

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
Учреждение образования
«Витебский государственный технологический университет»



Компьютерные технологии в дизайн-проектировании. Настройки V-Ray

Методические указания к практическим занятиям
для студентов специальностей
1-19 01 01-01 «Дизайн объёмный»,
1-19 01 01-02 «Дизайн предметно-пространственной среды»

Витебск
2017

УДК 004.925.4

Составитель:

Н. А. Абрамович

Рекомендовано к изданию редакционно-издательским советом УО «ВГТУ», протокол № 7 от 27.10.2017.

Компьютерные технологии в дизайн-проектировании. Настройки V-ray. : методические указания к практическим занятиям / сост. Н. А. Абрамович. – Витебск : УО «ВГТУ», 2017. – 44 с.

Методические указания являются руководством для практических занятий для специальностей 1-19 01 01-01 «Дизайн объёмный», 1-19 01 01-02 «Дизайн предметно-пространственной среды» по дисциплине «Компьютерные технологии в дизайн-проектировании» и содержат теоретический материал по настройке визуализатора V-Ray, принципах и особенностях его работы, определяют основные ошибки пользователей, рассматривают примеры использования ряда параметров, влияющих на итоговое качество получаемых изображений.

УДК 004.925.4

© УО «ВГТУ», 2017

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 ТИПЫ РЕНДЕРА	5
2 МЕТОДЫ ПРОСЧЕТА СЦЕНЫ	6
3 ОСНОВНЫЕ ПРИНЦИПЫ V-RAY ..	7
3.1 Вычисление освещенности и цвета произвольной точки трехмерной сцены	8
3.2 Непрямое освещение (Indirect Illumination)	9
3.3 Трассировка лучей (Raytracing)	11
3.4 Настройка GI	13
3.5 Алгоритм просчета GI Irradiance map	17
3.6 Алгоритм просчета GI Brute Force (QMC)	25
3.7 Алгоритм просчета GI Photon Map	26
3.8 Алгоритм просчета GI Light cache	26
3.9 Выбор связки алгоритмов для просчета GI	32
3.10 Свиток V-Ray Frame Buffer	33
3.11 Свиток V-Ray Global switches	33
3.12 Свиток V-Ray Image sampler (Antialiasing)	34
3.13 Свиток V-Ray Color mapping	39
3.14 Свиток V-Ray Environment	39
4 РАСПРОСТРАНЕННЫЕ ПРОБЛЕМЫ ВТОРИЧНОГО ОСВЕЩЕНИЯ	40
ЛИТЕРАТУРА	43

ВВЕДЕНИЕ

Многие элементы визуального окружения современного человека сегодня создаются с помощью программ компьютерной графики. Без визуализаций, сделанных 3D-художниками, не может обойтись ни архитектурная или дизайнерская студия, ни производители компьютерных игр. Технология создания подобного изображения – фотореалистического или имитирующего различные художественные техники – состоит из нескольких технологических этапов. Рендер – это важнейший из них, часто заключительный, от которого зависит конечный результат.

Рендеринг (англ. *rendering* – «визуализация») в компьютерной графике – процесс получения изображения по модели с помощью компьютерной программы. Слово «рендер» (или «рендеринг») пришло, как и многое, связанное с IT-технологиями, из английского языка. Происходит оно от старофранцузского *rendre*, означающего «делать», «дать», «возвратить», «вернуть». Более глубокие корни этого глагола восходят к древней латыни: *re* – префикс, означающий «назад», и *dare* – «давать» [1].

Отсюда – один из смыслов современного термина. Рендер – это в том числе процесс воссоздания плоскостного изображения на основе трехмерной модели, содержащей сведения о физических свойствах объекта – его форме, фактуре поверхности, освещенности и так далее (рис. 1).



Рисунок 1 – Рендер трехмерной модели

Программа рендеринга рассчитывает освещенность всех объектов сцены с учетом источников света, свойств материалов, световых эффектов, таких как тени и отражения, прозрачность, преломления лучей, атмосферные эффекты и создает на плоскости экрана перспективную проекцию изображения сцены с заданной точки наблюдения [2].

Вошедшее сначала в лексикон тех, кто профессионально занимается цифровыми технологиями создания изображений, это слово все чаще применяется и в повседневном обиходе. Предоставить готовый рендер просят, например, при заказе мебели – отдельного объекта или обстановки целого помещения, а при проектировании интерьера или всего здания рендер – это одно из основных средств донести до заказчика смысл идей архитектора или дизайнера. Имеется синоним, близкий по значению и чаще применяемый в обычной среде, – визуализация.

Среди профессионалов архитектурной или игровой компьютерной графики сегодня принято иметь узкую специализацию: есть те, кто занимается моделингом – создает трехмерные объекты, и те, кто обеспечивает рендеринг готовой сцены – выставляет освещение, выбирает точку зрения и настраивает, а потом и запускает рендер-программу.

1 ТИПЫ РЕНДЕРА

Различают два основных типа рендера в зависимости от скорости, с которой должно происходить получение готового изображения: **online и пререндеринг** [3].

Первый – рендеринг в реальном времени, необходимый в интерактивной графике, в основном в компьютерных играх.



Рисунок 2 – Online

Здесь нужен быстрый рендер, изображение должно выводиться на экран мгновенно, поэтому многое в сцене рассчитывается заранее и сохраняется в ней в виде отдельных данных. К ним относятся текстуры, определяющие внешний вид объектов и освещение. Программы, используемые для онлайн-рендера, используют в основном ресурсы графической карты и оперативной памяти компьютера и в меньшей степени – процессора.

Для рендера сцен, более сложных визуально, а также там, где вопрос скорости не так актуален, когда гораздо важнее качество рендера, используются другие методы и программы для рендеринга (*пререндеринг*). В этом случае используется вся мощь многоядерных процессоров, выставляются самые высокие параметры разрешения текстур, обсчета освещения [4]. Часто применяется и постобработка рендера, позволяющая добиться высокой степени фотореалистичности или нужного художественного эффекта.



Рисунок 3 – Пререндеринг

2 МЕТОДЫ ПРОСЧЕТА СЦЕНЫ

На сегодня в основе самых распространенных программ-рендеров лежат три основных вычислительных метода:

Растеризация (Scanline) – метод, при котором изображение создается просчетом не отдельных точек-пикселей, а целых граней-полигонов и крупных участков поверхностей. Текстуры, определяющие свойства объектов, как и свет в сцене, зафиксированы в виде неизменных данных. Получаемое изображение часто не отражает перспективных изменений освещенности, глубины резкости и

т. д. Чаще применяется в системах для просчета сцен в играх и в видеопродакшене.

Трассировка лучей (Raytracing) – физика сцены просчитывается на основе лучей, исходящих из объектива виртуальной камеры и анализа взаимодействия каждого луча с объектами, с которыми он встречается в сцене. В зависимости от количества и качества таких «отскоков» имитируется отражение или преломление света, его цвет, насыщенность и т. д. Качество получаемой картинке по сравнению с растеризацией значительно выше, но за её реалистичность приходится платить повышенным расходом ресурсов.

Расчет отраженного света (Radiosity) – каждая точка, каждый пиксель изображения наделяется цветом, который не зависит от камеры. На него влияют глобальные и местные источники света и окружение. Такой метод позволяет рассчитать появление на поверхности модели цветовых и световых рефлексов от рядом расположенных объектов [5].

Практика показывает, что самые продвинутые и популярные системы рендера используют сочетание основных методов. Это позволяет добиться максимального фотореализма и достоверности в отображении физических процессов в данной сцене.

Одни рендереры устаревают и умирают, другие покупают гиганты 3D-индустрии (встраивают и переименовывают), на смену им приходят новые революционные продукты. С выходом новой версии полностью устаревают технологии предыдущей. Исчезают старые понятия и появляются новые термины. Поэтому более-менее постоянным остаются лишь очень общие принципы, о которых ниже и пойдет речь.

3 ОСНОВНЫЕ ПРИНЦИПЫ V-RAY

Среди современных рендер-программ V-Ray пользуется наибольшей популярностью. V-Ray обладает целым рядом интересных инновационных технологических решений, обеспечивающих ему дополнительное преимущество в качестве и скорости расчетов [6]. Этот факт тем более удивителен и замечателен, что алгоритм и ядро программы разрабатывались всего несколькими людьми.

V-Ray – система рендеринга (визуализации изображения), разработанная компанией Chaos Group (Болгария), работает как плагин для Autodesk 3ds Max.

Рассматривать подробно какую-либо версию V-ray нет никакого смысла. Весь этот клад знаний быстро превратится в ненужный хлам по причине почти моментального устаревания. Поэтому далее будут отображены ключевые теоретические моменты визуализации, на которых она базируется, а также базовые параметры, постоянно кочующие из версии в версию.

3.1 Вычисление освещенности и цвета произвольной точки трехмерной сцены

Как уже было сказано, любой визуализатор рассчитывает освещенность всех объектов сцены с учетом источников света, свойств материалов, световых эффектов, таких как тени и отражения, прозрачность, преломления лучей, атмосферные эффекты и создает на плоскости экрана перспективную проекцию изображения сцены с заданной точки наблюдения. Для этого необходимо вычисление освещенности и цвета произвольной точки трехмерной сцены.

Сюда входят:

1. Прежде всего, **цвет объекта**.
2. Вторая компонента освещенности объектов определяется **зеркальным (или близким к зеркальному) отражением** от окружения **и прозрачностью** самого объекта. Эти две компоненты характеризуют материал объекта.
3. **Освещенность объектов от источников света**, находящихся в прямой видимости, когда объект и источник можно соединить прямой линией.
4. Многократные **диффузные переотражения** света окружающими объектами. Говоря на художественном языке, система рефлексов.
5. Пятая компонента освещенности занимается специальным случаем освещенности – рассчитывает световые эффекты, возникающие в результате фокусировки из-за преломлений или отражений лучей света в некоторой области поверхности. Эти эффекты получили название **caustic-эффектов освещения**, а прекрасным иллюстрирующим примером «из жизни» может служить линза, фокусирующая солнечный свет на поверхности объекта.

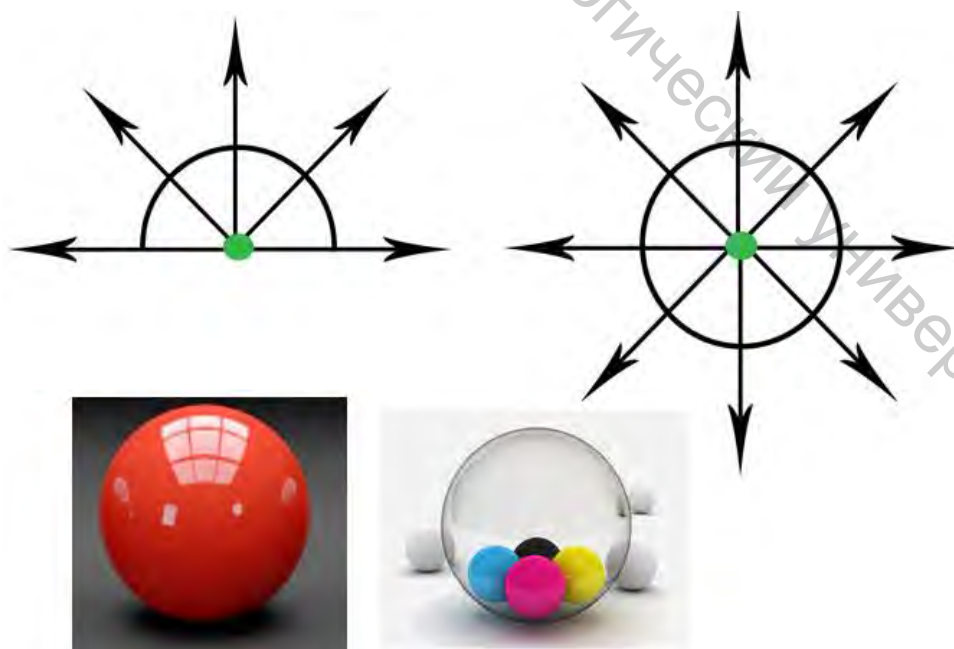


Рисунок 4 – Направления интегрирования освещенности

Для точного расчета всего света, падающего на данную точку поверхности, требуется просуммировать лучи света, приходящие в нее со всех направлений. Это приводит к необходимости интегрирования освещенности (рис. 4) по полусфере, окружающей точку, если она принадлежит непрозрачной поверхности, или – по сфере, если поверхность является еще и прозрачной.

