

# Texturas

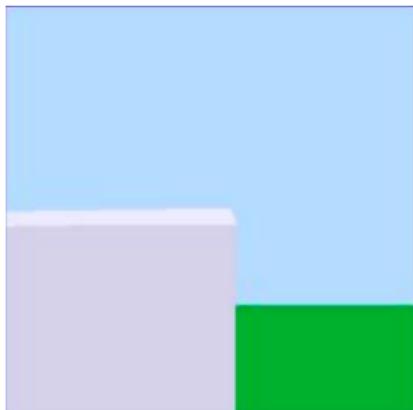
# Texturas

- ▶ Em alguns cenários, apenas malhas e cores não são suficientes para oferecer um nível de detalhe mais realista.
- ▶ Não é razoável representar todos os detalhes apenas com geometria, vértices, e cores.
- ▶ Texturas ajudam nesse processo.
- ▶ Podemos mapear uma imagem na superfície de nosso objeto 3D.



# Texturas

Exemplo<sup>1</sup>:



Cenário com 8 polígonos



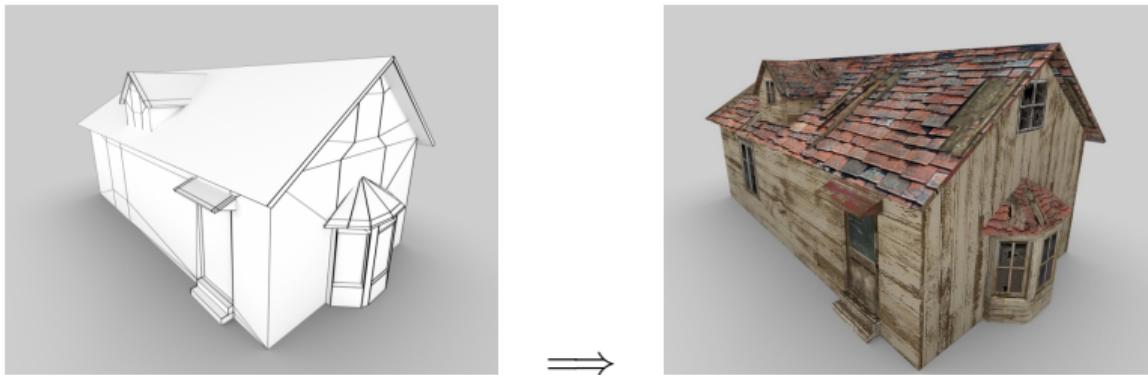
Aplicação de Texturas

---

<sup>1</sup>[http://www.siggraph.org/education/materials/HyperGraph/mapping/r\\_wolfe/r\\_wolfe\\_mapping\\_1.htm](http://www.siggraph.org/education/materials/HyperGraph/mapping/r_wolfe/r_wolfe_mapping_1.htm)

# Texturas

Exemplo<sup>2</sup>:



---

<sup>2</sup><https://free3d.com/pt/3d-model/abandoned-cottage-house-825251.html>

# Texturas

Texturização:

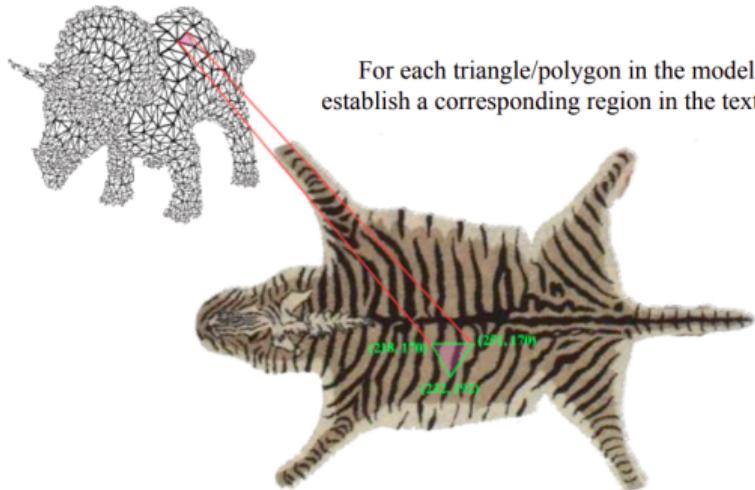
- ▶ Prover uma forma eficiente de lidar com as diferenças de reflectância difusa ponto-a-ponto em uma superfície.
- ▶ Mais eficiente do que tentar reproduzir o mesmo efeito apenas com geometria.



