

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA
SEDE DI CESENA
SECONDA FACOLTÀ DI INGEGNERIA CON SEDE A CESENA
CORSO DI LAUREA IN INGEGNERIA ELETTRONICA

TITOLO DELLA TESI

***SVILUPPO DI ALGORITMI E INTERFACCIA GRAFICA PER APPLICAZIONI IN AMBITO
MEDICO A SUPPORTO DELLO STUDIO PERFUSIONALE***

Tesi in

Reti Logiche LA

Relatore:

Prof. ALESSANDRO BEVILACQUA

Co-relatori:

Prof. GIAMPAOLO GAVELLI

Dott. DOMENICO BARONE

Dott. LUDOVICO CAROZZA

Presentata da:

JOSÉ SILVIO FALLA

Sessione I

Anno Accademico 2011-2012

PAROLE CHIAVE

Formato DICOM

Analisi TC

Perfusione

Software per Studio Perfusionale

Elaborazione di Immagini

Archiviazione e Visualizzazione Immagini

Alice : ..ma è impossibile!!
Il Cappellaio Matto : Solo se pensi che lo sia!!

Indice

Introduzione	7
Il progetto “PERFECT”	7
Lo studio perfusionale	9
Obiettivi della Tesi	10
1. Concetti Teorici e Problematiche d’interesse	12
1.1. Lo stato dell’arte in IRST all’inizio della mia attività .	12
1.2. Competenze mediche acquisite	13
1.2.1. Fondamenti della CT	13
1.2.2. Le HU (Hounsfield Unit)	15
1.2.3. Standard DICOM	16
1.2.4. Il file DICOM	18
1.2.5. L’Header File	19
1.2.6. Visualizzazione delle immagini CT	20
1.2.7. Lettura/scrittura file DICOM	22
1.2.8. Trattamento e filtraggio delle immagini	22
1.2.9. I campi dell’header file DICOM utilizzati	22
1.2.10. Tracciamento manuale ROI	23
1.2.11. Tracciamento automatico ROI	24
1.2.12. La PET	25
1.3 Elaborazione delle immagini	26
1.3.1 L’immagine e la sua formazione	26
1.3.2 L’immagine come matrice bidimensionale di pixel	28
1.3.3 RGB e Gray Scale	29
1.3.4 Filtri	31
1.3.5 ROI	32
1.3.6 Gradiente di un’immagine	32
1.3.7 Edge Detectors	33
1.3.8 Istogramma e Sogliatura	34
1.3.9 Maschere	36
1.3.10 Image Processing in Matlab	37
1.4 Attività Integrative svolte	38

2. Il Progetto	43
2.1. Fasi preliminari di progetto	44
2.1.1. Raccolta delle problematiche/esigenze dei medici	45
2.1.2. Identificare i vincoli	46
2.1.3. Analisi dei requisiti	46
2.2. Dalle esigenze dei medici ad un primo modello di software	47
2.3. Dal primo modello in Matlab al software in Java	48
2.4. L'architettura del software in Java	50
2.4.1. Strumenti software utilizzati	51
2.4.2. Il software PERFECT... i macroblocchi	52
3. Implementazione del progetto	56
3.1. Il software PERFECT... la Graphical User Interface 	60
4. Risultati	88
5. Conclusioni e Sviluppi Futuri	92
6. Ringraziamenti.....	94
Bibliografia	97
Indice delle Figure.....	99

Introduzione

Il seguente elaborato mira ad illustrare l'attività di tesi svolta presso IRCCS IRST di Meldola, cercando di mettere in risalto le problematiche affrontate in struttura ed all'interno del progetto "PERFECT". Si farà dapprima qualche cenno alla nozionistica acquisita per poter proseguire nel progetto, verrà fatto quindi qualche accenno alle attività integrative svolte nel periodo di tesi.

A seguire si illustreranno le varie fasi dell'attività, accompagnando l'esposizione con immagini esemplificative di quanto detto.

Il progetto "PERFECT"

La Tomografia Computerizzata (TC) è nata per fondere assieme le potenzialità diagnostiche della radiologia e quelle dell'elaborazione digitale, proprie dei computer. La TC a 256 *slice* di ultima generazione offre, rispetto alle TC di precedente generazione, la possibilità di studiare volumi più ampi in tempi ridotti, consentendo di ottenere una migliore resa, per risoluzione e definizione, soprattutto a livello di strutture vascolari, riducendo inoltre la dose di radiazioni a cui viene esposto il soggetto. Lo studio perfusionale rappresenta un valido strumento per la valutazione dell'efficacia delle terapie dirette a bloccare la vascolarizzazione tipica delle lesioni tumorali. Questo consentirebbe di valutare già in fase precoce (dopo il primo ciclo di terapia) se la neoangiogenesi tumorale si riduce, senza attendere i cambiamenti dimensionali, che avvengono più tardivamente. Il progetto PERFECT (CVG)^[2] si

pone come obiettivo lo sviluppo di una metodologia basata sull'analisi assistita (semiautomatica) di una sequenza temporale di immagini TC 3D in grado di consentire una valutazione quantitativa, o semiquantitativa, dell'efficacia del trattamento delle lesioni epatiche secondarie attraverso l'analisi dell'evoluzione temporale di alcuni parametri (fotometrici, geometrici, statistici, ecc.) da Regioni di Interesse (ROI) della lesione identificate da un radiologo esperto. Questi parametri, "tracciati" su tutte le immagini acquisite durante il trattamento, consentiranno di poter ottenere le misure necessarie per quantificare le modifiche nell'angiogenesi e, conseguentemente, l'efficacia della terapia.

Per approfondimenti: <http://www.irstunibo.it/perfect.html>



Lo studio perfusionale

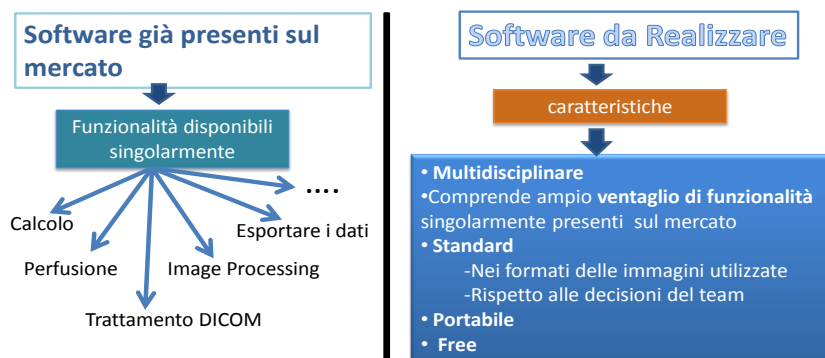
Lo studio perfusionale è una tecnica di diagnostica per immagini, atta a monitorare (come detto nel paragrafo precedente) l'evolversi della lesione tumorale sin dai primi giorni di terapia.

Per far questo, prima di tutto sarà necessario effettuare un esame CT, iniettando in vena al paziente una particolare sostanza detta "Mezzo di Contrasto", che grazie alle sue caratteristiche (a base di zuccheri), permette di evidenziare in modo chiaro i tessuti, i vasi, gli organi in cui transita. Acquisito dunque un certo numero di immagini (negli studi analizzati il numero è stato di 360 per ciascun esame), vengono archiviate. La fase successiva prevede l'intervento del radiologo che, dopo aver caricato le immagini le analizza, mette in evidenza le aree di particolare interesse, e ne esprime una diagnosi. Fatto questo, le immagini tornano ad essere archiviate.

Seguendo questa procedura per ogni esame CT, è possibile tenere sotto controllo gli eventuali cambiamenti nella morfologia della lesione tumorale. In gergo tecnico, questa operazione viene detta "Follow-up". Parte fondamentale dello studio perfusionale è quindi il poter confrontare blocchi di immagini appartenenti ad esami CT acquisiti in momenti diversi della terapia del paziente (anche qualche mese).

Obiettivi della Tesi

L'obiettivo della tesi è stato realizzare un nuovo software in grado di supportare il medico e gli ingegneri nelle rispettive attività di utente e progettista. Il software oggetto della tesi è totalmente innovativo, non tanto per gli strumenti messi a disposizione dell'utente, quanto per il fatto di racchiudere in un solo ambiente strumenti presenti singolarmente in software di altissima qualità già sul mercato.



Poiché il suo sviluppo è avvenuto a stretto contatto con l'utente finale, oltre alle caratteristiche poc'anzi citate, il software fornisce quanto richiesto dall'utente stesso e nei punti da esso stesso indicati. Ci troviamo quindi davanti ad uno strumento che è allo stesso tempo slegato da particolari vincoli d'utilizzo, e in grado di fornire configurazioni create su misura per l'utente.

L'elaborato che ci accingiamo a discutere si suddivide in :

- Capitolo 1: “Concetti Teorici e Problematiche d'interesse” – verranno esposti i concetti alla base dell'attività di tesi e le problematiche d'interesse affrontate. In fine, verranno citate le varie attività integrative svolte durante il periodo di tesi.
- Capitolo 2: “Il Progetto” – in questa parte dell'elaborato vengono illustrate le varie fasi di sviluppo del progetto. Si inizia con le fasi preliminari, per poi passare all'architettura del progetto.
- Capitolo 3: “L'implementazione” – verranno affrontati gli aspetti prettamente implementativi del progetto di tesi, commentando le finestre a disposizione dell'utente che andrà ad utilizzare il software e si metteranno in evidenza le parti salienti del codice sorgente.
- Capitolo 4: “Risultati”- poiché i dati elaborati dal software oggetto della tesi hanno necessitato di validazione costante, in questo capitolo verranno illustrate le modalità di validazione adottate.
- Capitolo 5: “Conclusioni e Sviluppi futuri” – dopo un'accurata discussione del progetto, è possibile trarre delle conclusioni. Essendo un progetto molto articolato, necessita di ulteriori sviluppi futuri.
- Capitolo 6: “Ringraziamenti”

Capitolo 1

Concetti Teorici e Problematiche d'interesse

Per poter affrontare, capire e cercare di operare al meglio, è stato necessario arricchire il mio bagaglio di conoscenze, apportando sostanziosi innesti di tipo multidisciplinare. Affacciarsi ad un ambito del tutto nuovo per me, ha richiesto l'apprendimento dei concetti descritti in seguito.

1.1. Lo stato dell'arte in IRST all'inizio della mia attività

All'inizio della mia esperienza in struttura, si è presentata ai miei occhi una panoramica di strumenti software tutti da studiare e tutta una vasta gamma di esigenze da parte dei medici tutte da soddisfare per poter rendere agevole l'indagine radiologica nello studio perfusionale. Dopo una fase preliminare di ascolto e raccolta delle esigenze delle figure professionali in gioco (fase che per poter avere un esito ottimale ha richiesto uno studio autonomo dei concetti di base di anatomia e terminologia medica), ha seguito una fase di studio, ricerca delle anomalie e bug dei software presenti in struttura e quotidianamente in uso.

1.2 Competenze mediche acquisite

L'acquisizione delle conoscenze relative al formato DICOM (Standard DICOM)^[15], come si presentano i relativi file generati dalla CT Philips, e il pre-trattamento delle immagini che saranno successivamente utilizzate per il calcolo dei parametri perfusionali, sono state oggetto di un precedente periodo di frequenza presso IRCCS IRST. Tutto ciò per poter acquisire le competenze necessarie, per effettuare le operazioni di elaborazione ed analisi di immagini necessarie al calcolo manuale delle curve caratteristiche nello studio perfusionale. Scopo della tesi è stato quindi realizzare nuovo software in grado di supportare il medico e gli ingegneri nelle rispettive attività di utente e progettista.

1.2.1 Fondamentali della CT

In radiologia la Tomografia Computerizzata, indicata con l'acronimo TC o CT (dall'inglese Computed Tomography), è una



Figura 1. Foto di una CT

metodica diagnostica per immagini, che sfrutta radiazioni ionizzanti (raggi X) e consente di ricostruire sezioni o strati (tomografia) corporei del paziente ed elaborazioni tridimensionali. Le

modalità di acquisizione delle immagini, possibili

grazie all'evoluzione delle tecniche in merito, sono principalmente due: assiale (da qui TAC) e spirale.

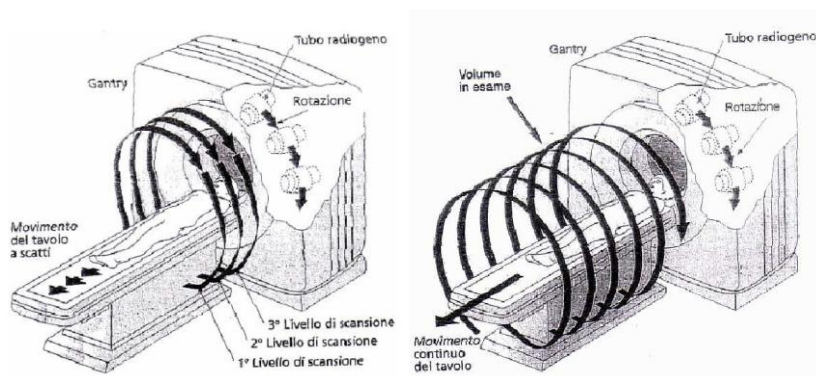
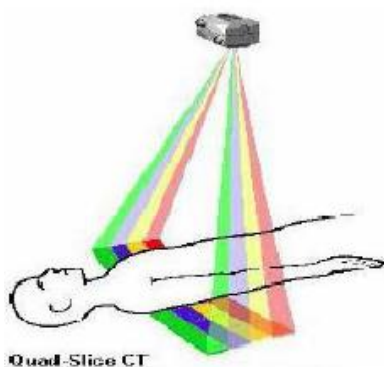


Figura 2: Tecniche di acquisizione in assiale (sinistra) e volumetrica

Il principio di funzionamento della CT si basa sostanzialmente



Quad-Slice CT

Figura 3: Quad-Slice CT

nell'irradiare (in modo appropriato e nelle tempistiche giuste) il corpo del paziente con un fascio di raggi X, opportunamente dosati in base alle esigenze d'indagine, e alle caratteristiche del paziente. I raggi (RX) emessi dal "Tubo radiogeno", investono il corpo del paziente, e ciò che rimane di loro a passaggio avvenuto (attenuazione), viene accolto da una schiera di detettori

(disposti in diverso modo al variare della tecnologia adottata) in grado acquisire la radiazione impattante su di essi. Tanto più il tessuto attraversato dalla radiazione è denso, tanto più si oppone al suo passaggio. Ecco che i tessuti ossei risultano meno attraversabili, impedendo quindi un passaggio inalterato della radiazione stessa. L'opposto accade per organi ricchi d'aria come i polmoni. La misura dell'intensità della radiazione (quindi dell'attenuazione causata dai tessuti) avviene in UH (def. al paragrafo 2.2.3). A partire dai profili di attenuazione è possibile, tramite tecniche di ricostruzione, generare immagini delle sezioni dei tessuti irradiati. In tali immagini, i tessuti a maggiore densità assumono una colorazione più chiara, mentre i meno densi una colorazione scura, fino al caso dell'aria che risulta molto vicina al nero. Si ha dunque una scala di grigi (quanto più possibile) ben definita, che permette il collegamento biunivoco tra livelli di grigio e tessuto osservato. Grazie a questa caratteristica è possibile

individuare delle anomalie all'interno dei tessuti. Questo tipo di operazione viene aiutata dall'inserimento in vena di un mezzo di contrasto (soluzione iodata) che, al suo passaggio, evidenzia le caratteristiche del tessuto, permettendo diverse valutazioni di tipo diagnostico. Va osservato come la scala di grigi sia solo una astrazione ottica per poter rappresentare i livelli di attenuazione.

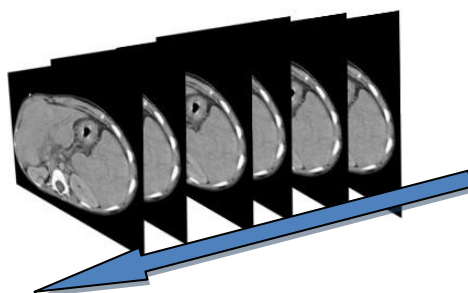


Figura 3: esempio sequenza immagini CT

Per approfondimenti sulle modalità di svolgimento dell'esame, e sulle specifiche tecniche del tomografo utilizzato presso IRST, si faccia riferimento all'elaborato di fine tirocinio (svolto presso IRST) di Mattia Gardini (Gardini)^[5].

1.2.2 Le HU (Hounsfield Unit)

L' HU (Hounsfield Unit), è l'unità di misura di alcune grandezze come "Radiodensity" e "Radioopacità". Per "Radiodensity" si intende l'incapacità delle radiazioni (in questo caso i raggi X) di attraversare un determinato corpo o tessuto (attenuazione). Si hanno dunque dei tessuti che si oppongono più di altri al passaggio della radiazione (radio densi) ed altri meno (radio trasparenti). Le HU prendono il nome dall'ingegnere Godfrey Newbold Hounsfield, inventore e realizzatore della metodica circolare alla base della TC. Esse rappresentano secondo una scala lineare i coefficienti di attenuazione dei tessuti, a seconda della diversa radiodensità/radioopacità. Come valori di riferimento si hanno quello dell'acqua distillata a pressione e temperatura standard

(0 HU) e quella dell'aria alle medesime condizioni (-1000 HU).

I relativi coefficienti di attenuazione vengono indicati con μ_{acqua} e μ_{aria} . Per calcolare il corrispondente valore in HU di un qualsivoglia materiale X, nota il relativo coefficiente di attenuazione μ_X , facciamo riferimento alla relazione seguente:

$$HU = \frac{\mu_X - \mu_{\text{water}}}{\mu_{\text{water}} - \mu_{\text{air}}} \times 1000$$

Nella tabella a fianco vengono riportati alcuni valori di radiodensità, espressi in HU, per i tessuti e le sostanze più comuni nelle indagini TC. I loro valori (fatta eccezione per l'aria e l'acqua) non sono fissi, ma dipendono dai parametri energetici (come i kV ed i mAs) del fascio radiante. (Mathias Prokop, 2010)

Sostanza	HU
Aria	-1000
Grasso	-120
Acqua	0
Sangue	30-45
Muscolo	40
Contrasto	130
Osso	400 o più

1.2.3 Standard DICOM

Lo standard DICOM (Digital Imaging and COmmunications in Medicine) definisce i criteri per la comunicazione, la visualizzazione, l'archiviazione e la stampa di informazioni di tipo biomedico quali ad esempio immagini radiologiche.

Lo standard DICOM è pubblico, nel senso che la sua definizione è accessibile a tutti. La sua diffusione si rivela estremamente vantaggiosa perché consente di avere una solida base di interscambio di informazioni tra apparecchiature di diversi produttori, server e PC, specifica per l'ambito biomedico.

Occorre notare che DICOM non è solo uno standard industriale ma anche uno standard de jure (ISO 12052:2006). Nel passato, prima del riconoscimento come standard ISO, esisteva una certa tolleranza nell'implementazione delle specifiche, al punto che forse non esistevano apparecchiature che potessero definirsi pienamente DICOM compliant, nel senso rigoroso che la

definizione di uno standard imporrebbe. Nella maggior parte dei casi, infatti, un'apparecchiatura risultava conforme ad una parte dello standard (ad esempio la modalità di archiviazione delle immagini), mentre adottava tecnologie proprietarie per altre funzionalità (ad esempio la gestione delle liste pazienti).

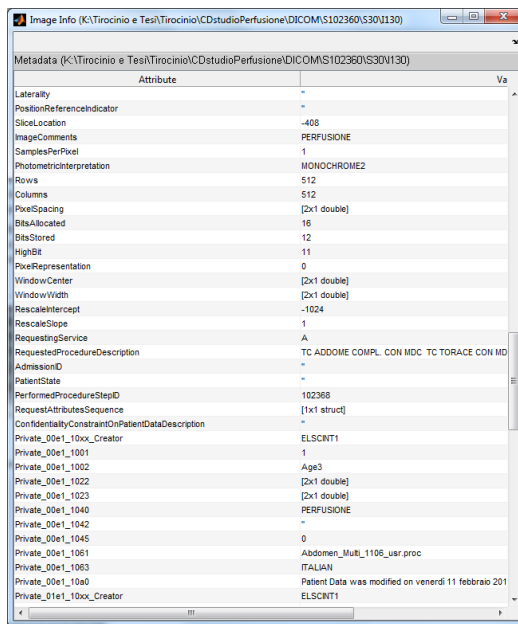
La compatibilità DICOM ed in generale di qualsiasi dispositivo DICOM compatibile deve essere certificata dal costruttore attraverso un documento autocertificativo, denominato Conformance Statement, che ne elenchi le funzionalità. La trattazione dettagliata del conformance statement che un costruttore deve emettere, per poter dichiarare una sua applicazione conforme allo standard DICOM, è contenuta nel volume 2 delle specifiche DICOM. Il buon esito di una connessione tra due apparecchiature DICOM è in prima istanza legato al confronto tra i due conformance statement, a meno di errori sui documenti od omissioni nell'implementazione, eventualità tutt'altro che remote.

