



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:

*«Моделирование изображения объекта в
неотполированном цветном зеркале»*

Студент ИУ7-53Б
(Группа)

(Подпись, дата)

Авдейкина В. П.
(И. О. Фамилия)

Руководитель курсовой работы

(Подпись, дата)

Новик Н. В.
(И. О. Фамилия)

2024 г.

РЕФЕРАТ

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 Аналитическая часть	5
1.1 Формализация задачи	5
1.2 Физическая природа зеркальных поверхностей	6
1.3 Существующие алгоритмы решения задачи синтеза изображения	7
1.3.1 Описание алгоритмов	7
1.3.2 Сравнение алгоритмов	8
2 Конструкторская часть	10
3 Технологическая часть	11
4 Исследовательская часть	12
4.1 Тестовая секция	12
ЗАКЛЮЧЕНИЕ	13
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	15

ВВЕДЕНИЕ

Компьютерная графика в 21 веке не перестала быть развивающейся наукой и упоминается в [1], [2], [3], [4], [5]. Одной из стандартных задач компьютерной графики является синтез изображения [2], [6].

За последние 20 лет визуализация зеркальных поверхностей (зеркал) остается актуальной проблемой, о чем свидетельствует ее обсуждение в [7], [8], [9], [10].

Цель работы — разработка программного обеспечения для моделирования статической сцены с изображением объекта в неотполированном цветном зеркале.

Для достижения поставленной цели требуется решить следующие задачи:

- 1) проанализировать предметную область зеркальных поверхностей, выполнить формализацию задачи синтеза изображения в контексте моделирования статической сцены, рассмотреть известные методы и алгоритмы ее решения;
- 2) спроектировать программное обеспечение;
- 3) выбрать средства реализации и разработать программное обеспечение;
- 4) исследовать характеристики разработанного программного обеспечения.

1 Аналитическая часть

Модель, используемая в работе, требует соблюдения физических законов.

1.1 Формализация задачи

На рисунке 1 представлена диаграмма IDEF0 формализуемой задачи.

