Aula – Computação Gráfica

Visualização
Projeção na Prática

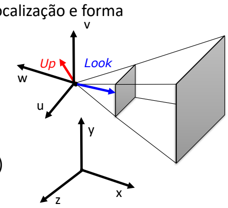
Slides para uso pessoal e exclusivo durante o período de aula. Distribuição ou qualquer uso fora do escopo da disciplina é expressamente proibido.

1

1

Visualização 3D Arbitrária

- Conhecendo a câmera virtual
  - Pode-se especificá-la através de localização e forma
- Localização
  - Posição (um ponto)
  - Look vector e Up vector
- Forma
  - Ângulo de abertura
  - Planos de corte (frontal e traseiro)

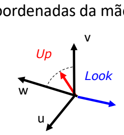


2

2

Visualização 3D Arbitrária

- Achando ***u, v*** e ***w***
  - Nosso Look vector estará ao longo do eixo negativo de ***w***
  - O vetor ***v*** será normal ao vetor Look e estará no plano formado pelos vetores Look e Up
  - ***u*** será mutuamente perpendicular a ***v*** e ***w*** para formar uma base de sistema de coordenadas da mão direita



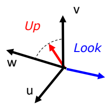
3

3

Visualização 3D Arbitrária

- Achando ***u, v*** e ***w***
  - Achar ***w*** é fácil
  - O vetor Look está no negativo de ***w***
  - ***w*** é um vetor unitário

$$\boldsymbol{w} = \frac{-\boldsymbol{Look}}{\|\boldsymbol{Look}\|}$$

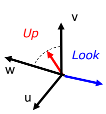


4

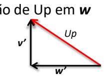
4

Visualização 3D Arbitrária

- Achando ***u, v*** e ***w***
  - Achar ***v***
  - Problema: Achar o vetor unitário perpendicular a ***w***
  - Solução:
    - Subtraia a componente ***w'*** de Up para obter ***v'*** e normalize
    - Para obter ***w'*** de Up, escale ***w*** pela projeção de Up em ***w***



$$\boldsymbol{Up} = \boldsymbol{w'} + \boldsymbol{v'}$$
$$\boldsymbol{v'} = \boldsymbol{Up} - \boldsymbol{w'}$$
$$\boldsymbol{v'} = \boldsymbol{Up} - (\boldsymbol{Up} \cdot \boldsymbol{w})\boldsymbol{w}$$
$$\boldsymbol{v} = \frac{\boldsymbol{v'}}{\|\boldsymbol{v'}\|}$$



Olhando diretamente para o plano-***wv***

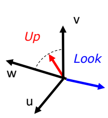
5

5

Visualização 3D Arbitrária

- Achando ***u, v*** e ***w***
  - Achar ***u***
  - Fazer produto vetorial entre ***v*** e ***w***

$$\boldsymbol{u} = \boldsymbol{v} \times \boldsymbol{w}$$



6

6

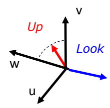
Visualização 3D Arbitrária

- Achando ***u, v*** e ***w***
  - Resumindo

$$\boldsymbol{w} = \frac{-\boldsymbol{Look}}{\|\boldsymbol{Look}\|}$$

$$\boldsymbol{v} = \frac{\boldsymbol{Up} - (\boldsymbol{Up} \cdot \boldsymbol{w})\boldsymbol{w}}{\|\boldsymbol{Up} - (\boldsymbol{Up} \cdot \boldsymbol{w})\boldsymbol{w}\|}$$

$$\boldsymbol{u} = \boldsymbol{v} \times \boldsymbol{w}$$



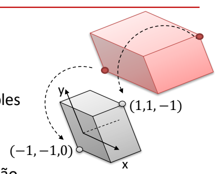
- Alternativamente: Calcular ***w***, fazer produto vetorial entre Up e ***w*** (ou Look e Up) para achar ***u***, fazer produto vetorial entre ***w*** e ***u*** para acha ***v***

