

Come vengono generate le mesh

- ✓ Catturate dalla realtà
 - ⇒ 3D Acquisition
 - ⇒ per es, laser scanning, fotogrammetria...
 - ⇒ (su questo, torneremo)
- ✓ Generazione procedurale
 - ⇒ Per es, simulazione
 - ⇒ Per es, CAD model generati da specifiche
- ✓ Modellazione manuale
 - ⇒ Da parte di artisti digitali
 - ⇒ Appositi strumenti, come 3D Studio Max, Maya, Blender, MudBox, Z-Brush... o molti altri

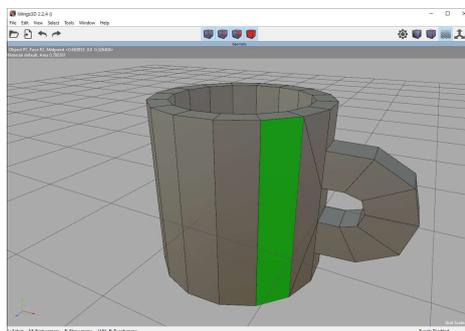


148

Generazione manuale di mesh (cenni)

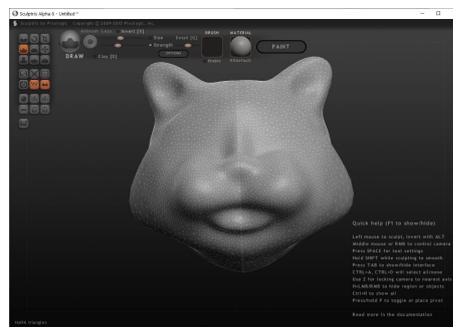
- ✓ Alcuni paradigmi di modellazione manuale:

Low-poly modelling



Esempio con Wings3D

Digital Sculpting



Esempio con Sculpttris Alpha



149

Generazione manuale di mesh (cenni)

✓ Low poly Modelling

- ⇒ Artista / Autore manipola connettività direttamente
- ⇒ (editando elementi come facce, edge, vertici)
- ⇒ Artista decide quali elementi usare, per es, quanti lati ogni faccia poligonale
- ⇒ Le operazioni mantengono automaticamente two-manifold-ness (e a volte chiusura) delle mesh
- ⇒ Tipicamente sono generate mesh poligonali a bassa risoluzione



150

Generazione manuale di mesh (cenni)

✓ Digital Sculpting

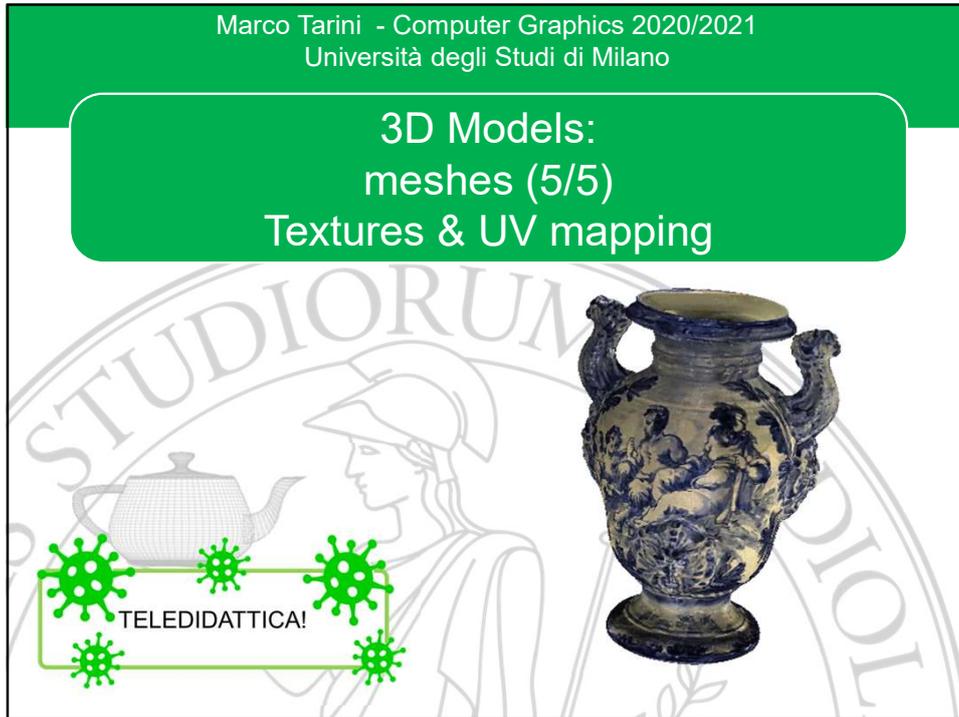
- ⇒ Artista scolpisce la forma con pennellate («brush strokes») che imitano la manipolazione della creta / pongo
- ⇒ (ma con rimozione o aggiunta di materia possibile)
- ⇒ «Pennelli» con effetti diversi, per es, local smoothing, appiattare, estrarre, etc
- ⇒ Sistema gestisce il meshing automaticamente in background (compreso, generazione di risoluzione adattiva), per es effettuando edge-collapse o edge split (il contrario) per avere una risoluzione adeguata
- ⇒ Tipicamente sono generate tri-mesh ad alta risoluz.



151

Marco Tarini - Computer Graphics 2020/2021
Università degli Studi di Milano

3D Models:
meshes (5/5)
Textures & UV mapping



TELEDIDATTICA!

152

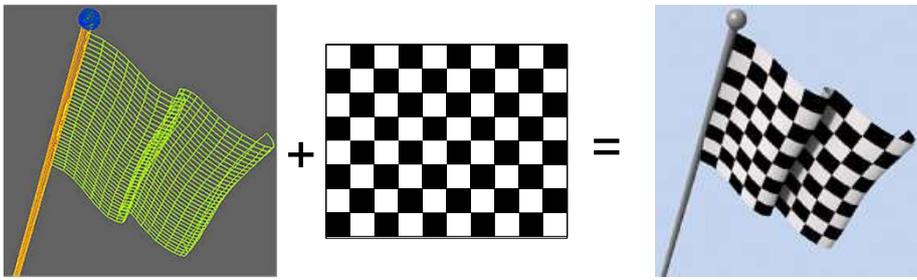
Textures

- ✓ Spesso vorremmo poter campionare un *attributo* (un segnale: come colore, normale, lucentezza, temperatura etc) sulla superficie in modo *molto più denso* di quanto campioniamo la stessa superficie per catturare la sua forma geometrica.
- ✓ Per es:
 - ⇒ forma: ci bastano 10 campioni per cm² (sufficiente per rappresentare una statuetta in dettaglio)
 - ⇒ colore: vorremmo 100 campioni per cm² (necessario per riprodurre il dettaglio pittorico sulla statuetta)
- ✓ Memorizzando gli **attributi per vertice** su di una mesh poligonale, ho:
1 campione di segnale = 1 campione di forma = 1 vertice ☹
 - ⇒ e invece attributi per faccia, guadagno solo un fattore 2 in media (su una mesh di triangoli - 1 su mesh di quad)
- ✓ In 2D, le immagini “raster” (griglie 2D regolari di pixel) sono un ottimo modo di campionare segnali in modo molto denso
 - ⇒ es: un’immagine a 100 DPI = 40 dpmc = 1600 campioni («pixel») per cm²
 - ⇒ es: immagine risoluzione 1000x1000 = 1 Milone di pixel
- ✓ Come ottenere un risultato simile per le superfici 3D?



