



UNIVERSITÀ DI PAVIA

Facoltà di Ingegneria

Dipartimento di Ingegneria Industriale e dell'Informazione

Corso di Laurea Magistrale in Bioingegneria.

Docente:

Prof. **Giovanni Magenes**

Relazione a cura di:

Vittoria Bianchi

« Ciò che dobbiamo imparare a fare,
lo impariamo facendolo »

- Aristotele

ESERCITAZIONI “HANDS ON”

Corso di Bioimmagini Multimodali
A.A. 2022/2023

Indice

1 REGISTRAZIONE DI IMMAGINI	2
1.1 Introduzione	2
1.2 Metodo dei Centroidi	4
1.3 Metodo dei Markers	5
1.4 Metodo PAR	6
1.5 Metodo della “Forza Bruta”	7
1.6 Metodo del Gradiente - SSD	8
1.7 Metodo della Cross-Correlazione	8
1.8 Metodo RIU	9
1.9 Metodo della Mutua Informazione	10
2 SEGMENTAZIONE DI IMMAGINI	12
2.1 Introduzione	12
2.2 Analisi Senza Componente Ossea	13
2.3 Analisi Con Scatola Cranica	14
3 ANALISI DI VOLUMI	17
3.1 Introduzione e Metodologia	17
3.2 Risultati Ottenuti	17
3.3 Curiosità Personale	20

Capitolo 1

REGISTRAZIONE DI IMMAGINI

1.1 Introduzione

Per riscontrare analogie e differenze tra due immagini dello stesso paziente, ad esempio per notare l'evoluzione di un tumore nel tempo, o di pazienti diversi, come uno sano e uno malato, è utile sovente la **sovraposizione di due o più immagini** (operazione di Fusione).

A tale scopo le immagini **devono riferirsi allo stesso sistema di riferimento**. L'operazione di registrazione si rende dunque necessaria tutte le volte in cui si vogliano porre due immagini nello stesso sistema di riferimento, ma esse siano state acquisite in due sistemi di riferimento differenti. Per cui la registrazione di fatto è l'operazione che va a trovare una **trasformazione geometrica in grado di mettere in corrispondenza i punti delle diverse immagini**.

Se si considerano come esempio le immagini A e B riportate di seguito, si può notare come esse non siano perfettamente sovrapponibili (Fig. 1.1 (a)):

l'immagine B risulta infatti rototraslata rispetto alla A, e la differenza tra di esse non restituisce l'immagine nulla (schermo nero). Tutto ciò è rilevabile anche a livello di istogramma congiunto che non appare come una retta definita, ma come una sfumatura di livelli di grigio differenti.

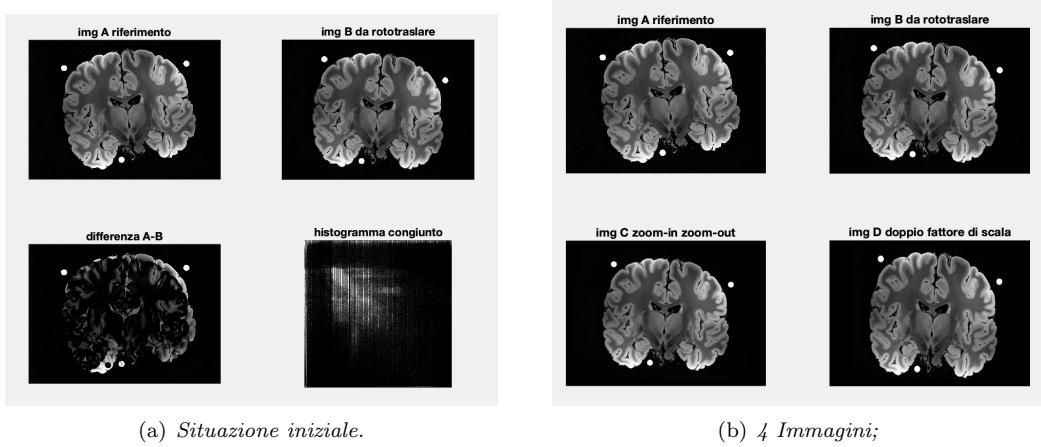


Figura 1.1:

L'obiettivo della prima esercitazione consiste dunque nel creare un programma per la registrazione delle diverse immagini a disposizione. In particolare vengono descritte e commentate le sezioni principali del codice Matlab, che per comodità non viene riportato nella sua interezza.

Esso risulta tuttavia disponibile all'interno della cartella Google Drive recante il mio nome: link alla cartella.

Vengono considerate le immagini 'Coronal' (riportate in figura 1.1 (b)), tra le quali:

- Immagine A: immagine originale da considerare come riferimento
- Immagine B: immagine rototraslata
- Immagine C: immagine rototraslata con singolo fattore di scala di tipo "Zoom-In Zoom-Out"
- Immagine D: immagine rototraslata con fattori di scala

Vengono utilizzati **8 metodi** di registrazione che vengono analizzati nel dettaglio nelle sezioni apposite. Per alcuni di essi, definiti appunto '**Point Based**', è necessario conoscere la **posizione dei Markers** esterni che sono stati accuratamente posizionati all'interno delle diverse immagini: a tale scopo viene attuata la seguente procedura.

A seguito della lettura delle immagini viene applicata una **maschera di convoluzione**. Al posto di utilizzare il **filtro Laplaciano Classico** [F1], viene utilizzata una sua leggera **variante** [F2] che sembra essere più efficace:

$$F1 = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad F2 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad (1.1)$$

Le immagini così ottenute subiscono un'operazione di '**fill holes**', operatore di morfologia matematico che riempie le zone povere di informazione, e in seguito un'operazione di **binarizzazione**; ottenendo il seguente risultato.

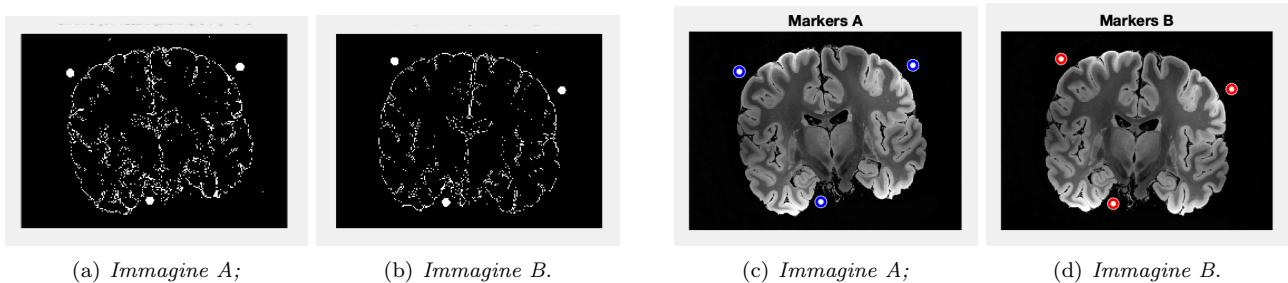


Figura 1.2: *Posizione dei Markers*.

Tramite la **Trasformata di Hough**, vengono trovate in ogni immagine le coordinate dei **tre cerchi più significativi** che corrispondono alle coordinate dei **Markers**; esse vengono poi plottate insieme alla relativa immagine per sottolinearne la posizione (Fig. 1.2 (c) e (d)).

Tale posizione risulta però essere riferita al **sistema di riferimento "up-sx"** (in alto a sinistra) e deve essere riportata in **coordinate cartesiane** (centro del sistema nel centro dell'immagine). A tale scopo viene utilizzata la funzione '**floor**' di Matlab che permette di riscalare la posizione di ogni pixel rispetto alle dimensioni totale dell'immagine stessa. Viene riportato il codice Matlab utilizzato a tal proposito.

```
marker_imgA(:,1) = centers_rif(:,1) - floor(pixel(2)/2);
marker_imgA(:,2) = floor(pixel(1)/2) - centers_rif(:,2);
marker_imgB(:,1) = centers_float(:,1) - floor(pixel(2)/2);
marker_imgB(:,2) = floor(pixel(1)/2) - centers_float(:,2);
```

1.2 Metodo dei Centroidi

Il **Metodo dei Centroidi** fa parte dei **Metodi Geometrici**, considera le immagini A e B (rototraslazione senza fattori di scala) e si basa sul calcolo dei coefficienti di rotazione e traslazione a partire dalle posizioni relative dei rispettivi **centri geometrici** dei **Markers** all'interno delle due immagini.

