# Computação Gráfica

André Brandão

# Aula 09

Visualização

# Sumário

- Transformações de visualização
- Transformações de projeção
- Projeções em perspectiva
- Propriedades das projeções em perspectiva
- Campo de visão

# Introdução

- No Ray Tracing, vimos que, para cada pixel, um raio é traçado. Em seguida, verifica-se se esse raio, a partir do pixel, tem alguma interseção.
- Na visualização, o processo é justamente o oposto. Sendo assim, para cada ponto do modelo, verifica-se qual pixel será utilizado para projetá-lo.
- Esse processo funciona bem para renderizações “aramadas” (wireframe rendering).

# Wireframe Rendering

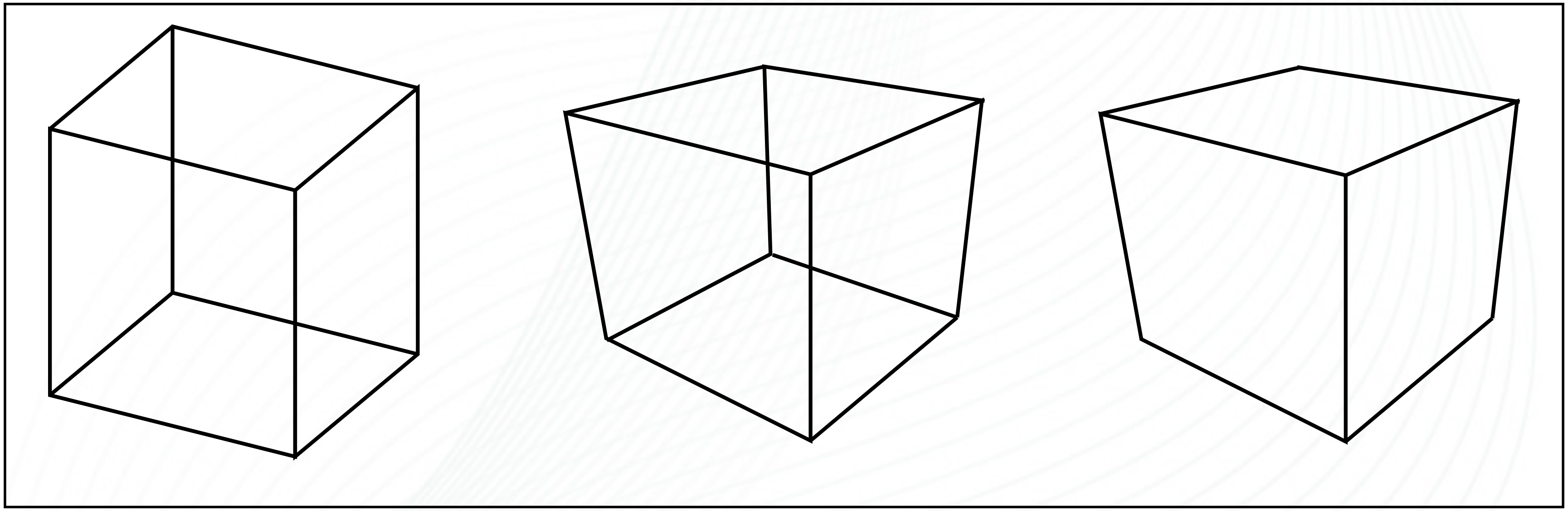


**Figure 7.1.** Left: wireframe cube in orthographic projection. Middle: wireframe cube in perspective projection. Right: perspective projection with hidden lines removed.

# Sumário

- **Transformações de visualização**
- Transformações de projeção
- Projeções em perspectiva
- Propriedades das projeções em perspectiva
- Campo de visão

# Transformações de visualização

- As transformações de visualização tem o papel de mapear localizações em 3D, representadas em (x,y,z), para coordenadas de imagem, expressadas em unidades de pixels.
- Algumas das variáveis são: posição e orientação da câmera, o tipo de projeção, o campo de visão e a resolução da imagem.

# Transformações de visualização

- Tratar essas variáveis pode ser uma tarefa complexa. Por isso, pode-se dividir esse tipo de transformação em três transformações:
  - **Transformação de câmera**: transforma o corpo rígido e posiciona a câmera como o ponto origem. Depende somente da posição e orientação da câmera
  - **Transformação de projeção**: projeta pontos do espaço de câmera, de modo que os pontos visíveis fiquem no intervalo [-1, 1] (volume de visualização canônica), tanto em *x* como em *y*. Depende somente do tipo de projeção desejado
  - **Transformação de janela**: mapeia o retângulo da transformação de projeção para o retângulo redesenhado em coordenadas de pixels. Depende somente do tamanho e posição da imagem que será exibida.

