

POLITECNICO DI TORINO

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

Tesi di Laurea Magistrale

Retopology di Modelli 3D

Studio Applicativo per il Cultural Heritage



Relatore

prof. Riccardo Antonio Silvio Antonino

Candidato

Giuseppe Garone

Dicembre 2017

Sommario

Nell'ambito del Cultural Heritage, la conservazione dei beni ha assunto una nuova forma grazie al processo di digitalizzazione. Da un lato, questo processo consente di generare una versione binaria del modello le cui caratteristiche non mutano con il passare del tempo. Dall'altro lato, pone l'accento sugli aspetti tecnici e le problematiche riscontrate non solo durante il processo di digitalizzazione, quanto anche a posteriori di tale processo. Le moderne tecnologie di scansione garantiscono piena corrispondenza tra il modello fisico e la sua versione digitale, aspetto prioritario in questo ambito. Tuttavia, per supportare questa caratteristica viene richiesta una quantità considerevole di risorse in termini di memoria, connessione ed elaborazione. Vista l'enorme quantità di dati da gestire, risulta problematico impiegare tali modelli in contesti interattivi. In questa circostanza prende posizione il processo di *retopology*. All'atto pratico, questa operazione consiste nel ricostruire il modello scansionato creandone una nuova versione che rispecchia determinate caratteristiche. Innanzitutto, il principale obiettivo è la riduzione del peso poligonale del modello, il quale consente di ridurre sensibilmente la quantità di risorse richieste. Questa caratteristica è fondamentale nei contesti real-time, dove i tempi di elaborazione concessi sono molto ristretti. Secondariamente, vengono curati anche gli aspetti topologici della nuova mesh poligonale, ovvero la disposizione degli elementi che la compongono in modo tale da rendere il modello adatto alle successive fasi di lavorazione. Questo secondo aspetto è importante per la fase di texture mapping, la quale consente di ricorrere alle moderne tecniche image-based per aumentare la resa finale del modello. L'operazione di decimazione, spesso eseguita a valle del processo di scansione, si preoccupa del primo aspetto, ma trascura del tutto il secondo. La *retopology* è una fase di lavorazione diventata ormai standard in molte pipeline di produzione nell'industria dell'intrattenimento, ma in generale è necessaria in tutti quei casi in cui il modello a disposizione risulti non adatto alle finalità del progetto. Dunque, è un vero e proprio processo di ottimizzazione globale del modello.

La presente tesi è incentrata sulle metodologie che consentono di eseguire *retopology* di un modello tridimensionale, applicandole ad un caso concreto. Questo processo è considerato come uno dei task più onerosi nell'ambito della modellazione tridimensionale, poiché richiede molte abilità e impegno da parte del modellatore. Tuttavia, nel corso degli ultimi anni sono state sviluppate diverse alternative in grado

di automatizzare, parzialmente o interamente, il processo.

Il lavoro di tesi si è articolato in due fasi. In una prima fase, è stato eseguito un ampio studio delle tecniche maggiormente impiegate e degli approcci più innovativi. Questo studio è stato effettuato con un tono critico, facendo emergere l'antitesi presente tra il livello di automatizzazione e la qualità dei modelli creati. Viene proposta una possibile catalogazione dei metodi analizzati, basata sull'entità geometrica con cui il modellatore si interfaccia durante la costruzione della mesh.

La seconda fase è invece applicativa. Vengono adoperate le diverse tecniche analizzate ad un caso concreto, quale la retopology delle statue Sekhmet, scansionate per volontà del Museo Egizio di Torino. Sono stati utilizzati diversi softwares, ciascuno dei quali offre strumenti differenti per eseguire l'operazione. Infine, particolare attenzione è stata riservata alle moderne tecniche image-based in grado di rendere il modello retopologizzato quanto più verosimile al modello scansionato. I modelli finali risultano visivamente e percettivamente del tutto analoghi ai reperti scansionati, pur richiedendo decisamente meno risorse per poter essere elaborati. Ciò consente di visualizzare e interagire con i modelli anche attraverso dispositivi con risorse hardware più limitate. Ogni aspetto critico incontrato durante le varie fasi di lavorazione è stato affrontato identificando le possibili soluzioni adatte ad assicurare un ottimo trade-off tra resa grafica e richiesta di risorse computazionali. Segue una breve descrizione del contenuto dei singoli capitoli, al fine di far intendere il percorso logico adottato.

Il capitolo 1 assume il ruolo di stato dell'arte, in cui vengono esposti i concetti fondamentali relativi alle mesh poligonali. Introduce una serie di termini tecnici indispensabili alla comprensione della restante parte dell'elaborato. Una trattazione esaustiva su questo argomento così vasto va oltre gli scopi di questa tesi, pertanto il focus è rivolto solamente agli aspetti direttamente coinvolti nella parte progettuale.

Il capitolo 2 espone gli aspetti topologici riguardanti le mesh poligonali. Viene presentato un insieme di caratteristiche proprie delle mesh quadrangolari, le più diffuse in qualsiasi campo della computer grafica. I concetti discussi vengono utilizzati come parametri per valutare i risultati ottenuti nella fase progettuale.

Il capitolo 3 è incentrato sul processo di retopology vero e proprio, fornendone una definizione e mostrando i contesti in cui tale operazione risulta necessaria. Viene esposta una possibile catalogazione delle varie tecniche disponibili, mettendo in rilievo gli aspetti critici connessi alla topologia e al peso poligonale del modello generato. Viene inoltre approfondito il processo di baking, con il quale vengono create particolari textures atte a simulare gli effetti di rilievo e i dettagli più fini sulla superficie del modello retopologizzato.

Nel capitolo 4 vengono presentati una serie di softwares con cui è possibile eseguire l'operazione di retopology. Non sono stati analizzati tutti i softwares disponibili sul mercato, ma soltanto i programmi e i tools che propongono tecniche originali oppure approcci alternativi alla retopology, al fine di poter supportare la catalogazione

esposta al capitolo 3. D'altronde, molti softwares riproducono in chiave leggermente diversa le tecniche già implementate in altri softwares. I programmi analizzati sono stati impiegati nella fase progettuale della tesi.

Il capitolo 5 si occupa di presentare il progetto di retopology eseguito sulle statue Sekhmet. Dopo aver contestualizzato il progetto, vengono dettagliate le fasi principali dell'intero lavoro, mettendo in rilievo le criticità affrontate e motivando le scelte adottate. Vengono eseguiti diversi esperimenti impiegando gli approcci esposti al capitolo 3 e gli strumenti presentati al capitolo 4, valutandone di ognuno l'idoneità al caso delle statue. L'obiettivo è quello di creare modelli digitali da inserire in un'applicazione interattiva.

I risultati ottenuti in termini di peso poligonale e resa dei modelli vengono mostrati nel capitolo 6. Viene mostrato anche come i modelli creati siano realmente adatti alla visualizzazione tramite browser web sia su PC che su mobile.

Infine, il capitolo 7 espone gli sviluppi futuri in cui adoperare le tecniche e le metodologie adottate nell'ambito di questo progetto.

Ringraziamenti

Desidero innanzitutto ringraziare il prof. Riccardo Antonino, nonché mio relatore, per la disponibilità e, soprattutto, l'interesse mostrato verso l'argomento trattato. Un particolare grazie per aver posto fiducia nelle mie capacità.

Ringrazio inoltre il personale del Museo Egizio di Torino per la partecipazione ed il coinvolgimento in tale iniziativa, avendomi fornito il materiale con il quale è stato svolto il progetto.

Indice

Elenco delle figure	VIII
1 Modelli tridimensionali	1
1.1 Rappresentazione di oggetti tridimensionali	1
1.2 Generazione di superfici tridimensionali	1
1.3 Mesh poligonali	3
1.3.1 Triangoli e quadrangoli	4
1.4 Superfici di suddivisione	5
1.4.1 Catmull-Clark	7
1.4.2 Doo-Sabin	7
1.4.3 Loop	8
1.5 Texturing di modelli tridimensionali	8
1.5.1 Tipologie e creazione di textures	9
1.5.2 UV-Mapping	9
1.5.3 Distorsioni e Padding	12
2 Aspetti topologici	13
2.1 Concetto di topologia	13
2.2 Mesh quadrangolari e anisotropia	14
2.3 Valenza e Singolarità	15
2.4 Edge-flow ed Edge-loops	17
2.5 Risoluzione variabile	18
3 Retopology: motivazioni, contesti e metodologie	20
3.1 Definizione e motivazioni	20
3.2 Retopology nel Cultural Heritage	21
3.2.1 Scansione 3D e Reverse modeling	22
3.2.2 Problematiche dei modelli acquisiti	24
3.3 Retopology per l'Entertainment	25
3.3.1 Sculpt 3D	25
3.3.2 Problematiche dei modelli scolpiti	27
3.4 Tecniche e approcci	27

3.4.1	Approcci tradizionali	28
3.4.2	Approcci ibridi	29
3.4.3	Remeshing automatico	30
3.5	Baking dei dettagli	31
3.5.1	Normali e Normal Map	32
3.5.2	Displacement Map	33
4	Strumenti per eseguire Retopology	36
4.1	Panoramica	36
4.2	Blender	37
4.2.1	Snap tool & Shrinkwrap	37
4.2.2	Bsurfaces e SpeedRetopo add-ons	38
4.3	Autodesk Maya	40
4.4	Pilgway 3D-Coat	40
4.5	SketchRetopo	42
4.5.1	Creazione di una patch	42
4.5.2	Codifica del sistema di curve	43
4.5.3	Pattern topologici	43
4.5.4	Modalità di editing	43
4.6	Instant Meshes	44
5	Retopology delle statue Sekhmet	46
5.1	Contestualizzazione	46
5.1.1	Librerie digitali e 3D repositories	46
5.1.2	Scansione di beni tridimensionali	47
5.1.3	Decimazione	48
5.2	Fase di scansione	50
5.2.1	Scansione laser con luce strutturata	51
5.2.2	PDF interattivi	52
5.3	Fase di Retopology	53
5.3.1	Considerazioni iniziali	54
5.3.2	Retopology manuale	55
5.3.3	Approccio a patch	56
5.3.4	Retopology automatica	57
5.3.5	Ottimizzazione dei modelli	59
5.4	Texture Mapping e Baking	61
5.4.1	UV-Mapping	62
5.4.2	Operazione di Baking	63

6	Discussione dei risultati	68
6.1	Considerazioni sui modelli	68
6.2	Riduzione del peso poligonale	68
6.3	Qualità della topologia	70
6.4	Valutazione dei tempi di produzione	70
6.5	Output finali e test	71
7	Conclusioni e sviluppi futuri	76
	Bibliografia	78

Elenco delle figure

1.1	Sfera generata tramite curve parametriche. I punti evidenziati sono i vertici di controllo, mentre le intersezioni tra le curve generano le patches (a). La stessa sfera può essere rappresentata tramite una mesh poligonale, i cui vertici, lati e facce sono direttamente accessibili (b).	3
1.2	Mesh poligonale con tassellatura quadrangolare (a). Mesh poligonale con tassellatura triangolare (b).	5
1.3	La mesh iniziale (a) diventa una gabbia di controllo della superficie di suddivisione (b), i cui vertici possono deformare porzioni più o meno estese della nuova mesh (c). Superficie suddivisa (d).	6
1.4	Applicazioni successive dello schema di suddivisione di Catmull-Clark. La superficie preserva una tassellatura quadrangolare.	7
1.5	Applicazioni successive dello schema di suddivisione di Doo-Sabin. Oltre a quadrangoli, la superficie suddivisa contiene anche triangoli e N-gons.	7
1.6	Applicazioni successive dello schema di suddivisione Loop, ottimizzato per mesh triangolari.	8
1.7	Esempi di texture maps. La diffuse map (a) contiene unicamente l'informazione cromatica. L'ambient occlusion map (b) determina quali parti della texture restano più in ombra a causa del fenomeno dell'occlusione ambientale. La normal map (c) viene utilizzata per simulare effetti di rilievo sulla superficie. La specular map (d) definisce le parti meno lucide e quelle più lucide.	10
1.8	Fasi del texture mapping. I seams (a) scompongono la mesh in shells, le quali vengono individualmente proiettate all'interno del texture space (b). Viene creata la texture (c) che poi verrà proiettata sul modello (d).	11
1.9	Inevitabilmente, la proiezione nello spazio (u,v) crea distorsioni. Per misurare l'entità delle distorsioni, viene adottata una scala di colori, visibile sia sul modello (a), sia sulla mappatura UV (b). Più la proiezione è isometrica, minori saranno le distorsioni.	12

2.1	Nel caso di volti umani, una topologia corretta (a) prevede che le linee della mesh seguano l'andamento delle fibre muscolari (b). Questa caratteristica è evidente soprattutto nelle zone di labbra e palpebre. Fonti: [22] (a) e [26] (b).	14
2.2	Quadrangolazione anisotropica. Localmente, è possibile individuare le due principali direzioni di curvatura. In rosso è evidenziata la direzione con il grado di curvatura maggiore, mentre in verde quella col grado di curvatura minore (a). I gradi di curvatura determinano le dimensioni dei lati dei quadrangoli: laddove la curvatura è maggiore, i lati saranno più corti, e viceversa (b).	15
2.3	In (a) sono evidenziate le singolarità con valenza tre (in verde) e cinque (in rosso). Generalmente, valenze di questo tipo non creano artefatti pesantemente visibili. Anzi, sono indispensabili per garantire un buon flusso di lati. Invece, valenze esageratamente elevate causano artefatti facilmente visibili quando si suddivide la mesh (b).	16
2.4	In (a) vi sono due esempi di edge-loops. In verde, una sequenza di lati chiusa forma un edge-loop, mentre la sequenza in rosso è un edge-loop interrotto da due singolarità. In (b) è visualizzato il concetto di edge-flow, caratteristica in parte dipendente dalle singolarità. Soprattutto nei modelli organici, questo aspetto è di fondamentale importanza. Fonte modello: [22].	18
3.1	Con il processo di retopology, la mesh mostrata in (a) è stata ricostruita fino a diventare la mesh mostrata in (b). Si può notare come i due modelli abbiano la stessa forma, ma il secondo contiene molti meno poligoni e presenta caratteristiche topologiche migliori.	21
3.2	Reverse modeling: pipeline di creazione di modelli digitali nell'ambito dell'heritage culturale.	22
3.3	Applicando le fasi descritte in 3.2.1, dalla point cloud si costruisce la versione digitale dell'oggetto scannerizzato. Si può notare come il modello ottenuto sia particolarmente denso di triangoli, oltre a presentare un flusso di lati del tutto caotico. La mesh risultante è dunque destrutturata, prestandosi poco a successive lavorazioni.	24
3.4	Generica pipeline di creazione di modelli digitali nell'industria dell'intrattenimento. In base al contesto, può prevedere ulteriori step.	26
3.5	Partendo da una semplice mesh, la forma desiderata viene modellata tramite il pennello digitale.	26
3.6	La retopology eseguita con tecniche tradizionali prevede la creazione e gestione dei vertici che costituiranno la mesh finale.	29
3.7	Con un approccio a patch, non si agisce direttamente sui vertici, bensì su entità geometriche superiori, quali le patches.	30

3.8	Complessi algoritmi di remeshing automatico sono in grado di produrre mesh poligonali richiedendo un limitato intervento al modellatore.	31
3.9	In blu sono indicate le normali ai vertici di una mesh poligonale (a). Per ogni vertice, i vettori normale, bitangente e tangente formano la base vettoriale del <i>tangent-space normal</i> (b). I valori di questi vettori vengono immagazzinati rispettivamente nei canali blu, verde e rosso della normal map.	33
3.10	Nel <i>tangent-space</i> , i vettori possono avere valori compresi nell'intervallo $[-1,1]$, ad esclusione del vettore normale, con valore compreso in $[0,1]$. Invece, utilizzando 24bit per pixel, i canali (R,G,B) di ciascun pixel contengono valori compresi nell'intervallo $[0,255]$. Avviene un'operazione di normalizzazione e scalamento, in cui l'intervallo $[-1,1]$ dei vettori viene normalizzato nell'intervallo $[0,255]$ dei canali del pixel. Pertanto, il vettore normale $(0,0,1)$ viene rappresentato dal pixel $(127,127,255)$, che corrisponde ad un violetto. Ciò indica che la normale al vertice non deve essere modificata. Tonalità tendenti al rosso indicano valori elevati del vettore tangente, mentre tendenti al verde valori elevati della binormale.	34
3.11	La displacement map si presenta come una texture in scala di grigi (a), mentre la normal map come una texture tendente al viola (c). La displacement map modifica la mesh, traslando i vertici e creando effettivamente dei rilievi (b). Invece, la normal map si limita a simulare il rilievo modificando solamente i valori delle normali e, di conseguenza, anche l'illuminazione (d). Tuttavia, la displacement map esige più geometria per poter offrire una resa migliore.	35
4.1	A partire da un piano (a), il modello viene retopologizzato tramite estrusione (b). Con questa tecnica, la retopology si traduce in una modellazione poligonale classica.	38
4.2	L'add-on <i>Bsurfaces</i> consente di disegnare una serie di curve (a), in base alle quali viene aggiunta una superficie poligonale (b). Con il plugin <i>SpeedRetopo</i> , viene aggiunta la funzione <i>Retopo MT</i> (c), che genera letteralmente una striscia di quadrangoli (d).	39
4.3	Dopo aver posizionato manualmente un certo numero di vertici (a), il sistema individua automaticamente i possibili quadrangoli che possono essere generati (b). Non è possibile generare triangoli, se non in una seconda fase di rifinitura.	40
4.4	Dopo aver marcato le zone più sensibili (in nero) e disegnato un set di curve sulla superficie (a), <i>Autopo</i> genera automaticamente una mesh quadrangolare (b).	41

4.5	La patch viene automaticamente creata e quadrangolata dal sistema (a), ma il modellatore può modificarla impostando le suddivisioni ai lati della patch (b). Le singolarità vengono automaticamente inserite.	42
4.6	L'algoritmo inizia con il calcolo dell'orientation field, visualizzato tramite una serie di linee colorate sulla mesh (a). Successivamente, calcola il position field (b), dal quale vengono poi estratti i vertici.	45
5.1	Le possibilità offerte dal digitale consentono di sviluppare nuovi metodi di fruizione dei beni. Con l'iniziativa <i>Google Arts & Culture</i> , si propone un'interazione basata sull'esplorazione, caratteristica tipica dei videogiochi open world (a). <i>African Fossils</i> invece consente di osservare un reperto e di interagire con esso potendolo ruotare e ingrandire (b).	47
5.2	Modello ottenuto a seguito della decimazione. sebbene il numero di triangoli sia drasticamente diminuito, la topologia risulta del tutto disordinata. Ciò rende impossibile l'operazione di UV-Mapping.	49
5.3	A seguito del processo di decimazione, non è più possibile riutilizzare la texture iniziale. L'assenza di molte facce nel texture space (a) causa vistose distorsioni sul modello (b).	50
5.4	Workflow con le fasi fondamentali del lavoro eseguito sulle statue Sekhmet.	50
5.5	Sistema di triangolazione per la scansione tramite scanner laser a luce strutturata.	52
5.6	Visualizzazione di una statua scansionata all'interno del .PDF interattivo. Oltre alla geometria, è possibile visualizzare anche la texture contenente le informazioni cromatiche.	53
5.7	Da ogni .PDF sono state estratte sia le informazioni geometriche che quelle cromatiche. Ciò ha permesso di utilizzare i modelli in software di grafica 3D più tradizionale, come <i>Blender</i> TM .	55
5.8	Tracciare delle linee a mano libera aiuta a definire le caratteristiche topologiche della mesh finale. Nelle parti organiche, questa fase iniziale è stata molto utile. Le singolarità nel volto (a) e nella mano chiusa (b) consentono di distribuire i lati in modo tale da creare un edge-flow quanto più ordinato e funzionale per la fase di texture mapping.	56
5.9	L'approccio a patch si rivela adatto anche alle parti organiche dei modelli, nelle quali i requisiti topologici (singolarità ed edge-loops) sono più stringenti (a). Tuttavia, spesso è necessario aggiungere patch veramente piccole, al fine di assicurare un edge-flow equilibrato. In presenza di danni, il sistema a curve necessita di ulteriori lavorazioni (b). Si nota facilmente la caratteristica anisotropica della mesh generata.	58

5.10	Le zone più sensibili sono quelle organiche. È necessario quindi inserire molte curve in corrispondenza di tali parti. <i>3D-Coat</i> [®] richiede di tracciare le curve prima di generare la mesh (a), utilizzate per allineare i quadrangoli (b). Invece, <i>Instant Meshes</i> offre una modalità interattiva real-time di editing (c). Sebbene gli sforzi richiesti al modellatore siano minimi, i risultati sono spesso disordinati, talvolta anche nelle zone geometriche.	60
5.11	Anomalie delle mesh esportate da <i>SketchRetopo</i> . I vertici multipli (a) e le facce doppie (b) rendono inutilizzabile la mesh per la fase di uv-mapping.	61
5.12	Non sempre <i>Instant Meshes</i> restituisce mesh conformi. Gruppi di quadrangoli possono essere staccati dalla mesh principale (a). È invece più frequente ottenere distorsioni con <i>Autopo</i> (b).	62
5.13	La topologia influenza notevolmente i risultati ottenibili nella fase di UV-Mapping. Una topologia funzionale consente di individuare facilmente i seams, garantendo poche distorsioni ed un layout UV ordinato (a). Viceversa, una topologia caotica causa estese distorsioni, mentre l'utilizzo di una mappatura UV automatica genera un layout UV caotico (b).	64
5.14	Al fine di migliorare la resa finale dei modelli, ad ogni texture è stato aggiunto del padding. Senza il padding (a) sono presenti discontinuità visibili lungo i seams (b). L'aggiunta di padding (c) nasconde questi artefatti (d).	65
5.15	Calcolando la normal map è possibile simulare i dettagli persi, dando l'illusione che questi siano presenti poligonalmente sul modello. Essendo molte statue colme di danni, la normal map consente di rappresentarli senza aggiungere geometria al modello.	66
5.16	Esempio di modello finale (a), con normal map (b) e con diffuse map (c).	67

6.1	La quadrangolazione isotropica di <i>Instant Meshes</i> costringe ad impostare un numero elevato di poligoni finali, al fine di poter rappresentare con maggiore precisione le zone più complesse (a) e (b). Sebbene sia fondamentale assicurare corrispondenza col modello scansionato, il peso poligonale ne risente negativamente. Pertanto, in questo contesto i metodi automatici non consentono di raggiungere un trade-off accettabile. I metodi manuali e patch-based (c) consentono di avere pieno controllo sia sulla topologia, sia sul numero di poligoni. La topologia è fondamentale per riprodurre con maggiore fedeltà le zone più difficili come quelle dei piedi, e gioca un ruolo importante per la fase di texture mapping. Difficilmente i metodi automatici producono risultati analoghi, tuttavia algoritmi come <i>Autopo</i> (d) considerano maggiormente l'aspetto topologico del modello, seppur in modo limitato.	72
6.2	Il modello a sinistra è stato creato ricorrendo alle tecniche manuali discusse in 4.2.2, mentre quello a destra utilizzando <i>SketchRetopo</i> . Entrambi presentano un edge-flow ordinato ed una buona distribuzione dei poligoni, sia nelle parti organiche che in quelle più geometriche. Ciò rende <i>SketchRetopo</i> un tool in grado di raggiungere risultati del tutto paragonabili a quelli ottenuti via retopology manuale classica.	73
6.3	Gli algoritmi di remeshing automatico non sono in grado di produrre modelli con un flusso di lati pulito, nonostante il tracciamento di molte curve per l'orientamento dei quadrangoli. La totale assenza di edge-loops e la presenza di molte singolarità rendono difficile la fase di texture mapping. Ciò vale principalmente per le zone organiche delle statue, dove i quadrangoli risultano pesantemente distorti, mentre per quelle geometriche i risultati sono più accettabili.	74
6.4	Visualizzazione tramite browser su PC (a) e su smartphone (b).	75

Capitolo 1

Modelli tridimensionali

1.1 Rappresentazione di oggetti tridimensionali

Nella realtà è molto intuitivo percepire un oggetto in base al volume che occupa nello spazio. Nella Computer grafica 3D si preferisce invece ricorrere ad una rappresentazione della sua *superficie*. Questa priorità è dovuta al fatto che la maggior parte degli algoritmi di grafica computazionale ad oggi impiegati operano attraverso tale rappresentazione. Anche i modelli di illuminazione agiscono sulla base dell'interazione tra luce e superficie. Qualunque software di grafica 3D è in grado di creare e manipolare le superfici tridimensionali, mettendo a disposizione un vasto set di strumenti sempre più comodi e veloci da usare.

Contemporaneamente, anche l'hardware dei calcolatori si è evoluto lungo tale direzione: i moderni sistemi grafici si basano su una stretta cooperazione tra *CPU* (*Central Processing Unit*) e *GPU* (*Graphics Processing Unit*). La CPU si occupa di eseguire l'applicativo grafico, mentre la GPU ha il compito di calcolare come mostrare a video le superfici, valutando fenomeni come occlusioni e ombreggiature. La GPU si occupa anche delle trasformazioni geometriche applicate alle superfici, come proiezioni, traslazioni e riscalamenti. Essendo queste trasformazioni delle moltiplicazioni tra matrici, le GPU sono state ottimizzate per eseguire efficientemente tali calcoli, al punto da essere definite processori vettoriali. La rappresentazione tramite il volume trova un impiego molto più ristretto, ad esempio quello delle simulazioni fisiche di fluidi e gas.

1.2 Generazione di superfici tridimensionali

Generare la superficie tridimensionale di un oggetto può rivelarsi un processo tanto complicato quanto tedioso. Nel corso degli anni sono state sviluppate molte tecniche di modellazione basate su differenti approcci. Alcune sono nate per soddisfare esigenze in contesti ben precisi, mentre altre trovano impiego in più settori. È quindi

fondamentale capire quale tecnica si addice di più nel contesto in cui andiamo ad operare. Ad esempio, la *Modellazione solida* viene preferita nella prototipazione di componenti meccanici, poiché il modello viene creato attraverso una serie di operazioni comparabili a quelle da effettuare nella realtà, quali smussature, forature ed estrusioni. Lo *Sculpt 3D* viene largamente utilizzato nella creazione di characters, poiché permette di creare forme organiche complesse in modo più naturale e intuitivo. Sono state sviluppate anche tecniche che creano superfici tramite l'impiego di curve parametriche. È il caso delle *Superfici spline*, adoperate soprattutto nella modellazione di carrozzerie di automobili ed oggetti di design, poiché consentono di creare superfici complesse completamente lisce e smussate. Infine, la *Scansione 3D* permette di creare modelli tridimensionali a partire direttamente da oggetti del mondo reale.

Nonostante la grande varietà di tecniche disponibili, gli output ottenibili possono essere raggruppati in due grandi famiglie:

- *Mesh poligonali*: superfici formate da un insieme di superfici piane poligonali più semplici connesse l'una con l'altra.
- *Superfici parametriche*: superfici ottenute attraverso la congiunzione di più curve polinomiali parametriche. L'unione di tre o quattro curve genera una superficie chiusa detta *patch*. Tipicamente, l'oggetto finale è costituito da un collage di più patches.

Nel primo caso abbiamo pieno accesso e controllo ad ogni singolo elemento che costituisce la maglia di poligoni, quali vertici, lati e facce, mentre nel secondo gli unici elementi disponibili sono pochi vertici che ne controllano l'andamento generale. Entrambe le soluzioni comportano vantaggi e svantaggi. Le superfici parametriche richiedono meno spazio di memorizzazione, poiché bastano pochi coefficienti per rappresentare una curva con un andamento piuttosto complesso, e semplificano alcune operazioni di manipolazione globale del modello. Ma i costi di elaborazione aumentano, soprattutto se si adottano curve di grado superiore al terzo. Inoltre, non sono direttamente gestibili dalle schede grafiche, che si occupano di convertirle in mesh poligonali prima di rasterizzarle a video. In contrasto, le mesh poligonali richiedono più memoria, ma sono maggiormente utilizzate poiché adatte praticamente in tutti i settori della computer grafica: scansione 3D, ingegneria, CAD, automotive, film, videogiochi. Software come Autodesk *3D Studio Max*¹, Autodesk *Maya*² e *Blender*³ consentono di lavorare sia con mesh poligonali che con superfici parametriche. Tali programmi offrono spesso soluzioni ibride che consentono di

¹ Autodesk 3ds Max, [35]

² Autodesk Maya, [37]

³ Blender, [39]

adoperare più tecniche di modellazione, adattandosi a più esigenze. Tuttavia, le ultime tendenze hanno visto la comparsa sul mercato di software specializzati nella modellazione con una precisa tecnica. È il caso di Autodesk *Mudbox*⁴ e Pixologic *ZBrush*⁵, ottimizzati per la modellazione tramite sculpting 3D, oppure di Autodesk *Alias*⁶, indirizzato verso la modellazione tramite curve.

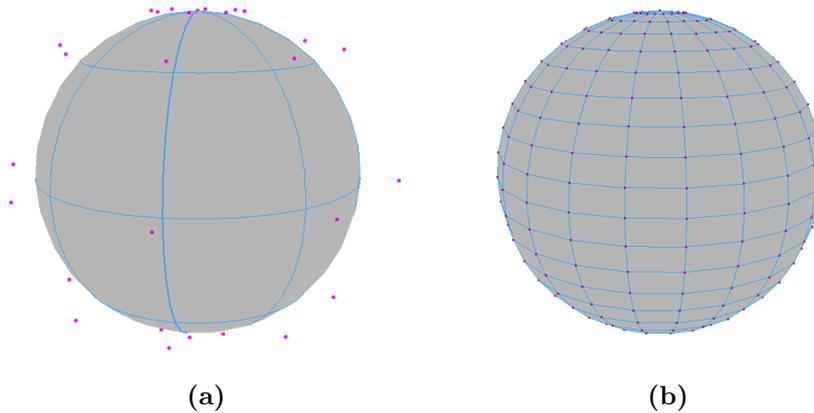


Figura 1.1: Sfera generata tramite curve parametriche. I punti evidenziati sono i vertici di controllo, mentre le intersezioni tra le curve generano le patches (a). La stessa sfera può essere rappresentata tramite una mesh poligonale, i cui vertici, lati e facce sono direttamente accessibili (b).

1.3 Mesh poligonali

Una *mesh poligonale* è una collezione di vertici, lati e facce che, uniti secondo precise regole di connettività, formano un reticolo che definisce la superficie di un oggetto. Ad ogni vertice viene attribuita una tripletta di valori che ne indicano la posizione nello spazio. La connessione di due vertici identifica un lato, mentre ogni faccia viene costruita da una sequenza chiusa di lati. Precise regole di connettività tra questi elementi garantiscono la creazione di una struttura bidimensionale *conforme*. Ogni lato deve essere condiviso da al più due facce, mentre due facce non possono condividere un vertice senza che condividano i propri lati con altre facce. Una mesh che non rispetta tali vincoli è chiamata mesh *non-manifold*. Oltre a rappresentare

⁴ Autodesk Mudbox, [38]

⁵ Pixologic ZBrush, [46]

⁶ Autodesk Alias, [36]

oggetti non realmente esistenti, queste mesh causano artefatti visivi durante le fasi di rendering e animazione, nonché errori durante simulazioni fisiche o stampa 3D.

Una mesh poligonale arriva facilmente a contenere centinaia di migliaia di elementi. Dunque, è importante capire come rappresentare queste informazioni in strutture dati che permettano minori tempi di elaborazione e minore spazio occupato in memoria. In generale, più vengono sfruttate le relazioni tra vertici, lati e facce, minore è il tempo di elaborazione, ma maggiore è lo spazio occupato. Le rappresentazioni più comuni sono:

- *Vertex-vertex mesh*: è il metodo di rappresentazione più grezzo, poiché consiste nel rappresentare il modello tramite una semplice lista di vertici. Ogni vertice è individuato dalle proprie coordinate, e vengono ignorate le relazioni tra essi che permettono di identificare facce e lati.
- *Face-vertex mesh*: si sfruttano le relazioni esistenti tra facce e vertici. È presente una lista di vertici, identificati dalle rispettive coordinate, ed una lista di facce che puntano ad essi. Ogni faccia è quindi definita da una lista di quattro puntatori alla lista dei vertici.
- *Winged-edge mesh*: vengono create una lista di vertici, una di lati ed una di facce. Ogni faccia della lista di facce è rappresentata mediante puntatori alla lista di lati, mentre ogni lato della lista di lati mediante puntatori alla lista di vertici. È la rappresentazione più esplicita possibile.

