

PROJETO LUCKY RABBIT: IMPLEMENTAÇÃO DE UM SOFTWARE PARA ANIMAÇÃO 2D EM AMBIENTE 3D

Mateus Moraes BUENO¹

Paulo César Rodacki GOMES²

INTRODUÇÃO

A animação de figuras ilustradas remonta ao período paleolítico evidenciado por pinturas em cavernas que exibem animais pré-históricos desenhados com várias pernas sobrepostas umas às outras, em uma provável tentativa de suscitar a ilusão de movimento, para os que observassem a gravura (THOMAS, 1958). Desde então, a animação tem tomado técnicas e formas com o intuito de entreter, divulgar ações e informar públicos variados.

As técnicas de animação tradicionais em duas dimensões, popularizadas a partir do começo do século XX no cinema, ainda são empregadas na indústria de entretenimento. Apesar de grande parte das produções contarem com a automatização de processos intermediada por computadores, o método ainda é similar ao dos primórdios da animação no cinema. Na técnica mais comum, conhecida como frame-by-frame (quadro a quadro), cada desenho representa uma etapa do movimento desejado, separado em várias folhas (quadros ou frames). Depois, esses desenhos são sobrepostos uns aos outros em uma velocidade determinada e a ilusão de movimento é atingida. Nas variações dessa técnica, os frames principais do movimento, isto é, o começo, o meio e o fim do movimento, são desenhados pelo artista e o computador calcula os frames intermediários (JOHNSTON; JOHNSTON; THOMAS, 1995).

No caso das animações tridimensionais, a computação gráfica auxilia na modelagem de personagens e elementos em três dimensões, de forma semelhante à manipulação de massas de modelar. Definidos os elementos e personagens, câmeras de projeção virtuais capturam as imagens dos modelos em diferentes ângulos. A ilusão de movimento é conseguida por um variado número de técnicas, que compreendem as mais tradicionais, como a frame-by-frame e técnicas mais avançadas como captura de movimentos e animação por esqueletos (BEANE, 2012).

A animação 3D depende muito do auxílio do computador. O artista mais tradicional acaba sendo substituído pelo modelador e as ilustrações que servem de base para a concepção do desenho acabam sendo todas substituídas pelos modelos em três dimensões. Por outro lado, a animação bidimensional ainda não contém técnicas otimizadas para a geração de imagens e ângulos de filmagem mais complexos,

¹Aluno do Tecnólogo em Análise e Desenvolvimento de Sistemas do Instituto Federal Catarinense – *campus* Blumenau.

² Orientador do Trabalho de Conclusão de Curso.

tornando o trabalho de animação mais moroso e difícil, quando se pretende renderizar cenas mais aperfeiçoadas (WHITED, BRIAN., et. al., 2012).

O presente trabalho visa desenvolver um software que automatize o processo de animação 2D em um ambiente 3D. A proposta é construir um ambiente de criação e edição de figuras geométricas bidimensionais em um ambiente com movimentação em três dimensões. A animação é definida pelo desenho de quadros-chave e gerada a partir do cálculo dos quadros intermediários. Como a ambientação do desenho situa-se em um mundo virtual 3D, vários ângulos de câmera podem ser utilizados na renderização da cena.

MATERIAIS E MÉTODOS

O software que este trabalho propõe, batizado de Projeto Lucky Rabbit, é desenvolvido em linguagem C++ com o auxílio da biblioteca gráfica OpenGL.

A linguagem de programação C++ foi criada pelo cientista Bjarne Stroustrup em 1979 a partir da linguagem C. A princípio denominada C with Classes (C com Classes) foi renomeada em 1983 e desde então é atualizada. Se trata de uma linguagem multiparadigma (procedural, funcional e orientada a objetos) com tipagem e estática e nominativa. É considerada híbrida, situando-se entre o nível de máquina (baixo nível) e o alto nível e, por causa de sua grande performance em processamentos que exigem desempenho rápido, foi a escolhida para este projeto (STROUSTRUP, 2013).

A biblioteca de código OpenGL foi desenvolvida inicialmente pela Silicon Graphics Inc. como uma API (Application Programming Interface) em 1991. Atualmente é gerenciado pelo consórcio Khronos Group e é distribuída. Esta biblioteca contém um conjunto de códigos que auxiliam na construção de elementos gráficos e cálculos de primitivas para gráficos 2D e 3D.

Apesar de ser escrita em linguagem de programação C, pode ser estendida por várias outras linguagens de programação (SHREINER; SELLERS; KESSENICH, 2013).

