CCI-36 – Computação Gráfica

Textura

Instituto Tecnológico de Aeronáutica

Prof. Carlos Henrique Q. Forster – Sala 121 IEC

forster@ita.br

ramal 5981

Tópicos da aula

- Interpolação de pixels
- Mipmaps
- Coordenadas de textura e normais
- Warping de imagens
- Mapeamento de textura

Livro para acompanhar essa aula

Interpolação de pixels

Sejam os pixels de valores de intensidade E(i,j), E(i+1,j), E(i+1,j+1) e E(i,j+1).

Pode ser necessário obter um valor de intensidade relativo a um par de coordenadas reais (x,y) ao invés das inteiras (i,j).

O valor pode ser aproximado por

$$
\begin{aligned}
E(x,y) = {} & \left[E(i,j)(i+1-x) + E(i+1,j)(x-i)\right]\ (j+1-y) \\
& + \left[E(i,j+1)(i+1-x) + E(i+1,j+1)(x-i)\right](y-j)
\end{aligned}
$$

(interpolação bi-linear)

Da mesma forma pode-se fazer uma interpolação bi-cúbica considerando uma grade de 16 pixels e polinômios de Lagrange.

Interpolação cúbica é dada por

$$E(x) = \frac{1}{6}\begin{bmatrix} E(i-1) & E(i) & E(i+1) & E(i+2) \end{bmatrix} \begin{bmatrix} -1 & 3 & -2 & 0 \\ 3 & -6 & -3 & 6 \\ -3 & 3 & 6 & 0 \\ 1 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} (x-i)^3 \\ (x-i)^2 \\ (x-i) \\ 1 \end{bmatrix}$$

Essa matriz é obtida do sistema de equações para determinar os coeficientes dos polinômios da forma $E(x) = ax^3 + bx^2 + cx + d$ para $E(-1), E(0), E(1), E(2)$.