# Transformações de visualização

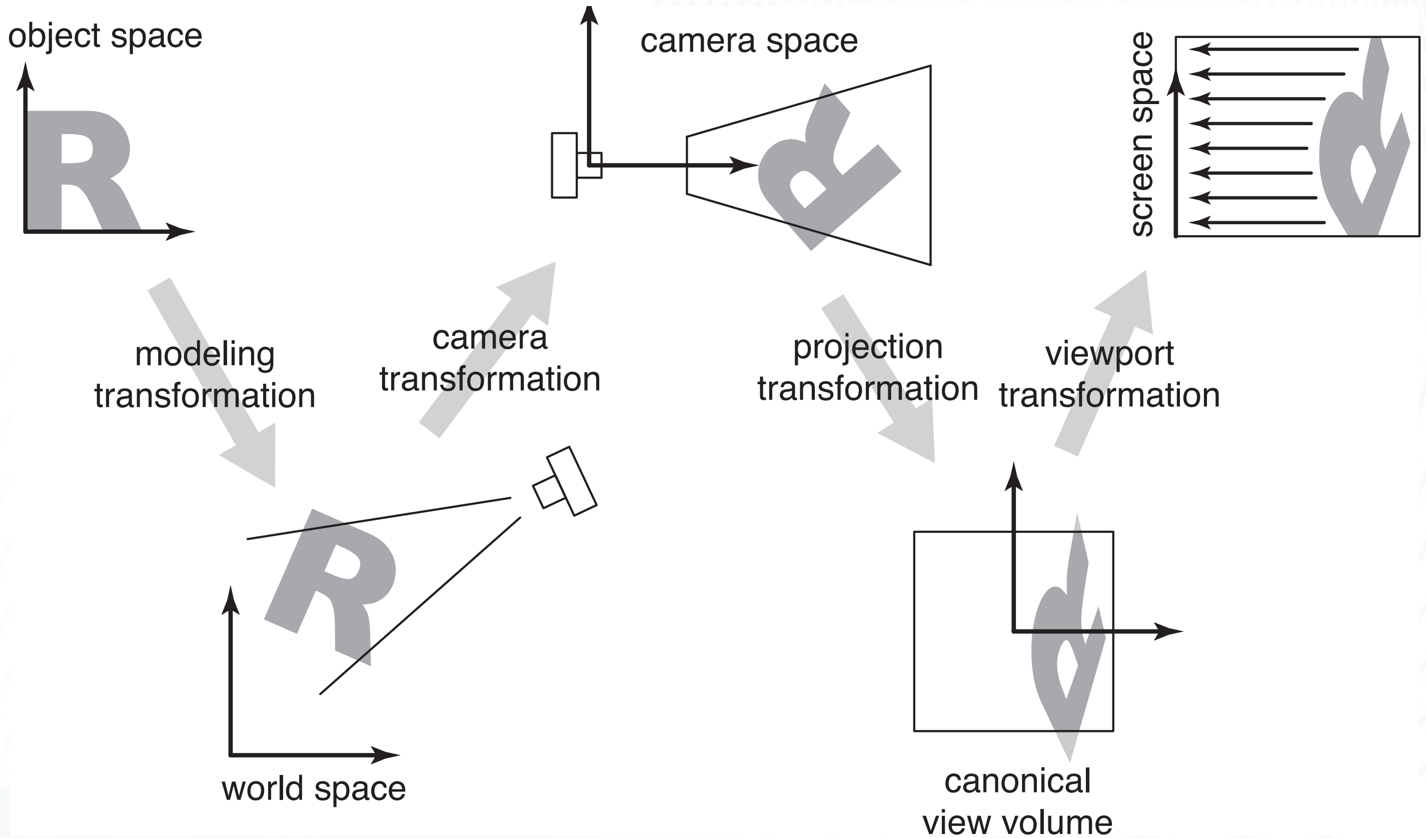
- Para facilitar, damos nomes aos sistemas de coordenadas que são entradas ou saídas das transformações.
- A transformação de câmera converte pontos de coordenadas canônicas (ou espaço de mundo) para coordenadas de câmera.

# Transformações de visualização

- A transformação de projeção passa pontos do espaço de câmera para volume de visualização canônica (pontos estão em [-1, 1]).
- A transformação de janela mapeia o volume de visualização canônica para espaço de tela.

- Cada uma das transformações será discutida para a projeção ortográfica.

