

Трёхмерная визуализация распространения Wi-Fi сигнала

СТУДЕНТ: ГРИШИН Е.Б.

ГРУППА: ИУ7-55Б

НАУЧНЫЙ РУКОВОДИТЕЛЬ: СТЕПАНОВ В.П.

Цель и задачи работы

Цель данной работы: реализация построения трехмерной сцены и визуализация распространения Wi-Fi сигнала в пространстве.

Задачи:

- описать структуру трехмерной сцены, включая объекты, из которых состоит сцена;
- выбрать и/или модифицировать существующие алгоритмы трехмерной графики, которые позволят визуализировать трехмерную сцену;
- выбрать подходящую модель распространения радиосигнала Wi-Fi;
- реализовать данные алгоритмы для создания трехмерной сцены;
- разработать программное обеспечение, которое позволит отобразить трехмерную сцену и визуализировать распространение Wi-Fi сигнала в пространстве.

Анализ алгоритмов удаления невидимых линий и поверхностей

- Алгоритм обратной трассировки лучей:

Плюсы: высокое качество изображения, реализация сложных оптических явлений.

Минусы: большое количество необходимых вычислений, сложность.

- Алгоритм, использующий Z-буфер:

Плюсы: простота, высокая скорость работы (важно для динамических сцен), возможность учесть особенности задачи.

Минусы: хранение буфера кадра и Z-буфера (незначительно для современных компьютеров), сложность реализации сложных оптических явлений.

- Алгоритм Робертса:

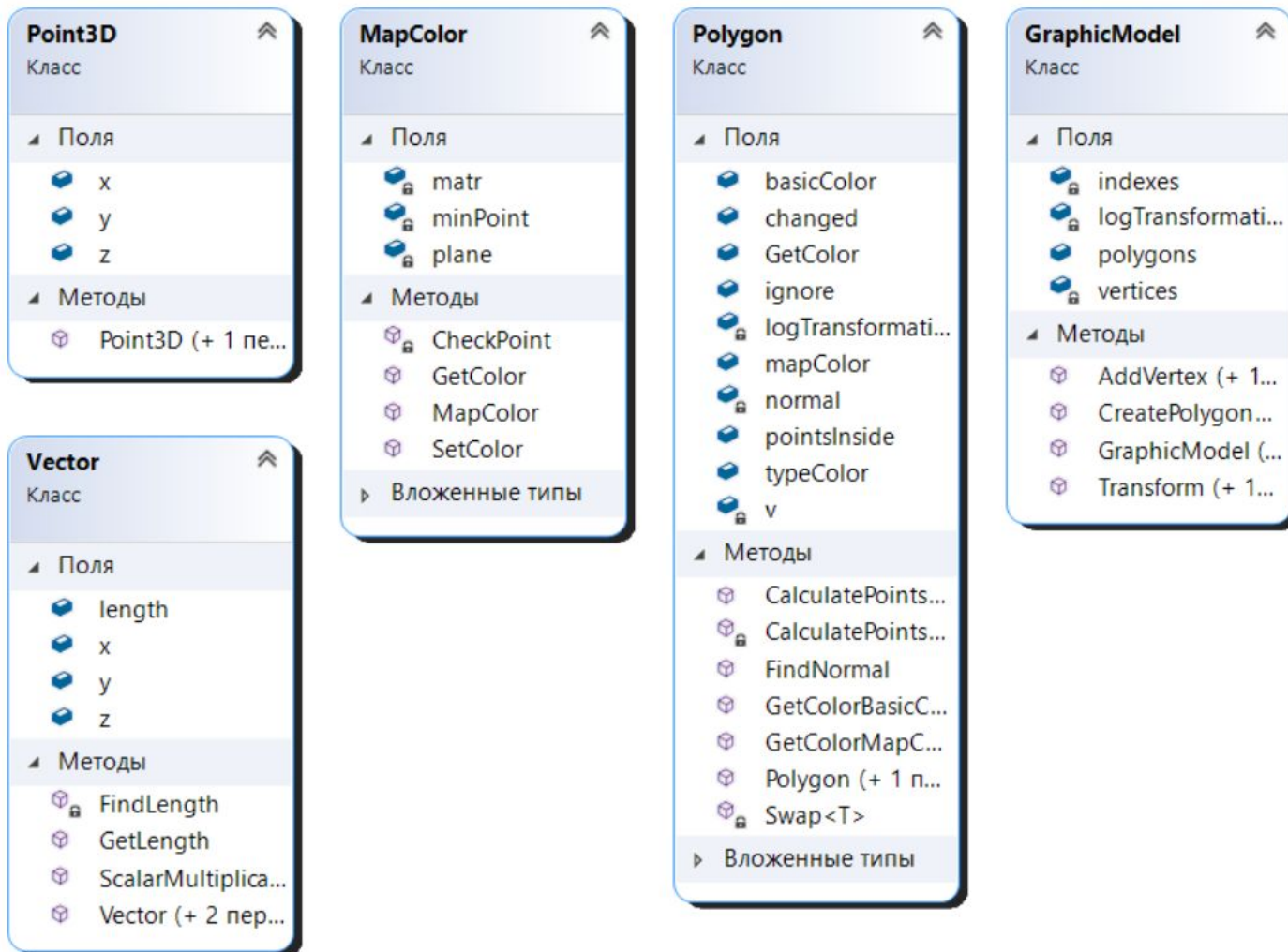
Плюсы: точность вычислений.

