

ДОСЛІДЖЕННЯ ЧАСОВОЇ ЕФЕКТИВНОСТІ ВІДОБРАЖЕННЯ КОМП'ЮТЕРНОЇ ГРАФІКИ РЕАЛЬНОГО ЧАСУ НА БАГАТОЯДЕРНИХ СИСТЕМАХ З ВИКОРИСТАННЯМ OPENGL ТА VULKAN API

Автор – Поліщук І.А., студент групи ПЗ1921

Науковий керівник – к.т.н., доц. Іванов О.П.

Дніпровський національний університет імені академіка В. Лазаряна

Україна

Передові досягнення науки і техніки в області комп'ютерної анімації, такі як імітація руху або відбиття світла загалом є дуже критичними до часу виконання завдань, які на них покладаються. Задачі, які вирішують більшість систем відображення комп'ютерної графіки є задачами так званого жорсткого реального часу (коли перевищення часу виконання поставлених завдань може призвести до невідворотних наслідків) або м'якого реального часу (коли перевищення часу вирішення небажане, але припустиме).

Саме тому ми звертаємо нашу увагу на можливість розподілу таких задач на декілька ядер процесору за для зменшення загального часу обробки кожного кадру анімації перед його відображенням.

Для виводу графіки на екран використовують центральний процесор та графічний процесор. Центральний процесор обчислює позицію, зміщення, форму об'єктів та передає ці дані до графічного процесору. Графічний процесор у свою чергу перетворює дані таких об'єктів у форму придатну для подальшого виведення на екран.

Історично склалося, що центральний процесор та графічний процесор розвиваються окремо один від одного. Тому зараз, коли центральний процесор досяг успіху у паралельному виконанні задач на декількох ядрах, одноядерні підходи до обробки та передачі даних на графічний процесор не є ефективними.

Один з таких підходів є використання OpenGL – програмного інтерфейсу до графічного пристрою. Цей інтерфейс розроблявся у 90-их років, коли багатоядерні процесори ще не були досить популярними. Тому методи до обробки та передачі даних які запроваджує цей інтерфейс не дає змогу робити це паралельно, оскільки будь-яка операція з OpenGL може змінювати його глобальний стан, який не передбачає одночасні зміни і не запроваджує синхронізації щоб захиститися від стану гонитви (ситуація в якій робота чи результат операції залежить від послідовності або тривалості виконання).

Щоб виправити такі недоліки може бути використаний інтерфейс Vulkan. Цей інтерфейс не використовує глобальних об'єктів які неможливо синхронізувати. Натомість робота з ним є складніша, оскільки об'єкти які були сховані у OpenGL відтепер мають бути створені та використані розробником.

Одним із таких об'єктів є командний буфер. Командний буфер це буфер, який зберігає заздалегідь заповнені команди які має виконати графічний процесор. Кожен командний буфер може використовуватись незалежно один від одного, тому кожен потік може обчислювати інформацію об'єктів та заповнювати такі буфери окремо. У кінці, ці буфери мають бути відправлені до графічного процесору, де з них буде створене зображення.

Інші дослідження зосереджують увагу на порівнянні різниці використання одного та усіх ядер процесору. У цьому дослідженні буде додатково з'ясовано, як саме буде змінюватись ефективність відображення анімації з використанням OpenGL та Vulkan зі збільшенням кількості ядер процесору, які можливості дає Vulkan з обробки об'єктів анімації паралельно на процесорі, які методи синхронізації мають бути використані за для такої паралельної обробки та залежність ефективності від кількості та складності анімованих об'єктів.