# Transformações de visualização

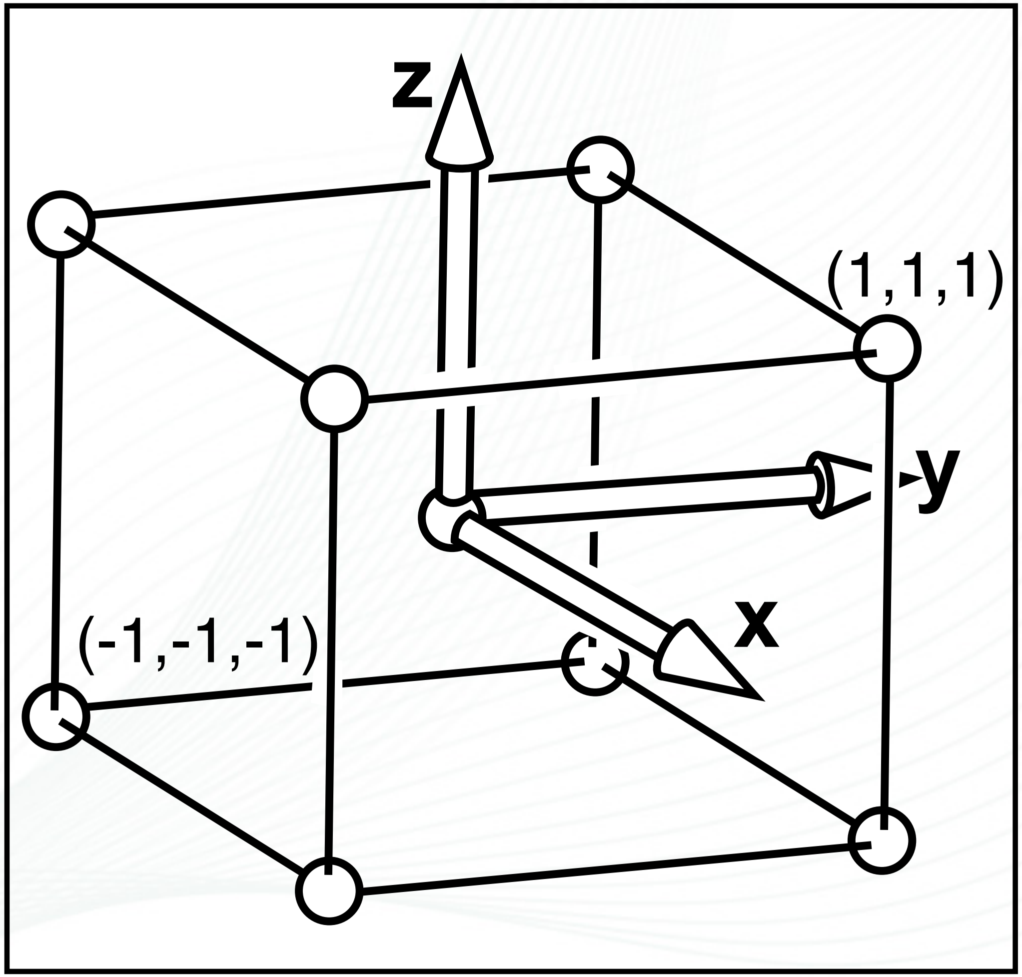


# Transformação de janela

- A geometria que queremos ver está no volume de visualização canônica (valores estão no [-1, 1]) e queremos vê-lo em uma câmera ortográfica na direção –z.

$$(x, y, z) \in [-1, 1]^3$$

# Transformação de janela



# Transformação de janela

- Cada pixel "tem" uma unidade ao quadrado centrado em coordenadas inteiras; e os limites da imagem ultrapassam bordas em meia unidade; e as menores coordenadas do centro do pixel são (0, 0).
- Se estamos a desenhar plano de projeçnao (ou janela na tela), que tem $n_x$ por $n_y$ pixels, precisamos mapear o quadrado $[-1, 1]^2$ para o retângulo $[-0.5, n_x - 0,5] \times [-0.5, n_y - 0,5]$.

# Transformação de janela

- Porque a transformação de janela mapeia um retângulo com os eixos alinhados para outro retângulo (retângulo de pixels), a transformação de janela é dada por:

$$\begin{bmatrix} x_{\text{screen}} \\ y_{\text{screen}} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{n_x}{2} & 0 & \frac{n_x-1}{2} \\ 0 & \frac{n_y}{2} & \frac{n_y-1}{2} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{\text{canonical}} \\ y_{\text{canonical}} \\ 1 \end{bmatrix}$$

# Transformação de janela

- Note que a coordenada z é ignorada na matriz de transformação de janela. Isso ocorre porque a profundidade do ponto não afeta a posição deste na imagem projetada.
- Porém, se for importante considerar a coordenada z, então uma linha e uma coluna são acrescentadas na matriz.

# Matriz de transformação de janela

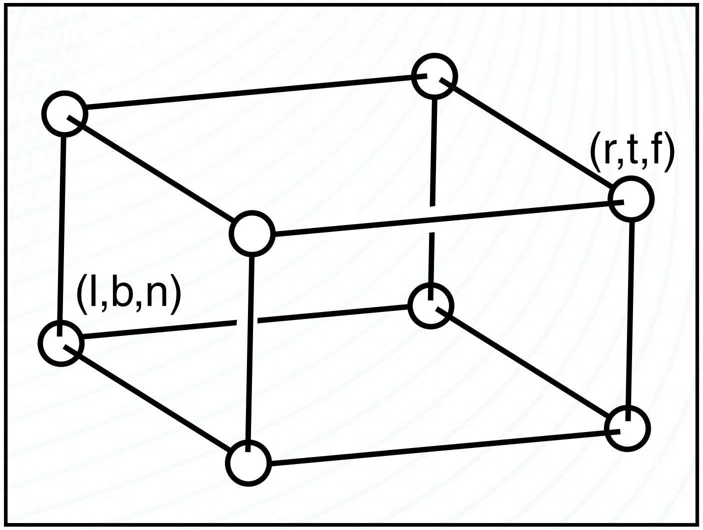
$$M_{\mathrm{vp}} = \begin{bmatrix} \frac{n_x}{2} & 0 & 0 & \frac{n_x-1}{2} \\ 0 & \frac{n_y}{2} & 0 & \frac{n_y-1}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Transformação de projeção ortográfica

- É usual projetar geometrias em algum espaço que não seja o volume de visualização canônica (valores estão no [-1, 1]).
- Primeiro vamos generalizar a vista e manteremos a direção da vista e orientação fixa, olhando ao longo de $-z$ e $+y$. Mas, permitiremos que retângulos arbitrários sejam visualizados.
- Em vez de substituir a matriz de transformação de janela, vamos aumentá-la, multiplicando-a por outra matriz.

