Navigazione e registrazione

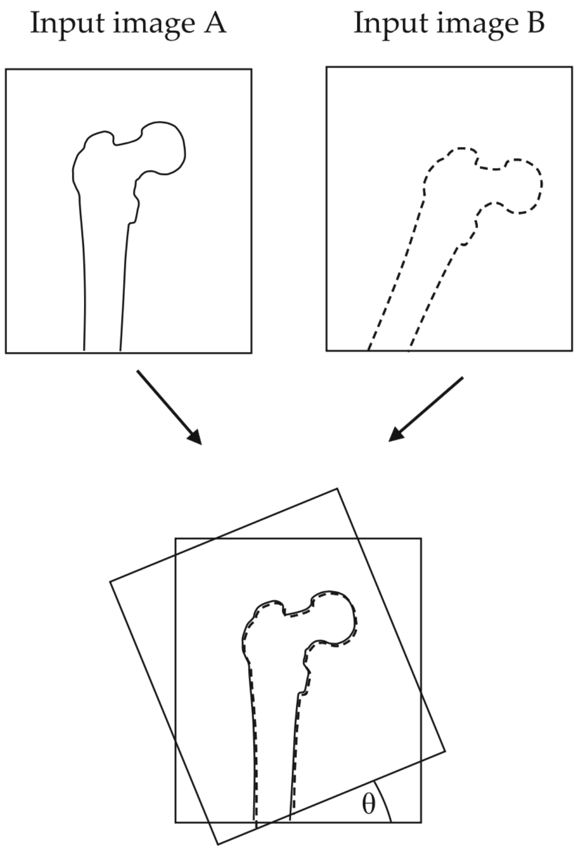
Introduzione

- La pianificazione del moto fa affidamento sul calcolo dello Jacobiano per interpolare i parametri dei giunti lungo una traiettoria

- La definizione di tale traiettoria di moto e velocità nello spazio operativo del robot necessita di interazione col mondo esterno e in particolare ti tecniche di *computer vision* e *image processing*

- Discuteremo in particolare di *calibrazione* e *registrazione* di immagini

Principio della registrazione

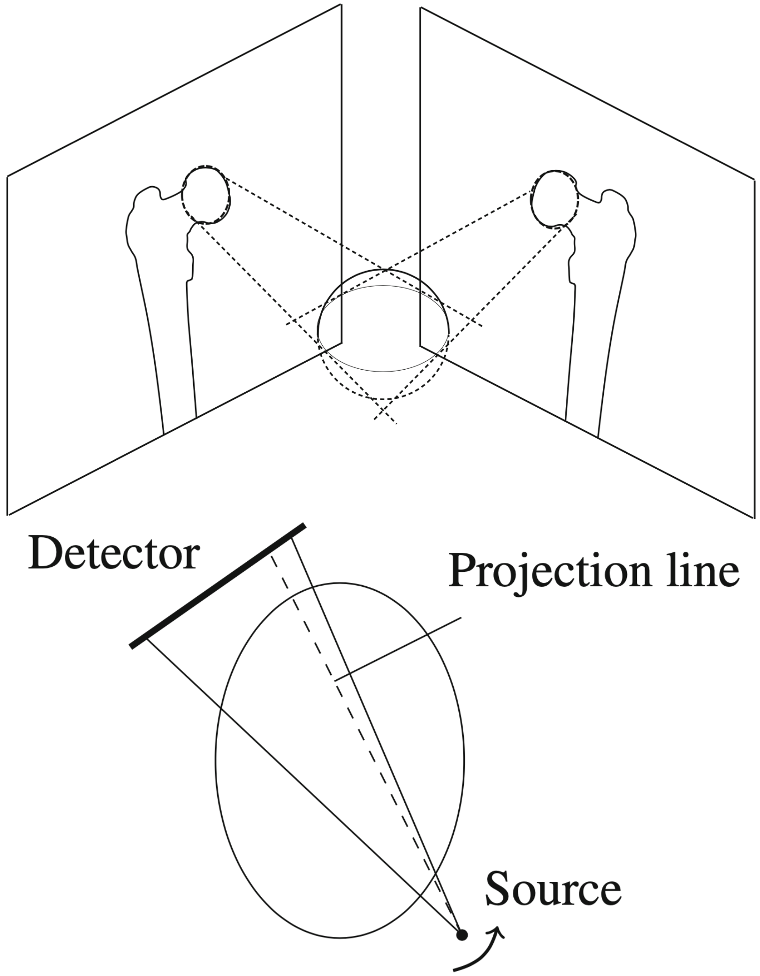


Output: θ, Δx, Δy

Tecniche per allineare diverse strutture, ad es. più viste di una RX oppure un robot e una telecamera

Individuare l'angolo θ e lo scostamento $\Delta x, \Delta y$ per allineare le due viste

Registrazione 2D-3D



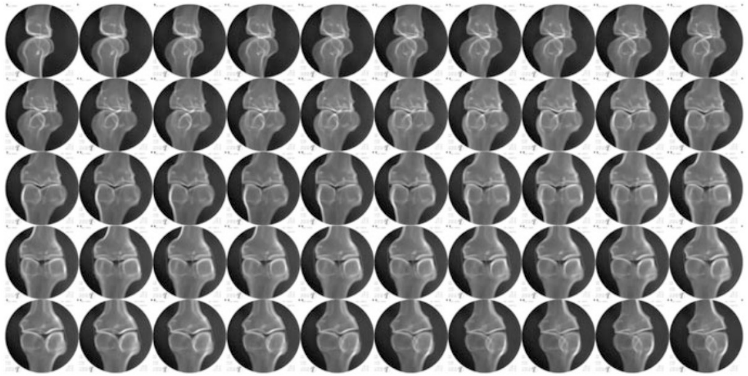
Supponiamo di prendere 2 immagini di un femore da 2 diverse posizioni fissate del C-Arm e mettiamo le 2 immagini nelle posizioni relative

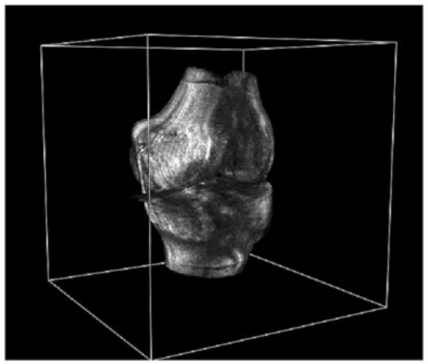
Consideriamo i 2 coni che partono dalle 2 immagini del femore.

L'intersezione genererà la sfera ricostruzione 3D del femore.