154

Mesh + Texture



The diagram illustrates the process of texturing a mesh. On the left, a green wireframe mesh of a flag is shown. In the middle, a black and white checkerboard texture is displayed. An equals sign follows, leading to the final rendered flag with the checkerboard texture applied.

Mesh
(pure quad mesh)

Texture
(griglia regolare di "texel" 2D)



155

Mesh + Texture



The diagram illustrates the process of texturing a mesh. On the left, a white tri-mesh of a vase is shown. In the middle, a texture composed of a patchwork of photos of vases is displayed. An equals sign follows, leading to the final rendered vase with the texture applied.

Tri-Mesh

Texture
(qui: un patchwork di foto)



156

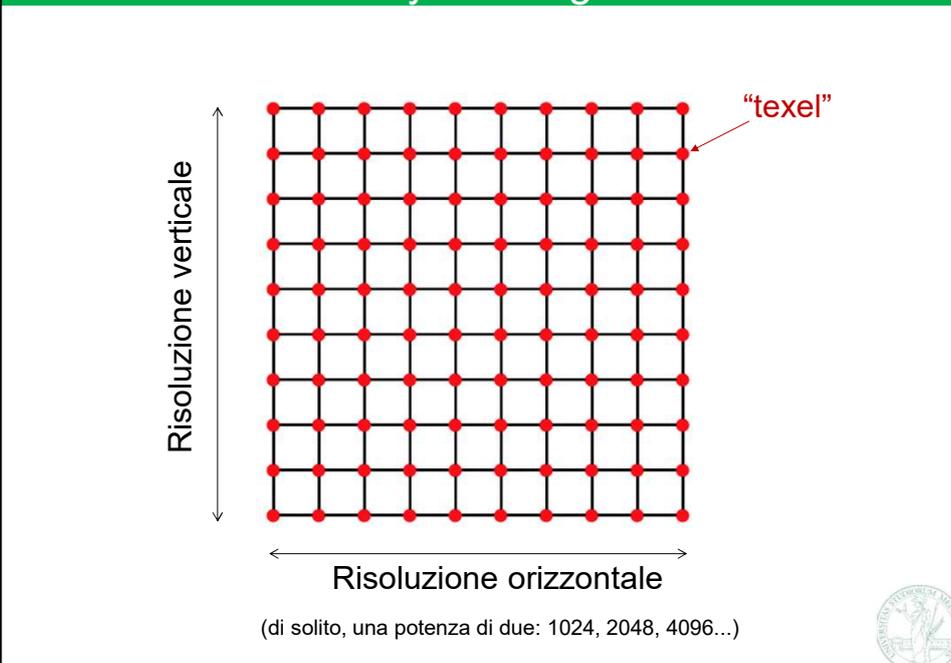
Texture Mapping

- ✓ Idea intuitiva: incolliamo un'immagine raster (detta **texture**, o **tessitura**) alla superficie di una mesh
 - ⇒ immagine (raster) = grigliato regolare di **pixel**
 - ⇒ texture = grigliato regolare di **texel** (texel = **texture element**)
- ✓ Ogni punto sulla superficie 3D **S** ← rappresentata dalla mesh viene associato ad un punto sulla rettangolo 2D **T**
 - ⇒ convenzione: le coordinate 2D su **T** ← anche dette coordinate texture o parametriche sono indicate da (u, v) – a volte da (s, t) per distinguerle dalle coordinate 3D su **S** che sono (x, y, z)
 - ⇒ questo mapping da **S** a **T** è detto **UV-map** o **UV-mapping** della mesh, ← in contesto industriale oppure la sua **parametrizzazione** ← in contesto accademico
- ✓ Domande a cui dobbiamo rispondere:
 - ⇒ come è strutturata una texture
 - ⇒ come memorizzare lo UV-map sulla mesh?
 - ⇒ come ottenerlo per una data superficie? («how do we **parametrize**» or «how do we **uv-map**» a mesh)



157

The Texture: basically an image



158

Tessiture e texel

- ✓ Ogni pixel della tessitura, chiamato texel, memorizza valori qualsiasi che variano sulla superficie
 - ⇒ Tipicamente: colori, normali, valori di trasparenza, coefficienti di lucidità – o altri modi di descrivere un «materiale» ...
- ✓ La tessitura è sostanzialmente un'immagine raster
 - ⇒ Un array 2D di «texel»
- ✓ Come tale, è caratterizzata da:
 - ⇒ Risoluzione orizzontale R_x (di solito, potenza di due $\leq 2^{13} = 8192$)
 - ⇒ Risoluzione verticale R_y (idem)
 - ⇒ Numero di canali per texel Ch (tipicamente, 1,2 o 4)
 - ⇒ Numero di bit per canale B (8 (spesso), 16, 24 ...)
 - ⇒ Spazio totale occupato in memoria (espressa in *bit*) = $R_x \cdot R_y \cdot Ch \cdot B$
numero di texel «bit-depth»
- ✓ In quanto immagine, può essere, per es, «dipinta» da un artista, catturata da foto, photoshopped, etc
 - ⇒ Esistono anche molte tecniche per sintetizzarle automaticamente
 - ⇒ Vedremo uno dei modi più comuni



159

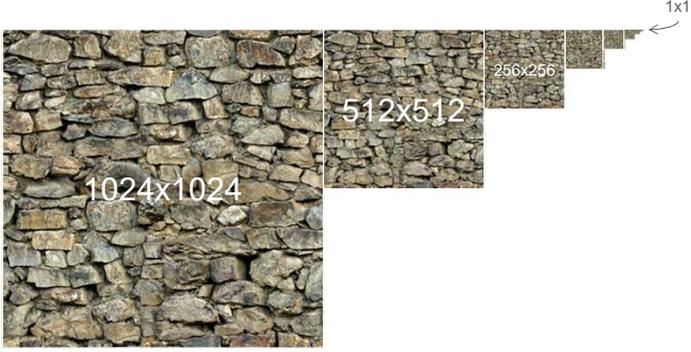
Tessiture e MIPmap level

- ✓ In quanto immagine raster, una tessitura si presta a semplici strutture multirisoluzione:
 - ⇒ Basta ridurre la risoluzione dell'immagine per ottenere un «livello» diverso di una piramide di livello di dettaglio
 - ⇒ Come si riduce la risoluzioni dell'immagine? Vedere corso di image processing (hint: interpolando fra i pixel / i texel)
- ✓ Struttura chiamata «MIP-map»:
Piramide di livelli di dettaglio per tessiture:
 - ⇒ Livello 0: la tessitura originale, risoluzione massima
 - ⇒ Livello 1: una tessitura ridotta con metà della risoluzione X e Y
 - ⇒ Livello 2: riduzione del livello 1
 - ⇒ ...
 - ⇒ Livello N: tessitura 1x1 (per tessiture quadrate)
 - ⇒ Nota: in tutto sono $\log_2(\text{RES})$ livelli
 - ⇒ Nota: in tutto, lo spazio totale occupato da tutti i livelli oltre lo 0 è solo $1/4 + 1/16 + 1/64 + 1/256 + \dots < 1/3$ quello del livello zero