3.2 Непрямое освещение (Indirect Illumination)

Для просчета первых трех компонент – прямого освещения, V-Ray обладает отдельным алгоритмом, способным работать независимо. Убедиться в этом просто, достаточно отрендерить трехмерную сцену без GI (убрать галочку в чекбоксе On на закладке V-Ray/Indirect Illumination). В арсенале средств модуля расчета прямого освещения имеется возможность обработки пространственных источников, так что посчитать мягкие тени не составляет никаких проблем. Настройки расчета прямого освещения присутствуют в параметрах источников света и теней (затухание, тип источника, параметры теней и др.).

Результат вряд ли будет реалистичным, так как в местах, закрытых от прямого света, будет чернота (рис. 5).

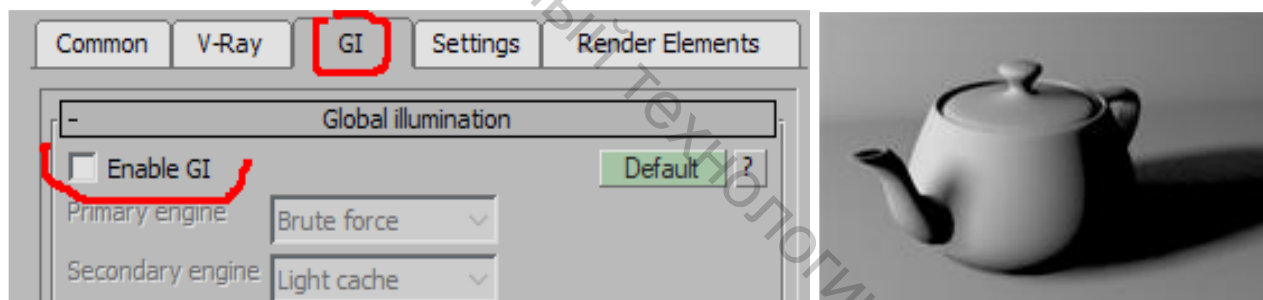


Рисунок 5 – Прямое освещение

Для понимания настроек непрямого освещения Indirect Illumination V-Ray и любого другого рендер-движка стоит разобраться, что же такое непрямоe освещение в принципе и почему оно оказывается важным в фотореалистичной визуализации.

Расчет отраженного диффузного освещения является одним из важнейших алгоритмов V-Ray и тоже достаточно независим от расчета других компонентов освещенности. Рассмотрим влияние переотраженного света и понятия:

- **прямое освещение;**
- **непрямое освещение** (переотраженный свет).

Прямое освещение – это, непосредственно, свет от источников света. Его вклад представлен на рисунках 5 и 6.

Непрямое освещение – это освещение объектов лишь диффузно

отраженным светом от других объектов, без участия прямого света от непосредственных источников света.

Светлые объекты попросту отражают падающие на него, от источника света, лучи. А они, в свою очередь, отражаясь, попадают и освещают неосвещенную сторону другого объекта. Из-за своего происхождения, то есть из-за того, что эти лучи – отраженные от других объектов, а не летящие непосредственно от источника света, их называют вторичными или непрямыми. Соответственно освещение, которое создается этими лучами, называется непрямым, т. е. Indirect Illumination.



Рисунок 6 – Прямое освещение



Рисунок 7 – Непрямое освещение

Далее подробно будем рассматривать только расчет непрямого диффузного освещения.

3.3 Трассировка лучей (Raytracing)

V-Ray – это рендерер.

Трассировка лучей (англ. Raytracing) – один из методов геометрической оптики, исследование оптических систем путем отслеживания взаимодействия отдельных лучей с поверхностями [7]. В узком смысле – технология построения изображения трехмерных моделей в компьютерных программах, при которых **отслеживается обратная траектория распространения луча** (от экрана к источнику (рис. 8)).

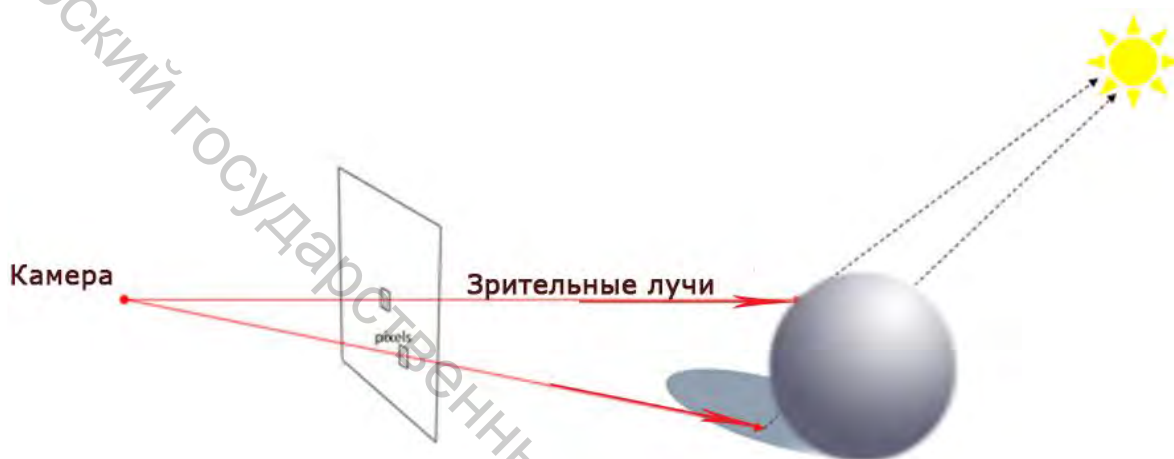


Рисунок 8 – Метод обратной трассировки

Таким образом, при обратной трассировке рассчитывается освещенность только тех объектов, которые видны из камеры. Чтобы определить, какие объекты просчитывать, а какие – нет, визуализация начинается с того, что лучи (rays) сначала направляются из камеры в сцену, чтобы собрать информацию о геометрии, которая будет видна в окончательном изображении. Лучи, которые исходят из камеры называются **первичные лучи** (Primary Samples, также их называют Camera Rays или Eye Rays) и управляются они с помощью **V-Ray Sampler Image** (также известный как Anti-Aliasing или AA).

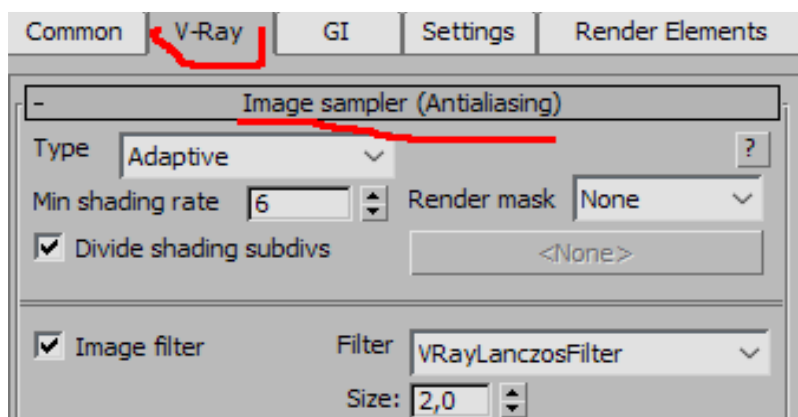


Рисунок 9 – V-Ray Sampler Image

Всякий раз, когда первичный луч пересекается с геометрией в сцене, дополнительные лучи будут посланы от этой точки пересечения в остальную часть сцены, чтобы собрать информацию о таких параметрах, как Shadows (тени), Lighting (освещение), Global Illumination (глобальное освещение), Reflection (отражение), Refraction (преломление), Sub-surface Scattering (SSS) (подповерхностное рассеивание) и т. д. Эти дополнительные лучи называются **вторичными лучами** и контролируются *V-Ray DMC Sampler*.

Первичный луч от источника прямого света, показанный на рисунке 10 жирной стрелкой, бьется о поверхность, после чего формируются вторичные лучи, показанные тонкими стрелками.



Рисунок 10 – Первичный луч от источника прямого света

Эти вторичные лучи исходят от поверхности диффузно, т. е. разлетаясь множеством лучей от нее во все стороны. В свою очередь, эти уже отраженные лучи, ударяясь о другие встречающиеся им поверхности, таким же образом снова формируют множество лучей каждый, и так дальше и дальше [8]. Луч вторичного освещения может отскакивать большое количество раз, в зависимости от количества вторичных отскоков, определяемого параметрами рендерера. Опираясь на изображение выше, не составляет труда представить, что они образуют полусферу, похожую на ежа. Векторы лучей расходятся на все 180 градусов в разные стороны, то есть по полусфере от точки отражения. Именно так ведут себя лучи в GI движке и именно из-за формы отражения их еще называют Hemispheric Rays, т. е. полусферическими лучами. Если объект имеет какую-то степень прозрачности – лучи расходятся по сфере (рис. 4).

Основной целью лучей (далее лучи – сэмплы) является сбор информации об освещенности со всей сцены для конечного просчета. Для того чтобы

выяснить, что происходит в сцене, нужно собрать как можно больше сэмплов как первичных, так и вторичных. Чем больше сцена их собирает, тем больше информации у V-Ray и тем меньше шума будет на финальной визуализации. Шум всегда вызван недостатком информации.

Отскоки лучей вторичного освещения в V-Ray разделяют на основные (первый отскок), названные **Primary bounces**, и дополнительные (все последующие), названные **Secondary bounces**.

Первый по счету отскок луча от объекта сцены неспроста назван Primary bounce, т. е. главным отскоком. Рендер-движок, имитирующий поведение реального света, устанавливает затухание лучам по мере их переотражения. И, как это случается в реальном мире, первый отскок несет большую долю отраженной световой энергии. Причем энергия этого первого настолько значительная, что в разы превышает световую энергию всех последующих отражений вместе взятых [9].



Рисунок 11 – Переотражение лучей

3.4 Настройка GI

Вкладка Indirect Illumination находится в окне Render Scene (F10). В данной вкладке главным управляющим свитком является GI/Global Illumination (GI). В нем осуществляется включение глобального освещения, выбор алгоритмов просчета главных и дополнительных отскоков и общие настройки GI.

Прежде всего, для активации глобального освещения в сцене следует установить галочку в чекбокс On. После этого станут доступны параметры глобального освещения V-Ray движка.

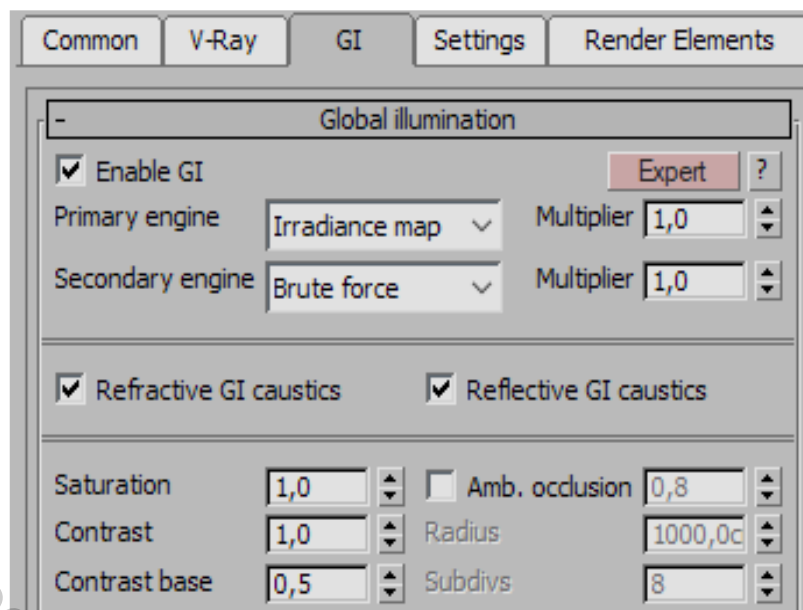


Рисунок 12 – Свиток GI/Global Illumination

В этих настройках, если все и не сильно просто, то, как минимум, все достаточно логично. Как было описано выше, есть основные, есть дополнительные отскоки луча, называемые **Primary** и **Secondary bounces**. Просчитываются одни и другие с помощью специальных алгоритмов – GI engine (**Primary engine** и **Secondary engine**), на которых остановимся позже.

Числовой параметр **Multiplier** – величина применения рендер-элемента данного светового отскока GI при формировании финального изображения. По умолчанию, для Primary и Secondary bounces значения Multiplier установлены равные единице. Однако это не самые удачные значения, потому что в виду излишней яркости лучей GI по сравнению с яркостью лучей от источников прямого света получают сильно пересветленные и неконтрастные изображения. Часто это сказывается еще и на тенях, которые становятся более светлыми, что на визуализации сразу вызывает эффект «летающих» объектов, под которыми нет привычной тени (рис.13).