Dove la posizione del centro geometrico dei triangoli definiti dalle posizioni dei markers A,B e C è data da:

$$X_C = \frac{X_A + X_B + X_C}{3} \quad (1.2)$$

$$Y_C = \frac{Y_A + Y_B + Y_C}{3} \quad (1.3)$$

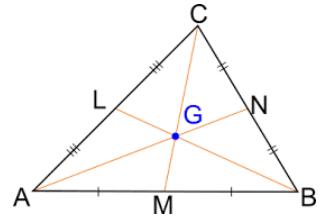
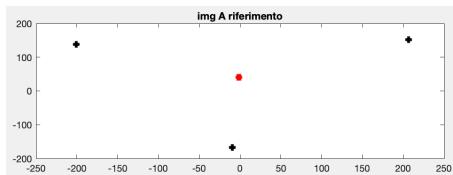
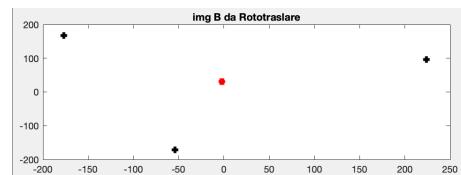


Figura: *Calcolo del Centroide*.

Seppur esista l'apposita funzione implementata in Matlab, a scopo didattico si è deciso di creare una funzione apposita, ***MyCentroide***, disponibile all'interno della cartella drive, che restituisce la posizione del Centroide. Essa può volendo essere implementata attraverso l'inserimento del valore di *W* (*weight*) nel caso in cui si voglia calcolare il centro di massa dell'oggetto. Vengono graficati (Fig. 1.3) i Markers (in nero) e i rispettivi centroidi (in rosso).



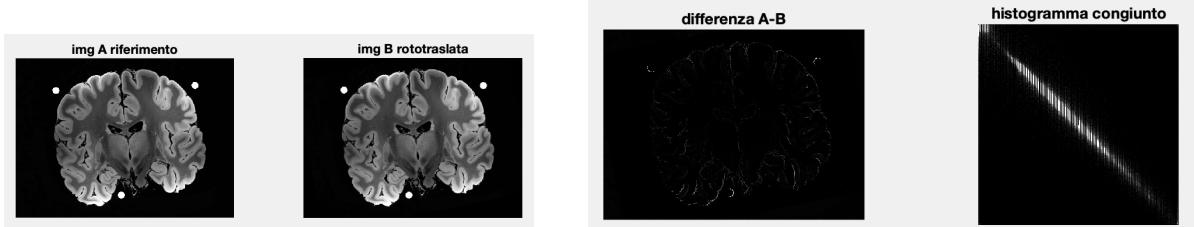
(a) *Immagine A*;



(b) *Immagine B*.

Figura 1.3: *Posizione di Markers e Centroidi*.

Le varie fasi dell'algoritmo non vengono approfondite poiché ricalcano fedelmente quanto spiegato nelle slide del corso. Il metodo funziona molto bene in quanto la differenza delle due immagini risulta coincidere con l'immagine nulla, dimostrando la perfetta sovrapponibilità delle due.



(a) *Rototraslazione delle Immagini*;

(b) *Sottrazione e Istogramma Congiunto*.

Figura 1.4: *Risultati con Metodo dei Centroidi*.

Vengono riportati i parametri ottenuti che sono stati applicati all'immagine B tramite **Trasformazione Affine** per dare luogo ai risultati riportati in figura 1.4.

Parametri metodo dei CENTROIDI					
alpha	tx	ty	s1	s2	0
-0.2045	7.5435	-11.8090	1.0000	1.0000	0

1.3 Metodo dei Markers

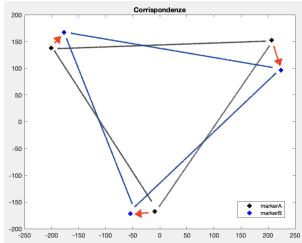


Figura: *Metodo dei Markers.*

Il **Metodo dei Markers** fa parte anch'esso dei **Metodi Geometrici**, considera le immagini A e B (rototraslazione senza fattori di scala) e basa il calcolo dei coefficienti a partire dalle posizioni dei **Markers corrispondenti** all'interno delle due immagini (metodo '**Point Based**'). Prendendo l'immagine A come riferimento, le posizioni dei markers nell'immagine B vengono viste come frutto di uno spostamento a partire da quelle iniziali dell'immagine A.

La chiave di volta sta quindi nell'**assegnazione delle corrispondenze** che viene fatta minimizzando le distanze euclidiene dei punti di riferimento all'interno delle due immagini, come riportato nel codice Matlab seguente.

```

for i=1:height(marker_imgA)
    for j=1:height(marker_imgB)

        % trovo le distanze
        dist(j) = sqrt((marker_imgA(i,1)-marker_imgB(j,1))^2 +...
            (marker_imgA(i,2)-marker_imgB(j,2))^2);

    end
    % minimizzo la distanza
    [~,indice_dist_min(i)] = min(dist);
end

% Matrice delle corrispondenze
corr = [marker_imgA(1,:),marker_imgB(indice_dist_min(1),:);
        marker_imgA(2,:),marker_imgB(indice_dist_min(2),:);
        marker_imgA(3,:),marker_imgB(indice_dist_min(3),:)];

```

Si ottengono infatti i seguenti valori dei parametri, che risultano essere molto simili a quelli trovati col Metodo dei Centroidi.

Parametri metodo MARKERS						
alpha	tx	ty	s1	s2	0	0
-0.2109	7.5391	-11.6066	1.0000	1.0000	0	0

L'applicazione dei parametri tramite Trasformazione Affine porta ai seguenti risultati che possono essere considerati più che soddisfacenti, forse i migliori tra tutti i metodi utilizzati. La sottrazione delle due immagini riconduce all'immagine nulla e l'istogramma congiunto sembra essere una retta perfettamente coesa.

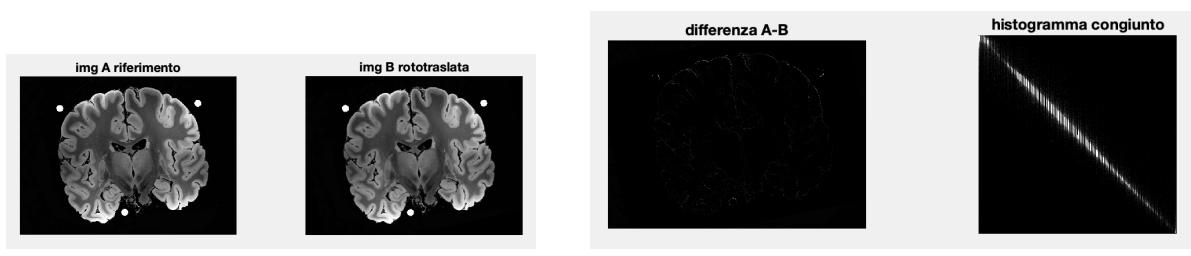


Figura 1.5: *Risultati con Metodo dei Markers.*

1.4 Metodo PAR

Il **Metodo PAR** (*Principal Axes Registration*) si basa sul calcolo dello spostamento relativo dei principali **Assi di Inerzia Ix e Iy** delle due immagini. Non vi è qui la necessità di avere dei punti di riferimento esterni, ma risulta fondamentale invece individuare la **posizione del baricentro geometrico dell'oggetto, che coinciderà con il punto nel quale andrà posto l'origine degli assi di inerzia**.

