurone si connette solamente ad alcuni neuroni dello stato precedente, riducendo il proprio campo d'interesse.

In una rete convoluzionale, il livello convolutivo svolge un ruolo fondamentale, applicando l'operazione di convoluzione per estrarre le caratteristiche significative dall'input. I parametri addestrabili di questo livello sono rappresentati da filtri (kernel) i quali scorrono sull'intero input per individuare pattern specifici.

Ogni kernel è specializzato nell'estrazione di caratteristiche diverse dell'input, generando diverse feature map che catturano informazioni differenti. L'operazione di convoluzione consiste nello scorrimento del kernel sulla finestra dell'input e nel calcolo del prodotto elemento per elemento tra i valori del kernel e quelli corrispondenti nella finestra stessa. Dopo aver elaborato una finestra iniziale, questa viene spostata di un numero prefissato di posizioni, detto stride o passo, ripetendo l'operazione fino a coprire l'intero volume.

Durante questa fase, ogni kernel applicato all'input possiede un insieme fisso di parametri che rimane uguale mentre scorre lungo l'intero input. Questo consente al kernel di rilevare lo stesso tipo di pattern indipendentemente dalla posizione in cui appare, una proprietà nota come invarianza traslazionale. Inoltre, tutti i neuroni appartenenti alla stessa feature map condividono gli stessi pesi e bias, ossia quelli del filtro che li ha generati, un principio chiamato weight sharing. Questa condivisione dei parametri riduce notevolmente il numero complessivo di pesi da apprendere, migliorando l'efficienza computazionale della rete.

Questo processo rende una rete convolutiva particolarmente adatta a individuare schemi spaziali ricorrenti nei dati di input, come i bordi, le texture e le forme.

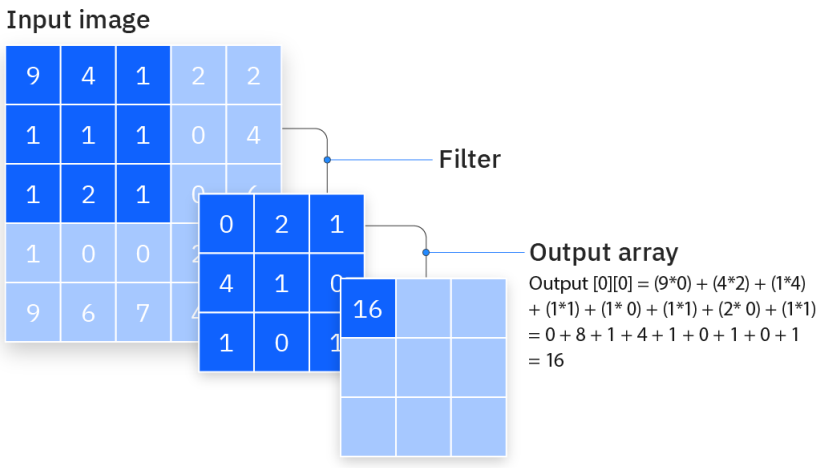

Figura 2.2: Operazione di Convoluzione

Per elaborare anche i bordi dell'input si usa la tecnica del padding, che consiste nell'aggiunta di dati extra ai margini dell'input, permettendo al kernel di applicare la convoluzione anche alle posizioni ai bordi senza perdere informazioni e mantenendo la dimensione dell'input dopo l'operazione.

Dopo ogni operazione di convoluzione, una CNN applica una trasformazione ReLU (Rectified Linear Unit) alla feature map, introducendo la non linearità nel processo di apprendimento, consentendo alla rete di apprendere rappresentazioni più complesse e varie dell'input.

$$\text{ReLU: } f(x) = \max(0, x)$$

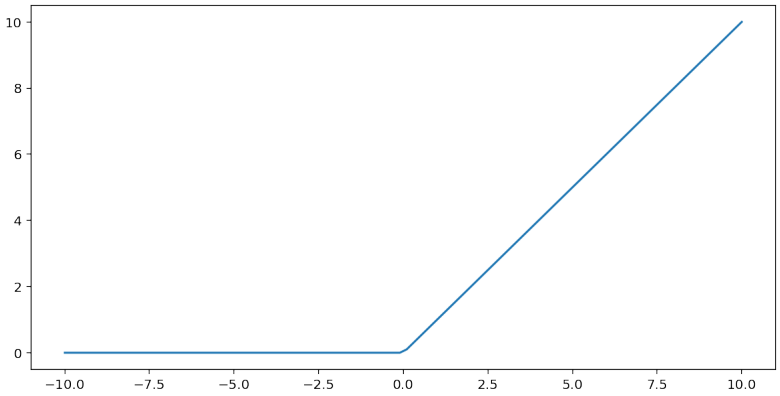

Figura 2.3: Grafico della Relu

L'operazione di convoluzione viene ripetuta con più kernel, il cui numero determina la profondità delle feature maps di output, le quali vengono concatenate per formare un output multidimensionale che rappresenta le caratteristiche estratte dall'input.

Per ridurre le dimensioni spaziali delle feature maps, vengono aggiunti degli strati di pooling (Pooling Layer), che riducono la dimensione spaziale dell'output delle convoluzioni del livello precedente, riducendo al contempo il numero di parametri e il costo computazionale del modello, mantenendo però le caratteristiche più importanti del livello precedente. Questa tecnica, definita downsampling, rende la rete meno sensibile alla posizione esatta delle caratteristiche.
In uno strato di Pooling viene definito una finestra di pooling (Filtro), la quale scorre sull'input applicando l'operazione di pooling generando così delle feature map sottocampionate che conservano le informazioni più importanti dal livello precedente. In questo strato, il calcolo del volume finale dipende sia dalle dimensioni dell'input che dai due iperparametri richiesti: lo stride, che definisce il numero di posizioni che la finestra di

pooling si deve spostare, e la dimensione di quest'ultima.
Uno dei pooling più comuni è il MaxPooling, usato in questa tesi con dimensioni 3x3x3, nella quale la feature map risultante dallo strato convoluzionale precedente viene divisa in regioni, chiamate finestre locali di pooling, e viene preso in considerazione solo il valore massimo presente all'interno della finestra. Un'altra tipologia comune di pooling è l'Average Pooling, la quale calcola la media tra i valori all'interno della finestra.

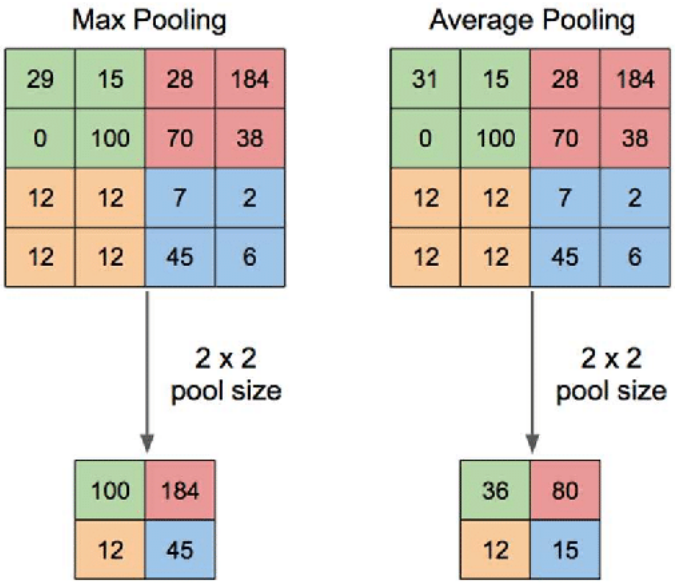

Figura 2.4: Esempio di Max e Avarage Pooling con kernel 2

Source: https://doi.org/10.32604/cmc.2023.037958

Di norma verso la fine della rete, dopo diversi strati di convoluzione e di pooling, le feature maps vengono appiattite in un vettore unidimensionale e passate attraverso uno o più strati completamente connessi, che funzionano come una rete neurale tradizionale. Qui, il modello apprende le combinazioni di alto livello delle caratteristiche estratte dai filtri per effettuare la classificazione o la regressione.[26] Ma visto che questa tesi non ha l'obiettivo di effettuare classificazione o regressione, non verrà analizzato questo livello di una tipica rete convoluzionale.

