



**Politecnico
di Torino**

Politecnico di Torino

Corso di Laurea Magistrale in Ingegneria Biomedica

A. A. 2022/2023

Sessione di Laurea Marzo 2023

Generazione di sMRI per pazienti affetti da ictus

Relatore:

Ing. Massimo Salvi

Correlatore:

Prof. Filippo Molinari

Candidato:

Alessandro Borio

Sommario

INDICE FIGURE	3
INDICE TABELLE	4
RIASSUNTO	6
1 INTRODUZIONE	8
2 STATO DELL'ARTE	11
3 CREAZIONE DEL DATASET:	12
3.1 ELASTIX:.....	14
3.2 REGISTRAZIONE	16
3.3 PRE-PROCESSING.....	19
3.4 MASCHERE LESIONE	21
3.5 DATA AUGMENTATION	22
4 MODELLI GENERATIVI	24
4.1 CGAN: PIX2PIX	24
4.2 GAN: CYCLEGAN	28
4.3 DIFFUSION MODELS.....	30
5 METRICHE DI VALUTAZIONE	32
6 DETTAGLI D'IMPLEMENTAZIONE	34
7 RISULTATI	37
7.1 TRADUZIONE TC-MRI FLAIR	38
7.2 TRADUZIONE TC-MRI DWI	44
8 CONCLUSIONI	50
8.1 SVILUPPI FUTURI.....	51
BIBLIOGRAFIA	53
RINGRAZIAMENTI	55

Indice Figure

FIGURA 1. ICTUS CEREBRALE	8
FIGURA 2: TC STD 2.5(1), RM FLAIR (2), RM DWI (3)	12
FIGURA 3: SOVRAPPOSIZIONE FRAME RM (VIOLA) E TC (VERDE) NON REGISTRATI	13
FIGURA 4: SCHEMA A BLOCCHI REGISTRAZIONE TRAMITE ELASTIX	14
FIGURA 5: PROCESSO DI REGISTRAZIONE E VALIDAZIONE. (1) IDENTIFICAZIONE DEL VOLUME FISSO (RM) E IN MOVIMENTO (TC). (2) FASE DI REGISTRAZIONE MEDIANTE "ELASTIX". (3) CONTROLLO VISIVO MEDIANTE PAIRING DEI VOLUMI RISULTANTI SU PIATTAFORMA MATLAB.	17
FIGURA 6. ESEMPIO FRAME TC NON PROCESSATO (1) E FRAME PROCESSATO E CONVERTITO (2).	18
FIGURA 7. SCHEMA A BLOCCHI SU TECNICA DI REGISTRAZIONE DATI PROCESSATI. (1) VOLUME FISSO (RM), (2) VOLUME MOBILE CT PROCESSATO DA (3), (4) PRIMA REGISTRAZIONE, (5) SECONDA REGISTRAZIONE TRA VOLUME OUTPUT USCENTE DA (4) POSTO COME FISSO E (3) COME MOBILE, (6) PAIRING DI VALIDAZIONE.	18
FIGURA 8. PRE-PROCESSING TC DA (1) A (2).	19
FIGURA 9. PRE-PROCESSING RM FLAIR DA (1) A (2).	20
FIGURA 10. DATA AUGMENTATION. A) FRAME ACCOPPIATI TC E RM FLAIR ORIGINALI, B) TRASFORMAZIONE 1 (FLIP ORIZZONTALE E ROTAZIONE RANDOM); C) TRASFORMAZIONE 2 (DISTRORSIONE GRIGLIA E ROTAZIONE RANDOM).	23
FIGURA 11 MASCHERA BINARIA: (A) FRAME IN RM POSITIVO AD ICTUS, (B) MASCHERA BINARIA OTTENUTA MEDIANTE THRESHOLDING.....	23
FIGURA 12. SCHEMA A BLOCCHI PIX2PIX	25
FIGURA 13. SCHEMA A BLOCCHI UNET CON SKIP CONNECTIONS [15]	26
FIGURA 14. PATCH GAN [11]	27
FIGURA 15. SCHEMA A BLOCCHI CYCLEGAN	28
FIGURA 16. NOISING PROCESS	30
FIGURA 17. DENOISING PROCESS	31
FIGURA 19. PIX2PIX TC-FLAIR: RMSE, PSNR E SSIM SU TRAINING E VALIDATION (VALOR MEDIO \pm STD)	39
FIGURA 19. PIX2PIX TC-FLAIR: MAE E ME CALCOLATI SU MASCHERA DI LESIONE E FRAME (VALOR MEDIO \pm STD)	39
FIGURA 20. CYCLEGAN TC-FLAIR: RMSE, PSNR E SSIM SU TRAINING E VALIDATION (VALOR MEDIO \pm STD)	41
FIGURA 21. CYCLEGAN TC-FLAIR: MAE E ME CALCOLATI SU MASCHERA DI LESIONE E FRAME (VALOR MEDIO \pm STD)	41
FIGURA 22. TRADUZIONE FRAME DI TEST MODELLI MIGLIORI TC-FLAIR	43
FIGURA 23. PIX2PIX TC-DWI: MAE E ME CALCOLATI SU MASCHERA DI LESIONE E FRAME (VALOR MEDIO \pm STD)	45
FIGURA 24. PIX2PIX TC-DWI: RMSE, PSNR E SSIM SU TRAINING E VALIDATION (VALOR MEDIO \pm STD)	45
FIGURA 25. CYCLEGAN TC-DWI: RMSE, PSNR E SSIM SU TRAINING E VALIDATION (VALOR MEDIO \pm STD)	47
FIGURA 26. CYCLEGAN TC-DWI: MAE E ME CALCOLATI SU MASCHERA DI LESIONE E FRAME (VALOR MEDIO \pm STD)	47
FIGURA 27. TRADUZIONE FRAME DI TEST MODELLI MIGLIORI TC-DWI	49

Indice Tabelle

TABELLA 1. COMPOSIZIONE CASISTICA A DISPOSIZIONE	12
TABELLA 2. PARAMETRIZZAZIONI PIX2PIX, DOVE NDF È IL NUMERO DI FILTRI PER DISCRIMINATORE, NGF DEL GENERATORE E GAN_MODE LA TIPOLOGIA DI GAN UTILIZZATA.....	34
TABELLA 3. PARAMETRIZZAZIONE CYCLEGAN, CON NDF NUMERO DI FILTRI DEL DISCRIMINATORE NGF DEL GENERATORE E GAN_MODE TIPOLOGIA DI GAN UTILIZZATA.....	35
TABELLA 4. PARAMETRIZZAZIONE DIFFUSION MODELS.....	36
TABELLA 5. RISULTATI PIX2PIX TC-FLAIR, PARAMETRIZZAZIONE SBILANCIATA	38
TABELLA 6. RISULTATI PIX2PIX TC-FLAIR, PARAMETRIZZAZIONE BILANCIATA.....	38
TABELLA 7. RISULTATI CYCLEGAN TC-FLAIR, PARAMETRIZZAZIONE SBILANCIATA.....	40
TABELLA 8. RISULTATI CYCLEGAN TC-FLAIR, PARAMETRIZZAZIONE BILANCIATA	40
TABELLA 9. RISULTATI DIFFUSION MODELS TC FLAIR	42
TABELLA 10. RISULTATI PIX2PIX TC-DWI, PARAMETRIZZAZIONE SBILANCIATA	44
TABELLA 11. RISULTATI PIX2PIX TC-DWI, PARAMETRIZZAZIONE BILANCIATA	44
TABELLA 12. . RISULTATI CYCLEGAN TC-DWI PARAMETRIZZAZIONE SBILANCIATA	46
TABELLA 13. . RISULTATI CYCLEGAN TC-DWI, PARAMETRIZZAZIONE BILANCIATA	46
TABELLA 14. RISULTATI DIFFUSION MODELS TC- DWI	48

Riassunto

L'ictus cerebrale, noto anche come ictus o colpo apoplettico, è una patologia cerebrovascolare che si verifica quando il flusso sanguigno del cervello viene interrotto o ridotto a causa di un'improvvisa chiusura o rottura di un vaso sanguigno.

Ciò comporta un grave danno alle cellule cerebrali dovuto dalla mancanza di ossigeno e nutrienti trasportati dal sangue (ischemia) o alla compressione provocata dalla fuoriuscita di sangue dal vaso (emorragia cerebrale).

La diagnosi precoce dell'ictus, sia ischemico che emorragico, è di fondamentale importanza per avere buone possibilità che il trattamento abbia effetti positivi nel salvare la vita al paziente.

Secondo il protocollo, la prima indagine richiesta quando ci si trova dinanzi a una sintomatologia sospetta è la TC (tomografia computerizzata) cerebrale del paziente. Esso, infatti, è un macchinario largamente diffuso ed accessibile, di rapida esecuzione e, cosa più importante, permette di identificare già in prima istanza l'evento emorragico rispetto a quello ischemico, fattore essenziale per una corretta impostazione del piano terapeutico.

Lo strumento diagnostico che però risulta essere più sensibile per la diagnosi è la risonanza magnetica, che, grazie alle scansioni ottenute per diffusione e perfusione, è in grado di identificare con maggior accuratezza l'ischemia in fase iperacuta e le aree di "penombra ischemica", ovvero aree non ancora infartuate ma che sono a rischio metabolico e dunque passabili di recupero con le opportune terapie.

Al contrario della TC però, tale strumento oltre a non essere presente in tutte le cliniche, possiede tempi d'attesa più lunghi, che possono compromettere la buona riuscita del trattamento.

Il seguente progetto di tesi ha come obiettivo il verificare se le scansioni TC di soggetti dichiarati positivi ad ictus possano contenere informazioni importanti per la diagnosi che in fase acuta non sono chiari ad occhio nudo, ma che con l'ausilio di mezzi esterni possono essere messi in evidenza.

A tale scopo sono state indagate le recenti nuove tecnologie nell'ambito delle intelligenze artificiali, ovvero le reti d'apprendimento automatico di tipo generativo condizionale.

Nello specifico verrà valutata la loro capacità nell'effettuare *domain transfer*, ovvero eseguire una traduzione a partire da un'immagine data in input ottenendo in uscita l'output desiderato.

In questo modo verrà verificato in prima istanza se tali modelli siano in grado di generare immagini sintetiche di risonanza magnetica realistiche che rappresentino la corrispondente scansione TC di partenza, e in secondo luogo se siano anche in grado di individuare pattern che rappresentano la lesione nella prima modalità, traducendola successivamente in output.

1 Introduzione

L'ictus cerebrale è una patologia cerebrovascolare che si verifica quando il flusso sanguigno del cervello viene interrotto o ridotto a causa di un'improvvisa chiusura o rottura di un vaso sanguigno.

Esso in Italia rappresenta la terza causa di morte [1], dopo le malattie cardiovascolari e le neoplasie, e la prima causa assoluta di disabilità.

Può essere classificato in due tipologie principali: ictus ischemico ed emorragico.

- L'ictus ischemico si verifica quando il flusso di sangue verso il cervello viene interrotto a causa di un coagulo di sangue o di un'ostruzione di un vaso sanguigno. Questo tipo di ictus rappresenta circa l'80% di tutti gli ictus cerebrali [2]. I sintomi dell'ictus ischemico possono includere paralisi o debolezza improvvisa in un lato del corpo, difficoltà nella parola o nel linguaggio, perdita improvvisa di vista o visione doppia, vertigini o difficoltà nella deambulazione.
- L'ictus emorragico si verifica quando un vaso sanguigno nel cervello si rompe e il sangue fuoriesce nel tessuto cerebrale circostante. Questo tipo di ictus rappresenta circa il 20% di tutti gli ictus cerebrali. I sintomi dell'ictus emorragico possono includere mal di testa intenso, nausea e vomito, perdita di coscienza, convulsioni e difficoltà respiratorie.

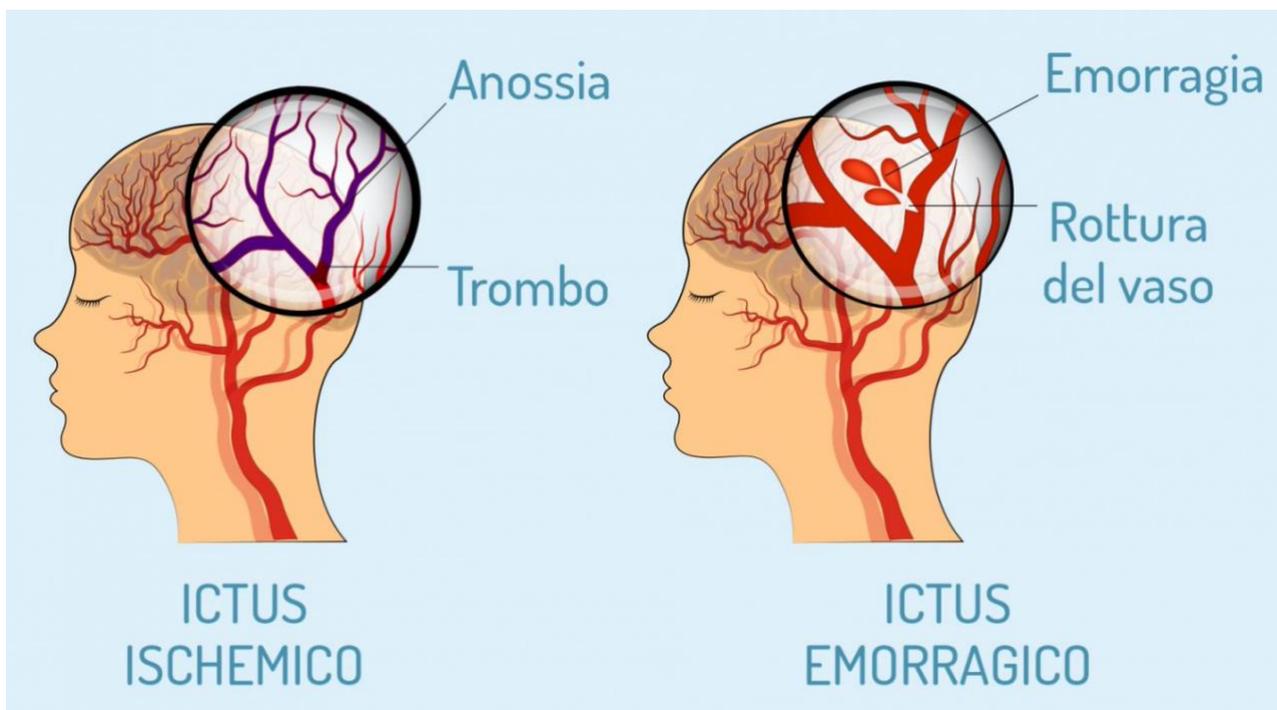


Figura 1. Ictus Cerebrale

Il protocollo diagnostico per l'ictus cerebrale prevede una valutazione neurologica e delle funzioni vitali, seguita da test di imaging del cervello, come la tomografia computerizzata (TC) o la risonanza magnetica (MRI).

La TC viene spesso utilizzata come test di screening iniziale in quanto è più rapida e meno costosa della MRI, ma può essere meno sensibile nella rilevazione di lesioni cerebrali acute.

La MRI è considerata il gold standard nella diagnosi dell'ictus cerebrale in quanto offre una maggiore sensibilità e specificità nella rilevazione di lesioni cerebrali acute, in particolare per gli ictus ischemici. La MRI può anche essere utilizzata per identificare la zona del cervello interessata dall'ictus e valutare la gravità dell'ictus.

Essa, però, non sempre è prontamente disponibile, il che può provocare ritardi di diagnosi rischiando di compromettere la buona riuscita del trattamento, essendo la tempestività nel programmare il piano terapeutico l'elemento cruciale.

Il seguente progetto di tesi ha come obiettivo il verificare se le scansioni TC di soggetti dichiarati positivi ad ictus possano contenere informazioni importanti per la diagnosi che in fase acuta non sono chiari ad occhio nudo, ma che con l'ausilio di mezzi esterni possono essere messi in evidenza.

A tale scopo verranno indagate le recenti nuove tecnologie nell'ambito delle intelligenze artificiali, ovvero le reti d'apprendimento automatico di tipo generativo condizionale.

Nello specifico verrà valutata la loro capacità nell'effettuare *domain transfer*, ovvero eseguire una traduzione a partire da un'immagine data in input ottenendo in uscita l'output desiderato.

La parte metodologica seguita nel corso di questo studio prevederà innanzitutto la raccolta della casistica necessaria allo sviluppo delle reti neurali.

In particolare, verranno costruiti due dataset distinti, uno dato dall'accoppiamento delle scansioni TC-MRI DWI (*Diffusion Weighted Imaging*), mentre il secondo da TC-MRI FLAIR (*Fluid Attenuated Inversion Recovery*).

Di questi verrà narrata la fase di registrazione, ovvero il riallineamento spaziale dei volumi, fattore essenziale per l'ottenimento di dataset accoppiati e per l'allenamento di modelli generativi che effettuino "*image to image translation*".

In seguito, verranno descritte le tre architetture utilizzate per la fase di training, ovvero le cGAN (conditional Generative Adversarial Networks) di tipologia Pix2Pix e CycleGAN e i recenti Diffusion Models.

Di queste verranno messe in evidenza le principali differenze ed elencate le prove di parametrizzazione interna che son state eseguite al fine di individuare la composizione che dessero le prestazioni migliori.