160

Texture: multirisoluzione



Piramide di livelli di dettaglio
etta MIP-map



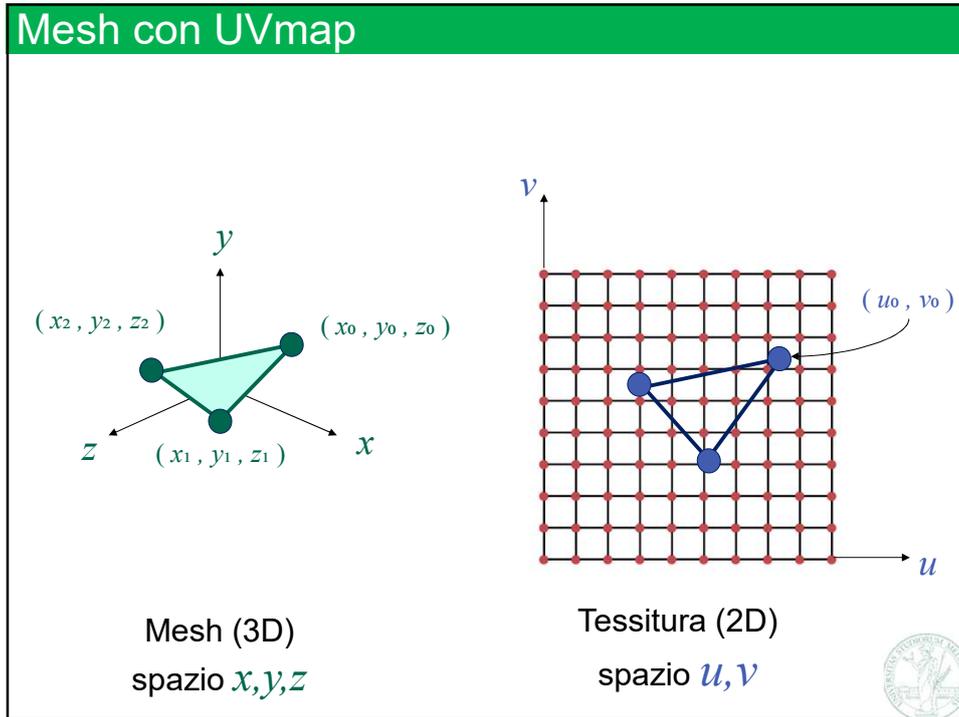
161

Tessiture e supporto Hardware

- ✓ Le tessiture sono estremamente utilizzate nella maggior parte dei contesti applicativi in cui si usino mesh poligonali
 - ⇒ Come: games, VR, movies, commercio elettronico, applicazioni architettonali, sci vis, beni culturali...
 - (ma spesso non: CAD, applicaz medicali)
- ✓ Parte della loro diffusione è dovuta al supporto HW:
 - ⇒ le GPU (il processore delle schede video) sono progettate per renderizzare mesh triangolari con tessitura associata
 - ⇒ Livello di MIPmap compresi
 - ⇒ I limiti e le caratteristiche tipiche di una tessitura (num di canali ≤ 4 , risoluzione massima, bit depth, risoluzione come potenza di 2) sono imposti da questo



162



163

Mesh con UVmap

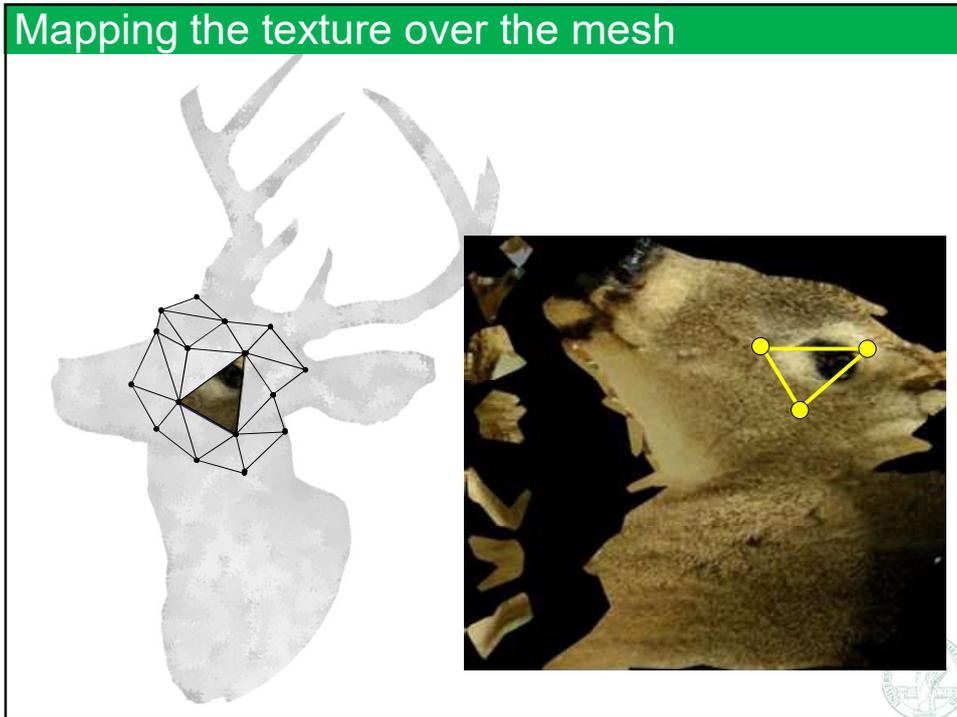
- ✓ Le posizioni UV sono memorizzate come attributo per vertice
 - ⇒ Come ogni attributo, si considera interpolato dentro alle facce della mesh (con le coordinate baricentriche)

	Geometria	Attributi
$V_0 \rightarrow$	x_0, y_0, z_0	u_0, v_0
$V_1 \rightarrow$	x_1, y_1, z_1	u_1, v_1
$V_2 \rightarrow$	x_2, y_2, z_2	u_2, v_2
$V_3 \rightarrow$	x_3, y_3, z_3	u_3, v_3
$V_4 \rightarrow$	x_4, y_4, z_4	u_4, v_4
$V_5 \rightarrow$	x_5, y_5, z_5	u_5, v_5

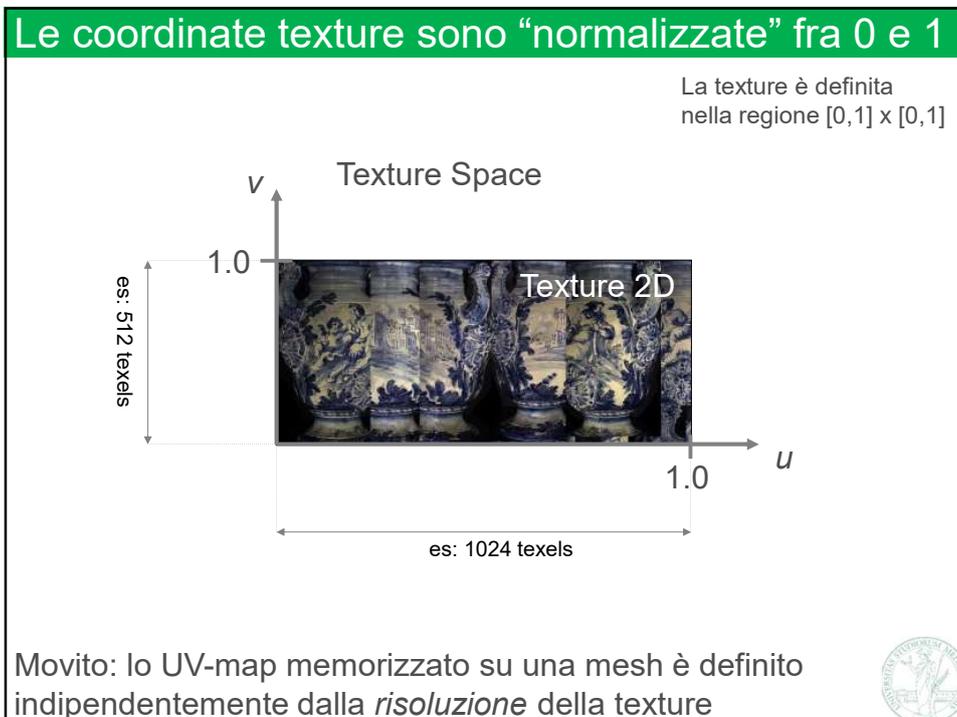
the
uv-map!

