

**ДВОЙНОЕ ПРЕДСТАВЛЕНИЕ ГЕОМЕТРИИ ДЛЯ УСКОРЕНИЯ
ТРАССИРОВКИ ЛУЧЕЙ В ОПТИЧЕСКИХ СИСТЕМАХ
С ПОВЕРХНОСТЯМИ СВОБОДНОЙ ФОРМЫ**© 2024 г. Д. Д. Жданов^{а,*}, И. С. Потемин^{а,**}, А. Д. Жданов^{а,***}^аСанкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики
197101 Санкт-Петербург, Кронверкский пр., 49, Россия

*E-mail: ddzhdanov@mail.ru,

**E-mail: ipotemin@yandex.ru,

***E-mail: andrew.gtx@gmail.com,

Поступила в редакцию: 19.01.2024 г.

После доработки: 19.01.2024 г.

Принята к публикации: 24.01.2024 г.

В статье исследуется возможность использования двойного представления геометрии для повышения скорости трассировки лучей и обеспечения устойчивости результатов моделирования распространения света в сложных оптических системах, содержащих поверхности свободной формы, заданные полиномами высокого порядка (до 34-го порядка) или полиномами Якоби. Был проведен анализ традиционных методов представления данной геометрии как в виде треугольной сетки, так и в виде аналитического выражения. Проведенный анализ продемонстрировал недостатки традиционных подходов, которые заключаются в недостаточной точности вычисления координат точки встречи луча с треугольной сеткой, а также в неустойчивости результатов поиска точки встречи касательных лучей с аналитической поверхностью при использовании существующих методов расчета. В результате было предложено использовать двойное представление геометрии в виде грубого приближения поверхности треугольной сеткой, которое в дальнейшем используется как начальное приближение для поиска точки встречи луча с поверхностью, заданной аналитическим выражением. Это решение позволило существенно ускорить сходимость аналитических методов и повысить устойчивость их решений. Использование библиотеки Intel® Embree для быстрого поиска точки встречи луча с грубой треугольной сеткой и векторной модели вычислений для уточнения координат точки пересечения луча с геометрией, представленной аналитическим образом, позволило разработать и реализовать алгоритм трассировки лучей в оптической системе, содержащей поверхности с двойным представлением геометрии. Эксперименты, проведенные с использованием разработанного и реализованного алгоритма, показывают значительное ускорение трассировки лучей при сохранении точности вычислений и высокой стабильности результатов. Результаты были продемонстрированы на примере расчета функции рассеивания точки и бликов для двух объективов с поверхностями свободной формы, заданными полиномами Якоби. Кроме того, для двух данных объективов был произведен расчет изображения, формируемого RGB-D-объектом, имитирующим реальную сцену.

Ключевые слова: трассировка лучей, рендеринг, треугольная сетка, поверхности произвольной формы

DOI: 10.31857/S0132347424030067, **EDN:** QANTVA

1. ВВЕДЕНИЕ

Для уменьшения габаритных размеров и числа линз при сохранении или повышении качества изображения в современных оптических системах применяются линзы с поверхностями свободной формы. Такие объективы имеют, как правило, малый размер и применяются в носимых устройствах, например, камерах смартфонов, системах сканирования, устройствах виртуальной, дополненной и смешанной реальностей.

На рис. 1 представлен пример объектива камеры дополненной реальности.

Способ задания поверхностей свободной формы может быть различен. Это могут быть, например, асферические поверхности, заданные полиномами высокого порядка, поверхности, заданные сплайном, поверхности, заданные полиномами Якоби. Для эффективного проектирования и компьютерного моделирования таких систем используются средства фотореалистичной

визуализации, которые позволяют формировать не только реалистичное изображение сцены, но и изображение, наблюдаемое через объектив камеры с учетом всех его абберрационных искажений. Существуют два основных подхода для формирования таких изображений. Во-первых, это применение передаточной функции объектива к точкам изображения сцены. При этом передаточная функция должна быть вычислена для всех видимых точек сцены с учетом их пространственного положения относительно объектива камеры. Второй подход основан на трассировке лучей, и в этом случае объектив камеры становится неотъемлемой частью сцены. Точки объектов сцены становятся источниками каустического освещения для изображения, формируемого камерой. Каждый из подходов имеет свои преимущества и недостатки. Преимущество первого подхода заключается в возможности физически корректного расчета функции рассеивания отдельных точек объекта. Передаточная функция, используемая при этом расчете, может быть вычислена с учетом дифракции на краях линз. Основным недостатком этого подхода является его трудоемкость, сложность сопряжения со стохастическими методами трассировки лучей в сцене и определение видимости точек трехмерной сцены со стороны объектива камеры. Все это не позволяет применять данный подход к построению изображений трехмерных сцен и ограничивает его применение синтезом изображений плоских объектов, ортогональных оси объектива. Основным недостатком второго подхода является невозможность учета дифракционных явлений в объективе каме-

ры и шум, вызванный стохастическим методом трассировки лучей в сцене. Однако этот подход позволяет физически корректно передать трехмерность изображения сцены и глубину резкости объектива.

Основой обоих подходов является трассировка лучей в объективах, которая может быть использована как для расчета передаточных функций, так и для передачи яркости точек объекта на матрицу изображения объектива. Поскольку поверхности объективов имеют достаточно сложную форму, описываемую полиномами высокой степени, например, полиномами Якоби [1], которые в последнее время все чаще стали использоваться для описания форм асферических поверхностей свободной формы, то задача поиска точки встречи луча с такими поверхностями становится крайне актуальной. В работе [2] предложены параметры разложения полиномов Якоби, которые позволяют избежать ошибок при вычислении значений полиномов высокого порядка, поскольку в ряде случаев степень полинома может достигать нескольких десятков. Кроме того, предложенные параметры разложения позволяют оценить проблемы, связанные с технологией изготовления асферических поверхностей [3].

В вычислительной оптике для поиска точки встречи луча с асферической поверхностью используется метод Федера [4]. Несмотря на то что метод был предложен более 70 лет назад, он не претерпел серьезных изменений до настоящего времени. Суть этого метода заключается в последовательном поиске точки встречи луча с касательной плоскостью к асферической

