

Textura

Watt – Capítulos 8 e 16.2

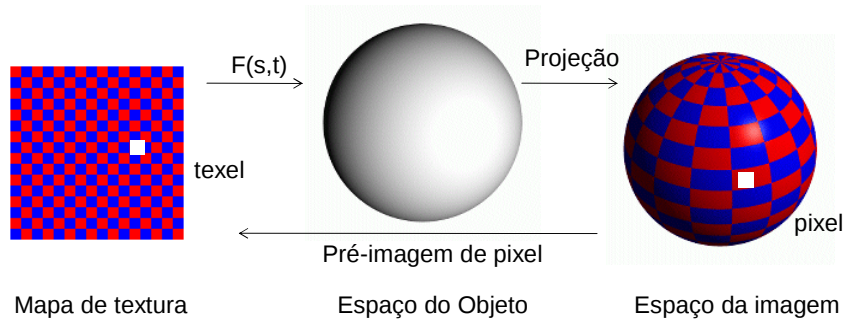
Textura em Síntese de Imagens

- Prover uma forma eficiente de lidar com as diferenças de reflectância difusa ponto-a-ponto em uma superfície (Catmull, 1975).
- Adicionar detalhes a uma superfície sem onerar o processo de renderização.



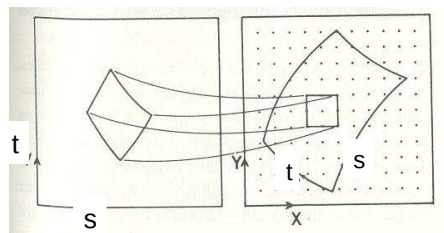
Texturização

- Funções ou imagens que alteram os atributos de uma figura geométrica. Tal função ou imagem é chamada de mapeamento de textura.



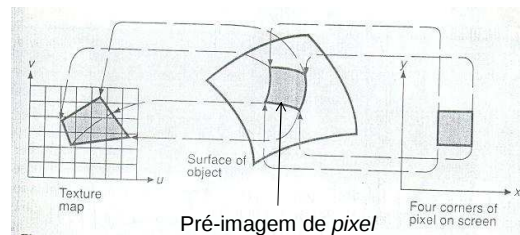
Mapeamento

▪ Método Direto



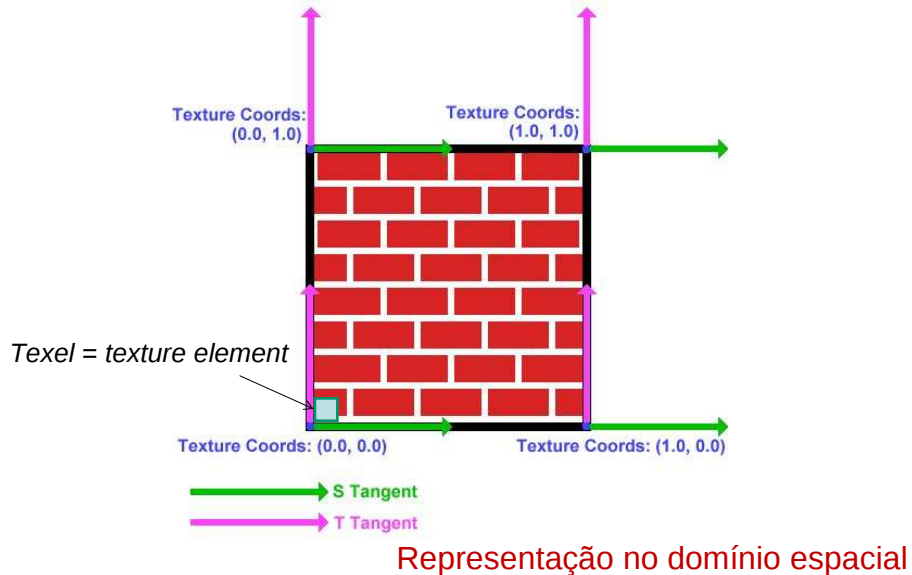
$$x = x(s,t)$$
$$y = y(s,t)$$

▪ Método Inverso



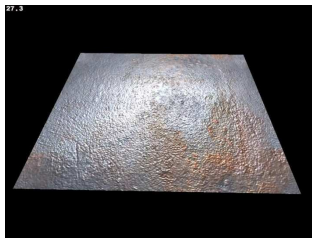
$$s = s(x_s, y_s)$$
$$t = t(x_s, y_s)$$

Espaço de Textura



Atributos “Moduláveis”