- ✓ Conseguenze:
 - ⇒ Ogni triangolo della mesh corrisponde a due triangoli: il triangolo T_3 nello spazio in 3D « x,y,z », il triangolo T_2 nello spazio 2D « u,v », dentro al rettangolo della texture
 - ⇒ Ogni punto dentro al triangolo in T_3 è mappato nel punto con le stesse coordinate baricentriche nel triangolo T_2
 - ⇒ L'intera mesh è mappata sul rettangolo di tessitura

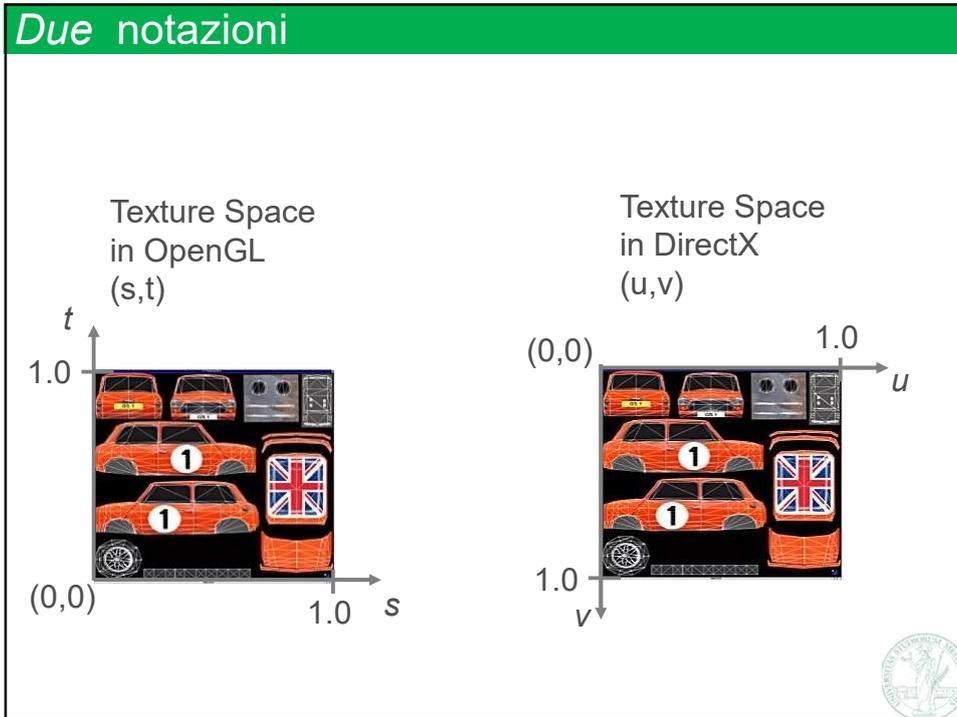
164



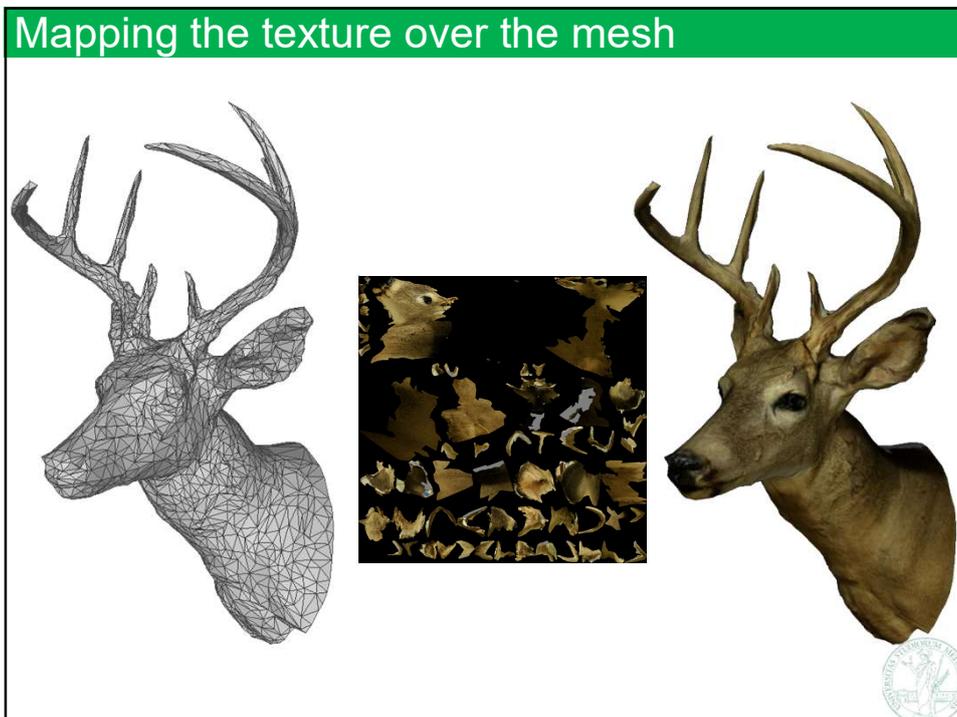
165



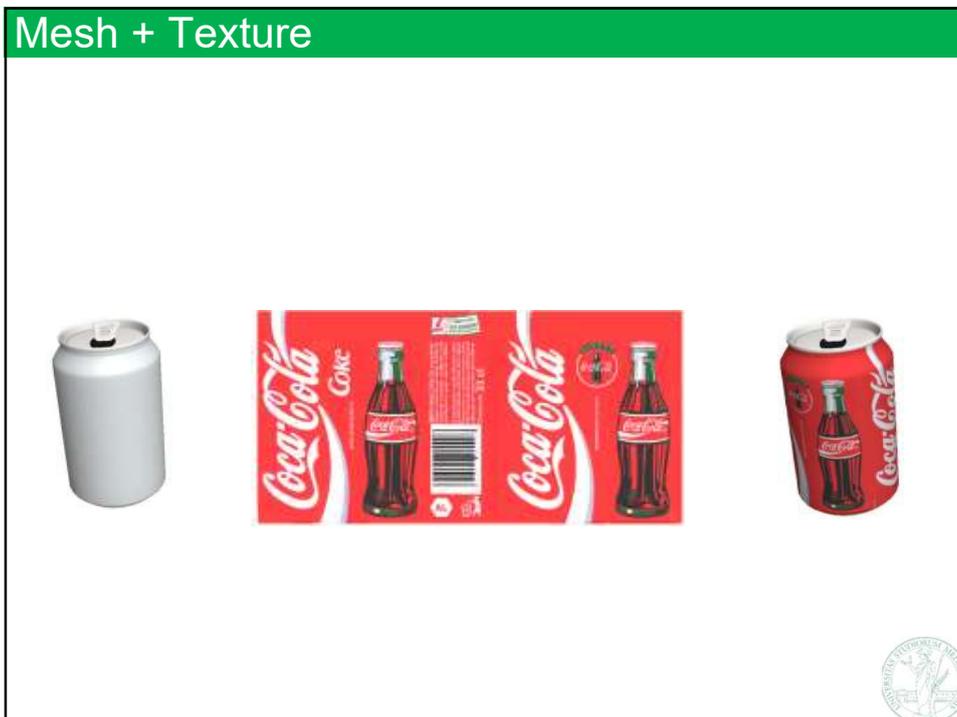
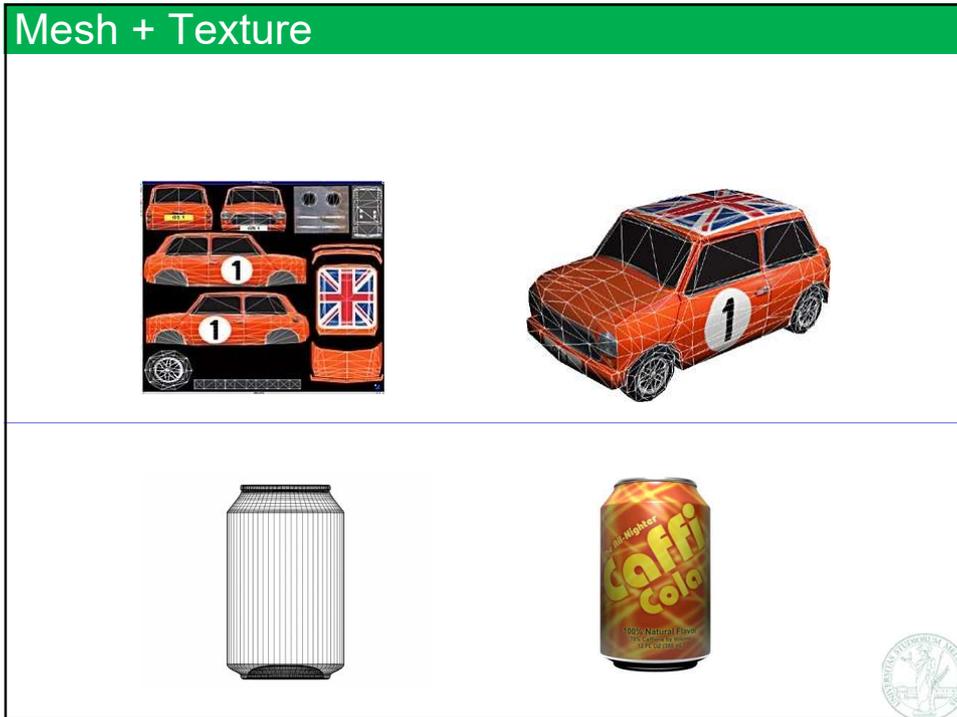
166



167



168



Costruire un UV-map di una mesh

- ✓ **UV-mapping di una mesh, o mesh parametrization**
assegnamento di coordinate UV per ogni suo vertice
 - ⇒ A volte, (raro) la mesh viene creata congiuntamente al suo UV map
