

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)

«Моделирование статической сцены расстановки шахматных фигур на шахматной доске»

Группа: ИУ7-53Б

Студент: Конкина Алина Николаевна

Научный руководитель: Мартынюк Наталья Николаевна

2023 г.

Цель и задачи

Цель: разработка программного обеспечения с пользовательским интерфейсом, реализовывающего моделирование статической сцены расстановки шахматных фигур на шахматной доске.

Задачи:

- проанализировать предметную область, рассмотреть известные подходы и алгоритмы решения задачи синтеза изображения в контексте моделирования статической сцены;
- спроектировать программное обеспечение;
- выбрать средства реализации и разработать программное обеспечение;
- исследовать характеристики разработанного программного обеспечения.

Формализация задачи

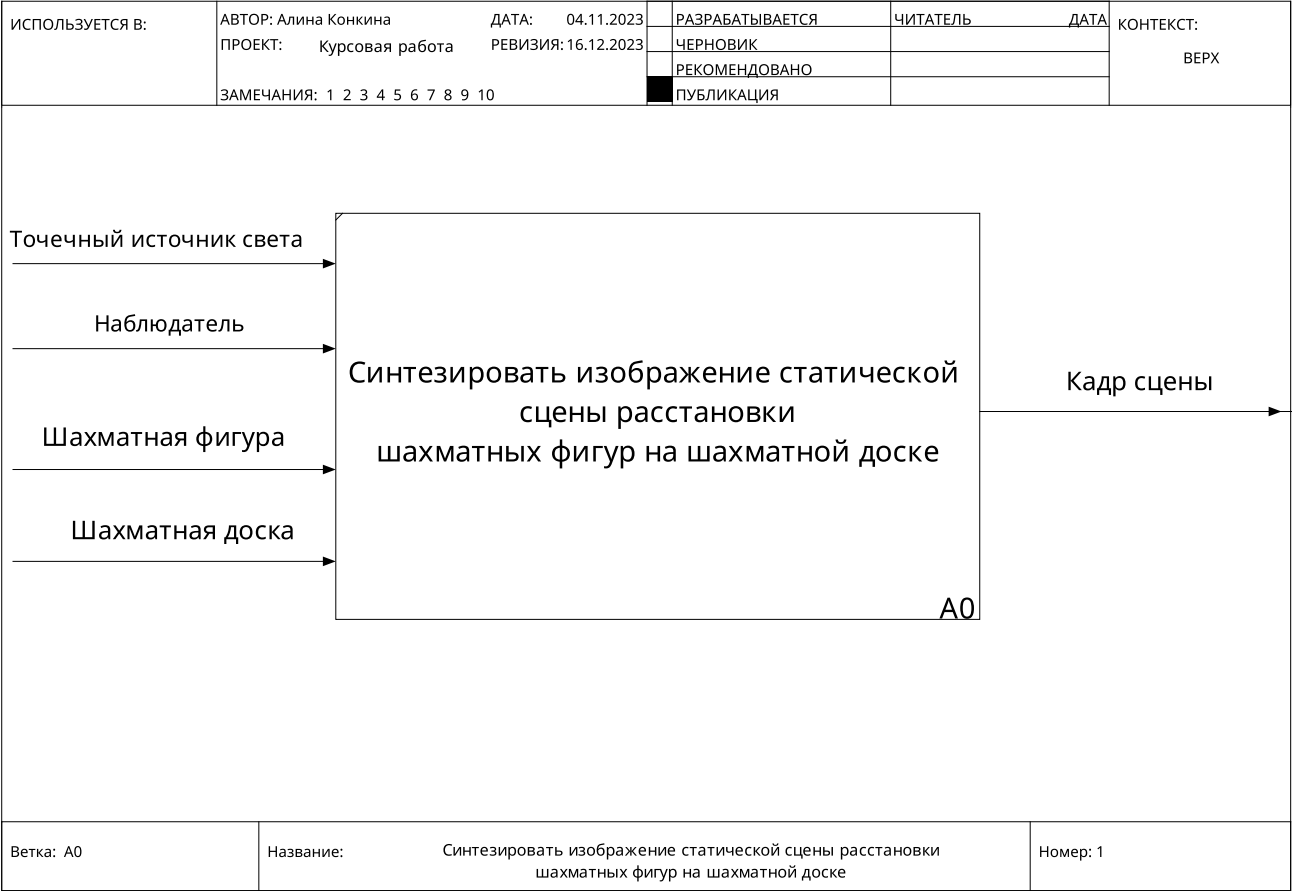


Рисунок 1 — IDEF0 диаграмма формализуемой задачи

Модель, способ хранения, изменение местоположения в пространстве

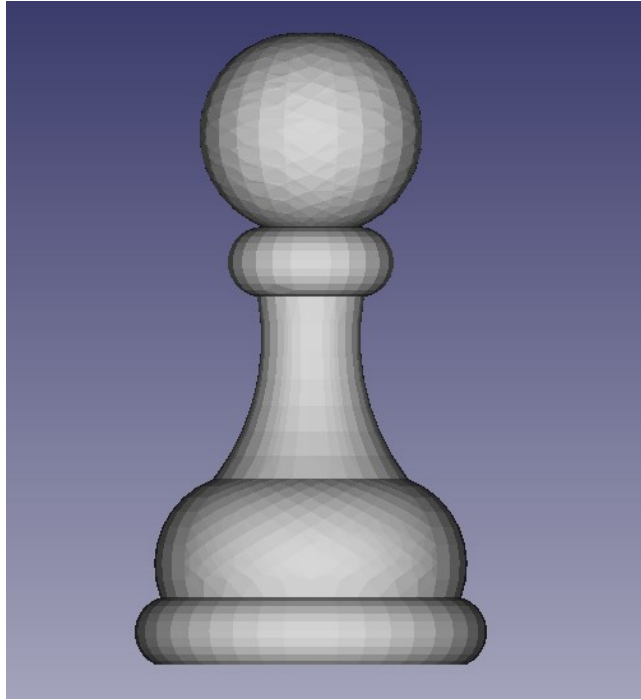


Рисунок 2 — пешка в формате
stl, созданная с использованием
стороннего пакета

$$M_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

$$M_y(\alpha) = \begin{pmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & 1 & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{pmatrix}$$

$$M_z(\alpha) = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$M(dx, dy, dz) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ dx & dy & dz & 1 \end{pmatrix}$$