7

7

O Volume de Visualização Canônico

- Como transformar um volume de visualização arbitrário para 2D?
  - Volume arbitrário é complexo
  - Reduza para um problema mais simples
    - Volume Canônico de Visualização
- Volume Canônico de Visualização
  - Possui parâmetros (orientação, posição, tamanho, etc.) específicos para facilitar operações de projeção e recorte
  - Antes de transformar objetos de 3D para 2D
    - Transforme todos os objetos da cena para o volume canônico

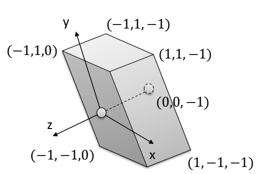


8

8

O Volume de Visualização Canônico Paralelo

- Começa na origem
  - Centro do plano frontal de corte (near plane) = (0,0,0)
- Olha ao longo do eixo negativo
  - Look vector = (0,0,-1)
- Up está em y
  - Up vector = (0,1,0)
- Janela de visualização normalizada
  - -1 a 1 na direções ***x*** e ***y***
- Planos de corte
  - Frontal z = 0
  - Traseiro z = 1

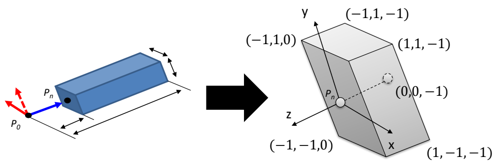


9

9

Transformação de Normalização

- Objetivo
  - Transformar um cena com um volume de visualização arbitrário em um volume canônico
  - Manter a relação entre o volume original e a cena
- Volume paralelo
  - Precisa de um translação, uma rotação e uma escala

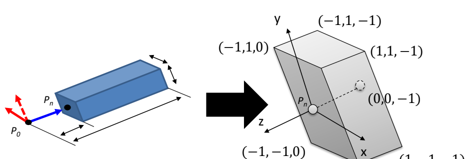


10

10

Transformação de Normalização

- A transformação 4x4 composta é chamada de transformação de normalização
- A inversa que transforma o canônico em arbitrário é chamada de transformação de visualização
- Lembrar que a câmera é somente um modelo
  - Transformação deve ser aplicada a cada vértice da cena



http://cs.brown.edu/courses/cs123/demos/camera/

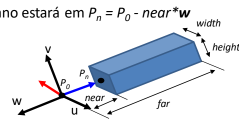
11

11

Transformação de Normalização

- Nosso objetivo é fazer os eixos ***u***, ***v*** e ***w*** do sistema de coordenadas da câmera coincidirem respectivamente com os eixos ***x***, ***y***, e ***z*** do sistema de coordenadas do mundo
- Translação
  - Mover a câmera para que o centro do plano de corte frontal esteja na origem
  - Dado uma câmera na posição P_o, o eixo ***w***, e as distâncias dos planos de corte frontal, *near,* e traseiro, *far*
    - O centro do plano estará em $P_n = P_0 - near*\boldsymbol{w}$

$$T_{trans} = \begin{bmatrix} 1 & 0 & 0 & -P_{nx} \\ 0 & 1 & 0 & -P_{ny} \\ 0 & 0 & 1 & -P_{nz} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



12

12

Transformação de Normalização

- Rotação
  - Precisamos achar a transformação que leve do mundo para a câmera
  - Porém, temos somente os vetores ***u***, ***v*** e ***w*** descritos no mundo
    - O que nos dá a transformação da câmera para o mundo

$$\begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix}$$

  - Solução: ache a inversa (transposta no caso da rotação)

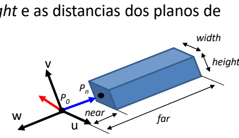
$$\boldsymbol{R_{rot}} = \begin{bmatrix} u_x & u_y & u_z \\ v_x & v_y & v_z \\ w_x & w_y & w_z \end{bmatrix} \xrightarrow[\text{homogêneas}]{\text{Coordenadas}} \boldsymbol{R_{rot}} = \begin{bmatrix} u_x & u_y & u_z & 0 \\ v_x & v_y & v_z & 0 \\ w_x & w_y & w_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

