

# Rendering e Projeção/Observação de cenas 3D

Igor Jordany Richtic Silva<sup>1</sup>   Lucas Afrânio Silva de Siqueira<sup>1</sup>  
Vinicius Antonio da Conceição<sup>1</sup>

<sup>1</sup>Instituto de Computação  
Universidade Federal de Mato Grosso

Agosto 2017

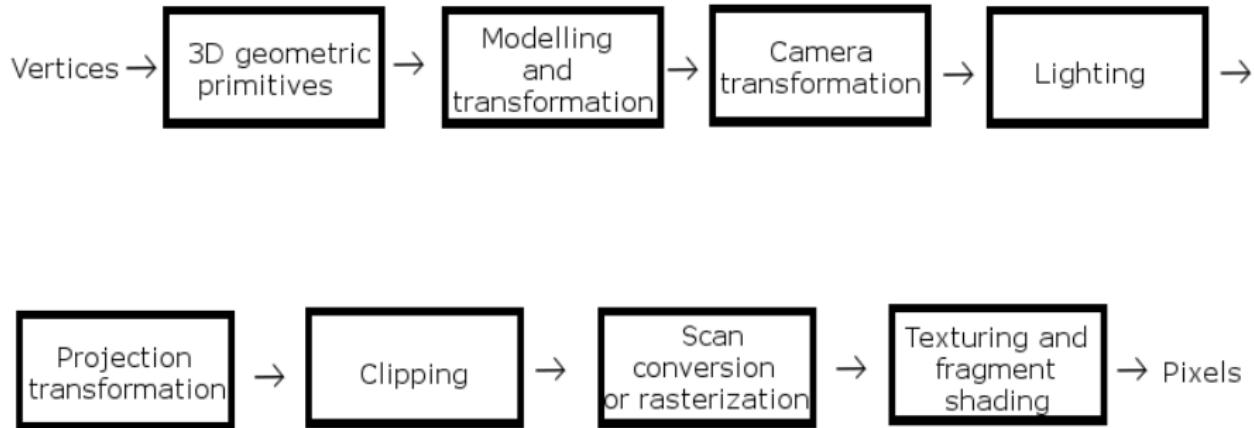
# Roteiro

- Rendering
  - Fases do processo de Realismo Visual
  - Realismo por passadas
  - Rasterização
  - Amostragem, Aliasing, e Anti-aliasing
  - Superamostragem
  - Exemplo de Anti-alising em linhas
  - Remoção de elementos ocultos
  - Z-Buffer
- Projeção/Observação
  - Projeção paralela
  - Projeção perspectiva
  - Leis da perspectiva
  - Tipos de perspectivas
- Dúvidas

# Rendering

- Podemos interpretar o processo de rendering como o de converter dados em uma imagem realística ou simplesmente sintetizar um objeto ou cena até ter-se deles uma aparência de algo real e não de formas inteiramente criadas no computador.
- Essa síntese de uma imagem (uma cena ou objeto) é criá-la em termos da definição dos dados dos objetos que a compõem. Isso se faz a partir da geometria da cena, das informações sobre os materiais de que são feitos os objetos (suas cores e texturas), das condições de iluminação ambiente e da posição de observação da cena.