1.2.4 Il file DICOM

Lo standard DICOM si basa su un modello concettuale definito "Service Object Pair" (SOP). Il modello SOP è ispirato al paradigma "Object Oriented" (OO) dell'analisi e della programmazione in informatica. Una entità del mondo reale come un ricovero, un'immagine, un paziente, viene definita come un oggetto; ogni oggetto contiene una serie di attributi. Per esempio l'oggetto paziente conterrà gli attributi: dati anagrafici, data di ricovero ecc. Definiti gli oggetti e tutti i loro attributi, lo standard DICOM definisce quali operazioni possono essere eseguite e su quali oggetti. La combinazione di un oggetto ed i corrispettivi servizi (DIMSE- DICOM Message Service Elements) prende il nome di SOP (Service Object Pair). L'insieme delle SOP relative ad un unico oggetto formano una SOP Class. Nel caso CT, ogni singola *Slice*, contenuta in un file, può essere modellata come un oggetto, e come tale, gode di tutte le caratteristiche attribuibili a tale entità.

1.2.5 L'Header File

I dati radiologici rappresentabili come immagini, o le immagini vere e proprie che vengono archiviate secondo lo standard DICOM sotto forma di file, vengono comunemente chiamate immagini DICOM. L'errore più comune che viene fatto nell'interpretazione del termine è che queste siano assimilabili ad altri formati di compressione dell'immagine (es. JPEG, GIF, etc.).



Attribute	Value
Laterality	-
PositionReferenceIndicator	-
SliceLocation	-408
ImageComments	PERFUSIONE
SamplesPerPixel	1
PhotometricInterpretation	MONOCHROME2
Rows	512
Columns	512
PixelSpacing	[2x1 double]
BitsAllocated	16
BitsStored	12
HighBit	11
TypeRepresentation	0
WindowCenter	[2x1 double]
WindowWidth	[2x1 double]
RescaleIntercept	-1024
RescaleSlope	1
RequestingService	A
RequestedProcedureDescription	TC ADDOME COMPL. CON MDC TC TORACE CON MD
AdmissionID	-
PatientState	-
PerformedProcedureStepID	102368
RequestedAttributesSequence	[1x1 struct]
ConfidentialityConstraintOnPatientDataDescription	-
Private_00e1_10cc_Creator	ELSCNT1
Private_00e1_1001	1
Private_00e1_1002	Age3
Private_00e1_1022	[2x1 double]
Private_00e1_1023	[2x1 double]
Private_00e1_1040	PERFUSIONE
Private_00e1_1042	-
Private_00e1_1045	0
Private_00e1_1061	Abdomen_Multi_1106_usr.proc
Private_00e1_1063	ITALIAN
Private_00e1_10a0	Patient Data was modified on venerdì 11 febbraio 201
Private_01e1_10cc_Creator	ELSCNT1

Figura 4: Header File letto con Matlab

In verità lo standard DICOM applicato alla codifica dei file è un metodo per incapsulare i dati e per definire come questi debbano essere codificati o interpretati, ma non definisce alcun nuovo algoritmo di compressione.

La maggior parte delle volte, l'immagine viene archiviata in forma non compressa, secondo la codifica con la quale

viene prodotta, ma esistono molti software che sono in grado di produrre o interpretare file DICOM contenenti dati compressi secondo vari algoritmi (JPEG, JPEG Lossless, JPEG Lossy, vari algoritmi dello standard JPEG2000, ecc.). Un file DICOM, oltre all'immagine vera e propria, include anche un "Header". Le informazioni contenute nell'Header DICOM sono molteplici, per esempio: nome e cognome del paziente, il tipo di scansione (Assiale/Spirale), posizione e dimensione dell'immagine (all'interno dell'esame CT), posizione del lettino, numero della *slice* all'interno dell'intera seduta osservativa, età del paziente ecc. Tutte le informazioni memorizzate nell'Header vengono catalogate

in gruppi di elementi, detti anche "Tag DICOM" . Proprio quest'ultimo aspetto (il numero dei campi DICOM presenti nell'Header file) si è rivelato essere uno dei fattori critici fortunatamente risolti.

1.2.6 Visualizzazione delle immagini CT

Al termine di ogni scansione (un esame completo consta generalmente di più strati) l'immagine numerica CT dello strato viene sia archiviata nella memoria di massa dell'elaboratore, sia immediatamente visualizzata su monitor televisivo, in quanto i livelli di grigio sono convertiti in segnali analogici compatibili con i segnali video di un tubo a raggi catodici (CRT). Il "windowing" (finestratura) è quell'attività preliminare, svolta nell'elaborazione di un'immagine. Grazie ad essa, è possibile enfatizzare alcuni aspetti dell'immagine e metterne in secondo piano altri, permettendo una valutazione più oculata dell'ambiente immagine. Nella tecnica CT vengono messe a disposizione del medico una serie di finestre "pre-impostate" i cui valori vengono assegnati a priori, poiché è noto che essi permettano una più corretta visione di un'immagine di tipo epatico, piuttosto che di tipo polmonare etc. Ecco che spesso si sente parlare di "finestra epatica", "finestra polmonare" e di tante altre. In realtà, non sono altro che dei buoni esiti di valutazione ottica di una tipologia d'indagine CT. Nella immagine CT visualizzata, la matrice ricostruita è dunque rappresentata da un insieme di piccoli quadrati (pixel) come delle tessere che formano un mosaico (si veda il paragrafo 3.2.1), con una luminosità o tono di grigio del singolo pixel corrispondente all'elemento di matrice calcolata (figura 62). La matrice dell'immagine numerica è caratterizzata dalle sue dimensioni (si utilizzano in genere matrici quadrate) e dal numero di bit per punto: ad esempio una matrice di 256 x 256 punti, ciascuno di 12 bit oppure una matrice 512 x 512 x 12 bit.

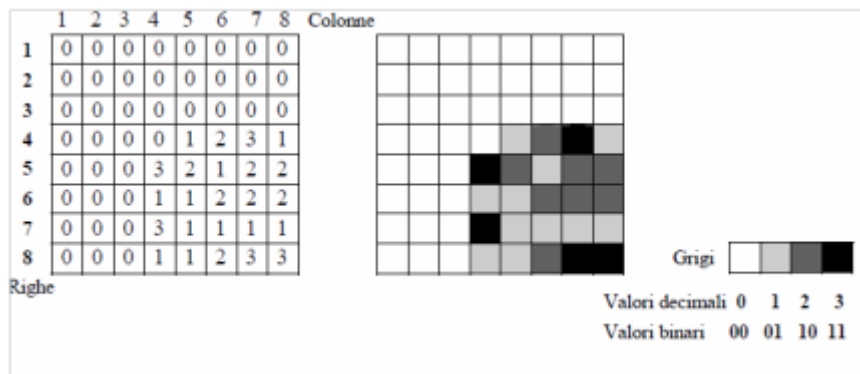


Figura 5: Visualizzazione di una immagine numerica rappresentata in matrice di 8x8 bit

L'intervallo di numeri CT prende il nome di larghezza della finestra o WW (Window Width), la quale rappresenta il numero dei coefficienti densitometrici da visualizzare, mentre il centro della finestra o WC (Window Center) è il numero CT che corrisponde al livello medio di grigio rispetto al quale tali valori sono centrati (fig.64).

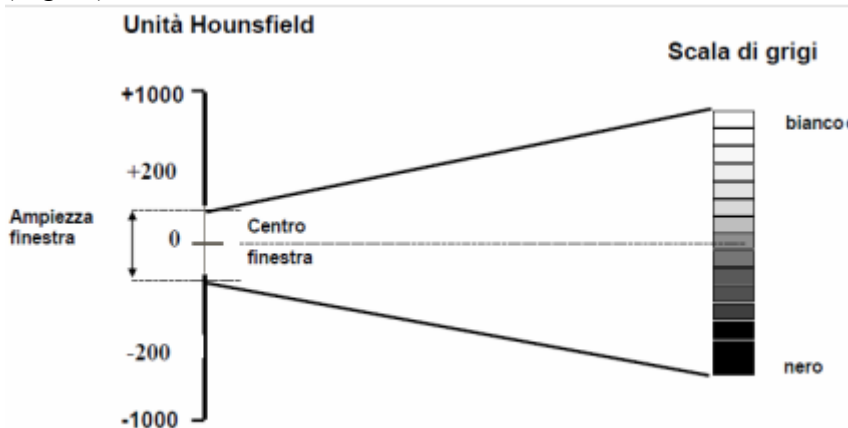


Figura 6: Operazione di posizionamento del "livello" e della "finestra" in CT

Riducendo l'ampiezza della finestra, ad esempio volendo rappresentare i CT numbers (Medical Dictionary)^[11] da +200 a -200 sempre con 64 tonalità, ognuna di queste comprenderà circa 6 numeri CT, mentre tutti i valori sopra a +200 e sotto -200 saranno rispettivamente completamente bianchi e completamente neri.

Per quanto si è detto, è chiaro che diminuendo l'ampiezza della finestra, il contrasto visivo corrispondente a due numeri CT posti all'interno della finestra aumenterà, consentendo di selezionare e meglio evidenziare piccole differenze densitometriche dei tessuti intorno al valore centrale 0 dell'acqua con esclusione di tutti gli altri. Se poi si alza o abbassa il centro della finestra non lasciandolo

al valore 0, ma per esempio ad un altro valore di tessuto di interesse (es.+30), nell'intorno di questo valore la diagnosi può essere più precisa. Numero medio e finestra sono quindi scelti in base ai tipi di tessuto che si vogliono esaminare, e possono essere variati in qualsiasi momento. La tecnica che consente di visualizzare due intervalli separati di numeri CT e quella che utilizza una doppia finestra sulla stessa immagine.

1.2.7 Lettura/scrittura file DICOM

Uno degli aspetti dell'esercitazione con Matlab (si veda il paragrafo 2.3.10), è stato l'utilizzo di apposite librerie come "dicomread" e "dicominfo", in grado trattare i file DICOM e leggerne i vari contenuti. Questo tipo di funzionalità vengono implementate anche in ambiente Java e si sono rivelate molto utili nella creazione del mio software.

1.2.8 Trattamento e filtraggio delle immagini

Aspetto fondamentale dell'analisi d'immagini è il trattamento e filtraggio delle immagini. Come descritto nei paragrafi precedenti, la mia attività ha previsto prima una fase di acquisizione delle conoscenze teoriche, e la successiva sperimentazione pratica tramite gli strumenti Matlab ed ImageJ. Così facendo, mi è stato possibile capire come tanti concetti e procedure utilizzate nell'indagine (apparentemente incorrelate tra loro), siano di fondamentale importanza per la buona riuscita dell'indagine stessa.

1.2.9 I campi dell'header file DICOM utilizzati

Uno degli scopi primari dello studio del formato DICOM è stato proprio il prendere familiarità con i campi dell'header file, per poter riuscire ad individuare in quali campi fossero presenti le informazioni necessarie al calcolo delle curve caratteristiche nello studio perfusionale. Poiché il software dedicato fornito da Philips ha generato diverse criticità, e poiché i file DICOM creati dalla CT Philips non presentavano diversi dei campi di cui si necessitava(si veda il paragrafo sulle criticità), ecco che lo studio approfondito in

tal senso è diventato prioritario. Tra tutti vari campi presenti nell'header file (si veda il paragrafo sullo standard DICOM), la nostra attenzione si è concentrata per lo più su quei campi contenenti l'esatta tempistica d'acquisizione della *slice* contenuta nel file, la relativa numerazione (posizione all'interno dell'analisi CT) e posizione del lettino su cui si trovava il paziente.

Andando avanti negli studi, si è potuto notare che alcuni parametri dati come fissi (prestabiliti inizialmente e mai più modificati da parte del costruttore) come ad esempio lo spessore della *slice* o lo step del lettino, in realtà venivano variati (sia pur di poco) nel susseguirsi degli aggiornamenti del software proprietario. Di queste problematiche si sta tutt'ora occupando Mattia Gardini.

1.2.10 Tracciamento manuale ROI

Come accennato nel paragrafo riguardante le ROI, il tracciamento delle stesse può avvenire in due modalità. Manuale ed Automatico. Il tracciamento manuale necessita della presenza e dell'abilità dell'utente, che utilizzando uno o più strumenti messi a disposizione dal software di elaborazione d'immagini, traccia manualmente la zona di interesse, su cui andranno poi fatte tutte le analisi, elaborazioni e misurazioni. Se ad un primo approccio la cosa può sembrare banale (come nel caso comunissimo in cui si trova l'utente medio alle prese col "ritagliare" una persona da un'immagine magari scattata ad una festa), in realtà non lo è, specie se le zone di interesse sono più di una all'interno della stessa immagine, se le immagini su cui tracciare le ROI diventano diverse decine, e se l'area da individuare non è affatto chiara o ben definita (come nel caso della CT). Fino ad oggi infatti (in ambito radiologico) per poter operare delle misurazioni di tipo quantitativo e qualitativo, ci si è affidati all'abilità dello specialista radiologo, alle prestazioni delle periferiche video (come i monitor ad alta definizione) ed al fattore tempo. Il fattore tempo, quando si tratta di intervenire su un soggetto colpito da patologia a qualsivoglia stadio, diventa uno dei vincoli fondamentali. Ecco che il tracciamento manuale da parte del medico, di diverse ROI su ciascuna delle decine di immagini raffiguranti "*Slice*" (fette dell'area osservata, componenti l'esame CT), diventa impegnativo

e vincola lo specialista ad un carico di lavoro consistente. Il fattore umano, seppur godendo di tante sfaccettature positive, soffre anche dell'errore soggettivo. Evento del tutto aleatorio, che talvolta può fare la differenza tra il diagnosticare qualcosa perché si è riusciti ad individuare un'anomalia o meno. Tutto ciò può anche significare sottoporre il paziente a dosi non appropriate di terapia o addirittura rischiare il decesso.

Per questo, e per altri aspetti critici della pratica fin'ora utilizzata, si necessita di automatizzare il processo, cercando di renderlo quanto più possibile standard ed automatizzato. Ecco che scende in campo il tracciamento automatico delle ROI.

1.2.11 Tracciamento automatico ROI

Aspetto fondamentale nel processo di automatizzazione, il tracciamento automatico delle ROI non deve necessitare dell'assistenza dell'utente, e nemmeno quindi del suo fattore decisionale sul cosa individuare e perché. Questo tipo di strumento è presente nei più comuni software di elaborazione d'immagini; può quindi sembrare banale affrontarlo così seriamente. In realtà, come detto nel paragrafo precedente, nell'ambito in cui ci troviamo l'accuratezza dello strumento che va ad operare il tracciamento di una ROI è di fondamentale importanza. Per rendere l'idea, basti pensare all'utilizzo dello strumento spesso chiamato "wizard" (mago) in grado di tracciare quasi immediatamente i contorni di una parte dell'immagine secondo i suoi parametri valutativi (ad esempio l'abito di una persona, se questo è tutto dello stesso colore). Spesso, se andiamo ad osservare la porzione individuata (aiutandoci col fattore di zoom) notiamo come i bordi di questa porzione non siano esattamente quelli desiderati (quindi ben definiti), ma all'incirca approssimabili a ciò che desideravamo. Ecco spiegato il nocciolo della questione. Un algoritmo in grado di individuare perfettamente un'area all'interno di un'immagine richiede del carico computazionale per niente banale. Quando si tratta di individuare delle lesioni all'interno delle immagini CT, la cosa si complica. Poiché (all'interno dell'immagine) aree di diversa gradazione di grigio rappresentano aree a diversa densità, e poiché in base alla loro forma o disposizione permettono di

identificare diverse patologie o addirittura banali artefatti ottici, ecco che affidare tutto ciò ad un algoritmo (e necessitare che tutto ciò sia perfettamente in grado di decidere cosa è “in” e cosa è “out”), è ben altra cosa che selezionare un abito da una foto normalissima. Alla base del progetto in svolgimento vi è anche questo aspetto.

1.2.12 La PET

La Tomografia a Emissione di Positroni (Positron Emission Tomography) è una tecnica di medicina nucleare e di diagnostica per immagini. Se grazie alla TC è possibile fornire delle informazioni di tipo morfologico dell'area (anatomica) osservata, grazie alla PET è possibile fornire informazioni di tipo fisiologico dell'area stessa. Dal punto di vista della metodica d'indagine, vediamo come esse siano molto simili, con la sostanziale differenza che la TC è un'indagine di tipo passivo (l'immagine creata è frutto dell'osservazione del flusso del mezzo di contrasto attraverso i vasi, gli organi etc.), mentre la PET è di tipo attivo (il mezzo di contrasto iniettato al paziente è ugualmente a base di zuccheri ma ha in aggiunta una parte radioattiva che grazie alle sue caratteristiche fisiologiche è in grado di irradiare) quindi si ha l'osservazione dell'irraggiamento dovuto al mezzo di contrasto.

La sovrapposizione tra immagini TC e PET permette di valutare il grado di attività (metabolismo) della zona osservata (e delle eventuali lesioni tumorali ivi presenti). Uno degli strumenti messi a disposizione dal nostro software è in grado di operare questa sovrapposizione, e le sue caratteristiche verranno discusse in seguito.

1.3 Elaborazione delle Immagini

L'elaborazione delle immagini (Image Processing) è una delle branche dell'elaborazione dei segnali. In questo ambiente l'input è un'immagine acquisita o un fotogramma video, e l'output è il risultato ottenuto trattando l'immagine (o il fotogramma video) con diverse tecniche standard di elaborazione, o misure fatte su di essa. L'avvento dell'era digitale ha permesso un enorme salto qualitativo e quantitativo del trattamento delle immagini, rendendo quindi più ampio il ventaglio di valutazioni possibili. Poiché il passaggio dall'analogico (vecchia macchina fotografica a pellicola) al digitale (fotocamere e altri strumenti d'acquisizione), ha richiesto una discretizzazione delle grandezze; ecco che il fare assumere un aspetto numerico alle immagini è diventato prioritario. In generale quindi, quando si parla di elaborazione delle immagini, ci si riferisce ad immagini digitali (o digitalizzate). Poiché l'immagine può presentare imperfezioni e/o artefatti dovuti ad innumerevoli motivi, per poter elaborare in maniera adeguata un'immagine è necessario procedere in modo accurato ed applicando concetti appresi in ambito teorico.

1.3.1 L'immagine e la sua formazione

L'immagine ha da sempre rappresentato un'ardua sfida alle esigenze dell'uomo nel catturare ciò che riesce a vedere grazie alla vista permessa dai suoi occhi. La visione è resa possibile dall'interazione tra un "raggio" elettromagnetico emesso da una sorgente (come ad esempio il sole) e la materia. Sin dagli albori l'uomo ha sempre cercato in tutti i modi di rappresentare tramite il disegno il mondo che lo circonda.



Figura 4: Prime forme di rappresentazione dal vero

Si è giunti quindi fino ai pittori, artisti in grado di catturare l'attimo su tela, raffigurando volti, paesaggi e quant'altro.

