



**UFABC**

**Computação Gráfica**

**André Brandão**

# Aula 17

Aula prática  
Cubo com textura

# Aula 16

- Trabalhamos com cubo colorido.

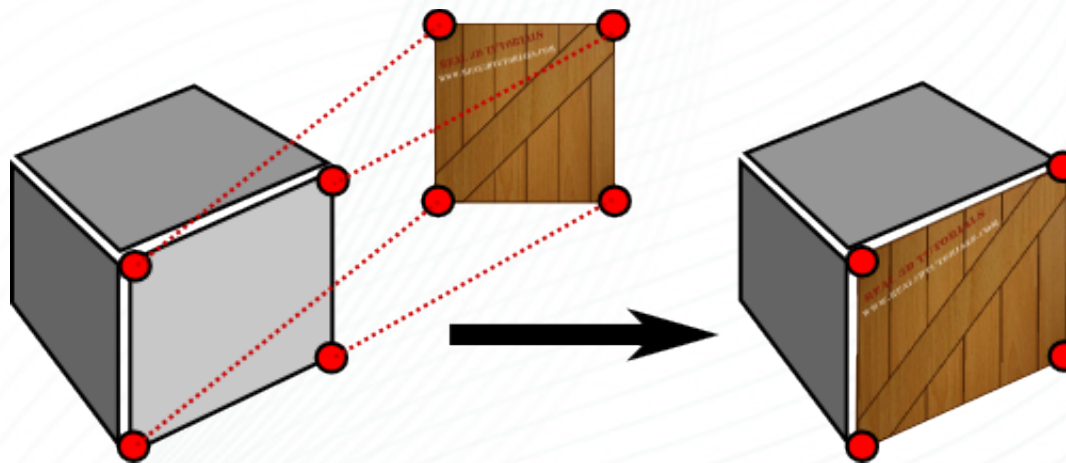
# Mapeamento de textura

- Texturas são imagens que são usadas para adicionar detalhes extras na tonalização de superfícies que, de outra forma, dariam uma aparência muito homogênea e artificial.
- Cada Shading é computada, nós lemos um dos valores usados na computação de tonalização de uma textura, ao invés de usar os valores de atributos que estão anexados na geometria renderizada.

# Mapeamento de textura

- O código de shading especifica a coordenada de textura, um ponto no domínio de textura e o sistema de mapeamento de textura encontra o valor no ponto da imagem de textura.
- Então, o valor da textura é usado na operação de Shading.

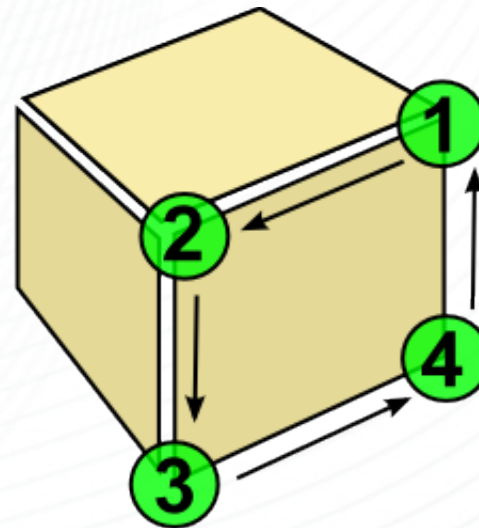
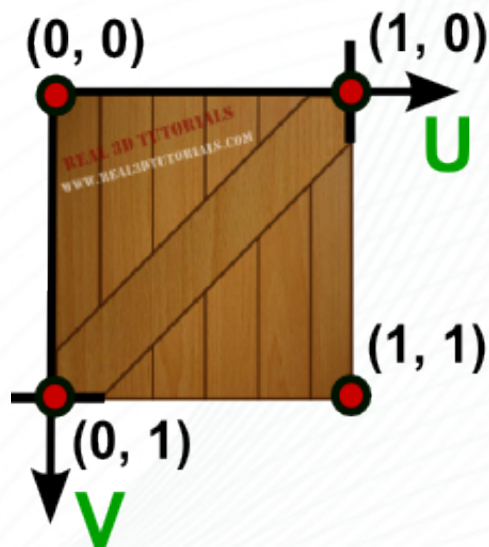
# Mapeamento de textura