# Mapeamento de Texturas

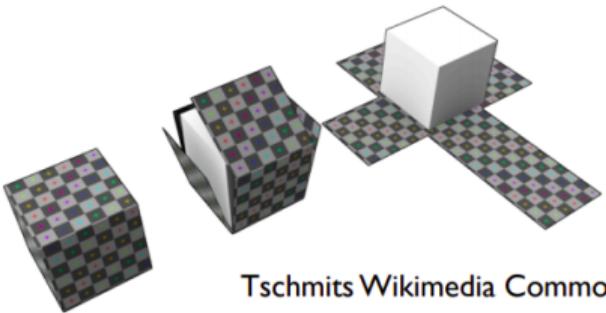
# Mapeamento de Texturas

Imagen (texturas) mapeada na superfície (triângulos) do modelo 3D.



# Mapeamento de Texturas

O arquivo/imagem de textura é gerado de forma apropriada para o modelo 3D.

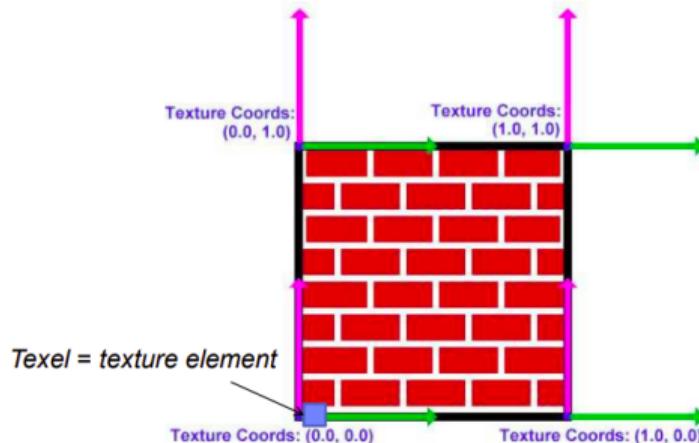


Tschmits Wikimedia Commons

Não estudaremos como gerar o arquivo de textura, mas conceitos do mapeamento entre texturas e modelos.

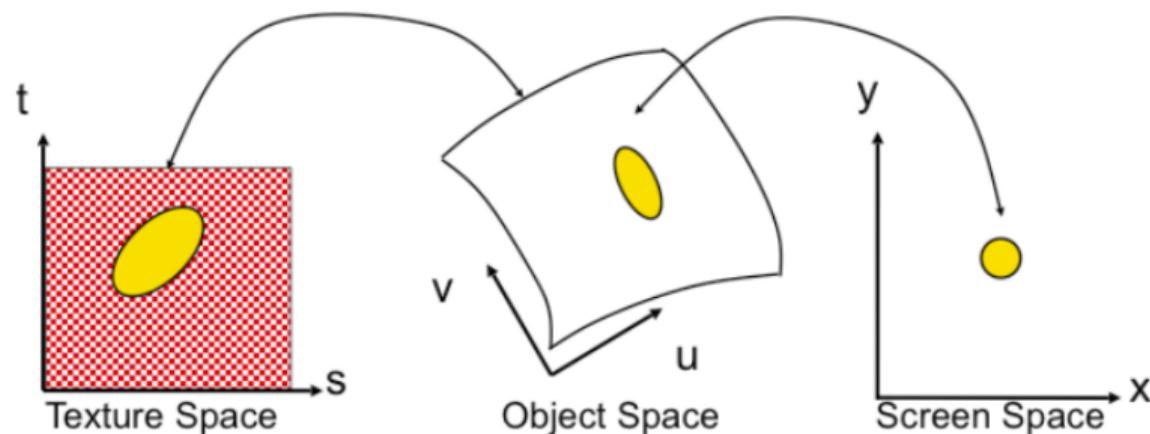
# Conceitos Básicos

- ▶ A textura é um arranjo 2D de tamanho  $s \times t$ .
- ▶ Cada elemento deste arranjo é chamado de texel (*texture element*).
- ▶ As coordenadas de texturas são definidas no intervalo  $[0, 1]$ .