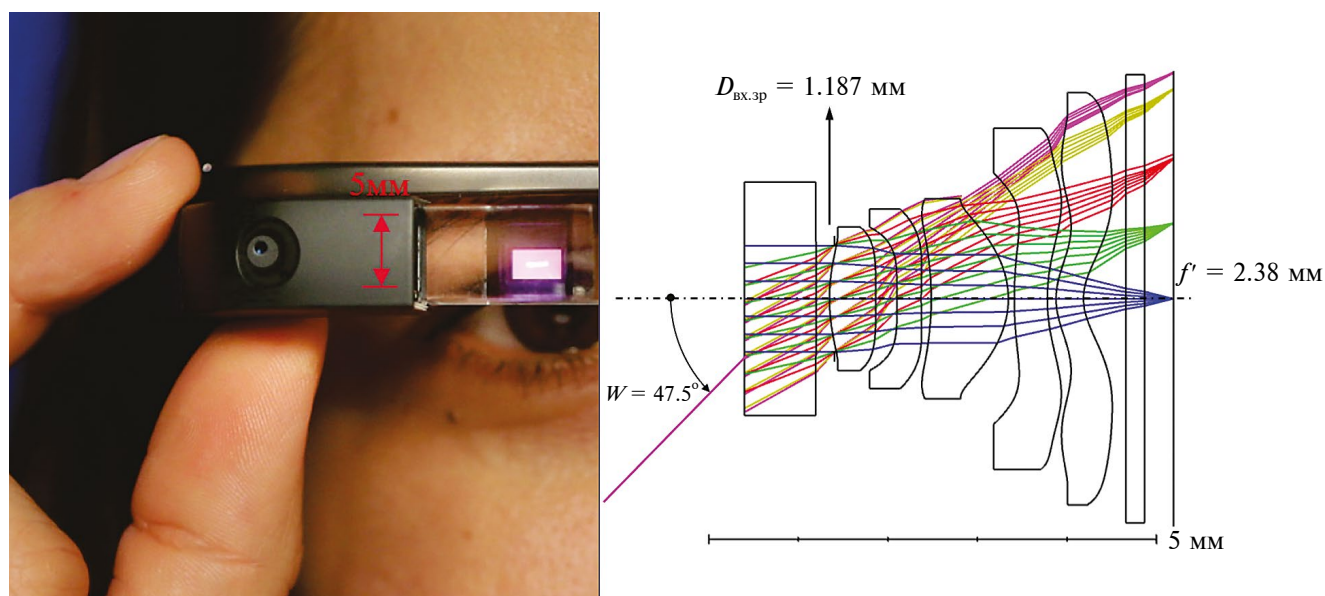


Рис. 1. Пример объектива камеры дополненной реальности.

поверхности. Параметры касательной плоскости уточняются после каждого найденного пересечения в соответствии с параметрами асферической поверхности для координат в найденной точке. В качестве первого приближения касательная плоскость устанавливается

$$z = 0, \quad (1)$$

где ось z — оптическая ось, а асферическая поверхность задана в локальной системе координат с вершиной в точке $(0, 0, 0)$.

Как правило, в компьютерных системах вычислительной оптики асферическая поверхность задается двумя типами полиномов. Это стандартное полиномиальное представление:

$$z(\rho) = \frac{c\rho^2}{1 + \sqrt{1 - \varepsilon c^2 \rho^2}} + \sum_{m=0}^M a_m \rho^{2m+4}, \quad (2)$$

или так называемая асферическая поверхность Q типа (Q -type), заданная полиномом Якоби:

$$z(\rho) = \frac{c\rho^2}{1 + \sqrt{1 - \varepsilon c^2 \rho^2}} + D_{con} \left(\frac{\rho}{\rho_{max}} \right), \quad (3)$$

где c , ε , a_m — коэффициенты асферической поверхности; $D_{con}(u) = u^4 \sum_{m=0}^M a_m Q_m^{con}(u^2)$, $\rho^2 = x^2 + y^2$,

а Q_m^{con} — задаются полиномами Якоби следующего вида:

$$Q_m^{con}(u) = P_m^{(0.4)}(2u - 1), \quad (4)$$

где $P_0^{(0.4)}(u) = 1$, $P_1^{(0.4)}(u) = -(5 - 6u)$, $P_2^{(0.4)}(u) = 15 - 14u(3 - 2u)$ и т. д.

Оба типа полиномиального представления используют одинаковый способ подгонки луча к поверхности. Для нецентрированных асферических поверхностей метод Федера претерпел незначительные изменения [5]. Данная модификация метода, предложенного Федером, касалась возможности его применения не только к нецентрированным поверхностям, но и к различного рода неосесимметричным поверхностям, включая поверхности дифракционных решеток [6].

Несмотря на очевидную простоту методов, предложенных Федером, многократное вычисление полиномов высокой степени, особенно полиномов Якоби, является очень затратной процедурой и требует оптимизации. Кроме того, при вычислении значений полиномов высокой степени возможна потеря точности, а в ряде случаев это может привести к переполнению данных.

Такая ситуация возникает, если начальная точка и направление луча значительно отличаются от расчетных значений, например, при расчете бликов в линзовых объективах.

Для ускорения поиска точки встречи луча с поверхностью и повышения стабильности вычислений можно использовать подходы, применяемые в компьютерной графике. Простейшим решением является аппроксимация асферической поверхности треугольной сеткой. При этом вершины треугольников совпадают с точками на асферической поверхности, а нормали в вершинах — с нормальными в соответствующих точках поверхности. Однако для достижения требуемой точности вычислений треугольная сетка должна иметь треугольники малого размера, что приводит к большому числу треугольников в сетке, т. е. увеличению объема памяти для хранения сетки, и замедлению вычислений. Для устранения этих недостатков в работе предлагается использовать двухуровневое представление геометрии, когда на первом уровне находится треугольная сетка, грубо аппроксимирующая асферическую поверхность и являющаяся ее начальным приближением для метода Федера, а на втором уровне — собственно асферическая поверхность.

2. ОЦЕНКА ТОЧНОСТИ ПРЕДСТАВЛЕНИЯ ГЕОМЕТРИИ

В качестве метрики, позволяющей оценить требуемую точность представления геометрии, использовалось среднеквадратичное отклонение функции рассеивания точки (ФРТ, Point Spread Function — PSF) от эталонной ФРТ, вычисленной классическим методом по формулам Федера [4]. Оценка проводилась для двух объективов, содержащих асферические поверхности, представленные формулами (2) и (3) соответственно. На рис. 2а показана схема четырехлинзового объектива с асферическими поверхностями, представленными формулой (2), а на рис. 2б показана схема пятилинзового объектива с асферическими поверхностями, представленными формулой (3). Диаметр входных зрачков обоих объективов составлял величину порядка 1 мм.

Для двух объективов был вычислен ряд ФРТ. ФРТ вычислялись для бесконечно удаленной точки, находящейся на оси объектива, для разных параметров модели геометрии объективов. Эталонные ФРТ, вычисленные по формулам Федера [4], представлены на рис. 3 для четырех- и пятилинзовых объективов соответственно. Поверхности исследуемой модели линзового