Рисунок 13 – Влияние числового параметра **Multiplier**

В опциях **GI caustics** можно включить или отключить так называемые рефлексивные и рефрактивные каустики для GI лучей.

Каустика – это геометрическое место, где сходятся отраженные или преломленные лучи от отражающих и прозрачных криволинейных поверхностей, вырисовывающие близи них специфические, легко заметные световые узоры [10]. Другими словами, это эффект искривленной линзы, проецирующей световые лучи на какую-нибудь поверхность, где линзами выступают отражающие и прозрачные объекты сцены. Самый банальный пример каустики – причудливый световой узор на столе от бокала, на который сбоку светит свет.

Расчет каустик в V-Ray сопровождается появлением большого количества артефактов, чрезмерным потреблением оперативной памяти и сложной предварительной настройкой. Все это – при минимально заметном, а то и отсутствующем эффекте в подавляющем большинстве сцен. В универсальных стартовых настройках V-Ray просчет каустик включать не следует.

Также следует понимать, что расчет каустик и GI каустик – это разные вещи. Каустики, созданные вышеупомянутой фотонной картой, рассчитываются лишь для первичных лучей от источника света. В то время как GI каустики просчитываются отдельно и настройки опции GI caustics свитка V-Ray/ Indirect Illumination (GI) отвечают как раз за эти каустики, произошедшие от вторичного освещения. Мало того, GI каустики названы каустиками лишь по некой аналогии с фотонными, на самом деле они таковыми не являются.

Опция **Reflective** позволяет включать просчет рефлексивных каустик для GI отскоков, что позволяет GI лучам учитывать отражательные свойства материалов и сильнее отражаться от отражающих объектов, таких как хромированные детали, зеркала и прочее.

Учитывая меньшую энергию лучей вторичных отскоков, по сравнению с лучами от источников прямого света, влияние рефлексивных каустик на общую освещенность сцены будет незначительное. Кроме того, в виду необходимости просчета Reflective caustics рендерером, их включение увеличит время просчета вторичного освещения, а также в виду сложности сэмплирования этого эффекта, возможно появление нежелательного шума в GI. Учитывая данные недостатки, эту опцию следует номинально оставлять выключенной в универсальных стартовых настройках V-Ray.

Опция **Refractive** позволяет включать просчет рефрактивных каустик для GI отскоков. Несмотря на схожее название с каустиками от прямых лучей источников света, где имитируются именно каустические световые эффекты, Refractive caustics просто позволяет проходить вторичному освещению через прозрачные поверхности. Например, это применимо в интерьерной сцене, которая освещается вторичным освещением от VRaySky, Skylight, Skylight Portal, находящимся за окном, а в оконных проемах стоят модели окон со стеклом. Отсутствие просчета Refractive caustics в этой сцене сделает весь рендер черным, просто потому что GI лучи не попадут внутрь интерьера, так как не смогут пройти даже через прозрачное оконное стекло.

Для большинства сцен такое поведение GI неприемлемо и поэтому опцию **Refractive** в универсальных настройках V-Ray следует активировать, установив чекбокс **Refractive**.

Область **Post-processing** содержит три параметра.

Saturation – очень важный параметр GI, который отвечает за цветовую насыщенность отраженных лучей (рис.14).



Рисунок 14 – Влияние параметра **Saturation**

Contrast и **Contrast base** – параметры, которые управляют контрастом GI, используемым рендерером при формировании финального изображения, аналогично как **Saturation** по сути всего лишь управляет цветовой насыщенностью GI. Параметр **Contrast** работает совместно с параметром **Contrast base**. Технически **Contrast base** – это одна из переменных алгоритма, вычисляющего контраст рендер-элемента GI, смещая диапазон его значений и тем самым влияя на его яркость. При базе **Contrast base** равной нулю, увеличение значения параметра **Contrast** осветляет GI; при базе **Contrast base** большей нуля, увеличивается общая яркость, и увеличение значения параметра **Contrast** приводит к затемнению GI.

Эти параметры можно рассматривать как аналог инструмента **Brightness/Contrast** растрового редактора **Photoshop**. В подавляющем

большинстве ситуаций совершенно нет никакой необходимости как-либо изменять контрастность GI. Поэтому в универсальных стартовых настройках V-Ray их значения не стоит изменять, оставив 1 для Contrast и 0,5 для Contrast base, как это выставлено по умолчанию.

Всего в V-Ray существует четыре алгоритма просчета отскоков GI:

Irradiance Map,
Brute Force (QMC),
Photon Map,
Light Cache.

3.5 Алгоритм просчета GI Irradiance map

Irradiance map – это адаптивный алгоритм просчета отскоков глобального освещения. Основная особенность его работы заключается в выявлении наиболее значимых детализированных зон визуализируемой сцены, вычислении в них GI и игнорировании менее важных зон, с последующим заполнением информации о GI в них путем интерполяции информации из уже просчитанных важных зон [11].

Как именно работает и что из себя представляет алгоритм Irradiance map, можно понять по карте, которая называется аналогично – Irradiance map.



Рисунок 15 – Сэмплирование в Irradiance map

Светлыми точками представлены сэмплы. Именно совокупностью этих точек и является карта Irradiance map, которую мы и наблюдаем на изображении. Эти зоны имеют наибольшую важность в получении красивого детализированного вторичного освещения.

Данные точки содержат информацию о цвете и яркости GI в этих зонах. Остальная часть изображения заполнена серым цветом. Серые зоны на Irradiance map – это части сцены, где Irradiance map не содержит информацию о GI. Однако это полностью компенсируется интерполяцией между уже вычисленными точками Irradiance map, т. е. Irradiance map вычисляет наиболее важные зоны, просчитывая лишь их, после чего недостающую информацию о не просчитанных зонах попросту компенсирует интерполяцией, взяв данные о цвете и яркости из уже имеющихся.

Алгоритм Irradiance map строит трехмерную карту точек, содержащих информацию о GI точно на поверхности объектов сцены. В объеме карта Irradiance map этой сцены выглядит следующим образом:



Рисунок 16 – Трехмерная карта точек

На скриншотах специальной утилиты **Irradiance map viewer**, представленных на рисунке 16, виден не только факт трехмерности Irradiance map, но и ее зависимость от вида из камеры, через которую происходит визуализация сцены. Это отсутствие сэмплов на невидимых из камеры частях трехмерных моделей.

В Irradiance map определение важных мест сцены, в которых ведется

просчет GI, реализовано на использовании концепции субсэмплирования и суперсэмплирования.

При **субсэмплировании** для определения цвета нескольких пикселей берется всего одна точка с фигуры в их усредненном геометрическом центре.

Принцип работы **суперсэмплирования** заключается в том, что для определения цвета пикселя берется не один образец цвета из центра, а пиксель разбивается на несколько частей, так называемых субпикселей, и образы цветов берутся из них. Таким образом, получается не просто цвет фигуры или фона, а конкретный суммарный цвет всех цветов субпикселей.

Для начала GI всей сцены просчитывается не адаптивно в минимальном разрешении, которое обычно меньше разрешения рендеринга. После чего из полученных данных выбираются наиболее важные, то есть вычисляются те самые зоны, где следует вести более точный просчет. И начинается просчет в следующем по величине разрешении, но лишь в требуемых зонах. Эта процедура поэтапно повторяется несколько раз, каждый раз увеличивая разрешение, и так, пока не достигнет максимально установленного разрешения просчета Irradiance map. Минимальное и максимальное разрешение просчета Irradiance map указывается в ее параметрах [12].

Шаг просчетов отличается вдвое большим или вдвое меньшим значением разрешения, то есть в четыре раза по площади изображения в пикселях. То есть карта Irradiance map может быть просчитана как на меньших, чем разрешение рендеринга, так и на больших разрешениях.

Каждая фаза просчета GI называется pass, т. е. проход. Фазы просчетов, с помощью которых происходит определение важных зон, называются Prepass, т. е. предпроход. Прогресс просчета предпроходов, их общее количество и то, какой предпроход вычисляется в данный момент можно наблюдать в системном окне Rendering, появляющемся после старта рендеринга.

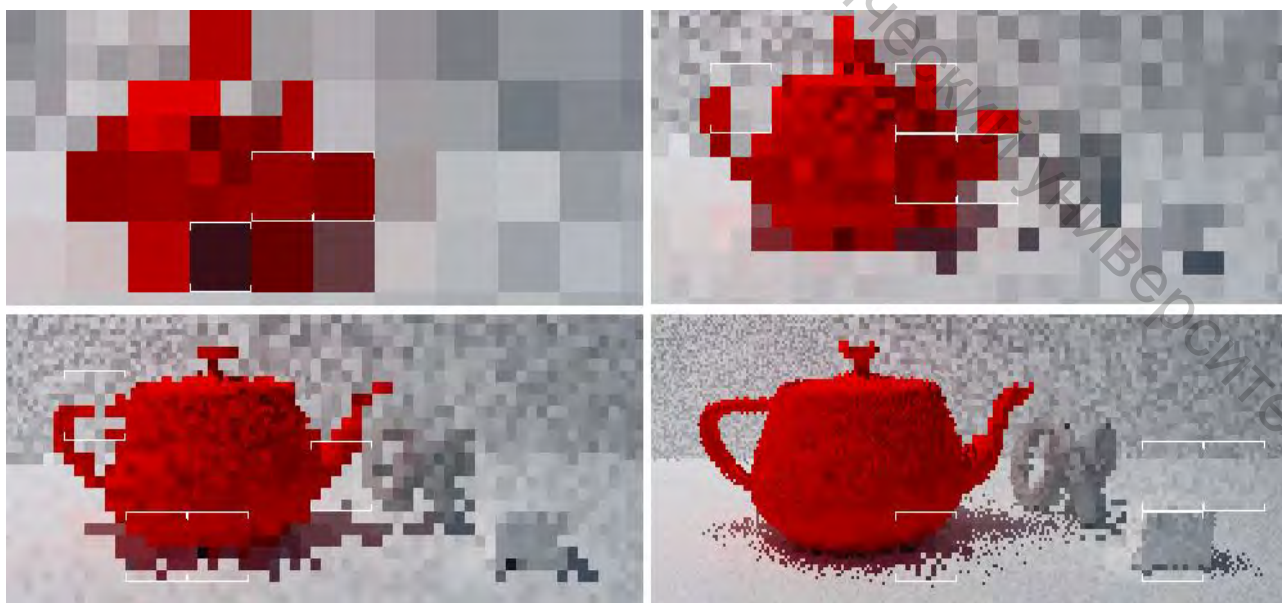


Рисунок 17 – Прогресс просчета предпроходов

За то, какие зоны алгоритм Irradiance map будет считать важными, какое количество вторичных лучей будет принимать каждая точка, а также в каких разрешениях производить определение важных деталей, отвечают параметры Irradiance map, расположенные в свитке GI/Irradiance map.