A tal fine, i migliori modelli per ogni tipologia verranno individuati mediante il calcolo delle principali metriche qualitative sulle immagini generate rispetto a quelle di riferimento. Nello specifico verranno indagate le prestazioni in merito sia alla qualità generale di traduzione, che per il riconoscimento dell'area con presenza di lesione.

Infine, verranno effettuate le valutazioni visive, confrontando i dati generati su pazienti di test positivi ad ictus tra le tipologie di rete selezionate.

2 Stato dell'arte

Nell'ambito dell'imaging medico, i modelli generativi sono utilizzati per la generazione di immagini a partire da dati di input, come ad esempio immagini di tomografia computerizzata (CT) o risonanza magnetica (MRI).

Attualmente, i modelli generativi più utilizzati in questo campo sono le reti generative avversarie (GAN) e le reti generative avversarie condizionali (cGAN). Questi modelli sono stati applicati con successo alla sintesi di immagini di radiologia, tomografia a emissione di positroni (PET) e risonanza magnetica.

Una delle applicazioni più comuni dei modelli generativi in ambito medico è la sintesi di immagini a bassa risoluzione in immagini ad alta risoluzione [3], migliorando così la qualità dell'immagine e facilitando la diagnosi.

Inoltre, i modelli generativi possono anche essere utilizzati per la generazione di dati di training aggiuntivi, utili per migliorare la precisione dei modelli di apprendimento automatico. Ad esempio, i dati sintetici generati possono essere utilizzati per la classificazione di immagini mediche o per la segmentazione di immagini in diverse regioni anatomiche [4].

Per quanto riguarda invece la loro capacità nell'effettuare traduzioni cross modali TC-RM, la ricerca in questo campo è particolarmente attiva negli ultimi anni.

In particolare, numerosi sono gli studi che indagano l'utilizzo di modelli generativi per la pianificazione di trattamenti radioterapici grazie alla sintesi RM-TC, la quale permetterebbe di ottenere immagini a maggior risoluzione e una minor esposizione da parte del paziente a radiazioni ionizzanti [5], [6].

In merito invece al seguente progetto di tesi, ovvero l'individuazione di aree affette da ictus cerebrale tramite immagini sintetiche di risonanza magnetica ottenute da input TC, solo di recente è diventato oggetto di studio.

A questo proposito si può citare [7], il quale propone lo sviluppo di modelli generativi allenati a tale scopo utilizzando come input le scansioni TC unite alle mappe ottenute in seguito alle indagini radiomiche.

3 Creazione del dataset:

Lo sviluppo del progetto di tesi è stato reso possibile grazie alla collaborazione con il reparto di neurologia dell'ospedale "Maria Vittoria" di Torino, il quale ha fornito la casistica necessaria per la formazione del dataset d'allenamento.

Nello specifico sono stati raccolti ed elaborati un complessivo di 15 pazienti contenenti le scansioni TC e le relative RM in modalità FLAIR ottenute entro le 72 ore dal primo esame, e 23 pazienti alle cui TC vengono abbinare le RM in modalità DWI.

Modalità	P	N	Tot
CT(Std 2.5)-MRI (FLAIR)	14	1	15
TC(Std 2.5)- MRI (DWI)	21	2	23

Tabella 1. Composizione casistica a disposizione

Gli esami diagnostici sono stati eseguiti all'interno dello stesso centro clinico di cui per i volumi CT son state utilizzate le serie "Std 2.5" ottenute tramite scanner della "GE Medical Systems – Revolution EVO" 42 kV, mentre per quanto riguarda le scansioni di risonanza magnetica è stato utilizzato il Siemens "Aera".

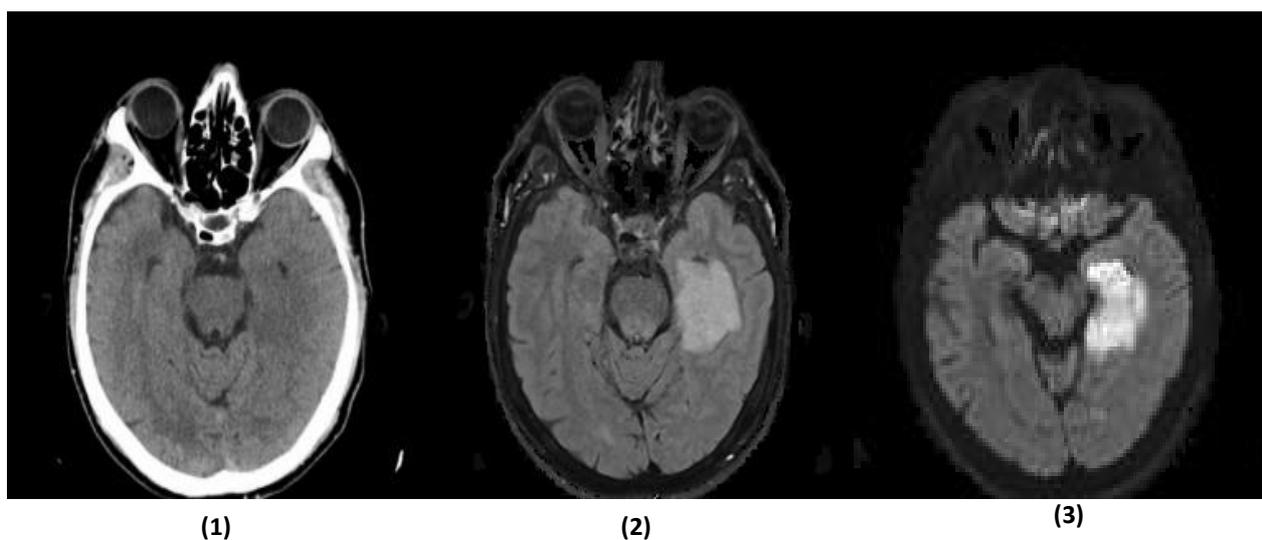


Figura 2: TC Std 2.5(1), RM FLAIR (2), RM DWI (3)

La condizione essenziale per l'allenamento di modelli generativi che eseguano *image to image translation* è la corrispondenza spaziale tra i frame di input e output.

In Fig.2 viene mostrato un esempio su quelle che sono le condizioni iniziali dei volumi in analisi. Come si può facilmente notare, a causa delle diverse risoluzioni spaziali lungo le tre dimensioni con le quali i due macchinari lavorano, i dati risultano essere disallineati, non garantendo dunque una corrispondenza biunivoca.

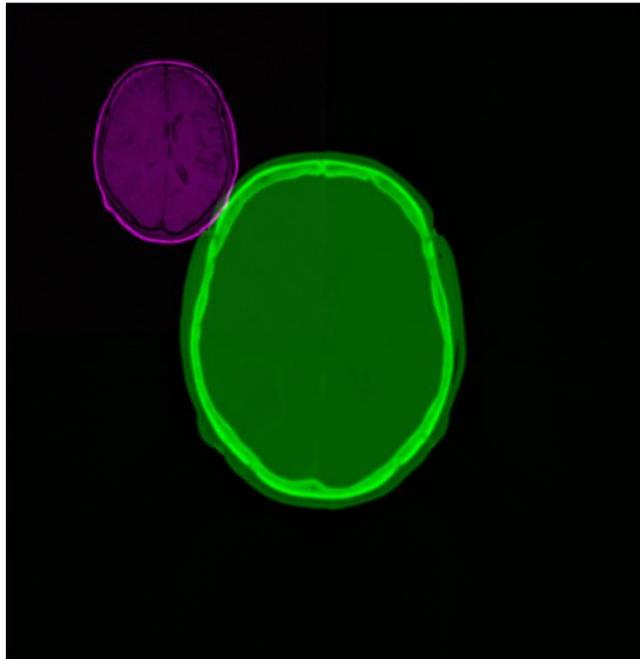


Figura 3: Sovrapposizione frame RM (viola) e TC (verde) non registrati

Per questo motivo il primo passaggio per la costruzione del dataset completo consiste nell'allineamento dei volumi a disposizione mediante registrazione.

La piattaforma utilizzata in questa prima fase è il software medico open source "*3D Slicer*", il quale permette di effettuare la renderizzazione automatica dei volumi a disposizione e l'installazione di tool esterni per le registrazioni.

A questo scopo è stata utilizzata l'estensione "*Elastix*", la quale permette l'esecuzione di registrazioni parametriche dati due volumi di input. Le caratteristiche tecniche verranno descritte in seguito.

3.1 Elastix:

Elastix è un software open-source per la registrazione di immagini medicali [8].

Il suo funzionamento base prevede l'identificazione di un'immagine in movimento $I_M(x)$ e di una fissa $I_F(x)$ di dimensioni d , in cui alla prima deve essere applicata una trasformazione $T(x)$ (eq. 1) tale per cui $I_M(T(x))$ risulti essere spazialmente allineata a $I_F(x)$.

$$T(x) = x + u(x) \quad (1)$$

La soluzione a questo quesito viene generalmente risolto mediante un problema di ottimizzazione in cui viene minimizzato un costo di funzione C (eq. 2)

$$\hat{T} = \arg \min C(T; I_F; I_M) \quad (2)$$

$$C(T; I_F; I_M) = -S(T; I_F, I_M) + \gamma P(T) \quad (3)$$

Dove S rappresenta la misura di similarità e γ il sistema dei pesi.

L'approccio utilizzato da Elastix è di tipo parametrico, in cui il numero di possibili soluzioni è limitato dall'introduzione di un vettore parametrizzato μ contenente tutti i valori di trasformazione necessari (eq. 4).

$$\hat{T}_\mu = \arg \min_{T_\mu} C(T_\mu; I_F, I_M) \quad (4)$$

$$\hat{\mu} = \arg \min_{T_\mu} C(\mu; I_F, I_M) \quad (5)$$

Le componenti dell'algoritmo di registrazione che permettono di attuare quanto detto vengono riassunte nello schema a blocchi in Fig. 4.

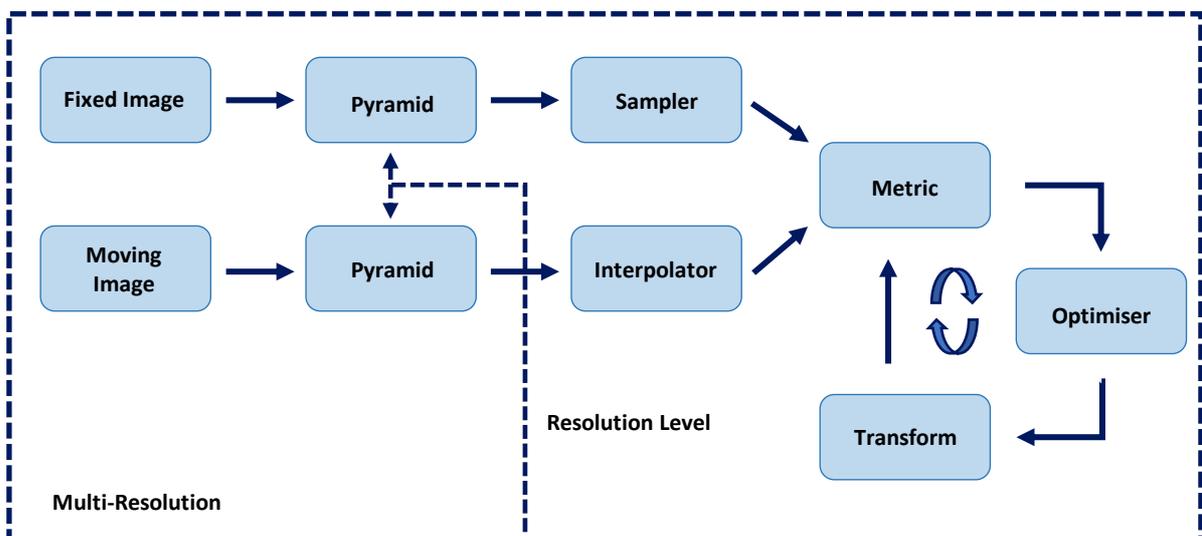


Figura 4: Schema a blocchi registrazione tramite Elastix

Per questo progetto di tesi tra le parametrizzazioni disponibili all'interno del tool sono state selezionate le seguenti opzioni:

- **Metrica:** Mutual Information (MI), la cui definizione fornita da [9] è descritta in (eq.6)

$$MI(\mu; I_F, I_M) = \sum_{m \in L_M} \sum_{f \in L_F} p(f, m; \mu) \log_2 \left(\frac{p(f, m; \mu)}{p_F(f) p_M(m; \mu)} \right) \quad (6)$$

dove L_M e L_F sono insiemi di centri di intensità regolarmente distanziati, p la probabilità congiunta discreta e p_F e p_M le probabilità discrete marginali dell'immagine fissa e in movimento, ottenute sommando p rispettivamente su m e f .

- **Ottimizzatore:** Adaptive Stochastic Gradient Descent, versione più avanzata formulata da [10] derivante dalla discesa del gradiente standard ove la direzione di ricerca si focalizza sul gradiente negativo della funzione di costo (eq.7)

$$\mu_{k+1} = \mu_k - a_k g(\mu_k) \quad (7)$$

- **Trasformazione:** B-splines, tipicamente utilizzata per trasformazioni elastiche (eq. 8)

$$T_\mu(x) = x + \sum_{x_k \in N_x} p_k \beta^3 \left(\frac{x - x_k}{\sigma} \right) \quad (8)$$

con x_k i punti di controllo, $\beta^3(x)$ il polinomio cubico multidimensionale B-spline [3], σ la spaziatura del punto di controllo e N_x l'insieme di tutti i punti di controllo all'interno del supporto B-spline in x

- **Interpolatore:** Lineare, sia in fase iniziale che finale.

3.2 Registrazione

Le scansioni volumiche di CT e RM pervenute dal centro ospedaliero si presentano in formato DICOM, tipologia di dati in cui ai singoli frame vengono allegati i metadata contenenti le informazioni riguardanti modalità, risoluzione (slice thickness e pixel spacing) e orientamento spaziale, componenti essenziali per i software prima descritti affinché eseguano una corretta renderizzazione e registrazione volumica.

A causa però del massivo attacco hacker avvenuto ai sistemi informatici dell'ASL di Torino nell'agosto 2022, oltre a limitare il quantitativo di pazienti ricevuti, ha comportato che il prelievo avvenisse su database in cui buona parte dei dati risultavano privi delle informazioni prima citate e già pre-processati.

Per ovviare a questa problematica, la fase di registrazione è stata suddivisa in due linee metodologiche differenti.

I volumi completi dei metadata sono stati inviati al Tool di registrazione "Elastix" tramite "Slicer 3D", dove son stati definiti come volumi fissi le MRI e come volumi in movimento le CT.

Per questa tipologia di dati, il preset utilizzato all'interno del file di parametrizzazione del Tool, oltre a quelli citati al capitolo 3.1 Elastix, prevede l'impostazione dei seguenti valori rispetto a quelli di default:

- Number of Resolution: 1
- Maximum Number of Iterations: 400
- Number of spatial samples: 4000

Il processo di validazione viene in seguito eseguito visivamente per ogni volume di output, i quali vengono esportati su piattaforma Matlab e abbinati tramite pairing (funzione "impair").

In Fig.4 viene riassunto un esempio di risultato per il procedimento descritto.

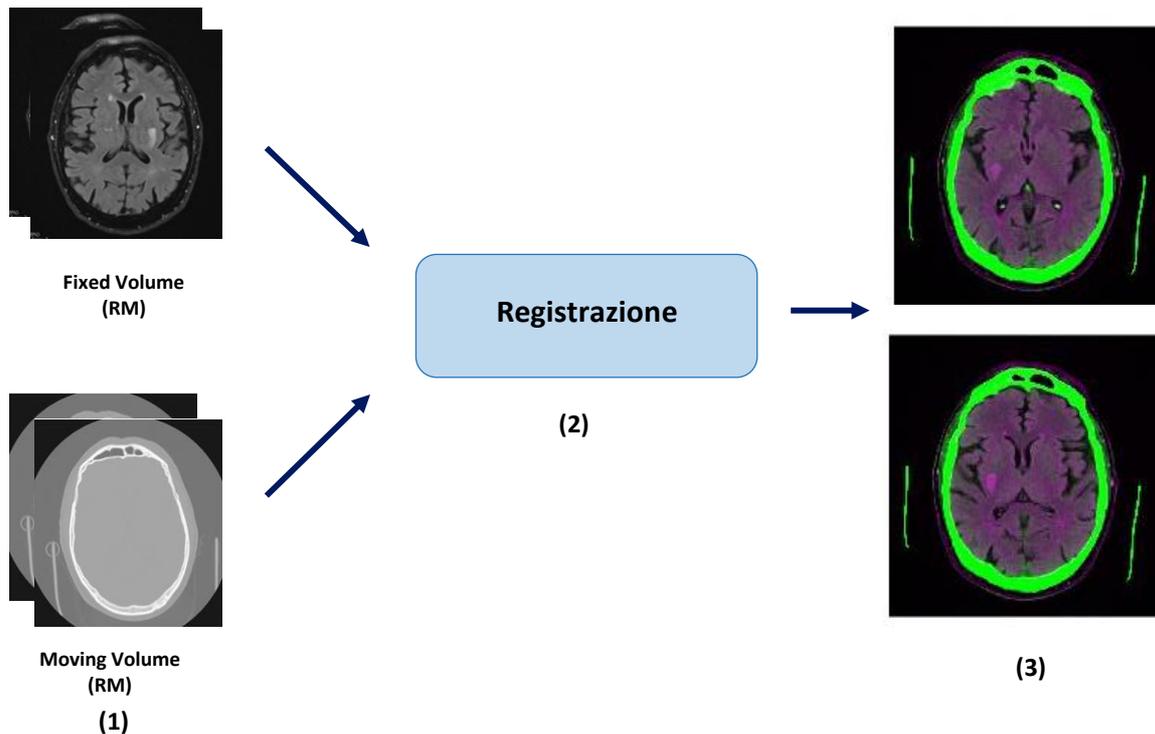


Figura 5: Processo di registrazione e validazione. (1) Identificazione del volume fisso (RM) e in movimento (TC). (2) Fase di registrazione mediante "Elastix". (3) Controllo visivo mediante pairing dei volumi risultanti su piattaforma Matlab.

