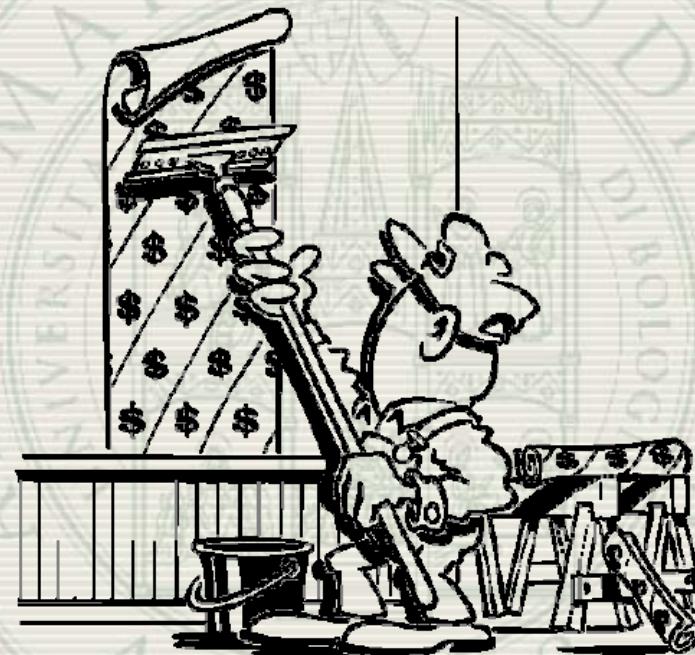


Texture nella CG



Texture Mapping

Con **Texture** ci si riferisce ad un'immagine di qualsiasi tipo utilizzata per rivestire la superficie di un oggetto o modello 3D; la superficie apparirà molto più complessa di quello che è, ma soprattutto più realistica.

Con **Texture Mapping** ci si riferisce al processo di applicazione di un'immagine sulla superficie di un modello 3D.



Texture Mapping

Se simuliamo l'illuminazione della scena quali attributi o parametri dell' **oggetto** devono essere modificati per produrre l'effetto desiderato?

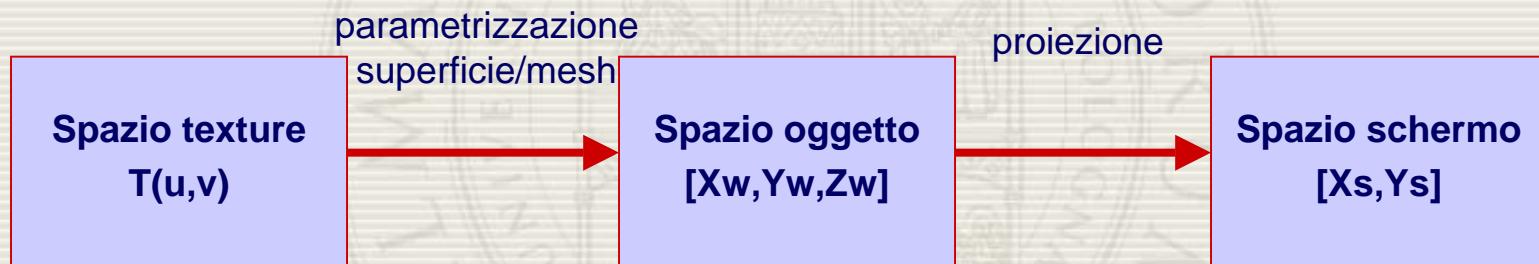
I parametri più comunemente usati per il texture mapping sono i **coefficienti di illuminazione ambiente e di riflessione diffusa (o colore)** dell'oggetto, cioè **K_a** e **K_d**.

Ancora si può agire sulla **normale** una cui perturbazione influenza la riflessione della luce producendo un effetto simile a quello di un materiale non liscio.

Texture Mapping 2D

Come si deve procedere ?

Una texture viene definita in un dominio (o spazio texture), mentre l'oggetto è nello spazio oggetto; si deve definire una trasformazione fra questi: **parametrizzazione**



Texture Mapping 2D

Esempio di mapping fra spazio texture e mesh 3D.



Problema difficile, sia che sia fatto a mano che, soprattutto, in modo automatico

Texture Mapping 2D

Texture Mapping 2D in un contesto Z-Buffer.

Lo **Z-Buffer** implica un processo pixel per pixel per ogni faccia/triangolo; questo significa che si deve individuare un singolo valore **texture (texel)** per ogni pixel, per essere inserito nello schema shading.

Per fare questo si fa uso di un **Inverse Mapping**; si trova la preimmagine del pixel corrente nel dominio **texture**.

Il **texture mapping** richiede un trattamento speciale di **Antialiasing** poiché tende a produrre effetti di **Aliasing**.

Texture Mapping 2D: esempio

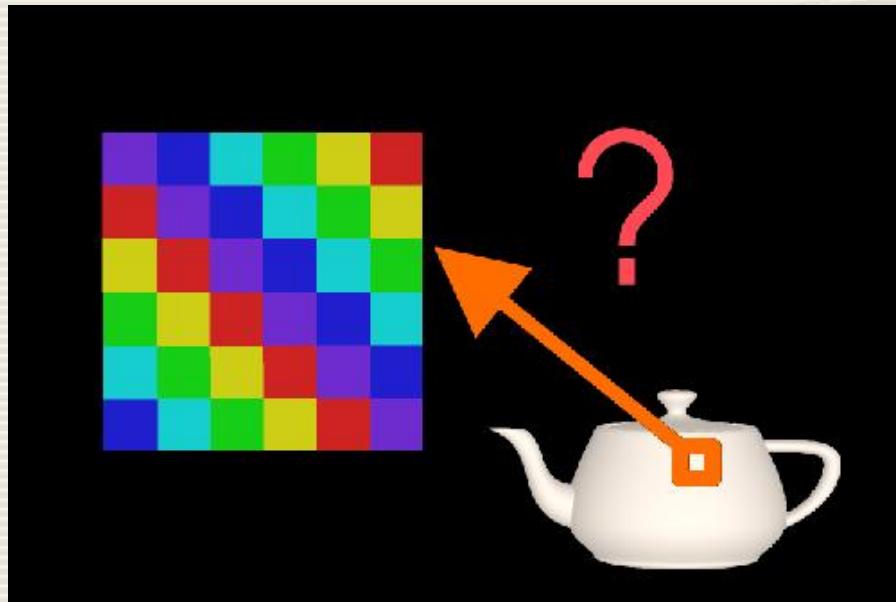
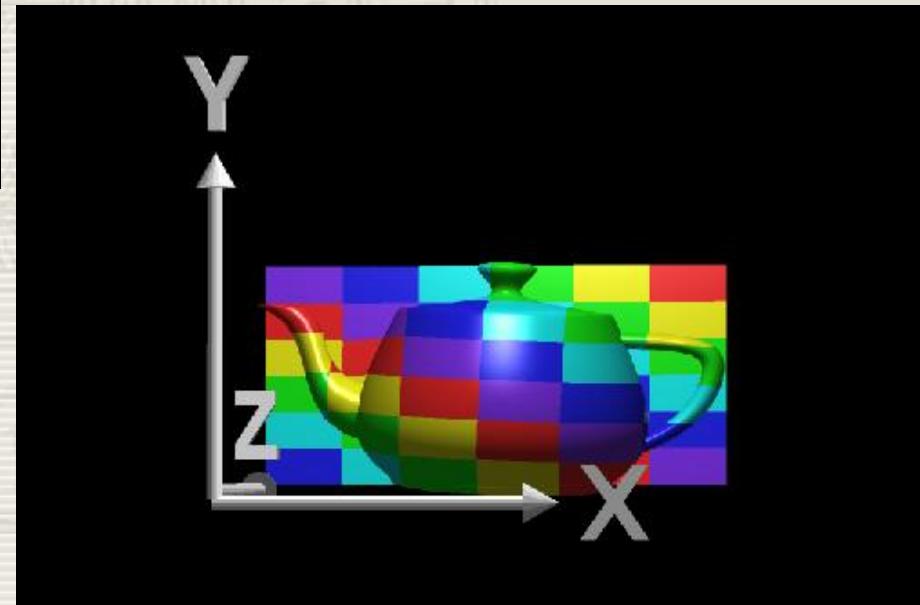