# Fases do processo de Realismo Visual



# Realismo por passadas



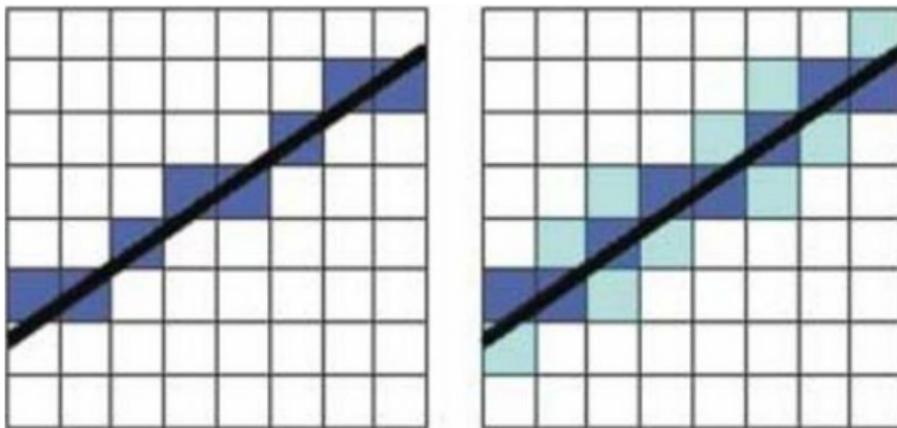
**THE BLUE GUARDIAN**

RENDERING PROCESS START TO FINISH

*Miguel 2013*

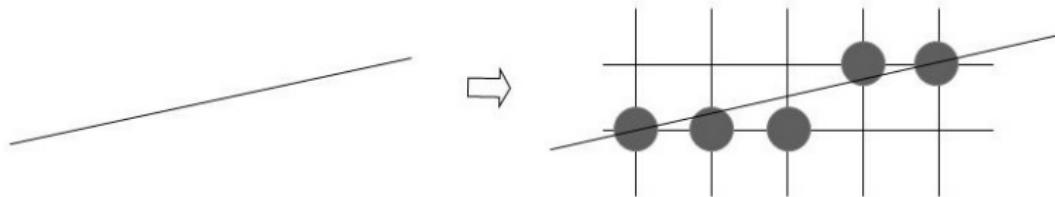
# Rasterização

- Rasterização é um processo de amostragem
- Problemas de aliasing são esperados
- Cada primitiva pode gerar um grande número de pixels
- Em geral, rasterização é feita por hardware



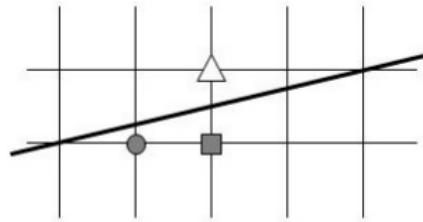
# Amostragem, Aliasing, e Anti-aliasing

- A linha, que no universo físico é contínua, é amostrada em uma matriz finita 2D de pixels.
- Tal discretização pode causar distorções visuais como cisalhamento ou efeito de escada. Essas distorções são chamadas de aliasing. Para reduzir o problema ,usa-se uma técnica chamada antialiasing.
- A técnica consiste em uma superamostragem (uma vez que o aliasing é causada por uma subamostragem).

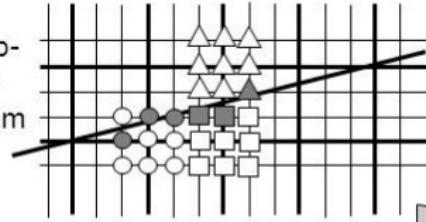


# Superamostragem

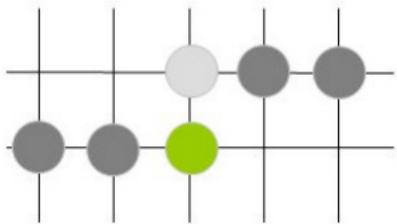
- Superamostragem = Amostrar um objeto numa resolução maior do que será reconstruído.



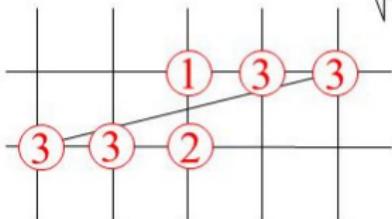
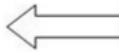
dividir os pixels em sub-pixels (i.e. 9), aplicar o algoritmo de Bresenham nesses sub-pixels



contar o número de sub-pixels “acesos” por pixel

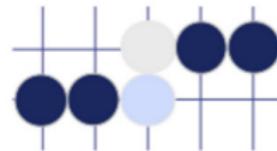
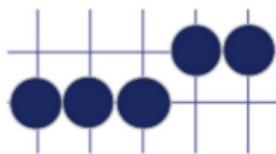


O pixel será aceso com intensidade proporcional ao número de sub-pixels acesos.