<http://www.real3dtutorials.com/images/img00017.png>

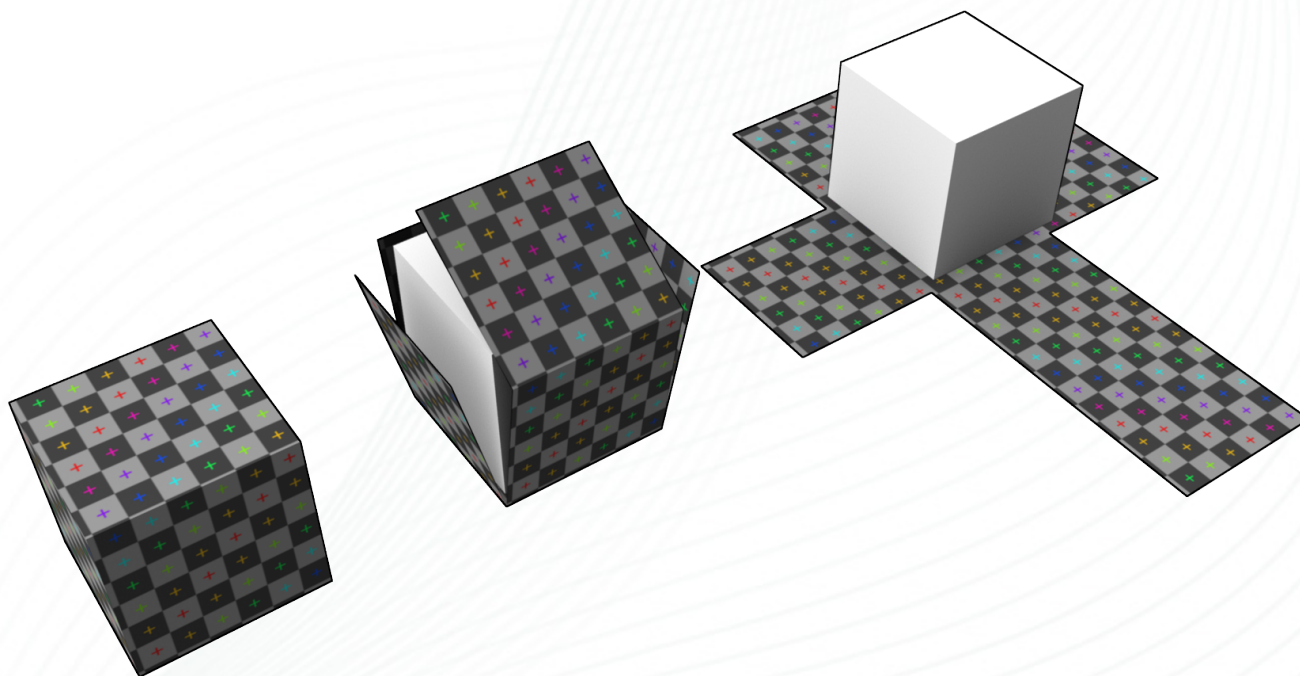


# Mapeamento de textura



<http://www.real3dtutorials.com/images/img00018.png>

# Mapeamento de textura



[https://upload.wikimedia.org/wikipedia/commons/f/fe/Cube\\_Representative\\_UV\\_Unwrapping.png](https://upload.wikimedia.org/wikipedia/commons/f/fe/Cube_Representative_UV_Unwrapping.png)