Questo non va confuso però con il centro di massa, che includerebbe nel calcolo la componente di densità (o massa) dell'oggetto stesso, qui rappresentata dal valore di grigio che l'immagine assume nel singolo pixel. L'oggetto viene approssimato come una **lastra piana sottile e di densità uniforme**. A tale scopo, per evitare di considerare nel calcolo del centroide anche le componenti di sfondo o di rumore, vengono dunque utilizzate le immagini A e B (rototraslazione senza fattori di scala) ma solo dopo un'elaborazione preliminare tramite il **Software ImageJ**.

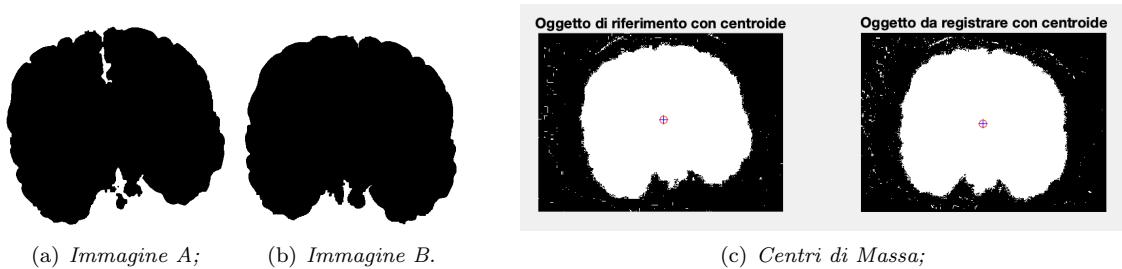


Figura 1.6: Immagini per Metodo PAR.

A seguito della **rimozione dei Markers** esterni ('anneriti' manualmente), l'immagine viene **binarizzata** per approssimarla ad una lastra uniforme, ma solo dopo essere stata **sogliata**, in modo da **isolare l'oggetto di interesse dallo sfondo**.

Siccome le dimensioni dei due oggetti nelle due immagini non variano in maniera sostanziale prima e dopo l'elaborazione tramite *ImageJ*, si assume che le posizioni dei centri di massa trovate **coincidano con le posizioni dei centri di massa delle immagini originali**.

Inoltre, nonostante esista una funzione implementata in Matlab per il calcolo del centro di massa, si decide di ricavare tale valore anche manualmente a scopo didattico. Il grafico in figura 1.12 (c) (sopra) mostra le immagini A e B binarizzate, all'interno delle quali viene sottolineata la posizione del Centroide. La croce blu rappresenta il punto trovato tramite la funzione '**regionprops**' di Matlab, mentre il cerchio rosso la posizione trovata '**manualmente**'.

Non viene riportata l'intera procedura, ma si può osservare come in questo caso vengano rototraslate entrambe le immagini: sia quella di riferimento (A), sia quella da rototraslare (B).

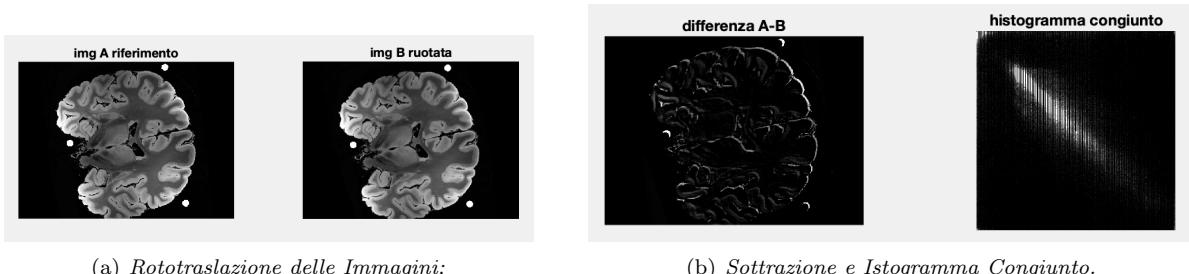


Figura 1.7: Risultati con Metodo PAR.

I parametri ottenuti si discostano da quelli ottenuti col Metodo dei Centroidi, proprio a causa della componente di rotazione e traslazione di entrambe le immagini e non solo di una rispetto all'altra. Tuttavia, la sovrapposizione delle due immagini in questo caso non risulta essere perfetta ed è confermata dalla poca coesione all'interno dell'istogramma congiunto. Esso infatti è indice della non convergenza dei risultati verso il valore ottimale.

Parametri metodo PAR					
alpha	tx	ty	s1	s2	0
1.2377	8.9884	9.9064	1.0000	1.0000	0

1.5 Metodo della “Forza Bruta”

Il **Metodo della “Forza Bruta”** (*Brute Force*) o di **Ottimizzazone** è stato applicato considerando le immagini A e B (rototraslazione senza fattori di scala). Tale metodo fa parte dei metodi definiti **Intensity Based** poichè non prevede più l'ausilio di markers esterni e non considera l'immagine come 'piatta', ma implementa la valutazione di un **funzionale di costo basato sui valori di grigio dei pixel dell'immagine stessa**.

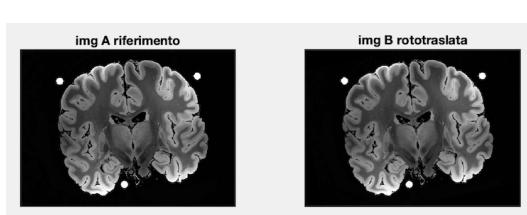
Per ogni parametro (α , t_x , t_y) viene considerato un range di valori che esso può assumere. Vengono eseguiti 3 cicli annidati su tali range per trovare il valore ottimo di ognuno dei parametri in esame. Come accennato, la ricerca di tale valore ottimo si basa sulla minimizzazione di un funzionale di costo dato nello specifico dalla **somma al quadrato delle differenze tra l'immagine di riferimento (A) e l'immagine B rototraslata applicando il valore corrente dei parametri** (ancora una volta tramite Trasformazione Affine).

Il funzionale da minimizzare viene riportato di seguito ricordando che il singolo pixel p su cui si basa il calcolo deve appartenere ad Ω , zona di intersezione delle due immagini.

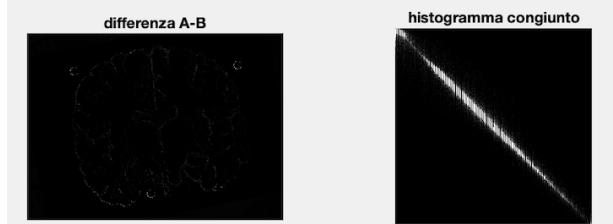
$$E(T) = \sum_{p \in \Omega} [ImgA(p) - ImgB(p; T)]^2 \quad (1.4)$$

I valori trovati si discostano leggermente da quelli riscontrati in precedenza ma producono ancora una volta un risultato degno di nota, riscontrabile soprattutto a livello di istogramma congiunto.

Parametri del metodo FORZA BRUTA					
alpha	tx	ty	s1	s2	0
-0.2094	8.0000	-12.0000	1.0000	1.0000	0



(a) Rototraslazione delle Immagini;



(b) Sottrazione e Istogramma Congiunto.

Figura 1.8: Risultati con Metodo della “Forza Bruta”.

1.6 Metodo del Gradiente - SSD

Il **Metodo del Gradiente** non è un vero e proprio metodo a sé stante ma si basa sul miglioramento del metodo precedentemente visto della cosiddetta “Forza Bruta” tramite il **calcolo del gradiente** (derivata dell’immagine).