2.3 Modello UNet 3D

In questa tesi viene proposto l'utilizzo di una rete convoluzionale basata sulla UNet 3D, normalmente utilizzata per la segmentazione e la riduzione del rumore nelle immagini. Questo modello si basa sull'architettura Encoder-Decoder, che suddivide la rete in due

fasi: una fase di encoding e una fase di decoding, le quali possono essere viste come due reti differenti.

La fase di encoding, la quale parte dal livello di input fino al livello intermedio, segue il principio tradizionale di una rete convoluzionale, in cui vengono effettuate operazioni di convoluzione seguite dall'applicazione di una funzione di attivazione e da uno strato di pooling. In questa fase definita come downsampling, avviene la riduzione spaziale dell'input estraendo le caratteristiche principali di quest'ultimo e passando al livello successivo una rappresentazione semplificata dei dati. L'output dell'encoder viene passato al livello intermedio, definito come "collo di bottiglia" o bottleneck, il quale contiene la rappresentazione più compressa dell'input ed è sia lo strato di output della rete di encoder che lo strato di input della rete di decoder. Un obiettivo fondamentale della progettazione e dell'addestramento di questa tipologia di rete è la scoperta del numero minimo di caratteristiche importanti (o dimensioni) necessarie per una ricostruzione efficace dei dati di input. La rappresentazione dello spazio latente, ovvero il codice, che emerge da questo livello viene quindi inserita nel decoder. La rete di decoder, definita come fase di upsampling, ha l'obiettivo di riportare la codifica nelle dimensioni originali, decomprimendo la rappresentazione dei dati, ricostruendo i dati nella loro forma originale precedente alla codifica, applicando in questa fase delle funzioni non lineari. L'output ricostruito viene poi confrontato con la "ground truth", così da valutare l'efficacia della rete.

Esistono delle reti caratterizzate da un sistema di encoder-decoder, denominati come Autoencoder, i quali sono progettati per l'allenamento non supervisionato con l'obiettivo di ricostruire i segnali di input, ma non tutti i modelli encoder-decoder sono considerati Autoencoder. [27]

Nel caso della UNet, l'architettura segue una struttura simile a quella di un Autoencoder. Inizialmente sviluppata da Ronneberger nel 2015 per la segmentazione delle immagini [28], adotta un'architettura a forma di U, da qui il nome, e consiste in una fase di encoding definita anche come percorso contraente o contracting path, e una fase di decoding definita come percorso espansivo o expansive path.

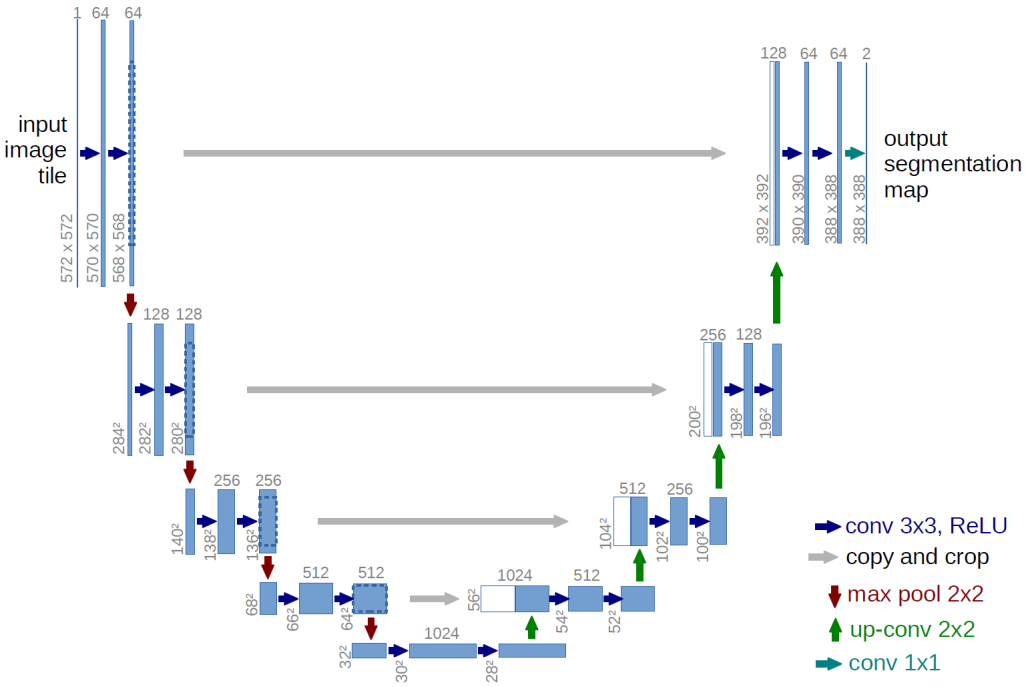

Figura 2.5: Architettura di una rete U-Net

Source: https://doi.org/10.48550/arXiv.1505.04597

La prima fase segue la tipica architettura di una rete convoluzionale, composta da una sequenza di blocchi convoluzionali, i quali definiscono il livello di profondità del modello, che comprendono due strati di convoluzioni, ognuno dei quali è seguito da una funzione di attivazione ReLu. (Figura 2.6).

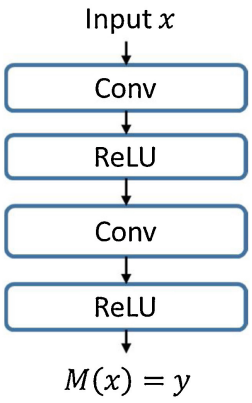

Figura 2.6: Blocco Convoluzionale

A ogni blocco convoluzionale è seguita una fase di Pooling con kernel 2 e stride 2, così da dimezzare la dimensione spaziale dopo ogni livello. Mentre a ogni fase di downsampling viene raddoppiato il numero di filtri.

Nella seconda fase (expansive path) viene eseguito l'up-sampling delle feature maps tramite una convoluzione (up-convolution) che dimezza il numero di filtri, le quali vengono poi concatenate alle feature map generate ai livelli corrispondenti nella prima fase, che vengono prima ritagliate e poi concatenate, questo grazie all'utilizzo delle skip-connections. Questo viene poi seguito da due blocchi convoluzionali (Figura 2.6) composti da due convoluzioni, ognuna seguita da una funzione ReLu, come nella fase di encoding.

Nell'ultima fase del modello, viene applicata una convoluzione con kernel 1, per ridurre le feature map al numero richiesto di canali, generando l'immagine segmentata.[28;29]

A differenza degli Autoencoder, nel modello UNet vi sono presenti delle skip-connections che collegano direttamente gli strati dell'encoder agli strati corrispondenti del decoder. Queste combinano le feature di basso livello dell'encoder con le feature di alto livello del decoder, migliorando così la qualità della ricostruzione finale.[30]

In questa tesi, si utilizza la UNet estesa al 3D, che permette di elaborare dati in tre dimensioni: lunghezza, larghezza e profondità; utilizzando blocchi convoluzionali con kernel 3 e stride 1, impostando il padding a "same", così da mantenere le dimensioni spaziali di input. Mentre nella fase di decoding, viene utilizzato un Transpose Convolutional Block 3D, la quale applica un'operazione che aumenta la dimensione spaziale dei volumi 3D, per riportare l'output alla dimensione originale, come se fosse una deconvoluzione, anche se non effettua propriamente un'operazione di deconvoluzione, in quanto non calcola il vero inverso di una convoluzione.[31]

2.4 Il Dataset

Il dataset utilizzato per allenare e testare il modello proposto è composto da coppie di volumi ricostruiti iterativamente.