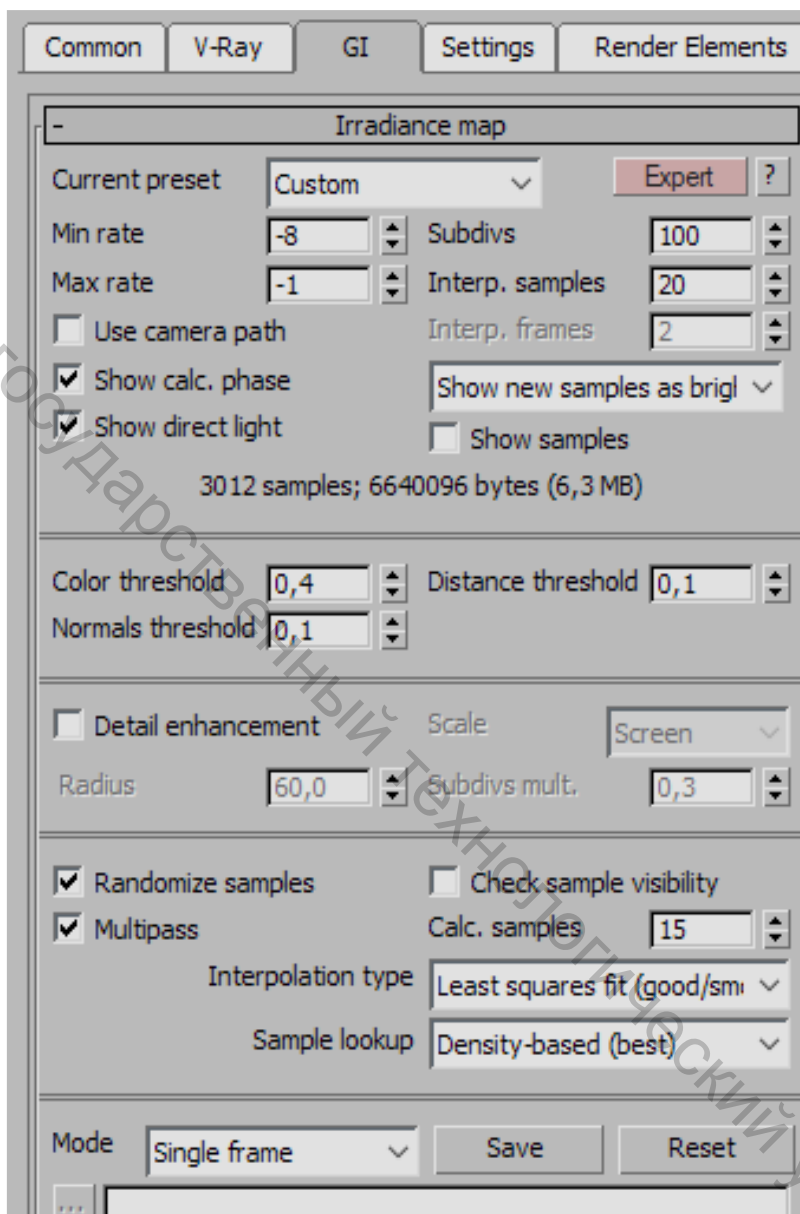


Рисунок 18 – Свиток GI/Irradiance map

Current preset – быстрые наборы предустановленных значений для основных параметров Irradiance map. Назначение каждого из них можно понять по их названиям: Very low, Low, Medium, Medium animation, High, High animation, Very high. Однако наибольший интерес представляет значение Custom. Как видно из его названия, это значение позволяет вручную настраивать параметры.

Следующие параметры по сути определяют число первичных лучей из камеры:

Min rate – значение, определяющее разрешение первого предпрохода, рассчитывающего GI.

Max rate – значение, определяющее финальное разрешение просчета GI или разрешение последнего предпрохода.

Значение rate, равное 0, означает, что просчет GI будет вестись в разрешении, равном разрешению финального рендера. Значение rate, равное 1, означает, что расчет GI будет вестись во вдвое меньшем разрешении, чем разрешение финального изображения. Значение rate, равное 2, означает, что расчет GI будет идти вчетверо меньше финального изображения. По аналогии, значение rate, равное 1, будет означать, что расчет GI будет вестись во вдвое большем разрешении, чем разрешение финального изображения и т. д.

Параметр **Subdivs** (HemiSphere Subdivisions, субделения полусферы) – количество лучей GI (квадратный корень из фактического числа лучей, например, 8 Subdivs = 64 луча), которые будут отражены от точки на поверхности объекта. Это параметр, определяющий количество диффузных лучей вторичного освещения принимаемых каждой точкой Irradiance map, как говорилось ранее – в виде полусферы. В подавляющем большинстве случаев значения Subdivs, равное 80, будет вполне достаточно.

Interp. Samples – параметр, определяющий качество интерполяции. Interp. samples определяет качество интерполяции не важных пропущенных зон из информации, имеющейся в важных просчитываемых зонах. Чем выше значение параметра Interp. samples, тем выше интерполяция, вследствие чего GI более размытое, в то время как при низких значениях Interp. samples GI карта менее «замыленная», однако и более шумная с большим количеством артефактов. Установленного по умолчанию значения в 20 сэмплов вполне достаточно.

Как уже отмечалось, Irradiance map с помощью предпроходов прощупывает сцену с целью выявления ее важных деталей и делает это на основании информации предыдущих предпроходов, руководствуясь тремя основными критичными в плане деталей свойствами. Это цвет, разница нормалей и взаимное расположение объектов. В Irradiance map эти параметры представлены параметрами Clr thresh, Nrm thresh и Dist thresh соответственно.

Clr thresh – определяет порог чувствительности алгоритма Irradiance map к цветам сцены. Чем выше этот порог, тем меньшие цветовые изменения на объектах сцены будут учтены как важные зоны и тем легче и быстрее будет проходить расчет. Снижение порога чувствительности сделает алгоритм Irradiance map более чувствительным к цветовым изменениям в сцене и позволит создавать более детализированную карту освещенности, а значит – и более качественные изображения. Значение параметра Clr thresh, равное 0,3, вполне достаточно.

Nrm thresh – определяет порог чувствительности алгоритма к геометрии сцены, в частности к интенсивности изменения нормалей объектов в ней. Чем ниже этот порог, тем больше геометрических деталей на объектах сцены будут учтены как важные зоны, соответственно качественнее будет финальное

изображение. Значение параметра **Nrm thresh** равное 0,1 вполне хватает для создания нормальной карты.

Dist thresh – определяет порог чувствительности алгоритма к взаимному расположению поверхностей в сцене. Чем выше этот порог, тем дальше находящиеся друг от друга поверхности объектов будут учтены как важные зоны, соответственно и качественнее будет финальное изображение. Следует учесть, что при уменьшении порога **Dist thresh**, будут учитываться меньшие дистанции между поверхностями объектов, тем самым уменьшая плотность сэмплов на карте освещенности, а значит и качество финального изображения. В отличие от **Clr** и **Nrm thresh**, где уменьшение порога наоборот улучшает его. Значение параметра **Dist thresh**, равное 0,1, вполне достаточно для создания хорошей карты.

Неактивный параметр **Interp. frames** отвечает за то, какое количество карт освещенности, просчитанных для разных кадров анимации в режиме **Animation (prepass)**, V-Ray будет использовать для просчета текущего кадра в режиме **Animation (rendering)**. Именно в последнем случае этот параметр будет активным и доступен к изменению.

Свиток **Irradiance map** содержит опции, отвечающие за визуализацию **Irradiance map** во фрейм буфере. Это позволит наглядно наблюдать за фазами просчета предпроходов IM, показывать прямой свет совместно с GI на визуализации предпроходов, а также позволит наблюдать за расположением сэмплов в сцене. За эти возможности отвечают функции **Show calc phase**, **Show direct light** и **Show samples** соответственно. Это тестовые функции, не влияющие на просчет **Irradiance map**.

Опция **Detail enhancement** позволяет создать гибрид между адаптивным, но создающим несколько размытую карту сэмплов алгоритмом **Irradiance map** и грубым, но прецизионным в деталях **Brute Force**.

Когда функция **Detail enhancement** активирована, **Irradiance map**, так же, как и раньше, с помощью предпроходов вычисляет детализацию сцены. Однако на последнем калькуляционном проходе, когда конечная информация о детализации сцены получена, начинает работать **Detail enhancement**. Суть работы **Detail enhancement** в том, что, основываясь на информации о детализации сцены, сэмплы GI наиболее важных зон просчитываются алгоритмом **Brute Force** на полном разрешении финальной визуализации, без субсэмплирования. Все остальные зоны, как и в штатном режиме работы IM, вычисляются с помощью интерполяции [13].

Стоит отметить, что при активированной функции **Detail enhancement**, алгоритм **Irradiance map** используется лишь для определения детализации сцены, а непосредственный просчет важных деталей ложится на **Brute Force**, составляющую **Detail enhancement**. То есть, при активной опции **Detail enhancement** можно понижать все настройки алгоритма **Irradiance map**, кроме **Interp. Samples**, который можно даже увеличить.

Активируется алгоритм путем установки галочки в чекбокс **On**. Тогда становятся активными его параметры **Scale**, **Radius** и **Subdivs mult**.

Scale – определяет единицы измерения параметра Radius.

Radius – величина радиуса полусферы, определяющей зону, в которой Brute Force будет просчитывать сэмплы GI относительно исходной точки. Расположение исходных точек применения Brute Force, как уже было сказано выше, определяется на стадии вычисления детализации сцены с помощью предпроходов.

Subdivs mult – определяет количество диффузных лучей. Указывается в процентном отношении от установленного в HSph. subdivs значения.

Interpolation type – выпадающий список, позволяющий выбрать метод интерполяции сэмплов, то есть алгоритм, который заполняет пропущенные зоны на карте, получая недостающие сэмплы из уже просчитанных. Всего в этом списке четыре метода, технически отличающиеся разными математическими принципами для вычисления результата. Совершенно нет никакой необходимости углубляться в принципы их работы. Оптимальный метод интерполяции – установленный по умолчанию Least squares fit.

Sample lookup – выпадающий список, позволяющий выбрать метод, который определяет, какие именно из просчитанных сэмплов будут участвовать в интерполяции пропущенных областей в соседних зонах. Всего четыре метода выбора сэмплов. Это – также алгоритмы, имеющие свои математические особенности в методах выбора нужных сэмплов, в которые не стоит вникать без особой необходимости.

Calc. pass interpolation samples – значение, определяющее число вычисленных сэмплов, которыми будет руководствоваться алгоритм Irradiance map при интерполяции во время последнего калькуляционного прохода. Установленного значения в 10 сэмплов вполне достаточно для получения качественного результата.

Multipass – функция, которая позволяет при просчете текущего предпрохода Irradiance map использовать абсолютно все доступные сэмплы, просчитанные на текущий момент. То есть и сэмплы от предыдущего предпрохода на более низком разрешении, и сэмплы, уже вычисленные в текущем предпроходе.

Irradiance map просчитывается, как и финальный рендер, во много потоков для полноценного использования потенциала современных вычислительных систем, которые зачастую многопоточны благодаря многопроцессорности и многоядерности. Изначально, при просчете Irradiance map многопоточной системой, каждая порция карты рассчитывается отдельным потоком, независимо друг от друга. Данные для текущего просчета берутся лишь из предыдущего предпрохода. Из-за этого получается, что каждая порция просчета, не имея информации об окружающих порциях, выдает несогласованный с соседними результат. На просчитанной таким образом карте проявляется решетка границ этих порций.

Randomize samples – функция, позволяющая генерировать сэмплы в случайном порядке, делая карту более реалистичной. Номинально алгоритм Irradiance map выбирает сэмплы на объектах сцены с регулярной

периодичностью, напоминающей сетку, даже если вышерассмотренная опция Multipass включена. Это наглядно можно посмотреть на рисунке 19.



Рисунок 19 – Регулярность сэмплов при отключенной функции Randomize samples

Чтобы устранить эту регулярность, явно выдающую компьютерное происхождение изображения и мешающую фотореалистичности визуализаций, функцию Randomize samples следует обязательно держать активированной.

Check sample visibility – функция, которая позволяет интерполировать сэмплы, лишь находящиеся в прямой видимости друг от друга. Очень часто, вследствие небольшой плотности ИМ, при интерполяции недостающих сэмплов, алгоритм ИМ берет соседние сэмплы, которые попадают в зону выборки сэмплов для данной точки, даже несмотря на то, что они могут находиться вне зоны прямой видимости относительно друг друга в сцене. Например, у нас есть книжный шкаф с тонкой деревянной полочкой, плотно прилегающей к задней стенке шкафа. На полочке сверху стоит светильник и освещает ее верхнюю часть. Под полочку прямые лучи света светильника никаким образом не попадают. Тем не менее, внизу под полочкой на задней стенке запросто могут появиться светлые пятна, несмотря на то, что их там просто не может быть. Это происходит, потому что алгоритм Irradiance map при интерполяции недостающих сэмплов в зоне под полочкой берет расположенные рядом сэмплы с ее верхней части без учета геометрии полочки. Чтобы этого не происходило, алгоритму ИМ при интерполяции необходимо указать, что нужно проверять, какие сэмплы находятся в зоне прямой взаимной видимости, а какие – нет. Разумеется, такая проверка потребует дополнительных вычислительных ресурсов.

В зонах **Mode** и **On render end** задается режим работы Irradiance map, нужный для рендеринга анимации.

3.6 Алгоритм просчета GI Brute Force (QMC)

Brute force (грубая сила) – базовый алгоритм, устанавливающий фиксированное количество вторичных лучей, отраженных от точки в сцене после попадания в нее первичного луча от источника света. На чем большее количество отраженных диффузных лучей вторичного освещения будет разбит луч первичного света, тем больше деталей и меньше шума будет на финальном изображении.

