

АНАЛИЗ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ ПОВЫШЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ ПРОЦЕССА РЕНДЕРИНГА

А. В. КИСЕЛЁВ

andrey.kise15@yandex.ru

ООО «РН-БашНИПинефть»

Поступила в редакцию 18 января 2022 г.

Аннотация. Рассматривается проблема актуальности проведения разработок с целью повышения производительности процесса рендеринга, вследствие чего проведен анализ рендеринга как процесса и его распространенности в современном мире. Отражены результаты проведенного обзора известных алгоритмов рендеринга: рассмотрены алгоритм растеризации (метода развертки строк), а также алгоритмы отслеживания лучей, представленные алгоритмами рейкастинга, трассировки лучей и трассировка пути. Приведен анализ результатов исследовательских разработок в области повышения производительности процесса рендеринга, в процессе которого было рассмотрено несколько разработок последних лет, на основе которых был сделан вывод об актуальности дальнейших разработок в данной области.

Ключевые слова: рендеринг; предварительный рендеринг; рендеринг в реальном времени; растеризация; метод развертки строк; отслеживание лучей; трассировка лучей; трассирование путей; методы глобального освещения; вычисление освещенности; адаптивная стратегия рендеринга.

ВВЕДЕНИЕ

В современном мире рендеринг стал неотъемлемой частью большого количества процессов в разных областях. Для примера рендеринг используется во множестве научных исследований и визуализаций, как в прикладных, так и в естественных науках.

В инженерной области процесс рендеринга позволяет изготавливать прототипы для дальнейшего их тестирования и обнаружения проблем. Этап прототипирования представлен практически в любом инженерном проекте (например, механическое или электрическое проектирование) [1].

В настоящее время производительность рендеринга существенно больше того, чем, например, 5 лет назад. Качество рендеринга также намного превосходит то, что могли предоставить технологии предыдущих по-

колений. Благодаря этому рендеринг стал для производителей самым простым и эффективным способом узнать, куда идет продукт и на что он может быть похож в конце концов.

Вопросами рендеринга занимаются ученые как в нашей стране, так и за рубежом. Например, из соотечественников можно привести деятельность Института прикладной информатики им. М. В. Келдыша РАН, а из зарубежных исследователей – Калифорнийский университет, ученые которого регулярно выступают на конференции SIGGRAPH, посвященной текущим проблемам рендеринга.

Однако, современные технологии рендеринга далеки от идеала. Несмотря на широкое распространение, она до сих пор не является совершенной и в ней существует

множество проблем, для которых все еще нет приемлемого или приближенного решения. Большинство из этих проблем связано с необходимостью повысить производительность рендеринга в реальном времени, например, ускорение и повышение фотореалистичности рендеринга теней, отражений, прозрачности (порядок наложения и освещение), проблемы в физическом представлении материалов, представлении геометрии и интеграции с конвейерами ресурсов.

АНАЛИЗ ПРОЦЕССА РЕНДЕРИНГА

Рендеринг – это процесс превращения двухмерных или трехмерных моделей в изображения или анимацию. Полученное в процессе рендеринга изображение или анимация называется рендером.

Обычно, в процессе рендеринга участвует не одна модель, а их совокупность. Это множество моделей составляет сцену, которая описывается определенным языком или структурой данных, и хранится в файле сцены. Внутри файла сцены есть вся необходимая информация для описания сцены, такая как геометрия и текстура моделей, точка обзора и характеристики наблюдателя, элементы освещения. Данные сцены передаются специальному прикладному программному обеспечению для последующей обработки и создания изображения сцены.

В зависимости от того, как быстро должен создаваться рендер, можно выделить две категории рендеринга: предварительный рендеринг и рендеринг в реальном времени.

Предварительный рендеринг: этот метод рендеринга используется в средах, где скорость не имеет значения, а вычисления изображений выполняются с использованием многоядерных центральных процессоров, а не специального графического оборудования. Этот метод рендеринга в основном используется в анимации и визуальных эффектах, где фотореализм должен соответствовать высочайшим стандартам.

Рендеринг в реальном времени: данный метод используется в интерактивной графике и играх, где изображения должны создаваться в быстром темпе. Термин может относиться ко всему, от рендеринга гра-

фического пользовательского интерфейса приложения до анализа изображений в реальном времени [2].

ОБЗОР ИЗВЕСТНЫХ АЛГОРИТМОВ РЕНДЕРИНГА

Среди множества алгоритмов рендеринга можно выделить растеризацию и отслеживание лучей как наиболее распространенные.

