

Mapeamento Window-Viewport

Claudio Esperança
Paulo Roma Cavalcanti

Problema

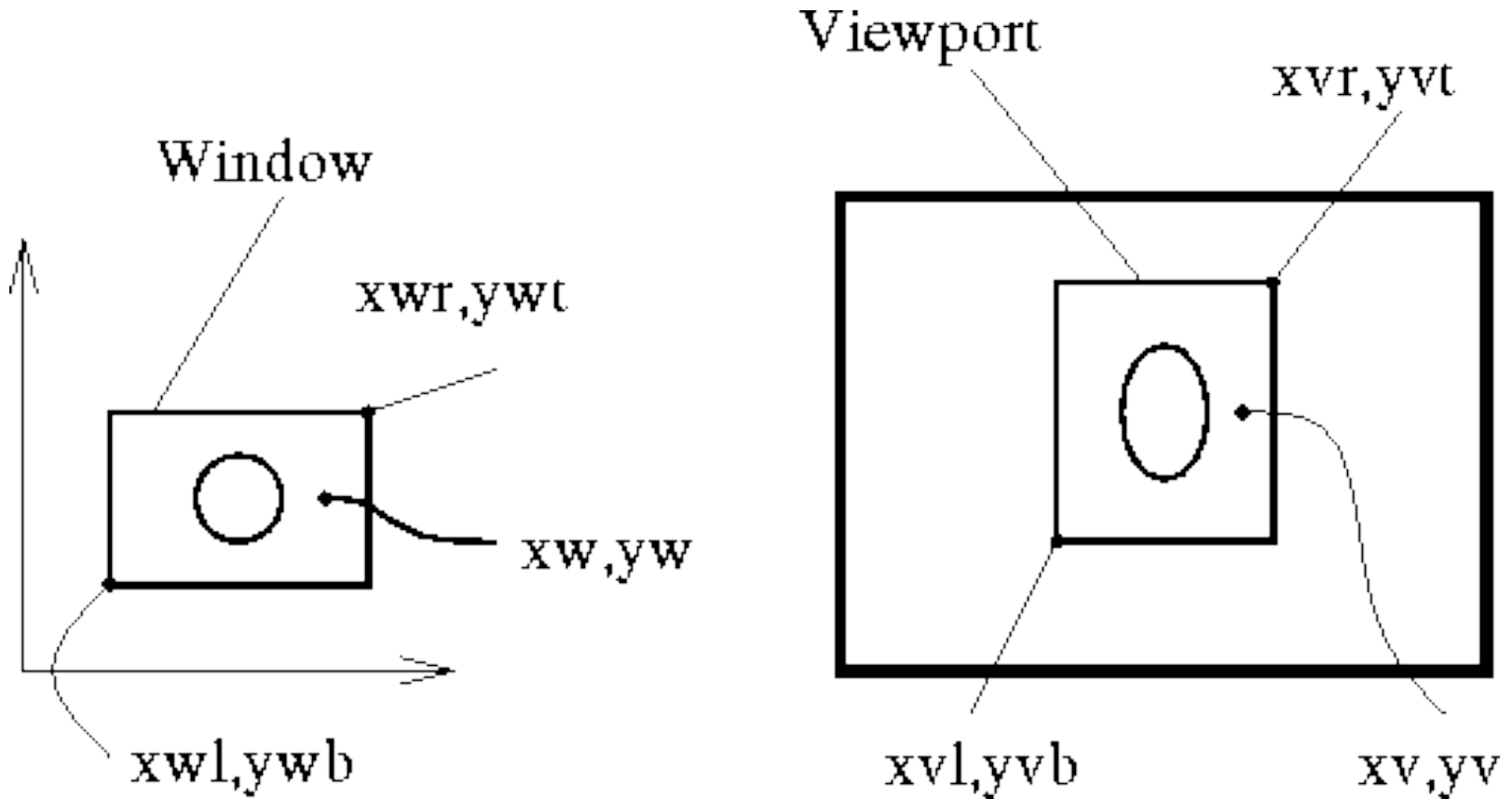
- Cena é 3D, mas eventualmente será projetada para 2D.
- Cena 2D está num plano infinito, mas o dispositivo possui uma área visível retangular finita.
- O que fazer?

Resposta: mapear uma região retangular da cena 2D para o dispositivo.

Definições

- Window:
 - Região retangular de interesse na cena.
- Viewport:
 - Região retangular no dispositivo.
- Normalmente, ambos os retângulos estão alinhados com o sistema de coordenadas.

Window - Viewport



Matemática – Regra de três

- Ponto (x_w, y_w) da Window mapeia sobre um ponto (x_v, y_v) da Viewport.
- Window tem cantos (x_{wl}, y_{wb}) e (x_{wr}, y_{wt}) .
- Viewport tem cantos (x_{vl}, y_{vb}) e (x_{vr}, y_{vt}) .
- Comprimento e altura da Window são L_w e H_w .
- Comprimento e altura da Viewport são L_v e H_v .

Manter a Proporção

Mapeie cada coordenada de acordo com:

$$\frac{\Delta x_w}{L_w} = \frac{\Delta x_v}{L_v}, \quad \frac{\Delta y_w}{H_w} = \frac{\Delta y_v}{H_v}.$$

Coordenada na Viewport

Para mapear x_w para x_v (y_v é equivalente):

$$\frac{x_w - x_{wl}}{L_w} = \frac{x_v - x_{vl}}{L_v} \Rightarrow x_v = \frac{L_v}{L_w} (x_w - x_{wl}) + x_{vl}.$$

Razão de Aspecto

- Se a razão de aspecto não for a mesma, a imagem será distorcida:

$$\frac{H_w}{L_w} \neq \frac{H_v}{L_v}$$

Resumo

- Converta x_w para uma distância w a partir do canto inferior esquerdo da Window.
- Escale esta distância w para obter uma distância v .
- Adicione a v o canto da Viewport para obter x_v .
- Equivalente a uma **escala** mais uma **translação**.

