

УДК 004.42+004.92

## ПРОГРАММНОЕ СРЕДСТВО ВИЗУАЛИЗАЦИИ ТРЕХМЕРНЫХ СЦЕН С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ ТРАССИРОВКИ ЛУЧЕЙ

*Башура А.И., студент*

*Белорусский государственный университет информатики и радиоэлектроники,  
Институт информационных технологий,  
г. Минск, Республика Беларусь*

*Кунцевич О.Ю. – канд. пед. наук, доцент каф. ИСиТ*

**Аннотация.** В статье рассмотрены вопросы разработки программного средства визуализации трехмерных сцен с использованием технологии трассировки лучей, представлен пользовательский интерфейс приложения, обоснован выбор средств разработки, перечислены недостатки аналогичных разработок, представлены основные функциональные возможности программного средства. В качестве языка программирования выбран язык C#.

**Ключевые слова.** Программное обеспечение, разработка приложений, трехмерная графика, трассировка лучей, C#.

**Введение.** Появление компьютерной графики значительно повлияло на многие отрасли. В настоящее время обширно используются различные технологии компьютерной графики. Трехмерная графика используется в производстве фильмов, цифровой фотографии, видеоиграх, рекламе, архитектуре, искусстве, образовании, анимации, науке, медицине и множестве других отраслей. В связи с этим, разработка программного средства визуализации трехмерных сцен с использованием технологии трассировки лучей является актуальной [1].

Существуют различные аналогичные разработки. Основными недостатками данных программных средств являются сложность в освоении и отсутствие встроенного графического интерфейса пользователя, поэтому для создания и редактирования сцен необходимо использовать текстовый редактор или сторонние программы [2].

Целью проекта является разработка программного средства визуализации трехмерных сцен с использованием технологии трассировки лучей, в котором будут устранены выявленные недостатки аналогичных разработок.

Для реализации проекта необходимо решить ряд задач, в частности: провести анализ предметной области проекта; проанализировать существующие аналогичные программные продукты и сформировать функциональные требования; разработать функциональную модель, диаграммы деятельности и вариантов использования, а также иную проектную документацию программного средства (далее – ПС); спроектировать интерфейс пользователя; разработать алгоритмы работы ПС и выполнить их программную реализацию; произвести тестирование и отладку разработанного ПС; разработать руководство по установке и использованию ПС; создать демонстрационную сцену.

В качестве языка разработки был выбран C#, так как он является современным языком с поддержкой объектно-ориентированного программирования и наличием большого числа возможностей, включающих обработку исключений, сборщик мусора, асинхронность, кроссплатформенность [3].

Для отображения пользовательского интерфейса будет использован фреймворк WPF, который позволяет с легкостью строить современный оконный интерфейс с помощью языка разметки XAML. Для осуществления возможности отображения и редактирования трехмерной интерактивной сцены будет использована библиотека OpenGL, позволяющая задействовать аппаратное ускорение при отрисовке [4].

Практическая значимость результатов разработки заключается в возможности применения ПС в области компьютерной графики.

**Основная часть.** Определим назначение и возможности ПС:

- обеспечение возможности получения двумерных изображений трехмерных сцен и сохранения их для дальнейшей обработки в основные форматы изображений (.png, .gif, .jpg);
- предоставление реализации интерактивного взаимодействия в трехмерном пространстве, включающая функции перемещения в пространстве, размещения и редактирования объектов;
- осуществление функций манипуляции с объектами: перемещение, поворот, изменение размеров, редактирование прочих свойств в зависимости от выбранного объекта;
- реализация загрузки файлов трехмерных моделей в формате GLTF, а также загрузки изображений в формате .png, .jpg в качестве текстурных карт и карт окружения;
- реализация следующих типов освещения: окружающий свет, направленные источники света, точечные источники света и прожекторные источники света;
- свет должен использовать модель физически корректного рендеринга с использованием карт металличности и шероховатости.

На рисунке 1 представлена диаграмма вариантов использования программного средства.

