***ВКР***

***<Титулы>***

Аннотация.

В представленной работе рассмотрены некоторые популярные рендеры, реализующие различные алгоритмы трассировки лучей. Эти алгоритмы делают возможным получение фотореалистичных изображений с помощью моделирования различных световых свойств, таких как глобальное освещение, тени, каустики, дисперсия, глубина резкости, подповерхностное рассеивание и т.д. Все алгоритмы объединяет идея симулирования пути света, но отличают реализации способов это сделать. В зависимости от этого алгоритм может по-разному изображать различные световые эффекты, и объём вычислений будет различным. Благодаря высокой вычислительной мощности современных процессоров и видеокарт становится актуальным рассмотрение достаточно сложных алгоритмов и программ, реализующих их. В представленной работе сравниваются различные алгоритмы и их реализации. Сделаны выводы о качестве получаемого изображения и реализованной многопоточности вычислений.

Оглавление

Аннотация

Введение

Литературный обзор

Глава 1 Физические модели и законы.

Глава 2 Уравнение рендеринга и PBR.

Глава 3 Рендеры и алгоритмы трассировки.

* appleseed
* Cycles
* LuxCore

Глава 4 Способ сравнения и метрика PSNR

Глава 5 Эксперименты

* 1
* 2
* 3
* 4
* Световод 5
* Бассейн 6

Глава 5 Построение графиков, сравнение, выводы

Глава 6 Заключение

Список сокращений

|  |  |
| --- | --- |
| Сокращение | Расшифровка |
| PBR | Physically-based rendering (рус. Физически-корректный рендеринг) |
| BSDF | Bidirectional scattering distribution function (рус. Двулучевая функция распределения рассеивания) |
| BRDF | Bidirectional reflectance distribution function ( рус. двулучевая функция отражательной способности) |
| BTDF | Bidirectional transmittance distribution function (рус. Двулучевая функция пропускной способности) |
| PSNR | peak signal-to-noise ratio |

Введение

Обоснование выбора темы диплома и актуальность

На сегодняшний день на рынке существует много программ трехмерной визуализации, который используют для реалистичного синтеза различные алгоритмы трассировки лучей. Эти программы применяются в кино и теле индустрии, дизайне, архитектуре и т.п. На сегодняшний день наибольшей популярностью пользуются платные пакеты, такие как Maxwell Render, Lumion, Solidworks Visualize, Marmoset Toolbag, Corona Render, Iray, V-Ray, Indigo Render. Для некоторых молодых компаний такой расход может быть критическим, поэтому они склонны искать альтернативные бесплатные аналоги. К счастью, такие существует, но тогда встает вопрос об качестве их работы. Ведь помимо цены, важными остаются такие характеристики, как время рендеринга и фотореалистичность получающегося изображения. В данной работе сравниваются эти характеристики у некоторых популярных бесплатных рендеров.

Также важным фактором для времени работы играет реализация распараллеливания алгоритмов рендеринга. Ведь они, благодаря идее имитации пути нескольких световых лучей, легко поддерживают эту возможность. С вычислительной мощности процессоров актуально рассмотреть и реализацию многопоточности у этих алгоритмов.

Формулировка цели и задач исследования

Цель исследования: сравнение реализованных алгоритмов трассировки лучей, использующихся в различных бесплатных программах для трехмерной визуализации.

Задачи:

1 построение трехмерных сцен

2 настройка идентичных параметров для каждой сцены

3 получение эталонной сцены, с которой будет проводится сравнение

4 запуск рендеров с ограничением по времени и/или потокам

5 получение графиков скорости и качества с помощью метрики PSNR (peak signal-to-noise ratio).

Понятия объекта и предмета исследования

Объектом исследования в данной работе является компьютерная графика, а предметом – алгоритмы трассировки лучей и их реализация.

Литературный обзор.

Глава 1. Физические модели и законы

Перед тем, как говорить о реалистичном отображении внешнего мира определимся с понятием «реалистичность». Все что видит человеческий глаз – это отраженный от разных поверхностей свет с разными длинами волн, лежащими в диапазоне от 380 нанометров до 780 нанометров. Поэтому для реалистичного рендеринга алгоритмам необходимо руководствоваться законами физической оптики, т.е. корпускулярно-волновыми свойствами света, а также законами взаимодействия света с различными поверхностями.

Законы геометрической оптики.

Геометрическая оптика описывает распространение электромагнитного поля как распространение лучей в пространстве. Луч – это линия (кривая или прямая), вдоль которой распространяется энергия светового поля.

* Закон прямолинейного распространения света (принцип Ферма)
* Оптическая длина луча (произведение геометрического пути светогого луча и показателя преломления среды, в которой распространяется свет) между двумя точками минимальна по сравнению со всеми другими линиями, соединяющими эти две точки. Отсюда очевидно следует, что в однородной среде световой луч движется по прямой линии.
* Закон независимого распространения лучей
* Если через точку пространства проходит несколько лучей, то каждый луч никак не влияет на любой другой луч. То есть он ведет себя так, будто бы никаких других лучей нет. Однако этот закон перестает выполнятся при высокой интенсивности света, например, такой как в лазерах.
* Закон преломления света (Закон Снеллиуса)
* При переходе светового луча из одной среды в другую, на границе двух сред световой луч разделяется на два: отраженный и преломленный. Угол падения света на поверхность связан с углом преломления соотношением ,

где показатель преломления первой среды,

показатель второй среды, в которую свет попадает, пройдя границу раздела,

угол между падающим лучом и нормалью к поверхности в первой среде,

угол между лучом и нормалью к поверхности во второй среде.

* Законы отражения света
* **Диффузное отражение**. При отражении света от шероховатой поверхности лучи отражаются в разные стороны.
* **Зеркальное отражение**. Падающий и отраженный лучи, а также нормаль к отражающей поверхности в точке падения лежат в одной плоскости и угол падения равен углу отражения.
* **Полное внутреннее отражение**. Ситуация, при которой интенсивность преломленного луча становится нулевой. Возникает на границе двух сред, где волна падает из среды с большим показателем преломления (т.е. более плотной).
* Закон обратимости светового луча.
* Траектория пути лучей не зависят от направления распространения. Другими словами, если луч, выпущенный из точки А в точку Б, пустить в обратном направлении от Б к А, то он будет иметь точно такую же траекторию.

. Также о

Электромагнитные свойства света.

Свет (или световое поле) – это электромагнитные волны в оптическом диапазоне длин волн (0 – 40 мкм). В этом диапазоне выполняются законы геометрической оптики и свет слабо взаимодействует с веществом. Оптический диапазон состоит из 4 видов излучения: рентгеновское, ультрафиолетовое, инфракрасное и видимое. В данной работе имеет значение только последнее.

Видимое излучение – отраженный от разных поверхностей свет видимого диапазона, лежащий в пределах от 380 нм до 780 нм.

Дифракция

Под дифракцией света понимают всякое уклонение от прямолинейного распространения света, если оно не может быть истолковано как результат отражения, преломления или изгибания световых лучей в средах с непрерывно меняющимся показателем преломления. Если в среде имеются мельчайшие частицы постороннего вещества (туман) или показатель преломления заметно меняется на расстояниях порядка длины волны, то в этих случаях говорят о рассеянии света.

Интерференция и когерентность

Колебания (и волны) называются когерентными, если разность их фаз постоянна во времени. Если оба колебания не согласованы друг с другом т.е. разность их фаз как-то изменяется во времени, то такие колебания называют некогерентными.

При суперпозиции когерентных волн происходит перераспределение интенсивности I в пространстве: в одних местах возникают максимумы, в других минимумы интенсивности. Такое явление называется интерференцией волн.