Алгоритм Brute force не имеет адаптивности и попросту просчитывает GI для каждого пикселя финального изображения сцены, в независимости от ее сложности, цветов и детализации объектов. Тем самым он расходует одинаковое количество вычислительных ресурсов как в заметных, так и в маловажных участках сцены. Настройки этого алгоритма представлены на рисунке 20.

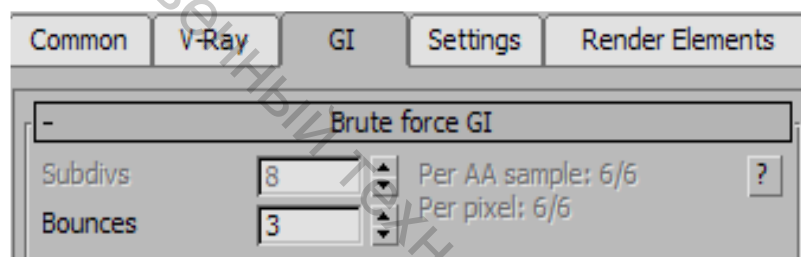


Рисунок 20 – Настройки алгоритма Brute force

Subdivs – параметр, определяющий количество вторичных лучей, на которое будет разбит каждый луч от первичного источника света, попавший на объект сцены. Числовое значение параметра Subdivs, как и в случае с Irradiance map, реальное число лучей будет равняться квадрату этого числа.

Secondary bounces – величина, задающая, какое количество раз будут отражаться лучи дополнительных отскоков. Он активен, только если Brute force алгоритм установлен как алгоритм просчета Secondary bounces в выпадающем списке GI engine, так как именно в этом случае он будет просчитывать нужное количество переотражений. Если Brute force установлен как Primary GI engine, то Secondary bounces параметр будет неактивным [14].

Алгоритм весьма иррационален, его использование приводит к большим срокам расчета сложных сцен при адекватном уровне качества. Его использование оправдано лишь в особых ситуациях, когда остальные алгоритмы из-за своей адаптивности не могут справиться с просчетом мелких деталей в сцене.

3.7 Алгоритм просчета GI Photon Map

Photon map – это устаревший рендер-движок, основная концепция работы которого заключается в том, что источник света бомбардирует все объекты сцены исходящими из него порциями энергии – фотонами. После чего формируется так называемая фотонная карта, то есть трехмерная карта точек соприкосновения фотонов с поверхностями объектов сцены, в каждой точке которой хранится информация о яркости и цвете вторичного освещения.

Этот алгоритм засыпает фотонами сцену даже в тех местах, которые не видны из данного ракурса и даже те, которые возможно никогда не будут видны из камеры в случае анимации. В алгоритме Photon map не предусмотрена достаточная адаптивность по отношению к мелким деталям сцены, следствием чего является плохая их проработка.

Вследствие набора ограничений и иррациональностей, устаревший алгоритм GI Photon map не следует использовать в V-Ray вообще. Он был популярен в ранних версиях V-Ray и использовался как движок для Secondary bounces до момента появления намного более продвинутого и уместного алгоритма Light cache.

3.8 Алгоритм просчета GI Light cache

Light cache выпускает множество лучей из камеры, формируя карту глобального освещения в точках пересечения этих лучей с объектами данной сцены. Данные лучи не являются лучами глобального освещения. Это пути, направленные из точки обозревания на сцену. Луч Light cache, попадая на объект сцены, вычисляет в точке попадания цвет, учитывая свойства материалов, и попадающее на нее освещение от источников прямого света или вторичного освещения.

Light cache считает лишь те зоны в сцене, которые видны из камеры. Он хорошо справляется с мелкими деталями, способен работать со всеми источниками света.

Subdivs – параметр, определяющий количество лучей Light cache, трассирующихся из камеры в сторону сцены для определения цветов сэмплов.

Алгоритм Light cache условно разделяет сцену на ячейки с помощью попавших в нее лучей Light cache, испускаемых из камеры. То есть каждая ячейка и есть неделимый цветовой образец, который будет использован в конечной карте GI. Чем больше количество лучей, определяемое значением параметра Subdivs, тем большим их числом будет просчитана одна ячейка и тем более точно будет определен ее цвет.

В стартовых настройках V-Ray можно использовать установленное значение в 1000 Subdivs и повышать его лишь при необходимости. Значения Subdivs в 2000–3000 более чем достаточно для финального рендеринга фотореалистичных изображений. Ориентироваться на нужное значение этого параметра для финального рендера необходимо, исходя из разрешения изображения. Ориентировочно это число должно быть примерно равно числу пикселей по большей стороне, умноженному на 1,5 [15].

Sample size – размер ячеек-образцов (samples). Когда размер ячейки задан, то алгоритм Light cache делит на них все изображение. После чего трассирующие лучи определяют цвет каждой ячейки. Для определения цвета ячейки необходим как минимум один луч. Если общее количество лучей, заданных параметром Subdivs, меньше общего количества ячеек. Тогда алгоритм Light cache будет вынужден окрасить некоторые из рядом стоящих ячеек одним цветом, что визуально будет выглядеть как увеличение размера некоторых ячеек.

Описанные параметры представлены на рисунке 21.

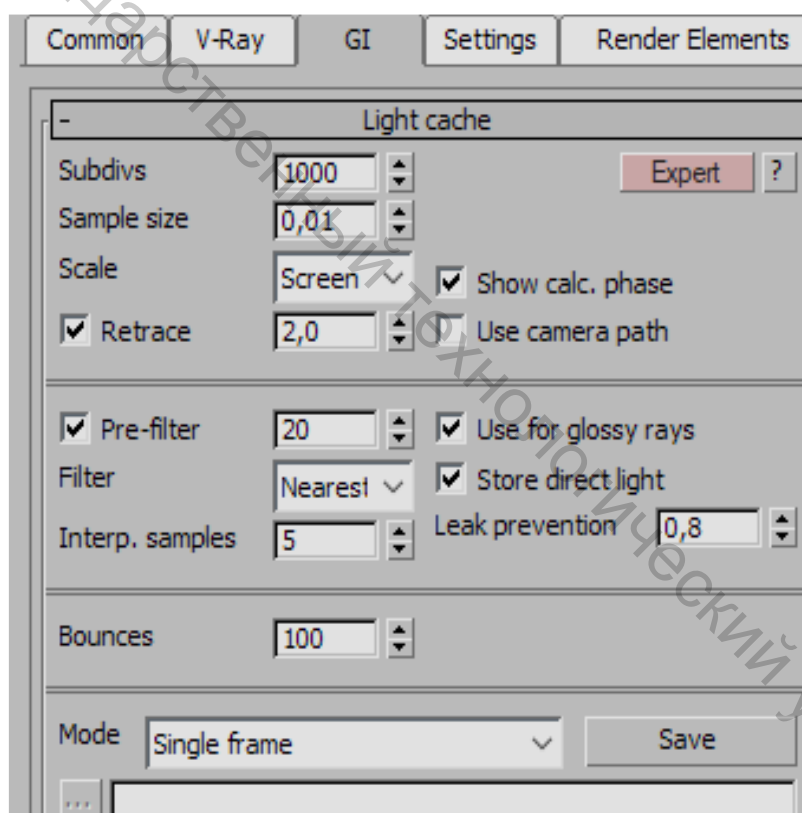


Рисунок 21 – Настройки алгоритма Light cache

Можно сказать, что визуальный размер ячеек косвенно зависит и от значения параметра Subdivs. Размер, заданный значением Sample size, будет достигнут только если в сцене будет достаточное количество лучей Light cache. В случае, если количества лучей окажется больше количества ячеек, то некоторые ячейки будут просчитаны несколькими лучами и будут иметь более точный цвет, что улучшит итоговое качество визуализации.



Subdivs - 500
Sample size - 0,05



Subdivs - 500
Sample size - 0,04



Subdivs - 500
Sample size - 0,01



Subdivs - 1000
Sample size - 0,01

Рисунок 22 – Сэмплирование в Light cache

Менять размер ячеек следует лишь, если в сцене находится большое количество мелких деталей и карта Light cache недостаточно детализирована для их верной передачи. В большинстве случаев номинально установленное значение Sample size, равное 0,02, вполне достаточно для большинства сцен.

Scale – параметр в виде выпадающего списка, позволяющий задать единицы измерения размера ячеек. Всего в выпадающем списке два значения, это Screen и World.

Значение **Screen** позволяет указывать размер ячеек в дробном соотношении от финального изображения. Например, если Sample size равен 0.02, а Scale выставлено в Screen, то это означает, что каждая ячейка будет занимать 2 % от размера всего изображения. Размер ячейки, заданный относительно экрана, не зависит от размера изображения в пикселях и от геометрии сцены. То есть в сцене, представленной на изображении выше, размер ячеек на чайнике и на стене за ним будут иметь одинаковый размер, несмотря на то, что геометрически эти поверхности находятся на разном расстоянии от камеры.

Значение **World** заставляет использовать для выбора размера ячеек системные единицы 3ds Max, установленные для данной сцены в System Unit Setup. При таком подходе размеры ячеек геометрически будут одинаковы, но визуально их размер на данном виде из камеры будет зависеть от геометрии сцены. В таком случае в сцене, представленной на изображении ранее, размеры ячеек на чайнике визуально будут меньше, чем размеры ячеек на фронтальной грани куба. Также, учитывая, что Light cache сэмплирует каждый участок изображения фиксированным количеством лучей, то на меньшую по размеру ячейку попадет меньшее количество лучей, по сравнению с большей. Это ухудшит качество сэмплирования меньших по размеру ячеек. На финальной визуализации это может проявиться как шум или артефакты на мелких деталях сцены.

Возможность задавания ячеек сэмплирования Light cache в системных единицах сцены создана с целью получения однородных карт Light cache для последующего сохранения на жесткий диск и использования в fly-through анимациях, т. е. анимациях облета. Такая техника применяется для того, чтобы иметь возможность просчета Light cache для анимации и при этом размеры ячеек Light cache на всех кадрах были одинаковы относительно геометрии сцены, не зависимо от положения камеры.

Для рендеринга статических сцен, а также динамических анимаций, когда карта Light cache для каждого кадра просчитывается индивидуально и не может быть использована для совокупности кадров, необходимо использовать режим Screen. Это позволит уменьшить количество нежелательных артефактов.

Store direct light – функция, позволяющая алгоритму Light cache сохранять не только информацию о GI, но и информацию о прямом свете сцены. Как уже было описано, Light cache трассирует лучи из камеры на объекты сцены. После пересечения луча с объектом сцены, алгоритм Light cache проверяет шейдер объекта для определения его цвета и продолжает трассировку луча от этой точки к источнику света, освещающему ее для определения яркости. То есть алгоритм Light cache вынужден проверять и прямой свет, попадающий на данную точку. Включение опции Store direct light позволяет кэшировать эту информацию о прямом свете в сцене. После того как прочитан рендер-элемент GI, V-Ray отдельно просчитывает прямое освещение сцены. Сэмплирование прямого света сцены, безусловно, требует вычислительных ресурсов и занимает определенное время. Кэширование света, то есть предварительное сохранение информации о нем из Light cache и последующее использование готовой информации значительно ускоряет финальный рендеринг, избавляя алгоритм просчета прямого света от вычислительных затрат. Качество полученного таким образом прямого света будет значительно хуже, чем при традиционном сэмплировании и будет напрямую зависеть от качества самой Light cache. Такой подход оправдан лишь в очень перегруженных сценах с огромным количеством источников света, прямой расчет света от которых займет долгие часы. Просчитанные таким образом изображения будут иметь существенные недостатки, явно выдавая компьютерное происхождение полученных

изображений. Они будут выходить неживыми с размыленными и нечеткими тенями. В фотореалистичном рендеринге такой подход не оправдан, даже с учетом существенной экономии вычислительных ресурсов. В большинстве случаев V-Ray функцию Store direct light следует выключать.

Show calc. phase – функция, позволяющая отображать информативное превью просчитываемого изображения уже в первые минуты рендера. Это позволяет избавиться от большого количества тестовых рендеров на низких разрешениях.

Следующие параметры отвечают за постобработку уже просчитанной карты Light cache при финальном рендеринге изображения.

Pre-filter – функция, отвечающая за интерполяцию сэмплов, ячеек-образцов Light cache между собой. Pre-filter интерполирует имеющуюся карту сэмплов для получения более сглаженного результата. И только после этого передает ее далее для просчета финального изображения. Степень интерполяции определяется числовым значением рядом с Pre-filter. Большие значения дают менее шумный, но более размытый результат, что неуклонно ведет к исчезновению мелких деталей и некорректных теней. Небольшие значения Pre-filter дают детализированную карту Light cache, однако из-за слабого размытия ее сэмплов, она может оказаться весьма шумной.