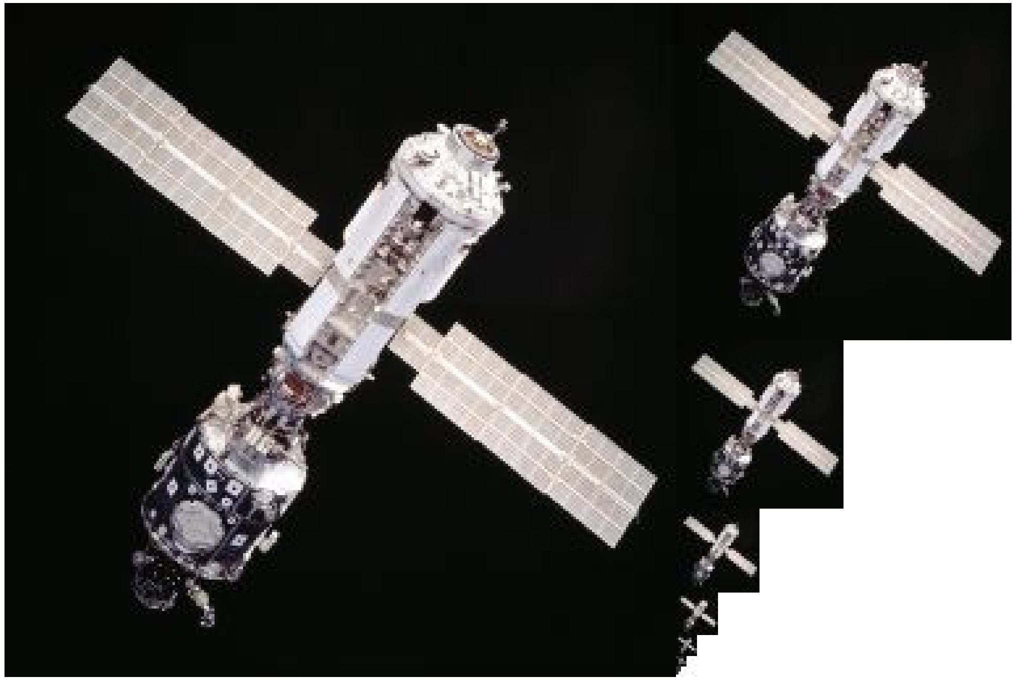
MIP-maps

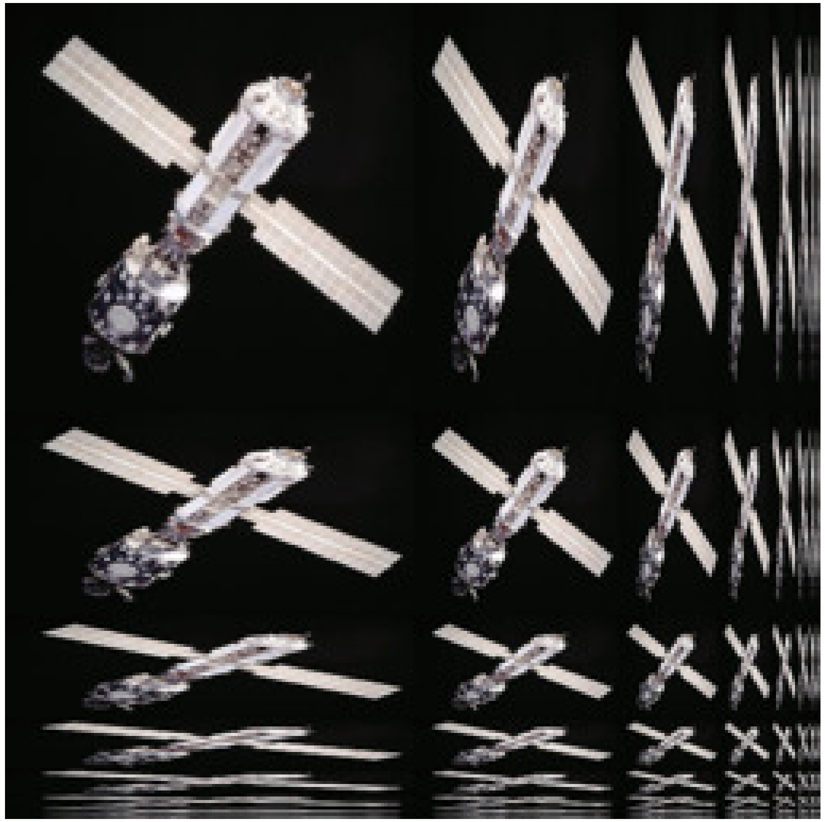
Quando muitos pixels são acumulados sobre um pixel da imagem resultante, utiliza-se uma de um conjunto de imagens filtradas.

Para a imagem a ser mapeada é feita uma representação em multi-escala que ocupa 4/3 do tamanho da imagem original. Cada nível possui ¼ dos pixels do nível anterior e o valor de intensidade dos pixels é calculado como uma imagem filtrada por um núcleo Gaussiano sobre os pixels do nível anterior.

Pode-se usar interpolação bi-linear, escolhendo-se a resolução apropriada na pirâmide de imagens, ou usar interpolação trilinear, considerando dois níveis de resolução.

