

Relatório do Projeto de CG

João Pimentel (a80874) Rodolfo Silva (a81716)
Pedro Gonçalves (a82313)

Março 2019

Universidade do Minho
Mestrado Integrado em Engenharia Informática
Computação Gráfica
Grupo 25
2^a Fase
Transformações Geométricas

Conteúdo

1	Introdução	4
2	Transformações Geométricas	5
2.0.1	Translate	5
2.0.2	Rotate	6
2.0.3	Scale	14
3	Sistemas de Coordenadas	16
4	Composição de Transformações Geométricas	19
5	Sistema Solar Estático e Classes	22
6	Conclusões e Trabalho Futuro	25

Lista de Figuras

1	Translação	5
2	Rotação 2D do ponto p por um ângulo α	7
3	Rotação de p' por um ângulo β	8
4	Rotação em redor de um eixo arbitrário.	9
5	Rotação de \vec{v} para ser colocado no plano YZ	10
6	Alinhamento de \vec{v} com o plano YZ	11
7	Rotação para alinhar \vec{v} com o eixo Z	11
8	Alinhamento do vetor \vec{v} com o eixo Z	12
9	Final do passo 2 da Rotação	12
10	Desfazer rotações nos eixos X e Y	13
11	<i>Scaling</i> do ponto p para o ponto p'	15
12	Translação de ponto	16
13	Translação de sistema de coordenadas	17
14	Rotação em torno do eixo Z	17
15	Composição de translações	19
16	Composição de translações de sistemas de coordenadas	20
17	Rotação após redimensionamento não uniforme	20
18	Classe <i>Figura</i>	23
19	Sistema Solar estático	24

1 Introdução

O presente relatório consiste na descrição da segunda fase da implementação de um mini cenário gráfico 3D que representa um modelo estático do sistema solar.

Nesta etapa, o objetivo centrou-se em usar transformações geométricas, tais como *translate*, *rotate* e *scale*, de modo a criar cenários hierárquicas. Neste sentido, desenvolveu-se uma hierarquia que inclui o Sol, os planetas e as luas.

2 Transformações Geométricas

2.0.1 Translate

Uma translação define-se pelo movimento de um ou mais pontos, numa determinada direção e comprimento. Aplicando uma translação a todos os pontos que constituem um objeto, a posição deste objeto será alterada. Sendo assim, esta transformação geométrica será bastante útil para mover figuras, gerar cenários 3D, colocar a câmara em posição e realizar animações [1].

Em termos matemáticos, uma translação pode ser descrita pela soma de um vetor com os valores de translocação a um ponto, originando, deste modo, um novo ponto, como se vê na equação 1 [1].

$$p' = p + \vec{v} \quad (1)$$

Em termos gráficos, na Figura 1, o ponto p sofre uma translação pelo vetor \vec{v} , obedecendo à equação anterior.

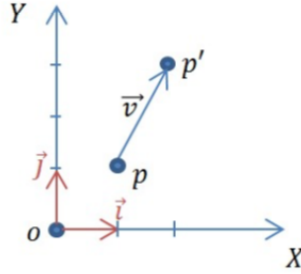


Figura 1 - Translação [1].

De forma simples é deduzível a equação 2 [1].

$$p' = (p_x + v_x, p_y + v_y, 1) \quad (2)$$

Aplicando a linha de pensamento a um sistema de coordenadas, a origem e os vetores unitários, tem-se a equação 3 [1].

$$p' = p + \vec{v} = p_x \vec{i} + p_y \vec{j} + o + v_x \vec{i} + v_y \vec{j} \quad (3)$$

Assumindo que o' se trata da soma da origem com as coordenadas do

vetor \vec{v} , obtém-se, na forma matricial [1]:

$$p' = \begin{bmatrix} \vec{i} & \vec{j} & o' \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix} \quad (4)$$

Considerando que \vec{i} possui o conjunto de coordenadas (1,0,0), \vec{j} (0,1,0) e o (0,0,1), o ponto p' pode ser descrito por [1]:

$$p' = \begin{bmatrix} 1 & 0 & v_x \\ 0 & 1 & v_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix} = \begin{bmatrix} p_x + v_x \\ p_y + v_y \\ 1 \end{bmatrix} \quad (5)$$

Deste modo, está comprovado que, como era expectável, um ponto p , com coordenadas $(p_x, p_y, 1)$, quando sofre uma translação por um vetor $\vec{v} = (v_x, v_y, 0)$, o resultado é $(p_x + v_x, p_y + v_y, 1)$ [1].

É de realçar que a matriz 3x3, presente na equação 5, pode ser usada para retratar uma translação cartesiana num ponto 2D [1].