# Transformação de projeção ortográfica

- O volume de visualização é uma caixa alinhada a eixo e vamos nomear as coordenadas dos seus lados de modo que o volume contém *[l,r]x[b,t],[f,n]*.
- Chamamos esta caixa de volume de visão ortográfica.



# Transformação de projeção ortográfica

- Referimo-nos aos planos, como segue:

*x = l = plano da esquerda*

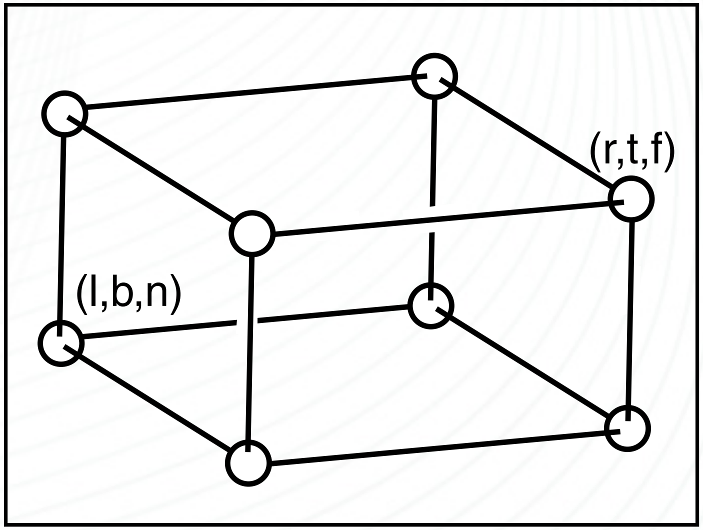
*x = r = plano da direita*

*y = b = plano da base*

*y = t = plano do topo*

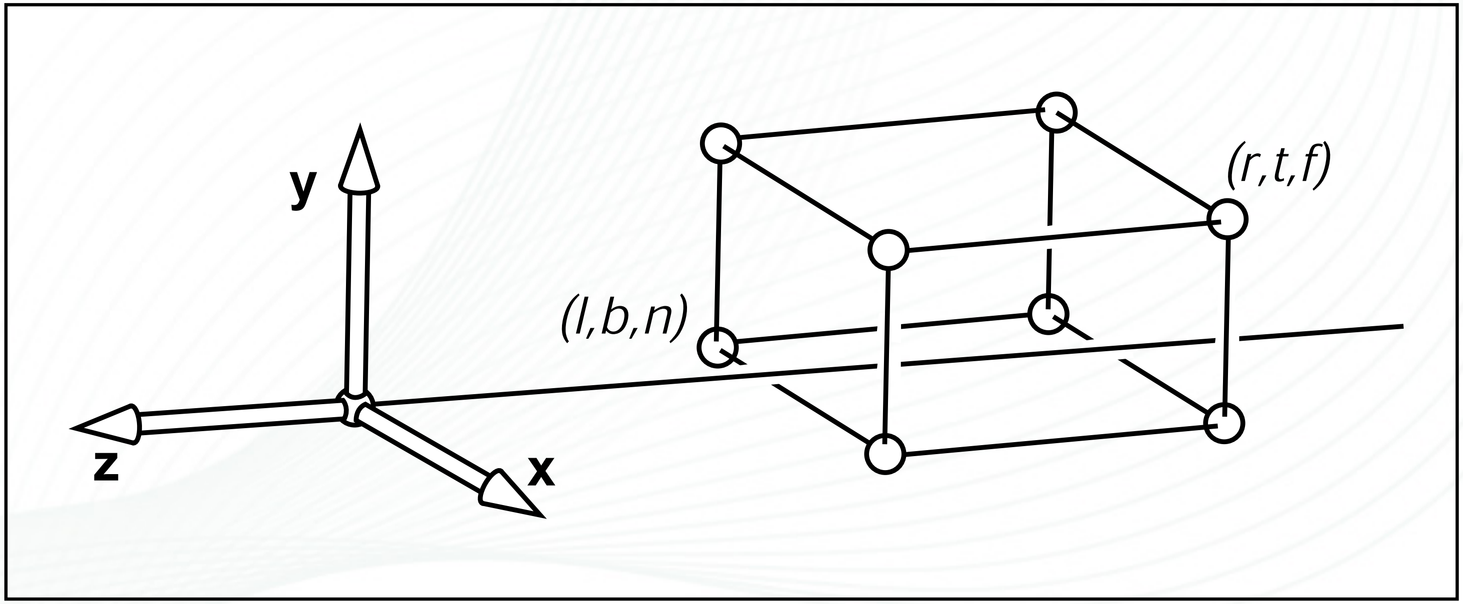
*z = n = plano próximo*

*z = f = plano longe*



# Transformação de projeção ortográfica

- Esse vocabulário assume que o observador está posicionado no eixo $-z$, com a sua face na direção de $y$.