Image Mapping

Parametrizzazione
 $[u,v]=[x,y] \rightarrow [x,y,z]$



Texture Mapping 2D: esempio

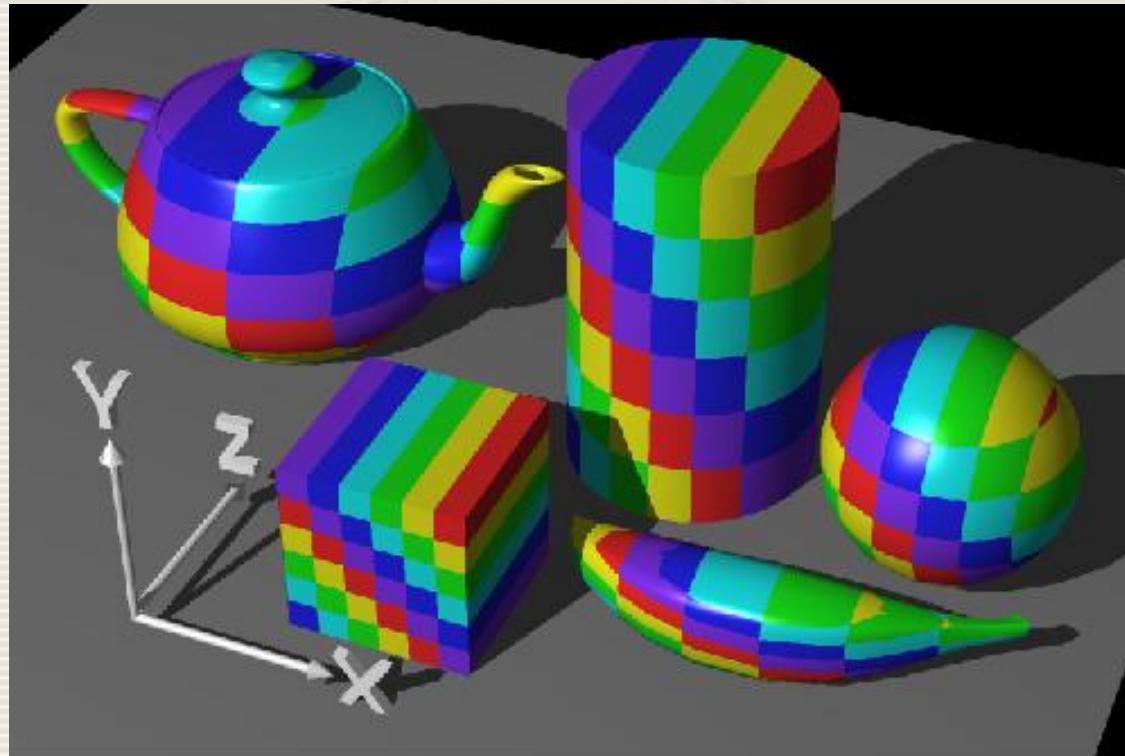


Image Mapping secondo parametrizzazione
 $[u,v]=[x,y] \rightarrow [x,y,z]$

Texture Mapping 2D: esempio



parametrizzazione $[u,v]=[x,y] \rightarrow [x,y,z]$

Texture Mapping 2D: esempio

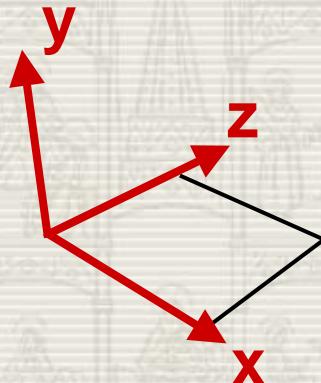
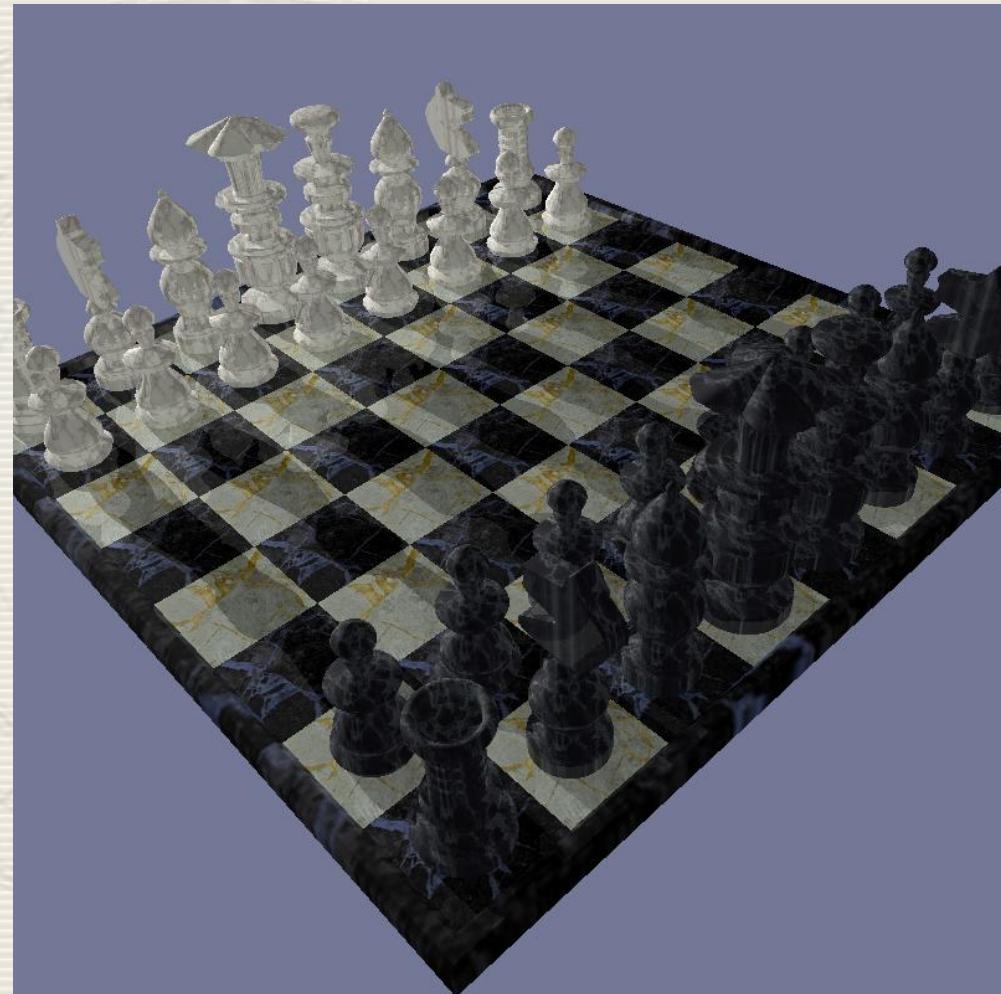
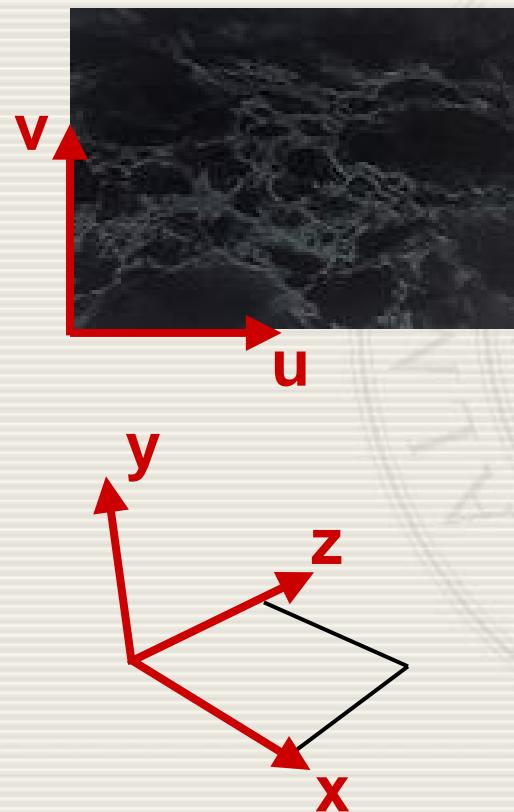


Image Mapping,
parametrizzazione $[u,v] = [x,z] \rightarrow [x,y,z]$;
lo spazio texture è stato posizionato alla base
dell'oggetto

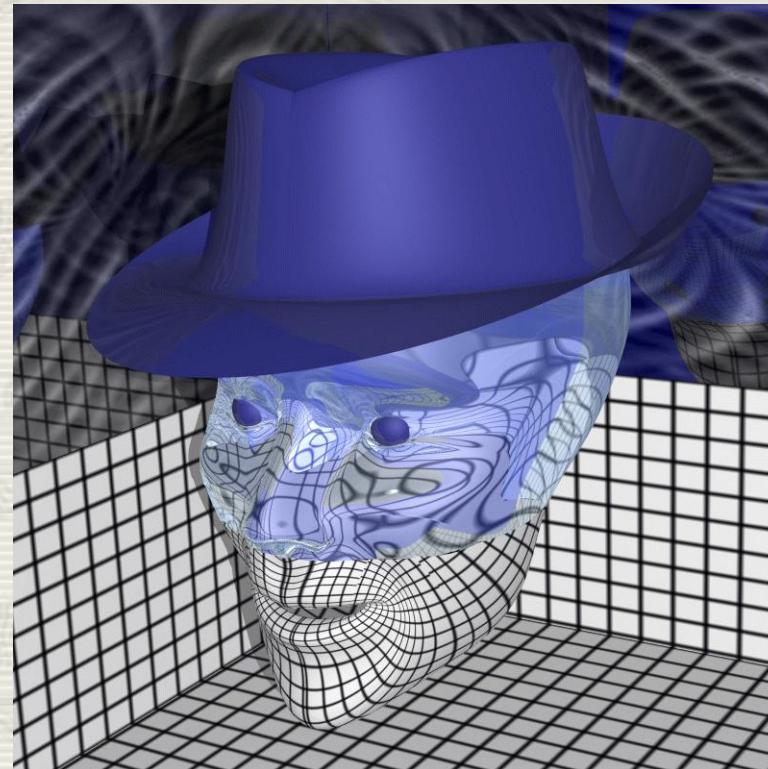
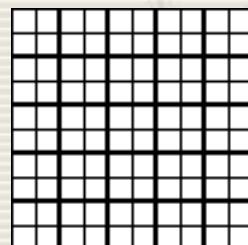
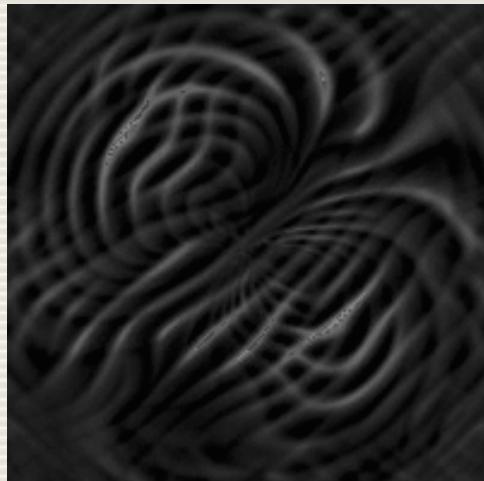
Texture Mapping 2D: esempio



Texture Mapping 2D: esempio



Texture Mapping 2D: esempio

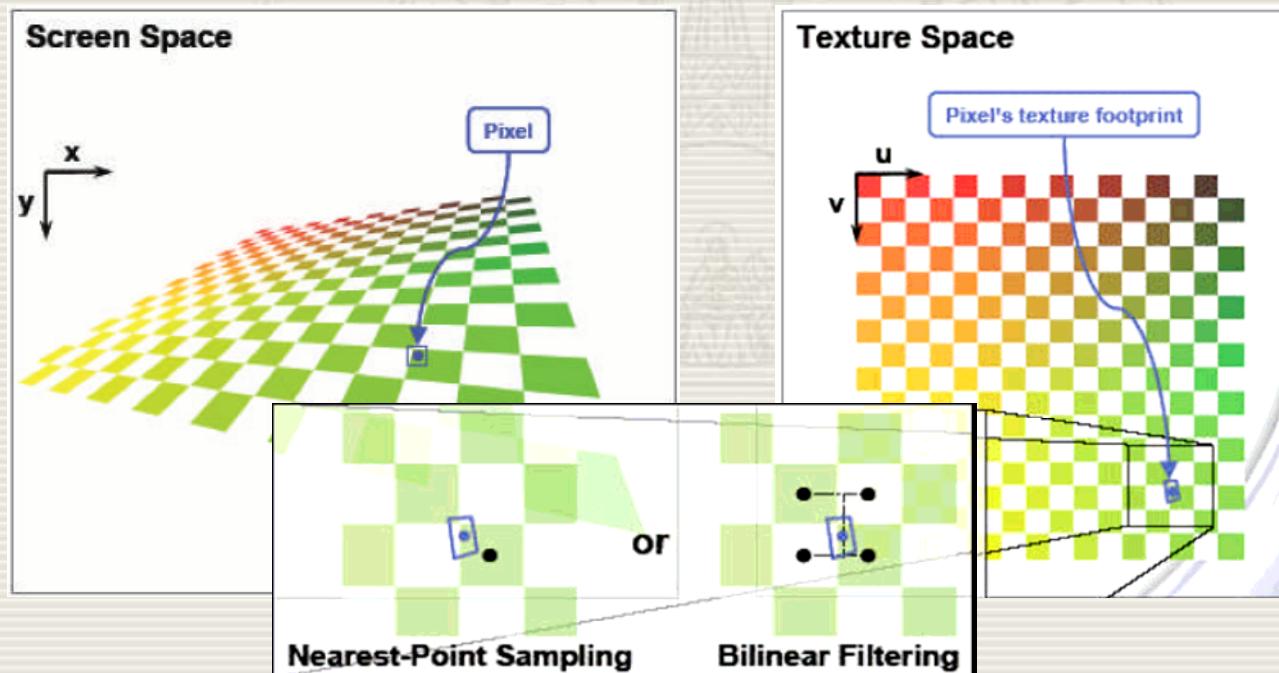


Logo Corso Grafica A.A.2003/04

Magnification e Minification

Pixel → Inverse Mapping → Texel

Cosa accade se un texel è più grande di un pixel?
Minification: la texture deve essere rimpicciolita.

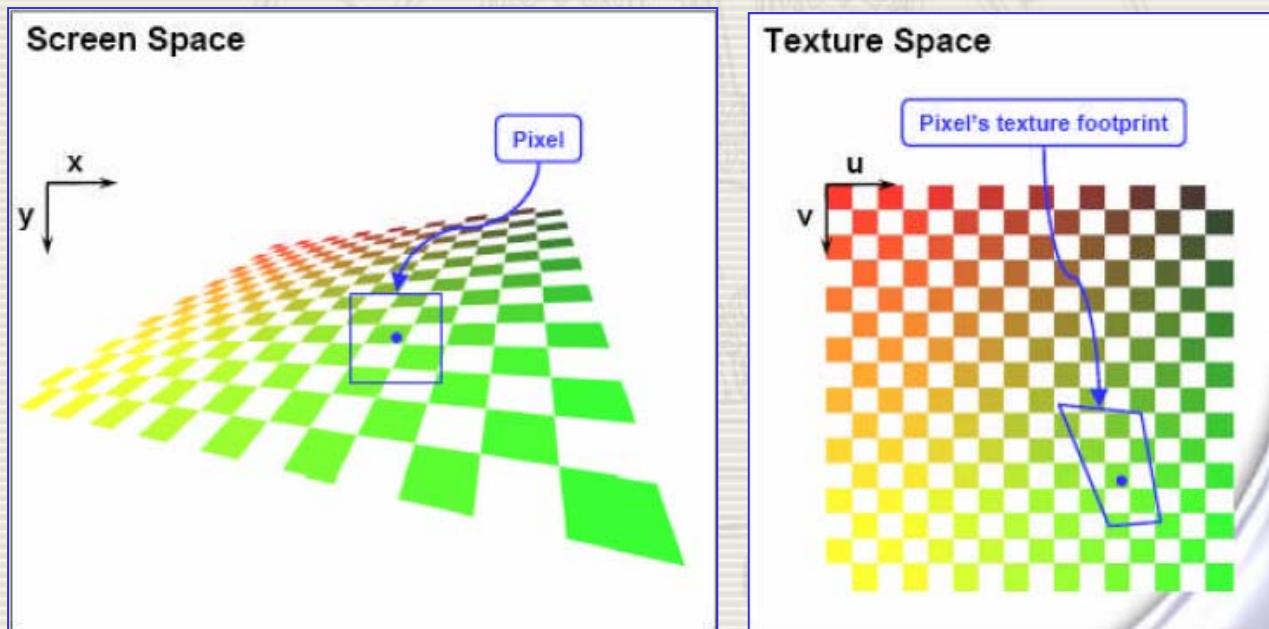


Magnification e Minification

Pixel → Inverse Mapping → Texel

Cosa accade se un texel è più piccolo di un pixel?