# Conceitos Básicos

- ▶ Problema: nosso modelo é uma malha (triangular) 3D.
- ▶ As texturas são mapas/arranjos 2D.
- ▶ Como mapeá-los?

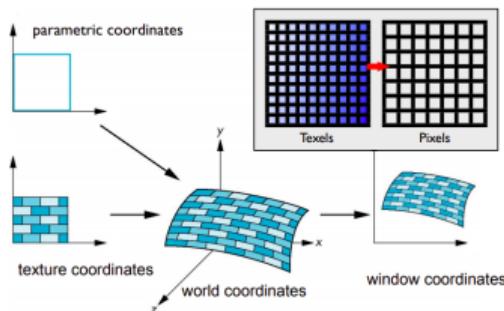


# Primeiros Passos

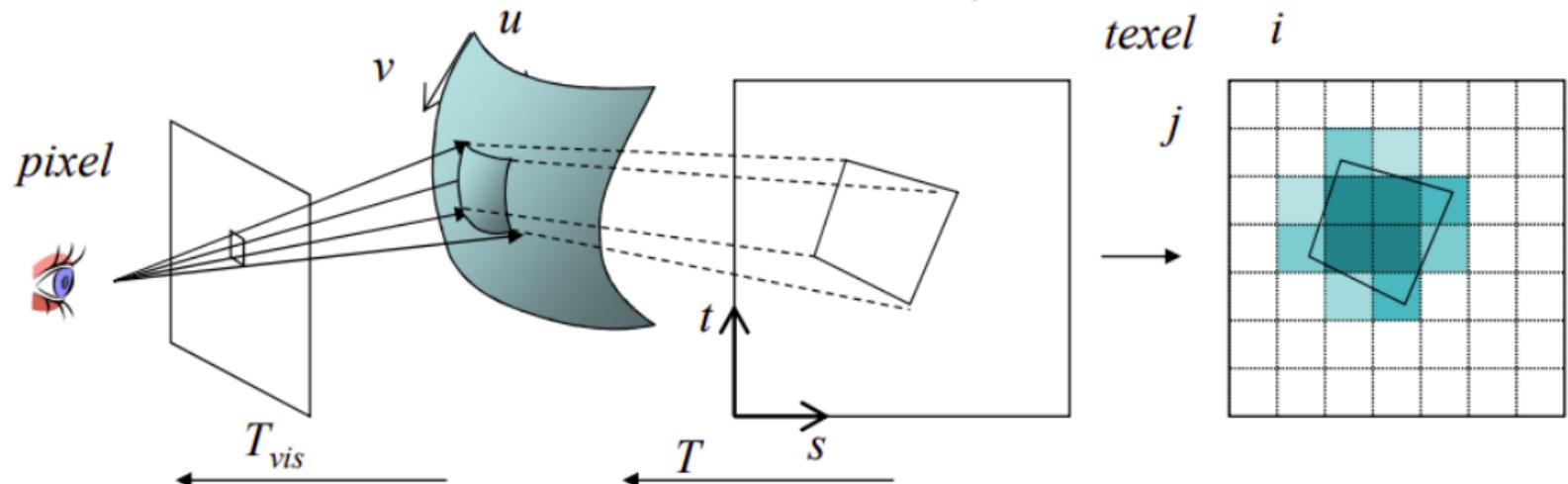
1. Dado um pixel (espaço *clip/tela*), precisamos identificar qual o modelo corresponde.
2. Dado um ponto do modelo, queremos saber a qual ponto da textura corresponde.