Filter – выпадающий список, позволяющий выбрать дополнительный алгоритм интерполяции карты Light cache при финальном рендеринге. Если выбран один из фильтров (не путать с Pre-filter), то при просчете каждой точки шейдирования алгоритм финального рендеринга будет интерполировать информацию о сэмплах карты Light cache для просчета данной точки.

None – пункт выпадающего списка Filter, исключающий использование фильтра. В этом случае алгоритм финального рендеринга будет использовать карту Light cache в том виде, как она есть, без дополнительной фильтрации.

Nearest – фильтр, использующий для интерполяции данной точки в карте Light cache определенное количество ближайших сэмплов. Количество ближайших сэмплов, участвующих в фильтрации, определяется параметром **Interpolation samples**.

Fixed – фильтр, аналогичный фильтру Nearest, с той разницей, что количество сэмплов, участвующих в модификации данной точки на карте GI, определяется квадратной зоной с центром в этой точке. Величину зоны определяет параметр **Filter size**. Единицы измерения Filter size зависят от единиц величины сэмплов, заданных параметром **Scale**, описанном ранее.

Фильтр Fixed просчитывает и модифицирует лишь области, попавшие в зону действия Filter size. Они выглядят как квадраты с информацией о вторичном освещении сцены, в то время как остальные зоны он оставляет нетронутыми, которые выглядят как черный фон.



Sample size - 0,02
Filter size - 0,01

Рисунок 23. Филт Fixed (зона действия Filter size)

Для того чтобы избежать появления таких существенных артефактов на карте Light cache, следует всегда держать размер этих зон больше, чем размер самих сэмплов, примерно вдвое.

По принципу работы Pre-filter и Filter понятно, что они в результате выполняют одну и ту же функцию – размывают (интерполируют) карту Light cache, делая ее более однородной. Эти механизмы различаются лишь технической реализацией. Алгоритм Pre-filter интерполирует сразу все сэмплы для всей карты Light cache. Однако его недостатком является то, что это однопоточный процесс и вне зависимости от количества потоков, доступных системе, он будет рассчитываться лишь одним потоком. В то же время алгоритм Filter рассчитывается более точно, т. к. это происходит отдельно для каждой отдельной порции рендеринга. Такой подход занимает больше вычислительных ресурсов, заставляя V-Ray каждый раз рассчитывать интерполяцию для каждой порции рендеринга. Однако процесс работы Filter, как и финальный рендеринг, многопоточный и использует все доступные системе потоки, за счет чего вычислительная нагрузка распределяется. Для одноядерных и двухъядерных систем оправдано использование Pre-filter или же их комбинации в виде Pre-filter + Filter, где количество сэмплов интерполяции распределены пополам или даже с большим перевалом в сторону префильтрации. Большинство высокопроизводительных ПК способны обрабатывать по четыре и более физических или виртуальных потоков одновременно. Для них будет оправдано использование именно алгоритма Filter [15].

В стартовых настройках V-Ray следует использовать алгоритм Filter: Nearest с Interpolation samples, равным 10. Этот алгоритм и такая степень

интерполяции дадут хорошее не размытое детализированное глобальное освещение без видимых артефактов.

3.9 Выбор связки алгоритмов для просчета GI

Несмотря на кажущийся большой выбор, он оказывается весьма однозначен.

Фотонная карта отвергается сразу по описанным выше причинам. Самым универсальным, способным решить любые задачи, является, конечно же, алгоритм Brute Force. Однако отсутствие встроенной адаптивности и связанная с этим ресурсоемкость просто не оставляют ему шансов на широкое использование в качестве GI движка в сложных сценах. Лишь в некоторых из них, когда оптимизированные движки не могут справиться с появляющимися на визуализации артефактами, которые и возникают вследствие их оптимизации, может быть оправдано использование Brute Force.

Сравнивая алгоритмы работы Irradiance map и Light cache, понятно, что качество карты GI, получаемое с помощью Irradiance map, значительно выше, чем качество карты GI, полученной при использовании Light cache. Это связано с тем, что, несмотря на частичное формирование результата интерполяцией, благодаря своей адаптивности Irradiance map способен рассчитать GI с большей точностью в важных участках сцены. В то время как Light cache лишь определяет цвета GI для зон сэмплирования в виде мозаики, без разбора деля на них сцену.

Конечно же, можно существенно завышать настройки Light cache, чтобы получить результат, сравнимый по визуальному качеству с результатом Irradiance map. Но в этом случае неизбежно возрастет время расчета Light cache. Такой подход плавно приблизит нас в сторону Brute Force, практически сведя на нет разницу во времени расчета между ним и Light cache с сильно завышенными настройками. Кроме того, расход оперативной памяти и проблемы с распределением нагрузки на все доступные системе потоки при большом размере карты Light cache будет также весьма существенным минусом.

Алгоритм Irradiance map рассчитывает более мягкую, по сравнению с Light cache, карту GI. Финальный рендер выходит более приятным, без шумного GI. И поэтому Irradiance map следует использовать как движок просчета глобального освещения.

Irradiance map в принципе не может быть установлен как Secondary bounces GI engine. Учитывая всё перечисленное, Irradiance map может быть выбран лишь как GI движок для расчета Primary bounces.

Алгоритм Light cache более грубый, нежели алгоритм Irradiance map. Однако влияние вторичных отскоков GI на общее освещение сцены менее существенно, чем влияние первого отскока GI. Именно поэтому менее точный Light cache прекрасно справится с задачей просчета Secondary GI bounces. Кроме того, его функция Show calc. phase сгенерирует превосходное превью.

И тем не менее для каждой сцены оптимальна своя связка, подобранная опытным путем.

3.10 Свиток V-Ray Frame Buffer

В дополнение к окну визуализируемого кадра 3ds Max (Rendered Frame Window RFW или VFB), V-Ray позволяет производить визуализацию в специальный буфер кадра V-Ray, который имеет некоторые дополнительные возможности:

- Позволяет видеть все элементы визуализации в одном окне и очень просто переключаться между ними.
- Хранит изображение в полном 32-битном формате с плавающей точкой.
- Позволяет производить простую корректировку цвета визуализированного изображения.
- Позволяет выбрать порядок, в котором просчитываются бакиты.

Использование **V-Ray Frame Buffer** значительно упростит работу с V-Ray своей функциональностью и удобством, как и исключит возможные недоразумения с гамма-коррекцией.

Активизация функции **Get resolution from MAX** избавит от необходимости постоянно заходить во вкладку V-Ray, свиток V-Ray/Frame Buffer и там ddjlbnn нужное разрешение рендера. Сама же стандартная функция смены разрешения 3ds Max всегда находится во вкладке Common, позволяя быстро и оперативно осуществлять смену разрешения.

Разделы **V-Ray raw image** и **Split render channels** отвечают за формирование форматов, необходимых для постобработки.

3.11 Свиток V-Ray Global switches

Общие настройки позволяют глобально управлять различными свойствами визуализатора. Наиболее интересными параметрами являются следующие:

Hidden lights (скрытые источники света) – разрешает или запрещает использование скрытых источников света. Когда этот параметр включен, источники света просчитываются вне зависимости скрыты они или нет.

Shadows (тени) – глобально разрешает или запрещает генерацию теней.

Show GI only (показать только GI) – когда эта опция включена, прямое освещение не будет включено в окончательную визуализацию. Заметим, что источники света все равно участвуют в расчете GI, однако в конце будет показан только непрямой свет.

Maps (карты) – разрешает или запрещает текстурные карты.

Glossy effects (эффекты размытия) – эта опция позволяет пользователю заменить все размытые отражения в сцене на четкие; полезно для тестовых просчетов.

Don't render final image (не просчитывать окончательное изображение) – когда эта опция включена, V-Ray будет считать только соответствующие карты глобального освещения (photon maps, light maps, irradiance maps). Эта опция полезна, если просчитываются карты для анимации со статической сценой и передвигающейся камерой (fly-through animation).

Стоит обратить внимание на параметр **Override Mtl**, позволяющий установкой галки в чекбоксе заменить все материалы в сцене на один тестовый для анализа освещения сцены, как правило, светло-серого цвета без преломлений и переотражений.

3.12 Свиток V-Ray Image sampler (Antialiasing)

В компьютерной графике алиасинг – это эффект, проявляющийся в виде зазубрин, возникающий при преобразовании векторной фигуры в растровую.

Как известно, все трехмерные объекты в некотором смысле являются векторными, так как на самом деле не имеют разрешения и могут быть отрендерены в растровые изображения любого размера, а их форма описана с помощью математических формул, а не с помощью массива цветов отдельных пикселей. Однако при попытке описать криволинейные формы с помощью пикселей, которые, по своей сути, являются квадратами, неизбежно возникнут зазубрины на криволинейных изгибах (рис. 24).

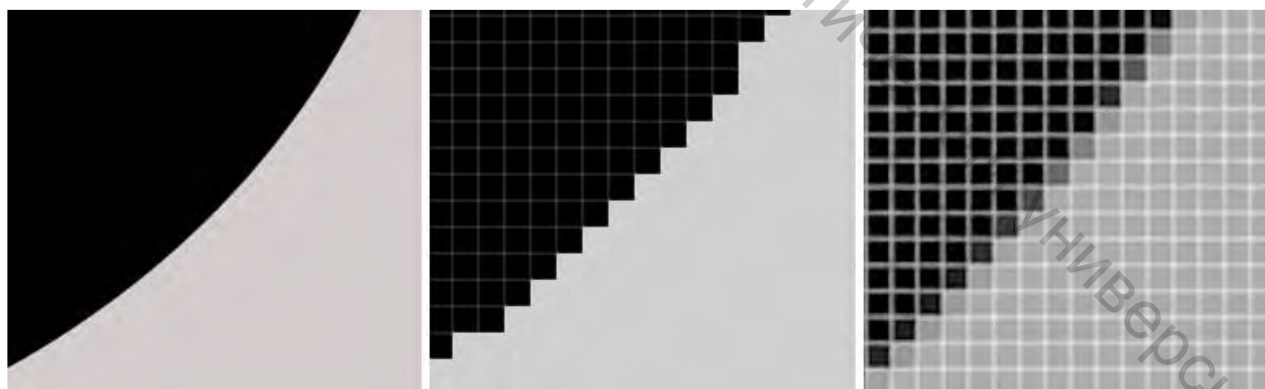


Рисунок 24 – Устранение алиасинга

Можно довести относительный размер пикселя до незаметного для глаза размера и таким образом визуально устранить этот эффект. Не всегда рационально производить растеризацию векторных фигур в сверхвысоком разрешении. Специальный метод призван устранить алиасинг и называется антиалиасингом [12].

Суть антиалиасинга в том, что пиксели-зубрины окрашиваются не в цвет фигуры или фона за ней, а усредняются, окрашиваясь в смешанный цвет фигуры и фона. Таким образом, как бы визуальнo размывая острые края фигуры. При этом происходит усреднение как пикселей самой фигуры, так и пикселей смежного с ней фона.

В процессе антиалиасинга применяется суперсэмплирование. Таким образом, получается не просто цвет фигуры или фона, а конкретный суммарный цвет всех цветов субпикселей.

В V-Ray существует несколько алгоритмов усреднения цветов пикселей на границах криволинейных фигур. Это Fixed rate sampler, Adaptive DMC sampler, Adaptive subdivision sampler и Progressive.

Именно для выбора одного из этих алгоритмов антиалиасинга предназначен свиток V-Ray/Image sampler (Antialiasing), а также выбора фильтра антиалиасинга (рис. 25).

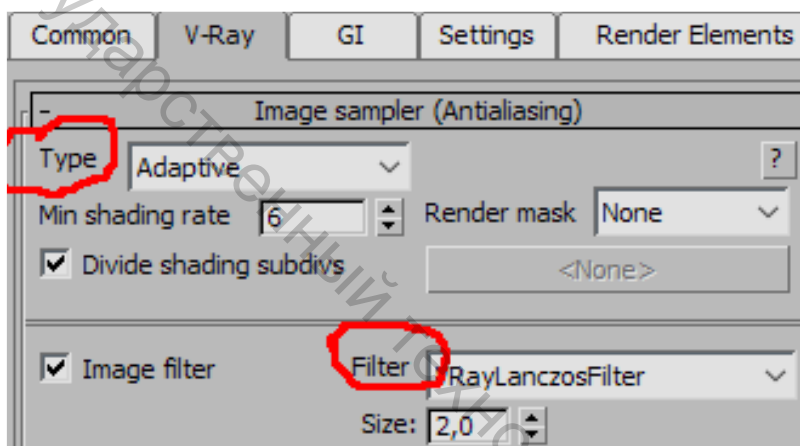


Рисунок 25 – Свиток V-Ray/Image sampler (Antialiasing)

Рассмотрим методы антиалиасинга. Следует отметить, что настройки выбранного метода отображаются не в свитке Image sampler (Antialiasing), а в одноименном с методом свитке, появляющемся после его выбора.

