

POLITECNICO DI TORINO

Dipartimento di Automatica e Informatica

Corso di Laurea in
Ingegneria del Cinema e dei Mezzi di Comunicazione

Tesi di Laurea Magistrale

Workflow per l'animazione di personaggi virtuali tramite
Artificial Intelligence-based Motion Capture

Move.ai per il cortometraggio "Poppy"



Relatore

Prof. Riccardo Antonio Silvio Antonino

Candidato

Flavia Gentile

Anno Accademico 2022-2023

Abstract

L'*Artificial Intelligence-based Motion Capture (AI-based MoCap)* è una tecnica che permette di catturare il movimento di un soggetto e applicarlo ad un personaggio virtuale, al fine di animarlo tramite l'utilizzo dell'Intelligenza Artificiale.

Sin dal 1918, anno in cui Max Fleischer impiegò per la prima volta il rotoscopio per la realizzazione della serie *Out of the Inkwell*, le tecniche di trasposizione del movimento del corpo umano hanno continuato ad evolversi adattandosi alle tecnologie a disposizione nei diversi periodi storici, assumendo diversi nomi. L'automazione di questo processo, conosciuto oggi come *Motion Capture*, fu reso possibile dai dispositivi di *tracking*, che sfruttano sistemi piuttosto complessi come sensori, *markers* e *wearable devices*.

Negli ultimi anni, la rivoluzione tecnologica dovuta all'esponenziale sviluppo e alla crescente diffusione dell'Intelligenza Artificiale (IA) ha favorito la nascita di nuove tecniche. L'IA ha permesso la realizzazione di applicativi in grado di catturare il movimento e di trasferirlo autonomamente ai personaggi virtuali, minimizzando il margine di errore, l'intervento di animatori professionisti sul prodotto finale e i costi dei dispositivi necessari per realizzare la *MoCap*.

La presente tesi di laurea illustra la tecnica della *AI-based MoCap* applicata nel processo di animazione dei personaggi di "Poppy", cortometraggio di animazione prodotto dall'azienda di comunicazione Robin Studio Srl.

A partire dall'analisi delle principali applicazioni presenti sul mercato, i loro punti di forza e i loro limiti, l'elaborato spiega i motivi che hanno portato alla scelta del programma "Move.ai" per la realizzazione delle animazioni tramite *Motion Capture*.

Scopo della tesi è quello di fornire una *pipeline* di lavoro che faccia da linea guida per l'animazione tramite l'utilizzo di "Move.ai" di due principali categorie di personaggi virtuali, umani e robotici. Infine, l'elaborato ha l'obiettivo di illustrare le criticità, le potenzialità e i risvolti futuri che questa tecnologia avrà sulla vita di tutti i giorni.

Indice

Elenco delle figure	4
1 Introduzione	6
1.1 L'evoluzione della <i>Motion Capture</i>	6
1.1.1 I sistemi di acquisizione del movimento	9
1.1.2 <i>Workflow</i> generale per l'animazione tramite <i>Motion Capture</i> . .	12
1.1.3 Ambiti di applicazione della <i>MoCap</i> ed esempi	13
2 Stato dell'arte	19
2.1 Stato dell'arte	19
2.1.1 Applicazioni disponibili	19
2.1.2 Esempi	22
3 "Poppy"	24
3.1 Il progetto	24
3.2 La sinossi	25
3.3 I personaggi	26
3.4 Gli obiettivi	28
3.5 Uno sguardo alla <i>Virtual Production</i>	28
4 Software utilizzati	30
4.1 <i>Software</i> per la realizzazione di "Poppy"	30
4.1.1 Blender	30
4.1.2 Marvelous Designer	31
4.1.3 Substance 3D Painter	32
4.1.4 Unreal Engine 5	33
4.2 <i>Software</i> per l'animazione tramite <i>AI Motion Capture</i>	35
4.2.1 La scelta del <i>software</i> da utilizzare	35
4.2.2 <i>Workflow</i> per l'utilizzo di Move.ai per animare i personaggi . .	39
5 Workflow per l'animazione dell'Astronauta	45
5.1 <i>Pipeline</i> di lavoro seguita	45
5.2 Problemi e soluzioni	54
5.3 <i>Workflow</i> per l'animazione di personaggi umani tramite Move.ai	58

6	<i>Workflow</i> per l'animazione di Poppy	62
6.1	<i>Pipeline</i> di lavoro seguita	62
6.2	Problemi e soluzioni	76
6.3	Workflow per l'animazione di personaggi robotici con Move.ai	79
7	Conclusioni	84

Elenco delle figure

1.1	Rotoscopio	7
1.2	The Lord of the Rings: The Fellowship of the Ring (2001)	8
1.3	The Polar Express (2004)	8
1.4	Avatar (2009)	9
1.5	Workflow per la Motion Capture	12
1.6	Alice in Wonderland (1951)	14
1.7	Pirates of Caribbean (2007)	15
1.8	Avatar: The Way of Water (2022)	15
1.9	Esempio di task motori del protocollo sperimentale utilizzato, LAM	16
1.10	Baldur's Gate 3	17
1.11	Star Wars Jedi: Survivor	18
3.1	Logo del progetto Poppy	24
3.2	Earth Experience	25
3.3	Fiore OLED aperto	26
3.4	Poppy	27
3.5	Astronauta	27
4.1	Poppy in Blender	31
4.2	Astronauta in Marvelous Designer	32
4.3	Texture del fiore OLED realizzata in Substance 3D Painter e visualizzata in Unreal Engine 5	33
4.4	Deserto	34
4.5	Scogliera	34
4.6	Passaggi del workflow seguiti per scegliere il software	35
4.7	DeepMotion	36
4.8	Move.ai	36
4.9	Rokoko Vision	37
4.10	Wonder Studio	37
4.11	Workflow per l'animazione con Move.ai	40
5.1	Workflow seguito per animare l'astronauta con Move.ai	46
5.2	Modello Astronauta senza testa	47
5.3	Accessori Astronauta	47

5.4	Armatura Astronauta	48
5.5	Volume Calculator di Move.ai	50
5.6	Astronauta con guanti e stivali	52
5.7	Astronauta con guanti, stivali e casco	53
5.8	Astronauta con piedi storti e sollevati da terra	57
5.9	Stella di ossa	58
5.10	Workflow ottimale per l'animazione di personaggi umani con Move.ai .	59
6.1	Workflow seguito per animare Poppy con Move.ai	63
6.2	Model sheet del personaggio Poppy	64
6.3	Modello 3D di Poppy	64
6.4	Armatura Modello A di Poppy	65
6.5	Armatura Modello B di Poppy	66
6.6	Mano Modello A di Poppy	67
6.7	Mano Modello B di Poppy	68
6.8	Fase di <i>recording</i>	70
6.9	Ossa di controllo per l'Inverse Kinematics	71
6.10	Osso radice	71
6.11	Gradi di libertà per le componenti di Poppy	73
6.12	Poppy segue i movimenti del performer	76
6.13	Workflow ottimale per l'animazione di personaggi robotici con Move.ai	79

Capitolo 1

Introduzione

Il movimento è una delle principali caratteristiche del corpo umano, comprenderne i meccanismi per provare a riprodurlo è una delle sfide più difficili che la scienza affronta. Con il passare del tempo, le scoperte fatte in diverse discipline, come la cinematica, la meccanica e l'informatica, hanno portato alla nascita della *Motion Capture*, altrettanto nota come *MoCap*. Conosciuta principalmente per le sue applicazioni nell'industria cinematografica, questa tecnologia ha potuto svilupparsi grazie alle ricerche nell'ambito dell'animazione, medico e militare. La *Motion Capture*, nel tempo, si è evoluta in modo esponenziale rendendo possibile la realizzazione di sistemi in grado di acquisire i dati del movimento e di trasferirli su un soggetto virtuale. Oggi, la *MoCap* continua ad innovarsi e i settori che la sfruttano sono sempre di più e molto diversi tra loro.

1.1 L'evoluzione della *Motion Capture*

La *Motion Capture* è la tecnica che si occupa della trasposizione del movimento. La procedura prevede una fase di acquisizione dei dati e una di trasposizione di questi. La *MoCap*, tramite diverse tipologie di sistemi, è in grado di codificare i dati del movimento di un soggetto reale e di trasferirli su un modello virtuale. Tecniche quali *Performance Capture*, *Hand Capture*, *Facial Capture*, che si occupano di trasferire movimenti specifici di un soggetto nel mondo virtuale, sono comprese nella definizione di *MoCap*.

La *Motion Capture* trova le sue origini a New York, nel 1918, quando Max Fleischer inventò il rotoscopio, un dispositivo che permette di ricalcare manualmente i fotogrammi di un film in *live-action*, utilizzandoli come linea guida per l'azione animata[1]. Nel 1919, Max e suo fratello Dave adoperarono per la prima volta la tecnica del *rotoscoping*¹ per animare la serie *Out of the Inkwell*[2]. Nel 1937, l'uscita del lungometraggio animato *Snow White and the Seven Dwarfs*, della Walt Disney, realizzato con il rotoscopio, segnò l'inizio di una nuova era dell'animazione.

¹Rotoscoping: tecnica che prevede l'utilizzo del rotoscopio.

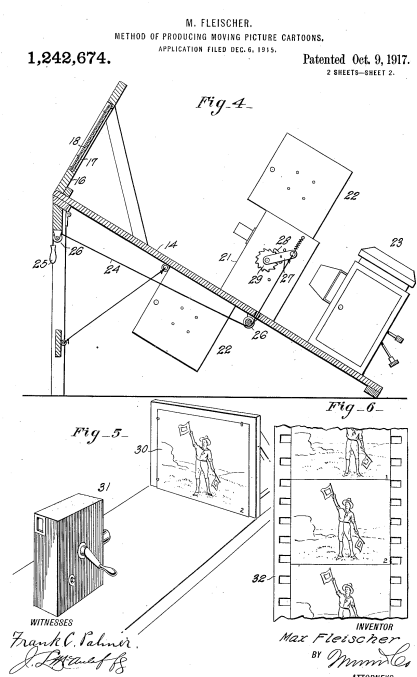
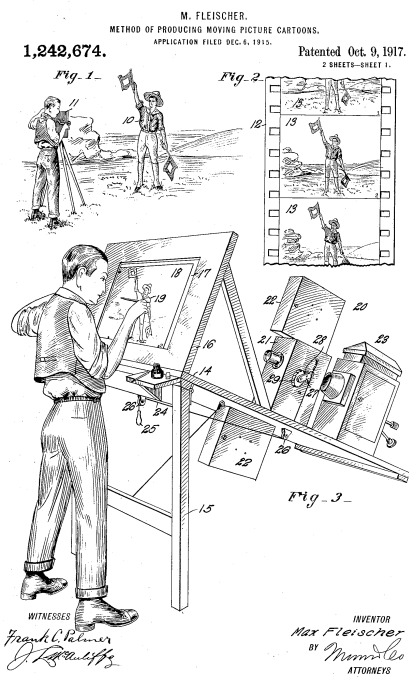


Figura 1.1: Rotoscopio

Fleisher, M. (Inventore). (1917, ottobre 9). *Method of producing moving picture cartoons* [immagini del brevetto]. Estratto da: patentimages.storage.googleapis.com.

Nel 1959, l'animatore Lee Harrison III creò la prima "Motion Capture Body Suit" una tuta realizzata con dei potenziometri² in grado di riprendere e animare il movimento di una persona in *real-time*[4].

Negli anni ottanta, Tom Calvert, professore di kinesiologia e informatica presso la Simon Fraser University, attaccò dei potenziometri al corpo umano e utilizzò i dati ottenuti per animare dei personaggi per scopi scientifici. In seguito, sperimentò l'utilizzo di esoscheletri per tracciare il movimento delle ginocchia[5].

Successivamente, il lavoro dei ricercatori del MIT portò alla nascita della *graphical marionette*. Il sistema consisteva nell'utilizzare due telecamere per rilevare la posizione di un insieme di LED attaccati ad un indumento indossato dal *performer*[6]. Questo

²Potenziometro: dispositivo realizzato da resistori la cui frequenza può essere modificata all'interno di un certo range[3].

dispositivo ha segnato la nascita della *MoCap* realizzata tramite sistemi ottici con *marker* attivi.

Grazie allo sviluppo di nuove tecnologie per la cattura e la trasposizione del movimento dal reale al virtuale, a partire dai primi anni 2000 l'industria cinematografica ha iniziato a vedere le nuove potenzialità della *Motion Capture*. Registi di calibro quali George Lucas, con *Star Wars: Episode 1-The Phantom Menace* (1999), Ridley Scott, con *Gladiator* (2000), Peter Jackson, con *The Lord of the Rings: the fellowship of the ring* (2001), Robert Zemeckis, con *The Polar Express* (2004) e James Cameron, con *Avatar* (2009), hanno favorito l'evoluzione della *MoCap* cercando tecniche innovative, sempre più all'avanguardia, per raggiungere il massimo realismo nei loro film.



Figura 1.2: The Lord of the Rings: The Fellowship of the Ring (2001)
Credits: CABIRIA Magazine. Estratto da: cabiriamagazine.it.



Figura 1.3: The Polar Express (2004)
Credits: pinterest e graphics8.nytimes.com. Estratto da: pinterest.com.



Figura 1.4: Avatar (2009)

Credits: Lukebeech.wordpress. Estratto da: lukebeech.wordpress.com.

Con il passare del tempo, le tecniche di trasferimento del movimento si sono evolute adattandosi alle tecnologie a disposizione nei diversi periodi storici. Oggi, la *Motion Capture* continua ad arricchirsi facendo propri i nuovi algoritmi di *computer vision* e *machine learning* per poter riprodurre il movimento sfruttando l'Intelligenza Artificiale.

1.1.1 I sistemi di acquisizione del movimento

La *Motion Capture*, sin dalle sue origini, si è sviluppata di pari passo con le tecnologie disponibili nei vari periodi storici. Infatti, i sistemi utilizzati per la sua realizzazione sono di vario genere.

Quando nacque, la *MoCap* era realizzata in modo completamente manuale tramite il roscopio, adesso, invece, il processo è quasi del tutto automatico grazie all'Intelligenza Artificiale. Ad oggi, i metodi per realizzare la *Motion Capture* sono numerosi e si differenziano per la tipologia di dispositivi che sfruttano e i loro costi, sia in termini di tempo sia economici. I sistemi per la cattura del movimento si suddividono in due macro categorie.

Sistemi *marker-based*

Questa tipologia di sistemi sfruttano dei marker, ovvero dei marcatori che hanno il compito di evidenziare determinati punti del soggetto di cui si vuole tracciare il movimento. I marker rappresentano dei punti di riferimento sia per la fase di codifica sia per quella di trasposizione del movimento.

- Sistemi acustici

Sono tecnologie che sfruttano le onde acustiche per determinare la posizione e l'orientamento di un soggetto nello spazio. Questi sistemi si avvalgono di emettitori e da recettori ultrasonici. I trasmettitori, piccoli dispositivi in grado di emettere frequenze caratteristiche ultrasoniche, sono posizionati sulle articolazioni del corpo in movimento. I ricevitori, solitamente tre, sono posizionati intorno all'area in cui si muove il soggetto. Per triangolare la posizione e l'orientamento dei trasmettitori nello spazio, questi emettono delle frequenze caratteristiche, diverse tra loro, in modo sequenziale. I ricevitori, in base al tempo che le frequenze impiegano per raggiungerli e alle eventuali distorsioni che presentano, riescono mapparli nello spazio.

Il vantaggio di questa tipologia di dispositivo è che sono molto economici e che non soffrono di interferenze da parte di oggetti metallici.

Questi sistemi, però, soffrono di interferenze, infatti, se c'è qualcosa che si frappone tra gli emettitori e i recettori oppure se vi sono altri ultrasuoni, i segnali vengono disturbati. Inoltre, poiché la velocità del suono dipende dal mezzo di propagazione, questa tecnologia è molto sensibile alle variazioni di temperatura, pressione e umidità all'ambiente in cui viene usata. Infine, i sistemi acustici limitano lo spazio d'azione del soggetto poiché non può superare l'area delimitata dai recettori.[7][8][9]

- Sistemi inerziali

Sono dispositivi basati sugli *IMU systems*, ovvero sull'unità di misura inerziale. L'IMU è un sensore composto da un accelerometro, un giroscopio e un magnetometro in grado di misurare, rispettivamente, l'accelerazione lineare (compresa l'accelerazione gravitazionale), la velocità angolare rotazionale e il campo magnetico globale[10]. I sensori IMU, nei dispositivi di *Motion Capture*, sono posizionati su una tuta indossata dal soggetto in movimento. Questa tecnologia permette ai

performers di muoversi liberamente nello spazio senza avere alcun vincolo fisico[11].

- Sistemi magnetici

Questi dispositivi utilizzano dei ricevitori, posti sul soggetto in movimento, e un trasmettitore di campo magnetico in posizione fissa. I sistemi magnetici calcolano la posizione e l'orientamento di ogni ricevitore rispetto al generatore di campo misurando le variazioni nello spazio. È il generatore del campo magnetico a definire l'orientamento degli assi nel sistema di riferimento. Rispetto ad altri sistemi sono abbastanza economici e, in condizioni ideali, ottengono risultati molto precisi. Il problema consiste nel fatto che è difficile utilizzare il sistema in condizioni ottimali, infatti i sistemi magnetici sono molto sensibili alle interferenze dovute a oggetti metallici, compresi i cavi elettrici presenti nei palazzi[7][8][9][11].

- Sistemi meccanici

Sono dei dispositivi che utilizzano una tecnologia meccanica. Consistono in strutture meccaniche, spesso esoscheletri, che permettono di tracciare il movimento. I sistemi meccanici sono solitamente composti da potenziometri e cursori posti sui giunti del soggetto in movimento per mapparne la posizione. Il vantaggio principale consiste nel fatto che i campi magnetici e le riflessioni acustiche non influenzano il risultato del tracciamento. Purtroppo, però, spesso consistono in strutture molto ingombranti che limitano la libertà di movimento del soggetto[7][8][9][11].

- Sistemi ottici

Sono dispositivi che si basano sulle informazioni visive. Solitamente si avvalgono di tute da far indossare al soggetto in movimento e da telecamere ad alta definizione. Le tute sono ricoperte da *markers*, attivi o passivi, posti in corrispondenza delle articolazioni del corpo. Le telecamere sono posizionate nell'area d'azione in modo da inquadrare il soggetto e i *marker* che si trovano sulla sua tuta. L'insieme dei dati raccolti durante la *performance* permette di identificare, grazie al processo di triangolazione, le coordinate spaziali di ciascun marker per quindi ricostruire i movimenti del soggetto. Questi dispositivi permettono di avere una grande libertà di movimento. Inoltre, i dati acquisiti dalla tecnologia sono tanto più precisi quanto è alto il numero di sensori posizionati sulla tuta. Purtroppo, però questi dispositivi, oltre ad essere molto costosi, soffrono sia per le interferenze luminose sia per le occlusioni dei sensori rispetto alle camere[7][8][9][11].

Come anticipato in precedenza, i sensori che vengono utilizzati possono essere attivi o passivi:

1. Attivi: sono dei *marker* emissivi, ovvero in grado di trasmettere luce propria. Solitamente sono luci a LED a infrarossi e, per distinguere i vari *marker* della tuta, possono essere di colori diversi o emettere luce a frequenze diverse .

2. Passivi: sono *marker* riflettori, infatti, essendo realizzati con materiali altamente riflettenti, sono in grado di riflettere la luce che gli arriva.

Sistemi *markerless*

Sono sistemi non intrusivi che non richiedono al soggetto di indossare alcun tipo di sensore. Sono basati sull'uso di telecamere e algoritmi di *computer vision*. Permettono al soggetto in movimento di essere libero, senza avere alcuna limitazione spaziale. Hanno, però, dei problemi nel catturare piccoli dettagli come movimenti minimi o di parti piccole del corpo come le dita. Le tecnologie *markerless* sono generalmente basate su sistemi ottici. Recentemente, grazie al crescente sviluppo dell'Intelligenza Artificiale e alla sua diffusione, sono stati sviluppati gli *AI-based Motion Capture Systems*.

1.1.2 *Workflow* generale per l'animazione tramite *Motion Capture*

Per poter utilizzare la *Motion Capture* e trasferire il movimento di un soggetto reale su un soggetto virtuale, si utilizza, come linea guida, una pipeline di lavoro. Il *workflow* mostrato di seguito spiega quali sono i principali passaggi per ottenere un buon risultato con la *MoCap*[12].

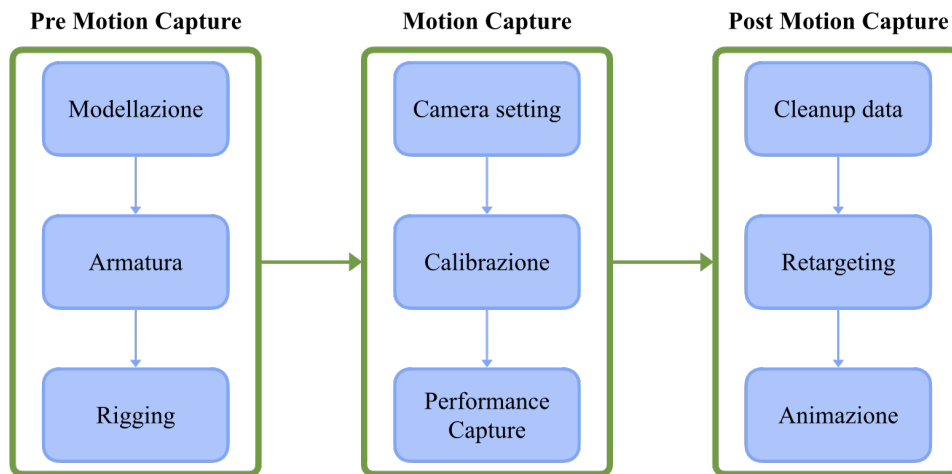


Figura 1.5: Workflow per la Motion Capture

1. **Modellazione:** fase in cui bisogna realizzare il modello tridimensionale del personaggio virtuale a cui trasferire il movimento.
2. **Armatura:** è necessario fornire al modello virtuale un'armatura che gli consenta di ricevere il movimento acquisito dal *performer*.
3. **Rigging:** questo passaggio permette di collegare ciascun componente dell'armatura ad una determinata parte del modello, proprio come le ossa del corpo umano sono collegate ai muscoli.
4. **Camera setting:** si impostano tutti i dispositivi di ripresa e vengono posizionati nello spazio secondo i requisiti della tipologia di dispositivo utilizzato.
5. **Calibrazione:** vengono calibrati i dispositivi in relazione al soggetto di cui devono catturare il movimento. Solitamente, in questa fase, il *performer* tenere alcune posizioni *standard* per pochi secondi in modo che le telecamere possano individuare tutti i *marker*. La precisione della calibrazione è fondamentale per ottenere un buon risultato finale. Si tenga conto che ogni dispositivo di cattura, in base alla tipologia di sistema usato e alla specifica tecnologia, ha un proprio processo di calibrazione.
6. **Performance Capture:** è il momento in cui vengono effettuate le riprese della *performance*. I dati acquisiti possono essere visualizzati in *real-time*, in base al *software* utilizzato e ai motori grafici di visualizzazione, oppure possono essere visualizzati al termine delle riprese.
7. **Cleanup data:** al termine delle riprese, è necessario pulire i dati raccolti. Infatti, è possibile che ci siano delle imprecisioni nella cattura del movimento.
8. **Retargeting:** è la procedura che permette di trasferire i dati di movimento acquisiti sul modello virtuale. Questa fase può essere fatta effettuando l'associazione manualmente oppure attraverso dei *software* o dei *plug-in* appositi.
9. **Animazione:** in questa fase è possibile visualizzare l'animazione sul modello virtuale. Spesso è necessaria un'ulteriore pulizia dei dati. È qui che si effettuano le correzioni manuali direttamente sul modello e che si aggiungono ulteriori *keyframe* per animare le componenti su cui non si è potuto trasporre le informazioni di movimento.

1.1.3 Ambiti di applicazione della *MoCap* ed esempi

Le potenzialità della *Motion Capture*, rese note principalmente dall'industria del cinema, hanno favorito il suo utilizzo in diversi settori. Infatti, la *MoCap*, sia da sola sia accostata ad altri sistemi, è ampiamente utilizzata anche in altri ambiti, tra cui quello

medico, militare, sportivo e videoludico.

Cinema

La *Motion Capture*, seppur utilizzata in molti settori, esprime in modo spettacolare le sue potenzialità nel mondo del cinema. Infatti, è proprio l'industria cinematografica che cerca di ottenere il massimo da questa tecnologia, facendo ricerca e mettendo in pratica idee innovative per realizzare prodotti sempre più realistici e tecnologicamente all'avanguardia.

Nel corso degli anni, Hollywood ha messo a frutto le potenzialità della *Motion Capture* facendola diventare una delle principali tecniche adoperate nel campo dei *Visual Effects*. Nei prodotti cinematografici, la *MoCap* è utilizzata per animare personaggi o oggetti virtuali a partire dal movimento di un attore reale. Se in passato la *MoCap* era utilizzata per rendere fluidi i movimenti dei personaggi disegnati in 2D, oggi è largamente usata per dare vita a personaggi in CG e renderli realistici. Il livello di dettaglio e fotorealismo raggiunti da questa tecnica permettono di far sembrare reali anche personaggi di pura fantasia.

Di seguito sono riportati alcuni esempi di prodotti realizzati con diversi tipi di *MoCap* in vari periodi storici.



Figura 1.6: Alice in Wonderland (1951)
Credits: Disney23. Estratto da: d23.com.



Figura 1.7: Pirates of Caribbean (2007)
Credits: Industrial Light and Magic. Estratto da: news.bbc.co.uk.