In particolare viene applicato l’**Operatore di Sobel** (sia in verticale sia in orizzontale) per effettuare ‘**edge detection**’, ovvero per trovare i bordi nei punti in cui il gradiente dell’immagine I è massimo, utilizzando l’approssimazione di Sobel alla derivata.

$$S_O = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} S_V = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (1.5)$$

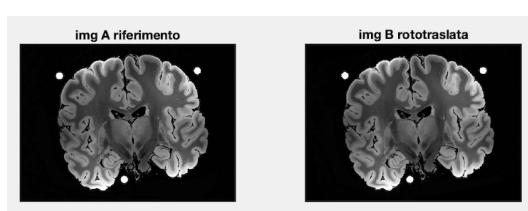
Nonostante ci si aspetterebbero risultati significativamente diversi tra i due metodi, il Metodo della “Forza Bruta” e il suo miglioramento SSD portano in questo caso agli **stessi risultati**: i valori dei parametri coincidono.

Parametri del metodo FORZA BRUTA

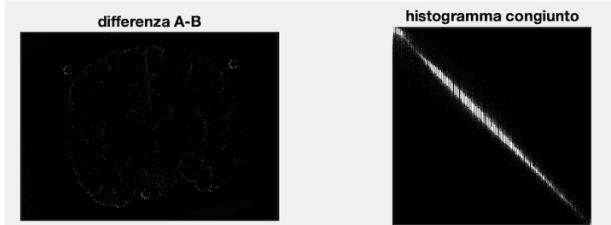
alpha	tx	ty	s1	s2	0
-0.2094	8.0000	-12.0000	1.0000	1.0000	0

Parametri metodo GRADIENTE

alpha	tx	ty	s1	s2	0
-0.2094	8.0000	-12.0000	1.0000	1.0000	0



(a) Rototraslazione delle Immagini;



(b) Sottrazione e Istogramma Congiunto.

Figura 1.9: Risultati con Metodo del Gradiente.

1.7 Metodo della Cross-Correlazione

Il **Metodo della Cross-Correlazione** fa anch’esso parte dei metodi **Intensity Based** e considera le immagini A (riferimento) e questa volta D (rototraslazione con doppio fattore di scala).

Come dice il nome stesso, basa la ricerca dei migliori parametri sulla **minimizzazione del coefficiente di Cross-Correlazione**.

$$CC = \frac{\sum_i (I_A(i) - \hat{I}_A)(I_D(i) - \hat{I}_D)}{\sqrt{\sum_i (I_A(i) - \hat{I}_A)^2 \sum_i (I_D(i) - \hat{I}_D)^2}} \quad (1.6)$$

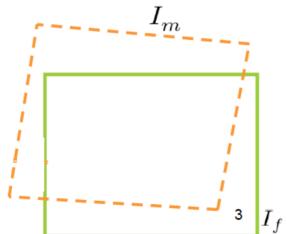


Figura: Intersezione.

Come sottolineato nella figura alla pagina precedente, tale formula considera però solamente la **zona di intersezione delle due immagini** per cui è necessario applicare una maschera che ricavi tale zona, in questo caso ponendo a zero i pixel che non appartengono ad essa. (Viene riportato la Funzione Matlab utilizzata).

```
%% MASCHERA
% funzione che trova omega intersezione delle due immagini

function [maschera] = crea_maschera_intersezione(alpha,tx,ty,sx,sy,sx,y,dim)

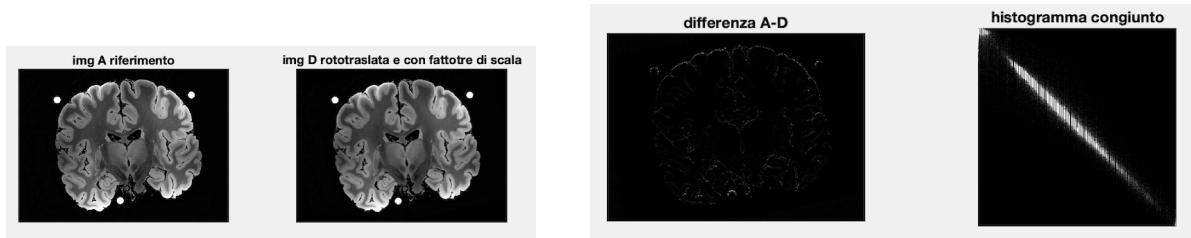
maschera = double(optimized_affine_trasformation(uint8(ones(dim)),[alpha,tx,ty,sx,sy,0]));

% i pixel fuori dalla maschera creata hanno valore NaN
maschera(maschera==0) = NaN;
end
```

I parametri trovati appaiono coerenti con quelli riscontrati in precedenza. Tuttavia si può notare come in questo caso compaiano due fattori di scala che risultano essere entrambi diversi dal fattore neutro pari a 1.

Parametri metodo della CROSS CORRELZIONE
alpha tx ty s1 s2 0
-0.2094 8.0000 -10.0000 1.0750 1.0750 0

Si possono notare buoni risultati anche se in questo caso i metodi '**Point Based**' sembrano comunque essere maggiormente performanti.



(a) *Rototraslazione delle Immagini;* (b) *Sottrazione e Istogramma Congiunto.*

Figura 1.10: *Risultati con Metodo della Cross-Correlazione.*

1.8 Metodo RIU

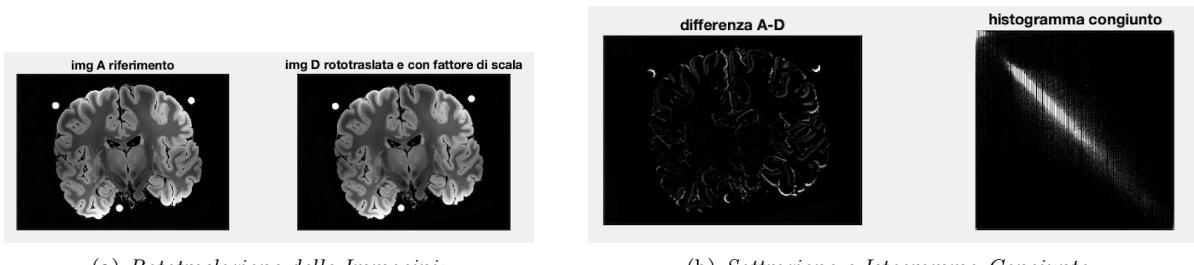
Il **Metodo RIU** (*Ratio Image Uniformity*) considera le immagini A (riferimento) e D (rototraslata con doppio fattore di scala). Viene calcolato inizialmente R che è il rapporto tra i pixel dell'immagine di riferimento (A) e quelli dell'immagine D rototraslata e riscalata secondo i parametri correnti. Si calcolano la media e la deviazione standard di R . Il **funzionale di costo RIU da minimizzare** è dato dal rapporto tra le due misure statistiche:

$$RIU = \frac{\sigma_R}{\mu_R} \quad (1.7)$$

Applicando il metodo si ottengono i seguenti risultati che non sembrano eguagliare l'eccezionalità vista in precedenza. Tutto ciò è visibile soprattutto a livello di sottrazione delle due immagini che non sembra coincidere con l'immagine nulla.

Già a colpo d'occhio si può notare come il problema principale risieda nel valore dell'angolo di rotazione che risulta troppo elevato, portando ad una rotazione eccessiva dell'immagine B. Tuttavia anche t_x e t_y differiscono dai casi precedenti e seppur si tratti di qualche pixel sono la causa dei risultati non ottimali che si riscontrano.

```
Parametri theta metodo della RIU
alpha      tx      ty      s1      s2      0
-0.2269    7.0000 -12.0000  1.0750   1.0750   0
```

(a) *Rototraslazione delle Immagini;*(b) *Sottrazione e Istogramma Congiunto.*Figura 1.11: *Risultati con Metodo della RIU.*

1.9 Metodo della Mutua Informazione

Il Metodo della Mutua Informazione fa parte di quei metodi che trovano il valore dei parametri a partire da **misure di similarità** basate sul calcolo della **densità di probabilità delle immagini**. In questo caso vengono considerate le immagini A (riferimento) e D (rototraslata con doppio fattore di scala).

