

УДК 004.925

DOI: [10.26102/2310-6018/2026.57.6.016](https://doi.org/10.26102/2310-6018/2026.57.6.016)

Метод адаптивной градации при настройке отсечения невидимых объектов

В.Г. Чёрный✉, М.В. Болсуновская

*Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация*

Резюме. В работе рассматривается применение метода адаптивной градации объектов на основе их геометрической сложности для автоматизации формирования набора загораживающих объектов в 3D-сценах, разрабатываемых с помощью графических движков. Предложенный подход предусматривает их классификацию по уровню геометрической сложности с последующим учетом этой характеристики при принятии решения об их использовании в качестве загораживающих. Метод реализовывался как расширение подхода для формирования набора окклюдеров, основанного на анализе пересечений лучей с объектами. Для ускорения анализа используется иерархия ограничивающих объемов. Подход подразумевает, что загораживающим объектом считается тот объект, попадания лучей в который превышают заданный параметрический порог. В рамках предлагаемого метода подразумевается расширение подхода путем использования адаптивного порога, вычисляемого на основе классификации объектов по числу содержащихся в них треугольников. Экспериментальная проверка выполнялась в движке Unity. Использовались две 3D-сцены, заполненные базовыми встроенными примитивами и низкополигональными моделями в разном количестве. Для небольших сцен с примитивами различие между базовым и адаптивным порогами оказывается незначительным, однако в более сложных случаях метод позволяет сузить выборку. Эксперименты демонстрируют, что применение метода обеспечивает более избирательную классификацию окклюдеров в сценах с высокой плотностью объектов, а также выявляет значимость дополнительных факторов, включая размеры и пространственное расположение моделей.

Ключевые слова: Unity, Occlusion Culling, статический анализ, оптимизация рендеринга 3D-сцен, инструменты автоматизации.

Для цитирования: Чёрный В.Г., Болсуновская М.В. Метод адаптивной градации при настройке отсечения невидимых объектов. *Моделирование, оптимизация и информационные технологии.* 2026;14(6). URL: <https://moitvvt.ru/ru/journal/article?id=2303> DOI: 10.26102/2310-6018/2026.57.6.016

Adaptive object gradation method for occlusion culling configuration

V.G. Chernyi✉, M.V. Bolsunovskaya

*Peter the Great St. Petersburg Polytechnic University, Saint-Petersburg,
the Russian Federation*

Abstract. The paper considers the application of an adaptive object gradation method based on geometric complexity for automating the formation of a set of occluders in 3D scenes developed using graphics engines. The proposed approach involves classifying objects according to their level of geometric complexity and incorporating this characteristic into the decision-making process for their use as occluders. The method is implemented as an extension of an approach for occluder selection based on ray-object intersection analysis. A bounding volume hierarchy is used to accelerate the analysis. The approach assumes that an object is considered an occluder if the number of ray hits exceeds a predefined threshold. The proposed method extends this approach by introducing an adaptive threshold calculated

based on the classification of objects according to the number of triangles they contain. Experimental evaluation was performed in the Unity engine using two 3D scenes populated with basic primitives and low-poly models in different quantities. For simple scenes, the difference between the base and adaptive thresholds is insignificant; however, in more complex cases, the method allows narrowing the set of selected occluders. The results demonstrate that the method provides a more selective classification of occluders in scenes with high object density and highlights the importance of additional factors, including object size and spatial distribution.

Keywords: Unity, Occlusion Culling, static analysis, 3D scene rendering optimization, automation tools.

For citation: Chernyi V.G., Bolsunovskaya M.V. Adaptive object gradation method for occlusion culling configuration. *Modeling, Optimization and Information Technology*. 2026;14(6). (In Russ.). URL: <https://moitvivr.ru/ru/journal/article?id=2303> DOI: 10.26102/2310-6018/2026.57.6.016

Введение

С помощью современных 3D-движков, таких как Unity и Unreal Engine осуществляется разработка интерактивных приложений, промышленных и образовательных симуляторов [1, 2]. Отдельные разработки позволяют имитировать различные предметные области и проводить исследования на основе визуализированных данных [3]. Развитие графических технологий и вычислительных возможностей сопровождается ростом сложности проектов и содержащихся в них 3D-сцен, что напрямую влияет на показатели производительности, такие как частота кадров и задержка [4]. Необходимость удерживать баланс между визуальным качеством и производительностью приложений сохраняет актуальность задач оптимизации рендеринга.