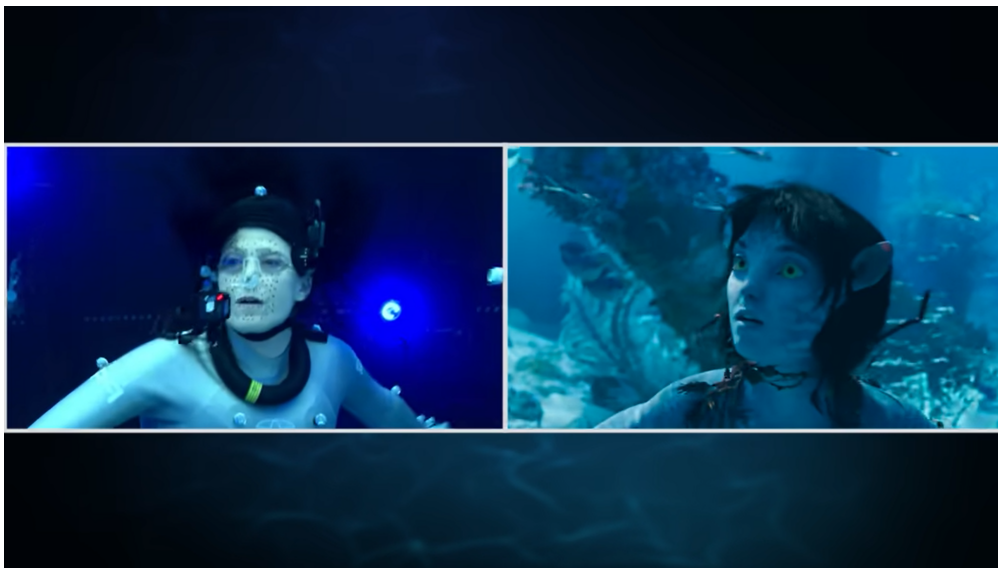


Figura 1.8: Avatar: The Way of Water (2022)
Credits: 20th century studios [immagine catturata dal video Avatar: The Way of Water — Sigourney Weaver]. Estratto da: www.20thcenturystudios.com.

Medicina

Il movimento rappresenta una delle principali funzioni del corpo umano. Da sempre, i disturbi del movimento costituiscono una sfida per la medicina. Il settore biomedico ha trovato, nella *Motion Capture*, un valido strumento per l'analisi del movimento, per la riabilitazione e per la previsione di disordini cognitivi e disturbi motori. Infatti, i dati del movimento acquisiti con le tecnologie della *MoCap*, principalmente tramite i sistemi ottici, inerziali e meccanici, sono di ausilio per lo studio di anomalie nei movimenti dei

pazienti per cercare di risalire alle cause dei disturbi. Inoltre, la *Motion Capture*, che permette di trasferire i dati cinetici su oggetti terzi come esoscheletri e arti robotici, abbinata alla robotica, consente di costruire strumenti finalizzati alla riabilitazione. Infine, l'analisi comparativa dei dati acquisiti supporta lo studio preventivo di disturbi motori degenerativi per poter trattare correttamente le malattie e somministrare cure preventive ai pazienti[13][14][15][16].



Figura 1.9: Esempio di task motori del protocollo sperimentale utilizzato, LAM Credits: Unità di medicina riabilitativa, Dipartimento di Neuroscienze/Riabilitazione, Ferrara. Estratto da: gpeम्मocap.wordpress.com.

Milizia

Così come le ricerche in ambito medico, anche le ricerche in campo militare portano avanti lo sviluppo di nuove tecnologie. Nel settore della difesa, la *Motion Capture* è spesso accostata alla Realtà Virtuale (VR) e alla Realtà Aumentata (AR).

Nel campo militare, la *MoCap* è solitamente impiegata per la riabilitazione dei soldati. Per esempio, «La Stanford Hall, nel Regno Unito, è probabilmente una delle strutture di riabilitazione più avanzate al mondo. Utilizzando la Motion Capture [...] i laboratori offrono ai ricercatori del Ministero della Difesa (MOD) alcuni degli strumenti più sofisticati e precisi disponibili per valutare, trattare e ricercare le principali condizioni e lesioni che colpiscono il personale militare in servizio»[17]. Inoltre, vi sono studi che mirano ad utilizzare la *MoCap* come strumento per il monitoraggio e la prevenzione di lesioni muscoloscheletriche[18]. Questo tipo di strumentazione è molto utile anche in fase di addestramento, per verificare che gli esercizi vengano effettuati correttamente senza rischi per il fisico.

Un ulteriore possibile impiego della *Motion Capture* è spiegato in un articolo del 2018[19] che propone di utilizzare la *Motion Capture*, unitamente alla *Mixed Reality*, per la realizzazione di robot per la difesa. L'idea è quella di sostituire i soldati con dei robot per evitare di perdere vite umane in guerra. Infatti, tramite la *Motion Capture*, i robot si muoverebbero esattamente come i militari che li manovrerebbero a distanza. Questa strategia potrebbe essere applicata per non far eseguire all'uomo i lavori pesanti e pericolosi.

Sport

Negli sport, la *Motion Capture* è molto usata come strumento ausiliare per l'allenamento. Infatti, molti atleti agonistici sfruttano questa tecnologia per catturare i dati del movimento effettuato durante un esercizio e analizzarli per poter migliorare le loro prestazioni sportive[20].

Solitamente, in questo ambito vengono sfruttati i sistemi inerziali, e talvolta ottici, in quanto lasciano una relativa libertà di movimento all'atleta, senza influire sulla *performance*, e perché rilevano i movimenti in modo veloce e abbastanza preciso.

Videogames

Così come nel mondo del cinema, anche l'industria dei *videogames* e dell'*entertainment* utilizzano la *Motion Capture* per realizzare prodotti realistici e spettacolari. La *MoCap*, infatti, può essere considerata una delle principali tecniche con cui vengono create le animazioni dei personaggi dei *videogames* al giorno d'oggi.

Tra i numerosi videogiochi presenti sul mercato, di seguito sono riportati due esempi, di giochi recenti, che hanno sfruttato la *MoCap* per coinvolgere il più possibile i giocatori nell'esperienza di gioco, rendendolo altamente verosimile.

Baldur's Gate 3



Figura 1.10: *Baldur's Gate 3*

Credits: NeoGamer - The Video Game Archive. Making of - *Baldur's Gate 3* [Behind the Scenes].
Estratto da: YouTube.

Il videogioco, sviluppato da Larian Studios e rilasciato il 3 agosto 2023, ha avuto un grande successo dovuto, in gran parte, alla *Motion Capture*. Infatti, questa tecnologia, oltre che per l'animazione dei protagonisti, è stata utilizzata per animare anche gli NPC tramite la cattura del movimento di 248 *performer*[21], ciò ha conferito al gioco vitalità e un realismo di altissimo livello[21].

Star Wars Jedi: Survivor



Figura 1.11: Star Wars Jedi: Survivor

Credits: EA Star Wars YouTube channel. Star Wars Jedi: Survivor - Trailer delle sessioni di coaching Jedi. Estratto da: YouTube.

Il videogioco, sviluppato da Respawn Entertainment e Respawn e rilasciato il 28 aprile 2023, sfrutta la *MoCap* per rendere fluidi e realistici i complessi movimenti eseguiti dai personaggi.

Capitolo 2

Stato dell'arte

2.1 Stato dell'arte

Il crescente sviluppo di algoritmi di *computer vision* e *machine learning* e il largo utilizzo di reti neurali ha permesso di realizzare sistemi in grado di imparare da loro stessi fino a ragionare autonomamente. Nell'ultimo anno, gli questi algoritmi che sono alla base dell'Intelligenza Artificiale, hanno iniziato ad affiancarsi a tecnologie preesistenti, portando innovazione e nuovi metodi procedure di utilizzo. L'IA è entrata a far parte anche del mondo della *Motion Capture* dando vita alla *AI-based MoCap*, altrettanto nota come *AI-powered MoCap* e *AI-driven MoCap*.

L'*AI-based Motion Capture* è una tecnica che permette di realizzare la *MoCap* tramite un processo quasi del tutto automatizzato, grazie all'Intelligenza Artificiale. Questa tecnologia è basta su sistemi *markerless*, infatti oltre alle telecamere necessarie per la ripresa della *performance*, non sfrutta alcun tipo di dispositivo fisico composto da sensori o *marker*. Infatti, gli algoritmi di *computer vision* e *machine learning*, a partire da un video, sono in grado di identificare i punti chiave del corpo umano senza che questi siano evidenziati[22]. Successivamente, grazie alle reti neurali e al loro allenamento, è possibile trasferire i dati acquisiti su dei modelli tridimensionali generando così dei modelli dinamici.

2.1.1 Applicazioni disponibili

Attualmente, i *software* che offrono sistemi di *AI-based Motion Capture* sono molti. Di seguito sono presentate solo le quattro applicazioni considerate idonee per l'animazione del cortometraggio "Poppy" e che sono state testate per la scelta del *software* finale da utilizzare per la realizzazione del progetto.

Si noti che, negli ultimi tempi, lo sviluppo di questi programmi ha subito una crescita esponenziale e che alcune delle funzionalità, ora disponibili, non erano ancora state

implementate quando si è effettuato lo studio dei *software* presenti sul mercato.

DeepMotion

DeepMotion si propone di portare in vita i personaggi animati tramite tecnologie innovative come l'*AI-powered Motion Capture* e il *real-time body tracking*. L'idea è quella di riuscire a catturare il movimento nel modo più realistico e fluido possibile, limitando l'uso di dispositivi di tracciamento pesanti e d'intralcio per i *performer*[23].

È possibile realizzare le animazioni direttamente dal sito DeepMotion. La piattaforma prevede che venga inserito in input un unico video e che vengano selezionate le impostazioni da utilizzare. DeepMotion permette di scegliere quale tipo di cattura fare, se di tutto il corpo, di parte e permette di attivare le opzioni di *Facial MoCap* ed *Hand MoCap*. Inoltre, è possibile caricare un modello proprio o di utilizzarne uno a disposizione dalla libreria.

Per informazioni dettagliate sull'utilizzo del *software* si rimanda alla documentazione di DeepMotion[24].

Move.ai

Move.ai è una delle società più emergenti nel settore dell'*AI Motion Capture*. Il suo obiettivo è quello di realizzare una tecnologia *markerless* che permetta di acquisire i dati del movimento a partire da un semplice video e di trasferirli su un modello tridimensionale. Inoltre, grazie al programma *Invisible*, di Move.ai, è possibile animare i modelli 3D in *real-time*. Il costo di questa tecnologia, a differenza dei sistemi ottici o inerziali, è bassissimo in quanto non vi sono *marker*, sensori o ulteriori dispositivi che non siano il *software* stesso. Anche i dispositivi di ripresa hanno un costo relativamente basso poiché vi è la possibilità di realizzare i video con i cellulari o con una qualunque telecamera HD[25]. I tempi di elaborazione sono brevi e i risultati precisi e fluidi.

Move.ai si avvale della piattaforma *web* chiamata *Move web app* e di un'applicazione per iPhone denominata *MoveAI*. Il *software* è molto versatile in quanto permette di scegliere il numero di attori da mettere in campo, da uno a tre, e dà indicazioni sul numero di dispositivi necessari per le riprese e la loro ideale collocazione nello spazio. Move.ai permette di acquisire il video della *performance* sia tramite l'applicazione *MoveAI* sia tramite telecamere standard in grado di riprendere in HD. Inoltre, è possibile scegliere se trasferire i dati acquisiti su uno dei modelli offerti dalla sua libreria oppure se caricare sulla *web app* un modello proprio. Nel secondo caso, è necessario fare il *retargeting* del modello sulla piattaforma in modo che il *software* possa rimappare i

dati correttamente[26].

È importante sottolineare che Move.ai è un *software* pensato per la cattura del movimento di esseri umani. Infatti, i modelli proposti dalla libreria hanno una fisionomia umana e i modelli che si possono caricare devono avere un *rig*, composto da 50 ossa, compatibile con lo scheletro di forma umana proposto dal *software*.

Per informazioni dettagliate sull'utilizzo del *software* si rimanda alla documentazione di Move.ai[26].

Rokoko Vision

Rokoko è una società danese specializzata in sistemi per la *Motion Capture*. Sebbene il prodotto di punta dell'azienda sia la tuta *Smartsuit Pro*, recentemente l'azienda ha sviluppato Rokoko Vision, un *tool* che sfrutta l'Intelligenza Artificiale per la cattura del movimento.

Il *software* di Rokoko Vision è *markerless* e acquisisce i dati del movimento del *performer* direttamente dal video, per poi rimapparli automaticamente su un modello tridimensionale per animarlo. Rokoko Vision dà la possibilità di caricare sulla piattaforma dei video già girati oppure di realizzarli direttamente tramite il sito. Inoltre, è possibile scegliere se usare una telecamera singola o se sfruttare l'opzione *dual camera*. Una volta che il *software* acquisisce i dati dai video e li elabora, è possibile visualizzare l'animazione su un modello 3D tramite Rokoko Studio. Quest'ultima è una piattaforma di *editing* che permette di modificare e aggiustare le animazioni.

Questa tecnologia, come la precedente, è economica rispetto ai tradizionali sistemi di *Motion Capture* in quanto non vi sono i costi di sensori, *marker*, tute o altri dispositivi. I tempi di elaborazione dei dati sono brevi e generano risultati molto precisi[27].

Per informazioni dettagliate sull'utilizzo del *software* si rimanda alla documentazione di Rokoko Vision[28].

Wonder Studio

Nella prima metà del 2023, l'AI *film company* Wonder Dynamics ha lanciato Wonder Studio, un'applicazione che utilizza l'Intelligenza artificiale e algoritmi di *machine learning* per la realizzazione della *Motion Capture*[29]. Questa tecnologia mira a semplificare, velocizzare e limitare i costi della *Motion Capture*, lasciando, però, agli artisti libertà di modifica e cambiamento. A differenza di altri *software*, Wonder Studio non si concentra solo sulla *performance* dell'attore, infatti la sua IA è in grado di animare,

illuminare e fare il *compositing* di personaggi in CG in una scena in *live action*[30].

Il *tool* è molto semplice da utilizzare, il processo è composto solo da tre step ovvero la realizzazione del video, l'assegnazione dei personaggi e l'impostazione dell'elaborazione dei dati. Per utilizzare il *software*, è sufficiente caricare il video della *performance* di cui si vuole tracciare il movimento e scegliere su quale personaggio 3D trasferire il movimento, se su uno preso dalla libreria o caricato dall'utente.

Per informazioni dettagliate sull'utilizzo del *software* si rimanda alla documentazione di Wonder Studio[31].

Si noti che tutte queste tecnologie sono in continua evoluzione. Lo stato dell'arte di ognuna delle applicazioni è cambiato notevolmente nel giro di pochi mesi. Infatti, dal momento in cui il presente progetto di tesi è iniziato, al momento in cui è stata realizzata la stesura dell'elaborato, le funzionalità e le caratteristiche dei quattro *software* si sono evolute e sono state aggiornate.

2.1.2 Esempi

Vi sono diversi modi di accostare la *Motion Capture* all'Intelligenza Artificiale. Esistono dei sistemi che usano l'IA come base per realizzare la *MoCap*, altri che utilizzano gli algoritmi di *computer vision* e *machine learning* per analizzare i movimenti catturati con i sistemi tradizionali e, altri ancora, che sfruttano i dati della cattura del movimento come *input* per far partire gli algoritmi di Intelligenza Artificiale. La presente tesi si concentra sulla prima tipologia di sistemi ma è interessante capire quali sono i sistemi sfruttati dai vari settori e a che punto sono arrivati con le loro ricerche e applicazioni.

Cinema

L'evoluzione dell'*AI MoCap* e, più in generale, dell'Intelligenza Artificiale è stata esponenziale, infatti, molte delle applicazioni sviluppate sono ancora in via di definizione. Pertanto, non sono ancora usciti prodotti cinematografici che sfruttano questa tecnologia.

Attualmente, i fratelli Anthony e Joe Russo, registi e produttori del lungometraggio *Electric State*, in uscita nel 2024, stanno utilizzando il software Wonder Studio, di Wonder Dynamics, per animare i personaggi del film con l'*AI-based Motion Capture*.

Medicina

Nell'ambito medico non sono ancora stati sfruttati sistemi di *AI-based Motion Capture*, però ai tradizionali sistemi di *MoCap* è stato abbinato l'utilizzo di algoritmi di Intelligenza Artificiale[32].

All'inizio del 2023, sono stati pubblicati due articoli scientifici che spiegano come l'Intelligenza Artificiale possa essere d'aiuto nel monitoraggio della progressione delle malattie e delle loro cure. Le ricerche vertono rispettivamente sulla distrofia di Duchenne[15] e sull'atassia di Friedreich[16], due malattie degenerative che, per cause diverse, nel tempo danneggiano il sistema motorio dei pazienti. Per la cattura del movimento sono stati utilizzati dispositivi indossabili dotati di sensori «per quantificare sistematicamente il comportamento del movimento nella sua complessità, utilizzando la cattura del movimento dell'intero corpo permettendo una piena libertà di movimento»[15]. Grazie alla cattura del movimento è stato possibile tracciare i movimenti dei pazienti nella loro quotidianità e ad intervalli di tempo costanti. Successivamente, sono stati utilizzati degli algoritmi di *machine learning* per valutare le differenze nei movimenti dei pazienti a distanza di tempo. In genere, la progressione di queste malattie e gli effetti delle cure si vedono in tempi molto lunghi. Grazie a questo approccio, che riesce a identificare le minime variazioni dei movimenti in finestre temporali relativamente brevi, è possibile predire la progressione della malattia e di monitorare il paziente durante la cura.

Si pensa che questa metodologia possa essere efficace per l'identificazione e il monitoraggio di altre malattie che portano alla degenerazione del sistema motorio.

Gaming

A giugno del 2023, la Electronic Arts Inc., una delle società leader nel settore dello sviluppo e della produzione di videogames, ha depositato il brevetto “*Enhanced animation generation based on video with local phase*”. Il sistema ideato permette di generare animazioni dinamiche a partire dal video di una persona in movimento. Una rete neurale, con il compito di analizzare il video, è in grado di acquisire i dati del movimento della persona. Grazie alle informazioni estrapolate e all'ausilio di una seconda rete neurale, è possibile generare delle animazioni in base ai fotogrammi dei video modificati[33].

Questo sistema potrebbe essere utilizzato per l'animazione dei personaggi in videogiochi futuri.

Capitolo 3

”Poppy”

3.1 Il progetto



Figura 3.1: Logo del progetto Poppy

“Poppy” è un cortometraggio di animazione, della durata di circa un minuto e mezzo, che, grazie al suo stile fotorealistico, mira a porre l’attenzione degli spettatori sulla bellezza della natura e mira a farli immergere nei magnifici paesaggi della storia narrata.

Il coinvolgimento del pubblico è l’elemento chiave in base al quale sono state prese le principali decisioni in fase di preproduzione. Infatti, la caratterizzazione dello stile, dai personaggi all’ambiente, la scelta del *mood*, dal *compositing* all’illuminazione delle scene, e la definizione delle tecnologie usate mirano a rendere il progetto realistico.

Grazie al motore di gioco, *Unreal Engine 5*, è stato possibile ricreare paesaggi naturalistici verosimili, ricchi di dettagli e con un alto livello di precisione. Inoltre, il realismo raggiunge livelli elevati grazie all’utilizzo della *Motion Capture* basata sull’Intelligenza Artificiale. Infatti, i movimenti dei personaggi sono fluidi e armoniosi in quanto derivano dalla cattura del movimento di un attore reale.

Il pubblico, quindi, è in grado di sentirsi coinvolto e di rendersi conto di giocare un ruolo fondamentale nella definizione di quello che sarà il nostro prossimo futuro. “Poppy” vuole rendere le persone consapevoli dell’importanza e dell’impatto che le loro azioni hanno, soprattutto nei confronti della natura.

3.2 La sinossi

Il cortometraggio “Poppy” è ambientato in un futuro distopico, post apocalittico, e narra un episodio della vita di Poppy, un robotto spaziale, che lavora per gli abitanti di un pianeta sconosciuto.

Nel deserto di un pianeta ignoto, brullo e desolato, si trova la *Earth Experience*, una struttura semisferica tecnologicamente avanzata. La cupola è progettata per offrire agli abitanti dell’arido pianeta l’opportunità di fuggire dalla loro triste realtà. L’installazione, grazie ai pannelli LED che ne ricoprono la superficie interna, mostra al pubblico i paesaggi naturalistici che caratterizzavano la Terra, prima che questa venisse degradata dall’inquinamento e dal cambiamento climatico derivanti dalle azioni distruttive degli esseri umani.



Figura 3.2: Earth Experience
Estratto dal progetto "Poppy"

Mentre Poppy passeggia all’interno della cupola, affascinato dalla perfezione della natura che lo circonda, decide di cogliere un fiore per ammirarlo da vicino. Il fiore, però, non è naturale, infatti è un oggetto realizzato con pellicole OLED¹ in grado di orientarsi nello spazio e raffigurare ciò che è previsto dall’installazione. Pertanto, la raccolta del fiore danneggia i circuiti e manda in tilt i sistemi del padiglione. Un uomo, con indosso una tuta da astronauta, responsabile della *Earth Experience*, vedendo la scena

¹OLED: *organic light-emitting diode*, è una tipologia di diodo organico composto da fogli in grado di emettere una luce diffusa[34].

da lontano, decide di rimproverare Poppy per aver rovinato la perfezione dell'esperienza simulata.

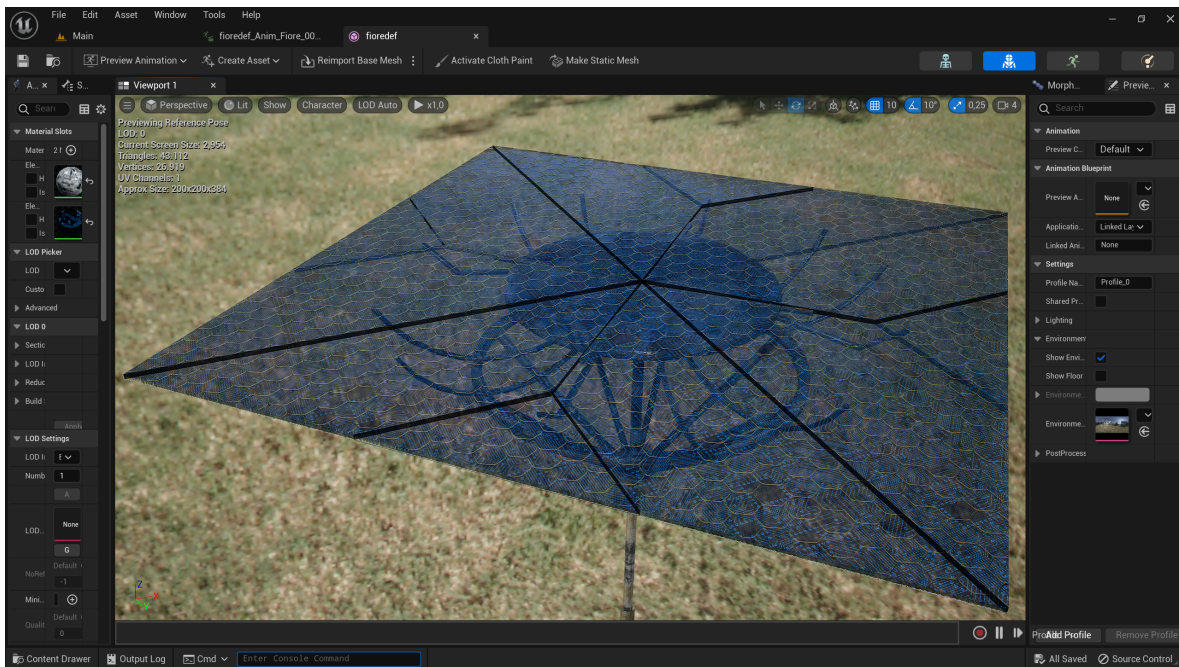


Figura 3.3: Fiore OLED aperto
Estratto dal progetto "Poppy"

3.3 I personaggi

"Poppy" mette in scena un breve episodio della vita dell'omonimo protagonista che lavora all'interno della *Earth Experience* sotto l'attenta supervisione degli abitanti del pianeta su cui si trova.

Poppy è un robottino spaziale dalle fattezze umanoidi e dotato di Intelligenza Artificiale. È un personaggio fortemente caratterizzato sia a livello fisico sia a livello caratteriale. Poppy, infatti, presenta molti dettagli, è visibilmente usurato, ricoperto di ruggine e presenta un'ammaccatura sul lato sinistro della fronte. Essendo dotato di Intelligenza Artificiale, Poppy conosce le cause che hanno portato alla rovina della Terra ma nonostante ciò, un giorno, prende coscienza della situazione, e resta sconvolto nel realizzare come la bellezza della natura terrestre sia stata distrutta.

In contrapposizione alla figura di Poppy, vi è l'astronauta. Questo è un personaggio poco caratterizzato, infatti il corpo e i tratti del volto sono coperti rispettivamente dalla tuta spaziale e dal casco. L'astronauta mostra un carattere aggressivo, infatti molto probabilmente è stato lui a procurare al robottino l'ammaccatura sulla fronte.



Figura 3.4: Poppy
Estratto dall'Animatic del progetto "Poppy"



Figura 3.5: Astronauta
Estratto dall'Animatic del progetto "Poppy"

L'uomo rivede le azioni del genere umano nei confronti della natura nel momento

in cui Poppy coglie il fiore. Quest'atto, seppur apparentemente innocuo, ricorda all'astronauta che l'insieme di tanti piccoli gesti come questo hanno fatto morire la Terra. Infatti, egli, mosso similmente dalla rabbia e dal rimorso, rimprovera severamente il robottino prendendolo a calci.

3.4 Gli obiettivi

Il cortometraggio vuole porre l'attenzione del pubblico sul futuro, infatti ha il duplice obiettivo di presentare, a livello narrativo, un potenziale futuro distopico al fine di sensibilizzare il pubblico e quello di presentare, a livello tecnico, le principali tecnologie che caratterizzeranno il campo dei *Visual Effects* nella produzione cinematografica.

In primo luogo, "Poppy" vuole muovere una critica al comportamento umano nei confronti della natura mirando alla sensibilizzazione del pubblico sulla tematica ambientale. La metafora che vede il protagonista cogliere un fiore artificiale e mandare involontariamente in tilt l'installazione vuole condannare le azioni umane nei confronti della Terra. Infatti, il corto mostra come piccole azioni, spesso considerate innocue nel quotidiano, possano avere pesanti ripercussioni sul pianeta. Poppy vuole spronare il pubblico ad avere un comportamento sostenibile per evitare di incorrere nel futuro distopico presentato, un futuro nel quale la bellezza della natura terrestre sarà solo un ricordo fruibile grazie ad esperienze simulate.

In secondo luogo, il progetto "Poppy" è stato realizzato tramite l'utilizzo della *Virtual Production* e dell'*AI - based Motion Capture*, due delle tecniche più all'avanguardia nel campo dei *Visual Effects*. Negli ultimi tempi, le due tecnologie, principalmente usate nelle grandi produzioni del cinema hollywoodiano, hanno iniziato a diffondersi velocemente grazie allo sviluppo di *Game Engine* sempre più potenti ed efficienti e alla crescente diffusione dell'Intelligenza Artificiale. Inoltre, grazie agli elevati livelli di qualità raggiunti e al rilevante abbattimento dei costi, queste due tecniche rappresentano il futuro delle produzioni cinematografiche, dai *colossal* alle piccole produzioni indipendenti.

