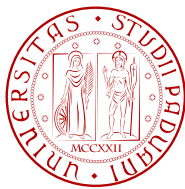


Giulio Marin

Modelli deformabili della mano a partire da dati acquisiti tramite laser scanner



Tesi di laurea triennale

Relatore: Prof. Pietro Zanuttigh
Co-relatore: Ing. Carlo Dal Mutto



Università degli Studi di Padova
Facoltà di Ingegneria
Dipartimento di Ingegneria dell'Informazione
Luglio 2011



Sommario

Questa tesina mostra come sia possibile realizzare il modello tridimensionale di una mano reale utilizzando uno scanner 3D economico. Lo scopo di questo lavoro è di creare un database contenente le mani di differenti persone a cui è associato uno scheletro. Attraverso l'uso di software specializzato è stato creato uno scheletro per ogni modello. Tali modelli verranno poi utilizzati per lo sviluppo di una tecnica di riconoscimento della mano da una camera 3D. Sono stati realizzati dei video-tutorial reperibili online che descrivono i procedimenti seguiti.

Il **Capitolo 1** contiene una breve descrizione del lavoro svolto. Nel **Capitolo 2**, dopo una breve presentazione dell'anatomia della mano, vengono elencate diverse metodologie per la creazione di un modello di mano al computer. Il **Capitolo 3** raccoglie una serie di informazioni sullo scheletro e relativi vincoli. Nel **Capitolo 4** vengono spiegati brevemente i concetti base per l'animazione di un modello, cinematica e Linear Blend Skinning. Nel **Capitolo 5** viene descritto il macchinario realizzato per la scansione delle mani. Nel **Capitolo 6** vengono presentate in dettaglio tutte le fasi di costruzione e animazione del modello.

Abstract

This work shows how it is possible to create a three-dimensional model of a real hand using a 3D scanner. Through the use of specialized software a skeleton for each model was also built. The aim of this work is to create a database containing the hands of different people. These models are then used to develop a technique for hand recognition by a 3D camera. Different video-tutorials describing each step, were created and are now available online.

Chapter 1 provides a brief description of the work. In **Chapter 2**, after a brief presentation of the hand's anatomy, several methodologies for the creation of a hand model using a computer are listed. **Chapter 3** contains some information about the skeleton and its constraints. In **Chapter 4** the basic concepts for the animation of a model are explained. In **Chapter 5** the structure that has been built for hand scanning is described. In **Chapter 6** all the steps necessary for the construction and the animation of the model are treated in detail.

Indice

1	Introduzione	1
2	La Mano	2
2.1	Anatomia	2
2.2	Modello	3
3	Lo Scheletro	4
3.1	Gradi di libertà e vincoli	5
4	Animazione	6
4.1	Cinematica diretta e inversa	6
4.2	Linear Blend Skinning	7
5	Hand Scanner	8
5.1	Premessa	8
5.2	Struttura	9
6	Procedimento seguito	12
6.1	Acquisizione della mano	13
6.2	Elaborazione immagini	15
6.3	Realizzazione dello scheletro	16
6.4	Animazione del modello	17
7	Conclusioni	18
A	Codice	19
	Bibliografia	24

Elenco delle figure

2.1	Scheletro di una mano [14].	2
3.1	Sistema di coordinate locale e posizione dello scheletro	4
4.1	Esempio di catena cinematica	6
5.1	Posizione della mano nelle diverse strutture	9
5.2	Componenti utilizzati	9
5.3	Alcune foto del macchinario	10
5.4	Braccio per il sostegno del laser	11
5.5	Supporto per la mano	11
6.1	Strumenti utilizzati	12
6.2	Marker applicati alla mano	13
6.3	Posizioni angolari delle varie scansioni rispetto alla verticale.	13
6.4	Mesh ottenuta da una scansione	14
6.5	Mesh ottenuta dopo l'allineamento e la fusione	15
6.6	Mesh ottenuta dopo aver applicato l'algoritmo di Poisson	15
6.7	Confronto tra mesh originale e mesh ridotta	15
6.8	Scheletro di un modello	16
6.9	Particolari della fase di animazione	17

Capitolo 1

Introduzione

Lo scopo di questa tesi è quello di fornire la base per lo sviluppo del tracciamento e riconoscimento del movimento di una mano tramite una camera 3D. Esistono sostanzialmente due modalità di tracciamento: utilizzando dispositivi meccanici da applicare alla mano, oppure utilizzando dispositivi ottici esterni, ad esempio una videocamera 3D. I dispositivi meccanici sono veloci e precisi, ma ingombranti e spesso costosi. Rientra in questa categoria anche la soluzione [20] di Wang e Popović che prevede l'utilizzo di un guanto colorato per il riconoscimento della posa della mano. Il tracciamento visivo, invece, risulta meno invasivo e non necessita di applicare alcun dispositivo alle mani dell'utente. Per i motivi sopra citati è stato deciso di seguire il secondo approccio.

Sono state scelte 16 diverse persone, maschi e femmine, ed è stato creato un database con le loro mani, rappresentate da un modello tridimensionale a cui è associato uno scheletro. Per ogni scheletro sono stati definiti dei vincoli di rotazione tali da garantire consistenza con il movimento reale della mano. La tecnica utilizzata per mappare i movimenti dello scheletro in una deformazione del modello è il *Linear Blend Skinning*, nel Capitolo 4 ne verranno definiti i concetti base. Inizialmente era stato deciso di seguire il metodo introdotto in *SCAPE* [8] e di adattarlo alla mano. Tale metodo prevede di trattare separatamente le deformazioni dovute a cambiamenti di posa e forma; richiede quindi due distinti database, uno comprendente i modelli di una stessa mano in diverse posizioni e l'altro comprendente modelli di mani diverse nella stessa posizione. Avere a disposizione un database che evidenzia la variabilità di forma e dimensioni delle mani, e del relativo scheletro, è indispensabile per il riconoscimento della posa di una mano, che è il passaggio successivo verso il tracciamento della mano. Supponendo di voler applicare il 'metodo *SCAPE*' ai modelli creati, sarebbe necessario aggiungere un database, per lo studio della posa, ad ognuno dei 16 modelli. Tale approccio richiederebbe troppo tempo per le acquisizioni e la costruzione dei modelli. Grazie alla possibilità di animare oggetti tridimensionali con il computer, è possibile evitare di acquisire la mano di un soggetto in diverse posizioni. Nello specifico è stato utilizzato il software Maya[®] per ricostruire le diverse posizioni a partire dalla posa base.

Capitolo 2

La Mano

2.1 Anatomia

La mano può essere pensata come l'espansione del nostro pensiero e interviene nella comunicazione tra individui, rendendo quest'organo parte integrante della vita di ciascun essere umano. Il meccanismo di movimento della mano è arrivato a un grado di complessità unico tra i mammiferi, contiene in totale 27 ossa, suddivise in tre gruppi: carpo, metacarpo e falangi (Figura 2.1).

Il *carpo* è costituito da due file di ossa, prossimale e distale, tutte di forma cubica che formano un piano osseo concavo ventralmente. Procedendo dal lato ulnare a quello radiale si trova:

- **fila prossimale:** l'osso pisiforme, il piramidale, il semilunare e lo scafoide;
- **fila distale:** l'osso uncinato, il capitato, il trapezoide e il trapezio.

Il *metacarpo* è costituito da cinque ossa lunghe, disposte l'una accanto all'altra, articolate con la fila distale del carpo.

Le *falangi* costituiscono lo scheletro delle dita, che risultano tutte costituite da tre ossa lunghe: prossimale, media e distale, ad eccezione del pollice che ne possiede due: prossimale e distale.



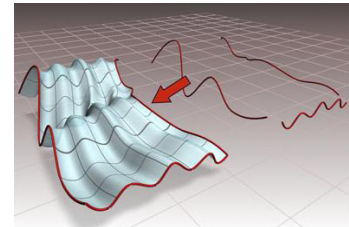
Figura 2.1: Scheletro di una mano [14].