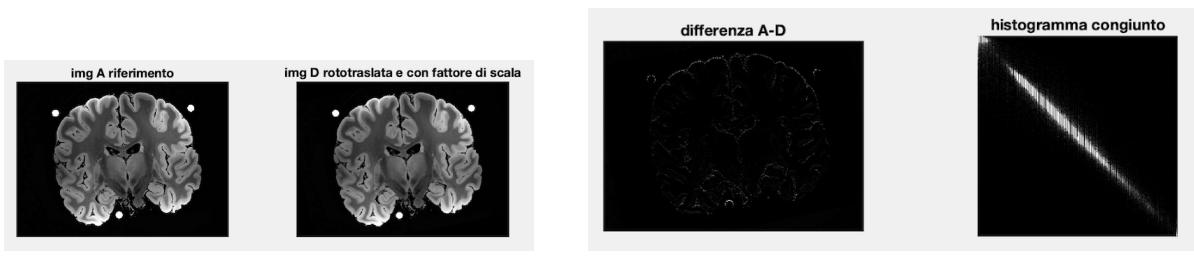
La mutua informazione è definita in termini di entropia:

$$E = \sum (P) \log_2(P); \quad (1.8)$$

In particolare nel calcolo compare sia l'entropia delle singole immagini, sia quella congiunta. In questo caso al posto della Mutua Informazione classica viene utilizzato il coefficiente normalizzato:

$$MI = \frac{E_A - E_{B(T)}}{E_{cong}} \quad (1.9)$$

Vengono riportati i risultati ottenuti, che possono essere considerati soddisfacenti.

(a) *Rototraslazione delle Immagini;*(b) *Sottrazione e Istogramma Congiunto.*Figura 1.12: *Risultati con Metodo della Mutua Informazione.*

Si ottengono i seguenti valori dei parametri che coincidono con quelli trovati col metodo precedente della Cross-Correlazione.

```
Parametri metodo della CROSS CORRELZIONE
alpha      tx      ty      s1      s2      0
-0.2094    8.0000 -10.0000  1.0750   1.0750   0
```

```
Parametri theta metodo della MUTUA INFORMAZIONE
alpha      tx      ty      s1      s2      0
-0.2094    8.0000 -10.0000  1.0750   1.0750   0
```

Per un paragone più immediato vengono riportati i valori di tutti i parametri trovati.

```
Parametri metodo dei CENTROIDI
alpha      tx      ty      s1      s2      0
-0.2045    7.5435 -11.8090  1.0000   1.0000   0
```

```
Parametri metodo PAR
alpha      tx      ty      s1      s2      0
1.2377    8.9884  9.9064  1.0000   1.0000   0
```

```
Parametri metodo MARKERS
alpha      tx      ty      s1      s2      0
-0.2109    7.5391 -11.6066  1.0000   1.0000   0
```

```
Parametri del metodo FORZA BRUTA
alpha      tx      ty      s1      s2      0
-0.2094    8.0000 -12.0000  1.0000   1.0000   0
```

```
Parametri metodo GRADIENTE
alpha      tx      ty      s1      s2      0
-0.2094    8.0000 -12.0000  1.0000   1.0000   0
```

```
Parametri metodo della CROSS CORRELZIONE
alpha      tx      ty      s1      s2      0
-0.2094    8.0000 -10.0000  1.0750   1.0750   0
```

```
Parametri theta metodo della MUTUA INFORMAZIONE
alpha      tx      ty      s1      s2      0
-0.2094    8.0000 -10.0000  1.0750   1.0750   0
```

```
Parametri theta metodo della RIU
alpha      tx      ty      s1      s2      0
-0.2269    7.0000 -12.0000  1.0750   1.0750   0
```

Capitolo 2

SEGMENTAZIONE DI IMMAGINI

2.1 Introduzione

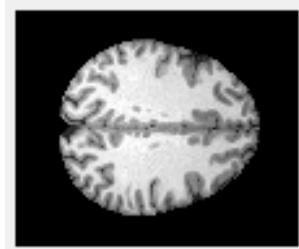
La segmentazione è il processo di **identificazione di strutture** che si trovano all'interno di una stessa immagine o di immagini differenti. Il riconoscimento e il conseguente isolamento delle diverse componenti anatomiche trova la sua utilità a livello clinico diagnostico e risulta essere sempre più importante in ambito biomedico.

Il Metodo utilizzato nella seconda esercitazione è il cosiddetto '**k-means**' e si basa sull'assegnazione di ogni punto dell'immagine ad un '**cluster**' sulla base della **minimizzazione della distanza** del punto considerato dal punto medio del cluster stesso. Nel caso dell'implementazione per immagini, **non** si tratta di una **distanza geometrica** ma di una distanza in termini di **livelli di grigio**.

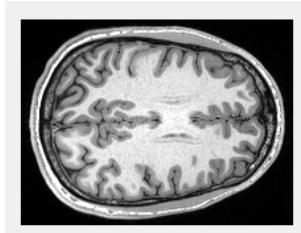
Lo scopo dell'esercitazione è la segmentazione delle diversi componenti di:

- sfondo;
- sostanza bianca;
- sostanza grigia;
- liquor.

L'esercitazione racchiude in realtà due tipologie di lavoro differenti: il primo caso prevede l'utilizzo di un volume al quale era già stata precedentemente **rimossa la componente ossea**, mentre nel secondo caso lo scopo era isolare anche la **componente di scatola cranica**, qui presente all'interno del volume. Quelle raffigurate sono due **slice** centrali che ben rappresentano il problema in esame.



(a) *Senza Ossa*;



(b) *Con Scatola Cranica*.

Figura 2.1: *Immagini da Segmentare*.

2.2 Analisi Senza Componente Ossea

Il Metodo k-means inizia con l'assegnazione per ogni cluster di un valore di grigio che rappresentante la **media iniziale** da cui far partire l'algoritmo. Per far convergere il tutto più velocemente si decide di prendere 4 valori che possano essere **rappresentativi dei 4 cluster scelti**.

Viene considerato l'istogramma della *slice* n° 56 (già vista in precedenza) poichè si tratta di una **generica slice centrale**. L'istogramma è ottenuto tramite il **Software ImageJ** (*Analyze → Histogram*), e vengono presi come centri i valori dei picchi principali sottolineati nell'immagine (escludendo il picco a 240 che appare privo di significato).

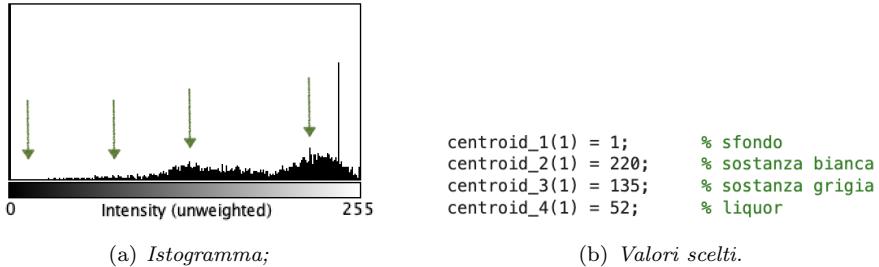


Figura 2.2: Scelta dei Centroidi dei Clusters.

Una volta inizializzati i Centroidi, si è preferito agire ***slice per slice*** al posto che sull'intero volume per limitare la quantità di cicli annidati e per evitare errori di indicizzazione. Si implementa dunque un **doppio ciclo annidato** su **max-iter** e sulla ***slice* i-esima** che risulta molto più semplice anche a livello computazionale rispetto al quadruplo ciclo (max-iter, x, y, z) come si otterrebbe invece agendo ***pixel per pixel***.

Vengono calcolate le distanze dei pixel dai Centroidi e si assegnano i punti al cluster con la minima distanza. A seguito di ogni assegnazione viene ricalcolato il nuovo centro e si procede 'spostando' nuovamente i pixel finché non si arriva a **convergenza**.