# Transformação de projeção ortográfica

- Se quisermos transformar a visão ortográfica em volume de visualização canônica (valores estão no [-1, 1]), aplicamos a matriz de transformação de projeção ortográfica:

$$\mathbf{M}_{\text{orth}} = \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & \frac{2}{n-f} & -\frac{n+f}{n-f} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Transformação de projeção ortográfica

- Para desenhar linhas 3D em um volume de visão ortográfica, os projetamos na tela com os valores das coordenadas $x$ e $y$ e ignoramos a coordenada $z$.
- Assim, multiplicamos as matrizes de transformação de janela e de projeção ortográfica.

# Matriz de transformação de janela

$$M_{\text{vp}} = \begin{bmatrix} \frac{n_x}{2} & 0 & 0 & \frac{n_x - 1}{2} \\ 0 & \frac{n_y}{2} & 0 & \frac{n_y - 1}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Matriz de transformação de projeção ortográfica

$$\mathbf{M}_{\text{orth}} = \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & \frac{2}{n-f} & -\frac{n+f}{n-f} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Transformação de projeção ortográfica

$$\begin{bmatrix} x_{\text{pixel}} \\ y_{\text{pixel}} \\ z_{\text{canonical}} \\ 1 \end{bmatrix} = (\mathbf{M}_{\text{vp}}\mathbf{M}_{\text{orth}}) \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

# Transformação de câmera

- Há muitas maneiras de se mudar o ponto de vista para visualizar algo no espaço 3D.
- Usaremos a seguinte notação:

***e*** é a posição dos olhos do observador

***g*** é a direção do olhar

***t*** é o vetor para olhar para cima

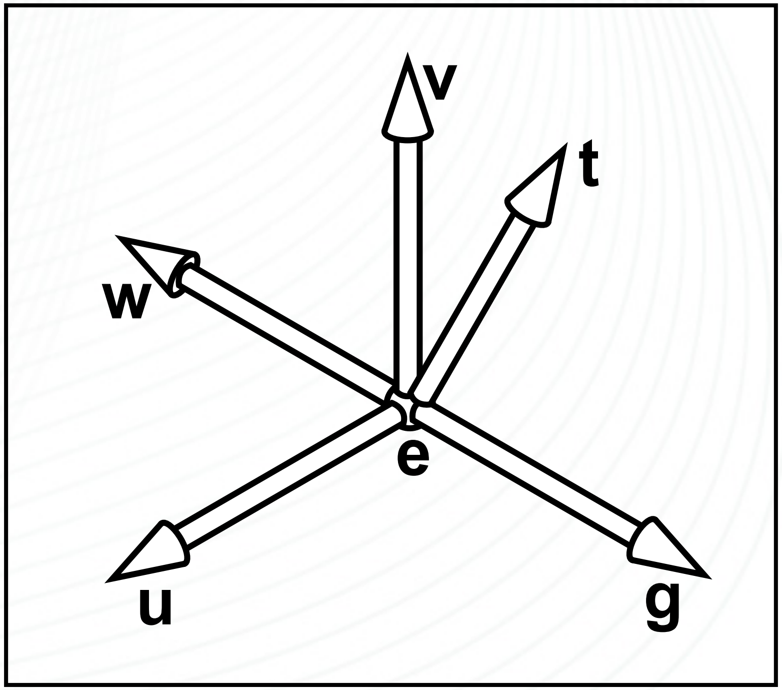
# Transformação de câmera

***e*** é a posição dos olhos do observador

***g*** é a direção do olhar

***t*** é o vetor para olhar para cima

- Assim, podemos definir um sistema de coordenadas com base nos vetores ***uvw***.



# Transformação de câmera

- Assim, podemos definir um sistema de coordenadas com base nos vetores ***uvw***.

$$\mathbf{w} = -\frac{\mathbf{g}}{\|\mathbf{g}\|},$$

$$\mathbf{u} = \frac{\mathbf{t} \times \mathbf{w}}{\|\mathbf{t} \times \mathbf{w}\|},$$