13

13

Transformação de Normalização

- Escala
  - Queremos ***x*** e ***y*** entre -1 e 1 e ***z*** de 0 a -1
  - O volume já está na origem e apontando para ***–z***
  - Dado a largura *width*, altura *height* e as distancias dos planos de corte *near* e *far*



$$\boldsymbol{S_{xyz}} = \begin{bmatrix} \frac{2}{width} & 0 & 0 & 0 \\ 0 & \frac{2}{hegiht} & 0 & 0 \\ 0 & 0 & \frac{1}{far - near} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

14

14

Transformação de Normalização

- A transformação de normalização do volume paralelo
  - É uma composição de $\boldsymbol{S_{xyz}R_{rot}T_{trans}}$

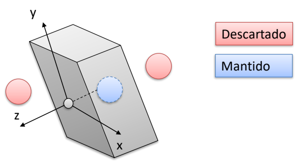
$$\begin{bmatrix} \frac{2}{width} & 0 & 0 & 0 \\ 0 & \frac{2}{height} & 0 & 0 \\ 0 & 0 & \frac{1}{far-near} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_x & u_y & u_z & 0 \\ v_x & v_y & v_z & 0 \\ w_x & w_y & w_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -P_{nx} \\ 0 & 1 & 0 & -P_{ny} \\ 0 & 0 & 1 & -P_{nz} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

15

15

Recorte

- Antes de projetar para 2D
  - Os objetos fora do volume de visualização são eliminados
  - Esse passo é feito mais facilmente no volume canônico
  - Objetos interceptando os planos
    - Devem ser cortados parcialmente



16

16

Projetar para 2D

- Como projetar um objeto para 2D?
  - Se quer projetar um ponto (x, y, z) basta eliminar z e ficar com (x,y)

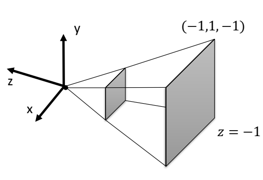
$$\begin{bmatrix}1 & 0 & 0 & 0\\0 & 1 & 0 & 0\\0 & 0 & 0 & 0\\0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}=\begin{bmatrix}x\\y\\0\\1\end{bmatrix}$$

17

17

O Volume de Visualização Canônico Perspectivo

- Começa na origem
  - posição = (0,0,0)
- Olha ao longo do eixo negativo
  - Look vector = (0,0,-1)
- Up está em y
  - Up vector = (0,1,0)
- Janela de visualização determinada pelo plano de corte traseiro
  - -1 a 1 na direções ***x*** e ***y***
- Planos de corte
  - Frontal z = c = -near/far (explicado de posteriormente)
  - Traseiro z = -1



18

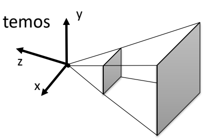
18

Transformação de Normalização

- Translação e Rotação
  - Similar ao volume paralelo
  - Translação é mais fácil, pois é direto da posição da câmera
  - Rotação é a mesma

$$\boldsymbol{T_{trans}} = \begin{bmatrix} 1 & 0 & 0 & -P_{0x} \\ 0 & 1 & 0 & -P_{0y} \\ 0 & 0 & 1 & -P_{0z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{R_{rot}} = \begin{bmatrix} u_x & u_y & u_z & 0 \\ v_x & v_y & v_z & 0 \\ w_x & w_y & w_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Após esses passos temos



19

19

Transformação de Normalização

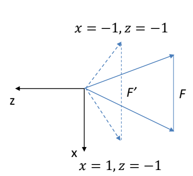
- Escala
  - Deve ser feita para colocar o plano de corte traseiro em
    - z = -1 e com x e y indo de -1 a 1
  - Em z é parecida com a do volume paralelo
    - Contudo, plano frontal não vai para z = 0
  - Passos
    - Ache a largura e altura do plano de corte traseiro (far plane)
    - Use os ângulos de abertura θ_w e θ_h e a distância do plano
    - Monte a matriz de escala $\boldsymbol{S_{xyz}}$ apropriada

