

Oponentský posudek diplomové práce

Jakub Hlaváček: Zpracování medicínských dat na GPU

Jakub Hlaváček se ve své práci zabývá zpracováním objemových dat na moderních grafických akcelerátorech. Cílem jeho práce bylo vytvořit framework pro snadné programování algoritmů zpracovávajících nebo zobrazujících objemová data na platformě .NET. Snadnost programování, možnost vizualizace výsledků v reálném čase (dokonce interaktivně) a efektivní využití moderních vícejádrových CPU i GPU – to byly hlavní cíle, kterých měl diplomant ve svém projektu dosáhnout.

Je zřejmé, že se to Jakobovi povedlo. Zpracoval nejdříve řešerši, do které zahrnul všechny významnější systémy v současné době používané (žádný z nich však nesplňoval všechny kladené požadavky) a nakonec navrhnul a implementoval vlastní systém VL, ve kterém kladl důraz na jednoduchost používání „uživatelé-programátorem“. Ač nejsem nadšencem pro jazyk C# (nebo .NET platformu obecně), musím uznat, že se v systému VL zřejmě budou psát pilotní nebo i finální aplikace snadno a za podpory mnoha moderních programovacích technologií (chránících např. programátora před špatně lokalizovatelnými chybami). Závěrečné testy prokázaly, že je sice VL platforma o trochu pomalejší než optimalizovaný C++ kód v běžném programu, díky chytrému a efektivnímu využití moderních vícejádrových procesorů a dobře navrženému systému řízení výpočtu (v podstatě zadarmo například programátor získá podporu pro paralelní výpočet) je celková rychlost načítání, zpracování a zobrazení minimálně tak dobrá, jako u běžných aplikací. Dále musím vyzdvihnout pěkně zpracovanou instruktážní část pro „uživatelé-programátora“ a dobrou kulturu zdrojových souborů. V neposlední řadě stojí za zmínku zajímavá technika, kterou diplomant vymyslel a realizoval – slouží k celkem pohodlné kombinaci mnoha variant „shaderů“ (to je v běžném programování GPU často velice obtížný problém – skládat dohromady několik komponent tak, aby šly použít všechny kombinace).

Práce je psaná slušnou angličtinou, je srozumitelná a vhodně doplněná obrázky a diagramy. Mám pouze pár připomínek / otázek:

- při čtení textu mi připadal trochu násilný přechod mezi kapitolami 2 a 3. Pár úvodních vět by jistě lépe uspokojilo čtenáře, který je nyní z řešerše vržen rovnou do návrhu API
- v tabulce 10 jsou uvedeny časy výpočtu pro několik variant implementace přístupových metod. Časy jsou uvedeny s přesností na setiny sekundy, což navozuje dojem, že byly získány průměrováním mnoha nezávislých měření. Ale přitom je ve většině případů „kontrolovaný“ kód rychlejší než ten „nekontrolovaný“ (což je nelogické a mohlo se snad náhodou stát při jednom spuštění testu..). Rád bych věděl, jak to bylo?

Tyto drobnosti nemohou zkazit výborný dojem z odevzdané práce, jako oponent **doporučuji** připustit Jakuba Hlaváčka k obhajobě a **uznat** předložené dílo jako diplomovou práci.

Praha, 16. května 2008

RNDr. Josef Pelikán, KSVI