

УДК 004.925.3

*Горшенін М.О., магістрант,
Горшенін О.Є., к.т.н., доцент
Державний університет «Житомирська політехніка»*

ПОРІВНЯЛЬНИЙ АНАЛІЗ CPU- ТА GPU- ОРІЄНТОВАНИХ ПІДХОДІВ ДО ВІДСІКАННЯ ОБ'ЄКТІВ У РЕНДЕРИНГУ РЕАЛЬНОГО ЧАСУ

Зі зростанням складності тривимірних сцен, та збільшенням кількості сцен з великою кількістю дрібних або повністю прихованих об'єктів питання ефективного відсікання невидимих об'єктів набуває критичного значення для рендерингу в реальному часі. Існують два фундаментальні підходи: відсікання на основі CPU, що використовує центральний процесор, і відсікання на основі GPU, де відсікання інтегрується у графічний конвеєр та виконується паралельно на тисячах потоків.

Незважаючи на велику кількість окремих досліджень щодо відсікання, застосування ієрархічних структур та GPU-паралелізму, у відкритій літературі відсутній системний порівняльний аналіз однотипних стратегій, реалізованих на CPU і GPU, із фіксацією їх реальних переваг, обмежень та сценаріїв доцільності застосування. Для проведення порівняння створено уніфіковане тестове середовище, що забезпечує повторювані експериментальні умови та ідентичні вхідні дані для всіх алгоритмів. Інструментальною основою обрано Unity Scriptable Render Pipeline . На базі SRP реалізовано набір алгоритмів відсікання, які повністю дублюють один одного на CPU та GPU.

Окрім цього, створено набір стандартизованих сцен-шаблонів зі змінною щільністю розташування об'єктів, їх розмірами та рівнем оклюзії. Це дозволяє перевірити алгоритми в умовах різного просторового розподілу геометрії та різної структури видимості.

Для кожного алгоритму проведено серію вимірювань, що дозволяють визначити: обчислювальну вартість: час виконання; ефективність відсікання: частку відкинутих об'єктів; поведінку у граничних випадках: сцени без оклюзії, повну оклюзію, надзвичайно велику кількість дрібних об'єктів; залежність продуктивності від кількості інстанцій об'єктів; взаємодію технік: які стратегії підсилюють одна одну, а які дублюють функціональність.

Алгоритми на CPU реалізовано як в однопотоковому режимі, так і з використанням багатопотокового паралелізму на основі Jobs/Burst. Методи на GPU реалізовано за допомогою обчислювальних шейдерів.

Особливу увагу приділено структурі HZB як універсальному ресурсу, здатному бути повторно використаним, що знижує загальну обчислювальну вартість рендерингу.

Також у роботі досліджено недоцільність обходу BVH-структур на GPU, зумовлену розбіжністю потоків. На основі попередніх вимірювань та даних із останніх досліджень очікується встановлення таких закономірностей: GPU-підходи забезпечують найменший час відсікання, особливо в поєднанні відсікання за пірамідою видимості та HZB-відсікання, завдяки паралельному виконанню та відсутності синхронізацій між CPU і GPU; CPU з BVH перевищує GPU за ефективністю у сценах із низьким рівнем оклюзії та упорядкованим розташуванням об'єктів, де деревоподібна структура прискорює відсікання цілих піддерев; обхід BVH на GPU є недоцільним для сцен із великою кількістю дрібних об'єктів через значні гілкування потоків виконання та втрату ефективності паралелізму; існують випадки, де відсікання взагалі не дає виграшу в продуктивності; комбіновані стратегії забезпечують найкращий баланс між точністю відсікання та швидкістю виконання. Прогнозується, що повністю GPU-орієнтований конвеєр має забезпечити найбільший приріст продуктивності на великих сценах (>50 тис. об'єктів), тоді як CPU+BVH залишатиметься конкурентоспроможним до 10 тис. об'єктів.

Подальші дослідження спрямовано на узагальнення отриманих даних для масштабних сцен, а також на вивчення адаптивних комбінованих стратегій, де вибір CPU- чи GPU-методу здійснюється автоматично на основі характеристик сцени.

Список використаних джерел:

1. Lee G. B., Lee S. Iterative GPU Occlusion Culling with BVH. Proceedings of High-Performance Graphics (HPG '20). Washington, D.C., USA, 2020. P. 1–2. DOI: 10.2312/hpg.20201194. URL: https://www.highperformancedgraphics.org/posters20/04_lee_iterative_occlusion_culling_abstract.pdf (дата звернення: 24.11.2025).
2. Schütz M., Kerbl B., Wimmer M. Software Rasterization of 2 Billion Points in Real Time. 2022. Vol. 5, No. 3. P. 1–16. DOI: 10.1145/3543863. URL: <https://www.cg.tuwien.ac.at/research/publications/2022/SCHUETZ-2022-PCC/> (дата звернення: 24.11.2025).