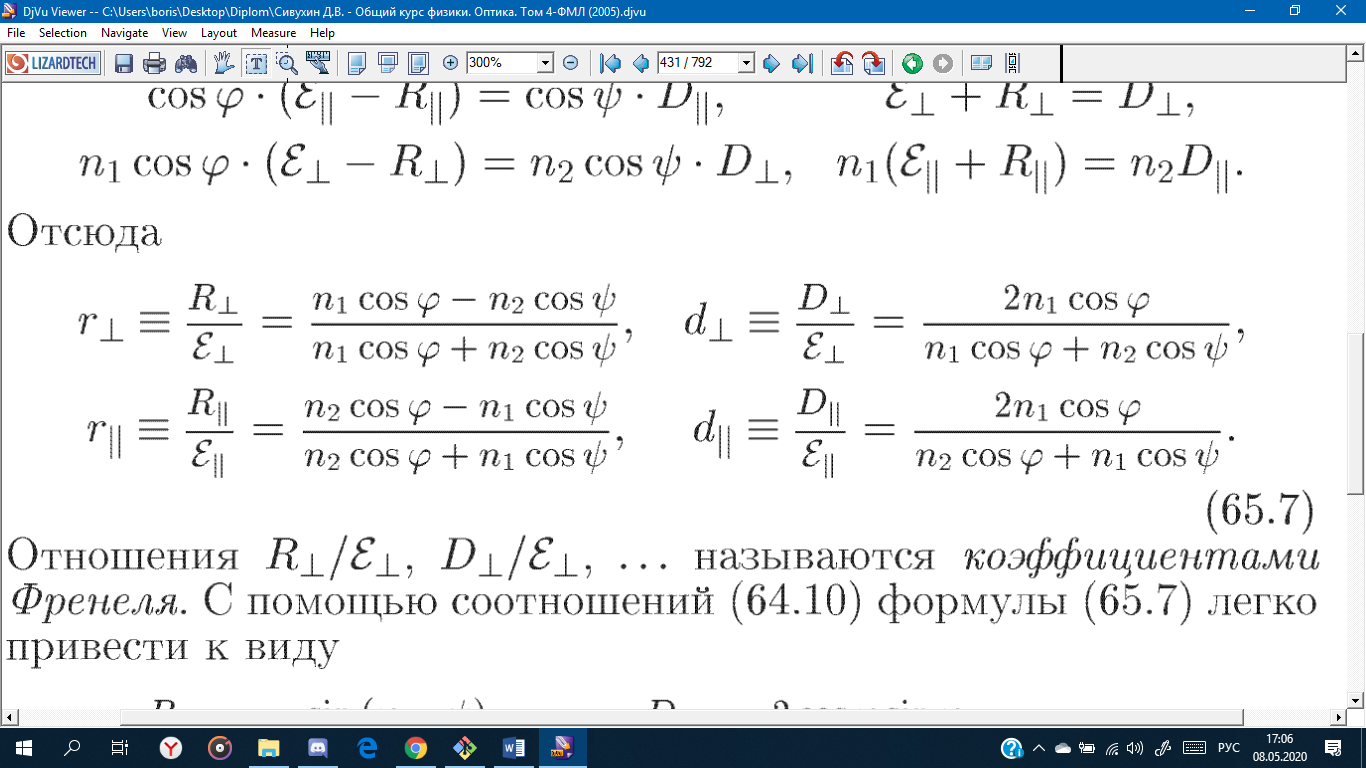
Принцип Гюйгенса-Френеля

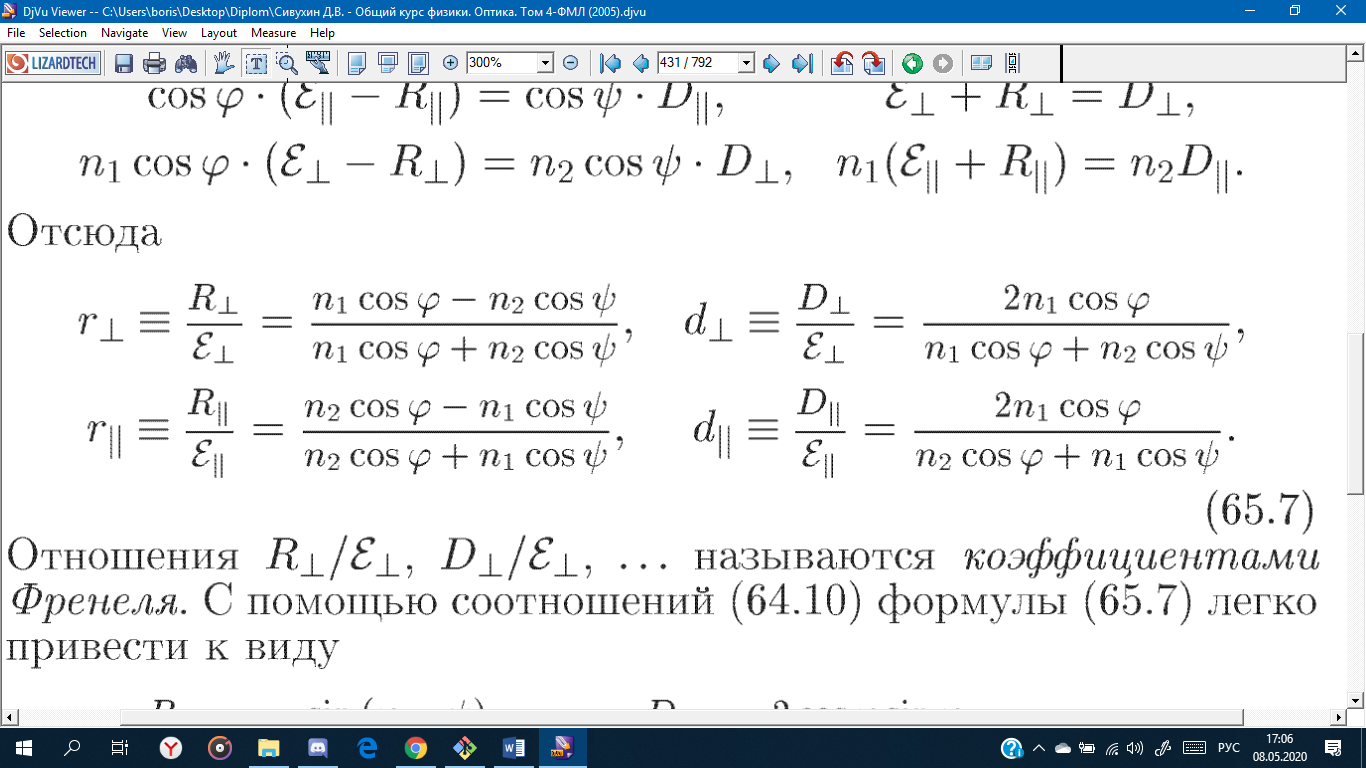
*Окружим все источники света S1, S2, S3, ... произвольной замкнутой поверхностью F. Каждую точку такой поверхности можно рассматривать как источник вторичных волн, распространяющихся во всех направлениях. Эти волны когерентны, поскольку все они возбуждаются одними и теми же первичными источниками. Световое поле, возникающее в результате их интерференции, в пространстве вне поверхности F совпадает с полем реальных источников света.* Таким образом, действительные источники света можно как бы заменить окружающей их светящейся поверхностью F с непрерывно распределенными по ней когерентными вторичными источниками. Отличие этой поверхности от реальной поверхности излучающего тела состоит в том, что она абсолютно прозрачна для всякого излучения. В такой формулировке принцип Гюйгенса-Френеля выражает весьма общее положение. Он означает, что волна, отделившаяся от своих источников, в дальнейшем ведет автономное существование, совершенно не зависящее от наличия источников.

Каустики

Каустики — это особые линии (в двумерном случае) и особые поверхности, вблизи которых резко возрастает интенсивность светового поля. Нужно ли тут подробно c математикой?

Уравнения Френеля

Уравнения Френеля описывают зависимость амплитуды электрического (магнитного) поля падающей волны и полей отраженной и преломленной волн: 



Здесь составляющая отраженной волны, лежащая перпендикулярно плоскости падения

составляющая преломленной волны, лежащая перпендикулярно плоскости падения

составляющей отраженной волны, лежащая в плоскости падения

составляющей преломленной волны, лежащая в плоскости падения

амплитуда составляющей отраженной волны, лежащая перпендикулярно плоскости падения

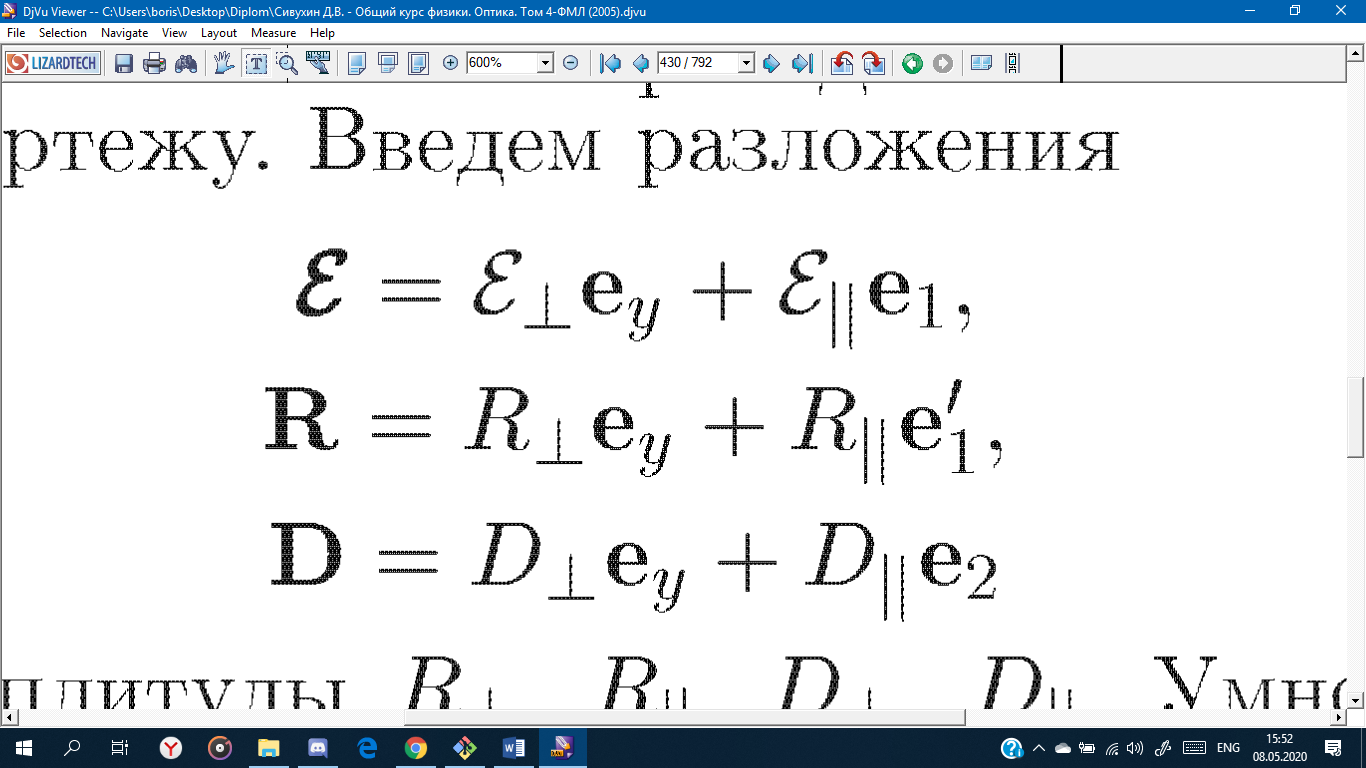
амплитуда составляющей отраженной волны, лежащая в плоскости падения

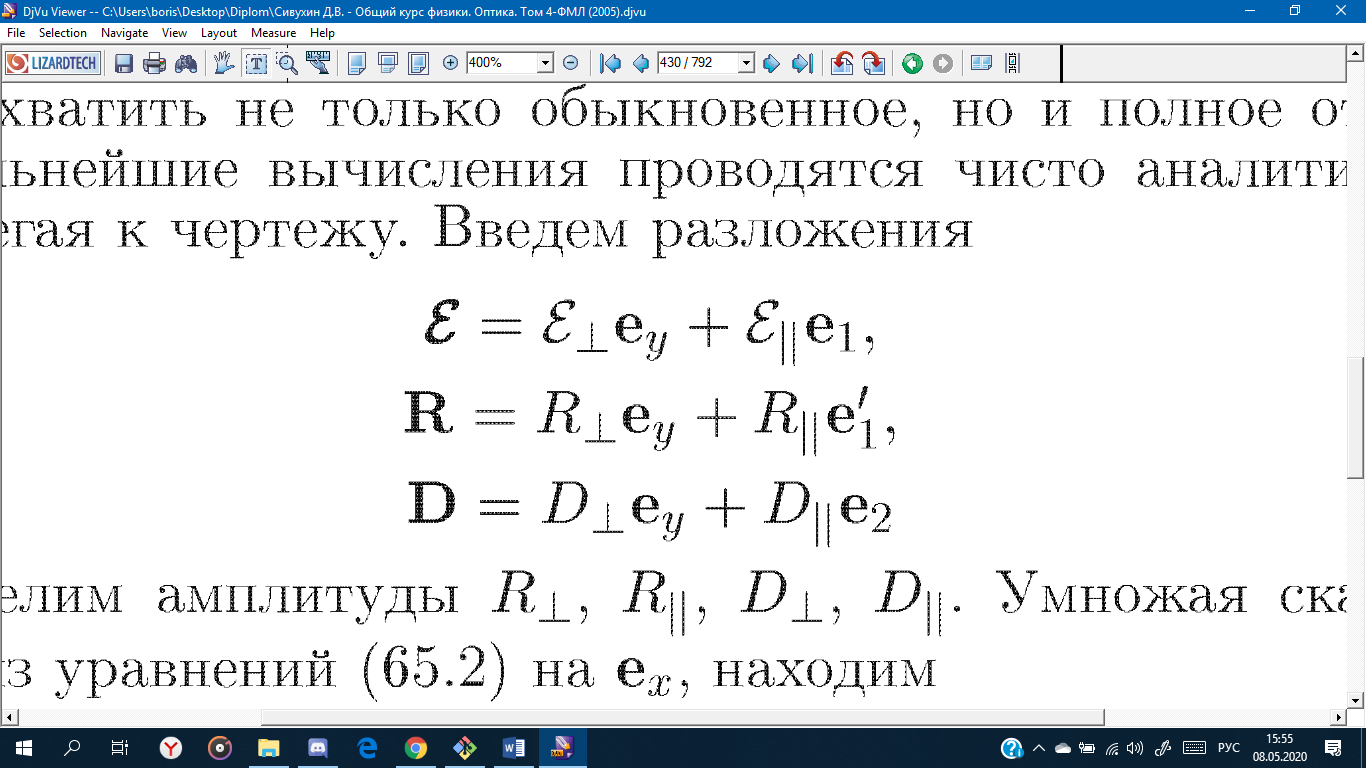
амплитуда составляющей преломленной волны, лежащая перпендикулярно плоскости падения

амплитуда составляющей преломленной волны, лежащая в плоскости падения

амплитуда составляющей отраженной волны, лежащая перпендикулярно плоскости падения