 - ⇒ Più spesso, la mesh viene creata (es. acquisita, o modellata) ed è sprovvista di un UV-map per una mesh che ne sia
- ✓ **Task non semplice**
 - ⇒ Automatizzazione: tema molto studiato nel Geometry Processing
 - ⇒ Artisti digitali intervengono manualmente con software di 3D modelling (es Blender, Maya)
- ✓ **Un «buon» UV map deve rispondere a molti criteri:**
 - ⇒ Inietività: ogni punto della tessitura può essere mappato in al più un punto della mesh (no triangoli sovrapposti in spazio UV)
 - ⇒ Bassa distorsione: ogni triangolo di mesh T_3 deve essere mappato in un triangolo di tessitura T_2 di forma simile e di area proporzionale
 - ⇒ Buon coverage: le parti di tessitura non coperte da nessun triangolo rappresentano uno spreco di memoria e vanno minimizzate
 - ⇒ Limitato numero di texture seams (see later)



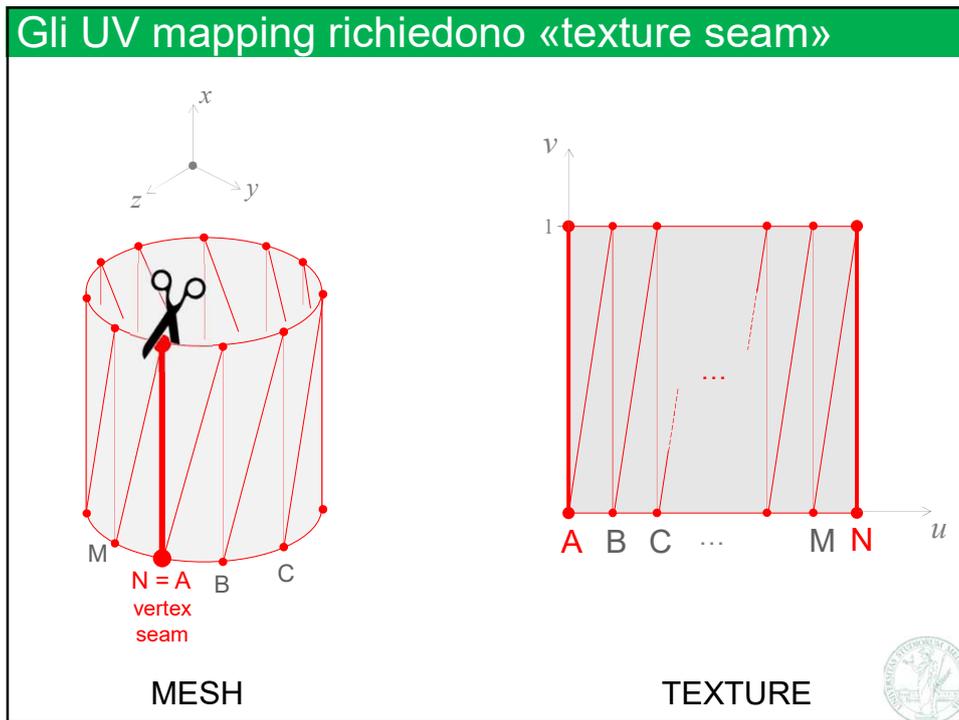
171

Texture seams (o cut)