Per quanto riguarda invece i volumi a disposizione che risultavano essere privi dei metadati è stata effettuata la seguente approssimazione. Siccome tutti gli esami diagnostici sono stati eseguiti all'interno della stessa clinica, e quindi con gli stessi macchinari, l'errore che si commette andando ad utilizzare i metadati dei pazienti completi risulta essere trascurabile per questa fase.

Con questo assunto si è allora proceduti a prelevare le sole informazioni delle modalità di interesse (TC Std. 2.5, RM FLAIR e RM DWI) e sovrascriverle a tutti i pazienti che ne erano privi. In questo modo si è riusciti nuovamente ad automatizzare il processo di registrazione parametrica mediante i software prima citati.

La seconda problematica riscontrata riguarda il fatto che i volumi in esame fossero già pre-processati e convertiti su 8 bit (255 livelli), invece che essere allo stato originale dove le intensità dei pixel vengono espressi in HU (Hounsfield Units) su 16 bit.

In Fig.5 è mostrata la differenza tra i due casi in analisi. Come si può vedere, gli effetti del taglio delle HU e la successiva normalizzazione che si ha tra (1) e (2) comporta oltre all'emergere delle informazioni cerebrali, anche ad una completa saturazione della calotta cranica con relativo inspessimento rispetto all'originale.

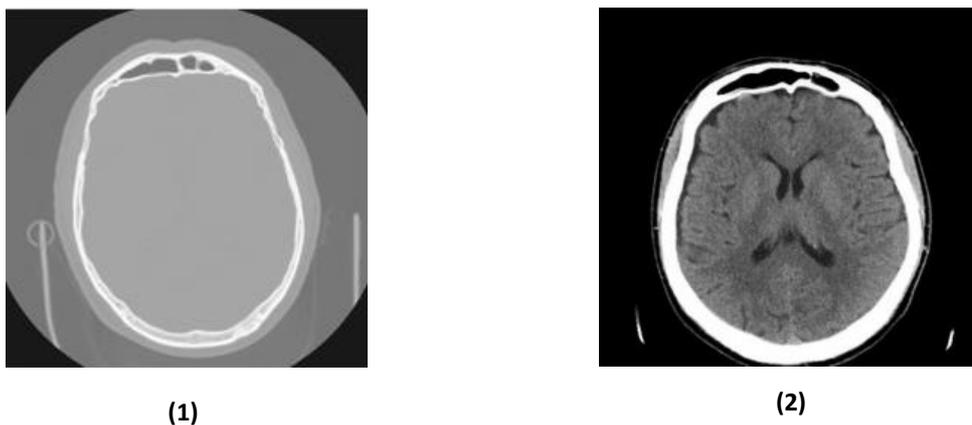


Figura 6. Esempio frame TC non processato (1) e frame processato e convertito (2).

Questa dinamica è fonte di disturbo per l'algoritmo di ottimizzazione per la registrazione, provocando deformazioni eccessive sul volume in uscita, non soddisfacendo dunque il requisito di corrispondenza spaziale tra i frame nelle due modalità.

Per cercare di limitare il più possibile questa problematica si è provveduti ad effettuare una prima registrazione tra i volumi RM e TC scorporati delle regioni saturate (i pixel bianchi vengono convertiti in nero), e, una volta verificata la buona corrispondenza tra il volume in uscita con i frame RM, si effettua una seconda trasformazione tra il risultato intermedio posto come fisso e volume originale posto come mobile.

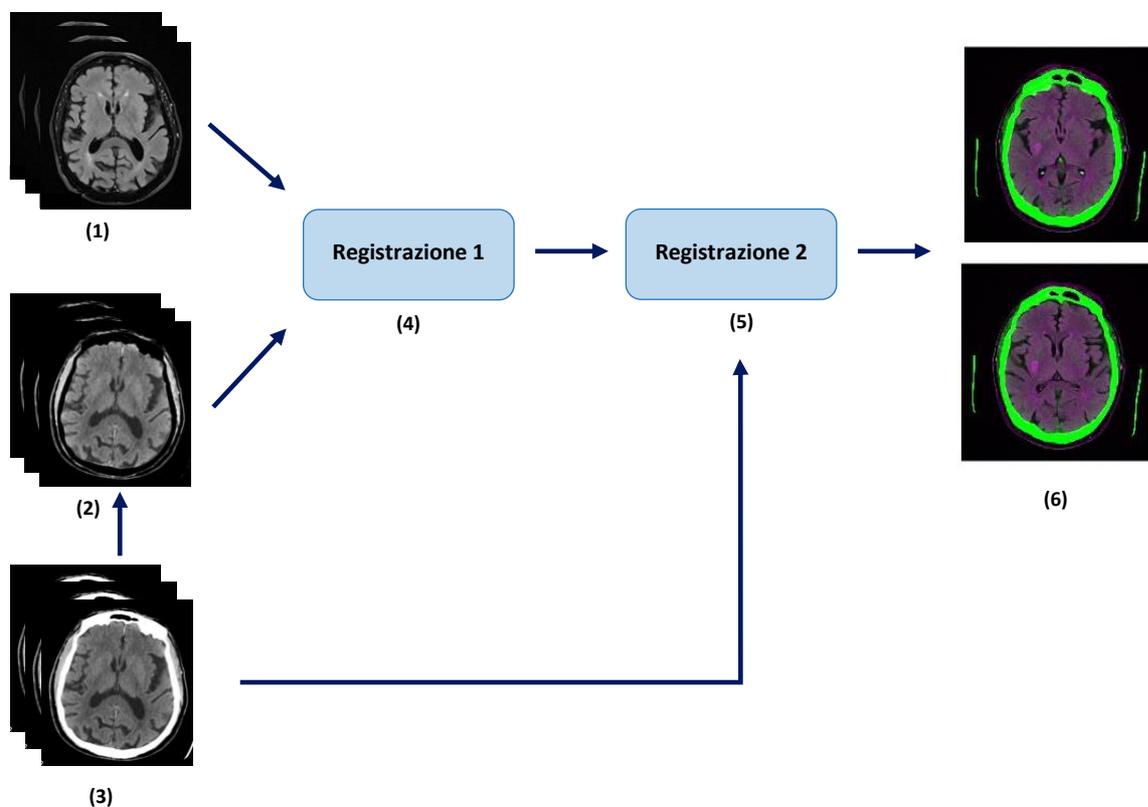


Figura 7. Schema a blocchi su tecnica di registrazione dati processati. (1) Volume fisso (RM), (2) volume mobile CT processato da (3), (4) prima registrazione, (5) seconda registrazione tra volume output uscente da (4) posto come fisso e (3) come mobile, (6) pairing di validazione.

3.3 Pre-Processing

Terminata la fase di registrazione, i volumi ottenuti vengono esportati da “Slicer 3D” ed inviati alla fase di pre-processing in ambiente Python.

Al fine di avere un dataset finale più omogeneo possibile, in questa fase si è proceduti ad uniformare i dati che si presentavano in formato originale con quelli che risultavano essere già processati e normalizzati.

Nello specifico, in seguito all’osservazione delle informazioni contenute all’interno dei file DICOM, è stato dedotto che per quanto riguarda i volumi TC il processamento automatico avvenga tramite taglio delle unità Hounsfield (HU) in funzione dei parametri *WindowCenter* e *WindowWidth*, pari rispettivamente a 40 e 100, e successiva normalizzazione su 8 bit.

Per questo motivo si è proceduti ad effettuare un taglio tra [-10,90] HU e successiva normalizzazione mediante min-max scaling (eq.9).

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \cdot 255 \quad (9)$$

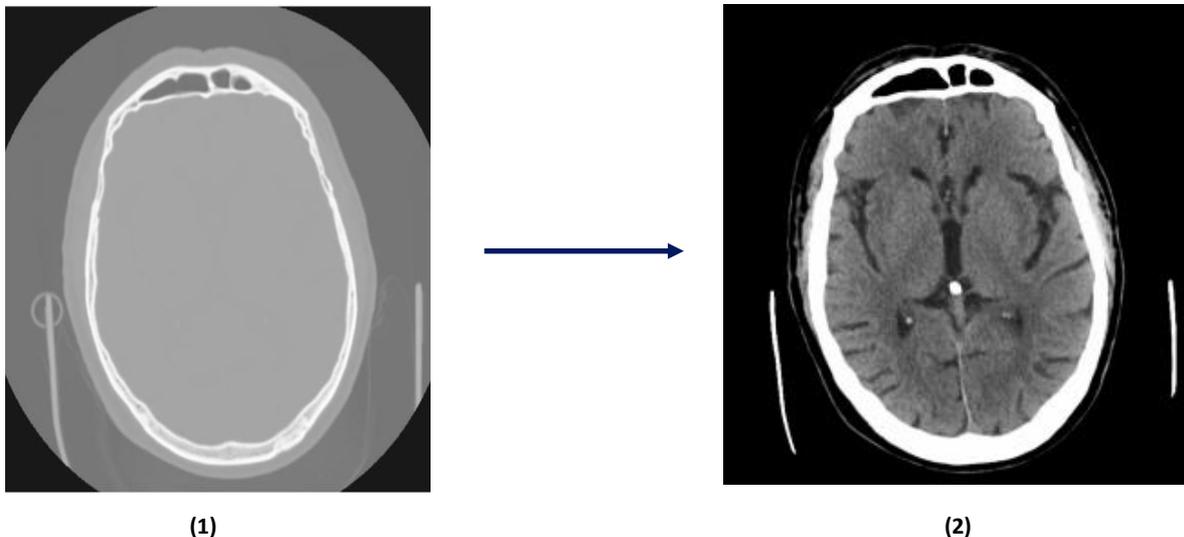


Figura 8. Pre-Processing TC da (1) a (2).

Analogamente alle TC, anche per i volumi di risonanza magnetica in modalità FLAIR si è proceduti ad effettuare un taglio tra [-10,550] HU e normalizzazione (Fig.8).

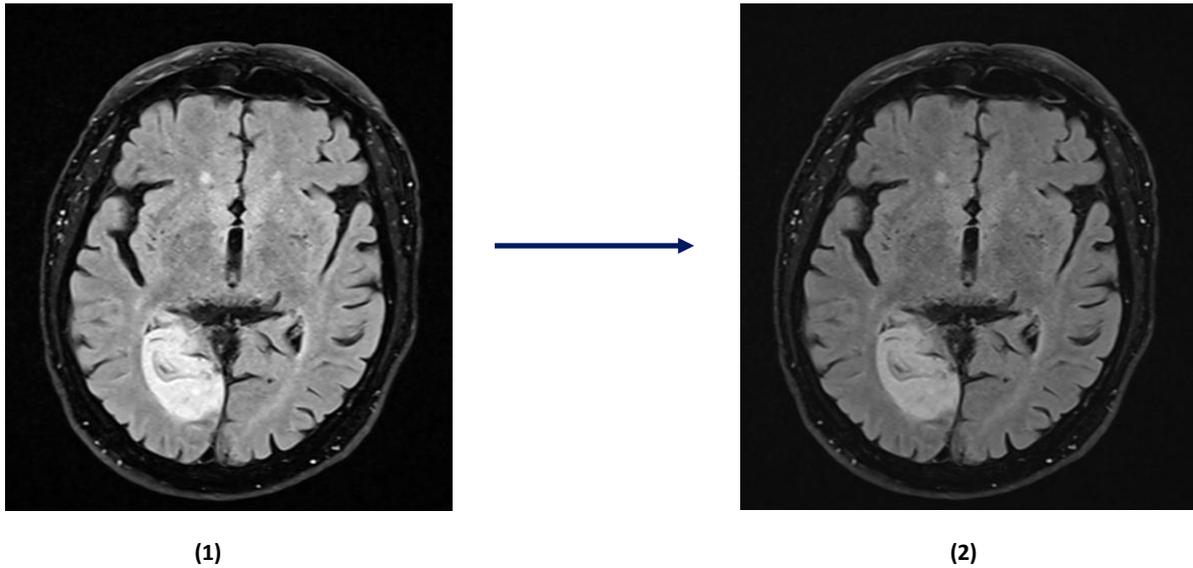


Figura 9. Pre-Processing RM FLAIR da (1) a (2).

Per quanto riguarda invece i volumi RM in modalità DWI, essi risultavano essere già tutti processati all'interno della clinica, e per questo motivo son stati mantenuti inalterati.

Infine, è stato effettuato un ridimensionamento di tutti i volumi CT e MR FLAIR a 256x256 pixels, mentre tra CT e MR DWI a 128x128 pixels, il tutto mediante la funzione *resize* della libreria open source "Skimage".

3.4 Maschere lesione

Una componente importante al fine di avere una maggiore comprensione sulle prestazioni ottenute dalle reti, sia in fase d'allenamento che di test, è il calcolo delle metriche anche per quanto riguarda le aree in cui si presentano le lesioni.

Per far questo sono state create manualmente le maschere binarie per ogni frame di ogni paziente in modo che tali aree vengano messe in evidenza, agevolando il processamento automatico durante la fase di calcolo dei risultati.

Come detto nella fase introduttiva, una diagnosi positiva ad ictus cerebrale si manifesta visivamente sulle immagini in RM come regione iper-intensa rispetto alle aree circostanti. L'ottenimento delle maschere è stato quindi ottenuto applicando un global-thresholding con soglia impostata e verificata manualmente caso per caso, selezionando poi successivamente solo le regioni d'interesse.

Il tutto è stato eseguito in ambiente Python per mezzo delle funzioni *"findContours"* e *"drawContours"* della libreria OpenCV [12].

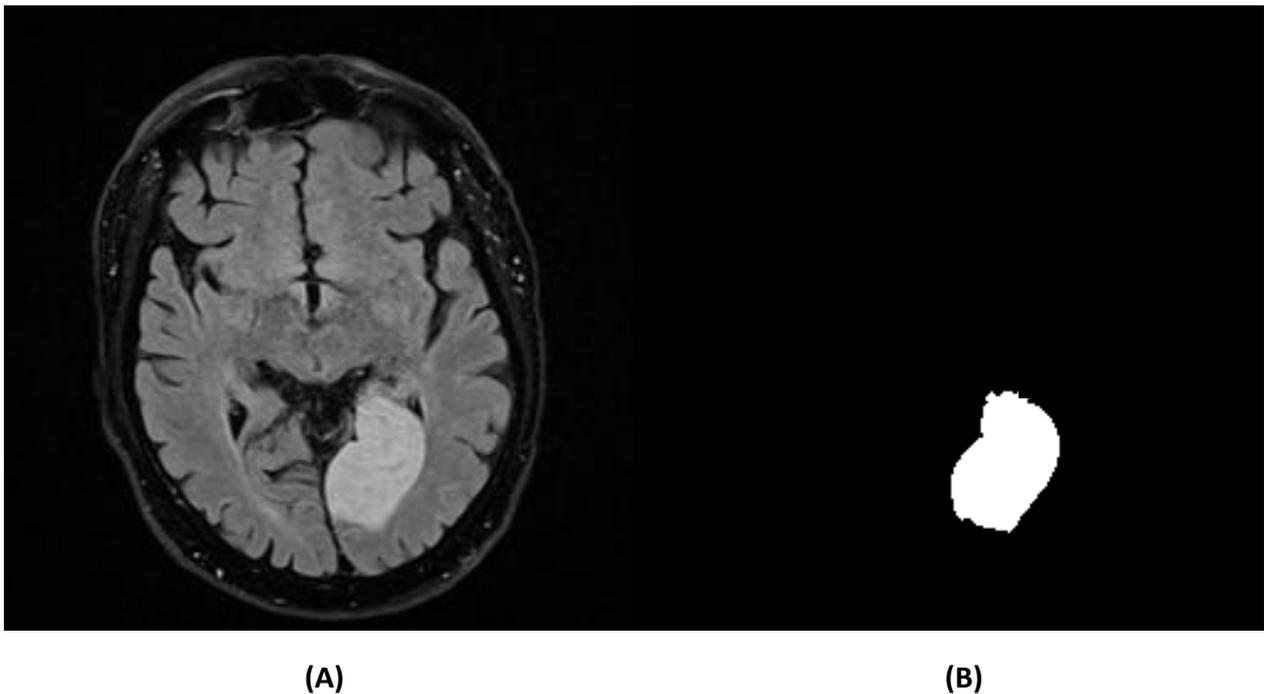


Figura 10 Maschera Binaria: (A) Frame in RM positivo ad ictus, (B) maschera binaria ottenuta mediante thresholding.