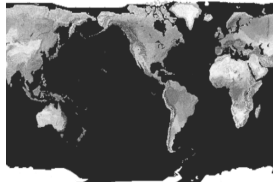
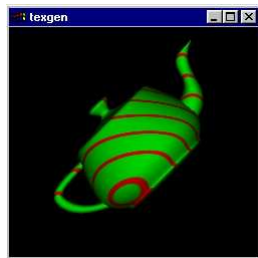
- Componente Difusa da Cor: (R_d, G_d, B_d)
- Componente Especular da Cor: (R_s, G_s, B_s)
- Vetores normais: (n_x, n_y, n_z)
- Posições: (x, y, z)
- Transparência: opacidade α



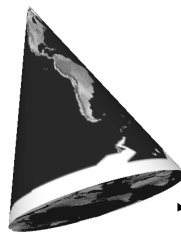
Dimensão de Espaço

Textura 1D

Objeto com textura
1D mapeada

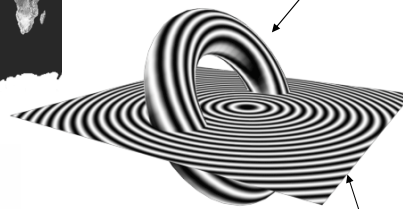


Textura 2D



Objeto com textura
2D mapeada

Objeto com textura
3D mapeada

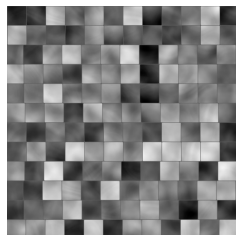


Seção cruzada
da textura 3D

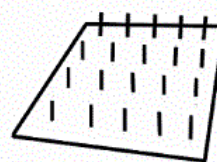
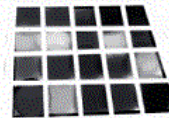
Textura 4D

Texturas 2D

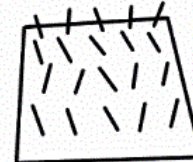
- Utiliza texturas para perturbar a direção do vetor normal de cada ponto da superfície (Blinn, 1978).