Imagem 1024 x 768 (4:3)



Mesma Imagem 1024 x 614 (10:6)



Desentrelaçamento

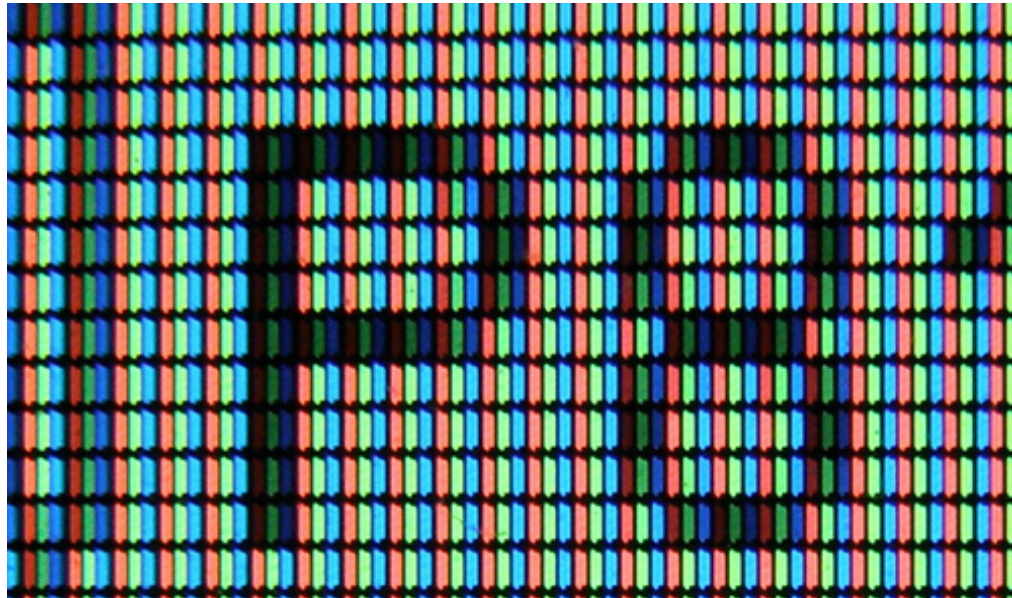


TV 4:3 e 16:9



Pixels

- Full HD: 1080 x 1920
- 4K: 2160 x 3840
- 8K: 4320 x 7680



Transformações Afim

- Transformações Afim = transformações lineares + translações
- Qualquer transformação linear fica determinada pelo efeito causado nos vetores da base de um espaço vetorial.
- são necessários apenas 3 pontos não colineares em 2D.

Mapeamento RPQ para unitário

- Logo, sabendo-se que o mapeamento do triângulo unitário $(0,0)$, $(1,0)$, $(0,1)$ no triângulo RPQ é dada pela matrix de transformação homogênea:

$$T = \begin{pmatrix} P_x - R_x & Q_x - R_x & R_x \\ P_y - R_y & Q_y - R_y & R_y \\ 0 & 0 & 1 \end{pmatrix}$$

Mapeamento RPQ to R'P'Q'

- e, usando-se o triângulo unitário como ponte, o mapeamento do triângulo RPQ, no frame A, sobre o triângulo R'P'Q', no frame B, é dado pela matrix:

$$T_{A \rightarrow B} = T^{-1} * \begin{pmatrix} P'_x - R'_x & Q'_x - R'_x & R'_x \\ P'_y - R'_y & Q'_y - R'_y & R'_y \\ 0 & 0 & 1 \end{pmatrix}$$

Mapeamento para viewport

- Em particular, para mapear o retângulo alinhado com os eixos coordenados, com diagonal $(x_1, y_1), (x_2, y_2)$, basta fazer o terceiro ponto ser: (x_1, y_2)
- Facilmente implementável com Numpy

Rio - Calgary via New York

The screenshot displays the APM Planner v2.0.26 interface. The main map shows a flight path starting at waypoint 0 in Rio de Janeiro, Brazil, and ending at waypoint 12 in Calgary, Canada, with a stopover at waypoint 1 in New York City. The path is marked with red diamonds and numbers. The interface includes a menu bar (File, Communication, Views, Tool Widgets, Help), a toolbar with icons for flight data, flight plan, initial setup, configuration, graphs, and donate. A status bar at the top right shows 'MAV ID: No Devices' and '115200'. The map has a compass and coordinate fields (LAT, LON, FIX, SATS, HDOP, ZOOM: 3). Below the map is a 'Mission Plan' table with columns for waypoint number, name, altitude, latitude, longitude, heading, speed, and time. The table contains three rows: 0 HOME, 1 Waypoint, and 2 Waypoint. At the bottom, there are buttons for 'Load WPs' and 'Save WPs'.

Waypoint	Name	Alt	Lat	Lon	Heading	Speed	Time
0	HOME	Abs Alt	lat: -22.9068470°	lon: -43.1728970°	-180°	5.00 m	0.00 s
1	Waypoint	Rel Alt	lat: -16.9832160°	lon: -44.9485160°	19°	5.00 m	0.00 s
2	Waypoint	Rel Alt	lat: -11.9317140°	lon: -46.5569270°	20°	5.00 m	0.00 s