

Viewing 3D

SCC0250 - Computação Gráfica

Prof. Rosane Minghim

[https://sites.google.com/site/computacaograficaicmc2017t2/
rminghim@icmc.usp.br](https://sites.google.com/site/computacaograficaicmc2017t2/rminghim@icmc.usp.br)

P.A.E. Nicolas Roque nrsantos@usp.br

Instituto de Ciências Matemáticas e de Computação (ICMC)
Universidade de São Paulo (USP)

25 de abril de 2017



Ambiente Visual 3D

O ambiente visual 3D é composto por uma câmera e uma projeção.

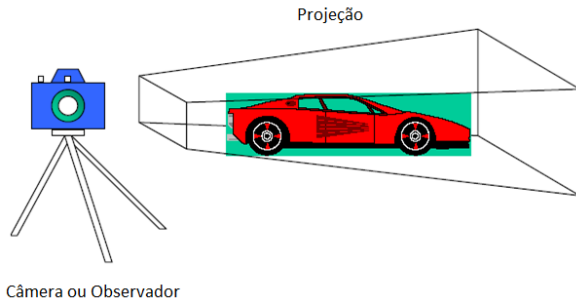


Figura: Exemplo do ambiente visual 3D.

Câmera

- A câmera define a região visível da cena para quem a observa.
- A região visível da cena é chamada de **volume de vista**.
- Chamamos de **frustum** o volume de vista na projeção perspectiva.

Câmera

- Em OpenGL, a câmera é posicionada por meio da seguinte função:
 - `gluLookAt (eyeX, eyeY, eyeZ, centerX, centerY, centerZ, upX, upY, upZ)`
 - Onde os parâmetros
 - `eyeX`, `eyeY` e `eyeZ` representam a posição onde a câmera será posicionada.
 - `centerX`, `centerY` e `centerZ` representam a posição para a qual a câmera está olhando.
 - `upX`, `upY` e `upZ` são as coordenadas de um vetor que representa a vertical da câmera.

Câmera

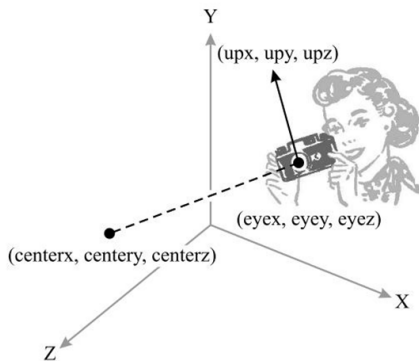


Figura: Exemplo dos pontos e vetores utilizados pela `gluLookAt`.

Câmera

- O vetor UP é também chamado de **VUP**.
- O ponto **eye** é também conhecido como **posição do observador** ou apenas **posição**.
- O ponto **center** é chamado de **ponto focal**.
- Existe um versor (vetor unitário) saindo da **posição do observador** na direção do **ponto focal** que é chamado de **DOP**.
- As coordenadas dos pontos **posição** e **ponto focal** são dadas em relação às coordenadas globais.

Projeções

- Obtém representações bidimensionais de objetos tridimensionais.
- A projeção é definida por raios de projeção (projetores) que passam através de cada vértice dos objetos e interceptam o plano de projeção.
- Projeções divididas em tipos:
 - Paralela ortográfica.
 - Perspectiva.

Projeções

- **Projeção Paralela Ortográfica:** projeta os pontos de um objeto ao longo de linhas paralelas. Utilizado para desenhos arquitetônicos e de engenharia.
- **Projeção Perspectiva:** projeta os pontos de um objeto ao longo de caminhos convergentes.
 - Produz cenas realísticas, onde objetos longe de posição de visão são menores.



(a) Projeção Paralela



(b) Projeção Perspectiva

Projeção Perspectiva

Os projetores tem início em um único ponto e está uma distância finita do plano de projeção.

