

Estrutura do TCC

Cristhian Grundmann

1 Introdução

O projeto consiste num software de visualização de curvas e superfícies, utilizando OpenGL (versão 4.6), C++, GLFW, etc. O usuário poderá definir (em uma linguagem própria) objetos de alguns tipos, incluindo curvas, superfícies, funções gerais e parâmetros. O texto é processado e os objetos podem ser desenhados e manipulados. O objetivo primário do projeto é o geodesic tracing: a visualização intrínseca de uma superfície por meio de geodésicos, de forma análoga à técnica de ray tracing.

A aplicação será uma janela com um painel desenhado inteiramente pelo OpenGL, com a biblioteca GUI <https://github.com/ocornut/imgui>.

O livro “Compilers: Principles, Techniques & Tools (segunda edição)” será a referência para a construção dos compiladores.

A especificação online de OpenGL: “The OpenGL R© Graphics System: A Specification (Version 4.6 (Core Profile) - October 22, 2019)” em <https://www.khronos.org/opengl/> será referência para as técnicas gráficas, juntamente com um livro online sobre os usos **típicos** de OpenGL: <https://learnopengl.com/>, ambos gratuitos.

2 Revisão teórica

uma breve revisão teórica sobre curvas e superfícies (com ênfase em superfícies), incluindo várias referências. Os principais elementos são: forma parametrizada, espaços tangente e geodésicos.

A teoria necessária sobre outros aspectos do projeto serão brevemente explicadas em suas seções, como em compiladores, OpenGL e shaders.

3 Compilador

O compilador processa o texto escrito pelo usuário. Há 3 estágios de compilação.

3.1 Linguagem de descrição

Nessa subseção será definida a linguagem de descrição. Cada texto na linguagem também é chamado de programa.

3.2 Análise léxica

Serve para abstrair as “palavras” do texto, por exemplo, identificar palavras chave, números, e outros símbolos. Também verifica a validade das palavras.

3.3 Análise sintática

Serve para abstrair as “formas gramaticais” das sentenças, por exemplo, resolvendo a ordem dos operadores, identificando declarações de objetos, entre outras coisas. Também serve para verificar a validade da gramática.

3.4 Geradores (ou “Análise semântica e síntese”)

Servem para verificar a validade dos objetos definidos conforme algumas regras. Além disso, geram algum produto final. Um gerador trivial não gera nada nem valida os objetos, e serve apenas para verificar a léxica e a sintaxe do texto. O gerador mais importante verifica algumas condições nos objetos e gera estruturas de dados mais tratáveis. Assim pode fazer derivadas simbólicas para gerar funções auxiliares automaticamente, além de gerar shaders que podem finalmente renderizar os objetos. Um gerador mais simples apenas aplica syntax highlighting no texto (provavelmente o software não terá isso).

A partir dessas estruturas, por exemplo, controles deslizantes poderão ser criados para os parâmetros, sem ser necessário shaders específicos para isso.

4 Shaders e Técnicas

Algumas técnicas serão utilizadas na renderização. Por exemplo, uma textura de normais será criada para cada superfície, para poder renderizá-la com alguma iluminação, aumentando a compreensão. Outras texturas auxiliarão outros processos. Por exemplo, uma cópia da visualização 3d será feita numa textura, porém as cores da textura indicarão as parâmetros UV. Isso facilita a seleção de pontos na janela 3d. Outro exemplo é a textura da primeira forma fundamental, onde $RGBA = EFGA$, onde EFG formam a primeira forma fundamental e A é o elemento de área calculado com EFG. Assim, se uma textura indica uma região, será fácil estimar sua área, basta somar as pequenas áreas estimadas dos pixels da textura calculada.

Os shaders serão gerados automaticamente pelo compilador.

Possivelmente haverá a opção de gerar texturas através de um shader escrito pelo usuário, porém a linguagem do shader pode ser muito avançada para um usuário comum.

5 Interface gráfica

Com o Dear ImGui, uma interface completa será criada, incluindo caixa de texto para a descrição dos objetos, botões de controle(ex: compilar), e controles num menu(ex: salvar, abrir, exemplos, ajuda, etc).

Após a compilação, algumas subjanelas são criadas conforme o texto (programa), por exemplo: visualização do geodesic tracing, renderização tradicional em 3D, imagens customizadas, etc. Parâmetros dos programas terão controles deslizantes gerados no ImGui.

Todas as texturas geradas poderão ser salvas pelo usuário. As texturas customizadas serão dadas pelo usuário.

6 Geodesic tracing

Essa seção deve definir e explicar a técnica, incluindo um método numérico para a resolução das EDOs que governam as geodésicas. Os métodos de Euler e de Runge-Kutta serão comparados.

Superfícies interessantes serão estudadas e descritas nessa seção: cone, esfera, faixa de Möbius e garrafa de Klein são exemplos.