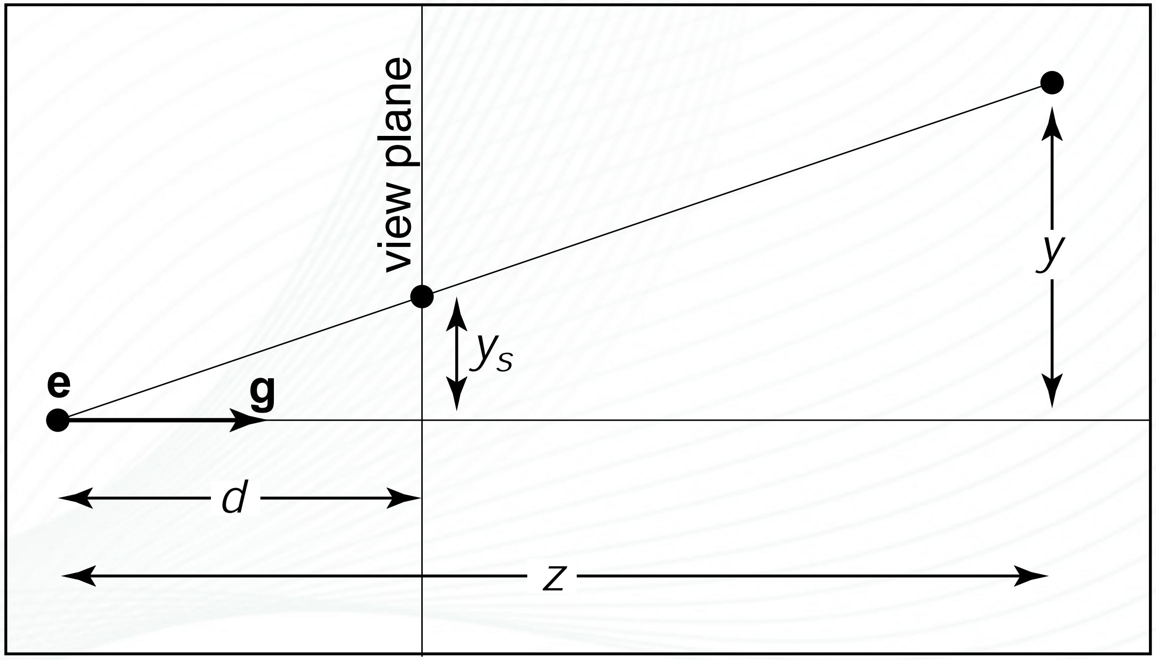
$$\mathbf{v} = \mathbf{w} \times \mathbf{u}.$$

# Sumário

- ~~Transformações de visualização~~
- **Transformações de projeção**
- Projeções em perspectiva
- Propriedades das projeções em perspectiva
- Campo de visão

# Transformações de projeção

- Na projeção em perspectiva, os olhos do observador estão localizados sobre o eixo ***z***.

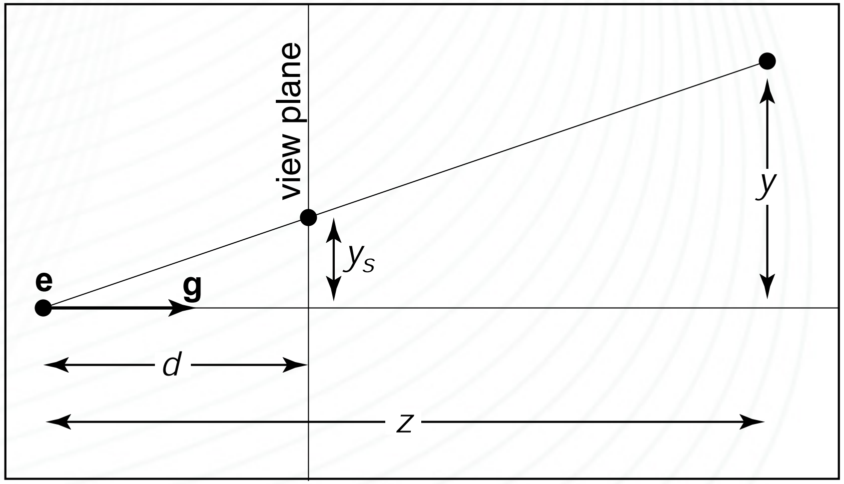


# Transformações de projeção

- Na projeção em perspectiva, o tamanho de um objeto é proporcional a ***1/z***.

$$y_s = \frac{d}{z} y$$

onde, ***y*** é a distância do ponto sobre o eixo y e $\boldsymbol{y_s}$ é onde o pondo deve ser desenhado na tela.



# Transformações de projeção

- Seria bastante conveniente utilizar a matriz de projeção ortográfica e multiplicá-la por alguma outra matriz para obter uma matriz de transformação de projeção em perspectiva.
- Porém, devemos saber, minimamente, o que são coordenadas homogêneas e transformações afins.

# Coordenadas Homogêneas

- Introduzida em Matemática
- Adiciona uma terceira coordenada *w* $\begin{bmatrix} x \\ y \\ w \end{bmatrix}$
- Um ponto 2D passa a ser um vetor com 3 coordenadas
- 2 pontos são iguais se e somente se: $\frac{x'}{w'} = \frac{x}{w} \text{ e } \frac{y'}{w'} = \frac{y}{w}$
- Homogeneizar: dividir por w
- Pontos homogeneizados: $\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$

# Transformações de projeção

- Com a mesma lógica das coordenadas homogêneas, que permitem transformações afins (tema para aulas após a P2), poderemos utilizar a matriz de projeção ortográfica para obtermos a matriz de projeção em perspectiva.
- As coordenadas homogêneas auxiliam na criação de novas bases ortogonais.