Aplicando uma translação a um vetor $\vec{t} = (t_x, t_y, 0)$, obtém-se o mesmo vetor, já que os vetores são caracterizados pelo seu comprimento e direção, não estando dependentes de um ponto fixo. Assim, como nenhuma das propriedades possuídas pelos vetores é afetada pelas translações, tem-se a equação 6 [1].

$$p' = \begin{bmatrix} 1 & 0 & v_x \\ 0 & 1 & v_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ 0 \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \\ 0 \end{bmatrix} \quad (6)$$

Já numa perspectiva 3D, a forma geral de representação matricial de uma translação é [1]:

$$T = \begin{bmatrix} 1 & 0 & 0 & v_x \\ 0 & 1 & 0 & v_y \\ 0 & 0 & 1 & v_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

Sendo assim, pela equação 7, tem-se que $p' = Tp$ [1].

Por fim, utilizando $-\vec{v}$ para construir a matriz T , é obtida a translação inversa da obtida com \vec{v} . Dito isto, $T^{-1} = -T$ [1].

2.0.2 Rotate

2.0.2.1 Eixos Principais

Num cenário 2D, todas as rotações são realizadas no sentido inverso dos ponteiros do relógio (Figura 2). Quando se trabalha num cenário 3D diz-se

que as rotações são aplicadas no eixo de rotação. Por exemplo, uma rotação no plano XY consiste numa rotação sobre o eixo do Z . Neste caso, seguindo a regra da mão direita, ter-se-á uma direção de rotação positiva [1].

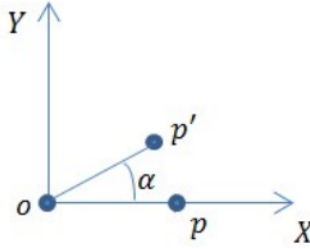


Figura 2 - Rotação 2D do ponto p por um ângulo α [1].

Considerando o ponto $p = (p_x, 0)$, então a rotação deste por um ângulo α é dada por [1]:

$$p' = (p_x \times \cos \alpha, p_x \times \sin \alpha) \quad (8)$$

Caso se pretenda rodar o ponto p' por um ângulo β de modo a obter um ponto p'' [1]:

$$p'' = (p_x \times \cos \alpha + \beta, p_x \times \sin \alpha + \beta) \quad (9)$$

Reescrevendo a parcela X de p'' utilizando as regras básicas de trigonometria ter-se-á a seguinte solução [1]:

$$p''_x = p_x \times \cos \alpha \cos \beta - p_x \times \sin \alpha \sin \beta \quad (10)$$

Considerando $p'_x = p_x \times \cos \alpha$ e $p'_y = p_x \times \sin \alpha$, a equação 10 passa a ser definida por [1]:

$$p''_x = p'_x \times \cos \beta - p'_y \times \sin \beta \quad (11)$$

Resolvendo em ordem à segunda parcela de p'' , $p''_y = p_x \times \sin \alpha + \beta$, e aplicando novamente as regras de trigonometria [1]:

$$p''_y = p_x \times \sin \alpha \cos \beta + p_x \times \cos \alpha \sin \beta \quad (12)$$

O resultado da combinação da equação 8 com a anterior apresenta-se abaixo [1].

$$p''_y = p'_x \times \sin \beta + p'_y \times \cos \beta \quad (13)$$

Assim, o ponto p'' pode ser definido do seguinte modo [1]:

$$p'' = (p'_x \times \cos \beta - p'_y \times \sin \beta, p'_x \times \sin \beta + p'_y \times \cos \beta) \quad (14)$$

O ponto p'' , na forma matricial, define-se por [1]:

$$p'' = \begin{bmatrix} \cos \beta & -\sin \beta & 0 \\ \sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p'_x \\ p'_y \\ 1 \end{bmatrix} \quad (15)$$

A demonstração gráfica que se apresenta na Figura 3 resume todo o processo até agora descrito [1].

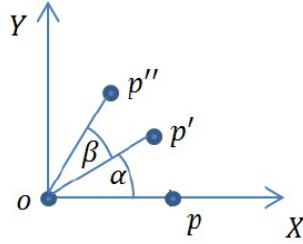


Figura 3 - Rotação de p' por um ângulo β [1].