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

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

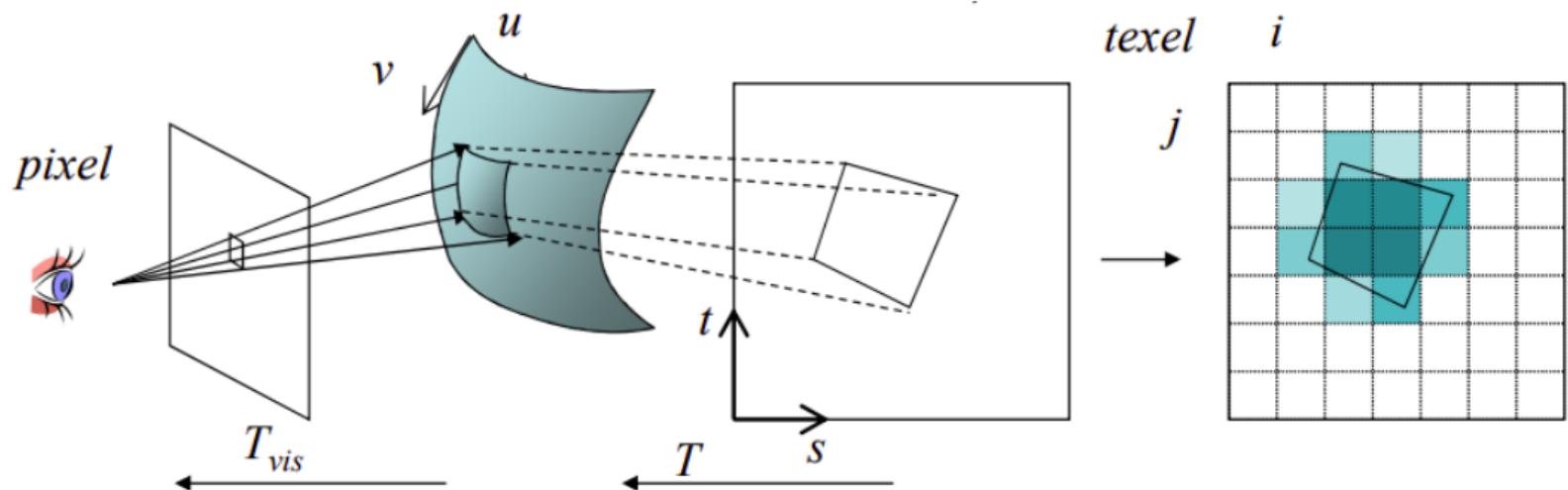


# Processo de Mapeamento de Texturas



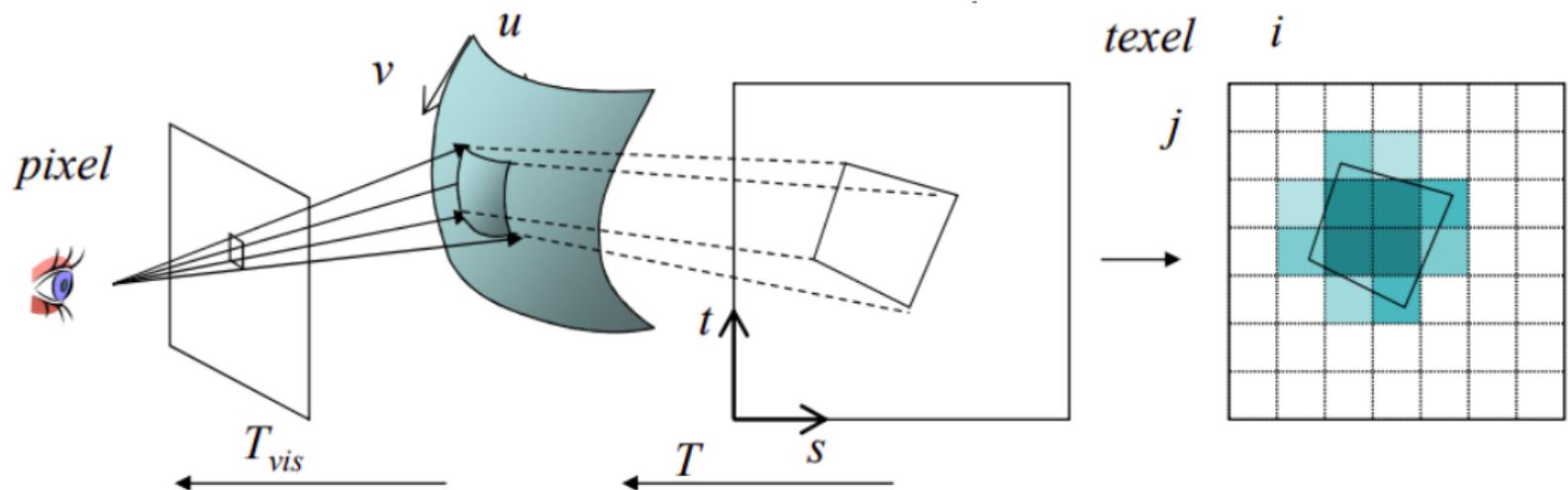
Projeção do pixel sobre a superfície: pontos da superfície correspondentes aos vértices do pixel.

