

Computação Gráfica

João Domingues

November 21, 2023

1 Introdução

A presente atividade teve como objetivo a aplicação prática dos conceitos aprendidos na disciplina de Computação Gráfica, com foco na renderização gráfica de objetos tridimensionais. O trabalho consistiu na importação e renderização de uma maçã a partir de um arquivo `.obj` num espaço tridimensional, além da implementação de controlos para a navegação nesse espaço, iluminação e uma simples textura.

2 Implementação

O código seguinte mostra como foi realizado o "load" de um ficheiro `.obj` contendo informações dos vértices, coordenadas de textura e normais.

Listing 1: "Load" do ficheiro `.obj`

```
std::vector<glm::vec3> vertices;
std::vector<glm::vec2> uvs;
std::vector<glm::vec3> normals;
bool res = loadOBJ("applew.obj", vertices, uvs, normals);
```

Neste pedaço de código, `vertices` armazena as coordenadas dos vértices da maçã; `uvs` armazena as coordenadas U e V das texturas e `normals` armazena as normais de cada vértice.

A linha a seguir demonstra como os triângulos são renderizados na aplicação.

Listing 2: Renderização dos triângulos

```
glDrawElements(GL_TRIANGLES, indices.size(), GL_UNSIGNED_SHORT, (void *)0);
```

Nesta linha, `glDrawElement` é uma função OpenGL que desenha a renderização na tela. `GL_TRIANGLES` especifica o modo de renderização; `indices.size()` indica o número total de elementos (índices) a serem desenhados; `GL_UNSIGNED_SHORT` especifica o tipo dos elementos nos dados do índice e `(void *)0` indica que a renderização deve começar a partir do início do buffer de índices.

2.1 Controlos, iluminação e texturas

Introduziu-se também uma "lâmpada" e um ficheiro de texturas `.DDS`, de modo que, ao serem combinados, seja possível observar com maior nitidez a luz presente no cenário tridimensional, proporcionando uma perspetiva mais eficiente na visualização da maçã.

Controlos de navegação foram implementados para permitir que o utilizador interaja com o ambiente tridimensional. Isso incluiu a capacidade de mover a câmara para diferentes posições e orientações, proporcionando uma experiência imersiva.

2.2 Conversão do ficheiro .obj

Foi usado um "script" onde era introduzido o ficheiro original .obj. Este ficheiro apresentava linhas com 4 posições de vértices, no entanto, a função "LoadOBJ" apenas consegue ler 3 destas posições. Assim esta conversão é necessária para que a renderização do .obj seja correta.

Este "script" converte as 4 posições para 2 posições adicionando mais uma linha no ficheiro .obj.

3 Conclusão

O projeto foi concluído com sucesso, alcançando os objetivos propostos. A implementação permitiu carregar um modelo obj de uma maçã, exibindo-a no espaço tridimensional. Os controlos implementados proporcionaram uma navegação eficiente no ambiente tridimensional e a iluminação apresentada proporciona uma melhor visualização do objeto.