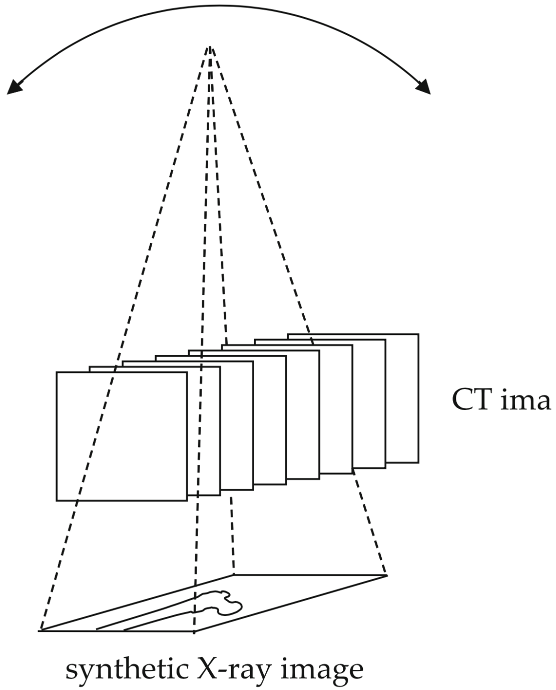
Dobbiamo conoscere le *esatte posizioni nello spazio della sorgente e del rilevatore*

Digitally Reconstructed Radiographs (DRR)

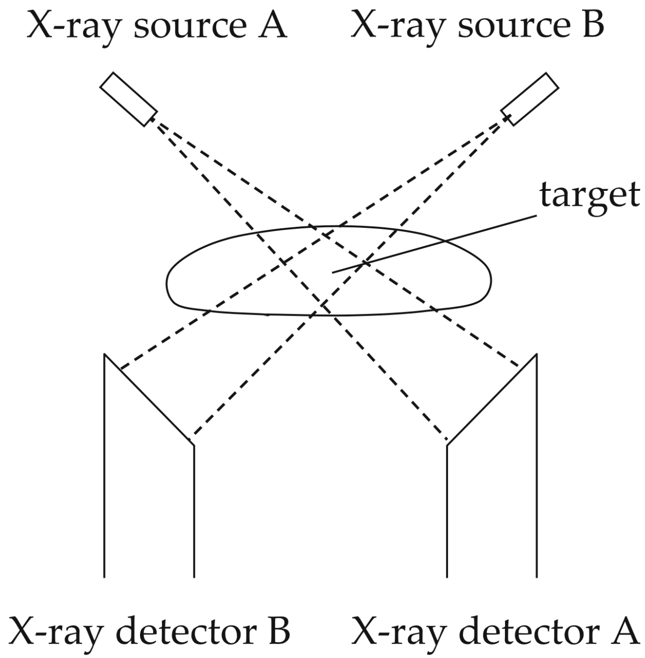




Modello 3D ricostruito dalla serie di RX 2D



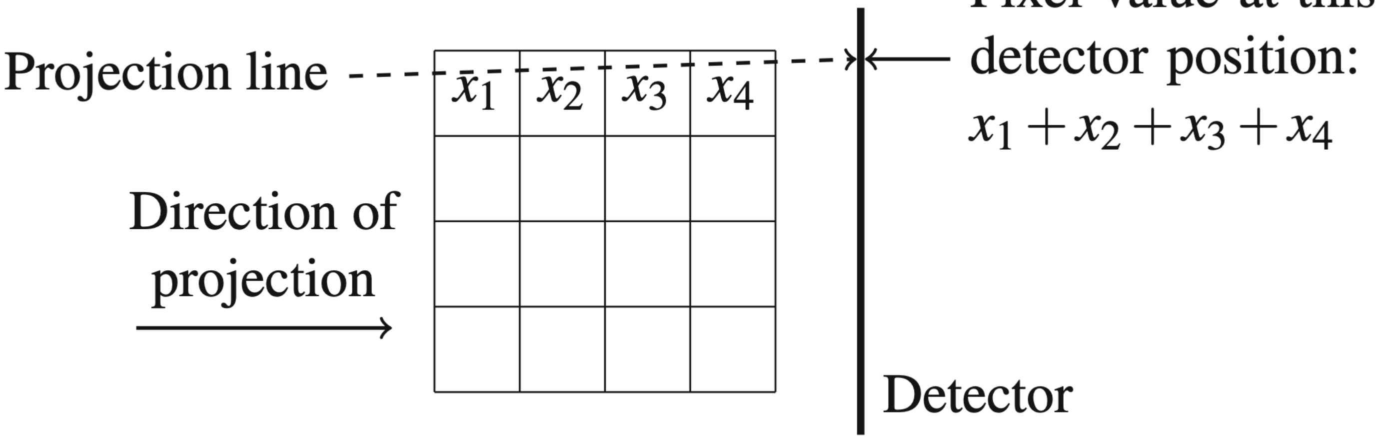
Proiezione del modello
lungo una direzione



RX stereo ottenute
durante l'intervento

Registrazione tramite sottrazione pixel a pixel:
si ottiene un errore e si considera la coppia di DDRs con l'errore minimo

TAC 3D da immagini RX



$x_1, x_2, \cdots$ rappresentano i voxel (pixel in 3D) che devono essere determinati

Sono ripresi da diversi angoli (ricostruzione della TAC per backprojection)

b_i rappresentano le attenuazioni dei raggi X in ogni pixel delle immagini RX 2D

La tecnica inversa (3D-2D) è nota anche come «Volume splatting»

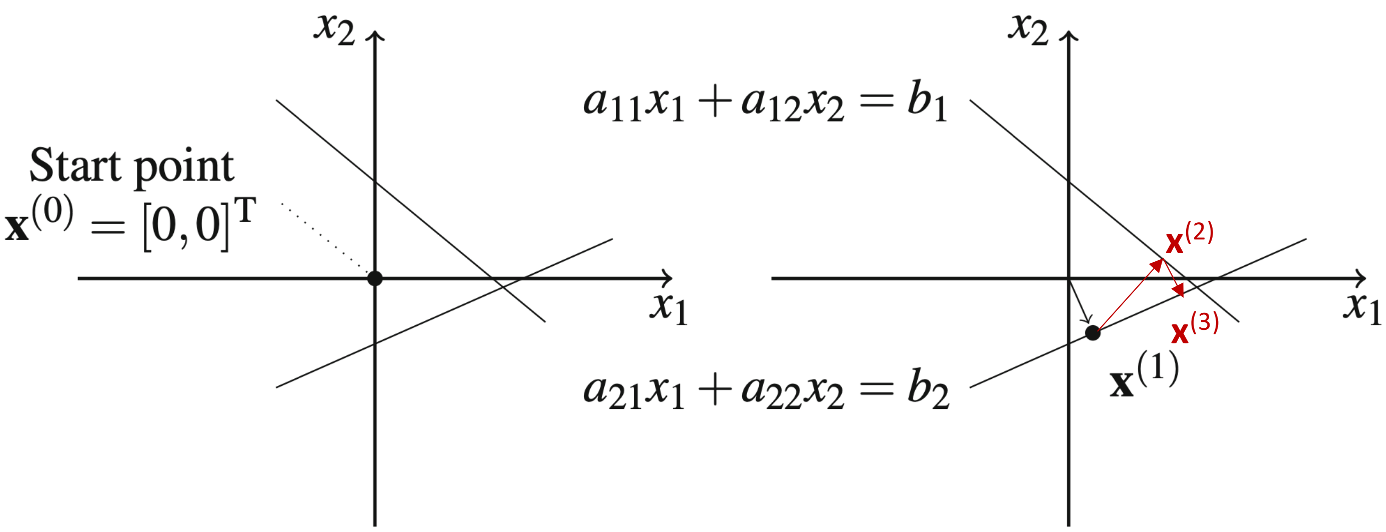
$$
\begin{gathered}
a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\
a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\
\vdots \\
a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m
\end{gathered}
$$