RESULTADOS E DISCUSSÃO

O primeiro passo foi implementar uma câmera virtual, que consiste na tela de visualização do desenho. A câmera é representada por uma matriz que contém coordenadas da posição da câmera no eixo x,y e z. Os elementos desenhados na tela sofrem transformações nas suas posições conforme a câmera é atualizada. Com o auxílio das setas do teclado e do movimento do mouse, o artista movimenta a câmera, isto é, as coordenadas da matriz da câmera são modificadas e os elementos são reposicionados de acordo com essas coordenadas. Desta forma, um ambiente de livre movimentação é conseguido e o artista pode desenhar os personagens e elementos em

qualquer lugar deste mundo virtual. Para orientação do posicionamento dos objetos e da movimentação de câmera, foi criada uma grade que contém 16 linhas dispostas verticalmente e horizontalmente.

Definida a câmera, agora são implementadas as primitivas geométricas que podem ser desenhadas. O ponto é representado no espaço por meio de um vértice apenas que contenha coordenadas nos eixos x e y , obtidos pelas coordenadas do ponteiro do mouse na tela de desenho a partir do clique do botão esquerdo. Uma reta é definida por dois pontos, a partir das coordenadas do clique e arrastar do mouse. De forma similar é obtido o quadrado e o retângulo, onde o clique define as coordenadas do primeiro vértice do quadrilátero e o arrastar e soltar do mouse definem as coordenadas dos outros três vértices. O círculo é definido a partir do centro com o clique do mouse.

Arrastando e soltando, o raio do círculo é calculado a partir das coordenadas do mouse.

Para obtenção da animação são definidos os quadros principais, posicionando a câmera no ângulo em que se deseja gerar a cena e apertando a tecla K no teclado. As posições dos objetos são armazenados em uma estrutura de dados. Após apertar a tecla Enter, uma nova janela se abre, reproduzindo a animação gerada. A geração da animação consiste em ler a estrutura de dados que contém a posição dos objetos e gerar as posições intermediárias a partir da interpolação linear. De forma geral a interpolação linear é dada pela equação:



Onde temos dois pontos (x_1, y_1) e (x_2, y_2) e dado uma coordenada x menor que x_2 e maior que x_1 , a coordenada y é encontrada e é definido o ponto (x, y) (HAZEWINKEL, 1997).

CONCLUSÃO

O Projeto Lucky Rabbit foi construído a partir da linguagem de programação C++ em conjunto com a API OpenGL. Ainda em construção, o projeto já contém uma tela de desenho que contém uma grade com 16 linhas e 16 colunas o qual serve de base para o posicionamento dos objetos. O desenho é feito pelo mouse, onde são desenhadas primitivas geométricas (pontos, linhas, quadriláteros e círculos) a partir das coordenadas do mouse. São selecionados quadros principais para a cena a partir da tecla K e a animação é gerada com a tecla Enter. A movimentação de câmera funciona a partir do movimento do mouse e setas do teclado. Ainda será implementado no projeto o desenho livre pelo mouse e a implementação de uma técnica de interpolação mais precisa, para gerar animações mais completas. É desejado, também, a implementação do desenho por meio de uma mesa digitalizadora e a modelagem de elementos tridimensionais, a fim de construir cenas com elementos

3D que interagem com os desenhos 2D.

REFERÊNCIAS

BEANE, Andy. **3D Animation Essentials**. Indianapolis: Sybex, 2012. 352 p.

HAZEWINKEL, Michiel. **Encyclopaedia of Mathematics: Supplement Volume I**. New York: Springer, 1997. 588 p.

JOHNSTON, Ollie; JOHNSTON, Clie; THOMAS, Frank. **The Illusion of Life: Disney Animation**. Glendale: Disney Editions, 1995. 576 p.

SHREINER, Dave; SELLERS, Graham; KESSENICH, John. **OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.3**. 8. ed. Boston: Addison-wesley Professional, 2013. 935 p.

STROUSTRUP, Bjarne. **The C++ Programming Language**, 4th Edition. 4. ed. Boston: Addison-wesley Professional, 2013. 1368 p.

THOMAS, Bob. **Walt Disney The Art Of Animation: The Story Of The Disney Studio Contribution to A New Art**. New York: Simon And Schuster, 1958. 181 p.

WHITED, Brian; DANIELS, Eric; KASCHALK, Michael; OSBORNE, Patrick; ODERMATT, Kyle . **Computer-Assisted Animation of Line and Paint in Disney's Paperman**. Los Angeles: Siggraph, 2012. 1 p.