

Manipuladores de transformações

O sucesso de um programa de modelagem 3D depende da intuitividade em suas operações. A manipulação da câmera e a capacidade de aplicar transformações geométricas em objetos da cena através do *mouse* são exemplos de operações comuns em tais softwares. Neste EP, você exercitará transformações e a utilização de quatérnios para rotação, associados a uma boa dose de interação humano-computador.

1 Manipuladores

Os manipuladores permitem uma representação visual dos comandos de transformação e permitem a translação, a rotação e a escala de objetos. Eles estão presentes em praticamente qualquer software de modelagem 3D. Neste EP, deverão ser implementados os manipuladores básicos, operados através do *mouse*. Antes, porém, você precisará carregar objetos para dentro da sua cena.

2 Objetivos

2.1 Carregamento de arquivos OBJ

Você deverá possibilitar o carregamento de arquivos `.obj` em uma mesma cena. O usuário deverá utilizar a tecla `<'Shift' + botão esquerdo do mouse>` para selecionar um objeto. Inicialmente, todos os objetos devem aparecer no centro do mundo.

Um arquivo `.obj` possui diversas funcionalidades que descrevem um modelo 3D. Para esse exercício, o seu programa deve considerar todos os vértices, coordenadas de textura e normais descritas em um arquivo `.obj`. Quando um vértice não possuir uma normal associada, você deve calculá-la considerando a face que ele está localizado. Quando um vértice não tem coordenada de textura, você pode mantê-la como `"(0,0)"`. Um modelo geométrico no arquivo `.obj` pode conter diversos grupos de faces que podem ser manipulados individualmente. Nesse exercício, vamos desconsiderar a divisão de grupo, considerando que todas as faces pertencem a apenas um objeto.

Um arquivo `.obj` também permite definir um material para um conjunto de faces e isso deve ser suportado por seu programa. Para isso considere que a linha iniciada por `"mtllib"` fornece um arquivo `".mtl"` que descreve materiais do modelo Blinn-Phong que deve ser carregado por seu programa. Quando o seu código de carregamento de arquivo `.obj` encontrar uma linha iniciada com `"usemtl"` seguida por um nome de material você deve associar esse material descrito no arquivo `".mtl"` com todas as faces seguintes até encontrar a próxima linha iniciada por `"usemtl"`. Defina um material padrão utilizado até encontrar a primeira linha iniciada por `"usemtl"`.

Outro detalhe de arquivos `.obj` que seu programa deve tratar são faces com três ou quatro vértices. Embora este formato permita faces planares com um número arbitrário de vértices, para esse exercício, você deve considerar que os modelos que serão utilizados possuem faces com três ou quatro vértices. Em anexo desse exercício você encontrará um script em Python (`"triangulate-obj-faces.py"`) que utilizando arquivos `.obj`, transforma as faces quadriláteras em faces triangulares. Você pode utilizá-lo como base para implementar essa funcionalidade no seu programa. Ainda em anexo, são fornecidos modelos `.obj` que seu programa deve ser capaz carregar (`"obj-models.zip"`). Os nomes dos criadores dos modelos estão descritos nos arquivos `README.txt` dentro das pastas.

2.2 Seleção

Ao clicar em um objeto ele deve aparecer ressaltado e entrar em modo de edição. Você deverá emitir um raio virtual que atravessará a cena a partir da câmera passando pelo objeto mais próximo da câmera. Para ressaltar o objeto selecionado, você poderá alterar a cor do objeto, poderá redesenhar o objeto em *wireframe*, ou desenhar o seu *bounding box*, mas a mudança deve ficar evidente. Dica: veja os arquivos `Ray.py` e `Camera.py` do código da disciplina.

2.3 Operações e transformações

Uma vez selecionado o objeto, o usuário poderá *i)* removê-lo da cena, ou *ii)* aplicar transformações geométricas. Você deverá utilizar a tecla 'x' ou 'X' ou 'Delete' para remover um objeto selecionado da cena.

