

## РЕАЛІЗАЦІЯ РЕЙТРЕЙСИНГУ У ВІДЕОКАРТАХ

**Олександр Никифорович Романюк**

**Вікторія Михайлівна Бажан**

**Оксана Володимирівна Романюк**

**Алла Василівна Денисюк**

Вінницький національний технічний університет

### Вступ

Рейтрейсінг (англ. Ray tracing) - це техніка рендеринга, яка використовує принципи реальних фізичних процесів. Для того, щоб побудувати тривимірну модель будь-якого об'єкта та застосувати до неї трасування променів, система відстежує траєкторію віртуального променя до цього об'єкта. При цьому, системі потрібно враховувати поверхню об'єкта та властивості його матеріалу. Ну, і нарешті, світло відстежується за допомогою декількох променів, що імітують відбите світло. Так відбувається трасування променів, яка враховує заломлення, відображення променів, а також коректна взаємодія світла з будь-якими поверхнями, в тому числі і дзеркальними. Після відбиття світла від об'єкта, світло, можливо, змінив свій колір - і це також необхідно врахувати.

На практиці, трасування променів - неймовірно трудомістка з технічного боку, адже частина променів може не відбитися взагалі, частина - відбитися всього кілька раз, а деякі промені в рамках однієї сцени можуть відобразитися нескінченну кількість разів. Повністю точний і коректний рейтрейсінг вимагає дуже високих обчислювальних потужностей апаратної частини, але навіть у цьому випадку це досить тривалий процес.

У вузькому сенсі трасування променів - технологія побудови зображення тривимірних моделей в комп'ютерних програмах, при яких відстежується зворотна траєкторія поширення променя (від екрану до джерела). Алгоритми трасування променів мають такі переваги:

- можливість рендеринга гладких об'єктів без апроксимації їх полігональними поверхнями (наприклад, трикутниками);

- обчислювальна складність методу слабо залежить від складності сцени;
- можливість розпаралелення обчислень - можна паралельно і незалежно трасувати кілька променів, розділяти ділянки (зони екрану) для трасування на різних вузлах кластера тощо;
- відсікання невидимих поверхонь, перспектива та коректне зміни поля зору є логічним наслідком алгоритму.

Розглянемо недоліки.

Серйозним недоліком методу зворотного трасування є низька продуктивність. Метод растеризації і сканування рядків використовує когерентність даних, щоб розподілити обчислення між пікселями. Метод трасування променів кожен раз починає процес визначення кольору пікселя спочатку, розглядаючи кожен промінь спостереження окремо. Втім, цей поділ тягне поява деяких інших переваг, таких як можливість трасувати більше променів, ніж передбачалося, для усунення контурних нерівностей. в певних місцях моделі.

Розробники кінофільмів вже давно використовують цю технологію при виробництві фільмів. Наприклад, у фільмі «Трон» 1982 року випуску. Зазвичай, трасування променів у фільмах додається на стадії монтажу, тому творцям фільмів не потрібно розраховувати поведінку джерел світла в реальному часі. Їм досить це зробити один раз при рендерингу стрічки. Але навіть у такому випадку, прорахунок променів на одному кадрі може займати багато часу. В іграх розробники ніколи не зможуть заздалегідь передбачити, куди рухається гравець і з якого боку буде дивитися на об'єкт, щоб прорахувати відображення та заломлення променів один раз, як це роблять в кіно. Тому в іграх мова йде виключно про трасування променів в реальному часі, а це неймовірно трудомісткий процес. Саме через складність рейтрейсінга, його «прихід» в ігрову індустрію сильно затримався.

NVIDIA першою впровадила апаратну підтримку рейтрейсінга в комп'ютерних іграх (RTX), а прорахунок променів відбувається в режимі реального часу за допомогою спеціальних тензорних RT-ядер. Всі нові карти

ґрунтуються на архітектурі NVIDIA з кодовою назвою Turing, створеної на 12-нм технічний процес. Саме вона дозволяє в реальному часі обробляти відображення світлових променів, допомагаючи зробити графіку в іграх в рази більш реалістичною та кінематографічною.

В іграх трасування променів не передбачає застосування на всю сцену. Тому тут використовується гібридний метод класичних технологій освітлення і безпосередньо трасування. Коли рейтрейсинг активний, то реалізуються високореалістичні ефекти відображень, тіней і освітлення. Графічні налаштування гри (низькі - середні - високі - ультра) регулюють число проходів трасувальника і вибирають, де використовувати звичайні методи, а де DXR. Широко використовується такий прийом, як сегментування зображення. Він визначає, де потрібна участь RT-ядер в обробці кадру.