I riferimenti in Figura 2.1 sono: **1**, 3^a falange; **2**, 2^a falange; **3**, 1^a falange; **4**, osso sesamoide; **5**, falange distale del pollice; **6**, falange prossimale del pollice; **7**, 1^o osso metacarpale; **8**, trapezoide; **9**, trapezio; **10**, scafoide; **11**, radio; **12**, ulna; **13**, semilunare; **14**, piramidale; **15**, pisiforme; **16**, capitato; **17**, uncinato; **18**, 5^o osso metacarpale.

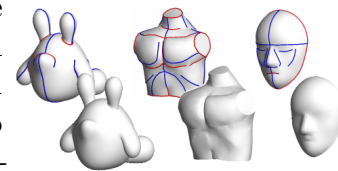
2.2 Modello

Il primo passo per la creazione di un oggetto animabile è la modellizzazione ed esistono diversi metodi per farlo.

- **NURBS:** Ci sono diversi strumenti per la creazione di oggetti tridimensionali a partire da forme geometriche primitive o NURBS [5], modificandole con opportune estrusioni, unioni, simmetrie, in modo da ottenere la forma desiderata. Esistono diversi software molto potenti che permettono tali operazioni, sia commerciali come 3ds Max[®] o Maya[®] [1] sia OpenSource come ad esempio Blender.



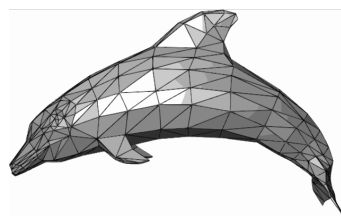
- **Schizzo interattivo:** Da qualche anno sono stati introdotti dei software per creare oggetti 3D partendo da una serie di schizzi 2D, un esempio è Teddy [19]. Il modellatore disegna lo schizzo in un piano bidimensionale e il sistema interpola i dati creando un oggetto tridimensionale. Un esempio più recente è FiberMesh [4] che permette di realizzare superfici nello spazio, da collezioni di curve 3D.



- **Scansione dell'oggetto** Un approccio completamente diverso è quello della scansione dell'oggetto, nello specifico della mano, con uno scanner 3D. Con questa tecnica è possibile riprodurre una copia della mano di un soggetto con una precisione inferiore al millimetro.



I dati ottenuti sono una nuvola di punti rappresentante la mano in uno spazio tridimensionale, dove ogni punto possiede diverse informazioni ad esempio la posizione, rappresentata dalle tre coordinate spaziali e il colore se lo scanner possiede una fotocamera a colori. A questa nuvola possono essere applicate diverse tecniche di ricostruzione della superficie per creare una mesh semplificata e priva di punti isolati e “buchi”, spesso presenti [17].



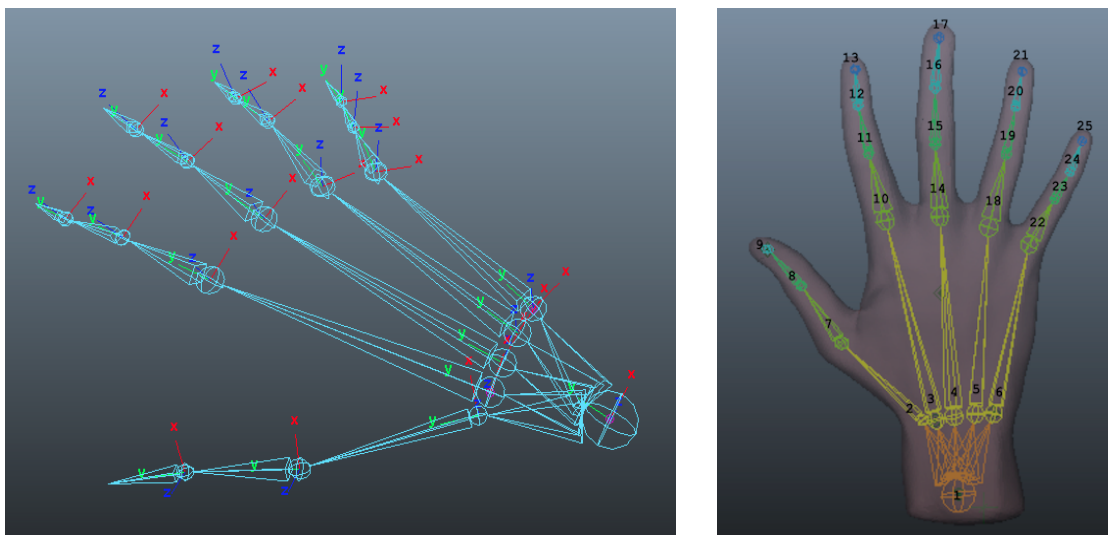
Il risultato di tali processi è una mesh di triangoli, cioè un insieme di triangoli che condividono uno spigolo o un vertice a due a due. Questo tipo di rappresentazione permette una completa facilità di rappresentazione, visualizzazione e manipolazione dell'oggetto ottenuto.

Come verrà spiegato nel Capitolo 6 il modello della mano verrà creato utilizzando uno scanner 3D, è stato dunque utilizzato il terzo metodo sopra menzionato.

Capitolo 3

Lo Scheletro

Lo scheletro è una struttura articolata gerarchicamente che rappresenta in maniera schematica la composizione di uno scheletro umano. È costituito da oggetti rigidi (ossa) collegati tra loro attraverso dei joints (giunti), la cui posizione e numero determina l'accuratezza del modello ad esso associato. Maggiore è il numero di joints utilizzati, maggiore sarà la precisione ottenuta nell'animazione del modello, ma più complesso sarà gestirne il tracciamento in seguito. Ogni joints può ruotare attorno a uno, due o tre assi ortogonali tra loro, che ne definiscono i gradi di libertà (DOF). Possono essere applicate delle limitazioni nella rotazione delle ossa per simulare il comportamento reale dello scheletro. Solitamente si preferisce definire un sistema di coordinate locale (Figura 3.1a), in questo modo ogni osso può riferire la propria rotazione rispetto al proprio genitore, procedendo fino a raggiungere la radice dello scheletro, la cui rotazione coinvolge l'intera struttura.



(a) Sistema di coordinate locale.

(b) Scheletro inserito nella mano e numerazione dei joint.

Figura 3.1: Sistema di coordinate locale e posizione dello scheletro

Questo concetto prende il nome di *Kinematic Chain* ed è spiegato nel dettaglio nel Capitolo 4.

Il modello adottato è uno scheletro composto da 24 ossa e 25 joints, posizionate come in Figura 3.1b. Per semplicità il carpo è stato modellato con 5 segmenti disposti a ventaglio, metacarpo e falangi invece rispecchiano l'anatomia umana. La disposizione dello scheletro è stata fissata manualmente, ma potrebbe essere eseguita automaticamente utilizzando diverse tecniche come riportato ad esempio in [9] oppure utilizzando una versione modificata di Pinocchio [10], un software per la collocazione automatica di uno scheletro all'interno di una mesh.

3.1 Gradi di libertà e vincoli

In totale ci sono 29 gradi di libertà (solo rotazioni), dei quali però 6 sono della radice (3 di rotazione e 3 di traslazione) che è stata omessa dalla Tabella 3.1; naturalmente per evitare pose non realistiche, non tutte le configurazioni di questi gradi di libertà sono possibili, perciò sono stati imposti dei vincoli di rotazione alle singole ossa come riassunto in tabella Tabella 3.1. Gradi di libertà e vincoli sono stati decisi autonomamente, tuttavia tali scelte trovano conferma su diversi articoli [13, 11]. La tabella riportata si riferisce al caso particolare di un modello; mentre i gradi di libertà imposti sono gli stessi per tutti i modelli nel database, i vincoli di rotazione sono propri di ciascuna mano. Tali vincoli sono espressi in gradi e rappresentano la massima rotazione ammessa da ciascun segmento, in relazione al proprio sistema di riferimento (Figura 3.1a), rispetto alla posa base (Figura 3.1b).