В 3D-движках есть встроенные инструменты оптимизации, которые реализуют наиболее распространенные подходы. Среди них можно выделить отсечение невидимых объектов (Occlusion Culling). Метод позволяет исключать из процесса рендеринга модели, которые не попадают в поле зрения пользователя из-за перекрытия другими объектами. Существует большое количество современных исследований по задачам определения видимости и отсечения невидимых объектов, многие из которых ориентированы на работу в реальном времени [5, 6]. Применяются различные подходы к анализу видимости объектов, к примеру, в основе некоторых решений лежит затенение объектов (ambient occlusion) [7]. Кроме того, рассматриваются подходы по оптимизации проектов для web-платформ [8].

В контексте практического применения встроенных инструментов оптимизации в движках возникает отдельная задача предварительной настройки сцен и проектов. К примеру, для применения Occlusion Culling в движке Unity требуется предварительная пометка загораживающих объектов (окклюдеров), которая осуществляется вручную. При работе с крупными и разнородными 3D-сценами это может приводить к временным затратам, что снижает эффективность решения.

Кроме временных затрат, на эффективность методов влияет и корректность предварительной настройки. В случае Occlusion Culling на качество итоговой оптимизации серьезное воздействие оказывают выделенные под нее объекты. Не всегда загораживающие объекты могут быть помечены обоснованно, особенно если речь идет о больших сценах, где помечены были редко встречающиеся преграды. Важно и количество помеченных объектов: в зависимости от сцены, слишком малый объем может сильно ограничить эффективность, а избыточное количество окклюдеров может критически увеличить вычислительные затраты, обратив весь эффект оптимизации.

В прикладных условиях, на сценах, в которых используются разнородные 3D-модели разной степени детализации и сложности, данный вопрос становится особенно

заметен в связи с тем, что ручная настройка оказывается слабо формализованной и опирающейся на субъективную оценку разработчика. Так, воспроизводимость настроек и перенос подходов к ним между сценами или проектами также затрудняется.

Несмотря на большое количество исследований по алгоритмам отсека невидимых объектов в реальном времени, вопросы автоматизации предварительного этапа их настройки, или же выявления потенциальных окклюдеров на уже сформированной сцене остаются недостаточно проработанными. При этом данный этап оказывает большое влияние на эффективность оптимизации с помощью Occlusion Culling, так как бесконтрольная пометка объектов может привести и к негативному для рендеринга результату [9].

В рамках предыдущих исследований был предложен подход к автоматизации подбора объектов для оптимизации рендеринга в Unity, использующий запуск лучей и иерархию ограничивающих объемов для оценки моделей в качестве загораживающих [10]. Следующим этапом исследования, расширяющим подход, является проработка вопроса влияния на анализ таких характеристик как геометрическая сложность объектов.

Таким образом, задача разработки методов автоматизированного анализа 3D-сцен, позволяющих определять загораживающие объекты на основе их характеристик, является актуальной.

Целью работы является разработка и исследование метода адаптивной градации объектов на основе их геометрической сложности в рамках автоматического определения окклюдеров, основанного на анализе пересечений лучей с использованием иерархии ограничивающих объемов.

Материалы и методы

В рамках работы задача определения потенциальных окклюдеров рассматривается как задача классификации объектов на 3D-сцене по степени их значимости для последующего применения к ним Occlusion Culling. Значимым должен считаться тот объект, который способен быть преградой для пользовательского обзора по отношению к другим элементам, отображающимся на сцене. В изначальном подходе, который расширяется в работе, оценка объекта проводилась на основе подсчета числа попаданий лучей, которые им перехватывались. В контексте анализа число попаданий становится характеристикой, которая отражает вероятные пространственные взаимодействия объектов с соседними и позволяет формализовать их выбор в качестве загораживающих. Более частые попадания лучей повышают вероятность пометки объекта.

При этом использование только одной метрики в виде числа попаданий не позволяет учитывать другие различия между объектами: их геометрическую сложность и общее положение в пространстве. Это ограничение может приводить к ситуациям, когда одинаково будут оцениваться разные по качеству и влиянию на производительность объекты со схожим количеством попаданий. На основе этого ограничения было решено рассмотреть расширение подхода на этапе принятия решения о пометке потенциального окклюдера.