Per questioni di memoria vengono estratti dei sotto-volumi dal volume principale, di dimensione 720 x 720 x 16, quest'ultima dimensione per indicare la profondità del sotto-volume. Vengono presi sotto-volumi ritagliati, partendo dall'angolo superiore del volume, spostandosi di volta in volta, su un'asse alla volta di metà volume, quindi 360 x 360 x 8, così da includere anche il contesto dei dati che si ritrovano all'intersezione di due volumi. Il dataset è organizzato per paziente, ognuno identificato da un codice univoco non riconducibile senza ulteriori dati al paziente reale. Ogni paziente ha un volume di una DBT di grandezza variabile, con una profondità che va da 62 a 65 slices.

Per il training sono stati impiegati i volumi di 3 pazienti, mentre il volume di una singola paziente viene utilizzato per la fase successiva di testing.

I volumi di input sono il risultato di 2 iterazioni dell'algoritmo iterativo SGP spiegato nel capitolo 2.1.1, mentre i volumi di target sono stati generati utilizzando lo stesso algoritmo dopo 30 iterazioni. Questo perché i volumi ottenuti con poche iterazioni contengono già un numero significativo di dettagli della struttura esaminata, quindi si cerca, tramite l'utilizzo delle reti neurali profonde, di completare la ricostruzione iniziata dall'algoritmo iterativo interrotto. Vengono usati come ground truth i volumi dopo 30 iterazioni visto che, come spiegato alla fine del capitolo 2.1, non è possibile ottenere i volumi reali in quanto essi devono essere pur sempre ricostruiti partendo da dati incompleti e imperfetti.

2.4.1 Normalizzazione

Nel caso dei volumi ottenuti tramite la DBT, la normalizzazione dei valori di intensità dei volumi rappresenta un passaggio importante per garantire la stabilità durante il training e una buona generalizzazione nella fase di testing. Poiché il range di intensità può variare in base alle impostazioni del macchinario e alle condizioni di acquisizione, si rende necessario uniformare i valori su un intervallo prestabilito, come [0, 1] utilizzando la seguente formula:

$$V_{\text{norm}} = \frac{V - \min(V^*)}{\max(V^*) - \min(V^*)}$$

Dove V_{norm} è il volume normalizzato in un range [0, 1], calcolato utilizzando il minimo e massimo di V^*.

In particolare modo in questa tesi, vengono utilizzati sotto-volumi presi da un volume principale, dove il minimo e massimo variano a seconda del paziente e della regione su cui viene estratto il sotto-volume. Pertanto la normalizzazione dipende da quale V^* viene presa in considerazione.

Vengono definite 3 tipologie di normalizzazione:

- Normalizzazione Locale: Dove si applica la normalizzazione tenendo in considerazione solamente i singoli sotto-volumi, dove V^* rappresenta il singolo sotto volume in utilizzo.
- Normalizzazione a Paziente: In cui si applica la normalizzazione tenendo conto dell'intero volume del paziente sulla quale vengono estratti i sotto-volumi. In questo caso V^* rappresenta il volume intero del paziente.
- Normalizzazione Globale: Viene calcolato il minimo e massimo dell'intero dataset, così da normalizzare tutti i valori secondo quella scala.

Durante l'addestramento del modello, viene presa in considerazione tutte le 3 tipologie di normalizzazione precedentemente elencate, andando ad analizzare come esse influenzino sia l'addestramento del modello, che i risultati predetti da quest'ultimo.

2.5 Iperparametri

Quasi tutti i modelli usati nel Machine Learning hanno due set di parametri: i parametri di training e i metaparametri, quest'ultimi definiti iperparametri.

I parametri di training vengono imparati durante la fase di addestramento del modello, mentre gli iperparametri vengono definiti prima che il training inizi.[32] Essi svolgono un ruolo cruciale in quanto influenzano direttamente la performance del modello, come il tempo di training e l'accuratezza delle predizioni. Questi iperparametri includono il numero di layer nascosti, il numero di feature map iniziali, la grandezza del batch, il numero di epoche, il learning rate, l'ottimizzatore, le funzioni di attivazione e la funzione di perdita.[33] La scelta degli iperparametri corretti per il modello permette di massimizzare le prestazioni e migliorare l'affidabilità.

Nello sviluppo dei vari modelli utilizzati per questa tesi, vengono definiti i seguenti iperparametri:

- Numero di Epoche: definisce il numero di volte che l'algoritmo andrà a lavorare sull'intero set di training. Un numero troppo basso può portare all'underfitting del modello, mentre alzare troppo il numero di epoche può portare all'overfitting.
- Numero di Batch (Batch Size): definisce il numero di campioni del training set che viene usato durante la fase di addestramsnto prima di aggiornare i pesi del modello.
- Funzione di Perdita (Loss Function): considerato come uno dei iperparametri più importanti di una rete neurale, misura la performance del modello calcolando la deviazione del valore predetto rispetto alla ground truth.
- Learning Rate: determina di quanto vengono aggiornati i pesi del modello a ogni iterazione dell'ottimizzatore, controllando così la velocità con cui il modello apprende i dati.
- Funzioni di attivazioni: sono componenti chiave in una CNN, in quanto introducono non linearità, permettendo alla rete di apprendere relazioni complesse nei dati.

In questo caso, la funzione di attivazione nei blocchi convolutivi della UNet3D è la ReLu, mentre per la funzione di attivazione finale del modello viene scelta la funzione sigmoide.

$$\text{ReLU: } f(x) = \max(0, x) \quad \text{Sigmoid: } f(x) = \frac{1}{1 + e^{-x}}$$

é stato scelto l'utilizzo di ADAM come optimizer, il quale implementa un metodo stocastico di discesa del gradiente che si basa sulla stima adattiva del momento di primo e secondo ordine. Questo metodo è considerato computazionalmente efficiente, richiede l'uso di poca memoria e performa bene per i problemi che hanno grandi quantità di dati e parametri.[34] Mentre per il learning rate si è deciso di mantenerlo costante a 10^{-4}.

Il numero di Batch è fissato a 1 per questioni di memoria, risultando così in una discesa del gradiente stocastico in cui vengono calcolati i gradienti e aggiornati i pesi del modello a ogni iterazione.

La scelta della funzione di perdita (Loss Function) ha giocato un ruolo cruciale nello sviluppo delle varie varianti analizzate. Infatti si è provato a utilizzare varie funzioni, tra cui quelle più comuni: la Mean Squared Error (MSE), chiamata L2, che calcola la differenza media quadratica tra la predizione del modello e il target; la Mean Absolute Error (MAE), detta L1, che calcola la media assoluta tra i due valori.

$$L2 = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

$$L1 = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i|$$

In entrambi le funzioni, N è il numero di campioni nel batch su cui si calcola la loss, nel nostro caso 1.

2.6 Funzione di Loss

Visto che l'obiettivo del modello è riuscire a ricostruire partendo dall'input un volume simile al target, si è proposto l'utilizzo di una Structural Similarity Index (SSIM) come funzione di perdita, in quanto non si limita a confrontare le immagini pixel per pixel, ma tiene conto delle caratteristiche strutturali e dei dettagli visivi, i quali sono cruciali nel campo dell'imaging medico.[35] Infatti la SSIM considera la luminosità definita dalla media dei pixel, il contrasto definito dalle varianze dei pixel e infine la struttura, definita dalla covarianza che cattura la relazione spaziale tra i pixel.

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

Nel campo dell'imaging medico, la SSIM può essere considerata la metrica più importante nella ricostruzione, in quanto si allinea di più con la percezione visiva umana ed è una delle metriche più utilizzate nel campo della valutazione qualitativa delle immagini (Imagine Quality Assessment).[36;37] Nella diagnostica per immagini, questa caratteristica è fondamentale, dove la fedeltà strutturale è più rilevante rispetto alla differenza tra i vari pixel.