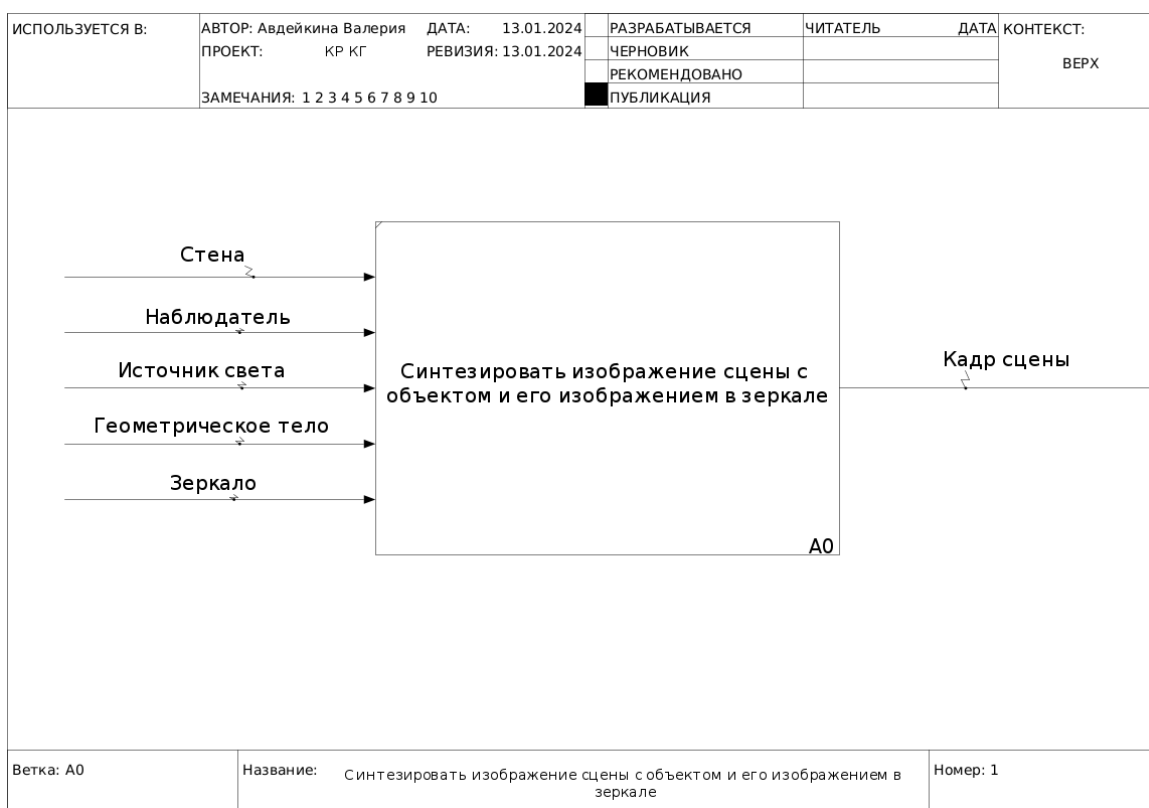


Рисунок 1 — Диаграмма IDEF0

Модель — отображение формы и размеров какой-либо сущности [6]. В данной работе используется поверхностное представление моделей, при котором они представляются аналитически (полностью или частично) [6]. Процесс получения модели называется моделированием.

Сцена состоит из набора сущностей:

- 1) *наблюдатель (камера)*, характеризующийся положением точки наблюдения в пространстве, направлениями взгляда и оси, направленной вверх [6];
- 2) *точечный источник света*, характеризующийся положением в пространстве, интенсивностью и цветом света [6];

- 3) *объект* — модель геометрического тела, геометрические свойства (радиус, высота и так далее) которой определяются заданными пользователем параметрами;
- 4) *зеркало* — модель ограниченной поверхности, обладающая степенью полировки, цветом, радиусом и описанием границ.

1.2 Физическая природа зеркальных поверхностей

Луч (световой) — узкий пучок света, представленный геометрической линией, вдоль которой распространяется свет [11], [12].

Если луч падает на поверхность в точку A и отражается от нее, а через эту точку к поверхности проведена нормаль n , то углы, заключенные между нормалью и направлениями падающего, отраженного лучей (a , b), называются углами *падения*, *отражения* соответственно [11], [12], [13]. Описанные элементы представлены на рисунке 2.