Aula 17

# TEXTURA COM OPENGL

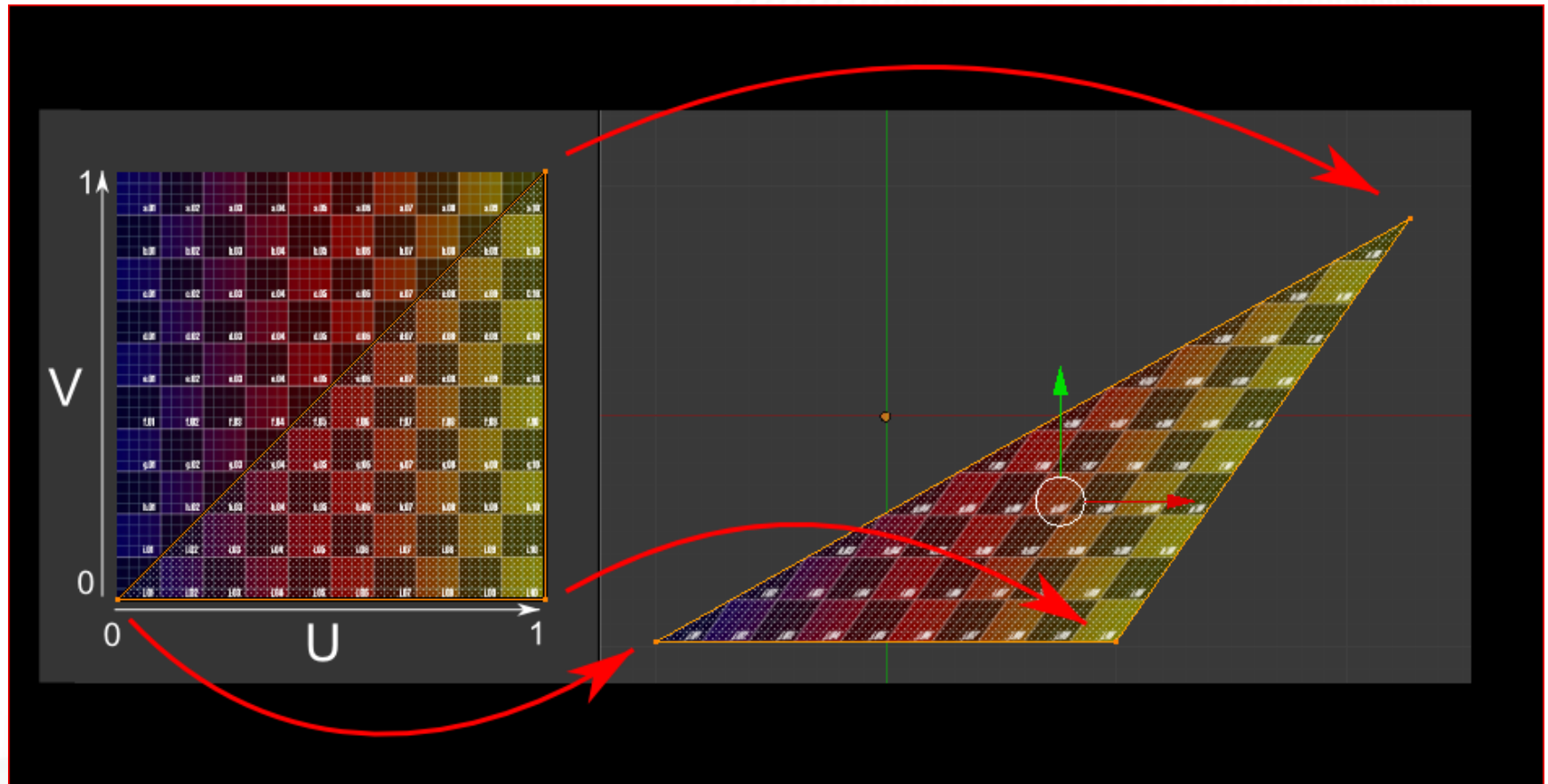
# Cubo com textura

- Coordenadas UV
- Comunicação entre *Vertex Shader* e *Fragment Shader*

# Coordenadas UV

- As coordenadas UV servem para, ao "texturizar" uma malha, passar a informação ao OpenGL qual parte da imagem deve ser usada para cada triângulo do cubo.
- Cada vértice pode ter, a cada posição, um par de valores do tipo *float*, que são os valores U e V. Estas coordenadas são usadas para acessar a textura, da seguinte maneira:

# Coordenadas UV



# Aplicação de textura

- Carregar um arquivo de imagem (BMP, DDS, etc)
  - Chamada de função
- Utilização da textura no OpenGL
  - Comunicação CPU – *Vertex Shader* – *Fragment Shader*
  - Mapeamento de coordenadas
  - Carregar textura com GLFW