As rotações em 3D no eixo do Z são semelhantes à rotação 2D apresentada anteriormente. Tal deve-se ao facto de os componentes Z e a origem não serem afetados pela rotação e por estes também não influenciarem os valores finais de x e y . Deste modo, uma rotação no eixo do Z pode ser definida por [1]:

$$R_{Z,\beta} = \begin{bmatrix} \cos \beta & -\sin \beta & 0 & 0 \\ \sin \beta & \cos \beta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (16)$$

De igual modo, as matrizes correspondentes às rotações sobre os eixos do X e Y , respetivamente, por [1]:

$$R_{X,\beta} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \beta & -\sin \beta & 0 \\ 0 & \sin \beta & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (17)$$

$$R_{Y,\beta} = \begin{bmatrix} \cos \beta & 0 & \sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (18)$$

O matriz inversa de uma rotação sobre o eixo do Z , por exemplo, por um ângulo α é facilmente calculada aplicando as regras trigonométricas, tal como se evidencia na próxima equação [1].

$$R_{Z,\beta}^{-1} = \begin{bmatrix} \cos -\beta & -\sin -\beta & 0 & 0 \\ \sin -\beta & \cos -\beta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos \beta & \sin \beta & 0 & 0 \\ -\sin \beta & \cos \beta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (19)$$

2.0.2.2 Eixos Arbitrários

Considere-se, agora, que é pretendido executar uma rotação sobre um eixo arbitrário com direção definida por \vec{v} .

Como se vê na Figura 4, o ponto q é obtido pela rotação de p em redor de \vec{v} , por um ângulo θ . Deste modo, pretende-se obter uma matriz de rotação do tipo [1]:

$$q = R_{v,\theta}p \quad (20)$$

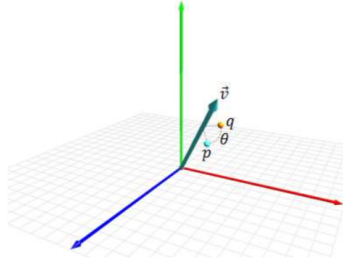


Figura 4 - Rotação em redor de um eixo arbitrário [1].

Sendo assim, para obter a matriz de rotação sobre um ponto 3D por um ângulo θ em redor de um eixo arbitrário \vec{v} , baseado no conhecimento até então adquirido, pode-se dividir o processo em três fases:

1. Rodar o vetor \vec{v} de modo a que se alinhe com um dos eixos principais;

2. Rodar o ponto em redor do eixo selecionado por θ ;
3. Desfazer as rotações feitas no primeiro passo [1].

A menos que o vetor \vec{v} coincida com um dos eixos principais, o passo 1 requer a execução de duas fases. Seja considerado o caso geral em que \vec{v} não coincide com nenhum eixo, como é visível na Figura 5.

Primeiramente, é executada uma rotação de modo a que \vec{v} fique no plano YZ ou XZ . Considerando que se pretende colocar no primeiro plano, uma rotação sobre o eixo Y é necessária. Para determinar a matriz, é crucial ter em atenção o ângulo β [1].

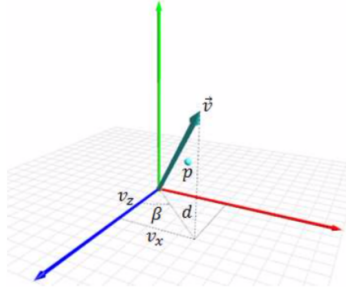


Figura 5 - Rotação de \vec{v} para ser colocado no plano YZ [1].

Considere-se que $\vec{v} = (v_x, v_y, v_z, 0)$. A projeção de \vec{v} no plano XZ é $\vec{v}_{xz} = (v_x, 0, v_z, 0)$ e a norma do mesmo é $d = \sqrt{v_x^2 + v_z^2}$. Além disso, com o auxílio da Figura 5, sabe-se que $\cos \beta = \frac{v_z}{d}$ e $\sin \beta = \frac{v_x}{d}$ [1].

No entanto, as rotações são executadas no sentido contrário dos ponteiros do relógio, sendo estas baseadas na regra da mão direita. A rotação em questão é aplicada de acordo com o sentido dos ponteiros do relógio, tendo que se considerar a rotação pelo ângulo $-\beta$, assim [1]:

$$\cos -\beta = \frac{v_z}{d}; \sin -\beta = -\frac{v_x}{d} \quad (21)$$

Deste modo, a matriz R_y e a sua inversa, que será necessária para o passo 3, são apresentadas na equação 22 [1].

$$R_y = \begin{bmatrix} \frac{v_z}{d} & 0 & -\frac{v_x}{d} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{v_x}{d} & 0 & \frac{v_z}{d} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; R_y^{-1} = \begin{bmatrix} \frac{v_z}{d} & 0 & \frac{v_x}{d} & 0 \\ 0 & 1 & 0 & 0 \\ -\frac{v_x}{d} & 0 & \frac{v_z}{d} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (22)$$

Por conseguinte, obtém-se $\vec{v} = R_y \vec{v}$, estando o vetor \vec{v} está no plano YZ , como se pode constatar na Figura 6.

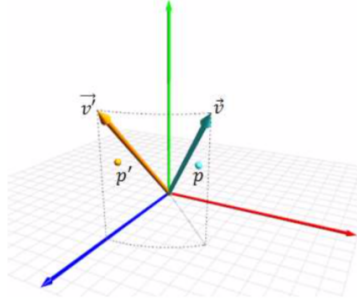


Figura 6 - Alinhamento de \vec{v} com o plano YZ [1].