Deverão ser implementados 3 opções de transformação: *i)* translação, *ii)* escala e *iii)* rotação. Ao escolher qualquer uma dessas opções, o objeto selecionado deverá mostrar, ao redor dele, os respectivos manipuladores da transformação escolhida. A figura abaixo demonstra uma forma de representar os manipuladores visualmente, conforme encontrado no *software Blender 3D*. Neste caso, os manipuladores identificam os eixos x , y e z .

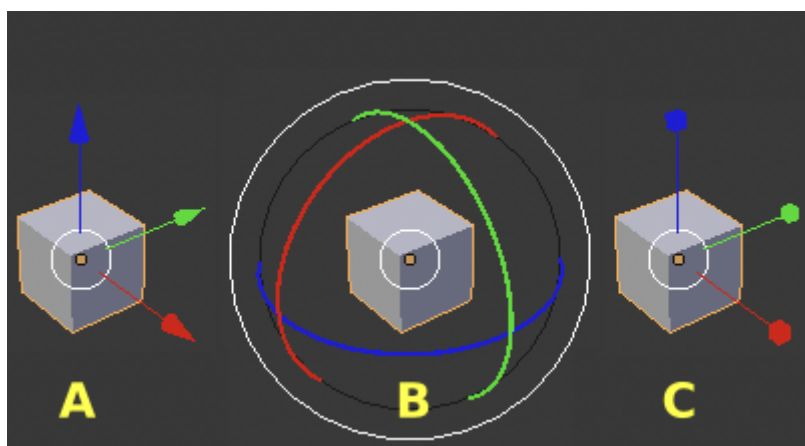


Figura 1: *Widgets* manipuladores para A) translação, B) rotação e C) escala.

Você não precisará interagir diretamente com a representação geométrica dos manipuladores, diferentemente do que é feito em programas mais complexos. Após a seleção, o usuário utilizará as teclas 't' ou 'T', para selecionar a translação, 'r' ou 'R' para selecionar a rotação, e 's' ou 'S' para escala. Após a seleção da transformação, o usuário deverá utilizar as teclas 'x' ou 'X'; 'y' ou 'Y'; e 'z' ou 'Z' para selecionar o eixo de transformação. O seu manipulador deverá refletir essa escolha!

Para efetuar qualquer das operações o usuário pressionará o botão da esquerda, e enquanto pressionado, rastreará os movimentos do *mouse* e os traduzirá em transformações incrementais de acordo com a operação e eixo escolhidos. A câmera não deverá ser modificada.

Assim, para a translação, os movimentos do *mouse* deverão transladar o objeto selecionado no eixo escolhido. Para a escala, você deverá mapear os movimentos do *mouse* para aumentar ou diminuir a escala do objeto selecionado no eixo escolhido. Já na rotação, você deverá definir um *trackball* virtual para mapear movimentos do *mouse* em rotações ao redor do centro do objeto, e não precisará da seleção dos eixos.

Quando o usuário soltar o botão esquerdo do *mouse*, a transformação será aplicada definitivamente. Para desativar a seleção, e conseqüentemente não ressaltar mais o objeto, o usuário deverá pressionar a tecla 'Esc' ou clicar em um ponto na cena onde não existam objetos. O objeto então será desselecionado. Neste instante, todas os movimentos do *mouse* voltam a ser convertidos para o controle da câmera, como de costume.

3 Avaliação

Tabela 1: Ficha de Avaliação

Item	Descrição	Pontos
1	Leitura correta de arquivos .obj	20%
2	Correta seleção de objetos com <i>feedback</i> visual	15%
3	Correta ativação e renderização dos manipuladores	15%
4	Correta implementação de translações	10%
5	Correta implementação de escalas	10%
6	Correta implementação de rotações	15%
7	Correta remoção de objetos da cena após seleção	15%

Observações gerais

Este será um EP feito em até duas pessoas. Forneça um arquivo `LEIAME.TXT` com quaisquer informações que sejam relevantes para a descrição do seu trabalho. Entregue um único arquivo compactado, contendo o código-fonte, e nomeado com os nomes dos participantes. Ex: Marcel-Dennis.zip.