Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ «САРАТОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра информатики и программирования

**Обзор алгоритмов глобального освещения**

Реферат

студента 3 курса 341 группы

направления 02.03.03 Математическое обеспечение и администрирование информационных систем (профиль Параллельное программирование) факультета компьютерных наук и информационных технологий

Акимов Артемия Андреевича

Проверено: Батраева И.А. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Саратов 2016

# Содержание

[Содержание 2](#_Toc450590647)

[1 Введение 3](#_Toc450590648)

[2 Глобальное освещение как явление 4](#_Toc450590649)

[3 Алгоритм излучательности (radiosity) 5](#_Toc450590650)

[4 Алгоритмы на базе трассировки лучей, стохастические алгоритмы (метод Монте-Карло). 6](#_Toc450590651)

[4.1 Алгоритм Ray-casting 6](#_Toc450590652)

[4.2 Алгоритмы Visibility Ray-tracing 7](#_Toc450590653)

[4.3 Распределенная трассировка лучей(distributed ray-tracing) 7](#_Toc450590654)

[4.4 Алгоритм трассировки путей 8](#_Toc450590655)

[5 Метод фотонных карт 10](#_Toc450590656)

[6 Гибридные алгоритмы 13](#_Toc450590657)

[7 Двунаправленная трассировка лучей (bidirectional ray-tracing ) 13](#_Toc450590658)

[8 Заключение 15](#_Toc450590659)

[9 Список использованных источников 16](#_Toc450590660)

# Введение

По ходу становления компьютерной графики самой злободневной задачей являлось создание таких изображений, которые при помощи геометрического положения сцены, информации о материалах, позиции и множества различных факторов отличаются от настоящей, действительной фотографии лишь на малую часть.

В основе алгоритмов получения реалистичного изображения лежит множество физических принципов. Если мы построим хорошо спроектированную модель света, удовлетворяющую законам физики, а часто и полному их выполнению, можно добиться неотличимого изображения от фотографии. Тем самым, за хорошим качеством и множеством зрительных эффектов стоит высокая вычислительная сложность таких алгоритмов получения изображений.

Но оценка похожести синтезируемого изображения с реальной фотографией проводится субъективно (так как она производится человеком). Можно построить несложную модель, которая не будет учитывать множество различных эффектов. Тогда качественный с точки зрения скорости алгоритм создания изображений будет синтезировать изображение, которое не будет отличаться от фотографии с точки зрения пользователя. Вся история фотореалистичного создания изображений – это поиск золотой середины между затратами на вычисление и приближением картинки к действительности.

Преступим к рассмотрению алгоритмов и теоретической составляющей.

# Глобальное освещение как явление

Global illumination(GL) это определение целого пласта алгоритмов объёмной компьютерной графики, которые используются для построения более качественной и реалистичной сцены. В них учитывается, как и прямые световые лучи, поступающие от источника, так и отраженный свет в сцене, часть которого поглотилась, часть попала на другие объекты, но все же осталась на сцене. Это называется непрямым освещением.

В теории - тени, отражения, преломления света и есть части глобального освещения, потому что при моделировании их, один объект влияет на рендеринг других объектов на сцене (в отличие от объекта под воздействием только прямого света). На практике лишь создание модели совмещенного диффузного отражения называется глобальным освещением.

Отображенные с использованием глобальных алгоритмов освещения часто являются более реалистичными изображениями, чем изображения только с использованием алгоритмов прямого освещения. Тем не менее, такие изображения являются вычислительно более дорогими, и, следовательно, гораздо медленнее для генерации. Один общий подход заключается в вычислении глобальной освещенности сцены и хранении информации о геометрии, например, излучательности. Далее сохранённые данные могут быть использованы для создания изображений с разных точек зрения для генерации сцены без необходимости проходить через дорогостоящие расчеты освещения неоднократно.

# Алгоритм излучательности (radiosity)

Для начала дадим определение фактора представления или форм-фактора – коэффициент того, как участки соотносятся друг с другом. Этот коэффициент будет рассчитан для каждой пары участков разбитой сцены, которую мы разделили заранее на маленькие поверхности. Чем ближе и чем меньше углы между участками, тем меньше их форм-фактор. Если же на пути будет встречены другие участки, то форм-фактор будет уменьшен. Также он может быть равен нулю, если между участками находится преграда, полностью преграждающая путь света.