Dove $a_{ij} = 1$ o 0 se il voxel si trova o non si trova nella direzione di proiezione del raggio

Sistema di m equazioni a n incognite con $m \gg n$

ART - Algebraic Reconstruction Technique

Esempio: 2 equazioni in 2 incognite



$$a_{11}x_1 + a_{12}x_2 = b_1$$
$$a_{21}x_1 + a_{22}x_2 = b_2$$

Ogni equazione rappresenta una retta

Soluzione: punto di incrocio

- Si proietta il punto $\boldsymbol{x}^0 = (0 \quad 0)^T$ sulla prima retta e si ottiene $\boldsymbol{x}^1$
- Si verifica se $\boldsymbol{x}^1$ è una soluzione accettabile
- Altrimenti si proietta $\boldsymbol{x}^1$ sulla seconda retta
- Si itera finché non si ottiene la soluzione accettabile

Algoritmo ART

$$\boldsymbol{Ax} = \boldsymbol{b}$$

- Si proietta il punto $\boldsymbol{x}^0 = (0 \quad 0)^T$ sulla prima retta e si ottiene $\boldsymbol{x}^1$
- Si verifica se $\boldsymbol{x}^1$ è una soluzione accettabile
- Altrimenti si proietta $\boldsymbol{x}^1$ sulla seconda retta
- Si itera lungo le righe di $\boldsymbol{A}$ finché non si ottiene la soluzione accettabile

$$\boldsymbol{x}^0 = (0 \quad 0)^T$$

$$\boldsymbol{x}^{(n+1)} = \boldsymbol{x}^{(n)} - \frac{\boldsymbol{a}_i \boldsymbol{x}^{(n)} - \boldsymbol{b}_i}{\boldsymbol{a}_i^2} \boldsymbol{a}_i^T$$

Si ottiene imponendo l'appartenenza di $\boldsymbol{x}^{(n+1)}$ alla retta definita dalla riga *i*-esima del sistema e il prodotto scalare nullo con $\left(\boldsymbol{x}^{(n+1)} - \boldsymbol{x}^{(n)}\right)$

$$\boldsymbol{b}^{(\boldsymbol{n}+\mathbf{1})} = \boldsymbol{Ax}^{(\boldsymbol{n}+\mathbf{1})}$$

$$\varepsilon = \left\|\boldsymbol{b}^{(\boldsymbol{n}+\mathbf{1})} - \boldsymbol{b}\right\|^2$$

E' un algoritmo molto usato per trovare una soluzione per un sistema lineare sovradeterminato

Registrazione basata su punti e landmark

Supponiamo di avere 2 nuvole A e B di n punti nello spazio. Dobbiamo trovare una rotazione $\boldsymbol{R}$ e una traslazione $\boldsymbol{t}$ in modo da far coincidere A e B.

Detto $\boldsymbol{a}$ il vettore di punti A e $\boldsymbol{b}$ il vettore di punti B occorre trovare la trasformazione $\boldsymbol{Rb} + \boldsymbol{t}$ tale che $\boldsymbol{a} \equiv \boldsymbol{b}$

Supponiamo che il numero di punti sia lo stesso tra A e B e di conoscere le corrispondenze tra i punti. Allora dobbiamo minimizzare la distanza tra A e B :

$$f = \sum_{i=1}^{n} \|\boldsymbol{Rb_i} + \boldsymbol{t} - \boldsymbol{a_i}\|^2$$

Nel caso bidimensionale: $\boldsymbol{R} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$ $\boldsymbol{t} = \begin{pmatrix} t_x \\ t_y \end{pmatrix}$

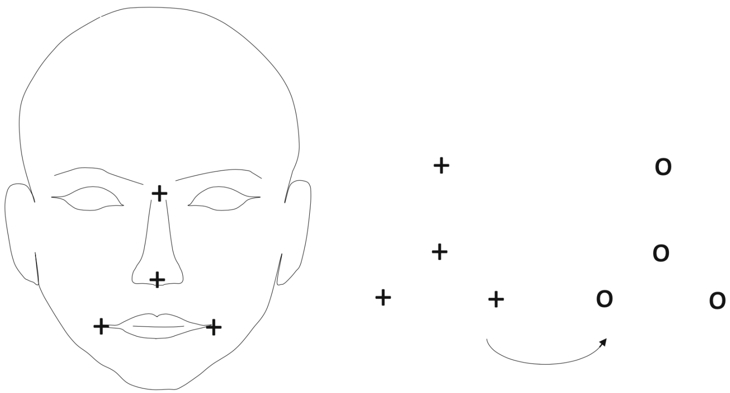
Siccome θ è piccolo si ha $\cos(\theta) = 1; \sin(\theta) = \theta$ $\boldsymbol{R} = \begin{pmatrix} 1 & -\theta \\ \theta & 1 \end{pmatrix}$

Trovare i valori di θ, t_x, t_y che minimizzano f: si deriva f rispetto θ, t_x, t_y e si pongono le derivate $= 0$
Sistema di 3 equazioni in 3 incognite

Registrazione basata su punti e landmark

Nel caso tridimensionale: $\boldsymbol{R} = \boldsymbol{D} = \begin{pmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{pmatrix}$ $\boldsymbol{t} = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}$

Matrice già impiegata nel calcolo degli angoli ai giunti mediante lo Jacobiano

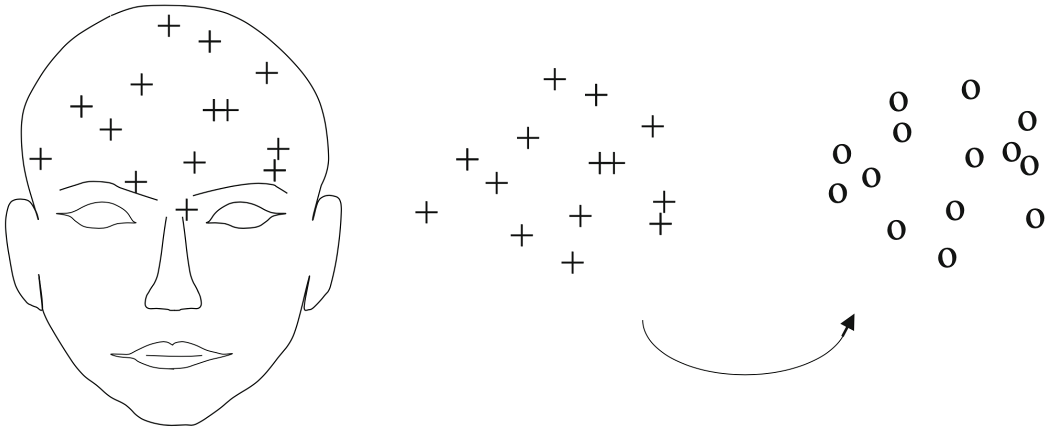


Trovare i valori di $\alpha, \beta, \gamma, t_x, t_y, t_z$ che minimizzano f:

si deriva f rispetto $\alpha, \beta, \gamma, t_x, t_y, t_z$ e si pongono le derivate $= 0$
Sistema di 6 equazioni in 6 incognite