Vista la natura discreta delle mesh poligonali, qualunque oggetto tridimensionale viene rappresentato con un certo grado di approssimazione. Gli oggetti con superfici piane vengono riprodotti correttamente, mentre per gli oggetti curvi una maggiore affidabilità si paga con un aumento di facce e vertici, nonché di calcoli e memoria.

1.3.1 Triangoli e quadrangoli

Le mesh poligonali sono basate sul concetto di *cell decomposition*: un oggetto complesso viene rappresentato da un insieme di elementi geometrici (*celle*) più semplici. Le figure geometriche maggiormente impiegate sono i triangoli ed i quadrangoli⁷. Nel primo caso otteniamo una *mesh triangolare*, mentre nel secondo una *mesh quadrangolare*. In figura 1.2 sono mostrate entrambe le tipologie. Mesh costituite da elementi diversi da triangoli e quadrangoli sono un'eccezione, e vengono trattate in modo da diventare triangolari o quadrangolari.

Il triangolo è la figura geometrica più semplice esistente, e presenta proprietà molto utili. È sempre planare e convesso, e questo semplifica il calcolo del vettore

⁷ [1], p. 1

normale alla superficie, fondamentale durante i calcoli relativi all'interazione tra luce e oggetto. Inoltre, per calcolare i valori di colore e ombreggiatura su tutta la superficie del triangolo, è possibile utilizzare una semplice interpolazione lineare dei valori presenti ai vertici sfruttando le coordinate baricentriche [49]. Inoltre, l'algoritmo di rasterizzazione dei triangoli è veloce ed eseguito direttamente via hardware. In contrasto, il quadrilatero non è sempre planare né convesso. La non planarità rende meno immediato il calcolo del vettore normale, mentre la non convessità non consente di applicare relazioni lineari. Di conseguenza, anche la rasterizzazione a video richiede più tempo. In virtù di queste caratteristiche, le mesh triangolari sono preferite dagli sviluppatori, che hanno dedicato loro molte ricerche di geometria computazionale. In letteratura sono disponibili molti algoritmi di triangolazione, atti a trasformare una qualunque superficie in un insieme finito di triangoli. Nonostante i notevoli vantaggi offerti dalle mesh triangolari, all'atto pratico vengono maggiormente adoperate le mesh quadrangolari, come verrà discusso nel paragrafo 2.2. Attività come il texture mapping, l'animazione o l'aggiunta di dettagli vengono di fatto semplificate se si lavora con una tassellatura quadrangolare piuttosto che triangolare.

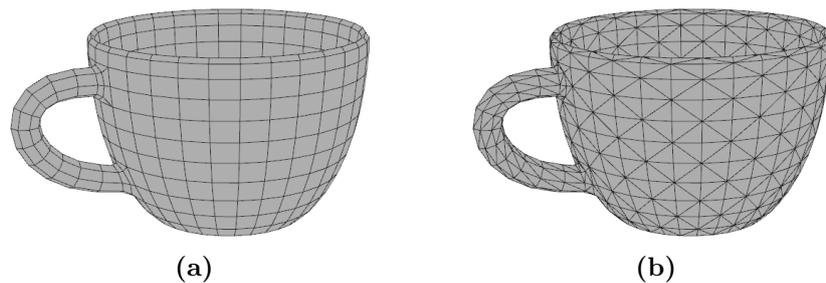


Figura 1.2: Mesh poligonale con tassellatura quadrangolare (a). Mesh poligonale con tassellatura triangolare (b).

Tuttavia, ancora oggi le mesh triangolari sono imprescindibili. Le schede grafiche sono in grado di processare solamente triangoli, e qualunque mesh viene convertita in un insieme di triangoli prima di essere rasterizzata a video. Anche nel contesto delle scansioni 3D, gli algoritmi di meshing esistenti sono in grado di generare solamente un insieme di triangoli. In conclusione, si può affermare che le mesh triangolari sono più comode da utilizzare per gli sviluppatori, mentre le mesh quadrangolari sono più versatili per gli artisti.

1.4 Superfici di suddivisione

L'idea delle *Superfici di suddivisione* è quella di poter generare superfici levigate, tipiche della modellazione con curve parametriche, a partire da mesh poligonali.

Applicando un preciso schema di suddivisione, la mesh iniziale viene approssimata da una superficie di tipo B-spline. Le B-spline sono curve realizzate tramite la congiunzione di più Curve di Bézier. La curva ottenuta è continua di tipo C^2 , ovvero nei punti di giunzione i vertici hanno la stessa tangente e la stessa curvatura. Nel caso fosse soddisfatta solamente la prima condizione, la continuità è di tipo C^1 . La mesh iniziale viene utilizzata come una gabbia di controllo, i cui vertici agiscono come punti di controllo, oltre che punti interpolanti, della superficie suddivisa. Quando le regole di suddivisione vengono applicate un numero infinito di volte, si ottiene una superficie limite, oltre il quale ulteriori suddivisioni non producono risultati notevoli.

In letteratura sono stati sviluppati una gran varietà di schemi di suddivisione, ed un'esaustiva esposizione delle loro caratteristiche è disponibile in [18]. Tuttavia, è bene precisare che gli schemi più utilizzati sono approssimanti, uniformi e stazionari. Un altro parametro di rilievo è il tipo di tassellatura restituita dallo schema. Alcuni algoritmi sono ottimizzati per lavorare con facce quadrangolari, alcuni invece con facce triangolari, mentre altri sono ibridi. Indipendentemente dallo schema adottato, le motivazioni che spingono gli artisti ad usare le superfici di suddivisione sono molteplici. Innanzitutto, vengono attenuate le sfaccettature tipiche delle mesh poligonali, incrementando la qualità della resa grafica. Inoltre, viene semplificato il processo di creazione della mesh iniziale, che può essere costituita da poche facce, risultando così più leggera e gestibile, come mostrato in figura 1.3 (a). Utilizzando questo approccio, il dettaglio finale della mesh è condizionato dal numero di applicazioni dello schema di suddivisione. Ciò consente di creare facilmente più versioni dello stesso modello che differiscono per dettaglio, utile soprattutto in contesti interattivi. Il principale svantaggio di questo metodo di modellazione consiste nell'incremento del numero di facce che compongono la superficie finale, anche se le schede grafiche moderne sono in grado di processare centinaia di migliaia di poligoni al secondo senza particolari problemi.

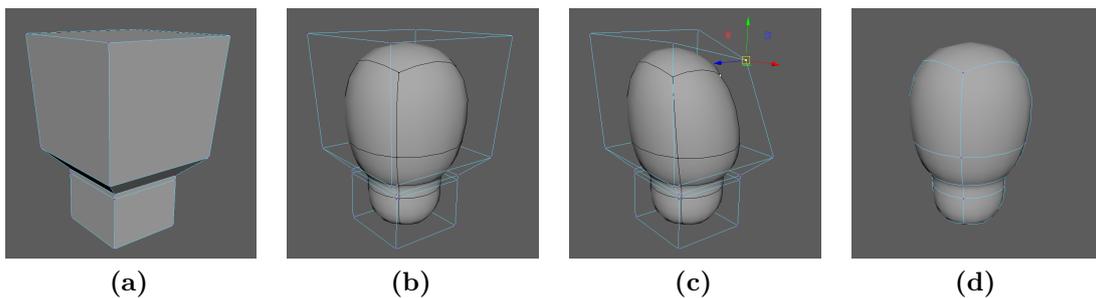


Figura 1.3: La mesh iniziale (a) diventa una gabbia di controllo della superficie di suddivisione (b), i cui vertici possono deformare porzioni più o meno estese della nuova mesh (c). Superficie suddivisa (d).

1.4.1 Catmull-Clark

L'algoritmo proposto da Edwin Catmull e Jim Clark nel 1978 in [2] è diventato lo standard de facto in tutti i campi della computer grafica. Questo schema di suddivisione è detto *primale*, poiché l'approccio adottato è quello della suddivisione delle facce. Nello specifico, prevede che ogni faccia con n lati venga suddivisa in n quadrangoli, producendo così una tassellatura uniformemente quadrangolare. La continuità C^2 è assicurata ovunque eccetto che nei vertici irregolari, dove viene garantita solamente la continuità C^1 (si faccia riferimento al paragrafo 2.3 per la nozione di singolarità).

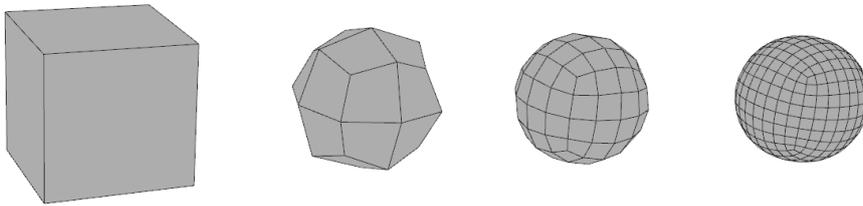


Figura 1.4: Applicazioni successive dello schema di suddivisione di Catmull-Clark. La superficie preserva una tassellatura quadrangolare.

1.4.2 Doo-Sabin

Lo schema ideato da Daniel Doo e Malcolm Sabin, esposto in [3], adotta un approccio differente da quello di Catmull e Clark: anziché suddividere le facce, duplica i vertici, e per questo è detto schema *duale*. Le mesh restituite sono composte principalmente da quadrangoli, ma è presente una certa quantità di triangoli ed N-gons (si veda il paragrafo 2.2 per la definizione di N-gon). La massima continuità raggiungibile dalla superficie limite è di tipo C^1 . A causa di tali fattori, questo algoritmo è stato messo da parte per far spazio al più efficiente Catmull-Clark.

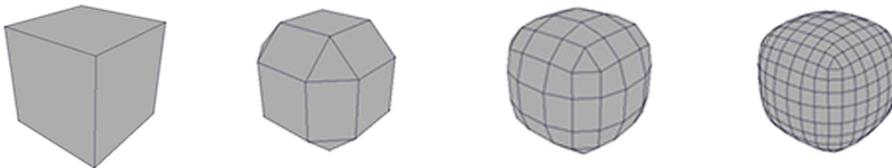


Figura 1.5: Applicazioni successive dello schema di suddivisione di Doo-Sabin. Oltre a quadrangoli, la superficie suddivisa contiene anche triangoli e N-gons.

1.4.3 Loop

A differenza dei due schemi citati, l'algoritmo studiato da Charles Loop in [10] adopera unicamente mesh triangolari. È di tipo primale: ogni faccia triangolare viene suddivisa in quattro triangoli. Come per lo schema di Catmull-Clark, la continuità della superficie limite è di tipo C^2 eccetto che nei verti irregolari. Questo algoritmo non trova molto impiego poiché una mesh triangolare non possiede caratteristiche topologiche favorevoli alle tipiche fasi di lavorazione (si rimanda al capitolo 2 per una trattazione completa sulla topologia).

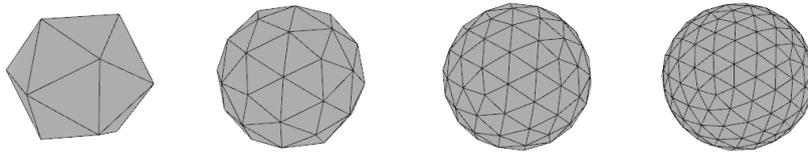


Figura 1.6: Applicazioni successive dello schema di suddivisione Loop, ottimizzato per mesh triangolari.

1.5 Texturing di modelli tridimensionali

Generare il modello tridimensionale è solamente la prima fase di molte pipeline di produzione. Tipicamente, il secondo step è il *texture mapping*, ovvero un processo che permette di corredare il modello di informazioni aggiuntive riguardanti l'aspetto esteriore. Queste informazioni possono riferirsi al colore, al rilievo o all'ombreggiatura da applicare al modello, e vengono immagazzinate sotto forma di immagini bidimensionali chiamate *textures*. Tramite un'opportuna applicazione sulla superficie del modello, la resa grafica finale viene notevolmente migliorata. Il texture mapping può essere scomposto in due fasi separate: la preparazione del modello, esposta in 1.5.2, e la creazione delle textures, discussa in 1.5.1. Queste ultime vengono poi rielaborate sulla superficie modello attraverso uno *shader*, ovvero un programma il cui compito è quello di determinare l'aspetto esteriore dell'oggetto tenendo in considerazione sia le condizioni di luce, sia le informazioni provenienti dalle textures. Si occupa di rappresentare il materiale da applicare alla superficie del modello.

Le textures rivestono un ruolo centrale in tutti i campi della computer grafica. In ambito architettonico, sono indispensabili soprattutto per riprodurre superfici non modellabili se non a costo di un massiccio impiego di facce e vertici. Nel settore dei videogiochi le textures vengono adoperate soprattutto per simulare effetti come il rilievo e l'ombreggiatura. In qualunque settore si lavori, il texturing incide in maniera considerevole sul risultato finale.

1.5.1 Tipologie e creazione di textures

In prima istanza, le textures possono essere suddivise in due grandi categorie: *procedurali* e immagini 2D. Le prime sono textures i cui valori immagazzinati variano secondo una precisa formula matematica. Vengono principalmente impiegate nella creazione di superfici con effetti casuali, come le striature del legno, le variazioni di tonalità di una pietra oppure di una lastra di marmo. Vista la loro natura matematica, sono *seamless*, ovvero ricoprono l'intera superficie del modello senza discontinuità visibili. Sono generate direttamente dal software in uso, e non richiedono immagini esterne. In virtù di queste caratteristiche, l'artista non si deve preoccupare della risoluzione finale. La principale difficoltà consiste nel sapere gestire i numerosi parametri che ne regolano la generazione. Al contrario, le immagini 2D sono vere e proprie fotografie proiettate sulla superficie del modello. Vengono acquisite tramite fotocamere digitali ed eventualmente elaborate per migliorarne la qualità. Tipicamente non sono seamless, e la resa finale del modello dipende dalla risoluzione e dal formato della fotografia. Inoltre, il loro impiego è ridotto ad un ristretto numero di situazioni.

La precedente distinzione è sicuramente valida, ma al giorno d'oggi si usa molto più frequentemente il termine *maps*. Le mappe sono textures che controllano aspetti specifici del materiale. Ogni mappa è responsabile di un fenomeno ben preciso: ad esempio, la *normal map* (dettagliata in 3.5.1) crea effetti di rilievo, la *specular map* controlla le riflessioni speculari, mentre la *diffuse map* aggiunge l'informazione cromatica (si veda la figura 1.7). Queste possono essere generate proceduralmente tramite sofisticati programmi come Allegorithmic *Substance Designer*⁸, oppure tramite un processo detto *baking*. Questa operazione consiste nel salvare le caratteristiche di un modello molto dettagliato in una mappa, che verrà poi applicata su un modello meno dettagliato per simulare determinati effetti. Soprattutto l'industria dei videogiochi ricorre a questo procedimento, come spiegato in 3.3.2. Il vantaggio di usare queste particolari immagini consiste nel poter gestire i singoli effetti lavorando in 2D anziché in 3D. Inoltre, alcune mappe possono essere utilizzate come maschere per differenziare caratteristiche diverse sulla superficie.

1.5.2 UV-Mapping

L'*UV-Mapping* è il metodo con il quale un'immagine 2D viene applicata alla superficie di un modello 3D. Il primo step consiste nel rappresentare la superficie del modello in uno spazio bidimensionale chiamato *texture space*. In questo spazio, la posizione di ogni vertice viene individuata attraverso una coppia di valori (u,v) compresi nell'intervallo $[0,1]$. Tramite interpolazione lineare vengono create anche le

⁸ Allegorithmic Substance Designer, [33]

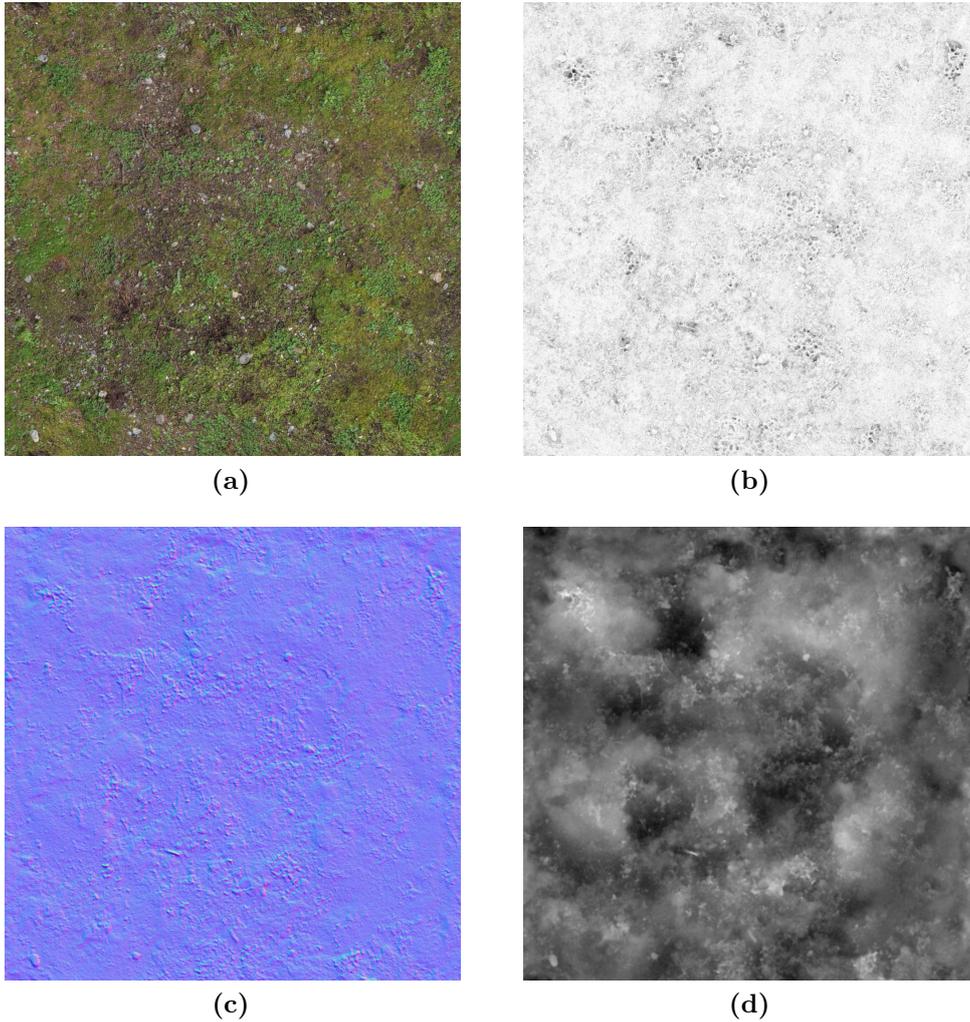


Figura 1.7: Esempi di texture maps. La diffuse map (a) contiene unicamente l'informazione cromatica. L'ambient occlusion map (b) determina quali parti della texture restano più in ombra a causa del fenomeno dell'occlusione ambientale. La normal map (c) viene utilizzata per simulare effetti di rilievo sulla superficie. La specular map (d) definisce le parti meno lucide e quelle più lucide.

coordinate (u, v) dei punti interni delle facce [15]. Questo processo viene chiamato *UV-Unwrapping*, e determina la mappatura UV della superficie. Nel texture space è presente anche la texture da applicare, che viene campionata in *texel* ad ognuno dei quali vengono associate delle coordinate (u, v) . Così facendo, vengono create precise corrispondenze tra le posizioni dei vertici nello spazio 3D e le posizioni dei medesimi vertici nello spazio 2D. Si può quindi affermare che la mappatura UV consiste in una sorta di proiezione della superficie 3D su un piano. I vari passaggi sono mostrati in

figura 1.8. Per ragioni di natura topologica, qualunque superficie chiusa necessita di tagli affinché possa essere distesa sul texture space. I tagli, chiamati *seams*, consistono in un insieme di lati contigui selezionati dalla mesh stessa. I seams vengono codificati duplicando i vertici appartenenti a tali lati ed assegnando loro differenti coordinate (u,v) . Le mesh più complesse vengono ripartite in più porzioni, dette *shells*, ciascuna delle quali viene trattata individualmente. È fondamentale che tutte le shells, una volta mappate, siano interamente contenute nel texture space.

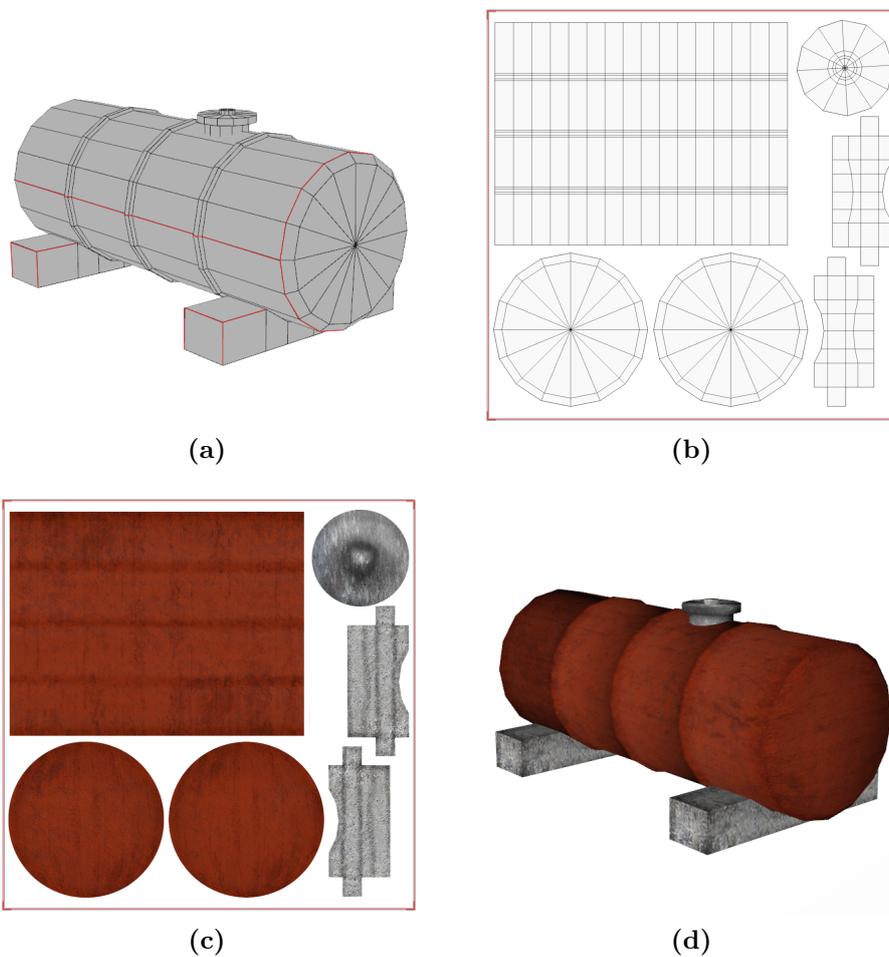


Figura 1.8: Fasi del texture mapping. I seams (a) scompongono la mesh in shells, le quali vengono individualmente proiettate all'interno del texture space (b). Viene creata la texture (c) che poi verrà proiettata sul modello (d).

1.5.3 Distorsioni e Padding

È noto che per mappare una superficie 3D nel texture space siano indispensabili uno o più tagli. Ciò è necessario sia per dispiegare superfici chiuse, sia per ridurre le distorsioni causate dalla proiezione sul piano, come mostrato in figura 1.9. Dato che molte superfici non sono direttamente sviluppabili su un piano, le shells create vengono proiettate su superfici intermedie più semplici quali cubi, cilindri e sfere, ma ciò richiede una certa corrispondenza topologica tra le due entità. Piuttosto, la maggior parte delle applicazioni tende a minimizzare le distorsioni rendendo la proiezione quanto più isometrica possibile. Una proiezione isometrica fa sì che restino invariate le aree delle facce e gli angoli formati dai lati della mesh [12].

Se da un lato i tagli minimizzano in parte le distorsioni, dall'altro condizionano in maniera considerevole la qualità della resa finale. Dopo aver applicato le textures, lungo i seams sono presenti discontinuità percepibili. Ridurre il numero di tagli è una buona prassi, ma risolve solo in parte il problema. Le distorsioni aumentano, mentre le discontinuità nei tagli presenti restano comunque visibili. La pratica più comunemente messa in atto è creare del *padding* intorno alle shells mappate. Duplicando verso l'esterno i pixel situati ai bordi delle shells, le discontinuità vengono notevolmente attenuate. Inoltre, questo è utile quando si devono creare le *MipMaps*, ovvero sequenze pre-calcolate di textures a risoluzione variabile. Le MipMaps sono largamente utilizzate in ambiti videoludici, in cui la texture di un oggetto viene dinamicamente caricata in base alla distanza a cui si trova dalla camera. Il padding evita la comparsa di tonalità di colore non volute man mano che si abbassa la risoluzione della texture.

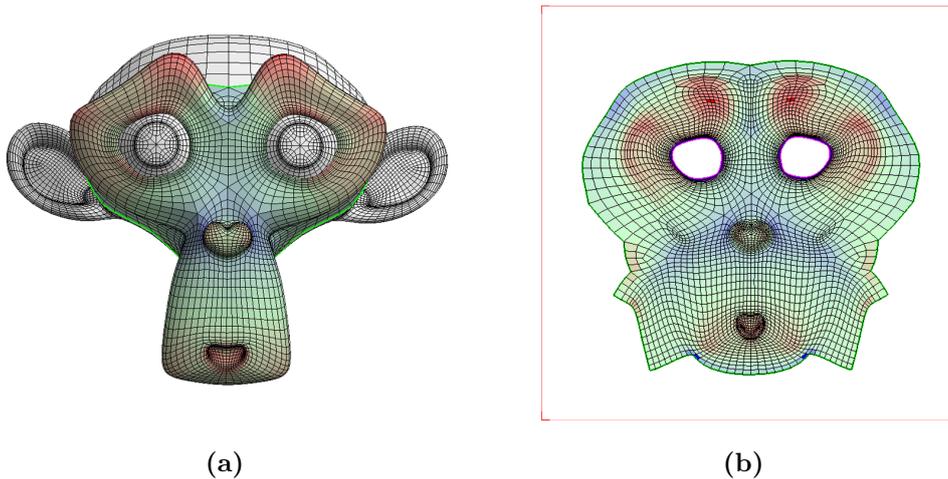


Figura 1.9: Inevitabilmente, la proiezione nello spazio (u,v) crea distorsioni. Per misurare l'entità delle distorsioni, viene adottata una scala di colori, visibile sia sul modello (a), sia sulla mappatura UV (b). Più la proiezione è isometrica, minori saranno le distorsioni.

Capitolo 2

Aspetti topologici

2.1 Concetto di topologia

Per qualunque oggetto del mondo reale è possibile notare l'andamento delle principali curve che lo compongono. Con questo processo attribuiamo il concetto di *forma* all'oggetto osservato. Durante la fase di modellazione, l'intento è quello di riprodurre digitalmente tale forma. Tuttavia, spesso non è sufficiente creare una superficie che semplicemente approssimi l'oggetto da riprodurre. Nonostante il modello ottenuto richiami le forme con un alto grado di affidabilità, può essere ritenuto scorretto ed inadeguato. Per valutare la qualità di un modello vengono prese in considerazione una serie di caratteristiche di natura geometrica che rientrano nel concetto di *topologia*. Sebbene la topologia sia un campo di studi molto vasto, nel contesto della computer grafica l'analisi è rivolta ad un insieme limitato di definizioni, regole e pratiche applicabili alle mesh poligonali, di cui si è discusso a pagina 3. Essendo queste formate da un insieme di vertici, lati e facce, le proprietà studiate sono il risultato di osservazioni eseguite su tali elementi. Tenere in considerazione questi aspetti permette di generare mesh funzionali per le successive fasi di lavorazione. In alcuni settori, un preventivo studio sulla topologia del modello è fondamentale. Un esempio si trova nella creazione di animazioni di volti umani. Le considerazioni topologiche derivano dagli studi anatomici eseguiti sui muscoli che compongono il corpo umano. Un importante risultato di tale studio è rappresentato dalle linee di Langer [5], indicanti le direzioni delle fibre muscolari presenti in tutto il corpo. Queste linee funzionano da guida per posizionare opportunamente i lati sulla mesh, come mostrato in figura 2.1. Ma anche in altri contesti è favorevole avere una topologia corretta, come il texture mapping che, come visto a pagina 9, necessita di un partizionamento della mesh. Anche la resa di un rendering fotorealistico è migliore quando la topologia è ben composta, poiché si evitano artefatti di ombreggiatura. In generale, molte comuni operazioni nella computer grafica restituiscono risultati migliori quando il modello possiede una topologia adeguata.

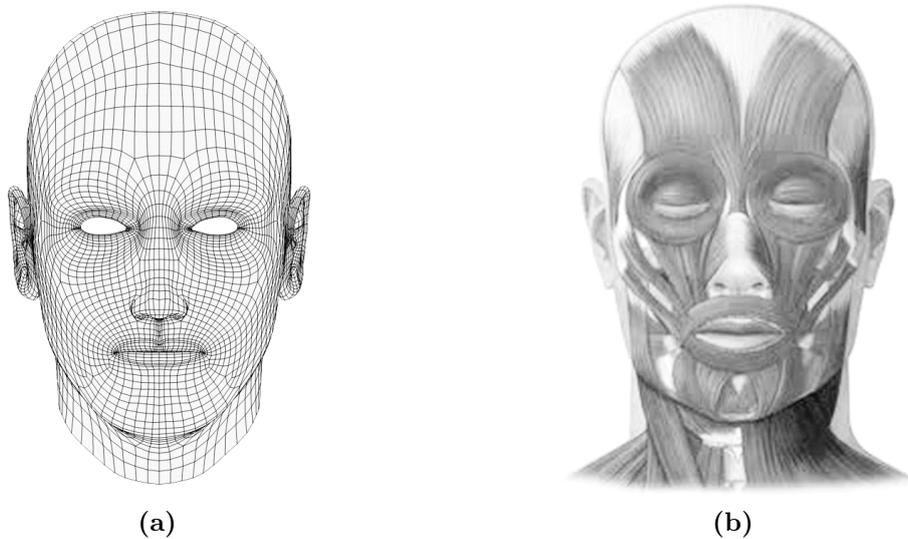


Figura 2.1: Nel caso di volti umani, una topologia corretta (a) prevede che le linee della mesh seguano l'andamento delle fibre muscolari (b). Questa caratteristica è evidente soprattutto nelle zone di labbra e palpebre. Fonti: [22] (a) e [26] (b).

2.2 Mesh quadrangolari e anisotropia

Nel paragrafo 1.3.1 sono stati discussi i numerosi vantaggi procurati dall'impiego di una tassellatura con triangoli piuttosto che con quadrangoli. Sebbene i triangoli comportino notevoli vantaggi a livello computazionale, un gran numero di ragioni hanno spinto l'industria ad adottare i quadrangoli come standard. I motivi risiedono nelle caratteristiche topologiche intrinseche delle mesh quadrangolari. Alcune di queste proprietà sono correlate alla capacità di approssimare con più precisione le forme volute, mentre altre riguardano l'adeguatezza a compiere particolari task. I quadrangoli sono più adatti a raffigurare le forme degli oggetti poiché riescono a plasmare le caratteristiche di curvatura con più naturalezza. Le zone di curvatura sono localmente caratterizzate da due direzioni principali, indicanti il grado di curvatura massima e minima presenti. Queste informazioni vengono codificate tramite tensori di curvatura, e quantificano quanto una figura geometrica si discosta dalla forma completamente piatta. Il quadrangolo, vista la sua natura geometrica, è in grado di allinearsi ad entrambe le direzioni. Le dimensioni dei suoi lati sono inversamente proporzionali ai gradi di curvatura: lungo la direzione di curvatura massima, i lati sono più corti, e viceversa lungo la direzione di curvatura minima (si osservi la figura 2.2). Questa caratteristica topologica prende il nome di quadrangolazione *anisotropica*, la quale prevede che le caratteristiche dei quadrangoli dipendano in base alle direzioni di curvatura considerate [9]. In caso contrario, la quadrangolazione è *isotropica*. In virtù di tali proprietà, le mesh quadrangolari risultano di qualità

superiore e molto più versatili ai fini artistici. Si possono apprezzare notevoli vantaggi in tutte le fasi della pipeline di produzione: operazioni che richiedono precisione come il rigging, il texturing e la suddivisione vengono agevolate in presenza di quadrangoli.