3.5 Data Augmentation

L'ultima fase della preparazione dei dataset da utilizzare in fase d'allenamento ha come obiettivo il creare una consistenza adeguata in termini di numerosità e, parallelamente, un buon bilanciamento tra il quantitativo di frame che presentano regioni iper-intense in RM, sintomo di positività ad ictus, e tra quelli che ne son privi.

Per far questo in ogni volume son stati scartati manualmente e randomicamente frame che non presentavano regioni infartuate tale da ottenere un rapporto di circa 1:2 tra positivi e negativi.

Infine, per tornare ad avere una buona consistenza in termini di numerosità, è stato applicato un data augmentation sull'intero dataset.

Il data augmentation è una tecnica utilizzata in deep learning per aumentare la quantità di dati disponibili per la formazione di un modello. L'importanza di tale procedura è dovuta al fatto che un maggior numero di dati può aiutare a migliorare l'accuratezza e la capacità di generalizzazione del modello. Esso consiste nella modifica delle immagini esistenti, ad esempio ruotandole, inclinandole o ingrandendole, generando così nuove immagini utilizzabili per la formazione del modello. Questo processo aumenta la diversità dei dati disponibili e migliora la robustezza del modello nei confronti di variazioni nei dati di input.

Nel caso in esame è stata utilizzata la libreria Python "*Albumentation*" [11], grazie alla quale sono state eseguite le seguenti trasformazioni:

- Trasformazione 1: Flip orizzontale e rotazione randomica [-10°,10°]
- Trasformazione 2: Distorsione di griglia e rotazione randomica [-10°,10°]

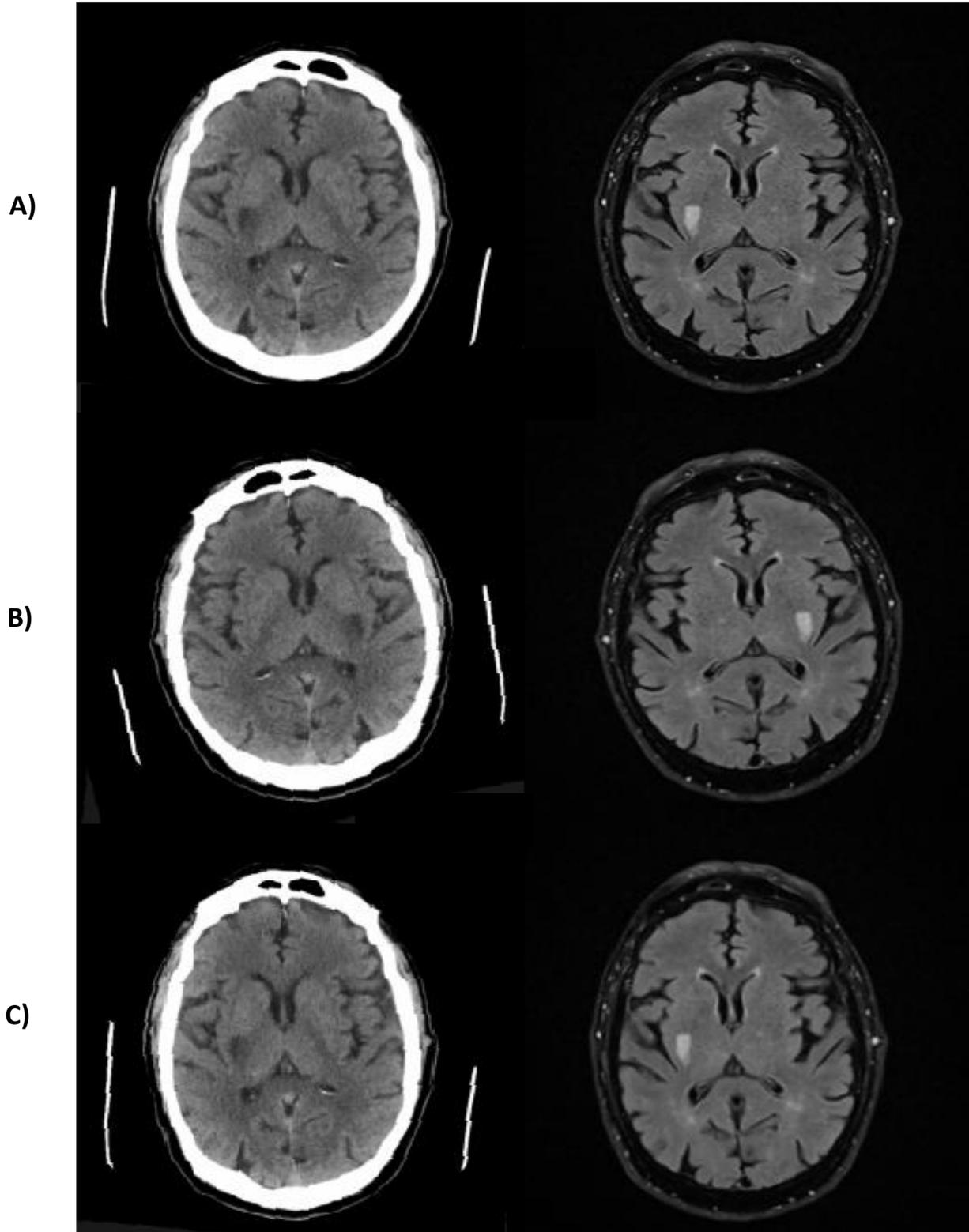


Figura 11. Data Augmentation. A) Frame accoppiati TC e RM FLAIR originali, B) Trasformazione 1 (Flip orizzontale e rotazione random); C) Trasformazione 2 (distorsione griglia e rotazione random).

4 Modelli Generativi

I modelli generativi sono una classe di algoritmi che vengono utilizzati nell'apprendimento automatico per generare immagini, testo e altri contenuti. Essi sono progettati per generare dati simili a quelli reali, che possono essere utilizzati per la creazione di contenuti di qualità.

Nel seguente progetto di tesi verrà verificato il loro utilizzo in ambito medicale per effettuare *domain transfer*, ovvero la traduzione di immagini appartenenti al dominio di partenza, in questo caso TC, nel corrispondente dominio d'arrivo in RM.

Nello specifico sono state confrontate le prestazioni di tre architetture differenti largamente utilizzate in questo momento in letteratura, ovvero le cGAN di tipologia "*pix2pix*" e "*cycleGAN*" e i recenti modelli generativi "*Diffusion Models*".

4.1 cGAN: pix2pix

Le cGAN, acronimo di Conditional Generative Adversarial Network [13], sono un tipo di rete neurale artificiale data dalla combinazione di due modelli: un generatore e un discriminatore.

La tipologia d'allenamento che si viene a definire è di tipo competitivo, in cui il generatore è addestrato a generare output che non possano essere distinti dalle immagini "reali", mentre il discriminatore cerca di migliorare la sua capacità di distinguere i "falsi" del primo.

Le pix2pix [14] (Fig. 11) utilizzate nel seguente progetto di tesi sono una tipologia di cGAN che permettono di effettuare *image-to-image translation*.

Rispetto alle GAN classiche, in cui il generatore apprende la mappatura a partire da un vettore di rumore casuale z verso l'immagine di output y , $G: z \rightarrow y$, nelle pix2pix il vettore rumore è abbinato all'immagine osservata x , $G: \{x, z\} \rightarrow y$.

La funzione obiettivo di una cGAN può essere espressa come:

$$\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_{x,z}[\log(1 - D(x, G(x, z)))] \quad (10),$$

dove G cerca di minimizzare l'obiettivo contro l'avversario D che cerca di massimizzarlo, $G^* = \arg \min_G \max_D \mathcal{L}_{cGAN}(G, D)$.

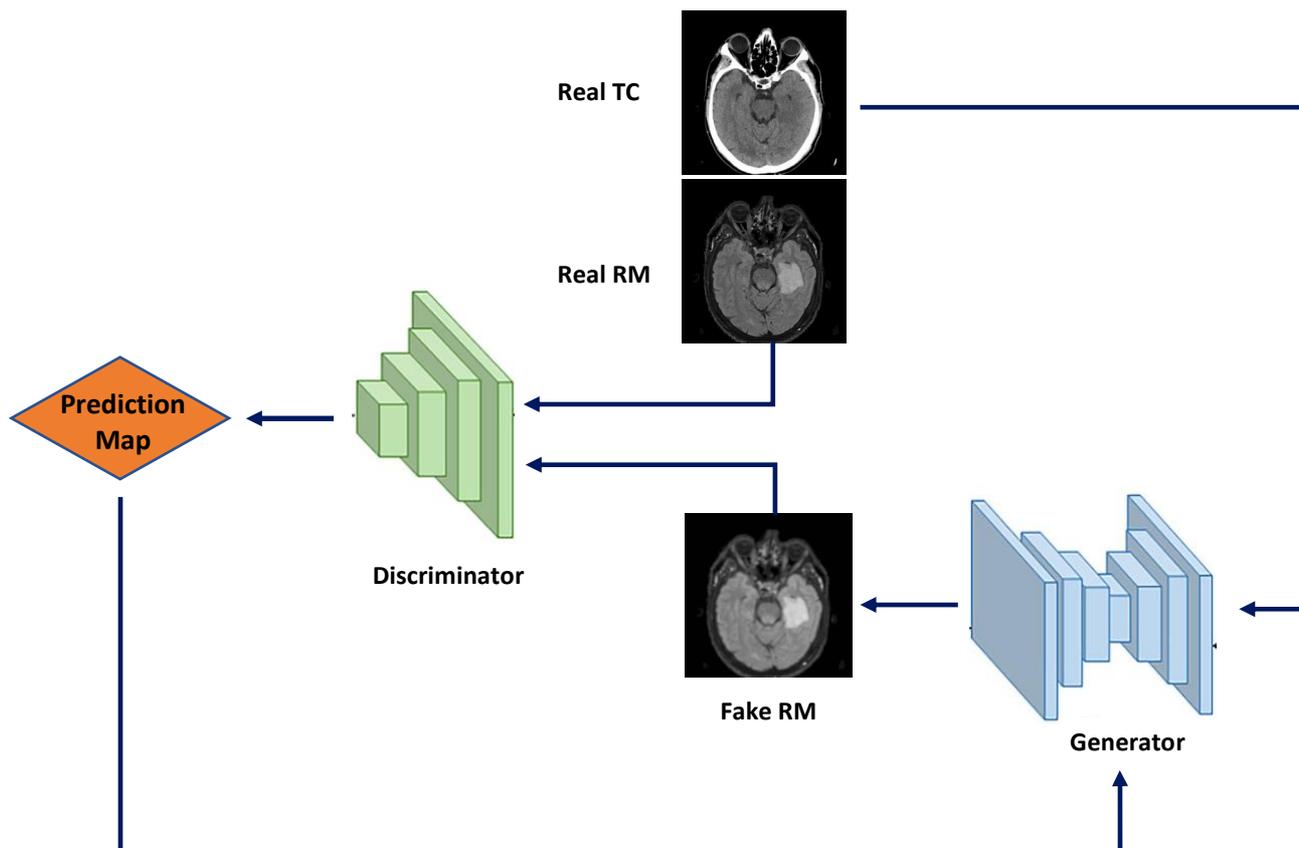


Figura 12. Schema a blocchi Pix2Pix

L'architettura del generatore Pix2Pix utilizza una rete neurale convoluzionale (CNN) denominata U-Net.

Essa è costituita da due parti principali: l'encoder e il decoder. L'encoder è costituito da una serie di strati di convoluzione che estraggono le caratteristiche dell'immagine di input. Il decoder, invece, utilizza una serie di strati di deconvoluzione per ricostruire l'immagine di output.

La caratteristica distintiva di U-Net è la presenza di skip connections, che collegano l'encoder e il decoder in modo da consentire un flusso di informazioni più rapido ed efficiente tra i due. In particolare, le skip connections connettono gli strati dell'encoder a quelli del decoder, consentendo al decoder di utilizzare informazioni dall'encoder per aiutare a generare l'immagine di output.

Le skip connections [15] sono particolarmente utili per affrontare il problema dell'immagine a bassa risoluzione, che è comune nelle traduzioni di immagini. Poiché l'immagine di input è compressa a una bassa risoluzione nell'encoder, le informazioni sono perse e l'immagine di output potrebbe diventare sfocata o poco dettagliata. Tuttavia, le skip connections consentono al decoder di utilizzare le informazioni a una risoluzione più elevata dall'encoder, consentendo un'immagine di output di maggiore qualità.

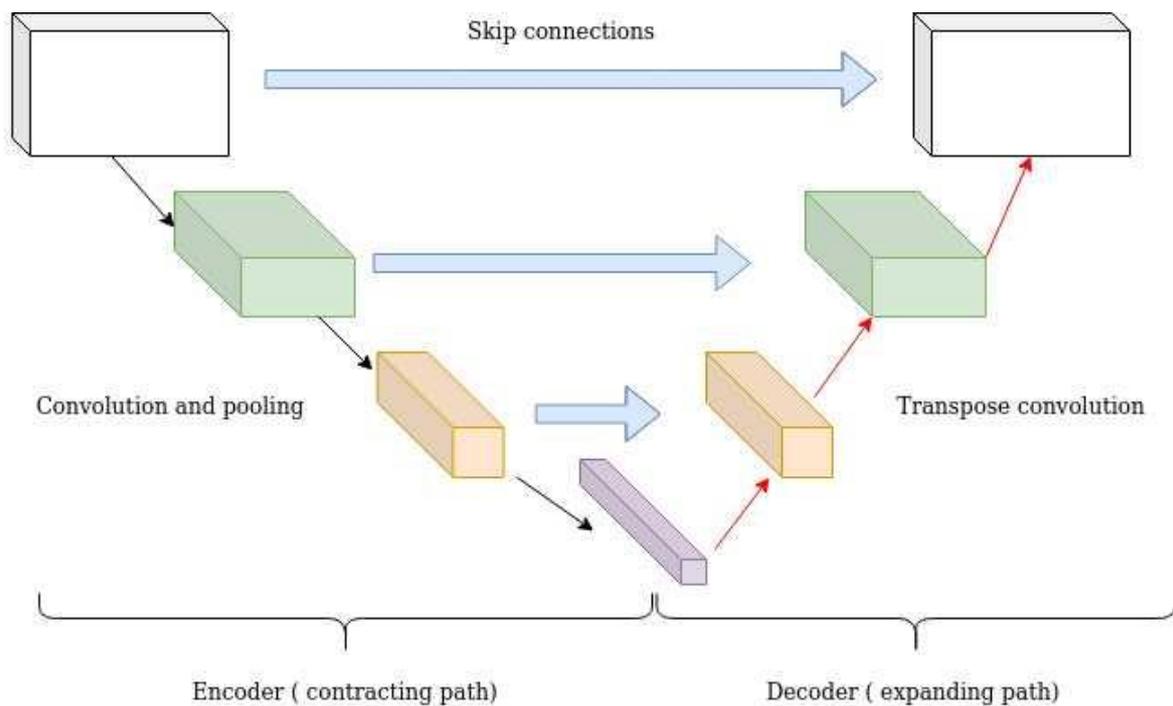


Figura 13. Schema a blocchi Unet con Skip Connections [15]

Per quanto riguarda l'architettura del discriminatore, i modelli pix2pix utilizzano la PatchGAN [16].

Essa è costituita da una rete neurale convoluzionale (CNN) che prende in input un'immagine e analizza piccole regioni di dimensione $N \times N$ in modo da poter predire per ogni area se appartenga al dominio reale o generato.

Per far questo, la rete utilizza una serie di strati di convoluzione per estrarre le caratteristiche dell'immagine in input e successivamente una serie di strati completamente connessi per produrre la mappa di probabilità. Il risultato della rete Patch-GAN è un numero tra 0 e 1 che indica la probabilità di appartenenza.