Em seguida, uma rotação sobre o eixo X é necessária de modo a alinhar o eixo de rotação com o eixo Z . Esta rotação é representada por um ângulo α , sendo que $\cos \alpha = \frac{d}{v}$ e $\sin \alpha = \frac{v_y}{v}$, como se vê na Figura 7.

Assumindo que \vec{v} está normalizado implica que os cálculos anteriores não necessitem de ser efetuados. Note-se que, como apenas a direção do vetor e não a sua magnitude são relevantes para o resultado final, não haverá perda de generalização nos cálculos [1].

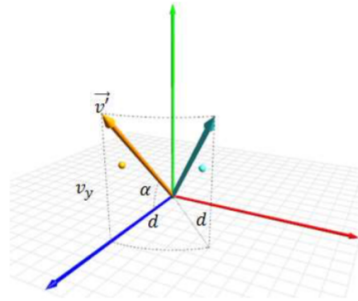


Figura 7 - Rotação para alinhar \vec{v} com o eixo Z [1].

Deste modo, a matriz para efetuar o pretendido e a sua inversa são [1]:

$$R_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & d & -v_y & 0 \\ 0 & v_y & d & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; R_x^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & d & v_y & 0 \\ 0 & -v_y & d & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (23)$$

Consequentemente, o primeiro passo do processo é finalizado, tendo-se as equações 24 e 25. Este processo de alinhamento do vetor \vec{v} com o eixo Z pode ser visto na Figura 8.

$$\vec{v}'' = R_x \vec{v}' = R_x R_y \vec{v} \quad (24)$$

$$p'' = R_x p' = R_x R_y p \quad (25)$$

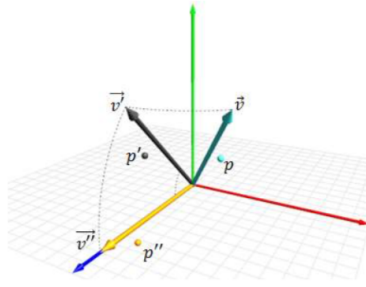


Figura 8 - Alinhamento do vetor \vec{v} com o eixo Z [1].

Posteriormente, já relativamente ao passo 2, a rotação em torno do eixo Z por um ângulo θ é deduzida através da equação 16, resultando na Figura 9.

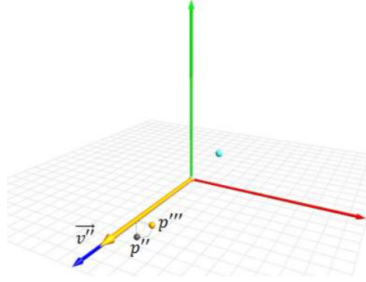


Figura 9 - Final do passo 2 da rotação [1].

Deste modo, o ponto p''' , presente na figura anterior pode ser representado pela matriz seguinte.

$$p''' = R_{z,\theta} p'' = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} p'' \quad (26)$$

Por fim, o último passo consiste em aplicar, pela ordem inversa, as matrizes inversas das transformações efetuadas, ou seja, R_x^{-1} e R_y^{-1} , obtendo-se algo semelhante à Figura 10.

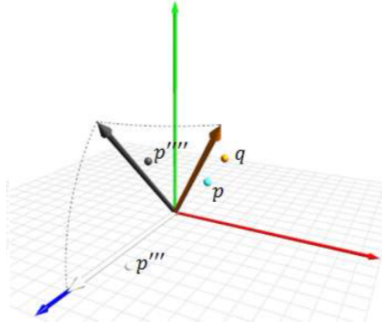


Figura 10 - Desfazer rotações nos eixos X e Y [1].

Posto isto, sabe-se que $p''' = R_{x-1} p''$ e $q = R_{y-1} p'''$ [1].

Representando todos os passos em sequência, a rotação de ângulo θ em redor do vetor \vec{v} pode ser escrita como na equação 27 [1].

$$R_{v,\theta} = R_y^{-1} \times R_x^{-1} \times R_{z,\theta} \times R_x \times R_y \quad (27)$$

Substituindo as variáveis pelas suas formas matriciais, tem-se [1]:

$$R_{v,\theta} = \begin{bmatrix} \frac{v_z}{d} & 0 & \frac{v_x}{d} & 0 \\ \frac{d}{d} & 1 & \frac{d}{d} & 0 \\ -\frac{v_z}{d} & 0 & \frac{v_z}{d} & 0 \\ 0 & 0 & \frac{d}{d} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & d & v_y & 0 \\ 0 & -v_y & d & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & d & -v_y & 0 \\ 0 & v_y & d & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{v_z}{d} & 0 & -\frac{v_x}{d} & 0 \\ \frac{d}{d} & 1 & \frac{d}{d} & 0 \\ \frac{v_z}{d} & 0 & \frac{v_z}{d} & 0 \\ \frac{d}{d} & 0 & \frac{d}{d} & 1 \end{bmatrix} \quad (28)$$

