



UNIVERSITÀ DEGLI STUDI DI ROMA
TOR VERGATA

Progetto di Robotica Industriale

Realizzazione di una mano robotica virtuale

Autori:

Marco CAGGIANO
Antonio DE STASIO
Mattia ADDUCCHIO

Titolare del corso:

Prof. Antonio TORNAMBÈ

A.A. 2011/2012

Indice

1	INTRODUZIONE	1
2	DESCRIZIONE DELLA MANO	2
3	DESCRIZIONE DELLE FUNZIONI	3
3.1	Movimentazione della mano	3
3.2	Cinematica diretta	3
3.3	Cinematica inversa	3
3.3.1	Algoritmo della cinematica inversa	3
3.3.2	Cinematica inversa del dito indice	4
3.4	Inseguimento traiettoria	5
3.5	Funzione numeri	5
3.6	Modalità tastiera	5
3.7	Gestione della visuale	5
3.8	Movimentazione della luce	5
3.9	Interfaccia grafica e altre funzioni	5
3.10	Menu help	5
4	STRUTTURA DEL CODICE	6

1 INTRODUZIONE

Il progetto della mano è stato sviluppato con l'idea di poter raggiungere un buon livello realistico sia dal punto di vista estetico che dal punto di vista dei movimenti.

Per raggiungere il secondo obiettivo sono stati implementati diversi vincoli sugli angoli di ciascun dito, facendo particolarmente attenzione al movimento del pollice, che rappresenta una caratteristica fondamentale di una mano reale.

In seguito, sono stati sviluppati due scenari diversi: nel primo viene applicata la cinematica diretta, permettendo alla mano di contare i numeri da zero a cinque; nella seconda viene applicata la cinematica inversa, premendo i tasti su una tastiera virtuale.

2 DESCRIZIONE DELLA MANO

La mano è stata realizzata con una base di dimensioni 80 x 80 x 10 che rappresenta il palmo della mano.

Le cinque dita hanno le seguenti dimensioni:

- **Mignolo**

- Prima falange: 30 mm
- Seconda falange: 20 mm
- Terza falange: 15 mm

- **Anulare**

- Prima falange: 35 mm
- Seconda falange: 30 mm
- Terza falange: 15 mm

- **Medio**

- Prima falange: 40 mm
- Seconda falange: 30 mm
- Terza falange: 15 mm

- **Indice**

- Prima falange: 35 mm
- Seconda falange: 35 mm
- Terza falange: 23 mm

- **Pollice**

- Prima falange: 5 mm
- Seconda falange: 30 mm
- Terza falange: 20 mm

Ogni dito può fare 4 movimenti: tre rotazioni intorno all'asse y che rappresentano il piegarsi del dito verso il palmo della mano e una rotazione intorno all'asse x che rappresenta il movimento laterale, se possibile.

Il pollice presenta una pre-rotazione di $\frac{\pi}{2}$ intorno all'asse X che permette di renderlo "opponibile" rispetto alle altre dita.

3 DESCRIZIONE DELLE FUNZIONI

3.1 Movimentazione della mano

Per selezionare il dito che si desidera muovere è sufficiente premere i tasti p , i , M , a , m che corrispondono rispettivamente a pollice, indice, medio, anulare e mignolo.

Con i tasti 1-2-3 si seleziona la prima-seconda-terza falange del dito.

Quindi è possibile muovere la falange selezionata usando i tasti direzionali, tramite i quali non si fa altro che incrementare/decrementare l'angolo corrispondente.

3.2 Cinematica diretta

La cinematica diretta è stata fatta considerando ogni dito come un struttura a se stante costituita da 4 giunti rotoidali. Il primo giunto rappresenta una rotazione intorno all'asse z e corrisponde a un piccolo movimento laterale del dito. Gli altri tre corrispondono a rotazioni intorno all'asse y e rappresentano i movimenti che permettono di piegare il dito. Nel caso del pollice è stata considerata un'ulteriore rotazione di 90 gradi rispetto all'asse x per rendere il pollice *opponibile* alle altre dita.

Per sapere la posizione del dito rispetto al sistema di riferimento inerziale si considera una traslazione di base che colloca la mano nello spazio e una seconda traslazione che permette di esprimere la posizione del dito rispetto al centro della mano.

Tramite il tasto q è possibile inserire manualmente il valore degli angoli e successivamente con il tasto c si calcola la posizione della punta del dito in coordinate assolute.

Per evitare che le dita raggiungano posizioni non realistiche sono stati inseriti dei limiti sugli angoli.

Inoltre vengono effettuati vari controlli per evitare che il pollice entri nel palmo della mano.

3.3 Cinematica inversa

Premendo il tasto x si inseriscono le coordinate desiderate per la punta del dito e tramite il tasto C si attiva il calcolo della cinematica inversa affinché la punta del dito raggiunga la posizione desiderata, ove possibile, altrimenti proverà ad avvicinarsi a essa.

3.3.1 Algoritmo della cinematica inversa

Partendo dal vettore di posizione del punto terminale di ogni dito, ricavato dalla cinematica diretta, è stata implementata la cinematica inversa, utilizzando il metodo di risoluzione numerica del gradiente.

Utilizzare il gradiente è stata una scelta obbligata visto che, essendo il Jacobiano (J) non quadrato, non era possibile implementare il metodo di Newton che prevede il calcolo dell'inversa di J .

Essendo la cinematica diretta diversa, a seconda della posizione della mano e a seconda se si sta agendo sul dito pollice o no, la cinematica inversa implementata è più di una e vengono usate in accordo con gli eventi sopra citati.