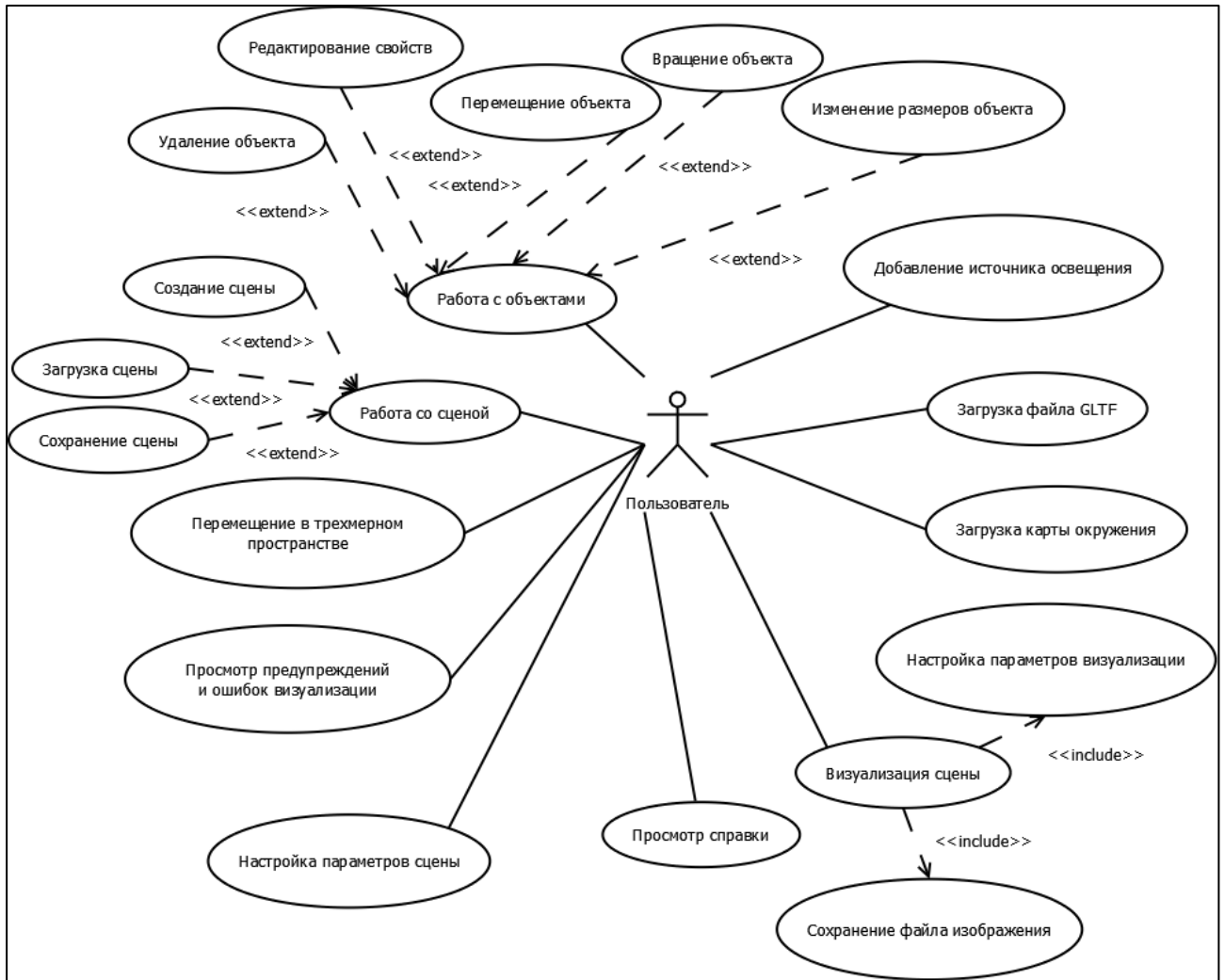


Рисунок 1 – Диаграмма вариантов использования

При запуске программы пользователь может создать новую сцену, а также загрузить существующую. Изначально будет создана пустая сцена, в которой можно продолжить работу. Далее пользователь может перейти к визуализации или редактированию.

В режиме редактирования пользователь может перемещаться в трехмерном пространстве, добавлять, редактировать и удалять объекты, редактировать параметры сцены и просматривать справочную информацию.

В режиме визуализации пользователю необходимо настроить параметры визуализации, после чего будет осуществлена проверка объектов сцены с отображением найденных ошибок. Если ошибок не обнаружено, будет произведена визуализация сцены, после чего пользователь сможет сохранить итоговое изображение.

По завершении работы с текущей сценой пользователь может сохранить сцену и продолжить работу с другой сценой, а также осуществить выход из программы.

Интерфейс главного окна представлен на рисунке 2.

Рассмотрим предложенный концепт более подробно:

- в верхней части экрана располагается меню, содержащее основные команды, а также панель инструментов с наиболее часто используемыми кнопками;
- в левой части экрана расположено дерево объектов, содержащее объекты сцены;
- выбор объекта в дереве приводит к отображению выбранного инструмента в опорной точке объекта в панели просмотра сцены, находящейся по центру экрана;
- в правой части экрана расположена панель, на которой отображаются изменяемые свойства выбранного объекта;
- панель отображения ошибок расположена в нижней части экрана.