3.5 Uno sguardo alla *Virtual Production*

La *Virtual Production* è una tecnica molto avanzata che, negli ultimi tempi, ha iniziato ad affermarsi nelle grandi produzioni cinematografiche. Tramite l'utilizzo di realtà virtuale, realtà aumentata e grafica real time direttamente sui set, la *Virtual Production* permette di fondere, già in fase di ripresa, il reale con il virtuale. Questo approccio permette di visualizzare immediatamente quello che sarà l'aspetto finale del prodotto.

Inoltre, nonostante un cospicuo investimento iniziale, consente un rilevante abbattimento dei costi sull'intero prodotto in quanto permette di rigirare subito la scena, senza dover aggiungere giorni di ripresa o dover apportare troppe modifiche in fase di post-produzione. Grazie all'utilizzo di *Game Engine* come *Unreal Engine 5*, adoperato in questo progetto, la *Virtual Production* ha raggiunto elevatissimi livelli di qualità a costi contenuti.

Capitolo 4

Software utilizzati

Per la realizzazione del progetto "Poppy" sono stati utilizzati diversi programmi, ognuno con uno scopo specifico. La prima parte di questo capitolo è dedicata alla presentazione dei *software* usati per la realizzazione del corto e alla loro funzione all'interno del progetto. La seconda parte è dedicata al programma utilizzato per realizzare la *Motion Capture*. La scelta è ricaduta sui *software* in grado di apportare un elevato livello di fotorealismo al prodotto finale.

4.1 *Software* per la realizzazione di "Poppy"

4.1.1 Blender

Blender è un *software* multipiattaforma, gratuito e *open-source* adatto alla realizzazione di diversi generi di prodotti multimediali. «È un programma di creazione 3D completamente integrato, che offre una vasta gamma di strumenti essenziali, includendo Modellazione, Rendering, Animazione & Rigging, Editing Video, VFX, Compositing, Texturing, e molti tipi di Simulazione»[35].

Nel progetto "Poppy", Blender è stato utilizzato per la modellazione 3D dei personaggi, per la creazione delle loro armature e per realizzare i *rigging*. Inoltre, sul programma sono stati importati i file .fbx per la correzione manuale delle animazioni realizzate con la *Motion Capture*. Infine, grazie all'*add-on*¹ "*Animation: Rokoko Vision Studio Live for Blender*", rilasciato da "Rokoko Vision", Blender è servito anche per il *retargeting* dell'armatura animata di Move.ai sul *rig* definitivo del robottino.

¹Add-on: sono degli *script* secondari che permettono di aggiungere determinate funzionalità a Blender[35].

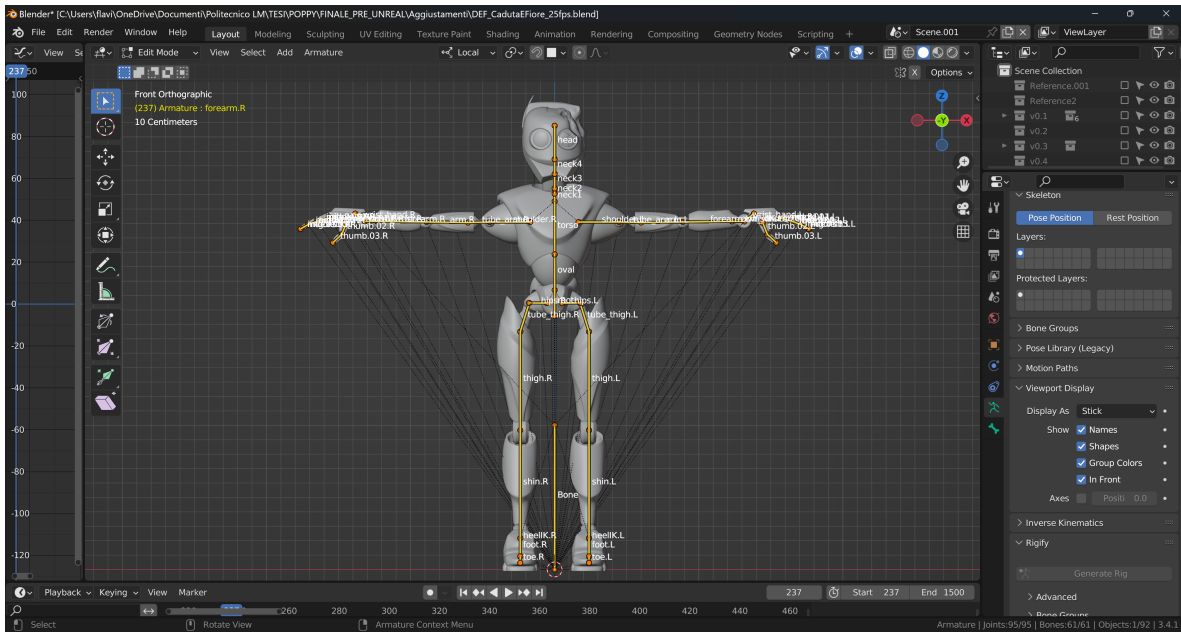


Figura 4.1: Poppy in Blender
Estratto dal progetto "Poppy"

4.1.2 Marvelous Designer

Marvelous Designer è un programma che permette di realizzare e modificare gli abiti-3D in modo semplice ed intuitivo. A rendere questo *software* uno degli standard di riferimento per il mondo della produzione cinematografica è la possibilità di simulare il comportamento dei vestiti in diverse situazioni, a seconda delle impostazioni fornite. Questa caratteristica permette di aumentare significativamente il livello di realismo nelle scene di animazione[36].

Nel progetto "Poppy", Marvelous Designer è stato usato per la creazione della tuta dell'astronauta. Infatti, a partire dal modello dell'uomo animato, è stata realizzata su misura la tuta spaziale. Grazie al simulatore che permette di ottenere il comportamento dell'abito in condizioni particolari, è stato possibile realizzare la tuta con tutte le caratteristiche necessarie affinché sembrasse che l'astronauta si trovasse su un pianeta diverso dalla Terra.

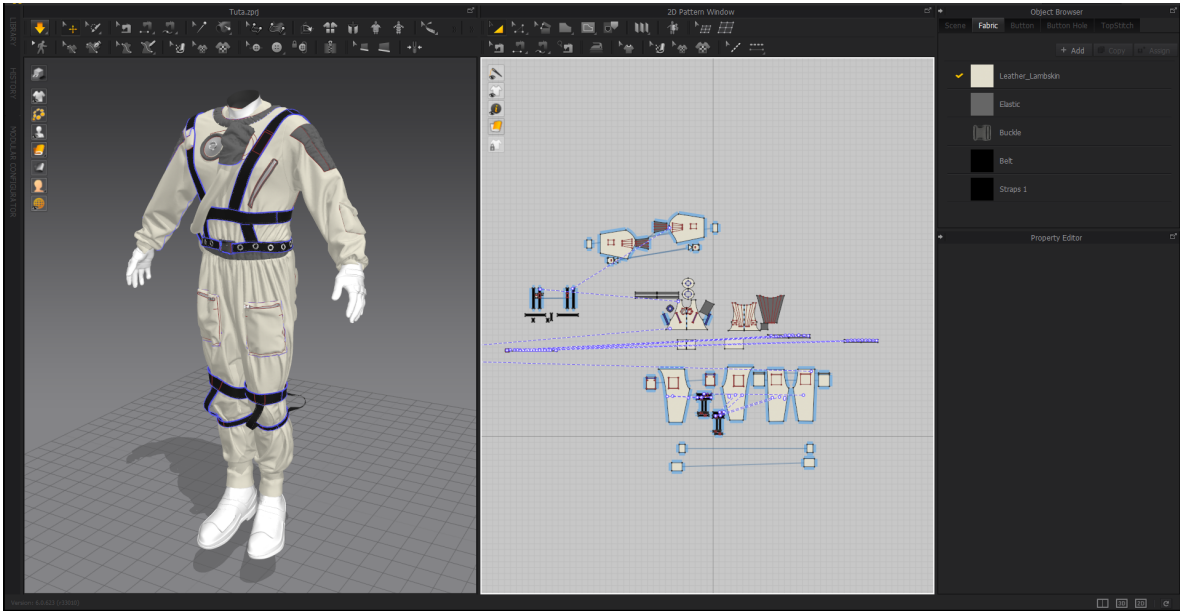


Figura 4.2: Astronauta in Marvelous Designer
Estratto dal progetto "Poppy"

4.1.3 Substance 3D Painter

Substance 3D Painter è un *software* che permette di realizzare *textures* altamente fotorealistiche. È possibile creare il disegno che si preferisce aggiungendo dettagli grazie agli strumenti a disposizione. È un programma molto versatile in quanto genera automaticamente le *UV maps* delle immagini realizzate, semplificando così la *pipeline* di lavoro. Negli ultimi anni, grazie alle sue *features*, agli strumenti che utilizza e al livello di dettaglio e realismo che raggiunge, Substance 3D Painter si è affermato come standard nell'industria cinematografica, videoludica e del design[37].

Nel progetto "Poppy", Substance 3D Painter è stato usato per la realizzazione delle *textures* di Poppy, del fiore OLED che il robotino prende in mano e degli accessori dell'astronauta, ovvero casco, guanti e stivali.

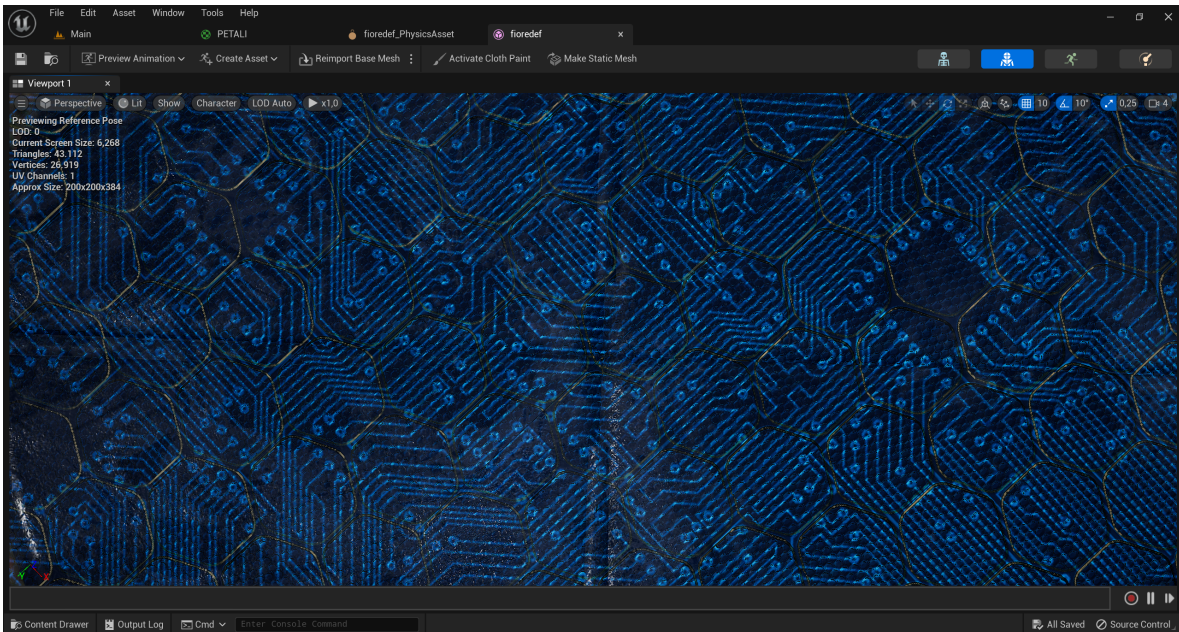


Figura 4.3: Texture del fiore OLED realizzata in Substance 3D Painter e visualizzata in Unreal Engine 5

Estratto dal progetto "Poppy"

4.1.4 Unreal Engine 5

Unreal Engine 5 è il motore grafico più recente rilasciato da "Epic Games". Esso permette di realizzare contenuti 3D *real-time* ed esperienze con un grado di libertà, affidabilità e flessibilità mai raggiunti precedentemente. Unreal Engine 5 è utilizzato in diversi settori dell'industria che vogliono avvicinarsi alle simulazioni in tempo reale ma gli ambiti di applicazione in cui è principalmente usato sono quelli del *gaming*, delle esperienze simulate e del cinema[38].

Nel progetto "Poppy", Unreal Engine 5 è stato utilizzato per creare l'ambientazione del cortometraggio. Infatti, tramite il motore di gioco, sono stati realizzati i diversi paesaggi in cui si vede camminare il protagonista. La distesa di girasoli, il deserto, le montagne innevate, la foresta tropicale, il campo di grano e la scogliera sul mare, visibili nei *post credits* del corto, sono state realizzate sfruttando gli *asset* a disposizione nel *software*. Inoltre, anche il pianeta sconosciuto su cui è ambientato l'episodio il campo di papaveri e la *Earth Experience* sono stati creati sfruttando dei modelli preesistenti. Successivamente, nel motore di gioco sono stati caricati i personaggi animati e sono state assegnate loro le rispettive *texture* esportate da Substance 3D Painter. Infine, su Unreal Engine 5 sono stati realizzati anche i movimenti di camera pensati in fase di riproduzione ed è stato esportato l'intero progetto.

4.2 *Software* per l'animazione tramite *AI Motion Capture*

Per la realizzazione dell'animazione dei personaggi di “Poppy” si è scelto di utilizzare l'*AI-based Motion Capture*. Per la cattura dei dati del movimento e la corretta trasposizione di questo sui modelli sono stati analizzati i *software* a disposizione e, in base ad alcuni criteri, è stato scelto il programma più consono alle esigenze del progetto.

4.2.1 La scelta del *software* da utilizzare

Per poter animare i personaggi tramite la *Motion Capture* basata sull'Intelligenza Artificiale, sono state individuate e analizzate nel dettaglio le applicazioni “Deepmotion”, “Move.ai”, “Rokoko Vision” e “Wonder Studio”, trattate nella sezione 2.1.1. A seguito di numerose prove effettuate su dei modelli di prova, all'interno di diversi *software*, è stato possibile mettere a confronto i risultati e scegliere il programma più adatto per la realizzazione del prodotto. Per poter avere dei prototipi da confrontare tra loro, è stata eseguita solo la fase di *Motion Capture* del *workflow* generale per l'utilizzo della *Motion Capture*, presentato nel capitolo precedente (1.1.2).

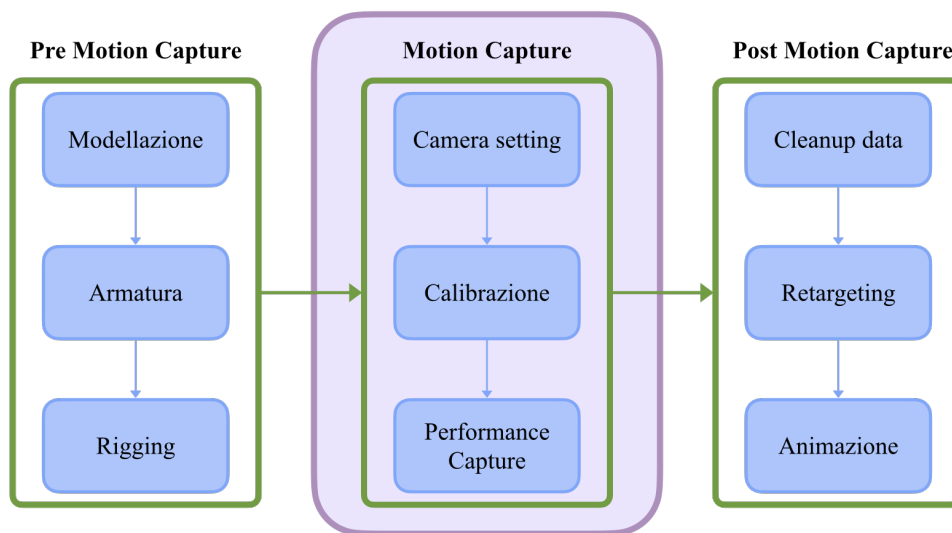


Figura 4.6: Passaggi del workflow seguiti per scegliere il software

Per ciascun'applicazione sono stati effettuati tutti i passaggi di calibrazione richiesti e, all'interno di ogni programma, è stato girato un video in cui l'attore eseguiva la medesima *performance*. I modelli su cui sono stati trasferiti i movimenti dell'attore erano

quelli di base forniti dai *software* stessi. Una volta ottenuto l'*output* dalle applicazioni, è stato possibile confrontare i risultati.

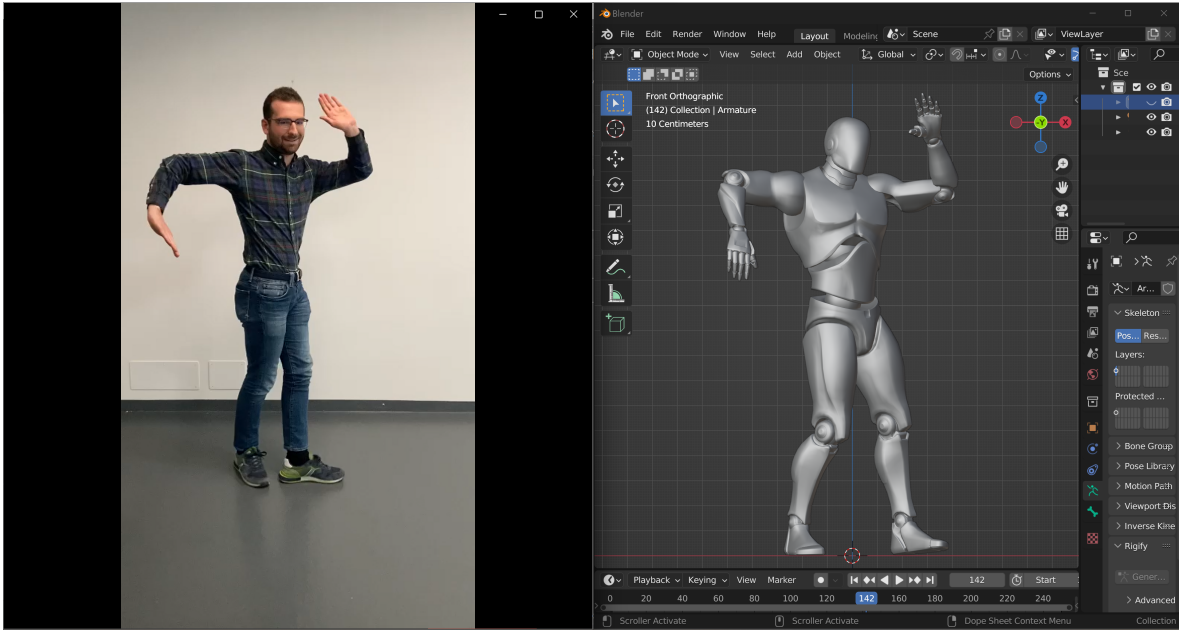


Figura 4.7: DeepMotion

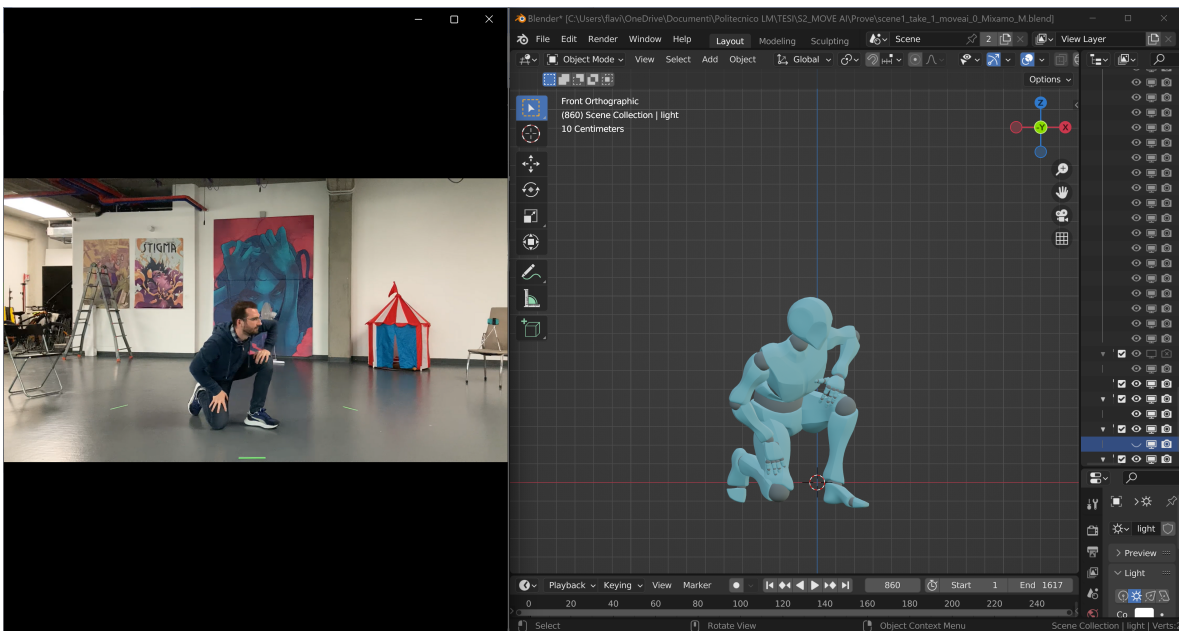


Figura 4.8: Move.ai

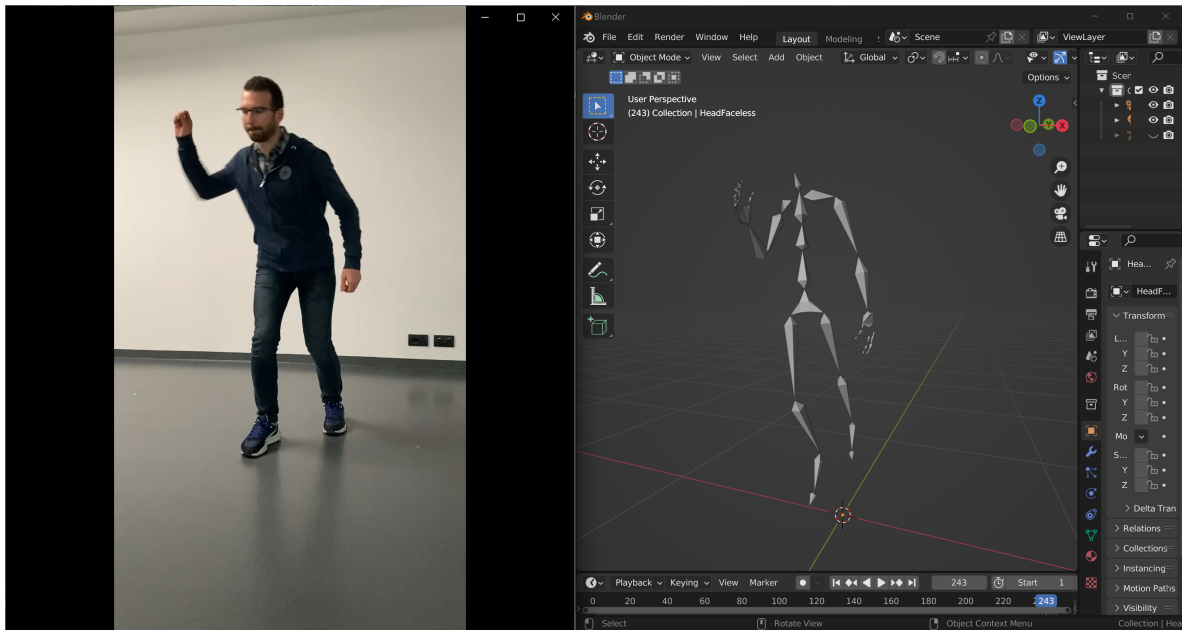


Figura 4.9: Rokoko Vision



Figura 4.10: Wonder Studio

Per poter analizzare i risultati in modo oggettivo, sono stati ideati tre parametri di giudizio: la fedeltà del movimento animato rispetto alla *performance* dell'attore, il tipo e la complessità dell'armatura richiesta per il modello, la libertà di movimento dell'attore.

- La fedeltà del movimento animato rispetto alla *performance* dell'attore è stato il primo parametro ad essere valutato in quanto l'obiettivo del progetto era quello di ottenere un cortometraggio fotorealistico. Infatti, è stata data molta importanza alla fedeltà con cui l'algoritmo è riuscito a tracciare i movimenti dell'attore sul modello tridimensionale. Inoltre, sono stati valutati con attenzione anche la fluidità e la naturalezza dei movimenti per poter ridurre al minimo l'intervento manuale sull'animazione alla fine del processo.
- Il tipo e la complessità dell'armatura richiesta dall'applicazione di *Motion Capture* sono parametri fondamentali nel processo decisivo. Per complessità dell'armatura si intende il numero di ossa e quindi il numero di giunti da cui questa è composta. Ogni giunto determina i gradi di libertà² che questo può avere per cui definisce il tipo di movimento che può effettuare ogni parte del modello. Le applicazioni di *Motion Capture* prese in analisi permettono di inserire dei modelli con armature non molto complesse. Inoltre, alcune applicazioni consentono di tracciare il movimento delle mani, e quindi prevedono un adeguato numero di ossa per animare queste, mentre altre preferiscono avere più ossa nella colonna vertebrale per rimappare i movimenti nel modo più fedele possibile. In questo contesto, si è preferito utilizzare un *software* che prevedesse la *Motion Capture* delle mani e che avesse la complessità dell'armatura minore.
- La libertà di movimento dell'attore è stato un altro importante elemento di valutazione. Infatti, nel cortometraggio si vede Poppy ammirare l'ambiente in cui si trova, esso cammina e si guarda intorno, libero di esplorare. La possibilità di far camminare l'attore libero nello spazio si può ottenere in due modi: facendo movimenti di camera in modo da seguirlo oppure tenendo la camera fissa ma considerando un'ampia area di ripresa.

Tendendo conto dei parametri sopra descritti, è stata fatta un'analisi delle animazioni realizzate dai quattro *software*. È stato possibile fare una prima selezione delle applicazioni migliori, infatti le animazioni che sono risultate più fluide e fedeli alla *performance* dell'attore sono state quelle realizzate da Move.ai e Wonder Studio.

I *software* selezionati, prevedono la *Hand Capture*, ovvero la cattura del movimento delle mani. Per questo motivo, entrambe le armature consentono di avere un adeguato numero di gradi di libertà delle mani e delle dita. Inoltre, il numero di ossa consentito dagli scheletri è molto simile, 50 ossa per Move.ai e 52 per Wonder Studio. Viste queste considerazioni, il metro di giudizio riguardante la tipologia e la complessità delle

²Gradi di libertà: è il numero delle variabili indipendenti necessarie per esprimere tutte le proprietà variabili di un sistema[39].

armature non ha evidenziato la preferibilità di un'applicazione rispetto all'altra.