# Processo de Mapeamento de Texturas



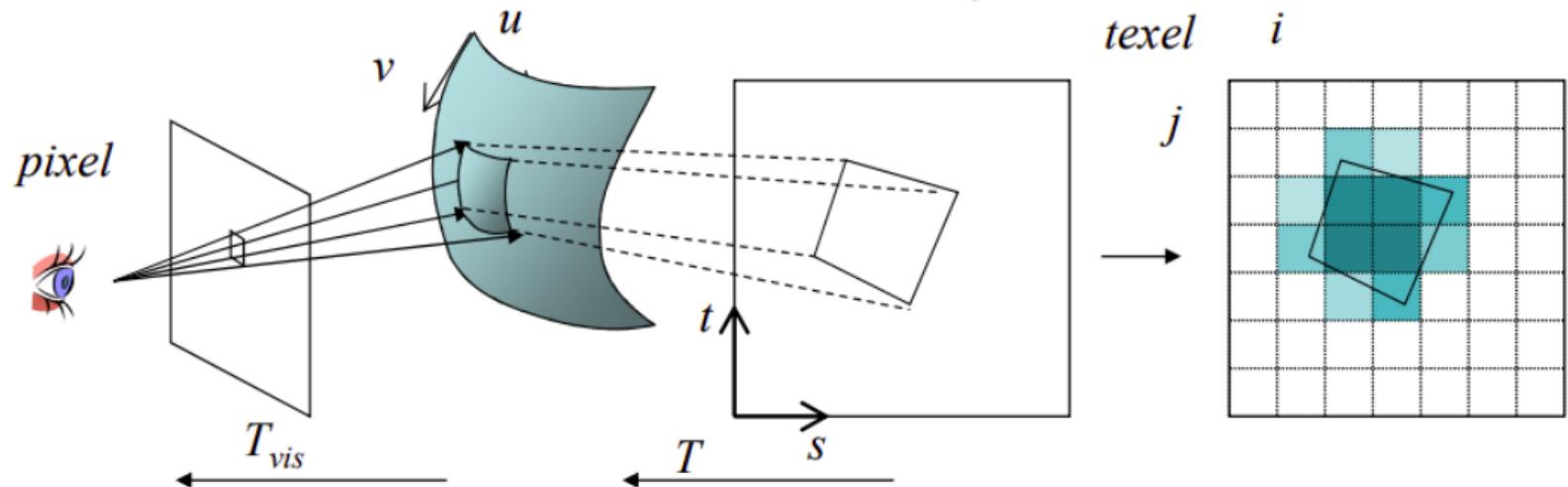
Parametrização: coordenadas paramétricas dos vértices do pixel projetados

# Processo de Mapeamento de Texturas



Mapeamento inverso: coordenadas dos vértices no espaço de textura.