Per un artista è decisamente più naturale e conveniente generare mesh costituite uniformemente da quadrangoli. Ciò nondimeno, in molti casi non è indispensabile verificare questa condizione piuttosto rigida. Un certo numero di triangoli viene comunque tollerato, a patto che non ostacolino eccessivamente le varie fasi di lavorazione. Mesh di questo tipo sono dette *quad-dominant*, e sono le più utilizzate all'atto pratico. Non vengono tollerate, invece, facce con più di quattro lati, dette *N-gons*, poiché causano deformazioni alquanto innaturali e problemi di shading durante le operazioni di suddivisione.

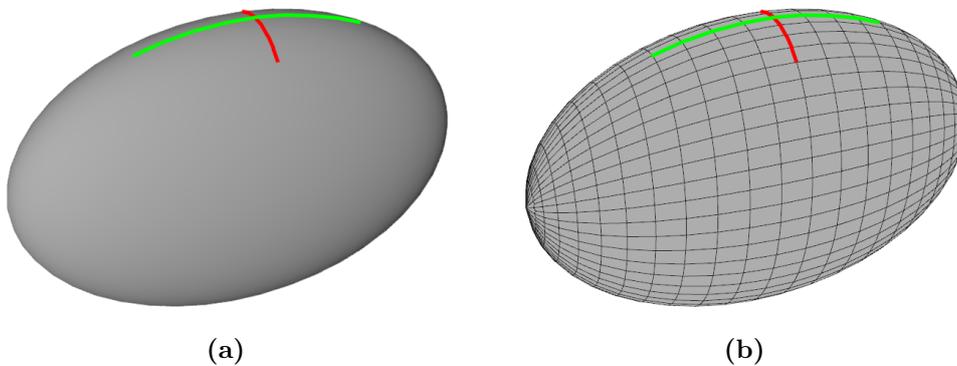


Figura 2.2: Quadrangolazione anisotropica. Localmente, è possibile individuare le due principali direzioni di curvatura. In rosso è evidenziata la direzione con il grado di curvatura maggiore, mentre in verde quella col grado di curvatura minore (a). I gradi di curvatura determinano le dimensioni dei lati dei quadrangoli: laddove la curvatura è maggiore, i lati saranno più corti, e viceversa (b).

Un ulteriore elemento che caratterizza le mesh quadrangolari sono i *cross-fields*, consistenti in una coppia di vettori ortogonali assegnati ad ogni punto della mesh [1]. Ciascun cross-field è costituito da due componenti: quella angolare e quella dimensionale. Queste vengono immagazzinate in due strutture separate, chiamate rispettivamente *orientation field* e *sizing field*. Dalla prima è possibile ottenere informazioni circa l'orientamento dei quadrangoli, mentre dalla seconda le dimensioni dei quadrangoli.

2.3 Valenza e Singolarità

La *valenza* è la caratteristica topologica che identifica qualunque vertice. È un valore numerico intero rappresentante il numero di lati incidenti al vertice stesso. In base

alla posizione, un vertice può essere interno o di confine, mentre in base alla valenza può essere regolare oppure irregolare. Per essere regolari, tutti i vertici interni devono avere valenza quattro, mentre quelli di confine valenza tre. I vertici di confine posti in angoli concavi sono regolari se hanno valenza quattro, mentre devono avere valenza due se sono situati in corrispondenza di angoli convessi [14]. Se le valenze citate non vengono rispettate, i vertici sono irregolari e vengono chiamati *singolarità*.

Le singolarità influenzano in maniera rilevante il risultato fornito da diverse operazioni, ad esempio la suddivisione. L'algoritmo di Catmull-Clark, esaminato in 1.4.1, restituisce superfici con continuità di tipo C^2 solamente in prossimità di vertici regolari, mentre in corrispondenza delle singolarità la massima continuità ottenibile è di tipo C^1 . Ciò crea degli artefatti sulla superficie, ma tipicamente questi sono vistosi solamente se le singolarità hanno valenze di molto superiori a cinque, come nel caso mostrato in figura 2.3. È da tenere in considerazione che i vertici irregolari vengono anche generati dai processi stessi di suddivisione (si noti la figura 1.4).

Ciononostante, vertici irregolari con valenza tre o cinque sono addirittura indispensabili, poiché contribuiscono alla determinazione di altre caratteristiche topologiche, quali quelle discusse in 2.4 e 2.5.

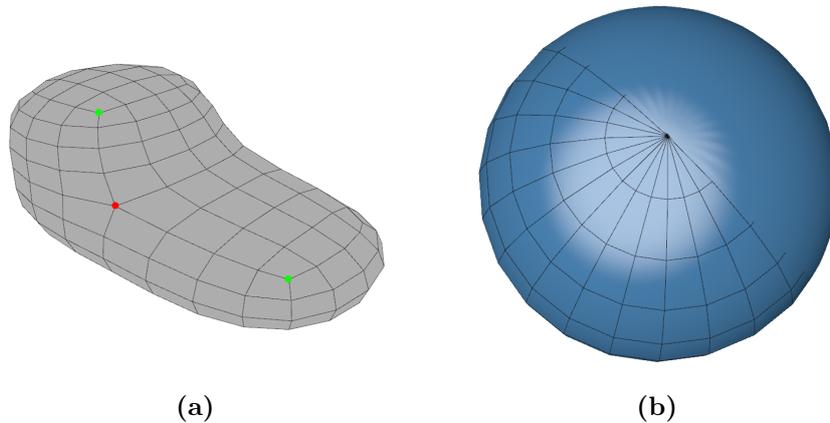


Figura 2.3: In (a) sono evidenziate le singolarità con valenza tre (in verde) e cinque (in rosso). Generalmente, valenze di questo tipo non creano artefatti pesantemente visibili. Anzi, sono indispensabili per garantire un buon flusso di lati. Invece, valenze esageratamente elevate causano artefatti facilmente visibili quando si suddivide la mesh (b).

In base al grado di regolarità dei vertici, le mesh quadrangolari possono essere suddivise nelle seguenti categorie:

- *Mesh regolari:* mesh composta unicamente da vertici regolari. Mesh di questo tipo trovano pochi contesti di utilizzo, oltre ad essere molto difficili da generare.

- *Semi-regolare*: mesh in cui la maggior parte dei vertici sono regolari. È la categoria di mesh più diffusa, poiché all'atto pratico una certa quantità di singolarità è anche proficua.
- *Destruzzurata*: quando la maggior parte dei vertici sono irregolari. Mesh di questo tipo non trovano un concreto impiego, poiché non consentono di sviluppare condizioni topologiche favorevoli.

2.4 Edge-flow ed Edge-loops

Le principali caratteristiche topologiche dei lati vengono derivate dal loro posizionamento. La rete di lati disposti globalmente su tutta la superficie è detta *edge-flow*, e determina come i lati percorrono interamente la superficie. I lati devono essere allineati lungo le principali curvature presenti nella forma dell'oggetto. Un buon flusso garantisce deformazioni più morbide e con meno distorsioni, oltre ad aggiungere un certo equilibrio visivo alla mesh. Nello specifico, le singolarità a valenza cinque consentono di controllare il flusso dei lati lungo più direzioni, indispensabile soprattutto nel contesto della modellazione organica. Un corretto esempio di edge-flow è rappresentato in figura 2.4 (b), in cui le linee di Langer rendono l'idea della disposizione ideale dei lati di una mesh che intende rappresentare un volto umano.

Di maggiore interesse è la posizione di un singolo lato rispetto ai lati adiacenti. In particolare, una sequenza di lati consecutivi individua un *edge-loop*. La caratteristica fondamentale degli edge-loops è che sono costituiti esclusivamente da vertici regolari ed al più due singolarità. Gli algoritmi di loop-selection, che consentono di selezionare rapidamente molti lati consecutivi, basano il loro funzionamento sulla verifica della regolarità dei vertici: il lato successivo viene selezionato solamente se il vertice all'estremità opposta è anch'esso regolare (si veda la figura 2.4 (a)). In caso contrario, la sequenza viene interrotta. Dunque, è importante capire dove posizionare al meglio i vertici irregolari, al fine di poter creare edge-loops funzionali, ad esempio per la fase di texture mapping. Come visto al paragrafo 1.5.2, i tagli sono indispensabili per poter distendere la mesh nel texture space. Spesso, i lati appartenenti ad un edge-loops vengono scelti come i più consoni in cui tagliare la mesh.

Benché si riferiscano a considerazioni eseguite su scale differenti, i concetti di edge-flow ed edge-loop sono molto correlati tra loro. Un flusso di lati ben strutturato consente di aggiungere facilmente nuovi edge-loops. Questo aspetto diventa molto utile quando si devono aggiungere catene di lati la cui funzione è quella di supporto. Questi aumentano la qualità della mesh lungo gli spigoli, poiché attenuano gradualmente il cambio di direzione della superficie, rendendola visivamente più morbida. Se poi la mesh dovrà essere suddivisa, questi lati sono utili per smussare gli spigoli. Se da un lato ciò consente una migliore visualizzazione del modello, a beneficiarne sono anche altre operazioni. Durante il texture mapping, le

distorsioni causate dai tagli (si veda 1.5.3) possono essere ulteriormente attenuate se questi vengono accompagnati da edge-loops di supporto. Anche le animazioni risultano più naturali poiché le deformazioni vengono interpolate attraverso più vertici, risultando anche più morbide.

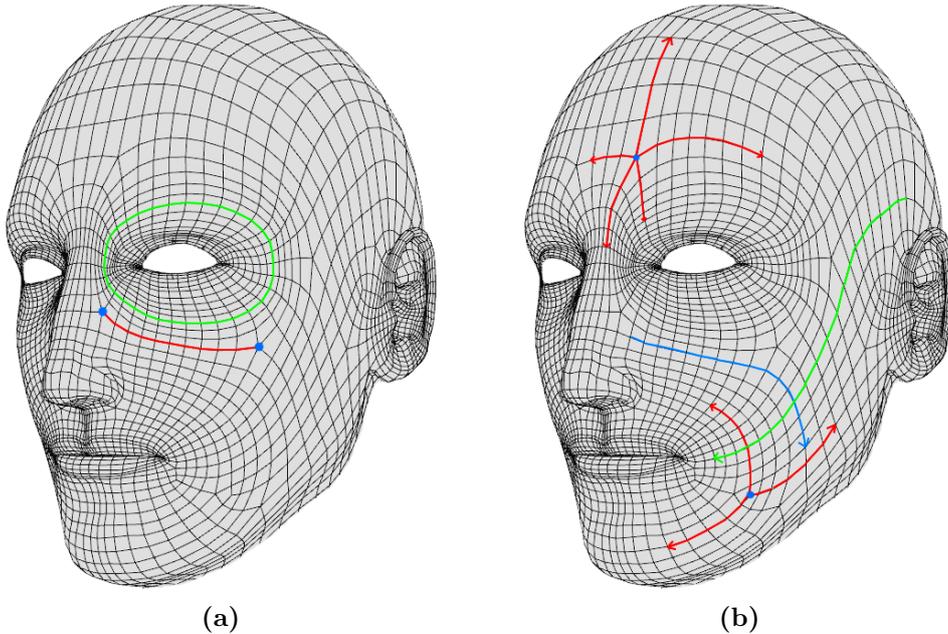


Figura 2.4: In (a) vi sono due esempi di edge-loops. In verde, una sequenza di lati chiusa forma un edge-loop, mentre la sequenza in rosso è un edge-loop interrotto da due singolarità. In (b) è visualizzato il concetto di edge-flow, caratteristica in parte dipendente dalle singolarità. Soprattutto nei modelli organici, questo aspetto è di fondamentale importanza. Fonte modello: [22].

2.5 Risoluzione variabile

Anche le qualità topologiche dei quadrangoli possono essere valutate facendo sia considerazioni locali che globali. Localmente parlando, la qualità di un quadrangolo viene valutata comparandolo idealmente ad un triangolo. Quest'ultimo è sempre planare, e i parametri presi in considerazione sono la misura dell'angolo più ampio e quanto differisce dalla caratteristica equilatera [1]. Allo stesso modo, per un quadrangolo si considerano il grado di planarità, l'ampiezza degli angoli e le lunghezze dei lati. Gli angoli devono essere quanto più retti possibile, mentre i lati opposti devono avere lunghezze simili.

Le considerazioni globali, invece, riguardano principalmente la distribuzione dei quadrangoli su tutta la superficie. In particolare, un parametro di rilievo è la densità

di quadrangoli. Soprattutto nei modelli più complessi, la concentrazione di poligoni può variare parecchio in alcune zone, affinché sia possibile rappresentare fedelmente i dettagli più fini. Ma ciò comporta vantaggi anche durante la fase di animazione, poiché è possibile riprodurre anche lievi deformazioni senza troppe distorsioni.

Per variare la densità di quadrangoli, è necessario inserire alcuni vertici irregolari, sia a valenza tre che cinque. Tuttavia, eseguendo queste accortezze il modello finale risulta più ottimizzato, poiché vengono aggiunti quadrangoli solamente laddove sono indispensabili per garantire risultati migliori. È possibile notare questa proprietà osservando la figura 2.4: le zone attorno agli occhi e alla bocca contengono quadrangoli più piccoli rispetto alle zone corrispondenti a guance e fronte, poiché i dettagli da rappresentare sono più fini. L'aumento di densità è giustificato anche dal fatto di essere le parti maggiormente coinvolte durante l'animazione.

Capitolo 3

Retopology: motivazioni, contesti e metodologie

3.1 Definizione e motivazioni

In base alla discussione trattata nel paragrafo 1.2, esiste una gran varietà di tecniche per creare modelli tridimensionali. Recentemente, la fase modellazione è stata arricchita con un'offerta di strumenti sempre più comodi da usare e tecniche in continuo miglioramento. Alcune di esse permettono di avere pieno controllo su ogni singolo elemento del modello, altre invece no, mentre altre ancora creano una modalità di controllo ibrida. Sebbene alcune di queste riducano i tempi di produzione oppure gli sforzi richiesti al modellatore, in certe circostanze il risultato ottenuto non è adatto alle finalità che si vogliono raggiungere. I problemi che si possono riscontrare sono principalmente di natura topologica, ma spesso riguardano anche il peso del modello in termini computazionali.

In questi casi, è necessario aggiungere uno step alla fase di modellazione chiamato *retopology*. Con questo termine si indica il processo di sostituzione di una mesh con una nuova che rispecchia determinate caratteristiche. La mesh ricostruita deve soddisfare due fondamentali requisiti: deve mantenere la forma della mesh originale e presentare una topologia conforme agli scopi del progetto. A tale proposito, si osservi la figura 3.1. Talvolta, ci si riferisce alla retopology con il termine *remeshing*, ad evidenziare il fatto che i vertici e le facce della mesh iniziale non vengono conservati.

Indipendentemente dal contesto in cui si opera, la fase di retopology comporta inevitabilmente un aumento dei tempi di produzione del modello, oltre a richiedere notevoli abilità da parte del modellatore. Ciò nonostante, determinate tecniche di modellazione (discusse in 3.2.1 e 3.3.1) dominano nei settori dell'entertainment e della visualizzazione, benché non garantiscano immediatamente i risultati ottimali. Le pipeline di produzione sono state irreversibilmente modificate, e ciò ha reso la retopology uno step inevitabile.

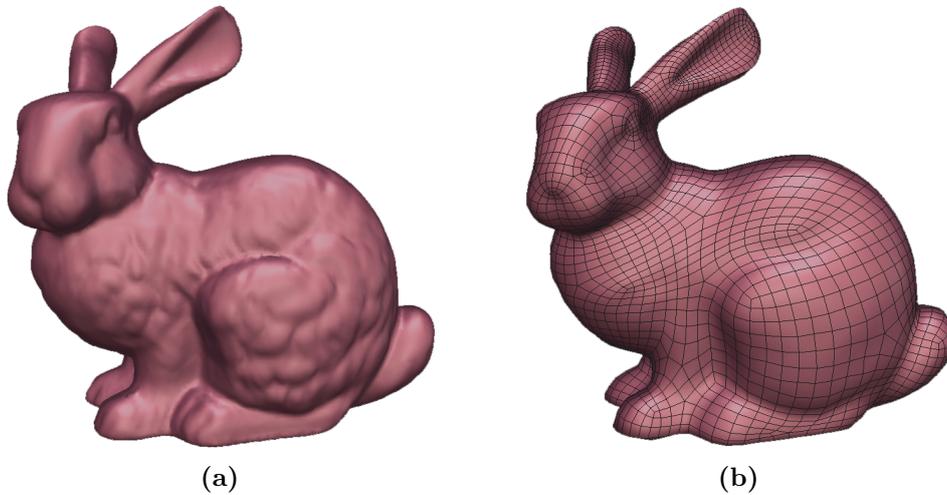


Figura 3.1: Con il processo di retopology, la mesh mostrata in (a) è stata ricostruita fino a diventare la mesh mostrata in (b). Si può notare come i due modelli abbiano la stessa forma, ma il secondo contiene molti meno poligoni e presenta caratteristiche topologiche migliori.

3.2 Retopology nel Cultural Heritage

Il termine *cultural heritage* si riferisce al complesso dei beni appartenenti al patrimonio dell'uomo che comprendono sia oggetti fisici come opere artistiche e letterarie, monumenti architettonici e resti archeologici, sia beni immateriali come tradizioni, pratiche sociali e conoscenze storiche tramandate in forma orale o scritta. La conservazione dei beni culturali è la pratica di proteggere tutte quelle proprietà che consentono di interpretare il significato associato ad un determinato bene. È un campo molto attivo e di grande portata, che richiede studi appartenenti a diverse discipline.

Soprattutto negli ultimi decenni, la conservazione dei beni culturali ha fatto largo impiego della computer grafica. La digitalizzazione dei beni ha portato alla continua nascita di iniziative che intendono sfruttare le potenzialità dei sistemi digitali per favorire la conservazione, divulgazione e consultazione dei beni. Digitalizzare significa, inevitabilmente, approssimare un oggetto tramite una sua rappresentazione discreta. Questo comporta un problema rilevante: il modello ottenuto non è in grado di rappresentare verosimilmente l'oggetto reale. L'aspetto esteriore, ed in particolare la forma geometrica, è la prima caratteristica a dover essere conservata senza alterazioni. Da questa criticità, la comunità scientifica ha optato per l'impiego di tecniche di scansione 3D. Tuttavia, tale scelta comporta problematiche non indifferenti relative tanto alla creazione di materiale digitale quanto all'usabilità di questi contenuti.

3.2.1 Scansione 3D e Reverse modeling

Tipicamente, lo sviluppo di un modello reale comincia con la creazione di un concept. Questo viene continuamente ridefinito fino ad assumere la forma definitiva. Soprattutto nei contesti di progettazione meccanica, l'intero processo si conclude con la conversione dell'oggetto digitale ad oggetto fisico. In pratica, il modello digitale rappresenta il dato di partenza, mentre quello finale il dato di arrivo. Invece, con la scansione 3D il processo è esattamente inverso: l'oggetto fisico reale rappresenta il dato di partenza da cui viene generato il modello digitale. Ci si riferisce a questo approccio con il termine *reverse modeling*. Il diagramma mostrato in figura 3.1 riassume le macro-fasi dell'intero processo. Benché la scansione 3D preveda l'esecuzione di una sequenza di operazioni fisse, non esiste una tecnica unica e standardizzata per acquisire i dati dall'oggetto reale. È indispensabile una fase preliminare in cui si valutano le alternative a disposizione, al fine di scegliere la tecnica più adatta. Oltre al contesto in cui adoperiamo, i principali fattori che condizionano questa scelta riguardano sia le caratteristiche morfologiche dell'oggetto quali dimensioni, materiale e complessità della forma, sia i principi di funzionamento dei dispositivi. Indipendentemente dalla tecnica utilizzata, la creazione necessita di fasi di lavorazione indispensabili.

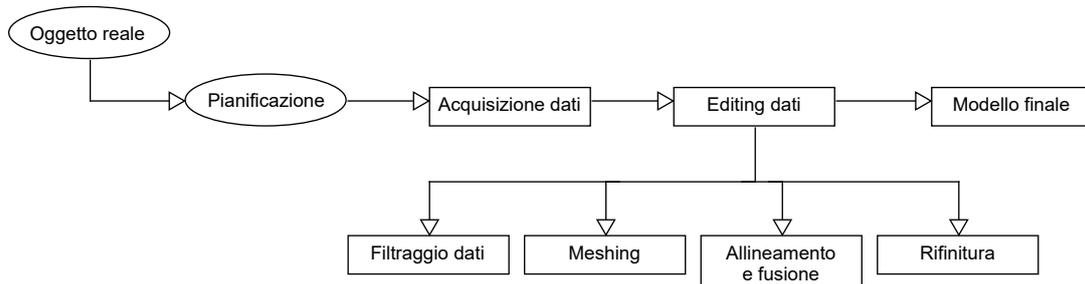


Figura 3.2: Reverse modeling: pipeline di creazione di modelli digitali nell'ambito dell'heritage culturale.

Acquisizione L'acquisizione dei dati geometrici è la fase centrale di tutto il processo di digitalizzazione. Al giorno d'oggi è disponibile un'ampia gamma di strumenti tecnologici che basano il loro funzionamento su approcci e tecniche diverse (per un'esauriva classificazione e trattazione si rimanda a [4], p. 13). Indipendentemente dalla tecnica adottata, il risultato di questa fase è la *point cloud*, ovvero un insieme di punti ciascuno dei quali viene individuato da una tripletta di valori rappresentanti le coordinate (x,y,z) nello spazio. Con determinate tecniche, oltre all'informazione spaziale viene aggiunta anche quella cromatica, costituita da una tripletta di valori rappresentanti le componenti (R,G,B) . Per la natura intrinseca di ogni oggetto reale,

può essere acquisita solamente una parte dell'oggetto alla volta. Dunque, è necessario eseguire più acquisizioni consecutive per poter ottenere l'oggetto intero. Il risultato di ciascuna acquisizione è detto *range map*.

Filtraggio La nuvola di punti è una rappresentazione molto grezza dei dati acquisiti, e generalmente presenta errori che non consentono di estrarre una mesh conforme. I punti rilevati vengono processati con operazioni di pulitura mirate sia all'eliminazione di dati non conformi alla geometria, sia alla riduzione del rumore. Questo viene descritto come variazioni locali casuali della posizione dei punti. La pulitura viene eseguita tramite algoritmi semi-automatici o completamente automatici. È una fase molto delicata, poiché il risultato restituito deve consentire la generazione di una mesh conforme.

Meshing La fase successiva è la creazione di una mesh poligonale i cui vertici corrispondono ai punti filtrati. Viste le proprietà (discusse a pagina 4) che rendono il triangolo una primitiva molto più semplice da gestire, la mesh generata è triangolare. In base alla regolarità con cui sono disposti i punti nello spazio, la nube di punti può essere strutturata o non strutturata. A determinare questa caratteristica è la tecnica utilizzata durante la fase di acquisizione. Nel caso in cui la nube sia strutturata, i punti vengono collegati a triplette seguendo semplici regole basate sull'ordine di lettura, mentre nel caso in cui la nube sia non strutturata è necessaria un'ulteriore fase di elaborazione dei dati, a seguito della quale è possibile applicare algoritmi di triangolazione.

Allineamento e fusione È la fase in cui tutte le range map precedentemente acquisite e trasformate in mesh vengono allineate e fuse per generare un unico reticolo poligonale. Dopo aver scelto arbitrariamente una range map iniziale, avviene l'allineamento con un'altra range map. Successivamente, un'ulteriore range map viene allineata alle due precedenti, che vengono considerate come un pezzo unico. Questa operazione continua fino a quando tutte le range map sono state allineate. L'intera procedura è semiautomatica, e necessita di onerosi calcoli computazionali. Infine, tutte le mesh vengono unite per formare una struttura poligonale unica.

Rifinitura La superficie ottenuta può presentare diversi errori di natura topologica. Talvolta, le regole di connettività per generare una superficie conforme possono non essere rispettate in alcune zone, e il risultato è una mesh non-manifold. I tipici errori consistono in facce sovrapposte, lati condivisi da più di due triangoli e normali invertite. Questi possono essere corretti manualmente, ma qualunque software di computer grafica dispone di strumenti automatici che agevolano l'operazione. Un problema molto più rilevante è costituito dalle lacune, spesso causate da occlusioni dovute alle forme del modello, che rendono inaccessibili alcuni punti. In questo caso,

gli algoritmi automatici riescono a valutare l'andamento delle curvature presenti ai lati della lacuna, riempiendola di conseguenza.

3.2.2 Problematiche dei modelli acquisiti

I problemi associati alle mesh generate tramite scansione riguardano sia aspetti computazionali che topologici. Se da un lato una mesh con un elevato numero di facce garantisce un alto livello di dettaglio, dall'altro implica una notevole richiesta di risorse in termini di memoria e calcoli. Soprattutto in contesti interattivi, questa richiesta non sempre può essere quasi mai soddisfatta. Inoltre, sebbene avere un modello preciso e dettagliato sia importante in un contesto come la conservazione dei beni, spesso nei dati acquisiti è presente una certa ridondanza. Il processo di *decimazione* è in grado di semplificare il modello sfruttando questa ridondanza. Esistono molti algoritmi in grado di semplificare le mesh triangolari, ed una completa trattazione è disponibile in [11], ma l'approccio più comune prevede l'unione di gruppi di triangoli con caratteristica di coplanarità. Ciò nonostante, percentuali di riduzione troppo elevate causano perdite notevoli di dettaglio del modello. A pagina 48 si discute di come questa operazione possa risultare poco adatta in certe circostanze.

In virtù dei vantaggi offerti dalle mesh quadrangolari (si veda il capitolo 2), il modello digitale generato si presta poco a lavorazioni come il texture mapping e l'animazione, qualora siano necessarie. È dunque necessario convertire il modello in una mesh quadrangolare. In questi casi, la retopology consente sia di ridurre drasticamente il numero di facce, sia di creare una topologia funzionale.

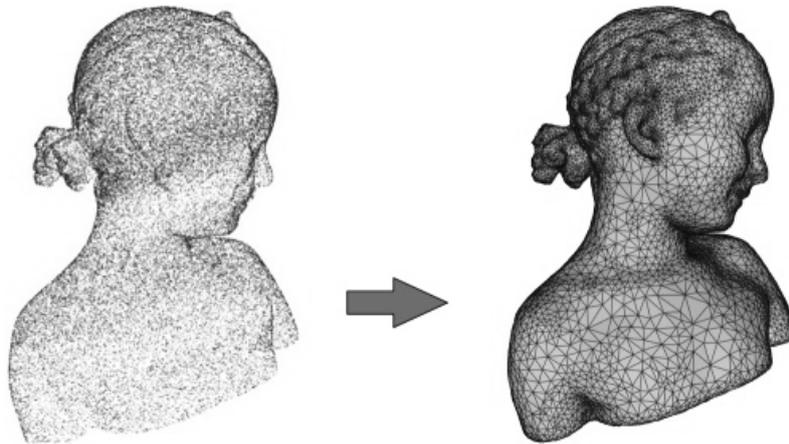


Figura 3.3: Applicando le fasi descritte in 3.2.1, dalla point cloud si costruisce la versione digitale dell'oggetto scannerizzato. Si può notare come il modello ottenuto sia particolarmente denso di triangoli, oltre a presentare un flusso di lati del tutto caotico. La mesh risultante è dunque destrutturata, prestandosi poco a successive lavorazioni.

3.3 Retopology per l'Entertainment

L'industria dell'intrattenimento è costituita principalmente da due settori: quello cinematografico e quello videoludico. Questi settori non possono fare a meno della CGI per creare i propri contenuti, e hanno adottato delle pipeline di produzione piuttosto rigide (si noti la figura 3.4). Sebbene, per certi versi, le due pipeline coincidano, creare modelli da usare in film d'animazione è diverso dal creare modelli da inserire in videogiochi. La principale differenza consiste nel tempo a disposizione per eseguire il rendering a video di un frame. I modelli per i film generalmente sono più dettagliati, ovvero costituiti da più poligoni. Poiché il rendering di ogni frame viene eseguito solamente una volta, avere più dati da elaborare non comporta enormi complicazioni. Invece, in ambito interattivo il rendering viene eseguito molte volte al secondo, e al fine di garantire una certa fluidità di gioco è necessario che i modelli siano quanto più leggeri possibile.

Le similitudini tra le due pipeline riguardano le tecniche di modellazione adottate. Spesso, in entrambe le produzioni la tecnica impiegata nelle prime fasi di modellazione è lo sculpt 3D. L'adozione di questa tecnica è giustificata per una serie di ragioni. Innanzitutto, è ampiamente apprezzata dai modellatori poiché permette di generare mesh senza tenere in considerazione la disposizione di vertici e facce, modellando in una maniera molto più naturale. Inoltre, consente di separare la fase di creazione della forma desiderata da quella di creazione di una topologia funzionale per tale forma¹. Un aspetto condiviso è che entrambe le produzioni necessitano di avere modelli adatti all'animazione. Dunque, la fase di retopology copre un ruolo fondamentale in ambedue i settori. Uno stesso modello può anche essere retopologizzato in più modi differenti, creando diverse versioni ciascuna ottimizzata per uno scopo ben preciso. In conclusione, sculpt 3D e modellazione poligonale sono le due tecniche impiegate di pari passo in entrambi i workflow di produzione.

3.3.1 Sculpt 3D

L'idea alla base dello sculpt 3D è quella di riprodurre al calcolatore l'esperienza della scultura reale, in cui un oggetto viene creato aggiungendo e rimuovendo continuamente materiale. Dopo aver istanziato una primitiva di base sufficientemente suddivisa, il modellatore usa un pennello digitale (*brush*) per deformarla. Le pennellate consentono di creare cuspidi e avvallamenti sulla superficie dell'oggetto, e possono essere controllate da proprietà del pennello quali intensità e ampiezza. A livello tecnico, qualunque deformazione consiste in una traslazione locale dei vertici che rientrano nel raggio d'azione del pennello. Tipicamente, il modellatore crea la forma voluta attraverso un processo per raffinamenti successivi. La figura 3.5 raffigura questa

¹ [13], p. 95

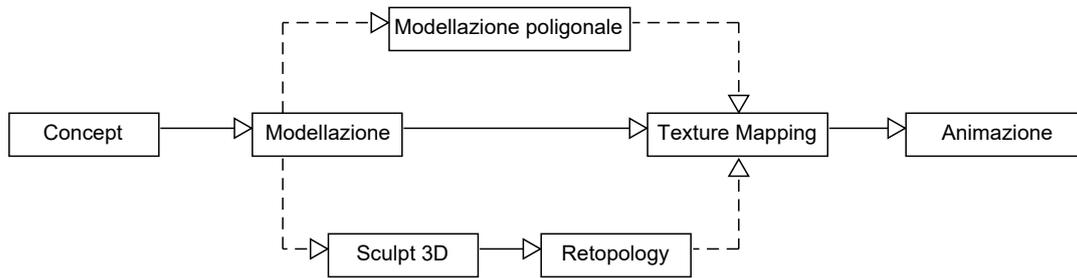


Figura 3.4: Generica pipeline di creazione di modelli digitali nell’industria dell’intrattenimento. In base al contesto, può prevedere ulteriori step.

particolare tecnica di modellazione. Per aggiungere dettagli sempre più accurati è possibile suddividere la mesh durante la lavorazione stessa. Per gestire al meglio il peso computazione causato dalle ripetute suddivisioni del modello, una feature molto utile è il *dynamic subdivision*, che consente di suddividere la mesh solamente nelle zone in cui si devono andare a scolpire i dettagli più fini.

Il livello di dettaglio raggiungibile è pressoché illimitato, e ciò consente di aggiungere dettagli che, tramite modellazione poligonale classica, sarebbero difficili da creare. Benché software generalisti come Autodesk *Maya*[®] e *Blender*[™] includano un modesto set di strumenti per eseguire sculpt, sul mercato esistono soluzioni più professionali come Autodesk *Mudbox*[®], *Pilgway 3D-Coat*² e *Pixologic ZBrush*[®]. Questi ultimi sono ottimizzati per lavorare con questo tipo di mesh senza rallentare le prestazioni del calcolatore, oltre a fornire strumenti più avanzati.