As matrizes representadas na equação 28, multiplicadas, resultam numa única matriz para executar uma rotação num espaço 3D em torno de um eixo arbitrário \vec{v} com um ângulo θ . A matriz resultante, $R_{v,\theta}$, está representada na equação 29, sendo que v_x, v_y e v_z foram substituídos por x, y e z , respetivamente, de modo a permitir uma leitura mais simples [1].

$$\begin{bmatrix} x^2 + (1 - x^2) \cos \theta & xy(1 - \cos \theta) - z \sin \theta & xz(1 - \cos \theta) + y \sin \theta & 0 \\ xy(1 - \cos \theta) + z \sin \theta & y^2(1 - y^2) \cos \theta & yz(1 - \cos \theta) - x \sin \theta & 0 \\ xy(1 - \cos \theta) - y \sin \theta & yz(1 - \cos \theta) + x \sin \theta & z^2 + (1 - z^2) \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (29)$$

Em suma, para obter um ponto q através da rotação de um ponto p em torno de um eixo \vec{v} por um ângulo θ tem-se [1]:

$$q = R_{v,\theta} \times p \quad (30)$$

2.0.3 Scale

A operação *scale* consiste no redimensionamento de um objeto que apresenta unidades diferentes das pretendidas. Esta transformação envolve a multiplicação das coordenadas de cada ponto que define o objeto por um certo valor [1].

$$p' = (s_x, s_y) \times p = (s_x \times p_x, s_y \times p_y), \quad (31)$$

onde p' representa o ponto redimensionado pelo valor p [1].

O redimensionamento deste ponto p através de uma matriz S apresenta-se de seguida [1].

$$p' = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix} = \begin{bmatrix} s_x \times p_x \\ s_y \times p_y \\ 1 \end{bmatrix} \quad (32)$$

Note-se que caso $s_z \neq 1$, o ponto resultante desta operação representa-se do seguinte modo [1]:

$$p' = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix} = \begin{bmatrix} s_x \times p_x \\ s_y \times p_y \\ s_z \end{bmatrix} \quad (33)$$

Desta forma, ter-se-á $p' = (s_x \times p_x, s_y \times p_y, s_z)$ [1].

A nível 3D a matriz S pode definir-se por [1]:

$$S = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (34)$$

Caso se pretenda realizar a operação inversa, a matriz correspondente será [1]:

$$S^{-1} = \begin{bmatrix} \frac{1}{s_x} & 0 & 0 & 0 \\ 0 & \frac{1}{s_y} & 0 & 0 \\ 0 & 0 & \frac{1}{s_z} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (35)$$

Na Figura 11 apresenta-se a operação *scale* a nível geométrico.

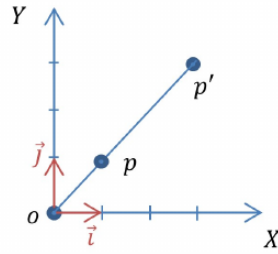


Figura 11 - *Scaling* do ponto p para o ponto p' [1].

3 Sistemas de Coordenadas

Um ponto p presente num sistema de coordenadas pode ser representado matematicamente por [1]:

$$p = x\vec{i} + y\vec{j} + z\vec{k} + o = \begin{bmatrix} \vec{i} & \vec{j} & \vec{k} & o \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (36)$$

É notório que as três primeiras colunas são vetores e representam os vetores do sistema de coordenadas e a última coluna é um ponto, representativo da origem [1].

A matriz presente na equação 36 é a matriz identidade, uma vez que ainda não foi efetuada nenhuma transformação [1].

Assim, de modo a deslocar um ponto p por um vetor \vec{v} a matriz de translação T pode ser descrita por [1]:

$$\begin{bmatrix} 1 & 0 & 0 & v_x \\ 0 & 1 & 0 & v_y \\ 0 & 0 & 1 & v_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (37)$$

Olhando para as colunas desta matriz, pode-se representar um novo sistema de coordenadas, interpretando as três primeiras colunas como os vetores e a última como a origem. Este novo sistema de coordenadas local pode ser desenhado relativamente ao sistema de coordenadas global, ou seja, $o' = o + \vec{v}$ [1].

A título de exemplo, considere-se o ponto p e a translação definida pelo vetor \vec{v} , resultante no ponto p' , como se vê na Figura 12.

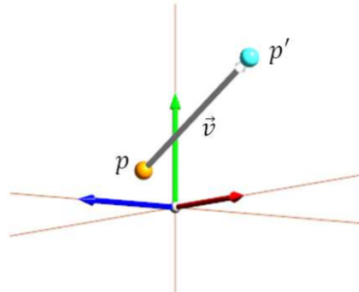


Figura 12 - Translação de ponto p [1].

