

Projekt do předmětu VGE – Výpočetní geometrie

Extrakce 3D povrchu z volumetrických dat

řešitelé: **Boris Burkalo**, xburka00
Tomáš Dubský, xdubsk08

Zadání

V rámci našeho zadání jsme měli implementovat tři algoritmy na extrakci 3D povrchu z volumetrických datasetů. Těmito třemi algoritmy jsou:

- [Marching Cubes](#)
- [Marching Tetrahedra](#)
- [Flying Edges](#)

Po implementaci zmíněných algoritmů jsme je měli porovnat v rámci závěrečné prezentace.

Použité technologie

Rozhodli jsme se vytvořit malou knihovnu, která tyto tři algoritmy implementuje. Knihovna je napsána v jazyce C++. Kromě standardní knihovny C++ jsme využili i matematickou knihovnu [GLM](#) pro usnadnění práce s vektory. Překlad probíhá pomocí nástroje CMake, a tudíž knihovna je multiplatformní.

Použité zdroje

K implementaci algoritmu Marching Cubes a Marching Tetrahedra byly použity materiály:

- [Marching cubes: A high resolution 3D surface construction algorithm](#) – originální článek popisující algoritmus
- [Polygonising a scalar field](#) – dokument popisující oba algoritmy

K implementaci algoritmu Flying Edges byly použity materiály

- [Flying edges: A high-performance scalable isocontouring algorithm](#) – vědecký článek popisující algoritmus Flying Edges od jeho tvůrců
- [miniIsosurface](#) – knihovna implementující algoritmus Flying Edges
- [The Visualization Toolkit](#) – knihovna implementující algoritmus Flying Edges

Nejdůležitější výstupy

Každý ze tří implementovaných algoritmů dokáže úspěšně sestavit 3D povrch z libovolného volumetrického datasetu.

Výstupem je kompletní [Wavefront .obj soubor](#), který lze importovat do libovolného 3D grafického nástroje.

Knihovnu je možné připojit k libovolnému volumetrickému rendereru a používat ji k sestavení povrchu z volumetrických dat v rámci jiné aplikace.

Ovládání vytvořeného programu

Program se spouští z příkazové řádky příkazem:

```
./SurfaceExt -a <algo> -i <input file> -o <output>.obj -iso <isovalue>
```

kde:

- <algo> je jeden z $[c \mid t \mid e]$, kde c značí Marching Cubes, t značí Marching Tetrahedra a e značí Flying Edges
- <input file> je jeden z testovacích vstupů:
 - bonsai.raw
 - foot.raw
 - head.raw
 - manix.raw

- skull.raw
- teapot.raw
- tooth.raw
- <output> je libovolný název výstupního souboru, který musí končit příponou .obj
- <isovalue> je desetinné číslo v rozsahu $(0, 1)$, které značí hranici mezi vnitřkem objemu a vnějškem.

Rozdělení práce v týmu

Tomáš Dubský: implementace algoritmu Marching Cubes a Marching Tetrahedra, vytvoření kostry programu

Boris Burkalov: implementace algoritmu Flying Edges, napsání dokumentace.

Co bylo nejpracnější

Algoritmus Marching Cubes je poměrně jednoduchý, a navíc lze najít mnoho implementací, takže šel implementovat jednoduše. Marching Tetrahedra byl náročnější, protože původní článek je přístupný pouze po zaplacení. Musel jsem tedy vycházet z neoficiálního zdroje a implementace, která využívala obrovský switch místo tabulky, což se mi vůbec nelíbilo. Takže nejtěžší práce byla vyextrahovat z tohoto switchu správnou tabulku indexů, aby implementace fungovala podobně jako Marching Cubes.

V rámci implementace algoritmu Flying Edges bylo nejpracnější se vůbec v algoritmu zorientovat. Jelikož na rozdíl od algoritmů Marching Cubes a Marching Tetrahedra neexistuje mnoho návodů a již implementovaných řešení, bylo složitější algoritmus pochopit a následně jej implementovat.

Zkušenosti získané řešením projektu

V rámci řešení projektu jsme se naučili jak je možné převést jak jednoduché, tak složité volumetrické datasety na jim ekvivalentní 3D povrchy, se kterými je následně možné libovolně pracovat. Dále bylo pro nás přínosné implementování daných metod na základě vědeckých článků.

Autoevaluace

Koncept řešení: 95% (analýza, výběr článků, dekompozice problému, volba vhodných prostředků, ...)

Materiály, vybrané k řešení zadání byly kvalitní a dostačující.

Realizace: 95% (kvalita získaných znalostí, kvalita a čitelnost kódu, obecnost řešení, znovupoužitelnost, ...)

Znalosti získané díky řešení projektu pro nás byly velice důležité. Zároveň si myslíme, že kód, který jsme napsali je kvalitní a čitelný. Knihovna byla námi implementována na Microsoft Windows 10 a Ubuntu 22.04, takže funguje na více platformách.

Využití zdrojů: 90% (využití existujícího kódu a dat, využití literatury, ...)

V rámci řešení jsme využili již existující řešení daného problému.

Hospodaření s časem: 100% (rovnoměrné dotažení částí projektu, míra spěchu, chybějící části, ...)

Projekt byl dokončen podle očekávání.

Spolupráce v týmu: 100% (komunikace, dodržování dohod, vzájemné spolehnutí, rovnoměrnost, ...)

Každý člen týmu udělal své části podle domluvy ve stanoveném termínu. Komunikace v týmu byla výborná.

Celkový dojem: 85% (pracnost, získané dovednosti, užitečnost, volba zadání, cokoliv, ...)

Vypracování projektu nebylo nejsložitější. Kód je však dle našeho názoru velice užitečný a použitelný i pro jiné projekty. Zadání bylo velice zajímavé.

Doporučení pro budoucí zadávání projektů

Zadávání projektu a projekt jako takový byl velice zajímavý. Líbil se nám styl konzultací, které k projektu byly povinné, jelikož pomůžou k usměrnění a stanoví očekávání, které zadávající od projektu má. Dále nám vyhovovalo, že na projekt je hodně času a tedy jsme jej mohli pečlivě vypracovat.