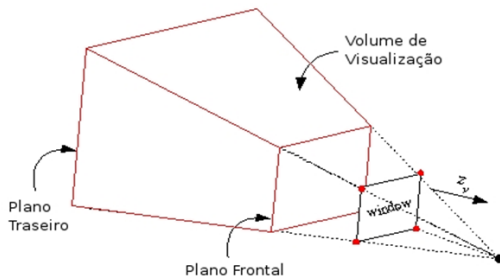


Figura: Exemplo de uma projeção perspectiva.

Projeção Perspectiva

- Em OpenGL a projeção perspectiva é definida por meio da seguinte função:
 - `gluPerspective (fovy, aspect, zNear, zFar)`
 - Onde os parâmetros
 - `fovy` indica o ângulo de abertura da câmera na direção de Y. Onde $0^0 \leq fovy \leq 180^0$.
 - `aspect` especifica a razão de aspecto que determina o campo de visão na direção de X.
 - `zNear` é a distância da câmera para o plano frontal.
 - `zFar` é a distância da câmera para o plano traseiro.

Projeção Perspectiva

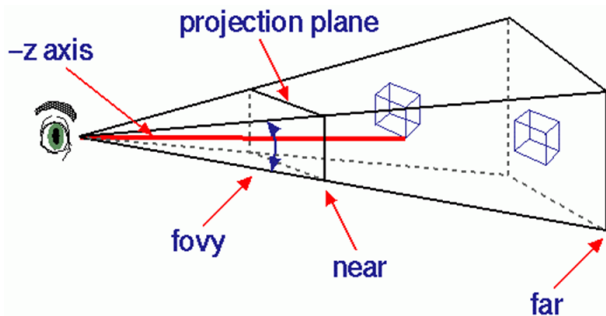


Figura: Exemplo dos parâmetros passados para a `gluPerspective`.

Projeção Perspectiva

- O frustum (volume de vista) é um troco de uma pirâmide finita.
- Em OpenGL o frustum é definido por meio da seguinte função:
 - `glFrustum (left, right, bottom, top, near, far)`

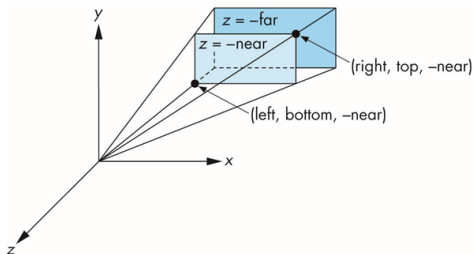


Figura: Exemplo de um frustum.

Projeção Paralela

- Os projetores são paralelos entre si e perpendiculares ao plano de projeção.
- O volume de vista é um paralelepípedo infinito ou finito, se os planos de frente e de fundo forem considerados.
- Não há alterações na medida do objeto.

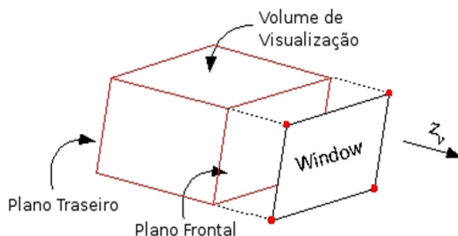


Figura: Exemplo de uma projeção paralela.

Projeção Paralela

- Em OpenGL a projeção paralela é definida por meio da seguinte função:
 - `gluOrtho` (left, right, bottom, top, near, far)
 - Onde os parâmetros definem os limites mínimo e máximo da janela de projeção em X, Y e Z.

Bibliografia

- **Básica:**

- Hearn, D. Baker, M. P. Computer Graphics with OpenGL, Prentice Hall, 2004. **(livro texto)**
- Angel, E. Interactive computer graphics: a top-down approach with OpenGL, Addison Wesley, 2000.
- Foley, J. et. al - Introduction to Computer Graphics, Addison-Wesley, 1993.
- Kessenich, J., Sellers, G., Shreiner, D. OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.5 with SPIR-V - Ninth Edition.

Bibliografia

- **Complementar:**

- Computer Graphics Comes of Age: An Interview with Andries van Dam. CACM, vol. 27, no. 7. 1982
- The RenderMan – And the Oscar Goes to... IEEE Spectrum, vol. 38, no. 4, abril de 2001.