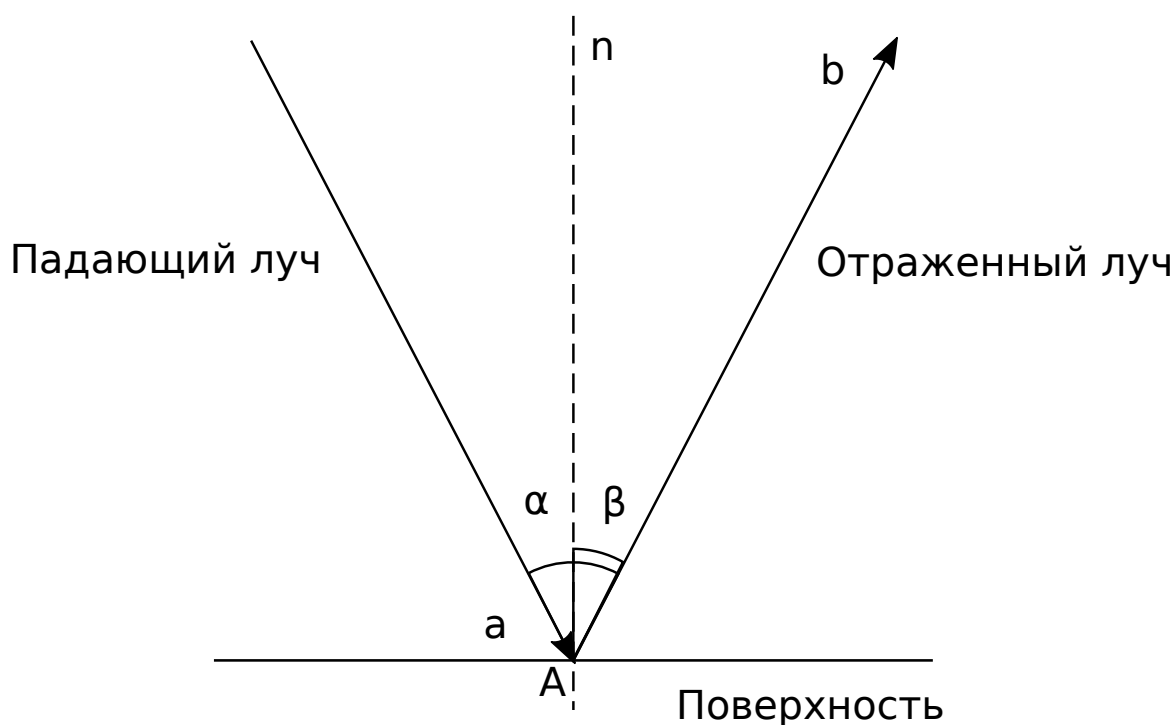


Рисунок 2 — Отражение луча от поверхности

Зеркало — поверхность, от которой отражаются лучи.

Зеркальное отражение — такое, для которого справедлива формула 1.1:

$$\alpha = \beta, \quad (1.1)$$

где α — угол падения, β — угол отражения, лучи и нормаль находятся в одной плоскости [2], [12], [13].

Если от зеркала происходит зеркальное отражение, оно называется идеальным, иначе — реальным [2]. Связь интенсивности отраженного луча I_b с интенсивностью падающего под углом α луча I_a представлена в формулах 1.2 и 1.3 для случаев идеального и реального зеркала соответственно [2], [12]:

$$I_b = I_a \quad (1.2)$$

$$I_b = I_a K_b \cos^p \alpha, \quad (1.3)$$

где K_b — некоторый числовой коэффициент пропорциональности, p — число от 1 до 200 [2].

1.3 Существующие алгоритмы решения задачи синтеза изображения

Среди известных алгоритмов визуализации поверхностей существуют такие алгоритмы, как отслеживание лучей (raycasting), пошаговое распространение лучей (ray marching), прямая и обратная трассировки лучей.

1.3.1 Описание алгоритмов

В соответствии с [14] *алгоритм отслеживания лучей (raycasting)* позволяет обрабатывать поверхностные модели и представляется в виде последовательности шагов (выполняемыми для каждого пикселя экрана):

- 1) провести луч из точки наблюдения;
- 2) рассчитать расстояние от места испускания луча до каждого из объектов сцены;
- 3) определить, для какого из них оно является наименьшим;
- 4) выполнить визуализацию вертикальной «полосы» этого объекта.

Алгоритм пошагового распространения лучей (ray marching) описывается в [15] для каждого луча, испускаемого из точки наблюдения через пиксели экрана, следующими действиями:

- 1) рассчитать значения функции расстояния со знаком до каждого объекта сцены относительно текущей точки начала распространения луча;
- 2) выбрать наименьшее из этих значений;
- 3) если выбранное расстояние меньше некоторой заданной малой величины, запустить рекурсивное распространение лучей из точки объекта в соответствии с оптическими свойствами его поверхности;
- 4) если выбранное расстояние больше некоторой заданной большой величины, визуализировать фон;
- 5) принять за следующую точку начала ту, которая находится на выбранном расстоянии по направлению распространения исходного луча, перейти на шаг 1.

Прямая трассировка лучей заключается

Обратная трассировка лучей представлена авторами [2], [16] с помощью этапов:

- 1) от наблюдателя в каждую точку (пиксель) экрана проводится луч;
- 2) луч достигает объекта, преломляясь или отражаясь, то есть порождает вторичные лучи;
- 3) обработка луча (аналогичная обработке луча в прямой трассировке лучей) прекращается, когда он перестает пересекать объекты сцены.

1.3.2 Сравнение алгоритмов

На основе источников, которые были использованы для приведенного выше описания, осуществлено сравнение анализируемых алгоритмов.