Magnification: la texture deve essere ingrandita.



Magnification e Minification

Tecniche di Antialiasing (o Filtering) più usate:

Nearest Neighbor:
ad ogni pixel viene
dato il colore del
texel più vicino;
più pixel avranno
stesso colore



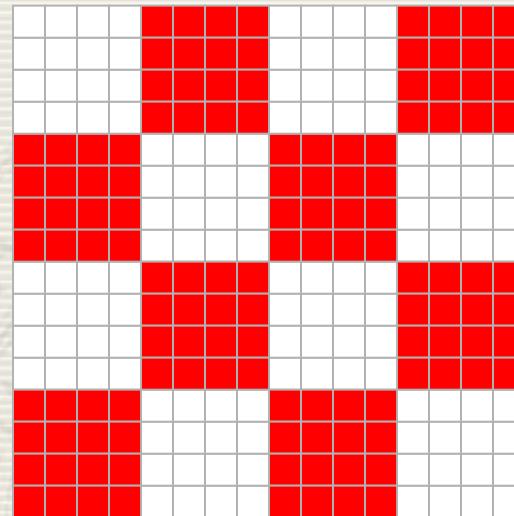
Effetto Pixelation



Effetto Blur

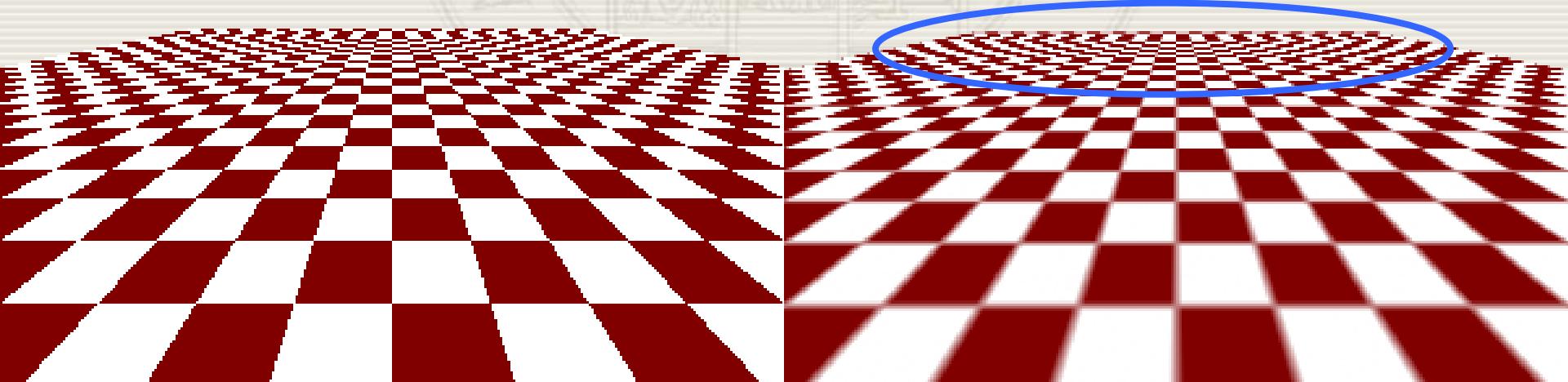
Bilinear Interpolation:
per ogni pixel
vengono presi i 4
texel più vicini e
interpolati
linearmenete nelle due
dimensioni per avere
un valore medio per
il pixel

Esempio di Minification



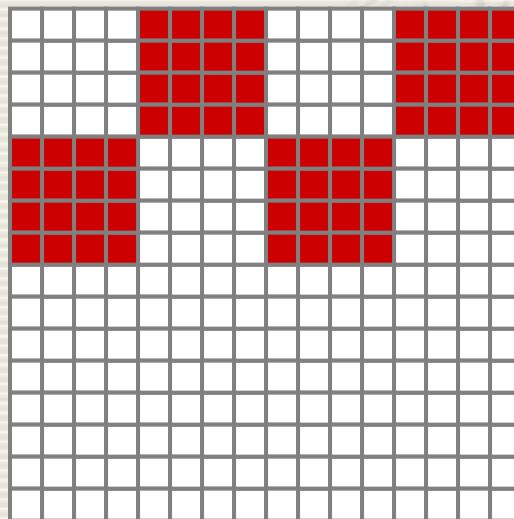
Nearest Neighbor

Bilinear Interpolation
non risolve il problema

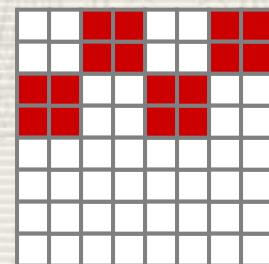


MIP Mapping

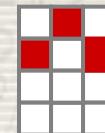
Multum In Parvo: molte cose in poco spazio



Texture originale

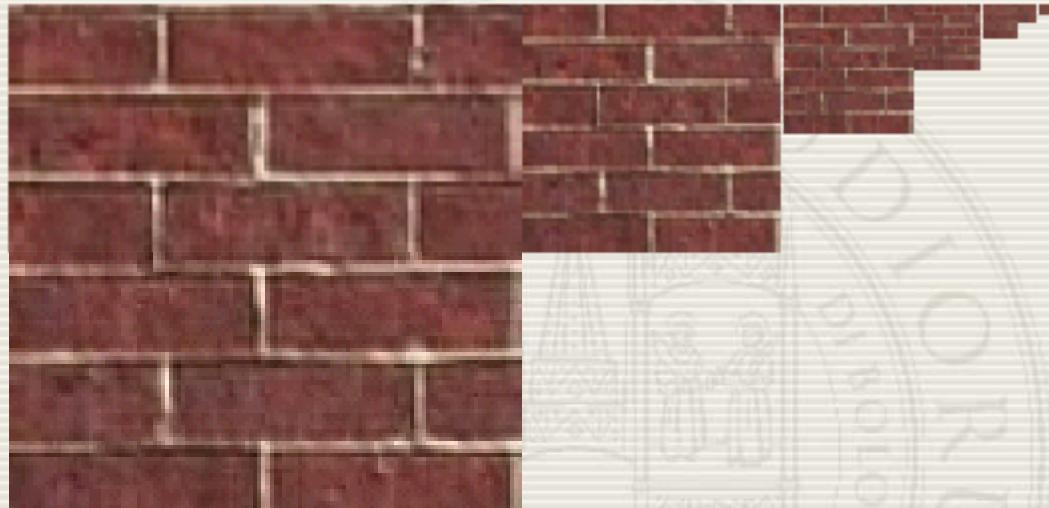


versioni a bassa risoluzione



Per una texture di $2^n \times 2^n$ texel, si calcolano $n-1$ texture,
ciascuna a metà risoluzione della precedente.

MIP Mapping



Generare una MIP-map da una texture è facile:

per ogni texel al livello i , si fa la media dei valori dei 4 corrispondenti texel al livello $i-1$.

Mip Mapping

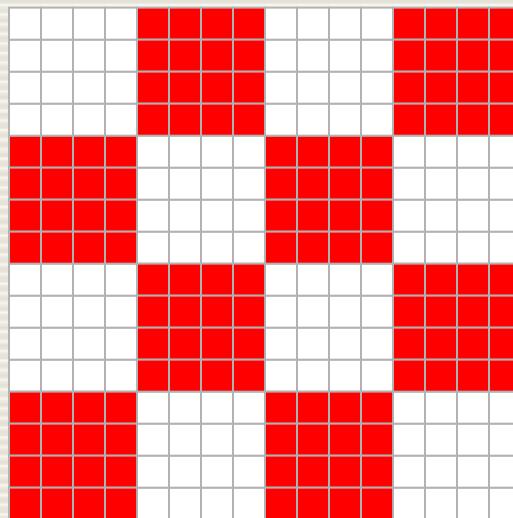
Ogni livello della MIP-map rappresenta una versione pre-blurred di texel multiple

- Un texel al livello n rappresenta 2^n texel originali

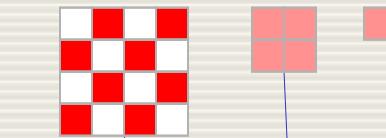
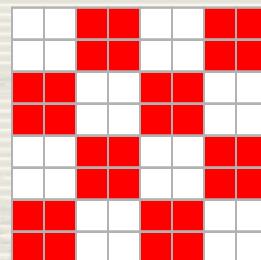
Quando fare il rendering?

- Immaginiamo la texture ricoperta di pixel (cioè la dimensione dei pixel in texel nella texture originale)
- Si determini il livello del MIP map in cui i texel sono approssimativamente in corrispondenza numerica con i pixel
- Si interpolino i valori dei 4 texel più vicini