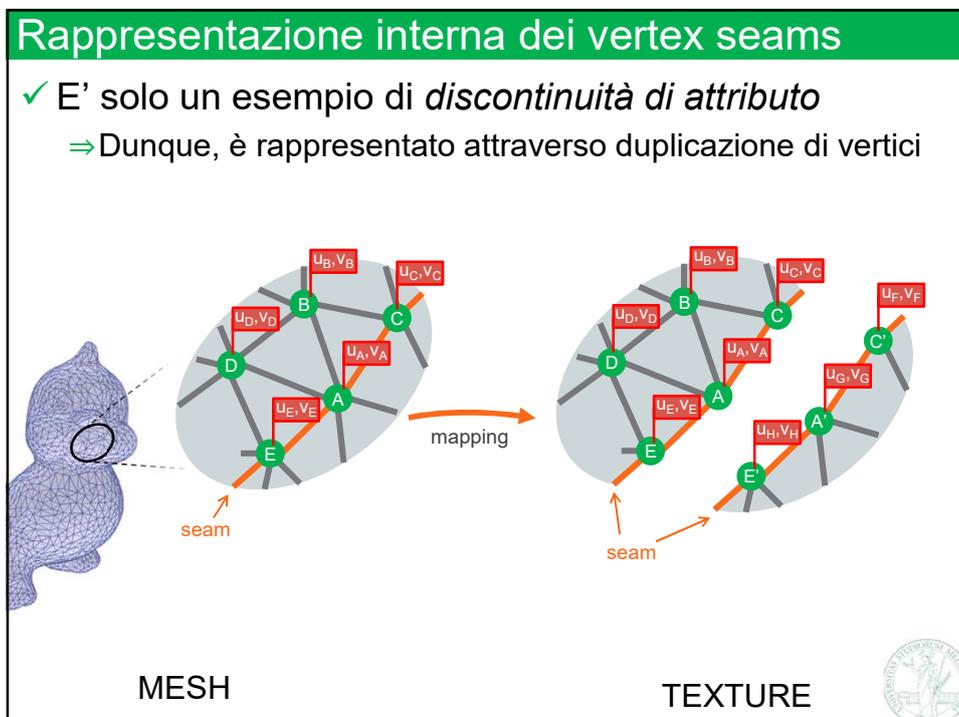
- ✓ **Nozione intuitiva:**
 - ⇒ il task di mesh parametrization ricorda quello di sbucciare una mela e disporre la buccia dentro un rettangolo
 - ⇒ Simile: come costruire un planisfero, cioè la mappa (2D) della superficie del pianeta (sfera in 3D)
(problema storicamente molto studiato: es. vedi: <http://vcg.isti.cnr.it/~tarini/spinnableworldmaps/>)
 - ⇒ La differenza è che la mesh ha una forma e una topologia arbitraria, piuttosto che sferica
- ✓ **Come questa intuizione suggerisce, un UV map richiede di introdurre dei tagli**
 - ⇒ detti «texture seams» o cuts
 - ⇒ questi tagli consentono di aprire la mesh per stenderla (unwrap) su un piano



172



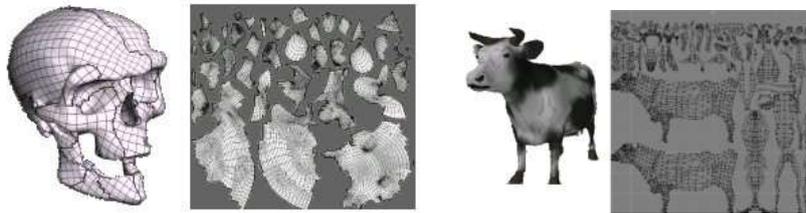
173



174

Geometry processing: mesh parametrization

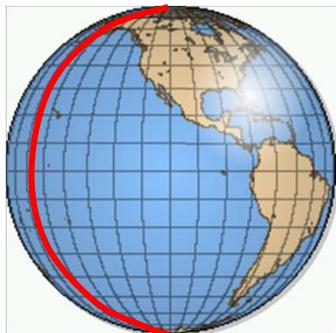
- ✓ Task: costruire automaticamente un UV-map per una mesh data
 - ⇒ problema difficile da risolvere in modo soddisfacente (soprattutto la scelta sul posizionamento/numero dei tagli)
 - ⇒ un gran numero di approcci algoritmici diversi
 - ⇒ rimane problema aperto: gli UV-map prodotti automaticamente sono spesso di scarsa qualità



