APLICAÇÃO DE *OPENGL SHADING LANGUAGE* NO AMBIENTE TRIDIMENSIONAL: FEI VIRTUAL

Glauber Oliveira da Rocha[1], Plinio Thomaz Aquino Júnior
Centro Universitário da FEI – Fundação Educacional Inaciana Pe. Sabóia de Medeiros
glauberoliveirarocha@gmail.com, plinio.aquino@fei.edu.br

Resumo: Este trabalho apresenta o processo de criação do ambiente virtual tridimensional que representa a infraestrutura real do Centro Universitário da FEI, aplicando *OpenGL Shading Language* (GLSL) como técnica para a geração de um ambiente virtual mais realístico.

1. Introdução

Os ambientes virtuais tridimensionais e interativos disponibilizam um novo modo de interação com usuário propiciando navegação, interação e imersão em tempo real. Este trabalho tem como objetivo integrar recursos computacionais em ambientes tridimensionais que aumentam o grau de realismo dos modelos 3D.

2. Material e Métodos

A reprodução tridimensional dos prédios foi realizada através do Google Sketchup [1] que além da modelagem, possibilitou a padronização da escala dos modelos a partir da utilização de imagens de satélite disponíveis pela própria ferramenta.

O OpenGL [2] foi utilizado como mecanismo para renderizar os modelos exportados do Google Sketchup, para que isso seja possível, foi desenvolvido um módulo para carregamento de modelos, que utiliza a técnica de *Vertex Buffer Objects* (VBO), possibilitando a alocação de dados referentes aos vértices do modelo na memória de vídeo, provendo o acesso rápido da GPU.

O estudo e aplicação da linguagem GLSL [3] possibilitou o aumento do realismo do ambiente virtual. Tal linguagem permite a geração de código a ser executado em cada vértice e em cada *pixel* a ser renderizado. O GLSL faz parte do padrão OpenGL e permite a alteração de suas funcionalidades padrões para iluminação, texturização, projeção e coloração permitindo o desenvolvimento de efeitos como *Iluminação por Pixel, Bump Mapping* e *Dynamic Ambient Occlusion*.

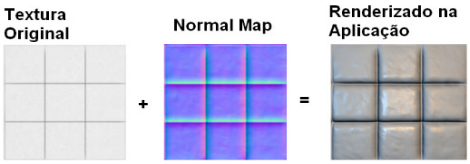

Figura 1 – Exemplo de Aplicação de *Bump Mapping*.

O *Bump Mapping* é utilizado para gerar realismo ao renderizar texturas através do cálculo de iluminação na textura. Para a realização desse cálculo é necessário a geração de *Normal Maps* (textura com o mapeamento das normais da textura original) para que seja utilizada no cálculo de luz difusa que é aplicado na textura (Figura 1).

3. Resultados

A utilização da linguagem GLSL possibilitou a aplicação de técnicas de realismo não disponíveis pelo OpenGL, possibilitando a geração de um ambiente virtual tridimensional mais imersivo para o usuário (Figura 2). Os prédios e objetos foram gerados através da utilização do Google Sketchup e importados para a aplicação em uma estrutura de dados que possibilita a utilização de VBOs.

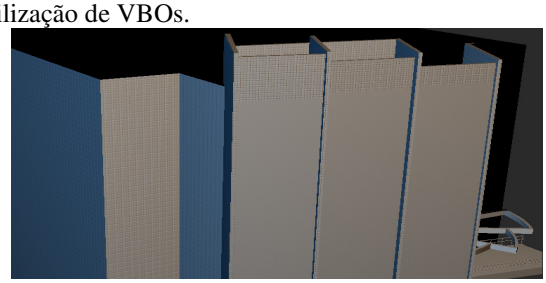

Figura 2 – Aplicação do *Bump Mapping* no Prédio K

A aplicação carrega as texturas do modelo e verifica se existe textura com a *Normal Map* relacionada. No momento da renderização dos vértices é executado o código em GLSL para cada vértice e subsequentemente para cada *pixel* [3] realizando o cálculo de iluminação (Figura 3) da textura através do *Normal Map* e cálculo de iluminação do objeto utilizando a posição da luz como referência.

```
//intensidade de luz desse ponto da textura
float texDifusa = max(dot(normalTexture, luzDirecao), 0.0)
//intensidade de luz do fragmento em relação a posicao da luz
pontoLuzDifusa = max(dot(luzDirecao,normalVertex),0.0);
//Aplicação do NormalMap em cima da textura
vec3 corPonto = texDifusa*texture2D(textura, TexCoord.st).rgb;
//Aplicação de iluminação difusa e ambiente
output = vec4(corPonto*pontoLuzDifusa,1.0) + ambiente;
```

Figura 3 – Cálculo de luz do ambiente com GLSL

4. Conclusões

O projeto produziu como artefato uma biblioteca de objetos que caracterizam o ambiente, possibilitando incrementos com novas características do mundo real e expansão para outros projetos.

A aplicação da linguagem GLSL apoiou o processo de criação de um ambiente virtual mais realístico através de técnicas como *Bump Mapping*, sendo necessário apenas maior conhecimento matemático, além do aprofundamento na linguagem de programação, para o desenvolvimento desses efeitos.

5. Referências

[1] Google SketchUp <sketchup.google.com>, 2011.
[2] Shreiner, D. OpenGL Programming Guide, 7ª Edição, 2009.
[3] Randi J. Rost. OpenGL Shading Language, 2ª Edição, 2006.

Agradecimentos

Ao Centro Universitário da FEI pela disponibilização do Laboratório de Engenharia de Usabilidade.

[1] Aluno de IC do Centro Universitário da FEI.