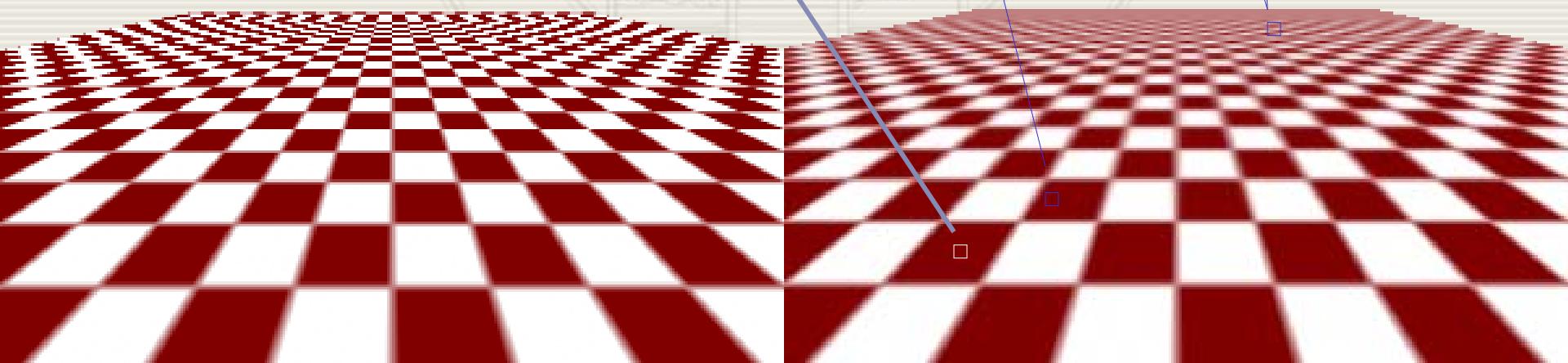
MIP Mapping: esempio



Bilinear Interpolation
non risolve il problema



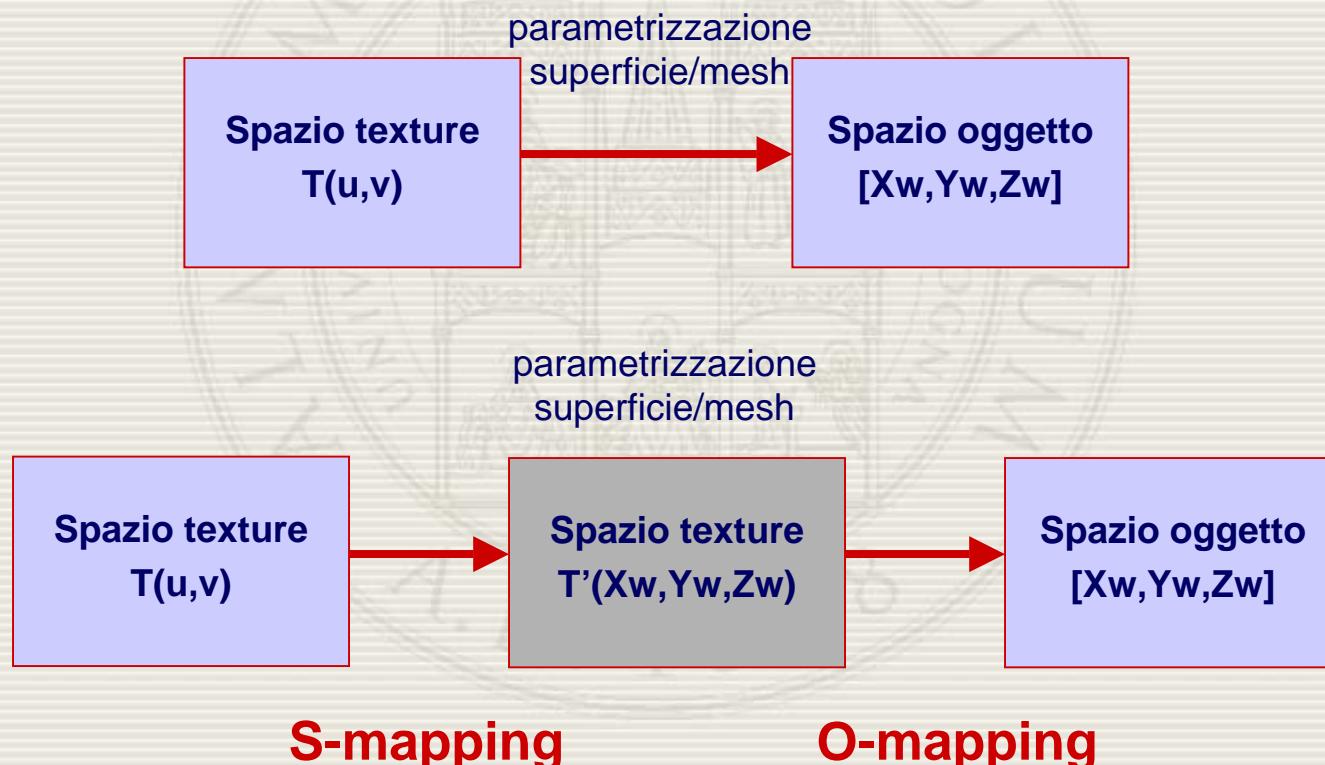
MIP-mapping



Texture Mapping 2D

Two-Part Mapping

1. S-mapping
2. O-mapping



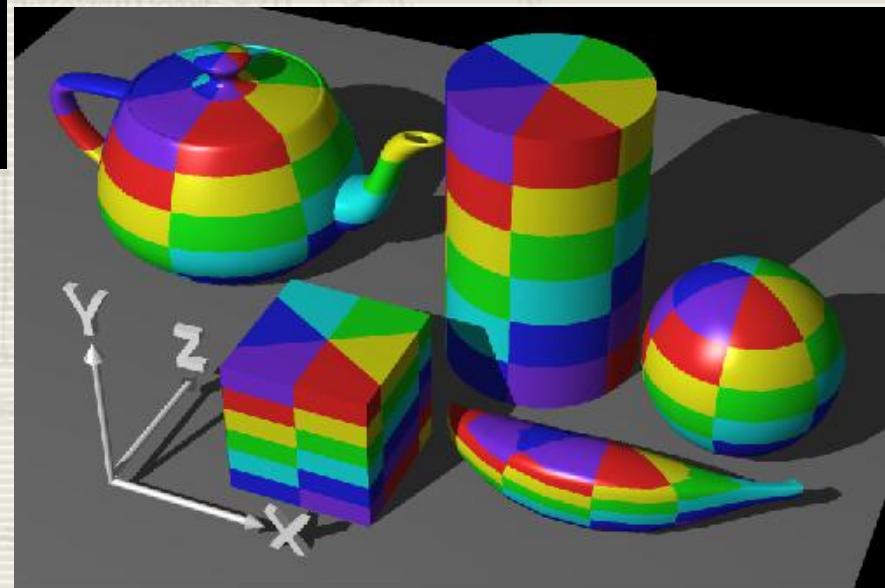
Texture Mapping 2D



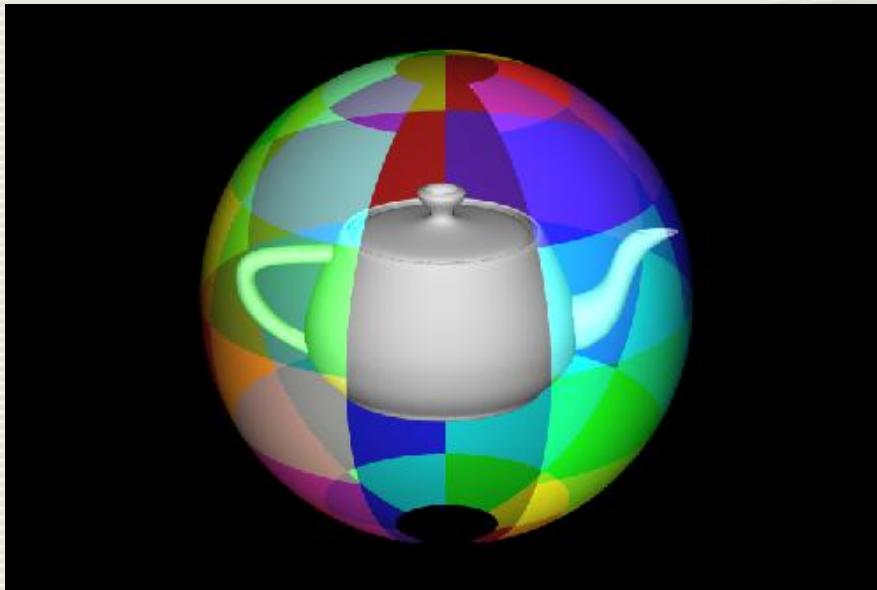
Superficie **S** intermedia:
Cilindro

Two-Part Mapping

1. S-mapping
2. O-mapping



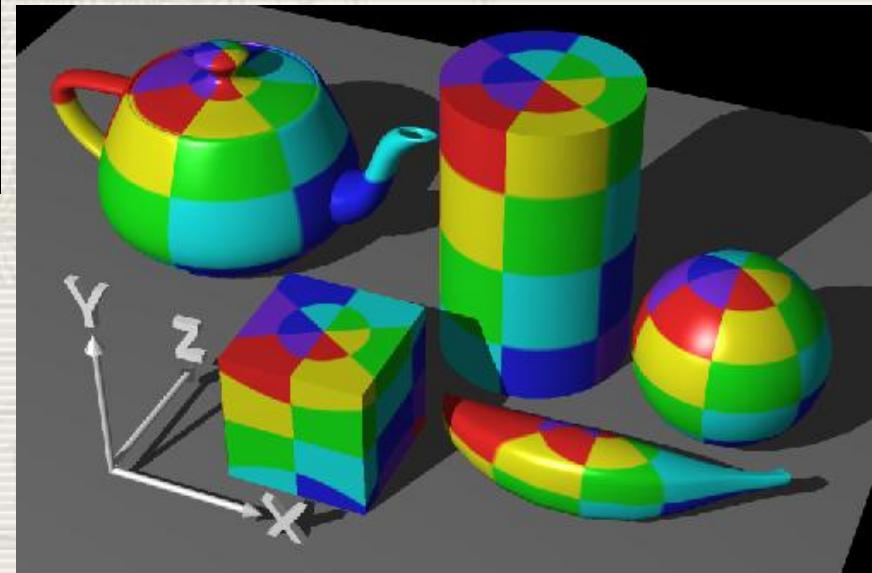
Texture Mapping 2D



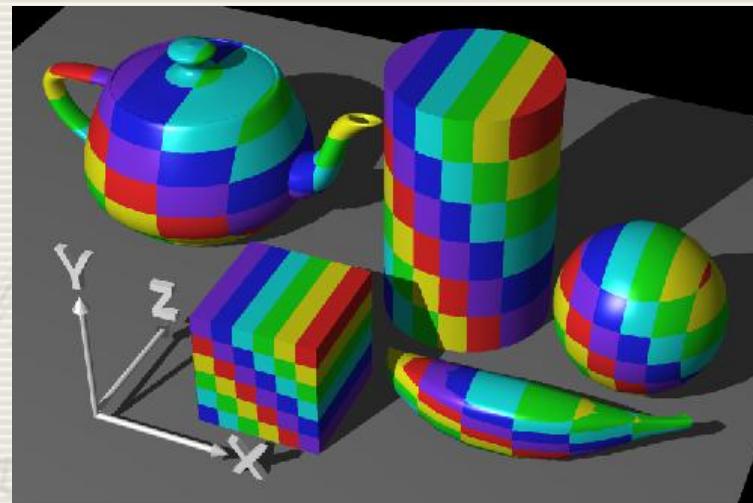
Superficie **S** intermedia:
Sfera

Two Part Mapping

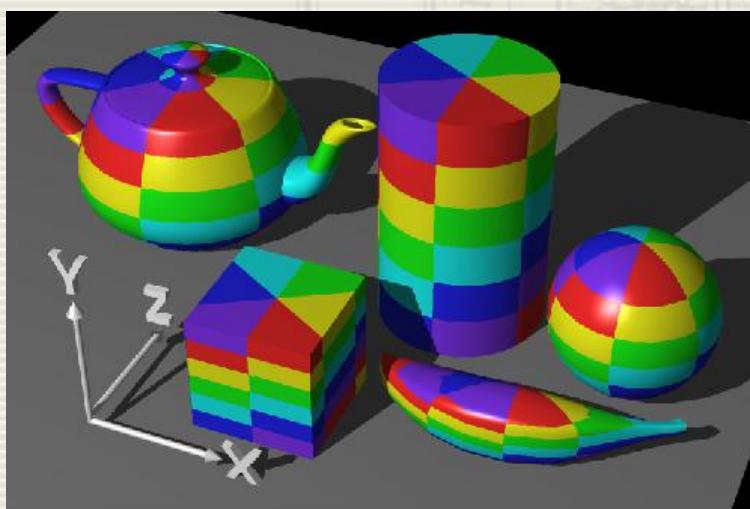
1. S-mapping
2. O-mapping