Dito	Joint	DOF	Range x [°]	Range y [°]	Range z [°]
Pollice	2	2	-50/10	-	-40/30
	7	1	-90/10	-	-
	8	1	-80/10	-	-
Indice	3	1	-3/0	-	-
	10	2	-90/0	-	-20/10
	11	1	-120/0	-	-
	12	1	-80/0	-	-
Medio	4	0	-	-	-
	14	2	-90/0	-	-20/20
	15	1	-120/0	-	-
	16	1	-80/0	-	-
Anulare	5	1	-3/0	-	-
	18	2	-90/0	-	-20/20
	19	1	-120/0	-	-
	20	1	-80/0	-	-
Mignolo	6	1	-4/0	-	-
	22	2	-90/0	-	-10/20
	23	1	-120/0	-	-
	24	1	-80/0	-	-

Tabella 3.1: Vincoli e gradi di libertà dello scheletro.

Capitolo 4

Animazione

Ci sono diverse tecniche per animare o semplicemente muovere il modello della mano, dal momento che lo scheletro è stato fissato alla mesh, per muoverla è necessario modificare la disposizione delle ossa. È possibile agire direttamente sullo scheletro oppure modificare la tensione dei muscoli provocando una riorganizzazione delle ossa. Se l'intento di questa tesi fosse di realizzare un'animazione realistica di una mano sicuramente si sarebbe impiegato un modello comprendente anche la struttura muscolare [16] o [11], ma ai soli fini del tracciamento, risulterebbe difficile da implementare e di scarsa utilità, perciò è stata adottata la prima soluzione malgrado comporti una semplificazione del modello.

4.1 Cinematica diretta e inversa

Le tecniche tradizionali per l'animazione di un oggetto sono la cinematica diretta (FK = Forward Kinematics) e la cinematica inversa (IK = Invers Kinematics). Entrambe hanno come scopo quello di muovere una catena di joints (catena cinematica) ma i procedimenti da seguire sono diversi. Prendendo in considerazione l'oggetto in Figura 4.1, possiamo immaginare che rappresenti schematicamente un braccio meccanico ancorato alla base (A), libero di muovere un utensile posto nell'altra estremità (B) grazie a una serie di giunti. Consideriamo il braccio in una certa configurazione iniziale (Figura 4.1a) e supponiamo di voler spostare l'utensile per raggiungere una nuova posizione (Figura 4.1b), ogni giunto dovrà eseguire delle rotazioni fino a raggiungere la nuova configurazione.

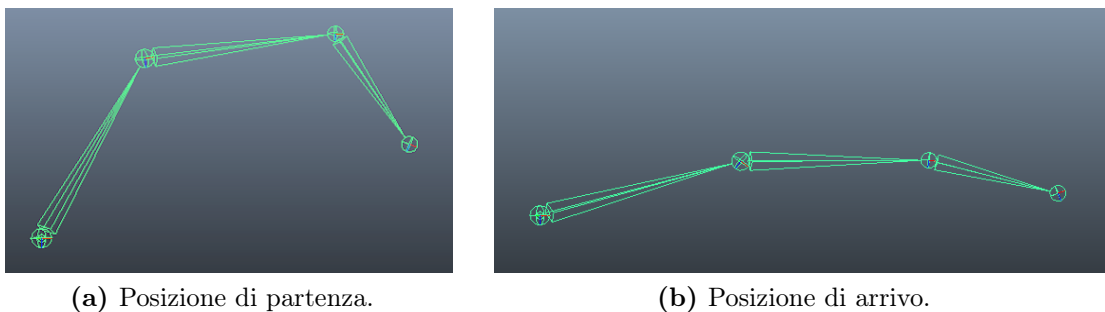


Figura 4.1: Esempio di catena cinematica

Con la cinematica diretta, è necessario ruotare ogni singolo pezzo fino a raggiungere la configurazione desiderata, mentre con la cinematica inversa è sufficiente traslare l'estremo giunto della catena (B) e il sistema provvederà a calcolare le singole rotazioni da applicare ai vari segmenti. Il primo approccio (FK) fornisce un controllo completo nel movimento, ma la definizione della posa deve essere fatta segmento per segmento e quando la struttura da animare diventa articolata la complessità aumenta. Per risolvere questo inconveniente è stata introdotta la cinematica inversa, questo metodo aggiunge una serie di vincoli allo scheletro per ottenere pose realistiche con meno sforzo. Solitamente il secondo approccio (IK) richiede un minor numero di operazioni da eseguire, nell'esempio considerato sono necessarie tre rotazioni con la cinematica diretta, mentre una sola traslazione utilizzando la cinematica inversa.

4.2 Linear Blend Skinning

Il "Linear Blend Skinning" (LBS) è un metodo comunemente usato per la deformazione della superficie di un modello [7]. Ogni vertice della mesh è in relazione con ogni segmento dello scheletro, che contribuisce alla deformazione del vertice. Nel caso specifico della mano, non saranno tutte le ossa a contribuire alla deformazione di tutti i punti della mesh, ma solo un sottoinsieme di esse; ad esempio è immediato verificare che il movimento del pollice ha un'influenza praticamente nulla nella deformazione del mignolo.

Ad ogni vertice è associato un insieme di pesi $(w_{i,j})$, uno per osso presente nello scheletro, in modo tale da rispettare le seguenti relazioni:

$$0 \leq w_{i,j} \leq 1 \quad , \quad w_i = \sum_j w_{i,j} = 1$$

dove i indica il vertice i -esimo, mentre j rappresenta l'osso j -esimo.

Assegnare un peso a ciascun vertice, per ogni segmento dello scheletro, risulta un approccio difficoltoso e difficilmente gestibile. Esistono dei software che ne facilitano l'assegnazione, fornendone un'iniziale ipotesi di distribuzione. In Maya[®] ad esempio viene fornita una distribuzione iniziale in funzione della distanza euclidea tra mesh e scheletro. L'utente può modificare tale assegnazione operando per via grafica con gli strumenti forniti dal software. Nei modelli realizzati, questi pesi sono stati assegnati sperimentalmente, valutando le deformazioni in pose differenti della mano e regolando i valori fino al raggiungimento di un'adeguata precisione.

Capitolo 5

Hand Scanner

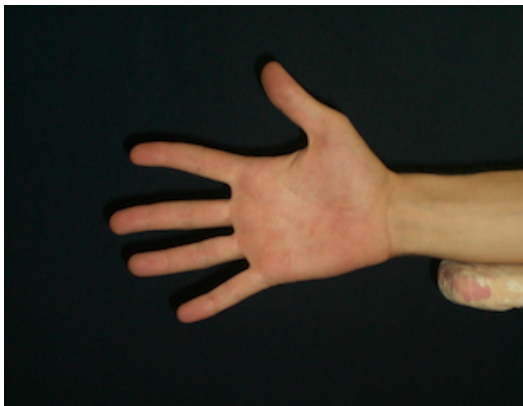
5.1 Premessa

In questo capitolo verrà descritto il macchinario realizzato per la scansione delle mani. Solitamente per ottenere una copia virtuale di un oggetto fisico utilizzando uno scanner 3D si posiziona l'oggetto in una struttura rotante e si eseguono diverse scansioni fino a ricoprire l'intera superficie. Nonostante le dimensioni ridotte, la mano di una persona è molto difficile da acquisire con uno scanner 3D. La difficoltà principale è dovuta al fatto che non si tratta di un oggetto indipendente, ma connesso all'estremità del braccio, di conseguenza non è possibile eseguire delle rotazioni complete attorno a tutti gli assi senza spostare il corpo. Un'ulteriore complicazione riguarda la forma, la presenza delle dita richiede un aumento del numero di scansioni per permettere allo scanner di ricoprire tutta la superficie da acquisire.