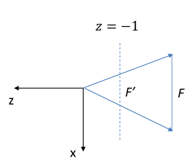
20

20

Transformação de Normalização

- Escala
  - Objetivo: Escalar para as linhas pontilhadas
  - Escale ao longo de ***z***
    - Pontos em *far* devem ir para -1
    - Então, $\boldsymbol{S_z} = 1/far$



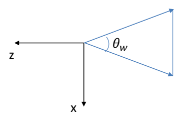


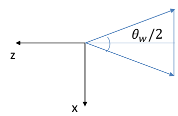
21

21

Transformação de Normalização

- Escala
  - Escale ao longo de ***x***
    - Dividir pelo tamanho do volume na direção ***x***
    - Usando o ângulo de abertura em x, θ_w,ache a largura do plano traseiro





22

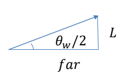
22

Transformação de Normalização

- Escala
  - Escale ao longo de ***x***
    - Divida pelo tamanho do volume na direção ***x***
    - Usando o ângulo de abertura em ***x***, θ_w,ache a largura do plano traseiro

$$\frac{L}{far} = \tan\left(\frac{\theta_w}{2}\right) \rightarrow L = far\tan\left(\frac{\theta_w}{2}\right)$$

$$S_x = \frac{1}{far\tan\left(\frac{\theta_w}{2}\right)}$$



23

23

Transformação de Normalização

- Escala
  - Escale ao longo de ***y***
    - Análogo a ***x***
    - Usando o ângulo de abertura em ***y***, θ_h,ache a altura do plano traseiro

$$S_y = \frac{1}{far\tan\left(\frac{\theta_h}{2}\right)}$$

  - Finalmente, monte a matriz

$$\boldsymbol{S_{xyz}} = \begin{bmatrix} \frac{1}{far\tan\left(\frac{\theta_w}{2}\right)} & 0 & 0 & 0 \\ 0 & \frac{1}{far\tan\left(\frac{\theta_h}{2}\right)} & 0 & 0 \\ 0 & 0 & \frac{1}{far} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

24

24

Transformação de Normalização

- A transformação de normalização do volume perspectivo
  - Até agora, tem a mesma forma do volume paralelo

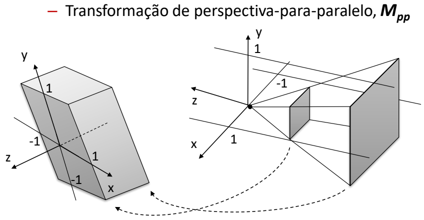
$$\boldsymbol{S_{xyz}R_{rot}T_{trans}} = \begin{bmatrix} \frac{1}{far\tan\left(\frac{\theta_w}{2}\right)} & 0 & 0 & 0 \\ 0 & \frac{1}{far\tan\left(\frac{\theta_h}{2}\right)} & 0 & 0 \\ 0 & 0 & \frac{1}{far} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_x & u_y & u_z & 0 \\ v_x & v_y & v_z & 0 \\ w_x & w_y & w_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -P_{0x} \\ 0 & 1 & 0 & -P_{0y} \\ 0 & 0 & 1 & -P_{0z} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

25

25

Perspectiva e Projeção

- Agora temos o volume canônico perspectivo
- Projetar esse volume para 2D é mais difícil do que no paralelo
- Vamos tornar o problema mais simples
  - Preservando a profundidade relativa
  - Transformação de perspectiva-para-paralelo, $\boldsymbol{M_{pp}}$



26

26

Perspectiva e Projeção

- A transformação perspectiva final será uma composição
  - $\boldsymbol{M_{pp}S_{xyz}R_{rot}T_{trans}}$
- Passo a passo
  - Vamos acompanhar o ponto P_n, centro do plano frontal

    $P_n = P_0 - near * w$
  - Ele será movido para sua nova posição

    $P_n' = \boldsymbol{S_{xyz}R_{rot}T_{trans}}P_n$
  - No eixo negativo, digamos:

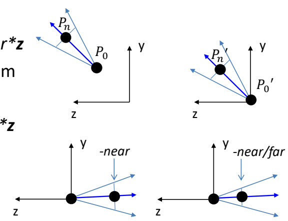
    $P_n' = \begin{pmatrix} 0 & 0 & c \end{pmatrix}$
  - Qual é o valor de c?