A giocare il ruolo decisivo nella scelta del programma da usare, è stato il parametro della libertà di movimento dell'attore. Da un lato vi è Move.ai che consente solo l'utilizzo di riprese statiche, infatti bisogna riprendere il movimento dell'attore con un minimo di due camere. In base al numero di camere usate per girare la *performance*, il volume di spazio in cui l'attore può muoversi varia, maggiore è il numero di dispositivi usati per la ripresa, maggiore è lo spazio d'azione. D'altro canto vi è Wonder Studio che necessita di un solo punto di vista per la ripresa e consente di muovere la camera per poter seguire i movimenti del *performer*.

Per queste ragioni, in un primo momento è sembrato più conveniente utilizzare Wonder Studio ma, ad una seconda analisi, è stato rilevato un problema. È stato osservato che l'animazione della camminata presentava degli slittamenti in alcuni punti. Il *software*, in alcune fasi del ciclo di camminata, aveva rimappato il movimento della camera sul movimento dell'attore creando così uno slittamento del modello nello spazio. Il problema avrebbe potuto essere risolto tenendo la camera fissa o correggendo manualmente l'animazione in una delle fasi successive del *workflow*. La prima soluzione avrebbe dato poco spazio d'azione all'attore, vincolando la resa finale della *performance*, la seconda soluzione avrebbe allungato troppo i tempi di lavorazione.

È stato quindi deciso di lavorare con il programma Move.ai e di realizzare le riprese utilizzando quattro dispositivi in modo da avere un maggiore volume in cui il performer potesse muoversi liberamente.

4.2.2 *Workflow* per l'utilizzo di Move.ai per animare i personaggi

Una volta individuato il *software* da utilizzare per realizzare le animazioni in *AI-based Motion Capture*, è stato necessario individuare una *pipeline* di lavoro adatta all'applicazione, in grado di rispettarne le funzionalità e i requisiti. Quindi, a partire dal flusso di lavoro descritto nel capitolo precedente, 1.1.2 sono state apportate alcune modifiche in modo da adattare il processo di animazione a Move.ai.

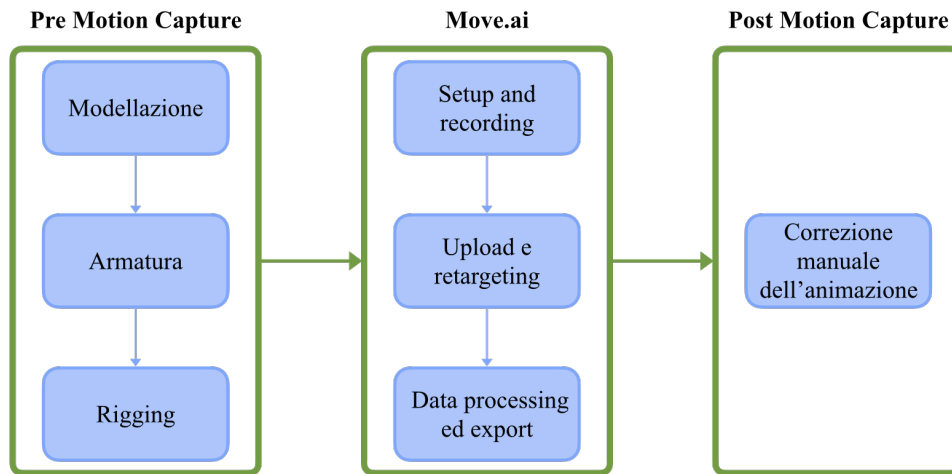


Figura 4.11: Workflow per l'animazione con Move.ai

1. Modellazione

Per poter utilizzare Move.ai è necessario avere un modello su cui rimappare il movimento del *performer*. L'applicazione dà due possibilità: quella di utilizzare uno dei modelli della propria libreria, messa a disposizione degli utenti, oppure, permette di caricare nel *software* un personaggio realizzato con qualunque piattaforma di modellazione 3D. Nel caso in cui si scelga di usare un proprio modello, è necessario dotarlo di un'armatura in modo da poter effettuare il *retargeting* di quest'ultima con un *rig* di riferimento fornito dall'applicazione.

Si noti che Move.ai è un *software* pensato per la *MoCap* di *human characters*, ovvero per il tracciamento del movimento umano da rimappare su modelli raffiguranti delle persone.

Per questo motivo, il modello tridimensionale caricato sull'applicazione può raffigurare qualsiasi cosa ma deve necessariamente possedere un'armatura che possa essere associata ad uno scheletro umano per poter riprodurre le animazioni fedelmente.

2. Armatura e *Rigging*

Nel caso in cui si scelga di utilizzare il proprio modello 3D, è necessario che questo sia dotato di un'armatura³ adeguata e di un *rigging*⁴ corretto. Per realizzare un modello che soddisfi le caratteristiche richieste da Move.ai, bisogna che l'armatura del personaggio possa essere rimappata, in fase di *retargeting*, sullo scheletro fornito dall'applicazione e che, quindi, sia dotata di 50 ossa. Una volta ultimato lo scheletro, il *rigging* e la nomenclatura corretta delle ossa, è possibile procedere con l'*export* del modello e del *rig*, in formato .fbx, per poterlo successivamente caricare su Move.ai.

3. Move.ai *setup* e *recording*

L'Intelligenza Artificiale di Move.ai riesce ad acquisire i dati del movimento direttamente dai video delle *performance* degli attori. È necessario che il *performer* venga ripreso da almeno due punti di vista differenti per far sì che i suoi movimenti vengano tracciati correttamente. Inizialmente, l'idea di Move.ai era quella di far realizzare agli utenti i video tramite l'applicazione "MoveAI", pensata per iPhone, per poi proseguire sulla "Move Web app". Di recente, la piattaforma ha messo a disposizione un *experimental mode*, ovvero una modalità tramite cui è possibile realizzare i video con dispositivi diversi dagli iPhone e di caricarli manualmente sulla *web app*. Questo metodo sperimentale dà la possibilità agli utenti di utilizzare qualunque dispositivo per riprendere la performance a patto che vengano soddisfatti determinati requisiti.

Qualunque sia il metodo di acquisizione dei dati scelto, è fondamentale che l'area d'azione del *performer* sia preparata come richiesto dall'applicazione e che i dispositivi di ripresa siano calibrati al meglio. La fase di calibrazione[26] è fondamentale in quanto l'obiettivo è quello di permettere al *software* di capire come sono posizionati i dispositivi di ripresa nello spazio, sia l'uno rispetto all'altro sia rispetto al *performer*. Bisogna effettuare una nuova calibrazione ogni volta in cui viene spostata una camera nell'ambiente. Per calibrare le telecamere bisogna seguire alcuni semplici passaggi:

- Far partire la registrazione su tutti i dispositivi.
- Fare in modo che l'attore si trovi al centro dello spazio d'azione.
- L'attore deve battere le mani tre volte al di sopra della propria testa nel modo più rumoroso possibile.

³Armatura: oggetto che rappresenta la struttura interna di un modello 3D. È un vero e proprio scheletro costituito da un numero variabile di elementi, chiamati ossa, interconnessi tra loro. Le ossa sono collegate tramite dei giunti che permettono loro di muoversi e di ruotare l'una rispetto all'altra.

⁴*Rigging*: è il processo che permette di associare un sistema di controllo, ovvero un armatura/*rig*, al modello 3D.

- L'attore deve restare per due secondi in *Y-pose*⁵ al centro dell'area di ripresa. Successivamente, rimanendo in *Y-pose* deve iniziare a camminare nell'area di ripresa nel seguente modo:
 1. Deve camminare in avanti, dal centro dell'area verso uno dei dispositivi, avendo cura di non uscire mai all'inquadratura di questo. Arrivato il più vicino possibile al dispositivo, deve fare una breve pausa per poi iniziare a tornare verso il centro dell'area restando rivolto verso la telecamera. La camminata deve essere il più naturale possibile.
 2. L'attore deve ripetere questa camminata verso ognuno dei dispositivi presenti avendo cura di tornare sempre al centro dell'area prima di andare verso il *device* successivo.
- Interrompere la registrazione sui dispositivi.

Quando si esegue la calibrazione è bene prendere alcuni accorgimenti:

- Annotare l'altezza dell'attore/degli attori al momento della ripresa tenendo in considerazione anche le calzature che indossano.
- Far sì che nello spazio d'azione siano presenti solo i *performer* e nessun altro. La presenza di altre persone non consentirebbe al *software* di lavorare in maniera ottimale.
- È utile che l'attore si muova nel modo più naturale possibile in tutto lo spazio a disposizione.
- Se anche solo una delle telecamere dovesse muoversi leggermente durante le riprese, è necessario effettuare una nuova calibrazione.

La procedura appena illustrata è stata trascritta dalla documentazione di Move.ai[26]. Dopo questa fase è possibile procedere con la ripresa della *performance*.

1. Far sì che il *performer* si trovi al centro dello spazio di ripresa.
2. Far partire la registrazione su tutti i dispositivi.
3. L'attore deve battere le mani tre volte al di sopra della propria testa nel modo più rumoroso possibile.

⁵*Y-pose*: è una posa di riferimento, usata principalmente nell'ambito dell'animazione, in cui il soggetto sta in piedi e, con le braccia alzate in aria, forma una Y. I gomiti devono essere piegati a 90°. Anche l'angolo tra le braccia e il busto deve essere di 90°.

4. L'attore deve mettersi in *T-pose*⁶ al centro del volume per almeno due secondi.
5. A questo punto è possibile iniziare la *performance*, l'attore può muoversi liberamente nel volume d'azione stabilito.
6. Interrompere la registrazione sui dispositivi.

Anche in questo caso, gli step da seguire per la registrazione sono stati trascritti dalla documentazione del *software*[26].

4. *Upload e retargeting* del personaggio

Per caricare il proprio modello 'riggato' su Move.ai è necessario che questo sia esportato dal *software* di modellazione in formato .fbx. Una volta caricato sulla *web app*, il modello deve essere 'retargettizzato'. Nella fase di *retargeting*, l'applicazione richiede l'intervento dell'utente che ha il compito di associare manualmente ogni osso del proprio modello a quello fornito da Move.ai. Terminato il processo, è tutto pronto per realizzare l'animazione del modello caricato a partire dai video girati.

5. *Data processing ed export*

Grazie agli step 3 e 4, sulla *web app* di Move.ai sono presenti tutti gli elementi necessari per far partire l'algoritmo ed elaborare i dati. In questa fase basta seguire le indicazioni fornite dalla piattaforma per impostare tutto ciò che serve per far partire la procedura. Non appena l'elaborazione dei dati termina, l'utente ha la possibilità di scaricare l'animazione in formato .fbx. Per la visualizzazione del risultato bisogna semplicemente importare il file scaricato in un programma in grado di aprirlo.

6. *Correzione manuale dell'animazione*

L'animazione tramite *Motion Capture*, grazie al supporto dell'Intelligenza Artificiale, è in grado di raggiungere risultati sorprendenti. Nonostante ciò, come avviene per ogni sistema di *MoCap*, anche in questo caso serve l'intervento manuale per correggere l'animazione ottenuta. Talvolta, è necessario l'intervento di un animatore per eliminare piccoli difetti o per integrare dei movimenti che sono stati persi durante l'elaborazione dei dati.

⁶*T-pose*: è una posa di riferimento, usata principalmente nell'ambito dell'animazione, in cui il soggetto sta in piedi e, con le braccia tese verso l'esterno, forma una T. Le braccia devono essere distese e l'angolo tra le braccia e il busto deve essere di 90°.

Si noti che la correzione manuale dell'animazione non è un processo standardizzato ma varia in base all'animazione, al *software* su cui vengono effettuati i ritocchi e dall'animatore incaricato di realizzare le modifiche. Tuttavia, grazie alla precisione raggiunta con l'AI, questa fase non richiede troppo tempo o risorse.

Capitolo 5

Workflow per l'animazione dell'Astronauta

L'animazione dell'astronauta è stata realizzata seguendo, come linee guida, sia il *workflow* generale per l'animazione tramite *Motion Capture* (descritta nella sezione 1.1.2) sia la *pipeline* di lavoro per l'utilizzo dell'applicazione Move.ai. Sulla base di questi due flussi di lavoro, si è proceduto tramite l'approccio *trial and error*, infatti, nel momento in cui sono stati riscontrati dei problemi, sono state ideate delle soluzioni e, in base alla fattibilità di queste in termini di costi e risorse, sono state applicate quelle più consone al progetto.

5.1 *Pipeline* di lavoro seguita

In questo paragrafo viene illustrata la *pipeline* di lavoro che ha portato alla realizzazione dell'animazione. Qui sono spiegate nel dettaglio e in ordine di esecuzione solo le fasi della procedura che hanno avuto successo e che, quindi, hanno permesso di portare a termine il processo di animazione dell'astronauta.

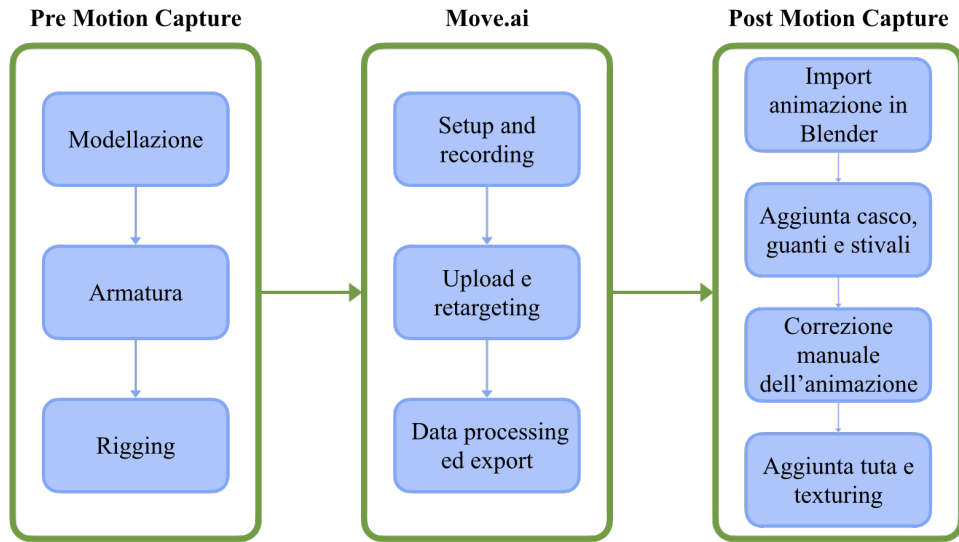


Figura 5.1: Workflow seguito per animare l’astronauta con Move.ai

1. Modellazione

L’astronauta è un essere umano i cui tratti sono nascosti dalla tuta spaziale che indossa. Il volto è coperto da un casco con la visiera oscurata mentre il corpo e gli arti sono rivestiti dalla tuta, dai guanti e dagli stivali.

Per la realizzazione della tuta spaziale è stato deciso di utilizzare Marvelous Designer che ha consentito di modellare la tuta direttamente sul modello animato che gli è stato fornito. Poiché il design dell’astronauta non prevedeva di mostrarne le caratteristiche fisiche se non gli abiti, per ottimizzare i tempi è stato deciso di non creare il modello dell’uomo ma di scaricarne uno preesistente. Questo modello, una volta animato, ha fatto da base per la modellazione degli abiti in Marvelous Designer.

È stato, quindi, scaricato dal sito “Free3D.com” [40] un modello 3D rappresentante un uomo. È stato scaricato il file .fbx del modello ed è stato importato in un nuovo progetto di Blender. Il modello è stato modificato eliminando le *mesh* che costituivano la testa.

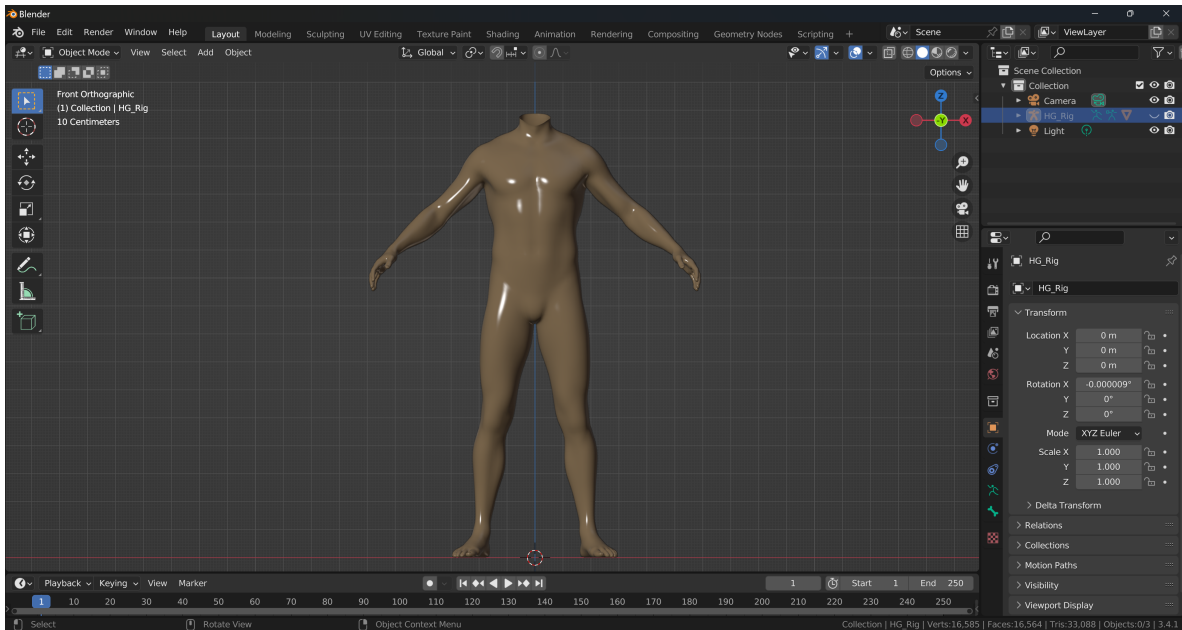


Figura 5.2: Modello Astronauta senza testa
Estratto dal progetto "Poppy"

Inoltre, dal sito "Sketchfab.com" [41] è stato scaricato il modello 3D di un'astronauta da cui sono stati esportati, separatamente, il casco, i guanti e gli stivali in formato .obj.

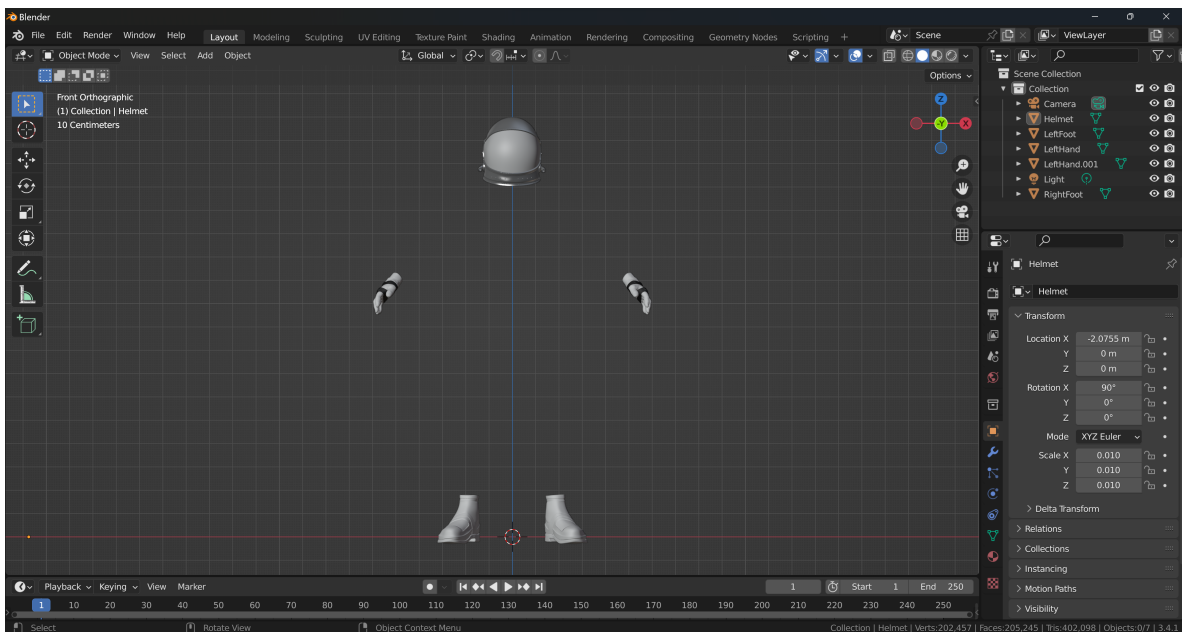


Figura 5.3: Accessori Astronauta
Estratto dal progetto "Poppy"

I tre file .obj sono stati importati nel file di Blender in cui era stato precedentemente importato il modello dell'uomo.

2. Armatura

L'armatura per il modello del corpo dell'uomo è stata realizzata manualmente. Per ottenere un buon risultato, sono state create le ossa solo su metà del corpo e, tramite l'opzione "*Armature & Symmetrize*" di Blender, il *software* ha generato automaticamente, in modo simmetrico, le componenti mancanti sull'altra metà del modello.

Lo scheletro è stato creato utilizzando 50 ossa, il numero previsto da Move.ai per la fase di *retargeting* (4.2.2). Inoltre, per agevolare il *retargeting*, ovvero l'associazione manuale tra le componenti del *rig* del modello con l'armatura d'esempio del *software*, è stata modificata la nomenclatura dello scheletro del corpo sulla base di quella indicata dalla *web app*.

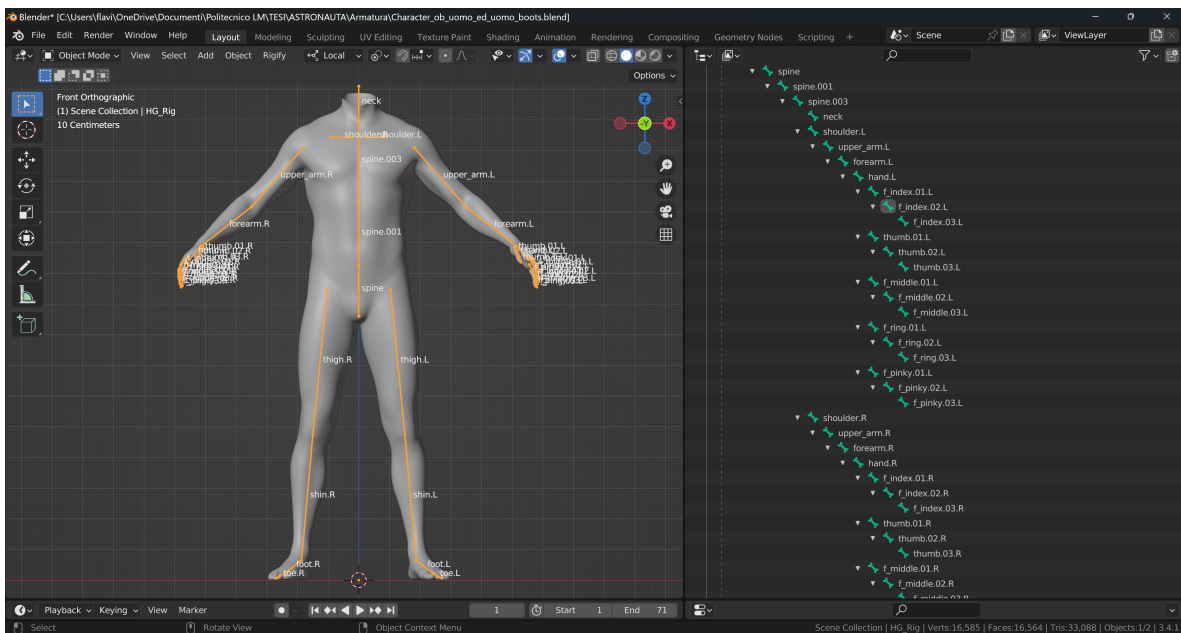


Figura 5.4: Armatura Astronauta
Estratto dal progetto "Poppy"

3. Rigging

Il *rigging* è stato realizzato usando l'opzione "*Set parent to Armature Deform With Automatic Weights*" di Blender. Questa funzione assegna automaticamente ai gruppi di vertici della *mesh* del modello, l'influenza che un particolare osso ha su di loro. Questa procedura è basata sull'algoritmo *bone-heat* che calcola la distanza tra i vertici del modello da ogni osso dell'armatura[35].

Con questa fase, è terminato il processo di preparazione del modello che è stato esportato, completo di *mesh* e *rig*, in formato *.fbx*.

Si noti che, nella schermata delle impostazioni di *export* del file, è stato necessario togliere la spunta alla voce “*Add leaf bones*”. Questa opzione aggiunge delle ossa di riferimento alle estremità dell’armatura. In questo caso, il *software* avrebbe aggiunto tredici ossa che, però, sarebbero risultate superflue in fase di animazione e avrebbero creato problemi per il *retargeting* sul *software*.

4. Move.ai

Terminati tutti i passaggi visti fin ora, si è proceduto con l’utilizzo di Move.ai per iniziare il processo di animazione vero e proprio. Partendo dalle impostazioni dell’applicazione e dal caricamento, sul programma, del materiale necessario per la trasposizione del movimento dal *performer* al modello virtuale, si è potuto procedere con l’elaborazione dei dati da rimappare.

- *Move.ai setup, calibration e recording*

Per poter utilizzare l’applicazione per iPhone “MoveAI” e la “Move web app”, è stato creato un account sul sito di Move.ai. Questo ha permesso di accedere ad uno spazio personale su cui sono stati caricati i modelli da animare e i video dai quali l’Intelligenza Artificiale ha ricavato i dati per tracciare il movimento.

Nel cortometraggio “Poppy”, l’astronauta è quasi sempre inquadrato da solo, per cui non è stata necessaria la presenza di un secondo attore. In base a ciò, è stato deciso il numero di dispositivi da utilizzare per la ripresa della *performance*, grazie al *Volume calculator*¹ messo a disposizione dal sito *web* di Move.ai. Sarebbe stato possibile utilizzare da due a sei telecamere ma, in base al numero di iPhone disponibili e ad alcune prove di ripresa effettuate, è stato deciso di utilizzare quattro dispositivi. Il *Volume calculator*, oltre a fornire il numero di *devices* ha anche indicato la disposizione ideale di questi nello spazio.

¹*Volume calculator*: consiste in un’interfaccia che permette all’utente di selezionare il numero di attori in campo e il numero di iPhone disponibili. Grazie ai dati inseriti, il calcolatore mostra il range di volume, minimo e massimo, all’interno del quale l’attore si può muovere e la disposizione ottimale dei *device*[42].

