Desenvolvimento de um Motor Físico 3D de Apoio a Ambientes Lúdico Educacionais para Uso em Hardwares Limitados

Joimar Brito Gonçalves Filho¹; Victor Travassos Sarinho²;

1. Bolsista PEVIC, Graduando em Engenharia de Computação, Universidade Estadual de Feira de Santana, e-mail: joimarbgf@gmail.com
2. Orientador, Departamento de Exatas, Universidade Estadual de Feira de Santana, e-mail: vsarinho@gmail.com

PALAVRAS-CHAVE: Jogos Digitais, Motor de Jogo, Motor Físico, Jogos Educacionais.

INTRODUÇÃO

A interatividade permitida por meio de jogos digitais, e o ambiente lúdico proporcionado pelos mesmos, colabora para a assimilação de conteúdos didáticos em geral. Jogos eletrônicos de sucesso proporcionam uma experiência imersiva de níveis variados, permitindo ao jogador sentir-se parte do universo fictício proposto pelo mesmo.

Interatividade e imersão são características importantes que podem ser bem exploradas em jogos educativos, tanto à nível de abstração quanto no ensino e na aprendizagem de assuntos diversos. Como resultado, tem-se atualmente no uso de jogos eletrônicos uma ferramenta viável e inovadora para a educação escolar em si (WARD, 2008).

Considerando a produção de jogos digitais, um motor de jogos é uma ferramenta que permite ao desenvolvedor agregar funções básicas para o funcionamento de um jogo, o que varia desde o gerenciamento de periféricos de entrada e saída até o som e a renderização de modelos gráficos (CLUA, 2005).

Visando a criação de uma ferramenta lúdica para uso didático em sala de aula, esse projeto almejou desenvolver um motor físico para integrar um motor de jogo e adaptá-lo a uma placa Raspberry PI 3, sendo um hardware limitado e acessível às escolas públicas.

MATERIAL E MÉTODOS OU METODOLOGIA (ou equivalente)

Uma vez começado os estudos voltados ao uso dos cálculos físicos em jogos eletrônicos, foi possível efetuar alguns testes em ambientes simples como Blender, que utiliza a linguagem de programação Python (Brito, 2006). Apesar da linguagem utilizada para o processo de desenvolvimento do motor gráfico ser C++, o uso de ferramenta Blender serviu para uma familiarização com conceitos básicos de simulação física em jogos eletrônicos. Os estudos para a finalização/acoplamento do motor gráfico se mostraram um desafio devido ao pouco material a respeito da utilização do OpenGL para o desenvolvimento de motor de jogos (Cohen, 2006). O OpenGL trata-se de uma API (Application Programming Interface) usada para a criação de aplicativos de computação gráfica (Cohen, 2006). A pouca quantidade de dados e materiais bibliográficos foi parcialmente sanada por tutoriais da comunidade de usuários da API em questão, que auxiliaram na configuração do ambiente de desenvolvimento. Tendo terminado a etapa de configuração, a etapa seguinte foi compreender as subrotinas do OpenGL a serem usadas no motor gráfico e, em seguida, aplicar as dinâmicas estudadas

com a ferramenta Blender, mas agora na linguagem C++. Esta etapa pode ser resumida nos seguintes tópicos:

- 1) Desenvolver funções que executem equações básicas de cinemática;
- 2) Criar um sistema de manipulação da câmera virtual;
- 3) Finalizar rotinas do motor gráfico;
- 4) Implementar o sistema de reconhecimento de periféricos;
- 5) Desenvolver o sistema de detecção de colisão;
- 6) Implementar a simulação de fluídos e corpos macios;

O estudo e desenvolvimento dos temas destes tópicos permitiu uma compreensão maior a respeito da interação do motor gráfico com o motor físico e como se dá a comunicação entre um jogo eletrônico e os diferentes periféricos do computador. Basicamente, as simulações de cinemática mais simples como quedas, se dá apenas ajustando a variável de intensidade da força puxando todos os objetos no sentido para baixo. O comportamento de objetos de massas diferentes se dá apenas usando um valor diferente para a força vertical.

O gerenciamento de periféricos pelo OpenGL permanece basicamente imutável das versões anteriores, o que facilitou a busca por material a esse respeito, uma vez que a documentação voltada para a comunicação com os periféricos é válida não importando qual a versão do OpenGL se está usando para desenvolvimento.

O passo seguinte ao desenvolvimento do motor físico é a adaptação do motor de jogo integrado ao motor gráfico, para que a game engine funcione numa placa Raspberry 3. O modelo do Raspberry disponível possui um processador GPU (Graphics Processing Unit), permitindo o uso do OpenGL, exigindo apenas uma versão mais modesta dos motores gráfico e físico.

RESULTADOS E/OU DISCUSSÃO (ou Análise e discussão dos resultados)

A pesquisa gerou um código fonte que desempenha as funcionalidades do motor físico e gráfico, sendo possível fazer uma diversidade de desenhos em duas ou três dimensões com variações de cor. O objeto 2D ou 3D pode ser manipulado por meio de teclado, mouse ou gamepad de Xbox 360 e a resposta aos comandos dados pelos periféricos em nenhum momento exhibe algum defeito.

A usabilidade voltada para a modelagem de objetos ainda é muito limitada, exigindo que o usuário use comandos em linha de código para informar o desenho a ser feito, mas tal limitação é contornada pela possibilidade de importação de modelos já prontos de programas específicos como Blender.

O funcionamento do motor de jogo na placa Raspberry pi 3 não pode ser concluído por limitações laboratoriais e de tempo.

Figura 1: Cubo 3D colorido gerado pelo motor de jogo, podendo ter sua rotação controlada por teclado e mouse ou gamepad.

CONSIDERAÇÕES FINAIS (ou Conclusão)

O processo de compreensão de conceitos da API OpenGL voltada para a produção de jogos e motor de jogos foi bem sucedida, tendo em conta a pouca quantidade de material publicado a esse respeito.

O estudo feito a respeito do uso do Raspberry Pi propiciou algum planejamento para a otimização do código para que funcione na placa, porém, as limitações laboratoriais e o tempo despendido na busca por materiais sobre a aplicação do OpenGL em jogos acabaram por deixar essa etapa de implementação de uma versão do motor para Raspberry incompleta.

REFERÊNCIAS

WARD, J. 2008. *What is a Game Engine?*-

http://www.gamecareerguide.com/features/529/what_is_a_game_.php - acessado em 29/03/2016

CLUA, E. W. G. e BITTENCOURT, J. R., 2005. *Desenvolvimento de Jogos 3D: Concepção, Design e Programação*. Anais da XXIV Jornada de Atualização em Informática do Congresso da Sociedade Brasileira de Computação, p. 1313-1356, São Leopoldo, Brazil.

Brito, A., (2006) Blender 3D: Guia do Usuário, Ed. Novatec, São Paulo.