Per non incorrere in un ciclo infinito si attuano due accorgimenti: innanzitutto viene limitato il numero massimo di iterazioni (**max-iter**), poi viene posta una **soglia di tolleranza** per cui se tutte le differenze fra i Centroidi calcolati al passo corrente e i Centroidi calcolati al passo precedente sono inferiori alla soglia **tol-abs** (posta qui a 0.05) allora l'algoritmo viene interrotto. Ciò se lo spostamento dei Centroidi tra le diverse iterazioni risulta essere minimo non ha senso continuare ad iterare e si considera che l'algoritmo abbia raggiunto la convergenza. Questo è il motivo per il quale non viene aggiornato il valore del Centroide ad ogni iterata ma viene memorizzato costruendo un vettore di valori distinti.

Partendo dagli indici trovati vengono costruiti i diversi cluster. Anche in questo caso viene rappresentata la *slice* n° 56 ma la clusterizzazione riguarda l'intero volume.

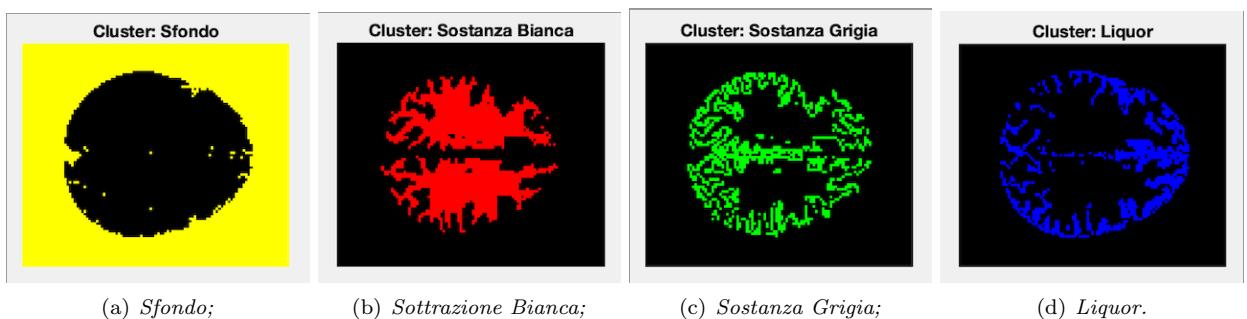


Figura 2.3: I 4 Clusters.

Per la rappresentazione viene quindi creato un cluster totale che permette di visualizzare le diverse componenti all'interno di una stessa immagine, ottenendo così la segmentazione di essa.

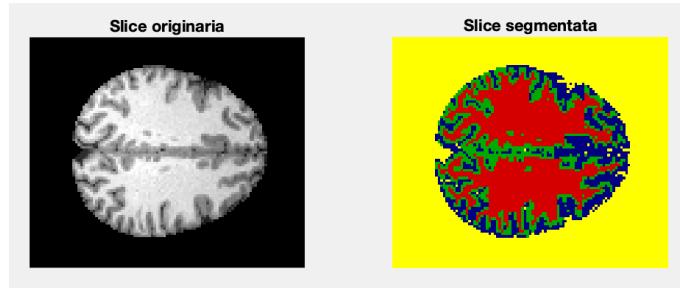
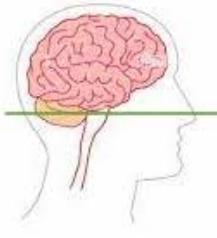


Figura 2.4: *Segmentazione.*

2.3 Analisi Con Scatola Cranica



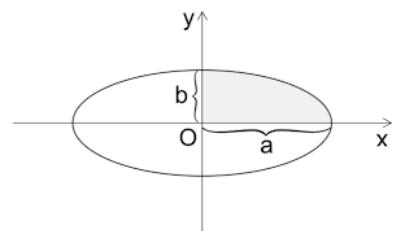
Nel caso più complesso in cui non sia stata rimossa la scatola cranica, vengono innanzitutto processate solamente le *slice* che si trovano sopra l'altezza del naso, ovvero dove effettivamente si trova il cervello.

Per isolare la componente ossea si è pensato di non utilizzare il classico ellissoide di rotazione (già implementato da molti compagni), ma un **ellisse parametrico che vari i suoi parametri al variare della slice** considerata.

Va tenuto a mente in tale implementazione che l'ordine delle *slice* all'interno del volume è '**bottom-up**' con la prima *slice* che si trova a livello della riga verde della figura e l'ultima *slice* che si trova ad altezza massima sopra la nuca.

Dato a il raggio maggiore, b quello minore, e (x,y) le coordinate del centro O , l'equazione dell'ellisse è data da:

$$\frac{(x - x_0)^2}{a^2} + \frac{(y - y_0)^2}{b^2} = 1 \quad (2.1)$$



Tramite il **Software Mango** sono stati misurati i '**raggi**' a e b **del cervello** rispettivamente nella prima *slice* (altezza naso), in una *slice* intermedia e nell'ultima *slice* (altezza nuca), così da poter costruire dei **range** entro i quali far variare i parametri.

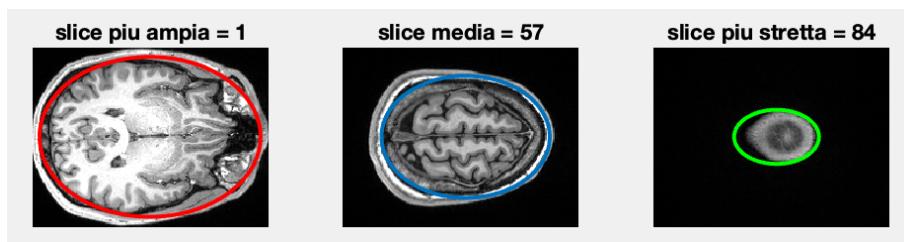


Figura 2.5: *Tre slice di riferimento.*

Inoltre è possibile notare come nell'ultimo terzo di *slice*, corrispondente alle *slice* in prossimità della nuca, il centro degli ellissi risulti essere leggermente traslato rispetto alla restante porzione di volume, pertanto è necessario applicare alcuni **aggiustamenti matematici**.

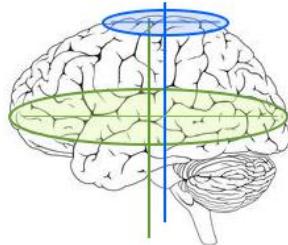


Figura 2.6: *Traslazione dei centri*.

La parametrizzazione sembra ben ricalcare le forme del cranio all'interno delle diverse *slice*. A titolo di esempio vengono presentate alcune *slice* campionate casualmente.

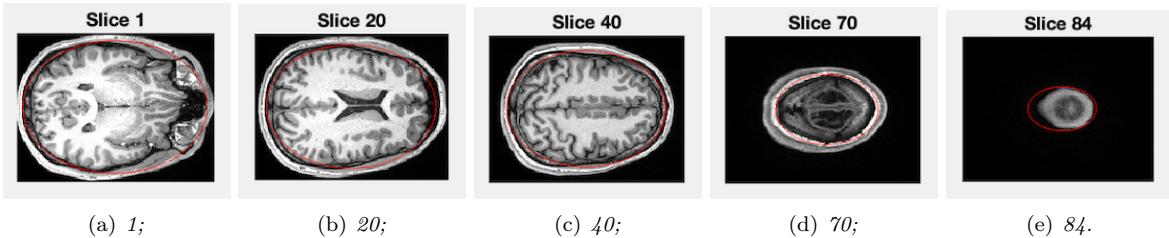


Figura 2.7: *Esempio di Risultati Ottenuti*.

Una volta individuate le coordinate degli ellissi e la loro formulazione vi è un primo processo di **isolamento della scatola cranica**: se i pixel stanno all'interno dell'ellisse vengono processati per la segmentazione, se stanno sul bordo dall'ellisse vengono automaticamente considerati nel cluster della componente ossea, così come se stanno fuori da esso vengono considerati parte dello sfondo.