Кількість вищезазначених RT-ядер в відеокарті визначає продуктивність в іграх при включеному трасуванні.

Нещодавно NVIDIA представила і випустила на ринок нові відеокарти RTX з приставкою «SUPER» в назві. Оновлення отримали три моделі - RTX 2060, RTX 2070 і RTX 2080. Всі refresh-версії відрізняються від «стандартних» більш потужним графічним процесором, а також збільшеною кількістю RT-ядер для обробки трасування променів. Все це допомогло збільшити продуктивність в іграх в середньому на 15%, а в деяких іграх приріст доходить аж до 24%.

Minecraft в версії для Windows 10 отримає підтримку трасування променів. Про це стало відомо зі спільної заяви розробників гри і виробника відеокарт Nvidia.

NVIDIA і розробник Mojang додали в версію гри під Windows 10 варіацію трасування променів, відому як «path tracing». Вона імітує те, як світло поширюється в сцені та являє собою уніфіковану модель розрахунку освітлення для багатьох типів ефектів, які традиційно створювалися окремо, використовуючи растеризацію або гібридні методи рендеринга. Це такі ефекти, як: пряме світло від сонця, неба і різних джерел; реалістичні жорсткі та м'які

тіні залежно від розміру, форми і відстані до джерела світла; попіксельне емісійне освітлення; непряме розсіяне освітлення; непряме бікове освітлення; прозорі матеріали з відображенням, заломленням і розсіюванням (матове скло, вода, лід); атмосферний розсіювання і щільність (об'ємний туман, світлові шахти, реалістичне небо).

Розглянемо різницю трасування променів і пастеризації.

У методі растеризації, загальноприйнятому зараз в сучасній графіці реального часу, для відтворення кожного об'єкта знаходиться проекція на площину екрану від геометричних примітивів (полігонів, найчастіше трикутників), з яких складається об'єкт. У трикутниках розраховуються пікселі з використанням буфера глибини, який містить значення відстані до площини екрану і необхідний для того, щоб ближні до камери трикутники перекривали далекі при рендерингу.

Крім вершин також зберігається інформація про колір, текстурні координати та нормаль, яка необхідна для того, щоб визначити лицьову та зворотну частини кожної поверхні. Колір пікселів визначається в результаті складних обчислень в вершинних і піксельних шейдерах, а такі ефекти, як тіні, промальовується за допомогою додаткових проходів, але також із застосуванням растеризації.

Процес затінення (shading) полягає в розрахунку кількості освітлення для пікселя з урахуванням накладення однієї або декількох текстур на піксель, що і визначає його кінцевий колір. Все це вимагає великої кількості обчислень, адже в сценах сучасних ігор міститься по кілька мільйонів полігонів і по кілька мільйонів пікселів в екранах з високою роздільною здатністю, а оновлення інформації на екрані повинно бути з частотою 60 FPS.

Графічні процесори працюють на дуже високій тактовій частоті та мають велику кількість апаратних блоків, спеціалізованих на певних обчисленнях. Растеризація дуже добре підлягає розпаралеленню. Тому особливих проблем з продуктивністю рендеринга немає, і переважна більшість комп'ютерних 3D-ігор використовує саме растеризацію.

У трасування же променів основна ідея зовсім інша. За допомогою трасування імітується поширення променів світла по 3D-сцені. Трасування променів може виконуватися в двох напрямках: від джерел світла або від кожного пікселя в зворотному напрямку, далі, зазвичай, визначається кілька відбитків від об'єктів сцени в напрямку камери або джерела світла, відповідно. Прорахунок променів для кожного пікселя сцени менш вимогливий обчислювально, а проектування променів від джерел світла дає більш високу якість рендеринга.

Зворотній трасування була вперше описана в 1969 році співробітником компанії IBM в роботі «Some Techniques for Shading Machine Renderings of Solids» і ця техніка прораховує шлях променя світла для кожного пікселя на екрані залежно від 3D-моделей в сцені.

Відома в галузі 3D-графіки по своїм тестовим пакетам компанія Futuremark показала технологічну демонстрацію DXR, зроблену на основі спеціально розробленого гібридного двигуна із застосуванням трасування променів для якісних відображень в реальному часі.

За допомогою апаратно прискореної на GPU трасування променів отримано фізично коректні відображення для всіх об'єктів сцени, включаючи динамічні.