Предложенный метод был реализован в виде экспериментального алгоритма в среде разработки Unity, представляющего собой отдельное окно во встроенном редакторе. Пользовательский интерфейс реализован с помощью класса EditorWindow стандартной библиотеки движка. При настройке экспериментов задаются параметры, среди которых количество генерируемых лучей и точек, из которых они будут выпускаться. Также возможно подключение визуализации при обработке сцены –

отрисовка лучей и других вспомогательных данных. Результаты анализа записываются в JSON-файл.

Работа алгоритма включает несколько этапов. На первом этапе подбираются 3D-объекты для анализа и вокруг них генерируются источники лучей. На втором этапе осуществляется запуск лучей в направлении целевых объектов, при котором подсчитываются попадания в возможные преграды. На заключительном этапе происходит подсчет попаданий и пометка объектов, превысивших порог, флагом Occluder Static. Дополнительно алгоритм может запустить встроенный в движок процесс настройки отсечения невидимых объектов на основе помеченных окклюдеров.

Для ускорения вычислений при пуске лучей используется иерархия ограничивающих объемов (BVH) [11, 12]. Данная структура формируется при запуске анализа и представляет собой дерево из блоков в пространстве, в котором узлы содержат информацию об объектах в соответствующих блоках. Использование BVH позволяет значительно сократить количество проверок пересечений и тем самым повышает эффективность обработки сцены.

Для определения потенциальных окклюдеров используется базовый порог количества попаданий лучей, определяемый как доля от общего числа выпущенных лучей:

$$T_{base} = N_{rays} \cdot N_{sources} \cdot a, \quad (1)$$

где N_{rays} – количество запущенных лучей; $N_{sources}$ – количество источников лучей; a – коэффициент, определяющий долю попаданий, необходимую для признания объекта в качестве загораживающего.

Текущая реализация подхода обладает единым настраиваемым вручную порогом попаданий для всей сцены, что ограничивает эффективность для разнородных сцен, требуя частых изменений. Также не анализируются свойства объектов, такие как их геометрическая сложность, размеры и пространственное положение.

На основе перечисленных ограничений предлагается модификация подхода, основанная на введении метода градации объектов на основе их геометрической сложности и применении его результатов при итоговом принятии решения.

В рамках предложенного метода объекты разделяются на три группы: простые, средние и сложные. Соответствие определенной группе вычисляется на основе количества треугольников, которые содержит объект. Пороговые значения для отнесения к каждой группе вычисляются автоматически, на основе статического анализа распределения количества треугольников для всех объектов на сцене. Все объекты сортируются по числу треугольников, после чего находятся первый и третий квартили (Q1 и Q3): Q1 (нижние 25 % объектов) – граница между простыми и средними; Q3 (верхние 25 %) – граница между средними и сложными. Таким образом, метод не зависит от содержания сцены и не требует ручной настройки порогов.

Количество треугольников, составляющих модель, используется в качестве критерия градации из-за своей информативности и доступности. На основе геометрической сложности можно сформировать представление о создаваемой объектом нагрузке на рендеринг. Данная характеристика не требует особых вычислений и легко получается с помощью средств 3D-движков, что позволяет не усложнять алгоритмы анализа.

Выбор трех групп обусловлен сохранением баланса между наглядностью для простоты настройки и возможностью качественно дифференцировать анализируемые объекты. Использование двух групп привело бы к сценариям грубого разделения объектов, с потенциально высоким количеством пограничных случаев. С другой

стороны, чрезмерный набор групп усложнил бы рассмотрение результатов, не предлагая большей наглядности.

При использовании градации логика определения окклюдеров изменяется. Для каждой из групп используются модификаторы, влияющие на вычисление итогового порога при принятии решения. Для оптимизации вычислительных затрат в ходе отсека более предпочтительными окклюдерами будут геометрически простые объекты, поэтому для них используется понижающий модификатор. Для сложных объектов, наоборот, применяется повышающее значение, которое отражает меньшую вероятность их использования в качестве окклюдеров.