La SSIM produce un valore tra -1 e 1, dove -1 indica immagini molto diverse o non correlate, mentre 1 rappresenta due immagini identiche. Si presuppone che tutti i valori di SSIM siano ≥ 0, in quanto è improbabile che nei casi reali utilizzati in questa tesi vengano fuori valori inferiori a 0. Per usare la SSIM come funzione di loss è necessario invertire i valori, pertanto per utilizzarla come funzione di loss si usa la seguente formula:

$$SSIM_{\text{Loss}} = 1 - SSIM(V_{\text{target}}, V_{\text{pred}})$$

La $SSIM_{\text{Loss}}$ è ottima per preservare la struttura e la percezione visiva dell'immagine, ma non è sensibile agli errori pixel per pixel e questo può portare a generare sfocature e perdita di dettagli. Infatti si è notato come usando solamente la SSIM come funzione di loss, nonostante l'output predetto sia visivamente simile al target in termini di percezione generale, si può notare come sia affetto da una leggera sfocatura generale.

Per mitigare i lati negativi di usare solo la SSIM Loss come unica funzione di perdita, si è deciso di provare con una combinazione di più funzioni, tra cui la L1 vista in precedenza e la Perceptual Loss, dando più importanza alla SSIM rispetto alle ultime due.
Nel modello, la funzione di loss combinata è definita dalla seguente formula:

$$\text{Loss}_{\text{combined}} = \alpha \cdot \text{Loss}_1 + \beta \cdot \text{Loss}_2 + \gamma \cdot \text{Loss}_3$$

Dove α, β e γ indicano i pesi che bilanciano le rispettive funzioni di loss.

Combinare più funzioni di loss consente di compensare le limitazioni che una singola funzione potrebbe presentare. Per questo motivo, è fondamentale definire con precisione l'obiettivo e il risultato desiderato, così da selezionare le funzioni di loss più adatte a guidare il modello verso una soluzione ottimale[38]. L'uso combinato di diverse funzioni di loss permette di bilanciare i loro punti di forza e debolezza. Ad esempio, combinando SSIM e MSE, il modello può convergere verso soluzioni con una bassa distorsione, mantenendo al contempo i dettagli strutturali dell'immagine.

Nel Capitolo 3 viene analizzato l'impatto significativo della scelta della funzione di loss sui risultati ottenuti con l'obiettivo di migliorare la qualità delle predizioni del modello.

Capitolo 3

Risultati

Nello sviluppo del modello analizzato in questa tesi, sono state generate diverse varianti, modificando di volta in volta vari iperparametri visti nel capitolo 2.5. Verranno analizzate le varianti con un impatto più significativo, quelle che hanno ottenuto risultati inaspettati o più significativi ai fini dello scopo di questa tesi.

Inoltre, si è andato a testare anche come le 3 normalizzazioni descritte nel capitolo 2.4.1 andassero a impattare la UNet3D sviluppata.

In merito a questo, si è notato come l'utilizzo di una normalizzazione locale sia la migliore in termini di qualità dell'output, in quanto riesce a generalizzare meglio con dati mai visti nella fase di testing; questo a discapito di un relativo rallentamento nella convergenza del modello nella fase di addestramento.

Mentre utilizzando una normalizzazione a paziente o una normalizzazione globale, si ottengono dei risultati non troppo diversi da quelli ottenuti rispetto a una normalizzazione locale, ottenendo anche una convergenza del modello leggermente più rapida. Al contempo però, si è notato come utilizzare normalizzazioni del genere rischi di far perdere informazioni locali che con una normalizzazione locale verrebbero più accentuate.

Oltre alla normalizzazione, che si è deciso di tenerla locale per tutte le varianti del modello, i risultati ottenuti da quest'ultimi hanno avuto esiti misti.

Per ogni variante del modello, vengono modificati diversi iperparametri, tra cui il numero di feature iniziali, la funzione di attivazione (sigmoid o ReLU), il tipo di normalizzazione applicata ai dati in fase di caricamento e il numero di epoche. Tuttavia, la dimensione del batch è stata mantenuta costantemente pari a 1 per questioni di memoria.

Le varianti del modello possono essere suddivise principalmente in base alla funzione di loss utilizzata, senza considerare altri iperparametri come il numero di feature iniziali o

la funzione di attivazione. Questo perché, tra tutte le modifiche apportate, la funzione di loss è risultata essere il fattore più influente, soprattutto per quanto riguarda la qualità visiva dei risultati. Il numero di epoche, invece, è stato adattato in base alla capacità della variante considerata di raggiungere una convergenza, con un massimo di 250 epoche prima dell'interruzione dell'addestramento.

Per una valutazione più obiettiva, nella fase di testing, viene calcolata una SSIM iniziale, tra il sotto-volume target e quello di input, così da avere una misura preliminare della loro somiglianza. Questo valore viene poi confrontato con la SSIM finale, calcolata tra il sotto-volume target e quello predetto dal modello in analisi. L'obiettivo è ottenere valori di SSIM finali superiori a quelli iniziali, garantendo al contempo valori di MSE adeguati.

Nella prima versione viene utilizzata la più comune funzione di loss, una MSE che ha dato risultati buoni nella fase di testing, se si dovesse considerare la MSE come unica metrica. Andando ad analizzare i valori di SSIM ottenuti tra il volume predetto e quello target, si nota come la differenza di quest'ultimo non sempre aumenti rispetto al valore della SSIM iniziale. Ciò significa che la qualità visiva e la somiglianza tra il volume target e il volume predetto peggiorano nella maggior parte dei sotto-volumi; infatti è notabile in particolar modo a livello visivo che, al posto di ottenere dei sotto-volumi più chiari e definiti, si ottengono dei risultati in cui si perdono molto i dettagli strutturali dell'immagine. Da notare inoltre, come già dopo 50 epoche, i valori della loss iniziano a oscillare e diminuire molto più lentamente.

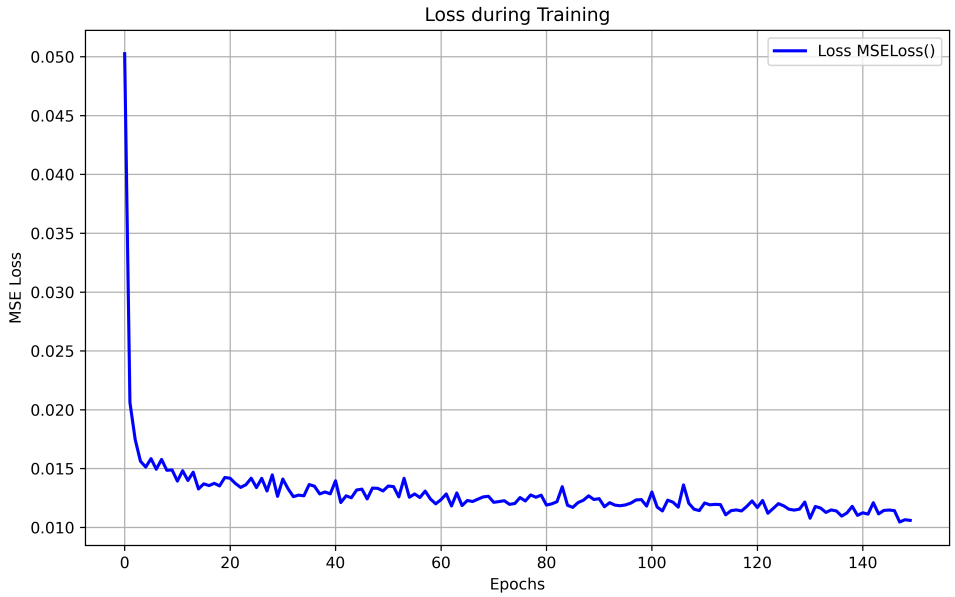

Figura 3.1: Andamento della funzione di loss MSE durante il training nella variante 1.