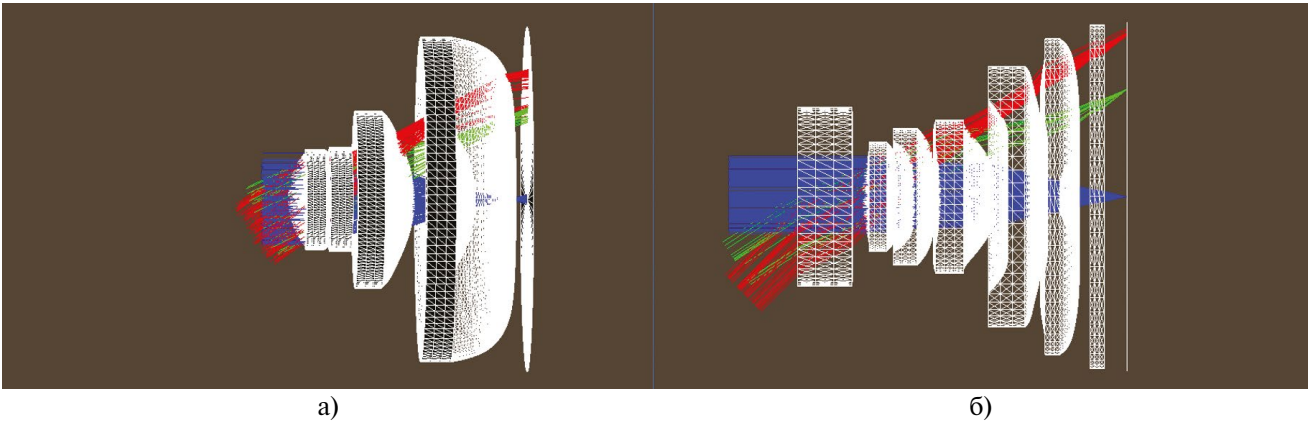


Рис. 2. Четырехлинзовый объектив с асферическими поверхностями, представленными формулой (2) – (а), пятилинзовый объектив с асферическими поверхностями, представленными формулой (3) – (б).

объектива представляли собой радиальную треугольную сетку, показанную на рис. 2, а исследуемый параметр модели – разрешение сетки, т. е. число треугольников. Пучок лучей от бесконечно удаленной точки равномерно заполнял входной зрачок объектива. Лучи в пучке распределялись по квадратной сетке с разрешением 10.000×10.000 . Вычисления производились на компьютере с восьмиядерным процессором AMD Ryzen 9 и 32 Гб оперативной памяти.

Были рассмотрены три варианта задания параметров радиального и углового разрешения треугольной сетки. Первый вариант – 24×54 (1.296 треугольников), второй вариант – 240×540 (129.600 треугольников), третий вариант – 2.400×5.400 (12.960.000 треугольников). Первый параметр – радиальное разрешение, второй – угловое. Были выполнены расчеты и проведена оценка отклонений ФРТ от эталонных значений. Поиск точки встречи лу-

ча с треугольной сеткой выполнялся средствами компьютерной графики [7]. Соответствующие изображения отклонений ФРТ от эталонных представлены на рис. 4. Размер изображения для четырехлинзового объектива, представленного на рис. 3а, был равен 100 мкм, для пятилинзового объектива, показанного на рисунке 3(б), – 5 мкм. Разрешение изображения в обоих случаях составляло 500×500 точек. Относительное среднее квадратичное отклонение ФРТ от эталона представлено в табл. 1.

Таблица 1. Относительное среднее квадратичное отклонение ФРТ от эталона

Параметры модели (разрешение)	RMS% модель рис. 2а	RMS% модель рис. 2б
24×54	133%	97%
240×540	96%	70%
2.400×5.400	1%	55%

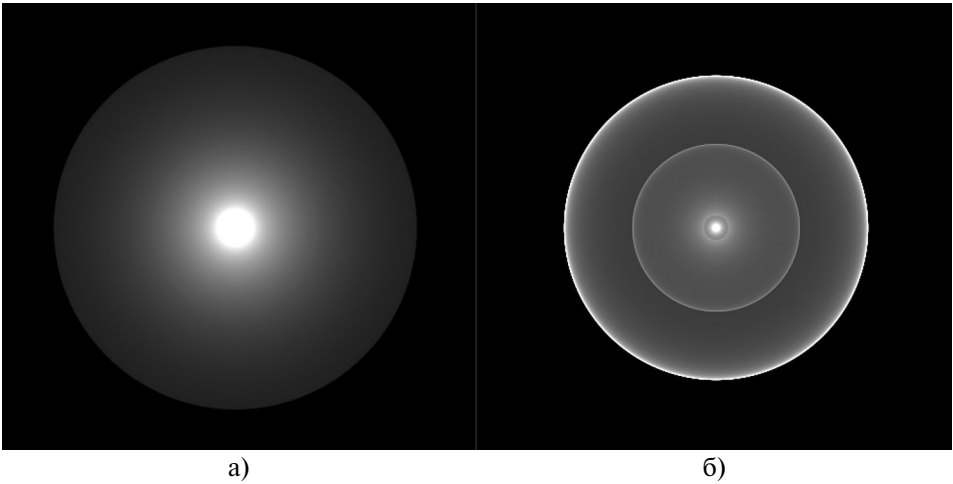


Рис. 3. ФРТ четырехлинзового объектива с асферическими поверхностями, представленными формулой (2) – (а), ФРТ пятилинзового объектива с асферическими поверхностями, представленными формулой (3) – (б).

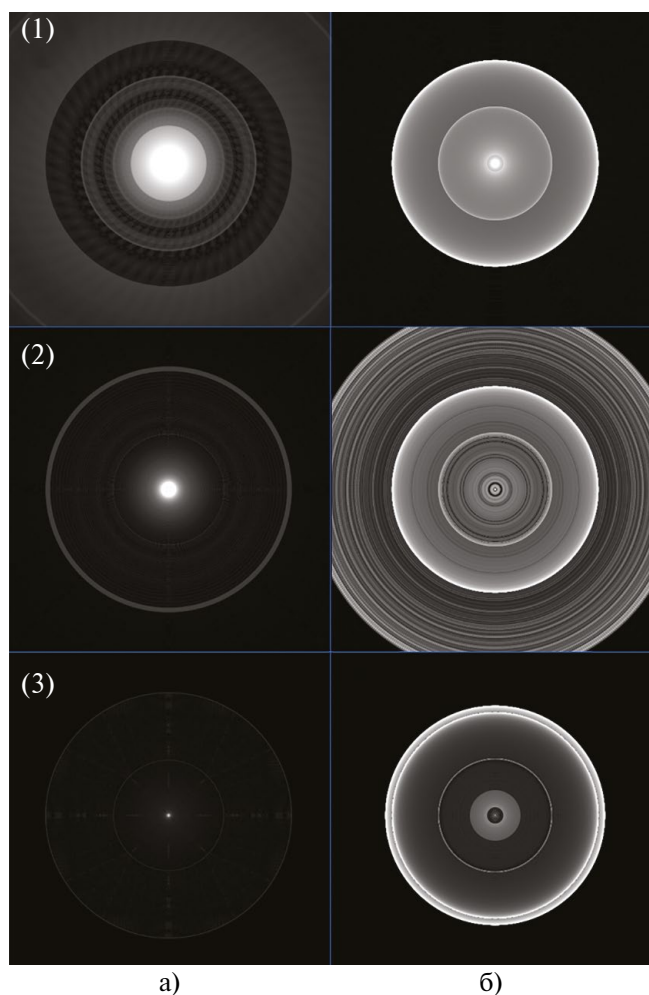


Рис. 4. Отклонения ФРТ от эталонов для четырехлинзового (а) и пятилинзового (б) объективов и трех вариантов разрешения треугольной сетки: 24×54 (1), 240×540 (2) и 2400×5400 (3).

По результатам, представленным на рис. 4 и в табл. 1, видно, что высокое разрешение треугольной сетки не всегда может обеспечить требуемую точность вычислений. Особенно это касается высококачественных объективов, у которых ФРТ имеет радиус порядка 1 мкм.