Bump
Texture
Map

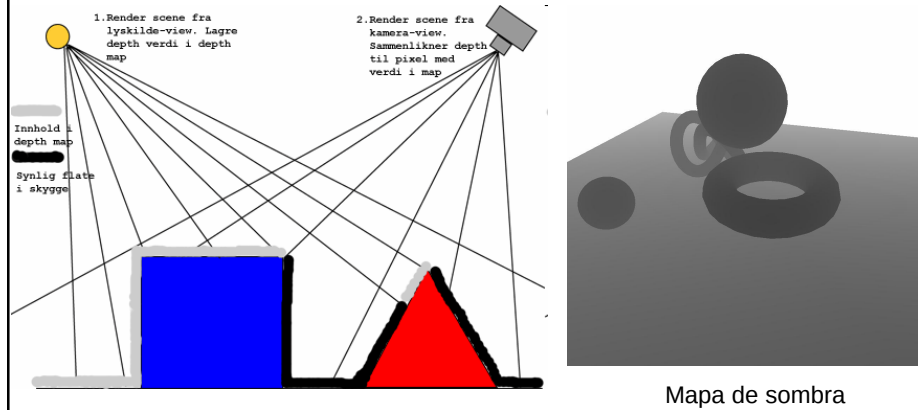


Normais
originais

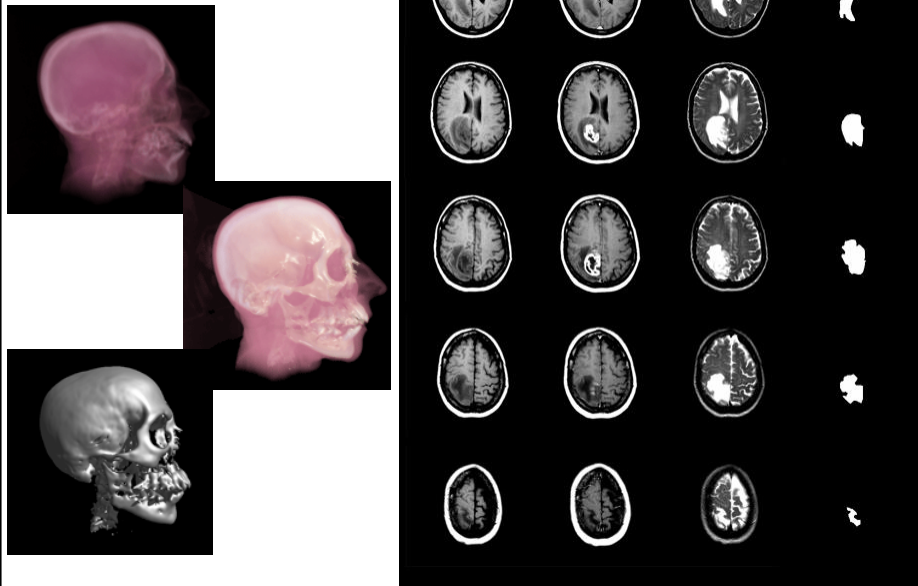


Normais
deslocados pelo
mapa de textura

Texturas 2D



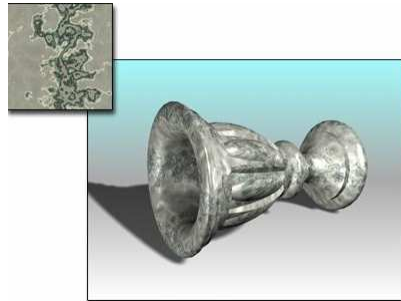
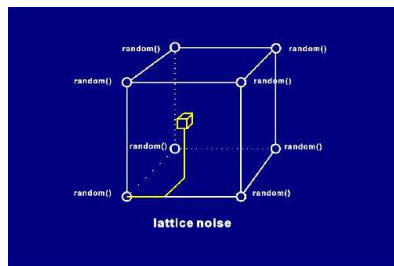
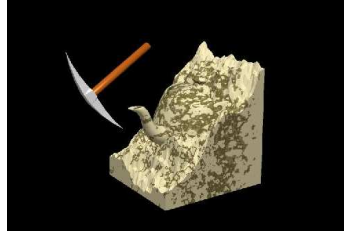
Texturas 3D



Textura Procedural

Perlin

- Utiliza ruídos (de Perlin) para perturbar os atributos de cada fragmento.
- Textura (de ruídos) pré-computada.
- Outros valores por interpolação trilinear.



Função de Ruído de *Perlin*

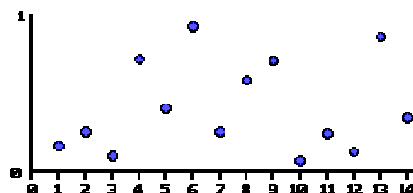
Gerador de valores aleatórios (x,y,z) → ruído