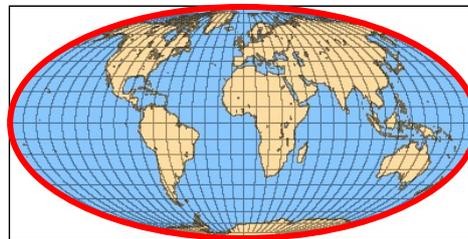
175

Cuts Or texture seams

- ✓ Clearly needed for closed surfaces



sphere in 3D



2D surface disk



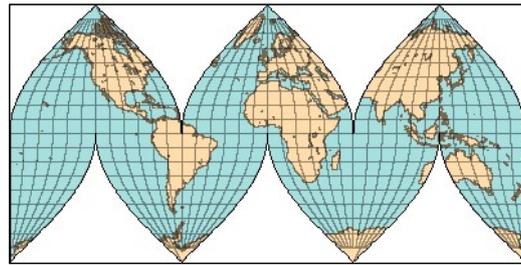
176

Cuts 2

✓ Usually more cuts -> less distortion



sphere in 3D



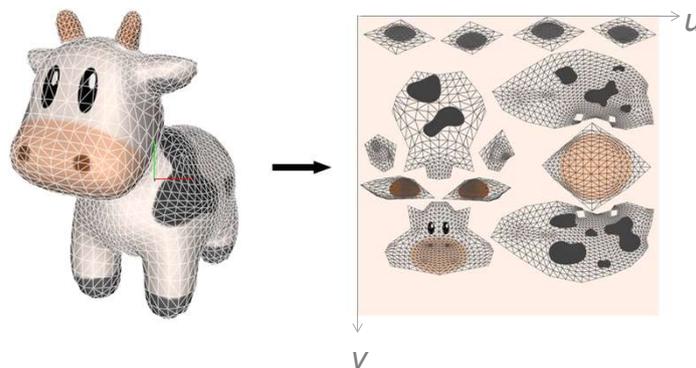
2D surface



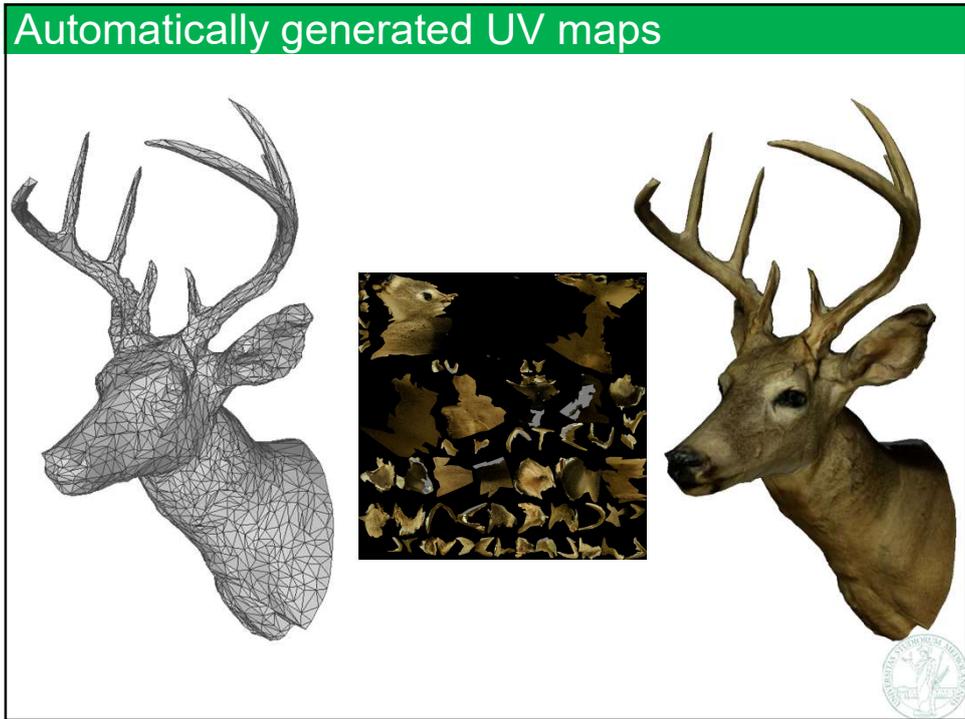
177

Texture Atlas

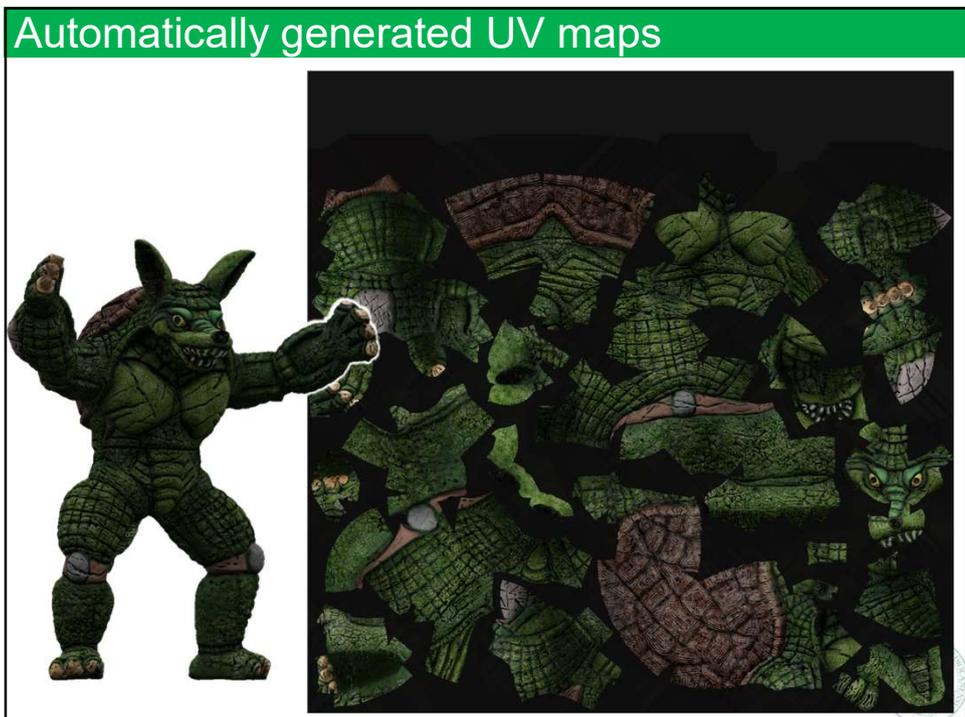
- Spesso la mesh viene divisa in zone, ciascuna parametrizzata in un "isola" di tessitura.
- Lo UV-map di questo tipo si dice «texture atlas» (un atlante di mappe)



178



179



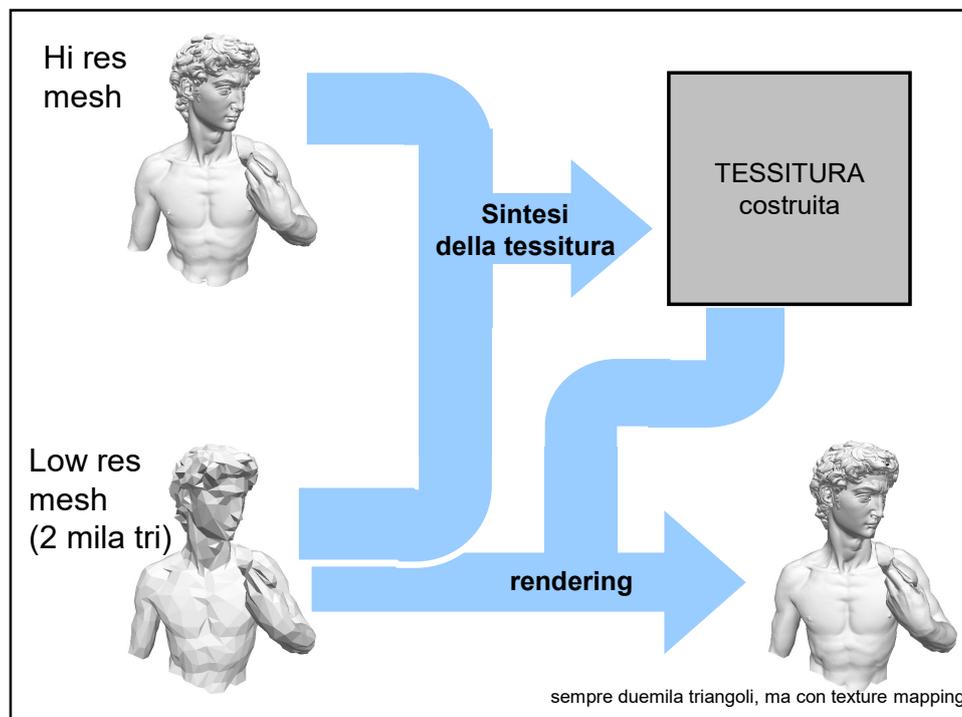
180

Campionare una texture da mesh hi-res

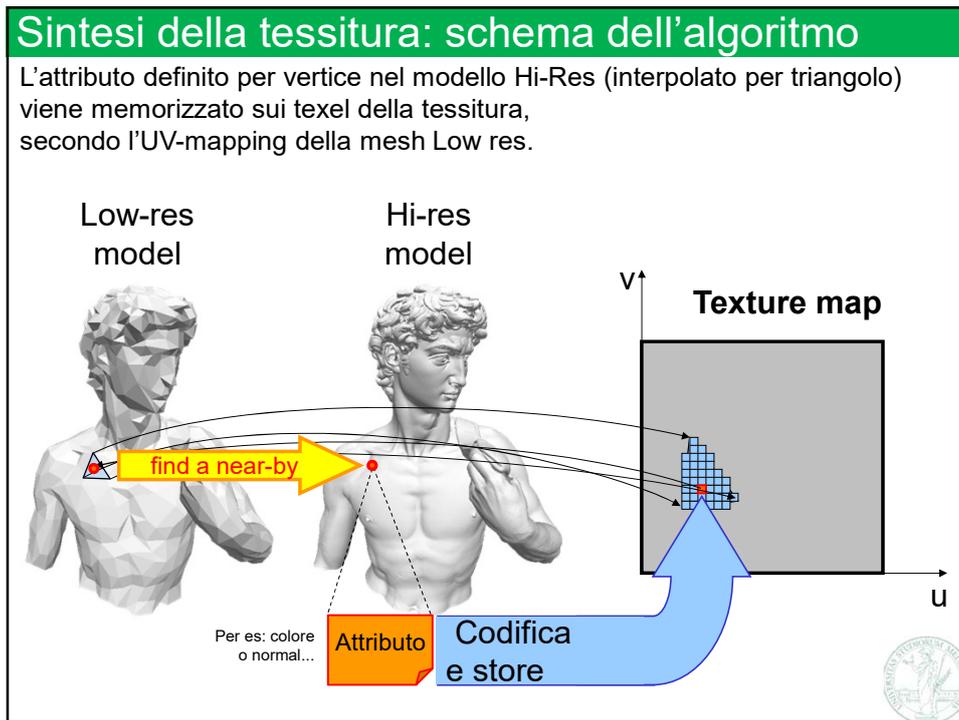
- ✓ Input:
 - ⇒ Mesh Hi-res M0, con attributi che ci interessa riprodurre
 - ⇒ Mesh Low-res M1, parametrizzata (con UV map)
- ✓ Output
 - ⇒ sintetizzare una tessitura per M1 che riproduca l'attributo definito su M0
- ✓ Un passaggio comune nel pipeline di creazione delle mesh
 - ⇒ Supportato per es da tutte le suit di modellazione 3D