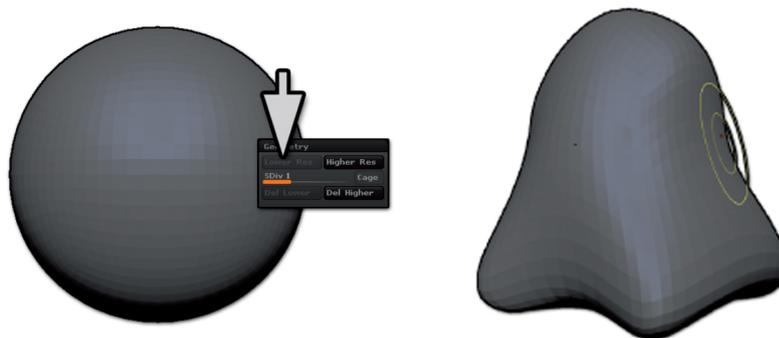


Figura 3.5: Partendo da una semplice mesh, la forma desiderata viene modellata tramite il pennello digitale.

² Pilgway 3D-Coat, [42]

3.3.2 Problematiche dei modelli scolpiti

Il primo grande problema relativo ai modelli digitalmente scolpiti riguarda l'elevato numero di facce. La libertà concessa da questa tecnica consente di aggiungere poligonalmente ogni sorta di dettaglio, come ad esempio rughe e pori della pelle. Il modello generato arriva facilmente a contare fino a milioni di poligoni, e mesh così dense non possono essere facilmente gestite da altri software, né tantomeno in contesti interattivi.

Sebbene molti software mettano a disposizione algoritmi di decimazione, il loro utilizzo risulta essere un controsenso, poiché non è più possibile rappresentare il dettaglio più fine presente sul modello. Inoltre, non risolve il secondo problema associato al modello, ovvero l'assenza di una topologia adeguata alle operazioni di animazione. Tramite la retopology, è possibile ridurre drasticamente il peso del modello, oltre a renderlo adatto alle deformazioni. Nell'ambito videoludico, il modello retopologizzato viene definito *low poly*, in virtù del ridotto numero di facce. Ovviamente i dettagli più fini vengono persi durante la fase di retopology, ma vengono recuperati ricorrendo a particolari textures quali normal map o displacement map, di cui si discuterà al paragrafo 3.5).

3.4 Tecniche e approcci

All'atto pratico, il processo di retopology può essere paragonato ad una vera e propria modellazione ex-novo dell'oggetto. In entrambi i casi, il modellatore deve creare una mesh poligonale adatta alle finalità del progetto. Anche gli strumenti impiegati hanno funzionalità pressoché simili, poiché permettono di eseguire operazioni direttamente su vertici, lati e facce. La principale differenza consiste nel dato di partenza. Nella modellazione poligonale tradizionale spesso vengono utilizzati disegni e immagini come riferimenti per ricreare correttamente le forme dell'oggetto. Invece, nella retopology il riferimento è costituito da un modello digitale precedentemente creato. Dunque, l'attenzione non è rivolta tanto verso la creazione della forma, quanto piuttosto verso la disposizione dei nuovi vertici sul modello già presente.

L'intero processo di retopology può essere suddiviso in due fasi. La prima consiste in uno studio topologico del modello su cui si deve lavorare. Sebbene ciò non sia obbligatorio, i modellatori trovano molto utile disegnare sul modello stesso delle linee guida che visualizzano l'edge-flow più adatto, soprattutto quando si lavora con modelli organici. Da questo studio viene dedotto come collocare opportunamente i vertici, in modo tale da attribuire alla mesh appropriate caratteristiche topologiche. La seconda fase consiste nella creazione vera e propria della nuova mesh. Nonostante il preliminare studio topologico, generalmente questa fase necessita di molto tempo, nonché lavoro manuale da parte del modellatore. Tuttavia, al fine di agevolare l'operazione, negli ultimi anni sono state sviluppate nuove tecniche, incluse tanto

nei software generalisti quanto in software più settoriali, che consentono di ridurre il peso dell'operazione. Non esiste una catalogazione unicamente valida e condivisa di tutte queste tecniche, poiché i parametri da considerare sarebbero innumerevoli: strumenti offerti, tempi di elaborazione, qualità del risultato, grado di intervento del modellatore. Tuttavia, è assodato che per venire incontro alle esigenze dei modellatori è necessario adottare un nuovo approccio alla costruzione della mesh. Da un lato, l'evoluzione delle tecniche di retopology si è orientata verso l'adozione di strutture geometriche poste ad un livello di astrazione superiore a quello di vertici e quadrangoli, mentre dall'altro all'automatizzazione parziale o completa del processo.

L'antitesi presente in tutti questi approcci risiede principalmente nel controllo sulla qualità della mesh in rapporto ai tempi di esecuzione richiesti. Come emergerà durante la discussione effettuata nella sezione progettuale (capitolo 5), spesso i metodi tradizionali garantiscono un risultato di qualità superiore ai metodi automatici e semi-automatici. Questi ultimi, inoltre, talvolta richiedono una fase di rifinitura volta a migliorare il risultato ottenuto dal calcolatore. In questo ambito non sempre automatizzare significa produrre ottimizzare il risultato.

3.4.1 Approcci tradizionali

I tradizionali approcci alla retopology prevedono l'impiego di tecniche che consentono di lavorare direttamente su vertici, lati e quadrangoli. Il processo eseguito con l'impiego di queste tecniche prende il nome di *retopology manuale*. Sono tecniche con cui è facile familiarizzare, poiché basano il loro funzionamento su azioni piuttosto semplici, ma alquanto ripetitive. Con queste tecniche, l'intero processo si rivela piuttosto tedioso e time-consuming. Ciononostante, resta l'approccio più comunemente impiegato dalla maggior parte degli artisti, e diversi software si impegnano ad offrire strumenti che cercano di renderlo più scorrevole possibile.

La tecnica più tradizionale prevede il posizionamento manuale di ogni singolo vertice sulla superficie. I quadrangoli vengono generati collegando i vertici a gruppi di quattro alla volta. In certe circostanze, e con certi strumenti, sono consentiti anche triangoli. Per adagiare automaticamente i vertici sulla superficie del modello, si ricorre alla funzione *snap*. Qualunque software di computer grafica dispone di questa funzione. Essendo il vertice la più piccola unità di informazione di cui è costituita una mesh, questa tecnica consente un pieno controllo sulla qualità del risultato. Ciò è particolarmente utile nei contesti di animazione, dove i requisiti topologici sono più stringenti. Un'estensione della funzione di *snap* che consente il posizionamento contemporaneo di più facce è la funzione *shrinkwrap*. Tramite questa è possibile adagiare direttamente una primitiva sul modello da retopologizzare. Trova impiego in contesti molto limitati (ad esempio terreni e paesaggi), poiché è necessaria una certa corrispondenza topologica tra la primitiva e il modello iniziale. In questo

caso, il parametro che controlla la qualità del risultato è il numero di suddivisioni della primitiva.

Una ulteriore modalità di creazione dei quadrangoli prevede l'ausilio di *curve* disegnate a mano sulla superficie del modello. Le intersezioni tra le curve individuano i nuovi vertici, e vengono automaticamente riempite con i quadrangoli. Sebbene questa tecnica consenta di generare molti quadrangoli contemporaneamente e con un andamento piuttosto definito, quasi sempre richiede correzioni aggiuntive da eseguire sui singoli vertici.

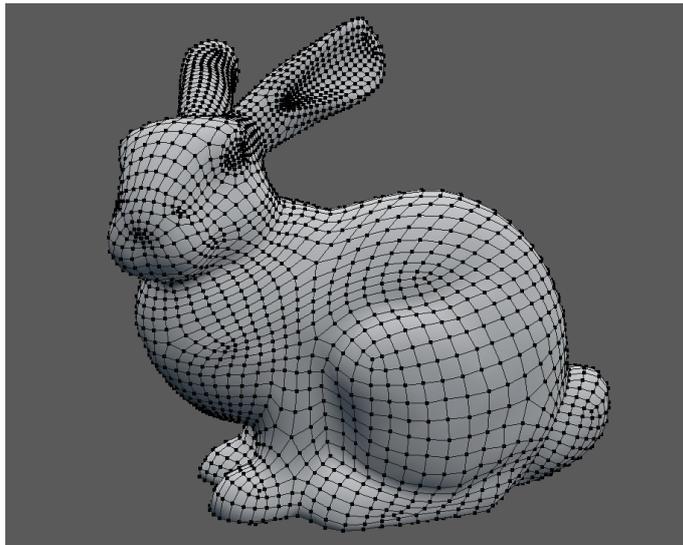


Figura 3.6: La retopology eseguita con tecniche tradizionali prevede la creazione e gestione dei vertici che costituiranno la mesh finale.

3.4.2 Approcci ibridi

Gli approcci ibridi comprendono una serie di tecniche che adottano un paradigma completamente diverso per generare una mesh. Queste tecniche consentono di lavorare con strutture gerarchicamente superiori ai quadrangoli. Tipicamente, strutture geometriche più complesse permettono di generare e gestire più quadrangoli contemporaneamente, ma non consentono di intervenire sui singoli vertici, se non in una fase successiva. Questo può rivelarsi un limite, anche se i tools che sfruttano tali strutture offrono strumenti ad hoc per modificare i poligoni.

I tools che sfruttano approcci diversi da quelli tradizionali sono pochi, e spesso non integrati nei software di grande richiamo, se non tramite plugin. Allo stato dell'arte corrente, risulta essere disponibile un solo tool che propone un approccio particolare per eseguire retopology. L'approccio adottato prevede l'utilizzo di una struttura geometrica detta *patch*, e al paragrafo 4.5 viene dettagliato questo approccio del

tutto originale. Risulta difficile ipotizzare nuovi approcci, anche perché ci sono molti aspetti da tenere in considerazione. In primis, l'interfaccia tramite cui il modellatore interagisce con la nuova tecnica: se non è ben studiata, può essere richiesta una fase di apprendimento che può inficiare sulle tempistiche di esecuzione.

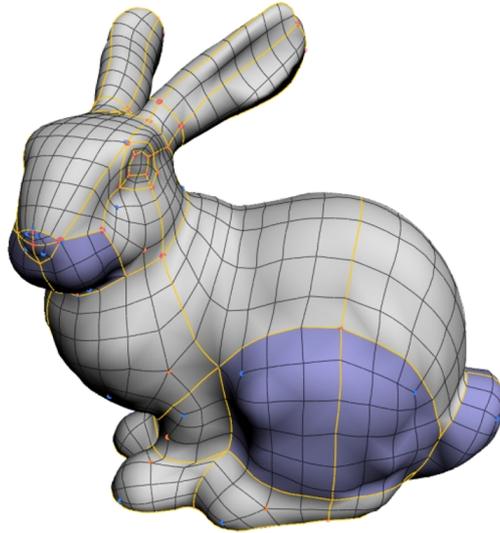


Figura 3.7: Con un approccio a patch, non si agisce direttamente sui vertici, bensì su entità geometriche superiori, quali le patches.

3.4.3 Remeshing automatico

La generazione automatica di mesh poligonali è ciò a cui più propriamente ci si riferisce con il termine *remeshing*. Benché questo tema sia argomento di studi da diversi anni (si veda [1]), solo recentemente gli algoritmi capaci di tale operazione vengono inclusi nei principali software commerciali.

Questi algoritmi eseguono complessi calcoli sulla mesh in input, tipicamente volti a capire le forme dell'oggetto studiandone i punti di curvatura. In base ai risultati ottenuti, generano e posizionano i quadrangoli. Nonostante la complessità, assicurano tempi di esecuzione abbastanza rapidi. Tuttavia, questa rapidità si paga con la scarsa possibilità di intervento sul risultato generato. Spesso, l'unica possibilità di editing consiste nel tracciare curve a mano libera che consentono di orientare sommariamente i quadrangoli. In generale, si ha uno scarso controllo sulle proprietà topologiche della mesh, poiché viene tutto gestito in automatico dal sistema. Dunque, è facilmente intuibile come l'applicazione di questi metodi sia limitata ai casi nei quali la mesh non deve possedere particolari caratteristiche topologiche.

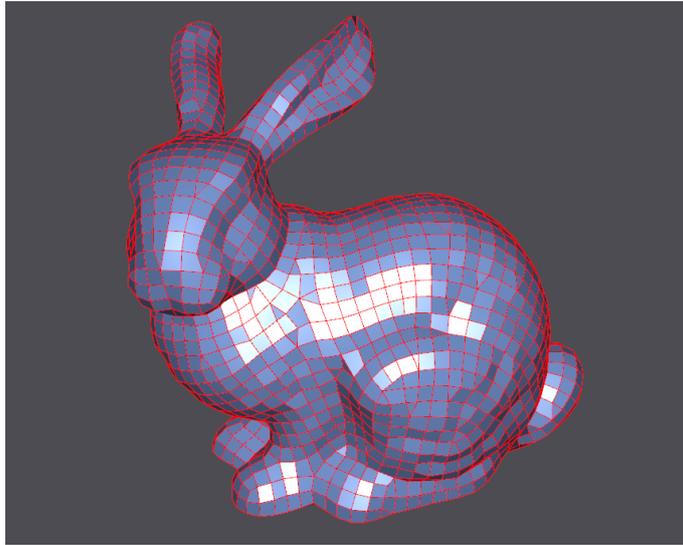


Figura 3.8: Complessi algoritmi di remeshing automatico sono in grado di produrre mesh poligonali richiedendo un limitato intervento al modellatore.

3.5 Baking dei dettagli

Il processo di retopology causa inevitabilmente una perdita di dettagli. Indipendentemente dalla tecnica impiegata, la nuova mesh non è in grado di contenere i dettagli del modello iniziale. Questo è dovuto principalmente a due fattori. Innanzitutto, l'obiettivo principale della retopology è quello di fornire una topologia adeguata al modello, in modo tale da poterlo texturare ed animare correttamente. Tenere in considerazione tutti i dettagli in conformità ad una topologia funzionale risulta una mansione pressoché impossibile, oltre a richiedere notevoli abilità del modellatore.

Secondariamente, se anche ciò fosse possibile, il modello generato risulterebbe molto denso di poligoni, ed una mesh così densa si presta poco a contesti interattivi. Bisogna quindi effettuare una sorta di selezione tra i dettagli, in modo tale da stabilire quale sia il dettaglio più fine che si intende rappresentare durante il remeshing. Tuttavia, in molti contesti è necessario che il modello retopologizzato contenga anche i dettagli più fini, affinché risulti interscambiabile con col modello originale. La soluzione comunemente impiegata sfrutta le moderne tecniche raggiunte dal processo di texture mapping. L'idea è quella di immagazzinare le informazioni relative ai dettagli più fini in particolari mappe 2D. Le mappe più comunemente utilizzate sono la *normal map* e la *displacement map*. Queste adottano approcci diversi per immagazzinare le informazioni sul rilievo, ma entrambe servono a simulare i dettagli che non sono stati rappresentati poligonalmente. La figura 3.11 mostra la resa finale delle due textures. Usarle insieme consente di incrementare notevolmente la resa grafica del modello, anche se in molti contesti è sufficiente utilizzare solo la prima.

La generazione di queste textures avviene tramite un'operazione detta *baking*. La maggior parte dei software di grafica 3D consentono di eseguire il baking anche di altre mappe, ad esempio per simulare effetti come l'occlusione ambientale. Tuttavia, esistono software ottimizzati per eseguire questa operazione, come *xNormal*³, *Allegorithmic Substance Painter*⁴ oppure *Quixel Suite*⁵.

3.5.1 Normali e Normal Map

Le *normali* sono uno degli elementi essenziali di un modello tridimensionale. Una normale è un vettore a modulo unitario ortogonale alla superficie di ogni poligono che compone la mesh. Viene determinato tramite prodotto vettoriale, ed il suo maggiore impiego risulta nel calcolo degli effetti di ombreggiatura (*shading*) ed illuminazione. Algoritmi noti come il *Phong shading* ed il *Gouraud shading* calcolano l'illuminazione valutando l'angolo tra la luce incidente e la normale alla superficie.

Nei sistemi moderni, non viene considerata una sola normale per ogni poligono, bensì una normale per ogni vertice di ogni poligono (si guardi la figura 3.9 (a)). Ciò consente di produrre un cambiamento più graduale dell'ombreggiatura tra poligoni adiacenti, poiché le normali vengono condivise da più poligoni. In questo caso, si dice che lo shading del modello è *smooth*. In caso contrario, lo shading è *hard*. Oltre al vettore normale, per ogni vertice vengono considerati due ulteriori vettori: il vettore *tangente* e *bitangente*. Questi sono ortogonali sia al vettore normale, sia tra di loro, e le loro direzioni vengono ricavate sfruttando le coordinate dei vertici nel texture space. I tre vettori formano il *tangent-space normal*, ovvero il sistema di riferimento tipicamente adottato per calcolare le normali ai vertici (si guardi la figura 3.9 (b)). Questo sistema di riferimento garantisce che le normali vengano calcolate rispetto ai punti dell'oggetto stesso, utile soprattutto quando questo viene deformato. Se fossero calcolate rispetto al sistema di riferimento globale (*world-space normal*), si riscontrerebbero incongruenze nell'illuminazione del modello. La normal map non è altro che una texture 2D i cui pixel contengono informazioni su questi tre vettori, immagazzinate nei canali (R,G,B). Rispettivamente, nei canali (R,G,B) troviamo i valori di normale, bitangente e tangente, normalizzati e scalati per adattarsi all'intervallo di valori di ciascun canale. Per ogni pixel, la normale viene calcolata tramite interpolazione delle normali ai vertici. Questa particolare texture può essere vista come una mappa di scostamento, in cui il colore di un pixel indica uno scostamento più o meno notevole dal vettore normale presente al vertice. Modificando i valori delle normali alla superficie, si cambia l'illuminazione della superficie, simulando la presenza di dettagli durante il rendering

³ xNormal, [48]

⁴ Allegorithmic Substance Painter, [34]

⁵ Quixel Suite 2, [43]

a video. In figura 3.10 è mostrato e descritto al dettaglio un esempio di normal map. Benché il sistema di ricorrere al tangent-space normal sia condiviso da qualunque applicazione, a fare la differenza è *come* questo spazio vettoriale viene interpretato dall'applicazione. La considerazione da effettuare riguarda l'API grafica alla base del programma che gestisce i modelli tridimensionali. Le due grandi famiglie di API grafiche sono *DirectX* e *OpenGL*. Queste leggono il tangent-space normal in modi differenti. OpenGL considera tutti e tre i versori positivi, mentre DirectX considera il versore della bitangente negativo. Ciò è dovuto al fatto che il verso della direzione V del texture space è orientato verso il basso. Pertanto, anche il canale *Green* della normal map risulta essere negativo. Entrambe le librerie vengono continuamente impiegate nelle applicazioni, pertanto è necessario calcolare la normal map in funzione dell'applicazione target considerata.

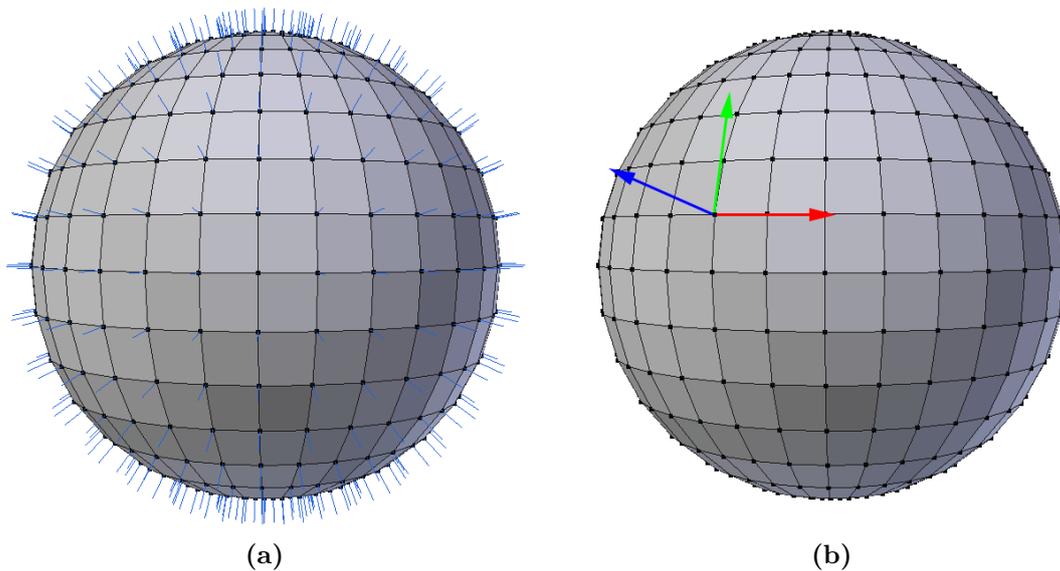


Figura 3.9: In blu sono indicate le normali ai vertici di una mesh poligonale (a). Per ogni vertice, i vettori normale, bitangente e tangente formano la base vettoriale del *tangent-space normal* (b). I valori di questi vettori vengono immagazzinati rispettivamente nei canali blu, verde e rosso della normal map.

3.5.2 Displacement Map

Il principale vincolo imposto dalla normal map consiste nel limitarsi a simulare gli effetti di rilievo senza effettivamente ricrearli sulla superficie. Questo effetto è chiaramente visibile quando l'oggetto proietta ombre su una superficie piana. La displacement map va oltre questo vincolo, creando veri e propri rilievi sulla mesh. L'elevazione dei vertici avviene lungo la relativa direzione normale. Dunque, questa

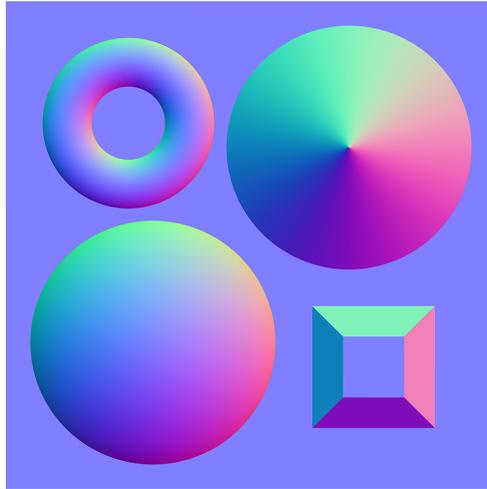


Figura 3.10: Nel *tangent-space*, i vettori possono avere valori compresi nell'intervallo $[-1,1]$, ad esclusione del vettore normale, con valore compreso in $[0,1]$. Invece, utilizzando 24bit per pixel, i canali (R,G,B) di ciascun pixel contengono valori compresi nell'intervallo $[0,255]$. Avviene un'operazione di normalizzazione e scalamento, in cui l'intervallo $[-1,1]$ dei vettori viene normalizzato nell'intervallo $[0,255]$ dei canali del pixel. Pertanto, il vettore normale $(0,0,1)$ viene rappresentato dal pixel $(127,127,255)$, che corrisponde ad un violetto. Ciò indica che la normale al vertice non deve essere modificata. Tonalità tendenti al rosso indicano valori elevati del vettore tangente, mentre tendenti al verde valori elevati della binormale.

mappa contiene informazioni sullo scostamento geometrico da applicare ad ogni vertice.

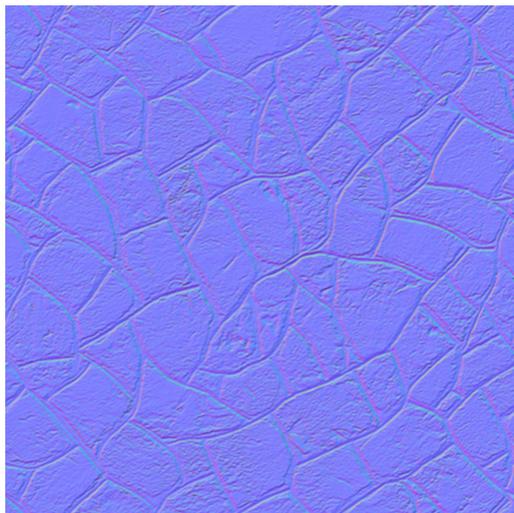
La displacement map si presenta come un'immagine in scala di grigi, dove il valore di grigio di ogni pixel rappresenta la quantità di offset da applicare al vertice. I pixel più chiari rappresentano scostamenti maggiori, mentre quelli neri non modificano i vertici. Per questa sua proprietà, questa mappa viene chiamata anche *height map*. Sebbene questa mappa consenta di generare superfici complicate ricorrendo ad una semplice immagine, gli aspetti più delicati riguardano la mesh a cui tale mappa viene applicata. Al fine di produrre un risultato di qualità, è necessario che la risoluzione della mesh sia piuttosto alta. Se la mesh contiene pochi vertici, la mappa di scostamento non produce gli effetti desiderati, poiché solo pochi pixel individuano un vertice nella corrispondente posizione. Allo stesso tempo, una displacement map produrrà scostamenti più fini quando la sua risoluzione sarà maggiore. Inevitabilmente, i tempi di elaborazione e rendering vengono aumentati, impedendo talvolta un impiego real-time del modello.



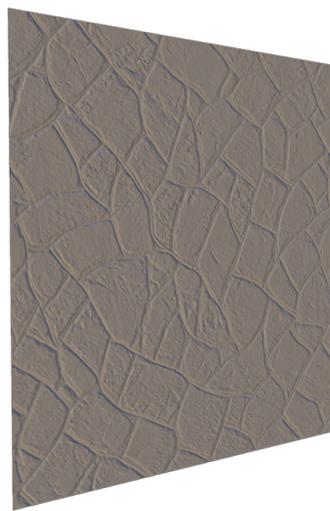
(a)



(b)



(c)



(d)

Figura 3.11: La displacement map si presenta come una texture in scala di grigi (a), mentre la normal map come una texture tendente al viola (c). La displacement map modifica la mesh, traslando i vertici e creando effettivamente dei rilievi (b). Invece, la normal map si limita a simulare il rilievo modificando solamente i valori delle normali e, di conseguenza, anche l'illuminazione (d). Tuttavia, la displacement map esige più geometria per poter offrire una resa migliore.

Capitolo 4

Strumenti per eseguire Retopology

4.1 Panoramica

Fino a qualche decennio fa il panorama dei software per CGI era poco variegato. Vi erano pochi programmi in grado di creare e gestire modelli tridimensionali, e spesso richiedevano l'acquisto di una licenza per poterli utilizzare. Una possibile classificazione poteva essere eseguita solamente tenendo in considerazione il settore di impiego: meccanico, design, intrattenimento oppure architettonico. La loro caratteristica principale era l'integrazione di molte funzioni e strumenti finalizzati a compiere diverse operazioni. Con questo si intende la loro capacità di gestire qualunque fase della pipeline di produzione, a partire dalla modellazione fino all'animazione, passando per il texture mapping e comprendendo anche simulazioni fisiche. Questa caratteristica li rendeva molto versatili, al punto da poterli definire generalisti. In questa categoria rientrano Autodesk *3D Studio Max*[®] e *Maya*[®], oltre a *Blender*[™].

Al contrario, al giorno d'oggi l'offerta è decisamente più ampia, e include sia soluzioni commerciali, sia open-source. Tuttavia, il principale cambiamento non consiste nell'aspetto economico, quanto più nel loro impiego. Sebbene continuano ad esserci programmi generalisti, nel mercato vengono immessi sempre più pacchetti software in grado di eseguire solamente una determinata operazione. Troviamo quindi software ottimizzati per generare rendering di alta qualità, altri per creare mappature uv con poche distorsioni, mentre altri ancora per realizzare simulazioni fisiche reali. Ad esempio, programmi come Allegorithmic *Substance Designer*[®] e *Substance Painter*[®] sono dedicati alla creazione di texture maps di alta qualità e con immediatezza, mentre un software come Headus *UVLayout*[®]¹ è unicamente finalizzato all'un-wrap di una mesh. A lato della modellazione, le principali novità riguardano i programmi ottimizzati per la modellazione tramite sculpt 3D. È il caso di Pixologic *ZBrush*[®] e *Sculptris*[®], ma anche Pilgway *3D-Coat*[®].

¹ Headus UVLayout, [40]

Sebbene anche i software generalisti cerchino di offrire possibilità sempre più ampie, nell'industria si preferisce ricorrere a soluzioni ottimizzate. In virtù della pipeline mostrata in figura 3.2, è importante garantire piena comunicazione tra tutti i software impiegati nelle varie fasi di lavoro. Ma le novità comprendono anche programmi dedicati ad eseguire retopology. In base alla trattazione del paragrafo 3.4, la distinzione proposta delle tecniche di retopology tiene in considerazione l'approccio adottato. Approcci orientati ai vertici oppure ai quadrangoli sono tradizionali, mentre esistono tecniche che sfruttano altre strutture oppure automatizzano interamente o parzialmente l'operazione. Tipicamente, i software generalisti consentono di retopologizzare i modelli ricorrendo a tecniche tradizionali, mentre i programmi più specifici includono ulteriori strumenti che consentono di superare l'approccio orientato ai singoli vertici.

Vengono quindi mostrate diverse alternative disponibili, facendo sempre riferimento alla trattazione a pagina 27. Al capitolo 5 si esploreranno queste alternative su un caso concreto.

4.2 Blender

*Blender*TM è un programma open-source multiplatforma, distribuito sotto licenza GPL. In base alla classificazione precedentemente fornita, appartiene alla categoria dei software generalisti. Consente di modellare, texturare, animare e renderizzare scene tridimensionali. Fornisce inoltre strumenti base per la scultura 3D, oltre ad un sistema di simulazione dei fluidi ed un game engine integrato. Nonostante non venga massivamente impiegato nell'industria, la sue funzionalità lo rendono un software molto valido. Inoltre, il libero accesso al codice consente agli sviluppatori di creare e distribuire più facilmente plugins, spesso integrati sotto forma di *addons*, script che consentono di aggiungere funzionalità e strumenti. Particolarità di *Blender*TM è l'impiego dei *modificatori*, ovvero specifici effetti non distruttivi applicabili ad una mesh, quale suddivisione o deformazione. Anche per la fase di retopology, *Blender*TM offre interessanti alternative che meritano di essere approfondite. Alcune rientrano nella categoria delle tecniche tradizionali, mentre altre utilizzano un approccio più particolare. Tuttavia, pochi altri software concedono così tante alternative per retopologizzare modelli.

4.2.1 Snap tool & Shrinkwrap

Questa modalità è la più elementare possibile, e necessita di alcune operazioni di preparazione iniziali. La mesh retopologizzata viene costruita a partire da un piano, le cui dimensioni determineranno in buona parte le dimensioni finali dei quadrangoli della nuova mesh. Successivamente, è necessario attivare la funzione snap, la quale può funzionare secondo varie modalità. Ad esempio, è possibile stabilire se i nuovi

vertici dovranno essere adagiati alle facce, ai lati, oppure ai vertici della mesh ad alta risoluzione. Risulta essere comodo aggiungere un modificatore per suddividere il piano iniziale, poiché consente di avere un maggiore dettaglio lavorando però con meno vertici. Infine, per mantenere piena corrispondenza con il modello scansionato, è indispensabile aggiungere il modificatore che attiva la funzione shrinkwrap, in cui si indica la mesh su cui si posizioneranno i nuovi vertici.

Con questa modalità, l'intero processo di retopology risulta del tutto simile ad una modellazione poligonale classica, in cui si traslano vertici ed estrudono lati. Grazie ai settaggi iniziali, il sistema si occupa in automatico di adagiare i nuovi vertici sul modello ad alta risoluzione. Come discusso in 3.4.1, la funzione shrinkwrap richiede una certa corrispondenza topologica tra le due mesh per avere un effetto positivo: in questo caso, essendo la mesh costruita a mano, tale corrispondenza è assicurata.