Рисунок 3 — матрицы поворота
и переноса

Трассировка лучей

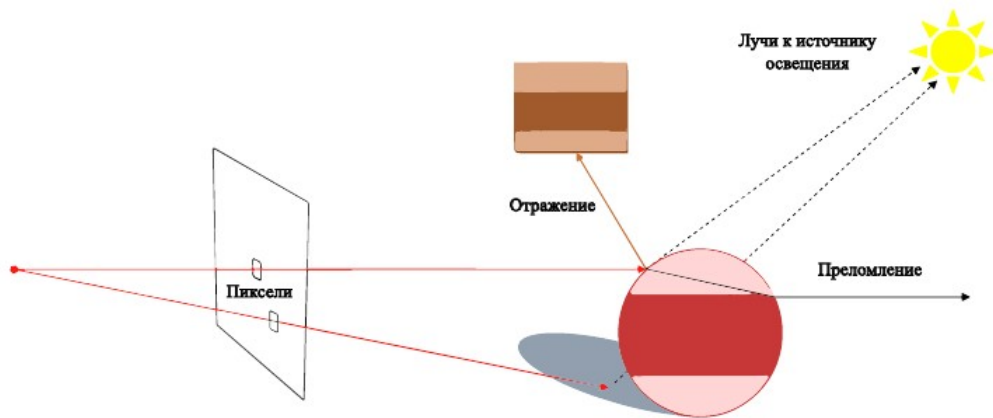


Рисунок 4 — обратная трассировка лучей

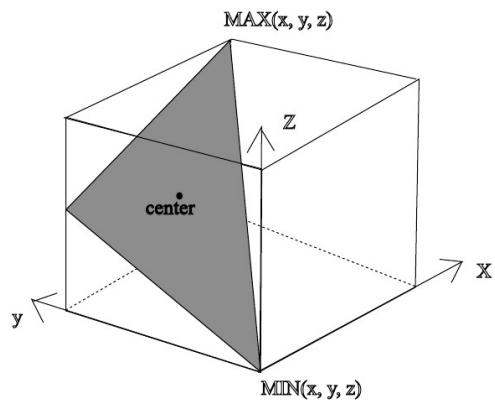


Рисунок 5 — ограничивающий параллелепипед

$$I = I_d \cdot C + I_s + K_r \cdot I_r + K_t \cdot I_t$$

Рисунок 5 — формула интенсивности точки объекта по модели Уиттеда

Декомпозиция ПО

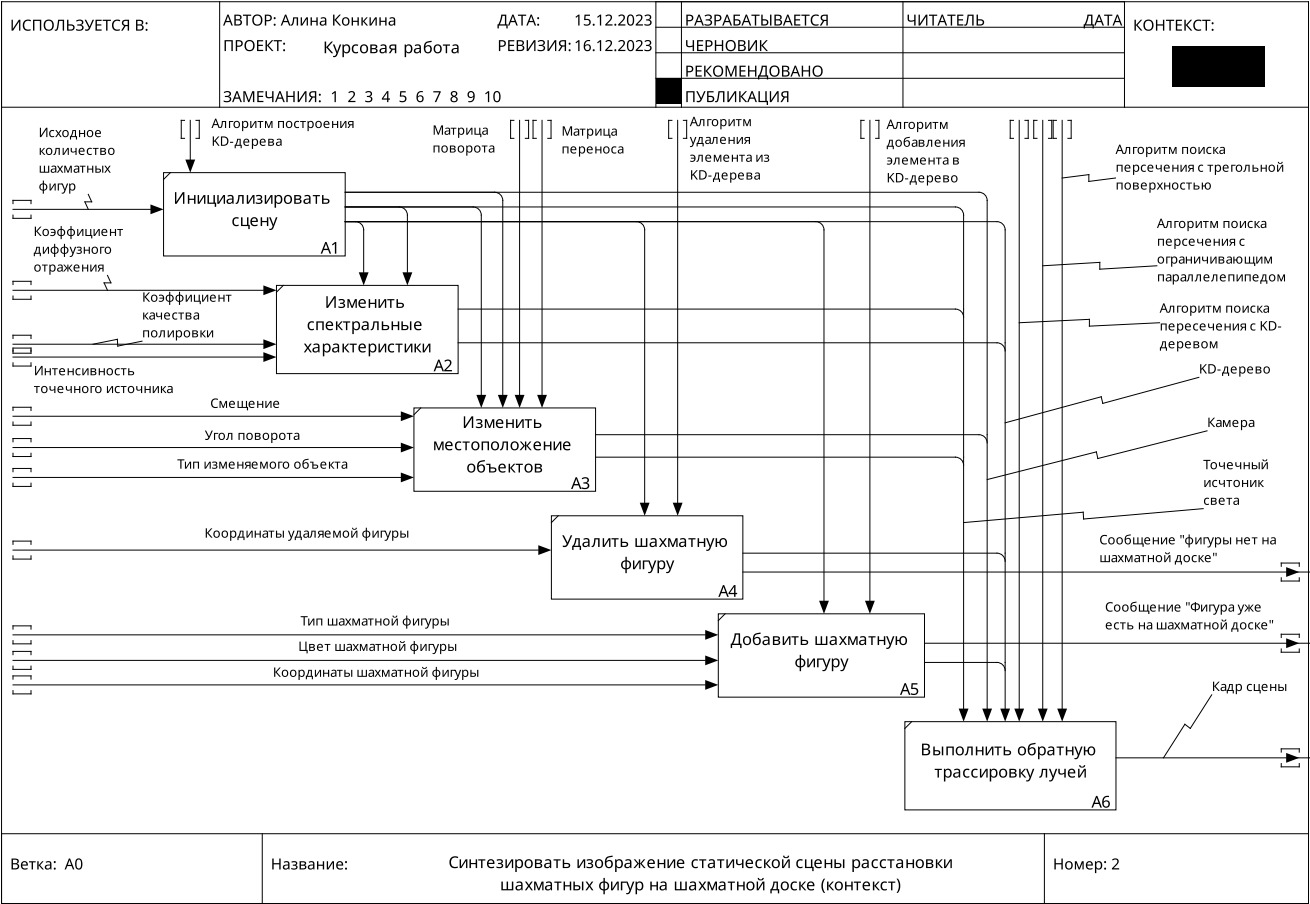


Рисунок 6 — IDEF0 диаграмма функциональной декомпозиции программного обеспечения

Средства реализации

- язык программирования c++;
- кроссплатформенная библиотека Qt для разработки графического интерфейса;
- библиотека Intel® Threading Building Blocks (Intel® TBB), предназначенная для параллелизма на уровне инструкций во время выполнения для эффективного использования ресурсов процессора;
- библиотека stl_reader, состоящая для одного заголовочного файла и предназначенная для чтения stl файлов, а также преобразования содержимого в пользовательские контейнеры;
- утилита smake.