Nella relazione verrà mostrato il caso specifico di cinematica inversa del dito indice, nel caso la mano si trovi in posizione di *home*.

3.3.2 Cinematica inversa del dito indice

Dalla cinematica diretta si è calcolato il vettore di posizione del punto terminale dell'indice, come mostrato nelle (3.1).

$$x(\hat{q}) = \cos(\hat{q}_0)(L_3 \cos(\hat{q}_3 + \hat{q}_2 + \hat{q}_1) + L_2 \cos(\hat{q}_2 + \hat{q}_1) + L_1 \cos(\hat{q}_1)) \quad (3.1a)$$

$$y(\hat{q}) = \sin(\hat{q}_0)(L_3 \cos(\hat{q}_3 + \hat{q}_2 + \hat{q}_1) + L_2 \cos(\hat{q}_2 + \hat{q}_1) + L_1 \cos(\hat{q}_1)) \quad (3.1b)$$

$$z(\hat{q}) = -(L_3 \sin(\hat{q}_3 + \hat{q}_2 + \hat{q}_1) + L_2 \sin(\hat{q}_2 + \hat{q}_1) + L_1 \sin(\hat{q}_1)) \quad (3.1c)$$

ottenendo così il vettore delle stime $h(\hat{q})$ (3.2).

$$h(\hat{q}) = [x(\hat{q}) \quad y(\hat{q}) \quad z(\hat{q})] \quad (3.2)$$

Attraverso il vettore delle stime di posizione è possibile ricavare l'errore tra la posizione attuale del dito e la posizione desiderata,

$$e_x = X_{des} - x(\hat{q}); \quad (3.3a)$$

$$e_y = Y_{des} - y(\hat{q}); \quad (3.3b)$$

$$e_z = Z_{des} - z(\hat{q}); \quad (3.3c)$$

$$e = [e_x \quad e_y \quad e_z] \quad (3.4)$$

dove le X_{des} , Y_{des} e Z_{des} sono le coordinate del punto desiderato ed e il vettore errore di posizione.

Successivamente dal vettore $h(\hat{q})$ (3.2) si è calcolato lo Jacobiano, come mostrato nella (3.5)

$$J(\hat{q}) = \frac{\delta h}{\delta q} \quad (3.5)$$

ottenendo così l'equazione (3.6) che permette di calcolare le variabili di giunto da assegnare nell'istante successivo.

$$\hat{q}(t + T) = \hat{q}(t) + \frac{\lambda T}{2} (J^t(\hat{q}) \cdot e) \quad (3.6)$$

3.4 Inseguimento traiettoria

E' stata implementata una traiettoria in modo da mostrare il funzionamento della cinematica inversa per l'inseguimento di traiettorie. Attivando questa funzione con il tasto *C* una pallina rossa tratterà una traiettoria e il dito proverà a seguirla.

3.5 Funzione numeri

Con il tasto *n* si attiva la funzione numeri. In questa modalità premendo uno dei numeri da 0 a 5 la mano mostrerà il numero scelto.

3.6 Modalità tastiera

Premendo il tasto *k* viene mostrata una tastiera e la mano viene ribaltata in modo che sia possibile farla muovere sulla tastiera.

Alla pressione di un qualsiasi tasto la mano si posizionerà sul tasto corrispondente della tastiera virtuale e lo premerà.

Inoltre con i tasti 1-2-3 è possibile passare da uno scenario all'altro in modo da avere più visuali della tastiera e della mano.

Per uscire dalla modalità tastiera è sufficiente premere il tasto *BACKSPACE*

3.7 Gestione della visuale

La gestione della visuale è stata fatta utilizzando il pacchetto *Proscene*.

Grazie alla funzionalità di questo pacchetto è possibile ruotare la telecamera con il click del mouse, fare lo zoom con lo scroll e spostare la telecamera con i tasti direzionali.

Per evitare conflitti tra i comandi base della mano e i comandi già implementati nel Proscene il tasto *ENTER* permette di passare dall'una all'altra modalità.

In qualsiasi momento il tasto *v* permette di tornare alla visuale predefinita.

3.8 Movimentazione della luce

La luce della scena può essere controllata con i tasti direzionali. Inoltre con i tasti *x-y-z* è possibile indicare intorno a quale asse deve ruotare la luce. Questa funzione è attivabile con il tasto *l*.

3.9 Interfaccia grafica e altre funzioni

I messaggi per l'utente vengono scritti in sovraimpressione nella parte inferiore dello schermo. E' sempre possibile cancellare questi messaggi con il tasto *CANC*.

Inoltre con il tasto *t* si attiva la modalità testo e quindi sarà possibile scrivere qualsiasi messaggio nella parte superiore dello schermo.

3.10 Menu help

Infine il tasto *h* attiva il menu help che descrive tutti i comandi necessari per le funzioni finora descritte.

4 STRUTTURA DEL CODICE

Il codice è scritto in linguaggio JAVA ed è stato strutturato in classi in modo da sfruttare i vantaggi della programmazione ad oggetti.

La mano è costituita da un insieme di oggetti *dito* ognuno con i suoi attributi ma tutti con gli stessi metodi.

Tra gli attributi di ogni dito ci sono i valori degli angoli di ogni falange, i limiti minimi e massimi degli angoli, le dimensioni delle varie parti del dito, la posizione del dito rispetto al centro della mano, la posizione della punta del dito e il colore.

Tra i metodi della classe invece ci sono, oltre al costruttore, funzioni che permettono di ruotare le falangi del dito, impostare gli angoli (in gradi o in radianti), impostare le coordinate per la punta del dito, calcolare la cinematica diretta e infine disegnarlo.