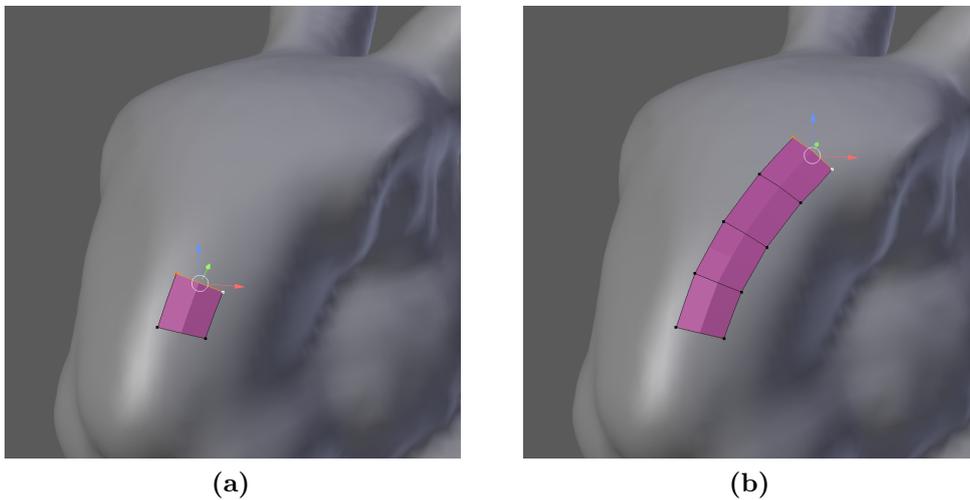


Figura 4.1: A partire da un piano (a), il modello viene retopologizzato tramite estrusione (b). Con questa tecnica, la retopology si traduce in una modellazione poligonale classica.

4.2.2 Bsurfaces e SpeedRetopo add-ons

Blender[™] offre un'altra interessante modalità di retopology tramite l'add-on *Bsurfaces*, di default incluso nel programma. Tramite questo utile script, è possibile disegnare a mano libera un insieme di curve sulla superficie del modello ad alta risoluzione. Per tracciare le curve, *Blender*[™] mette a disposizione la *Grease Pencil*, ovvero un pennello virtuale da attivare insieme al plugin. Dopo aver tracciato almeno due curve, il sistema crea una superficie i cui quadrangoli sono allineati secondo tali curve. È possibile cambiare il numero di suddivisioni per ogni coppia di lati della superficie, aumentando o diminuendo il numero di quadrangoli lungo una direzione

piuttosto che un'altra. Rispetto al sistema precedente, non richiede settaggi iniziali, poiché la funzione di snap non è necessaria. Lo snap viene garantito dalle curve stesse tracciate sulla superficie. È utile attivare la funzione shrinkwrap, per seguire meglio le forme del modello. Questa modalità di retopology viene formalizzata dal plugin esterno *SpeedRetopo*². Questo script automatizza l'attivazione della Grease Pencil e imposta in automatico la funzione shrinkwrap con la mesh selezionata. Offre un'interfaccia che rende più accessibili i comandi più frequentemente utilizzati, come il mirroring della mesh oppure il relax dei vertici. Tra i vari comandi offerti, uno dei più particolari è *Retopo MT*, che consente letteralmente di creare strisce di quadrangoli con un andamento definito da una curva tracciata a mano libera.

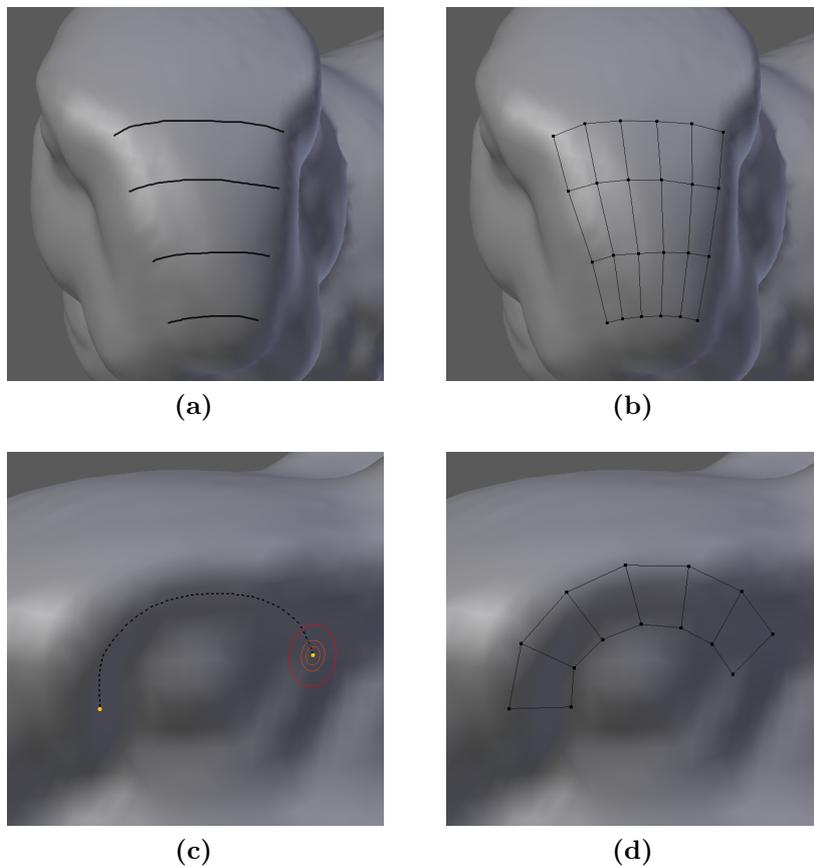


Figura 4.2: L'add-on *Bsurfaces* consente di disegnare una serie di curve (a), in base alle quali viene aggiunta una superficie poligonale (b). Con il plugin *SpeedRetopo*, viene aggiunta la funzione *Retopo MT* (c), che genera letteralmente una striscia di quadrangoli (d).

² SpeedRetopo, [45]

4.3 Autodesk Maya

Maya[®] è un software commerciale prodotto dalla famosa azienda americana Autodesk. È un programma generalista, ma il suo maggiore campo d'impiego è l'animazione, sia per l'industria cinematografica che videoludica. L'interfaccia user-friendly ed il vasto set di strumenti per modellazione, texture mapping e animazione lo hanno reso lo standard de facto dell'industria dell'intrattenimento. Relativamente all'operazione di retopology, *Maya*[®] offre un solo strumento per eseguire retopology, ovvero il *Quad-Draw*. È uno strumento molto intuitivo che opera secondo uno schema basilare. Un certo numero di vertici vengono posizionati manualmente sulla mesh, ed il sistema si occupa di individuare i quadrangoli che possono essere generati connettendo a gruppi di quattro tali vertici. Con questo strumento è possibile creare unicamente facce quadrangolari. Successivamente, è possibile muovere sia i singoli vertici, che i lati della mesh creata. Al fine di agevolare la creazione dei quadrangoli, è possibile estrarre singoli lati oppure gruppi di lati, creando così più poligoni contemporaneamente.

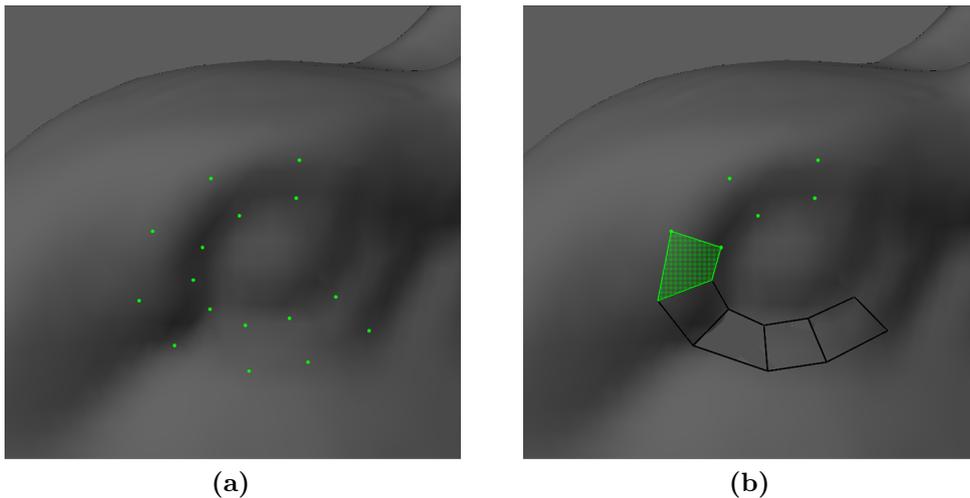


Figura 4.3: Dopo aver posizionato manualmente un certo numero di vertici (a), il sistema individua automaticamente i possibili quadrangoli che possono essere generati (b). Non è possibile generare triangoli, se non in una seconda fase di rifinitura.

4.4 Pilgway 3D-Coat

3D-Coat[®] è un programma specializzato nelle operazioni quali sculpt 3D, texture painting e retopology. Proprio in quest'ultimo ambito, *3D-Coat*[®] si distingue per essere uno dei più completi software disponibili sul mercato. Oltre a mettere a

disposizione un vasto set di strumenti per retopology manuale, incorpora anche un sistema di remeshing automatico.

Quando si intende effettuare retopology manuale, il sistema attiva in automatico la funzione snap, consentendo di iniziare subito con la costruzione vera e propria della mesh. Gli strumenti consentono di: creare un singolo quadrangolo posizionando consecutivamente i quattro vertici; aggiungere un quadrangolo a partire dal lato di un quadrangolo già esistente; posizionare un certo numero di vertici e fare individuare al software i possibili quadrangoli generabili; disegnare curve a mano libera sulla superficie, le cui intersezioni individuano i nuovi vertici. In generale, ripropongono i metodi presenti in *Maya*[®] e *Blender*[™] con un'altra ottica. Questi strumenti rendono *3D-Coat*[®] adatto a qualunque tipologia di superficie, sia organica che geometrica.

Il programma dispone anche di un algoritmo di retopology automatica, chiamato *Autopo*. Data una mesh in input, il modellatore indica innanzitutto il massimo numero di triangoli che comporranno la mesh retopologizzata. Successivamente, è possibile indicare le zone in cui si desidera una maggiore densità di poligoni. Il sistema cerca in automatico di individuare un edge-flow appropriato in base alle caratteristiche di curvatura del modello. Tuttavia, a causa delle imperfezioni che possono essere presenti sulla mesh, l'edge-flow risultante può non essere ottimale. Dunque, il modellatore può specificare l'andamento desiderato dei poligoni tracciando a mano libera delle curve sulla superficie. Queste verranno usate come linee guida per orientare i quadrangoli della nuova mesh.

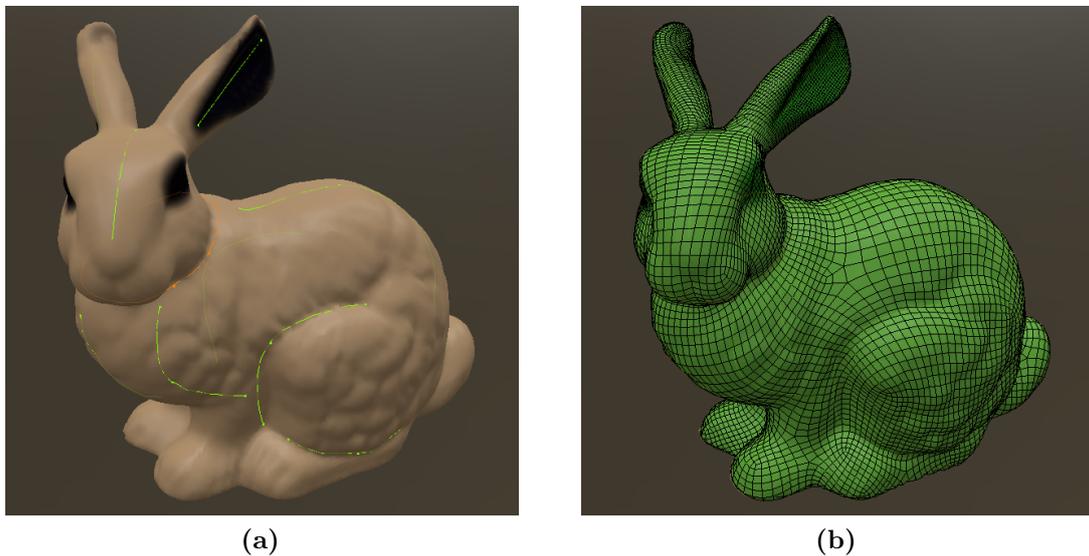


Figura 4.4: Dopo aver marcato le zone più sensibili (in nero) e disegnato un set di curve sulla superficie (a), *Autopo* genera automaticamente una mesh quadrangolare (b).

4.5 SketchRetopo

SketchRetopo ([8], [44]) è un tool di retopology presentato durante la conferenza SIGGRAPH del 2013. È stato sviluppato dal gruppo di ricerca Interactive Geometry Lab, appartenente all’Institute of Visual Computing dell’università ETH Zurich. Questo programma introduce un approccio del tutto originale nel campo del remeshing. L’obiettivo è quello di poter superare il conflitto esistente tra le tecniche di retopology manuale, che garantiscono pieno controllo sulle caratteristiche topologiche, ed i sistemi di retopology automatica, che riducono i tempi di produzione e richiedono meno sforzi al modellatore. Il concetto alla base è quello di poter generare interattivamente una mesh quadrangolare tramite un sistema di curve disegnato a mano libera dall’utente. A differenza delle tecniche basate su curve presenti in *Blender*[™] e *3D-Coat*[®] (sezioni rispettivamente 4.2.2 e 4.4), le intersezioni creano una struttura geometrica chiamata *patch*. Sulla base degli input forniti dall’utente, questa viene automaticamente quadrangolata dal sistema, che si occupa inoltre di inserire automaticamente i vertici irregolari laddove necessari. La mesh finale è quindi costruita tramite la congiunzione di patch quadrangolari, e risulta di conseguenza interamente quadrangolare.

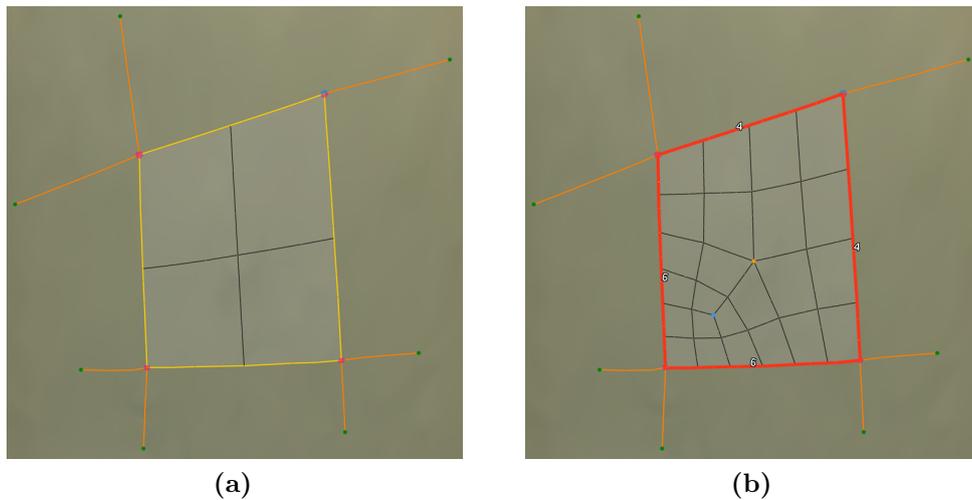


Figura 4.5: La patch viene automaticamente creata e quadrangolata dal sistema (a), ma il modellatore può modificarla impostando le suddivisioni ai lati della patch (b). Le singolarità vengono automaticamente inserite.

4.5.1 Creazione di una patch

Una patch è sostanzialmente una struttura poligonale formata da un insieme di quadrangoli. Il modellatore disegna a mano libera un insieme di curve sulla superficie

in input. Quando quattro curve formano una regione chiusa, questa viene automaticamente quadrangolata. Il modellatore può modificare la topologia all'interno della patch cambiando il numero di suddivisioni per ogni lato e muovendo i vertici irregolari. Il sistema fornisce supporto anche per le patches triangolari, pentagonali ed esagonali.

4.5.2 Codifica del sistema di curve

Ciascuna curva tracciata viene densamente campionata, creando nuovi punti detti *curve network vertices* e nuovi segmenti detti *curve network halfedges*. Quando due curve si intersecano in un punto, il sistema assegna un *flag corner* a tutti gli halfedges incidenti in quel punto. Il sistema individua anche le *halfchains*, ovvero l'insieme di halfedges che congiungono due vertici. Quando più halfchains consecutive formano un anello chiuso, viene generata una patch. Ogni halfchain è associata ad una patch, e ad ogni patch vengono associate le halfchains che la compongono.

4.5.3 Pattern topologici

Alla base dell'algoritmo di quadrangolazione vi è il seguente presupposto: il numero di lati di confine di qualunque mesh quadrangolare è pari³. La dimostrazione di tale affermazione è presente in [7]. Per estensione, è possibile quadrangolare una patch solo se la somma del numero di suddivisioni presenti in ogni lato è pari. Verificata questa condizione, la patch viene quadrangolata ricorrendo a *pattern topologici* prestabiliti dagli stessi sviluppatori. La selezione avviene in funzione del numero di suddivisioni presenti in ogni lato, stabilito dal modellatore. Una volta scelto il pattern, è necessario calcolare il valore di specifici parametri al fine di soddisfare determinate relazioni tra le suddivisioni impostate ai lati. I parametri consistono nel numero di edge-loops e nel numero di intersezioni tra essi. Inizialmente, i pattern sono stati stabiliti solamente per le patches triangolari e quadrangolari [7], mentre successivamente sono state studiate anche le patches con cinque e sei lati [6].

4.5.4 Modalità di editing

Il programma offre diverse modalità secondo cui può lavorare. Ogni modalità è accessibile con la pressione di un preciso tasto, ed è possibile passare da una modalità all'altra senza soluzione di continuità.

- **Basic sketching.** Questa modalità consente di disegnare le curve sulla mesh da retopologizzare. La funzione snap, automaticamente attivata dal sistema,

³ [7], p. 1

consente di ancorare una nuova curva all'estremo di una già esistente. È possibile inoltre ancorare l'estremità di una curva ad un punto centrale di un'altra curva, creando una *T-junction*. Un set di curve formanti una regione chiusa generano una patch, che viene automaticamente quadrangolata.

- **Edit topology.** In questa modalità si possono cambiare le suddivisioni presenti ad ogni lato della patch. Il sistema si occupa in automatico di aggiungere le eventuali singolarità. Il numero di suddivisioni può essere aumentato solamente di due unità alla volta, al fine di soddisfare la condizione enunciata al paragrafo 4.5.3 (dimostrata in [7]). Se sono presenti eventuali singolarità, nelle patches quadrangolari è possibile traslarle manualmente, mentre in quelle triangolari o pentagonali vengono messe a disposizione delle frecce, indicanti la direzione della traslazione.
- **Deform curve:** Questa modalità permette di eseguire modifiche alle curve. È possibile traslare sia gli estremi della curva, sia i punti centrali, in modo da deformarla. Con una combinazione di tasti è inoltre possibile ridisegnare l'andamento di una curva.
- **Edit corner** L'obiettivo di questa modalità è quello di gestire i vertici presenti agli angoli della patch, consentendo di eliminarli oppure di cambiarne la valenza. Se la valenza è pari a due, il vertice è detto *corner*, altrimenti *non-corner*.
- **Cylinder sketching** Quando si hanno due curve chiuse adagiante su piani paralleli, con questa modalità è possibile generare i quadrangoli lungo la direzione che collega i due piani. Il sistema genera automaticamente altre quattro curve lungo tale direzione. È possibile effettuare tale operazione anche nel caso di due curve concentriche.

4.6 Instant Meshes

Instant Meshes ([17], [41]) è un tool di remeshing automatico, sviluppato dal gruppo di ricerca Interactive Geometry Lab dell'università ETH Zurich. È stato presentato durante conferenza SIGGRAPH del 2015. Questo programma, in grado di elaborare sia mesh molto dense sia nuvole di punti, è in grado di generare mesh quadrangolari e triangolari in breve tempo.

L'algoritmo è suddiviso in due fasi principali: il calcolo dell'*orientation field* ed il calcolo del *position field*. L'*orientation field* consiste in un insieme di direzioni, visualizzate sulla superficie in input, che fanno da guida per l'orientamento dei poligoni e l'allineamento dei lati della mesh finale. L'algoritmo è studiato in modo tale da garantire un preciso allineamento senza interventi da parte del modellatore. La seconda fase calcola una parametrizzazione (u,v) della mesh in input, la quale viene

impiegata per posizionare uniformemente i poligoni ed allineare correttamente i lati. Dopo queste due operazioni vengono estratte le posizioni dei vertici, successivamente connessi per creare il reticolo poligonale finale.

Il più importante vantaggio offerto è la velocità di esecuzione dell'algoritmo. Questo vantaggio è stato sfruttato inserendo un set di strumenti interattivi che possono essere utilizzati per modificare alcune caratteristiche della mesh in output. Con questi strumenti, il sistema combina parzialmente metodi di remeshing automatico con metodi di retopology manuale. I principali strumenti interattivi messi a disposizione consistono in una coppia di pennelli virtuali. Con un tipo di pennello è possibile modificare l'andamento delle direzioni dell'orientation field, in modo da garantire un migliore allineamento dei quadrangoli. Un altro tipo di pennello invece lavora sul position field, ed è in grado di forzare la creazione di catene di lati in una determinata direzione. Le singolarità vengono automaticamente inserite dal sistema, mentre appositi strumenti permettono di riposizionarle.

L'algoritmo genera mesh con alto grado di isotropia (si veda pagina 14). Le proprietà dei quadrangoli non variano in base alla direzione considerata, generando quindi una tassellatura piuttosto uniforme. Di default, la mesh finale contiene una certa percentuale di triangoli e N-gon. Tuttavia, è possibile forzare la creazione di solamente quadrangoli prima di estrarre la mesh. Per far ciò, il sistema applica uno step dell'algoritmo di Catmull-Clark [2].

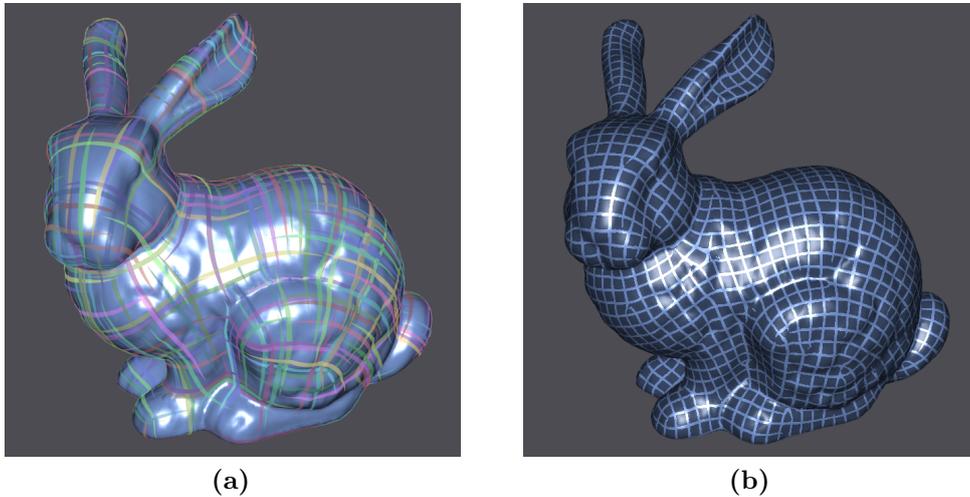


Figura 4.6: L'algoritmo inizia con il calcolo dell'orientation field, visualizzato tramite una serie di linee colorate sulla mesh (a). Successivamente, calcola il position field (b), dal quale vengono poi estratti i vertici.

Capitolo 5

Retopology delle statue Sekhmet

5.1 Contestualizzazione

Negli ultimi decenni, la conservazione, visualizzazione e divulgazione dei beni materiali che costituiscono il nostro patrimonio culturale hanno assunto forme del tutto nuove. Le storiche funzioni di conservazione e tutela dei beni sono correlate ormai ad una visione museale piuttosto classica. Al giorno d'oggi, queste funzioni sono state notevolmente arricchite facendo ricorso alla *digitalizzazione* dei beni materiali. La conversione da reale a digitale permette di avere un modello binario impiegabile in più contesti, quali un progetto di restauro oppure conservazione. In aggiunta, l'impiego delle moderne strutture di telecomunicazione ha permesso la divulgazione e condivisione della cultura su scala mondiale. Su questi presupposti sono nate molteplici iniziative che offrono agli utenti nuovi metodi di fruizione del sapere.

5.1.1 Librerie digitali e 3D repositories

Le più classiche iniziative hanno portato alla creazione di librerie digitali liberamente consultabili online dagli utenti, creando veri e propri *musei virtuali*. Uno dei primi progetti di questo tipo è *Europeana*¹, lanciato nel 2008 e promosso dall'Unione Europea, con l'obiettivo di creare una biblioteca digitale contenente il patrimonio dell'intero continente europeo. In esso è possibile consultare opere d'arte, libri e manoscritti in formato digitale. Le possibilità offerte dal digitale consentono di creare ulteriori nuovi metodi di fruizione che vanno oltre la semplice visualizzazione. Ad esempio, il progetto *Google Arts & Culture*² nasce principalmente come contenitore di opere d'arte a livello internazionale, ma offre anche la possibilità di esplorare virtualmente diversi musei e luoghi caratteristici situati in qualunque parte del

¹ Europeana Collections, [21]

² Google Arts & Culture, [23]

mondo. Questa modalità di fruizione richiama molto l'idea di gamification, ovvero la divulgazione del sapere tramite un contesto videoludico.

I *3D repositories* sono collezioni di modelli digitali dotati di tridimensionalità. Oltre a poter essere visionabili online, queste raccolte offrono la possibilità di interagire con i modelli 3D, ruotandoli oppure ingrandendoli. Questa è la missione di *African Fossils*³, il quale si occupa di scansionare tridimensionalmente i reperti archeologici per poi renderli disponibili e visualizzabili online dagli utenti. Anche singoli musei hanno avviato iniziative atte a favorire la visualizzazione tridimensionale dei reperti da loro custoditi. È il caso del *The British Museum*⁴ e del *Museu d'Arqueologia de Catalunya*⁵, i quali si sono serviti della piattaforma *Sketchfab*⁶ per rendere disponibili pubblicamente i propri modelli digitali⁷.

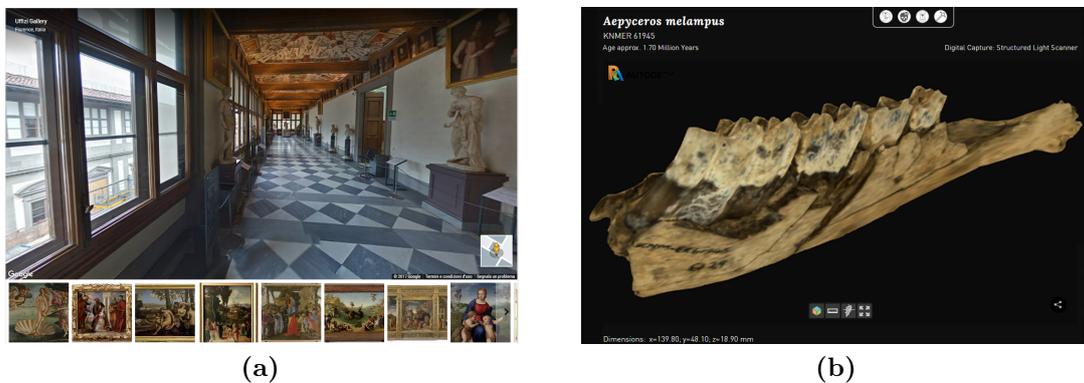


Figura 5.1: Le possibilità offerte dal digitale consentono di sviluppare nuovi metodi di fruizione dei beni. Con l'iniziativa *Google Arts & Culture*, si propone un'interazione basata sull'esplorazione, caratteristica tipica dei videogiochi open world (a). *African Fossils* invece consente di osservare un reperto e di interagire con esso potendolo ruotare e ingrandire (b).

5.1.2 Scansione di beni tridimensionali

Digitalizzare un bene culturale è sempre un compito di notevole portata. Bisogna tenere in considerazione aspetti legati tanto alla qualità della digitalizzazione quanto alla non-invasività del processo stesso, che potrebbe deteriorare il reperto. Nel caso dei beni dotati di tridimensionalità, il processo diventa più oneroso e richiede le fasi

³ African Fossils.org, [19]

⁴ The British Museum, [30]

⁵ Museu d'Arqueologia de Catalunya, [25]

⁶ Sketchfab, [27]

⁷ 3D repositories: The British Museum [29], Museu d'Arqueologia de Catalunya [28]

di lavorazione discusse a pagina 22. Il principale aspetto da tenere in considerazione è la dualità tra la capacità di creare un modello digitale verosimile a quello reale e la richiesta di risorse computazionali per supportare tale verosimiglianza durante la visualizzazione all'utente finale⁸. La verosimiglianza è la caratteristica chiave che rende interscambiabili il modello fisico con la sua versione digitale. Ciò è indispensabile soprattutto nei contesti di documentazione o restauro, ma in generale serve a mantenere immutato il valore culturale attribuito al modello. Come spiegato al paragrafo 3.2, la scansione 3D è la migliore tecnica per garantire un altissimo grado di attendibilità del modello digitale. Tuttavia, genera un'ingente mole di dati non facilmente gestibili dai comuni calcolatori in contesti interattivi. Questa è la conseguenza di un fatto ben noto: la precisione della scansione è determinata dal dettaglio più piccolo che si intende catturare⁹. Ciò fissa la frequenza con cui la superficie dell'oggetto viene campionata, la quale si traduce nella distanza tra i punti che costituiscono la point cloud.

5.1.3 Decimazione

A valle del processo di acquisizione, spesso viene eseguita l'operazione di *decimazione*. Questa operazione consente di ridurre il numero di triangoli che compongono la mesh, basandosi principalmente sulla ridondanza presente nel modello finale, localizzata soprattutto nelle zone planari del modello. Queste necessitano di poche facce per poter essere rappresentate con un buon grado di fedeltà. Tuttavia, avendo campionato la superficie del modello con una frequenza molto elevata, anche queste zone risultano dense di triangoli. I moderni algoritmi di decimazione sfruttano la coplanarità di gruppi di triangoli adiacenti, unendoli in modo da ottenere un solo triangolo. Sebbene con questa tecnica si possa ridurre sensibilmente il peso poligonale del modello in brevissimo tempo, nell'ambito di questo progetto la sua applicazione risulta piuttosto limitante per una serie di ragioni. Dopo aver decimato il modello, non vi è più corrispondenza tra la mappatura UV e la texture iniziali. A causa della rimozione di alcuni triangoli, molti texel non trovano un poligono a cui corrispondere, e ciò causa notevoli distorsioni quando si riapplica la texture al modello decimato, come mostrato in figura 5.3. È dunque necessaria una nuova mappatura UV del modello decimato, in modo da poter operare la procedura di baking per prelevare correttamente l'informazione cromatica dal modello scansionato. Tuttavia, il modello decimato risulta difficile da rimappare nel texture space, a causa della presenza di una topologia del tutto caotica (si veda la figura 5.2). Si potrebbe ricorrere a tecniche di mappatura UV automatica, disponibili in qualunque software di computer grafica,

⁸ [16], p. 4

⁹ [16], p. 6

ma il risultato non è ottimale. Generalmente vengono create numerose shells, la maggior parte delle quali molto piccole, e sono presenti molte distorsioni.

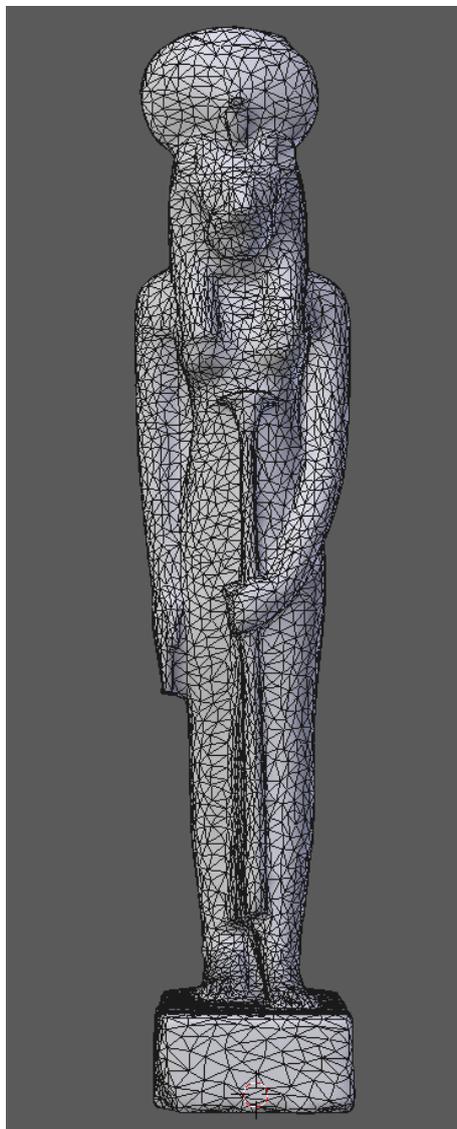


Figura 5.2: Modello ottenuto a seguito della decimazione. sebbene il numero di triangoli sia drasticamente diminuito, la topologia risulta del tutto disordinata. Ciò rende impossibile l'operazione di UV-Mapping.