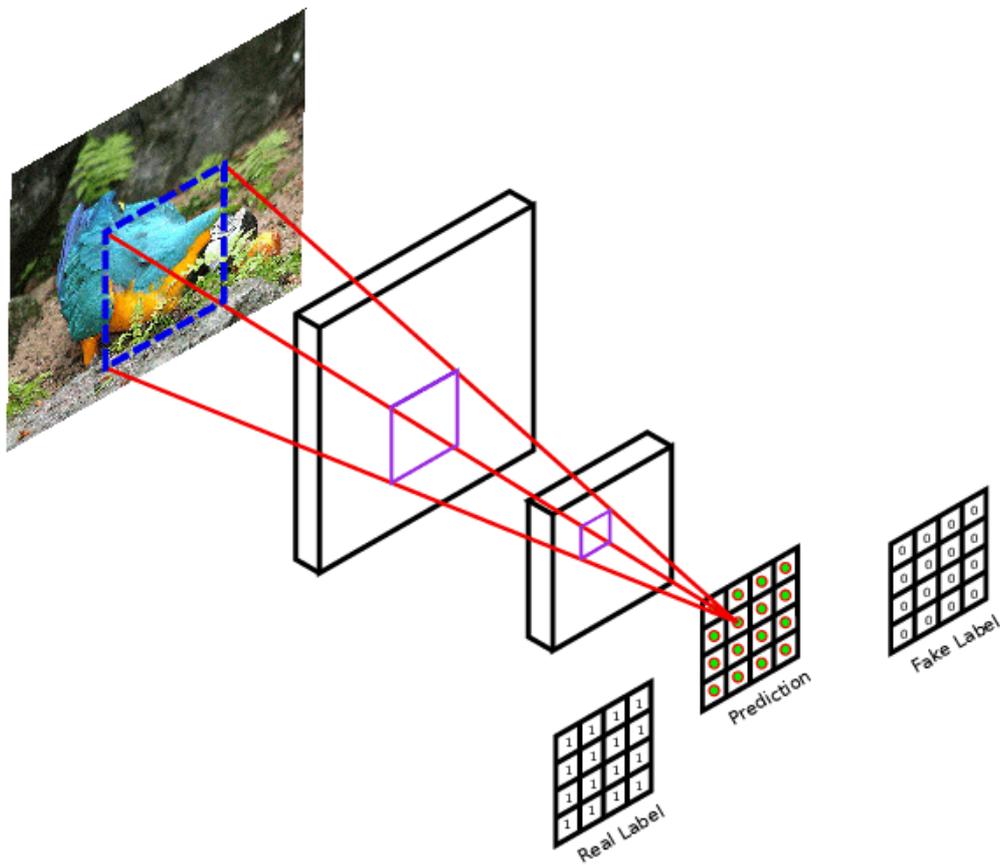


Figura 14. Patch Gan [11]

4.2 GAN: cycleGAN

Rispetto alle cGAN classiche e, di conseguenza, alle Pix2Pix prima citate, le cycleGAN [17] raddoppiano il processo di predizione forzando una trasformazione inversa, al fine di aumentare l'accuratezza delle immagini di output.

L'architettura del modello prevede infatti la presenza di due generatori e due discriminatori che lavorano simmetricamente, ognuno per ciascun dominio.

Nello specifico, la fase di addestramento consiste nell'apprendimento da parte della rete a mappare le immagini del dominio X in quelle del dominio Y e viceversa, cercando di minimizzare la divergenza tra le distribuzioni delle immagini reali e delle immagini generate.

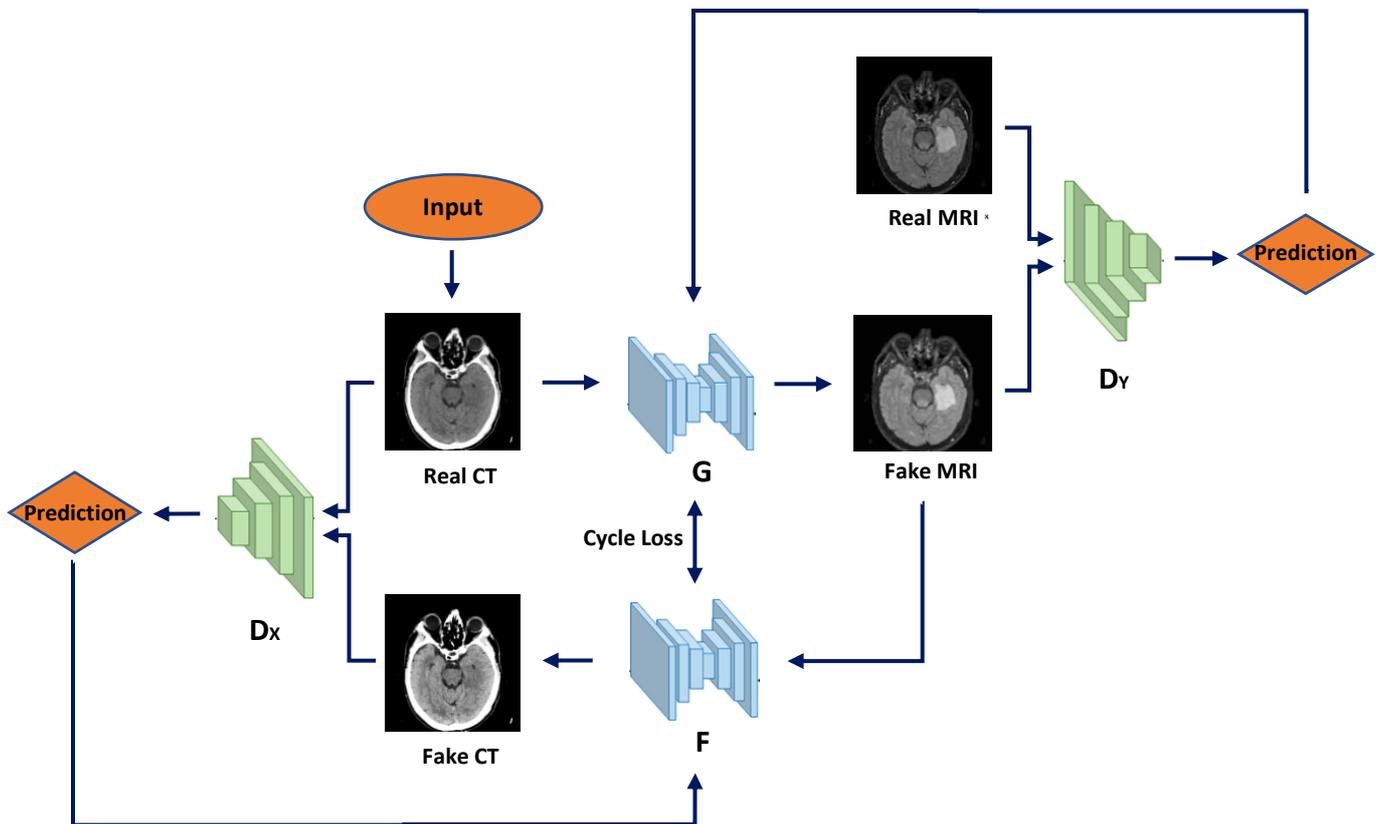


Figura 15. Schema a blocchi CycleGAN

Come mostrato in Fig. 14, l'architettura della CycleGAN proposta si compone di due generatori, $G : X \rightarrow Y$ e $F : Y \rightarrow X$, ai quali sono abbinati due discriminatori avversari di tipologia patchGAN D_y e D_x .

La funzione obiettivo è data dalla combinazione di due termini: perdite contraddittorie \mathcal{L}_{cGAN} (eq. 10), per abbinare la distribuzione delle immagini generate alla distribuzione dei dati nel dominio di destinazione, e la *Cycle Consistency Loss* \mathcal{L}_{cyc} per evitare che le mappature apprese da G e F si contraddicano a vicenda.

La funzione obiettivo completa viene così definita:

$$\mathcal{L}(G, F, D_x, D_y) = \mathcal{L}_{cGAN}(G, D_y, X, Y) + \mathcal{L}_{cGAN}(F, D_x, Y, X) + \lambda \mathcal{L}_{cyc}(G, F) \quad (11)$$

dove λ è il coefficiente di controllo dei due obiettivi.
Compito dell'allenamento è dunque la risoluzione di:

$$G^*, F^* = \arg \min_{G, F} \max_{D_x, D_y} \mathcal{L}(G, F, D_x, D_y) \quad (12)$$

4.3 Diffusion Models

I Diffusion Models [18] sono modelli generativi di recente introduzione, i quali differiscono dalle cGAN classiche per il fatto che il processo di apprendimento nella generazione di immagini sintetiche avviene secondo un processo di *noising-denoising* dell'input.

Nello specifico viene definita una catena di Markov [19] all'interno della quale nella prima fase avviene l'aggiunta progressiva di rumore Gaussiano ai dati, in questo caso immagini, in un numero definito di time-step T.

Come mostrato in Fig. 15, data una variabile x_0 campionata a partire da una distribuzione reale $q(x)$, per ogni step della catena di Markov vi è l'aggiunta di rumore con varianza β_t tale per cui da x_{t-1} si produce una nuova variabile latente x_t con distribuzione $q(x_t|x_{t-1})$.

Il seguente processo può essere formulato nel seguente modo:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \mu_t = \sqrt{1 - \beta_t}x_{t-1}, \Sigma_t = \beta_t I) \quad (13)$$

dove il processo di diffusione è definito dalla media μ_t e dalla varianza Σ , ed I è la matrice identità, la quale sta ad indicare che ogni dimensione ha la stessa deviazione standard β_t .

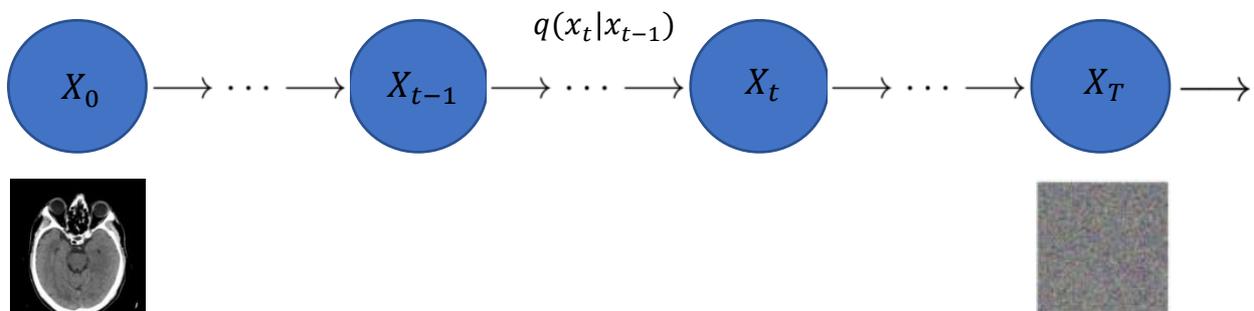


Figura 16. Noising Process

Una volta raggiunto lo step T, la variabile latente x_T è vicina ad essere una distribuzione isotropica Gaussiana. Da questo momento il modello cerca di apprendere nella la fase di training il processo inverso, ovvero come effettuare denoising fino a fornire in uscita l'output desiderato (Fig. 16).

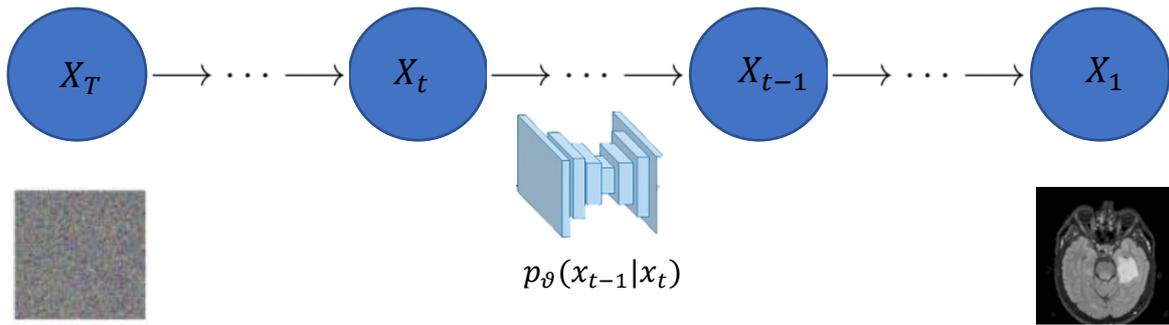


Figura 17. Denoising process

Ciò consiste nel trovare la transizione inversa della catena di Markov tale per cui venga massimizzata la probabilità dei dati d'allenamento.

Per ogni time-step viene quindi definito un modello U-Net in grado di effettuare il processo di denoising, dove l'allenamento consiste nell'apprendere i parametri di transizione Gaussiana dipendenti dal tempo.

Matematicamente si si traduce nell'ottimizzazione della seguente funzione obiettivo:

$$\mathbb{E}_{(x,y)} \mathbb{E}_{(\epsilon,\gamma)} \|f_\theta(x, \sqrt{\gamma}y_0 + \sqrt{1-\gamma}\epsilon, \gamma) - \epsilon\| \quad (14)$$

dove $f_\theta(x, \tilde{y}, \gamma)$ rappresenta la parametrizzazione del modello neurale al quale viene data in input la condizione x , l'immagine rumorosa \tilde{y} e il livello di rumore corrente γ .

5 Metriche di valutazione

Le prestazioni di ogni modello allenato vengono analizzate mediante lo studio dell'evoluzione delle principali metriche qualitative calcolate tra le immagini generate e quelle di riferimento, sia sul training che sul test.

Nello specifico la bontà delle traduzioni è stata valutata mediante l'utilizzo dei coefficienti RMSE, PSNR e SSIM.

L'RMSE, o errore quadratico medio, è una misura assoluta in cui le deviazioni tra output e riferimento vengono elevate al quadrato, in questo modo si evita che segni opposti possano annullarsi e vengono messi in risalto gli errori maggiori. Il tutto viene infine posto sotto radice (eq. 14).

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(\hat{y}_i - y_i)^2}{n}} \quad (15)$$

Il PSNR, *Peak to Signal-to-Noise Ratio*, è una misura che viene adottata per valutare la qualità di un'immagine rispetto all'originale. Esso viene definito come rapporto tra il massimo valore di intensità del pixel dell'immagine rispetto al rumore (rappresentato come RMSE) e viene espresso in scala logaritmica decibel (eq.15). Maggiore è il suo valore, più la somiglianza a livello visivo tra le due è alta.

$$PSNR = 20 \cdot \log_{10} \left(\frac{MAX(I)}{\sqrt{MSE}} \right) \quad (16)$$

Infine, il SSIM rappresenta l'indice di similarità strutturale, il quale fornisce una predizione sulla somiglianza di due immagini fornendo un valore compreso tra [0, 1].

Rispetto alle altre metriche presentate, il SSIM non misura l'errore tra l'immagine di output e il riferimento, ma la somiglianza delle informazioni spaziali calcolate come interdipendenze tra pixel vicini.

Esso matematicamente è definito come:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (17)$$

dove μ_x e μ_y rappresentano la media dei campioni di x e y, σ_x^2 e σ_y^2 la varianza, σ_{xy} la covarianza tra x e y.

c_1 e c_2 invece rappresentano due variabili che hanno la funzione di stabilizzare il denominatore affinché non risulti troppo debole e si definiscono come:

$$c_1 = (k_1L)^2; c_2 = (k_2L)^2 \quad (18)$$

Dove L rappresenta la gamma dinamica dei valori dei pixel ($2^n - 1$), mentre k_1 e k_2 sono due costanti che di default vengono fissate rispettivamente pari a 0.01 e 0.03.

Siccome però, oltre alla valutazione della qualità generale di traduzione dell'immagine, lo scopo del progetto è di verificare la capacità dei modelli generativi allenati nel rilevare la lesione, son state prese in esame l'evoluzione di altre due metriche calcolate sia all'interno dell'area delineata dalla maschera binaria che nello spazio circostante escluso lo sfondo, in modo da valutare eventuali scostamenti di prestazioni tra le due e individuare il migliore.

Nello specifico son stati utilizzati l'errore medio (ME) e l'errore medio assoluto (MAE).

$$ME = \sum_{i=1}^n \frac{\hat{y}_i - y_i}{n} \quad (19)$$

$$MAE = \sum_{i=1}^n \frac{abs(\hat{y}_i - y_i)}{n} \quad (20)$$

6 Dettagli d'implementazione

Nel seguente capitolo verranno esplicitati i dettagli di implementazione delle architetture utilizzate per la fase di training e test con le relative parametrizzazioni, al fine di giudicare nella fase di valutazione dei risultati quale modello dia le migliori prestazioni per qualità di traduzione e rilevamento lesione.

Tutti i modelli son stati allenati in ambiente Python (v 3.8.5) mediante il framework di apprendimento automatico "Pytorch" della libreria Torch e su Workstation dotata di GPU integrata "GeForce RTX 3080 12 GB".

- Pix2Pix: per questa architettura son stati utilizzati i codici open-source proposti da [20], impostando batch size pari a 1, generatore "UNET_256" e "UNET_128" rispettivamente per MRI FLAIR e MRI DWI, discriminatore PatchGAN "basic" e allenate per 150 epoche con salvataggio dei checkpoint predisposto ogni 5.

Le prove eseguite per questa modalità si sono concentrate nel trovare la giusta parametrizzazione del numero di filtri di generatore e discriminatore. Essi, infatti, di default vengono posti pari a 9, ma siccome il numero di layer tra i due blocchi risultano essere diversi date le diverse architetture, il numero di parametri complessivo che si ottiene in funzione degli input in ingresso sarà di conseguenza sbilanciato.

Per questo motivo sono state eseguite una serie di prove che tenessero conto di questo fattore e valutando le prestazioni tra modelli bilanciati e no.

Le parametrizzazioni testate vengono riassunte in Tab. 1.

ndf	GAN_mode	ngf
9	lsgan	9
9	vanilla	9
24	lsgan	24
24	vanilla	24
32	lsgan	32
32	vanilla	32
9	lsgan	40
9	vanilla	40
24	lsgan	96
24	vanilla	96
32	lsgann	128
32	vanilla	128
64	lsgan	12
64	vanilla	12

Tabella 2. Parametrizzazioni Pix2Pix, dove *ndf* è il numero di filtri per discriminatore, *ngf* del generatore e *GAN_mode* la tipologia di GAN utilizzata

- CycleGAN: oltre alle Pix2Pix, i codici proposti da [20] permettevano di selezionare come preset d'allenamento la modalità cycleGAN in abbinamento all'utilizzo di dataset accoppiati, in modo da ottenere in uscita un modello che effettui *domain transfer*.