Il primo problema prevede due soluzioni: effettuare un calco della mano e acquisire quello (poco pratico se si vuole acquisire più di una mano), oppure invertire il ruolo di mano e scanner: la mano è mantenuta ferma e lo scanner esegue delle rotazioni attorno ad essa (soluzione adottata). Il secondo problema invece è stato più difficile da risolvere, esiste un trade-off tra numero di scansioni e integrità della superficie acquisita. Grazie alla presenza di numerose tecniche di ricostruzione di superfici, ad esempio algoritmo ICP (Iterative Closest Point) [6] o tecniche per "riempimento di buchi" nelle mesh [17, 12], è possibile ridurre il numero di scansioni da effettuare, ottenendo comunque un buon risultato.

Dopo qualche settimana di test effettuati con un rudimentale marchingegno sono sorti alcuni problemi: per permettere di effettuare una scansione completa, veniva richiesto all'utente di appoggiare l'avambraccio in un sostegno e di mantenere immobile la mano per qualche minuto (Figura 5.1a). Questa è una condizione necessaria ai fini di ottenere un buon modello e difficilmente una persona riesce a mantenere ferma la mano senza ulteriori punti di appoggio, diverse infatti sono state le persone che hanno giudicato tale approccio alquanto scomodo. Un'ulteriore limitazione di questo congegno è data dai gradi di libertà dello scanner, era possibile effettuare delle rotazioni attorno alla mano solamente in un piano, con piccoli

aggiustamenti di inclinazione. Per questi motivi è stato deciso di realizzare una struttura unica, robusta, di facile impiego e che garantisca all'utente una posizione più comoda per l'acquisizione (Figura 5.1b).



(a) Appoggiata solamente sull'avambraccio



(b) Appoggiata all'intera struttura

Figura 5.1: Posizione della mano nelle diverse strutture

5.2 Struttura

L'intera struttura è realizzata con profili di alluminio, uniti tra loro da angolari di fissaggio e innesti a vite (Figura 5.2), che rendono pratica qualsiasi modifica e riposizionamento delle singole parti.

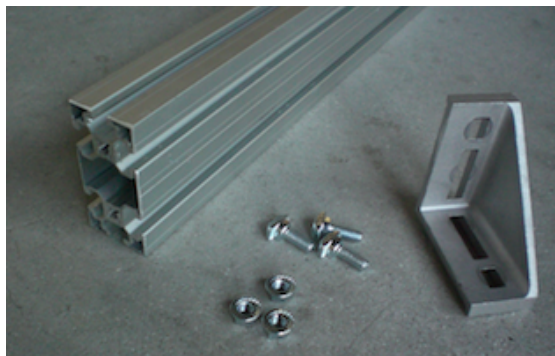


Figura 5.2: Componenti utilizzati

La base ha una forma ad "H" per garantire una sufficiente robustezza, alle quattro estremità sono stati posizionati dei piedini regolabili in altezza per stabilizzare la struttura anche in presenza di un fondo non regolare. Un ponte sorregge un motoriduttore 1:100 che tramite un'asse di acciaio trasmette il moto a un braccio regolabile alla cui estremità è fissato lo scanner. Tramite una manovella posizionata nel riduttore è possibile regolare la posizione del braccio con una precisione inferiore al grado. Per avere un riferimento della posizione angolare del braccio, è stato inserito un conta giri.



Figura 5.3: Alcune foto del macchinario

Il braccio è composto da due profili di alluminio ortogonali tra loro, che è possibile far scorrere per posizionare lo scanner alla distanza ottimale, in base al tipo di acquisizione. All'estremità del braccio orizzontale è fissato uno snodo al quale va fissato lo scanner tramite un pratico aggancio. Queste caratteristiche conferiscono allo scanner molti gradi libertà (Figura 5.4).



Figura 5.4: Braccio per il sostegno del laser

Nella parte verticale del ponte è ancorato il supporto per la mano che risolve il problema del primo macchinario, fornendo un comodo supporto all'utente. La Figura 5.5 evidenzia i molteplici snodi del supporto per la mano. Tale supporto è stato progettato in modo da ridurre al minimo lo sforzo per mantenere ferma la mano durante l'acquisizione. Oltre all'appoggio per l'avambraccio è stata utilizzata una lastra di Polyglass[®] sopra la quale è possibile appoggiare la mano riducendo al minimo qualsiasi causa di tremolio o movimento accidentale. Tale lastra interferisce minimamente con il laser dello scanner, sono però necessari alcuni accorgimenti; ad esempio bisogna evitare di posizionare lo scanner parallelamente alla lastra per evitare che riflessioni del laser interferiscano con la normale rilevazione, inoltre è necessario che l'utente non eserciti troppa pressione nella lastra, altrimenti verrebbe creato un modello inaccurato della mano (con il palmo schiacciato).



Figura 5.5: Supporto per la mano

Capitolo 6

Procedimento seguito

In questo capitolo verranno descritte le fasi seguite per la realizzazione e l'animazione di un modello di mano. Tali procedure sono state eseguite per tutti i 16 modelli nel database. All'indirizzo <http://vimeo.com/album/1642558> è possibile trovare tutti i video-tutorial realizzati, citati in questa tesi. Tali video sono protetti da password, se si è interessati al contenuto, si invii una e-mail al seguente indirizzo: giulio.marin.1@studenti.unipd.it.

Per la scansione delle prime mani è stata utilizzata una struttura provvisoria, per le successive, il macchinario descritto nel Capitolo 5 e il *3D Scanner HD* della Next Engine [2] con il supporto software *ScanStudio HD*. Oltre agli strumenti forniti da questo software, è stato utilizzato *MeshLab* [3], software OpenSource che implementa molti algoritmi per l'elaborazione di mesh triangolari. Per la creazione dello scheletro, il *rigging*¹ e l'animazione è stato utilizzato Maya[®] [1], uno tra i più potenti prodotti commerciali per la grafica 3D.

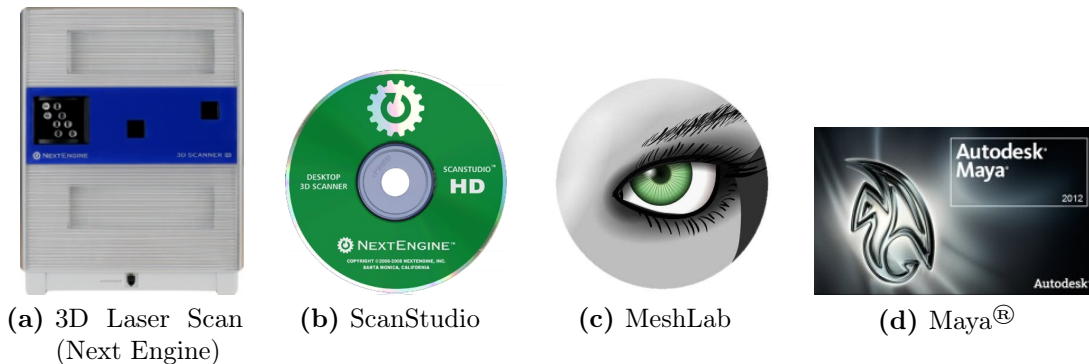


Figura 6.1: Strumenti utilizzati

¹Il *rigging* è quel processo che ingloba lo scheletro nella mesh.[9, 10]

6.1 Acquisizione della mano

Dopo lo studio di numerosi articoli riguardanti il tracciamento e la modellazione 3D di mani al computer, è iniziata la fase di acquisizione. Sono state effettuate diverse scansioni di prova utilizzando *ScanStudio*, variando impostazioni e parametri dello scanner, fino ad ottenere il giusto compromesso tra precisione e velocità delle acquisizioni. Per rendere più agevole la fase di elaborazione (Capitolo 6.2) sono stati evidenziati alcuni punti della mano con un pennarello. Tali marcatori sono stati applicati nelle zone critiche della mano e in altri punti strategici, come evidenzia la Figura 6.2. Le zone critiche sono quelle porzioni di pelle che si trovano all'estremità di una scansione e spesso è necessario utilizzare nella fase di allineamento.