Pertanto, per ottimizzare tali modelli si è adoperato un workflow che prevede due principali fasi. La prima è la creazione di modelli poligonali più leggeri utilizzando le tecniche di retopology discusse al capitolo 4, mentre nella seconda è stata adoperata la tecnica di baking per arricchire i modelli retopologizzati con metodi image-based. La figura 5.4 mostra il workflow impiegato in questo progetto.

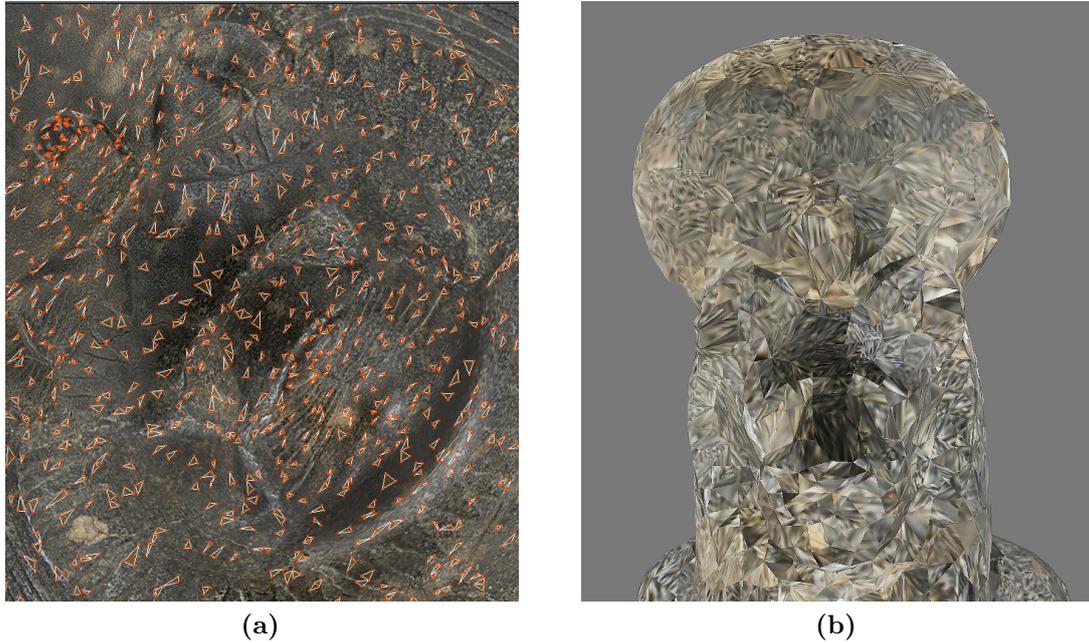


Figura 5.3: A seguito del processo di decimazione, non è più possibile riutilizzare la texture iniziale. L'assenza di molte facce nel texture space (a) causa vistose distorsioni sul modello (b).

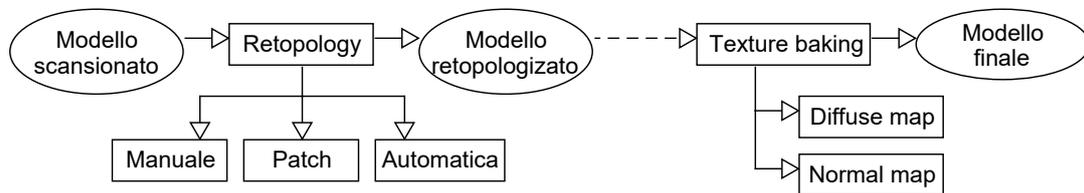


Figura 5.4: Workflow con le fasi fondamentali del lavoro eseguito sulle statue Sekhmet.

5.2 Fase di scansione

Le statue *Sekhmet* risalgono al Nuovo Regno egizio, periodo storico che comprende le dinastie XVII, XIX e XX. Esistono più di seicento statue, risalenti al periodo 1390–1352 a.C., il cui scopo è venerare la dea Sekhmet. Alcune di queste sono state scansionate 3D per volontà del *Museo Egizio di Torino*¹⁰. La scansione è stata eseguita ad opera dello studio Azimut¹¹.

¹⁰ Museo Egizio di Torino, [24]

¹¹ Studio Azimut, [20]

5.2.1 Scansione laser con luce strutturata

Per le statue Sekhmet, è stata impiegata la tecnica di scansione laser basata su luce strutturata. Questa tecnica appartiene alla categoria delle tecniche di scansione attive, le quali basano il loro funzionamento sull'emissione di un raggio luminoso che colpisce l'oggetto da rilevare. Queste tecniche sono non a contatto, e sono le più utilizzate poiché non danneggiano l'artefatto.

In generale, un sistema a scansione laser è basato sul principio della triangolazione, in cui l'oggetto da scansionare, la sorgente luminosa ed il sensore di cattura formano un triangolo. Attraverso una fase di calibrazione, si definiscono sia l'inclinazione della sorgente luminosa, sia la distanza tra sorgente luminosa e sensore. Questi valori restano costanti durante tutto il processo di acquisizione. Il raggio laser emesso dalla sorgente è in grado di colpire sia un singolo punto dell'oggetto, sia più punti allineati contemporaneamente, in modo da catturare un profilo completo dell'oggetto. Nel caso dei sensori a luce strutturata, un particolare proiettore emette contemporaneamente un certo numero di lame di luce sulla superficie. Questo insieme di lame prende il nome di *pattern*. Tipicamente, il pattern è formato da una sequenza in cui si alternano verticalmente strisce luminose e strisce scure. La camera digitale si occupa di catturare le deformazioni che il pattern subisce durante la proiezione sulla superficie dell'oggetto. Ogni striscia luminosa genera un profilo, il quale mappato sul sensore della camera pixel per pixel. Per ogni punto del profilo vengono identificate le deviazioni rispetto agli assi del sistema di riferimento del sensore. Ciò serve per calcolare la distanza tra lo scanner e l'oggetto, punto per punto e per ogni lama di luce. La qualità della scansione dipende fortemente dal numero di lame emesse dalla sorgente. Più strisce luminose e strisce scure vengono alternativamente emesse, più fine sarà il campionamento, dunque maggiori dettagli verranno catturati. In figura 5.5 è rappresentato schematicamente il sistema di triangolazione adottato.

Generalmente, non viene proiettato un singolo pattern, ma più pattern consecutivamente, che differiscono per il numero di linee luminose emesse. Ad ogni step, viene raddoppiato il numero di strisce, arrivando fino ad un valore massimo. Questa prassi è utile per identificare con più precisione ogni singola linea del pattern, poiché aumentando notevolmente il numero di linee si potrebbero avere ambiguità nell'associare un preciso punto ad una linea di luce. Per far ciò si stabilisce una codifica binaria in cui, per ciascun pattern, ogni punto viene etichettato come illuminato oppure no, ricorrendo ad una codifica 0 oppure 1. La sequenza completa di numeri binari, associata al singolo punto, riconduce in modo inequivocabile alla striscia di luce emessa.

A seguito della fase di acquisizione, sono state eseguite le operazioni di filtraggio, meshing, allineamento e rifinitura, discusse in 3.2.1. Queste sono indispensabili per poter generare una mesh triangolare conforme.

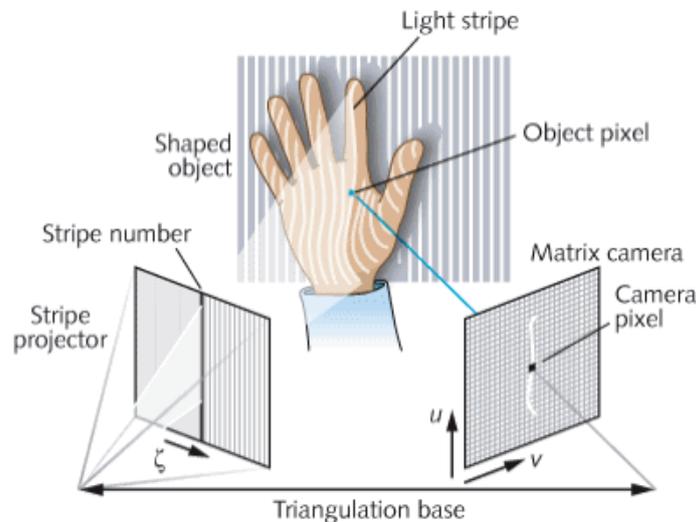


Figura 5.5: Sistema di triangolazione per la scansione tramite scanner laser a luce strutturata.

5.2.2 PDF interattivi

Oltre all'informazione geometrica, è stata catturata anche quella cromatica, salvata in una apposita texture ad alta risoluzione. Ciascuna mesh tridimensionale è stata impacchettata con la relativa texture in un file in formato *.PDF* (*Portable Document Format*). Il formato *.PDF* è molto comodo per la condivisione di documenti, poiché i contenuti vengono rappresentati in modo indipendente dal software utilizzato per generarli. Ciò consente di poter inserire qualsiasi tipo di contenuto con la sicurezza che venga visualizzato da chiunque alla stessa maniera. Oltre ad essere contenuti testo, immagini, font e oggetti di grafica vettoriale, è possibile includere anche geometrie tridimensionali texturate. Il programma Adobe *Acrobat Reader DC*¹² consente di visualizzare facilmente questi contenuti, ed offre la possibilità di interagire col modello.

Per aggiungere i contenuti tridimensionali ai file *.PDF* è stato impiegato Adobe *Photoshop CC*¹³. Il famoso programma di fotoritocco dell'azienda Adobe offre la possibilità di gestire i modelli tridimensionali impiegando uno spazio di lavoro a tre dimensioni anziché due. Non mette a disposizione gli strumenti di editing offerti dai programmi di modellazione 3D come *Maya*[®] oppure *Blender*[™], tuttavia consente di esportare una scena tridimensionale direttamente in formato *.PDF*, includendo in automatico anche le textures (si noti la figura 5.6). Vista l'ingente quantità di

¹² Adobe Acrobat Reader DC, [31]

¹³ Adobe Photoshop CC, [32]

dati che compongono i modelli, anche i relativi file .PDF risultano essere pesanti in termini di memoria, e ciò limita la fluidità dell'interazione con il modello.

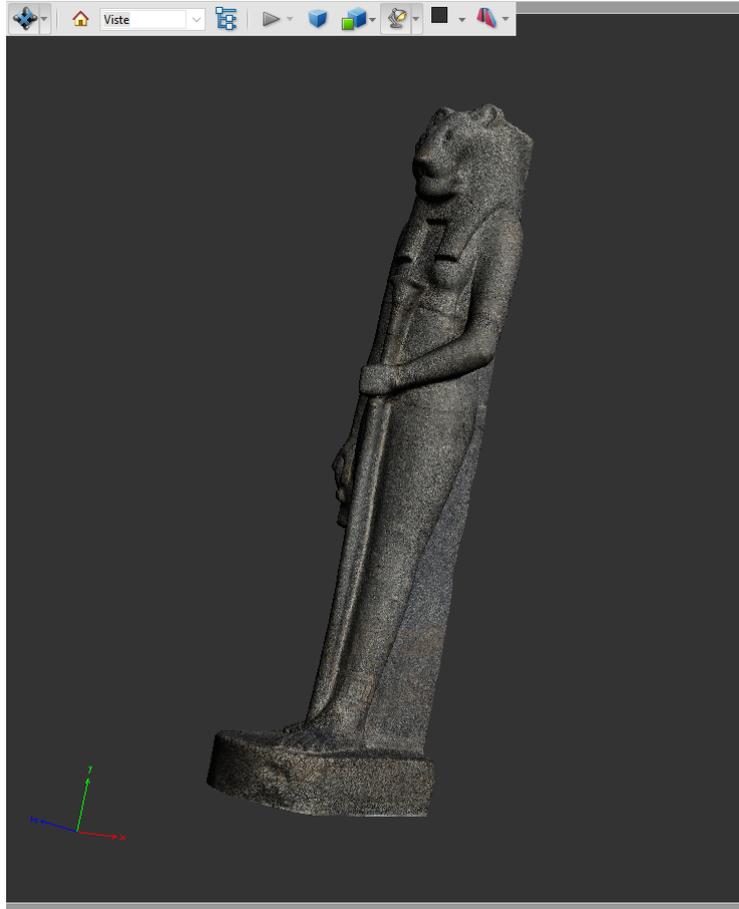


Figura 5.6: Visualizzazione di una statua scansionata all'interno del .PDF interattivo. Oltre alla geometria, è possibile visualizzare anche la texture contenente le informazioni cromatiche.

5.3 Fase di Retopology

I modelli finali delle statue sono molto dettagliati, sia per quanto riguarda le caratteristiche geometriche, sia quelle cromatiche. Questi sono i due principali aspetti che devono essere mantenuti invariati durante la digitalizzazione, al fine di conservare i valori associati all'oggetto fisico. Una geometria molto densa è necessaria al fine di poter rappresentare fedelmente le numerose crepe e irregolarità presenti sulla superficie. Questo si traduce in un considerevole aumento di poligoni. Ciascuna

statua Sekhmet contiene infatti centinaia di migliaia di poligoni. In base alle osservazioni discusse al paragrafo 5.1.3, la retopology diventa una fase essenziale per poter avanzare iniziative che coinvolgano tali modelli in contesti interattivi. Nell'ambito di questo progetto, la priorità è stata principalmente verso la riduzione del numero di poligoni necessari a rappresentare i vari modelli, al fine di poterli inserire in un contesto videoludico. Il trade-off si colloca sempre tra il mantenimento delle forme originali e la diminuzione delle risorse computazionali richieste.

In base alla discussione effettuata nel capitolo 3, esistono molti programmi e tools ottimizzati per eseguire retopology. In questo contesto, la scelta è stata indirizzata in modo tale da poter sperimentare diversi approcci e strumenti, al fine di poter eseguire una trattazione quanto più completa delle varie alternative disponibili. Pertanto, sono stati analizzati i metodi offerti sia dai software generalisti quali *Blender*TM e *Maya*[®], sia da un software più specifico quale *3D-Coat*[®]. I primi due sono stati scelti per via del loro massiccio impiego nel settore dell'entertainment, mentre l'ultimo per la vasta gamma di strumenti offerti. Con il tool *SketchRetopo* [8] è stata sperimentata anche la retopology tramite patches, mentre con *Instant Meshes* [17] e *3D-Coat*[®] il processo di remeshing automatico.

5.3.1 Considerazioni iniziali

Il formato .PDF si presta molto alla condivisione di documenti, poiché garantisce indipendenza dal software di creazione dei contenuti. Sebbene ciò valga anche per contenuti rappresentanti modelli tridimensionali, tale scelta non è la migliore quando è necessario eseguire operazioni specifiche su tali modelli. È stato necessario estrarre da ogni .PDF le informazioni geometriche in un formato più usuale, quale .OBJ. Questo formato, sviluppato dalla *Wavefront Technologies*, è piuttosto semplice, poiché rappresenta in modo grezzo le coordinate dei vertici sia nello spazio 3D che nel texture space, oltre alla direzione delle normali e le facce che compongono il modello. Vista la sua semplicità, può essere tranquillamente letto da qualsiasi software, al punto da venire spesso utilizzato come formato di interscambio. Anche le textures del colore sono state esportate, e salvate in formati non compressi come .TGA, .TIFF oppure .PNG. Entrambe le operazioni sono state eseguite all'interno di Adobe *Photoshop CC*[®].

Prima di iniziare la fase di costruzione della nuova mesh, è stato eseguito uno studio topologico, finalizzato a definire l'edge-flow generale dei lati. Topologicamente parlando, le statue sono suddivisibili in due categorie: quelle erette, e quelle in posizione seduta. Ciò ha reso possibile la condivisione di alcune caratteristiche topologiche, quali l'edge-flow e alcune singolarità, tra più statue. Inoltre, sono figure parzialmente organiche, pertanto è stata studiata anche la posizione di alcune singolarità, soprattutto nelle zone di volto, mani e piedi (si osservi la figura 5.8). Un limite imposto dalle statue è la loro caratteristica asimmetrica, che non ha

consentito di accelerare il processo di retopology tramite i moderni sistemi di mirroring automatico offerti da qualunque software. È stato necessario modellare ad hoc entrambi i lati di ogni statua.

Un'ultima considerazione riguarda i danni presenti sulle statue. Alcune di esse risultano molto danneggiate, altre invece solo sommariamente. Naturalmente, non è possibile rappresentare tutti i danni poligonalmente, anche perché andrebbe in contraddizione con le finalità di questo progetto. Dunque, è stato necessario stabilire la minima entità di danno da rappresentare attraverso i poligoni. Questo comporta una prima, ma necessaria, approssimazione delle forme originali.

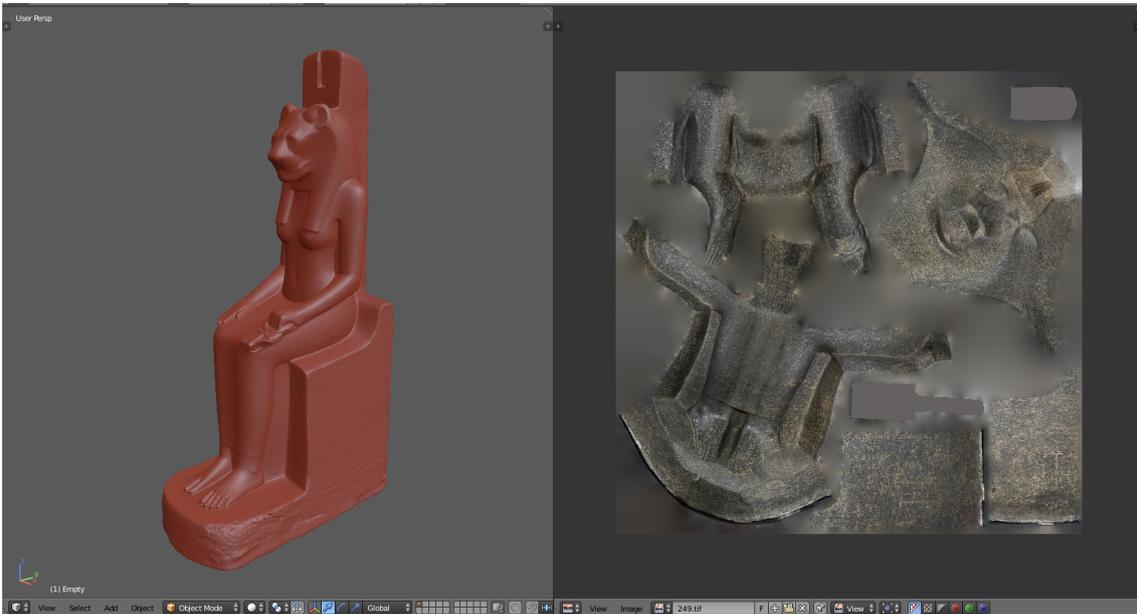


Figura 5.7: Da ogni .PDF sono state estratte sia le informazioni geometriche che quelle cromatiche. Ciò ha permesso di utilizzare i modelli in software di grafica 3D più tradizionale, come *Blender*TM.

5.3.2 Retopology manuale

Come visto al paragrafo 3.4.1, le numerose tecniche tradizionali consentono di lavorare unicamente sulle entità minime che costituiscono una mesh, quali vertici, lati e facce. Esperimenti di retopology manuale sono indispensabili, poiché resta tutt'oggi l'approccio più diffuso nella maggior parte dei casi. Pertanto, sono stato impiegati software generalisti come *Blender*TM e *Maya*[®], ed un software più specializzato in retopology come *3D-Coat*[®], al fine di eseguire un'analisi più completa.

Vista la complessità delle statue Sekhmet, quasi tutte le tecniche di retopology manuale si adattano a tali modelli, al punto da impiegarne più di una contemporaneamente su una singola statua. Gli approcci a curve (descritti in 4.2.2 e 4.4) si

rivelano alquanto comodi da utilizzare nelle zone più geometriche dei modelli, quali il piedistallo e la colonna posteriore, ma meno nelle parti organiche, quali volto, mani e piedi. Questo è dovuto al fatto che, per assicurare un buon edge-flow, in queste zone è necessario inserire alcune singolarità a valenza cinque. L'approccio a curve non agevola in questo senso la creazione di una topologia adatta. Pertanto, sono stati posizionati manualmente i vertici, basandosi su un preventivo studio topologico eseguito in 2D, mostrato in figura 5.8. Nelle zone più geometriche, invece, l'edge-flow è determinato dalla forma stessa, senza necessariamente ricorrere alle singolarità: le curve possono essere semplicemente posizionate lungo gli spigoli.

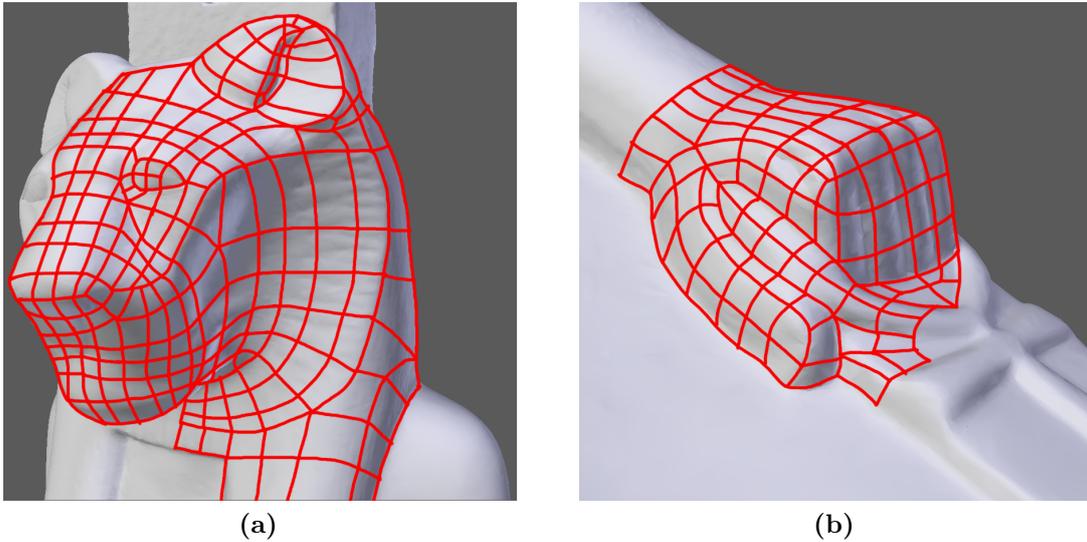


Figura 5.8: Tracciare delle linee a mano libera aiuta a definire le caratteristiche topologiche della mesh finale. Nelle parti organiche, questa fase iniziale è stata molto utile. Le singolarità nel volto (a) e nella mano chiusa (b) consentono di distribuire i lati in modo tale da creare un edge-flow quanto più ordinato e funzionale per la fase di texture mapping.

A causa dell'ingente quantità di danni che affliggono alcuni modelli, è inevitabile ricorrere al posizionamento manuale dei vertici. Per rappresentare al meglio queste zone, è infatti necessario far variare la densità di poligoni, e, come visto al paragrafo 2.5, ciò necessita dell'inserimento di alcune singolarità.

5.3.3 Approccio a patch

In base alla trattazione a pagina 42, *SketchRetopo* consente di lavorare unicamente con una struttura geometrica detta patch, contenente al suo interno un numero variabile di quadrangoli. Il sistema, nel suo complesso, risulta facile da apprendere e comodo da utilizzare. Si occupa di adagiare automaticamente i vertici interni delle patches sulla superficie, implementando una sorta di shrinkwrap per ogni patch.

Prima di retopologizzare ogni singola statua, è stato studiato un sistema di curve di base in grado di garantire caratteristiche topologiche favorevoli per la fase di texture mapping. Lo studio è stato effettuato per entrambe le categorie di statue, ovvero quella eretta e quella seduta.

È fondamentale capire come generare le patches sulla superficie di ogni statua, poiché le caratteristiche topologiche dipenderanno principalmente da queste. Per ottenere l'edge-flow ideale, inizialmente si possono tracciare le curve in corrispondenza degli spigoli più netti della mesh. Successivamente, è necessario capire dove aggiungere le singolarità, responsabili di un edge-flow ordinato. Per fare ciò, bisogna sperimentare col numero di suddivisioni per ogni lato della patch. Le interfacce offerte dal software (discusse in 4.5.4) consentono di retopologizzare anche le zone organiche con molta naturalezza, dove è necessario un maggiore controllo sulle singolarità e sugli edge-loops (figura 5.9 (a)). Con queste modalità, risulta agevole anche la creazione di edge-loops tra patches adiacenti.

L'inserimento automatico delle singolarità in funzione delle suddivisioni ai lati agevola il cambiamento di densità di quadrangoli laddove necessario (si veda 2.5). Soprattutto nelle zone di mani e piedi è indispensabile avere quadrangoli più piccoli, per poter seguire al meglio le forme.

Il sistema a patches si dimostra meno agevole da utilizzare in corrispondenza di danni ingenti (figura 5.9 (b)). Talvolta, superfici molto complesse vengono rappresentate a fatica poiché le curve vengono considerevolmente distorte. In questo caso, il poter lavorare unicamente con le patches si rivela limitante. Per rappresentare i dettagli più fini, è necessario aumentare il numero di patches, creandone di più piccole. Sebbene la creazione di più patches non sia un problema, ciò richiede sforzi per suddividere i lati in modo tale da allinearle correttamente.

Nonostante il preventivo studio del set di curve, il numero finale di patches può variare notevolmente da statua a statua. Ciò non comporta un aumento del peso del modello finale in termini di memoria, poiché dipende unicamente dal numero di facce poligonali di cui è composto, ma aumenta le dimensioni del file che contiene le informazioni sulle curve e sulle patches.

Le mesh generate da questo tool risultano completamente quadrangolari, con forte caratteristica anisotropica (si veda pagina 2.2).

5.3.4 Retopology automatica

I metodi di remeshing automatico discussi in 4.4 e 4.6 consentono di generare una mesh con un intervento piuttosto blando da parte del modellatore. Questo consente una drastica riduzione dei tempi di produzione, poiché non si lavora sui vertici della mesh. Indipendentemente dal software scelto, questi algoritmi condividono gli stessi principi alla base. Nel caso delle statue Sekhmet, si sono valutati gli aspetti critici dei software *Instant Meshes* e *3D-Coat*[®].

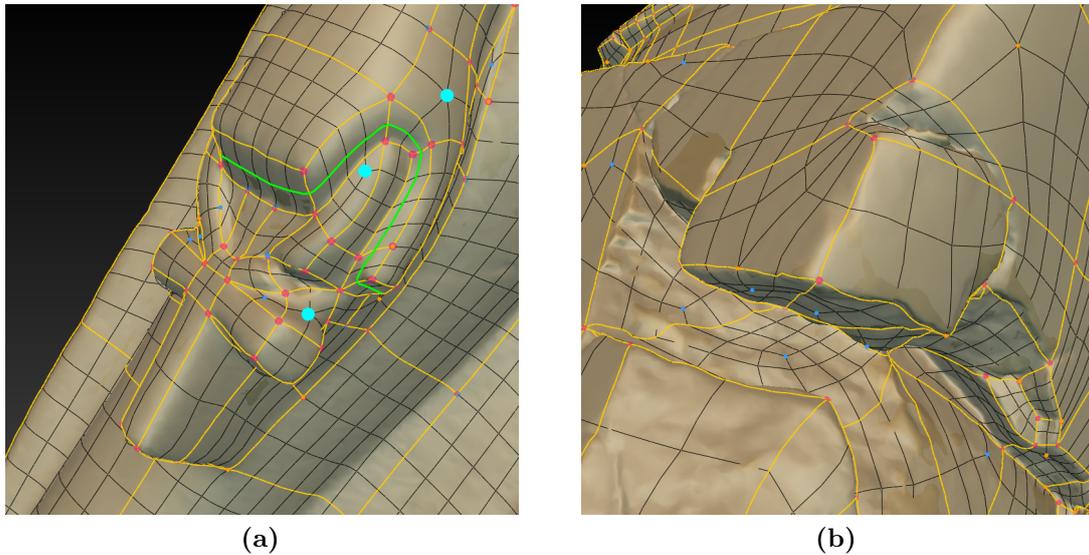


Figura 5.9: L’approccio a patch si rivela adatto anche alle parti organiche dei modelli, nelle quali i requisiti topologici (singolarità ed edge-loops) sono più stringenti (a). Tuttavia, spesso è necessario aggiungere patch veramente piccole, al fine di assicurare un edge-flow equilibrato. In presenza di danni, il sistema a curve necessita di ulteriori lavorazioni (b). Si nota facilmente la caratteristica anisotropica della mesh generata.

Entrambi i software consentono di impostare il massimo numero di quadrangoli di cui sarà composta la mesh finale. Nel contesto di un remeshing automatico, questo valore è importante, poiché generalmente vale la seguente relazione: più quadrangoli si hanno a disposizione, migliore sarà la verosimiglianza con la forma originale. Aumentando in maniera considerevole il target di vertici, si ottiene una mesh quadrangolare con lo stesso dettaglio del modello scansionato. Gli algoritmi rispettano sempre il target di poligoni scelto. Tuttavia, un valore troppo elevato potrebbe andare contro l’obiettivo non ultimo della retopology, ovvero la significativa riduzione del numero di facce. Bisogna trovare il giusto trade-off tra qualità della rappresentazione e percentuale di riduzione dei poligoni. Nel contesto delle statue Sekhmet, le zone più organiche vengono rappresentate più adeguatamente con un alto numero di quadrangoli target, mentre per le parti più geometriche il risultato è pressoché indipendente dal valore impostato. Questa caratteristica della mesh, di cui si è discusso a pagina 18, è funzionale per poter assicurare ovunque una buona approssimazione della forma originale. Tuttavia, solamente *3D-Coat*[®] tiene in considerazione questo fattore. Tramite un pennello virtuale, sono state specificate le zone in cui è necessario avere più quadrangoli corrispondenti alle zone dei piedi, delle mani e del volto. Contrariamente, *Instant Meshes* non valuta questa opzione, e la tassellatura finale risulta essere uniforme. Si tratta di una quadrangolazione

isotropica, secondo cui le caratteristiche dei quadrangoli non variano in funzione della curvatura locale (si veda pagina 14). Di conseguenza, con questo software il vincolo sul numero finale di facce risulta essere molto più restrittivo.

Le mesh restituite da *Instant Meshes* non sono puramente quadrangolari, ma del tipo quad-dominant (si veda 2.2). Pertanto, sono presenti triangoli e N-gons, che possono risultare scomodi per la fase di texture mapping. Come visto a pagina 44, *Instant Meshes* offre la possibilità di avere una mesh puramente quadrangolare. Pertanto, la scelta è ricaduta sulla generazione di una mesh puramente quadrangolare.

Le tecniche di retopology automatica sono in grado di garantire un edge-flow adatto solamente nelle zone più geometriche delle statue. Invece, le zone più sensibili sono quelle organiche e il disco solare, quando presente: le caratteristiche di curvatura non sempre vengono plasmate correttamente. Questo fattore compromette la possibilità di ottenere un edge-flow ordinato e funzionale per la fase di texture mapping. Entrambi i programmi tengono in considerazione questo aspetto, seppur in modi differenti. L'idea alla base è quella di indicare l'andamento sommario dei quadrangoli attraverso curve tracciate a mano libera sulla superficie. In generale, trova riscontro pratico la considerazione secondo la quale, per avere un edge-flow corretto e ordinato, è necessario disegnare un numero considerevole di curve. *3D-Coat*[®] impone che questa fase venga eseguita prima del remeshing vero e proprio; non è possibile modificare l'andamento dei quadrangoli dopo averli generati, a meno di non ripetere l'intera operazione. Invece, *Instant Meshes* consente una modalità interattiva e più avanzata. Dopo aver calcolato il campo di orientamento, tramite un particolare pennello è possibile modificarlo, ed il sistema aggiornerà in real-time i valori del campo. Successivamente, dopo aver calcolato il campo di posizionamento, tramite un ulteriore pennello è possibile forzare la creazione di catene di lati nelle zone più complesse, quali mani, orecchie e piedi.