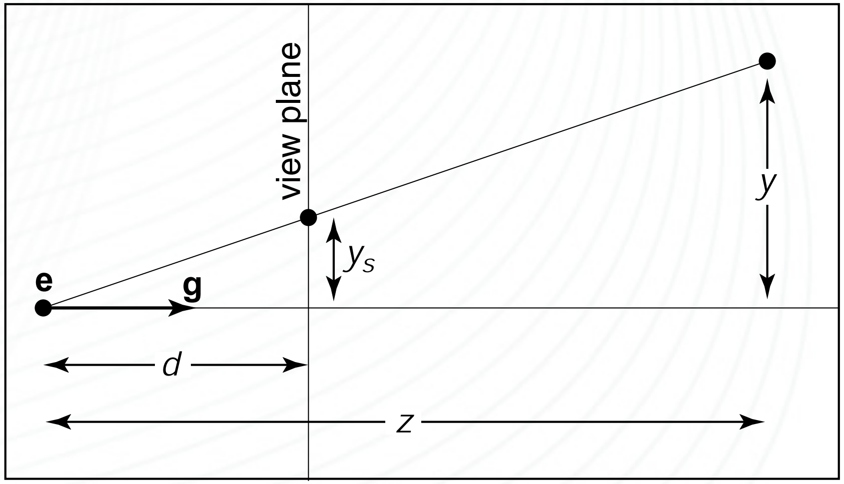
# Sumário

- ~~Transformações de visualização~~
- ~~Transformações de projeção~~
- **Projeções em perspectiva**
- Propriedades das projeções em perspectiva
- Campo de visão

UFABC

# Projeções em perspectiva

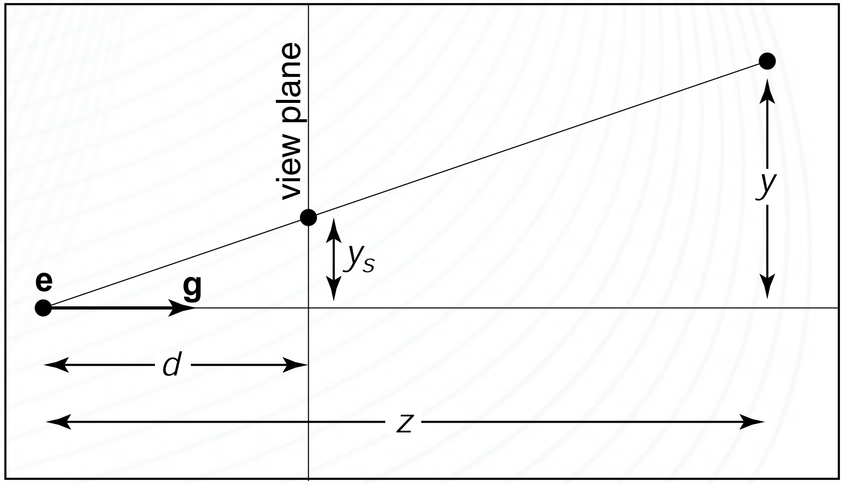
- Na projeção em perspectiva, todas as coordenadas nos eixos $y_s$ e d mudam em função de y. Portanto, há um deslocamento apenas em um eixo e o valor de $y_s$ e d devem ser computados a cada movimento da câmera no eixo y.



# Projeções em perspectiva

- Então, para implementar a transformação dos pixels da figura, pode-se aplicar a matriz de transformação:

$$\begin{bmatrix} y_s \\ 1 \end{bmatrix} \sim \begin{bmatrix} d & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} y \\ z \\ 1 \end{bmatrix}$$



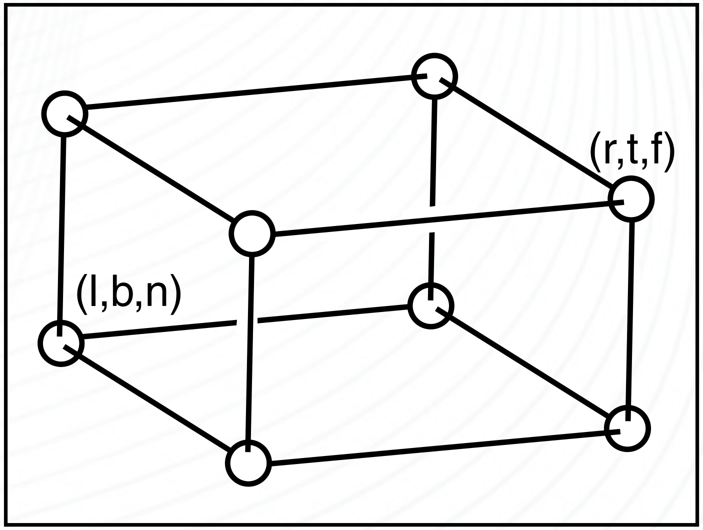

para $y_{s=\frac{d}{z}y}$

# Transformação de projeção ortográfica

- Na projeção ortográfica, adotamos que

  *z = n = plano próximo*
- Porém, na projeção em perspectiva, o plano da imagem é ***–n***.
- Assim, o mapeamento a ser feito é $y_s = \left(\frac{n}{z}\right) y$ e isso é similar para o eixo x.

Projeção ortográfica
*z = n = plano próximo*



# Projeções em perspectiva

- Assim, a matriz de perspectiva é:

$$\mathbf{P} = \begin{bmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & n+f & -fn \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

- Há muitas matrizes que mapeiam projeção ortográfica para perspectiva, todas elas distorcem a coordenada z de forma não linear.