Figura 6.2: Marker applicati alla mano

Sono state effettuate in media 4 scansioni per mano, il minimo per acquisire l'intera superficie. Come si può notare nella sequenza in Figura 6.3, la differenza angolare tra due scansioni è di circa 90° . In alcuni casi è stato necessario effettuare scansioni aggiuntive, per permettere al laser dello scanner di raggiungere tutte le zone da acquisire. È utile far notare che gli angoli riportati in Figura 6.3 non sono da considerarsi vincolanti. Dal momento che la mano di ciascun individuo ha una propria morfologia, le posizioni dello scanner sono state valutate caso per caso.

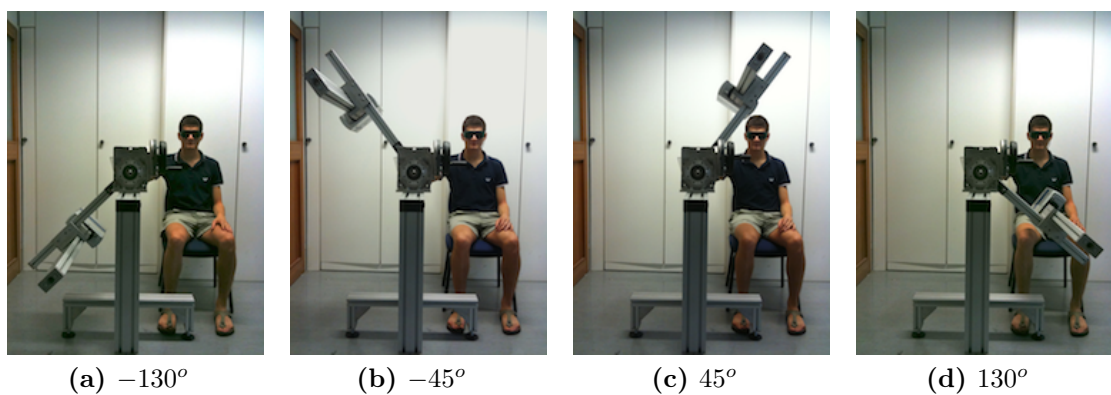


Figura 6.3: Posizioni angolari delle varie scansioni rispetto alla verticale.

Le scansioni sono state salvate in formato *.ply* per poter essere elaborate anche da un software esterno. La Figura 6.4 mostra un esempio di superficie ottenuta con una scansione. Tipicamente una mesh è composta da 10000 vertici e ognuno contiene le informazioni di posizione nello spazio. Nell'esempio considerato i vertici contengono anche l'informazione di colore della mano reale. Nonostante la scansione a colori richieda dei passaggi aggiuntivi rispetto alla scansione in bianco e nero, il tempo impiegato è pressoché lo stesso. Manipolare una mesh a colori comporta dei grossi benefici nella fase di allineamento, descritta nel prossimo paragrafo, perché permette di riconoscere più facilmente i marcatori applicati.



Figura 6.4: Mesh ottenuta da una scansione

Questa fase ha richiesto molto tempo per le scansioni effettuate con il primo macchinario, che richiedeva una lunga procedura di calibrazione e un numero maggiore di scansioni per garantire l'acquisizione dell'intera superficie. Con la seconda struttura (Figura 5.3) invece l'acquisizione non ha richiesto grossi sforzi e la qualità delle scansioni è migliorata, grazie soprattutto alla presenza di un supporto stabile per la mano.

È stato realizzato un video-tutorial che mostra le fasi del processo di acquisizione ed è reperibile al seguente indirizzo: <http://vimeo.com/giuliomarin/handscanner>.

6.2 Elaborazione immagini

I dati provenienti dalla fase precedente sono un insieme di scansioni della stessa mano che necessitano di essere unite e adattate per riprodurre fedelmente la mano acquisita. Un primo allineamento delle mesh è stato eseguito con lo strumento *Align* di ScanStudio. Per allineare due superfici è sufficiente posizionare 3 marcatori in entrambe le mesh e avviare il processo ICP. In aggiunta è possibile raffinare l'allineamento con un comando apposito. La fase successiva è quella di fondere le varie mesh in una superficie unica (Figura 6.5) con lo strumento *Flatten Visible Layer* di MeshLab.



Figura 6.5: Mesh ottenuta dopo l'allineamento e la fusione

Per ottenere un modello accurato è necessario correggere la mesh con algoritmi di ricostruzione superficiale, in particolare è stato utilizzato l'algoritmo di Poisson [15] implementato in MeshLab. In questo modo si ottiene una mesh unica (Figura 6.6) di circa 50000 vertici che sarà il modello della mano.

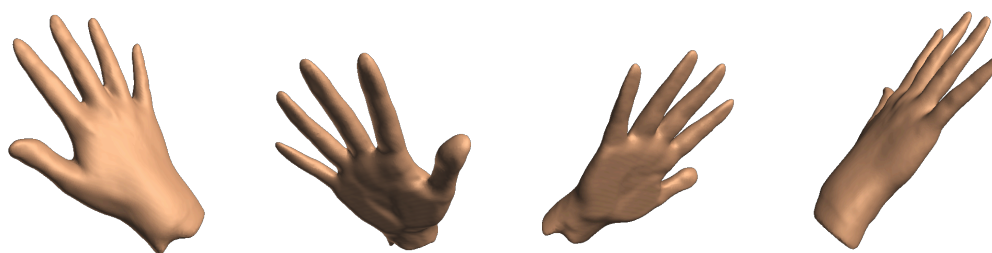


Figura 6.6: Mesh ottenuta dopo aver applicato l'algoritmo di Poisson

Per facilitare le operazioni di animazione è stato eseguito un ulteriore processo di semplificazione, utilizzando il metodo *Quadric Edge Collapse Decimation* [18], ottenendo una riduzione di circa 80% del numero di vertici della mesh (Figura 6.7).

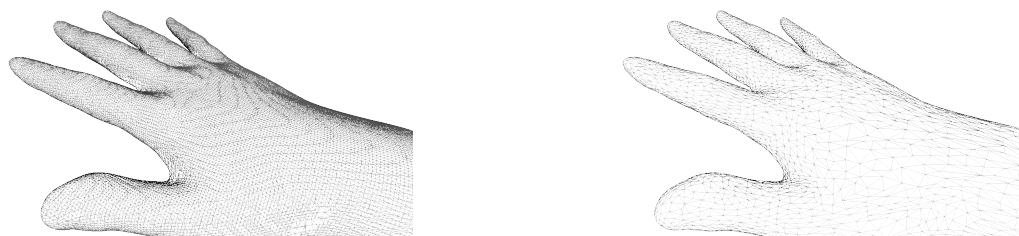


Figura 6.7: Confronto tra mesh originale e mesh ridotta

È stato realizzato un video-tutorial che mostra tali procedimenti ed è reperibile al seguente indirizzo: <http://vimeo.com/giuliomarin/meshlabtools>.

6.3 Realizzazione dello scheletro

Per costruire uno scheletro in Maya[®] è necessario utilizzare il comando *Joint Tool* della scheda *Skeleton*. Come specificato nel Capitolo 3 e 4, è stato creato uno scheletro formato da 25 joints. Si procede posizionando il primo joint radice in corrispondenza del polso, successivamente si collocano i 5 joint a livello del carpo, poi gli altri fino a raggiungere l'estremità delle dita. Si viene così a creare una struttura ad albero. Nella costruzione dello scheletro è consigliabile però procedere secondo l'algoritmo di visita in profondità di un grafo piuttosto che in ampiezza, questo accorgimento permette di velocizzare la costruzione.