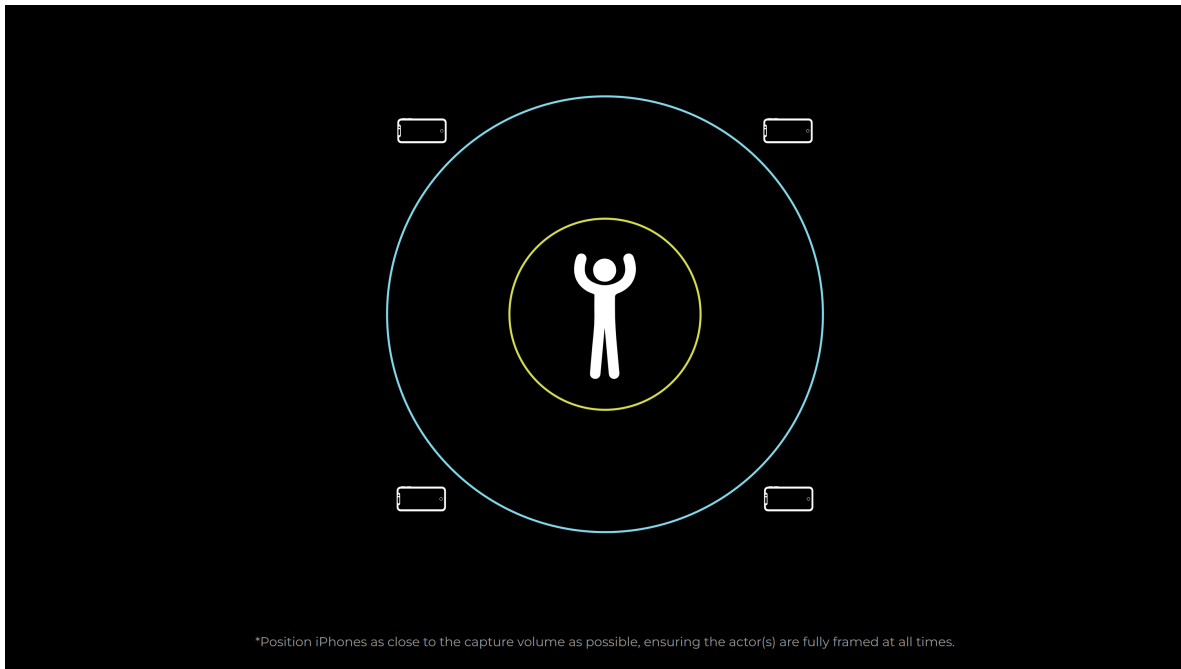


Figura 5.5: Volume Calculator di Move.ai

Credits: Move.ai Volume Calculator

Dopo aver stabilizzato gli iPhone su dei supporti e dopo averli disposti correttamente nello spazio, è stata circoscritta l'area di movimento dell'attore posizionando del nastro adesivo sul pavimento. Organizzato lo spazio, è stata effettuata la calibrazione dei dispositivi seguendo le specifiche della documentazione di Move.ai. Per ogni prova di ripresa è stata effettuata una nuova calibrazione. Successivamente, è stato girato il video della *performance* e sono state caricate le riprese sulla *web app* direttamente dall'applicazione MoveAI.

- *Upload e retargeting* del modello

In questa fase il file .fbx del modello dell'astronauta è stato caricato sulla "Move web app". Dopo il caricamento è stato effettuato il *retargeting* del *rig* del personaggio rispetto ad un'armatura base fornita direttamente da Move.ai. La procedura è stata semplice da seguire, infatti, il *software* permette di fare l'associazione tra il modello d'esempio e quello caricato semplicemente cliccando su un osso del primo e scegliendo, da una lista di nomi che compare, il nome del corrispettivo osso del secondo.

- *Data processing ed export*

Avendo il modello e i video a disposizione sulla piattaforma, è stato possibile

creare un progetto sulla "Move web app", sono stati impostati i dati richiesti, necessari per far partire l'algoritmo ed è stato indicato il modello su cui trasferire il movimento del *performer*. A questo punto, era tutto a disposizione sul software per far partire il processo di animazione.

Si noti che, prima di elaborare i dati, è stata attivata anche l'opzione di *hand capture* disponibile su Move.ai in modo da avere un tracciamento del movimento completo dell'attore, sia del corpo sia delle mani.

Terminata l'elaborazione dei dati, è possibile scaricare il file .fbx del modello animato direttamente dalla "Move web app".

5. *Import* dell'animazione in Blender

Terminato il processo di elaborazione dei dati della fase precedente, si è proceduto con il *download* del file .fbx dalla "Move web app". In seguito, è stato importato il file su Blender, dove è stato necessario scalare l'armatura del personaggio, in "*Edit mode*", per poterla visualizzare bene.

Dopo aver aggiustato lo scheletro e dopo aver eliminato gli oggetti superflui presenti in scena, è stato possibile visualizzare l'astronauta animato.

6. Aggiunta accessori

In questa fase sono stati aggiunti i guanti e gli stivali al corpo dell'uomo. Per prima cosa, tramite scalamenti, rotazioni e spostamenti, sono state sovrapposti gli stivali ai piedi del modello. Una volta ottenuto il posizionamento corretto, si è potuto procedere con il *rigging* degli stivali sullo scheletro dell'uomo.

Per il *rigging* sono stati seguiti due step:

- Le mesh degli stivali e l'armatura sono state imparentate tra loro tramite l'opzione "*Set parent to Armature Deform With Empty Groups*" di Blender. Questa funzione crea, sulla *mesh* degli stivali, dei gruppi di vertici vuoti, ovvero senza alcun peso assegnato. In questo modo è possibile associare manualmente i gruppi di vertici alle ossa da cui devono essere influenzati[35].
- Selezionando gli stivali e il modello umano, cliccando su F3, e selezionando, nel menu che appare, rispettivamente le voci "*Transfer mesh data*", "*Vertex groups*" e "*Source layer selected: all layers*" sono stati trasferiti i pesi del modello umano sui gruppi di vertici degli stivali creati precedentemente.

Alla fine di questa procedura, le scarpe seguivano correttamente il movimento dei piedi del personaggio.

Un procedimento analogo è stato eseguito per attaccare i guanti al modello. Infatti, per prima cosa, i guanti sono stati posizionati in corrispondenza delle *mesh* delle mani dell'uomo. Nonostante l'uso di rotazioni, scalamenti e spostamenti, la sovrapposizione non è risultata perfetta. Proprio per questo motivo, si è proceduto con la modifica del modello dei guanti tramite gli strumenti a disposizione nello “*Sculpt Mode*” di Blender. Una volta ottenuto un corretto accavallamento delle *mesh* è stato possibile procedere con il *rigging* dei guanti sull'armatura dell'uomo seguendo i due step sopra descritti utilizzati per gli stivali.

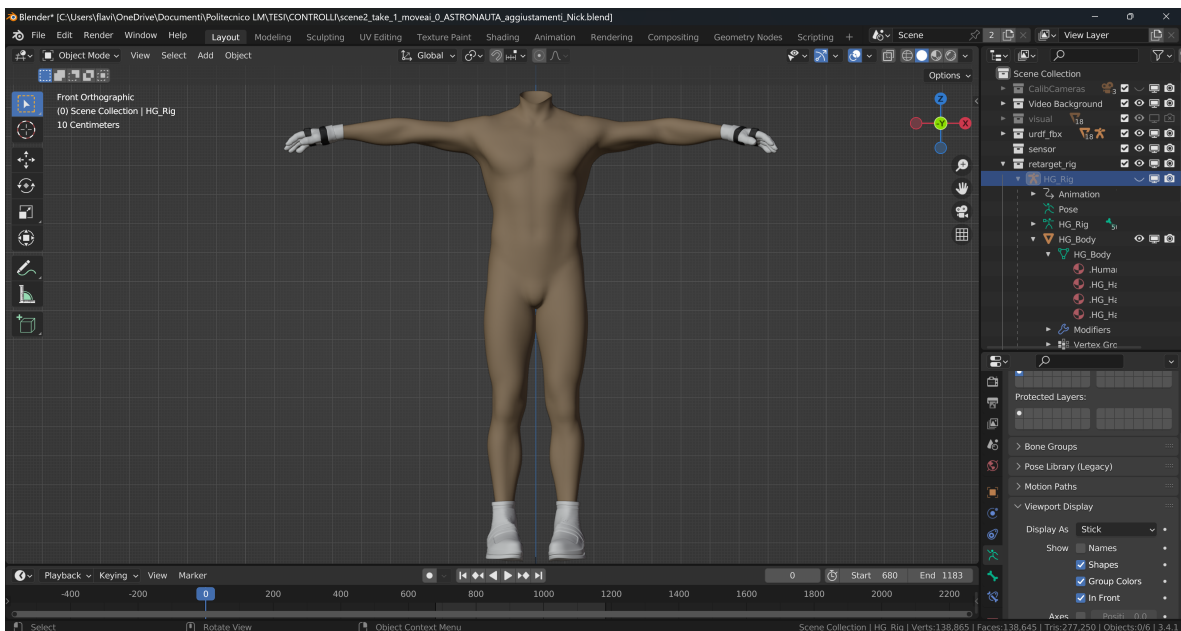


Figura 5.6: Astronauta con guanti e stivali
Estratto dal progetto “Poppy”

A questo punto, è stato aggiunto il casco al modello. Per prima cosa, il casco è stato posizionato in corrispondenza del collo del personaggio. Dopo aver selezionato l'osso del collo, nella modalità “*Edit mode*” di Blender, e, successivamente, il casco, nella modalità “*Object mode*”, questi sono stati imparentati tramite l'opzione “*Set parent to Bone*”. In questo modo, anche il casco segue i movimenti che compie l'armatura dell'astronauta.

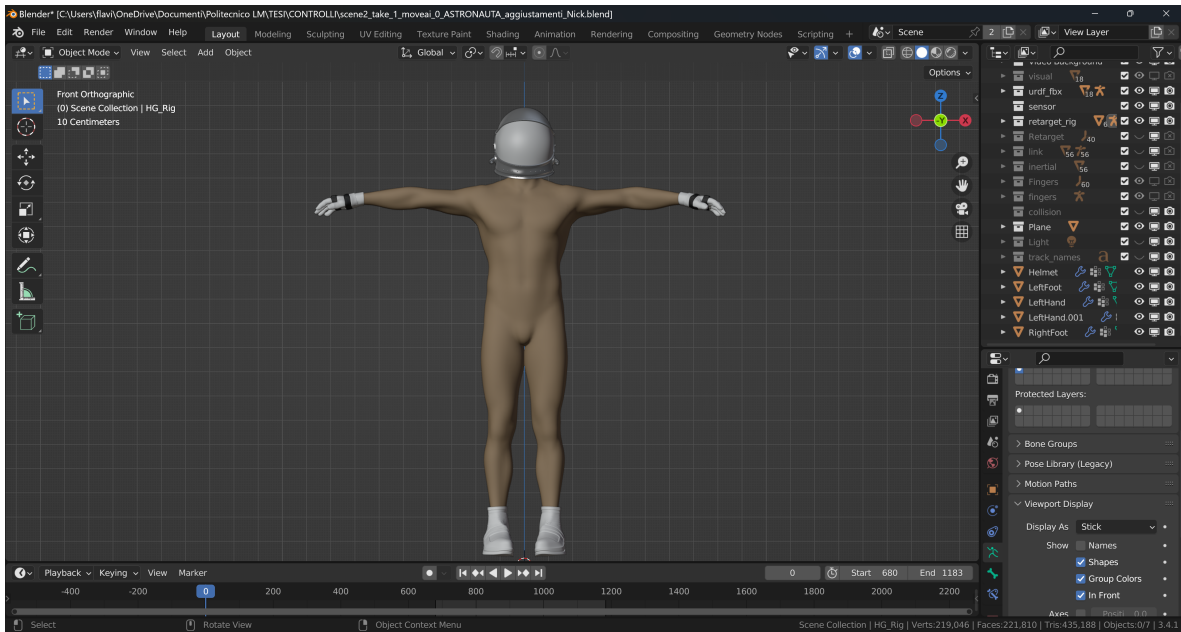


Figura 5.7: Astronauta con guanti, stivali e casco
Estratto dal progetto "Poppy"

7. Correzione manuale dell'animazione

L'animazione, benché fosse molto fedele ai movimenti fatti dal *performer*, aveva due difetti da aggiustare manualmente. Il primo consisteva nell'intersezione di alcuni punti delle *mesh* del personaggio, infatti, in certi momenti dell'animazione le braccia entravano all'interno del busto dell'astronauta. Tramite l'inserimento manuale di alcuni *keyframe* nell'animazione, sono state eliminate tutte le intersezioni errate. Il secondo difetto consisteva nei movimenti delle mani che, seppur molto simili al movimento originale del *performer*, avevano delle imprecisioni. Anche in questo caso, l'aggiunta di *keyframe* manuali ha risolto il problema.

8. Aggiunta tuta e *texturing*

Una volta terminata la fase di animazione, il modello è stato passato al "Reparto 3D" di Robin Studio. Qui è stata aggiunta la tuta spaziale al personaggio tramite il *software* Marvelous Designer che ha permesso la creazione di un indumento in grado di adattarsi ai movimenti dell'astronauta e alla gravità del pianeta su cui è ambientata la storia di "Poppy". Inoltre, per la realizzazione delle *texture* la scelta è ricaduta su Substance Painter, *software* che ha permesso di 'texturizzare' i modelli in stile fotorealistico aumentando, così, il realismo della scena.

5.2 Problemi e soluzioni

Durante il processo di animazione sono stati riscontrati diversi problemi. L'approccio seguito per la risoluzione di questi è stato il *trial and error* che, grazie alle modifiche apportate volta per volta, ha permesso di identificare le soluzioni ottimali.

Gli errori evidenti sin da subito sono stati corretti sul momento, prima che si procedesse con le fasi successive del *workflow* di animazione. Al contrario, alcuni problemi sono emersi solamente dopo l'elaborazione dei dati da parte di Move.ai. Infatti, il *software*, fornendo in *output* l'animazione, evidenziava delle imprecisioni che in altre fasi del processo (descritte nel capitolo 5.1) non si notavano. Importando i file dell'animazione in Blender e analizzando gli errori, ne sono state individuate le cause e si è proceduto con le correzioni.

Di seguito, sono illustrati i problemi e sono motivate le scelte prese durante il processo di lavorazione.

Gli accessori

I guanti, gli stivali ed il casco, essendo parte del modello 3D, avrebbero dovuto essere aggiunti nella fase di modellazione.

È stata fatta una prima prova eliminando i vertici delle *mesh* delle mani, dei piedi e della testa dell'uomo e sostituendole con i file .fbx di stivali e guanti e con il file .obj del casco. Siccome Move.ai permette il caricamento di un modello costituito da un unico *rig*, è stato necessario attaccare i modelli di corpo, guanti, stivali e casco ad un'unica armatura. In fase di *rigging*, però, è sorto un problema, infatti l'opzione "*Set parent to Armature Deform With Automatic Weights*" ha funzionato per il corpo ma non per gli accessori. La soluzione sarebbe stata quella di 'riggare' manualmente le ossa dei piedi delle mani e del collo rispettivamente agli stivali, ai guanti e al casco. Per effettuare questa procedura manualmente si sarebbe potuto utilizzare il "*Weight Painting*" unito all'associazione manuale dei gruppi di vertici alle ossa. Questo approccio sarebbe stato dispendioso a livello di tempo e sarebbe stato facilmente soggetto ad imprecisioni.

È stato fatto un secondo tentativo facendo coincidere, sempre in fase di modellazione, gli accessori alle corrispondenti parti dell'uomo. In fase di *rigging*, l'opzione "*Set parent to Armature Deform With Automatic Weights*" ha funzionato ma, dopo la fase di elaborazione dei dati, quando il modello è stato esportato dalla "Move web app", l'animazione aveva dei difetti. Infatti, la *mesh* delle mani dell'uomo fuoriusciva dal modello dei guanti, creando delle intersezioni errate ed indesiderate. Il problema avrebbe potuto essere risolto tramite la correzione manuale dei *keyframe* ma avrebbe allungato

molto i tempi di lavorazione.

La soluzione è stata trovata animando il modello umano nell'applicazione e aggiungendo, solo successivamente, gli accessori che hanno seguito correttamente l'armatura del personaggio (si veda il la sezione 5.1).

Il casco

Al contrario degli altri accessori dell'astronauta che sono stati esportati in .fbx, il casco è stato esportato in .obj poiché il file .fbx risultava errato.

Inizialmente sono state effettuate diverse prove per esportare il casco in .fbx ma in fase di *import* dell'*asset* nel file Blender, la *console* riportava un errore per cui il modello non poteva essere caricato. Facendo delle prove con i vari formati supportati da Blender, si è visto che l'.obj permetteva di importare correttamente il modello nel file, per cui è stato utilizzato questo.

Solo successivamente, è stato osservato che la causa dell'errore era dovuta ad un *UV mapping* del modello errato. Una soluzione avrebbe potuto essere quella di rifare l'*UV mapping* da zero, in modo corretto. In ogni caso, alla fine del *workflow*, sarebbe stato necessario ricontrollare gli *UV mapping* per rifare le *textures* di tutti gli accessori, per cui, la soluzione adottata ha fatto risparmiare del tempo all'inizio della *pipeline* di lavoro.

Rigging errato

In una delle animazioni realizzate, si notava che, durante la camminata, le gambe e le braccia dell'astronauta facevano dei movimenti incoerenti rispetto a quelli eseguiti dal *performer*.

Per capire quale fosse l'origine del problema, gli arti sono stati mossi manualmente nella "*Pose mode*" di Blender. Le rotazioni effettuate hanno portato alla deformazione della *mesh* e all'intersezione tra punti di vertici che non avrebbero dovuto verificarsi. Quindi, è stato controllato il "*Weight Painting*" dell'animazione per verificare la correttezza del *rigging*. Si è visto che il *rigging* di alcune ossa dell'armatura, in particolare quelle degli arti, avevano dei pesi sbagliati.

L'errore è stato risolto creando una nuova associazione di pesi tra le ossa dell'armatura e la *mesh* dell'uomo tramite la funzione "*Set parent to Armature Deform With Automatic Weights*" di Blender.

Video con orizzonte sbagliato

In uno dei file .fbx contenente l'animazione, si vedeva che il piano d'appoggio del personaggio era inclinato rispetto alla linea d'orizzonte. Il piano avrebbe potuto essere semplicemente ruotato ma, il vero problema è stato che il personaggio ha subito degli slittamenti sull'asse verticale.

Per capire quale fosse la causa della rotazione del piano d'appoggio, è stata rielaborata l'animazione utilizzando un personaggio virtuale della libreria di Move.ai. Anche in questo caso, l'errore era il medesimo. Si è proceduto, quindi, rifacendo la calibrazione dei dispositivi, girando nuovamente i video della *performance* ed rielaborando l'animazione sul personaggio dell'astronauta. In questo caso, il risultato ottenuto era buono, l'errore era nel girato. Infatti, analizzando i video si è visto che, durante la prima ripresa, uno dei *devices* era storto rispetto all'orizzonte, questo ha introdotto un errore nel tracciamento del movimento.

Per la realizzazione dei video successivi è stata posta una maggiore attenzione alla correttezza delle posizioni dei *devices*.

Piedi storti e sollevati da terra

In alcune delle animazioni realizzate con Move.ai si notava che l'astronauta camminava con i piedi divaricati verso l'esterno e che non toccava il piano del pavimento.

Analizzando i video delle riprese del *performer*, è stato notato che l'attore, nella fase di calibrazione dei dispositivi, teneva i piedi uniti e divaricati verso l'esterno. Questa calibrazione ha portato il *software* a prendere come posizione di riferimento dell'attore, quella in cui i suoi piedi sono storti, per cui la camminata è stata rimappata male sull'astronauta.

L'errore avrebbe potuto essere risolto in due modi. Il primo metodo era quello di correggere manualmente la rotazione dei piedi e per riportarli ad una posizione corretta. Questo approccio è stato provato ma, oltre a richiedere molto tempo, introduceva delle piccole imprecisioni nel resto della camminata, compromettendo l'intera animazione. L'alternativa scelta per correggere l'errore è stata quella girare nuovamente la scena prestando attenzione alla posizione tenuta dall'attore per ottenere una calibrazione esatta.

Per quanto riguarda la non tangenza tra i piedi dell'astronauta e il piano del pavimento, la causa è stata trovata nelle suole bianche delle scarpe indossate dall'attore. Infatti, i piedi, nell'animazione, risultavano sollevati da terra tanto quanto era alta la parte bianca delle scarpe.



Figura 5.8: Astronauta con piedi storti e sollevati da terra
Estratto dal progetto "Poppy"

Il problema avrebbe potuto essere risolto girando nuovamente il video della *performance* facendo togliere le scarpe all'attore o facendogliene indossare un paio di colore più scuro. L'alternativa scelta, più semplice e veloce, è stata quella traslare il personaggio con la relativa armatura leggermente più in basso, in modo che toccasse il pavimento. L'animazione ha funzionato bene senza che vi fosse alcuna introduzione di imprecisioni.

Import animazione in Blender

Nel momento in cui le prime animazioni realizzate con Move.ai sono state importate in Blender, si sono potute notare due particolarità nell'armatura del personaggio. La prima era che il numero di ossa dell'armatura ammontava a 63, la seconda è che l'armatura si presentava come un ammasso di ossa disposte senza logica e non come uno scheletro umano.

Per quanto riguarda il numero di ossa, si poteva notare che le 13 ossa in più si trovavano in corrispondenza delle 13 estremità dell'armatura del personaggio: le 10 dita delle mani, le due punte dei piedi e l'osso del collo.

Questo non ha rappresentato un vero e proprio problema in quanto non ha influito in alcun modo sull'animazione dell'astronauta, però avrebbe rappresentato una difficoltà per la fase successiva, la correzione manuale. L'inconveniente è stato risolto reimportando l'animazione in Blender eliminando la spunta alla voce "Add leaf bones" nella

finestra di *import*.

Per il problema dell'armatura deformata si è proceduto per tentativi. È stato necessario scalare, in “*Edit mode*”, l'armatura del personaggio. Inizialmente, lo scalamento è avvenuto senza impostare un fattore preciso ma dopo alcuni tentativi, l'opzione di scalamento migliore è risultata essere quella con un fattore pari ad 1/100. Questa operazione è stata effettuata per tutti i modelli animati importati in Blender da Move.ai.

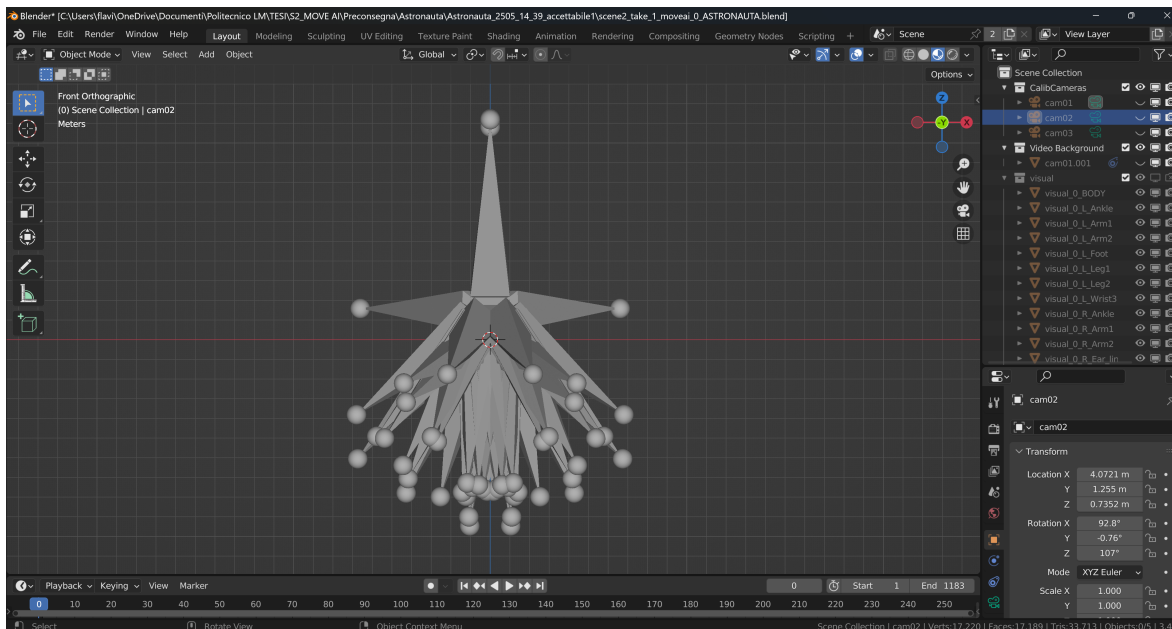


Figura 5.9: Stella di ossa
Estratto dal progetto “Poppy”

Si può notare che, nella maggior parte dei casi, si è preferito ripetere alcuni step del *workflow* di animazione piuttosto che ricorrere all'animazione manuale. Infatti, la correzione manuale, agendo su ogni *keyframe* dell'animazione, richiede un elevato livello di precisione ed una cospicua quantità di tempo.

5.3 *Workflow* per l'animazione di personaggi umani tramite Move.ai

In base al *workflow* che ha portato ad ottenere il prodotto finale, ai problemi riscontrati e alle soluzioni trovate, di seguito viene proposto un *workflow* ottimale per l'animazione di personaggi umani. Oltre ai passaggi da effettuare, sono elencati, per ciascuno step, gli aspetti a cui prestare particolare attenzione per ottimizzare i tempi in fase di lavorazione e per migliorare la resa finale del prodotto.

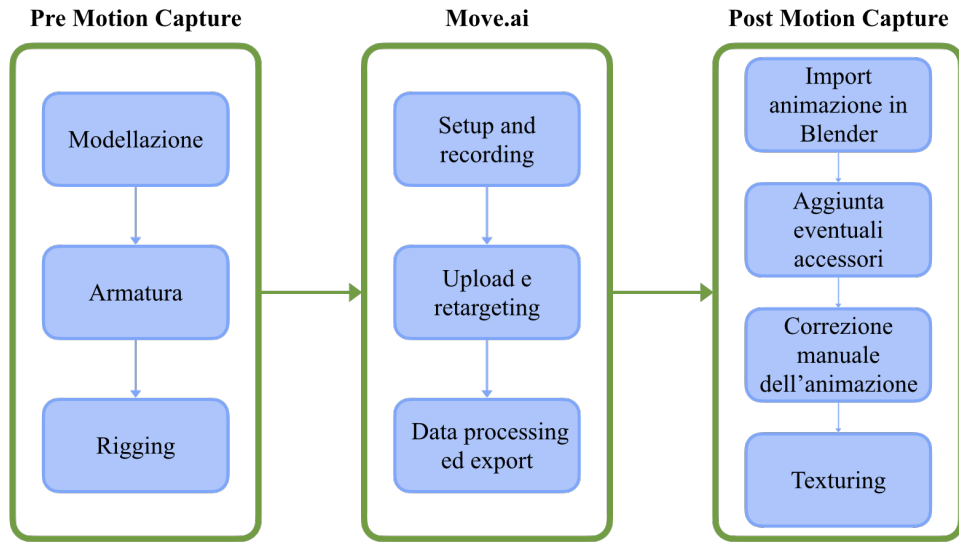


Figura 5.10: Workflow ottimale per l’animazione di personaggi umani con Move.ai

1. Modellazione

In questa fase bisogna creare il modello su cui fare la trasposizione del movimento. Nonostante vi sia la possibilità di animare un modello grezzo e di sostituirlo con quello definitivo solo in una fase successiva, è preferibile caricare sul *software* il modello definitivo per accorciare i tempi di lavorazione. In questa fase è bene assicurarsi che tutti i pezzi che comporranno il modello siano compatibili tra loro.