В качестве коэффициентов СЛАУ используются как раз именно эти форм-факторы. Решение этой системы уравнений приводит к нахождению яркости каждого отдельного участка сцены, учитывая тени, преграды и поглощения света.

В более сложной версии radiosity решает систему итеративно с промежуточными значениями излучательности для обновленной версии, в соответствии с новым шагом. То есть, после каждой итерации, мы знаем, как сцена выглядит после одного светового отражения, после двух итераций - двух отражений, и так далее. Это полезно для получения интерактивного просмотра сцены. Ко всему прочему, при просмотре можно остановить шаги после того, как изображение приближено к действительной фотографии с хорошей точностью, а не ждать вычисления до конца, пока алгоритм численно сходятся.

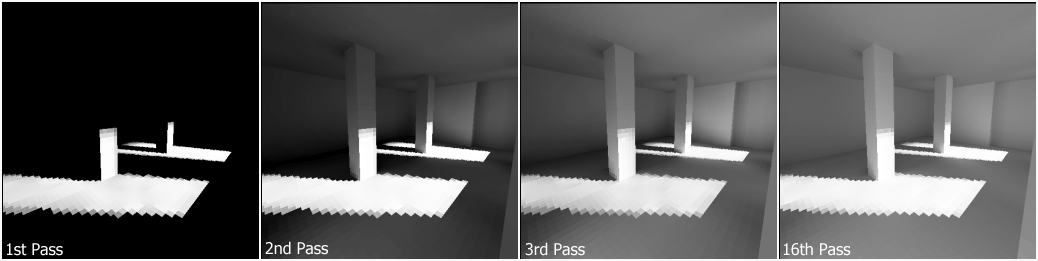


Рисунок 1. Radiosity [1]

# Алгоритмы на базе трассировки лучей, стохастические алгоритмы (метод Монте-Карло).

Для построения фотореалистического изображения основывается на вычислении интеграла освещенности. Для решения разобьем пространство равномерной сеткой. В зависимости от числа узлов, которое зависит от размерности пространства напрямую, мы будем получать более сложный в плане вычислительной стоимости алгоритм. Такое решение приведет нас к квадратурному решению интеграла.

Но более простым и интуитивно понятным является метод Монте-Карло. Сэмплы выбираются в произвольном порядке, их количество не зависит от размерности пространства, по которому вычисляется интеграл. Чем больше узлов – тем меньше шумов, что приведет к большей точности изображения.

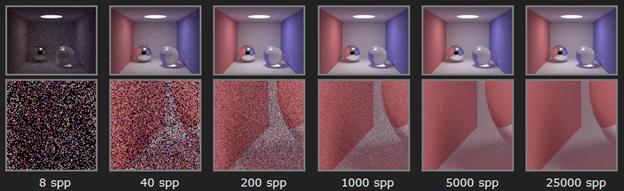


Рисунок 2. Применение метода Монте-Карло при

различном количестве узлов на пиксель [2]

Алгоритмы на базе метода Монте-Карло часто применяются из-за свой линейной зависимости их вычислительной сложности и количества узловых точек на сетке. Не стоит также забывать про простоту реализации таких алгоритмов. Проблемы с зашумлением изображения обычно решают при помощи специального размытия изображения. [3]

## Алгоритм Ray-casting

Этот алгоритм совмещает в себе как расчет теней, так и первичное освещение от источника. Идея алгоритма проста – лучи света, исходящие из камеры достигают конечных объектов. Точки пересечения объектов и лучей запоминаются. Потом из каждой такой точки пускается луч света к источнику. Произведем подсчет энергии достигших источника лучей. Но если тот луч достигает какого-то другого объекта на пути к источнику, то это означает, что точка пересечения объекта и луча, идущего к источнику, находится в тени.