Il batch size è stato impostato pari a 1, i generatori di ogni modello sono entrambi "UNET_256" o "UNET_128", discriminatori patchGAN "basic" e la fase di training è stata impostata per un totale di 150, salvando sempre i checkpoint ogni 5 epoche.

Condividendo di fatto la stessa architettura delle Pix2Pix, ma con l'aggiunta del processo inverso di traduzione, vengono anche per esse fatte le precedenti considerazioni in merito al numero di filtri da impostare, le cui parametrizzazioni vengono riassunte in Tab. 2.

ndf	GAN_mode	ngf
9	lsgan	9
9	vanilla	9
24	lsgan	24
24	vanilla	24
32	lsgan	32
32	vanilla	32
9	lsgan	40
9	vanilla	40
24	lsgan	96
24	vanilla	96
32	lsgann	128
32	vanilla	128
64	lsgan	12
64	vanilla	12

Tabella 3. Parametrizzazione cycleGAN, con *ndf* numero di filtri del discriminatore *ngf* del generatore e *GAN_mode* tipologia di GAN utilizzata

- Diffusion Models: per questa tipologia di modelli ci si è ispirati alla formulazione proposta da Saharia [21] impostata per effettuare super-resolution, ovvero allenamento di modelli in grado di ricevere in input immagini a bassa risoluzione generando la corrispondente ad alta risoluzione.

Nello specifico sono stati prelevati i codici open-source presenti in [22] adattandoli al nostro scopo.

Le prove riassunte in Tab. 3 sono state eseguite impostando l'allenamento su 100 epoche, modello Unet "ddpm" dotato di 2 blocchi "resnet" e 16 "Attention Residual" e ottimizzatore "Adam" con learning rate pari a $1e^{-4}$.

Batch size	Schedule
1	Cosine
1	Linear
2	Cosine
2	linear

Tabella 4. Parametrizzazione Diffusion Models

A causa dell'elevato costo computazionale che i seguenti modelli hanno rispetto alle altre architetture nella fase di *sampling*, l'inferenza per la valutazione delle prestazioni durante la fase di training è stata eseguita solo sulle immagini di test ogni 5 epoche.

7 Risultati

Al termine della fase di training delle reti neurali, le loro prestazioni sono state valutate andando a prelevare per ognuna l'epoca migliore in funzione delle metriche ottenute.

Nello specifico, grazie al salvataggio dei checkpoint eseguito ogni 5 epoche, è stato possibile eseguire l'inferenza di tutti i frame, sia training che di test, per ogni fase d'allenamento.

In questo modo sono stati calcolati i valori medi e deviazioni standard delle metriche narrate al capitolo 5 Metriche di valutazione, studiandone l'evoluzione e identificando l'epoca in cui si ottengono le migliori prestazioni.

Infine, una volta selezionati i migliori modelli per tipologia d'architettura, è stata effettuata la verifica visiva dei frame ottenuti al fine di valutare l'effettiva qualità di traduzione in termini di realismo, ma soprattutto per verificare se le zone con presenza di lesione sulle immagini di riferimento vengano o meno rilevate e messe in evidenza in fase di generazione.

A seguire verranno mostrati i risultati ottenuti differenziando i modelli allenati per effettuare traduzione TC-MRI (FLAIR) e TC-MRI (DWI).

7.1 Traduzione TC-MRI FLAIR

Pix2Pix:

In tabella 5 e 6 vengono riassunte le prestazioni ottenute nell'epoca migliore per ogni parametrizzazione impostata.

Nello specifico vengono riportati i valori medi di RMSE, PSNR e SSIM calcolati sui frame di test, e i valori medi di MAE e ME calcolati sulle aree in presenza di lesione.

In giallo viene infine evidenziato il modello scelto come migliore per la seguente tipologia di architettura.

ndf	GAN_mode	ngf	Best Epoch	RMSE	PSNR	SSIM	MAE	ME
9	lsgan	9	95	15.3	24.61	0.73	26.76	-23.7
9	vanilla	9	70	15.93	24.35	0.72	27.34	-24.4
24	lsgan	24	80	15.19	24.82	0.76	26.62	-23.4
24	vanilla	24	75	15.69	24.54	0.75	27.19	-25.5
32	lsgan	32	45	14.96	25.00	0.77	27.3	-24.1
32	vanilla	32	135	15.65	24.56	0.74	26.9	-24.3

Tabella 5. Risultati Pix2Pix TC-FLAIR, parametrizzazione sbilanciata

ndf	GAN_mode	ngf	Best Epoch	RMSE	PSNR	SSIM	MAE	ME
9	lsgan	40	75	14.82	25.05	0.77	26.53	-21.23
9	vanilla	40	25	16.84	23.88	0.73	27.43	-27.1
24	lsgan	96	55	15.07	24.91	0.78	26.89	-23.5
24	vanilla	96	95	16.52	24.07	0.74	28.5	-26.3
32	lsgann	128	45	15.25	24.79	0.76	26.64	-23.8
32	vanilla	128	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
64	lsgan	12	100	14.71	25.11	0.78	26.81	-23.9
64	vanilla	12	110	14.88	25.03	0.77	26.85	-24.1

Tabella 6. Risultati Pix2Pix TC-FLAIR, parametrizzazione bilanciata

L'evoluzione delle metriche durante la fase d'allenamento viene qui riassunta.

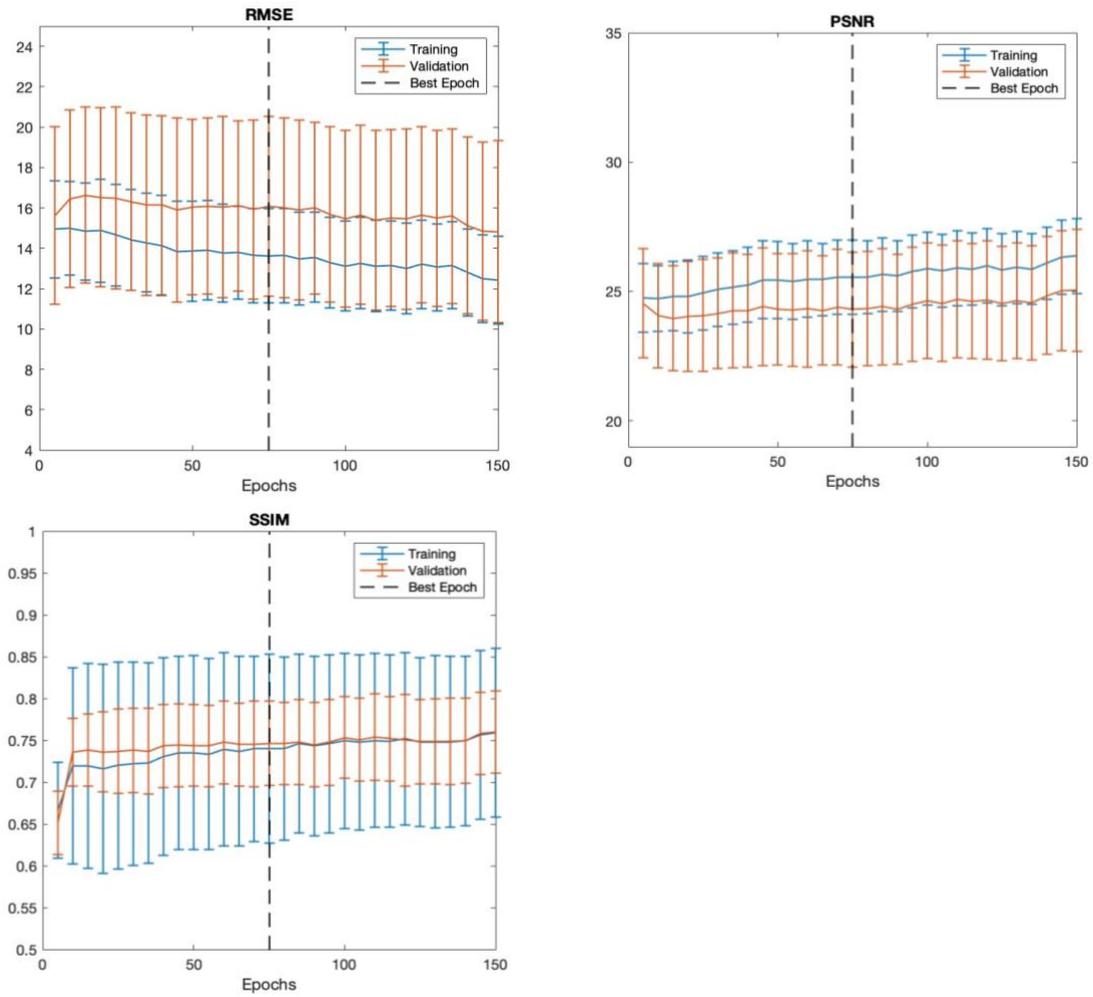


Figura 19. Pix2Pix TC-FLAIR: RMSE, PSNR e SSIM su training e Validation (valor medio \pm std)

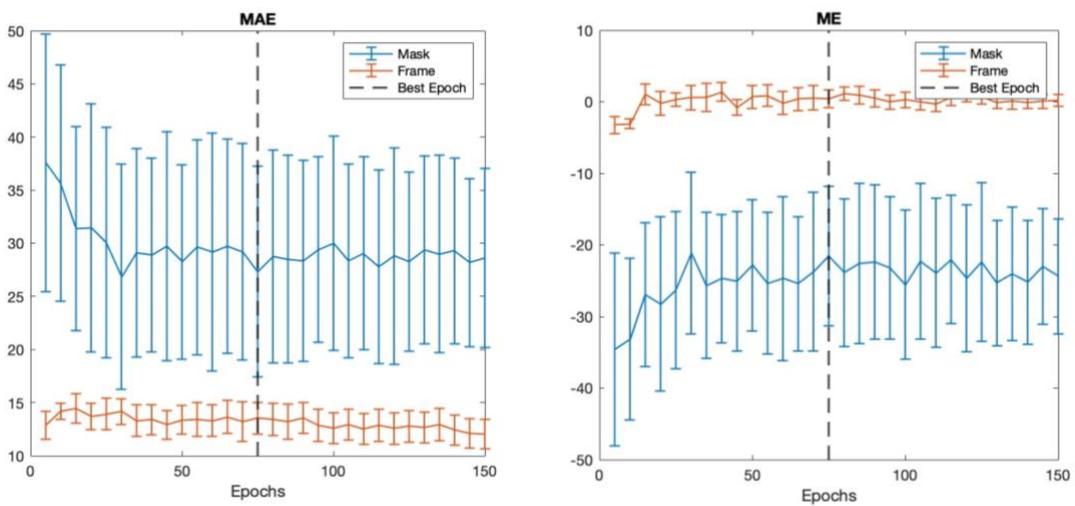


Figura 19. Pix2Pix TC-FLAIR: MAE e ME calcolati su maschera di lesione e frame (valor medio \pm std)

CycleGAN:

Allo stesso modo, in tabella 7 e 8 vengono riassunti i risultati ottenute per le epoche migliori sui frame di Test, evidenziando in giallo il modello vincente.

ndf	GAN_mode	ngf	Best Epoch	RMSE	PSNR	SSIM	MAE	ME
9	lsgan	9	120	17.66	23.45	0.72	36.32	-33.67
9	vanilla	9	145	19.44	22.57	0.68	38.23	-34.42
24	lsgan	24	105	16.34	24.17	0.74	34.92	-32.51
24	vanilla	24	130	16.67	24.02	0.73	35.19	-33.53
32	lsgan	32	115	16.30	24.21	0.73	34.54	-32.34
32	vanilla	32	85	16.43	24.12	0.74	34.93	-32.78

Tabella 7. Risultati CycleGAN TC-FLAIR, parametrizzazione sbilanciata

ndf	GAN_mode	ngf	Best Epoch	RMSE	PSNR	SSIM	MAE	ME
9	lsgan	40	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
9	vanilla	40	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
24	lsgan	96	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
24	vanilla	96	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
32	lsgann	128	75	17.13	23.77	0.73	36.15	-34.28
32	vanilla	128	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
64	lsgan	12	115	15.81	24.49	0.74	35.63	-33.25
64	vanilla	12	100	17.00	23.83	0.71	38.42	-35.13

Tabella 8. Risultati CycleGAN TC-FLAIR, parametrizzazione bilanciata

L'evoluzione delle metriche durante la fase d'allenamento viene qui riassunta.

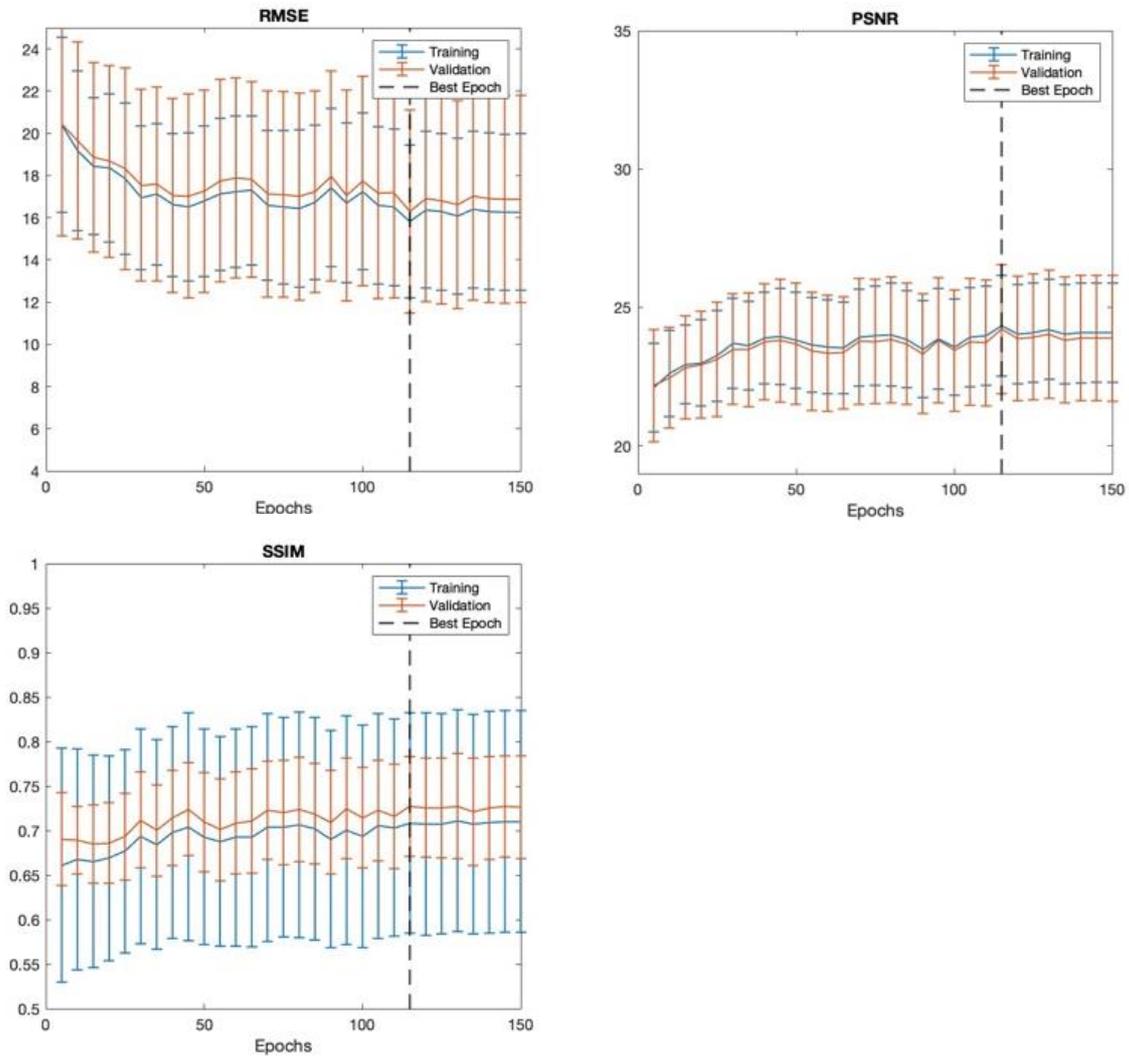


Figura 20. CycleGAN TC-FLAIR: RMSE, PSNR e SSIM su training e Validation (valor medio \pm std)

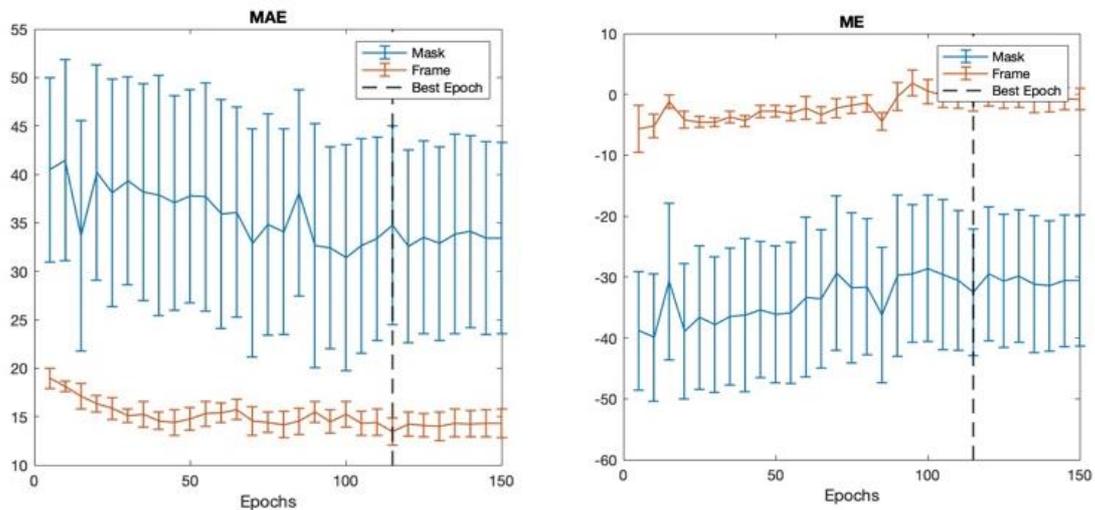


Figura 21. CycleGAN TC-FLAIR: MAE e ME calcolati su maschera di lesione e frame (valor medio \pm std)

Diffusion Models:

A causa dell'impossibilità di eseguire l'inferenza sui frame di training dovuti agli alti costi computazionali in fase di generazione, le metriche di valutazione sono state calcolate solo sul test, prelevando in seguito l'epoca migliore per ogni parametrizzazione e rapportandole tra loro.