Отличие алгоритма *растеризации* (rasterization), или как его еще называют – *метода развертки строк* (scanline rendering), от большинства заключается в том, что видимая поверхность определяется строка за строкой. Для этого все полигоны сначала сортируются по наибольшей координате Y, затем вычисляется каждое пересечение строки развертки с полигонами, при этом список полигонов обновляется, чтобы удалить невидимые области. Активная линия развертки при этом продвигается вниз (рис. 1).

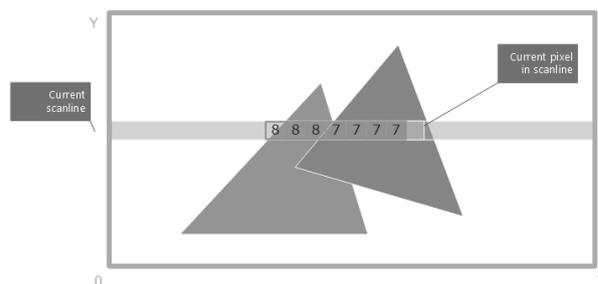


Рис. 1. Метод развертки строк

Для уменьшения количества сравнений между ребрами используется сортировка вершин по нормали к плоскости сканирования. Еще одно преимущество этого метода состоит в том, что нет необходимости переводить координаты всех вершин из основной памяти в рабочую память – только вершины, определяющие ребра, которые пересекают текущую строку сканирования, должны находиться в активной памяти, и каждая вершина считывается только один раз. Основная память часто очень медленная по сравнению со связью между центральным процессором и кэш-памятью, и, таким образом, отказ от повторного доступа к вершинам в основной памяти может обеспечить существенное ускорение.

Алгоритм метода развертки на примере многоугольников можно представить следующим образом [3]:

1. Для каждого многоугольника ставится в соответствие флаг.

2. При перемещении по линии сканирования флаг многоугольника меняет свое значение на противоположное, когда край этого многоугольника пересекается.

3. Далее происходит определение отображаемого многоугольника:

– если ни один из флагов не является истинным, то ничего не отображается;

– если один флаг истинен, используется цвет этого многоугольника;

– если более одного флага истинно, то должен быть определен самый передний многоугольник.

Алгоритмы отслеживания лучей заключается в распространении лучей из точки обзора на сцену для получения цвета отображаемого объекта. Однако трассировка даже небольшого количества лучей занимает слишком много времени и вычислительных ресурсов.

На данный момент можно выделить три сформированных множества алгоритмов, которые являются более эффективными, чем трассировка всех лучей света на сцене.

Рейкастинг (ray casting). На сцене располагается точка наблюдения (камера, глаз). Далее, как и у всех алгоритмов, основанных на трассировании лучей, из точки наблюдения выпускаются лучи, которые при столкновении с объектами сцены определяют цвет пиксела на двумерном экране (рис. 2). Однако, в отличие от алгоритма трассировки лучей или трассировки пути, лучи останавливаются при обнаружении коллизии с объектом или фоном.

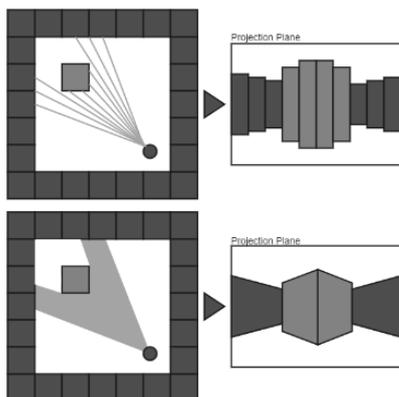


Рис. 2. Метод рейкастинга

Также есть возможность добавления простых оптических эффектов, например эффекта перспективы, когда лучи проецируются под углом.

Традиционный алгоритм рейкастинга включает в себя два этапа [4]:

1. Присвоение всем дискретным трехмерным вершинам оптических свойств, таких как цвет и непрозрачность, в соответствии с их значением серого.

2. Применение процесса выборки и компоновки. Для каждого пиксела выходного изображения последовательно выполнить следующие действия:

– провести луч через объект;

– выбрать цвет C_i и непрозрачность A_i в каждой точке выборки вдоль луча;

– установить цвет выходного пикселя в соответствии с формулой 1.

$$C_{out} = \sum_{i=0}^{n-1} C(i) \prod_{j=0}^{i-1} 1 - A(j). \quad (1)$$

Трассировка лучей (ray tracing). В отличие от алгоритма рейкастинга, луч не останавливается при столкновении с объектом. Вместо этого он разделяется на три луча компонента. Эти три луча также влияют на результирующий цвет пикселя: отраженный, теневой и преломленный. Регулируя количество компонентов, можно контролировать глубину трассировки и качество изображения (рис. 3).