Outra forma de retratar esta transformação geométrica é deslocar o próprio sistema de coordenadas, como é visível na Figura 13. O sistema de coordenadas global está representado na parte inferior esquerda da imagem e o resultante na parte superior direita. Se for considerado um ponto p_G no sistema global e for efetuada uma translocação por um vetor \vec{v} , é obtido o mesmo local onde está desenhado o ponto p_L , como se fosse desenhado com as suas coordenadas originais, relativamente ao sistema de coordenadas translocado [1].

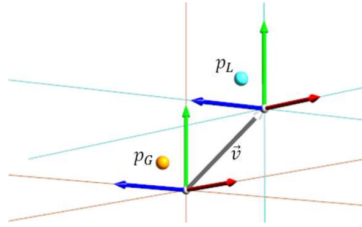


Figura 13 - Translação de sistema de coordenadas.

Deste modo, existem duas formas de interpretar as transformações geométricas: mover o ponto dentro do sistema de coordenadas, ou mover o próprio sistema de coordenadas [1].

No caso de uma rotação, a forma de analisar o processo torna-se mais intuitiva. Na Figura 14 é possível observar-se o sistema de coordenadas global (cinzento) e o local (azul).

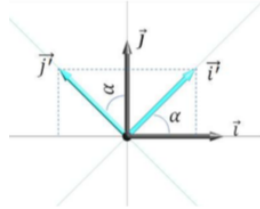


Figura 14 - Rotação em torno do eixo Z [1].

Note-se que o eixo Z é perpendicular ao papel. Os versores do sistema local $(\vec{i}', \vec{j}', \vec{k}')$ podem ser escritos em função dos versores do sistema global $(\vec{i}, \vec{j}, \vec{k})$. Como o versor associado ao eixo Z permanece inalterado, tem-se [1]:

$$\vec{k}' = 0\vec{i} + 0\vec{j} + \vec{k} \quad (38)$$

$$\vec{i}' = \cos \alpha \vec{i} + \sin \alpha \vec{j} + 0 \vec{k} \quad (39)$$

$$\vec{j}' = -\sin \alpha \vec{i} + \cos \alpha \vec{j} + 0 \vec{k} \quad (40)$$

Assumindo que a matriz para as transformações terá, nas suas colunas, os seus versores e a origem, que permanece inalterada pela rotação, a mesma pode ser definida por [1]:

$$R_{z,\alpha} = \begin{bmatrix} \vec{i}' & \vec{j}' & \vec{k}' & o \end{bmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 & 0 \\ \sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (41)$$

O resultado obtido na equação 41 é exatamente o mesmo que o obtido na equação 16, como seria expectável [1].

De igual modo é possível desenhar o sistema de coordenadas a partir de uma matriz, seguindo o raciocínio até agora desenvolvido, ou seja, assumindo que as três primeiras colunas representam os versores do sistema local e a última coluna retrata a origem [1].

4 Composição de Transformações Geométricas

A composição de transformações geométricas resulta numa matriz 4x4 com o mesmo formato que cada matriz possui individualmente. Tome-se como exemplo um vetor $\vec{v} = \vec{a} + \vec{b}$. As translações descritas pela aplicação dos vetores \vec{a} e \vec{b} podem ser descritas pelas matrizes T_a e T_b , respetivamente [1].

$$T_a = \begin{bmatrix} 1 & 0 & 0 & a_x \\ 0 & 1 & 0 & a_y \\ 0 & 0 & 1 & a_z \\ 0 & 0 & 0 & 1 \end{bmatrix}; T_b = \begin{bmatrix} 1 & 0 & 0 & b_x \\ 0 & 1 & 0 & b_y \\ 0 & 0 & 1 & b_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (42)$$

Tendo em conta estas matrizes, imagine-se que se pretende aplicar primeiramente a translação de um ponto p pelo vetor \vec{a} e depois por \vec{b} . O resultado final será dado pela seguinte equação [1]:

$$p'' = T_b p' = T_b T_a p \quad (43)$$

A composição destas duas translações pode ser observada na Figura 15, onde se verifica que o sistema de coordenadas é estático.

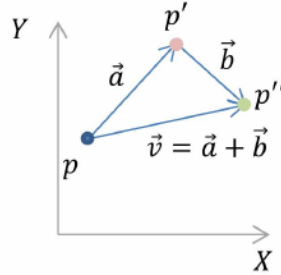


Figura 15 - Composição de translações [1].

No entanto, a expressão da equação 43 pode ser interpretada de outro modo. Neste caso, o foco está nos sistemas de coordenadas e representa-se o ponto p com as suas coordenadas originais noutro sistema. Na Figura 16 verifica-se este exemplo.

