

О. О. Кулик

А. А. Яровий

Д. Г. Пасічник

О. В. Небожанов

ПРИКЛАДНІ АСПЕКТИ ЗАСТОСУВАННЯ ГЕТЕРОГЕННИХ ОБЧИСЛЕНЬ ПРИ РОЗРОБЦІ СИСТЕМ ВІРТУАЛЬНОЇ РЕАЛЬНОСТІ

Вінницький національний технічний університет

Анотація

Розроблено програмний засіб на основі гетерогенної програмної платформи із використання CPU та GPU ядер, призначений для побудови 3D-сцен, що дозволило досягти підвищення швидкодії процесу оброблення 3D-сцен.

Ключові слова: гетерогенні обчислення, віртуальна реальність, GPGPU, 3D-сцени.

Abstract

The heterogeneous software platform based software package for 3D-scene creation with using of CPU and GPU cores was developed, which allowed to achieve 3D-scene processing speed increase.

Keywords: heterogeneous computing, virtual reality, GPGPU, 3D-scenes.

Вступ

З розвитком сучасних технологій в багатьох галузях постає потреба у розв'язанні задач великої та надвеликої обчислювальної складності [1]. Зокрема, це є актуальним при розробці систем віртуальної реальності (ВР), тобто створення технічними засобами такого світу (об'єктів та суб'єктів), який передається людині через її органи відчуттів та сприймається нею як справжній, оскільки чим складнішою є 3D-сцена, тим більшими стають вимоги до програмного та апаратного забезпечення [2]. Більш того, для комфортного сприйняття ВР важливим є якість та стабільність (частота кадрів у секунду тощо) зображення, що додатково підвищує вимоги до швидкодії.

Наразі існує велика кількість технологій високопродуктивних обчислень, проте одними з найбільш перспективних, у тому числі при реалізації ВР-технологій, є технології гетерогенних обчислень, у тому числі через те, що вони за своєю структурою добре підходять для обробки графічних даних [3, 4, 5].

Результати дослідження

Для задачі моделювання з метою оцінки доцільності застосування технологій гетерогенних обчислень було обрано задачу відтворення візуальних ефектів грози шляхом використання системи частинок. Ефект грози включає в себе частинки крапель дощу, частинки бризок від крапель на поверхні, частинки опалого листя з дерев, частинки блискавок, та частинки легкого туману. Вищенаведені частинки було реалізовано двома способами: на основі CPU-ядер та на основі GPU-ядер. При реалізації на основі CPU-ядер частинки обробляються і обчислюються цілком за допомогою центрального процесора. При використанні GPU-ядер частинки спочатку з'являються на процесорі, а потім обробляються і обчислюються цілком за допомогою графічної карти.

В обох випадках частинки в цілому поведуться аналогічно, але все ж таки мають деякі ключові відмінності. Деякі функції, доступні на CPU (наприклад, світловипромінювання), не підтримуються на GPU. Однак, обчислення частинок на GPU-ядрах є набагато ефективнішим методом. Візуальний ефект, що побудований на CPU-ядрах, може нормально відтворювати лише тисячі частинок. При обробці більшої кількості, деякі з них можуть бути відтворені некоректно або не відтворені взагалі через обмеження потужності CPU. Візуальний ефект, що побудований на GPU-ядрах, в свою чергу дозволяє створювати десятки і сотні тисяч частинок без значного впливу на продуктивність, що дуже важливо при відтворенні масштабних ефектів, таких як снігопад, дощ або іскри. Принциповим моментом є, у тому числі, велика кількість ядер у сучасних графічних картах, яка дозволяє в повній мірі використати усі переваги технологій паралельних обчислень і обробляти багато тисяч частинок одночасно.

Тестування проводилось в спеціально побудованій 3D-сцені у трьох режимах: без використання візуальних погодних ефектів, з використанням візуальних погодних ефектів побудованих на основі GPU-ядер та з використанням візуальних погодних ефектів побудованих на основі CPU-ядер. При тестуванні вимірювалась частота кадрів (fps) обробки побудованої 3D-сцени, тобто частота, з якою на дисплеї з'являються послідовні зображення, звані кадрами.