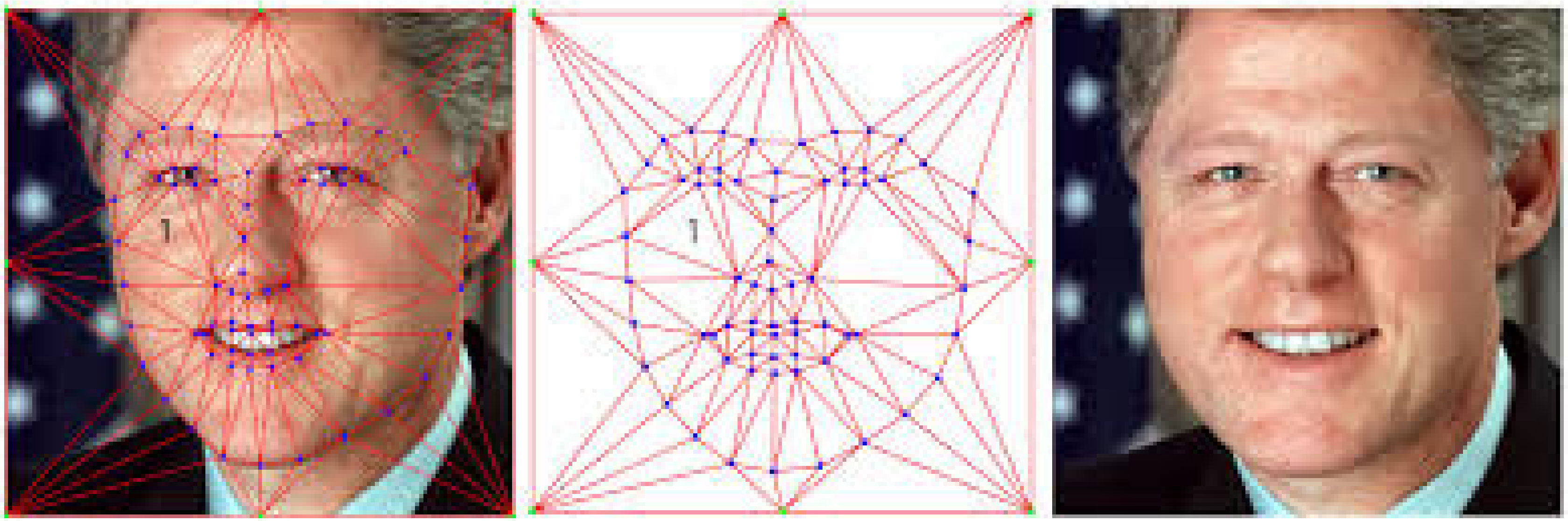
Mipmap isotrópico e anisotrópico



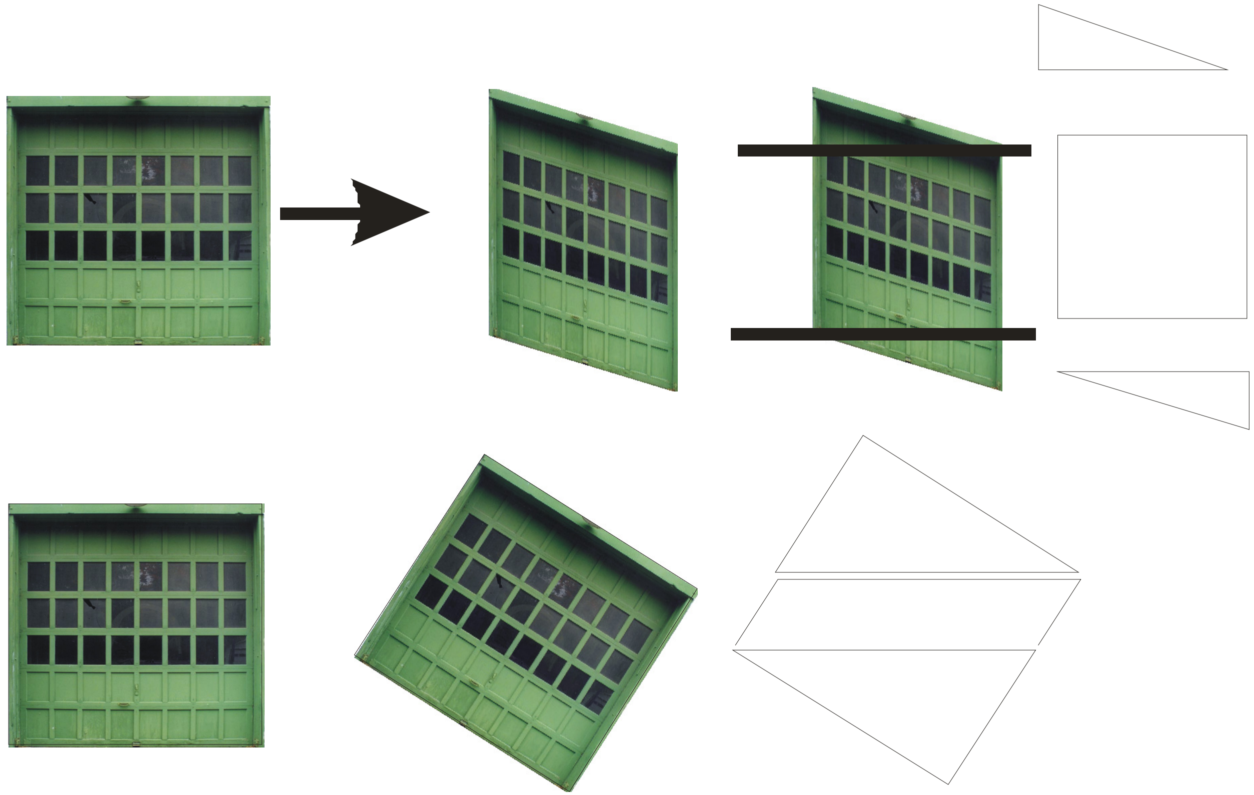


Warping

O segredo do warping de imagens é utilizar a transformação inversa. Para cada pixel da região da imagem alvo, procura-se o pixel correspondente da imagem original, obtendo o valor por interpolação ou escolhendo o nível adequado em um MIP-map.



A região destino da transformação pode não ser retangular e não estar alinhada com os eixos. É interessante decompô-la em uma união de trapézios alinhados com um dos eixos para percorrer os pixels da região.



Base Afim

Encontrar a transformação afim em 2D que mapeia os três pontos P1, P2 e P3 nos pontos Q1, Q2 e Q3.

$$Q_1 = \begin{bmatrix} q_{x1} \\ q_{y1} \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_{x1} \\ p_{y1} \\ 1 \end{bmatrix}$$

Para cada par de pontos PQ tenho duas equações.

Para encontrar as 6 incógnitas a,b,c,d,e,f, preciso de 3 pares de pontos.

Lembrando a solução linear