Таким образом, при использовании градации, итоговое принятие решения по поводу пометки объекта в качестве окклюдера осуществляется на основе заданного порога попаданий лучей, умноженного на модификатор, определяемый на основе группы, к которой относится объект:

$$T_{adapt}(o) = T_{base} \cdot M(c(o)), \quad (2)$$

где o – анализируемый объект; $c(o)$ – функция классификации объекта по его геометрической сложности; $M(c(o))$ – модификатор порога на основе его группы.

После вычисления адаптивного порога для каждого объекта принимается решение о его пометке в качестве окклюдера. Объект помечается как загораживающий, если количество попавших в него лучей превышает адаптивный порог:

$$H(o) \geq T_{adapt}(o), \quad (3)$$

где $H(o)$ – число лучей, попавших в объект; $T_{adapt}(o)$ – адаптивный порог попаданий.

Предложенный подход позволяет учитывать характеристики объектов при принятии решения и делает настройку алгоритма более гибкой. Также задается основа для дальнейшего расширения путем подключения иных характеристик, определяющих пространственное положение и другие свойства объекта в 3D-пространстве.

В результате, с учетом расширений на основе предложенного подхода, основными параметрами, используемыми в ходе анализа, являются количество генерируемых лучей, число их источников, базовый порог попаданий и модификаторы, формирующие адаптивный порог.

Увеличение числа лучей может позволить получить не только более устойчивую оценку для каждого объекта, но и повысить вычислительную сложность. Рост источников расширяет пространство для запуска лучей, что сильнее влияет на результаты анализа, так как большее покрытие зон позволяет более точно оценить потенциал объекта в качестве преграды обзора. Базовый порог позволяет определить степень попаданий, которая на той или иной сцене будет считаться критической. Слишком низкий порог может привести к пометке множества объектов, даже если они в контексте сцены не будут являться серьезными преградами. Значительное увеличение порога наоборот сделает анализ более избирательным, что может сократить количество помеченных окклюдеров. С учетом разнообразия разрабатываемых в прикладных условиях 3D-сцен, данные параметры должны определяться вручную под контекст анализируемого пространства.

Реализация метода градации выполняется как расширение программы для автоматизации подбора окклюдеров в Unity и использует стандартную библиотеку движка [12]. При обходе каждого из объектов вычисляется количество треугольников, из которых он состоит. На основе их количества происходит распределение объектов по трем группам.

Результаты

На Рисунке 1 демонстрируется интерфейс реализации подхода в движке Unity и параметры, которые использовались в ходе экспериментов.

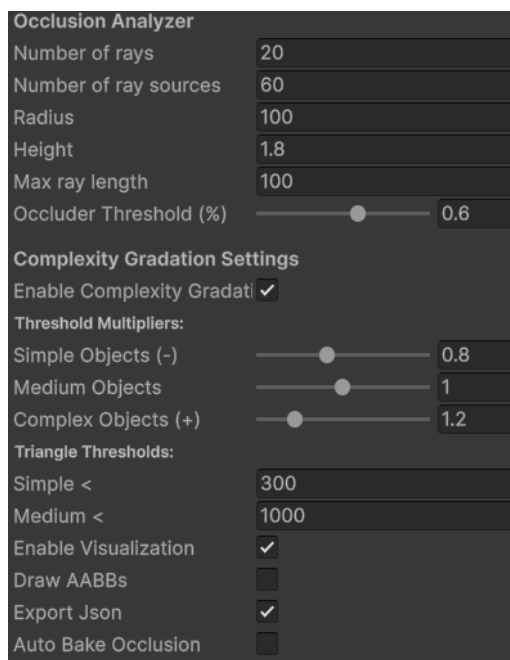


Рисунок 1 – Интерфейс имплементации в Unity
Figure 1 – Unity implementation interface

Проверка градации осуществлялась на двух тестовых сценах, отличающихся по геометрической сложности содержащихся на них объектов и плотности их расположения.

Первую тестовую сцену можно увидеть на Рисунке 2. На ней использовался набор из 40 примитивных объектов, предлагаемых движком и отличающихся низкой геометрической сложностью. Среди них были расположены три явно выраженных загораживающих объекта, представляющих собой большие вытянутые плоскости (стены).