Диаграмма классов

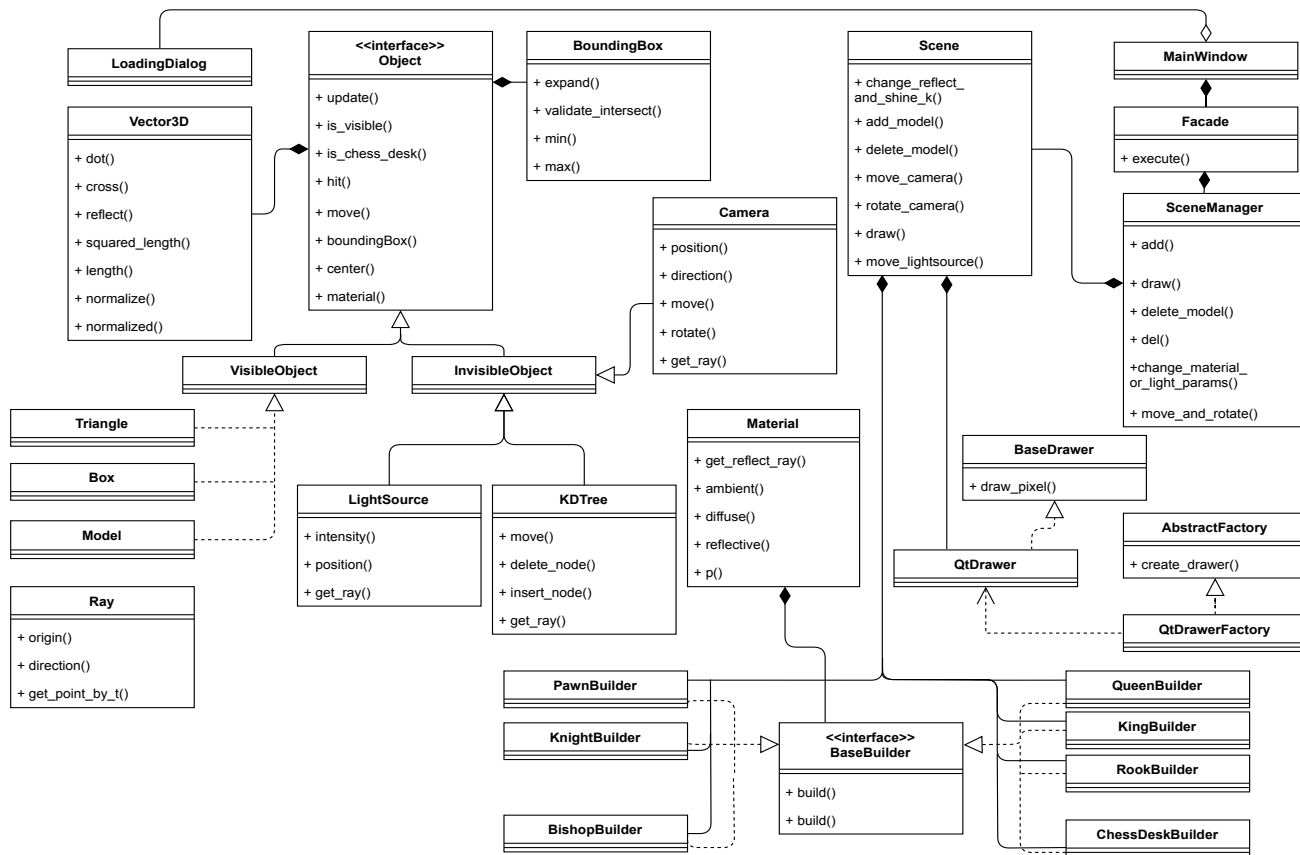


Рисунок 7 — Диаграмма классов

Интерфейс

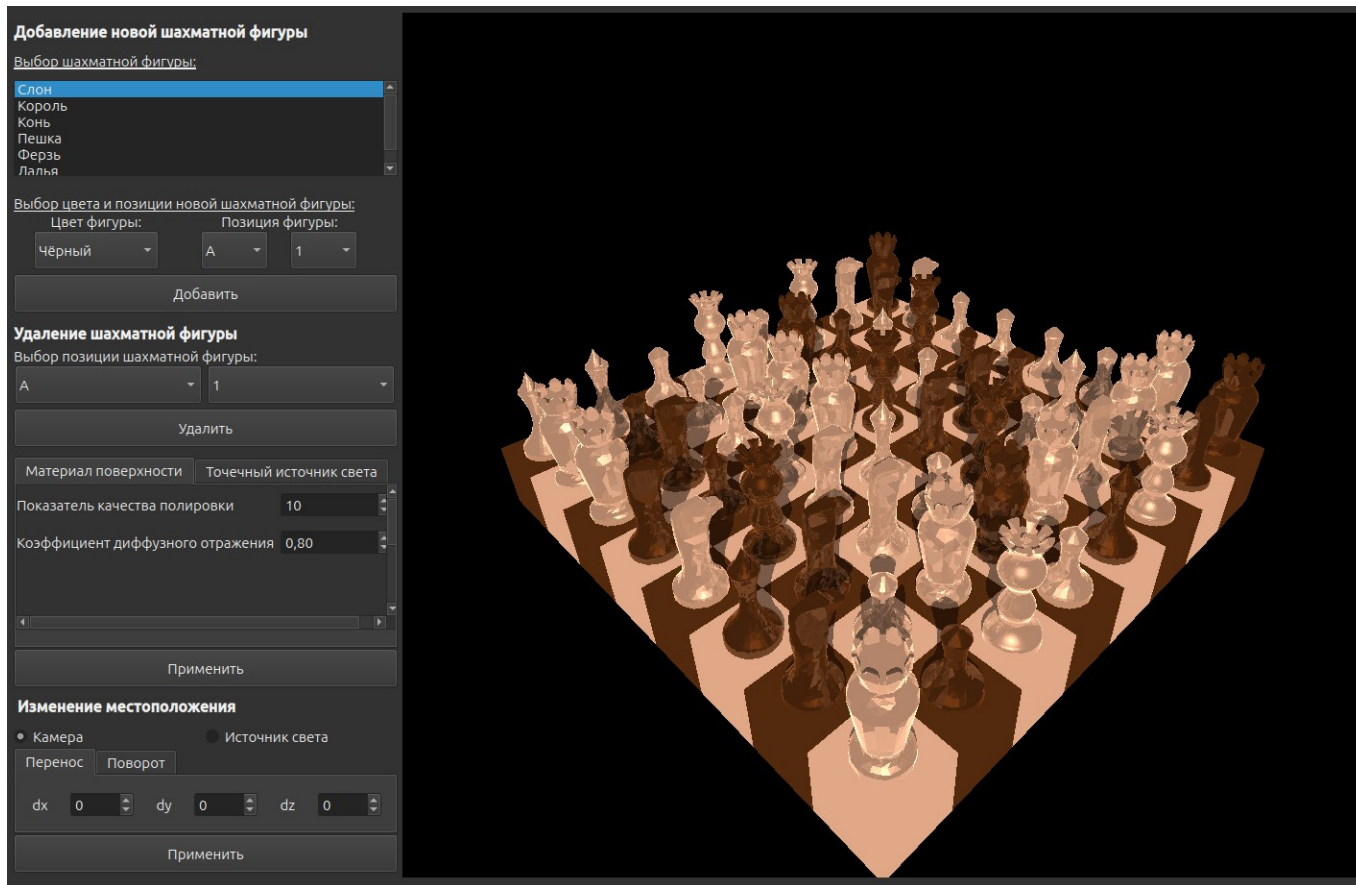


Рисунок 8 — Интерфейс

Добавление и удаление шахматных фигур

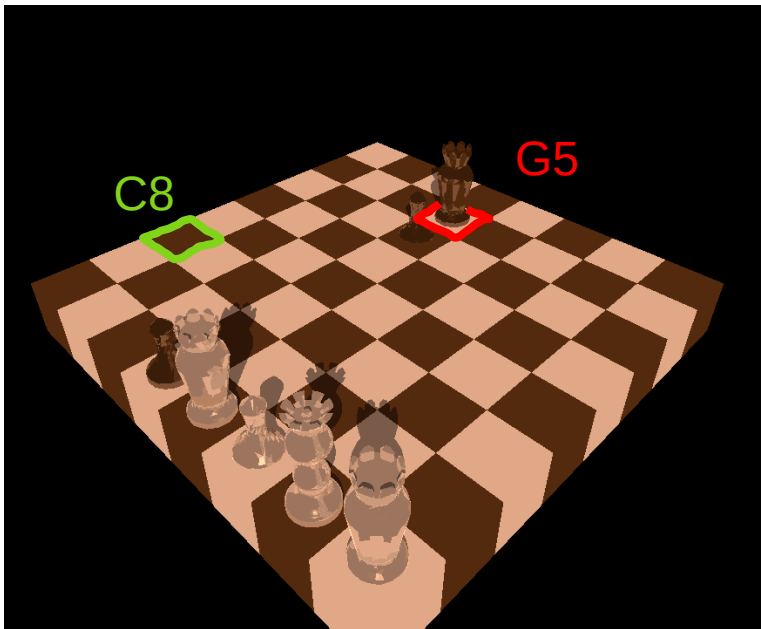


Рисунок 9 — Изображение сцены перед добавлением и удалением фигур