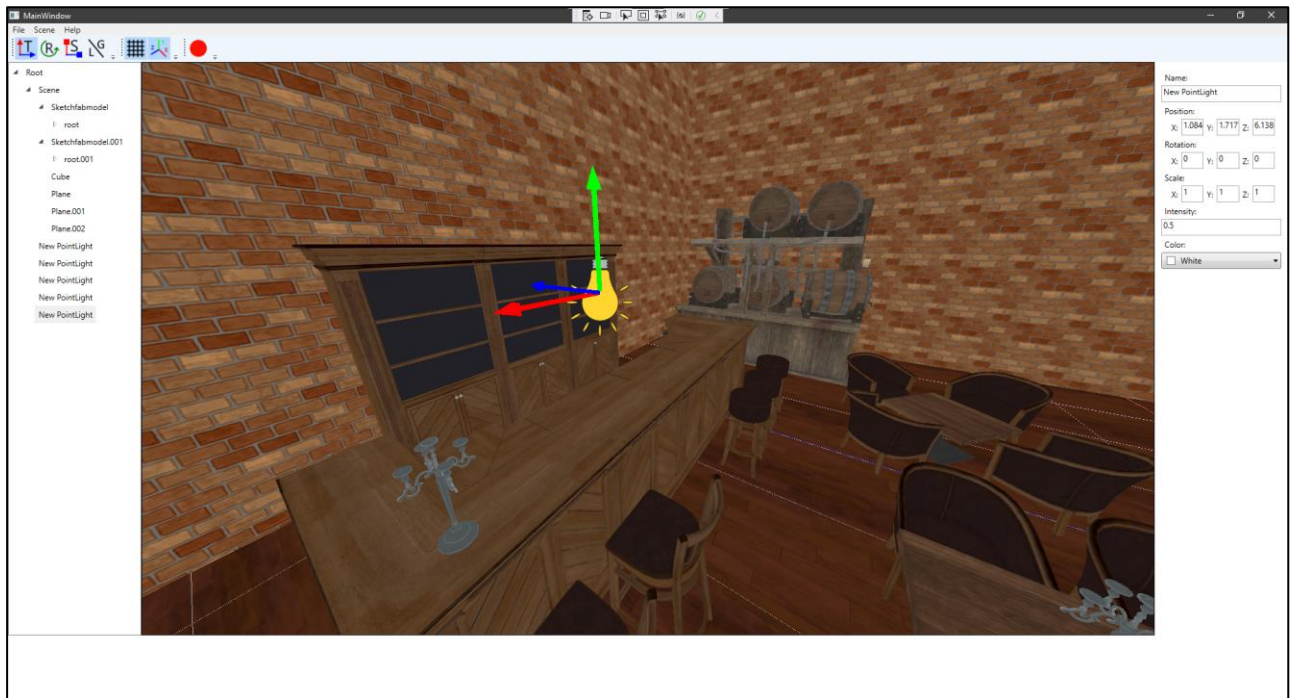


Рисунок 2 – Интерфейс главного окна

Для компонентов трехмерной сцены была создана особая система для регистрации моделей и моделей представления.

В процессе работы приложения при выборе объекта в дереве объектов сцены динамически создаются необходимые представления и модели-представления с использованием механизма рефлексии языка C#, учитывая иерархию классов.

**Заключение.** Разрабатываемое приложение позволяет быстро создавать сцены из готовых объектов и получать готовое изображение трехмерных объектов. Применение технологии трассировки лучей позволяет получить детальное отображение отраженного и преломленного изображения на поверхности объектов.

Система спроектирована с учетом возможности легкого расширения новыми компонентами и возможностью их регистрации всего одной строкой кода.

**Список использованных источников:**

1. Задорожный, А. Г. Компьютерная графика: введение в трассировку лучей: учебное пособие / А. Г. Задорожный. – Новосибирск: Изд-во НГТУ, 2021. – 64 с.
2. Sunflow Global Illumination Rendering System – [Электронный ресурс]. – Режим доступа: <https://github.com/fpsunflower/sunflow>.
3. Прайс, М. С# 10 и .NET 6. Современная кроссплатформенная разработка / М. Прайс. – СПб.: Питер, 2023. – 848 с.
4. Gordon, S. Computer Graphics Programming in OpenGL with C++ / S. Gordon, J. Clevenger – Mercury Learning and Information, 2020. – 514 с

UDC 004.42+004.92

## SOFTWARE TOOL FOR VISUALISATION OF THREE-DIMENSIONAL SCENES USING RAY TRACING TECHNOLOGY

*Bashura A.I.*

*Institute of Information Technologies of the Belarusian State University of Informatics and Radioelectronics,  
Minsk, Republic of Belarus*

*Kuntsevich O.Yu. – Candidate of Pedagogic Sciences, Associate Professor*

**Annotation.** The article deals with the development of a software tool for visualisation of three-dimensional scenes using ray tracing technology, presents the user interface of the application, justifies the choice of development tools, lists the disadvantages of similar developments, presents the main functionalities of the software tool. The C# language is chosen as the programming language.

**Keywords.** Software, application development, 3D graphics, ray tracing, C#.