Минусы: большое количество вычислений (растет как квадрат количества объектов сцены).

Алгоритм Z-буфера

1. Всем элементам буфера кадра присвоить фоновое значение
2. Инициализировать Z-буфер минимальными значениями глубины
3. Выполнить растровую развертку каждого полигона сцены:
 - 3.1. Для каждого пикселя, связанного с полигоном вычислить его глубину $z(x, y)$
 - 3.2. Получить цвет точки полигона с координатами (x, y, z) - colorPoint
 - 3.3. Если $z(x, y) > zBuf(x, y)$:
 - 3.3.1. $zBuf(x, y) = z(x, y)$
 - 3.3.2. $color(x, y) = colorPoint$
4. Отобразить результат

Структура и состав классов



Структура и состав классов

Scene
Класс

- Поля
 - ground
 - logTransformati...
 - models
- Методы
 - Count
 - CreateCube
 - CreateCubeOn...
 - CreateGround
 - CreateSpecialSl...
 - Remove
 - Scene (+ 1 пер...
 - Transform (+ 1...

ZBuffer
Класс

- Поля
 - img
 - imgSaved
 - shiftXFromSun
 - shiftYFromSun
 - sizeFromsSun
 - sizeScreen
 - zBackground
 - Zbuf
 - ZbufFromSun
 - ZbufSaved
- Методы
 - CopyBuf
 - FillBuf
 - GetImage
 - InitBuf
 - ProcessModel
 - ProcessModelF...
 - ProcessScene
 - ProcessSceneW...
 - Save
 - ZBuffer


WifiModel
Класс


- Поля
 - antennaX
 - antennaY
 - antennaZ
 - arrMapColor
 - lenX
 - lenY
 - lenZ
 - mapBarrier
 - mapCountBarri...
 - maxPowerLoss
 - minX
 - minY
 - minZ
 - powerLoss
- Методы
 - AddBarrier
 - GetMap
 - ProcessArrMap...
 - ProcessMapCo...
 - ProcessModel
 - ProcessPowerL...
 - RemoveAllBarri...
 - SendingRayMa...
 - SetAntenna
 - WifiModel


Simulation
Класс


- Поля
- Методы
 - AddBarrier
 - ChangeOptimis...
 - ChangeShadows
 - ChangeShowIn...
 - LightSourceLeft
 - LightSourceRight
 - LightSourceTop
 - MoveIndicatorS...
 - MoveIndicatorS...
 - RedrawScene
 - RemoveAllBarri...
 - RotateLeft
 - RotateRight
 - SetAntenna
 - Simulation
 - StartMovingInd...


Структура и состав классов


Transformation 
Static Класс


 Поля


 centerX


 centerY


 Методы


 RotateX (+ 1 пе...


 RotateY (+ 1 пе...


 RotateZ (+ 1 пе...


 SetCenter


 Transform (+ 2...


 Вложенные типы


WritingTransfor... 
Struct


 Поля


 costeta


 sinteta


 teta


 tetaRadian


 type


 Методы


 WritingTransfor...


LogTransformat... 
Класс


 Поля


 log


 reverseLog


 Свойства


 this

 Методы

 Add (+ 1 перер...

 GetLength

 GetReverse

 LogTransformat...

Структура и состав классов

Colors
Static Класс

Методы

Mix

LightSource
Класс

Поля

color

direction

tetax

tetay

tetaz

Методы

LightSource

Matrix<T>
Универсальный Класс

Поля

m

mass

n

Свойства

M

N

this

Методы

Matrix (+ 1 пер...