Nella seconda variante, viene proposto l'utilizzo della L1 come funzione di loss, in quanto la L1 tende a produrre soluzioni preservando meglio i dettagli fini e strutturali nelle immagini, visto che incentiva la minimizzazione degli errori piccoli senza penalizzare eccessivamente quelli più grandi.[39] Per quanto riguarda i risultati ottenuti da questa variante, si nota che, come con la variante precedente, non riesce a ricostruire sempre i volumi in maniera ottimale, producendo sotto-volumi sfocati e con un valore di SSIM finale più basso rispetto a quello di partenza.
Come per la variante precedente, si nota che la funzione di loss inizia a stabilizzarsi e a oscillare intorno alla 50esima epoca.

Visto che si utilizza principalmente la SSIM come metrica di valutazione, dalle varianti successive si è deciso di provare a utilizzare la SSIM come funzione di loss principale.

Infatti, nella terza variante in cui viene utilizzata esclusivamente la $SSIM_{Loss}$, si osserva un netto miglioramento nei risultati predetti. In questa variante, il modello viene addestrato per più epoche, fino a 250, in quanto si è notato come la loss continui a scendere ancora durante tutta la fase di training del modello.

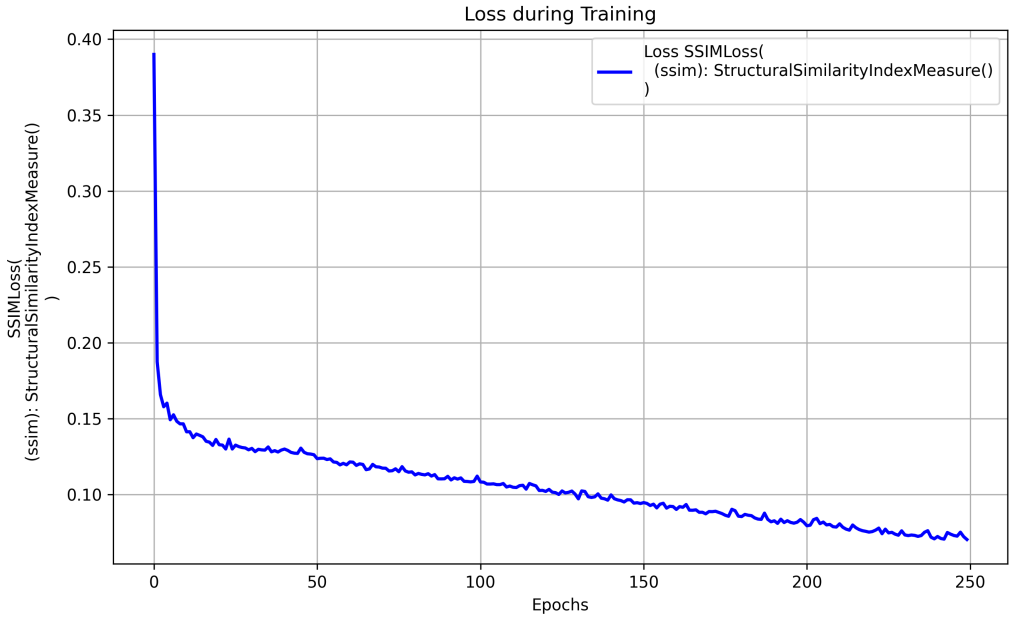

Figura 3.2: Andamento della funzione di loss durante il training nella variante 3.

Durante la fase di testing di questa variante, la SSIM finale è aumentata del 10-20% rispetto alla SSIM iniziale per tutti i sotto-volumi analizzati; in nessun sotto-volume usato si è riscontrato un peggioramento come avvenuto nelle varianti precedenti. Infatti,

anche visivamente, si nota come i sotto-volumi convergano progressivamente verso il target, mettendo in risalto dettagli precedentemente poco visibili nell'input.

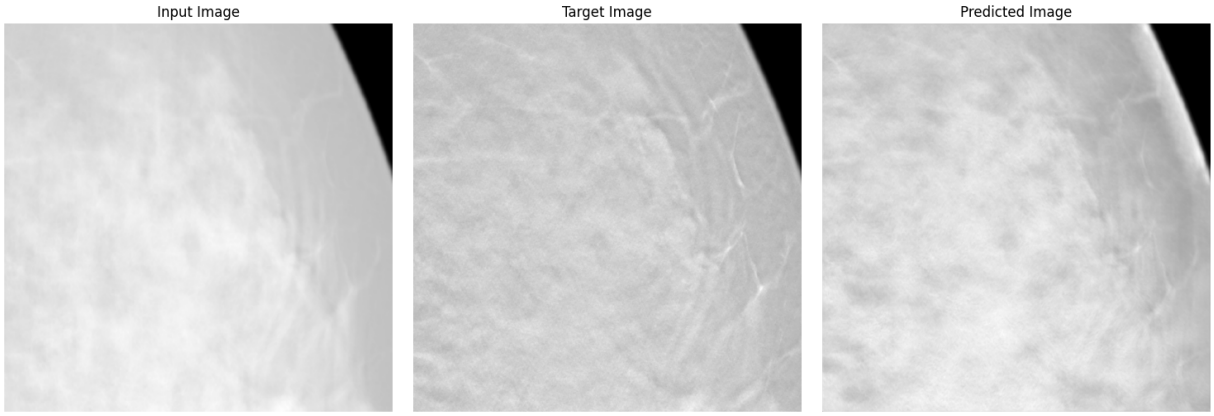

Figura 3.3: Confronto tra input, target e predetto nella variante 3.

Paziente 56550, sotto-volume 12, slice 13

Un difetto che si è potuto notare visivamente nei volumi predetti è che utilizzando solamente la $SSIM_{Loss}$ continuava a rimanere una leggera sfocatura, in particolar modo nelle zone in cui vi sono dettagli più fini.

Per tentare di risolvere questo problema, si è provato a usare una funzione di loss combinata, mantenendo la $SSIM_{Loss}$ come funzione di loss principale, aggiungendo una L1 e una Perceptual Loss.
In questo caso si ottiene una funzione di loss combinata, formata da:

$$Loss_{\text{combined}} = 0.5 \cdot (1 - SSIM) + 0.3 \cdot L1 + 0.2 \cdot Perceptual_{\text{Loss}}$$

I risultati ottenuti da quest'ultima variante sono i migliori sia per l'incremento del valore della SSIM, sia per la qualità visiva dei sotto-volumi generati.
Come per la variante precedente, è stato osservato, come evidenziato nel grafico della figura 3.4, che la loss continua a diminuire in modo costante, contrariamente alle varianti che utilizzano la MSE o L1 come funzioni di loss, come mostrato nella figura 3.1.

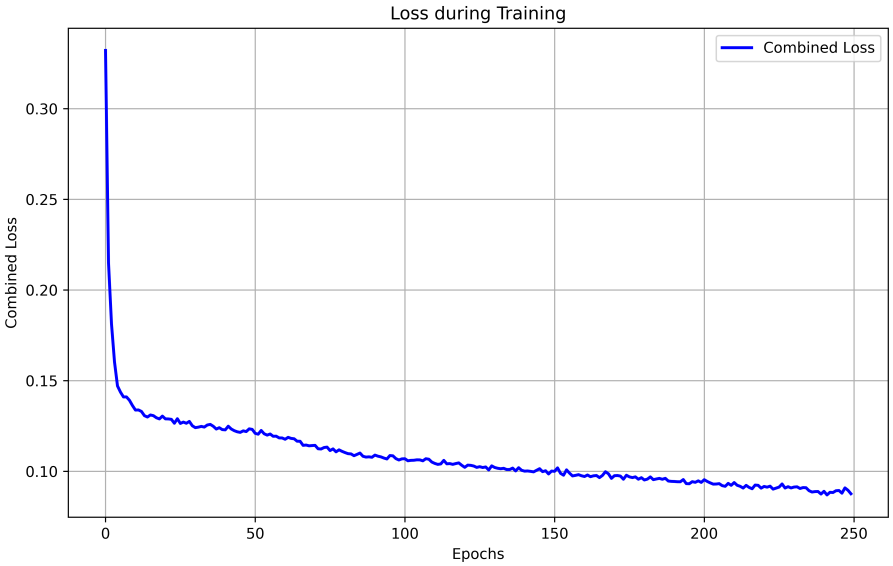

Figura 3.4: Andamento della funzione di loss durante il training nella variante 4.