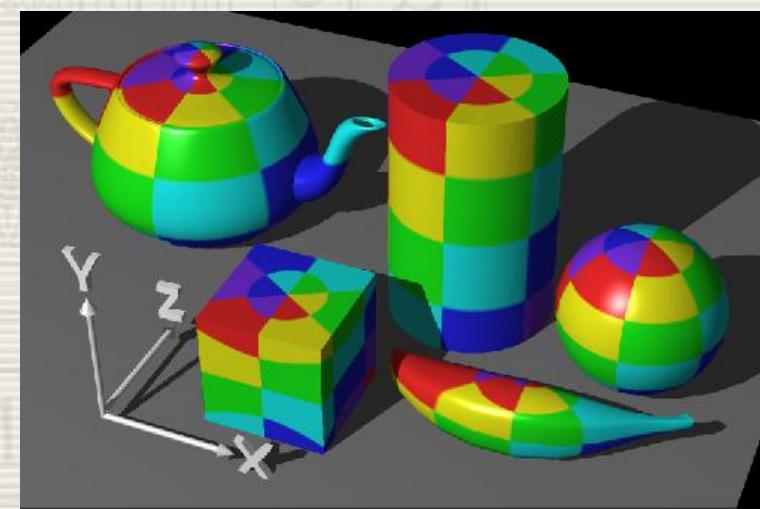
Texture Mapping 2D: confronto



Piano XY

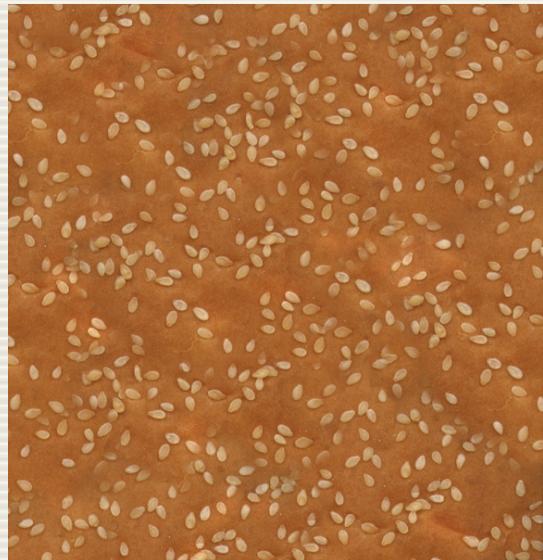


Cilindro

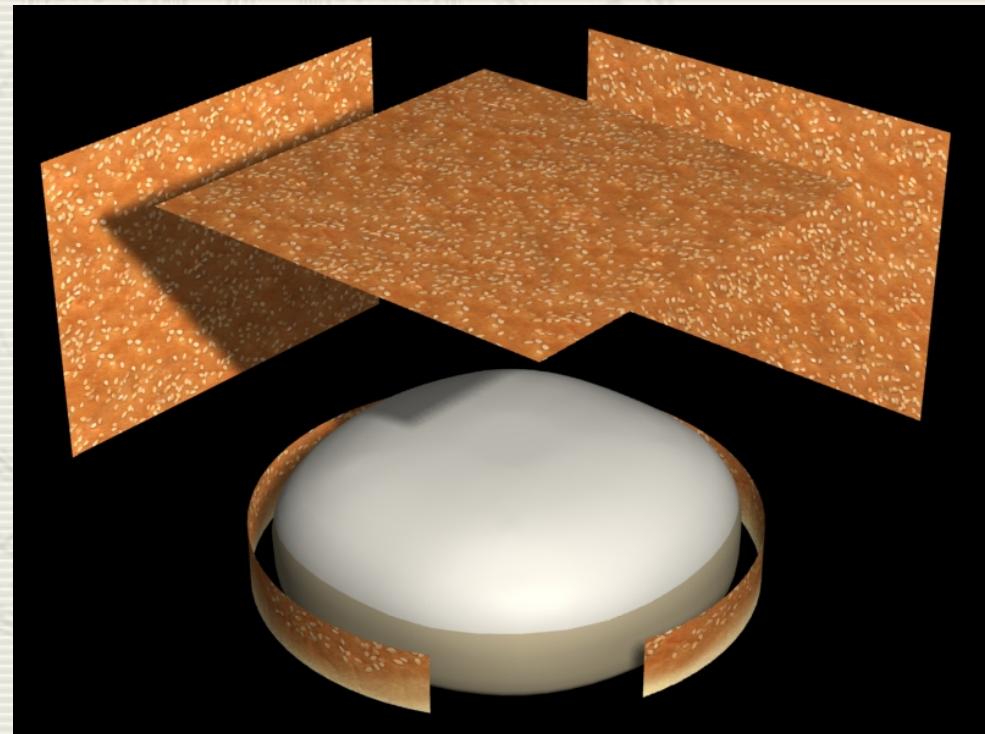


Sfera

Texture Mapping 2D: esempio



Two Part Mapping



Texture Mapping 2D: esempio



Texture Mapping 2D: esempio

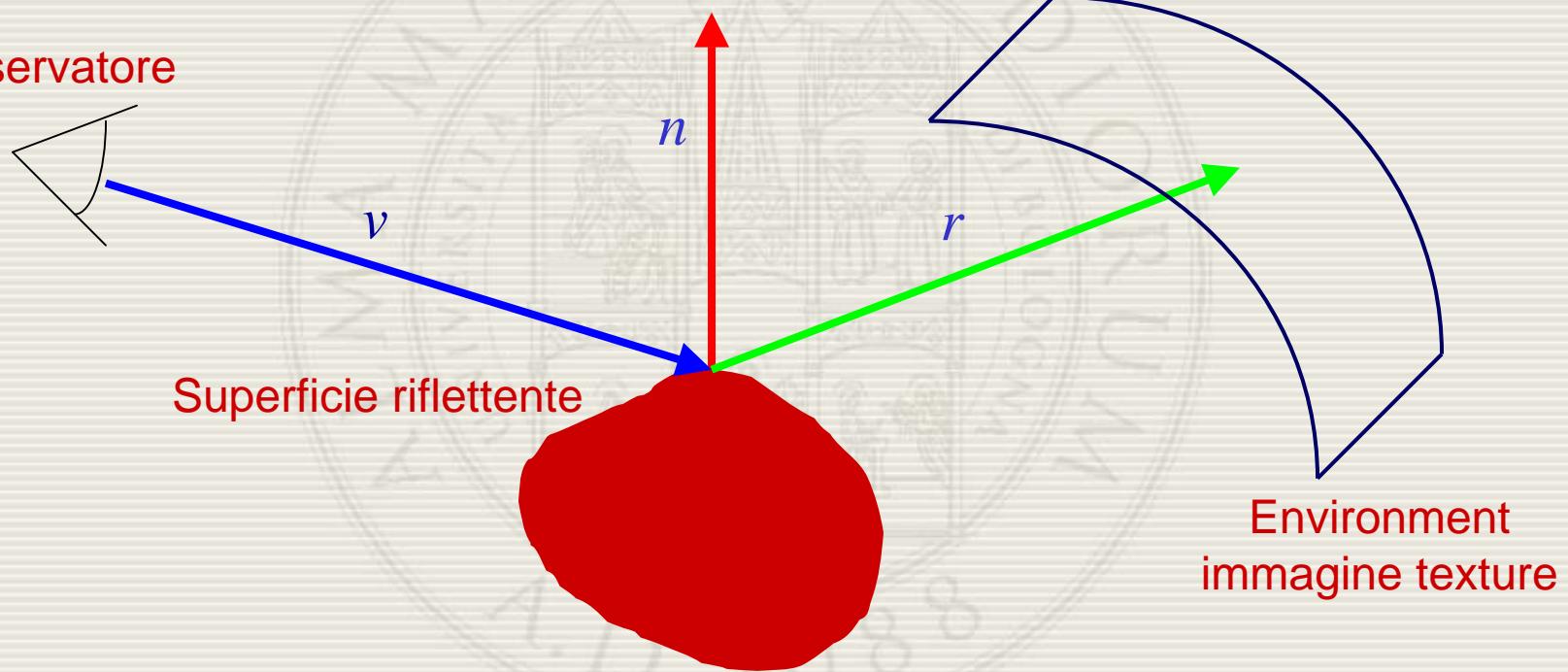
Two Part Mapping:
cilindro



Environment Mapping

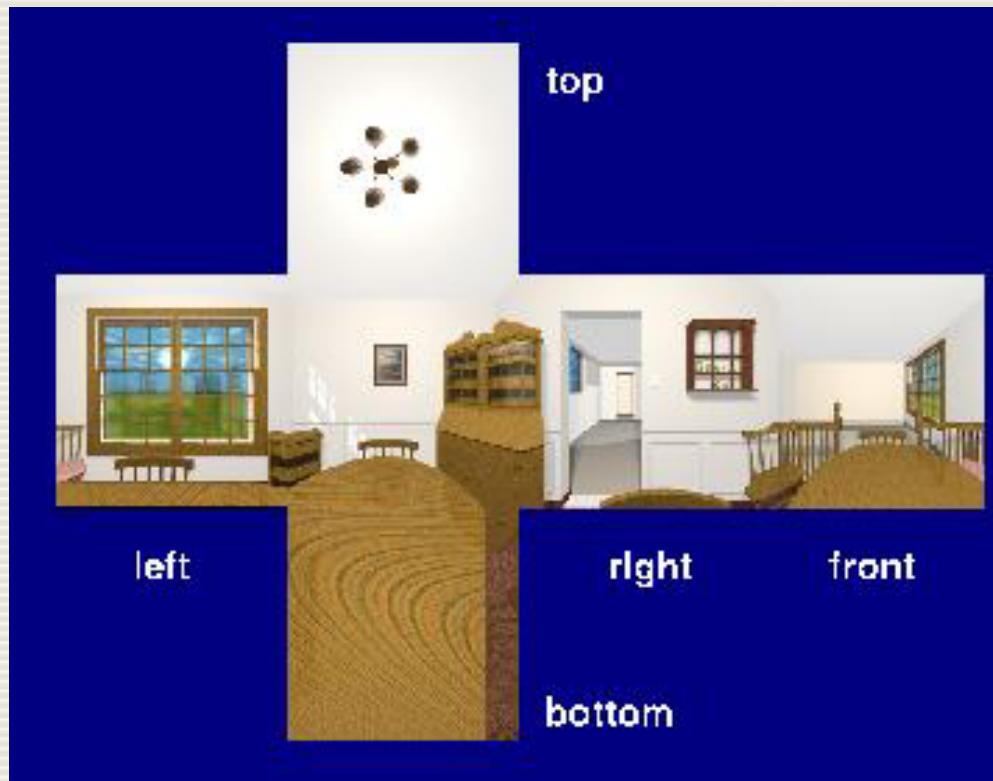
$$\text{Raggio riflesso: } r = 2(n \cdot v)n - v$$

Osservatore



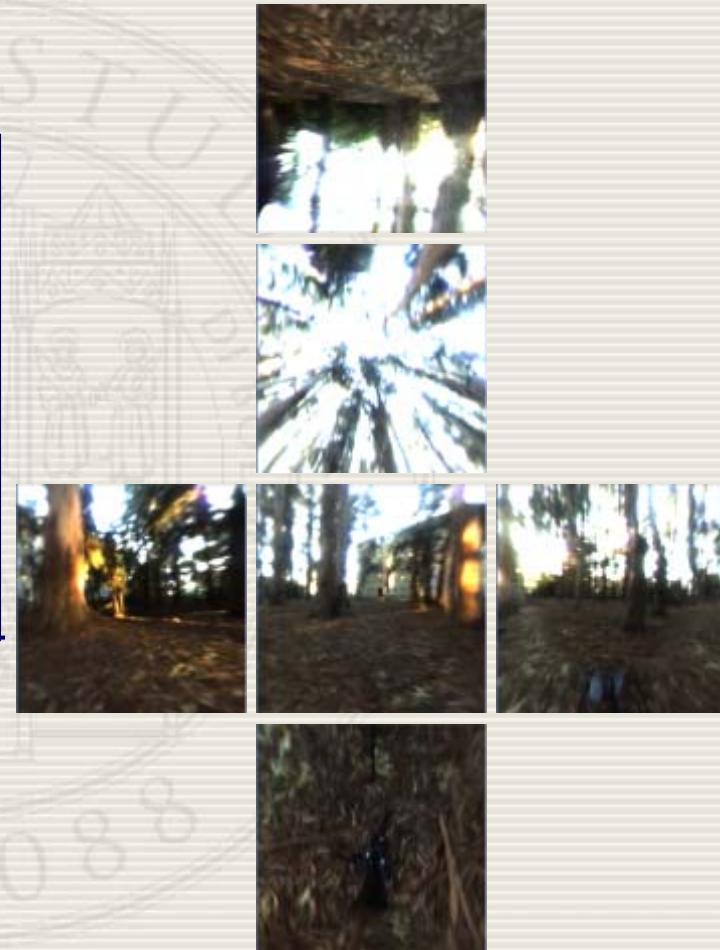
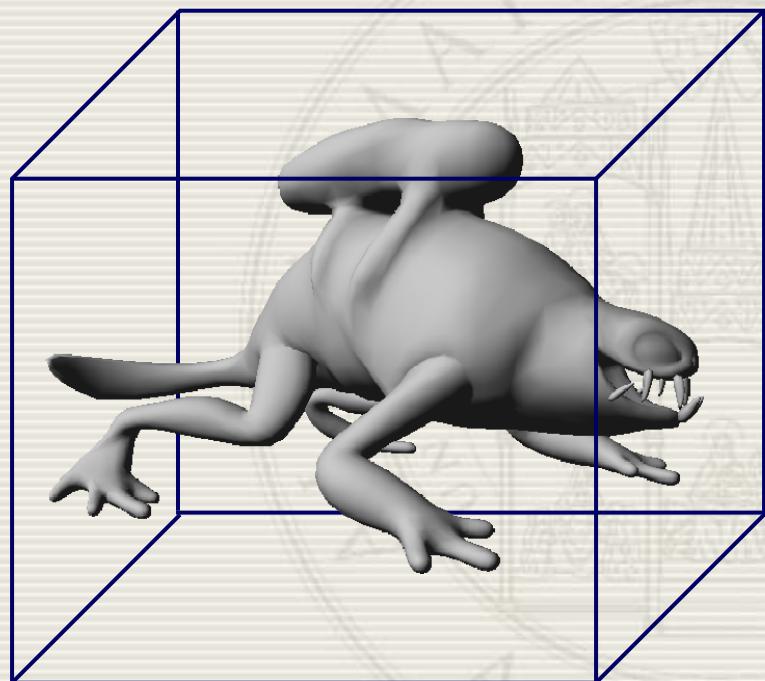
La texture viene applicata nella direzione del raggio riflesso r , dall'environment immagine texture sull'oggetto