2. Armatura

Per creare un’armatura adatta al proprio modello si può partire da scheletri già completi come quello del “*human-meta rig*” fornito da Blender oppure se ne può creare una da zero. La scelta di uno dei due metodi è a discrezione del professionista incaricato di realizzare l’armatura, l’importante è che il *rig* definitivo sia composto esattamente da 50 ossa, come richiesto dal *software*. Anche la nomenclatura delle ossa non è vincolante in alcun modo per la resa definitiva del personaggio. Infatti, basta che la persona incaricata di eseguire il *retargeting* sia in grado di associare correttamente ogni osso a quelle dello scheletro d’esempio.

3. Rigging

Così come per la creazione dell'armatura, anche per il *rigging* si può scegliere di avvalersi di strumenti come il “*Set parent to Armature Deform With Automatic Weights*” di Blender oppure di realizzarlo da zero. Prima di esportare il file .fbx con il modello e il *rig* associato, conviene effettuare delle prove, muovendo tutte le ossa, per verificare che non si creino deformazioni nelle *mesh* e che, quindi, la fase di *rigging* sia stata completata con successo.

4. Move.ai

- *Move.ai setup, calibration e recording*
La calibrazione delle camere e delle persone è fondamentale per ottenere un buon risultato finale. In fase di calibrazione bisogna accertarsi che le camere siano dritte rispetto alla linea d'orizzonte e che siano disposte correttamente nello spazio. Più dispositivi si utilizzano per la ripresa della *performance*, più preciso è il risultato finale. In questa fase, l'attore deve eseguire correttamente tutti i passaggi richiesti dalla documentazione di Move.ai. È importante che il *performer* sia al centro dell'area ripresa dalle telecamere, che tenga le braccia perfettamente dritte e i piedi uniti e paralleli in modo che il *software* mappi correttamente tutti i movimenti. Inoltre, è bene che l'attore vesta con abiti piuttosto aderenti, di colori che lo evidenzino rispetto all'ambiente circostante, e che eviti il bianco.
- *Upload e retargeting* del modello
Per questa fase bisogna accertarsi che tutte le ossa del *rig* caricato siano mappate correttamente con quelle corrispondenti dell'armatura presentata da Move.ai.
- *Data processing ed export*
Prima di eseguire l'elaborazione dei dati, è bene controllare che tutte le impostazioni siano corrette. Se serve anche l'*hand capture* bisogna ricordarsi di attivare l'opzione prima di avviare il processo. Terminato l'elaborazione si può procedere al *download* del file .fbx generato.

5. Import dell'animazione in Blender

Durante l'*import* del file .fbx in Blender, bisogna accertarsi di togliere la spunta alla voce “*Add leaf bones*” in modo da non doverlo fare in una fase successiva. In questa fase conviene importare in Blender tutto ciò che contiene il file, comprese le camere in quanto possono rivelarsi dei buoni punti di riferimento rispetto all'animazione.

6. Aggiunta accessori

Nel caso in cui non si riuscisse ad attaccare correttamente gli accessori al modello di base nella fase 5.3, è questo il momento di farlo. Il collegamento degli accessori alla *mesh* di base può avvenire nel modo con cui ci si trova meglio. La procedura spiegata nel punto 5.1 è uno dei metodi possibili per eseguire questo step.

7. Correzione manuale dell'animazione

Se portati a termine con successo tutti gli step precedenti, questa fase dovrebbe risultare molto breve. La correzione manuale può consistere in diverse operazioni, da una semplice pulizia del movimento all'eliminazione dell'intersezioni tra mesh, dall'eliminazione di alcuni *keyframe* all'aggiunta o alla modifica di altri. Anche in questo caso, la scelta del metodo con correggere le animazioni dipende dal professionista che le effettua.

8. *Texturing*

In questa fase si aggiungono le *textures* al modello. Le *textures*, seppur attaccate al modello solo in questo step, possono essere realizzate anche prima. Infatti, se il modello del personaggio è quello definitivo, parallelamente all'animazione, è possibile realizzare le *textures*.

Capitolo 6

Workflow per l'animazione di Poppy

Inizialmente, in fase di preproduzione, Poppy non era stato pensato per essere animato con l'*AI-based MoCap* bensì tramite animazione manuale, un *keyframe* alla volta. Dopo un'attenta analisi di Move.ai e grazie alla struttura antropomorfa di Poppy, si è deciso di animare anche il protagonista con l'*AI-based MoCap*.

Per realizzare l'animazione è stato seguito un *workflow* analogo a quello usato per l'astronauta, adattandolo alle necessità di Poppy e cercando di mettere in pratica tutti gli accorgimenti illustrati nel capitolo 5.2.

6.1 *Pipeline* di lavoro seguita

Questo paragrafo è dedicato all'illustrazione dei passi della procedura seguita per realizzare l'animazione di Poppy.

Per animare il robot, è stato necessario lavorare contemporaneamente su due modelli costituiti dalla stessa *mesh* ma da armature diverse. Questo approccio ha consentito di utilizzare Move.ai per animare il personaggio dalle fattezze umanoidi nonostante esso necessitasse, a causa della struttura fisica, un numero di ossa superiore a quello consentito dall'applicazione.

Poppy, essendo il protagonista, è presente per quasi tutta la durata del cortometraggio. Per l'animazione sono stati realizzati due progetti distinti, uno in cui il robot cammina, usato per la scena dei *post-credits*, e un secondo in cui il robot si guarda intorno e raccoglie il fiore.

Di seguito, sono spiegate nel dettaglio e in ordine di esecuzione solo le fasi della procedura che sono andate a buon fine e che, quindi, hanno contribuito alla realizza-

zione dell’animazione del robot. La procedura spiegata è quella utilizzata per entrambi i progetti.

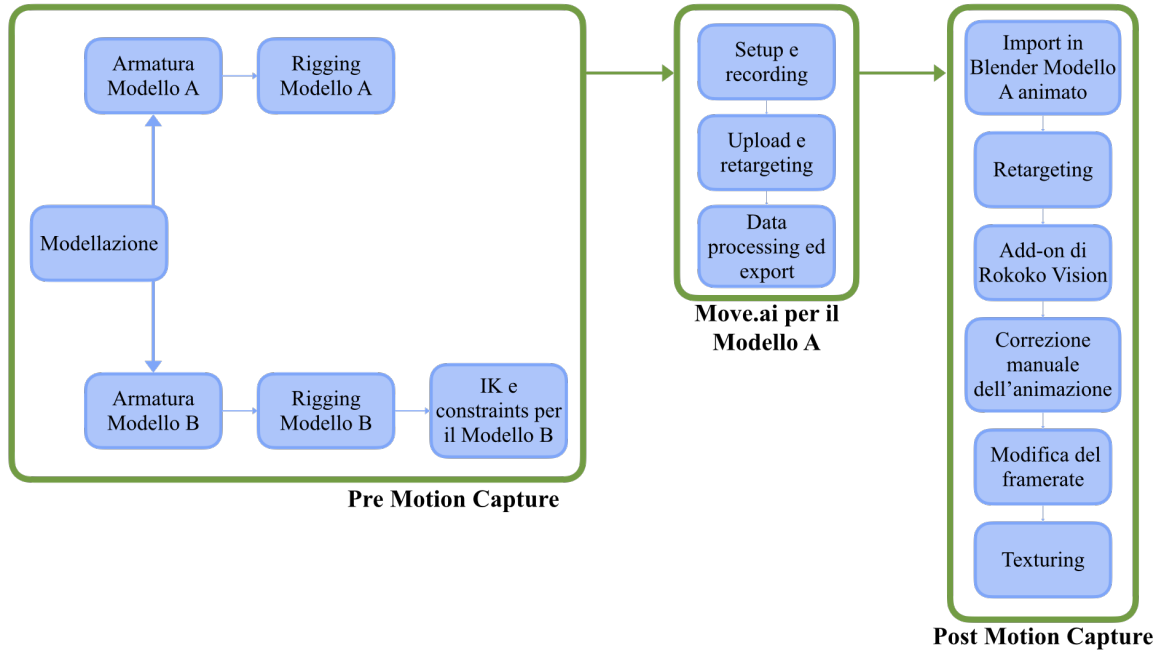


Figura 6.1: Workflow seguito per animare Poppy con Move.ai

1. Modellazione

Poppy presenta una struttura fisica molto simile a quella umana, almeno esteriormente. Essendo un robot, però, è composto da pezzi meccanici e da perni che consentono loro di ruotare gli uni rispetto agli altri.

Il “Reparto 3D” di Robin Studio si è occupato della modellazione del personaggio in Blender, seguendo le indicazioni fornite dal “*Character design sheet*” realizzato in riproduzione.

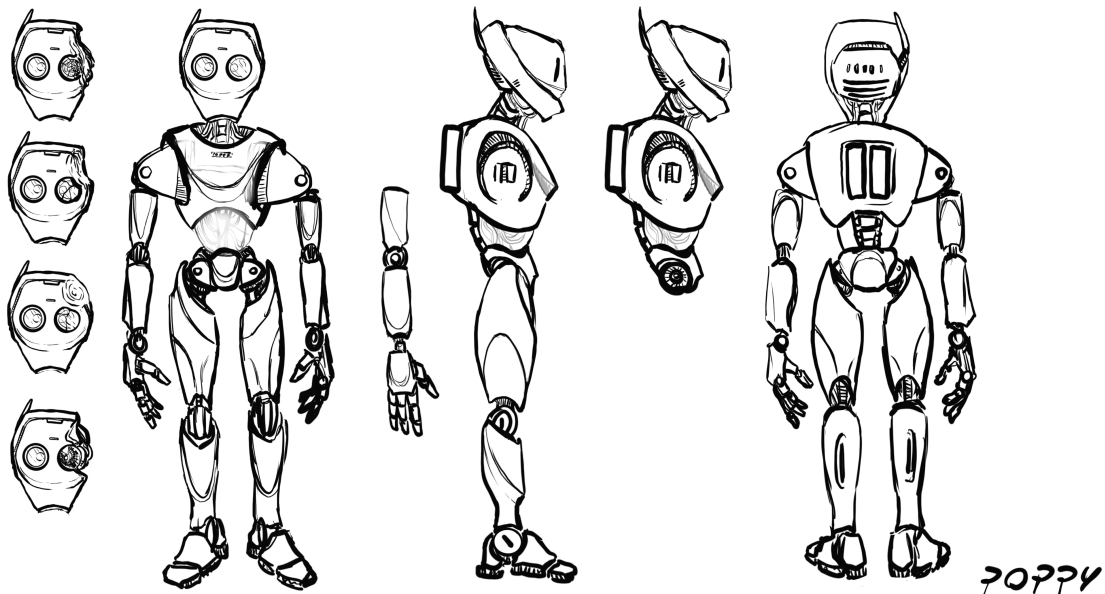


Figura 6.2: Model sheet del personaggio Poppy
Estratto dal progetto "Poppy"

Ultimato il modello, sono state create due copie di questo, una da caricare e animare su Move.ai (d'ora in poi indicata come "Modello A"), e un'altra destinata al *retargeting* e alle aggiunte manuali per ottenere l'animazione finale (d'ora in poi indicata come "Modello B").

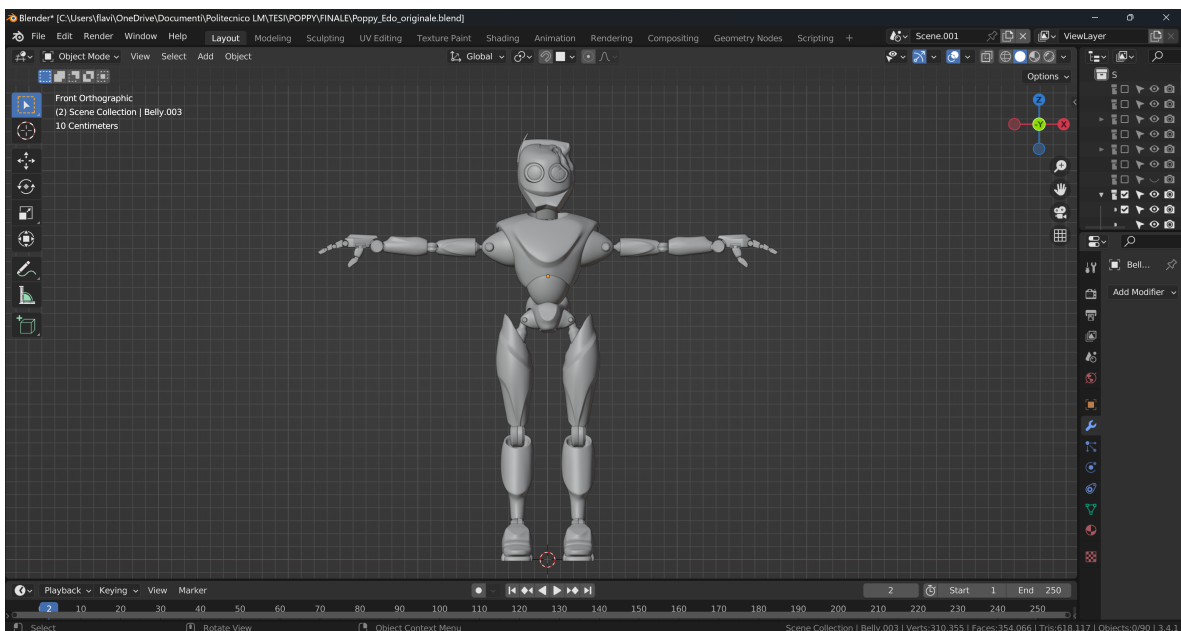


Figura 6.3: Modello 3D di Poppy
Estratto dal progetto "Poppy"

2. Armatura

- Modello A

Per questo modello, l'armatura è stata realizzata manualmente. Per ottenere un risultato preciso, le ossa sono state create solo su metà del corpo e, grazie all'opzione “*Armature > Symmetrize*” di Blender, il *software* ha generato automaticamente, in modo simmetrico, le componenti mancanti sull'altra metà del modello. L'armatura è stata realizzata con 50 ossa, il numero richiesto da Move.ai per poter caricare il modello e fare il *retargeting*. Per agevolare il processo di *retargeting*, la nomenclatura delle componenti dello scheletro è stata fatta sulla base di quella usata dal *software*.

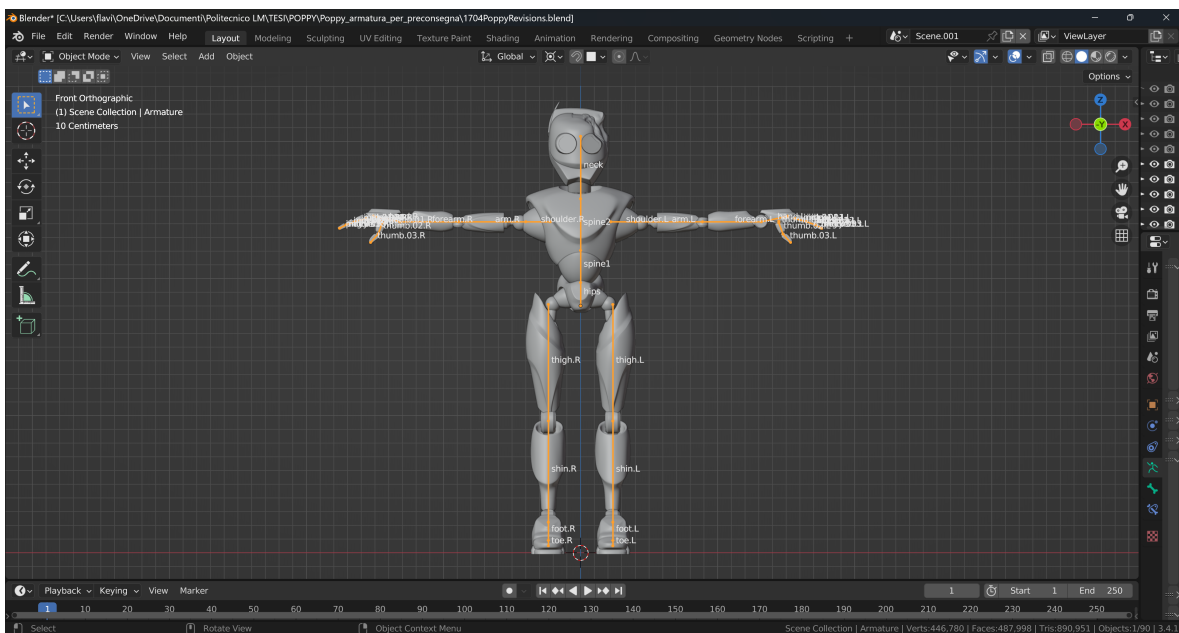


Figura 6.4: Armatura Modello A di Poppy
Estratto dal progetto “Poppy”

- Modello B

Per il Modello B l'armatura è stata realizzata manualmente. In questo caso, sono state utilizzate 56 ossa per poter ricreare il movimento dell'attore in modo fedele, rispettando, però, tutti i vincoli e i gradi di libertà dei giunti del robot. Per ottenere un'armatura precisa, anche qui sono state create le ossa solo su metà del corpo e, tramite l'opzione “*Armature > Symmetrize*” di Blender, il *software* ha generato quelle mancanti. Per facilitare il *retargeting* finale tra le due armature dei modelli A e B, le ossa di questo scheletro sono state nominate in modo riconoscibile.

Si noti che, in una delle fasi successive, descritta nel punto 6.1, sono state aggiunte altre 5 ossa all'armatura per migliorare il *retargeting* finale del personaggio.

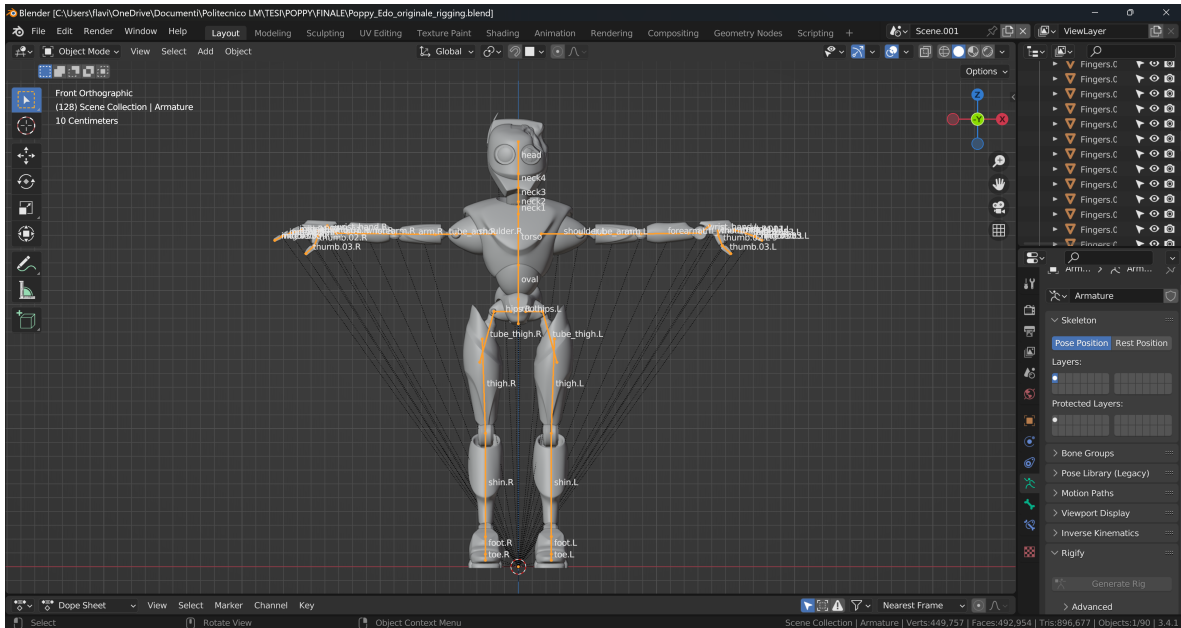


Figura 6.5: Armatura Modello B di Poppy
Estratto dal progetto "Poppy"

Le 61 ossa totali sono così suddivise:

- Una che serve da radice dell'armatura: "Bone".
- Tre per il busto: "root", "oval", "torso".
- Quattro per il collo: "neck1", "neck2", "neck3", "neck4".
- Una per la testa: "head".
- Quattro per ognuna delle due gambe: "hips.L", "tube_thigh.L", "thigh.L", "shin.L", "hips.R", "tube_thigh.R", "thigh.R", "shin.R".
- Due per ognuno dei due piedi: "foot.L", "toe.L", "foot.R", "toe.R".
- Cinque per ognuna delle due braccia: "shoulder.L", "tube_arm.L", "arm.L", "forearm.L", "arm_wrist.L", "shoulder.R", "tube_arm.R", "arm.R", "forearm.R", "arm_wrist.R".
- Tredici per ognuna delle due mani: "wrist_hand.L", "thumb.01.L", "thumb.02.L", "thumb.03.L", "index.01.L", "index.02.L", "index.03.L", "middle.01.L", "middle.02.L", "middle.03.L", "ring.01.L", "ring.02.L", "ring.03.L", "wrist_hand.R", "thumb.01.R", "thumb.02.R", "thumb.03.R", "index.01.R", "index.02.R", "index.03.R", "middle.01.R", "middle.02.R", "middle.03.R", "ring.01.R", "ring.02.R", "ring.03.R".

- Quattro ossa di controllo per l’IK: “heelIK.L”, “wristIK.L”, “heelIK.R”, “wristIK.R”.

3. Rigging

- Modello A

Il *rigging* del modello è stato fatto tramite l’opzione “*Set parent to Armature Deform With Empty Groups*” [35]. Questo metodo ha permesso di assegnare manualmente i gruppi di vertici delle *mesh* di Poppy alle diverse ossa. Trattandosi di un robot, costituito da componenti di acciaio che si muovono l’una rispetto all’altra, è stato associato un osso a ciascun pezzo.

Si noti che il modello 3D di Poppy presenta solo quattro dita per ogni mano. L’armatura, invece, dovendo essere compatibile con quella di Move.ai, presenta delle mani con cinque dita ciascuna. Questa incongruenza non ha rappresentato alcun problema poiché, grazie al *rigging* tramite “*empty groups*”, le ossa dei mignoli sono state associate alla *mesh* con pesi nulli.

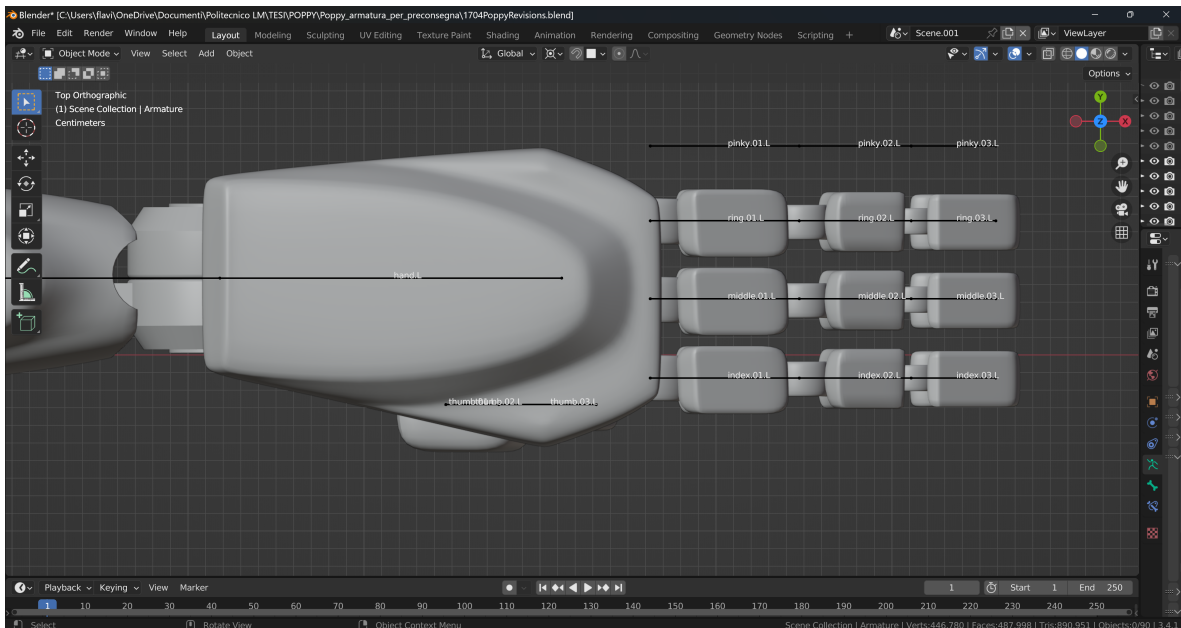


Figura 6.6: Mano Modello A di Poppy
Estratto dal progetto “Poppy”

Terminato il *rigging*, si è proceduto con l’esportazione del modello ‘riggato’ in formato .fbx. In fase di *export*, è stato necessario togliere la spunta alla voce “*Add leaf bones*” per evitare che venissero esportate troppe ossa.

- **Modello B**

Il *rigging* del modello B ha richiesto dei passaggi in più rispetto alla procedura utilizzata per quello A. Infatti, anche in questo caso, è stata sfruttata l'opzione “*Set parent to Armature Deform With Empty Groups*” per attaccare il modello allo scheletro.

In questo caso, non vi è stata l'incongruenza tra il numero di dita del modello 3D e il numero di dita del *rig*, in quanto l'armatura non aveva necessità di rispettare i requisiti di Move.ai.