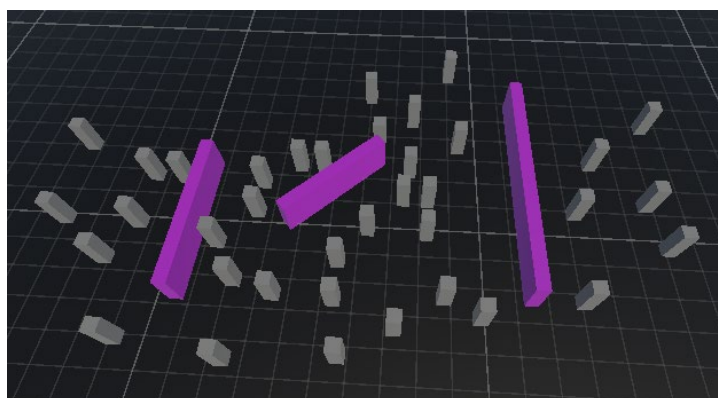


Рисунок 2 – Сцена с примитивными объектами
Figure 2 – Scene with primitive objects

Всего было расположено 40 примитивных объектов, которые должны относиться к группе с низкой сложностью. При значении базового порога 0,6 и с учетом

понижающего модификатора для простых объектов, были помечены исключительно очевидные окклюдеры (3 из 40 объектов). Данный результат демонстрирует корректную работу метода в условиях, когда на сцене содержится небольшое число явно значимых преград для обзора.

Снижение базового порога до 0,2 привело к увеличению количества помеченных объектов. Окклюдерами оказалось 42,5 % моделей от общего количества на сцене. Это свидетельствует о снижении точности градации при небольших значениях порога и избыточности классификации в рамках сцен с очень низкой геометрической сложностью, когда достаточно данных предоставляется на основе анализа количества попаданий лучей.

Для оценки поведения метода в более комплексных условиях стоит рассмотреть сцену со сложным расположением объектов, где шансы перекрытий еще выше. Вторая сцена представлена на Рисунке 3. В рамках более сложного случая было сформировано пространство, содержащее 1000 низкополигональных моделей зданий. Объекты расположены случайным образом с высокой плотностью, и в зависимости от положения камеры почти любой из них может быть загораживающим. При этом пометка всех будет крайне неэффективной и увеличит потенциальные вычислительные затраты при подключении Occlusion Culling.

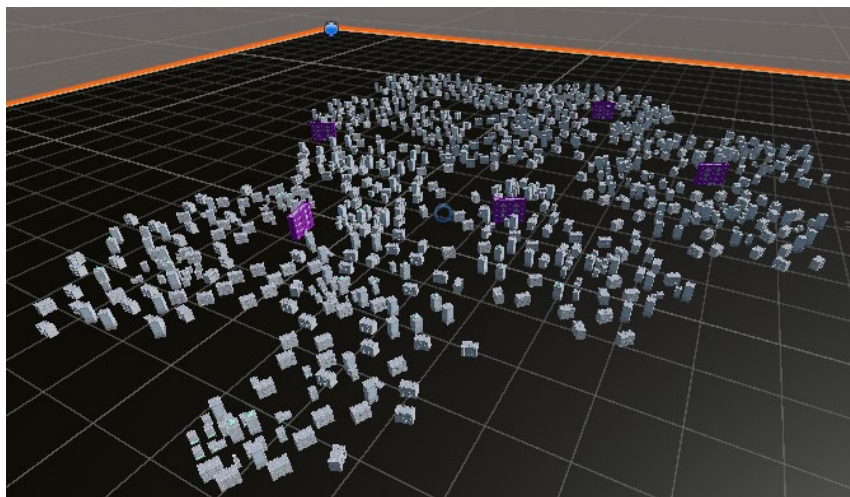


Рисунок 3 – Сцена с низкополигональными объектами
Figure 3 – Scene with low-poly objects

Результаты анализа второй сцены приведены в Таблице 1. При низком значении порога (0,2) использование градации не оказало значительного влияния на итоговое количество найденных окклюдеров, что может быть обусловлено плотностью расположения моделей и их взаимным перекрытием.

С увеличенным порогом (0,6) результат оказался более выраженным: количество окклюдеров при отключенной градации выше, чем при ее использовании.

