

Моделирование голограммы

Лузгина А.А.*

Ноутбук фирмы "Maxwell equations"

(Дата: 23 мая 2025 г.)

Работа представляет программную модель записи и восстановления голограмм Габора основанную на численном моделировании интерференции и дифракции. Разработаны модули генерации геометрии объекта (сетка точек), расчёта интерференционной картины и восстановления изображения через FFT. Для точечных источников продемонстрирована корректная симуляция: интерференционные паттерны и изменение изображения при смене ракурса. Для объёмных объектов выявлены ошибки, связанные с расчётом фаз, предложено решение через декомпозицию на точечные источники. Результаты подтверждают возможность моделирования голограмм Габора и важность точного учёта фазовых данных.

I. Введение

Голография — это метод записи и воспроизведения трёхмерных изображений объектов. В отличие от традиционных методов получения изображений, которые фиксируют только интенсивность света, голография позволяет записывать как амплитуду, так и фазу световой волны, что делает возможным восстановление полного трёхмерного изображения объекта.

Проект "Симуляция голограммы" направлен на создание программной модели, которая позволяет:

1. Рассчитывать запись голограммы в зависимости от заданной геометрии объекта.
2. Визуализировать восстановленное изображение с учётом положения наблюдателя.

II. Теоретическая справка

II.1. Принципы голографии

Голография основывается на интерференции и дифракции света.

1. Запись голограммы:

Для записи голограммы используют два когерентных световых пучка:

- (a) **Опорный пучок:** когерентный свет, освещающий фотопластинку.
- (b) **Объектный пучок:** свет, отражённый от объекта.

На фотопластинке создаётся интерференционная картина, где записывается амплитудно-фазовая информация об объекте.

2. Восстановление изображения: При освещении голограммы опорным пучком происходит дифракция света, в результате чего наблюдатель видит трёхмерное изображение объекта.

II.2. Интерференция и запись голограммы

Формула интерференции двух когерентных пучков света:

$$I(x, y) = |E_r(x, y) + E_o(x, y)|^2,$$

где $E_r(x, y)$ — вектор опорной волны, $E_o(x, y)$ — вектор волны, отражённой от объекта.

Разложив эту формулу, получим:

$$I(x, y) = |E_r|^2 + |E_o|^2 + 2\text{Re}(E_r E_o^*),$$

где E_o^* — комплексное сопряжение амплитуды объектной волны.

Таким образом, интенсивность света в точке зависит не только от амплитуд, но и от фазовых сдвигов между волнами.

II.3. Восстановление изображения

При восстановлении изображения дифракция света на голограмме формирует три пучка:

1. **Нулевой-дифракционный пучок** (нулевой порядок): содержит лишь интенсивностную информацию.
2. **Пучок +1 порядка:** создаёт изображение объекта.
3. **Пучок -1 порядка:** формирует виртуальное изображение.

II.4. Голограмма Габора

В проекте моделируется голограмма Габора — один из первых методов записи объёмных изображений

* luzgina.aa@phystech.edu;
<https://old.mipt.ru/education/chair/physics/laboratory-praktikum/>

(придуман в 1962г. Юрием Николаевичем Габором). При этом голограмма записывается и восстанавливается с использованием одного и того же источника света.

III. Описание реализации проекта

III.1. Архитектура проекта

Проект состоит из нескольких модулей, каждый из которых отвечает за определённую часть симуляции:

- **Генерация геометрии объекта:** Модуль 'geometry' отвечает за задание формы объекта, разбиение его поверхности на сетку точек и проверку корректности геометрии.
- **Запись голограммы:** Модуль 'hologram_prep' рассчитывает интерференционную картину, создаваемую объектным и опорным пучками, и сохраняет её в виде матрицы интенсивностей.
- **Восстановление изображения:** Модуль 'calculate_visible_field' позволяет восстанавливать видимое изображение объекта с учётом положения наблюдателя.
- **Визуализация:** Используется библиотека OpenGL для отображения восстановленного изображения.

III.2. Этапы работы

1. Генерация геометрии объекта

Объект задаётся в виде набора поверхностей, каждая из которых представлена четырьмя вершинами:

$$\text{vertexes} = \{\{P_1, P_2, P_3, P_4\}, \dots\},$$

где $P_i = (x, y, z)$.