In tabella 9 vengono riassunti i risultati mettendo in evidenza la vincente.

Batch size	Schedule	Best Epoch	RMSE	PSNR	SSIM	MAE	ME
1	Cosine	55	17.81	23.32	0.75	37.31	-33.21
1	Linear	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
2	Cosine	45	16.38	24.53	0.78	36.29	-32.15
2	linear	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.

Tabella 9. Risultati Diffusion Models TC FLAIR

In Fig. 22 vengono mostrati quattro esempi significativi di generazione su frame di test, paragonando le tre architetture migliori rispetto al riferimento.

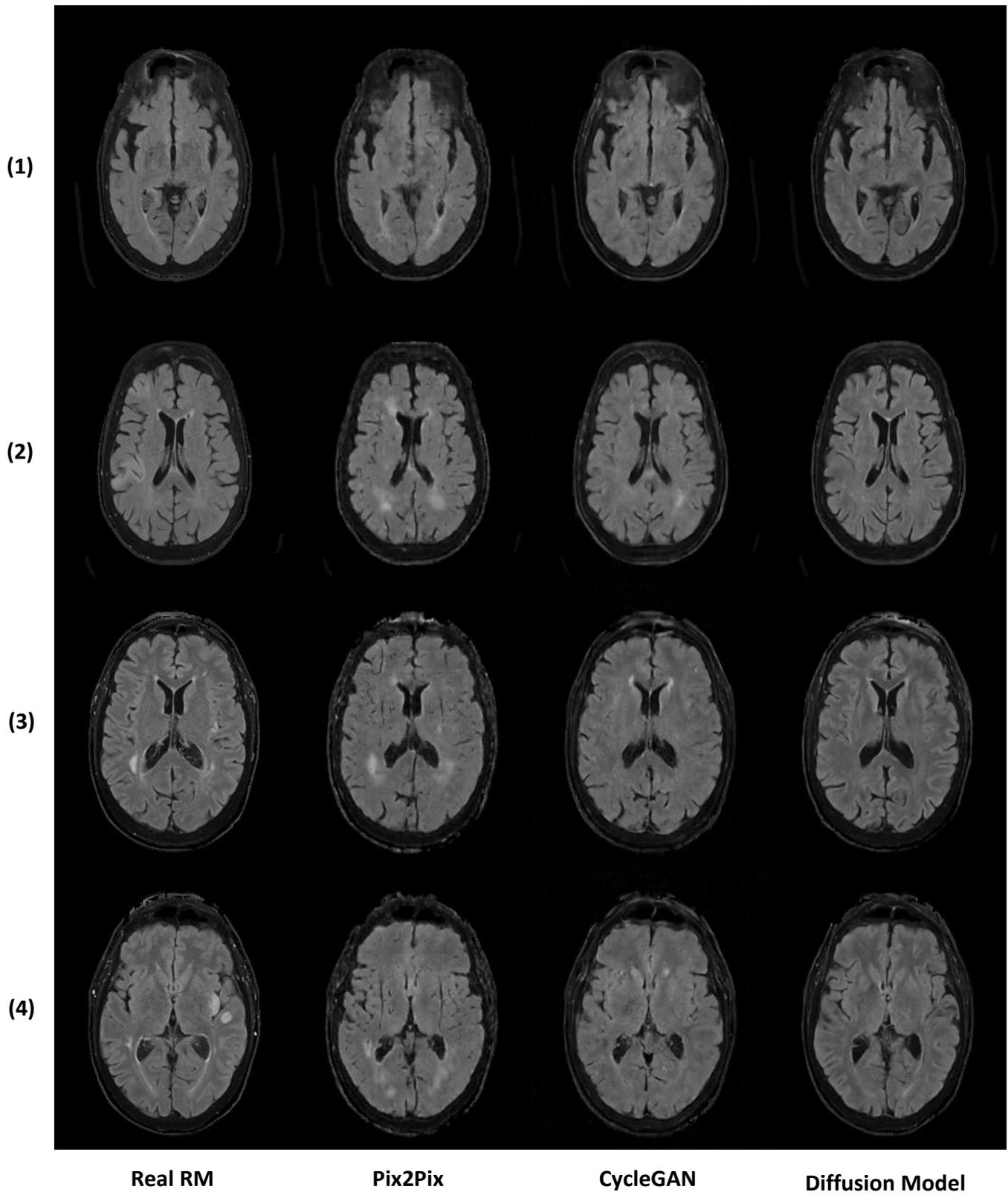


Figura 22. Traduzione frame di test modelli migliori TC-FLAIR

7.2 Traduzione TC-MRI DWI

Pix2Pix:

In tabella 9 e 10 vengono riassunte le prestazioni ottenute nell'epoca migliore per ogni parametrizzazione impostata.

Nello specifico vengono riportati i valori medi di RMSE, PSNR e SSIM calcolati sui frame di test, e i valori medi di MAE e ME calcolati sulle aree in presenza di lesione.

In giallo viene infine evidenziato il modello scelto come migliore per la seguente tipologia di architettura.

ndf	GAN_mode	ngf	Best Epoch	RMSE	PSNR	SSIM	MAE	ME
9	lsgan	9	90	11.96	26.45	0.81	31.32	-29.67
9	vanilla	9	90	11.64	26.87	0.79	31.23	-29.82
24	lsgan	24	80	18.61	23.20	0.67	34.92	-32.51
24	vanilla	24	120	18.70	23.15	0.66	35.19	-33.53
32	lsgan	32	120	19.23	23.21	0.67	34.54	-32.34
32	vanilla	32	45	19.43	23.12	0.65	34.93	-32.78

Tabella 10. Risultati Pix2Pix TC-DWI, parametrizzazione sbilanciata

ndf	GAN_mode	ngf	Best Epoch	RMSE	PSNR	SSIM	MAE	ME
9	lsgan	40	60	11.82	26.55	0.78	30.13	-29.23
9	vanilla	40	25	12.44	25.88	0.75	31.43	-29.13
24	lsgan	96	75	18.07	23.21	0.68	26.89	-23.5
24	vanilla	96	95	16.52	22.07	0.67	33.17	-32.45
32	lsgann	128	45	18.95	23.14	0.66	34.38	-31.82
32	vanilla	128	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
64	lsgan	12	75	18.93	23.19	0.67	34.54	-33.12
64	vanilla	12	80	18.52	23.03	0.66	32.53	-32.11

Tabella 11. Risultati Pix2Pix TC-DWI, parametrizzazione bilanciata

L'evoluzione delle metriche durante la fase d'allenamento viene qui riassunta.

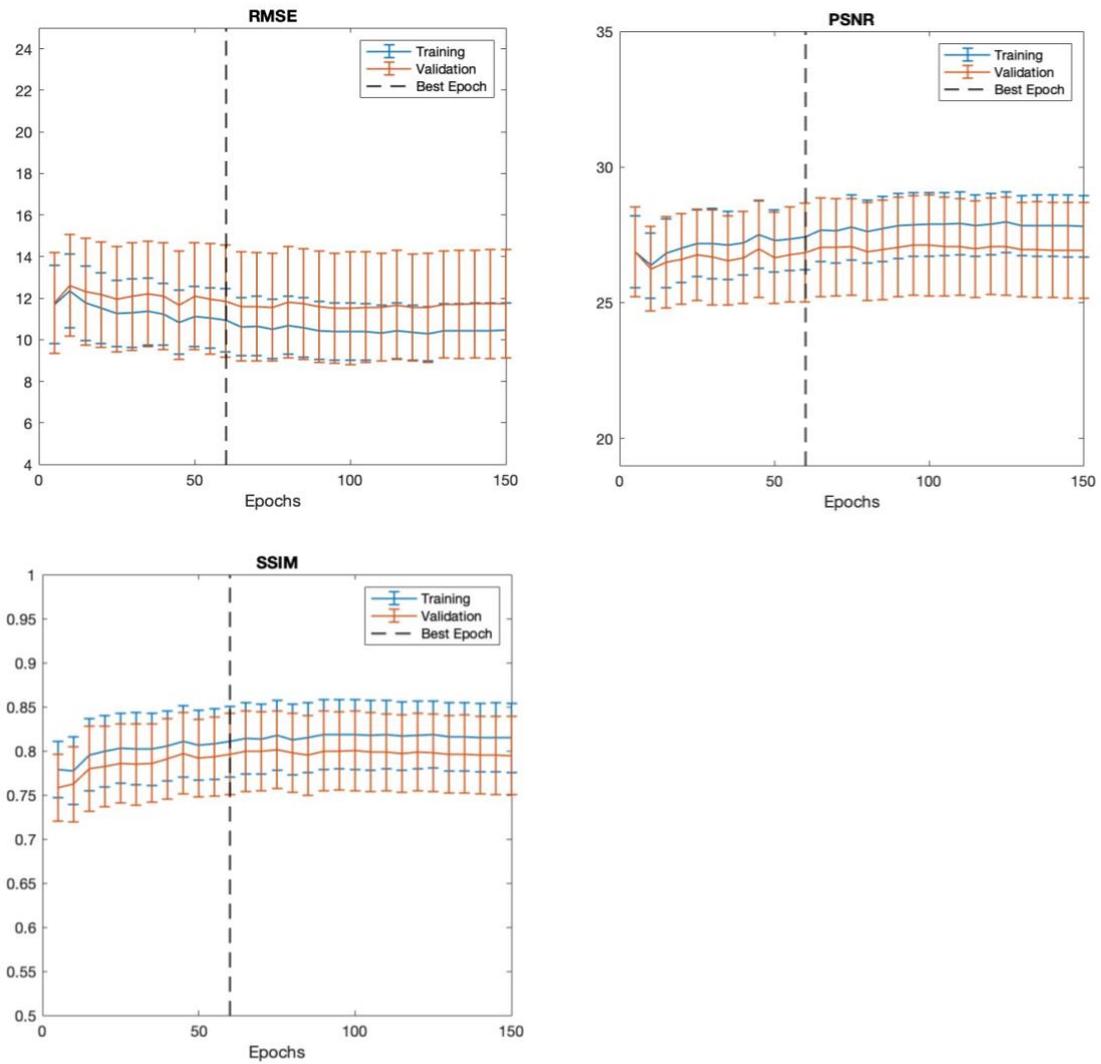


Figura 24. Pix2Pix TC-DWI: RMSE, PSNR e SSIM su training e Validation (valor medio \pm std)

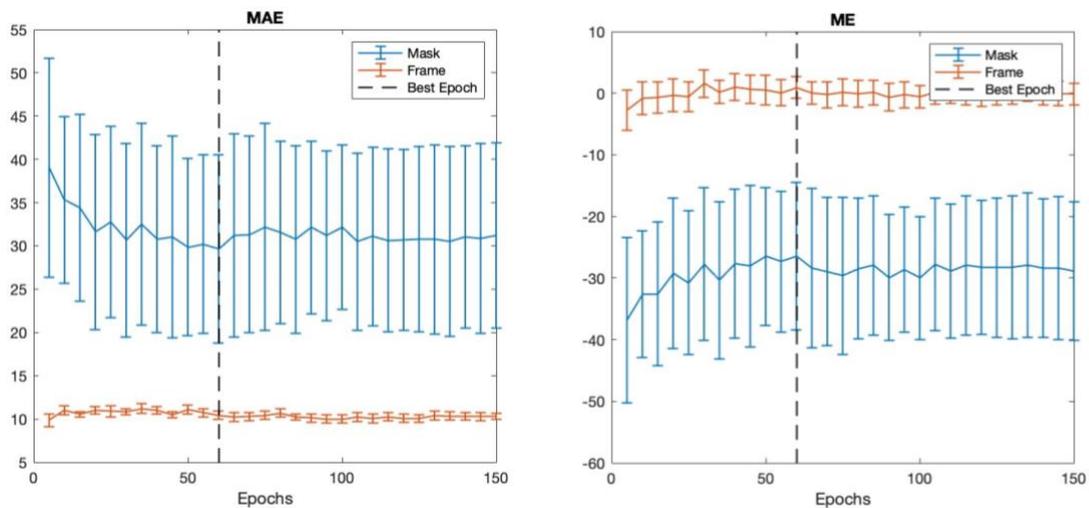


Figura 23. Pix2Pix TC-DWI: MAE e ME calcolati su maschera di lesione e frame (valor medio \pm std)

CycleGAN:

Allo stesso modo, in tabella 11 e 12 vengono riassunti i risultati ottenute per le epoche migliori sui frame di Test, evidenziando in giallo il modello vincente.

ndf	GAN_mode	ngf	Best Epoch	RMSE	PSNR	SSIM	MAE	ME
9	lsgan	9	95	19.31	22.75	0.66	31.32	-29.67
9	vanilla	9	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
24	lsgan	24	80	19.71	22.88	0.66	34.92	-32.51
24	vanilla	24	65	19.76	23.15	0.65	35.19	-33.53
32	lsgan	32	85	19.31	23.10	0.66	34.54	-32.34
32	vanilla	32	75	19.83	22.87	0.65	34.93	-32.78

Tabella 12. . Risultati CycleGAN TC-DWI parametrizzazione sbilanciata

ndf	GAN_mode	ngf	Best Epoch	RMSE	PSNR	SSIM	MAE	ME
9	lsgan	40	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
9	vanilla	40	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
24	lsgan	96	85	19.41	22.59	0.67	26.89	-23.5
24	vanilla	96	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
32	lsgann	128	65	19.55	22.46	0.66	34.38	-31.82
32	vanilla	128	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
64	lsgan	12	90	19.45	23.11	0.66	34.54	-33.12
64	vanilla	12	70	19.45	22.93	0.65	32.53	-32.11

Tabella 13. . Risultati CycleGAN TC-DWI, parametrizzazione bilanciata

L'evoluzione delle metriche durante la fase d'allenamento viene qui riassunta.