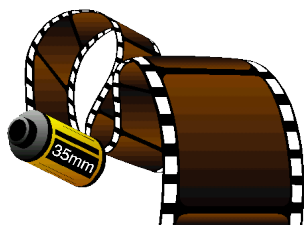


Figura 7: Pellicola fotografica 35mm

Ecco che il concetto del catturare l'attimo su un supporto (in questo caso semplice come la tela, o un intonaco) diventa di fondamentale importanza. Andando velocemente avanti nei secoli, ci rendiamo conto di come le esigenze in merito da parte dell'uomo siano sempre le stesse, e come esso cerchi di rendere quanto più veritiera la cattura del tanto agognato attimo. Ecco nascere la tecnica fotografica e cinematografica. Senza scendere tanto nei particolari, è bene notare come tutto si riduca macroscopicamente nel catturare l'attimo (o sequenze di attimi, dunque fotogrammi nel caso video) su un supporto che in questo caso non è più la tela ma la pellicola. Il concetto di cattura dell'attimo è talmente tanto potente e radicato nella natura dell'uomo, che tutt'ora alcune tribù si rifiutano di essere fotografate, perché credono che questo rubi loro l'anima. Tutto ciò è catalogabile come acquisizione di tipo analogico, ed è alla base di quanto avviene sotto ai nostri occhi già da qualche anno. Stiamo parlando dell'era digitale. Anche qui, le esigenze dell'uomo sono identiche a quelle precedenti, solo che grazie al progresso tecnologico, ad esse se ne aggiungono altre, magari di carattere più scientifico (come la fotometria e l'utilizzo dell'immagine nel campo diagnostico). Come d'altronde è inevitabile, il passaggio da un ambiente analogico ad uno digitale richiede una discretizzazione delle grandezze in questione, e con essa la nascita di nuove tecniche di riproduzione dell'informazione acquisita e relativo trattamento. Grazie all'avvento dell'era digitale, i confini di cattura dell'attimo non riguardano più solo la gamma dello spettro elettromagnetico percepibile dall'occhio umano (banda del visibile),

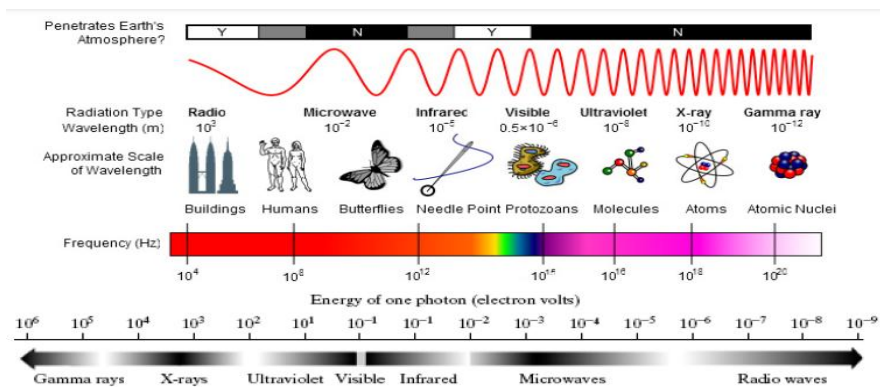


Figura 8: Spettro elettromagnetico e alcune informazioni sulle lunghezze d'onda

ma come nel caso CT, ci è permessa l'osservazione dell'interazione tra radiazioni a frequenze affatto percepibili dall'occhio umano e materiali comunemente osservabili. Tutto ciò, è possibile grazie a tecniche di ricostruzione dell'immagine (visibile dall'uomo), partendo da eventi appartenenti a bande che per l'occhio umano non sono percepibili (come per esempio l'infrarosso e i raggi X). In definitiva, l'immagine non è altro che uno dei tanti metodi con cui è possibile rappresentare la realtà o (come nel caso dell'elaborazione delle immagini) i dati ottenuti o acquisiti.

1.3.2 L'immagine come matrice bidimensionale di pixel

Come accennato nel paragrafo precedente, il passaggio da un'immagine di tipo analogico ad una digitale ha richiesto la reinterpretazione spaziale dell'oggetto immagine, per poterlo definire accuratamente, e trattarne i vari aspetti che fisiologicamente si concatenano, come l'acquisizione grazie ai sensori ottici, il suo salvataggio e riproduzione su appositi dispositivi come i monitor. Si arriva quindi a schematizzare l'immagine come matrice bidimensionale, che come tale è possibile suddividere in vettori riga e

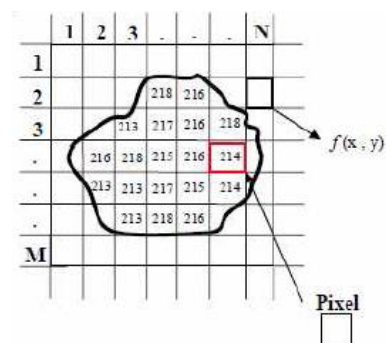


Figura 9: esempio immagine come matrice di pixel e relativi valori

vettori colonna (Gonzalez, 2001). Affrontando la questione con questo tipo di approccio è possibile utilizzare tutti gli strumenti di calcolo che l'algebra matriciale ci mette a disposizione. Ogni singolo punto della matrice diventa facilmente ed univocamente identificabile

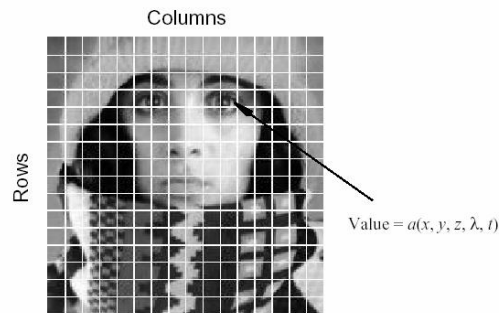


Figura 10: esempio immagine come matrice di pixel. Visualizzazione valori

,oltre che un elemento trattabile dinamicamente e separatamente dagli altri punti della matrice stessa. L'unione di tutte queste (e tante altre) proprietà dell'immagine come matrice di punti, ha reso vincente l'approccio digitale a tutto il mondo dell'ottica. Nell'immagine a fianco viene descritto come un sensore CCD (per approfondimenti si veda il corso di Bioimmagini e Visione) (Bevilacqua, 2011) ^[2]realizza il campionamento dell'immagine acquisita. L'immagine digitale monocromatica quindi, è una matrice $Y(i,j)$ di valori discreti di intensità luminosa (livelli di grigio) costituita da $M \times N$ pixel (picture elements) ciascuno dei quali ha un valore appartenente all'intervallo $[0, 2^b-1]$. Per esempio, con 8 bit si ha la possibilità di rappresentare un numero di livelli (256) tale da consentire una discriminazione dei grigi accettabile nella maggior parte delle applicazioni, in quanto abbastanza prossima a quella dell'occhio umano.

1.3.3 RGB e Gray Scale

RGB è il nome di un modello di colori le cui specifiche sono state descritte nel 1931 dalla CIE (Commission internationale de l'éclairage). Tale modello di colori è di tipo additivo e si basa sui tre colori rosso (Red), verde (Green)

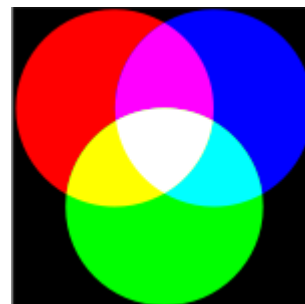


Figura 11: Colori (RGB) e relativa sovrapposizione

e blu (Blue), da cui appunto il nome RGB, da non confondere con i colori primari sottrattivi giallo, ciano e magenta. Un'immagine può infatti essere scomposta, attraverso filtri o altre tecniche, in questi colori base che miscelati tra loro danno quasi tutto lo spettro dei colori visibili, con l'eccezione delle porpore.

L'RGB è un modello additivo: unendo i tre colori con la loro intensità massima si ottiene il bianco (tutta la luce viene riflessa). La combinazione delle coppie di colori dà il ciano, il magenta e il giallo. Nell'immagine che segue, è possibile osservare l'effetto della tricromia additiva di un'immagine reale.



Figura 12: Immagine originale e scomposizione nei tre canali RGB

Gray Scale (scala di grigi)

Un'immagine in scala di grigi (o livelli di grigio), non è altro che un'immagine in cui gli unici colori presenti sono le sfumature di grigio (si veda la figura a fine paragrafo).



Figura 13: Immagine in gray scale

L'aspetto che differenzia un'immagine in scala di grigi da un'immagine a colori (spazio RGB), è la mancata necessità di fornire ad ogni singolo elemento grafico (pixel), tre valori di intensità (una per ogni colore R-

G-B), bensì solamente uno. In effetti, si può notare come nel "grigio", i componenti Rosso, Verde e Blu, abbiano la stessa intensità (spazio RGB), risulta quindi evidente come basti fornire solamente un valore d'intensità ad ogni pixel.

Spesso la scala di grigi viene memorizzata come un intero a 8-bit, fornendo così $2^8=256$ tonalità diverse di grigio possibili, dal nero al bianco.



Figura 14: scala di grigi

1.3.4 Filtri

In Image Processing i filtri di elaborazione vengono utilizzati principalmente per attenuare o addirittura eliminare determinati aspetti dell'immagine grezza, cercando di migliorarla o renderla più appropriata allo scopo prefissato. Si agisce quindi smussando determinate componenti frequenziali presenti nell'immagine. Questo tipo di attività è presente in una qualsivoglia elaborazione di segnale. Un'immagine può essere filtrata nel dominio delle frequenze o nello spazio. Il primo caso riguarda il dominio delle frequenze. In questo campo si procede moltiplicando l'immagine con una funzione rappresentante il filtro di frequenze e successivamente ri-trasformando il risultato ottenuto nello spazio. Per esempio, una semplice funzione passa-basso, può valere 1 per frequenze inferiori alla frequenza di taglio fissata e 0 diversamente.

Nel dominio spaziale il processo corrispondente è la convoluzione tra l'immagine (caratterizzata dalle relative coordinate spaziali) ed un filtro avente funzione di risposta impulsiva $h(i,j)$. Tutto ciò può essere scritto come

$$g(i, j) = h(i, j) \odot f(i, j)$$

Esistono diverse categorie di filtri, a seconda dell'elaborazione da effettuare. Alcuni di essi sono il "Mean Filter", "Median Filter", "Gaussian Smoothing", "Filtro di contrasto" etc.

Dietro ciascuno di essi vi è un'accurata trattazione di tipo teorico-matematico che giustifica l'importanza dei filtri, nell'attività di

Image Processing. Per approfondimenti, si faccia riferimento al corso di “Bioimmagini e Visione”.

1.3.5 ROI

La ROI (Region of Interest), è quell’area d’interesse dell’immagine sulla quale si vogliono effettuare dei rilevamenti, misure, elaborazioni. Essa è delimitata da un indicatore visivo che può essere di vario tipo (es.

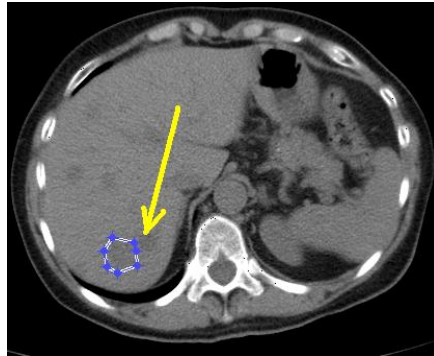


Figura 15: CT epatica con tracciamento ROI su metastasi

poligonale, circolare, ellittico).

Il “tracciamento” della ROI

può avvenire tramite

l’interazione con l’utente tramite il mouse, o automaticamente fornendo al software le coordinate dell’area a cui siamo interessati.

L’utilità della ROI nasce dal fatto che, delimitando l’area a cui siamo interessati, tutti i nostri sforzi di tipo “osservativo” e computazionali sono concentrati solo su quell’area, tralasciando la restante parte dell’immagine. Tutto ciò ha un grossissimo impatto sul carico computazionale a cui sottoporre il calcolatore, rendendo più “veloce” l’attività da svolgere.

Il tracciamento semiautomatico ed automatico delle ROI è uno degli aspetti dell’attività svolta e che si svolgerà nel contesto del progetto PERFECT.

1.3.6 Gradiente di un’immagine

Il gradiente dell’immagine (o gradiente della morfologia dell’immagine) è il risultato del calcolo della dilatazione in scala di grigi e la relativa erosione tramite un kernel K :

$$\begin{aligned} \text{Gradient}(I) &= \frac{1}{2}(D_G(I, K) - E_G(I, K)) \\ &= \frac{1}{2}[(I \oplus_g K) - (I \ominus_g K)] \end{aligned}$$

where \oplus_g and \ominus_g are grayscale dilation and erosion

Il calcolo del gradiente è molto simile al rilevamento standard dei bordi ed è bene distinguere tra gradiente esterno ed interno.

1.3.7 Edge Detectors

Lo scopo del riconoscimento dei contorni (edge detection) è marcare i punti di un'immagine digitale in cui l'intensità luminosa cambia

bruscamente. Bruschi cambiamenti delle proprietà di un'immagine sono di solito il sintomo di eventi o cambiamenti importanti del mondo fisico di cui le immagini sono la rappresentazione. Questi cambiamenti possono essere ad esempio: discontinuità della profondità, discontinuità

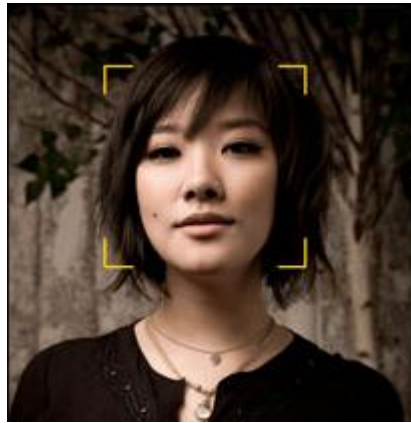


Figura 16: Immagine Originale campione

dell'orientamento delle superfici, modifica delle proprietà dei materiali, e variazioni dell'illuminazione proveniente dall'ambiente circostante. Il riconoscimento dei contorni è un campo di ricerca del trattamento delle immagini e della visione artificiale, in particolare della branca del riconoscimento delle caratteristiche (feature extraction). L'operazione di riconoscimento dei contorni genera immagini contenenti molte meno informazioni rispetto a quelle originali, poiché elimina la maggior parte dei dettagli non rilevanti al fine dell'individuazione dei contorni, conservando invece le informazioni essenziali per descrivere la forma e le caratteristiche strutturali e geometriche degli oggetti rappresentati.

Esistono molti metodi per riconoscere i contorni principalmente classificabili secondo due categorie: metodi basati sulla ricerca (search-based) e metodi basati sull'attraversamento dello zero (zero-crossing). I metodi basati sulla ricerca riconoscono i contorni cercando i massimi ed i minimi della derivata del primo ordine dell'immagine, di solito cercando la direzione in cui si ha il massimo gradiente locale. I metodi zero-crossing cercano i punti in cui la derivata del secondo ordine passa per lo zero, solitamente la funzione laplaciana o un'espressione differenziale di una funzione non-lineare. Per approfondimenti, si faccia riferimento al corso di Bioimmagini e Visione.



Figura 17: Immagine con effetto Edge Detectors

1.3.8 Istogramma e Sogliatura

L'istogramma di un'immagine (o di una porzione di essa) è la rappresentazione sotto forma di istogramma, dunque visiva, dei valori in scala (RGB e di grigi). Nell'istogramma di un'immagine in scala di grigi, sulle ascisse troviamo tutti i "toni" della scala e sulle ordinate il numero delle occorrenze dei toni nell'area di interesse. Tramite questo strumento è possibile fare diversi studi e considerazioni.

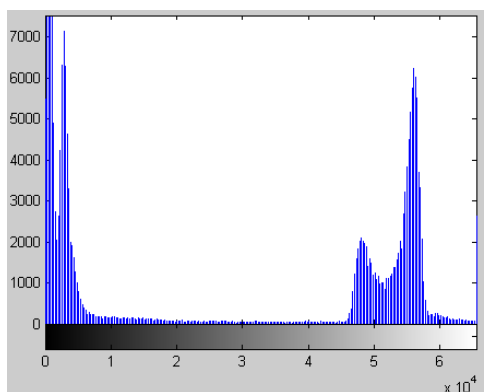


Figura 18: Esempio istogramma

Tutto ciò rende l'istogramma uno strumento fondamentale.

La sogliatura (anche detta in inglese *thresholding*) nell'elaborazione digitale delle immagini è un semplice metodo per segmentare

un'immagine. Da un'immagine a livelli di grigio, la sogliatura restituisce un'immagine binaria.

Durante un processo di sogliatura, singoli pixel dell'immagine sono catalogati come "pixel oggetto" se il loro valore è maggiore di una certa soglia e come "pixel di sfondo" se il valore è sotto la soglia. Solitamente l'immagine binaria in uscita ha valore pari a "1" dove è presente l'oggetto e pari a "0" per lo sfondo, ottenendo quindi un'immagine in bianco e nero. Ovvero per un'immagine originale a livelli di grigio sogliata $I_{originale}(x)$ la sogliatura si ottiene così:

$$I_{binaria}(x) = \begin{cases} 0 & \text{se } I_{originale}(x) < \text{soglia} \\ 1 & \text{se } I_{originale}(x) \geq \text{soglia} \end{cases}$$

dove x rappresenta il digital number (DN) ovvero il valore del pixel nell'immagine. Il parametro chiave in una sogliatura è la scelta del valore di soglia. Esistono diverse metodologie per la scelta automatica di questo valore, oppure è possibile sceglierlo a mano. L'effetto della sogliatura è evidente nelle immagini seguenti.



Figura 19: Immagine prima (sinistra) e dopo la sogliatura

1.3.9 Maschere

La maschera di un'immagine è un'immagine binaria costituita esclusivamente da "1" e "0". All'interno dell'immagine rappresentante una maschera si avrà la presenza di uni, se quella area è d'interesse, mentre nella restante parte si avranno degli zeri. E' quindi evidente come i pixel di quest'immagine assumeranno un colore bianco o del tutto nero. Se una maschera binaria viene applicata ad un'altra immagine di qualsiasi tipo, l'effetto sarà quello di ottenere un'immagine a sfondo nero e l'area di intersezione tra immagine da elaborare e maschera assolutamente inalterata. In modo intuitivo, si ha lo stesso effetto ottenuto praticando un'apertura (di forma a piacere) su di un foglio di cartoncino nero, appoggiandolo poi su una foto. Ecco che l'unica parte visibile della foto sottostante, risulta essere solo quella in cui il cartoncino è mancante. La procedura di applicazione di una maschera ad un'immagine, può avvenire mediante moltiplicazione dei rispettivi pixel o a livello logico (AND logico). Nelle immagini seguenti, è possibile osservare, nell'ordine, un'immagine originale, la maschera che si desidera utilizzare e il risultato dell'applicazione della maschera.

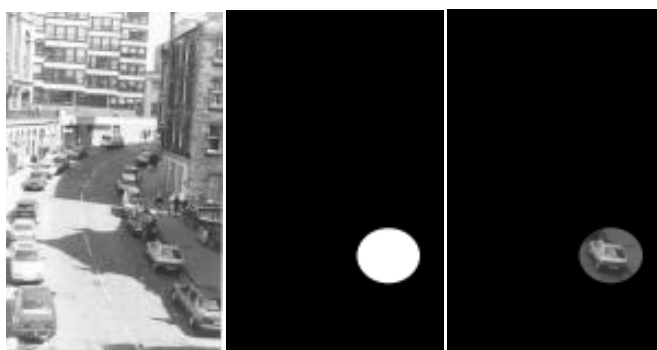


Figura 20: Immagine esempio (sinistra), maschera, dato estratto con la maschera

1.3.10 Image Processing in Matlab

Una delle prime esercitazioni che svolsi quando frequentai l'attività di tirocinio si basò sull'applicazione pratica dei concetti teorici dell'elaborazione d'immagine. Il fatto di poter utilizzare uno strumento come Matlab (Matlab) ^[9] per l'elaborazione delle immagini non era per me nemmeno immaginabile. Di certo era più facile associare l'elaborazione d'immagine a software di fotoritocco di più comune utilizzo. La potenza degli strumenti che Matlab mette a disposizione dell'utente (le librerie), il loro facile utilizzo, mi ha permesso di toccare con mano quanto possa essere affascinante l'elaborazione delle immagini vista dal retroscena basato sul calcolo. Facendo riferimento all'help di Matlab e alla documentazione trovata sul web, ho scoperto quanto fosse ricco e ampio il ventaglio di librerie pronte all'uso ed utilizzabili in modo dinamico. Ve ne sono per ogni esigenza di elaborazione, incarnando quindi gli aspetti teorici su cui si basa l'elaborazione delle immagini, tra cui quelli descritti nei paragrafi precedenti. Purtroppo però, poiché una delle caratteristiche fondamentali del nuovo software che sarei andato a sviluppare per la tesi era l'essere free, mi ha obbligato ad abbandonare questo ambiente per sceglierne un altro (Sun)^[16].

1.4 Attività Integrative svolte

Il periodo di apprendimento delle tecniche, nozioni teoriche e sviluppo del materiale oggetto della mia tesi, non è stato affatto scevro di numerose attività collaterali integrative, che mi hanno permesso di verificare direttamente sul campo quanto queste fossero efficaci ed utili in un contesto reale e di alto livello.

Avere avuto la possibilità di vivere e lavorare quotidianamente all'interno di un gruppo fortemente affiatato e di alto livello, altro non ha fatto che accendere in me la passione per quello che da sempre è stato un divertimento. Grazie all'invito del prof. Bevilacqua, ho avuto la possibilità di partecipare a due congressi:

- **“Visione per competere”**- Innovare i prodotti e i processi con le nuove tecnologie di visione artificiale - svoltosi nell'Aprile 2011 presso “c/o Soilmec – Gruppo Trevi” a Cesena.
- 2nd International Conference - **“Translational Research in Oncology: a New Approach to Personalized Medicine”** – svoltasi nel Maggio 2012 a Forlì e Meldola in occasione del conferimento dello status di IRCCS all'IRST di Meldola.



Figura 21: Foto ricordo al termine del "Translational Research in Oncology".

Da sinistra verso destra: Josè Falla, Marco Aldo Prattichizzo, Mattia Gardini, Silvia Malavasi, Ilaria Fantigrossi.