Для повышения точности, стабильности и ускорения вычисления точки встречи луча с поверхностью было реализовано комбинированное решение, заключающееся в двойном представлении геометрии, когда поиск точки встречи луча с поверхностью разбивается на два этапа, сперва происходит быстрый поиск точки встречи луча с грубой треугольной сеткой, от которой далее осуществляется подгонка к асферической поверхности по алгоритмам [4] и [1]. Это решение аналогично алгоритму “displacement mapping” [8], однако цель данного решения не уменьшение объема данных, а повышение стабильности, точности и скорости расчета.

3. ДВОЙНОЕ ПРЕДСТАВЛЕНИЕ ГЕОМЕТРИИ

Для исследования преимуществ и недостатков метода, основанного на двойном представлении геометрии, была реализована программа синтеза изображений, основанная на детерминистическом и квазидетерминистическом методе трассировки лучей. Хотя эта программа не является полноценной рендер системой, она может в детерминистическом режиме выполнять расчет ФРТ и изображений RGB-D-объектов, формируемых линзовыми или зеркально-линзовыми объективами. В квазидетерминистическом режиме эта программа позволяет выполнять расчет бликов, т. е. ФРТ, формируемую в результате переломлений между поверхностями линз. Ядром данной программы являлся метод расчета точки встречи луча с поверхностью, для реализации которого использовалась библиотека Embree от компании Intel [9]. Основные причины выбора API Embree заключались в следующем:

1. Исключительная ориентация на методы поиска точки встречи луча с поверхностью.
2. Встроенные высокопроизводительные методы поиска точки встречи луча с треугольной сеткой.
3. Поддержка векторных инструкций процессора (принцип Одиночного потока Команд, Множественного потока Данных – ОКМД, Single Instruction, Multiple Data – SIMD).
4. Поддержка параллельных вычислений на базе Intel oneTBB API [10].
5. Поддержка пользовательской геометрии и специальных фильтров, позволяющих модифицировать форму поверхности.
6. Интуитивный программный интерфейс, наличие большого числа примеров.

Трассировка лучей в линзовых объективах, формирующих изображение, предполагает строго фиксированный порядок следования поверхностей, т. е. траекторий луча, например, после первой поверхности должна идти вторая поверхность, после второй – третья и т. д. Если идет расчет бликов, то порядок следования может быть нарушен, однако он остается фиксированным, т. е. в случае блика порядок следования меняет знак, например, после отражения на третьей поверхности следующая поверхность будет второй. Кроме того, фиксированный порядок следования поверхностей позволяет избежать неоднозначности при поиске точки встречи луча с поверхностью в случае их близкого расположения, например, при наличии воздушного зазора между двумя поверхностями. Такая специфика формирования

трасс лучей в линзовых объективах позволяет эффективно использовать возможности API Embree, которое позволяет иметь неограниченное число сцен в одном вычислительном устройстве. Исходя из специфики трассировки лучей в объективе, формирующем изображение, и возможностей API Embree каждая поверхность объектива представляет собой отдельную сцену, а порядок следования поверхностей заменяется на порядок следования сцен в API Embree. Данный подход решает сразу несколько проблем. Во-первых, поиск точки пересечения для Embree ограничен одной поверхностью, например, одной треугольной сеткой. Для сложных объективов это позволяет значительно уменьшить число объектов (в десятки раз), с которыми происходит поиск точки встречи луча с поверхностью, и, соответственно, повысить скорость поиска. Во-вторых, это автоматически решает проблему неоднозначности поиска точки встречи луча с близкорасположенными или соприкасающимися поверхностями, поскольку эти поверхности принадлежат разным сценам. Данная организация сцены объектива проиллюстрирована на рис. 5.

Детерминистическая трассировка лучей в объективе может использоваться не только для расчета ФРТ, но и для расчета волновых aberrаций на его выходном зрачке, что позволяет вычислить передаточную функцию объектива. Кроме того, квазидетерминистическая трассировка лучей может использоваться не только для расчета бликов, где порядок следования поверхностей определяется случайным образом по коэффициенту зеркального отражения линзовой поверхности, но и для расчета оптических характеристик многоканальных и мультиконфигурационных

оптических систем. Для данных систем задается несколько траекторий лучей, соответствующих разным каналам (конфигурациям).

Одним из недостатков API Embree с точки зрения вычислительной оптики является то, что все расчеты с вещественными числами выполняются с вещественными числами одинарной точности, тогда как в вычислительной оптике используются вещественные числа с двойной точностью. Одинарная точность может быть недостаточна, особенно если степень полинома асферической поверхности достигает нескольких порядков, а луч сильно отклоняется от своей расчетной траектории, например, это возникает при расчете бликов. Поэтому для повышения точности вычислений используются специальные фильтрующие callback-функции, устанавливаемые средствами API Embree (`rtcSetGeometryIntersectFilterFunction`). Функция фильтрации вызывается при каждом выполнении операции поиска точки встречи луча с поверхностью (`RTCIntersect...`) и может изменить параметры точки встречи луча с поверхностью, т. е. точку пересечения, найденная с одинарной точностью, может быть пересчитана в двойную точность. Для вычислений, использующих векторные инструкции процессора, обработка несколько усложняется, поскольку вектор, состоящий из восьми вещественных чисел одинарной точности, необходимо преобразовать в два вектора из четырех вещественных чисел двойной точности, которые будут последовательно обрабатываться функцией фильтрации. Данный процесс проиллюстрирован на рис. 6.

Однако основное применение функций фильтрации – это реализация двойного представления геометрии. Основная идея метода заключается в том, что геометрия асферической поверхности должна иметь два представления. Первое, грубое представление, – это треугольная сетка низкого разрешения. Средства API Embree (функция `RTCIntersect...`) позволяют найти точку встречи луча с некоторой аппроксимацией истинной формы поверхности, заданной в виде треугольной сетки. Для небольшого числа треугольников в сетке поиск точки встречи выполняется исключительно быстро. Далее от найденной точки происходит итерационный поиск точки встречи луча с асферической поверхностью алгоритмами [4] и [1]. Итерационный поиск выполняется функцией фильтрации API Embree, которая возвращает точку пересечения луча с асферической поверхностью, и функция `RTCIntersect...` возвращает полученное значение как собственное пересечение луча с поверхностью. Очевидно, чем бли-

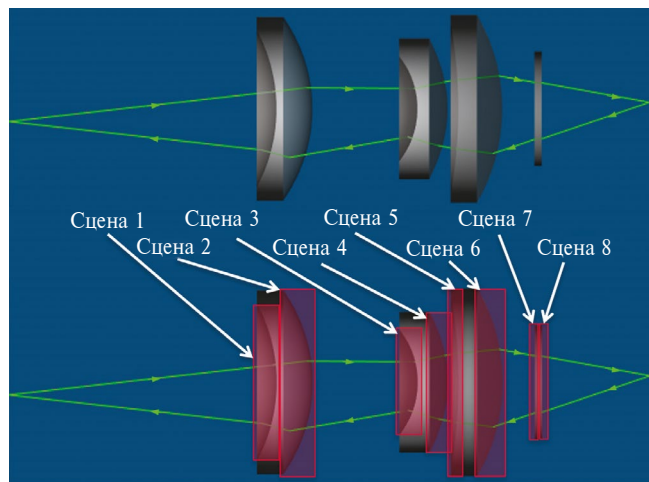


Рис. 5. Оптическая система объектива (верхний рисунок), представленная в виде набора Embree сцен (нижний рисунок).