Добавление новой шахматной фигуры

Выбор шахматной фигуры:

- Слон
- Король
- Конь
- Пешка
- Ферзь**
- Ладья

Выбор цвета и позиции новой шахматной фигуры:

Цвет фигуры: Белый Позиция фигуры: С 8

Добавить

Удаление шахматной фигуры

Выбор позиции шахматной фигуры:

С 5

Удалить

Рисунок 10 — Параметры добавления и удаления фигур

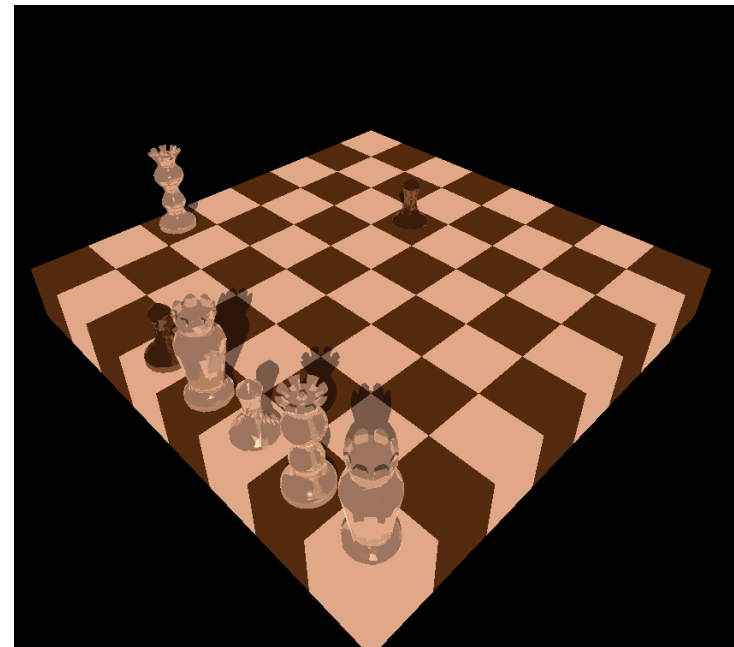


Рисунок 11 — Сцена после добавления и удаления фигур

Изменение качества полировки и коэффициента диффузного отражения

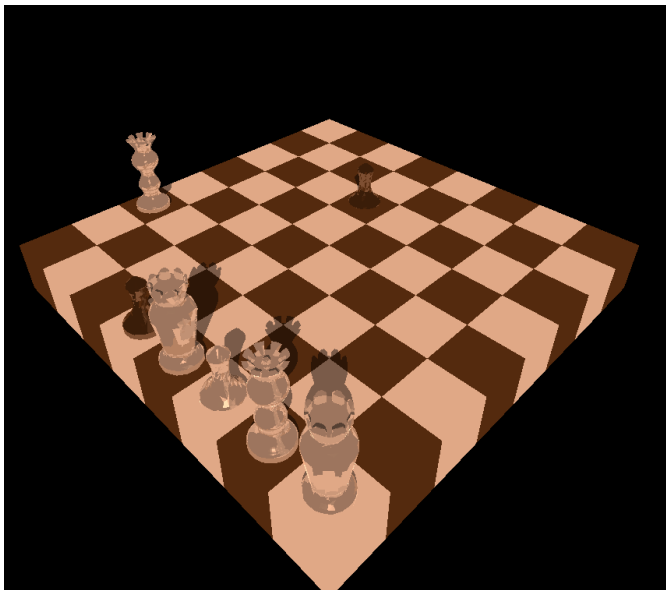


Рисунок 12 — Изображение сцены перед изменением коэффициентов ($p = 100$, $kd = 0.7$)