Parallelamente al raggiungimento dei target fissati con il prof. riguardo all'attività prima di tirocinio curriculare e poi di tirocinio per tesi, ho avuto la possibilità di assistere ad una seduta di manutenzione ordinaria



Figura 22: Seduta Manutenzione ordinaria TC Philips Presso IRST. Da sinistra verso destra: Mattia Gardini, Silvia Malavasi, Marco Prattichizzo, Josè Falla

presente in struttura, e di partecipare ad un meeting con i responsabili commerciali e di sviluppo e ricerca della Philips presso IRST. Durante la seduta di manutenzione, grazie alla gentilezza e alla disponibilità del tecnico manutentore, ho potuto assistere alle varie fasi di smontaggio, verifica/sostituzione delle varie parti della macchina, commentando contemporaneamente con lui i vari componenti, le tecniche utilizzate e i principi fisici su cui si basano. A questa seduta hanno partecipato anche altri miei colleghi e amici presenti in foto. Partecipando all'incontro con i responsabili europei della Philips, ho potuto apprendere come si svolgono questo tipo di incontri e di capire come avviene la comunicazione tra produttori/sviluppatori ed utente finale, il tutto si è ovviamente svolto in inglese e ho cercato di imparare quanto più possibile sulle dinamiche in gioco.

Sin dai primi giorni di attività in struttura è stato necessario fare un

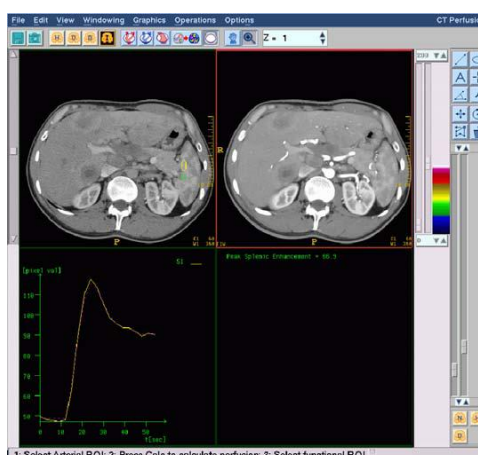


Figura 23: Software Philips per lo studio perfusionale

quadro generale sui software in grado di supportare l'attività del medico nello studio perfusionale. Mi è stato quindi richiesto di studiare, analizzare e testare i software già presenti in struttura e che per diversi aspetti non andavano a soddisfare le esigenze di indagine del medico.

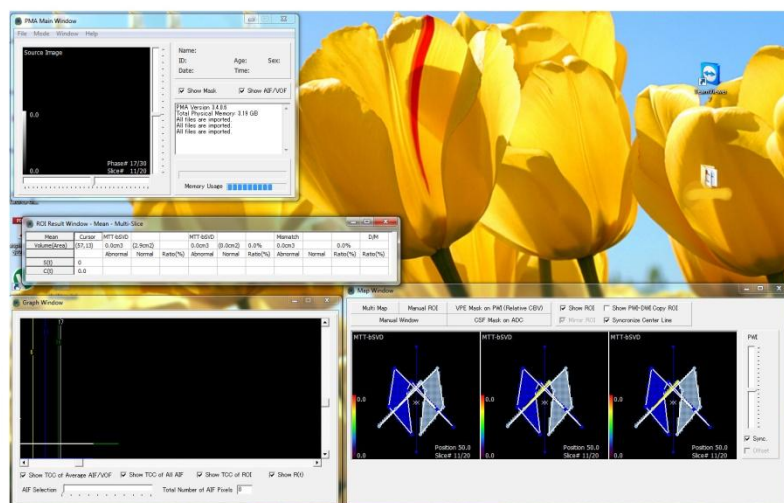


Figura 24: Screenshot di una sessione di PMA

Questa attività apparentemente semplice e veloce da fare, si è rivelata invece l'apertura di una sorta di grosso vaso di Pandora. Ciascuno dei diversi aspetti da analizzare e testare richiedeva abilità e conoscenze diverse e multidisciplinari. Conoscenze che per buonissima parte ho dovuto acquisire studiando articoli scientifici e materiale didattico reperito da diverse fonti. Terminata una prima fase di studio e test di un primo software, mi è stato richiesto di fare la medesima cosa su un altro software che per motivi di licenza era utilizzabile solo da un terminale fisico residente in ARCES. Per ovviare a ciò e per ottimizzare l'utilizzo della macchina fisica, ho potuto svolgere le mie attività da remoto grazie al software "TeamViewer" (Viewer)^[17] - la soluzione All-In-One per controllo remoto, il tutto in notturna per non intralciare le

normali attività diurne di ricerca. Lo studio e la ricerca di altri software perfusionali presenti sul mercato si è allargata al web, ed ecco che oltre studiare altri Viewers di immagini DICOM, abbiamo potuto studiare e testare software sviluppati da altri gruppi di ricerca come il PMA (PMA)^[13] realizzato principalmente per lo studio di immagini cerebrali, quindi per diversi aspetti non utile al nostro scopo. Per necessità di progetto inoltre, è stato necessario acquisire tantissime altre conoscenze in elaborazione delle immagini, e su strumenti software come le librerie Matlab e software di elaborazione delle immagini come ImageJ utili ad applicare i concetti teorici precedentemente studiati.

Oltre all'elaborazione delle immagini, poiché il progetto ha richiesto la realizzazione di un tool di archiviazione e recupero dei dati degli studi perfusionali effettuati in struttura, ecco che su indicazione del prof. Bevilacqua i miei studi hanno compreso anche un'idea di base su

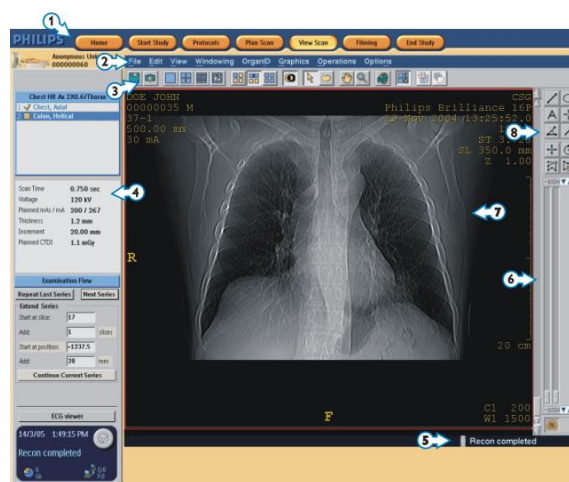


Figura 25: Philips DICOM Viewer

quali siano le tipologie di Forme Normali dei database relazionali, in modo da garantirne la qualità sin dalla fase di progetto.

Vivendo a stretto contatto sia con gli altri colleghi di progetto, sia con le figure professionali presenti in struttura, la mia attività ha compreso diverse occasioni in cui i concetti di “Problem Solving” (Wikipedia) e le basi del “Lean Thinking” appresi nei corsi di “Cultura d’impresa” e di “Gestione informatizzata dei processi” si sono rivelati fondamentali. E’ stato in oltre utile/necessario creare piccoli tool per agevolare lo svolgimento delle attività dei miei colleghi.

La realizzazione quindi del software oggetto della tesi ha permesso l'acquisizione e l'elaborazione di parte dei dati necessari per lo studio dei parametri perfusionali e la creazione di software per il calcolo delle relative mappe colorimetriche. Acquisizione che si è resa possibile (come citato in precedenza) grazie all'interazione diretta del Medico (si ringrazia il dott. Barone) con il software, garantendo una massima precisione e qualità dei dati prontamente formattati ed inviati agli altri componenti del progetto "PERFECT". In una particolare occasione ho potuto persino preparare una seduta di acquisizione dati con il medico in cui, oltre al tracciamento delle ROI (si veda nei capitoli precedenti), è stato possibile acquisire tutta l'attività di valutazione visiva per arrivare a tale tracciamento, grazie a dei software come Debut Video Capture Software (Debut)^[4]. Utilizzando al meglio le funzionalità messeci a disposizione dal Debut Video Capture Software, oltre al video è stato possibile registrare i commenti del medico durante il tracciamento e l'intervista che il sig. Mattia Gardini provvedeva a fargli, fornendo quindi al prof. Bevilacqua e a quanti ne necessitassero ogni tipo di informazione sulle fasi preliminari di studio delle immagini CT e relativo tracciamento delle ROI.

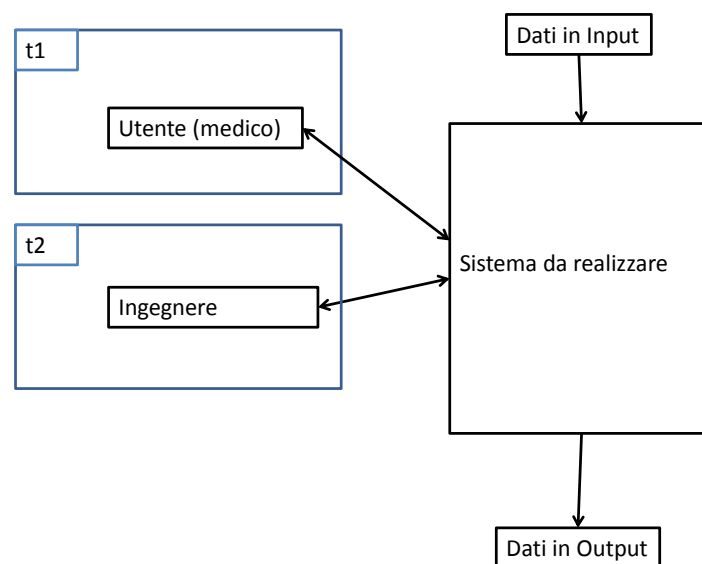
I dati e i risultati ottenuti dalla ricerca conseguente sono già stati oggetto di presentazioni in conferenze e lo saranno in pubblicazioni scientifiche su riviste internazionali del settore.

Capitolo 2

Il Progetto

Il software che ci accingevamo a sviluppare prevedeva sostanzialmente l'interazione tra l'utente ed un ambiente in grado di avere in input dei dati, elaborarli, e fornire in output dei risultati.

L'idea di base era la seguente:



Vediamo quindi come l'utente e l'ingegnere condividano l'interazione con il sistema in fasi diverse dell'indagine. Si vedrà più avanti che l'output del primo (il medico) è l'input per il secondo (l'ingegnere).

2.1 Fasi preliminari di progetto

Cercando di sfruttare al massimo quanto appreso nel corso di "Gestione informatizzata dei processi", sin dai primi giorni di attività in struttura, ho cercato di procedere alle fasi di progetto in modo standard e quanto più operativo possibile. Ecco che strumenti come il Gantt (Gantt project)^[4] sono stati di fondamentale importanza.

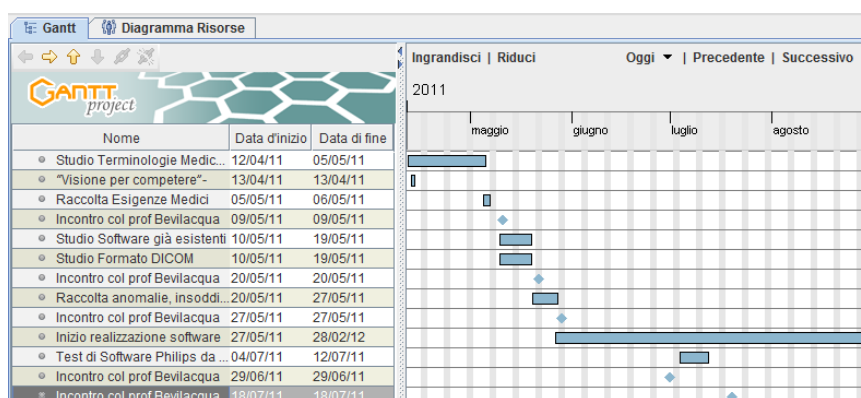
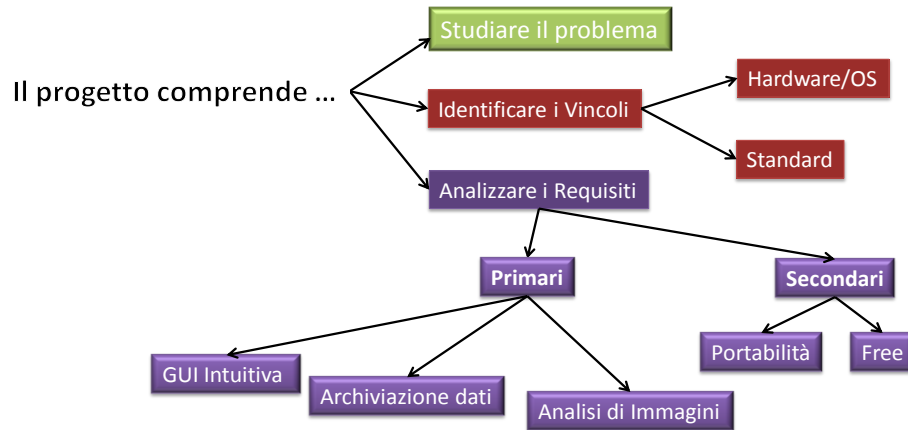


Figura 26: Stralcio di grafo di Gantt del progetto

Come si può notare dell'immagine precedente, le attività preliminari sono state molto intense ed hanno previsto anche studi/attività svolte in parallelo tra loro ed attentamente seguite dal prof. Bevilacqua (come si può notare dalle milestone presenti sul grafo).

Facendo una breve sintesi delle fasi fondamentali del progetto, possiamo cercare di schematizzarle così:



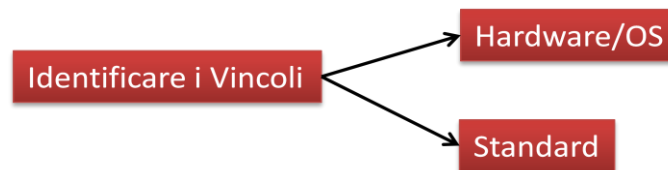
2.1.1 Raccolta delle problematiche/esigenze dei medici

Dopo una serie di incontri coi medici (a cui sono stati somministrate delle interviste di diversa natura da parte del sig. Mattia Gardini e da parte mia) per cercare di raccogliere quante più informazioni possibili sulle loro insoddisfazioni e desideri verso i nuovi strumenti che andavamo a sviluppare, sono stati fissati alcuni punti cardine di quello che sarebbe diventato uno strumento di acquisizione dati.



2.1.2 Identificare i Vincoli

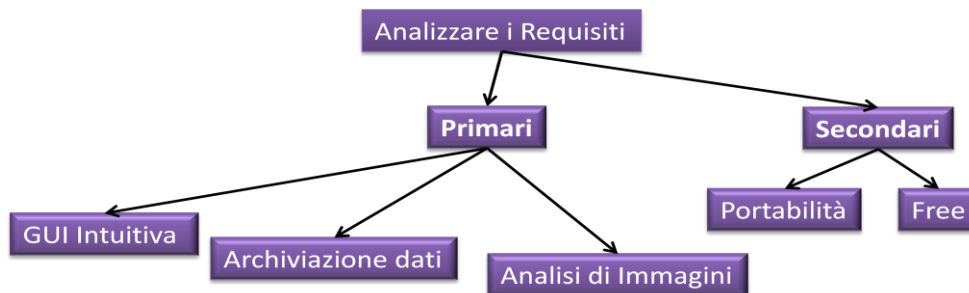
Dopo una prima fase di studio del problema, è stato necessario studiare bene i vincoli che si andavano ad avere nel contesto in cui ci accingevamo a lavorare. Fondamentalmente, i vincoli erano di tipo Hardware/SO e per quanto riguarda gli standard (protocolli esecutivi di analisi sul paziente, archiviazione dei dati, metodiche d'indagine).



Va da sé che non poteva essere l'UO ad adeguarsi al mio software cambiando tutte le macchine delle workstation, ma bensì il software ad adeguarsi ad esse. Eravamo quindi in presenza di macchine dalle prestazioni hardware e software di tipo standard, e con quelle necessitavamo di lavorare.

2.1.3 Analisi dei requisiti

I requisiti richiesti sono stati principalmente di due tipi. Primari e Secondari.



Alcuni tra i primari sono:

- GUI intuitiva
L'utente a cui è rivolta non deve avere particolari competenze informatiche e deve lavorare in un ambiente che gli permetta di operare velocemente e comodamente

- Archiviazione dei dati
Poiché tutte le operazioni previste comprendevano la gestione dei dati, creare uno standard di salvataggio degli stessi è stato fondamentale. Mi riferisco ai nomi da attribuire alle directory in cui si salveranno i dati e i nomi dei file creati.
- Analisi delle Immagini
Alla base del progetto PERFECT c'è proprio l'analizzare le immagini per poter offrire delle valutazioni quantitative. L'analisi delle immagini quindi è un aspetto fondamentale del software che si andrà a realizzare.

Alcuni dei secondari sono:

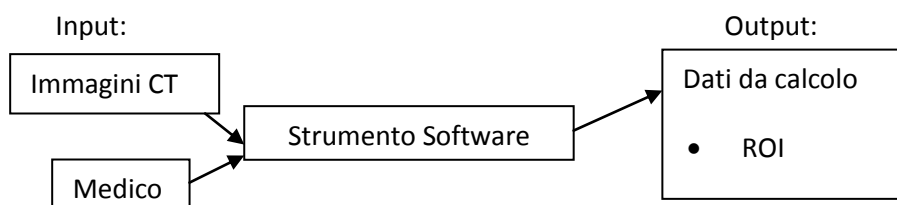
- Portabilità
Facendo riferimento ai vincoli, vediamo che si ha a che fare con hardware e software standard, quindi una caratteristica fondamentale del software che si andrà a realizzare è proprio la portabilità.
- Software free
Non si dispone di particolari licenze software. Il contesto in cui ci muoviamo deve necessariamente essere free. Soddisfare questo requisito ha richiesto un'attenta indagine di quanto offerto dal mercato attuale del software.

2.2 Dalle esigenze dei medici ad un primo modello di software

Riassumendo le informazioni raccolte, mi veniva richiesto di realizzare una sorta di script in grado di velocizzare un minimo l'attività del medico nel tracciare le ROI da acquisire. La scelta del linguaggio, il software da utilizzare come background a tutto ciò è stato lasciata a me. Questo primo piccolo strumento doveva essere pronto e funzionale entro pochi giorni da quel momento per poter essere presentato in seduta di meeting con il prof Gavelli e il dott. Barone. Oltre alle strette esigenze temporali e alla scelta dello strumento software da utilizzare, una voce importante è stata l'interfacciare una figura professionale come la mia con i medici. Ecco che quanto appreso nei miei studi universitari mi ha permesso

di trasformare quelle che erano inizialmente delle vere e proprie esigenze/problematiche in specifiche di progetto.

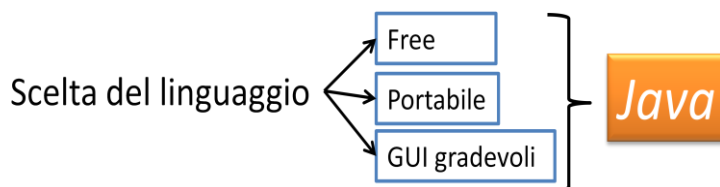
Nasceva quindi un primo modello realizzato in Matlab^[9] in grado di permettere all'utente di caricare immagini DICOM singole, modificare un minimo il windowing e di tracciarvi sopra la ROI di interesse. L'idea di base prevedeva un'architettura più o meno di questo tipo:



Si pensava quindi di avere un input dato da immagini CT acquisite in sede di analisi del paziente e dati inseriti dal medico tramite il tracciamento delle ROI. Un'elaborazione da parte del software e un output costituito dai dati estrapolati dalle due fasi precedenti.

2.3 Dal primo modello in Matlab al software in Java

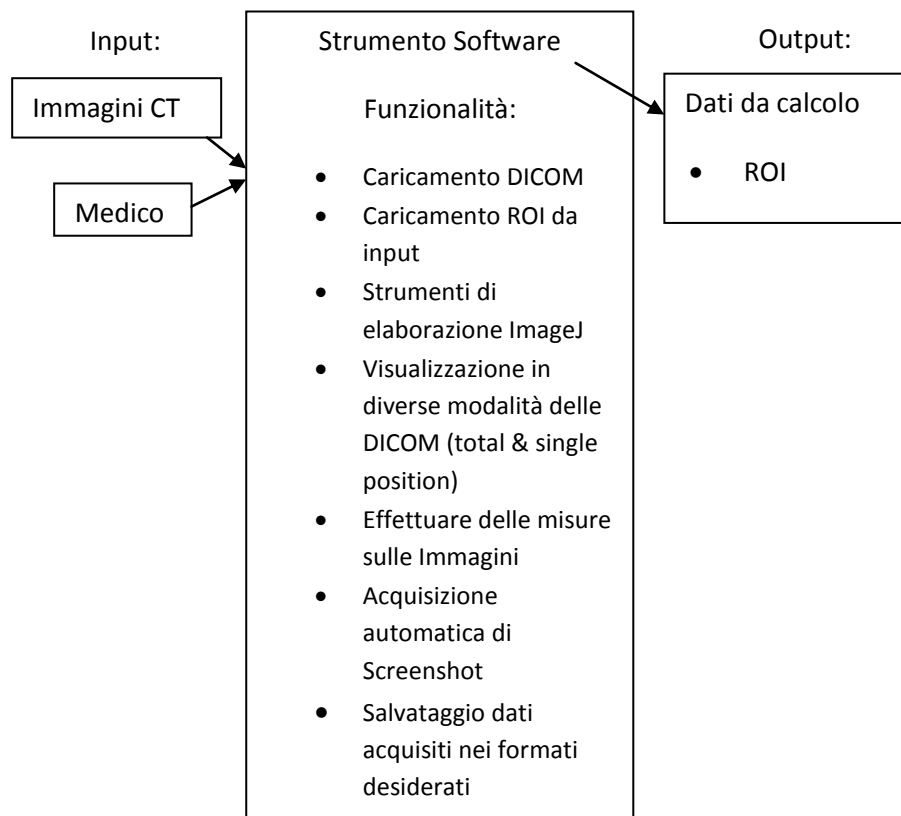
Poiché l'operazione di acquisizione che ci apprestavamo a fare richiedeva una massiccia interazione tra utente (medico) e software, ecco che studiare e raffinare un'interfaccia accuratamente realizzata su misura diventava una parte fondamentale del software. Basandomi sulle mie conoscenze e facendo una accurata ricerca sul web, ecco che il connubio tra ImageJ e l'ambiente Java mi è sembrato essere la soluzione ottimale al mio problema.