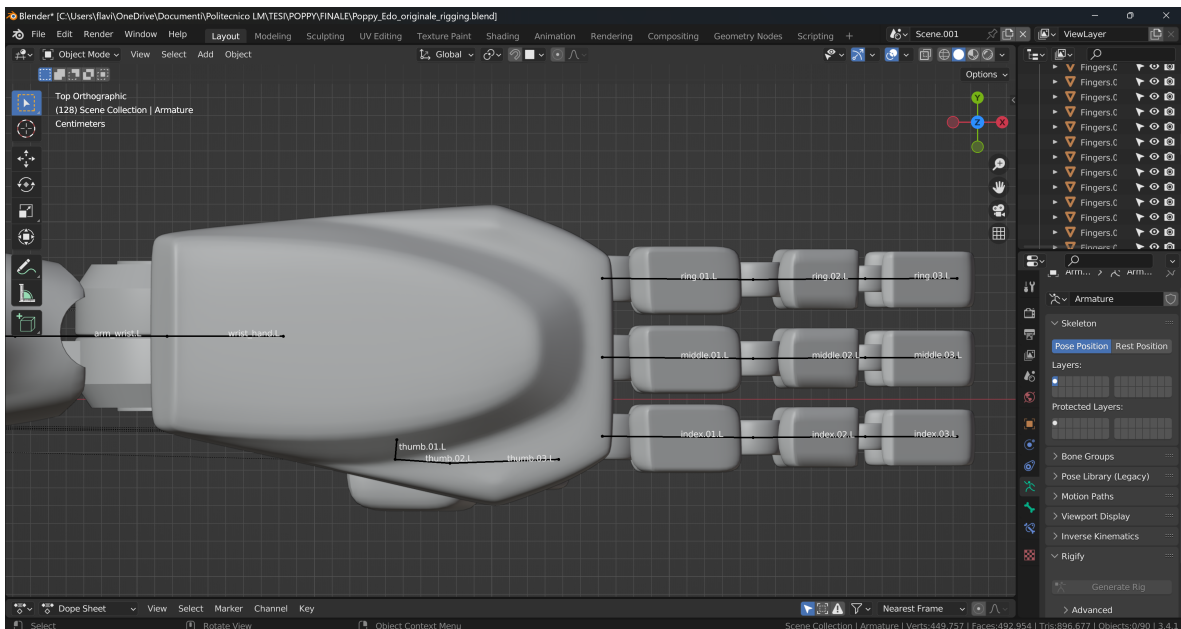


Figura 6.7: Mano Modello B di Poppy
Estratto dal progetto “Poppy”

4.a Move.ai per il Modello A

Terminati i passaggi precedenti, è stato possibile iniziare a lavorare con Move.ai.

- *Move.ai setup, calibration e recording*

Per poter utilizzare l'applicazione per iPhone “MoveAI” e la “Move web app”, è stato creato un nuovo *account* sul sito di Move.ai. Tramite l'*account* è stato possibile accedere alla piattaforma contenente tutti i dati caricati, modelli, *rig*, video e animazioni già realizzate.

Nel cortometraggio, Poppy viene quasi sempre inquadrato da solo, per cui, in scena, vi è sempre stato un unico attore. Questo dato è stato inserito nel *Volume Calculator* e, in base al numero di *devices* a disposizione, è stato deciso di utilizzare quattro telecamere per la ripresa della *performance*. Questo numero di dispositivi ha consentito di ottenere un tracciamento del movimento preciso e una buona trasposizione di questo sul modello 3D.

Gli iPhone sono stati disposti nello spazio secondo lo schema fornito dal *volume calculator* di Move.ai. L'area all'interno della quale l'attore poteva recitare è stata delimitata dal nastro adesivo posizionato sul pavimento. Predisposto lo spazio d'azione, si è proceduto con la calibrazione dei *devices*. Durante questa fase, è stata prestata particolare attenzione a far sì che i dispositivi non fossero ruotati rispetto alla linea di orizzonte. Inoltre, ci si è accertati che l'attore tenesse la posizione per permettere al *software* di individuare i suoi giunti e di rimapparne correttamente i movimenti. Per la procedura di calibrazione sono state seguite le indicazioni fornite dalla documentazione del *software*[26].

Terminata la fase di calibrazione, sono state effettuate le riprese delle *performance*. Successivamente, i video sono stati caricati dagli iPhone direttamente sulla "Move web app".

- *Upload e retargeting* del modello

Realizzati i video, si è proceduto con il caricamento del modello A di Poppy, in formato .fbx, sulla "Move web app". Terminato il caricamento del file, è stato effettuato il *retargeting* tra il *rig* di Poppy e l'armatura di riferimento di Move.ai. Il *retargeting* è consistito in un'associazione manuale delle ossa dello scheletro del *software* con quelle del personaggio, tramite l'elenco contenente i nomi delle componenti.

- *Data processing ed export*

In questa fase è stato creato un nuovo progetto sulla "Move web app". Sono stati inseriti nel *software* tutti i dati richiesti, necessari per far partire l'algoritmo, sono stati indicati i video da cui acquisire i dati del movimento e il *rig* su cui trasferirlo. Tra le impostazioni è stata attivata l'opzione di *hand capture*. Terminato l'inserimento dei dati si è proceduto facendo iniziare la procedura di trasposizione del movimento.



Figura 6.8: Fase di *recording*
Estratto dal progetto "Poppy"

4.b *Inverse Kinematics e constraints* Modello B

Per preparare il Modello B alla fase di *retargeting* presentata al punto 7, sono state impostate sia l'*Inverse Kinematics (IK)* sia le *constraints* per far rispettare ai giunti del robot i vincoli per il movimento.

L'*Inverse Kinematics* o cinematica inversa è un metodo di animazione che permette di ottenere risultati precisi con meno sforzo rispetto all'animazione tramite cinematica diretta. L'*IK* permette di posizionare nello spazio solo l'ultimo osso di una catena lasciando al *software* il compito di posizionare automaticamente nello spazio tutte le ossa rimanenti[35].

Per prima cosa, sia le braccia sia le gambe del Modello B sono state predisposte all'*Inverse Kinematics*. I passaggi effettuati sono stati i seguenti:

- In "*Edit mode*", sono state aggiunte 4 ossa, una per ognuna delle due caviglie e una per ciascuno dei due polsi: "heelIK.L", "heelIK.R", "wristIK.L", "wristIK.R".

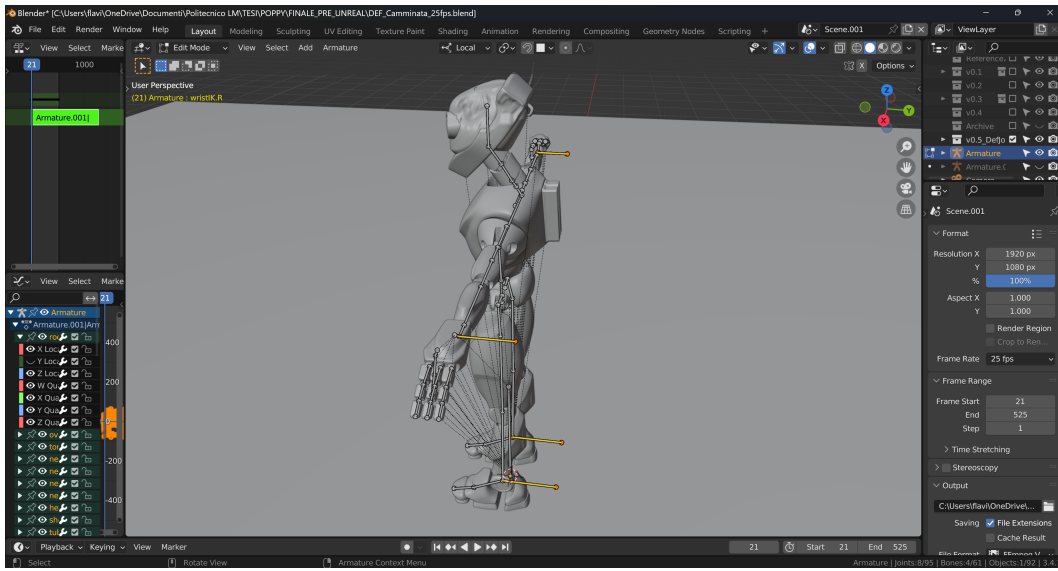


Figura 6.9: Ossa di controllo per l'Inverse Kinematics
Estratto dal progetto "Poppy"

- In "Edit mode", è stato aggiunto un quinto osso, nominato "Bone", che fungesse da origine di tutta l'armatura. Infatti, quest'osso è stato indicato come "Parent", nella sezione "Bone properties" di Blender, sia delle 4 ossa di controllo per l'IK appena create sia per l'osso "root", origine del resto dello scheletro.

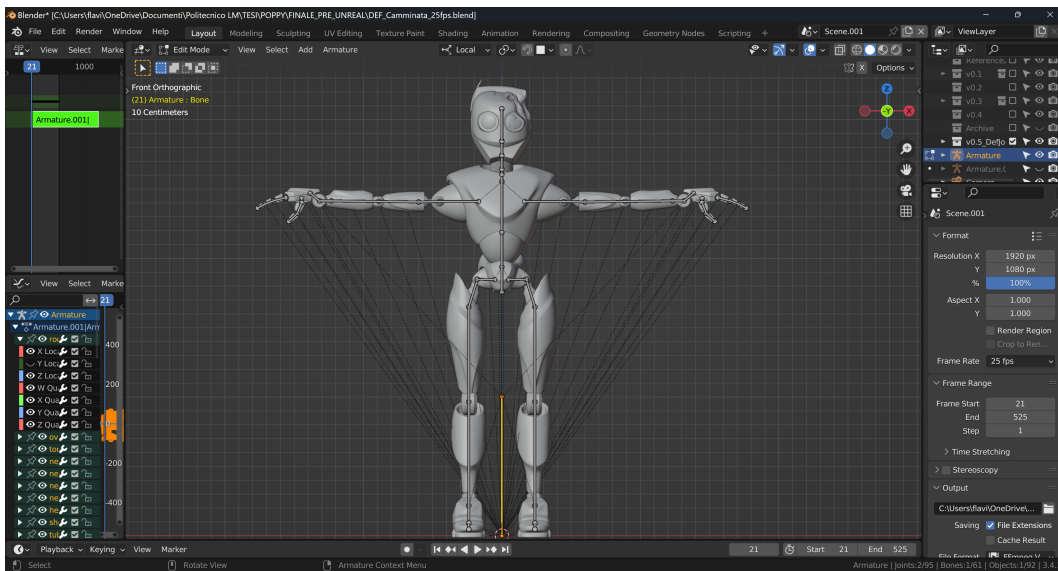


Figura 6.10: Osso radice
Estratto dal progetto "Poppy"

- In "Edit mode", per ognuna delle quattro ossa di controllo è stata tolta la spunta alla voce "Deform", nel pannello "Bone properties", in modo che queste ossa non

influenzassero la geometria del modello.

- In “*Pose mode*”, è stata aggiunta l’*IK*.

Per la gamba sinistra, è stato selezionato l’osso del polpaccio, shin.L, nel pannello “*Bone Constraint properties*” e nella tendina di “*Add Bone Constraint*” è stata aggiunta la voce “*Inverse Kinematics*”. Come “*Target*” è stata indicata l’armatura stessa e come osso il “heelIK.L”. La lunghezza della catena è stata impostata a 4 in modo che il movimento dell’osso influenzasse tutte le ossa della gamba, compreso quello del bacino. Successivamente, per non far muovere troppo l’heelIK.L rispetto al piede, in “*Edit mode*” sono state selezionate le ossa “foot.L” e “heelIK.L” e sono state imparentate con l’opzione “*Keep offset*”. Infine, è stato selezionato l’osso del piede “foot.L” e tramite il pannello “*Bone Constraint properties*”, nella tendina di “*Add Bone Constraint*”, è stata aggiunta la voce “*Copy Location*” in cui è stata inserita l’armatura come “*Target*” e l’osso del polpaccio, “shin.L”, come osso di riferimento. Qui, il parametro della voce “*Head/Tail*” è stato impostato a 1.0 per evitare che l’osso dell’*IK* si spostasse dalla posizione di base. Per la gamba destra è stato seguito esattamente lo stesso procedimento ma con le corrispettive ossa (identificate del suffisso -.R). Per il braccio sinistro, è stato selezionato l’osso del polso, “arm_wrist.L”, nel pannello “*Bone Constraint properties*” e nella tendina di “*Add Bone Constraint*” è stata aggiunta la voce “*Inverse Kinematics*”. Come “*Target*” è stata indicata l’armatura stessa e come osso il “wristIK.L”. La lunghezza della catena è stata impostata a 5 in modo che il movimento dell’osso influenzasse tutte le ossa del braccio, compreso quello della spalla. Successivamente, per non far muovere troppo il “wristIK.L” rispetto all’osso polso-mano, in “*Edit mode*” sono state selezionate le ossa “wrist_hand.L” e “wristIK.L” e sono state imparentate con l’opzione “*Keep offset*”. Infine, è stato selezionato l’osso polso-mano “wrist_hand.L” e tramite il pannello “*Bone Constraint properties*”, nella tendina di “*Add Bone Constraint*”, è stata aggiunta la voce “*Copy Location*” in cui è stata inserita l’armatura come “*Target*” e l’osso del polso, “arm_wrist.L”, come osso di riferimento. Qui, il parametro della voce “*Head/Tail*” è stato impostato a 1.0 per evitare che l’osso dell’*IK* si spostasse dalla posizione di base. Per il braccio destro è stato seguito esattamente lo stesso procedimento ma con le corrispettive ossa (identificate del suffisso -.R).

Terminate le impostazioni per l’*Inverse Kinematics* sono stati aggiunti i vincoli a tutte le ossa dell’armatura. In fase di riproduzione, sono stati definiti i gradi di libertà di ogni giunto tra una componente e l’altra di Poppy. I vincoli sono stati impostati per non far muovere in modo errato i vari pezzi del robot.

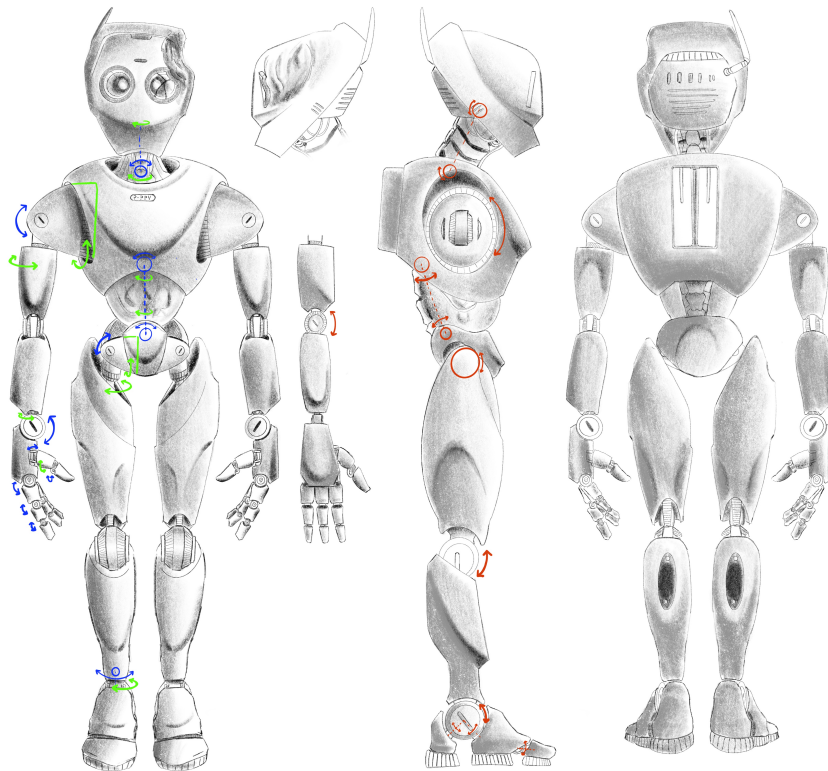


Figura 6.11: Gradi di libertà per le componenti di Poppy
Estratto dal progetto "Poppy"

Per fare ciò, in *"Pose mode"* è stato selezionato, uno per volta, ogni osso dell'armatura e, nel pannello *"Bone Constraint Properties"* è stato aggiunto il vincolo *"Limit Rotation"*. Nel pannello delle impostazioni del vincolo di rotazione è stata selezionata la dicitura *"Owner: Local Space"* in modo che i vincoli venissero applicati alle ossa rispetto al sistema di riferimento dell'armatura indipendentemente dalla posizione di questa nello spazio. I vincoli sono stati aggiunti su tutti e tre gli assi x, y e z per ogni osso. Ruotando ogni osso, su ognuno dei tre assi, sono stati individuati i gradi, minimo e massimo, di rotazione che questo poteva effettuare senza che vi fossero intersezioni con altre parti del corpo di Poppy.

Terminato l'inserimento dei vincoli delle ossa dello scheletro, il Modello B era pronto per ricevere l'animazione dal Modello A.

5. *Import* animazione del Modello A in Blender

Terminata l'elaborazione dei dati sulla "Move web app", si è proceduto con il *download* del file .fbx contenente l'animazione. Successivamente, il file è stato importato in Blender per poterne visualizzare il contenuto. In fase di *import*, nella finestra delle impostazioni

di Blender, è stata tolta la spunta all’opzione “*Add leaf bones*”, in modo da non avere ossa superflue. Aperto il progetto, l’armatura è stata subito scalata, in “*Edit mode*” di un fattore 1/100 per poterla vedere correttamente.

Dopo aver aggiustato lo scheletro e dopo aver eliminato gli oggetti superflui presenti in scena, è stato possibile visualizzare l’animazione di Poppy.

6. *Add-on* di Rokoko Vision

Come anticipato in precedenza, il Modello A è stato ‘retargettizzato’ sul Modello B in modo da avere dei movimenti più precisi e che rispettassero i gradi di libertà dei giunti del robot. Per effettuare il *retargeting*, è stato sfruttato l’*add-on* “*Animation: Rokoko Vision Studio Live for Blender*”. Questo *tool* ha permesso di trasferire agevolmente i dati di movimento del Modello A sul Modello B.

7. *Retargeting*

Per effettuare il *retargeting* in modo semplice e veloce, è stato utilizzato l’*add-on* per Blender di Rokoko Vision.

Al fine di comprendere meglio la procedura effettuata è importante sottolineare che la differenza principale tra le due armature sta nel numero di ossa presenti negli arti, infatti, il Modello B ha due ossa in più per ogni braccio e un osso in più in ogni gamba rispetto al Modello A. Questo sovrannumero di ossa è stato necessario per l’animazione definitiva di Poppy in quanto ogni pezzo dei suoi arti è stato progettato per ruotare su un solo asse. Per far sì che ogni pezzo ruotasse attorno ad un unico asse, nella fase 6.1, ciascun osso è stato lasciato libero di ruotare solo sull’asse previsto ed è stato limitato sugli altri due assi impostando una rotazione variabile tra 0° e 0° , ovvero nulla.

Per prima cosa, è stato creato sul sito di Rokoko Vision un *account*. Grazie all’*account* è stato possibile fare l’accesso da Blender ed utilizzare lo strumento per il *retargeting*. Per effettuare l’associazione e trasferire il movimento da un modello all’altro, è stato sufficiente indicare l’armatura dalla quale prendere il movimento, in questo caso lo scheletro del Modello A, e l’armatura a cui trasferire il movimento, in questo caso lo scheletro del Modello B.

L’*add-on* offre due possibilità, quella di associare manualmente un osso al corrispondente oppure di fare il collegamento in modo automatico. Nel caso di Poppy ha funzionato bene il collegamento automatico in quanto i nomi delle ossa dei due *rig* erano uguali. Ovviamente, avendo un numero di ossa differente, lo scheletro B aveva delle ossa a cui non era associato nulla. Quindi, si è proceduto aggiungendo manualmente i

collegamenti mancanti. Per fare ciò, alcune ossa del Modello A sono state associate a due ossa del Modello B, seguendo una mappatura 1:2 anziché 1:1.

In questo modo, teoricamente, le ossa dell'armatura B a cui è stato assegnato il movimento dello stesso osso dello scheletro A, avrebbero dovuto avere lo stesso movimento. Sennonché, grazie ai vincoli di rotazione impostati al punto 6.1 della procedura, le ossa del Modello B hanno preso solo la componente del movimento dell'osso del Modello A su cui non avevano alcun vincolo di rotazione.

Grazie a questo *escamotage*, è stato possibile scomporre la rotazione di un osso del Modello A, nelle tre componenti x, y e z, ed associarle separatamente alle ossa del Modello B. In questo modo si è potuta trasferire l'animazione realizzata con Move.ai ad un modello con una morfologia non umana.

8. Modifica del *framerate*

Prima di effettuare la correzione manuale, è stato notato che i video della *performance* sono stati girati a 60fps e che, quindi, anche l'animazione aveva questo *framerate*. Poiché il progetto di Unreal Engine 5, in cui sarebbe stata importata l'animazione finale di Poppy, era stato realizzato a 25fps, è stato necessario modificare il *framerate* dell'animazione in modo che fosse compatibile con il resto.

Per modificare il *framerate*, è stato calcolato il numero di *frame* che avrebbe dovuto avere l'animazione a 25fps. La *timeline* è stata impostata per mostrare esattamente il numero di *frame* trovati. In seguito, sono stati selezionati tutti i *keyframe* delle ossa e sono stati riscritti sul nuovo numero di frame. In questo modo l'animazione è stata portata al *framerate* desiderato senza modificarne la velocità.

9. Correzione manuale dell'animazione

Benché i movimenti di Poppy fossero abbastanza fluidi e fedeli alla *performance* dell'attore, il *retargeting* ha introdotto delle imprecisioni nel movimento.

In alcuni punti, l'animazione presentava delle intersezioni tra *mesh* che sono state eliminate modificando, uno per uno, i *keyframe* interessati. Inoltre, verosimilmente causa della modifica del *framerate*, in certi punti l'animazione risultava scattosa. Per ovviare a questo problema sono stati modificati, aggiunti e, talvolta, eliminati, i *keyframe* che causavano i movimenti errati.

Per limitare le possibilità di commettere degli errori in fase di animazione manuale, sono stati importati in Blender i video della *performance* da usare come *references* per

l'animazione. Infatti, nei punti in cui è stato necessario eliminare più *keyframe*, è stato possibile ricreare le posizioni corrette del modello, nei momenti giusti, sulla base delle posizioni dell'attore nel video.

Grazie agli accorgimenti presi in tutte le fasi precedenti, le correzioni da apportare manualmente sono state poche e limitate agli errori introdotti dal *retargeting*.

10. *Texturing*

Terminata la fase di animazione, il modello è stato passato al “Reparto 3D” di Robin Studio che ha aggiunto, al modello animato, le *texture* realizzate con Substance 3D Painter.

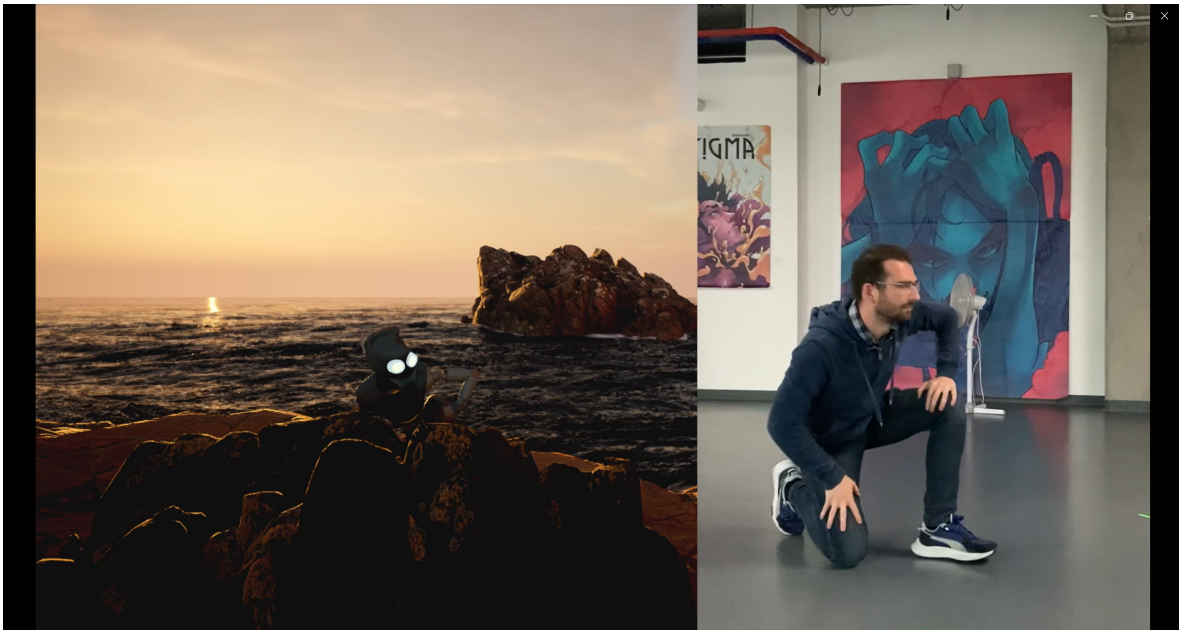


Figura 6.12: Poppy segue i movimenti del performer
Estratto dal progetto “Poppy”

6.2 Problemi e soluzioni

Durante il *workflow* di lavoro per l'animazione di Poppy sono stati presi diversi accorgimenti che hanno permesso di ridurre al minimo i problemi riscontrati. Infatti, grazie all'analisi dei problemi individuati nel processo di animazione dell'astronauta e alle soluzioni adoperate, è stato possibile evitare di incorrere negli stessi errori (si veda il capitolo 5.2).

La pipeline di lavoro usata come linea guida è stata quella presentata nel capitolo 5.3. A causa delle differenze strutturali dei due personaggi, però, sono state apportate alcune modifiche eliminando alcuni passaggi e introducendone di nuovi. Come per l'animazione dell'astronauta, anche in questo caso, l'approccio utilizzato è stato quello del *trial and error* che, grazie alle modifiche apportate volta per volta, ha permesso di identificare le soluzioni ottimali.

I problemi individuati appartengono principalmente alle fasi finali del *workflow*, sia per sbagli commessi durante la realizzazione di questi step sia per l'ordine in cui queste procedure sono state effettuate. Dopo diversi tentativi, sono stati trovati e corretti gli errori e, successivamente, è stato trovato l'ordine corretto in cui eseguire i passi delle procedura.

Di seguito, sono presentati i problemi e sono illustrate e motivate le soluzioni messe in pratica durante il processo di lavorazione.

Centri di rotazione

Terminata la fase di *rigging*, sia per il Modello A sia per il Modello B, sono state effettuate delle prove per verificare che il movimento dell'armatura e la deformazione delle *mesh* fossero corretti. Grazie a questi controlli, è stato notato che alcuni pezzi non ruotavano perfettamente rispetto agli altri, infatti, alcuni pezzi meccanici, se mossi, fuoriuscivano dalla loro sede.