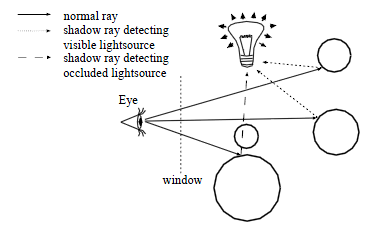


Рисунок 3. Ray-casting [4]

## Алгоритмы VisibilityRay-tracing

Visibility Ray-tracing-программа математически определяет и воспроизводит путь, по которому каждый световой луч следует в обратном направлении от глаз обратно в исходную точку. Путь каждого луча света состоит из нескольких компонентов, прямолинейных и почти всегда включает в себя отражение, преломление, или теневые эффекты от объектов в пределах сцены. Стоит сказать, что все изменения пути света проходят по идеальным законам отражения и преломления, а материалы имеют постоянные коэффициенты преломления, вне зависимости от углов.

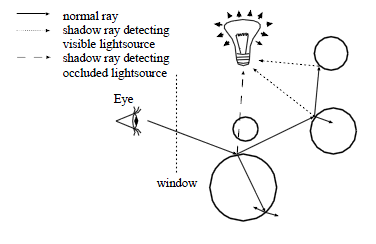


Рисунок 4.Visibility Ray-tracing [4]

## Распределенная трассировка лучей(distributed ray-tracing)

Distributed ray tracing – более новый подход к синтезу изображений. Основная идея заключается в том, что нам не требуются дополнительные лучи для обработки теней в пространстве. Например, вместо того, чтобы вычислять множество раз в каждом пространственном положении значения форм-факторов, мы лишь один раз сгенерируем множество отраженных лучей. Таким образом, мы построим не один путь, а множество путей, которые будут независимы друг от друга. Это и видно на рисунке ниже.

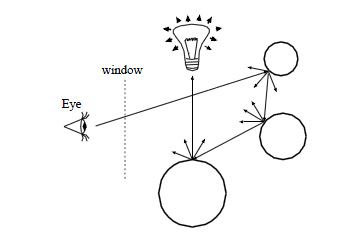


Рисунок 5. Распределенная трассировка лучей [4]

## Алгоритм трассировки путей

Трассировка пути использует случайную выборку для постепенного вычисления итогового изображения. Процесс случайной выборки позволяет сделать некоторые сложные явления, которые не обрабатываются в отслеживании ray tracing, но это обычно занимает больше времени, чтобы привести изображение к высокому качеству.

Случайная выборка является причиной появления шумов, но шумы удаляются. Алгоритм позволяет генерировать больше выборок, то есть варьировать значение цвета в результате прохода одного луча.

Случайная выборка в пути трассировки лучей распределены случайным образом в пределах каждого пикселя в пространстве камеры, и на каждом пересечении с объектом в сцене новое отражение луча, проходя в случайном направлении, генерируется. После некоторого количества отскоков каждый луч, в конце концов, выходит из сцены или поглощается. Когда луч закончил проход по сцене значение сэмпла вычисляется на основе объектов, от которых луч отскочил. Значение сэмпла добавляется к среднему значению для исходного пикселя.

Сэмплы в путях в изображении распределены равномерно по всем пикселям. Цвет каждого пиксела является средним значением всех значений выборок, вычисленных для этого пикселя.

Шум уменьшается с течением времени все больше и больше при вычислении новых сэмплов.

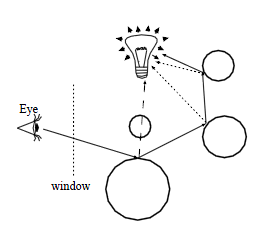


Рисунок 6. Трассировка путей [4]