Registrazione tra i landmark sul corpo del paziente e i corrispettivi punti su RM per il corretto posizionamento del braccio robotico

Registrazione basata su punti e landmark



Supponiamo che il numero di punti tra A e B sia diverso e le corrispondenze non siano note:

punti presi da due scansioni diverse in tempi differenti

Algoritmo ICP (Iterative Closest Point) per calcolare la corrispondenza m, la rotazione $\boldsymbol{R}$ e la traslazione $\boldsymbol{t}$

- Inizialmente $\boldsymbol{R} = \boldsymbol{I}$; $\boldsymbol{t} = 0$; corrispondenza casuale m_{new} tra A e B
- Ripeti:
  - $m_{old} = m_{new}$
  - Moving step: dato m_{new} calcolare $\boldsymbol{R}$ e $\boldsymbol{t}$
  - Matching step: calcolare m_{new} dopo aver applicato $\boldsymbol{R}$ e $\boldsymbol{t}$ a B
- Finchè $m_{new} = m_{old}$

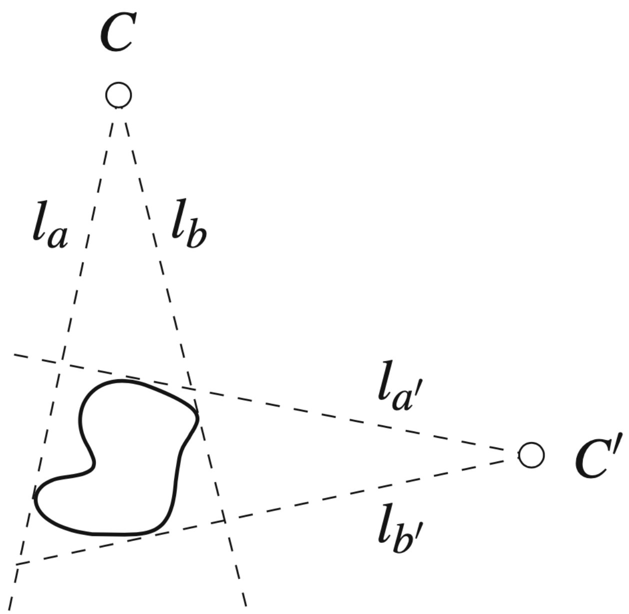
Matching step: per ogni punto $\boldsymbol{a} \in A$, prendere il punto $\boldsymbol{b} \in B$ più vicino → *Potrebbe non esserci una corrispondenza 1:1*

Registrazione basata su contorni

È un'applicazione particolare del caso precedente in cui gli insiemi di punti formano contorni chiusi

Si confrontano 2 immagini, es. RX e RM, e si allineano i rispettivi contorni → *È necessario prima segmentare le immagini*

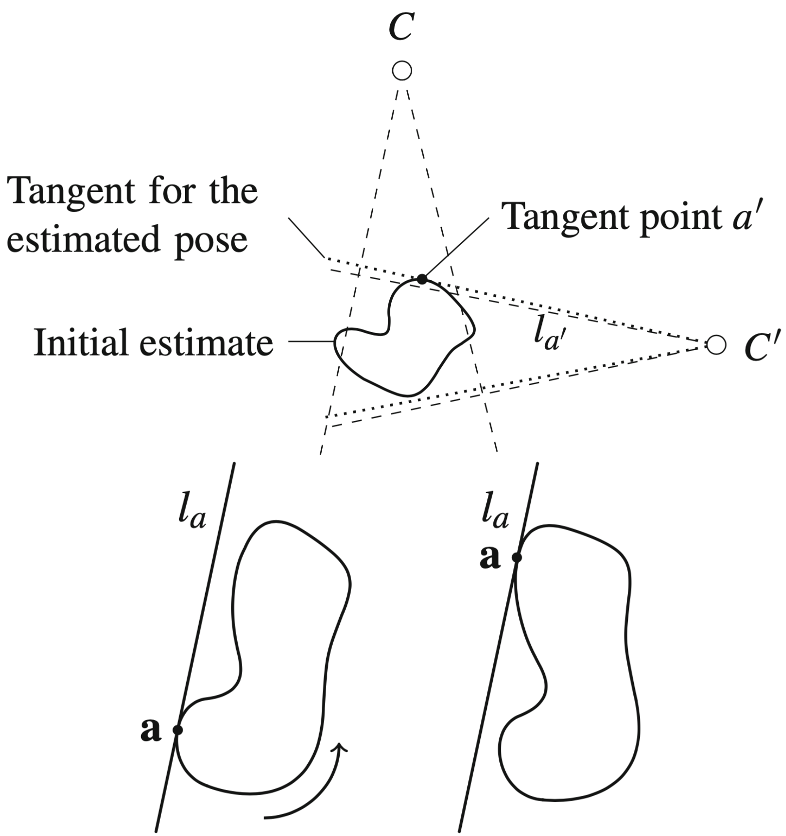
Caso 2D (non significativo per applicazioni pratiche)



Osso ripreso da 2 camere C e C' in posizioni note: otteniamo le rette tangenti $l_a, l_b, l_{a'}, l_{b'}$

Nota una stima iniziale della posizione dell'osso a partire da un volume TAC e note le posizioni delle immagini RX, possiamo stimare i punti di contorno tangenti $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{a}', \boldsymbol{b}'$

Registrazione basata su contorni



Dobbiamo raffinare la stima: trovare una rotazione $\boldsymbol{R}$ e una traslazione $\boldsymbol{t}$ che spostano la stima della posizione iniziale in modo che $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{a}', \boldsymbol{b}'$ siano i punti tangenti a $l_a, l_b, l_{a'}, l_{b'}$

Si minimizza la distanza:

$$f(\theta, t_x, t_y) = d(\boldsymbol{R}\boldsymbol{a} + \boldsymbol{t}, l_a)^2 + \cdots + d(\boldsymbol{R}\boldsymbol{b}' + \boldsymbol{t}, l_{b'})^2$$

dove $d(\boldsymbol{a}, l_a)$ è la distanza del punto $\boldsymbol{a}$ dalla retta $l_a = \boldsymbol{u} + \lambda(\boldsymbol{v} - \boldsymbol{u})$:

$$d(\boldsymbol{a}, l_a)^2 = \big(\boldsymbol{a} - \boldsymbol{u} - \lambda_0(\boldsymbol{v} - \boldsymbol{u})\big)^2$$

Problema: piccole rotazioni di un oggetto irregolare come un osso possono portare a grandi variazioni della posizione dei punti tangenti

Nel caso 3D invece dei punti tangenti avremo delle curve tangenti, che sono elaborate come insiemi di punti.