амплитуда составляющей падающей волны, лежащая перпендикулярно плоскости падения

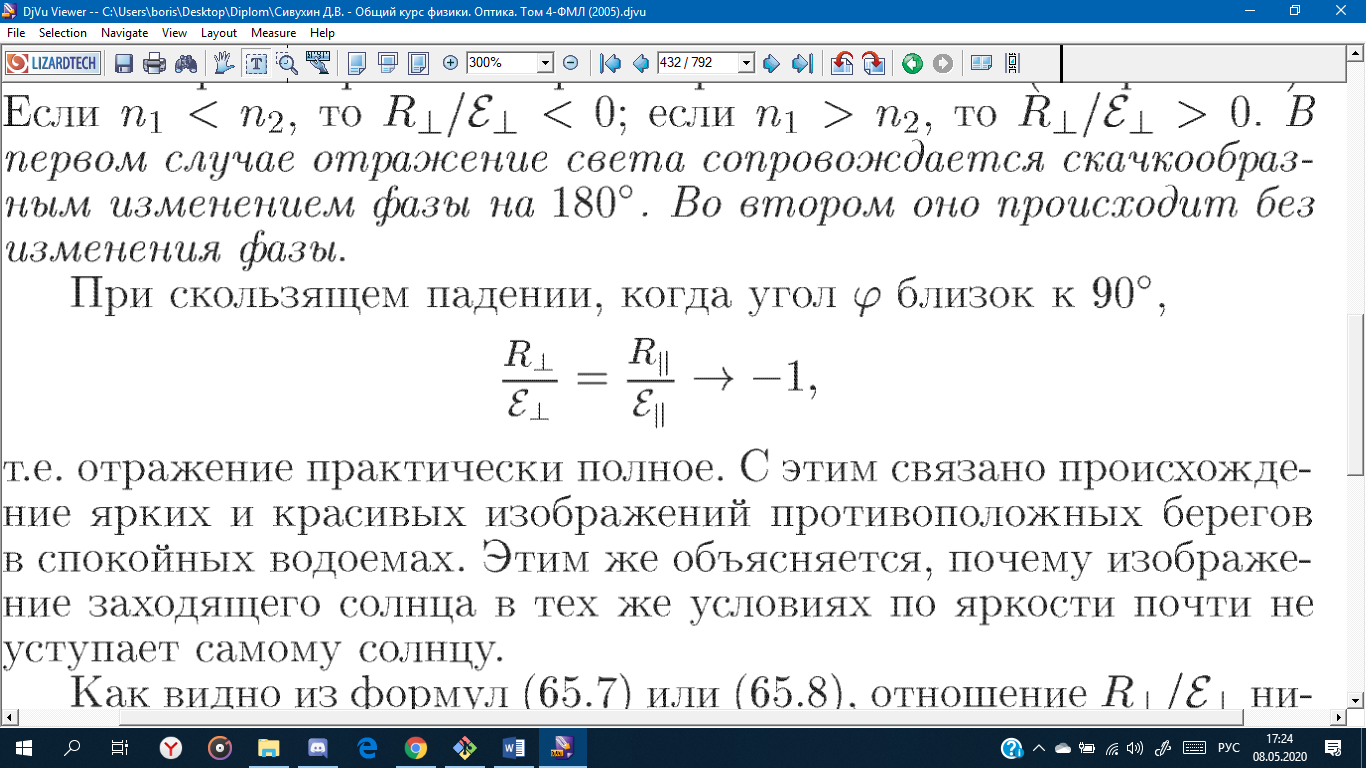


 амплитуда составляющей падающей волны, лежащая в плоскости падения

угол падения, угол преломления.

показатели преломления 1-й и 2-й сред.

Эти отношения называются коэффициентами Френеля. При обыкновенном отражении угол , а с ним и все коэффициенты Френеля вещественны. При скользящем падении, когда угол близок к 90°, отражение практически полное.



С этим связана видимость подводных объектов вбили к наблюдателю и постепенным «размытием» при отдалении. Этим же объясняется, почему изображение заходящего солнца в тех же условиях по яркости почти не уступает самому солнцу.

Вставить в литературу [Сивухин. Общий курс физики. Оптика]

Подповерхностное поглощение и рассеяние света.

При прохождении электромагнитной волны света через какое-либо вещество энергия этой волны расходуется на возбуждение колебаний электронов. Часть этой энергии переходит в другие виды энергии, а часть возвращается в виде вторичных волн, вызванных колебанием. Тут следует учесть взаимную интерференцию таких волн: в однородной среде вторичные волны полностью гасят друг друга, поэтому рассеяние света возникает только в неоднородной среде. Дифракционная картина в такой среде выглядит как равномерное распределение интенсивности света во всех направлениях и называется рассеянием света. Поглощением же света называется явление уменьшения энергии и, вследствие, интенсивности, световой волны при ее распространении в веществе из-за преобразования энергий.

Статьи\ сайты по оптике:

<http://aco.ifmo.ru/el_books/basics_optics/>

В настоящее время не существует способа полностью сымитировать сложное поведение света, учитывая все его физические свойства. Однако в 1986 году на конференции SIGGRAPH Джеймс Т. Каджия [] представил интегральное уравнение, которое является лучшей моделью симулирования освещения на сегодняшний день.

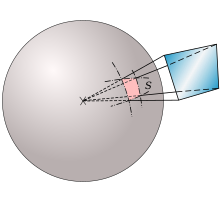
Глава 2 Уравнение рендеринга и PBR.

Энергетическая яркость

Энергетическая яркость – радиометрическая величина, которая характеризует наблюдаемую энергию, другими словами, количество света, на некоторой площади.

Чтобы дать точное определение, необходимо предварительно разъяснить следующие понятия:

* Телесный угол – это объемный угол, который характеризует долю пространства, которая может быть спроецирована на некоторую поверхность, обычно сферу, площади S. Мерой такого угла является отношение площади S к квадрату радиуса сферы : . В системе СИ поверхность с площадью вырезанная из сферы радиуса является телесным углом в 1 Стерадиан.



* Поток излучения Φ – мощность излучения, определяемая отношением энергии, переносимой излучением, ко времени переноса, значительно превышающему период электромагнитных колебаний. Энергия, испускаемая источником света, представляется как функция, зависящая от длин волн. Однако использование самих длин волн в компьютерной графике нецелесообразно, поэтому используются упрощения, такие как RGB.
* *Сила излучения – физическая величина, определяемая отношением потока излучения Φ, распространяющегося от источника излучения внутри малого телесного угла , содержащего рассматриваемое направление, к этому углу: –* не участвует дальше

Энергетическая яркость (*или лучистость*) – физическая величина, определяемая отношением потока излучения Φ к произведению телесного угла ω, в котором он распространяется, и проекции площади A, излучающего элемента поверхности на плоскость, перпендикулярную нормали к поверхности:

Уравнение рендеринга и уравнение отражения

Уравнение рендеринга вычисляет энергетическую яркость точки поверхности , исходящую в направлении (направление условного наблюдателя), как сумму собственной энергетической яркости поверхности и отраженного излучения, которое выражается интегралом. В частности, когда поверхность не излучает свет, т.е. при , выходит уравнение отражения.

Поверхность S телесного угла принимается бесконечно малой и фактически он становится вектором направления , а поверхность A – точкой .

В интеграле:

– интеграл по полусфере , необходимый для учета всех входящих векторов ,

эн. яркость падающего света в точку , направленная вдоль вектора ,

скалярное произведение векторов, которое обозначает угол между входящим (падающим) вектором и вектором нормали к поверхности ,

двулучевая функция отражательной способности (англ. Bidirectional reflectance distribution function[BRDF]) функция, которая на основе свойств поверхности, определяет количество отраженного света, другими словами, вклад, который вносит каждый падающий луч . Она является характеристикой конкретной поверхности, возвращает долю энергии, отраженной от материала в зависимости от направления . Следует уточнить, что здесь эта функция обозначена условно, потому как вариантов ее реализации достаточно много и количество принимаемых параметров также варьируется. Более подробно об этом далее.

Тут следует отметить, что данное уравнение рассматривается в контексте использования в компьютерной графике, поэтому в нем отсутствуют параметры t и , обозначающие время и длину волны соответственно.

Также существует BTDF (bidirectional transmittance distribution function) – Двулучевая функция пропускной способности, которая определяет количество поглощенного света и BSDF (Bidirectional Scattering Distribution Function) – Двулучевая функция распределения рассеивания, которая объединяет BRDF и BTDF и которая иногда, в зависимости от реализации, подставляется в уравнение рендеринга.

Аналитического решения этого уравнения нет, поэтому разные алгоритмы используют численное приближение. Для этого строится сетка, потом вычисляются значения для всех ее дискретных шагов, которые впоследствии усредняются.

Нельзя не упомянуть, что уравнение рендеринга не учитывает некоторых физических явлений света, например:

* Коэффициент прохождения волн
* Подповерхностное рассеивание
* Поляризацию
* Интерференция

Но это не сильно влияет на качество получаемых изображений, поэтому данный интеграл является лучшим приближением реального поведения света и используется во всех физически реалистичных рендерах.

PBR или Физически-корректный рендеринг

Physically-Based Rendering – концепция, которая основываясь на физических законах, стремиться добиться максимальной реалистичности в симулировании материалов и, тем самым, фотореалистичности получаемых рендерингом изображений. Физически-корректный рендеринг базируется на таких фактах, как:

* Теория микрограней – поверхность имеет микроскопические неровности.
* Закон сохранения энергии – поверхность не может отразить больше энергии, чем принимает
* Закон Френеля – поверхности имеют разную отражательную способность в зависимости от угла просмотра.
* Альбедо – учет отражения материалом волн определенной длины, другими словами, чистый цвет
* Проводник/диэлектрик – учет способности проводников и диэлектриков по-разному отражать свет, (60-90%) и (0-20%) соответственно.

Теория микрограней (Microfacet Theory)

В основе этой теории лежит представление о поверхности сред, как поверхности с микроскопическими неровностями, но тем не менее значительно большими, чем длина волны. Поэтому при падении световых лучей на такую поверхность может возникнуть три ситуации:

1. Самозатенение (англ. self shadowing) – луч света не долетает до поверхности, от которой он должен отразиться
2. Самоперекрытие (англ. masking) – отраженный луч сталкивается с препятствием и не вылетает.
3. Многократное отражение (англ. retroreflection) – ситуация, когда луч отражается от поверхности несколько раз, перед тем как улететь.

Таким образом, если микрограни ориентированы в разные стороны, то лучи рассеиваются в разные стороны – поверхность более шероховатая. В компьютерной графике используется коэффициент шероховатости, число принадлежащее отрезку [0, 1], применяемое к медианному вектору. Он определяет количество микрограней, направленных вдоль него: чем их меньше, тем более шероховатая поверхность.

Вектор , где

* вектор, направленный в сторону наблюдателя,
* вектор, направленный вдоль падающего света

Литература [Microfacet Models for Refraction through Rough Surfaces]

Закон сохранения энергии

Как уже упоминалось ранее, при столкновении с поверхностью часть света отражается, а часть поглощается. Из этого очевидно следует простой вывод: свет, который отражается от поверхности никак не может быть ярче света, который на нее падает.

Это реализуется за счет аналогичного разделения на зеркальную, которая отвечает за отражение, и диффузную, которая соответствует поглощению, компоненты. Тут следует уточнить, что концепция PRB не учитывает вторичные волны, т.е. подповерхностное рассеивание, которое реализуется другими способами, например, с помощью функции BSSRDF (Bidirectional Surface Scattering Distribution Function) - Двунаправленная Функция Поверхностного Рассеивания и Распределения, которая является ответвлением от BSDF.

[<https://render.ru/ru/articles/post/10906>] [<http://www.gamedev.ru/code/articles/Subsurface_scattering>]

Альбедо

Альбедо (англ. Albedo) – текстура, которая похожа на диффузную с той разницей, что она не содержит информации о затенении, а только информацию об отражательной способности материала, те только цвет.

Металлы

Геометрические законы отражения света от металлов такие же, что и для непоглощающих сред. Различие есть лишь в законах преломления. Преломленные волны затухают очень быстро, что по сути означает, что у металлов нет вторичных волн, а, следовательно, рассеивания. Таким образом у металлических поверхностей нет диффузной компоненты, а есть только зеркальная.

Ссылки на разные источники про PBR

<http://www.codinglabs.net/article_physically_based_rendering.aspx>

<https://blog.selfshadow.com/publications/s2013-shading-course/>

[s2013\_pbs\_physics\_math\_notebook.pdf](https://blog.selfshadow.com/publications/s2013-shading-course/hoffman/s2013_pbs_physics_math_notebook.pdf)

<http://graphicrants.blogspot.com/2013/08/specular-brdf-reference.html>

Статьи на хабре:

<https://habr.com/ru/post/426123/>

<https://habr.com/ru/post/326852/>

Ядром Физически-Корректного рендеринга является уравнение рендеринга и функция BRDF (а также ее аналоги). Существует большое количество разных вариантов этой функции, как физически правдоподобных, так и нет, как константных, так и линейных. От «физической полноты» этой функции сильно зависит реалистичность поверхности, другими словами, она является определяющей материала. В данной работе подробно рассмотрены только некоторые из этих вариантов, в контексте исследуемых рендеров и алгоритмов.

Двулучевая функция распределения рассеивания (BSDF)

Для appleseed:

* Glass BSDF

В данном шейдере используются функции, описанные в работе [].