Visivamente si è potuto osservare che, oltre a mantenere la struttura, è riuscito ad accentuare alcuni dettagli più fini dell'immagine, ottenendo anche un'intensità simile a quella del target.

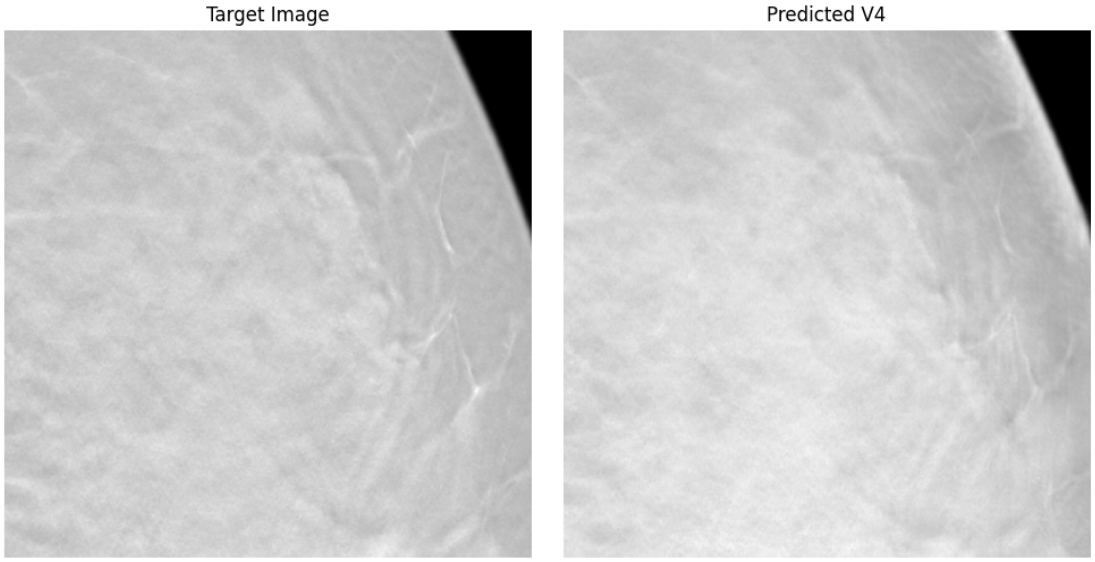

Figura 3.5: Confronto tra target e volume predetto dalla variante 4.
Paziente 56550, sotto-volume 12, slice 13

Si osserva che, tra la variante 3 in cui si utilizza solamente la $SSIM_{Loss}$ e la variante 4 in cui viene utilizzata una Loss combinata, in quest'ultima si riesce a percepire una minore presenza di rumore, con un incremento anche dei dettagli più fini che si possono notare anche nel target.

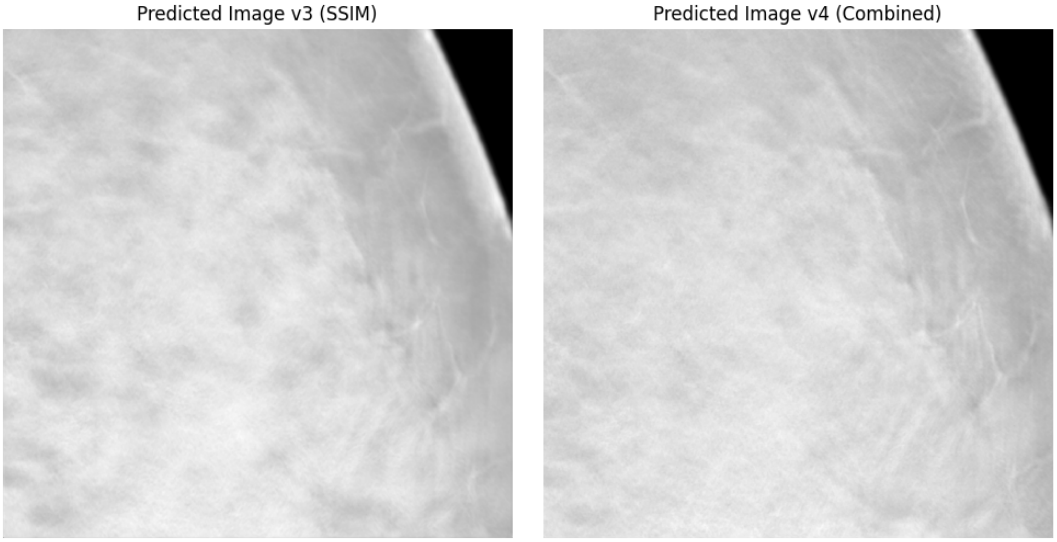

Figura 3.6: Confronto dei risultati della variante 3 e della variante 4
Paziente 56550, sotto-volume 12, slice 13

Inoltre, si è notato come in particolare nei sotto-volumi che presentano delle micro-calcificazioni, il modello sia riuscito a enfatizzare queste zone, rendendo ancora più evidenti la presenza di esse rispetto all'input iniziale.

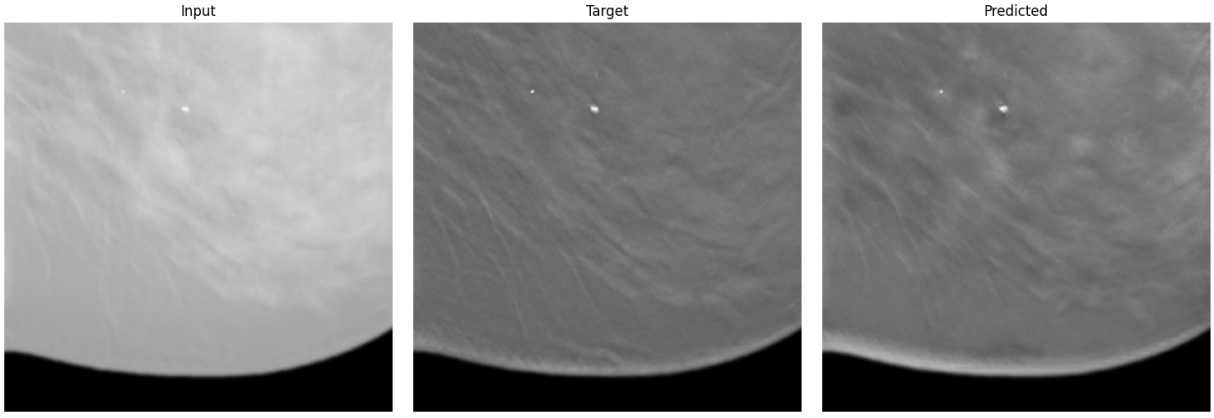

Figura 3.7: Confronto volumi con microcalcificazioni.
Paziente 56550, sotto-volume 25, slice 2

Per ottenere una visione generale della differenza tra le intensità, è stata calcolata e generata un'immagine che rappresenta la differenza pixel per pixel tra il volume predetto e quello target, utilizzando la seguente formula:

$$I_{\text{diff}}(x, y, z) = |I_{\text{target}}(x, y, z) - I_{\text{predicted}}(x, y, z)|$$

Come mostrato nella figura 3.8, il volume derivante dalla differenza tra quello di target e quello predetto evidenzia il grado di accuratezza del modello. In questa rappresentazione, i pixel più scuri indicano una maggiore corrispondenza tra il valore predetto e quello target, mentre i pixel più chiari segnalano una maggiore discrepanza.

Andando ad analizzare i valori dell'intensità nella differenza, si osserva che anche nelle regioni più luminose il valore rimane relativamente basso, attestandosi intorno a 50. Considerando che per un'immagine a 8 bit il range massimo è 255, l'errore risulta contenuto. Se invece si utilizzassero immagini a 16 bit, la differenza apparirebbe ancora meno significativa.