27

27

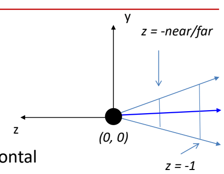
Perspectiva e Projeção

- P_0 é movido para a origem
- P_n é rotacionado para $-near$***z**
- A escala em ***x*** e ***y*** não afetam
- A escala em ***z***
  - Move P_n para *(-near/far)****z***
  - Então *c= -near/far*



Perspectiva e Projeção

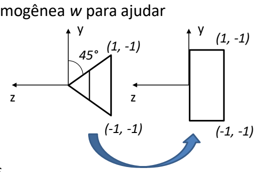
- Note que o plano traseiro
  - Já está com o tamanho correto
  - Já esta na posição correta
- O plano frontal foi parar em
  - *–near/far*
- Precisamos transformar o plano frontal
  - Para z = 0
- Linhas passando pela origem
  - Dever se tornar paralelas a ***z***



Perspectiva e Projeção

- Como montar a matriz M_{pp}?
  - Pontos nas linhas passando pela origem
    - Basta dividir pela distância em ***z***
  - Pode-se usar a coordenada homogênea *w* para ajudar
    - x' = x/-z e y'=y/-z

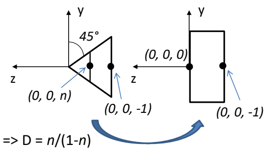
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ A & B & C & D \\ 0 & 0 & -1 & 0 \end{bmatrix}$$ Incógnitas



  - Resta descobrir as 4 incógnitas

Perspectiva e Projeção

- Não existe cisalhamento nessa transformação
  - Então A e B são zero
- Podemos achar C e D usando dois pontos conhecidos
  - $(0,0,-1,1)^T = \boldsymbol{M_{pp}}(0,0,-1,1)^T$
  - $(0,0,0,1)^T = \boldsymbol{M_{pp}}(0,0,-n,1)^T$
  - Em que, $n = -c = near/far$
- Resolvendo
  - $-C + D = -1$
    - => $D = C - 1$ => $D = 1/(1-n) - 1$ => $D = n/(1-n)$
  - $-Cn + D = 0$
    - => $-Cn + C - 1 = 0$ => $C = 1/(1-n)$



$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{1-n} & \frac{n}{1-n} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

31

31

Perspectiva e Projeção

- Após a aplicação da transformação perspectiva
  - Será necessário dividir por w
  - Isso resulta em

$$x' = -\frac{x}{z} \quad y' = -\frac{y}{z}$$

    - E uma transformação não linear em z:

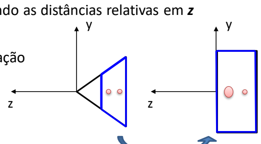
$$z' = \frac{c - z}{z(1 + c)}$$

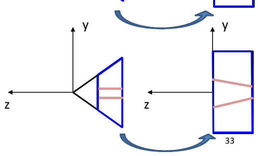
32

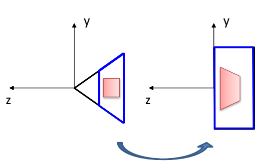
32

Perspectiva e Projeção

- Exemplos práticas
  - Quanto mais próximo do plano frontal um objeto está
    - Mais ele é escalado preservando as distâncias relativas em $\boldsymbol{z}$
  - Linhas paralelas
    - Convergem após a transformação





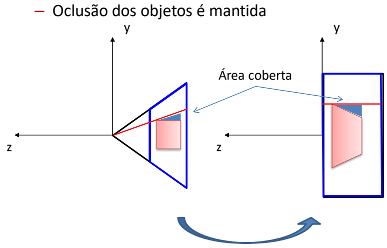


33

33

Perspectiva e Projeção

- Exemplos práticas
  - Oclusão dos objetos é mantida



34

34

Perspectiva e Projeção

- Transformação perspectiva causa compressão do ***z***
  - Quanto mais próximos do plano de corte traseiro
    - Mais o ***z*** fica comprimido
- Vamos olhar um caso geral

