

Posudek oponenta diplomové práce

Student: Macenauer Pavel, Ing.
Téma: Detekce objektů na GPU (id 16358)
Oponent: Polok Lukáš, Ing., UPGM FIT VUT

1. **Náročnost zadání** průměrně obtížné zadání
2. **Splnění požadavků zadání** zadání splněno s drobnými výhradami

Autor prostudoval a popsal metody detekce objektů pomocí algoritmu WaldBoost a LBP. Bylo navrženo několik způsobů paralelizace, které byly implementovány a porovnány. Měření úspěšnosti detekce v práci zcela chybí, měření výkonu je provedeno avšak chybí zde souhrnná tabulka všech měření - autor slibuje měření pěti různých algoritmů na čtyřech různých datových sadách, avšak jediné výsledky jsou měření čtyř algoritmů na neznámé sadě. Zadání ale bylo v podstatě splněno.
3. **Rozsah technické zprávy** je v obvyklém rozmezí

Rozsah technické zprávy je v obvyklém rozsahu, možná mírně delší.
4. **Prezentační úroveň předložené práce** 70 b. (C)

Práce je strukturována logicky a jednotlivé kapitoly na sebe navazují. Avšak, chybí zde nějaký formální úvod, práce začíná desetistránkovou kapitolou o GPGPU, obsahující dosti vyčerpávající historii a popis architektury. Obvyklý odstavec s popisem rozdělení obsahu práce do následujících kapitol tu není. Rozsahy dalších kapitol jsou přiměřené. Není jasné s jakými daty byly naměřeny grafy na obrázcích 5.1 - 5.3. Vyhodnocení detekce na CPU v práci chybí, i když je v textu zmiňované. Text je místy mírně nepřehledný, některé implementační detaily jsou nejasné.
5. **Formální úprava technické zprávy** 70 b. (C)

Obsah práce je velmi dlouhý, bylo by záhodno skrýt v něm kapitoly třetích úrovní. Na straně 6 přetéká kus kódu za okraj textu. Ve výpisech kódu nejsou nějak zvýrazňována klíčová slova. Některé vzorce (například v kapitole 2) nejsou číslovány. Na straně 20 se poznámka pod čarou pomíchala s textem. Na straně 21 je " $2^{(1/\text{number_of_images})}$ " vysázeno zcela nevhodným způsobem.

Práce je psána anglicky, na dobré úrovni až na pár drobných překlepů. Anglický a český abstrakt si neodpovídají, zatímco český abstrakt popisuje co je cílem práce, anglický to samé prohlašuje za její přínosy.
6. **Práce s literaturou** 90 b. (A)

Práce cituje velké množství povětšinou kvalitních zdrojů, citace jsou v textu dobře rozmístěny, byť spíše na konci kapitol než na začátku.
7. **Realizační výstup** 85 b. (B)

Realizačním výstupem je funkční implementace čtyř různých schémat pro rychlý výpočet kaskády klasifikátorů, založených na LBP příznacích.
8. **Využitelnost výsledků**

Jedná se o práci kompilačního charakteru, implementuje však schéma konsolidace aktivních vláken jež by řešilo zásadní nevýhodu předchozích GPU implementací, a sice nutnost počítat klasifikaci nad pixely které byly zavrženy jako neobsahující detekovaný objekt. Taková metoda dokáže snížit zbytečné využití GPU, avšak sama o sobě nedokáže snížit výpočetní čas! Na to by bylo nutné počítat poslední "přežívající" pixely paralelně, což se podle všeho neděje nebo to aspoň není z textu jasné. Bylo by také rozumné experimentovat s počtem a pozicemi kroků konsolidace.
9. **Otázky k obhajobě**
 - Dají se vlákna synchronizovat globálně pomocí globálních atomických operací, například pomocí atomické výměny?
 - Jaký vliv by na práci mělo využití dynamického paralelismu?
 - Jak se výsledky porovnávají s existujícími GPU implementacemi LBP?
10. **Souhrnné hodnocení** 70 b. dobře (C)

I přesto, že se práce věnuje zajímavému tématu paralelizace detekce objektů pomocí WaldBoost na GPU, dopouští se autor v textu technické zprávy mnoha formálních chyb a samotná implementace přináší jiné než očekávané výhody: i přes to, že se podařilo snížit zbytečné využití GPU a tedy i snížit spotřebu, nemohou navržené přístupy zvýšit rychlost. Doba detekce bude vždy, pokud je v obraze detekován alespoň jeden objekt, omezena časem výpočtu všech klasifikátorů pro jeden pixel: klasifikátory jsou vyhodnocovány sériově. Pokud

bylo cílem zvýšit rychlost, je navržené schéma konsolidace aktivních vláken bez paralelního vyhodnocování klasifikátorů spíše na obtíž.

V Brně dne: 11. června 2015

.....
podpis