BSDF,

BRDF, BTDF

=, где

вектор, направленный в сторону отражения

вектор направления падающего луча

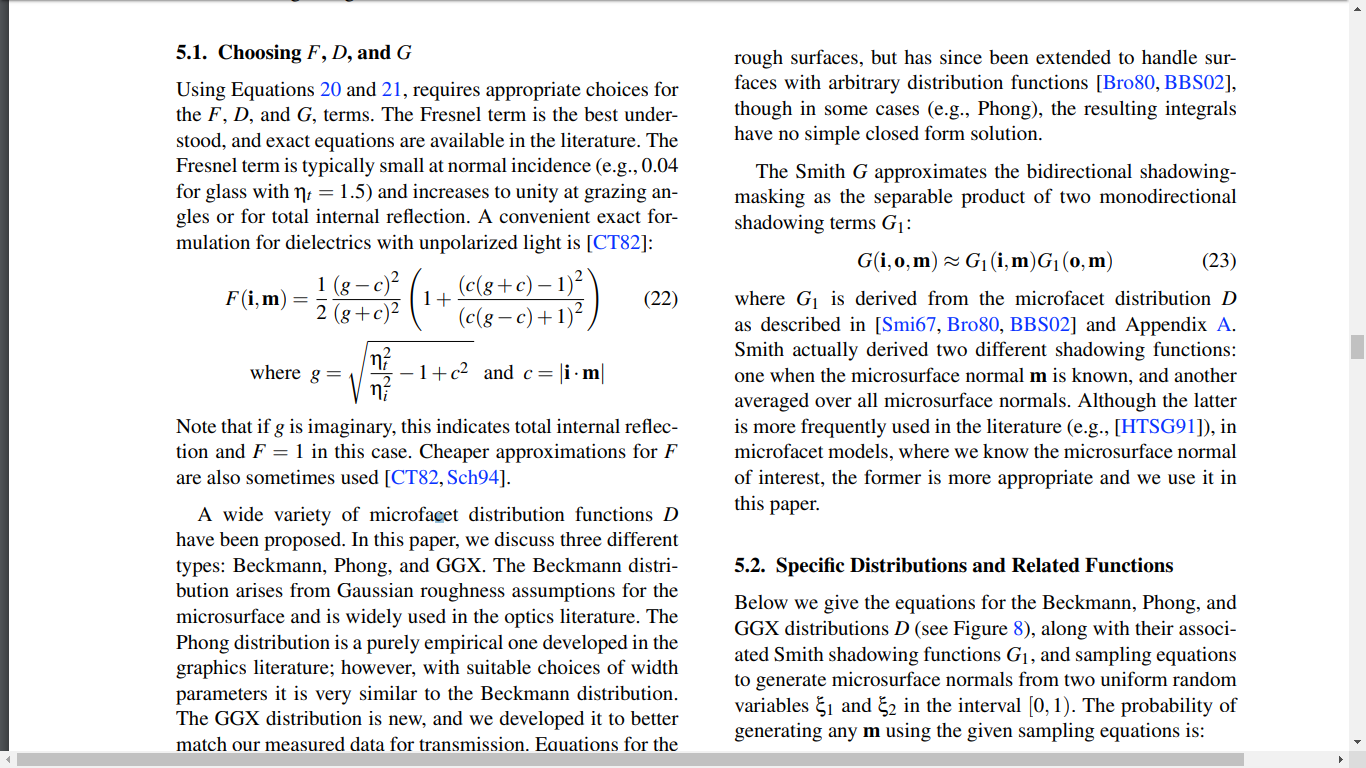
вектор микронормали, т.е. нормали к конкретной микрограни

BRDF

усредненный, по сути, обычный вектор нормали к поверхности

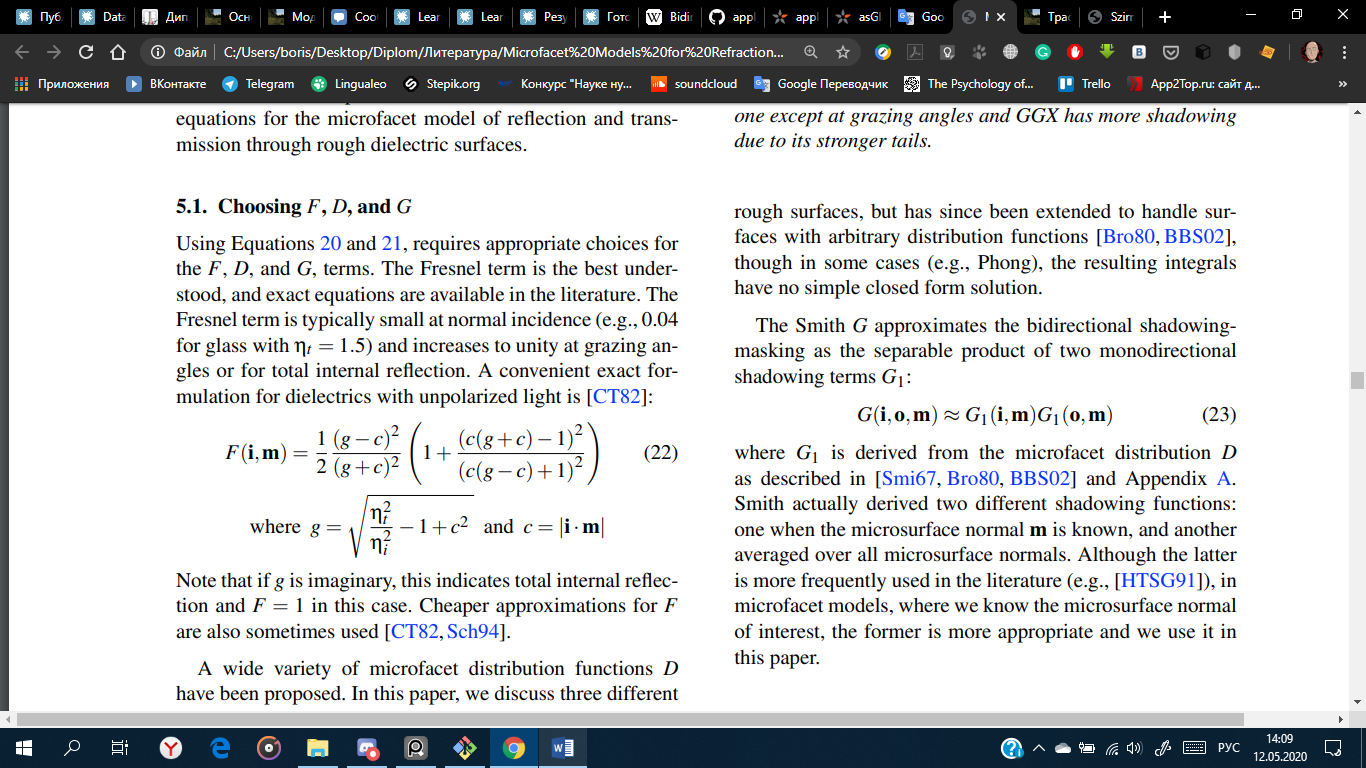
, где – медианный вектор

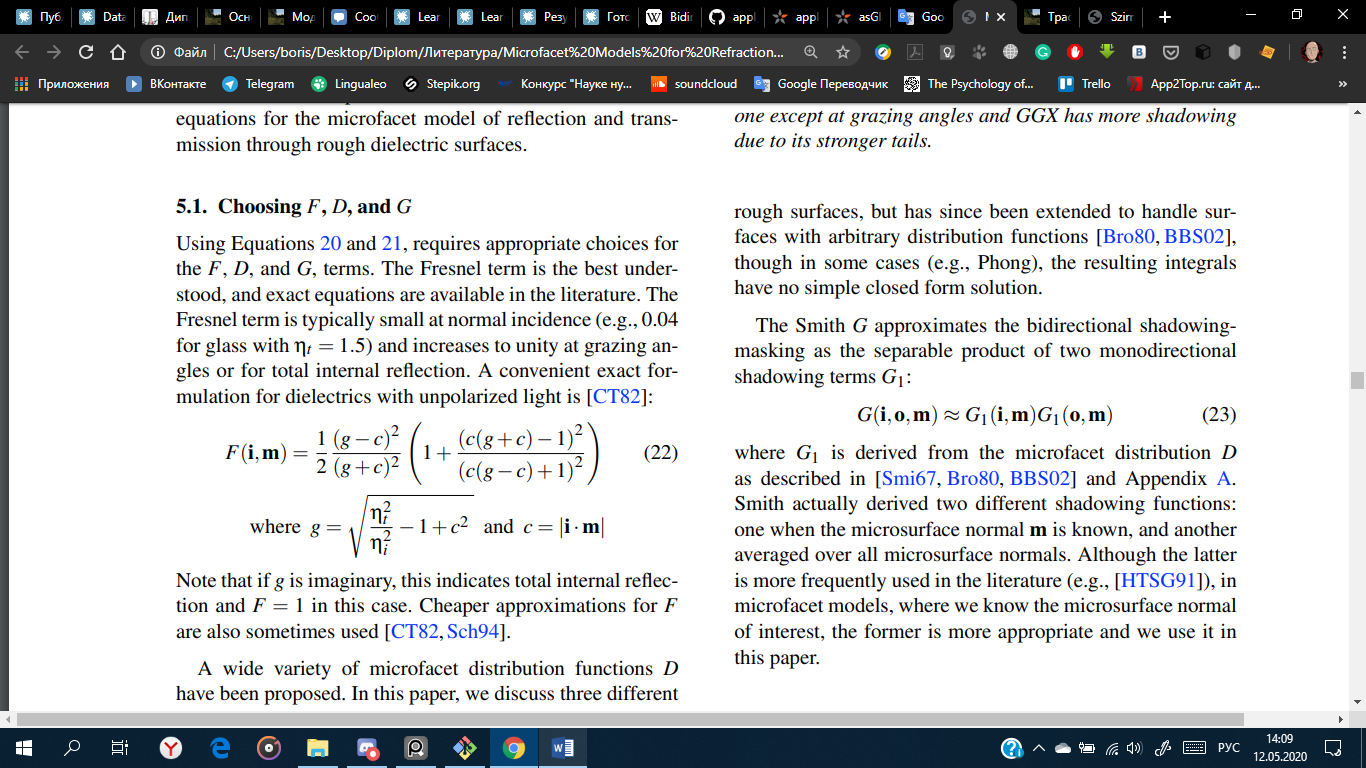
Функция уравнения Френеля (Fresnel equation) - описывает отношение отраженного и преломленного света, которое зависит от угла, под который мы смотрим на поверхность. Она возвращает процент отраженного света на основании угла, под которым условный наблюдатель видит эту поверхность.