# Processo de Mapeamento de Texturas

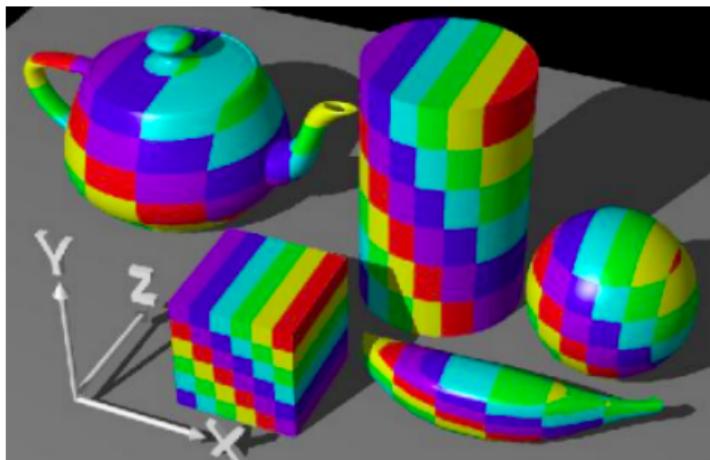
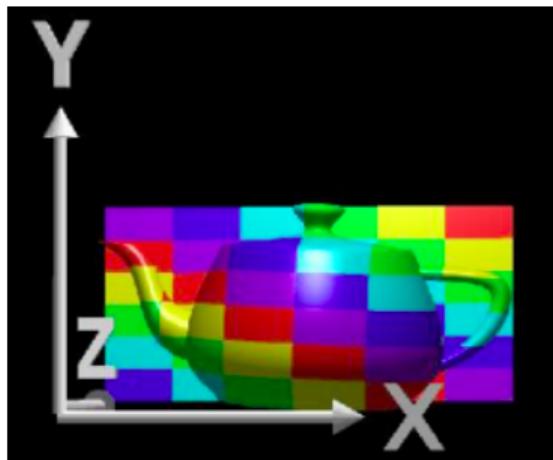


Média: cor média dos "texeis" proporcional à área coberta pelo polígono. Cada texel pode ocupar vários pixels ou vice-versa.

# Tipos de Mapeamento

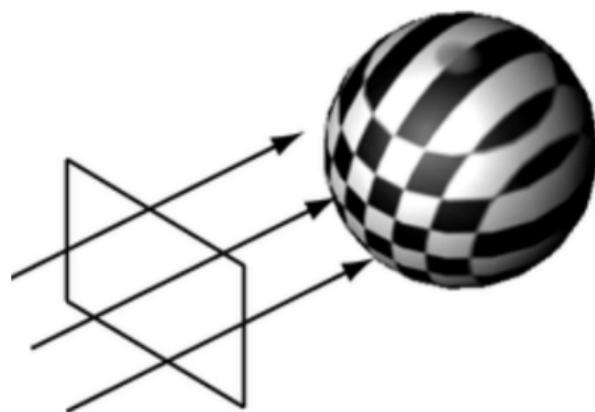
# Tipos de Mapeamento

Planar: coordenadas  $uv$  mapeadas ortogonalmente.



# Tipos de Mapeamento

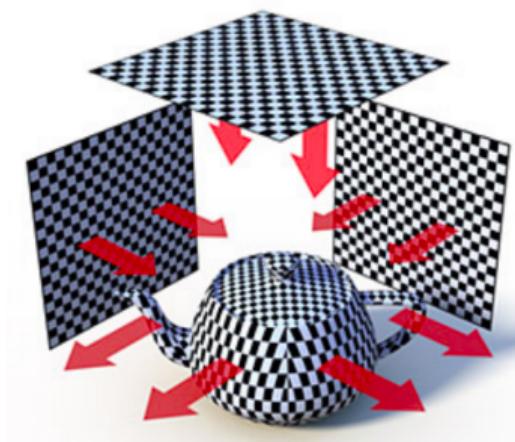
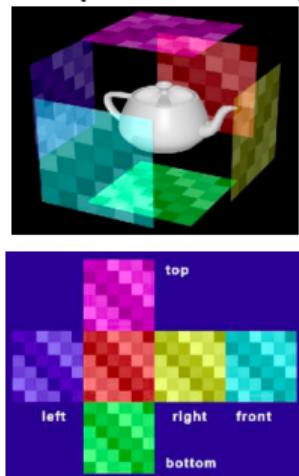
Planar: coordenadas  $uv$  mapeadas ortogonalmente.



Simples, mas apresenta efeito ruim em normais perpendiculares.

# Tipos de Mapeamento

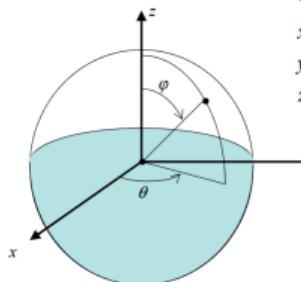
Cúbico: coordenadas *uv* mapeadas ortogonalmente nos seis planos de um cubo.



# Tipos de Mapeamento

Esférico: coordenadas  $uv$  são mapeadas segundo coordenadas polares esféricas.