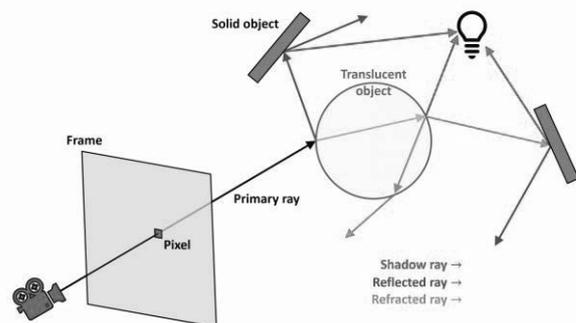


Рис. 3. Метод трассировки лучей

Данный метод позволяет создавать очень фотореалистичные изображения, но его использование влечет за собой очень большую трату ресурсов, что негативно сказывается на времени рендеринга.

Вариативность алгоритма повлекла за собой создание множества его реализа-

ций, однако первая процедура трассировки лучей в том виде, в котором она привычна в настоящее время, описана Дж. Уиттелом. Она и будет рассмотрена в статье как первоисточник алгоритма (рис. 4).

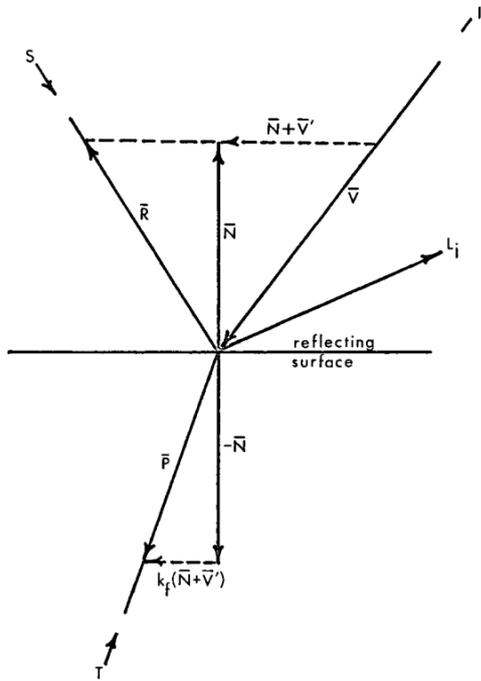


Рис. 4. Формула Уиттела

Уиттел предложил следующую формулу для нахождения яркости рассматриваемого пикселя:

$$I = I_a + k_d \sum_{j=1}^{j=ls} (\bar{N} \times \bar{L}_j) + k_s S + k_t T. \quad (2)$$

Здесь V, R, P – направления распространения света; l – отраженная интенсивность; I_a – отражение от окружающего света; k_d – постоянная диффузного отражения; k_s – коэффициент зеркального отражения; N – нормаль к единице поверхности; L_j – вектор в направлении j -го источника света; S – интенсивность падающего света со стороны вектора R ; k_t – коэффициент передачи; T – интенсивность света по направлению от вектора P .

При этом направление R определяется простым правилом, согласно которому угол отражения должен быть равен углу падения. Точно так же направление P проходящего света должно подчиняться закону Снеллиуса [5].

$$\bar{V}' = \frac{\bar{V}}{|\bar{V} \times \bar{N}|}, \quad (3)$$

$$\bar{R} = \bar{V}' + 2\bar{N}, \quad (4)$$

$$\bar{P} = k_j(\bar{N} + \bar{V}') - \bar{N}, \quad (5)$$

$$k_j = (k_n^2 |\bar{V}'|^2 - |\bar{V}' + \bar{N}|^2)^{-\frac{1}{2}}, \quad (6)$$

где k_n – показатель преломления.

Трассировка пути (path tracing). Является наибольшее приближенной к физическим законам распространения света версией алгоритма трассировки лучей, вследствие чего также является и самым ресурсоемким алгоритмом.