Homogeneização

$$M_{pp}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&\frac{1}{1+c}&\frac{-c}{1+c}\\0&0&-1&0\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}=\begin{bmatrix}x\\y\\\frac{z-c}{1+c}\\-z\end{bmatrix}\rightarrow\begin{bmatrix}-x/z\\-y/z\\\frac{c-z}{z+zc}\\1\end{bmatrix}$$

  - Analisando o novo valor de z, $z'=\dfrac{c-z}{z(1+c)}$
    - Se mantivermos *c* constante em -0.1
    - E plotarmos um gráfico

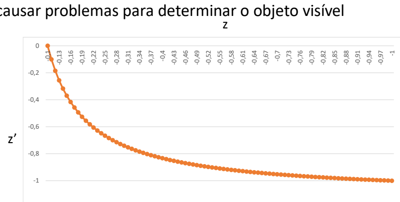
35

35

Perspectiva e Projeção

- Transformação perspectiva causa compressão do ***z***
  - Pode-se ver uma compressão dos valores próximos do plano de corte traseiro
  - Pode causar problemas para determinar o objeto visível

$z'=\dfrac{c-z}{z(1+c)}$



36

36

Perspectiva e Projeção

- Pode ser tentador colocar
  - O plano frontal de corte em zero
  - O plano traseiro de corte no infinito
- Porém, $c = -near/far$ tende a zero em qualquer dos dois casos
  - Isso resulta em: $z' = \frac{c - z}{(z - z * c)} = -\frac{z}{z} = -1$
- Se todos os planos caírem na mesma distância
  - Não teremos como identificar quem está na frente de quem

37

37

Transformação de Normalização Completa

- A transformação de normalização completa
  - Transforma do volume perspectivo
  - Para um volume canônico paralelo

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{1+c} & \frac{-c}{1+c} \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{far \tan\left(\frac{\theta_w}{2}\right)} & 0 & 0 & 0 \\ 0 & \frac{1}{far \tan\left(\frac{\theta_h}{2}\right)} & 0 & 0 \\ 0 & 0 & \frac{1}{far} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_x & u_y & u_z & 0 \\ v_x & v_y & v_z & 0 \\ w_x & w_y & w_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -P_{0x} \\ 0 & 1 & 0 & -P_{0y} \\ 0 & 0 & 1 & -P_{0z} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Após a transformação é necessário homogeneizar
  - Dividir por *w*
- Agora, assim como no volume paralelo
  - Basta eliminar a coordenada z

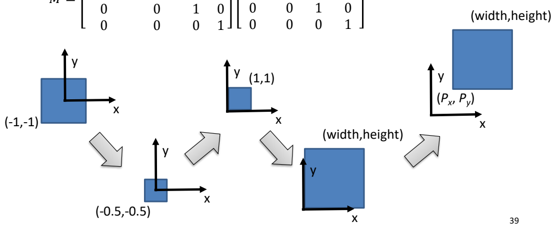
38

38

Transformação Window-to-Viewport

- O último passo é redimensionar os objetos
  - Para atender ao tamanho do viewport

$$M = \begin{bmatrix} width & 0 & 0 & P_x \\ 0 & height & 0 & P_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1/2 & 0 & 0 & 1/2 \\ 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



39

39

Resumo da Visualização

- Para todos os pontos da cena
  - Aplique as transformações de modelagem
  - Aplique as transformações da câmera
    - Transformar no volume canônico
  - Projete para o filme (2D)
  - Redimensione a janela de visualização para o viewport
  - Mapeie as cores dos pontos (x, y) da janela para os pixels (u, v) do viewport
- Alguns passos foram omitidos e serão vistos em aulas futuras
  - Recorte
  - Determinação da superfície visível
  - ...

40

40

Visualização em OpenGL

- Em OpenGL
  - A matriz *ModelView* engloba
    - Transformações de modelagem do mundo
    - Transformação da câmera
      - Rotação e Translação
  - A matriz *Projection* engloba
    - Escala
    - Perspectiva para paralela

41

41

Perguntas ?????

42

42