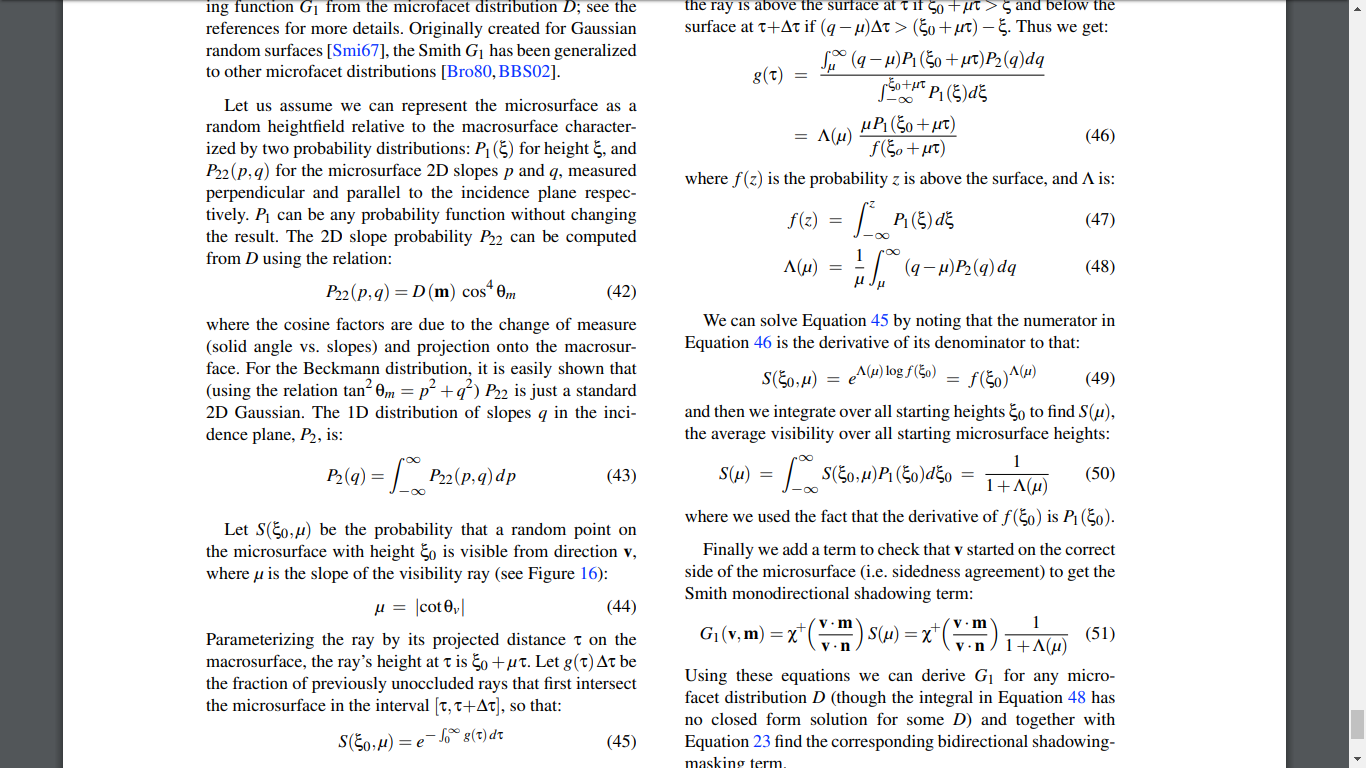


Функция геометрии (Geometry function), которая статистически аппроксимирует относительную площадь поверхности, где происходит самозатенение и перекрытие. Эта функция зависит от функции нормального распределения D и деталей микроповерхности. В общем случае используется приближение Смита [(Smith 1967, «Geometrical shadowing of a random rough surface»).]:









Функция нормального распределения (Normal Distributions Functions) аппроксимирует количество микрограней поверхности, ориентированных по медианному вектору, основываясь на шероховатости поверхности; это основная функция, аппроксимирующая микрограни. Существует несколько функций распределения, которые дают немного разный результат:

1. Распределение Бэкманна (Beckmann)





1. Распределение Фонга (Phong)





1. Распределение GGX





Bruce Walter, Stephen R. Marschner, Hongsong Li, and Kenneth E. Torrance. Microfacet models for refraction through rough surfaces. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques*, EGSR‘07, 195–206. Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association. URL: <http://dx.doi.org/10.2312/EGWR/EGSR07/195-206>, [doi:10.2312/EGWR/EGSR07/195-206](https://doi.org/10.2312/EGWR/EGSR07/195-206).

<https://github.com/appleseedhq/appleseed/tree/master/src/appleseed.shaders/src/appleseed>

Для Cycles:

<https://developer.blender.org/diffusion/C/browse/master/src/kernel/shaders/;e6d1b5efe5dfe1253f92030ec1bd2da21f673914>

<https://developer.blender.org/project/profile/26/>

Anisotropic BSDF <https://developer.blender.org/diffusion/C/browse/master/src/kernel/shaders/node_anisotropic_bsdf.osl>

<https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/anisotropic.html>

[Diffuse BSDF](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/diffuse.html)

[Emission](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/emission.html)

[Glass BSDF](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/glass.html)

[Glossy BSDF](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/glossy.html)

[Principled BSDF](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/principled.html)

[Principled Volume](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/volume_principled.html)

[Refraction BSDF](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/refraction.html)

[Specular BSDF](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/specular_bsdf.html)

[Subsurface Scattering](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/sss.html) ?

[Translucent BSDF](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/translucent.html)

[Transparent BSDF](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/transparent.html)

[Volume Absorption](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/volume_absorption.html)

[Volume Scatter](https://docs.blender.org/manual/en/latest/render/shader_nodes/shader/volume_scatter.html)

Для appleseed:

<https://github.com/appleseedhq/appleseed/tree/master/src/appleseed.shaders/src/appleseed>

* Lambertian BRDF (purely diffuse)
* Specular BRDF (perfect mirror)
* Specular BSDF (clear transparency)
* Diffuse BTDF (diffuse transparency)
* Oren-Nayar Microfacet BRDF
* Ward Microfacet BRDF
* Blinn Microfacet BRDF
* GGX Microfacet BRDF
* Disney's principled, layered BRDF
* Anisotropic Ashikhmin-Shirley BRDF
* Kelemen BRDF
* Glossy BRDF
* Physically-based metal BRDF
* Physically-based plastic BRDF
* Physically-based glass BSDF
* Sheen BRDF
* Arbitrary mixtures of BRDFs
* Energy compensation in microfacet models

Для LuxCore:

<https://wiki.luxcorerender.org/LuxCoreRender_Materials>

Также надо будет сказать в главе про Рендеринг, что в рендерах они реализованы в шейдерах

Дописать, основываясь на статьях  
<http://www.codinglabs.net/article_physically_based_rendering.aspx>

[https://blog.selfshadow.com/publications/s2013-shading-course/#course\_content](https://blog.selfshadow.com/publications/s2013-shading-course/" \l "course_content)

!! <http://blog.wolfire.com/2015/10/Physically-based-rendering>

<https://habr.com/ru/post/326852/> , <https://habr.com/ru/post/426123/>

<http://graphicrants.blogspot.com/2013/08/specular-brdf-reference.html>

Решение уравнения рендеринга дает цвет пиксела на изображении, однако осталось только его решить. Для этого существует множество подходов, таких как алгоритм излучения – radiosity. Однако намного большую популярность получили численные алгоритмы, основанные на многократном обсчете модели с использованием случайных чисел, для получения как можно большего кол-ва случайных исходов. Такой подход называется Метод Монте-Карло и используется в большом количестве областей науки. В данной работе описаны только некоторые из таких методов в контексте исследуемых рендеров.

Глава 3 Рендеры и алгоритмы трассировки.

Алгоритм Path Tracing



Рис. \_ Трассировка пути без (слева) и с (справа) учетом источника прямого освещения

Алгоритм трассировки пути был предложен Джеймсом Т. Каджия в той же работе, в которой было описано уравнение рендеринга. Он заключается в симуляции пути светового луча для каждого пикселя в изображении: когда луч пересекается с поверхностью, в псевдослучайном направлении выпускается отраженный луч и в обратном направлении теневой луч.

1. Из точки на экране выпускается луч и далее ищется пересечение его с каким-либо объектом.
2. Если пересечений нет, то возвращаем свет окружения, другими словами, фон.

Если пересечение произошло, то определяем альбедо объекта с как сумму его собственного излучения + прямое освещение источник света.

1. Функция BRDFSampling находит новое направление newray. Она возвращает 0, что означает, что новые лучи находить нельзя или не имеет смысла. В определении последнего участвует метод Русской Рулетки.
2. Рекурсивный вызов строит новый луч и возвращает энергетическую яркость объекта, с которым пересекся отраженный луч. Функция () возвращает число, которое определяет степень влияния этой эя на первоначальный объект. Таким образом в данной строке происходит учет влияния отражений луча от других объектов.

Trace(ray)

(object,) = FirstIntersect(ray)

**if** no intersection **then**

**return**

color = (, -ray.direction ) + DirectLightsource(, -ray.direction)

prob = BRDFSampling(-ray.direction, normal, newray)

**if** prob = 0 **then**

**return** color

color += Trace(newray)(newray.direction, normal, -ray.direction)/prob

**return** color

**end**

Следует отметить, что этот алгоритм генерирует все пути на основе семплирования BRDF\*, кроме последних. Последний получается путем семплирования источника света[60 стр]. Таким образом, если поверхность при отражении света является бликовой (близко к идеальной отражающей или идеальному поглощательной), то качество выборки по важности может быть довольно плохим. Поскольку почти идеальные поверхности, близкие к источникам света, ответственны за каустику, трассировка пути – плоха в рендеринге эффектов каустики.

\* Сэмплирование – это просто способ вычисления некоторой функции на множестве с помощью ее значений в точках – сэмплах. Самый простой способ – вычислить N значений в случайных точках и усреднить их. Чем больше сэмплов, тем точнее вычисляется функция.[58 стр]

[Szirmay-Kalos - MONTE-CARLO METHODS IN GLOBAL ILLUMINATION]

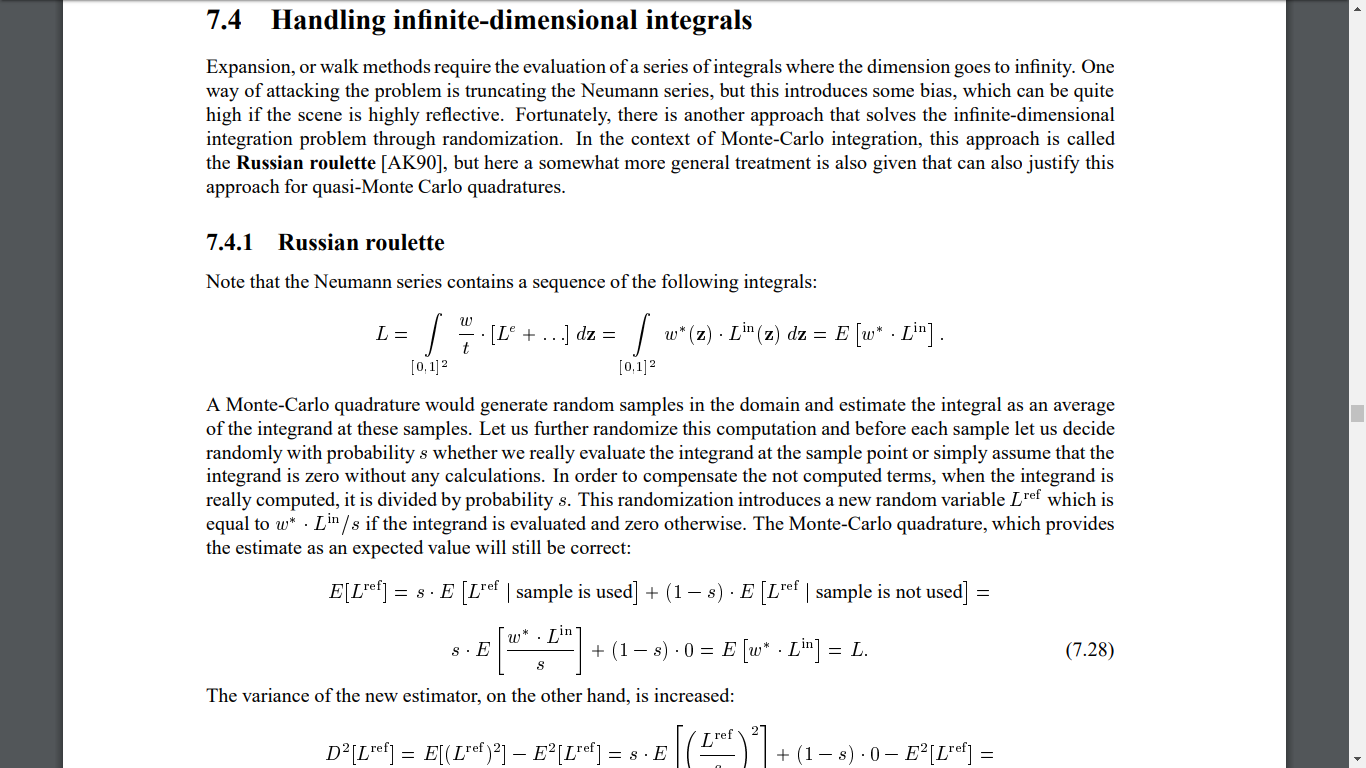
importance sampling

Отражение лучей в полностью случайном направлении может привести к большим и нецелесообразным затратам, поэтому имеет смысл выбирать направление в зависимости от возвращаемого значения BRDF. Если оно близко к нулю, то скорее всего такой луч не привнесет в сцену заметных изменений. И наоборот, в направления, которые приводят к большим значениям BRDF, следует выпускать больше лучей.