$$\begin{bmatrix} p_{x1} & p_{y1} & 1 \\ p_{x2} & p_{y2} & 1 \\ p_{x3} & p_{y3} & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} q_{x1} \\ q_{x2} \\ q_{x3} \end{bmatrix} \quad e \quad \begin{bmatrix} p_{x1} & p_{y1} & 1 \\ p_{x2} & p_{y2} & 1 \\ p_{x3} & p_{y3} & 1 \end{bmatrix} \begin{bmatrix} d \\ e \\ f \end{bmatrix} = \begin{bmatrix} q_{y1} \\ q_{y2} \\ q_{y3} \end{bmatrix}$$

Qualquer transformação afim em 2D pode ser representada como a transformação dos pontos (0,0), (0,1) e (1,0). Os três pontos resultantes dessa transformação formam uma base afim (se não colineares).

Quantos pontos são necessários em 3 dimensões?

$$P' = \begin{bmatrix} a & b & c & l \\ d & e & f & m \\ g & i & j & n \\ 0 & 0 & 0 & 1 \end{bmatrix} P$$

São doze incógnitas, mas cada ponto contribui com 3 equações. Assim, são necessários quatro pontos (não-coplanares) para formar uma base afim. Correspondem às transformações de (1,0,0), (0,1,0), (0,0,1) e (0,0,0).