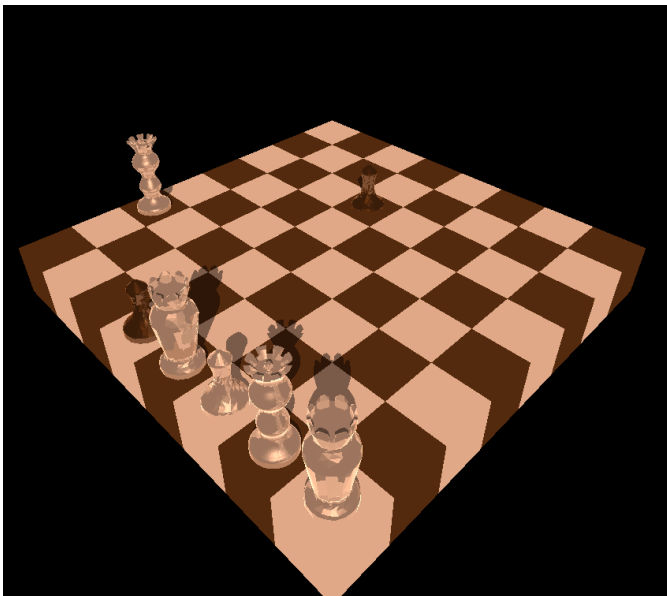


Рисунок 13 — $p = 10$

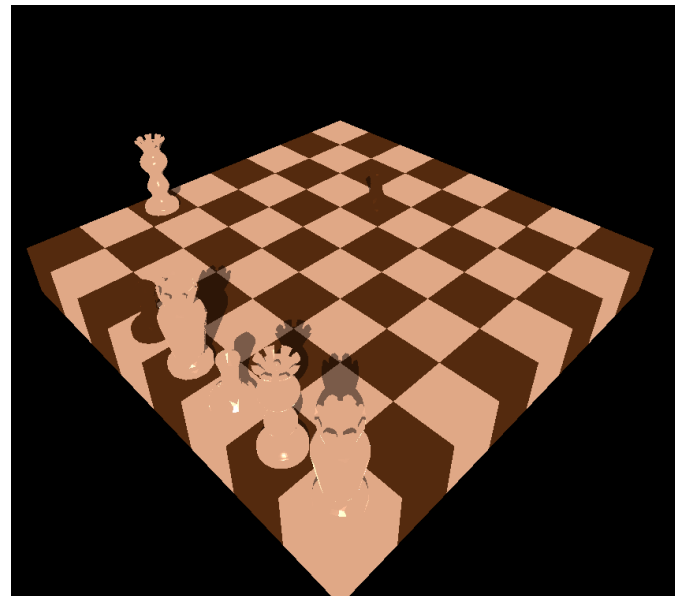


Рисунок 14 — $kd = 1$

Изменение интенсивности точечного источника света и изменение его местоположения

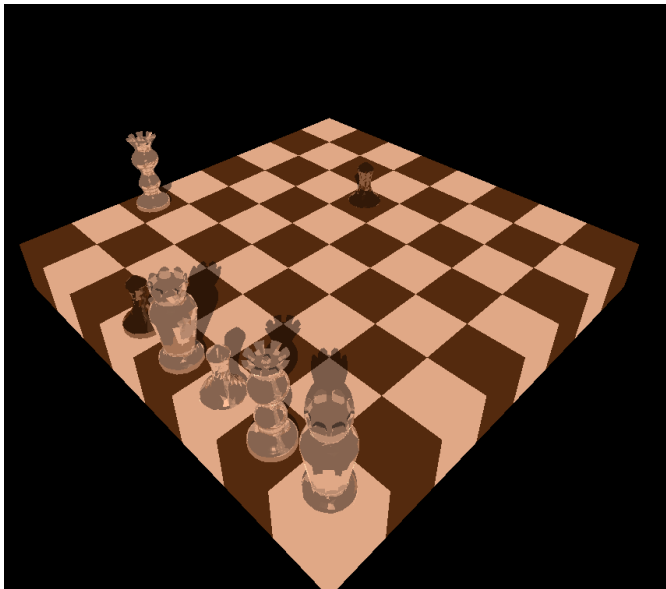


Рисунок 15 — Изначальная сцена

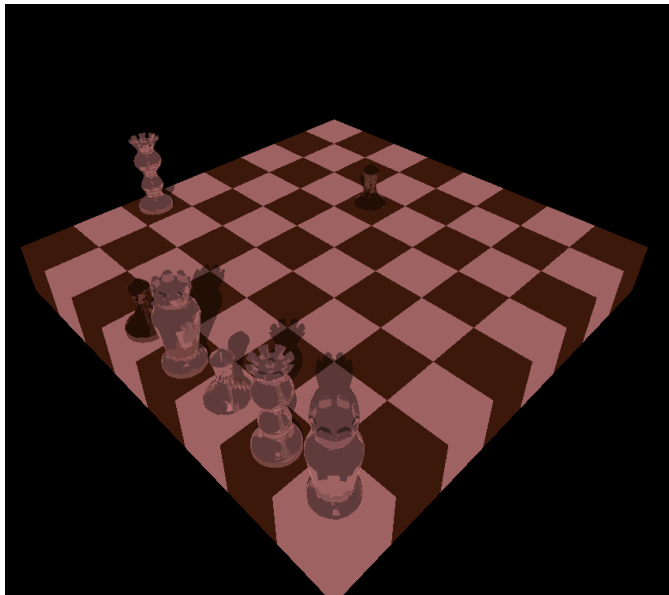


Рисунок 16 — $I = (255, 170, 255)$

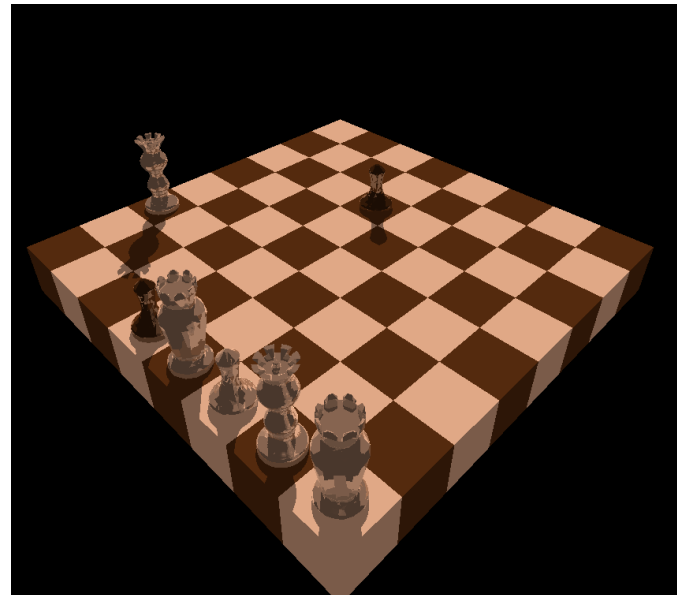


Рисунок 17 — $dx = 500, dy = 500$

Изменение местоположение и поворот камеры

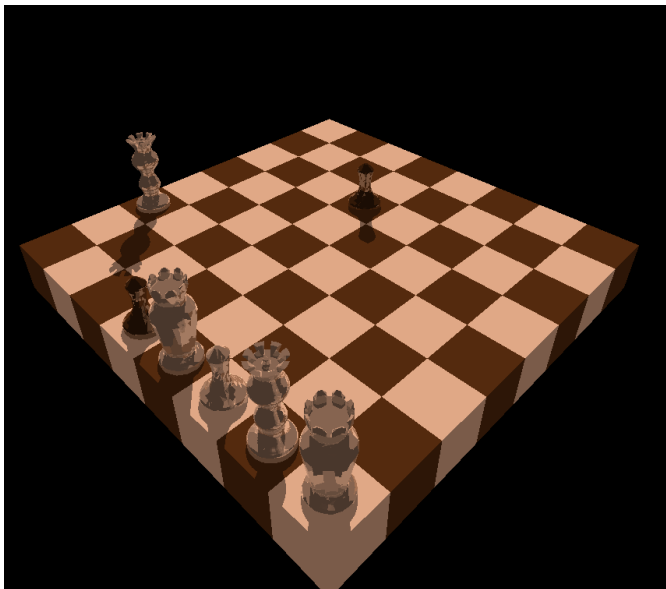


Рисунок 18 — Изначальная сцена

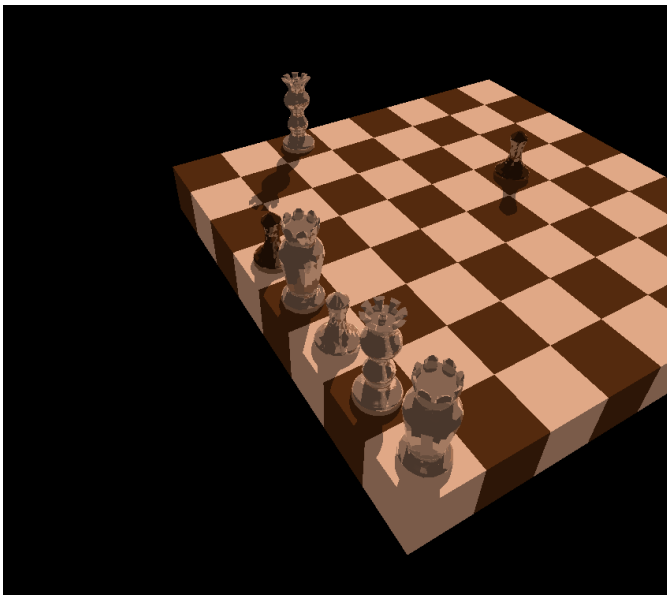