# Метод фотонных карт

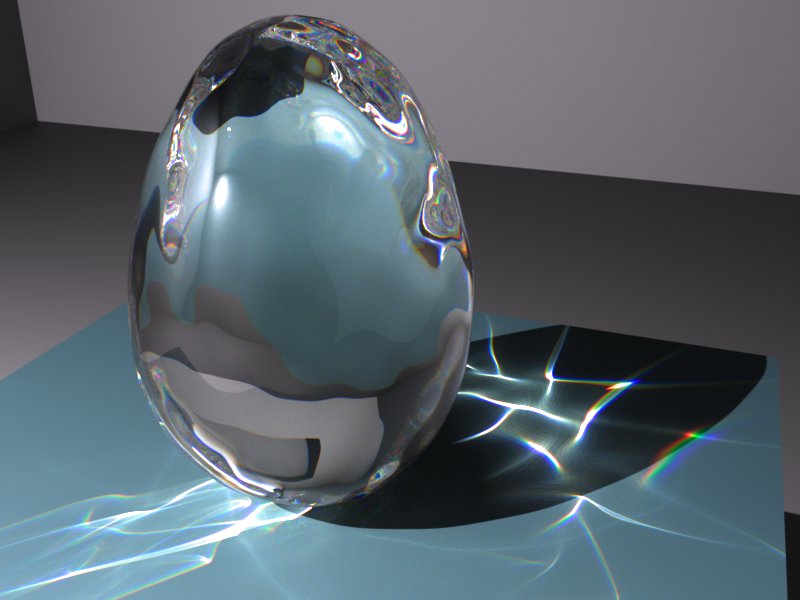


Рисунок 7. Каустик

Этот метод занимает гораздо меньше ресурсов, чем традиционные методы Монте-Карло трассировки лучей. Методы Монте-Карло вычислительно сложно отразить в изображении сложные эффекты, такие как каустик. Метод фотонных карт способен быстро оценить глобальное освещение, создаваемое источником. Эти лучи фактически и есть фотоны, которые возвращаются на протяжении сцен. Photon mapping позволяет легко визуализировать размытия, а также каустик.

Общий метод включает в себя три этапа: излучения фотонов, сбора фотонов и рендеринга. Они разделены на два прохода, один проход генерирует карту фотонов, а следующий проход делает сцену с помощью фотонной карты.

Эмиссия фотонов испускает фотоны от источника света в различных направлениях. В зависимости от типа источника света, фотоны могут перемещаться по-разному. Точечные источники света испускают фотоны в любом общем направлении. Источники в виде плоскостей испускают фотоны из данной области в случайном направлении, которое меньше 90 градусов от нормали.

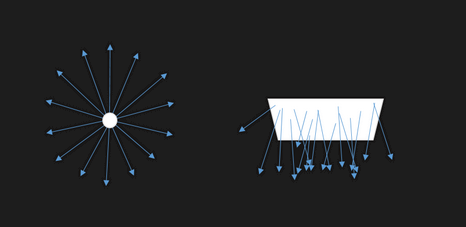


Рисунок 8. Точечный источник и источник в виде области

Когда фотоны достигли объекта, они могут либо отражаться зеркально, либо отражаться диффузно или поглощаться. Для определения направления используем так называемый метод Русской рулетки. В основном, это делается путем выбора случайного числа в диапазоне от 0 до 1. От числа зависит – будет ли фотон зеркально отражен или диффузно отражен или поглощен полностью. После того, как все фотоны излучаются, они сохраняются в фотонной карте.

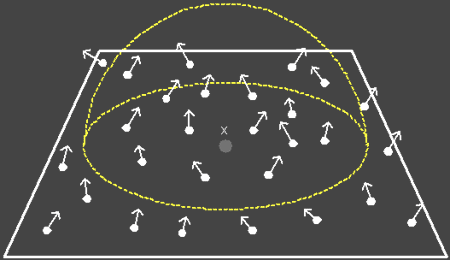


Рисунок 9. Полусфера представления сцены [5]

На этапе рендеринга, фотоны должны быть собраны на основе их расположения. Чтобы повысить эффективность, строят КD-дерево для карты фотонов, так что фотоны сортируются в зависимости от их положения в кадре. При визуализации, трассировка лучей необходима для того, чтобы найти место соприкосновения. В этот момент, N близкие фотоны находятся в данной области, а их значения суммируются. Затем эти значения делятся на площадь поверхности сферы. Этот алгоритм по мере работы производит сбор фотонов: в том месте, где фотонов больше, радиус сбора наиболее маленький, и наоборот. При помощи этого уменьшается зашумление получаемого изображения. [5]

Отображение фотона используется, чтобы вычислить эффекты косвенного освещения и каустика. Слишком большое число фотонов потребовалось бы в карте фотона, чтобы точно взаимодействовать с зеркальными/глянцевыми поверхностями и прямым освещением.

