



Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Визуализация решения задачи Стефана

Студент: Готов Илья Анатольевич ИУ7-52Б

Руководитель курсового проекта: Кострицкий Александр Сергеевич

Цель и задачи

Целью курсовой работы является разработка программного обеспечения, позволяющего получить реалистичное изображение полого кусочка льда с жидкостью внутри.

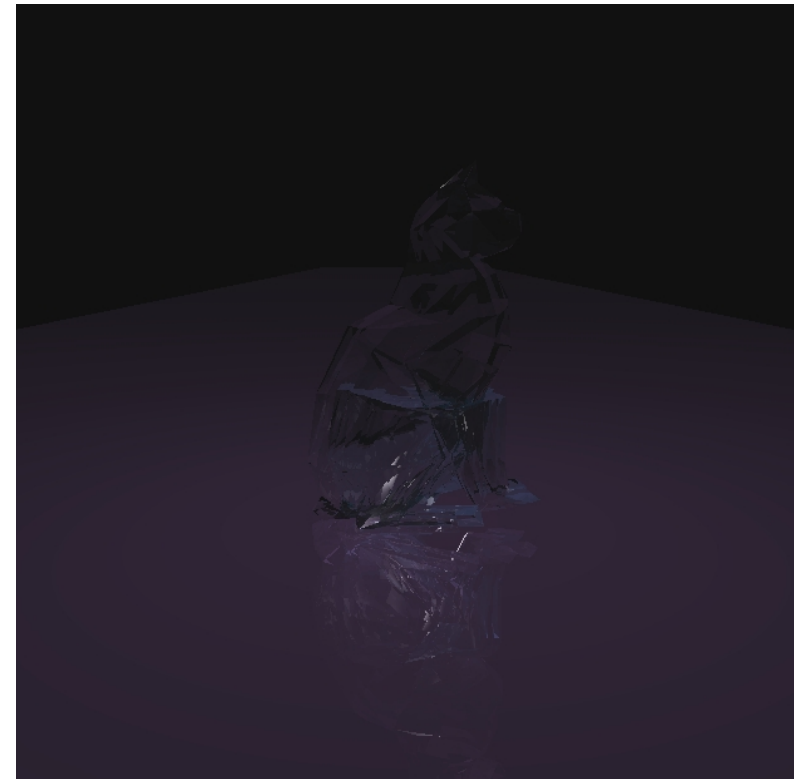
Задачи:

- провести анализ алгоритмов построения реалистичных изображений;
- разработать метод построения реалистичного изображения полого кусочка льда с жидкостью внутри;
- реализовать метод построения реалистичного изображения полого кусочка льда с жидкостью внутри;
- исследовать зависимость времени выполнения однопоточной и многопоточной реализаций метода построения реалистичного изображения полого кусочка льда с жидкостью внутри от размера изображения.

Описание и формализация объектов сцены

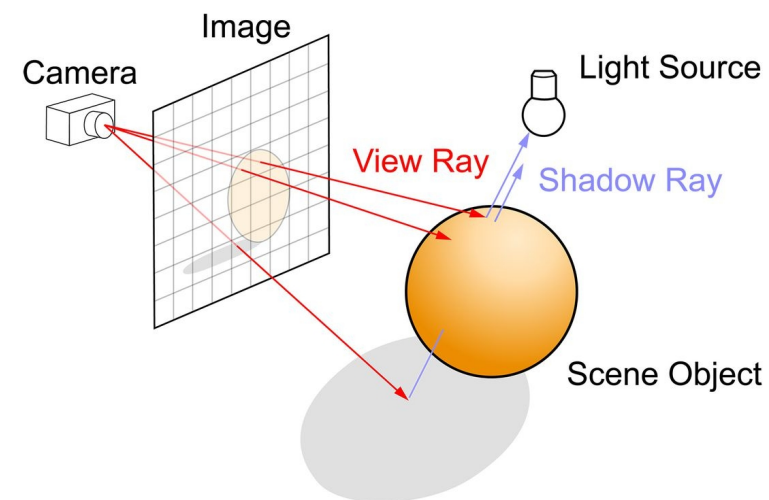
Объекты сцены:

- Геометрический объект (представляется в виде полигональной сетки).
- Источник света (задаётся расположением в пространстве, цветом излучения и интенсивностью излучения).
- Камера (задаётся расположением в пространстве и направлением взгляда).