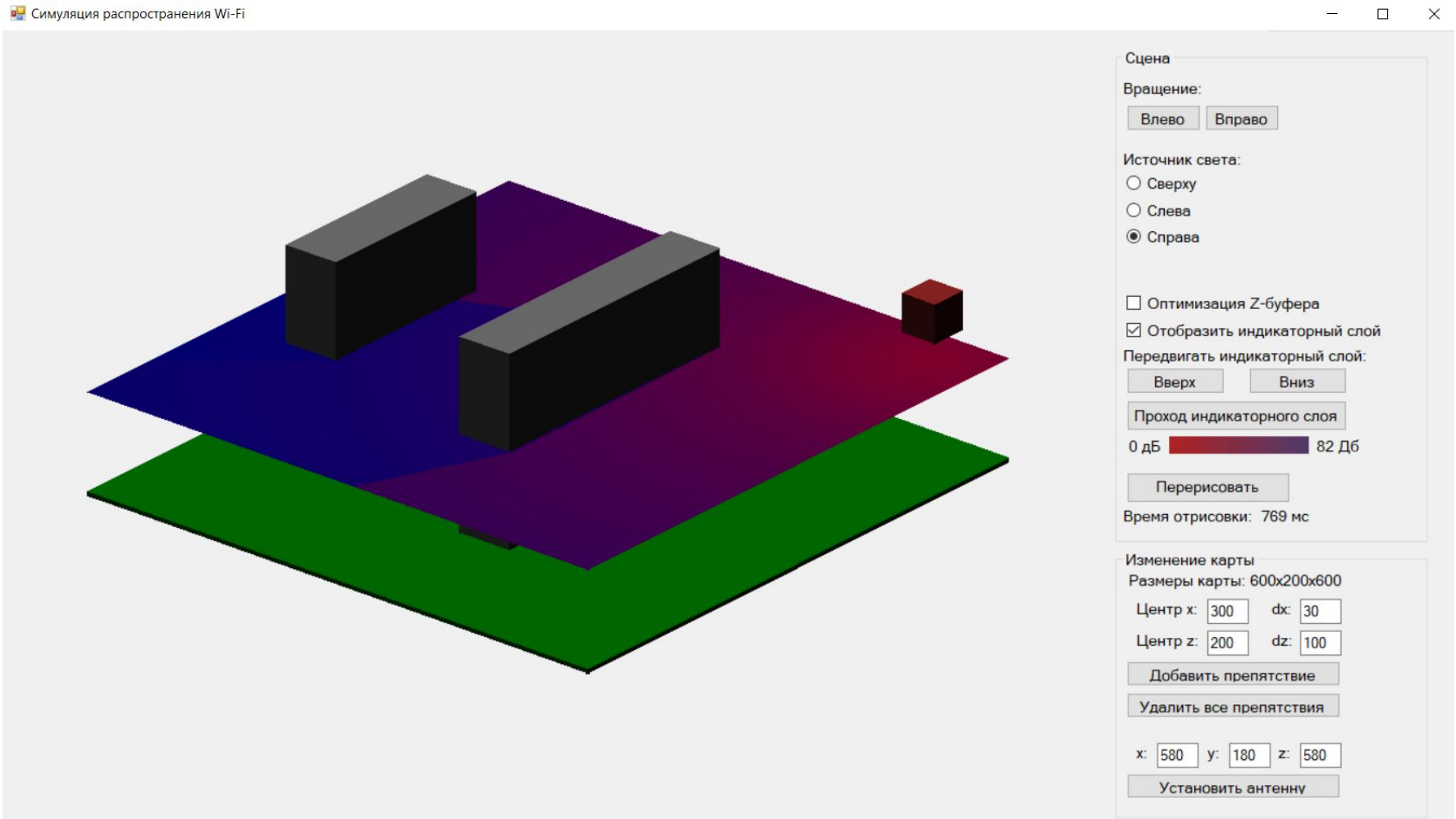
Form1
Класс
→ Form

Program
Static Класс

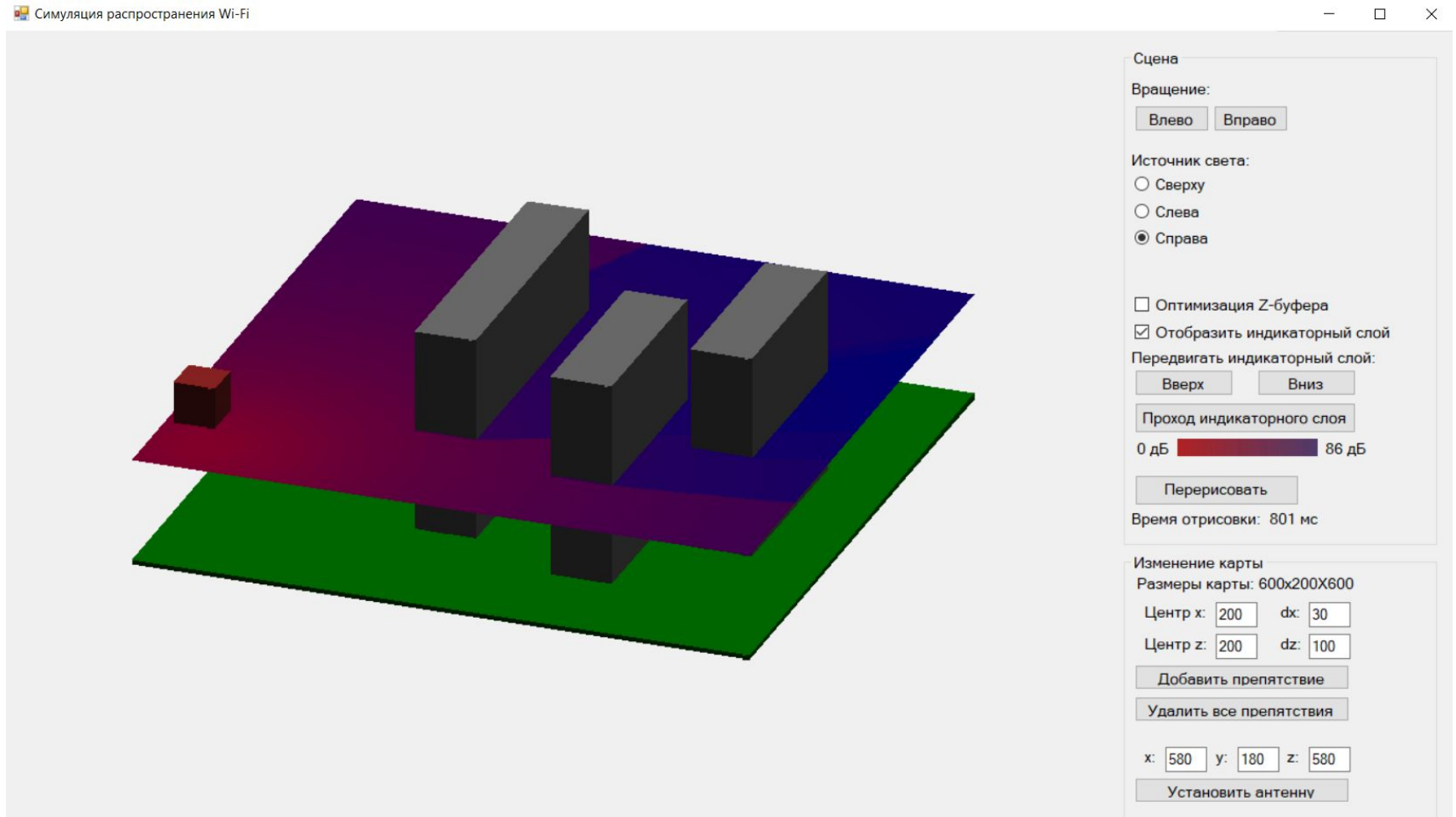
Методы

Main

Пример работы



Пример работы



Эксперимент

Характеристики компьютера:

- Операционная система Windows 10 64-bit;
- Процессор AMD Ryzen 5 3500U CPU @ 2.1GHz;
- 8 ядер;
- 20 Гб оперативной памяти.

Результаты эксперимента:

- 554 мс – обычный алгоритм, 1 препятствие;
- 248 мс – оптимизированный алгоритм, 1 препятствие;
- 618 мс – обычный алгоритм, 2 препятствия;
- 239 мс – оптимизированный алгоритм, 2 препятствия;
- 646 мс – обычный алгоритм, 3 препятствия;
- 242 мс – оптимизированный алгоритм, 3 препятствия.

Заключение

Во время выполнения курсового проекта была описана структура трехмерной сцены, были рассмотрены основные алгоритмы удаления невидимых линий, методы закрашивания, модели распространения радиосигнала. Были проанализированы их достоинства и недостатки, выбраны и реализованы наиболее подходящие для решения поставленной задачи. Было разработано программное обеспечение для визуализации распространения Wi-Fi сигнала.

Программа реализована таким образом, что пользователь может удалять препятствия, добавлять новые, изменять положение антенны, источника света.

В ходе выполнения поставленной задачи были изучены возможности Windows Forms, получены знания в области компьютерной графики.