## Parametrização da Esfera



Função de mapeamento

$$\begin{aligned}x(\varphi, \theta) &= \sin \varphi \cos \theta & \varphi &= \pi \cdot t \\y(\varphi, \theta) &= \sin \varphi \sin \theta & \theta &= 2\pi \cdot s \\z(\varphi, \theta) &= \cos \varphi\end{aligned}$$

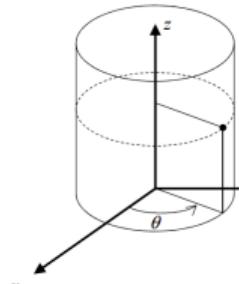
Função de mapeamento inversa

$$\begin{aligned}\varphi &= \arccos z & t &= \frac{\arccos z}{\pi} \\ \theta &= \arctan \frac{y}{x} & s &= \frac{\arctan \frac{y}{x}}{2\pi}\end{aligned}$$

# Tipos de Mapeamento

Cilíndrico: coordenadas  $uv$  são mapeadas segundo coordenadas polares cilíndricas.

## Parametrização do Cilindro



*Função de mapeamento*

$$\begin{aligned}x &= \cos \theta & \theta &= 2\pi \cdot s \\y &= \sin \theta & z &= t \\z &= z\end{aligned}$$

*Função de mapeamento inversa*

$$\begin{aligned}\theta &= \arctan \frac{y}{x} & s &= \frac{\theta}{2\pi} \\z &= z & t &= z\end{aligned}$$

# Mapeamento em Duas Fases

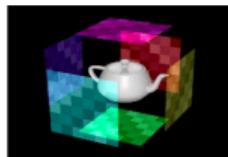
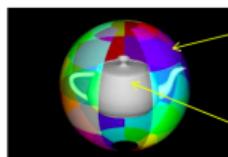
Modelos "complexos" podem ser englobados por uma superfície simples de um modelo intermediário (cubo, esfera, cilindro):

1. A textura é aplicada na superfície simples do modelo intermediário.
2. A textura é mapeada do objeto intermediário para o objeto alvo.

# Mapeamento em Duas Fases

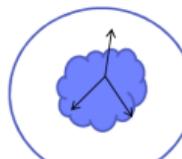
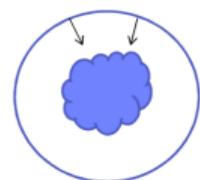
Modelos "complexos" podem ser englobados por uma superfície simples de um modelo intermediário (cubo, esfera, cilindro):

Fase 1: Escolher  
um modelo/super-  
fície  
intermediário.

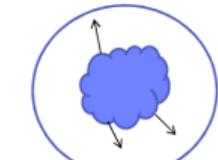


Superfície  
intermediária  
Superfície de  
interesse

Na direção da normal da  
superfície intermediária



A partir do centro  
do objeto de  
interesse



Na direção da normal da  
superfície de interesse

Fase 2: Escolher  
uma estratégia de  
mapeamento.

# Informações de Texturas no *WaveFront*

```
# Blender3D v249 OBJ File
v 1.000000 -1.000000 -1.000000
v 1.000000 -1.000000 1.000000
v -1.000000 -1.000000 1.000000
v -1.000000 -1.000000 -1.000000
v 1.000000 1.000000 -1.000000
v 0.999999 1.000000 1.000001
v -1.000000 1.000000 1.000000
v -1.000000 1.000000 -1.000000
vt 0.746573 0.750412
vt 0.749279 0.501284
vt 0.999110 0.501077
vt 0.999455 0.750380
vt 0.250471 0.500702
vt 0.249682 0.749577
vt 0.001085 0.750380
f 5/1/1 1/2/1 4/3/1
f 5/1/1 4/3/1 8/4/1
f 3/5/2 7/6/2 8/7/2
```

O caractere inicial de cada linha indica a função:

**#** Comentários

**v** Vértices

**vn** Normal

**vt** Coordenadas de textura

**f** Faces

Cada face é um triângulo.