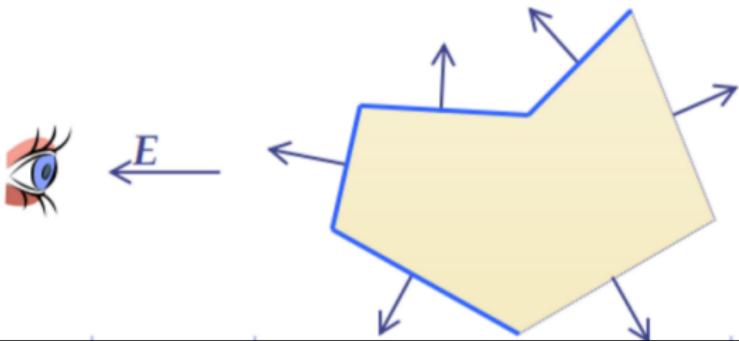
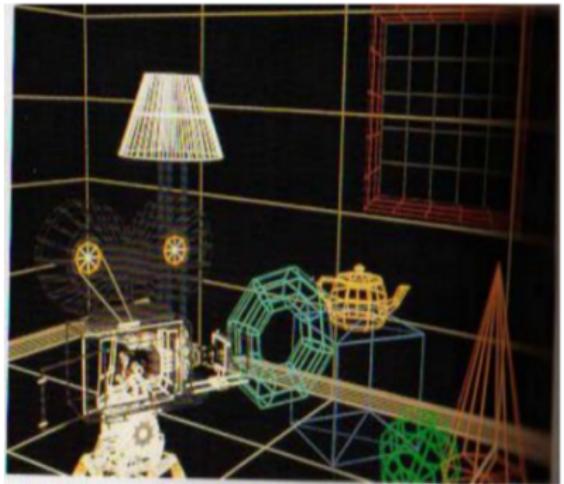


## Exemplo de Anti-aliasing em linhas

- Observe que quando a cor de fundo não é preto, o antialiasing deve fazer uma composição da intensidade com a cor de fundo.
- Anti-aliasing é necessário não só para linhas, mas também para polígonos e texturas (o que já é mais complicado).



# Remoção de elementos ocultos



# Z-Buffer

- Buffers são áreas reservadas de memória utilizadas para determinados propósitos.
- Em aplicações de animação, por exemplo, o double-buffer permite que as sucessivas renderizações sejam feitas de modo suave, sem o efeito indesejável de piscar entre cada atualização da janela de visualização.
- O z-buffer é bastante comum em aplicações gráficas e é utilizado para calcular a distância do observador e remover superfícies ou partes ocultas de objetos sobrepostos.

```
glutInitDisplayMode(GLUT_DOUBLE|GLUT_RGB|GLUT_DEPTH)
```

# Projeção

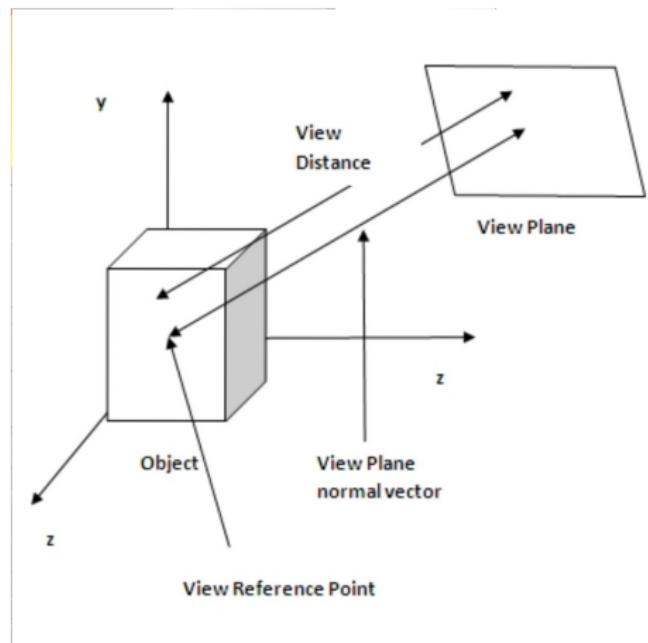
- No mundo real temos um objeto ex: uma torre. A torre está em um plano 3D. Você pode mover se ao redor da torre sobre o solo ou sobre o ar, e tirar uma fotografia que será convertida para uma imagem 2D. Dependendo de onde estiver a câmera você verá uma diferença na imagem, como sobras, nível de luz etc..