$$n = x + y * 57 + z * 13;$$

$$n = (n < 13) ^ n;$$

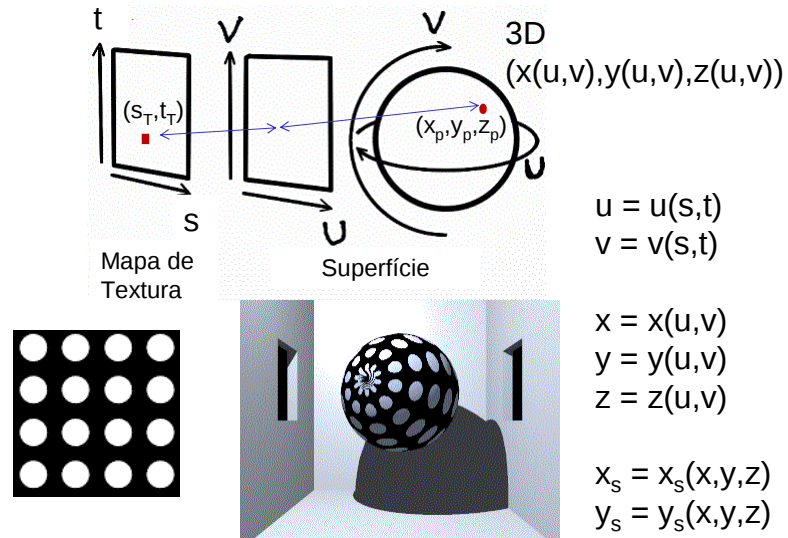
$$\text{noise} = 1.0 -$$

$$((n * (n * 15731 + 789221) + 1376312589) \& 0x7fffffff) / 1073741824.0);$$



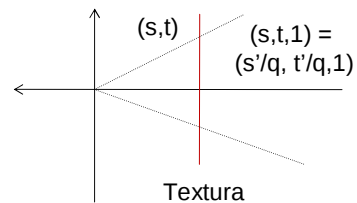
<http://cs.nyu.edu/~perlin/doc/oscar.html>

Mapeamento Método Direto



Texturização de Malhas Poligonais Mapeamento direto

$$\begin{bmatrix} x' \\ y' \\ w \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} s' \\ t' \\ q \end{bmatrix}$$



$$(x, y) = (x' / w, y' / w)$$

$$(s, t) = (s' / q, t' / q)$$

$$i = 1$$

Texturização de Malhas Poligonais

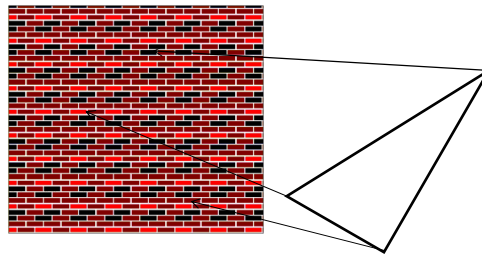
Mapeamento Inverso

$$\begin{bmatrix} s' \\ t' \\ q \end{bmatrix} = \begin{bmatrix} ei - fh & ch - bi & bf - ce \\ fg - di & ai - cg & cd - af \\ dh - eg & bg - ah & ae - bd \end{bmatrix} \begin{bmatrix} x' \\ y' \\ w \end{bmatrix}$$

$$(x, y) = (x' / w, y' / w)$$

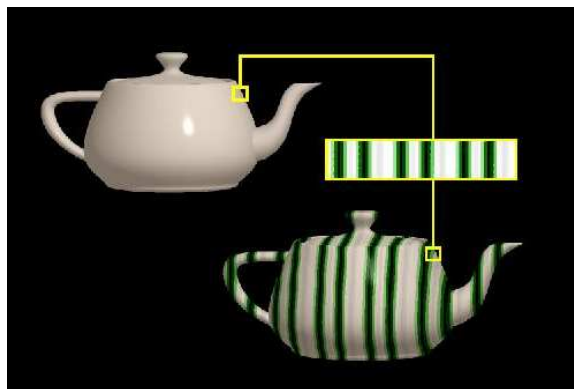
$$(s, t) = (s' / q, t' / q)$$

$$i = 1$$



Mapeamento

Método Inverso



Texturização Modo Inverso

Funções de projeção: derivar coordenadas (s,t) a partir das coordenadas do modelo geométrico (x,y)

Modo de Correspondência: modo de tratamento quando as coordenadas (s,t) estiverem fora do domínio

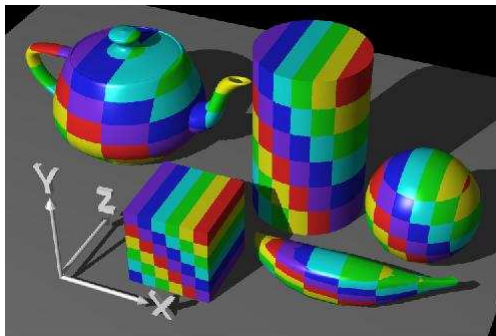
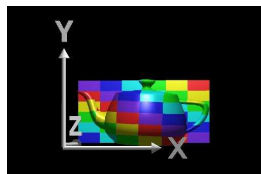
Aplicação da textura sobre fragmentos: modo de aplicação dos valores contidos no texel (s,t) sobre os valores do fragmento (x,y)

Mapeamento Linear Método Inverso

- A textura 2D é projetada ortogonalmente em uma dada direção.