Base Projetiva

Encontrar transformação projetiva em 2D que mapeia 4 pontos.

$$\begin{bmatrix} hx'_1 \\ hy'_1 \\ h \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & j & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$hx'_i = ax_i + by_i + c$$

$$hy'_i = dx_i + ey_i + f$$

$$h = gx_i + jy_i + 1$$

Substituindo h,

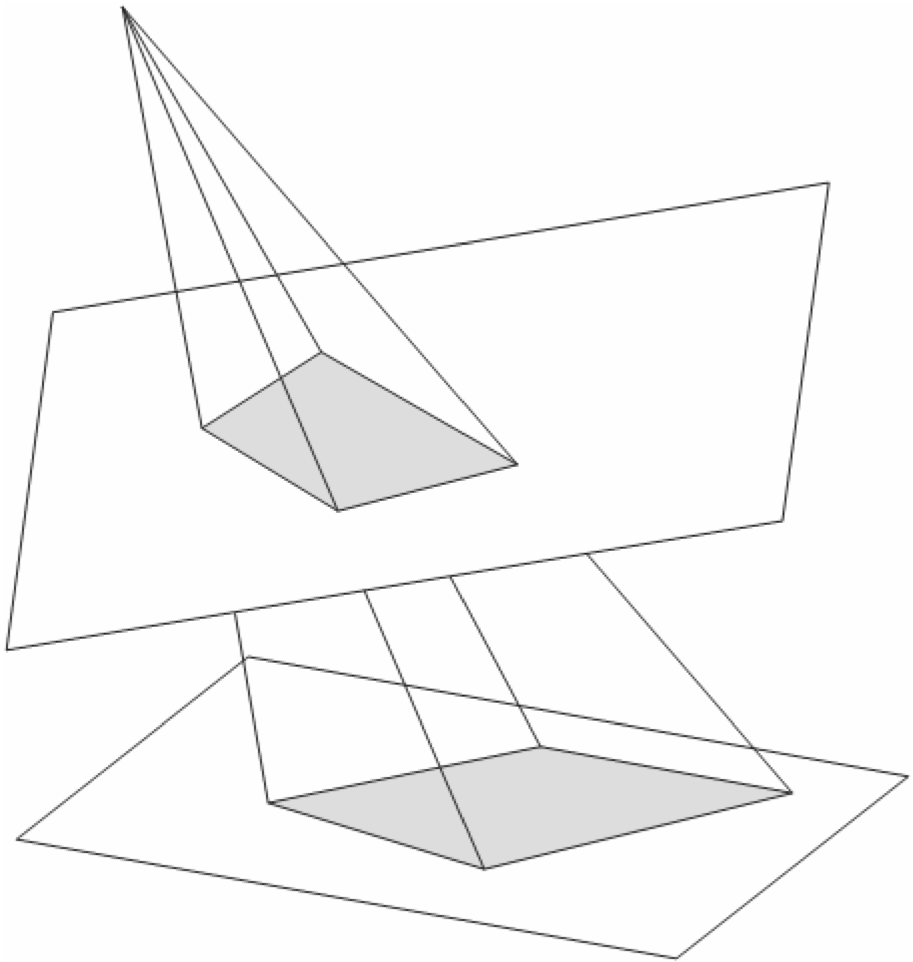
$$gx_i x'_i + jy_i x'_i + x'_i = ax_i + by_i + c$$

$$gx_i y'_i + jy_i y'_i + y'_i = dx_i + ey_i + f$$

Construímos o sistema 8x8, são duas equações para cada ponto.