Si noti che le ossa di un'armatura, in Blender, sono composte da una testa, *head*, e una coda, *tail*. La testa dell'osso è l'estremità attorno a cui avviene la rotazione, per cui è necessario che la *i* si trovi nel perno su cui si vuole far ruotare la *mesh*.

Per risolvere il problema, quindi, sono state riposizionate le ossa prestando attenzione a sistemarne le teste in corrispondenza dei fulcri sui quali farle ruotare. Sono state prese le *mesh*, una per una, ed è stata effettuata la seguente procedura:

1. In “*Edit mode*”, è stata selezionata la *mesh* interessata, è stato individuato il punto su cui effettuare le rotazioni e, tramite “*Shift+S -j Cursor to selected*”, è stato spostato il cursore nella posizione desiderata.
2. In “*Object mode*”, tramite il tab “*Object -j set origin -j origin to 3D cursor*” è stato spostato il cursore sul punto selezionato il precedenza.
3. In “*Edit mode*”, è stata selezionata la testa dell'osso che muove il pezzo interessato e, tramite “*Shift+S -j selection to cursor*” questa è stata spostata nel nuovo centro di rotazione.

Effettuando nuovamente dei controlli per verificare la correttezza dei movimenti e delle deformazioni delle *mesh*, si è visto che non si verificavano più errori.

Inverse Kinematics

In un primo momento, è stato effettuato il *retargeting* dal Modello A al Modello B senza che su quest'ultimo fosse applicata l'*Inverse Kinematics*. La trasposizione dell'animazione, però, non andava molto bene. Infatti, le gambe del personaggio subivano degli slittamenti durante la camminata e le braccia si muovevano in modo scorretto facendo degli scatti nella direzione opposta rispetto a quella prevista.

Per questi errori nel trasferimento del movimento da un'armatura all'altra, si è deciso di provare a predisporre l'armatura ricevente, quella del Modello B, all'animazione tramite l'*Inverse Kinematics*.

Aggiungendo all'armatura le ossa di controllo per l'*IK* e impostando tutti i parametri in modo che questa funzionasse, si è proceduto con un nuovo *retargeting*. In questo caso, la trasposizione dell'animazione ha funzionato bene e i movimenti del robot seguivano perfettamente quelli del modello animato con Move.ai.

Constraints

Essendo un robot, Poppy può effettuare dei movimenti ben precisi dettati dalla struttura meccanica da cui è costituito. Per questo motivo è stato necessario limitare le rotazioni di alcuni pezzi, sia per evitare le intersezioni tra *mesh* sia per rispettare la morfologia delle componenti.

Poiché Move.ai non accetta *rig* con delle *constraints* applicate, allo scheletro del Modello A non è stato applicato nessun vincolo mentre per l'armatura del Modello B sono stati impostati i limiti di rotazione. In un primo momento, le *constraints* sui vari pezzi erano state inserite impostando i parametri dell'opzione "*Limit rotation*" dal pannello "*Bone Constraint Properties*". I vincoli posti in questa modalità sono stati efficaci finché non si è effettuato il *retargeting* dell'armatura A sull'armatura B. Infatti, dopo la trasposizione del movimento da uno scheletro all'altro, i vincoli non erano rispettati. In seguito, è stato fatto un secondo tentativo effettuando prima il *retargeting* dall'armatura A sulla B e impostando, solo in un secondo momento, i vincoli sull'armatura finale, sempre tramite l'opzione "*Limit rotation*". Anche in questo caso, l'animazione non rispettava i vincoli di rotazione impostati.

Si è notato che, se all'armatura su cui veniva fatto il *retargeting* si toglieva l'impostazione dell'*Inverse Kinematics* allora i "*Limit rotation*" funzionavano. Non potendo

lavorare senza l'IK, in quanto l'animazione finale non avrebbe funzionato bene, si è capito di dover mettere i vincoli modificando le impostazioni dell'IK stessa. È stato, quindi, provato un terzo approccio facendo il *retargeting* e, successivamente, aggiungendo i vincoli. Al contrario delle prove precedenti, questa volta le *constraints* sono state applicate dal pannello “*Bone Properties*”, sotto la voce “*Inverse Kinematics*”. Qui è stato possibile bloccare l'IK su ciascuno dei tre assi x, y e z e limitarne la rotazione specificando il grado minimo e massimo raggiungibile. In questo modo, i vincoli sono stati rispettati e l'animazione è rimasta intatta.

6.3 Workflow per l'animazione di personaggi robotici con Move.ai

In base al *workflow* che ha portato ad ottenere l'animazione finale di Poppy, ai problemi riscontrati e alle soluzioni trovate, di seguito viene proposto un *workflow* ottimale per l'animazione di personaggi robotici e, più in generale, di personaggi con un'armatura non umanoide. Oltre ai passaggi da effettuare, sono elencati, per ciascuno step, gli aspetti a cui prestare particolare attenzione per ottimizzare i tempi in fase di lavorazione e per migliorare la resa finale del prodotto.

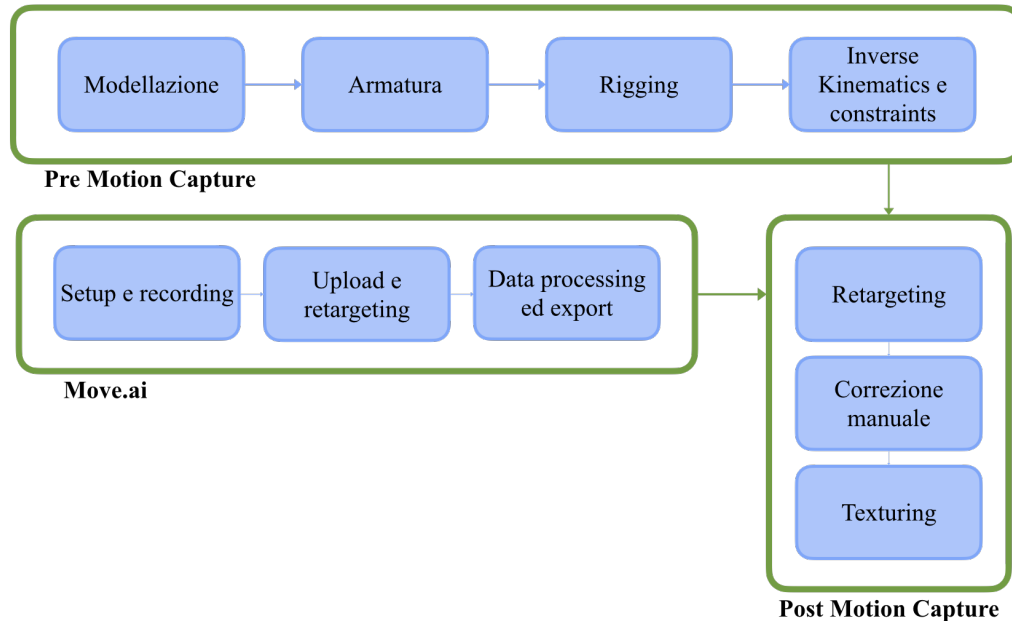


Figura 6.13: Workflow ottimale per l'animazione di personaggi robotici con Move.ai

1. Modellazione

In questa fase bisogna creare il modello 3D definitivo del personaggio da animare. Il *software* di modellazione è a discrezione del professionista che se ne occupa.

La realizzazione del modello finale è importante in quanto può ridurre i tempi di produzione. Infatti, se vi sono abbastanza risorse disponibili, è possibile realizzare più passaggi parallelamente.

In questo caso particolare, in cui bisogna animare un modello la cui armatura finale non è compatibile con Move.ai, e, quindi, il *retargeting* è necessario a prescindere, è possibile svolgere la fase di modellazione contemporaneamente alla fase di animazione con Move.ai. Infatti, anziché creare due copie del modello, due armature ed effettuare due *rigging*, è possibile dimezzare i tempi di lavorazione realizzando l'animazione con Move.ai sfruttando uno dei modelli già 'riggati' a disposizione nella sua libreria e 'retargettizzare' quest'animazione sul modello definitivo.

Nel caso di personaggi robotici o, più in generale, di personaggi composti da pezzi rigidi, non malleabili e non deformabili, è conveniente realizzare il modello mantenendo separate le *mesh* che lo compongono. Questo *escamotage* semplifica molto la fase di *rigging*.

2. Armatura modello definitivo

Per creare l'armatura è possibile sfruttare scheletri già esistenti. Blender offre la possibilità di generare automaticamente diversi tipi di *rig*, da quelli base come “*Basic Human Meta-Rig*” e “*Basic Quadruped Meta-Rig*” a quelli più specifici come “*Bird Meta-Rig*”, “*Cat Meta-Rig*”, “*Horse Meta-Rig*”, “*Wolf Meta-Rig*”, “*Shark Meta-Rig*”. A partire da questi, è possibile aggiungere, eliminare e modificare le ossa necessarie per adattare l'armatura al modello in questione. In alternativa, è possibile comporre l'armatura da zero. La scelta di uno dei due metodi è a discrezione del professionista. Durante la realizzazione del *rig* conviene nominare le ossa in modo chiaro per poter effettuare il *retargeting* in modo più semplice.

3. *Rigging* modello definitivo

Per 'riggare' correttamente il personaggio bisogna scegliere il metodo più conveniente in base al modello con cui si lavora. Infatti, è possibile avvalersi di strumenti che effettuano il *rigging* automaticamente come il “*Set parent to Armature Deform With Automatic Weights*” oppure di *tool* che si limitano a predisporre il modello per il *rigging* come il “*Set parent to Armature Deform With Empty Groups*” di Blender. È anche possibile realizzare i *rigging* completamente da zero, creando i gruppi di vertici manualmente.

Nel caso di personaggi robotici, solitamente composti da parti meccaniche, il modo più conveniente è quello di predisporre il modello creando automaticamente i gruppi di vertici vuoti e assegnarli manualmente alle *mesh* dei vari componenti.

4. *Inverse Kinematics e Constraints*

La predisposizione del modello all'*Inverse Kinematics* è importante per poter ottenere un'animazione fluida. L'*IK* è particolarmente utile per animare gli arti in modo semplice e veloce, infatti muovendo solo l'ultimo pezzo della catena cinematica, tutti gli altri si muovono di conseguenza. In questo caso, in cui l'animazione viene realizzata trasferendola da un altro modello, è importante dotare l'armatura ricevente dell'*IK* per non avere errori nella trasposizione del movimento.

5. Move.ai

- *Move.ai setup, calibration e recording*

In questa fase è fondamentale eseguire correttamente il processo di calibrazione. Gli aspetti principali a cui prestare attenzione sono gli stessi descritti nella sezione 5.2.

Un altro aspetto da sottolineare è quello del *framerate*. Infatti, è bene accertarsi che il *framerate* con il quale si girano i video della *performance* sia quello definitivo. Girare ad un *framerate* diverso da quello voluto per il progetto finale non costituisce un problema ma allunga il processo di lavorazione e rischia di introdurre degli errori da gestire in fase di correzione manuale.

- *Upload del modello e retargeting*

In questo caso, come spiegato nel punto 6.3, dovendo 'retargettizzare' l'armatura su un modello con un *rigging* diverso, non è conveniente caricare un modello proprio sul *software*. Per l'animazione, infatti, conviene utilizzare uno dei modelli già presenti nella libreria di Move.ai.

- *Data processing ed export*

Per l'elaborazione dei dati serve selezionare il modello della libreria che si vuole utilizzare, impostare i video da usare per la cattura della *performance* e accertarsi che sia attiva l'opzione della *hand capture*. Terminato l'elaborazione si può procedere al *download* del file .fbx generato.

6. *Import* animazione in Blender

Durante l'*import* del file .fbx in Blender, bisogna accertarsi di togliere la spunta alla voce “*Add leaf bones*” in modo da non doverlo fare in una fase successiva. In questo caso è necessario importare solo l'armatura del modello. Talvolta può essere utile importare a *mesh* del modello per avere dei punti di riferimento in più.

7. *Add-on*

Per effettuare il *retargeting* vi sono diverse opzioni tra cui scegliere. È possibile effettuare il *retargeting* manualmente, ad esempio utilizzando le “*Object Properties*” di Blender come “*Copy Location*” e “*Copy Rotation*”. In alternativa, si possono usare degli strumenti di supporto che facilitano la procedura e la rendono più veloce. Una delle opzioni è l'*add-on* “*Animation: Rokoko Vision Studio Live for Blender*” di Rokoko Vision. Questa funzionalità permette di far eseguire il *retargeting* automaticamente al *software* lasciando comunque la possibilità di modificare le impostazioni manualmente.

8. *Retargeting*

Grazie all'*add-on* di Rokoko Vision, questo passaggio è semplice da eseguire. Nonostante ciò, è importante prestare attenzione all'associazione che il *software* fa. Talvolta, è necessario aggiungere o eliminare alcune delle associazioni generate da Rokoko Vision. Infatti, nel caso di personaggi non umanoidi è possibile che debbano essere fatte associazioni tra ossa che nelle due armature rappresentano cose diverse. Inoltre, è molto probabile che le associazioni tra i due *rig* non siano 1:1 bensì 1:n, con n variabile. Questa mappatura non univoca può risultare necessaria per scomporre il movimento di un osso dell'armatura animata sui tre assi, x, y e z e per ricomporlo in maniera differente sull'armatura che riceve l'animazione.

9. *Correzione manuale dell'animazione*

La correzione manuale, seppur in dosi variabili, è sempre necessaria alla fine di una procedura di animazione tramite *Mocap* poiché, nelle varie fasi del *workflow*, possono verificarsi delle imprecisioni nei movimenti. Se possibile, è preferibile correggere ciò che si può in altre fasi del processo di animazione per ridurre al minimo gli aggiustamenti da effettuare manualmente in quanto sono molto onerosi. La scelta del metodo migliore da applicare per la correzione dipende dal professionista che se ne occupa.

10. *Texturing*

Le *textures*, seppur attaccate al modello solo in questo step, possono essere realizzate anche prima. Infatti, se il modello del personaggio è quello definitivo, parallelamente all'animazione, è possibile realizzare le *textures*.

Capitolo 7

Conclusioni

Gli algoritmi di *computer vision*, *machine learning* e le reti neurali, che costituiscono le basi dell'Intelligenza Artificiale, stanno rivoluzionando le tecnologie conosciute. Il campo dei *Visual Effects*, in continua evoluzione, ha trovato nell'Intelligenza Artificiale la soluzione per ridurre i costi e i tempi per la produzione di contenuti di alto livello.

La *Motion Capture*, procedura ultracentenaria che ha attraversato diverse fasi evolutive, insieme alla potenza dell'Intelligenza Artificiale è divenuta una delle tecniche più all'avanguardia degli ultimi anni. L'*AI-based Motion Capture* consente di trasferire il movimento dal mondo reale al mondo virtuale, dando vita ad oggetti e personaggi inanimati.

La *MoCap*, da sempre prodotto di nicchia accessibile solo ai *colossal* hollywoodiani, ai centri di ricerca e alle grandi compagnie, grazie alla diffusione dell'Intelligenza Artificiale, è oggi una tecnologia accessibile a tutti, in qualunque momento, ovunque. I costi ridotti dell'*AI MoCap* e la sua facilità di utilizzo, permettono anche alle piccole aziende di sperimentare e di contribuire allo sviluppo di questa tecnologia.

La presente tesi di laurea, tramite l'animazione dei personaggi di "Poppy", dimostra come l'accessibilità dell'*AI MoCap* abbia permesso la realizzazione di un prodotto tecnicamente innovativo e all'avanguardia, pur essendo *low-budget*. Inoltre, lo studio dei software presenti sul mercato e l'approccio sperimentale utilizzato hanno consentito di ideare un *workflow* di lavoro ottimale per animare determinate categorie di personaggi e di individuare i pregi e i limiti di tale tecnica.

Al momento, l'*AI Motion Capture* continua ad evolversi offrendo nuove idee e opportunità per la cattura del movimento. Impiegata in diversi settori, dal cinema allo sport, dalla scienza alla medicina, l'*AI MoCap* sta rivoluzionando il modo di catturare e riprodurre il movimento, portando soluzioni innovative nella vita di tutti i giorni.

Bibliografia e sitografia

- [1] The Editors of Encyclopaedia Britannica. «Fleischer brothers». In: *Encyclopedia Britannica* (2023). Ultimo accesso: 02 Ottobre 2023. URL: <https://www.britannica.com/biography/Fleischer-brothers>.
- [2] Adobe. «Animazioni in rotoscoping». In: *Adobe.com* (). Ultimo accesso: 02 Ottobre 2023. URL: <https://www.adobe.com/it/creativecloud/video/discover/rotoscoping-animation.html>.
- [3] The Editors of Encyclopaedia Britannica. «resistor». In: *Encyclopedia Britannica* (2023). Ultimo accesso: 02 Ottobre 2023. URL: <https://www.britannica.com/biography/Fleischer-brothers>.
- [4] Adobe. «What is motion capture and how does it work?». In: *Adobe.com* (). Ultimo accesso: 02 Ottobre 2023. URL: <https://www.adobe.com/uk/creativecloud/animation/discover/motion-capture.html>.
- [5] David J. Sturman. «A Brief History of Motion Capture for Computer Character Animation». In: *MEDIALAB* (1999). Last changed March 13, 1999, G. Scott Owen. Ultimo accesso: 02 Ottobre 2023. URL: https://education.siggraph.org/static/HyperGraph/animation/character_animation/motion_capture/history1.htm.
- [6] Delle Rae Maxwell. «GRAPHICAL MARIONETTE: A MODERN-DAY PINOCCHIO». In: (1983). Master of Science in Visual Studies, Department of Architecture. Department of Distinctive Collections, MIT Libraries, Cambridge, Massachusetts. URL: <https://dspace.mit.edu/bitstream/handle/1721.1/76176/11521682-MIT.pdf?sequence=2&isAllowed=y>.
- [7] Pedro Nogueira. «Motion Capture Fundamentals A Critical and Comparative Analysis on Real-World Applications». In: (2011). Programa Doutoral em Engenharia Informática, Instituto de Telecomunicações. Ultimo accesso: 03 Ottobre 2023. URL: https://paginas.fe.up.pt/~prodei/dsie12/papers/paper_7.pdf.
- [8] Andrea Bottino. «Slide del corso di Realtà Virtuale». Politecnico di Torino. 2021.
- [9] Anna Sophia Reuter e Maike Schindler. «Motion Capture Systems and Their Use in Educational Research: Insights from a Systematic Literature Review». In: *Education Sciences* 13.2 (2023). ISSN: 2227-7102. DOI: 10.3390/educsci13020167. URL: <https://www.mdpi.com/2227-7102/13/2/167>.

- [10] Chang June Lee e Jung Keun Lee. «Inertial Motion Capture-Based Wearable Systems for Estimation of Joint Kinetics: A Systematic Review». In: *Sensors (Basel, Switzerland)* (2022). Ultimo accesso: 03 Ottobre 2023. URL: <https://doi.org/10.3390/s22072507>.
- [11] Maureen Furniss. «Motion Capture». In: *mit communications forum* (). Ultimo accesso: 03 Ottobre 2023. URL: <https://web.mit.edu/comm-forum/legacy/papers/furniss.html>.
- [12] Riccardo Antonio Silvio Antonino. «Slide del corso di Visual Effects». Politecnico di Torino. 2023.
- [13] Liu S., Zhang J. e Zhang et al Y. «A wearable motion capture device able to detect dynamic motion of human limbs». In: *Nat Commun* 11, 5615 (2020). Ultimo accesso: 03 Ottobre 2023. URL: <https://www.nature.com/articles/s41467-020-19424-2>.
- [14] Mündermann L., Corazza S. e Andriacchi TP. «The evolution of methods for the capture of human movement leading to markerless motion capture for biomechanical applications». In: *J Neuroeng Rehabil* (2006). Ultimo accesso: 03 Ottobre 2023. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1513229/>.
- [15] Ricotti V., Kadirvelu B. e Selby et al V. «Wearable full-body motion tracking of activities of daily living predicts disease trajectory in Duchenne muscular dystrophy». In: *Nat Med* 29, 95–103 (2023). Ultimo accesso: 03 Ottobre 2023. URL: <https://doi.org/10.1038/s41591-022-02045-1>.
- [16] Kadirvelu B., Gavriel C. e Nageshwaran et al. S. «A wearable motion capture suit and machine learning predict disease progression in Friedreich’s ataxia». In: *Nat Med* 29, 86–94 (2023). Ultimo accesso: 03 Ottobre 2023. URL: <https://doi.org/10.1038/s41591-022-02159-6>.
- [17] Duffy K. «The evolution of methods for the capture of human movement leading to markerless motion capture for biomechanical applications». In: *Vicon* (2020). Ultimo accesso: 03 Ottobre 2023. URL: <https://www.technologynetworks.com/tn/blog/motion-capture-trends-in-biomechanical-science-329070>.
- [18] Hando BR et al. «Association Between Markerless Motion Capture Screenings and Musculoskeletal Injury Risk for Military Trainees: A Large Cohort and Reliability Study». In: *Orthop J Sports Med* (2021). Ultimo accesso: 04 Ottobre 2023. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8558809/>.
- [19] Yellamma Pachipala et al. «Implementing Robots in Defence Through Motion Capture with Mixed Reality». In: *International Journal of Engineering and Technology(UAE)* (2018). Ultimo accesso: 04 Ottobre 2023. URL: https://www.researchgate.net/publication/329377160_Implementing_Robots_in_Defence_Through_Motion_Capture_with_Mixed_Reality.

- [20] Yeadon M.R. e Pain M.T.G. «Fifty years of performance-related sports biomechanics research». In: *Journal of Biomechanics, Volume 155* (2023). Ultimo accesso: 04 Ottobre 2023. URL: <https://www.sciencedirect.com/science/article/pii/S002192902300235X?via%3Dihub>.
- [21] Ted Litchfield. «Baldur's Gate 3 used motion capture from 248 actors to bring its NPCs to life: 'You're not only hearing the actors' voices, but you're also seeing their physical performances'». In: *pcgamer.com* (2023). Ultimo accesso: 04 Ottobre 2023. URL: <https://www.pcgamer.com/baldurs-gate-3-used-motion-capture-from-248-actors-to-bring-its-npcs-to-life-youre-not-only-hearing-the-actors-voices-but-youre-also-seeing-their-physical-performances/>.
- [22] D. Zhang et al. «Using AI Motion Capture Systems to Capture Race Walking Technology at a Race Scene: A Comparative Experiment». In: *Applied Sciences, 13(1), 113* (2022). Ultimo accesso: 29 Settembre 2023. URL: <https://doi.org/10.3390/app13010113>.
- [23] *DeepMotion*. Ultimo accesso: 29 Settembre 2023. URL: <https://www.deepmotion.com/about>.
- [24] *DeepMotion documentation*. Ultimo accesso: 29 Settembre 2023. URL: <https://www.deepmotion.com/documentation>.
- [25] *Move.ai*. Ultimo accesso: 29 Settembre 2023. URL: <https://www.move.ai/aboutus>.
- [26] *Move.ai documentation*. Ultimo accesso: 29 Settembre 2023. URL: <https://knowledgebase.move.ai/knowledge>.
- [27] *Rokoko Vision*. Ultimo accesso: 29 Settembre 2023. URL: <https://www.rokoko.com/products/vision>.
- [28] *Rokoko Vision documentation*. Ultimo accesso: 29 Settembre 2023. URL: <https://docs.rokoko.com/rkk-vision-documentation/>.
- [29] Business Wire Writers. «Wonder Dynamics Officially Launches Wonder Studio, a First of Its Kind AI Tool for the Film and TV Industry». In: *Business Wire: A BERCKSHIRE HATHAWAY COMPANY* (2023). Ultimo accesso: 29 Settembre 2023. URL: <https://www.businesswire.com/news/home/20230309005814/en/Wonder-Dynamics-Officially-Launches-Wonder-Studio-a-First-of-Its-Kind-AI-Tool-for-the-Film-and-TV-Industry>.
- [30] *Wonder Studio*. Ultimo accesso: 29 Settembre 2023. URL: <https://help.wonderdynamics.com/intro-to-wonder-studio/introduction>.
- [31] *Wonder Studio documentation*. Ultimo accesso: 29 Settembre 2023. URL: <https://help.wonderdynamics.com/working-with-wonder-studio/getting-started>.
- [32] Autori di HealthTech360. «Avatar: dalla fantascienza alla Medicina». In: *Health-Tech360* (2023). Ultimo accesso: 29 Settembre 2023. URL: <https://www.healthtech360.it/salute-digitale/intelligenza-artificiale/avatar-medicina-motion-capture/>.

- [33] Electronic Arts Inc. «Enhanced animation generation based on video with local phase». In: (*Brevetto statunitense n. 20230005203*). *World Intellectual Property Organization*. (2023). Ultimo accesso: 29 Settembre 2023. URL: <https://patentscope.wipo.int/search/en/detail.jsf?docId=US383823772>.
- [34] SSL Research & Development Office of Energy Efficiency & Renewable Energy. «OLED Basics: Solid-State Lighting». In: *Office of ENERGY EFFICIENCY & RENEWABLE ENERGY* (). Ultimo accesso: 03 Ottobre 2023. URL: <https://www.energy.gov/eere/ssl/oled-basics>.
- [35] Blender Documentation Team licensed under a CC-BY-SA v4.0. *Blender 4.0 Manual*. Last accessed 16 September 2023. 2023. URL: <https://docs.blender.org/manual/en/dev/index.html>.
- [36] Copyright 2023 ©CLO Virtual Fashion Inc. All Rights Reserved. *Marvelous Designer*. Last accessed 16 September 2023. 2023. URL: <https://www.marvelousdesigner.com/>.
- [37] Copyright ©2023 Adobe. All rights reserved. *Substance 3D Painter*. Last accessed 16 September 2023. 2023. URL: <https://www.adobe.com/products/substance3d-painter.html>.
- [38] Inc. All rights reserved ©2004-2023 Epic Games. *Unreal Engine 5*. Last accessed 16 September 2023. 2023. URL: <https://www.unrealengine.com/en-US/unreal-engine-5>.
- [39] T. Editors of Encyclopaedia Britannica. *Encyclopedia Britannica degree of freedom*. Last accessed 17 September 2023. 2018. URL: <https://www.britannica.com/science/degree-of-freedom-mathematics-and-statistics>.
- [40] *Free 3D*. Ultimo accesso: 28 Agosto 2023. URL: <https://free3d.com/it/>.
- [41] *Sketchfab*. Ultimo accesso: 28 Agosto 2023. URL: <https://sketchfab.com/>.
- [42] *VolumeCalculator*. Ultimo accesso: 28 Agosto 2023. URL: <https://www.move.ai/mobile-calculator>.