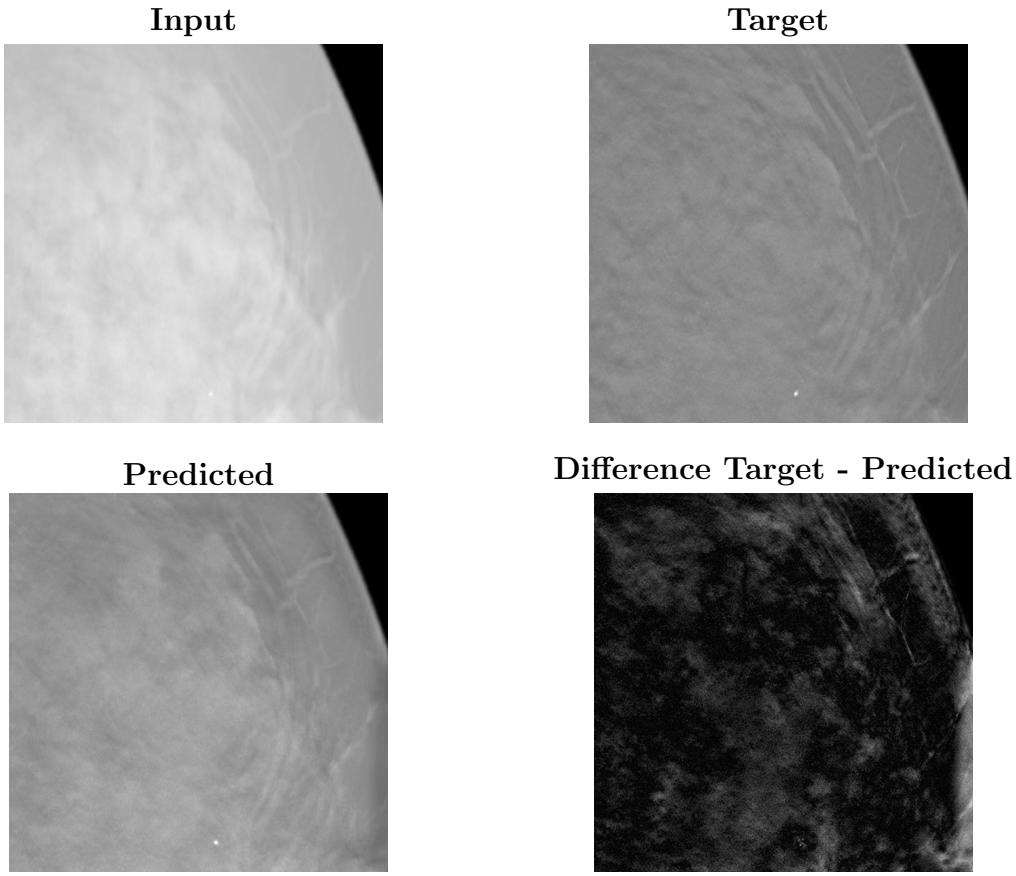

Figura 3.8: Confronto a 4 tra input, target, predetto e differenza.
Paziente 56550, sotto-volume 13, slice 2

Capitolo 4

Conclusioni

L'intelligenza artificiale sta in prima linea nella rivoluzione del campo della diagnostica medica, in particolare con l'avanzamento della ricerca nel campo delle reti neurali profonde è stato possibile introdurre tecniche rivoluzionarie come quelle delle reti convoluzionali. [40]

Questa tesi dimostra che l'impiego delle reti convoluzionali rappresenta un possibile approccio ibrido alla ricostruzione delle immagini 3D di tomosintesi mammaria digitale. In particolare, l'utilizzo di una rete convoluzionale basata sull'architettura di una UNet 3D si è rivelato efficace nella fase intermedia di ricostruzione, iniziata da un metodo iterativo. Nell'ambito reale, è fondamentale che si riesca a ottenere dei risultati accurati dal processo di tomosintesi in poco tempo, così da diagnosticare accuratamente il maggior numero possibile di pazienti, visto che le ricostruzioni tramite i metodi iterativi comportano un elevato uso di risorse computazionali e di tempo.

È stato analizzato, utilizzando varie varianti della UNet 3D, come la funzione di loss impatti significativamente sui risultati predetti dal modello e come la SSIM giochi un ruolo determinante nella qualità dei volumi ottenuti.
Si è osservato inoltre che combinando più funzioni di loss è possibile ottenere risultati che uniscono i punti di forza di ciascuna delle singole loss. In particolare, nella quarta variante, in cui si utilizza la combinazione della L1 e della Perceptual Loss insieme alla SSIM, si è ottenuto un miglioramento significativo nella qualità dei volumi ricostruiti.

É importante evidenziare che, nonostante i risultati ottenuti dal modello proposto, non si è raggiunto ancora un risultato ottimale assoluto. Partendo da questi risultati, si potrebbe provare a sperimentare utilizzando altre funzioni di loss e se necessario, modificare anche l'architettura utilizzata. Per le varianti che utilizzano la SSIM come funzione di loss, comprese quelle combinate, si è notato come nonostante il numero di epoche elevate, il valore della loss continuava a scendere abbastanza rapidamente e sarebbe interessante

vedere se, continuando il training per un numero di epoche maggiori, il modello riuscisse ad arrivare a un punto di convergenza.

Si potrebbe considerare di provare a dare in input alla rete volumi generati dopo più iterazioni, così da poter osservare se il modello riesce a generare dei volumi più vicini a quelli target.

Inoltre, si potrebbero testare architetture simili, come la Residual UNet, l'Attention Res-Unet o la UNet++, per valutare se possano ulteriormente migliorare le prestazioni del modello e generare volumi più nitidi, accentuando i dettagli strutturali più fini.

Riferimenti bibliografici

[1] Davide Evangelista, Elena Morotti, and Elena Loli Piccolomini. Rising: A new framework for model-based few-view ct image reconstruction with deep learning. *Computerized Medical Imaging and Graphics*, 103:102156, 2023.

[2] Dóra Göndöcs and Viktor Dörfler. Ai in medical diagnosis: Ai prediction & human judgment. *Artificial Intelligence in Medicine*, 149:102769, 2024.

[3] Felix Ritter, Tobias Boskamp, A. Homeyer, Hendrik Laue, Michael Schwier, Florian Link, and H.-O. Peitgen. Medical image analysis. *IEEE Pulse*, 2(6):60–70, 2011.

[4] Braxton Norwood. The History of Medical Imaging: From X-Rays to AI-assisted Diagnostics. `https://www.braxtonnorwood.com/the-history-of-medical-imaging-from-x-rays-to-ai-assisted-diagnostics/`, 2025. Accesso il 25 febbraio 2025.

[5] Jacob Beutel. *Handbook of medical imaging*, volume 3. Spie Press, 2000.

[6] Thorsten M Buzug. Computed tomography. In *Springer handbook of medical technology*, pages 311–342. Springer, 2011.

[7] RGMedicali. TAC - Tomografia Computerizzata cos'è e a cosa serve. `https://www.rgmedicali.it/2019/02/18/tac-tomografia-computerizzata-cos-%C3%A8-e-a-cosa-serve/`, 2025. Accesso il 22 febbraio 2025.

[8] Amy Berrington De Gonzalez and Sarah Darby. Risk of cancer from diagnostic x-rays: estimates for the uk and 14 other countries. *The lancet*, 363(9406):345–351, 2004.

[9] James T Dobbins III and Devon J Godfrey. Digital x-ray tomosynthesis: current state of the art and clinical potential. *Physics in medicine & biology*, 48(19):R65, 2003.

[10] Anand K Narayan, Christoph I Lee, and Constance D Lehman. Screening for breast cancer. *Medical Clinics*, 104(6):1007–1021, 2020.