Кожне тестування здійснювалось протягом 60 секунд. Частота кадрів вимірювалась за допомогою програми Fgaps. Параметрами частоти кадрів були кількість кадрів за весь час тестування, мінімальна та максимальна частота кадрів, середня частота кадрів за весь час тестування. Тестування здійснювалось на персональному комп'ютері такої конфігурації: операційна система – Windows 10 version 1703(Fall Creators Update); центральний процесор - Intel Core I5 4690K(4ядра архітектури,4 потоки, частота 3.5 hz); відеоадаптер – Nvidia Geforce GTX980 (2048 ядер архітектури Maxwell, 4Gb відеопам'яті, частота відеопроцесора 1152, частота відеопам'яті 7010); оперативна пам'ять — 8Gb,1866hz.

Таблиця 1 – Результати тестування

	Час тестування	Мінімальна частота кадрів	Максимальна частота кадрів	Сумарна кількість кадрів	Середня частота кадрів
Без частинок	60000ms	60	68	4002	66.700
3 частинками на основі GPU	60000ms	60	67	3869	64.483
3 частинками на основі CPU	60000ms	51	66	3494	58.233

Висновки

Отримані результати досліджень підтверджують доцільність застосування технологій гетерогенних обчислень при розробці систем віртуальної реальності, у тому числі при побудові 3D сцен. Окрім того, перспективним може виявитися подальша інтеграція технологій віртуальної реальності з іншими технологіями паралельних обчислень, у тому числі з Multi-GPU Programming.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Форсайт Д. Компьютерное зрение. Современный подход / Форсайт Д., Понс Д. – М.: Мир, 2004. – 610с.
2. Що таке віртуальна реальність? [Електронний ресурс]. Режим доступу - <https://futurum.today/shcho-take-virtualna-realist/>
3. Гергель В.П. Высокопроизводительные вычисления для многоядерных многопроцессорных систем. / Гергель В.П. – Н.: ННГУ им. Н.И.Лобачевского, 2010. – 421 с.
4. Яровий А. А. Паралельно-ієрархічне перетворення плямоподібних зображень на основі Multi-GPU систем / Яровий А. А., Кулик О. О., Кокряцька Н. І.// Інформаційні технології та комп'ютерна інженерія. – 2015. – №3(34). – С. 72-80.
5. Пасічник Д.Г. Аналіз підходів до підвищення якості растрових зображень на основі технології GPGPU / Пасічник Д.Г., Яровий А.А.: Збірник матеріалів XLV Науково-технічної конференції Вінницького національного технічного університету, (Вінниця, 23-24 березня 2016 р.). – В.: ВНТУ, 2016. – С. 1-4. [Електронний ресурс] – Режим доступу: <http://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2016/paper/view/510/404>

Кулик Олександр Олександрович — аспірант кафедри комп'ютерних наук, Вінницький національний технічний університет, м. Вінниця, Хмельницьке шосе, 95, e-mail: o.kulyk@vntu.edu.ua.

Яровий Андрій Анатолійович — д.т.н., професор, завідувач кафедри комп'ютерних наук, Вінницький національний технічний університет, м. Вінниця, Хмельницьке шосе, 95, e-mail: a.yarovyy@vntu.edu.ua.

Пасічник Дмитро Геннадійович — аспірант кафедри комп'ютерних наук, Вінницький національний технічний університет, м. Вінниця, Хмельницьке шосе, 95.

Небожанов Олександр Володимирович — магістр комп'ютерних наук.

Olexandr O. Kulyk — Postgraduate Student of Computer Science Department, Vinnytsia National Technical University, Vinnytsia, Khmelnytske Shose, 95, e-mail: o.kulyk@vntu.edu.ua.

Andrii A. Yarovyi — Doctor of Science (Eng.), Professor, Head of Computer Science Department, Vinnytsia National Technical University, Vinnytsia, Khmelnytske shose, 95, e-mail: a.yarovyy@vntu.edu.ua.

Dmytro G. Pasichnyk — Postgraduate Student of Computer Science Department, Vinnytsia National Technical University, Vinnytsia, Khmelnytske Shose, 95.

Olexandr V. Nebozhanov — Master of Computer Science.