# Projeção



# Projeção

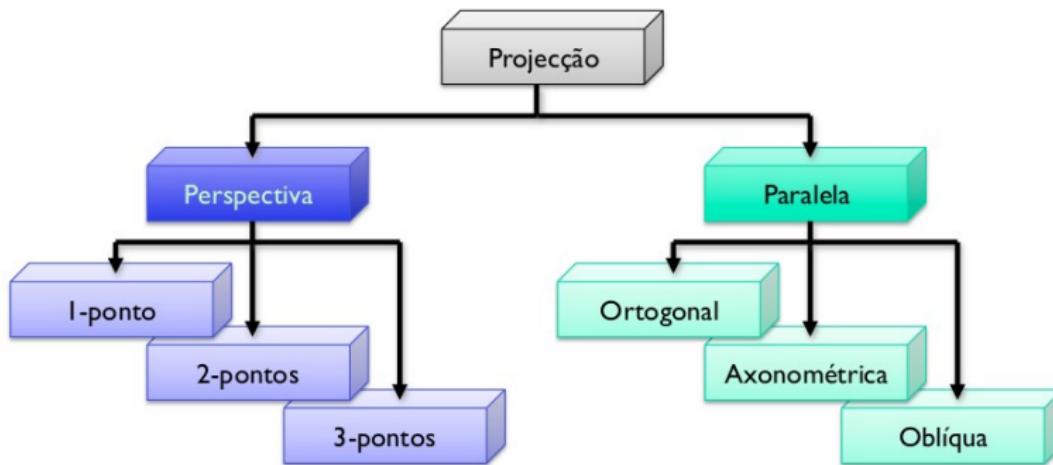
- No computador temos que simular um ponto de visão e a partir dele gerar nossos cenários.



# Projeção

Os tipos de projeções dependem de 2 fatores:

- Posição do observador
- Localização e orientação do plano de projeção

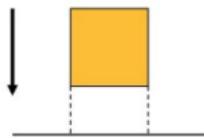


# Projeção paralela

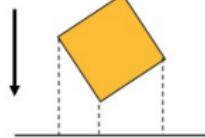
- Se o objeto está alinhado com os eixos, o resultado é uma projeção ortogonal.  
Caso contrário, é uma projeção axonométrica.
- Se o plano de projeção intersecta os eixos XYZ à mesma distância relativamente à origem, o resultado é uma projeção isométrica.

# Projeção paralela

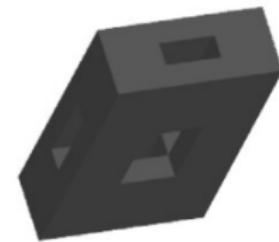
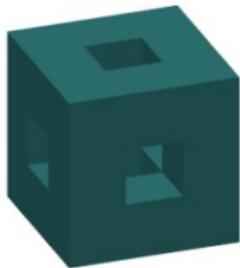
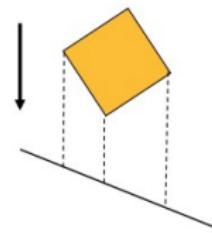
ortogonal