Da quel momento in poi, ho abbandonato lo sviluppo del modello in Matlab per dedicarmi ad un nuovo modello Java. Ad imporre

ulteriormente il passaggio da Matlab ad altro linguaggio è stato anche il dover soddisfare un requisito fondamentale del nuovo software, ovvero l'essere free. Vediamo quindi che Matlab non poteva soddisfare quanto detto.

Si è passati quindi dal semplice modello Matlab in grado di caricare singole DICOM e permettere il tracciamento delle ROI ad un primo modello Java in grado di permettere un approccio molto user-friendly e allo stesso tempo veloce nelle varie fasi dell'acquisizione. Nel susseguirsi delle versioni, il software è stato in grado di offrire sempre più funzionalità, partendo sempre dalle richieste fatte dai medici man mano che andavano a provare il software (la validazione dei risultati viene discussa nel capitolo 5).



Come si evince dal grafico precedente, il passaggio dal modello Matlab a quello Java ha permesso di fornire all'utente tutta una vasta gamma di funzionalità. Grazie all'interazione diretta con il medico in sede di raccolta esigenze, è stato possibile disporre i nuovi strumenti forniti, in modo da risultare (oltre che efficaci) comodi e veloci da usare.

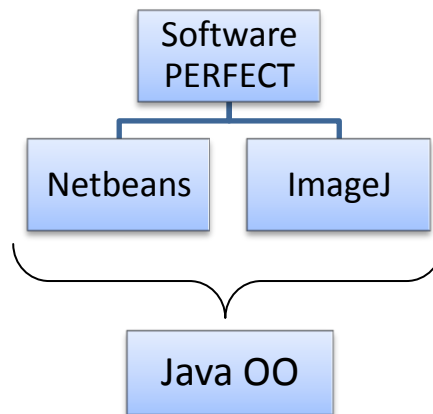
Man mano che si andavano a sviluppare le varie parti del software, era possibile notare come lo strumento non acquisiva più solo mera funzione di visualizzazione delle immagini, ma anche di calcolo e formattazione di questi ultimi. Bisogna inoltre evidenziare come la piega che si è voluta dare sin dall'inizio a questo nuovo software, sia stata quella di creare uno strumento assolutamente standard, facilmente portabile, comodo, veloce da utilizzare ed in grado di trattare tutte le tipologie di immagini che l'utente sarebbe stato interessato a trattare. Poiché l'indagine del Radiologo richiede la comparazione di immagini in specifiche posizioni (o sequenze di esse), è stato necessario affidare la gestione delle lettura delle informazioni delle singole immagini ad apposite librerie in grado di accedere all'Header File ed estrapolarne i dati desiderati (leggendo l'Header File si è certi di selezionare l'immagine desiderata scavalcando ogni tipo di errore di lettura dovuto a standard di salvataggio e nome file). Dapprima si è anche proposto ai medici la realizzazione di una funzionalità di apertura immagini su doppio monitor per permettere la comparazione di più immagini contemporaneamente, ma si è poi rivelato superfluo allo scopo del progetto e all'acquisizione dei dati.

2.4 L'architettura del Software in Java

Come anticipato in precedenza, in questo capitolo verranno illustrati i macroblocchi che vanno a comporre il software realizzato per la soluzione dei problemi evidenziati dai medici. I problemi in questione sono fondamentalmente il poter tracciare manualmente una ROI nelle modalità desiderate, poterne fare delle misure, esportarle nel formato desiderato e prima di tutto poter elaborare l'immagine per trarne le considerazioni necessarie al tracciamento della ROI stessa.

2.4.1 Strumenti software utilizzati

Attingendo a quanto avevo già studiato nei corsi, seguendo i consigli del prof. Bevilacqua e rielaborando tutto, ecco che gli strumenti software scelti come fondamenta di questo nuovo strumento sono stati ImageJ e Java. Per sviluppare il codice poi, mi è stato di grande aiuto Netbeans. Si potrebbe quindi schematizzare il tutto così:



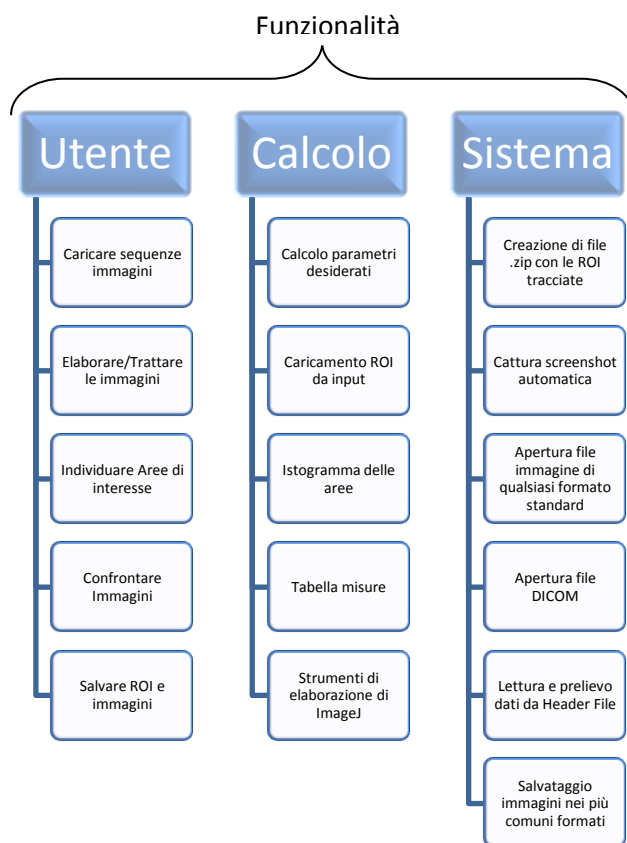
Si è preferito quindi utilizzare **ImageJ** (ImageJ)^[7], perché grazie alla sua natura open-source nativa Java, mi ha permesso di avere accesso diretto alle librerie di cui lui stesso fa uso, estenderle e poterle quindi utilizzare per il mio scopo. C'è da evidenziare comunque quanto queste librerie (quasi totalmente open) siano spesso insufficientemente documentate. Il che ha richiesto da parte mia un ulteriore studio del codice. Ciò detto, l'apporto dato dagli strumenti di ImageJ è stato indubbiamente fondamentale per la buona riuscita del mio progetto.

La scelta di **Java OO**, partendo da quanto appreso in precedenza è stata pressoché scontata, non tanto per la mia mancata apertura allo studio di nuovi linguaggi, quanto per il fatto che le tempistiche di

realizzazione del progetto e le specifiche richieste abbiano trovato in esso ampia soddisfazione. Per poter quindi lavorare agevolmente con le GUI, è stato di insostituibile utilità **Nebeans** (NetBeans)^[11], che grazie alla sua versatilità ha permesso quanto detto nei paragrafi precedenti.

2.4.2 Il software PERFECT.. i macroblocchi

Cercando di riassumere le funzionalità offerte dal software realizzato, eccone una prima rappresentazione grafica sottoforma di macroblocchi funzionali:



- **Utente**
 - **Caricare Sequenze di immagini**
Come accennato in precedenza, l'indagine visiva del radiologo richiede il caricamento di una serie di immagini. Grazie alle funzionalità implementate nel software, questo tipo di attività è largamente consentita.
 - **Elaborare/Trattare le immagini**
Per poter individuare le aree di interesse le immagini devono essere sottoposte a delle elaborazioni per poter visualizzare determinati tessuti piuttosto che altri (si veda il paragrafo riguardante il windowing)
 - **Individuare Aree di interesse**
una volta adeguatamente elaborate, le immagini permettono al radiologo l'individuazione di anomalie negli organi e/o nei vari tessuti. Il tracciamento delle ROI in diverse modalità è reso possibile dagli strumenti di tracciamento in dotazione di ImageJ
 - **Confrontare Immagini**
Poiché il software permette l'apertura simultanea di diverse immagini in modalità differenti, il radiologo può operare, se necessario, un confronto. Tutto ciò rende più agevole l'indagine, e (come nel nostro caso) la fase di acquisizione dei dati.
 - **Salvare ROI e relative immagini**
Lo scopo principale del software realizzato era quello di permettere il salvataggio delle ROI accuratamente tracciate dal radiologo in sede di acquisizione. Quindi la realizzazione di questa funzionalità è stata di primaria importanza. E' inoltre possibile grazie alle funzioni di ImageJ salvare tutte tutte le immagini aperte.

- **Calcolo**
 - **Calcolo parametri desiderati**
Quando l'utente ha terminato il tracciamento della ROI, è possibile far analizzare quest'area all'appositi strumenti di calcolo di alcuni parametri ritenuti

interessanti dai radiologi (parametri che ci sono stati comunicati in sede di intervista).

- **Caricamento ROI da input**
In una prima fase di test dei dati è stato necessario caricare delle ROI tracciate da altra fonte (le ROI realizzate con il software PERFeCT sono state spedite al dott. Ludovico Carozza che le ha utilizzate per i suoi studi e rilievi per poter poi calcolare le mappe colorimetriche)
- **Istogramma delle aree**
Altro strumento di valutazione dell'immagine è l'istogramma (si veda il paragrafo 2.4.6).
- **Tabella misure**
La tabella delle misure altro non è che la GUI su cui vengono visualizzati i valori dei parametri che si è deciso di calcolare. Questo strumento permette una valutazione quantitativa dei dati acquisiti con l'osservazione dell'immagine.
- **Strumenti di elaborazione di ImageJ**
Come anticipato in precedenza, ImageJ grazie alle sue caratteristiche, mi ha permesso di mettere a disposizione dell'utente tutta una gamma di strumenti di elaborazione e trattamento delle immagini desiderate. Nel nostro caso sono state trattate prevalentemente delle immagini in formato DICOM

- **Sistema**

- **Creazione di file .zip contenenti le ROI tracciate**
Sin dall'inizio si è presentata l'esigenza di acquisire i dati e formattarli nel modo ottimale in modo da rendere agevole il loro invio. Una delle directory create automaticamente dal software all'atto della inizializzazione, contiene tutti i file zip creati automaticamente per impacchettare i file ROI. Così facendo l'invio risulta agevole come richiesto dalle specifiche.

- **Cattura automatica screenshot**
L'acquisizione di ogni tipo di dato/indizio della sessione d'indagine radiologica non può tralasciare nessun aspetto. La cattura automatica di screenshot e relativo salvataggio in apposita directory creata automaticamente dal software, permette di ottenere oltre agli altri dati precedentemente descritti, anche un'idea di quello che il medico aveva in quel momento sul monitor.
- **Apertura immagini di qualsiasi formato standard**
Grazie agli strumenti presenti in ImageJ è possibile aprire ogni tipo di immagine ed elaborarla
- **Apertura file DICOM**
a differenza di altri software di elaborazione d'immagine, ImageJ è in grado di trattare immagini nel formato DICOM. Funzionalità sfruttata al massimo dal software realizzato.
- **Letture e prelievo dati da Header File**
Come discusso in precedenza, poiché la valutazione delle immagini richiede l'apertura di queste ultime con criteri particolari, è necessario accedere all'Header File di ogni singola immagine sia per poter leggere manualmente i dati desiderati, sia per poterli estrapolare ed utilizzarli. La funzione di lettura e visualizzazione dell'Header File è svolta dallo strumento "Show info" automaticamente chiamato dal software, mentre la lettura ed estrapolazione automatica per agevolare l'apertura (ad opera di ImageJ) delle immagini desiderate è assegnata ad apposite librerie sviluppate da SANTEC/TUDOR (SANTEC/TUDOR)^[14].
- **Salvataggio immagini nei più comuni formati**
le immagini precedentemente caricate possono essere agevolmente salvate nei vari formati previsti da ImageJ

Come è facile evincere dalla descrizione delle funzionalità offerte, si è cercato di far assumere al software un carattere quanto più standard possibile, in modo da essere flessibile e riusabile anche in altri ambiti se sarà necessario.

Capitolo 3

Implementazione del progetto

Dopo una descrizione macroscopica delle funzionalità, in questo capitolo verranno discusse le porzioni salienti del codice con cui è stato implementato il progetto.

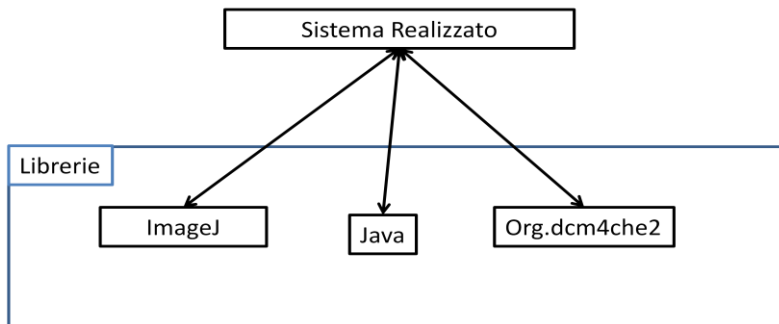
Come si evince dalla figura seguente, il software realizzato interagisce principalmente con tre blocchi di librerie.

Il primo blocco è quello di ImageJ. E' stato proprio grazie alle tante funzionalità presenti in questo software che si è potuto offrire le funzionalità richieste sia all'utente che agli ingegneri.

Per la gestione e l'archiviazione dei file si è ricorso spesso alle librerie di Java.

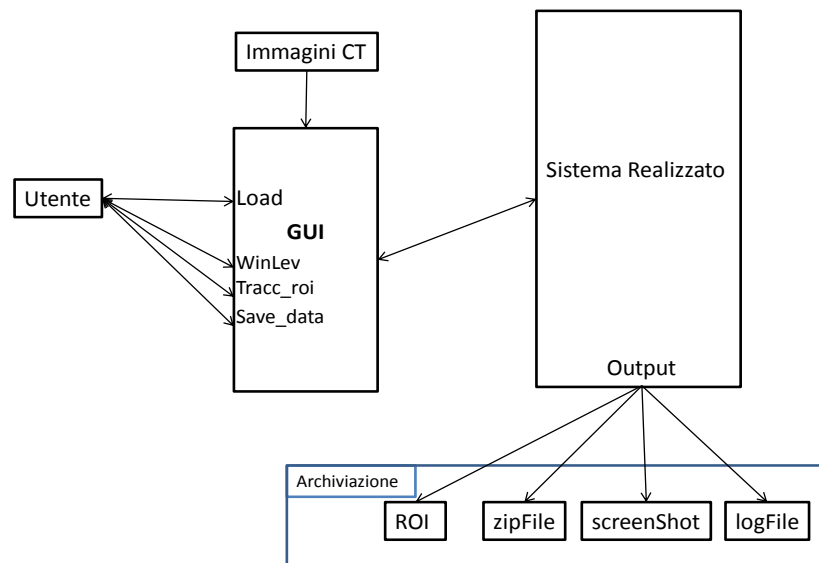
Grazie alle librerie "org.dcm4che2" è stato possibile trattare in modo celere i file DICOM, in particolare la lettura dell'header file per le operazioni desiderate.

Architettura di alto livello del software



Riallacciandoci a quanto detto in merito alle fasi dell'analisi perfusionale, ecco come queste fasi avvengono interagendo col nostro software:

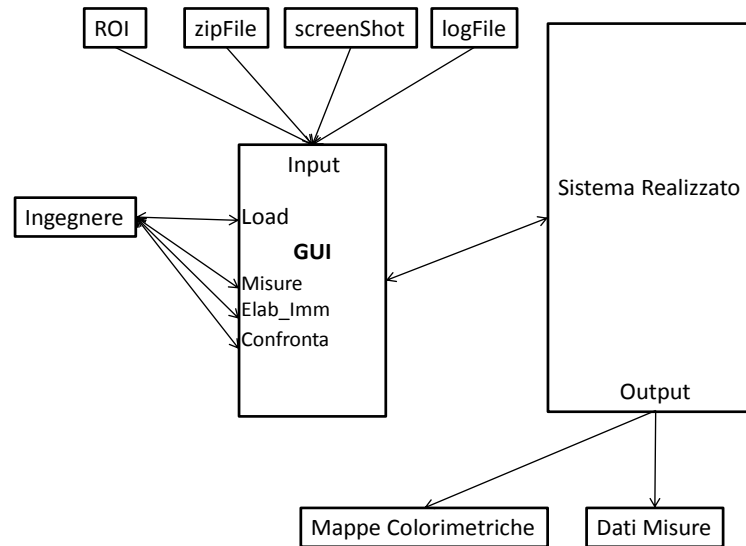
Fase 1 dell'esame perfusionale - Il medico (utente)



Notiamo come l'utente (il medico) interagisca con la GUI, caricando le immagini CT dell'esame prescelto. Utilizzando il Window Level può visualizzare in modo opportuno le immagini, ne traccia le ROI e salva sia le ROI che gli altri dati. L'output di questa fase è quindi l'archiviazione dei dati.

Passiamo alla Seconda fase:

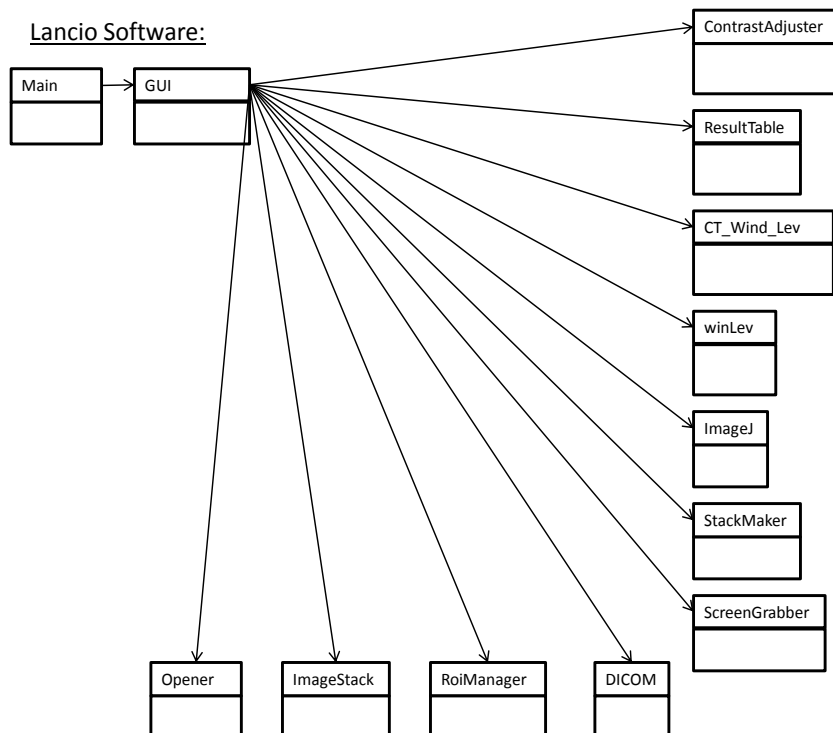
Fase 2 dell'esame perfusionale - l'Ingegnere



In questa fase è possibile notare che ad interagire con il software non sia più il medico, ma l'ingegnere. Egli (come si può notare dalla figura) interagisce con la medesima interfaccia, con la differenza che ha in input i dati creati (come output) dal medico.