Dobbiamo trovare i parametri $\alpha, \beta, \gamma, t_x, t_y, t_z$ che minimizzano f

Registrazione basata su intensità

La registrazione diretta pixel a pixel non necessita della conoscenza di punti fiduciali e/o di contorni, ma in genere va effettuata da modalità diagnostiche diverse che hanno differenti dinamiche

CT

1	1	3	3
1	1	3	3
2	2	3	3
2	2	3	3

MR

3	3	3	3
3	3	3	3
1	1	2	2
1	1	2	2

Analisi dell'intensità dei pixel da 2 sorgenti diverse:
Rotazione di RM di 90°

CT

1	1	3	3
1	1	3	3
2	2	3	3
2	2	3	3

MR

2	2	2	2
2	2	2	2
3	3	1	1
3	3	1	1

In questo caso, è necessaria una corrispondenza tra i valori dei pixel:

CT	MR
1	3
2	1
3	2

Mutua informazione

La mutua informazione è una misura di quanto i valori di una variabile statistica A possano essere predetti osservando i valori di un'altra variabile statistica B

Si definisce probabilità congiunta $p_{AB}(a, b)$ la probabilità che la variabile A assuma il valore a e che contemporaneamente la variabile B assuma il valore b

La mutua informazione si definisce come:

$$I(A, B) = \sum_{a,b} p_{AB}(a, b) \log \frac{p_{AB}(a, b)}{p_A(a) p_B(b)}$$

Se A e B sono statisticamente indipendenti allora $p_{AB}(a, b) = p_A(a) p_B(b)$ e $I(A, B) = 0$

Immagini uguali

A

0	1
1	0

B

0	1
1	0

$$p_A(0) = 0.5$$
$$p_A(1) = 0.5$$
$$p_B(0) = 0.5$$
$$p_B(1) = 0.5$$

$$p_{AB}(0,0) = 0.5$$
$$p_{AB}(1,1) = 0.5$$
$$p_{AB}(0,1) = 0$$
$$p_{AB}(1,0) = 0$$

La probabilità che un pixel di A sia 0 e il corrispondente pixel di B sia 0 è pari a 0.5

La probabilità che un pixel di A sia 0 e il corrispondente pixel di B sia 1 è pari a 0

$$I(A,B) = 1$$

Immagini opposte

A

0	1
1	0

B

1	0
0	1

$$I(A,B) = 1$$

La perfetta inversione dei livelli di grigio di A *indica comunque un elevato grado di ordine* nella relazione rispetto a B e quindi una elevata predicibilità di una rispetto all'altra.

Immagini simili

A

0	1
1	0

B

0	1
1	1

$$I(A,B) = 0.32$$

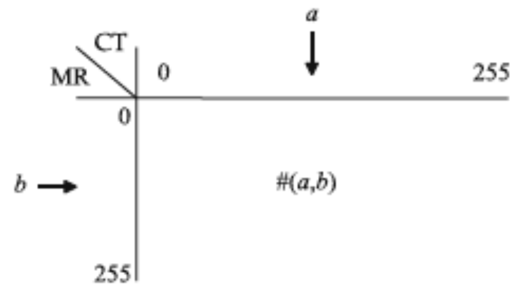
Allineare due immagini significa renderle massimamente simili e quindi cercare di *massimizzare* $I(A,B)$

Registrazione basata su mutua informazione

Supponiamo che l'immagine A (es. TAC) e l'immagine B (es. RM) siano a 256 livelli di grigio

Approssimazione di p_A e p_B: si costruiscono gli istogrammi per A e per B: $\begin{array}{c} 0 \\ 1 \\ \vdots \\ 255 \end{array} \begin{bmatrix} n.\,pixel\ 0 \\ n.\,pixel\ 1 \\ \vdots \\ n.\,pixel\ 255 \end{bmatrix}$ e si normalizza

Approssimazione di p_{AB}: istogramma congiunto, matrice quadrata

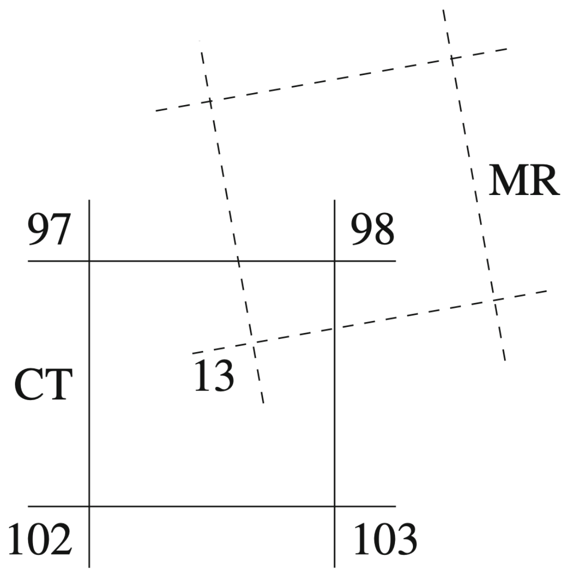


Si sovrappongono le 2 immagini. Per ogni coppia di pixel con livelli di grigio (a, b) si incrementa di 1 l'elemento (a, b) dell'istogramma congiunto. Alla fine si normalizza.

Registrazione basata su mutua informazione

Si muove ad esempio l'immagine B (floating) rispetto l'immagine A (reference) per massimizzare $I(A, B)$

Registrazione: trovare i parametri di rotazione e traslazione $\alpha, \beta, \gamma, t_x, t_y, t_z$ di B che massimizzano $I(A, B)$

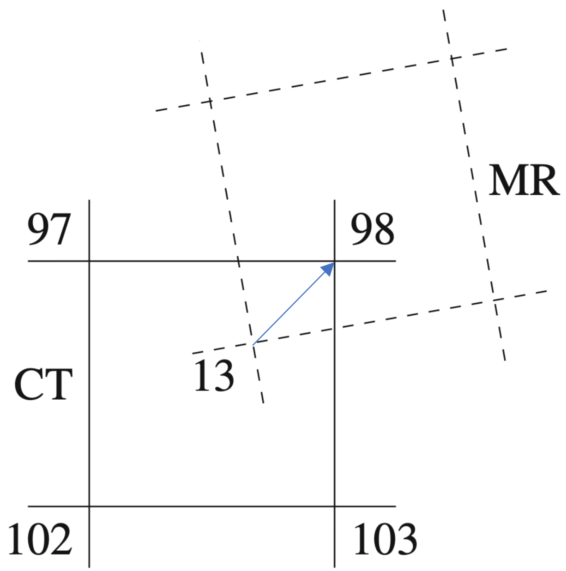


Necessità di effettuare interpolazioni quando le immagini non sono perfettamente allineate

Registrazione basata su mutua informazione

Si muove ad esempio l'immagine B (floating) rispetto l'immagine A (reference) per massimizzare $I(A, B)$

Registrazione: trovare i parametri di rotazione e traslazione $\alpha, \beta, \gamma, t_x, t_y, t_z$ di B che massimizzano $I(A, B)$