I punti sul bordo dell'ellisse non vengono presi come perfettamente coincidenti con l'equazione che lo descrive, ma viene considerata una soglia per motivi computazionali e di arrotondamento dei valori decimali.

La restante parte dell'algoritmo è analoga a quella vista per il caso senza componente ossea e produce i seguenti risultati. Essi appaiono leggermente meno brillanti di quelli ottenuti nel caso precedente, ma tuttavia rimangono soddisfacenti.

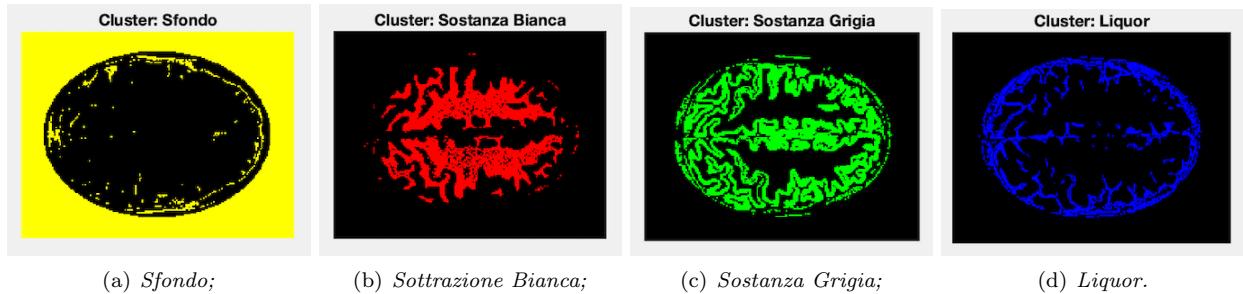


Figura 2.8: *I 4 Clusters*.

A sinistra viene rappresentata l'immagine originale della *slice* n° 40 considerata sufficientemente centrale; in centro viene riportato il risultato del *ciclo if* di riconoscimento della componente ossea tramite equazione dell'ellisse. A destra il risultato finale di segmentazione.

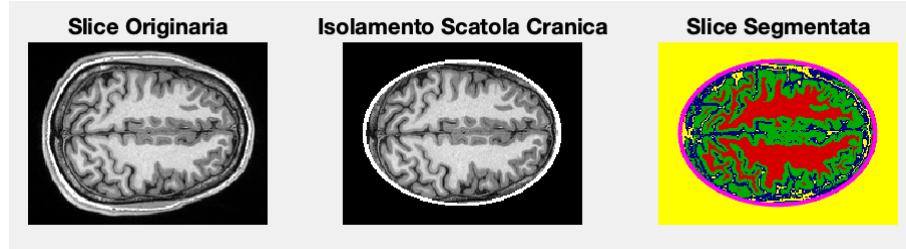


Figura 2.9: *Segmentazione*.

Capitolo 3

ANALISI DI VOLUMI

3.1 Introduzione e Metodologia

La terza esercitazione riguarda la **segmentazione** e la conseguente **analisi di volumi** attraverso l'utilizzo del **Software 3D Slicer**. In particolare sono stati considerate le CT volumetriche di due pazienti anonimizzati: una donna sana (MDC) e un uomo (AFG) con un grave tumore a livello renale.

Il **Metodo** utilizzato è il cosiddetto ***Grow from Seeds***. Viene applicata una pesante sogliatura che ha lo scopo di eliminare il più possibile tutto ciò che non appartiene all'organo da isolare, facendo attenzione invece di mantenere correttamente tutto l'organo target.

Si procede poi **'piantando il seme'** (*seed*), ovvero assegnando al volume da isolare una piccola porzione dell'organo con l'obiettivo di far riconoscere all'algoritmo quelli che sono i valori di grigio di appartenenza. Sempre per lo stesso motivo vengono punitinati, etichettandoli come **'other objects'** i livelli di grigio che non devono rientrare nel volume oggetto della segmentazione.

A partire dal seme, l'algoritmo procederà automaticamente **accrescendo** intorno ad esso il volume target, fermandosi quando non sono più rispettate le condizioni riguardanti i valori dei livelli di grigio.

3.2 Risultati Ottenuti

Gli organi scelti per la segmentazione sono **reni, milza e vescica**. Tra i tre il più complicato da analizzare è stata la vescica del paziente sano: infatti si può notare come nonostante il tentativo del miglioramento del contrasto, tale organo abbia valori di grigio molto simili agli organi e tessuti circostanti, rendendo ostico il suo isolamento.



Figura 3.1: *Vescica, pz. sano.*

Vengono riportati i risultati comparati dei due pazienti, per notare per prima cosa qualitativamente eventuali analogie e differenze tra di essi. Viene inizialmente riportata l'analisi relativa all'organo **vescica**. Le due forme sono comparabili anche se i livelli di grigio di partenza sono notevolmente differenti.

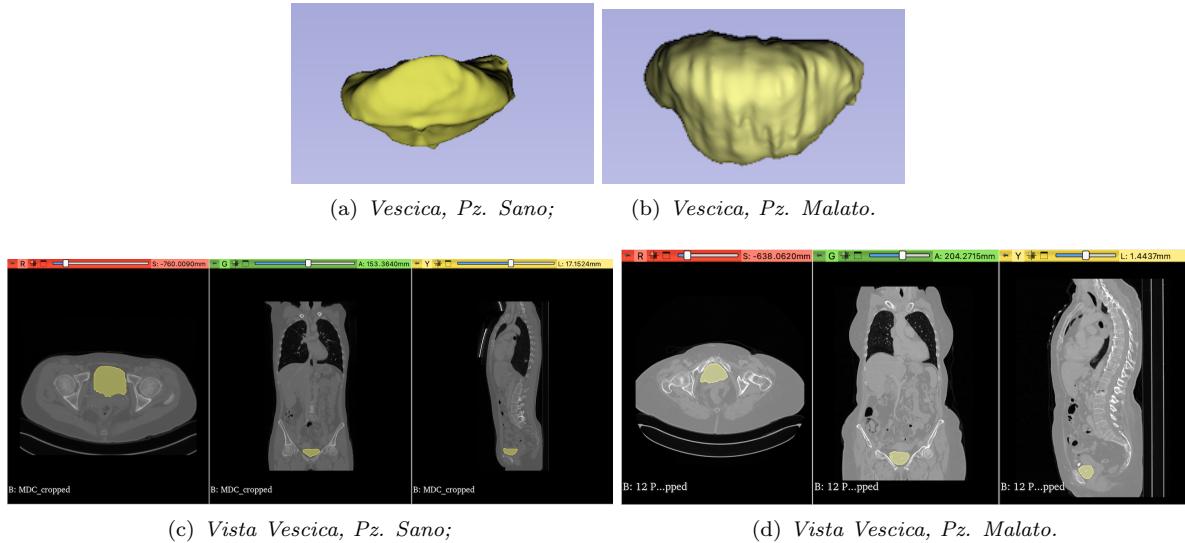


Figura 3.2: *Volumi e Viste, Organo Vescica.*

Tramite i Moduli **Radiomics** e **Segment Statistics** è possibile ottenere invece informazioni quantitative sui volumi considerati.

Come ci si aspetterebbe, la vescica della paziente donna è visibilmente più piccola di quella del paziente uomo e infatti presenta un **volume** ridotto. (Quanto riportato sotto la dicitura 'equivalenze' è una conversione pratica a scopo di una maggiore comprensione delle dimensioni in gioco, ma non compare all'interno dei risultati ottenibili tramite il Software *3d Slicer*).

Tuttavia la **superficie** della vescica della paziente sana risulta essere maggiore di quella del paziente malato, il che indica che abbia un andamento più irregolare e sia verosimilmente più frastagliata.