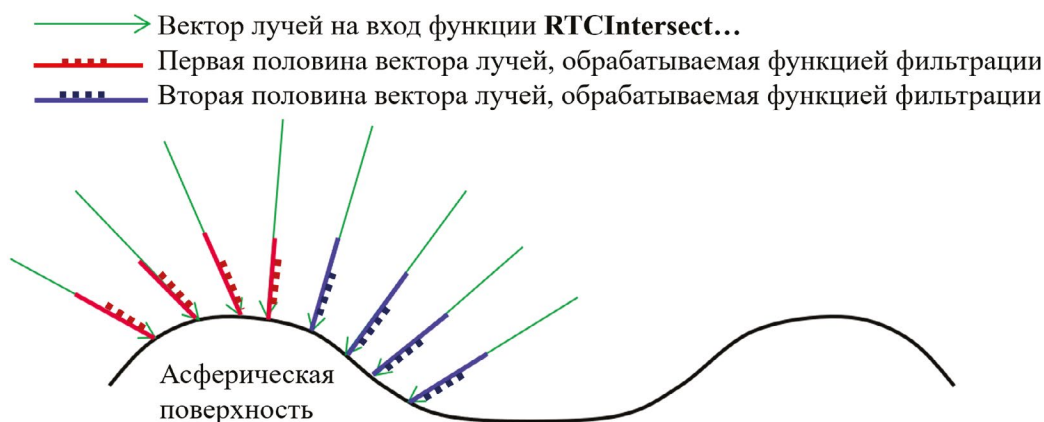


Рис. 6. Разбиение ОКМД-вектора из восьми вещественных чисел одинарной точности на два вектора из четырех вещественных чисел двойной точности в функции фильтрации.

же первое приближение оказалось к истинной поверхности, тем меньшее число итераций будет выполнено в процессе подгонки. Рис. 7 иллюстрирует этот алгоритм.

На рис. 7 также можно увидеть, что основное преимущество данного алгоритма появляется при наклонном падении лучей на поверхность. Если луч идет параллельно оптической оси, то

преимущество двойного представления геометрии пропадает, поскольку начальная точка также оказывается близко от истинной поверхности, а время, затрачиваемое на поиск точки пересечения луча с треугольной сеткой, не является нулевым. С точки зрения практического применения преимущество двойного представления геометрии возникает при расчете внеполевых точек

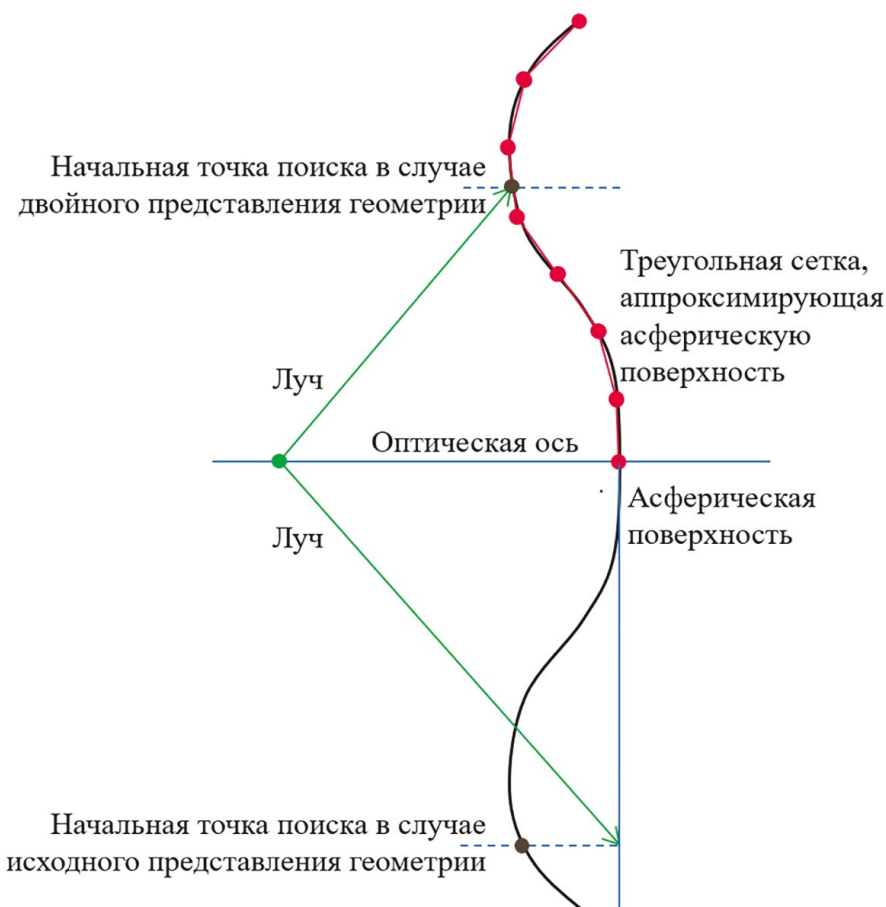


Рис. 7. Определение начальной точки встречи луча с асферической поверхностью, верхняя часть рисунка — случай двойного представления геометрии, нижняя часть рисунка — прямая подгонка по алгоритмам [4] и [1].

изображения и расчете бликов в линзовых системах, когда лучи идут не по расчетным траекториям. Двойное представление геометрии также решает проблему стабильности вычислений точки встречи луча с асферическими поверхностями, заданными полиномами высокой степени. Если луч пытается найти точку встречи с поверхностью в области, находящейся намного выше ее расчетной высоты, то может возникнуть переполнение разрядов вещественных чисел или потеря точности при сложении (вычитании) больших чисел.

Поскольку разработанная программа была ориентирована на синтез изображений, формируемых оптическими системами, то единственным источником формирования освещенности изображения является яркость прямого видения. Данное ограничение позволило минимизировать программный интерфейс методов работы с данными оптической системы. Для поиска точки встречи луча с поверхностью использовалась библиотека Embree. Для преобразования луча на поверхности задавались коэффициенты отражения (для линзовых поверхностей эти коэффициенты рассчитывались по формулам Френеля) и применялись соответственно законы отражения и преломления. Для оптических сред рассчитывалось поглощение в материале стекла. Лучи испускались источником света (точкой или RGB-D-изображением с заданными световыми характеристиками) и после прохождения через всю оптическую систему их световой поток аккумулировался на приемнике излучения.

Важным моментом ускорения вычислений является организация вычислений в параллельном режиме. Компания Intel предлагает библиотеку oneTBB для организации параллельных вычислений. Важно, что функции Embree оптимизированы на работу в параллельном режиме с функциями oneTBB, и это позволяет достичь практически стопроцентной загрузки всех ядер процессора при трассировке лучей. Алгоритм расчета ФРТ в параллельном режиме представлен на рис. 8.