В отличие от трассировки лучей, в данном алгоритме луч отскакивает от предметов до тех пор, пока полностью не поглотится или рассеется, вследствие чего алгоритм позволяет имитировать множество эффектов, такие как мягкие тени, глубина резкости, размытие при движении, каустика, окружающая окклюзия и не прямое освещение, в то время как остальные алгоритмы должны добавлять эти эффекты отдельно. Расширенная версия алгоритма реализована путем объемной трассировки пути, которая учитывает светорассеяние сцены (рис. 5).

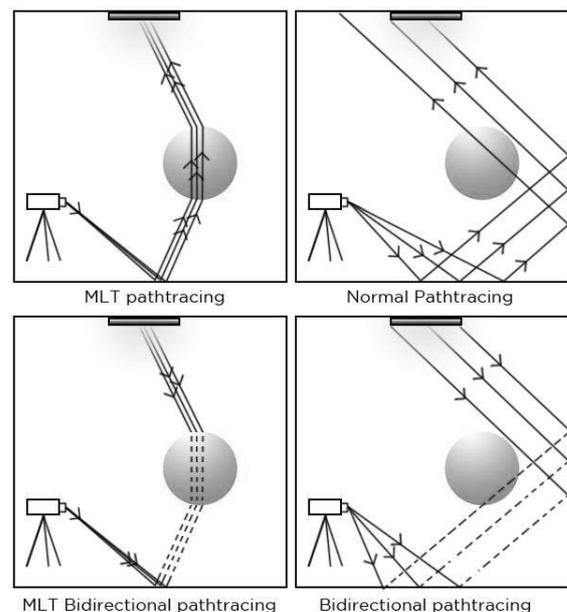


Рис. 5. Методы трассировки пути

Благодаря тому, что алгоритм является очень точным и непредвзятым, его часто используют для рендеринга эталонных изображений, которые в дальнейшем будут использоваться в качестве тестовых для других алгоритмов.

Алгоритм трассировки пути пытается решить уравнение рендеринга – интегральное уравнение, которое определяет количество светового излучения в определенном направлении как сумму собственного и отраженного излучения.

Функцию рендеринга для трассировки путей можно представить следующей формулой [6]:

$$L(P \rightarrow D_v) = L_e(P \rightarrow D_v) + \int_A F_s(D_v, D_i) |\cos \theta| L(Y_i \rightarrow D_i) dD_i. \quad (7)$$

Здесь $L(P \rightarrow D_v)$ – яркость (сила света), исходящая от точки P на поверхности в направлении D_v ; $L_e(P \rightarrow D_v)$ – самоизлучаемая яркость от точки P в направлении D_v (0, если точка P не является источником света); $\int F_s(D_v, D_i) |\cos \theta| L(Y_i \rightarrow D_i) dD_i$ – отраженный свет. Здесь складывается весь свет, попадающий в точку P со всех направлений, модулированный вероятностью того, что он рассеивается в направлении D_v (на основе двухлучевой функции отражательной способности).

АНАЛИЗ РЕЗУЛЬТАТОВ ИЗВЕСТНЫХ НАУЧНЫХ ИССЛЕДОВАНИЙ

В нашей стране также существует множество исследователей, занимающихся проблемой повышения эффективности процесса рендеринга.

Например, исследование Д. К. Боголепова в области интерактивного синтеза изображений. В своей диссертационной работе «Методы глобального освещения для интерактивного синтеза изображений сложных сцен на графических процессорах» он представил новую эффективную ускоряющую структуру, которая подходит как для статических, так и для динамических сцен и основанная на иерархии ограничивающих объемов, а также новую структуру хранения и обработки трехмерных сцен со специфичным двухуровневым представлением моде-

ли, где на верхнем уровне модель работает как граф сцены, а на нижнем – сериализует данные для обработки на GPU.

Также им был представлен новый программный конвейер трассировки лучей, реализующий типовые блоки (операции) алгоритмов визуализации, основанных на бросании лучей. Его отличие состоит в универсальности методов глобального освещения, а также ориентированности на параллельные вычислительные архитектуры.

Им были также представлены построенная на основе концепции атомарных материалов Монте-Карло, подсистема материалов, обеспечивающая компактность описания и эффективности обработки на GPU и алгоритм усеченной двунаправленной трассировки путей, отличающийся фиксированным объемом потребляемой памяти при обработке путей любой длины, а также полным исключением фазы соединения путей [7].

Эффективный алгоритм вычисления освещенности предложил А. В. Мальцев в своей диссертационной работе, целью которой является обеспечение высокореалистичной визуализации трехмерных виртуальных сцен в реальном режиме времени.