Fixed – простейший алгоритм, настраивается только одним параметром, Subdivs, определяющим, сколько лучей будет испущено в каждый пиксел (рис. 26).



Рисунок 26 – Настройки алгоритма Fixed

Количество лучей равно значению **Subdivs** в квадрате. При единице – изображение без антиалиасинга, при двух будет испущено четыре луча, трех – девять лучей и т. д.

Данный метод просчитывает каждый случай антиалиасинга с одинаковой тщательностью, независимо от условий. Поэтому этот алгоритм стоит применять в случаях, когда в сцене множество DOF, blur эффектов, высокодетализированных текстур и мелких деталей, на которых появляются артефакты, от которых нельзя избавиться никаким иным способом. Иногда бывает так, что просчет адаптивности другими алгоритмами может занять больше времени, чем просчет фиксированным значением. Однако в остальных случаях, за отсутствие приспособления к условиям, придется заплатить большим временем рендеринга [15].

Adaptive – это тот же Fixed, только с адаптивностью. Адаптивность заключается в автоматическом определении нужного числа субпикселей для каждого пикселя отдельно, а не использование фиксированного значения для всех пикселей изображения.

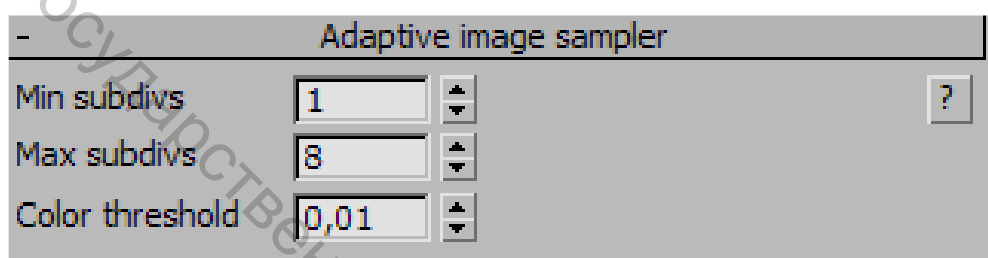


Рисунок 27 – Настройки алгоритма Adaptive

Параметры **Min subdivs** и **Max subdivs** определяют минимальное и максимальное число субделений каждого пикселя соответственно, в зависимости от требуемой величины адаптивности.

Color threshold – чем меньше пороги, тем чаще запускается процесс уточнения, и заканчивается либо при достижении пороговых значений, либо значения Max Rate.

Adaptive DMC sampler стоит применять в тех же случаях, что и Fixed, но, когда в сцене меньше проблемных деталей. Это позволит сэкономить время рендеринга при том же уровне качества, что и при фиксированном количестве субпикселей.

Adaptive subdivision – наиболее продвинутый адаптивный метод антиалиасинга в V-Ray. Его основное отличие от Adaptive, это возможность использования не только суперсэмплирования, но и субсэмплирования на маловажных участках генерируемого изображения, тем самым экономя львиную долю вычислительного времени. Минимальное и максимальное значение субсэмплирования и суперсэмплирования определяют значения Min. rate и Max. rate. Этими значениями определяется количество сэмплов, т. е. образцов цвета, используемых для определения цвета получаемых пикселей.

В частности, если значение rate равняется единице, то это значит, что для вычисления цвета четырех соседних пикселей используется один сэмпл. Значение ноль означает, что для каждого пикселя используется один сэмпл, а

значение rate и т. д. Значение rate – это степень числа четыре в конечном количестве субпикселей-образцов цвета конкретного пикселя растрируемых векторных фигур 3d-сцены. Как и в Adaptive sampler, за автоматический выбор нужного значения Min. rate и Max. rate отвечает Color thresh. параметр.

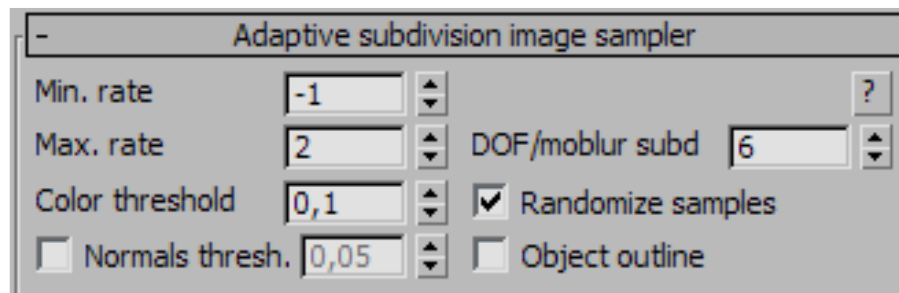


Рисунок 28 – Настройки алгоритма Adaptive subdivision

Adaptive subdivision sampler является наиболее гибким, наиболее технологичным методом антиалиасинга, применяемым в V-Ray. Именно он дает наибольшую скорость вычислений при наилучшем качестве получаемого изображения. Только при наличии не устранимых другими способами артефактов на мелких деталях следует использовать менее адаптивный и менее гибкий Adaptive или грубый Fixed метод.

Progressive – достаточно новый метод, который позволяет во время рендеринга видеть постепенное улучшение картинки. Вы задаете V-Ray число подразбиений, и он просто будет работать пока не достигнет этого значения. Таким образом для Progressive rendering просто устанавливается максимальное подразбиение на очень большое значение.

Min. subdivs и **Max. subdivs** (минимальное и максимальное подразбиение) – ключевая часть настроек прогрессивного рендеринга. Инструмент сэмплирования изображения отвечает за улучшение вашего рендеринга, чтобы вся геометрия была предоставлена точно, насколько это возможно на экране. При минимальном и максимальном подразбиении V-Ray подбирает подходящее решение. Он начинает с минимального значения подразбиения и если значение подходящее, то берёт его для вычислений. Если потребуется, то он будет продолжать двигаться до максимального подразбиения.

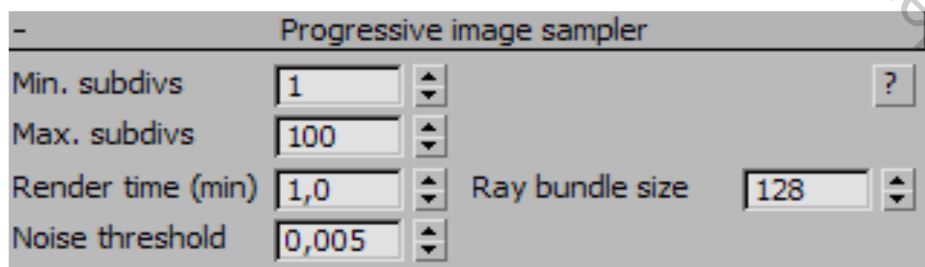


Рисунок 29 – Настройки алгоритма Progressive

Noise threshold (шумовой порог) – устанавливает желаемый уровень шума. Небольшое значение (в том случае, если времени рендеринга будет достаточно) сделает конечное изображение с минимальным количеством шума. По умолчанию это 0, что означает, что рендер будет сэмплировать равномерно, пока максимальное значение подразбиения не будет достигнуто или же не будет достигнут предел времени рендеринга.

Render time (максимальное время рендеринга) – максимальное значение времени в минутах, в рамках которого будет работать рендер. Это время не включает в себя проходы глобального освещения, как и Light cache или Irradiance map. Если нет уверенности, что времени на рендеринг хватит, то необходимо установить это значение на 0.

Antialiasing фильтры, несмотря на большой выбор и споры по их выбору, по сути, относительно равномерно размывают все изображение.

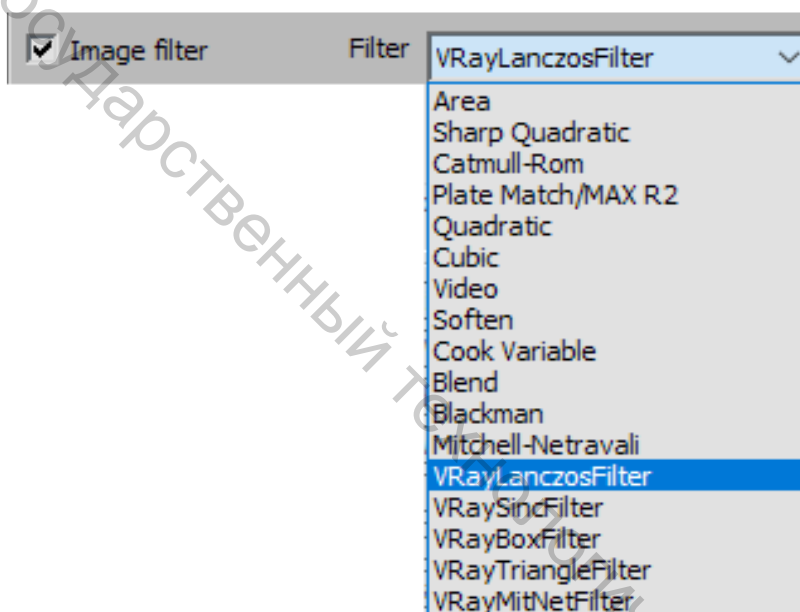


Рисунок 30 – Antialiasing фильтры

Разницу в действии различных фильтров, в подавляющем большинстве, можно увидеть только при значительном увеличении изображения. Применение Antialiasing фильтров очень сокращает свободу постобработки полученного изображения, лишая возможности качественно навести резкость нужной величины вручную. Кроме того, фильтры, наводящие резкость на границы объектов, очень усиливают паразитный эффект муара, даже при высоких настройках Image sampler.

Именно по этим причинам не стоит применять Antialiasing фильтры вообще. Лишь в некоторых случаях, например, в анимации, когда нет возможности наводить резкость на каждый из сотни кадров вручную, использование Antialiasing фильтров оправдано.

3.13 Свиток V-Ray Color mapping

В этом свитке находятся параметры, отвечающие за экспозицию и гамма-коррекцию визуализации.

Экспозиция – термин, пришедший в компьютерную графику из фотографии. Под экспозицией подразумевается количество света, попавшего на пленку во время фотографирования. Фотография может быть удачно экспонирована, т. е. выглядеть хорошо, недоэкспонированной – быть слишком темной, тусклой и переэкспонированной – быть излишне светлой. В компьютерной графике экспозиция обозначает примерно то же самое, а именно – яркость, насыщенность цветов сгенерированного изображения.

Гамма-коррекция – это необходимая коррекция и просчет изображения с определенной степенью нелинейности градиента тонов, от темного к светлому.

V-Ray имеет множество алгоритмов контроля экспозиции. Наиболее популярные из них – это Exponential и Linear multiply.

3.14 Свиток V-Ray Environment

Группа параметров **GI Environment** (Окружающая среда для расчета GI) позволяет перекрыть настройки для окружающей среды 3ds Max для вычислений непрямого освещения. Эффект изменения окружающей среды для GI подобен использованию света неба (skylight).

Группа параметров **Reflection/Refraction environment** (Окружающая среда для расчета отражений/преломлений) позволяет перекрыть настройки для окружающей среды 3ds Max для вычислений отражений и преломлений. Также можно перекрыть настройки окружающей среды для отражений/преломлений на уровне материала (VRayMtl) или на уровне текстурной карты (VRayMap).

Если активировано перекрытие настроек для преломлений (группа Refraction environment override), эта группа параметров будет влиять и на отражения, и на преломления. Если активировано перекрытие настроек для преломлений, то эта группа параметров будет влиять только на отражения.

Группа параметров **Refraction environment** (Окружающая среда для расчета преломлений) переключает настройки окружающей среды только для преломлений. Если такое перекрытие не активировано, то при вычислении преломлений V-Ray будут использованы параметры окружающей среды, указанные в группе Reflection/refraction environment override.

On – опция разрешает перекрытие параметров окружающей среды.

Color – позволяет указать цвет для окружающей среды (цвет света из внешнего пространства). Этот параметр игнорируется, если для окружающей среды указана текстурная карта.

Multiplier – множитель для значения цвета. Множитель не влияет на текстуру (если она есть) для окружающей среды. Для управления яркостью карты для окружающей среды используется карта Output.