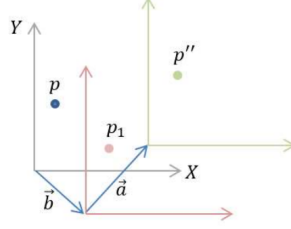


Figura 16 - Composição de translações de sistemas de coordenadas [1].

Note-se que a aplicação de rotações em sistemas de coordenadas que não foram previamente redimensionadas de modo uniforme pode não ter o mesmo resultado que uma rotação do sistema de coordenadas. Considere-se uma matriz S e uma rotação de 45 graus sobre o eixo do Z através da matriz R [1].

$$S = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; R = \begin{bmatrix} \cos 45 & -\sin 45 & 0 & 0 \\ \sin 45 & \cos 45 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (44)$$

Aplicando a rotação de R à matriz S , obtém-se [1]:

$$M = \begin{bmatrix} 3 \cos 45 & -3 \sin 45 & 0 & 0 \\ \sin 45 & \cos 45 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (45)$$

O sistema de coordenadas definido por M encontra-se representado na Figura 17.

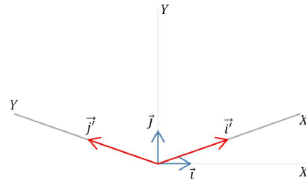


Figura 17 - Rotação após redimensionamento não uniforme, onde os versores originais se encontram a azul e os transformados a vermelho [1].

Caso se pretenda mover um objeto por um vetor e depois rodá-lo, a matriz resultante será dada por M [1].

$$M = TR = \begin{bmatrix} l & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} r & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} r & t \\ 0 & 1 \end{bmatrix}, \quad (46)$$

onde T representa a matriz de translação e R define a matriz de rotação [1].

5 Sistema Solar Estático e Classes

Através da utilização de um ficheiro *XML* é possível tirar partido do facto de existirem grupos intrínsecos a outros. Desta forma, as transformações efetuadas no elemento mais exterior serão passadas, também, aos mais internos. Ou seja, sendo o elemento exterior um planeta e os internos as suas luas, estas estarão dependentes das transformações efetuadas anteriormente para desenhar o planeta, possuindo, ainda, as suas próprias transformações, que são aliadas às anteriores.

A título de exemplo tenha-se uma translação pelo vetor \vec{v} para o planeta X. A sua lua sofre, em seguida, uma outra translação pelo vetor \vec{t} . Quer isto dizer que a translação total que afetará a lua será:

$$planeta = \begin{bmatrix} 1 & 0 & 0 & v_x \\ 0 & 1 & 0 & v_y \\ 0 & 0 & 1 & v_z \\ 0 & 0 & 0 & 1 \end{bmatrix} origem \quad (47)$$

$$lua = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} planeta \quad (48)$$

Dito isto, a transformação total será:

$$lua = \begin{bmatrix} 1 & 0 & 0 & v_x + t_x \\ 0 & 1 & 0 & v_y + t_y \\ 0 & 0 & 1 & v_z + t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} origem \quad (49)$$

No caso de ser necessária uma rotação além desta translação, o formato da matriz será semelhante ao da equação 46, como pode ser comprovado no capítulo anterior.

No que toca à leitura do ficheiro onde estão definidas todas as transformações e corpos a desenhar, tem que existir uma forma de guardar a ordenação das mudanças geométricas, pois, como já foi provado anteriormente, esta afeta o resultado final.

A forma encontrada para o fazer foi guardar, juntamente com os vetores de pontos, que representam cada figura, valores que representem cada uma das transformações geométricas possíveis. Sabendo que estas são *translate*, *rotate* e *scale*, e que podem ocorrer numa qualquer ordem, têm que possuir um indicador da sua posição de execução, como se vê na Figura 18. Assim,

cada figura possui os valores da possível translação, rotação e redimensionamento, tomando o valor de ordenação de zero caso não exista.

```
class figura {
private:
    std::vector<std::vector<Triangulo>> pontos;
    std::vector<figura> filhos;

    float trX;
    float trY;
    float trZ;
    int T_order;

    float rA;
    float rX;
    float rY;
    float rZ;
    int R_order;

    float sX;
    float sY;
    float sZ;
    int S_order;
};
```

Figura 18 - Classe *Figura*.

Por exemplo, se a figura N possuir um *scale* por $X = 3$ e, em seguida, um *translate* pelo vetor $\vec{v} = (1, 0, 0)$, não tendo sido efetuada qualquer alteração ao estado inicial da matriz, os seus valores dos atributos na classe serão:

- $trX = 1;$ • $rA = 0;$ • $sX = 3;$
- $trY = 0;$ • $rX = 0;$ • $sY = 1;$
- $trZ = 0;$ • $rY = 0;$ • $sZ = 1;$
- $T_order = 2;$ • $R_order = 0;$ • $S_order = 1.$

Onde trX , trY e trZ correspondem às translações nos eixos do X , Y e Z , respetivamente; rA consiste no ângulo de rotação; rX , rY , rZ , sX , sY e sZ são as rotações e *scalings* nos respetivos eixos, e, por fim, T_order , R_order e S_order correspondem à ordem pela qual se efetuam as transformações.