<https://habr.com/ru/post/436790/>

[The rendering equation](https://dl.acm.org/doi/pdf/10.1145/15922.15902)

Russian Roulette



Cycles

([https://docs.blender.org/manual/en/latest/render/cycles/render\_settings/sampling.html#sub-samples](https://docs.blender.org/manual/en/latest/render/cycles/render_settings/sampling.html" \l "sub-samples))

* Способы оптимизации – Importance Sampling (<https://habr.com/ru/post/326852/>) (<https://docs.blender.org/manual/en/latest/render/cycles/optimizations/index.html>)
* Упомянуть используемые функции BRDF

Рендер appleseed

* Алгоритм Path Tracing
* Алгоритм SPPM (<https://appleseedhq.net/features.html>)
* Алгоритм Light Tracing
* Способы оптимизации: Rendering Modes
* Упомянуть используемые функции BRDF

Рендер LuxCore

* Алгоритм Path Tracing (<https://wiki.luxcorerender.org/Render_Configuration>)
* Алгоритм Bidirectional Path Tracing
* Importance Sampling [notes\_on\_importance\_sampling.html](https://www.tobias-franke.eu/log/2014/03/30/notes_on_importance_sampling.html)

[developer.nvidia.com (chapter-20-gpu-based-importance-sampling)](https://developer.nvidia.com/gpugems/gpugems3/part-iii-rendering/chapter-20-gpu-based-importance-sampling)

* Упомянуть используемые функции BRDF

Алгоритмы описывать с помощью книги PBR

<http://www.pbr-book.org/3ed-2018/contents.html>

**Русская рулетка и расщепление**

Это два связанных метода, призванных улучшить сходимость метода Монте-Карло за счет повышения вероятности того, что каждая выборка внесет существенный вклад в результат.

Под «русской рулеткой» понимается стохастический процесс , основанный на методе Монте-Карло, позволяющий избавиться от «лишних» фотонов. Суть метода заключается в следующем: для конкретного фотона известно с какой поверхностью он столкнулся. Эта поверхность имеет коэффициенты диффузного () и зеркального () отражений. Для каждого отдельного фотона берётся некоторая случайная величина P и сравнивается со значениями этих коэффициентов. Рассматриваются три ситуации:

 → фотон отражается диффузно  
  
 → фотон отражается зеркально  
  
 → фотон поглощается

5 Алгоритм трассировки пути (Path-tracing)

Этот метод прослеживает путь отдельной частицы света, взаимодействующей с окружающей средой, до поглощения с использованием BRDF-семплирования. При попадании луча на поверхность испускается два новых луча: один - в произвольном направлении другой – до источника света. Доля отразившейся энергии при этом рассчитывается на основе ДФОC (либо испускается луч с одной и той же энергией по принципу «русской рулетки», при этом вероятность испускания вычисляется на основе ДФОC).

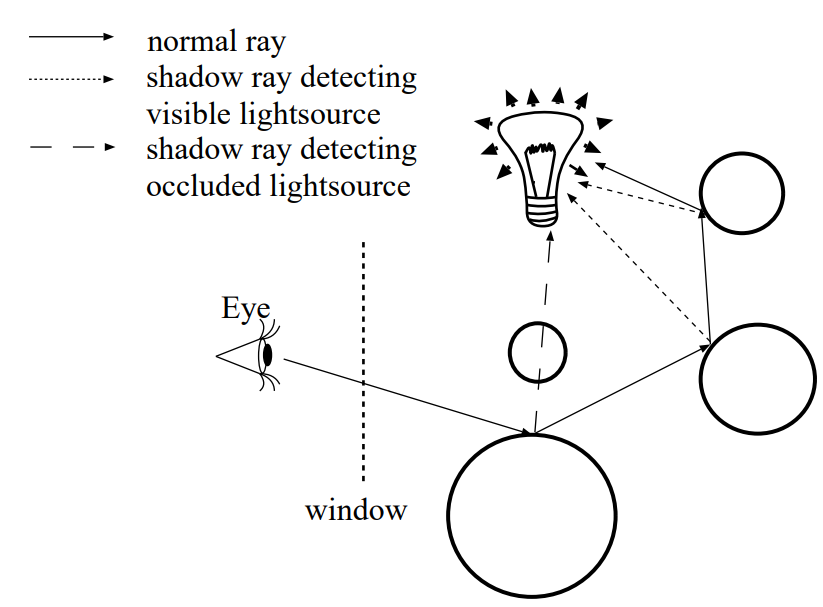


Рис. 6 Схема алгоритма трассировки пути с прямым вычислением источника света.

Данный алгоритм генерирует все пути фотона с помощью BRDF-семплирования, кроме последнего, который, в свою очередь, получается при помощи семплирования источника света. Таким образом, если поверхность, отражающая свет, является блестящей (близкой к идеальному зеркалу или идеальному рефрактору), то качество выборки по важности может быть довольно плохим. Поскольку почти идеально гладкие поверхности, фокусирующие отраженный свет, создают каустику, трассировка пути - как и другие gathering алгоритмы - плоха в симулировании этого эффекта.

Алгоритм двунаправленной трассировки лучей (Bi-directional path-tracing)

Двунаправленная трассировка (см. Рис. 7) подразумевает испускание лучей от камеры и от источника (максимальная глубина трассировки с каждой стороны фиксирована). Далее пути от камеры и источника начинают соединяться различными способами (так называемый детерминированный шаг алгоритма). В случае пересечения соединяющих лучей с другими объектами вклад всего пути будет равен нулю, иначе - не ноль.

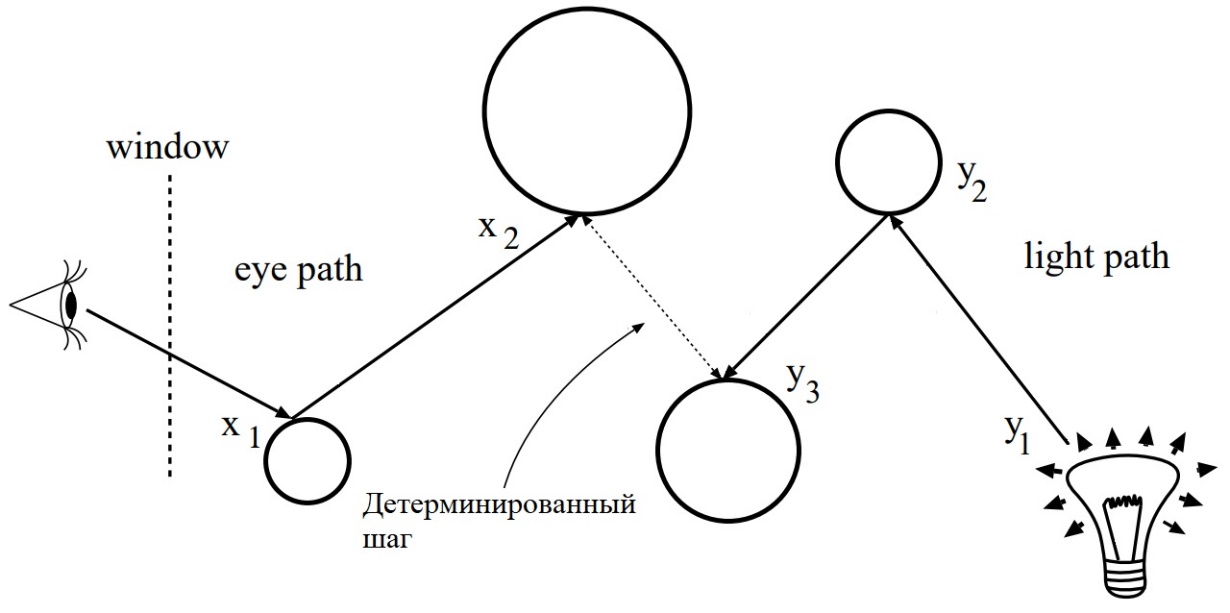


Рис. 7 Схема алгоритма двунаправленной трассировки пути с единичным детерминированным шагом.

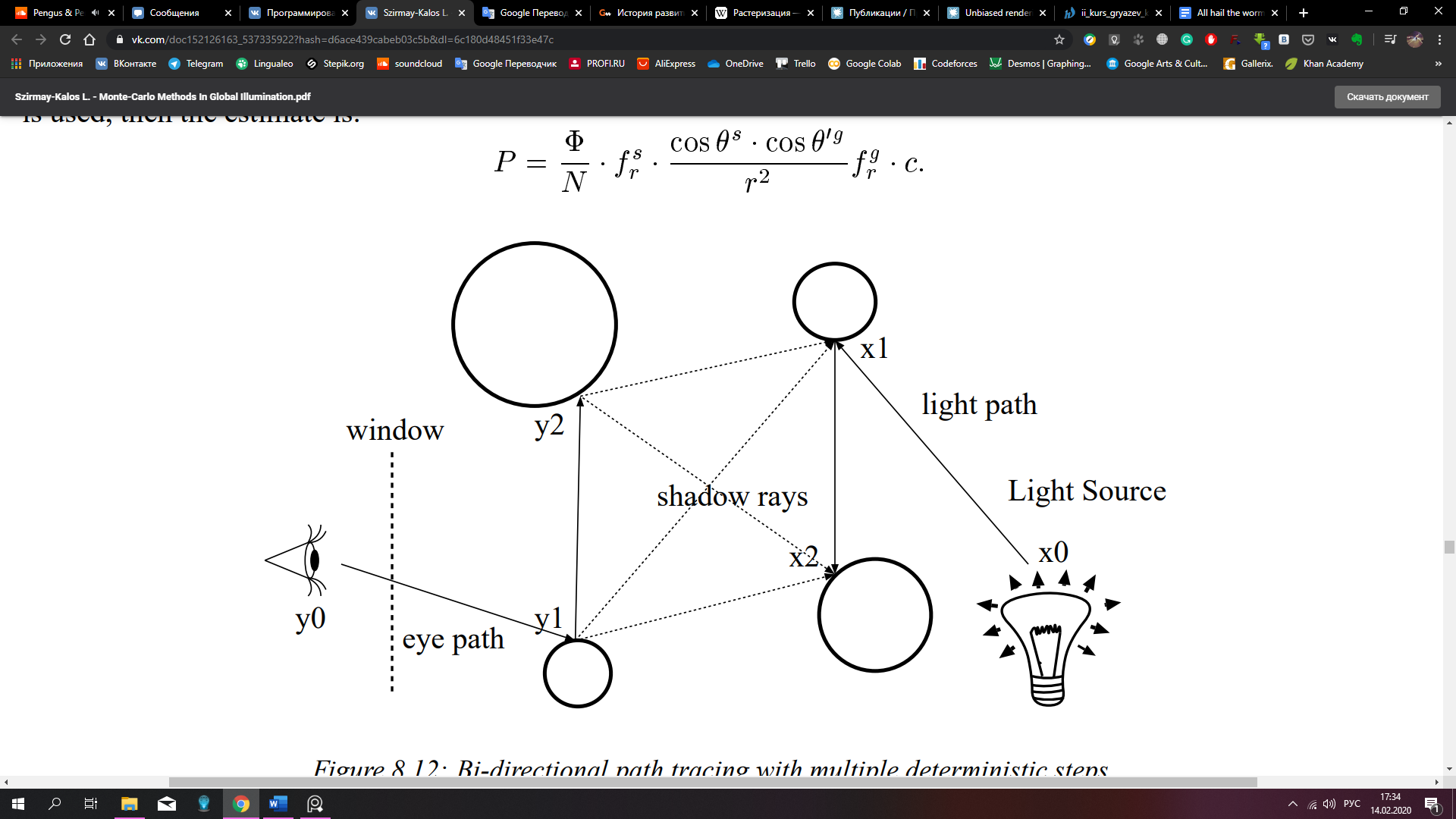


Рис. 8 Схема алгоритма двунаправленной трассировки пути с множественным детерминированным шагом.