# Projeções em perspectiva

- Ao concatenar a matriz perspectiva com a matriz de transformação de projeção ortográfica, teremos:

$$M_{per} = M_{orth}P$$

- Assim, a $M_{per}$ tem a seguinte aparência:

$$\mathrm{M_{per}} = \begin{bmatrix} \frac{2n}{r-l} & 0 & \frac{l+r}{l-r} & 0 \\ 0 & \frac{2n}{t-b} & \frac{b+t}{b-t} & 0 \\ 0 & 0 & \frac{f+n}{n-f} & \frac{2fn}{f-n} \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

# Projeções em perspectiva

- As explicações matemáticas sobre como se chega a cada matriz são melhor detalhadas nas leituras.
- Também, em disciplinas do programa de pós-graduação da UFABC, as explicações matemáticas podem ser revisadas.
- Por questões de priorização, as matrizes de transformação são apenas apresentadas nesta disciplina, sem as deduções.

# Sumário

- ~~Transformações de visualização~~
- ~~Transformações de projeção~~
- ~~Projeções em perspectiva~~
- **Propriedades das projeções em perspectiva**
- Campo de visão

# Propriedades das projeções em perspectiva

- As transformações em perspectiva levam linhas a linhas e planos a planos.
- Levam segmentos de reta do volume de visão a segmentos de reta no volume canônico.
- Assim, ao expandir essas propriedades, concluímos que as transformações em perspeciva levam vértices e arestas de triângulo para outro triângulo. Também, levam planos de uma projeção para planos em outra projeção.

# Sumário

- ~~Transformações de visualização~~
- ~~Transformações de projeção~~
- ~~Projeções em perspectiva~~
- ~~Propriedades das projeções em perspectiva~~
- **Campo de visão**

UFABC

# Campo de visão

- Nós podemos especificar qualquer janela utilizando o (***l, r, b, t***) e ***n*** valores, às vezes gostaríamos de ter um sistema mais simples, no qual nós olhamos através do centro da janela. Isto implica a restrição de que:

$$l = -r$$
$$b = -t$$

# Campo de visão

- Se também adicionarmos a restrição de que os pixels são quadrados, isto é, não há distorção na forma da imagem, então a relação de ***r*** a ***t*** deve ser a mesma que a relação entre o número de pixels horizontais com o número de pixels verticais:

$$\frac{n_x}{n_y} = \frac{r}{t}$$

UFABC

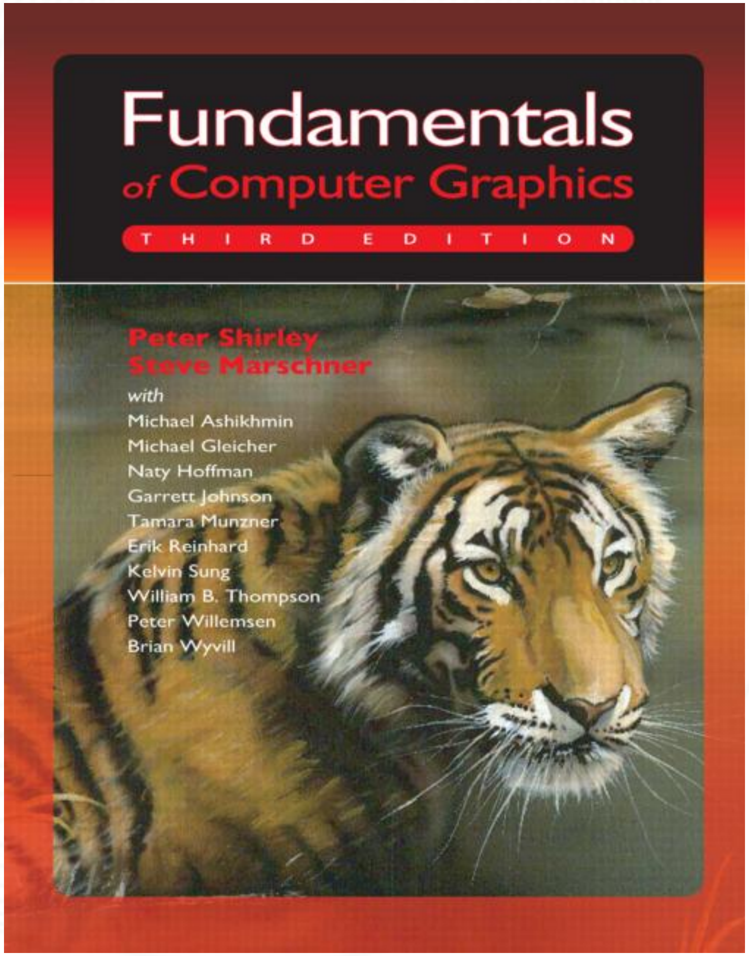
# Sumário

- ~~Transformações de visualização~~
- ~~Transformações de projeção~~
- ~~Projeções em perspectiva~~
- ~~Propriedades das projeções em perspectiva~~
- ~~Campo de visão~~

# Aula de hoje

Shirley, Peter, Michael Ashikhmin, and Steve Marschner. Fundamentals of computer graphics. CRC Press, 3rd Edition, 2009.

- **Capítulo 7**



# Fim da Aula 09

André Luiz Brandão