Використання трасування променів дає точні відображення з корекцією перспективи на всіх поверхнях сцени в реальному часі. Добре видно, що трасування ближче до реалізму, ніж більш звичні для нас screen-space відображення, що застосовуються в більшості сучасних ігор.

На сучасних GPU вже можна застосовувати гібридний рендеринг з використанням растеризації для більшої частини роботи і порівняно невеликим вкладом трасування для поліпшення якості тіней, відображень та інших ефектів, з якими складно впоратися за допомогою традиційних технік з растеризуванням.)

Компанія Epic Games спільно з ILMxLAB і Nvidia також показала свій варіант включення можливостей по трасуванні променів в реальному часі в

двигун Unreal Engine 4. Показ відбувся на відкритті GDC 2018, де три зазначені компанії презентували експериментальну кінореалістичну демонстрацію на тематику кіносеріалу «Star Wars» з використанням персонажів з серій «The Force Awakens» і «The Last Jedi».

Демонстраційна програма Epic Games використовує модифіковану версію Unreal Engine 4 і технологію Nvidia RTX, можливості якої розкриваються через DirectX Raytracing API. Для побудови 3D-сцени розробники використовували реальні ресурси з фільмів Star Wars: The Last Jedi з Captain Phasma в блискучих обладунках і двома штурмовиками зі сценою в ліфті корабля First Order.

Підтримка DirectX Raytracing і Nvidia RTX відкриває для Unreal Engine 4 шлях до нового класу технік і алгоритмів, які не були доступні раніше при засилля растеризації. В недалекому майбутньому, розробники ігор зможуть використовувати гібридний підхід з частковим використанням якісної трасування променів для деяких ефектів і високопродуктивного растеризування для більшої частини роботи.

Черговим розробником, хто зацікавився трасуванням променів через DXR, стала студія SEED з Electronic Arts, яка створила спеціальну демо-програму Pica Pica, яка застосовує експериментальний двигун Halcyon, що використовує гібридний рендеринг, як і попередні демонстраційні програми. Також ця демка цікава тим, що в ньому був створений процедурний світ без будь-яких попередніх розрахунків.

Цього року трасування променів в реальному часі підтримують тільки відеокарти Nvidia. AMD заявила, що впритул займеться рейтресингом у майбутньому.

Нижче наведено список сучасних відеокарт, апаратно підтримують рейтрейсінг. Nvidia GeForce RTX 2060 - 30 RT ядер, 6GB GDDR6, Nvidia GeForce RTX 2060 Super - 34 RT ядра, 8GB GDDR6, Nvidia GeForce RTX 2070 - 36 RT ядер, 8 GB GDDR6, Nvidia GeForce RTX 2070 Super - 40 RT ядер, 8GB GDDR6, Nvidia GeForce RTX 2080 - 46 RT ядер, 8GB GDDR6

Nvidia GeForce RTX 2080 Super - 48 RT ядер, 8GB GDDR6, Nvidia

GeForce RTX 2080 Ti - 68 RT ядер, 11GB GDDR6, Nvidia Titan RTX - 72 RT.

### Висновки

На даному етапі розвитку комп'ютерної графіки широко використовують технологію рейтресингу, яка дозволяє досягти фотореалістичної якості формування тривимірних графічних зображень. Фірма NVIDIA використовує рейтресинг в останніх відеокартах.

### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Рейтресінг [Електронний ресурс] – Режим доступу до ресурсу: <https://dtf.ru/gameindustry/37057-zachem-nuzhen-voobshche-etot-vash-reytreysing-shmeytreysing>.

2. Что такое рейтресинг и как работает трассировка лучей [Електронний ресурс] – Режим доступу до ресурсу: [https://arsplus.ru/wk/chto\\_takoe\\_reytreysing\\_i\\_kak\\_rabotaet\\_trassirovka\\_luchey\\_v\\_i\\_grakh/](https://arsplus.ru/wk/chto_takoe_reytreysing_i_kak_rabotaet_trassirovka_luchey_v_i_grakh/).

3. Трассировка лучей: что, как и почему [Електронний ресурс] – Режим доступу до ресурсу: <https://cubiq.ru/trassirovka-luchey/>.

4. Трассировка лучей [Електронний ресурс] – Режим доступу до ресурсу: [https://ru.wikipedia.org/wiki/%D0%A2%D1%80%D0%B0%D1%81%D1%81%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0\\_%D0%BB%D1%83%D1%87%D0%B5%D0%B9](https://ru.wikipedia.org/wiki/%D0%A2%D1%80%D0%B0%D1%81%D1%81%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0_%D0%BB%D1%83%D1%87%D0%B5%D0%B9).