Aula 17

# EXERCÍCIOS



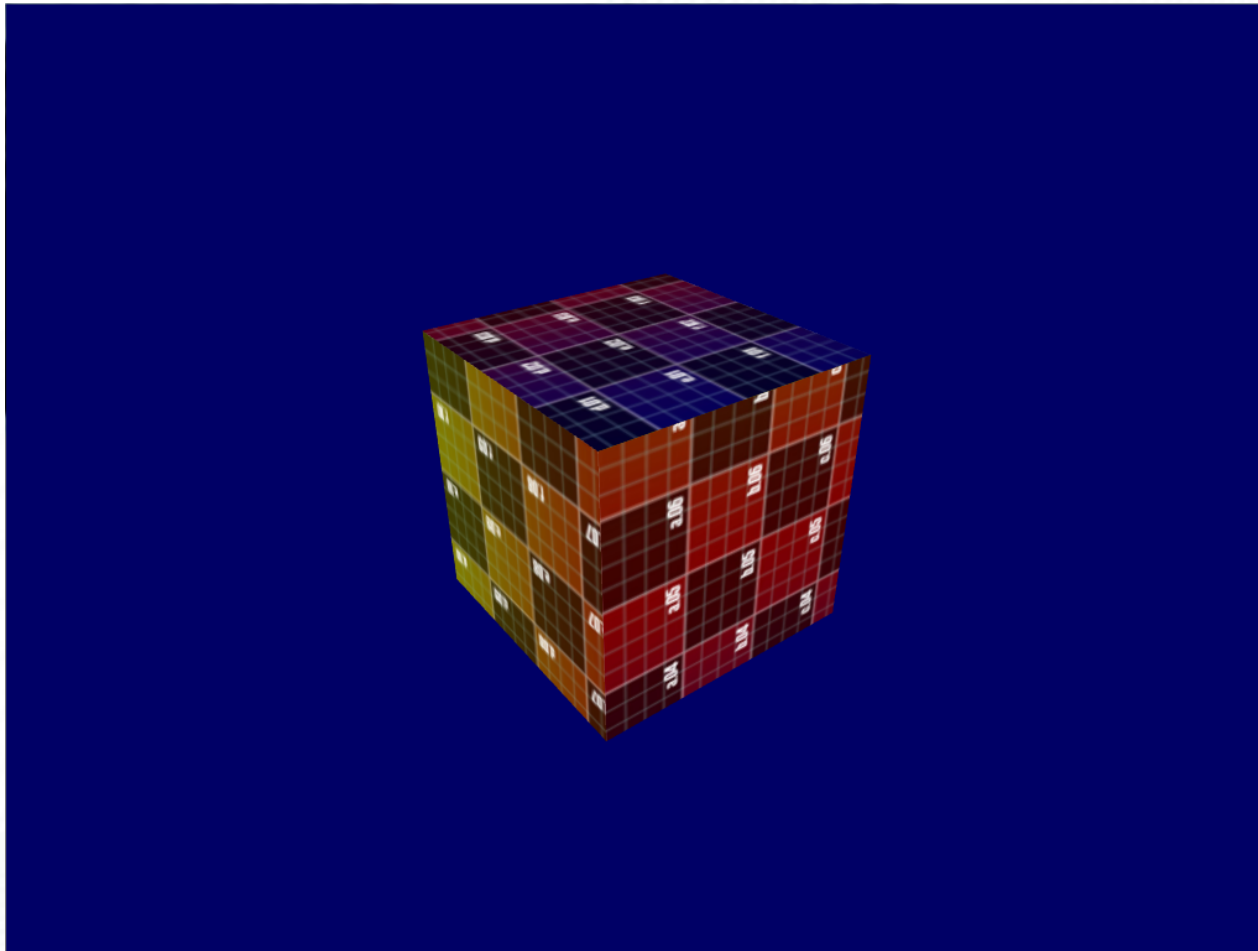
# Atividade 04

Cubo com textura

# Baixe os arquivos do Tidia

- Baixe os arquivos que estão no TIDIA – CG-Aula17;
- Faça o projeto que apresente o resultado final, a seguinte saída:

# Resultado



# Atividade 04

- Explique:
  - Como se carrega um arquivo de imagem, no projeto? Qual é a chamada de função para tal?
- Explique como acontece:
  - a comunicação CPU – *Vertex Shader* – *Fragment Shader*
  - o mapeamento de coordenadas
  - Carregar textura com GLFW
- Altere a textura do cubo.

# Sugestão

- Siga o tutorial descrito em <http://www.opengl-tutorial.org/beginners-tutorials/tutorial-5-a-textured-cube/>

# Referências

- OpenGL Tutorial

<http://www.opengl-tutorial.org/beginners-tutorials/>



# Fim da Aula 16

André Luiz Brandão