Алгоритм Metropolis Light Transport

На первом шаге алгоритм ищет какой-то путь от источника до камеры, дающий ненулевой вклад. На втором этапе путь модифицируется специальным образом, основываясь на предыдущем пути. Новый путь обычно лучше, чем старый, если он вносит больший вклад в изображение. Если старый путь оказывается хуже, чем новый, то далее модификации подвергается новый путь, иначе – опять старый. Алгоритмы, отыскивающие «хорошие» пути потенциально быстрее сходятся, чем остальные алгоритмы трассировки.

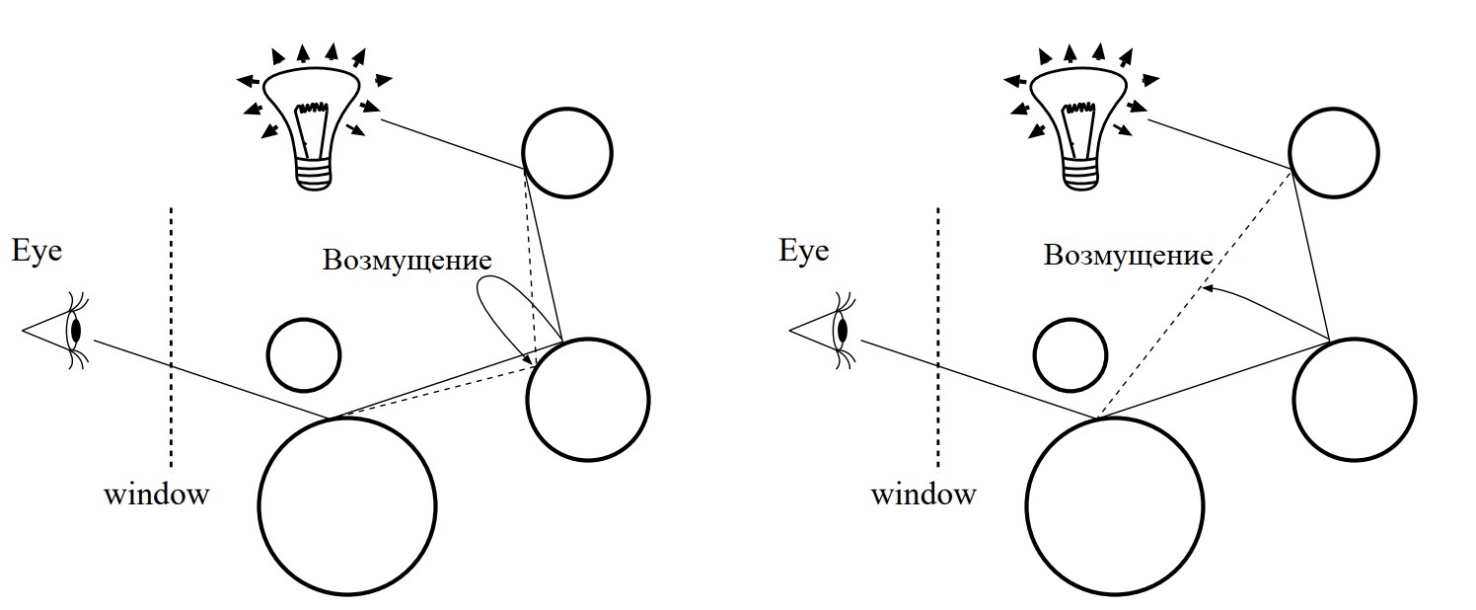
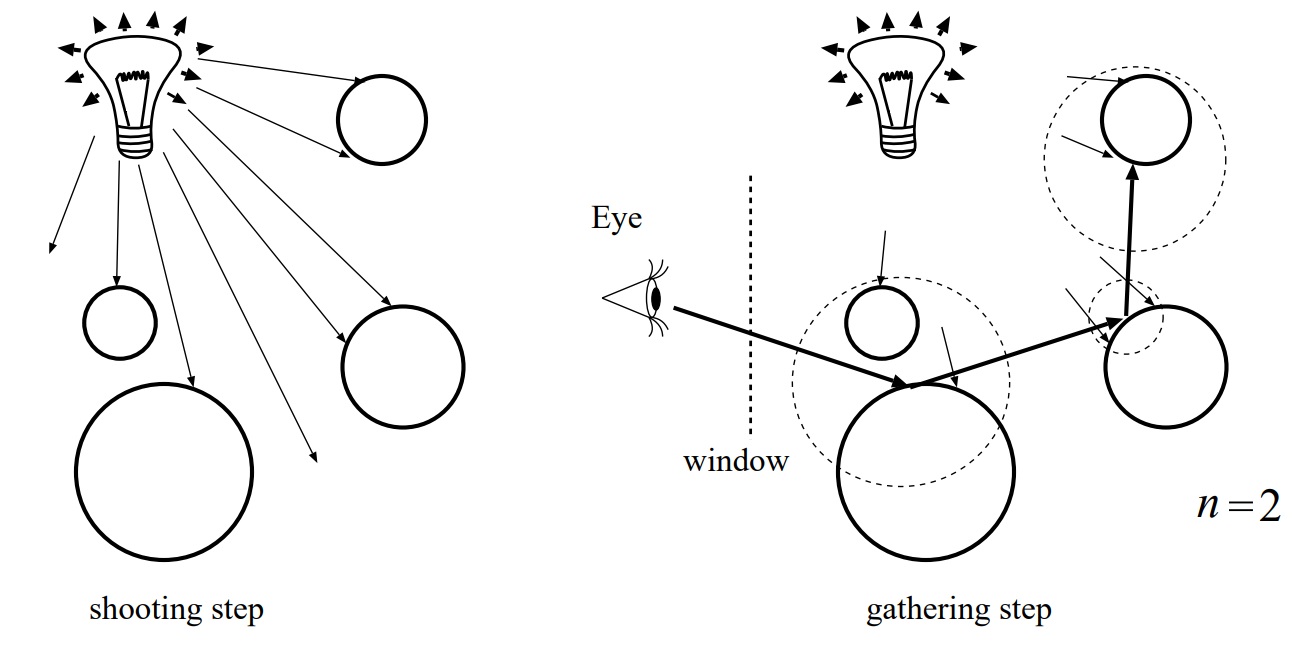


Рис. 9 Схема генерации модификаций пути в Metropolis Light Transport

Метод фотонных карт (Photon mapping)

Алгоритм состоит из двух этапов: трассировка и сохранение фотонов в фотонной карте и после расчёт освещённости с использованием данных из фотонной карты. На первом этапе трассировки из источника света испускаются фотоны. Фотоны испытывают ряд переотражений и преломлений, оставляя часть своей энергии на поверхностях объектов. Совокупность энергий фотонов, оставленных на поверхностях объектов, образуют фотонную карту. На следующем этапе необходимо собрать энергию в фотонных картах. Сбор освещенности можно делать по-разному. Чаще всего сбор идет по сфере по каждой отдельной поверхности объекта. В каждой точке сбора мы просматриваем окрестность этой точки и суммируем значения точек фотонной карты, попавших в эту окрестность, нормируем результат – это и будет результирующая освещенность в точке.

  
Рис. 10 Схема этапов метода фотонных карт

Глава 4 Способ сравнения

Основная метрика для сравнения изображений – отношение между максимумом возможного значения сигнала к мощности искажающего шума, PSNR (peak signal-to-noise ratio). Поскольку многие сигналы имеют широкий динамический диапазон, PSNR обычно измеряется в логарифмической шкале в децибелах [18].

PSNR наиболее часто используется для измерения уровня искажений при сжатии изображений. Эту меру можно связать со среднеквадратичной ошибкой (СКО, MSE, mean square error). Для двух монохромных изображений *I* и *K* размера *m×n*, одно из которых считается зашумленным приближением другого, MSE вычисляется по формуле



Выражение PSNR с использованием MSE представлено в формуле



здесь – максимальное значение яркости пикселя изображения. Чаще всего, . Для цветных изображений *MSE* – это комбинация отклонений по трём каналам (красный, зелёный и синий), поэтому в аналоге формулы для цветного изображения знаменатель умножается на 3.

Глава 5 Эксперименты

В экспериментальной части исследования на представленном наборе данных было протестировано несколько алгоритмов классификации. Их результаты сравниваются в выводах работы. Реализациям алгоритмов трассировки предоставлялось примерно одинаковое количество времени для подготовки изображения (около 1000 – 1500 с.). Результаты их работы сравнивались с эталонным изображением.

Далее эксперимент продолжался примерно до 15000 – 20000 с. Качество полученных изображений сравнивалось на промежуточных этапах и по истечении этого времени.

**3.1. Эксперимент 1**

В данном эксперименте на белой диффузной плоскости расположена модель граненого стакана. Материал стакана – гладкое бесцветное стекло. Фон окружения полностью черный. Исследуется корректность визуализации прозрачного стекла и оптического эффекта каустики.

Рендер Cycles использует Path tracing, LuxCore – Bidir + Metropolis, appleseed – SPPM.

Таблица 1. Время работы (с точностью до единиц) и качество (с точностью до 0.01 дБ) полученного изображения реализациями алгоритмов трассировки appleseed, Cycles и LuxCore при отрисовке модели прозрачного стеклянного стакана.

|  |  |  |
| --- | --- | --- |
| Реализация алгоритма трассировки | Затраченное время, с | Качество изображения, дБ |
| Appleseed | 1186 | 34.09 |
| 1745 | 35.52 |
| 3325 | 38.01 |
| 6247 | 41.05 |
| 11952 | 45.48 |
| Cycles | 1006 | 11.64 |
| 2116 | 11.66 |
| 4113 | 11.69 |
| 7907 | 11.76 |
| 16301 | 11.90 |
| Luxcore | 917 | 14.35 |
| 1846 | 14.52 |
| 3592 | 14.59 |
| 7261 | 14.64 |
| 14467 | 14.66 |

По результатам, представленным в таблице 1, можно сделать вывод о том, что Cycles и Luxcore дают примерно к 1000-й секунде изображения, схожие по качеству. Их результаты дальнейшем улучшаются очень слабо. За то же время Appleseed даёт существенно лучшее изображение. При этом, качество результата в начале эксперимента растёт почти линейно со временем, затем темп роста качества со временем снижается, но отличается от других реализаций на порядок.

**3.2. Эксперимент 2**

На плоскости с деревянной текстурой расположено 5 стеклянных бутылок: 2 с острыми углами, две без углов и одна многогранная. 2 из них окрашены в постоянный цвет. Слева расположен источник света типа Area. Исследуется корректность визуализации стекла в сложных случаях.

Рендер Cycles использует Path tracing, LuxCore – Bidir + Sobol, appleseed – Path tracing.

Таблица 2. Время работы (с точностью до секунды) и качество (с точностью до 0.01 дБ) полученного изображения реализациями алгоритмов трассировки appleseed, Cycles и LuxCore при отрисовке полупрозрачных стеклянных бутылок

|  |  |  |
| --- | --- | --- |
| Реализация алгоритма трассировки | Затраченное время, с | Качество изображения, дБ |
| Appleseed | 1318 | 38.10 |
| 3612 | 43.02 |
| 5525 | 45.38 |
| 7405 | 47.21 |
| 10710 | 50.14 |
| Cycles | 1569 | 19.20 |
| 3110 | 19.25 |
| 6319 | 19.28 |
| 12627 | 19.30 |
| Luxcore | 1304 | 20.28 |
| 2643 | 20.90 |
| 5586 | 21.26 |
| 10100 | 22.33 |

Результаты этого эксперимента схожи с предыдущим опытом: Cycles и Luxcore примерно к 1500-й секунде получают результаты, которые в дальнейшем улучшаются слабо. Темп роста качества изображения, построенного appleseed, со временем постепенно снижается, но даже к 10000-й секунде он на порядок выше, чем в других алгоритмах.

**3.3. Эксперимент 3**

В этом эксперименте исследовалась способность визуализировать материал неплотной ткани, которая должна пропускать свет. Также на этот материал нанесено несколько текстур – изображений в формате JPEG.

Рендер Cycles использует Path tracing, LuxCore – Bidir + Metropolis, appleseed – Path tracing.