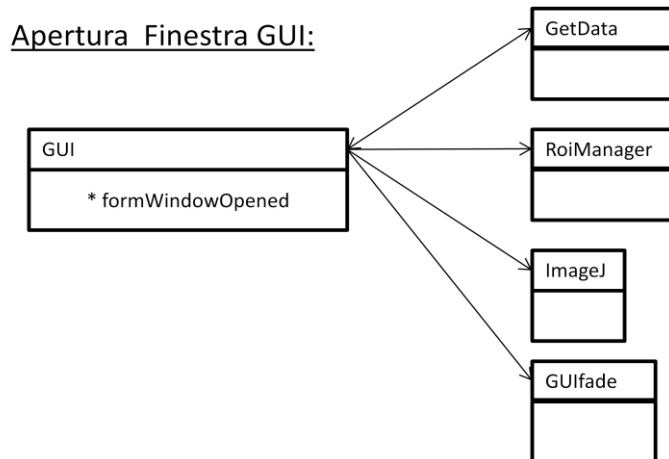
Utilizzando questi dati per le sue valutazioni (confronti e calcoli), è in grado di generare delle mappe colorimetriche. E' inoltre possibile un salvataggio delle valutazioni quantitative effettuate.

Osservando adesso quanto accade al lancio del software...



Possiamo notare come il main crei e lanci la GUI. Essa, a sua volta, crea diversi oggetti necessari all'utente all'avvio del software. Per farlo attinge alle classi opportune.

Il seguente invece, è quello che accade all'apertura della finestra GUI:



Vediamo quindi come la GUI crei degli oggetti di tipo diverso, tra cui il “RoiManager” , “ ImageJ” e il “GUIfade” che è l’interfaccia grafica del “FadeMaker”.

Ciò detto, si passa a descrivere la parte implementativa della GUI.

3.1 Il software PERFECT.. la Graphical User Interface

Parte fondamentale per rendere agevole l’interazione tra l’utente e il software, la GUI è stata una delle prime esigenze emerse nella fase di progetto. Essa ha quindi assunto man mano l’aspetto di un ambiente in cui il medico può trovare a portata di mano un certo numero di pannelli di comando ricchi di strumenti che lo agevolano nella sua indagine radiologica. Poiché l’aspetto della stessa è stato pilotato dall’esigenza di creare uno strumento su misura dell’utente finale e non basandosi su canoni di design, non è stato curato nessun aspetto si mera apparenza, cercando piuttosto di utilizzare colori in grado di rendere quanto più rilassato possibile l’occhio

dell'utente. In genere si è cercato di mantenere un aspetto sobrio e per quanto possibile elegante e funzionale.

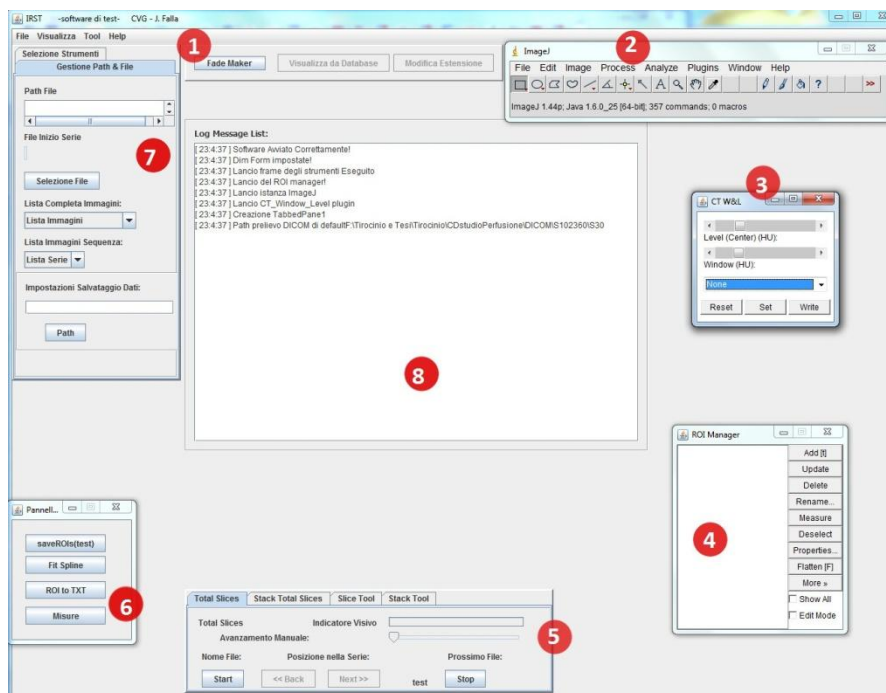


Figura 27: Interfaccia al lancio del software

1. Pannello Tool Supplementari

Il pannello permette all'utente il lancio di altri tool supplementari aggiungibili successivamente al software. I Tool attualmente presenti sono:

- **Fade Maker**

Questo tool permette il caricamento delle immagini, creazione di fade e il relativo salvataggio.

Vediamo come:

Premendo il pulsante “Selezione File Back” l’utente può selezionare l’immagine di background per i fade che si andranno a creare. Premendo i pulsanti “Selezione File Fade 1-2-3” l’utente può selezionare una delle tre immagini che desidera sovrapporre al background precedentemente selezionato.

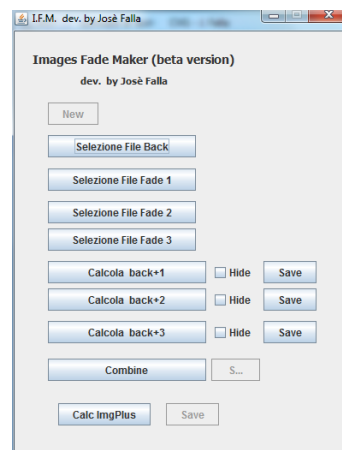


Figura 28: GUI del Fade Maker

FadeMaker:

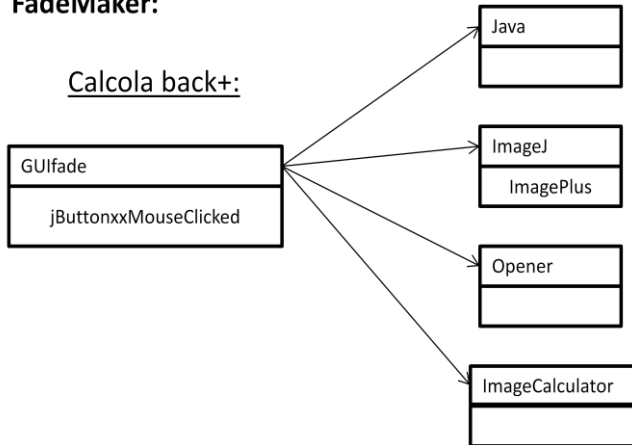
Selezione File :



Per procedere poi alla creazione vera e propria dei fade, basta premere il pulsante “Calcola back+1-2-3” per ottenere quanto detto. Con “Hide” è possibile nascondere il l’immagine sovrapposta per permettere dei confronti e/o altre valutazioni.

FadeMaker:

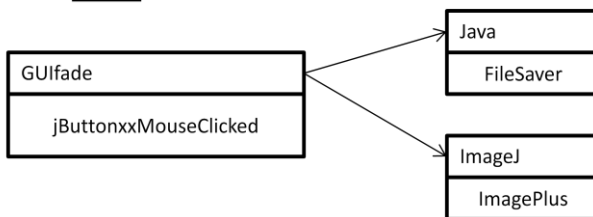
Calcola back+:



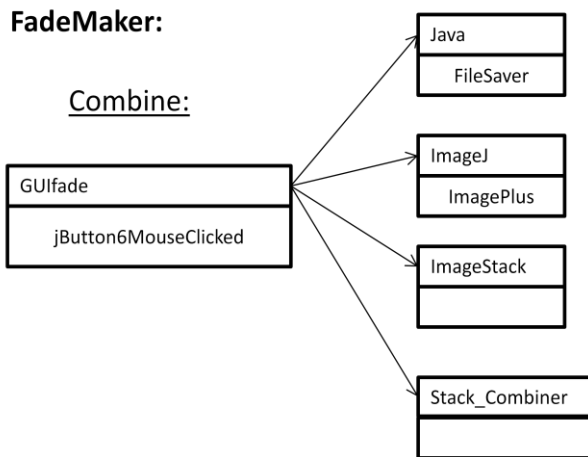
Con il tasto “Save” è poi possibile salvare l’immagine creata nella directory desiderata. L’immagine creata sarà in formato .jpg.

FadeMaker:

Save:



Grazie al pulsante “Combine” è quindi possibile creare una combine delle immagini create, in modo da visualizzare in unico file.



```

private void jButton6MouseClicked_
(java.awt.event.MouseEvent evt)
{
imgStk=stkComb.combineHorizontally(img3.getImageStack(),
img5.getImageStack());

    stkComb.run("Prova");

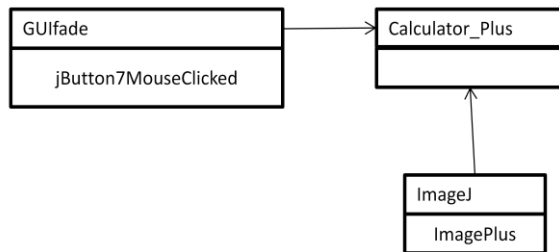
    img3.show();

    img5.show();
}
  
```

Grazie al pulsante “Calc ImgPlus” è possibile calcolare quanto detto variandone i parametri a piacimento. I fade creati con le funzioni precedenti hanno una dissolvenza del 50%.

FadeMaker:

Calc ImgPlus:



```
private void jButton7MouseClicked_  
(java.awt.event.MouseEvent evt) {  
    calcPls.calculate(img1, img2, 0.2, 0.0);  
    calcPls.run("Prova");  
}
```

Una volta terminata una prima creazione, è possibile procedere a nuove fasi di creazione premendo il tasto “New”.

L’utilizzo del “Fade Maker” ha permesso la creazione di fade tra immagini CT e immagini PET. La creazione di questi file è stata utile all’attività del sig. Mattia Gardini.

Nel capitolo riservato ai risultati è possibile consultare i Fade creati.

- **Visualizza da Database**

Il “Visualizza Database” è un software pensato per poter popolare ed accedere ad un database di immagini di pazienti sottoposti allo studio perfusionale, per poter permettere una ricerca veloce ed accurata in base ai parametri desiderati. La sua realizzazione è stata pensata nell’ultima parte della mia attività, quindi per motivi di tempo non è stato possibile uno sviluppo ulteriore (si veda il capitolo 8). Un’anteprima dell’interfaccia è la seguente:

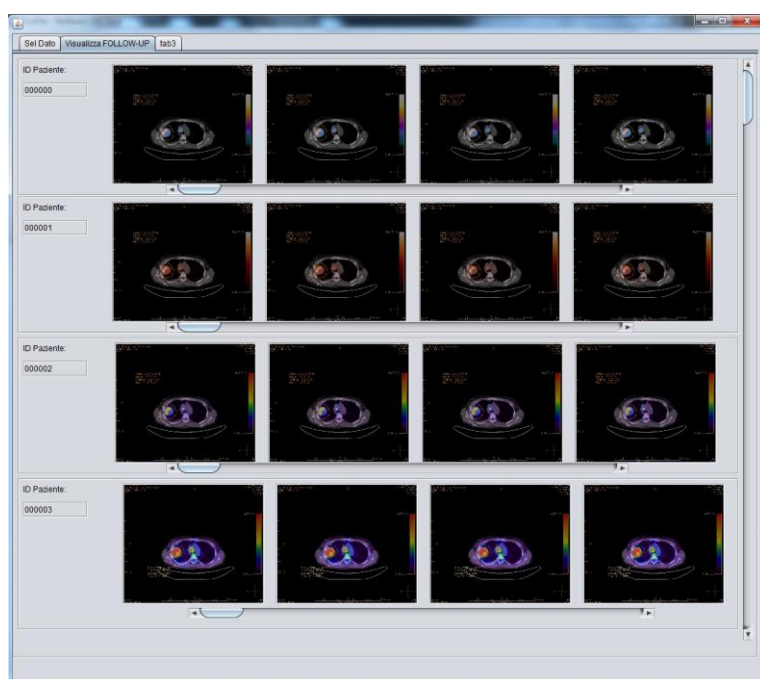


Figura 29: GUI di ricerca e visualizzazione immagini da database

- **Modifica Estensione**

Il “Modifica Estensione” è un semplice tool che permette di cambiare in “.dcm” l’estensione di tutti i file presenti nella cartella desiderata dall’utente. Questo tool è nato per andare incontro alle esigenze di tesi dell’ing. Marco Aldo Prattichizzo, permettendogli una più agevole gestione delle immagini DICOM che desiderava trattare.

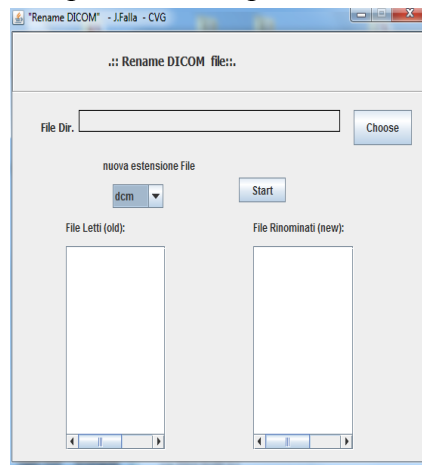


Figura 30: GUI del "Modifica Estensione"

Altri tool saranno sviluppati in futuro.

2. *Interfaccia ImageJ*

Come spiegato in precedenza, alla base delle funzionalità offerte c’è ImageJ. Esso è presente anche come GUI in modo da ampliare al massimo la gamma dei tool offerti all’utente. Per una accurata descrizione delle funzionalità di ImageJ si invita a fare riferimento al manuale utente presente sul sito del produttore.

3. *CT W&L*

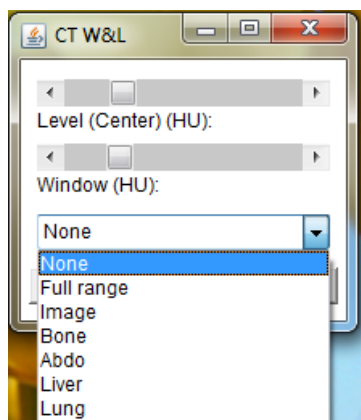


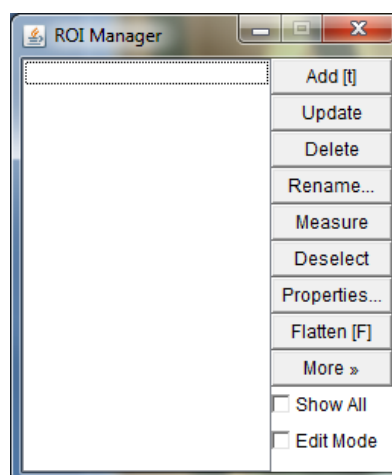
Figura 31: interfaccia CT W&L

Questo strumento permette la variazione del Level del Window dell’immagine CT in modo da permettere al radiologo le valutazioni necessarie. In ImageJ è già presente uno strumento del genere, ma grazie a numerose mie ricerche sul web è stato possibile trovare uno strumento ancor più completo che oltre alla sola

variazione dei valori di Level e Window, permette di visualizzare velocemente alcuni preset di valori. La combinazione di questi valori rende possibile evidenziare determinati tessuti piuttosto che altri (così come discusso nel paragrafo 2.4.1).

4. *ROI Manager*

Il “ROI Manager” è uno strumento di ImageJ che permette di gestire tutte le ROI tracciate sulle varie immagini ed operare su quella selezionata alcune modifica. Per approfondimenti si faccia riferimento al manuale utente di ImageJ.



5. *Pannello Multi Scheda*



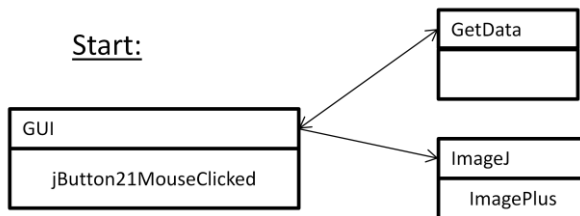
Nel “Pannello Multi Scheda” sono presenti le varie interfacce che consentono all’utente la visualizzazione delle immagini nelle varie modalità desiderate.

- Total Slices

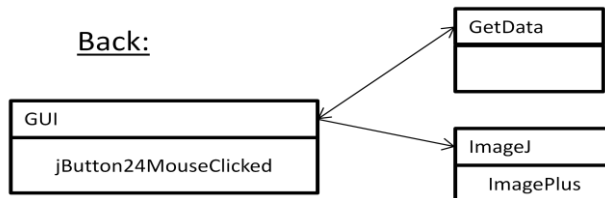
Il Pannello “Total Slices” consente una visualizzazione sequenziale delle immagini di tutto l’esame. In alto è presente un indicatore visivo che permette di capire in quale punto dell’esame ci si trovi. Grazie ai tasti “Back” e “Next” è possibile passare da un’immagine all’altra. Col tasto “Stop” è possibile azzerare quanto fatto per iniziare una nuova navigazione.

Di seguito vengono illustrati i rapporti tra le classi alla pressione del mouse su un determinato pulsante.

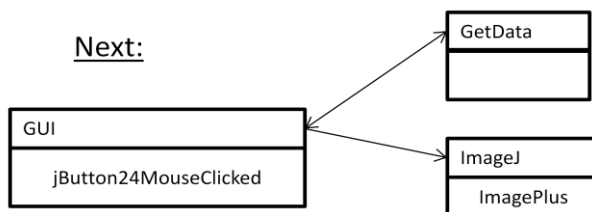
totalSlices:



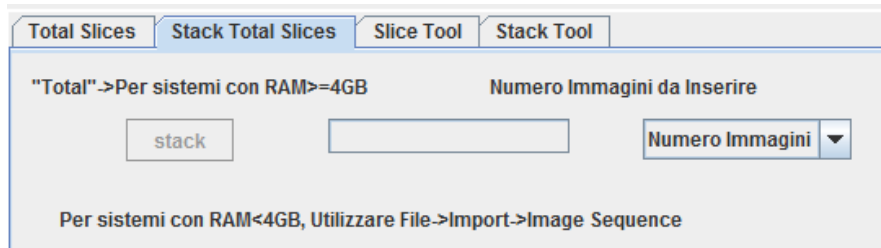
totalSlices:



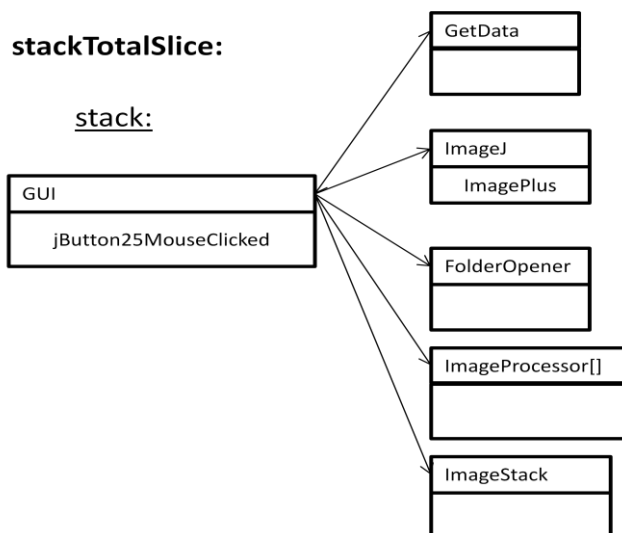
totalSlices:



- Stack Total Slices



Questo pannello permette la creazione di uno stack delle immagini dell'esame posto in sequenza. Poiché lo spazio occupato da un certo numero di immagini (negli studi fatti sono sempre state in totale 360 per ogni esame) richiede una certa quantità di memoria, è stato necessario inserire il menù a tendina posto sulla destra dell'interfaccia, in modo da selezionare quante immagini dovrà comprendere lo stack che si sta per creare. Sarà possibile la creazione di stack comprendenti 5-10-50 immagini o con "Total" tutte le immagini dell'esame.

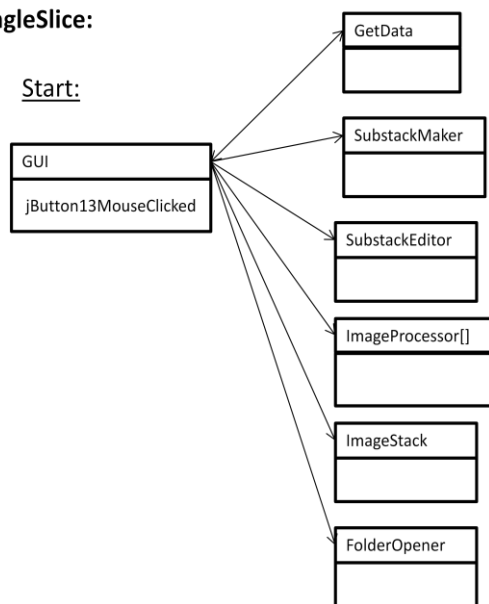


- Slice Tool

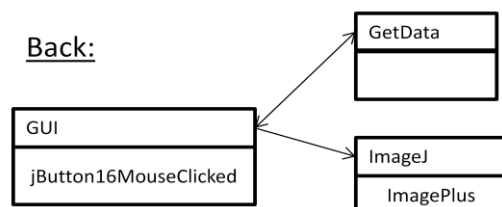
Lo “Slice Tool” è l’equivalente del “Total Slices” per la visualizzazione delle immagini della singola posizione del lettino in tutte le varie fasi dell’esame. Questo strumento è stato molto utile all’inizio, ma poi abbandonato perché le funzionalità implementate nello “Stack Tool”, hanno reso lo stesso molto più funzionale.