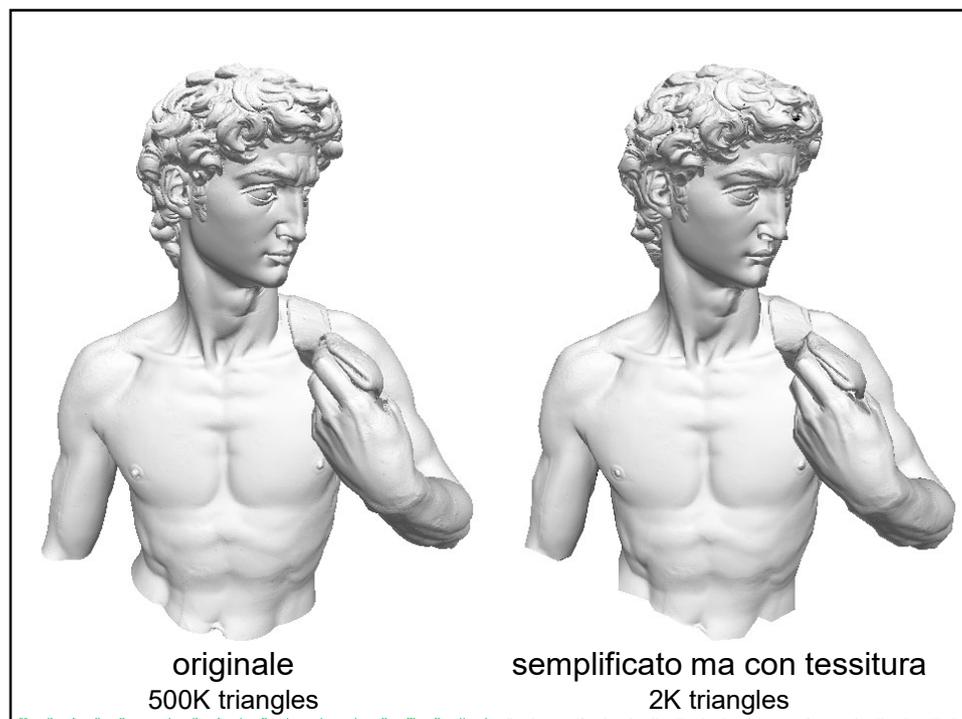
181



182



183



184

Textured Mesh VS Hi-resolution meshes

Hi-res mesh (con attributi)

- ✓ densità di campionamento segnale = densità di campionamento geometria
- ✓ aumentare i vertici richiede di incrementare anche la connettività (e quindi di processare più triangoli durante il rendering)
- ✓ non richiede vertex seams (se il segnale è continuo)
- ✓ normale & posizione per vertex: la forma è catturata per intero, compreso nei suoi dettagli (questo permette di tenerne conto in tutti i tipi di processing)

Low-res mesh con texture

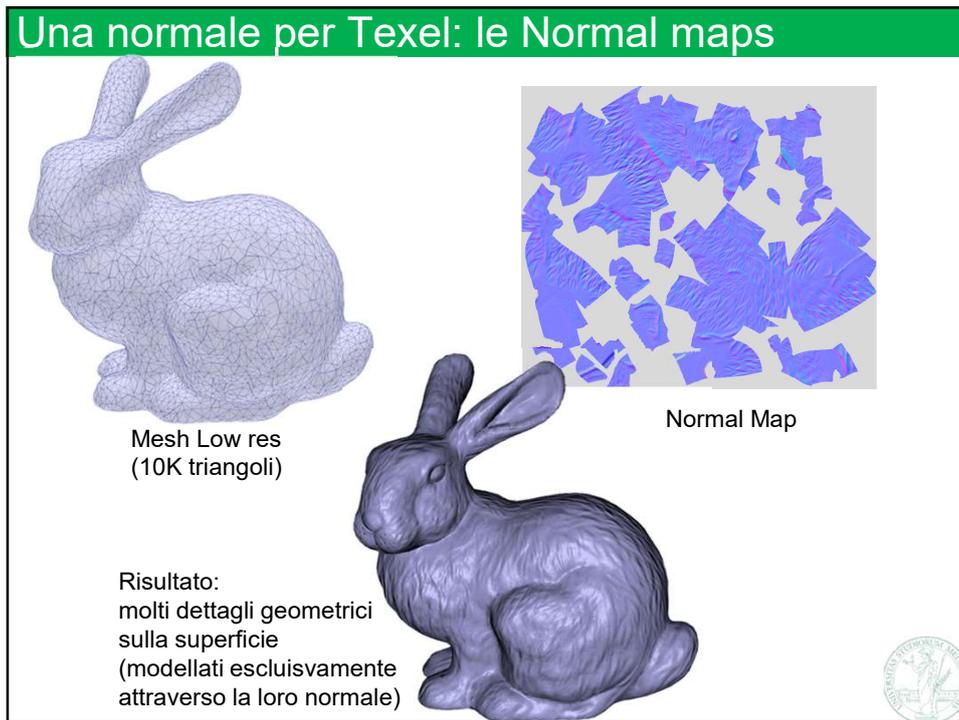
- ✓ risoluzione texture (colore, etc) indipendenti da risoluzione mesh (geometria)
- ✓ incrementare i campioni di tessitura ha un costo piccolo
- ✓ costo extra: UV-map sulla mesh. Va costruito (difficile); va mantenuto in memoria; richiede di introdurre alcuni vertex seams.
- ✓ in totale, **meno onerosa** in termini di spazio (in memoria), e tempo (rendering)
- ✓ normal-map: i dettagli di forma sono solo riprodotti visualmente attraverso il **lighting**

185

Una normale per Texel: le Normal maps

- ✓ Una **normal-map** è una texture che campiona le normali della superficie
 - ⇒ in questo modo, contribuisce a specificare la forma geometrica dell'oggetto
 - ⇒ (introducendo però una approssimazione)
 - ⇒ tipicamente, la normal map riproduce dettagli geometrici minuti, ad alta frequenza (come gli avvallamenti su una buccia di arancia), mentre la mesh riproduce la forma generale dell'oggetto (come la forma sferica dell'arancia)
- ✓ Uno stesso oggetto può essere rappresentato, con simile accuratezza e una resa simile, attraverso ...
 - ⇒ una mesh ad alta risoluzione, oppure
 - ⇒ una mesh a bassa risoluzione ma provvista tessitura per dettagli «ad alta frequenza» (come una normal-map e/o una tessitura colore)
- ✓ Quali vantaggi e svantaggi?

187



188