Пример генерации объекта — плоского квадрата:

c++

```
std::vector<std::vector<Point>> vertexes = {
{ {0, 0, 0}, {1, 0, 0}, {0, 1, 0}, {1, 1, 0} }
};
obj_geometry geometry(1e-6, {100, 100},
vertexes);
geometry.calculate_points();
```

2. Запись голограммы

Запись голограммы моделируется путём расчёта интерференционной картины.

```
std::string name_from = "file_with_geometry";
std::string name_to =
"file_where_to_save_intensity.txt";
double width = 10.0;
double height = 10.0;
double scale = 1e-6;
std::vector<unsigned int> np = {1024, 1024};
obj_plate plate(scale, np, width, height);
plate.readDataFromFile(name_from);
plate.calculate_intensity_matrix();
plate.saveIntensityToFile(name_to);
```

3. Восстановление изображения

Восстановление голограммы выполняется методом быстрого преобразования Фурье (FFT). Использовалась библиотека 'fftw3'.

- **FFT** позволяет выделить пучок +1 порядка, содержащий информацию об объекте.
- Результат восстанавливается в виде матрицы интенсивностей, которая затем отображается.

Пример восстановленного изображения:

```
std::string name_from = "file_with_intensity";
v_plate = obj_visible_plate(1e-6);
v_plate.read_intensity_matrix(name_from);
...
while(...){ //main loop
...
v_plate.update_visible_matrix(cameraPos.x,
cameraPos.y, cameraPos.z);
...
}
```

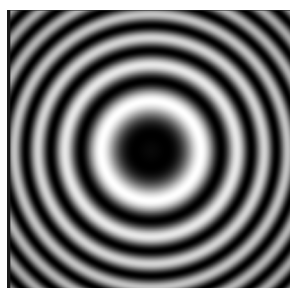
III.3. Результаты

(Получилось мало чего, но мы честно пытались, а еще мысли что делать дальше)

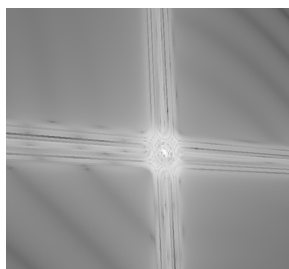
IV. Обсуждение результатов

Скорее всего, моделирование не получилось для объемного источника из-за того, что где-то появилась ошибка в задании фазы.

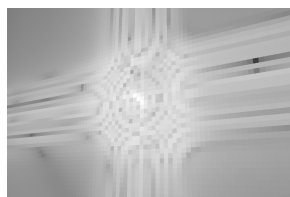
Один из вариантов, для исправления этой ситуации, это решать задачу путем фурье преобразований. Для точечного источника результат известен, а так как объемное изображение можно собрать из множества точек, то таким образом можно так же моделировать голограмму.



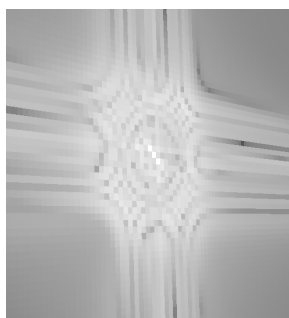
(a)интерференционная картина для точечного источника



(b)



(c)



(d)

Рис. 1. Интерференционная картина, которая бы была на прозранке и изображение точеной голограммы с разных ракурсов (можно заметить, что немного картины интенсивностей, который бы видел человеческий глаз меняются)

V. Итоги и выводы

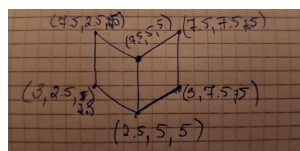
1. Реализована программная модель записи и восстановления голограммы. 2. Используются методы численного моделирования, такие как FFT, для восстановления изображения. 3. Полученные результаты демонстрируют возможность симуляции голограммы Денисюка с учётом положения наблюдателя.

VI. Ссылки на литературу

1. Hecht, E. **Optics**, Fifth Edition. Pearson Education, 2017. 2. Бутиков Е. И. **Оптика: Учебное пособие**. 3-е изд. Санкт-Петербург: Лань, 2016.

VII. Приложения

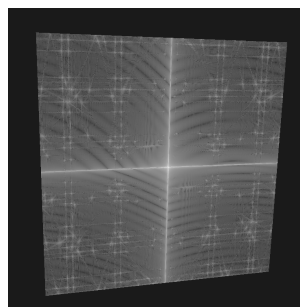
Код проекта расположен по [ссылке](#).



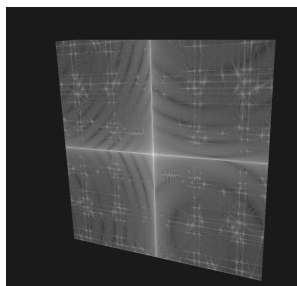
(a) геометрия, рассчитанная в этом варианте



(b) интерференционная картина для точечного источника



(c)



(d)

Рис. 2. Геометрия, интерференционная картина, которая бы была на транспаранте и изображение объемной голограммы с разных ракурсов (можно заметить, что немного картины интенсивностей, который бы видел человеческий глаз меняются)