В результате его работы было разработано три новых алгоритма:

1. Метод расчета освещенности сцен при использовании шейдерной визуализации. Созданы алгоритмы, которые моделируют свойства поверхностей объектов в реальном времени, которые основаны на этом методе.

2. Метод статической оптимизации шейдерных программ, которые написаны на низкоуровневых ЯП, используемых для расчетов попиксельного освещения.

3. Метод цветowych карт, который позволил ускорить процесс обработки информации при расчете освещенности трассировкой первичных лучей.

Им же была предложена новая эффективная технология создания изображений высокополигональных сцен в реальном времени с помощью метода совмещенной визуализации с применением цветowych карт, который использует современные компьютерные методы обработки графической информации [8].

Еще один способ ускорения рендеринга сцен в реальном времени в своей диссертационной работе предложил А. Д. Жданов.

Разработку автор проводил с целью повышения эффективности алгоритмов реалистичной визуализации моделей трехмерных сцен методами стохастической трассировки лучей на современных многопроцессорных рабочих станциях. С этой целью он разработал три новых метода: метод использования прогрессивных обратных фотонных карт для учета каустической и вторичной составляющих освещения; метод, который использует отдельные карты, чтобы рассчитывать компонент яркости каустического освещения, которое сформировано световыми лучами, и расчета компонент яркости вторичного освещения, которое формирует лучи, идущие от наблюдателя и метод трехуровневой параллельной обработки данных модели сцены, сочетающий синхронные, частично-синхронные и асинхронные вычисления, для расчета физически корректного значения яркости изображения методами обратных фотонных карт.

Эти методы позволили в несколько раз ускорить процесс фотореалистичной реализации сцен, а также уменьшить количество памяти, которая необходима для хранения и обработки фотонных карт [9].

Также стоит привести в пример разработанную В. И. Гонахьяном адаптивную стратегию рендеринга динамических трехмерных сцен.

Целью его диссертационной работы было исследование и разработка эффективной стратегии рендеринга динамических трехмерных сцен, сочетающей в себе методы удаления невидимых объектов, отложенные аппаратные проверки видимости, кэширование командных буферов, а также реализующей адаптивный выбор методов и техник для ускорения процессов рендеринга и обеспечения возможностей разработки перспективных интерактивных графических приложений.

Как итог, предложенная им модель производительности графического конвейера для однопроходной схемы рендеринга динамических трехмерных сцен позволила оценивать требуемые ресурсы (время обра-

ботки и передачи графических данных, объем основной и графической памяти) в зависимости от применяемых базовых методов и характеристик отображаемой сцены, а созданная адаптивная стратегия рендеринга динамических трехмерных сцен, реализующая выбор и настройку базовых методов удаления невидимых объектов, техник отложенных аппаратных проверок видимости и кэширования командных буферов на основе модели производительности графического конвейера непосредственно в процессе отображения сцены повысила эффективность рендеринга динамических сцен.

Также в своей работе он предложил метод генерации динамических трехмерных сцен, который позволяет синтезировать семейства сцен с разными характеристиками, определяемыми количеством и сложностью индивидуальных объектов, пространственной разреженностью сцены, интенсивностью событий и их пространственно-временной когерентностью [10].

Наличие множества свежих научных работ в области рендеринга подтверждает факт того, что разработки в направлении повышения эффективности рендеринга актуальны в настоящее время и ведутся в различных направлениях.

ЗАКЛЮЧЕНИЕ

В данной статье был исследован вопрос актуальности исследований процесса рендеринга, а также проблем, существующих на данный момент в представленной области. Был проведен анализ процесса рендеринга, а также его видов. Далее были приведены основные алгоритмы рендеринга, используемые в разработке программного обеспечения на данный момент. Также были представлены некоторые вариации данных алгоритмов. В завершении статьи был проведен анализ результатов известных научных исследований и сделан вывод об актуальности дальнейшей разработки в данном направлении.

СПИСОК ЛИТЕРАТУРЫ

1. **Schechter S.** 3D Rendering Examples. [Электронный ресурс]. URL: <https://www.marxentlabs.com/3d-rendering-examples/> (дата обращения 24.10.21). [S. Schechter (2021,

Oct. 24). 3D Rendering Examples. [Online]. Available: <https://www.marxentlabs.com/3d-rendering-examples/>]

2. **What** is the difference between pre-rendering and real-time rendering? [Электронный ресурс]. URL: <https://www.foxrenderfarm.com/share/what-is-the-difference-between-prerendering-and-real-time-rendering> (дата обращения 24.10.21). [What is the difference between pre-rendering and real-time rendering? (2021, Oct. 24). [Online]. Available: <https://www.foxrenderfarm.com/share/what-is-the-difference-between-pre-rendering-and-real-time-rendering>]

3. **Computer** graphics: principles and practice (3rd ed.) / J. F. Hughes, et al. Boston, MA, USA: Addison-Wesley Professional, 2013. 1264 p.