Cube Environment Mapping

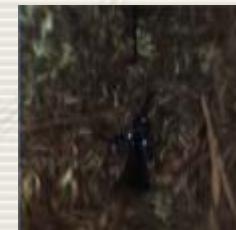
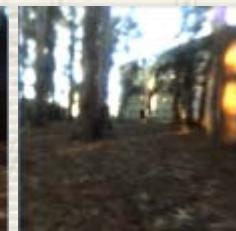
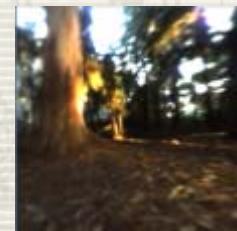
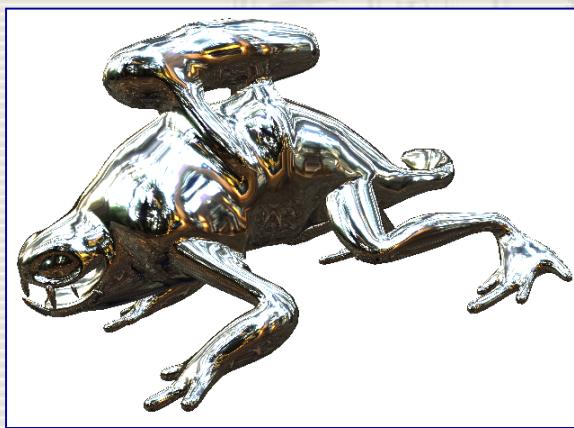
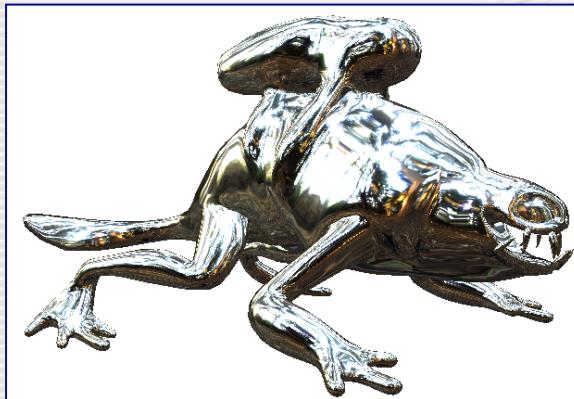


Esempio: `opengl_1516/opengl_advanced/texture/envSMap.c`

Cube Environment Mapping



Cube Environment Mapping

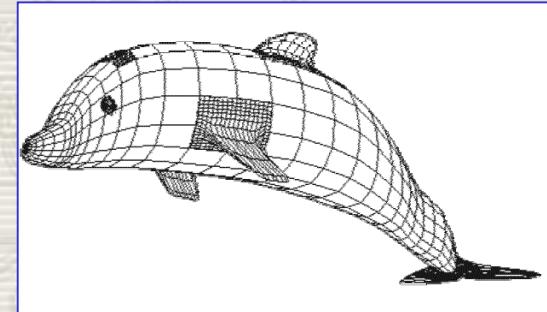
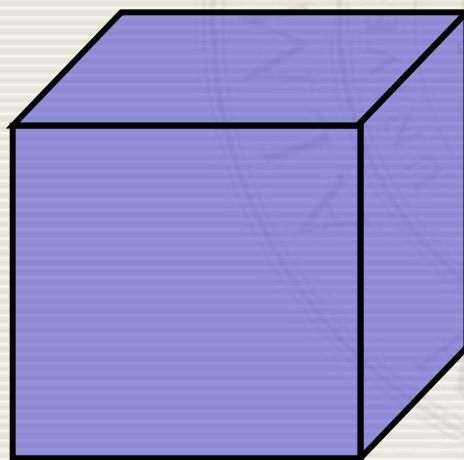


Solid Texturing o 3D Texture

(anche Texture Mapping 3D o Texture Procedurale)

Texture a 3 dimensioni che viene applicata ad un modello 3D

L'oggetto 3D viene “immerso” nello spazio texture 3D e il colore di ogni punto dell'oggetto viene definito dalla sua posizione nello spazio texture.



Solid Texturing

Vantaggi:

- .Non è più necessario applicare texture 2D a modelli 3D**
- .Possibilità di simulare in maniera realistica materiali naturali (uso di texture procedurali 3D)**
- .Nuove possibilità di applicazioni**

Svantaggi

- .Alta occupazione di memoria delle texture (una texture 64x64x64 RGBA occupa 1MB, una texture 256x256x256 RGBA occupa 64MB)**
- .Per texture procedurali overhead computazionale che ne limita l'uso in applicazioni real-time (a meno che non siano state pre-calcolate)**
- .Difficoltà d'uso per l'applicazione all'oggetto di immagini (texture come immagine e non come procedural texture, es: volto di una persona)**

Solid Texturing: esempio



wood texture

Lo spazio 3D texture consiste in cilindri concentrici e alterni in colore; come un tronco d'albero.

Solid Texturing: esempio

cube texture



Lo spazio 3D texture consiste in una scacchiera 3D

Texture Procedurali

L'approccio procedurale non nasce con le texture, ...

...ma trova la sua massima espressione e sviluppo con le texture 3D. E' possibile costruire texture procedurali anche 2D o 4D (dove le 4^a dimensione può essere il tempo – utile per creare per es. nuvole realistiche).

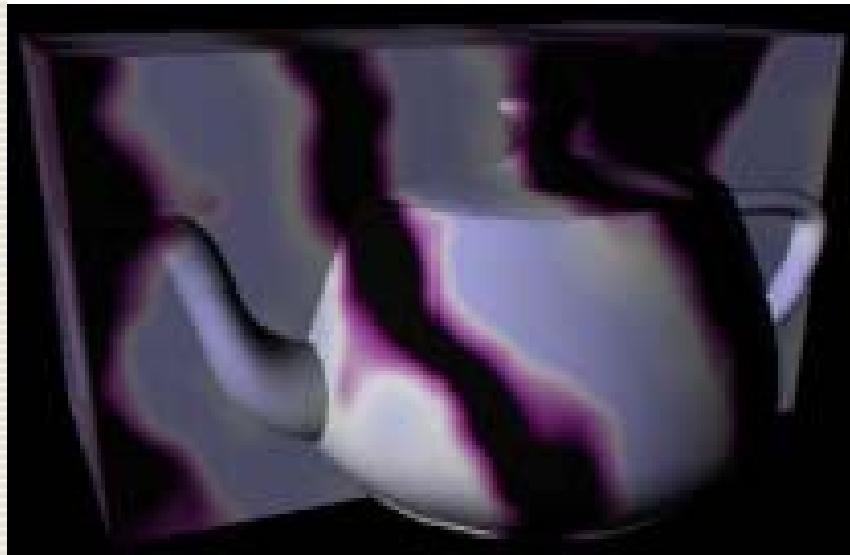
Per metodo procedurale si intende un metodo matematico, tipicamente pseudo-casuale o di tipo frattale (sono anche dette solid noise function). Questi metodi sono di particolare importanza proprio per quanto riguarda la costruzione di texture che rappresentano materiali naturali quali appunto il legno, il marmo, la roccia, ecc...

Texture Procedurali



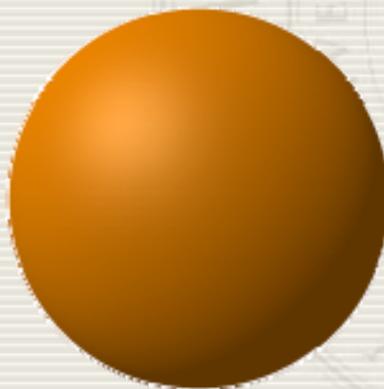
3D Perlin Noise

Texture Procedurali o Solid Texturing: esempio



Bump/Normal Mapping

- Si sostituiscono i colori R,G,B con coordinate X,Y,Z
- Si interpretano i pixel come vettori normali
- L'immagine prodotta dallo shading fa apparire la geometria molto più complessa di quello che è.

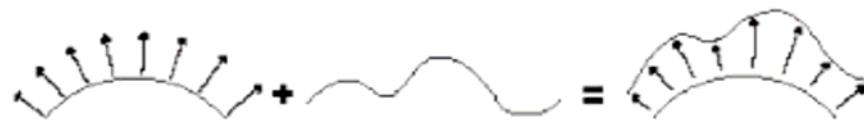


Dopo il bump
mapping



Bump Mapping

Offset surface position



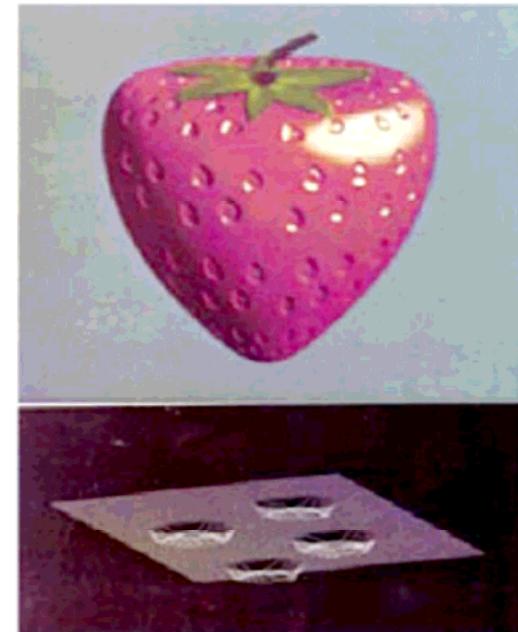
Displacement

$$\mathbf{N}(u, v) = \frac{\mathbf{P}_u(u, v) \times \mathbf{P}_v(u, v)}{|\mathbf{P}_u(u, v) \times \mathbf{P}_v(u, v)|}$$

$$\mathbf{P}'(u, v) = \mathbf{P}(u, v) + h(u, v)\mathbf{N}(u, v)$$

Perturb normal

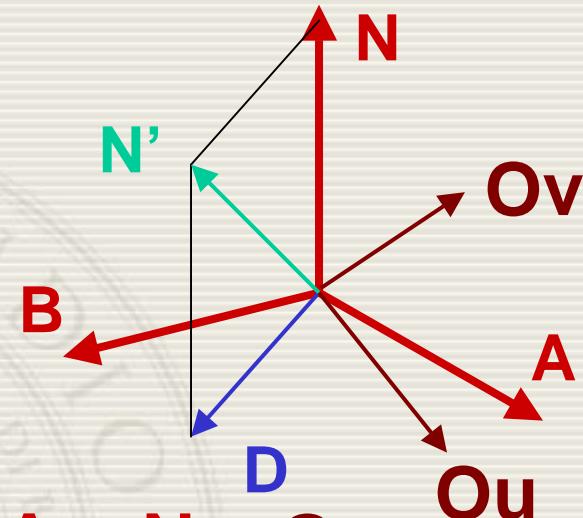
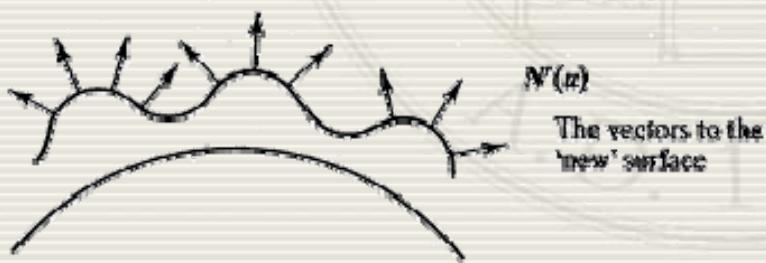
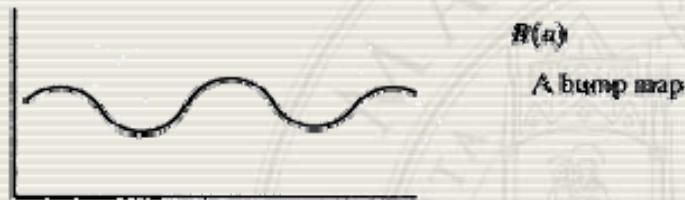
$$\mathbf{N}'(u, v) = \frac{\mathbf{P}'_u(u, v) \times \mathbf{P}'_v(u, v)}{|\mathbf{P}'_u(u, v) \times \mathbf{P}'_v(u, v)|}$$