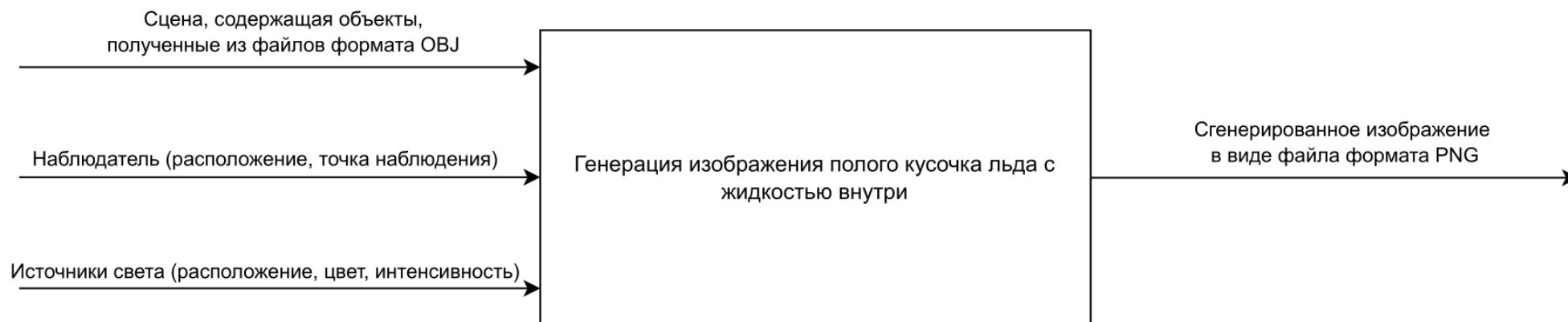


Алгоритмы трёхмерной графики

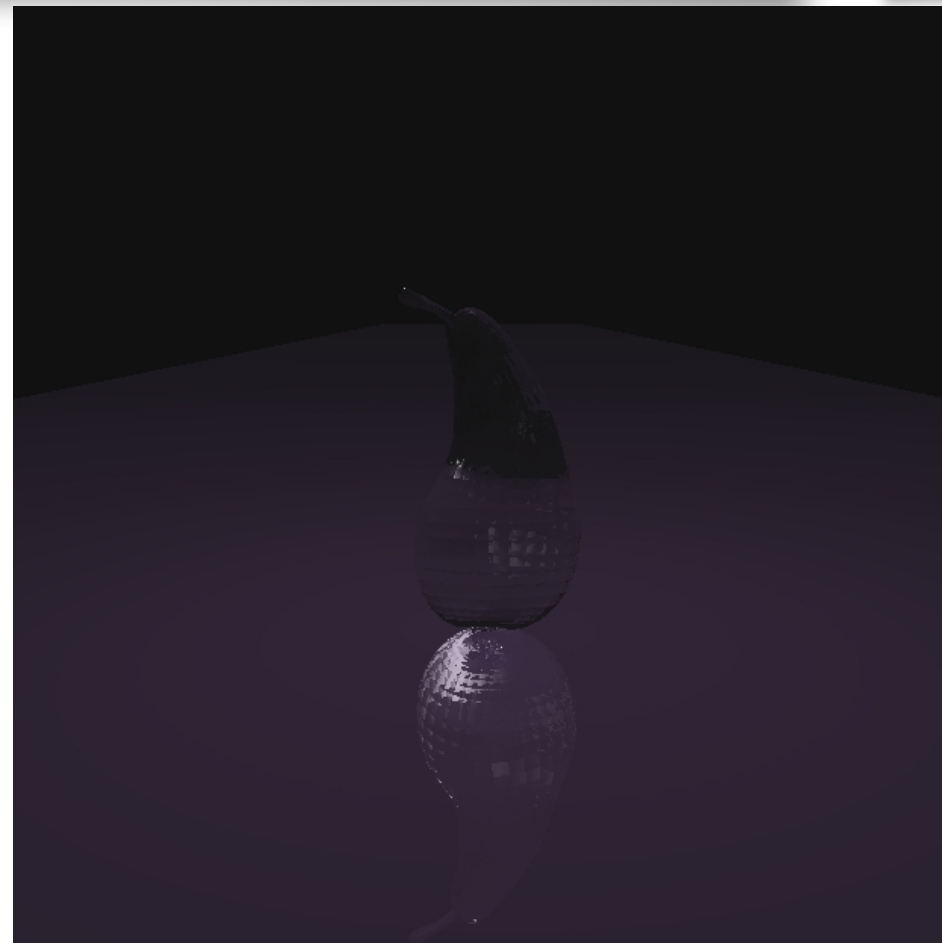
В качестве алгоритма удаления невидимых рёбер и поверхностей был выбран алгоритм обратной трассировки лучей с глобальной моделью освещения из-за высокой реалистичности синтезируемого изображения и возможности визуализации зеркальных и прозрачных поверхностей.