Per entrambi i programmi, non si ha la possibilità di intervenire minuziosamente sulle singolarità. Queste vengono automaticamente generate e posizionate dall'algoritmo, e ciò non consente di creare edge-loops funzionali per la fase di texture mapping.

5.3.5 Ottimizzazione dei modelli

La maggior parte delle mesh poligonali generate con *SketchRetopo* presentano alcuni errori topologici tali da renderle inutilizzabili nelle successive fasi di lavorazione. Sono presenti sia vertici duplicati che set di quadrangoli sovrapposti (si veda la figura 5.11). In questi casi, la mesh risultante non è conforme, in base alla definizione fornita a pagina 3. Questi difetti emergono quando si cambia la modalità di shading da flat a smooth (si rimanda a 3.5.1), poiché si notano spigoli netti. Inoltre, mostrando le normali per ogni vertice, è possibile notare tanti vettori quanti sono i vertici che condividono le stesse coordinate nello spazio. Dunque, è necessaria una fase

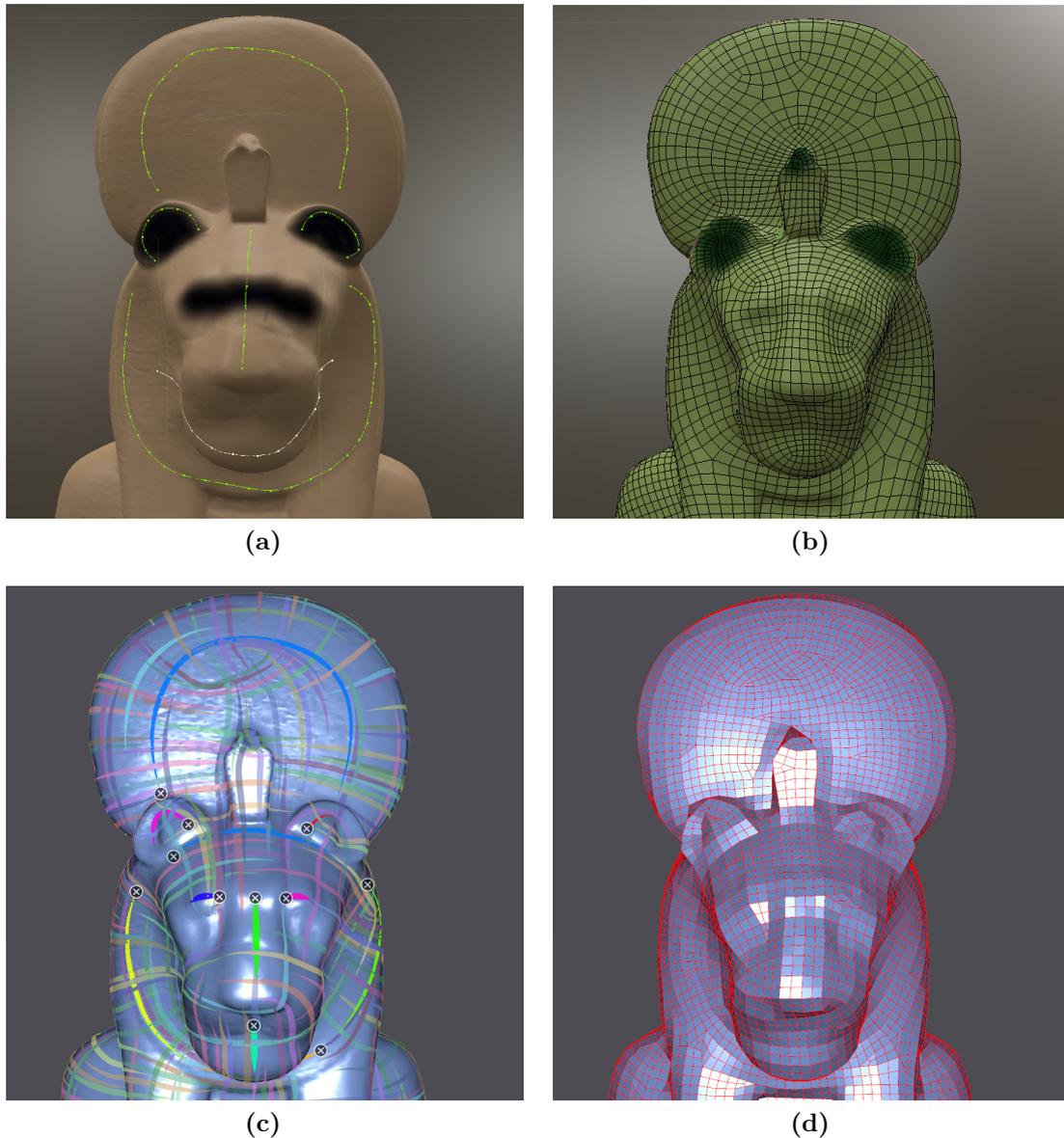


Figura 5.10: Le zone più sensibili sono quelle organiche. È necessario quindi inserire molte curve in corrispondenza di tali parti. *3D-Coat*[®] richiede di tracciare le curve prima di generare la mesh (a), utilizzate per allineare i quadrangoli (b). Invece, *Instant Meshes* offre una modalità interattiva real-time di editing (c). Sebbene gli sforzi richiesti al modellatore siano minimi, i risultati sono spesso disordinati, talvolta anche nelle zone geometriche.

di pulitura dei modelli. *SketchRetopo* non include strumenti per eseguire questa operazione, dunque è necessario ricorrere a programmi esterni. Qualunque software di computer grafica mette a disposizione strumenti che consentono di congiungere

vertici doppi automaticamente, mentre le facce necessitano di una rimozione manuale.

Di rado, anche le mesh generate con *Instant Meshes* possono risultare non conformi. Può capitare che gruppi di quadrangoli condividano solamente un vertice con l'intera mesh, risultando del tutto scollegati a questa (si guardi la figura 5.12 (a)). Ciò accade soprattutto in corrispondenza dei danni, poiché causano netti cambi di direzione. Per rimuoverli è necessario agire manualmente, ricorrendo sempre a software esterni.

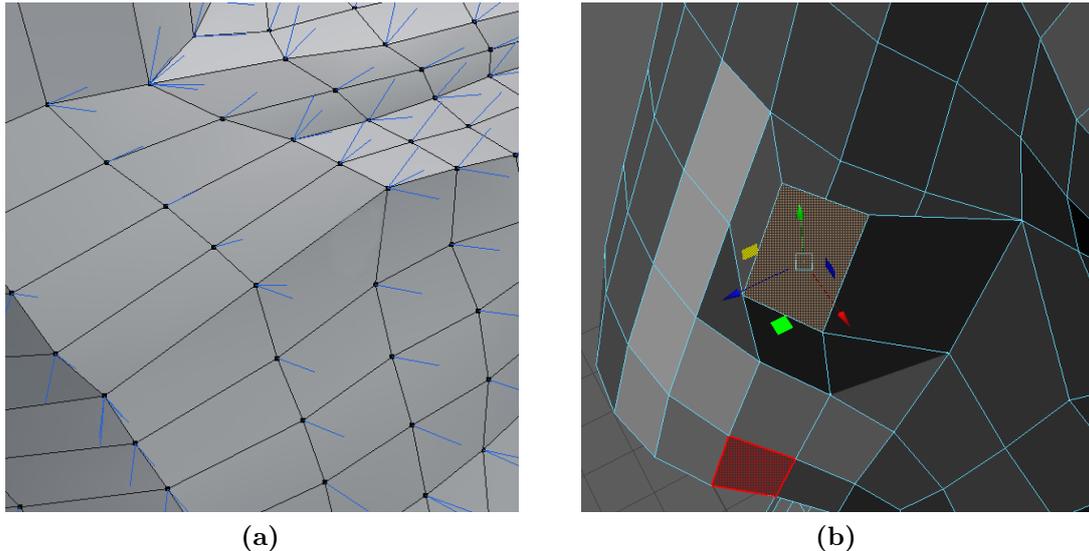


Figura 5.11: Anomalie delle mesh esportate da *SketchRetopo*. I vertici multipli (a) e le facce doppie (b) rendono inutilizzabile la mesh per la fase di uv-mapping.

Autopo cerca di generare automaticamente una mesh che soddisfi quanto più possibile le esigenze del modellatore. Può capitare che gruppi di quadrangoli risultino pesantemente distorti, poiché assolutamente non planari e non allineati agli spigoli del modello. Tuttavia, è possibile ottimizzare il modello all'interno di *3D-Coat*[®] stesso, ricorrendo agli strumenti di retopology manuale presentati a pagina 40.

5.4 Texture Mapping e Baking

Nell'ambito di questo progetto, la fase di texture mapping assume un ruolo fondamentale. Le textures sono indispensabili per raggiungere un obiettivo fondamentale, ovvero rendere indistinguibili le statue retopologizzate da quelle scansionate. Sommarientemente, questa fase può essere suddivisa in due step. Il primo prevede la creazione di una mappatura UV adeguata, che tiene in considerazione gli aspetti discussi al paragrafo 1.5. In virtù di tali considerazioni, è stato scelto il software commerciale Headus *UVLayout*[®]. La scelta del software *UVLayout*[®] è dovuta principalmente al fatto di essere un software dedicato ad eseguire mappature UV. Pertanto, offre

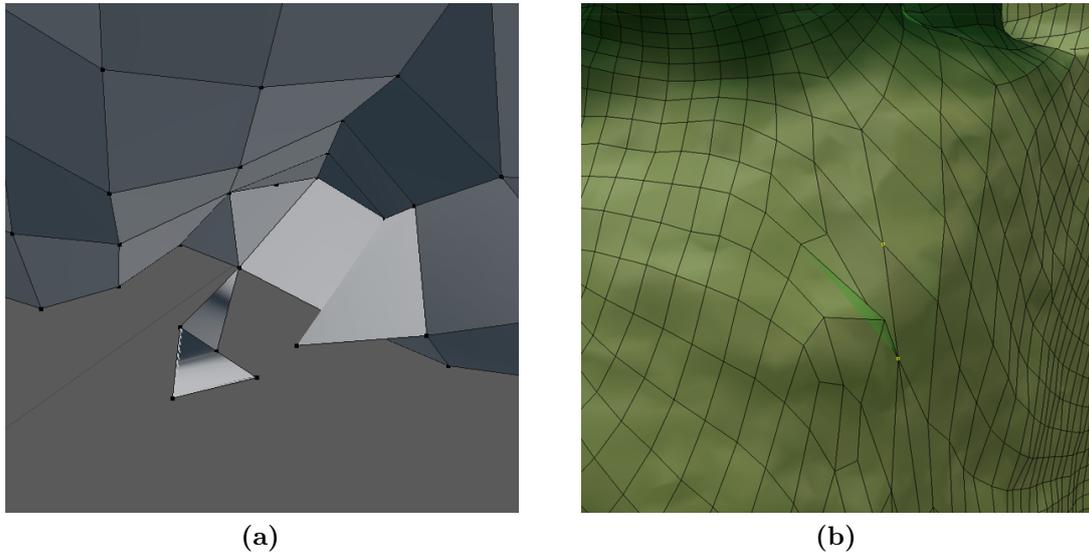


Figura 5.12: Non sempre *Instant Meshes* restituisce mesh conformi. Gruppi di quadrangoli possono essere staccati dalla mesh principale (a). È invece più frequente ottenere distorsioni con *Autopo* (b).

strumenti molto immediati da utilizzare, e semplifica alcune operazioni che, eseguite in altri software, richiedono più azioni. Inoltre, impiega un moderno algoritmo in grado di minimizzare le distorsioni causate durante la mappatura. Il secondo consiste nel recupero dei dettagli persi durante la retopology attraverso il processo di baking, dettagliato al paragrafo 3.5.

5.4.1 UV-Mapping

Gli accorgimenti fondamentali per mappare un modello nel texture space sono piuttosto indipendenti dal modello in esame. Tuttavia, nel caso delle statue Sekhmet, una considerazione utile è il fatto che ogni statua è parzialmente organica e parzialmente geometrica. Queste due parti sono state inizialmente divise per poter essere trattate con un approccio leggermente differente. Per le parti geometriche, quali piedistallo e colonna, i tagli possono essere individuati lungo gli spigoli. Posizionare i tagli nei punti di maggiore curvatura assicura poche distorsioni durante la distensione dei quadrangoli nel texture space. Inoltre, è possibile comparare parzialmente queste zone a cubi, il quale semplifica il posizionamento dei tagli. Le zone organiche, invece, necessitano di tagli unicamente in funzione delle distorsioni.

La qualità della mappatura UV dipende fortemente dalle caratteristiche topologiche del modello. I modelli che presentano una topologia adeguata, come quelli prodotti manualmente oppure tramite *SketchRetopo*, restituiscono risultati migliori. Gli edge-loops consentono di partizionare la mesh anche semanticamente, ad esempio

per separare le parti organiche da quelle geometriche. L'algoritmo di distensione riesce a minimizzare le distorsioni, localizzandole al più nelle zone organiche con forte curvatura. Il corrispondente layout UV risulta ordinato e con poche shells (si guardi la figura 5.13 (a)).

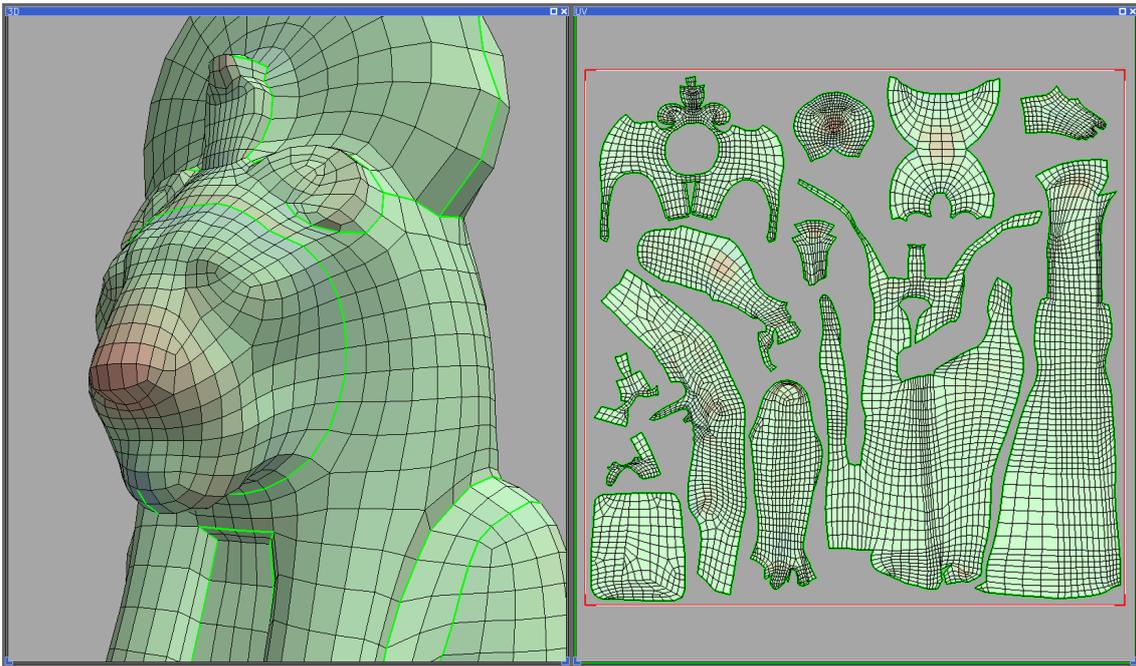
Queste considerazioni non valgono per i modelli generati in automatico. L'assenza di edge-loops funzionali non consente di settare, se non con molto lavoro, i seams in posizioni funzionali. Mesh con una topologia così disordinata possono essere mappate ricorrendo ad algoritmi di mappatura automatica. Questi sono inclusi in qualunque software generalisti di computer grafica, ma funzionano bene con modelli non molto complessi. Lavorano valutando l'angolo formato da due quadrangoli adiacenti, notando se questo valore supera un certo valore soglia impostato dal modellatore. In generale, il risultato ottenuto è un layout UV con molte shells, la maggior parte delle quali molto piccole (si noti la figura 5.13 (b)). Una tale mappatura UV genera moltissime distorsioni quando viene applicata una texture.

5.4.2 Operazione di Baking

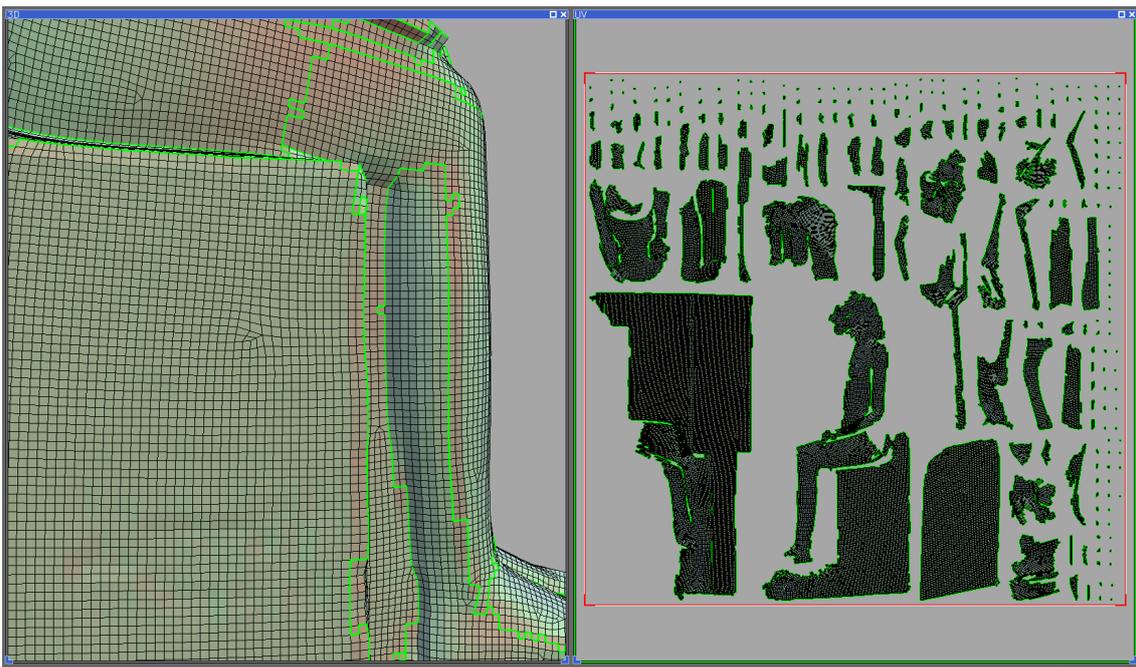
Per le statue Sekhmet, è stato eseguito il baking di due textures: quella contenente l'informazione cromatica (diffuse map) e quella contenente l'informazione sulle normali (normal map). Affinché tale operazione abbia esito positivo, è necessario che la statua scansionata e quella retopologizzata condividano la stessa posizione nello spazio e lo stesso fattore di scala. Non è stata calcolata una displacement map, poiché la sua applicazione avrebbe richiesto una geometria più ricca di quadrangoli.

La normal map è stata calcolata secondo il formato *OpenGL*, il quale considera positivi tutti e tre i vettori del tangent-space (si rimanda a 3.5.1 per un'esaustiva trattazione). Dopo aver aggiunto la normal map al modello retopologizzato, si avrà l'illusione di avere un modello molto più ricco di dettagli. Questo perché tale mappa genera cambiamenti di shading ed illuminazione che danno l'impressione di avere rilievi. Nel caso delle statue Sekhmet, la normal map è particolarmente utile per simulare le superfici frastagliate in corrispondenza dei danni, poiché solo una parte di questi sono stati rappresentati poligonalmente (si noti la figura 5.15).

La fase di baking si conclude con l'aggiunta di padding (si veda il paragrafo 1.5.3) ad ogni textures, al fine di attenuare le discontinuità presenti lungo i tagli. In figura 5.14 è mostrato un esempio delle textures finali prodotte, mentre in figura 5.16 uno dei modelli finali con entrambe le varie textures applicate. Al fine di garantire brevi tempi di caricamento e fluidità di fruizione, entrambe le textures sono state calcolate con una dimensione pari a 2048x2048 pixels. Utilizzare un formato maggiore, quale 4096x4096 avrebbe inutilmente contribuito ad aumentare le risorse richieste al sistema, rischiando di tralasciare i dispositivi con caratteristiche tecniche più povere.



(a)



(b)

Figura 5.13: La topologia influenza notevolmente i risultati ottenibili nella fase di UV-Mapping. Una topologia funzionale consente di individuare facilmente i seams, garantendo poche distorsioni ed un layout UV ordinato (a). Viceversa, una topologia caotica causa estese distorsioni, mentre l'utilizzo di una mappatura UV automatica genera un layout UV caotico (b).

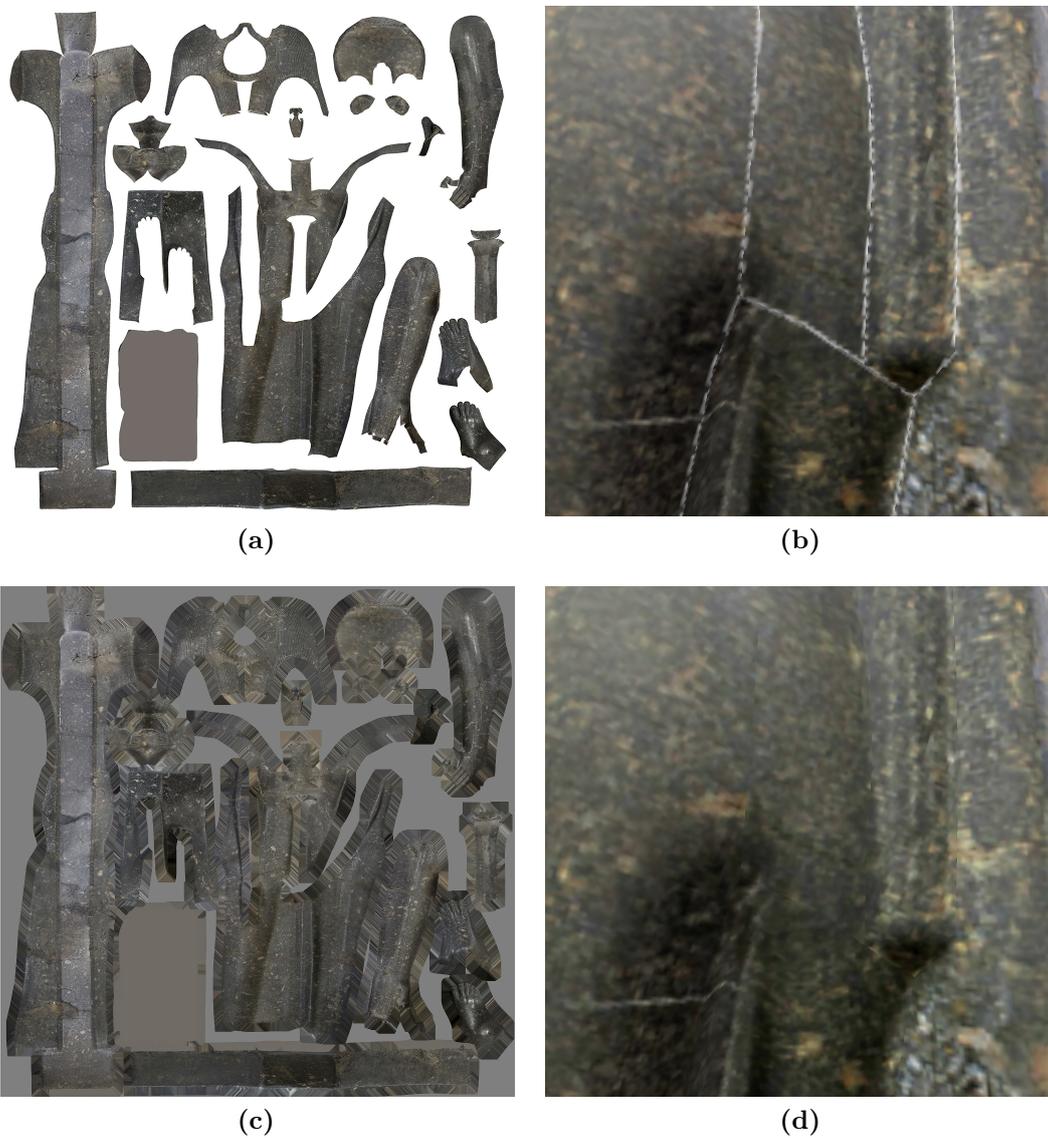
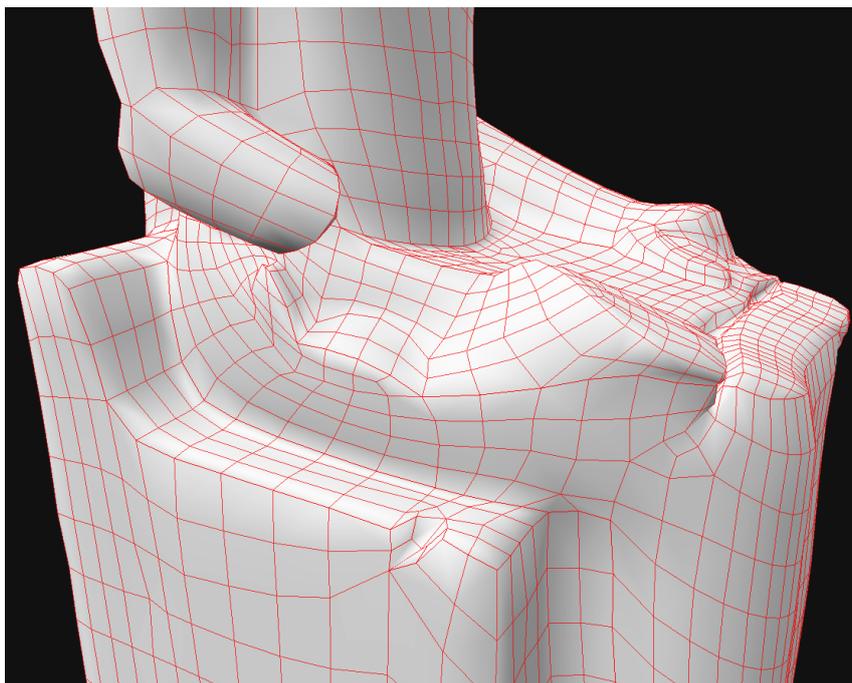
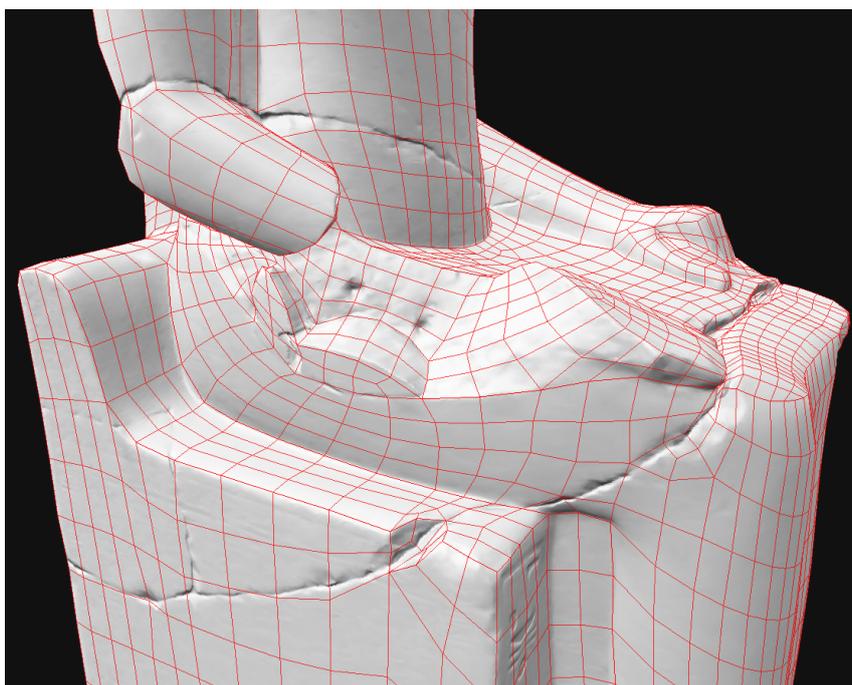


Figura 5.14: Al fine di migliorare la resa finale dei modelli, ad ogni texture è stato aggiunto del padding. Senza il padding (a) sono presenti discontinuità visibili lungo i seams (b). L'aggiunta di padding (c) nasconde questi artefatti (d).

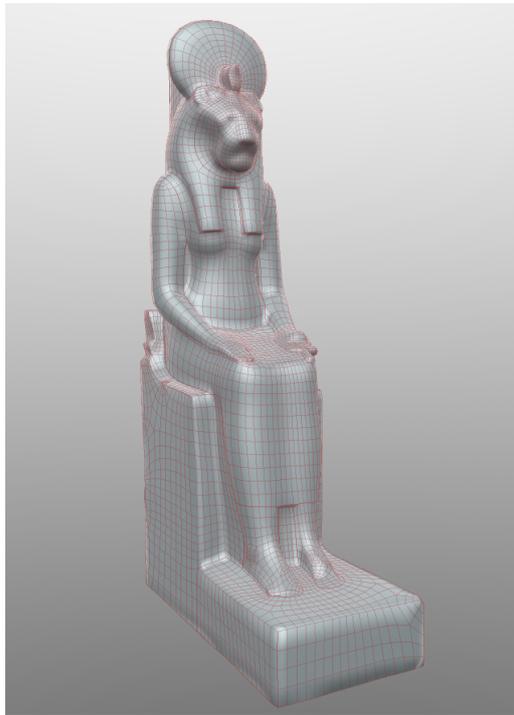


(a)



(b)

Figura 5.15: Calcolando la normal map è possibile simulare i dettagli persi, dando l'illusione che questi siano presenti poligonalmente sul modello. Essendo molte statue colme di danni, la normal map consente di rappresentarli senza aggiungere geometria al modello.



(a)



(b)



(c)

Figura 5.16: Esempio di modello finale (a), con normal map (b) e con diffuse map (c).

Capitolo 6

Discussione dei risultati

6.1 Considerazioni sui modelli

In questa sezione vengono discussi i risultati ottenuti al termine del lavoro, valutando sia gli aspetti quantitativi che qualitativi dei modelli retopologizzati. Vengono mostrati i dati relativi alla riduzione del peso poligonale dei modelli, e di come questo aspetto incida su un ambiente di visualizzazione interattivo. Vengono presi in considerazione anche gli aspetti topologici dei modelli finali. Essendo questi dipendenti dalla strategia adottata in fase di remeshing, emerge un confronto tra le possibili alternative percorribili, e del loro impatto sulla resa grafica finale.

6.2 Riduzione del peso poligonale

Il principale obiettivo di questo progetto è stato la riduzione del numero di facce di ogni statua. I modelli scansionati presentano un numero di triangoli nell'ordine delle centinaia di migliaia di unità, troppo elevato per poter essere inseriti in un contesto real-time. I modelli finali retopologizzati sono detti low poly, indicando con questo termine la ridotta quantità di triangoli che compongono la superficie. Tipicamente, un modello low poly deve rimanere nell'ordine delle centinaia o migliaia di poligoni, ma con i sistemi tecnologici moderni questo valore si è alzato, arrivando a contare fino a decine di migliaia di triangoli per un singolo modello. Non esistono valori numerici standard che consentano di capire se un modello è low poly oppure no. Tuttavia, un parametro che spesso viene preso in considerazione come veritiero è la distanza dell'oggetto dalla camera virtuale. Più l'oggetto è lontano, meno dettaglio servirà per poterlo rappresentare.