From Blinn 1978

Formulazione originale di Blinn (1978)

Bump Mapping



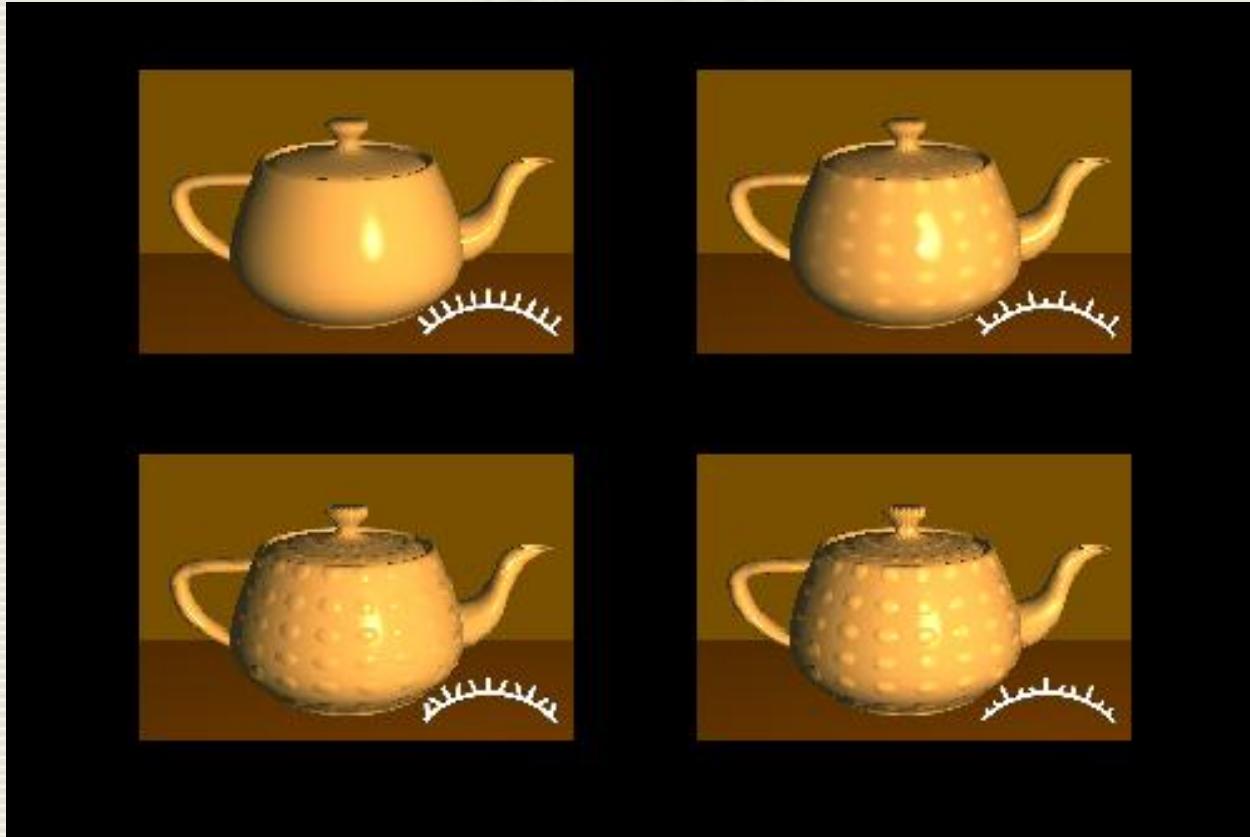
$$N' = N + D$$

dove $D = Bu A - Bv B$

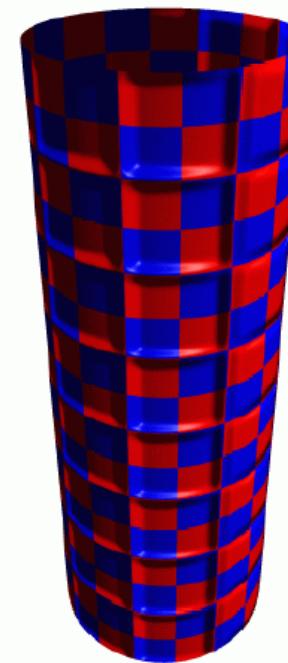
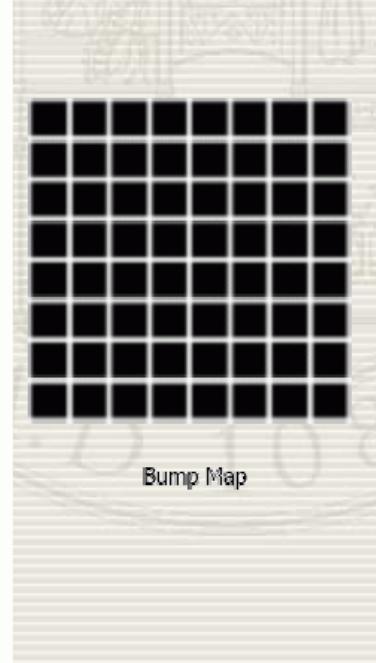
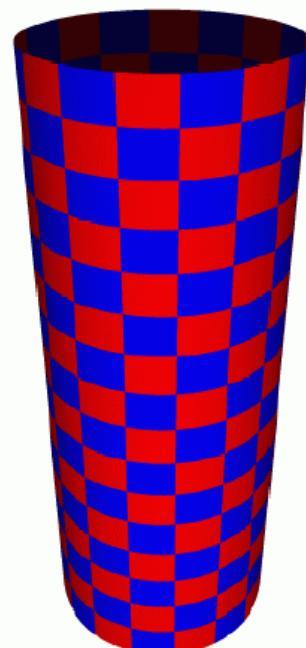
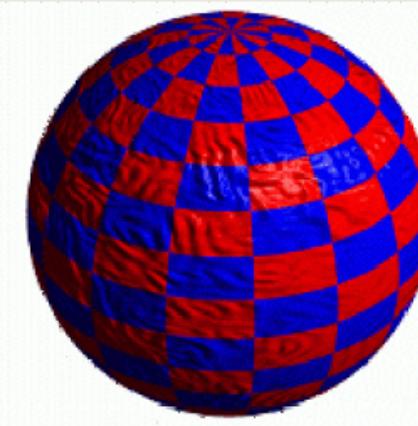
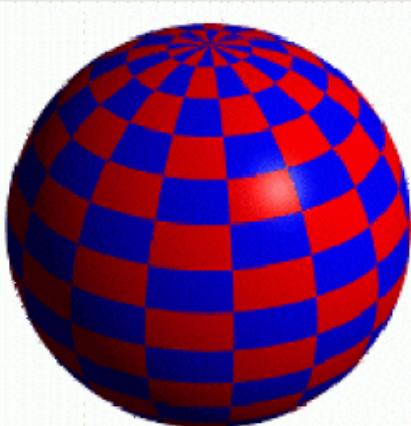
$$Bu = dB(u,v)/du$$

$$Bv = dB(u,v)/dv$$

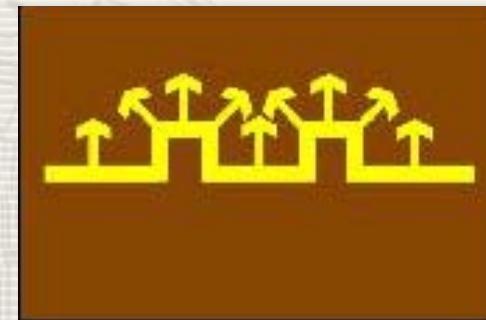
Bump Mapping: esempi



Bump Mapping: esempi

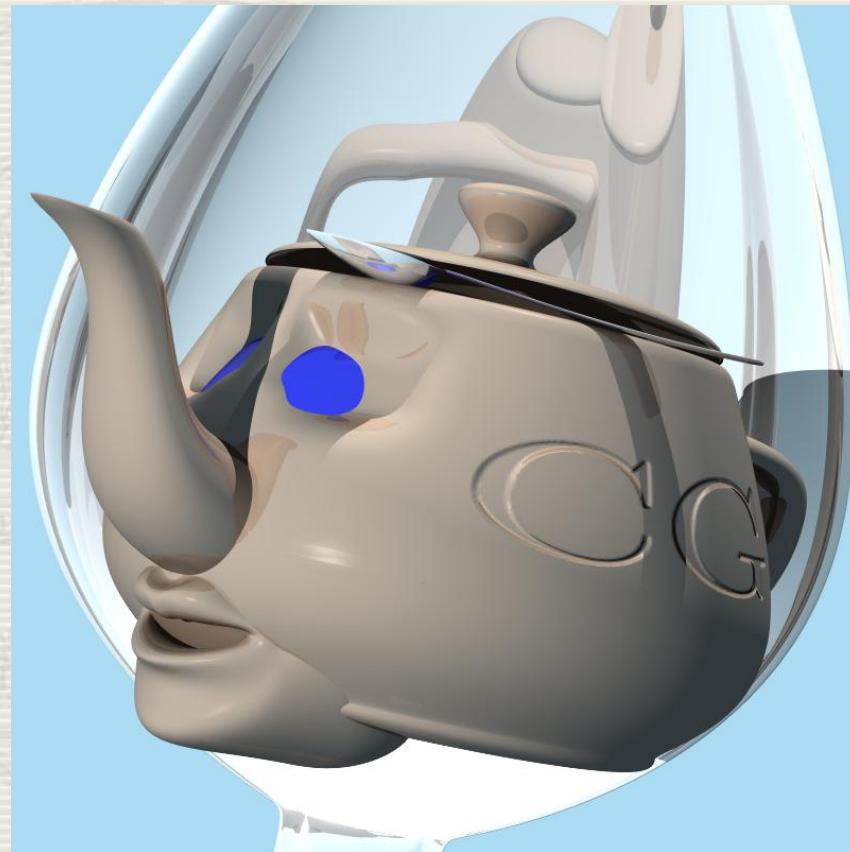


Bump Mapping: esempio



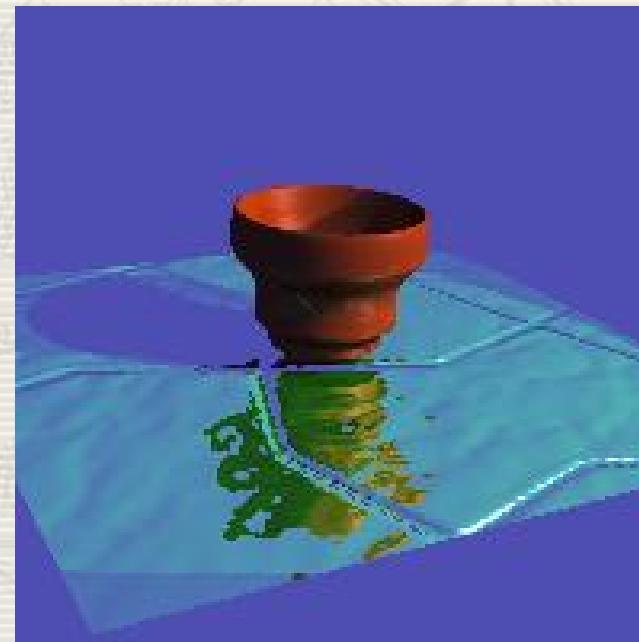
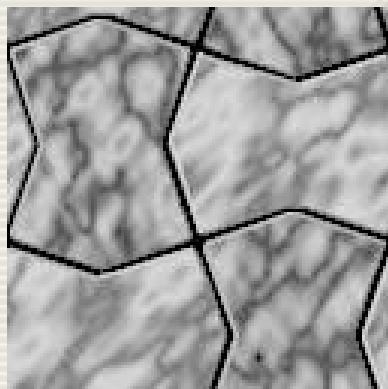
Bump Mapping: esempio