Direção z: $s=x$

$t=y$



Mapeamento Esférico

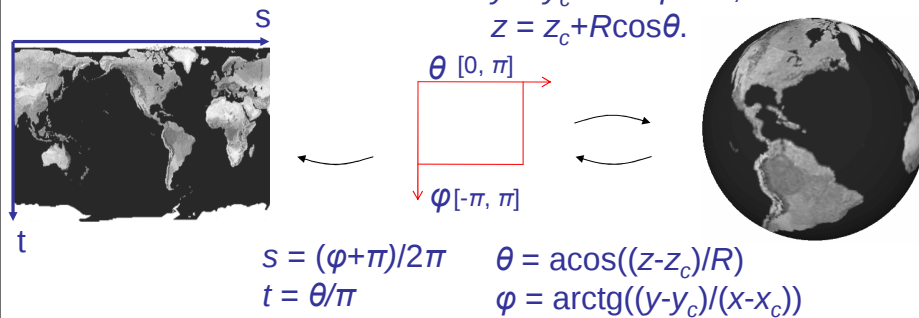
Método Inverso

- Mapeamento de $(u,v) \in [0,1]^2$ sobre a longitude e latitude de uma esfera.
- Para uma esfera de raio R com centro em (c_x, c_y, c_z) , a equação paramétrica dessa esfera é:

$$x = x_c + R \cos \varphi \sin \theta,$$

$$y = y_c + R \sin \varphi \sin \theta,$$

$$z = z_c + R \cos \theta.$$



Modo de Correspondência



Repetição

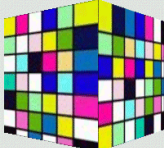
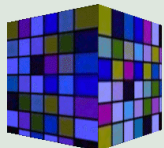
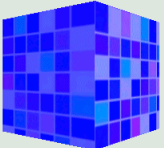


Supressão/clamp

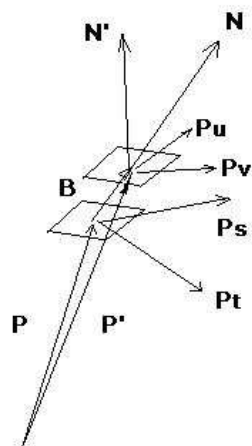


Aplicação de *Texels* sobre Fragmentos

Mode	Result color	Result transparency
REPLACE	T_{rgb}	T_a
DECAL	$S_{rgb} * (1 - T_a) + T_{rgb} * T_a$	S_a
MODULATE	$S_{rgb} * T_{rgb}$	$S_a * T_a$
BLEND	$S_{rgb} * (1 - T_{rgb}) + B_{rgb} * T_{rgb}$	$S_a * T_a$

Modelo	Replace/Decal	Modulate	Blend
			
Textura			

Bump mapping (Blinn)



$$\vec{N} = \vec{P}_u \times \vec{P}_v$$

$$\vec{P}'(u, v) = \vec{P}_u + B(u, v) \vec{N}$$

$$\vec{N}' = \vec{P}'_u \times \vec{P}'_v$$

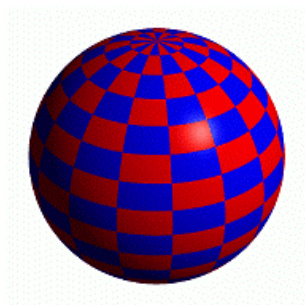
$$\vec{P}'_u = \vec{P}_u + B_u \vec{N} + B \vec{N}_u$$

$$\vec{P}'_v = \vec{P}_v + B_v \vec{N} + B \vec{N}_v$$

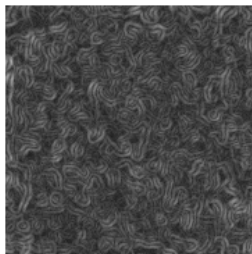
$$\vec{N}' = \vec{N} + B_u \vec{N} \times \vec{P}_v + B_v \vec{P}_u \times \vec{N} = \vec{N} + B_u \vec{N} \times \vec{P}_v - B_v \vec{N} \times \vec{P}_u$$