Non torna il calcolo del **numero di voxel** che compongono i due organi: ci si aspetterebbe che ad un volume maggiore corrispondesse un numero di voxel maggiore, ma non è così. Si ipotizza che la causa possa essere un errore nella segmentazione che ha dato origine a zone cave non visibili esternamente, tuttavia da un'analisi più approfondita delle tre viste bidimensionali non sembrano esserci buchi evidenti.

ORGANO	PAZIENTE	VOLUME [cm ³]	VOLUME [voxel]	SUPERFICIE [cm ²]	EQUIVALENZA
Vescica	Sano	78.143	141861	118.104	Cubo di lato 4
	Malato	87.585	111393	109.757	Cubo di lato 4.5

Figura 3.3: *Statistiche Vescica.*

Vengono poi considerate le segmentazioni dei **reni** e della **milza**. Mentre la milza sembra essere sostanzialmente equivalente nei due pazienti, per i reni non si può constatare lo stesso. Il rene sinistro del paziente malato presenta infatti un **grosso tumore** (rappresentato in viola scuro) che è stato segmentato in maniera indipendente rispetto all'organo di appartenenza per poterne misurare le caratteristiche fondamentali.

Vengono riportati volumi e le viste ottenuti.

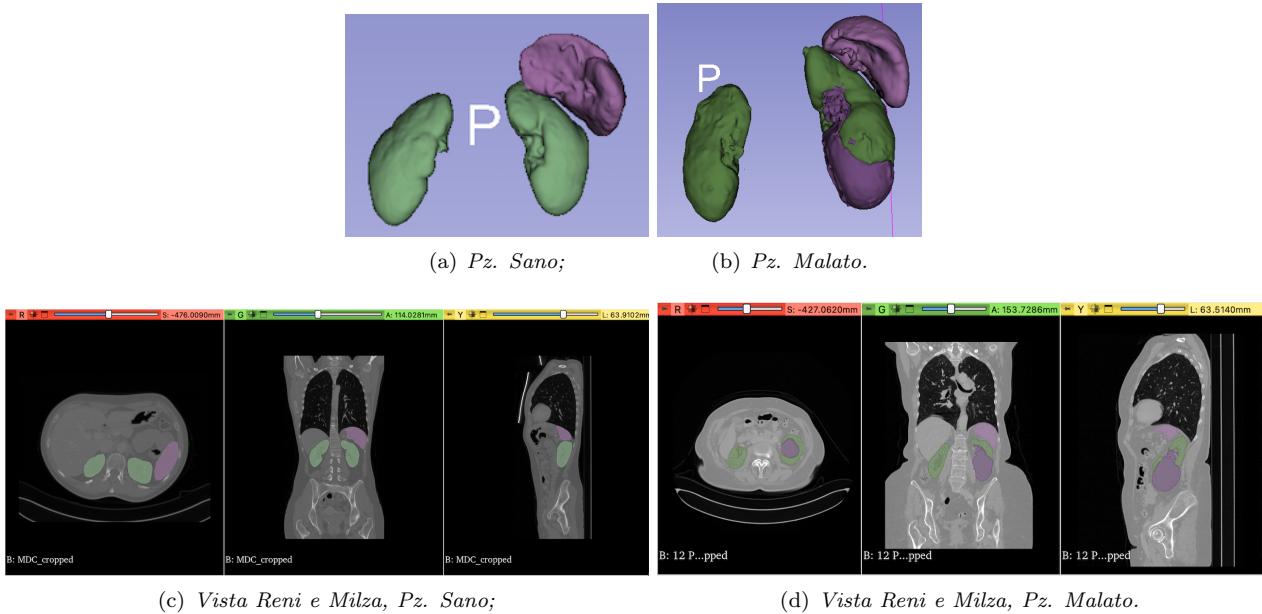


Figura 3.4: Viste Bidimensionali, Reni e Milza.

Le **grandezze dei due reni** nei pazienti non si discostano molto: normalmente i reni dei pazienti di sesso maschile sono molto più grossi, ma probabilmente in questo caso parte del volume è occupato dalla massa tumorale. Anche in questo caso la maggiore area superficiale è da attribuirsi ad una superficie probabilmente più irregolare (nonostante l'applicazione di filtri di *smoothing*).

La grandezza della **massa tumorale** non risulta essere molto inferiore a quella del rene stesso e indica uno stadio avanzato della malattia. Anche in questo caso il volume espresso in termini di voxel suscita delle perplessità.

ORGANO	PAZIENTE	VOLUME [cm ³]	VOLUME [voxel]	SUPERFICIE [cm ²]	EQUIVALENZA
Ren	Sano	354.641	643815	378.246	Cubo di lato 7
	Malato	348.907	443750	487.293	Cubo di lato 7
Tumore	Malato	262.16	333422	262.769	Cubo di lato 6.5
Milza	Sano	188.942	343006	206.728	Cubo di lato 6
	Malato	199.051	253158	216.622	Cubo di lato 6

Figura 3.5: Statistiche Reni, Milza e Massa Tumorale.

3.3 Curiosità Personale

La curiosità ha spinto ad un tentativo di segmentazione di più organi del paziente malato, anche al fine di osservare quali fossero gli organi più difficili da isolare e quali invece apparissero meno complessi in termini di differenze nei livelli di grigio.

Sono stati isolati: **la vena cava superiore** (blu), **l'aorta** (rosso), **l'arteria polmonare** (rosa), **il fegato** (marrone), **la milza** (viola), **il pancreas** (arancione), **i reni** (verde) con la **massa tumorale** (viola scuro), **la vescica** (giallo), **la prostata** (beige) e **la componente ossea** (panna) per meglio visualizzare la posizione dei diversi organi all'interno del corpo.



Figura 3.6: *Paziente Malato.*

I risultati possono essere considerati soddisfacenti nonostante le difficoltà riscontrate per quanto riguarda gli organi fegato e pancreas. Il primo perché presenta livelli di grigio simili ad altri organi e quindi facilmente confondibili da parte dell'algoritmo; il secondo perché risulta essere particolarmente danneggiato, probabilmente a causa della malattia.

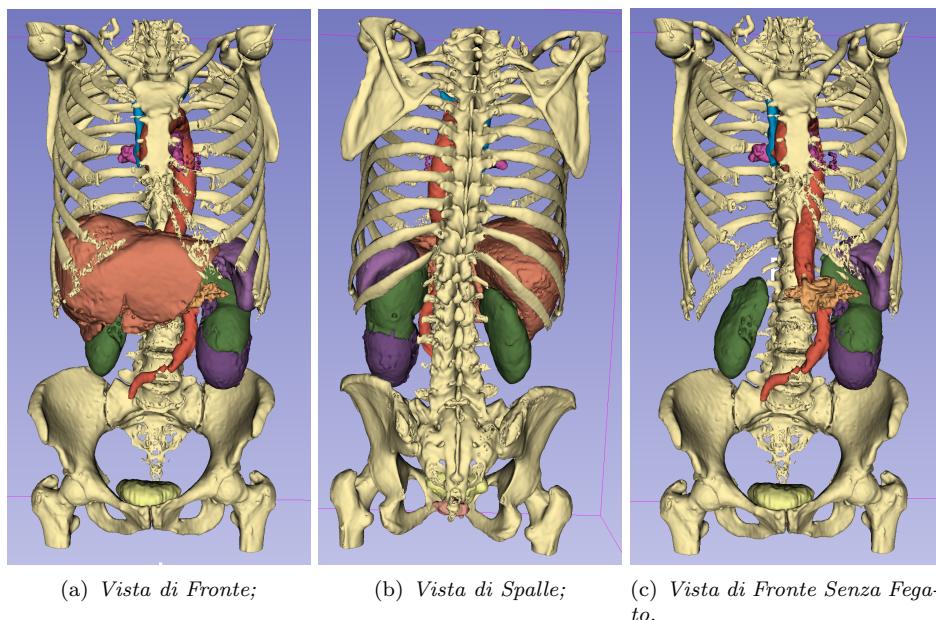


Figura 3.7: *Volumi pz. Malato.*