CG



Logo Corso Grafica A.A.2005/06

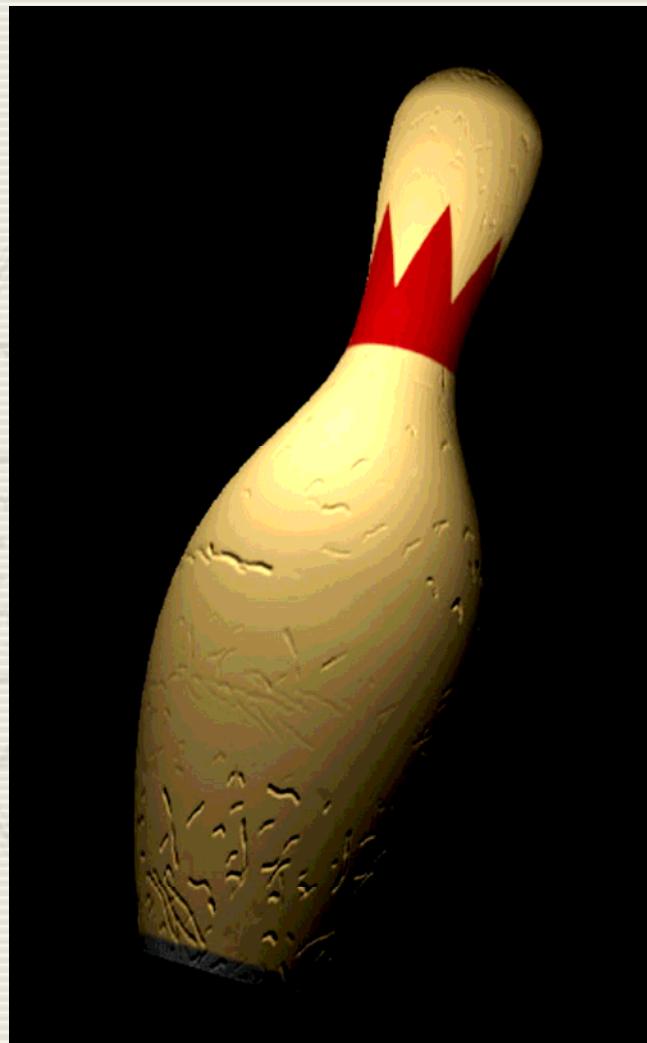
Bump Mapping: esempio



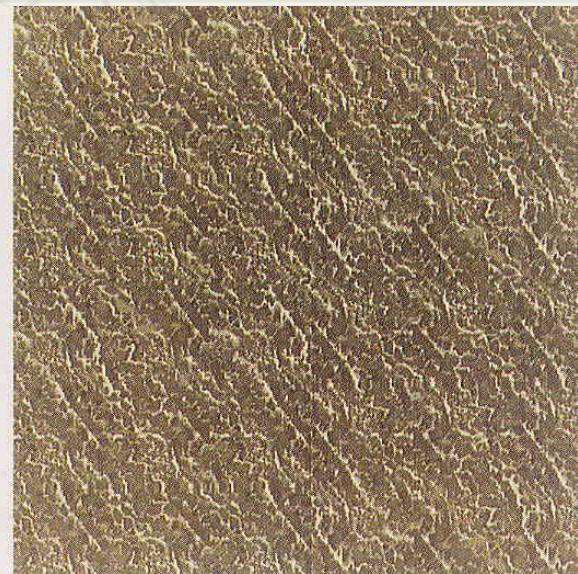
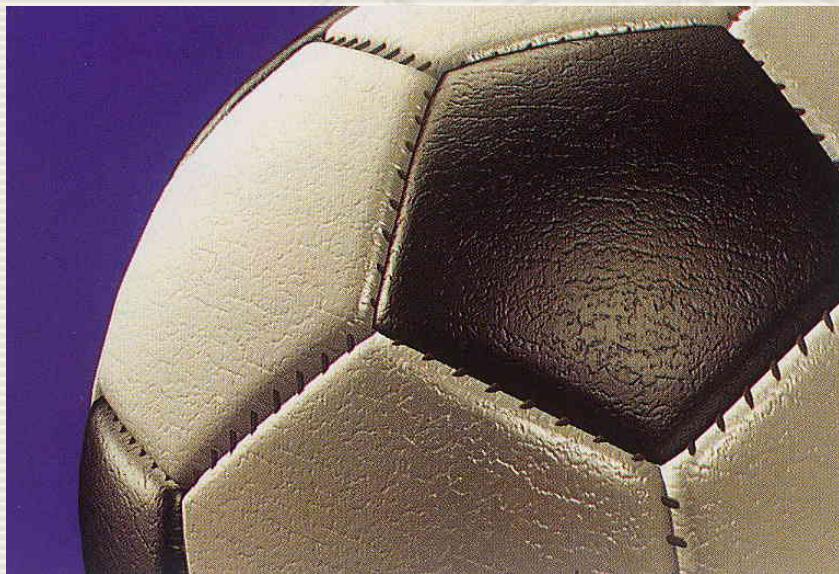
Bump Mapping: esempio



Bump Mapping: esempio



Bump Mapping



Multitexturing

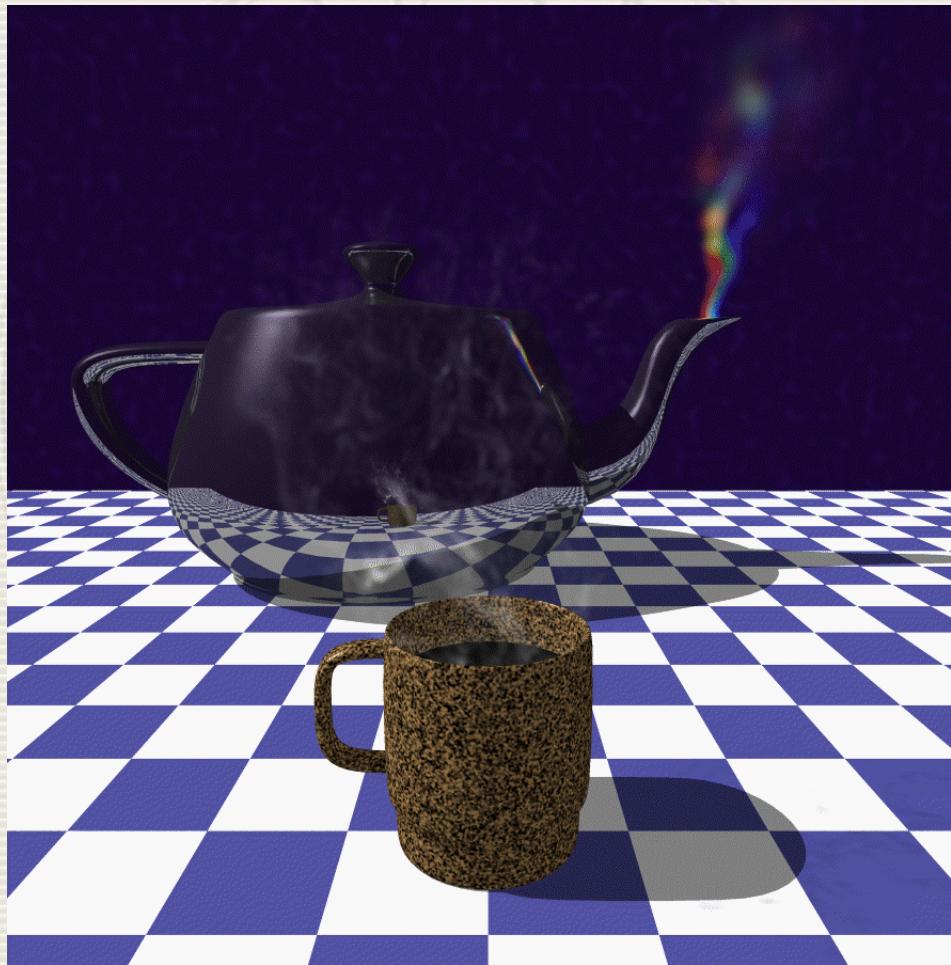


Drops

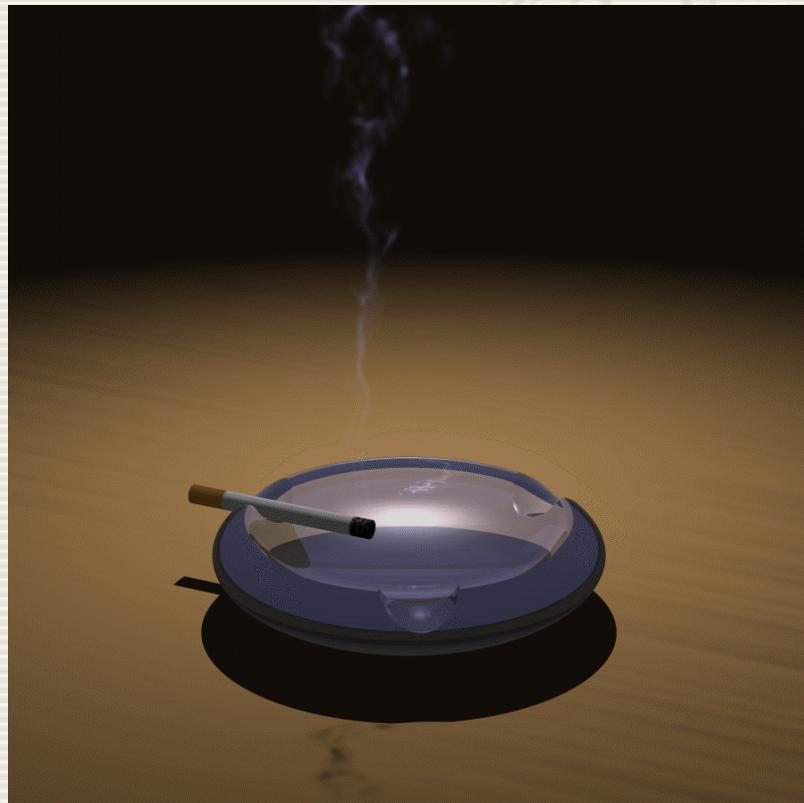


Texture Mapping 2D + Bump Mapping

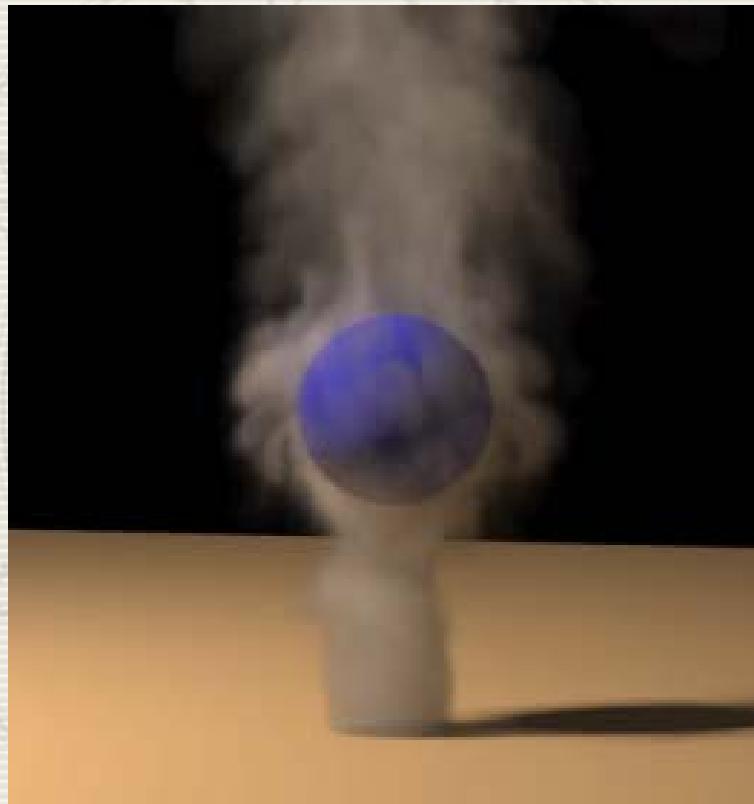
Smoke Simulation: esempio



Smoke Simulation: esempi



Smoke Simulation: esempio



Si noti come il fumo si muove correttamente intorno alla sfera