Данный алгоритм подходит не только для расчета ФРТ, но и для расчета бликов и изображений RGB-D-объектов. В случае расчета бликов меняется алгоритм выбора следующей точки по списку трассировки лучей. Если при расчете обычной ФРТ поверхности идут строго по списку, то при расчете бликов список инвертируется, когда на поверхности выбирается событие отражения. При расчете изображений RGB-D-объектов происходит случайный выбор точки на RGB-изображении, а начало луча переносится на дистанцию, хранящуюся в карте глубин изображения,



Рис. 8. Пример алгоритма расчета ФРТ.

относительно центра входного зрачка. Координата луча на зрачке, которой соответствует угол относительно оси объектива, выбирается случайным образом. Поток, переносимый лучом, в данном случае будет пропорционален потоку, испускаемому соответствующей точкой RGB-D-изображения.

4. ПРИМЕРЫ РАСЧЕТА ФРТ И ИЗОБРАЖЕНИЙ, ФОРМИРУЕМЫХ ЛИНЗОВЫМИ ОБЪЕКТИВАМИ

Разработанный программный модуль распространения лучей в объективе может использоваться не только в задачах прикладной оптики.

Данное решение может найти применение как в системах компьютерной графики для синтеза фотореалистичных изображений трехмерных сцен, формируемых оптической системой, так и в задачах физически корректного виртуального прототипирования линзовых объективов в условиях реальной эксплуатации. Основное преимущество данного решения заключается в возможности внедрения детерминистической модели трассировки лучей в задачи рендеринга, основанного на методах Монте-Карло, который использует не только прямую, но и обратную трассировку лучей. В данном случае модуль трассировки лучей в объективе выступает в виде черного ящика, на вход которого падает луч, а на выходе формируется луч, покидающий объектив. Входом и выходом луча могут служить зрачки объектива или его первая и последняя поверхности, если речь идет о расчете бликов. Используя модель черного ящика [11], модуль трассировки лучей в объективе был встроен в систему компьютерного моделирования Lumiscept [12]. Данное решение позволило упростить задание линзовых систем и линзовой камеры в системе реалистичного рендеринга.

Для разработанного программного модуля было проведено тестирование времени расчета

ФРТ по разработанным алгоритмам и исходным решениям, описанным в [4] и [1]. Для тестирования использовались объективы, представленные на рис. 2. Тестирования производились на компьютере с восьмиядерным процессором AMD Ryzen 9 и 32 Гб оперативной памяти. В модели двойного представления геометрии поверхности линзы использовались 1.296 треугольника, что соответствовало разрешению треугольной сетки 24×54 . Число трассируемых лучей задавалось исходя из регулярной решетки с разрешением 10.000×10.000 на входном зрачке объектива, что в сумме давало 100.000.000 лучей. Поля зрения ω для четырехлинзового объектива составляли 0° , 26° и 37° по вертикали, для пятилинзового объектива поля зрения $\omega - 0^\circ$, 19° и 45° по вертикали. Размеры изображений для соответствующих полей зрения составили 100, 1.000 и 2.000 мкм для четырехлинзового объектива и 5, 15 и 100 мкм для пятилинзового объектива. Разрешение изображения составило 500×500 точек. Длина волны, на которой выполнялся расчет, — 0.5876 мкм. Изображения ФРТ, полученные в результате тестирования, представлены на рис. 9. Время, затраченное на трассировку данного числа лучей для разработанного алгоритма и алгоритма, описанного в работах [4] и [1], представлено в табл. 2. Изображения, полученные с использованием новых и старых алгоритмов, совпадают и не приводятся на рис. 9.

Исследование показало, что для осевых пучков разработанный алгоритм не дает ускорения расчета (скорее появляется небольшое замедление), однако для внеосевых пучков, располагающихся под большими углами к оптической оси объектива, новый алгоритм дает значительное ускорение, которое достигает 2.5 раз. Следует отметить, что при синтезе изображений, формируемых объективом, преобладают внеосевые пучки, поэтому

Таблица 2. Время расчета трех полей зрения для четырех- и пятилинзового объективов

Четырехлинзовый объектив	$\omega = 0^\circ$	$\omega = 26^\circ$	$\omega = 37^\circ$
Двухуровневая модель (время расчета, мс)	4.025	3.788	3.067
Модель по формуле (2) (время расчета, мс)	3.943	4.894	5.161
Пятилинзовый объектив	$\omega = 0^\circ$	$\omega = 19^\circ$	$\omega = 45^\circ$
Двухуровневая модель (время расчета, мс)	7.291	4.910	2.984
Модель по формуле (3) (время расчета, мс)	7.051	7.657	6.923

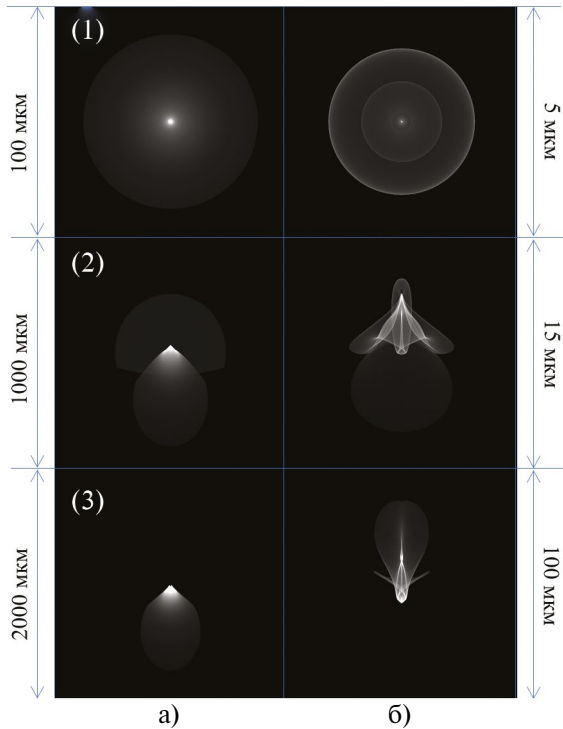


Рис. 9. ФРТ четырехлинзового объектива (а) для трех полей зрения: 0° (1), 26° (2) и 37° (3) и ФРТ пятилинзового объектива (б) для трех полей зрения: 0° (1), 19° (2) и 45° (3).

новый алгоритм может обеспечить значительное ускорение расчета. Также новый алгоритм обеспечивает большую стабильность вычислений трасс лучей, если они сильно отклоняются от расчетных траекторий, например, при расчете бликов.