Рисунок 19 — $O_x=10$, $O_y=20$

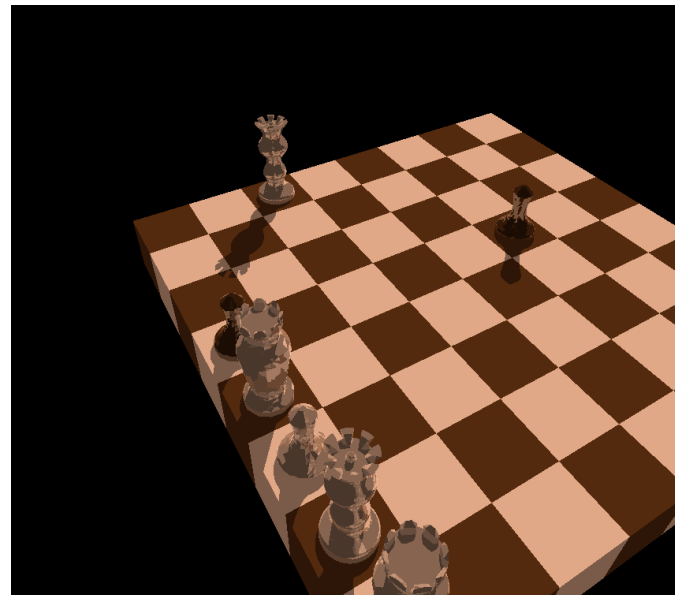


Рисунок 20 — $dx=50$, $dy=50$

Исследование

Цель исследования: сравнение времени выполнения реализации алгоритма обратной трассировки лучей в случае, если коэффициент отражения равен 0, и в случае, если коэффициент отражения не равен 0, от количества шахматных фигур на сцене.

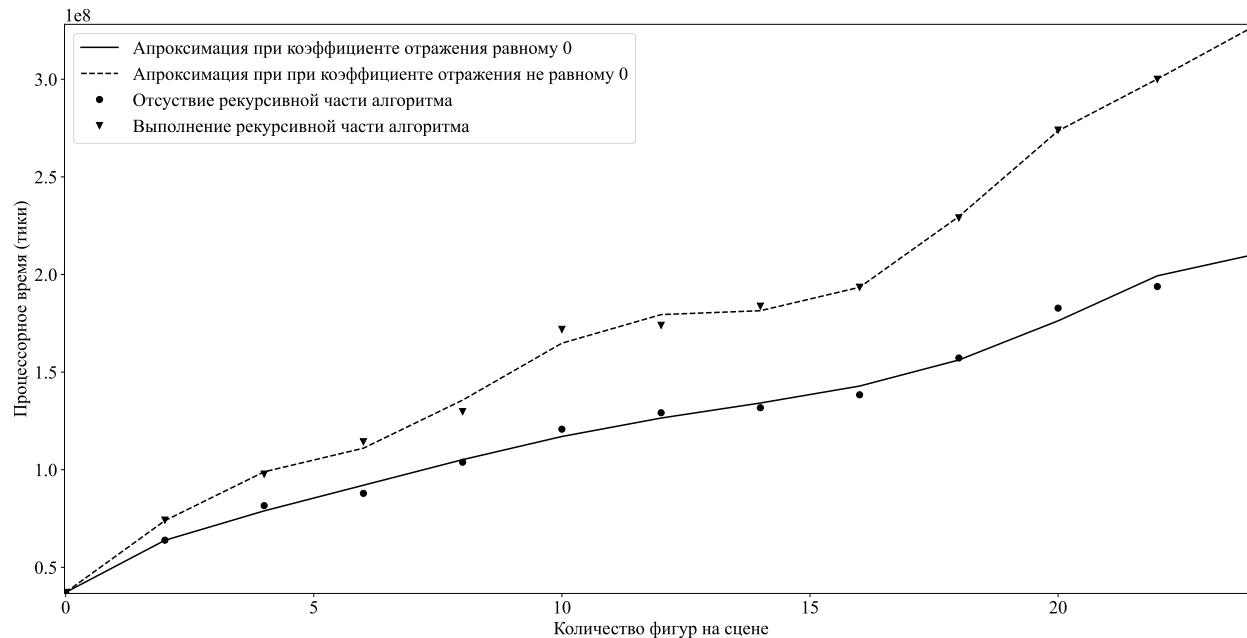


Рисунок 21 — Зависимость времени (в тиках) от количества фигур на шахматной сцене при коэффициенте отражения, равному 0, и коэффициенте отражения, не равному 0

Заключение

В ходе выполнения курсовой работы поставленная цель была достигнута: разработано программное обеспечение с пользовательским интерфейсом, реализовывающего моделирование статической сцены расстановки шахматных фигур на шахматной доске.

В ходе выполнения курсовой работы были решены все задачи.

- проанализирована предметная область, рассмотрены известные подходы и алгоритмы решения задачи синтеза изображения в контексте моделирования статической сцены;
- спроектировано программное обеспечение;
- выбраны средства реализации и разработано программное обеспечение;
- исследованы характеристики разработанного программного обеспечения.