Таблица 3. Время работы (с точностью до секунды) и качество (с точностью до 0.01 дБ) полученного изображения реализациями алгоритмов трассировки appleseed, Cycles и LuxCore при отрисовке полупрозрачной шторы с узором

|  |  |  |
| --- | --- | --- |
| Реализация алгоритма трассировки | Затраченное время, с | Качество изображения, дБ |
| Appleseed | 1925 | 19.76 |
| 4816 | 23.42 |
| 6327 | 24.68 |
| 18981 | 31.20 |
| Cycles | 1890 | 19.65 |
| 3800 | 19.91 |
| 7655 | 20.07 |
| 15178 | 20.15 |
| Luxcore | 1709 | 16.78 |
| 3650 | 16.98 |
| 7304 | 17.09 |
| 14742 | 17.15 |

Характер результатов эксперимента несколько отличается от результатов предыдущего опыта. В условиях ограниченного времени, примерно к 2000-й секунде все рендеры получают изображение примерно одинакового качества. При этом, темп роста качества изображений, построенных с помощью Cycles и Luxcore близок к нулю, что говорит о достижении пика своих возможностей, как и в предыдущих экспериментах. За то же время качество изображения, полученного при работе appleseed, возрастает.

**3.4. Эксперимент 4**

В данном эксперименте исследовалась способность визуализации материала полупрозрачной неплотной ткани, пропускающей прямые лучи света без рассеивания. В отличие от предыдущего эксперимента, где в качестве источника света была плоскость, на некотором расстоянии отдаленная от отверстия, здесь используется освещение окружения (World).

Рендер Cycles использует Path tracing, LuxCore – Path tracing, appleseed – Path tracing.

Таблица 4. Время работы (с точностью до единиц) и качество (с точностью до 0.01 дБ) полученного изображения реализациями алгоритмов трассировки Appleseed, Cycles и Luxcore при отрисовке модели полупрозрачной шторы

|  |  |  |
| --- | --- | --- |
| Реализация алгоритма трассировки | Затраченное время, с | Качество изображения, дБ |
| Appleseed | 1480 | 9.80 |
| 2917 | 9.85 |
| 5727 | 9.88 |
| Cycles | 625 | 12.97 |
| 1247 | 12.98 |
| 2507 | 12.99 |
| 5002 | 13.00 |
| Luxcore | 1168 | 23.79 |
| 1714 | 25.88 |
| 2376 | 27.58 |
| 4580 | 32.35 |
| 6819 | 36.69 |

При отображении этой сцены лучшим оказался Luxcore. Качество построенного им изображения лучше, чем у других рендеров, в условиях ограничения по времени (примерно к 1500-й секунде), темп роста качества изображения со временем – наилучший. Изображения, построенные Cycles и appleseed, фактически со временем не улучшаются.

Глава 5 Построение графиков, сравнение, выводы

В представленном исследовании произведено сравнение рендеров appleseed, Cycles и Luxcore по качеству построенных изображений трёхмерных сцен на основе алгоритма трассировки лучей. Результаты их работы сравнивались при ограниченном времени работы (около 1500 с.), а также рассмотрена динамика улучшения качества изображения при увеличении времени работы.

В большинстве тестов наилучшую динамику показал рендер appleseed, который и в условиях ограниченного времени давал наиболее фотореалистичное изображение, и с течением времени демонстрировал наилучший темп роста качества изображения. При этом, рендеры Cycles и Luxcore демонстрировали схожие результаты: они дали изображения приблизительно равного качества в условиях ограниченного времени, а с течением времени качество изображения менялось мало.

При этом, в эксперименте 4 рендер Luxcore продемонстрировал лучший результат и за органиченное время, и наилучшую динамику изменения качества изображения со временем.

**Приложения**

Получившиеся в результате экспериментов изображения доступны по ссылке:

**Литература**

1. Weinmann F. Comparison of Intersection Algorithms for SBR Ray Tracing on NURBS // Proceedings of the 2012 IEEE International Symposium on Antennas and Propagation. – 2012. – IEEE, Chicago, IL, USA. doi:10.1109/aps.2012.6349081
2. Wang J., Shen, Y., Ding, W. A Method for Ink-Wash Painting Rendering for 3D Scenes // Proceedings of the 2018 International Joint Conference on Information, Media and Engineering (ICIME). – 2018. – Pp. 205 – 210. – IEEE, Osaka, Japan. doi:10.1109/icime.2018.00050
3. Gu K., Liu M., Zhai G., Yang, X., Zhang, W. Quality Assessment Considering Viewing Distance and Image Resolution // IEEE Transactions on Broadcasting. – 2015. – Vol. 61(3). - Pp. 520–531. doi:10.1109/tbc.2015.2459851
4. Cho S., Shin S., Choi S. Haptic texture rendering using random fractal surface // Proceedings of the 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI). – 2017. – Pp. 290 – 292. – IEEE, Jeju, South Korea. doi:10.1109/ urai.2017.7992732
5. Shumskiy V. GPU Ray Tracing – Comparative Study of Ray-Triangle Intersection Algorithms // Transactions on Computational Science XIX. Lecture Notes in Computer Science. – 2013. – Vol. 7870. – Springer, Berlin, Heidelberg. – Pp. 78 – 91. doi:10.1007/978-3-642-39759-2
6. Zlatuska M., Havran V. Ray Tracing on GPU with CUDA - Comparative Study of Three Algorithms // WSCG 2010: Communication Papers Proceedings: 18th International Conference in Central Europe. – 2010. URL: https://www.researchgate.net/publication/228974703\_Ray\_Tracing\_on\_a\_GPU\_with\_CUDA-comparative\_study\_of\_three\_algorithms
7. Open Shading Language in Cycles [Электронный ресурс]. – Режим доступа: http://opensource.imageworks.com/?p=osl
8. Max Bounced [Электронный ресурс]. – Режим доступа: <https://docs.blender.org/manual/ru/dev/render/cycles/render_settings/light_paths.html>
9. Caustic [Электронный ресурс]. – Режим доступа: https://en.wikipedia.org/wiki/Caustic\_(optics)
10. Blender Motion Blur [Электронный ресурс]. – Режим доступа: https://docs.blender.org/manual/ru/dev/render/cycles/render\_settings/motion\_blur.html
11. Hachisuka, T., & Jensen, H. W. (2009). Stochastic progressive photon mapping. ACM Transactions on Graphics, 28(5), 1. doi:10.1145/1618452.1618487
12. OSL in appleseed [Электронный ресурс]. – Режим доступа: https://blenderseed.readthedocs.io/en/latest/osl/osl.html
13. appleseed Motion Blur [Электронный ресурс]. – Режим доступа: https://appleseed.readthedocs.io/projects/appleseed-blenderseed/en/latest/panels/render/motion\_blur.html
14. appleseed SSS [Электронный ресурс]. – Режим доступа: https://github.com/appleseedhq/appleseed/blob/master/docs/source/features/features.rst
15. Tiled Path (CPU/OpenCL) [Электронный ресурс]. – Режим доступа: https://wiki.luxcorerender.org/Render\_Configuration
16. И. М. Соболь. Численные методы Монте-Карло. М.: Наука, 1973 г.
17. PhotonGI cache [Электронный ресурс]. – Режим доступа: https://wiki.luxcorerender.org/PhotonGI
18. Peak signal-to-noise ratio [Электронный ресурс]. – Режим доступа: https://en.wikipedia.org/wiki/Peak\_signal-to-noise\_ratio

**References**

1. Weinmann F. Comparison of Intersection Algorithms for SBR Ray Tracing on NURBS// Proceedings of the 2012 IEEE International Symposium on Antennas and Propagation. – 2012. – IEEE, Chicago, IL, USA. doi:10.1109/aps.2012.6349081
2. Wang J., Shen, Y., Ding, W. A Method for Ink-Wash Painting Rendering for 3D Scenes // Proceedings of the 2018 International Joint Conference on Information, Media and Engineering (ICIME). – 2018. – Pp. 205 – 210. – IEEE, Osaka, Japan. doi:10.1109/icime.2018.00050
3. Gu K., Liu M., Zhai G., Yang, X., Zhang, W. Quality Assessment Considering Viewing Distance and Image Resolution // IEEE Transactions on Broadcasting. – 2015. – Vol. 61(3). - Pp. 520–531. doi:10.1109/tbc.2015.2459851
4. Cho S., Shin S., Choi S. Haptic texture rendering using random fractal surface // Proceedings of the 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI). – 2017. – Pp. 290 – 292. – IEEE, Jeju, South Korea. doi:10.1109/ urai.2017.7992732
5. Shumskiy V. GPU Ray Tracing – Comparative Study of Ray-Triangle Intersection Algorithms // Transactions on Computational Science XIX. Lecture Notes in Computer Science. – 2013. – Vol. 7870. – Springer, Berlin, Heidelberg. – Pp. 78 – 91. doi:10.1007/978-3-642-39759-2
6. Zlatuska M., Havran V. Ray Tracing on GPU with CUDA - Comparative Study of Three Algorithms // WSCG 2010: Communication Papers Proceedings: 18th International Conference in Central Europe. – 2010. URL: https://www.researchgate.net/publication/228974703\_Ray\_Tracing\_on\_a\_GPU\_with\_CUDA-comparative\_study\_of\_three\_algorithms
7. Open Shading Language in Cycles [Электронный ресурс]. – Режим доступа: http://opensource.imageworks.com/?p=osl
8. Max Bounced [Электронный ресурс]. – Режим доступа: <https://docs.blender.org/manual/ru/dev/render/cycles/render_settings/light_paths.html>
9. Caustic [Электронный ресурс]. – Режим доступа: https://en.wikipedia.org/wiki/Caustic\_(optics)
10. Blender Motion Blur [Электронный ресурс]. – Режим доступа: https://docs.blender.org/manual/ru/dev/render/cycles/render\_settings/motion\_blur.html
11. Hachisuka, T., & Jensen, H. W. (2009). Stochastic progressive photon mapping. ACM Transactions on Graphics, 28(5), 1. doi:10.1145/1618452.1618487
12. OSL in appleseed [Электронный ресурс]. – Режим доступа: https://blenderseed.readthedocs.io/en/latest/osl/osl.html
13. appleseed Motion Blur [Электронный ресурс]. – Режим доступа: https://appleseed.readthedocs.io/projects/appleseed-blenderseed/en/latest/panels/render/motion\_blur.html
14. appleseed SSS [Электронный ресурс]. – Режим доступа: https://github.com/appleseedhq/appleseed/blob/master/docs/source/features/features.rst
15. Tiled Path (CPU/OpenCL) [Электронный ресурс]. – Режим доступа: https://wiki.luxcorerender.org/Render\_Configuration
16. И. М. Соболь. Численные методы Монте-Карло. М.: Наука, 1973 г.
17. PhotonGI cache [Электронный ресурс]. – Режим доступа: https://wiki.luxcorerender.org/PhotonGI
18. Peak signal-to-noise ratio [Электронный ресурс]. – Режим доступа: https://en.wikipedia.org/wiki/Peak\_signal-to-noise\_ratio

Об авторах

Гогинян Борис Андреевич, студент группы КМБO-03-16 кафедры Высшей математики Института кибернетики ФГБОУ ВО «МИРЭА – Российский технологической университет» (119454, Россия, Москва, пр-т Вернадского, д. 78).

Парфенов Денис Васильевич, кандидат технических наук, доцент кафедры Высшей математики Института кибернетики ФГБОУ ВО «МИРЭА – Российский технологической университет» (119454, Россия, Москва, пр-т Вернадского, д. 78).

Петрусевич Денис Андреевич, кандидат физико-математических наук, доцент кафедры Высшей математики Института кибернетики ФГБОУ ВО «МИРЭА – Российский технологической университет» (119454, Россия, Москва, пр-т Вернадского, д. 78).

About authors

Boris A. Goginyan, student of the Higher Mathematics Chair, Institute of Cybernetics, Russian Technological University (78, Vernadskogo Pr., Moscow 119454, Russia).

Denis V. Parfenov, PhD, associate professor of the Higher Mathematics Chair, Institute of Cybernetics, Russian Technological University (78, Vernadskogo Pr., Moscow 119454, Russia).

Denis A. Petrusevich, PhD, associate professor of the Higher Mathematics Chair, Institute of Cybernetics, Russian Technological University (78, Vernadskogo Pr., Moscow 119454, Russia).