Для двух рассмотренных моделей объективов был произведен расчет бликов, т. е. ФРТ, формирующейся с учетом зеркального переотражения между поверхностями линз. Событие зеркального отражения выбиралось случайным образом в соответствии с коэффициентом отражения, вычисленным по формуле Френеля для границы двух диэлектриков. При выборе события отражения менялся порядок следования поверхностей и луч детерминистически распространялся в обратную сторону. В результате четного числа переотражений формировалась “бликовая” ФРТ. В данном тесте исходные лучи распределялись случайным образом (равномерно) на первой поверхности объектива. Число трассируемых лучей — 1.000.000.000. Поля зрения ω для четырехлинзового объектива составляли 40° и 45° по диагонали, а для пятилинзового объектива поля зрения ω — 48.5° и 50° по диагонали. Размеры изображений составили 2.8 мм для четырехлинзового объектива и 3.6 мм — для пятилинзового объектива. Разрешение изображения составило 500×500 точек. Длина волны, на которой вы-

полнялся расчет, — 0.5876 мкм. Время расчета составило 16011 мс для четырехлинзового объектива и 15034 мс — для пятилинзового объектива. Несмотря на небольшой шум изображения, обусловленный недостаточным числом лучей, блики и их характер хорошо видны. Результаты расчета представлены на рис. 10.

Кроме того, для рассмотренных моделей объектива были синтезированы изображения RGB-D-объектов, представляющих собой цветную карту глубин, полученную с помощью компьютерной системы Lumiccept [12]. На рис. 11 представлены исходные данные для моделирования:

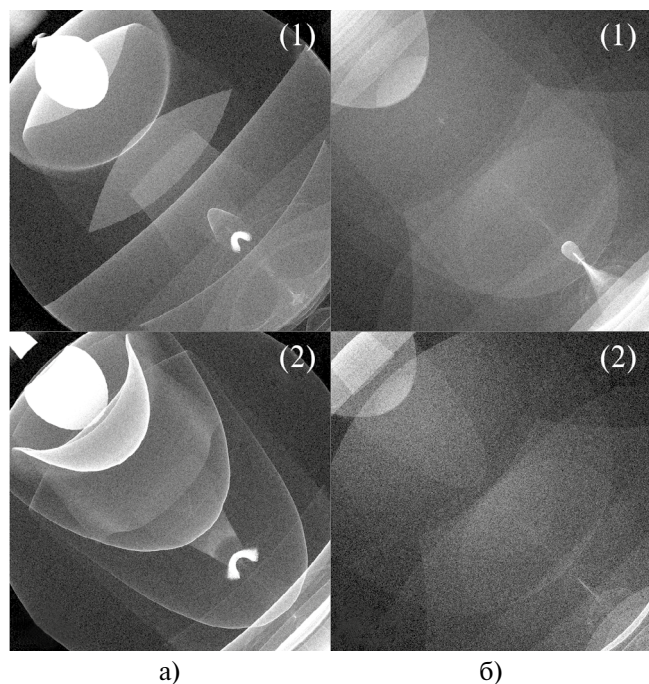


Рис. 10. ФРТ блика четырехлинзового объектива (а) для двух полей зрения: 48.5° (1) и 50° (2) и ФРТ пятилинзового объектива (б) для двух полей зрения: 48.5° (1), 50° (2).

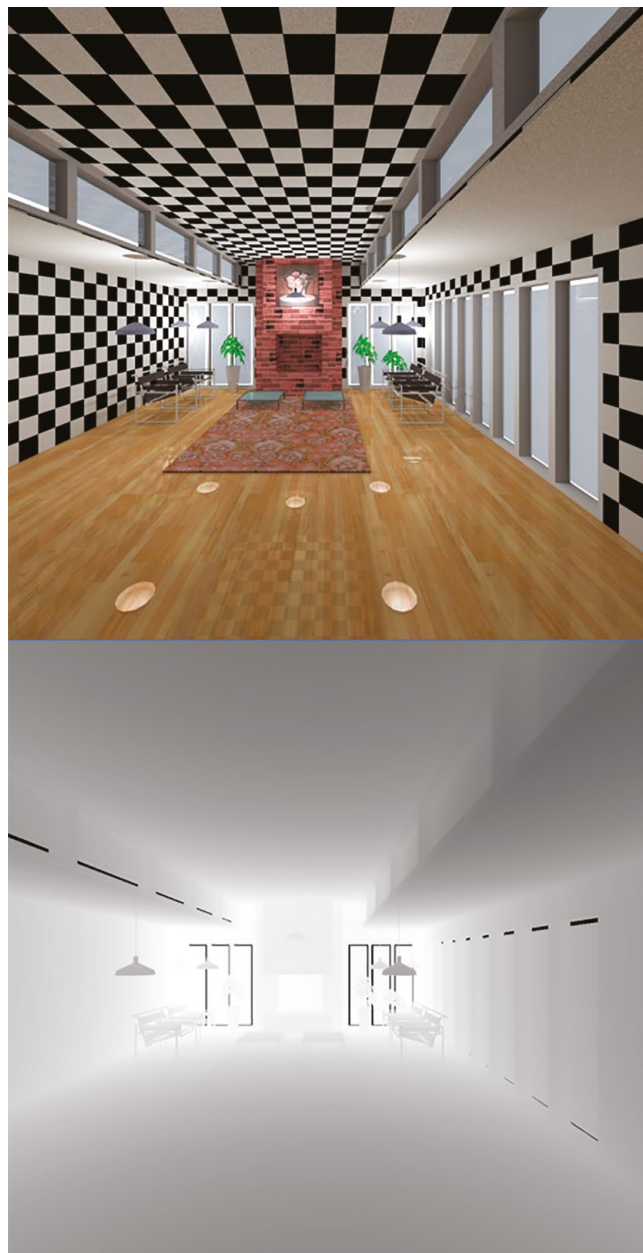


Рис. 11. RGB-изображение (сверху) и карта глубин (снизу).

RGB-изображение (RGB-карта яркостей) и соответствующая карта глубин сцены. Изображения были построены для двойного поля зрения 110° с разрешением 1200×1200 .

Изображения, формируемые четырех- и пятилинзовым объективами были рассчитаны для разрешения 600×600 точек и длины волны 0.5876 мкм. Число трассируемых лучей — $2.000.000.000$. Диагональные поля зрения четырех- и пятилинзового объективов составили 37° и 47.5° соответственно. Время расчета составило 79.534 мс для четырехлинзового объектива и 122.134 мс — для пятилинзового объектива.



Рис. 12. Изображения, построенные четырехлинзовым (сверху) и пятилинзовым объективом (снизу).

Изображение, построенное для четырехлинзового объектива, представлено на верхней части рис. 12, для пятилинзового объектива — на нижней части рис. 12.

Изображение, формируемое четырехлинзовым объективом, сильно размазано, что соответствует большим aberrациям объектива и согласуется с ФРТ, представленными на рис. 9. На изображении пятилинзового объектива видна небольшая дисторсия и размытие изображения по краю поля зрения. Размытие изображения, видимое как “затекание” светлых квадратов шахматной текстуры на черные квадраты на краю поля зрения, соответствует ФРТ, представленным на рис. 9.

5. ЗАКЛЮЧЕНИЕ

В результате проведенного исследования были продемонстрированы недостатки традиционных методов представления геометрии поверхностей свободной формы в системах компьютерного моделирования, а именно, представления в виде треугольной сетки или аналитического выражения. На основе анализа недостатков был предложен метод двойного представления геометрии свободной формы.

