Д. Ю. Дешкович И. В. Белый

# ТРАССИРОВКА ЛУЧЕЙ В ТРЁХМЕРНОЙ КОМПЬЮТЕРНОЙ ГРАФИКЕ

*Рассматривается набор уравнений для построения алгоритма трассировки лучей, использующийся в трёхмерной компьютерной графике.*

## Введение

Для реализации трассировки лучей будут заданы следующие условия. Точка обзора фиксирована. Точка обзора — это место, в котором располагается глаз в нашей аналогии. камера расположена в начале системы координат, то есть O = (0, 0, 0). Ориентация камеры фиксирована камера смотрит вниз по положительной оси Z,положительная ось Y направлена вверх, а положительная ось X — вправо.

## I. Трассирование лучей

Предполагается реализация обратной трассировки лучей, где исследование начинается с луча, испускаемого из камеры и направленного через точку в окне просмотра, после чего продвигается вперед до столкновения с объектом в трехмерной сцене. Точка столкновения определяет объект, видимый из камеры через данную точку окна просмотра, так что начальное приближение к цвету в этой точке представляет собой цвет этого объекта, воспринимаемый как "цвет света, проходящего через данную точку". Таким образом, задачей является определение соответствующих уравнений для этого процесса.

## II. Уравнение лучей

Для текущей задачи используется параметрическое уравнение. Луч проходит через O, и его направление из ( O в V ),из этого следует, что любая точка P выражается как $P = O + t(V - O)$ , где t - произвольное действительное число. 0бозначим $(V - O)$, то есть направление луча, как $\vec{D}$, тогда уравнение примет вид: $P = O + t\vec{D}$ (1)

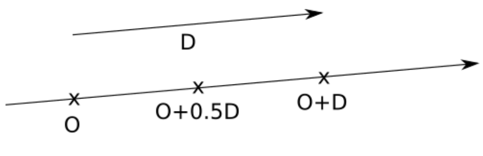


Рис. 1 – Визуализация параметрического уравнения

## III. Уравнение сферы

Для реализации следующего пункта необходимо добавить простейший примитив сферы. По определению сферы являются полыми. Если C - центр сферы, а r радиус сферы, то точки P на поверхности сферы удовлетворяют следующему уравнению: $distance(P, C) = r$ .Расстояние между P и C- это длина вектора из P в C. Длина вектора — это квадратный корень его скалярного произведения на себя. В результате преобразований получаем: $\langle P - C, P - C\rangle = r^2$ (2)

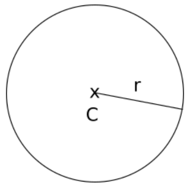


Рис. 2 – Сфера C с радиусом r

## IV. Пересечение луча и сферы

Имея два уравнения, уравнение 2 и уравнение 1 , одно из которых описывает точки сферы, а другое — точки луча. Точка P, в которой луч падает на сферу, является одновременно и точкой луча, и точкой на поверхности сферы, поэтому она должна удовлетворять обоим уравнениям одновременно.

Единственная переменная в этих уравнениях — это параметр t, потому что O, $\vec{D}$, C и r заданы, а P — это точка, которую необходимо найти. Поскольку P — это одна и та же точка в обоих уравнениях P заменяется в первом на выражение для P во втором.

Подставив получится: $\langle O + t\vec{D} - C, O + t\vec{D} - C\rangle = r^2$. Разложив скалярное произведение на его компоненты, воспользовавшись его дистрибутивностью, переместив параметр t из скалярных произведений и упростив его получим следующее:

$$t_1, t_2 = \frac{-k_2 \pm \sqrt{{k_2}^2 - 4k_1k_3}}{2k_1} \quad (3)$$

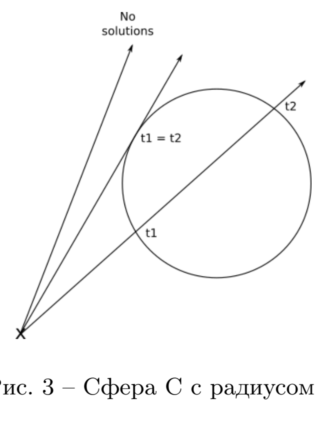


Рис. 3 – Сфера C с радиусом r

## V. Рендеринг сферы

Для каждого пикселя на холсте теперь возможно вычислить соответствующую точку в окне просмотра. Зная положение камеры, выражается уравнение луча, который исходит из камеры и проходит через заданную точку окна просмотра.

Имея сферу, вычисляется точка, в которой луч пересекает эту сферу. То есть достаточно только вычислить пересечения луча и каждой сферы, сохранить ближайшие к камере точки и закрасить пиксель на холсте соответствующим цветом. Однако стоит уделить особое внимание параметру t.

Вернувшись к уравнения луча: $P = O + t(V - O)$ , поскольку исходная точка и направление луча постоянны, меняя t во множестве действительных чисел, мы получим каждую точку P на этом луче. При $t = 0$ мы получим $P = O$, а при $t = 1$ мы получим $P = V$. При отрицательных числах мы получим точки в противоположном направлении, то есть за камерой.

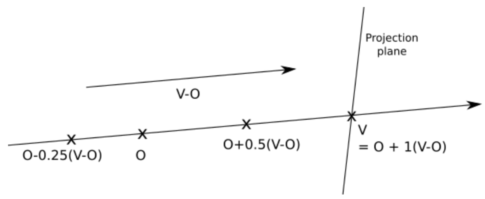


Рис. 4 – Плоскость отражения

Получается три области параметров: $t < 0$ (за камерой), $0 \leq t \leq 1$ (между камерой и плоскостью), $t > 1$ (сцена).

## VI. Выводы

Таким образом, были рассмотрены основные уравнения реализующие алгоритмы трассировки лучей. Хотелось бы отметить, что трассировка лучей представляет собой неотъемлемый инструмент в области компьютерной графики, обеспечивающий высокую степень реализма в визуальных эффектах. Хотя использование трассировки лучей может замедлить процесс рендеринга, его оптимизация для работы в специализированных редакторах обеспечивает эффективность и удобство в использовании.

1. Habr [Электронный ресурс]/ Режим доступа: https://habr.com/ru/articles/342510/ / Дата доступа: 17.03.2024
2. Adobe [Электронный ресурс]/ Режим доступа: https://www.adobe.com/products/substance3d/discover/what-is-ray-tracing.html / Дата доступа: 17.03.2024
3. AutoDesk [Электронный ресурс]/ Режим доступа: https://www.autodesk.com/solutions/ray-tracing / Дата доступа: 17.03.2024

*Дешкович Даниил Юрьевич*, студент 3 курса факультета информационных технологий и управления, daniil.deshkovich@gmail.com.

*Белый Илья Владимирович*, студент 3 курса факультета информационных технологий и управления, ilya.belyj.2003@mail.ru.

*Научный руководитель: Коршикова Дарья Валерьевна*, ассистент кафедры вычислительных методов и программирования БГУИР