Diamo uno sguardo ai diagrammi funzionali dei pulsanti di questo tool:

singleSlice:

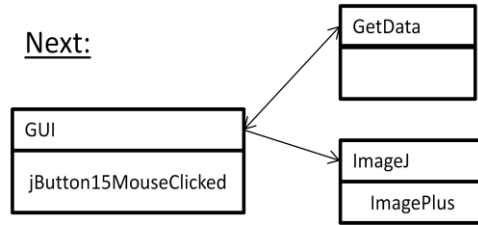


singleSlice:



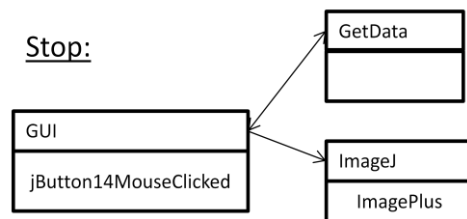
singleSlice:

Next:

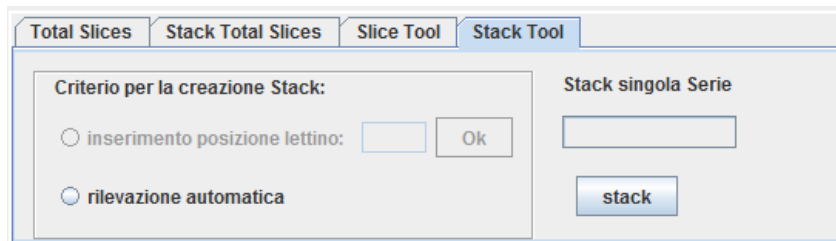


singleSlice:

Stop:

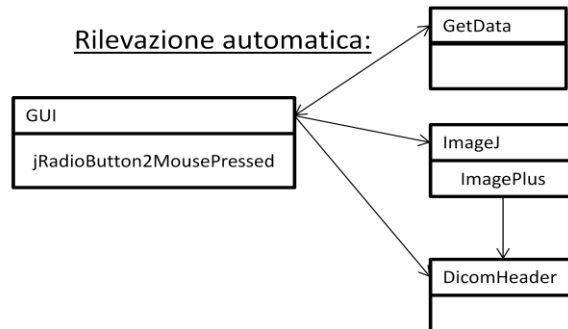


- Stack Tool



Una volta selezionata l'immagine relativa alla posizione del lettino desiderata, il radiologo può creare uno stack di tutte le immagini della stessa posizione, semplicemente facendola rilevare automaticamente al software. Per far questo, basta cliccare su "Rilevazione Automatica" e premere sul tasto "Stack".

stackSerie:



La seguente porzione di codice è particolarmente importante per la funzione svolta:

```
private void jRadioButton2MousePressed
(java.awt.event.MouseEvent evt) {

    jTextField9.setEditable(false);
    jTextField9.setEnabled(false);
    jButton41.setEnabled(false);
    //rileviamo quale immagine è al momento attiva e ne
    ricaviamo la posizione del lettino

    try {
        int indCodice;

        DicomHeader dh = new DicomHeader(ope);

        String stringa = dh.toString();
        String sottostringa = "Slice Location:";
        String valSliceLocation="    ";

        // al momento non è stata effettuata alcuna ricerca
        // ne conseguito alcun risultato
        boolean cerca = false;
        int indStr=0;
```

```

        // calcoliamo lunghezza di stringa e sottostringa
// la differenza sarà condizione di terminazione per il ciclo for

    int max = stringa.length() - sottostringa.length();
    // ricerchiamo la sottostringa ciclando il contenuto di
//quest'ultima
    test:
    for (int i = 0; i <= max; i++) {
        int n = sottostringa.length();
        int j = i;
        int k = 0;

        while (n-- != 0) {
            if (stringa.charAt(j++) != sottostringa.charAt(k++)) {
                continue test;

                //memorizza l'indice di dove si trova la frase "Slice
                Location:"

            }
        }

        // a questo punto è stata effettuata una ricerca
        // sarà possibile produrre un output
        cerca = true;

        indStr=j;

        break test;
    }

    int valPosition=0;
    int inizVal=indStr;
    int finVal=0;
    int flagSpazio=0;
    char spazio=' ';

    valSliceLocation=stringa.substring(indStr+1,indStr+6);
    //inizio e fine valore

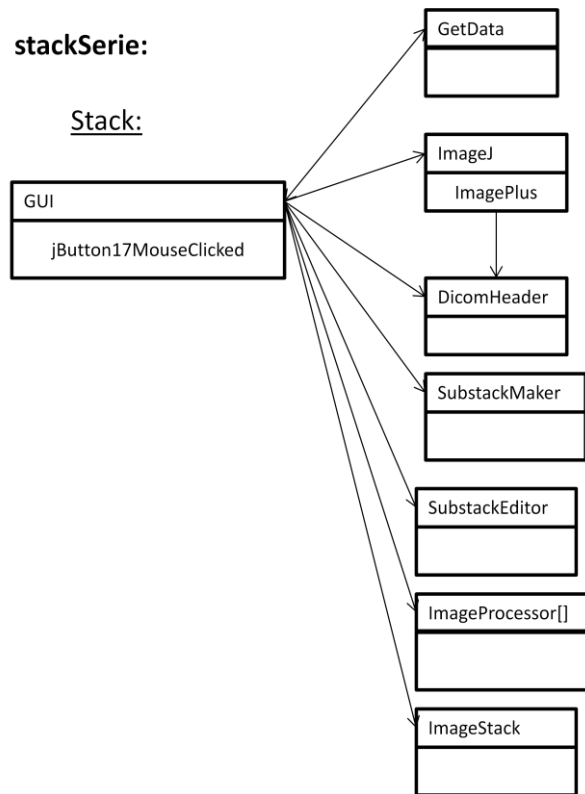
```

```

        //valSliceLocation=stringa.substring( inizVal-5,
        inizVal+20);
        //valSliceLocation=stringa.substring( inizVal, finVal);
        GetData g;
        g =new GetData();
        jTextArea2.append(g.getStringTime()+"Valore dello
        Slice Location: "+valSliceLocation+" \n");
// jTextField4.setText(valSliceLocation);
        jLabel28.setText(valSliceLocation);
        lettinoRilevato=valSliceLocation;
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```

Una volta rilevata la posizione, è possibile creare lo stack premendo l'apposito pulsante.



Una porzione di codice della creazione dello stack...

```

//ciclo di controllo della posizione del lettino dell'immagine e
in base a quello decide se inserirla nello stack
FolderOpener ordinatoreDiFile= new FolderOpener();
  
```

```

listaFileOrdinata=ordinatoreDiFile.sortFileList(listaFile);
  
```

```

ImageProcessor[] stackAllSlice = new
ImageProcessor[fileFrom.list().length];
// ImageProcessor[] stackAllSlice = new ImageProcessor[36];
// now the results array is full. Just show them in a stack:
ImageStack stack2 = new ImageStack(512,512);
int Max=fileFrom.list().length;
for(int i=0; i<Max;i++){
  
```

```

        int indCodice;
        trovDIRFILE=
        listaFileOrdinata[i].compareTo("DIRFILE");

        if(trovDIRFILE!=0){
ope = opener.openImage(path+file.separatorChar+
_listoFileOrdinata[i].toString());
        }
        DicomHeader dh = new DicomHeader(ope);
        /*
        //visualizza la finestra di showinfo della dicom
        dh.setInfo(dh);
        dh.setVisible(true);
        */

        // definiamo la stringa in cui effettuare la ricerca
// e la sottostringa da ricercare
String stringa = dh.toString();
String sottostringa = "Slice Location:";
String valSliceLocation=" ";

// al momento non è stata effettuata alcuna ricerca
// né conseguito alcun risultato
boolean cerca = false;
int indStr=0;
// calcoliamo lunghezza di stringa e sottostringa
// la differenza sarà condizione di terminazione per il ciclo for
int max = stringa.length() - sottostringa.length();
// ricerchiamo la sottostringa ciclando il contenuto di
quest'ultima
test:
for (int ii = 0; ii <= max; ii++) {
    int n = sottostringa.length();
    int j = ii;
    int k = 0;

```

```

while (n-- != 0) {
    if (stringa.charAt(j++) != sottostringa.charAt(k++)) {
        continue test;

        //memorizza l'indice di dove si trova la frase "Slice
Location:"

        }
    }

    // a questo punto è stata effettuata una ricerca
    // sarà possibile produrre un output
    cerca = true;
    indStr=j;

    break test;
}

int valPosition=0;
int inizVal=indStr;
int finVal=0;
int flagSpazio=0;
char spazio=' ';
valSliceLocation=stringa.substring(indStr+1,indStr+6);
//inizio e fine valore
//valSliceLocation=stringa.substring( inizVal-5, inizVal+20);
//valSliceLocation=stringa.substring( inizVal, finVal);
jTextArea2.append("Valore dello Slice Location:
"+valSliceLocation+" \n");
// jTextField4.setText(valSliceLocation);
jLabel28.setText(valSliceLocation);
jTextArea2.append("Valore_
lettinoRilevato:"+lettinoRilevato+"\n");
sonoUguali=valSliceLocation.compareTo(lettinoRilevato);
if(sonoUguali==0){

```

```

        //blocco per la creazione dello stack. fare copia/incolla dal
        bottone header1
        //scrittura su log message:
        //g=new GetData();
        JTextArea2.append(g.getStringTime()+"Creazione Stack
        _in corso. Attendere prego..."+"\n");
//ImageProcessor[] stackAllSlice = new
ImageProcessor[listFile.list().length];
//for (int i=0; i< Max; i++) {

jTextArea2.append(g.getStringTime()+listaFileOrdinata[i].toStr
ing()+"\n");
        // ope = opener.openImage(path+file.separator+
        _listaFileCartella[i]);
        //==> ope =
opener.openImage(path+file.separatorChar+listaFileOrdinata[i]
.toString());
        // stack = ope.getStack();
//ope.setStack(stack);
stackAllSlice[i]=ope.getProcessor();
        //stack.addSlice(Integer.toString(i),stackAllSlice[i] );
stack2.addSlice(ope.getTitle(),stackAllSlice[i] );
        //}
} //if lettino=valore rilevato
jProgressBar2.setValue(i);
} //for
//if(stack2.)
new ImagePlus("Stack", stack2).show();
// ope.show();
} catch (Exception e) {
        e.printStackTrace();
        }
img.requestFocus();

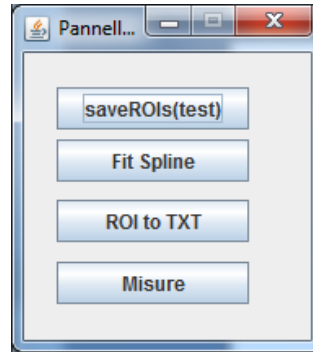
```

Sullo stack creato è possibile operare con tutti i tool di ImageJ e scorrere le *slice* della sequenza grazie allo scroll del mouse.

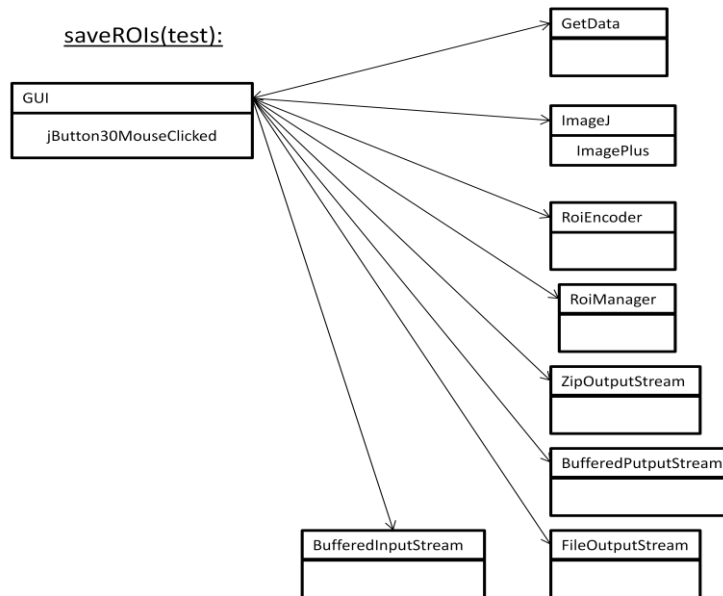
6. *Pannello Strumenti*

Il “Pannello Strumenti” è composto da quattro pulsanti.

- “saveROIs(test)”
permette il salvataggio delle ROI tracciate ed inserite in ROI manager. Crea in automatico un file .zip per contenerle e scatta in automatico uno screenshot salvato nell’apposita cartella “screenShot” all’interno della cartella in cui si è scelto di salvare i dati dell’acquisizione. Il file .zip avrà come nome l’immagine da cui sono state estratte le ROI e l’ora.



pannelloStrumenti:



Creazione dei file .roi

```
//ciclo for che scorre tutto l'array di ROI e le salva nell'apposito file
```

```
for( int i=0;i<roi.getCount();i++){
```

```

        saveRoi=new
RoiEncoder(pathSalvRisult+file.separatorChar+"roi"+file.separ
atorChar+ope.getShortTitle()+"_"+Integer.toString(i)+".roi");
    // saveTextRoi=new
RoiEncoder(pathSalvRisult+file.separatorChar);
        roiDa=arrayDiRoi[i];
//saveRoi.write(roiDa);

try
    {
//roiWriter.saveRoi(ope);
saveRoi.write(arrayDiRoi[i]);
//saveTextRoi.saveTextRoi(arrayDiRoi[i].toString());

        JTextArea2.append(g.getStringTime()+"Ho salvato la
ROI:
"+ope.getShortTitle()+"_"+Integer.toString(i)+".roi"+"\\n");
System.out.printf("Ho salvato la ROI");
    }
    catch(Exception e)
    {
        // GetData g=new GetData();
        JTextArea2.append(g.getStringTime()+"Non ho salvato la
ROI!\\n");
        System.out.printf("Non ho salvato la ROI\\n");
    }
} //fine for che crea i file .roi

```

Adesso è possibile creare il file compresso (.zip):

```

/// creatore di zip

//show all
roi.runCommand("Show All");

//salvataggio dello screen shot

```

```

        immagineScreen= captureScreen.captureScreen();
        //GetData g=new GetData();

immagineScreen.setTitle("ScreenProva"+Integer.toString(g.get
Ora()+Integer.toString(g.getMinuti()+Integer.toString(g.getSe
condi())));

IJ.saveAs(immagineScreen,"jpg",pathSalvRisult+file.separator
Char+"screenShot"+file.separatorChar+immagineScreen.getTitl
e()+"_"+ope.getShortTitle()+".jpg" );

        JTextArea2.append(g.getStringTime()+"Salvataggio
screen shot
:"+immagineScreen.getTitle()+"_"+ope.getShortTitle()+".jpg"+
"\n");
        //fine screenShot

//File zipRoi=new
File(pathSalvRisult+file.separatorChar+"outfileProvaZip.zip");

        // il metodo try/catch di permette di gestire le eccezioni, se
qualcosa
        // non funziona in try allora catch notifica l'errore

try
{

//pathSalvRisult+file.separatorChar+"NomeFile"+Integer.toStri
ng(2)+".roi"

        // definiamo l'output previsto che sarà un file in formato zip
//ZipOutputStream out = new ZipOutputStream(new
//BufferedOutputStream(new FileOutputStream(zipRoi)));

ZipOutputStream out = new ZipOutputStream(new

```

```

        BufferedOutputStream(new
FileOutputStream(pathSalvRisult+file.separatorChar+"fileZip"
+file.separatorChar+ope.getShortTitle()+"_"+g.getStringTimeN
oChar()+".zip")));

        // byte[] data = new byte[1000];

for(int i=0;i<roi.getCount();i++){
    // definiamo il buffer per lo stream di bytes
    byte[] data = new byte[1000];

    //for(int i=0;i<roi.getCount();i++){

        // indichiamo il nome del file che subirà la compressione
        BufferedInputStream in = new BufferedInputStream
            (new
FileInputStream(pathSalvRisult+file.separatorChar+"roi"+fi
le.separatorChar+ope.getShortTitle()+"_"+Integer.toString(i
)+".roi"));
        int count;
        // processo di compressione
        out.putNextEntry(new
ZipEntry(ope.getShortTitle()+"_"+Integer.toString(i)+".roi"));

//file che vogliamo mettere dentro il file zip (senza path)

        while((count = in.read(data,0,1000)) != -1)
        {
            out.write(data, 0, count);
        }
        in.close();
    }//for