Для метода двойного представления геометрии был разработан алгоритм трассировки лучей в оптических системах, содержащих поверхности свободной формы, включая полиномы высокого порядка или полиномы Якоби. Данный алгоритм был реализован и встроен систему компьютерного моделирования Lumicert и протестирован на ряде сцен для решения задач оптического моделирования.

При тестировании программной реализации алгоритма исследовались скорость процедуры поиска точки встречи луча с поверхностью и объемом памяти, необходимый для хранения дополнительной информации о поверхности. Результаты тестирования показали высокую эффективность предложенного подхода при решении таких задач, как расчет функции рассеивания точки и расчет рассеянного света в оптической системе. Кроме того, была подтверждена стабильность вычислений точки встречи луча с поверхностью для лучей, чьи траектории сильно отличаются от расчетных, например, при анализе рассеянного света.

6. ИСТОЧНИК ФИНАНСИРОВАНИЯ

Работа была выполнена при финансовой поддержке Российского научного фонда, грант № 22-11-00145.

СПИСОК ЛИТЕРАТУРЫ

1. *Forbes G.W.* Shape specification for axially symmetric optical surfaces // *Opt. Express*. 2007. № 15. P. 5218–5226.
2. *Forbes G.W.* Robust, efficient computational methods for axially symmetric optical aspheres // *Opt. Express*. 2010. № 18. P. 19700–19712.
3. *Thompson K.P., Fournier F., Rolland J.P., Forbes G.W.* The Forbes Polynomial: A more predictable surface for fabricators // *International Optical Design Conference and Optical Fabrication and Testing*. OSA Technical Digest (CD) (Optica Publishing Group, 2010), paper OTuA6.
4. *Feder D.* Optical Calculations with Automatic Computing Machinery. *J. Opt. Soc. Am.* 1951. № 41. P. 630.
5. *Allen W., Snyder J.* Ray Tracing through Uncentered and Aspheric Surfaces // *J. Opt. Soc. Am.* 1952. № 42. P. 243.
6. *Spencer M.G.H., Murty V.R.K.* General Ray-Tracing Procedure // *J. Opt. Soc. Am.* 1962. № 52. P. 672–678.
7. *Pharr M., Jakob W., Humphreys G.* Physically Based Rendering: From Theory to Implementation. М.: Morgan Kaufmann, 2016. 1266 p.
8. *Karhu K.* Displacement Mapping. Tik-111.500 Seminar on computer graphics. Telecommunications Software and Multimedia Laboratory. Spring 2002: Rendering high-quality 3D graphics.
9. GitHub – embree/embree: Embree ray tracing kernels repository. <https://github.com/embree/embree>
10. Intel® oneAPI Threading Building Blocks. <https://www.intel.com/content/www/us/en/developer/tools/oneapi/onetbb.html>
11. *Волобой А.Г., Галактионов В.А., Жданов Д.Д.* Технология оптических элементов в компьютерном моделировании оптико-электронных приборов // *Информационные технологии в проектировании и производстве*. 2006. № 3. С. 46–56.
12. Integra® Lumicep. <https://integra.jp/en/products/lumicep>

DUAL REPRESENTATION OF GEOMETRY FOR RAY TRACING ACCELERATION IN OPTICAL SYSTEMS WITH FREEFORM SURFACES

© 2024 D. D. Zhdanov^a, I. S. Potemin^a, A. D. Zhdanov^a

^aSt. Petersburg National Research University of Information Technologies, Mechanics, and Optics, Kronverkskii pr. 49, St. Petersburg, 197101 Russia

This paper explores the possibility of using dual geometry representation to improve the speed of ray tracing and ensure the robustness of light propagation simulations in complex optical systems containing free-form surfaces defined by high-order polynomials (up to order 34) or Jacobi polynomials. An analysis was carried out of traditional methods of representing this geometry both in the form of a triangular mesh and in the form of an analytical expression. The analysis demonstrated the disadvantages of traditional approaches, which consist in the insufficient accuracy of calculating the coordinates of the meeting point of the ray with a triangular mesh, as well as the instability of the results of searching for the hit point of tangent rays with the analytically defined surface when using existing calculation methods. As a result, it was proposed to use a dual representation of the geometry in the form of a rough approximation of the surface by a triangular mesh, which is subsequently used as an initial approximation to find the point where the ray hits the surface specified by the analytical expression. This solution made it possible to significantly speed up the convergence of analytical methods and increase their stability. Moreover, using the Intel® Embree library to quickly find the point of intersection of a ray with a coarse triangular mesh and a vector calculation model to refine the coordinates of the point of intersection of the ray with the geometry represented analytically allowed authors to develop and implement a ray tracing algorithm in an optical system containing surfaces with dual geometry representation. Experiments conducted using the developed and implemented algorithm show the significant acceleration of ray tracing while maintaining computational accuracy and high stability of results. The results were demonstrated by calculating the point and flare spread function for two lenses with free-form surfaces defined by Jacobi polynomials. In addition, for these two lenses, the image formed by an RGB-D object simulating a real scene was calculated.

Keyword: ray tracing, rendering, triangle mesh, free-form surfaces

REFERENCES

1. *Forbes G.W.* Shape specification for axially symmetric optical surfaces // *Opt. Express*. 2007. № 15. P. 5218–5226.
2. *Forbes G.W.* Robust, efficient computational methods for axially symmetric optical aspheres // *Opt. Express*. 2010. № 18. P. 19700–19712.
3. *Thompson K.P., Fournier F., Rolland J.P., Forbes G.W.* The Forbes Polynomial: A more predictable surface for fabricators // *International Optical Design Conference and Optical Fabrication and Testing*. OSA Technical Digest (CD) (Optica Publishing Group, 2010), paper OTuA6.
4. *Feder D.* Optical Calculations with Automatic Computing Machinery // *J. Opt. Soc. Am.* 1951. № 41. P. 630.

5. *Allen W., Snyder J.* Ray Tracing through Uncentered and Aspheric Surfaces // J. Opt. Soc. Am. 1952. № 42. P. 243.
6. *Spencer M.G.H., Murty V.R.K.* General Ray-Tracing Procedure // J. Opt. Soc. Am. 1962. № 52. P. 672–678.
7. *Pharr M., Jakob W., Humphreys G.* Physically Based Rendering: From Theory to Implementation. M.: Morgan Kaufmann, 2016. 1266 p.
8. *Karhu K.* Displacement Mapping. Tik-111.500 Seminar on computer graphics. Telecommunications Software and Multimedia Laboratory. Spring 2002: Rendering high-quality 3D graphics.
9. GitHub — embree/embree: Embree ray tracing kernels repository.
<https://github.com/embree/embree>
10. Intel® oneAPI Threading Building Blocks.
<https://www.intel.com/content/www/us/en/developer/tools/oneapi/onetbb.html>
11. *Voloboi A.G., Galaktionov V.A. and Zhdanov D.D.* Technology of optical elements in computer modeling of optical-electronic devices // Inf. Tekhnol. Proekt. Proizvod. 2006. № 3. P. 46–56.
12. Integra® Lumicept.
<https://integra.jp/en/products/lumicept>