Além disso, como mencionado anteriormente, as figuras relativas aos "filhos" estarão dependentes da matriz final do elemento "pai". Note-se que, como não pode existir repetição de uma mesma transformação geométrica por figura, ou seja, uma figura apenas pode ter um *translate* seu e o do pai, permitindo que apenas seja necessário guardar os valores uma vez como demonstrado na Figura 18.

Deste modo, torna-se mais simples efetuar o posicionamento de cada planeta, relativamente ao Sol, e, conseqüentemente, dos seus satélites naturais,

dando, ainda, uso às funções de *push* e *pop matrix* presentes na *framework OpenGL*, em detrimento da anulação das transformações geométricas efetuadas, para voltar ao ponto pretendido.

Tendo em conta que o Sol se encontra na posição central deste Sistema Solar, de modo a desenhar um planeta, é necessário efetuar uma translação para as coordenadas do mesmo. Uma vez que existe mais do que um planeta, a matriz com o sistema de coordenadas relativa ao Sol deve ser guardada, de modo a que esteja acessível após a inclusão do planeta e respetivos satélites naturais, para permitir a rápida continuação do processo.

No que toca às luas ou anéis, no caso de Saturno e Urano, é utilizada a matriz relativa ao planeta à qual pertencem, sendo que qualquer transformação geométrica é feita sobre a mesma, nunca deixando de a guardar, pois pode existir mais que um satélite natural para representar.

Já a Cintura de Asteróides, existente entre Marte e Júpiter foi representada na forma de um anel, de modo a destacar a sua existência, seguindo a mesma lógica de trabalho dos astros já mencionados. Posteriormente, será implementada uma primitiva gráfica que permita a sua representação mais realista.

Dito isto, e, tendo em consideração os conceitos base de Computação Gráfica, apresentados nos capítulos anteriores, o método escolhido permite uma representação estática do Sistema Solar, como se vê na Figura 19. Note-se que os corpos celestes e as suas distâncias não foram implementadas à escala, permitindo, assim, um retrato onde todos os astros são visíveis em simultâneo, não deixando de ser visível qual o corpo em questão, devido ao seu posicionamento, dimensão, luas e/ou anéis.

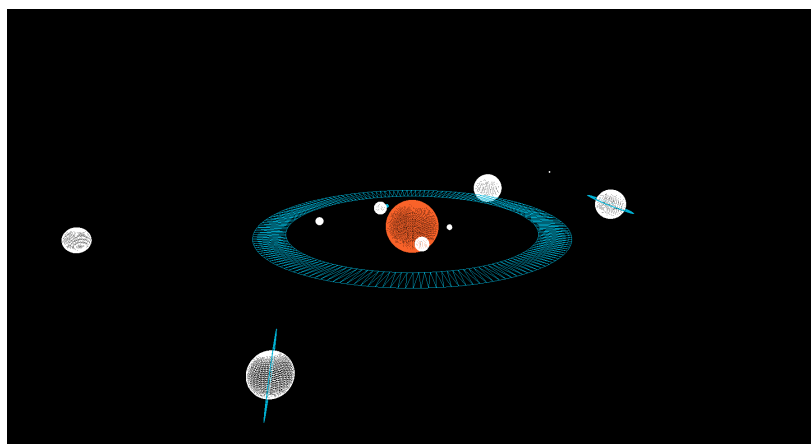


Figura 19 - Sistema Solar estático.

6 Conclusões e Trabalho Futuro

O desenvolvimento deste projeto permitiu um aumento no conhecimento relativamente a transformações geométricas, nomeadamente a ordem pela qual são efetuadas e a forma como podem e devem ser implementadas num panorama prático.

Tendo em conta que todos os objetivos propostos foram implementados com sucesso, nomeadamente o desenvolvimento de um modelo de um Sistema Solar estático e a implementação de hierarquia na representação de figuras, o grupo considera esta segunda fase do projeto um sucesso.

Em suma, foi possível solidificar os conhecimentos adquiridos durante as aulas da Unidade Curricular de *Computação Gráfica*, bem como aliar a teoria à prática, permitindo a melhor execução de propostas que envolvam hierarquia na representação de figuras, juntamente com transformações geométricas.

Referências

- [1] António Ramires. *The Geometric Pipeline: Notes for an Undergraduate Course in Computer Graphics*. University of Minho, 2019.