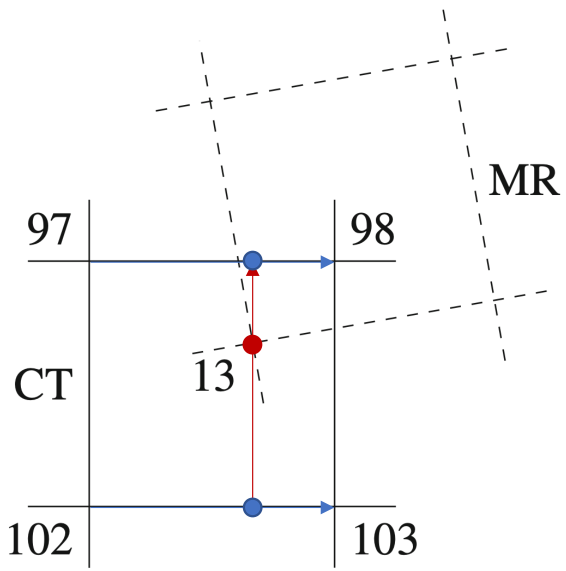


Nearest Neighbor: la coppia (a, b) è data dai voxel più vicini

Registrazione basata su mutua informazione

Si muove ad esempio l'immagine B (floating) rispetto l'immagine A (reference) per massimizzare $I(A, B)$

Registrazione: trovare i parametri di rotazione e traslazione $\alpha, \beta, \gamma, t_x, t_y, t_z$ di B che massimizzano $I(A, B)$

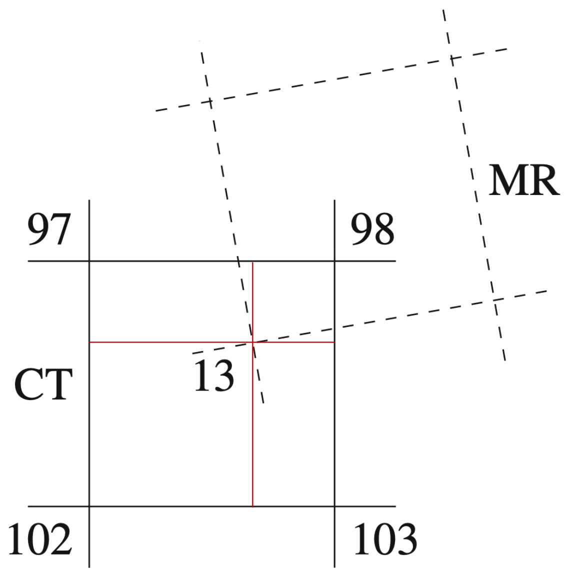


Interpolazione Trilineare: la coppia (x, b) è ottenuta interpolando linearmente i valori degli otto vicini di b separatamente lungo le tre dimensioni del voxel

Registrazione basata su mutua informazione

Si muove ad esempio l'immagine B (floating) rispetto l'immagine A (reference) per massimizzare $I(A,B)$

Registrazione: trovare i parametri di rotazione e traslazione $\alpha, \beta, \gamma, t_x, t_y, t_z$ di B che massimizzano $I(A,B)$



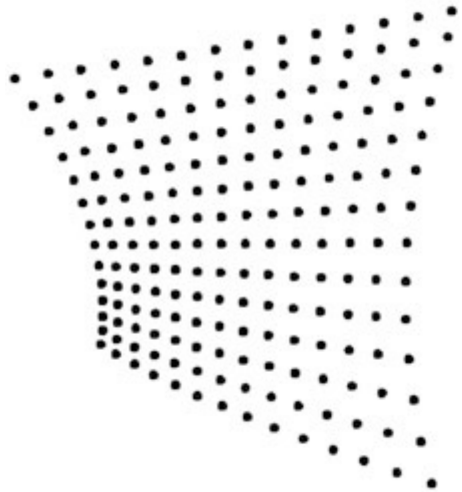
Interpolazione ai volumi parziali: la coppia (x, b) è ottenuta pesando i valori degli otto vicini di b ciascuno con un peso che è l'inverso del volume del parallelepipedo ottenuto in cui b e il singolo vicino formano la diagonale principale

Deformazione dell'immagine

La registrazione vista finora è rigida, ovvero basata su una trasformazione affine.

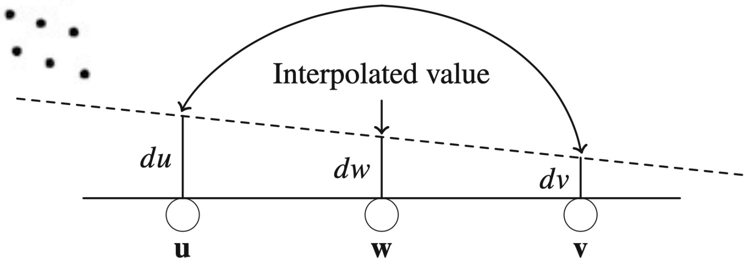
L'acquisizione delle immagini mediche può generare distorsioni.

Le patologie investigate possono generare una deformazione tra le strutture che devono essere allineate per cui è necessaria una registrazione di tipo elastico.



Supponiamo di avere una griglia G e una griglia deformata G'

Inizialmente si ha un allineamento approssimato tramite landmark e poi si deforma G:



Nota la corrispondenza tra i punti, si vede che $\boldsymbol{u}$ deve essere spostato di du e $\boldsymbol{v}$ di dv.

Interpolazione **bilineare**: il generico punto intermedio $\boldsymbol{w}$ sarà spostato di dw ottenuto come interpolazione lineare separata tra gli scostamenti di $\boldsymbol{u}$ e $\boldsymbol{v}$ separatamente lungo *x* e *y*.

Interpolazione per spline cubiche

L'interpolazione bilineare produce una funzione di deformazione che non varia dolcemente punto a punto. Si preferisce descrivere le funzioni di scostamento con delle leggi cubiche i cui parametri si ricavano con il cosiddetto metodo Gaussian Least Squares a partire da un insieme di landmark.