В таблице 1 приведены результаты сравнения, критерии которого расположены по горизонтали и включают в себя:

- 1) сложность алгоритма в зависимости от чисел пикселей C и объектов N ;
- 2) рабочее пространство алгоритма (сцены — «С», экранное — «Э»);
- 3) форма моделей, которая может быть использована при использовании алгоритма (каркасная — «К», поверхностная — «П», объемная — «О»);

- 4) возможность использования глобальной модели освещения;
- 5) принадлежность обрабатываемых точек объекту,
- 6) наличие обработки только тех лучей, которые образуют текущее изображение на экране (без учета вторичных лучей).

Таблица 1 — Сравнение существующих алгоритмов решения задачи синтеза изображения в контексте моделирования статической сцены

	1	2	3	4	5	6
Отслеживание лучей	$O(C \cdot N)$	Э	П	Нет	Да	Да
Пошаговое распространение лучей	$O(C \cdot N)$	Э	О	Да	Нет	Да
Прямая трассировка лучей	$O(C \cdot N)$	Э	О	Да	Да	Нет
Обратная трассировка лучей	$O(C \cdot N)$	Э	О	Да	Да	Да

Вывод

2 Конструкторская часть

бла-бла-схемы-бла

3 Технологическая часть

я люблю си плюс плюс

4 Исследовательская часть

4.1 Тестовая секция

боже царя храни

ЗАКЛЮЧЕНИЕ

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. *Залогова Л. А.* Компьютерная графика // Элективный курс: Практикум. — М., 2005.
2. *Порев В.* Компьютерная графика // СПб.: БХВ-Петербург. — 2002. — Т. 432. — С. 3.
3. *Большаков В. П., Тозик В. Т., В Ч. А.* Инженерная и компьютерная графика. — БХВ-Петербург, 2013. — С. 288.
4. *Митин А. И., Свертилова Н. В.* Компьютерная графика : справочно-методич. пособие. — М.–Берлин : Директ-медиа, 2016. — С. 251.
5. *Турлюн Л. Н.* Компьютерная графика как особый вид современного искусства. — Издательство АлтГУ, 2014.
6. *Куров А. В.* Компьютерная графика : неопубликованный конспект лекций. — 2023.
7. Realistic materials in computer graphics / Н. Р. Lensch [и др.] // ACM SIGGRAPH 2005 Courses. — 2005.
8. *Reshetouski I., Ihrke I.* Mirrors in computer graphics, computer vision and time-of-flight imaging // Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications: Dagstuhl 2012 Seminar on Time-of-Flight Imaging and GCPR 2013 Workshop on Imaging New Modalities. — Springer. 2013. — С. 77—104.
9. *Miguel A. L., Nogueira A. C., Goncalves N.* Real-time 3D visualization of accurate specular reflections in curved mirrors a GPU implementation // 2014 International Conference on Computer Graphics Theory and Applications (GRAPP). — IEEE. 2014. — С. 1—8.
10. Demonstration of light reflection concepts for rendering realistic 3D tree images / Т. Hiranyachattada [и др.] // Journal of Physics: Conference Series. Т. 2145. — IOP Publishing. 2021. — С. 12—74.
11. *Тюрин Ю. И., Чернов И. П., Крючков Ю. Ю.* Физика. Оптика: учебник. — Томск : Издательство Томского политехнического университета, 2009. — С. 240.

12. *Ландсберг Г. С.* Оптика. Учебное пособие: Для вузов. — М : ФИЗМАТ-ЛИТ, 2003. — С. 848.
13. *Родионов С.* Основы оптики // СПб.: СПб ГИТМО (ТУ). — 2000.
14. *Евстратов В. В.* Создание программы визуализации псевдотрехмерного изображения с помощью рейкастинга // Молодой ученый. — 2020. — № 50. — С. 12—15.
15. *Bredenbals A.* Visualising Ray Marching in 3D / Bredenbals Anton. — Groningen : University of Groninge, 2022. — Режим доступа: <https://fse.studenttheses.ub.rug.nl/id/eprint/27977> (дата обращения: 05.12.2023).
16. *Демин А. Ю.* Основы компьютерной графики: учебное пособие. — Томск : Издательство Томского политехнического университета, 2011. — С. 191.