Таблица 1 – Результаты анализа сцены с низкополигональными объектами
Table 1 – Low-poly objects scene analysis results

Порог	Статус градации	Найденные окклюдеры
0,2	Выкл.	767
	Вкл.	769
0,6	Выкл.	397
	Вкл.	315

Согласно информации в Таблице 2, накладные расходы предложенного метода адаптивной градации составили 1,1 мс при общем времени анализа сцены 14 059 мс для сцены с 1000 объектов, что не оказывает критического влияния на длительность выполнения. Таким образом, вычислительная стоимость расширения не оказывает значимого влияния на общую производительность предварительного анализа сцены.

Таблица 2 – Время работы метода в зависимости от количества треугольников на сцене
Table 2 – Method execution time vs. scene triangle count

Кол-во треугольников	Кол-во объектов	Общее время анализа (м/с)	Время выполнения градации (м/с)
480	40	268	0,037
1 479 616	1000	14059	1,128

Обсуждение

Полученные результаты демонстрируют, что эффективность предложенного метода зависит от используемых в рамках анализа характеристик объектов, общего состояния сцены и заданных параметров анализа. Основная логика для выявления загромождающих объектов все еще зависит от запуска лучей. Так, выполненные расширения не заменяют изначальный подход, а позволяют углубить базовый анализ в случаях, когда геометрическая сложность является ключевым фактором для принятия решений или сцена является очень неоднородной и требует учета множества данных.

Сцены, обладающие небольшим количеством простых или примитивных объектов, одинаково эффективно анализируются как с базовым порогом, так и с адаптивным, вычисленным на основе градации. Так, можно сделать вывод об избыточности применения дополнительных модификаторов при анализе подобных пространств. Этот результат объясняется тем, что в случаях, когда большинство объектов обладает схожими характеристиками, полученные пороги не приводят к существенному изменению итогового порога. Можно сделать вывод о важности параметров генерации лучей и базового порога для сцен, обладающих крайне низким разнообразием объектов.

На сцене из 1000 объектов с высокой плотностью их расположения градация позволила убрать из окклюдеров часть более сложных объектов, что может быть важно для сцен, содержащих большое количество перекрытий. При заданных параметрах значительное количество моделей оказалось отнесенным к окклюдерам, что также можно объяснить высокой вероятностью пересечений лучей. В контексте сцены с плотным и случайным расположением объектов высокий порог попаданий также оказался значительно важнее.

В результате, в сцене с высокой плотностью, где преобладают сложные объекты, они же и оказались в большинстве своем загромождающими, что является ожидаемым поведением. При этом комбинация из градации и порога попаданий выше среднего снизила количество определенных окклюдеров, относительно того же решения с выключенной градацией. При большом количестве объектов важно не только подобрать потенциальные преграды, но и не допустить их чрезмерной пометки, что чревато потерей эффективности при дальнейшем применении инструментов оптимизации. В таком контексте градация по геометрической сложности является дополнительным механизмом ограничения, который позволяет снизить число кандидатов при анализе нагруженных сцен. Кроме того, метод не вызывает серьезного увеличения вычислительной нагрузки в ходе обработки сцены, оставаясь легковесным.

Заключение

В работе предложен метод адаптивной градации объектов на основе их геометрической сложности для автоматизации процесса определения окклюдеров в 3D-сценах.

Проведенные эксперименты демонстрируют, что использование метода позволяет осуществлять более избирательный подбор окклюдеров на сценах с высокой плотностью расположения загораживающих друг друга объектов. Это, в свою очередь, может позволить повысить эффективность применения Occlusion Culling в случаях, когда претендентов на оптимизацию очень много, не оказывая серьезных нагрузок на вычисления. При этом для менее сложных сцен влияние градации является ограниченным, что сохраняет важность базового порога, определяющего относить ли объект к окклюдерам.