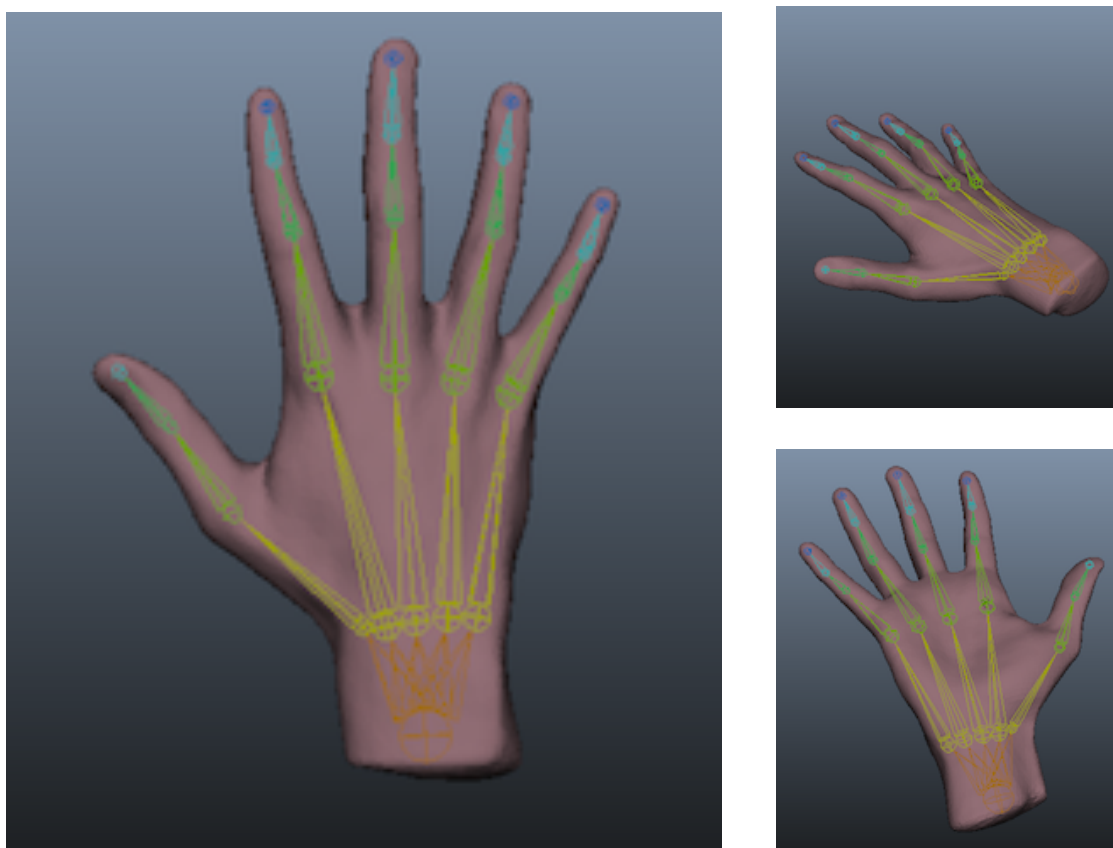


Figura 6.8: Scheletro di un modello

Una volta creato lo scheletro è necessario adattarlo alla forma della mano, per una corretta modellazione la posizione dei joint deve rispettare la posizione reale delle ossa. Quando lo scheletro avrà assunto la posizione definitiva è possibile inglobarlo all'interno della mesh con il comando *Smooth Bind* della scheda *Skin*. Questo comando prevede diverse opzioni, è sufficiente verificare che il tipo di 'skinning' sia lineare e che i pesi vengano assegnati relativamente alla distanza dalle ossa, le altre opzioni non influiscono pesantemente sul risultato.

È stato realizzato un video-tutorial per la creazione dello scheletro in Maya ed è reperibile al seguente indirizzo: <http://vimeo.com/giulioamarin/skeletonmaya>.

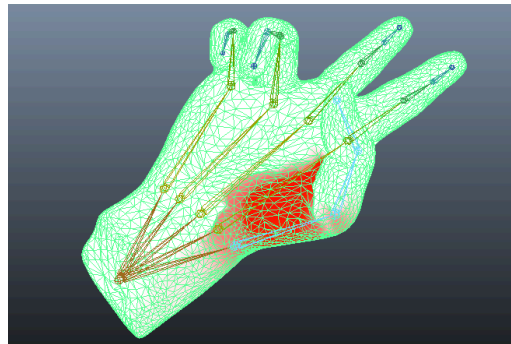
6.4 Animazione del modello

Ora che il modello della mano con lo scheletro è stato realizzato, resta da implementare l'animazione. Come spiegato nel Capitolo 4, è necessario creare delle catene cinematiche, che in Maya[®] si realizzano con il comando *IK Handle Tool* della scheda *Skeleton*. Si possono creare ad esempio 5 catene, che partono dal giunto nel carpo e arrivano all'astremità di ogni singolo dito. Per ottenere dei buoni risultati è comunque consigliabile adottare sia l'approccio diretto (FK), sia l'approccio inverso (IK), questo perchè è stato notato che la cinematica inversa non permette di ottenere tutte le combinazioni possibili. Dopo aver salvato tutte le posizioni come spiegato in Appendice A, è possibile animare la mano secondo una sequenza assegnata a piacimento.

In questa fase sono stati anche calibrati i pesi assegnati in automatico dal programma, in particolare è stato utilizzato il comando *Paint Weight Tool* della scheda *Skin* (Figura 6.9b). Con questo strumento è possibile modificare agevolmente i pesi relativi a ciascun osso, dal momento che permette di aggiungere o togliere peso ad un insieme di vertici, o di sovrascrivere il valore desiderato.



(a) Screenshot dell'animazione



(b) Assegnazione dei pesi

Figura 6.9: Particolari della fase di animazione

Per rendere più agevole l'interazione con Maya[®] sono stati creati degli script in MEL (Maya Embedded Language) riportati nel Appendice A.

È stato realizzato un video-tutorial per la modifica dei pesi e l'animazione in Maya ed è reperibile al seguente indirizzo: <http://vimeo.com/giuliomarin/animationmaya>.

Capitolo 7

Conclusioni

Come introdotto nel Capitolo 1, in questa tesina sono state descritte le procedure per la realizzazione di modelli tridimensionali di mani di persone, ad ognuno dei quali è associato uno scheletro.

I risultati ottenuti forniscono la base di un progetto che prevede di sostituire le mani al mouse, nell'interazione con il computer. Da qualche anno, l'introduzione degli schermi touch screen, ha permesso agli utenti di interagire mediante una o più dita, il progetto prevede invece di sfruttare la posa dell'intera mano, servendosi solamente di una videocamera, ormai presente in tutti i computer.

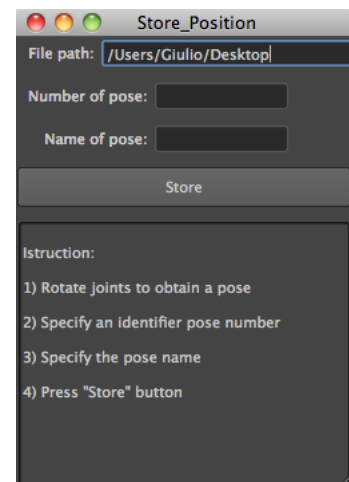
Il database creato fornisce inoltre un utile supporto alle applicazioni di computer grafica 3D, dato che non è disponibile online una collezione di modelli tridimensionali di mani.