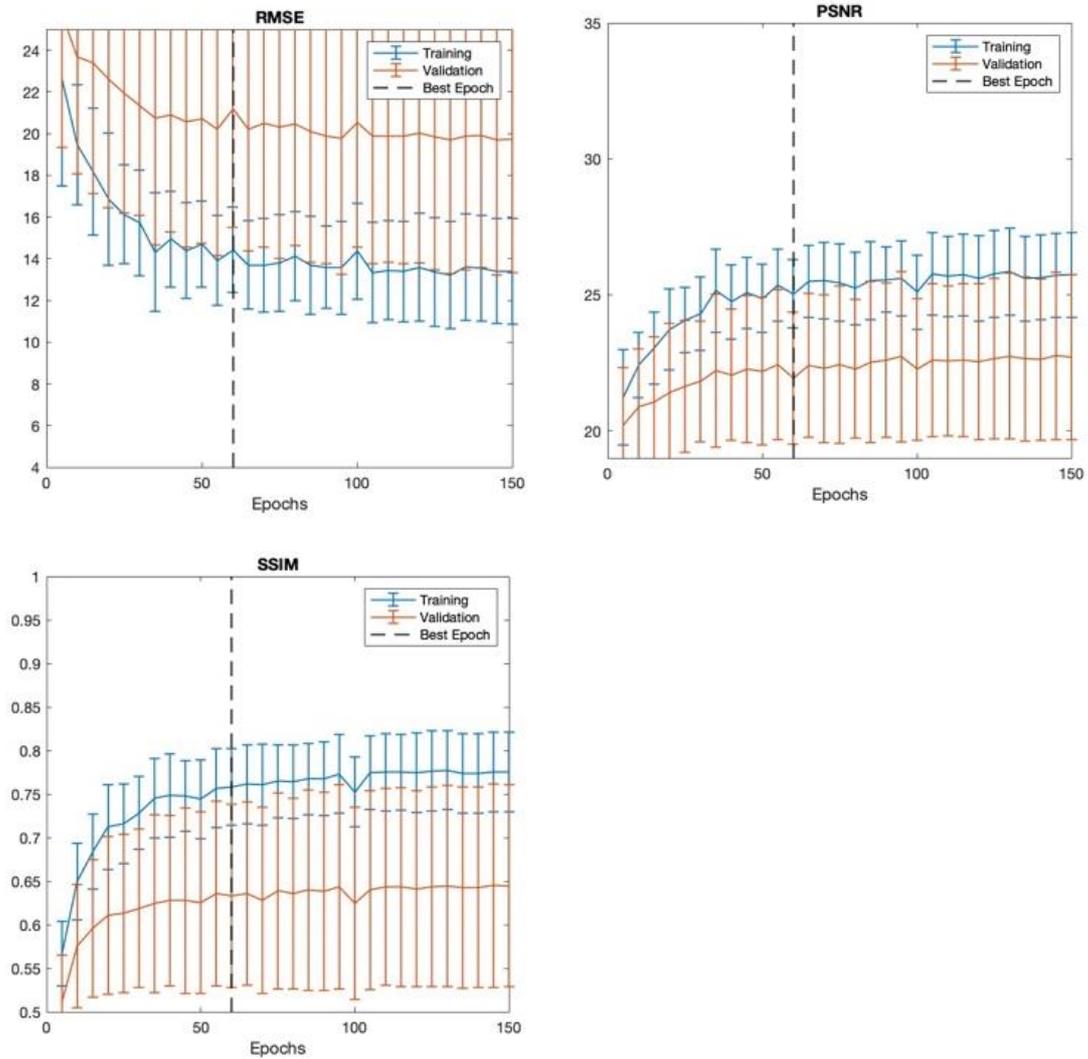


Figura 25. CycleGAN TC-DWI: RMSE, PSNR e SSIM su training e Validation (valor medio \pm std)

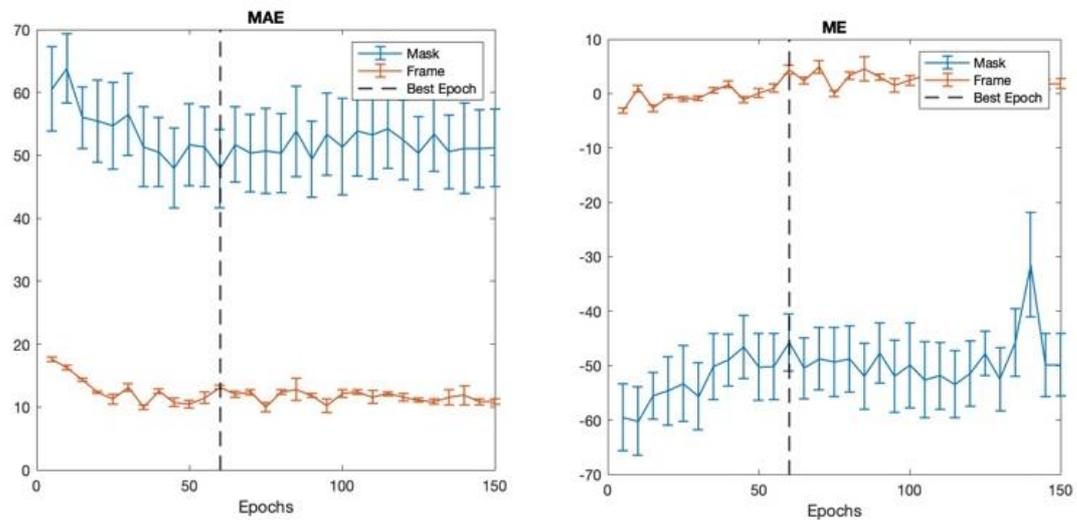


Figura 26. CycleGAN TC-DWI: MAE e ME calcolati su maschera di lesione e frame (valor medio \pm std)

Diffusion Models:

Anche per questa modalità di traduzione, a causa dell'impossibilità di eseguire l'inferenza sui frame di training dovuti agli alti costi computazionali in fase di generazione, le metriche di valutazione sono state calcolate solo sul test, prelevando in seguito l'epoca migliore per ogni parametrizzazione e riportandole tra loro.

In tabella 9 vengono riassunti i risultati mettendo in evidenza la vincente.

Batch size	Schedule	Best Epoch	RMSE	PSNR	SSIM	MAE	ME
1	Cosine	55	17.81	23.32	0.75	37.31	-33.21
1	Linear	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
2	Cosine	45	16.38	24.53	0.78	36.29	-32.15
2	linear	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.

Tabella 14. Risultati Diffusion Models TC- DWI

In Fig. 22 vengono mostrati quattro esempi significativi di generazione su frame di test, paragonando le tre architetture migliori rispetto al riferimento.

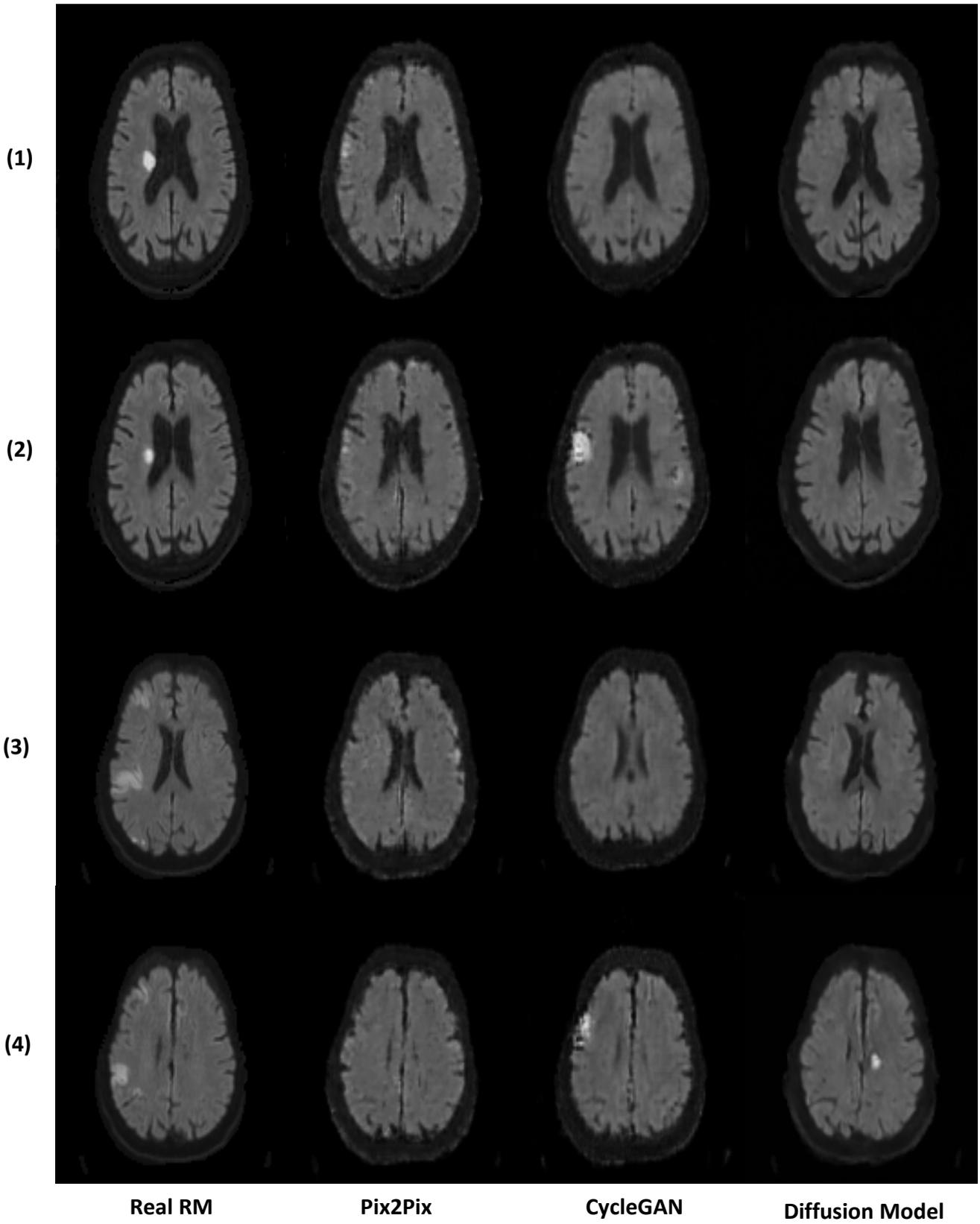


Figura 27. Traduzione frame di test modelli migliori TC-DWI

8 Conclusioni

In seguito all'analisi comparativa dei risultati ottenuti dai migliori modelli, si possono trarre le seguenti conclusioni.

In generale, nonostante lo scarso quantitativo di pazienti a disposizione nella strutturazione dei dataset d'allenamento, ogni architettura proposta risulta essere in grado di effettuare una buona traduzione TC-MRI, sia per quanto riguarda la modalità FLAIR che DWI.

In particolare, si vuole porre l'attenzione sull'alta qualità espressa in fase di *domain transfer* dai Diffusion Models di recente introduzione, i quali risultano essere i migliori a livello di realismo percepito rispetto alle Pix2Pix e CycleGAN.

Per quanto riguarda invece la capacità dei modelli di apprendere informazioni in modalità TC che siano sintomo di lesione in corso, traducendola in regione iper-intensa nell'MRI corrispondente, l'unica architettura che risulta in grado di effettuare questo task è la Pix2Pix in modalità TC-MRI FLAIR.

Infatti, come si può apprezzare dalla Fig. 22, ove nell'immagine di riferimento si rileva presenza di ictus, il modello riesce a mettere in evidenza parte dell'area, permettendo di classificare il paziente come positivo alla patologia.

Le stesse conclusioni non si possono però trarre per la traduzione in modalità TC-MRI DWI, in quanto nessuna architettura è in grado di rilevare correttamente le aree con lesione.

Infatti, nonostante nell'analisi visiva compaiano regioni iper-intense in fase di traduzione sui frame di test, esse non si trovano mai in corrispondenza a quelle presenti sul riferimento, da cui si può dedurre che ciò non sia frutto di un reale apprendimento.

Molto probabilmente questo è dovuto alla minor risoluzione delle immagini presenti questo dataset (128x128) rispetto alla modalità MRI FLAIR (256x256), in cui il *down sampling* rende più complicato la rilevazione delle informazioni in modalità TC fondamentali per l'allenamento.

8.1 Sviluppi Futuri

In funzione delle considerazioni precedentemente espresse, possibili futuri miglioramenti potranno riguardare i seguenti aspetti:

- L'incremento in termini di numerosità dei pazienti a disposizione è la prima componente essenziale per attendersi miglioramenti in termini di prestazioni durante la fase di training.
Ciò, infatti, permetterebbe di aumentare la variabilità dei dati, rendendo quindi maggiormente generalizzabile il modello in uscita per il riconoscimento delle diverse tipologie di lesione.
- L'introduzione di un parametro aggiuntivo all'interno della funzione di *Loss* che tenga conto della qualità di traduzione della lesione potrebbe aumentare il focus da parte dei modelli generativi per la loro rilevazione in fase di training.
Come si può vedere infatti nei grafici rappresentanti l'evoluzione delle metriche, l'andamento dell'errore medio calcolato sulle aree con lesione raggiunge lo stato asintotico prima dei coefficienti di qualità di traduzione.
Questo potrebbe indicare che siccome le aree infartuate hanno dimensioni inferiori rispetto al resto del frame, il modello si concentri maggiormente ad aumentare le proprie prestazioni sulla qualità di traduzione a discapito della lesione, la quale viene considerata come "anomalia".
- Una miglior registrazione dei volumi potrebbe portare a miglioramenti prestazionali da parte dei modelli in esame.
Infatti, a causa della casistica in possesso che risultava essere già processata e formattata, la qualità dei risultati in termini di corrispondenza spaziale tra i volumi in uscita sono state inferiori, comportando quindi uno scorporo dei dati eliminando i frame maggiormente deformati.

In conclusione, lo studio preliminare proposto pone l'attenzione su come i modelli generativi possano potenzialmente rappresentare uno strumento complementare al protocollo diagnostico per pazienti affetti da ictus cerebrale, meritando quindi ulteriori approfondimenti.

Bibliografia

- [1] «Humanitas,» [Online]. Available: <https://www.humanitas.it/malattie/ictus-cerebrale/>.
- [2] «Fondazione Umberto Veronesi,» [Online]. Available: <https://www.fondazioneveronesi.it/magazine/articoli/neuroscienze/diminuiscono-gli-ictus-e-anche-la-mortalita>.
- [3] H. A. Z. S. S. A. Waqar Ahmad, «A new generative adversarial network for medical images super resolution,» *Scientific Reports*, June 2022.
- [4] O. S. T. Y. X. J. Tang Y, «CT realistic data augmentation using generative adversarial network for robust lymph node segmentation.,» *Medical Imaging: Computer- Aided Diagnosis*, 2019.
- [5] H. J. A. H. J. J. K. E. K. C. J. M. P. Seung Kwan Kang, «Synthetic CT generation from weakly paired MR images using cycle-consistent GAN for MR-guided radiotherapy,» 2021.
- [6] H. K. M. L. W. J. S. J. E. P. Y. S. A. I. H. H. J. X. C. Cheng-Bin Jin, «Deep CT to MR Synthesis Using Paired and Unpaired Data,» 2019.
- [7] P. Q. R. C. J. Z. Q. W. Y. M. P. W. Eryan Feng, «MRI Generated from CT for Acute Ischemic Stroke Combining Radiomics and Generative Adversarial Networks,» *IEEE Journal of Biomedical and Health Informatics*, 2022.
- [8] M. S. Stefan Klein, «Elastix The Manual,» 2023.
- [9] P. T. a. M. Unser, «Optimization of mutual information for multiresolution image registration,» *IEEE Trans. Image Process.*, 9(12):2083 – 2099, 2000.
- [10] J. P. W. P. M. S. a. M. V. S. Klein, «Adaptive stochastic gradient descent optimisation for image registration,» *International Journal of Computer Vision*, 81(3):227 – 239, March 2009.
- [11] «Albumentations,» [Online]. Available: <https://albumentations.ai>.
- [12] «OpenCV,» [Online]. Available: <https://opencv.org>.
- [13] J. P.-A. M. M. B. X. D. W.-F. S. O. A. C. a. Y. B. I. Goodfellow, «Generative adversarial nets,» *NIPS*, 2014.
- [14] J.-Y. Z. T. Z. A. A. E. Phillip Isola, «Image-to-Image Translation with Conditional Adversarial Networks,» 2018.
- [15] «AISUMMER,» [Online]. Available: <https://theaisummer.com/skip-connections/>.
- [16] G. U. Uğur Demir, «Patch-Based Image Inpainting with Generative Adversarial Networks,» March 2018.
- [17] T. P. P. I. A. A. E. Jun-Yan Zhu, «Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks,» 2020.
- [18] Z. Z. Y. S. Ling Yang, «Diffusion Models: A Comprehensive Survey of Methods and Applications,» 2022.
- [19] «AssemblyAI,» [Online]. Available: <https://www.assemblyai.com/blog/diffusion-models-for-machine-learning-introduction/>.
- [20] T. P. Jun-Yan Zhu, «GitHub,» [Online]. Available: <https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix>.
- [21] J. H. W. C. T. S. D. J. F. M. N. Chitwan Saharia†, «Chitwan Saharia†, Jonathan Ho, William Chan, Tim Salimans, David J. Fleet, Mohammad Norouzi,» 2021.
- [22] «GitHub,» [Online]. Available: <https://github.com/Janspiry/Image-Super-Resolution-via-Iterative-Refinement>.

Ringraziamenti

Desidero concludere questo elaborato ringraziando il reparto di neurologia dell'ospedale "Maria Vittoria" di Torino, il quale ci ha messo gentilmente a disposizione la casistica necessaria allo sviluppo del progetto.

Un sentito grazie al mio Relatore Ing. Massimo Salvi e Correlatore Prof. Filippo Molinari per la loro grande disponibilità dimostrata in questi mesi, sempre presenti in ogni fase e pronti a darmi le corrette indicazioni in caso di bisogno.

Grazie ai miei genitori, senza i cui sacrifici e il loro sostegno in ogni momento oggi non sarei qui a celebrare la conclusione del mio percorso di studi.

A mia mamma, che a suo dire ha patito più del sottoscritto tutte le difficoltà e le ansie per ogni esame sostenuto, e per questo motivo meriterebbe una laurea ad honorem.

A mio papà che, nonostante i nostri battibecchi in stile adolescenziale, ha sempre saputo smorzare i momenti più difficili con qualche battuta o frase "filosofica".

Grazie a mio fratello Gianluca che, malgrado la distanza fisica che ci separa, non ha mai fatto mancare la sua vicinanza sia nei momenti felici che in quelli più complessi.

A Chiara, che più di tutti in questi due anni ha saputo spronarmi e sostenermi in ogni scelta e con la quale non vedo l'ora di cominciare il nuovo capitolo di vita che ci attende.

Grazie agli amici di una vita, senza i quali questi anni universitari non sarebbero stati sicuramente la stessa cosa.

A Davide, inseparabili dal primo momento e fedele compagno di ogni avventura. A Bagna e le nostre fondamentali riunioni al "Fuego Bar". A Jack e le nostre degustazioni di birra che non termineranno mai. A Cappi per la sua positività di vita. E infine, ultimo ma non ultimo, a Bigna e al nostro supporto reciproco durante tutto questo percorso.