$B \ll 0$

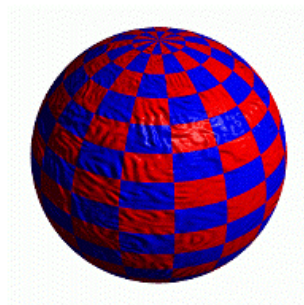
Bump mapping Exemplo



Esfera com textura difusa

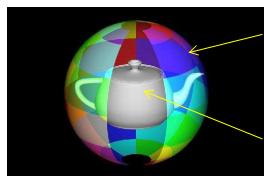


Bump map



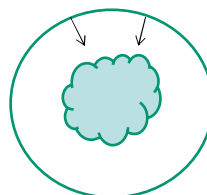
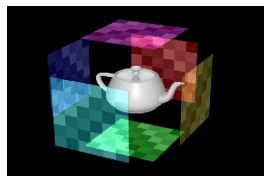
Esfera com textura difusa
e bump mapping

Mapeamento em 2 passos

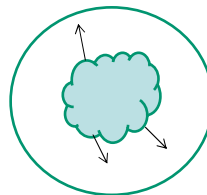


Superfície
intermediária

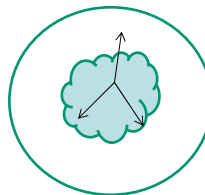
Superfície de
interesse



Na direção da normal da
superfície intermediária

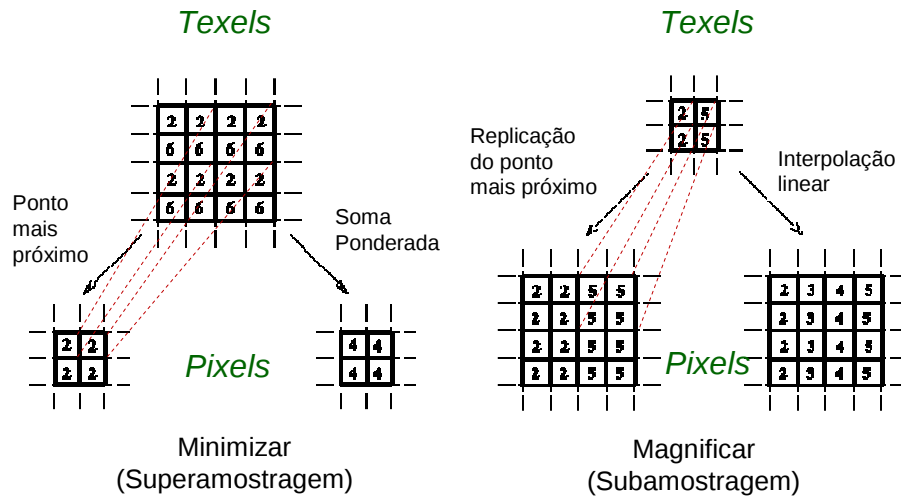


Na direção da normal da
superfície de interesse



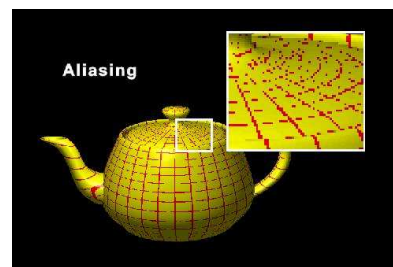
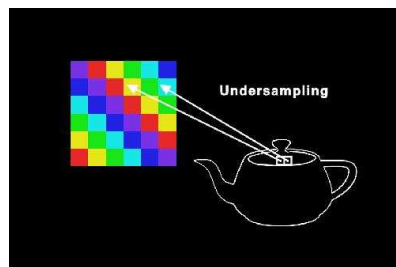
A partir do centro
do objeto de
interesse

Subamostragem e Superamostragem



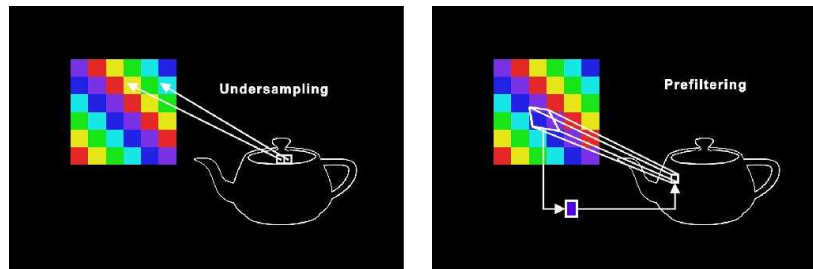
Aliasing

- Resultado da subamostragem da textura.
 - Em mapeamento de textura, *aliasing* ocorre quando dois *pixels* adjacentes do objeto são mapeados em *texels* diferentes, não-adjacentes, do mapa de textura.