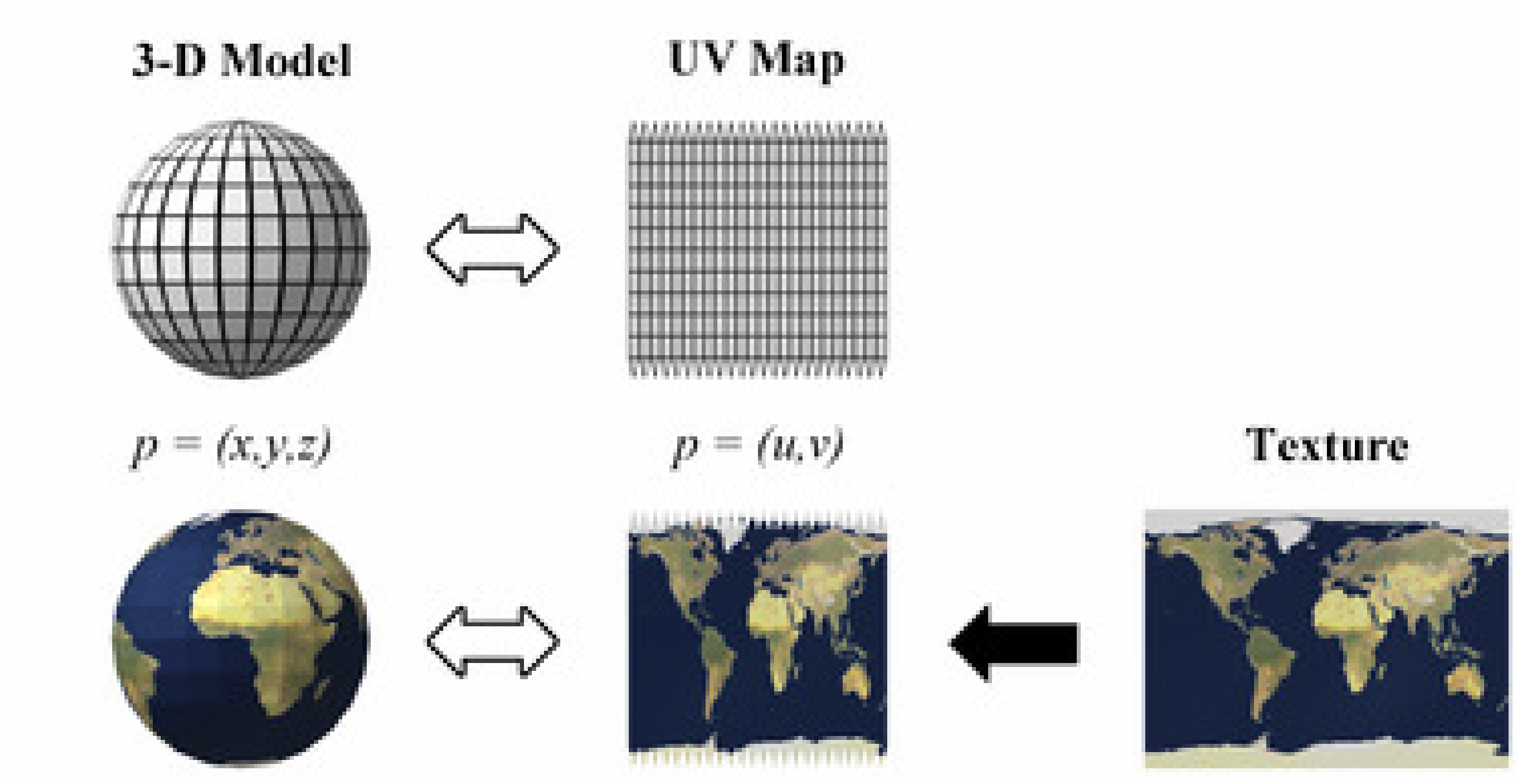
$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x'_1\,x_1 & y_1x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -y'_i\,x_i & y_1y'_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x'_2\,x_2 & y_2x'_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -y'_2\,x_2 & y_2y'_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x'_3\,x_3 & y_3x'_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -y'_3\,x_3 & y_3y'_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x'_4\,x_4 & y_4x'_4 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -y'_4\,x_4 & y_4y'_4 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ j \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \\ x'_4 \\ y'_4 \end{bmatrix}$$

Podemos formar uma base pela transformação dos pontos (1,0), (0,1) (0,0) e (1,1).



A transformação encontrada é uma transformação de plano para plano ou então de imagens planas para imagens planas

UV Map



Textura sólida – UVW mapping