Le operazioni descritte nel Capitolo 6 sono state ripetute per 16 mani di persone differenti. Il database ottenuto contiene un numero di modelli sufficiente a garantire lo sviluppo del progetto di tracciamento e riconoscimento di una mano. Tale database verrà presto aggiornato con nuovi modelli di mano e scheletro. In ogni caso, seguendo le procedure descritte nei video-tutorial realizzati, è possibile aggiungere modelli coerenti con quelli del database, utilizzando uno scanner 3D e qualsiasi software equivalente a quelli utilizzati.

Appendice A

Codice

Sono state registrate diverse configurazioni tipiche della mano in file di testo esterni al programma e ad ognuna di esse è stato associato un numero identificatore e un nome. Per farlo è stato creato lo script MEL sotto riportato (`store_pose.mel`). Oltre a specificare il percorso dove salvare i file, è necessario inserire il numero identificatore della posa ed un nome che permetta di distinguerla dalle altre. Premendo il tasto Store viene creato il file 'PosaX.txt' dove X è il numero identificatore assegnato in precedenza. Al suo interno sono memorizzati il nome assegnato (prima riga) e le rotazioni relative di tutti i joint dello scheletro.



store_pose.mel

```
string $instruction = "1) Rotate joints to obtain a pose\n\n2) Specify an identifier\npose number\n\n3) Specify the pose name\n\n4) Press \"Store\" button";

// User Interface
if (`window -exists Store_Position`==1) { deleteUI Store_Position; }

window -rtf 1 -wh 100 100 Store_Position;
    columnLayout -rs 10 -cw 250 -cat "both" 0;
        textFieldGrp -l "File path: " -cw2 60 220 -tx "/Users/Giulio/Desktop"path;
        textFieldGrp -l "Number of pose: " -cw2 100 100 pose_number;
        textFieldGrp -l "Name of pose: " -cw2 100 100 pose_name;
        button -h 30 -w 200 -l "Store" -c "startStore()";
        scrollLayout -h 200;
        text -al "left" -l ("\nInstruction:\n\n" + $instruction);
showWindow;

proc startStore() {

    // Define File Path
    string $path = `textFieldGrp -q -tx path`;
    string $filePath = $path + "/Pose" + (`textFieldGrp -q -tx pose_number`) + ".txt";

    // Open File
    $fileId = `fopen $filePath "w"`;

    string $joint;
    float $values[];
```

```

// Pose name
string $pose_name = `textFieldGrp -q -tx pose_name` + "\n";
fprintf $fileId $pose_name;

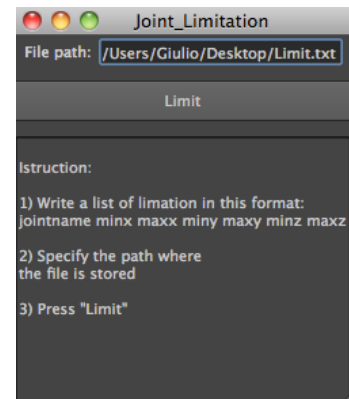
// Print into File
for($i=1; $i<26; $i++) {
    $joint = "joint" + $i;
    $values = getAttr ($joint + ".rotate");
    fprintf $fileId ($joint + "\ ");
    for($j=0; $j<3; $j++){
        fprintf $fileId ($values[$j]+"\ ");
    }
    fprintf $fileId "\n";
}

// Close File
fclose $fileId ;

// Clear textField
textFieldGrp -e -tx "" pose_number;
textFieldGrp -e -tx "" pose_name;
}

```

Ogni modello può avere dei limiti nelle rotazioni differenti, per semplicità è stato creato uno script (`read_input.mel`) per assegnare i vincoli di rotazione di ogni joint da un file di testo. Dopo aver specificato il file che si desidera utilizzare, premendo 'Limit' si assegnano i vincoli a tutti i joint dello scheletro. Il file deve contenere tante righe quanti i joint dello scheletro, nella sintassi:



```
nomeJoint minRotX maxRotX minRotY maxRotY minRotZ maxRotZ
```

`read_input.mel`

```

string $instruction = "1) Write a list of position in \"Input.txt\" file\n\n2) Specify
    the path directory where\nthere are input file and poses files\n\n3) Press \"Load\"
    button\n\n4) Press \"Play/Stop\" button and\nlook the hand\n\n5) You can change
    the animation speed\nand select a frame you want to see";

// User Interface
if (`window -exists Animation`==1) { deleteUI Animation; }

window -rtf 1 -wh 100 100 Animation;
    columnLayout -rs 10 -cw 300 -cat "both" 0;
        textFieldGrp -l "File path: " -cw2 60 220 -tx "/Users/Giulio/Desktop"path;
        button -l "Load" -h 50 -c "process()";
        button -l "Play/Stop" -h 50 -command "playButtonForward" playAnimation;
        floatSliderGrp -l "speed: " -cw3 40 50 100 -cc "speedAnimation()" -field true -
            min 0.1 -max 10 -value 1 speed;
        floatSliderGrp -l "Frame: " -cw3 40 50 100 -cc "gotoFrame()" -field true -min 1
            -max 24 -value 1 -pre 0 frame;
        scrollLayout -h 350;
        text -al "left" -l ("\nInstruction:\n\n" + $instruction) list_id;
showWindow;

```



```

// Control speed of animation
proc speedAnimation() {
    playButtonForward;
    playbackOptions -ps `floatSliderGrp -q -v speed`;
    playButtonForward;
}

// Goto a specific frame
proc gotoFrame() {
    currentTime `floatSliderGrp -q -v frame`;
}

// Main method
proc process() {

    // Delete previous keys
    cutKey -hi "below" joint1;

    // Define file path and file name
    string $path = `textFieldGrp -q -tx path`;
    string $input_file = $path + "/Input.txt";
    string $poses_file = $path + "/Pose";

    string $list = "\nList of positions:\n[ number_of_frame) Position_name ]\n\n1) Base
    \n\n";

    // Open file
    int $open_result = $fileId = `fopen $input_file "r"`;
    if ($open_result == 0){
        button -e -en false playAnimation;
        error "There isn't correct input file in this directory";
    }
    button -e -en true playAnimation;

    // Get string
    string $sequence = `fgetline $fileId`;
    $sequence = strip($sequence);

    // Create array
    string $buffer[];
    string $current_joint[];

    // Tokenize input string
    int $total = `tokenize $sequence "\ " $buffer`;

    // Set first pose
    currentTime 1;
    select joint1;
    AssumePreferredAngle;
    SetKey;

    // Set other poses
    for($i=0; $i<$total; $i++){

        currentTime (24*($i+1));

        int $id = $buffer[$i];

        // Open file
        $open_result = $filePose = `fopen ($poses_file + $id + ".txt") "r"`;
        if ($open_result == 0){
            button -e -en false playAnimation;
            error "There aren't all poses file necessary in this directory.";
        }
        button -e -en true playAnimation;

        // Get the first line
        string $nextLine = `fgetline $filePose`;
        $list += (($i+1)*24) + " " + $nextLine + "\n";

        // Loop until the string size is zero (No data)