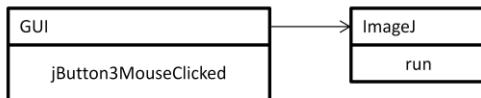
    out.flush();
    out.close();

```

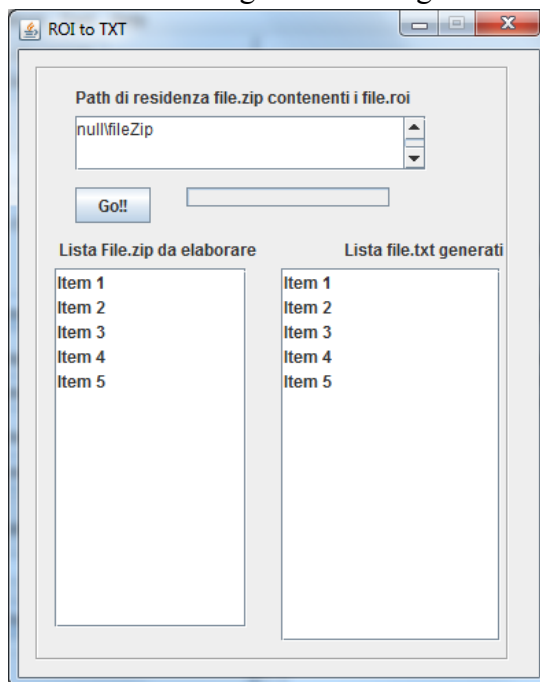
- “Fit Splice”
permette di aggiustare la ROI tracciata punto per punto. Per far questo viene chiamato in causa lo strumento di ImageJ.

pannelloStrumenti:

Fit Spline:



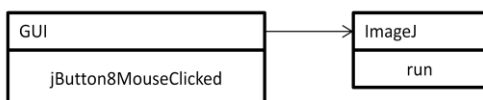
- “ROI to TXT”
Premendo su “ROI to TXT” sarà possibile convertire i file .roi in file .txt. Lo strumento verrà sviluppato in futuro.
La sua interfaccia grafica è la seguente:



- “Misure”
Premendo questo pulsante è possibile effettuare delle misure sulla ROI selezionata. I parametri misurati, sono quelli indicati dai medici come più significativi per l’indagine.

pannelloStrumenti:

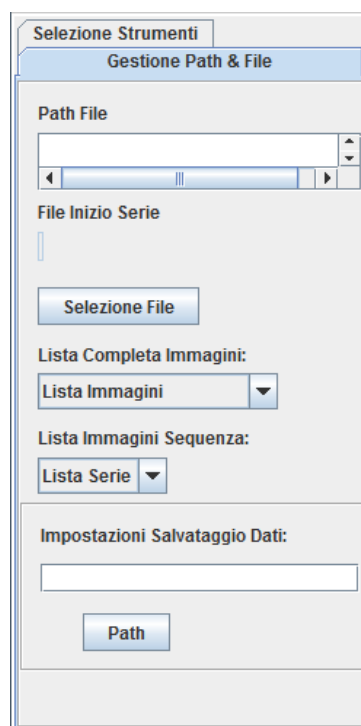
Misure:

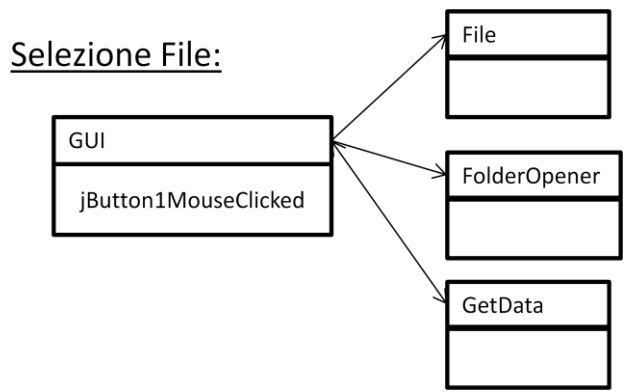


Le suddette misure vengono effettuate lanciando un plugin di ImageJ.

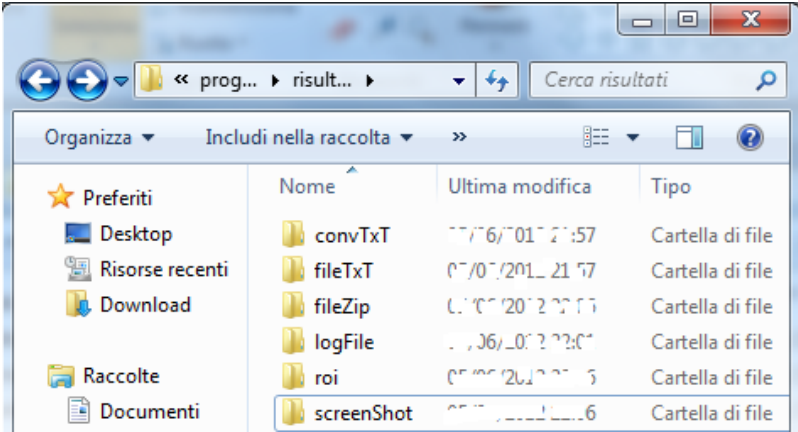
7. *Pannello di Gestione Path & File*

Grazie al “Pannello di Gestione Path & File” l’utente può selezionare da quale directory caricare le immagini premendo sul tasto “Selezione File”.





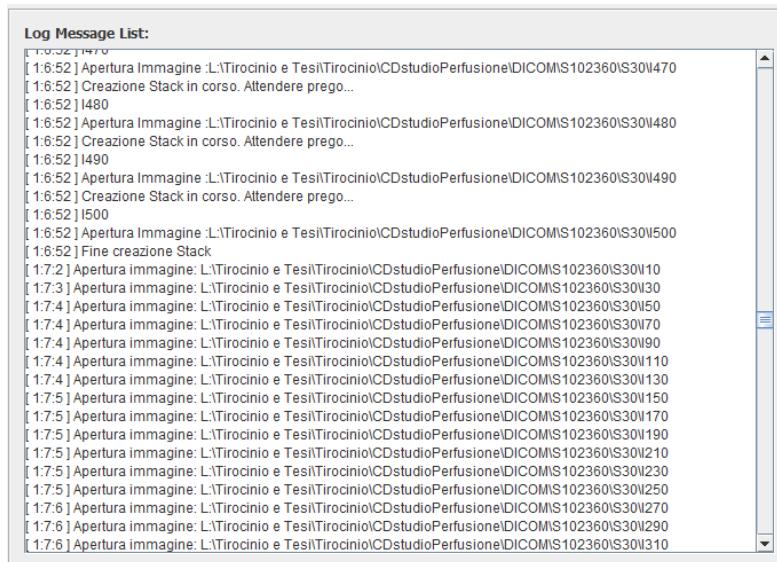
Verrà visualizzato il file di inizio serie dell’esame selezionato. Dal menù a tendina “Lista Completa Immagini” è possibile scorrere tutte le immagini e cliccando su una di esse aprirla automaticamente. Il menù a tendina “ Lista Immagini Sequenza” è stato utilizzato nelle prime versioni ed attualmente è stato abbandonato perché sostituito da altri strumenti. Premendo su “Path” è possibile selezionare in quale directory si desidera salvare tutti i dati che verranno automaticamente creati dal software. Una volta conclusa questa fase, nella directory selezionate verranno create automaticamente diverse sotto- directory. Il suo aspetto sarà questo:



Dal secondo pannello “Selezione Strumenti” è poi possibile quale degli strumenti (non) visualizzati si desidera (visualizzare) nascondere.

8. *Log Message List*

Quest'area del software permette all'utente di capire quali operazioni sono state fatte dal software in modo da rilevare eventuali anomalie e/o messaggi d'errore che non sono stati volutamente dati tramite apposite finestre. Ad ogni azione è possibile notare che sia stata associata l'ora esatta.



Al termine della sessione del programma, quanto avvenuto viene scritto in un apposito logFile salvato nella sotto-cartella a lui riservata nella directory selezionata come destinazione dei dati.

Capitolo 4

Risultati

Lo sviluppo delle varie parti del software oggetto di questa tesi ha richiesto innumerevoli fasi di test e validazione dei risultati sperimentali ottenuti. Poiché queste fasi (per essere considerate tali) non possono essere svolte dalla stessa persona, è stato necessario individuare una persona che incarnasse le esigenze dell'utente finale che mettesse in risalto le anomalie rilevate per poi correggerle. Fondamentalmente si è agito in “doppio Cieco” in modo da non avere “contaminazioni” tra i vari tester. Le fasi salienti sono state le seguenti:



La fase di raccolta delle esigenze in sede di validazione dei risultati non va confusa con la fase di uguale nome ma che si è verificata all’inizio, poiché la prima ha avuto prevalentemente carattere conoscitivo del campo che si andava ad esplorare, mentre la seconda aveva carattere assolutamente operativo e tecnico. La realizzazione del progetto e la realizzazione dei prototipi è stata descritta nei capitoli precedenti.

Nelle figure seguenti, possiamo apprezzare quanto ottenuto in sede di risultati.

Ecco l’aspetto del software in funzione:



Figura 32: screenshot di una sessione del software PERFECT

E’ possibile distinguere i vari pannelli contenenti gli strumenti, un’immagine CT aperta singolarmente (a sinistra) e uno stack di immagini (a destra) su cui sono state tracciate due ROI.

Le immagini seguenti invece, sono il risultato dell’utilizzo del “Fade Maker”:

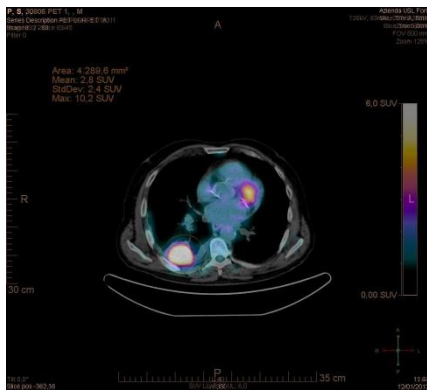


Figura 33: Fade generato con "Fade Maker" tra immagini CT e PET con maschera GE

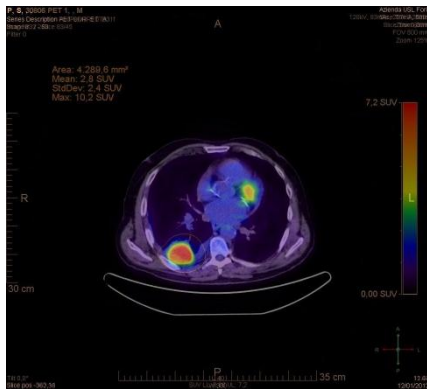


Figura 34: Fade generato con "Fade Maker" tra immagine CT e PET con maschera S_PET

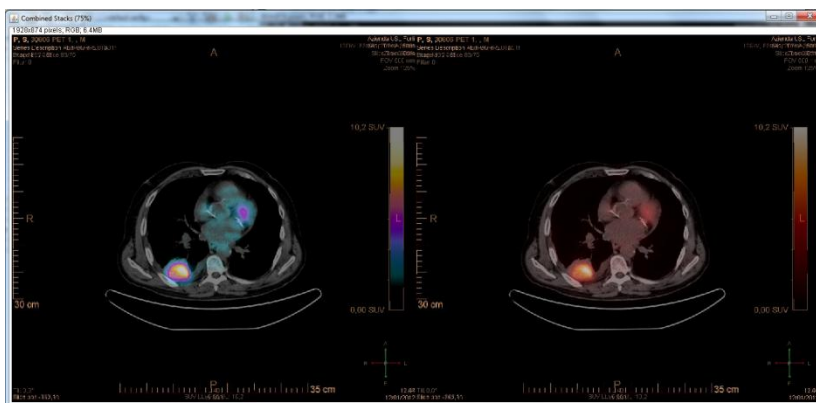


Figura 35: "Combined" di due Fade di esempio generata con il "Fade Maker"

Infine, ma non per grado di importanza, è bene evidenziare come la disponibilità e le abilità informatiche del dott. Barone siano state di fondamentale importanza per poter realizzare al meglio quanto richiesto dai Medici e che il mio intervento doveva trasformare in specifiche di progetto. Le tecniche, le esigenze dei medici evidenziate e quanto appreso dal dott. Barone sono per me un bagaglio di conoscenze di insostituibile valore.

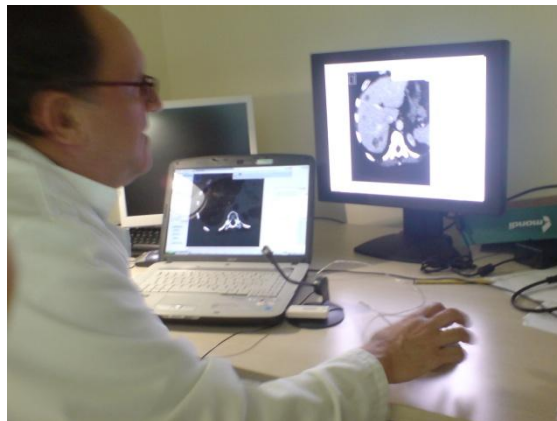


Figura 36: Seduta di acquisizione e validazione dati sperimentali presso IRST. Nella foto il Dott. Domenico Barone

Capitolo 5

Conclusioni e Sviluppi Futuri

Alla fine di questa mia attività presso IRST, non posso che rendermi conto di quanto possa essere dinamica la figura dell'ingegnere. E' stupefacente notare come una struttura apparentemente di esclusiva pertinenza medica, in realtà abbracci decine di figure professionali, che se sapientemente coordinate (come avviene già in struttura), possano divenire importanti o addirittura fondamentali per la buona riuscita di progetti, e delle normali attività di lavoro quotidiano. Di certo è affascinante l'approccio di un aspirante ingegnere elettronico ad un ambito del tutto nuovo ai suoi occhi. Tutto ciò, che banalmente potrebbe essere attribuito ad una non correlazione tra quanto studiato durante il percorso accademico e quanto si andrà a studiare, in realtà si è dimostrato essere un modo altamente stimolante e formativo, con cui applicare concetti appartenenti ad altri ambiti, idee, permettendo quindi una contestualizzazione del sapere, e del saper fare. Al di là della professionalità di ogni medico, infermiere e tecnico dell'UO, è bene ricordare l'enorme disponibilità e celerità dimostratami. Un aspetto a cui tengo tanto (e di cui ho avuto modo di verificare le potenzialità) è l'aspetto umano. Questo aspetto, per una figura come la mia avente a che fare con rigide relazioni di tipo matematico e logico, spesso tende a cadere in secondo piano. Figure professionali aventi rapporti con persone spesso necessitanti di cure, sanno bene quanto invece questo aspetto sia importante. Penso quindi, che la fusione tra questi due macro aspetti, possa solo portare a qualcosa di buono.

Leggendo quanto detto fino a questo capitolo è facile notare come la mia attività di tesi abbia avuto aspetti diversi man mano che si andava avanti. Quello che inizialmente doveva essere un semplice script si è poi trasformato in un piccolo software in grado di permettere l'interazione tra il medico e le immagini acquisite. Sviluppando le caratteristiche del software e incontrando di continuo i medici, il prof. Bevilacqua e gli altri colleghi, le funzionalità che si desiderava implementare sono diventate numerose e non sempre di veloce realizzazione. Quindi come descritto in precedenza ad esempio, lo strumento che permette di archiviare delle immagini e di richiamarle velocemente è stato pensato e parzialmente realizzato per motivi di tempo.

Si spera che sia questo che altri strumenti verranno sviluppati dal sottoscritto magari in sede di collaborazioni future.

Un altro strumento è stato sviluppato in parallelo (ma con altro scopo) dal dott. Carozza che permette la creazione, date le immagini, di creare delle mappe colorimetriche della perfusione (il tutto sviluppato in Matlab). Si pensava inizialmente di permettere l'interazione del software da me sviluppato con questo modulo Matlab, ma per gli stessi motivi l'implementazione di tutto ciò non è stata possibile. Quanto offerto in sede di tesi non ha nessuna presunzione di essere uno strumento ottimo di calcolo, poiché tutto ciò richiede tempistiche di lavoro da dedicare assolutamente non paragonabili a quanto speso da me in questo periodo. E' mia intenzione cercare di raffinarlo quanto prima.

Capitolo 6

Ringraziamenti

Per questa mia stupenda esperienza, non posso che ringraziare il Prof. Alessandro Bevilacqua, per l'enorme disponibilità, l'interessamento e la fiducia dimostratami. Ringrazio anche il Prof. Giampaolo Gavelli ed il Dott. Domenico Barone, per avermi accolto nell' UO e per essermi stati vicini e disponibili durante tutta l'attività. In oltre, di altrettanta importanza per la buona riuscita della mia attività, desidero ringraziare il Dott. Ludovico Carozza e l'instancabile e paziente Mattia Gardini, che oltre ad un ottimo "collega" di lavoro, si è dimostrato essere un amico di altrettanto calibro. In fine, ma non da meno, un grazie va ai ragazzi e le ragazze della UO.



Figura 37: foto ricordo di fine meeting del CVG presso ARCES

Andando fuori dall'ambito accademico, desidero ringraziare prima di tutto i miei genitori, che con enormi sacrifici mi hanno permesso di proseguire gli studi senza farmi mai pesare le loro inevitabili

rinunce. Il loro è stato un apporto dal duplice aspetto. Come sempre, oltre ad essere dei genitori esemplari, sono sempre stati anche degli ottimi amici e confidenti. Punti di riferimento solidi e comprensivi nei momenti di difficoltà. Un Ringraziamento va a quell'uomo (sia pure ancora ragazzo) che è mio fratello Amedeo, che con il suo carattere, pur a migliaia di chilometri di distanza ha sempre condiviso con me tutti i momenti di gioia e di difficoltà. Pilastro saldo in questo periodo della mia vita è stata Eva, la mia dolce metà. Grazie alla sua tempra mi ha permesso di andare avanti incoraggiandomi ed aiutandomi nei periodi più difficili di quest'avventura, condividendo con me sia i momenti di gioia, che le battaglie. Un grazie di cuore va a tutta la restante parte della famiglia: le mie cugine, Chiara e Roberta, le mie zie Antonella e Veronica, mio zio Massimo, e i miei nonni materni e paterni, che pur costretti a lunghissime telefonate, hanno reso possibili tutte le varie fasi del mio "vivere lontano da casa". Ringrazio per il loro contributo i miei fidati e cari amici di sempre: Giovanni Liuzzo, Giancarlo e Salvo Pluchino. Un enorme ringraziamento va a tutte quelle persone splendide che ho conosciuto qui a Cesena ed in Romagna in genere. In particolare, mi riferisco ai ragazzi di via Ravennate 190: Gigi, Damiano, Michele, Alessandro, Danilo, Prà, Jack, Alex e Rinaldo, con cui ho condiviso tantissimi momenti di festa e di studio. Desidero ringraziarli perché ho sempre potuto contare su di loro per qualsiasi cosa, e pur impegnandomi, non sono mai riuscito a ricambiare quanto ricevuto. Il buon vecchio e caro Genna. A lui va un ringraziamento speciale, perché è il compagno di trincea che tutti vorrebbero avere. Buona parte delle battaglie intraprese e vinte, sono state possibili solo perché combattute assieme a lui. Come dimenticare i compaesani e ritrovati amici di Forlì e di Rimini. In particolare mi riferisco a mr. Manenti, e l'allegra brigata di via Biondini; a Daniele Puccia ed Andrea Voi. Grazie a loro ho potuto trascorrere innumerevoli serate anti stress e di puro divertimento. Un grazie desidero porgerlo a tutti i ragazzi con cui ho condiviso la casa. Gli ultimi in ordine cronologico: Federico e Carlo. Grazie per avermi sopportato e supportato in questi ultimi mesi.

Avrei voluto citare tutti quanti, ma poiché ciascuno di loro è stato speciale, la lista sarebbe stata lunghissima.

In fine, desidero ringraziare quanti non hanno mai creduto in me e nelle mie capacità; chi mi ha messo a dura prova e cercato di rendere la mia vita difficile. E' proprio grazie a loro che ho potuto sviluppare una certa "reiezione" che mi aiuta tutti i giorni a superare gli ostacoli.

"Non ho un'intelligenza superiore o uno sguardo affascinante. Non riesco a correre 1 miglio in meno di 6 minuti o a conquistare una platea con un discorso. Ma riesco comunque ad avere successo perché continuo a lavorare quando tutti vanno a dormire."

Greg Evans.

Bibliografia

- [1] Bevilacqua, A. (2011). - *BIOIMMAGINI E VISIONE LM - Università di Bologna*.
- [2] *CVG*. (s.d.). Tratto da <http://cvg.deis.unibo.it/>
- [3] *Debut*. (s.d.). Tratto da <http://www.nchsoftware.com/capture/index.html>
- [4] *Gantt project*. (s.d.). Tratto da <http://www.ganttproject.biz/>
- [5] Gardini, M. (s.d.). Relazione di Tirocinio presso IRST.
- [6] Gonzalez, R. C. (2001). *Gonzalez:2001:DIP:559707*. Boston, MA, USA, Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.
- [7] *ImageJ*. (s.d.). Tratto da <http://rsbweb.nih.gov/ij/>
- [8] Mathias Prokop, M. G. (2010). *Tomografia computerizzata - Spirale e multistrato*. Aart J. Van Der Molen, Conella Schaefer-Prokop, Lorenzo Bonomo.
- [9] *Matlab*. (s.d.). Tratto da <http://www.mathworks.it/products/matlab/>
- [10] *Medical Dictionary*. (s.d.). Tratto da <http://medical-dictionary.thefreedictionary.com/CT+number>
- [11] *NetBeans*. (s.d.). Tratto da <http://netbeans.org/>
- [12] Petralia, G. a. (2010). CT perfusion in solid-body tumours. Part I: technical issues. *Radiol Med* , 843-857.
- [13] *PMA*. (s.d.). Tratto da <http://asist.umin.jp/index-e.htm>

[14] SANTEC/TUDOR. (s.d.). Tratto da
<http://santec.tudor.lu/project/optimage/dicom/start>

[15] *Standard DICOM*. (s.d.). Tratto da <http://medical.nema.org/>

[16] Sun. (s.d.). *Java*. Tratto da <http://www.java.com/it/>

[17] Viewer, T. (s.d.). Tratto da
<http://www.teamviewer.com/it/index.aspx>

Indice delle Figure

Figura 1. Foto di una CT.....	13
Figura 2:Tecniche di acquisizione in assiale (sinistra) e volumetrica	14
Figura 3: esempio sequenza immagini CT.....	15
Figura 4: Header File letto con Matlab	19
Figura 5: Visualizzazione di una immagine numerica rappresentata in matrice di 8x8x2 bit.....	21
Figura 6: Operazione di posizionamento del "livello" e della "finestra" in CT	21
Figura 7: Pellicola fotografica 35mm	27
Figura 8: Spettro elettromagnetico e alcune informazioni sulle lunghezze d'onda.....	28
Figura 9: esempio immagine come matrice di pixel e relativi valori.....	28
Figura 10: esempio immagine come matrice di pixel. Visualizzazione valori.....	29
Figura 11: Colori (RGB) e relativa sovrapposizione	29
Figura 12: Immagine originale e scomposizione nei tre canali RGB	30
Figura 13: Immagine in gray scale.....	30
Figura 14: scala di grigi	31
Figura 15: CT epatica con tracciamento ROI su metastasi.....	32
Figura 16: Immagine Originale campione	33
Figura 17: Immagine con effetto Edge Detectors	34
Figura 18: Esempio istogramma	34
Figura 19: Immagine prima (sinistra) e dopo la sogliatura.....	35
Figura 20: Immagine esempio (sinistra), maschera, dato estratto con la maschera	36
Figura 21: Foto ricordo al termine del "Translational Research in Oncology".	38
Figura 22: Seduta Manutenzione ordinaria TC Philips Presso IRST. Da sinistra verso destra: Mattia Gardini, Silvia Malavasi, Marco Prattichizzo, Josè Falla.....	39
Figura 23: Software Philips per lo studio perfusionale.....	39
Figura 24: Screenshot di una sessione di PMA	40
Figura 25: Philips DICOM Viewer.....	41
Figura 26: Stralcio di grafo di Gantt del progetto	44
Figura 27: Interfaccia al lancio del software.....	61
Figura 28: GUI del Fade Maker.....	62
Figura 29: GUI di ricerca e visualizzazione immagini da database.....	66
Figura 30: GUI del "Modifica Estensione"	67
Figura 31: interfaccia CT W&L.....	67
Figura 32: screenshot di una sessione del software PERFECT	89
Figura 33: Fade generato con "Fade Maker" tra	90
Figura 34: Fade generato con "Fade Maker"	90
Figura 35: "Combined" di due Fade di esempio generata con il "Fade Maker".....	90

Figura 36: Seduta di acquisizione e validazione dati sperimentali presso IRST. Nella foto il Dott. Domenico Barone	91
Figura 37: foto ricordo di fine meeting del CVG presso ARCES	94