$a_0 + a_1x + a_2y + a_3xy + a_4x^2 + a_5y^2 + a_6x^2y + a_7xy^2 + a_8x^3 + a_9y^3$ Spostamento lungo x

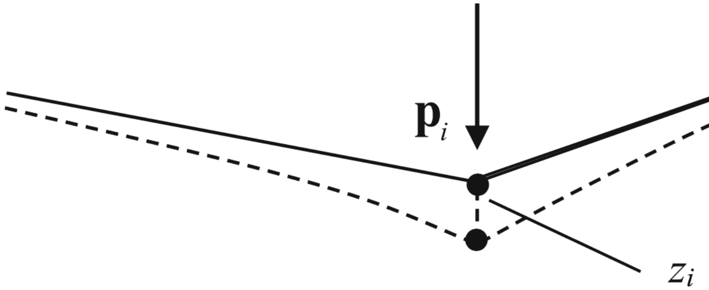
$b_0 + b_1x + b_2y + b_3xy + b_4x^2 + b_5y^2 + b_6x^2y + b_7xy^2 + b_8x^3 + b_9y^3$ Spostamento lungo y

Dato il punto (x_m, y_m) della griglia G non deformata e il punto (x_d, y_d) della griglia deformata G' dobbiamo trovare i parametri $a_0, a_1, \cdots, a_9, b_0, b_1, \cdots, b_9$ che minimizzano:

$$SSD = \sum_i \left\| \left(x_{m,i}, y_{m,i}\right)^T - \left(a_0 + a_1x_{d,i} + \cdots + a_9y_{d,i}^3, b_0 + b_1x_{d,i} + \cdots + b_9y_{d,i}^3\right)^T \right\|^2$$

La minimizzazione si ottiene ponendo $\frac{\partial SSD}{\partial a_0} = 0, \frac{\partial SSD}{\partial a_1} = 0, \ldots, \frac{\partial SSD}{\partial b_9} = 0$ e si ottengono 20 equazioni in 20 incognite

Thin-plate spline



L'interpolazione cubica fornisce una legge di deformazione che varia dolcemente, ma non tiene conto delle possibili cause fisiche della deformazione.

Nella thin-plate spline interpolation si ipotizza di deformare una superficie piana sottile mediante una forza verso il basso: il punto di controllo noto $\boldsymbol{p}_i$ si sposta di z_i per effetto della forza

La superficie deformata sarà descritta da:

$$f(\boldsymbol{p}) = \sum_{i=1}^{N} w_i F(\|\boldsymbol{p} - \boldsymbol{p}_i\|)$$

dove i $\boldsymbol{p}_i$ sono i punti di controllo $(x_i, y_i)^T$ della griglia deformata e $F(\cdot)$ è una funzione kernel che descrive la deformazione puntuale e che tipicamente si sceglie $F(r) = r^2$ (l'influenza di $\boldsymbol{p}_i$ decresce all'aumentare della distanza)

I parametri da trovare sono le w_i: sistema da N equazioni in N incognite (punti della griglia)

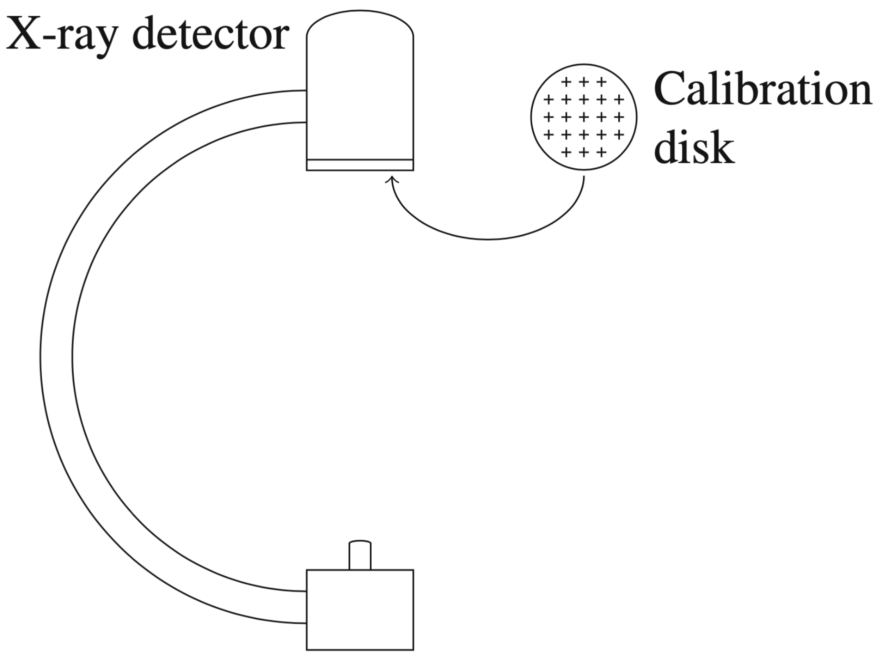
Estensione per considerare anche lo scostamento traslazionale (parametri a_0, a_1, a_2):

$$f(\boldsymbol{p}) = \sum_{i=1}^{N} w_i F(\|\boldsymbol{p} - \boldsymbol{p}_i\|) + a_0 + a_1 x + a_2 y$$

$$\boxed{\sum_{i=1}^{N} w_i = 0; \sum_{i=1}^{N} w_i x_i = 0; \sum_{i=1}^{N} w_i y_i = 0;}$$

Vincoli per gestire il sistema di equazioni che è indeterminato

Esempio



Il campo magnetico terrestre influenza la traiettoria dei raggi X inducendo una deformazione dell'immagine

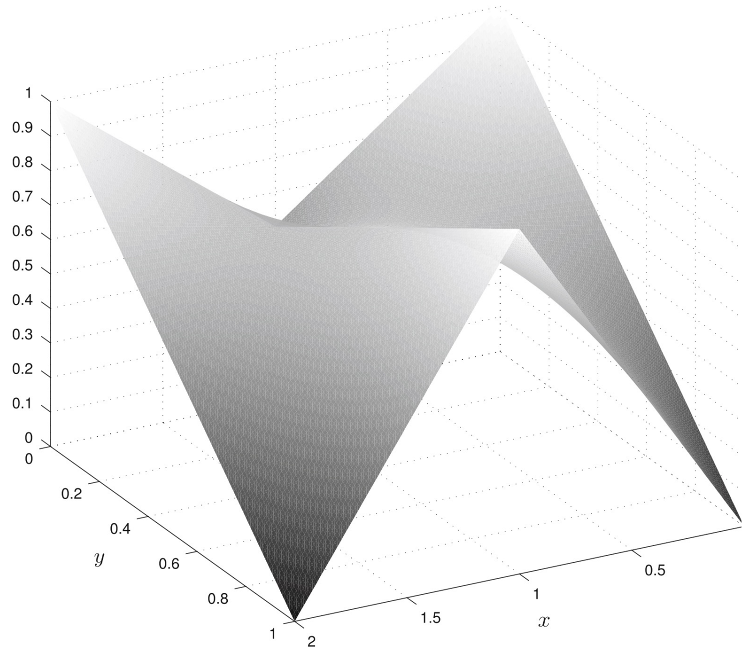
Il disco di calibrazione fornisce G e la proiezione di G sull'immagine fornisce G'

N.B. la legge di deformazione cambia quindi con la rotazione del braccio a C!!