Per le statue Sekhmet, la riduzione è stata valutata in percentuale. I sistemi di retopology manuale e a patch si rivelano essere i più flessibili: le percentuali di riduzione si aggirano intorno al 96%, riuscendo a mantenere una discrepanza con i modelli originali molto limitata. È constatabile come l'approccio a patch offerto

da *SketchRetopo* consenta di variare il numero di poligoni con maggiore flessibilità rispetto ai metodi manuali, che possono richiedere di rieseguire l'intero processo.

Tendenzialmente, è possibile constatare un drastico calo del peso della geometria retopologizzata con i metodi manuali e a patch. Questo valore si colloca intorno alle centinaia di kilobyte, contro le decine di megabyte richieste dal modello scansionato. La tabella 6.2 mostra la diminuzione della geometria del modello in termini di occupazione di spazio in memoria. Ciò garantisce una fruizione anche attraverso dispositivi con hardware ridotto.

Percentuali di riduzione così elevate non si raggiungono con i sistemi di retopology automatica. Al fine di mantenere un'approssimazione affidabile ovunque, è necessario avere un numero di facce piuttosto elevato (come mostrato in figura 6.1). Ciò è dovuto dal limite dei sistemi di remeshing automatico, che risultano essere poco adatti in questo contesto.

In conclusione, si può affermare che i pochi poligoni creati con le tecniche manuali e a patch sono distribuiti meglio sulla superficie, e ciò garantisce una maggiore corrispondenza con le forme del modello iniziale pur mantenendo verosimiglianza con le forme iniziali. La seguente tabella mostra numericamente i risultati ottenuti.

Approccio	TrisCount	Riduzione
Manuale	12.115	96,10 %
Patch	12.447	96,06 %
Automatico	49.291	84,31 %

Tabella 6.1: I valori riportati in tabella corrispondono ad una media eseguita su tutti i modelli retopologizzati. Come è possibile constatare, gli approcci manuale e a patch riescono a raggiungere gli stessi risultati a livello di peso poligonale. Come si vedrà al paragrafo 6.3, anche sul piano topologico i modelli risultano del tutto equivalenti.

Modelli scansionati	Modelli retopologizzati
54.55 MB	894 kB

Tabella 6.2: I valori riportati sono la media eseguita considerando tutti i modelli. Per misurare questi valori, il formato scelto è .OBJ, uno dei più semplici nell'ambito della computer grafica.

6.3 Qualità della topologia

Le considerazioni qualitative fanno riferimento alle caratteristiche topologiche dei modelli retopologizzati. Queste proprietà dipendono fortemente dal metodo di remeshing adottato. Anche in questo ambito, gli approcci manuale e a patch si rivelano essere i più flessibili. La libertà concessa al modellatore consente di avere pieno controllo sul posizionamento di singolarità, la creazione di edge-loops funzionali e la resa di un edge-flow ordinato. Inoltre, la possibilità di variare la densità di quadrangoli consente di rappresentare con maggiore fedeltà le zone organiche e quelle maggiormente colpite dai danni. Tutti questi aspetti procurano notevoli vantaggi durante la fase di texture mapping, in cui si possono apprezzare risultati nettamente migliori. In figura 6.2 è mostrato come i due approcci producano risultati topologicamente affini.

Viceversa, i metodi di remeshing automatico, soffrono da questo punto di vista. L'edge-flow restituito risulta accettabile solamente nelle zone geometriche delle statue, mentre in quelle organiche è necessario intervenire manualmente. Benché ciò richieda solamente di disegnare delle curve sul modello, gli altri aspetti topologici quali singolarità ed edge-loops non vengono in alcun modo presi in considerazione. *Instant Meshes* produce mesh con una forte caratteristica isotropica, la quale condiziona il peso poligonale del modello finale. Proprio per questa caratteristica, non è possibile variare la densità laddove servirebbero quadrangoli più piccoli o più grandi. A causa di tali motivi, questi modelli sono difficili da impiegare nella fase di texture mapping, al punto da ricorrere anche per questa operazione a metodi automatici. La figura 6.3 mostra i risultati ottenibili con i metodi automatici.

6.4 Valutazione dei tempi di produzione

Uno degli aspetti da tenere in considerazione è la quantità di tempo richiesta per retopologizzare un singolo modello. Per eccellenza, la retopology è un'operazione particolarmente time-consuming. È plausibile dunque come la ricerca in questo ambito si sia indirizzata verso soluzioni in grado di abbassare i tempi di produzione, automatizzando quanto più possibile le ripetitive azioni richieste dai metodi manuali. Tuttavia, per valutare le tempistiche non si deve tenere in considerazione solamente il tempo relativo alla creazione vera e propria della mesh, quanto anche il tempo richiesto da eventuali correzioni da apportare ai modelli.

I metodi manuali risultano essere i più esigenti in termini di tempistiche. Questo è dovuto al fatto di poter agire solamente su vertici e lati. Invece, l'approccio a patch risulta essere molto meno pretenzioso, arrivando a dimezzare le tempistiche. Ciononostante, nella maggior parte dei casi è richiesto un secondario intervento manuale da parte del modellatore per correggere gli errori topologici. Tuttavia, ciò non incide in modo rilevante sulle tempistiche, rimanendo comunque più basse di quelle manuali.

Per ridurre i tempi, gli approcci automatici risultano i più indicati, poiché producono mesh poligonali in tempi veramente ridotti. *Instant Meshes* è molto rapido, anche in presenza di molte curve. La rapidità di *Autopo*, invece, è piuttosto dipendente dal numero di curve disegnate, oltre che dalla complessità del modello. Tuttavia, è da tenere in considerazione che non sempre restituiscono il risultato sperato, e per rimediare a tale incombenza, è necessario agire manualmente.

In conclusione, è possibile notare come i due estremi siano rappresentati dai metodi manuali e da quelli automatici. Al centro, si colloca l'approccio a patch, che si mostra un ibrido tra le due modalità precedentemente citate. Tuttavia, non è possibile affermare che l'automatizzazione porti dei vantaggi tangibili riguardo la qualità finale del modello. Pertanto, nello scenario analizzato risultano validi i metodi manuali ed a patch.

6.5 Output finali e test

I modelli retopologizzati sono stati rilasciati in due formati, quali .OBJ e .HTML. La scelta del formato .OBJ è stata motivata dalle ragioni esposte al paragrafo 5.3.1, mentre il formato .HTML è stato scelto per via della sua versatilità nell'ambito delle applicazioni web. Si è scelto di visualizzare i modelli tramite un browser web. Pertanto, è stato scelto il plugin open source *Blend4Web*. Questo tool è disponibile solamente per *Blender*[™], ed è destinato allo sviluppo di applicazioni web con contenuti tridimensionali. Gli output creati consistono in file .HTML letti e visualizzati tramite la libreria grafica *WebGL*, inclusa nei moderni browser web.

Il plugin consente di lavorare unicamente all'interno di *Blender*[™], esportando direttamente un file .HTML contenente informazioni relative alla geometria, alle textures e all'illuminazione. È inoltre fornito un set base di controlli per manipolare l'oggetto. Dopo aver creato un semplice sistema di illuminazione attorno al modello, è possibile vedere subito i risultati ottenuti in un qualunque browser.

I pacchetti .HTML sono stati depositati all'interno di un server, in modo da poterne visualizzare i contenuti da remoto tramite un browser web. I test sono stati eseguiti sia su dispositivi PC, sia su dispositivi mobile quali smartphone e tablet. Si è constatato come anche i dispositivi mobile dotati di hardware di basso livello siano in grado di manipolare l'oggetto, consentendo di ruotarlo, traslarlo e scalarlo con fluidità. Questo è indice di come la geometria sia stata efficientemente semplificata. La figura 6.4 mostra la visualizzazione di un modello su browser *Google Chrome* sia da PC che da smartphone. In entrambi i casi, i tempi di caricamento sono nell'ordine del secondo.

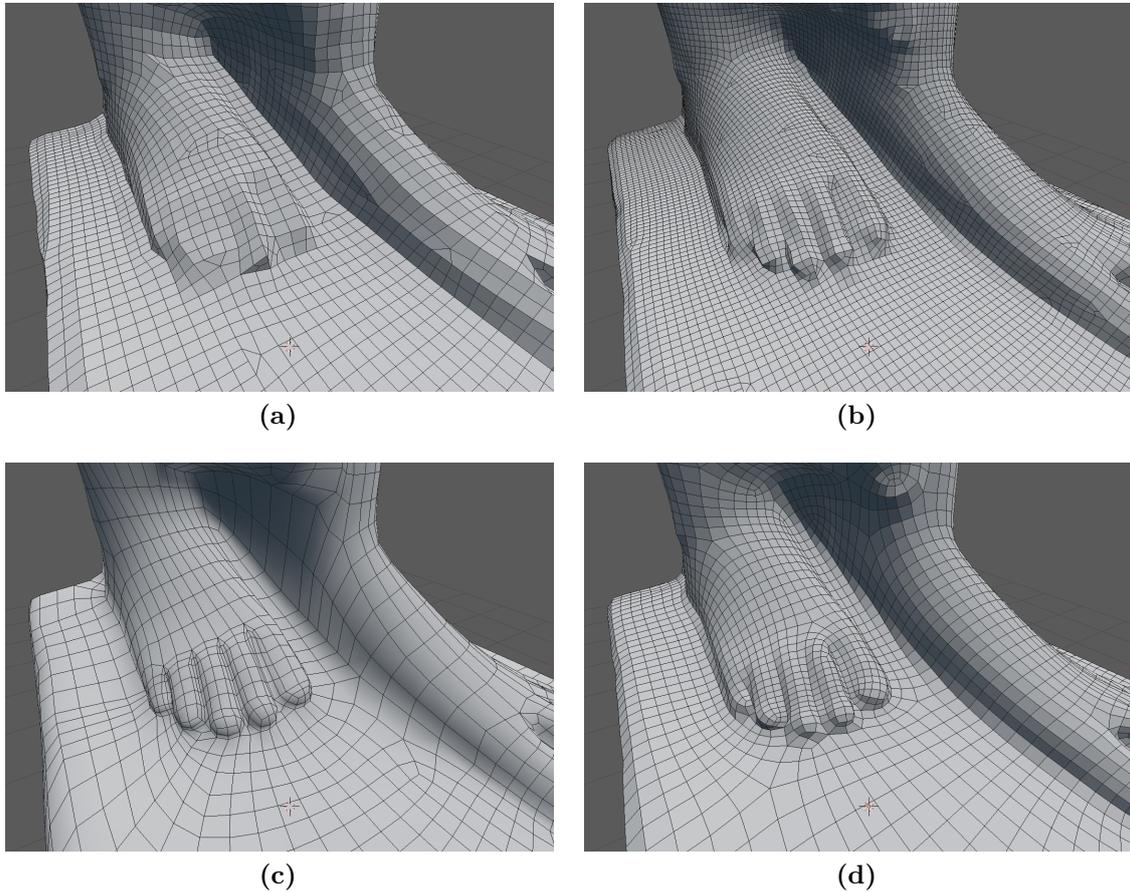


Figura 6.1: La quadrangolazione isotropica di *Instant Meshes* costringe ad impostare un numero elevato di poligoni finali, al fine di poter rappresentare con maggiore precisione le zone più complesse (a) e (b). Sebbene sia fondamentale assicurare corrispondenza col modello scansionato, il peso poligonale ne risente negativamente. Pertanto, in questo contesto i metodi automatici non consentono di raggiungere un trade-off accettabile. I metodi manuali e patch-based (c) consentono di avere pieno controllo sia sulla topologia, sia sul numero di poligoni. La topologia è fondamentale per riprodurre con maggiore fedeltà le zone più difficili come quelle dei piedi, e gioca un ruolo importante per la fase di texture mapping. Difficilmente i metodi automatici producono risultati analoghi, tuttavia algoritmi come *Autopo* (d) considerano maggiormente l'aspetto topologico del modello, seppur in modo limitato.

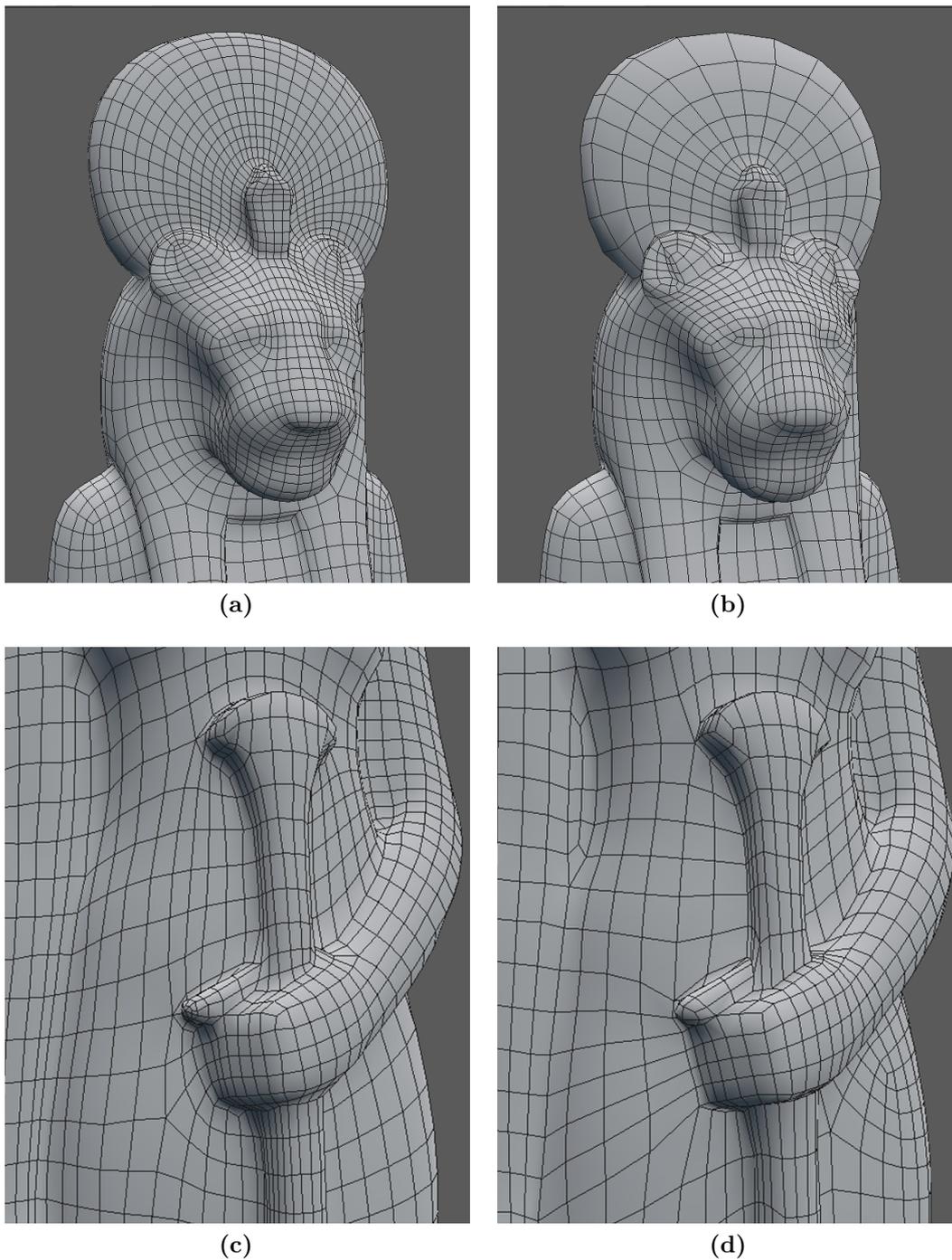


Figura 6.2: Il modello a sinistra è stato creato ricorrendo alle tecniche manuali discusse in 4.2.2, mentre quello a destra utilizzando *SketchRetopo*. Entrambi presentano un edge-flow ordinato ed una buona distribuzione dei poligoni, sia nelle parti organiche che in quelle più geometriche. Ciò rende *SketchRetopo* un tool in grado di raggiungere risultati del tutto paragonabili a quelli ottenuti via retopology manuale classica.

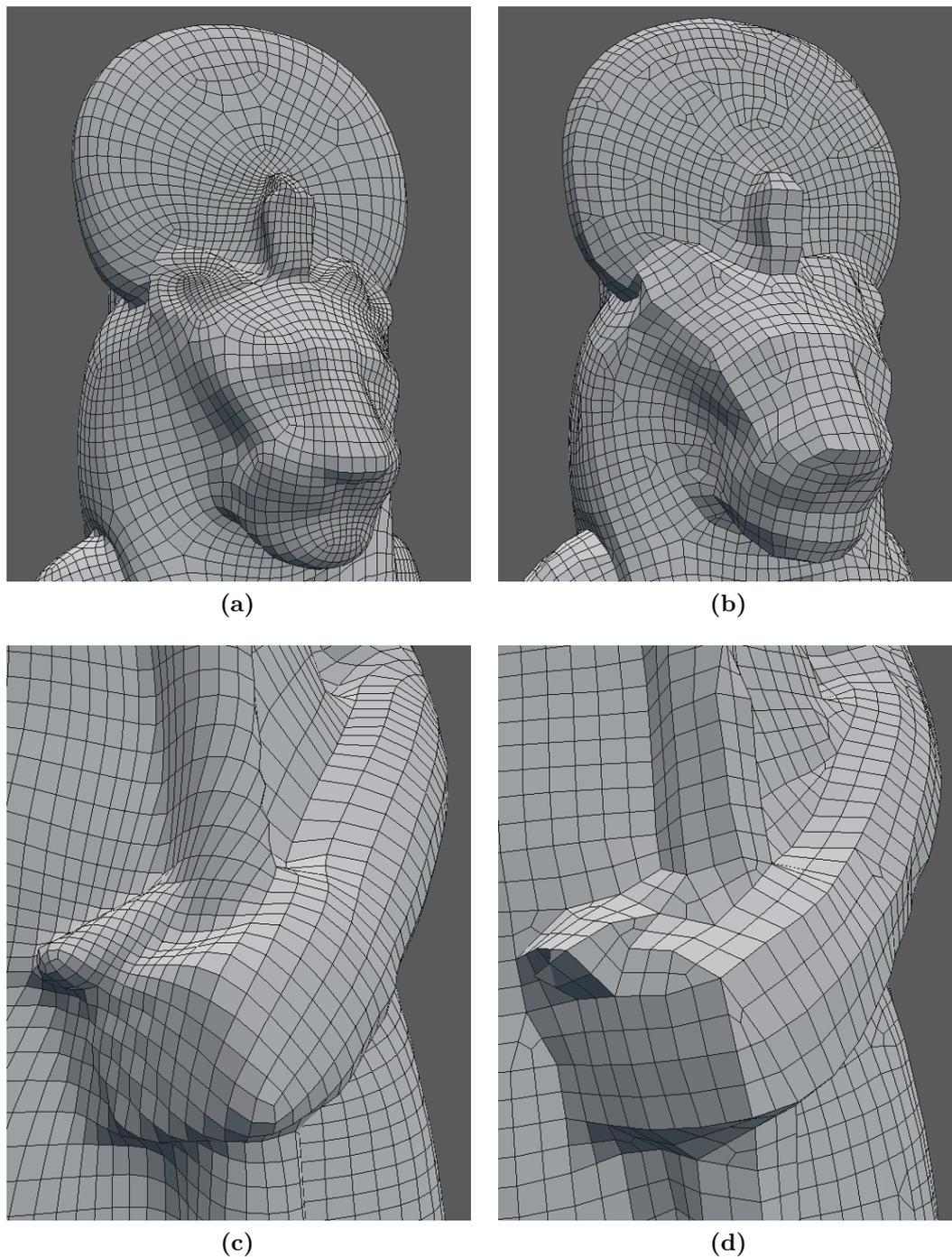
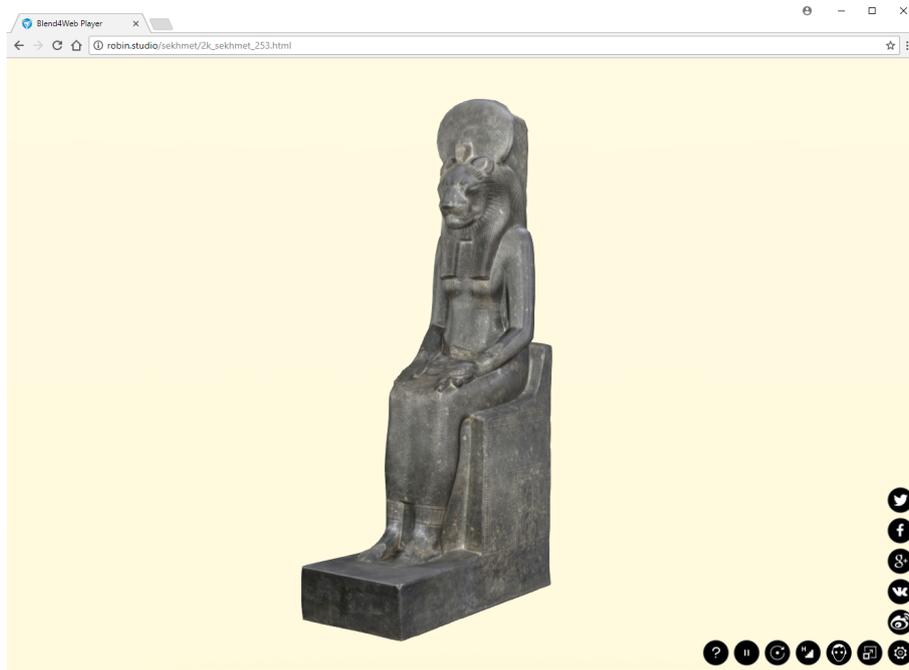


Figura 6.3: Gli algoritmi di remeshing automatico non sono in grado di produrre modelli con un flusso di lati pulito, nonostante il tracciamento di molte curve per l'orientamento dei quadrangoli. La totale assenza di edge-loops e la presenza di molte singolarità rendono difficile la fase di texture mapping. Ciò vale principalmente per le zone organiche delle statue, dove i quadrangoli risultano pesantemente distorti, mentre per quelle geometriche i risultati sono più accettabili.



(a)



(b)

Figura 6.4: Visualizzazione tramite browser su PC (a) e su smartphone (b).

Capitolo 7

Conclusioni e sviluppi futuri

La retopology è una fase della modellazione ormai diventata standard in molte pipeline di produzione. È imprescindibile nei casi in cui i modelli iniziali non siano adatti alle finalità del progetto che si intende sviluppare. In particolare, nei contesti che prevedono il fattore di interattività oppure animazione, i modelli coinvolti devono rispettare determinati requisiti. Nonostante l'aumento delle prestazioni dei calcolatori, al giorno d'oggi non si è ancora in grado di processare un elevato numero di geometrie in modo da garantire fluidità di fruizione. È necessario che i modelli siano poligonalmente leggeri. In virtù delle caratteristiche dei modelli poligonali ottenuti tramite scansioni 3D, nel settore del Cultral Heritage la retopology risulta essere una fase di ottimizzazione vera e propria. Non si abbassa solamente il numero di elementi che compongono la mesh, quanto si ottimizza l'intero modello per le successive lavorazioni, quali texture mapping oppure animazione. In quest'ottica, la topologia del modello è fondamentale, ed il termine retopology fa riferimento anche a questo aspetto.

In questa tesi sono stati esposti diversi metodi per retopologizzare modelli tridimensionali. A causa del gravoso lavoro richiesto dagli approcci manuali, negli anni sono stati proposti metodi in grado di automatizzare l'intero processo. Nonostante questi algoritmi siano stati perfezionati, ancora oggi sono lontani dal garantire i risultati conseguibili con le tecniche manuali. Soprattutto quando si lavora con un modello particolarmente complesso, le tecniche manuali risultano notevolmente più adatte, al punto da essere maggiormente apprezzate dagli artisti 3D.

SketchRetopo è un esempio di come questa antinomia possa essere superata, e dimostra anche come sia necessario adottare un approccio del tutto diverso per creare mesh poligonali. Non si tratta solamente di eseguire uno schema applicativo diverso, quanto di capire le fondamenta di un nuovo punto di vista.

I risultati ottenuti lavorando sulle statue Sekhmet confermano tale situazione. Si riscontrano evidenti analogie con i metodi di retopology manuale sia nella topologia che nel peso poligonale dei modelli. L'elevata percentuale di riduzione del numero di triangoli ha consentito di abbassare notevolmente il peso finale dei modelli,

rendendoli idonei sia ad una visualizzazione online, sia ad essere integrati in un ambiente interattivo. Anche i dispositivi con hardware di basso livello sono in grado di supportare tale visualizzazione. Invece, la topologia ordinata e funzionale ha consentito di sfruttare al meglio le moderne procedure image-based di texture mapping, rendendo i modelli quanto più verosimili a quelli scansionati.

Il risultato può essere riassunto in un principio tipico dell'ingegneria dell'informazione: rappresentare la stessa quantità di informazioni diminuendo la quantità di dati da elaborare. In questo campo, l'informazione è corredata di una componente grafica non indifferente. Pertanto, importanti attenzioni vanno rivolte anche verso tale aspetto.

Gli sviluppi futuri consistono nell'applicazione delle tecniche e delle metodologie illustrate nella presente tesi in altre iniziative di Cultural Heritage, al fine di promuovere futuri progetti basati sull'utilizzo delle nuove tecnologie digitali.

Bibliografia

- [1] Bommes, David and Lévy, Bruno and Pietroni, Nico and Puppo, Enrico and Silva, Claudio and Tarini, Marco and Zorin, Denis. “Quad-Mesh Generation and Processing: A Survey”. In: *Computer Graphics Forum*, 32.6, (Settembre 2013), pp. 51–76.
- [2] Catmull, E. and Clark, J. “Recursively Generated B-spline Surfaces on Arbitrary Topological Meshes”. In: *Computer-Aided Design*, 10.6, (Novembre 1978), pp. 350–355.
- [3] Doo, D. and Sabin, M. “Behaviors of recursive division surfaces near extraordinary points”. In: *Computer-Aided Design*, 10.6, (Novembre 1978), pp. 356–360.
- [4] Guidi, Gabriele, Russo, Michele, Beraldin, Jean-Angelo (2013), *Acquisizione 3D e modellazione poligonale*, McGraw-Hill, Milano.
- [5] James Bush, Mark W.J. Ferguson, Tracey Mason, and Gus McGrouther. “The dynamic rotation of Langer’s lines on facial expression”. In: *Journal of Plastic, Reconstructive & Aesthetic Surgery*, 60.4 (Aprile 2007), pp. 393–399.
- [6] Kenshi Takayama and Daniele Panozzo and Olga Sorkine-Hornung. “Pattern-Based Quadrangulation for N -Sided Patches Patches”. In: *Computer Graphics Forum (proceedings of EUROGRAPHICS Symposium on Geometry Processing)*, 33.5, (Agosto 2014), pp. 177–184.
- [7] Kenshi Takayama and Daniele Panozzo and Alexander Sorkine-Hornung and Olga Sorkine-Hornung. “Robust and controllable quadrangulation of triangular and rectangular regions”. *Tech Report*, ETH Zurich.
- [8] Kenshi Takayama and Daniele Panozzo and Alexander Sorkine-Hornung and Olga Sorkine-Hornung. “Sketch-Based Generation and Editing of Quad Meshes”. In: *ACM Transactions on Graphics (proceedings of ACM SIGGRAPH)*, 32.4,

- (Luglio 2013), 97:1–97:8.
- [9] Kovacs, Denis and Myles, Ashish and Zorin, Denis. “Anisotropic Quadrangulation”. In: *Computer Aided Geometric Design*, 28.8, (Novembre 2011), pp. 449–462.
- [10] Loop, Charles. “Smooth Subdivision Surfaces Based on Triangles”. Department of Mathematics, The University of Utah, Masters Thesis, 1987.
- [11] P. Cignoni and C. Montani and R. Scopigno. “A comparison of mesh simplification algorithms”. In: *Computers & Graphics*, 22.1, (Febbraio 1998), pp. 37–54.
- [12] Poranne, Roi and Tarini, Marco and Huber, Sandro and Panozzo, Daniele and Sorkine-Hornung, Olga. “Autocuts: Simultaneous Distortion and Cut Optimization for UV Mapping”. In: *ACM Trans. on Graphics - Siggraph Asia 2017*, 36.6, (Novembre 2017).
- [13] Simonds, Ben (2010), *Blender Master Class: a hands-on guide to modeling, sculpting, materials, and rendering*, No Starch Press, San Francisco.
- [14] Surynková, Petra. “Constructions of Quadrilateral Meshes: a Comparative Study”. In: *South Bohemia Mathematical Letters*, 24.1, 2016, pp. 43-48.
- [15] Tarini, Marco. “Volume-encoded UV-maps”. In: *ACM Trans. Graph.*, 35.4, (Luglio 2016), 107:1–107:13.
- [16] Verdiani, Giorgio (a cura di), Di Tondo, Sergio, Fantini, Filippo, Picci, Mirco (2011), *Il ritorno all'immagine, nuove procedure image based per il Cultural Heritage*, Lulu.
- [17] Wenzel Jakob and Marco Tarini and Daniele Panozzo and Olga Sorkine-Hornung. “Instant Field-Aligned Meshes”. In: *ACM Transactions on Graphics (Proceedings of SIGGRAPH ASIA)*, 34.6, (Novembre 2015).
- [18] Zorin, D. and Schröder, P.: “Subdivision for Modeling and Animation”. ACM SIGGRAPH Course notes, 2000.
- [19] African Fossils.org, <http://africanfossils.org/>
- [20] Azimut s.a.s., <http://studioazimut.com/>

- [21] Europeana Collections, <https://www.europeana.eu/portal/it>
- [22] giantformlabs. *Max- Male Anatomy*, <https://www.turbosquid.com/3d-models/realistic-male-3d-model/530373>
- [23] Google Arts & Culture, <https://www.google.com/culturalinstitute/beta/>
- [24] Museo Egizio di Torino, <http://www.museoegizio.it/>
- [25] Museu d'Arqueologia de Catalunya, <http://www.mac.cat/>
- [26] Outlander Anatomy. *Anatomy Lesson 11: Muscles of Facial Expression* <http://www.outlanderanatomy.com/jamies-face-or-ye-do-it-face-to-face/>
- [27] Sketchfab, <https://sketchfab.com/>
- [28] Sketchfab, *Museu d'Arqueologia de Catalunya*, <https://sketchfab.com/macb3d>
- [29] Sketchfab, *The British Museum*, <https://sketchfab.com/britishmuseum>
- [30] The British Museum, <http://www.britishmuseum.org/>
- [31] Adobe *Acrobat Reader DC*[®]. Software. <https://acrobat.adobe.com/it/it/acrobat/pdf-reader.html>
- [32] Adobe *Photoshop CC*[®]. Software. <http://www.adobe.com/it/products/photoshop.html>
- [33] Allegorithmic *Substance Designer*. Software. <https://www.allegorithmic.com/products/substance-designer>
- [34] Allegorithmic *Substance Painter*. Software. <https://www.allegorithmic.com/products/substance-painter>
- [35] Autodesk *3ds Max*[®]. Software. <https://www.autodesk.com/products/3ds-max/overview>
- [36] Autodesk *Alias*[®]. Software. <https://www.autodesk.it/products/alias-products/overview>

- [37] Autodesk *Maya*[®]. Software. <https://www.autodesk.com/products/maya/overview>
- [38] Autodesk *Mudbox*[®]. Software. <https://www.autodesk.com/products/mudbox/overview>
- [39] *Blender*[™]. Software. <https://www.blender.org/>
- [40] Headus *UVLayout*[®]. Software. <http://www.uvlayout.com/>
- [41] *Instant Meshes*. Software. <http://igl.ethz.ch/projects/instant-meshes/>
- [42] Pilgway *3D-Coat*[®]. Software. <https://3dcoat.com/home/>
- [43] Quixel *Suite 2*[®]. Software. <https://quixel.se/suite2/>
- [44] *SketchRetopo*. Software. <http://igl.ethz.ch/projects/sketch-retopo/>
- [45] *SpeedRetopo*. Add-on per Blender. <https://gumroad.com/l/speedretopo>
- [46] Pixologic *ZBrush*[®]. Software. <http://pixologic.com/features/>
- [47] *Topogun*[®]. Software. <http://www.topogun.com/>
- [48] *xNormal*. Software. <http://www.xnormal.net/>
- [49] Wikipedia - L'enciclopedia libera e collaborativa, *Coordinate baricentriche*, https://it.wikipedia.org/wiki/Coordinate_baricentriche