[11] Francesca Caumo, Manuel Zorzi, Silvia Brunelli, Giovanna Romanucci, Rossella Rella, Loredana Cugola, Paola Bricolo, Chiara Fedato, Stefania Montemezzi, and Nehmat Houssami. Digital breast tomosynthesis with synthesized two-dimensional images versus full-field digital mammography for population screening: outcomes from the verona screening program. *Radiology*, 287(1):37–46, 2018.

[12] Mostafa Alabousi, Akshay Wadera, Mohammed Kashif Al-Ghita, Rayeh Kashef Al-Ghetaa, Jean-Paul Salameh, Alex Pozdnyakov, Nanxi Zha, Lucy Samoilov, Anahita Dehmoobad Sharifabadi, Behnam Sadeghirad, et al. Performance of digital breast tomosynthesis, synthetic mammography, and digital mammography in breast cancer screening: a systematic review and meta-analysis. *JNCI: Journal of the National Cancer Institute*, 113(6):680–690, 2021.

[13] Henry Krebs, MD. Traditional vs. 3D mammography: Which is better for you? `https://www.cancercenter.com/community/blog/2021/08/traditional-3d-mammography`, 2021. Accesso il 22 febbraio 2025.

[14] Keri Stephens. Ten-year study shows tomosynthesis improves breast cancer detection. *AXIS Imaging News*, 2024.

[15] Liane Elizabeth Philpotts, Jaskirandeep Kaur Grewal, Laura Jean Horvath, Michelle Young Giwerc, Lawrence Staib, and Maryam Etesami. Breast cancers detected during a decade of screening with digital breast tomosynthesis: Comparison with digital mammography. *Radiology*, 312(3):e232841, 2024.

[16] Emil Y Sidky, Chien-Min Kao, and Xiaochuan Pan. Accurate image reconstruction from few-views and limited-angle data in divergent-beam ct. *Journal of X-ray Science and Technology*, 14(2):119–139, 2006.

[17] Emil Y Sidky, Xiaochuan Pan, Ingrid S Reiser, Robert M Nishikawa, Richard H Moore, and Daniel B Kopans. Enhanced imaging of microcalcifications in digital breast tomosynthesis through improved image-reconstruction algorithms. *Medical physics*, 36(11):4920–4932, 2009.

[18] Emil Y Sidky, Jakob H Jørgensen, and Xiaochuan Pan. Convex optimization problem prototyping for image reconstruction in computed tomography with the chambolle–pock algorithm. *Physics in Medicine & Biology*, 57(10):3065, 2012.

[19] E Loli Piccolomini, Vanna Lisa Coli, Elena Morotti, and Luca Zanni. Reconstruction of 3d x-ray ct images from reduced sampling by a scaled gradient projection algorithm. *Computational Optimization and Applications*, 71:171–191, 2018.

[20] S Bonettini and M Prato. New convergence results for the scaled gradient projection method. *Inverse Problems*, 31(9):095008, aug 2015.

[21] S. Bonettini, Riccardo Zanella, and Luca Zanni. A scaled gradient projection for constrained image deblurring. *Inverse Problems*, 25:015002, 01 2009.

[22] Elena Morotti et al. Reconstruction of 3d x-ray tomographic images from sparse data with tv-based methods. 2018.

[23] Jakob Heide Jørgensen, Tobias Lindstrøm Jensen, Per Christian Hansen, Søren Holdt Jensen, Emil Y Sidky, and Xiaochuan Pan. Accelerated gradient methods for total-variation-based ct image reconstruction. *arXiv preprint arXiv:1105.4002*, 2011.

[24] E. Loli Piccolomini, V. L. Coli, E. Morotti, and L. Zanni. Reconstruction of 3d x-ray ct images from reduced sampling by a scaled gradient projection algorithm. *Comput. Optim. Appl.*, 71(1):171–191, September 2018.

[25] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[26] Elia Giacomini. Convolutional neural networks per il riconoscimento di nudità nelle immagini. 2017.

[27] Dave Bergmann, Cole Stryker. What is an autoencoder? `https://www.ibm.com/think/topics/autoencoder`, 2023. Accesso il 03 marzo 2025.

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 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*, pages 234–241. Springer, 2015.

[29] Nahian Siddique, Sidike Paheding, Colin P. Elkin, and Vijay Devabhaktuni. U-net and its variants for medical image segmentation: A review of theory and applications. *IEEE Access*, 9:82031–82057, 2021.

[30] Song-Toan Tran, Ching-Hwa Cheng, Thanh-Tuan Nguyen, Minh-Hai Le, and Don-Gey Liu. Tmd-unet: Triple-unet with multi-scale input features and dense skip connection for medical image segmentation. In *Healthcare*, volume 9, page 54. MDPI, 2021.

[31] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Rob Fergus. Deconvolutional networks. In *2010 IEEE Computer Society Conference on computer vision and pattern recognition*, pages 2528–2535. IEEE, 2010.

[32] Razvan Andonie and Adrian-Catalin Florea. Weighted random search for cnn hyperparameter optimization. *arXiv preprint arXiv:2003.13300*, 2020.

[33] Prasanna Balaprakash, Michael Salim, Thomas D Uram, Venkat Vishwanath, and Stefan M Wild. Deephyper: Asynchronous hyperparameter search for deep neural networks. In *2018 IEEE 25th international conference on high performance computing (HiPC)*, pages 42–51. IEEE, 2018.

[34] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[35] Daniel Gourdeau, Simon Duchesne, and Louis Archambault. On the proper use of structural similarity for the robust evaluation of medical image synthesis models. *Medical Physics*, 49(4):2462–2474, 2022.

[36] Vicky Mudeng, Minseok Kim, and Se-woon Choe. Prospects of structural similarity index for medical image analysis. *Applied Sciences*, 12(8):3754, 2022.

[37] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[38] Hostin Marc-Adrien, Pirró Nicolas, Bellemare Marc-Emmanuel, et al. Combining loss functions for deep learning bladder segmentation on dynamic mri. In *2021 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI)*, pages 1–4. IEEE, 2021.

[39] Vahid Ghodrati, Jiaxin Shao, Mark Bydder, Ziwu Zhou, Wotao Yin, Kim-Lien Nguyen, Yingli Yang, and Peng Hu. Mr image reconstruction using deep learning: evaluation of network structure and loss functions. *Quantitative imaging in medicine and surgery*, 9(9):1516, 2019.

[40] Johnson Oyeniyi and Paul Oluwaseyi. Emerging trends in ai-powered medical imaging: enhancing diagnostic accuracy and treatment decisions. *International Journal of Enhanced Research In Science Technology & Engineering*, 13:2319–7463, 2024.

Ringraziamenti

Un ringraziamento speciale alla Prof.ssa Piccolomini per avermi proposto l'argomento di questa tesi, per essere stata sempre disponibile e sorridente durante tutto il mio percorso di sperimentazione e stesura della tesi, per tutti i chiarimenti e le risorse preziose fornitemi in questi ultimi mesi.

Un ringraziamento speciale va anche al Prof. Evangelista che con la sua gentilezza e competenza mi ha supportato nella parte sperimentale e tecnica della tesi, riuscendo a farmi tornare la voglia di esplorare e provare nuove tecnologie mai utilizzate.

Ringrazio entrambi i professori che mi hanno dedicato il loro tempo e mi sono venuti incontro per tutte le esigenze che avevo.

Ringrazio anche tutti i colleghi universitari con cui ho avuto il piacere di collaborare in tutti questi anni, con cui ho scambiato appunti, con cui ho fatto progetti e preparato gli esami.

Un ringraziamento va anche a tutti gli amici *online* che ho conosciuto durante il mio periodo universitario, che mi hanno fatto divertire e che mi hanno supportato moralmente in tutti questi anni, in particolar modo coloro con cui ho avuto il piacere di incontrarmi anche dal vivo.

E infine per ultimo, ma non per importanza, un ringraziamento va anche a tutta la mia grande famiglia che in tutti questi anni mi ha supportato e non mi ha fatto pesare l'eccessivo tempo trascorso iscritto all'università.