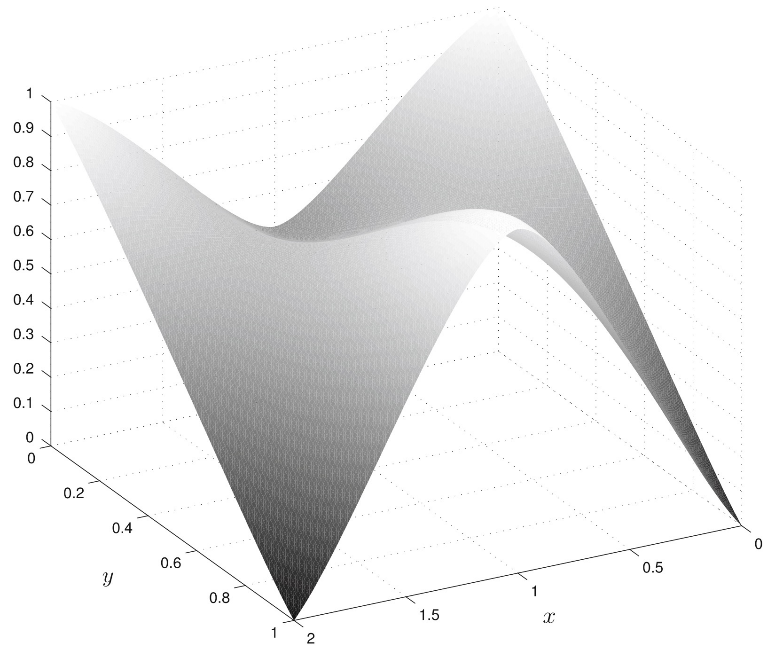
Trovare la funzione che interpola G' per riottenere G

Esempio

Punti	Scostamenti
(0,0)	1
(1,1)	1
(1,0)	0.5
(0,1)	0
(2,1)	0
(0,2)	1



Interpolazione bilineare



Thin-plate spline

Calibrazione «occhio-mano»

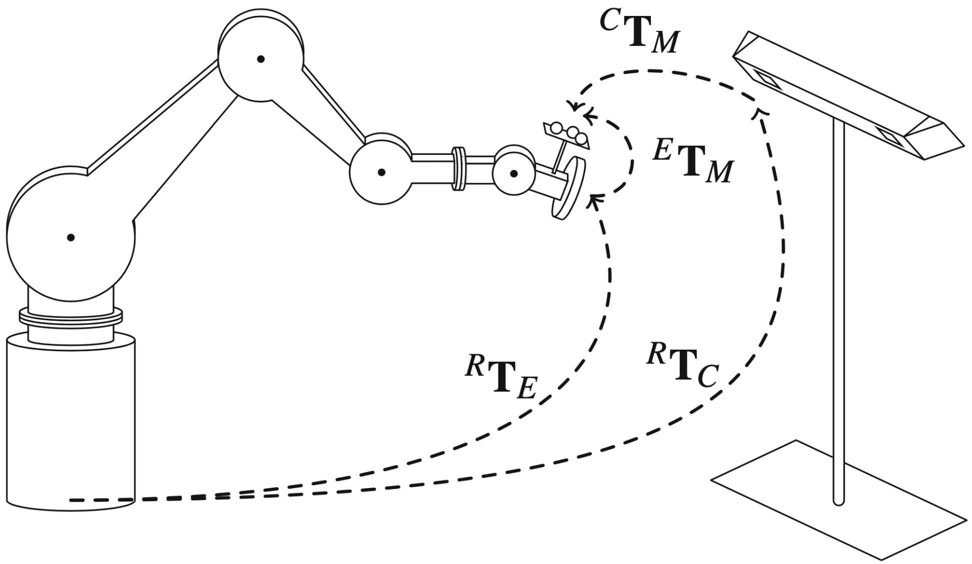
Si consideri la seguente operazione per guidare un trapano osseo:

- Il chirurgo individua il punto dell’osso da trapanare con il puntatore del sistema di tracciamento
- Si calcolano gli angoli ai giunti del braccio
- Il braccio si muove
- Si effettua l’operazione con trapano

Il robot calcola la sua posizione e orientazione rispetto al proprio sistema di coordinate, ma necessitiamo di conoscere questa posizione nel sistema della coppia puntatore/telecamera:

Problema di calibrazione

Calibrazione «occhio-mano»



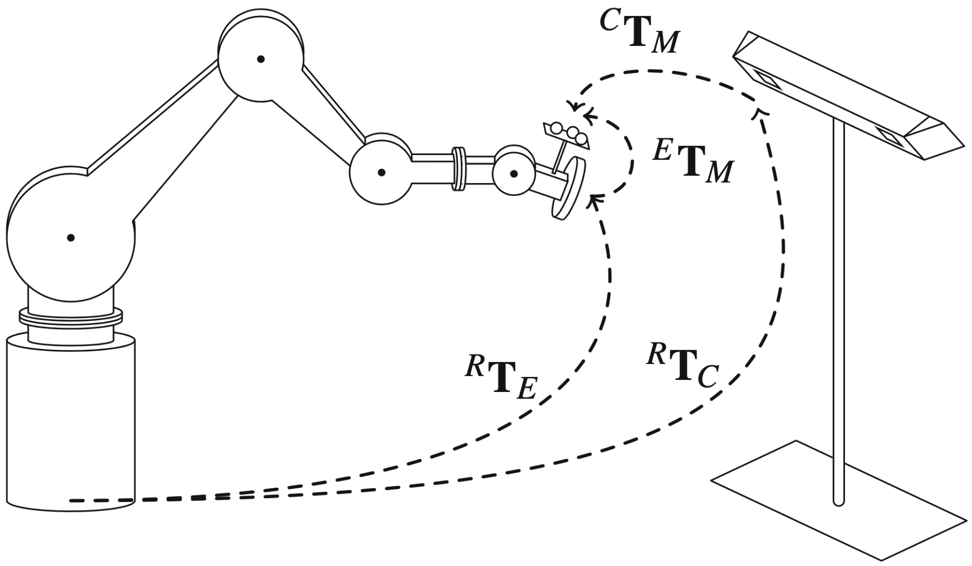
Marker M solidale all'effettore E
C è la camera e R è la base del robot

Matrici in gioco:
$\boldsymbol{T}_E^R$ Dalla base del robot all'effettore
$\boldsymbol{T}_M^E$ Dall'effettore al marker
$\boldsymbol{T}_C^R$ Dalla base del robot alla camera
$\boldsymbol{T}_M^C$ Dalla camera al marker

$\boldsymbol{T}_E^R$ è nota dalla cinematica del robot
$\boldsymbol{T}_M^C$ è nota dal sistema di tracciamento

Le altre due matrici non sono note

Calibrazione «occhio-mano»



La posizione del marker M è la stessa sia dal punto di vista del robot che dal punto di vista della camera:

$$\boxed{\boldsymbol{T}_E^R \cdot \boldsymbol{T}_M^E = \boldsymbol{T}_C^R \cdot \boldsymbol{T}_M^C}$$

Equazione matriciale del tipo $\boldsymbol{AX} = \boldsymbol{YB}$ con incognite $\boldsymbol{X}, \boldsymbol{Y}$

Muoviamo il robot in n posizioni diverse:

$$\boldsymbol{A_i X} = \boldsymbol{Y B_i}, \qquad i = 1, \cdots n$$

possiamo scrivere: $\boldsymbol{A_i X B_i^{-1}} = \boldsymbol{Y}$

Le incognite $\boldsymbol{X}, \boldsymbol{Y}$ sono matrici di rototraslazione a 12 parametri incogniti.

Per $n > 2$ abbiamo un sistema di $12n$ equazioni a 24 incognite.

Sistema sovradeterminato*: si risolve con ART*