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

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К КУРСОВОЙ РАБОТЕ

НА ТЕМУ:

Разработка программного обеспечения для моделирования упругих столкновений объектов в пространстве.

Студент ИУ7-54Б _____ К. А. Рунов
(Группа) (Подпись, дата) (И. О. Фамилия)

Руководитель курсовой работы _____ А. А. Павельев
(Подпись, дата) (И. О. Фамилия)

2023 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

На сегодняшний день компьютерная графика является неотъемлемой частью нашей жизни и используется повсеместно. Ей находится применение в самых разных областях человеческой деятельности: она используется в науке, в бизнесе, в кино, играх и везде, где нужно визуальное представление информации на электронном дисплее.

Перед разработчиками графического программного обеспечения часто стоит задача синтеза реалистического изображения. Для решения этой задачи существует множество алгоритмов, но, как правило, алгоритмы, дающие наилучшие результаты являются наиболее трудозатратными как по объёму требуемой памяти, так и по количеству требуемых вычислений, потому что учитывают множество световых явлений (дифракция, интерференция, преломление, поглощение, множественное отражение). Поэтому, в зависимости от задачи и от имеющихся вычислительных мощностей, программистам приходится выбирать наиболее целесообразные в их случае алгоритмы и жертвовать либо временем генерации кадра, либо его реалистичностью.

Но генерация реалистического изображения — это лишь одна из многих задач, которые стоят перед программистами графических приложений. При разработке игр или приложений для моделирования физики твёрдого тела возникает задача обнаружения коллизий (столкновений) между объектами виртуального пространства и реагирования на них, например, разрушение объектов в результате столкновения, деформация объектов, или их упругое соударение.

В данной курсовой работе было решено создать программу — песочницу, в которой пользователь сможет размещать объекты в виртуальном пространстве, изменять их свойства (такие как размер, местоположение, цвет, масса), задавать им начальные скорости; после чего наблюдать их перемещение и столкновения.

Цель работы — разработка программы для моделирования упругих столкновений объектов в пространстве.

Для достижения поставленной цели требуется решить следующие задачи:

— описать свойства объекта, которыми он должен обладать для моделирования его движения и столкновения с другими объектами;

— проанализировать существующие способы представления объектов и обосновать выбор наиболее подходящего для решения поставленной задачи;

— проанализировать существующие алгоритмы построения изображения и обосновать выбор тех из них, которые в наибольшей степени подходят для решения поставленной задачи;

— проанализировать существующие алгоритмы обнаружения коллизий и обосновать выбор тех из них, которые в наибольшей степени подходят для решения поставленной задачи;

— проанализировать существующие алгоритмы разрешения коллизий и обосновать выбор тех из них, которые в наибольшей степени подходят для решения поставленной задачи;

— проанализировать существующие модели освещения и обосновать выбор модели, наиболее подходящей для решения поставленной задачи;

— реализовать выбранные алгоритмы;

— разработать программное обеспечение для решения поставленной задачи;

— провести анализ производительности работы программы в зависимости от количества объектов на сцене и их типов.

1 Аналитическая часть

В данной части под «сценой» понимается виртуальное пространство, в котором расположены объекты, предназначенные для визуализации.

1.1 Описание объектов сцены

В данном подразделе будут описаны свойства объектов, которыми он должен обладать для моделирования его движения и столкновения с другими объектами.

Для визуализации объекта, он должен содержать следующую информацию:

— геометрическая информация, какую форму имеет объект;

— информация о местоположении в пространстве (с учётом поворота и масштабирования);

— информация о цвете и/или текстуре объекта.

Для моделирования движения и столкновения объектов, объекты должны содержать следующую информацию:

— информация о «коллайдере» (как правило, упрощённая форма исходного объекта, которая используется при обнаружении столкновений);

— информация о физических свойствах объекта (скорость, ускорение, масса).

1.2 Выбор представления объектов сцены

Далее будут рассмотрены возможные способы представления объектов сцены.

1.2.1 Каркасная модель

Объект представляется с помощью его вершин и рёбер, без каких-либо поверхностей.

Преимущества:

— простота;

— позволяет получить базовое представление о форме объекта;

— быстрая визуализация.

Недостатки:

— не позволяет производить реалистическое освещение, для которого требуется информация о гранях объекта;

— не содержит информации, нужной для обнаружения столкновений.

1.2.2 Поверхностная модель

Объект представляется в виде набора поверхностей. Поверхность можно задать разными способами, например, уравнением или системой уравнений. Часто поверхность представляется в виде набора граней (обычно треугольных), которые, в свою очередь, состоят из набора вершин исходного объекта.

Преимущества:

— простота;

— возможность учёта освещения;

— возможность достижения высокого уровня реализма;

— содержит информацию о поверхностях, которая нужна при обнаружении столкновений объектов;

Недостатки:

— не математически точное представление, аппроксимация.

1.2.3 Твердотельная модель

Существует несколько методов представления твердотельных моделей: метод констркутивного представления (англ. Constructive representation, сокращённо C – rep) и метод граничного представления (англ. Boundary representation, сокращённо B – rep). Оба метода предоставляют наиболее полное описание объекта, включая его внешнюю форму и внутреннюю структуру.

Преимущества:

— математически точное представление;

— наиболее полное описание структуры объекта.

Недостатки:

— сложность;

— требовательность к памяти;

— содержит излишнюю информацию, которая не будет использована при обнаружении столкновений объектов.

Вывод

На основе проведённого анализа различных способов представления объектов сцены, была выбрана поверхностная модель, так как она обладает всей информацией, нужной для обнаружения столкновений объектов и учёта освещения, а также является менее требовательной к памяти по сравнению с твёрдотельной моделью.

1.3 Выбор алгоритма удаления невидимых линий и поверхностей

1.3.1 Алгоритм Робертса

1.3.2 Алгоритм обратной трассировки лучей

1.3.3 Алгоритм Варнока

1.3.4 Алгоритм, использующий z-буфер

Вывод

1.4 Выбор алгоритмов обнаружения коллизий

1.4.1 Алгоритм обнаружения коллизий с использованием сфер

1.4.2 Алгоритм AABB

1.4.3 Алгоритм OBB

1.4.4 Алгоритм GJK

Вывод

1.5 Выбор алгоритмов разрешения коллизий

1.5.1 Алгоритм EPA

Вывод

1.6 Выбор модели освещения

Вывод

2 Конструкторская часть

2.1 Требования к программному обеспечению

2.2 Разработка алгоритмов

2.2.1 Общий алгоритм решения поставленной задачи

2.2.2 Алгоритм, использующий z-буфер

2.2.3 Алгоритм AABB

2.2.4 Алгоритм GJK

2.2.5 Алгоритм EPA

2.2.6 Модель освещения Фонга

2.3 Выбор типов и структур данных

2.4 Общая архитектура разрабатываемой программы

Вывод

3 Технологическая часть

3.1 Средства реализации

3.1.1 Выбор языка программирования

3.1.2 Выбор среды разработки

3.2 Структура программы

3.3 Интерфейс

3.4 Работа программы

Далее будут приведены замечания относительно работы программы.

3.5 Демонстрация работы программы

4 Исследовательская часть

4.1 Технические характеристики

4.2 Влияние количества объектов на скорость генерации кадра

4.3 Влияние типов объектов на скорость генерации кадра

Вывод

ЗАКЛЮЧЕНИЕ

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