

## ВИДИМІСТЬ І ОКЛЮЗІЯ В VR

Сушинський І.К.

Науковий керівник – ст.в. Новіков Ю.С.

Харківський національний університет радіоелектроніки  
(61166, Харків, пр. Науки, 14, каф. Програмної інженерії,  
e-mail: [ihor.sushynskyi@nure.ua](mailto:ihor.sushynskyi@nure.ua)

Virtual reality (VR) is an interactive computer-generated experience taking place within a simulated environment. Frustum culling is process of discarding objects not visible on the screen. As we don't see them, we don't need to spend resources on computer to prepare it for rendering and rendering itself. Unreal Engine 4 (UE4) provides methods of culling for visibility and occlusion. These culling methods are useful for optimizing game performance. Each method works to reduce the number of visible Actors in the Level by setting whether they should be drawn to the screen or not.

Віртуальна реальність (VR) - створений технічними засобами світ, який передається людині через його відчуття. Користувач може впливати на ці об'єкти в злагоді з реальними законами фізики. Основним пристроєм для передачі VR-зображення є окуляри віртуальної реальності. Розділяють кілька типів VR-окулярів: повністю самостійні; залежні від потужностей смартфона; залежні від потужностей персонального комп'ютера. Основні окуляри представлені в таблиці 1.

Таблиця 1

| VR-окуляри           | Тип                  | Вартість  | Роздільна здатність на кожне око                    |
|----------------------|----------------------|-----------|---|
| Oculus GO            | Standalone           | \$ 199    | 1280x1440   |
| Oculus Quest         | Standalone           | \$ 399    | 1600x1440   |
| Oculus Rift          | PC dependent         | \$ 349    | 1.080x1.200   |
| HTC Vive             | PC dependent         | \$ 499    | 1.080x1.200 / 1.440x1.600(Pro)                      |
| Gear VR              | Smartphone dependent | \$ 129.99 | Залежить від телефону (Samsung Galaxy S9 2960x1440) |
| Google Daydream View | Smartphone dependent | \$ 99.99  | Залежить від телефону (Google Pixel 3 XL 1440x2960) |

Автономна гарнітура VR Oculus GO є найдоступнішим вибором на сьогоднішній день. Основним чіпсетом Oculus GO є Qualcomm Snapdragon 821 з тактовою частотою до 2,4 ГГц і графічним чіпом Andreno 530. При тестуванні даного чіпсета результати тестів не задовільні, що підтверджує необхідність оптимізації VR додатків, для більш плавною картинки, так як будь-який провисання fps буде погано впливати на враження і може негативно впливати на організм користувача. Одним з головних методів оптимізації ігор є Frustum Culling. Основна ідея полягає у визначенні області видимості в формі фрустума, яка проектується від камери (див. Рис. 1).

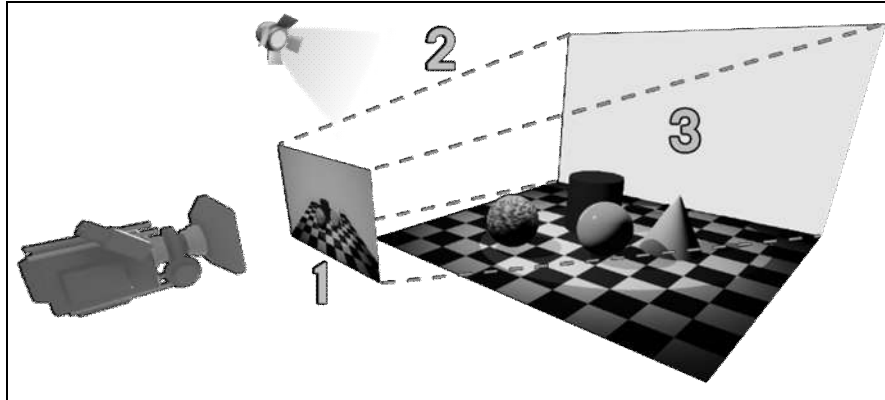


Рисунок 1 – Фрустум/піраміда видимості

Об'єкти, які не потрапляють в зону видимості слід ігнорувати і не посилати на рендер. Для визначення попадання об'єкта в зону видимості, кожен об'єкт можна апроксимувати найпростішою геометрією, наприклад, сферою або боксом. Зокрема, розглянемо застосування Frustum Culling в Unreal Engine 4. Distance Culling - дозволить налаштувати відстань до об'єкта; Dynamic Occlusion - використовується апаратні запити оклюзії, щоб зменшити складність сцен і кількість запитів. Це робиться шляхом відправлення запитів в графічний процесор і подальшого читання їх на один кадр пізніше. Precomputed Visibility Volume - Попередньо обчислені обсяги видимості зберігають.

Для оптимізації продуктивності, необхідно використовувати комбінацію кулінг методів - Distance Culling, для його настройки використовується масив пари значень - розмір і відстань. Оптимальні настройки залежать від типу гри, але досвідченим шляхом були обчислені оптимальні значення: [(0 1000); (128, 1024); (256, 2048); (512, 4096); (1024, 8192); (2048, 0)]. Precomputed Visibility Volume - налаштовується для побудови освітлення над геометрією відкидання тіней, де актор видимості зберігається від позиції кожного осередку. Налаштування залежить від трудомістких об'єктів на сцені.

Комбінація цих методів дозволяє гнучко налаштувати кожне VR-додаток для оптимальної продуктивності, в залежності від необхідного результату. Що дозволить знайти баланс між кількістю відображуваних об'єктів в полі зору і кількістю кадрів в секунду.