```

```

while (size($nextLine) > 0) {
    // Get next line and continue
    $nextLine = `fgetline $filePose`;

    // Strip whitespace from the beginning and end of the line
    string $cleanLine = strip($nextLine);

    // Tokenize input string
    tokenize $cleanLine "\ " $current_joint;

    // Save values
    int $rot_x = $current_joint[1];
    int $rot_y = $current_joint[2];
    int $rot_z = $current_joint[3];

    // Perform joint rotation
    setattr ($current_joint[0] + ".rotate") $rot_x $rot_y $rot_z;
}

// Close file
fclose $filePose;

// Set Key
select joint1;
SetKey;
}

AssumePreferredAngle;

// Close File
fclose $fileId;

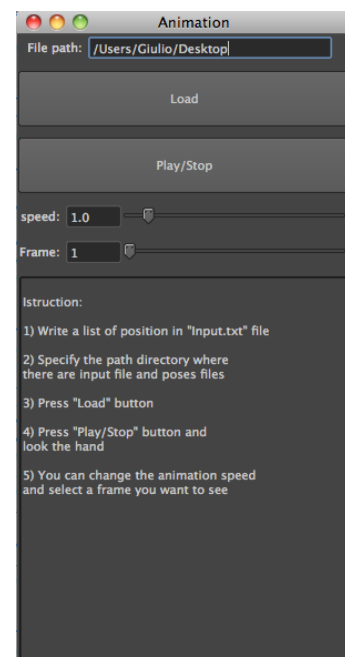
select -cl;

// Edit playback option
playbackOptions -max ($total*24) -aet ($total*24);
floatSliderGrp -e -max ($total*24) frame;

$list += "Ready to start animation.\n\nPress \"Play/Stop\" button\n";
text -e -label $list list_id;
}

```

Per visualizzare un'animazione di una sequenza di pose utilizzando lo script sotto riportato (`read_limit.mel`), è sufficiente riportarne in ordine gli identificatori, separati da uno spazio, in un file di testo di nome 'Input.txt'. È necessario specificare il percorso della cartella dove sono memorizzati i file delle posizioni e il file di input. Premendo 'Load', se tutti i file sono stati correttamente creati, viene memorizzata la sequenza desiderata, successivamente premendo 'Play/Stop' si avvia l'animazione. Con il primo cursore è possibile regolare la velocità di riproduzione, mentre con il secondo è possibile posizionarsi in un dato frame. Il display mostra la sequenza di pose nel formato:



numeroFrame) Nome assegnato alla posa

read_limit.mel

```

string $instruction = "1) Write a list of limitation in this format:\njointname minx maxx
    miny maxy minz maxz\n\n2) Specify the path where\nthe file is stored\n\n3) Press
    \"Limit\"";

// User Interface
if (`window -exists Joint_Limitation`==1) { deleteUI Joint_Limitation; }

window -rtf 1 -wh 100 100 Joint_Limitation;
    columnLayout -rs 10 -cw 250 -cat "both" 0;
        textFieldGrp -l "File path: " -cw2 60 180 -tx "/Users/Giulio/Desktop/Limit.txt"
            input_name;
        button -h 30 -w 200 -l "Limit" -c "limit()";
        scrollLayout -h 200;
        text -al "left" -l ("\nInstruction:\n\n" + $instruction);
showWindow;

proc limit() {
    // Define file path
    string $path = `textFieldGrp -q -tx input_name`;

    // Open file
    int $open_result = $fileId = `fopen $path "r"`;
    if ($open_result == 0){
        error "There isn't the correct file in this directory";
    }

    // Get the first line
    string $nextLine = `fgetline $fileId`;

    // Loop until the string size is zero (no data)
    while (size($nextLine) > 0) {

        // Strip whitespace from the beginning and end of the line
        string $cleanLine = strip($nextLine);

        // Create empty array
        string $current_joint[];

        // Tokenize input string
        tokenize $cleanLine "\ " $current_joint;

        // Save values
        int $min_x = $current_joint[1];
        int $max_x = $current_joint[2];
        int $min_y = $current_joint[3];
        int $max_y = $current_joint[4];
        int $min_z = $current_joint[5];
        int $max_z = $current_joint[6];

        // Perform joint rotation
        joint -e -lx $min_x $max_x -ly $min_y $max_y -lz $min_z $max_z $current_joint
            [0];

        // Get next line and continue
        $nextLine = `fgetline $fileId`;
    }
    print "\nDone\n";
}

```

Per comodità è stata creata un'icona rapida per ogni script con in comando *Save Script to Shelf...* dello Script Editor. È stato inoltre creato un video-tutorial che mostra tutte le operazioni da compiere per importare ed utilizzare gli script MEL. È disponibile all'indirizzo <http://vimeo.com/giuliomarin/melscript>.

Bibliografia

- [1] URL: <http://usa.autodesk.com/maya/>. (Cit. alle pp. 3, 12).
- [2] URL: <http://www.nextengine.com/>. (Cit. a p. 12).
- [3] URL: <http://meshlab.sourceforge.net/>. (Cit. a p. 12).
- [4] Nealen A. et al. “Fibermesh: Designing freeform surfaces with 3d curves”. In: *ACM Transactions on Graph* (2007). (Cit. a p. 3).
- [5] Piegl Les A. e Tiller Wayne. *The NURBS Book*. Springer, 1997. (Cit. a p. 3).
- [6] Amberg Brian, Romdani Sami e Vetter Thomas. “Optimal Step Nonrigid ICP Algorithms for Surface Registration”. In: *IEEE Conference on Computer Vision and Pattern Recognition* 14.1-3 (2007), pp. 1–8. (Cit. a p. 8).
- [7] Jacka David, Reid Ashley e Merry Bruce. “A comparison of linear skinning techniques for character animation”. In: (2007), pp. 177–186. (Cit. a p. 7).
- [8] Anguelov Dragomir et al. “SCAPE: shape completion and animation of people”. In: *ACM Trans. Graph* 24 (2005), pp. 408–416. (Cit. a p. 1).
- [9] Aujay Grégoire et al. “Harmonic Skeleton for Realistic Character Animation”. In: *ACM SIGGRAPH/Eurographics symposium on Computer animation* (2007), pp. 151–160. (Cit. alle pp. 5, 12).
- [10] Baran Ilya e Popović Jovan. “Automatic Rigging and Animation of 3D Characters”. In: *ACM SIGGRAPH* (2007). (Cit. alle pp. 5, 12).
- [11] Albrecht Irene, Haber Jörg e Seidel Hans-Peter. “Construction and Animation of Anatomically Based Human Hand Models”. In: *Eurographics/SIGGRAPH Symposium on Computer Animation* (2003). (Cit. alle pp. 5, 6).
- [12] Davis James et al. “Filling Holes in Complex Surfaces using Volumetric Diffusion”. In: *First International Symposium on 3D Data Processing, Visualization, and Transmission* (2002). (Cit. a p. 8).
- [13] John Lin, Ying Wu e T. S. Huang. “Modeling the constraints of human hand motion”. In: *Proceedings of the Workshop on Human Motion (HUMO’00)* (2000). (Cit. a p. 5).
- [14] Artico Marco e Castano Paolo et al. *ANATOMIA UMANA - PRINCIPI*. Edi-Ermes, 2005. (Cit. alle pp. iii, 2).
- [15] Kazhdan Michael, Bolitho Matthew e Hoppe Hugues. “Poisson Surface Reconstruction”. In: *Eurographics Symposium on Geometry Processing* (2006). (Cit. a p. 15).

- [16] Pratscher Michael et al. “Outside-In Anatomy Based Character Rigging”. In: *Eurographics/ACM SIGGRAPH Symposium on Computer Animation* (2005). (Cit. a p. 6).
- [17] Liepa Peter. “Filling Holes in Meshes”. In: *Eurographics Symposium on Geometry Processing* (2003). (Cit. alle pp. 3, 8).
- [18] Heckbert Paul S. e Garland Michael. “Optimal Triangulation and Quadric-Based Surface Simplification”. In: *Journal of Computational Geometry: Theory and Applications* (1999). (Cit. a p. 15).
- [19] Igarashi T., Matsuoka S. e Tanaka H. “Teddy: a sketching interface for 3d freeform design”. In: *Proceedings of the ACM SIGGRAPH* (1999). (Cit. a p. 3).
- [20] Wang Robert Y. e Popović Jovan. “Real-time hand-tracking with a color glove”. In: *ACM Transactions on Graphics (TOG)* 28.3 (2009). (Cit. a p. 1).