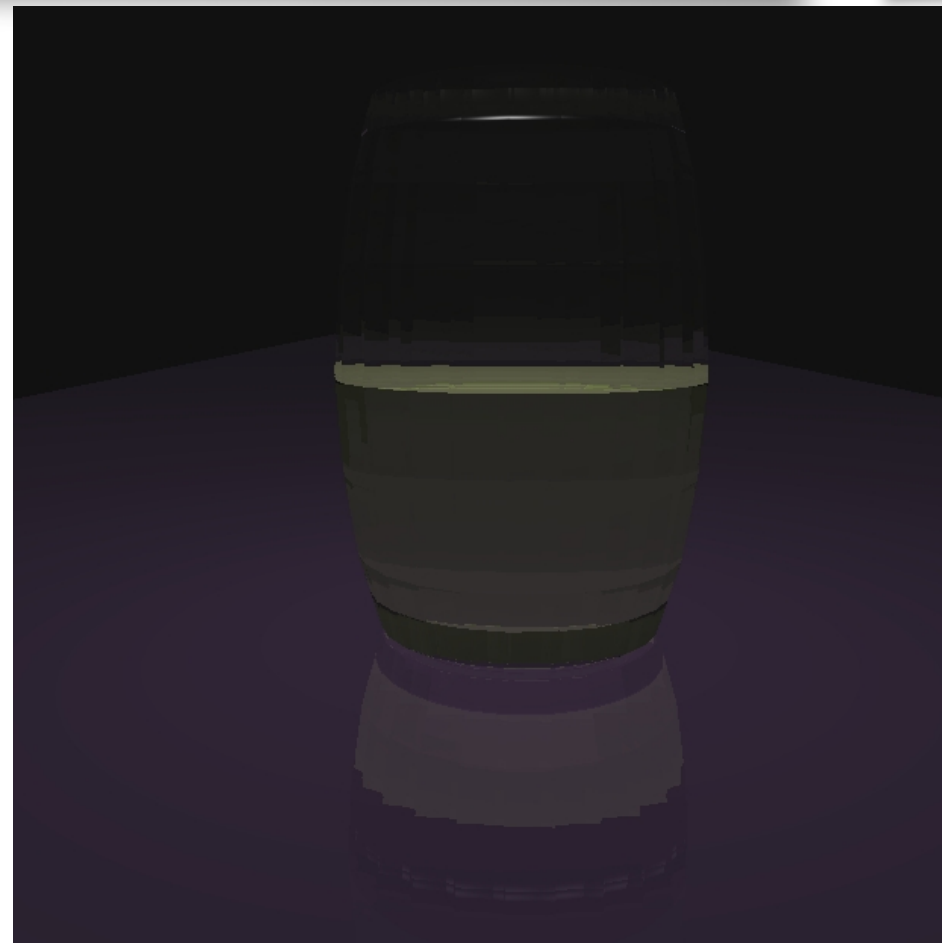
Требования к ПО



Примеры работы программного обеспечения



Примеры работы программного обеспечения

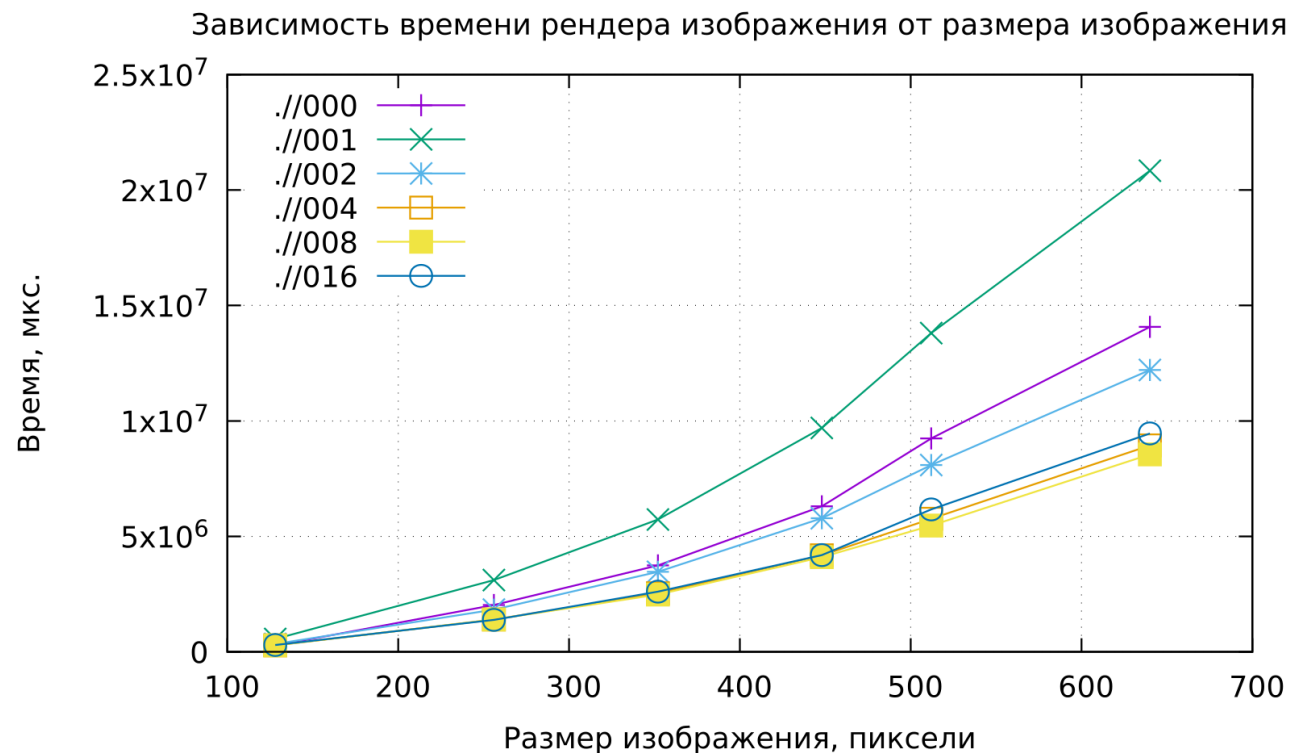


Проведение исследования

Цель: определение зависимости времени генерации изображения от размера изображения при различном количестве потоков.

Размеры изображения	Время выполнения без использования распараллеливания, мкс	Время выполнения с использованием распараллеливания, мкс				
		1	2	4	8	16
128x128	273,311	556,595	339,305	278,957	279,667	292,878
256x256	2,023,442	3,103,714	1,829,316	1,395,264	1,397,786	1,384,682
352x352	3,739,252	5,731,390	3,464,986	2,533,300	2,478,940	2,608,794
448x448	6,302,434	9,693,502	5,783,250	4,178,672	4,100,138	4,188,324
512x512	9,245,352	13,801,280	8,086,786	5,764,098	5,463,540	6,166,600
640x640	14,072,760	20,832,840	12,204,240	8,941,954	8,551,252	9,542,858

Проведение исследования



- Лучшее время распараллеленный алгоритм показал при 8 потоках, что соответствует количеству логических процессоров компьютера, на котором проводилось измерение
- Не распараллеленная реализация работает быстрее однопоточной, поскольку в однопоточной тратится дополнительное время на создание потока

Заключение

В ходе выполнения курсового проекта были выполнены все задачи:

- описан список доступных к размещению на сцене моделей, а сами модели были формализованы;
- выбраны существующие алгоритмы компьютерной графики для визуализации сцены и объектов на ней;
- выбран язык программирования и среда разработки;
- реализованы выбранные алгоритмы визуализации;
- реализовано программное обеспечение для визуализации и редактирования городского пространства.

Поставленная цель была достигнута.