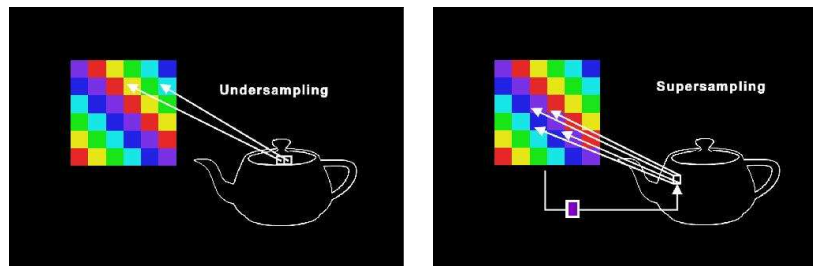
Redução de *Aliasing*

- **Pré-filtragem** (Catmull, 1978): Cada *pixel* do objeto é tratado como uma área. A área do *pixel* é mapeada na textura. A cor média é calculada para os *texels* contidos nesta área.

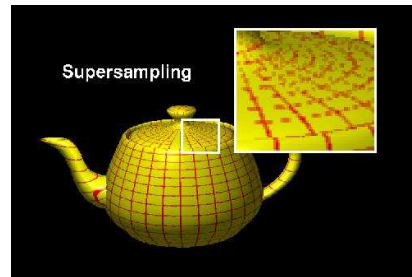
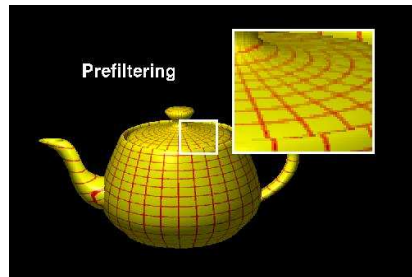


Redução de *Aliasing*

- **Superamostragem** (Crow, 1981): Também calcula uma cor média. No exemplo abaixo, cada canto da área do pixel é mapeado na textura. A média dos valores obtidos produz a cor final do objeto.

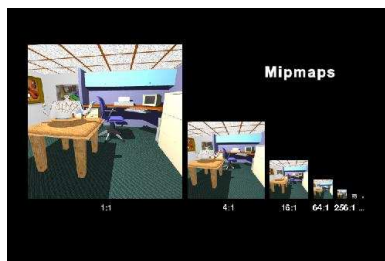


Redução de *Aliasing* Exemplos



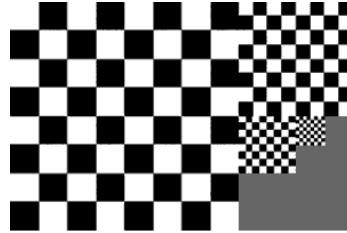
Redução de *Aliasing*

- MIP (*Multum In Parvo* – muito em pouco espaço).
- Método de aceleração do cálculo da cor média para áreas de amostragem da textura (Williams, 1983).
 - Várias versões da textura são criadas (*mipmaps*).
 - Cada *texel* de uma versão contém a cor média de 4 *texels* da versão **anterior**.



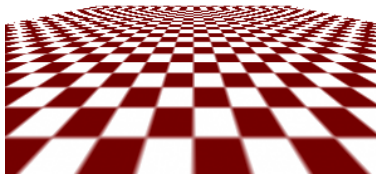
Redução de *Aliasing*

- Superamostragem

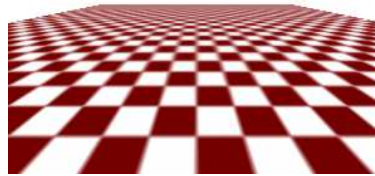


This simple MIP map holds four copies of a chequered texture, all at different scales

Efeito de Moiré



Sem mipmapping



Com mipmapping