Несомненным плюсом этого алгоритма является то, что его очень просто выполнить на распределенных системах, то есть параллельно.

# Гибридные алгоритмы

При появлении множества алгоритмов встал вопрос о комбинировании алгоритмов трассировки лучей и алгоритма излучаемости для того, чтобы совместить плюсы обоих направлений в глобальном освещении – быстрота от алгоритмов излучаемости и возможность применения ко всему спектру 3D сцен от трассировки лучей.

## Двунаправленная трассировка лучей (bidirectional ray-tracing )

Хотя стандартная трассировка Монте-Карло и прямое освещение вместе обеспечивают адекватную методику отбора проб для большинства ситуаций, они все еще могут быть не предназначенными для определенных типов сцен. Например, предположим, что у нас есть сцена, где свет источника света попадает на всю площадь зеркала и зеркало отражает его на стену. В этом случае наиболее подходящий метод отбора проб должен был бы начать съемки лучей от источника света, обойдя сцену один раз, и соединить этот сегмент пути в маршрут, выходящий из камеры. Подобные случаи могут быть построены для почти всех видов техники выборки.

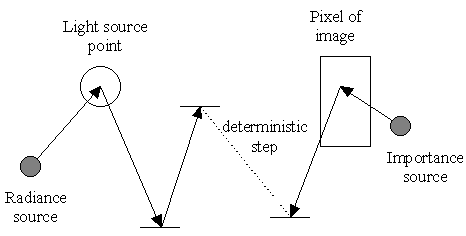
Более общий алгоритм генератора световой путь был разработан независимо друг от друга Вич и Гуйбис и Лафорюн и Виллемс. Алгоритм генерирует световой дорожки с обоих концов. Один сегмент строится, начиная от камеры, а другой сегмент, начиная с произвольной точки на поверхности источника света. И, наконец, два сегмента связаны с детерминированным шагом. 

Рисунок 10. Гибридный алгоритм

Из камеры: первый шаг заключается в создании точки на диафрагме объектива. Сейчас это единственная детерминированная точка, но мы оставим этот шаг, потому что дальнейшие улучшения применим использование этого, например, размытость может быть введена, если точки камеры генерируются на отрезок линии. Со стороны источника света шаг заключается в том, чтобы создать точки на светоизлучающий поверхности. Такой подход допустим для источников света в виде плоскости. (Точечные источники света могут быть легко введены.) Число шагов камеры и источника света предопределены. [6]

# Заключение

В заключении хотелось бы сказать о том, что на создание каждого алгоритма ушло множество лет. И самое удивительное, что алгоритмы, созданные для устаревших вычислительных машин, до сих пор используются во множестве современных игр, графических приложений. А какой из алгоритмов больше всех подходит для решения конкретной задачи зависит от необходимых скоростей или качества изображения.

# Список использованных источников

1. Radiosity algorithm [Электронный ресурс] : [Сайт ]. URL : Режим доступа -[http://freespace.virgin.net/hugo.elias/radiosity/radiosity.htm](http://freespace.virgin.net/hugo.elias/radiosity/radiosity.htm%20), свободный (дата обращения: 8.05.16)
2. Beason K. Smallpt. Global Illumination in 99 lines of C++.
3. История развития алгоритмов глобального освещения [Электронный ресурс] : [Сайт]. URL : Режим доступа - <http://cgm.computergraphics.ru/issues/issue19/globalillum#lit48>, свободный (дата обращения: 8.05.16)
4. Szirmay-Kalos L. Photorealistic image synthesis using ray-bundles.
5. Shulz D. Photon Mapping [Электронный ресурс] : [Сайт ]. URL : Режим доступа -http://users.csc.calpoly.edu/~zwood/teaching/csc572/final15/dschulz/index.html, свободный (дата обращения: 8.05.16)
6. Péter D. Implementation of bidirectional ray tracing algorithm [Электронный ресурс] : [Сайт ]. URL : Режим доступа - http://www.cescg.org/CESCG98/PDornbach, свободный (дата обращения: 8.05.16)