Texture – позволяет выбрать текстуру для окружающей среды для расчета GI. Если текстура указана, то параметр Color игнорируется.

4 РАСПРОСТРАНЕННЫЕ ПРОБЛЕМЫ ВТОРИЧНОГО ОСВЕЩЕНИЯ

У вторичного освещения всего два основных вида артефактов. Это шум и грязь. Причина возникновения и способы устранения шума абсолютно тривиальны. Причиной появления этих распространенных проблем – недостаточное качество сэмплирования GI. В Irradiance map за это отвечает HSph. subdivs, а в Brute force и Light cache – Subdivs. Повышение этих параметров уменьшит количество шума на картах GI или уменьшение параметра Color Threshold.

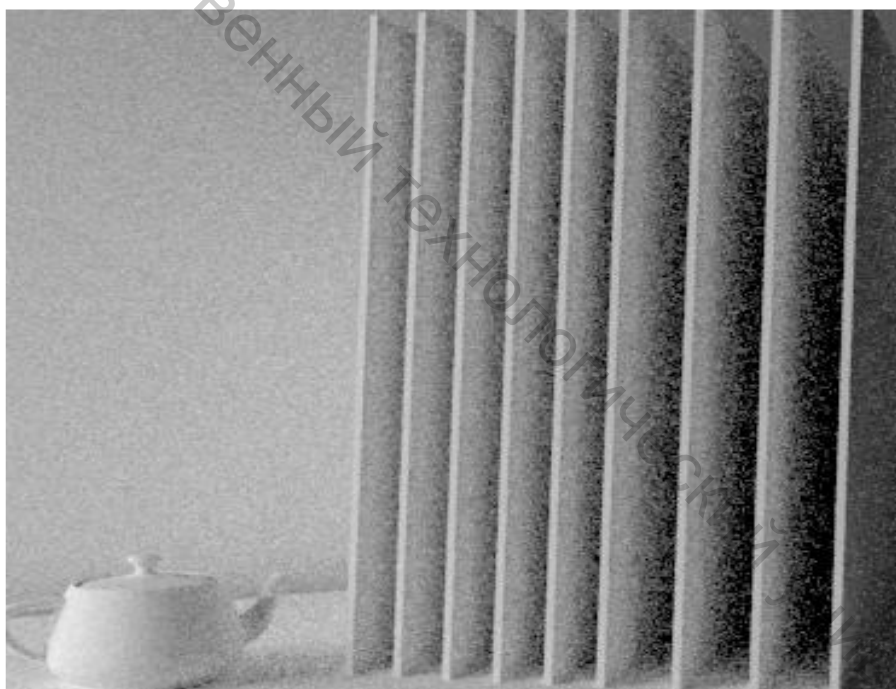


Рисунок 30 – Шум на картах GI

Недостаточная детализация карты Irradiance map – причина возникновения грязных пятен на визуализации или как их еще называют «вата», «хлопья», которыми частенько усыпаны мелкие детали сцены.

Грязь, а именно черные, серые и даже светлые пятна могут возникать в карте GI, созданной алгоритмом Irradiance map. Это происходит из-за недостаточной детализации самой карты Irradiance map.

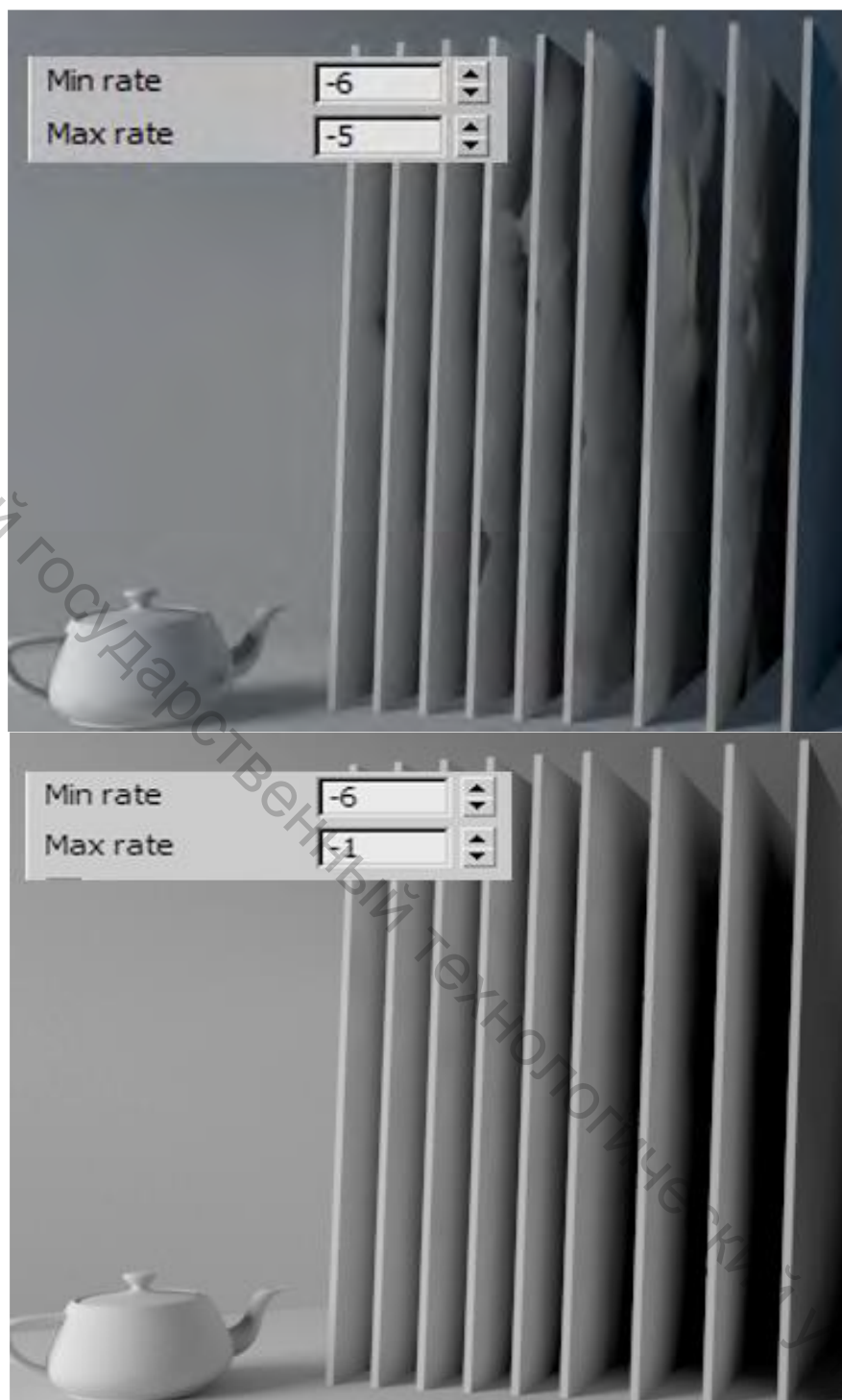


Рисунок 31 – «Вата» на картах GI

Для получения детализированной Irradiance map и избавления от пятен в углах необходимо увеличивать разрешение последнего предпрохода Irradiance map, а не увеличивать Subdivs.

Эффект ступенек на тонких линиях возникает также из-за недостаточного сэмплирования в процессе антиалиасинга.

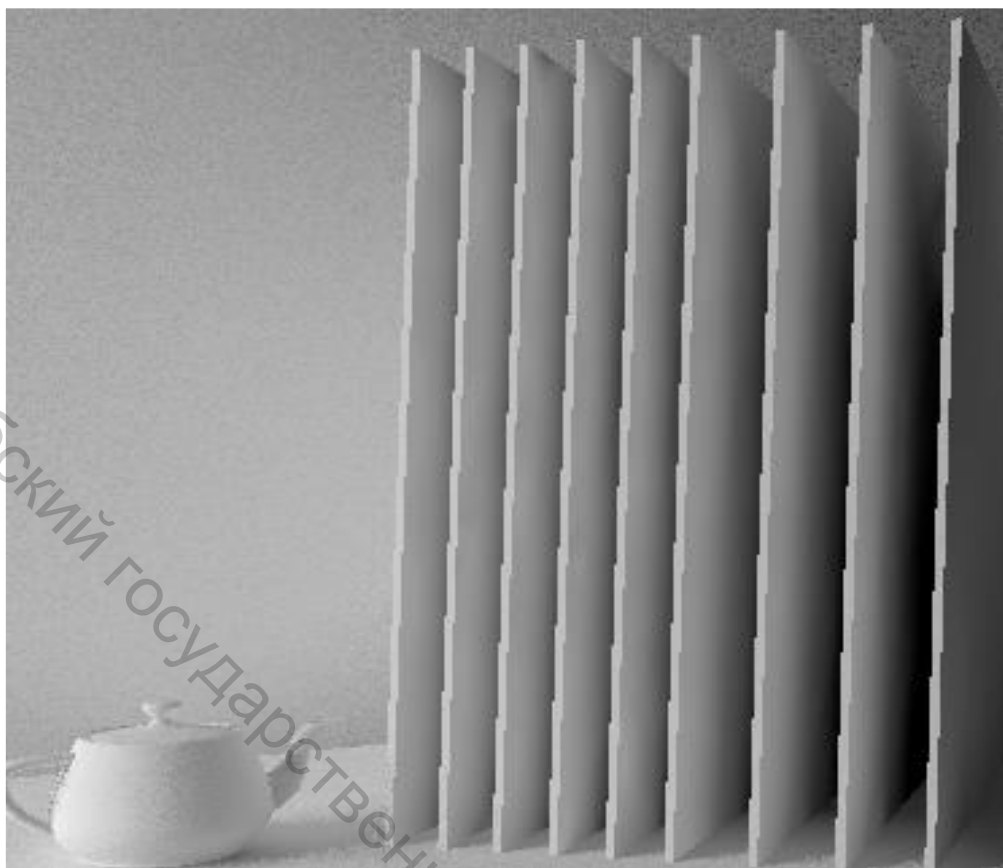


Рисунок 32 – Недостаточное сэмплирование в процессе антиалиасинга на картах GI

Любые артефакты необходимо рассматривать в контексте сцены и освещения, комбинируя и подбирая нужные настройки. Лучший советник в этом вопросе, конечно же, личный опыт, который необходимо приобретать экспериментальным путем, порой методом проб и ошибок.

ЛИТЕРАТУРА

1. <https://docs.chaosgroup.com/display/VRAY3MAX/Brute+Force+Settins/>
2. <http://fb.ru/article/236311/render-eto-cto-takoe-kak-ego-nastroit/>
3. <http://ru.renderstuff.com/irradiance-map-vray-urok-177/>
4. <http://ru.renderstuff.com/indirect-illumination-vray-urok-176/>
5. http://checkago.3dn.ru/_ld/0/49_Vray_1.5__24.08.pdf
6. <http://3dyuriki.com/2010/11/30/izuchaem-vizualizaciyu-vray-renderman-fryrender-maxwell-mental-ray/>
7. <http://3drnd.ru/prezhde-chem-nachat-vrayit-10-sovetov-po-vray/>
8. <http://www.studfiles.ru/preview/4437280/page:18/>
9. http://vraydoc.narod.ru/vray150sp1/vfb_index.html
10. <http://www.3dgrafia.ru/antialiasing-v-v-ray/>
11. <http://render.ru/book/931>
12. <http://www.3dmodelizm.ru/uroki-3d-max/3dmax-lessons/173-v-ray-3-0-progressive-rendering>
13. http://vraydoc.narod.ru/vray20/render_params_environment.htm
14. <http://www.3dmax-dvd.ru/Environment.html>
15. <https://docs.chaosgroup.com/display/VRAY3MAX/Environment+Settigs/>

Учебное издание

**Компьютерные технологии в дизайн-проектировании.
Настройки V-Ray**

Методические указания к практическим занятиям

Составитель:

Абрамович Наталья Анатольевна

Редактор *Н. В. Медведева*

Корректор *Т. А. Осипова*

Компьютерная верстка *Н. А. Абрамович*

Подписано к печати 11.12.17. Формат 60x90 ¹/₁₆. Усл. печ. листов 2,75.

Уч.-изд. листов 2,8. Тираж 30 экз. Заказ № 409.

Учреждение образования «Витебский государственный технологический университет»
210035, г. Витебск, Московский пр., 72.

Отпечатано на ризографе учреждения образования

«Витебский государственный технологический университет».

Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий № 1/72 от 12 февраля 2014 г.

Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий № 3/1497 от 30 мая 2017 г.