# Informações de Texturas no *WaveFront*

```
# Blender3D v249 OBJ File
v 1.000000 -1.000000 -1.000000
v 1.000000 -1.000000 1.000000
v -1.000000 -1.000000 1.000000
v -1.000000 -1.000000 -1.000000
v 1.000000 1.000000 -1.000000
v 0.999999 1.000000 1.000001
v -1.000000 1.000000 1.000000
v -1.000000 1.000000 -1.000000
vt 0.746573 0.750412
vt 0.749279 0.501284
vt 0.999110 0.501077
vt 0.999455 0.750380
vt 0.250471 0.500702
vt 0.249682 0.749577
vt 0.001085 0.750380
f 5/1/1 1/2/1 4/3/1
f 5/1/1 4/3/1 8/4/1
f 3/5/2 7/6/2 8/7/2
```

Cada triângulo é formato por três vértices, representados por v/vt/vn. Exemplo: 5/1/1:

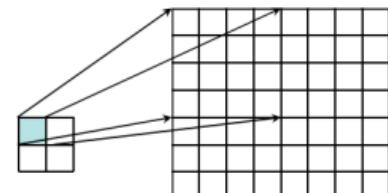
- ▶ A primeira parte indica o vértice na posição cinco (v 1.000000 1.000000 -1.000000).
- ▶ A segunda parte indica a coordenada da textura (vt 0.748573 0.750412).

# Detalhes do Mapeamento

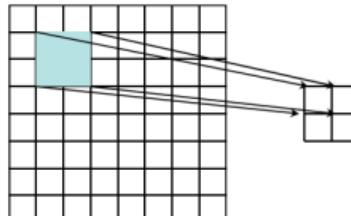
Pixeis não são texéis:

**Magnificação** Quando o polígono é maior que a textura.

**Minificação** Quando a textura é maior que o polígono.



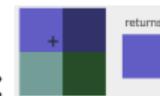
Textura  
Magnificação



Textura  
Minificação

# Detalhes do Mapeamento

Nesses casos devemos aplicar filtros:



**NEAREST** Escolhe o texel mais próximo da coordenada de textura:



**LINEAR** Escolhe o texel interpolando os vizinhos e usa o valor aproximado:



**NEAREST×LINEAR**

# Detalhes do Mapeamento

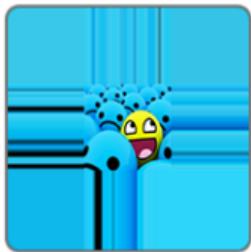
- ▶ Repetição de texturas: quando uma mesma textura é repetida várias vezes no polígono.
- ▶ Padrões de repetição:
  - REPEAT** Repetição simples.
  - CLAMP** "Estica" o último texel.



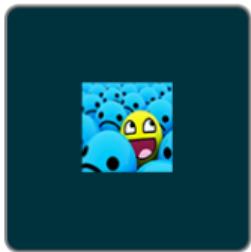
GL\_REPEAT



GL\_MIRRORED\_REPEAT



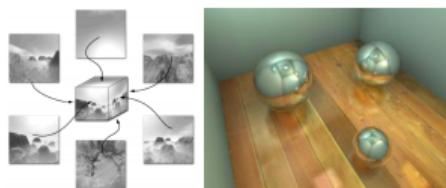
GL\_CLAMP\_TO\_EDGE



GL\_CLAMP\_TO\_BORDER

# Outros Tipos de Mapeamento

Mapeamento de Reflexão: reflete na superfície dos objetos os elementos que compõem a cena.



Mapeamento *Bump*: técnica de perturbação para dar efeito de superfície áspera.



# Material de base para a aula

- ▶ Tamar Shinar (2012). CS230 : Computer Graphics. Lecture 7: Texture Mapping. Computer Science & Engineering. UC Riverside.
- ▶ Stanford CS348V (2018). Lecture 16. Texturing mapping algorithms and hardware. Visual Computing Systems Course. Stanford University.
- ▶ Claudio Esperança e Paulo Roma Cavalcanti (2011). Introdução à Computação Gráfica Texturas. UFRJ.
- ▶ ANGEL, Edward et al (2012). Interactive computer graphics: a top-down approach with shader-based OpenGL. Boston: Addison-Wesley.
- ▶ Computação Gráfica: Aula 09. Slides de Ricardo M. Marcacini. Disciplina SCC0250/0650, ICMC/USP, 2021.