В качестве направлений дальнейших исследований можно выделить расширение набора учитываемых факторов, добавляя в анализ такие характеристики как размеры объектов и их расположение относительно камеры, что позволяет определить дальнейшие этапы исследований.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Koch J., Gomse M., Schüppstuhl Th. Digital game-based examination for sensor placement in context of an Industry 4.0 lecture using the Unity 3D engine – a case study. *Procedia Manufacturing*. 2021;55:563–570. <https://doi.org/10.1016/j.promfg.2021.10.077>
2. Krajčovič M., Gabajová G., Matys M., et al. 3D Interactive Learning Environment as a Tool for Knowledge Transfer and Retention. *Sustainability*. 2021;13(14):7916. <https://doi.org/10.3390/su13147916>
3. Helbig C., Becker A.M., Masson T., et al. A game engine based application for visualising and analysing environmental spatiotemporal mobile sensor data in an urban context. *Frontiers in Environmental Science*. 2022;10:952725. <https://doi.org/10.3389/fenvs.2022.952725>
4. Geris A., Cukurbasi B., Kilinc M., et al. Balancing performance and comfort in virtual reality: A study of FPS, latency, and batch values. *Software: Practice and Experience*. 2024;54(12):2336–2348. <https://doi.org/10.1002/spe.3356>
5. Lee E.-S., Shin B.-S. Vertex Chunk-Based Object Culling Method for Real-Time Rendering in Metaverse. *Electronics*. 2023;12(12):2601. <https://doi.org/10.3390/electronics12122601>
6. Ding Y., Song Y. Vision-Degree-Driven Loading Strategy for Real-Time Large-Scale Scene Rendering. *Computers*. 2025;14(7):260. <https://doi.org/10.3390/computers14070260>
7. Rossi A., Barbiero M., Scremin P., et al. Robust Visibility Surface Determination in Object Space via Plücker Coordinates. *Journal of Imaging*. 2021;7(6):96. <https://doi.org/10.3390/jimaging7060096>
8. Ye L., Liu G., Chen G., et al. 3D Model Occlusion Culling Optimization Method Based on WebGPU Computing Pipeline. *Computer Systems Science and Engineering*. 2023;47(2):2529–2545. <https://doi.org/10.32604/csse.2023.041488>
9. Kiusya Z., Gikunda P., Musumba G. Adaptive mesh compression algorithm for near real-time rendering of large-scale static scenes. In: *2025 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA), 07–09 August 2025, Antalya, Turkiye*. IEEE; 2025. <https://doi.org/10.1109/ACDSA65407.2025.11166552>

10. Чёрный В.Г., Болсуновская М.В. Автоматизация подбора объектов для оптимизации рендеринга в Unity. В книге: *Развитие интеллектуальной экономики и промышленности на основе искусственного интеллекта*. Санкт-Петербург: Политех-Пресс; 2025. С. 755–771.
Chernyi V.G., Bolsunovskaya M.V. Automation of object selection for rendering optimization in Unity. In: *The development of the intellectual economy and industry based on artificial intelligence*. Saint Petersburg: Polytech-Press; 2025. P. 755–771. (In Russ.).
11. Meister D., Ogaki Sh., Benthin C., et al. A Survey on Bounding Volume Hierarchies for Ray Tracing. *Computer Graphics Forum*. 2021;40(2):683–712. <https://doi.org/10.1111/cgf.142662>
12. Meister D., Bittner J. Performance Comparison of Bounding Volume Hierarchies for GPU Ray Tracing. *Journal of Computer Graphics Techniques*. 2022;11(3):1–19.

ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATION ABOUT THE AUTHORS

Чёрный Виталий Григорьевич, аспирант, Санкт-Петербургский политехнический университет Петра Великого, Санкт-Петербург, Российская Федерация.
e-mail: vitaly.g.cherny@gmail.com
ORCID: [0009-0003-6281-6726](https://orcid.org/0009-0003-6281-6726)

Vitaliy G. Chernyi, Postgraduate, Peter the Great St. Petersburg Polytechnic University, Saint-Petersburg, the Russian Federation.

Болсуновская Марина Владимировна, кандидат технических наук, заведующая лабораторией «Промышленные системы потоковой обработки данных» Передовой инженерной школы «Цифровой инжиниринг», Санкт-Петербургский политехнический университет Петра Великого, Санкт-Петербург, Российская Федерация.
e-mail: marina.bolsunovskaia@spbpu.com
ORCID: [0000-0001-6650-6491](https://orcid.org/0000-0001-6650-6491)

Marina V. Bolsunovskaya, Candidate of Engineering Sciences, Head of the Laboratory "Industrial Systems for Streaming Data Processing" at the Advanced Engineering School "Digital Engineering", Peter the Great St. Petersburg Polytechnic University, Saint-Petersburg, the Russian Federation.

Статья поступила в редакцию 05.04.2026; одобрена после рецензирования 16.06.2026; принята к публикации 26.06.2026.

The article was submitted 05.04.2026; approved after reviewing 16.06.2026; accepted for publication 26.06.2026.