4. **Plane-Based** Sampling for Ray Casting Algorithm in Sequential Medical Images / Lin L., et al. // Computational and Mathematical Methods in Medicine. 2013. Vol. 2013. Article number 874517.

5. **Whitted J. T.** An improved illumination model for shaded display // Proceedings of the 6th annual conference on Computer Graphics and Interactive Techniques, (Chicago, Ill, USA, Aug. 8–10 1979). New York, USA: Association for Computing Machinery, 1979.

6. **Kajiya J. T.** The rendering equation // Proceedings of the 13th annual conference on Computer Graphics and interactive techniques, (Boston, MA, USA, July 31–Aug. 4 1989). New York, USA: Association for Computing Machinery, 1989.

7. **Боголепов Д. А.** Методы глобального освещения для интерактивного синтеза изображений сложных сцен на графических процессорах: автореф. дис. ... канд. тех. наук: 05.13.17. Нижний Новгород: НГТУ им. Р. Е. Алексеева, 2013. 20 с. [D. A. Bogolepov, *Global Illumination Methods for Interactive Image Synthesis of Complex Scenes on GPUs*: Cand. Diss. Abstr., (in Russian). Nizhny Novgorod: NGTU im. R. E. Alekseeva, 2013.]

8. **Мальцев А. В.** Методы и алгоритмы эффективного вычисления освещенности трехмерных виртуальных сцен в реальном режиме времени: автореф. дис. ... канд. физ.-мат. наук: 05.13.01. М., 2011. 18 с. [A. V. Maltsev, *Methods and algorithms for efficient calculation of illumination of three-dimensional virtual scenes in real time*: Cand. Diss. Abstr., (in Russian). Moscow: Nauch.-issled. in-t sistemnykh issledovaniy RAN, 2011.]

9. **Жданов А. Д.** Методы и алгоритмы эффективной визуализации моделей трехмерных сцен на многопроцессорных рабочих станциях: автореф. дис. ... канд. техн. наук: 05.13.11. СПб., 2020. [A. D. Zhdanov, *Methods and algorithms for efficient visualization of 3D scene models on multi-processor workstations*: Cand. Diss. Abstr., (in Russian). Saint Petersburg, 2020.]

10. **Гонахчян В. И.** Адаптивная стратегия рендеринга динамических трехмерных сцен: автореф. дис. ... канд. тех. наук: 05.13.11. М., 2021. 20 с. [V. I. Gonakhchyan, *Adaptive rendering strategy for dynamic 3D scenes*: Cand. Diss. Abstr., (in Russian). Moscow, 2021.]

ОБ АВТОРЕ

КИСЕЛЁВ Андрей Владимирович, вед. специалист, ООО «РН-БашНИПНефть».

METADATA

Title: Analysis of technologies to increase the performance of the rendering process.

Author: A. V. Kiselyov

Affiliation: RN-BashNIPneft

Email: andrey.kise15@yandex.ru

Language: Russian.

Source: SIIT (scientific journal of Ufa State Aviation Technical University), vol. 4, no. 1 (8), pp. 56-62, 2022. ISSN 2686-7044 (Online), ISSN 2658-5014 (Print).

Abstract: This article discusses the problem of the relevance of development in order to improve the performance of the rendering process, as a result of which an analysis of rendering as a process and its prevalence in the modern world is carried out. The presented article reflects the results of a review of known rendering algorithms: the rasterization algorithm (scanline rendering), as well as ray tracing algorithms, represented by ray casting algorithms, ray tracing and path tracing, are considered. An analysis of the results of research developments in the field of improving the performance of the rendering process is given, during which several developments of recent years were considered, on the basis of which a conclusion was made about the relevance of further developments in this area.

Key words: rendering; pre-rendering; real-time rendering; rasterization; scanline rendering; ray tracing; path tracing; global illumination methods; illumination calculation; adaptive rendering strategy.

About author:

KISELYOV, Andrey Vladimirovich, Leading Specialist, RN-BashNIPneft.