axonométrica



oblíqua

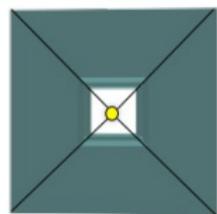


# Projeção perspectiva

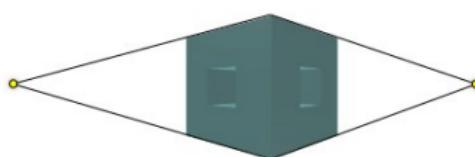
- O observador está a uma distância finita do/a objeto/cena. pontos (observadores).
- As projetantes não são paralelas e convergem para um ou mais.

# Projeção perspectiva

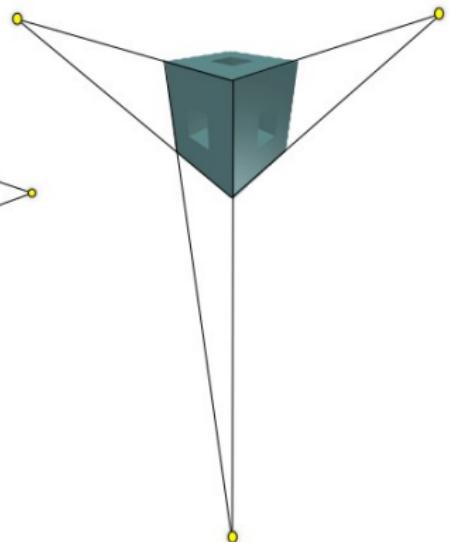
perspectiva com 1-ponto



perspectiva com 2-pontos

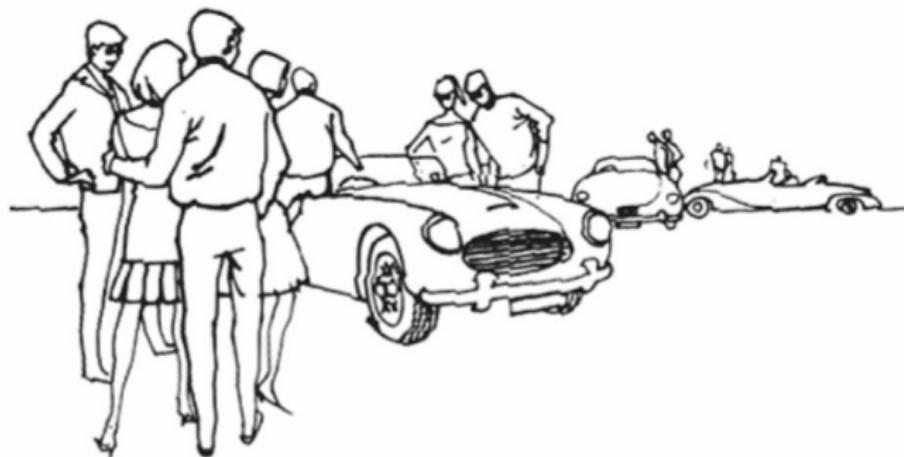


perspectiva com 3-pontos



# Leis da perspectiva

- Tudo parece diminuir à medida que se afasta do observador (desenhista).

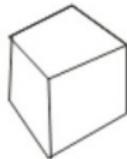


# Leis da perspectiva

- Linhas retas horizontais ou verticais tendem a se apresentar diagonais quando “entram” para o fundo.
- Já quando “estão” à sua frente, mantém sua perpendicularidade.



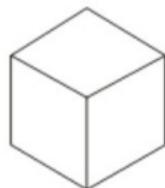
# Tipos de Perspectivas



Perspectiva cônica



Perspectiva Cavaleira



Perspectiva Isométrica

- Cada tipo de perspectiva mostra o objeto de um jeito.
- Comparando as três formas de representação, nota-se que a perspectiva isométrica é a perspectiva que dá a sensação de menor deformação do objeto.

# Dúvidas?

